

TRABALHO DE GRADUAÇÃO

**ESTUDO ACERCA DA IMPLEMENTAÇÃO DE UM SISTEMA DE  
IDENTIFICAÇÃO VEÍCULAR NO CAMPUS DARCY RIBEIRO**

VITOR CARDOSO BORGES LEAL

Brasília – DF

2016

**UNIVERSIDADE DE BRASÍLIA**

**FACULDADE DE TECNOLOGIA  
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

TRABALHO DE GRADUAÇÃO

**ESTUDO ACERCA DA IMPLEMENTAÇÃO DE UM SISTEMA DE  
IDENTIFICAÇÃO VEÍCULAR NO CAMPUS DARCY RIBEIRO**

**Vitor Cardoso Borges Leal**

*Relatório submetido como requisito parcial para obtenção  
do grau de Engenheiro de Redes de Comunicação.*

Banca Examinadora

Prof. Dr. Georges Daniel Amvame Nze,  
ENE/UnB  
*Orientador*

\_\_\_\_\_

Prof. Dr. Ugo Silva Dias, ENE/UnB  
*Examinador*

\_\_\_\_\_

Prof. Dr. Diego Martins de Oliveira, Esp., IFB  
*Examinador*

\_\_\_\_\_

*Dedico este trabalho à minha família,  
amigos, professores e todos que  
contribuíram para minha formação  
pessoal e acadêmica.*

## **Agradecimentos**

*Agradeço, a Deus pelo privilégio e a força principalmente nos momentos difíceis.*

*A minha família que é o maior pilar de sustentação .*

*Aos meus amigos e minha namorada que sempre estiveram ao meu lado.*

*Ao meu orientador, Professor Georges, por toda a atenção e conselhos.*

---

## RESUMO

A internet das coisas abre cada vez mais portas que resultam em soluções inteligentes para facilitar o cotidiano dos usuários, economizando recursos, disponibilizando um maior controle de sistemas remotamente e até mesmo uma maior quantidade de dados que podem ser apurados. As soluções desenvolvidas utilizando o conceito de IoT propõe melhorias em processos rotineiros e que pode ajudar a melhorar problemas de larga escala da nossa sociedade. Mobilidade urbana é um tema que vem sendo destaque principalmente em grandes cidades. O impacto social, econômico e ambiental dos avanços na área é muito significativo. A maior utilização de transporte público, novas estratégias para diminuir o tempo gasto em trânsito ou para criar rotas mais eficientes tentam solucionar o problema em questão gerado pelo aumento populacional e de carros nas ruas. Com esse aumento do número de carros, a escassez do número de vagas nos pontos de grande concentração populacional vem se agravando. Este trabalho tem como principal objetivo apresentar uma solução de baixo custo para um estacionamento mais eficaz e inteligente com levantamento de dados, facilitando assim, decisões futuras para uma melhoria generalizada tanto para quem administra o espaço quanto para o usuário, e conseqüentemente a vida de uma comunidade. O projeto é focado em um sensor de presença e uma solução de RFID para identificação. Como resultado, obteve-se um sistema de presença com indicação luminosa de vagas, identificação do veículo e o levantamento de todos os dados cadastrados em um banco de dados, possibilitando melhorias na segurança e na eficiência do uso do espaço, utilizando componentes de baixo custo

---

## ABSTRACT

*The Internet of Things is becoming more popular everyday. The smart solutions are used on the daily tasks, making everything easier, optimizing the use of resources and even a better control of the system getting more information virtually. The developed solutions using the IoT concept are focused on helping society to solve big problems that is faced on the daily life. Mobility is one of the most important topics nowadays, especially on the big cities. The social, economic and environmental impacts are really significant. Using public transportation, new strategies to decrease the time spent on traffic or even new and more efficient routes are proposed to solve the problem of the population growth. With this population growth, the number of cars also increase, while the availability of the parking lots are getting lower each day. This project is focused on testing a solution of a smart parking lot, with a presence sensor, while using the RFID technology to control the parking lot, reaching a higher-level security and more information for the administrator of the area. This solution results can be used to make better decisions and is focused to be cheap, making the implementation something possible. As a result there is a low cost system with a ultrasonic sensor, showing the parking lot usage and the LED indication. Besides that, there is an identification through the RFID technology, getting all the information on a MySQL database, improving the security and the efficiency of the space.*

# SUMÁRIO

1. INTRODUÇÃO.....	2
1.1. Motivação .....	2
1.2. Objetivo.....	3
1.3. Organização do Trabalho .....	3
2. FUNDAMENTAÇÃO TEÓRICA.....	4
2.1. Breve Histórico .....	4
2.2. Raspberry Pi.....	4
2.2.1. Raspberry Pi 2 .....	5
2.3. Sistema Operacional .....	6
2.3.1. Linux .....	7
2.3.2. Raspbian.....	7
2.4. Servidor LAMP.....	8
2.4.1. MySQL.....	8
2.4.2. Apache.....	8
2.4.3. PHP .....	8
2.5. Python .....	8
2.6. Radiofrequência .....	9
2.6.1. RFID.....	9
2.7. Sensores .....	13
2.7.1. Sensor Ultrassônico.....	14
3. METODOLOGIA.....	17
3.1. Fluxograma .....	17
3.2. Descrição das etapas do projeto.....	18
3.3. Implementação.....	19
3.3.1. Instalação do Raspbian.....	19
3.3.2. Acendimento de um LED.....	20
3.3.3. Sensor Ultrassônico.....	22
3.3.4. Implementação do RFID.....	26
3.3.5. Instalação do servidor LAMP .....	28
3.3.6. Desenvolvimento do site .....	28
3.3.7. Construção da Maquete.....	31
4. PROPOSTA.....	33
4.1. Estudo de Caso.....	33
4.2. Resultados e Discussão .....	34
5. CONCLUSÃO.....	38
Referências Bibliográficas.....	39

## LISTA DE FIGURAS

Figura 1. Arquitetura do Raspberry Pi. Fonte[4] .....	5
Figura 2. Diagramas de pinos do Raspberry Pi 2 Model B. Fonte[6] .....	6
Figura 3. Comparação entre sistemas de identificação. Fonte [14]. .....	10
Figura 4. Comparação de Tags passivas e ativas. Fonte: [15] .....	11
Figura 5. RFID-RC522 kit. Fonte[16] .....	12
Figura 6. Diagrama de blocos do RFID-RC522. Fonte[17]. .....	13
Figura 7. Sensor ultrassônico HC-SR04. Fonte[21] .....	15
Figura 8. Especificações HC-SR04. Fonte[21] .....	15
Figura 9. Diagrama de tempo do funcionamento do Sensor HC-SR04. Fonte[21]. .....	16
Figura 10. Fluxograma do trabalho. Fonte: Autor .....	18
Figura 11. NOOBS. Fonte [22]. .....	20
Figura 12. Configuração do circuito para o acendimento do LED. Fonte[23] .....	21
Figura 13. Circuito de divisão de tensão. Fonte: Autor .....	23
Figura 14. Circuito final montado. Fonte: Autor. ....	24
Figura 15. Saída do programa Read.py. Fonte: Autor. ....	27
Figura 16. Arquivos da página Web. Fonte: Autor .....	29
Figura 17. PHP My Admin. Fonte: Autor. ....	30
Figura 18. Pagina Inicial. Fonte: Autor.....	30
Figura 19. Pagina Inicial 2. Fonte: Autor.....	31
Figura 20. Vista aérea do estacionamento da Faculdade de Tecnologia. Fonte[25] .....	32
Figura 21. Circuito com o sistema de presença indicando vaga livre. Fonte: Autor .....	34
Figura 22. Tabela com os dados na página Web. Fonte: Autor.....	35
Figura 23. Maquete do estacionamento da FT. Fonte:Autor .....	36

# LISTA DE TABELAS

Tabela 1. Pinos do Leitor RFID. Fonte: Autor.....	25
Tabela 2. Medições de distancia para lógica de presença. Fonte:Autor.....	34

# LISTA DE SIGLAS E SÍMBOLOS

## Siglas

BIOS	<i>Basic Input/Output System</i>
EEPROM	<i>Electrically-Erasable Programmable Read-Only Memory.</i>
FIFO	<i>First in First out</i>
GND	<i>Ground</i>
GPIO	<i>General Purpose Input/Output</i>
HF	<i>High Frequency</i>
IoT	<i>Internet of Things</i>
LED	<i>Light Emiting Diode</i>
LF	<i>Low Frequency</i>
NOOBS	<i>New out of the box software</i>
RAM	<i>Random access memory</i>
RF	<i>Radio Frequência</i>
RFID	<i>Radio frequency Identification</i>
RTC	<i>Real-time clock</i>
SD	<i>Secure Digital</i>
SoC	<i>System on a chip</i>
SO	<i>Sistema Operacional</i>
SPI	<i>Serial Peripheral Interface</i>
SRAM	<i>Static Random Access Memory</i>
UART	<i>Universal asynchronous receiver/transmitter</i>
UHF	<i>Ultra High Frequency</i>
UID	<i>Unique Identifier</i>

# 1. INTRODUÇÃO

## 1.1. MOTIVAÇÃO

Desde a criação do termo IoT até os tempos atuais, cada vez mais nos aproximamos do tema. Redes de sensores e soluções de automação aliadas com outras tecnologias, lideram um mercado crescente que criam novas soluções para facilitar a vida de todos. Muitas das tecnologias e principalmente os conceitos desses sensores são antigos, mas devido ao crescimento do mercado de placas como Arduino e Raspberry Pi, esses sensores têm se tornado cada vez mais acessíveis e aplicados em pequenas soluções inovadoras.

O crescimento da frota de carros e motocicletas no Brasil é um grande alvo de preocupação, e por isso tem sido acompanhado mais de perto desde 2001. No Distrito Federal, o Departamento de Trânsito (DETRAN) disponibiliza a informação acerca do crescimento da frota veicular desde 2005. Os dados revelam que o crescimento nos últimos 10 anos foi de aproximadamente 100% [1]. No entanto, sabemos que o número de vagas não acompanhou esse crescimento, principalmente em locais onde há grande concentração de pessoas,.

Na UnB especificamente, vemos o crescimento do número de cursos e alunos, mas há um déficit na parte de infraestrutura, que claramente não acompanha esse crescimento. Todas as pessoas de dentro do Campus percebem que o número de vagas está cada vez mais escasso, e que o número de carros estacionados em locais proibidos é cada vez maior.

Por outro lado, a própria UnB também não consegue obter todas as informações necessárias para planejar melhorias. Ainda, outro fator que preocupa a comunidade do Campus é a falta de segurança. Vários desses acontecimentos poderiam ser melhor identificados e até mesmo evitados caso o acesso às informações necessárias fosse apropriado.

A maior eficiência do serviço público na área de segurança ajudaria significativamente a melhorar a situação. Porém, existem outras estratégias capazes de contribuir para que melhorias na infraestrutura sejam efetuadas de uma maneira mais correta, eficiente, e à curto prazo.

## **1.2. OBJETIVO**

Com o intuito de otimizar a utilização do estacionamento público, o presente estudo avalia o uso de sensores de presença que indicam a disponibilidade de vagas em um estacionamento público através de um LED integrado com o RFID para que se realize a identificação do veículo. Além disso, avalia-se os possíveis sistemas a serem utilizados no projeto, visando baixo custo, eficiência, praticidade e segurança.

Outro objetivo do projeto disponibilizar a coleta de informações importantes sobre a situação e utilização do espaço do estacionamento, além de informações do veículo em uma interface voltada para um administrador.

As informações coletadas são colocadas em um banco de dados e disponibilizada para para o administrador ou usuário. Assim é possível trabalho o controle de acesso, análise estatística da utilização do espaço e demais implementações.

## **1.3. ORGANIZAÇÃO DO TRABALHO**

No capítulo 1 foi apresentada uma introdução ao trabalho, expondo uma visão global do problema e suas principais motivações e justificativas. O capítulo 2 traz a fundamentação teórica, com os principais conceitos e fundamentos necessários ao desenvolvimento do trabalho, elucidando os aspectos mais importantes. No capítulo 3, apresenta-se a metodologia, com todas as etapas seguidas. Em seguida, no capítulo 4, as implementações dos cenários. No capítulo 5, os resultados obtidos a partir das simulações. Finalmente, no capítulo 6, a conclusão do projeto e sugestões para trabalhos futuros.

## **2. FUNDAMENTAÇÃO TEÓRICA**

### **2.1. BREVE HISTÓRICO**

Historicamente a humanidade busca mecanizar suas atividades. Desde a invenção da roda até a utilização de controles remotos, o homem tem investido em estratégias para poupar esforço na realização de diversas tarefas cotidianas [2].

Desde meados do século XVIII, o ser humano já buscava avanços na área de automação industrial, visando um aperfeiçoamento de processos rotineiros na produção. Desde então, a automação industrial foi se consolidando até que na década de 1950, período dos anos dourados da indústria, se popularizou [2].

Após se popularizar no ramo industrial, as soluções de automação passaram a ser disponibilizadas nas residências. A utilização de raios infravermelho e RF para controles centrais, entre outras soluções, introduziram a automação residencial. Ao longo do tempo e com a diminuição dos custos, a automação residencial passou a ser utilizada em maior escala.

No final da década de 90, a automação residencial sofreu algumas mudanças. O termo “Internet das Coisas” foi primeiramente citado em 1999. Com a utilização do RFID, a automação residencial deu um grande passo, que proporcionou a sua união com a internet. Isso fez com que ela se tornasse cada vez mais comum não só nas residências, como em locais populares, como shoppings, estacionamentos, hospitais, entre outros [3].

Desde então, os esforços foram direcionados para o aperfeiçoamento dessas soluções, sempre buscando uma maior facilidade no desenvolvimento, que conseqüentemente permitiu que a tecnologia se tornasse mais acessível. Atualmente as placas controladoras e os módulos disponíveis no mercado abrem um leque de oportunidades na área.

### **2.2. RASPBERRY PI**

O Raspberry Pi foi criado pela “The Raspberry Pi Foundation”, com o intuito de disponibilizar um microcomputador de baixíssimo custo e com uma boa performance, inspirado no caso do BBC Micro, que foi utilizado para ensinar programação. Esse

microcomputador possui uma arquitetura ARM e sua construção também foi voltada para a aprendizagem de modo geral, facilitando o acesso à programação, computação, etc.

Atualmente já existem 4 diferentes versões da placa. A primeira foi lançada em 2012 e em 201X foi lançada a segunda geração (Raspberry Pi 2). Em 201X, uma versão ainda mais barata chamada “Raspberry Pi Zero”, e por fim, a terceira versão, “Raspberry Pi 3”, em 2016.

Diferentemente da arquitetura de um computador normal, o Raspberry Pi é um sistema dentro de um chip (arquitetura SoC), que integra processador, GPU e memória dentro de uma única unidade [4]. As principais portas e estruturas do Raspberry Pi são duas saídas de vídeo (Composite e HDMI), portas USB, entrada de cartão SD, saída de áudio, conector de rede RJ45 e pinos GPIO (imagem abaixo).

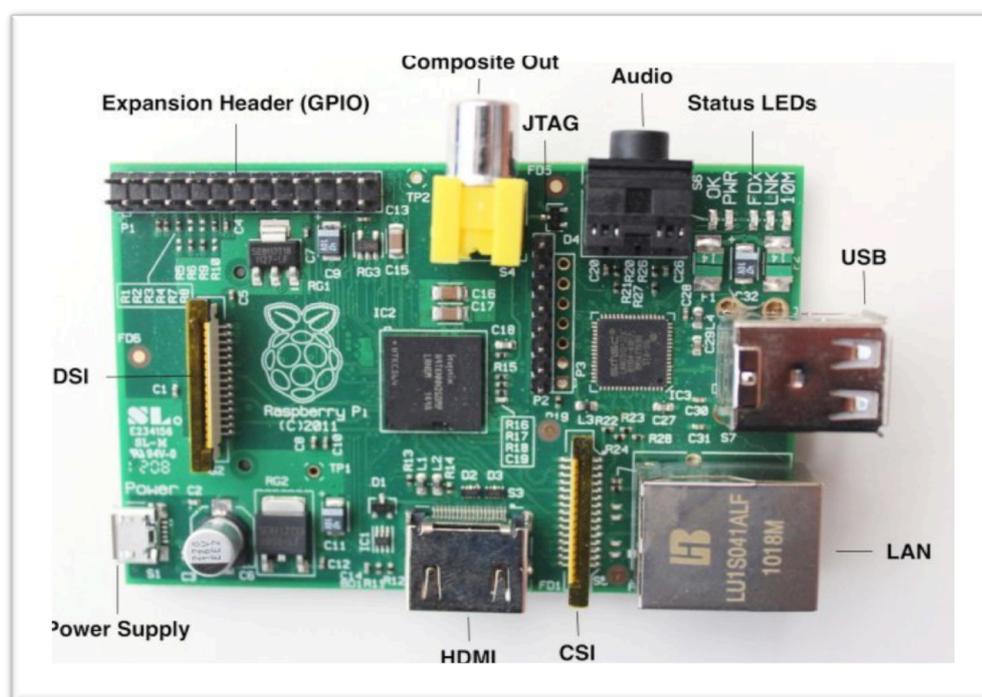


Figura 1. Arquitetura do Raspberry Pi. Fonte[4]

Apesar de todos os pontos positivos do Raspberry PI citados anteriormente, vale ressaltar que até a versão 2 não possuía Bluetooth e Wi-fi interno. Além disso, o Raspberry não possui uma BIOS e nem um RTC, o que exige a configuração via um servidor NTP para que ele informe o horário correto [5].

### 2.2.1. Raspberry Pi 2

A segunda versão da placa foi lançada no início de 2015. O processador foi atualizado, com mais núcleos e melhor performance, e a memória RAM dobrou. Além disso, a The Raspberry Pi Foundation possibilitou a utilização de outros sistemas operacionais, sendo compatível com o Windows 10.

Para realizar os projetos na placa com componentes externos, é necessário utilizar os 40 pinos disponíveis na placa. No caso do Raspberry Pi 2, o diagrama dos pinos é o seguinte:

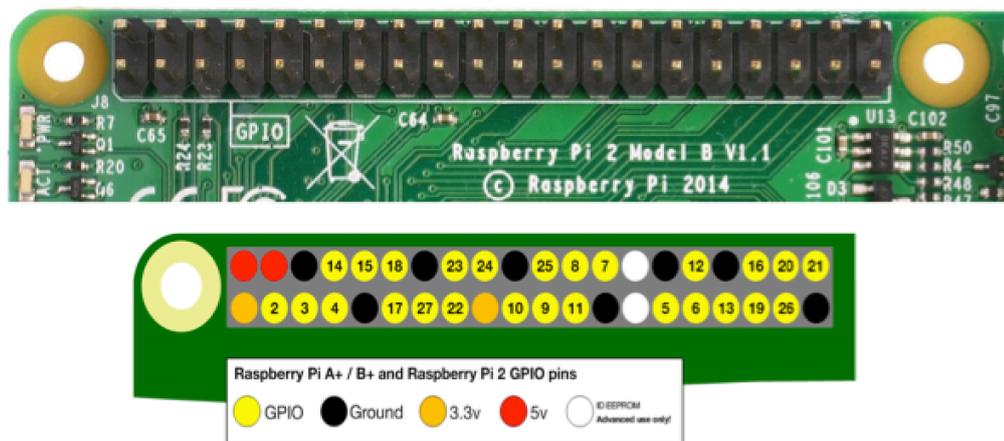


Figura 2. Diagramas de pinos do Raspberry Pi 2 Model B. Fonte[6]

Para realizar o controle do sistema, existiam opções como arduino, PICs e 8051. Algumas placas, são limitadas e não teriam condições de realizar todas as partes do projeto, principalmente quanto ao servidor LAMP e à implementação do leitor RFID. O Raspberry Pi, foi utilizado por possibilitar a construção de todos os sistemas projetados.

No presente projeto, foi utilizado o Raspberry Pi 2, porém, visando uma solução de baixo custo, o Raspberry Pi Zero que possui um valor ainda mais baixo atende às propostas do projeto.

### 2.3. SISTEMA OPERACIONAL

O Raspberry Pi é um computador com suporte a diversos sistemas operacionais, que incluem variações do Linux, Windows 10, Recalbox e OSMC. Cada um desses sistemas é mais apropriado de acordo com a finalidade do uso do Raspberry, seja ele um computador comum, uma central de mídia ou até mesmo um sistema específico para rodar jogos.

Dentre os sistemas suportados pelo Raspberry Pi, destaca-se o Raspbian. Esse sistema operacional é uma variação do Linux Debian, realizada em parceria com a The Raspberry Pi

Foundation, e que visa trazer o melhor ambiente de trabalho possível para atender as finalidades do Raspberry.

### **2.3.1. Linux**

O Linux é um sistema operacional baseado em UNIX, que por sua vez foi criado em 1969 pela Bell Labs [7]. A intenção do UNIX era unificar e simplificar os sistemas que diversos computadores utilizavam anteriormente, uma vez que estes precisavam de um sistema específico para realizar as tarefas.

O código fonte do UNIX foi escrito na linguagem C. Por conta do monopólio da AT&T no setor de telefonia e do desenvolvimento do código fonte através da AT&T Bell Laboratories, o sistema teve que ser inicialmente disponibilizado em código aberto, o que aumentou muito sua popularidade.

Posteriormente a AT&T foi dividida e o UNIX passou a ser comercializado. Linus Torvalds, um estudante de Ciência da Computação na época, resolveu criar um sistema baseado no UNIX mas que fosse voltado para o meio acadêmico e totalmente gratuito, e foi assim que o Linux foi concebido. Desde então o apoio para seu desenvolvimento foi cada vez maior e várias variações do sistema foram criadas, sendo as mais populares o Debian, Ubuntu e Red Hat.

Atualmente o Linux além de ser um sistema ideal para estações de trabalho, é muito utilizado em servidores middle-end e high-end e até mesmo em aplicações de sistemas embarcados, celulares e outros dispositivos móveis [7].

### **2.3.2. Raspbian**

O Raspbian é o sistema operacional mais comumente utilizado no Raspberry Pi. Foi elaborado com base no Linux Debian e adaptado especialmente para ser utilizado no microcomputador da The Raspberry Pi Foundation, incluindo customizações específicas e alguns softwares nativos.

O Linux Raspbian foi utilizado por se mostrar o sistema mais confiável e mais completo para todas as soluções relacionadas ao Raspberry Pi. O Windows ainda se mostrava um sistema recente na plataforma, e outras opções foram descartadas por não se atender os requisitos.

## **2.4. SERVIDOR LAMP**

Conhecido como LAMP stack, o servidor LAMP é a abreviação de todas as aplicações utilizadas para a solução. O sistema operacional utilizado é o Linux, o servidor web é o Apache, o banco de dados MySQL e a programação da página web é PHP. A combinação é comprovadamente segura e eficiente [8].

A utilização do LAMP foi determinada pela principal característica do servidor, ser open source com alta performance.

### **2.4.1. MySQL**

O MySQL é um banco de dados gratuito que possui alta performance com as funções de leitura e escrita. Foi adotado por várias empresas e até mesmo órgãos governamentais, principalmente por ter velocidade equivalente ao banco de dados Oracle e outros bancos de dados que são pagos, o que permite escrever e ler os dados através de um script PHP [8].

### **2.4.2. Apache**

Desenvolvido pela “Apache Software Foundation”, o Apache é o servidor web “open source” mais utilizado no mundo. Oferece segurança e funciona perfeitamente no Linux. Com ele, é possível criar hosts virtuais e hospedar diversos sites na mesma máquina.

### **2.4.3. PHP**

Com suporte a diversos protocolos, PHP é a linguagem de programação voltada para scripts e normalmente utilizada para a plataforma web.

## **2.5. PYTHON**

O Python é a principal linguagem de programação utilizada no Raspberry Pi para desenvolvimento de jogos e outros softwares, assim como scripts. “É uma linguagem de

altíssimo nível (em inglês, *Very High Level Language*) orientada a objeto, de tipagem dinâmica e forte, interpretada e interativa” [9].

A linguagem possui uma sintaxe simples e minimalista e possui diversas estruturas de alto nível como listas e dicionários que facilitam o desenvolvimento, além de recursos como geradores, introspecção, persistência, metaclasses e unidades de teste [9].

## 2.6. RADIOFREQUÊNCIA

A tecnologia de radiofrequência (RF) é a exploração do fenômeno das ondas eletromagnéticas dentro do espectro entre as frequências de 3kHz e 300GHz (1Hertz = um ciclo de onda por segundo), e pode ser considerada uma das mais importantes da sociedade moderna [10]. As ondas eletromagnéticas são as perturbações eletromagnéticas constituídas por campos elétricos e magnéticos que variam com o tempo e podem se propagar de uma região do espaço para outra [11].

Cada onda tem um comprimento, velocidade e frequência específica. É possível determinar essas características de cada onda utilizando a relação:

$$v = \lambda f \quad (1)$$

Onde  $\lambda$  é o comprimento de onda e  $f$  a frequência dessa onda. Nesse caso temos uma relação inversamente proporcional entre frequência e comprimento, e podemos concluir que os alcances de ondas com frequência menores são maiores do que as ondas de frequência maiores.

O grande avanço do RF veio com o surgimento de circuitos integrados e dos semicondutores, o que abriu portas para equipamentos mais compactos e portáteis. Esse progresso resultou na revolução da comunicação móvel. Desde então o tamanho dos equipamentos vem diminuindo progressivamente [10].

### 2.6.1. RFID

Uma das tecnologias que surgiu com os avanços na área de RF, é a denominada tecnologia de identificação por rádio frequência, cuja sigla em inglês é RFID (*Radio Frequency Identification*) [12]. O termo descreve o sistema de identificação de objetos que utiliza a rádio frequência para transportar as informações [13].

Atualmente, existem diversas soluções de identificação que visam principalmente atender às demandas da indústria. O código de barras foi o principal sistema utilizado durante

um longo tempo na maioria dos ramos industriais. Existem hoje, diversas tecnologias que são utilizadas para realizar essa tarefa, incluindo OCR, leitura biométrica, Smart card e o já citado RFID [14]. Dentre as principais aplicações do RFID estão o controle de acesso, armários inteligentes para controle do conteúdo interno, sistema para localização de itens, entre outros [13].

O OCR é uma identificação dada ao produto com uma escrita legível para o ser humano (diferentemente do código de barras), mas que também pode ser lida por uma máquina (leitor de OCR). Atualmente é utilizado em bancos e outros tipos de serviços, porém sua popularização não foi possível devido ao custo de implementação da tecnologia.

Sistemas biométricos como digitais e reconhecimento por voz surgiram como soluções de identificação, porém o foco principal dessa tecnologia é a altíssima segurança no reconhecimento de pessoas.

Smart cards são cartões que também são utilizados na identificação de objeto. Possuem um microprocessador e conseguem guardar uma quantidade maior de dados. O RFID possui um sistema parecido, mas a troca de dados é feita através da rádio frequência, que é mais simples e não necessita, portanto, de um contato galvânico [14].

Pode-se verificar um comparativo completo entre todas as soluções de identificações citadas:

System parameters	Barcode	OCR	Voice recognition	Biometry	Smart card	RFID systems
Typical data quantity (bytes)	1–100	1–100	–	–	16–64 k	16–64 k
Data density	Low	Low	High	High	Very high	Very high
Machine readability	Good	Good	Expensive	Expensive	Good	Good
Readability by people	Limited	Simple	Simple	Difficult	Impossible	Impossible
Influence of dirt/damp	Very high	Very high	–	–	Possible (contacts)	No influence
Influence of (optical) covering	Total failure	Total failure	–	Possible	–	No influence
Influence of direction and position	Low	Low	–	–	Unidirectional	No influence
Degradation/wear	Limited	Limited	–	–	Contacts	No influence
Purchase cost/reading electronics	Very low	Medium	Very high	Very high	Low	Medium
Operating costs (e.g. printer)	Low	Low	None	None	Medium (contacts)	None
Unauthorised copying/modification	Slight	Slight	Possible* (audio tape)	Impossible	Impossible	Impossible
Reading speed (including handling of data carrier)	Low ~4 s	Low ~3 s	Very low >5 s	Very low >5–10 s	Low ~4 s	Very fast ~0.5 s
Maximum distance between data carrier and reader	0–50 cm	<1 cm Scanner	0–50 cm	Direct contact**	Direct contact	0–5 m, microwave

Figura 3. Comparação entre sistemas de identificação. Fonte [14].

Os dois componentes principais do sistema RFID são as tags, que são os transponders posicionados nos objetos a serem identificados; e o leitor, que é o equipamento que recebe as informações [13].

Os leitores contêm normalmente um módulo de rádio frequência, que é composto por um transmissor e um receptor, uma unidade de controle, e a interface de leitura do transponder. Além disso, alguns leitores possuem interface serial para a transferência dos dados coletados para um sistema (Computador, Sistema de controle, etc.) [14].

As tags que contêm a informação do objeto possuem uma interface de comunicação e um microchip eletrônico que guarda as informações do item identificado.

Existem dois tipos de tags, ativas e passivas, que podem ser em formato de moeda, cartão plástico ou de vidro. As tags passivas não possuem nenhuma fonte de energia direta, sendo alimentada pela energia gerada pelo campo eletromagnético gerado pelo leitor. Já as ativas possuem uma fonte própria de energia que normalmente é uma bateria ou uma célula solar. Na prática, a principal diferença é o alcance do sistema [14].

### **Tags Ativas x Passivas**

	<b>Tags Ativas</b>	<b>Tags Passivas</b>
<b>Alimentação</b>	<b>Bateria</b>	<b>Alimentado pelo leitor</b>
<b>Distancia</b>	<b>Longa</b>	<b>Curta</b>
<b>Custo</b>	<b>Caro</b>	<b>Barato</b>
<b>Durabilidade</b>	<b>Limitado pela bateria</b>	<b>Alta durabilidade</b>
<b>Memória</b>	<b>Alta</b>	<b>Pequena</b>
<b>Transmissão de dados</b>	<b>Transmissão periódica</b>	<b>Somente quando ativada</b>
<b>Capacidade de armazenamento</b>	<b>Capacidade maior</b>	<b>Capacidade menor</b>
<b>Aplicação</b>	<b>Monitoração contínua</b>	<b>Leitura específica</b>

Figura 4. Comparação de Tags passivas e ativas. Fonte: [15]

O sistema do RFID funciona desde milímetros até uma distância acima de quinze metros (aproximadamente trinta metros dependendo do sistema) [14][15]. Existem quatro faixas de frequências principais na utilização de sistemas RFID:

- Low-Frequency(LF): 125-134.2kHz. Nesse caso, as tags são utilizadas para sistemas de aproximadamente um metro e que precisam penetrar em diferentes materiais [15].
- High-Frequency(HF): 13.56MHz. Utilizada para soluções com distância por volta de dois metros, possuem uma taxa de 106Kbps [15].

- Ultra-High-Frequency(UHF): 869-915MHz. Frequência utilizada para distâncias maiores e sua frequência depende do país em questão. Nesse range de frequência o consumo de energia do sistema é maior [15].
- Frequência de micro-ondas: 2.45GHz. Frequência normalmente utilizada quando as tags ativas são utilizadas. Faixa de frequência mais sujeita a interferências [15].

Uma das grandes vantagens do RFID é a possibilidade de uma comunicação duplex. As tags possuem as informações a serem lidas, mas o leitor pode escrever dados nas tags. As duas opções das comunicações de tags mais complexas são half-duplex (transferência de dados em momentos distintos) e full-duplex (troca de informações simultaneamente em ambos sentidos). A utilização da capacidade de memória do sistema varia de acordo com o tipo de tag. Algumas tags possuem memória EEPROM e SRAM para receberem informações, sendo a memória EEPROM para uma quantidade menor de dados, e a SRAM para sistemas mais complexos [14].

Existem também tags de baixíssimo custo que possuem um transponder de 1bit, utilizado principalmente em sistemas antifurtos, uma vez que indica a presença ou ausência do objeto [14].

Vale ressaltar entre os tipos de transponders o sistema RFID utilizado no conceito do projeto RFID-RC522. O sistema contém um módulo de leitura/escrita e uma tag do tamanho de uma moeda. A figura abaixo representa o sistema de RFID:



Figura 5. RFID-RC522 kit. Fonte[16]

A frequência utilizada é a de 13.56MHz (HF), possui uma taxa de comunicação entre o leitor e a tag de 424Kb/s, com suporte a interface UART, SPI e I2C. No caso da interface

SPI, é possível alcançar uma taxa de 10Mb/s entre o módulo e o PC. Possui um buffer FIFO que armazena e envia 64 bytes. A alimentação do módulo deve ser entre 2.5V e 3.3V [17].

O diagrama de blocos do leitor do sistema RFID-RC522 pode ser observado a seguir:

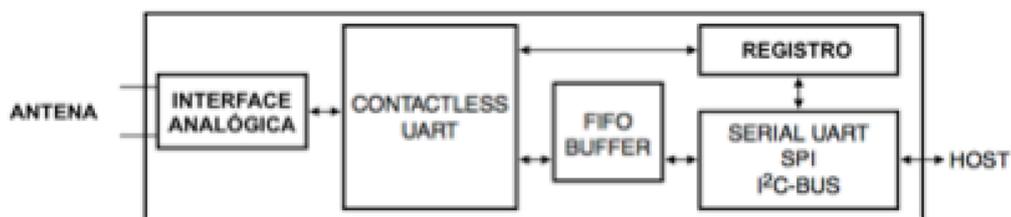


Figura 6. Diagrama de blocos do RFID-RC522. Fonte[17].

O RFID-RC522 é um sensor HF, sendo necessária a leitura próxima do leitor, sendo assim uma solução diferente do proposto para uma leitura real de longo alcance (utilizando frequência UHF), o projeto foi realizado em uma escala menor, testando o conceito do sistema como um todo, próximo da realidade. Para implementar a leitura completa seria necessário adquirir um outro sensor mais caro, porém a implementação de ambos os casos é muito próxima.

## 2.7. SENSORES

Diversos sistemas e equipamentos precisam de dados que são colhidos do ambiente. Os sensores são os responsáveis por fazer essa coleta de dados do mundo externo e entregá-los aos equipamentos eletrônicos. “Na eletrônica, um sensor é conhecido como qualquer componente ou circuito eletrônico que permita a análise de uma determinada condição do ambiente” [18].

Os sensores podem ser considerados como um tipo de transdutor específico. Transdutor é o componente capaz de transformar energia, e no caso dos sensores são energias como luz, calor e movimento, transformadas em energia elétrica. A energia elétrica de saída do sensor é o que traz as informações do ambiente a serem lidas [18]. Existem diversos tipos de sensores e cada um trabalha de maneira diferente com as grandezas físicas. Estes são divididos em dois grandes grupos: analógico e digital.

O sensor analógico responde a sinais analógicos, o que permite um infinito número de possibilidades nas leituras, mesmo que dentro de uma faixa de valores. Já os sensores digitais respondem a sinais digitais e com isso utilizam lógicas binárias com níveis bem definidos

[18]. Dentro desses dois grandes grupos podemos dividir os sensores de acordo com seu tipo de recurso.

Os sensores mecânicos são aqueles que detectam movimentos utilizando um recurso mecânico, como por exemplo chaves [19]. Os circuitos chaves fim-de-curso e os reed-switch são exemplos de sensores mecânicos.

Sensores fotoelétricos são sensores que trabalham com a luz e possuem um tempo de resposta menor do que os sensores mecânicos. Sensores foto-resistores (LDR), fotocélula, fotodiodo e foto transistor são exemplos de sensores fotoelétricos [20].

Os sensores capacitivos são os que detectam mudanças em campos eletrostáticos, e os sensores indutivos são aqueles que emitem sinais, detectando sem contato direto corpos metálicos que atravessaram o seu campo magnético [19].

Existem também os sensores térmicos, que assim como os fotoelétricos são variados, e trabalham com a variação da temperatura. Entre eles estão os sensores NTC, PTC e piroelétrico, além de sensores de aceleração (ou acelerômetro), sensores de velocidade, pressão, nível, vazão, entre outros [19,20].

Pode-se avaliar uma infinidade de parâmetros captados por sensores. Alguns parâmetros podem ser captados por mais de um tipo de sensor, isso é, dependendo da aplicação, sensores diferentes podem atender à mesma finalidade só que por meios diferentes. Por fim, devemos destacar os sensores ultrassônicos, que foram os escolhidos para desenvolver o projeto.

### **2.7.1. Sensor Ultrassônico**

O sensor ultrassônico é considerado um sensor de posição ou distância. O funcionamento do sensor baseia-se no envio e recepção de ondas ultrassônicas. Um oscilador emite ondas ultrassônicas em uma frequência de aproximadamente 40Hz que resultam em ondas com comprimento na ordem de centímetros, possibilitando assim a detecção de objetos pequenos [20,21].

Ao emitir as ondas ultrassônicas pelo oscilador, um clock é acionado e assim as ondas trafegam até refletir no objeto e voltar para serem captadas pelo sensor, trazendo assim as informações de presença e do tempo de viagem da onda [20].

O sensor ultrassônico mais comum no mundo da eletrônica é o HC-SR04, que está ilustrado na Figura 7.

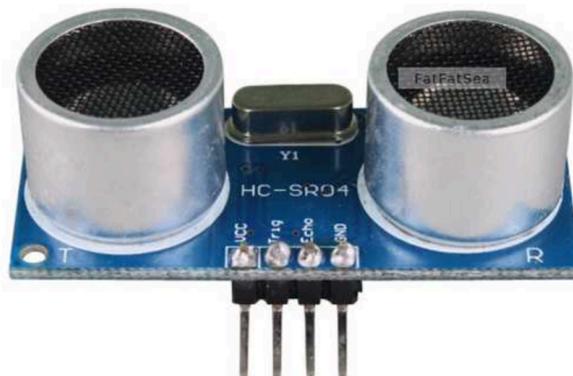


Figura 7. Sensor ultrassônico HC-SR04. Fonte[21]

Com ele é possível realizar medições entre dois centímetros e quatro metros [21]. As especificações elétricas do componente podem ser verificadas na seguinte imagem:

### Parâmetros Elétricos

<b>Vttagem</b>	<b>DC 5 V</b>
<b>Corrente</b>	<b>15mA</b>
<b>Frequência de funcionamento</b>	<b>40Hz</b>
<b>Distância máxima</b>	<b>4m</b>
<b>Distância mínima</b>	<b>2cm</b>
<b>Angulação</b>	<b>15 Graus</b>
<b>Trigger Input</b>	<b>Pulso 10uS TTL</b>
<b>Echo Output</b>	<b>Sinal Input TTL e a distância em proporção</b>
<b>Dimensões</b>	<b>45*20*15mm</b>

Figura 8. Especificações HC-SR04. Fonte[21]

O sensor começa a funcionar com um trigger inicial para começar o distanciamento. Assim que o trigger é acionado, o sensor envia oito ondas ultrassônicas em modo “burst” (jarrada) e assim eleva seu echo. A distância é então calculada a partir da informação do tempo que a onda levou entre o trigger enviado e o echo recebido [21]. O diagrama de funcionamento baseado no tempo pode ser observado na seguinte imagem:

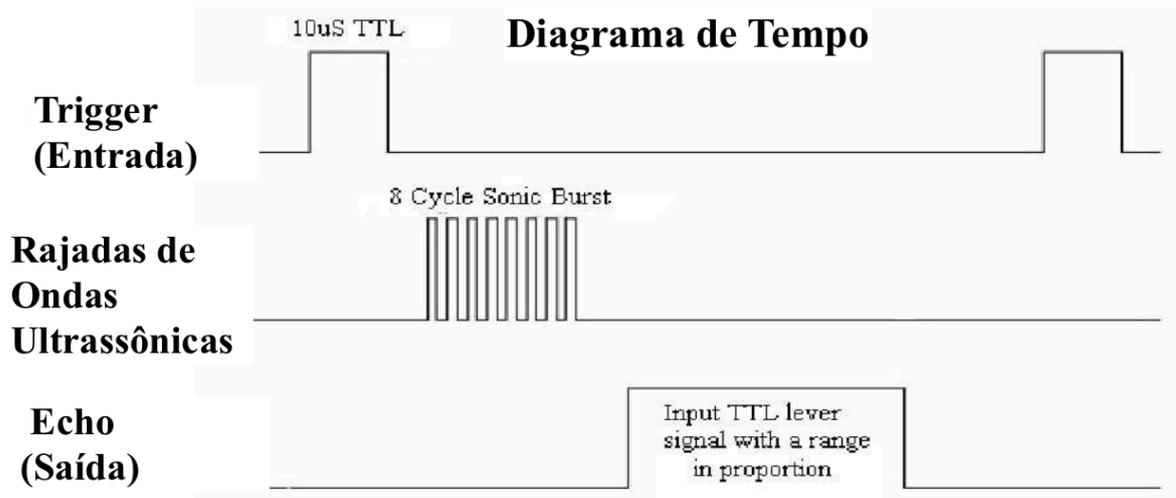


Figura 9. Diagrama de tempo do funcionamento do Sensor HC-SR04. Fonte[21].

Entre todas as opções de sensores de presença, vale destacar as três principais: Ultrassônico, infravermelho e indutivo. O projeto é focado em uma solução de baixo custo, desse modo, foi realizada a comparação dos sensores. Todos os sensores possuem alta confiabilidade, realizando diversas leituras com baixíssima taxa de erro. O sensor ultrassônico é o sensor mais barato entre as opções citadas, nesse pressuposto o sensor foi escolhido para contribuir com o custo final do projeto.

### **3. METODOLOGIA**

#### **3.1. FLUXOGRAMA**

O fluxograma a seguir indica as etapas que foram executadas para a conclusão do projeto:

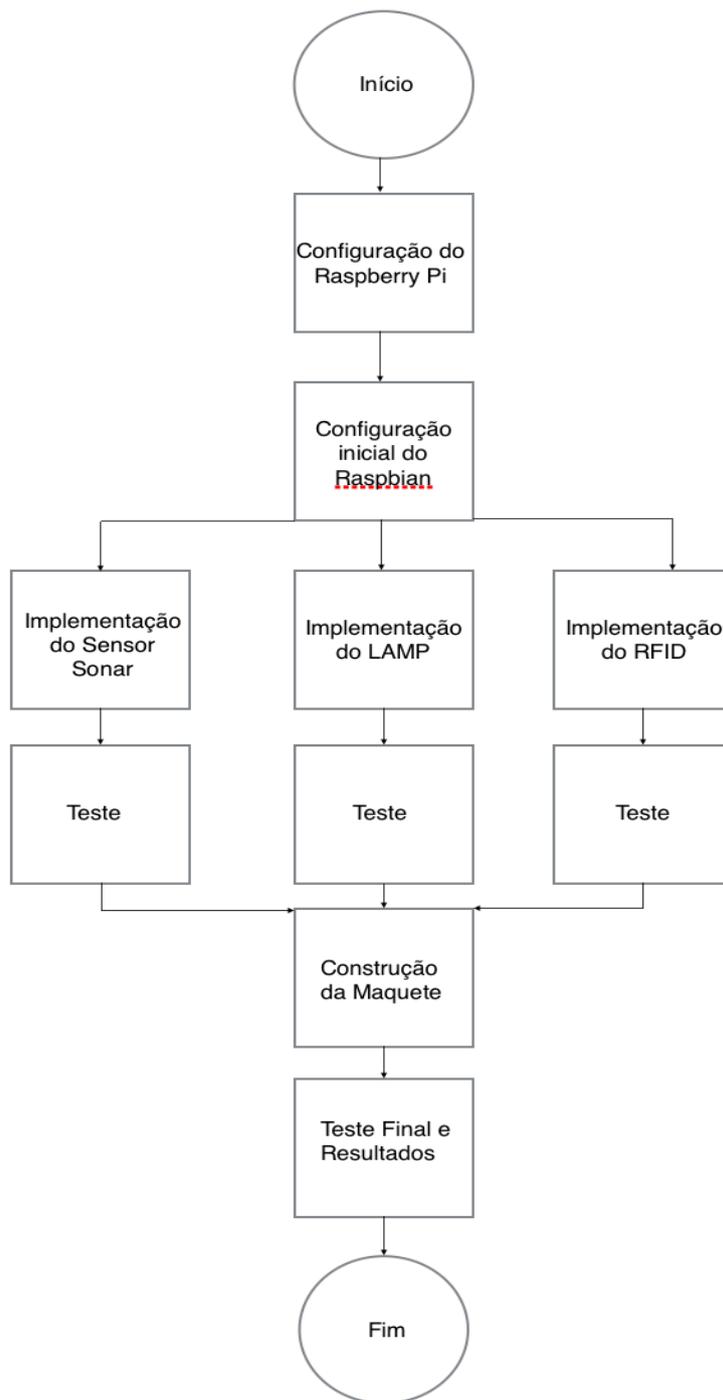


Figura 10. Fluxograma do trabalho. Fonte: Autor

### 3.2. DESCRIÇÃO DAS ETAPAS DO PROJETO

Baseado no fluxograma do projeto, pode-se elencar sete etapas dentro do projeto. Primeiramente foi necessário configurar o ambiente de trabalho, nesse caso o Raspberry Pi.

O Raspberry Pi, como citado na sessão 2.3, não possui um sistema operacional predefinido, e portanto, primeiramente é necessário escolher um sistema para instalar. Depois de instalado, o Raspbian requer alguns ajustes iniciais de instalação e atualização, para que assim seja feita toda a implementação do projeto.

Inicialmente foi realizada uma etapa mais simples de ambientação com a linguagem Python com o acendimento de um LED. Depois disso, foi possível ligar o sistema fisicamente e começar a escrever o código para efetuar a leitura, interpretar os dados e realizar o acendimento do LED de acordo com a lógica. Em seguida, foram realizadas instalações do Apache, MySQL e PHP (LAMP server).

A parte de implementação do RFID se deu de forma paralela ao desenvolvimento do site, que foi feito em Wordpress. Nesse caso, o RFID era implementado para realizar as leituras, fazer a autenticação da tag e assim escrever no banco de dados, enquanto que a primeira versão do site era construída.

Na última etapa, elaborou-se a maquete do projeto e realizou-se um teste final com uma versão final do site, o conceito testado e os resultados obtidos.

### **3.3. IMPLEMENTAÇÃO**

Nessa seção estão descritas todas as etapas de implementação do projeto conceito de estacionamento inteligente voltado para o Campus Darcy Ribeiro. No projeto, foi utilizado o Raspberry Pi como computador central, um sensor ultrassônico para detectar presença, indicação luminosa de vagas livres/ocupadas e o sistema de identificação por rádio frequência (RFID).

As informações do sistema são gerenciadas através de uma página web, utilizando o Raspberry Pi como servidor LAMP.

#### **3.3.1. Instalação do Raspbian**

Para realizar a instalação do Raspbian, primeiramente foi necessário instalar o NOOBS no cartão de memória SD. O NOOBS é um instalador de SO com uma interface gráfica e mais intuitiva, facilitando assim o processo inicial com o Raspberry Pi.

Após a instalação do NOOBS, o cartão de memória foi inserido no Raspberry Pi e a partir daí o microcomputador já pôde ser acessado. O Raspbian foi escolhido na seguinte tela:

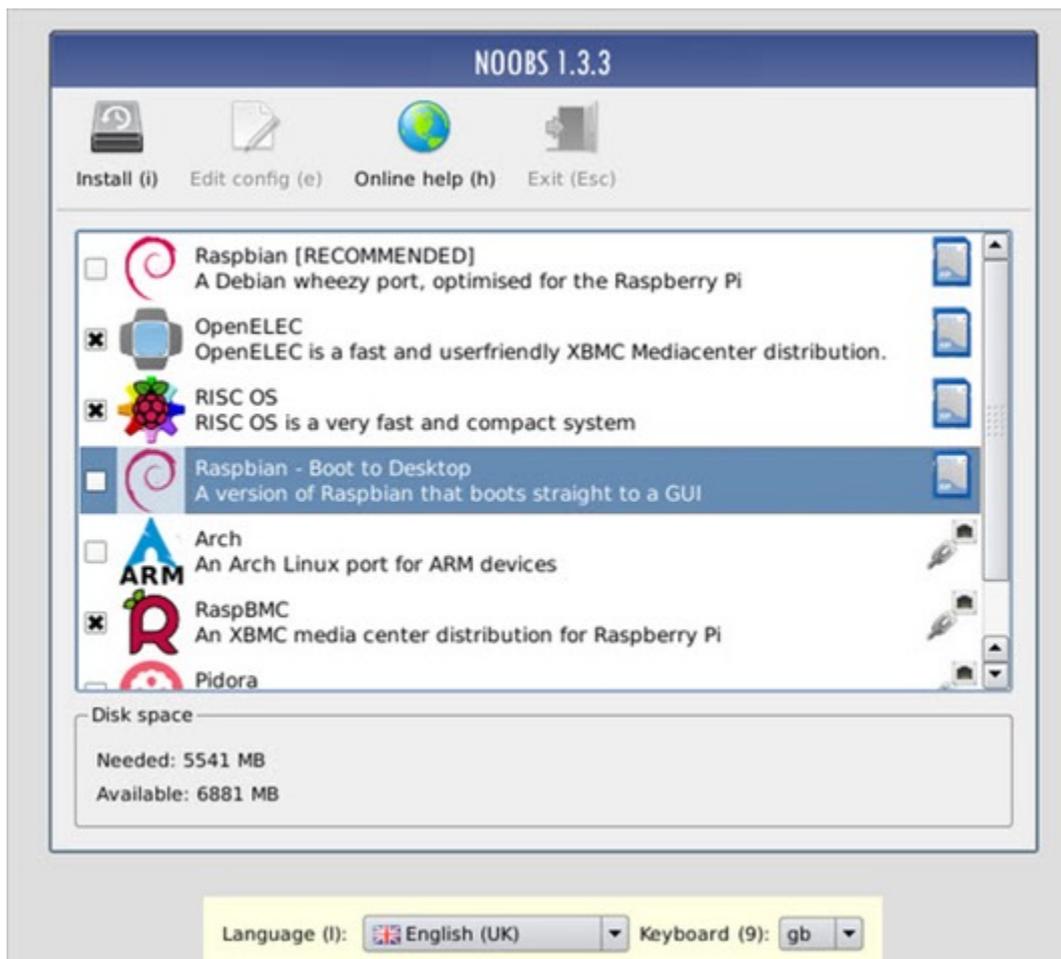


Figura 11. NOOBS. Fonte [22].

Para uma maior praticidade no trabalho com o Raspberry Pi, foi utilizado um adaptador Wi-fi (já que o Raspberry Pi 2 não possui Wi-fi nativo como já citado na seção 2.2), que faz parte do Kit do Raspberry Pi adquirido.

Assim que finalizada a instalação do Raspbian, foi realizado um estudo dos códigos mais simples da linguagem Python para um melhor entendimento da linguagem e do sistema.

### 3.3.2. Acendimento de um LED

O primeiro e mais básico passo do projeto foi o acendimento do LED. Além de ser utilizado no projeto final, o programa foi ideal para um início de aprendizado da linguagem Python e do funcionamento do próprio Raspberry Pi.

Para isso, foi construído um circuito com o LED e um resistor conectados na protoboard. O diagrama do circuito foi o seguinte:

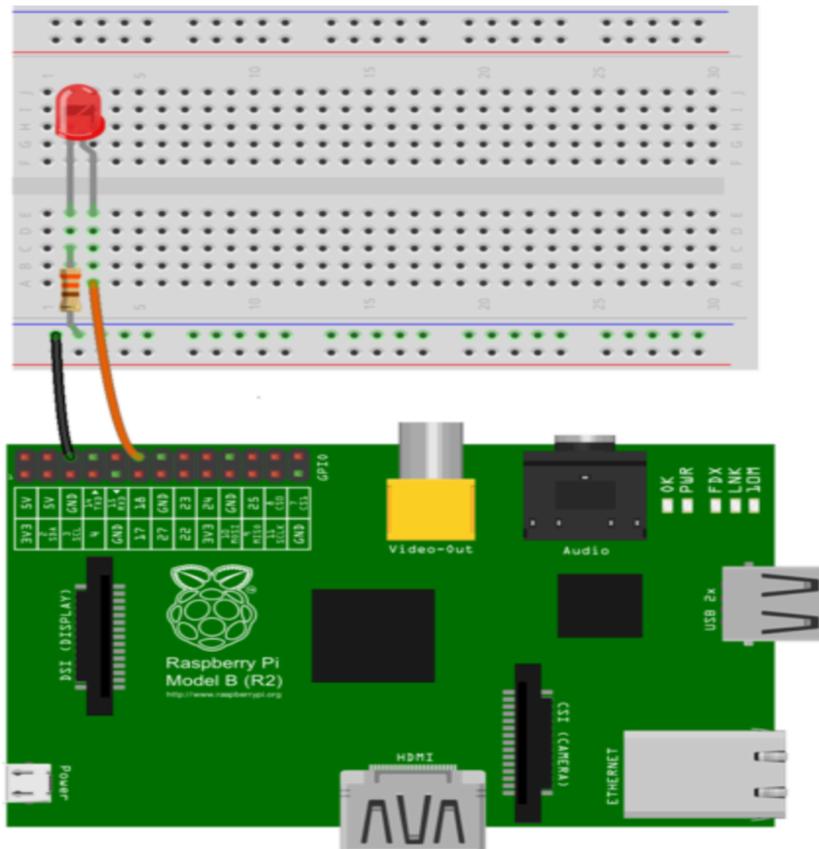


Figura 12. Configuração do circuito para o acendimento do LED. Fonte[23]

Para realizar o acendimento, foi realizada a conexão do LED no pino 12 GPIO que é para a alimentação do sistema junto com a parte lógica para mudança de estado do pino.

A alimentação do sistema é feita com 3,3V do lado positivo do LED (perna mais longa). O pino 6 de GND ou 0V é ligado no resistor que limita a corrente e fica lado negativo do LED (perna mais curta).

Um resistor de 330 Ohms foi escolhido para o projeto, uma vez que sua função no circuito é limitar a corrente do sistema e consequentemente a luminosidade do LED. Foi criado um arquivo de texto com a extensão `.py`:

```
nano LED.py
```

Dentro desse arquivo foi feita a lógica de programação para o script de acendimento do LED da seguinte maneira [23]:

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
GPIO.setup(18,GPIO.OUT)
print "LED aceso"
GPIO.output(18,GPIO.HIGH)
time.sleep(1)
print "LED apagado"
GPIO.output(18,GPIO.LOW)
```

### 3.3.3. Sensor Ultrassônico

Para a implementação do sensor de presença para a detecção dos carros estacionados foi utilizado um sensor ultrassônico que fornece a distância do objeto como citado na seção 2.7.1.

Com a informação da distância, pode-se identificar a presença de algum corpo entre o sensor e a superfície de leitura. Caso o tempo de leitura seja o padrão do sistema, a distância lida será igual à distância do sensor até a superfície, e conseqüentemente não existe nada entre o sensor e a superfície; mas caso a leitura do tempo seja mais curta, entende-se que a onda emitida foi refletida por algum corpo entre o sensor e a superfície.

Para a detecção de um carro, foi calculado o valor de distância de leitura do sensor quando a vaga está livre, e a mudança da distância pela presença de um carro entre o sensor e o chão.

Primeiramente analisou-se o diagrama de pinos do sensor, composto por quatro entradas: (1) alimentação de 5V, (2) Pulso de trigger (input), (3) Pulso de echo (output), (4) GND.

A alimentação e o pino de GND são conectados diretamente no Raspberry Pi nos pinos 2 e 39. O pino do trigger que inicia a leitura também foi ligado diretamente entre o Raspberry Pi e o sensor (pino 16), já que o Raspberry Pi que manda o pulso na entrada do sensor. Dessa maneira, o sensor dispara as ondas ultrassônicas e aguarda o recebimento.

Já o pino de ECHO funciona para receber o pulso que indica o retorno da onda ultrassônica e, portanto requer um cuidado especial. Nesse caso, o sensor vai mandar um

pulso com uma voltagem de 5V, enquanto que a voltagem indicada de entrada do pino GPIO do Raspberry Pi é de 3.3V.

No projeto foram utilizados dois resistores para fazer um circuito de divisão de voltagem, esse circuito foi necessário pois o sensor possui uma alimentação de 5V, porém, a porta GPIO do Raspberry Pi que o sensor vai retornar o pulso Echo, possui a voltagem de 3.3. Caso a tensão não seja dividida, o pino GPIO queima e para de funcionar. O primeiro resistor de 1K Ohms e o segundo de 2.2K Ohms, resultando no seguinte circuito:

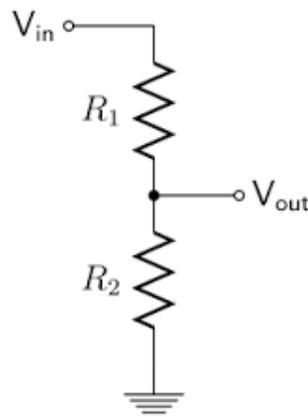


Figura 13. Circuito de divisão de tensão. Fonte: Autor

Nesse caso a tensão de saída  $V_{out}$  não foi exatamente 3.3V, e sim 3.43V, mas para o funcionamento do sistema essa margem de erro foi aceitável e o cabo jumper foi ligado entre os dois resistores na porta 18 GPIO. Segue o cálculo realizado desses valores:

$$\frac{V}{5} = \frac{R_2}{R_1 + R_2}$$

$$\frac{V}{5} = \frac{2200}{3200}$$

$$V = 3.4375$$

Depois de todas as ligações feitas obteve-se o seguinte circuito montado:

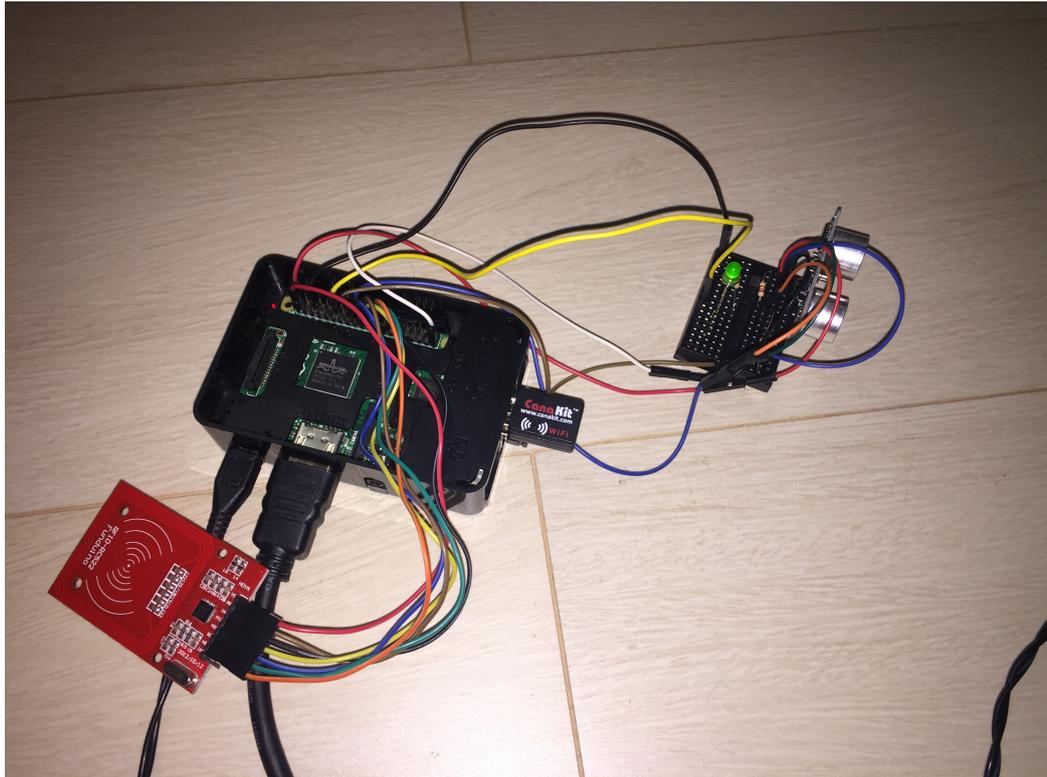


Figura 14. Circuito final montado. Fonte: Autor.

Com o circuito já montado foi possível iniciar a programação da lógica do script. O sensor faz a leitura do tempo de viagem da onda ultrassônica, e caso a leitura seja a padrão (sem um carro detectado), o LED que foi implementado na seção anterior fica aceso indicando a vaga livre; mas caso contrário, a presença é identificada e o LED apaga. O script LED.py tem o seguinte código parcial:

```
import RPi.GPIO as GPIO
import time

TRIG = 23
ECHO = 24
LED = 18

GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
GPIO.setup(LED, GPIO.OUT)

GPIO.setup(TRIG, GPIO.OUT)
GPIO.setup(ECHO, GPIO.IN)
```

```

GPIO.output(TRIG,False)
time.sleep(3)
GPIO.output(TRIG,True)
time.sleep(0.00000001)
GPIO.output(TRIG,False)

while True:

    while GPIO.input(ECHO)==0:
        trigger = time.time()
    while GPIO.input(ECHO)==1:
        echo = time.time()

    tempo = echo - trigger
    d = tempo*17150
    d = round(d, 2)

    if (d>5):
        GPIO.output(LED,GPIO.HIGH)
        print "LED ON"
    else:
        GPIO.output(LED,GPIO.LOW)
        print "LED OFF"

    print "Distancia = ",d, "cm"

```

Como pode-se verificar, o código faz a conversão do tempo de viagem da onda para uma distância real em centímetros da seguinte maneira:

$$V = \Delta s / \Delta t$$

$$V_{som} = 340m/s$$

$$34000 = \frac{\Delta s}{\frac{\Delta t}{2}}$$

$$\Delta s = \textit{distancia lida}$$

Por fim, foi preciso utilizar a função cron do Linux para que o script realizasse essa leitura recursivamente. O crontab é uma tabela de processos que devem ser executados recursivamente dentro de um determinado período. Para isso foi realizado o seguinte comando:

```
crontab -e
```

Com isso foi possível editar a tabela, incluindo o script LED.py para ser executado a cada minuto:

```
*/1 * * * * * /home/pi/documents/LED.py
```

Depois de incluído, foi possível verificar as tarefas incluídas no crontab através do comando:

```
crontab -l
```

Além dessa confirmação, foi instalado um programa com interface gráfica disponível no Raspberry Pi para facilitar o gerenciamento das tarefas através do comando:

```
sudo apt-get install gnome-schedule
```

### 3.3.4. Implementação do RFID

Para entender melhor o funcionamento e a viabilidade do sistema de RFID, foi realizado um teste, utilizando novamente o Raspberry Pi com o módulo Mifare MFRC522.

O Leitor de RFID foi conectado ao Raspberry Pi e uma tag passiva foi utilizada para a validar as informações. Nesse caso, obteve-se o mapa de pinos do leitor e as ligações foram feitas da seguinte maneira [17]:

Tabela 1. Pinos do Leitor RFID. Fonte: Autor.

Nome	Pino utilizado
VCC	Pino 1 (3.3V)
RST	Pino 22 (GPIO)
GND	Pino 25 (Ground)
MISO	Pino 21 (GPIO)
MOSI	Pino 19 (GPIO)
SCK	Pino 23 (GPIO)
NSS	Pino 24 (GPIO)
IRQ	Não foi utilizado

Para a comunicação entre o Raspberry Pi e o módulo, foi utilizado o protocolo SPI. Para possibilitar a comunicação, habilitou-se o módulo de comunicação SPI nas configurações do Raspberry Pi.

Depois de verificado o funcionamento do barramento SPI, é necessário utilizar um módulo no Python para realizar a leitura e escrita. O primeiro passo foi instalar o python-dev com o comando:

```
sudo apt-get install python-dev
```

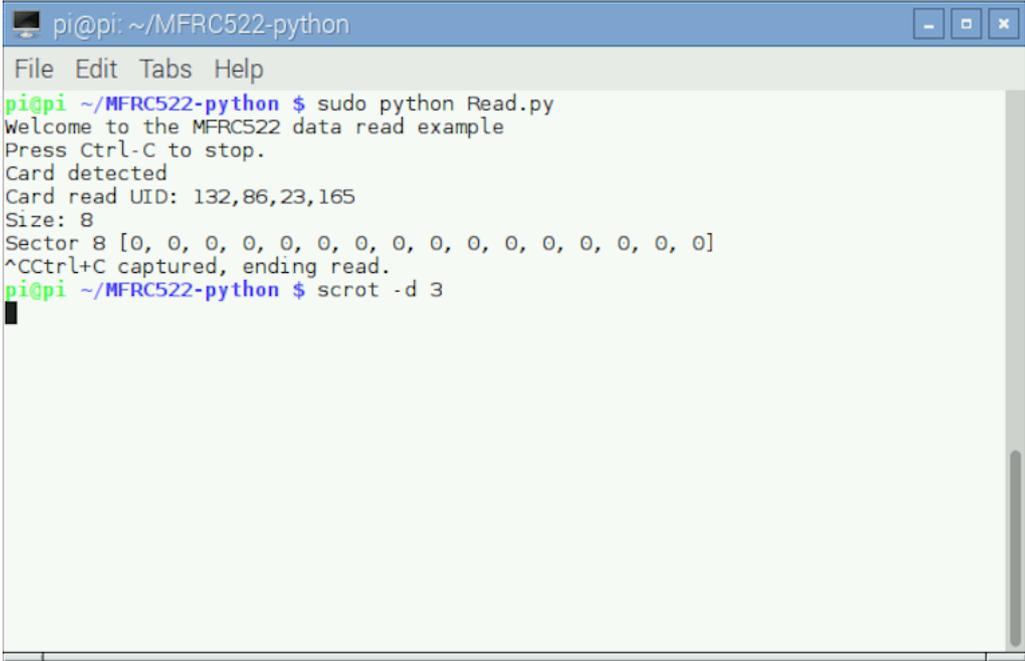
Após a instalação do python-dev, foi utilizado como base do trabalho módulos python de comunicação SPI escritos pelo usuário lthiery do repositório github. Para isso foi utilizado o comando git clone para copiar o repositório e realizar a instalação com os seguintes comandos [24]:

```
git clone https://github.com/lthiery/SPI-Py.git
cd SPI-Py/
sudo python setup.py install
```

Depois da configuração da parte de comunicação do barramento, prosseguiu-se para a instalação da biblioteca MFRC522-python [24]:

```
git clone https://github.com/mxgxw/MFRC522-python.git
cd MFRC522-python/
```

Com isso, já é possível utilizar o módulo de escrita e leitura. Para ler a tag utilizada no experimento rodou-se o programa Read.py, obtendo a seguinte saída:



```
pi@pi: ~/MFRC522-python
File Edit Tabs Help
pi@pi ~/MFRC522-python $ sudo python Read.py
Welcome to the MFRC522 data read example
Press Ctrl-C to stop.
Card detected
Card read UID: 132,86,23,165
Size: 8
Sector 8 [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
^C^C captured, ending read.
pi@pi ~/MFRC522-python $ scrot -d 3
```

Figura 15. Saída do programa Read.py. Fonte: Autor.

O funcionamento do RFID foi comprovado, e então obteve-se a informação de identificação da tag, uma sequência de 5 números entre 0 e 255 ao fazer a primeira alteração do código, assim foi possível visualizar o último número da sequência que no caso da tag utilizada era [132, 86, 23, 165, 96].

Com a informação do UID, pode-se fazer alterações no código Read.py para atender à tag especificamente, realizando autenticação, conectando ao banco de dados e inserindo informações, assim foi realizado a simulação de um controle de acesso, e a utilização do banco de dados.

### 3.3.5. Instalação do servidor LAMP

Primeiramente foi realizada a atualização do apt-get do Raspbian com o seguinte comando:

```
sudo apt-get update && sudo apt-get upgrade
```

Depois iniciou-se a instalação dos servidores Apache, MySQL e PHP. O servidor web Apache2 foi o primeiro a ser instalado, com o seguinte comando:

```
sudo apt-get install apache2 -y
```

Após a instalação, foi possível verificar o funcionamento do servidor web através do endereço <http://localhost> no Epiphany web browser. Em seguida, foi realizada a instalação do PHP e de pacotes de biblioteca do apache para o funcionamento adequado do servidor com o comando:

```
sudo apt-get install php5 libapache2-mod-php5 -y
```

Por último foi instalado o banco de dados MySQL com o comando:

```
sudo apt-get install mysql-server php5-mysql -y
```

Durante a instalação foi escolhido o nome de usuário e senha do banco de dados para obter acesso e criar as devidas tabelas.

### 3.3.6. Desenvolvimento do site

Para o desenvolvimento da página web foi utilizada a plataforma do Wordpress. Antes de iniciar o processo de instalação do wordpress foi necessário remover todos os arquivos localizados no diretório /var/www/html , que possuem os arquivos da página web padrão do servidor com o comando:

```
cd /var/www/html/  
sudo rm *
```

Após a remoção dos arquivos a instalação do wordpress foi efetuada:

```
sudo wget http://wordpress.org/latest.tar.gz  
sudo tar xzf latest.tar.gz  
sudo mv wordpress/* .  
sudo rm -rf wordpress latest.tar.gz
```

Após a instalação, o diretório /var/www/html ficou com todos os arquivos da página criada pelo wordpress, como pode ser visto na seguinte imagem:

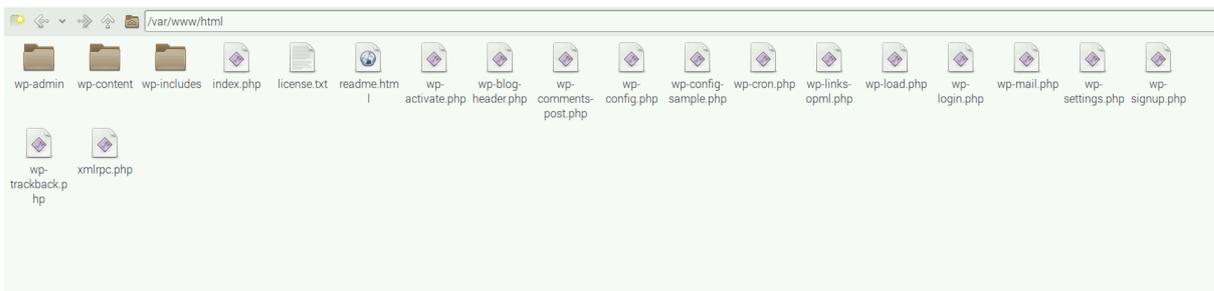


Figura 16. Arquivos da página Web. Fonte: Autor

Depois disso, a ferramenta phpmyadmin foi instalada para facilitar a utilização do banco de dados MySQL. A ferramenta escrita em PHP possibilita o gerenciamento do banco de dados MySQL quando acessado pelo web browser como pode ser verificado na seguinte figura:

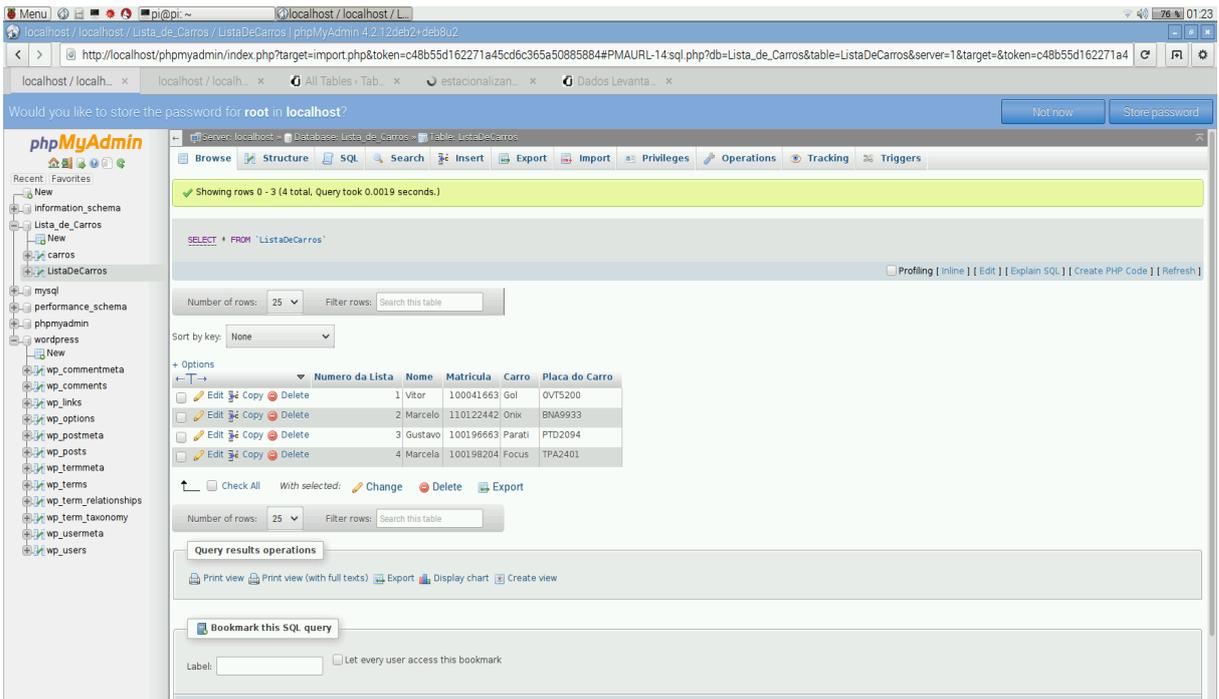


Figura 17. PHP My Admin. Fonte: Autor.

Em seguida, acessou-se o Wordpress para configurar a página web. A página inicial com a apresentação do conceito, contato, vista do Google Maps do estacionamento utilizado e demais informações, encontra-se na figura abaixo:



Figura 18. Pagina Inicial. Fonte: Autor.

Pensando na possível implementação de um sistema de geolocalização utilizando um módulo SIG no projeto, adicionou-se o mapa referente ao estacionamento utilizado no estudo de caso pelo Google Maps:

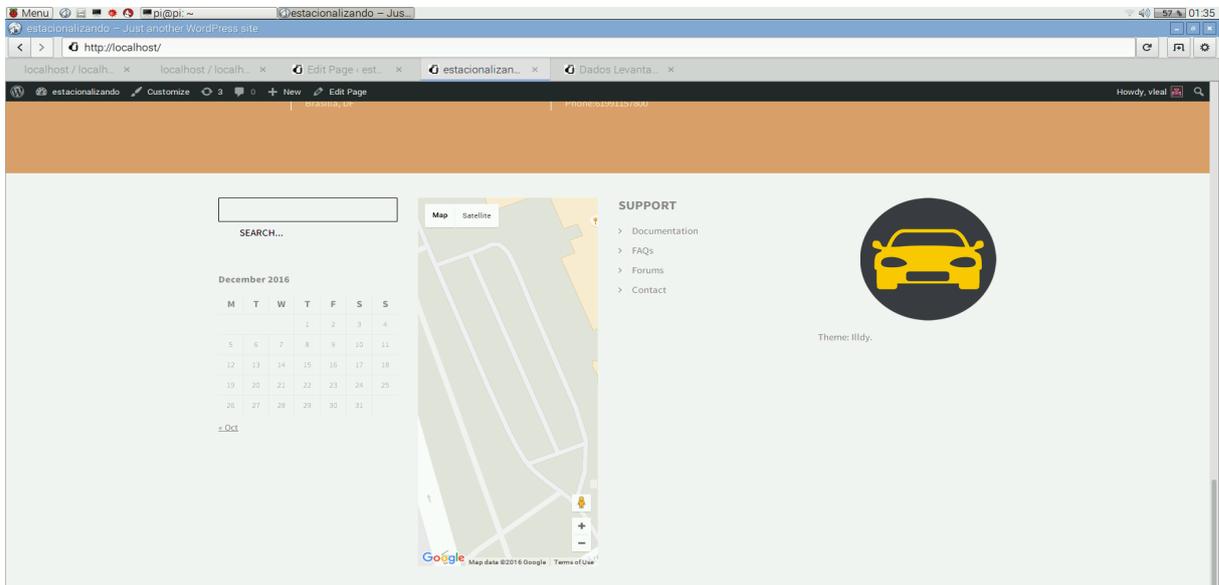


Figura 19. Pagina Inicial 2. Fonte: Autor.

### 3.3.7. Construção da Maquete

O projeto foi realizado tendo em vista sua aplicação no campus Darcy Ribeiro da Universidade de Brasília. Sendo assim, foi realizada uma maquete do estacionamento da Faculdade de Tecnologia da UnB para tornar os testes mais fidedignos e apresentar o conceito.

A vista aérea do estacionamento em questão pode ser observada na seguinte imagem:



Figura 20. Vista aérea do estacionamento da Faculdade de Tecnologia. Fonte[25]

Para a maquete, construiu-se uma parte com proporções menores e mais realista, onde foram colocadas miniaturas de pessoas, carros, postes, galhos e árvores; e uma segunda parte com uma proporção maior para a simulação do sistema.

## **4. PROPOSTA**

### **4.1. ESTUDO DE CASO**

O projeto tinha como principal proposta estudar um sistema de identificação veicular de baixo custo com a utilização de sensores para criar um estacionamento inteligente. Vale ressaltar que alguns projetos similares já foram desenvolvidos, porém, na prática, estacionamentos com esses recursos não são comuns, principalmente devido ao custo de implementação.

O estudo de caso foi realizado em uma escala menor, mas utilizando recursos idênticos ou muito similares ao utilizado na escala real. A proposta do projeto foi estudar a tecnologia utilizada no sistema, realizar a implementação do sensor de presença e RFID e sobretudo estudar os temas, verificando a viabilidade do sistema e o funcionamento final do conceito focando na utilização de componentes baratos.

O teste de conceito utilizou o Raspberry Pi 2, mas vale destacar que o Raspberry Pi Zero, que custa apenas cinco dólares, atenderia aos requisitos do projeto. O Raspberry Pi Zero possui o mesmo valor de um micro controlador 8051 e outros PICs utilizados em projetos similares avaliados, além de apresentar uma melhor performance e maior possibilidade de recursos a serem implementados como o servidor LAMP, controle de um número maior de componentes e a facilidade de utilização do sistema.

O sensor sonar utilizado no projeto possui o valor de aproximadamente dez reais, que é seis vezes menor do que sensores de indutância que foram utilizados em projetos similares. Os resultados das medições foram satisfatórios, não ocorreram erros significativos.

Outra análise importante diz respeito ao sistema RFID, uma vez que o sistema utilizado para os testes trabalha na frequência HF, que resulta em uma possibilidade de leitura mais próxima. Uma outra opção mais apropriada para o projeto real é a utilização de frequências maiores (UHF) para que seja possível obter um alcance maior, tornando a solução mais próxima do ideal, já que apenas um leitor faria a cobertura total do espaço. O preço de um leitor comum, que trabalha na frequência de 13.56MHz custa em média cento e cinquenta reais, já o leitor que trabalha nas frequências maiores possui um preço de aproximadamente quatro mil reais.

Um fato relevante no tema do RFID é o SINIAV (Sistema de identificação automática de veículos), programa do DENATRAN (Departamento Nacional de Transito) para

identificação utilizando a tecnologia de identificação por rádio frequência em todos os automóveis. O programa surgiu em 2006, porém só teve início em 2016. O identificador será utilizado para outras soluções, e provavelmente vai abrir maiores espaço para soluções como a desenvolvida no projeto. Pode-se concluir que o custo das tags seja contabilizado fora do projeto, já que será em breve uma obrigatoriedade.

## 4.2. RESULTADOS E DISCUSSÃO

O resultado final do projeto foi positivo. O estudo de caso foi concluído com um estudo sobre as tecnologias RFID e sensores, bem como a utilização desses módulos na prática com o sistema completo, incluindo leitura da presença do veículo com o sensor sonar, indicação com o LED, sistema RFID, páginas web com as informações iniciais e dados levantados pelo RFID, e demonstração através da maquete realizada.

O sensor sonar permitiu realizar diversas leituras com baixíssimos índices de erros. A indicação do LED de uma vaga livre (sem nenhum corpo na frente) pode ser observada na seguinte imagem:

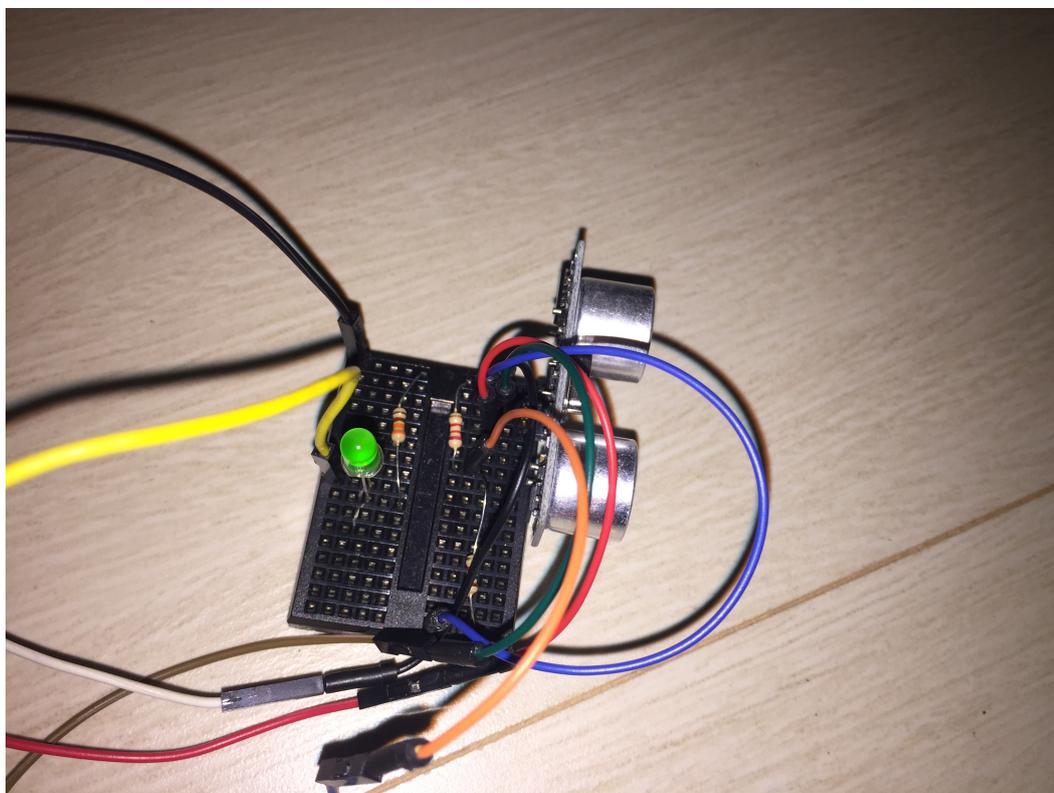


Figura 21. Circuito com o sistema de presença indicando vaga livre. Fonte: Autor

Alguns erros que ocorreram durante o processo de implementação foram:

- Acendimento do LED por mal contato;
- Erro de leitura do sensor por conta da margem de erro;
- Problemas no desenvolvimento da interface web
- Problemas com escrita no banco de dados e;
- Problemas na leitura e escrita da tag RFID.

As falhas citadas foram contornadas e a versão final do sistema se mostrou confiável, com leituras corretas, e o LED indicou corretamente a disponibilidade da vaga na maquete.

Além disso o site foi implementado visando uma integração maior com o usuário, já que ao utilizar o Raspberry no projeto, pode-se hospedar o site disponibilizando as informações de funcionamento do estacionamento pelas leituras do RFID. A demonstração da tabela com as informações dentro da página web pode ser visualizada a seguir:

**DADOS LEVANTADOS**

Show  entries

Search:

1	Vitor	100041663	Gol	OVT5200
2	Marcelo	110122442	Onix	BN49933
3	Gustavo	100196663	Parati	PTD2094
4	Marcela	100198204	Focus	TPA2401

Edit

Showing 1 to 3 of 3 entries

◀ Previous Next ▶

Figura 22. Tabela com os dados na página Web. Fonte: Autor.

Novamente, ao verificar o funcionamento da página web no próprio Raspberry Pi, confirma as possíveis soluções que podem ser desenvolvidas como notificações no e-mail cadastrado caso o carro seja retirado sem a validação da tag, um sistema completo de cadastro, entre outras. Porém, com as informações do sistema, já é possível realizar o controle de acesso (utilizando as informações para a liberação de uma catraca) e, com os indicadores das vagas, já seria possível realizar um mapa com indicação total de todas as vagas ocupadas ou livres.

A maquete foi realizada para uma melhor visualização do conceito sendo totalmente integrada ao sistema final:



Figura 23. Maquete do estacionamento da FT. Fonte: Autor

Diversos testes foram realizados do sistema já instalado na maquete. Como o sensor de presença utilizado, depende da distância, somente depois da instalação, os valores foram ajustados para garantir o funcionamento correto das leituras. O primeiro teste foi realizado sem nenhum corpo entre o sensor e a maquete, depois foi realizada as medições com dois tipos de carros. Os valores obtidos estão na seguinte tabela:

Tabela 2. Medições de distância para lógica de presença. Fonte: Autor.

	Sem nenhum objeto	Carro Pequeno	Carro Médio
<b>Medida 1</b>	5.81	5.45	4.89
<b>Medida 2</b>	5.80	5.45	4.90
<b>Medida 3</b>	5.77	5.44	4.91
<b>Medida 4</b>	5.82	5.42	4.90
<b>Medida 5</b>	5.81	5.43	4.90
<b>Medida 6</b>	5.80	5.40	4.93
<b>Medida 7</b>	5.84	5.39	4.92
<b>Medida 8</b>	5.83	5.44	4.90
<b>Medida 9</b>	5.80	5.41	4.88
<b>Medida 10</b>	5.82	5.40	4.91
<b>Média dos valores obtidos</b>	5.81	5.423	4.904

Com as médias obtidas, pode-se concluir o trabalho utilizando o sensor e o RFID com a lógica adaptada para as medidas realizadas na escala da maquete. Todas as medições acima de 5,5 demonstra a disponibilidade da vaga, enquanto as medições menores demonstram a utilização da vaga.

## 5. CONCLUSÃO

Pode-se concluir que o presente projeto conceito, apesar de ser um teste, atende aos propósitos iniciais, sendo uma forma viável e de baixo custo capaz de contribuir para comunidade com uma solução de estacionamento inteligente com ferramentas de identificação e sensores, que além de ajudarem os motoristas, também auxiliam no maior controle do estacionamento por parte dos supervisores, tornando o local mais seguro e eficiente.

As leituras realizadas tiveram uma precisão altíssima, foram realizadas leituras recursivas, indicando corretamente a disponibilidade da vaga. Além disso, o sistema de identificação RFID realizado, apesar de conter somente uma tag e conseqüentemente apenas um cadastro, foi realizado por completo, identificando o usuário e colocando essa informação no banco de dados. A tag cadastrada, quando lida, libera o acesso ao estacionamento. Ao retirar o cadastro da tag e tentar o acesso, o acesso é negado.

O projeto de tamanho real, se mostrou viável, utilizando os componentes de sensor já utilizados e adaptando o sistema de RFID para leituras mais distantes. Com as indicações luminosas é possível mapear todas as vagas e disponibilizar essa informação em conjunto com os dados coletados pelo RFID.

### **Trabalhos Futuros**

Por se tratar de um projeto conceito, a tentativa de implementação do projeto levando a proposta para possíveis interessados é um trabalho que ainda deve ser realizado.

Ainda assim, algumas melhorias importantes podem ser implementadas. Aqui estão listados os pontos principais que devem ser trabalhados em uma segunda versão do projeto:

- Melhorias gerais do circuito montado para desenvolver um produto;
- Inclusão de um sistema de georeferenciamento (SIG);
- Realização de um aplicativo para iOS e Android;
- Interface para cadastro de Tags;
- Implementar notificações para usuários e administradores.
- Implementar a catraca para realizar o controle de acesso físico

## REFERÊNCIAS BIBLIOGRÁFICAS

- [1] GDF/SSP/DETRAN/DIRTEC, Frota de Veículos Registrados no Distrito Federal, 2016.
- [2] Automação Industrial. Disponível em: <https://www.comatreleco.com.br/automacao-industrial-historia/>. Acesso em: 02/12/2016
- [3] Automação Residencial e RFID. Disponível em: <http://fluxoconsultoria.poli.ufrj.br/blog/tecnologia-informacao/automacao-residencial/> . Acesso em: 02/12/2016
- [4] SCHMIDT, Maik. Raspberry Pi, a quick start guide. 2nd edition. The pragmatic programmers, 2014.
- [5] MONK, Simon. Raspberry Pi Cookbook. O`Reilly Media,2013
- [6] The Raspberry Pi Foundation. Documentação do Raspberry Pi. Disponível em <https://www.raspberrypi.org/documentation/usage/gpio-plus-and-raspi2>. Acesso em: 15/11/2016.
- [7] GARRELS, Machtelt. Introduction to Linux, a Hands on Guide. Version 1.18 20050524 Edition. Machtelt Garrels , 2002.
- [8] ROSEBROCK, Eric & FILSON, Eric. Setting Up LAMP: Getting Linux, Apache, MySQL and PHP working together. Sybex, 2004.
- [9] BORGES, Luiz E. Python para Desenvolvedores. 2a edição.
- [10] COLEMAN, Christopher. An Introduction to Radio Frequency Engineering. Cambridge University Press, 2004.
- [11] YOUNG, Hugh D. & FREEDMAN, Roger A. Física III Eletromagnetismo. 12ª edição. Pearson, 2009.
- [12] VIERA, Angel F. G., VIERA, Sonia D. G. & VIERA, Lourdes E. G. Tecnologia de Identificação por Radiofrequência: fundamentos e aplicações em automação de bibliotecas. Enc. Bibli. R. Eletr. Bibliotecon. Ci. Inf., Florianópolis, n. 24, 2007.

- [13] GLOVER, B. & BHATT, H. RFID Essentials. O'Reilly, Sebastopol, 2006.
- [14] FINKENZELLER, Klaus. RFID Handbook. Third Edition. Wiley, 2010\*
- [15] AGARWAL, Parag e GESUALE, Brian. RFID : Read My Chips!. PiperJaffray, 2004.
- [16] Produto do Kit RFID. Disponível em: <http://www.autobotic.com.my/rc522-rfid-s50-fudan-card-s50-fudan-card-keychain>. Acesso em: 01/12/2016
- [17] MFRC522 product data sheet. Disponível em: <http://www.sunrom.com/p/mifare-rfid-readerwriter-1356mhz-rc522>. Acesso em: 25/11/2016
- [18] PATSKO, Luís F. Tutorial Aplicações, funcionamento e utilização de sensores. Maxwell Bohr, 2006.
- [19] AGARWAL, Parag e GESUALE, Brian. RFID : Read My Chips!. PiperJaffray, 2004.
- [20] THOMAZINI, Daniel e ALBUQUERQUE, Pedro U. B. de. Sensores Industriais – Fundamentos e Aplicações. 4a Edição. Editora Érica, 2009.
- [21] Ultrasonic Ranging Module HC-SR04. Disponível em: <http://www.micropik.com/PDF/HCSR04.pdf>. Acesso em 08/10/2016
- [22] Instalação do Noobs. Disponível em: <https://pplware.sapo.pt/linux/noobs-v1-3-3-escolha-qual-o-sistema-a-instalar-no-raspberry-pi/> .Acesso em: 22/10/2016
- [23] Projeto de acendimento de LED. Disponível em: <https://pplware.sapo.pt/tutoriais/raspberry-pi-como-acender-um-led/> . Acesso em 20/11/2016
- [24] How to use RFID RC522 on Raspberry Pi. Disponível em: [https://www.sunfounder.com/wiki/index.php?title=How to Use an RFID RC522 on Raspberry Pi](https://www.sunfounder.com/wiki/index.php?title=How_to_Use_an_RFID_RC522_on_Raspberry_Pi). Acesso em 29/11/2016
- [25] GOOGLE MAPS. [Vista aérea do estacionamento da Faculdade de Tecnologia]. 2016. Disponível em: <https://www.google.com/maps/>.Acesso em: 20/11/2016..