



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO PIAUÍ
CAMPUS TERESINA CENTRAL
TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

RUBENS VICTOR SOUSA DE CARVALHO

**VAGAFÁCIL - SISTEMA DE SELEÇÃO DE VAGAS EM ESTACIONAMENTOS
ROTATIVOS**

TERESINA, PIAUÍ

2020

RUBENS VICTOR SOUSA DE CARVALHO

VAGAFÁCIL - SISTEMA DE SELEÇÃO DE VAGAS EM ESTACIONAMENTOS
ROTATIVOS

Projeto apresentado à Banca Examinadora como requisito para aprovação na disciplina de Trabalho de Conclusão de Curso I do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Piauí.

Orientador: Prof. M^e. Fernando Castelo Branco Gonçalves Santana

Coorientador: Prof. D^r. Thiago Alves Elias da Silva

TERESINA, PIAUÍ

2020

Dados Internacionais de Catalogação na Publicação (CIP) de acordo com ISBD

Carvalho, Rubens Victor Sousa de
C331v Vagafácil - sistema de seleção de vagas em estacionamentos rotativos /
Rubens Victor Sousa de Carvalho. - 2022.
42 f.: il. color.

Trabalho de Conclusão de Curso (Tecnologia em Análise e
Desenvolvimento de Sistemas) - Instituto Federal de Educação, Ciência e
Tecnologia do Piauí, Campus Teresina Central, 2022.

Orientador : Prof Me. Fernando Castelo Branco Gonçalves Santana .

1. Mobilidade. 2. Sistema. 3. Estacionamento rotativo. I. Título.

CDD - 004.21

Elaborado por Maria Rosismar Farias CRB 3/631

RUBENS VICTOR SOUSA DE CARVALHO

VAGAFÁCIL - SISTEMA DE SELEÇÃO DE VAGAS EM ESTACIONAMENTOS
ROTATIVOS

Projeto apresentado à Banca Examinadora como requisito para aprovação na disciplina de Trabalho de Conclusão de Curso I do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Piauí.

Aprovado pela banca examinadora em _____.

BANCA EXAMINADORA:

**Prof. M^º. Fernando Castelo
Branco Gonçalves Santana**
Instituto Federal de Educação,
Ciência e Tecnologia do Piauí - IFPI

**Prof. D^º. Thiago Alves Elias da
Silva**
Instituto Federal de Educação,
Ciência e Tecnologia do Piauí - IFPI

**Prof. M^º. Duanny Dreyton
Bezerra Sousa**
Instituto Federal de Educação,
Ciência e Tecnologia do Piauí - IFPI

**Prof. M^º. Rogerio da Silva
Batista**
Instituto Federal de Educação,
Ciência e Tecnologia do Piauí - IFPI

TERESINA, PIAUÍ

2020

AGRADECIMENTOS

Agradeço aos meu pais que, durante o andamento do curso, me proporcionaram apoio e segurança na minha escolha a este curso curso.

Agradeço ao IFPI e seu corpo docente pela possibilidade de me proporcionar acesso a um bom ambiente educacional e científico.

“Nós só podemos ver um pouco do futuro, mas o suficiente para perceber que há muito a fazer”
(Alan Turing)

RESUMO

Nas cidades por muitas vezes há uma concentração de indústrias, repartições públicas entre outros locais de grande movimentação e consequentemente grande fluxo de pessoas e veículos. Esse fluxo tem como consequência imediata a formação de engarrafamentos a demora para chegar nos locais que a pessoa deseja e dificuldade de estacionar o veículo com segurança. Devido ao aumento do número de veículos em circulação, surgiu a ideia de estacionamentos rotativos que são locais onde o condutor deixa o veículo durante algum tempo e paga de acordo com o tempo em que ele deixou o veículo no local. Estacionamentos rotativos ajudam no funcionamento das cidades pois são locais seguros e também desafogam o trânsito pois os carros não estacionam nas ruas. Porém, com o crescimento rápido da frota de veículos, a quantidade de vagas de estacionamentos não acompanhou essa demanda. Diante do exposto, este trabalho tem o objetivo em automatizar o processo de seleção de vagas em estacionamentos rotativos através da construção de uma aplicação web em comunicação com uma aplicação mobile para o manejo dessas vagas.

Palavras-chaves: mobilidade, sistema, estacionamentos rotativos

ABSTRACT

In cities there is often a concentration of industries, public other places of great movement, this has as consequence a great flow of people and vehicles. This flow has the immediate consequence of the formation of traffic jams at delay in arriving at the places the person wants and difficulty parking the vehicle with safety. To ease the large flow of vehicles around high-flow regions where the driver leaves the vehicle for some time and paid according to the time was the vehicle on the spot. Car parkings help the cities to function because it is a safe place and also vehicles will not be parked in the streets, but with growth fleet of vehicles the number of parking lots did not follow this demand problems of mobility and safety, so this work has the objective of to propose a tool to reserve parking spaces in rotating parking lots with the objective of efficient allocation of them. After the tool are build was see that performed good in reservation of vacancies, but can are need the plus of funcionalities to gain new users for the plataform.

Key-words: mobility, system, car parks

LISTA DE ILUSTRAÇÕES

Figura 1 – Arquitetura do sistema	19
Figura 2 – Diagrama do Proprietário do Estacionamento	22
Figura 3 – Diagrama do cliente	23
Figura 4 – Login Proprietário	25
Figura 5 – Registro Proprietário	25
Figura 6 – Home Proprietário	26
Figura 7 – Cadastrar Vaga	26
Figura 8 – Transações	27
Figura 9 – Login Cliente	28
Figura 10 – Registro Cliente	29
Figura 11 – Registro Cliente	30
Figura 12 – Tela de vagas	31
Figura 13 – Tela de confirmação da reserva	32
Figura 14 – Tela de perfil do usuário	33
Figura 15 – Tela de saída da vaga	34
Figura 16 – Tela de confirmação de pagamento	35
Figura 17 – Arquitetura do arduino	36
Figura 18 – Componentes	36
Figura 19 – Teste 1	37
Figura 20 – Teste 2	38
Figura 21 – Teste 3	38

LISTA DE TABELAS

Tabela 1 – Tabela de requisitos do cliente do estacionamento	20
Tabela 2 – Tabela de requisitos do dono do estacionamento	21

LISTA DE ABREVIATURAS E SIGLAS

LBS	Location-Based Service
SIG	Sistemas de Informação Geográfica
DRF	Django Rest Framework
JSON	JavaScript Object Notation
cm	Centímetros
LAN	Local Area Network
IBGE	Instituto Brasileiro de Geografia e Estatística
UTFPR	Universidade Tecnológica Federal do Paraná

SUMÁRIO

	Lista de tabelas	8
1	INTRODUÇÃO	12
1.1	Justificativa	13
1.2	Objetivos	13
1.2.1	Objetivo Geral	13
1.2.2	Objetivos Específicos	13
2	FUNDAMENTAÇÃO TEÓRICA	14
2.1	Arquitetura Cliente/Servidor	14
2.2	Estacionamentos Rotativos	14
2.3	Mobilidade Urbana relacionados a estacionamentos	15
2.3.1	Pesquisas sobre estacionamentos rotativos	15
3	TRABALHOS RELACIONADOS	16
3.1	Desenvolvimento e Integração de uma Aplicação Móvel a um Sistema Distribuído: Estudo de Caso de um Sistema de Gerenciamento de Vagas	16
3.2	Implementação de Recursos em um Smart Parking ~ baseado em Sistemas Multiagente	16
3.3	Aplicativo de Reserva de Vagas	17
4	VAGAFÁCIL - SISTEMA DE SELEÇÃO DE VAGAS EM ESTACIONAMENTOS ROTATIVOS	18
4.1	Modelagem do Sistema	18
4.1.1	Arquitetura do Sistema	18
4.1.2	Levantamento de Requisitos	19
4.1.3	Casos de Uso	21
4.2	Ferramentas usadas para a construção do módulo do proprietário do estacionamento	23
4.2.0.1	Python	23
4.2.0.2	Django	23
4.2.0.3	Django Rest Framework	24
4.3	Ferramentas usadas para a construção do módulo do Cliente	24
4.3.0.1	Typescript	24
4.3.0.2	Ionic	24
4.4	Proprietário do Estacionamento	24
4.4.1	Telas	24

4.5	Cliente	27
4.5.1	Telas	27
4.6	Testes de localização	35
4.6.1	Arduino	35
4.6.2	Sensores	36
4.6.3	Testes	37
5	CONCLUSÃO	39
5.1	Trabalhos futuros	39
	REFERÊNCIAS	40

1 INTRODUÇÃO

Este capítulo tem o objetivo de expor os problemas sobre os estacionamentos rotativos e a importância dos estacionamentos para as cidades.

Atualmente, as cidades possuem uma frota crescente de veículos automotores sejam carros, ônibus, caminhões ou qualquer outro. Segundo [AUTOESPORTE, 2017] em 2017 a frota de veículos era de 43.371.420 (carros, ônibus e caminhões).

Esta frota em constante crescimento, principalmente no começo do século 21, traz consigo problemas de mobilidade urbana como congestionamentos e desgaste das vias [LAMAS, 2014].

Aliada ao aumento da frota está o fato que não foi possível adaptar as cidades a uma frota de veículos em constante crescimento e como consequência a sua utilização para os mais diversos fins seja lazer ou trabalho [LAMAS, 2014]. Um exemplo da dificuldade das cidades brasileiras em se adaptar ao aumento da frota de veículo é a cidade de São Paulo que registra grandes engarrafamentos constantemente [VEJA, 2013].

Um bom exemplo de como a relação entre o crescimento da frota de veículos e o mal preparo das cidades em receber esses novos veículos é a grande quantidade de veículos circulando em pouco espaço. Tendo em vista esse problema os estacionamentos rotativos foram construídos para ser lugares onde as pessoas estacionam seus carros em um local seguro durante algum tempo mediante ao pagamento de uma taxa correspondente a duração em que o carro ficou estacionado.

Estacionamentos rotativos tem importância para as cidades a partir do momento em que é possível deixar o veículo em um local seguro assim evitando problemas recorrentes como furtos. Outra vantagem do uso dos estacionamentos rotativos é a flexibilidade do seu uso além de desafogar o trânsito pois o número de carros estacionados na rua diminui possibilitando o fluxo normal dos outros veículos [LAMAS, 2014].

Nas primeiras décadas do século 21, os dispositivos e serviços de informática se tornaram mais acessíveis dentre eles estão smartphones, notebooks, acesso a servidores, redes de internet entre outros. Dentre os dispositivos o smartphone merece destaque pois tem boa capacidade de processamento, leve e portátil podendo ser utilizado nos mais diversos fins. Segundo o IBGE em 2018 o número de smartphones no Brasil atingiu o número de 220 milhões de aparelhos em funcionamento no país, mais que a quantidade de pessoas que é 207,6 milhões.

Este trabalho tem como objetivo a construção de um sistema para a gerência de vagas em estacionamento rotativos.

1.1 JUSTIFICATIVA

O uso de ferramentas computacionais para resolução de obstáculos têm sido de grande importância no acesso da população a bens e serviços. Para conseguir esse acesso as pessoas fazem uso de elementos computacionais, como smartphones e computadores. Foi demonstrado em [ESTADÃO, 2018] que em 2018 65% dos consumidores compraram pelo menos uma vez por mês pela internet.

O uso de aplicativos para encontrar estacionamentos tornou-se de grande importância para os motorista como explica [FRANÇOSO; MELLO, 2016] que em sua pesquisa observou que 85,3% dos entrevistados utilizavam algum aplicativo de trânsito para otimizar o uso do veículo durante seu dia-a-dia.

Devido ao crescente número de usuários de meios eletrônicos na prestação de serviços surgiram ideias para facilitar o acesso aos mesmos. Um exemplo prático em que o uso de aplicativos para reservar vagas em estacionamentos rotativos foi a proposta feita por [PEREIRA, 2018] onde os autores fizeram um estudo sobre a viabilidade da construção de um estacionamento rotativo na cidade de Criciúma.

1.2 OBJETIVOS

1.2.1 Objetivo Geral

Este trabalho tem como objetivo o automação na seleção de vagas para motoristas em um estacionamento rotativo através da construção de uma aplicação web em comunicação com uma aplicação mobile para o manejo dessas vagas.

1.2.2 Objetivos Específicos

- Diminuição do tempo de procura da vaga de estacionamento.
- Previsibilidade do fluxo de veículos no estacionamento.
- Planejamento financeiro por parte do cliente do estacionamento.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão apresentados os fundamentos que os sistema se utilizará para o seu funcionamento. Será vista a arquitetura cliente/servidor e sua importância para o desenvolvimento de software, será visto um exemplo de estacionamento rotativo a fim de exemplificar o seu funcionamento e por último será mostrada um exemplo de proposta de aplicativo para reserva de vagas em estacionamentos rotativos.

2.1 ARQUITETURA CLIENTE/SERVIDOR

A arquitetura cliente/servidor representou grande avanço para o desenvolvimento de software pois possibilitou o aumento de produtividade, aumento de segurança, previsibilidade e divisão de responsabilidades das diferentes partes de um software [RABELO, 1998].

Apesar do uso em larga escala, atualmente a arquitetura cliente/servidor precisou da evolução da computação e do desenvolvimento de software para ser implantada. O principal modelo de arquitetura a surgir foi o modelo centralizado no qual grandes computadores, chamados mainframes, eram utilizados para processar os dados e estes computadores eram acessados por terminais que não possuíam capacidade de processamento. Com o avanço da tecnologia, como o surgimento da LAN (*Local Area Network*) e dos microcomputadores, o trabalho com os computadores ficou mais rápido, já que os microcomputadores possuíam maior capacidade de processamento e segurança. Devido o desenvolvimento tecnológico de equipamentos que acessam outros a fim de obter informações, chamado cliente, e o surgimento de computadores que cuidam do processamento e segurança, chamados servidores foi possível com que a arquitetura cliente/servidor se tornar a principal forma de trabalhar em projetos de Tecnologia da Informação [RABELO, 1998].

2.2 ESTACIONAMENTOS ROTATIVOS

Estacionamentos rotativos se fazem necessários nas cidades a partir do momento em que o crescimento da frota de veículos automotores afeta o funcionamento das cidades. De acordo com [AUTOESPORTE, 2017] o volume de veículos circulando no país chegou a 43 milhões. Esse número crescente da frota de veículos, puxado principalmente por carros e motos, tem como algumas consequências o aumento de tempo de deslocamento, aumento da probabilidade de acidentes e lentidão no transporte de mercadorias e serviços.

Tendo em vista os problemas causados pelo aumento da frota de veículos surge os estacionamentos rotativos cujo o objetivo é guardar o veículo do condutor, geralmente carros e motos, durante um certo período de tempo, mediante pagamento, proporcionando conforto

para o condutor e melhorando o trânsito além de ser fonte de políticas públicas para as cidades [LAMAS, 2014].

2.3 MOBILIDADE URBANA RELACIONADOS A ESTACIONAMENTOS

Mobilidade urbana é um tema recorrente em discussões em vários setores da sociedade. Estacionamentos rotativos são de grande importância para o tema pois seu público alvo é basicamente carros que representam a maior parte da frota de veículos do país com 36.007.536 segundo [AUTOESPORTE, 2017]. Nesta seção serão apresentados pesquisas conceitos relacionados a relação entre estacionamentos rotativos e mobilidade urbana.

2.3.1 Pesquisas sobre estacionamentos rotativos

No mundo moderno o tema mobilidade urbana vem sendo recorrente em debates dos mais diferenciados nichos e por isso a ciência busca meios de entender e propor soluções para o tema da mobilidade.

Para entender esses problemas [RIBAS; GUERREIRO; GUERREIRO, 2017] fizeram um trabalho de pesquisa da qualidade do serviço de estacionamento rotativo público na cidade de Caxias do Sul e descobriram, por exemplo, que 12% dos entrevistados moram em cidades próximas, indicando que Caxias do Sul é uma cidade atrativa para moradores de municípios vizinhos. Seguindo no mesmo problema de pesquisa sobre os usuários do sistema de estacionamentos rotativos os pesquisadores de [CRUZ; CRUZ; CERETTA, 2016] fizeram uma pesquisa sobre os estacionamentos rotativos públicos em duas zonas na cidade de Bagé usando o modelo da satisfação dos clientes ECSI que a faixa etária mais frequente é a de 26-35 anos com 36,7% dos entrevistados, 89,1% usam o carro como principal forma de locomoção pela cidade.

3 TRABALHOS RELACIONADOS

Neste capítulo serão apresentados os trabalhos que de alguma forma se relacionam com o objetivo proposto neste trabalho.

3.1 DESENVOLVIMENTO E INTEGRAÇÃO DE UMA APLICAÇÃO MÓVEL A UM SISTEMA DISTRIBUÍDO: ESTUDO DE CASO DE UM SISTEMA DE GERENCIAMENTO DE VAGAS

Os pesquisadores [CUNHA; BRITO; COUTINHO, 2016] propuseram uma aplicação que une uma aplicação de gerenciamento de vagas que usa um componente de aplicativos android em conjunto a um componente de sistema embarcado na plataforma arduino além de usar servidores para a comunicação destes e banco de dados para a persistência dos dados.

O sistema faz o uso do arduino para monitorar o estado da vaga por meio de componentes de hardware e a partir disso integrar com as outras partes do sistema. Para fazer isso os autores usaram conceitos de sistemas distribuídos, estabeleceram protocolos de comunicação entre as diferentes partes sistema. Como esse aplicativo não é focado na seleção de motoristas para vagas em estacionamentos rotativos a aplicação não seleciona o motorista no momento da requisição, além de não possuir um sistema web para a aplicação [CUNHA; BRITO; COUTINHO, 2016].

3.2 IMPLEMENTAÇÃO DE RECURSOS EM UM SMART PARKING ~ BASEADO EM SISTEMAS MULTIAGENTE

Encontrar vagas de estacionamentos livres, principalmente em grandes cidades, tem se tornado um desafio para o urbanismo. Com isso em discussão os pesquisadores [DUCHEIKO; CASTRO; ALVES1, 2016] propuseram a construção de uma interface gráfica com informações sobre o uso do estacionamento e a persistência de dados dentro do projeto MAPS da UTFPR.

O sistema mostra em alguns gráficos como a quantidade de carros estacionados, fila de espera e histórico do uso do estacionamento. Também permite a persistência dos dados em um banco de dados para posterior uso [DUCHEIKO; CASTRO; ALVES1, 2016].

Esse aplicativo, que é parte do sistema MAPS da UTFPR, não faz uso de uma aplicação móvel para requisição da vaga além de não permitir interação de quem vai gerenciar as vagas com o sistema, como por exemplo excluindo ou adicionando vagas [DUCHEIKO; CASTRO; ALVES1, 2016].

3.3 APLICATIVO DE RESERVA DE VAGAS

Com o aumento das cidades, do poder aquisitivo das famílias brasileiras e a facilitação das linhas de crédito foi possível que boa parte das famílias brasileiras adquirissem um veículo seja para trabalhar, locomover-se ou por necessidade física, como no caso de cadeirantes. Como consequência desse aumento da frota de veículos as cidades, principalmente as capitais, tiveram um aumento na piora do trânsito assim demandando novas soluções para o controle deste [SANTOS et al., 2018].

Várias soluções estão sendo postas frequentemente dentre elas destaca-se o uso de aplicativos atrelado ao uso de dispositivos móveis. Por isso o autores do estudo vendo o rápido crescimento de dispositivos móveis observaram que estes eram muito usados em eventos esportivos, pontos comerciais e outros locais de grande fluxo de pessoas e poderiam ser usados para a obtenção de vagas em estacionamentos rotativos. No projeto do aplicativo foram definidas algumas regras como a negociação do preço da vaga entre o dono do estacionamento e o possível cliente, a multa em caso a pessoa cancelar a requisição da vaga em menos de duas horas. Já na parte de experiência de usuário foi projetadas telas de smartphones, opção de pagamento com cartão de crédito, autenticação por rede social e a visualização do dono da vaga por parte do cliente [SANTOS et al., 2018].

Esse aplicativo é principalmente focado no uso por donos de carros, o que tem como consequência a não construção de um módulo para os donos do estacionamento, além da não interação das vagas com o sistema via algum dispositivo de hardware como o Arduino [SANTOS et al., 2018].

4 VAGAFÁCIL - SISTEMA DE SELEÇÃO DE VAGAS EM ESTACIONAMENTOS ROTATIVOS

Neste capítulo apresentamos a plataforma de reserva de vagas e seu funcionamento. O sistema é formado por dois módulos sendo uma plataforma web para o dono do estacionamento contendo as principais informações para o acompanhamento das vagas e um módulo mobile para os clientes terem a oportunidade de selecionar a vaga de sua preferência.

No decorrer do capítulo é mostrado o planejamento e execução da construção e testes do sistema proposto passando pela modelagem, arquitetura do sistema, ferramentas usadas e testes da aplicação.

4.1 MODELAGEM DO SISTEMA

Nesta seção é apresentada como a aplicação foi planejada a fim de ter documentos que guiem o processo de construção do software.

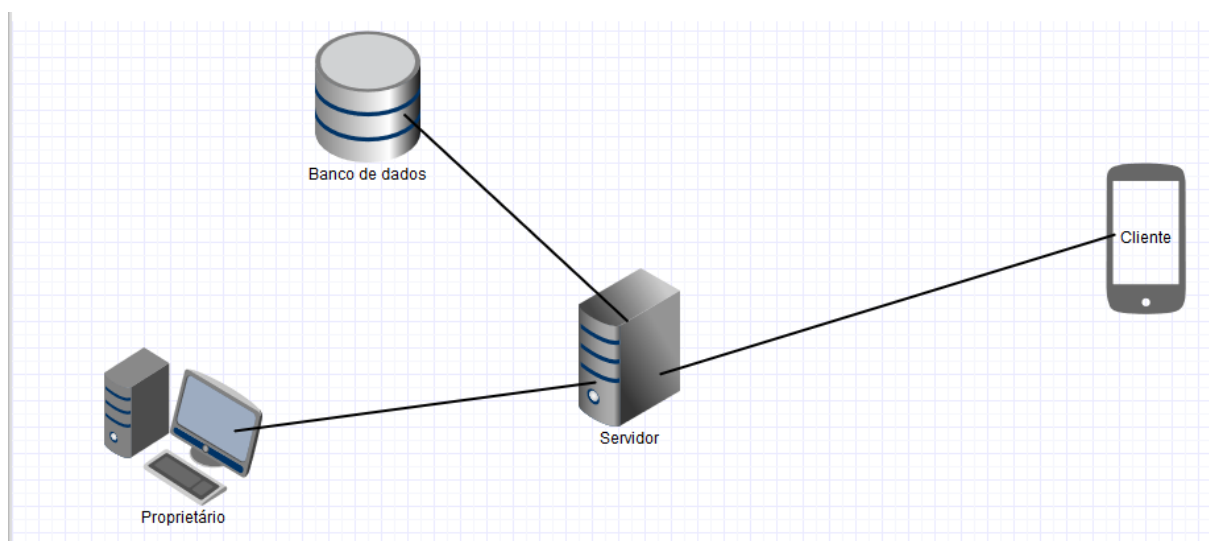
4.1.1 Arquitetura do Sistema

Para obter uma visualização geral do fluxo de informações foi necessário o desenho básico de como e quais dispositivos informacionais o sistema usa. Para isso foi feito o desenho de arquitetura do sistema usando a ferramenta gliffy [GLIFFY, 2019].

A arquitetura consiste no seguinte esquema um cliente manifesta interesse em uma vaga de estacionamento e a partir disso o sistema retorna a resposta sobre a liberação ou não da vaga para a pessoa, caso positivo ela guarda em um banco de dados as informações sobre aquela reserva e mostra para o proprietário o estado das vagas do seu estacionamento no momento.

Abaixo é mostrada um esquema da arquitetura usada para a aplicação:

Figura 1 – Arquitetura do sistema



Fonte: Próprio autor

Nessa imagem é visto o servidor, onde é o elemento que faz a distribuição dos dados e requisições da aplicação. Também é possível ver os dispositivos que o dono estacionamento e o cliente vão utilizar, no caso do dono do estacionamento será usado um computador para a melhor gerencia das vagas, já no caso do cliente será usado um dispositivo móvel para melhor comodidade. Por fim o banco de dados fará a persistência dos dados da aplicação.

4.1.2 Levantamento de Requisitos

O levantamento de requisitos consiste em analisar e identificar as características e funcionalidades que o software deve possuir. O levantamento de requisitos é dividido em duas formas requisitos funcionais e requisitos não funcionais. Os requisitos funcionais definem o que a aplicação deve fazer para obter um determinado fim [FILHO; MARTINEZ, 2006]. Já os requisitos não funcionais expressão condições e características, como performance, que o software tem de possuir para o seu devido funcionamento [CYSNEIROS; LEITE, 1998].

Abaixo é mostrada uma tabela com os requisitos do módulo do Cliente com suas respectivas classificações em requisito Funcional ou requisito Não Funcional.

Requisito	Tipo do Requisito
Fazer Registro no Aplicativo	Funcional
Fazer login no aplicativo	Funcional
Ver vagas	Funcional
Requisitar vaga	Não funcional
Sair da vaga	Não funcional
Ver vaga alocada	Funcional
Ver o quanto gastou durante o tempo que o cliente permaneceu no estacionamento	Funcional
Confiabilidade no cálculo do valor da duração do carro no estacionamento	Não Funcional
Consumo da API fornecida pelo módulo do proprietário	Não Funcional

Tabela 1 – Tabela de requisitos do cliente do estacionamento

Em seguida é mostrada uma tabela com os requisitos do módulo do Proprietário do Estacionamento com suas respectivas classificações em requisito Funcional ou requisito Não Funcional.

Requisito	Tipo do Requisito
Fazer registro no aplicativo	Funcional
Fazer login no aplicativo	Funcional
Cadastrar vaga	Funcional
Ver vagas ocupadas e vagas livres	Funcional
Confiabilidade de que os carros estão realmente ocupando as vagas	Não Funcional
Tela com as transações em andamento e a últimas que foram concluídas	Funcional
Fornecer uma API para o módulo do Cliente	Não Funcional

Tabela 2 – Tabela de requisitos do dono do estacionamento

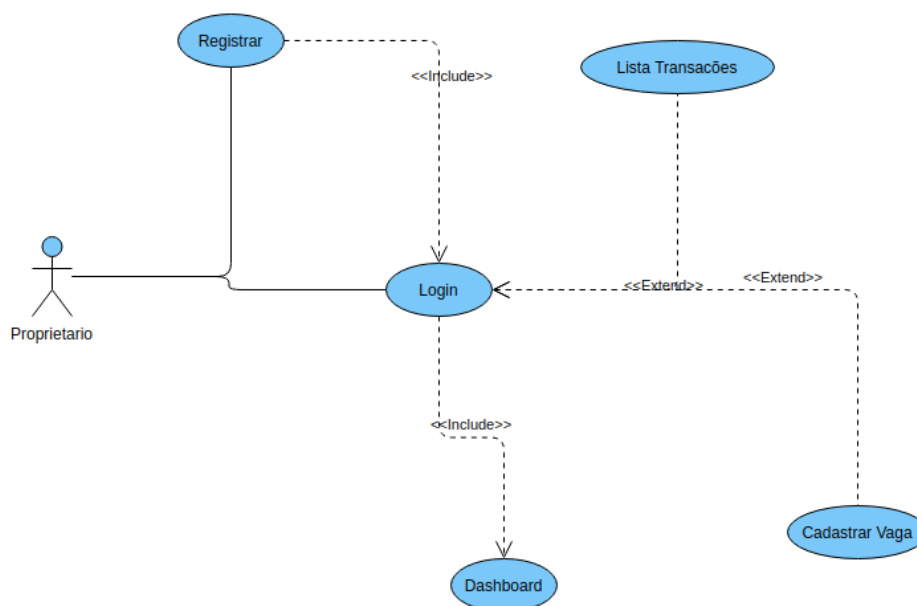
4.1.3 Casos de Uso

O diagrama de casos de uso do UML é importante pois mostra uma possível visualização de quais funcionalidades o sistema terá e como usuário fará a manipulação delas [KON, 2005].

Neste trabalho foram feitas dois diagramas de caso de uso, um para o cliente e outro para o proprietário do estacionamento.

No primeiro diagrama é mostrado um esquema de casos de usos do proprietário do estacionamento em que quando ele faz o login, ou o registro na aplicação, é mostrada a tela com a lista de vagas ocupadas e vagas livres. A partir da tela de vagas ocupadas e vagas livres o dono do estacionamento pode entrar na tela de cadastro de vaga para colocar mais uma vaga no sistema, também é possível ver as transações em andamento e as últimas transações feitas.

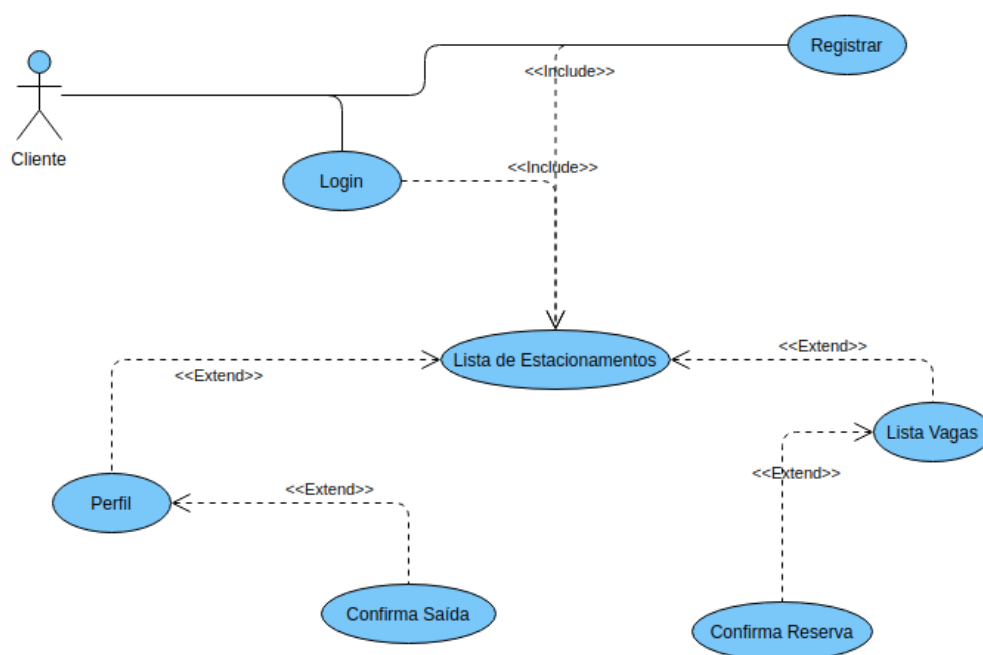
Figura 2 – Diagrama do Proprietário do Estacionamento



Fonte: Autor

Já no diagrama de casos de uso do cliente é mostrado onde após fazer o login, ou o registro na aplicação, é mostrada uma tela com todos os donos de estacionamentos disponíveis. Caso o cliente clique em um estacionamento é mostrado uma tela com todas as vagas disponíveis naquele estacionamento, se o cliente clicar em uma vaga livre será mostrada uma tela de confirmação de reserva de vaga. Caso o cliente reserve uma vaga ele poderá sair da vaga indo na tela de perfil e clicar em sair da vaga, que o levará a uma outra tela onde será requisitado a confirmação da saída da vaga.

Figura 3 – Diagrama do cliente



Fonte: Autor

4.2 FERRAMENTAS USADAS PARA A CONSTRUÇÃO DO MÓDULO DO PROPRIETÁRIO DO ESTACIONAMENTO

Nesta seção serão apresentadas as ferramentas que foram usadas para construir a parte que o proprietário irá usar.

4.2.0.1 Python

A linguagem python foi escolhida para fazer a parte do back-end pois apresenta facilidade na leitura de código, curva de aprendizado relativamente rápida e largamente usada por desenvolvedores [ROSSUM, 2019].

Python é uma linguagem robusta que apresenta grande quantidade de desenvolvedores além de ser adotada nos mais variados projetos [ROSSUM, 2019]

4.2.0.2 Django

O framework de desenvolvimento web Django foi escolhido para construir a parte onde o dono estacionamento interage com a aplicação além de fornecer a API onde o cliente web irá se comunicar com o cliente mobile [HOLOVATY; WILLISON, 2019].

A escolha do Django foi feita pois apresenta robustez, curva de aprendizado rápida e possui grande comunidade de adeptos e contribuidores [HOLOVATY; WILLISON, 2019].

4.2.0.3 Django Rest Framework

O Django Rest Framework foi utilizada para fazer a comunicação via REST entre a parte web e o mobile pois apresenta boa documentação, grande comunidade e estabilidade na sua manutenção [ZUPR, 2019].

O Django Rest Framework usa como forma de comunicação com as outras aplicações usando principalmente o formato JSON, formato largamente usado na comunicação entre aplicações [ZUPR, 2019].

4.3 FERRAMENTAS USADAS PARA A CONSTRUÇÃO DO MÓDULO DO CLIENTE

Nesta seção será apresentada as ferramentas que foram usadas para construir a parte que o cliente irá usar.

4.3.0.1 Typescript

O Typescript foi utilizado na construção da parte do cliente pois apresenta curva de aprendizagem rápida, estabilidade no mercado e boa documentação [MICROSOFT, 2016].

A linguagem Typescript tornou-se uma linguagem bastante expansível, intuitiva e flexível. Por isso vem sendo utilizada em nos mais diversos projetos tendo destaque o uso nos framework's Angular e Ionic [MICROSOFT, 2016].

4.3.0.2 Ionic

O Ionic é um framework para fazer aplicativos para dispositivos mobile baseado no angular e que usa o typescript como linguagem, esse framework foi escolhido para a parte do cliente da aplicação pois permite criar aplicativos para android e ios com um mesmo código, tem estabilidade na manuntenção e boa dosumentação [LYNCH; SPERRY; BRADLEY, 2019].

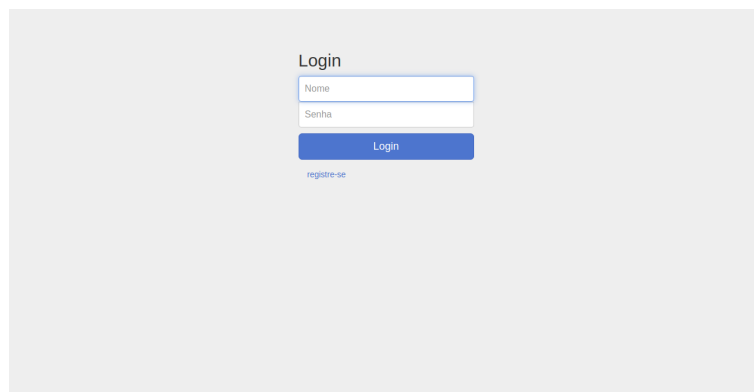
O framework Ionic é um framework híbrido, ou seja um mesmo código pode gerar um aplicativo para Android e para IOS, com muitos anos de uso em aplicações dos mais diversos tipos [LYNCH; SPERRY; BRADLEY, 2019].

4.4 PROPRIETÁRIO DO ESTACIONAMENTO

Nesta seção será apresentada as funcionalidades da parte do Proprietário do estacionamento na aplicação.

4.4.1 Telas

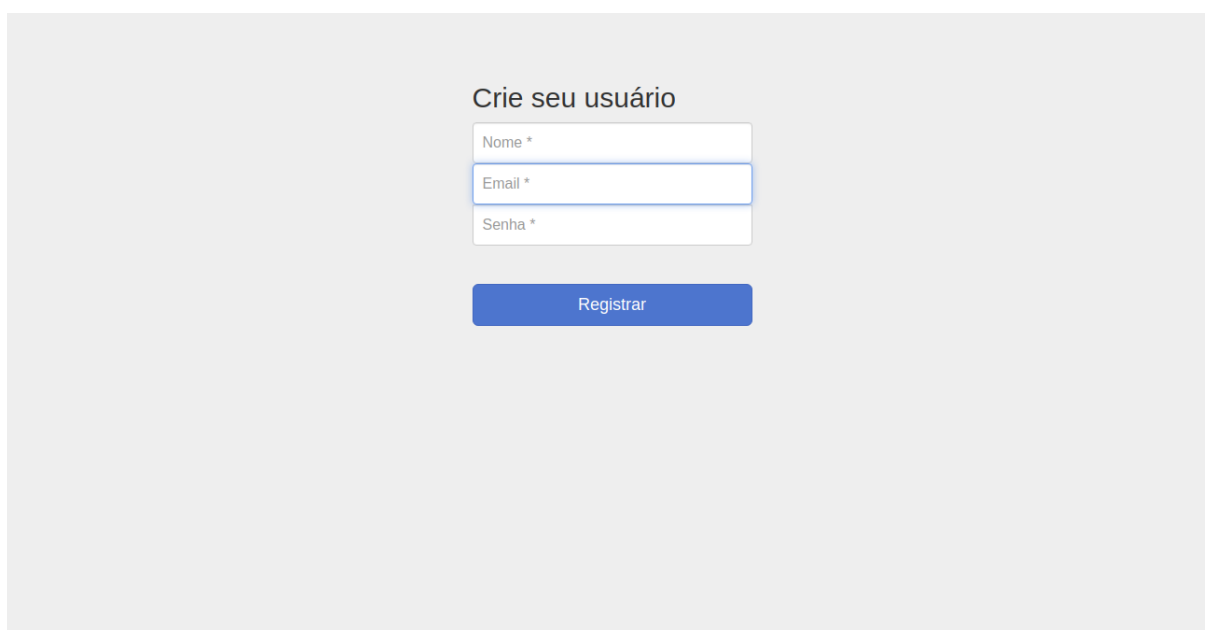
A primeira que aparece quando o usuário entra na aplicação é a tela de login. Nessa tela o proprietário terá de informar o seu nome de usuário e senha para poder entrar no sistema além de um link para a tela de registro caso o proprietário não seja cadastrado no sistema.



The image shows a login form titled "Login". It contains two input fields: "Nome" (Name) and "Senha" (Password). Below these fields is a blue button labeled "Login". At the bottom of the form, there is a small, faint link that says "registre-se" (register).

Figura 4 – Login Proprietário

Ao clicar no botão registre-se ele pode fazer seu cadastro. Para fazer seu cadastro ele deverá apresentar um nome de usuário e e-mail diferentes dos que já existem no sistema bem como uma senha e assim ganhando direito a um estacionamento no sistema.



The image shows a registration form titled "Crie seu usuário" (Create your user). It contains three input fields: "Nome *" (Name), "Email *" (Email), and "Senha *" (Password). Below these fields is a blue button labeled "Registrar" (Register).

Figura 5 – Registro Proprietário

Entrando na aplicação é apresentado ao usuário uma tela que possui as vagas ocupadas e vagas livres, links para cadastrar vaga e para fazer logout e também um ícone de lixeira que o usuário pode excluir uma vaga.

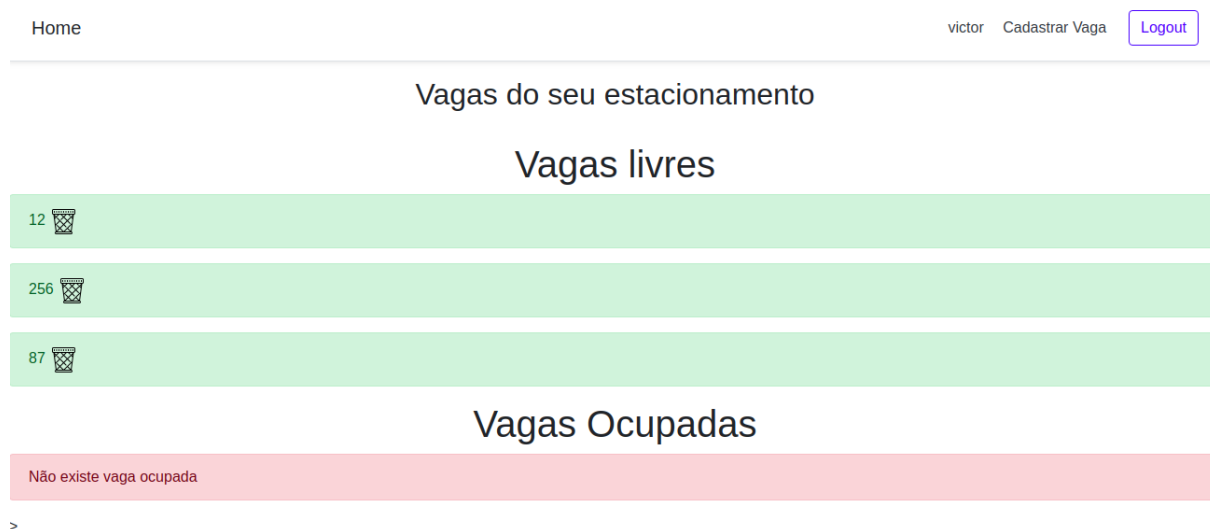


Figura 6 – Home Proprietário

Caso o usuário queira cadastrar uma vaga ele poderá entrar na seguinte tela e efetuar o cadastro. Nessa tela é mostrada o nome do proprietário, link para voltar para a home e um link caso ele queira sair do sistema. No formulário de cadastro o usuário informa um número de vaga e o tipo da vaga.

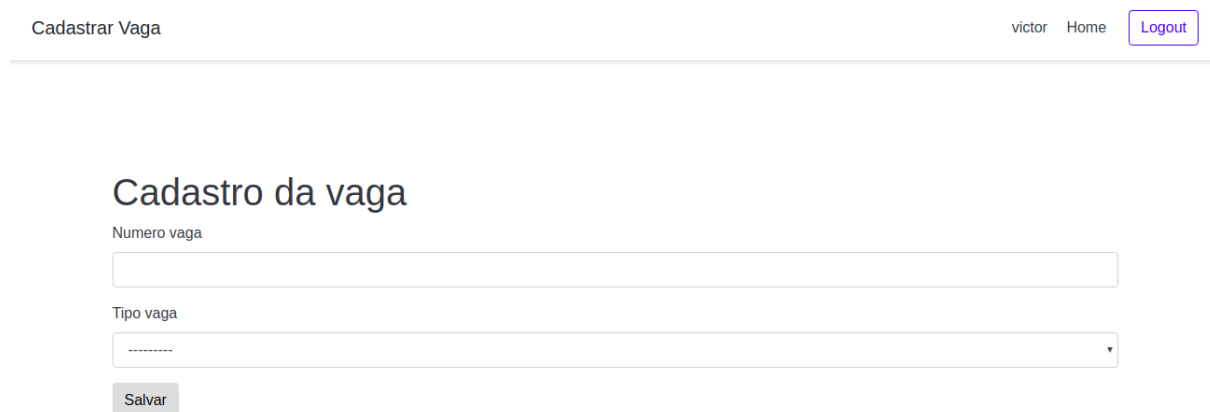


Figura 7 – Cadastrar Vaga

O Proprietário pode ainda ver as transações em andamento e as últimas transações feitas pelo sistema indo para a tela Transações do aplicativo.

Home	victor	Cadastrar Vaga	Home	Logout
Transações ainda em andamento				
Vaga: 12 Status: Transação em Andamento Hora de entrada: Oct. 18, 2020, 3:09 p.m. Cliente: igor				
Últimas transações terminadas				
Vaga: 12 Status: Transação Terminada Total da transacao: 40.0 centavos Cliente: igor Hora de entrada: Oct. 17, 2020, 8 a.m.				
Vaga: 12 Status: Transação Terminada Total da transacao: 10.0 centavos Cliente: igor Hora de entrada: Oct. 9, 2020, 10:15 a.m.				
Vaga: 12 Status: Transação Terminada				

Figura 8 – Transações

4.5 CLIENTE

Nesta seção será apresentada as funcionalidades da parte do Cliente na aplicação.

4.5.1 Telas

Quando o usuário entra na aplicação é mostrada o login. Quando a pessoa está registrada e vai entrar novamente no aplicativo ela precisa dar duas credenciais (nome de usuário e senha) para ter permissão de entrar no aplicativo.

Faça seu Login

Username

Password

LOGIN

Forgot Password?

REGISTRAR

Figura 9 – Login Cliente

Caso o usuário ainda não esteja cadastrado na aplicação, ele pode clicar no botão *Register* para ir a tela de cadastro. Nessa parte o usuário faz o registro na aplicação para assim ter permissão de ter seus dados persistidos no banco de dados do aplicativo e poder usar as funcionalidades do mesmo.

Registre-se

Username

Email

Password

Confirme o Password

REGISTRAR

Figura 10 – Registro Cliente

Entrando na aplicação o usuário pode ver a tela com todos os estacionamentos cadastrados na aplicação. Nessa tela todos os estacionamentos cadastrado são mostrados ao cliente para que ele possa clicar e um deles para ver as vagas livres.



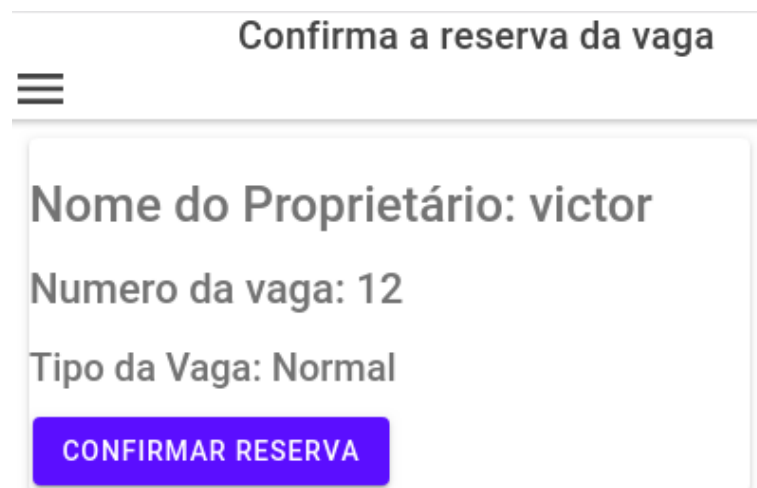
Figura 11 – Registro Cliente

Ao clicar em alguma imagem de estacionamento o usuário é levado a tela de vagas livres daquele proprietário. Nessa tela o cliente vê apenas as vagas livres além de ver o número e o tipo de vaga.



Figura 12 – Tela de vagas

Ao clicar em alguma das vagas livres o usuário é direcionado para uma tela de confirmação da reserva de vaga. Na tela é mostrada o nome do proprietário, o número da vaga e o tipo dela e um botão para confirmar a *Reserva da Vaga*.



A interface de confirmação da reserva apresenta um cabeçalho com o título "Confirma a reserva da vaga" e um ícone de menu hambúrguer à esquerda. Abaixo, um formulário branco com uma sombra cinza contém as seguintes informações: "Nome do Proprietário: victor", "Numero da vaga: 12" e "Tipo da Vaga: Normal". No final do formulário, há um botão retangular de cor amarela com o texto "CONFIRMAR RESERVA" em letras maiúsculas e cor preta.

Figura 13 – Tela de confirmação da reserva

Caso confirmada a reserva da vaga o usuário é direcionado a tela de perfil. Nesta interface é mostrada as principais informações do usuário. Essa tela é onde o cliente pode observar o número da vaga que ele está ocupando, um botão de *Sair da Vaga* bem como o horário em que ela foi alocada.

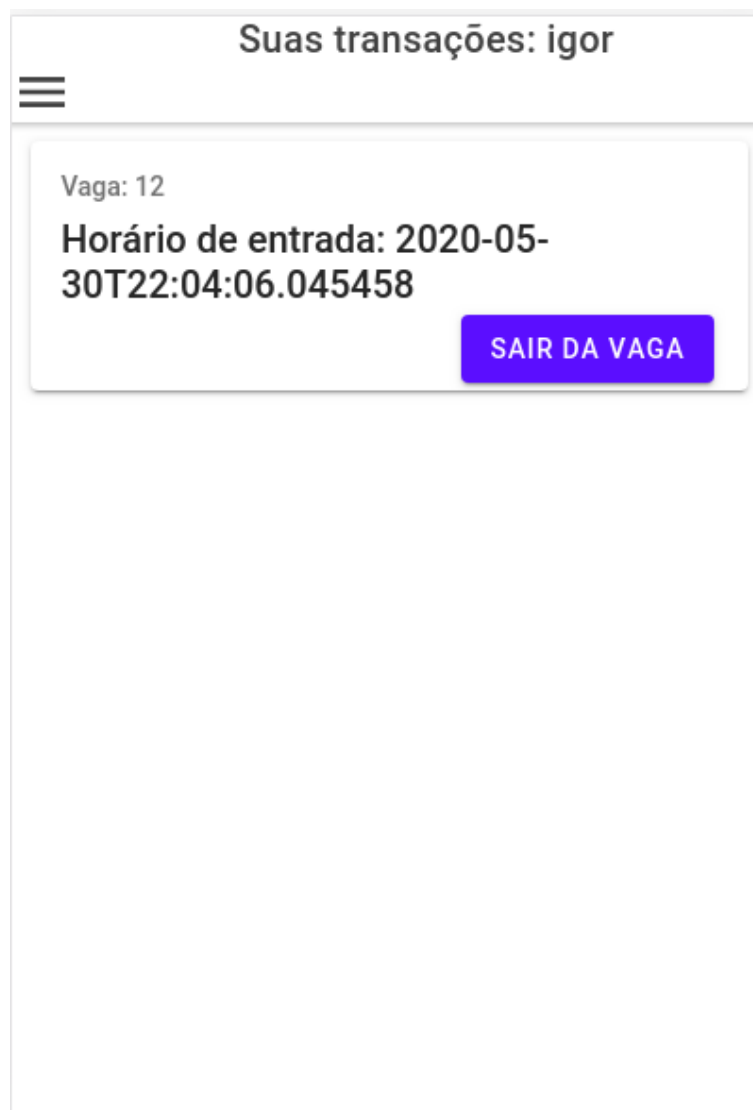


Figura 14 – Tela de perfil do usuário

Caso o cliente queira sair da vaga, a primeira ação que tem de ser feita é a de ir na tela de perfil e clicar no botão *Sair da Vaga*. Feita essa ação o usuário será conduzido a tela de confirmar saída da vaga. Nessa tela é mostrada o uma mensagem alertando se a pessoa quer sair da vaga, o botão *Sair da Vaga* e ainda informações como número, tipo e dono da vaga.

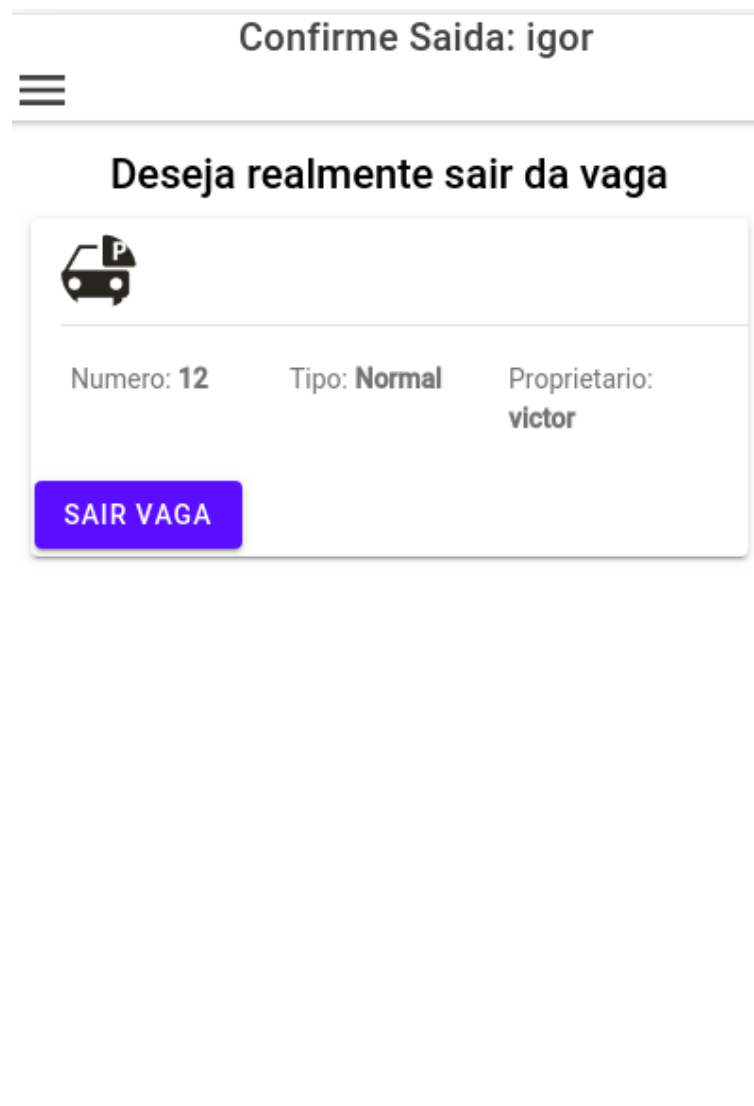


Figura 15 – Tela de saída da vaga

Se o usuário quer sair da vaga ele deve clicar no botão *Sair Vaga* e então ele será conduzido a tela de estacionamentos da aplicação e aparecerá um aviso com a informação de quanto ele gastou durante a permanência na vaga do estacionamento.

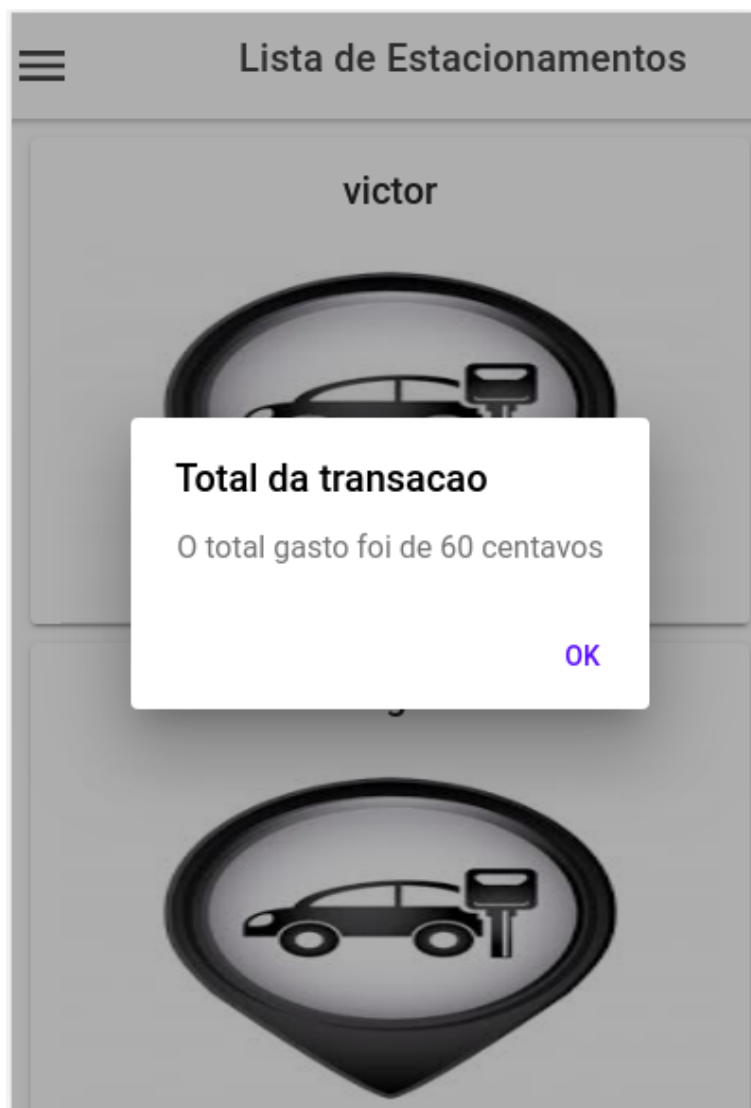


Figura 16 – Tela de confirmação de pagamento

4.6 TESTES DE LOCALIZAÇÃO

Quando o cliente faz a requisição para o sistema que quer sair é necessário saber se o veículo está verdadeiramente fora da vaga, por isso foi necessário a construção de um ambiente que simule um estacionamento real para testes do sistema utilizando a plataforma de prototipagem arduino.

Os testes consistiram em usar uma placa arduino[ARDUINO, 2005], como base de processamento, junto com um conjunto de sensores mandando constantemente sinais ao sistema para saber se o carro está fora da vaga quando o motorista requisitar sair da vaga.

4.6.1 Arduino

Para que o arduino funcionasse da maneira correta foi montada uma protoboard conectada com três pinos ao arduino juntamente com dois GND's e uma válvula de 5 volts. A arquitetura é

mostrada na imagem abaixo.

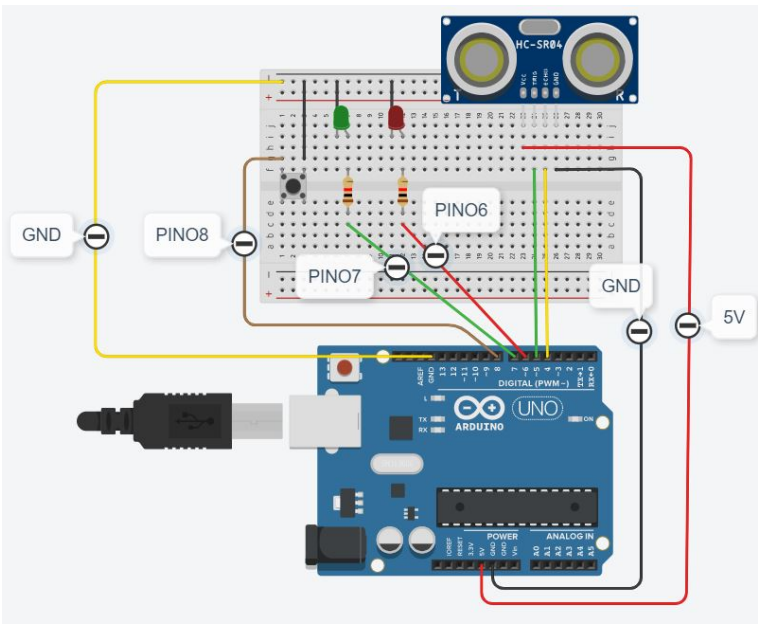


Figura 17 – Arquitetura do arduino

4.6.2 Sensores

Para fazer com que os testes funcionassem foi necessário utilizar componentes externos a placa.

Na imagem abaixo é possível ver os componentes utilizados na construção dos testes.

Nome	Quantidade	Componente
U3	1	Arduino Uno R3
S1	1	Botão
D1	1	Vermelho LED
D2	1	Verde LED
DIST1	1	Sensor de distância ultrassônico
R1 R2	2	1 kΩ Resistor

Figura 18 – Componentes

4.6.3 Testes

Montada a estrutura do arduino juntamente com os componentes externos acoplados a placa chegou o momento de fazer os testes.

O teste consiste em quando o cliente apertar o botão de saída da vaga o sistema confere se o carro ainda está na vaga, usando uma margem de 70 cm. Caso o carro esteja a menos de 70 cm o sistema não libera a vaga caso contrário o sistema a libera. Tendo em vista essas regras descritas foram feitos alguns testes.

No primeiro exemplo temos um carro que estacionou e pouco tempo depois ele saiu da vaga, após isso outro carro entrou na vaga e também saiu da vaga.

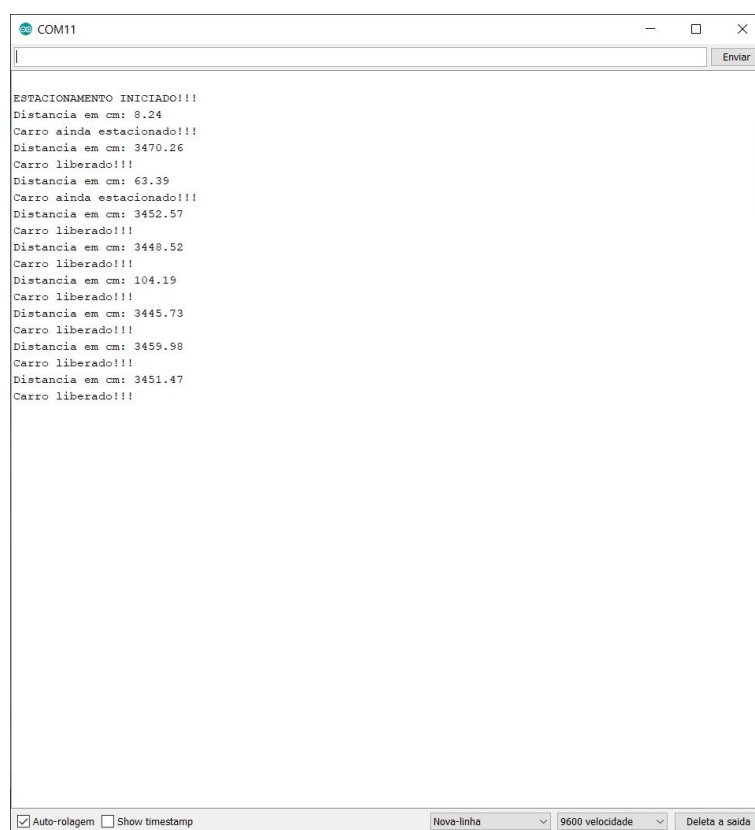
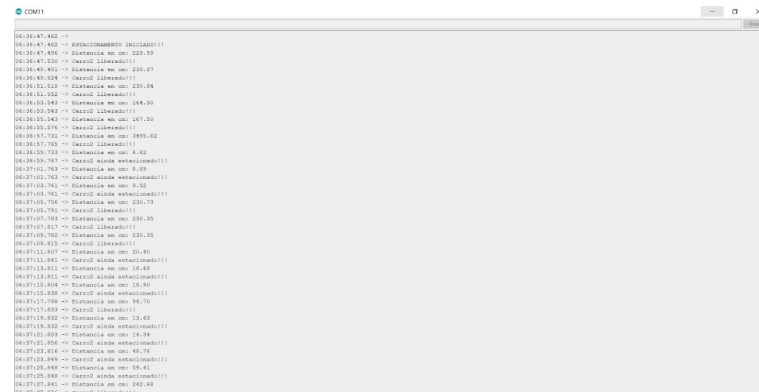


Figura 19 – Teste 1

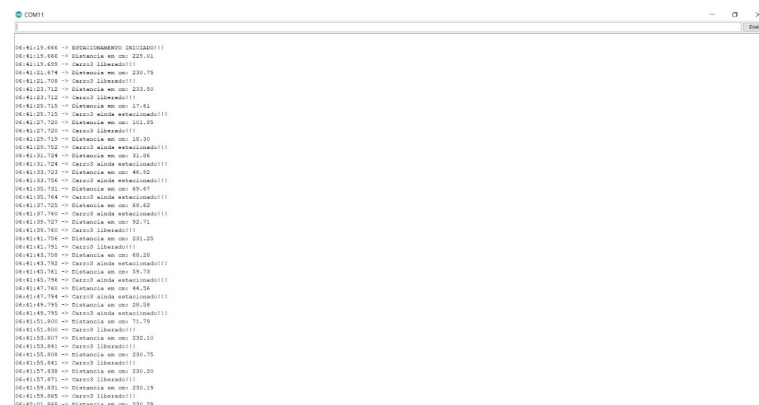
No segundo exemplo temos o primeiro carro entrou as 06:36 e saiu pouco tempo depois, logo após entrou mais dois carros na mesma vaga e a aplicação performou como o esperado.



```
COM1
[06:36:47.402] -> [VAGA000000000] [ENTRADA] [06:36:47.402]
[06:36:47.405] -> [VAGA000000000] [SAÍDA] [06:36:47.405]
[06:36:47.420] -> [VAGA000000000] [ENTRADA] [06:36:47.420]
[06:36:47.430] -> [VAGA000000000] [SAÍDA] [06:36:47.430]
[06:36:47.450] -> [VAGA000000000] [ENTRADA] [06:36:47.450]
[06:36:47.460] -> [VAGA000000000] [SAÍDA] [06:36:47.460]
[06:36:47.480] -> [VAGA000000000] [ENTRADA] [06:36:47.480]
[06:36:47.490] -> [VAGA000000000] [SAÍDA] [06:36:47.490]
[06:36:47.510] -> [VAGA000000000] [ENTRADA] [06:36:47.510]
[06:36:47.520] -> [VAGA000000000] [SAÍDA] [06:36:47.520]
[06:36:47.540] -> [VAGA000000000] [ENTRADA] [06:36:47.540]
[06:36:47.550] -> [VAGA000000000] [SAÍDA] [06:36:47.550]
[06:36:47.570] -> [VAGA000000000] [ENTRADA] [06:36:47.570]
[06:36:47.580] -> [VAGA000000000] [SAÍDA] [06:36:47.580]
[06:36:47.600] -> [VAGA000000000] [ENTRADA] [06:36:47.600]
[06:36:47.610] -> [VAGA000000000] [SAÍDA] [06:36:47.610]
[06:36:47.630] -> [VAGA000000000] [ENTRADA] [06:36:47.630]
[06:36:47.640] -> [VAGA000000000] [SAÍDA] [06:36:47.640]
[06:36:47.660] -> [VAGA000000000] [ENTRADA] [06:36:47.660]
[06:36:47.670] -> [VAGA000000000] [SAÍDA] [06:36:47.670]
[06:36:47.690] -> [VAGA000000000] [ENTRADA] [06:36:47.690]
[06:36:47.700] -> [VAGA000000000] [SAÍDA] [06:36:47.700]
[06:36:47.720] -> [VAGA000000000] [ENTRADA] [06:36:47.720]
[06:36:47.730] -> [VAGA000000000] [SAÍDA] [06:36:47.730]
[06:36:47.750] -> [VAGA000000000] [ENTRADA] [06:36:47.750]
[06:36:47.760] -> [VAGA000000000] [SAÍDA] [06:36:47.760]
[06:36:47.780] -> [VAGA000000000] [ENTRADA] [06:36:47.780]
[06:36:47.790] -> [VAGA000000000] [SAÍDA] [06:36:47.790]
[06:36:47.810] -> [VAGA000000000] [ENTRADA] [06:36:47.810]
[06:36:47.820] -> [VAGA000000000] [SAÍDA] [06:36:47.820]
[06:36:47.840] -> [VAGA000000000] [ENTRADA] [06:36:47.840]
[06:36:47.850] -> [VAGA000000000] [SAÍDA] [06:36:47.850]
[06:36:47.870] -> [VAGA000000000] [ENTRADA] [06:36:47.870]
[06:36:47.880] -> [VAGA000000000] [SAÍDA] [06:36:47.880]
[06:36:47.900] -> [VAGA000000000] [ENTRADA] [06:36:47.900]
[06:36:47.910] -> [VAGA000000000] [SAÍDA] [06:36:47.910]
[06:36:47.930] -> [VAGA000000000] [ENTRADA] [06:36:47.930]
[06:36:47.940] -> [VAGA000000000] [SAÍDA] [06:36:47.940]
[06:36:47.960] -> [VAGA000000000] [ENTRADA] [06:36:47.960]
[06:36:47.970] -> [VAGA000000000] [SAÍDA] [06:36:47.970]
[06:36:47.990] -> [VAGA000000000] [ENTRADA] [06:36:47.990]
[06:36:48.000] -> [VAGA000000000] [SAÍDA] [06:36:48.000]
```

Figura 20 – Teste 2

No terceiro exemplo temos um carro que entrou às 06:41 e saiu rapidamente e pouco tempo depois mais dois carros fizeram o mesmo.



```
COM1
[06:41:19.444] -> [VAGA000000000] [ENTRADA] [06:41:19.444]
[06:41:19.450] -> [VAGA000000000] [SAÍDA] [06:41:19.450]
[06:41:19.460] -> [VAGA000000000] [ENTRADA] [06:41:19.460]
[06:41:19.470] -> [VAGA000000000] [SAÍDA] [06:41:19.470]
[06:41:19.480] -> [VAGA000000000] [ENTRADA] [06:41:19.480]
[06:41:19.490] -> [VAGA000000000] [SAÍDA] [06:41:19.490]
[06:41:19.510] -> [VAGA000000000] [ENTRADA] [06:41:19.510]
[06:41:19.520] -> [VAGA000000000] [SAÍDA] [06:41:19.520]
[06:41:19.540] -> [VAGA000000000] [ENTRADA] [06:41:19.540]
[06:41:19.550] -> [VAGA000000000] [SAÍDA] [06:41:19.550]
[06:41:19.570] -> [VAGA000000000] [ENTRADA] [06:41:19.570]
[06:41:19.580] -> [VAGA000000000] [SAÍDA] [06:41:19.580]
[06:41:19.600] -> [VAGA000000000] [ENTRADA] [06:41:19.600]
[06:41:19.610] -> [VAGA000000000] [SAÍDA] [06:41:19.610]
[06:41:19.630] -> [VAGA000000000] [ENTRADA] [06:41:19.630]
[06:41:19.640] -> [VAGA000000000] [SAÍDA] [06:41:19.640]
[06:41:19.660] -> [VAGA000000000] [ENTRADA] [06:41:19.660]
[06:41:19.670] -> [VAGA000000000] [SAÍDA] [06:41:19.670]
[06:41:19.690] -> [VAGA000000000] [ENTRADA] [06:41:19.690]
[06:41:19.700] -> [VAGA000000000] [SAÍDA] [06:41:19.700]
[06:41:19.720] -> [VAGA000000000] [ENTRADA] [06:41:19.720]
[06:41:19.730] -> [VAGA000000000] [SAÍDA] [06:41:19.730]
[06:41:19.750] -> [VAGA000000000] [ENTRADA] [06:41:19.750]
[06:41:19.760] -> [VAGA000000000] [SAÍDA] [06:41:19.760]
[06:41:19.780] -> [VAGA000000000] [ENTRADA] [06:41:19.780]
[06:41:19.790] -> [VAGA000000000] [SAÍDA] [06:41:19.790]
[06:41:19.810] -> [VAGA000000000] [ENTRADA] [06:41:19.810]
[06:41:19.820] -> [VAGA000000000] [SAÍDA] [06:41:19.820]
[06:41:19.840] -> [VAGA000000000] [ENTRADA] [06:41:19.840]
[06:41:19.850] -> [VAGA000000000] [SAÍDA] [06:41:19.850]
[06:41:19.870] -> [VAGA000000000] [ENTRADA] [06:41:19.870]
[06:41:19.880] -> [VAGA000000000] [SAÍDA] [06:41:19.880]
[06:41:19.900] -> [VAGA000000000] [ENTRADA] [06:41:19.900]
[06:41:19.910] -> [VAGA000000000] [SAÍDA] [06:41:19.910]
[06:41:19.930] -> [VAGA000000000] [ENTRADA] [06:41:19.930]
[06:41:19.940] -> [VAGA000000000] [SAÍDA] [06:41:19.940]
[06:41:19.960] -> [VAGA000000000] [ENTRADA] [06:41:19.960]
[06:41:19.970] -> [VAGA000000000] [SAÍDA] [06:41:19.970]
[06:41:19.990] -> [VAGA000000000] [ENTRADA] [06:41:19.990]
[06:42:00.000] -> [VAGA000000000] [SAÍDA] [06:42:00.000]
```

Figura 21 – Teste 3

5 CONCLUSÃO

Este trabalho apresentou uma ferramenta de seleção de motoristas em vagas de estacionamentos utilizando de características como idade e distância a fim de aferir quais motoristas ficariam com as vagas requeridas. Para isso foi feito um aplicativo para o cliente ver as vagas disponíveis e um aplicativo para o dono do estacionamento ver quais vagas estão ocupadas a fim de obter os resultados desejados com comodidade e segurança já que se fosse feito manualmente poderia haver problemas.

Para os testes do sistema foram feitos testes utilizando personagens fictícios para o momento da reserva de vaga a fim de ver como o programa se comporta nesse cenário e também foi utilizado personagem a fim de simular o gerenciamento do sistema no módulo do proprietário.

Este trabalho contribui para comodidade, segurança e distribuição eficiente de vagas em estacionamentos rotativos podendo ajudar a solucionar o problema da dificuldade de encontrar vagas.

5.1 TRABALHOS FUTUROS

A aplicação proporciona a visualização e a interação entre o cliente e o prestador de serviços em um ramo dinâmico da economia proporcionando comodidade para ambos os lados, porém é possível estender essa aplicação para outros ramos de atuação como shoppings e restaurantes.

Além disso uma outra possibilidade é a inclusão de pagamentos ser feito dentro do próprio sistema possibilitando comodidade e segurança para o cliente e o dono do estacionamento.

REFERÊNCIAS

- ARDUINO. *Arduino*. 2005. Disponível em: <<https://www.arduino.cc/>>. Citado na página 35.
- AUTOESPORTE. *Frota brasileira de veículos cresce 1,2% em 2017, diz Sindipeças*. 2017. Disponível em: <<https://g1.globo.com/carros/noticia/frota-brasileira-de-veiculos-cresce-12-em-2017-diz-sindipecas.ghtml>>. Citado 3 vezes nas páginas 12, 14 e 15.
- CRUZ, F. C. da; CRUZ, A. C. da; CERETTA, P. S. Mensuração da satisfação dos usuários do sistema municipal de estacionamento rotativo pago. 2016. Citado na página 15.
- CUNHA, D. T. L. da; BRITO, P. H. de S.; COUTINHO, E. F. Desenvolvimento e integração de uma aplicação móvel a um sistema distribuído: Estudo de caso de um sistema de gerenciamento de vagas. 2016. Citado na página 16.
- CYSNEIROS, L. M.; LEITE, J. C. S. do P. Utilizando requisitos não funcionais para análise de modelos orientados a dados. 1998. Citado na página 19.
- DUCHEIKO, F. F.; CASTRO, L. F. S. de; ALVES¹, G. V. Implementação de recursos em um smart parking baseado em sistemas multiagente. 2016. Citado na página 16.
- ESTADÃO. *Compra regular na internet é feita por quase 7 a cada 10 brasileiros, diz PwC*. 2018. Disponível em: <<https://istoe.com.br/compra-regular-na-internet-e-feita-por-quase-7-a-cada-10-brasileiros-diz-pwc/>>. Citado na página 13.
- FILHO, E. M. A.; MARTINEZ, A. L. Requisitos funcionais de um sistema de informações para gestão de custos no setor público. 2006. Citado na página 19.
- FRANÇOOSO, M. T.; MELLO, N. C. de. Influência dos aplicativos de smartphones para transporte urbano no trânsito. In: . [S.l.: s.n.], 2016. Citado na página 13.
- GLIFFY. *Gliffy*. 2019. Disponível em: <<https://www.gliffy.com/>>. Citado na página 18.
- HOLOVATY, A.; WILLISON, S. *Django Project*. 2019. Disponível em: <<https://www.djangoproject.com/>>. Citado na página 23.
- KON, F. *Uma Visão Geral de UML*. [S.l.], 2005. Citado na página 21.
- LAMAS, J. *Estacionamentos, os novos vilões da mobilidade urbana*. 2014. Disponível em: <<https://exame.abril.com.br/brasil/estacionamentos-os-novos-viloes-da-mobilidade-urbana/>>. Citado 2 vezes nas páginas 12 e 15.
- LYNCH, M.; SPERRY, B.; BRADLEY, A. *Ionic Framework*. 2019. Disponível em: <<https://ionicframework.com/>>. Citado na página 24.
- MICROSOFT. *Typescript*. 2016. Disponível em: <<https://www.typescriptlang.org/>>. Citado na página 24.
- PEREIRA, F. S. O sistema de estacionamento rotativo municipal: Estudo sobre a viabilidade econômica no ano de 2018 no município de Criciúma. 2018. Citado na página 13.

- RABELO, N. S. Utilização de bancos de dados e arquitetura cliente/servidor em ambientes comerciais. 1998. Citado na página 14.
- RIBAS, A. C.; GUERREIRO, V.; GUERREIRO, K. M. da S. A satisfação dos usuários do sistema de estacionamento rotativo zona azul e verde da cidade de caxias do sul. 2017. Citado na página 15.
- ROSSUM, G. van. *python3.6.9*. 2019. Disponível em: <<https://www.python.org/>>. Citado na página 23.
- SANTOS, M. dos et al. Uma nova maneira de estacionar veículos de passeio em grandes centros urbanos: proposta do aplicativo “minha vaga”. 2018. Citado na página 17.
- VEJA. *Congestionamento em São Paulo é o maior da história*. 2013. Disponível em: <<https://veja.abril.com.br/brasil/congestionamento-em-sao-paulo-e-o-maior-da-historia/>>. Citado na página 12.
- ZUPR. *Django Rest Framework*. 2019. Disponível em: <<https://www.django-rest-framework.org/>>. Citado na página 24.