



UNIVERSIDADE
EDUARDO
MONDLANE

FACULDADE DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA ELECTROTECNIA
LICENCIATURA EM ENGENHARIA INFORMÁTICA

TRABALHO DE LICENCIATURA

Tema:

**Proposta de um Sistema de Gestão de Vagas de Estacionamento baseado em
Internet of Things (IoT)**

**Caso de estudo: Parque Municipal de Estacionamento do Mercado Central –
Cidade de Maputo**

Autor:

- Cumbana, Hermenegildo Bernardo

Supervisor:

- eng^o. Ruben Manhiça

Maputo, Novembro de 2023



FACULDADE DE ENGENHARIA
DEPARTAMENTO DE ENGENHARIA ELECTROTECNIA
LICENCIATURA EM ENGENHARIA INFORMÁTICA

TRABALHO DE LICENCIATURA

Tema:

**Proposta de um Sistema de Gestão de Vagas de Estacionamento baseado em
Internet of Things (IoT)**

**Caso de estudo: Parque Municipal de Estacionamento do Mercado Central –
Cidade de Maputo**

Discente:

- Cumbana, Hermenegildo Bernardo

Supervisor:

- eng^o. Ruben Manhiça

Maputo, Novembro de 2023



UNIVERSIDADE
EDUARDO
MONDLANE

FACULDADE DE ENGENHARIA

DEPARTAMENTO DE ENGENHARIA ELECTROTECNIA

LICENCIATURA EM ENGENHARIA INFORMÁTICA

TERMO DE ENTREGA DE RELATÓRIO DO TRABALHO DE LICENCIATURA

Declaro que o estudante **Hermenegildo Bernardo Cumbana** entregou no dia ___ / ___ / 2023 as 03 cópias do relatório do seu Trabalho de Licenciatura com a referência: **2023EITLN201** intitulado: **Proposta de um sistema de gestão de vagas de estacionamento baseado em Internet of Things (IoT).**

Maputo, ___ de _____ de 2023

O Chefe da Secretaria

Dedico este trabalho a todos que sempre apoiaram-me e têm apoiado nesta caminhada, em especial aos meus pais, Bernardo Naiene Cumbane e Catarina Pedro Nhaquila.

Todos estes, significam muito para mim e são a força que preciso para lutar dia-pós-dia para poder alcançar os meus objectivos.

Agradecimentos

A conclusão deste processo não teria sido possível sem o apoio e incentivo de pessoas incríveis ao meu redor.

Primeiro, gostaria de agradecer à minha família, em especial a minha esposa Cristeza Dengo que sempre esteve ao meu lado, oferecendo suporte emocional e compreensão durante todo esse processo. Sua paciência e encorajamento foram fundamentais para minha perseverança.

Aos meus amigos e colegas, que compartilharam risadas nos momentos difíceis e celebraram as conquistas, agradeço por serem fontes valiosas de apoio social e por tornarem esta jornada acadêmica mais leve e memorável. Em especial, quero expressar minha profunda gratidão ao colega Abubacar Massuque, cuja amizade e colaboração foram inspiradoras e essenciais para a conclusão deste trabalho, seu apoio fez toda a diferença.

Um agradecimento especial ao meu supervisor Ruben Manhiça, pela orientação precisa, pelo comprometimento incansável e pelos valiosos insights que foram cruciais para o desenvolvimento deste trabalho. Sua dedicação e expertise foram fundamentais para o meu crescimento acadêmico.

Agradeço também a todos os professores e profissionais da instituição, cujo conhecimento e orientação foram fundamentais para a construção do meu conhecimento ao longo deste curso.

Este trabalho representa não apenas minha dedicação, mas também o apoio contínuo de todos vocês. Obrigado por fazerem parte desta jornada e por ajudarem a transformar um sonho acadêmico em realidade.

À todos, muito obrigado!

Resumo

O presente estudo focou-se na ineficiência na gestão de estacionamento no Parque Municipal do Mercado Central em Maputo, propondo uma solução baseada em IoT. Os objetivos incluíram analisar a gestão actual, comparar soluções IoT, criar um modelo integrado e desenvolver um protótipo funcional. Concluiu-se que a abordagem actual é ineficiente, resultando em congestionamentos e frustrações. A ausência de tecnologias modernas foi identificada como uma limitação nos estacionamentos urbanos.

A metodologia envolveu a criação de um modelo IoT para otimizar o uso do espaço e fornecer informações em tempo real. A implementação do protótipo foi bem-sucedida em um ambiente de teste, validando a viabilidade prática da solução proposta.

As conclusões destacaram a necessidade de um Sistema de Gestão de Vagas de Estacionamento baseado em IoT para melhorar a eficiência e a experiência do usuário em Maputo. Recomendações incluem a implementação do sistema no Parque Municipal, avaliação contínua, treinamento de pessoal e revisão de políticas de preços.

As referências incluem leis de trânsito locais, conceitos de estacionamento, e estudos relacionados à gestão de estacionamentos e IoT.

O estudo oferece uma abordagem inovadora para resolver os desafios de estacionamento em áreas urbanas, proporcionando benefícios tangíveis à comunidade e à gestão municipal.

Palavras-chave: "estacionamento urbano," "Internet das Coisas," "gestão de vagas de estacionamento".

Abstract

This study focused on the inefficiency in parking management at the Central Market Municipal Park in Maputo, proposing an IoT-based solution. Objectives included analyzing current management, comparing IoT solutions, creating an integrated model, and developing a functional prototype. It was concluded that the current approach is inefficient, leading to congestion and frustration. The absence of modern technologies was identified as a limitation in urban parking.

The methodology involved creating an IoT model to optimize space usage and provide real-time information. Prototype implementation was successful in a test environment, validating the practical viability of the proposed solution.

Conclusions highlighted the need for an IoT-based Parking Management System to enhance efficiency and user experience in Maputo. Recommendations include implementing the system in the Municipal Park, continuous evaluation, personnel training, and reviewing pricing policies.

References encompass local traffic laws, parking concepts, and studies related to parking management and IoT.

The study offers an innovative approach to address urban parking challenges, providing tangible benefits to the community and municipal management.

Keywords: "urban parking," "Internet of Things," "parking space management"

ÍNDICE

Agradecimentos	iii
Resumo.....	iv
Abstract.....	v
LISTA DE TABELAS.....	ix
LISTA DE FIGURAS.....	x
CAPÍTULO I – INTRODUÇÃO.....	1
1.1. Contextualização	1
1.2. Definição do Problema	2
1.3. Objectivos.....	2
1.3.1. Objectivo Geral	2
1.3.2. Objectivos Específicos.....	2
1.4. Metodologia	3
1.4.1. Metodologia do trabalho científico	3
1.4.2. Classificação da metodologia	3
1.4.3. Atividades desenvolvidas para alcançar os Objectivos Específicos.....	4
1.5. Estrutura do trabalho	7
CAPÍTULO II – REVISÃO DA LITERATURA.....	9
2.1. Estacionamento.....	9
2.1.1. Evolução Histórica do Estacionamento	9
2.1.2. Impacto do Estacionamento na Mobilidade Urbana	10
2.1.3. Tendências Globais em Soluções de Estacionamento.....	10
2.2. Parque de estacionamento.....	11
2.2.1. Sistemas para Localização de Vagas de Estacionamento	11
2.2.2. Tipos de Sistemas de Localização de Vagas de Estacionamento	12
2.2.3. Sistemas de Localização de Vagas de Estacionamento em Moçambique	14
2.2.4. Desafios Locais e Possíveis Soluções	14
2.3. Internet das Coisas.....	15
2.3.1. Arquitetura da IoT	16
2.3.2. Aplicações de IoT	17
2.4. Análise das diferentes Soluções de Localização de Vagas de Estacionamento baseado em IoT.....	19

2.4.1. Soluções de Localização de Vagas de Estacionamento baseado em IoT usando Câmeras Inteligentes	19
2.4.2. Soluções de Localização de Vagas de Estacionamento baseado em IoT usando Sensores de Radar a Laser	20
2.4.3. Soluções de Localização de Vagas de Estacionamento baseado em IoT usando Sensores Veículos Terrestres	21
2.4.4. Critérios para Comparação Localização de Vagas de Estacionamento baseado em IoT	22
2.4.5. Escolha da tecnologia	23
2.5. Sistemas de Localização de Vagas de Estacionamento baseado em IoT usando Sensores Veículos Terrestres	24
2.6. Principais vantagens de um Sistema de Gestão de Vagas de Estacionamento baseado em IoT	26
CAPÍTULO III – CASO DE ESTUDO	27
3.1. Parque de Estacionamento do Conselho Municipal	27
3.1.1. Localização	27
3.1.2. Organização	27
3.1.3. Descrição do processo actual (fluxo de entrada)	29
3.1.4. Descrição do processo actual (fluxo de saída)	31
3.2. Constrangimentos e limitações existentes do processo actual gestão de vagas do Parque de Estacionamento do Conselho Municipal	32
3.3. Proposta de solução	34
3.3.1. Descrição da proposta de solução	34
3.3.1.1. Diagrama de sequência (fluxo de entrada)	35
3.3.1.2. Diagrama de sequência (fluxo de saída)	35
3.3.2. Composição do Sistema	36
3.3.3. Princípio de Funcionamento	37
3.3.4. Custo de Implementação e Instalação do Sistema de estacionamento IoT	37
3.3.5. Benefícios da Solução	38
CAPÍTULO IV – DESENVOLVIMENTO DO PROTÓTIPO	39
4.1. Visão Geral do Sistema	39
4.2. Análise e definição de requisitos	39
4.2.1. Âmbito do Sistema	39

4.2.2.	Grupos de Interesse	40
4.2.3.	Convenções	40
4.2.3.1.	Identificação dos Requisitos.....	40
4.2.3.2.	Identificação dos Casos de Uso	40
4.2.4.	Requisitos do Sistema	40
4.2.4.1.	Prioridades dos Requisitos.....	41
4.2.5.	Requisitos funcionais	41
4.2.6.	Requisitos não funcionais	42
4.3.	Projecto de sistema e software	42
4.3.1.	Requisitos de software.....	42
4.3.2.	Requisitos de hardware	43
4.3.3.	Arquitectura.....	44
5.2.3.1.	Camada Física e Percepção.....	45
5.2.3.2	Camada de Rede.....	45
5.2.3.1	Camada de Aplicação.....	45
4.3.4.	Modelo de funcionamento.....	45
4.4.	Implementação e teste de unidade	46
4.4.1.	Implementação.....	47
4.4.1.1.	Configuração de Adafruit IO.....	47
4.4.1.2.	Programação do NodeMCU	54
4.5.	Integração e teste de sistema	59
4.5.1.	Integração de Componentes.....	60
4.6.	Operação e manutenção	60
4.6.1.	Implantação do Sistema	61
4.6.2.	Monitoria e Colecta de Dados.....	61
4.6.3.	Manutenção Proativa e Preventiva	62
4.6.4.	Actualizações e Melhorias	62
CAPÍTULO V – CONSIDERAÇÕES FINAIS E RECOMENDAÇÕES		63
5.1.	Conclusão.....	63
5.2.	Recomendações.....	64
5.3.	Referencias / Bibliografia.....	65
ANEXO 1: GUIÃO DA ENTREVISTA		A
ANEXO 2: CÓDIGO DO PROGRAMA		C

LISTA DE TABELAS

Tabela 1: Critérios para Comparação Sistemas de Localização de Vagas de Estacionamento baseado em IoT	23
Tabela 2: Custo de Implementação e Instalação do Sistema de estacionamento IoT	37
Tabela 3: Requisitos funcionais	41
Tabela 4: Requisitos não funcionais	42
Tabela 5: Implantação do Sistema	61
Tabela 6: Manutenção Proativa e Preventiva	62

LISTA DE FIGURAS

Figura 1: Número de dispositivos de IoT conectados 2015-2025	16
Figura 2: Diagrama de Arquitetura de Internet das Coisas	16
Figura 3: Câmera Inteligente em Nova Iorque.....	19
Figura 4: Sensor de Radar a Laser em Tóquio	20
Figura 5: Sensor IoT de Veículos Terrestres em Barcelona.....	21
Figura 6: Organograma do parque	28
Figura 7: Fluxo de entrada de viaturas no Parque.....	29
Figura 8: Fluxo de saída de viaturas no Parque	31
Figura 9: Fluxo de entrada de viaturas no Parque (sistema proposto).....	35
Figura 10: Fluxo de saída de viaturas no Parque (sistema proposto)	35
Figura 11: Sensores de Infravermelho (IR)	36
Figura 12: Servo Motores	36
Figura 13: ESP8266	36
Figura 14: Plataforma Adafruit IO	36
Figura 15: Arquitectura do modelo de solução	44
Figura 16: Pagina Principal do site Adafruit.....	47
Figura 17: Obtenção da Chave AIO	48
Figura 18: AIO Key	48
Figura 19: Acesso a aba criar um feed	49
Figura 20: Campo de criação de um feed	49
Figura 21: Acesso a aba criar um dashboard	50
Figura 22: Campo de criação de uma dashboard	50
Figura 23: View de dashboards	51
Figura 24: Criação de novo bloco.....	51
Figura 25: Tipos de blocos	52
Figura 26: Feeds criados	52
Figura 27: Configuração de bloco.....	53
Figura 28: Dashboard do admin	54
Figura 29: Acesso a menu preferences do Arduino IDE	54
Figura 30: Campo de add de URL.....	55
Figura 31: Instalação de drives da board	55
Figura 32: Selecção de drives de ESP8266	56
Figura 33: Diagrama de Circuitos	60

LISTA DE SIGLAS E ABREVIATURAS

AIO: Adafruit IO

API REST: Interface de Programação de Aplicativos Representacionais

API: Interface de Programação de Aplicativos

CM: Conselho Municipal

CRUD: Create, Read, Update, Delete (Criar, Ler, Atualizar, Excluir)

E/S ou I/O: Input/Output

GPS: Sistema de Posicionamento Global

GSM: Sistema Global para Comunicações Móveis

HTTP: Protocolo de Transferência de Hipertexto

IDE: Integrated Development Environment

IEEE: Institute of Electrical and Electronic Engineers

IoT: Internet of Things

IP: Protocolo da Internet

IR: Sensor Infravermelho

JSON: Notação de Objetos JavaScript

LED: Díodo Emissor de Luz

MCU: Microcontrolador

MQTT: Protocolo de Telemetria de Mensagem de Fila

NTP: Network Time Protocol

PCB: Placa de Circuito Impresso

QoS: Qualidade de Serviço

RF: Requisito funcional

RFID: Identificação por Radiofrequência

RNF: Requisito não funcional

SQL: Linguagem de Consulta Estruturada

TCP: Protocolo de Controle de Transmissão

UC/CU: Casos de Uso

UDP: Protocolo de Datagrama de Usuário

UI: Interface do Usuário

UX: User Experience

Wi-Fi: Wireless Fidelity

WLAN: Wireless Local Area Network

CAPÍTULO I – INTRODUÇÃO

1.1. Contextualização

O Parque Municipal de Estacionamento do Mercado Central em Maputo enfrenta um desafio premente relacionado à gestão ineficiente de suas vagas de estacionamento, o que se traduz em congestionamentos frequentes e frustrações para os motoristas. Este problema torna-se ainda mais acentuado em virtude do rápido crescimento populacional observado na região, resultando em um aumento significativo no número de veículos que competem por um espaço limitado (Jones et al., 2020).

A falta de uma abordagem sistemática na atribuição e monitoramento das vagas de estacionamento tem levado a situações onde o estacionamento atinge sua capacidade máxima, deixando muitos motoristas em busca desesperada por espaços disponíveis. Esse cenário não apenas contribui para o congestionamento do tráfego nas proximidades do Mercado Central, mas também gera tensões entre os usuários do estacionamento, impactando negativamente a experiência urbana.

A literatura ressalta que problemas de estacionamento mal geridos não apenas causam transtornos para os cidadãos, mas também têm implicações econômicas, uma vez que a circulação suave do tráfego é essencial para o funcionamento eficiente das actividades comerciais na área central da cidade (Greenberg, 2021). A ausência de sistemas automatizados e tecnologias modernas para mitigar esses desafios tem sido uma limitação evidente no panorama actual de estacionamentos urbanos (Johnson, 2017).

Ao focar no estudo deste problema específico, o presente trabalho busca abordar lacunas críticas na gestão de vagas de estacionamentos urbanos, propondo uma solução inovadora baseada em tecnologias emergentes. A introdução de dispositivos IoT e a aplicação de algoritmos avançados visam não apenas otimizar o uso do espaço, mas também resolver questões fundamentais que afetam diretamente a qualidade de vida dos cidadãos que dependem do Parque Municipal de Estacionamento do Mercado Central em Maputo.

1.2. Definição do Problema

O problema essencial é a falta de um sistema eficiente que permita aos usuários obter informações em tempo real sobre a disponibilidade de vagas de estacionamento na cidade de Maputo. A ausência desse sistema dificulta a capacidade dos motoristas/condutores de saber se uma vaga de estacionamento está ocupada ou disponível antes de se deslocarem para um determinado local. Isso resulta em frustrações, desperdício de tempo e recursos, uma vez que os usuários não têm a capacidade de tomar decisões informadas ao buscar estacionamento. Portanto, a necessidade premente é possibilitar aos usuários acesso a informações actualizadas sobre a disponibilidade de vagas de estacionamento, tornando o processo de estacionamento na cidade mais conveniente e eficiente.

1.3. Objectivos

Os objectivos delineados neste capítulo visam orientar o desenvolvimento e a implementação de um Sistema de Gestão de Vagas de Estacionamento baseado em Internet das Coisas (IoT) no Parque Municipal de Estacionamento do Mercado Central, na cidade de Maputo. Estes objetivos específicos direccionam o estudo para compreender, projectar e implementar uma solução que atenda às necessidades de gestão de estacionamento na região.

1.3.1. Objectivo Geral

- ✓ Propor um Sistema de Gestão de Vagas de Estacionamento baseado em Internet of Things (IoT) no Parque Municipal de Estacionamento do Mercado Central – Cidade de Maputo

1.3.2. Objectivos Específicos

- ✓ Descrever como é feita a gestão da disponibilidade das vagas de estacionamento no parque municipal de estacionamento do mercado central actualmente;
- ✓ Pesquisar e comparar soluções de IoT que possam atender aos requisitos e especificidades do caso de estudo;
- ✓ Criar um modelo que inclui componentes de Software e Hardware baseado em IoT para gestão de Vagas no parque municipal de estacionamento do mercado central

- ✓ Desenvolver um protótipo funcional que sirva para testar o modelo de gestão de Vagas no parque municipal de estacionamento do mercado central baseado em IoT.

1.4. Metodologia

Neste capítulo, é apresentada a metodologia adotada para a realização do estudo, incluindo a classificação da metodologia e as actividades desenvolvidas para alcançar os objetivos estabelecidos.

1.4.1. Metodologia do trabalho científico

De modo a alcançar os objetivos traçados para realização do presente trabalho, serão tomados como bases os objetivos específicos para que de forma mais efectiva concretizar o objetivo geral. São identificadas as técnicas metodológicas e os processos tomados para elaboração do projecto. De acordo com Albuquerque (2015) a tipologia da pesquisa pode ser classificada quanto à natureza, aos objetivos e aos procedimentos.

1.4.2. Classificação da metodologia

❖ *Quanto à natureza*

A pesquisa é classificada como qualitativa, seguindo a definição de Creswell (2013), que afirma que a pesquisa qualitativa permite uma compreensão aprofundada e a exploração de experiências e percepções. O caso de estudo usa teorias e conhecimentos gerados em informática e algumas bases de electrónica com o intuito de desenvolver um sistema que vai resolver um problema específico.

❖ *Quanto aos objetivos*

A pesquisa é exploratória e descritiva. Segundo Gil (2010), a pesquisa exploratória visa a compreensão inicial de um problema complexo, enquanto a pesquisa descritiva busca descrever as características de determinado fenómeno ou relação.

❖ *Quanto aos procedimentos técnicos*

A pesquisa é experimental, envolvendo o desenvolvimento de um protótipo funcional. De acordo com Kerlinger (1986), a pesquisa experimental envolve a manipulação de variáveis para testar hipóteses e compreender relações de causa e efeito.

❖ *Quanto à abordagem*

A pesquisa adopta uma abordagem qualitativa, conforme recomendado por Creswell (2013), que destaca a pesquisa qualitativa como a interpretação de dados narrativos, percepções e análise para dar resposta ao problema levantado.

❖ **Quanto a técnicas de recolha de dados**

A pesquisa utiliza técnicas de colecta de dados quantitativas e qualitativas, alinhadas com a definição de Creswell (2013), Os dados usados para a pesquisa foram colhidos em base experimental. O tipo de tecnologia usada e as condições do parque de estacionamento são factores que podem influenciar nos resultados da pesquisa. Por isso foram experimentados e colhidos factores que pudessem fornecer os resultados esperados.

❖ **Quanto à técnicas de análise de dados**

A análise de dados é realizada com métodos quantitativos e qualitativos, em conformidade com Creswell (2013), que descreve a análise quantitativa como envolvendo estatísticas descritivas e a análise qualitativa como a interpretação de dados narrativos.

1.4.3. Atividades desenvolvidas para alcançar os Objetivos Específicos

Para atender aos objetivos específicos do estudo, foram realizadas as seguintes actividades:

- ✓ **Objectivo Específico 1:** *Descrever como é feita actualmente a gestão do sistema de estacionamento.*

O autor tinha como objectivo compreender a gestão actual das vagas de estacionamento no local de estudo. Para alcançar esse objectivo, as actividades desenvolvidas incluíram:

- Revisão da literatura sobre sistemas de gestão de estacionamento, fornecendo uma base de conhecimento.
- Entrevista com o gestor do parque de estacionamento para colectar informações detalhadas sobre as práticas de gestão.
- Investigação em sites relacionados para obter informações adicionais e contexto.

- ✓ **Objectivo Específico 2:** *Pesquisar e comparar soluções de IoT que possam atender aos requisitos e especificidades do caso de estudo.*

O autor buscou pesquisar e comparar soluções baseadas em IoT que se adequassem às necessidades do caso de estudo. Para alcançar esse objectivo, as actividades desenvolvidas incluíram:

- Revisão da literatura sobre IoT e suas aplicações em sistemas de gestão de estacionamento.
- Identificação de critérios específicos para avaliar as soluções de IoT.
- Comparação das soluções disponíveis no mercado em relação aos critérios definidos.

Objectivo Específico 3: *Criar um modelo que inclui componentes de Software e Hardware baseado em IoT para gestão de vagas no parque municipal de estacionamento do mercado central.*

O autor pretendia desenvolver um modelo completo que combinasse componentes de software e hardware baseados em IoT. Para alcançar esse objectivo, as actividades desenvolvidas incluíram:

- Definição dos requisitos de hardware e software do sistema.
- Projectar um modelo que integra esses componentes, considerando as especificidades do caso de estudo.

Objectivo Específico 4: *Desenvolver um protótipo funcional que sirva para testar o modelo de gestão de vagas no parque municipal de estacionamento do mercado central baseado em IoT.*

O autor tinha como objectivo criar um protótipo funcional que permitisse testar o modelo proposto. Para alcançar esse objectivo, as actividades desenvolvidas incluíram:

- Implementação do modelo desenvolvido, incluindo hardware e software.
- Teste do protótipo em um ambiente de estacionamento de teste para avaliar seu desempenho.

Para desenvolver um protótipo para o problema identificado usando os princípios de engenharia de software de forma a garantir a qualidade do produto final, entretanto foi usada a metodologia Waterfall (Cascata).

O modelo em cascata é proposta de desenvolvimento de projecto, na qual é preciso terminar uma etapa para começar outra. Trata-se de um modelo de desenvolvimento de software sequencial e sistemático, no qual o desenvolvimento é visto como um fluir constante para frente, como uma cascata. Segundo SOMMERVILLE (2011) existem cinco fase (ou etapas) do Ciclo de vida do software:

✓ **Análise e definição de requisitos:**

Nesta etapa do ciclo faz-se a recolha dos requisitos do sistema e dos usuários que utilizarão o sistema, a listagem e entendimento dos serviços, restrições e objetivos do sistema, que devem ser definidos detalhadamente e servirem como uma especificação para o projecto.

✓ **Projecto de sistema e software:**

Para esta fase, se estabelece uma arquitectura geral do sistema que envolve a identificação e a descrição da estrutura de dados, detalhes procedimentos, relacionamentos e caracterização da interface. Como os requisitos, o projecto é documentado e torna-se parte da configuração do software.

✓ **Implementação e teste de unidade:**

Durante essa etapa, o projeto de software é realizado como um conjunto de programas, que são codificados em linguagem que a máquina entenda. O teste unitário faz parte do processo de codificar onde é preciso testar o que foi feito para saber se está funcionando adequadamente ao esperado por aquele sistema e, então, seguir com a codificação.

✓ **Integração e teste de sistema:**

Nesta etapa, depois de todo o código for gerado, inicia-se a realização de testes para verificar se os aspectos funcionais estão de acordo com os requisitos e sem erros. Após os testes, o software é liberado para o cliente.

✓ **Operação e manutenção:**

Depois do software ser instalado e colocado em operação, o processo de manutenção começa. Envolve a correcção de erros não detectados nos estágios anteriores e a adaptação à medida que novos requisitos são identificados. A manutenção re replica cada uma das etapas do método em cascata a um software existente, ao invés de um novo.

1.5. Estrutura do trabalho

Para uma melhor organização e compreensão dos conceitos abordados, estruturou-se o trabalho em cinco capítulos.

Capítulo I – Introdução

O capítulo apresenta uma introdução detalhada ao tema, abordando a contextualização do problema no Parque Municipal de Estacionamento do Mercado Central em Maputo. Destaca-se o desafio da gestão ineficiente de vagas, agravado pelo rápido crescimento populacional. O problema central é a falta de um sistema eficiente para informar os motoristas sobre a disponibilidade de estacionamento em tempo real. Os objectivos incluem a proposição de um Sistema de Gestão de Vagas baseado em IoT, e a metodologia adoptada é qualitativa, exploratória e experimental, seguindo o modelo em cascata do ciclo de vida do software.

Capítulo II – Revisão de literatura

O Capítulo II aborda a relevância do estacionamento na gestão urbana, destacando a distinção entre estacionamento e paragem. Explora a evolução histórica, impactos na mobilidade urbana e tendências globais em soluções de estacionamento. Examina o conceito de parques de estacionamento e sistemas de localização de vagas, com foco em tendências, desafios e soluções em Moçambique. Introduce a Internet das Coisas (IoT) e sua arquitectura, evidenciando aplicações variadas. Analisa três soluções de localização de vagas baseadas em IoT: Câmaras Inteligentes, Sensores de Radar a Laser e Sensores de Veículos Terrestres. A comparação considera critérios como precisão, integração, escalabilidade, consumo de energia e custos. O capítulo destaca a importância de escolher tecnologias alinhadas às necessidades locais para uma gestão eficiente e sustentável de estacionamentos urbanos.

Capítulo III – Caso de estudo

O Capítulo III apresenta um estudo de caso focado no Parque de Estacionamento do Conselho Municipal em Maputo. Descreve a localização, organização e processo atual de gestão de vagas, destacando o organograma e o fluxo operacional. Identifica limitações, como a falta de informações em tempo real, alocação manual de vagas e complexidade na verificação da disponibilidade. Propõe uma solução baseada em IoT para melhorar eficiência e experiência, detalhando a composição do sistema com sensores, NodeMCU e plataforma Adafruit IO. O princípio de funcionamento, custo de implementação e benefícios são discutidos, visando modernizar e otimizar o estacionamento. O custo estimado é de 147,500MT, e os benefícios incluem disponibilidade em tempo real, conveniência, eficiência operacional, segurança aprimorada e monitoria remota.

Capítulo IV – Desenvolvimento do protótipo

O Capítulo IV aborda o desenvolvimento do Sistema de Gestão de Vagas de Estacionamento IoT para o Parque Municipal de Estacionamento do Mercado Central em Maputo. Destaca componentes como sensores IR, servo motores e NodeMCU, define requisitos, prioridades, requisitos funcionais e não funcionais, detalha o projecto de sistema e software, e apresenta a implementação do protótipo com a configuração do Adafruit IO. O capítulo destaca a integração e teste do sistema, além de abordar a operação, manutenção e implantação do sistema.

Capítulo V– Considerações finais e recomendações

O Capítulo V destaca a conclusão do estudo sobre gestão de estacionamento em Maputo, propondo solução IoT. Objectivos foram alcançados com sucesso, proporcionando inovação na gestão de vagas. Recomendações incluem a implementação do sistema proposto e ajustes contínuos, enquanto referências abrangem legislação, conceitos de IoT, e estudos relevantes sobre gestão de estacionamentos.

CAPÍTULO II – REVISÃO DA LITERATURA

2.1. Estacionamento

De acordo com o Artigo 49, ponto nr.2 da Lei nº 2/97 de 18 de fevereiro que aprova a Postura de Transito do Município de Maputo, considera-se estacionamento a imobilização de um veículo que não constitua paragem e que não seja motivada por circunstâncias próprias da circulação. Esta definição legal destaca a distinção entre estacionamento e paragem, evidenciando que o primeiro envolve a permanência prolongada de um veículo em um local específico, sem razões directamente relacionadas ao fluxo de tráfego.

A gestão eficaz de vagas de estacionamento torna-se, assim, um elemento crucial na administração urbana, impactando não apenas a fluidez do tráfego, mas também questões relacionadas ao desenvolvimento sustentável, eficiência energética e qualidade de vida nas áreas urbanas. Ao longo deste capítulo, exploraremos a evolução dos conceitos de estacionamento, desde sua definição legal até as soluções inovadoras baseadas em Internet das Coisas (IoT) para otimização de vagas, com foco especial no contexto moçambicano.

2.1.1. Evolução Histórica do Estacionamento

A história dos espaços de estacionamento é intrinsecamente ligada à evolução do modo como as sociedades se deslocam. Inicialmente, os estacionamentos eram improvisados, nas ruas e em garagens de residências mais abastadas. Segundo informações da BrasilPark, ao longo do tempo, com o crescimento exponencial de veículos e actividades urbanas, a demanda por espaços de estacionamento tornou-se um desafio. O estacionamento nas ruas, uma prática inicialmente comum, revelou-se inadequado para acomodar o aumento na frota de veículos e a densidade populacional nas áreas urbanas.

A transição para soluções mais estruturadas, como terrenos e edifícios de estacionamento, marcou uma mudança significativa na abordagem da sociedade em relação à mobilidade. De acordo com BrasilPark, a evolução dos espaços de estacionamento reflete não apenas o desenvolvimento tecnológico, mas também as mudanças nas expectativas sociais em relação à acessibilidade e conveniência.

2.1.2. Impacto do Estacionamento na Mobilidade Urbana

O impacto do estacionamento na mobilidade urbana é uma questão complexa que vai além da simples disponibilidade de espaços. Segundo informações do Chivalry Club postadas por Lorena Cavalcanti, à medida que as cidades crescem, a busca por estacionamento influencia directamente a circulação viária. O fenómeno de veículos buscando espaços disponíveis não apenas gera congestionamentos, mas também contribui para emissões de poluentes e aumenta o tempo de viagem.

No entanto, reconhecer o impacto do estacionamento também abre portas para soluções inovadoras. Conforme apontado por Chivalry Club, a implementação de tecnologias avançadas, como sistemas de localização de vagas e aplicativos de estacionamento, pode melhorar significativamente a eficiência do uso desses espaços. Além disso, estratégias que promovem o compartilhamento de veículos e o estímulo ao uso de transporte público contribuem para reduzir a demanda por estacionamentos individuais, aliviando a pressão sobre a mobilidade urbana.

2.1.3. Tendências Globais em Soluções de Estacionamento

As tendências globais em soluções de estacionamento refletem a busca contínua por inovação e sustentabilidade nas áreas urbanas. De acordo com BrasilPark, tecnologias como sensores inteligentes, IoT e aplicativos de estacionamento vêm transformando a maneira como as pessoas encontram e utilizam espaços de estacionamento. A transição para estacionamentos mais eficientes e ecológicos está em ascensão, com iniciativas que incluem estacionamentos verdes, que incorporam áreas de vegetação e práticas sustentáveis.

A integração de sistemas inteligentes não apenas melhora a experiência do usuário, mas também contribui para uma gestão mais eficaz dos espaços disponíveis. Além disso, a tendência crescente em direcção a estacionamentos compartilhados e modelos de negócios inovadores sinaliza uma mudança nas percepções tradicionais de propriedade de espaços de estacionamento. Segundo informações da BrasilPark, essas tendências refletem não apenas avanços tecnológicos, mas também uma conscientização crescente sobre a importância de criar ambientes urbanos mais sustentáveis e eficientes.

2.2. Parque de estacionamento

O conceito de parques de estacionamento representa um elemento essencial na gestão urbana, desempenhando um papel crucial na optimização dos espaços e na melhoria da mobilidade. Como ressalta Donald Shoup, especialista em planeamento urbano, em seu livro "The High Cost of Free Parking," os parques de estacionamento bem planeados são essenciais para equilibrar a oferta e demanda de espaços, evitando congestionamentos e promovendo uma circulação viária mais eficiente. A importância desses parques transcende a mera comodidade, influenciando directamente a qualidade de vida nas áreas urbanas.

A evolução dos parques de estacionamento ao longo do tempo é destacada por Jane Jacobs, urbanista renomada, em "The Death and Life of Great American Cities." Inicialmente simples extensões de terreno, esses espaços evoluíram para estruturas mais complexas à medida que as cidades cresceram, refletindo a necessidade de uma abordagem mais planejada para a gestão urbana. A visão geral dos parques de estacionamento, portanto, revela não apenas sua função prática, mas também sua importância na configuração e funcionalidade das cidades modernas.

2.2.1. Sistemas para Localização de Vagas de Estacionamento

A implementação de sistemas para localização de vagas de estacionamento representa um avanço significativo na gestão eficiente desses espaços. Conforme destacado por Susan Shaheen, professora de Engenharia Civil e Ambiental na Universidade da Califórnia, Berkeley, em suas pesquisas sobre mobilidade urbana, sistemas avançados de localização de vagas, como sensores inteligentes e aplicativos móveis, têm o potencial de reduzir significativamente o tempo gasto pelos motoristas procurando por estacionamento. Essas tecnologias não apenas melhoram a experiência do usuário, mas também contribuem para a redução do congestionamento viário.

Além disso, o estudo "Smart Parking Systems Market - Growth, Trends, and Forecast" destaca a relevância desses sistemas para a optimização do uso de espaços urbanos. Ao oferecer informações em tempo real sobre a disponibilidade de vagas, os sistemas facilitam uma abordagem mais eficaz na gestão dos parques de estacionamento, promovendo uma distribuição mais equitativa dos recursos urbanos.

2.2.2. Tipos de Sistemas de Localização de Vagas de Estacionamento

Ao analisar os diferentes tipos de sistemas de localização de vagas de estacionamento, é evidente que cada abordagem possui vantagens específicas. Como mencionado no relatório da International Parking & Mobility Institute, sensores de ocupação oferecem uma precisão valiosa, permitindo uma visão em tempo real da utilização de cada vaga. Por outro lado, plataformas colaborativas, como o aplicativo Waze, baseiam-se na contribuição da comunidade para atualizar informações sobre vagas disponíveis, promovendo a participação cidadã na gestão urbana.

✓ Sensores de Ocupação

Sensores de ocupação, como descrito por Susan Shaheen em suas pesquisas sobre mobilidade urbana (Shaheen, 2021), representam uma inovação crucial na gestão de estacionamentos urbanos. Esses dispositivos utilizam tecnologia avançada para monitorar em tempo real a ocupação de cada vaga, proporcionando dados precisos que facilitam uma distribuição eficiente dos espaços disponíveis. A precisão desses sensores não apenas melhora a experiência do usuário, permitindo a localização rápida de vagas, mas também contribui para a redução do congestionamento viário.

✓ Plataformas Colaborativas

O paradigma colaborativo, exemplificado pelo aplicativo Waze, destaca a importância da participação activa da comunidade na gestão de estacionamentos (Cavalcanti, 2022). Ao permitir que os usuários contribuam com informações em tempo real sobre a disponibilidade de vagas, essas plataformas criam uma rede dinâmica e interativa. A pesquisa de Lorena Cavalcanti, compartilhada pelo Chivalry Club, enfatiza como a colaboração entre os motoristas pode não apenas otimizar a utilização de estacionamentos, mas também melhorar a eficiência do tráfego e reduzir o impacto ambiental associado a busca desordenada por vagas.

✓ Aplicativos Móveis Específicos

Aplicativos móveis dedicados à localização de vagas, conforme evidenciado por estudos da International Parking & Mobility Institute (IPMI, 2020), transformam a experiência de estacionamento para os usuários urbanos. Essas soluções não apenas informam sobre a disponibilidade de vagas, mas também oferecem

funcionalidades como navegação direcionada e, em alguns casos, reserva antecipada. Ao integrar tecnologia móvel e dados em tempo real, esses aplicativos simplificam o processo de estacionamento, tornando-o mais conveniente e eficiente para os motoristas.

✓ **Sistemas de Pagamento Eletrônico**

A integração de sistemas de pagamento eletrônico com a localização de vagas, conforme destacado por especialistas da indústria (BrasilPark, 2022), representa um passo significativo para a modernização da gestão de estacionamentos urbanos. Além de oferecer informações sobre a disponibilidade de vagas, esses sistemas permitem que os usuários efetuem pagamentos eletronicamente, eliminando a necessidade de transações em dinheiro físico. Isso não apenas simplifica a experiência do usuário, mas também contribui para a redução do tempo gasto em processos de pagamento, melhorando a fluidez do tráfego.

✓ **Sistemas Integrados IoT (Internet das Coisas)**

A aplicação de tecnologias IoT nos sistemas de localização de vagas proporciona uma gestão mais inteligente e conectada dos estacionamentos urbanos (BrasilPark, 2022). Ao utilizar sensores e dispositivos conectados para monitorar a ocupação em tempo real, esses sistemas oferecem dados precisos para uma melhor tomada de decisões. O relatório da BrasilPark destaca como a integração de IoT não apenas aprimora a experiência do usuário, mas também contribui para uma gestão mais eficaz dos espaços disponíveis, promovendo a eficiência no uso dessas áreas urbanas.

✓ **Plataformas de Reservas Online**

Plataformas de reservas online, conforme discutido no estudo "Smart Parking Systems Market - Growth, Trends, and Forecast" (Futuro Insights, 2021), proporcionam uma abordagem inovadora para a gestão de estacionamentos. Permitindo que os usuários reservem vagas com antecedência, essas plataformas oferecem uma solução conveniente para a demanda crescente por espaços em áreas movimentadas. Ao integrar-se com sistemas de localização em tempo real, essas plataformas fornecem informações precisas sobre a disponibilidade, melhorando a eficácia na gestão do estacionamento.

✓ **Sistemas de Reconhecimento de Matrículas**

A implementação de sistemas de reconhecimento de matrículas, como destacado em estudos de mobilidade urbana (Future Transportation, 2022), representa uma evolução na automação da gestão de estacionamentos. Utilizando câmeras para identificar matrículas de veículos, esses sistemas não apenas informam sobre a ocupação de vagas, mas também facilitam processos como controle de acesso e segurança. A integração eficiente desses sistemas com a localização de vagas contribui para uma abordagem mais eficaz na gestão dos espaços urbanos.

✓ **Soluções de Estacionamento Autônomo**

Soluções de estacionamento autônomo, alinhadas às tendências globais, introduzem inovações radicais na gestão de espaços urbanos. Incorporando tecnologias avançadas, como veículos autônomos, esses sistemas buscam otimizar o estacionamento de forma autônoma e eficiente (BrasilPark, 2022). A implementação bem-sucedida dessas soluções não apenas reduz a dependência de recursos humanos na gestão de estacionamentos, mas também promove uma abordagem mais sustentável e eficaz na utilização de espaços urbanos.

Cada tipo de sistema apresenta desafios únicos, como custos de implementação e manutenção. No entanto, a diversidade de opções destaca a necessidade de uma abordagem personalizada ao escolher o sistema mais adequado para cada contexto urbano específico. A flexibilidade na escolha dessas tecnologias é crucial para garantir uma gestão eficiente dos parques de estacionamento.

2.2.3. Sistemas de Localização de Vagas de Estacionamento em Moçambique

O cenário de estacionamento em Moçambique, como discutido no relatório do Ministério dos Transportes e Comunicações, destaca desafios específicos. O rápido crescimento urbano e a falta de infraestrutura adequada contribuem para a pressão sobre os parques de estacionamento, resultando em congestionamentos e dificuldades na gestão eficaz desses espaços.

2.2.4. Desafios Locais e Possíveis Soluções

Conforme apontado por especialistas locais, como o urbanista Alberto Cossa, os desafios em Moçambique incluem a falta de conscientização sobre práticas de

estacionamento sustentáveis, a necessidade de infraestrutura aprimorada e a integração de tecnologias acessíveis à realidade local. Estratégias educacionais, investimentos em infraestrutura adaptada à expansão urbana e parcerias com empresas de tecnologia que compreendem as nuances do contexto moçambicano são cruciais. Ao abordar esses desafios, Moçambique pode desenvolver soluções adaptadas à sua realidade, promovendo uma gestão mais eficiente e sustentável dos parques de estacionamento.

2.3. Internet das Coisas

A Internet das Coisas, abreviada como IoT, engloba a concepção global de objectos do cotidiano, como eletrodomésticos, móveis e veículos, dotados de capacidade de leitura, reconhecimento, localização e endereçamento por meio de dispositivos de colecta de informações, além de serem controláveis através da Internet. Essa conectividade independe dos meios de comunicação utilizados, podendo ocorrer via RFID, LAN sem fio, redes de longa distância ou outros meios (Patel & Patel, 2016). Cada "coisa" associada a um endereço IP permite a transferência de dados por meio de uma rede.

Conforme destacado por Patil et al. (2012), essa abordagem proporcionará a base para diversas aplicações inovadoras, tais como monitoramento de energia, sistemas de segurança de transporte e segurança predial. O número de dispositivos conectados à Internet, que engloba máquinas, sensores e câmaras integrantes da IoT, continua a aumentar de forma consistente. Segundo o relatório elaborado pela IoT Analytics (2018), prevê-se um crescimento anual de 10% na contagem de dispositivos de 2018 a 2025. O documento ainda estima que essa contagem pode atingir 21,5 bilhões até 2020.

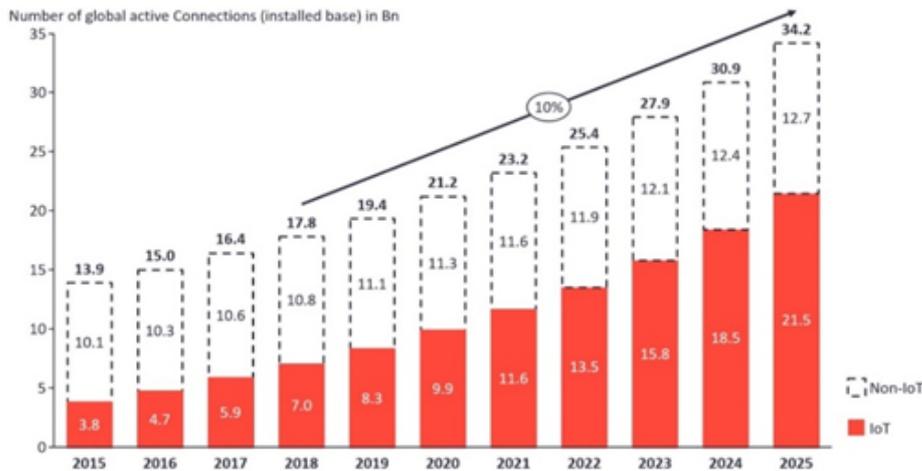


Figura 1: Número de dispositivos de IoT conectados 2015-2025
 Fonte: IoT Analytics (2018)

2.3.1. Arquitetura da IoT

A arquitetura da Internet das Coisas (IoT) está estruturada em quatro camadas, compostas por elementos e funções distintas que prevalecem independentemente do tipo de sistema ou rede de IoT em execução.

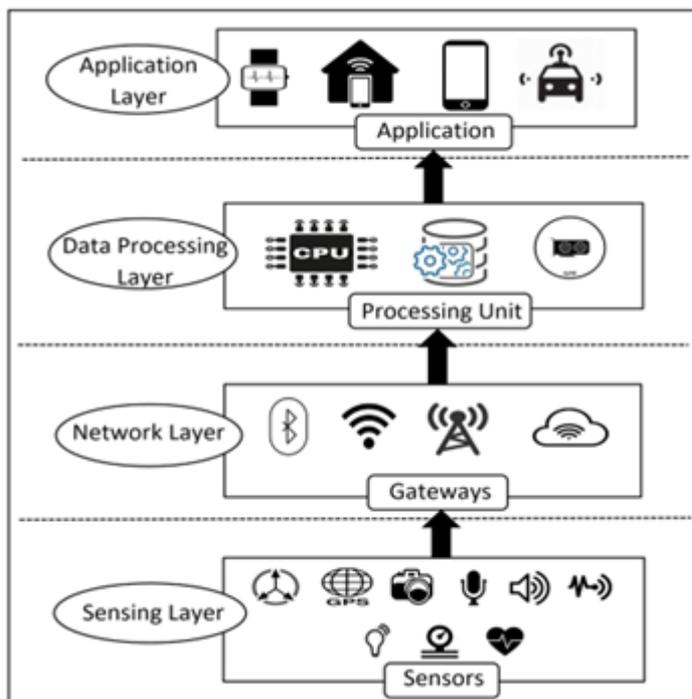


Figura 2: Diagrama de Arquitetura de Internet das Coisas
 Fonte: Joan (2018).

A primeira camada é a camada física, constituída por sensores responsáveis por detectar e colectar dados do ambiente. Seu papel abrange a colecta de parâmetros físicos ou a identificação de outros objetos inteligentes. A seguir, temos os

actuadores, que têm a capacidade de provocar mudanças no ambiente. Por exemplo, um sensor pode perceber alterações nas temperaturas e baixa luminosidade, levando um atuador a ligar automaticamente as luzes da rua ao anoitecer.

A segunda camada é a camada de gateway da internet. Os dados provenientes dos sensores precisam ser preparados antes de ingressar na etapa final de processamento. Essencialmente, essa camada realiza a agregação e conversão de dados do formato analógico para o digital. Um gateway de Internet encaminha esses dados por meio de WLANs ou outras redes para processamento adicional.

Os dados coletados entram nos sistemas de computação para execução de processamento e análise adicionais. Embora as duas camadas mencionadas estejam localizadas fisicamente no dispositivo, um sistema de processamento é necessário. Dada a magnitude dos dados da IoT, enviá-los directamente para o Data Center ou servidor pode consumir significativa largura de banda de rede, sobrecarregando seus recursos (Stack, 2018). Portanto, os sistemas realizam análises para reduzir a carga na infraestrutura principal.

A camada de aplicação é crucial para os usuários, pois actua como uma interface que fornece os módulos necessários para controlar e monitorar vários aspectos do sistema IoT. As aplicações permitem aos usuários visualizar e analisar o status do sistema no estágio actual de acção, e por vezes, oferecem previsões de perspectivas futuras.

2.3.2. Aplicações de IoT

A Internet das Coisas (IoT) tem diversas aplicações em setores variados, como agricultura, transporte, saúde, meio ambiente, gestão da cadeia de suprimentos, monitoramento de infraestrutura, entre outros. Abaixo, estão algumas destas aplicações:

✓ Gestão de Vagas de Estacionamento

- **Sensores de Ocupação:** Sensores instalados em vagas de estacionamento para monitorar a ocupação em tempo real, fornecendo informações em aplicativos móveis ou painéis informativos.
- **Pagamento Automático:** Sistemas integrados que utilizam dados de sensores para calcular automaticamente o tempo de estacionamento e efetuar o pagamento sem a necessidade de intervenção humana.

✓ **Gestão da Cidade:**

- Semáforos Inteligentes: Controle adaptativo de semáforos com base no fluxo de tráfego em tempo real, otimizando o tempo de espera e reduzindo congestionamentos.
- Colecta de Dados Urbanos: Sensores distribuídos pela cidade para colectar dados sobre tráfego, qualidade do ar e níveis de ruído, possibilitando tomada de decisões baseadas em informações precisas.
- Iluminação Pública Eficiente: Sistemas de iluminação pública que ajustam a intensidade da luz com base na presença de pessoas, economizando energia.

✓ **Gestão da Segurança:**

- Câmeras Inteligentes: Câmeras equipadas com análise de vídeo em tempo real para detecção de atividades suspeitas, auxiliando na vigilância e prevenção de crimes.
- Sensores de Perímetro: Sensores ao redor de áreas críticas que alertam automaticamente as autoridades em caso de intrusão ou movimentos não autorizados.
- Monitoramento de Tráfego em Tempo Real: Análise em tempo real de padrões de tráfego, possibilitando a identificação rápida de incidentes e acções para melhorar a segurança viária.

✓ **Gestão da Saúde:**

- Monitoria Remoto de Pacientes: Dispositivos de IoT que colectam dados vitais de pacientes remotamente, permitindo o acompanhamento contínuo por profissionais de saúde.
- Dispositivos Vestíveis: Sensores integrados em dispositivos vestíveis para monitorar actividades físicas, padrões de sono e sinais vitais, contribuindo para a promoção da saúde.
- Alertas de Saúde Pública: Utilização de sensores para monitorar a propagação de doenças contagiosas e enviar alertas automáticos para autoridades de saúde e a população.

2.4. Análise das diferentes Soluções de Localização de Vagas de Estacionamento baseado em IoT

Este ponto aborda a análise das diferentes soluções de localização de vagas de estacionamento baseadas em IoT, destacando três abordagens distintas: Câmeras Inteligentes, Sensores de Radar a Laser e Sensores de Veículos Terrestres. Cada uma dessas soluções possui uma arquitetura única, impactando a eficácia e eficiência na gestão de estacionamentos urbanos.

2.4.1. Soluções de Localização de Vagas de Estacionamento baseado em IoT usando Câmeras Inteligentes

As câmeras inteligentes, fundamentais na infraestrutura dos sistemas de estacionamento baseados em IoT, têm evidenciado seu impacto significativo em várias cidades globais. Um estudo de caso conduzido por Smith e colaboradores (2021) em Nova Iorque destaca a eficácia dessas câmeras na otimização da gestão do estacionamento urbano. A arquitetura intrincada desses sistemas envolve a instalação de câmeras de alta resolução em locais estratégicos, como cruzamentos movimentados e áreas centrais, garantindo uma cobertura abrangente. O processamento em nuvem dessas imagens permite uma detecção precisa e em tempo real da ocupação de vagas, resultando em uma redução significativa do tempo gasto pelos motoristas procurando estacionamento.

A cidade de Nova Iorque, pioneira na implementação de câmeras inteligentes em sua infraestrutura de estacionamento, testemunhou não apenas uma melhoria na eficiência operacional, mas também uma diminuição nos congestionamentos e nas emissões de poluentes relacionados à busca por vagas. A figura 3 apresenta um exemplo visual dessas câmeras em operação, destacando como a arquitetura robusta contribui para a gestão dinâmica dos espaços de estacionamento.



Figura 3: Câmera Inteligente em Nova Iorque
Fonte: Filipa Cardoso (2016)

Além disso, a arquitectura dessas câmeras inteligentes permite a implementação de recursos avançados, como reconhecimento de placas e análise de fluxo de veículos. Esses recursos adicionais não apenas melhoram a precisão na detecção de vagas, mas também contribuem para iniciativas de segurança urbana e monitoramento do tráfego. A cidade de Nova Iorque, ao adoptar essa abordagem inovadora, exemplifica como a arquitectura robusta de câmeras inteligentes pode revolucionar a gestão de estacionamentos em ambientes urbanos complexos.

2.4.2. Soluções de Localização de Vagas de Estacionamento baseado em IoT usando Sensores de Radar a Laser

Os sensores de radar a laser, destacados por sua eficácia na detecção de veículos, têm sido implementados com sucesso em cidades como Tóquio, proporcionando uma infraestrutura sólida para a gestão inteligente de estacionamentos. Yamamoto e equipe (2020) conduziram um estudo abrangente sobre a arquitectura desses sensores na capital japonesa, evidenciando a importância do design preciso para maximizar a eficácia.

A infraestrutura de Tóquio demonstra que a instalação estratégica desses sensores em pilares específicos, com alturas otimizadas, é crucial para garantir uma visão abrangente do estacionamento. A figura 4 exemplifica essa infraestrutura, ilustrando como a arquitectura bem planeada contribui para a detecção eficiente de veículos estacionados. A implementação desses sensores resultou em benefícios tangíveis, como a redução de congestionamentos, uma vez que os motoristas são guiados de maneira eficaz para vagas disponíveis.

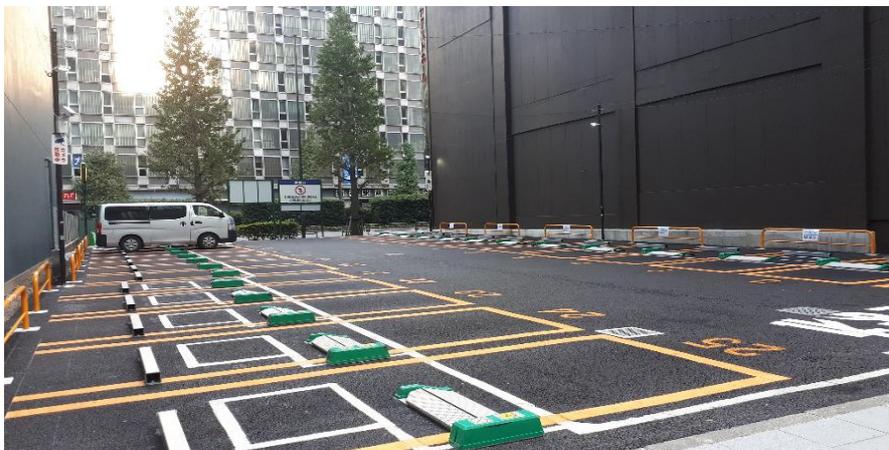


Figura 4: Sensor de Radar a Laser em Tóquio
Fonte: Vitor Matsubara(2021)

Além disso, a arquitetura dos sensores de radar a laser é integrada a sistemas de processamento massivo, que transformam dados brutos em insights úteis para a gestão dinâmica de vagas. Esses sistemas têm o potencial não apenas de otimizar o uso de espaços de estacionamento, mas também de contribuir para estratégias mais amplas de mobilidade urbana sustentável.

2.4.3. Soluções de Localização de Vagas de Estacionamento baseado em IoT usando Sensores Veículos Terrestres

Os sensores IoT de veículos terrestres, notáveis por sua simplicidade de instalação, têm sido adotados em cidades europeias, como Barcelona, transformando a experiência de estacionamento. Um relatório de implementação citado por García et al. (2019) destaca a arquitetura acessível desses sensores, que são fixados directamente no solo de cada espaço de estacionamento.

A infraestrutura de Barcelona exemplifica como a facilidade de instalação desses sensores contribui para sua adaptabilidade em diferentes ambientes urbanos. A figura 5 apresenta um exemplo visual desses sensores em operação, destacando como a simplicidade arquitetônica se traduz em eficiência na detecção de veículos estacionados. Esses sensores têm impactado positivamente a mobilidade urbana, oferecendo uma detecção precisa de vagas e contribuindo para uma gestão eficiente do espaço urbano.



Figura 5: Sensor IoT de Veículos Terrestres em Barcelona
Fonte: Panagiotis Tsarchopoulos (2014)

Além disso, a arquitectura desses sensores IoT permite uma fácil integração com sistemas centrais de comunicação, garantindo uma transferência eficiente de dados para a gestão centralizada do estacionamento. A implementação bem-sucedida desses sensores em Barcelona ilustra como a simplicidade arquitectónica pode ser uma vantagem significativa na busca por soluções práticas e eficazes para a gestão de estacionamentos urbanos.

2.4.4. Critérios para Comparação Localização de Vagas de Estacionamento baseado em IoT

A avaliação e comparação de sistemas de localização de vagas de estacionamento baseados em IoT requerem a consideração cuidadosa de vários critérios essenciais. Estes critérios são fundamentais para determinar a eficácia, eficiência e aplicabilidade prática desses sistemas inovadores.

a) Precisão na Detecção

Autores como Silva et al. (2018) ressaltam que a precisão na detecção de ocupação de vagas é crucial para garantir a confiabilidade dos sistemas de estacionamento baseados em IoT, influenciando directamente a eficácia na gestão dos espaços urbanos.

b) Integração com Tecnologias Existentes

A pesquisa de Chen et al. (2019) destaca que a integração eficaz com aplicativos móveis e plataformas urbanas é um factor determinante para a adoção bem-sucedida de sistemas de estacionamento baseados em IoT, proporcionando uma experiência de usuário mais completa.

c) Escalabilidade

Autores como Wang et al. (2021) enfatizam a importância da escalabilidade em sistemas IoT, salientando que a capacidade de expansão é vital para atender ao crescimento urbano e às demandas dinâmicas por estacionamento.

d) Consumo de Energia

Estudos de Oliveira e Santos (2018) indicam que a eficiência energética é uma consideração essencial em sistemas IoT, destacando que tecnologias que otimizam o consumo de energia são mais sustentáveis e economicamente viáveis a longo prazo.

e) Custo de Implementação e Manutenção

Autores como Santos et al. (2022) argumentam que uma abordagem econômica e sustentável é essencial na escolha de sistemas de localização de vagas de estacionamento baseados em IoT, considerando não apenas os custos iniciais, mas também os operacionais e de manutenção a longo prazo.

A Tabela 1 apresenta uma comparação entre os sistemas de localização de vagas de estacionamento baseados em IoT, utilizando diversos critérios para avaliação. Cada critério é atribuído a um valor (Alto, Médio, Baixo) para cada sistema, e a pontuação percentual reflete a performance relativa de cada sistema em relação aos critérios estabelecidos.

2.4.5. Escolha da tecnologia

A escolha da tecnologia para a implementação de sistemas de localização de vagas de estacionamento baseados em IoT é uma decisão crucial, influenciada por vários fatores, conforme evidenciado na Tabela 1 Critérios para Comparação Sistemas de Localização de Vagas de Estacionamento baseado em IoT. Cada sistema, seja Câmeras Inteligentes, Sensores de Veículos Terrestres ou Sensores de Radar a Laser, apresenta características distintas que devem ser consideradas ao tomar essa decisão estratégica.

Tabela 1: Critérios para Comparação Sistemas de Localização de Vagas de Estacionamento baseado em IoT

Critérios	Câmeras Inteligentes	Sensores Veículos Terrestres	Sensores de Radar a Laser
Precisão na Detecção	Alto	Alto	Alto
Integração com Tecnologias Existentes	Médio	Alto	Médio
Escalabilidade	Alto	Alto	Alto
Consumo de Energia	Alto	Baixo	Médio
Custo de Implementação e Manutenção	Alto	Baixo	Médio
% de Pontuação	70%	100%	80%

Fonte: Autor(2023)

A análise comparativa entre os sistemas de localização de vagas de estacionamento baseados em IoT revela insights valiosos sobre suas performances em diversos

critérios. No quesito precisão na detecção, todos os sistemas - Câmeras Inteligentes, Sensores de Veículos Terrestres e Sensores de Radar a Laser - apresentam uma alta capacidade de identificação confiável da ocupação de vagas.

Quanto à integração com tecnologias existentes, os Sensores de Veículos Terrestres obtêm a pontuação mais alta, indicando uma capacidade eficiente de se integrar a aplicativos móveis, sistemas de navegação e outras plataformas urbanas. As Câmeras Inteligentes e os Sensores de Radar a Laser demonstram desempenho médio nesse critério.

A escalabilidade, fundamental para atender ao crescimento urbano, é uma característica distintiva positiva para todos os sistemas analisados, sugerindo que podem ser facilmente expandidos para lidar com demandas crescentes.

No que diz respeito ao consumo de energia, os Sensores de Veículos Terrestres apresentam a pontuação mais baixa, indicando eficiência energética. Por outro lado, as Câmeras Inteligentes revelam um consumo alto, enquanto os Sensores de Radar a Laser possuem um consumo intermediário.

No critério de custo de implementação e manutenção, os Sensores de Veículos Terrestres lideram, indicando serem uma opção mais econômica ao longo do tempo. As Câmeras Inteligentes e os Sensores de Radar a Laser têm custos mais elevados, embora permaneçam dentro de uma faixa intermediária.

Em síntese, os Sensores de Veículos Terrestres emergem como a opção mais equilibrada, apresentando alta pontuação em vários critérios. As Câmeras Inteligentes, apesar de uma pontuação significativa, podem ser mais dispendiosas e demandar maior consumo de energia. Os Sensores de Radar a Laser ocupam uma posição intermediária, oferecendo desempenho equilibrado em diversos aspectos. A escolha entre esses sistemas deve ser guiada pelas necessidades específicas de cada ambiente urbano.

2.5. Sistemas de Localização de Vagas de Estacionamento baseado em IoT usando Sensores Veículos Terrestres

Os sistemas de localização de vagas de estacionamento baseados em IoT, que utilizam sensores de veículos terrestres, apresentam uma arquitetura eficiente e simplificada. Esses sistemas são compostos por elementos-chave que

desempenham funções específicas para proporcionar uma experiência inteligente de estacionamento.

✓ **Sensor de Detecção de Presença**

O componente central desses sistemas é o sensor de detecção de presença, projectado para identificar a ocupação de vagas por veículos. Esses sensores são estrategicamente instalados no solo de cada espaço de estacionamento, oferecendo uma detecção precisa e confiável. A tecnologia de sensor pode ser magnética, ultrassônica ou baseada em outros princípios, proporcionando flexibilidade na escolha da tecnologia mais adequada ao ambiente.

✓ **Microcontrolador para Processamento de Dados**

Cada sensor de detecção de presença está equipado com um microcontrolador dedicado responsável pelo processamento dos dados colectados. Esses microcontroladores desempenham um papel crucial na interpretação das informações do sensor, garantindo uma resposta rápida e precisa sobre o status de ocupação da vaga. O processamento local dos dados otimiza a eficiência do sistema, reduzindo a dependência de recursos externos.

✓ **Plataforma em Nuvem para Armazenamento e Recuperação de Dados**

Os dados processados pelos microcontroladores são transferidos para uma plataforma em nuvem dedicada. Essa plataforma actua como um repositório centralizado, armazenando e gerindo as informações sobre a ocupação das vagas em tempo real. Além de oferecer um armazenamento seguro, a plataforma em nuvem facilita a acessibilidade aos dados de qualquer local, permitindo uma gestão centralizada e remota do sistema.

✓ **Aplicativo Móvel para Controle do Estacionamento Inteligente**

Para fornecer uma experiência interativa aos usuários, esses sistemas integram um aplicativo móvel intuitivo. Esse aplicativo permite que os motoristas acessem informações em tempo real sobre a disponibilidade de vagas, facilitando a busca por estacionamento. Além disso, oferece funcionalidades de reserva de vagas, navegação até o local escolhido e até mesmo a realização de pagamentos, proporcionando uma solução completa para a gestão de vagas de estacionamento.

2.6. Principais vantagens de um Sistema de Gestão de Vagas de Estacionamento baseado em IoT

- ✓ Um sistema de gestão de vagas de estacionamento baseado em IoT pode trazer muitas vantagens, incluindo melhorias na eficiência operacional, redução de congestionamentos e emissões de gases de efeito estufa, melhoria na experiência do usuário, redução de fraudes e aumento da segurança. (Kumar et al., 2020);
- ✓ A implementação de um sistema de gestão de vagas de estacionamento com IoT pode ajudar a melhorar a eficiência do uso de espaços de estacionamento, aumentar a rotatividade de veículos, reduzir o tempo gasto na busca de vagas e, conseqüentemente, reduzir o tráfego nas ruas adjacentes. (Yin et al., 2020);
- ✓ A implementação de um sistema de gestão de vagas de estacionamento com IoT pode reduzir o número de infrações de trânsito e melhorar a segurança, evitando o congestionamento de veículos na entrada do estacionamento. (Zhang et al., 2018).

CAPÍTULO III – CASO DE ESTUDO

3.1. Parque de Estacionamento do Conselho Municipal

Nesta secção, aprofundaremos a localização, organização e processo actual de gestão de vagas no Parque de Estacionamento do Conselho Municipal. Serão destacados os elementos fundamentais da estrutura organizacional, bem como o fluxo operacional desde a entrada até a saída dos veículos. Além disso, identificaremos os principais constrangimentos e limitações do processo actual que impactam a eficiência e a experiência dos usuários.

3.1.1. Localização

O sistema de gestão de vagas de estacionamento baseado em IoT será concebido para a instalação no parque de estacionamento do conselho municipal da cidade de Maputo, sita na esquina entre as avenidas 25 de setembro e Filipe Samuel Magaia na extensão de recinto do mercado central de Maputo.

O parque de estacionamento escolhido para o estudo, encontra-se no momento em reabilitação (construção do silo de estacionamento com três (3) pisos) e ira albergar quatrocentas (400) viaturas).

3.1.2. Organização

A organização do parque de estacionamento do conselho municipal é composta pelo conselho municipal que dita as regras que deverão ser seguidas e todos os acontecimentos são ser reportados a ele.

O gerente que explica e cobra aos seus subordinados as regras impostas pelo conselho municipal. Cabe também ao gerente administrar folhas de ponto, banco de horas, escalas de serviço e finanças.

O supervisor é responsável por todos os acontecimentos dentro do estacionamento no seu horário de serviço, tanto que deverá reportar ao gerente todas as ocorrências e alterações de serviço. Deverá administrar as guaritas determinando as ocupações para evitar falhas na prestação do serviço.

O funcionário (atendente) é responsável pelo serviço de controle de entrada e saída dos veículos.

O segurança é responsável por garantir a segurança dos veículos que estão dentro do estacionamento.

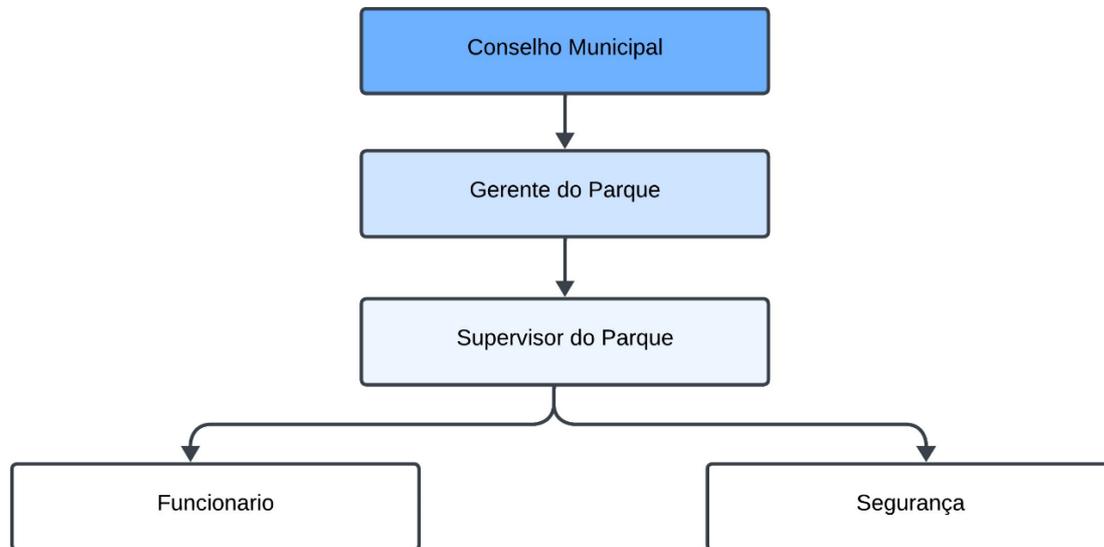


Figura 6: Organograma do parque
Fonte: Autor (2023)

O parque de estacionamento do conselho municipal cobra uma taxa fixa de 10,00MT (dez meticais) por hora. Os funcionários do parque são isentos dessa cobrança e sua identificação é feita através da apresentação da sua carteira de funcionário.

Os clientes que desejam estacionar deverão se dirigir para qualquer guarita de entrada e adquirir o seu ticket. O período laboral é o que possui maior movimentação de veículos, fazendo assim com que um maior número de funcionários trabalhe neste período.

No momento de entrada no estacionamento, o cliente recebe um ticket que contem o número de identificação do ticket que identifica a quantidade de veículos que fizeram uso do estacionamento até o momento. O cliente após receber seu ticket, efetua busca por uma vaga ainda não ocupada.

Na saída do veículo, o cliente se dirige para uma guarita destinada para saída, efetua o pagamento do ticket e o funcionário do parque aciona a cancela liberando o veículo.

Em caso de perda do ticket, o cliente só poderá retirar o seu veículo mediante apresentação do documento do veículo e um documento de identificação pessoal. O funcionário da guarita, por sua vez, deverá conferir os dados do veículo contidos no documento, caso os documentos estejam de acordo o cliente é liberado.

No final de cada turno é efetuado registo da contabilidade dos lucros. Os caixas das guaritas não poderão finalizar o dia em débito, caso aconteça, o funcionário responsável pelo caixa deverá pagar o débito em questão.

3.1.3. Descrição do processo actual (fluxo de entrada)

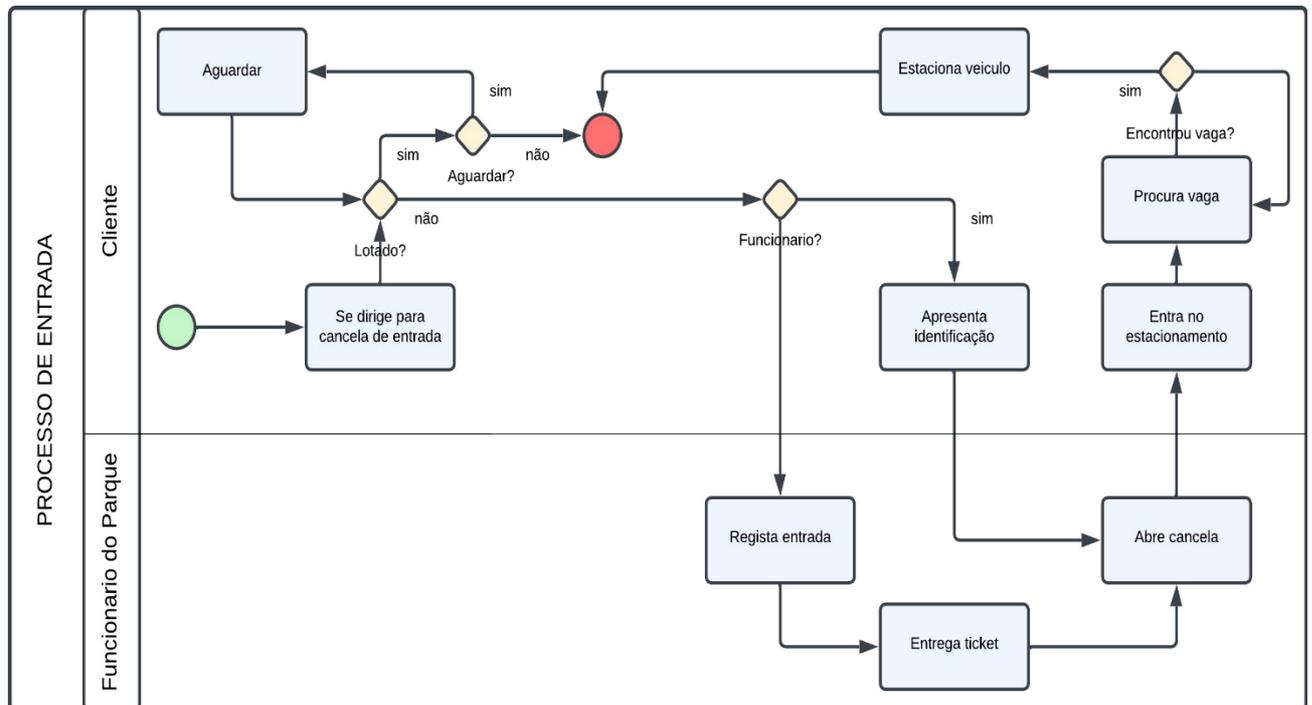


Figura 7: Fluxo de entrada de viaturas no Parque
Fonte: Autor (2023)

❖ Descrição Detalhada do Fluxograma (Processo de entrada)

F1 - (Cliente) Se dirige para a cancela de entrada

O cliente chega ao estacionamento e se dirige para uma das guaritas de entrada que estiver em funcionamento.

F2 - (Cliente) Estacionamento lotado?

Após se dirigir para uma das guaritas de atendimento o usuário poderá se deparar com o estacionamento lotado.

F3 - (Cliente) Aguardar?

Quando o estacionamento estiver lotado o usuário poderá ou não aguardar a desocupação de alguma vaga.

F4 - (Cliente) Aguardar

Se o usuário optar por aguardar ele deverá ser avisado sobre a disponibilidade de uma nova vaga para enfim poder estacionar seu veículo.

F5 - (Cliente) Funcionário?

Há possibilidade de ser cliente ou funcionário do parque. A diferença entre esses dois é que o cliente comum deverá efetuar pagamento para usar o estacionamento, já o funcionário do parque poderá fazer uso mediante apresentação de identificação.

F6 - (Cliente) Apresenta identificação

Caso seja funcionário do parque, deverá apresentar seu cartão de identificação.

F7 - (Funcionário do parque) Regista entrada

Caso seja cliente o funcionário (atendente) deverá registrar a entrada, anotando a placa, hora e data da entrada. Logo, emite o ticket de identificação do cliente.

F8 - (Funcionário do parque) Entrega ticket

Após registo da entrada do cliente e da emissão do ticket, o funcionário entrega o ticket para o usuário.

F9 - (Funcionário Auto Park) Abre cancela

Após a entrega do ticket para o cliente, o funcionário Auto Park deverá abrir a cancela liberando o uso do estacionamento.

F10 - (Cliente) Entra no estacionamento

Após a liberação feita pelo funcionário, o cliente deverá entrar no estacionamento com seu veículo.

F11 - (Cliente) Procura vaga

O cliente deverá estacionar seu veículo em uma vaga não ocupada, para isso deverá antes fazer uma busca dentro do estacionamento.

F12 - (Cliente) Encontrou?

Muitos clientes têm por preferência estacionar o seu veículo em lugares prediletos, eles se dirigem primeiramente para essas vagas. Mas, há possibilidade dessas vagas já se encontrarem em uso. Nesse caso ele deverá procurar outra vaga.

F13 - (Cliente) Estaciona veículo

Se o cliente encontrar a vaga predileta ou até mesmo uma outra vaga, deverá então estacionar seu veículo nesta vaga

3.1.4. Descrição do processo actual (fluxo de saída)

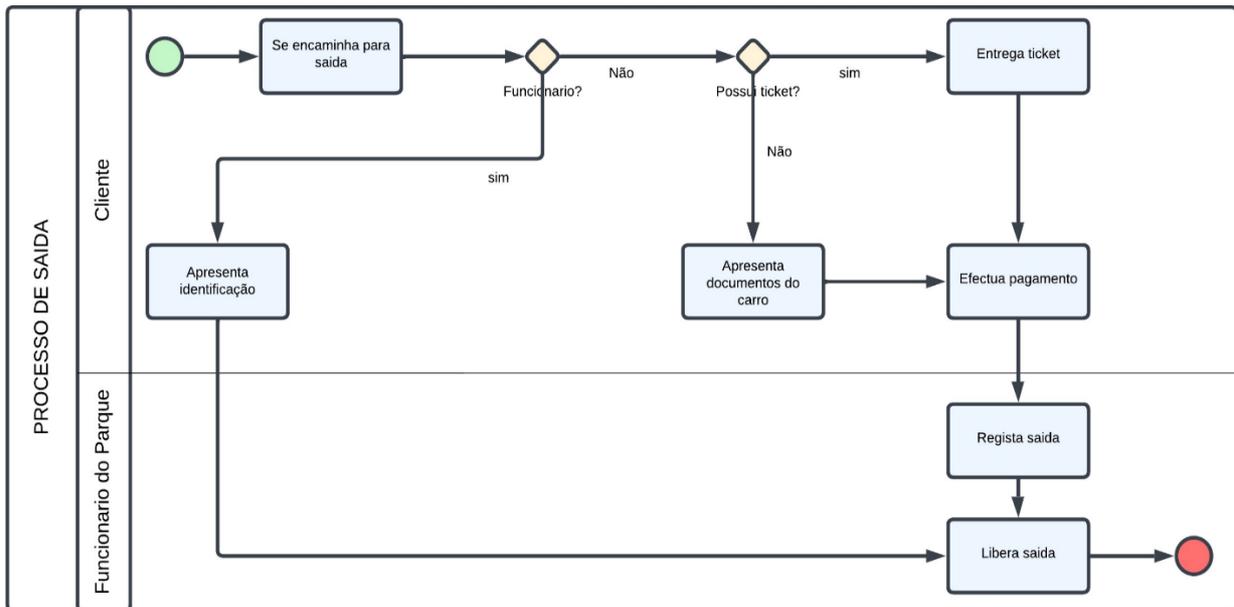


Figura 8: Fluxo de saída de viaturas no Parque
Fonte: Autor (2023)

❖ Descrição Detalhada do Fluxograma (processo de saída do cliente)

F1 - (Cliente) Se encaminha para saída

O cliente após encontrar seu veículo, se encaminha para uma guarita destinada à saída de veículos.

F2 - (Cliente) Possui ticket?

O funcionário deverá verificar se o cliente está portando o ticket de identificação.

F3 - (Cliente) Entrega ticket

Se o cliente possuir o ticket, deverá então entregá-lo para o funcionário para que seja feito o registo da saída de seu carro.

F4 - (Cliente) Apresenta documentos do carro

O cliente não possuindo o ticket, deverá então apresentar os documentos pessoais e os documentos do veículo

F5 - (Cliente) Efectua Pagamento

O cliente efectua o pagamento da taxa para o funcionário do parque.

F6 - (Funcionário do parque) Regista saída

O funcionário do parque então após os procedimentos de identificação e pagamento regista a saída do veículo.

F7 - (Funcionário do parque) Libera saída

Após registar a saída do cliente, o funcionário deverá então liberar sua saída.

❖ **Descrição Detalhada do Fluxograma (processo de saída do funcionário do parque)**

F1 - (Cliente) Se encaminha para saída

O funcionário do parque após encontrar seu veículo, se encaminha para uma guarita destinada à saída de veículos.

F2 - (Cliente) Funcionário?

Há possibilidade de ser cliente ou funcionário do parque. A diferença entre esses dois é que o cliente deverá efetuar pagamento para utilizar o estacionamento, já o funcionário do parque poderá fazer uso gratuito mediante apresentação de identificação.

F3 - (Cliente) Apresenta identificação

O funcionário do parque estará isento do pagamento da taxa após a apresentação de sua identificação.

F4 - (Funcionário do parque) Libera saída

Após a apresentação de identificação, o funcionário (atendente) deverá então liberar a saída do funcionário do parque

3.2. Constrangimentos e limitações existentes do processo actual gestão de vagas do Parque de Estacionamento do Conselho Municipal

O actual processo de gestão de vagas no Parque de Estacionamento do Conselho Municipal apresenta diversos constrangimentos e limitações que contribuem para a ineficiência e a falta de informações em tempo real sobre a disponibilidade de vagas. Esses desafios impactam directamente a experiência dos usuários e a eficácia da operação do estacionamento. A seguir, são apresentados alguns dos principais constrangimentos e limitações identificados:

a) Falta de Informações em Tempo Real sobre Disponibilidade

Descrição: A ausência de um sistema eficiente dificulta a obtenção de informações em tempo real sobre quais vagas estão ocupadas ou livres.

Impacto: Os usuários não conseguem planejar suas chegadas com antecedência, resultando em busca desordenada por vagas e aumento do tempo gasto procurando estacionamento.

b) Processo Manual de Alocação de Vagas

Descrição: A alocação de vagas é realizada manualmente, sem um sistema automatizado.

Impacto: Isso contribui para a falta de eficiência na utilização do espaço disponível, com potencial subutilização de vagas e congestões em áreas específicas.

c) Incapacidade de Monitorar a Duração da Ocupação de Vagas

Descrição: O sistema atual não permite rastrear por quanto tempo um veículo ocupou uma vaga específica.

Impacto: A falta de monitoria preciso dificulta a identificação de padrões de uso e a otimização do tempo de permanência nas vagas, levando a possíveis abusos ou subutilização.

d) Complexidade na Verificação da Disponibilidade de Vagas

Descrição: A falta de um sistema centralizado dificulta a verificação rápida da disponibilidade de vagas em diferentes áreas do estacionamento.

Impacto: Os usuários enfrentam desafios ao tentar determinar as áreas com vagas livres, resultando em busca desordenada e congestionamento interno.

e) Dificuldade em Planejar Chegadas devido à Incerteza da Disponibilidade

Descrição: A incerteza sobre a disponibilidade de vagas impede que os usuários planejem suas chegadas de forma eficaz.

Impacto: A busca desordenada por vagas aumenta o tráfego interno, gerando congestionamentos e frustrações entre os usuários.

f) Limitação na Comunicação de Vagas Livres

Descrição: A falta de um sistema de comunicação eficiente impede a notificação rápida dos usuários sobre vagas que foram desocupadas.

Impacto: Os usuários não são informados prontamente sobre vagas recém-libertadas, resultando em atrasos na ocupação eficiente e possível aumento do tempo de busca.

A abordagem desses constrangimentos é crucial para o desenvolvimento de um sistema baseado em IoT que resolva efetivamente os desafios relacionados à localização, alocação e disponibilidade de vagas no Parque de Estacionamento do Conselho Municipal.

3.3. Proposta de solução

Diante dos constrangimentos de melhorar a gestão de vagas no Parque de Estacionamento do Conselho Municipal em Maputo, propõe-se a implementação de um Sistema de Gestão de Vagas de Estacionamento baseado em Internet das Coisas (IoT). A intenção é aprimorar a eficiência operacional, a segurança e a experiência do usuário no estacionamento.

3.3.1. Descrição da proposta de solução

❖ Funcionamento do Sistema:

Monitoria em Tempo Real: O sistema irá acompanhar em tempo real a ocupação das vagas, utilizando sensores de entrada e saída para obter informações precisas.

Alocação Automática: A alocação de vagas será automatizada para otimizar o uso do espaço disponível, evitando congestionamentos e facilitando a experiência dos usuários.

Controle de Portões: A abertura e fechamento dos portões de entrada e saída serão automatizados, proporcionando conveniência aos usuários e aumentando a eficiência operacional.

Comunicação e Notificação: Utilizando uma plataforma de comunicação eficiente (MQTT), o sistema informará a quantidade de carros estacionados e o status de cada vaga, garantindo uma comunicação rápida e confiável.

Registo do Tempo de Estacionamento: Será mantido um registo do tempo de ocupação de cada vaga, fornecendo informações valiosas sobre os padrões de uso e contribuindo para a gestão eficaz do espaço.

Essa proposta visa não apenas solucionar os constrangimentos identificados, como a falta de informações em tempo real e a alocação manual de vagas, mas também proporcionar uma gestão moderna e eficiente do estacionamento, melhorando a experiência dos usuários e otimizando o uso do espaço disponível.

3.3.1.1. Diagrama de sequência (fluxo de entrada)

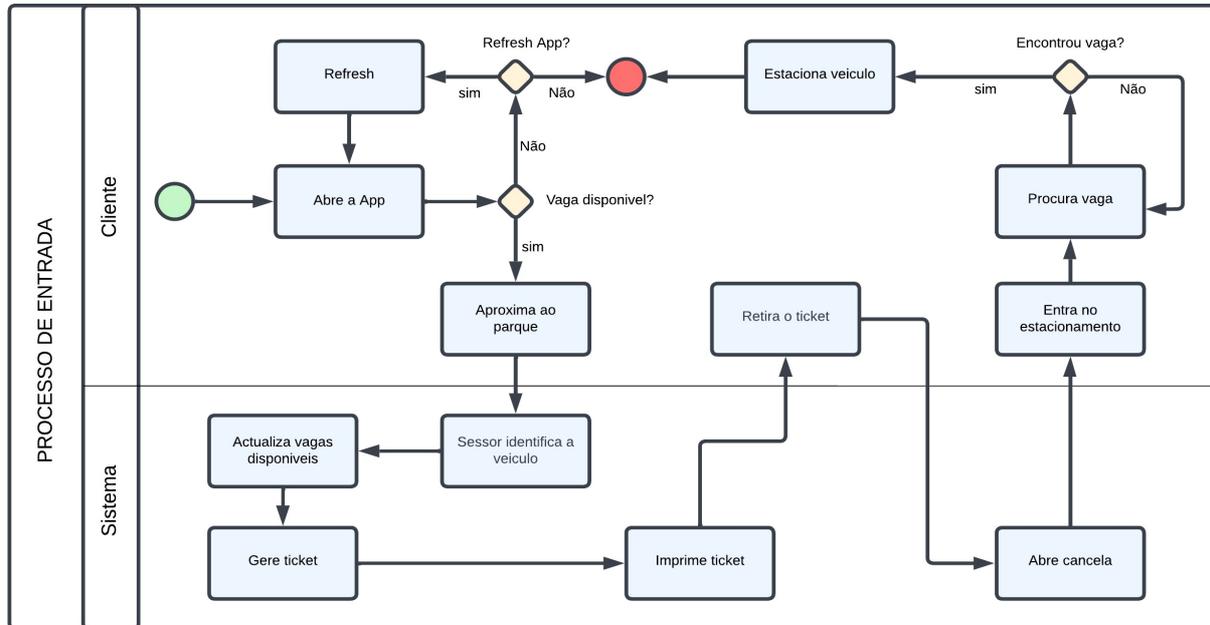


Figura 9: Fluxo de entrada de viacturas no Parque (sistema proposto)
Fonte: Autor (2023)

3.3.1.2. Diagrama de sequência (fluxo de saída)

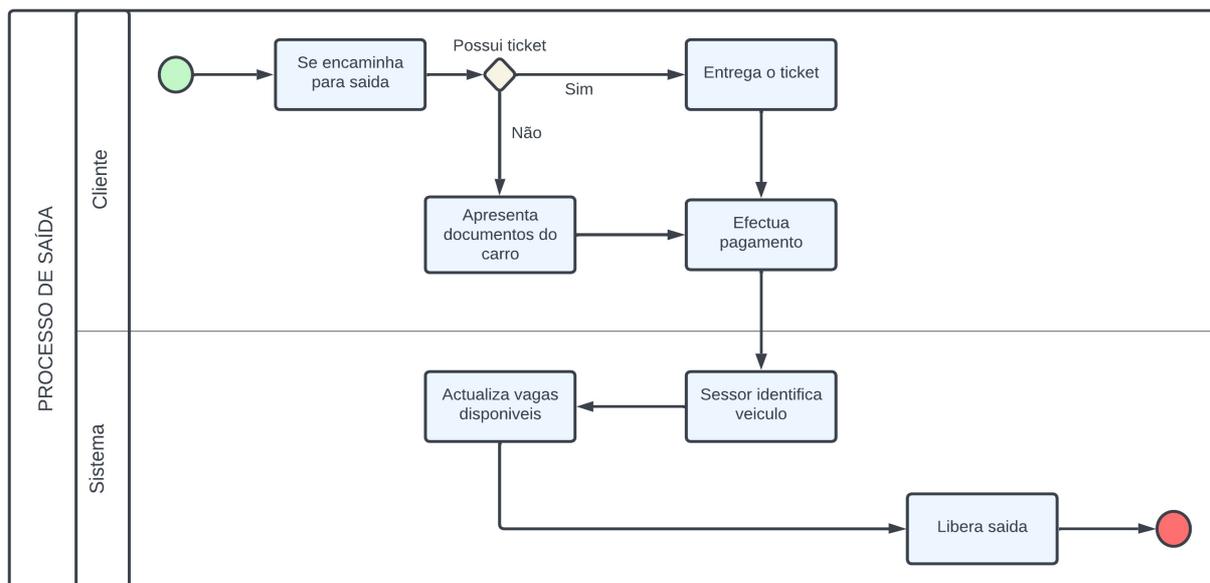


Figura 10: Fluxo de saída de viacturas no Parque (sistema proposto)
Fonte: Autor (2023)

3.3.2. Composição do Sistema

A solução será composta pelos seguintes componentes:

Sensores de Infravermelho (IR): Cinco sensores IR serão implantados em locais estratégicos dentro do parque de estacionamento. Dois desses sensores serão usados nas entradas e saídas para detectar a presença de veículos durante a entrada e saída.



Figura 11: Sensores de Infravermelho (IR)
Fonte: Robocraze(2019)



Figura 12: Servo Motores
Fonte: Robocraze(2019)

Servo Motores: Dois servos motores serão instalados nas portas de entrada e saída do parque de estacionamento. Esses motores serão controlados para abrir e fechar automaticamente as portas quando um veículo se aproximar.

NodeMCU: Um NodeMCU (um dispositivo IoT baseado em ESP8266) será o cérebro do sistema. Ele se comunicará com os sensores IR para detectar a disponibilidade de vagas e controlar os servo motores para a abertura das portas.



Figura 13: ESP8266
Fonte: Robocraze(2019)



Figura 14: Plataforma Adafruit IO
Fonte: Adafruit IO (2023)

Plataforma Adafruit IO: Os dados coletados pelos sensores serão enviados para a plataforma Adafruit IO por meio do NodeMCU. Isso permitirá o monitoramento remoto em tempo real e o acesso aos dados da disponibilidade das vagas.

3.3.3. Princípio de Funcionamento

Detecção de Veículos: Os sensores IR na entrada e saídas detectam a presença de veículos à medida que entram e saem do parque de estacionamento.

Controle de Portas: Os servo motores controlam a abertura e o fechamento das portas automaticamente em resposta aos veículos detectados.

Detecção de Vagas Disponíveis: Três sensores IR adicionais são usados para verificar a disponibilidade de vagas de estacionamento em tempo real. Esses sensores são colocados em locais estratégicos para cobrir várias áreas do parque.

Comunicação com a Plataforma Adafruit IO: Os dados coletados pelos sensores, incluindo a disponibilidade de vagas e os tempos de entrada e saída dos veículos, são enviados para a plataforma Adafruit IO por meio do NodeMCU.

Monitoria Remoto: Os administradores do parque de estacionamento e os motoristas podem acessar as informações de disponibilidade de vagas em tempo real por meio da plataforma Adafruit IO, que pode ser acessada de qualquer lugar com conexão à Internet.

3.3.4. Custo de Implementação e Instalação do Sistema de estacionamento IoT

A tabela 2: Custo de Implementação e Instalação do Sistema de estacionamento IoT apresenta uma estimativa detalhada das despesas associadas à implementação e instalação do Sistema de Estacionamento baseado em IoT. Essas despesas incluem a aquisição de componentes, mão-de-obra de instalação, configuração de rede e manutenção ao longo de um período de 12 meses.

Tabela 2: Custo de Implementação e Instalação do Sistema de estacionamento IoT

Descrição da Despesa	Qtd.	P. Unitário (MT)	P. Total (MT)
Microcontrolador NodeMCU ESP8266	1	15,000	15,000
RI Sensor (350unid)	350	214	75,000
Servo Motor (2)	2	5,000	10000
Placa de Circuito Impresso	1	5,000	5,000
Material eléctrico (cabos, resistências, LED, etc.)	1	7,500	7,500
Mão-de-obra de instalação	n/a	10,000	10,000
Configuração de Rede Adafruit IO	n/a	5,000	5,000
Manutenção e suporte técnico (12 meses)	n/a		20,000
		Total	147,500

Fonte: Autor (2023)

3.3.5. Benefícios da Solução

Disponibilidade em Tempo Real: Os motoristas podem verificar a disponibilidade de vagas em tempo real, economizando tempo e combustível na busca por vagas.

Conveniência: A abertura automática das portas e a detecção de vagas tornam o processo de estacionamento mais conveniente para os motoristas.

Eficiência Operacional: O sistema ajuda a otimizar a utilização do espaço disponível, reduzindo congestionamentos e melhorando o fluxo de veículos.

Segurança Aprimorada: A detecção de veículos e a abertura automática das portas melhoram a segurança e reduzem o risco de acidentes.

Monitoria Remoto: Os administradores podem monitorar e gerir o parque de estacionamento de qualquer lugar, tornando a gestão mais eficaz.

Essa solução proporcionará uma experiência de estacionamento mais eficiente e segura para os motoristas, ao mesmo tempo em que melhora a gestão e a operação do Parque de Estacionamento do Conselho Municipal de Maputo. Ela representa um passo importante em direção à modernização e automação dos serviços de estacionamento na cidade.

CAPÍTULO IV – DESENVOLVIMENTO DO PROTÓTIPO

4.1. Visão Geral do Sistema

Nesta secção, forneceremos uma visão geral do Sistema de Gestão de Vagas de Estacionamento baseado em IoT que está sendo desenvolvido para o Parque Municipal de Estacionamento do Mercado Central na Cidade de Maputo. Esta visão geral destacará os principais componentes e funcionalidades do sistema.

Componentes do Sistema:

O sistema é composto pelos seguintes componentes:

- **Sensores de Infravermelho (IR):** Utilizados para detectar a presença de veículos nas entradas, saídas e espaços de estacionamento.
- **Servo Motores:** Controlam a abertura e o fechamento das portas de entrada e saída.
- **NodeMCU:** O dispositivo IoT que controla o sistema, coleta dados dos sensores e os envia para a plataforma Adafruit IO.
- **Plataforma Adafruit IO:** Recebe e armazena dados em tempo real sobre a disponibilidade de vagas e os tempos de entrada e saída dos veículos.

4.2. Análise e definição de requisitos

A análise e definição de requisitos são etapas críticas no desenvolvimento do sistema. Elas garantirão que o sistema atenda às necessidades específicas do Parque Municipal de Estacionamento do Mercado Central – Cidade de Maputo. Essa análise inclui a definição do âmbito do sistema e a identificação dos grupos de interesse.

4.2.1. Âmbito do Sistema

O âmbito do Sistema de Gestão de Vagas de Estacionamento baseado em IoT engloba todas as funcionalidades e componentes que estão sendo desenvolvidos para atender às necessidades do Parque Municipal de Estacionamento do Mercado Central em Maputo. O sistema visa otimizar o uso de espaço de estacionamento,

melhorar a eficiência operacional e proporcionar uma experiência de estacionamento conveniente para os motoristas.

4.2.2. Grupos de Interesse

Os principais grupos de interesse envolvidos no Sistema de Gestão de Vagas de Estacionamento incluem:

Administradores do Parque de Estacionamento: Este grupo é responsável pela gestão e operação do parque de estacionamento. Eles desejam melhorar a eficiência operacional, otimizar o uso de espaço e fornecer um serviço de estacionamento de alta qualidade aos motoristas.

Motoristas: Os motoristas que utilizam o parque de estacionamento são um grupo importante. Eles desejam encontrar vagas de estacionamento facilmente, evitar atrasos na entrada e saída e receber informações em tempo real sobre a disponibilidade de vagas.

Operadores de Serviços de Estacionamento: Em alguns casos, empresas de serviços de estacionamento podem estar envolvidas na gestão do parque de estacionamento. Eles estão interessados em maximizar a eficiência da operação e garantir que os motoristas tenham uma experiência positiva.

4.2.3. Convenções

4.2.3.1. Identificação dos Requisitos

Os requisitos do sistema serão identificados e numerados de acordo com a seguinte convenção: RF (Requisito Funcional) seguido de um número sequencial. Por exemplo, RF01, RF02 e assim por diante.

4.2.3.2. Identificação dos Casos de Uso

Os casos de uso serão identificados e documentados usando a seguinte convenção: CU (Caso de Uso) seguido de um número sequencial. Por exemplo, CU001, CU002 e assim por diante.

4.2.4. Requisitos do Sistema

Os requisitos do sistema são elementos essenciais para o desenvolvimento bem-sucedido do Sistema de Gestão de Vagas de Estacionamento baseado em IoT. Eles

serão detalhados na seção seguinte e abordarão tanto os requisitos funcionais quanto os não funcionais do sistema.

4.2.4.1. Prioridades dos Requisitos

Os requisitos do sistema serão priorizados com base em sua importância e impacto nas funcionalidades e no desempenho geral do Sistema de Gestão de Vagas de Estacionamento. As prioridades dos requisitos serão categorizadas da seguinte forma:

- **Prioridade Alta:** Requisitos considerados críticos para a funcionalidade básica do sistema e a experiência do usuário. Eles serão implementados como prioridade máxima.
- **Prioridade Média:** Requisitos importantes, mas não críticos para a funcionalidade principal. Eles serão abordados após os requisitos de alta prioridade.
- **Prioridade Baixa:** Requisitos desejáveis, mas não essenciais para a funcionalidade básica do sistema. Eles serão considerados em fases futuras de desenvolvimento, se viáveis.

4.2.5. Requisitos funcionais

Os requisitos funcionais definem as funções específicas que o sistema deve desempenhar para atender aos objetivos e necessidades do Parque Municipal de Estacionamento do Mercado Central.

Tabela 3: Requisitos funcionais

Ref	Descrição do Requisito Funcional	Critério de Aceitação	Prioridade
RF01	Detecção de Entrada de Veículos	O sistema deve detectar a entrada de veículos quando o sensor de entrada for activado.	Alta
RF02	Detecção de Saída de Veículos	O sistema deve detectar a saída de veículos quando o sensor de saída for activado.	Alta
RF03	Controle das Portas de Entrada e Saída	O sistema deve ser capaz de controlar as portas de entrada e saída de acordo com a detecção de veículos.	Alta
RF04	Armazenamento de Dados	O sistema deve ser capaz de publicar dados relevantes, como contagem de vagas e horários de entrada/saída, no Adafruit IO.	Média

Fonte: Autor (2023)

4.2.6. Requisitos não funcionais

Os requisitos não funcionais definem as características não relacionadas a funções do sistema, mas que são igualmente importantes para o seu desempenho e eficiência.

Tabela 4: Requisitos não funcionais

Ref	Descrição do Requisito Funcional	Critério de Aceitação
RNF01	Disponibilidade do Sistema	O sistema deve estar disponível 24 horas por dia, 7 dias por semana, com um tempo de inatividade mínimo planejado para manutenção.
RNF02	Desempenho e Escalabilidade	O sistema deve ser capaz de lidar com um grande volume de veículos e usuários simultaneamente sem degradação significativa no desempenho.
RNF03	Confiabilidade	O sistema deve ser altamente confiável, minimizando falhas e erros que possam afetar a experiência do usuário.
RNF05	Integração com Adafruit IO	O sistema deve integrar-se perfeitamente com a plataforma Adafruit IO para publicação e subscrição de dados.
RNF06	Facilidade de Uso	A interface do usuário deve ser intuitiva e fácil de usar, tanto para motoristas quanto para operadores do parque de estacionamento.
RNF07	Segurança dos Dados	Todos os dados sensíveis, incluindo informações de entrada/saída de veículos, devem ser armazenados e transmitidos de forma segura, usando protocolos de criptografia apropriados.
RNF08	Tempo de Resposta	O sistema deve responder rapidamente às detecções de veículos e atualizações em tempo real, minimizando qualquer atraso perceptível.

Fonte: Autor (2023)

4.3. Projecto de sistema e software

Os requisitos de software são fundamentais para o desenvolvimento do sistema de gestão de estacionamento baseado em IoT. Eles descrevem as características e funcionalidades do software que serão necessárias para atender às necessidades do sistema. Abaixo estão os requisitos de software para o projecto:

4.3.1. Requisitos de software

a) Requisitos Funcionais de Software:

Detecção de Entrada de Veículos (RF01): O software deve ser capaz de detectar a entrada de veículos quando o sensor de entrada for ativado.

Detecção de Saída de Veículos (RF02): O software deve ser capaz de detectar a saída de veículos quando o sensor de saída for ativado.

Contagem de Vagas Disponíveis (RF03): O software deve manter uma contagem atualizada das vagas de estacionamento disponíveis.

Controle das Portas de Entrada e Saída (RF04): O software deve ser capaz de controlar as portas de entrada e saída de acordo com a detecção de veículos.

Integração com Adafruit IO (RF05): O software deve ser capaz de se integrar com a plataforma Adafruit IO para publicação e subscrição de dados, incluindo a contagem de vagas e horários de entrada/saída.

b) Requisitos Não Funcionais de Software:

Segurança dos Dados (RNF01): O software deve garantir a segurança de todos os dados sensíveis, incluindo informações de entrada/saída de veículos, por meio do uso de protocolos de criptografia.

Disponibilidade do Sistema (RNF02): O software deve estar disponível 24 horas por dia, 7 dias por semana, com tempo de inatividade mínimo planejado para manutenção.

Desempenho e Escalabilidade (RNF03): O software deve ser capaz de lidar com um grande volume de veículos e usuários simultaneamente, mantendo um tempo de resposta aceitável.

Confiabilidade (RNF04): O software deve ser altamente confiável, minimizando falhas e erros que possam afetar a experiência do usuário.

Facilidade de Uso (RNF05): A interface do usuário do software deve ser intuitiva e fácil de usar, tanto para motoristas quanto para operadores do parque de estacionamento.

4.3.2. Requisitos de hardware

Os requisitos de hardware são essenciais para garantir que o sistema funcione de maneira eficiente e confiável. Eles descrevem os componentes físicos necessários

para a implementação do sistema. Abaixo estão os requisitos de hardware para o projecto:

NodeMCU (ESP8266): O hardware deve incluir um NodeMCU ou módulo semelhante para controlar as funcionalidades do sistema e se comunicar com a rede Wi-Fi.

Sensores de Entrada e Saída: Serão necessários sensores de entrada e saída para detectar a presença de veículos nas portas de entrada e saída.

Servo Motores: O sistema deve incluir servo motores para controlar as portas de entrada e saída, permitindo que sejam abertas e fechadas automaticamente.

Plataforma Adafruit IO: O sistema deve ser compatível com a plataforma Adafruit IO para publicação e subscrição de dados.

Conexão Wi-Fi: O hardware deve ser equipado com uma conexão Wi-Fi estável para se comunicar com a rede e a plataforma Adafruit IO.

Fonte de Alimentação: Será necessário um sistema de alimentação adequado para alimentar todos os componentes eletrônicos, como NodeMCU, sensores e servo motores.

4.3.3. Arquitectura

O modelo de solução proposto está dividido em três camadas nomeadamente, camada física e percepção, camada de rede e camada de aplicação, onde a forma de interação entre elas é da base para cima, isto é, tudo começa da camada física e percepção e culmina na camada de aplicação.



Figura 15: Arquitectura do modelo de solução
Fonte: Autor (2023)

5.2.3.1. Camada Física e Percepção

A primeira camada está dividida em duas partes tendo a primeira parte física que trata da transmissão de bits normais por um canal de comunicação (Tanenbaum A. S., 2011) e a segunda a percepção que é responsável por captar as grandezas físicas do ambiente e convertê-las para um formato digital que possa ser facilmente transportado pela camada de rede (Serafim, 2014). Nesta camada é usado os sensores IR Sensor (5), Servo Motor (2) e um microcontrolador NODE MCU ESP8266.

5.2.3.2 Camada de Rede

A camada de rede tem como função a transmissão dos dados obtidos pelos dispositivos da camada de percepção para a camada de aplicação (Serafim, 2014), nesta camada a transmissão de dados é feita usando o protocolo IEEE802.11.

5.2.3.1 Camada de Aplicação

A camada de aplicação tem como função o armazenamento, processamento, análise dos dados e tomada de decisão enviados pela camada de rede (Serafim, 2014). Nesta camada para o armazenamento é usada a nuvem onde depois os dados são processados e apresentado em um gráfico.

4.3.4. Modelo de funcionamento

O sistema opera detectando a presença de veículos nas entradas e saídas do estacionamento e nas vagas de estacionamento. Com base nesses dados, ele determina a disponibilidade de vagas e controla automaticamente as cancelas de entrada e saída. Os dados são enviados para a nuvem Adafruit IO, para que os usuários possam monitorar remotamente o status do estacionamento. Além disso, os usuários têm a opção de controlar manualmente as cancelas quando necessário.

a) Detecção de Veículos na Entrada e Saída:

O sistema utiliza dois sensores infravermelhos (IR) colocados nas entradas e saídas do estacionamento. Quando um veículo se aproxima da entrada, o sensor IR na entrada detecta sua presença. Da mesma forma, na saída, o sensor IR na saída identifica a presença de um veículo.

b) Controle das Cancelas de Entrada e Saída:

Quando um veículo é detectado na entrada, o sistema ativa um servo motor associado à cancela da entrada. O servo motor gira a cancela da entrada para abrir, permitindo que o veículo entre no estacionamento.

Após um atraso apropriado, a cancela da entrada é fechada, e o servo motor retorna à sua posição inicial. O mesmo processo é repetido para a saída, onde a cancela é aberta quando um veículo é detectado e, em seguida, fechada após um atraso.

c) Detecção de Disponibilidade de Vagas:

Três sensores infravermelhos adicionais são posicionados nas vagas de estacionamento. Esses sensores determinam se uma vaga de estacionamento está ocupada ou disponível. Os dados desses sensores são enviados para o NodeMCU.

d) Processamento de Dados no NodeMCU:

O NodeMCU é o cérebro do sistema e processa os dados dos sensores.

Com base nas informações dos sensores de vaga, ele determina a disponibilidade de vagas de estacionamento. Ele também mantém o controle do tempo que um veículo passa no estacionamento.

e) Comunicação com o Adafruit IO:

O NodeMCU se comunica com a plataforma Adafruit IO por meio da internet.

Ele envia informações sobre a disponibilidade de vagas e os tempos de permanência dos veículos. Isso permite que os usuários monitorem o estado do estacionamento em tempo real de qualquer lugar do mundo com acesso à internet.

f) Controle Manual:

A plataforma Adafruit IO fornece uma interface de controle que inclui botões para operar manualmente as cancelas de entrada e saída.

Isso dá aos usuários a capacidade de controlar as cancelas quando necessário, além do funcionamento automatizado baseado em sensores.

4.4. Implementação e teste de unidade

A fase de Implementação e Teste de Unidade desempenha um papel fundamental no processo de desenvolvimento de software. Nesta etapa, vamos transformar os requisitos detalhados em código executável, e cada componente individual do sistema é submetido a testes minuciosos. A ideia primordial é garantir que cada

unidade de software funcione conforme o planeado, produza os resultados correctos e esteja livre de erros.

4.4.1. Implementação

Na etapa de Implementação, vamos traduzir o design do sistema em código de programação. Cada unidade de software, que pode ser uma função, classe, módulo ou componente específico, é codificada de acordo com as especificações.

4.4.1.1. Configuração de Adafruit IO

O Adafruit IO é uma plataforma de dados abertos que permite a agregação, visualização e análise de dados em tempo real na nuvem. Com o Adafruit IO, é possível carregar, exibir e monitorar dados via Internet, possibilitando a habilitação de projectos de IoT. Podemos controlar motores, ler dados de sensores e criar aplicações de IoT fascinantes pela Internet utilizando o Adafruit IO. Para começar a utilizar esta plataforma, é necessário criar uma conta no Adafruit IO. Para fazer isso, vamos acessar o site da Adafruit IO clicando em 'Get Started for Free' localizado no canto superior direito da tela.

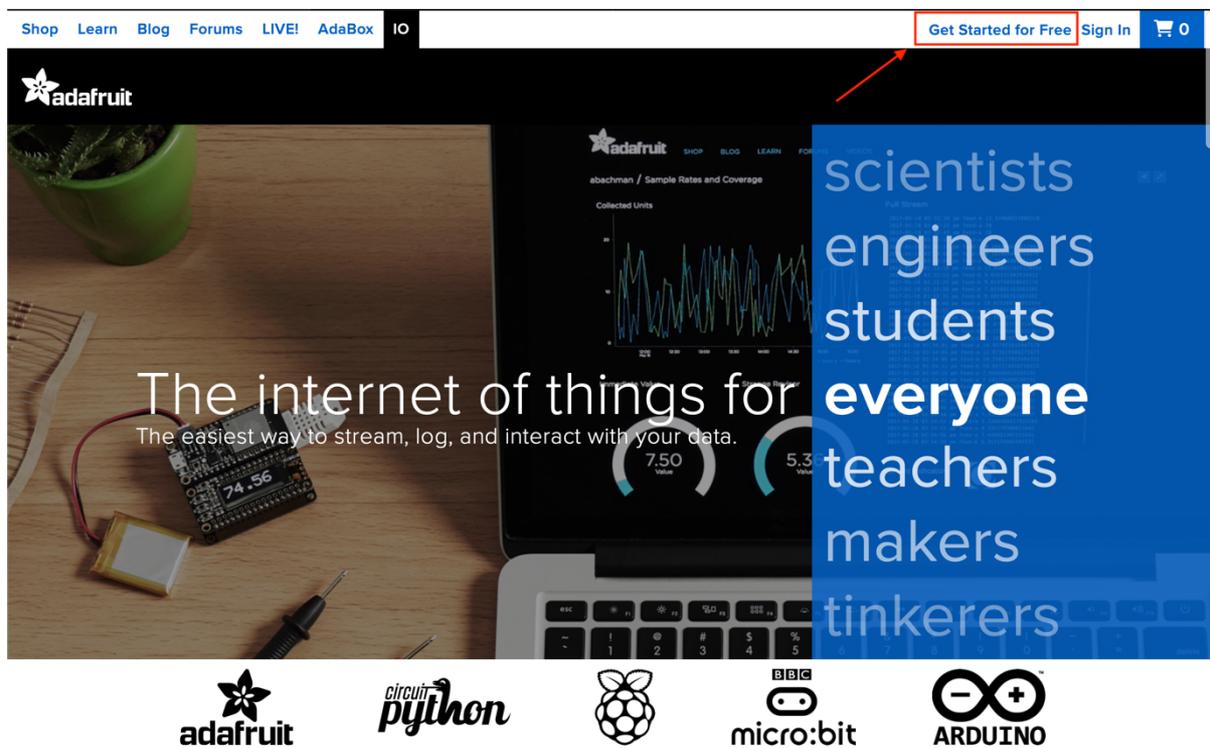


Figura 16: Pagina Principal do site Adafruit
Fonte: Autor (2023)

Após concluir o processo de criação da conta, realize o login em sua conta e, em seguida, clique na opção 'API Key' localizada no canto superior direito da tela. Isso permitirá que você obtenha o nome de usuário da sua conta e a chave AIO necessária para prosseguir.

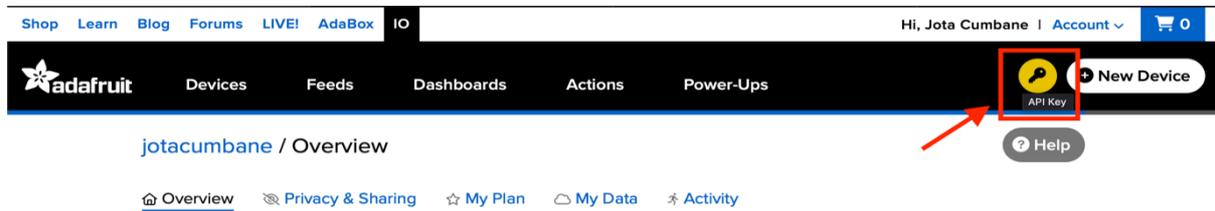


Figura 17: Obtenção da Chave AIO

Fonte: Autor (2023)

Ao clicar em 'AIO Key', uma janela será exibida, contendo sua chave AIO (AIO Key) e o nome de usuário associado à sua conta no Adafruit IO. Certifique-se de copiar essas informações, que serão necessárias posteriormente para escrever o código ou configurar seu projecto.

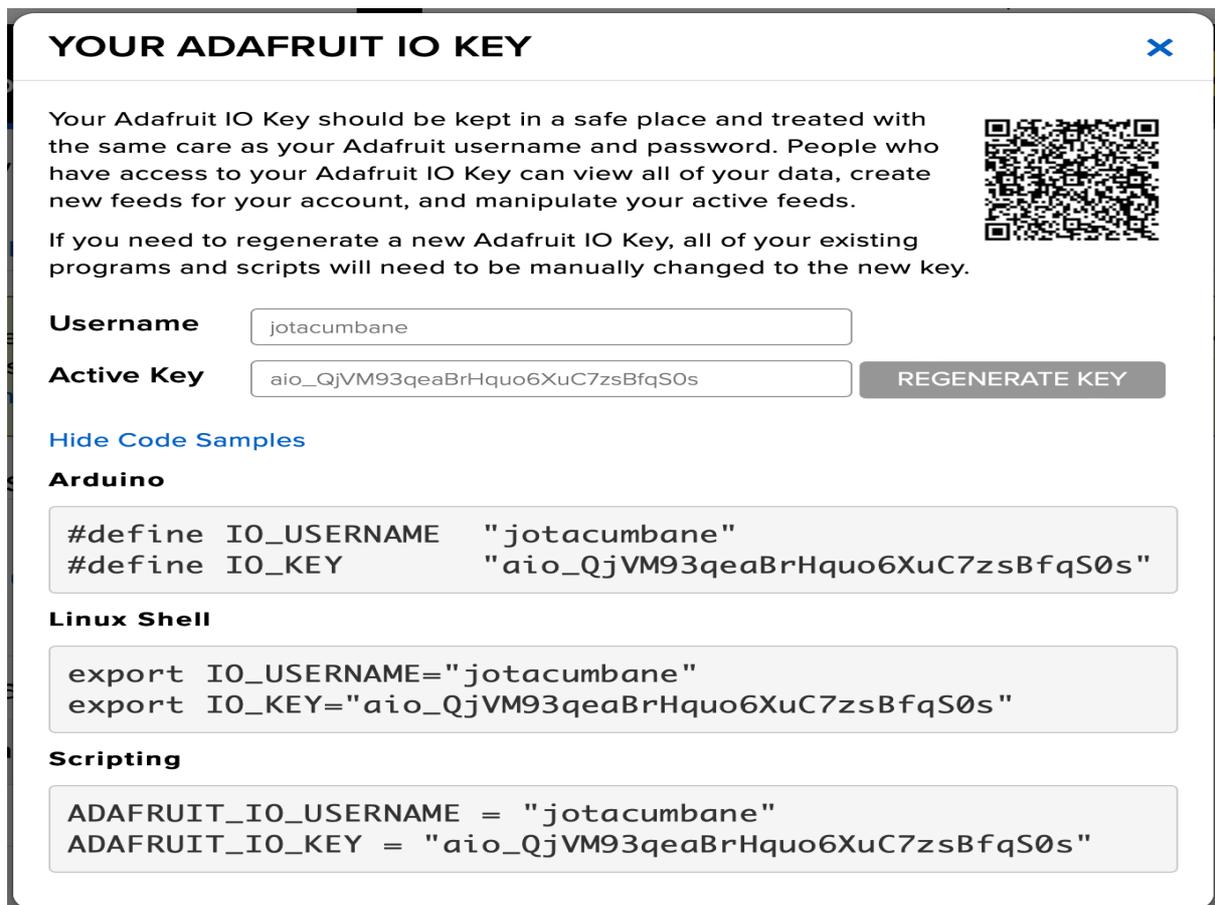


Figura 18: AIO Key

Fonte: Autor (2023)

Agora, após obter sua chave AIO e nome de usuário, é necessário criar um feed. Para fazer isso, siga os passos a seguir:

- ✓ Clique na opção 'Feeds'.
- ✓ Selecione a opção 'New Feed', conforme demonstrado na imagem.

Este procedimento permitirá a criação de um novo feed para armazenar e gerenciar os dados do seu projeto no Adafruit IO.

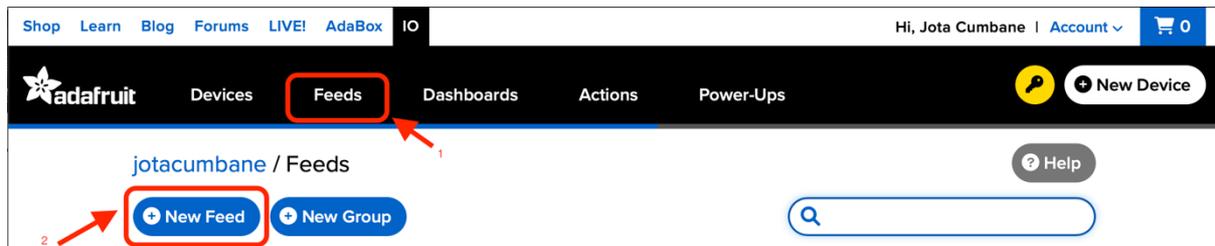


Figura 19: Acesso a aba criar um feed
Fonte: Autor (2023)

Após selecionar a opção 'New Feed', uma nova janela será aberta para inserir o Nome e, opcionalmente, a Descrição do feed. Esses detalhes ajudarão a identificar e organizar seus feeds de maneira mais clara e informativa. A descrição é opcional, mas pode ser útil para documentar a finalidade ou contexto do feed.

A imagem mostra uma janela modal intitulada 'Create a new Feed' com um ícone de fechar (X) no canto superior direito. O formulário contém dois campos de entrada: 'Name' e 'Description'. O campo 'Name' é um campo de texto único com uma borda azul. Abaixo dele, há o texto 'Maximum length: 128 characters. Used: 0'. O campo 'Description' é um campo de texto de área com uma borda cinza. No canto inferior direito, há dois botões azuis: 'Cancel' e 'Create'.

Figura 20: Campo de criação de um feed
Fonte: Autor (2023)

Após inserir o nome e, opcionalmente, a descrição do feed, clique em 'Create'. Isso criará o feed e seremos redirecionado para a página do feed recém-criado.

Para o presente projecto, criamos um total de nove feeds para as portas de saída, portões de entrada, entradas do slot 1 e saídas do slot 1, entradas do slot 2 e saídas do slot 2, bem como entradas do slot 3 e saídas do slot 3.

Agora, para visualizar esses feeds em uma única página, é necessário criar um painel de E/S do Adafruit. Para fazer isso, vamos seguir os passos a seguir:

- Clique na opção 'Dashboards'.
- Selecione 'New Dashboards'.

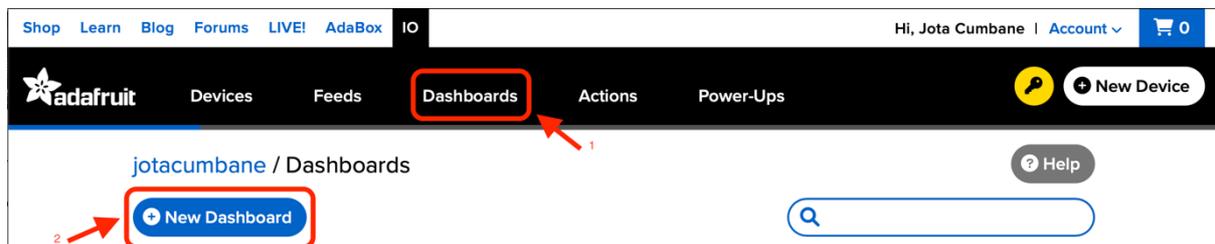


Figura 21: Acesso a aba criar um dashboard
Fonte: Autor (2023)

Uma nova janela será aberta, permitindo que insira o nome do dashboard e a descrição.

A screenshot of a modal window titled 'Create a new Dashboard'. The window has a close button (X) in the top right corner. It contains two input fields: a text field for 'Name' and a larger text area for 'Description'. At the bottom right, there are two blue buttons: 'Cancel' and 'Create'.

Figura 22: Campo de criação de uma dashboard
Fonte: Autor (2023)

Após inserir o nome e a descrição, vamos em 'Create'. Isso vai nos ajudar a criar um dashboard/painel que vai exibir os feeds e monitorar dados de forma organizada. Agora que o painel foi criado, Vamos adicionar os feeds ao painel.

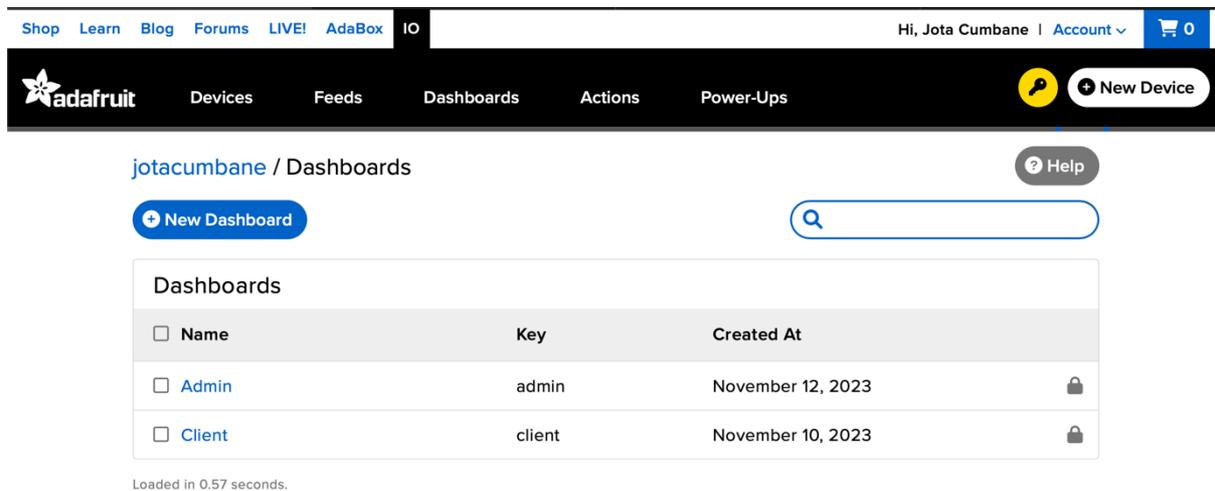


Figura 23: View de dashboards
Fonte: Autor (2023)

Para adicionar um feed ao painel, vamos seguir as seguintes etapas:

- ✓ No painel que criamos, vamos clicar no botão '+ Create New Block' no canto superior direito. Isso abrirá a opção de adicionar um feed ao painel.

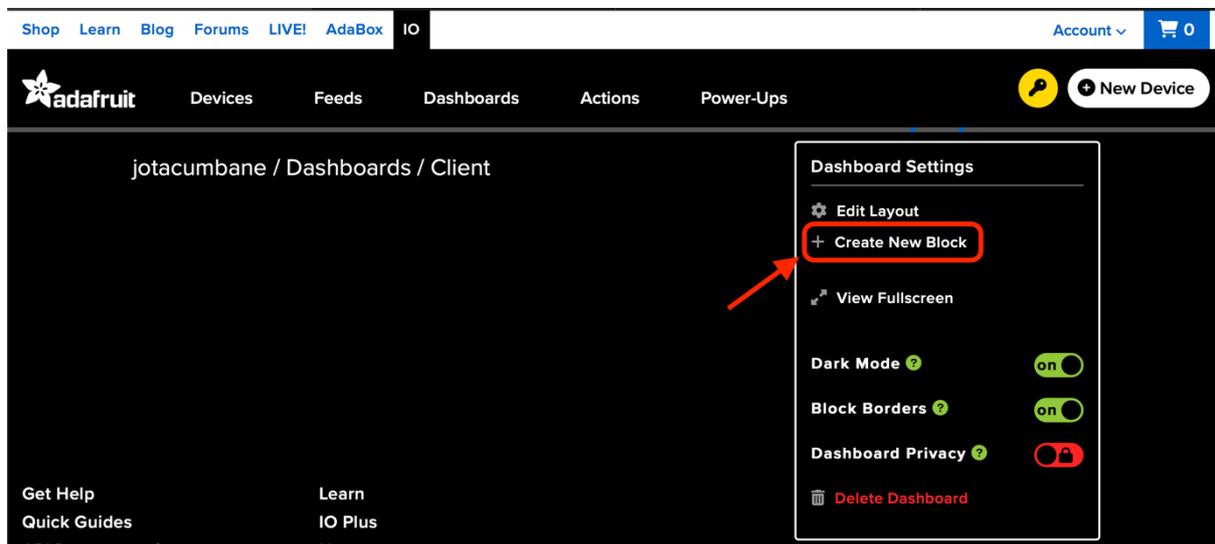


Figura 24: Criação de novo bloco
Fonte: Autor (2023)

- ✓ Selecionamos o bloco que desejamos adicionar ao painel. No nosso caso, vamos escolher o bloco de Texto.

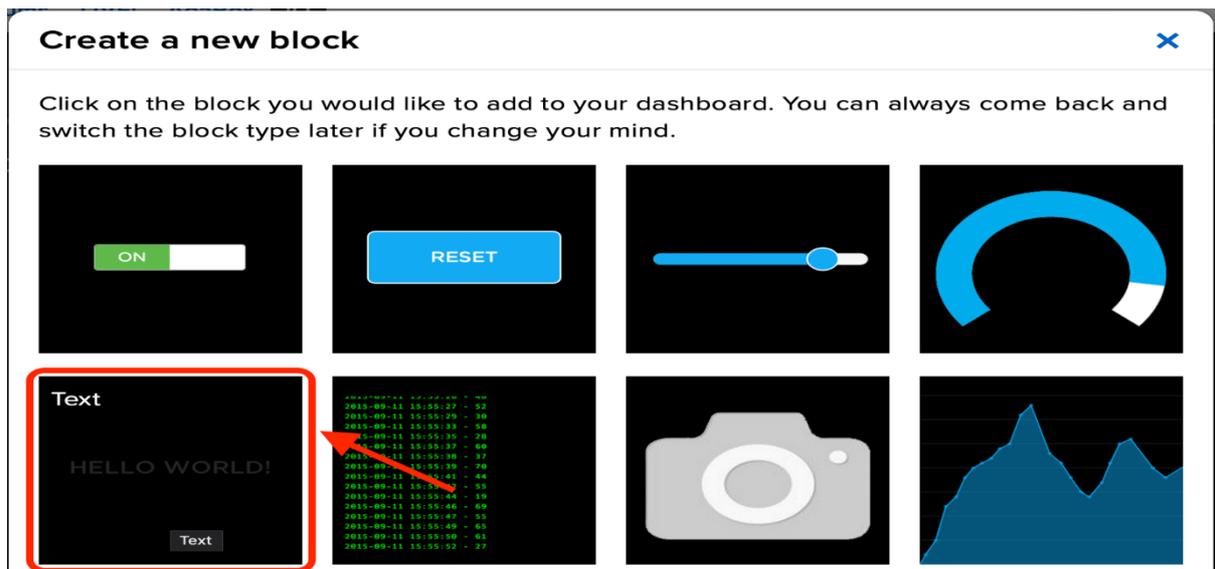


Figura 25: Tipos de blocos
 Fonte: Autor (2023)

- ✓ Na próxima janela selecionamos o feeds que desejamos adicionar ao painel. No nosso caso, vamos escolher os feeds relacionados à Vagas Disponíveis e Vagas Ocupadas, de acordo com o projeto, clique em 'Next step'.

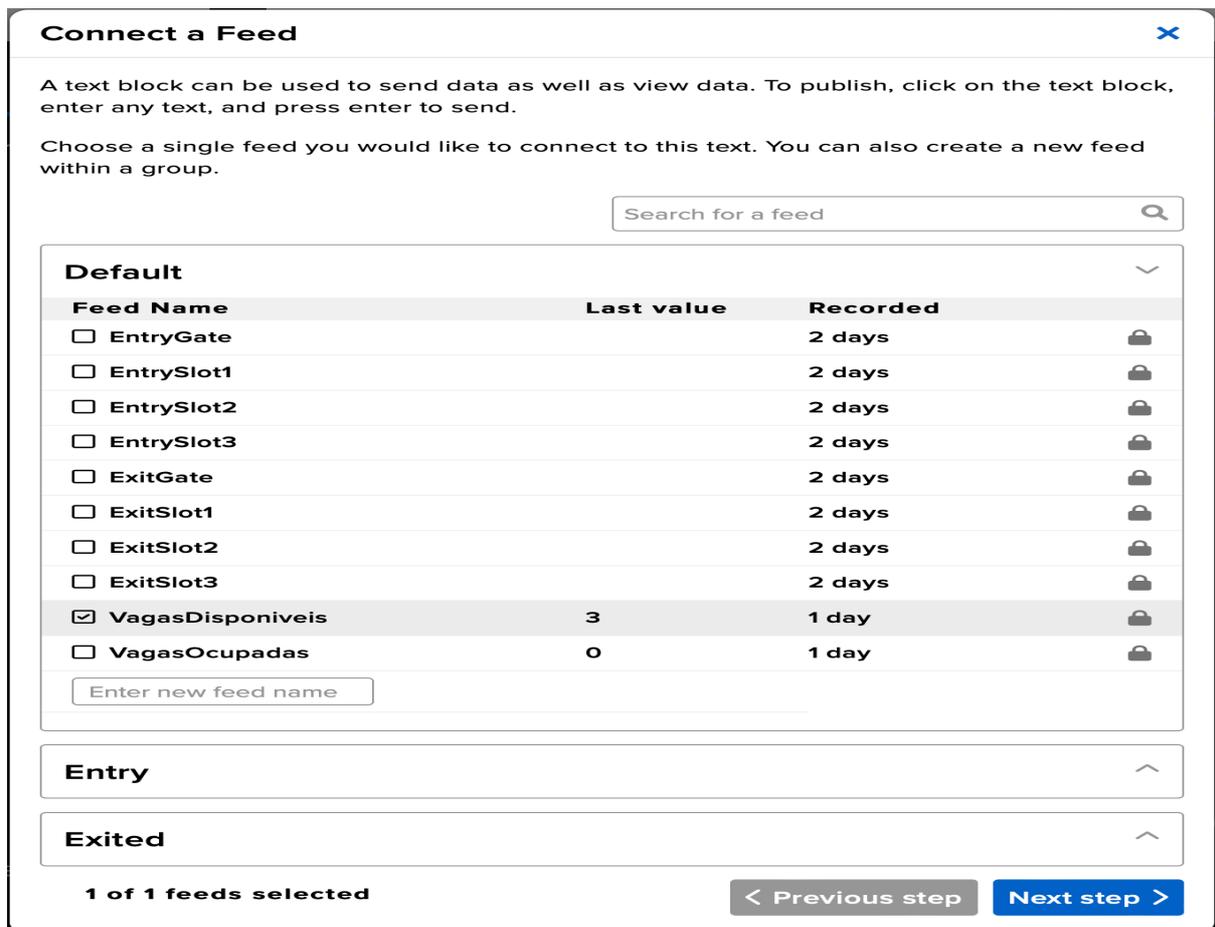


Figura 26: Feeds criados
 Fonte: Autor (2023)

✓ Na próxima janela será de configurações, depois clique em 'Create block'.

Block settings ✕

In this final step, you can give your block a title and see a preview of how it will look. Customize the look and feel of your block with the remaining settings. When you are ready, click the "Create Block" button to send it to your dashboard.

Block Title (optional)

Font Size

Static Text
When checked, ignore feed value and show the selected 'Static Text Value' all the time.

Static Text Value

When 'Static Text' is checked, use this value. Limited to 256 characters.

Decimal Places

Number of decimal places to display when value is a number. Defaults to -1 (unlimited).

Show Icon
When checked, show an icon with the value.

Icon

Show this icon next to the value.

Block Preview

Vagas Disponíveis

45

Text A text block can be used to send data as well as view data. To publish, click on the text block, enter any text, and press enter to send.

Test Value

Published Value

Figura 27: Configuração de bloco

Fonte: Autor (2023)

Após criar o bloco para visualizar o número de vagas disponíveis, seguimos o mesmo procedimento para criar outro bloco para vagas ocupadas.

Para criar os blocos de botões para o portão de entrada e saída devemos seguir as etapas semelhantes, mas, em vez de criar um bloco de TEXTO, vamos criar um bloco RESET ou bloco de botão.

Após criar todos os blocos, seu painel deve conter os controles para os portões de entrada e saída, bem como os blocos de texto para os detalhes do estacionamento.



Figura 28: Dashboard do admin
Fonte: Autor (2023)

4.4.1.2. Programação do NodeMCU

1. Depois de concluir a configuração do hardware, é necessário programar o ESP8266 NodeMCU. Para isso, é possível carregar o código no ESP8266 usando o Arduino IDE, seguindo as etapas abaixo:

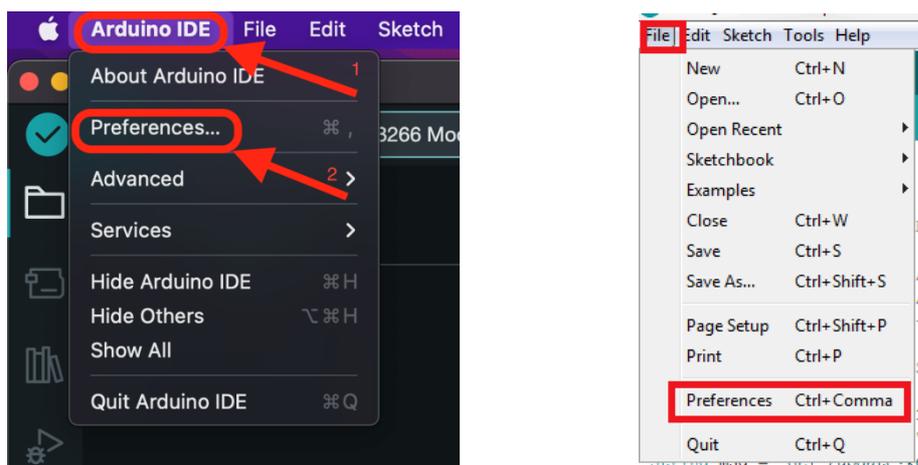


Figura 29: Acesso a menu preferences do Arduino IDE
Fonte: Autor (2023)

2. Digite `https://arduino.esp8266.com/stable/package_esp8266com_index.json` no campo 'Additional board manager URLs' e clique em 'ok'.

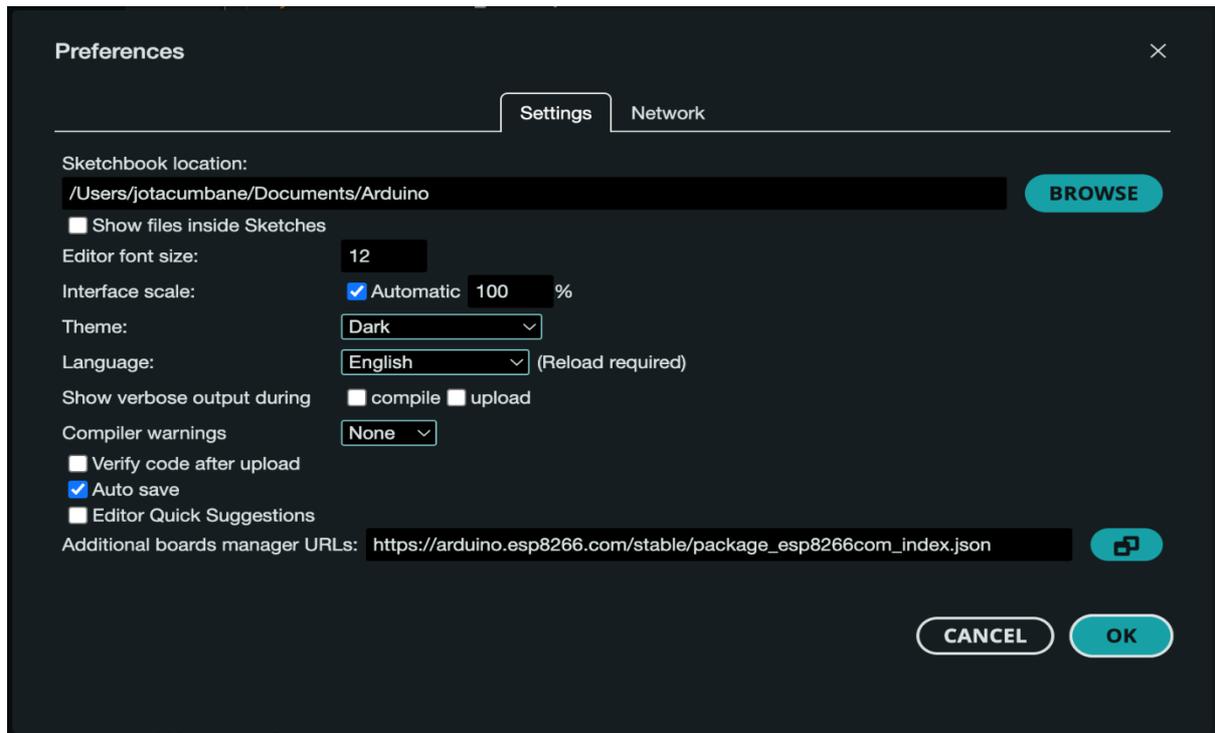


Figura 30: Campo de add de URL
Fonte: Autor (2023)

3. Vamos as **Tools > Board > Boards Manager**. Na janela Boards Manager, digite **ESP8266** na caixa de pesquisa, selecione a versão mais recente da placa e clique em 'install'.

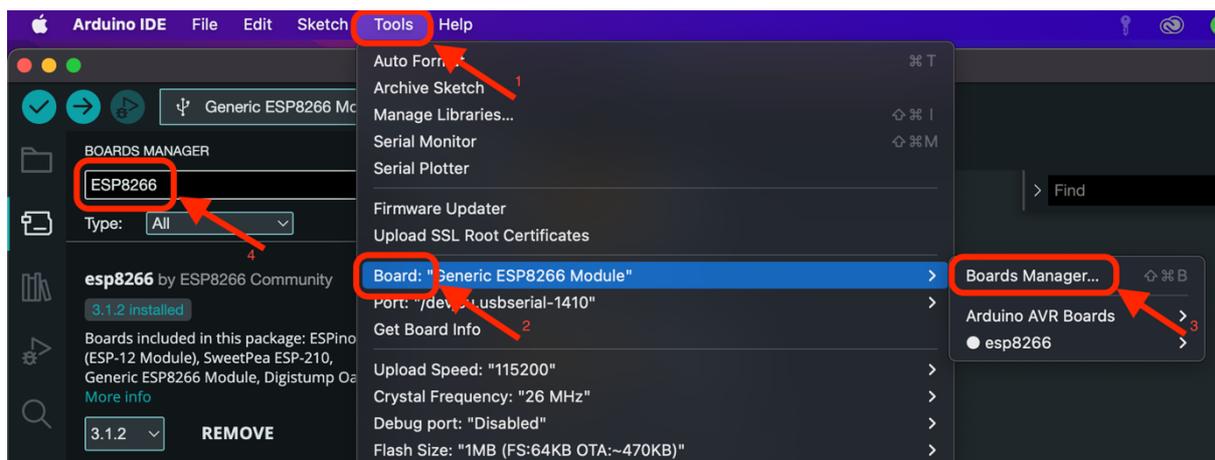


Figura 31: Instalação de drives da board
Fonte: Autor (2023)

4. Após a conclusão da instalação, vamos a **Tools ->Board -> e seleccione NodeMCU 1.0(ESP-12E Module)**. Onde podemos programar o ESP8266 com o Arduino IDE.

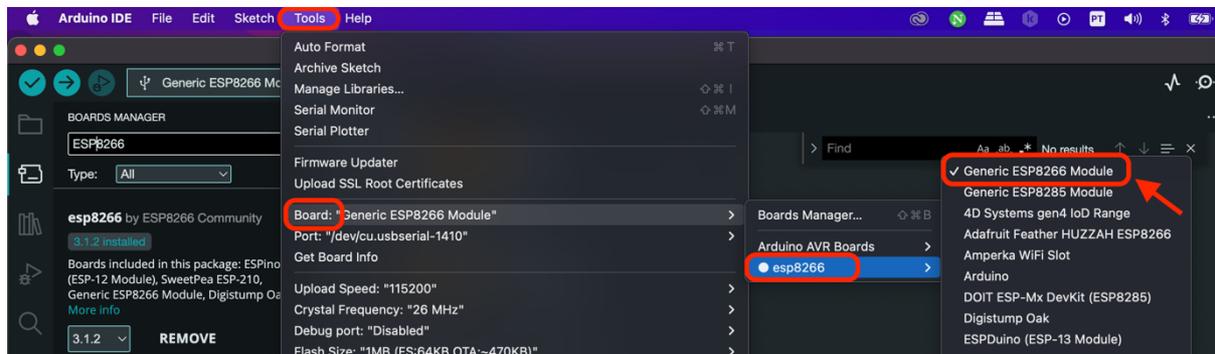


Figura 32: Selecção de drives de ESP8266

Fonte: Autor (2023)

4.4.1.3. Programação do ESP NodeMCU 8266

A programação do ESP NodeMCU 8266 é feita em linguagem C e é compilada por um software que permite a geração de executável do código que foi criado. O software utilizado foi o Arduino IDE que oferece um ambiente de desenvolvimento integrado e propicio para projectos IoT. O Programa é executado na sua própria memória, sem a necessidade de estar ligado fisicamente a outro tipo de dispositivo como computador ou microcontrolador, sendo apenas alimentado por uma fonte CC, e sua comunicação é por meio da rede wi-fi. Antes de iniciar com a programação do ESP NodeMCU é necessário instalar a sua biblioteca como ilustra a imagem a seguir.

Após terminar o download da biblioteca segue-se o processo da sua instalação que é feita no gerenciador de placas, feito isso já esta pronto o ambiente para a sua programação.

- **Importação das bibliotecas**

Antes de mais nada, devemos certificar-se de incluir todas as bibliotecas necessárias. No IDE, as bibliotecas Wi-Fi do ESP8266 e Servo.h já estão instaladas. Caso necessário, podemos baixar o cliente NTP e as bibliotecas MQTT da Adafruit nos seguintes links:

[NTPClient.h](#)

[Adafruit MQTT.h](#)

```
#include <ESP8266WiFi.h>
#include <Servo.h>
#include <NTPClient.h>
#include <WiFiUdp.h>
#include <NTPClient.h>;
#include <WiFiUdp.h>
#include "Adafruit_MQTT.h"
#include "Adafruit_MQTT_Client.h"
```

Em seguida, vamos incluir as credenciais de E/S Wi-Fi e Adafruit que copiamos do servidor de E/S da Adafruit. Estes incluirão o servidor MQTT, o número da porta, o nome de usuário e a chave AIO

```
const char *ssid = "WiFi Name";
const char *pass = "Password";

String time1= "89";

#define MQTT_SERV "io.adafruit.com"

#define MQTT_PORT 1883

#define MQTT_NAME "User Name"
```

"Agriculture Data" é o nome do feed.

```
Adafruit_MQTT_Subscribe EntryGate = Adafruit_MQTT_Subscribe(&mqtt, MQTT_NAME
"/f/EntryGate");

Adafruit_MQTT_Subscribe ExitGate = Adafruit_MQTT_Subscribe(&mqtt, MQTT_NAME
"/f/ExitGate");#define MQTT_PASS "AIO Key"
```

Em seguida conectamos o servo motor de entrada e saída aos pinos D4, D5 do

```
myservo.attach(D4);

myservos.attach(D5);

pinMode(carExited, INPUT);

pinMode(carEnter, INPUT);

pinMode(slot1, INPUT);

pinMode(slot2, INPUT);

pinMode(slot3, INPUT);
```

```
timeClient.update();  
  
hh = timeClient.getHours();  
  
mm = timeClient.getMinutes();  
  
ss = timeClient.getSeconds();
```

Após isso, mova o servo motor para abrir o portão de entrada e, em seguida, incremente a contagem para o portão de entrada e decrémente a contagem para o portão de saída. Por fim, publique os dados no painel de E/S da Adafruit.

```
if (entrysensor == 1) {  
    count= count+1;  
    myservos.write(OPEN_ANGLE);  
    delay(3000);  
    myservos.write(CLOSE_ANGLE);  
}  
  
if (exitsensor == 1) {  
    count= count-1;  
    myservo.write(OPEN_ANGLE);  
    delay(3000);  
    myservo.write(CLOSE_ANGLE);  
}  
  
if (! CarsParked.publish(count)) {}
```

Condição para a publicação dos dados da variável no feed de E/S do Adafruit.

```
if (s1 == 1 && s1_occupied == false) {  
    EntryTimeSlot1 = h + ":" + m;  
    s1_occupied = true;  
    if (! EntrySlot1.publish ((char*) EntryTimeSlot1.c_str())){} } }
```

Condição para verificar se o pino do sensor IR mudar para zero e a função booleana for true, publique o tempo de saída no feed de E/S do Adafruit.

```

if(s1 == 0 && s1_occupied == true) {
    ExitTimeSlot1 = h + " :" + m;
    s1_occupied = false;
    if (! ExitSlot1.publish((char*) ExitTimeSlot1.c_str())){ } }

```

Condição para verificar diretamente uma palavra específica em nosso feed assinado, e se a palavra coincidir com nossa palavra especificada, ou seja, 'ON', ela girará o servo motor para abrir o portão.

```

if (subscription == &EntryGate) { Serial.println((char*) EntryGate.lastread);
    if (!strcmp((char*) EntryGate.lastread, "ON")) {
        myservos.write(OPEN_ANGLE);
        delay(3000);
        myservos.write(CLOSE_ANGLE); }

```

Condição para verificar as etapas semelhantes aos sensores do slot2 e do slot 3.

```

if (s2 == 1&& s2_occupied == false) { EntryTimeSlot2 = h + " :" + m;
    s2_occupied = true;
    if (! EntrySlot2.publish((char*) EntryTimeSlot2.c_str())){ } }
if(s2 == 0 && s2_occupied == true) { ExitTimeSlot2 = h + " :" + m; s2_occupied = false;
    if (! ExitSlot2.publish((char*) ExitTimeSlot2.c_str())){ } }
if (s3 == 1&& s3_occupied == false) { EntryTimeSlot3 = h + " :" + m; s3_occupied = true;
    if (! EntrySlot3.publish((char*) EntryTimeSlot3.c_str())){ } }
if(s3 == 0 && s3_occupied == true) { ExitTimeSlot3 = h + " :" + m; s3_occupied = false;
    if (! ExitSlot3.publish((char*) ExitTimeSlot3.c_str())){ } }

```

4.5. Integração e teste de sistema

A fase de Integração e Teste de Sistema é fundamental no desenvolvimento do sistema de estacionamento inteligente baseado em IoT. Nesta etapa, todas as peças individuais do projecto são unidas e testadas como um sistema completo. Isso é crucial para garantir que o sistema funcione de maneira eficaz e cumpra com os objetivos.

4.5.1. Integração de Componentes

Durante a Integração de Componentes, as várias partes do sistema são unidas de acordo com o design e a arquitectura previamente estabelecidos.

NodeMCU e Componentes Físicos: As placas NodeMCU, sensores IR e servomotores são interligados de acordo com o diagrama de circuito abaixo:

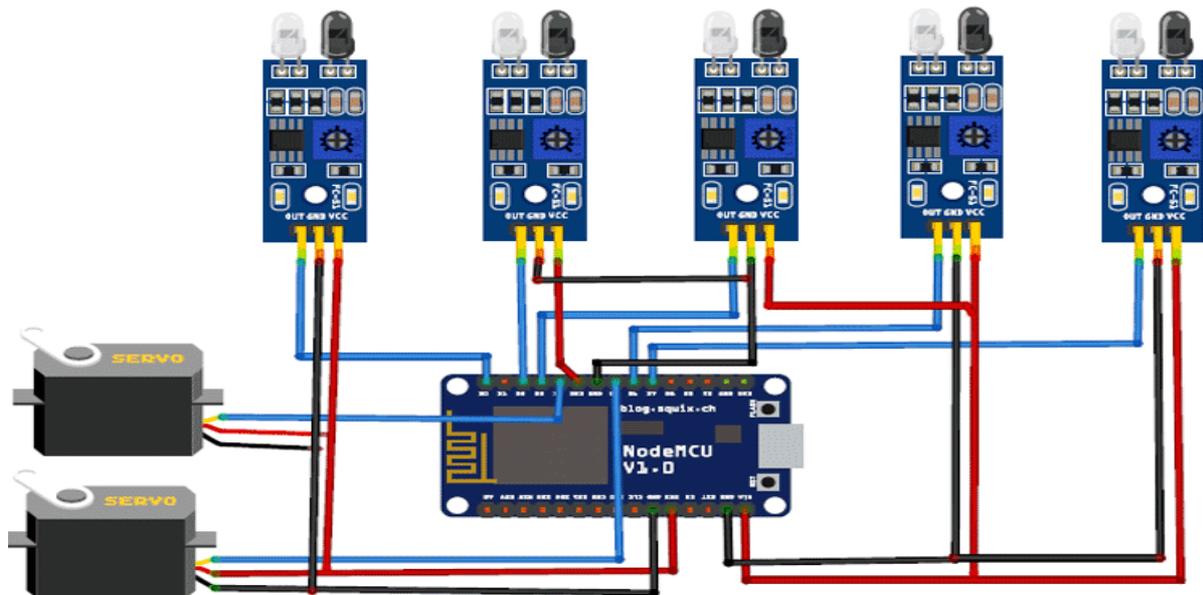


Figura 33: Diagrama de Circuitos

Fonte: Autor (2023)

Comunicação com Adafruit IO: Configuramos o NodeMCU para se comunicar com o serviço de nuvem Adafruit IO. Isso inclui a configuração das credenciais de autenticação, como o nome de usuário e a chave AIO conforme explicado no ponto **Configuração de Adafruit IO**.

Lógica de Controle: As funções lógicas do sistema, como a detecção de carros nas vagas de estacionamento, o acionamento dos servomotores e a publicação de dados no Adafruit IO, são integradas para criar um fluxo de funcionamento contínuo, conforme explicado no ponto **Importação das bibliotecas**.

4.6. Operação e manutenção

A fase de Operação e Manutenção é uma etapa contínua e crucial no ciclo de vida de um sistema de estacionamento inteligente baseado em IoT. Durante essa fase, o

sistema é implantado para uso real e requer monitoria constante, manutenção proativa e acomodação de mudanças. Aqui estão os principais aspectos da Operação e Manutenção:

4.6.1. Implantação do Sistema

Após a fase de teste bem-sucedida, o sistema é implantado em seu ambiente operacional real, como um estacionamento público. Isso envolve a instalação de todos os componentes físicos, como sensores IR, servomotores e dispositivos NodeMCU. Além disso, a configuração do sistema na infraestrutura de rede, incluindo a conexão com a Internet, é concluída.

Para a instalação do sistema no local do estudo serão necessários 2 meses conforme discriminado na Tabela 5: Implantação do Sistema.

Tabela 5: Implantação do Sistema

Mes	Dezembro 2023				Janeiro 2024			
	1	2	3	4	1	2	3	4
Semana								
Início de projecto	■							
Mapeamento da area		■						
Elaboração da planta			■					
Aquisição do hardware e licença			■	■	■			
Instalação do hardware						■		
Desenvolvimento do software						■	■	
Teste do sistema							■	
Formatação								■
Testes finais								■
Entrega do sistema								■

Fonte: Autor(2023)

4.6.2. Monitoria e Colecta de Dados

Durante a operação, o sistema colecta continuamente dados sobre a ocupação das vagas de estacionamento, as entradas e saídas de veículos e outras métricas relevantes. Esses dados são essenciais para garantir a funcionalidade contínua e podem ser usados para análises futuras.

4.6.3. Manutenção Proativa e Preventiva

A manutenção proativa e preventiva é fundamental para evitar falhas inesperadas do sistema. Isso envolve a realização de verificações regulares de todos os componentes, como sensores e servomotores. Qualquer componente que apresente sinais de desgaste, defeitos ou mau funcionamento deve ser substituído ou reparado imediatamente. O Plano Manutenção Preventiva do Sistema s e apresentado na Tabela 6: Manutenção Proativa e Preventiva, onde e descrito os Itens que compõem o sistema, os níveis de criticidade para cada componente do sistema se realmente precisam de uma intervenção de substituição ou apenas uma limpeza ou desligar o Sistema.

Tabela 6: Manutenção Proativa e Preventiva

Tipo de Manutenção	Descrição	Frequência	Responsável	Recursos necessários
Proactiva	Avaliação do estado dos sensores IoT	Mensal	Técnico de manutenção	Equipamento de teste
Proactiva	Inspeção visual de hardware	Semestral	Técnico de manutenção	Ferramentas de Inspeção
Proactiva	Monitoramento de uso e desgaste	Contínuo	Sistemas de monitoramento	Software de monitoramento
Proactiva	Avaliação de segurança do sistema	Semestral	Especialista em segurança	Ferramenta de segurança
Preventiva	Actualização de firmware	Trimestral	Administrador de sistema	Conexão a interne, software de Actualização
Preventiva	Teste de conectividade IoT	Mensal	Administrador de sistema	Equipamento de teste
Preventiva	Substituição de baterias (se aplicável)	Anual	Técnico de manutenção	Baterias de reposição
Preventiva	Backup de dados e configurações	trimestral	Administrador de sistema	Dispositivos de armazenamento

Fonte: Autor (2023)

4.6.4. Actualizações e Melhorias

À medida que o sistema é usado e os requisitos dos usuários evoluem, podem ser necessárias actualizações e melhorias. Isso pode incluir a adição de mais sensores para expandir a capacidade do estacionamento, ajustes no algoritmo de detecção de vagas ou melhorias na interface do usuário. As actualizações devem ser cuidadosamente planeadas e testadas antes de serem implementadas.

CAPÍTULO V – CONSIDERAÇÕES FINAIS E RECOMENDAÇÕES

5.1. Conclusão

O presente trabalho teve como objetivo principal abordar a ineficiência na gestão de vagas de estacionamento no Parque Municipal do Mercado Central em Maputo, propondo uma solução baseada em Internet das Coisas (IoT). No decorrer da pesquisa, identificamos e analisamos de forma abrangente os desafios enfrentados devido à falta de um sistema eficiente de gestão vagas de estacionamento na cidade.

No que diz respeito ao primeiro objetivo, que consistia em descrever como é feita a gestão da disponibilidade das vagas de estacionamento no parque, concluímos que a abordagem atual é carente de eficiência. A falta de informações em tempo real contribui para congestionamentos frequentes e frustrações para os motoristas, afetando negativamente a experiência urbana.

No segundo objetivo, que envolvia a pesquisa e comparação de soluções de IoT, constatamos que a ausência de tecnologias modernas, como IoT, tem sido uma limitação evidente no panorama atual de estacionamentos urbanos. A revisão da literatura e a análise de soluções disponíveis destacaram a necessidade de inovações para melhorar a eficiência na gestão de estacionamentos.

Quanto ao terceiro objetivo, relacionado à criação de um modelo que integre componentes de software e hardware baseados em IoT, desenvolvemos uma proposta que visa otimizar o uso do espaço e fornecer informações em tempo real sobre a disponibilidade de vagas. Este modelo busca preencher lacunas críticas na gestão de estacionamentos urbanos.

O quarto objetivo, que consistia no desenvolvimento de um protótipo funcional para testar o modelo proposto, foi alcançado com a implementação bem-sucedida do sistema. O protótipo foi testado em um ambiente de estacionamento de teste para avaliar seu desempenho, evidenciando a viabilidade prática da solução proposta.

Em resumo, todos os objetivos estabelecidos foram alcançados, proporcionando uma abordagem inovadora para a gestão de vagas de estacionamento. Concluímos que a implementação de um Sistema de Gestão de Vagas de Estacionamento baseado em

IoT pode não apenas otimizar o uso do espaço, mas também melhorar significativamente a experiência dos usuários na busca por estacionamento na cidade de Maputo.

5.2. Recomendações

Com base nas conclusões deste estudo, apresentamos as seguintes recomendações:

- ✓ **Implementação do Sistema Proposto:** Recomendamos a implementação do sistema de estacionamento inteligente baseado em IoT no Parque de Estacionamento do Conselho Municipal para melhorar a eficiência na gestão de vagas.
- ✓ **Avaliação Contínua:** Sugerimos que seja realizada uma avaliação contínua do sistema após a implementação, a fim de ajustar e aprimorar sua eficácia.
- ✓ **Treinamento de Pessoal:** Recomendamos o treinamento adequado dos operadores e equipes responsáveis pela gestão do sistema.
- ✓ **Políticas de Preços:** Recomendamos a revisão das políticas de preços de estacionamento para refletir o novo sistema e incentivar a adoção.
- ✓ **Para pesquisas futuras:** sugerimos explorar ***Segurança e Privacidade em Sistemas de IoT para Estacionamento, Mobilidade Urbana e Sustentabilidade, Acessibilidade e Inclusão em Parques de Estacionamento Inteligentes, Avaliação de Experiência do Usuário (UX)***, que podem aprofundar ainda mais nosso entendimento e aprimorar as soluções em gestão de vagas de estacionamento

Essas recomendações são projectadas para apoiar a implementação bem-sucedida do sistema proposto e garantir sua eficácia ao longo do tempo.

5.3. Referencias / Bibliografia

- Kerlinger, F. N. (1986). Foundations of Behavioral Research. 3ª Edição. Holt, Rinehart, and Winston.
- Gil, A. C. (2010). Como Elaborar Projetos de Pesquisa. 5ª Edição. São Paulo: Atlas.
- Sommerville, I. (2011). Engenharia de Software. 9ª Edição. Pearson.
- Albuquerque, J. P. (2015). Metodologia de Investigação Científica. 11ª Edição. Lisboa: Edições Sílabo.
- Creswell, J. W. (2013). Research Design: Qualitative, Quantitative, and Mixed Methods Approaches. Sage Publications.
- Johnson, T. R. (2017). Urban Parking: How the Lack of Parking Management Affects Cities. Routledge.
- Jones, M., et al. (2020). Urban Growth and the Challenge of Parking: The Case of Maputo, Mozambique. Sustainability, 12(18), 7643.
- Greenberg, D. (2021). Parking and the City. Island Press.
- BrasilPark. (2022)[BrasilPark](#)
- Oracle. (11 de 12 de 2022). Oracle - Gerenciamento de Cadeia de Suprimentos. Obtido de Internet das Coisas: <https://www.oracle.com/br/internet-of-things/what-is-iot/>
- AWS. (10 de 11 de 2022). AWS - Hub de conceitos de computação em nuvem. Obtido de Serviços de IoT: <https://aws.amazon.com/pt/what-is/iot/>
- Zikria, Y., et al. (2020). Smart Parking System using IoT and Cloud Computing. Procedia Computer Science, 171, 2260-2268. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1877050920312547>
- Hassija, V., et al. (2020). An Internet of Things (IoT) Based Smart Parking Management System: A Review. IEEE Access, 8, 109822-109842. Disponível em: <https://ieeexplore.ieee.org/document/9128935>
- Fathy, M., Abouelsoud, M., & Amer, M. (2019). Smart parking management using IoT. In 2019 7th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud) (pp. 105-112). IEEE.

- Ghiasi, A., Rahmani, A. M., Anzanpour, A., & Pahl, C. (2020). Smart parking management: An IoT-based framework. *Future Generation Computer Systems*, 103, 225-240.
- Hassija, V., Rishi, R., Singh, A. K., & Kumar, V. (2020). Smart parking system based on internet of things: A comprehensive review. *Sustainable Cities and Society*, 61, 102304.
- Pham, T. A., Pham, M. H., Nguyen, H. D., & Le, V. H. (2019). Smart parking management system based on IoT and data analytics. *Journal of Science and Technology*, 16(2), 34-43.
- Singh, R., Singh, S. K., & Singh, S. P. (2019). IoT based smart parking management system using machine learning algorithm. *International Journal of Computer Applications*, 179(51), 25-29.

ANEXO 1: GUIÃO DA ENTREVISTA

Objectivo da Entrevista: Coletar informações sobre o funcionamento e desafios do parque de estacionamento do Conselho Municipal de Maputo.

Perguntas e Respostas:

1. Sobre o Parque de Estacionamento

1.1. **Pergunta:** Qual é a capacidade total do parque de estacionamento?

Resposta: A capacidade total é de 340 vagas.

1.2. **Pergunta:** Como o parque de estacionamento é gerenciado actualmente?

Resposta: Actualmente, usamos um sistema manual para a gestão, incluindo a cobrança de taxas.

1.3. **Pergunta:** Quais são os principais desafios na gestão do parque de estacionamento?

Resposta: Os desafios incluem o congestionamento, a falta de informações em tempo real sobre a ocupação das vagas e a necessidade de manter a segurança.

2. Sobre a Possibilidade de Implantação de IoT

2.1. **Pergunta:** Você acredita que a implantação de um sistema baseado em Internet of Things (IoT) pode melhorar a gestão do parque de estacionamento?

Resposta: Sim, acredito que pode melhorar a eficiência, proporcionando informações em tempo real sobre a ocupação das vagas.

2.2. **Pergunta:** Quais benefícios você vê na adoção de um sistema IoT para gestão de vagas de estacionamento?

Resposta: Os benefícios incluem melhor controle de ocupação, cobrança automatizada e a capacidade de fornecer informações em tempo real aos motoristas.

3. Sobre a Experiência com Tecnologia

3.1. **Pergunta:** O parque de estacionamento já utilizou alguma tecnologia avançada anteriormente?

Resposta: Até agora, utilizamos apenas um sistema de câmeras de segurança.

3.2. **Pergunta:** Qual é a sua opinião sobre a confiabilidade e a manutenção de tecnologias avançadas?

Resposta: Acredito que a confiabilidade é fundamental, e a manutenção deve ser bem planejada para evitar interrupções no funcionamento.

4. Sobre Custos e Orçamento

4.1. **Pergunta:** Qual é o orçamento actual para a gestão do parque de estacionamento?

Resposta: O orçamento anual actual é de XXX.

4.2. **Pergunta:** Está disposto a investir em um sistema IoT de gestão de estacionamento?

Resposta: Estamos abertos à ideia, desde que os custos sejam justificáveis e haja um retorno claro do investimento.

ANEXO 2: CÓDIGO DO PROGRAMA

```
#include <dummy.h>
#include <ESP8266WiFi.h>
#include <Servo.h>
#include <NTPClient.h>
#include <WiFiUdp.h>
#include "Adafruit_MQTT.h"
#include "Adafruit_MQTT_Client.h"

const char *ssid = "Talunga Wi-Fi"; // Insira o nome da sua rede Wi-Fi
const char *pass = "TaaaaaaaTa_2!!!!"; // Insira a senha da sua rede Wi-Fi

#define MQTT_SERV "io.adafruit.com"
#define MQTT_PORT 1883
#define MQTT_NAME "jotacumbane"
#define MQTT_PASS "aio_QjVM93qeaBrHquo6XuC7zsBfqS0s"
WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP, "pool.ntp.org", 7200, 60000); // Ajuste para GMT+2
Servo myservo; // Servo da barreira de entrada
Servo myservos; // Servo da barreira de saída

int carEnter = 16; // Sensor de entrada
int carExited = 4; // Sensor de saída
int slot3 = 13;
int slot2 = 12;
int slot1 = 0;
int totalVagas = 3; // Número total de vagas
int count = 0;
int CLOSE_ANGLE = 180; // Ângulo de fechamento do servo motor
int OPEN_ANGLE = 0; // Ângulo de abertura do servo motor
int hh, mm, ss;
String h, m, EntryTimeSlot1, ExitTimeSlot1, EntryTimeSlot2, ExitTimeSlot2, EntryTimeSlot3,
ExitTimeSlot3;
boolean entrysensor, exitsensor, s1, s2, s3;
boolean s1_occupied = false;
boolean s2_occupied = false;
boolean s3_occupied = false;

WiFiClient client;
Adafruit_MQTT_Client mqtt(&client, MQTT_SERV, MQTT_PORT, MQTT_NAME, MQTT_PASS);

// Configuração do tópico que está sendo inscrito
Adafruit_MQTT_Subscribe EntryGate = Adafruit_MQTT_Subscribe(&mqtt, MQTT_NAME
"/f/EntryGate");
```

```
Adafruit_MQTT_Subscribe ExitGate = Adafruit_MQTT_Subscribe(&mqtt, MQTT_NAME
"/f/ExitGate");
```

```
// Configuração dos tópicos para publicação
```

```
Adafruit_MQTT_Publish VagasOcupadas= Adafruit_MQTT_Publish(&mqtt, MQTT_NAME
"/f/VagasOcupadas");
```

```
Adafruit_MQTT_Publish EntrySlot1 = Adafruit_MQTT_Publish(&mqtt, MQTT_NAME "/f/EntrySlot1");
```

```
Adafruit_MQTT_Publish ExitSlot1 = Adafruit_MQTT_Publish(&mqtt, MQTT_NAME "/f/ExitSlot1");
```

```
Adafruit_MQTT_Publish EntrySlot2 = Adafruit_MQTT_Publish(&mqtt, MQTT_NAME "/f/EntrySlot2");
```

```
Adafruit_MQTT_Publish ExitSlot2 = Adafruit_MQTT_Publish(&mqtt, MQTT_NAME "/f/ExitSlot2");
```

```
Adafruit_MQTT_Publish EntrySlot3 = Adafruit_MQTT_Publish(&mqtt, MQTT_NAME "/f/EntrySlot3");
```

```
Adafruit_MQTT_Publish ExitSlot3 = Adafruit_MQTT_Publish(&mqtt, MQTT_NAME "/f/ExitSlot3");
```

```
Adafruit_MQTT_Publish VagasDisponiveis = Adafruit_MQTT_Publish(&mqtt, MQTT_NAME
"/f/VagasDisponiveis");
```

```
void setup()
```

```
{
```

```
  delay(1000);
```

```
  Serial.begin(9600);
```

```
  mqtt.subscribe(&EntryGate);
```

```
  mqtt.subscribe(&ExitGate);
```

```
  timeClient.begin();
```

```
  myservo.attach(2); // Pino do servo para D6
```

```
  myservos.attach(14); // Pino do servo para D5
```

```
  pinMode(carExited, INPUT);
```

```
  pinMode(carEnter, INPUT);
```

```
  pinMode(slot1, INPUT);
```

```
  pinMode(slot2, INPUT);
```

```
  pinMode(slot3, INPUT);
```

```
  WiFi.begin(ssid, pass);
```

```
  Serial.print("Conectando a ");
```

```
  Serial.print(ssid);
```

```
  while (WiFi.status() != WL_CONNECTED)
```

```
  {
```

```
    Serial.print(".");
```

```
    delay(500);
```

```
  }
```

```
  Serial.println();
```

```
  Serial.print("Conectado a ");
```

```
  Serial.println(ssid);
```

```
  Serial.print("Endereço IP: ");
```

```
  Serial.println(WiFi.localIP());
```

```
}
```

```
void loop()
```

```

{
  MQTT_connect();
  timeClient.update();
  hh = timeClient.getHours();
  mm = timeClient.getMinutes();
  ss = timeClient.getSeconds();
  h = String(hh);
  m = String(mm);
  h + " ." + m;

  entrysensor = !digitalRead(carEnter);
  exitsensor = !digitalRead(carExited);
  s1 = digitalRead(slot1);
  s2 = digitalRead(slot2);
  s3 = digitalRead(slot3);

  if (entrysensor == 1)
  {
    count = count + 1;
    myservos.write(OPEN_ANGLE);
    delay(3000);
    myservos.write(CLOSE_ANGLE);
  }

  if (exitsensor == 1 && count > 0)
  {
    count = count - 1;
    myservo.write(OPEN_ANGLE);
    delay(3000);
    myservo.write(CLOSE_ANGLE);
  }

  // Limita o valor de count para garantir que não seja negativo
  count = max(0, count);

  if (!VagasOcupadas.publish(count))
  {
  }

  if (s1 == 1 && !s1_occupied)
  {
    Serial.println("Ocupada1 ");
    EntryTimeSlot1 = h + " ." + m;
    Serial.print("EntryTimeSlot1 ");
    Serial.println(EntryTimeSlot1);
  }
}

```

```

s1_occupied = true;
if (!EntrySlot1.publish((char *)EntryTimeSlot1.c_str()))
{
}
}
if (s1 == 0 && s1_occupied)
{
Serial.println("Disponível1 ");
ExitTimeSlot1 = h + " ." + m;
Serial.print("ExitTimeSlot1 ");
Serial.println(ExitTimeSlot1);

s1_occupied = false;
if (!ExitSlot1.publish((char *)ExitTimeSlot1.c_str()))
{
}
}

if (s2 == 1 && !s2_occupied)
{
Serial.println("Ocupada2 ");
EntryTimeSlot2 = h + " ." + m;
Serial.print("EntryTimeSlot2 ");
Serial.println(EntryTimeSlot2);

s2_occupied = true;
if (!EntrySlot2.publish((char *)EntryTimeSlot2.c_str()))
{
}
}
if (s2 == 0 && s2_occupied)
{
Serial.println("Disponível2 ");
ExitTimeSlot2 = h + " ." + m;
Serial.print("ExitTimeSlot2 ");
Serial.println(ExitTimeSlot2);

s2_occupied = false;
if (!ExitSlot2.publish((char *)ExitTimeSlot2.c_str()))
{
}
}

if (s3 == 1 && !s3_occupied)

```

```

{
  Serial.println("Ocupada3 ");
  EntryTimeSlot3 = h + " ." + m;
  Serial.print("EntryTimeSlot3 ");
  Serial.println(EntryTimeSlot3);

  s3_occupied = true;
  if (!EntrySlot3.publish((char *)EntryTimeSlot3.c_str()))
  {
  }
}
if (s3 == 0 && s3_occupied)
{
  Serial.println("Disponível3 ");
  ExitTimeSlot3 = h + " ." + m;
  Serial.print("ExitTimeSlot3 ");
  Serial.println(ExitTimeSlot3);

  s3_occupied = false;
  if (!ExitSlot3.publish((char *)ExitTimeSlot3.c_str()))
  {
  }
}

// Calcula o número de vagas disponíveis
int vagasDisponiveis = totalVagas - count;

// Exibe e publica o número de vagas disponíveis
Serial.print("Vagas disponíveis: ");
Serial.println(vagasDisponiveis);

if (!VagasDisponiveis.publish(vagasDisponiveis))
{
  Serial.println(F("Falha ao publicar vagas disponíveis no MQTT."));
}

Adafruit_MQTT_Subscribe *subscription;
while ((subscription = mqtt.readSubscription(5000))
{
  if (subscription == &EntryGate)
  {
    Serial.println((char *)EntryGate.lastread);
    if (!strcmp((char *)EntryGate.lastread, "ON"))
    {
      myservos.write(OPEN_ANGLE);
    }
  }
}

```

```

    delay(3000);
    myservos.write(CLOSE_ANGLE);
  }
}
if (subscription == &ExitGate)
{
  Serial.println((char *)ExitGate.lastread);
  if (!strcmp((char *)ExitGate.lastread, "ON"))
  {
    myservo.write(OPEN_ANGLE);
    delay(3000);
    myservo.write(CLOSE_ANGLE);
  }
}
}
}

```

```

void MQTT_connect()
{
  int8_t ret;

  // Para se já estiver conectado.
  if (mqtt.connected())
  {
    return;
  }

  uint8_t retries = 3;
  while ((ret = mqtt.connect()) != 0)
  {
    mqtt.disconnect();
    delay(5000);
    retries--;
    if (retries == 0)
    {
      while (1)
      ;
    }
  }
}
}

```