CIDAI Centre of Innovation for Data tech and Artificial Intelligence

# Sustainable micro-mobility

## Mobility services for bike-sharing systems

Amb la collaboracio de

Ajuntament de Barcelona

bicing

PEDALEM barcelona!

CIDAI-PAI 01-2022-D06



Suggested citation: Sustainable micro-mobility: mobility services for bike-sharing. Final report. Ref.: CIDAI-PAI 01-2022-D06.

# Executive Summary

The project "Sustainable micro-mobility: Mobility services for bike-sharing systems" has been executed within the CIDAI's (Centre of Innovation in Data Technologies and Artificial Intelligence) activity plan 2022. Companies Barcelona de Serveis Municipals (BSM) and Pedalem have sponsored the project by defining a set of needs concerning the Barcelona de Serveis Municipals (BSM) service, by providing and sharing the entire set of data used in the various work packages developed to address and solve these needs and by steering the project towards this objective. CIDAI members Eurecat, i2CAT and IDEAI-UPC have conducted all the work, under Eurecat's management. All participants signed a Memorandum of Understanding at the project inception whereby all parties committed for a safe end ethical usage of the shared data.

This project harnesses data-driven, computational methodologies to extract valuable insights within micro-mobility systems, with a specific focus on Barcelona's bike-sharing system, Barcelona de Serveis Municipals (BSM). The first objective is to use advanced artificial intelligence techniques to identify abnormal demands in the Barcelona de Serveis Municipals (BSM) stations of the Barcelona city that could be associated with unexpected events in the neighbourhood. Furthermore, explainable artificial intelligence techniques are used to automatically interpret the reasons of the abnormality and provide relevant information to the Barcelona de Serveis Municipals (BSM) manager to understand the reasons of the prediction and support decision-making. The second objective focuses on creating a system capable of forecasting the availability of electric bicycles and their corresponding battery levels, shedding light on electric bike mobility patterns. The third objective aims to develop an artificial intelligence system for predicting the lifetime cycle of bike components to streamline maintenance procedures, ensuring seamless operations and optimal user experiences. And finally, the fourth objective is to develop a proof of concept for a Mobility Data Space platform that aggregates mobility-related data from Barcelona. This platform enables the integration of diverse data sources, supporting the development of analytical and computational applications, including AI models, while adhering to European and local standards and regulations.

This project successfully demonstrates the potential of data-driven methodologies for sustainable urban mobility, particularly within Barcelona's Barcelona de Serveis Municipals (BSM) system. Several important developments have been made: The project has developed a data space that collects and organizes data related to the bike-sharing system. The applications developed, predictive models, dashboard, and API are accessible in the Data Space App Store. Through machine learning, intelligent data visualization and explainable AI the project has made significant progress in predicting electric bike battery levels, mobility patterns and abnormalities in the mobility patterns. Two distinct approaches have been explored for mobility modelling: a data-driven method and Markov chains, offering both performance and scenario-building capabilities. The project's use of survival analysis has yielded excellent results for predicting maintenance needs and enhancing safety and reliability within the bike fleet. To make data and model insights accessible, an API and a dashboard have been developed, showcasing key metrics and facilitating access to the information. In regards of the detection of abnormal behaviours a combination of machine learning based on graphs, non-parametric advanced statistical inferential models, case-based reasoning for prediction, and explainable AI techniques have been successfully used to anticipate the need of installing virtual stations in specific locations of the city at specific schedules and dates and creating an Intelligent Decision Support System to recommend when and where these virtual stations are required.

This project represents an initial step in realizing the potential of AI and big data solutions to extract actionable knowledge from micro-mobility data and enhance sustainable urban mobility. Combining data-driven insights with predictive models, explainable AI and decision-support systems makes eco-friendly modes of transportation, like bike-sharing systems, more efficient and reliable, and ultimately aims to promote sustainability and enhance urban commuting.

# Introduction

2

In the urban mobility landscape data-driven solutions have emerged as a pivotal asset offering the potential to positively impact on the way we manage transportation systems. At the heart of this journey lies Artificial Intelligence (AI) techniques, which play a significant role in driving the development of mobility services such as decision-support systems. These AI-driven solutions aim to promote a more sustainable and multimodal urban mobility, effectively addressing one of the most critical challenges in modern city management. AI technologies are instrumental in processing and analysing the vast amounts of data generated by urban transportation systems. Machine learning algorithms, for instance, can predict traffic patterns, optimize public transit routes, simulated mobility scenarios and even enable autonomous vehicles to navigate cities safely. This synergy between AI and urban mobility can foster efficiency, lower emissions, detect and interpret unexpected behaviour of the cities and ultimately improve the quality of life for urban residents.

Data serves as the core asset in the development of these AI systems, and the concept of data spaces plays a crucial role in facilitating their effectiveness. The concept of data spaces originated from the European Data Strategy, which was unveiled by the European Commission in 2020. This strategy aims to establish a unified data market that not only boosts Europe's global competitiveness but also upholds its data sovereignty. The European Commission has proposed the development of nine common data spaces, each tailored to a key sector of the European economy. Among these, the mobility data space stands out as a particularly significant initiative. This data space is poised to facilitate data exchange between public administrations, public transport providers, light mobility services, and other stakeholders in the urban mobility ecosystem. In this project AI technologies and Data Spaces are combined to support the management of bike-sharing system in Barcelona with the goal of enhancing the efficiency of the service and the overall urban experience for citizens.

In this context, the project has developed a proof of concept of a Mobility Data Space platform that incorporates data from the bike-sharing system, Bicing, relevant for understanding micro-mobility in Barcelona. The availability of data from Bicing within this space will facilitate the integration of this data and the development of Artificial Intelligence applications safely, directly and in compliance with European and local regulations. We propose to integrate into the platform the data necessary for the implementation of the following use cases: prediction of battery levels and the predictive maintenance of bike parts. Once the data is integrated within the data space, CIDAI partners develop predictive models capable of predicting in advance the status of batteries and potential preventive maintenance needs related to bikes. These models and applications are then made available in the data space's App Store, from where they can be directly installed and executed within the data space without the data leaving its secure boundaries. In this way, it is possible to optimize the logistics and availability of electric bicycles in these situations.

On the other hand, a second use case is related with the prediction of abnormal behaviours of the Bicing stations and its automatic interpretation based on the online access to public open data like the agenda of the city. An independent intelligent decision support system for the recommendation of where and when to install virtual stations in the city of Barcelona has been built, with online access of the public agenda of the city to be used for the explainable AI capabilities of the system.

This document will provide an overview of the implemented methodologies and the conducted work, describing the defined use cases, the design of the data space, the predictive models developed, the automatic interpretation of predictions and construction of explanations, the construction of the intelligent decision support system, and analysis of the obtained results. Furthermore, this report will also highlight potential improvements and will discuss the recommendations in case of potential scale up of the project.

# Project scope and objectives

The overall scope of the project is to leverage data-driven, computational and artificial intelligence methodologies for extracting valuable insights within micro-mobility systems, focusing particularly on Barcelona's bike-sharing system, namely Bicing.

The project aims to develop a proof of concept for a Mobility Data Space platform that gathers mobility-related data from Barcelona. This platform intends to enable the integration of diverse data sources, fostering the development of analytical and computational applications, including AI models, in compliance with European and local standards and regulations. The integrated data is used to develop predictive models for preventive bicycle maintenance, ultimately optimizing electric bicycle logistics and availability.

A proof of concept has been implemented that combines different relevant data sources into a platform, identifying and integrating data from both internal and external sources (administrative, open, and data from the bike-sharing system) and the real time connection between the external open data sources and the system. The integrated data is used to develop predictive models for managing virtual stations during demand peaks and supporting decision-making for alert triggers and preventive bicycle maintenance, ultimately optimizing electric bicycle logistics and availability. In addition, it generates alerts and related recommendations, provides automatic explanations of the proposed recommendations, and support the construction of an intelligent decision-support system that helps decision-makers in their virtual stations policies.

Within this context, four overarching technical objectives have been outlined:

- TO1. Special events in the city: development of a predictive framework capable of anticipating the mobility needs of planned and unexpected events in the urban space. This predictive tool is pivotal in addressing scenarios that necessitate the strategic allocation and management of virtual stations.
- TO2. Electric bikes and battery levels: formulation of a system capable of forecasting the availability of electric bicycles and their corresponding battery levels. This system shed light in the mobility patterns of electric bikes and the eventual need to activate protocols of relocation or maintenance of electric bikes.
- TO3. Predictive maintenance systems for bikes: development of an artificial intelligence system for predicting the lifetime cycle of bike pieces. The key to streamlining maintenance procedures for the extensive bicycle fleet, thus ensuring seamless operations and optimal experiences for the users.
- TO4. Mobility Data Space: deployment of a data space that does not only accommodate the data of the bike-sharing system but also encompass other pertinent mobility-related data points relevant to the Barcelona landscape. By amalgamating various data sources, this Data Space will serve as a repository for enhancing urban mobility insights.

This project stands as a proof of concept of different computational methodologies applied to a very practical demands of sustainable urban mobility, exemplified through Barcelona's bike-sharing system, Bicing. Through the accomplishment of these objectives, the project endeavours to contribute to conceptualize data-driven methods, tools, and services for micro-mobility management paradigms.

# Work performed

# 4

## 4.1. WP1 Project coordination

This package is designed to ensure efficient project coordination. It encompasses a range of tasks and responsibilities dedicated to overseeing project activities, monitoring progress, presenting results, and streamlining the closure process. The primary tasks include:

1. Developing a project plan: this involves creating a well-structured project plan that outlines timelines, key milestones, and efficient resource allocation.
2. Conducting regular project team meetings: the goal here is to foster effective communication and collaboration among project team members. These meetings will ensure alignment with project objectives and strategic goals.
3. Finally, reporting on project outcomes and future directions: the package includes the responsibility of delivering a comprehensive report on the project's outcomes and providing insight into future directions.

## 4.2. WP2 Data sources Identification and implementation of Data Spaces infrastructure

### T2.1 Data sources identification

First of all, it was necessary to establish the origin, composition, and volume of all source data. Since the source was unique (Bicing-Pedalem), we used templates to obtain the following information for each source:

- Source Type: Database, files, API, etc.
- Format: SQL, REST, CSV, etc.
- Estimated Volume: KBs, MBs, GBs, etc.
- Update Frequency: Real-time, daily, weekly, etc.
- Source: Cloud, VM, file system, etc.
- Access Data
- Access Strategy: Push/pull

### T2.2 Architecture proposal

Once we had defined the data to be used, the next step was to propose a solution to share this data with the other components of the solution, using a data exchange system based on Data Spaces.

The final solution was implemented based on the IDS-RAM 4.0 reference architecture[1]. The solution includes a provider connector responsible for supplying the data from Pedalem, a consumer connector that receives this data through Data Spaces protocols, and an Authentication Authority (DAPS) to ensure the privacy and security of communications. On the other hand, the developed data analytics, and AI models as well as dashboards were made available as applications within a data space App Store. These applications are installed from the App Store and executed through the same consumer connector.

1 https://github.com/International-Data-Spaces-Association/IDS-RAM_4_0

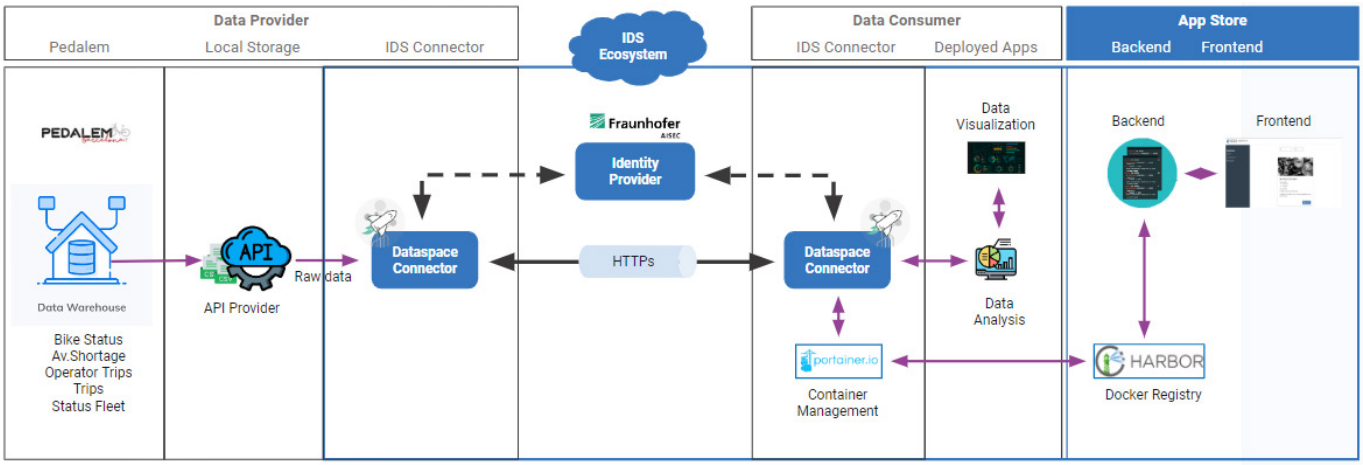



Figure 1. Architecture and components of the implemented Data Space solution.

## T2.3 Data Spaces Implementation

As seen in the Architecture proposal, Bicing's data is accessed through an API. This API is configured with two endpoints for each data source: one for listing the files available for each source and another for retrieving the data from a specific file. These endpoints are registered in the catalogue of the Provider Connector, which is located on the same machine as i2CAT. We can see a sample of them:

On another machine, we have deployed the Consumer Connector, which is responsible for communicating with the Provider Connector to obtain Pedalem's data. On this same machine, we have the App Store where all applications (models, dashboards, APIs, databases, etc.) are stored. The original idea was that when you want to use the models, the Consumer Connector requests the necessary applications to be deployed from the App Store to enable the use of the dashboard and execute the necessary models.

**List bike state battery csv available files**

GET `/api/pedalem/list_bike_states_batt` List Bike States Batt

**Get bike state battery csv data from specified filename**

GET `/api/pedalem/get_bike_states_batt` Get Bike States Batt

**List average shortage files**

GET `/api/pedalem/list_average_shortage` List Average Shortage

**Get average shortage csv data from specified filename**

GET `/api/pedalem/get_average_shortage` Get Average Shortage

Figure 2. Provider Connector endpoint list.

## 4.3. WP3: Definition and implementation of the use case Virtual Stations

The objective of this WP is to prove the viability of training an intelligent system able to understand the normal pulse of the mobility in a city and detect unexpected events, which represent a challenge in the management of the Bicing service. Indeed, events occurring in the city (either planned or not) can unexpectedly provide abnormally high demand of bikes in certain stations of the city or too many bikes arriving, so that the station overflows or remains empty and the users cannot flow normally. The celebration of concerts, sportive events, big congresses, are among the planned events that might broke the availability of either bikes or anchorages in a certain station, and moreover, when the users do not find bike or place to park the bike in their preferred station, they move to the closest one, so that after few time the entire zone is affected by the phenomenon. It is even worse when the event is non planned, unexpected (traffic constraints, demonstrations, failure of some convoy from public transportation, etc). Being able to prevent this situation with at least one day anticipation can help the company to plan the installation of virtual stations in the required places, so that the availability of the service can be guaranteed. In this case study, we will try to build the corresponding predictive model, and going further, to interpret the meaning of unexpected events and raising informed alerts and recommendations with sufficient anticipation so that extra virtual stations can be properly installed in the right place at the right time to guarantee the service of the Bicing system and support the extra demand of the unexpected event.

A virtual station is a temporary delimited space located on the public road for a certain period of time in an area where there is an exceptional demand for the Bicing service, mainly due to an event that concentrates a large number of users. The virtual station can be deployed at the beginning of the reference event, so that the bikes can be left by the people when there is no empty anchors in the nearby stations, or at the end of the event, that is once the event has finished, to cover the pick of bikes demand occurring after the event. Virtual stations are a very valuable resource for micro-mobility and offer a solution at certain critical moments, but at the same time they represent an organizational and logistical challenge that this project wants to face, by providing relevant information in advance about where and when it is convenient to plan the installation of a virtual station. The relevant added value of this use case is that the detection of virtual station needs will be done automatically, only based on mobility data and additional open data on the agenda of the city devoted to automatically interpret and explain the need of installing a virtual station.

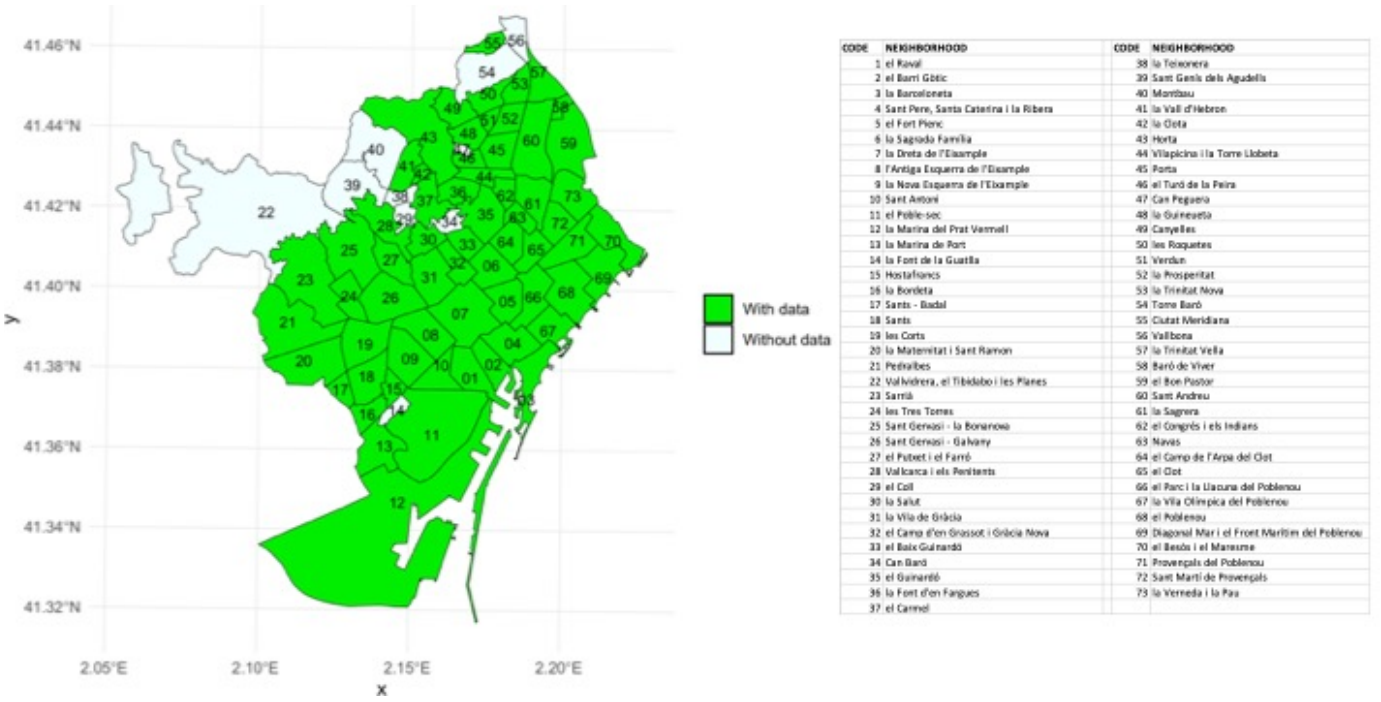
### T3.1. Use case definition and goals

In this task, the team identified and clearly defined the specific use case that needed to be addressed. This involved understanding the context, the goals, and the challenges associated with the bike-sharing system under study. The analytics objectives were also refined during this phase, outlining what the team aimed to achieve through computational models based on the availability of the data. Once we identified the challenges and data available, the use case was defined as:

1. Finding a learning model to recognize a scenario of mobility as normal or abnormal.
2. Building a predictive model to anticipate abnormal behaviour at a certain place and time slot.
3. Building the IDSS that provides the recommendations associated with abnormal expected behaviour.
4. Include explainability capabilities in the IDSS according to the open information about city activities.

### T3.2. Relevant data identification, data collection preprocessing and metainformation

In this task the relevant data sources where identified. After a review of the available data at Pedalem regarding the goals of the use case, we have been analysing data from three different sources: Data from bikes and stations, data from users and general information. For the use case developed in this WP, the relevant information is basically contained in the track of *data from bikes and stations*. This kind of data is useful to understand the dynamics of mobility generated by users.




| CODE | NEIGHBORHOOD | CODE | NEIGHBORHOOD |
|---|---|---|---|
| 1 | el Raval | 38 | la Teixonera |
| 2 | el Barri Gòtic | 39 | Sant Genís dels Agudells |
| 3 | la Barceloneta | 40 | Montbau |
| 4 | Sant Pere, Santa Caterina i la Ribera | 41 | la Vall d'Hebron |
| 5 | el Fort Pienc | 42 | la Clota |
| 6 | la Sagrada Família | 43 | Horta |
| 7 | la Dreta de l'Eixample | 44 | Vilapicina i la Torre Llobeta |
| 8 | l'Antiga Esquerra de l'Eixample | 45 | Porta |
| 9 | la Nova Esquerra de l'Eixample | 46 | el Turó de la Peira |
| 10 | Sant Antoni | 47 | Can Peguera |
| 11 | el Poble-sec | 48 | la Guineueta |
| 12 | la Marina del Prat Vermell | 49 | Canyelles |
| 13 | la Marina de Port | 50 | les Roquetes |
| 14 | la Font de la Guatlla | 51 | Verdun |
| 15 | Hostafrancs | 52 | la Prosperitat |
| 16 | la Bordeta | 53 | la Trinitat Nova |
| 17 | Sants - Badal | 54 | Torre Baró |
| 18 | Sants | 55 | Ciutat Meridiana |
| 19 | les Corts | 56 | Vallbona |
| 20 | la Maternitat i Sant Ramon | 57 | la Trinitat Vella |
| 21 | Pedralbes | 58 | Baró de Viver |
| 22 | Vallvidrera, el Tibidabo i les Planes | 59 | el Bon Pastor |
| 23 | Sarrià | 60 | Sant Andreu |
| 24 | les Tres Torres | 61 | la Sagrera |
| 25 | Sant Gervasi - la Bonanova | 62 | el Congrés i els Indians |
| 26 | Sant Gervasi - Galvany | 63 | Navas |
| 27 | el Putxet i el Farró | 64 | el Camp de l'Arpa del Clot |
| 28 | Vallcarca i els Penitents | 65 | el Clot |
| 29 | el Coll | 66 | el Parc i la Llacuna del Poblenou |
| 30 | la Salut | 67 | la Vila Olímpica del Poblenou |
| 31 | la Vila de Gràcia | 68 | el Poblenou |
| 32 | el Camp d'en Grassot i Gràcia Nova | 69 | Diagonal Mar i el Front Marítim del Poblenou |
| 33 | el Baix Guinardó | 70 | el Besòs i el Maresme |
| 34 | Can Baró | 71 | Provençals del Poblenou |
| 35 | el Guinardó | 72 | Sant Martí de Provençals |
| 36 | la Font d'en Fargues | 73 | la Verneda i la Pau |
| 37 | el Carmel | | |

Figure 3. Neighbourhoods of Barcelona city involved in the prove of concept.

With this information it will be useful to represent the mobility of the city by using the Bicing service and understand the activities of arrivals and departures in the different stations around the city. Data provided by the company includes the following period: Abril 2019 to Decembre 2022. From the 73 neighbourhoods of the city of Barcelona, data provided is a sample related with 63 neighbourhoods (Fig. 3). Those missing correspond either to very peripheric neighbourhoods or those containing important mountains like Carmel Mountain.

Because of the COVID and lockdown, the behaviour of the city regarding Bicing was disrupted in 2020, and this makes difficult to ensure that the mobility patterns from 2019 are the same as those spontaneously followed after the pandemics. For this reason, this prove of concept will consider data from 2021 and 2022.

### Data describing the Bicing stations:

Specifically, the data from the stations will be used to geolocalize the stations to model the demand flows:

- Station Id
- Name of the Station
- Capacity: Total number of anchors of the station
- Location (latitude, longitude and altitude)
- Postal address of the station
- City
- Zip Code
- District where the station is located.
- Neighbourhood where the station is located.
- Phase:

### Register of Bicing trips:

Also, for each station it will be relevant to know the history of bicycles that stop there. The company has a database of trips that contains information relevant to our purposes: the internal historical data about the trips registered in the Bicing. This database contains trips in the rows and for each trip the information provided is:

- Trip id: Identification of the trip
- Bike id: Identification of the bike
- Bike model: Model of bike
- Customer id: Identification of the bike user
- Start time: Date and Instant of beginning of the trip.
- Start Station id: Which is the bike station where the trip starts.
- End time: Date and Instant of ending the trip.
- End Station id: Which is the bike station where the trip finishes.

### Additional open data:

Also, when abnormal behaviour of a station will be identified we will be interested in explaining the reasons of these anomaly, so that additional open data about activities planned in the city will be also considered to relate to the mobility data for the ex-

plainable functionality of the system. After a first screening, some relevant data sources have been identified:

Agenda for events and daily activities of the Barcelona city: https://opendata-ajuntament.barcelona.cat/data/en/dataset/agenda-diaria it contains the events with the beginning date and ending date as well as the scheduled timings among others:

- Name of event
- Beginning date of the event
- Ending date of the event
- Address of the event venue
- Neighbourhood of the event venue
- District of the event venue
- Coord X: Geographical coordinate of the event venue
- Coord Y: Geographical coordinate of the event venue
- Status: Pending, Held, Held in virtual format, Held with restrictions, Cancelled.

Agenda of sportive activities: https://opendata-ajuntament.barcelona.cat/data/en/dataset/agenda-activitats-esportives It contains data about the sportive events that provide relevant information, but do not include the dates of celebration of the event and this makes non useful this database to our purposes.

Data related with the festivals celebrated in Barcelona city: https://opendata-ajuntament.barcelona.cat/data/en/dataset/dades-festes-ciutat It contains relevant information about the equipment ownership, the topic of the equipment, the event organizer, dates of the event, etc. Only data from 2022.

Data related with the Mercè festival in Barcelona city: https://opendata-ajuntament.barcelona.cat/data/en/dataset/dades-merce Created in 2021 it contains information about the events in the program of the La Mercè patronal festival in Barcelona, including the number of assistants to the activity.

## T3.3 Data exploration and determination of characteristic variables

After analysing all available data sources, and regarding the main goals of this WP, the databases of trips and stations will be combined in order to include the geolocalization of the starting and ending stations each trip, by using them as primary keys to connect both tables. Including the geolocalization of Bicing stations in the working database opens the door to make visualizations of the mobility flows with geographical information by adding the possibility to have visual insights to the end users of the tool.

On the other hand, according to the goals of the case study, there is no need to use all available variables in the trips database. We will approach the problem by learning the normal behaviours of the stations along the city through advanced machine learning techniques and developing a formal way to detect when a certain day the behaviour is too far away from normal. And for this, it is enough to associate the behaviour of the station with the trips starting of finishing there. So that, dates, origin and destination stations of the trip with their corresponding geolocalizations will be relevant to this purpose at this stage.

Regarding the open data with information about events After a first analysis of the open data sources, the one with an API available to be consulted online and in real-time with more guarantees to take advantage of the information contained is the agenda of daily activities of the Barcelona city. Thus, for the present prove of concept the Barcelona agenda will be used.

### Knowledge representation model

From all available data, and once detected the relevant information needed to model the use case, we will consider the possibility to represent the behaviour of the Bicing network through the concept of Scenario and its associated Mobility matrix as it will be defined bellow.

## T3.4 Development of predictive models on unexpected events

In this task, predictive models are developed based on the data and objectives defined earlier. This includes several steps that will be developed in the following section.

1. A methodology to learn on the normal behaviour of the Bicing stations in the city. Here it is relevant to be based on the formalization of a Scenario provided in previous section. Also, the mobility graph associated to a scenario will be an essential element to build this component and identify the normal MS associated to a given scenario S.
2. A technique to build an indicator to identify when the behaviour of a certain station is too far from normality, given the historical mobility data. Here, non-parametric distributional characteristics of the behaviour of a scenario will be built, and Fisher permutation tests will be used to identify when a given observed behaviour for a certain scenario is significantly abnormal and this will be used to tag unexpected events. This step is very relevant, since it makes possible to generate supervised data based on unsupervised data, by inducing a learned indicator for the unexpected events in a given neighbourhood, like concerts, demonstrations, or popular cultural events that can alter the normal trends of the Bicing station on a certain time slot. Inducing a class variable opens the door to build a classifier to predict its values.
3. A method to predict unexpected events with sufficient anticipation to react on time. In this case case-based reasoning techniques will be used to build a predictive model for the unexpected event of tomorrow given the data from today.

## T3.5. Design and implementation of the Intelligent Decision Support System:

In this task the previous components will be encompassed with other cognitive, visualization and reasoning components required for the IDSS.

1. First, the Architecture of the IDSS will be designed.
2. Next, the components required to surround the predictive models developed in previous task will be developed and tested. This includes, GUI, data visualization, reasoning and explainability components.
3. The Intelligent decision support system that raises alerts and recommendations about virtual stations installation encompassing all previous components in a global system that can integrate everyday observations, with the predictive models built and the reasoning and visualization components, so that the user can receive information about the need of install virtual stations one day in anticipate and some explanations about the reason why these recommendations are provided.

## T3.6 Semantic interpretation and explainability components

This is a special component that requires a dedicated task as it confers the system a differential capability.

In this task a semantics interpretation component will be developed. It is a cognitive component that provides automatic conceptual explanations about the reasons to forecast an unexpected event that will use de real time online connection with the agenda of the city and provides associated information to answer simple questions from the user about the technical reasons for declaring a scenario as significantly abnormal.

## T3.7. Validation of results

The models are trained on a portion of the data and validated to ensure they perform well on unseen data. Model validation techniques are used to assess several metrics like global accuracy, and sensitivity and specificity to identify unexpected events. The performance of the entire IDSS will be tested, component by component and at the integration level.

# 4.4. WP4 Definition and implementation of predictive mobility models

Overall, these tasks encompass the end-to-end process of using data analytics to solve a specific problem, from defining the problem and objectives, to collecting and preparing the data, conducting analysis and modelling, and finally presenting the results through a dashboard for effective communication.

## T4.1 Definition of the use case and the analytics objective

In this task, the team identified and clearly defined the specific use case that needed to be addressed. This involved understanding the context, the goals, and the challenges associated with the bike-sharing system under study. The analytics objectives were also refined during this phase, outlining what the team aimed to achieve through computational models based on the availability of the data. Once we identified the challenges and data available, we defined two use cases:

1. Predicting the mobility of e-bikes and the level of battery for each bike and station.
2. Predict the remaining useful life of bicycle parts.

## T4.2 Collection and pre-processing of relevant data and associated metadata

This task involved gathering the necessary data for the analysis. It included identifying the sources of data, collecting the data, and organizing it in a suitable format for further processing. In this case, the most important data came from Bicing. Pedalem shared with us data about trip data, station data, customer data, maintenance data, vandalism data, and batteries data. After the analysis of the different datasets only data on trips, stations, maintenance and batteries is used for building the different models. Data preprocessing was also conducted, which included data cleaning, data transformation, handling missing values, and ensuring data quality. This process was iterated until the team had access to correct data hosted in Pedalem servers.

## T4.3 Data exploration and feature selection

Data exploration involves conducting an in-depth analysis of the collected data. Exploratory data analysis techniques are applied to understand the characteristics of the data, identify patterns, outliers, and relationships between variables. Feature importance analysis is performed to determine which features have the most impact on the outcome or target variable. For instance, we consider variables such as distance, use time or altitude of the station have been considered for predicting the mobility patterns and levels of battery, which helps in modelling and improving the model's performance.

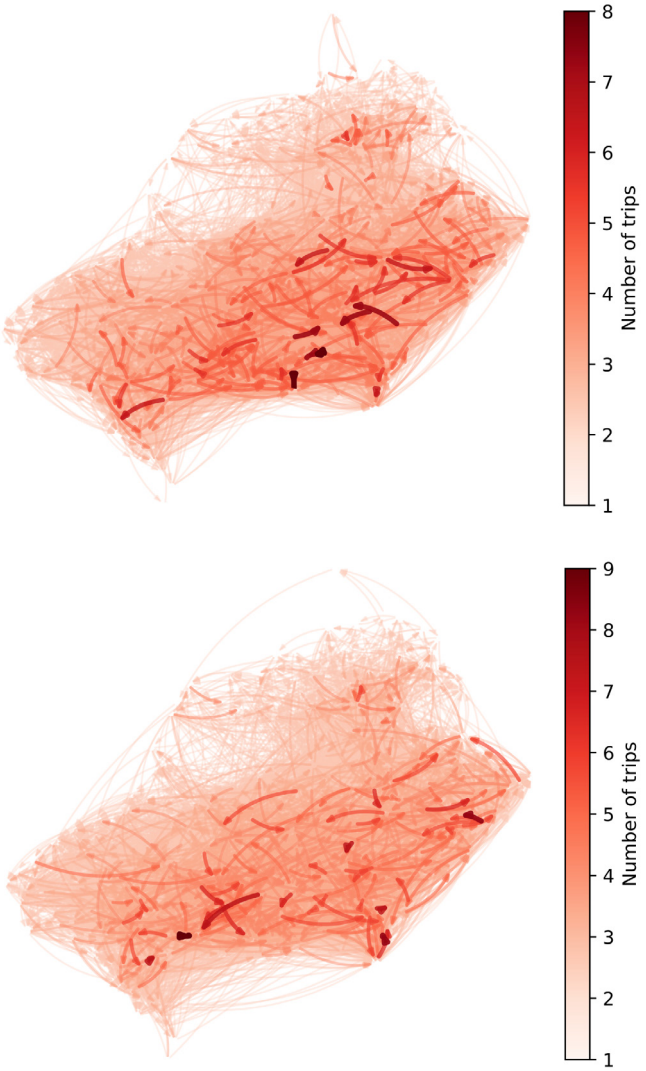



Figure 4. Number of trips between stations after six hours of simulations. In the left starting at 8:00 a.m. and in the right starting at 16:00 p.m.

## T4.4 Development and validation of predictive models

In this task, predictive models are developed based on the data and objectives defined earlier. This includes probabilistic models for predicting mobility, physics models for modelling battery levels and machine learning algorithms for predicting the remaining lifespan of bicycle parts. The models are trained on a portion of the data and validated to ensure they perform well on unseen data. This is especially critical with the battery data and maintenance data. Model validation techniques are used to assess the model's accuracy, generalizability, and potential overfitting. The models are refined and optimized iteratively to achieve the desired performance.

## T4.5 Development of dashboard

A dashboard is created to present the results of the analytics work in a visually informative and user-friendly manner. The dashboard includes visualizations and stats for the three models:

1. Micro-mobility models predictions of mobility and batteries
2. Markov models predictions of mobility and batteries
3. Predictive maintenance models.

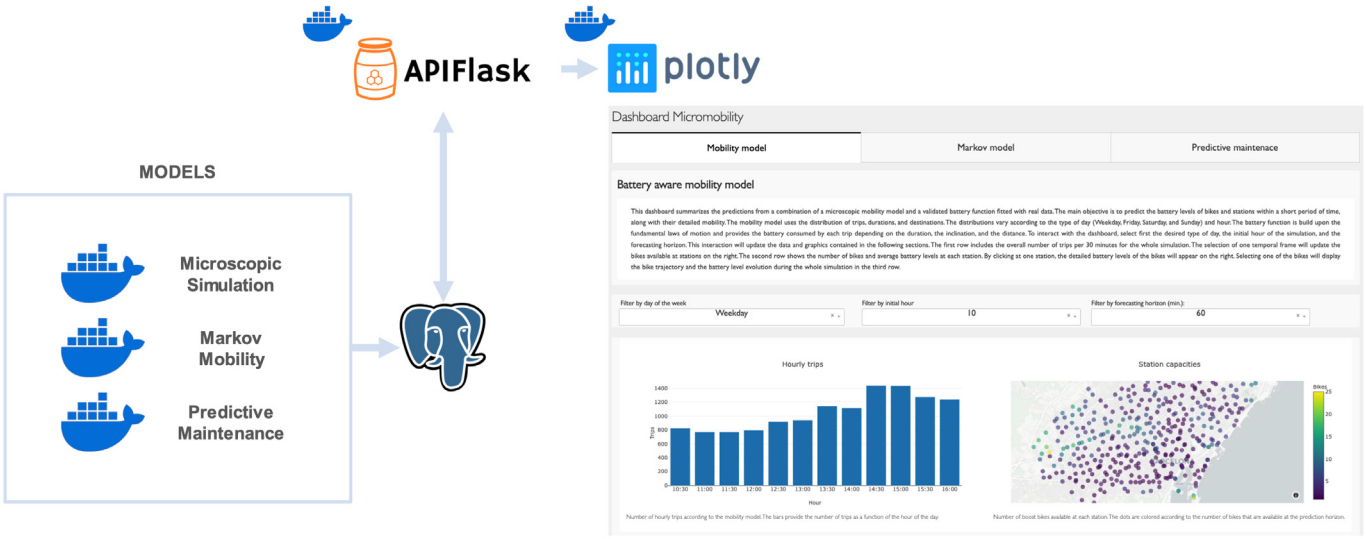



Figure 5. Deployment architecture for visualize model predictions, including predictive models, database, API and dashboard.

## 4.5. WP5 Integration and demonstrator

The objective of this point is to provide a detailed view of how the demonstration of the PAI Demonstrator for sustainable micro-mobility services applied to Bicing will be conducted.

In the first section, we can see an overview of the architecture, indicating which components will be deployed and where, as well as the interaction between them, along with a diagram of the proposal.

In the second section, we define the sequences between the different components, specifying the calls and results in each case.

### T5.1 Solution architecture

#### Locations

The components of the solution are distributed across different participants:

1. Eurecat: develops three predictive models, trained and persisted in Eurecat's servers where outcomes of the models also persisted. Two more applications, an API and a Dashboard are developed on Eurecat's side. All these five applications are dockerized and included in the App Store.
2. i2CAT: all Data Space components are deployed here, including the models used in the demo and the dashboard for visualization. In this case, there are two instances to simulate the provider connector (dockerBicing01.i2cat.net) and the consumer connector (dockerBicing02.i2cat.net).
3. IDEAI-UPC: generates models for the virtual stations use case.
4. Pedalem: their servers contain the source data for Bicing.

These various locations play specific roles within the solution architecture, facilitating different aspects of data processing and analysis.

#### Components

Next, we will detail each of the components, grouping them by the locations mentioned earlier:

| Pedalem | |
|---|---|
| Component | Description |
| Onedrive | File server where Pedalem shares data extractions from its systems. Currently, automatic access is not possible, and manual replication to the i2CAT Filesystem is performed. |

| EURECAT | |
|---|---|
| Component | Description |
| us-predictiu | Self-contained application in a Docker container where all the logic for generating the predictive model is contained. |

| i2CAT (dockerBicing01.i2cat.net) | |
|---|---|
| Component | Description |
| idsc-provider | Connector that acts as a data provider for Pedalem to the rest of the connectors. |
| api-provider | REST-API server responsible for accessing the source data. |
| DAPS | It is the Identity Provider, where component certificates are validated to determine whether they have access to the ecosystem or not. |

| i2CAT (dockerBicing02.i2cat.net) | |
|---|---|
| Component | Description |
| idsc-consumer | Connector that acts as a consumer and retrieves data from the provider. |
| ids-appstore | Application manager where we will register images of the different apps that we will use. |
| appstore-ui | App Store frontend where we can view registered applications. |
| dashboard-app | Dashboard where we will infer Bicing data with the models generated for the use cases. |
| filesystem | Files used to infer Bicing data with the models generated for the use cases. |
| us-predictiu | Applications that we will use from the dashboard to obtain the results of the predictive usage model. |
| estacions-virtuals[2] | Application that we will use from the dashboard to obtain the results of the virtual stations usage model. |

| i2CAT (External registry) | |
|---|---|
| Component | Description |
| harbor | Image repository where we will store the apps. |

2 This module is not integrated

## Diagram

Below, we can see the final proposal of the architecture to be implemented:

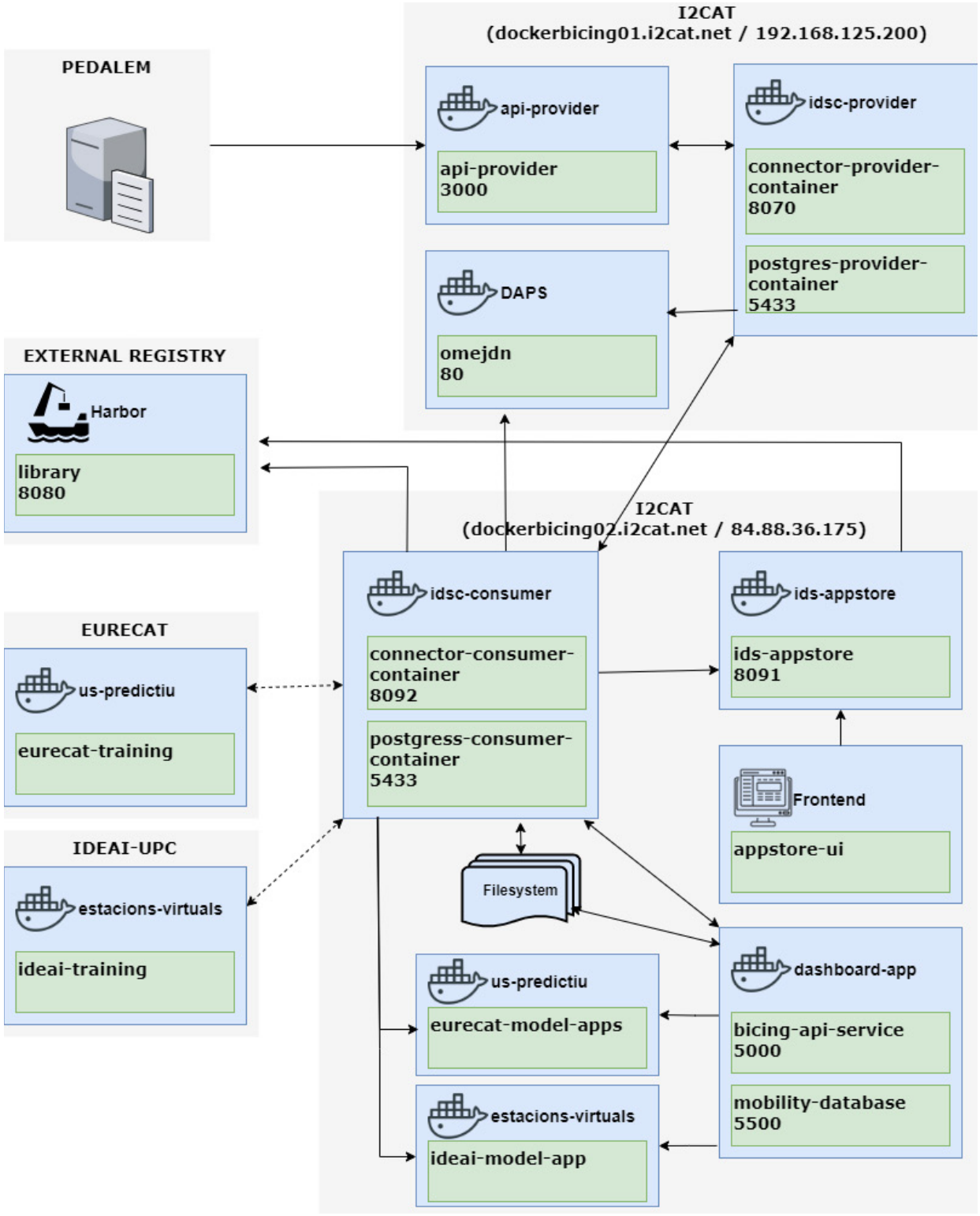



Figure 6. Deployment architecture of the integrated solution.

## T5.2 Interactions between components

### Database Deployment

The models require interacting with databases to query and persist their data. In this demo, the databases will be deployed on the same instance as the model applications, allowing them to be accessed locally. To expedite the demo, these databases will already be operational and loaded with data.

### Obtaining Bicing source data

The first of the interactions is to obtain the Bicing data, which is used both for training the models and for inferring results. To do this, the dashboard will call the consumer, which in turn will connect to the provider to get its catalogue.

Once we have the catalogue, we will search for the data we need and invoke the necessary provider artifact endpoints, which in turn will obtain the data through the API provider.

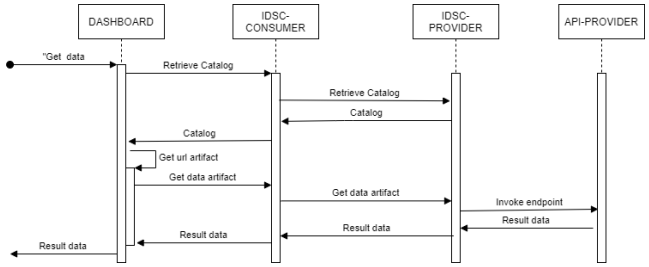



Figure 7. Iterations to obtain data from Pedalem.

### Deployment of applications

Once we have the data, we will ensure that the apps are deployed. To do this, we will check the catalogue of the App Store using the consumer, select the apps to use, and call them. The system automatically verifies if the app is deployed or not. If it is not deployed or is not the latest version, the app is downloaded from the app repository.

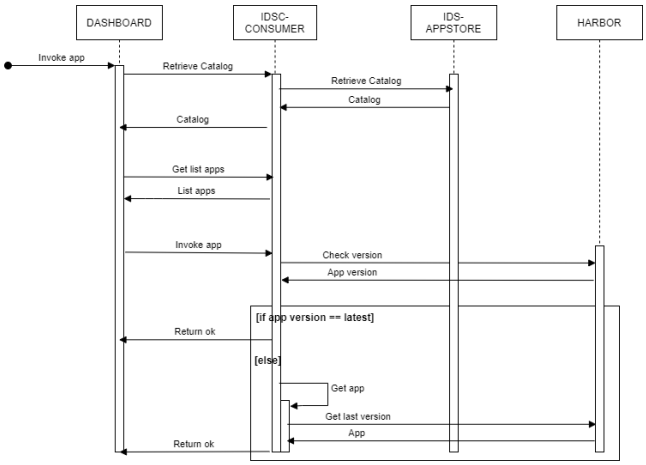



Figure 8. Application deployment diagram

### Model Visualization

Once we have the data and the apps deployed, we can invoke the apps to infer new data using the trained models and visualize the results on the dashboard.

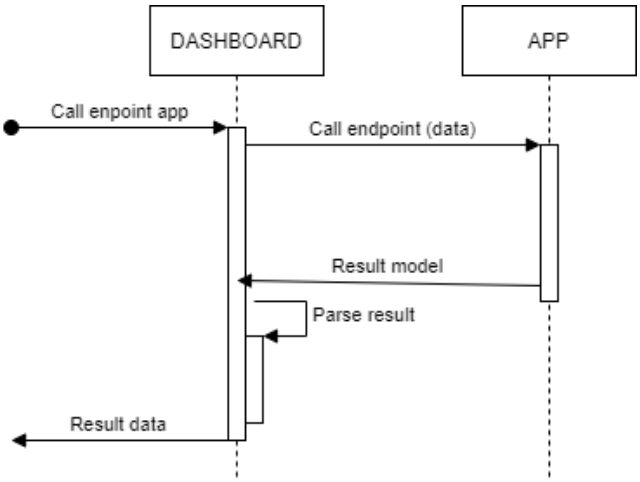



Figure 9. Interaction between dashboard and apps.

# Key results

5

## 5.1. Data Space

The proposed architecture has been implemented successfully in its entirety. The provider connector was originally intended to be installed on local machines at the source (Pedalem) but ended up being installed on an external machine, which resulted in asynchronous and manual data access. This process can be improved in the future by accessing the source files from the local machines in Pedalem directly or mounting a file mirroring server to automatically upload the files to the external machine where the connector is deployed.

On the other hand, the rest of the components have been used correctly, all communication has been carried out using a custom Identity Provider (DAPS), and the applications have been registered and deployed successfully in the App Store. This installation process required modifications to the Open-Source version of the App Store to ensure the correct communication with the connector, and the App Store was also supplemented with a GUI for easier user interaction.

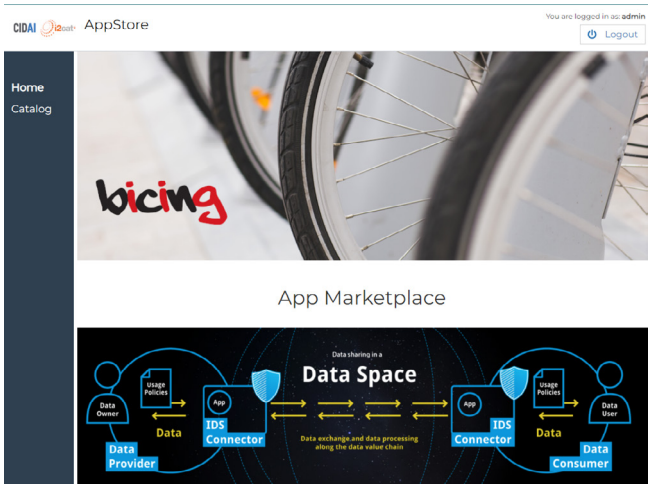



Figure 10. Main screen of the App Store frontend.

Once we have logged into the UI, we can access the catalogue, where we can see which applications are available to us:

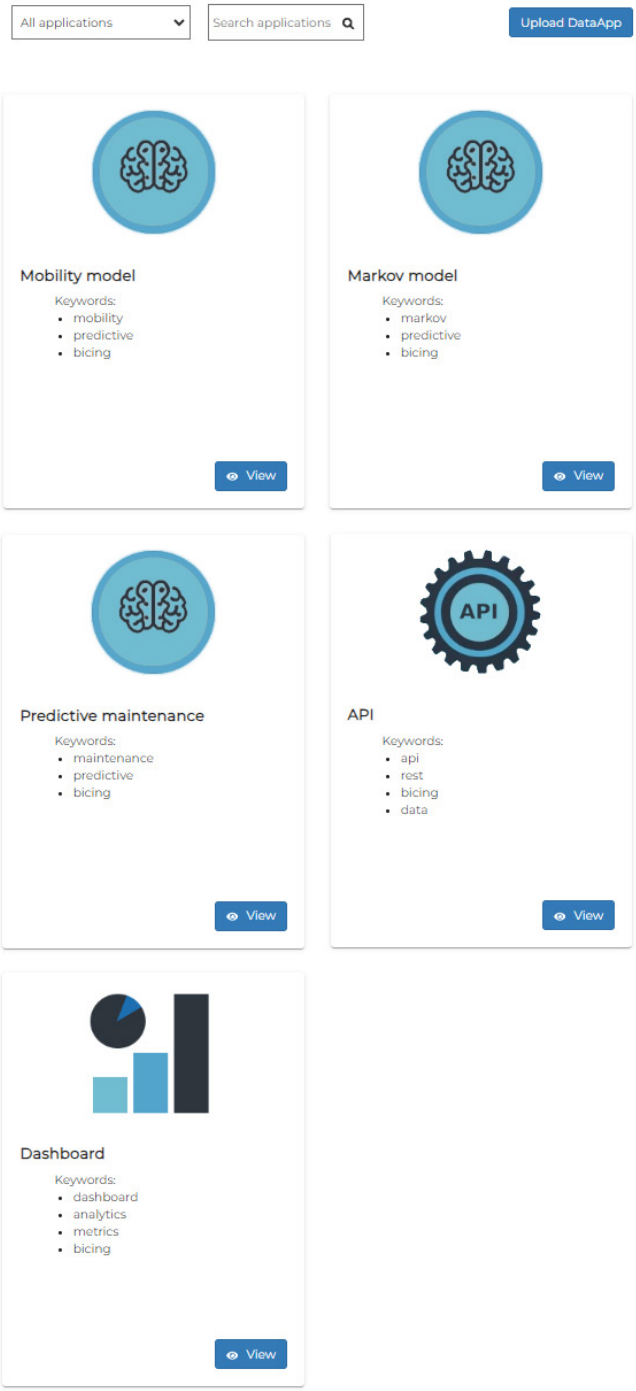



Figure 11. Catalogue with all available applications on the App Store.

In the catalogue listing itself, we can click on the 'Upload DataApp' button to access the application registration form. This process is not yet automated, but the necessary steps to complete the registration are provided.

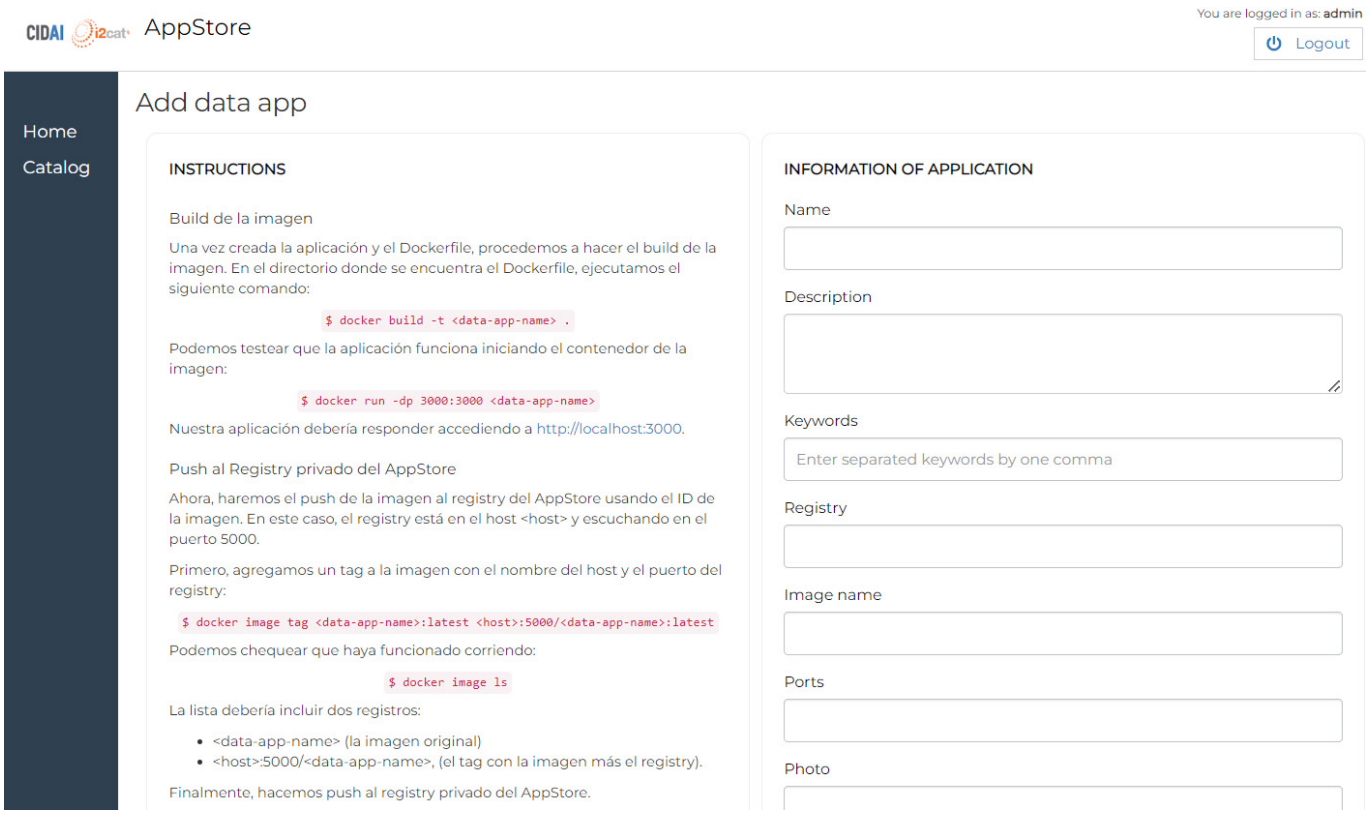



Figure 12. Form for registering an application on the App Store.

Due to the size of some applications and the volume of data from some source systems, both the databases and models have been deployed directly within the demo environment. Originally, the idea was to deploy everything from the App Store, but some processes took hours to execute and were not feasible for the demo.

The successful deployment of applications through a Data Space App Store and connector constitutes a significant milestone in the European Data Spaces ecosystem. Our deployment proves that the installation and execution of applications within the boundaries of a secure Data Space is feasible, thus further guaranteeing the privacy of the shared data and ensuring that the data is not shared outside of the Data Space boundaries.

## 5.2. The knowledge representation model for the unexpected events detection

As said before the input data for the WP3 is based on the fusion of databases on trips and geolocalization of stations and this information is going to be used to understand the normal pulse of the Bicing stations of the city and to build an IDSS that detects the unexpected events with sufficient anticipation to activate the protocols for installing virtual stations in the right locations and the right time zone, so that the continuity of the service can be guaranteed.

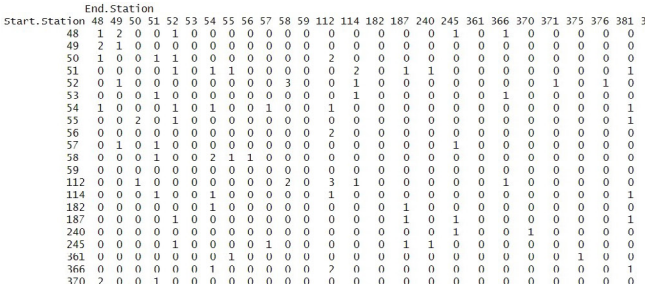
The basic concept in our design is the concept of Scenario.

A Scenario represents a geotemporal unit where we can assume a stable behaviour of the mobility. According to the literature and considering that mobility is a phenomenon affected by geographical and temporal stationarity, we will consider a bounded geographical zone (not much big, so that we can assume inside it the stable behaviour of mobility, like a neighbourhood or a street), and a temporal bound that maintain this stability. From the temporal point of view, two stational components must be considered: the day of the week (as for example Monday might show very different mobility than Saturday) and the time zone inside a day, as there is a second stational component where mobility in the mornings might be different from mobility in the evenings.

According to the previous considerations, we formalize the Scenario as a n-tuple:
(Z, d, h, t, c), where:

- Z is a zone (it might represent a district, or a neighbourhood, or any other territorial division represented in the dataset)
- d is a weekday:
  - Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday
- h is a time zone (after consulting with the stakeholders, we will consider the following time zones, associated with a classification also used internally in the Bicing company:)
  - Morning: from 6:00 h to 12:00 h
  - Afternoon: from 12:00 h to 18:00 h
  - Evening: from 18:00 h to 21:00 h
  - LateEvening: from 21:00h to 24:00h
  - Night: from 0:00 am to 6:00 am
- t is a temporal window defining a period between two bounding dates
- c is a cluster of stations, ideally close to each other. Along the modelling we will see the criteria for determining these sets of stations.

From these formalizations, given a Scenario a mobility matrix MS will be derived so that it contains the adjacency matrix of a graph where the nodes are those stations in C and the cells contain the number of trips going from origin to destination station in the day d time zone h and observation period t (Fig. 13). On this graph representation, learning and reasoning components of the proposed system will be built.



| | End.Station | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Start.Station | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 | 57 | 58 | 59 | 112 | 114 | 182 | 187 | 240 | 245 | 361 | 366 | 370 | 371 | 375 | 376 | 381 | 3 |
| 48 | 1 | 2 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | |
| 49 | 2 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 50 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 51 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | |
| 52 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 3 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | |
| 53 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | |
| 54 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | |
| 55 | 0 | 0 | 2 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | |
| 56 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 57 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 58 | 0 | 0 | 0 | 1 | 0 | 0 | 2 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 59 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 112 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 3 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | |
| 114 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | |
| 182 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 187 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | |
| 240 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | |
| 245 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| 361 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | |
| 366 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | |
| 370 | 2 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |

Figure 13. Partial view of MS= Raval, Wednesday, Evening, 20 Sep. 2021.

## 5.3. Visualization of the mobility graphs

The system, crossing the trips data with the geolocations of the stations involved, can select all the trips in a scenario and visualize the mobility matrix. It is a directed graph and represents in different colours the going and return trips between two nodes. Fig. 13 represents the MS corresponding to the scenario S= (Z=Dreta de l'Eixample, d= Saturday, h=Afternoon, t=[10 Sep 2022:00:00 – 10 Sep 2022:23:59], c= List of Bicing stations on the Dreta de l'Eixample neighbourhood (there are 34 stations represented in the figure). The module that visualizes these mobility matrices can work with any time interval to instantiate the parameter t.

The location of the stations in the image is related to their geographic position and therefore the proximity relationships between the stations correspond to the real ones (that is, two stations s1, s2 spaced twice as far as other two stations s3, s4, maintain a real distance between them which is twice the distance between s3 and s4).

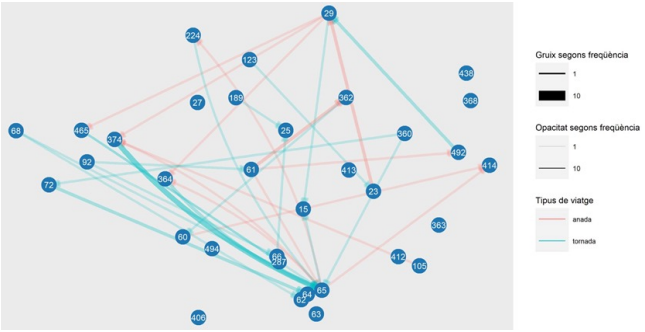



Figure 14. Visualization of a Scenario mobility matrix: (Z=Dreta de l'Eixample, d= Saturday h=Afternoon, t=[10 Sep 2022-10 Sep 2022], c= list of Bicing stations located in Dreta de l'Eixample (34 stations in figure).

## 5.4. Determining the Bicing stations of a given zone

The Bicing stations are labelled in the stations database with the neighbourhood where they belong. A simple approach of the problem would be based on working in an isolated way from every neighbourhood. However, in the reality the frontiers of the neighbourhoods are fuzzy. So that, we developed a method that parameterizes with a surrounding frontier of thickness δ (in Kms) every neighbourhood (namely δ-frontier). It is able to detect all Bicing stations inside these frontiers, so that they can be considered in MS as well. This permits to include in the normal behaviour of a certain Scenario the trips starting or finishing out of the neighbourhood but in very close stations to them. If δ =0 the strict list of Bicing stations of the neighbourhood is obtained. In the prove of concept we have been working with frontiers of thickness δ=0.1Kms, meaning that Bicing stations placed at less than 100 meters from a neighbourhood are also considered in the corresponding $M_S$.

An important remark is to take into account that the distance d(C1, C2) between a pair of Bicing stations have been computed upon their geolocalizations and using geodesic distance, so that real distances are obtained including the effects of the Earth curvatures.

This function has been used to pre-calculate matrix Δ, the pairwise geodesic distances between all pairs of Bicing stations in the city of Barcelona, which is based on the Stations Database and uses the geolocalizations of the Bicing stations.

From the same database, the list of all stations in a given neighbourhood Z can be easily obtained by a simple filtering. Using Δ, and δ=0.1Kms, additional Bicing stations can be identified in the δ-frontier. Thus, the component C of a given scenario will be composed by all Bicing stations strictly belonging to Z plus all those included in its δ-frontier.

## 5.5. Learning normality in a given scenario

A methodology to learn the normal behaviour of a given scenario is developed. So that, for the Bicing stations C of a certain neighbourhood (enlarged with its δ-frontier), weekday d, time zone h, and a given period t (fixed to 2021-2022 in this prove of concept and eventually divided between summer and winter periods), the normal behaviour is learned based on the construction of the average mobility graph of the scenario $\overline{M_S}$.

The average $\overline{M_S}$ is calculated by building the $M_S$ of all S in t with constant d and h and averaging every cell of these $M_S$ into $\overline{M_S}$.

A particularity to be considered to keep robustness is that we have been using the truncated averages to 0,95% for each Scenario.

The result of this process is a database of average mobility matrices for every scenario that can be learned off-line and periodically recalibrated. They represent the standard mobility of the different areas of the city at different weekdays and time zones. The computation of these $\overline{M_S}$ is very cheap from the computational point of view as the biggest C is the one of La Dreta de L'Eixample neighborhood and contains 60 stations, meaning that the biggest $M_S$ contains 3600 cells and the smallest one is La Marina del port with 5 stations and an $M_S$ of 25 cells. In the prove of concept we have been computing the average mobility graph of 2157 scenarios corresponding to 63 neighbourhoods per 7 weekdays and 5 time zones per day with a normal personal laptop in less than 20 minutes. There are 48 scenarios missing because they do not register any trip in the studied period. We insist to remark that this information can be pre-computed of line in the learning step of the project.

## 5.6. Detecting unexpected events in the behaviour of the city

For a given scenario S, let us define $\mathcal{M}_S = \{M_{S\tau} : \tau \in t\}$ the set of all mobility matrices corresponding to the Zone Z, weekday d, time zone h for each individual date in t.

Given the average mobility graph for Scenario S, let us name $\overline{M_S}$ and all individual mobility graphs in $M_S$, the distances $d_{S\,\tau}(M_{S\tau}, \overline{M_S})$ are computed. The distance between an instantaneous mobility matrix $M_{S\tau}$ and its corresponding average mobility matrix $M_S$ is computed as a standard distance between adjacency matrices of two graphs. The result is a vector of distances $D_S$ of length equal to card ($M_S$). Several tests were done with different d functions: Euclidean, MSE and $X^2$ distances. The tests indicated that the one providing better performances was Euclidean distance.

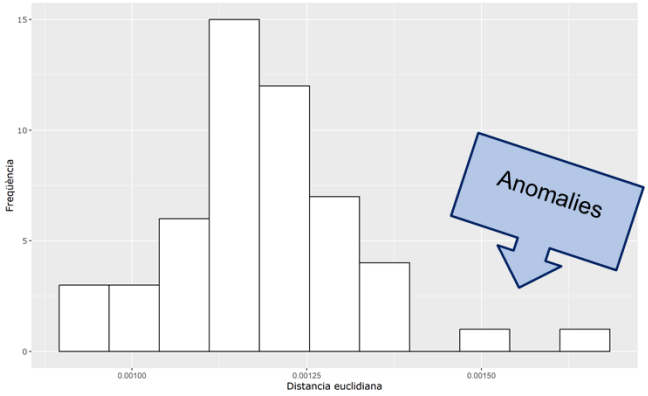



Figure 15. The reference distribution of the distances between instantaneous mobility matrix and normal behaviour in Poble Sec on Mondays between 21p.m. and 14p.m. in Autumn.

On the basis of vector $D_S$ a permutation Fishers test is computed so that the reference distribution of the distances between instantaneous mobility matrix and normal behaviour of the scenario is computed. The 0.95 percentile of the corresponding reference distribution is identified as the critical threshold to raise decision rules for tagging an instantaneous mobility matrix as normal or significantly abnormal (see Fig. 15)

As a result of doing this process for each of the neighbourhoods a normality thresholds data matrix H can be precalculated so that new daily observa-

tions of the Bicing trips can easily lead to normal/abnormal tagging with a very simple reasoning rule:

If $d_{S\tau}(M_{S\tau}, \overline{M_S}) > d_{S\alpha}$ then tag $M_{S\tau}$ as abnormal behaviour.

The thresholds data set has the following structure:

Z, d, h, t, C, $d_{S\alpha}$

## 5.7. Learning the data tags (from the unsupervised to supervised paradigm)

From a completely unsupervised original dataset, we have now all the elements to convert the problem in a supervised one. The process to assign a tag normal/abnormal to an instantaneous observation τ is:

1. to build $M_{S\tau}$
2. to retrieve $\overline{M_S}$ from the precomputed data structures
3. to compute $d_{S\tau}(M_{S\tau}, \overline{M_S})$
4. to find $d_{S\alpha}$ from the T table
5. to apply the decision rule of the Fisher's permutation test
   If $d_{S\tau}(M_{S\tau}, \overline{M_S}) > d_{S\alpha}$ then tag $M_{S\tau}$ as abnormal behaviour.
   , otherwise tag as normal.

At the training phase, this methodology is used to build a supervised training database where all instant observations receive their corresponding tag (like in Fig. 16).

Provided that the goal is to predict the abnormal behaviour one day in advance, the tag will be lagged one day in the dataset, so we will try to predict the situation of tomorrow from the data from today and all pre-calculated components with statistics.

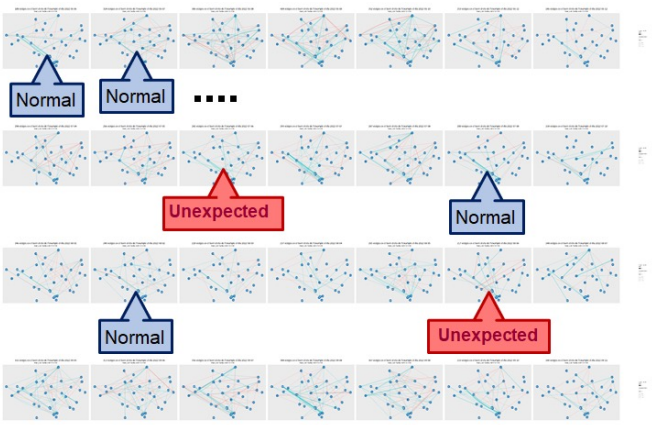



Figure 16. Intelligent tagging of the instantaneous observations of mobility in a certain scenario.

## 5.8. Anticipated prediction of unexpected events

In this task the database described in the previous section (Fig. 17) is used to train a case-based reasoning model where the target variable is the lagged tag (Normal/Unexpected). So, we try to predict the tag from tomorrow with the mobility matrix from today. The MS are serialized and the lagged response variable added at the end of the data matrix.

A simple case of case-based reasoning technique is used, reduced to the K-nearest neighbour. Special optimization of the k is done for each specific scenario. Each model is trained with the 80% of available data and tested with the remaining 20%. The goodness of fit metrics used are the accuracy, the precision, the sensitivity and specificity of identifying the abnormal event. The predictive models will be trained locally to every scenario as we chose the scenarios in such a way that the behaviours are homogeneous inside each S.

Balancing techniques have been used to overrepresent the "unexpected" events tagged in the data base, which is unbalanced by construction and requires a better equilibrium between normal/unexpected than the original one, biased towards normal tag.

The quality of the models is relatively promising if this is a prove of concept trained using only 2-years history and enlarging the historical data the results will still improve.

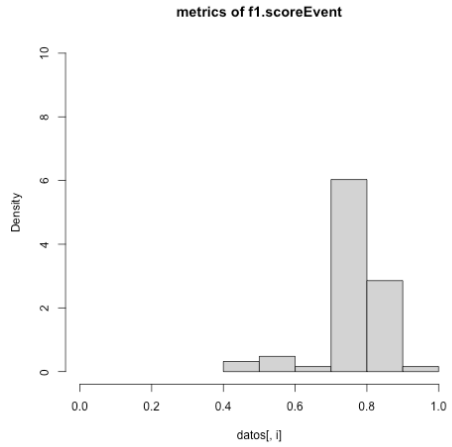
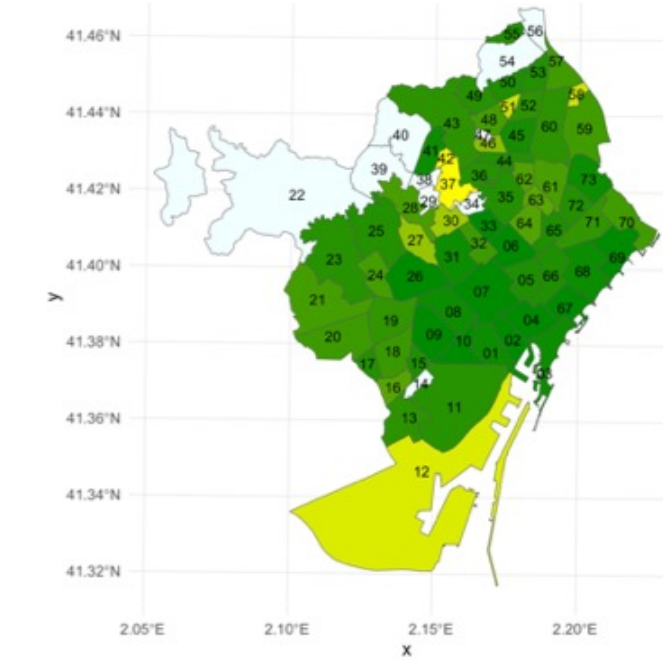



Figure 17. Specific score F1 for Event modality.





| CODE | NEIGHBORHOOD | CODE | NEIGHBORHOOD |
|---|---|---|---|
| 1 | el Raval | 38 | la Teixonera |
| 2 | el Barri Gòtic | 39 | Sant Genís dels Agudells |
| 3 | la Barceloneta | 40 | Montbau |
| 4 | Sant Pere, Santa Caterina i la Ribera | 41 | la Vall d'Hebron |
| 5 | el Fort Pienc | 42 | la Clota |
| 6 | la Sagrada Família | 43 | Horta |
| 7 | la Dreta de l'Eixample | 44 | Vilapicina i la Torre Llobeta |
| 8 | l'Antiga Esquerra de l'Eixample | 45 | Porta |
| 9 | la Nova Esquerra de l'Eixample | 46 | el Turó de la Peira |
| 10 | Sant Antoni | 47 | Can Peguera |
| 11 | el Poble-sec | 48 | la Guineueta |
| 12 | la Marina del Prat Vermell | 49 | Canyelles |
| 13 | la Marina de Port | 50 | les Roquetes |
| 14 | la Font de la Guatlla | 51 | Verdun |
| 15 | Hostafrancs | 52 | la Prosperitat |
| 16 | la Bordeta | 53 | la Trinitat Nova |
| 17 | Sants - Badal | 54 | Torre Baró |
| 18 | Sants | 55 | Ciutat Meridiana |
| 19 | les Corts | 56 | Vallbona |
| 20 | la Maternitat i Sant Ramon | 57 | la Trinitat Vella |
| 21 | Pedralbes | 58 | Baró de Viver |
| 22 | Vallvidrera, el Tibidabo i les Planes | 59 | el Bon Pastor |
| 23 | Sarrià | 60 | Sant Andreu |
| 24 | les Tres Torres | 61 | la Sagrera |
| 25 | Sant Gervasi - la Bonanova | 62 | el Congrés i els Indians |
| 26 | Sant Gervasi - Galvany | 63 | Navas |
| 27 | el Putxet i el Farró | 64 | el Camp de l'Arpa del Clot |
| 28 | Vallcarca i els Penitents | 65 | el Clot |
| 29 | el Coll | 66 | el Parc i la Llacuna del Poblenou |
| 30 | la Salut | 67 | la Vila Olímpica del Poblenou |
| 31 | la Vila de Gràcia | 68 | el Poblenou |
| 32 | el Camp d'en Grassot i Gràcia Nova | 69 | Diagonal Mar i el Front Marítim del Poblenou |
| 33 | el Baix Guinardó | 70 | el Besòs i el Maresme |
| 34 | Can Baró | 71 | Provençals del Poblenou |
| 35 | el Guinardó | 72 | Sant Martí de Provençals |
| 36 | la Font d'en Fargues | 73 | la Verneda i la Pau |
| 37 | el Carmel | | |

Figure 18. Neighbourhoods with better or worse F1 quality scoring of the predictive model can be seen that most of the Scenarios provide very good results, with F1 score over 75%, whereas some few scenarios show worse results.

Only few neighbourhoods like El Carmel and la Clota are impacted by a step hill that makes uncomfortable to bike, followed by la Marina del Prat Vermell that includes the entire Zona Franca, the harbour vicinity area of Barcelona where ordinary citizens are not allowed to enter. We presume that in these cases very few trips have been recorded and the entire neighbourhood should be not considered into the analysis.

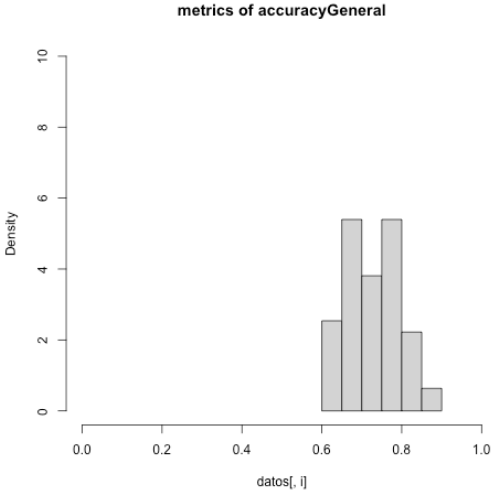



Figure 19. Global accuracy conditioned to the scenarios.

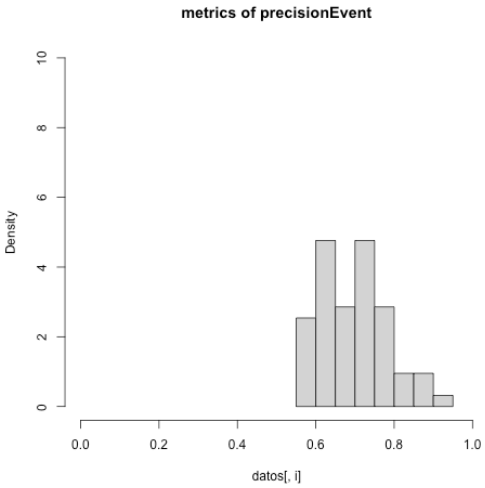



Figure 20. Global precision of the models.

## 5.9. Automatic interpretation of the unexpected events

In this task a specific module to provide explanations and interpretations of the recommendations given to the stakeholder provided the predictions require two main elements:

1. To communicate technical information for the recommendation.
- The average graph of the generic scenario associated to the instant of the prediction.
- The instantaneous graph from today
- The distance between both
- The critical threshold associated to the target scenario and the quantile of the observed distance.
- A natural language processing paragraph explaining that the distance to the average graph is too high, and the day has been tagged as anomaly.

2. The events from the city of Barcelona programmed by the predicted day (the day after the event) in the target zone and time schedule. Direct on-line connection to the agenda of the city allow to tag the remaining testing and new data.

The recommendations of the system may be performed upon demand or automatically every day at a certain fixed time. They consist of a list of neighbourhoods where the expected demand for the next day is unexpected and the recommendation is to install a virtual station. Clicking to every "alert/recommendation" one can get all the info given above, understand if the differences towards standard behaviour of the observed data that day are different from the standard and reading about the events scheduled in the agenda of the city that day. This information will provide relevant details to the Bicing manager to understand the reasons of the prediction and support decision-making by activating the protocol for installing the virtual stations or not.

For example, the tests done in the Scenario corresponding to Saturday night between September and Novembre 2022 at neighbourhood Poble Sec, allows to identify two unexpected events happening on September 23rd and 26th respectively. The city agenda identifies for sept 23rd a concert of the "La Mercè" festivity and the fireworks of the same festivity on the 26th. These two events are extremely popular, with crowded streets that completely alter the behaviour of the city. Indeed, extra virtual stations would have been needed near Plaça Espanya, located at the centre of this studied Poble Sec neighbourhood.


Figure 21. From the time series to the Barcelona city agenda.

## 5.10. IDSS for the predictive installation of virtual Bicing stations

The system is named i-ViSta4bike (intelligent recommender for Virtual Station for Bikes) and with the pieces resulting from the previous machine learning applications, some learned models can be properly combined among them to provide a piece of software to be placed in production for managing the detection of places where a virtual station is required with one day anticipation (Fig. 22). We are assuming in this prove of concept that the system is not learning online. We performed an offline learning process and the production i-Vista4bike system will incorporate the learned components. This helps to improve the sustainability of the code and the trends of the Bicing system are not that much volatile to require on-line adaptive behaviour. We will provide bellow some more recommendations about how to update the system along time.

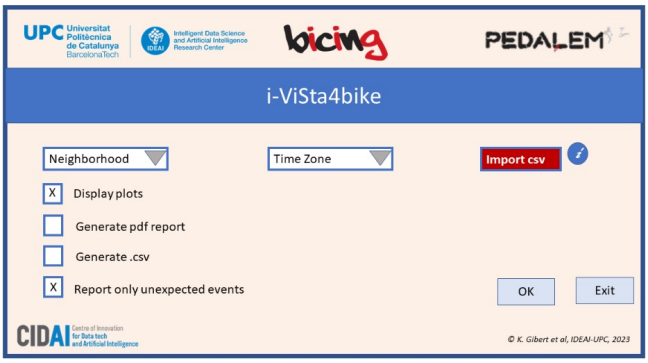



Figure 22. Home of i-ViSta4bike.

Input:

- Zone: the user can select the neighbourhood to be inspected
- Time zone: The user selects the time zone to be inspected (All option is included, so that the analysis for all time zones of the selected neighbourhood is analysed)
- Trips from today (following the structure of the dataset trips delivered by the company. This file has to be sent every day. In a future step the data will be received through the data space created in the first use case.

File System required for the working of the recommender:

- Folder with all the automatically learned average mobility matrices from the 2147 Scenarios.

Output:

- Visualization of the trips from today, in the neighbourhood selected and time zone selected by the user (observed mobility)
- Visualization of the reference average mobility graph (same weekday, same neighbourhood, same time zone) (expected mobility)
- Information about the distance between the current Mobility matrix and the reference average one (the two graphs displayed on the screen)
- Prediction for tomorrow, same time zone and area (unexpected event/normal)
- Verbal interpretation of possible causes of unexpected behaviour for tomorrow including the information obtained from the city agenda (build by using regular expressions).

This information is visualized in a pop-up window for every scenario contained in the user selection. The system offers the possibility to activate some additional options like:

- Build a pdf report with the results.
- Export a csv file with the list of neighbourhoods and time zones expecting abnormal behaviour for tomorrow.
- Filter the results to the scenarios expecting abnormalities so that the normality is neither displayed nor included into the pdf reports.
- For the cases where massive queries are required (for example compute predictions for all neighbourhoods and all time zones), there is the possibility to unselect the generation of graphical pop-up results and concentrate the entire output into the pdf or CVS exportable files.

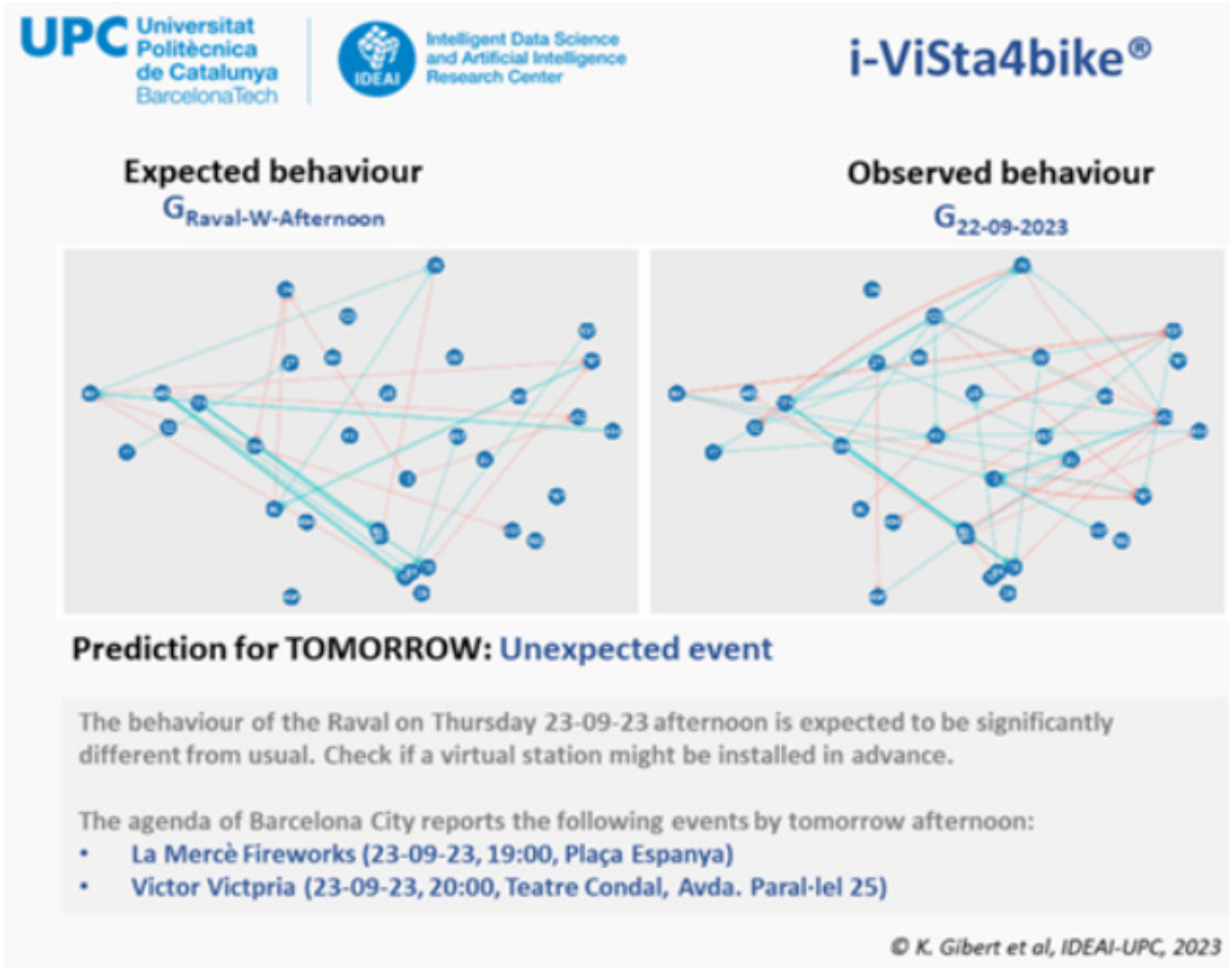



Figure 23. The output of i-ViSta4bike with the recommendation and explanation.

From time to time, it is convenient to retrain the predictive model periodically to keep adaptability of the system to changes in the systemic behaviours. To this, the update of the average mobility graphs only requires updating the average number of trips of every cell in the mobility matrices with the new data arrived from the last update. After that all predictive models from each scenario can be retrained. For a more resilient update, significant thresholds for abnormal behaviour of a scenario can as well be updated from time to time if big changes in the behaviour of the city are observed.

The architecture of the IDSS i-ViSta4bike follows this structure:

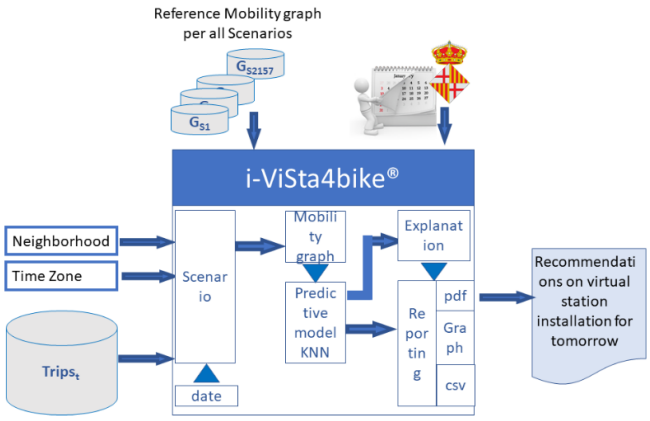



Figure 24. Architecture of the IDSS i-ViSta4bike.

## 5.11. Modelling the battery levels

We model the change in battery levels of the Bicing boost bikes with a method based on fundamental laws of motion. The instantaneous power consumed when riding a bike is given by the power needed to overcome the main three forces opposing the motion (Burani et al. 2022, Steyn & Warnich, 2014) (i.e., gravity resistance, rolling resistance, and aerodynamic drag).

Thus, to compute the total power consumed in a trajectory it is needed to consider the instantaneous speeds, which is an information do not have. Instead, we calculate it by dividing the total travelled distance by the trip time. Since we do not have approximate values for the parameters participating in each force, and they could vary depending on the customer and the bike status, we aggregate them into fittable parameters. Moreover, the distance travelled is approximated using the shortest path between stations since we do not have details on the routes taken.

The final fittable expression to calculate the percentage of battery consumed in a trip includes one constant for each force and a global constant to weigh the amount of battery consumed by the trip. Similarly, we also fit the rate at which the battery recovers which is the resting time multiplied by the charging constant. We fit the battery function on the 30-minute snapshots dataset provided over three weeks (from the 29th of March to the 17th of April) using 75% for training and 25% for validation and achieving a Pearson correlation coefficient of around 0.67.

## 5.12. Micro-mobility model

### Single trip prediction

As a first application, we have coupled the distribution of possible destinations given an origin and an hour of the day to the battery function to obtain insights on the future movement of a bike and its battery consumption (Fig. 25).

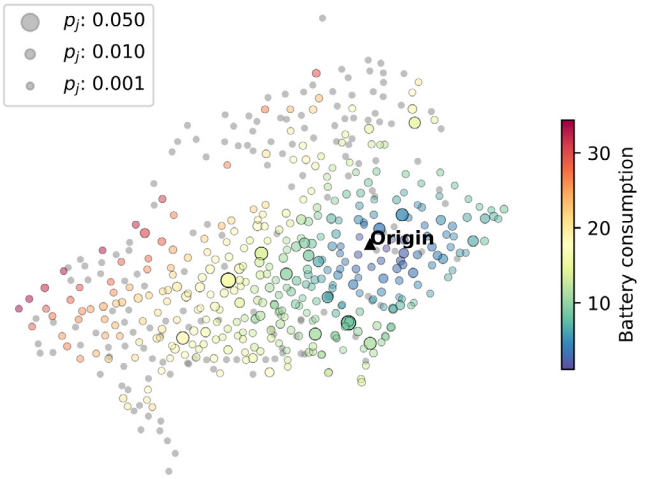



Figure 25. Probability distribution of bikes with origin in station 42 (c/ Ciutat de Granada, 168 | Av. Diagonal) at 8:00 AM on a weekday and levels of battery consumption (in %) in destination. The size of the circle represents the probability to use this station and the colour the battery consumption. The battery consumption is shown as a percentage (%).

The probability of the destinations changes significantly between morning (e.g., 8:00) and afternoon (e.g., 18:00) despite the origin station being the same. While in the afternoon the trips tend to be shorter to close-by stations, morning trips are longer and concentrate towards a few workplace hotspots. We also observe significant changes in the battery consumption depending on the inclination that the bike has to go over with trips going downhill implying a lower battery consumption within the same distance. This approach allows us to compute an impact index for each station that depends on the probability of each destination given an origin station, the battery consumption of each possible trip, and the average hourly trips.

### Microscopic Simulation Model

In this section, we describe the microscopic simulation model built to reproduce the daily flows

through the bike-sharing system of Barcelona and the evolution of battery levels of boost bikes (Fernandez, 2010, Azevedo et al. 2017, Fernandez et al. 2020). The model simulates the movement of each bicycle and tracks the change in battery levels of boost bikes using the model mentioned before. Each hour is modelled sequentially according to the following steps:

- We generate the hourly trips departing from each station and distribute them homogeneously in five minutes windows. The number of hourly trips generated at each station depends on the type of day and hour. For simplicity, we have only considered four types of days: weekdays (Monday to Thursday), Fridays, Saturdays, and Sundays.
- We assign the hourly relocation from each station and relocate them following the distribution of relocations. For simplicity, we consider that all relocations are instantaneous.
- We iterate over the five minutes windows in this way:
  - We execute the corresponding trips for a time window, selecting one bike at random among those available. As in the real system, trips with boost bikes below 20% are not allowed. The destination and travel time are sampled from the empirical distributions. The distribution of destinations and travel times depend on the type of bike since boost bikes tend to do longer and faster trips. The trips only happen if the origin is not empty, and the destination is not full.
  - We check the travel time of each bike on route and execute the arrival if it is equal to or lower than zero. If the destination station is full, we redirect the bike to the closest station.
  - For each boost bike at a station, we charge the percentage of battery corresponding to the fitted parameters.
  - We subtract five minutes from each of the bikes on route.
- The current hour ends, and the process goes back to the beginning only if the global clock is lower than the desired forecasting horizon.

All the distributions are computed from the trip data that occurred during 2022. The model allows us to provide a prediction for the global behaviour of the Bicing system that includes the number of trips, their origin and destination, the capacity of stations, and the evolution of battery levels. To validate the model, we use each of the 30 minutes snapshots spanning from the 29th of March to the 17th of April dataset as input conditions and compare the system after Δt minutes. Although the correlation is far larger for the location of bikes, it remains significant within a range of 120 minutes in both cases.

We further check if the dynamics of the Bicing system in the simulation have spatial patterns compatible with the observations. In Fig. 26, we display the station outflow for a simulation of 6 hours starting at 08:00 and 16:00. During the morning, the station at *Provença 215* seems to be the main hub, while the station at *Glòries* is the most relevant in the afternoon, in line with the observations in the data analysis section. Other stations (e.g., *Comerç* or *Enric Granados*) also are important hubs in our simulations.

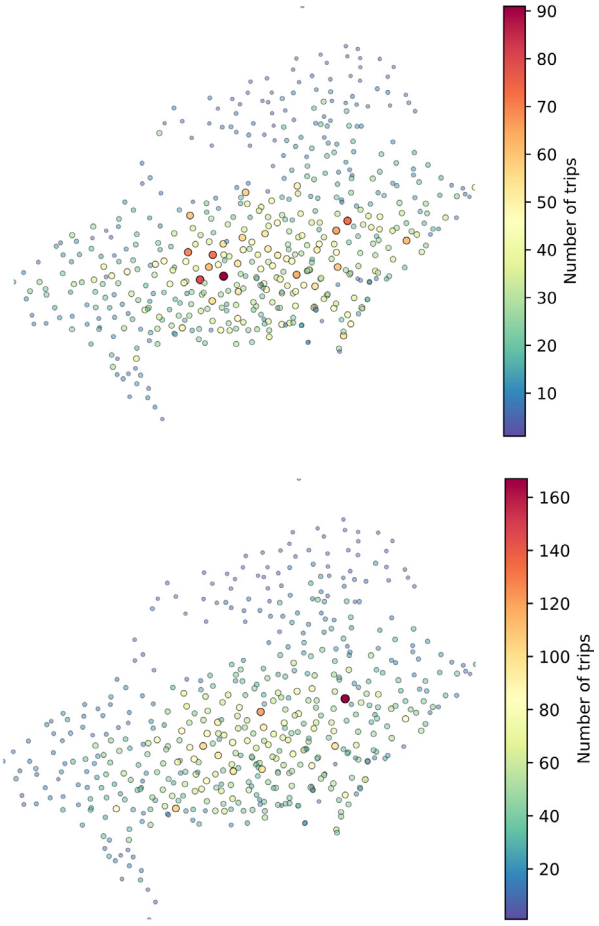



Figure 26. Number of outbound trips in each station after six hours of simulations. On the left starting at 8:00 a.m. and on the right starting at 16:00 p.m.

The coupling between the mobility model and the fitted battery function provides predictions for the bikes available and the battery levels of the stations by

tracking the location of bikes and their change in battery levels. Fig. 26 shows the number of bikes available at each station together with the average battery levels after two hours for two simulations starting in the morning and in the afternoon. In the morning, stations in the city centre have fewer bikes available and lower battery levels, where lower availability and battery levels are an expected outcome. Since bikes charge at night, the lower average battery levels achieved in the afternoon are expected due to their progressive battery level decrease along the day.

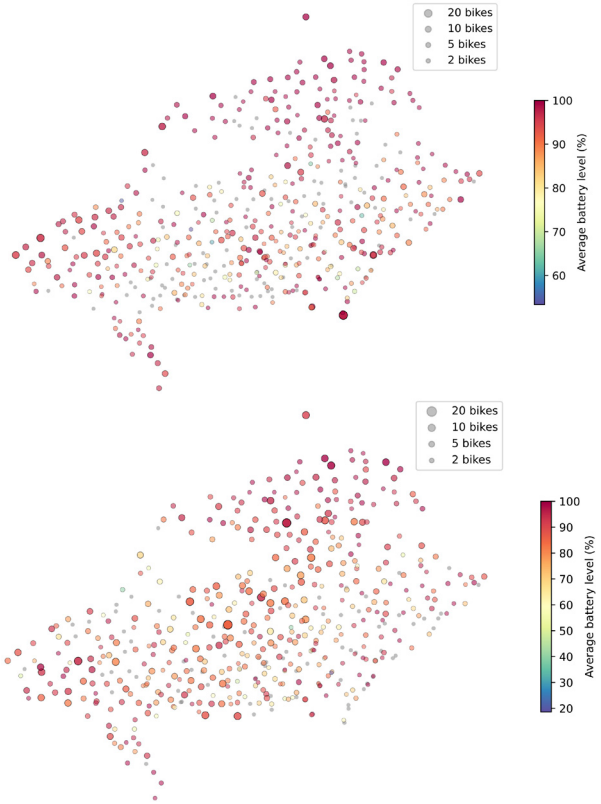



Figure 27. Prediction of bikes available and battery levels. Prediction of the number of bikes available and battery levels after two hours according to the simulations on a weekday starting at 08:00 (left) and 18:00 (right). The size of the dots corresponds to the number of bikes available and the colour to the average battery level of bikes. The battery level is shown as a percentage (%).

Beyond aggregated measures, our model simulates the movement of every single bike providing insights on the future movement of bikes and their battery change. To illustrate the level of detail at which we could potentially predict the movement of a bike, we show in Fig. 28-29 the trajectory of a bike during a weekday, along with the battery change.

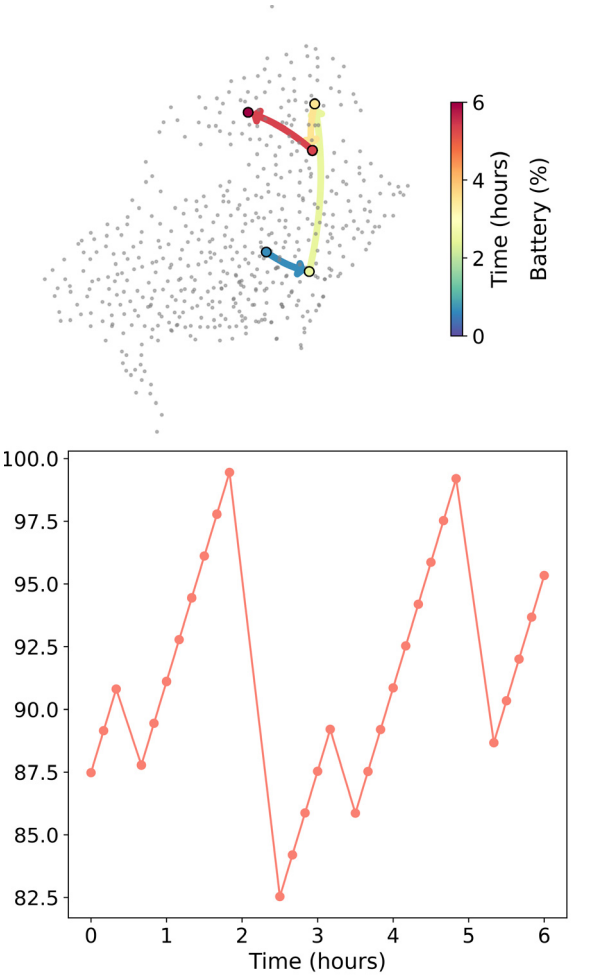
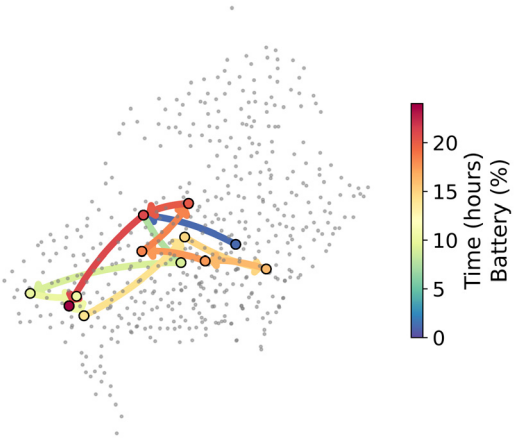



Figure 28. On the left, we represent the trajectory of a particular bike during a period of six hours. The arrows are coloured according to the time at which the trip occurred. On the right, we can observe the evolution of the battery level. The battery is sampled every 10 minutes only when the bike is docked.



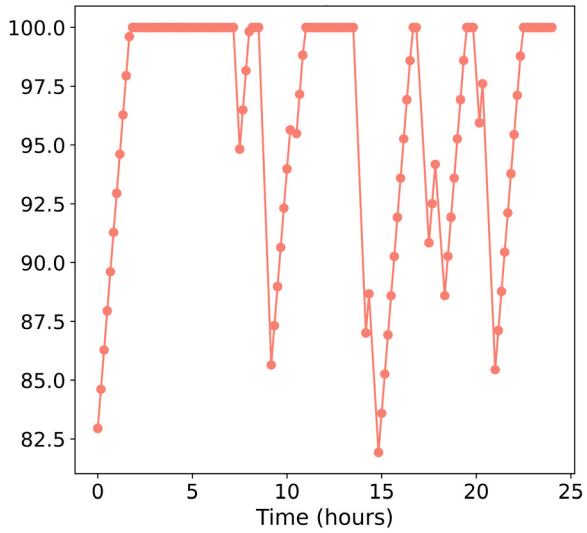



Figure 29. On the left, we represent the trajectory of a particular bike during a period of twenty-four hours. The arrows are coloured according to the time at which the trip occurred. On the right, we can observe the evolution of the battery level. The battery is sampled every 10 minutes only when the bike is docked.

## 5.13. Markov-mobility model

We have built a second model to predict the location and battery of bikes based on a Markov-chain approach (Lu et al. 2013, Huang et al. 2015). Each bicycle in the park is modelled individually according to the probability of transition at its current location.

Given a starting location for a bike, the algorithm follows the sequence:

- An inter-event time is sampled from the resting time distribution at a given station based on the time elapsed between the arrival of a bike at the station and its next departure.
- If the time elapsed after the inter-event time is higher than the forecasting horizon the process stops, and the final location of the bike will correspond to the current station.
- Once the inter-event time has elapsed, a destination is sampled from the probability of having a destination given a weekday, hour and origin station. The destination probability is calculated from the 2022 trip data including both regular and relocation trips.
- The bike moves to the destination adding up the average travel time between those stations.
- If the time elapsed after the trip is larger than the forecasting horizon, the final predicted location will be station.

Since it is a stochastic process, we perform 200 realizations per bike for an overview of the possible scenarios. The main result we obtain is the probability of ending at station after an increment of time based on the weekday, hour and origin station, which is the times that a bike ended in station per numeral of iterations. Since we track the change of the battery levels, we will also have the expected battery level of a particular bike at a destination station after a given increment of time. The average over realizations can produce non-integer bikes at each possible destination since they are averages of multiple realizations. We perform the validation using the evolution of bikes per station and the average battery level using as inputs the 30 minutes snapshot dataset going from the 29th of March to the 17th of April as with the mobility model. The correlations are higher than for the mobility model, again smaller for the battery levels compared to the bike number.

We display in Fig. 30 the probability of finding a bike departing from two stations on a weekday during the morning after two hours together with the ex-

pected battery level. The Markov chain approach models the movement of individual bikes and battery changes, and thus, provides a prediction for the bikes available and battery levels by summing the individual bike probabilities. In contrast to the mobility model, which provides one possible scenario, the probabilistic approach provides a broader perspective with an average over multiple possible scenarios. The trend of central stations with lower battery levels is clearer with the Markov chain approach, with an epicentre near Plaça Catalunya. We also observe lower battery levels in the afternoon due to battery consumption during the day. The previous results can be aggregated at the level of the origin station to provide the average battery levels of bikes after departing from a station. The stations with higher battery consumption are close to the main city centre, although they vary significantly with the hour of the day.

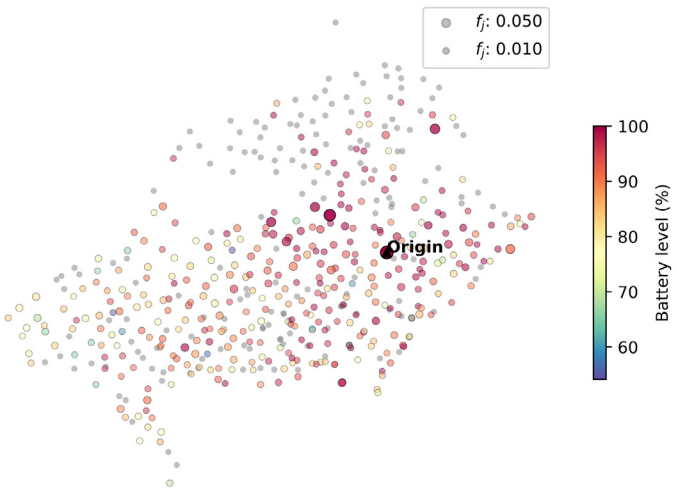



Figure 30. Destination probability and battery level depending on the origin predicted by the Markov-chain approach. Prediction for the probability of finding a bike at any given station and its battery level after two hours for a simulation starting at 08:00 am. at station 42 (c/ Ciutat de Granada, 168 | Av. Diagonal). The is shown in percentage (%).

# 5.14. Predictive maintenance model

## Survival analysis

Survival analysis methodologies were initially developed in the biomedical sciences to analyse life tables (Cox, 1972), however the concept of survival analysis extends its applicability to various scenarios, encompassing not just mortality, but also the recurrence of a disease, failures in machine components, or any other event that marks a significant change in the state of a subject. The three main objectives of this field are:
i) to study the behaviour of a particular event (to obtain this knowledge two kind of function estimations are usually performed *Estimate Survival Functions and Estimate Hazard Functions*)
ii) to identify risk factors
iii) predicting future events. Given the nature of a predictive maintenance system, in this use case, the main objective is to predict when a certain bike part needs repair. To our knowledge, there are no applications of survival analysis to bike-sharing systems such as Bicing.

## Censoring

One key challenge in predicting the occurrence of an event is the presence of censored data, which refers to incomplete information about survival times (Leung et al. 1997, Gijbels, 2010). Censored data appears when the event of interest has not occurred for a subject by the end of the study or observation period. This could be due to various reasons like the study ending before the event occurred, loss to follow-up, or the event not yet happening. In such scenarios, survival analysis models demonstrate greater effectiveness compared to conventional statistical and machine learning techniques.

Bicing's maintenance data contains left- and right-censored data. Due to the nature of a data set, information starts and finishes in a particular arbitrary date. Therefore, the first and last maintenance orders (MOs) of each bike are incomplete. In the first MO, it is not possible to know when this bike part started working, and in the last MO unit, when this bike part finally will break. With the utilization of survival analysis, uncensored and right-censored data can now be effectively utilized to estimate the survival curves and generate the predictions.

## Inputs and Outputs

A survival analysis dataset is organized so that each row represents a distinct instance related to the object of investigation, which, in this case, corresponds to specific MO types we aim to model. Furthermore, within each row, it must be indicated the occurrence of the event of interest: denoted as 0 if absent or 1 if present. Importantly, the data set also encompasses for each row the covariate (e.g., weather, bike usage, bike model, other repairs) values that can potentially aid in estimating the survival function. The final objectives of the MO types for failure predictions were the brake pads, the wheel spokes and the chains.

## Models

According to (Wang et al. 2019), survival methods can be classified into statistical and machine learning methods. Statistical methods focus on characterizing the distribution of event times and the statistical properties of parameter estimation, such as estimating survival curves. Machine learning methods combine traditional survival analysis techniques with machine learning algorithms. To predict the bike parts' days of survival, the following statistical and machine learning models have been employed: Cox Proportional Hazard model (CPH) (Cox, 1972), Multi-Task Logistic Regression model (MTLR) (Yu et al., 2011), Conditional Survival Forest model (CSF) (Wright et al., 2017) and CPH Deep Neural network model (DeepSurv) (Katzman et al., 2018).

## Predictions

Predictions were generated using the uncensored MO from both the training and test sets to assess the performance of the models. The decision to exclude the right-censored MO was made to prevent comparisons of predictions with the time duration of MO that do not conclude with a repair.

Classical CPH models have exhibited the poorest accuracy. This can be attributed to the fact that the CPH model generates predictions through linear equations. As the wear and
tear behaviour of the components presents non-linearities at both the beginning and end of their lifespans, CPH models struggle to accurately capture these patterns.

To address the CPH model's incapability of handling non-linearities and potential violations of the proportional hazard assumption, the MTLR and CSF models were employed. Both models exhibited superior precision compared to the CPH model without data filtering. In contrast, the MTLR models performed nearly as well as or better than the CPH model with data filtering, despite not excluding any data.

Finally, DeepSurv was employed to incorporate deep learning methodologies into the classical CPH model, with the aim of enhancing the modelling of the non-linearity present in the data. These models have achieved the best results, as they have produced superior metrics without the need for data exclusion, outperforming even the MTLR predictions.

| | | All MO | Uncensored MO | Right-censored MO |
|---|---|---|---|---|
| Brake pads | Actual | 89.44 (97.81) | 86.54 (95.33) | 106.6 (103.73) |
| | Predicted | 86.54 (87.78) | 81.61 (85.27) | 113.73 (91.69) |
| Wheel spokes | Actual | 81.26 (98.42) | 68.33 (83.26) | 158.86 (140.18) |
| | Predicted | 83.03 (98.62) | 66.94 (79.91) | 178.62 (140.96) |
| Chains | Actual | 273.54 (160.1) | 279.8 (151.64) | 255.23 (169.19) |
| | Predicted | 295.9 (152.33) | 283.73 (144.53) | 297.27 (162.1) |

Table 1. Descriptive table of means and standard deviations (in parenthesis) for actual and predicted lifespan of bike parts (in days). Both statistics are presented for all available MO units and for the uncensored and right-censored MO units' subset.

## 5.15. Dashboard

Understanding mobility trends and indicators is crucial for making informed decisions. To facilitate this, we have developed a Dashboard using the Dash framework for the frontend and a Flask API for data retrieval and manipulation. This dashboard provides valuable insights into the mobility metrics of three distinct models, enabling users to analyse and visualize data effortlessly.

The Micro-mobility Dashboard offers an intuitive and user-friendly interface that facilitates users in exploring the mobility metrics of three different models: Micro-mobility model, Markov model, and Predictive Maintenance Model. Let's delve into the key features and components of this dashboard.

First of all, users can choose among three models. Selecting a specific model will dynamically update all the dashboard's visualizations and metrics to display data relevant to the chosen model. The dashboard displays essential mobility indicators for the selected model.

For the micro-mobility model, the user selects the day of the week, an initial hour, and the forecasting horizon from 30 to 360 minutes. In the dashboard, we can observe the distribution of trips in the next hours, the capabilities of each station, the battery levels in each station, and the battery levels per bike. Once we select a bike, we can see the trips carried out during the day and the evolution of the battery levels.

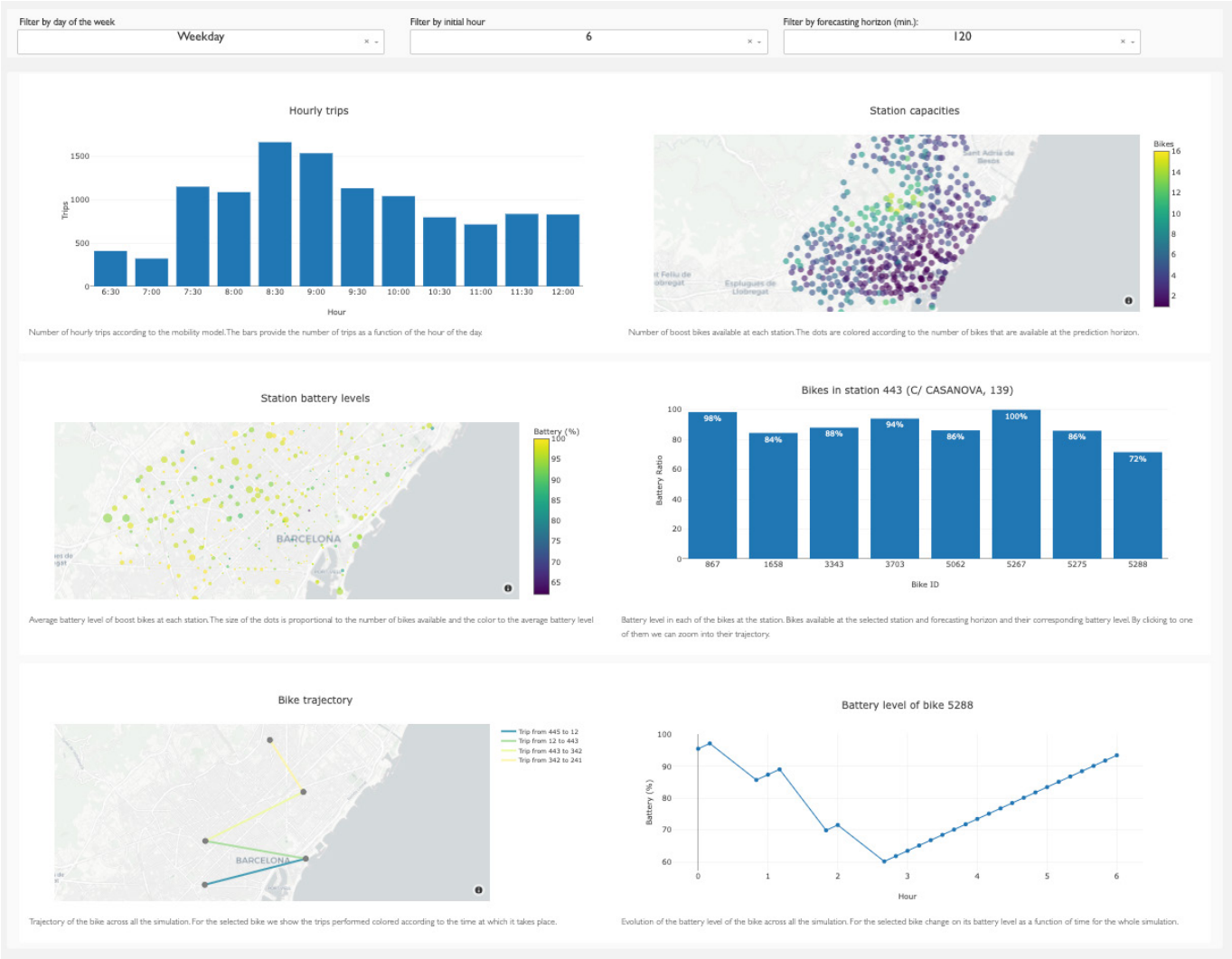



Figure 31. Dashboard with the results of the micro-mobility model, showing six plots representing forecast of mobility and batteries at bike level.

Regarding the Markov model, the user selects the same variables as in the micro-mobility model. In this case, the dashboard displays three plots: two maps showing the probability of finding a bike after the time horizon for a given origin station, another displaying the average battery level at each station, and a third graph with a bar plot showing the distribution of batteries at each station.

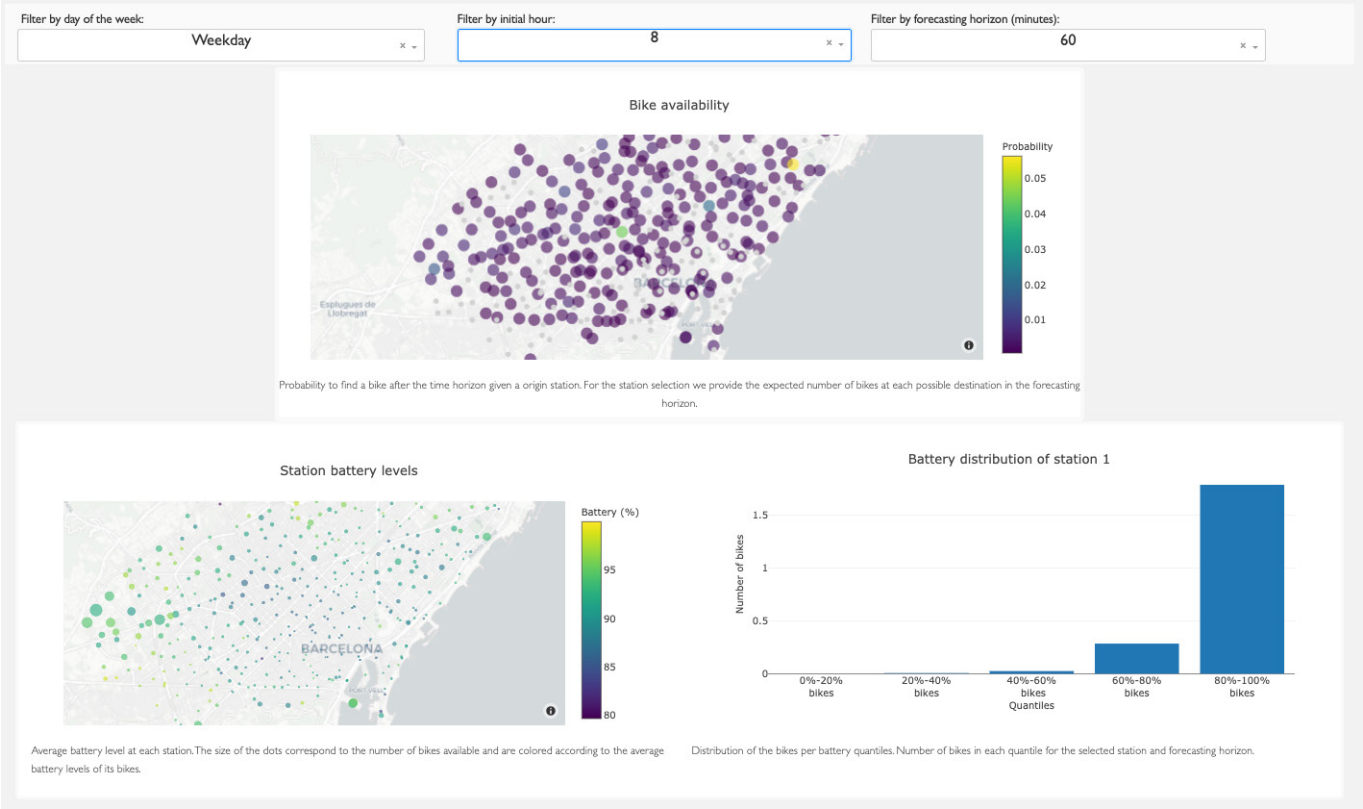



Figure 32. Dashboard with three plots representing the metrics generated by Markov model.

For the predictive maintenance models, users select a bike part (e.g., brake pads, wheel spokes, chain, etc.). The dashboard then provides information about the number of repairs for that specific part, the distribution of repairs over time, and the model's performance with accuracy metrics (e.g., RMSE, MAPE, and R2). Once a maintenance order is selected, users can view all maintenance orders for a particular bike and track the evolution of the most important model metrics: distance and inclination. Additionally, the dashboard displays other information related to repairs.

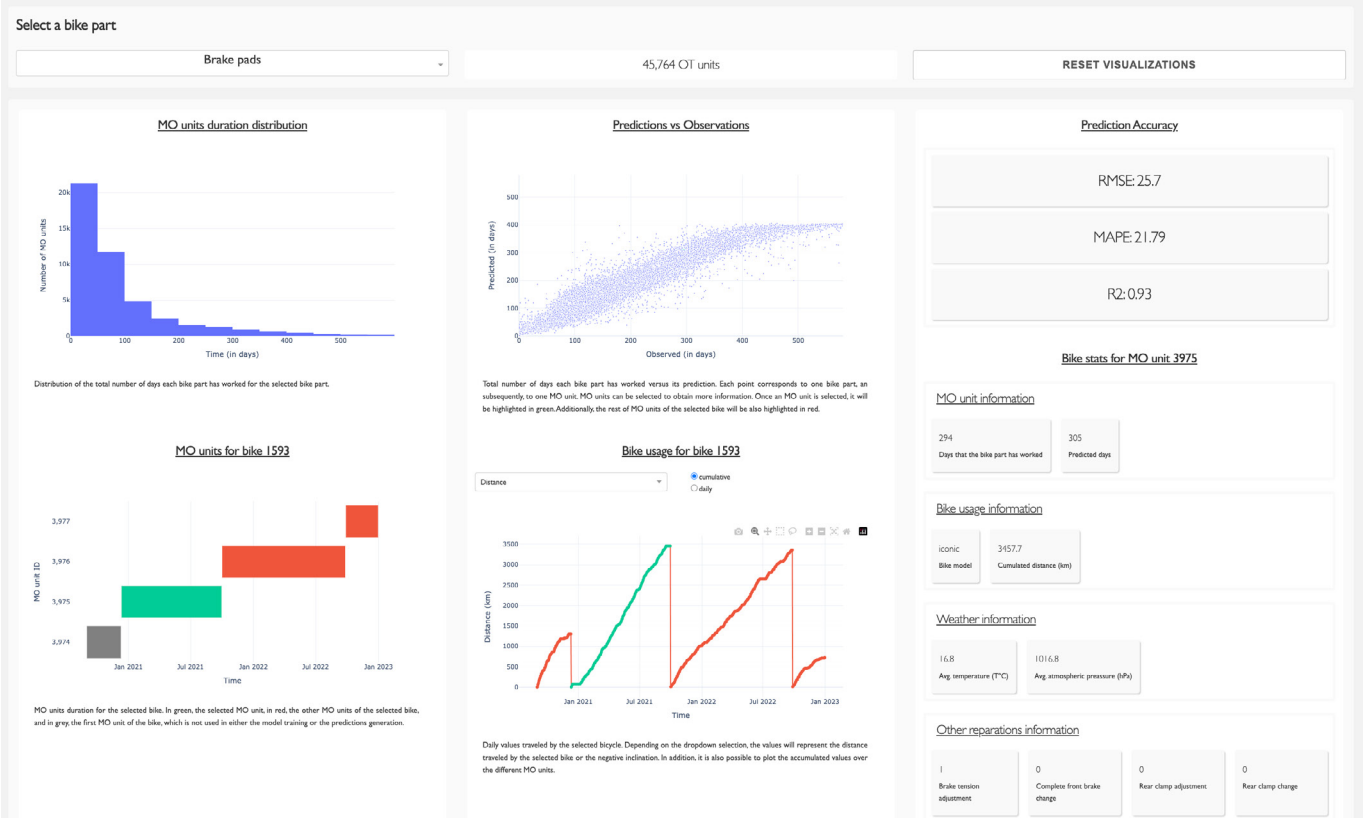



Figure 33. Dashboard with the metrics generated by the predictive maintenance model, DeepSurv, applied to one part of the bike.

The backend of this dashboard is powered by a Flask API, which serves as the bridge between the frontend and the databases. The Flask API is responsible for retrieving data for the selected model and model variables (e.g., prediction interval, weekday, bike part). This data is then sent to the Dash frontend for visualization and display. This dashboard combines the flexibility of Dash's frontend with the robustness of Flask's API, ensuring an efficient and user-friendly experience for mobility data analysis.

The micro-mobility dashboard offers a powerful tool for analysing mobility trends and making data-driven decisions. By providing insights into the mobility indicators of three different models, users can gain valuable insights into their operations and optimize strategies accordingly.

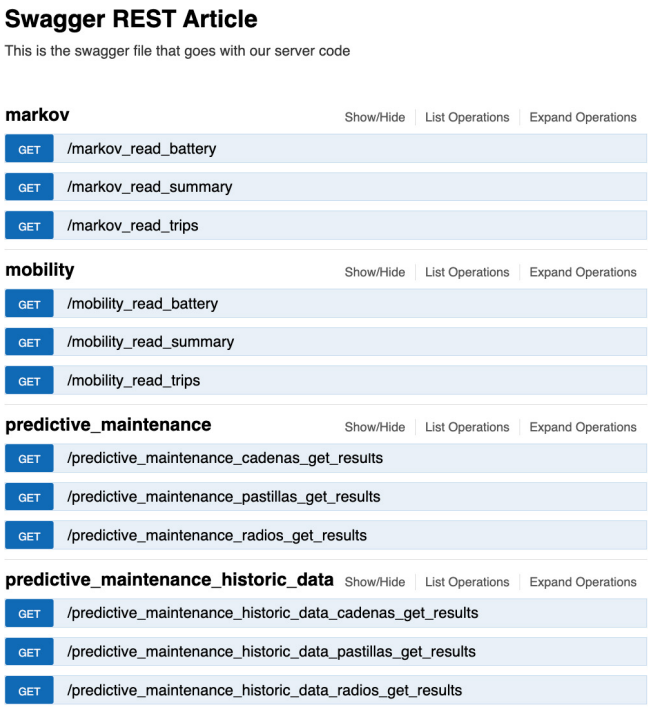



Figure 34. Swagger REST API calls to the database, displaying the outcomes of the model.

# Conclusions

In the pursuit of sustainable urban mobility, our project has taken a multifaceted approach, combining data-driven solutions with predictive models and maintenance strategies, and user-friendly interfaces. By focusing on Barcelona's bike sharing system, namely Bicing, we have successfully addressed several key aspects of micro-mobility to potentially impact in the benefit of commuters and, therefore, to reduce environmental impact of mobility in the city.

This project developed mobility data space, collecting and organizing vast amounts of data related to bike sharing system. This data serves as the backbone for our subsequent efforts in modelling mobility, predict bike maintenance, and visualization metrics. With respect to our mobility models, we have focused on the management of electric bikes within the system. Through data analysis and machine learning, we can predict battery levels and mobility patterns. This enables a more efficient bike distribution, reducing instances of depleted batteries and enhancing the overall user experience.

Regarding the mobility models, we followed two different approaches. One of them is entirely data-driven, showing good performance, as it reproduces most of the usual mobility patterns. The second approach focuses on Markov chains, allowing the creation of scenarios by fixing various initial conditions within the system. This model requires more computation power, due to the number of simulations needed to generate results which makes it slower but more suitable for scenario-building situations.

Predicting the battery levels, however, has been a challenge due to the lack of data on the actual battery levels of bikes. However, we have built a physics model to achieve a prediction of the bike battery consumption. The model has acceptable results achieving a correlation coefficient of around 0.67. In general, we can see patterns like the batteries are more charged in the morning, due to the lack of activity during the night, and during the day, while the number of trips increase, the battery levels in general decrease.

Predictive maintenance can shift the paradigm in maintenance. By harnessing data-driven insights, we can predict maintenance needs for critical components like brakes and wheels before they become safety concerns. This proactive approach not only saves resources but also ensures the safety and reliability of the bike fleet. In this line, we have worked with multiple models, classifiers, and time series, but finally, survival analysis was the method applied in this use case. Survival analysis techniques has yielded very good results for this problem. We combine both mechanical and electrical bikes in our model, which have different mobility patterns and maintenance needs, making our sample nonlinear. However, the deep learning model, Deep Surv, successfully captured the nonlinearities and achieved very good performances, even with the inherent randomness in bike part failures, which poses great challenges to predictions. Furthermore, the wide dispersion in bike part lifespans, particularly for wheels and brakes, added to the complexity of the modelling task.

To make data and model insights accessible, we developed a user-friendly dashboard. It offers visualizations, showcasing key metrics related to bike sharing, battery levels, maintenance alerts, and more. This facilitates potential stakeholders in making informed decisions, optimizing bike operations and maintenance efforts. All components of our project, including models, dashboards, and APIs, have been seamlessly integrated into an app store within the mobility data space. This centralized hub enables easy access to our tools and insights for various stakeholders, from city planners to bike users.

This project can have positive implications for sustainable mobility in Barcelona, ranging from reducing the environmental impact by optimizing electric bike usage, thus decreasing the carbon footprint associated with traditional transportation methods, to enhancing commuters' experiences. Predictive models and maintenance strategies improve the reliability and convenience of bike sharing, making it a more attractive option for commuters. Ultimately, by promoting active transport and boosting the use of bikes as a sustainable mode of transportation, we contribute to healthier, more active lifestyles for residents and visitors.

Overall, our prototype is a first step in demonstrating the strong potential of AI and big data solutions to extract actionable knowledge from mobility data and promote more sustainable urban mobility. By combining data-driven insights with predictive models and user-friendly interfaces, we can transform bike sharing systems into a more efficient, reliable, and eco-friendly mode of transportation. Our integrated approach serves as a model for other cities seeking to promote sustainability and enhance the urban commuting experience.

Regarding the second case study on building and intelligent recommender to indicate where and when

it is convenient to install virtual stations around the city, an innovative methodology to predict the neighbourhoods and time zones where these unexpected events might happen has been developed and implemented in a prototype. The i-ViSta4bike demonstrates the viability of building such a system and introduces the novelty of providing textual interpretations of the recommendations including the events published in the agenda of Barcelona. The results of the predictive models are acceptable, even if we have been training the machine learning models with quite few datasets coming after the pandemics. The proposed solution is a combination of graph theory, advanced non-parametric statistics infer a frequentist approach, case-based reasoning and regulars expressions management, so that the proposed solution can provide the predictions for special cases with one day anticipation. The proposed solution is precomputing the learning part and building a software that incorporated the results of the learning and do not show adaptive capacity in principle, but it can easily have it if in the future the size of dataset allows it. In this moment, the time required to elaborate the predictions is almost instantaneous thanks to the precomputation of the machine learning models in the training step.

# Recommendations for future activities

7

## 7.1. Benefits of advancing in this project

The services developed in this project provide a solid foundation for implementing data-driven decision-making systems in bike-sharing networks. These services can be instrumental in designing strategies for managing data, understanding mobility patterns, monitoring bike availability based on battery levels, conducting predictive maintenance, and ultimately promoting sustainable mobility.

The primary benefits of advancing this project include:

- Knowledge Base on Data-Driven Technologies: New projects can utilize modules from this project to develop new services based on mobility data. These may include data spaces, analytical and artificial intelligence modules, dashboards, repositories, and potential new use cases.
- Knowledge Base on Mobility Models: Follow-up projects can use the resources built in this project as a knowledge base for developing enhanced mobility models. This includes incorporating new variables and delving deeper into specific aspects of mobility that can be addressed.
- Enhancing e-bike Availability with High Battery Levels: Models can automatically assess the number of bikes and battery levels at targeted stations. Integrating these models into a real system and accompanying the data outcomes with re-location actions in the field could significantly improve bike availability.
- Improving Bike Maintenance Efficiency: Models can predict, with a certain margin of error, when a particular part of a bike will need repair or replacement. Implementing this system in a real scenario could facilitate predictive maintenance, bringing associated benefits.
- Optimizing User Waiting Times: Predictive models, when integrated with decision-making systems that take action in the field, can increase bike availability either due to well-maintained bikes or optimized battery levels, thereby reducing waiting times for users.
- Incentive Sustainable Mobility: As a consequence of having a more efficient bike-sharing system with a high availability of well-maintained bikes, there is an incentive for the use of this sustainable mode of transport, which can lead to a reduction in the emission of polluting gases.

Overall, the services developed in this project represent a step forward in harnessing the power of data-driven decision-making and artificial intelligence for bike-sharing systems. This project exemplifies the pivotal role that data-driven and AI technologies play in shaping the future of mobility systems, with the potential to impact in urban transportation for the better.

## 7.2. Considerations for future developments

In case we decide to scale this project and put it into production, different factors must be considered. This section explains important considerations that affect the implementation of a follow-up project.

### A. Access to bike-sharing system data

The primary source of information for the models is mobility data from the bike sharing system. It's important to have access to this data in an efficient manner to make viable the decision-making systems works.

- Quasi-real-time data access. The current models access to monthly data, this is a dataset with all trips performed during the previous month. In development phase this approach has been useful for training and fine-tuning the models, however in a real scenario we should access to data in real time. In this way, we could generate predictions based on the current status of the bike fleet. Currently, the predictions do not consider the current status of the bike-sharing network, but a statistical approximation of the bike fleet based on the past observations.
- Data space access to data. The provider connector should be installed as close as possible to the data sources. Otherwise, it must be ensured that this connector can have direct access to the data. This proximity to data sources helps optimize data access and performance. Currently, data space is no connected to the provider servers.
- Usage policy definition and enforcement. The data space components offer the functionalities to define and enforce usage policies by the data provider, embedding the ad-hoc policies and restrictions to the shared data. This option can be further explored in order to share the data with a wider audience and interested third parties through a public Data Space.

## B. Modelling battery levels

The approach to model battery consumption in this use case is based on a physics model instead of machine learning techniques due to the insufficient resolution of the battery status data. We do not have actual data on the battery status to carry out a data-driven approach. In the current model, we do not have information on the status of batteries when they are in the docks or the battery consumption of trips, hindering the implementation of fully data-driven approaches. We only have access to snapshots every eight hours (and a short period of two weeks with snapshots every 30 minutes). The main drawback of snapshots is that we do not have information when the bike is in movement; only when the bike is docked. All of this has a very important impact on the way we model batteries. This snapshot approach also presents significant anomalies, with consecutive snapshots having the same exact values of battery level, voltage, or temperature and bikes in rest that do not increase their battery levels.

In case a follow-up project is planned, we should define a different way to collect data on battery status. The ideal scenario is one where we track the battery status constantly. If this is not possible, at the very least, we would need to track the battery status when the bikes are docked. The worst-case scenario is one where we still have snapshots of the battery status more often, every 5 or 10 minutes, and during a larger period; however, snapshots are not the best way to assess battery levels. The more realistic approach, however, is collecting battery level logs at the end of each trip, as well as a reference database of key battery information such as year of manufacture, charge cycles experienced, and the duration of those cycles.

## C. Predictive mobility models

The mobility models developed combine mobility data with battery data. The models can be applied in real scenarios; however, there are some considerations due to their design. Both models are built based on trip data, which is an origin-destination matrix. The path between the origin and destination is unknown so in our case is calculated as the shortest path between both stations.

Both models have particularities in case they can be scaled up.

- The microscopic model should be retrained with new data and could be refined based on different parameters. The current version is a general model that predicts mobility based on the day of the week and the current hour; however, it can be refined with other parameters if there is enough data to disaggregate the results. This model is entirely data-driven, and in the case that it is applied to another context, the models have to be computed again from scratch capturing the particularities of the new context.
- The Markov model, however, needs actual initial conditions to deploy its maximum potential. The model simulates different scenarios given a status of the bike fleet, so it has great potential for scenario building but needs further development depending on where and how it is to be applied. Since it is mainly a tool for short term forecasting, accessing the status of the bike park in real time becomes essential. The current version predicts mobility and the level of battery based on the day of the week and hours. Markov models are quite computationally costly, so these models need to be optimized for a very specific use case.

Moreover, the enrichment of predictive mobility models can be added with the so-called latent demand or hidden demand, which consists of the one hand of the number of potential trips that would occur if empty events did not occur in certain stations. at certain times, and the number of trips that are lengthened due to not finding available spaces to park the bicycles. For the first part, the data from the bicycle reservation application available in the Bicing application could be incorporated, and for the second, the data on time credits granted per full station (+10 minutes at no cost to search for a nearby station to park).

## D. Predictive maintenance

The predictive maintenance models developed in this project allow us to approximate when a particular part of a specific bike will need to be changed. The information required for this prediction is based on the bike's typology (mechanical or electric), its usage (distance, elevation, etc.), weather data, and its historical repair data. These models also have some particularities that should be taken into consideration if it is to be improved and put into production.

- The final models are trained for three specific parts of the bike. In the event that these models need to be applied to another bike part, the results become uncertain. During the project, we have tested other model architectures. Since each part is affected by usage in different ways, these architectures can also perform differently.
- The outcome of the model is the number of days that a particular part of the bike can work before requiring maintenance. This is based on the data we have, which essentially includes the time elapsed between one repair and the next. If we want to predict other metrics, such as the number of times a part of the bike can resist usage, we need to collect additional data. For instance, if we want to predict how many times a brake is actuated, we currently lack that information.
- The models also function without any data regarding the actual use of the components. We do not have data on the temperature of brakes or batteries, or the paths followed by bikes, which can affect wheels, among other things. To build more precise predictive models, we should explore the possibility of adding sensors to the bikes to collect technical data that could be useful, such as temperature, wheel pressure, user weight, etc.
- The breakdown dynamics are influenced by crucial factors such as the terrain type and the distance and speed of trips. However, in this project, trip paths and speeds were not directly available. Instead, they were inferred by calculating the shortest paths between stations. Consequently, the inclusion of GPS signals for the bikes could potentially enhance the accuracy of predictions.

## E. Predicting the unexpected events of the city and recommending virtual station installation.

The prove of concept presented shows the viability of using these types of solutions and enables some additional aspects to be considered to scale the solution to a broader scope:

- Train the predictive models again if new data becomes available.
- Introduce additional information about the trips (like the one about the kind of user, available in the trips dataset) to be able to establish richer interpretations of what happens and why a Bicing stations collapse.
- Include rash hours in the time zone model  7 a 9, de 13 a 15 de 17 a 19.
- Take into account the altitude for the predictive models.
- Allow the user to rename the files with results that the system produces.
- Integrate other agendas like the one with sportive events with number of assistants and consider the possibility to use them in the explanatory functions of the system.
- Add a new functionality to the i-ViSta4bike system so that a visualization in a map of Bicing stations one can quickly visualize for a certain time zone all the stations in the city that received a prediction of unexpected event. This is an alternative way to display results that we already can calculate and are currently displaying in csv files. Also, the visualization can incorporate an image of Barcelona city as a background and show the information on the real locations of the city. In fact here it would be also possible to connect the reasoning subsystem to a GIS tool so that real geographical information becomes lighter.
- In the future it might also be interesting to allow the user to configurate the delta of the neighbourhood frontiers and to see them in the visualizations of the trips in the different neighbourhoods.
- Integrate the system in the dataspace built for case study 1.
- Make the system adaptive and provide the mechanisms to make it learning along the time. This requires updating the learning matrices incrementally.
- Establish a guideline so that the models of learning can be properly retrained periodically.

## F. Data collection

The data utilized during the project was opportunistically collected and models were adapted to it, but it is not data collected with the objective of developing a data-driven decision-making system. If the project scales up, there is a need to develop a plan for establishing a data collection and sharing structure and services to access the data required for the models. Furthermore, it's important to acknowledge that incorporating additional information not currently available to the project could significantly enhance predictions. For instance, the inclusion of physical sensors on the bikes could provide insights into the stress levels experienced by the riders. Similarly, having access to data about the specific routes people take while cycling could offer a more accurate assessment of bike usage patterns, including factors such as terrain conditions.

# Deliverables

| Deliverable | Responsible | Dissemination Level |
|---|---|---|
| D1. Data space infrastructure | i2CAT | CO |
| Description: The Data Spaces infrastructure consists of two connectors, a provider, and a consumer, along with a DAPS (Identity Provider) and an App Store. The provider connector is responsible for obtaining Bicing data through a specially created API and making it available to consumers. The App Store, which includes a Harbor for storing applications, contains all the apps and their metadata so that they can be deployed by the consumer connector. The consumer connector not only retrieves data through the provider but also enables the use of the apps registered in the App Store. Finally, the DAPS ensures that all communications are secure. | | |
| D2. ViSta4Bike | IDEAI-UPC | CO |
| Description: Software that includes the entire developments of the predictive model for the abnormal unexpected behaviour of Bicing stations. It is an intelligent decision support system that includes a case-based reasoning model to recommend the installation of virtual e-bike stations the previous day of expected need. These recommendations include an explanation about the events registered in the public agenda of the city. The system is also visualizing the Bicing trips of a given district. | | |
| D3. Predictive models | Eurecat | CO |
| Description: Three predictive models have been delivered. Two models for predicting mobility and battery levels, the first one is a micro-mobility model and the second one Markov model, among other parameters they provide, given an initial state of the levels of batteries on destination. The models have been dockerized and integrated into the Data Space App Store and the code is stored in Eurecat's servers. A third model has been developed for predictive maintenance, a DeepSurv model has also been dockerized and integrated into the Data Space App Store, and the code is stored in Eurecat's servers. | | |
| D4. Dashboard | Eurecat | CO |
| Description: Dashboard developed with Dash and Flask API to access the databases where the data generated by predictive models is stored. The dashboard presents indicators of the three predictive models delivered in D3. These applications are integrated into the Data Space in the App Store, and the source code is stored in Eurecat's servers. | | |
| D5. Exploitation plan | Eurecat | CO |
| Description: This report presents a preliminary approach to potential business models for the service and a market overview. The market overview mainly analyses potential competitors and customers across Europe. In addition, a characterization of the main Key Exploitable Results (KERs) obtained through the project development by the different partners has been carried out and it has been studied the most appropriate mechanisms to protect the IPR in each stage of the current and further development of the product. | | |
| D6. Final Report | Eurecat | PU |
| Description: This report summarizes the entire project's objectives, process, outcomes, findings, and future directions. It serves as a record of the project's completion and includes an introduction, description of work performed, key results, conclusions, and recommendations for future activities in the field of artificial intelligence applications for bike-sharing systems. | | |

# Authoring

Technical report written by:

Julià Vicens (Eurecat – Project Coordinator)

Aleix Bassolas (Eurecat)

Jordi Grau (Eurecat)

Daniel Alzueta (i2CAT)

Eros Zaupa (i2CAT)

Rizkallah Touma (i2CAT)

Roger Calaf (Eurecat)

Estefanía de María Bellassai (Eurecat)

Karina Gibert (IDEAI-UPC)

Sergi Ramírez (IDEAI-UPC)

Miquel Umbert (IDEAI-UPC)

Hassnain Shafqat (IDEAI-UPC)

Umut Cindiloglu (IDEAI-UPC)

Reviewed by:

Irene Giménez (Pedalem)

Marco Orellana (CIDAI)

Ángel Martín (i2CAT)

Joan Mas-Albaigès (CIDAI)

# Bibliography

10

Burani, E., Cabri, G., & Leoncini, M. (2022). An algorithm to predict e-bike power consumption based on planned routes. Electronics, 11(7), 1105.

Steyn, W. J. V., & Warnich, J. (2014). Comparison of tyre rolling resistance for different mountain bike tyre diameters and surface conditions. South African Journal for Research in Sport, Physical Education and Recreation, 36(2), 179-193.

Fernández, A., Billhardt, H., Ossowski, S., & Sánchez, Ó. (2020). Bike3S: A tool for bike sharing systems simulation. Journal of Simulation, 14(4), 278-294.

Fernández, R. (2010). Modelling public transport stops by microscopic simulation. Transportation Research Part C: Emerging Technologies, 18(6), 856-868.

Azevedo, C. L., Deshmukh, N. M., Marimuthu, B., Oh, S., Marczuk, K., Soh, H., ... & Ben-Akiva, M. E. (2017). Simmobility short-term: An integrated microscopic mobility simulator. Transportation Research Record, 2622(1), 13-23.

Lu, X., Wetter, E., Bharti, N., Tatem, A. J., & Bengtsson, L. (2013). Approaching the limit of predictability in human mobility. Scientific reports, 3(1), 2923.

Huang, W., Li, S., Liu, X., & Ban, Y. (2015). Predicting human mobility with activity changes. International Journal of Geographical Information Science, 29(9), 1569-1587.

Cox, D. R. (1972). Regression models and lifetables. Journal of the Royal Statistical Society: Series B (Methodological), 34(2), 187-202.

Leung, K. M., Elashoff, R. M., & Afifi, A. A. (1997). Censoring issues in survival analysis. Annual review of public health, 18(1), 83-104.

Gijbels, I. (2010). Censored data. Wiley Interdisciplinary Reviews: Computational Statistics, 2(2), 178-188.

Wang, P., Li, Y., & Reddy, C. K. (2019). Machine learning for survival analysis: A survey. ACM Computing Surveys (CSUR), 51(6), 1-36.

Yu, C. N., Greiner, R., Lin, H. C., & Baracos, V. (2011). Learning patient-specific cancer survival distributions as a sequence of dependent regressors. Advances in neural information processing systems, 24.

Wright, M. N., Dankowski, T., & Ziegler, A. (2017). Unbiased split variable selection for random survival forests using maximally selected rank statistics. Statistics in medicine, 36(8), 1272-1284.

Katzman, J. L., Shaham, U., Cloninger, A., Bates, J., Jiang, T., & Kluger, Y. (2018). DeepSurv: personalized treatment recommender system using a Cox proportional hazards deep neural network. BMC medical research methodology, 18(1), 1-12.

# Change control

# 11

| Version | Date | Modified section | Brief description of the modification |
|---|---|---|---|
| 0.1 | 30/06/23 | Table of contents, Project scope and Objectives, Work performed | Eurecat creates the structure of the report, writes up the content in the specified sections and distributes tasks among the partners. |
| 0.2 | 28/09/23 | Work performed and Key results | Eurecat and i2CAT write up the content for the mentioned sections. |
| 0.3 | 04/10/23 | Introduction, Conclusions and Recommendations for future activities | Eurecat write up the content for the mentioned sections |
| 0.4 | 17/10/23 | All sections | Reviewed by Eurecat, i2CAT and Pedalem |
| 1.0 | 17/10/23 | All sections | Comments and suggestions addressed by Eurecat |
| 1.1 | 02/11/23 | Section 4.3 included | Inclusion of reporting by IDEAI-UPC and overall harmonization of documents contents |

# CIDAI Centre of Innovation for Data tech and Artificial Intelligence

C/Bilbao, 72 Edif. A.
08005 - Barcelona
Tel. +34 93 7419 100
info@cidai-catalonia-ai.eu
www.cidai.eu

Promotors

Membres