Universidade do Minho
Escola de Engenharia

Beatriz José Cunha Rodrigues

Pervasive Modular *Data Science* – Otimização e interoperabilidade

Pervasive Modular *Data Science* – Otimização e interoperabilidade
Beatriz José Cunha Rodrigues
UMinho

Outubro de 2023

Universidade do Minho
Escola de Engenharia

Beatriz José Cunha Rodrigues

Pervasive Modular *Data Science* – Otimização e interoperabilidade

Relatório de Dissertação de Mestrado
Mestrado Integrado em Engenharia e Gestão de Sistemas de Informação

Trabalho efetuado sob a orientação do/da/de
Professor Carlos Filipe da Silva Portela

Outubro de 2023

DIREITOS DE AUTOR E CONDIÇÕES DE UTILIZAÇÃO DO TRABALHO POR TERCEIROS

AGRADECIMENTOS

Com o encerramento de uma das fases mais marcantes da minha vida, gostaria de expressar a minha gratidão a todos que, de alguma forma, me incentivaram e ajudaram a tornar isso possível.

Primeiramente, quero agradecer ao meu orientador, o Professor Carlos Filipe da Silva Portela, que me apoiou durante toda a elaboração desta tese e sempre esteve disponível para esclarecimentos.

Quero também agradecer a toda a equipe da IOTech pelo tempo e conhecimento compartilhados, o que tornou a elaboração desta tese possível. Um agradecimento especial ao Daniel Carneiro, que sempre se mostrou disponível para me ajudar em qualquer dificuldade que surgiu ao longo do projeto.

Aos meus colegas e amigos que, ao longo da minha jornada académica, não só me ajudaram na realização de trabalhos, mas também encheram essa jornada de diversão e alegria. Gostaria de fazer um agradecimento especial à Ana Carolina Pereira, cujas piadas secas ao longo de 18 anos de amizade sempre conseguem me surpreender, e à Daniela Ferreira, agradeço por estar sempre ao meu lado e pelos conselhos valiosos.

Um grande agradecimento para toda a minha família, em especial aos meus pais e ao meu irmão, que sempre me incentivaram e proporcionaram todas as oportunidades que me permitiram atingir todos os meus objetivos académicos.

Por fim, quero agradecer à Universidade do Minho, mais especificamente ao Departamento de Sistemas de Informação, e a todos os seus membros que, de alguma forma, contribuíram para a minha formação e para as minhas conquistas académicas.

DECLARAÇÃO DE INTEGRIDADE

Declaro ter atuado com integridade na elaboração do presente trabalho académico e confirmo que não recorri à prática de plágio, nem a qualquer forma de utilização indevida ou falsificação de informações ou resultados em nenhuma das etapas conducente à sua elaboração.

Mais declaro que conheço e que respeitei o Código de Conduta Ética da Universidade do Minho.

RESUMO

Pervasive Modular *Data Science* – Otimização e interoperabilidade

Num contexto de avanços tecnológicos, a criação constante de conjuntos de dados tornou-se uma realidade marcante na sociedade atual. Neste cenário, a IOTech concebeu o ioScience com o objetivo de superar as dificuldades inerentes à análise em tempo real, proporcionando soluções eficazes para o tratamento de conjuntos de dados complexos e variados. Assim, procurando otimizar esta plataforma no âmbito do projeto ioCity, a IOTech precisa que ela seja melhorada ao nível dos módulos de interoperabilidade, visualização e previsão.

Durante este projeto, foram aplicadas diferentes metodologias: Design Science Research, no que diz respeito à investigação, e a *framework* SCRUM e o CRISP-DM, no que diz respeito ao desenvolvimento da componente prática.

Na primeira fase desta dissertação, foram adquiridos os conceitos necessários para o seu desenvolvimento, como, por exemplo, *Data Science* e Data Mining, juntamente com uma compreensão da patente ioScience. Após a fase inicial, foi iniciada a componente prática desta dissertação, na qual foi possível responder positivamente à questão de pesquisa "Qual a viabilidade de integrar um módulo preditivo no ioScience, seguindo as regras de construção deste?" com a implementação do módulo preditivo no ioScience, segundo as regras do mesmo, e a subsequente alteração na arquitetura da solução. Além disso, os demais objetivos do trabalho foram alcançados com a obtenção dos seguintes resultados: Implementação do Módulo Preditivo no ioScience; Alteração na arquitetura da solução; Otimização e implementação de componentes visuais no ioScience; Otimização e implementação de funcionalidades no ioScience; Reorganização no código; Teste utilizando dados o projeto do ioCity; Elaboração de 24 modelos de previsão; e Criação da API de previsão em Python. De forma geral, pode-se afirmar que este trabalho resultou na apresentação de um protótipo aprimorado da plataforma.

Para este caso de estudo e como prova de conceito, a empresa IOTech foi utilizada, uma vez que esta foi responsável por fornecer os dados necessários para apoiar esta dissertação.

Palavras-chave: análises em tempo-real; *Data Science*; módulo preditivo; otimização.

ABSTRACT

Pervasive Modular *Data Science* - Optimisation and interoperability

In a context of technological advances, the constant creation of data sets has become a marked reality in today's society. In this scenario, IOTech designed ioScience with the aim of overcoming the difficulties inherent in real-time analysis, providing effective solutions for handling complex and varied data sets. In order to optimize this platform as part of the ioCity project, IOTech needs to improve its interoperability, visualization and forecasting modules.

During this project, different methodologies were applied: Design Science Research for the research, and the SCRUM framework and CRISP-DM for the development of the practical component.

In the first phase of this dissertation, the concepts necessary for its development were acquired, such as *Data Science* and Data Mining, along with an understanding of the ioScience patent. After the initial phase, the practical component of this dissertation began, in which it was possible to positively answer the research question "What is the viability of integrating a predictive module into ioScience, following its construction rules?" with the implementation of the predictive module in ioScience, according to its rules, and the subsequent change in the solution's architecture. In addition, the other objectives of the work were achieved with the following results: Implementation of the Predictive Module in ioScience; Change in the solution's architecture; Optimization and implementation of visual components in ioScience; Optimization and implementation of functionalities in ioScience; Reorganization of the code; Testing using data from the ioCity project; Development of 24 prediction models; and Creation of the prediction API in Python. In general, it can be stated that this work resulted in the presentation of an enhanced prototype of the platform.

For this case study and as a proof of concept, the company IOTech was utilized since it was responsible for providing the necessary data to support this dissertation.

Keywords: real-time analytics; *Data Science*; predictive module; optimization.

ÍNDICE

LISTA DE ABREVIATURAS E SIGLAS

API	Application Programming Interface
BI	Business Intelligence
CRISP- DM	Cross-Industry Standard Process for Data Mining
CRM	Customer Relationship Management
CSV	Comma-Separated Values
DM	Data Mining
DS	Design Science
DsaaS	*Data Science* as a Service
DSR	Design Science Research
DT	Decision tree
ERP	Enterprise Resource Planning
ETL	Extract, Transform, Load
HOLAP	Hybrid On-Line Analytical Processing
I&D	Investigação e Desenvolvimento
IoT	Internet of Things
JSON	JavaScript Object Notation
KDD	Knowledge Discovery in Databases
KPI	Key Performance Indicator
LR	Linear Regressions
MAE	Mean absolute error
MOLAP	Multidimensional On-Line Analytical Processing
MSE	Mean squared error

NB	Naive Bayes
NN	Neural Networks
OLAP	Online Analytical Processing
RAE	Relative absolute error
RF	Random Forest
ROLAP	Relational On-Line Analytical Processing
SCM	Supply Chain Management
SOW	Statements Of Work
XML	Extensible Markup Language

LISTA DE FIGURAS

LISTA DE TABELAS

1. INTRODUÇÃO

Neste primeiro capítulo é apresentado o enquadramento do projeto de dissertação, bem como, uma breve exposição da motivação para a realização do mesmo. Além disto, são apresentados os objetivos e os resultados esperados, de forma a ser possível no final do mesmo responder à questão de investigação identificada. Por último, é apresentada a estruturação deste documento.

1.1 Enquadramento e Motivação

Este projeto de dissertação leva em consideração o aumento considerável na formação de conjunto de dados, devido essencialmente aos diversos avanços tecnológicos na atualidade. Isto pode ser verificado quando é observado o ambiente à nossa volta, onde é possível identificar diversos dispositivos que realizam a recolha destes dados constantemente, sendo exemplos destes, smartphones, sites da internet e leitores automáticos de matrículas.

Com isto em mente, o apoio à decisão em tempo-real com base em dados gerados no momento é visto pelas organizações, cada vez mais, como um fator decisivo para o sucesso na tomada de uma decisão. No entanto, as organizações têm dificuldades em fazer uma análise desses dados em tempo-real, devido à complexidade, quantidade e diversidade dos mesmos. Tendo por base este conceito, a IOTech desenvolveu o ioScience (Filipe Portela & Gisela Fernandes, 2022) e precisa que o mesmo seja otimizado, incluindo os resultados originais e outros desenvolvidos no âmbito do projeto ioCity, melhorando os módulos de interoperabilidade, visualização e previsão. Em relação a estes módulos, é importante clarificar o seguinte:

- O módulo da interoperabilidade refere-se à solidificação da característica interoperabilidade dos dados, isto é, a implementação e integração de diferentes conjuntos de dados na plataforma do ioScience como forma de teste da mesma;
- O módulo de visualização refere-se a otimizações visuais e funcionais de forma a tornar a utilização da plataforma pela parte do utilizador mais intuitiva, apelativa e eficiente;
- O módulo de previsão refere-se à criação e implementação deste módulo na plataforma.

No que diz respeito à IOTech, esta trata-se de uma *start-up*/empresa portuguesa de investigação e desenvolvimento (I&D), esta insere-se em várias áreas como *Data Science*, Inteligência Artificial, Internet

das Coisas (*Internet of Things - IoT*) - comunicação de sistemas, integração, interoperabilidade e desenvolvimento web - e gamificação.

O ioScience resulta de uma patente da IOTech de "um modelo e arquitetura sistémica orientado para os dados, materializado por uma aplicação que inclui receber dados a partir de fontes estruturadas e/ou não estruturadas, processá-los, e apresentar os resultados analíticos ao utilizador final. Refere-se especificamente a um sistema que possibilita a realização de uma análise de dados em modo não conectado (do inglês offline) com a possibilidade de conexão a um módulo de Inteligência Artificial e respetivo método." (Filipe Portela & Gisela Fernandes, 2022). Sendo a patente do ioScience a base para este projeto de dissertação, todo o desenvolvimento deste, desde a seleção das tecnologias, vai ao encontro com o que é identificado na patente.

A motivação para este projeto de dissertação baseia-se no interesse desenvolvido na área de *Data Science* ao longo do percurso académico e na oportunidade de aumentar conhecimentos tanto na parte analítica, como preditiva desta área.

Por fim, é de acrescentar que os resultados deste projeto foram testados com dados reais provenientes da Indústria.

1.2 Objetivos

Este projeto de dissertação tem como questão de investigação "Qual a viabilidade de integrar um módulo preditivo no ioScience, seguindo as regras de construção deste?"

Os objetivos principais para este projeto inserem-se em:

- O1 - Otimizar o protótipo de uma solução web;
- O2 - Criar um modelo preditivo;

Os objetivos estruturantes para este projeto dividem-se em:

- O1.1 - Testar e documentar o protótipo;
- O1.2 - Melhorar o processo de análise de dados em tempo-real;
- O1.3 - Adaptar a solução para diferentes projetos;
- O2.1 - Explorar algoritmos inteligentes;
- O2.2 - Testar modelos preditivos;

- O2.3 - Implementar APIs em Python.

Para alcançar esses objetivos, e responder à questão de investigação definida, foi utilizada a metodologia de investigação *Design Science Research (DSR)*. Esta metodologia permitiu desenhar toda a solução e desenvolver um artefacto para o problema identificado, que se divide em dois momentos. No primeiro momento, foi desenvolvido o relatório intermédio, no qual, com base na patente existente, realizou-se uma revisão da literatura sobre os conceitos fundamentais (*Data Science*, *Big Data*, Análise de Dados, *Data Mining, Business Intelligence*), bem como a análise da patente do ioScience. O objetivo deste momento foi compreender o estado atual da patente e da literatura disponível. No segundo momento, além da documentação do relatório final, o projeto entrou na vertente prática desta dissertação, onde se adotou a *framework* SCRUM para o desenvolvimento e otimização da solução e o CRISP-DM para o desenvolvimento dos modelos de previsão.

1.3 Estrutura do Documento

Este documento está dividido nos seguintes capítulos:

- Introdução: aqui é apresentado o enquadramento e motivação para o projeto de dissertação, bem como os objetivos e resultados esperados;
- Revisão da Literatura: neste capítulo é apresentada a patente do ioScience, bem como o resultado da investigação dos temas relacionados e outras patentes semelhantes a esta. Este capítulo está dividido em seis secções principais: Introdução, ioScience, *Data Science*, *Big Data*, *Data Analysis* (Análise de Dados) e Patentes parecidas ao ioScience;
- Abordagem Metodológica, Materiais e Métodos: apresentação das metodologias selecionadas para este projeto (CRISP-DM, SCRUM e DSR), bem como das ferramentas que foram utilizadas para o projeto;
- Trabalho Realizado: Nesta seção, é apresentado o trabalho desenvolvido ao longo deste projeto, que engloba a atualização da arquitetura da solução, as otimizações da plataforma ioScience ao nível visual e funcional e a descrição de todo o processo de *data mining*.
- Discussão de resultados: aqui é realizada uma apresentação de todo o projeto de dissertação, com a apresentação dos todos os resultados obtidos, bem como quais objetivos do projeto estes alcançaram.

- Conclusão: aqui é realizada uma análise global de todo o documento, apresentando as conclusões referentes tanto à questão de investigação e objetivos do trabalho, quanto aos resultados alcançados durante a elaboração do projeto. Além disso, são apresentadas a tabela de riscos do projeto e indicações de possíveis questões a serem abordadas em trabalhos futuros.
- Referências: lista de referências bibliográficas utilizadas ao longo do presente documento.

Para além dos capítulos apresentados existe ainda a secção de anexos, onde encontram-se informações que não podem ser colocadas no corpo do documento e que fornecem detalhes adicionais e relevantes sobre tópicos abordados no documento, podendo se apresentar sob a forma de gráficos, tabelas, organogramas, esquemas, etc.

2. REVISÃO DE LITERATURA

Neste capítulo, são apresentados os diversos conceitos fundamentais para o desenvolvimento da dissertação, os quais contribuíram para uma melhor compreensão do trabalho. Também são abordados trabalhos relacionados ao tema atual, bem como uma exposição de estudos semelhantes.

2.1 Introdução

A revisão de literatura decorreu entre novembro de 2022 e fevereiro de 2023. Inicialmente foram definidas algumas estratégias para minimizar o desperdício de trabalho e tempo, restringindo quais os documentos que podem ser realmente relevantes para este trabalho (existindo exceções ao longo do documento). As estratégias adotadas foram as seguintes:

- Utilizar diferentes serviços de indexação, como por exemplo: Scopus, Web of Science, ScienceDirect, IEEE, ResearchGate, Google Scholar e Google Patents;
- Os documentos selecionados devem ser apresentados em inglês ou português;
- O ano de publicação dos documentos deve ser a partir de 2010, exceto em alguns casos de documentos relevantes na área de estudo;
- Os documentos devem ser de um dos seguintes tipos: artigos de revistas, artigos, livros ou outras dissertações;
- Quando selecionado um documento, deve ser lido em primeiro lugar o resumo e as conclusões para avaliar o conteúdo do mesmo.

Os temas abordados neste capítulo foram selecionados tendo em conta a patente do ioScience, onde este projeto se baseia, bem como o que seria importante para o desenvolvimento do projeto. Posto isto os conceitos selecionados são:

- *Data Science;*
- *Big Data;*
- Análise de Dados;
- *Data Mining;*
- *Bussiness Intelligence*.

2.2 ioScience

A patente do ioScience descreve um modelo e arquitetura sistémica orientado para os dados, que se apresenta como *Data Science as a Service* (DsaaS), materializando-se numa aplicação web/mobile. Esta inclui a capacidades de "receber dados a partir de fontes estruturadas e/ou não estruturadas, processá-los, e apresentar os resultados analíticos ao utilizador final." (Filipe Portela & Gisela Fernandes, 2022). Um dos pontos diferenciais deste sistema é o facto de este possibilitar a análise de dados em modo offline (não conectado) com a possibilidade de ser conectado a um módulo de Inteligência Artificial.

O facto desta solução promover o conceito DsaaS significa que esta se apresenta como um sistema global que pode ser enquadrado em qualquer contexto empresarial, sendo que esta está preparada para se ligar aos mais variados tipos de bases de dados e a tratar de questões de escalabilidade futuras.

"Toda a solução é interoperável, modular, escalável, segura, multiplataforma, e permite ter uma experiência de Ciência de Dados em modo offline." (Filipe Portela & Gisela Fernandes, 2022).

Neste sistema é possível identificar seis camadas, sendo estas as seguintes (Filipe Portela & Gisela Fernandes, 2022):

i. uma Base(s) de Dados como uma Fonte de Dados e que permite alimentar o sistema;
ii. uma Interface de Programação de Aplicações (API) que processa os dados e permite obter informações;
iii. um Armazém de Dados onde o modelo multidimensional preenchido é armazenado;
iv. uma camada de processamento analítico online OLAP (do Inglês *On-line Analytical Processing*) para consultar os dados e fornecer diferentes perspetivas sobre os mesmos;
v. uma camada memória "Cache" que gere o armazenamento das consultas; e
vi. uma camada de Visualização onde os dados ficam disponíveis para o utilizador final através de um conjunto de painéis de gestão (do inglês *dashboards*) apresentados numa aplicação.

Para uma compreensão mais detalhada das seis camadas, elas serão descritas mais minuciosamente nos pontos a seguir.

2.2.1 Fonte de Dados

A camada "Fontes de Dados" representa a(s) base(s) de dados, bem como outras fontes selecionadas de ficheiros JSON, CSV, XML, ou outros sistemas como ERPs (Planeamento de Recursos Empresariais),

SCMs (Gestão da Cadeia de Suprimentos), e CRM (Gestão de Relacionamento com o Cliente), que fornecem dados a todo o sistema. De preferência, armazena o MongoDB e algumas fontes de dados complementares (Filipe Portela & Gisela Fernandes, 2022).

2.2.2 API (Interface de Programação de Aplicações)

A camada de Interface de Programação com Transferência Representacional de Estado (do inglês "RESTful API") contém parte do trabalho essencial a esta inovação. Pois é aqui que é feito o processamento de dados, transformando os dados brutos armazenados nas fontes de dados anteriormente apresentadas em informação relevante para serem utilizados na camada OLAP (Filipe Portela & Gisela Fernandes, 2022).

2.2.3 Armazém de dados

A camada "Armazém de Dados" surge no âmbito de tratar devidamente as questões de velocidade e armazenamento. Assim, esta camada consiste em obter o modelo multidimensional preenchido na fase final da fase API e transferi-lo para a base de dados do Armazém de Dados. Isto é feito até ao final do processo ETL (Extração, Transformação e Carregamento) utilizando cursores e conectores aplicados aos tipos de bases de dados utilizados para apoiar o Armazém de Dados (Filipe Portela & Gisela Fernandes, 2022).

2.2.4 OLAP

Uma estrutura OLAP possui elementos adequados selecionados a partir de dimensões, indicadores, filtros e hierarquias. Para que isso aconteça, é definido um conjunto de cubos individuais, cada um selecionando dados de uma dimensão diferente, e depois é criado outro cubo individual, com base nas relações com os outros que apresentam os dados da tabela de factos. Preenchida uma estrutura OLAP, segue-se a fase de definir as suas funções (em inglês *drill down, roll-up, slice e dice*), e assim por diante (Filipe Portela & Gisela Fernandes, 2022).

2.2.5 Camada memória “Cache”

A camada memória "Cache" está relacionada com o suporte de um sistema de cache através de uma base de dados do browser, permitindo que as diferentes consultas solicitadas desde a camada "Visualização" até à camada "OLAP" sejam armazenadas, tornando possível manter os painéis da

aplicação preenchidos com as informações mais atualizadas, mesmo sem ligação à Internet (Filipe Portela & Gisela Fernandes, 2022).

2.2.6 Camada de Visualização

A última fase do sistema envolve a perceção do utilizador sobre o valor do trabalho. Isto torna esta fase extremamente importante, uma vez que, se o valor da solução não for percebido, a solução é uma falha, independentemente do que exista além dela. Por outras palavras, o trabalho substancial realizado antes dos painéis de instrumentos/relatórios não tem valor para o utilizador final se estes elementos visuais não apresentarem informação relevante, estruturada e organizada, nem forem atraentes para este (Filipe Portela & Gisela Fernandes, 2022).

2.3 *Data Science*

No que se refere a *Data Science*, dependendo do autor podemos observar diferentes perspetivas e definições para este tema. Algumas destas definições são, por exemplo, de Cady (2016) "*Data Science* significa fazer um trabalho analítico que, por uma razão ou outra, requer uma quantidade substancial de conhecimentos de engenharia de software.", de Provost & Fawcett (2013) "*Data Science* é um conjunto de princípios fundamentais que apoiam e orientam a extração de informação e conhecimento dos dados tendo por base princípios definidos." ou de Gibert et al. (2018) "campo multidisciplinar que combina a análise de dados com métodos de processamento de dados e conhecimentos especializados no domínio, transformando os dados em conhecimentos compreensíveis e acionáveis relevantes para uma tomada de decisão informada". No entanto, independentemente da abordagem tomada nestas definições, podemos obter o consenso que *Data Science* envolve a capacidade de analisar dados de maneira a obter informação e conhecimento.

2.3.1 Contexto do surgimento de *Data Science*

Para compreender melhor o contexto do surgimento da *Data Science*, é apresentada a Figura 1. Nela, é representado que, devido aos avanços nas tecnologias de informação e à súbita explosão de dados, entramos na era do *Big Data*. Um dos grandes impulsionadores disto foi o surgimento da *Big Data*, que com ela trouxe diversas oportunidades que colidiram na criação de um paradigma denominado *data-driven paradigm* (Mayer-Schönberger & Cukier, 2013), que está relacionado com resolver problemas de diversos ramos, sendo um deles o empresarial, bem como a capacidade de resolver estes mesmos

problemas de novas maneiras e abrir oportunidade para novas questões surgirem. No sentido de abordar estas oportunidades que a *Big Data* disponibilizou, a *Data Science* de incorporou os seguintes fatores: infra-estrutura de *Big Data*, um ciclo de vida de análise de *Big Data*, competências de gestão de dados, e disciplinas comportamentais.

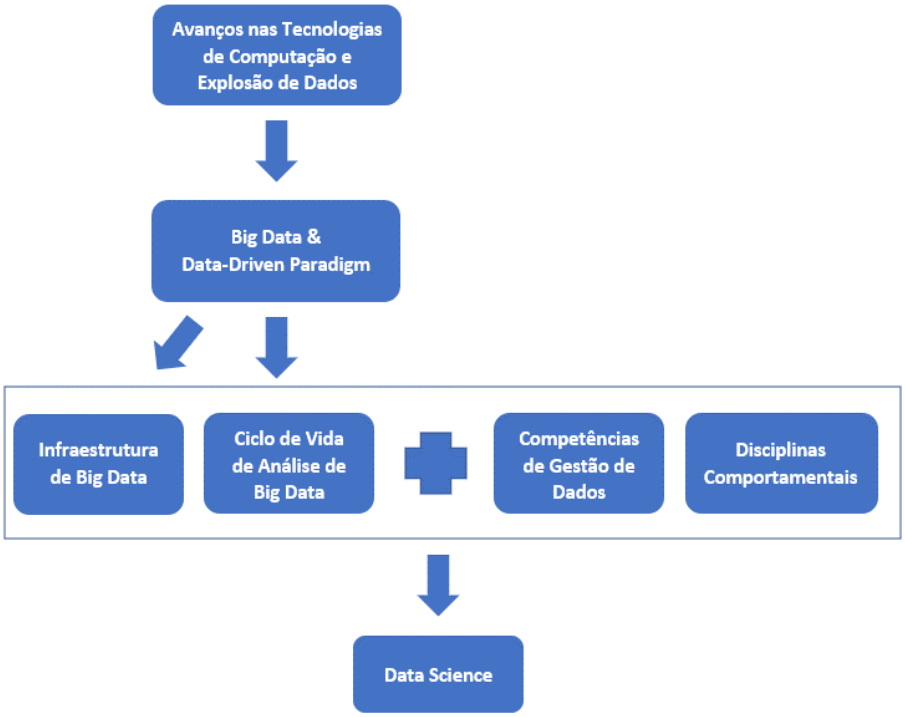


Figura 1 - Surgimento de Ciência dos dados (*Data Science*) (adaptado de Song, I. Y., & Zhu, Y., 2015).

2.3.2 Modelo do processo de *Data Science*

No que diz respeito á descrição do processo de *Data Science*, o *road map* apresentado na Figura 2 criado por Candy (2017) é um ponto de vista essencial para compreender o caminho para resolver um problema de *Data Science*.

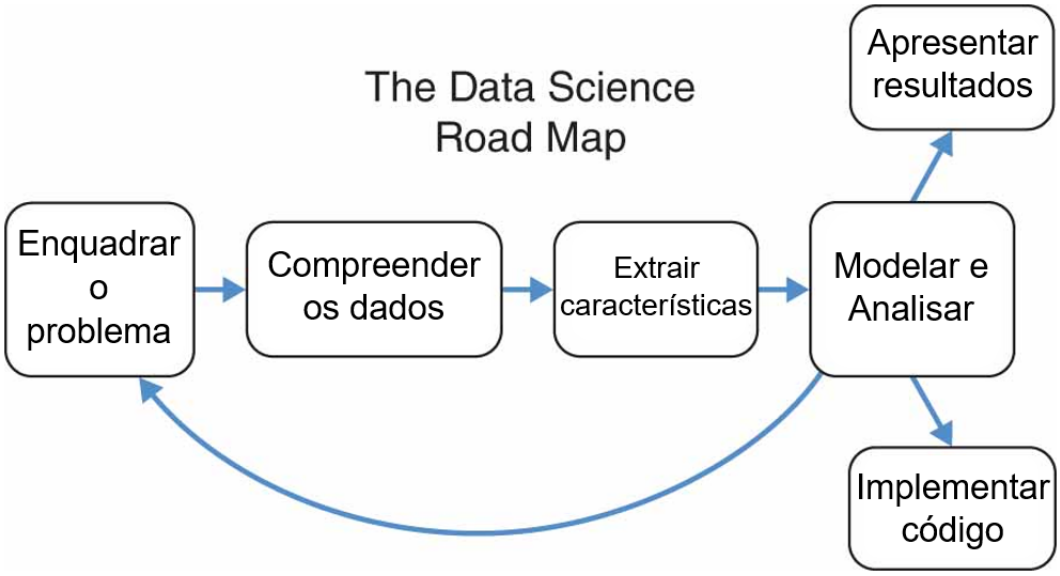


Figura 2 - O Roteiro da Ciência de Dados (The *Data Science* Road Map) (adaptado de Cady, 2017).

Por este modelo se tratar de uma ferramenta essencial para a compreensão do processo de *Data Science,* este será descrito nos pontos seguintes.

A. Enquadrar o problema

No primeiro passo deste *road map*, enquadramento do problema, é realizada uma compreensão do caso de uso do negócio para poder ser bem estabelecido o problema analítico. De seguida, é esclarecido o que exatamente constituiria uma solução para este problema, sendo importante saber fazer as perguntas certas. É fundamental estabelecer que critérios constituem um projeto como concluído e o que seria necessário para este ser considerado um sucesso. Estes critérios, quando estabelecidos num grande projeto, por norma, são apresentados num documento, como por exemplo um SOW *("statements of work")*, que tem como principal objetivo fazer com que todos os integrantes do projeto compreendam o que deve ser feito, quais são as prioridades, e quais são as expectativas realistas (Cady, 2017).

B. Compreender os dados

O segundo passo do *road map* representa a análise dos dados, onde estes são confrontados com as coisas reais que representam. Esta fase compreende três sub-fases: Questões Básicas, *Data Wrangling* e Análise Exploratória (Cady, 2017).

Questões Básicas

Uma vez que se tenha acesso aos dados, é ideal possuir uma lista de perguntas padrão para utilizar nos mesmos, sendo este um bom método para salvaguardar rapidamente possíveis problemas que se possam vir a verificar nos dados. Estas perguntas, como sugerido pelo autor, poderão ser:

- Qual é o tamanho do conjunto de dados?
- Este é o conjunto de dados completo?
- Estes dados são suficientemente representativos? Por exemplo, talvez só tenham sido recolhidos dados para um subconjunto de utilizadores.
- Há probabilidades de haver *outliers* "brutos" ou fontes extraordinárias de ruído? Por exemplo, 99% do tráfego de um servidor Web pode ser um único ataque de negação de serviço.
- Poderá haver dados artificiais inseridos no conjunto de dados? Isto acontece muito em ambientes industriais.
- Existem alguns campos que sejam identificadores únicos? Estes são os campos que poderá utilizar para a junção entre conjuntos de dados, etc.
- Os identificadores supostamente únicos são realmente únicos? O que significa se não forem?
- Se há dois conjuntos de dados A e B que precisam de ser unidos, o que significa se algo em A não corresponder a nada em B?
- Quando as entradas de dados estão em branco, de onde é que isso vem?
- Quão comuns são as entradas em branco?

No entanto, a pergunta mais importante a ser feita sobre os dados é se estes podem resolver o problema comercial em questão. Caso isto não se verifique, poderá ser necessário procurar outras fontes de dados ou alterar o trabalho planeado (Cady, 2017).

Data Wrangling

Data wrangling é o processo transformar os dados no seu formato bruto em algo adequado para uma análise mais convencional. Tipicamente, isto significa criar um *software pipeline* para realizar a limpeza ou filtragem dos dados necessária. Aqui é onde as competências de *data scientists* são mais necessárias, para lidar com *pipelines* complexas e dados confusos (Cady, 2017).

Análise Exploratória

Esta fase tem como propósito a visualização dos dados perante diversas perspetivas, experimentando diferentes formas de os transformar. Por norma, a partir desta fase são obtidos dois resultados: desenvolvimento de uma perceção intuitiva dos dados com a visualização dos padrões dos mesmos; e uma lista de hipóteses concretas sobre o que se está a passar nos dados (Cady, 2017).

C. *Extrair características*

A terceira fase deste *road map* possui muitas sobreposições com a análise exploratória e *data wrangling*, sendo que estas podem ocorrer em simultâneo. Em termos práticos, a extração de características significa obter "dados tabulados" a partir dos dados em bruto. Nestes "dados tabulados", cada linha corresponde a alguma entidade do mundo real, e cada coluna dá uma única peça de informação (geralmente um número) que descreve essa entidade. Para garantir a extração de boas características, essenciais para que a análise funcione, é necessário que os *data scientists* trabalhem em colaboração com peritos de domínio para compreender o significado destes fenómenos e obter números da melhor forma possível. Embora as características extraídas sirvam para prever algo, por vezes também será necessário extrair a variável alvo, que representa o que se está a prever (Cady, 2017).

D. *Modelar e Analisar*

Na quarta fase deste *road map*, na maioria dos projetos, são utilizados um conjunto padrão de modelos de *machine learning* para ligar os dados e ver qual deles funciona melhor. Em projetos mais particulares, existe uma necessidade de afinar cuidadosamente todos os pontos de desempenho de um modelo.

É também nesta fase que existe a possibilidade de realizar correções no projeto, permitindo obter ideias do que fazer de diferente numa nova iteração do projeto (Cady, 2017).

E. *Apresentar resultados*

Sendo uma das duas possibilidades de quinta fase deste *road map*, esta aplica-se a casos em que o cliente final é um humano, sendo possível também ser aplicada muitas vezes quando o cliente é uma máquina, e consiste em descrever num *slide deck* ou um relatório onde estão descritos o trabalho realizado e os resultados finais. Para isto, a capacidade de conseguir comunicar os conteúdos muito técnicos de *Data Science* para pessoas de diferentes áreas de especialização é um ponto fundamental nesta fase (Cady, 2017).

F. *Implementar código*

Nesta outra possibilidade de quinta fase deste *road map,* os clientes finais são computadores, sendo o trabalho dos *data scientists* produzir código para ser executado por outrem. Aqui podem-se enquadrar duas categorias: *Batch Analytics Code* (utilizado para refazer uma análise semelhante em dados futuros) ou *Real-Time Code* (tipicamente, consiste num modelo analítico). Há três *deliverables* típicos nesta fase: o código em si; alguma documentação sobre como executar o código; e uma forma de testar o código para assegurar que este funciona corretamente (Cady, 2017).

2.3.3 Pervasive *Data Science*

Após a exposição apresentada neste documento do que se trata *Data Science*, é necessário compreender o que *pervasive Data Science* envolve. Para isto, podemos analisar a definição de Davies e Clinch (2017) que define *Pervasive Data Science* como "investigação que existe na intersecção de *pervasive computing* e *Data Science* caracteriza-se por um foco sobre a recolha, análise (inferência) e utilização de dados (atuação) em busca da visão de computação ubíqua". Nesta definição é mencionado *pervasive computing* e, como o conceito ainda não foi abordado, este é apresentado a seguir, de forma a contribuir para uma melhor compreensão.

Segundo Kurkovsky (2007), "o conceito de *pervasive computing* baseia-se numa ideia simples de que, com os avanços da tecnologia, o equipamento informático ficará mais pequeno e ganhará mais potência; isto permitiria que os pequenos dispositivos fossem incorporados de forma omnipresente e invisível no ambiente humano quotidiano e, por conseguinte, proporcionaria um acesso fácil e omnipresente a um ambiente informático".

Tendo em conta estas definições, existem desafios associados tanto a *pervasive computing,* como a *Data Science* que são importantes de refletir por serem tidos em consideração quando se pensa nos desafios *de pervasive Data Science*. No que diz respeito aos principais desafios de *Data Science,* estas advêm das características originais de *big data*, os "3Vs" (Volume, Variedade e Velocidade). Os desafios de *pervasive computing*, por outro lado, incidem no desenvolvimento de arquiteturas de sistemas apropriadas, novas formas de interação com os utilizadores e preocupações transversais, como, por exemplo, facilidade de implantação e a capacidade de manutenção do sistema. Neste sentido, segundo Davies e Clinch (2017), e como podem ser observados na Tabela 1, alguns dos desafios associados com *pervasive Data Science* podem ser relacionados com:

- Recolha de dados: Proveniência e Propriedade; Privacidade e Consentimento;
- Inferência: Desafios à *Data Science* Tradicional; Novas Arquiteturas;
- Acionamento: *Pervasive Technology* para Visualização da Informação; Novas formas de Acionamento *Data-drive.*

Tabela 1 - Exemplos de desafios emergentes em ciência de dados omnipresente (*pervasive Data Science*) (adaptado de Davies & Clinch, 2017).

	Teoria	**Sistemas**	**Pessoas**
Coleção	Proveniência e Propriedade		
	Algoritmos para assinatura de dados de *streaming* que são otimizados para utilização em sensores de baixa potência.	Arquiteturas e protocolos seguros de ponta-a-ponta para fluxos de dados de alta velocidade.	Técnicas de apresentação da proveniência aos utilizadores. Modelos de propriedade de dados.
	Privacidade e Consentimento		
	Novos algoritmos de desnaturalização de dados.	Arquiteturas de sistema que abrangem a privacidade por conceção.	Novas técnicas de IU para obter o consentimento informado.
Inferência	Desafios à *Data Science* Tradicional		
	Técnicas de compensação de dados altamente variáveis e de baixa qualidade. dados.	Sistemas de armazenamento para grandes volumes, alta velocidade	Ferramentas para ajudar os utilizadores a compreender as inferências retiradas dos seus dados.
	Novas Arquiteturas		
	Algoritmos otimizados para operação distribuída na borda da nuvem.	Arquiteturas inovadoras para apoiar o *off-loading* computacional.	Quais são os modelos comerciais aceitáveis para o tratamento de dados e a prestação de serviços.
Atuação	Tecnologia Pervasiva para Visualização de Informação		
	Novos algoritmos para info-viz em ecrãs distribuídos.	Arquiteturas para coordenação da exposição.	Diretrizes para a prevenção da sobrecarga de informações em ambientes difusos.
	Novas formas de Atuação Data-drive		

	Teoria	Sistemas	Pessoas
	Modelos formais de como os ambientes respondem às diferentes entradas de dados.	Protocolos para comunicação segura com atuadores de IoT.	Interfaces para o controlo da utilização dos dados pelo utilizador.

2.3.4 Relação com o ioScience

A patente do ioScience descreve um modelo e arquitetura sistémica orientada para os dados, oferecendo *Data Science* as a Service (DsaaS) por meio de uma aplicação web/mobile. Neste sentido, os conceitos de *Data Science* e, mais especificamente, *Pervasive Data Science*, são relevantes de ser analisados, uma vez que, estes são conceitos essenciais na compreensão da patente de ioScience, sendo que esta plataforma consiste inicialmente numa ferramenta de análise e compreensão de dados, onde o processo de *Data Science* é implementado como a base do sistema.

2.4 Big Data

Com os avanços de tecnologias, a quantidade de mecanismos de captação de dados tem vindo a aumentar, que por consequente leva a um aumento dos dados em grande escala em diferentes setores. Neste ambiente de grande quantidade de dados, surge então o termo "*Big Data*". Mas a que realmente se refere este conceito?

Para Chen et al. (2014), "Sob o aumento explosivo dos dados globais, o termo *big data* é utilizado principalmente para descrever enormes conjuntos de dados. Comparado com conjuntos de dados tradicionais, os grandes dados incluem tipicamente massas de dados não estruturados que necessitam de mais análises em tempo real.".

Na perspetiva de Nugent et al. (2013), "*Big Data* não é uma tecnologia única, mas uma combinação de tecnologias antigas e novas que ajuda as empresas a obter uma visão acionável. Portanto, os grandes dados são a capacidade de gerir um enorme volume de dados díspares, à velocidade certa, e dentro do prazo certo para permitir análises e reações em tempo real."

Segundo Zikopoulos et al. (2011), "o termo *Big Data* aplica-se à informação que não pode ser processada ou analisada utilizando processos ou ferramentas tradicionais.".

Para além disto, estes autores também expõem que *Big Data* pode ser definido nas características Volume, Velocidade e Variedade, como é representado na Figura 3. Sendo que Chen et al. (2014) também mencionam a existência de uma quarta caraterística, Valor.

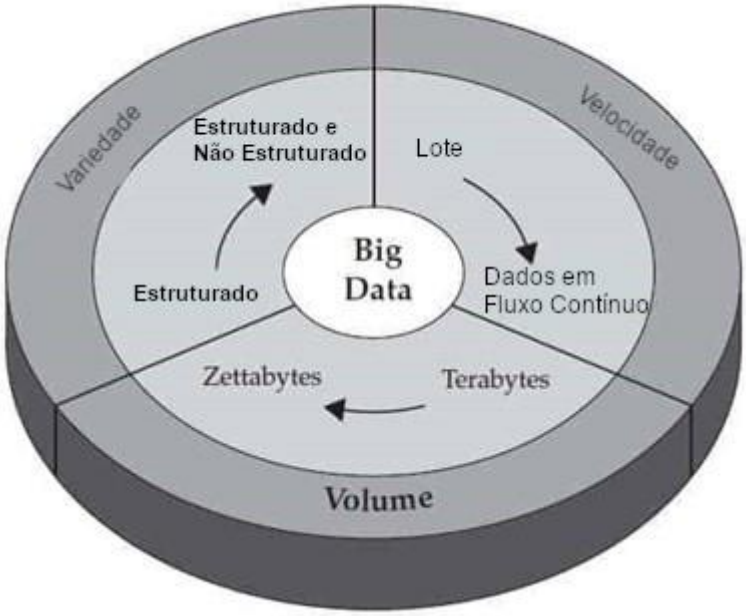


Figura 3 - A IBM caracteriza Grandes Dados *(Big Data)* (adaptado de Zikopoulos et al., 2011).

2.4.1 Características de *Big Data*

Neste ponto, serão descritas as características do modelo dos 3V's (Volume, Velocidade e Variedade) e a característica apresentada por Chen et al. (2014), Valor. Para além destas características, em estudos mais recentes, podem ser encontradas diversas outras como é o caso de Veracidade, Validade e Volatilidade (Ali-ud-din Khan et al., 2014), criando assim um total de 7V's. No entanto, é importante destacar que estas características variam ainda de autor para autor, não estando tão consolidadas como as que são apresentadas nesta secção.

A. Volume

A característica de volume refere-se à grande quantidade de dados que é constantemente gerada e recolhida. O aumento da quantidade de dados gerados é percebido quando observamos que os "volumes de dados mudou de *terabytes* para *petabytes* com uma inevitável mudança para *zettabytes*, e todos estes dados não podem ser armazenados nos seus sistemas tradicionais" (Zikopoulos et al., 2011).

B. Velocidade

Uma das compreensões para esta característica é a da rapidez em que os dados são obtidos e armazenados. Por outro lado, podemos considerar que a velocidade significa que os procedimentos de tratamento dos dados devem ser realizados rapidamente e em tempo útil, de forma a assegurar que a velocidade que os dados estão a fluir representará o máximo valor comercial possível.

C. Variedade

A variedade diz respeito aos diversos tipos de dados (dados relacionais tradicionais, dados brutos, semiestruturados e não estruturados), que com o surgimento de sensores e dispositivos inteligentes, passaram a ser possíveis o armazenamento de, por exemplo, áudios, vídeos, páginas de web e texto. Esta variedade foi o que levou a que sistemas tradicionais começassem a ter "dificuldade em armazenar e realizar as análises necessárias para obter a compreensão do conteúdo destes registos" (Zikopoulos et al., 2011).

D. Valor

Esta característica, segundo Chen et al. (2014), diz respeito ao "problema mais crítico dos grandes dados, que é como descobrir valores de conjuntos de dados com uma escala enorme, vários tipos, e geração rápida."

2.4.2 Tipos de *Big Data*

No que diz respeito ao Big Data, existem diferentes tipos de dados que o compõem, sendo os dois principais: dados estruturados e não estruturados. Neste sentido, serão definidos ambos e apresentados alguns exemplos de cada um, uma vez que os dados podem ser gerados por computadores/máquinas ou pelo ser humano.

A. Dados Estruturados

Segundo Nugent et al. (2013), dados estruturados são geralmente dados que possuem um comprimento e um formato definidos, como números, datas, e grupos de palavras e *strings*. Embora estes dados sejam os mais comuns de ser utilizados, "peritos concordam que este tipo de dados representa cerca de 20 por cento dos dados que existem no mercado" (Nugent et al., 2013).

Alguns exemplos destes dados são as seguintes:

- Gerados por computadores/máquinas: Dados de sensores; dados financeiros; dados de pontos de venda; dados de registo na Web.

– Gerados pelo ser humano: Dados relacionados com jogos; dados *Click-stream*; dados de *input*.

B. Dados Não Estruturados

Segundo Nugent et al. (2013), em contraste com os dados estruturados, os dados não estruturados são dados que não uma formatação específica. No entanto, isto não quer dizer que documentos de dados não estruturados não possuem uma estrutura específica ou formatação baseada no *software* que lhes deu origem, mas que o que é interno ao documento ser verdadeiramente não estruturado. Embora os peritos apontem que 80 por cento dos dados disponíveis sejam não estruturados, até recentemente "a tecnologia não suportava realmente fazer muito com eles, exceto armazená-los ou analisá-los manualmente" (Nugent et al., 2013).

Alguns exemplos destes dados são as seguintes:

– Gerados por computadores/máquinas: Imagens de satélite; Dados científicos; Fotografias e vídeo; Dados de radar ou sonar.

– Gerados pelo ser humano: Texto interno à sua empresa; Dados dos meios de comunicação social; Dados móveis; Conteúdo do site.

2.4.3 Relação com o ioScience

O conceito de Big Data, as suas características e tipos de dados são fundamentais para compreender as diferenças entre dados estruturados e não estruturados, bem como para entender como esses tipos de dados podem ser relevantes e utilizados na patente do ioScience. O conhecimento e a análise desses conceitos são cruciais para o desenvolvimento e o uso eficaz da plataforma ioScience, que lida com dados de diferentes naturezas e fontes. Isso permite que a plataforma processe e apresente informações analíticas aos usuários finais, conforme necessário.

2.5 Análise de Dados

De acordo com Baesens (2014), devido às empresas estarem a ser inundadas por tsunamis de dados, estas cada vez mais possuem um potencial inexplorado de análise para melhor compreender, gerir, e explorar estrategicamente os dados que tem acesso. Sendo que com quantos mais dados são gerados maior é relevância e importância de estes serem analisados.

"A análise de dados é o processo de limpeza, alteração e processamento de dados em bruto e extração de informação acionável e relevante" (Kelley, 2023) tanto para as empresas a tomarem decisões

informadas, como para os indivíduos no seu dia a dia. “O procedimento ajuda a reduzir os riscos inerentes à tomada de decisões, fornecendo informações e estatísticas úteis, frequentemente apresentadas em gráficos, imagens, tabelas e gráficos.” (Kelley, 2023).

Segundo Lepenioti et al. (2020) e Bousdekis et al. (2022), “análise de dados é categorizada em três fases principais caracterizadas por diferentes níveis de dificuldade, valor e inteligência: (i) análise descritiva, respondendo às perguntas "O que aconteceu? "Porque aconteceu?", mas também "O que está a acontecer agora?". (principalmente num contexto de *streaming*); (ii) análise preditiva, respondendo às perguntas "O que acontecerá?" e "Por que acontecerá?" no futuro; (iii) análise prescritiva, respondendo às perguntas "O que devo fazer?" e "Porque deveria fazê-lo?".”

Como é possível observar na Figura 4, cada fase necessita da anterior como pré-requisito, tornando-se assim numa sequência. É ainda identificada uma fase zero, onde é realizado o pré-processamento de dados e onde os dados brutos são transformados para um formato capaz de ser processado posteriormente pelos algoritmos de análise de dados.

Figura 4 - Os níveis de inteligência de acordo com a maturidade analítica dos dados. (adaptado de Bousdekis et al., 2022)

No que diz respeito à análise descritiva, esta é a que envolve o nível mais aprofundado de investigação, mas depende fortemente do conhecimento na área. Por outro lado, a análise preditiva faz uso de dados disponíveis em maior escala, enquanto a análise prescritiva é a área menos explorada.

2.5.1 Relação com o ioScience

No que diz respeito à patente ioScience, esta baseia-se na capacidade de receber, processar e analisar dados de diversas fontes, sejam eles estruturados ou não estruturados. A plataforma ioScience utiliza técnicas de análise de dados para extrair informações significativas a partir desses dados, permitindo aos usuários finais tomar decisões informadas.

2.6 Data Mining

No que diz respeito a *Data Mining*, é difícil optar por uma definição única que forneça uma imagem completa quanto possível do fenómeno. Por esta razão, estas são algumas das definições que se podem encontrar de *Data Mining*:

- A pesquisa automática de padrões em grandes bases de dados, utilizando técnicas computacionais de estatística, aprendizagem mecânica e reconhecimento de padrões;
- A extração não trivial de informação implícita, anteriormente desconhecida e potencialmente útil dos dados;
- A ciência da extração de informação útil a partir de grandes conjuntos de dados ou bases de dados;
- A exploração e análise automática ou semiautomática de grandes quantidades de dados, a fim de descobrir padrões significativos;
- O processo de descoberta automática de informação. A identificação de padrões e relações 'ocultas' nos dados.

Exemplos de técnicas de Data Mining seriam CRISP-DM (Cross-Industry Standard Process for Data Mining) e "Knowledge Discovery in Databases" (KDD). No entanto, após uma pesquisa sobre os mesmos, foi considerado que, embora ambos sigam pontos semelhantes, o facto de CRISP-DM possuir unicamente seis fases, em comparação com o KDD que possui nove, possibilita uma utilização mais simples e clara deste.

2.6.1 Relação com o ioScience

O conceito de *Data Mining* corresponde ao processo para criação de modelos preditivos em desenvolvimento na solução do ioScience, que representa um dos principais focos do projeto de dissertação atual. *Data Mining* desempenha um papel fundamental na capacidade da plataforma de extrair informações valiosas a partir de dados brutos e é uma parte essencial da funcionalidade do ioScience para análise de dados e geração de *insights* significativos.

2.7 Business Intelligence

Segundo Scheps (2008), uma das definições para *Business Intelligence* (BI) é qualquer atividade, ferramenta, ou processo utilizado para obter a melhor informação para apoiar a processo de tomada de decisões.

No entanto a definição que Scheps (2008) considera como mais relevante no seu livro é que "*Business Intelligence* é essencialmente uma visão empresarial oportuna, precisa, de alto valor e acionável, e os processos de trabalho e tecnologias utilizadas para a sua obtenção.". Nesta definição Scheps (2008) faz referência aos "BI's Big Four" (Os Quatro Grandes do BI), isto é, as 4 principais características de BI:

- **Respostas precisas**: Para que BI tenha qualquer valor no processo de tomada de decisão, as suas respostas devem ser precisas de forma a refletir corretamente a realidade objetiva da organização, pois sem precisão, os conhecimentos que são o produto do BI podem-se tornar prejudiciais para a empresa.
- **Perceções valiosas**: o objetivo da BI não incide unicamente em produzir informação correta, mas sim em produzir informação que tenha um impacto positivo na organização, podendo este tomar diferentes formas desde redução de custos até à melhoria das operações.
- **Informação oportuna**: a informação ser oportuna é um essencial, pois qualquer informação que pode ser considerada boa pode tornar-se inútil se não for obtida no momento certo.
- **Conclusões acionáveis**: isto é referente ao facto de que, caso as conclusões retiradas do processo de BI não apresentem orientações para ações futuras, estas tornam-se inúteis. É necessário que, com base nas conclusões deste processo, seja possível escolher um caminho de ações futuras.

Os valores de BI provêm da difusão de bons hábitos na tomada de decisões. Para adquirir uma abordagem racional no processo de tomada de decisões nas empresas, pode-se utilizar um ciclo contínuo de ações baseadas em evidências. Conforme apresentado na Figura 5, é possível compreender como esse ciclo funciona.

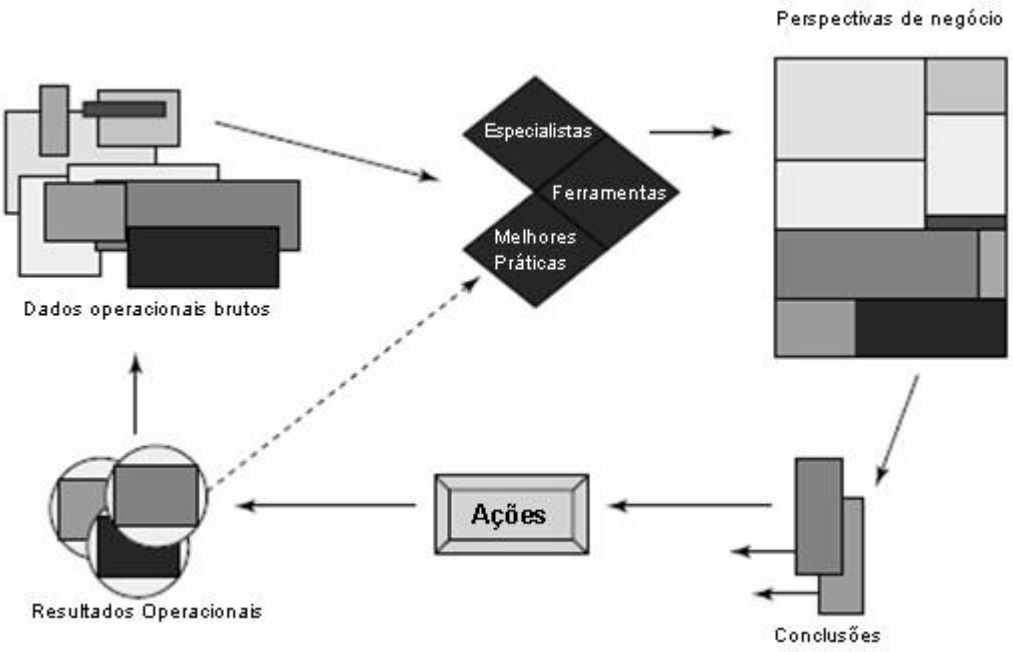


Figura 5 - Ciclo contínuo de ações baseadas em evidências (adaptado de Scheps, 2008).

O ciclo apresentado inicia-se com a utilização de conceitos e ferramentas de BI para obter perceções significativas a partir dos dados operacionais. Se essas perceções seguirem as características anteriormente mencionadas (oportuno, preciso, de alto valor e acionável), as empresas podem então aplicá-las ao seu processo de tomada de decisão regular. Essas decisões levam à escolha de ações que, se tudo correr como esperado, resultarão em melhores resultados operacionais, recomeçando o ciclo mais uma vez.

2.7.1 OLAP

OLAP (Processamento Analítico *On-Line*) é uma técnica de BI que fornece a capacidade de olhar para os dados de uma forma verdadeiramente nova, isto é, "é um software concebido para permitir aos utilizadores navegar, recuperar, e apresentar dados comerciais. Em vez de retirar dados de um sistema relacional, escrevendo consultas complexas para os recuperar, e depois inserindo-os manualmente num relatório para análise, as ferramentas OLAP cortam os passos intermédios, armazenando de facto os dados num formato pronto para relatório." (Scheps, 2008).

Segundo Scheps (2008), no núcleo do OLAP está o conceito do cubo OLAP (também chamado cubo multidimensional, ou hipercubo). Isto é, "os dados são moldados em cubos de factos uniformemente estruturados, consistindo em valores analíticos, normalmente de tipo numérico, referidos como medidas,

determinados unicamente por valores descritivos extraídos de um conjunto de dimensões." (Mansmann & Scholl, 2007).

A. Arquiteturas

Existem três arquiteturas principais de OLAP, segundo Mansmann and Scholl (2007):

- MOLAP - Processamento Analítico Multidimensional Online - é a arquitetura baseada em cubos. Esta é construída com foco na velocidade, armazena dados em estruturas lógicas construídas exclusivamente para acelerar a recuperação.
- ROLAP - Processamento Analítico Relacional On-Line - simula uma camada cúbica, inserindo uma camada semântica entre a base de dados e a ferramenta do utilizador final que imita as ações do cubo de dados. As ferramentas de acesso OLAP acedem à camada semântica como se estivessem a falar com o cubo OLAP. Isto surge para resolver os problemas que os RDBMS tinham com OLAP.
- HOLAP - Processamento Analítico Híbrido On-Line - é uma tentativa de combinar o melhor de ambos os mundos. A estrutura do cubo está no lugar para lidar com um grande número de dimensões que abrangem muitos níveis de hierarquia, oferecendo um desempenho rápido e tempos de atualização rápidos para os trabalhadores que realizam análises e criam relatórios complexos. Entretanto, os sistemas híbridos podem contar com a arquitetura ROLAP economizadora de espaço para armazenar maiores volumes de dados em bruto, canalizando apenas a informação resumida necessária para o cubo.

B. Operações OLAP

De acordo com Mansmann and Scholl (2007), o OLAP permite uma poderosa análise de dados em tempo real e disponibiliza operações de consulta especializadas para a manipulação de dados multidimensionais, apresentadas na Tabela 2.

Tabela 2 - Operações OLAP

Operação	**Descrição**
DRILL-DOWN	Aprofunda o nível de granularidade ao longo de uma dimensão
ROLL-UP	Diminui o nível de granularidade ao longo de uma dimensão.
Operação	**Descrição**
SLICE&DICE	Seleciona um *sub-cubo,* especificando condições de seleção em múltiplas dimensões no *drill path*.

RANKING	Produz as n células do topo/fundo do cubo em relação ao valor do agregado.
PIVOT	Muda a orientação dimensional da vista, por exemplo, troca colunas e linhas numa tabela pivot.
DRILL-THROUGH	Mostra as entradas de facto originais por detrás dos agregados.
DRILL-WITHIN	Desce a uma hierarquia de classificação diferente, da mesma dimensão.
DRILL ANYWHERE	Aumenta a dimensionalidade ao descer numa dimensão que ainda não se encontra na trajetória da perfuração.
DRILL-ACROSS	Junta múltiplos cubos de dados relacionados ao longo das suas dimensões partilhadas para combinar ou comparar as suas medidas.
SLICE	Reduz a dimensionalidade do conjunto de dados, filtrando uma das dimensões no *drill path* para um único valor.
DICE	Especifica os valores a serem excluídos de uma dimensão no *drill path*.
SELECT	Reduz uma dimensão no *drill path* a um conjunto de valores ou a um determinado intervalo de valores.
FILTER	Especifica as condições de seleção nas dimensões fora do *drill path*, resultando assim em valores agregados alterados.
CONDITIONAL HIGHLIGHTING	Marca os agregados que satisfazem uma condição especificada no contexto do conjunto de dados original.
PUSH	Permite especificar uma medida a partir de uma categoria arbitrária da dimensão.
PULL	É o inverso de PUSH que permite converter uma medida numa dimensão.

2.7.2 Key Performance Indicators

Key Performance Indicators (Indicadores-Chave de Desempenho) (KPIs), segundo Scheps (2008), são métricas e medidas que indicam o estado da empresa. "Os KPIs ajudam as organizações a compreender o seu desempenho em relação às suas metas e objetivos estratégicos." (Marr, 2015). Por outras palavras, KPIs fornecem a informação de desempenho que permite às empresas compreender se a estratégia adotada atualmente está adequada as suas necessidades de negócio.

De acordo com Marr (2015), existem diferentes tipos de KPIs que podem ser implementados nas empresas e que podem assumir as seguintes categorias:

- KPIs financeiros: são o tipo de KPI mais comum na maioria das empresas. Isto deve-se ao facto de que, muitas das vezes, o sucesso de uma empresa é medido através do desempenho financeiro que esta apresenta.
- KPIs de clientes: Um dos pontos mais importantes no sucesso de uma empresa baseia-se na satisfação dos seus clientes, bem como qual a quota da empresa no mercado.
- KPIs operacionais: Estes referem-se à forma como a empresa produz os seus produtos e serviços, sendo que este tem uma ligação muito clara e direta com as receitas, o lucro e o crescimento.
- KPIs de empregados: Na maioria das empresas, os empregados representam o maior custo, por esta razão faz, portanto, sentido obter dados referentes, por exemplo, à produtividade e aos despedimentos em grande número.

A principal função das KPIs é de possibilitar responder as perguntas de negócio mais críticas, que identificam o sucesso da estratégia de negócio aplicada. Essencialmente, segundo Marr (2015), os KPIs permitem medir o desempenho com precisão, o que, por sua vez, permite:

- Aprender com os resultados do passado e melhorar os resultados no futuro;
- Relatar externamente e demonstrar a conformidade;
- Focaliza o esforço e monitoriza os resultados.

Marr (2015) identifica ainda os "quatro passos principais" para ser possível tirar o máximo partido dos KPIs:

1. Usar os KPIs para verificar se a sua estratégia escolhida é válida.
2. Criar KPIs estratégicos de alto nível para o ajudar a estabelecer se está dentro ou fora do curso em relação a essa estratégia.
3. Criar um quadro de desempenho para medir todos os elementos críticos que irão resultar nessa estratégia.
4. Adicionar KPIs operacionais que lhe permitirão medir essas áreas em tempo real.

2.7.3 Relação com o ioScience

Todos os conceitos abordados em *Business Intelligence* desempenham um papel essencial na compreensão do que já foi realizado no contexto da patente ioScience e também na sua possível otimização. Isso é evidente, por exemplo, na compreensão das KPIs e na estrutura OLAP.

2.8 Patentes semelhantes ao ioScience

Dado que o projeto envolve uma patente, não foram encontradas patentes que sejam completamente idênticas a esta, com todas as mesmas características. No entanto, como demonstrado na Tabela 3, existem patentes que apresentam características consideravelmente semelhantes.

Tabela 3 - Patentes semelhantes ao ioScience.

Patente	**Descrição**
US10386827	sistema que permite a monitorização e análise de dados num sistema de controlo de processos distribuídos
US20170103103	conjunto de técnicas para solicitar e fornecer dados de instalação do processo utilizando uma consulta padronizada independente da fonte.
US20170228470	sistema e método que visa fornecer uma pluralidade de objetos de dados numa plataforma de partilha de conteúdos para a consulta ou pesquisa, utilizando um índice de alto nível e uma pluralidade de índices de baixo nível.
US20190303113	sistema que pode receber dados pessoais de um utilizador a partir de um primeiro dispositivo móvel e, com base neste, atribuir um modelo de interface à identidade do utilizador, onde neste são incluídos dados de *scorecard* a partir de um segundo dispositivo móvel, associado a um parâmetro de pontuação relacionado com um campo de entrada do modelo de interface.
US10348581	sistemas, métodos e *software* para facilitar o processamento e análise de dados com base na nuvem num ambiente de automação industrial.
US9159024B2	plataforma de inteligência preditiva em tempo real que, através de uma meta API para inteligência preditiva, visa receber de um utilizador artefactos que descrevem um domínio de um sistema de transações online para, pelo menos, uma entidade empresarial.

Patente	**Descrição**
US10425383	plataforma *online* que implementa uma *framework* analítica para segurança de DNS e facilitar a deteção do fluxo de domínio.
CN105279603B	sistema de análise de big data e um método capaz de ser configurado dinamicamente, em que o sistema compreende um módulo de gestão de armazenamento de dados em tempo real, um módulo de análise e cálculo de fluxo em tempo real, um módulo de análise *offline*, um módulo de visualização e afins.

3. ABORDAGEM METODOLÓGICA, MATERIAIS E MÉTODOS

Neste terceiro ponto do documento, são apresentadas as metodologias utilizadas na execução do projeto. Como metodologia de investigação, foi selecionada a *Design Science Research* (DSR) (Peffers et al., 2007), apresentada na secção 3.1, e do ponto de vista do desenvolvimento prático, foram utilizadas a SCRUM *Framework* (Schwaber & Sutherland, 2015), apresentada na secção 0, e CRISP-DM (Chapman et al., 2000), apresentada na secção 3.3.

Para além disto, também são apresentadas as ferramentas, na secção 3.4 e os conjuntos de dados que foram utilizados na execução do projeto, na secção 3.5.

3.1 Design Science Research

A metodologia *Design Science Research* (DSR), segundo Peffers et al. (2007), envolve quatros pontos, sendo estes: "um processo rigoroso de conceção de artefactos para resolver os problemas observados, fazer contribuições de investigação, avaliar os desenhos, e comunicar os resultados a audiências apropriadas.".

No entanto, no processo de *Design Science* (DS) desenvolvido por Peffers et al. (2007), foram incluídas seis etapas: identificação do problema e motivação, definição dos objetivos de uma solução, conceção e desenvolvimento, demonstração, avaliação e comunicação. Na Figura 6, é possível verificar as interações do processo, para além das etapas mencionadas.

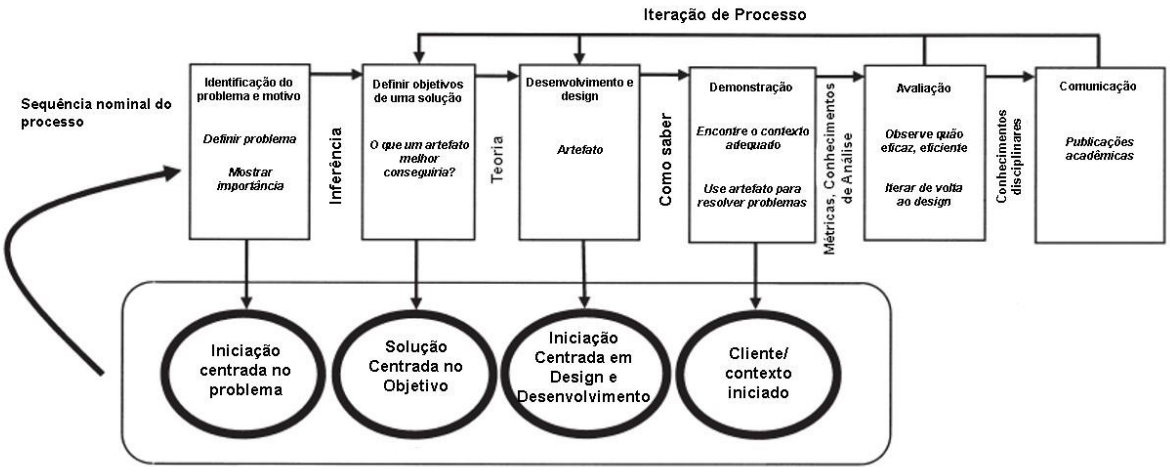


Figura 6 - Modelo do Processo DSRM (adaptado de Peffers et al. (2007))

1. **Identificação do problema e motivação**: Aqui, o objetivo é definir o problema do projeto e justificar o valor de uma solução para este. A justificação do valor de uma solução realiza duas coisas: motiva o investigador e o público da investigação a prosseguir a solução e aceitar os resultados e ajuda a compreender o raciocínio associado à compreensão do problema por parte do investigador. Para completar esta etapa, os recursos necessários incluem: conhecimento do estado do problema e da importância da sua solução. No que diz respeito à presente dissertação, esta etapa é refletida no ponto de enquadramento e motivação, estando presente neste documento na secção 1.1.

2. **Definir os objetivos de uma solução**: Nesta segunda etapa, são definidos os objetivos de da solução a partir da definição do problema e do conhecimento do que é possível e viável. Os objetivos podem ser quantitativos ou qualitativos, e são estruturados a partir da especificação do problema. Para completar esta etapa, é necessário compreender as soluções atuais para o problema, caso existam, e da sua eficácia. No que diz respeito à presente dissertação, esta etapa é refletida no ponto objetivos e resultados esperados, estando presente, neste documento, na secção 1.2.

3. **Conceção e desenvolvimento**: Após estas duas etapas, inicia-se a etapa de desenvolvimento do artefacto. Esta atividade inclui a determinação da funcionalidade desejada para o artefacto, a sua arquitetura e a sua "criação". Para completar esta etapa, é necessário obter o conhecimento teórico associado à mesma (revisão de literatura). No que diz respeito à presente dissertação, esta etapa define a conceção e desenvolvimento do protótipo modular preditivo no ioScience, estando presente neste documento na secção 4.

4. **Demonstração**: Nesta etapa é testada a funcionalidade do artefacto em algum contexto adequado, sendo que isto poderá ser feito através de experiências, simulações, casos de estudo, entre outros. Recursos necessários para a demonstração incluem o conhecimento efetivo de como utilizar o artefacto para resolver o problema. No que diz respeito à presente dissertação, esta etapa insere-se no teste e validação do funcionamento total da solução, estando presente neste documento na secção 4.

5. **Avaliação**: Esta etapa implica comparar os objetivos de uma solução com os resultados observados na utilização do artefacto na etapa anterior. No final desta atividade, existe a oportunidade dos investigadores decidirem se devem de voltar à terceira etapa para tentar melhorar a eficácia do artefacto. No que diz respeito à presente dissertação, nesta etapa é avaliado se a solução resolve o problema inicial, estando presente neste documento na secção 5.

6. **Comunicação**: Esta última etapa envolve a comunicação do problema e a importância do artefacto, a sua utilidade e novidade, o rigor da sua conceção, e a sua eficácia aos investigadores e outros públicos-alvo relevantes. No que diz respeito à presente dissertação, esta etapa é realizada através da elaboração do relatório de dissertação, da apresentação da dissertação e da escrita, revisão e publicação de dois de artigos científicos.

   No que diz respeito aos artigos desenvolvidos no decorrer desta dissertação, estes estão apresentados no Anexo II – Artigo 'Data Mining Models to predict parking lot availability (Rodrigues et al., 2023) e Anexo III – Artigo 'Pervasive Real-Time Analytical Framework—A Case Study on Car Parking Monitoring (Barros et al., 2023). O primeiro anexo, Anexo II – Artigo 'Data Mining Models to predict parking lot availability, intitula-se 'Data Mining Models to predict parking lot availability' e descreve o processo de Mineração de Dados (*Data Mining*), utilizando dados de movimentos dos parques de Lisboa. O objetivo deste é compreender como as condições meteorológicas influenciam a previsão da ocupação dos parques de estacionamento e identificar o algoritmo de previsão mais eficaz.

   Quanto ao segundo artigo, Anexo III – Artigo 'Pervasive Real-Time Analytical Framework—A Case Study on Car Parking Monitoring, este é intitulado 'Pervasive Real-Time Analytical Framework—A Case Study on Car Parking Monitoring' e refere-se ao trabalho anteriormente realizado pela Francisca Barros, que foi responsável pela elaboração da versão anterior da plataforma ioScience. No entanto, com a saída desta do projeto, houve a necessidade de continuar a rever e modificar o artigo na elaboração deste projeto, uma vez que o artigo aborda toda a estrutura OLAP utilizada na parte analítica deste projeto.

3.2 SCRUM *Framework*

A SCRUM *framework*, representada na Figura 7, é utilizada durante as tarefas três, Conceção e desenvolvimento, quatro, Demonstração, e cinco, Avaliação, do DSR e refere-se a "uma estrutura leve que ajuda as pessoas, equipas e organizações a gerar valor através de soluções adaptativas para problemas complexos" (Schwaber & Sutherland, 2020).

Segundo Schwaber and Sutherland (2020), o SCRUM utiliza uma abordagem iterativa e incremental para otimizar a previsibilidade e para controlar o risco. No que diz respeito à utilização bem-sucedida do SCRUM, esta depende das pessoas se tornarem mais proficientes em viver cinco valores: Compromisso, Foco, Abertura, Respeito e Coragem. Estes valores dão orientação à SCRUM *Team* no que diz respeito

ao seu trabalho, ações e comportamento. Os membros da SCRUM *Team* aprendem e exploram os valores enquanto trabalham com os eventos e os artefactos do SCRUM, sendo que quando estes valores são incorporados pela SCRUM *Team*, e pelas pessoas com quem esta trabalha, os pilares empíricos SCRUM da transparência, inspeção e adaptação ganham vida construindo confiança.

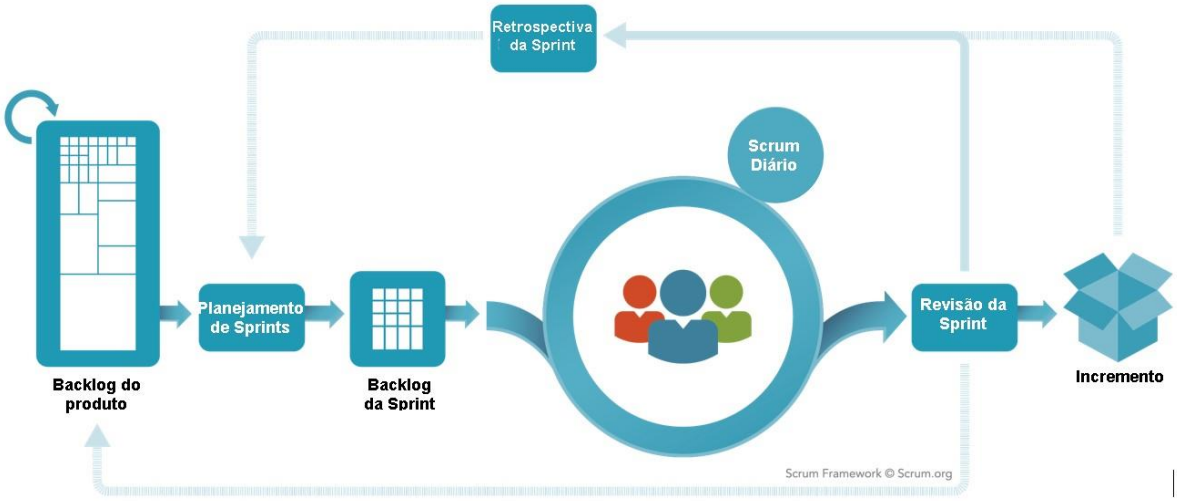


Figura 7 - SCRUM *framework* (adaptado de Schwaber & Sutherland, 2020)

SCRUM *Team* - A SCRUM *Team* é uma pequena equipa de pessoas, em que não existem subequipas ou hierarquias, e que é focada em um objetivo de cada vez, o *Product Goal*. As SCRUM *Teams* são autogeridas, o que significa que decidem internamente quem faz o quê, quando e como.

- ***Developers*** - Os *Developers* são as pessoas da SCRUM *Team* que estão empenhados em criar qualquer aspeto de um *Increment* utilizável em cada Sprint. No que diz respeito à presente dissertação, a equipa de *developers* é composta por Beatriz Rodrigues.
- ***Product Owner*** - O *Product Owner* é responsável por maximizar o valor do produto resultante do trabalho da SCRUM *Team*. No que diz respeito à presente dissertação, o *Product Owner* é a IOTech.
- **SCRUM *Master*** - O SCRUM *Master* é responsável pela implementação do SCRUM tal como definido no Guia do SCRUM. Fá-lo ajudando todos a compreender a teoria e a prática do SCRUM. No que diz respeito à presente dissertação, o SCRUM *Master* é o Professor Carlos Filipe Portela.

Eventos SCRUM - SCRUM combina quatro eventos formais (Planeamento do Sprint, SCRUM Diário, Revisão do Sprint, Retrospetiva do Sprint) para inspeção e adaptação dentro de um evento de contenção, o Sprint. Estes eventos funcionam porque implementam os pilares empíricos SCRUM de transparência, inspeção, e adaptação.

- **O Sprint** - Os Sprints são o ponto essencial do SCRUM, pois é aqui onde as ideias são transformadas em valor. São eventos de duração fixa de um mês ou menos para criar consistência, sendo que um novo Sprint começa imediatamente após a conclusão do Sprint anterior.
- **Sprint *Planning*** - O Sprint *Planning* inicia o Sprint e é aqui onde é determinando, por toda a equipa, o trabalho a ser realizado para o mesmo. O Sprint *Planning* aborda os seguintes tópicos: Porque é que este Sprint é valioso? O que se pode fazer neste Sprint? Como será feito o trabalho escolhido?
- ***Daily* SCRUM** - O objetivo da *Daily* SCRUM é inspecionar o progresso em direção ao Sprint *Goal* e adaptar o Sprint *Backlog* conforme necessário, ajustando o trabalho planeado.
- **Sprint *Review*** - O objetivo da Sprint *Review* é inspecionar o resultado do Sprint e determinar adaptações futuras. A SCRUM *Team* apresenta os resultados do seu trabalho aos principais *stakeholders* e são discutidos os progressos rumo ao *Product Goal*.
- **Sprint *Retrospective*** - O objetivo da Sprint *Retrospective* é planear formas de aumentar a qualidade e eficácia. A SCRUM *Team* inspeciona como correu o último Sprint e discute o que correu bem durante o Sprint, que problemas encontrou e como esses problemas foram (ou não) resolvidos.

Artefactos do SCRUM - Os artefactos do SCRUM representam trabalho ou valor. São concebidos para maximizar a transparência da informação chave, para assim, todos os que inspecionam possuírem a mesma base para adaptação.

- ***Product Backlog*** - O *Product Backlog* é uma lista emergente e ordenada do que é necessário para melhorar o produto. É a única fonte de trabalho levada a cabo pela SCRUM *Team*.
- **Sprint *Backlog*** - O Sprint *Backlog* é composto pelo Sprint *Goal* (porquê), o conjunto de itens do Produto *Backlog* selecionados para o Sprint (o quê), bem como um plano acionável para a entrega do *Increment* (como). O Sprint *Backlog* é um plano de e para os *Developers*., onde estes

conseguem visualizar o trabalho que planeiam realizar durante o Sprint, a fim de alcançar o Sprint *Goal*.

- ***Increment*** - Um *Increment* é um degrau concreto em direção ao *Product Goal*, isto é um *Increment* representam pequenas partes do trabalho.

3.2.1 Product Backlog

Na Tabela 4, apresentamos o *Product Backlog* relacionado a este projeto de dissertação. Conforme mencionado anteriormente, o *Product Packlog* é representado por uma lista emergente e ordenada do que é necessário para melhorar o produto, sendo que esta deve refletir os requisitos do proprietário do projeto, que, neste caso específico, é a empresa IOTech. Na Tabela 4, a coluna de "prioridade" classifica a importância dos diferentes requisitos, enquanto a coluna de "esforço" fornece uma estimativa do trabalho necessário para atender a cada requisito. Ambas são avaliadas em uma escala de 1 a 5, onde 1 é o mínimo e 5 é o máximo.

A terceira variável, denominada "realizado", indica o estado de cumprimento de cada requisito e é classificada como "sim" ou "não". É importante destacar que "não" não significa que o requisito não tenha sido implementado, mas sim que não foi completamente alcançado.

Tabela 4 - Product Backlog

ID	Requisitos	Prioridade	Esforço	Realizado
1	Compreensão do estado atual da plataforma	5	2	Sim
2	Otimização visual da plataforma	3	3	Sim
3	Otimização das funcionalidades da plataforma	4	4	Sim
4	Criação de novas funcionalidades na plataforma	5	5	Sim
5	Processo de Data Mining	5	5	Sim
6	Adaptação para o projeto ioCity	4	2	Sim
7	Consolidação da nova versão da plataforma	3	2	Sim

3.2.2 Sprint *Backlog*

Na tabela a seguir, Tabela 5, é apresentado o sprint *backlog* associado a esta dissertação. É importante notar que cada sprint deve ter um período associado, e, no caso desta dissertação, foram considerados

sprints quinzenais. Além da identificação do sprint e do período associado a ele, nesta tabela, serão também incluídos os IDs dos requisitos desenvolvidos em cada sprint.

Tabela 5 - Sprint *Backlog*

ID	SPRINTS	Início	Fim	ID Product Backlog
1	Sprint 1	27/02/2023	10/03/2023	1
2	Sprint 2	13/03/2023	24/03/2023	1
3	Sprint 3	27/03/2023	07/04/2023	2,3,4
4	Sprint 4	10/04/2023	21/04/2023	2,3,4
5	Sprint 5	24/04/2023	05/05/2023	4
6	Sprint 6	08/05/2023	19/05/2023	4
7	Sprint 7	22/05/2023	02/06/2023	4, 5
8	Sprint 8	05/06/2023	16/06/2023	4, 5, 6
9	Sprint 9	19/06/2023	30/06/2023	5, 6
10	Sprint 10	03/07/2023	14/07/2023	5
11	Sprint 11	17/07/2023	28/07/2023	5
12	Sprint 12	31/07/2023	11/08/2023	4, 5
13	Sprint 13	14/08/2023	25/08/2023	4, 5
14	Sprint 14	28/08/2023	08/09/2023	4, 5
15	Sprint 15	11/09/2023	22/09/2023	4, 7
16	Sprint 16	25/09/2023	06/10/2023	4, 7
17	Sprint 17	09/10/2023	20/10/2023	7
18	Sprint 18	23/10/2023	31/10/2023	7

3.3 CRISP-DM

Como apresentado pela IBM (2021), um dos modelos mais utilizados para *Data Mining* é o CRISP-DM, que significa Processo Padrão para a Mineração de Dados entre Indústrias (Cross-Industry Standard Process for Data Mining). Este foi utilizado durante as tarefas três, Conceção e Desenvolvimento, e quatro, Demonstração, do DSR do presente projeto, uma vez que é uma forma comprovada pela indústria para orientar os esforços de mineração de dados.

- Como metodologia, inclui descrições das fases típicas de um projeto, as tarefas envolvidas em cada fase, e uma explicação das relações entre estas tarefas.
- Como modelo de processo, o CRISP-DM fornece uma visão geral do ciclo de vida de *data mining*.

Como é apresentado na Figura 8, e segundo Chapman et al. (2000), o ciclo de vida de um projeto de *data mining* consiste nas seguintes seis fases:

- Compreensão do negócio: compreender os objetivos do projeto e os requisitos sob uma perspetiva de negócios;
- Compreensão dos dados: recolha inicial de dados e prosseguindo com atividades que permitem familiarizar-se com os dados, identificar problemas de qualidade dos dados, descobrir as primeiras perceções sobre os dados e/ou detetar subconjuntos interessantes para formular hipóteses sobre informações ocultas;
- Preparação de dados: atividades necessárias para construir o conjunto de dados final a partir dos dados brutos iniciais;
- Modelação: selecionar e aplicar as diversas técnicas de modelação;
- Avaliação: realizar a avaliação dos modelos criados utilizando as métricas definidas;
- Implementação: organizar e apresentar os resultados obtidos neste processo de forma a estes possam ser utilizados pelo utilizador.

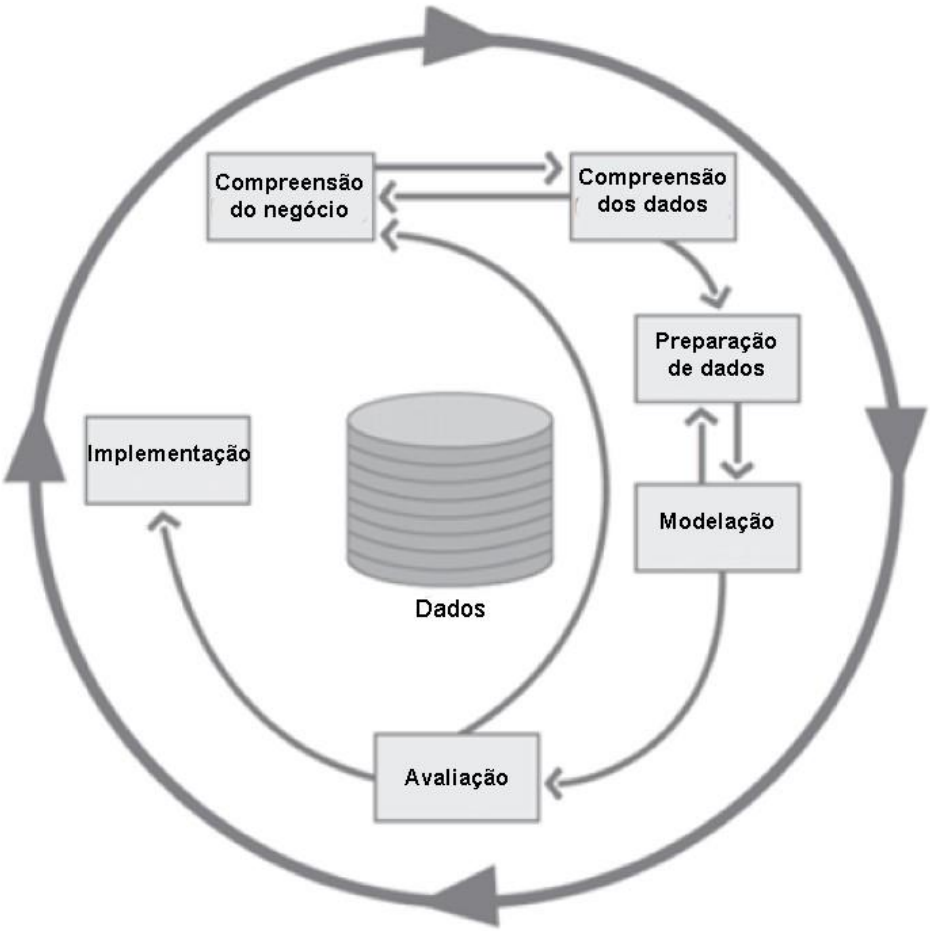


Figura 8 - Fases do modelo de referência CRISP-DM (adaptado de Chapman et al., 2000).

"A sequência das fases não é fixa. Alternar entre as diferentes fases é sempre necessário. O resultado de cada fase determina qual fase ou tarefa específica deve ser executada em seguida. As setas indicam as dependências mais significativas e comuns entre as fases." (Chapman et al., 2000).

3.4 Tecnologias e Ferramentas

Aqui são apresentadas as tecnologias utilizadas no projeto, bem como as ferramentas que lhe dão suporte. Tendo em consideração que este projeto é baseado na patente do ioScience, foi necessário ter em consideração o que é utilizada na mesma. Na Tabela 6, são apresentadas todas as tecnologias e ferramentas utilizadas, bem como a respetiva justificação de utilização de cada uma.

Tabela 6 - Tecnologias/Ferramentas utilizadas nesta dissertação

Tecnologias/Ferramentas	Justificação	Tipo
Python	Utilizada para o desenvolvimento da RESTful API dedicado ao processo de ETL.	Linguagem de Programação

Tecnologias/Ferramentas	Justificação	Tipo
JavaScript	Utilizada como linguagem de programação para desenvolver componentes da interface de web.	Linguagem de Programação
Vue.js	Desenvolver a interface web.	Framework
Pandas	Utilizada para permitir o armazenamento de dados em *dataframes* através de algumas etapas do processo ETL.	Biblioteca
Cube JS	Utilizada para desenvolver a camada OLAP.	Biblioteca
Pinia	Utilizada como uma biblioteca de armazenamento e estrutura de gerenciamento de estado para Vue.js.	Biblioteca
Scikit-learn	Utilizada como uma biblioteca de aprendizado de máquina de código aberto, onde algoritmos de aprendizagem podem ser chamados para criação de modelos.	Biblioteca
Jupyter Notebook	Utilizada no processo de análise de dados como uma extensão do python.	Aplicação web de fonte aberta
Apache ECharts	Utilizada na criação de gráficos devidos as suas componentes de visualização.	Biblioteca
imblearn	Utilizada na criação de modelos preditivos a quando a utilização da técnica SMOTE.	Biblioteca
Visual Studio Code	Utilizada como ambiente de desenvolvimento.	IDE
DBeaver	Interface para gerir a base de dados.	Interface
Docker	Utilizada para ciar ambientes onde seriam executados as diferentes etapas do projeto.	Plataforma de software
Postman	Utilizada para testar o funcionamento das APIs utilizadas neste projeto.	Plataforma de software

3.5 Dados do projeto

Durante a execução do presente do projeto de dissertação foram utilizados quatro conjuntos de dados:

- Conjunto de dados de Vila Nova de Famalicão, onde se inserem os dados dos movimentos dos parques de Vila Nova de Famalicão, que foram fornecidos e referem-se ao período de dia 01-02-2021 até dia 20-03-2021, representando 11574 registos;

- Conjunto de dados de Lisboa, onde se inserem os dados dos movimentos dos parques de Lisboa, que estão disponibilizados na página dados.gov[1], referente ao período de 01-01-2020 até 30-12-2022, originalmente representando 5.178.222 registos. No entanto, após o tratamento destes, explicado na secção 4.3.3, foram utilizados aproximadamente 23% destes;
- Conjunto de dados de Meteorológicos referentes ao período dos movimentos dos parques registados, obtidos através da API Weatherbit[2], representando 26808 registos;
- Conjunto de dados de Localização, onde se inserem os dados de localização referentes aos parques de Lisboa, obtidos através da API geoapi.pt[3], representando 26808 registos.

[1] https://dados.gov.pt/pt/datasets/ocupacao-de-parques-de-estacionamento-historico/#_
[2] https://www.weatherbit.io/api/historical-weather-api
[3] https://geoapi.pt/

4. TRABALHO REALIZADO

Este capítulo assume um papel central, uma vez que representa todo o processo de desenvolvimento guiado pela metodologia de Design Science Research (DSR). Este corresponde ao ponto três, conceção e desenvolvimento, referente à construção da nova arquitetura do projeto e todos os passos para atingir os resultados obtidos, e ao ponto quatro, demonstração, no que diz respeito à apresentação dos resultados obtidos através do teste das funcionalidades. Dividimos este capítulo em três principais subcapítulos que abordam os aspetos cruciais do projeto.

O primeiro subcapítulo apresenta o estado da arquitetura anterior à elaboração deste projeto de dissertação e o estado atual da mesma, com a inserção de um novo módulo de previsão, realizando assim conceção da aplicação.

O segundo subcapítulo abrange as otimizações realizadas na plataforma ioScience, destacando as novas funcionalidades do sistema e as melhorias implementadas ao nível da visualização e do desempenho do mesmo, apresentando o desenvolvimento e a demonstração destas funcionalidades.

O terceiro subcapítulo é dedicado à descrição do processo de *data mining* que foi seguido, com o objetivo de criar um módulo preditivo no ioScience e, consequentemente, uma API e uma *dashboard* otimizadas capazes de realizar previsões aplicadas ao âmbito da ocupação de parques de estacionamento, apresentando o desenvolvimento e a demonstração deste módulo.

4.1 Arquitetura da Solução

No que diz respeito à arquitetura da solução, esta teve de ser alterada com a adição dos resultados da subsecção 4.3.

A arquitetura da solução originalmente era representada pela Figura 9, onde existem quatro camadas de desenvolvimento: camada de desenvolvimento de dados; camada de desenvolvimento de análise; camada de desenvolvimento de cache; e camada de desenvolvimento de visualização.

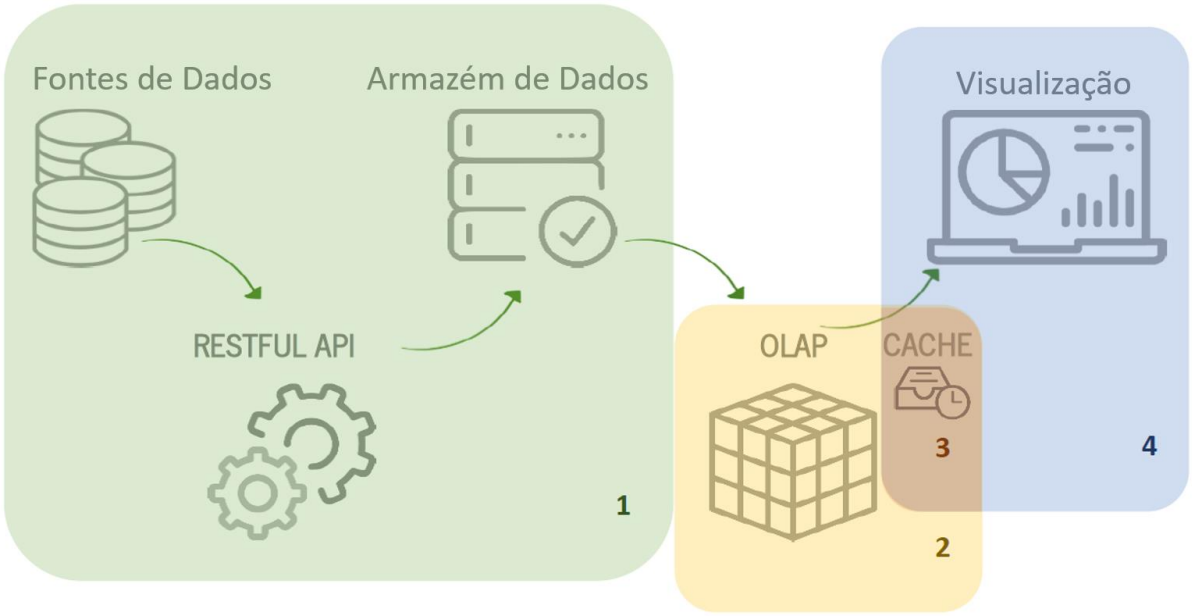


Figura 9 - Arquitetura antiga da solução.

Para uma melhor compreensão das camadas de desenvolvimento da arquitetura antes da realização deste projeto (Figura 9), uma pequena descrição de cada uma seria:

1. **Camada de Desenvolvimento de Dados** (Camada 1): aqui estão incluídos todos os mecanismos necessários para conduzir os dados desde as fontes de dados até ao armazém de dados, utilizando uma RESTful API capaz de realizar o processo ETL que tem como objetivo a melhoria da qualidade dos dados;
2. **Camada de Desenvolvimento de Análise** (Camada 2): Nesta camada estão incluídos os mecanismos subjacentes à construção da camada OLAP, representando assim esta fase da arquitetura de alto nível;
3. **Camada de Desenvolvimento de *Caching*** (Camada 3): esta camada não faz parte integrante da arquitetura de alto nível, mas pode ser considerada uma ponte entre a camada OLAP e a camada de Visualização. Esta camada intermediária abriga os mecanismos que viabilizam a existência de um 'banco de dados em cache', permitindo visualizações limitadas *offline* dos dados;
4. **Camada de Desenvolvimento da Visualização** (Camada 4): Por último, mas não menos importante, nesta camada inclui-se toda a construção das *dashboards* e sua integração na aplicação web.

Compreendendo o estado da arquitetura antes da realização deste projeto, é possível apresentar o estado atual da arquitetura onde é acrescentado a Camada de Desenvolvimento de Previsão, como apresentado na Figura 10.

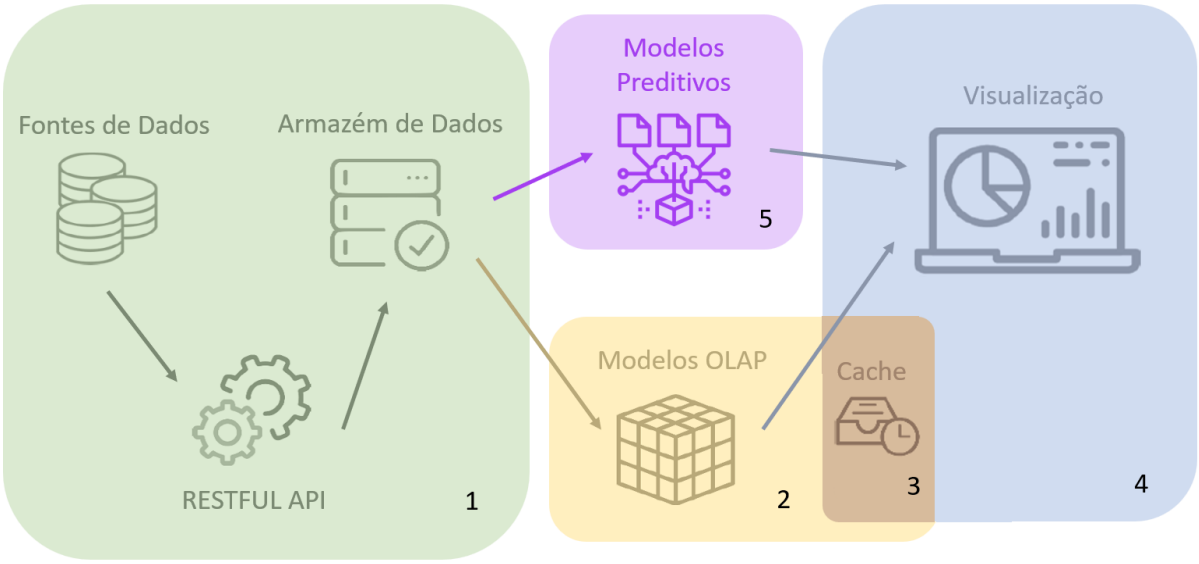


Figura 10 - Arquitetura atual da solução.

Esta nova camada identificada com o número 5 na Figura 10, consiste no resultado da secção 4.3, onde é desenvolvido um mecanismo que utiliza modelos preditivos em conjunto com uma API, que retornará a previsão referente aos dados utilizados quando se é feito um pedido com os parâmetros corretos.

4.2 Otimização da plataforma ioScience

Neste subcapítulo do trabalho realizado, serão abordadas todas alterações e otimizações do sistema bem como novas implementações de funcionalidades.

Aqui serão apresentados quatro tópicos: Otimização visual da plataforma, onde será abordada a vertente visual de comparação do estado da plataforma atual com a versão antiga; Funcionalidades da plataforma ioScience, onde serão comparadas as funcionalidades atuais com as que existiam anteriormente, bem como a explicação de novas funcionalidades; Adaptação para o projeto ioCity, onde será abordada a colaboração realizada entre a plataforma de ioScience com o projeto de ioCity desenvolvido pela mesma empresa; Reorganização do código, onde são apresentadas reorganizações *(refactoring)* do código, isto é, alterações no código que não alteram as funcionalidades implementadas no mesmo mas que tornam mais otimizada a disposição do mesmo.

4.2.1 Otimização visual da plataforma

Neste ponto, abordamos as alterações visuais feitas na plataforma atual em comparação com a versão anterior, a fim de apresentar de forma mais eficaz as otimizações listadas na Tabela 7.

Tabela 7 - Otimização visual da plataforma.

ID	**Otimização**
1	Barra do menu
2	Gráficos
3	KPIs

Para uma melhor compreensão destas otimizações, elas serão descritas a seguir, com a apresentação das alterações por meio de duas figuras. Uma figura representará a versão anterior da plataforma, e a outra representará a versão atual da plataforma.

a) Barra do menu

Uma das mudanças mais notáveis na plataforma representa a barra do menu, onde como apresentado na Figura 11, anteriormente constituía uma única barra superior com as três opções de páginas.

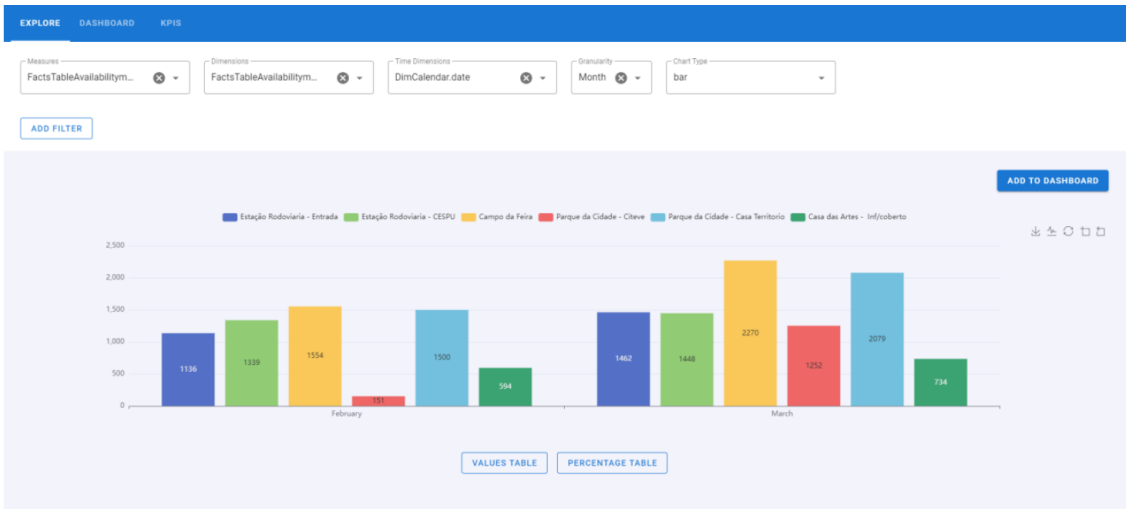


Figura 11 - Versão antiga da barra do menu.

Atualmente, como apresentado na Figura 12 , o menu é constituído por uma barra superior, onde as funcionalidades serão discutidas no ponto seguinte, e uma barra lateral colapsável, onde estão categorizadas todas as páginas que esta plataforma possui.

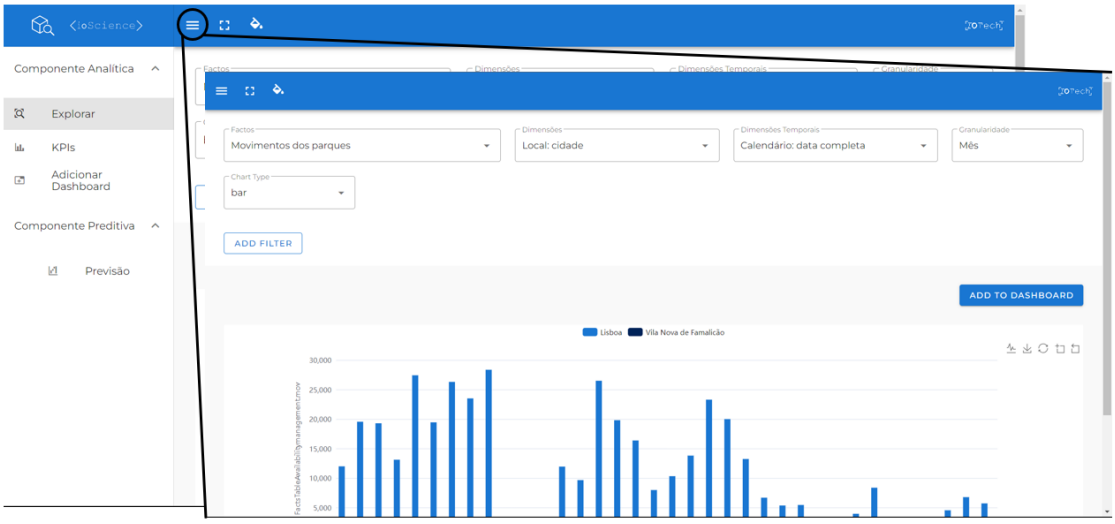


Figura 12 - Versão atual da barra do menu.

b) *Gráficos*

Os gráficos da plataforma foram otimizados ao longo deste projeto. Em algumas situações, foram feitas apenas pequenas mudanças visuais nos mesmos; no entanto, também houve casos em que foram implementados novos tipos de gráficos para enriquecer as opções de criação de gráficos.

No que diz respeito a alterações unicamente visuais, como apresentado na Figura 13, anteriormente os gráficos não seguiam as cores da plataforma e em alguns casos possuíam componentes arredondadas e noutras retas, como é o caso do gráfico de linhas.

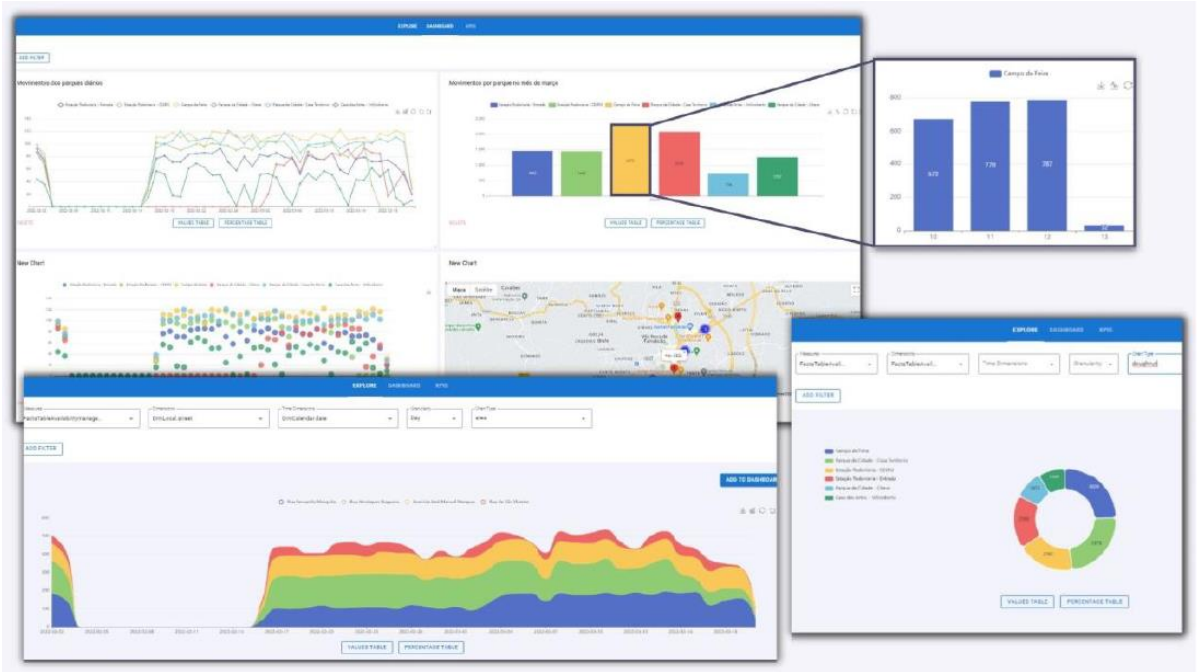


Figura 13 - Apresentação visual dos gráficos na versão antiga.

Atualmente, como apresentado na Figura 14, tanto as cores dos gráficos como as características das componentes mantêm uma apresentação uniforme.

Figura 14 - Apresentação visual dos gráficos atual.

Além disso, foram implementados os gráficos: de barra lateral, meio-donut *(halfpie)* e de gabarito *(gauge)*, conforme ilustrado na Figura 15. O gráfico de gabarito possui cores distintas, uma vez que foi criado para representar o limite de ocupação de um parque, sendo levadas em consideração as cores que melhor representariam esses limites.

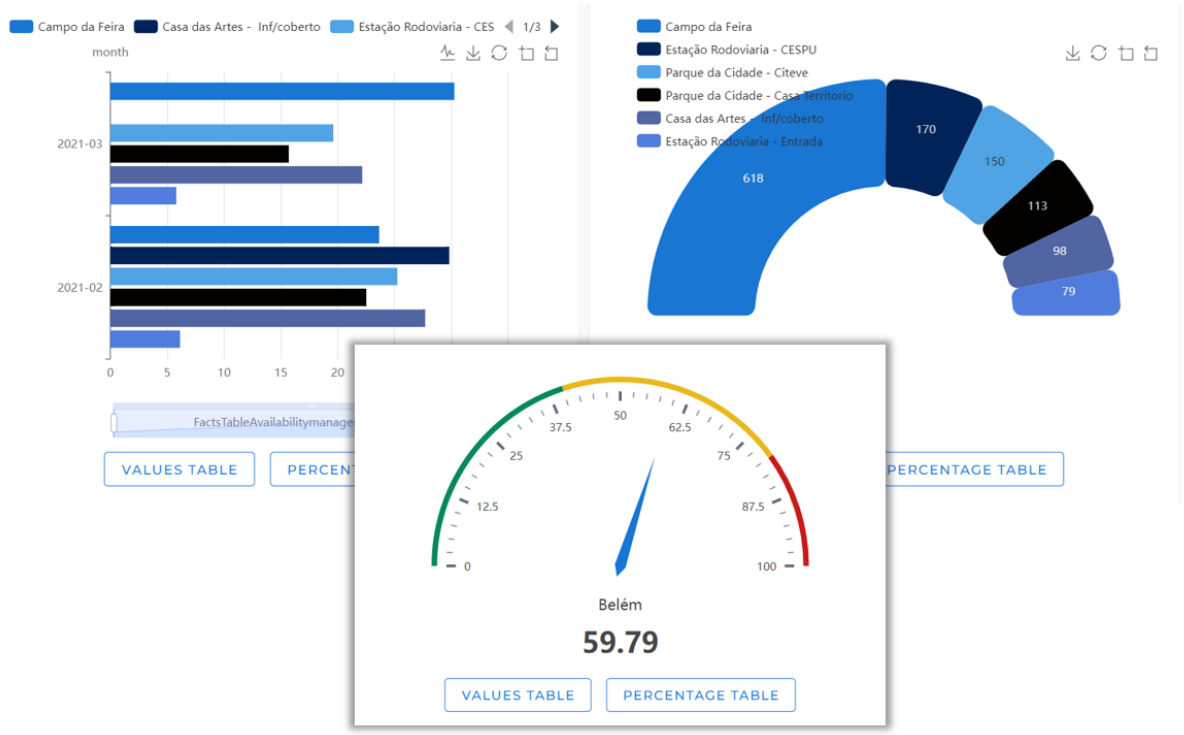


Figura 15 - Novos gráficos implementados.

c) KPIs

No que diz respeito às KPIs, foi necessária uma reformulação, tanto na escolha das KPIs relevantes a serem apresentadas, como na criação de novas maneiras mais atrativas de apresentá-las. Essa otimização tem como objetivo a construção global da página de KPIs, sendo que, para testá-la, foram utilizados os dados do projeto ioCity para criar KPIs adequadas.

A versão anterior da página de KPIs, como mostrada na Figura 16 , consistia em gráficos que ocupavam a página inteira sem divisão para permitir comparações. As KPIs na versão anterior incluíam as seguintes:

- Áreas com mais movimento;
- Parques mais movimentados;
- Dias mais movimentados;
- Parques com maior capacidade;
- Taxa de ocupação;
- Horas com maior afluência.

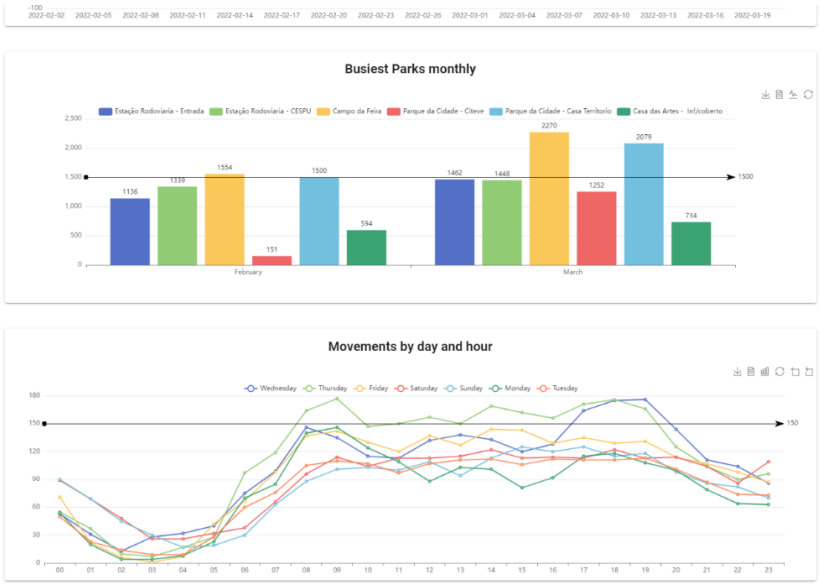


Figura 16 - Versão antiga das KPIs.

Em relação à versão atual da página de KPIs, conforme apresentada na Figura 17 , agora ela inclui, além dos gráficos, uma primeira linha de cartões que tornam a visualização dos valores numéricos imediata. Além disso, a página possui um gráfico que exibe a localização de todos os parques. A disposição dos gráficos também foi testada, e chegou-se à conclusão de que dois gráficos por linha tornam a página mais atrativa para o utilizador. Além disso, foi adicionada a opção de visualizar cada gráfico em *fullscreen* para uma análise mais detalhada. Por fim, KPIs selecionadas pela equipa são as seguintes:

- Número de parques com segurança;
- Número de parque públicos;
- Número de movimentos anuais;
- Número de parque com pagamento eletrónico;
- Taxa de ocupação por parque;
- Média de lugares disponíveis;
- Média de lugares ocupados;
- Movimentos por parque;
- Parques mais movimentados.

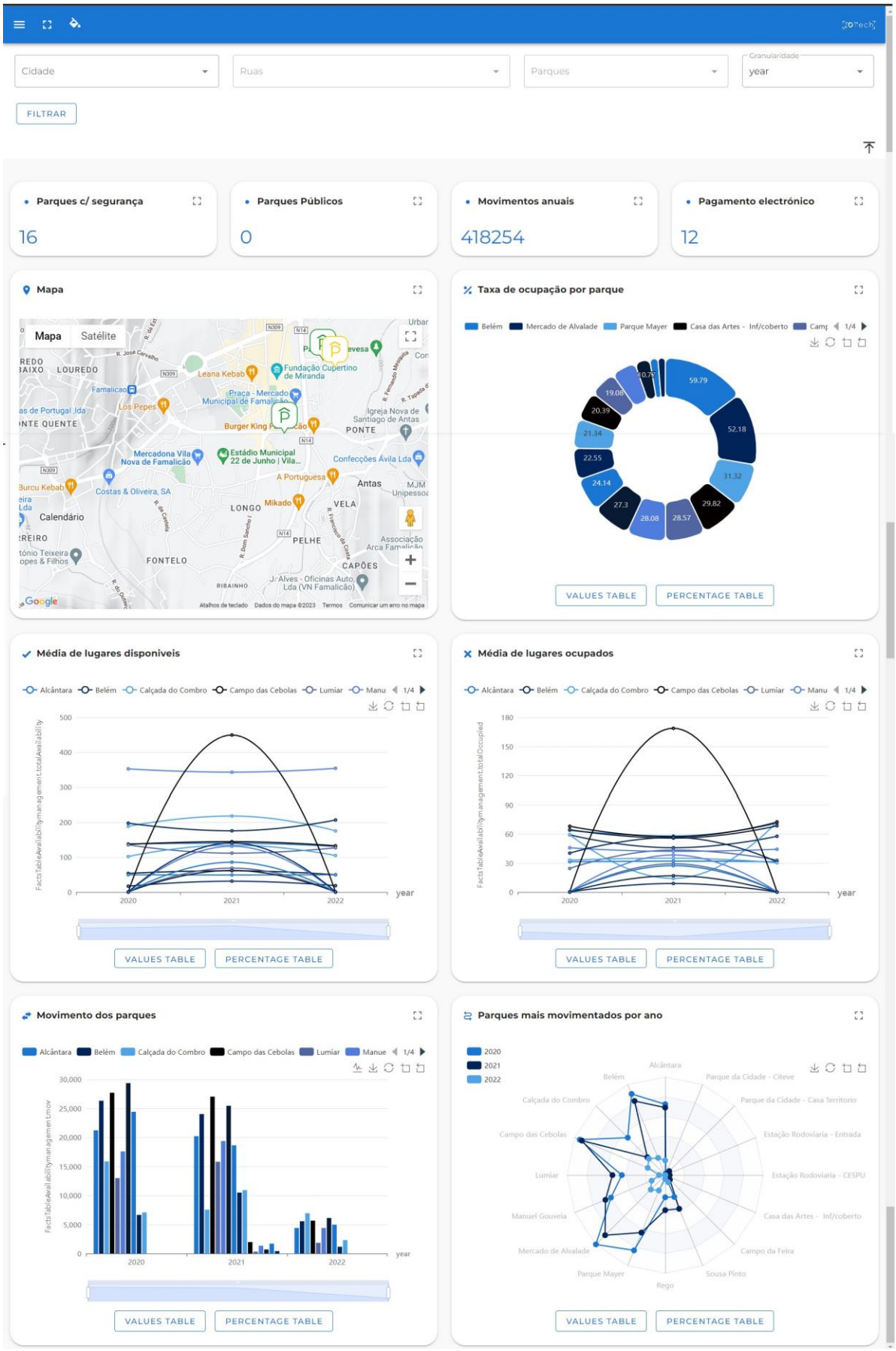


Figura 17 - Versão atual das KPIs.

4.2.2 Funcionalidades da plataforma ioScience

No que diz respeito à plataforma do ioScience, no início da elaboração deste projeto, esta já possuía uma lista diversificada de funcionalidades implementadas, portanto, o objetivo aqui foi otimizar as funcionalidades existentes e expandi-las. Para facilitar a compreensão, foi criada a Tabela 8 , que apresenta as funcionalidades atuais da plataforma e o estado delas antes e após a conclusão deste projeto de dissertação.

A Tabela 8 foi estruturada da seguinte forma: na coluna 'Antes', a funcionalidade já completamente implementada é indicada com o valor 'Sim' em verde; se a funcionalidade estava em desenvolvimento, mas não completamente estruturada, é marcada como 'Parcial' em amarelo; e se a funcionalidade é completamente nova em comparação com a versão anterior da plataforma, é marcada como 'Não'.

Tabela 8 - Estados das funcionalidades da plataforma antes e depois.

Número	Funcionalidade	Antes	Depois
1	Combinar dados	**Sim**	**Sim**
2	Adicionar filtro	**Sim**	**Sim**
3	Sugerir dados no processo de filtragem	**Sim**	**Sim**
4	Criar vários gráficos de visualização e mapa	**Sim**	**Sim**
5	Visualizar dados em dois formatos (numéricos e percentagem)	**Sim**	**Sim**
6	Guardar gráficos na *dashboard*	**Sim**	**Sim**
7	Atribuir nome aos gráficos quando estes são adicionados a *dashboard*	**Sim**	**Sim**
8	Efetuar *Drill-Down*	**Sim**	**Sim**
9	Efetuar *Rollup*	**Sim**	**Sim**
10	Adicionar várias *Dashboards*	**Parcial**	**Sim**
11	Guardar gráficos na *dashboard* selecionada	**Parcial**	**Sim**
12	Guardar layout dos gráficos após mudar a dimensão e localização nas *dashboards*	**Parcial**	**Sim**
13	Descarregar gráficos	**Sim**	**Sim**
14	Mensagens amigáveis	**Sim**	**Sim**

Número	Funcionalidade	Antes	Depois
15	Validações	**Sim**	**Sim**
16	Botão das tabelas dentro do gráfico	**Sim**	**Sim**
17	Opção de seleção de tema de cores	**Não**	**Sim**
18	Capacidade de realizar *fullscreen* da plataforma	**Parcial**	**Sim**
19	KPIs	**Sim**	**Sim**
20	Efetuar *Drill-Down* e *Rollup* nos KPIs	**Não**	**Sim**
21	Módulo analítico	**Sim**	**Sim**
22	Módulo preditivo	**Não**	**Sim**
23	Página inicial para a seleção do projeto pretendido	**Não**	**Sim**

Ao analisar a Tabela 8, podemos compreender que, antes da realização deste projeto, aproximadamente 35% das funcionalidades atuais da plataforma estavam parcialmente incompletas ou não existiam. Nota-se que existe o mesmo número de funcionalidades parcialmente incompletas e funcionalidades que não existiam antes da execução deste projeto. Atualmente 100% das funcionalidades da plataforma encontram-se totalmente completas.

A seguir, descrevemos com mais detalhes o propósito e as capacidades de cada uma das novas funcionalidades.

a) Adicionar várias Dashboards

A funcionalidade "Adicionar várias *Dashboards*" surge da ideia de o utilizador ser capaz de criar *dashboards* dependendo das duas necessidades de visualização de dados e de forma a poder aceder aos gráficos de forma direta sem estar constantemente a ter de utilizar a ferramenta explorar, onde se realiza a construção dos gráficos.

Esta funcionalidade, utilizando a biblioteca Pinia, é criada utilizando a definição de uma "*store*" (um tipo de armazenamento de estado), onde serão geridos os dados do painel de controlo das *dashboards*, utilizando a cache do sistema.

Para sermos capazes de adicionar novas *dashboards*, é necessário a realização de três passos simples, como apresentados na Figura 18:

1. Na barra principal, entrar no menu lateral e na área de Componente Analítica para ser realizado a seleção da opção “Adicionar *Dashboard*”;
2. Após isto, surgirá um *pop-up* onde será necessário inserir o nome para a nova *dashboard* e em seguida, simplesmente selecionar a opção “Criar” ("*Create*") apresentada no *pop-up* para criar a nova *dashboard*;
3. Após isto, a *dashboard* criada aparecerá no menu lateral com o nome atribuído. Ao clicar nela, poderá aceder à mesma. Além disso, se desejar eliminá-la, é possível fazê-lo através do botão com o símbolo de lixo localizado à direita do nome da *dashboard*.

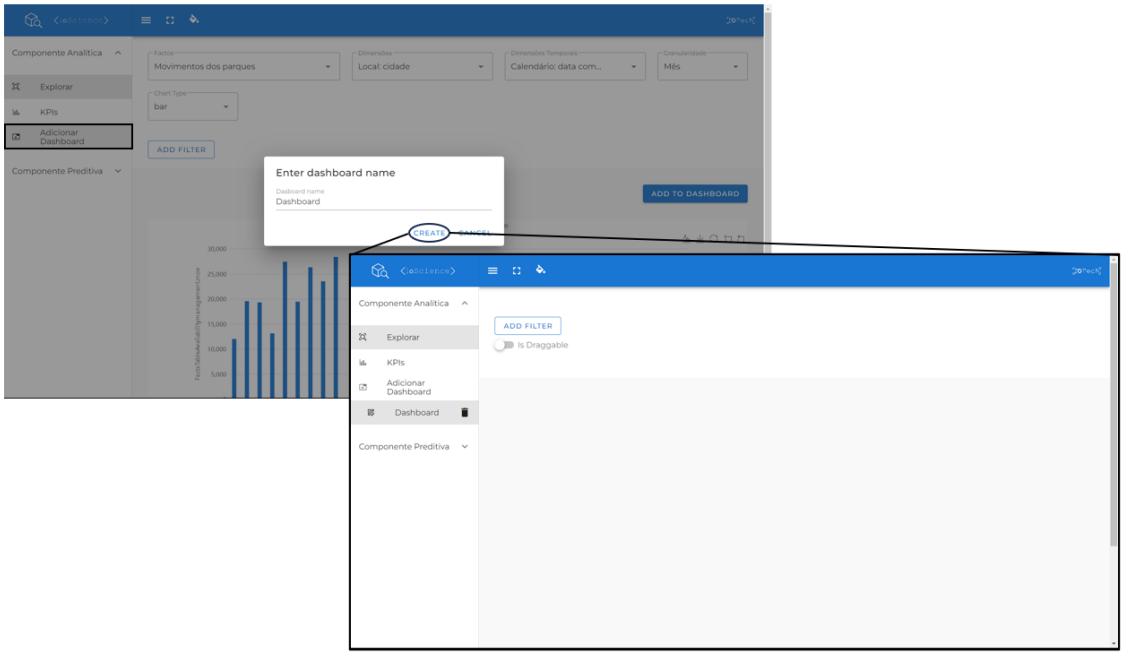


Figura 18 - Adicionar várias Dashboards.

b) *Guardar gráficos na dashboard selecionada*

A funcionalidade "Guardar gráficos na *dashboard* selecionada" segue a funcionalidade anterior, no sentido de que, quando existem várias *dashboards* nas quais os gráficos podem ser inseridos, é necessário permitir a seleção da *dashboard* na qual o gráfico deve ser guardado.

Esta funcionalidade é incorporada no programa da “*store*”, mencionado na funcionalidade anterior, com o objetivo de adicionar os dados dos gráficos, como itens à *cache* de dados de cada *dashboard*, utilizando os métodos definidos nessa “*store*”.

Para sermos capazes de guardar gráficos na *dashboard* selecionada, é necessário a realização de três passos simples, como apresentados na Figura 19:

1. Na página "Explorar", após criar o gráfico desejado, selecione a opção “Adicionar à *Dashboard*” ("*Add To Dashboard*") na lateral direita, acima do gráfico;
2. Após isto, irá aparecer um *pop up*, onde será necessário preencher o nome do gráfico e escolher em qual das *dashboards* criadas deseja inseri-lo. Em seguida, basta selecionar a opção "Guardar" apresentada no *pop-up* para adicionar o gráfico à *dashboard*;
3. Após o gráfico ser adicionado à *dashboard*, este passará a ser exibido na mesma. Se desejar removê-lo, é possível fazê-lo através do botão “Eliminar” (“*Delete*") apresentado na parte inferior do gráfico.

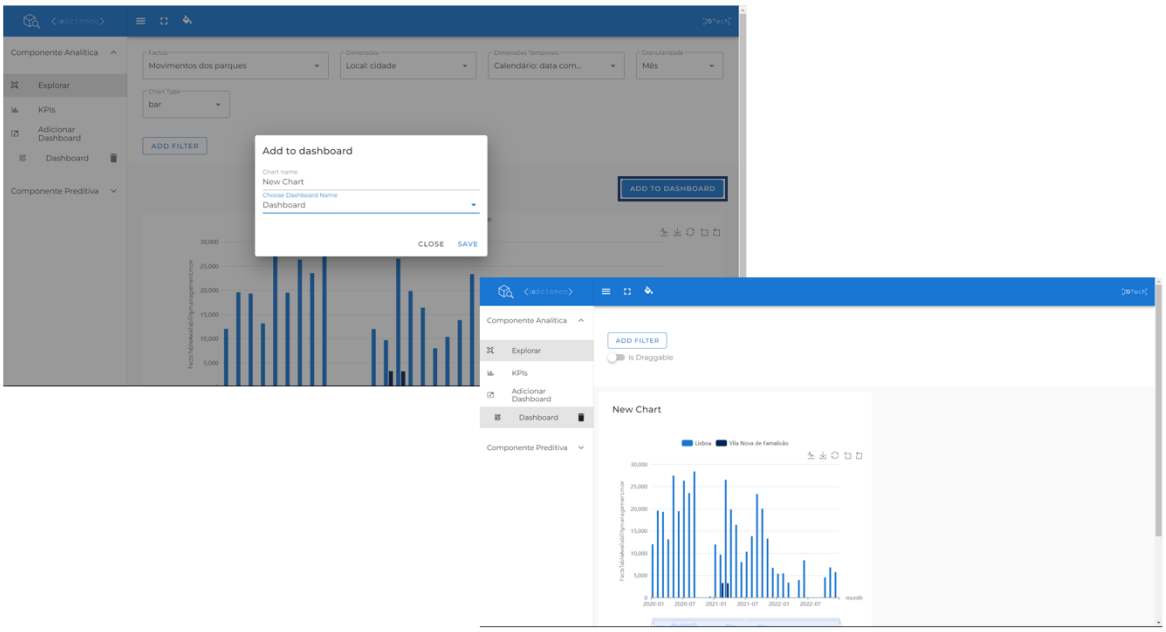


Figura 19 - Guardar gráficos na *dashboard* selecionada.

c) *Guardar layout dos gráficos após mudar a dimensão e localização nas dashboards*

A funcionalidade “Guardar *layout* dos gráficos após mudar a dimensão e localização nas *dashboards*” segue as funcionalidades anterior, no sentido de que, após a inserção do gráfico na *dashboard*, poderá ser necessário alterar as suas dimensões e posições na *dashboard*.

Esta funcionalidade, semelhante à anterior, é incorporada no programa da "*store*" e faz parte da descrição do gráfico. Esta inicialmente é definida com um valor, mas sempre que a sua localização ou tamanho são alterados, este valor é atualizado.

Para ser possível guardar o *layout* dos gráficos após mudar a dimensão e localização nas *dashboards,* é necessário a realização de quatro passos simples, como apresentados nas Figura 20 e Figura 21:

1. Na página da *dashboard* selecionada, após a inserção de gráficos na mesma, é selecionada a opção de "É arrastável" ("*Is Draggable*");

2. Após isto, como é apresentado na Figura 20, no caso de se querer alterar a dimensão do gráfico, é selecionado o canto inferior direito do gráfico e alterado as dimensões do mesmo;

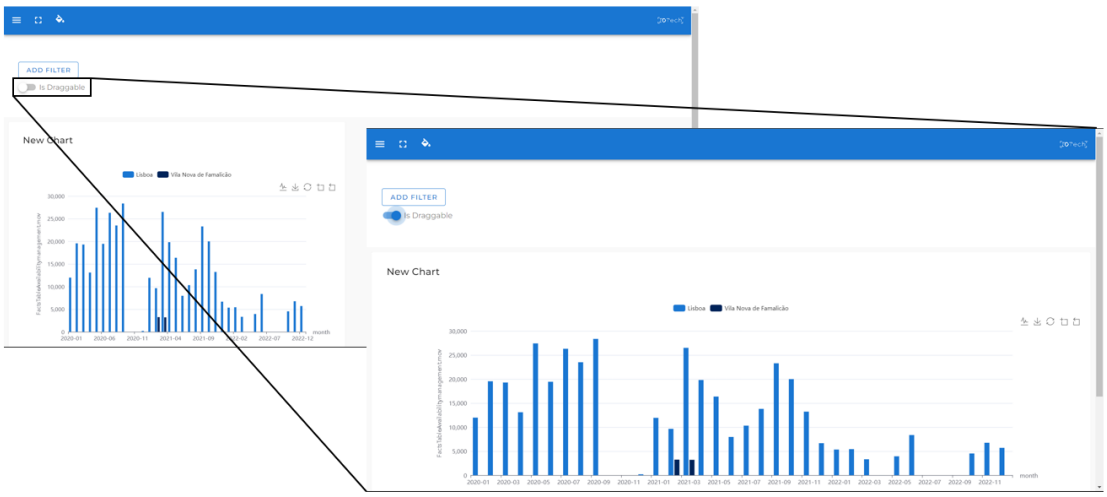


Figura 20 - Guardar layout dos gráficos após mudar a dimensão nas *dashboards*.

3. Por outro lado, como é apresentado na Figura 21, no caso de se querer alterar a localização pode-se selecionar qualquer parte da zona superior do gráfico, onde se encontra o titulo do mesmo, e alterar a localização deste na *dashboard*;

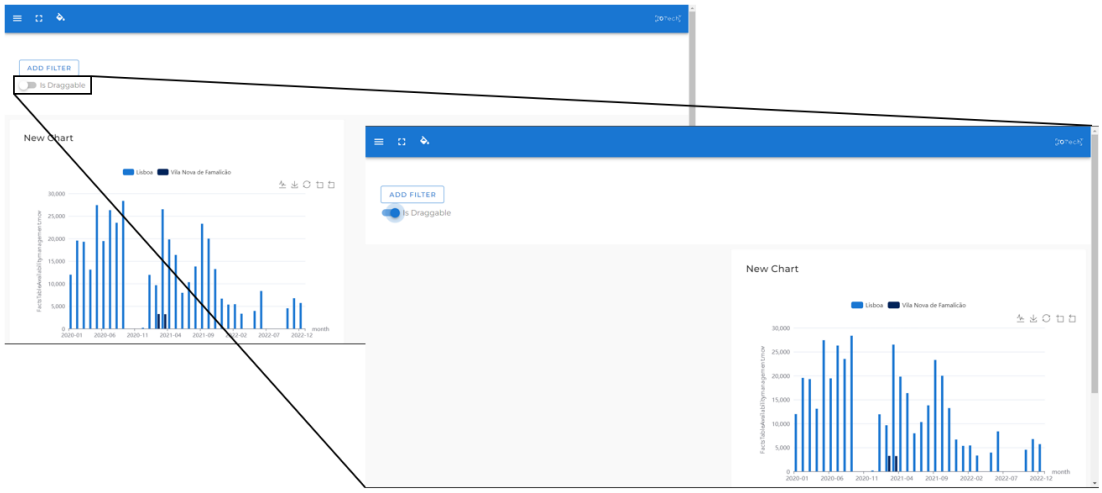


Figura 21 - Guardar *layout* dos gráficos após mudar a localização nas *dashboards*.

4. Após efetuar alterações nas dimensões e/ou localização do gráfico, é necessário voltar a selecionar a opção "É arrastável" ("*Is Draggable*") para desativar a capacidade de fazer alterações no gráfico.

d) *Opção de seleção de tema de cores*

A funcionalidade "Opção de seleção de tema de cores" surge da ideia de o utilizador ser capaz de alterar as cores da *dashboard* para a sua preferência.

Para tal, foi criada uma matriz que contém objetos, representando diferentes temas de cores. Utilizando essa matriz e a biblioteca Pinia, é definida uma "*store*" (um tipo de armazenamento de estado), na qual será gerido e armazenado qual dos temas da matriz criada que estará ativo para ser apresentado no sistema.

Para ser possível selecionar o tema de cores para o sistema, é necessário a realização de dois passos simples, como apresentados na Figura 22:

1. Na barra principal selecionar o *icon* de "Balde de tinta" e escolher o tema desejado;
2. Após isso, o tema de cores da plataforma será atualizado. No entanto, para os gráficos que já haviam sido gerados, é necessário regerá-los.

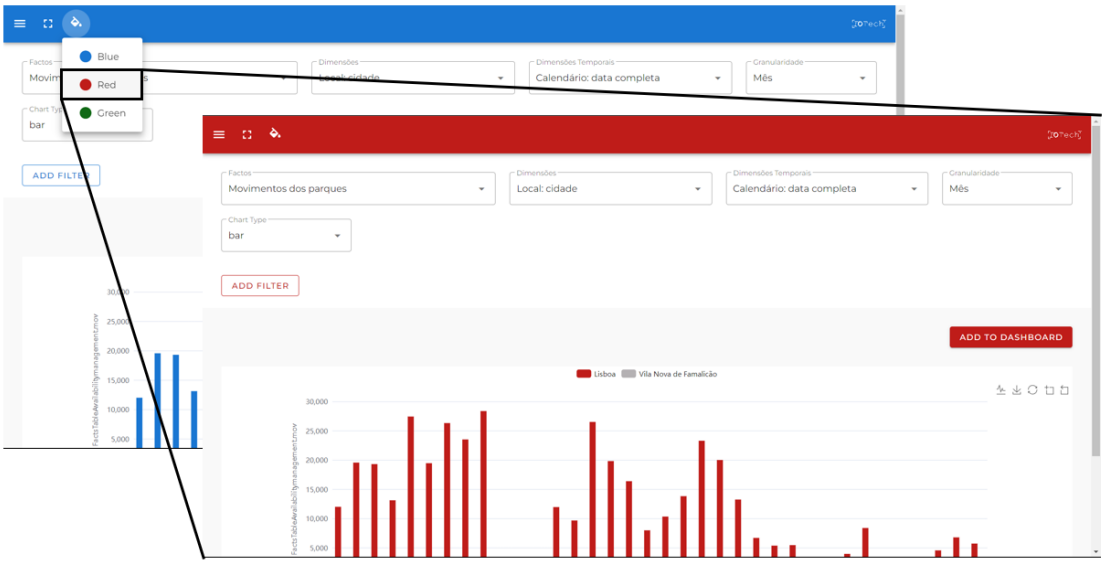


Figura 22 - Opção de seleção de tema de cores.

e) *Capacidade de realizar fullscreen da plataforma*

A funcionalidade de "Capacidade de realizar *fullscreen* da plataforma" foi concebida para permitir ao utilizador colocar a plataforma em modo de ecrã inteiro, se necessário, para uma melhor visualização dos dados. Esta funcionalidade foi implementada exclusivamente para melhorar a experiência do utilizador.

Esta funcionalidade utiliza APIs específicas de diferentes navegadores para ativar a opção de ecrã inteiro, que está associada ao botão de *fullscreen*.

Para ser possível realizar *fullscreen* da plataforma, é necessário a realização de dois passos simples, como apresentados na Figura 23:

1. Na barra principal selecionar o *icon* de "*Fullscreen*";
2. Após isto, se desejar voltar ao tamanho normal, basta carregar no mesmo botão.

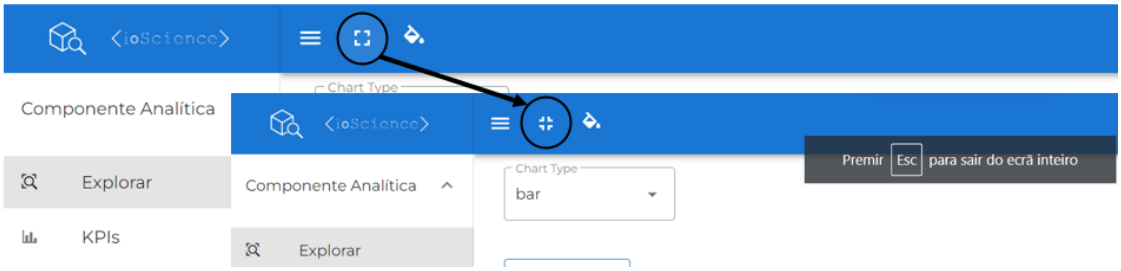


Figura 23 - Capacidade de realizar *fullscreen* da plataforma

f) Efetuar Drill-Down e Rollup nos KPIs

A funcionalidade "Efetuar *Drill-Down* e *Rollup* nos KPIs" não é considerada uma funcionalidade inteiramente nova, uma vez que já estava implementada na versão anterior da plataforma, na página das *dashboards*. Portanto, trata-se da importação de uma funcionalidade já existente para a página das KPIs.

Para ser possível efetuar *Drill-Down* e *Rollup* nos KPIs, é necessária a realização de três passos simples, como apresentados nas Figura 24 e Figura 25:

1. Quando nos encontramos na página das KPI's do projeto, é importante refletir no que cada gráfico representa e perceber que os cartões numéricos, o mapa e o gráfico circular não são capazes de *drill-drown* ou *rollup*;
2. Após isso, nos restantes gráficos, se se selecionar uma barra ou ponto onde se deseja realizar o *drill-down*, será descendido um nível, conforme mostrado na Figura 24 . Este processo pode ser repetido várias vezes, dependendo do nível em que se encontra.

Figura 24 - Efetuar *Drill-Down* nos KPIs.

3. Após realizar o *drill-down*, é apresentada a opção de efetuar o *rollup*. No primeiro botão, como mostrado na Figura 25 , é possível subir um nível. Já no segundo botão ao lado, se tiver descido mais do que um nível, ao clicar nele, todos os níveis serão revertidos ao estado original.

Figura 25 - Efetuar *Rollup* nos KPIs.

g) Módulo Preditivo

Esta funcionalidade envolveu a criação de um novo setor na plataforma, juntamente com uma nova página. Após a criação de uma API de previsão, a página é capaz de apresentar a resposta dessa API num gráfico no estilo *heatmap*.

Em outras palavras, foi desenvolvida uma página que pode enviar pedidos a uma API e apresentar a resposta de forma acessível ao utilizador.

No caso do presente projeto, a apresentação final desta página é realizada na secção 4.3.6, onde existe a aplicação desta para a previsão da ocupação dos parques de estacionamento.

h) Página inicial para a seleção do projeto pretendido

Esta última funcionalidade consiste na criação de uma página inicial para a plataforma, na qual o utilizador pode selecionar o projeto que deseja abrir dentro da sua conta, onde terá acesso unicamente aos seus dados. Após essa seleção, a plataforma carregará o conteúdo de acordo com a escolha feita pelo utilizador.

Para esta funcionalidade, em semelhança com a funcionalidade "Opção de seleção de tema de cores", foi criada uma matriz para conter objetos representando diferentes projetos, e utilizando esta matriz e a biblioteca Pinia, é definida uma "*store*" (um tipo de armazenamento de estado), onde serão geridos e armazenados qual dos projetos da matriz criada que estará ativo para ser apresentado na plataforma.

Para sermos capazes de selecionar o projeto, de modo a ser apresentado na plataforma, é necessária realização de um passo simples, como apresentados na Figura 26:

1. Na página inicial da plataforma, à direita, é apresentada a lista de projetos disponíveis. Pode-se selecionar um projeto desta lista para visualizá-lo na plataforma. Após essa seleção, o restante da plataforma ioScience é carregado com os dados do projeto escolhido.

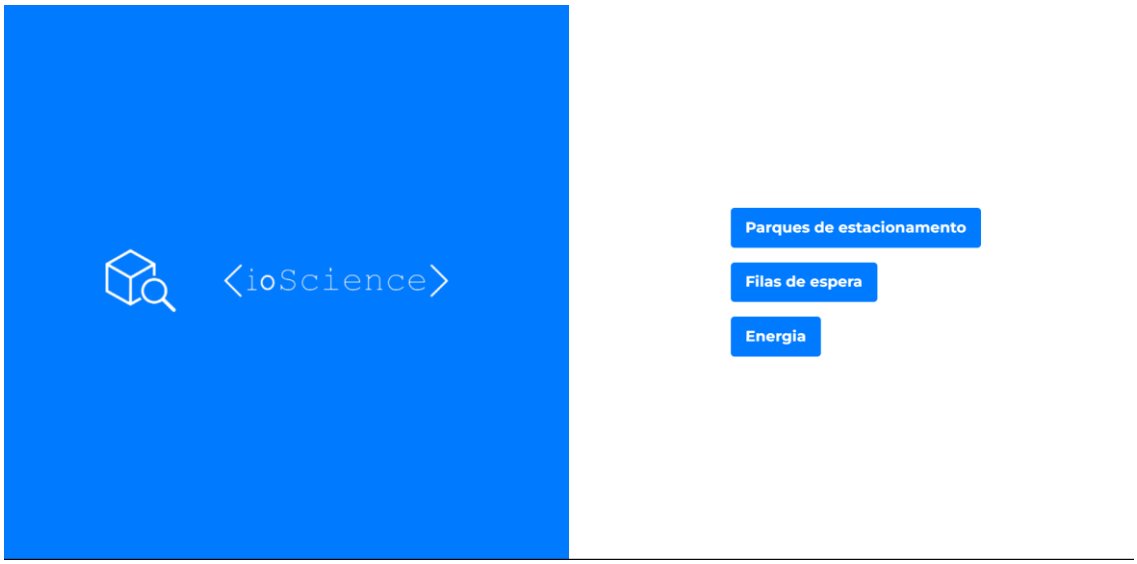


Figura 26 - Página inicial para a seleção do projeto pretendido.

4.2.3 Adaptação para o projeto ioCity

Durante a elaboração deste projeto, foi necessário criar uma adaptação da plataforma ioScience para o projeto ioCity, a qual precisava de ser adequada para ecrãs grandes utilizados em *cockpits* de gestão. Sendo que, houve a necessidade de criação de três páginas para este projeto: *Dashboard*, Estatísticas e Modelos Preditivos.

Neste projeto, a página "Estatísticas" foi criada com base em pequenas adaptações à página das KPIs no ioScience. Além disso, foi adicionada a página "*Dashboard*", que inicialmente não apresenta nenhum gráfico, como ilustrado na Figura 27.

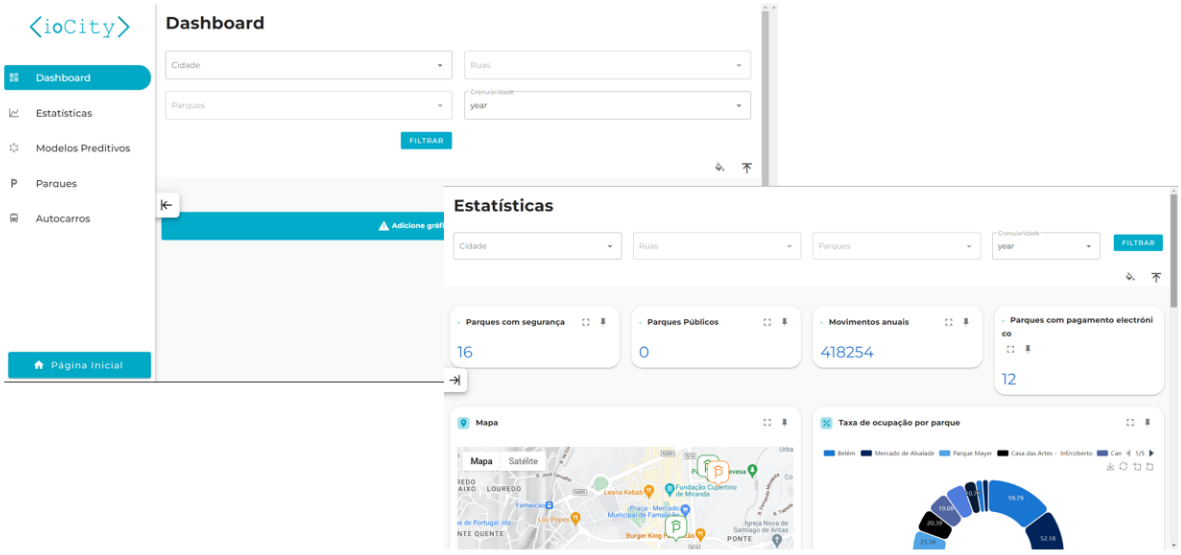


Figura 27 - Projeto ioCity estado inicial das páginas "Estatísticas" e "*Dashboard*".

No entanto, ao selecionar o ícone de "pin" num gráfico na página de "Estatísticas'', esse gráfico será exibido na página "*Dashboard*", conforme pode ser observado na Figura 28.

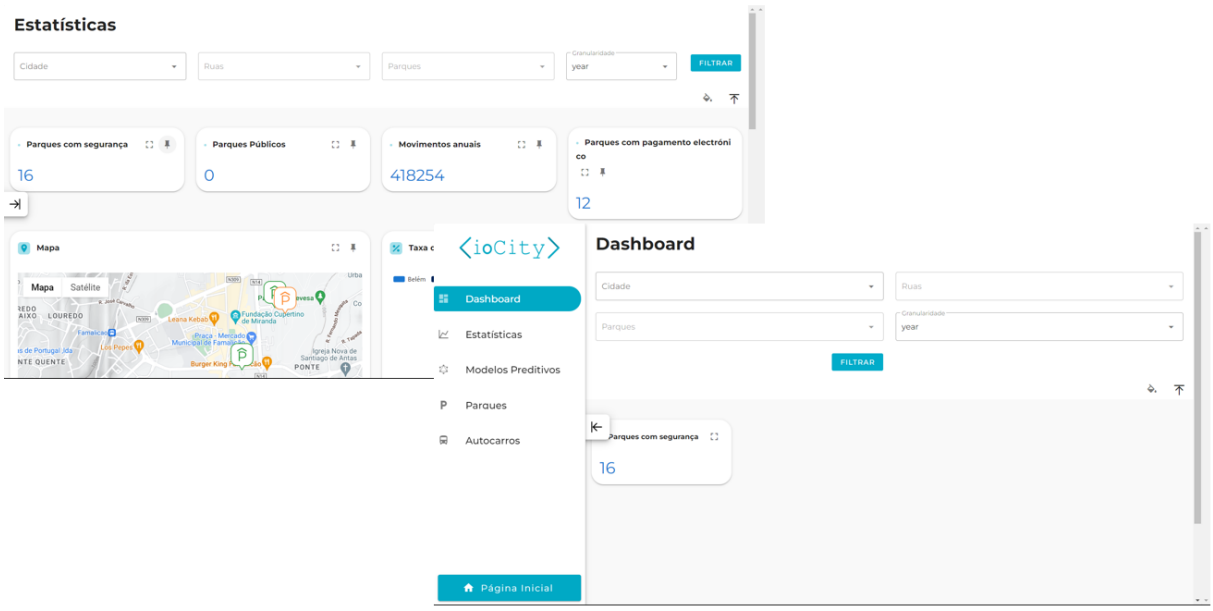


Figura 28 - Execução da funcionalidade "Pin"

Além desta funcionalidade, os gráficos mantêm todas as funcionalidades da página das KPIs, como é o caso da funcionalidade de *fullscreen*, conforme pode ser observado na Figura 29.

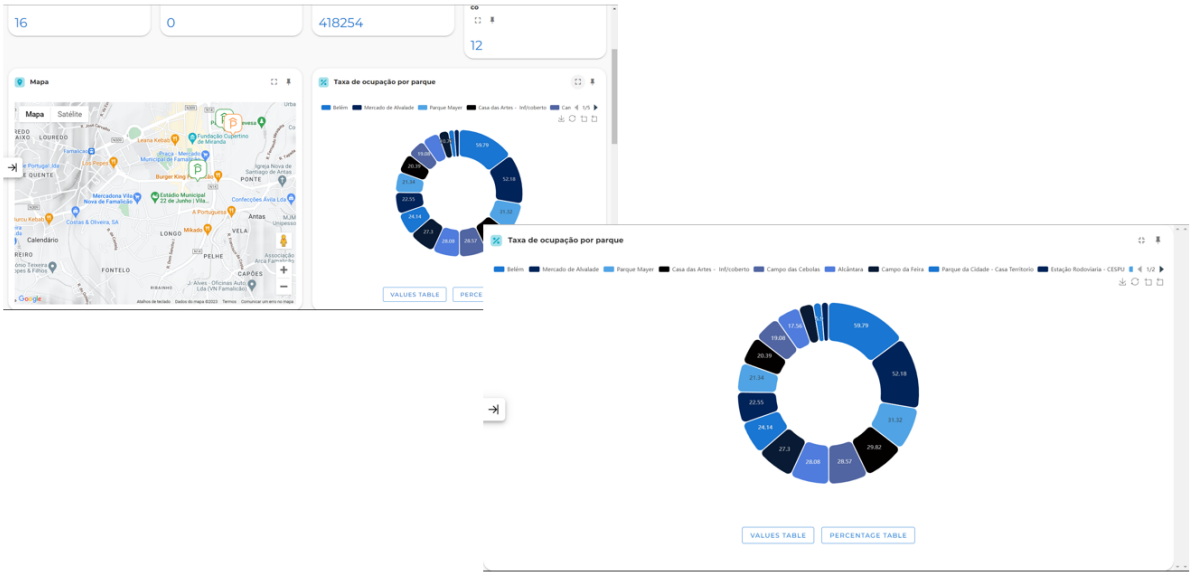


Figura 29 - Execução da funcionalidade "*Fullscreen*" do gráfico.

Por fim, no projeto ioCity, foi também implementada a página de previsão do ioScience, com a única alteração das cores para as do projeto ioCity, como pode ser observado na Figura 30.

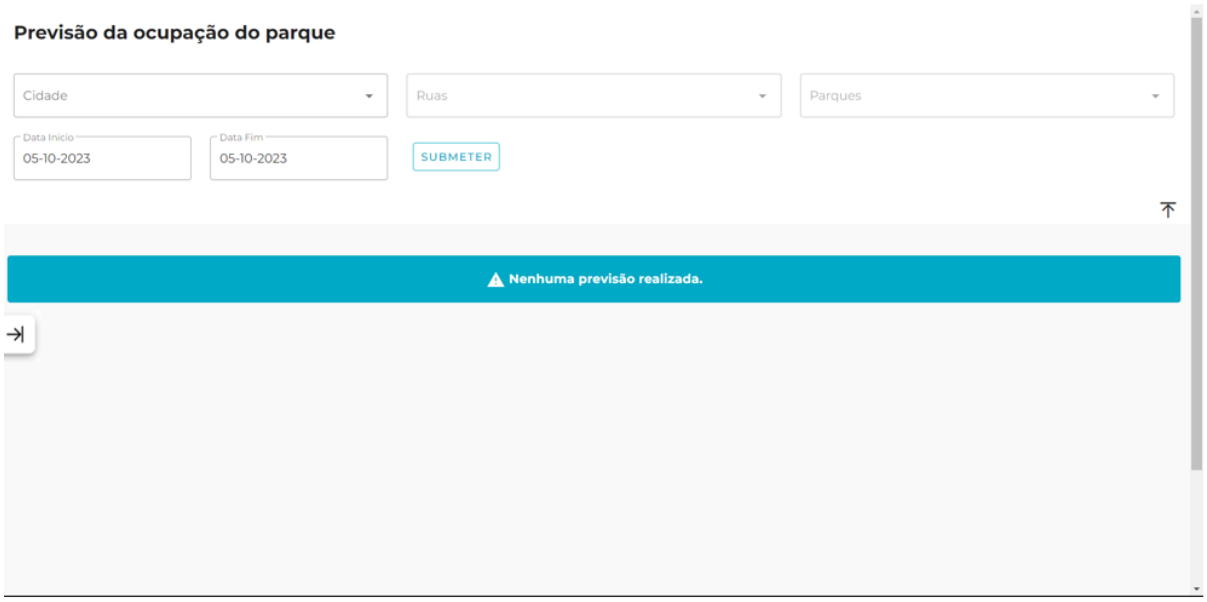


Figura 30 - Adaptação da página de previsão para o ioCity.

4.2.4 Reorganização do código

No que diz respeito ao código na plataforma do ioScience, foram realizadas algumas otimizações na forma como ele é organizado. Essa reorganização não envolve necessariamente mudanças nas funcionalidades da plataforma. As reorganizações no código estão listadas na Tabela 9.

Tabela 9 - Reorganização no código.

ID	**Reorganização no código**
1	Localização da definição das cores da plataforma
2	Localização da definição dos gráficos das KPIs
3	Página intermediária de KPIs
4	Página intermediária de Previsão

Para uma compreensão mais clara dessas reorganizações no código, cada uma delas será descrita a seguir.

a) Localização da definição das cores da plataforma

Esta reorganização derivou da implementação da funcionalidade de alteração do tema de cores, pois anteriormente a definição das cores do sistema era apresentada na página html, onde as diferentes

componentes da página eram definidas. Após esta reorganização, nas páginas html é unicamente chamada a cor necessária para aquela componente.

A definição das cores para cada tema da plataforma passou a ser realizada num arquivo JavaScript. Isto torna a expansão ou alteração de temas de cores muito mais fácil de ser realizada, dado que bastará a adição ou alteração da lista de cores do tema neste ficheiro.

b) *Localização da definição dos gráficos das KPIs*

No que diz respeito à definição dos gráficos das KPIs, antes da otimização da visualização, eles eram definidos na página HTML das KPIs. Isso tornava essa página extensa, uma vez que havia repetição das mesmas linhas de cabeçalho e chamada de *queries* do gráfico para cada KPI utilizada.

Com esta reorganização do código, no html existirá uma única instância do bloco de código para a apresentação do gráfico. No entanto, este é implementado de forma que consiga apresentar todos os gráficos.

A definição dos gráficos das KPIs da plataforma passou a ser realizada num arquivo JavaScript. Isso torna a expansão ou alteração de gráficos muito mais fácil de ser realizada, pois basta a adição ou alteração da lista de gráficos neste ficheiro.

c) *Página intermediária de KPIs*

De modo a permitir a atribuição de uma página de KPIs a cada projeto, e devido às características específicas que cada projeto possui em termos de KPIs, foi necessário criar uma página intermediária para as KPIs. Isto decorre da funcionalidade de 'Página inicial para a seleção do projeto pretendido'.

Esta página é responsável por encontrar a página atribuída a um projeto específico com base no projeto selecionado, usando o ID do projeto e o nome da página de KPIs associada a esse projeto.

d) *Página intermediária de Previsão*

À semelhança das páginas de KPIs, para ser possível a atribuição de uma página de Previsão a cada projeto, e devido às características específicas que cada projeto possui em termos da página de previsão, foi necessário criar uma página intermediária para estas. Isso decorre da funcionalidade de 'Página inicial para a seleção do projeto pretendido'.

Esta página é responsável por encontrar a página atribuída a um projeto específico com base no projeto selecionado, usando o ID do projeto e o nome da página de previsão associada a esse projeto.

4.3 Módulo Preditivo

Na presente secção do documento, é abordado o tema da mineração de dados (*data mining*), um processo crucial que desempenha um papel determinante na descoberta de padrões, relações e conhecimento útil a partir de dados brutos. Neste contexto, segue-se o modelo CRISP-DM, que é dividido em seis fases: Compreensão do negócio; Compreensão dos dados; Preparação de dados; Modelação; Avaliação; e Implementação.

O objetivo desta secção representa uma demonstração do que foi feito para testar o módulo de previsão implementado na solução.

4.3.1 Compreensão do negócio

No contexto deste projeto de dissertação, e de forma a fornecer um enquadramento sólido para o processo de *Data Mining*, o objetivo principal passa pela criação de modelos de previsão, abordando tanto algoritmos de classificação como de regressão. Em relação à classificação, o foco passou por determinar o estado do parque de estacionamento, ou seja, se o número de lugares livres representa 25% ou mais do total de lugares disponíveis, considera-se que o parque está livre, caso contrário, encontra-se ocupado. Por outro lado, os modelos de regressão têm como propósito estimar o número exato de lugares livres no parque de estacionamento.

De forma a avaliar o desempenho dos modelos, foram definidas métricas específicas, tanto para os modelos de classificação como para os de regressão. A atribuição de valor a cada uma dessas métricas teve como base o conhecimento de domínio da equipa do projeto.

No que diz respeito à classificação, e como é apresentado na Tabela 10, as métricas de interesse incluem a acuidade, sensibilidade, especificidade, precisão e o F1-Score. Estas métricas permitem avaliar quão bem o modelo classifica os estados do parque de estacionamento Assim sendo, os valores estipulados basearam-se em conhecimento adquirido pela a equipa que apoiou a elaboração desta dissertação.

Tabela 10 - Métricas de modelos de classificação em *Data Mining*.

Métrica	Descrição	Justificação	Valor
Acuidade (*Accuracy*)	Percentagem de registos classificados corretamente (positivos ou negativos).	É importante que o modelo desenvolvido consiga acertar nas suas previsões com uma taxa de erro reduzida.	>= 80 %

Métrica	Descrição	Justificação	Valor
Precisão (*Precision*)	Percentagem de registos classificados positivos que são verdadeiros positivos.	Esta métrica é importante para perceber a taxa de acertos das previsões. Uma precisão alta significa que o modelo apresenta o alto nível de confiança.	>= 70 %
Sensibilidade (*Recall or Sensitivity*)	Percentagem de registos positivos corretamente classificados como tal.	Sob uma perspetiva da análise de modelos, é a capacidade para identificar corretamente se a ocupação do parque está abaixo dos 25% de lotação.	>= 85 %
F1-Score	Percentagem da média harmónica entre a precisão e o *recall*	O F1-Score é uma abordagem simplificada para avaliar uma única métrica em vez de considerar duas (precisão e *recall*) em certos cenários. Em geral, quanto maior o F1-score, melhor.	>= 80 %
Especificidade (*Specificity*)	Percentagem de registos negativos corretamente classificados como tal.	Sob uma perspetiva da análise de modelos, é a capacidade para identificar corretamente se a ocupação do parque está acima dos 25% de lotação.	>= 80 %

No que concerne à regressão, e como é apresentado na Tabela 11, as métricas utilizadas englobam o erro quadrático médio (MSE), o R-Quadrado (R2 score), o erro absoluto médio (MAE) e o Erro Absoluto Relativo (RAE). A atribuição de um valor alvo aqui é realizada apenas para o R-Quadrado (R2 score), pois as restantes métricas possuem especialidades de cálculo e dependências dos valores que incapacitam a atribuição deste, mas é realizada a observação destas, no sentido de realizar a comparação de valores entre modelos para determinar os melhores.

Tabela 11 - Métricas de modelos de regressão em *Data Mining*.

Métrica	Descrição	Justificação	Valor
Erro Quadrático Médio (*MSE - Mean squared error*)	Média do erro das previsões ao quadrado	Diferença entre o valor previsto pelo modelo e o valor real é elevada ao quadrado. Esse processo é repetido para todos os outros pontos, os resultados são somados e depois divididos pelo	—

Métrica	Descrição	Justificação	Valor
		número de elementos previstos. Quanto maior este número, pior é o desempenho do modelo.	
R-Quadrado (*R2 score*)	Percentual da variância dos dados que é explicado pelo modelo	Quanto maior é o valor de R-Quadrado, mais explicativo é o modelo em relação aos dados previstos.	>= 85 %
Erro Absoluto Médio (*MAE - Mean absolute error*)	Média da diferença entre o valor real com o previsto.	Medida da magnitude média dos erros de previsão do modelo. Quanto menor for o valor, melhor o modelo está em fazer previsões precisas.	—
Erro Absoluto Relativo (*RAE - Relative absolute error*)	Divisão do erro absoluto total e o erro absoluto total do preditor simples.	Um bom modelo de previsão terá um rácio próximo de zero, enquanto um modelo fraco terá um rácio superior a um.	—

4.3.2 Compreensão dos dados

Nesta etapa, é realizada uma análise profunda dos dados que serão a base de todo o modelo, alicerçando a compreensão do contexto e das características dos conjuntos de dados que irão nortear o projeto. Aqui é mapeado com precisão as nuances dos nossos dados, entendendo as relações, padrões, desafios e problemas que estes possuem.

Neste projeto em particular, lidamos com quatro conjuntos de dados distintos, cada um desempenhando um papel fundamental na nossa análise. Todavia, é importante salientar que dois destes conjuntos podem ser considerados como os conjuntos de dados principais, sendo referentes a dados de Vila Nova de Famalicão e Lisboa, e que os outros dois são dados complementares relativos a dados de Localização para complementar os dados de Lisboa e a dados meteorológicos para completar ambos os conjuntos principais.

a) Cidade de Vila Nova de Famalicão

O primeiro conjunto de dados, que é apresentado na Tabela 12, é relativo aos movimentos de parques de estacionamento de Vila Nova de Famalicão. Como este já foi objeto de tratamento num projeto anterior, serviu como uma base consolidada para o processamento dos restantes dados.

Tabela 12 - Análise dos dados Vila Nova de Famalicão.

Variáveis	Descrição	Formato	Exemplo de Valores Possíveis
name	Nome do parque	String	Estação Rodoviária – Entrada
totalplaces	A capacidade total de lugares que o parque tem	Inteiro	79
totaloccupied	Total de lugares ocupados no parque	Inteiro	5
city	Cidade/concelho onde o parque se localiza	String	Vila Nova de Famalicão
county	Freguesia onde o parque se localiza	String	Crato
district	Distrito onde o parque se localiza	String	Braga
date	Data completa do registo do movimento	Date	2021-01-01
fulltime	Hora completa em que foi realizado o registo do movimento	Tempo	00:00:00
latitude	Latitude da localização do parque (Graus)	Decimal	-41.407140
longitude	Longitude da localização do parque (Graus)	Decimal	-8.515050

b) Cidade de Lisboa

O segundo conjunto de dados é composto por informações relativas aos movimentos de parques de estacionamento em Lisboa. Este conjunto de dados será o centro das atenções na próxima fase do CRISP-DM, na qual foi realizada a preparação dos dados para análise. Numa primeira tabela referente a estes dados, Tabela 13, é feita uma apresentação detalhada das diferentes variáveis deste conjunto de dados, bem como exemplos de valores para cada uma delas.

Tabela 13 - Análise dos dados Lisboa.

Variável	Descrição	Formato	Exemplo de Valores Possíveis
id_parque	O número identificador do parque	String	P040
nome	Nome do parque	String	Mercado de Alvalade
ocupacao	Número de lugares ocupados no parque quando os dados foram recolhidos	Inteiro	88
capacidade_max	A capacidade máxima de lugares do parque	Inteiro	118
position	Informação de localização do parque	String	"{'coordinates': [-9.164886, 38.761512], 'type': 'Point'}"
data_ocupacao	Hora e data da recolha dos dados	Data	31/12/2019 23:59

Numa segunda tabela referente aos dados de Lisboa, Tabela 14, é apresentada uma análise estatística (média, desvio-padrão, mínimo, quartis e máximo) das variáveis numéricas: ocupação e capacidade_max.

Tabela 14 - Análise estatística dos dados Lisboa.

variable	total	Média	std	min	25%	50%	75%	máx
ocupacao	5178222	153.68	204.06	-69	31	75	21	1772
capacidade_max	5178222	329.08	355.69	0	118	238	400	2000

Na terceira tabela, referente aos dados de Lisboa, Tabela 15, é apresentada uma análise das variáveis numéricas: id_parque, nome, position e data_ocupacao. Neste contexto, identificou-se um possível problema na variável "*position"*, devido à variação na forma de registo ao longo do período de obtenção dos dados. Esta questão é abordada na próxima fase de preparação dos dados. A suspeita surgiu devido ao maior número de valores únicos na variável "*position*" em comparação com as variáveis "id_parque'' e ''nome''. Isso sugere que um parque pode ter mais de um valor ''*position*'', o que não deveria acontecer. Além disso, é necessário tratar esta variável de forma a dividi-la em duas, representando latitude e longitude.

Tabela 15 - Análise das variáveis numéricas Lisboa.

variable	total	nº Valores Únicos	top	freq
id_parque	5178222	47	P002	123349
nome	5178222	47	Picoas Plaza	123349
position	5178222	88	{''coordinates'':[-9.128867,38.716819],''type'':''P...	327414
data_ocupacao	5178222	1162388	2022-11-03T18:24:33.000Z	72

Por fim, numa última análise dos dados deste conjunto, como é apresentado na Figura 31 , foi possível compreender que existem parques com mais de um valor de ''capacidade máxima'' associado. Uma vez que estamos a utilizar os dados de Vila Nova de Famalicão como base, tais discrepâncias nos valores não podem ocorrer.

```
# group the DataFrame by park_name and find the unique values for occupied_places for each group
grouped = df.groupby('nome')['capacidade_max'].unique().reset_index()

# create a new DataFrame with the park names and a list of unique values of occupied places
new_df = pd.DataFrame({'nome': grouped['nome'], 'unique_occupied_places': grouped['capacidade_max']})

new_df
```

✓ 0.6s

	nome	unique_occupied_places
0	Alcântara	[202, 222, 82]
1	Alto do Parque	[474]
2	Alves Redol	[166]
3	Ameixoeira	[501, 521, 71]
4	Arco Cego	[255]
5	Areeiro	[180, 200, 60]
6	Atrium Saldanha	[251]
7	Avenida Lusíada	[92, 112]
8	Belém	[76, 90, 75]
9	Berna	[354]
10	Calçada do Combro	[248, 160, 20, 89]
11	Campo Grande	[196, 208, 88]
12	Campo das Cebolas	[205, 225, 105]
13	Campo de Ourique	[0, 54]
14	Casal Vistoso	[256, 240, 55]
15	Centro Campo Pequeno	[1250]
16	Chão do Loureiro	[192, 70]
17	Cidade Universitaria	[620]

Figura 31 - Análise de valores de capacidade_max.

c) Localização

Além disso, visando enriquecer o conjunto de dados de Lisboa, recorremos à API geográfica, geoapi.pt[4] que nos forneceu um conjunto de dados geográficos relevantes para complementar os dados existentes. Na Tabela 16 são apresentadas as seguintes variáveis: lon, lat, distrito, concelho, freguesia.

Tabela 16 - Análise dos dados geográficos.

Variáveis	Descrição	Formato	Exemplo de Valores Possíveis
lon	Longitude da localização do parque (Graus)	Decimal	-9.135017
lat	Latitude da localização do parque (Graus)	Decimal	38.712416

[4] https://geoapi.pt/

Variáveis	Descrição	Formato	Exemplo de Valores Possíveis
distrito	Distrito a qual o parque pertence	String	Lisboa
concelho	Concelho a qual o parque pertence	String	Lisboa
freguesia	Freguesia a qual o parque pertence	String	Santa Maria Maior

d) Meteorologia

Por último, é apresentado na Tabela 17, o conjunto de dados meteorológicos, também obtido através da API Weatherbit[5]. Este conjunto de dados meteorológicos foi utilizado tanto para os dados de Vila Nova de Famalicão quanto para os de Lisboa, enriquecendo a nossa compreensão das correlações entre as condições meteorológicas e a ocupação dos parques de estacionamento, isto é, a influência que o clima tem sobre a ocupação dos parques de estacionamento.

Tabela 17 - Análise dos dados meteorológicos.

Variáveis	Descrição	Formato	Exemplo de Valores Possíveis
timestamp_local	Registo de data e hora na hora local	Data	2020-10-25T01:00:00
night_day	Parte do dia (d = dia / n = noite)	String	d
precip	Precipitação acumulada em equivalente líquido (por defeito mm) no local	Decimal	0.0
temp	Temperatura (por defeito Celcius) no local	Decimal	10.2

[5] https://www.weatherbit.io/api/historical-weather-api

4.3.3 Preparação dos dados

A fase de Preparação dos Dados, no contexto do CRISP-DM (Cross-Industry Standard Process for Data Mining) marca um ponto crucial na jornada de qualquer projeto de análise de dados. Nesta etapa, a nossa atenção está voltada para a limpeza, transformação e enriquecimento dos conjuntos de dados, com o objetivo de torná-los prontos para análise e modelagem.

No caso deste projeto, as maiores alterações foram realizadas no conjunto de dados referentes aos movimentos de parques de estacionamento em Lisboa.

a) Cidade de Lisboa

Uma das tarefas fundamentais na preparação deste conjunto de dados de Lisboa é a correção de variáveis, as quais podem incluir dados inconsistentes, valores em falta ou outros problemas que possam afetar a qualidade dos resultados. Estas correções basearam-se no que foi observado na fase anterior, sendo algumas delas:

- Eliminar linhas em que o número de lugar ocupados é superior à capacidade máxima do parque;
- Tratamento da variável 'position', de forma a ser possível obter as varáveis longitude e latitude;
- Tratamento da variável 'data_ocupação', de maneira a ser possível obter as variáveis data (ex.: 15/03/2021) e hora (ex.: 13:00:00);
- Tratamento da variável correspondente aos nomes dos parques, de modo a substituir caracteres que possam influenciar processos futuros, como acentos.

Para além disto, devido à quantidade elevada de parques a serem tratados neste conjunto de dados, foi concluído pela equipa de desenvolvimento do projeto que se deveria diminuir à quantidade de registos para os dez parques com mais movimentos registados.

De seguida, foi dado início à junção dos dados de Lisboa com os dados obtidos a partir da API geográfica. Para isto, foram selecionadas as variáveis latitude e longitude dos parques de Lisboa e realizada uma chamada à API para obter os dados desta. Isso enriquece o conjunto de dados, permitindo uma análise mais abrangente das características espaciais dos parques de Lisboa e sua relação com a ocupação.

Após isto, procedeu-se ao enriquecimento do conjunto de dados de Lisboa através de variáveis já existentes no mesmo:

- Utilização da variável corresponde à data completa para obter as variáveis: 'month', 'dayofweek', 'weekend', 'weekofyear';

– Utilização da variável corresponde à hora completa para obter as variáveis: 'hour', 'minutes', 'timeofday'.

De forma a incorporar o conjunto de dados meteorológicos, obtidos através da API de meteorologia histórica, foram utilizadas as variáveis longitude, latitude, hora e data para realizar a interligação com os restantes dados.

Após discussão com a equipa de projeto, foi sugerido também serem acrescentadas algumas características dos parques através de uma pesquisa sobre as características dos mesmos, de maneira a que numa futura fase possa ser testada a relevância destas. Estas características representam variáveis binárias e são as seguintes:

- 'park_roof' que indica que o parque é coberto ou não;
- 'park_paid' que indica se o parque é pago;
- 'park_electric' que indica se o parque tem estacionamento elétrico;
- 'park_currency' que indica se o pagamento do parque pode ser em dinheiro; e
- 'park_ATM' que indica se o pagamento do parque pode ser com cartão de crédito.

b) Cidade de Vila Nova de Famalicão

No que diz respeito ao conjunto de dados de Vila Nova de Famalicão, dado que este já se encontrava tratado de um projeto anterior, como preparação para as fases seguintes do CRISP-DM, apenas foi realizado o tratamento da variável correspondente aos nomes dos parques, de forma a substituir carateres que podiam influenciar processos futuros, como acentos.

Após isto, foi realizado o enriquecimento do conjunto de dados de Vila Nova de Famalicão através de variáveis já existentes no mesmo:

– Utilização da variável corresponde à data completa para obter as variáveis: 'month', 'dayofweek', 'weekend', 'weekofyear'

– Utilização da variável corresponde à hora completa para obter as variáveis: 'hour', 'minutes', 'timeofday'

Seguido disto, de forma a incorporar o conjunto de dados meteorológicos, obtidos através da API de meteorologia histórica, como nos dados de Lisboa, foram utilizadas as variáveis longitude, latitude, hora e data para realizar a interligação com os restantes dados.

Por fim, como nos dados de Lisboa, foram acrescentadas algumas características dos parques através de uma pesquisa sobre as características dos mesmos:

- 'park_roof' que indica que o parque é coberto ou não;
- 'park_paid' que indica se o parque é pago;
- 'park_electric' que indica se o parque tem estacionamento elétrico;
- 'park_currency' que indica se o pagamento do parque pode ser em dinheiro; e
- 'park_ATM' que indica se o pagamento do parque pode ser com cartão de crédito.

c) Junção de todos os dados

Após todos os tratamentos e enriquecimento dos dados, tanto de Lisboa e Vila Nova de Famalicão, foi levado em conta a atribuição do mesmo nome a todas as variáveis em ambos os conjuntos de dados, de modo a ser possível a junção de ambos sem qualquer contratempo. Posto isto, na Tabela 18, é apresentado o nome do conjunto de dados que representa todos os conjuntos de dados em trabalho e o nome que estes possuíam no correspondente conjunto de dados de Lisboa ou Vila Nova de Famalicão.

Tabela 18 - Nome final das variáveis.

Final	**Vila Nova de Famalicão**	**Lisboa**
nome	name	nome
ocupacao	totalplaces	ocupacao
capacidade_max	totaloccupied	capacidade_max
lat	latitude	lat
lon	longitude	lon
distrito	district	distrito
concelho	city	concelho
freguesia	county	freguesia
park_roof	park_roof	park_roof
park_paid	park_paid	park_paid
park_electric	park_electric	park_electric

Final	Vila Nova de Famalicão	Lisboa
park_currency	park_currency	park_currency
park_ATM	park_ATM	park_ATM
data	date	data
month	month	month
dayofweek	dayofweek	dayofweek
weekend	weekend	weekend
weekofyear	weekofyear	weekofyear
hora	fulltime	hora
hour	hour	hour
minutes	minutes	minutes
timeofday	timeofday	timeofday
precip	precip	precip
temp	temp	temp
night_day	night_day	night_day

Após ser realizada a junção de todos os conjuntos de dados, foram consideradas as fases seguintes que necessitaram que as variáveis estivessem representadas da melhor forma possível para o processo da testagem dos modelos. Para isto, seguindo os critérios do Instituto Português do Mar e da Atmosfera[6], foi realizada a categorização dos dados de temperatura (ex.: 1-Muito frio (inferior a 5°C), 6-Muito quente (superior a 30°C)) e precipitação (ex.: 1-Fraca (inferior a 0,5mm/h), 3-Forte (superior a 4mm/h))., transformando assim a variável "temp" em "class_temp". Para além desta categorização, também foi realizada a categorização da variável "minutes" onde, por exemplo, a classe 1 se estende do valor 0 ao valor 15 e a classe 4 de 45 a 60.

[6] https://www.ipma.pt/en/index.html

Finalmente, foram calculadas as variáveis *target* que são utilizadas nas próximas fases:

- "Lugares livres", que foi criada através da diferença entre a capacidade máxima do parque e o número de lugares ocupados no parque.
- Variável binária que representa se o parque se encontra livre ou não, classificada de acordo com a percentagem de lugares livres: se o número de lugares livres for maior que 25% o parque encontra-se livre (0); se for inferior a 25% encontra-se ocupado (1).

4.3.4 Modelação

Nesta etapa, os esforços são concentrados na criação de modelos robustos de classificação e regressão que sejam capazes de prever com precisão um aspeto fundamental da gestão de parques de estacionamento. Ao nível de modelos de classificação, se o parque está ''livre'' ou ''ocupado'', e ao nível de regressão para estimar o número exato de lugares livres. De todos os dados tratados anteriormente, os dados que serão usados nestes modelos são: 'nome', 'distrito', 'concelho', 'freguesia', 'park_roof', 'park_paid', 'park_electric', 'park_currency', 'park_ATM', 'month', 'dayofweek', 'weekend', 'weekofyear', 'hour', 'class_minutes', 'timeofday', 'class_precip', 'class_temp', 'night_day'.

Para realizar esta tarefa, estabelecemos quatro cenários distintos, que são apresentados na Tabela 19 Cada um destes atua como um ponto de partida para os nossos modelos de classificação e regressão. Um ponto a ser referido é que os cenários A e B representam os mesmos dados de forma diferente, isto é, ao serem realizados os testes destes pretende-se descobrir se a especificação sobre mais detalhes do parque, para além do nome deste, é relevante para obtenção de melhores resultados nos modelos. Os outros dois cenários, no entanto, serviram para perceber a influencia que os dados de meteorologia e data têm nos resultados dos modelos respetivamente.

Tabela 19 - Cenários de teste.

Cenário	Atributos
A	**Localização + Hora** ['distrito', 'concelho', 'freguesia', 'park_roof', 'park_paid', 'park_electric', 'park_currency', 'park_ATM', 'hour', 'class_minutes', 'timeofday']
B	**Localização + Hora** ['nome', 'hour', 'class_minutes', 'timeofday']

Cenário	Atributos
C	**Localização + Hora + Meteorologia** ['nome', 'hour', 'class_minutes', 'timeofday', 'class_precip', 'class_temp', 'night_day']
D	**Localização + Data + Hora + Meteorologia** ['nome', 'month', 'dayofweek', 'weekend', 'weekofyear', 'hour', 'class_minutes', 'timeofday', 'class_precip', 'class_temp', 'night_day']

a) Modelos de Classificação

A classificação tem como objetivo identificar o estado de um parque de estacionamento, considerando-o como 'livre' quando o número de lugares disponíveis exceder 25%. Esse valor é uma referência inicial e o modelo será ajustável para outros critérios no futuro. Essa abordagem auxiliará na tomada de decisões rápidas e eficazes em relação à disponibilidade de estacionamento.

Como apresentado na Tabela 20, trabalhámos com três algoritmos diferentes para a tarefa de classificação, explorando as suas capacidades de previsão e identificando qual deles oferece o desempenho mais sólido para a classificação de estados dos parques de estacionamento.

Tabela 20 - Algoritmos de Classificação.

Algoritmo de classificação	Descrição
Decision tree **(DT)**	Permite identificar os critérios necessários das variáveis para classificar os diversos alvos. Maior restrição e eficiência na aprendizagem.
Naive Bayes **(NB)**	Classificador probabilístico simples que utiliza o teorema de Bayes, pressupondo que os atributos são independentes no contexto do problema.
Neural Networks **(NN)**	Algoritmo que imita o funcionamento do cérebro humano, utilizando camadas de neurónios artificiais para aprender padrões complexos nos dados. Estes são eficazes em tarefas complexas, mas podem ser intensivas em recursos.

Para além destes algoritmos, também foi considerado o algoritmo *Support Vector Machines* (SVM), no entanto, a execução deste não foi possível devido à sua complexidade que resultaria em tempos de execução muito altos. Portanto, no que é referente a modelos de classificação, e como representado no cálculo a seguir apresentado, foram criados 12 modelos.

$$M_{c,a,t} = \begin{cases} c = A \dots D \\ a = DT, NB \text{ e } NN \\ t = Ocupacao \end{cases}$$

Modelos =4 cenários x3 algoritmos x1 target=12 modelos

Como é apresentado na Figura 32, o presente conjunto de dados possui um desequilíbrio no que diz respeito a *target*, isto é, num total de 1 193 849 registos 82,2% (981530 registos) apresentam o valor 0 e 17,8% (212319 registos) apresentam o valor 1.

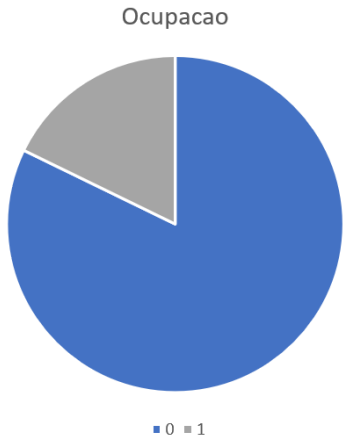


Figura 32 - Valores da target antes da técnica SMOTE.

Para corrigir este desequilíbrio foi utilizada uma técnica denominada SMOTE. Esta trata-se de uma técnica de *oversampling*, esta é responsável por equilibrar o conjunto de dados de forma a que exista o mesmo número de entradas de dados para ambas as classes da variável *target.* Esta técnica gera amostras sintéticas para a classe minoritária a partir de pontos de dados que estão mais próximos uns dos outros. Dessa forma, é possível obter resultados de sensibilidade aprimorados.

b) *Modelos de Regressão*

Na criação de modelos de regressão, nos quais a variável alvo será o número exato de lugares livres num parque de estacionamento. Mais uma vez, como apresentado na Tabela 21, aplicámos três algoritmos de regressão distintos, analisando como cada um se adapta ao desafio de estimar a ocupação de estacionamento de maneira precisa.

Tabela 21 - Algoritmos de Regressão.

Algoritmo de regressão	**Descrição**
Decision tree **(DT)**	Algoritmo que utiliza uma estrutura de árvore para prever valores numéricos a partir de dados, identificando os critérios necessários das variáveis para fazer essas previsões.
Random Forest **(RF)**	Algoritmo que combina múltiplas árvores de decisão para prever valores numéricos a partir de dados, resultando em previsões mais precisas e robustas.
Linear Regressions **(LR)**	Algoritmo de aprendizagem de máquina que modela a relação entre variáveis independentes e uma variável dependente numérica, permitindo prever ou entender essa relação de forma linear.

Portanto, no que é referente a modelos de regressão, e como representado no cálculo a seguir, foram criados 12 modelos.

$$M_{c,a,t} = \begin{cases} c = A \dots D \\ a = DT, RF \text{ E } LR \\ t = Lugares\_livres \end{cases}$$

Modelos =4 cenários x3 algoritmos x1 target=12 modelos

c) Construção dos modelos

No que diz respeito à construção, ambos os tipos de modelos (classificação ou regressão) começam pela escolha de variáveis do cenário correspondente, seguido do *encode* das variáveis necessárias para o funcionamento do algoritmo do modelo.

Após isto, segue-se a identificação da variável *target* e a divisão dos dados em dados de treino e teste.

Aquando da definição do algoritmo, foram escolhidos os hiperparâmetros que podem influenciar a execução do mesmo, e para melhoria disto, foi utilizado o módulo de GridSearchCV, que combina os diferentes parâmetros de forma a encontrar os mais otimizados.

Por fim, foi realizado o treino do modelo e a apresentação dos resultados através das métricas, que, no caso dos modelos de classificação, são obtidas com *cross-validation*, isto é, o conjunto de dados designado para testes é dividido em partes, treinando e testando o modelo várias vezes para garantir resultados mais robustos.

4.3.5 Avaliação

A fase de Avaliação, no âmbito do CRISP-DM (*Cross-Industry Standard Process for Data Mining*), representa a fase onde os esforços e dedicações anteriores convergem para a análise dos resultados obtidos com os modelos desenvolvidos. Nesta etapa, apresentamos os resultados de um total de 12 modelos de classificação e 12 modelos de regressão, divididos em quatro cenários distintos.

a) Modelos de Classificação

No que diz respeito aos modelos do cenário A, como se pode observar na Tabela 22, o algoritmo que obteve o melhor desempenho foi o DT, embora este não tenha atingido os valores das métricas definidos na fase de compreensão do negócio.

Tabela 22 - Resultados de classificação do cenário A.

Algoritmo	Acuidade (%)		Precisão (%)		Sensibilidade (%)		F1-Score (%)		Especificidade (%)	
	Média	Desvio	Média	Desvio	Média	Desvio	Média	Desvio	Média	Desvio
DT	80.82	17.24	79.74	21.70	80.82	17.24	78.80	20.99	71.50	0.16
NB	79.28	16.84	79.44	20.89	79.28	16.84	76.69	20.88	65.99	0.86
NN	76.68	19.43	73.25	26.88	76.68	19.43	73.09	24.69	70.41	3.25

No que diz respeito aos modelos do cenário B, como se pode verificar na Tabela 23, o algoritmo que obteve o melhor desempenho foi o DT, atingido os valores das métricas definidos na fase de compreensão do negócio.

Tabela 23 - Resultados de classificação do cenário B.

Algoritmo	Acuidade (%)		Precisão (%)		Sensibilidade (%)		F1-Score (%)		Especificidade (%)	
	Média	Desvio	Média	Desvio	Média	Desvio	Média	Desvio	Média	Desvio
DT	88.32	13.60	89.45	11.66	88.32	13.60	87.46	15.73	93.31	2.90
NB	68.87	11.05	76.03	13.13	68.87	11.05	66.70	11.97	85.24	3.09
NN	87.86	14.04	88.81	12.47	87.86	14.04	86.86	16.58	93.34	3.12

No que diz respeito aos modelos do cenário C, como se pode observar na Tabela 24, o algoritmo que obteve o melhor desempenho foi o DT, atingido os valores das métricas definidos na fase de compreensão do negócio.

Tabela 24 - Resultados de classificação do cenário C.

Algoritmo	Acuidade (%)		Precisão (%)		Sensibilidade (%)		F1-Score (%)		Especificidade (%)	
	Média	Desvio	Média	Desvio	Média	Desvio	Média	Desvio	Média	Desvio
DT	91.60	8.66	93.27	6.14	91.60	8.66	91.31	9.18	96.99	1.53
NB	68.82	11.03	75.95	13.12	68.82	11.03	66.64	11.95	85.91	3.87
NN	88.88	11.91	89.99	10.26	88.88	11.91	88.43	12.95	94.83	2.22

No que diz respeito aos modelos do cenário D, como se pode verificar na Tabela 25, o algoritmo que obteve o melhor desempenho foi o NN, atingido os valores das métricas definidos na fase de compreensão do negócio.

Tabela 25 - Resultados de classificação do cenário D.

Algoritmo	Acuidade (%)		Precisão (%)		Sensibilidade (%)		F1-Score (%)		Especificidade (%)	
	Média	Desvio	Média	Desvio	Média	Desvio	Média	Desvio	Média	Desvio
DT	88.96	11.86	90.08	10.01	88.96	11.86	88.46	13.04	94.57	2.27
NB	69.21	11.78	75.52	13.93	69.21	11.78	67.19	12.84	85.76	3.79
NN	89.31	9.84	90.82	7.70	89.31	9.84	88.96	10.50	93.54	2.02

Por fim, comparando os algoritmos com melhor desempenho em cada cenário dos modelos de classificação, apresentados na Tabela 26, chegámos à conclusão de que, no que diz respeito ao conjunto de dados referentes ao parque, os atributos, para além do nome do parque, não ajudam na melhoria dos resultados que, pelo contrário, pioram consideravelmente. Isto também se pode verificar para o conjunto de dados referentes à data, uma vez que, comparando o cenário C e D, que diferem na presença destes dados, podemos observar que o cenário sem este conjunto de dados foi o que obteve os melhores resultados.

Tabela 26 - Comparação de resultados entre cenários dos Modelos de Classificação.

Cenário	Acuidade (%)		Precisão (%)		Sensibilidade (%)		F1-Score (%)		Especificidade (%)		Algoritmo
	Média	Desvio	Média	Desvio	Média	Desvio	Média	Desvio	Média	Desvio	
A	80.82	17.24	79.74	21.70	80.82	17.24	78.80	20.99	71.50	0.16	DT

Cenário	Acuidade (%)		Precisão (%)		Sensibilidade (%)		F1-Score (%)		Especificidade (%)		Algoritmo
	Média	Desvio	Média	Desvio	Média	Desvio	Média	Desvio	Média	Desvio	
B	88.32	13.60	89.45	11.66	88.32	13.60	87.46	15.73	93.31	2.90	DT
C	91.60	8.66	93.27	6.14	91.60	8.66	91.31	9.18	96.99	1.53	DT
D	89.31	9.84	90.82	7.70	89.31	9.84	88.96	10.50	93.54	2.02	NN

b) Modelos de Regressão

No que diz respeito aos modelos do cenário A, como se pode observar na Tabela 27, o algoritmo que obteve o melhor desempenho foi o RF, embora este não tenha atingido o valor estipulado para a métrica R2Score na compreensão do negócio.

Tabela 27 - Resultados de regressão do cenário A.

Algoritmo	MSE	R2Score	MAE	RAE
DT	7543.88	0.21	59.66	43.04%
LR	7740.84	0.19	61.54	44.40%
RF	7544.02	0.21	59.66	43.04%

No que diz respeito aos modelos do cenário B, como se verifica na Tabela 28, o algoritmo que obteve o melhor desempenho foi o DT, atingindo o valor estipulado para a métrica R2Score na compreensão do negócio e possuindo os melhores resultados nas restantes métricas.

Tabela 28 - Resultados de regressão do cenário B.

Algoritmo	MSE	R2Score	MAE	RAE
DT	1372.67	0.86	18.84	13.59%
LR	9461.81	0.01	73.73	53.19%
RF	1372.74	0.86	18.84	13.59%

No que diz respeito aos modelos do cenário C, como se observa na Tabela 29, o algoritmo que obteve o melhor desempenho foi o DT, atingindo o valor estipulado para a métrica R2Score na compreensão do negócio e possuindo os melhores resultados nas restantes métricas.

Tabela 29 - Resultados de regressão do cenário C.

Algoritmo	MSE	R2Score	MAE	RAE
DT	1302.81	0.86	18.28	13.18%
LR	9426.11	0.01	73.80	53.24%
RF	1309.62	0.86	18.33	13.22%

No que diz respeito aos modelos do cenário D, como se pode observar na Tabela 30, o algoritmo que obteve o melhor desempenho foi o DT, atingindo o valor estipulado para a métrica R2Score na compreensão do negócio e possuindo os melhores resultados nas restantes métricas.

Tabela 30 - Resultados de regressão do cenário D.

Algoritmo	MSE	R2Score	MAE	RAE
DT	365.85	0.96	5.82	4.20%
LR	9412.88	0.01	73.52	53.04%
RF	984.84	0.90	18.88	13.62%

Por fim, comparando os algoritmos com melhor desempenho em cada cenário dos modelos de regressão, apresentados na Tabela 31, podemos concluir que, em semelhança aos modelos de classificação, os atributos do parque, para além do nome do mesmo, pioram os resultados. No entanto, em contraste com os resultados dos modelos de classificação, os atributos de data incluídos no cenário D melhoram significativamente os resultados em todas as métricas observadas, tonando o cenário D o que possui o melhor modelo.

Tabela 31 - Comparação de resultados entre cenários dos Modelos de Regressão.

Cenário	MSE	R2Score	MAE	RAE	Algoritmo
A	7544.02	0.21	59.66	43.04%	RF
B	1372.67	0.86	18.84	13.59%	DT
C	1302.81	0.86	18.28	13.18%	DT
D	365.85	0.96	5.82	4.20%	DT

4.3.6 Implementação

A fase de Implementação, no âmbito do CRISP-DM (*Cross-Industry Standard Process for Data Mining*), representa o ponto onde os modelos e as análises desenvolvidos se tornam ferramentas práticas e aplicáveis.

Nesta etapa, focalizámo-nos inicialmente na criação de uma API versátil, adaptável para a integração de qualquer modelo criado, para depois a utilizar na página anteriormente criada, na secção 4.2.2, na realização de previsões, que possui todos os mecanismos necessários para as previsões.

a) Criação da API

No que diz respeito à criação da API, inicialmente, esta foi configurada com modelo de regressão caraterizado por ser inserido no cenário D e utilizar o algoritmo *Random Forest* (RF). Esta configuração inicial serviu como um ponto de partida, permitindo que a API fosse testada e validada com sucesso. Os resultados obtidos com essa configuração ajudam a refinar o desempenho da API e a garantir que esteja pronta para implantação em cenários do mundo real.

No que diz respeito aos requisitos da API, ela funciona quando recebe os seguintes dados: o nome do parque, a data de início da previsão e a data de fim da previsão. No entanto, o modelo utilizado para validar a API requer outros parâmetros, sendo estes: nome do parque codificado, mês, hora, dia da semana, semana do ano, variável binária se é fim de semana ou não, variável binária (se é dia ou noite), classe de temperatura, classe de precipitação, classe do minuto e classe de altura do dia.

Para obter todas as informações necessárias para o modelo, foram realizadas transformações nos dados fornecidos, como a utilização de ficheiros de codificação criados aquando da criação do modelo. Isso ocorreu porque o modelo utilizado só aceita variáveis numéricas e chamadas à API de meteorologia (Weatherbit), onde foram impostos limites para não permitir o prosseguimento da previsão se esta ultrapassar 10 dias a partir do dia atual, uma vez que este é o limite da API meteorológica.

Relativamente à resposta, e como apresentado na Figura 33 da ferramenta Postman, a API fornece os dados da capacidade máxima do parque, a data da previsão no formato DD-MM-AAAA, a hora da previsão e o valor da previsão, que neste caso representa o número de lugares livres disponíveis no parque.

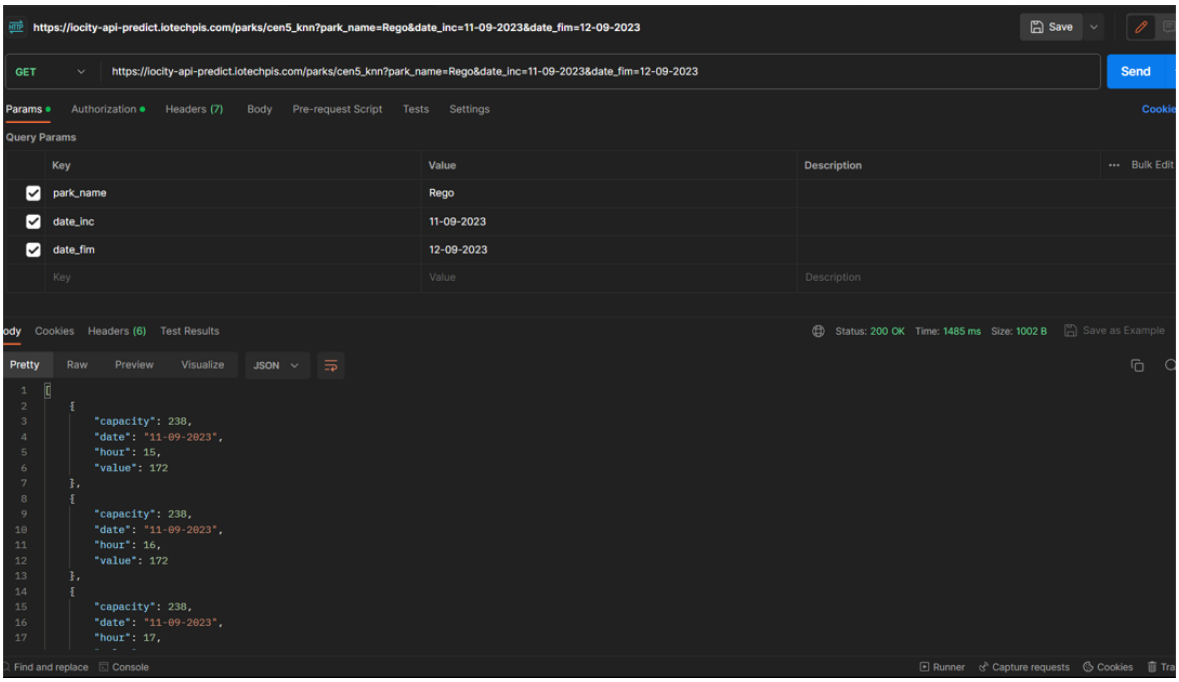


Figura 33 - Resposta da API no Postman.

b) *Visualização ioScience*

Após a configuração da API, esta foi integrada no projeto ioScience, especificamente na página de previsão que foi previamente atualizada. Essa integração permite que as previsões de ocupação de parques de estacionamento geradas pelos modelos sejam acessíveis e utilizáveis de forma prática, facilitando a tomada de decisões informadas no contexto da gestão de parques de estacionamento.

Para ser possível concretizar um pedido à API, de modo a respeitar todos os seus requisitos anteriormente mencionados, foi criado um formulário de introdução dos dados na página destinada à previsão, como ilustrado na Figura 34 , de forma a não aceitar valores inválidos, como uma data de início anterior ao dia atual. Além disso, também foi criado um filtro para facilitar o acesso ao nome do parque, onde se pode inserir o nome da cidade e rua a que este pertence para filtrar o campo do nome do parque e apenas apresentar os parques relacionados com a cidade e rua selecionadas. O campo de rua não é obrigatório ser selecionado.

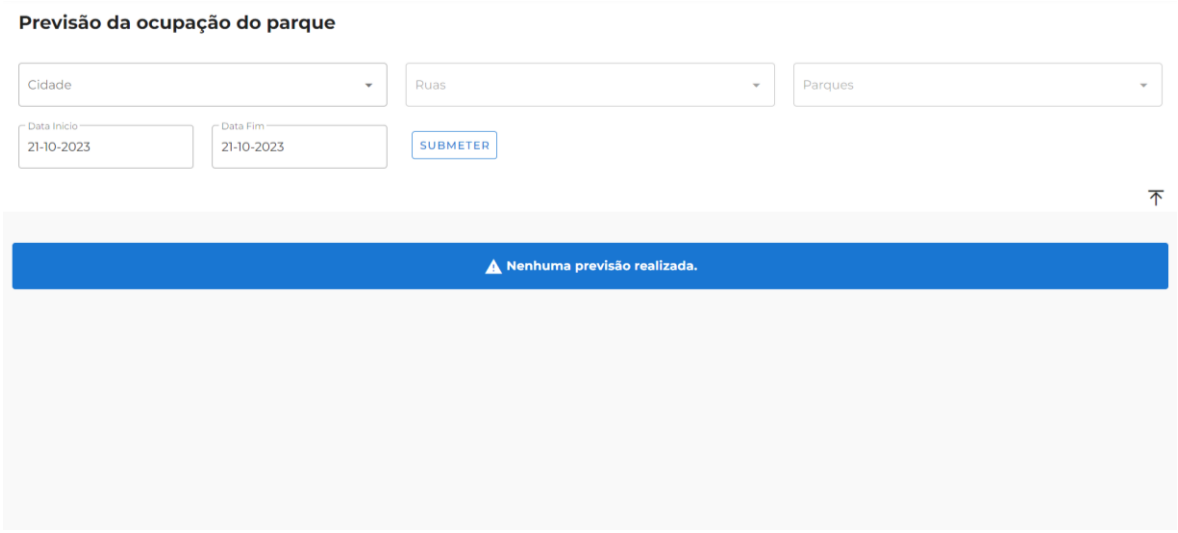


Figura 34 - Campos a preencher para o funcionamento da API.

Por fim, a resposta da API é adaptada de maneira a ser apresentada ao utilizador de forma mais fácil de compreensão na página de previsão. Neste caso, conforme demonstrado na Figura 35, passou por ser a utilização de um gráfico de *heatmap,* onde se faz a divisão dos dias que foram pretendidos para a previsão e as horas correspondentes desses dias. Se o dia atual for selecionado, a página apresenta unicamente os valores das horas futuras.

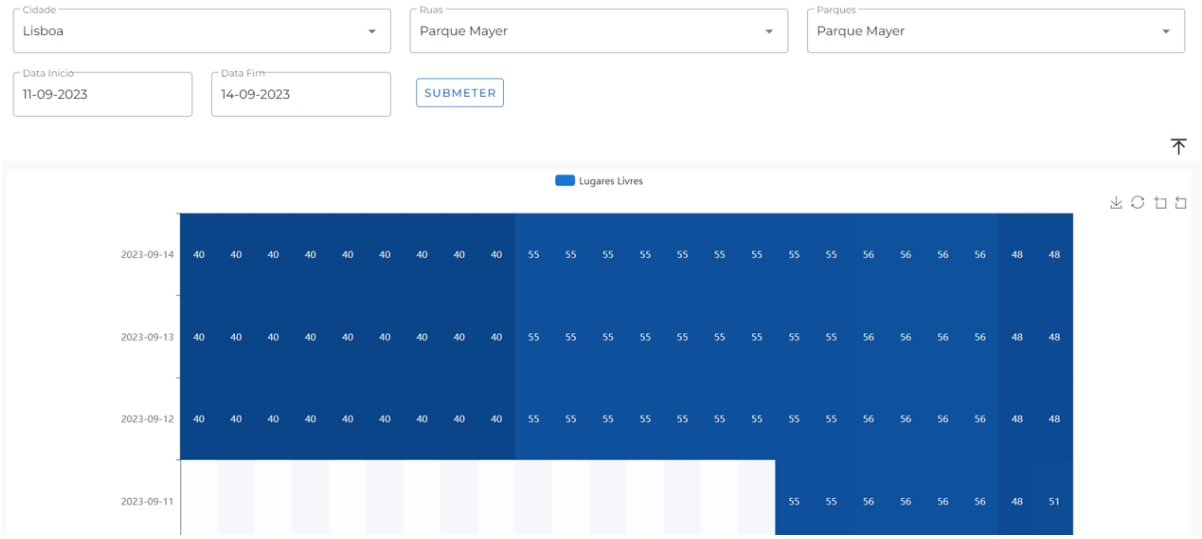


Figura 35 - Apresentação da resposta da API na página de previsão

5. DISCUSSÃO DE RESULTADOS

Em relação à parte prática desenvolvida neste projeto, conseguem-se identificar duas componentes principais de resultados. A primeira consiste na otimização da plataforma ioScience, abrangendo a otimização de todo o módulo analítico. A segunda componente refere-se à implementação do novo módulo preditivo da solução, com a subsequente apresentação do mesmo, utilizando os dados de ocupação de parques de estacionamento como exemplo.

No que diz respeito à primeira componente, como é apresentado na secção 4.2, esta pode ser dividida em otimizações ao nível visual e ao nível funcional. Ao nível visual, o objetivo deste passava por facilitar ao utilizador a visualização e compreensão dos dados. Para além da plataforma ser totalmente responsiva e escalável para todos os tipos de dispositivos, como computadores, tablets, smartphones e ecrãs táteis de 75 polegadas, foram efetuadas as seguintes alterações:

- Otimização da barra de menu, para uma melhor apresentação das diferentes páginas da plataforma;
- Otimização dos gráficos, para uma uniformização das suas componentes, bem como a implementação de novos gráficos para uma maior capacidade de apresentação de dados;
- Otimização das KPIs, para uma melhor experiência de visualização, que culmina numa melhor compreensão dos dados apresentados nas novas KPI's criadas pela equipa. Alguns dos exemplos destas KPI's são: Número de parque públicos; Número de parque com pagamento eletrónico; Média de lugares disponíveis; e Movimentos por parque.

Ainda na primeira componente, no que diz respeito à otimização funcional da plataforma, existiu uma otimização, bem como, uma criação de funcionalidades para a plataforma do ioScience, como é apresentado na Tabela 8. No que diz respeito à funcionalidades otimizadas e desenvolvidas, estas são as seguintes:

- Adicionar várias *Dashboards;*
- Guardar gráficos na *dashboard* selecionada;
- Guardar layout dos gráficos após mudar a dimensão e localização nas *dashboards;*
- Opção de seleção de tema de cores;
- Capacidade de realizar *fullscreen* da plataforma;

- Efetuar *Drill-Down* e *Rollup* nos KPIs;
- Módulo preditivo;
- Página inicial para a seleção do projeto pretendido.

Esta primeira componente foi a responsável por concluir os seguintes objetivos deste projeto:

- Otimizar o protótipo de uma solução web;
- Melhorar o processo de análise de dados em tempo-real;
- Adaptar a solução para diferentes projetos.

Já no que implica a segunda componente de implementação do módulo preditivo, foram construídos dois tipos de modelos, modelos de classificação e modelos de regressão, sendo que para cada tipo foram utilizados 4 cenários.

No que diz respeito aos modelos de classificação, estes tiveram como *target* uma variável binária, que se foca na quantidade de lugares livres. Se esta fosse maior que 25%, o parque encontra-se livre, caso contrário, encontra-se ocupado. Os modelos de classificação possuíam 5 métricas de avaliação (Acuidade, Precisão, Sensibilidade, F1-Score e Especificidade) e 3 algoritmos (*Decision tree* (DT), *Naive Bayes* (NB), *Neural Networks* (NN)), resultando assim no desenvolvimento de 12 modelos de classificação.

Os melhores resultados obtidos para estes modelos por cenário são apresentados na Tabela 26, sendo que o melhor destes corresponde ao modelo que utilizou o cenário C em combinação com o algoritmo DT. Estes resultados assumem os seguintes valores: 91.60% de Acuidade, 93.27% de Precisão, 91.60% de Sensibilidade, 91.31% de F1-Score e 96.99% de Especificidade.

No caso dos modelos de regressão, estes tiveram como *target* o número de lugares livres no parque de estacionamento. Os modelos de classificação possuíam 4 métricas de avaliação (Erro Quadrático Médio (MSE), R-Quadrado (R2 score), Erro Absoluto Médio (MAE) e Erro Absoluto Relativo (ERA)) e 3 algoritmos (*Decision tree* (DT), *Random Forest* (RF), *Linear Regressions* (LR)), sendo assim desenvolvidos 12 modelos de regressão.

Os melhores resultados obtidos para estes modelos por cenário são apresentados na Tabela 31, sendo que o melhor destes corresponde ao modelo que utilizou o cenário D em combinação com o algoritmo DT. Estes resultados assumem os seguintes valores: 365.85 de MSE, 0.96 de R2Score, 5.82 de MAE e 4.20% de ERA.

Para além disto, ao observar os melhores resultados de ambos os modelos de classificação e regressão, podemos compreender que ambos superaram as métricas estipuladas na compreensão do negócio, sendo isto sinónimo de um desempenho notável e alinhado com os objetivos propostos. Essa superação das expectativas reforça a eficácia dos modelos implementados, evidenciando a sua capacidade de fornecer informações valiosas e contribuir positivamente para as estratégias empresariais.

A partir dos modelos desenvolvidos, foi implementado o módulo preditivo na solução através de uma API desenvolvida em Python. Essa API é utilizada para, dependendo dos pedidos realizados, retornar as previsões realizadas pelo modelo selecionado, sendo que os resultados são apresentados na página de previsões da plataforma ioScience na forma de um *heatmap*.

Esta segunda componente, para além de responder à questão de investigação "Qual a viabilidade de integrar um módulo preditivo no ioScience, seguindo as regras de construção deste?", demonstrando que a implementação do módulo preditivo segundo as regras de construção definidas no ioScience é possível, também foi responsável por seguintes objetivos do projeto:

- Criar um modelo preditivo;
- Implementar APIs em Python.

Por fim, e para concluir o objetivo "Testar e documentar o protótipo", disponibilizaram-se os dados do projeto ioCity para testar a plataforma do ioScience, conseguindo assim solidificar a característica de interoperabilidade dos dados da plataforma, uma vez que no projeto ioCity foram utilizadas diferentes fontes de dados. Adicionalmente, procedeu-se à adaptação da plataforma ioScience para que pudesse ser utilizada no projeto ioCity, conforme apresentado na secção 4.2.3. Com isso, foi possível garantir o sucesso do atual projeto de dissertação.

6. CONCLUSÃO

Este capítulo resume às principais conclusões decorrentes do desenvolvimento da solução e de todo o processo de pesquisa associado. Além disso, a segunda subsecção descreve o trabalho futuro previsto com base na solução atual. Por fim, para avaliar os riscos e destacar as medidas adotadas para mitigar seu impacto no projeto, na última subsecção é apresentada a tabela de riscos.

6.1 Considerações finais

O presente documento teve como propósito descrever todas as diferentes etapas deste projeto de dissertação, desde a compreensão do mesmo até a apresentação dos resultados finais.

No que diz respeito ao título da presente tese, "Pervasive Modular Data Science – Otimização e interoperabilidade", este engloba a essência do que este projeto atingiu. No que diz respeito à "Pervasive Data Science", a aplicação final proporciona a qualquer utilizador a possibilidade de aceder ao conhecimento que a análise dos dados proporciona e de ter uma perspetiva abrangente do estado do objeto em análise, ao qual os dados se referem. Por outro lado, a presente dissertação foi capaz de inserir mais um módulo na plataforma do ioScience, o módulo preditivo, como se pode verificar na arquitetura da solução na secção 4.1. Por outro lado, existiu uma "Otimização e interoperabilidade" da plataforma, como é possível concluir depois de analisadas as otimizações visuais e funcionais, bem como o desenvolvimento de novas funcionalidades, que foram suportadas pela implementação de novos conjuntos de dados (Dados dos Parques da Cidade de Lisboa).

A fase inicial do projeto de dissertação passou por realizar uma revisão de literatura destes conceitos relevantes, bem como a apresentação da patente do ioScience que foi a base deste projeto. Para além disto, foram também expostas algumas patentes que possuem semelhanças com o ioScience, para compreender o estado de arte de soluções semelhantes à em questão.

Os resultados apresentados após a realização desta etapa foram utilizados para melhor definir uma estratégia para responder à questão apresentada no início deste documento, sendo que, para lidar com a complexidade deste projeto foram selecionadas duas metodologias que tiveram um papel crucial. O *Design Science Research* foi utilizado como metodologia de investigação e o SCRUM *Framework* para a componente prática do projeto.

Numa segunda etapa, foi apresentada a componente prática do mesmo onde, como apresentado na Tabela 32, cada objetivo definido anteriormente foi atingido aquando da obtenção dos resultados correspondentes, sendo que a questão de investigação, "Qual a viabilidade de integrar um módulo preditivo no ioScience, seguindo as regras de construção deste?" foi respondida através da implementação do módulo preditivo, segundo as regras de construção definidas no ioScience. O módulo foi desenvolvido de forma modular, com multidados, de fácil utilização e configuração, e com versão offline, além da subsequente alteração na arquitetura da solução.

Tabela 32 - Objetivos VS Resultados Finais.

Objetivos	**Resultados Finais**
O1 - Otimizar o protótipo de uma solução web	Otimização e implementação de componentes visuais; Otimização e implementação de funcionalidades; Reorganização no código.
O2 - Criar um modelo preditivo	Implantação do modulo preditivo no ioScience através do processo de *data mining*.
O1.1 - Testar e documentar o protótipo	Teste utilizando dados o projeto do ioCity.
O1.2 - Melhorar o processo de análise de dados em tempo-real	Otimização das funcionalidades da plataforma ioScience.
O1.3 - Adaptar a solução para diferentes projetos	Implementação da funcionalidade capaz de selecionar o projeto desejado e consequente teste das funcionalidades da plataforma em diferentes projetos.
O2.1 - Explorar algoritmos inteligentes	Elaboração de 24 modelos de previsão.
O2.2 - Implementar APIs em Python	Criação da API de previsão em Python.

Para quantificar todo o trabalho apresentado, destacam-se os principais resultados alcançados:

- 3 otimizações visuais da plataforma ioScience implementadas;
- 8 funcionalidades da plataforma ioScience implementadas;

- 1 nova página da plataforma ioScience implementada;
- 9 KPIs desenhadas no contexto do projeto ioCity;
- 2 páginas adaptadas para o projeto ioCity;
- 24 modelos de previsão criados e testados, 12 modelos de classificação e 12 modelos de regressão;
- 1 API de previsão implementada;
- 2 artigos publicados.

Ao observar a Tabela 32 acima, é justo afirmar que o projeto foi concluído com sucesso, com todos os objetivos iniciais alcançados. Naturalmente, isso não implica que o projeto esteja perfeito e pronto para ser implementado no mercado. Portanto, na subseção 6.3 (Trabalho Futuro), são sugeridas algumas melhorias.

6.2 Tabela de Riscos

Na Tabela 33 são apresentados os riscos identificados para este projeto, com a informação se o risco se verificou (Sim ou Não) e, caso se tenha verificado, as respetivas ações de mitigação. Para além disto, são também apresentadas estimativas da probabilidade de ocorrência e impacto, sendo as repetitivas escalas de 1 (baixo) a 5 (alto). A partir da multiplicação destas estimativas, é realizado o cálculo de severidade de cada risco (P*I=S). A análise dos riscos inerentes a dissertação permitem que os mesmos possam ser prevenidos ou que o dano associados a estes diminua.

Tabela 33 - Lista de Riscos.

ID	Risco	P[7]	I[8]	S[9]	Verificou	Mitigação
1	Falta de experiência em projetos da área	4	4	16	Sim	Estudo das ferramentas e técnicas a utilizadas no projeto, de forma a aumentar o conhecimento das mesmas.

[7] Probabilidade
[8] Impacto
[9] Severidade

ID	Risco	P[7]	I[8]	S[9]	Verificou	Mitigação
2	Elevado grau de complexidade da área de investigação	4	3	12	Sim	Foi aplicado com rigor a metodologia SCRUM por forma a lidar com a complexidade do projeto
3	Incompreensão dos objetivos do projeto e dos resultados esperados	2	4	8	Não	-
4	Reduzida qualidade dos dados	2	3	6	Sim	Foi realizada uma análise rigorosa dos dados e foram identificados e classificados os erros, inconsistências e incoerências.
5	Alteração dos Objetivos e Resultados Esperados	1	5	5	Sim	Realizado um ajuste do plano de trabalho.
6	Incumprimento dos resultados e objetivos esperados	1	5	5	Não	-
7	Má comunicação com o orientador	1	4	4	Não	-
8	Planeamento inadequado	1	4	4	Não	-
9	Avarias nas infraestruturas utilizadas	1	4	4	Não	-
10	Perda de arquivos	1	3	3	Não	-

6.3 Trabalho Futuro

No que diz respeito ao presente projeto, todos os objetivos e resultados esperados foram alcançados, no entanto, existem diferentes desafios futuros que este projeto poderá abordar. Numa futura fase deste projeto, existe a possibilidade de disponibilizar ao utilizador a capacidade de realizar o processo de modelação analítica com a utilização de agentes de inteligência artificial (IA). Além disso, podemos aumentar o tamanho do conjunto de dados utilizado para testar a eficiência e a eficácia da plataforma,

sendo que a diversificação de setores industriais nesses dados também é relevante a ser testada. Por último, no que se refere aos modelos preditivos, num projeto futuro poderá também ser considerada a implementação de algoritmos mais complexos, bem como o aumento de variáveis, como, por exemplo, dados de eventos no conjunto de dados para a criação dos modelos.

REFERÊNCIAS

Ali-ud-din Khan, M., Fahim Uddin, M., & Gupta, N. (Eds.). (2014). Seven V's of Big Data Understanding Big Data to extract Value. Proceedings of the 2014 Zone 1 Conference of the American Society for Engineering Education. https://doi.org/10.1109/ASEEZone1.2014.6820689

Baesens, B. (2014). *Analytics in a Big Data World: The Essential Guide to Data Science and its Applications*. Wiley.

Barros, F., Rodrigues, B., Vieira, J. M. P., & Portela, F. (2023). Pervasive Real-Time Analytical Framework—A case study on car parking monitoring. Information, 14(11), 584. https://doi.org/10.3390/info14110584

Bousdekis, A., Lepenioti, K., Apostolou, D., & Mentzas, G. (2022). Data analytics in quality 4.0: literature review and future research directions. *International Journal of Computer Integrated Manufacturing*, 1–24. https://doi.org/10.1080/0951192x.2022.2128219

Cady, F. (2017). *The Data Science Handbook* (1st ed.). Wiley.

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). *CRISP-DM 1.0: Step-by-step data mining guide*.

Chen, M., Mao, S., & Liu, Y. (2014). Big Data: A Survey. *Mobile Networks and Applications*, *19*(2), 171–209. https://doi.org/10.1007/s11036-013-0489-0

Davies, N., & Clinch, S. (2017). Pervasive *Data Science*. *IEEE Pervasive Computing*, *16*(3), 50–58. https://doi.org/10.1109/mprv.2017.2940956

DBeaver Community. (n.d.). *DBeaver Community | Free Universal Database Tool*. Retrieved February 23, 2023, from https://dbeaver.io/

Filipe Portela, Gisela Fernandes. Method to execute offline data analysis. Portugal PT. ID. 116393, IPC: G06F 16/00 (2019.01). IOTECHPIS – Innovation on Technology, LDA . (2022).

Gibert, K., Horsburgh, J. S., Athanasiadis, I. N., & Holmes, G. (2018). Environmental *Data Science*. *Environmental Modelling &Amp; Software*, *106*, 4–12. https://doi.org/10.1016/j.envsoft.2018.04.005

Gorunescu, F. (2011). *Data Mining: Concepts, Models and Techniques*. Springer Publishing.

Hevner, March, Park, & Ram. (2004). Design Science in Information Systems Research. *MIS Quarterly*, *28*(1), 75. https://doi.org/10.2307/25148625

IBM. (2021, March 8). *IBM SPSS Modeler CRISP-DM Guide*. Retrieved February 27, 2023, from **{**HYPERLINK https://www.ibm.com/docs/it/SS3RA7_18.3.0/pdf/ModelerCRISPDM.pdf**}**

Introduction | Cube Docs. (n.d.). Cube Cloud. Retrieved February 23, 2023, from **{**HYPERLINK https://cube.dev/docs/introduction/**}**

Kelley, K. (2023, February 7). *What is Data Analysis? Methods, Process and Types Explained*. Simplilearn.com. **{**HYPERLINK https://www.simplilearn.com/data-analysis-methods-process-types-article**}**

Kurkovsky, S. (2007). Pervasive computing: Past, present and future. *2007 ITI 5th International Conference on Information and Communications Technology*. https://doi.org/10.1109/itict.2007.4475619

Lepenioti, K., Bousdekis, A., Apostolou, D., & Mentzas, G. (2020). Prescriptive analytics: Literature review and research challenges. *International Journal of Information Management*, *50*, 57–70. https://doi.org/10.1016/j.ijinfomgt.2019.04.003

Maimon, O., & Rokach, L. (2010). *Data Mining and Knowledge Discovery Handbook* (Second Edition). Springer. https://doi.org/10.1007/978-0-387-09823-4

Mansmann, S., & Scholl, M. H. (2007). Extending the Multidimensional Data Model to Handle Complex Data. *Journal of Computing Science and Engineering*. https://doi.org/10.5626/jcse.2007.1.2.125

Marr, B. (2015). *Key Performance Indicators For Dummies*. Wiley.

Megida, D. (2021, April 29). *What is JavaScript? A Definition of the JS Programming Language*. freeCodeCamp.org. Retrieved February 23, 2023, from {HYPERLINK https://www.freecodecamp.org/news/what-is-javascript-definition-of-js/}

Nugent, A., Halper, F., Hurwitz, J. S., & Kaufman, M. (2013). *Big Data For Dummies* (1st ed.). For Dummies.

pandas - Python Data Analysis Library. (n.d.). Pandas. Retrieved February 23, 2023, from {HYPERLINK https://pandas.pydata.org/}

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A Design Science Research Methodology for Information Systems Research. *Journal of Management Information Systems*, *24*(3), 45–77. https://doi.org/10.2753/mis0742-1222240302

Provost, F., & Fawcett, T. (2013). *Data Science* and its Relationship to Big Data and Data-Driven Decision Making. *Big Data*, *1*(1), 51–59. https://doi.org/10.1089/big.2013.1508

Python.org. (n.d.). *What is Python? Executive Summary*. Retrieved February 22, 2023, from {HYPERLINK https://www.python.org/doc/essays/blurb/}

Rodrigues, B., Fernandes, C., Vieira, J. M. P., & Portela, F. (2023). Data mining models to predict parking lot availability. In Lecture Notes in Computer Science (pp. 535–547). https://doi.org/10.1007/978-3-031-49011-8_42

Scheps, S. (2008). *Business Intelligence For Dummies*. Wiley.

Schwaber, K., & Sutherland, J. (2020). *The SCRUM Guide*. SCRUM Guides. Retrieved February 9, 2023, from {HYPERLINK https://SCRUMguides.org/SCRUM-guide.html}

SCRUM.org. (n.d.). *What is SCRUM?* Retrieved February 9, 2023, from {HYPERLINK https://www.SCRUM.org/resources/what-is-SCRUM}

Song, I. Y., & Zhu, Y. (2015). Big data and *Data Science*: what should we teach? *Expert Systems*, *33*(4), 364–373. https://doi.org/10.1111/exsy.12130

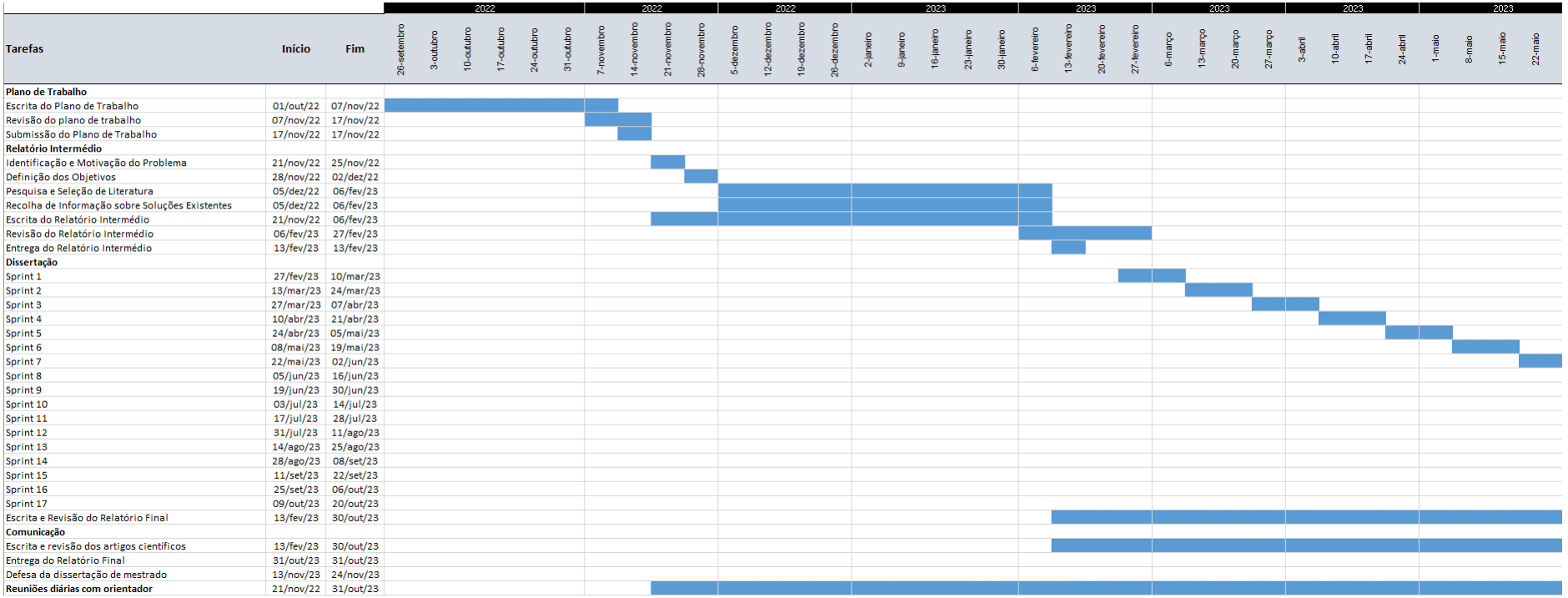
Steele, B., Chandler, J., & Reddy, S. (2016). *Algorithms for Data Science* (1st ed. 2016). Springer.

Visual Studio Code. (2021, November 3). *Visual Studio Code Frequently Asked Questions*. Retrieved February 23, 2023, from {HYPERLINK https://code.visualstudio.com/docs/supporting/FAQ}

vuejs.org. (n.d.). *Introduction | Vue.js*. Vue.js. Retrieved February 23, 2023, from {HYPERLINK https://vuejs.org/guide/introduction.html}

Zikopoulos, I., Eaton, C., & Zikopoulos, P. (2011). *Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data* (1st ed.). McGraw-Hill Osborne Media.

ANEXO I - DIAGRAMA DE GANTT



Tarefas	Início	Fim
Plano de Trabalho		
Escrita do Plano de Trabalho	01/out/22	07/nov/22
Revisão do plano de trabalho	07/nov/22	17/nov/22
Submissão do Plano de Trabalho	17/nov/22	17/nov/22
Relatório Intermédio		
Identificação e Motivação do Problema	21/nov/22	25/nov/22
Definição dos Objetivos	28/nov/22	02/dez/22
Pesquisa e Seleção de Literatura	05/dez/22	06/fev/23
Recolha de Informação sobre Soluções Existentes	05/dez/22	06/fev/23
Escrita do Relatório Intermédio	21/nov/22	06/fev/23
Revisão do Relatório Intermédio	06/fev/23	27/fev/23
Entrega do Relatório Intermédio	13/fev/23	13/fev/23
Dissertação		
Sprint 1	27/fev/23	10/mar/23
Sprint 2	13/mar/23	24/mar/23
Sprint 3	27/mar/23	07/abr/23
Sprint 4	10/abr/23	21/abr/23
Sprint 5	24/abr/23	05/mai/23
Sprint 6	08/mai/23	19/mai/23
Sprint 7	22/mai/23	02/jun/23
Sprint 8	05/jun/23	16/jun/23
Sprint 9	19/jun/23	30/jun/23
Sprint 10	03/jul/23	14/jul/23
Sprint 11	17/jul/23	28/jul/23
Sprint 12	31/jul/23	11/ago/23
Sprint 13	14/ago/23	25/ago/23
Sprint 14	28/ago/23	08/set/23
Sprint 15	11/set/23	22/set/23
Sprint 16	25/set/23	06/out/23
Sprint 17	09/out/23	20/out/23
Escrita e Revisão do Relatório Final	13/fev/23	30/out/23
Comunicação		
Escrita e revisão dos artigos científicos	13/fev/23	30/out/23
Entrega do Relatório Final	31/out/23	31/out/23
Defesa da dissertação de mestrado	13/nov/23	24/nov/23
Reuniões diárias com orientador	21/nov/22	31/out/23

Figura 36 - Diagrama de Gantt - Parte 1

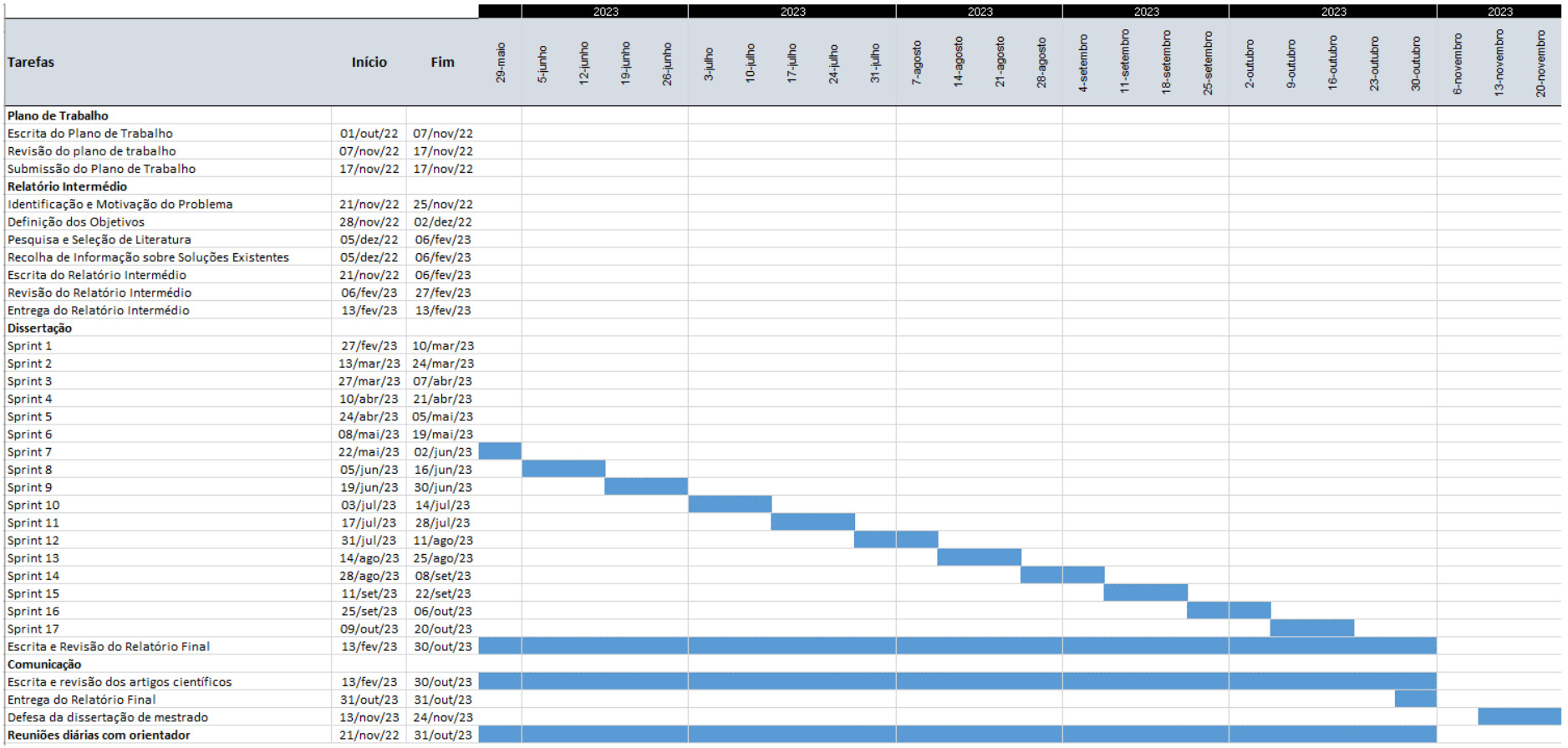


Figura 37 - Diagrama de Gantt - Parte 2

ANEXO II – ARTIGO 'DATA MINING MODELS TO PREDICT PARKING LOT AVAILABILITY'

Artigo com o objetivo de compreender como as condições meteorológicas influenciam a previsão da ocupação dos parques de estacionamento e identificar o algoritmo de previsão mais eficaz.

Estado: Aceite para publicação

Local de publicação: *Lecture Notes in Computer Science* (EPIA 2023). Springer.

Referencia: Beatriz Rodrigues, José Vieira, Carlos Fernandes and Filipe Portela (2023), Data Mining Models to predict parking lot availability. Progress in Artificial Intelligence - Lecture Notes in Computer Science (EPIA 2023). LNAI Volume 14116. ISBN: 978-3-031-49010-1. Springer. DOI:10.1007/978-3-031-49011-8_42

Data Mining Models to predict parking lot availability

Beatriz Rodrigues[1], Carlos Fernandes[1], José Vieira[1] and Filipe Portela[1,2*]

[1]IOTECH- Innovation on Technology, Trofa, Portugal
[2]Algoritmi Centre, University of Minho, Guimarães, Portugal
cfp@dsi.uminho.pt

Abstract. With the growth of IoT (Internet of Things) technologies, there has been a significant increase in opportunities to enhance various aspects of our daily lives. One such application is the prediction of car park occupancy using car park movement data, which can be further improved by incorporating weather data. This paper focuses on investigating how weather conditions influence car park occupancy prediction and aims to identify the most effective prediction algorithm. To achieve more accurate results, the researchers explored two primary approaches: Classification and Regression. These approaches allow for a comprehensive analysis of the parking scenario, catering to both qualitative and quantitative aspects of predicting car park occupancy. In this study, a total of 24 prediction models, encompassing a wide range of algorithms were induced. These models were designed to consider various details, including parking features, location specifics, time-related factors and crucially, weather conditions. Overall, this study showcased the potential of leveraging IoT technologies, car park movement data, and weather information to predict car park occupancy effectively. By exploring both classification and regression approaches, each yielding accuracy and R2Score values surpassing 85%.

Keywords: Smart Cities, Data Mining, Parking Lot.

1 Introduction

The amount of data generated every day is staggering. With the rise of digital technologies and the Internet of Things, we now produce vast amounts of data both in a digital context and in our day-to-day activities. This data both presents a significant challenge for storage and management, and a tremendous opportunity for innovation and problem-solving. In combination with this, due to the fast-paced of the world, efficient urban planning has become crucial, and predicting the occupancy of parking lots has emerged as an essential aspect of this process. According to a study by [1], in Stuttgart, Germany roughly 15% of traffic on busy city streets is caused by drivers searching for a vacant parking spot. Moreover, drivers in dense city districts usually spend from 3.5 minutes to 15.4 [1] minutes looking for a parking spot, which means spending money and producing pollution, thus affecting the general societal costs.

Predictive models can assist parking lot managers in making more informed decisions and adjusting parking availability and pricing to meet changing demand patterns.

ANEXO III – ARTIGO 'PERVASIVE REAL-TIME ANALYTICAL FRAMEWORK—A CASE STUDY ON CAR PARKING MONITORING'

Artigo que apresenta uma visão geral de todo o processo até à criação da *framework* OLAP.

Estado: Publicado

Local de publicação: MDPI (*Multidisciplinary Digital Publishing Institute*)

Referencia: Francisca Barros, Beatriz Rodrigues, José Vieira, Filipe Portela (2023), Pervasive real-time analytical framework - A case study on car parking . Information - Information for Business and Management-Software Development for Data. Volume 14, Issue 11, 584. MDPI. DOI:10.3390/info14110584

information

MDPI

Article

Pervasive Real-Time Analytical Framework—A Case Study on Car Parking Monitoring

Francisca Barros [1], Beatriz Rodrigues [1], José Vieira [2] and Filipe Portela [1,2,*]

[1] Algoritmi Centre, University of Minho, 4800-058 Braga, Portugal
[2] IOTECH, 4785-588 Trofa, Portugal; josevieira@iotech.pt
* Correspondence: cfp@dsi.uminho.pt

Abstract: Due to the amount of data emerging, it is necessary to use an online analytical processing (OLAP) framework capable of responding to the needs of industries. Processes such as drill-down, roll-up, three-dimensional analysis, and data filtering are fundamental for the perception of information. This article demonstrates the OLAP framework developed as a valuable and effective solution in decision making. To develop an OLAP framework, it was necessary to create the extract, transform and load the (ETL) process, build a data warehouse, and develop the OLAP via cube.js. Finally, it was essential to design a solution that adds more value to the organizations and presents several characteristics to support the entire data analysis process. A backend API (application programming interface) to route the data via MySQL was required, as well as a frontend and a data visualization layer. The OLAP framework was developed for the ioCity project. However, its great advantage is its versatility, which allows any industry to use it in its system. One ETL process, one data warehouse, one OLAP model, six indicators, and one OLAP framework were developed (with one frontend and one API backend). In conclusion, this article demonstrates the importance of a modular, adaptable, and scalable tool in the data analysis process and in supporting decision making.

Keywords: framework; OLAP model; ioCity; ioScience; business intelligence; data warehousing

check for updates

Citation: Barros, F.; Rodrigues, B.; Vieira, J.; Portela, F. Pervasive Real-Time Analytical Framework—A Case Study on Car Parking Monitoring. *Information* **2023**, *14*, 584. https://doi.org/10.3390/info14110584

Academic Editor: Kostas Vergidis

Received: 26 July 2023
Revised: 16 October 2023
Accepted: 17 October 2023
Published: 25 October 2023

1. Introduction

The current volume, variety, and velocity of data production pose significant challenges for industries. The sheer amount of data being generated requires efficient storage and management solutions. The diverse types of data (structured, semi-structured, and unstructured) complicate the extraction of meaningful insights. Additionally, the fast-paced nature of real-time data streams demands quick processing capabilities. To address these challenges, an efficient OLAP framework is essential, enabling organizations to perform multidimensional analysis, gain insights, and make informed decisions in the face of the overwhelming data landscape.

Online analytical processing (OLAP) is a technical approach that allows users to analyze relevant information from different points of view and in various combinations. By connecting to a database, the system enables the visualization of statistical reports that allow extracting, consulting, and retrieving data [1].

The end-users usually do not know how to easily create dashboards that interact with the data, or of the existence of open-source platforms that are able to do that. For example, through cube.js, an OLAP model can be derived. Cube.js is an open-source headless business intelligence (BI) that allows pre-calculating and pre-aggregating data, creating an OLAP cube that makes analysis faster. Cube.js enables the connection to several databases, and consequently, the schema is defined for each dimension and fact table in several cubes. The foreign keys and the relationships between the other cubes are limited for fact tables, and the facts are calculated. We can save the desired combinations through pre-aggregations for faster analysis [2]. To test it, a proof of concept was developed using a