



**Interreg**



MARITTIMO-IT FR-MARITIME



**MOBI  
MART**

# PROJET MOBIMART

## T 2.2.1 Ensemble d'interfaces de programmation d'applications (API)

Maggio- Mai 2020



La cooperazione al cuore del Mediterraneo  
La coopération au cœur de la Méditerranée

Projet n°	168
Acronyme	MOBIMART
Titre du projet	MOBilité Intelligente MARE Terra
Début/Fin du projet	01.04.2018- 31.03.2021
Durée	36 mois

Composante	T 2
Activité à laquelle se réfère le livrable	Développement d'interfaces de programmation d'application (API)
Titre du livrable	Test de fonctionnement de la plateforme nuage.
Phase de référence	P4
Période de référence (début/fin)	1/09/2019 – 31/03/2020

Auteur	Silvia Antonelli, Alexandre Tardo, Domenico Lattuca
Version	2019-2 (deuxième plan de développement)
Date	31/5/2019
Responsable de la validation	Paolo Pagano
Date de révision	
Récapitulation des modifications	
Auteur des modifications	

## Histoire du document

Version	Date	Notes
1.0	21/01/2020	Réalisation de la structure du document
1.1	27/01/2020	Ecriture de la procédure de test
1.1	05/02/2020	Achèvement du livrable, première révision
1.2	10/02/2020	Deuxième révision
1.3	27/05/2020	Mise à jour du paragraphe « Mise en œuvre des API »

## Bibliographie

Réf.	Objet	Lien
1	Rapport Task Force 17 TM2.0 – MaaS (ébauche)	-
2	PMIS	<a href="http://www.guardiacostiera.gov.it/mezzi-e-tecnologie/Pages/PMIS.aspx">http://www.guardiacostiera.gov.it/mezzi-e-tecnologie/Pages/PMIS.aspx</a>

## Indice/Table des matières

<b>Introduction</b>	<b>page 2</b>
<b>La plateforme « multi-port »: l'intégration des ports de Cavo, Portoferraio et Capraia Isola</b>	<b>page 4</b>
<b>Mise en œuvre des API</b>	<b>page 5</b>

## 1. Introduction

Depuis sa création, la plateforme MOBIMART a connu un processus d'évolution continue et rapide. L'idée des développeurs a toujours été de créer un point de collecte de données, lié aux principaux ports de la Toscane, capable de changer en s'adaptant aux principales tendances du marché.

Partant de ce principe, une fois définis et testés les services offerts par la plateforme MOBIMART pour le port de Livourne, l'étape suivante a élargi l'offre en ajoutant l'ensemble des ports toscans qui, avec le port de Livourne, constituent l'épine dorsale du trafic régional de marchandises et de passagers.

Cependant, le processus d'évolution de la plateforme ne s'est pas arrêté avec l'ajout des nouveaux ports mentionnés ci-dessus. En fait, deux API ont été mises en place : des services pour améliorer la fonctionnalité de MOBIMART, qui est maintenant capable d'offrir de nouvelles méthodes de téléchargement automatique, par exemple, à un développeur intéressé par la liste des navires au départ d'un port donné

En outre, contrairement à la version précédente de la plateforme, qui ne pouvait afficher que les navires au départ du port de Livourne, la version actuelle est en mesure de fournir la liste actualisée tant des navires au départ d'un port spécifique, sélectionné par l'utilisateur, que des navires arrivant dans un port spécifique également sélectionné par l'utilisateur. Enfin, une autre mise à jour concerne l'insertion de nouveaux fichiers GTFS spécifiques aux ports de Piombino et Portoferraio (la figure 1 montre une capture d'écran de la plateforme).

Le choix du logiciel de test s'est porté sur un outil entièrement open source, tel que Static GTFS Manager, capable d'importer un fichier GTFS et d'afficher son contenu en recréant automatiquement les différents tableaux qui composent le fichier inséré, de générer les dépendances et d'afficher le contenu (en affichant sur une carte les différents ports et les différentes routes à l'aide des fichiers contenus dans l'archive GTFS). Il s'agit donc d'un excellent outil capable de mesurer la qualité des données générées par la plateforme.

Enfin, le résultat du test s'est avéré parfaitement conforme à nos attentes. Cela n'exclut pas de nouvelles modifications à l'avenir pour améliorer encore la qualité des données dont nous disposons ou pour augmenter la quantité de données.

GTFS Livorno GTFS Piombino GTFS Portoferraio

LIVORNO  PARTENZE  Data  Search:

Show 10 entries

DATA	ORARIO	COMPAGNIA	NOME NAVE	PORTO DI DESTINAZIONE/ARRIVO	COD. PORTO	BANCHINA
27/05/2020	04:45	GRIMALDI LINES	EUROCARGO CAGLIARI	SAVONA	ITSVN	24/SUD
27/05/2020	08:30	TOREMAR	LIBURNA	CAPRAIA	ITCPA	66/NORD/R
27/05/2020	10:30	GRIMALDI LINES	EUROCARGO BARI	SALERNO	ITSAL	24/NORD
27/05/2020	11:00	GRIMALDI LINES	CRUISE BONARIA	OLBIA	ITOLB	14/F/R
27/05/2020	13:30	SARDINIA FERRIES	PASCAL LOTA	BASTIA	FRBIA	62/R
27/05/2020	13:30	SARDINIA FERRIES	MEGA EXPRESS	BASTIA	FRBIA	62/R
27/05/2020	16:30	TOREMAR	LIBURNA	CAPRAIA	ITCPA	66/NORD/R
27/05/2020	20:00	TIRRENIA	BENIAMINO CARNEVALE	CAGLIARI	ITCAG	20/R
27/05/2020	21:00	SARDINIA FERRIES	CORSICA MARINA SECONDA	GOLFO ARANCI	ITGAI	64/NORD/R
27/05/2020	21:30	GRIMALDI LINES	EUROCARGO PALERMO	SAVONA	ITSVN	24/NORD

Showing 1 to 10 of 39 entries 1 2 3 4



**Figure 1 – Plateforme Mobimart actualisée**

## 2. La plateforme « multi-port » : l'intégration des ports de Cavo, Portoferraio et Capraia Isola

La première version de la plateforme MOBIMART prévoyait que tous les services offerts ne concerneraient que le port de Livourne, en tant que centre névralgique et centre opérationnel du trafic portuaire de la région Toscane. Ce choix s'est traduit, d'une part, par la création d'un tableau horaire contenant différents types d'informations concernant les navires au départ du port de Livourne (par exemple, date, heure, compagnie, quai) et, d'autre part, par la création d'un fichier General Transit Feed Specification (GTFS) qui peut être téléchargé par un utilisateur désireux de connaître les données relatives aux navires passant par le port en question.

Ce qui vient d'être décrit ne représente que la première étape de la réalisation de la plateforme MOBIMART, après quoi une étape de tests a été lancée, achevée avec succès et dont les résultats seront résumés ci-après. La prochaine étape concernant l'extension des services, déjà présents et activés pour le port de Livourne, aux autres ports de la Toscane s'est donc avérée presque physiologique.

La liste des ports intégrés, accompagnée d'une brève description de chacun d'entre eux, est détaillée ci-dessous :

Port de Piombino : il s'agit d'un port polyvalent, principalement dédié au trafic de passagers avec l'île d'Elbe, la Sardaigne et la Corse (de façon saisonnière) et au trafic de fret, principalement en vrac, destiné aux marchés nationaux et internationaux ;

Port de Portoferraio : c'est le principal port d'escale de l'île d'Elbe, situé sur la côte nord dans une position bien protégée au sein d'une grande baie. Son utilisation est étroitement liée au trafic régulier des ferries avec Piombino, au trafic des croisières et au trafic des bateaux de plaisance ;

Port de Rio Marina : il a acquis dans le passé une importance considérable grâce à l'exportation du minerai de fer extrait dans les carrières qui dominent la ville de Rio Marina. Aujourd'hui, l'escale consiste essentiellement en un terminal pour les ferries, en liaison avec le port de Piombino, qui sont amarrés dans la dernière partie du barrage (environ 80 m de long) ;

Pontile di Cavo : c'est un port touristique, avec quelques fonctions commerciales, qui accueille le poste d'amarrage des hydrofoils réguliers pour la ligne Piombino-Cavo-Portoferraio et en été le poste d'amarrage des ferries sur la ligne Piombino-Cavo ;

Port de Capraia Isola : Capraia est par extension la troisième île de l'archipel toscan avec une superficie de 19,3 kilomètres carrés. En vertu de la distance élevée de la côte continentale située à l'est (54 km) est l'une des îles les plus éloignées de la péninsule italienne.

Après cette brève introduction concernant les ports qui ont ensuite rejoint la plateforme MOBIMART, on va décrire les différentes étapes de mise en œuvre des nouveaux services.

### 3. Mise en œuvre des API

Les API (acronyme de Application Programming Interface – interface de programmation d'applications) sont des ensembles de définitions et de protocoles à travers lesquels les logiciels d'application sont développés et intégrés. Les API permettent la communication entre différents services sans qu'il soit nécessaire de savoir comment ces services sont mis en œuvre, ce qui simplifie le développement des différentes applications et permet un gain net de temps et d'argent.

Lorsqu'il s'agit de créer de nouveaux outils et produits ou de gérer des produits existants, les API offrent une certaine souplesse, simplifient la conception, l'administration et l'utilisation et offrent des possibilités d'innovation.

Dans le cas de la plateforme MOBIMART, la mise en œuvre des API décrites ci-après a été rendue possible par l'utilisation de la spécification OpenAPI (par le passé « Swagger Specification »), c'est-à-dire un format de description pour les API de type REST (acronyme de "REpresentational State Transfer") qui indique une représentation du transfert d'état d'une donnée spécifique.

Le processus de mise en œuvre d'une API consiste à créer une série de points terminaux (des URL) auxquels correspondent certaines opérations qui sont effectuées en fonction des demandes d'un programme externe, permettant ainsi de répondre aux demandes faites par un développeur.

Un fichier écrit selon la spécification « OpenAPI » permet de décrire l'ensemble des API :  
les points terminaux (/ utilisateurs) disponibles et les opérations possibles sur chacun de ces points terminaux (GET / utilisateurs, POST / utilisateurs) ;  
Paramètres opérationnels d'entrée et de sortie pour chaque opération ;  
Méthodes d'authentification ;  
Informations de contact, licence, conditions d'utilisation et/ou autres informations.

Il est possible d'écrire les spécifications au format tant YAML que JSON : tous les deux sont faciles à apprendre et très lisibles pour l'homme et la machine. Une fois que la spécification OpenAPI et les outils Swagger (utiles pour la conception, la construction et la documentation des API REST) ont été définis, le développement peut être articulé de différentes manières.

Dans le cas de la plateforme MOBIMART, le choix a été d'écrire les API en utilisant un format YAML comme référence. Les points terminaux créés et mis à la disposition de l'utilisateur qui veut utiliser les API prévues par le projet sont les suivants :

Point terminal /départs/{port} : lorsqu'il est interrogé, il fournit comme résultat la liste des navires au départ d'un port, passé comme paramètre, à partir d'une date saisie par l'utilisateur (ou de la date courante si aucune date n'est fournie) jusqu'au dernier jour présent dans la prévision ;

Point terminal /gtfs/{port} : lorsqu'il est interrogé, il fournit comme résultat le fichier GTFS (codifié en JSON) des navires partant d'un port et arrivant dans un autre port, tous les deux passés comme paramètres, dans la période spécifiée par l'utilisateur (ou de la date courante jusqu'au dernier jour présent dans la prévision si aucune période n'est spécifiée).

La figure 2 montre la description détaillée des deux points terminaux que l'on vient de mentionner, ainsi que leurs paramètres et les réponses qui reviennent en cas de succès ou d'échec de l'interrogation.

```

/departures:
  get:
    description: 'Returns the list of departures from port specified in {port} and starting from current date-time or from that (eventually) specified in the query'
    parameters:
      - $ref: '#/components/parameters/PortName'
      - $ref: '#/components/parameters/DeparturesLimit'
      - $ref: '#/components/parameters/StartingDepartureDate'
      - $ref: '#/components/parameters/ArrivalPortName'
    responses:
      '200':
        description: Successfully return a list of departures
        content:
          application/json:
            schema:
              type: array
              items:
                $ref: '#/components/schemas/Departure'
      '400':
        $ref: '#/components/responses/400Error'

/gtfs:
  get:
    description: "Returns a set of GTFS encoded in JSON syntax"
    parameters:
      - $ref: '#/components/parameters/PortName'
    responses:
      '200':
        description: Successfully return a set of JSON GTFS
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/GTFS'
      '400':
        $ref: '#/components/responses/400Error'

```

Figure 2 – Structure des points terminaux

La figure 2 montre les caractéristiques que doivent avoir les paramètres « passés » par l'utilisateur en plus du schéma des deux points terminaux, c'est-à-dire le type des différents champs qui les composent et l'indication des champs obligatoires.

Pour entrer dans le détail, les paramètres requis ne sont que cinq et sont énumérés ci-dessous :

- PortName : représente le nom du port de départ ;
- StartingDepartureDate : date initiale saisie par l'utilisateur et de référence pour l'interrogation ;
- DepartureLimit : date de référence finale pour l'interrogation si aucune date de début n'est spécifiée (si cette date est postérieure à la dernière date utile dans la prévision, le résultat de l'interrogation ne prendra en compte que cette dernière) ;
- ArrivalPortName : représente le nom du port d'arrivée.

En ce qui concerne le schéma des points terminaux, il est immédiat de constater que, même si ceux derniers sont représentés par le même type d'objet, les propriétés sont sensiblement différentes.

Le point terminal pour les fichiers GTFS nécessite uniquement et exclusivement qu'un paramètre soit saisi, de sorte que l'utilisateur puisse spécifier le port de référence (dans le schéma de la figure 3 appelé PortName) pour lequel le fichier GTFS est demandé (par exemple Livourne, Piombino ou Portoferraio). Une fois interrogé, le point terminal attribuera à chaque champ une simple référence relative au fichier de texte situé dans l'archive GTFS correspondante.<sup>1</sup>

<sup>1</sup> La structure d'un fichier GTFS (*General Transit Feed Specification*) a été discutée en détail dans le précédent livrable 3.3, concernant la réalisation des interfaces d'échange et d'interopérabilité.

