



**UNIVERSIDADE DO SUL DE SANTA CATARINA**  
**LUCIANO LUIZ DA SILVA**

**DESENVOLVIMENTO DE UMA SOLUÇÃO PARA AUTOMAÇÃO  
RESIDENCIAL USANDO A PLATAFORMA ANDROID E ARDUINO**

Palhoça  
2013

**LUCIANO LUIZ DA SILVA**

**DESENVOLVIMENTO DE UMA SOLUÇÃO PARA AUTOMAÇÃO  
RESIDENCIAL USANDO A PLATAFORMA ANDROID E ARDUINO**

Trabalho de Conclusão de Curso apresentado ao  
Curso de Graduação em Ciência da Computação  
da Universidade do Sul de Santa Catarina, como  
requisito parcial à obtenção do título de Bacharel  
em Ciência da Computação.

Prof. e Orientadora Maria Inés Castiñeira, Dra.

Palhoça

2013

**LUCIANO LUIZ DA SILVA**

**DESENVOLVIMENTO DE UMA SOLUÇÃO PARA AUTOMAÇÃO  
RESIDENCIAL USANDO A PLATAFORMA ANDROID E ARDUINO**

Este Trabalho de Conclusão de Curso foi julgado adequado à obtenção do título de Bacharel em Ciência da Computação e aprovado em sua forma final pelo Curso de Graduação em Ciência da Computação da Universidade do Sul de Santa Catarina.

Palhoça, \_\_\_\_ de \_\_\_\_\_ de \_\_\_\_\_.

---

Professora e Orientadora Maria Inés Castiñeira, Dra.  
Universidade do Sul de Santa Catarina

---

Prof. Jean Carlo Rossa Hauck, Dr.  
Universidade do Sul de Santa Catarina

---

Prof. Robinson Pizzio, M.Eng.  
Universidade do Sul de Santa Catarina

Dedico este trabalho a minha esposa, Elisandra Salete Valentim e ao meu filho, Rafael Valentim da Silva, os quais amo muito, que mesmos nos momentos em que nos privamos da companhia, estiveram torcendo, me dando amor e carinho, para que eu pudesse vencer mais uma etapa em minha vida.

Aos meus irmãos, Douglas Luiz e Fabiana da Silva. Aos meus pais, Jorge Luiz e Maria Albertina, que apesar das dificuldades encontradas, puderam superá-las, dando educação e me fortalecendo com ensinamentos, para que hoje, eu me tornasse o Homem que sou.

## **AGRADECIMENTOS**

A Universidade do Sul de Santa Catarina.

A professora, orientadora e amiga, Dra. Maria Inés Castiñeira, que esteve junto, durante um dos trajetos mais importantes de minha vida.

A todos os funcionários do Hospital Universitário de Florianópolis, que estiverem ao meu lado, durante uma fase muito triste, me ajudando e de certa forma, fazendo com que eu não perdesse as esperanças de continuar vivo, para concluir o que há um tempo eu tinha começado, minha graduação no curso de Ciência da Computação.

Um agradecimento especial à Dra Giovanna Steffenello, Médica Hematologista, que ainda me acompanha, ao setor de coleta de sangue e todo o setor de Quimioterapia do Hospital Universitário de Florianópolis.

A todos os professores, colegas e amigos pelos ensinamentos e experiências compartilhadas. Um agradecimento especial ao Professor e Amigo Marcelo Medeiros.

Aos meus pais, irmãos, cunhados, ao meu sogro (João Feliciano) e sogra (Maria Alves) por serem além de família, AMIGOS.

Aos meus amigos Angela Maria Diniz e Osmar Bento Soares pela enorme amizade que nos cerca, e que, para mim, são mais que grandes Amigos.

Nenhuma palavra aqui seria suficiente para expressar o amor e carinho que tenho pela minha esposa e pelo meu filho.

Há 12 anos e alguns meses, nos votos de casamento, a minha esposa prometeu estar ao meu lado, na alegria e na tristeza, na saúde e na doença. Não foi diferente!

No período de doença e tristeza, durante toda a fase de tratamento contra a Leucemia, grávida, ficou ao meu lado, em um quarto de hospital sem nenhum conforto, cuidando e zelando pela minha saúde. Hoje, com saúde, muita alegria e com ela ao meu lado, estou aqui, com esse trabalho concluído, podendo agradecer e dizer mais uma vez que a AMO!

Um agradecimento muito especial ao meu Filho, que é a minha maior fonte de riqueza e inspiração, e que, atualmente, apesar de ter 4 anos, me incentivou para que eu concluísse este trabalho, e que, ainda me incentiva, para que o meu caminhar não se encerre por aqui.

Durante o curso de Ciência da Computação, uma Leucemia chegou sem ser convidada. Tranquei a faculdade durante 1 ano, em 2009, para que eu, passasse pelo tratamento apropriado, porém, este contratempo não foi motivo para que eu interrompesse meu caminhar. Hoje estou aqui, com saúde e com o tão esperado Trabalho de Conclusão de Curso concluído, significando para mim uma grande Vitória!

A todos, do fundo do meu coração, os meus enormes Agradecimentos!

## **RESUMO**

Nos últimos anos, os sistemas de automação residencial vêm crescendo a um ritmo bem acelerado. Esses sistemas estão apresentando maior diversidade de produtos e de serviços, com preços cada vez mais acessíveis, despertando assim o interesse tanto de consumidores quanto de empreendedores. Automação residencial consiste em sistemas utilizados para o controle de equipamentos eletroeletrônicos encontrados em residências, facilitando dessa forma o dia a dia das pessoas, além de proporcionar conforto, agilidade e economia com os recursos utilizados, como luz e água. Para as pessoas portadoras de necessidades especiais não só proporcionam o bem estar mas também geram plena autonomia. Com base nesse contexto, o trabalho apresenta a especificação e o desenvolvimento de um protótipo para automação residencial de baixo custo, controlado a distância via comunicação sem fio, utilizando a tecnologia Bluetooth. Este protótipo é responsável por controlar os principais dispositivos de uma residência, como a parte de iluminação, tomadas, portão, televisão e alarme, utilizando como meio de interface gráfica para o usuário final a plataforma Android, disponível hoje em tablets e celulares. O protótipo utiliza como hardware a plataforma Arduino responsável pelo controle eletrônico dos dispositivos automatizados e pelo monitoramento dos sensores. Com o objetivo de testar e validar as funcionalidades de automação desenvolvidas, foi criada uma maquete residencial. Os resultados finais obtidos com os testes aplicados à maquete mostraram que o protótipo desenvolvido atingiu seus objetivos.

Palavras-chave: Android, Arduino. Bluetooth. Automação Residencial. Domótica.

## LISTA DE FIGURAS

Figura 1 – Leitor Biométrico .....	22
Figura 2 – Câmera de vigilância.....	22
Figura 3 – Controle de Iluminação via iPhone .....	23
Figura 4 – Controlador Central.....	24
Figura 5 – Esquemático do sistema de automação residencial.....	24
Figura 6 – Interface de Configuração do Miniserver. ....	25
Figura 7 – Telas do sistema Loxone para Android.....	26
Figura 8 – Interface do sistema Iluflex para iPad e iPhone .....	27
Figura 9 – Módulo Receptor de RF. ....	27
Figura 10 – Pulsador.....	28
Figura 11 – Módulo Wi-fi – IR .....	28
Figura 12 – Módulo Receptor RF para Portão. ....	29
Figura 13 – Interface Ethernet RF .....	29
Figura 14 – Arquitetura do Sistema Iluflex. ....	30
Figura 15 – Conceito de Automação Residencial. ....	32
Figura 16 – Topologia do sistema X-10. ....	33
Figura 17 – Diagrama do sistema UPB .....	35
Figura 18 – Estrutura Centralizada.....	39
Figura 19 – Estrutura Distribuída. ....	40
Figura 20 – Arquitetura Mista. ....	40
Figura 21 – Estrutura do Cabo Coaxial. ....	43
Figura 22 – Cabo UTP sem Blindagem.....	44
Figura 23 – Cabo UTP com Blindagem.. ....	45
Figura 24 – As camadas da Fibra. ....	46
Figura 25 – Velocidade de Transmissão.....	47
Figura 26 – Categorias de Redes sem Fio. ....	48
Figura 27 – Aplicação do protocolo ZigBee.. ....	51
Figura 28 – Camadas do protocolo ZigBee. ....	52
Figura 29 – Rede ZigBee.....	54



Figura 30 – Topologias rede ZigBee .....	55
Figura 31 – Exemplo de uma rede Can. ....	56
Figura 32 – Frame Standard Can. ....	58
Figura 33 –Estrutura de comunicação Bluetooth .....	60
Figura 34 – Dispositivos de um Sistema de Automação .....	61
Figura 35 – Exemplos de Interface. ....	62
Figura 36 – Protótipo do Arduino.....	65
Figura 37 – Arduino Uno. ....	65
Figura 38 – Arduino Nano.....	66
Figura 39- Arduino Lilypad.....	66
Figura 40 – Mega 2560.....	67
Figura 41 – Shield Ethernet .....	68
Figura 42 – Shield XBee. ....	68
Figura 43 – Tela da IDE. ....	69
Figura 44 – Cenário da computação móvel. ....	70
Figura 45 – Exemplos de dispositivos móveis. ....	71
Figura 46 – Logotipo do Android.....	73
Figura 47 – T-Mobile G1. ....	73
Figura 48 – Arquitetura da plataforma Android.....	75
Figura 49 – Componentes da Plataforma Android. ....	77
Figura 50 – Emulador simulando uma aplicação .....	79
Figura 51 - Arquitetura macro do projeto.....	85
Figura 52 – Ator do Sistema.....	88
Figura 53 – Diagrama de caso de uso.....	89
Figura 54 – Diagrama de classe do Controle do Portão Eletrônico.....	95
Figura 55 – Diagrama de classe do Controle de Televisão.. ....	96
Figura 56 – Diagrama de classe do Controle de Iluminação.....	97
Figura 57 – Diagrama de classe do Controle e Alarme.....	98
Figura 58 – Modelo para acionamento da Lâmpada Manual ou por Horário.. ....	100
Figura 59 – Modelo para acionamento da Lâmpada por Sensor... ....	101
Figura 60 – Modelo para acionamento Manual ou por Horário da Tomada.. ....	102
Figura 61 – Modelo para acionamento da Tomada por Sensor.....	103
Figura 62 – Modelo para acionamento do Portão.....	104
Figura 63 – Modelo para acionamento do Alarme... ..	105

Figura 64 – Modelo para acionamento da Televisão.....	106
Figura 65 – Primeiros componentes recebidos.....	109
Figura 66 – Tablet Genesis GT - 7240.....	109
Figura 67 – Características do Tablet Genesis GT – 7240.....	110
Figura 69 – Visão Macro do ambiente de desenvolvimento de Hardware.....	113
Figura 69 – Maquete Residencial Semi Acabada.....	114
Figura 70 – Tecnologias e Ferramentas utilizadas no projeto.....	114
Figura 71 – Protocolo de Comunicação – Ações. ....	120
Figura 72 – Protocolo de Comunicação – Configurações.. ....	122
Figura 73 – Arquitetura MVC.. ....	123
Figura 74 – Módulo Relé.....	125
Figura 75 – Módulo Sensor.. ....	126
Figura 76 – Módulo Controle Remoto.... ....	126
Figura 77 – Servo Motor .....	127
Figura 78 – Módulo Motor de Passo L298N.....	128
Figura 79 – Módulo Driver Motor de Passo L298N. ....	128
Figura 80 – Protoboard.....	129
Figura 81 – Módulo Sirene.....	129
Figura 82 – Módulo Bluetooth... ..	130
Figura 83 – Módulo RTC.. ....	130
Figura 84 – Modelo da Maquete. ....	132
Figura 85 – Maquete Residencial.. ....	132
Figura 86 – Maquete Residencial com seus pontos de Iluminação. ....	133
Figura 87 – Maquete Residencial com as Tomadas.. ....	133
Figura 88 – Tela Principal. ....	134
Figura 89 – Opções de Configurações.....	135
Figura 90 – Cômodos para configuração da Iluminação. ....	136
Figura 91 – Modo de Acionamento Horário. ....	137
Figura 92 – Botão Confirmar.....	138
Figura 93 – Botão Cancelar.....	138
Figura 94 – Modo de Acionamento Sensor.. ....	139
Figura 95 – Modo de Acionamento Manual.....	140
Figura 96 – Opção de Portões para Configuração.....	141
Figura 97 – Configuração para portão.....	142

Figura 98 – Configurações para Tomadas.....	143
Figura 99 – Configuração para Alarme.....	144
Figura 100 – Controle da Iluminação – Lâmpada Desligada... ..	146
Figura 101 – Controle da Iluminação – Lâmpada Ligada.....	147
Figura 102 – Controle do Portão – Portão Fechado... ..	148
Figura 103 – Controle do Portão – Portão Aberto.....	149
Figura 104 – Controle do Alarme – Alarme Desativado.....	150
Figura 105 – Controle do Alarme – Alarme Ativado.....	151
Figura 106 – Controle da Televisão.....	152
Figura 107 – Controle da Tomada.....	153

## LISTA DE QUADROS

Quadro 1 – Principais APIs .....	76
Quadro 2 – Tipos de pesquisa.....	81
Quadro 3 – Fontes Bibliográficas.....	83
Quadro 4 – Requisitos Funcionais.....	90
Quadro 5 – Requisitos Não Funcionais .....	91
Quadro 6 – Regras de Negócio.....	92

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO.....</b>	<b>16</b>
1.1	PROBLEMÁTICA .....	17
1.2	OBJETIVOS .....	18
1.2.1	Objetivo Geral .....	18
1.2.2	Objetivos Específicos.....	18
1.3	JUSTIFICATIVA .....	19
1.4	ESTRUTURA DO TRABALHO .....	20
<b>2</b>	<b>REVISÃO BIBLIOGRÁFICA .....</b>	<b>21</b>
2.1	ESTADO DA ARTE .....	21
2.1.1	Intelicasa .....	21
2.1.2	Loxone .....	23
2.1.3	Iluflex.....	26
2.1.4	Considerações Finais Sobre as Soluções Pesquisadas .....	31
2.2	AUTOMAÇÃO .....	31
2.3	AUTOMAÇÃO RESIDENCIAL .....	32
2.3.1	Histórico .....	33
2.3.2	Características .....	36
2.3.3	Domótica .....	37
2.3.4	Estrutura dos sistemas de automação .....	39
2.4	INTELIGÊNCIA ARTIFICIAL .....	41
2.5	MEIOS DE COMUNICAÇÃO .....	41
2.5.1	Comunicação cabeada.....	42
2.5.1.1	Cabo Coaxial .....	43
2.5.1.2	Cabo de Par Trançado .....	44
2.5.1.3	Fibra Ótica .....	45
2.5.2	Comunicação sem fio .....	47
2.6	PROTOCOLOS .....	49
2.6.1	ZigBee.....	51
2.6.1.1	Dispositivos da rede ZigBee.....	53
2.6.1.2	Topologias ZigBee .....	54

<b>2.6.2 Cebus .....</b>	<b>55</b>
<b>2.6.3 Can.....</b>	<b>56</b>
2.7 DISPOSITIVOS DO SISTEMA .....	61
2.8 SISTEMAS EMBARCADOS .....	63
<b>2.8.1 Arduino .....</b>	<b>64</b>
2.8.1.1 Acessórios.....	67
2.8.1.2 Software.....	69
2.9 DISPOSITIVOS MÓVEIS .....	70
<b>2.9.1 Desenvolvimento para Dispositivos móveis .....</b>	<b>72</b>
<b>2.9.2 Android .....</b>	<b>72</b>
2.9.2.1 Arquitetura.....	74
2.9.2.2 Componentes de uma Aplicação .....	76
2.9.2.3 Desenvolvimento para Android.....	78
2.9.2.4 Máquina Virtual.....	79
<b>3 MÉTODO DE PESQUISA.....</b>	<b>80</b>
3.1 CARACTERIZAÇÃO DO TIPO DE PESQUISA.....	80
3.2 ETAPAS .....	83
3.3 ARQUITETURA DA SOLUÇÃO PROPOSTA .....	84
3.4 DELIMITAÇÕES.....	86
<b>4 MODELAGEM.....</b>	<b>87</b>
4.1 SOFTWARE.....	87
<b>4.1.1 Uml – Linguagem de Modelagem Unificada.....</b>	<b>87</b>
<b>4.1.2 Atores.....</b>	<b>87</b>
<b>4.1.3 Casos de Uso .....</b>	<b>88</b>
<b>4.1.4 Requisitos .....</b>	<b>89</b>
4.1.4.1 Requisitos Funcionais.....	90
4.1.4.2 Requisitos Não Funcionais .....	91
<b>4.1.5 Regras de Negócio .....</b>	<b>91</b>
<b>4.1.6 Diagrama de Classe.....</b>	<b>94</b>
4.2 HARDWARE .....	99
<b>4.2.1 Acionamento da Lâmpada .....</b>	<b>99</b>
<b>4.2.2 Acionamento da Tomada.....</b>	<b>101</b>
<b>4.2.3 Acionamento do Portão .....</b>	<b>103</b>
<b>4.2.4 Acionamento do Alarme .....</b>	<b>104</b>

4.2.4	Acionamento da Televisão .....	105
5	DESENVOLVIMENTO .....	107
5.1	HISTÓRICO DO DESENVOLVIMENTO .....	107
5.1.1	Modelagem.....	107
5.1.2	Aquisição de Componentes.....	108
5.1.3	Desenvolvimento para Android .....	110
5.1.4	Protocolo de Comunicação entre Android Arduino .....	111
5.1.5	Desenvolvimento para Arduino .....	111
5.1.5.1	Caraerísticas do Arduíno Mega .....	112
5.1.6	Hardware .....	112
5.1.7	Maquete.....	113
5.2	TECNOLOGIAS E FERRAMENTAS UTILIZADAS.....	114
5.3	PROTOCOLO DE COMUNICAÇÃO.....	118
5.3.1	Tratamento do protocolo para Ações .....	118
5.3.2	Tratamento do protocolo para Configurações .....	120
5.4	DESENVOLVIMENTO ANDROID .....	122
5.5	DESENVOLVIMENTO ARDUINO .....	124
5.6	DESENVOLVIMENTO DE HARDWARE .....	125
5.7	MAQUETE.....	131
5.8	APRESENTAÇÃO DO SISTEMA.....	134
5.8.1	Tela Principal.....	134
5.8.2	Tela de Configurações.....	135
5.8.3	Configurações da Iluminação.....	136
5.8.4	Modo de Acionamento Horário .....	137
5.8.5	Modo de Acionamento Sensor.....	138
5.8.6	Modo de Acionamento Manual.....	139
5.8.7	Configuração do Portão.....	140
5.8.8	Tempo de Fechamento .....	141
5.8.9	Configuração da Tomada .....	142
5.8.10	Configuração do Alarme .....	143
5.8.11	Considerações sobre as Configurações.....	144
5.8.12	Controle da Iluminação .....	145
5.8.13	Controle do Portão .....	147
5.8.14	Controle do Alarme.....	149

<b>5.8.15 Controle da Televisão.....</b>	<b>151</b>
<b>5.8.16 Controle da Tomada .....</b>	<b>152</b>
<b>5.9 AVALIAÇÃO .....</b>	<b>153</b>
<b>6 CONCLUSÕES E TRABALHOS FUTUROS.....</b>	<b>155</b>
6.1 CONCLUSÕES .....	155
6.2 DIFERENCIAL DO TRABALHO .....	158
6.3 TRABALHOS FUTUROS .....	158
<b>REFERÊNCIAS .....</b>	<b>160</b>
<b>APÊNDICES .....</b>	<b>167</b>
<b>APÊNDICE A - CRONOGRAMA.....</b>	<b>168</b>



## 1 - INTRODUÇÃO

Devido ao crescimento acelerado e desordenado das cidades, as pessoas vêm pagando um alto preço por um mínimo de qualidade de vida e bem estar no seu cotidiano. Paralelo a essa transformação a tecnologia vem se tornando uma grande aliada dentro do papel social, não somente por prover conforto, mas por ampliar e melhorar as interações dos usuários com o mundo que os cerca. (BOLZANI, 2004).

Uma das aplicações da tecnologia pode ser na automação residencial.

Em uma de suas afirmações, Nunes (2002) define que um dos objetivos da automação residencial é oferecer maior comodidade para as pessoas, melhorando a qualidade de vida, reduzindo os trabalhos domésticos, aumentando o bem estar e dando maior segurança ao patrimônio. Outro aspecto positivo é a contribuição para uma vida mais sustentável, fazendo uso racional de energia, água e uma melhor gestão de outros recursos utilizados em habitação. A Automação Residencial proporciona ótimas soluções de acessibilidade no que diz respeito a pessoas com necessidade especiais.

Situações que antigamente pareciam ficção científica hoje são realidades. Já é possível executar grandes tarefas a partir de simples toques, ligar ar condicionado, programar alarmes, fechar cortinas, controlar iluminação, entre outros recursos.

A casa inteligente é aquela que está dotada de um controlador central responsável por manipular e gerenciar todos os periféricos relacionados à residência, juntamente com uma interface de comunicação para o usuário. Este controlador central é o hardware do sistema, denominado de sistema embarcado, composto principalmente por um microcontrolador. (EUZÉBIO et al., 2011).

O microcontrolador é um componente que possui processador, memória e interfaces de entrada e saída, responsável por desempenhar funções específicas, como a parte de controle, mas que não possuem interface com o usuário (TANENBAUM, 2007).

Uma plataforma de hardware bastante conhecida relacionada ao conceito de sistema embarcado é o Arduino. Um dos seus criadores, Banzi (2008) a descreve como sendo uma plataforma para desenvolvimento ágil, ajudando na prototipação de projetos eletrônicos.

Em questões de plataforma para o desenvolvimento de Interface com o usuário, em se tratando de dispositivos móveis, Android é uma boa opção de escolha, por ser

desenvolvida em software livre e código aberto, podendo ser integrada com outras aplicações.

Para Lecheta (2010), Android é a nova plataforma de desenvolvimento para aplicativos móveis. Contém seu sistema operacional baseado em Linux, uma interface visual rica, GPS (*Global Positioning System*), diversas aplicações instaladas e, ainda, um poderoso ambiente de desenvolvimento, além de ser flexível e inovador. Estas aplicações são escritas em linguagem Java, por esses motivos, esta plataforma vem sendo utilizada cada vez mais para o uso comercial.

Este trabalho tem como tema o desenvolvimento de um sistema envolvendo Software e Hardware para fazer o controle de Automação Residencial. O Aplicativo faz uso da plataforma Android como interface para controle do hardware baseado no sistema embarcado Arduino.

## 1.1 PROBLEMÁTICA

Vários fatores atualmente vêm contribuindo para que a questão da automização residencial venha a ser considerada não apenas item de luxo, mas como parte de uma ideologia voltada para a questão da sustentabilidade e necessidade.

Arana (1999) afirma que a sustentabilidade está voltada para o uso racional dos recursos naturais, ou seja, está relacionada com a capacidade de poluir menos o meio ambiente, de se modificar menos os ecossistemas existentes e de se trazer um maior retorno social.

Deseja-se poder estabelecer o controle dos dispositivos eletrônicos residenciais de forma prática e automatizada para o usuário, em outras palavras, fazer com que o usuário tenha o controle de sua residência na palma de sua mão.

Dessa forma, os recursos consumidos em uma residência podem ser gerenciados de forma eficaz, tendo como consequência a queda de consumo de energia e água, assim também como o bem estar dos moradores.

## 1.2 OBJETIVOS

Os objetivos deste trabalho estão divididos em: objetivo geral e objetivos específicos.

### 1.2.1 OBJETIVO GERAL

Desenvolver uma aplicação que permita ao usuário configurar, gerenciar e monitorar os dispositivos eletrônicos residenciais. Tal aplicação consiste no desenvolvimento de uma interface para os dispositivos móveis, usando a plataforma Android integrada à placa de hardware Arduino.

### 1.2.2 OBJETIVOS ESPECÍFICOS

- estudar a tecnologia da plataforma Android;
- realizar um estudo aprofundado sobre os conceitos e práticas da automação residencial;
- apresentar a modelagem do sistema proposto nos padrões da linguagem UML;
- desenvolver a interface a partir da modelagem proposta;
- permitir que o usuário tenha controle sobre os dispositivos eletrônicos residenciais assim como a parte de monitoramento;
- efetuar testes que garantam a estabilidade do projeto;
- expor os resultados obtidos.

### 1.3 JUSTIFICATIVA

No Brasil o mercado de Automação Residencial é uma realidade, porém existe uma falta de profissionais com conhecimentos suficientes para atender, projetar e instalar sistemas de Automação Residencial (SCLASS, 2012). Essa tendência não acontece apenas no país, as grandes empresas de tecnologia, como Apple, Google e Microsoft, também têm interesse na área de Automação Residencial (AURESIDE, 2013a), pois, além das questões de segurança, essa automação também permite uso mais racional dos recursos. Dessa forma, pode-se perceber a relevância da temática a ser abordada nesta monografia.

Segundo reportagem do G1 (2013), empresa de automação residencial aberta em 2010 faturou R\$ 1 milhão no ano passado. O custo de um projeto de automação de ambientes pode variar entre R\$ 10 mil e R\$ 30 mil, ficando à escolha do cliente os serviços que estarão a sua disposição.

A automação residencial é cada vez mais procurada pelas pessoas. Nos últimos 4 anos, o serviço cresceu 300%. Em 2012, o seguimento faturou R\$ 4 bilhões e pode crescer mais 30% em 2013. Estima-se que 1,8 milhão de residências estejam aptas a receber sistemas de automação.

Muitas são as opções de tecnologias que podem ser utilizadas para tais projetos. O uso das plataformas Android e Arduino faz parte das definições iniciais deste trabalho, assim, serão abordados, a seguir, alguns motivos que levaram a essa escolha.

A cada ano que passa, o uso do sistema operacional Android vem crescendo como opção nos aparelhos móveis.

Uma pesquisa da Online Publishers Association, uma organização sem fins lucrativos, dedicada a provedores de conteúdo online, afirma que atualmente o sistema operacional Android lidera o mercado nos Estados Unidos, rodando em 46% dos smartphones, contra 35% do segundo colocado UOL (2012).

Os recursos oferecidos pela plataforma Android e Arduino, aliados a sua grande popularidade contribuem para o desenvolvimento de soluções para os problemas encontrados na sociedade.

Além da relevância da área e das tecnologias a ser utilizadas, também existem motivações de cunho acadêmico e pessoal. Através deste trabalho, deseja-se pôr em

prática parte dos conteúdos estudados no curso de ciência da computação na elaboração de um sistema prático, necessário e rentável para o uso em automação residencial.

#### 1.4 ESTRUTURA DO TRABALHO

Este trabalho está organizado em 6 capítulos. O primeiro capítulo apresenta a introdução, a problemática, os objetivos e a justificativa do tema proposto.

No capítulo 2, tem-se a revisão bibliográfica. Nela são abordadas os temas pertinentes ao projeto, assuntos referente à plataforma Android e Arduino. Definições de Wi-fi, linguagem de programação Java, C e C++, assim como assuntos referentes à automação residencial.

Logo após, no capítulo 3, é apresentado o método científico da proposta. No capítulo 4, descreve-se a modelagem da solução proposta para o problema, usando a notação UML. Em seguida, nos capítulos 4 e 5, são abordados o método de desenvolvimento, a descrição da etapa de desenvolvimento e a arquitetura do projeto proposto.

Finalmente, no capítulo 6, são descritas as conclusões obtidas com a realização deste projeto e as sugestões para os trabalhos futuros.

## **2 REVISÃO BIBLIOGRÁFICA**

Este capítulo descreve os assuntos teóricos abordados por este trabalho, relatando o estado da arte sobre facilidades referente à automação residencial, principais conceitos sobre sistemas de automação residencial e sistemas embarcados, com destaque para a plataforma Arduino e programação para dispositivos móveis, apresentando dentro deste último uma seção sobre a plataforma Android.

### **2.1 ESTADO DA ARTE**

Esta expressão, “estado da arte”, é usada frequentemente em teses acadêmicas para descrever o que está feito atualmente no campo de estudo. Seu objetivo é mostrar as abordagens concorrentes a que se vai usar para solucionar o problema, considerando o nível mais alto de desenvolvimento, seja de um aparelho, de uma técnica ou de uma área específica (BOTELHO, 2009).

Seguindo o conceito de estado da arte definido por Botelho (2009), alguns sistemas de automação residenciais, disponíveis no mercado, foram pesquisados, utilizando a internet, com o propósito de estudar as suas abordagens em relação ao tema.

#### **2.1.1 Intelicasa**

A Intelicasa é um produto projetado com a finalidade de oferecer conforto, segurança, praticidade, economia e satisfação aos usuários residenciais. (INTELICASA, 2012). Em questões de conforto, o sistema permite ajustar o nível de luminosidade de acordo com o ambiente e a ocasião. No cenário cinema, a intensidade da luz diminui automaticamente, as persianas são fechadas, a tela desce ao ponto exato e o projetor é ligado.

O jardim também pode ser irrigado de forma automática.

Em questões de acessibilidade, são usadas fechaduras biométricas, que permitem a abertura das portas por meio da identificação das impressões digitais, previamente cadastradas, restringindo inclusive o acesso de pessoas em determinados horários e dias da semana. O leitor Biométrico pode ser visualizado na Figura 1.

Figura 1 – Leitor Biométrico.



Fonte: INTELICASA, 2012.

O sistema também permite ligar e desligar aparelhos eletrodomésticos, controlar e monitorar temperaturas ambientes e acionar e desligar alarmes.

Na parte de segurança, são disponibilizadas câmeras, sensores de presença, sensores de vazamento de gás, água e fumaça. (INTELICASA, 2012). Um dos modelos de câmeras de vigilância pode ser visto na Figura 2.

Figura 2 – Câmera de vigilância.



Fonte: INTELICASA, 2012.

Sua interface foi desenvolvida em versão para iPhone. Mas também há a possibilidade do controle ser via “Pulsadores” (equivalente aos interruptores nos sistemas convencionais). O controle de iluminação pelo sistema Intelicasa está ilustrado na Figura 3.

Figura 3 – Controle de Iluminação via iPhone.



Fonte: INTELICASA, 2012.

De um modo geral, o sistema possui uma grande variedade de recursos para a automação residencial. Por ser um sistema complexo, seu foco está voltado para os grandes empreendimentos.

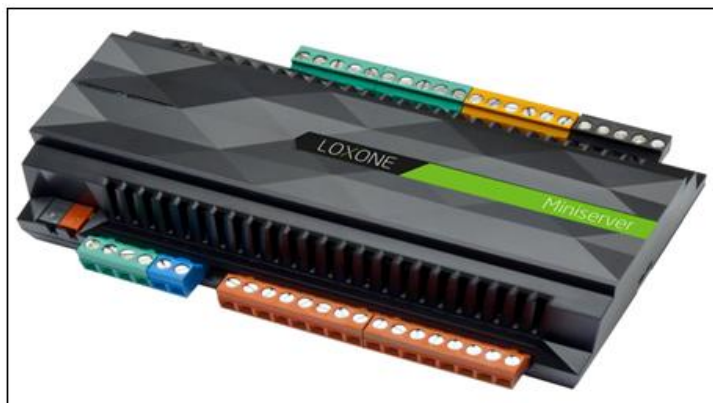
### **2.1.2 Loxone**

Segundo seus distribuidores, esse sistema possui controle automático para iluminação, aparelhos eletrônicos, sistemas de vídeos, áudio, alarmes e monitoramento. (LOXONE, 2012). As persianas são controladas com base nas horas do dia. Ao amanhecer, elas se abrem e, ao anoitecer, elas se fecham, possui também um ajuste automático para proteção da luz solar.



O sistema possui um controlador central, chamado de Miniserver, que permite o controle de todas as tarefas residenciais. Controla desde a abertura das cortinas até a temperatura do ambiente. A imagem do Miniserver pode ser vista na Figura 4.

Figura 4 – Controlador Central.

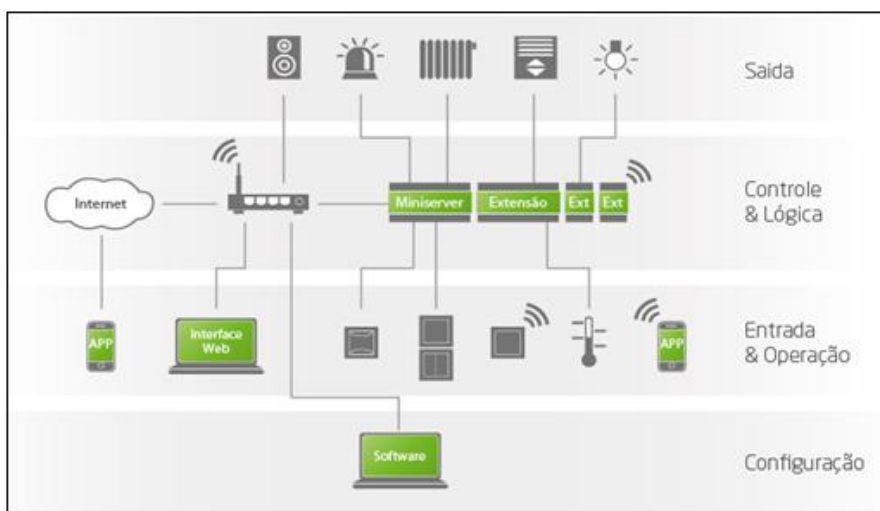


Fonte: LOXONE, 2012.

Este controlador central possui entradas às quais podem ser conectados aparelhos como: botões, detectores, contatos de porta e seus dispositivos móveis usados como interface. Na saída, são conectadas as cortinas, iluminação, ar-condicionado e demais aparelhos para controle.

A Figura 5 demonstra a estrutura do sistema Loxone.

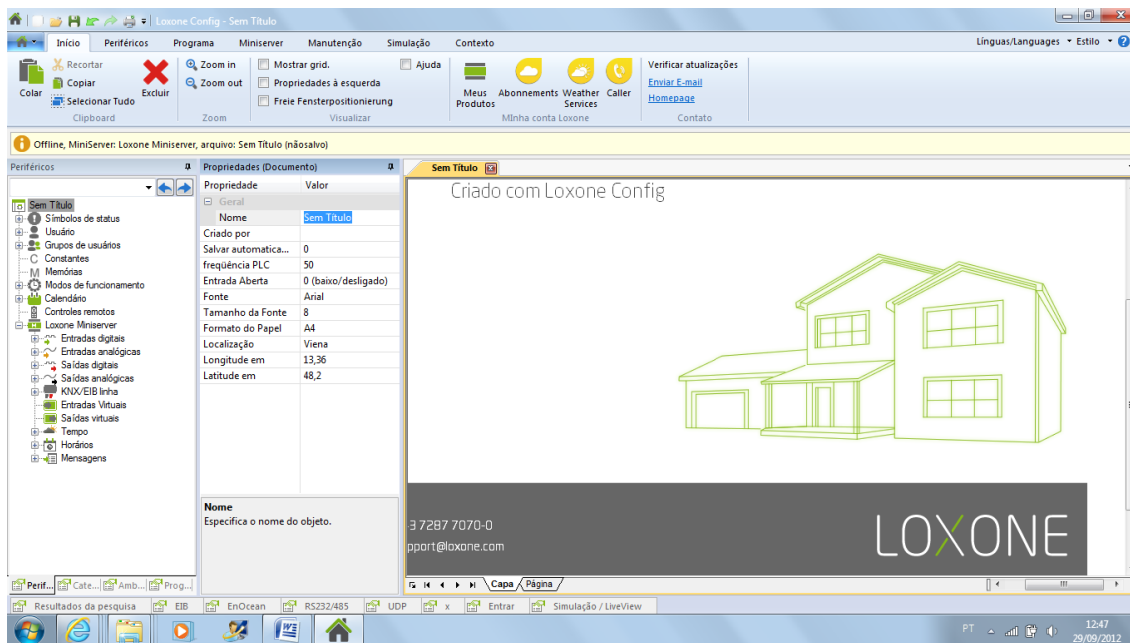
Figura 5 – Esquemático do sistema de automação residencial.



Fonte: LOXONE, 2012.

O Software de configuração (Figura 6) permite configurar e gerenciar todas as funções do Miniserver.

Figura 6 – Interface de Configuração do Miniserver.



Fonte: LOXONE, 2012.

O Loxone possui controle inteligente, permitindo que seus históricos sejam usados para adequar o ambiente à temperatura agradável. Também possui um sistema de comunicação por DTMF (*Dual-Tone Multi-Frequency*), que facilita a comunicação, via chamada telefônica.

Suas interfaces são disponibilizadas nas versões para iPhone, Ipad e Android. Possui interface simples e intuitiva. A Figura 7 apresenta algumas telas da versão Android.

Figura 7 – Telas do sistema Loxone para Android.



Fonte: LOXONE, 2012.

Como vantagem ao sistema da Intelicasa, este possui uma interface de configuração destinada ao usuário final, possibilitando autonomia em algumas configurações. Outra vantagem é possuir interface desenvolvida para o Sistema Android.

### 2.1.3 Iluflex

Segundo a Iluflex (2012), o sistema é completo e toda a comunicação para a Automação Residencial é feita sem fios, utilizando comandos via Infra-Vermelho, Rádio e Internet.

Com uma interface gráfica ou com uso de pulsadores, é possível controlar vários equipamentos, tais como equipamentos de áudio, vídeo, jogos, cortina, câmeras, alarme, portão e iluminação.

Possui integração com as tecnologias iPad, iPhone, iPod e Android. Ela permite programar e configurar cada módulo receptor encontrado no sistema, além de possuir os principais comandos implementados com protocolo proprietário. A interface para iPad e iPhone pode ser visualizada na Figura 8.

Figura 8 – Interface do sistema Iluflex para iPad e iPhone.



Fonte: ILUFLEX, 2012.

O módulo de RF tem o papel de receber comandos de um dispositivo central ou de pulsadores com a finalidade de controlar lâmpadas fluorescentes com reatores eletrônicos convencionais. Este módulo é indicado para acender, apagar e variar a intensidade de iluminação (dimerização) em três níveis diferentes (100 %, 75% e 50%).

Este único módulo permite o controle de até 6 reatores de 36 watts cada. Na Figura 9, é possível ver o módulo Receptor de RF.

Figura 9 – Módulo Receptor de RF.



Fonte: ILUFLEX, 2012.

Os pulsadores transmitem comandos para os módulos receptores, permitindo controlar a iluminação, alterando entre cenários pré-configurados. Seu funcionamento é via Rádio Frequência, podendo alcançar distancias de até 25 metros.

Sua energia é garantida através de uma bateria de lítio de longa duração e fácil substituição (ILUFLEX, 2012). Na Figura 10, é possível visualizar o Pulsador.

Figura 10 – Pulsador.



Fonte: ILUFLEX, 2012.

O módulo IR permite controlar até 3 aparelhos eletrônicos, exemplos: Televisores, Ar Condicionado, Home Theater, DVD, Blu-Ray, Sistemas de TV a Cabo. Utiliza a interface Wi-fi – IR. Segundo a Iluflex( 2012), estes aparelhos promovem um novo nível de automação e flexibilidade, controlando diversos equipamentos sem o uso de cabos. Na Figura 11, é exibida a imagem do Módulo Wifi – IR.

Figura 11 – Módulo Wi-fi – IR.



Fonte: ILUFLEX, 2012.

O Módulo Receptor RF (Figura 12) para Portão recebe comandos de abertura e fechamento, via Rádio Frequência. É indicado para portões e fechaduras eletromagnéticas

Figura 12 – Módulo Receptor RF para Portão.



Fonte: ILUFLEX, 2012.

A interface de controle Ethernet/RF envia os comandos para os módulos receptores de RF através das interfaces gráficas para tablets e celulares,

O modulo interface Ethernet/RF pode ser visto na Figura 13.

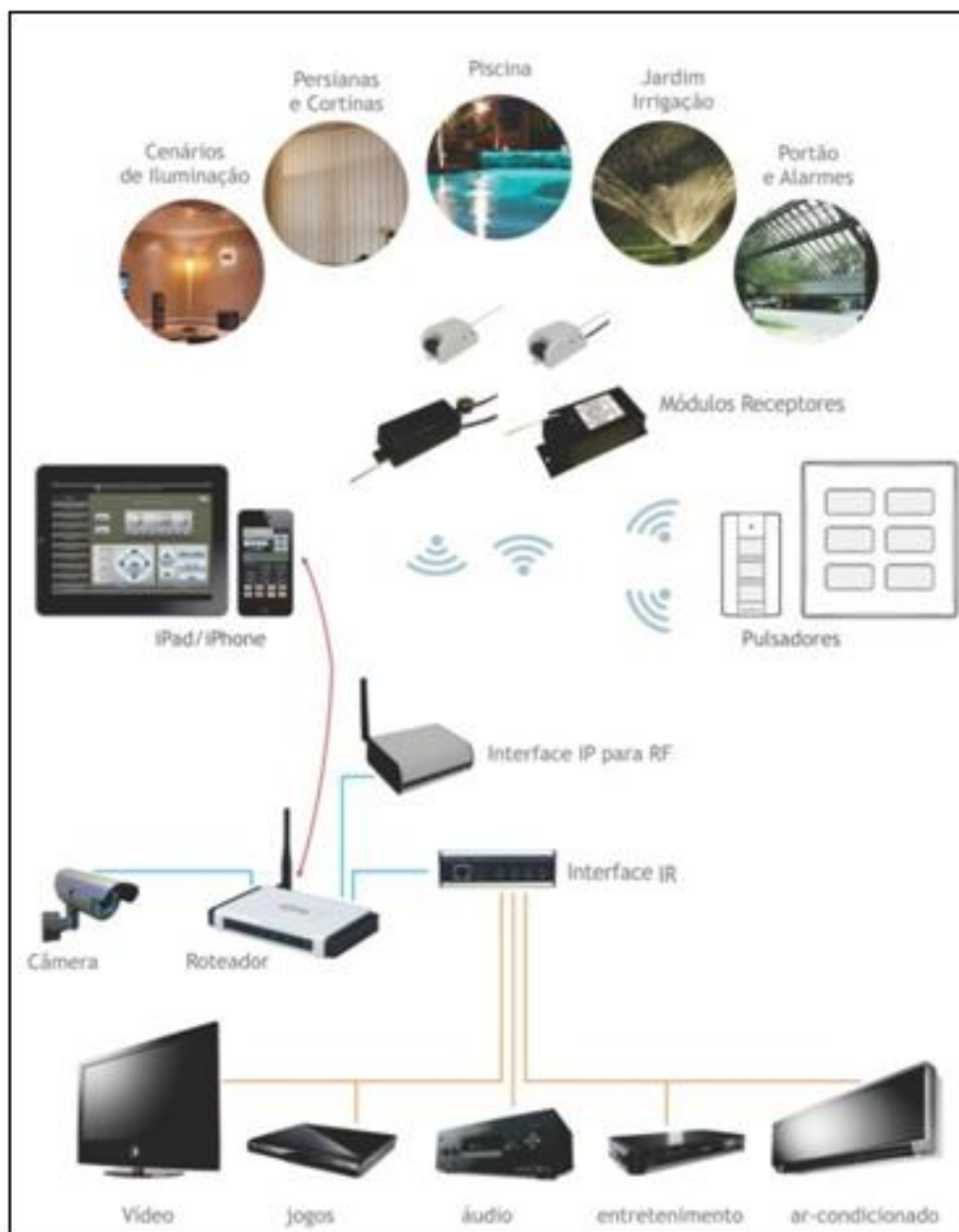
Figura 13 – Interface Ethernet RF.



Fonte: ILUFLEX, 2012.

A estrutura do sistema Iluflex pode ser visualizada na Figura 14.

Figura 14 – Arquitetura do Sistema Iluflex.



Fonte: ILUFLEX, 2012.

#### 2.1.4 Considerações Finais Sobre as Soluções Pesquisadas.

Os sistemas pesquisados para automação residencial têm seu projeto e desenvolvimento focado nos grandes empreendimentos, visando sempre às pessoas com maior poder aquisitivo.

Dessa forma, tende-se a utilizarem sempre tecnologia de ponta e sistemas complexos. (INTELICASA, 2012). Como exemplo, tem-se as soluções de automação residencial com interfaces disponíveis somente para o sistema operacional iOS (sistema da APPLE).

Segundo Silva (2013), a principal diferença entre os dois sistemas operacionais é o preço. Os aparelhos que rodam o Android são mais baratos que os que rodam o sistema da APPLE.

Em contrapartida, neste trabalho, tende-se a desenvolver um sistema menos complexo, utilizando softwares de código livre e componentes de menor valor, mas com desempenho, robustez e qualidade compatível com os demais sistemas, buscando, ao final, um protótipo com valor compatível às pessoas de média renda.

## 2.2 AUTOMAÇÃO

“Automação (do inglês *Automation*) é um sistema automático de controle pelo qual os mecanismos verificam seu próprio funcionamento, efetuando medições e introduzindo correções, sem a necessidade da Interferência do homem.” (PEREZ, 2009).

Groover (2001) define automação como a tecnologia preocupada com a aplicação de eletrônica, mecânica e sistemas baseados em computadores para controlar e operar um ou mais processos.

Automação é a mudança tecnológica na qual ocorre a substituição do homem por máquinas. Mudança essa que começou na Revolução Industrial e continua nos dias de hoje. (ROBBINS, 1999). Este mesmo autor descreve ainda que não há diferença entre o termo automação e automatização.



## 2.3 AUTOMAÇÃO RESIDENCIAL

Para Bolzani (2004), cada nova tecnologia traz acoplado um novo vocabulário. Conceito este aplicado também ao termo residência inteligente, que traz consigo vários outros sinônimos: casa automática, casa inteligente, automação residencial, domótica, etc. Mas todos esses termos podem ser resumidos em uma só palavra: conforto. Na automação residencial, os equipamentos devem ser unificados, controles e processos devem ser centralizados, tornando tudo mais simples.

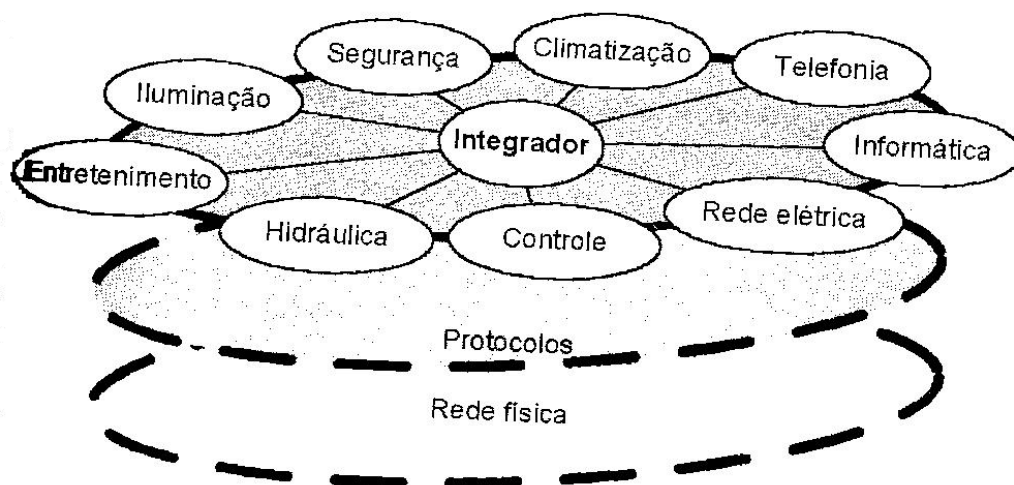
Segundo GDS (2012), automação residencial trata de um sistema baseado em automação para todas as funções encontradas no ambiente, fazendo integração de seus acionamentos e visando sempre à praticidade, simplificando a execução dos comandos. Tendo como resultado beleza, conforto e valorização do ambiente.

Sobre este conceito, Perez (2009) afirma:

Automação Residencial - Aplicação das técnicas de automação para melhoria no conforto e segurança de residências e conjuntos habitacionais, tais como: Controle de acesso por biometria, porteiro e portões eletrônicos, circuitos Fechados de Televisão (CFTV), controle de luminosidade de ambientes, controle de umidade, temperatura e ar condicionado (HVAC), etc.

Na figura 15, pode ser visto o conceito de Automação residencial, representado por Bolzani.

Figura 15 – Conceito de Automação Residencial.



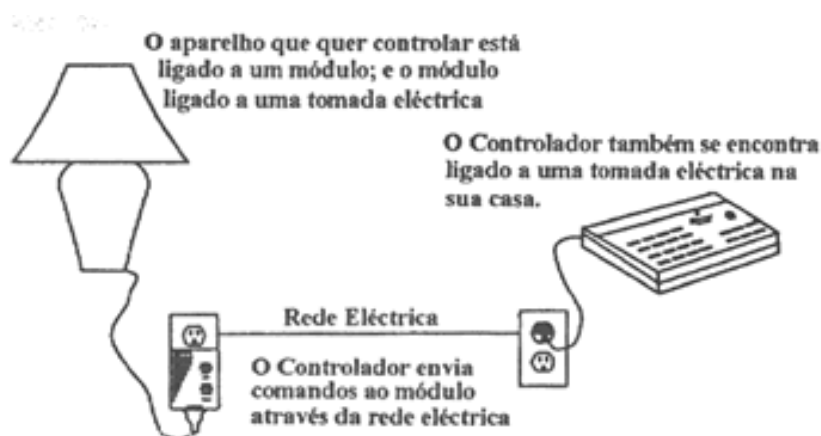
Fonte: Bolzani, 2004.

### 2.3.1 Histórico

Nos últimos anos, tem-se notado uma enorme evolução nos sistemas voltados para as residências, atraindo principalmente os olhares de grandes investidores, que buscam por novas funcionalidades e tecnologias em seus imóveis. (IHC, 2012). Isso inclui a própria automação residencial, que tem seu crescimento alavancado devido ao impulso do mercado de segurança, como alarmes e monitoramentos. O bem estar das pessoas e a necessidade de economizar alguns recursos, também, o motivaram, assim como a difusão dos conceitos de “*home theater*” e som ambiente, juntamente com a grande oferta de produtos e equipamentos para essa tecnologia.

Segundo Aureside (2012a), o grande precursor voltado exclusivamente para sistemas de automação residencial foi o protocolo X10, desenvolvido pela empresa “Pico Eletronics” na Escócia. Utilizavam a rede elétrica de 110V para a troca de comandos. Seu funcionamento era baseado na transmissão de dados binários através da corrente elétrica, usando pulsos de frequência de 60hz AC. Através do envio de mensagens dos transmissores para os receptores, era possível executar comandos básicos do tipo liga/desliga e dimerização. Essa tecnologia era definida como PLC (*Power Line Carrier*), que mais tarde deu origem a uma categoria de redes denominadas de *Powerline Networks*. Sua grande vantagem estava na facilidade de implantação, pois utilizava o cabeamento elétrico já existente. A patente do X10 expirou em dezembro de 1997, possibilitando, assim, o estudo e surgimento de novas tecnologias. A topologia do sistema é apresentada na Figura 16.

Figura 16 – Topologia do sistema X-10



Fonte: Digitat, 2008.

Em seguida, surgiu um novo protocolo, chamado UPB (*Universal Powerline Bus*) que seguia a mesma filosofia da tecnologia “Powerline Networks”. Foi desenvolvido em 1999 pela empresa americana Powerline Control e teve como base o protocolo X10, provendo melhorias como transmissão de dados e, principalmente, aumento na confiabilidade da rede. (AURESIDE, 2012a).

Segundo Cybertronics (2012), os principais benefícios do protocolo UPB são:

- confiabilidade: Usa frequência baixa e possui um sinal muito forte. Por usar 2 vias de comunicação, permite o feedback entre módulos, fazendo com que o módulo saiba se o comando foi executado com sucesso ou erro, mantendo assim uma comunicação confiável. Também, possibilita saber se um módulo está ativo ou não;
- alta velocidade: Em comparação com outras tecnologias *powerline*, o sistema é de 20 a 40 vezes mais rápido. Seu tempo de resposta é de 0,3 segundos ou 10 comandos completos por segundo;
- flexibilidade: Por ser compatível com comunicação centralizada ou ponto a ponto, permite ser adaptado a um grande número de aplicações;
- baixo custo: Por usar a rede elétrica existente, dispensa a passagem de novo cabeamento. Por ter equipamentos simples e confiáveis não há necessidade de grandes projetos para automatização e sua instalação quase nunca dá manutenção.

A Figura 17 mostra o diagrama representativo do sistema.



### 2.3.2 Características

Existe hoje, no mercado um leque de opções em sistemas de automação, desde o simples até o mais complexo. Exemplo disso pode ser observado na ExpoPredialtec, feira de automação e tecnologia que ocorreu em São Paulo em 2012. (PREDIALTEC, 2012). A feira superou as expectativas de público e negócio. Um dos grandes destaques foi à maquete 100% automatizada de uma casa já construída. Com total funcionalidade automatizada, disponibiliza um sistema de contingência, acionamento de modo próprio, entre outras características inovadoras.

Outro importante projeto apresentado foi a “casa do futuro”, projeto que destaca a casa móvel, também toda automatizada, comandada por uma tela sensível ao toque. Um dos detalhes mais interessantes é na sala de jantar, que vira uma pista de dança com um simples comando. Na feira foi apresentada outra grande novidade, o protocolo de automação com padrão europeu chamado KNX, responsável por facilitar a integração com inúmeros produtos de controle disponível no mercado. (PREDIALTEC, 2012).

Para Bolzani (2004), cabe ao usuário escolher a programação apropriada para as suas necessidades, levando em conta, é claro, o valor disposto a se pagar por um sistema desse tipo. Agregar e alterar funções aos equipamentos de segurança, home-theater, som ambiente, TV, climatização, telefonia, iluminação e todos os eletrodomésticos são algumas dessas programações.

Segundo depoimento de Hattori, o que há de mais moderno na automação não é necessariamente a engenharia usada na arquitetura da conexão entre os variados tipos de equipamentos, mas sim, a forma com que as opções de controle são acessadas pelo Usuário, ou seja, qual interface será usada para prover o acesso às funcionalidades disponíveis. Nesse caso, celulares e tablets se misturam, dando aos usuários grandes atrativos a ter automação em uma residência. (SMAAL, 2011).

Na parte de engenharia, diversos são os modelos das redes internas usadas para a transmissão de dados, ou seja, o envio de comandos pela interface, desde as baseadas em Ethernet até a rede elétrica residencial. Nesse aspecto, são encontradas tecnologias para essa finalidade do tipo *FireWire*, que apresenta boa qualidade de serviço e velocidade, porém seu cabeamento é limitado a pequenas distâncias. As redes sem fios também são muito usadas, mas em contrapartida, tem um alto custo em relação aos

equipamentos utilizados. Novas tecnologias são pensadas e pesquisadas com a finalidade de garantir que todos os requisitos sejam alcançados. (BOLZANI, 2004).

Bolzani (2004) descreve algumas características que são referências em sistemas de automação, são elas:

- integração entre todos os sistemas eletrônicos residenciais;
- possibilidade de manutenção e acesso remoto;
- atuar em condições adversas, como alteração de clima, vibrações e até mesmo na falta de energia;
- memória suficiente para armazenar as configurações e históricos dos comandos;
- prover noção temporal, aspectos como dia e noite e noções climáticas a fim de possibilitar a execução de atividades baseadas nestes fatores;
- usabilidade, mantendo uma relação amigável e de fácil entendimento com o usuário;
- autocorreção e Programação, sendo capaz de sugerir soluções para problemas e facilidade no ato de reprogramação, respectivamente.

### **2.3.3 Domótica**

O termo “Domótica” resulta da junção da palavra “Domus” (casa) com “Robótica” (automação). A Domótica permite a gestão de todos os recursos habitacionais, aquecimento, luz, alarme, eletrodomésticos, fechaduras, através de uma interface. (ARAÚJO, 2000).

Para Angel (1993), o significado está relacionado com a aplicação de tecnologia em residências, trazendo benefícios, como qualidade de vida, segurança, e viabilizando o uso racional de recursos.

A Domótica está dividida em dois tipos de arquiteturas. ABA (Arquitetura baseada em Automação), conhecida como Domótica estática e a ABC (Arquitetura baseada em Comportamento), essa conhecida como Domótica Inteligente, fazendo uso das técnicas de Inteligência Artificial para se adaptar ao comportamento humano. (LINS et al., 2009).

A arquitetura baseada em automação tem sua estrutura baseada em dispositivos, como controles remotos (controladores), sensores, e atuadores. Tais dispositivos são ajustados e configurados automaticamente pelo usuário. As necessidades do usuário não alteram as configurações do sistema. Já na arquitetura, baseada em comportamento, o sistema se adapta às necessidades do ambiente e do usuário, procurando sempre ajustar as regras já existentes. Essa adaptação se dá de forma automática e sem a intervenção do indivíduo. (MITCHELL, 1997).

Ainda, considerando a arquitetura ABC, Tonidandel (2004) descreve que a casa inteligente não deve ser somente provida de um controle central operada a risca por um usuário, simplesmente obedecendo a comandos. Ela deve ter vida própria. Portanto, deve ter características de um sistema inteligente e deve interagir com os habitantes da residência, aprendendo dinamicamente com os seus históricos de comportamento. Assim, os dados dos comandos, sensores e demais dispositivos envolvidos no ambiente devem se adaptar de forma automática aos hábitos dos usuários.

Barbosa (2011) descreve que a casa inteligente é capaz de aprender sobre as preferências do morador, aplicando esses costumes de forma autônoma, ou seja, sem a interferência humana. Por exemplo, no sistema de iluminação, ao invés do usuário acender ou apagar cada lâmpada via controle remoto, ter a possibilidade de que elas acendam automaticamente quando estiver escuro, caso tenha alguém no cômodo, e mais, ter o brilho controlado conforme a claridade. Outro exemplo pode ser o caso de um morador ter o costume de sempre ligar a TV ao chegar em casa, fazendo desse procedimento algo automatizado com a abertura da porta da casa. Ou ao invés de ligar a TV, preparar o banho, enchendo a banheira com água na temperatura ideal, e somente , após o banho, ligar a TV, ligando junto o ar condicionado controlado com base na temperatura ambiente. Ao dormir as luzes são apagadas, o gás é desligado, as portas e janelas são fechadas. Nesse caso, a casa se adapta as necessidades do morador, sendo que o sistema deve estar atento às possíveis mudanças comportamentais dos seres humanos que se alteram ao longo do tempo e deve ser dotado de técnicas de aprendizado.

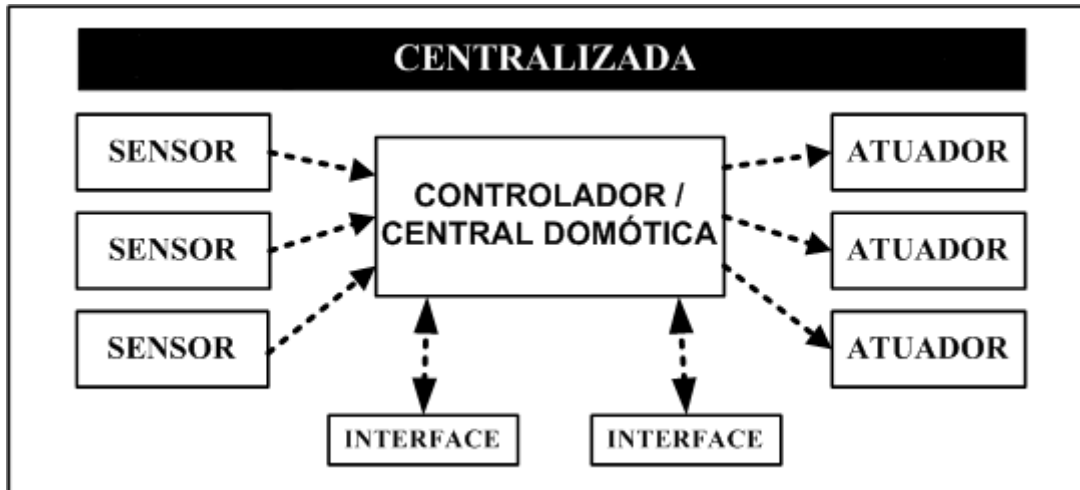
### 2.3.4 Estrutura dos sistemas de automação

Os sistemas de automação podem ser planejados de diferentes maneiras. Segundo Quicaze (2012), as soluções encontram-se dentro dos seguintes modelos:

➤ **Sistema Central:** O sistema está conectado a uma central que mantém toda a parte de inteligência e que é responsável por receber e enviar os comandos dos demais dispositivos do sistema de automação. Nesse modelo a interface pode ser ligada a central ou conectada individualmente em cada modulo atuador. Quando ligada diretamente ao atuador, mesmo com a central indisponível, é possível a execução de comandos, nesse caso, ficando sem a inteligência para a tomada de decisão. No cenário em que o comando não parte da central e, sim, de um atuador, todo e qualquer status deve ser repassado posteriormente para a central.

A seguir, na Figura 18, pode ser visualizado o esquema do modelo apresentado.

Figura 18 – Estrutura Centralizada.

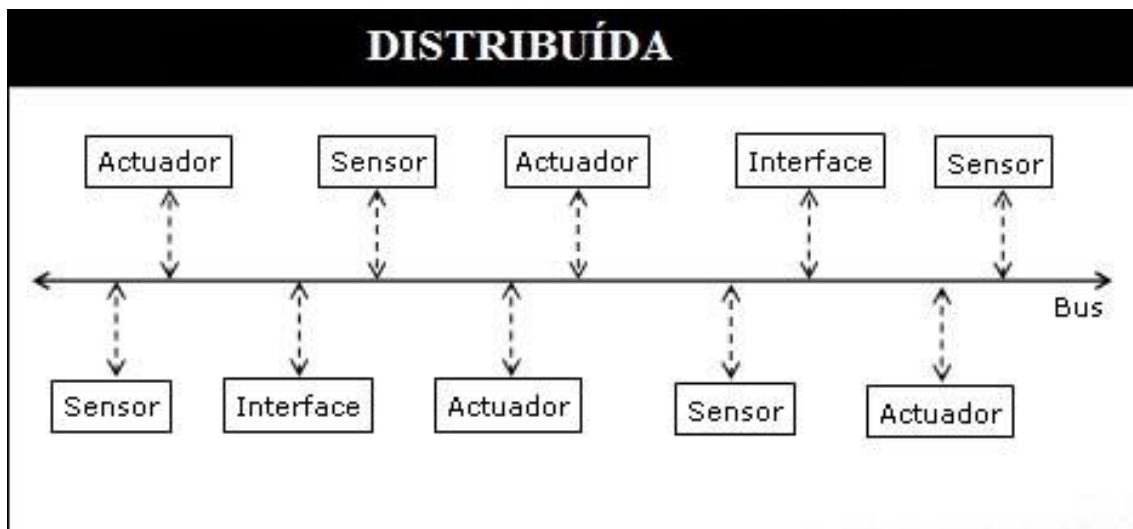


Fonte: CASADOMO, 2012.

➤ **Sistema Distribuído:** Essa topologia consiste na ligação de todos os dispositivos em um único barramento, conforme pode ser visto na Figura 19. Todos os dispositivos, atuadores e sensores possuem autonomia para atuação mesmo sem uma central.



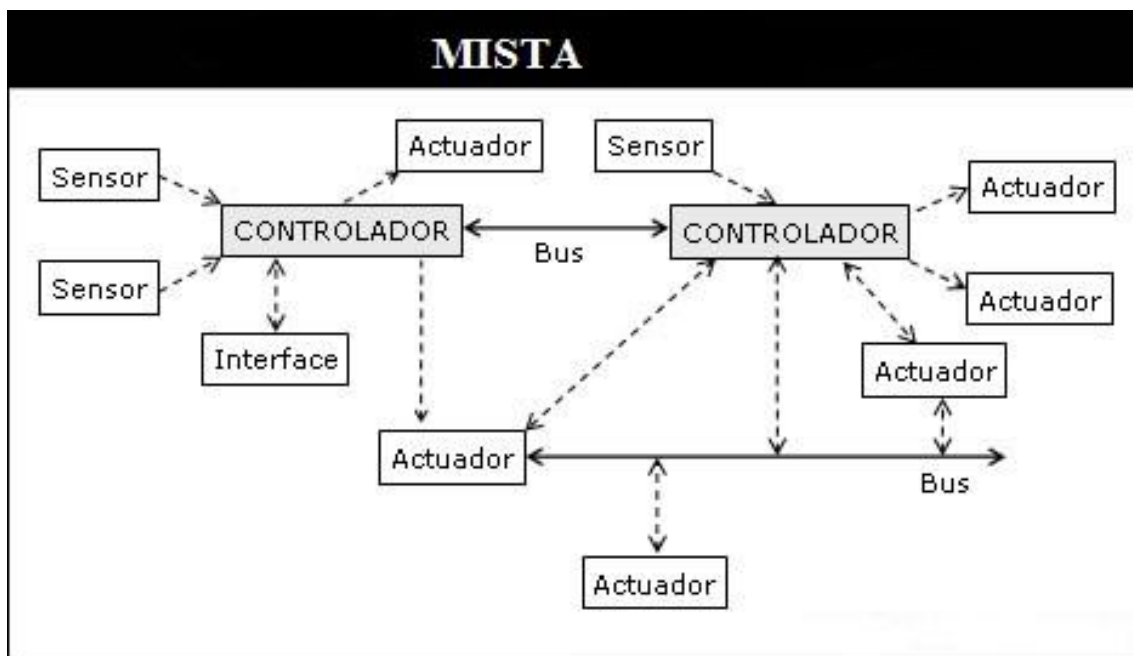
Figura 19 – Estrutura Distribuída.



Fonte: CASADOMO, 2012

➤ Sistema Misto: É o sistema cuja arquitetura está construída com várias centrais, uma em cada nó do sistema (atuador e sensor). Juntos mantêm a capacidade computacional e autonomia para a tomada de decisão. Os comandos são retransmitidos de nó para nó, até chegar ao seu destino, conforme pode ser visualizado na Figura 20.

Figura 20 – Arquitetura Mista.



Fonte: CASADOMO, 2012.

## 2.4 INTELIGÊNCIA ARTIFICIAL

Na apresentação sobre Domótica, foi observado que um sistema de automação residencial, pode utilizar técnicas de inteligência artificial, apresentando flexibilidade a tomada de decisões a partir de contextos e análise de dados. Por esse motivo, esta seção faz uma breve descrição desse conceito.

É uma área de pesquisa da ciência da computação que tem como objetivo desenvolver métodos para que seja possível criar e ou melhorar sistemas capazes de realizar processos que somente o homem é capaz. A Inteligência Artificial sistematiza e automatiza tarefas intelectuais e, portanto, é potencialmente relevante para qualquer esfera dentro das atividades humana. (SILVA, 2005).

Seus estudos foram iniciados nos anos de 1940 com o propósito de encontrar novas funcionalidades para o computador. Mas foi, após a Segunda Guerra Mundial, que seu estudo começou a ser aprofundando, com intenções voltadas para a indústria bélica.

Novas linhas de estudos surgiram, umas dela a biológica, que tinha como objetivo estudar o desenvolvimento de ideias que representassem o conceito de redes neurais humanas em máquinas.

Hoje em dia, são várias as aplicações que fazem uso da IA: jogos, robótica, sistemas para automação, dispositivos de reconhecimento de padrões. (RUSSELL et al., 2004).

## 2.5 MEIOS DE COMUNICAÇÃO

Um meio de comunicação é responsável por transmitir um conjunto de informações quantificáveis de um lugar para outro. Partindo do emissor para o receptor., essa informação, codificada pelo emissor, seria enviada por um canal para o receptor, que faria a sua decodificação. Nesse caso, o processo de comunicação está relacionado ao transporte da informação.

O desenvolvimento dos meios de comunicações alavancou o crescimento do mercado de automação residencial. (MURATORI, 2012).

### **2.5.1 Comunicação cabeada.**

Conceitualmente, o cabeamento residencial deve ser tratado como uma distribuição interna de cabos, com o intuito de garantir a transmissão de sinais de dados e ou voz, provendo flexibilidade a mudanças, longevidade, conveniência e conforto em relação às outras tecnologias. (MURATORI et al., 2012).

Cabeamento Residencial não Estruturado: A instalação de cabos é feita por demanda, ou seja, a cada nova necessidade é feita a passagem de um cabo para atender o serviço, pois a infraestrutura atual não está preparada para tal. Como vantagens estão a rapidez na instalação do serviço atual e o baixo consumo com a implantação do sistema. Alto custo de manutenção e pouco ou nenhuma flexibilidade são desvantagens desse cabeamento.

Cabeamento Residencial Estruturado: Instalação feita de forma planejada visando atender tanto as necessidades atuais como as futuras. Vários pontos de um mesmo serviço são disponibilizados em um mesmo ambiente. Nessa estrutura, a instalação é realizada com base nos conceitos de Telecomunicações. As desvantagens desse sistema são o alto custo inicial de instalação e uma maior demora na implantação do cabeamento, em contrapartida, como vantagem, temos a rapidez em na mudança de configuração do layout no ambiente, e o baixo custo com a alteração. (MURATORI et al., 2012).

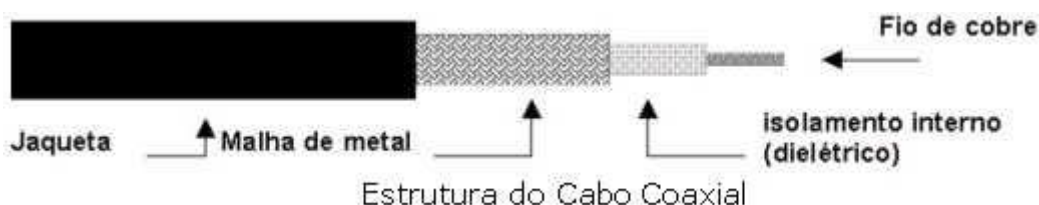
Além da rede elétrica como meio de comunicação cabeada, há disponível outros tipos de cabos usados em sistemas de automação residencial, por exemplo: cabo coaxial, cabo de par trançado e fibra ótica. (QUINCAZE, 2012).

### 2.5.1.1 Cabo Coaxial

Segundo Tecnosolution (2012), o cabo coaxial foi o primeiro tipo disponível no mercado, e era, até alguns anos atrás, o meio de transporte mais moderno em termos de cabeamento. Estes cabos são formados por 4 camadas: por fora uma camada chamada de jaqueta, abaixo uma malha de metal cobrindo as 2 camadas internas, uma camada isolante de plástico (chamado dielétrico que envolve o cabo interno) e no interior o condutor ( fio de cobre).

Na Figura 21, são apresentadas as 4 camadas:

Figura 21 – Estrutura do Cabo Coaxial.



Fonte: Tecnosolution, 2012.

Esse tipo de cabo caiu em desuso devido as suas desvantagens. Algumas delas são: custo elevado, instalação mais difícil, fragilidade e pouca flexibilidade a mudanças de layout.

Os que vêm sendo usados possuem algumas características específicas que possibilitam o seu uso em determinadas situações. Alguns são melhores para transmissão em alta frequência, outros têm atenuação mais baixa, e outros são imunes a ruídos e interferência. Ao contrário do cabo de par trançado, ele mantém uma capacidade constante e baixa, independente do seu comprimento. Devido a esse fator, ele fornece velocidade na ordem de megabits/segundos, não havendo a necessidade de regeneração do sinal. (SENA, 2005).

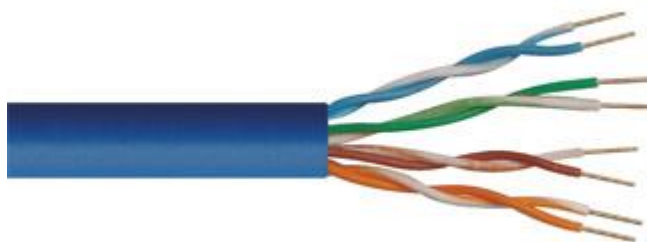
### 2.5.1.2 Cabo de Par Trançado

Devido a sua característica, de ser um cabo flexível e por ter uma taxa de transmissão mais alta, tornou-se um dos meios de transmissão físico mais usado. Esse cabo possui no seu interior dois ou mais fios entrelaçados em forma de espiral, desta forma, reduz o ruído e mantém constantes as propriedades elétricas em todo o seu comprimento. Por ele é possível tanto o tráfego de sinais analógicos como digitais.

Uma das grandes desvantagens desse cabo está relacionada ao grau de interferência a ruídos, seja ele derivado de fontes eletromagnéticas ou rádio frequência. Como solução a esse problema, foi desenvolvido o cabo blindado. (SENA, 2005).

➤ Cabo UTP (*Unshielded Twisted Pair*): Cabo sem blindagem. É composto por pares de fios trançados, sendo que cada par é isolado do outro, dentro de uma mesma cobertura externa, conforme pode ser visto na Figura 22. O entrelaçamento dos cabos cria um mecanismo de cancelamento mútuo, ou seja, é criado um campo eletromagnético que faz o papel de barreira, de tal forma a evitar a absorção de ruídos. (TECNOSOLUTION, 2012).

Figura 22 – Cabo UTP sem Blindagem.



Fonte: Tecnosolution, 2012.

➤ Cabo STP (*Shielded Twisted Pair*). Cabo blindado. É formado pelos mesmos pares de fios trançados, como o UTP, ele possui uma blindagem para diminuir ainda mais os efeitos de ruídos. Alguns tipos de cabos possuem uma blindagem em volta de cada par de fios e outra blindagem em volta deles todos, enquanto outros cabos possuem uma única blindagem, que envolve todos os pares de fios. (SENA, 2005).

Conforme Tecnosolution (2012), sua instalação é mais apropriada em locais que possuem fortes índices de ruídos, como ao lado de motores elétricos, redes elétricas, e estações de rádios. Na Figura 23, pode ser visto um exemplo de cabo com 2 blindagens:

Figura 23 – Cabo UTP com Blindagem.



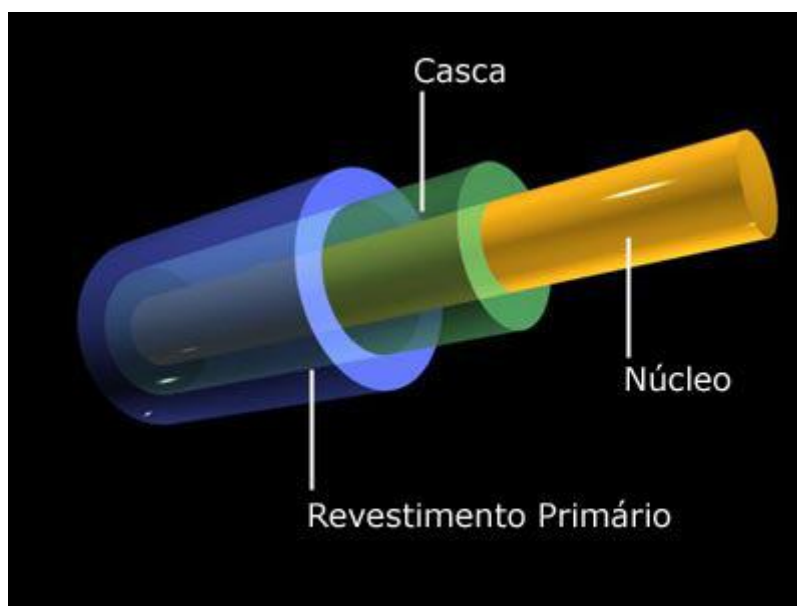
Fonte: Tecnosolution, 2012.

#### 2.5.1.3 Fibra Ótica

Conforme afirma Tecnosolution (2012), a fibra é um pedaço de vidro e ou de plástico com capacidade de transmitir luz. Foi desenvolvida pelo físico indiano Narinder Singh Kapany, podendo ser encontrada em tamanhos de diâmetro ínfimos, da ordem de micrômetros (mais fino que um fio de cabelo) até vários milímetros.

Seu funcionamento baseia-se na transmissão de luz, que segue o mesmo mecanismo independente da aplicação. Um feixe de luz é lançado em uma das extremidades e pelas características do meio (reflexões sucessivas) ela chega ao outro lado. A fibra óptica possui ao menos duas camadas, o núcleo, constituído de um filamento de vidro e o revestimento que serve de isolante elétrico. A imagem da fibra óptica é apresentada na Figura 24.

Figura 24 – As camadas da Fibra.



Fonte: Tecnosolution, 2012.

O princípio da transmissão de luz no interior da fibra é garantida devido a uma diferença de índice de refração entre o núcleo e o revestimento. Pelo fato do índice ser mais elevado no núcleo, unida à característica do ângulo de incidência do feixe de luz, possibilita o que é chamado de reflexão total. (TECNOSOLUTION, 2012).

Entre as suas enormes vantagens estão associados o autodesempenho, podendo transmitir uma enorme quantidade de dados, as dimensões reduzidas no meio físico, atenuação baixíssima, possibilitando que os repetidores sejam instalados em distâncias superiores a centenas de quilômetros um do outro, imunidade a interferências eletromagnéticas e pela presença de matéria prima em abundância. Porém, como toda solução possui algum tipo de desvantagem, essa não seria diferente. Assim, pode ser citado o alto custo com sua implantação, devido a seus equipamentos utilizados e sua instalação, à fragilidade e à falta de padronização dos componentes ópticos.

A Figura 25 apresenta um comparativo referente à velocidade de transmissão, quando comparada com outros meios de conexão:

Figura 25 – Velocidade de Transmissão.



Fonte: TECNOSOLUTION, 2012.

### 2.5.2 Comunicação sem fio.

Atualmente, a rede sem fio vem sendo usada constantemente para a transmissão de dados em redes locais, deixando de ser usada somente para comunicações a longa distancias através de satélites. Nas redes locais, elas são baseadas em ligações que utilizam ondas de rádio e ou comunicação via infravermelho. (KIOSKEA, 2012).

Um dos segmentos em que mais tem crescido o seu uso é em redes domésticas, sendo utilizadas em aplicações voltadas para automação comercial e residencial. Com essa tecnologia, podemos controlar vários equipamentos remotamente, como: televisores, home theaters, iluminação, ar condicionado, portão eletrônico. (PINHEIRO, 2006).

Para Bolzani (2004), uma das vantagens em relação ao seu uso está no fato da eliminação de fios, tornando o sistema mais flexível e prático. No ambiente residencial, várias soluções têm sido adotadas para a comunicação entre os vários dispositivos inteligentes.

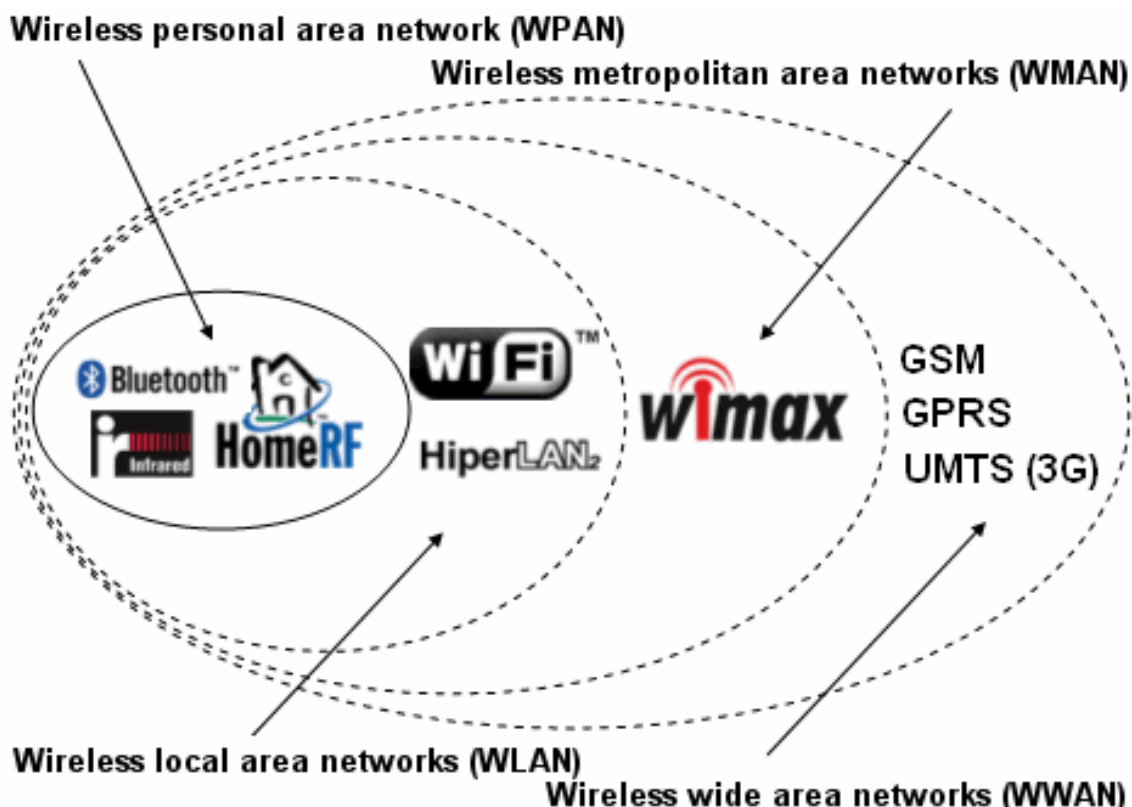


- **Radiofrequência:** São ondas de rádio, formadas por radiações eletromagnéticas com características semelhantes ao infravermelho, mas com parâmetros diferentes. São usadas para a comunicação em rádio amador, radiofusão (rádio e televisão), telefonia móvel.
- **Infravermelho:** Radiação não ionizante na porção invisível do espectro eletromagnético que está adjacente aos comprimentos de onda longos, ou final vermelho do espectro de luz visível. Usados geralmente em controles remotos.

Além das tecnologias acima citadas, utilizadas em residências, um padrão que vem sendo adotado é o IEEE802.11b, também conhecido como Wi-Fi (*Wireless Fidelity*).

Kioskea (2012) relata que as redes sem fio são divididas em categorias, conforme sua área de cobertura. Essas categorias podem ser vistas na Figura 26.

Figura 26 – Categorias de Redes sem Fio.



Fonte : Kioskea, 2012.

Definições referentes às categorias (SILVA, 2008):

- WPAN (*Wireless Personal Area Network*): Está normalmente relacionada ao Bluetooth, IR e HomeRF. Foi projetada para pequenas distâncias, baixo custo e baixas taxas de transferência.
- Wireless LAN ou WLAN (*Wireless Local Area Network*): Rede local que usa ondas de rádios na conexão com a internet ou entre redes. Considerada hoje muito importante nas várias áreas de negócio. Inicialmente, foi projetada para universidades e aeroportos.
- WMAN (*Wireless Metropolitan Area Network*): Usadas em redes Metropolitanas.
- WWAN (*Wireless wide Area Network*): É uma rede utilizada para longa distância, também conhecida como Rede geograficamente distribuída. Sua abrangência envolve uma grande área geográfica, com frequência um país ou continente.

## 2.6 PROTOCOLOS

Albuquerque (2001) define protocolos de comunicação como sendo um conjunto de regras padronizadas que permitem a comunicação, ou seja, troca de informações entre diferentes tipos dispositivos que compartilham de uma mesma ligação, utilizando da troca de unidades de dados chamados de PDUs (*Protocol Data Units*). São os tipos de protocolos que definem o formato das unidades de dados. Esses PDUs são regidos por alguns elementos chaves. A seguir, uma descrição de quais são esses elementos:

- sintaxe: formato dos dados e a ordem em que aparecem. Define a funcionalidade de cada byte em sua respectiva posição;
- semântica: é o significado de cada conjunto sintático para dar sentido a mensagem;

- timing: define qual será a velocidade de transmissão dos dados, sendo levados em consideração ambos os dispositivos envolvidos na comunicação.

Funções do protocolo em uma comunicação:

- endereçamento: especificação do destino da mensagem;
- numeração e Sequência: cada mensagem é indexada por um código sequencial;
- estabelecimento da conexão: estabelecimento de um canal entre transmissor e receptor;
- controle de erros: identificação e correção de erros durante a comunicação;
- confirmação de recebimento: envio de um sinal para confirmação de mensagem recebida;
- retransmissão: repetição da mensagem quando a confirmação de mensagem não é recebida;
- conversão da mensagem: adequação da mensagem em código perceptível ao destinatário.

Para Quincaze (2012) indiferente da função, origem ou fabricante, todos os equipamentos que compõem o sistema de automação residencial precisam falar o mesmo idioma. Diferentes tipos de protocolos podem ser usados, tanto os destinados à rede sem fio quanto aos projetados para rede cabeada.

Dentre os protocolos mais usados, atualmente, estão: o protocolo ZigBee (comunicação sem fios), o protocolo CEBus (utilizando a rede elétrica), o protocolo Consumer IR (infravermelho) e, também, o protocolo CAN, muito utilizado em automação automobilística, mas com ótimas aplicações residenciais.

### 2.6.1 ZigBee

Foi desenvolvido pela ZigBee Alliance, uma aliança formada por grandes empresas do ramo de tecnologia, para servir de alternativa de comunicação sem fio em sistemas simples, cujas soluções necessitavam de baixo custo e o mínimo de consumo de energia. Trata-se de uma tecnologia baseada em protocolos para a transferência de pacotes de dados, oferecendo flexibilidade quanto ao dispositivo que pode ser controlado (FARAHANI, 2008).

As aplicações do protocolo ZigBee podem ser vistos na Figura 27:

Figura 27 – Aplicação do protocolo ZigBee.



Fonte: Messias, 2008.

Segundo Pinheiro (2006), esse protocolo, também conhecido como *HomeRF Lite*, corresponde a norma IEEE 802.15.4, homologada em maio de 2003, como camada física e camada de acesso ao meio.

Farahani (2008) descreve algumas características deste protocolo, como:

- baixo custo;
- facilidade de implementação;
- diferentes topologias de rede: malha, estrela ou árvore;
- baixo consumo de energia devido ao modo de operar em modo *sleep*;
- segurança e confiabilidade, fazendo uso de recursos de criptografia com a implementação do padrão AES (*Advance Encryption Standard*) de 128 bits.

Segundo Pinheiro (2006), o protocolo ZigBee está dividido em cinco camadas do modelo OSI. Esta divisão pode ser vista na Figura 28

Figura 28 – Camadas do protocolo ZigBee.



Fonte: Moecke, 2008.

- PHY (*Physical layer*): Camada física. Definição IEEE 802.15.4.
- MAC (*Médium Address Control*): Enlace. Definição IEEE 802.15.4.
- NWK: Rede. Definida pela Zigbee Alliance.
- Suporte a Aplicação. Definida pela Zigbee Alliance.
- Aplicação Perfil. Definida pelo usuário.

Segundo Messias (2008), esse sistema de transmissão é baseado no DSS (*Direct Sequence Spread Spectrum*), Espalhamento Espectral por Sequência Direta, trabalhando de forma robusta e operando em frequências ISM (*Industrial, Scientific and Medicinal*), sendo na Europa de 868 MHz (1 canal), 915 MHz (10 canais) nos Estados Unidos e 2,4

GHz (16 canais) em outras partes do mundo. Essa faixa de frequência não requer licença para uso.

As redes formadas por esse protocolo têm a capacidade de abranger milhares de dispositivos, mantendo uma excelente imunidade contra interferências, mantendo taxas de transmissão entre 20 Kbps a 250 Kbps. (MESSIAS, 2012).

#### 2.6.1.1 Dispositivos da Rede ZigBee

Em uma estrutura do tipo ZigBee existem dois tipos de dispositivos. (MESSIAS, 2008).

- **FFD (*Full Function Device*):** Dispositivos de funções completas, assim sendo, necessitam de hardwares mais completos para a implantação da pilha de protocolos. Podem assumir o papel de Coordenador, Roteador ou até mesmo um dispositivo Final. Não há restrições de conexão entre os dispositivos.
- **RFD (*Reduced Function Device*):** Dispositivos mais simples. Basta hardwares menos complexos para implementar a pilha de protocolos. Na rede assumem o papel de dispositivo final (*End Device*). Podem ser dispositivos do tipo: interruptores, dimmers, controle de relês, sensores, entre outros. Cada dispositivo RFD só pode manter contato com um dispositivo da categoria FFD

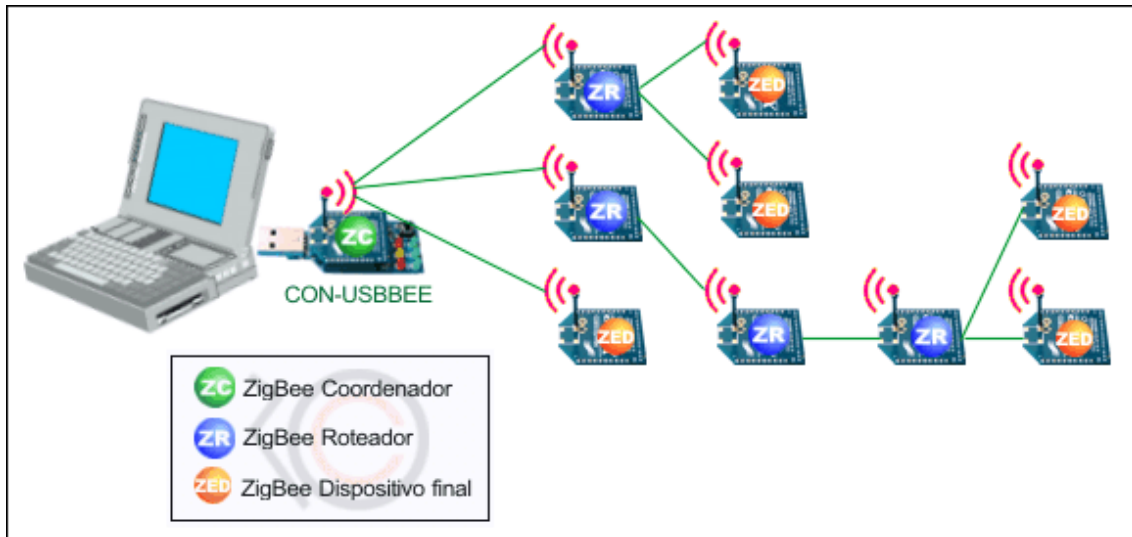
As categorias dos dispositivos são divididas em três:

- **ZC (*Coordinator*):** o coordenador tem função de inicialização, distribuição de endereços, manutenção e reconhecimento de todos os nós da rede. Uma função destes dispositivos entre várias outras, também é de servir de ponte para demais redes ZigBee;
- **ZR (*Router*):** tem como função servir de ponte entre os nós sem o auxílio do coordenador. Uma das principais características é poder amplificar o sinal das redes podendo, assim, expandi-las. Possui as demais características de um nó da rede;

- **ZED (End Device):** como é hospedeiro dos atuadores e sensores, tem o menor consumo de energia, pois fica em estado de repouso na maioria das vezes.

Na continuação, segue a imagem do uso das categorias, Figura 29.

Figura 29 – Rede ZigBee.



Fonte: Messias, 2008.

#### 2.6.1.2 Topologias ZigBee

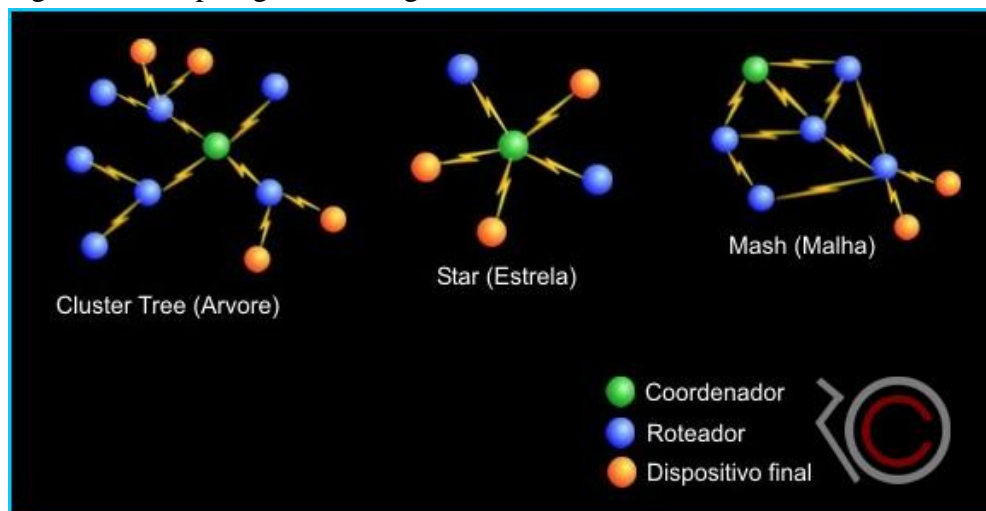
Segundo Pinheiro (2006), esse protocolo define uma estrutura de rede baseada nos conceitos de redes ad hoc, conexões em malhas e multi-hop.

- **Estrela:** Topologia de uso simples, indicada para ambientes que ofereçam poucos obstáculos para a transmissão do sinal. Formada pelo coordenador e os elementos finais, tem como vantagem a facilidade de implantação e coordenação, porém o alcance do sinal de RF não é tão grande e gera um alto grau de dependência entre coordenadores e clientes.
- **Árvore:** Um elemento coordenador para cada rede. As redes se comunicam através dos roteadores. Sua aplicação é indicada para ambientes separados por paredes.

- **Malha:** A cada nova entrada de um dispositivo, a rede ajusta suas configurações automaticamente na sua inicialização, com isso, possibilitando a otimização do tráfego de sinais.

As topologias citadas acima estão presentes na Figura 30.

Figura 30 – Topologias rede ZigBee.



Fonte: Messias (2008).

## 2.6.2 Cebus

“O Protocolo CEBus (*Consumer Electronic Bus*), criado pela EIA (Associação de Indústrias Eletrônicas) em 1984 e transformado em padrão internacional em 1995, tem como objetivo a padronização da utilização e da industrialização de produtos de comunicação infravermelhos, usados em controle remotos.” (TEZA, 2002, pg. 64).

Suas características são:

- possibilita automação de residências existentes mediante a reforma;
- existe uma variedade de dispositivos compatíveis;
- boa relação de custo e benefício;
- suporte a distribuição de serviços como áudio e vídeo;
- portabilidade de dispositivos.



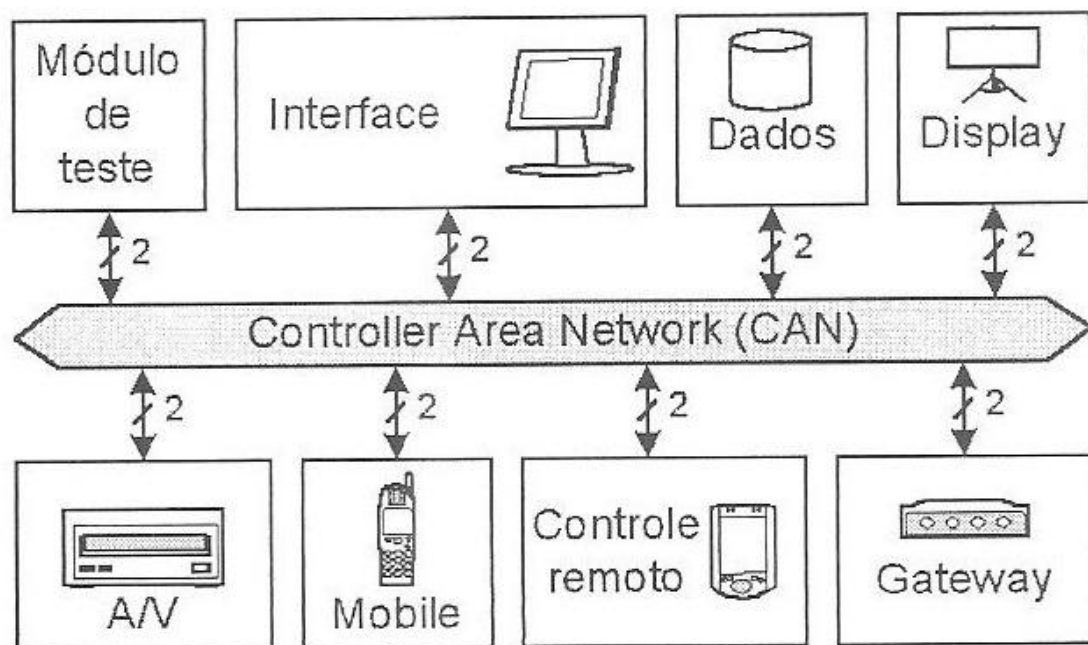
### 2.6.3 Can

*Controller Area Network* é um protocolo de comunicação serial desenvolvido pela Bosch para ser usado dentro do setor automobilístico no início dos anos 80.

Seu desenvolvimento tinha como objetivo prover um barramento de dados com grande imunidade a interferências elétricas e habilidade em detectar erros, oferecendo, ao mesmo tempo, uma boa taxa de transmissão. Após sua criação, o protocolo se tornou uma norma internacional (ISO 11898 e ISO 11519). Com isso, a Bosch licenciou algumas empresas, permitindo que desenvolvessem controladores e outros dispositivos compatíveis com a norma CAN. (BOLZANI, 2004).

A base de seu funcionamento permite que sensores, controladores e atuadores possam se comunicar entre si, em tempo real, com velocidade de até 1 Mbps, utilizando, como meio de transmissão serial, um par trançado. A Figura 31 apresenta a arquitetura CAN.

Figura 31 – Exemplo de uma rede Can.



Fonte: Bolzani, 2004.

A DaimlerChrysler foi a primeira montadora de automóveis a fazer uso dessa tecnologia, em seguida, o setor têxtil também optou pelo uso do CAN, equipando os teares da década de 90.

Conforme Bolzani (2004), devido à enorme adequação desse protocolo, na indústria automobilística, automação industrial e predial, algumas empresas estão investindo mais na sua utilização em automação residencial, criando novas opções tecnológicas para agregar ao uso do protocolo, como o desenvolvimento de interfaces.

Sua utilização torna-se uma escolha econômica de rede de dados para muitas aplicações residenciais, interligando os sensores, atuadores e todo o controle real de dados através de par trançado e outros meios físicos como a fibra óptica ou cabo coaxial.

Intel, Motorola, NEC e Siemens são algumas fabricantes que optaram por produzir dispositivos CAN. Consequentemente, os preços ficaram mais acessíveis, tornando a arquitetura uma boa opção de escolha para automação residencial.

Relação de algumas características:

- faixa de operação de até 1Mbps para distancias curtas;
- boa detecção de erros;
- configuração flexível;
- sistema de mensagens para envio de dados.

Os dispositivos da rede CAN utilizam o modo *multicast* para se comunicar com o barramento. Esse método consiste em rotular a mensagem com um identificador único na rede. Cada membro executa uma verificação na mensagem através do identificador para saber se deve ou não processá-la. Também possui critério de validação para prioridade das mensagens via o valor do número no identificador.

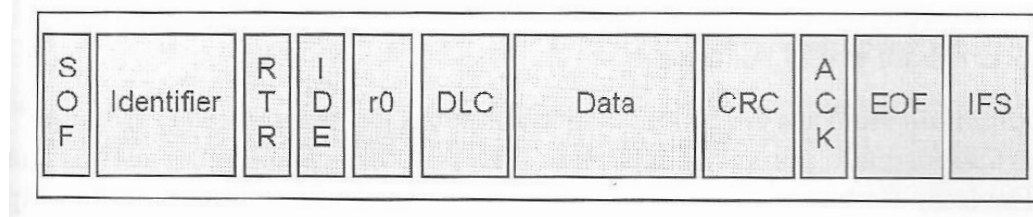
O sistema suporta uma conexão com até 110 dispositivos em distâncias não superior a 1 Km (queda considerável na taxa de transmissão). Para barramentos de 50 metros, a velocidade pode chegar a 1 Mbps, enquanto que, para um de 500 metros, a taxa é de 100 Kbps.

Existem dois formatos de *frames* que são utilizados, o *Standard* e o *Extended* (BOLZANI, 2004). A seguir são apresentadas as características do modelo *Standard*:

- **SOF (*Start of Frame*):** indica o início do frame.
- **RTR (*Remote Transmission Request*):** o campo possui 12 bits, 11 para o identificador e 1 para o RTR. Bit RTR 0 indica que a mensagem é um *Data Frame*, caso contrário, indica que é um RTR, requisição de envio de dados de outro nó do barramento.
- **DLC (*Data length Code*):** contém 6 bits, dos quais r0 e r1 são campos reservados e 4 bits para o DLC, que indica o número de bytes de dados que o frame carrega.
- **Data Field:** campo de dados, contendo 8 bytes, com o bit mais significativo do primeiro byte sendo transmitido primeiro.
- **CRC:** possui 15 bits e um bit recessivo delimitador.
- **ACK (*Acknowledge*):** consiste de um bit Slot, recessivo, porém é sobrescrito por bits dominantes, transmitidos de um outro nó que recebe a mensagem com sucesso. O segundo bit é recessivo delimitador.
- **EOF (*End of Frame*):** possui 7 bits recessivos. O IFS (*Intermission Frame Space*) contém 3 bits recessivos, que indica quando todos os controladores CAN devem se preparar para a próxima tarefa a ser executada.

Abaixo, na Figura 32, segue a ilustração do frame tipo *Standard*:

Figura 32 – Frame Standard Can.



Fonte: Bolzani, 2004.

Quanto ao frame *Extended CAN*, esse surgiu para prover compatibilidade com outros protocolos de comunicação serial. Possui um identificador maior de 29 bits, havendo um bit para identificar qual tipo de mensagem pertence o frame. Dependendo do identificador base, as mensagens *Standard* e *Extended* terão prioridades diferentes.

#### 2.6.4 Bluetooth

Em meados de 1994 a Ericsson iniciou os estudos com o objetivo de desenvolver uma tecnologia que permitisse a comunicação entre telefones celulares e acessórios. Em consequência, surge um sistema de rádio de curto alcance, que, recebeu o nome de *MC-Link*.

Dois anos depois, surge o *Bluetooth SIG (Bluetooth Special Interest Group)*, fruto do apoio ao projeto *MC-Link* pelas companhias Ericsson, Intel, IBM, Toshiba e Nokia.

O nome *bluetooth*, foi dado em homenagem ao rei dinamarquês Harald Blatand, conhecido como Harald *Bluetooth ((Haroldo Dente-Azul)*, o unificador da Dinamarca, assim *bluetooth* significa união. (INFOWESTER, 2013).

*Bluetooth* é um protocolo padrão de comunicação sem fio de baixíssimo custo e com baixo consumo de energia, utilizado em pequenas distâncias. Por meio dessa tecnologia, é possível a comunicação sem fio entre aparelhos eletrônicos e dispositivos móveis, desde que possua um chip para recepção e transmissão *bluetooth*.

Um das grandes desvantagens dessa tecnologia estão a distância limite de conexão, que pode variar de 1 a 100 metros, e também, o número máximo de dispositivos que podem se conectar ao mesmo tempo, limitado em cada dispositivo que recebe as conexões. (FLÀVIA, 2010).

Segundo Infowester (2013), o alcance máximo do Bluetooth, está dividido em 3 classes.

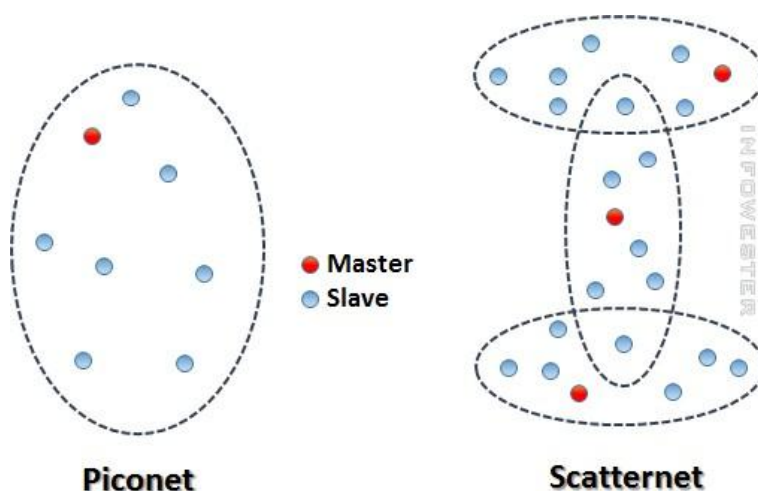
- Classe 1: potência máxima de 100 mW, alcance de até 100 metros;
- Classe 2: potência máxima de 2,5 mW, alcance de até 10 metros;

- Classe 3: potência máxima de 1 mW, alcance de até 1 metro.

Esta comunicação é realizada por meio de ondas de rádio na faixa de 2.4GHZ, isenta de licença. Em relação a velocidade de transmissão de dados, a classe 1 pode alcançar, no máximo 1Mb/s, a de classe 2, até 3.0Mb/s, já as de classe 3 podem chegar até 24Mb/s.

Segundo Tanenbaum (2003), a rede de comunicação estabelecida por dispositivos Bluetooth, são denominadas de *piconet*, que consiste em um nó mestre e até sete nós escravos ativos. É possível utilizar-se de diversas *piconets* dentro de um mesmo ambiente, estando conectadas entre si por um nó de ponte, esquema este denominado de *scatternet*. A estrutura de comunicação entre dispositivos Bluetooth pode ser visto na Figura 33.

Figura 33 – Estrutura de comunicação Bluetooth.



Fonte: InfoWester, 2012.

O mestre é o dispositivo que inicia a conexão, enquanto os demais dispositivos se tornam escravos, cabendo ao mestre, a tarefa de regular a transmissão de dados na rede e o sincronismo entre os dispositivos.

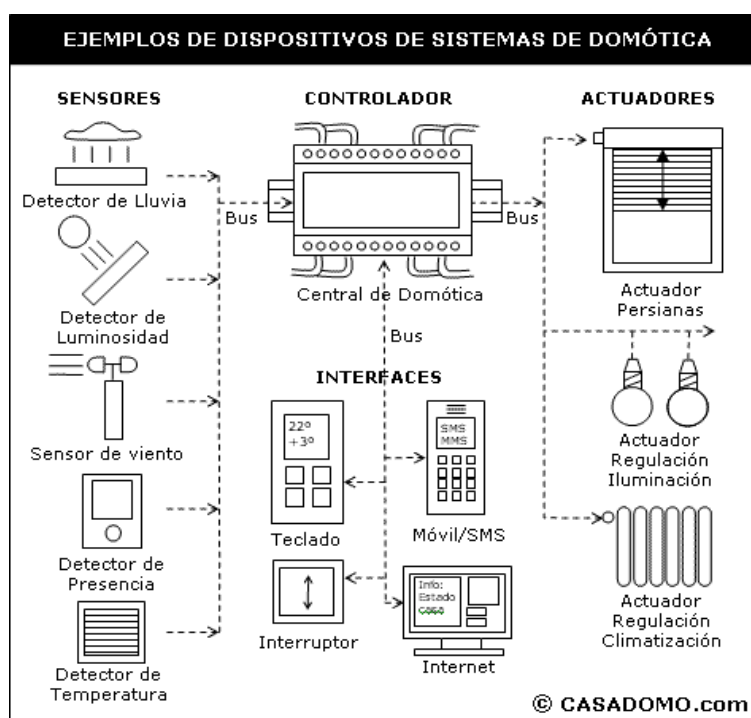
Para estabelecer conexões no Bluetooth, são necessários três procedimentos básicos.

- **Scan:** é utilizado para economia de energia. Faz com que os dispositivos ociosos entrem em modo *stand-by*. A cada 10 segundo, uma nova varredura é realizada, verificando se existe algum dispositivo tentando estabelecer conexão.
- **Page:** executado pelo equipamento que procura por conexão. A cada 1,25 milissegundos são transmitidos dois pedidos de conexão. O dispositivo verifica duas vezes se há resposta.
- **Inquiry:** mensagens enviadas com a finalidade de descobrir quais dispositivos *bluetooth* estão na área de cobertura, assim como suas características.

## 2.7 DISPOSITIVOS DO SISTEMA

A amplitude de uma solução de automação residencial pode variar, dependendo dos dispositivos utilizados no projeto. (CASADOMO, 2012). A estrutura de um sistema de automação residencial pode ser visto na Figura 34.

Figura 34 – Dispositivos de um Sistema de Automação.



Fonte: Casadomo, 2012.

- Controlador: fundamental no projeto de automação. É ele que gerencia os demais dispositivos da rede, podendo haver um único controlador ou vários outros distribuídos.
- Atuador: dispositivo responsável por executar e ou receber comandos do controlador. Eles trabalham em conjunto com os equipamentos eletrônicos, aplicando os comandos como ligar, desligar, subir e descer. São exemplos de atuadores: válvulas, motores, relés, etc.
- Sensor: capta as informações do meio em que está inserido. Exemplos: sensores de água, temperatura, presença, gás, luz, entre outros.
- Interface: está associada ao meio como as informações são exibidas ao usuário e como os comandos podem ser escolhidos, ou seja, qualquer forma de interação com o usuário. alguns exemplos podem ser vistos na Figura 35.

Figura 35 – Exemplos de Interface.



Fonte: Unacasa, 2011.

## 2.8 SISTEMAS EMBARCADOS

Sistemas embarcados são circuitos integrados, equipamentos ou sistemas com capacidade computacional (ELETRÔNICA, 2008).

Embedded (2012) define sistema embarcado (sistema embutido) como um sistema microprocessado, dedicado ao dispositivo ou aplicação que ele controla. Um sistema embarcado realiza funções predefinidas, ou seja, tarefas específicas. Esse difere do computador pessoal que realiza vários tipos de funções. Como vantagem tem-se a redução do custo no produto final, pois, através de engenharia, é possível otimizar seu tamanho, deixando-o modelado para o seu propósito.

Com exemplo prático e conhecido por muita gente, tem-se o sistema embarcado do microondas, responsável por todo o seu funcionamento. Ao pressionar a opção pipoca, o sistema deve saber ajustar a potência equivalente, selecionar e monitorar o tempo em que o aparelho deve ficar ligado e iniciar o motor que faz o prato girar. Para iniciar esse processo, o sistema também deve ser capaz de identificar, através do sensor da porta, se a mesma foi fechada. (SABERELETRONICA, 2008).

Mais alguns exemplos de sistemas embarcados:

- computadores de bordo automotivo;
- smartphones;
- PDAs;
- MP3 players;
- Impressoras;
- sistema de controle biométrico;
- controle de temperatura do ar-condicionado.

Esses sistemas são formados por quase todos os componentes de um computador pessoal, processador, memória, alguns dispositivos de armazenamento e interface por mais simples que seja. (MORIMOTO, 2007). Porém, algumas vezes, o sistema é executado com recursos limitados, sem teclado, sem mouse, sem tela e com pouca memória.



Conforme Morimoto (2007) descreve, esse software, escrito para esse tipo de sistema é, muitas vezes, chamado de firmware e armazenado em uma memória ROM ou memória flash ao invés de um disco rígido, como acontece no computador pessoal.

### 2.8.1 Arduino

O projeto Arduino surgiu na Itália, em 2005. Seu propósito era de ser usado em projetos escolares, substituindo os protótipos já usados que tinham um alto custo. Em 2006 ganhou o prêmio na categoria Comunidade Digitais, pela Prix Ars Eletrônica. No final de 2008, tinha 50.000 placas vendidas. (EMPRETEC, 2012).

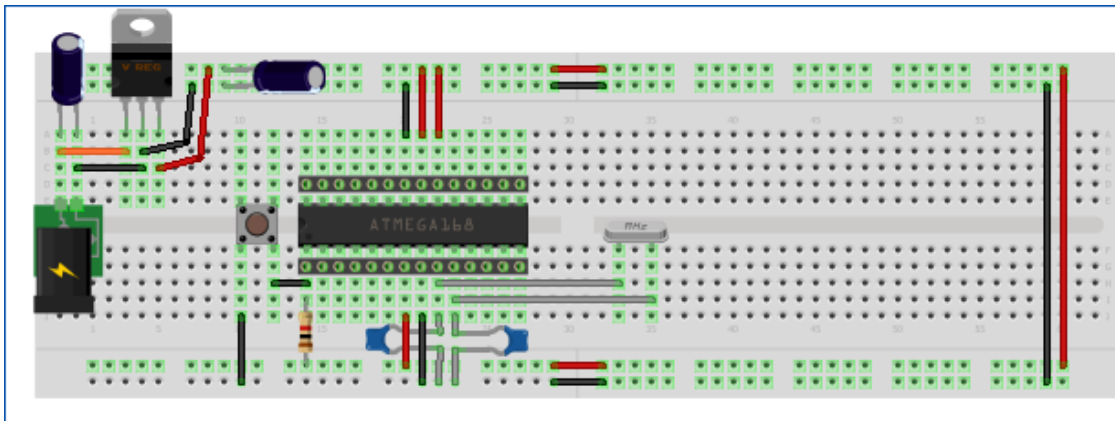
Segundo Empretec (2012), seu principal objetivo é facilitar a vida dos fanáticos por eletrônica, buscando facilitar a montagem de seus protótipos. O Arduino é baseado em um microcontrolador de placa única e um software para programá-lo.

Arduino é uma plataforma *open-source* de protótipo eletrônico, baseado em hardware, proporcionando flexibilidade e facilidade no manuseio. (VIEIRA, 2011). Seu Hardware é constituído por uma placa de circuito impresso com um microprocessador, pinos de entrada e saída (I/O) para gerenciamento de dispositivos eletrônicos, como: LED, servos, botões, sensores, atuadores e outros. Também possui memória RAM, memória Flash, temporizadores, contadores, dentre outras funcionalidades (JUSTEN, 2011).

Segundo Vieira (2011), existem várias opções de Arduino no mercado. Cabe ao usuário escolher a melhor opção para o seu projeto.

A imagem de um protótipo básico de Arduino é apresentada na Figura 36.

Figura 36 – Protótipo do Arduino.



Fonte :Vieira, 2012.

O Arduino Uno (Figura 37) é uma boa opção para quem está começando, com várias opções de acessórios.

Figura 37 – Arduino Uno.



Fonte: Vieira, 2011.

O Arduino Nano é parecido com o Uno, mas tem somente 1/3 do tamanho do Uno. É mais complexo quanto ao uso de dispositivos. (VIEIRA, 2011). O Nano pode ser visto na Figura 28.

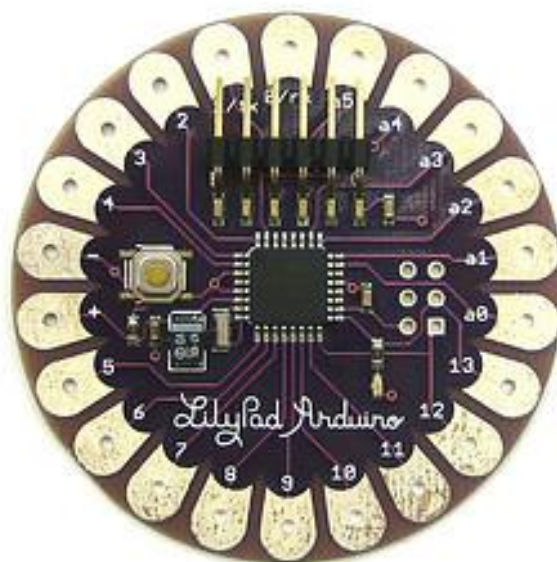
Figura 38 – Arduino Nano.



Fonte: Vieira, 2011.

Como opção para projetos *wearable* (do inglês “usável como vestimenta”), moda ou arte, foi desenvolvido o projeto Arduino Lilypad (Figura 39). Possui o formato de uma “vitória régia”, podendo ser costurado facilmente em tecidos.

Figura 39- Arduino Lilypad.



Fonte: Vieira, 2012.

Já, o Mega 2560, visível na Figura 40, possui mais vantagens em relação aos outros, tem mais memória e mais pinos I/O, entre outras melhorias. Por ser mais poderoso que os outros, é indicado para projetos maiores.

Figura 40 – Mega 2560.



Fonte: Vieira, 2011.

#### 2.8.1.1 Acessórios

Segundo Vieira (2011), os acessórios são conhecidos como Shields e adicionam funcionalidades específicas ao Arduino. As placas Shields podem ser adicionadas de acordo com a necessidade de seu projeto. Suas funcionalidades vão desde buscas na internet através da placa Ethernet até o controle remoto de um dispositivo via RF, utilizando para isso a Shield RF.

Uma das diferenças do Shield e da placa de Arduino é a falta de processamento, ficando esse processo concentrado somente no Arduino.

A Figura 41 é a imagem de um Shield Ethernet.



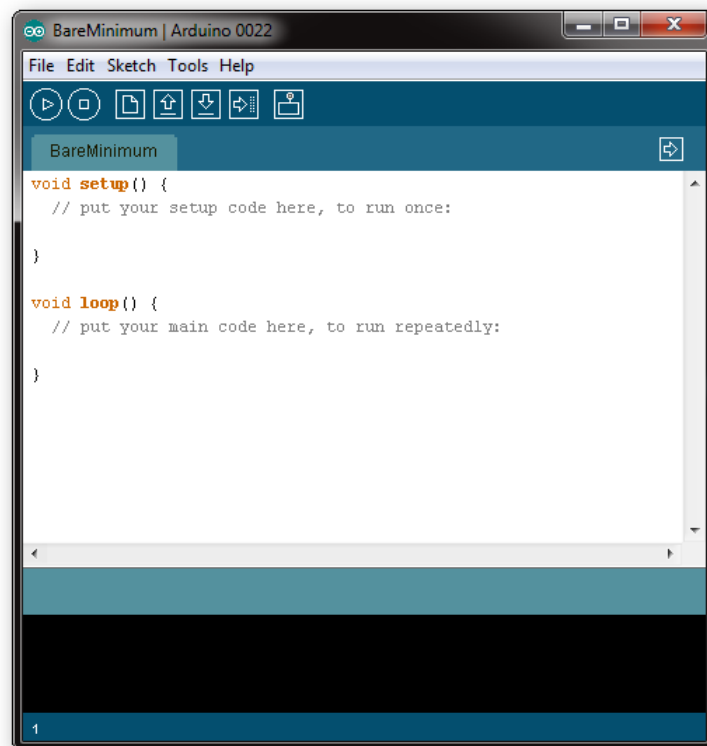
### 2.8.1.2 Software

Como qualquer sistema embarcado, o Arduino precisa de um software para executar suas funções. Para isso, o kit Arduino possui um ambiente de desenvolvimento integrado (IDE - *Integrated Development Environment*) de desenvolvimento multiplataforma desenvolvida em Java. Nela, podem ser desenvolvidos os códigos em C e C++ para a placa Arduino. Depois de escrito o código, é compilado e transferido via USB para a memória flash do Arduino. (JUSTEN, 2011).

Após a transferência do programa para o Arduino e sua alimentação na rede elétrica, se tudo estiver de acordo, o software é automaticamente executado.

Vieira (2011) informa que, além da própria IDE do Arduino (Figura 43), pode-se usar o Eclipse ou Notepad++ para desenvolver os códigos, porém esses deixam a desejar em quesito praticidade quanto à compilação e à documentação de exemplos prontos. A seguir, uma imagem da IDE:

Figura 43 – Tela da IDE.



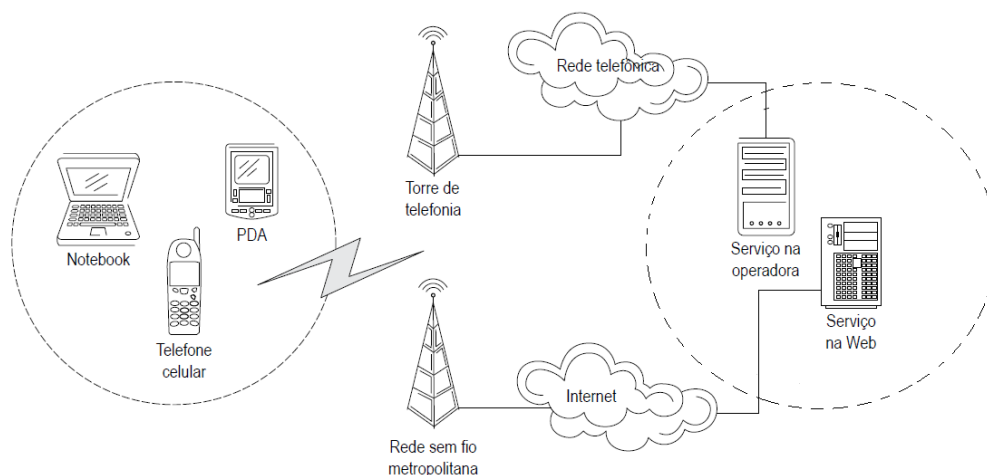
Fonte: Vieira, 2011.

## 2.9 DISPOSITIVOS MÓVEIS

Segundo definições em seu livro, Johnson (2007) define dispositivos móveis como o meio de prover serviços aos usuários independente do local que estejam. Para isso, itens como infraestruturas sem fios são necessárias.

A seguir, na Figura 44, pode ser observada a ilustração do cenário da computação móvel:

Figura 44 – Cenário da computação móvel.



Fonte: Johnson, 2007.

Entre os serviços “móveis”, estão o acesso à internet, envio de mensagens instantâneas, correio eletrônico, chats, entre outros. Mas as facilidades não se resumem às supracitadas, mais do que acessar à internet, os dispositivos móveis podem prover serviços aos seus usuários, ajudando no seu meio profissional ou social. Uma pessoa pode efetuar uma compra via celular, o médico pode visualizar a ficha do seu paciente no seu computador de bolso, uma aplicação de automação residencial pode ser controlada via tablet, seja em qual cômodo o usuário estiver. (JOHNSON, 2007).

Alguns exemplos desses dispositivos podem ser vistos na figura 35.



Figura 45 – Exemplos de dispositivos móveis.



Fonte: Porfírio, 2012.

Conforme matéria na Exame (2012) , representantes de operadoras, fornecedores e indústria decretaram a falência do computador desktop, alguns afirmando que ainda nesta geração e outros na próxima. Isso, devido ao incremento no mercado de tablets e smartphones, cada vez mais poderosos.

Porfírio (2012) relata que, segundo pesquisa da Accenture, os smartphones tornaram-se o meio mais popular de acessar à internet e que o Brasil e a África do Sul são os países que mais acessam à internet por meio de dispositivos móveis. Descreve, ainda, que 78% dos brasileiros pretendem adquirir um smartphone em um futuro próximo, bem acima da média mundial (40%). Outros dados da pesquisa relatados por Porfírio (2012) são:

- 69% dos usuários de internet acessaram à rede por um dispositivo móvel;
- 61% usaram um smartphone, 37% um netbook e 22% um tablet;
- 58% usam por motivos pessoais, enquanto 20% para assuntos relacionados ao trabalho;
- 45% das pessoas acima dos 50 anos utilizam a internet móvel;
- 62% acessam comunidades online, como o Facebook;
- 46% já fizeram transações bancárias a partir de seu dispositivo móvel;
- 70% ainda têm preocupações em relação à segurança;
- 36% já acessaram à Internet por meio de uma TV e 27% por meio de um console.



### 2.9.1 Desenvolvimento para Dispositivos móveis

Varias podem ser as linguagens de programação para aplicações voltadas aos dispositivos móveis, algumas mais simples, com menos recursos e outras mais avançadas. As linguagens, C++, Visual Basic (VB) e Java proporcionam um desenvolvimento mais dinâmico, voltado para aplicações mais complexas. (JOHNSON, 2007). Um fator discriminante é o Hardware, que define basicamente a linguagem de programação suportada. Outro fator está relacionado com os recursos disponibilizados pelo fabricante, exemplo:

- **Tela do Dispositivo:** As aplicações devem ser desenvolvidas, levando em consideração o tamanho reduzido das telas. Isso demanda tempo e um bom projeto. A divisão de elementos gráficos como caixas de texto, imagens, comandos e listas é indispensável.
- **Recursos Físicos:** Por se tratar de dispositivos móveis, em alguns aparelhos, alguns recursos são “enxutos”, como o caso do processador e da memória. O uso de *threads* permite que o processamento de uma tarefa seja executado em paralelo com o gerenciamento do aplicativo.

Enfim, tentar alterar um projeto depois de pronto é sempre mais trabalhoso do que tomar precauções no início do desenvolvimento, sem contar que o custo também aumenta e muito, tornando certos projetos inviáveis comercialmente.

### 2.9.2 Android

O Android foi criado pela OHA (*Open Handset Alliance*) em resposta a evolução tecnológica móvel e com o objetivo de satisfazer aos usuários. Trata-se de uma plataforma de desenvolvimento para aplicativos móveis, construída com base no sistema operacional Linux, possuindo várias aplicações já instaladas (LECHETA, 2011). O logotipo do Android pode ser visto na Figura 46.

Figura 46 – Logotipo do Android.



Fonte: Sam, 2012.

O grupo OHA é formado pelos gigantes do mercado de telefonia celular liderados pelo Google. LG, Motorola, Samsug, Sony, Ericsson, Sprint Nextel são alguns dos integrantes desse grupo.

O primeiro celular, lançado com a versão do Android, foi o T-Mobile G1 (Figura 47). Superando suas expectativas de venda, seu lançamento entusiasmou todos os seus usuários.

Figura 47 – T-Mobile G1.



Fonte: Exolink, 2012

Dentre as facilidades providas por essa plataforma está o acesso à internet, a tela *touch screen* (sensível ao toque), o reconhecimento de códigos de barra e acesso aos recursos do Google *maps*. (LECHETA, 2011).

Segundo Rabello (2012), como a plataforma foi desenvolvida em Linux, ela conta com todas as características desse sistema, sistemas de arquivo, o Kernel, os servidores de terminais, entre outros. Rabello (2012), ainda, cita outras características dessa plataforma como:

- Framework de Aplicação: permite reuso de componentes;
- Máquina Virtual Dalvik: otimizada para dispositivos móveis;
- Gráficos Otimizados: isso graças as bibliotecas de gráficos 2D e gráficos 3D baseados na especificação OpenGL 1.0;
- SQLite: para armazenamento de dados, na forma de estruturas relacionais;
- Suporte para Mídias: áudio (MP3, AAC, AMR), vídeo (MPEG4 e H.264) e imagens (JPG, PNG, GIF);
- Poderoso ambiente de desenvolvimento: inclui emulador de dispositivo, ferramentas para debug, analisador de memória e desempenho, plugin para IDE Eclipse.

Algumas outras características dependem de Hardware, por exemplo:

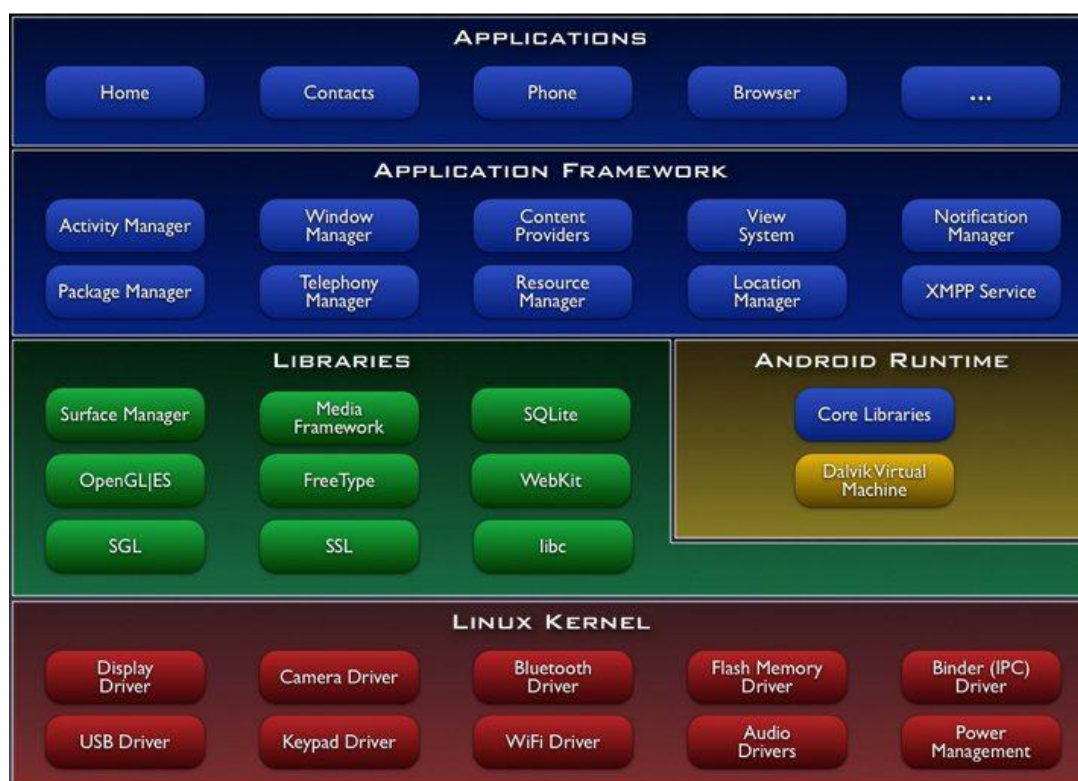
- telefonia GSM;
- bluetooth, 3G e WiFi;
- câmera, GPS, bússola e acelerômetro.

Segundo Lecheta (2011), o Android foi desenvolvido pensando na satisfação do cliente final, com o objetivo de ser uma plataforma única e aberta, além de proporcionar uma plataforma moderna e flexível para o desenvolvimento de aplicações corporativas.

#### 2.9.2.1 Arquitetura

Segundo Rabello (2012), a arquitetura do Android está dividido nas seguintes camadas: *Applications*, *Application Framework*, *Libraries*, *Runtime* e *kernel*, apresentadas na Figura 48.

Figura 48 – Arquitetura da plataforma Android.



Fonte: Rabello, 2012.

É na camada da *Applications* que se encontram as aplicações disponibilizadas para o usuário final, todas elas desenvolvidas em linguagem Java. Já, na camada *Application Framework*, estão os componentes utilizados pelos grupos de desenvolvedores na criação dos aplicativos. Esses componentes ajudam no desenvolvimento das estruturas; tais componentes podem ser: listas, *grids*, caixas de texto, botões, entre outros (RABELLO, 2012). Ainda, nessa camada estão concentrados os gerenciadores de recursos, mensagens e atividades, além do provedor de conteúdo.

Abaixo da camada *Application Framework*, estão o *Runtime* e a *Libraries*. Na primeira, respectivamente, está instalada a máquina virtual Dalvik e, na segunda, as bibliotecas escritas em C/C++. Por fim, o *Kernel (núcleo)*, disponível na versão 2.6 para Android.

Segundo Tosin (2011), é no *Kernel* que se encontra o sistema operacional da plataforma, responsável por gerenciar os recursos de baixo nível, como memória, processos e segurança.

Segundo Rabello (2012), as principais APIs da plataforma estão apresentadas no Quadro 1:

Quadro 1 – Principais APIs.

Pacote	Descrição
<b>android.util</b>	Contém várias classes utilitárias (classes de containers, utilitários XML)
<b>android.os</b>	Contém serviços referentes ao sistema operacional, passagem de parâmetros e comunicação entre processos.
<b>android.graphics</b>	Pacote principal dos recursos gráficos.
<b>android.text</b> <b>android.text.method</b> <b>android.text.style</b> <b>android.text.util</b>	Suporte para um conjunto de ferramentas de processamento de texto, suporte ao formato de texto rico (RTF), métodos de entradas, etc.
<b>android.database</b>	Contém APIs para comunicação com o banco de dados SQLite
<b>android.content</b>	APIs de acesso a dados no dispositivo, como as aplicações instaladas e seus recursos.
<b>android.view</b>	O pacote principal que contém os principais componentes de interface gráfica.
<b>android.widget</b>	Contém widgets prontos (botões, listas, gerenciadores de layout, etc) para serem utilizados nas aplicações
<b>android.app</b>	APIs de alto-nível referentes ao modelo da aplicação. É implementada por meio de Activities (Atividades)
<b>android.provider</b>	Contém várias APIs para padrões de provedores de conteúdos (content providers)
<b>android.telephony</b>	APIs para interagir com funcionalidades de telefonia
<b>android.webkit</b>	Inclui várias APIs para conteúdos de context web, bem como um navegador embutido que pode ser utilizado por qualquer aplicação.

Fonte: Rabello, 2012.

### 2.9.2.2 Componentes de uma Aplicação

Segundo Tosin (2011), desenvolver aplicações significa criar uma série de componentes com um objetivo bem definido, uni-los para ter, como produto final, uma aplicação. Na Figura 49, são apresentados os componentes que fazem parte da plataforma Android.

Figura 49 – Componentes da Plataforma Android.



Fonte: Tosin, 2011.

➤ *Activities* (Atividades): de acordo com Tosin (2011), são as representações da tela da aplicação. A *view* é quem define como será feita a exibição visual da imagem e, geralmente, está associada com a *Activities*. Também, é papel desse componente o gerenciamento dos eventos de telas decorrentes na aplicação.

Para Rabello (2012), uma *Activities* representa uma tela em branco, que pode ser desenvolvida, após estender uma classe *Activity*. Nas aplicações reais, as *Activities* são as telas que compreendem um sistema, como, por exemplo, as páginas de uma agenda telefônica.

➤ *Services*: responsável pelas tarefas executadas em segundo plano, ou seja, serviços executados, muitas vezes, por trás da tarefa principal. (TOSIN, 2011).

➤ *Content Providers*: os provedores de conteúdo facilitam na integração de dados entre aplicações. Tosin (2011) cita o exemplo de uma aplicação desenvolvida por terceiros precisar dos dados da agenda nativa do Android. Nesse pode-se usar o *Content Providers* para facilitar o acesso a esses dados.

➤ *Broadcast Receivers*: associada com a ocorrência de eventos. Seu papel é escutar e avisar sobre a ocorrência de eventos, quando solicitada por alguma aplicação, seja qual for sua fonte. (TOSIN, 2011).

- *AndroidManifest.xml*: considerado por Lecheta (2011) o principal arquivo do projeto. É nele que estão armazenadas todas as configurações da aplicação.

### 2.9.2.3 Desenvolvimento para Android

Desenvolver para Android não é um mistério como algumas pessoas pensam. Inicialmente, é necessário ter as ferramentas certas instaladas. (TOSIN, 2011).

- Java SDK (JDK): primeira ferramenta a ser instalada. A versão mínima recomendada é o Java 5.
- Android SDK: é o kit de desenvolvimento para aplicações Android. Vai desde o modulo de compilação até os emuladores. Graças ao emulador (Figura 36), é possível executar a aplicação no próprio desktop.
- Eclipse: permite o desenvolvimento de códigos. O Eclipse não é a única ferramenta disponível para esse tipo de finalidade. Mas, caso seja usada, é necessário instalar o plug-in ADT (*Android Development Tools*).

A Figura 50 apresenta o emulador gerado a partir do Eclipse.

Figura 50 – Emulador simulando uma aplicação.



Fonte: Tosin, 2011.

#### 2.9.2.4 Máquina Virtual

Na plataforma Android, não existe uma máquina virtual nativa Java (*Java Virtual Machine*) que atua como interpretador do código intermediário. Existe, sim, uma máquina virtual chamada Dalvik, desenvolvida de forma otimizada, projetada de acordo com os recursos disponíveis nos dispositivos móveis. (LECHETA, 2011). O código desenvolvido em java é compilado, formando um arquivo com a extensão .dex (*Dalvik Executable*), em seguida, é compactado junto com os demais recursos, como imagens e áudio, formando, por fim, um arquivo .apk (Android Package File). Nesse formato, o arquivo está pronto para ser distribuído e instalado.



### 3 MÉTODO DE PESQUISA

Para Wazlawick (2009), o fato da área de computação ou informática estar em constante desenvolvimento não justifica que o método científico específico da área de Computação tenha de ser vago e que tantas monografias sejam escritas sem conhecimento metodológico apropriado para o assunto. Segundo o autor, metodologia seria o estudo dos métodos, definido como: “O método consiste na sequência de passos necessários para demonstrar que o objetivo proposto foi atingido, ou seja, se os passos definidos no método foram executados, os resultados obtidos deverão ser consistentes.”

Segundo Rampazzo (2005), método é o conjunto de técnicas de investigação ou conjunto de procedimentos utilizados por uma determinada ciência no processo de pesquisa.

#### 3.1 CARACTERIZAÇÃO DO TIPO DE PESQUISA

Através da pesquisa, é possível descobrir novos fatos, dados e soluções em qualquer área do conhecimento. Para isso, faz-se necessário o uso de procedimentos sistemáticos, de observação e investigativos. Enfim, a pesquisa tem como objetivo o encontro de soluções para problemas por meio dos processos do método científico. (RAMPAZZO, 2005). A seguir, os três elementos que caracterizam a pesquisa conforme o autor:

- levantamento do Problema;
- a solução a qual se chega;
- meios escolhidos para alcançar a solução, instrumentos científicos e procedimentos adequados.

Para Rampazzo (2005), a curiosidade do homem pelo aprendizado e descoberta de novas ideias levam-no a investigar os assuntos sob vários aspectos de abordagem. Porém cada uma com foco em técnicas específicas, conforme o interesse de estudo, por isso, a existência de vários tipos de pesquisa.

Segundo Rodrigues (2007) os tipos de pesquisa são classificadas segundo:

- a área da ciência;
- a natureza;
- aos objetivos;
- aos procedimentos;
- ao objeto;
- a forma de abordagem.

No Quadro 2, é apresentada a classificação detalhada dos tipos de pesquisa:

Quadro 2 – Tipos de pesquisa.

<b>Tipos de Pesquisa</b>	
<ul style="list-style-type: none"> <li>• <b>Classificação quanto:</b> <ul style="list-style-type: none"> <li>– À área da ciência <ul style="list-style-type: none"> <li>• Pesquisa teórica</li> <li>• Pesquisa metodológica</li> <li>• Pesquisa empírica</li> <li>• Pesquisa prática</li> </ul> </li> <li>– À natureza <ul style="list-style-type: none"> <li>• Trabalho científico original</li> <li>• Resumo de assunto</li> </ul> </li> <li>– Aos objetivos <ul style="list-style-type: none"> <li>• Pesquisa exploratória</li> <li>• Pesquisa descritiva</li> <li>• Pesquisa explicativa</li> </ul> </li> </ul> </li> </ul>	
	<ul style="list-style-type: none"> <li>– Aos procedimentos <ul style="list-style-type: none"> <li>• Pesquisa de campo</li> <li>• Pesquisa de fonte de papel</li> </ul> </li> <li>– Ao objeto <ul style="list-style-type: none"> <li>• Pesquisa bibliográfica</li> <li>• Pesquisa de laboratório</li> <li>• Pesquisa de campo</li> </ul> </li> <li>– À forma de abordagem <ul style="list-style-type: none"> <li>• Pesquisa quantitativa</li> <li>• Pesquisa qualitativa</li> </ul> </li> </ul>

Fonte: Rodrigues, 2007.

Em relação à área da ciência, essa pesquisa encaixa-se no tipo de pesquisa prática ou aplicada, pois seu principal objetivo é prover um protótipo com base em um modelo proposto para aplicação em automação residencial. “Os conhecimentos da pesquisa aplicada são utilizados para aplicação prática voltados para a solução de problemas concretos da vida moderna”. (RODRIGUES, 2007).

Quanto à natureza da pesquisa, trata-se de uma pesquisa de resumo de assunto, cuja solução é desenvolvida com foco em outras soluções propostas, conforme descreve Rampazzo (2005, p. 49):

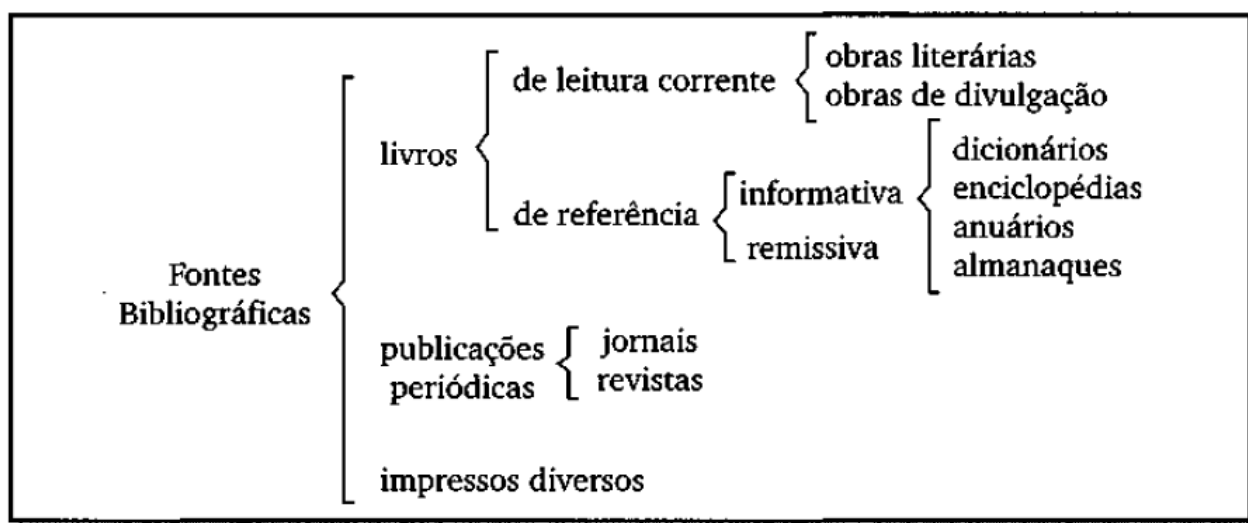
O objetivo dos iniciantes é a aprendizagem e o treino das técnicas de investigação, refazendo o caminho percorrido pelos pesquisadores. O resumo de assunto de uns segue a trilha dos trabalhos científicos originais de outros. Entende-se por trabalho científico original aquela pesquisa cujos resultados venham apresentar novas conquistas para determinada área do saber. Trata-se, pois de uma pesquisa sobre um determinado assunto levada a efeito pela primeira vez.

No que se refere ao seu objetivo, à utilização da pesquisa exploratória é a que melhor se adequa a este trabalho referente à busca de conhecimentos específicos sobre a teoria do tema, através de bibliografias. Ciribelli (2003) descreve a pesquisa exploratória como sendo o primeiro passo para a elaboração de qualquer trabalho científico. Proporciona ao pesquisador adquirir maiores informações sobre o tema abordado, ajudando também na definição dos seus objetivos, delimitações, formas de trabalho e desenvolvimento do assunto.

Em se tratando de procedimentos, foi escolhida a pesquisa chamada fonte de papel, prevalecendo como fonte de informações documentos impressos como: livros, documentos, artigos, teses, e outros. Segundo Gil (2002), “o elemento mais importante para a identificação de um delineamento é o procedimento adotado para a coleta de dados. Assim, podem ser definidos dois grandes grupos de delineamentos: aqueles que se valem das chamadas fontes de “papel” e aqueles cujos dados são fornecidos por pessoas”. Na pesquisa fonte de papel, estão inclusas a pesquisa bibliográfica e a pesquisa documental.

A pesquisa bibliográfica foi o objeto de pesquisa escolhido por se tratar de uma pesquisa mais centralizada na obtenção de informações, com base em materiais já elaborados. Esse tipo de pesquisa visa à leitura de artefatos como livros e artigos científicos, assim como documentações referentes às soluções paralelas. (GIL, 2002). No quadro 3, são apresentados os tipos de fontes bibliográficas.

Quadro 3 – Fontes Bibliográficas.



Fonte: Gil, 2002.

Em relação à solução do problema, trata-se de uma pesquisa qualitativa, baseada na apresentação de uma forma diferente de resolver certo tipo de problema. Neste tipo de pesquisa, as informações obtidas não podem ser quantificáveis. (WAZLAWICK, 2009).

Como forma de levantamento de dados foi utilizada o estudo de caso. Esse estudo descreve o fato observado no segmento empresarial, suas características e componentes, analisando-os com objetivo de prover solução para o problema inicialmente levantado. O estudo de caso, ou *case*, é a técnica que mais se alia à pesquisa de natureza qualitativa. (SANTOS et al., 2006).

Para Gil (2002), estudo de caso é aquele que constitui o próprio objeto de pesquisa.

### 3.2 ETAPAS

Em um projeto de pesquisa, deve-se ter claro quais são as etapas que serão desenvolvidas e quais os recursos que devem ser reservados para a obtenção do seu objetivo. Também, deve ser suficientemente detalhado para uma clara avaliação do processo de pesquisa (GIL, 2002).

As etapas para o desenvolvimento e apresentação da monografia são:

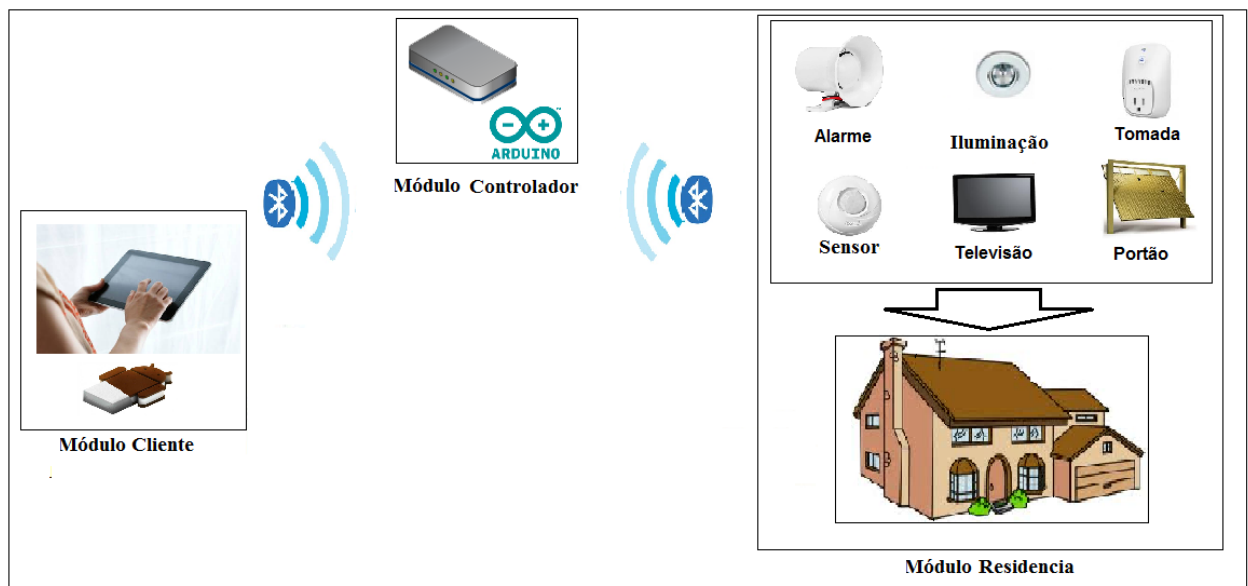
- 1 – formulação do problema;
- 2 – busca dos dados teóricos relativos ao objeto de pesquisa;
- 3 – levantamento de informações sobre o estado da arte;
- 4 – modelagem;
- 5 – aquisição de Hardware Arduino, dispositivos (sensores, atuadores, motores e demais dispositivos eletrônicos) e plataforma *mobile* Android (*tablet* ou *Smartphone*) para o desenvolvimento do projeto.
- 6 – criação da base de desenvolvimento;
- 7 – desenvolvimento;
- 9 – testes de desenvolvimento;
- 10 – construção da Maquete (casa) para simulação de testes;
- 11 – testes com protótipo e validação;
- 12 – finalização da monografia e defesa.

Estas etapas são melhor visualizadas no cronograma (Apêndice A) elaborado, organizadas por atividade e período.

### 3.3 ARQUITETURA DA SOLUÇÃO PROPOSTA

Na Figura 51, segue a proposta da solução do problema descrita no projeto, assim como as tecnologias escolhidas.

Figura 51: Arquitetura macro do projeto.



Fonte: Elaboração do Autor, 2012.

A arquitetura referente à solução física do problema está composta por três módulos, descrita a seguir:

- **Módulo cliente:** nesse nível, estão os dispositivos móveis, *smartphones* e *tablets*, que através do software desenvolvido para a plataforma Android com conexão *wi-fi* acessam o módulo controlador.
- **Módulo Controlador:** hardware Arduino responsável por interagir diretamente com os componentes eletrônicos do módulo residência, controlando assim seus estados de acionamento e monitoramento.
- **Módulo Residência:** residência que contém os componentes gerenciados pelo sistema de automação residencial, componentes esses como: motores para abertura de cortinas, sensores, alarmes, iluminação e outros.

### 3.4 DELIMITAÇÕES

O projeto não será aplicado em uma residência, ou seja, nessa etapa da monografia, o hardware e os demais componentes eletrônicos da aplicação serão instalados em uma maquete (casa), simulando um ambiente residencial. O controle dos dispositivos automatizados pelo controlador se dará apenas via sistema operacional Android, através de conexão via Bluetooth, ficando limitado a conexões remotas, via internet ou a outros meios de comunicação.

Sobre a comunicação via Bluetooth, não é foco do trabalho, estudar e nem aplicar conceitos de segurança.

Não serão aplicados conceitos de Inteligência Artificial no desenvolvimento do projeto, fazendo-se necessário a programação manual de qualquer regra de automação ou nova funcionalidade.

Não será foco deste protótipo desenvolver um sistema de segurança que envolva controle de acesso e imagens via câmera de segurança. O sistema de segurança envolverá somente o uso de alarme via sensor de presença.

A aplicação de sensores ficará limitada ao controle de alarme, iluminação e tomadas.

Não será desenvolvido nenhum processo relacionado ao monitoramento de recursos como energia, água e gás.

Em relação à parte de hardware, não é objetivo deste trabalho, desenvolver e nem criar novos circuitos ou módulos eletrônicos, e, sim, à integração dos módulos já encontrados disponíveis no mercado necessários à criação do protótipo para a automatização dos dispositivos residenciais que serão apresentados no capítulo da modelagem.

## **4 MODELAGEM**

Neste capítulo, será abordado o modelo de solução para o problema proposto através de uma visão geral da solução projetada. Segundo Ramos (2006), modelagem é a arte de criar modelos para uma determinada realidade a qual se deseja analisar, além de facilitar e prover a comunicação entre várias pessoas. Esses modelos podem ser representados por um tipo de linguagem, utilizando, para isso, estruturas na forma de diagramas.

### **4.1 SOFTWARE**

Nesta seção, serão apresentados alguns diagramas da modelagem de software para o protótipo proposto pelo projeto.

#### **4.1.1 UML – Linguagem de Modelagem Unificada**

A UML é uma linguagem que serve para especificar, construir, visualizar e documentar artefatos de sistemas de software. Esta linguagem é independente das ferramentas e da linguagem de programação utilizada (RAMOS, 2006).

#### **4.1.2 Atores**

De acordo com Ramos (2006), o conceito de ator está relacionado, em geral, com o papel que um usuário desempenha no sistema em análise, porém pode representar qualquer dispositivo eletrônico que possa realizar interações com o sistema. Dentro da notação UML, os atores são representados por figuras de bonecos.



Para a solução proposta por este trabalho, o usuário será o único ator, conforme descrito na Figura 52.

Figura 52 – Ator do Sistema.



Fonte: Elaboração do Autor, 2012.

Cada morador da residência, que pode interagir com o sistema, será definido como usuário do sistema. O Perfil do usuário pode ser considerado como o de uma pessoa com habilidade para o uso de dispositivos móveis, idade e escolaridade variada (pode ser um adulto ou menor de idade que mora na residência) e de classe média.

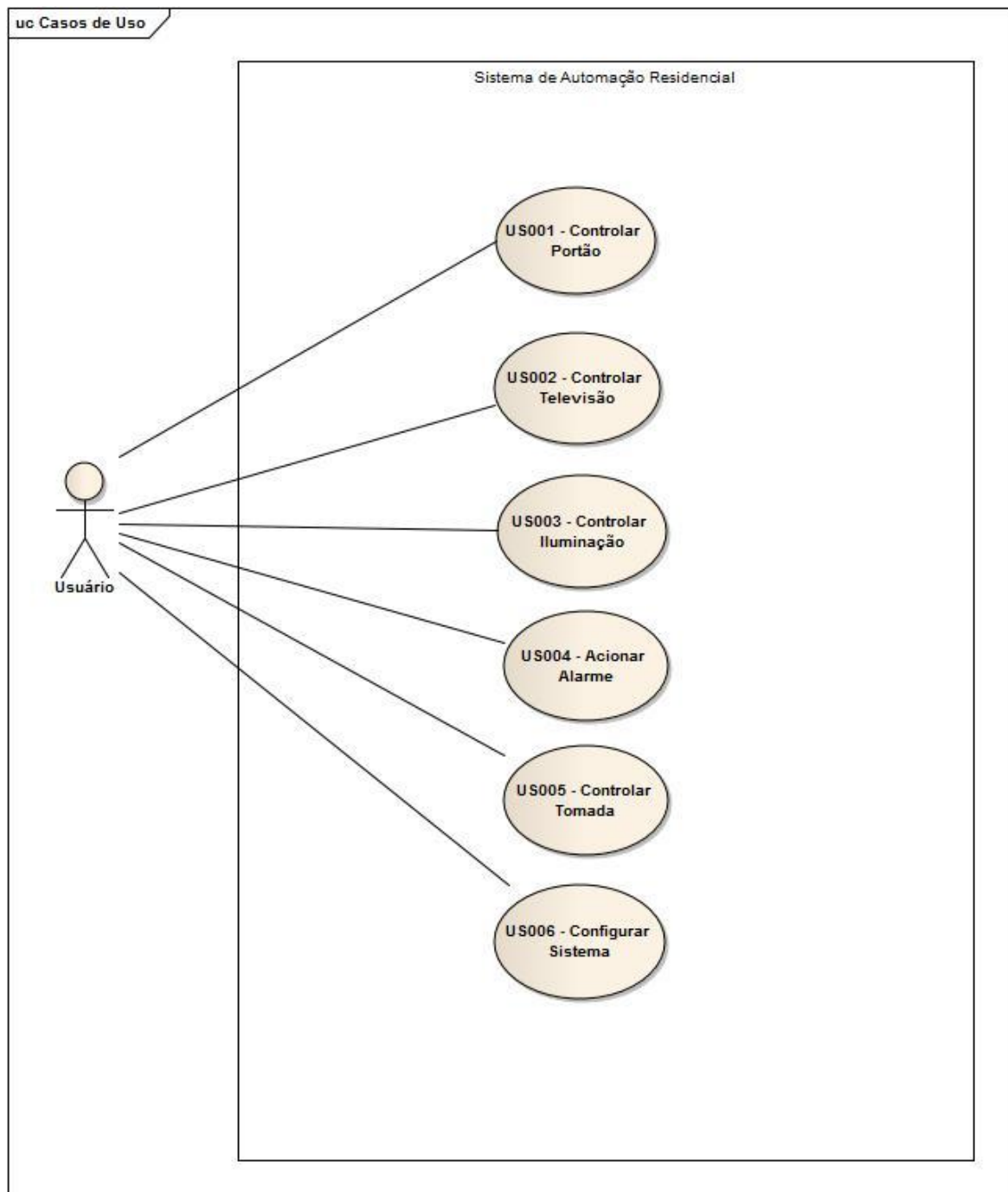
As atividades executadas pelo ator no sistema são descritas no modelo de requisitos, juntamente com os casos de uso.

#### 4.1.3 Casos de Uso

Bezerra (2002) define o modelo de casos de uso como sendo uma representação das funcionalidades do sistema, executadas pelos elementos externos, neste caso os atores. Este modelo define o relacionamento entre cada funcionalidade do sistema (caso de uso) e o ator.

Na Figura 53, é apresentado o diagrama de caso de uso para a solução proposta neste trabalho.

Figura 53 – Diagrama de caso de uso.



Fonte: Elaboração do Autor, 2012.

#### 4.1.4 Requisitos

Os requisitos, segundo as definições de Ramos (2006), são as características, propriedades ou comportamentos definidos para um produto. O requisito nada mais é do

que um contrato estabelecido entre ator e sistema, deixando claro o que se espera que o sistema faça.

Conforme Martins (2007), os requisitos podem ser divididos em requisitos funcionais e não funcionais.

#### 4.1.4.1 Requisitos Funcionais

Martins (2007, p 209) afirma que “os requisitos funcionais são aqueles que definem o comportamento do sistema, capturados através de casos de uso, que documentam as entradas, os processos e a saída gerada”.

Os requisitos funcionais do sistema foram levantados, com base na problemática e consequentemente com os objetivos do projeto, aliados ao estudo sobre o estado da arte na seção 2.1.

No Quadro 4, são listados os requisitos funcionais do sistema.

Quadro 4 – Requisitos Funcionais.

Referência	Nome	Descrição
RF001	Controlar Portão.	O Sistema deve permitir ao usuário controlar um portão eletrônico de sua residência.
RF002	Controlar Tv.	O Sistema deve permitir ao usuário controlar uma Televisão de sua residência.
RF003	Controlar Iluminação.	O Sistema deve permitir ao usuário controlar a iluminação de sua residência.
RF004	Acionar alarme.	O sistema deve permitir controlar o acionamento do alarme de sua residência
RF005	Acionar Tomada.	O Sistema deve permitir ao usuário controlar o acionamento de energia em uma determinada tomada.
RF006	Configurar Sistema.	O Sistema deve permitir ao usuário configurar algumas ações no sistema.

Fonte: Elaboração do autor, 2012.

#### 4.1.4.2 Requisitos Não Funcionais

Os requisitos não funcionais geralmente estão relacionados com as tecnologias utilizadas, assim como as arquiteturas empregadas. A outra parte dos requisitos são pertinentes a pequenos grupos de casos de uso, restringindo dessa maneira como a funcionalidade será implementada. Todas elas relacionadas com a qualidade do sistema. (MARTINS, 2007).

Para o sistema apresentado, os requisitos não funcionais são apresentados no Quadro 5.

Quadro 5 – Requisitos Não Funcionais.

Referência	Nome	Descrição
RNF001	Idioma	O sistema apresentará todo o seu conteúdo gramatical na língua portuguesa.
RNF002	Botões	O sistema apresentará seus botões com imagens para cada tipo de funcionalidade.
RNF003	Mensagens	Para cada comando executado, o sistema responderá com mensagens de aviso.
RNF004	Telas	Para cada serviço, o sistema disponibilizará uma tela diferente.
RNF005	SGDB	O Sistema gerenciador de Banco de Dados utilizado é do próprio framework Android.

Fonte: Elaboração do autor, 2012.

#### 4.1.5 Regras de Negócio

Considerado como informação essencial para o processo de modelagem do sistema, as regras de negócio são consideradas como condições, restrições e até mesmo como políticas para a execução de cada rotina em um processo existente. (BEZERRA, 2002).

As regras de negócio identificadas para o sistema estão descritas no Quadro 6.

Quadro 6 – Regras de Negócio.

Referência	Nome	Descrição
RN01	Acionamento Manual do Portão Eletrônico.	<p>O acionamento do Fechamento do portão pelo usuário deve ser feito de forma manual, através de um botão Abrir/Fechar.</p> <p>Deve ser possível controlar, ao menos, um portão eletrônico.</p> <p>O acionamento Manual deverá ser realizado independente do tempo de fechamento configurado.</p>
RN02	Fechamento Automático do Portão Eletrônico.	<p>O Fechamento automático do portão deve ser realizado com base em um determinado tempo após a abertura do portão.</p>
RN03	Controle da Televisão.	<p>O controle da televisão deve conter os seguintes comandos:</p> <p>Ligar e Desligar.</p> <p>Aumentar e Diminuir o Canal.</p> <p>Aumentar e Diminuir o Volume.</p>
RN06	Acionamento da Lâmpada por Sensor.	<p>A lâmpada deverá ser acesa automaticamente com base no sensor de presença. Caso o sensor não detecte outro movimento durante um determinado tempo, a lâmpada deve ser desligada.</p> <p>Deve ser possível controlar ao menos um ponto de iluminação neste modo.</p>
RN07	Acionamento Manual da Lâmpada.	<p>O acionamento da iluminação deverá ser feita de modo manual, havendo, para isso, um botão na interface ligar/desligar.</p> <p>Deve ser possível controlar ao menos três pontos de iluminação nesse modo.</p>

RN08	Acionamento Automático da iluminação.	<p>O acionamento da iluminação deverá ser feita de forma automática, ou seja, deverá ser feito com base em um horário para ligar e outro para desligar.</p> <p>Deve ser possível controlar, ao menos, um ponto de iluminação neste modo.</p>
RN09	Acionar alarme.	O acionamento do alarme deverá ser feito de forma manual pelo usuário.
RN10	Acionar tomada.	<p>Esta função deverá permitir que o usuário controle de forma manual ao menos um ponto de energia. Este ponto de energia poderá ser usado para controlar um ventilador, carregador de celular ou qualquer outro dispositivo ligado à tomada.</p>
RN11	Configuração do Sistema.	<p>O sistema deverá conter a opção de configurar o tempo em que o portão eletrônico deverá fechar.</p> <p>O sistema deverá conter a opção em que a iluminação de determinado ponto deverá ficar acesa sem que o sensor detecte outro movimento.</p> <p>O Sistema deverá conter a opção para configurar a hora inicial e a hora final para acionamento da iluminação no modo automático.</p> <p>O Sistema deverá conter a opção para que o modo de iluminação possa ser alterado.</p>

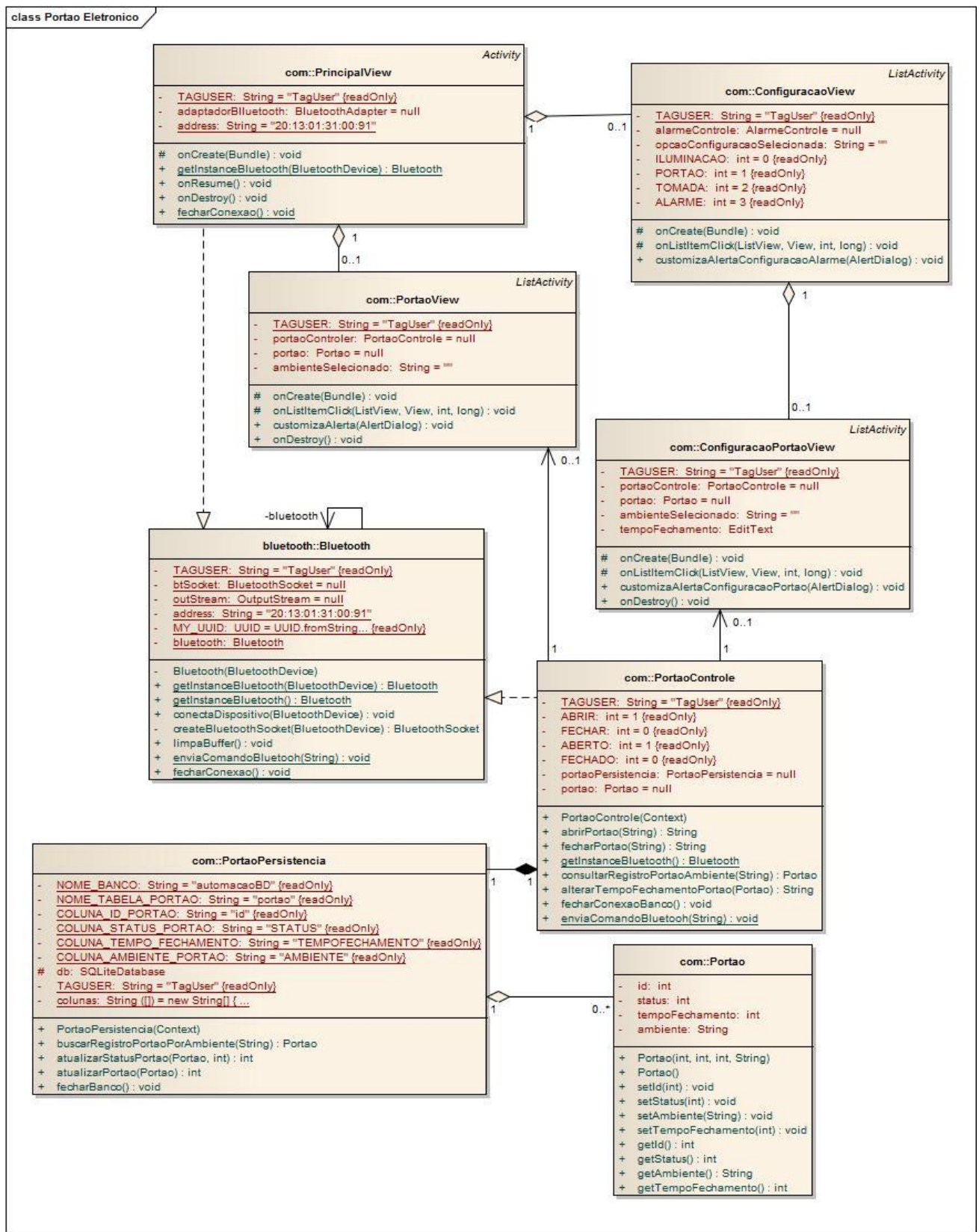
Fonte: Elaboração do autor, 2012.

#### **4.1.6 Diagrama de Classe**

O diagrama de classe tem como objetivo organizar e documentar a estrutura do sistema. “Os Diagramas de classe são utilizados para representar a estrutura estática do sistema, composta pelas classes de negócio, classes de interface com o usuário e com outros sistemas e as classes de controle, responsáveis pelo controle de transações”. (MARTINS 2007, p. 435).

Na Figura 54, é apresentado o diagrama de classe para o controle do portão eletrônico.

Figura 54 – Diagrama de classe do Controle do Portão Eletrônico.



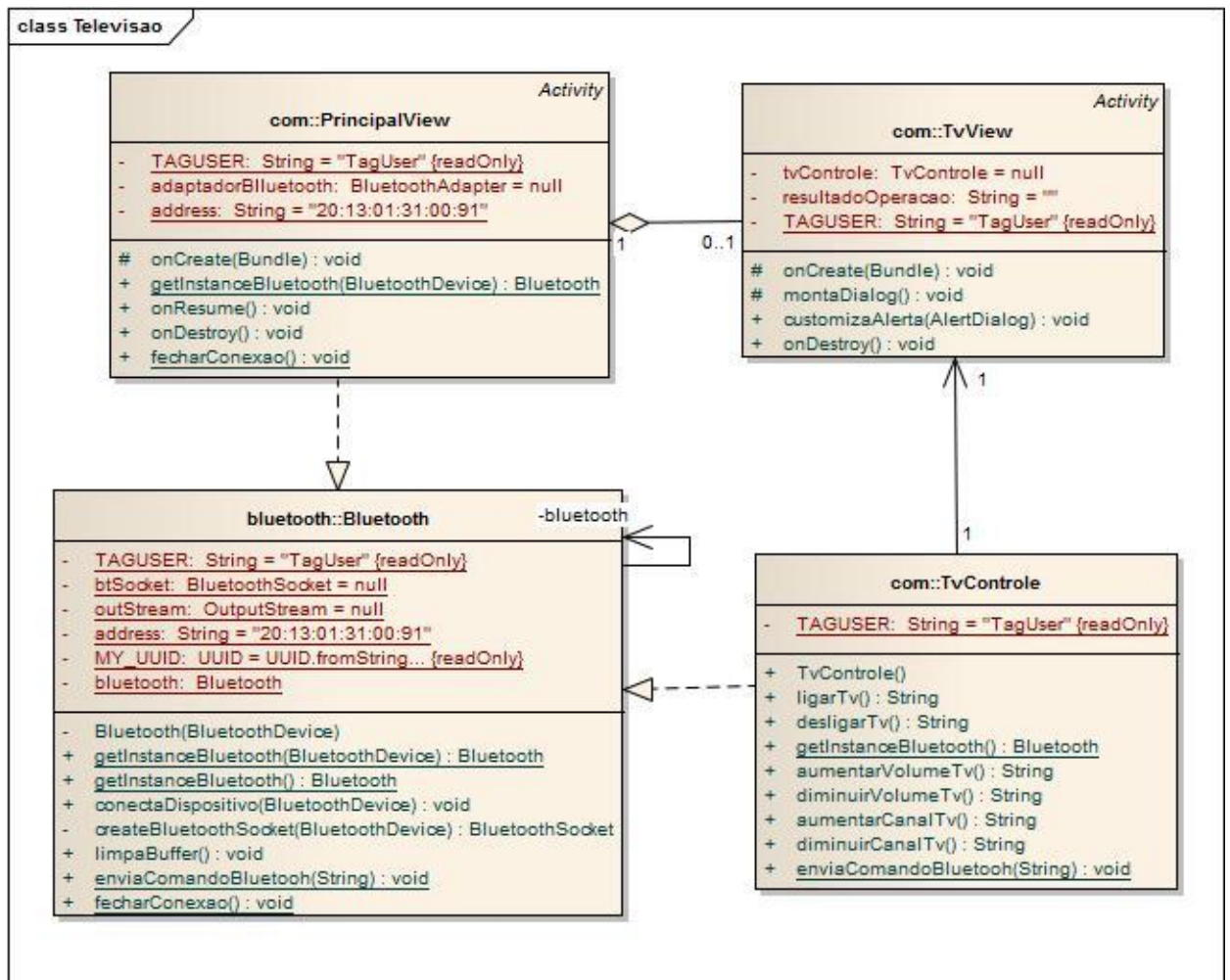
Fonte: Elaboração do autor, 2012.



A Figura 55 representa a modelagem do diagrama de classe destinado ao controle da Televisão, assim como as classes responsáveis pela comunicação Bluetooth.

A modelagem referente à comunicação do Bluetooth segue para os demais diagramas de classes, que não serão exibidos devido à quebra de legibilidade, causada pelo tamanho das imagens.

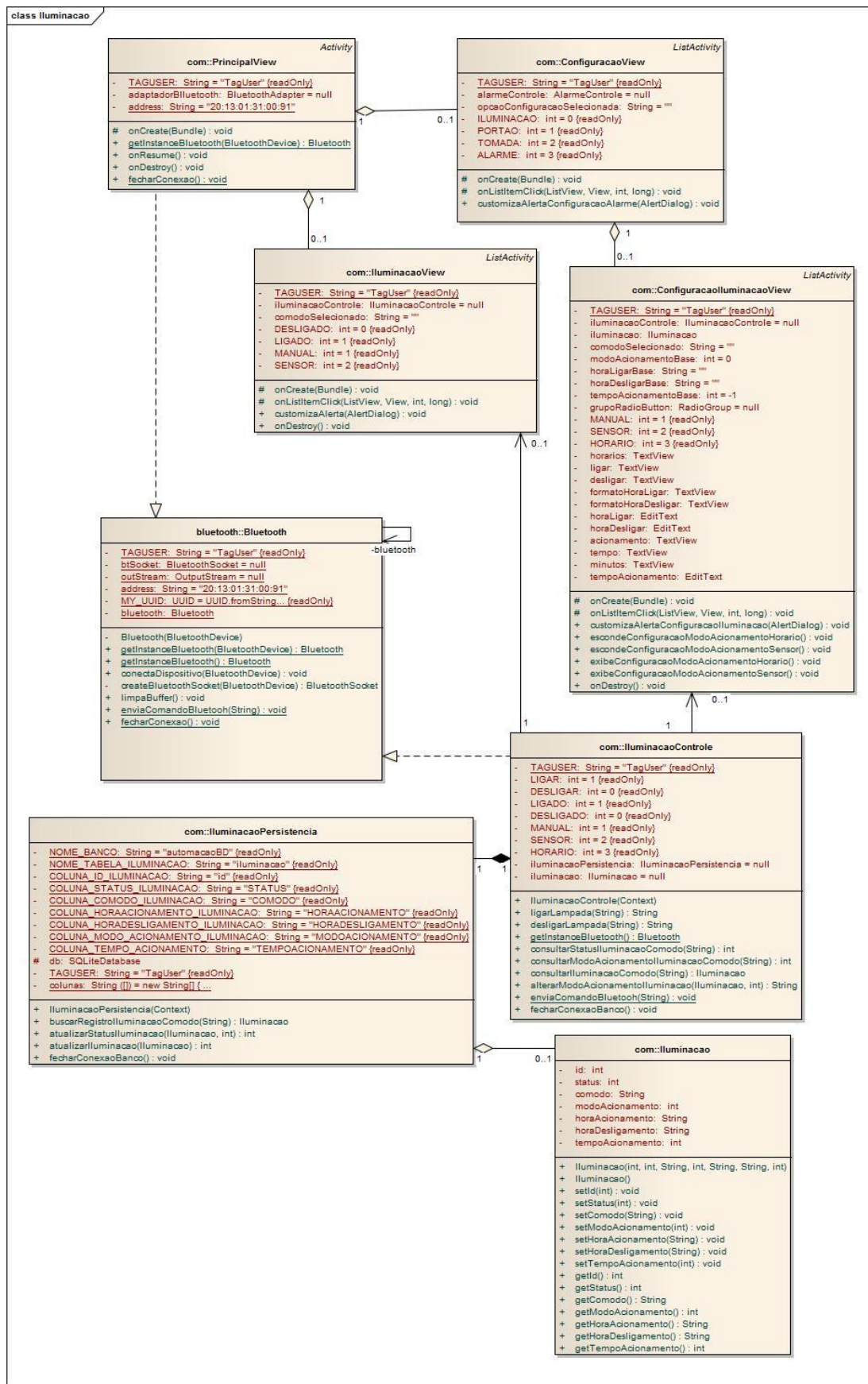
Figura 55 – Diagrama de classe do Controle de Televisão.



Fonte: Elaboração do autor, 2012.

O controle da iluminação foi modelado, conforme a Figura 56.

Figura 56 – Diagrama de classe do Controle de Iluminação.

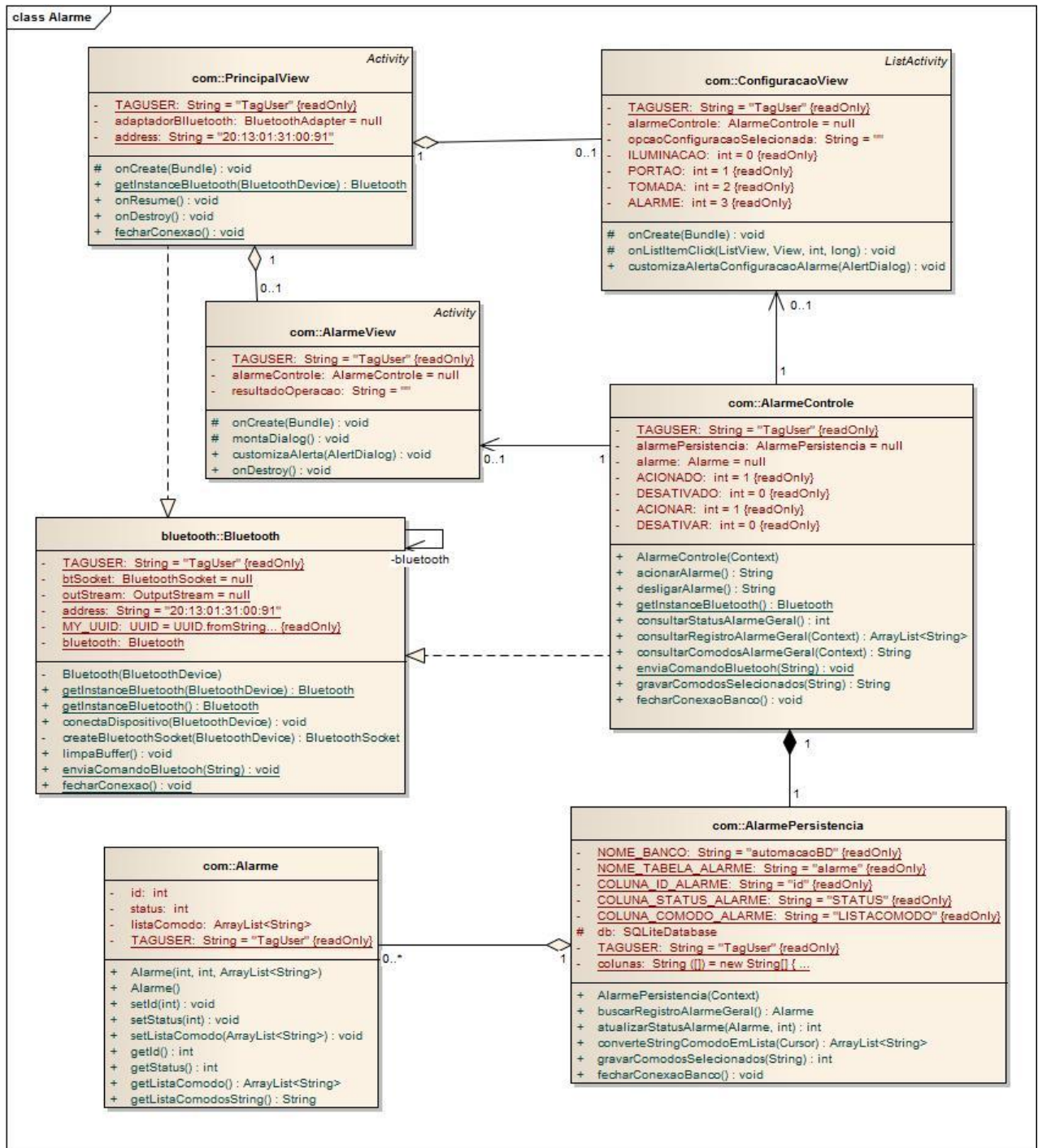


Fonte: Elaboração do autor, 2012.

A Figura 57 representa a modelagem para o diagrama de classe responsável pelo acionamento do alarme, proposto pelo sistema.

Figura 57 – Diagrama de classe do Controle e Alarme.

Fonte: Elaboração do autor, 2012.



Fonte: Elaboração do autor, 2012.

## 4.2 HARDWARE

Nesta seção, serão apresentados alguns modelos propostos para o desenvolvimento de hardware do sistema. Esses modelos mostram de modo virtual como seria a montagem física em *protoboard*<sup>1</sup>.

### 4.2.1 Acionamento da Lâmpada

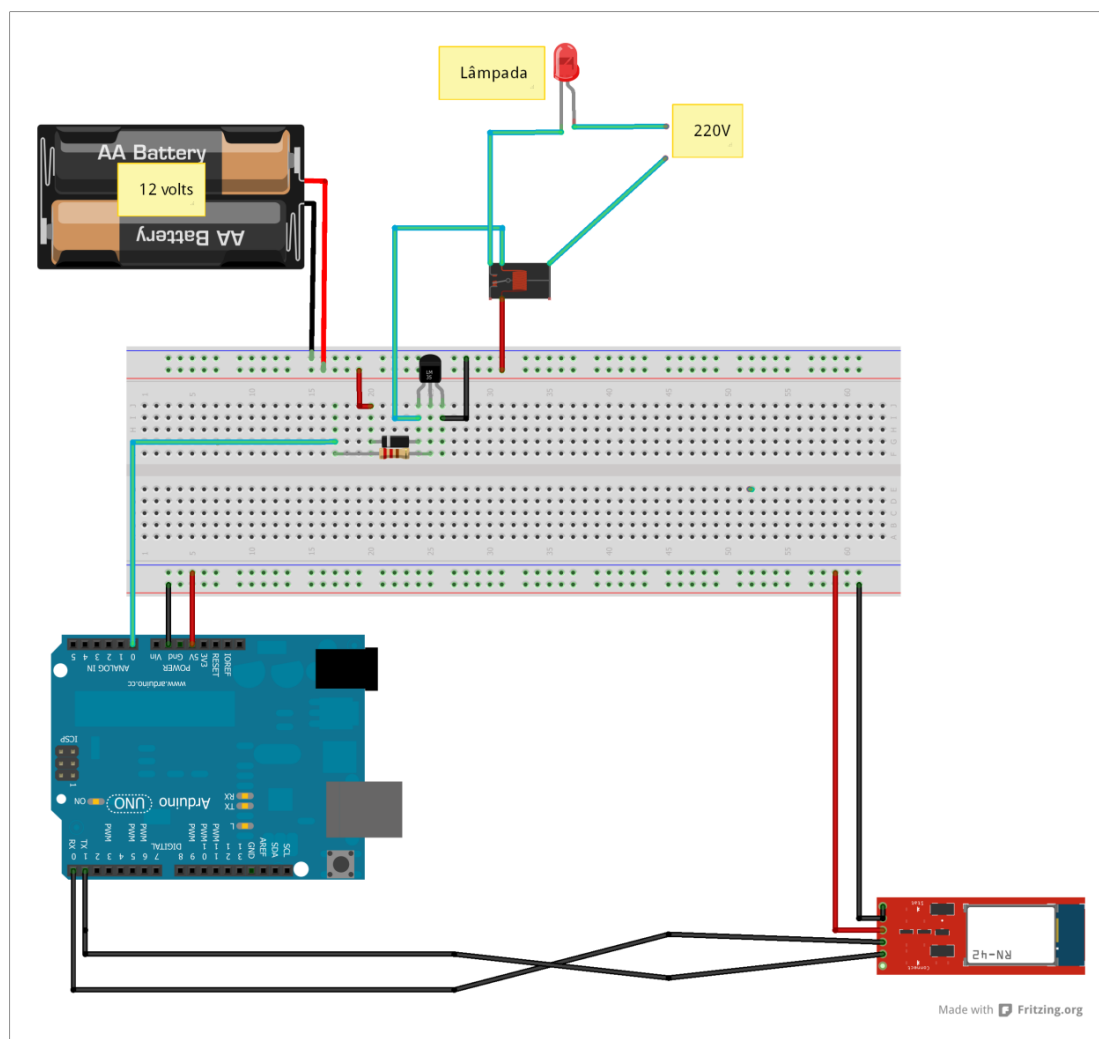
A Figura 58 representa o modelo para controle de uma lâmpada pelos modos de acionamento manual e por horário, composto pela placa Arduino, o módulo Bluetooth, uma bateria para alimentação do relé de 12 volts, a lâmpada a ser acionada e o módulo relé, composto por um transistor, um resistor, um diodo e um relé.

O relé possui um contato interno e uma bobina. Quando há corrente passando pela bobina, um campo magnético é induzido, atraindo um pino interno e fechando o contato. Para seu funcionamento, basta aplicar 5V do Arduíno entre 2 pinos, para acionar o contato dos outros dois pinos. Na prática, a fonte do Arduíno, pode não aguentar fornecer diretamente os 25mA que o relé necessita. Nesta situação é necessário um pequeno circuito eletrônico, composto, principalmente, de um transistor, para que essa corrente seja gerada.

---

<sup>1</sup> Protoboard: também chamado de matriz de contato. É usada para fazer montagens provisórias, teste de projetos eletrônicos. Na superfície da matriz há uma base de plástico em que existem centenas de orifícios onde são encaixados os componentes, já em sua parte inferior, são instalados contatos metálicos. (LABORATÓRIO AVANÇADO DE SISTEMAS DE AUTOMAÇÃO, 2013).

Figura 58 – Modelo para acionamento Manual ou por Horário da Lâmpada.

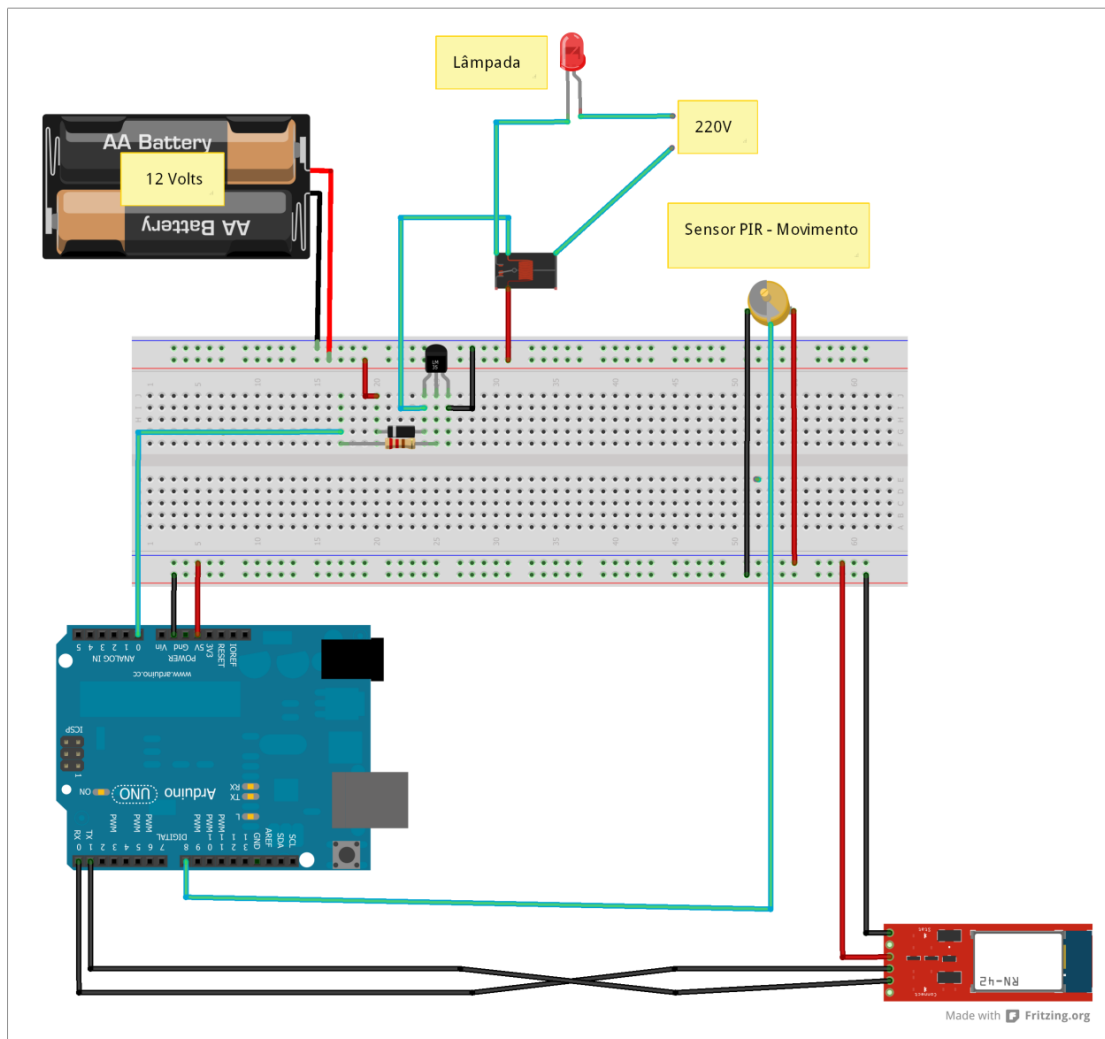


Fonte: Elaboração do autor, 2012.

O modelo para controle de uma lâmpada pelo modo de acionamento sensor é ilustrado na Figura 59, sendo adicionado para esta funcionalidade um sensor de presença em relação ao modelo da Figura 58.



Figura 59 – Modelo para acionamento da Lâmpada por Sensor.

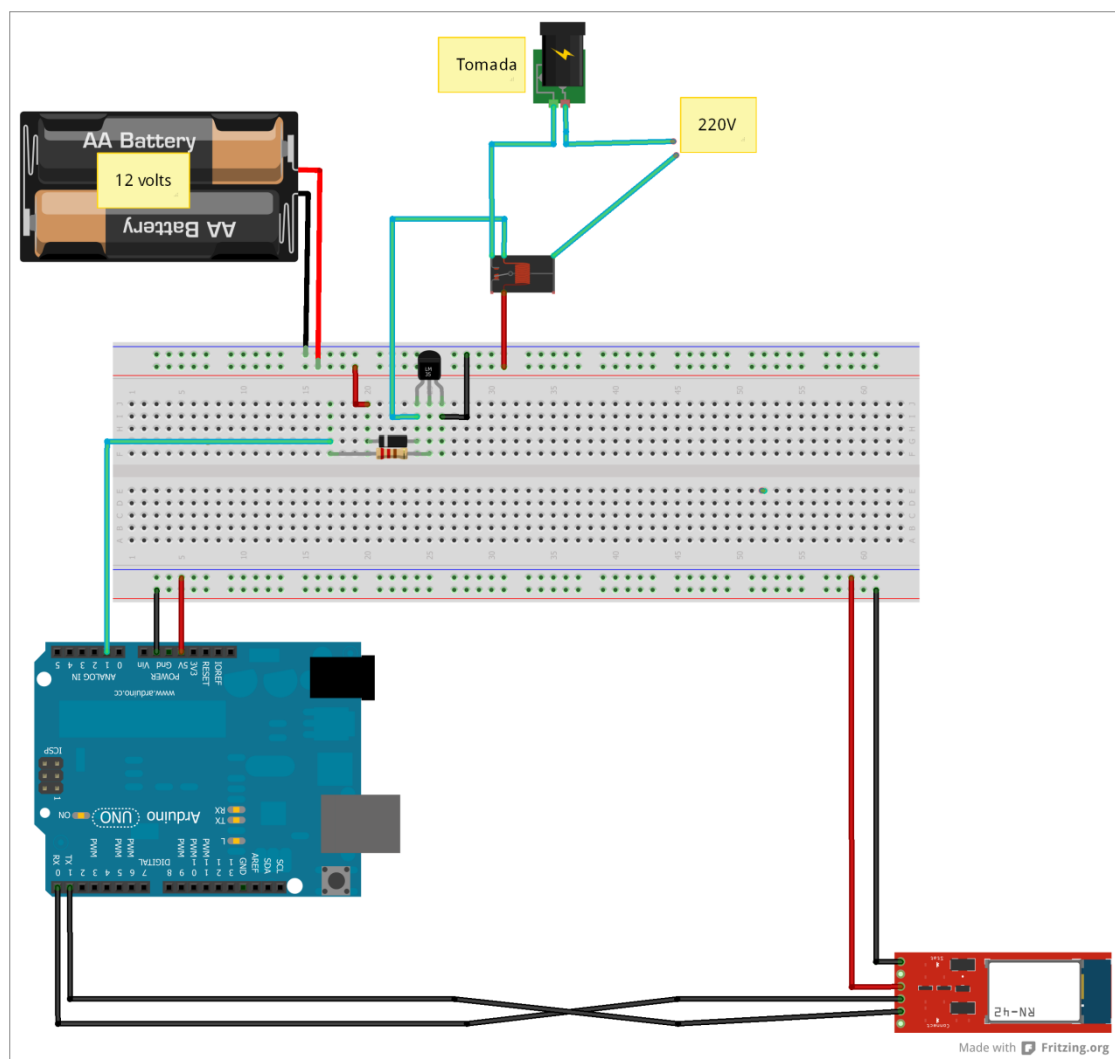


Fonte: Elaboração do autor, 2012.

#### 4.2.2 Acionamento da Tomada

O Modelo para controle e uma tomada é similar ao modelo de acionamento da lâmpada, exceto que ao invés de uma lâmpada, uma tomada é que será acionada. Este modelo é apresentado na Figura 60.

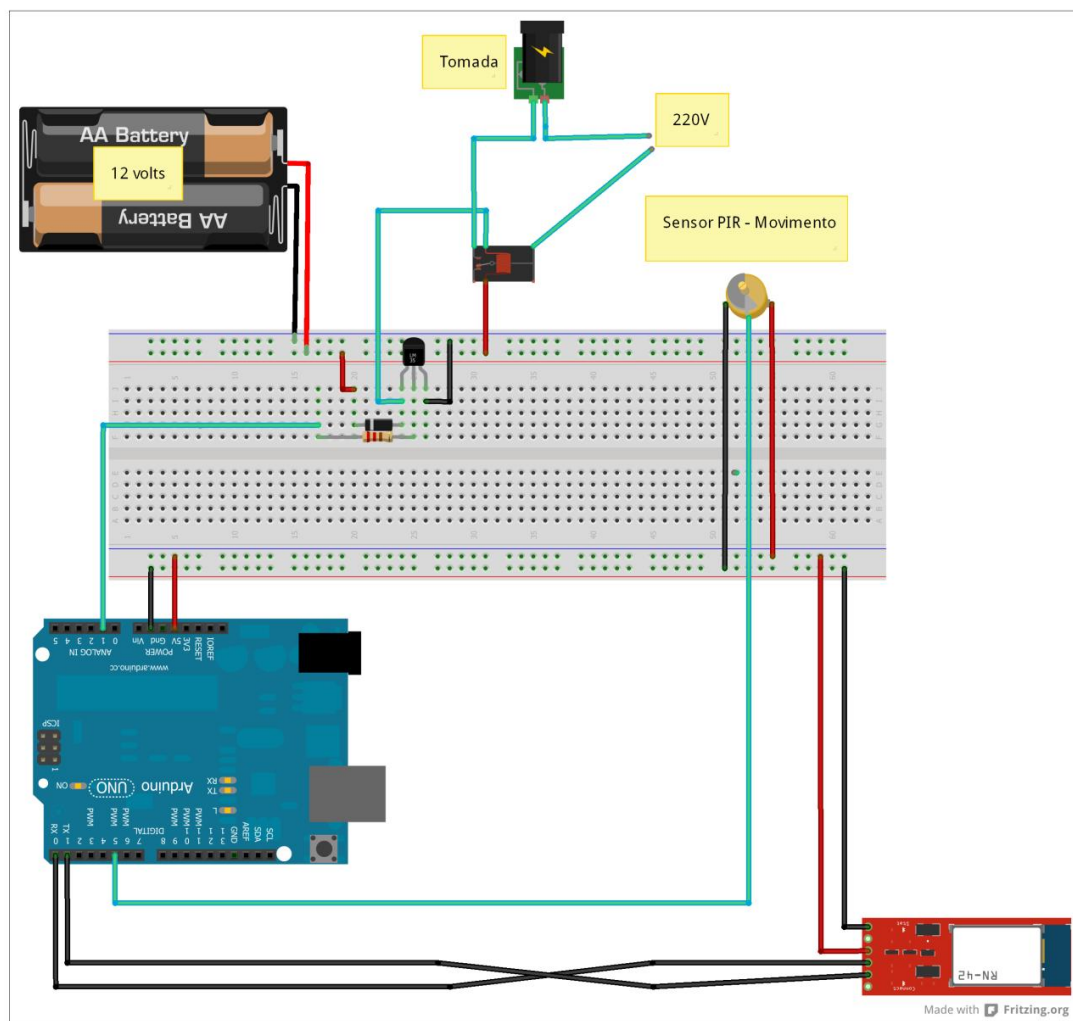
Figura 60 – Modelo para acionamento Manual ou por Horário da Tomada.



Fonte: Elaboração do autor, 2012.

O modelo para o controle de uma tomada pelo modo de acionamento por sensor é ilustrado na Figura 61, sendo adicionado para este um sensor de presença em relação ao modelo da Figura 60.

Figura 61 – Modelo para acionamento da Tomada por Sensor.



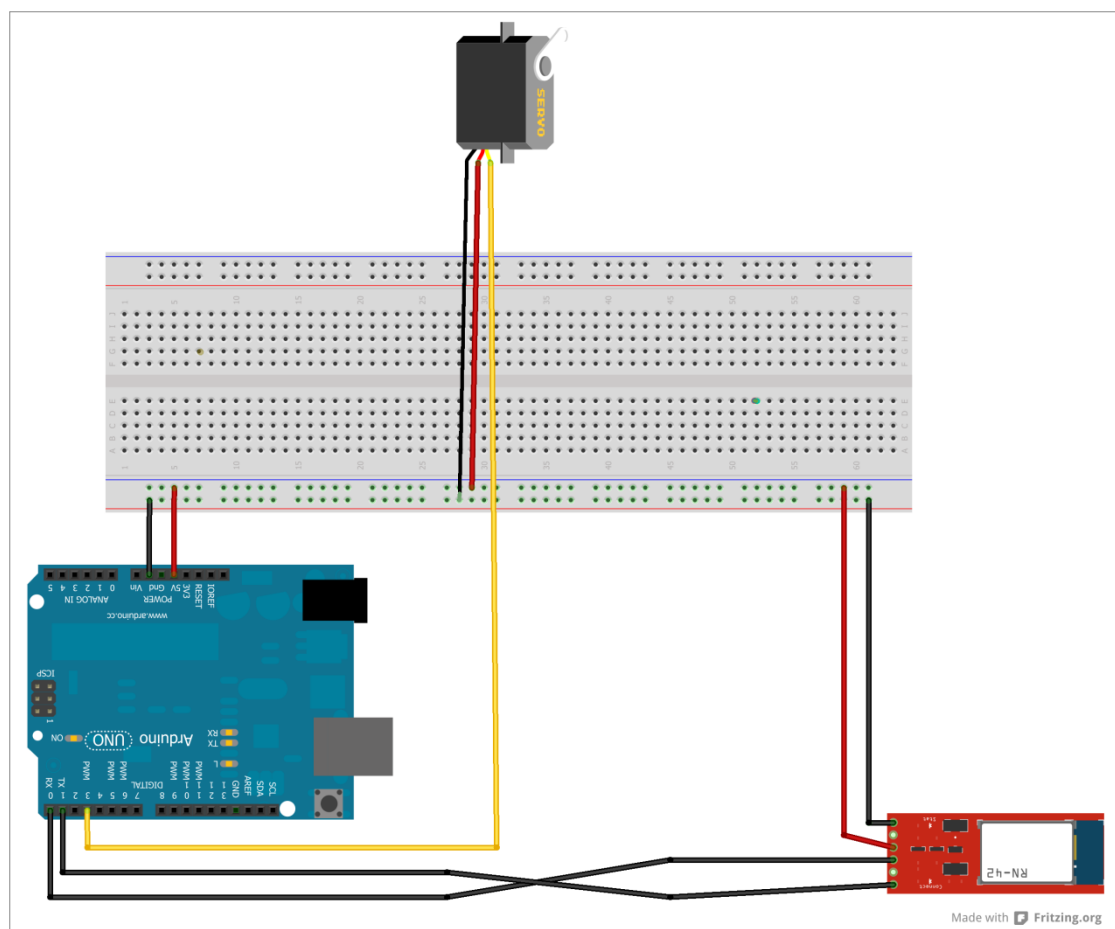
Fonte: Elaboração do autor, 2012.

#### 4.2.3 Acionamento do Portão

A Figura 62 representa o modelo para o controle de um portão, composto pela placa Arduino, o módulo Bluetooth e o servo motor, responsável pelo acionamento do portão.



Figura 62 – Modelo para acionamento do Portão.



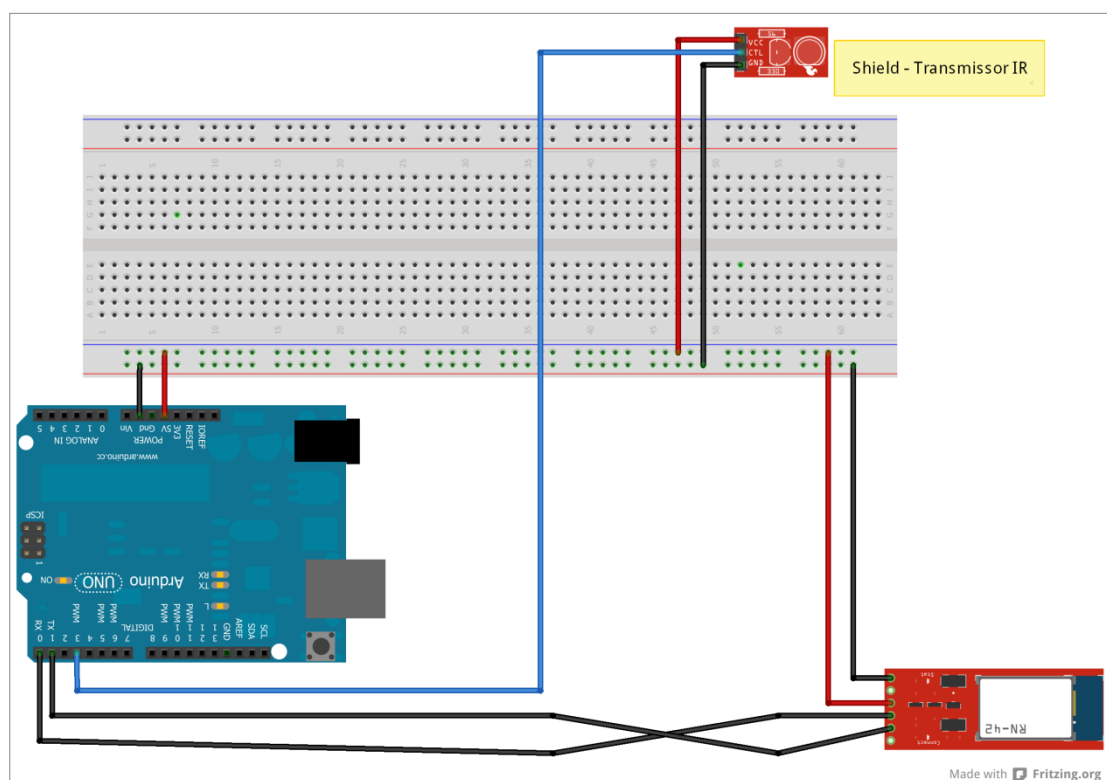
Fonte: Elaboração do autor, 2012.

#### 4.2.4 Acionamento do Alarme

A Figura 63 representa o modelo para o acionamento do alarme. Fazem parte desse modelo a placa Arduino, o módulo Bluetooth e o módulo alarme, composto pelo sensor de presença e o *piezo speaker* (reprodutor de som).



Figura 64 – Modelo para acionamento da Televisão.



Fonte: Elaboração do autor, 2012.

## 5 DESENVOLVIMENTO

Nesta fase do trabalho, após o processo de modelagem, serão apresentadas as etapas de desenvolvimento da proposta de solução para um sistema de automação residencial. Este capítulo começará abordando o histórico de desenvolvimento, as tecnologias utilizadas, as funcionalidades do sistema e suas telas, partindo para a maquete na qual o sistema foi implantado, o protocolo de comunicação e finalizando com a validação.

### 5.1 HISTÓRICO DO DESENVOLVIMENTO

O início do desenvolvimento aconteceu na fase de modelagem, na busca de conhecimentos específicos sobre os processos e tecnologias a serem empregadas no projeto.

#### 5.1.1 Modelagem

Foi nesta etapa, que surgiu a necessidade de focar os estudos nas tecnologias a serem utilizadas, a fim de conhecer os principais recursos e limitações que poderiam influenciar no desenvolvimento e, principalmente, definir o escopo do sistema. Para o software, foram criados os diagramas de classes, casos de usos e os requisitos, já para a parte de hardware foram criados os modelos de *protoboard* com a estrutura dos componentes e suas conexões, conforme pôde ser visto no capítulo 4.

### 5.1.2 Aquisição de componentes

O baixo orçamento disponível foi um fator impactante para o desenvolvimento do projeto, percebido logo após a modelagem, quando se deu início ao processo de compra dos componentes a serem utilizados na montagem de hardware.

A ideia de adquirir alguns componentes sobressalentes e alguns outros que não estariam no escopo do projeto contribuiu para o aumento no valor gasto. Componentes sobressalentes no caso de algum outro se danificar e os demais no caso da possibilidade de maximizar as funcionalidades do sistema caso houvesse tempo. Por indicações de amigos e pesquisas na internet, os componentes foram comprados pelo site <http://dx.com>, loja localizada na China, que ofereciam os componentes pela metade do preço em relação às lojas brasileiras, porém com tempo de entrega bem superior. A compra foi realizada mesmo sabendo do risco que se corria em não receber os componentes em tempo hábil.

Após o término do desenvolvimento do software para Android e do firmware para Arduino, surgiu a necessidade de adquirir os componentes em lojas brasileiras, visto que os que foram comprados na China não haviam chegados, alterando completamente o orçamento utilizado. Um dia depois de realizado o orçamento em uma loja no Brasil, o primeiro pacote dos 4 chegou, contendo um Arduino Mega, o módulo Bluetooth, o transmissor Ir, um receptor Ir e um sensor de distância, conforme pode ser visto na Figura 65, descartando a ideia de compra no Brasil. Em seguida, chegaram os demais pacotes.



Os principais requisitos presentes no *tablet* necessários ao desenvolvimento do projeto são: Sistema Operacional Android 4.0 e o Bluetooth, conforme apresentados na Figura 67.

Figura 67 – Características do Tablet Genesis GT – 7240.

CARACTERÍSTICAS	
<ul style="list-style-type: none"> <li>• <b>OS</b> Android 4.0 Ice Cream Sandwich</li> <li>• <b>CPU + GPU</b> Cortex A9 1.2Ghz+ Mali400</li> <li>• <b>RAM</b> 1GB</li> <li>• <b>ROM</b> 8GB Interno Expansível até 32GB</li> <li>• <b>TELA</b> 7" Polegadas 5 pontos Multitouch Capacitiva</li> <li>• <b>RESOLUÇÃO</b> 1024x600 HD</li> <li>• <b>CAMERA</b> Frontal 0.3MP / Traseira 2MP</li> </ul>	<ul style="list-style-type: none"> <li>• <b>HDMI</b> 480P, 720P, 1080P</li> <li>• <b>WIFI</b> 802.11 b/g/n</li> <li>• <b>Bluetooth</b></li> <li>• <b>3G</b> 3G Dongle Opcional</li> <li>• <b>TV DIGITAL ISDB-T</b> Dongle DTV Opcional</li> <li>• <b>BATERIA</b> 3.7V 4000mAh</li> <li>• <b>ENTRADA</b> USB / AUDIO 3.5mm MICRO SDHC SLOT / 3G HOST</li> <li>• <b>GOOGLE Play Store</b> Mais de 700.000 aplicativos</li> </ul>

Fonte: TodaOferta, 2013.

### 5.1.3 Desenvolvimento para Android

O conhecimento sobre a plataforma Android era básico, todos adquiridos na matéria sobre dispositivos móveis, apresentada na 8ª fase da universidade no curso de Ciência da Computação. Tais conhecimentos não foram suficientes para que o desenvolvimento da aplicação fosse iniciado sem qualquer tipo de estudo, porém ajudaram a determinar por quais caminhos seguir. Os estudos sobre a plataforma ocorreram de acordo com as dificuldades encontradas no caminhar do desenvolvimento.

Os principais pontos de travamento foram no início do projeto com a configuração do ambiente de desenvolvimento, no entendimento do ciclo de vida do Android, na criação da base de dados e, principalmente, em colocar em funcionamento a troca de informações via Bluetooth com o Arduino. A principal vantagem em relação ao

desenvolvimento do software para Android foi o conhecimento adquirido na linguagem de programação Java durante toda a fase acadêmica.

#### **5.1.4 Protocolo de Comunicação entre Android e Arduino**

Para que fosse possível a comunicação entre o Android e Arduino, foi necessária a criação de um protocolo particular, conforme será detalhado na seção 5.3 deste trabalho. Essa etapa foi finalizada sem grandes dificuldades.

#### **5.1.5 Desenvolvimento para Arduino**

O fato do desenvolvimento para Arduino ser essencialmente baseado na linguagem de programação C/C++ contribuiu para o bom desempenho no desenvolvimento do sistema. Isso se deu devido ao bom conhecimento do autor sobre a linguagem utilizada. Outro fator muito importante para a agilidade do desenvolvimento foi a farta documentação encontrada. A preocupação nessa fase ficou atrelada às limitações de memória flash e processamento, portanto, o desenvolvimento foi realizado de forma a utilizar o mínimo de recursos disponíveis.

Após a metade do desenvolvimento, houve-se a necessidade de migração do modelo de Arduino utilizado. Nesta fase o tamanho do código fonte encontrava-se com 31K, quase 32K, limite suportado pelo modelo Uno. O novo Arduino utilizado passou a ser o modelo Mega, com capacidade de 256K de memória flash.

O tempo de resposta, em relação aos comandos de controle dos dispositivos, executados em maquete, mostrou-se satisfatório, não sendo observado nenhum fator negativo em relação ao desempenho.

A dificuldade maior foi programar e colocar em prática a comunicação serial via Bluetooth com o Android.



#### 5.1.5.1 Características do Arduino Mega

O Arduino Mega, utilizado para o desenvolvimento final do protótipo, é uma placa de microcontrolador, baseada no ATmega2560. Ele possui 54 portas digitais e 16 analógicas, nas quais podem ser utilizadas como entradas ou saídas, conforme configuração, realizada no código fonte.

A quantidade de componente que pode ser gerenciado está relacionada com o modelo utilizado, pois alguns componentes necessitam somente de uma porta, enquanto, outros, de duas ou mais portas.

O modelo contém quatro portas seriais, um oscilador de cristal de 16 MHz, uma conexão USB (utilizada para transferência do sistema e alimentação, através do computador), uma entrada de alimentação, uma conexão ICSP (*In System Chip Programming*) e um botão reset. O Mega é compatível com quase todos os demais componentes utilizados pelo Arduino Uno e Duemilanove.

Em relação à memória, o ATmega2560 tem 256KB de memória *flash* para armazenamento de código, dos quais 8KB são usados pelo *bootloader*. Possui 8KB de SRAM (*Static Random Access Memory*, que significa memória estática de acesso aleatório em Português) e 4KB de EEPROM (*Electrically-Erasable Programmable Read-Only Memory*). A memória EEPROM pode ser lida e escrita utilizando a biblioteca EEPROM.

#### 5.1.6 Hardware

Esta foi uma das maiores preocupações durante o processo de desenvolvimento, etapa esta relacionada com a montagem dos componentes de hardware. Pois, apesar de ter todos os modelos desenhados e alguns componentes sobressalentes, o mau uso, como, por exemplo, uma conexão errada ou qualquer outro problema, poderia danificar alguns componentes, atrasando a solução final do trabalho. Por fim, o protótipo foi montado e nenhuma complicação grave ocorreu. Outro fator que contribuiu para o sucesso da montagem está relacionado com o conhecimento básico do autor em relação

aos conceitos de eletrônica, já que o mesmo é formado no curso Técnico em Telecomunicações.

A Figura 68 apresenta uma visão macro do ambiente de desenvolvimento de hardware.

Figura 68 – Visão macro do ambiente de desenvolvimento de Hardware.



Fonte: Elaboração do autor, 2013.

### 5.1.7 Maquete

A construção da maquete exigiu conhecimentos sobre a manipulação de algumas ferramentas e de mecânica para a construção do portão. Primeiro a maquete foi modelada em papelão e só depois finalizada com material de plástico, madeira e acrílico, todos de fonte reciclada. Em seguida, foi feita a adaptação dos componentes eletrônicos na maquete. A criação da maquete em papelão foi pensada com a finalidade de economizar madeira e tempo, já que a manipulação do papelão é mais fácil.

A Figura 69 apresenta a imagem da maquete semi acabada.

Figura 69 – Maquete Residencial Semi Acabada.



Fonte: Elaboração do autor, 2013.

## 5.2 TECNOLOGIAS E FERRAMENTAS UTILIZADAS

As tecnologias e ferramentas utilizadas para que o projeto pudesse ser desenvolvido são apresentadas na Figura 70.

Figura 70 – Tecnologias e Ferramentas utilizadas no projeto.



Fonte: Elaboração do autor, 2013.

➤ Android: é um sistema operacional concebido em Linux de código fonte aberto, utilizado também como plataforma de desenvolvimento. Por se tratar de um sistema operacional gratuito e ser facilmente adaptável tanto para *tablets* quanto para *smartphones*, foi utilizado para a criação da Interface.

➤ Arduino: como ilustrado na seção 2.8.1, trata-se de uma plataforma de prototipagem eletrônica de hardware livre, com suporte a entrada e saída, processamento e armazenamento de dados, com base em um microcontrolador. Seu objetivo é proporcionar as pessoas que deem início a seus projetos com um mínimo de recursos possíveis.

➤ SQLite: o SQLite além de ser leve, é um poderoso banco de dados e possui integração com o Android, permitindo dessa forma que fosse utilizado no projeto. No Android o banco pode ser criado de várias formas, utilizando a API do Android para SQLite ou através das ferramentas de gerenciamento disponíveis. (LECHETA, 2011).

➤ Bluetooth: é uma tecnologia de comunicação sem fio utilizada em pequenas distâncias, com alcance médio de no máximo 8 metros. (GOOGLE, 2013). Permite que computadores, *smartphones*, *tablets* e afins troquem dados entre si. Por esse motivo, decidiu-se aplicar essa tecnologia no projeto, garantindo, assim, o controle dos dispositivos em qualquer cômodo da residência, desde que esteja dentro do raio de cobertura do sinal.

Outra boa vantagem em utilizá-la está no fato do baixo consumo de energia.

➤ Dev ++: completo e poderoso ambiente de desenvolvimento para linguagens de programação C e C++. Possui suporte ao CVS (*Concurrent Version System*) que significa controlador de versão, localização e substituição de strings, bibliotecas estáticas e DDLs (*Data Definition Language*), suporte a impressão, funções nativas para listas, suporte ao *Makefile*, ferramenta para debug, console, assim como outras funcionalidades. (BLOODSHEDSOFTWARE, 2013). Pela facilidade de instalação e desenvolvimento, esta ferramenta foi utilizada como suporte ao desenvolvimento do código para o Arduino.

- Java: É considerada como uma linguagem de programação de alto nível, multiplataforma e orientada a objetos (KEN et al., 2007). O bom conhecimento nessa linguagem facilitou o desenvolvimento do sistema. Seu uso se deve ao fato de a plataforma Android executar softwares escritos nessa linguagem.
- Google: empresa multinacional de serviços online e software dos Estados Unidos. O Google desenvolve e hospeda vários serviços baseados na web, uma delas consiste em uma ferramenta de pesquisa, que, por sinal, foi muito utilizada em todas as fases deste projeto. Também lidera o desenvolvimento do sistema operacional Android, juntamente com outras empresas.
- Eclipse: o Eclipse consiste em uma ferramenta para desenvolvimento e gerenciamento de softwares. Sua aplicação se deu ao fato da fácil integração com o desenvolvimento para softwares destinados a plataforma Android, juntamente com a instalação do *plugin* ADT (*Android Development Tools*), conjunto de ferramentas na qual estão inclusos um compilador e um emulador, utilizado para validações e testes dos aplicativos. (LECHETA, 2011).
- Enterprise Architect: segundo Makesys (2013), trata-se de um ambiente de modelagem que contempla todo o ciclo de vida de um produto, com alta performance visual e ferramentas para a modelagem de negócio, arquitetura do sistema, gerenciamento de requisitos, projeto de software, geração de código fonte, gerenciamento de testes e muito mais. O uso da ferramenta no processo de desenvolvimento do projeto conteve a criação dos casos de uso, requisitos e diagramas de classes, descritos na seção 4.1.
- Fritzing: ferramenta gráfica de apoio a pesquisadores e desenvolvedores para trabalhar com a parte de prototipação eletrônica, voltada para montagens virtuais de projetos com Arduino. É muito fácil de instalar e usar, sua interface é em português e possui vários exemplos para o auxílio no aprendizado. Os projetos criados podem ser visualizados no modo *protoboard*, esquemático e placa de circuito Impresso. (FRITZING, 2011). Com o apoio dessa ferramenta foram criados os modelos do projeto com base na visão *protoboard*, conforme ilustrados na seção 4.2.

- **C/C++:** Atualmente, empregam-se cada vez mais sistemas computacionais na automação, seja ela industrial, predial ou residencial. Esses sistemas variam desde um simples circuito até os circuitos mais complexos, envolvendo uso de microcomputadores. Nesse contexto, a programação é de suma importância, pois a inteligência dos sistemas automatizados é implementada através de programas computacionais, efetuando o controle dos diversos componentes de hardware para executar tarefas pré-definidas. Nesta área, a linguagem de programação “C” é a mais utilizada, por ser considerada uma linguagem de alto nível muito poderosa, portátil, flexível, de execução rápida e utilizar o mínimo de memória possível. Posteriormente, a sua sucessora a linguagem “C++”, é resultado da união da programação “Orientada a Objetos” à linguagem “C” (UFSC, 2013). Por todas essas características citadas, o Arduino faz delas sua linguagem de desenvolvimento.
  
- **Mozilla Firefox:** trata-se de um navegador gratuito, criado por uma organização global sem fins lucrativos. Trata-se de um navegador personalizável e composto de muitas ferramentas. Possui navegação em abas, corretor ortográfico, sincronizador de dados, restauração de sessão, sistema de segurança, pesquisa inteligente, proteção de sites falsos, memorizador de formulários, bloqueador de janelas *popup* e sua instalação é super rápida. O mais interessante é que pode ser adicionado *plugins*, ou seja, inseridas extensões que criam novas funcionalidades. (MOZILLA, 2010).
  
- **SQLiteManage:** extensão do Mozilla Firefox que permite o gerenciamento de banco de dados SQLite no computador local diretamente pelo navegador. É uma boa alternativa para usuários que necessitam de velocidade e buscam métodos de gerenciamento práticos e inteligentes. (TECHTUDO, 2013). Por essas características e por ser de fácil manuseio e manutenção, foi utilizado para o desenvolvimento do projeto.

### 5.3 PROTOCOLO DE COMUNICAÇÃO

Como definido na seção 2.6, protocolo de comunicação é um conjunto de regras padronizadas, que permite a comunicação entre dispositivos que partilham da mesma ligação. Um protocolo tem como finalidade facilitar a troca de informações.

O protocolo criado para uso no projeto foi desenvolvido, pensando em alguns aspectos, como praticidade e facilidade no seu entendimento e manuseio. Para isso, sua estrutura foi dividida em duas partes, a primeira responsável por tratar as ações efetuadas pelo usuário e outra responsável por comunicar as configurações solicitadas.

Para fins de esclarecimento, o termo “ações” foi tratado como sendo uma solicitação do usuário na mudança de status de um dispositivo, como por, exemplo: o acionamento de uma lâmpada. O termo “configurações” foi tratado como sendo uma solicitação de alteração no modo de funcionamento de um dispositivo, como, por exemplo: alteração do modo de acionamento de uma lâmpada. A descrição detalhada de cada termo será apresentada nas seções 5.3.1 e 5.3.2.

Cada ação ou configuração efetuada pelo usuário interfere diretamente no status do dispositivo relacionado a ele na residência. Após a alteração de uma funcionalidade qualquer na interface pelo usuário, uma sequência de caracteres é enviado, via Bluetooth, do Android para o Arduino. Um método desenvolvido no firmware recebe e trata este protocolo, com base no primeiro caractere, ele consegue identificar se o comando é de configuração ou ação, encaminhando a alteração para o método seguinte responsável pelo tratamento.

#### 5.3.1 Tratamento do protocolo para Ações

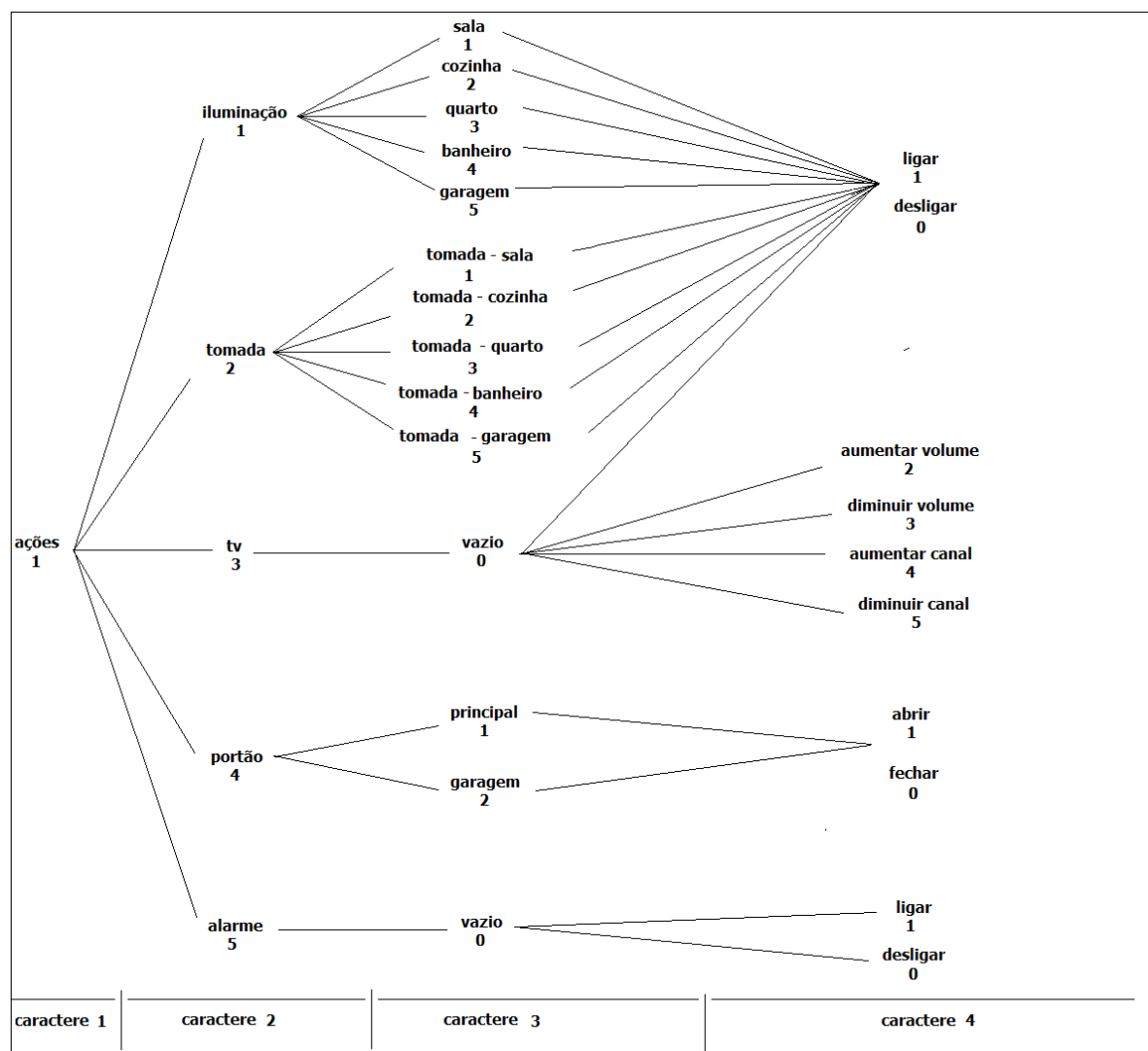
No método desenvolvido, após o tratamento do primeiro caractere, segue o tratamento dos demais caracteres.

Com base no segundo caractere, o método distingue para qual dispositivo o comando é destinado. O terceiro caractere está relacionado com o ambiente, cômodo ou descrição do dispositivo. Já o quarto caractere pertence à funcionalidade requisitada.

A Figura 71 exibe a parte do protocolo referente às ações, juntamente com o caractere correspondente a cada dispositivo (Iluminação, Tomada, Televisão, Portão e Alarme), cômodo (Sala, Cozinha, Quarto, Banheiro e Garagem), ambiente (Principal e Garagem), descrição (tomada – sala, tomada - cozinha, tomada - quarto, tomada banheiro e tomada - garagem) e funcionalidade (Ligar, Desligar, Aumentar Volume, Diminuir Volume, Aumentar Canal, Diminuir Canal, Abrir e Fechar).



Figura 71 – Protocolo de Comunicação – Ações.



Fonte: Elaboração do autor, 2013.

### 5.3.2 Tratamento do protocolo para Configurações

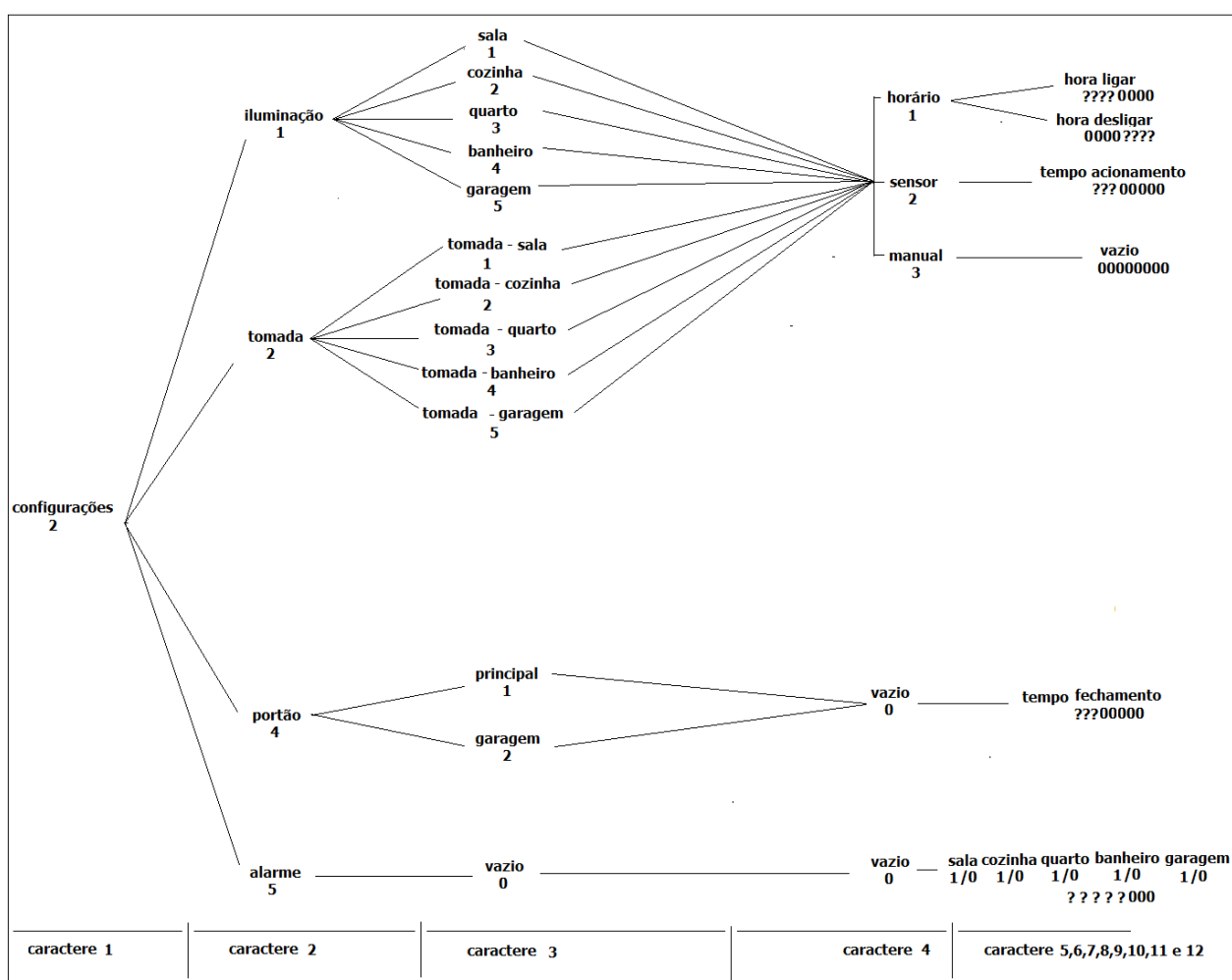
Após o tratamento do primeiro caractere, os demais entram na sequência de validação.

Com base no segundo caractere o método distingue para qual dispositivo o comando é destinado. O terceiro caractere pertence ao ambiente, cômodo ou descrição do dispositivo. Do quarto ao décimo segundo caractere fazem parte da configuração.

- Iluminação: o quarto caractere corresponde ao modo de acionamento da iluminação. O quinto, sexto, sétimo e oitavo indicam a hora de ligar para o modo de acionamento por Horário. Ainda, no mesmo modo, o nono caractere, o décimo caractere, o décimo primeiro caractere e o décimo segundo pertencem à hora de desligar. Já para o modo Sensor, o quinto, sexto e o sétimo caractere representam o tempo de acionamento.
- Portão: o quinto, sexto e sétimo indicam o tempo de fechamento.
- Alarme: Do quinto ao décimo segundo, cada bit corresponde a um cômodo. O caractere com o valor 1 indica que o cômodo correspondente está incluso na configuração do alarme e o 0 indica que o cômodo está desassociado ao alarme.

A Figura 72 exibe a parte do protocolo referente às Configurações, juntamente com o caractere correspondente a cada dispositivo (Iluminação, Tomada, Televisão, Portão e Alarme), cômodo (Sala, Cozinha, Quarto, Banheiro e Garagem), ambiente (Principal e Garagem), descrição (tomada – sala, tomada - cozinha, tomada - quarto, tomada banheiro e tomada - garagem) e configuração (hora ligar, hora desligar, tempo acionamento, tempo fechamento e cômodos para Alarme).

Figura 72 – Protocolo de Comunicação – Configurações.



Fonte: Elaboração do autor, 2013.

## 5.4 DESENVOLVIMENTO ANDROID

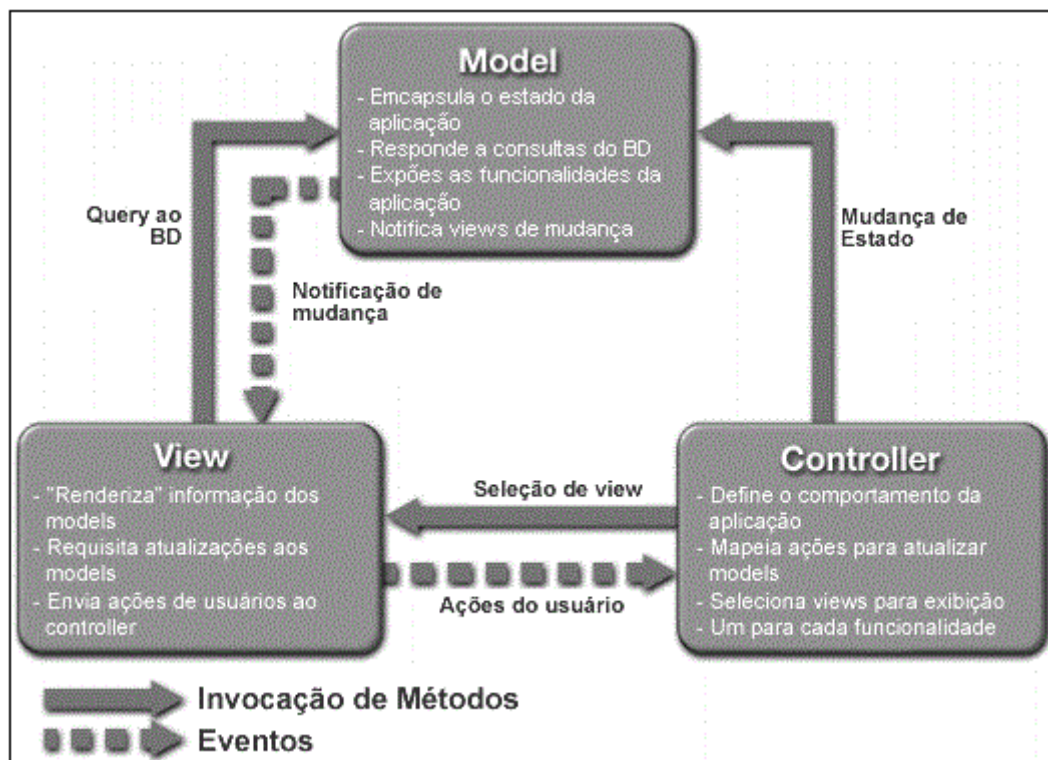
A configuração do ambiente de desenvolvimento Android com Eclipse foi à primeira tarefa realizada. Envolveu a instalação do Eclipse, do JDK (*Java Development Kit*), do Android SDK e do *plugin* ADT (*Android Development Tools*). Em seguida foi instalada a plataforma 4.0 do Android, definida para o desenvolvimento do projeto por ser a versão atual a rodar em *smartphones* e *tablets*.

Por fim foi configurado um dispositivo virtual, que nada mais é que um emulador, utilizado para testes de desenvolvimento, conforme descrito na seção 2.9.2.

Depois de configurado todo o ambiente, partiu-se para o desenvolvimento do código fonte com base na modelagem descrita no capítulo 4. O projeto foi construído na arquitetura MVC (*Model View Controller*), que consiste em um modelo de desenvolvimento, considerado como um *Design Pattern* (padrão de projeto), muito utilizado na Engenharia de Software. Este modelo isola entre si as três camadas, interface, lógica e persistência de dados. (GAMMA et al., 2004).

Na Figura 73, é apresentada a arquitetura MVC.

Figura 73 – Arquitetura MVC.



Fonte: Zemel, 2009.

Outro fator importante, além da estrutura MVC, utilizado no desenvolvimento do projeto e que influenciou em sua agilidade, mantendo também, ao final, um software mais estável, foi a realização de testes durante todo o ciclo de vida do projeto.

## 5.5 DESENVOLVIMENTO ARDUINO

Para o desenvolvimento do firmware, foi utilizado o Arduino IDE, descrito na seção 2.8.1.2. As funções desenvolvidas são invocadas de acordo com o protocolo recebido, via Bluetooth, conforme descrito na seção 5.3. Os principais métodos utilizados e que fazem parte da própria estrutura são:

- *setup()*: função inserida logo no início para inicialização das configurações. Nesta função foram definidas as portas de entrada e saída correspondente a cada dispositivo, no caso deste projeto, as portas para os motores responsáveis pelo acionamento dos portões, os relés e sensores relacionados ao acionamento da iluminação e das tomadas, o transmissor Infravermelho para o controle da televisão e do *clock* responsável pela definição de data e hora. Foram definidas, também, as configurações da porta serial para comunicação com Android.
- *loop()*: método repetido desde o acionamento do Arduino até o seu desligamento. Esse método é responsável por gerenciar todas as solicitações recebidas pelo protocolo e tratar os eventos enviados pelos sensores, destinando o fluxo de dados para os métodos correspondentes às tomadas de ações.

O desenvolvimento do software foi dividido em três pontos:

- **Lógica**: nessa etapa foi desenvolvido todo controle e fluxo do software, definições de métodos e variáveis.
- **Comunicação**: fase em que foi desenvolvida e testada a comunicação serial entre o Android e Arduino.
- **Hardware**: foi a etapa mais complexa do desenvolvimento. Consistiu na integração do software com a parte de hardware. Incluiu a montagem de hardware, testes, revisões e alterações de código.

## 5.6 DESENVOLVIMENTO HARDWARE

O desenvolvimento do protótipo de hardware proposto consistiu na ligação física entre todos os componentes definidos no projeto de modo que se obteve o pleno funcionamento das funcionalidades descritas nos casos de uso, levantados no início do projeto.

Cada módulo adquirido corresponde a uma funcionalidade do sistema. Os módulos utilizados no projeto e suas especificações são descritas a seguir:

- Módulo Arduino: consiste no microcontrolador Arduino MEGA (Figura 40), responsável pelo controle de todos os demais módulos. Por ter mais poder de processamento e mais portas de saídas e entradas em relação aos demais modelos do gênero, foi utilizada no protótipo.
- Módulo Relé: este módulo, ilustrado na Figura 74, contempla uma placa com oito relés de 12 volts e todos os componentes (resistores e transistores) para ganho de corrente necessário pelo relé, visto que o mesmo necessita de 40mA para o seu funcionamento, não entregue pelo Arduino. O controle de cada dispositivo funciona de modo independente, pois cada relé é ligado a uma porta de saída diferente no Arduino e a um único componente, seja ele uma lâmpada ou uma tomada.

Figura 74 – Módulo Relé.



Fonte: Dx, 2013.

➤ Módulo Sensor: formado por um sensor PIR (*Passive Infrared*). Cada sensor é ligado a uma porta de entrada do Arduino e corresponde a um cômodo. Seu funcionamento baseia-se na detecção de movimento por calor (infravermelho). A cada movimento detectado, ele envia um sinal para o Arduino, que, dependendo da configuração, pode ligar uma lâmpada, acionar uma tomada, disparar o alarme ou ativar as três opções em paralelo. O Sensor PIR é demonstrado na Figura 75.

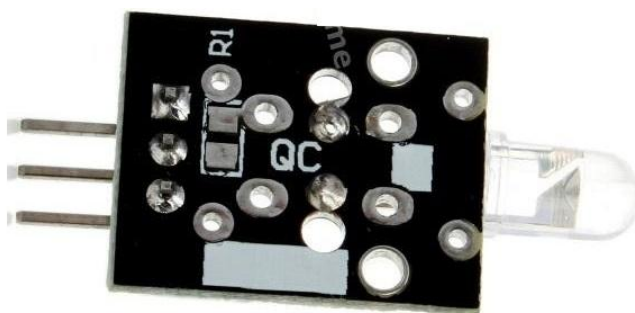
Figura 75 – Módulo Sensor.



Fonte: Dx, 2013.

➤ Módulo Controle Remoto: formado por um emissor Infravermelho, responsável no projeto pelo controle da televisão da marca CCE, modelo 5130, representado pela Figura 76. Este módulo é conectado a uma porta de saída do Arduino e é capaz de enviar diversos formatos de código, variando de acordo com o modelo do equipamento controlado.

Figura 76 – Módulo Controle Remoto.



Fonte: Dx, 2013.

➤ **Módulo Servo Motor:** é constituído de um motor que apresenta movimentos baseados em ângulos, ao invés de girar livremente como acontece com os demais motores. O tipo de servo motor, utilizado no projeto, possui a faixa de posicionamento de aproximadamente 180 graus, ou seja, ele consegue girar de 0 a 180 graus com movimentos precisos. Essa precisão, no movimento, fez com que fosse escolhido para efetuar o controle do portão de elevação na maquete residencial, com a seguinte relação: portão fechado, posição do motor em 0 grau, portão aberto, posição do motor em 90 graus.

Esse dispositivo possui três fios, o vermelho para alimentação, o preto para o terra e o amarelo ligado ao pino de saída do Arduino. Este é o cabo pelo qual chegam os sinais que informam o ângulo no qual o servo deve se posicionar.

Na Figura 77, é apresentado o Servo Motor.

Figura 77 – Servo Motor.



Fonte: Dx, 2013.

➤ **Módulo Motor de Passo:** esse motor é utilizado em dispositivos que necessitam de movimentos precisos e contínuos, amplamente utilizados em impressoras, robôs, câmeras de vídeo, brinquedos, automação industrial, entre outros dispositivos. Por conta disso, foi escolhido para controlar a abertura e fechamento do portão de trilho da maquete, porém é necessário ser conectado ao módulo driver motor de passo, Figura 78.



Figura 78 – Módulo Motor de Passo L298N.



Fonte: Dx, 2013

➤ Módulo *Driver* do Motor de Passo: consiste em um *driver* para acionamento do motor de Passo, conhecido como Pont H. Sua função é controlar a velocidade e o sentido de rotação do motor. Esse módulo tem como vantagem o poder de controlar até dois motores simultâneos.

A ligação entre o *driver* e o Arduino segue o seguinte esquema:

In 1: recebe o sinal PWM vindo do Arduino, ele diz quão rápido o motor 1 deve funcionar, variando entre 0 e 255, sendo 0 parado e 255 velocidade máxima.

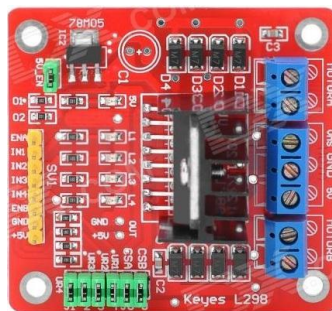
In 2: recebe o sinal digital que informa o sentido da rotação do motor 1, sendo que 0 significa reverso e 1 frente.

In 3: recebe sinal PWM para a velocidade do motor 2.

In 4: recebe sinal digital para direção do motor 2.

Este módulo está representado na Figura 79.

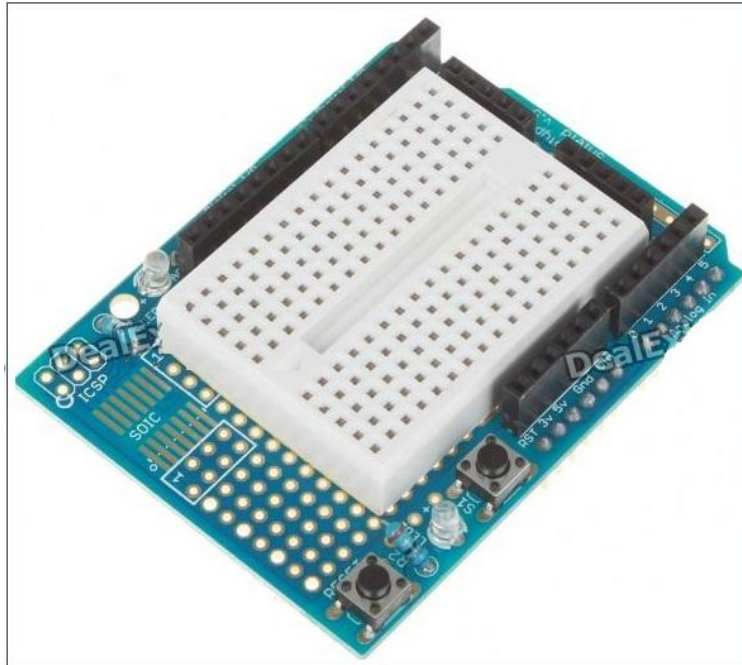
Figura 79 – Módulo Driver Motor de Passo L298N.



Fonte: Dx, 2013.

- Protoboard: utilizado para prover a conexão entre o controlador e os demais módulos do sistema, conforme ilustrado na Figura 80.

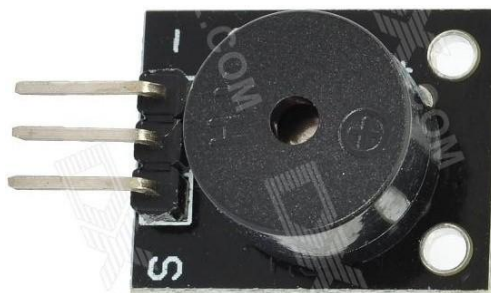
Figura 80 – Protoboard.



Fonte: Dx, 2013.

- Módulo Speaker Buzzer: responsável por emitir o aviso sonoro em consequência do acionamento do alarme, simulando uma sirene. Este módulo, conforme apresentado na Figura 81, é conectado diretamente a uma porta de saída do controlador.

Figura 81 – Módulo Sirene.



Fonte: Dx, 2013.

- Módulo Bluetooth: responsável por implementar a tecnologia de comunicação sem fio entre o Android e Arduino. É conectado diretamente às portas Tx e Rx do controlador. Este módulo é apresentado na Figura 82.

Figura 82 – Módulo Bluetooth.



Fonte: Dx, 2013.

- Módulo RTC (*real-time clock*): Sua tradução para o Português significa relógio de tempo real. O modelo adquirido é conhecido como DS1302 e tem seu uso em aplicações em que se faz necessário ter informações de data e hora atual. No protótipo desenvolvido, ele é utilizado para o acionamento da iluminação e tomada, quando configurados no modo de acionamento por horário. Além dos fios de alimentação, ele possui os pinos RST, I/O (dados de linha) e o SCLK (relógio de serie).

Esse módulo é apresentado na Figura 83.

Figura 83 – Módulo RTC.



Fonte: Dx, 2013.

## 5.7 MAQUETE

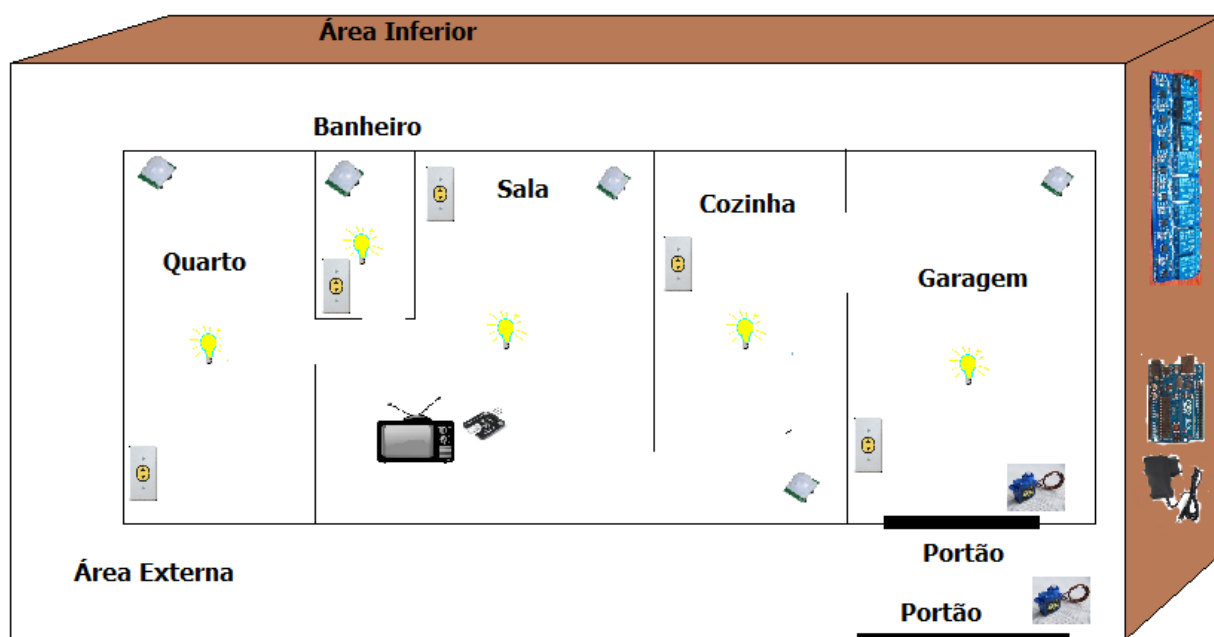
Para a realização do protótipo, além do desenvolvimento do software e do hardware, foi construída uma maquete residencial, com a finalidade de testar e validar o projeto desenvolvido, assim como demonstrar a viabilidade de solução de baixo custo.

A estrutura da maquete ficou dividida nos seguintes cômodos:

- Sala: possui uma lâmpada, uma tomada, um sensor e um transmissor de Ir para controle da televisão.
- Cozinha: Esse cômodo possui uma lâmpada, uma tomada e um sensor.
- Quarto: possui uma lâmpada, uma tomada, um sensor.
- Banheiro: Esse cômodo possui uma lâmpada, uma tomada e um sensor.
- Garagem: Esse cômodo possui uma lâmpada, uma tomada, um sensor, um sensor e um portão.
- Ambiente Inferior: semelhante a um porão, destinado à instalação da placa Arduino, fonte de alimentação, *protoboard* e a placa de relé.
- Ambiente Externo: possui o portão principal.

A Figura 84 apresenta o modelo da estrutura da maquete com seus cômodos automatizados e a distribuição física dos dispositivos.

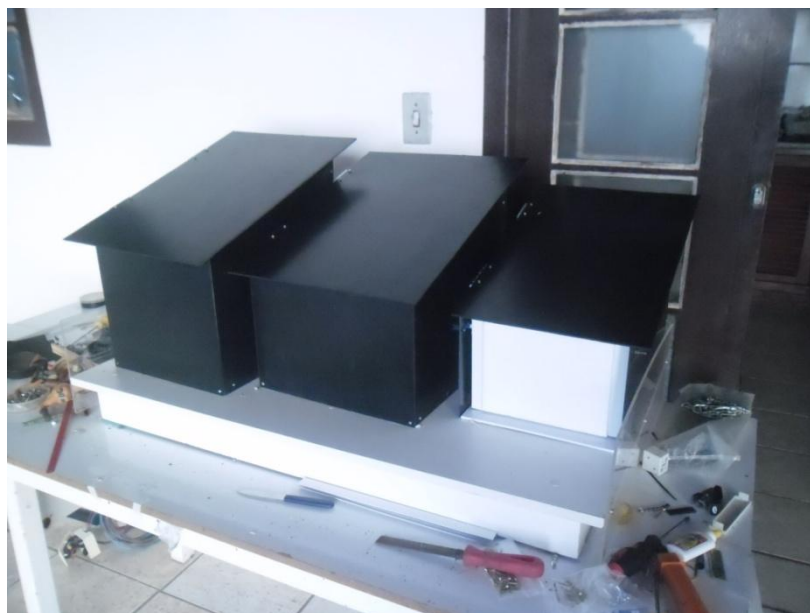
Figura 84 – Modelo da Maquete.



Fonte: Elaboração do autor, 2013.

A maquete final, desenvolvida em sua grande parte de plástico pode ser vista na Figura 85.

Figura 85 – Maquete Residencial.



Fonte: Elaboração do autor, 2013.

A Figura 86 apresenta o interior da maquete com seus pontos de iluminação.

Figura 86 – Maquete Residencial com seus pontos de Iluminação.



Fonte: Elaboração do autor, 2013.

A localização dos pontos de tomadas automatizadas pode ser visualizada na Figura 87.

Figura 87– Maquete Residencial com as Tomadas.



Fonte: Elaboração do autor, 2013



## 5.8 APRESENTAÇÃO DO SISTEMA

O protótipo proposto por este trabalho foi construído com base na modelagem de software e hardware, apresentada no capítulo 4 e com base nas tecnologias descritas na seção 5.2.

### 5.8.1 Tela Principal

A tela principal representa o menu do sistema, possui todas as opções de controle dos dispositivos do protótipo e uma opção para as configurações do sistema.

A tela principal é demonstrada na Figura 88.

Figura 88 – Tela Principal.

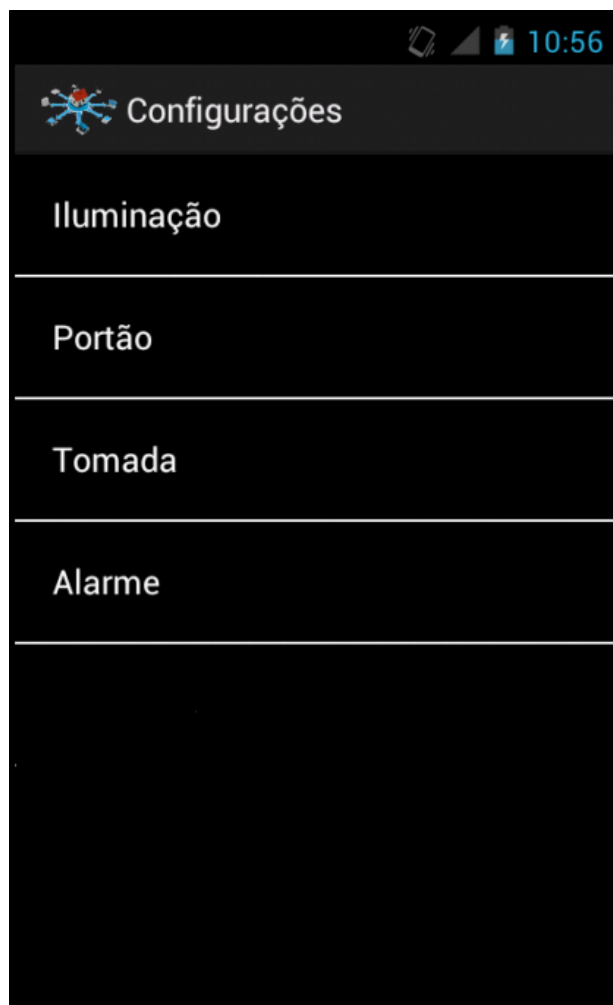


Fonte: Elaboração do autor, 2013.

### 5.8.2 Tela de Configurações

Na tela principal, ao clicar no botão com a descrição “Configurações”, a tela com as opções de dispositivos é apresentada, conforme mostra a Figura 89.

Figura 89 – Opções de Configurações.



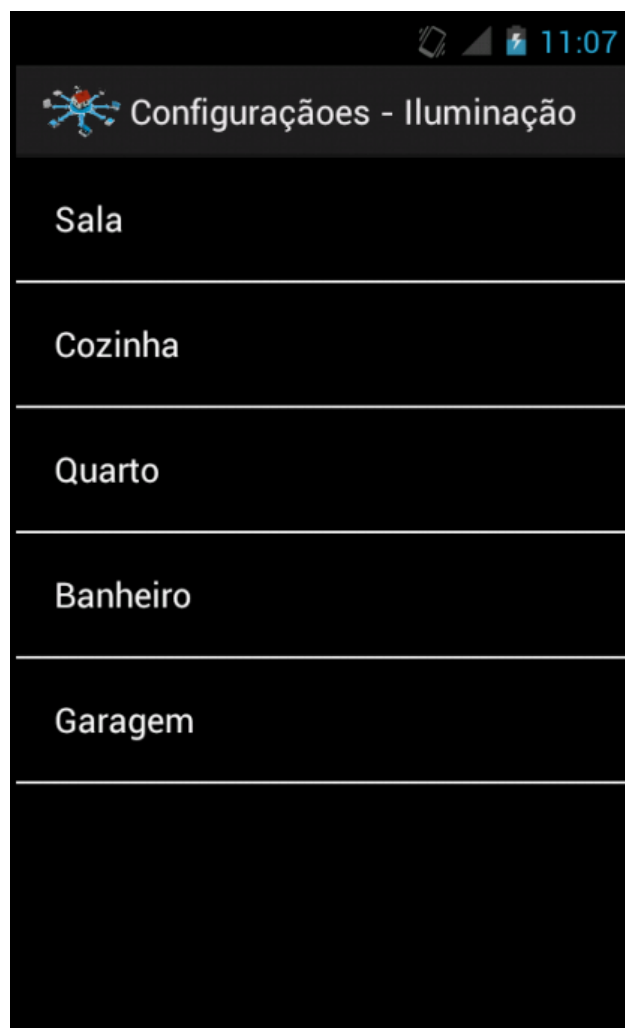
Fonte: Elaboração do autor, 2013.



### 5.8.3 Configurações da Iluminação

Essa tela está relacionada com as opções de cômodos na qual a configuração pode ser aplicada, conforme mostra a Figura 90.

Figura 90 – Cômodos para configuração da Iluminação.



Fonte: Elaboração do autor, 2013.

#### 5.8.4 Modo de Acionamento Horário

A configuração da iluminação está relacionada com o modo de acionamento de cada lâmpada associada ao seu cômodo. Os três modos de acionamento configuráveis são: Horário, Sensor e Manual.

É importante frisar que somente um modo de acionamento pode ser selecionado por vez.

Nesse modo de acionamento, o usuário configura a hora em que a lâmpada deverá ser acionada, assim como a hora em que ela deverá ser desligada, conforme ilustrado na Figura 91. Nessa configuração, o controle ligar/desligar é feito de forma automática, sem a intervenção do usuário. Essa configuração é persistida até que seja feita nova alteração, assim funciona tanto na configuração por Sensor quanto na Manual.

Figura 91 – Modo de Acionamento Horário.



Fonte: Elaboração do autor, 2013.

Para confirmar a alteração no modo de acionamento, é necessário que o usuário selecione o botão confirmar da Interface. A imagem do botão Confirmar pode ser visualizado na Figura 92.

Figura 92 – Botão Confirmar.



Fonte: Elaboração do autor, 2013.

Para cancelar a alteração ou fechar a janela de configuração do modo de acionamento, é necessário que o usuário selecione o botão cancelar da Interface. A imagem do botão Cancelar pode ser visualizado na Figura 93.

Figura 93 – Botão Cancelar.



Fonte: Elaboração do autor, 2013.

### **5.8.5 Modo de Acionamento Sensor**

Para esse modo de acionamento, o controle da lâmpada é feito com base em um sensor de presença. Assim que este sensor detecta um movimento, ele envia um sinal para o controlador que se encarrega de ligar a lâmpada. O tempo de acionamento que deve ser configurado nesse caso está associado ao tempo em que a lâmpada ficará acesa sem que ela receba outro sinal de sensor. Este tempo é dado em minutos.

A Figura 94 apresenta a tela de configuração para o modo Sensor.

Figura 94 – Modo de Acionamento Sensor.



Fonte: Elaboração do autor, 2013.

### 5.8.6 Modo de Acionamento Manual

No modo de acionamento Manual, o controle da lâmpada é feito de forma manual pelo usuário do sistema, ou seja, ele tem que acessar à interface para ligar ou desligar o dispositivo.

A Figura 95 mostra a configuração da iluminação da sala selecionada no modo de acionamento Manual.

Figura 95 – Modo de Acionamento Manual.



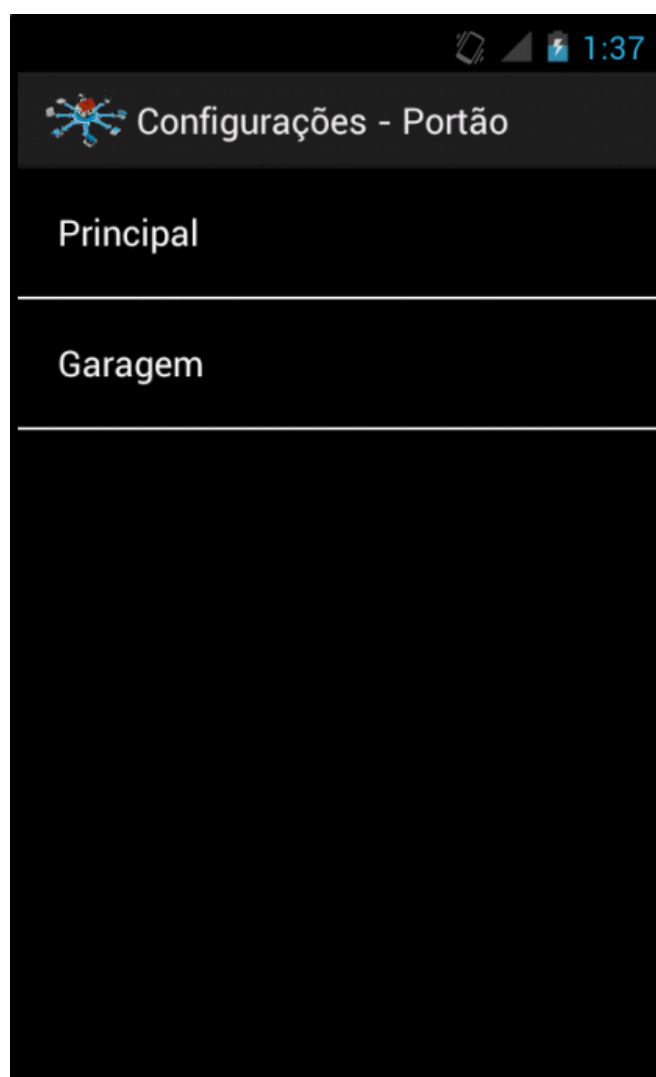
Fonte: Elaboração do autor, 2013.

### 5.8.7 Configuração do Portão

Essa tela está relacionada com as opções de ambientes que possuem portão na qual a configuração pode ser aplicada. Na maquete residencial, existem dois portões, um principal que dá acesso à rua e outro que opera entre a área interna e externa, descrito como garagem.

A Tela com as descrições dos portões existentes na maquete residencial pode ser visualizada na Figura 96.

Figura 96 – Opção de Portões para Configuração.

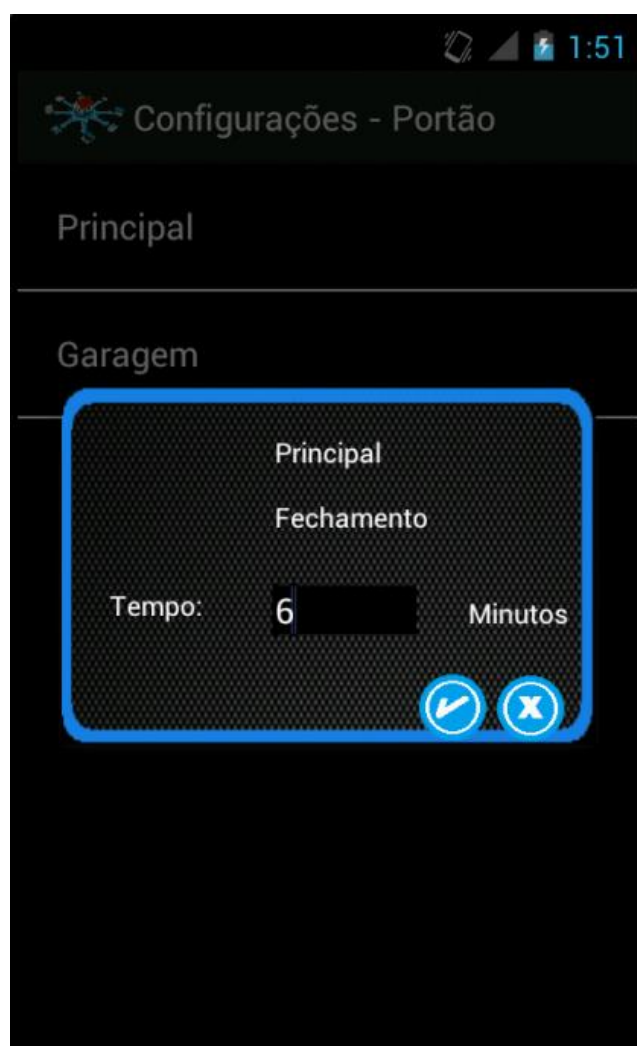


Fonte: Elaboração do autor, 2013.

### 5.8.8 Tempo de Fechamento

Essa configuração está relacionada com o tempo em que o portão ficará aberto. Esse tempo é dado em minutos, conforme mostra a Figura 97.

Figura 97 – Configuração para portão.



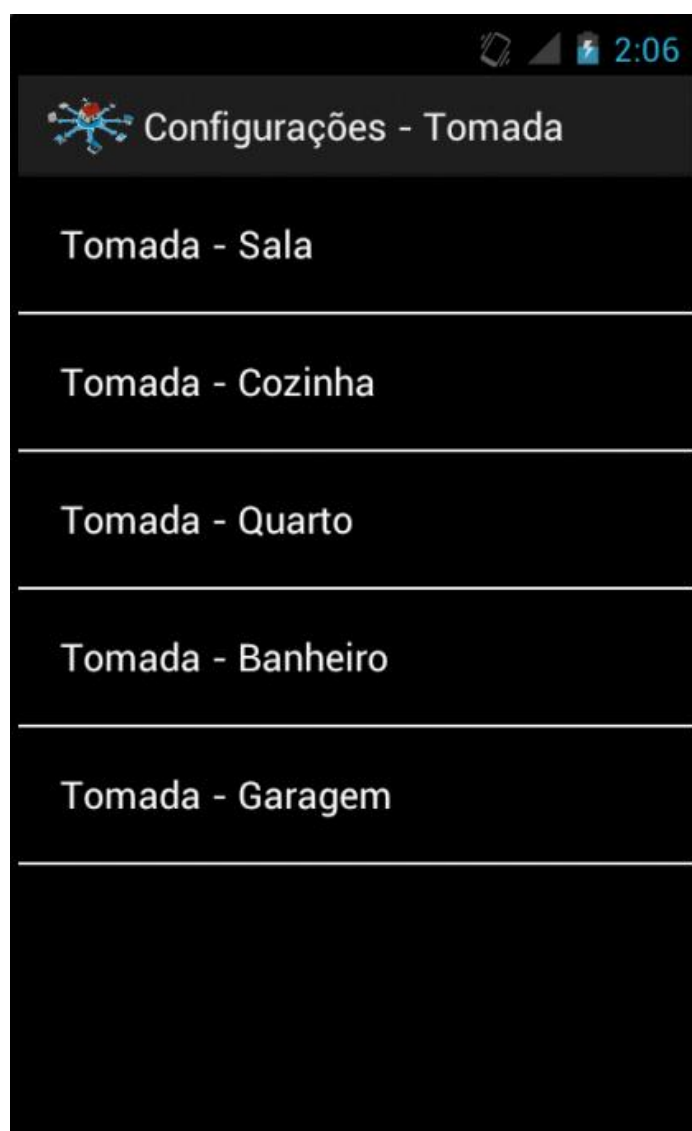
Fonte: Elaboração do autor, 2013.

### 5.8.9 Configuração da Tomada

Essa tela está relacionada com as opções de cômodos que possuem tomada com controle automatizado. Nessas tomadas, a configuração pode ser aplicada conforme ilustrado na Figura 98.

As opções de configuração para cada tomada são as mesmas aplicadas à iluminação, modo de acionamento por Horário (seção 5.8.1.4), Sensor (seção 5.8.1.5) e Manual (seção 5.8.1.6).

Figura 98 – Configurações para Tomadas.



Fonte: Elaboração do autor, 2013.

#### 5.8.10 Configuração do Alarme

Na tela de configurações do Alarme, é possível selecionar os cômodos que estarão associados ao acionamento do alarme, conforme apresentado na Figura 99. Dessa maneira, com o alarme acionado, somente os cômodos escolhidos serão monitorados.



Figura 99 – Configuração para Alarme.



Fonte: Elaboração do autor, 2013.

#### 5.8.11 Considerações sobre as Configurações

Todas as configurações efetuadas pelo usuário são salvas na base de dados SQLite do Android, posteriormente, enviadas via Bluetooth para o Arduino, que são salvas na memória EPROM (*Erasable Programmable Read-Only Memory*). EPROM é um tipo de chip de memória de computador que mantém seus dados mesmo quando a energia é desligada. Essa característica é uma grande vantagem para a execução do

protótipo, pois mesmo sobre uma queda de energia, o controlador consegue retomar as suas rotinas com os status dos dispositivos anterior à queda.

#### **5.8.12 Controle da Iluminação**

Para controlar o dispositivo, é necessário selecionar o cômodo associado à lâmpada. O controle é executado de forma independente sobre cada ponto de iluminação. Com a tela de controle aberta, é possível ligar ou desligar a lâmpada.

Para ligar a lâmpada, o usuário deve pressionar o botão com a descrição “ON”, conforme a Figura 100. Para desligar, basta pressionar o botão com a descrição “OFF”.

Esse tipo de controle é liberado somente com o modo de acionamento selecionado como Manual. Caso contrário, após a seleção do cômodo, uma mensagem é exibida ao usuário, informando que o processo não pode ser executando e qual o modo de acionamento está configurado.

Figura 100 – Controle da Iluminação – Lâmpada Desligada.



Fonte: Elaboração do autor, 2013.

A Figura 101 apresenta a tela de controle da Iluminação, com o botão apresentando a descrição “OFF”, indicando que a lâmpada está ligada.

Figura 101 – Controle da Iluminação – Lâmpada Ligada.



Fonte: Elaboração do autor, 2013.

### 5.8.13 Controle do Portão

Para controlar o dispositivo, é necessário selecionar o ambiente associado ao portão. O controle do portão é executado de forma independente em relação ao outro, visto que, na maquete residencial, existem dois portões. Com a tela de controle aberta, é possível abrir ou fechar cada portão.

Para abrir o portão, o usuário deve pressionar o botão com a descrição “Abrir”, conforme Figura 102. Para fechar, basta pressionar o botão com a descrição “Fechar”.

Em paralelo ao controle manual do portão, existe o tempo de fechamento configurado pelo usuário. Estando esta variável configurada com um valor acima de zero, o portão é fechado de forma automática, levando em consideração esse valor, caso contrário (valor igual à zero), o portão só é fechado decorrente a solicitação do usuário.

Figura 102 – Controle do Portão – Portão Fechado.



Fonte: Elaboração do autor, 2013.

A Figura 103 apresenta a tela de controle do portão, com o botão apresentando a descrição “Fechar”, indicando que o mesmo está aberto.

Figura 103 – Controle do Portão – Portão Aberto.



Fonte: Elaboração do autor, 2013.

#### 5.8.14 Controle do Alarme

Com a tela de controle aberta, é possível acionar ou desativar o alarme. Nessa mesma tela, são visíveis os cômodos associados ao alarme.

Para acionar o alarme, o usuário deve pressionar o botão com a descrição “ON”, conforme a Figura 104. Para desativar, basta pressionar o botão com a descrição “OFF”.

Figura 104 – Controle do Alarme – Alarme Desativado.



Fonte: Elaboração do autor, 2013.

A Figura 105 apresenta a tela de controle do alarme, com o botão apresentando a descrição “OFF”, indicando que o alarme está acionado.

Figura 105 – Controle do Alarme – Alarme Ativado.



Fonte: Elaboração do autor, 2013.

#### 5.8.15 Controle da Televisão

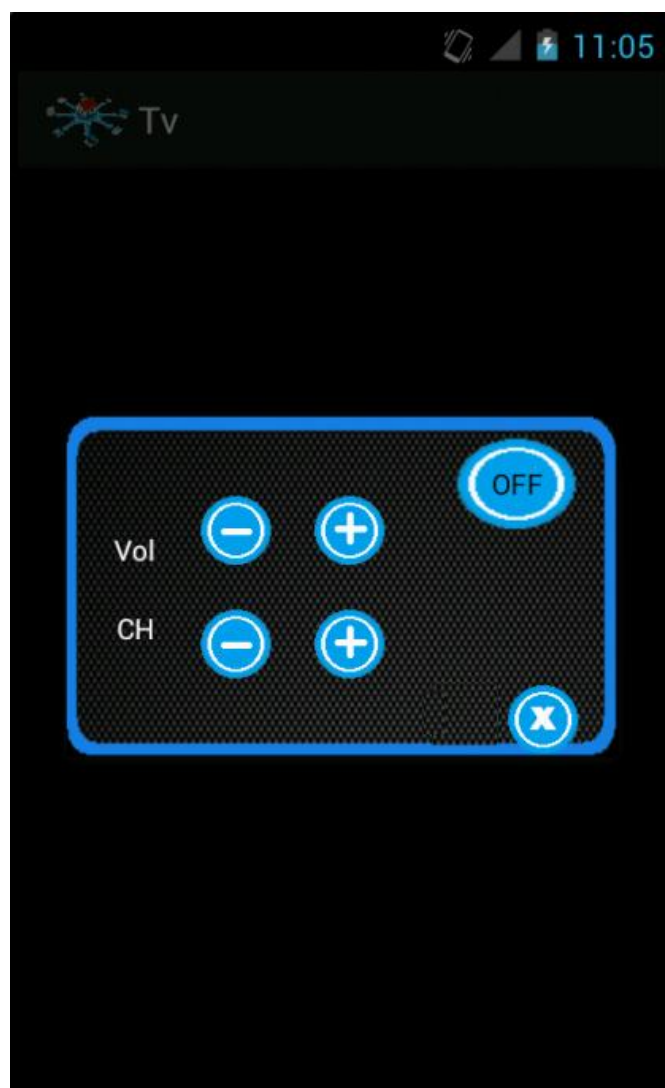
Para ligar a televisão, o usuário deve pressionar o botão com a descrição “ON” e para desligar, basta pressionar o botão com a descrição “OFF”, da mesma forma que funciona o controle da iluminação.

O volume e o canal são controlados nos botões com a descrição “+” (aumentar) e “-” (diminuir).



A Figura 106 apresenta a tela de controle da televisão com todos os botões de controle.

Figura 106 – Controle da Televisão.



Fonte: Elaboração do autor, 2013.

#### 5.8.16 Controle da Tomada

As tomadas do protótipo têm o mesmo princípio de controle que os dispositivos de iluminação, conforme descrito na seção 5.8.1.12.

A Figura 107 apresenta a tela de configuração da tomada.

Figura 107 – Controle da Tomada.



Fonte: Elaboração do autor, 2013.

## 5.9 AVALIAÇÃO

O controle dos dispositivos na maquete residencial por uma rede sem fio e unificados em um único controle apresenta vantagens em relação aos atuais métodos.

A comodidade, facilidade e acessibilidade são os aspectos que mais se destacam, principalmente, levando em consideração o uso deste protótipo por pessoas com algum tipo de deficiência física, por exemplo: locomoção.

Em relação ao consumo de energia, devido ao fato do projeto ser aplicado a uma maquete, não foi possível validar o baixo consumo de energia em relação ao sistema tradicional.

Em ambiente aberto, sem interferências de paredes e outros objetos, a comunicação via Bluetooth entre Android e Arduino se manteve estável em um raio aproximado de 9 metros, já, com obstáculos entre os dispositivos, a distância caiu para 7 metros.

Umas das diferenças entre o protótipo aplicado na maquete e uma residência, seria a substituição do módulo motor aplicado no controle do portão da maquete por um módulo relé, para o acionamento do motor apropriado ao portão da residência. Os demais módulos continuariam como apresentados neste trabalho.

Com base no funcionamento do protótipo na maquete, a continuação do projeto se mostrou viável, a fim de torná-lo um produto comercialmente rentável.

## 6 CONCLUSÕES E TRABALHOS FUTUROS

Este capítulo apresenta as conclusões finais deste trabalho em relação ao protótipo de automação residencial desenvolvido com base nos modelos propostos, assim como sugestões de trabalhos futuros para outras aplicações voltadas para a mesma área. Também, é apresentada uma breve descrição sobre as partes do trabalho que o diferencia.

### 6.1 CONCLUSÕES

Para atingir o objetivo proposto por este trabalho, vários assuntos referentes à área de automação residencial foram estudados, inclusive soluções já prontas e vendidas atualmente no mercado, procurando nesses algo que pudesse ser adicionado ao trabalho desenvolvido tornando-o um diferencial.

O resultado final do trabalho envolveu a criação de um protótipo para um sistema de automação residencial, constituído de uma interface desenvolvida em Android e a montagem de hardware, baseado em Arduino e seus periféricos, responsáveis pelo controle e configurações dos dispositivos de automação residencial aplicado a uma maquete residencial construída pelo próprio autor. Com base nos atuais sistemas de automação residenciais existentes, a ideia foi obter um sistema, que implementasse boa parte das funcionalidades encontradas nos demais sistemas e ainda assim pudesse ser oferecido e vendido a um baixo custo, compatível com a renda da classe média.

A demora relacionada ao recebimento dos componentes de hardware, comprados na China, fez com que o autor, aplicasse um maior esforço (horas trabalhadas por dia) na etapa do desenvolvimento e testes de hardware, para que o projeto fosse finalizado em tempo hábil.

Em relação ao monitoramento dos dispositivos, foi pensado na criação de uma tela na qual todos os dispositivos ficassem visíveis e com seus status atualizados, dando uma visão macro de todos os dispositivos residenciais automatizados.

A comunicação entre Android e Arduino, via o módulo Bluetooth adquirido, mostrou-se satisfatória para os testes executados na maquete, porém, para a aplicação em uma residência, este meio de comunicação teria que ser melhorado para que fosse possível abranger distâncias maiores.

O protótipo desenvolvido como objeto resultante da pesquisa realizada para a solução do trabalho atendeu as expectativas do autor em relação ao controle de forma automatizada dos diferentes tipos de dispositivos residenciais, como: televisão, iluminação, tomadas, portões e alarme.

A fase da montagem dos componentes foi a mais preocupante e a mais delicada, todas as conexões e montagens ocorreram de forma cautelosa e sistemática, garantindo que nenhum componente fosse danificado. O êxito deste processo se deu ao fato do estudo aprofundado sobre cada detalhe e a modelagem construída. Desta forma, constata-se a relevância da etapa de modelagem em todo o projeto de desenvolvimento de hardware/software.

Todos os componentes de hardware comprados corresponderam com o esperado, exceto o módulo responsável por gerenciar o tempo (hora e data). Nos testes realizados, o componente só exibia as horas, minutos e segundos ímpares, quando pares, apareciam com valor zero. Isso ocasionou impacto no controle da iluminação e das tomadas para o modo de acionamento por horário, pois qualquer configuração de controle associada a um valor par não era correspondido. Sobre a solução desse problema, pouco se conseguiu de apoio. Em um único caso encontrado na internet, o usuário solucionou o problema trançando os fios de ligação com o Arduino entre si, diminuindo ou eliminando uma possível causa de interferência/ruído. Essa dica foi realizada pelo autor, porém sem sucesso. A solução paliativa foi fazer o tratamento na mão, ou seja, cuidar com desenvolvimento de software o controle da hora par com base nos dados ímpares da hora.

Em relação ao aspecto comercial, com base na aplicação em maquete, ficou aparente a viabilidade que o protótipo tem de se tornar um produto final, com as principais funcionalidades dos sistemas atuais e, principalmente, a um preço inferior aos vendidos no mercado.

A principal diferença entre a aplicação do protótipo na maquete a uma residência fica por conta do controle do portão, já que para tal seria necessário o uso de um módulo relê (o mesmo utilizado na maquete para controle da iluminação e tomada) para o acionamento do motor apropriado a cada tipo de portão.

Algumas e importantes lições aprendidas com este projeto não devem ser deixadas passarem em branco, tais como:

- Compras Internacionais: apesar do bom preço oferecido por produtos que no Brasil custariam três vezes mais, caso haja necessidade de rapidez na entrega, a compra acaba não valendo tanto a pena pela demora estabelecida entre o ato da compra e o recebimento em mãos do produto, pois o impacto pode ser grande de acordo com a necessidade dos mesmos.
- Produtos e Ferramentas com boa documentação: antes de começar a desenvolver um software ou um produto, é de suma importância que haja sempre uma boa pesquisa em torno do material utilizado ou desenvolvido, pois na ocorrência de um problema, com boas documentações em mãos, fica bem mais fácil de ser solucionado.
- Aquisição de componentes sobressalentes de Hardware: muitas vezes, o orçamento acaba sendo um gargalo para o desenvolvimento de projetos, porém o ideal é que os principais componentes, aqueles que se queimarem ou derem algum tipo de problema acabam com o andar do projeto, sejam comprados duplicados. Isso, com certeza, garante um bom desempenho no andar do projeto, servindo, também, como forma de validar comportamentos inesperados.
- Gosto pelo tema desenvolvido: ter gosto pelo que está sendo desenvolvido é um fator óbvio, mas que vale a pena ser ressaltado. Nessa situação, o desenvolvimento ocorre de certa forma tranquila e prazerosa, apesar da pressão dos prazos, e a gratificação vem ao final do processo.
- Conhecimento na língua Inglesa: em relação a esse assunto, o importante no desenvolvimento de um projeto não é nem saber falar ou ouvir o idioma, mas, sim, saber se virar nas traduções de documentos, pois muitas das documentações boas encontradas estão neste idioma.

A complexidade exigida pelo trabalho em relação à integração entre o módulo de hardware e software levou o autor a passar por desafios que o fizeram amadurecer como profissional da área de tecnologia.

## 6.2 DIFERENCIAL DO TRABALHO

Algumas construtoras estão apostando na automação residencial como diferencial na estratégia de marketing e como forma de driblar a concorrência. Com a tecnologia presente no dia a dia das pessoas, os moradores acabam buscando cada vez mais esses itens. Dessa forma, os imóveis agregados com esses valores acabam se destacando no mercado e atraindo cada vez mais compradores. (AURESIDE, 2013b).

Com base na fundamentação teórica e com as descrições acima, o autor acredita que o assunto abordado por este trabalho possa ser considerado bastante relevante, possibilitando oportunidades voltadas para o crescimento profissional e até mesmo relacionadas a área do empreendedorismo.

O envolvimento (estudo e desenvolvimento) com a parte de hardware proporcionou ao autor obter conhecimentos que poderão ser aplicados não somente na área de automação residencial, mas também na área de automação industrial e automobilística.

Como resultado deste trabalho, obteve-se um protótipo de sistema para automação residencial, que melhor trabalhado, ou seja, com agregação de mais funcionalidades e algumas melhorias pode ser tornar um produto comercialmente rentável.

O autor considera o desenvolvimento deste trabalho completo. Trabalho este que envolveu o desenvolvimento de software como interface para o usuário manipular o sistema, desenvolvimento de *firmware* para o controlador de dispositivos, que nesse caso corresponde ao Arduino, ligação entre o controlador e os componentes de automação com os dispositivos automatizados, a integração entre software e hardware e, para finalizar, a construção da maquete residencial como forma de testar o protótipo.

## 6.3 TRABALHOS FUTUROS

Embora exista satisfação com a solução deste projeto, o mesmo ainda não está encerrado e são inúmeras as funcionalidades que podem ser agregadas em trabalhos futuros. Algumas das melhorias ou novas funcionalidades são descritas a seguir:

- Aplicação de alguns conceitos de Inteligência Artificial de maneira que o sistema aprenda com a rotina do usuário.
- Melhorar o sistema de segurança através de acesso biométrico e utilização de câmeras na residência.
- Aplicar e adaptar o protótipo desenvolvido para a maquete em uma residência.
- Melhorar aspectos de usabilidade e ergonomia para a aplicação desenvolvida em Android, como, por exemplo, mostrar os status dos dispositivos em forma de ícones, ao invés de mensagens exibidas somente após a execução das funcionalidades.
- Garantir o controle através de botões e interruptores de todos os dispositivos automatizados pelo protótipo, permitindo que funcionem em paralelo ao controle do software.
- Melhorar a comunicação entre Android e Arduino, assim como a segurança, seja via Bluetooth ou outro meio de comunicação, de modo que a comunicação possa ser alcançada em distâncias maiores que as apresentadas pelos testes do projeto.
- Permitir acesso web ao controle dos dispositivos.
- Implementar módulo (software e hardware) para controle de equipamentos, via sinais de infravermelho, de modo que o módulo consiga gravar os comandos de qualquer controle remoto e os utilize de forma automática.
- Implantar demais sensores para automatização de outras funções na residência, tais como, sensor de gás (sistema anti-incêndio), temperatura (acionamento do ar condicionado ou ventilador), umidade (controle de cortinas) e som (controle de lâmpadas e ou tomadas).
- Desenvolver módulo que possa analisar, controlar e monitorar o consumo dos principais recursos utilizados pela residência, tais como luz, água, gás e telefone.



## REFERÊNCIAS

- ALARMESHOW, Soluções em Segurança. **Produtos**. Disponível em: <  
[http://www.alarmshow.com.br/?conteudo=produtos&id\\_sub=40](http://www.alarmshow.com.br/?conteudo=produtos&id_sub=40)> Acessado em: 18 de  
 Maio de 2013.
- ALBUQUERQUE, Fernando. **Tcp/Ip Internet Protocolos e Tecnologias**. 3. Edição. Rio  
 de Janeiro: Axcel, 2001.
- ANGEL, P. M. **Introducción a la domótica; Domótica: control y automación**.  
 Escuela Brasileño-Argentina de Informática. EBAI. (1993)
- ARANA, Luiz Alejandro Vinatea. **Aqüicultura e desenvolvimento sustentável:**  
 subsídios para a formulação de políticas de desenvolvimento da aqüicultura brasileira.  
 Florianópolis: Editora da UFSC, 1999.
- AURESIDE, Associação Brasileira de Automação Residencial. **Protocolos**.  
 Disponível em: <  
<http://www.aureside.org.br/temastec/default.asp?file=protocolos09.asp>>. Acesso em: 30  
 Set. 2012a.
- AURESIDE, Associação Brasileira de Automação Residencial. **Panorama do  
 mercado**. Disponível em:  
 <[http://www.revistaautomatizar.com.br/2011/painel/cobertura/apresentacoes/aureside.p  
 df](http://www.revistaautomatizar.com.br/2011/painel/cobertura/apresentacoes/aureside.pdf)>. Acessado em: 30 Set. 2012b.
- AURESIDE, Associação Brasileira de Automação Residencial. **Wall Street Journal  
 indica interesse de grandes empresas em AR**. Disponível em:  
 <[http://www.aureside.org.br/noticias\\_recentes/default.asp?file=01.asp&id=344](http://www.aureside.org.br/noticias_recentes/default.asp?file=01.asp&id=344)>  
 Acessado em: 24 março de 2013a.
- AURESIDE, Associação Brasileira de Automação Residencial. **Diferencial  
 Competitivo**. Disponível em:  
 <[http://www.aureside.org.br/temas/artigo\\_construcao.pdf](http://www.aureside.org.br/temas/artigo_construcao.pdf)> Acessado em: 02 maio de  
 2013b.
- ARAÚJO, Lucínio Preza de. **Domótica: Automação da casa**. Disponível em: <  
[www.prof2000.pt/users/lpa/Domótica.ppt](http://www.prof2000.pt/users/lpa/Domótica.ppt)> Acessado em 04 Out. 2012.
- BANZI, M. **Getting Started With Arduino**. Sebastopol, California, Estados Unidos:  
 O'reilly Media, 2008
- BARBOSA, Andre G. **Automação Residencial e Inteligência Artificial – Redes  
 Neurais**. Disponível em:  
 < <http://handcorp.blogspot.com.br/2011/02/redes-neurais-artificiais-rna.html>> Acessado  
 em 14 Set. 2012.
- BEZERRA, Eduardo. **Princípios de análise e projeto de sistemas com UML**. Rio de  
 Janeiro: Editora Campus, 2002.

BOEIRA JUNIOR, Marcelo. **Conectando Xbee + 2 Arduinos**. Disponível em: <<https://sites.google.com/site/marceloboeirajr/tutoriais/eletronica-senai/arduino/6-shields-componentes/6-1-conectando-xtbee-2-arduinos>> Acessado em 15 Out. 2012.

BOLZANI, Caio Augustus Moraes. **Residências Inteligentes**. 1ª Edição. São Paulo: Editora Livraria da Física, 2004.

BOTELHO, Luís. **Guia para o capítulo de revisão da literatura**. Lisboa: ISCTE-IUL, 2009.

BLODSHEDSOFTWARE. **DevC++**. Disponível em: <<http://www.bloodshed.net/devcpp.html>>. Acessado em 13 Maio de 2013.

CARVALHO, Paulo. **Tecnologias da casa do futuro se tornam itens básicos de prédios**. Disponível em: <<http://www.carvalhoconsultoria.com.br/2011/04/09/tecnologias-da-casa-do-futuro-se-tornam-itens-basicos-de-predios/>>. Acessado em 14 Set. 2012.

CASADOMO. **Dómotica – Introducción**. Disponível em: <<http://www.casadomo.com/noticiasDetalle.aspx?c=14>> Acessado em 06 Out. 2012

CIRIBELLI, Marilda Corrêa. **Como elaborar uma Dissertação de Mestrado através da pesquisa científica**. Rio de Janeiro: Viveiros de Castro, 2003.

CYBERTRONICS AUTOMAÇÃO. **Tecnologia UPB**. Disponível em: <<http://www.cybert.com.br/index.php/universal-powerline-bus-sistemas-de-automacao-residencial>> Acessado em 30 Set. 2012

Do UOL. **Pesquisa diz que Android lidera mercado nos EUA, mas iPhone vende mais aplicativos**. Disponível em: <<http://www.treecnologia.uol.com.br/noticias/redacao/2012/08/27/pesquisa-diz-que-lidera-mercado-nos-eua-mas-iphone-vende-mais-aplicativos.html>>. Acessado em 02 Ago. 2012.

DIGITAT. **Protocolo X10**. Disponível em: <<http://digitat.info/modules/sections/index.php?op=viewarticle&artid=5>>. Acessado em 30 Set. 2012.

ELETRÔNICA, Saber. **O que são Sistemas Embarcados**. Disponível em: <<http://www.sabereletronica.com.br/secoes/leitura/274>> Acessado em 14 Out 2012.

EMBEDDED, Architects. **O que é um Sistema Embarcado**. Disponível em: <<http://www.embarc.com.br/p1600.aspx>> Acessado em 14 Out. 2012

EMPRETEC. **Como surgiu o Arduino**. Disponível em: <[http://www.empretecnet.com.br/do/Noticia/57/como\\_surgiu\\_o\\_arduino](http://www.empretecnet.com.br/do/Noticia/57/como_surgiu_o_arduino)> Acessado em 14 Out. 2012.

EUZÉBIO, Michel Vinicius de M.; MELLO, Emerson R. **DroidLar: Automação Residencial através de um celular Android**. IF-SC, 2011.

EXAME. **Dispositivos móveis deverão ocupar o mercado no PC.** Disponível em: <<http://exame.abril.com.br/tecnologia/noticias/dispositivos-moveis-deverao-ocupar-o-mercado-do-pc>> Acessado em 14 Out. 2012

EXOLINK. **T-Mobile G1 Android Hard Reset.** Disponível em: <<http://exolink.blogspot.com.br/2012/08/t-mobile-g1-android-hard-reset.html>> Acessado em 15 Out. 2012.

FARAHANI, Shahn. **ZigBee Wireless Network and Transceivers.** Burlington: Newnes, 2008

FLÁVIA, Jobstraibizer. **Desvendando as Redes sem fio.** São Paulo: Digerati Books, 2010.

FRITZING. **From prototype to product.** Disponível em: <<http://fritzing.org/>>. Acessado em 12 Maio de 2013.

GAMMA, Erich. RICHARD, Helm. JOHNSON, Ralph. VLISSIDES, John. **Padrões de Projeto.** Porto Alegre: Bookman, 2000.

GDS Automação Residencial. **O que é Automação Residencial.** Disponível em: <[http://www.gdsautomacao.com.br/public/index.php?option=com\\_content&view=article&id=51:o-que-e-automacao-residencial&catid=1:latest-news](http://www.gdsautomacao.com.br/public/index.php?option=com_content&view=article&id=51:o-que-e-automacao-residencial&catid=1:latest-news)> Acessado em 30 Set. 2012.

GIL, Antonio Carlos. **Como Elaborar Projetos de Pesquisas.** 4. Ed. São Paulo: Atlas, 2002.

GROOVER, M. P. – **Automation Production Systems and Computer-Integrated Manufacturing.** New Jersey. Prentice – Hall, 2001.

G1. **Empresa fatura R\$ 1 milhão com serviços de automação residencial.** Disponível em: <<http://g1.globo.com/economia/pme/noticia/2013/05/empresa-fatura-r-1-milhao-com-servicos-de-automacao-residencial.html>>. Acessado em 03 Junho de 2013.

KEN, Arnaldo. GOSLING, James. HOLMES, David. **A Linguagem de programação Java.** 4. Ed. Porto Alegre: Bookman, 2007.

KIOSKEA. **Redes sem Fio – Wireless Networks.** Disponível em: <<http://pt.kioskea.net/contents/wireless/wlintro.php3>> Acessado em 12 Out. 2012.

INTELICASA. **Soluções inteligentes em automação residencial.** Disponível em: <<http://www.intelicasa.net>>. Acessado em: 29 Set. 2012.

INFOWESTER. **Tecnologia Bluetooth: o que é e como funciona.** Disponível em: <<http://www.infowester.com/bluetooth.php>>. Acessado em: 24 Jun. 2013

IHC, Technologies. **Automação Residencial: Um mercado em expansão.** Disponível em: <<http://ihclub.com.br/site/index.php/2011/12/29/automacao-residencial-um-mercado-em-expansao>> . Acessado em 30 Set. 2012.

ILUFLEX. **Automação sem Fio**. Disponível em: < <http://iluflex.com.br>>. Acessado em: 18 Março de 2012.

JOHNSON, Thienne M. **Java para Dispositivos Móveis**. São Paulo:Novatec, 2007.

JUSTEN, Álvaro. **Curso de Arduino**. Disponível em: < <http://www.CursoDeArduino.com.br/apostila>> Acessado em 14 Out 2012.

LABORATÓRIO AVANÇADO DE SISTEMAS DE AUTOMAÇÃO. **Matriz de contatos**. Material didático da Fundação Municipal de Ensino de Piracicaba – FUMEP – Escola de Engenharia de Piracicaba. Disponível em: < <http://www.ceset.unicamp.br/~leobravo/>>. Acessado em 29 Maio de 2013.

LECHETA, Ricardo R. **Google Android Aprenda a criar aplicações para dispositivos móveis com o Android SDK**. 2ª Edição. São Paulo: Novatec, 2011.

LINS, Vitor. MOURA, Waldson. **Domótica: Automação Residencial**. Recife – PE: UNIBRATEC, 2009.

MAKESYS. **Fábrica de Software: Enterprise Architecture (EA) / RUP / UML**. Disponível em: < [http://www.makesys.com.br/EA\\_RUP\\_UML.html](http://www.makesys.com.br/EA_RUP_UML.html) > Acessado em 11 Maio de 2013.

MARTINS, José Carlos Cordeiro. **Gerenciando projetos de desenvolvimento de software com PMI, RUP e UML**. 4º Ed. Rio de Janeiro: Brasport, 2007.

MARTINS, José Carlos Cordeiro. **Técnicas para Gerenciamento de Projetos de Software**. Rio de Janeiro: Brasport, 2007.

MESSIAS, Antônio Rogério. **Controle remoto e aquisição de dados via XBee/ZigBee (IEE 802.15.4)**. Disponível em: < <http://www.rogercom.com/ZigBee/ZigBee.htm>> Acessado em 13 Out. 2012.

MITCHELL, Tom. **Machine Learning**. WCB / McGrall-Hill – Computer Science Series. Boston, MA. 1997.

MOECKE, Marcos. **ZigBee**. Disponível em: <<http://wiki.sj.ifsc.edu.br/wiki/index.php/ZigBee>> Acessado em 13 Out. 2012.

MONZILLA. **Firefox**. Disponível em: <<http://br.mozdev.org/>> Acessado em 12 Maio de 2013.

MORIMOTO, Carlos E. **Entendendo os Sistemas Embarcados**. Disponível em: <<http://www.hardware.com.br/artigos/entendendo-sistemas-embarcados/>> Acessado em 14 Out. 2012

MURATORI, José Roberto e Paulo Henrique Dal Bó. **Soluções em automação residenciais**. Disponível em: <<http://www.exatron.com.br/artigos/10-3.pdf>>. Acessado em 14 Set. 2012.

PEREZ, Anderson Luiz Fernandes, **Automação Residencial: A Tecnologia a nosso Favor**. Florianópolis: UFSC, 2009.

PINHEIRO, José Mauricio Santos. **ZigBee em Home Area Network**. Disponível em: <[http://www.projetoderedes.com.br/artigos/artigo\\_zigbee\\_em\\_home\\_area\\_network.php](http://www.projetoderedes.com.br/artigos/artigo_zigbee_em_home_area_network.php)> Acessado em 13 Out. 2012.

PREDIALTEC. **Feira de automação e tecnologia em São Paulo**. Disponível em: <<http://www.predialtec.com/sp/>>. Acessado em 30 Set. 2012.

PORFÍRIO, Alexandre. **Brasileiros lideram acesso a web por dispositivos móveis**. Disponível em: <<http://www.alexandreporfirio.com/2012/10/11/noticias/brasileiros-lideram-acesso-a-web-por-dispositivos-moveis/>> Acessado em: 14 Out. 2012.

QUINCAZE. **A tecnologia por trás da mágica**. Disponível em: <<http://quicaze.com/a-tecnologia-por-tras-da-magica/>> Acessado em: 06 Set. 2012.

RABELLO, Ramom Ribeiro. **Android: um novo paradigma de desenvolvimento móvel**. Disponível em: <[http://www.cesar.org.br/site/files/file/WM18\\_Android.pdf](http://www.cesar.org.br/site/files/file/WM18_Android.pdf)> Acessado em 15 Out. 2012.

RAMPAZZO, Lino. **Metodologia Científica**. Para alunos dos cursos de graduação e pós-graduação. 3. Ed. São Paulo: Loyola, 2005.

RAMOS, Ricardo Argenton. **Treinamento Prático em UML**. Desenvolva e Gerencie seus Projetos com essa Sensacional Ferramenta. São Paulo: Digerati Books, 2006.

ROBBINS, Stephen P. **Comportamento Organizacional**. 8ª Edição. Rio de Janeiro: LTC, 1999.

RODRIGUES, Willian Costa. **Metodologia Científica**. Paracambi: FAETEC/IST, 2007.

RUSSELL, Stuart. Peter Norvig. **Inteligência Artificial**. Tradução 2ª Edição. Rio de Janeiro: Elsevier, 2004.

SAM, M.K. **How to make an Android smartphone work faster**. Disponível em: <<http://www.cybermunks.com/how-to-make-an-android-smartphone-work-faster/>> Acessado em 15 Out. 2012.

SANTOS, Vanice dos. CANDELORO, Rosana J. **Trabalhos acadêmicos**. Uma orientação para a pesquisa e normas técnicas. Porto Alegre: RS: AGE, 2006

SENA, Diane Cristina Souza, **Automação Residencial**. Dissertação de Graduação, Universidade do Espírito Santo. Vitória – ES, 2005

SILVA, Renato Afonso Cota. **Inteligência Artificial Aplicada a Ambientes de Engenharia de Software: Uma Visão Geral**. Departamento de Informática – Universidade Federal de Viçosa. Viçosa – MG, 2005

SILVA, Renato Lopes. **Manual – Tecnologia Wireless**. Programa Nacional de Tecnologia Educacional – Ministério da Educação. MEC/SEED/DITEC/CETE. 2008.

SILVA, Guilherme Silva. **iOS x Android: conheça as características de cada um e faça a sua escolha**. Disponível em: <[http://gazetaonline.globo.com/\\_conteudo/2012/12/voce\\_ag/vida/1378588-ios-x-android-conheca-as-caracteristicas-de-cada-um-e-faca-a-sua-escolha.html](http://gazetaonline.globo.com/_conteudo/2012/12/voce_ag/vida/1378588-ios-x-android-conheca-as-caracteristicas-de-cada-um-e-faca-a-sua-escolha.html)>. Acessado em 26 Março 2013.

SMAAL, Beatriz. **Automação Residencial: A Tecnologia Invade sua casa**. Disponível em: <<http://www.tecmundo.com.br/casas/9907-automacao-residencial-a-tecnologia-invade-a-sua-casa.htm>>. Acessado em 01 Out. 2012.

SCLASS. **Automação Residencial**. Disponível em: <<http://www.sclass.com.br>>. Acessado em: 17 Março de 2013.

TANENBAUM, A. S, **Redes de Computadores**. Ed. Elsevier. Tradução autorizada do idioma inglês da edição publicada por Pretince Hall. Rio de Janeiro, 2003.

TANENBAUM, A. S, **Organização Estrutural De Computadores**. 5ª Edição. São Paulo: Prentice-Hall, 2007.

TECNOSOLUTION. **Cabeamento de rede**. Disponível em <<http://tecnosolution.blogspot.com.br/2012/07/cabeamento-de-rede-cabo-par-trancado.html>> Acessado em 13 Out. 2012.

TECHTUDO. **Vizualize e gerencie bancos de dados no navegador**. Disponível em <<http://www.techtudo.com.br/tudo-sobre/s/sqlite-manager.html>> Acessado em 12 Maio 2013.

TODAOFERTA. **Tablet Genesis Gt-7240**. Disponível em <<http://todaoferta.uol.com.br/comprar/tablet-genesis-gt7240-tela-7-hd-android-4-bluetooth-1gb-ram-UU0OCHZSXI#rmcl>> Acessado em 11 Maio de 2013.

TONIDANDEL, F., Takiuchi, M., Melo, E. **Domótica Inteligente: Automação baseada em comportamento**. Congresso Brasileiro de Automática. (2004)

TOSIN, Carlos. **Conhecendo o Android**. Disponível em: <[http://www.dicas-l.com.br/arquivo/conhecendo\\_o\\_android.php](http://www.dicas-l.com.br/arquivo/conhecendo_o_android.php)> Acessado em 15 Out. 2012.

UFSC. **Curso de Linguagem Computacional C/C++**. Disponível em <[http://s2i.das.ufsc.br/downloads/Apostila\\_C\\_Cplusplus.pdf](http://s2i.das.ufsc.br/downloads/Apostila_C_Cplusplus.pdf)> Acessado em 13 de Maio de 2013.

ZEMEL, TÀRCIO. CodeIgniter Brasil: **MVC (Model – View- Controler)**. Disponível em < <http://codeigniterbrasil.com/passos-iniciais/mvc-model-view-controller/> > Acessado em 18 de Maio 2013.

VIEIRA, Vinícius. **Saiba tudo sobre Arduino.** Disponível em: <  
<http://sejalivre.org/saiba-tudo-sobre-arduino/>> Acessado em 14 Out. 2012.

WAZLAWICK, Raul Sidnei. **Metodologia de Pesquisa para Ciência da Computação.** Rio de Janeiro: Elsevier, 2009.

## **APÊNDICES**



## APÊNDICE A – CRONOGRAMA

