

**FACULDADE DE TECNOLOGIA  
FATEC SANTO ANDRÉ**

**Tecnologia em Mecânica Automotiva  
Tecnologia em Eletrônica Automotiva**

**JONATHAN LEANDRO GRANJA**

**MARK HARUKI YAMAWAKI**

**VIRTUAL KEY TECHNOLOGY: Aplicativo Para a Substituição das  
Chaves Veiculares**

Santo André  
2019

# **Tecnologia em Mecânica Automotiva Tecnologia em Eletrônica Automotiva**

**JONATHAN LEANDRO GRANJA**

**MARK HARUKI YAMAWAKI**

## **VIRTUAL KEY TECHNOLOGY: Aplicativo Para a Substituição das Chaves Veiculares**

Trabalho de Conclusão de Curso entregue à Fatec Santo André como requisito parcial para obtenção do título de Tecnólogo em Mecânica Automotiva e Tecnólogo em Eletrônica Automotiva.

Orientador: Prof.  
Wesley Medeiros Torres  
Coorientador: Tecnólogo Caio Roberto dos Santos

Santo André  
2019

## FICHA CATALOGRÁFICA

G759v

Granja, Jonathan Leandro  
Virtual Key Technology - VKTEC / Jonathan Leandro Granja,  
Mark Haruki Yamawak. - Santo André, 2019. – 72f. il.

Trabalho de Conclusão de Curso – FATEC Santo André.  
Curso de Tecnologia em Eletrônica Automotiva, 2019.

Orientador: Prof. Wesley Medeiro Torres

1. Eletrônica. 2. Automóveis. 3. Chave virtual. 4. Plataforma Arduino IDE. 5. Aplicativo. 6. Sistemas de comunicação. 7. Microcontrolador ESP32. 8. App inventor. 9. Smartphone. 10. Desenvolvimento. I. Yamawak, Mark Haruki II. Virtual Key Technology – VKTEC.

629.2

**LISTA DE PRESENÇA**

SANTO ANDRÉ, 25 DE JUNHO DE 2019.

LISTA DE PRESENÇA REFERENTE À APRESENTAÇÃO DO  
TRABALHO DE CONCLUSÃO DE CURSO COM O TEMA "VIRTUAL  
KEY TECHNOLOGY: APLICATIVO PARA A SUBSTITUIÇÃO DAS  
CHAVES VEICULARES" DOS ALUNOS DO 6º SEMESTRE DESTA  
U.E.

**BANCA**PRESIDENTE:  
PROF. WESLEY MEDEIROS TORRES \_\_\_\_\_MEMBROS:  
PROF. FERNANDO GARUP DALBO \_\_\_\_\_

SR. CAIO ROBERTO DOS SANTOS \_\_\_\_\_

**ALUNOS:**

JONATHAN LEANDRO GRANJA \_\_\_\_\_

MARK HARUKI YAMAWAKI \_\_\_\_\_

*Aos nossos familiares, amigos e professores que nos apoiaram, auxiliaram, aconselharam e ensinaram durante todo o decorrer do curso e neste projeto.*

## **AGRADECIMENTOS**

Primeiramente, agradeço a Deus pela oportunidade de cursar um curso superior e por ter chegado até esse momento pois, sem Ele, eu nada seria ou nada alcançaria.

Agradeço aos meus pais que me apoiaram desde o início, desde o momento em que decidi ingressar em uma instituição de ensino superior. Agradeço-os por sempre terem feito tudo aquilo que podiam e muito mais por mim. Muitas das vezes, se sacrificaram, renunciaram a muitas coisas para poder me fornecer o melhor. Agradeço-os pelos conselhos e pelas orações que me ajudaram chegar até aqui e me tornar quem eu sou.

Agradeço a minha noiva por sempre estar ao meu lado nas minhas decisões, por ser minha companheira, amiga e ajudadora. Me dando forças nas vezes em que pensei em desistir e apoio nos momentos de dificuldade. Me ajudando inclusive em ter uma ideia para realizar esse trabalho – mesmo sendo da área de humanas.

Agradeço também a todos os professores que conheci, os quais tive a sorte e oportunidade de ter sido lecionado, onde passaram grandes conhecimentos técnicos e sociais, me ajudando a crescer como profissional e como pessoa. Em especial, agradeço ao professor Wesley Medeiros Torres, pela orientação neste trabalho de graduação, ao nosso colega e coorientador Tecnólogo Caio Roberto dos Santos indicado pelo professor Fernando Garup Dalbo, que nos ajudaram muito a realizar nosso projeto, onde todos contribuíram para que colocássemos a cereja do bolo da nossa formação acadêmica.

(Jonathan Leandro Granja)

Agradeço primeiramente a Deus, por permanecer comigo em toda a minha caminhada e nunca permitir que nada me falte.

Agradeço a instituição e aos professores da Fatec Santo André que nos ensinaram e ensinam novos conhecimentos todos os dias, disponibilizando ferramentas para o desenvolvimento do trabalho e sempre estão à disposição para nos ajudar. Além de todos os funcionários que fazem parte da equipe da universidade, que mesmo indiretamente, fazem parte do meu sonho de graduação.

Agradeço aos meus pais, familiares e minha namorada por sempre me apoiarem, independentemente da situação. Estão ao meu lado nos momentos bons e ruins e fazem além do que podem pelo meu crescimento pessoal e profissional. Por fim, agradeço aos meus amigos que estão comigo no dia-a-dia, auxiliando meu caminho, tornando-o mais fácil de prosseguir e que serão para sempre lembrados pelos anos de convivência.

(Mark Haruki Yamawaki)

“Não to mandei eu? Esforça-te, e tem bom ânimo; não temas, nem te espantes; porque o Senhor teu Deus é contigo, por onde quer que andares.”

JOSUÉ 1:9

## RESUMO

Este trabalho consiste no desenvolvimento de uma chave virtual, denominada *Virtual Key Technology* (VKTec) que possui o intuito de substituir as tradicionais chaves mecânicas veiculares, através de um aplicativo instalado no celular do proprietário do veículo, será estabelecida a comunicação com o automóvel, permitindo realizar o trancamento e destrancamento das portas. Este aplicativo será desenvolvido através da plataforma online do MIT, denominada de *App Inventor*® o qual esta será a interface de comunicação do usuário com o veículo. A chave virtual também terá a possibilidade de compartilhamento de usuários, dessa forma o proprietário do veículo poderá compartilhar o automóvel com quem estiver cadastrado no aplicativo, permitindo que outro usuário possa fazer uso do possante. Caso o veículo seja acionado por alguém não autorizado pelo proprietário, será enviada uma mensagem de alerta para o celular “matriz”, funcionando como alerta em uma tentativa furto. O sistema de controle é baseado no microcontrolador ESP32, que possui a tecnologia *Bluetooth Low Energy* (BLE) para realizar o pareamento com o veículo, permitindo comunicação com o *Smartphone* tendo sua programação feita através da plataforma Arduino IDE.

Palavras chaves: Smartphone, ESP32, Aplicativo, App Inventor, Chave, Bluetooth, Virtual, Veículo.

## ABSTRACT

This work consist in the development of a virtual key called *Virtual Key Technology* (VKTec) aiming for substitute the traditional vehicle mechanical keys through an app where it is possible a link with the car, allowing lock and unlock the doors. An App will be assembly through MIT Online Platform, called App Inventor, which it will be the communication interface between user and vehicle. Also, this work has the possibility of user sharing, in other words, the owner of car can share with someone else who are registered in the App a code for who desire to use the vehicle. In cases where the vehicle to be activated by someone who are not authorized by the owner, it will send a warn message to the master phone, warning the user of a possible theft attempt. The process of triggering has been developed in the ESP32 with The *Bluetooth Low Energy* (BLE) to activate the pairing with the vehicle, allowing the communication with the smartphone across Arduino IDE code.

**Keys Words:** Smartphone, ESP32, Application, App Inventor, Key, Bluetooth, Virtual, Vehicle.

## SUMÁRIO

<b>LISTA DE FIGURAS</b> .....	<b>14</b>
<b>LISTA DE ABREVIATURAS</b> .....	<b>17</b>
<b>1 INTRODUÇÃO</b> .....	<b>15</b>
1.1 OBJETIVO GERAL .....	21
1.2 OBJETIVOS ESPECÍFICOS.....	21
1.3 MOTIVAÇÃO.....	22
<b>2 REFERÊNCIAS TEÓRICAS</b> .....	<b>24</b>
2.1 SMARTPHONE.....	24
2.1.1 APLICATIVOS.....	25
2.2 BLUETOOTH .....	27
2.2.1 FUNCIONAMENTO .....	28
2.2.2 BLUETOOTH LOW ENERGY.....	29
2.3 APP INVENTOR.....	30
<b>3 METODOLOGIA</b> .....	<b>32</b>
3.1 LOGOTIPO.....	32
3.2 DESENVOLVIMENTO DO APLICATIVO UTILIZANDO O APP INVENTOR .....	35
3.3 SISTEMA EMBARCADO NO VEÍCULO. ....	43
3.4 HARDWARE MONTADO .....	49
3.5 SISTEMA COMPLETO.....	52
<b>4 RESULTADOS</b> .....	<b>54</b>
4.1 CUSTOS DO PROTÓTIPO .....	58
<b>5 CONCLUSÕES</b> .....	<b>59</b>
5.1 CONCLUSÕES FINAIS .....	59
5.2 PROPOSTAS FUTURAS .....	61
CRIPTOGRAFIA ASSIMÉTRICA .....	63
<b>6 REFERÊNCIAS BIBLIOGRÁFICAS</b> .....	<b>65</b>

<b>APÊNDICE - PROGRAMAÇÃO.....</b>	<b>71</b>
------------------------------------	-----------

## LISTA DE FIGURAS

Figura 1 - Utilização de Smartphones por horas diárias pela população dos principais países do mundo. ....	17
Figura 2 - Gráfico Faturamento/Nº de Locadoras.....	18
Figura 3 – Gráfico referente à quantidade de locadoras por regiões do Brasil .....	19
Figura 4 - Gráfico referente à frota de veículos em locadoras por regiões do Brasil.	19
Figura 5 - Frota total de veículos por categoria em Locadoras no Brasil (2016) .....	20
Figura 6 - Mapa da concentração de frotas nas regiões brasileiras.....	20
Figura 7 - Ilustração da ideia do projeto.....	23
Figura 8 - Número de usuários de telefones celulares em todo o mundo de 2015 a 2020, em bilhões.....	25
Figura 9 - Sketch Smartphone .....	26
Figura 10 – Conexão Bluetooth entre diversos dispositivos .....	27
Figura 11 - Exemplo de uma <i>Piconet</i> .....	28
Figura 12 - Exemplo de uma <i>scatternet</i> .....	29
Figura 13 - Características BLE .....	30
Figura 14 - Logotipo MIT App Inventor.....	30
Figura 15 - Exemplo de Programação utilizando o APP Inventor .....	31
Figura 16 - Esquema do Desenvolvimento do Projeto.....	32
Figura 17 - Opções de logos disponíveis no site.....	33
Figura 18 - Logotipo Virtual Key Technology .....	34
Figura 19 - Página Inicial do Aplicativo - App Inventor .....	35
Figura 20 – Formatação/Gerenciamento de componentes App Inventor .....	36
Figura 21 – Ferramentas do App Inventor .....	36
Figura 22 - Programação Tela Inicial - App Inventor .....	37
Figura 23 - Tela de Login - App Inventor.....	38

Figura 24 - Programação Tela Login - App Inventor.....	38
Figura 25 - Lista de Veículos - App Inventor.....	39
Figura 26 - Programação Lista de Veículos - App Inventor.....	39
Figura 27 - Tela do veículo selecionado 1 .....	40
Figura 28 - Tela de Comunicação Bluetooth e Acionamento da Trava Elétrica .....	41
Figura 29 – Programação Comunicação Bluetooth e Acionamento do Relé .....	42
Figura 30 – Programação GPS .....	42
Figura 31 - ESP-WROOM-32 - Dimensões: 27,5 x 51,0 x 7,0 mm .....	43
Figura 32 - Diagrama dos componentes do ESP32 .....	44
Figura 33 - Pinagem ESP32 .....	45
Figura 34 - Pinos utilizados no projeto .....	46
Figura 35 - Programação Arduino IDE .....	47
Figura 36 - Fluxograma da programação ES32 .....	48
Figura 37 - Acionamento Relé.....	50
Figura 38 - Módulo Relé 5V - 2 Canais .....	51
Figura 39 - Sistema Completo.....	52
Figura 40 - Erro ao Compilar a Programação .....	55
Figura 41 - Botão Boot .....	55
Figura 42 - Sinal Bluetoooh da ESP32 disponível.....	57
Figura 43 - Sistema Criptográfico .....	62
Figura 44 - Ilustração didática sobre encriptação.....	63
Figura 45 - GPS NEO 6M .....	63

## LISTA DE TABELAS

Tabela 1 - Celulares vendidos em outubro de 2014.....	16
Tabela 2 - Conexões ESP32 e Relé .....	53
Tabela 3 - Resultados Esperados .....	57
Tabela 4 - Custos do Protótipo.....	58

## LISTA DE ABREVIATURAS

ABLA	Associação Brasileira de Locadoras de Veículo
AC	Corrente Alternada
APP	Aplicativo
BLE	Bluetooth Low Energy
BR	Bluetooth Basic Rate
CNH	Carteira Nacional de Habilitação
CPF	Cadastro de Pessoa Física
DDC	Display Data Channel
DENATRAN	Departamento Nacional de Trânsito
EEPROM	Electrically-Erasable Programmable Read-Only Memory
ESP	Espressif
EUA	Estados Unidos da América
GB	Gigabyte
GHz	Gigahertz
GM	General Motors
GPS	Global Positioning System
GND	Filtro Graduado de Densidade Neutral
HCI	Host Controller Interface
IBM	International Business Machines Corporation
IBGE	Instituto Brasileiro de Geografia e Estatística
IDE	Integrated Development Environment
IN1	INPUT 1
IN2	INPUT 2
ISM	Industrial, Scientific, Medicinal

KB	Kilobyte
KPCB	Kleiner Perkins Caufield and Byers
mA	MiliAmpère
MB	Megabyte
MHz	Megahertz
MIT	Massachusetts Institute of Technology
PWM	Pulse Width Modulation
RAM	Random Access Memory
RG	Registro Geral
ROM	Read-Only Memory
RSA	Rivest Shamir Adleman
SIG	Bluetooth Special Interest Group
SPI	Serial Peripheral Interface
UART	Universal Asynchronous Receiver/Transmitter
URL	Uniform Resource Locator
USB	Universal Serial Bus
V	Voltage (Tensão)
VCC	Voltagem (Tensão) em Corrente Contínua
VDC	Voltage Direct Current
VKTEC	Virtual Key Technology

## 1 INTRODUÇÃO

*“My job is to not be easy on people. My job is to make them better.”*

*Steve Jobs*

Inovações e ideias tecnológicas renovam-se a cada dia. Pessoas são pagas apenas para ter criatividade suficiente para inventar ou desenvolver algo que não existe ou que nunca foram criados, já outras possuem apenas o *hobby* de criar, apenas por gostar e possuir prazer de inventar, de serem criativos. Mas fato é que novas ideias surgem a milhões, onde uma vai superando a outra. A tecnologia se desenvolve em uma velocidade inigualável a ponto de que, nos atuais dias, é praticamente impossível viver sem ela. Faz parte do nosso cotidiano. Dentro deste conceito, esse trabalho situa-se na integração de um *Smartphone* com um veículo, usaremos esse pequeno aparelho de comunicação como exemplo: Na década de 90, apenas pessoas com alto poder aquisitivo possuíam um aparelho celular e como se tratava de um aparelho tão grande, não era se quer possível carregá-lo no bolso e sua única finalidade era realizar ligações telefônicas. Atualmente, até a palavra celular desapareceu de nosso vocabulário, sendo substituída pela denominação inglesa “*Smartphone*”, um pequeno celular (na maioria dos casos) que, por mais simples que seja, grande parte das pessoas possuem um aparelho e que hoje os permite acesso à internet, redes sociais, câmeras para fotos, filmagens e as mais diversas funções, deixando em segundo plano a realização de chamadas telefônicas.

Com base nesse avanço tecnológico, a popularização do uso dos *Smartphones* e a busca contínua por facilidade e praticidade da população, nosso projeto vem com o intuito de inserir ainda mais esses aparelhos em nosso dia a dia. Eles serão utilizados para a substituição das chaves automotivas, se tornando uma chave virtual, a qual denominamos de *VKTec – Virtual Key Technology*, onde o smartphone estará conectado com o veículo através de um protocolo padrão de comunicação sem fios primariamente projetado para baixo consumo de energia com baixo alcance e que está presente em todos os smartphones, o *Bluetooth Low Energy (BLE)*, permitindo realizar o travamento e destravamento das portas.

Um levantamento feito em 2016 pela Opus Software e publicado pela revista Exame, o brasileiro já utiliza mais o aparelho celular do que o computador pessoal. No final de 2014 o Brasil já era o 6º mercado mundial de smartphones, superado apenas por China, EUA, Índia, Japão e Rússia (OPUS SOFTWARE, 2016). Esses dados mostram que o mercado brasileiro de smartphones está amadurecendo, e muito rapidamente. Dados divulgados pelo instituto IBGE mostram que 77,1% da população, com 10 anos ou mais de idade, tinham um aparelho de celular próprio já em 2016 (IBGE, 2018). Quando esses dados são cruzados com os que foram disponibilizados pela Anatel no período, temos uma média de 1,7 aparelho/linha ativa por usuário (MOBILE TIME, 2018). De acordo com a pesquisa realizada pela Teleco e com base nos dados da Anatel, o Brasil encerrou o ano de 2014 com uma estatística de 279,4 milhões de celulares, sendo 137,5 celulares/100 habitante (DINO. Estatísticas de uso de celular no Brasil. *Exame*, n. 1, p. 1, 2012). Na Tabela 1, segundo dados do IBGE, é possível verificar o crescimento do número de celulares ativos entre os anos de 2013 e 2014.

Tabela 1 - Celulares vendidos em outubro de 2014

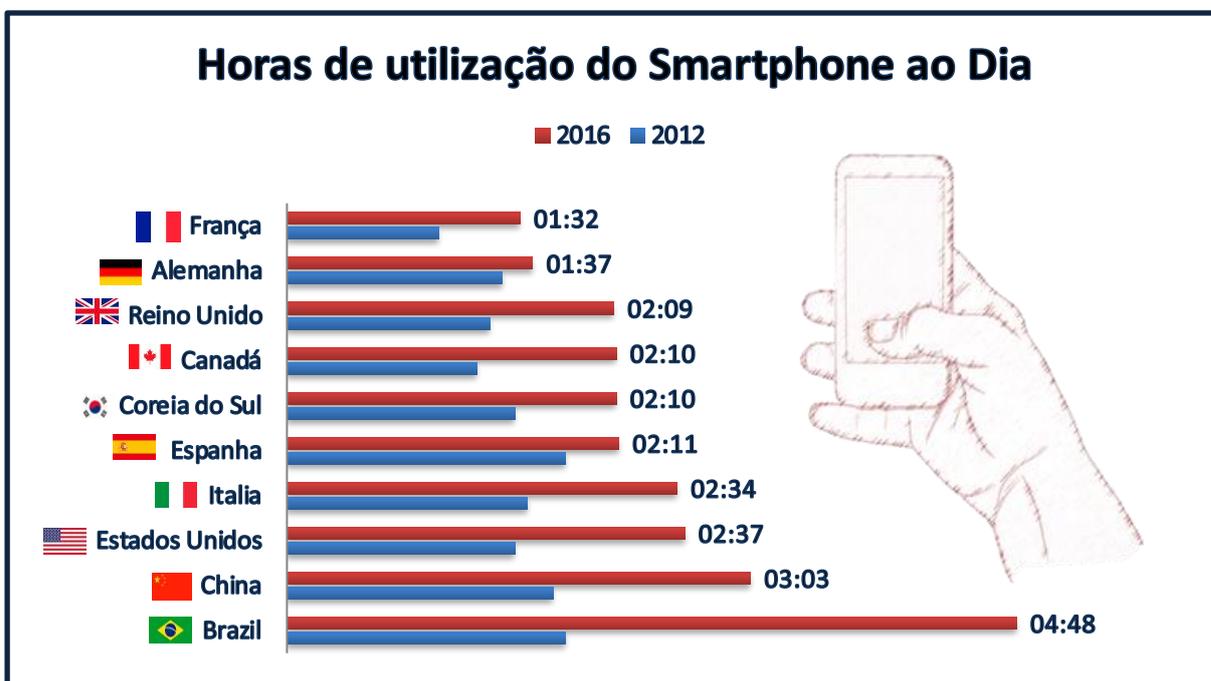
<b>Celulares em Out/14</b>				
	<b>out/13</b>	<b>dez/13</b>	<b>set/14</b>	<b>out/14</b>
<b>Celulares</b>	269.923.975	271.099.799	278.482.125	279.354.157
<b>Pré-pago</b>	78,58%	78,05%	76,49%	76,3%
<b>Densidade*</b>	133,97	134,36	137,1	135,5
<b>Crescimento Mês</b>	1.657.153	580.924	1.073.566	872.032
	0,60%	0,20%	0,40%	0,30%
<b>Crescimento Ano</b>	8.116.072	9.291.896	7.382.326	8.254.358
	3,10%	3,50%	2,7%	3%
<b>Crescimento em 1 ano</b>	10.618.124	9.291.896	10.125.303	9.430.182
	4,10%	3,50%	3,8%	3,50%

Nota: Celulares ativos na operadora. Densidade calculada com a projeção de pop. Do IBGE (Rev. 2013) para o mês respectivo.

Fonte – IBGE (2014)

Um outro aspecto importante, é o crescimento no número de horas despendidas na utilização do *smartphone* em diversos países no mundo, onde na Figura 1 é possível verificar quantitativamente o número de horas (STATISTA, 2016).

Figura 1 - Utilização de Smartphones por horas diárias pela população dos principais países do mundo.

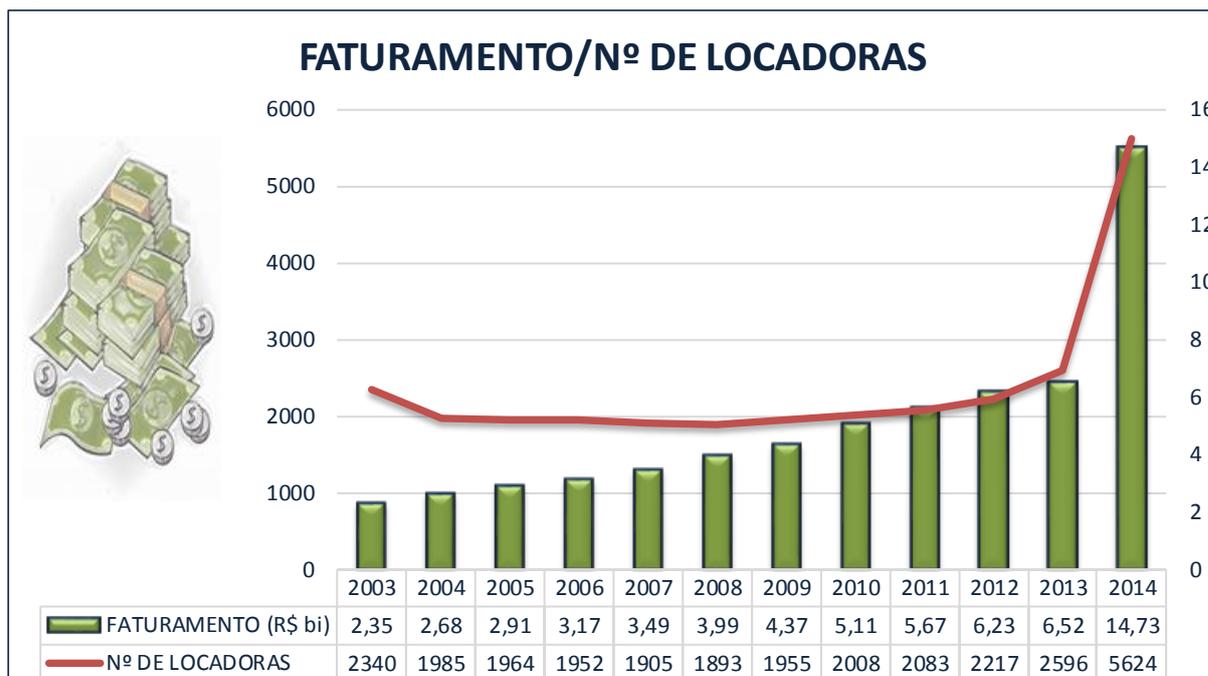


Fonte - <http://zydigital.com.br> – Pesquisa Internacional Statista

Os dados demonstram claramente que a utilização de *Smartphones* pela população é cada vez maior. Baseado nessa informação, podemos concluir que o celular possui papel muito importante no dia a dia das pessoas, então nosso projeto vem de encontro com essa tendência e a utilização do smartphone como uma chave virtual para abertura do veículo mostra-se muito interessante e prático.

O segundo ponto qual motivou a realização desse trabalho foi o intuito de contribuir com as empresas do mercado de locação de automóveis. Segundo a Associação Brasileira de Locadoras de Automóveis (ABLA), o faturamento bruto na locação de veículos cresceu 12,3% em 2017, atingindo a marca de R\$15,5 bilhões em vendas (ABLA, 2018). Na Figura 2 é demonstrado o faturamento em relação a quantidade de locadoras existente no Brasil. Segundo o jornal Valor Econômico, o número de clientes cresceu 17,2%, atingindo 23,2 milhões de usuários, sendo nestes 58% no segmento de frota, 23% referente à lazer e 19% entre clientes de negócio. (VALOR ECONÔMICO, 2018)

Figura 2 - Gráfico Faturamento/Nº de Locadoras



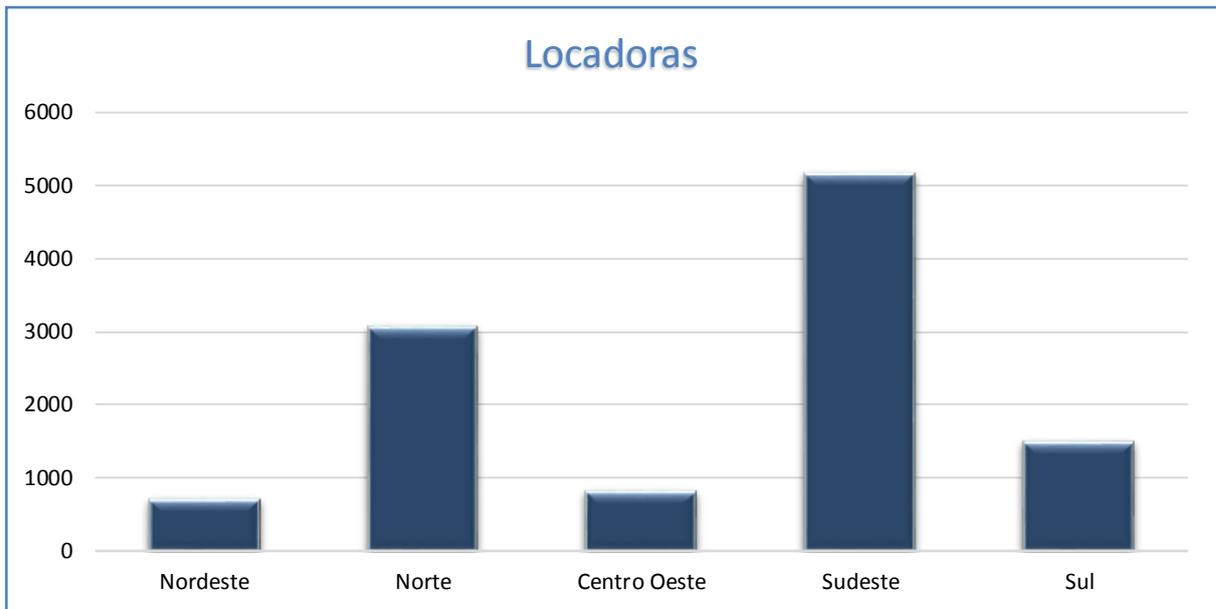
Fonte – ABLA (2015)

Ainda segundo o levantamento da ABLA publicado pelo jornal, foi apontado um crescimento no número de carros na frota total das locadoras, passado de 632,9 mil em 2016 para 709,03 mil veículos em 2017. A *General Motors* (GM) é a montadora que mais vende carro para as locadoras, com 19,6% de participação de mercado, seguida pela Fiat (19,2%), Renault (15,9%), Ford (14,8%) e Volkswagen (14,8%) (VALOR ECONÔMICO, 2018).

Grande parte da frota dos veículos adquiridos pelas locadoras são os famosos carros “populares”, motivo que leva a Unidas acreditar ser pelo fato do baixo poder aquisitivo da população brasileira, com a GM liderando o mercado com o modelo Onix (G1, 2018).

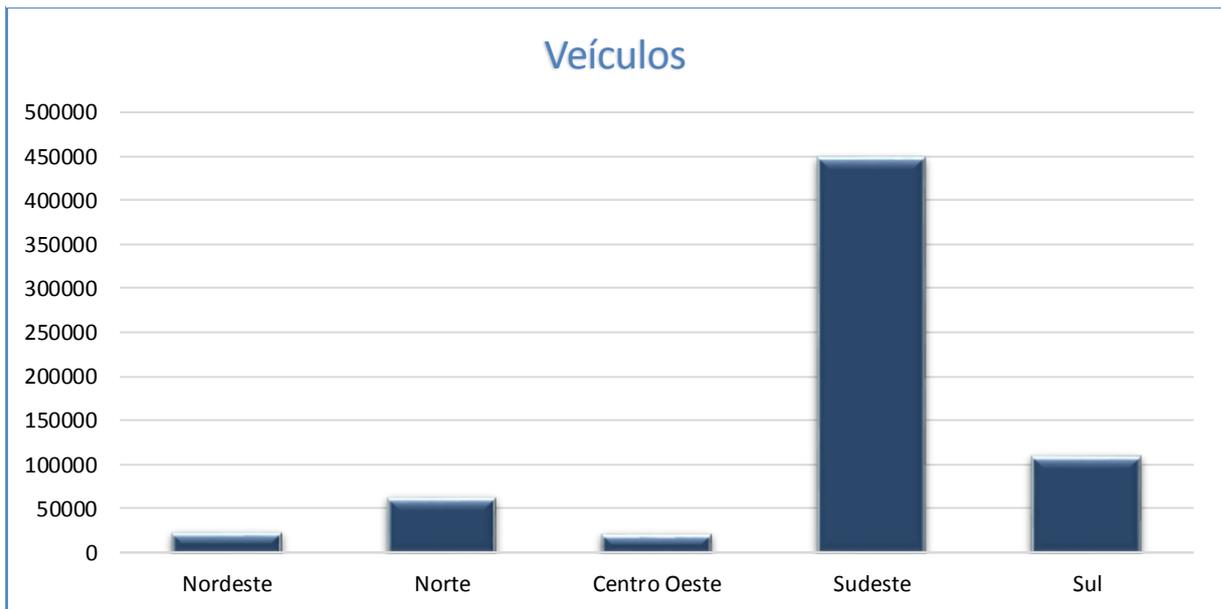
Nas Figura 3 e 4 é possível verificar a quantidade de veículos disponíveis pelas locadoras por todas as regiões do Brasil e a quantidade de locadoras por região no Brasil.

Figura 3 – Gráfico referente à quantidade de locadoras por regiões do Brasil



Fonte - ABLA

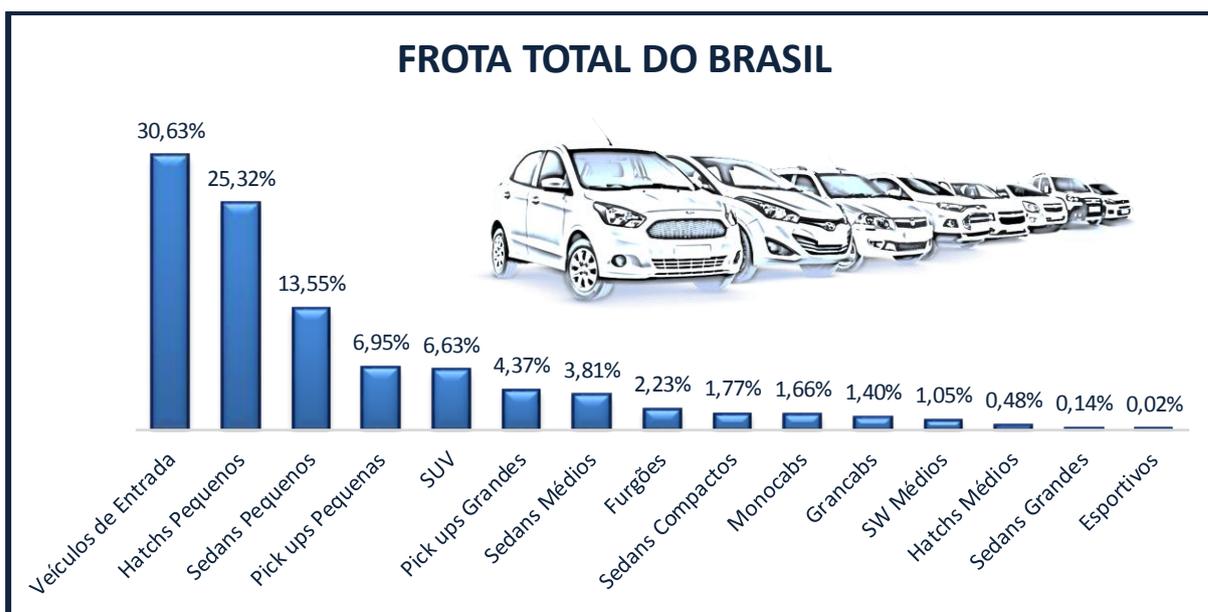
Figura 4 - Gráfico referente à frota de veículos em locadoras por regiões do Brasil.



Fonte - ABLA

Além disso, na Figura 5 está a distribuição de veículo por classe e suas respectivas quantidades.

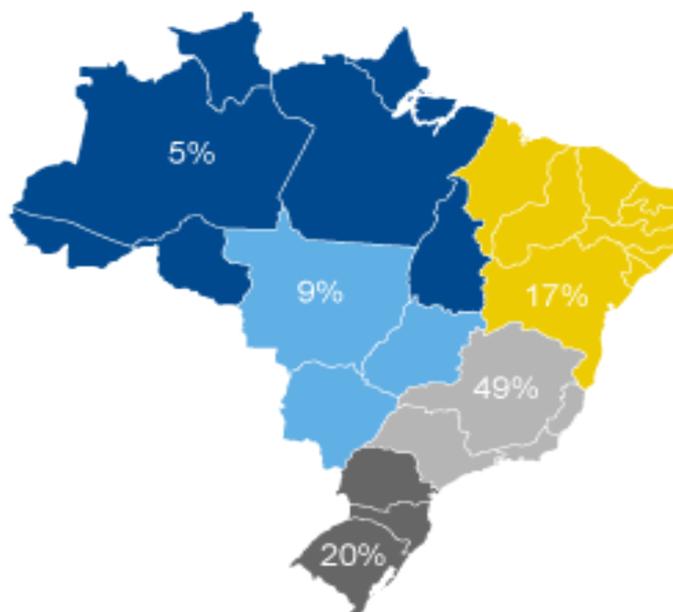
Figura 5 - Frota total de veículos por categoria em Locadoras no Brasil (2016)



Fonte - Associação Brasileira de Locadoras de Automóveis (ABLA)

Na Figura 6 é demonstrado a distribuição de locadoras por região no Brasil. Baseado nos gráficos das figuras 5 e 6, podemos concluir que a maior concentração de locadoras de veículos está no estado de São Paulo devido a concentração de maior de recursos econômicos. (DENATRAN, 2017)

Figura 6 - Mapa da concentração de frotas nas regiões brasileiras



Fonte - DENATRAN

Os levantamentos dos dados da ABLA, locadoras e mídias, demonstram o tamanho do mercado de veículos para locação no Brasil. Além disso, segundo dados da locadora Unidas, grande parte dos veículos utilizados por empresas de locação de veículos, são veículos populares e empregam chaves mecânicas, o que leva às empresas de locação de veículos possuírem um sistema de controle correto e seguro para armazenamento de todas as chaves (BLOG DAS LOCADORAS, 2017). Todo esse processo de logística e segurança no controle de todas as chaves, ocupa espaço para armazenamento e tempo, resultando em dinheiro perdido, além da necessidade do controle “humano”, o que pode também criar eventos de perda. Visando o preenchimento dessa lacuna, nosso projeto possui o intuito de contribuir e facilitar a vida dessas empresas, facilitando seu trabalho e dando inclusive a oportunidade de adquirir maior número de clientes, além de ser transformado em lucro, economia e fornecer a possibilidade de prestação de serviço personalizado.

Quando o aplicativo desenvolvido é empregado, qualquer locadora poderá ter um cadastro exclusivo, onde terá o registro e controle de todos os veículos de sua frota, cada um com um código diferente para realizar o acesso virtual, podendo compartilhá-lo de forma rápida e segura com o cliente desejado, facilitando a logística da empresa num geral pois, se tratando de uma franquia, por exemplo, todas as unidades terão o registro dos veículos no aplicativo, permitindo que o cliente possa ter o direito de locar o automóvel em uma unidade e devolvê-lo em outra, agilizando o processo para a empresa, e permitindo maior liberdade e facilidade ao cliente.

## **1.1 Objetivo Geral**

Construção de uma arquitetura tecnológica que permita adaptar smartphones para comutar a chave mecânica em chave virtual permitindo também, o compartilhamento com segurança e confiabilidade para qualquer usuário autorizando a utilização do automóvel.

## **1.2 Objetivos Específicos**

Visa-se desenvolver um projeto eletrônico que se adeque aos padrões de evolução tecnológica atual, integrando a tecnologia usual dos telefones móveis com os veículos urbanos. Para isso, desenvolveu-se um aplicativo para smartphones que

se comunica com o veículo substituindo a tradicional chave mecânica e o denominamos de *VKTec – Virtual Key Technology*.

Para a realização do projeto, foi simulado o funcionamento das travas elétricas do automóvel através de um relé, utilizando uma plataforma de desenvolvimento com microcontrolador em conjunto com a comunicação sem fios *Bluetooth* para realizar a comunicação com o veículo. Por fim, será desenvolvido um aplicativo exclusivo com o nome *VKTec* para, através dele, poder ser compartilhado o código com qualquer usuário desejado para que este possa também usufruir do veículo.

### 1.3 Motivação

Ao examinar o contexto atual do mundo globalizado, atesta-se uma intrínseca relação dinâmica e de busca por praticidade nas relações cotidianas. Diante disso, a inovação tecnológica serve como base para a evolução humana, onde se alcança maior comodidade e funcionalidade nas atividades regulares.

Em contrapartida, com às preocupações corriqueiras do dia-a-dia, e às atividades regulares a que nos submetemos, esquecer objetos como a chave do carro, pode ser um entrave para a praticidade e para o gerenciamento de tempo. Além disso, a situação de carregar itens consigo, é incômoda e constitui outro fator de perda de funcionalidade prática.

Em 2017 foram vendidos 1,85 milhão de automóveis e, uma das maiores agências de aluguel de carros, a Localiza, possui uma frota de 185 mil veículos (GLOBO G1, 2018). Com a tecnologia *VKTec*, podemos implementá-las nos veículos e ajudar o dia a dia das pessoas que o utilizarão, além de gerar lucros para a empresa que irá fabricar a tecnologia. Na Figura 7, é demonstrado a ideia do projeto. A partir disto, desenvolvemos o aplicativo para a substituição de chaves veiculares, que conta com o dispositivo usual do *smartphone* com aplicativos para exercer a função de uma chave digital que substitui a chave mecânica. Com isso, não corremos o risco de eventualmente esquecer a chave, e eliminamos o empasse de carregar mais um objeto consigo durante o dia-a-dia.

Figura 7 - Ilustração da ideia do projeto



Fonte - Google

## 2 REFERÊNCIAS TEÓRICAS

### 2.1 Smartphone

O primeiro *smartphone* da história a ser lançado foi no ano de 1994, pela empresa *IBM Simon Personal*. Nessa época ainda não existia o sistema operacional, ele apenas possuía 1 MB de memória, enviava e recebia *e-mails*, faxes e páginas permitindo acesso ao calendário e relógio mundial (TECMUNDO, 2014).

Com o passar do tempo e a evolução da tecnologia, passamos para uma nova era dos celulares. Os novos smartphones são mais leves, possuem sistemas operacionais, processadores, aplicativos, capacidade de memória, câmeras entre outras características muito mais avançadas. Hoje, é possível armazenar até 256 GB de dados em um smartphone. Além de poder compartilhar informações com outro smartphone através do *Bluetooth*.

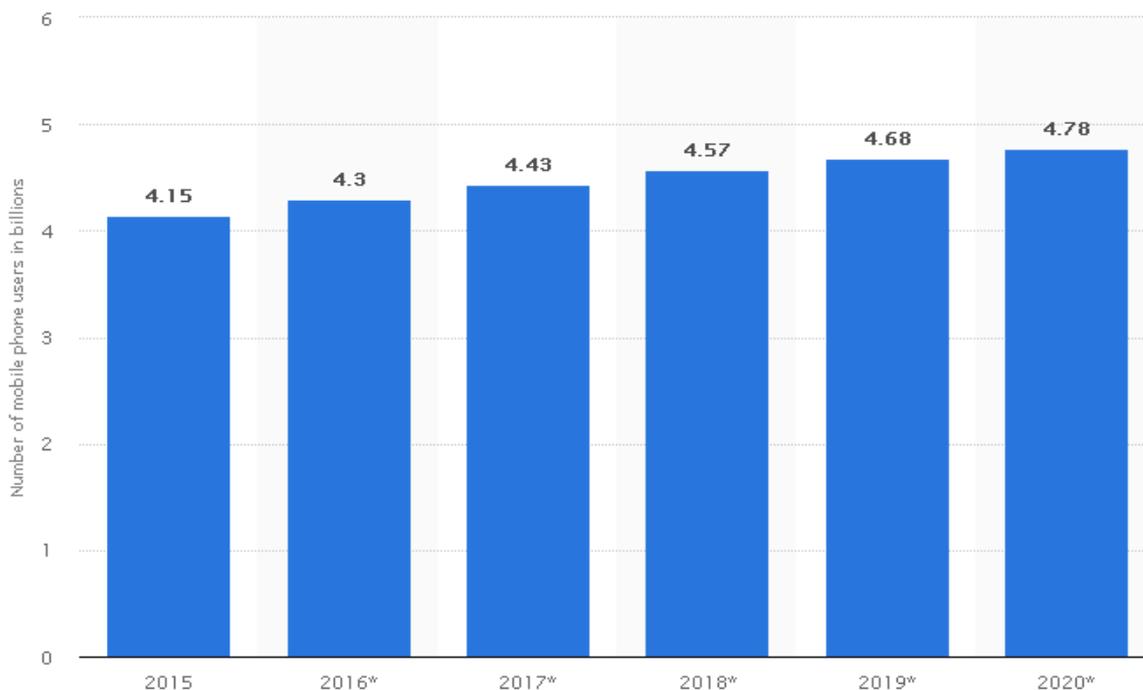
Desde 2000, quando foi disponibilizado para comércio o primeiro *Smartphone* inteligente da História, o Ericsson R380 (TECMUNDO, 2014), grandes partes das pessoas passaram a substituir seus computadores e *notebooks* por *smartphones*, pois através dele, é possível realizar as mesmas coisas que no computador. Carregando menos peso e economizando tempo e espaço. O celular é como uma dependência do ser humano nos dias atuais pois ele agrega diversas funcionalidades, como: Meios de comunicação, compartilhamento de informações, despertador, calendário, entre outros aplicativos utilizados para o dia a dia.

Uma pesquisa feita pela ZDNet, aponta que dos 7,5 bilhões de pessoas do mundo, 1,6 bilhão já estão conectadas através de dispositivos móveis. Estima-se que 350 milhões desses dispositivos são para uso corporativo. E a tendência é aumentar mais ainda com o tempo (ZDNet, 2016).

No Brasil, de acordo com a FGVcia, em 2019, temos mais de um *smartphone* por habitante. São 230 milhões de celulares em uso. Adicionando os *Notebooks* e *Tablets*, são 324 milhões de dispositivos portáteis. (FGV, 2019)

Na Figura 8, é demonstrado o número de usuários de telefones celulares em todo o mundo de a partir de 2015 e já com as perspectivas para os anos de 2019 e 2020.

Figura 8 - Número de usuários de telefones celulares em todo o mundo de 2015 a 2020, em bilhões.



Fonte - <https://www.statista.com>

### 2.1.1 Aplicativos

O aplicativo móvel, conhecido como APP, é um *software* desenvolvido por diversas empresas para ser instalado em dispositivos móveis ou smartphones, através de uma loja *online*, como por exemplo, *Apple Store®* e *Google Play®*.

São disponibilizados nestas plataformas, diversas aplicações gratuitas e pagas. Além destes aplicativos que você precisa instalar em seu dispositivo, existem também os aplicativos que já vem instalado no *smartphone* de fábrica, como calculadora, alarme, calendário, entre outros. Existem milhões de aplicativos para diversas aplicações. Os donos de *smartphones* no Brasil possuem, em média, cerca de 15 aplicativos instalados e os aplicativos mais utilizados atualmente são os de comunicação, como *WhatsApp*, *Facebook*, *Instagram* e *Twitter*. (OPUS SOFTWARE, 2016). Porém, o uso de aplicativos não está restrito à comunicação, os aplicativos de serviços financeiros, por exemplo, aplicativo bancário, onde não é mais necessário o cliente se deslocar até o agência do banco para o pagamento de contas, com o uso do aplicativo da instituição bancária instalado no *smartphone* e usando a câmera disponível no celular permite-se a leitura do código de barras para efetuar o pagamento,

além de realizar transferências bancárias, consulta de saldo etc, isso também tem levado alguns bancos a decidirem fechar agências bancaria devido a diminuição de clientes presentes em agências bancárias (UOL ECONOMIA, 2019).

Além das classes de aplicativos citados, também existem os aplicativos para jogos, realização de compras dos mais diversos tipos, assistir filmes e ouvir músicas, suportam também diferentes plataformas de sistema operacional como *iOS* (dispositivos *Apple*) e *Android*. Segundo pesquisa realizada pelo site Sensor Tower, foram realizados 194 bilhões de *downloads* de aplicativos em 2018 sendo destes 105 bilhões de *downloads* de aplicativos feitos por meio da *Google Play Store* (com quase três quartos dos *downloads*) e da *iOS App Store* (BUSINESS OF APPS, 2019)

Figura 9 - Sketch Smartphone



Fonte – Google 2015

## 2.2 Bluetooth

O *Bluetooth* é denominado uma tecnologia de comunicação sem fios que permite a comunicação e transmissão de dados entre computadores, *smartphones*, impressoras, *vídeo games*, entre outros dispositivos que tenham esta tecnologia.

A tecnologia foi desenvolvida primeiramente pela empresa Ericsson, em 1994, a fim de obter uma tecnologia para a comunicação entre celulares e acessórios sem a necessidade de utilizar cabos. Esta comunicação seria através de sinais de rádio de baixo custo. Na Figura 10 demonstramos uma conexão *Bluetooth* entre diversos dispositivos.

Figura 10 – Conexão Bluetooth entre diversos dispositivos



Fonte - <https://www.teleco.com.br>

Em 1998, formou-se o grupo SIG (*Bluetooth Special Interest Group*), que unia empresas como Nokia, IBM, Toshiba, Intel, entre outras empresas fizeram parte do grupo com o passar do tempo. Com estas empresas de diversos ramos no mercado, foi possível desenvolver diversos padrões que fornecessem interoperabilidade nos mais variados dispositivos.

O Bluetooth é uma tecnologia desenvolvida para funcionar no mundo todo, com isso, foi necessário a adoção de uma frequência de rádio aberta e aceita em praticamente qualquer lugar do planeta. A faixa ISM (Industrial, Scientific, Medicinal)

opera à frequência de 2,45 GHz, podendo variar de 2,4 GHz a 2,5 GHz. Pode-se utilizar até 79 frequências (com restrições em alguns países, passando para 23 frequências). O padrão Bluetooth é definido pelo IEEE 802.15.1 (ALECRIM, 2018).

### 2.2.1 Funcionamento

A comunicação entre dispositivos *Bluetooth* é chamada de *Piconet* (rede formada via *Bluetooth*). Esta comunicação ocorre entre mestre e escravo.

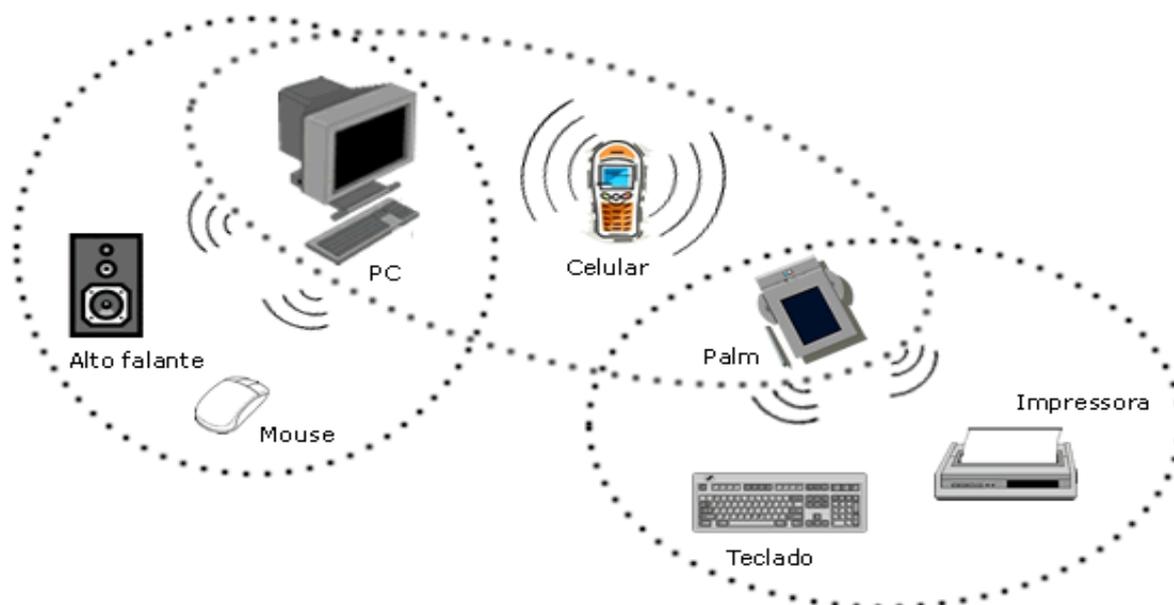
O dispositivo que está selecionado para atuar como mestre, pode se conectar a outros dispositivos como escravos e enviar/receber dados para aqueles que estiverem conectados a ele. O mestre trabalha com até 7 escravos simultaneamente. Já o escravo, pode-se conectar apenas a um mestre e é apenas com ele que há troca de informações. Na Figura 11, é possível visualizar um exemplo de uma comunicação *piconet* onde o celular atua como mestre e os outros periféricos são os escravos.

Figura 11 - Exemplo de uma *Piconet*



Fonte - <https://www.teleco.com.br>

Um dispositivo pode participar ao mesmo tempo de duas ou mais *piconets* independentes, apenas como escravo. Mesmo conectado a outras *piconets*, pode haver apenas um mestre. A comunicação de várias *piconets* simultaneamente é chamada de *scatternet*. Demonstrado na Figura 12, o celular é apenas o único escravo de todas as *piconets* e os demais periféricos são os escravos.

Figura 12 - Exemplo de uma *scatternet*

Fonte - <https://www.teleco.com.br>

### 2.2.2 Bluetooth Low Energy

O *Bluetooth Low Energy* (BLE) foi lançado juntamente ao *Bluetooth* 4.0, em 2010. Como o nome já diz, este tipo de *Bluetooth* consome uma quantidade muito pequena de energia, comparado ao *Bluetooth* comum.

Dois fatores levam o BLE ser a melhor opção para aplicações que necessitam transmitir poucos dados: o baixo custo do produto e o baixo consumo de energia. Um dispositivo BLE pode permanecer a maior parte do tempo em “modo de descanso”, pois apenas em milissegundos consegue enviar ou receber todas as informações necessárias (ALECRIM, 2018).

Como neste projeto a utilização do *Bluetooth* não necessita transmitir muitos dados, a solução foi utilizar o módulo ESP32 que possui a tecnologia do *Bluetooth Low Energy*. Através deste, conseguiremos fazer a comunicação do *smartphone* com o aplicativo, de forma que o sistema consuma o mínimo de energia possível. Na Figura 13, é destacado as características do BLE.

Figura 13 - Características BLE



Fonte - Google

## 2.3 App Inventor

Figura 14 - Logotipo MIT App Inventor

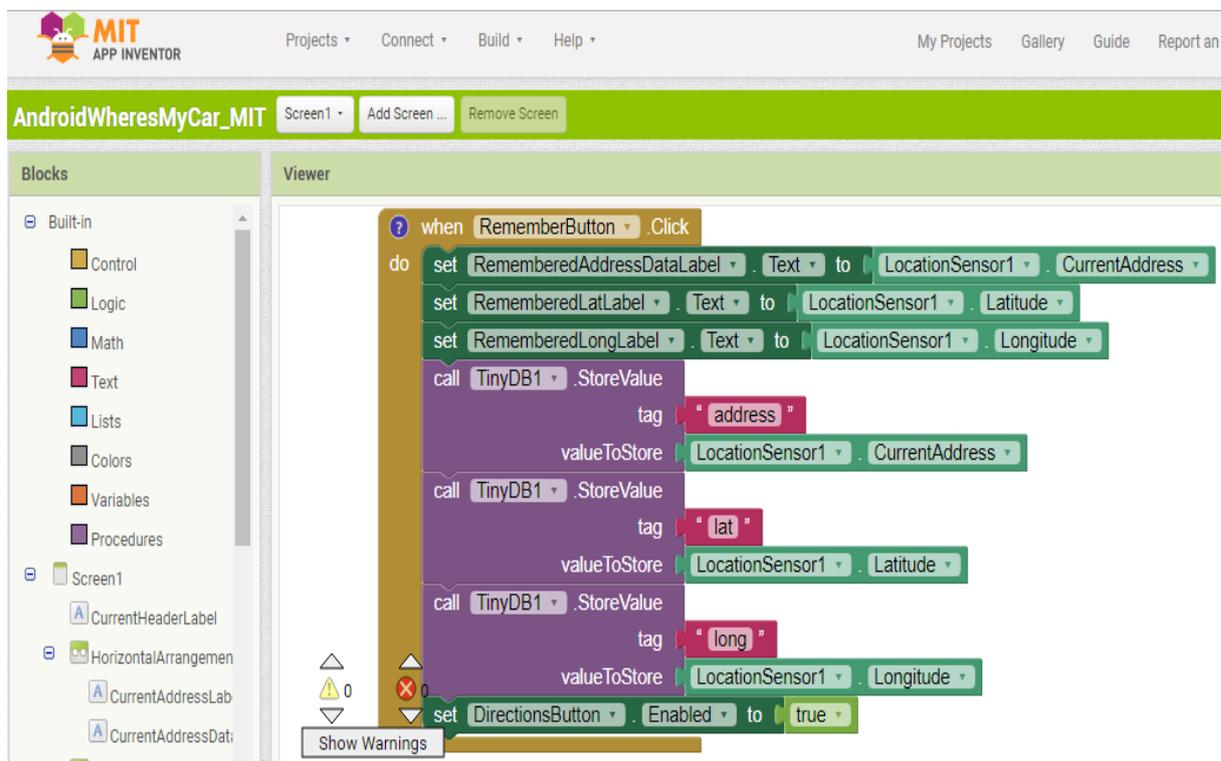


Fonte - Google

O *App Inventor* é uma ferramenta desenvolvida pela parceria entre a Google e o MIT – Instituto de Tecnologia de Massachusetts onde toda a programação é desenvolvida utilizando blocos e permite o desenvolvimento de aplicativos para dispositivos móveis e que utilizem o sistema operacional *Android*. Seu ponto forte é que o desenvolvimento não requer conhecimento elevado em programação, além de ser simples e didática. Os aplicativos podem ser desenvolvidos utilizando uma interface web, que funciona por meio do encaixe de blocos lógicos disponíveis. Além disso,

dentro dessa *interface web*, é possível realizar testes com um emulador, dessa forma, a desenvolvimento se torna intuitivo. Um exemplo de desenvolvimento utilizando o APP Inventor é demonstrado na Figura 15

Figura 15 - Exemplo de Programação utilizando o APP Inventor

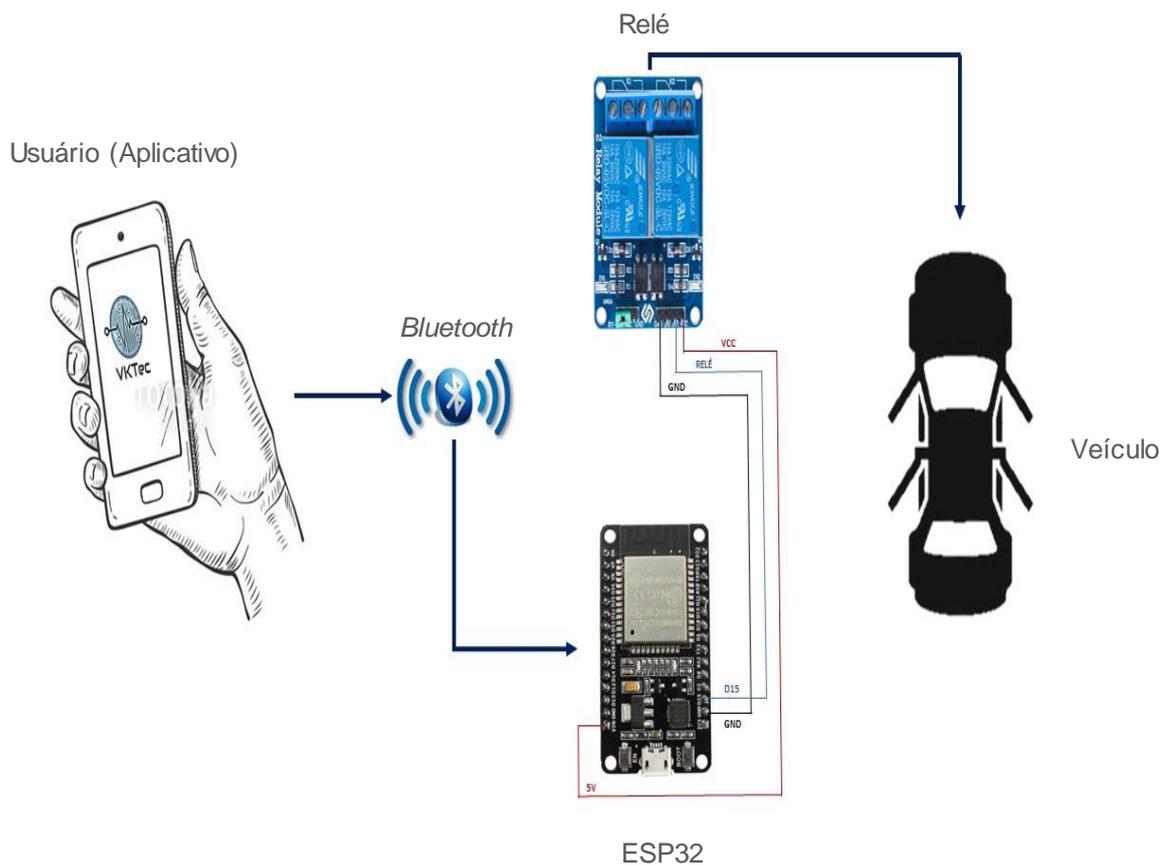


Fonte - <http://ai2.appinventor.mit.edu>

### 3 METODOLOGIA

O diagrama do sistema VK TEK pode ser visto na Figura 16 onde o usuário irá utilizar o aplicativo no *Smartphone* que estará conectado via *Bluetooth* com o sistema embarcado, baseado na placa ESP 32 que é responsável pelo controle do relé de acionamento das travas elétricas do veículo.

Figura 16 - Esquema do Desenvolvimento do Projeto



Fonte - Autoria Própria

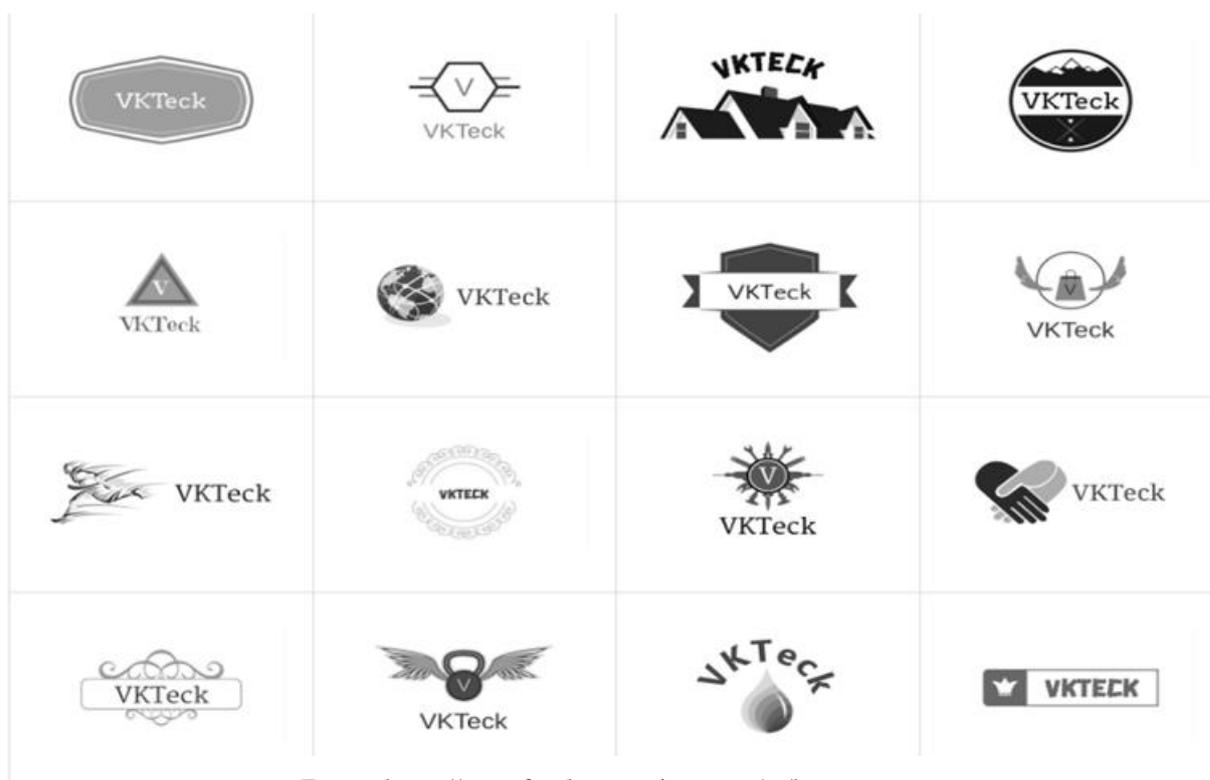
#### 3.1 Logotipo

Devido a intenção de tornar o sistema desenvolvido em um produto comercial, se faz necessário possuir uma identidade visual para despertar o interesse dos usuários e dessa forma, facilitar a venda do produto.

Para esse propósito, foi utilizada uma ferramenta disponível no *site Free Logo Services* cujo propósito é a criação de logos, ainda aliado a simplicidade para a criação.

Após alguns menus de seleção e o tipo de *design* o site fornece alguns modelos de logotipos diferentes para escolha, como podem ser vistos na Figura 17.

Figura 17 - Opções de logos disponíveis no site



Fonte - <https://www.freelogoservices.com/pt/home-return>

Dentre os diferentes modelos apresentados pelo site, mas tendo em vista a ideia inicial de produto, foi escolhido logo que melhor traduzia a ideia do projeto, como pode ser visto na Figura 18.

Figura 18 - Logotipo Virtual Key Technology

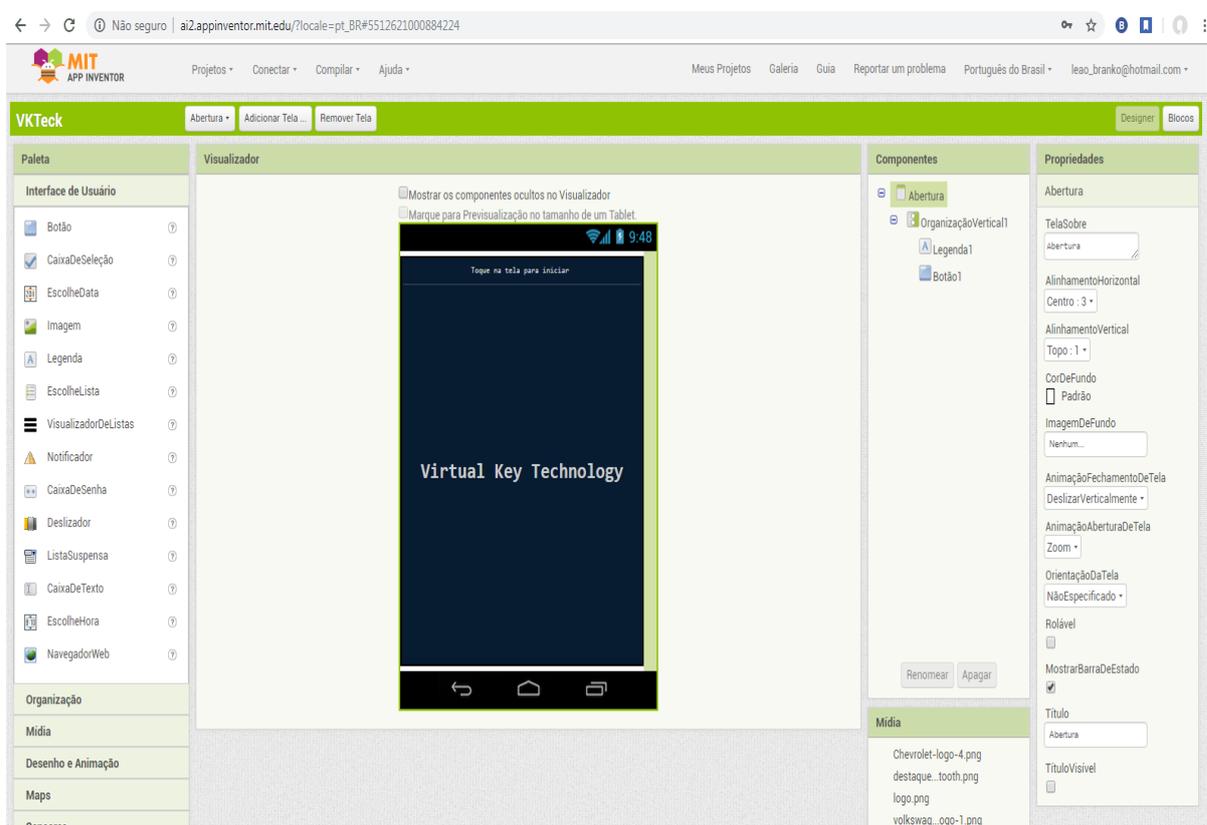


Fonte – Autoria Própria

### 3.2 Desenvolvimento do aplicativo utilizando o App Inventor

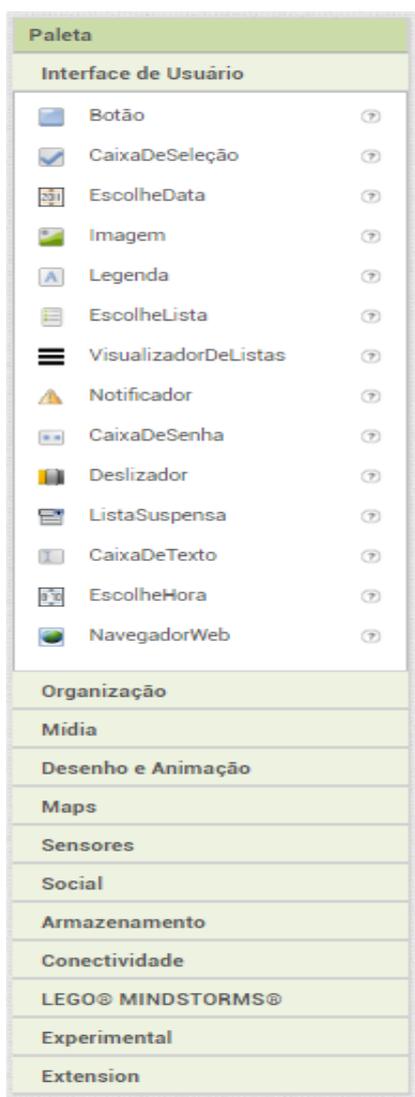
Na Figura 19, temos a página de “Designer” do *App Inventor*, com a tela inicial do nosso aplicativo desenvolvida. É nessa página que podemos criar todo o *layout* do aplicativo a ser desenvolvido. Todas as ferramentas a serem adicionadas (Figura 20), o gerenciamento e a formatação de todos os componentes (Figura 21) inseridos em uma única tela do app podem ser manuseados nessa tela da ferramenta *App Inventor*.

Figura 19 - Página Inicial do Aplicativo - App Inventor



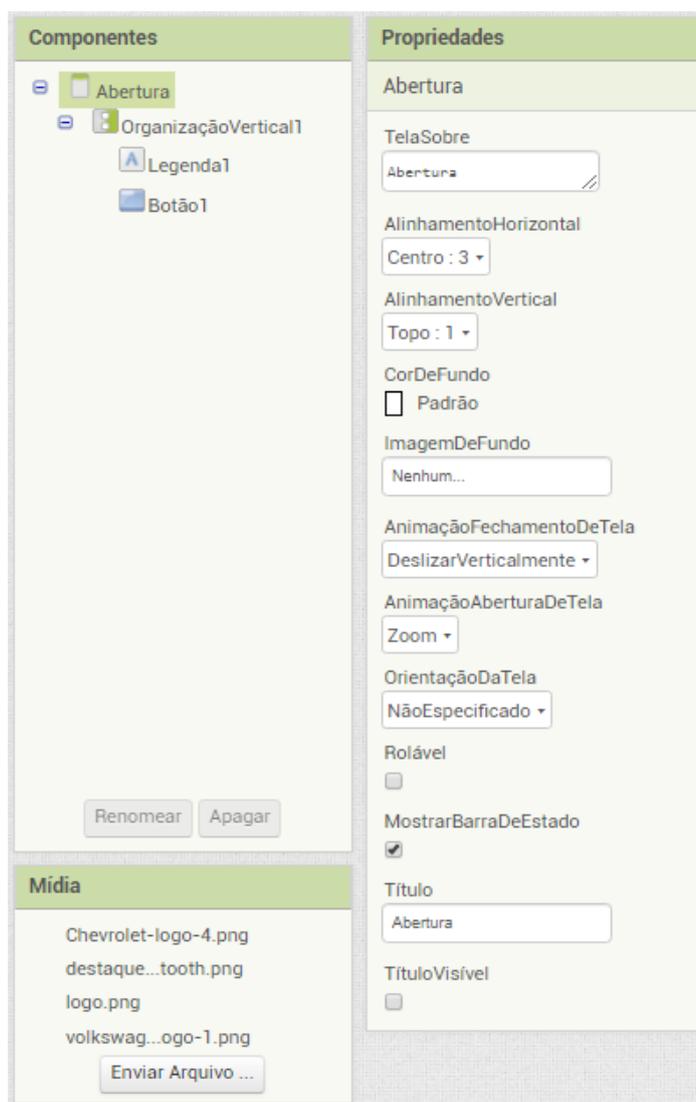
Fonte - <http://ai2.appinventor.mit.edu>

Figura 20 – Ferramentas do App Inventor



Fonte - <http://ai2.appinventor.mit.edu>

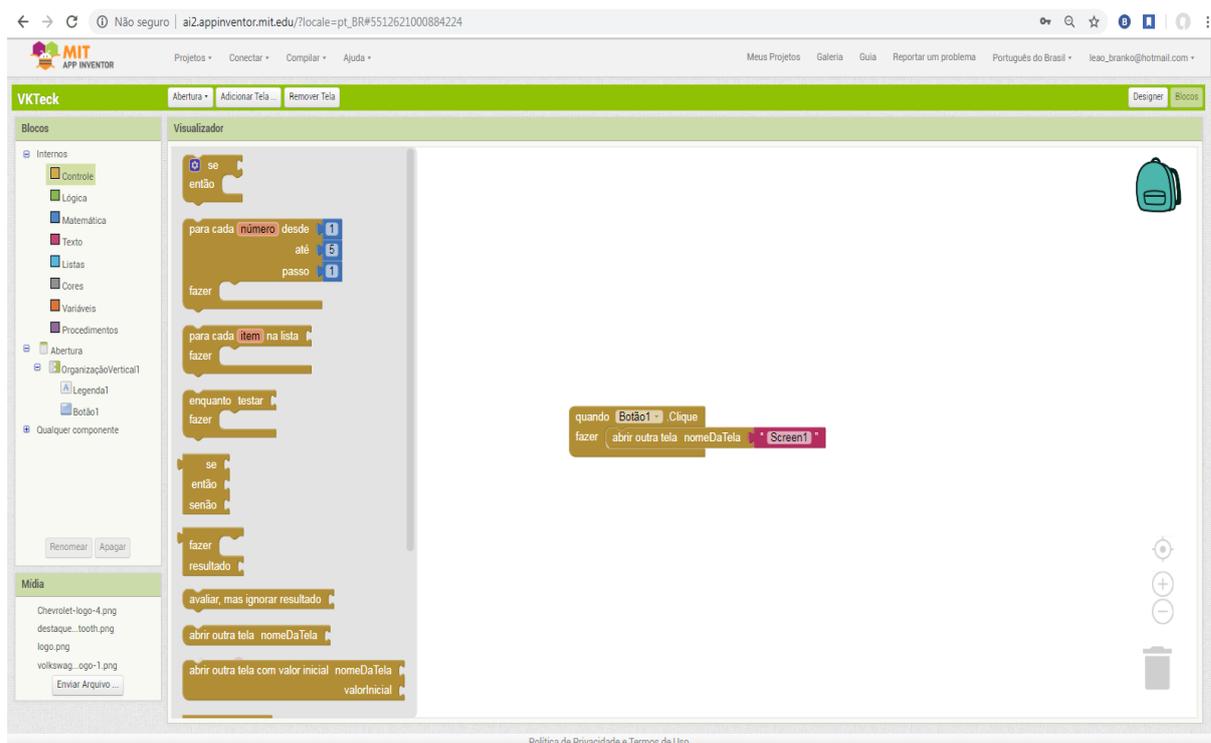
Figura 21 – Formatação/Gerenciamento de componentes App Inventor



Fonte - <http://ai2.appinventor.mit.edu>

Na tela inicial foi criado um botão abrangendo toda a tela do celular e com uma pequena legenda na parte superior indicando o que fazer, ou seja, basta apenas dar um clique na tela do celular para que esta seja alterada para a tela na sequência, denominada "Screen1" então, devido a isso temos uma programação extremamente simples, mostrado na Figura 22.

Figura 22 - Programação Tela Inicial - App Inventor



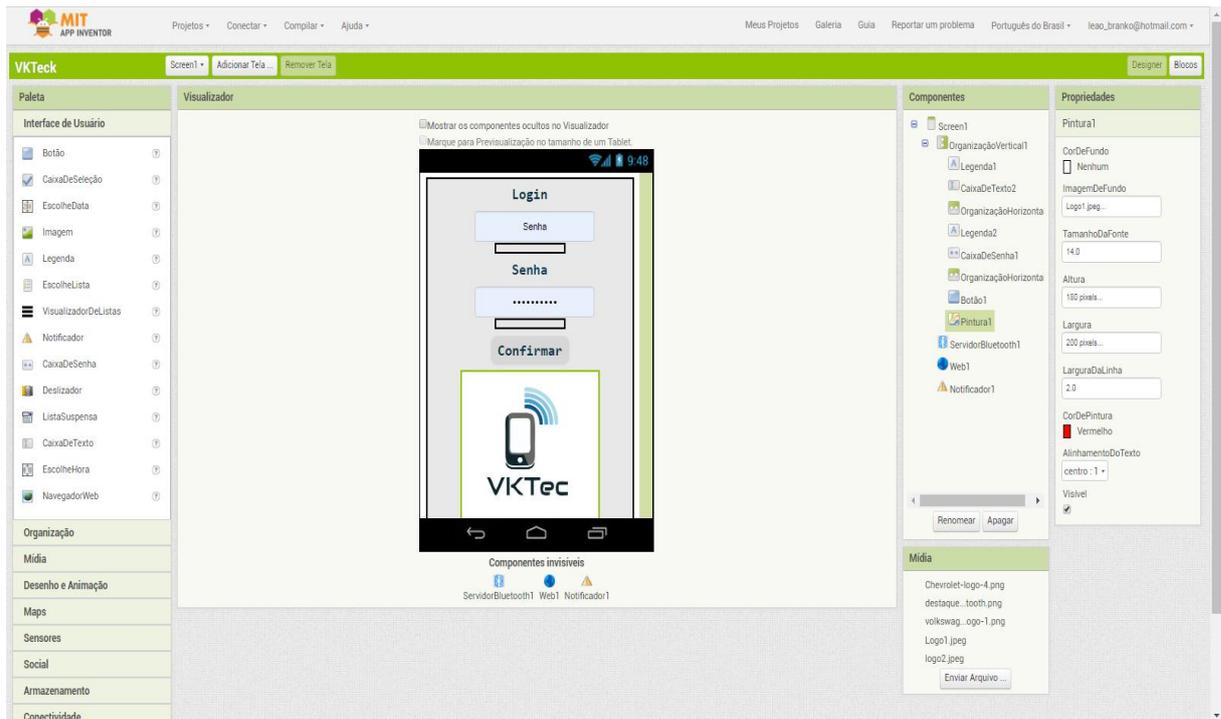
Fonte - <http://ai2.appinventor.mit.edu>

Como podemos ver, quando o “Botão1” for clicado, deverá ser aberto uma outra tela, no caso a tela “Screen1”. Podemos ver na Figura 22 a página “Blocos” do *App Inventor*. É nessa página que se permite desenvolver toda a programação de uma determinada tela do aplicativo. É possível ver também a simples didática utilizada pelo *App Inventor* para auxiliar e facilitar a vida dos usuários que não possuem nenhum conhecimento sobre programação.

Dando um toque na tela, a seguinte será aberta (Figura 23). Nela permite-se inserir um *login* e uma senha determinada pelo programador para se ter acesso aos veículos cadastrados no aplicativo. A caráter de demonstração, o *Login* e a Senha utilizados são, respectivamente, “VKTec” e “tcc2019”. Caso o *Login* ou Senha estiverem errados, o aplicativo mostrará uma mensagem de alerta informando qual dos dois está incorreto, não permitindo seguir para a próxima tela da aplicação.

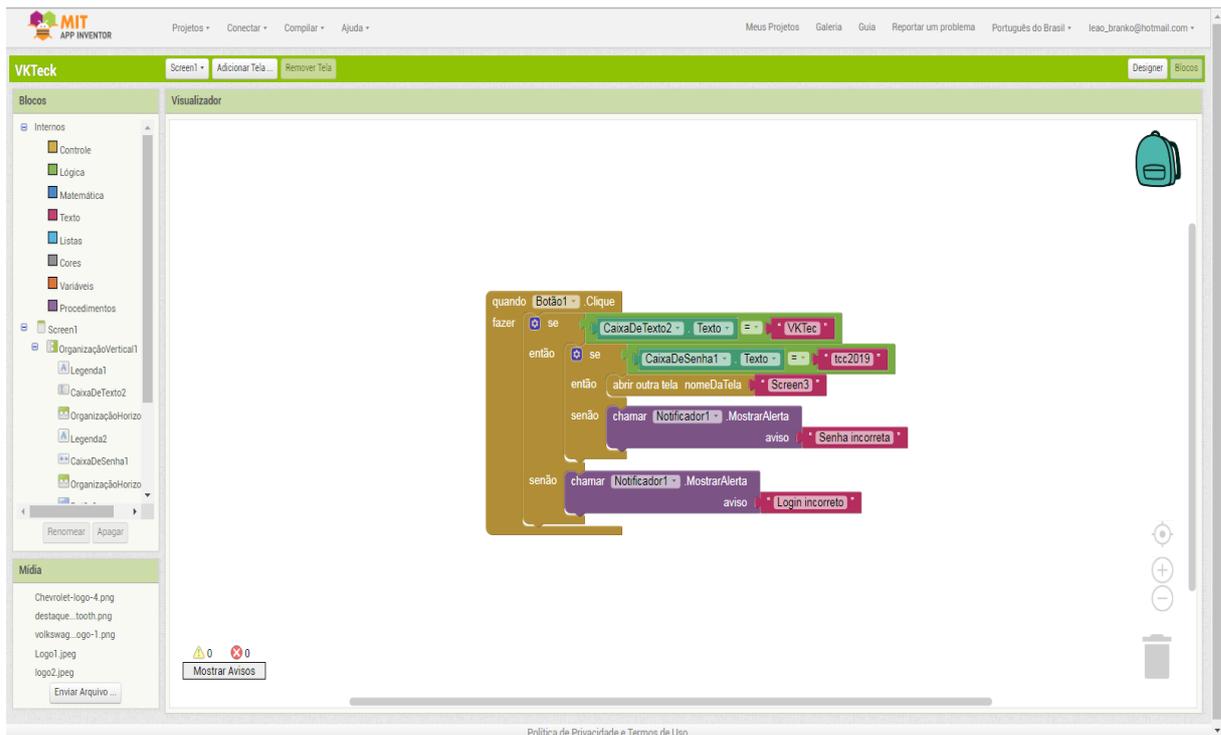
Veremos nas Figuras 23 e 24, as telas de “*designer*” e “*blocos*” com o *layout* e sua respectiva programação utilizada na tela de *login* do aplicativo da VKTec:

Figura 23 - Tela de Login - App Inventor



Fonte - <http://ai2.appinventor.mit.edu>

Figura 24 - Programação Tela Login - App Inventor

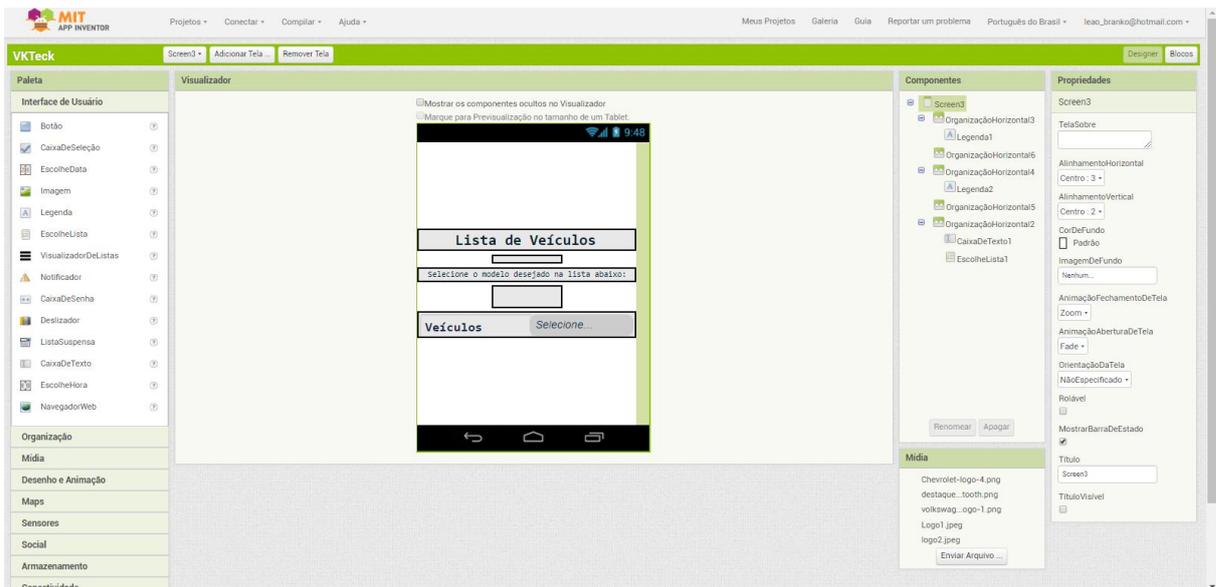


Fonte - <http://ai2.appinventor.mit.edu>

Com o *login* e senha correto, ao clicar no botão “*Confirmar*”, o usuário será direcionado para a próxima tela da aplicação. Nessa tela, haverá a possibilidade de selecionar o veículo desejado na “*Lista de Veículos*”.

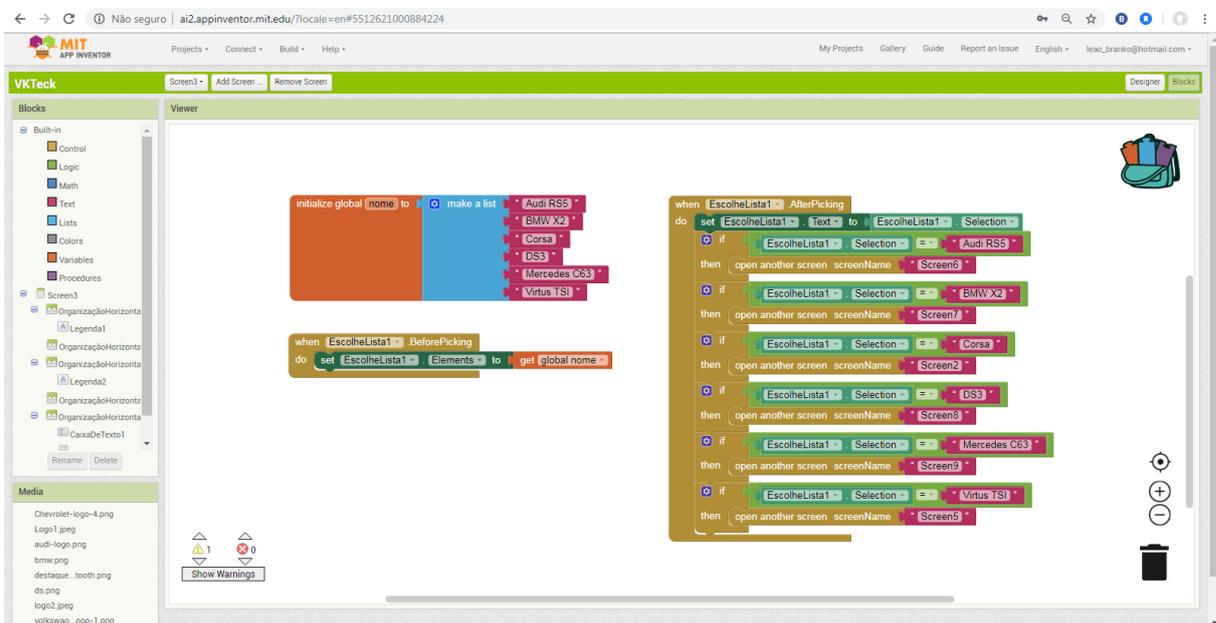
Ao clicar no botão “*Selecione...*”, abrirá uma lista com todos os veículos cadastrados nesse login do aplicativo. Nas Figuras 25 e 26 é ilustrado o *layout* e a programação dessa tela:

Figura 25 - Lista de Veículos - App Inventor



Fonte - <http://ai2.appinventor.mit.edu>

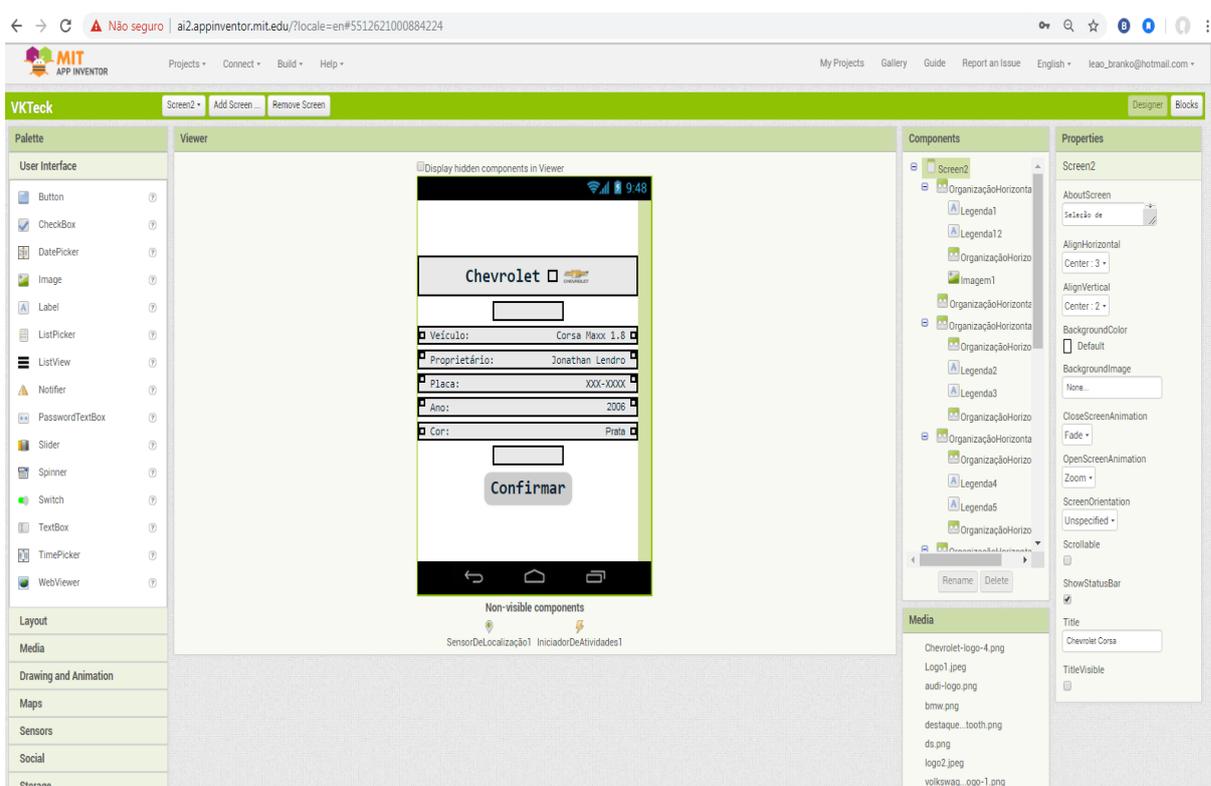
Figura 26 - Programação Lista de Veículos - App Inventor



Fonte - <http://ai2.appinventor.mit.edu>

Ao selecionar um veículo da lista, o usuário será direcionado para a próxima tela, onde consta todos os dados necessários do automóvel selecionado como a placa, a cor do veículo, quem é o proprietário, a montadora (marca do veículo), ano de fabricação e o modelo. Se as informações estiverem corretas confirmando que o automóvel selecionado está correto, basta clicar no botão “*Confirmar*” conforme ilustrado na Figura 27.

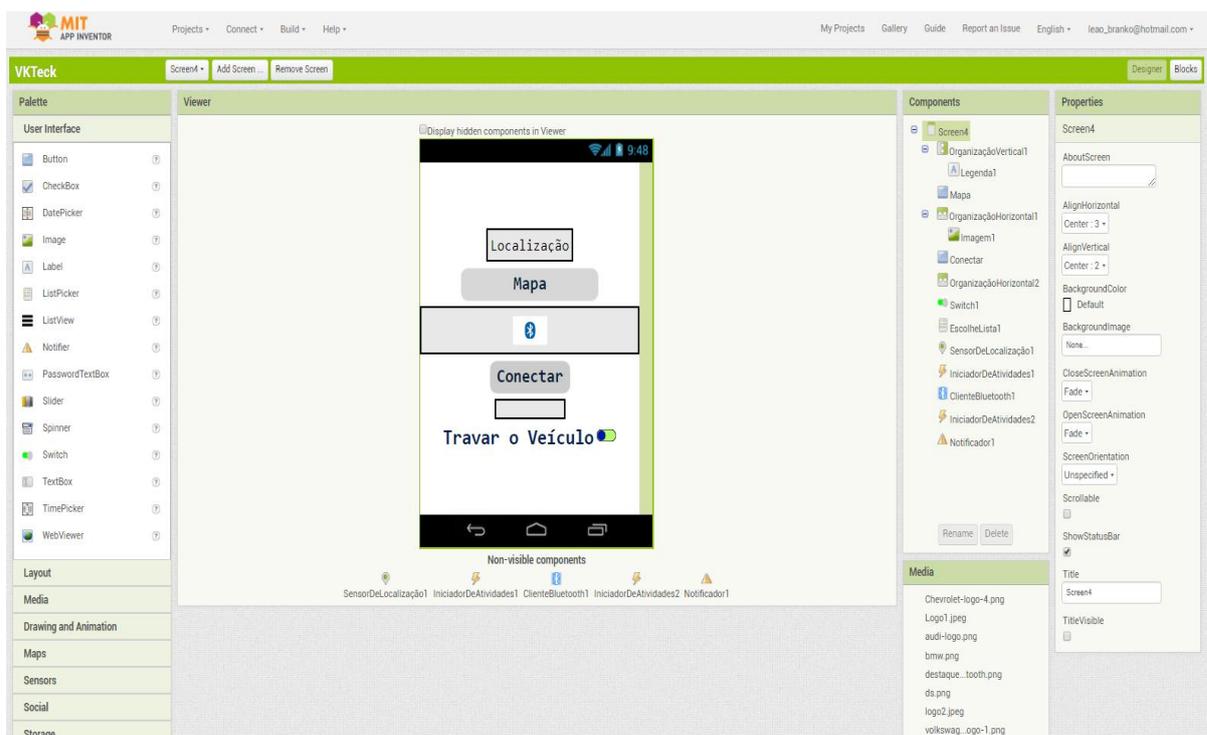
Figura 27 - Tela do veículo selecionado 1



Fonte - <http://ai2.appinventor.mit.edu>

Ao confirmar os dados e o veículo selecionado corretamente, o usuário será direcionado para a próxima tela, ilustrado na Figura 28. Nela, existirá informações sobre a localização atual do veículo selecionado. Aparecerá descrito na tela o endereço completo de onde se encontra o automóvel e a opção de selecionar a localização através de um mapa. Para isso, basta clicar no botão “Mapa”, onde este solicitará que a escolha de um dos aplicativos disponíveis, sendo o *Waze* ou o *Google Maps*.

Figura 28 - Tela de Comunicação Bluetooth e Acionamento da Trava Elétrica



Fonte - <http://ai2.appinventor.mit.edu>

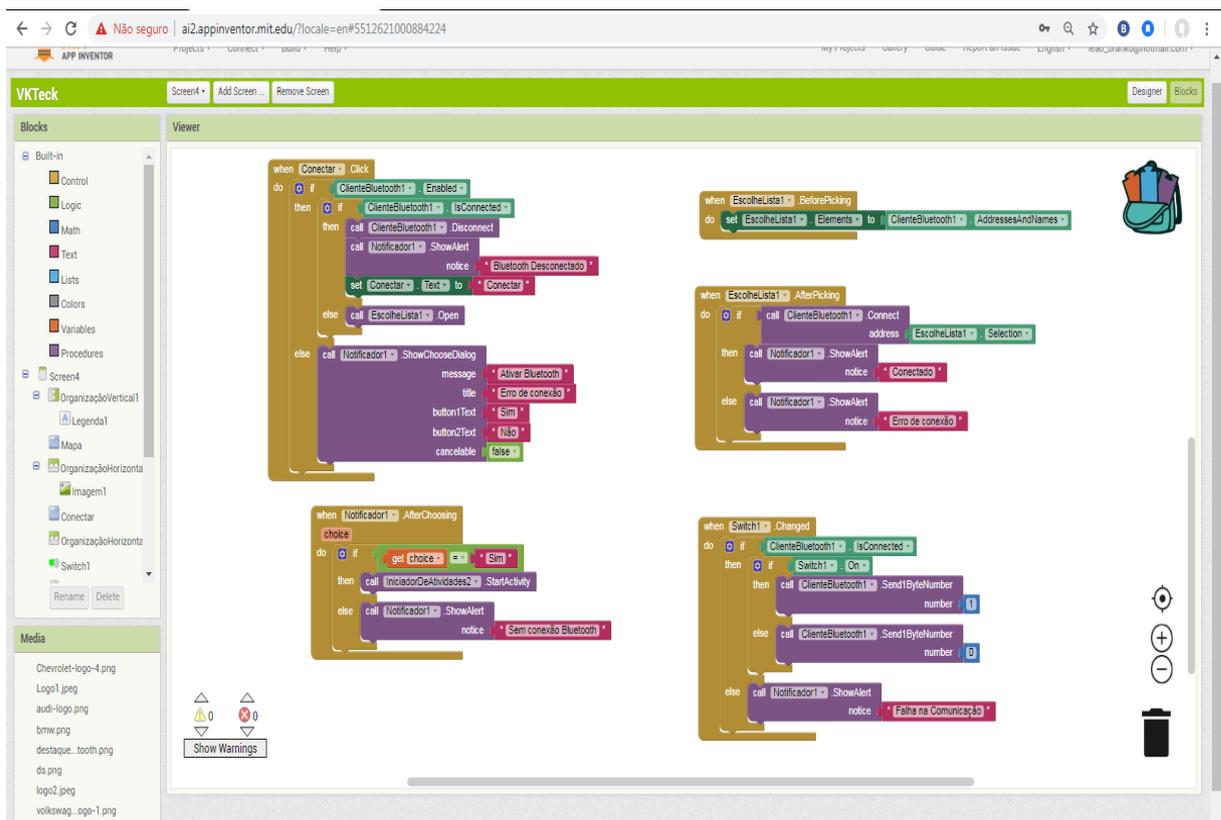
Com a localização informada e o veículo encontrado, basta agora você se conectar ao veículo através do *Bluetooth*. Na mesma tela de localização, terá um botão denominado “*Conectar*”. Ao ser clicado, caso o dispositivo *Bluetooth* do *smartphone* estiver desabilitado, o aplicativo mostrará uma mensagem de “*Erro de conexão*” e perguntará ao usuário se ele deseja ativar o *Bluetooth*. Ao clicar em “*Sim*”, o *smartphone* habilitará o *Bluetooth* automaticamente. Agora, já com o dispositivo *Bluetooth* do celular ativado, o usuário deverá clicar novamente no botão “*Conectar*”. Nesse momento, aparecerá uma lista com todos os dispositivos *Bluetooth* disponíveis para serem pareados com o aplicativo, devendo o usuário selecionar o dispositivo que está denominado com o nome do modelo e a marca do veículo selecionado anteriormente. Feito isso, o *smartphone* será pareado com o veículo desejado. Veremos, na Figura 29 e Figura 30 a programação realizada para que tudo isso funcione perfeitamente:

Figura 29 – Programação GPS



Fonte - <http://ai2.appinventor.mit.edu>

Figura 30 – Programação Comunicação Bluetooth e Acionamento do Relé

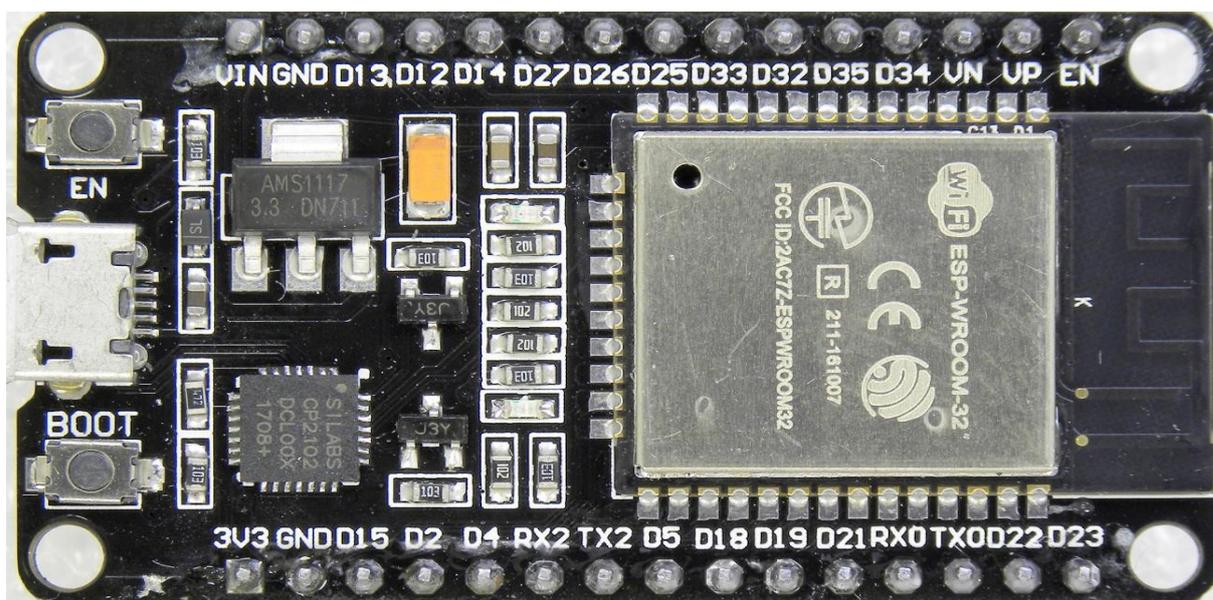


Fonte - <http://ai2.appinventor.mit.edu>

### 3.3 Sistema Embarcado no Veículo.

Para realizar a implementação do sistema, foi utilizada a placa ESP-WROOM-32 pois, em seu *hardware* está disponível a comunicação sem fios *Bluetooth*, diversas portas de E/S, canais A/D, comunicação CAN. Na Figura 31, é possível verificar o tamanho da placa e seus pinos disponíveis.

Figura 31 - ESP-WROOM-32 - Dimensões: 27,5 x 51,0 x 7,0 mm



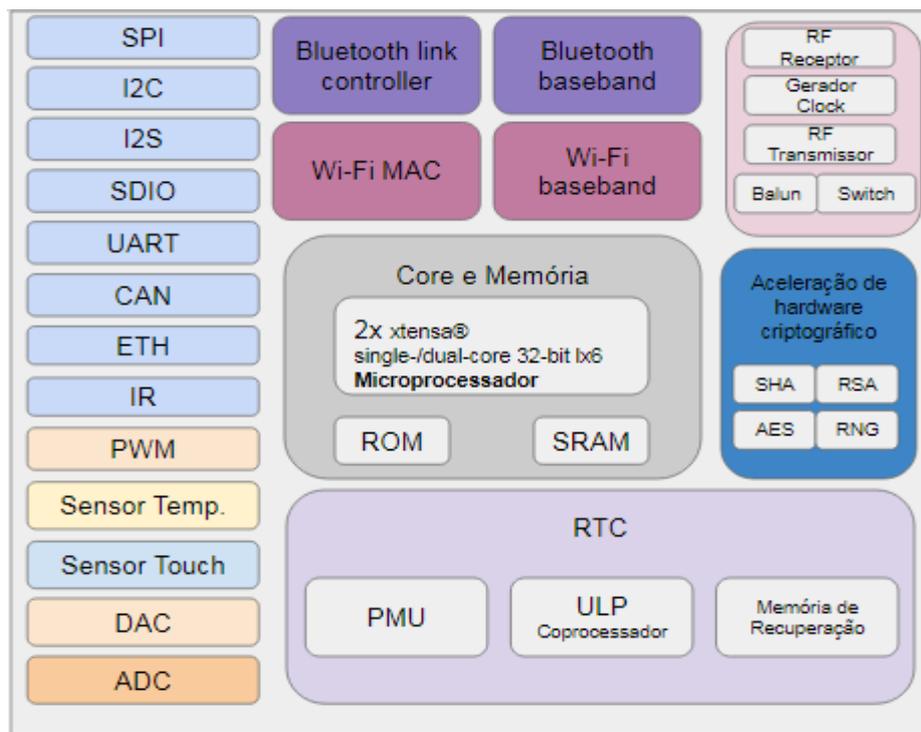
Fonte - <https://www.fernandok.com>

Desenvolvido pela empresa Espressif, o ESP32 apresenta-se como um meio inovador no desenvolvimento de projetos automatizados. Trata-se de um *chip* dotado de diversos recursos. São várias possibilidades de escolha como análogo digital, digital análogo ou ainda se quer trabalhar a porta como digital.

Este pequeno componente apresenta-se como uma opção completa no desenvolvimento de sistemas automatizados, já que dispõe de tantas opções para a interação com projetos sem fio, processamento de informações de forma ajustável, sensores embutidos e múltiplas funções próprias que simplificam a lógica de programação para placas ESP32.

No diagrama ilustrado na Figura 32 consta um breve exemplo sobre a distribuição dos componentes presentes em um ESP32.

Figura 32 - Diagrama dos componentes do ESP32



Fonte - Curto Circuito - <https://www.curtocircuito.com.br>

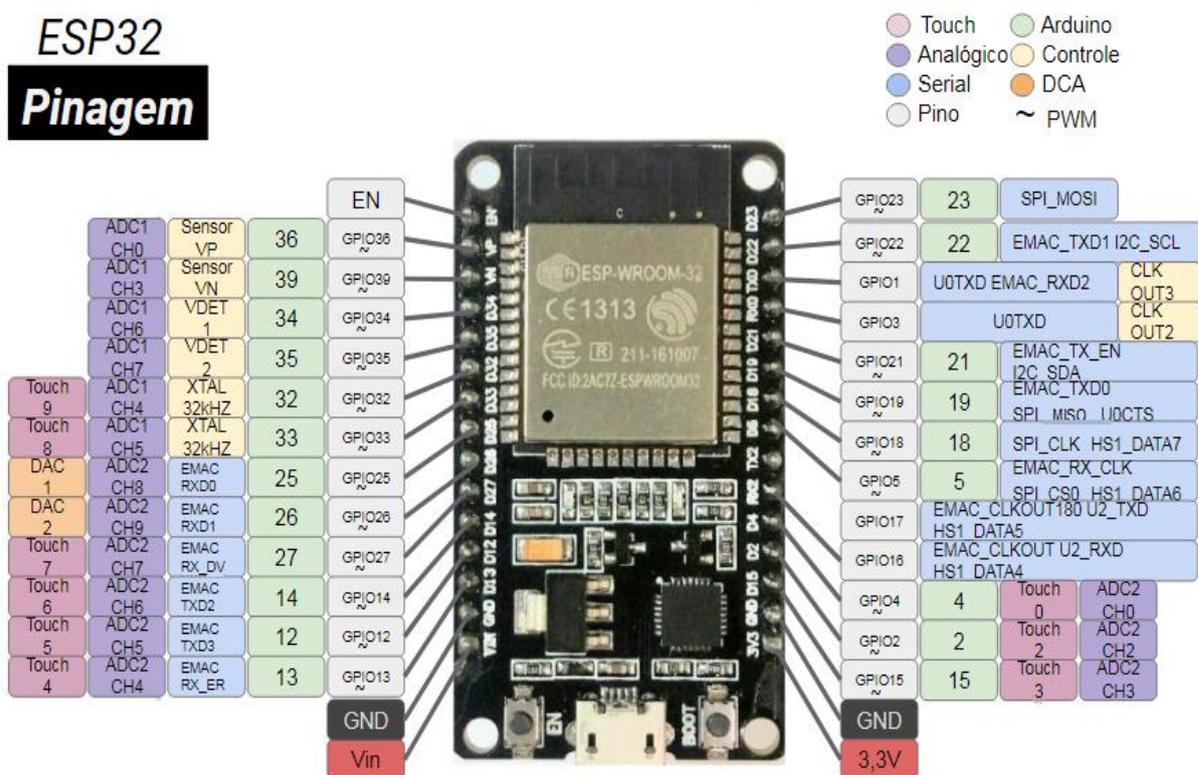
### Especificações:

- CPU: Xtensa® *Dual-Core* 32-bit LX6;
- ROM: 448 *KBytes*
- RAM: 520 *Kbytes*;
- *Flash*: 4 MB;
- *Clock* máximo: 240MHz;
- *Wireless* padrão 802.11 b/g/n;
- Conexão *Wifi* 2.4Ghz (máximo de 150 Mbps);
- Antena embutida;
- Conector *micro-usb*;
- *Wifi Direct* (P2P), *P2P Discovery*, *P2P Group Owner mode* e *P2P Power Management*;
- Modos de operação: STA/AP/STA+AP;
- *Bluetooth* BLE 4.2;
- Portas GPIO: 11;
- GPIO com funções de PWM, I2C, SPI, etc;

- Tensão de operação: 4,5 ~ 9V;
- Taxa de transferência: 110-460800bps;
- Suporta *Upgrade* remoto de *firmware*;
- Conversor analógico digital (ADC);
- Distância entre pinos: 2,54 mm;
- Dimensões: 52 mm x 28 mm x 5 mm (desconsiderando os pinos);

Na Figura 33, veremos a pinagem completa do chip que foi utilizado nesse projeto – 30 pinos.

Figura 33 - Pinagem ESP32



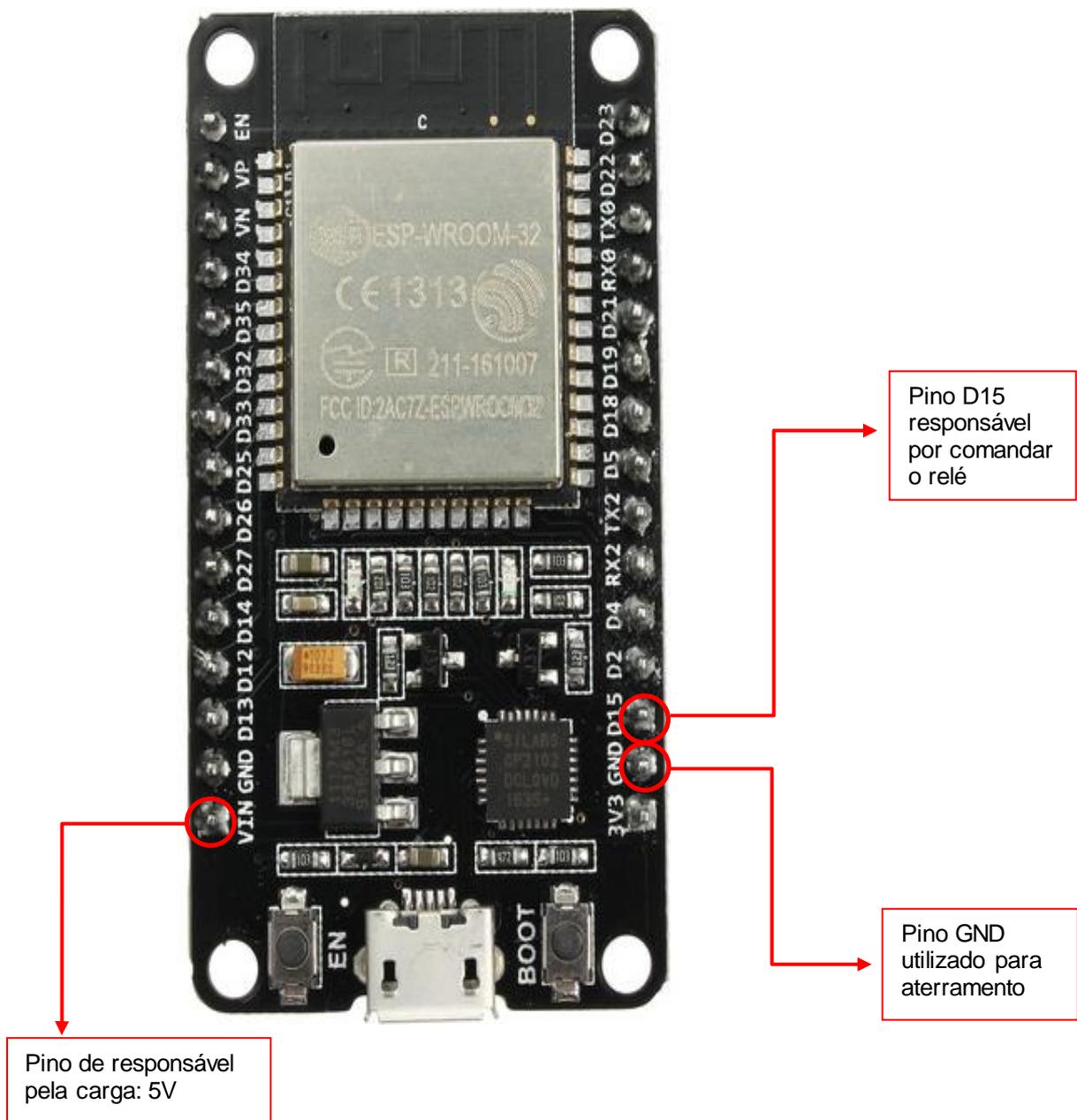
Fonte - Curto Circuito - <https://www.curtocircuito.com.br>

A Figura 33 possui uma legenda bem específica pois, nesse projeto utilizaremos o Arduino IDE como plataforma de programação e a legenda demonstrada na imagem, indica que os valores destacados em verde podem ser utilizados na declaração de pinos em uma programação via Arduino IDE. Exemplificando: Ao ler a entrada analógica de um sensor *Touch*, com o sensor presente no pino GPIO04 (D4), haverá a possibilidade de programador optar por duas opções: Declarar o pino D4 ou *Touch* T0 como INPUT.

Veremos agora na Figura 34, os pinos utilizados para desenvolver o protótipo do projeto VKTec.

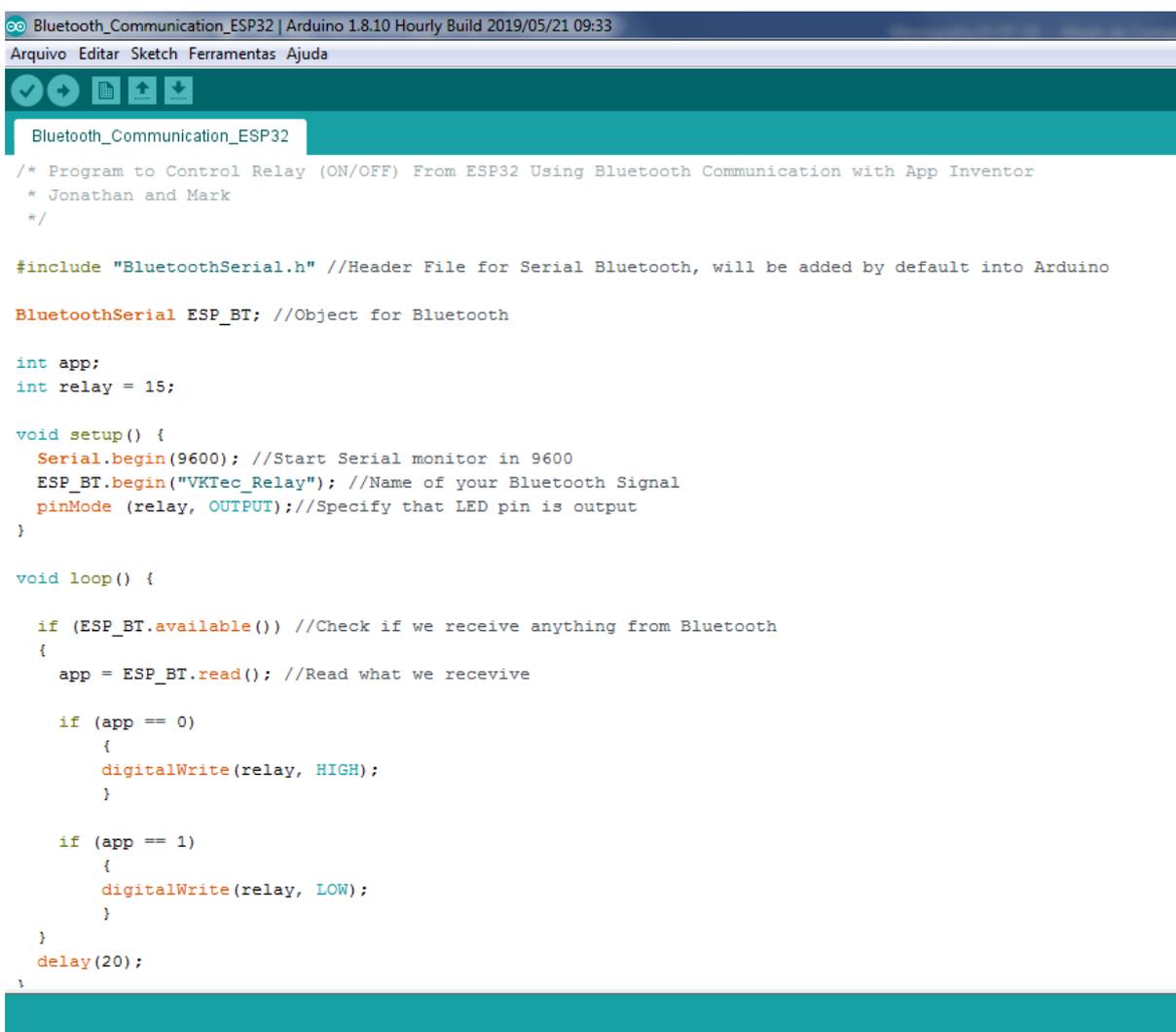
- VIN (5V);
- GND;
- D15 (GPIO15 – Para acionar o relé);

Figura 34 - Pinos utilizados no projeto



Para desenvolver a programação foi utilizado o *framework* Arduino IDE (Plataforma para programação de Arduino), devido à sua facilidade de programação e a possibilidade de iniciar rapidamente o desenvolvimento do projeto. A programação realizada no Arduino IDE está demonstrada na Figura 35 e disponibilizada no Apêndice desse trabalho.

Figura 35 - Programação Arduino IDE



```
Bluetooth_Communication_ESP32 | Arduino 1.8.10 Hourly Build 2019/05/21 09:33
Arquivo  Editar  Sketch  Ferramentas  Ajuda

Bluetooth_Communication_ESP32

/* Program to Control Relay (ON/OFF) From ESP32 Using Bluetooth Communication with App Inventor
 * Jonathan and Mark
 */

#include "BluetoothSerial.h" //Header File for Serial Bluetooth, will be added by default into Arduino

BluetoothSerial ESP_BT; //Object for Bluetooth

int app;
int relay = 15;

void setup() {
  Serial.begin(9600); //Start Serial monitor in 9600
  ESP_BT.begin("VKTec_Relay"); //Name of your Bluetooth Signal
  pinMode (relay, OUTPUT); //Specify that LED pin is output
}

void loop() {

  if (ESP_BT.available()) //Check if we receive anything from Bluetooth
  {
    app = ESP_BT.read(); //Read what we receive

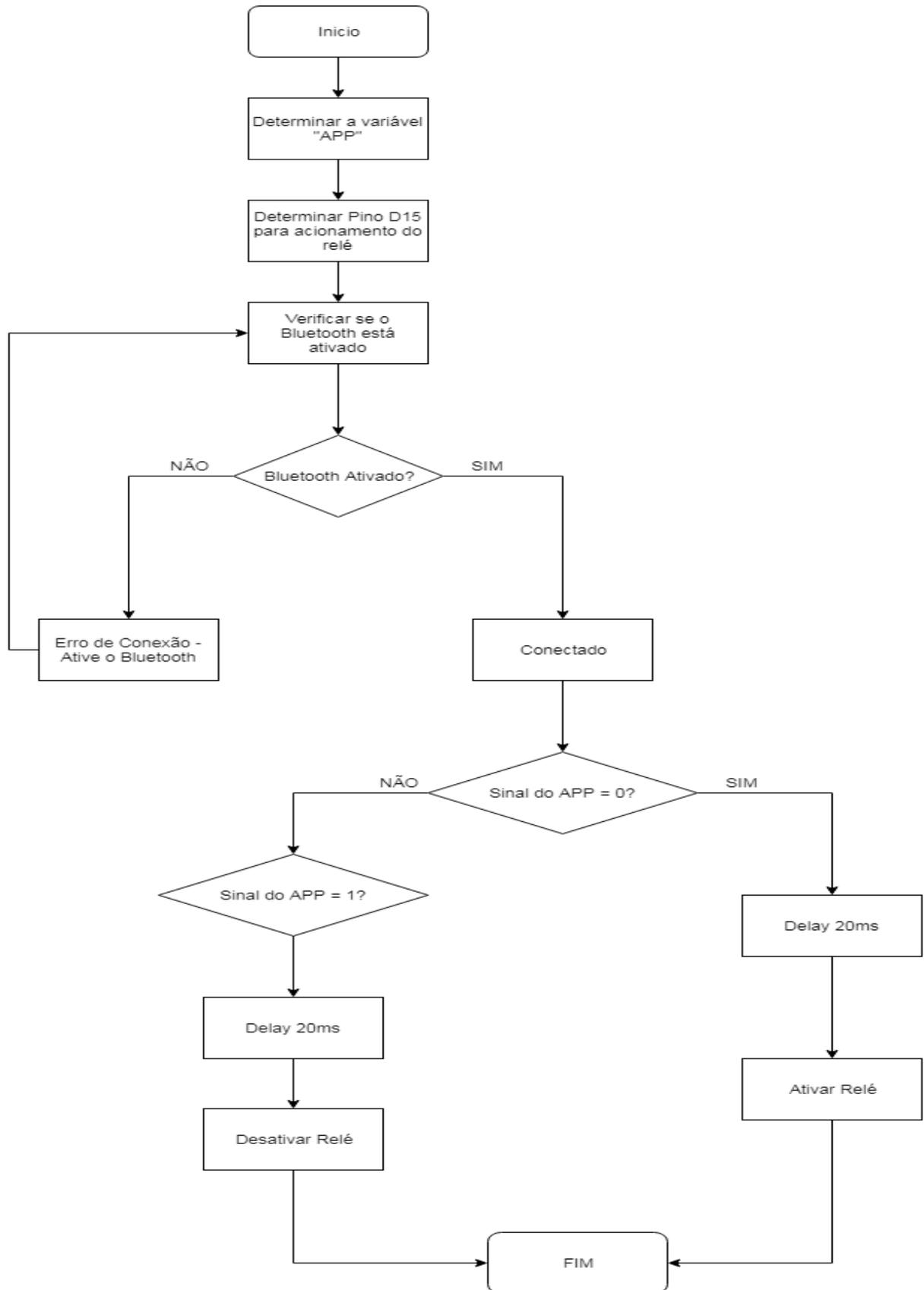
    if (app == 0)
    {
      digitalWrite(relay, HIGH);
    }

    if (app == 1)
    {
      digitalWrite(relay, LOW);
    }
  }
  delay(20);
}
```

Fonte - Autoria Própria

Na Figura 36 está ilustrado através de um fluxograma, o conceito utilizado na programação desenvolvida.

Figura 36 - Fluxograma da programação ES32



Fonte - Autoria Própria

Para o acionamento do relé de testes, o pino D15 do ESP32 foi determinado para realizar esta função.

O nome do sinal do *Bluetooth* que estará disponível foi denominado como “VKTecRelay”, ou seja, quando o aplicativo procurar um sinal disponível para ser pareado, este será o nome visível a ser encontrado.

Ao tentar realizar uma conexão, o *software* verificará se o *Bluetooth* do *Smartphone* estará habilitado. Se não, ocorrerá um erro de conexão e será solicitado para ativar o *Bluetooth*. Se sim, a conexão será realizada.

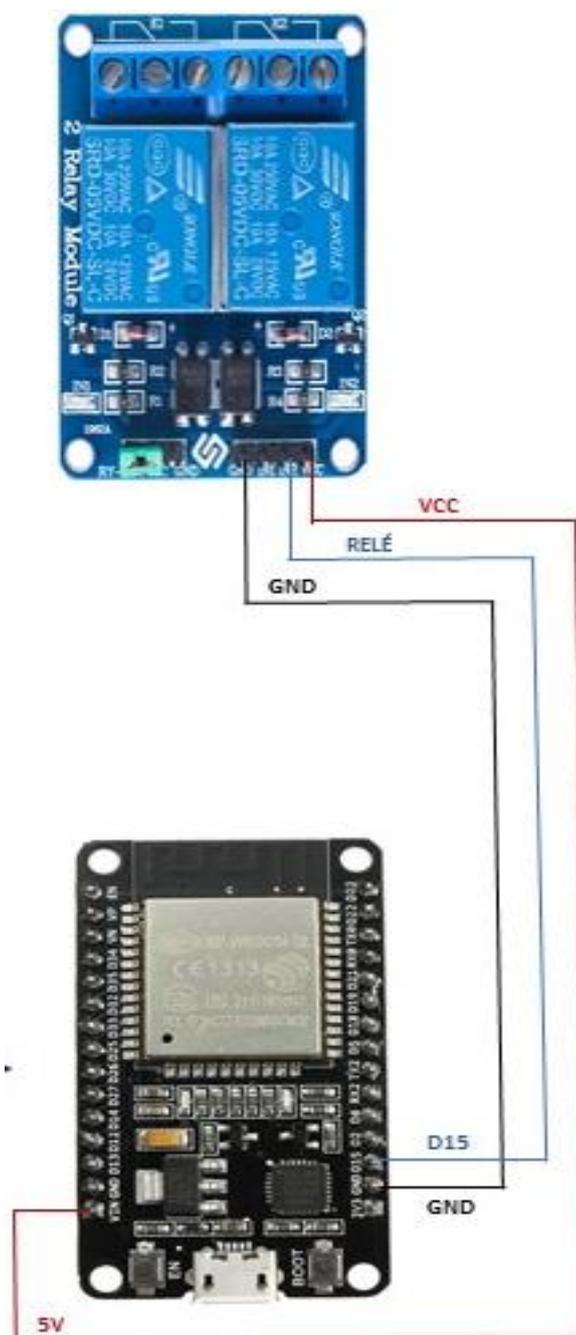
Estabilizado a comunicação, a programação do ESP trabalhará em conjunto com a programação do aplicativo onde, ao ser acionado o *Switch*, o *Bluetooth* enviará um sinal lógico “1” responsável por acionar o relé, travando a porta do veículo. Ao desativarmos o *Switch*, será enviado um sinal lógico “0”, desacionando o relé e travando as portas do automóvel.

### 3.4 Hardware Montado

Para simular o acionamento das travas elétricas do veículo, foi utilizado o módulo relé 5V com dois canais pois serviu como uma alternativa perfeita para nosso projeto, afinal além de ser um módulo compacto, requer baixa tensão para seu acionamento, evitando assim a necessidade de desenvolvimento de um driver para conversão da tensão.

Como a placa ESP32 possui dois pinos de tensão, sendo um com 3.3V e outro com 5V, o desenvolvimento do *hardware* do projeto foi algo bem simples. O único componente necessário eram os fios para que pudéssemos conectar o ESP32 diretamente ao relé, como mostra a Figura 37:

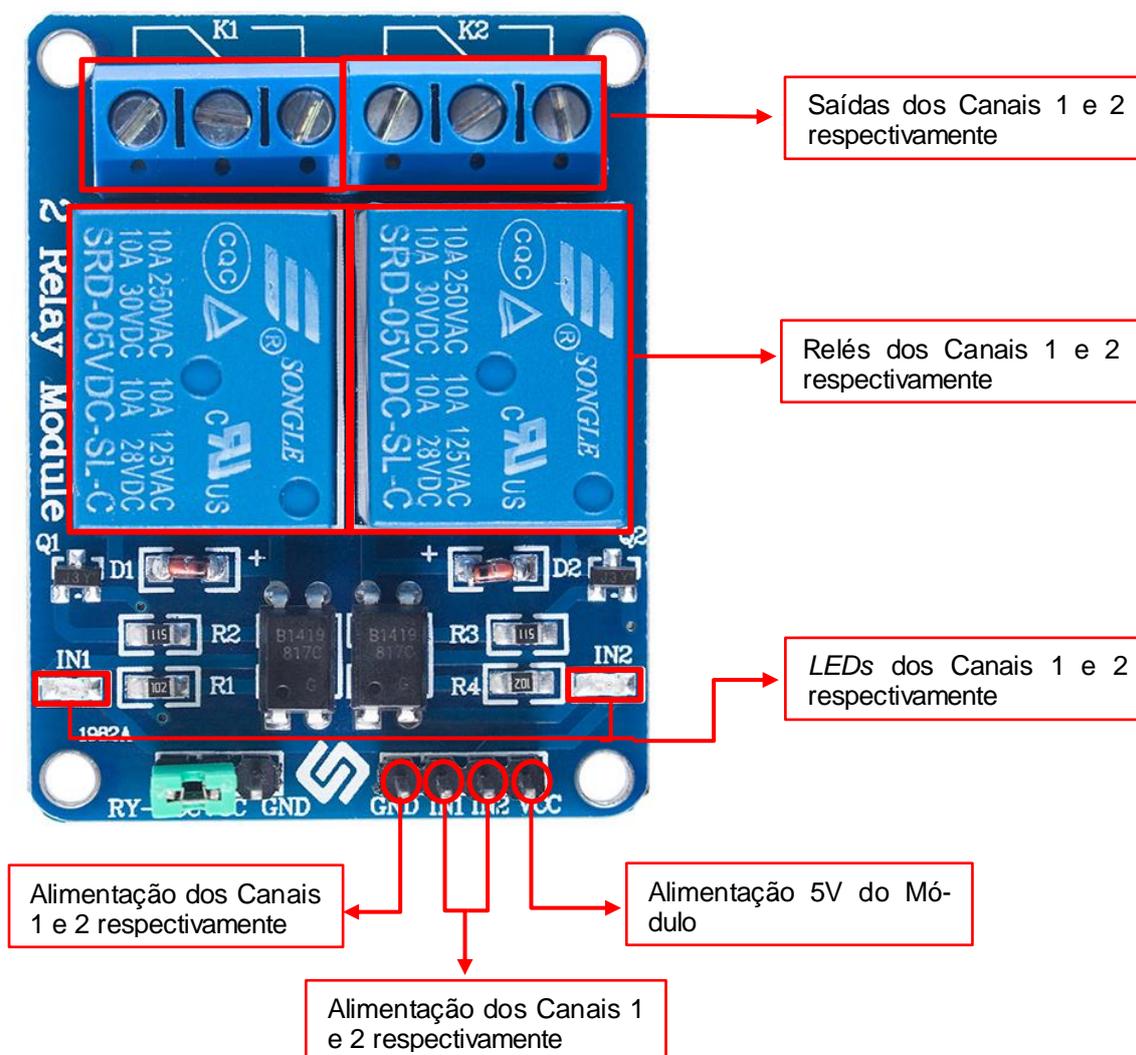
Figura 37 - Acionamento Relé



Fonte – Autoria Própria

Esse módulo permite realizar o acionamento de cargas de 200V AC (Lâmpadas, equipamentos eletrônicos, motores etc.). Esse módulo é constituído por transistores, LEDs, diodos e relés de alta qualidade. Cada canal possui um LED para indicar o estado de saída em que o relé se encontra. Encontra-se ilustrado na Figura 38, o relé utilizado para o projeto.

Figura 38 - Módulo Relé 5V - 2 Canais



Fonte - Autoria Própria

Para realizar as conexões de forma correta, o pino VIN da placa ESP32 deve ser ligado ao pino VCC do relé, assim como os GNDs, permitindo desta forma a alimentação com 5V do módulo. Durante a programação, o pino D15 do ESP32 foi selecionado para enviar os sinais responsáveis para acionar o relé, então deve-se realizar a conexão entre este pino e um dos pinos de alimentação dos canais do relé (IN1 ou IN2) mostrado na Figura 38. Desta forma, o *hardware* está montado e pronto para ser utilizado.

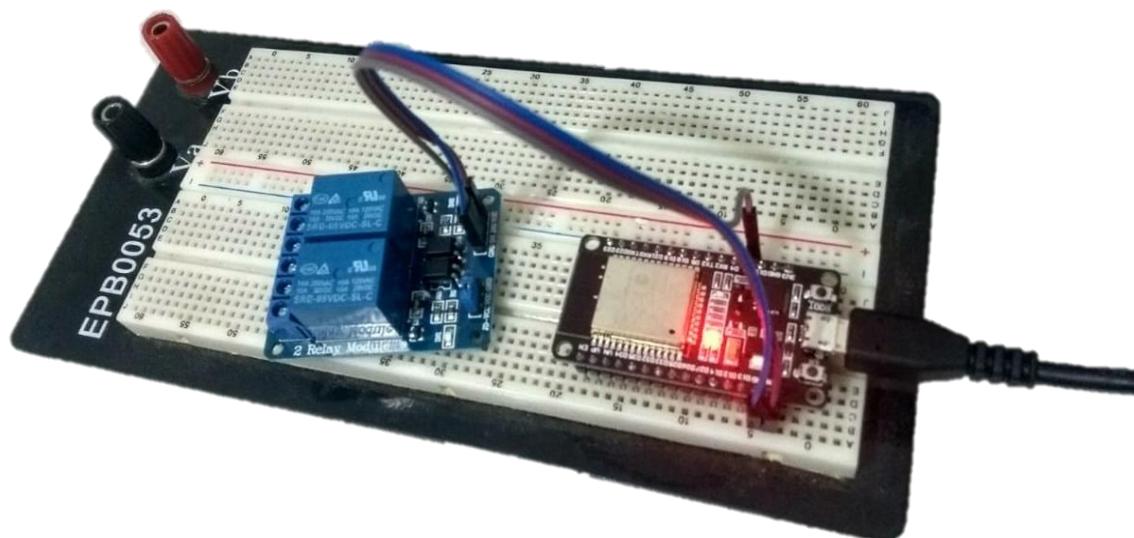
Especificações Módulo Relé 5V:

- Modelo: SRD-05VDC-SL-C

- Tensão de operação: 5VDC
- Permite controlar cargas de 220V AC
- Corrente típica de operação: 15~20mA
- *LED* indicador de status
- Pinagem: Normal Aberto, Normal Fechado e Comum
- Tensão de saída: (30 VDC a 10A) ou (250VAC a 10A)
- Furos de 3mm para fixação nas extremidades da placa
- Tempo de resposta: 5~10ms
- Dimensões: 51 x 38 x 20mm
- Peso: 30g

### 3.5 Sistema Completo

Figura 39 - Sistema Completo



Fonte - Autoria Própria

Através da Figura 39, é possível observar o projeto funcionando na prática. Nesta demonstração, o *hardware* está montado em um *protoboard* para proporcionar uma melhor qualidade dos testes. Nesta figura, vemos o ESP32 alimentado através

do conector USC-C o qual é indicado pelo *LED* vermelho. Também permite-se ver as seguintes conexões:

Tabela 2 - Conexões ESP32 e Relé

ESP32		RELÉ
VIN (5V)		VCC
D15		IN1 (Canal 1)
GND		GND

Fonte - Autoria Própria

## 4 RESULTADOS

O trabalho foi desenvolvido com o intuito de realizar e viabilizar a comunicação virtual entre *smartphone* e automóvel. Após diversos obstáculos atravessados vimos como um resultado que sim, é possível realizar essa comunicação.

A princípio, iríamos utilizar a ferramenta de desenvolvimento de aplicativos *Android Studio*. Podemos dizer que é uma ferramenta semiprofissional, porém requer do usuário um alto conhecimento em programação, especialmente em Java, e foi onde esbarramos no primeiro obstáculo pois era algo que não contávamos no elenco do trabalho. Com isso, pesquisamos alternativas possíveis para desenvolvermos um App. Lendo um trabalho de conclusão de curso, o *Sistema On Board de Monitoramento de Frotas* que inclusive foi realizado por alunos da mesma instituição que estudamos, a Fatec Santo André, notamos que foi desenvolvido um aplicativo pelo *App Inventor* e, após pesquisarmos sobre ele, descobrimos que era sim possível desenvolver o nosso projeto através dessa ferramenta, e assim foi. O aplicativo foi desenvolvido com relativa facilidade e tudo indicava que funcionaria perfeitamente com a montagem eletrônica.

Finalizado o *layout* do aplicativo, partimos para o desenvolvimento da eletrônica e programação. E nesse momento foi onde encontramos o nosso maior obstáculo, afinal não dominamos a habilidade de programador e nossos conhecimentos a respeito são muito escassos, além de estarmos trabalhando com um microcontrolador que não conhecíamos. Após muitas pesquisas e com uma grande ajuda de nosso coorientador de trabalho, descobrimos formas mais fáceis de realizar a programação do ESP32, como no caso do Arduino IDE.

Com a programação desenvolvida, era hora de compilarmos o programa no micro para que ele possa compreender os comandos provenientes do aplicativo realizando corretamente sua função, acionando as travas elétricas do veículo. Após realizada a verificação da programação no Arduino IDE sem nenhum imprevisto, demos início a compilação do programa no micro e então tivemos um problema. Durante a compilação, após alguns minutos de espera, éramos notificados com um erro descrito na Figura 40, ocasionando na não gravação do programa.

Figura 40 - Erro ao Compilar a Programação

```
A fatal error occurred: Failed to connect to ESP32: Timed out waiting for packet header
A fatal error occurred: Failed to connect to ESP32: Timed out waiting for packet header
A fatal error occurred: Failed to connect to ESP32: Timed out waiting for packet header
```

Fonte - Autoria Própria

Após algumas pesquisas realizadas para tentarmos solucionar esse erro, descobrimos que, no ESP32 existe um “segredo” para fornecer a disponibilidade de gravações de programas. Na placa do ESP32 existe dois botões, o “EN” e o “BOOT” reportado na Figura 41 que, no caso da marca responsável por desenvolver a placa que utilizamos, optou por localizá-los na parte inferior do componente.

O botão BOOT é responsável por permitir a gravação de programações no ESP32, funcionando como uma espécie de porta. Quando tentávamos gravar um programa e gerava o erro mencionando anteriormente era porque a porta estava fechada. Então para realizarmos a gravação do programa, bastava abrir essa porta apertando o botão Boot quando o programa estava sendo carregado.

Feito isso, a programação foi concluída com sucesso.

Figura 41 - Botão Boot



BOTÃO BOOT – Responsável por permitir a gravação de programas no microcontrolador

Fonte - <https://www.fernandok.com> + Edições Particulares

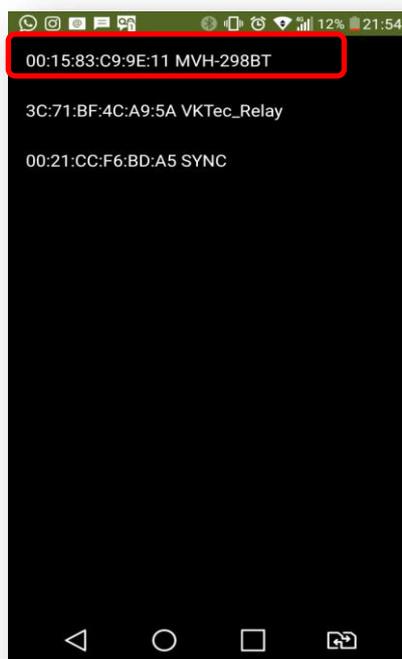
Depois de gravado o programa no micro, partimos para o momento de realizar a montagem das ligações elétricas corretas entre o ESP32 e o relé. Como não conhecíamos esta placa antes de realizarmos esse projeto e pelo que havíamos pesquisado até então, achávamos que ela continha apenas um único pino de alimentação contendo 3.3V e nosso relé requisitava 5V então, após conversar com alguns professores vimos que seria necessário a construção de um *driver* contendo um *transistor* pra realizar chaveamento. Porém, após realizarmos pesquisas mais a fundo sobre o ESP32, descobrimos que ela possuía um pino chamado “VIN” onde ele é responsável por fornecer uma tensão de 5V, o que para a nossa simulação facilitaria muito afinal bastaria apenas realizar a ligação direta dos fios entre o ESP32 e o relé. Porém, vale ressaltar que para o funcionamento do projeto no veículo, deverá ser desenvolvido esse *driver* pois o relé responsável pelas travas elétricas é de 12V.

Com as conexões elétricas corretamente montadas, demos início ao primeiro teste junto ao aplicativo. Ao chegarmos na tela responsável por realizar o pareamento *Bluetooth* e pesquisarmos os dispositivos disponíveis, sinal *Bluetooth* da ESP32 não constava na lista de dispositivos encontrados. Testamos também o teste através da busca própria do *smartphone* e não encontramos. Alteramos o nome do sinal determinado durante a programação para “*VKTec\_Relay*” (Antes estava como *ESP32\_Relay*), fizemos uma busca novamente pelo celular e então foi localizado. Realizamos então o pareamento do *smartphone* com a ESP32 e aí sim ele se tornou disponível na lista fornecida pelo aplicativo, como vemos na Figura 42.

Com o Bluetooth devidamente ativado, ao selecionarmos o sinal do VKTec, o ESP32 será pareado com o *Smartphone* permitindo então, ser enviado o Nível Lógico “1” para travar o veículo quando o *Switch* for ativado e Nível Lógico “0” para destravar o veículo com o *Switch* for desativado.

E foi assim que conseguimos colocar o nosso protótipo em prática demonstrando a possibilidade da substituição das chaves mecânicas do veículo por chaves virtuais.

Figura 42 - Sinal Bluetooth da ESP32 disponível



Fonte - Autoria Própria

Veremos na Tabela 3, os resultados alcançados ao desenvolvermos o projeto até sua apresentação final:

Tabela 3 - Resultados Esperados

RESULTADOS OBTIDOS	
Desenvolver um Logotipo	✓
Desenvolver um Aplicativo	✓
Programar ESP32 com Arduino IDE	✓
App reconhecer sinal Bluetooth do ESP32	✓
Estabelecer comunicação entre App e ESP32	✓
ESP32 acionar o relé com acionamento do APP	✓

Fonte – Autoria Própria

#### 4.1 Custos do Protótipo

Devido o projeto ainda ser um protótipo e os componentes iniciais tenham sido utilizados para testes, o custo monetário do Hardware desenvolvido foi consideravelmente baixo, porém um dos intuitos do projeto é não ser apenas eficiente, mas também acessível a todas as classes sociais. Na Tabela 4 é informado os itens utilizados e seus respectivos gastos. Também é informado os gastos com a impressão deste trabalho e o Banner para apresentação.

Tabela 4 - Custos do Protótipo

<b>Componente</b>	<b>Quantidade</b>	<b>Valor Unitário</b>
<b>ESP32</b>	<b>1</b>	<b>R\$ 35,70</b>
<b>Relé 5V 2 Canais</b>	<b>1</b>	<b>R\$ 10,00</b>
<b>Kit Jumper Fêmea - 40 pçs</b>	<b>1</b>	<b>R\$ 9,00</b>
<b>Total</b>		<b>R\$ 54,70</b>
<b>Monografia</b>	<b>3</b>	<b>R\$ 30,00</b>
<b>Banner</b>	<b>1</b>	<b>R\$ 50,00</b>
<b>Total Geral</b>		<b>R\$ 194,70</b>

Fonte - <https://www.curtocircuito.com.br> - Custos referentes ao dia 16/07/2019

## 5 CONCLUSÕES

### 5.1 Conclusões Finais

Este projeto foi desenvolvido com o intuito de aproximar ainda mais o ser humano das tecnologias e do mundo virtual, fazendo jus a expressão “homem-máquina”. Através dele, vimos que com o avanço da ciência, abre-se cada vez mais espaços para novas ideias e novos projetos.

Através desse trabalho, desenvolvemos um protótipo capaz de dar um passo para substituir mais um objeto físico de nosso cotidiano – a chave – transformando-a em algo virtual. Isso tem se tornado cada vez mais comum em nossas vidas. Documentos estão sendo transcritos para o mundo virtual, temos exemplos como a Carteira Nacional de habilitação (CNH), Título de Eleitor, o Cadastro de Pessoa Física (CPF) e o Registro Geral (RG) que são importantíssimos em nossas vidas e que já estão sendo utilizados e consultados através dos *Smartphones*. Até mesmo acessamos nossa conta bancária, pagamos contas, fazemos transferências financeiras através de um celular, o que prova que sim, essa ferramenta pode ser útil e confiável para mais uma nova função.

Vimos que utilizando as ferramentas e componentes corretos, é possível desenvolver um projeto extremamente eficiente.

Se tratando de um protótipo, utilizamos a ferramenta online e gratuita do MIT em parceria com o Google, o *App Inventor*, para desenvolver o aplicativo. Embora seja uma ferramenta simples, é muito didática, o que nos auxiliou muito para desenvolver o aplicativo com um *layout* interativo e com a identidade visual do projeto, cumprindo muito bem com as exigências necessárias e sendo extremamente eficaz para realizar as funções requisitadas no trabalho.

O cérebro do projeto, o responsável por interagir com o veículo, foi o ESP32. Esse microcontrolador nos surpreendeu muito durante o desenvolver do projeto, pela sua qualidade e capacidade. Além de permitir a comunicação sem fio via *Bluetooth Low Energy* (BLE) que é a essência de nosso trabalho, permite também a utilização de *Wifi*, dando a oportunidade de desenvolver novos aprimoramentos para o projeto como até mesmo, comando com controle de voz. No começo, tivemos dificuldade com o desenvolvimento da programação do ESP32, pois ele tem a “*Lua Script*” como linguagem principal de programação, e nós não tínhamos nenhum co-

nhecimento a respeito dessa linguagem. Após pesquisas, descobrimos que era possível emular, realizando uma programação em C/C++ através do Arduino IDE (Plataforma de programação em Arduino) e gravar no ESP32. Isso nos ajudou, fazendo com que desenvolvêssemos com maior facilidade a programação do projeto. Tivemos apenas uma dificuldade ao gravar o programa pois não sabíamos que, para gravar no ESP32, se faz necessário pressionar o botão “Boot” presente na placa, que funciona como uma porta onde, ao ser pressionado, permite a gravação da programação desenvolvida. Essa placa se encaixou como uma luva no projeto, pois, além de suas diversas funções condizentes ao que procurávamos, ela é pequena e compacta, permitindo que em uma futura instalação no veículo, venha ocupar pouco espaço.

Por fim, para simularmos o acionamento das travas elétricas do veículo, utilizamos um módulo relé de 5V. A princípio, achamos que seria necessário desenvolver um driver, contendo transistor, resistor e diodo para acioná-lo, afinal, o relé requer uma tensão de 5V e o ESP32 possui um *Port* com 3.3V. Porém descobrimos que a placa possuía um *Port* denominado “VIN” onde dele fornecia os 5V direto, eliminando o acréscimo de componentes, tornando o projeto físico mais enxuto. Então realizamos as ligações corretas e testamos todo o conjunto. Porém, vale ressaltar que, para a instalação em um veículo, deverá ser desenvolvido um *driver* pois o relé presente nos automóveis possui tensão de 12V. A interação entre o aplicativo no smartphone, o ESP32 e o relé casou perfeitamente, sendo acionado ou desacionado sempre pelo comando do celular, ilustrando para nós que mais um objeto físico extremamente antigo como a chave, pode ser substituído por uma tecnologia virtual.

Este projeto possui um potencial gigantesco para futuros aprimoramentos e para novas propostas. Temos o intuito de futuramente inserir novas tecnologias como acionamento das travas simplesmente com a proximidade do *smartphone* e integrar o GPS, permitindo informar ao usuário a localização do veículo através dos aplicativos populares como *Waze* e *Maps*. Queremos também substituir as chaves por completo, não somente para acionamento das travas elétricas, mas também para permitir a ignição do motor do veículo apenas com a presença do celular (Em veículos que contenham o sistema “*Star-Stop*”). Tudo isso protegido com um forte sistema de segurança virtual de criptografia assimétrica, evitando hackeando do siste-

ma e possíveis furtos do veículo, fornecendo qualidade, confiança e conforto ao usuário.

## 5.2 Propostas Futuras

Esse trabalho possui um grande potencial de aprimoramentos e novos desenvolvimentos. Abaixo, listaremos algumas propostas que devemos implementar posteriormente em nosso projeto, deixando-o ainda mais completo e eficaz.

### Segurança Virtual

Requisito fundamental para a realização desse projeto é a segurança. Se tratando de comunicação virtual, a possibilidade de *hackers* invadirem o sistema para aquisição dos dados que são compartilhados é extremamente relevante. Ainda mais quando coloca em risco um bem muito valioso, como o carro. Bastaria um *hacker* ouvir a senha digital compartilhada, copiá-la, e usar para furtar o veículo. Então, pensando nisso, com o intuito de promover a maior segurança possível, usaremos o método de encriptação assimétrica, proporcionando máxima proteção no compartilhamento dos dados entre o *smartphone* e o veículo. Esse método é extremamente funcional, seguro e de confiança. Para se ter noção de sua confiabilidade, é esta forma de encriptação que se utiliza em bancos para proteger os dados de clientes no momento de troca de dados. Iremos explicar o seu funcionamento a seguir.

### Criptografia

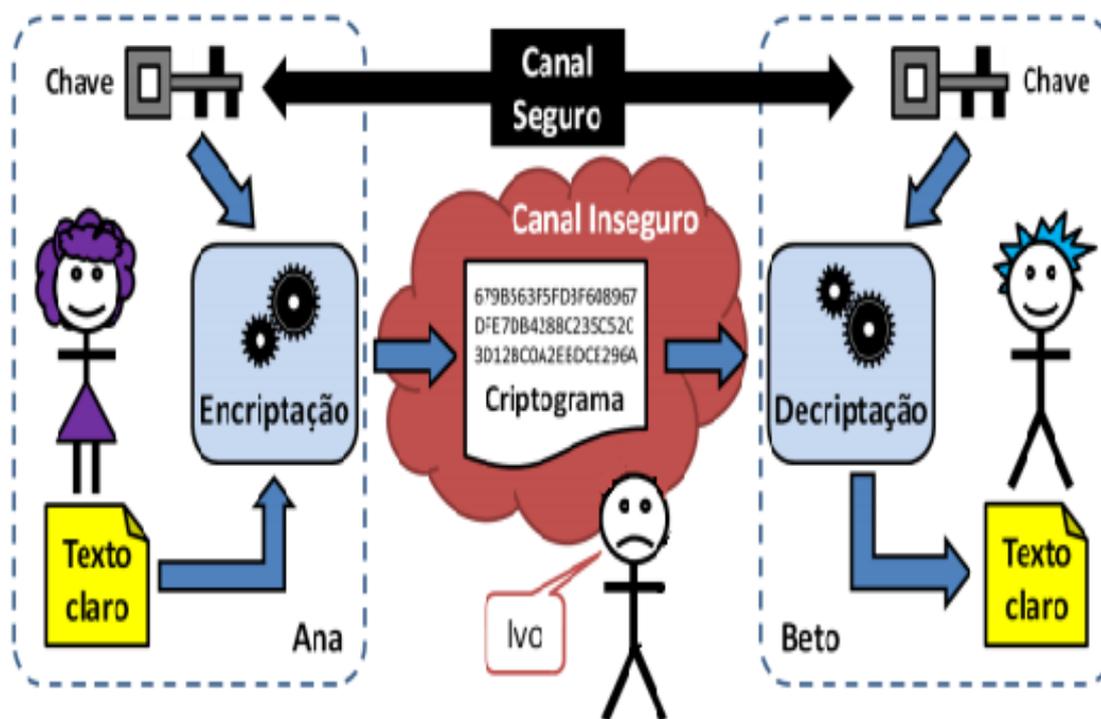
O livro “Introdução à Criptografia para Programadores: Evitando Maus Usos da Criptografia em Sistemas de Software”, escrito por Alexandre Braga e Ricardo Dahab, nos diz que, basicamente a encriptação atua com foco em fornecer confiabilidade, autenticação, integridade e irrefutabilidade:

- A confidencialidade (ou sigilo) - obtida através da utilização da criptografia com a função de manter uma informação secreta, confidencial.
- A autenticação - obtida através da utilização da criptografia para a validação de identidade de uma entidade. Por exemplo, uma assinatura digital para verificar a autora de um documento eletrônico.

- A integridade - obtida através da utilização da criptografia afim de garantir que determinados dados não foram alterados desde o momento que foram criados.
- A irrefutabilidade - obtida através da utilização da criptografia como meio de garantia que um autor de uma mensagem autêntica não negue para outro a sua autoria.

Estes serviços são utilizados juntos em uma encriptação. Ao mesmo tempo que uma mensagem encriptada tenha que fornecer completo sigilo, também tem que manter a integridade, sua autenticação e também sua irrefutabilidade. Veremos uma ilustração representando o sistema criptográfico na Figura 43.

Figura 43 - Sistema Criptográfico

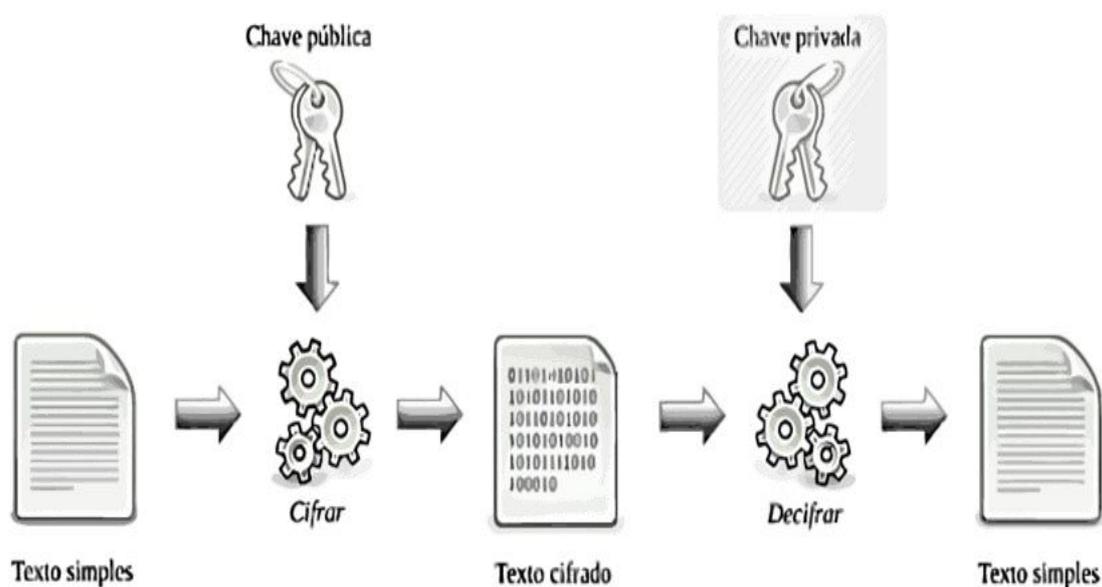


Fonte - Introdução à Criptografia para Programadores: Evitando Maus Usos da Criptografia em Sistemas de Software. (Página 4)

## Criptografia Assimétrica

A criptografia de chave pública é referente a utilização de duas chaves, as quais são desenvolvidas para trabalharem em conjunto pois são matematicamente relacionadas. Uma das chaves do par é denominada como chave privada, sendo mantida em sigilo onde somente o dono, ou melhor, o desenvolvedor do criptograma a conhece. A outra chave do par é denominada como chave pública pois essa pode ser compartilhada publicamente, permitindo o conhecimento de qualquer um. Esse tipo de criptografia pode ser usado para obter maior nível de sigilo além de que, a encriptação assimétrica também permite que qualquer um envie criptogramas para o dono da chave privada. Temos um exemplo ilustrado na Figura 44.

Figura 44 - Ilustração didática sobre encriptação



Fonte - Google

## Rolling Code

Para promover ainda mais segurança, iremos inserir em nosso projeto o *Rolling Code* para trabalhar em conjunto com o sistema de criptografia assimétrica. Ele é basicamente um contador e sua finalidade é evitar ataques de repetição, em que um interceptor grava transmissão e a reproduz em um momento posterior para fazer com que o receptor desbloqueie o sistema.

Sua utilização em nosso projeto, será para que, a partir do momento em que um usuário utilizar o aplicativo com a senha disponível no momento para desbloquear o veículo, a próxima vez em que o sistema for utilizado, o *Rolling Code* entrará em ação, gerando uma nova senha de acesso. Isso evita que, utilizando sempre a mesma senha, um hacker possa conhecê-la e quebrar o sistema de algoritmo criptográfico.

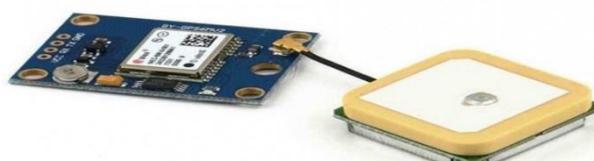
### Localização GPS

Outra proposta futura é a inserção de localização GPS. Essa função permitirá obter a localização exata do veículo em conjunto com os populares aplicativos *Waze* e *Google Maps*. Para isso, será integrado o módulo GPS NEO 6M (Figura 45).

O NEO 6M faz parte de uma família de receptores *stand-alone* de GPS da ublox, sendo esse um receptor flexível e de baixo custo que oferece inúmeras possibilidades de conectividade em seu pequeno *chip*, sendo ideal para aplicações em dispositivos móveis alimentados por bateria onde o custo e espaço são muito restritos.

Esse receptor possui uma estratégia de localização de 50 canais em 6 posições, possuindo um tempo abaixo de um segundo para primeiro “*fix*”, termo esse normalmente utilizado quando o GPS se encontra no espaço, conhecido também como *Time-To-First-Fix*. (UBLOX, 2011) O seu sistema dedicado de aquisição, com 2 milhões de correlacionadores, é capaz de massivas buscas paralelas em espaço de tempo e frequência, possibilitando encontrar satélites instantaneamente. (UBLOX, 2011) Requer alimentação de 2,7V até 3,6V, possuindo como tecnologias de *interface*: UART, USB, SPI e DDC, além de trabalhar nos protocolos: NMEA, UBX e RTCM. (UBLOX, 2011).

Figura 45 - GPS NEO 6M



Fonte - Google

## 6 Referências Bibliográficas

ABLA. **Setor de Locação**. Disponível em: <https://www.abla.com.br/setor-de-locacao/>

ABLA. **Quem somos**. Disponível em: <https://www.abla.com.br/quem-somos/>

ADILSON THOMSEN. FILIPEFLOP, **O que é Arduino?** Disponível em: <https://www.filipeflop.com/blog/o-que-e-arduino/>

ALEXANDRE BRAGA; RICARDO DAHAB; **Introdução à Criptografia para Programadores: Evitando Maus Usos da Criptografia em Sistemas de Software**. Disponível em: <https://siaiap34.univali.br/sbseg2015/anais/Minicursos/MC1.pdf>

ALUISIU ALVES. **Itaú Unibanco planeja fechar até 400 agências no país, dizem fontes**. Disponível em: <https://economia.uol.com.br/noticias/reuters/2019/05/13/itau-unibanco-planeja-fechar-ate-400-agencias-no-pais-dizem-fontes.htm>

ALYSSON PERCICOTTY DA COSTA; FERNANDA CINIELLO SERMANN; GUSTAVO GRZYBOWSKI DA SILVA. **Desenvolvimento de um Protótipo para Medição de Energia Elétrica**. Disponível em: [https://nupet.daelt.ct.utfpr.edu.br/tcc/engenharia/doc-equipe/2015\\_1\\_06/2015\\_1\\_06\\_final.pdf](https://nupet.daelt.ct.utfpr.edu.br/tcc/engenharia/doc-equipe/2015_1_06/2015_1_06_final.pdf)

BUNO MAJORES RELA; EDSON RAMOS DA SILVA; GUILHERME ALENCAR RODRIGES. **Sistema de Sinalização e Acionamentos Veiculares por Comando de Voz**.

CANAL TECH. **WhatsApp, Facebook e Instagram são os apps sociais mais usados pelos brasileiros**. Disponível em: <https://canaltech.com.br/apps/whatsapp-facebook-e-instagram-sao-os-apps-sociais-mais-usados-pelos-brasileiros-99064/>

CARLOS GONÇALVES; GUSTAVO ORTEGA; VITOR SANTOS. **Sistema On Board de Monitoramento de Frotas**. Disponível em:

<http://fatecsantoandre.edu.br/arquivos/TCC/135-Eletronica/135-TCC0027.pdf>

CONTINENTAL. **Car access via smartphone and tablet**. Disponível em:

<http://continental-carkey.com/smart-device-access>

CURTO CIRCUITO. **TUTORIAL: CONHECENDO O ESP32**. Disponível em:

<https://www.curtocircuito.com.br/blog/conhecendo-esp32/>

DINO. EXAME, **Estatísticas de uso de celular no Brasil**. Disponível em:

<https://exame.abril.com.br/negocios/dino/estatisticas-de-uso-de-celular-no-brasil-dino89091436131/>

DO BIT AO BYET. **Primeiros passos com ESP32 e algumas IDEs**. Disponível em:

<https://www.dobitaobyte.com.br/primeiros-passos-com-esp32/>

DYOGO FAGUNDES. UOL, **Vendas locadoras primeiro semestre**. Disponível em:

<https://motor1.uol.com.br/news/269757/vendas-locadoras-primero-semester/>

ELETROGATE. **Primeiros passos com o Arduino**. Disponível em:

<http://blog.eletrogate.com/arduino-primeiros-passos/>

EMERSON ALECRIM. INFO WESTER, **Tecnologia Bluetooth: o que é e como funciona?**. Disponível em: <https://www.infowester.com/bluetooth.php>

FERNANDO K. **Introdução ao ESP32**. Disponível em:

<https://www.fernandok.com/2017/11/introducao-ao-esp32.html> e em Vídeo Aula em:

[https://www.youtube.com/watch?v=iZc8\\_X8JxYs](https://www.youtube.com/watch?v=iZc8_X8JxYs)

<https://www.youtube.com/watch?v=QM0bC6hAiZs>

FERNANDO K. **Introdução ao ESP8266**. Disponível em:

<https://www.fernandok.com/2017/10/introducao-ao-esp8266.html> e em Vídeo Aula

em: <https://www.youtube.com/watch?v=QXB5WHH9GOE>

FERNANDO PAIVA. MOBILE TIME, **138 milhões de brasileiros têm celular**. Disponível em: <https://www.mobiletime.com.br/noticias/21/02/2018/138-milhoes-de-brasileiros-tem-celular/>

FERNANDO S. MEIRELLES. FGV, **Pesquisa anual do FGVcia - Centro de Tecnologia de Informação Aplicada da FGV-EAESP**. Disponível em: [https://eaesp.fgv.br/sites/eaesp.fgv.br/files/pesti2019fgvciappt\\_2019.pdf](https://eaesp.fgv.br/sites/eaesp.fgv.br/files/pesti2019fgvciappt_2019.pdf)

GUILHERME DA SILVA AMARAL; VICTOR EMANUEL RIBEIRO SILVA. **AUTOMAÇÃO RESIDENCIAL UTILIZANDO A PLATAFORMA ARDUINO E DISPOSITIVOS MÓVEIS**. Disponível em: <https://memoria.ifrn.edu.br/bitstream/handle/1044/1373/Automa%C3%A7%C3%A3o%20residencial%20utilizando%20a%20plataforma%20arduino%20e%20dispositivos%20m%C3%B3veis.pdf?sequence=5&isAllowed=y>

GUSTAVO SUMARES. OLHAR DIGITAL, **Números de downloads e receita gerada por aplicativos batem recorde global**. Disponível em: <https://olhardigital.com.br/pro/noticia/numero-de-downloads-e-receita-gerada-por-aplicativos-batem-recorde-global/71922>

HELTON SIMÕES GOMES. GLOBO G1, **Brasil tem 116 milhões de pessoas conectadas à internet, diz IBGE**. Disponível em: <https://g1.globo.com/economia/tecnologia/noticia/brasil-tem-116-milhoes-de-pessoas-conectadas-a-internet-diz-ibge.ghtml>

HUGO SILVA NOGUEIRA. **Desenvolvimento de um Sistema de Controle de Iluminação por Meio de Arduino e Plataforma Android**. Disponível em: [https://www.monografias.ufop.br/bitstream/35400000/773/1/MONOGRRAFIA\\_DesenvolvimentoSistemaControle.pdf](https://www.monografias.ufop.br/bitstream/35400000/773/1/MONOGRRAFIA_DesenvolvimentoSistemaControle.pdf)

JEFFERSON BELIZÁRIO. TUDO CELULAR, **Pesquisa do IBGE revela que 138 milhões de brasileiros possuem um smartphone**. Disponível em:

<https://www.tudocelular.com/android/noticias/n120658/Pesquisa-revela-indice-uso-smartphones-brasil.html>

JOÃO JOSÉ OLIVEIRA. VALOR, **Setor de locação de veículos cresce 12,3% em 2017, aponta associação**. Disponível em:

<https://www.valor.com.br/empresas/5384437/setor-de-locacao-de-veiculos-cresce-123-em-2017-aponta-associacao>

LARRY HARDESTY. MIT NEWS, **The MIT roots of Google's new software**. Disponível em: <http://news.mit.edu/2010/android-abelson-0819>

LOCALIZA. **Sobre a localiza. Plataforma Localiza**. Disponível em:

<https://www.localizahertz.com/brasil/pt-br/sobre-a-localiza/quem-somos>

MANSOOR IQBAL. BUSINESS OF APPS, **App Download and Usage Statistics**.

Disponível em: <https://www.businessofapps.com/data/app-statistics/#1>

MARTIN EVANS; JOSHUA NOBLE; JORDAN HOCHENBAUM. **Arduino em ação** – Edição Original em inglês publicada pela Manning Publications Co., Sound View CT. #3B, CT 06830 USA. Copyright 2013 pela Manning Publications. Edição em português para o Brasil copyright pela Novatec Editora.

OPUS SOFTWARE. **Estatísticas de uso de celular no Brasil**. Disponível em:

<https://www.opus-software.com.br/estatisticas-uso-celular-brasil>

OTAKEYS. **OTA keys carsharing solution**. Disponível em: <http://otakeys.com/>

RAFAEL MIOTTO. GLOBO, **Venda de veículos novos sobe 9% no Brasil em 2017, após 4 anos seguidos de quedas**. Disponível em:

<https://g1.globo.com/carros/noticia/venda-de-veiculos-novos-sobe-9-no-brasil-em-2017-apos-4-anos-seguidos-de-quedas.ghtml>

RAFFCOM. **Invasão Mobile**. Disponível em:

<https://www.raffcom.com.br/blog/invasao-mobile/>

RAMON VOLTOLINI. TECMUNDO, **Conheça o primeiro smartphone da história.**

Disponível em: <https://www.tecmundo.com.br/celular/59888-conheca-primeiro-smartphone-historia-galerias.htm>

RAPHAEL CORACCINI. PORTAL VAREJO, **Vendas de aplicativos disparam no**

**mundos**. Disponível em: <https://portalnovarejo.com.br/2018/01/vendas-de-aplicativos-dispara-2017-app-annie/>

RECOMÉRCIO. **No Brasil há mais celulares do que habitantes.** Disponível em:

<http://www.recomercio.com.br/brasil-ha-mais-celulares-que-habitantes/>

RANDY NELSON. SENSOR TOWER, **The Top Mobile Apps, Games, and Pub-**

**lishers of 2018: Sensor Tower's Data Digest.** Disponível em:

<https://sensortower.com/blog/top-apps-games-publishers-2018>

Statista Research Department. STATISTA, **Number of mobile phone users world-**

**wide from 2015 to 2020 (in billions).** Disponível em:

<https://www.statista.com/statistics/274774/forecast-of-mobile-phone-users-worldwide/>

TELECO. **Bluetooth: O que é?** Disponível em:

[https://www.teleco.com.br/tutoriais/tutorialblue/pagina\\_1.asp](https://www.teleco.com.br/tutoriais/tutorialblue/pagina_1.asp)

TRACTO. **Entenda por que você não pode ignorar o crescimento do mobile.**

Disponível em: <https://www.tracto.com.br/entenda-por-que-voce-nao-pode-ignorar-o-crescimento-do-mobile/>

UNIDAS. **Perfil da Locação de Veículos no Brasil.** Disponível em:

<http://www.blogdaslocadoras.com.br/locadoras-de-carros/perfil-da-locacao-de-veiculos-no-brasil.html>

WIKIPÉDIA. **Aplicativo móvel.** Disponível em:

[https://pt.wikipedia.org/wiki/Aplicativo\\_m%C3%B3vel](https://pt.wikipedia.org/wiki/Aplicativo_m%C3%B3vel)

ZACK WHITTAKER. ZDNET, **One billion smartphones by 2016, says Forrester.**

Disponível em: <https://www.zdnet.com/article/one-billion-smartphones-by-2016-says-forrester/>

## Apêndice - Programação

A seguir, temos a programação desenvolvida para ser gravada no ESP32:

```
/* Program to Control Relay (ON/OFF) From ESP32 Using Bluetooth Commu-
nication with App Inventor
```

```
* Jonathan and Mark
```

```
*/
```

```
#include "BluetoothSerial.h" //Header File for Serial Bluetooth, will be added by
default into Arduino
```

```
BluetoothSerial ESP_BT; //Object for Bluetooth
```

```
int app;
```

```
int relay = 15;
```

```
void setup() {
```

```
  Serial.begin(9600); //Start Serial monitor in 9600
```

```
  ESP_BT.begin("VKTec_Relay"); //Name of your Bluetooth Signal
```

```
  pinMode (relay, OUTPUT); //Specify that LED pin is output
```

```
}
```

```
void loop() {
```

```
  if (ESP_BT.available()) //Check if we receive anything from Bluetooth
```

```
  {
```

```
    app = ESP_BT.read(); //Read what we receive
```

```
if (app == 0)
{
digitalWrite(relay, HIGH);
}

if (app == 1)
{
digitalWrite(relay, LOW);
}
}
delay(20);
}
```