

## LIS\_07: To create a new Multimodal Digital Mobility Services (MDMS)

### Description of the measure and main outcomes expected

Transportes Metropolitanos de Lisboa (TML) has been studying the global state of the art of a MaaS/MDMS platform in general, and its application in the Lisbon metropolitan area. Under this measure, developments were envisaged with implementation carried out in different stages, aiming at the integration and unification of the different transport services on a single platform, offering simpler access to mobility, with real-time passenger information and access to different integrated mobility services, both existing as well as those to be developed.

Currently, there are some MaaS platforms in use, belonging to different companies, but with very limited functionalities, serving only to provide information about their transport services or to load passes for transport network users.

TML, as the authority responsible for managing the ticketing system in the Lisbon metropolitan area, intends to develop a comprehensive MaaS service that will serve all transport users in an integrated and metropolitan way. This service will promote simpler and more agile access to mobility, contributing to greater adoption of public transport, a significant increase in demand and decarbonization of cities, with the consequent reduction of emissions through the adoption of more sustainable mobility to the detriment of individual transport.

### Preparation of the measure

TML designed a MaaS platform of a metropolitan nature, with the official brand of the navigation system®, a system where all transport operators operating in the 18 municipalities of the Lisbon metropolitan area cooperate. The platform consists of a mobile application, available for Android and iOS, for information and consumption of mobility services that incorporate the following features:

- a. Customer Registration and Authentication.
- b. Reserved area:
  - i. User Account.
  - ii. Online Ticket Office.
  - iii. Interactions.
- c. Transport Network and Services.
- d. Ticketing System.
- e. Online Counter.
- f. Auto News and Fines Management.
- g. Online Surveys.
- h. Travel Planner.
- i. Visitor/Tourist.



- j. TML Services.
- k. Institutional Information.
- l. Other Features.

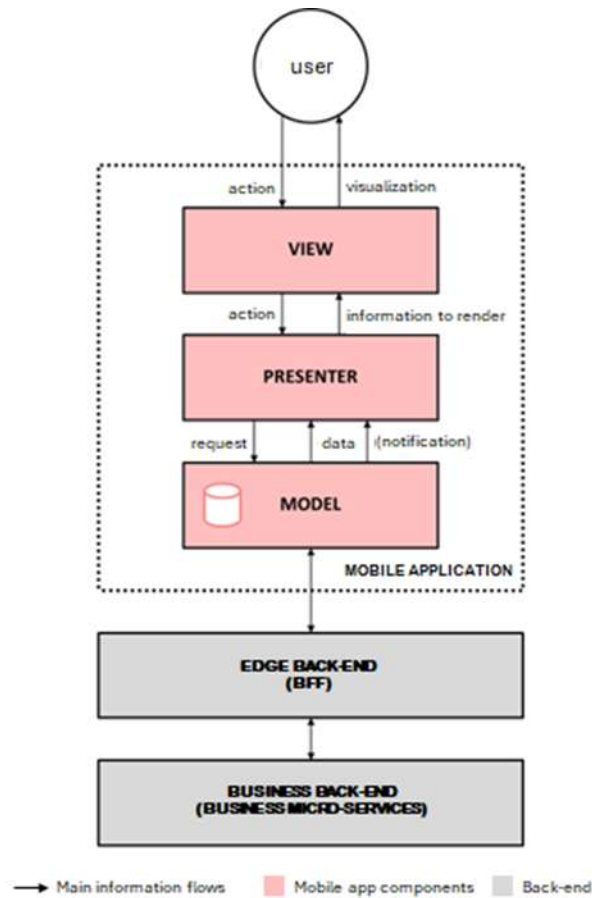
The Mobile Application is intended to run on smartphones, tablets or similar equipment, using Back-End Micro-Services. Its implementation must be carried out using a modular development pattern.

Aiming for a balanced distribution of functions, the implementation of the Application must follow an MVP (Model – View – Presenter) type pattern, in which the business logic and data management, interaction and usage logic functions are compartmentalized into blocks well defined:

- View - Receives the actions triggered through the user interface and updates it accordingly, through the mediation, in cascade, of the remaining modules. Unlike the MVC pattern, this interface is passive, receiving all the information ready to render, meaning there is no connection between the View and Model blocks.
- Presenter - Interprets the actions triggered by users, uses the Model to process them and returns the information for the user to view, to be rendered by the Vision.
- Model - Implements the business rules associated with the functions required by the Presenter, using external services if necessary.

It is intended that the implementations, on Android and iOS, are as identical as possible, both in terms of architecture and development patterns, as well as the nomenclature of Classes, Methods and Fields.

Taking into account the implementation and evolution planned for the Mobile Application (using the smartphone's internal devices, such as GPS, camera, NFC, Bluetooth, etc.) development will be carried out in the development environments and languages native to each operating system (Android and iOS). The architecture of the application is presented in Figure 1.



**Figure 1.** Architecture of the Mobile Application.

The development of the Application complied with the following requirements shown in Table 1:

**Table 1.** Requirements of the application.

Requirement	Description
Operating System	Android and iOS
Programming language	Android: Kotlin or Java. iOS: Swift.
Development platform	Android: Android Studio. iOS: Xcode.

Having identified the requirements, a development and implementation plan was defined with an evolutionary perspective in three main phases and a new subsequent phase.

Phase 1: Reading and loading prepaid passes and titles onto personalized Calypso cards, activating access to discounts that grant free access to young students, allowing the pass to be loaded at no cost and access to useful information about transport (cards, discounts, transport tickets, timetables, routing to the passenger support area, location of stores and in-person service points, trip planner, FAQs and news).



Phase 2: Access to the registered area with information on trips made, card request, updating customer information, adding favourites, receiving notices and alerts. Network map with real-time timetable information. Provision of information in more languages, information dedicated to the visitor.

Phase 3: Card virtualization. App starts to function as support for validating the title. Connection to a CRM with a record of customer interactions.

Phase 4: Possibility of loading cards for occasional use.

In April 2024, TML concluded the first phase of development and in May of the same year, it began Phase 0, of internal testing, launching the possibility of extended testing to a set of 991 Beta Testers who tested the application and gave their feedback. contributions.

Currently, Phase 1 has been completed with very positive results. In the first month, 7 thousand people used the Navigator App and in June the number rose to 20 thousand.

### **Challenges & Mitigations**

To mitigate some of the challenges, testing of the app was launched to a group of Beta Testers to whom a satisfaction survey was subsequently launched. 374 responses were obtained, of which 76% would give a score above four, on a scale of zero to five; 96% considered the app to be very intuitive and 70% indicated that the app lived up to expectations.

A brainstorming session was also held using the design thinking method with some beta testers to discuss identified improvement points and envision the next phases of evolution.

In this session, some new aspects were identified to be incorporated as improvements for Phase 1 and rethinking some developments in Phase 2, to implement in 2025. One of the changes to Phase 2, still under analysis, could be the dismissal of a dedicated planner, as it is concluded that users already use official planners, perhaps requiring a redirect instead of incorporating a planner into the application itself.

One of the solutions to be developed that seems to be more suited to users' needs will be the construction of a dedicated network map with information on all modes of transport in real time that pass through a given stop/station/terminal, as this was one of the services with greater representation in terms of needs to be served, identified in the brainstorming session.

TML has collected information from some providers of MaaS applications and services, such as Moovit, CityMapper, Trafi, Transit, InstantSystems and HenseCom. Some of these MaaS application providers and service providers are very advanced in route planning, as they incorporate the entire PT network and some new innovative mobility services, allowing for good route planning, but they do not yet sell tickets or services. Other applications, such as UBER and BOLT were also contacted. They have a different strategy, as they started to provide a mobility service (ridehailing or shared services), using the platform to sell it, and are building the route planner and aggregating other services, thus providing less interesting route planning, but already selling services.



Figure 2. App store and Play Store promotion.

### Next steps towards implementation

The navegante® App is already on the market. Other developments include debugging, usability improvements and communication campaigns to extend its use to a greater number of people