Intelligent Parking Assistant - A Showcase of the MOBiNET

Platform Functionalities

Lars Mikkelsen\textsuperscript{1*}, Raphael Toledo\textsuperscript{2}, Niels Agerholm\textsuperscript{3}

1. Aalborg University, Denmark, lmm@es.aau.dk
2. Eurocom, France, toledo@eurocom.fr
3. Aalborg University, Denmark, na@civil.aau.dk

Abstract
The Intelligent Parking Assistant (IPA) system helps the user chose, find and pay for parking. IPA automatically fetches parking lot information from a relevant service, automatically assesses if the vehicle is stopped, evaluates if stopped inside a paid parking lot, and automatically initiates payment for the parking. IPA is published via the MOBiNET platform: A European-wide ITS platform offering functionalities enabling easy migration of services. Migration is enabled by defining a common methodology for publishing services, which covers interfaces, data formats used, coverage area of the service and others. IPA is implemented as proof of concept utilizing MOBiNET as a platform integration use case demonstration. The performance of IPA is evaluated in terms of correctness of decisions made during runtime and based on user input in a trial period, involving 40+ test users using the system on a regular basis.

Keywords: PARKING, MOBiNET, ITS

1 Introduction

Today the technology has evolved and it allows users to always be connected and always having a powerful and very capable device with them, namely the smartphone. The constant connectivity of mobile devices allows very high mobility of the user, which means that the services used also must support mobility, and not just mobility within a city but across countries. One particular area related to mobility is Intelligent Transportation Systems (ITS), offering services related to transportation. It is within this sub-area that the Intelligent Parking Assistant (IPA) system has been developed. The IPA system is an in-car parking guidance system that intelligently supports the users in connection with parking, more specifically in finding and selecting parking lots, and paying for parking sessions. The IPA system has been developed taking advantage of the MOBiNET[1] platform and the functionalities this
provides, ensuring that the system easily can be migrated to cities all over Europe. The MOBiNET platform aims at acting as a broker of information and services related to transportation. This is done by offering a number of functionalities to support business-to-business (B2B) and business-to-consumer (B2C) services. The services are not hosted on the platform but the platform contains information about services including pricing, coverage area, endpoint, protocol, input, output, and other[2].

Previous approaches to developing an application aimed at helping users finding and selecting parking lots and paying for parking include "SmartPark Trondheim Parkering"[3], "Parker, Find available parking"[4], "ParkMe Parking"[5], and other. Each of these offer a subset of the following: live information about availability; turn-by-turn route guidance; price information; payment in various forms; find your parked car; and other functionalities. However, where most of the existing systems do well in offering various functionalities, they are limited to operating within certain cities and areas. This could be explained with it being very demanding and cumbersome to obtain the needed information from different locations, in different languages, different pricing schemes, make agreements with local parking companies, ensure that the users can pay even though they cross country and currency borders. These are just some of the obstacles met when migrating services across countries.

In the present effort the IPA system is developed and it is described how it takes advantage of the MOBiNET platform which offers a set of very useful functionalities for developing and publishing ITS services in relation to cross-country usage. The IPA system is to a large extent based on the experiences and development made in the Danish ITS Platform project[6], which was carried out 2010-2013. This project was a large-scale field operation test with GNSS-based on board units (OBU). It consisted of a back-end server, OBUs in 425 cars, and four applications; driver log; dynamic traffic information; traffic statistics tool for road operators; and full-automatic parking payment system for the users. Around 20 million km of driving data has been collected as part of the project. The full-automatic parking payment system was only in operation for a few months. Even though it showed convincing results, much more could be experienced regarding reliability, users attitude and acceptance, and experiences from the parking attendees, just to mention a few topics.

The rest of the paper is structured as follows: Section 2 gives an overview of the components in the IPA system and the components used by it is given; Section 3 describes the functionalities of the MOBiNET platform most relevant to the IPA system; Section 4 describes the technical aspects and functionalities of the IPA system; Section 5 describes use cases of the IPA system; And Section 6 describes how the performance of the IPA system is evaluated.

2 Intelligent Parking Assistant (IPA) System Overview

The IPA system is made as a very modular system, enabling easy addition of supported areas
or regions, by adding services via the Service Directory in the MOBiNET platform. The goal of the IPA system is to help users find and select parking lots based on user input and live information about availability, guide users to the parking lot, intelligently evaluate if the user is inside parking lot when stopped, and automatically initiate a parking session and pay for it. The IPA system consists of the following components: Park Assist Android app which acts as the main user interaction interface; the Parking Manager service which contains two sub services: Parking Information Provider and Parking Session Manager; the Map Matching server providing map matching functionalities for the application; and the MOBiNET platform that acts as orchestrator and a platform for migration assistance, and where the system is published through. This is illustrated in Figure 1.

![IPA System Overview](image.png)

**Figure 1 IPA System Overview**

With this system architecture the ParkAssist Android app is in the centre of operation, as it is installed on the end user device and follows the user wherever he goes. The MOBiNET platform is the orchestrator that depending on where the user is geographically, provides the ParkAssist app with locations of the relevant back end services. That is, the Parking Manager and the Map Matching server are interchangeable and can vary from city to city and country to country.

3 **MOBiNET platform functionalities**

MOBiNET is a project aimed at making Internet of mobility for ITS services of Europe. The idea is to make a platform the one-stop shop for all services related to ITS in Europe. The
services include information provider services, front-end services, back-end services, data format translator services, billing and identity management, and other.

MOBiNET is realized via the MOBiNET platform that offers support to B2B and B2C interactions. In Figure 2 the architecture of the MOBiNET platform can be seen where the key components are: Dashboard, Service Directory, Identity Manager, Billing Manager, Software Development Kit (SDK), and MOBiAGENT. The Dashboard offers access to most of the other components for developers and service providers. The Service Directory contains metadata about services in terms of endpoint, interface type, input and output data types, ownership, pricing scheme, and many other. The Identity Manager handles the identities of both business users and private users, and the component is used both in connection with the Dashboard and by services. The Billing Manager offers payment handling for transactions for B2B and B2C service use. The SDK offers a number of support tools to help developers generate and publish services in the platform. The MOBiAGENT is the client front end software, or app, that users can install on their devices through which interface to searching for services are offered. The MOBiAGENT also offers a number of interfaces for services to use, e.g. to a communication agent for vehicle to infrastructure communication.

Figure 2 Architecture of the MOBiNET Platform[2].
In Figure 3 the main flow and interactions between Dashboard and Service Directory can be seen, along with how end user services interact with the Service Directory.

![Figure 3 Dashboard and Service Directory interactions](image)

In the scenario of the IPA system the components of the MOBiNET platform are used in the following way.

- Service description of the front-end service (app) ParkAssist is added to the Service Directory so end users can find and download the app.
- Service descriptions are added to the Service Directory of the back-end service providing parking information and parking session management, and of the Map Matching server.
- The back-end services are discovered by the app by contacting the Service Directory during runtime and querying services based on service type and the location of the user device.
- The Identity Manager is used to verify the user credentials, and to link a user with a parking bill.
- The Billing Manager ensures that transactions are performed from the user to the parking service provider.

4 Intelligent Parking Assistant System

4.1 App Functionalities

The IPA app, ParkAssist, has a number of functionalities, both in terms of communication with back-end services and user interaction. In Figure 4 an overview of the layout of the ParkAssist app can be seen.
The ParkAssist app runs under some assumptions about the context, that is needed for proper interaction with the user, and to be able to make correct decisions. It is assumed that the device is fixed within the vehicle until at least some seconds after the vehicle is stopped. It is also assumed that the screen is on and visible to the user while searching for an available parking spot and parking the car. The main interactions with the app for the user are:

- **Automatic Park & Pay** Presents all the parking lots to the user, from which he can choose to be guided to or just park.
- **Take Me To Nearest Parking** Proposes the nearest parking lot to the user, which the user can choose to be guided to.
- **Car Info** Allows the user to add and store cars and information about them.
- **User Info** Where the user can add personal information, register and log in to the system.

When either Automatic Park & Pay or Take Me To Nearest Parking is chosen, the app will initiate automatic payment whenever the car is stopped within a parking lot. In order to be able to use the automatic payment functionality the user need to identify himself in terms of a user id and password, and enter the license plate of the vehicle used. This is a one time operation performed by the user. This information are used for identification with the Parking
Session Manager service, to let it know which user to bill for the parking sessions. The license plate is used in connection with starting and stopping parking sessions at the parking session manager service. In this way it is possible for a user to use the system with multiple vehicles but with the same user credentials. For this reason the app contains two pages where the user can input and manage information. A page for user identification and information, and another page for car information and details. In the latter page it is possible to see the location of the car, based on information the user has provided about the car. This will also be used in connection with parking where the GNSS location of where the car is parked will be saved.

One of the key back-end communication functionalities of the app is that it communicates with a Parking Information Provider service and fetches the available parking lot information from the current relevant parking information service. The Parking Information Provider service is chosen by making a service discovery query to the Service Directory in the MOBiNET platform. The query includes service type and the location of the device to match service coverage area. If there is no match the search area is expanded. If a service matches the discovery query it is queried for live availability information of the parking lots covered by the service. The information is fetched from the service when initially discovered when the app is opened, and periodically hereafter when the map is open in the app and when any kind of search for parking is performed.

A parking session is automatically initiated with the Parking Session Manager if the car is evaluated to be stopped, at which point the current location is compared to the area of the parking lot. Whether the car is stopped is evaluated based on the GNSS location samples and information about velocity. The location is sampled and if two sequential samples are sufficiently close, and if the speed is sufficiently low, the car is assumed to be stopped. Due to the GNSS location having an accuracy of some meters, it is possible that the location samples will vary even though the device is actually not moving. The same is the case for the speed. Because of this uncertainty the user also has the option to manually let the app know that the car is stopped, after which the intelligent parking session initiation will run as normal.

The inaccuracy of the location samples is a big issue, as the user could end up paying for parking sessions that never took place. The location inaccuracy is therefore counteracted by using the Map Matching service. Map matching is estimating the true location of location samples with inaccuracies by including digital maps of roads where the locations are matched to. For more information about the map matching service see the description in Section 4.2. The estimated true location of the device that is returned from the Map Matching service is used to evaluate whether the vehicle is within a parking lot that requires payment, or if the vehicle is parked elsewhere.

The app implements a backup solution to the map matching, which simply checks the last of the non-corrected location samples for whether the vehicle is within the parking lot area. This is both in case the Map Matching service is unavailable, but also in case of the Map Matching service not having the maps of the current area available. The latter is possible because the
Map Matching service only is optional in order for the system to function, as opposed to the Parking Manager service which is mandatory as it contains parking availability and pricing information. If the car is estimated to be stopped within a parking lot where payment is required and supported by the Parking Session Manager, the parking session is started automatically. If this is done a notification is shown to the user. This notification has two sequential states. In the first state there is an initial grace period of a few minutes where the user can cancel the parking session without having to pay for anything. This is to ensure that the user has the final saying whether he want to start a parking session and ultimately pay for parking. In the second state the user has the option to stop the parking session, and at this point pay for the parking session. The notification is persistent such that the user is forced to interact with it to make it go away, i.e. cancel or stop the parking session. In the case where the notification for some reason disappears, for instance due to an app crash or other, the user has the possibility to access the Parking Session Manager via a web interface. The web interface requires login with user id and password, which is the same credentials used in the app. The web interface allows the user to see all his parking sessions, and to manually end active parking sessions.

4.2 Back-end Services and Parking Lot Information

Besides the application the core component of the IPA system is the Parking Manager service. This service has 2 main functionalities: providing information about parking lots, and managing parking sessions. The Parking Manager service is illustrated in Figure 5.

The information providing functionality of the service offers information about parking lots in terms of location, capacity, live availability, pricing, and other information. This information is used in the app to provide the user with information about parking lots in the area, and to help him select a parking lot to use. This is provided via a Representational State Transfer (RESTful) interface where the information is formatted as JavaScript Object Notation (JSON) for easy parsing and processing.

The parking session management functionality of the service provides an interface to start new parking sessions and to stop active parking sessions. This functionality is also accessible via a web interface, where users can log in and see all their parking sessions, and as a fail safe stop active parking sessions. When a parking session is stopped the related user is automatically billed for the parking, by the Parking Session Manager interacting with a billing service.

The Map Matching service is provided on a server as a web service. The service provides an interface to submit an array of GNSS location samples, including speed, accuracy, and orientation information. The map matching is done using an algorithm that is developed and described in [9]. The algorithm takes location samples as input, which are matched with digital maps of the corresponding area. The digital maps are stored at the server, and contains machine interpretable information about the road sections and intersections. Based on the
input locations and the attached information the map matching algorithm returns estimates of the actual locations, and a confidence in the updated estimates. The map matching step is not mandatory for the IPA to function, but it adds greater confidence to the decision of initiating parking sessions automatically.

Figure 5 Parking Manager service overview.

5 Use Case

There are two main use cases of the IPA system for the end user. The first use case is as follows: A user drives to work regularly in a big city, and each day when he approaches the city he opens the ParkAssist app on his smartphone and initiates the Automatic Park & Pay functionality. He knows where the parking lot is so he navigates to it, parks, and leaves the car with his smartphone. Once he stops his car the app transmits a number of GNSS locations to the Map Matching service. The location logging is initiated as soon as the car approaches the parking lot and stopped when the car is parked. The Map Matching service replies with the GNSS locations adjusted according to the digital road map of the area. This reply is used in the app to evaluate if the car is parked outside or inside the parking lot. If it is estimated that the car is parked inside the parking lot a parking session is initiated with the Parking Session Manager. When the user is ready to go to his car and leave he opens the app and selects his car, and is now presented with directions to where he parked his car. When he reaches his car he
stops the parking session, by interacting with the persistent notification on his smartphone, and the payment is automatically performed from his account to the parking company.

The other use case is more oriented at showing the migration of the system, and is as follows: The same user as described above, is going on a holiday to another city in another country. When he approaches the city, he opens the app on his smartphone and searches for the attraction that he is going to visit. The app now proposes the parking lot nearest to the attraction, and is presented with information related to the parking lot such as the number of available slots and the pricing scheme. The user selects the parking lot and receives turn-by-turn route guidance to the parking lot. When he arrives at the parking lot the same automatic sequence, as described in the first use case, is done of starting a parking session and paying when done. At the time when the user opened the app when he was initially approaching the city in the new country, the app contacted the MOBiNET platform, more specifically the Service Directory, and discovered the Parking Info Provider service relevant for the new city. When the car was stopped the Map Matching server was contacted to check if the new city is supported by digital road maps. If it is not supported the fall-back method is used where the most recent location sample is evaluated locally on the device along with the parking lot area.

From the two scenarios it can be seen how the process of finding a parking lot and paying for it can be done just as simple in a new town as in a town the user knows well. In both scenarios the user saves time spent driving around looking for parking lots, and thereby reduces congestion on the roads and reduces CO2 emission. The user also saves some time that is spent on paying for the parking session and he does not need to learn and understand a new payment methodology for each parking lot.

6 Evaluation of the System

6.1 Performance Indicators

The IPA system will be evaluated based on two types of parameters; user related, and accuracy of decision.

The user related parameters include how the user perceives the system and its functionalities, meaning to check if the system does what the users expect and does it properly, and to understand if the users think that the system actually makes the task of parking easier. Also statistics of usage of the system and its functionalities will be evaluated, to see if the functionalities of the system are actually used and how much.

The accuracy of decision parameters include how accurate the decision of initiating a parking payment is, both in terms of when the decision is made, if the decision is correct, and if no decision is made. Furthermore, as the decisions rely heavily on the GNSS location, and on having an accurate estimate of it, the location samples are also evaluated manually to check if they are correct and make sense, and to understand the general quality of the samples.
6.2 **Evaluation Scenario**

The IPA system will be evaluated based on a test performed in Aalborg in the northern part of Denmark, where approximately 40 users will be part of a test, where they will be using the application when searching for and parking on parking lots, and when paying for the parking sessions. In Aalborg 11 parking lots with time-based parking payment will be a part of the test and the cars of the users will be equipped with a sticker in the windshield that parking attendants can scan to check if the car in question currently has an active parking session, or if the car should be fined. This evaluation scenario is what is described in the first use case described in Section 5. Later on it is planned that additional cities across Europe will implement a Parking Manager service, effectively realizing the second use case described in Section 5.

7 **Next Steps**

Currently, besides the live test with real users, the IPA system is in the process of being migrated to Trikala Greece. To make such a migration possible a local parking information service is made available. This is the only mandatory step for a migration. For the rest of the IPA system the existing components can be used for the Greek users without modification. However, additionally both the app and the web interface are translated to Greek for convenience. The reason for so few steps being required to perform a migration of the system to a city in another county is the MOBiNET platform. Key functionalities offered by MOBiNET in this context are identity management, billing management, and discovery of services on the go based on location and output data format. These functionalities allow for easy reusability of software system components in other contexts.

Another aspect, however much more technical, is evaluating the impact of poor network conditions on the performance of the system. Particularly the performance of the network while the map matching is being performed is of interest. The map matching process consists of collecting location samples, transmitting the samples from device to server, running the map matching algorithm, transmitting the matched location samples from server to device, and evaluating the matched locations on the device. If the network performance is poor during the transmissions of the samples back and forth, the user might have moved before the reply arrives at the device. This could lead to a wrong decision based on the reply, or that the reply is never received, why no decision can be made. In either case it would be preferable to use the backup solution and evaluate locally if the device is within the parking lot or not, despite the lower accuracy of the data to base the decision on. For this reason, the performance of the connection is an important factor in performance of the system.

**References**
Intelligent Parking Assistant - A Showcase of the MOBiNET Platform Functionalities


