

Relatório Final: FGParking

Camila Carneiro Ferrer Santos

Programa de Engenharia Eletrônica
email: camila.c.ferrer@gmail.com

Matrícula: 14/0080619

Jéssica de Souza Alves

Programa de Engenharia Eletrônica
email: jessicaturunenn@gmail.com

Matrícula: 14/0042784

Abstract—O projeto consiste em validar a entrada do veículo por reconhecimento de imagem da placa: ao entrar no estacionamento um sensor de movimento aciona uma câmera que captura a imagem da placa do veículo e, após processado os dados da placa, o usuário tem acesso à quantidade de vagas no estacionamento por um aplicativo/site. Além de recebe uma notificação pelo aplicativo, assim que entra informando o horário de entrada do veículo e, assim que sai, informando o horário de saída do estacionamento.

Index Terms—Sistemas Embarcados, estacionamento.

1 REVISÃO BIBLIOGRÁFICA

A Pesquisa Distrital por Amostra de Domicílios elaborada pela CODEPLAN mostra que quatro em cada 10 brasileiros usam o carro para chegar ao trabalho. [1] O que torna o espaço público cada vez mais disputado pelos motoristas, pois os veículos necessitam circular e estacionar antes de atender às necessidades de quem os dirige.

O estacionamento é componente essencial na qualidade de vida e do transporte urbano, sendo assim fundamental a qualquer Política de Mobilidade Urbana estabelecer programas de disponibilização de diferentes níveis de oferta de estacionamento.

A maioria dos condutores de veículos já encontrou dificuldade em encontrar estacionamento em áreas centrais urbanas, trata-se de uma questão global. Nesses locais os veículos estão em competição contínua a fim de satisfazer suas necessidades, promovendo uma verdadeira batalha entre veículos que buscam circular e estacionar. [2]

Além da dificuldade de encontrar local para estacionar, outro problema muito grande é a segurança dos estacionamento rotativos. Um

exemplo muito claro desse problema é o estacionamento da Universidade de Brasília, campus do Gama, onde diversos assaltos já aconteceram desde sua inauguração. Fato que levou os a adesão de 95% da comunidade acadêmica local em 2016 a iniciar uma paralisação para pedir mais segurança no local. [3]

Em tempos de integração e conectividade, adaptar sistemas já utilizados com as novas tecnologias se torna essencial, visto a necessidade de atualização nas prestações de serviços aos usuários.

Uma possível aplicação para uma realidade cotidiana seria a utilização de um sistema que consiste em controlar a entrada e a saída de veículos em um estacionamento e armazenar em um banco de dados as informações necessárias para monitoramento do mesmo.

Podendo-se existir a possibilidade de pagamento pelo serviço apenas pelo seu smartphone possibilitando maior controle para o usuário e economia para os estabelecimentos que usarem esse sistema, pois não haveria necessidade de utilização de ticket ou cartão de estacionamento.

Utilizando sistemas embarcados é possível, de maneira efetiva, criar uma solução para

os problemas associados aos sistemas de estacionamentos atuais e populares.

2 JUSTIFICATIVA

Investir em sistemas de automação que utilizam tecnologia para aumentar o controle e a segurança de determinado espaço, ao identificar veículos a partir de fotografias tiradas no momento de entrada e saída do mesmo, pode trazer uma grande mudança nas operações realizadas. Além de aumento do resultado operacional, já que a utilização de pessoas para verificação da atividade seria desnecessária e a utilização de cartões de acesso também.

A fim de trazer o conceito de sistemas embarcados para estacionamentos, este projeto busca uma solução de um problema prático e presente na vida acadêmica da FGA quanto a organização e segurança de seu estacionamento.

3 OBJETIVOS

O projeto destina-se a implementar um sistema de apoio computadorizado utilizando uma Raspberry Pi 3 Model B para controle de entrada e saída de veículos do estacionamento da Universidade de Brasília Campus Gama. Assim, tendo-se os dados de entrada e saída dos veículos visitantes do Campus, o usuário do estacionamento tem maior segurança no mesmo.

4 REQUISITOS

Para elaborar as funcionalidades do sistema, foram elencados os seguintes requisitos:

4.1 Controle de Entradas

REQ01: A câmera do sistema deve permanecer em Standby até acionamento de botão na entrada do estacionamento.

REQ02: O sistema deve informar ao usuário cadastrado quando seu carro entra no estacionamento.

REQ03: O sistema deve decrementar em uma unidade a quantidade de vagas disponíveis.

4.2 Controle de Saídas

REQ04: A câmera do sistema deve permanecer em Standby até acionamento de botão na saída do estacionamento.

REQ05: O sistema deve informar ao usuário cadastrado quando seu carro sai do estacionamento.

REQ06: O sistema deve incrementar em uma unidade a quantidade de vagas disponíveis.

4.3 Controle de Lotação

REQ07: O sistema deve especificar a lotação do estacionamento no aplicativo para ilustrar a ocupação do estacionamento.

4.4 Cadastro de Usuários

REQ08: O sistema deve requerer as seguintes informações na hora do cadastro: placa, modelo, ano de fabricação e cor do veículo, nome completo, RG e CPF do usuário responsável.

REQ09: O sistema permite a inserção, remoção e alteração de usuários cadastrados pelo próprio usuário.

5 BENEFÍCIOS

Dentre os benefícios envolvidos no projeto, destacam-se:

5.1 Economia e visão sustentável

Com a utilização exclusiva do sistema pelos formatos digitais, não se tem o gasto com as impressões realizadas na entrada do estacionamento e para o comprovante de pagamento. Com isso, há uma economia nas impressões, além da diminuição ou exclusão total dos locais de recebimento de tickets (os quais se localizam dentro do estabelecimento, afim apenas de realização dos pagamentos do serviço). A economia está vinculada tanto as impressões quanto os custos relacionados à esses guichês de recebimento. No entanto também tem-se o lado sustentável na implantação do sistema. O consumo de papéis e produção de lixo são reduzidos ou extintos, evitando e eliminando o desperdício de papel.

5.2 Segurança

O sistema dá a possibilidade da empresa responsável pelo estacionamento - a prestadora de serviços, de implantar um sistema de segurança atrelado as informações das placas dos carros. Por exemplo, um banco de dados com os registros de entrada e saída de cada veículo, sobretudo, o maior controle dos veículos que estão no estacionamento e dando assim maior sensação de segurança também para os usuários do serviço. Isso possibilita também maior segurança vista pelo usuário, o qual sabe que a prestadora de serviços tem controle dos veículos que entram e saem do estacionamento. Também como forma de segurança, ao entrar no estabelecimento, o usuário recebe uma notificação no celular. E nas saídas este também é notificado - assim o usuário sabe se seu veículo ainda está no estacionamento, para maior prevenção de roubos.

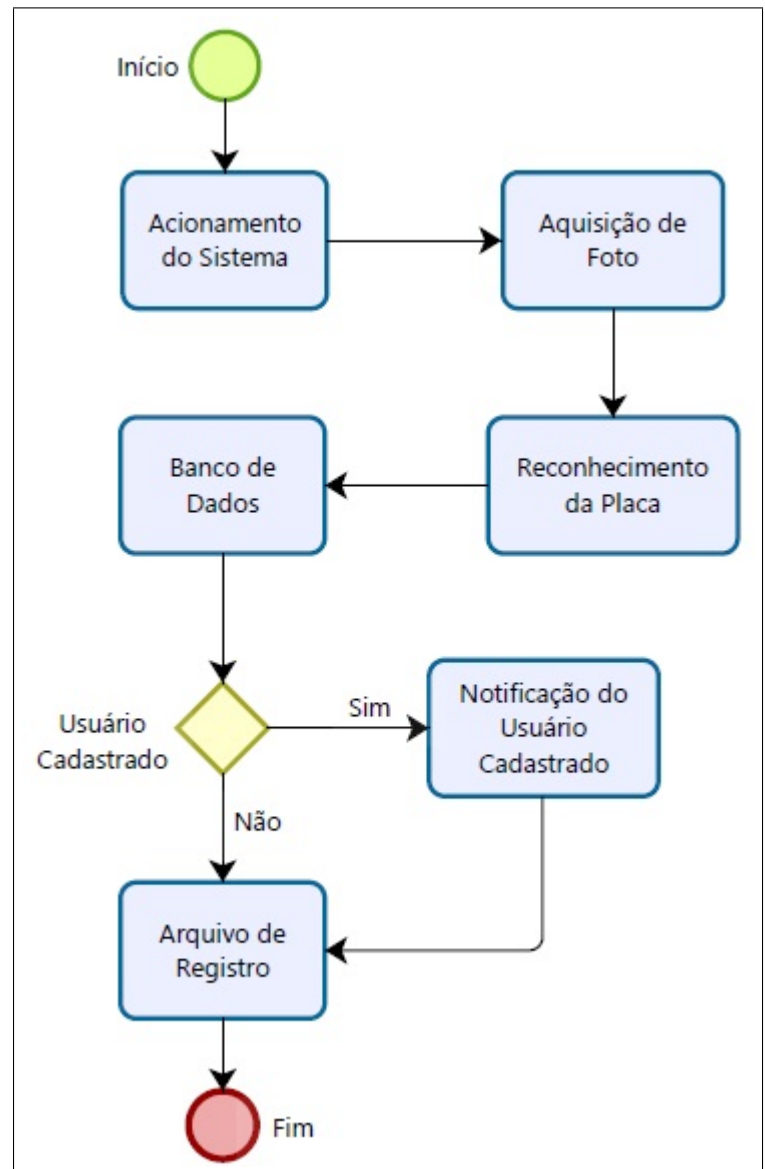
5.3 Praticidade

A partir de um sistema de pagamento em que o usuário não precisa procurar os pontos ou guichês de pagamento do mesmo, há uma facilidade e ganho de tempo na realização do serviço. Isso, tendo em vista que o usuário pode realizar seu pagamento direto do seu smartphone, não precisando enfrentar filas para pagamento, e até evitando transtornos na saída do estabelecimento, por exemplo, caso já esteja na saída e tenha lembrado só então de realizar o pagamento já estando no automóvel, o usuário não precisa sair do seu carro para realizar o pagamento. Essa praticidade é garantida pelo projeto apresentado. Seu usuário realiza os pagamentos no seu tempo e no seu próprio celular, ganhando tempo com isso e evitando transtornos na saída do estabelecimento.

6 DESENVOLVIMENTO

Primeiramente elaborou-se um diagrama de blocos (Figura 1) para entender o relacionamento entre cada processo existente no sistema e, com isso, poder modular cada processo com seus devidos insumos e artefatos. Para esse Ponto de Controle os processos a seguir foram implementados.

Fig. 1: Diagrama de Blocos



6.1 Acionamento do Sistema

Inicialmente o sistema de captura de foto seria realizado por um sensor de movimento para o disparo da câmera, logo na entrada do estacionamento. No entanto, para fins práticos utilizar o sensor de movimento para ativação da câmera não era solução viável, visto que quando um veículo ativasse o sistema as fotos poderiam sair desfocadas ou com problemas de nitidez, tendo em vista que o carro ainda estaria em movimento durante a captura da imagem. Dessa forma, adaptou-se o sistema para o disparo da câmera por um botão na

entrada do estacionamento, onde o usuário tem a necessidade de parar o veículo para a captura da imagem, garantindo assim que as fotos serão feitas com os carros sem movimento. Contudo, o usuário deve apertar um botão na entrada do estacionamento e a foto assim ser retirada.

6.2 Aquisição de Foto

A aquisição das fotos é feita por uma câmera posicionada de forma a capturar a placa do carro. Assim, o usuário ao entrar no estacionamento, aperta um botão, a foto é capturada com o registro da data e horário da entrada. O botão para acionamento da câmera é acionado pelos pinos GPIO da RaspberryPi, onde a entrada para os comandos da biblioteca "fswebcam" são as entradas acionadas dos pinos. Para o acionamento pelos pinos GPIO, fez-se um Script que executa os comandos da biblioteca "fswebcam". A biblioteca "fswebcam" autoconfigura com qualidade e dimensão mínima as fotos retiradas por uma entrada USB (no caso, a câmera como entrada USB) da RaspberryPi.

6.3 Reconhecimento da Placa

O processo recebe como insumo a foto que registrou o momento que o usuário apertou o botão do estacionamento e a ação do veículo, caso ele tenha saído ou entrado do estacionamento. Utilizando o programa OpenALPR realiza-se o processamento de imagem para gerar algumas possibilidades de placas. OpenALPR é uma biblioteca de reconhecimento de placa de veículos de código aberto, escrita em C++ e com extensões em C#, Java, Node.js, Go e Python. Essa biblioteca analisa imagens e vídeos para identificar placas de veículos. A OpenALPR permite a execução na nuvem ou localmente, e a saída após a execução é a representação de texto dos caracteres da placa. Depois de analisar as imagens dos veículos tem-se os dados do mesmo (como a cor, a marca e o modelo), além dos caracteres da placa. Para o trabalho em questão, preferiu-se usar a execução local da biblioteca, por questões de desempenho e tempo de resposta. Por isso o único dado que o OpenALPR irá processar da imagem será a placa. Os demais dados serão

introduzidos pelo usuário. O resultado gerado pelo OpenALPR é salvo em um arquivo de texto, a partir dele se extrai a placa com maior chance de acerto. Com o código da placa extraído, realiza-se o pedido para registrar no Banco de Dados o acesso do veículo ao estacionamento.

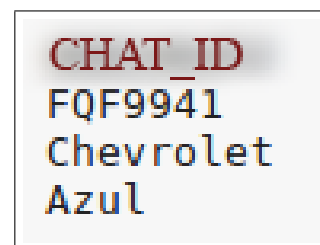
6.4 Banco de Dados

Ao reconhecer a placa do veículo, é realizada uma busca no Banco de Dados de Usuários Cadastrados para identificar se existe algum registro para aquele veículo. Antes verificar o Banco de Dados, faz-se uma busca por novos cadastros que são salvos <CHAT_ID>.txt (Figura 2) e modifica-se o arquivo .txt para que ele seja salvo com o nome <PLACA>.txt na pasta *Banco_de_Dados*. Dentro desse novo arquivo .txt serão armazenadas 4 informações em cada linha: ID do Chat, Placa do Veículo, Modelo do Veículo e Cor do Veículo. Essas informações serão utilizadas para enviar a notificação (Figura 3).

Fig. 2: Novos cadastros



Fig. 3: Dados do cadastro



6.5 Notificação do Usuário Cadastrado

Após atualizar o Banco de Dados, o programa irá notificar os usuários cadastrados. Para isso,

optou-se por automatizar o procedimento de notificação, por ser repetitivo, por meio da utilização do bot do Telegram. Como forma de comunicação com o usuário da plataforma foi selecionado o Telegram, um serviço de mensagens instantâneas que através de um bot pré implementado no mensageiro envia a notificação ao usuário em tempo real, com as entradas e saídas de acesso ao estacionamento. A escolha do aplicativo se deu pelo grande número de usuários a utilizar o mesmo, e a praticidade que um serviço de mensagens instantâneas têm, pois a partir dele não é necessário que os usuários instale outro aplicativo ou similar para a utilização do serviço. Garantindo assim comodidade ao usuário do estacionamento. Para o cadastro do Usuário, ele deverá procurar pelo bot no aplicativo do Telegram (Figura 4) e iniciar uma conversa com ele (Figura 5). Após iniciar a conversa, o usuário manda o comando *\registar* para iniciar o processo de registrar o veículo (Figuras 6 e 7). Enquanto o usuário está cadastrando o veículo, o programa rodando na RaspberryPi armazena seus dados no arquivo *<CHAT_ID>.txt*. Após a atualização do Banco de Dados, o programa procura na pasta *Banco_de_Dados* se existe algum arquivo com aquela placa registrada e, caso exista, pega o chat ID, o modelo e a cor e a ação do carro (entrar ou sair) para criar o arquivo php responsável por enviar a mensagem por meio do bot do Telegram atualizando o usuário do status do veículo (Figura 8).

Fig. 4: Busca por bot no Telegram

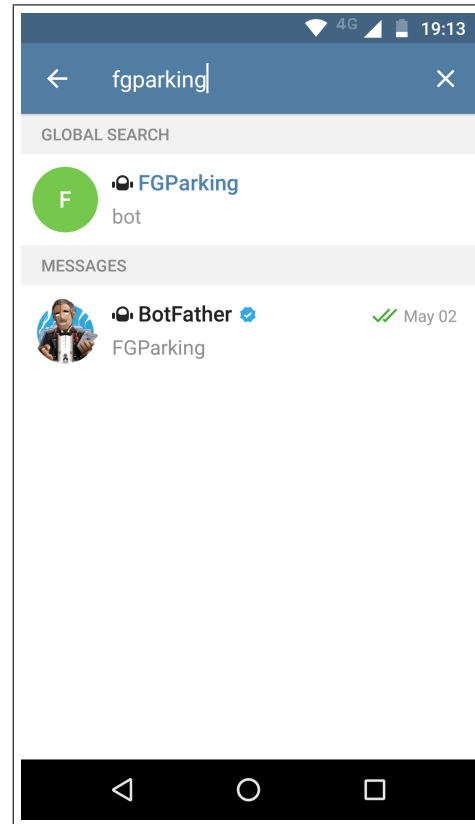


Fig. 5: Iniciar conversa com bot no Telegram

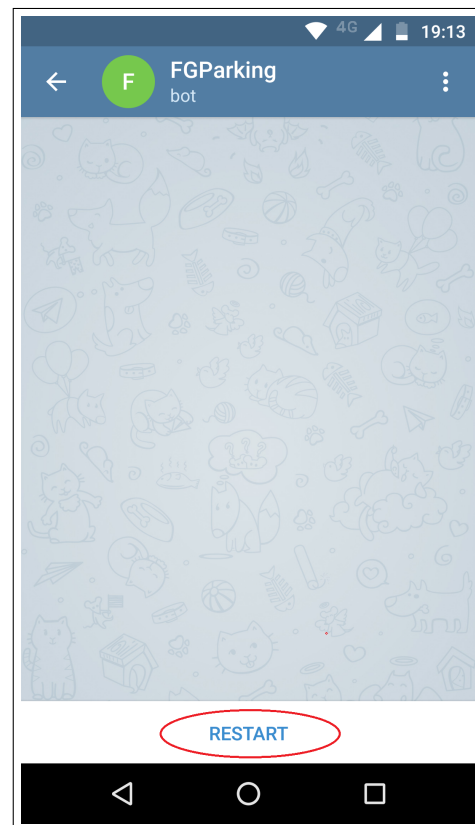


Fig. 6: Início do registro

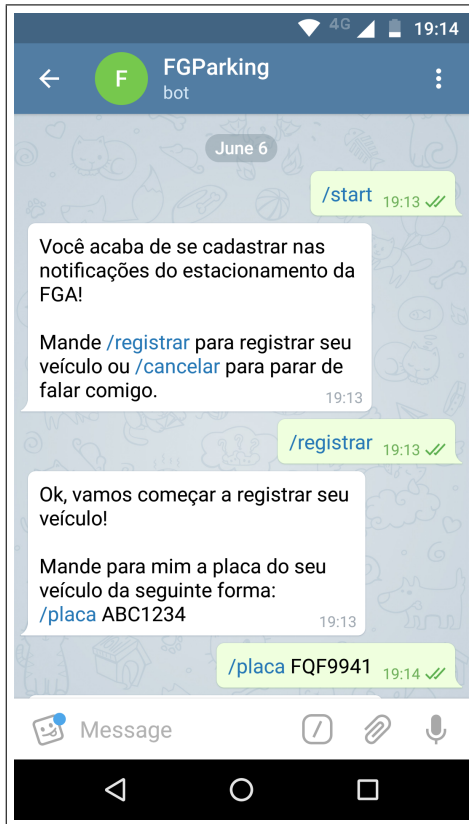


Fig. 8: Status do veículo

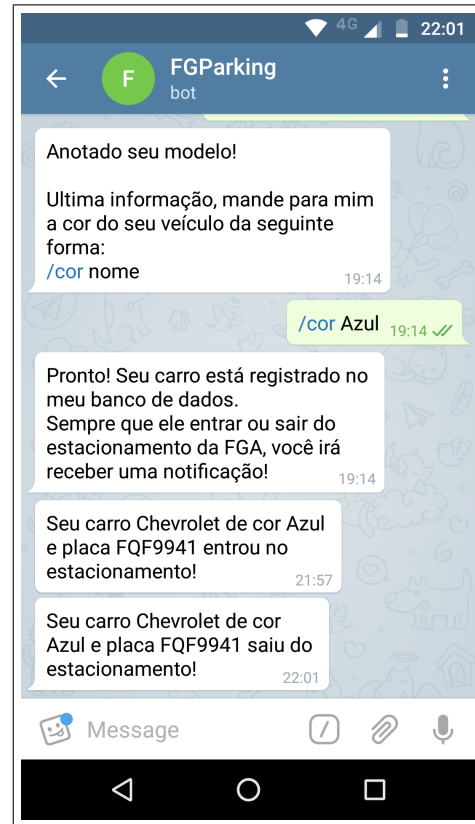
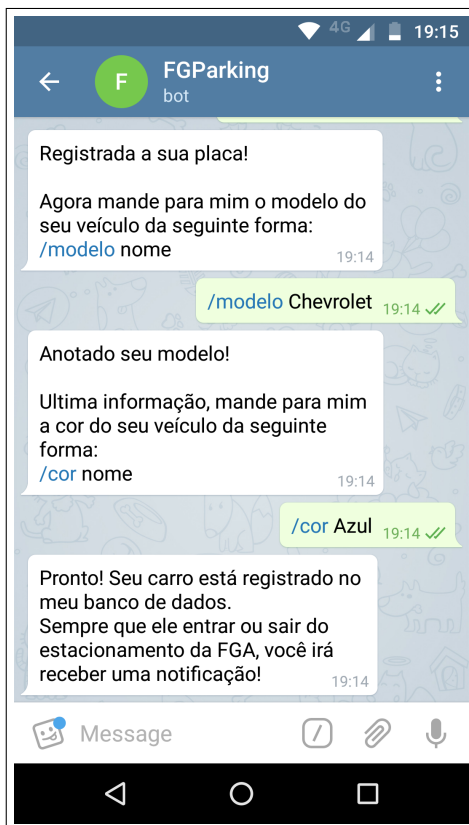


Fig. 7: Termino do registro



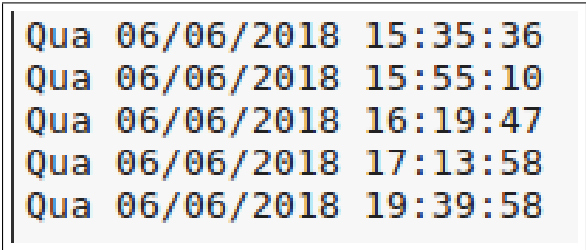
6.6 Arquivo de Registro

Caso o usuário tenha sido notificado ou caso o carro não tenha cadastro, decidiu-se registrar o acesso de veículo ao estacionamento gerando um arquivo `<PLACA>-acessos.txt` (Figura 9) para registrar o dia da semana, dia, mês e ano, além da hora, minuto e segundo que o veículo entrou e saiu do estacionamento (Figura 10). Caso o arquivo já exista, a nova entrada é colocada ao final dele. E caso o arquivo ainda não exista, ele é criado e colocado o primeiro registro nele. Ao atingir 30 registros, o mais antigo é deletado para a inserção do mais novo ao final.

Fig. 9: Arquivos do Banco de Dados



Fig. 10: Termino do registro



7 RESULTADOS

7.1 Validação

Para a validação do sistema, os critérios adotados foram o tempo de execução e resposta do sistema, além da eficiência do sistema na identificação correta das placas na entrada do estacionamento e a garantia que notificação é recebida pelo usuário.

7.1.1 Tempo de execução e resposta do sistema

Uma das preocupações na execução do sistema, estava relacionada ao tempo em que o usuário apertaria o botão de entrada do estacionamento, ao tempo do reconhecimento da imagem para então abertura da cancela. A cancela é aberta assim que a placa é processada no reconhecimento. Os testes realizados obtiveram resultados satisfatórios a esse tempo de execução (o qual seria o tempo mais crítico, visto que a cancela só iria ser aberta após o reconhecimento), sendo 2,336 segundos o tempo máximo e 2,306 segundos o tempo médio nos testes realizados.

Fig. 11: Tempo de execução da placa JEN1944



Fig. 12: Tempo de execução da placa PAD6305



Fig. 13: Tempo de execução da placa PAV2686



Fig. 14: Tempo de execução da placa PBE7976



Fig. 15: Tempo de execução da placa PUM4594



7.1.2 Reconhecimento correto da placa

Outro ponto crítico do projeto, é a identificação correta da placa. Para que o erro fosse minimizado (o erro era de 3 para 7 inicialmente), pelos testes foi observado que o posicionamento da câmera era característica fundamental para esse ponto. Assim, com um novo posicionamento e novos testes realizados, o erro associado a identificação da placa foi de 1 para 7. Outra forma de minimizar o erro estava ao utilizar a API online, direto do servidor, no entanto isso comprometia o tempo de execução do sistema.

7.1.3 Notificação ao usuário

A notificação do usuário estava diretamente ligada ao reconhecimento correto da placa. Dessa forma os testes em que o usuário não recebia a notificação, eram apenas quando a placa não era identificada ou quando o veículo não estava cadastrado. A notificação está na figura 21.

7.2 Análise Crítica

Inicialmente a notificação seria recebida por e-mail, mas ao longo do projeto percebeu-se que por Telegram a notificação seria mais prática e dinâmica.

E como citado anteriormente, o erro na identificação da placa foi resolvido com o posicionamento da câmera.

7.3 Sistema em Funcionamento

A seguir pode-se verificar o sistema em funcionamento durante apresentação do projeto finalizado.

Fig. 16: Posicionamento Webcam



Fig. 17: Conexão da Raspberry

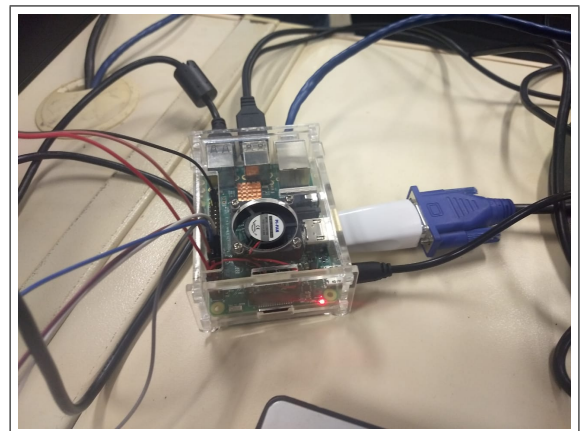


Fig. 18: Instalação dos botões de entrada e saída

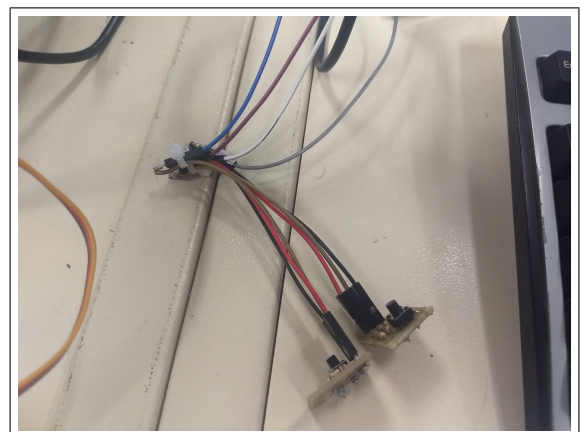


Fig. 19: Terminais em execução com a aplicação

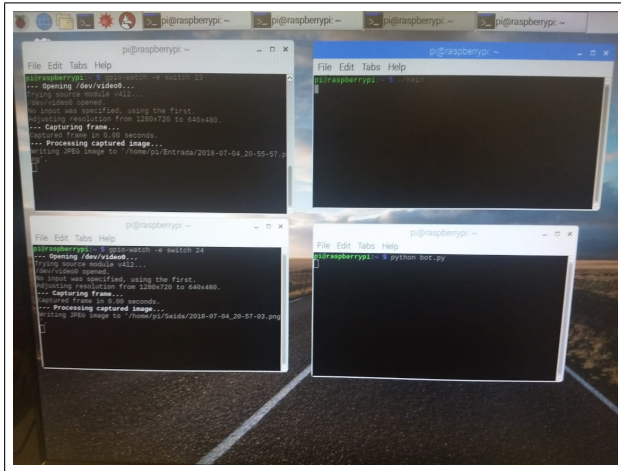


Fig. 20: Cancela acionada para passagem do veículo

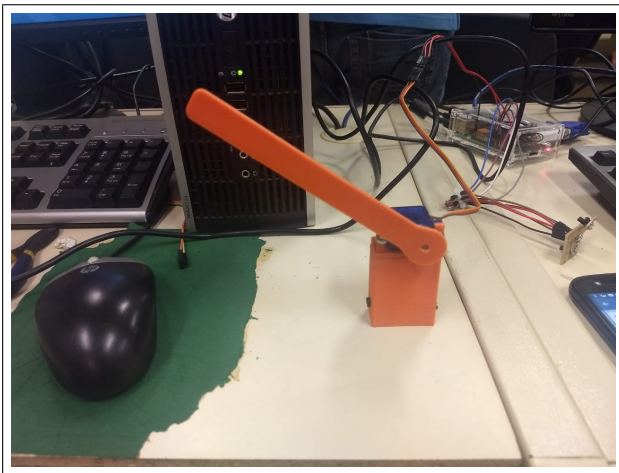
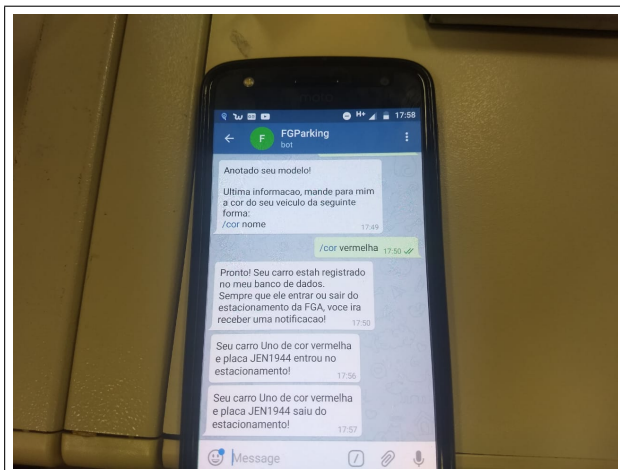


Fig. 21: Notificação pelo Telegram



8 CONCLUSÃO

Observou-se a falta de sistemas automatizados no controle e segurança de estacionamentos. Com isso, esse projeto foi proposto para auxiliar nas operações desses serviços ao agilizar a atividade de registro de entrada e saída de um estacionamento e de notificação do usuário do status de seu veículo.

Utilizou-se o Raspberry Pi 3 model B com um processador de 1.2GHz, CPU de 64-bit quad-core ARMv8, memória RAM de 1GB, cartão de memória de 32GB e Sistema Operacional Raspbian como Sistema Embarcado para a implementação da solução em linguagem C. Porém para a implementação de algumas funcionalidades da aplicação, foi necessária a utilização de outras linguagens de programação para auxiliar como python e PHP.

Para a notificação do status de acesso de seu veículo, primeiro optou-se pela utilização de email. Porém, visto a grande utilização de serviços de mensagens instantâneas baseados na nuvem, optou-se por utilizar o aplicativo Telegram. Para realizar o cadastro do veículo do usuário optou-se por implementar um bot em linguagem python utilizando a biblioteca [python-telegram-bot](#). Os dados do usuário são salvos em um arquivo .txt que é manipulado por um programa em C para extrair as informações necessárias para registro no Banco de Dados.

Para enviar ao usuário uma notificação que seu veículo entrou ou saiu do estacionamento, optou-se por utilizar a linguagem PHP, por ser de melhor manuseio dentro do escopo delimitado da disciplina. Para isso, utiliza-se uma função em C que cria o arquivo de extensão .php e o executa com uma chamada de sistema. Logo após o envio da notificação, o arquivo .php é apagado.

Além dos itens citados anteriormente, foi adicionado ao projeto uma cancela, a qual não estava prevista no início do projeto. Dessa forma, quando a imagem é capturada pelo botão de entrada ou saída, a cancela é aberta por 15 segundos para permitir a movimentação do veículo.

REFERENCES

- [1] CODEPLAN. *Pesquisa Distrital por Amostra de Domicílios*. Disponível em: www.codeplan.df.gov.br/pdad/. Acesso em: 02 de abril de 2018.
- [2] ELIAS, A. C. C. *Estacionamento rotativo pago em via pública – Racionalização do uso da via x disposição em pagar pelo serviço*. Mestrado submetido a Universidade Federal do Rio Grande do Sul. Porto Alegre, 2001. Disponível em: www.lume.ufrgs.br/handle/10183/2841. Acesso em: 02 de abril de 2018.
- [3] CARDIM, N. *Estudantes do Campus do Gama da UnB fazem paralisação para pedir segurança*. Disponível em: www.correiobraziliense.com.br/app/noticia/cidades/2016/04/20/interna_cidadesdf,528434/. Acesso em: 02 de abril de 2018.