

CORSA: An Open Solution for Social Oriented Real-time Ride Sharing

Simone Bonarrigo, Vincenza Carchiolo, Alessandro Longheu, Mark Phillips Loria, Michele Malgeri and Giuseppe Mangioni

Dipartimento di Ingegneria Elettrica, Elettronica e Informatica Universita` degli Studi di Catania - Italy

Abstract. The combination of the interest in environmental questions on one hand and the massive use of web based social networks on the other recently led to a revival of carpooling. In particular, the exploitation of social networks promotes the information spreading for an effective service (e.g. reducing the lack of confidence among users) and endorses carpooling companies via viral marketing, finally acting as a basis for trust based users recommendation system. In this work we outline CORSA, an open source solution for a real time ride sharing (RTRS) carpooling service that endorses the role of social networks by using them as a conveying scenario for the virtual credits reward mechanism CORSA is based on.

1 Introduction

Environmental related factors as the reduction in both the vehicles number and the expenses for gas as well as the energy consumption and pollution recently endorsed a renewed interest in carpooling [1, 2]

Moreover, the increasing adoption of web based social networks [3] offers (1) a solution to the lack of confidence among users, thanks to the spreading of information, (2) a rating – and recommendation – based approach both for users and services, thus e.g. trust networks [4, 5] can be used as a tool for a robust users management and (3) the possibility of discovering and leveraging concealed relationships, e.g. if a community of person that lives/work at close places is detected [6], the system can put them into contact.

Carpooling Open source Ride Sharing Application (or CORSA) is the carpooling solution here described. Our goal was to build an application accessible anytime from anywhere with high usability, effectiveness and efficiency in automatically providing users with helpful solutions.

In CORSA the role of social networks is also emphasized, indeed users are rewarded with virtual credits when using the service and we push them to spread the adoption of the service via existing social networks they belong to according to the viral marketing paradigm [7], therefore increasing their personal credit.

To exploit the use of social networks from the very beginning, the initial set of users involved in the testing phase consists of students attending the same University, so a (possibly strong connected) virtual social network is already present. Finally, the work was developed using only open sources libraries, to spread its adoption.

In section 2 the main issues we tackled in developing our proposal are discussed, whereas in section 3 we

outline the architecture of CORSA, finally providing concluding remarks and future works in section 4.

2 Challenges of a RTRS platform

2.1 Main Issues

The terms “*real time*” and “*micro carpooling*” correspond to a set of challenges that most carpooling platforms don’t face.

The fact that a micro carpooling system targets short-range trips means that the definition of pickup points is a crucial aspect. Whereas long distance carpooling can easily make use of points of interest such as bus stations, airports, etc. micro carpooling requires finding a pickup point in the proximity of starting locations of driver and rider.

The real time factor reinforces this problem since both users must reach pick up points quickly. Their automatic selection requires considerable work to determine in advance which pickup points would be convenient. This would limit the spread of the app as it would only be usable in towns that the system targets. A workaround is to let users freely decide pickup points by chatting before the ride.

From a technical viewpoint, the challenges that a real time ride-sharing platform implies derive mainly from the constraints on usability and response time of the system. We can describe these constraints with three features we aim at implementing in our proposal: automatic selection of compatible rides, fast response and ease of use.

Considering the first one of them, the usage scenario of a real time ride-sharing app implies that the system must make all the possible choices for the end user. Many

^a Corresponding author: Michele.malgeri@dieei.unict.it

carpooling websites and services closely resemble a bulletin board where the user must find a compatible ride.

The short span of useful time and the limited display screen on mobile devices (reasonably, the most adopted in such context) create the need for an intelligent behind the scenes selection of compatible rides to display. Ideally, the system must be able to show a limited selection of compatible rides, as an excess of options would consume useful time.

The question of a fast response (second feature) lies in the nature of the use cases of such system. Indeed, short-range travellers that make use of public transportation might want the ability to make choices very quickly in order to make decisions regarding the rest of their journey.

A system that offers such slow performance would be useless to such users, representing a waste of time and not allowing the user to effectively make use of the system by integrating it with the already present transportation network. Operational speed and quick response time translate in optimization of ride lookup and reducing network data transfer to the bare minimum.

Finally, considering the ease of use, the expected context of use is on the go, the app we intend to use for a real time ride-sharing system must be usable in the simplest of ways. This aspect has more to do with the usability of the user interface but also with how well the use cases represent the effective desired usage.

2.2 An ideal real time carpooling platform

The majority of carpooling platforms often resemble a bulletin board with a number of proposed journeys the user can choose from, usually presented according to the compatibility of start and finish point only.

The platforms that allows searching intermediate journeys do allow this feature by letting the driver insert during the creation of the route in between stops that he plans to do. This time consuming operation can be easily automated by a system that considers compatibility between rides based on the actual scheduled route.

This solution however forces the developer to face a new set of challenges, mainly related to performance, as the lookup is a more resource intensive operation. In order to ensure extremely fast response time vertical and horizontal scalability might not always represent the ideal solution. We further discuss lookup optimization and the problems we have encountered in the next paragraph.

The innovation of our carpooling idea is to use virtual credits as the sole source of reimbursement for the service. The purpose of carpooling is to share a resource in order to obtain various benefits for the environment, social interaction and urban mobility. Our aim is not to create a moneymaking platform for the driver, by creating an alternative to taxis or Transportation Network Companies.

In short distance micro-carpooling, the payment procedure must be addressed differently than a long distance carpooling. The short distances involved indeed would lead to ridiculously low reimbursements that

cannot guarantee the coverage of costs implied by the extra mileage.

The usage of virtual credits seems to be the best for several reasons:

- In a micro carpooling context, the resulting compensation is minimal. The driver might not be inclined to share their time and resources for a low fee. The idea of rewarding the driver with credits reusable in different scenarios might be an incentive for the driver to be an active member of the community.
- By using virtual credits, institutions and retail companies would be directly involved in creating campaigns that would generate economic, cultural and social benefits.
- By using virtual credits it is easy to implement fun and engaging recreational initiatives, incentivizing users to be active car-poolers. Integration with social networks is a key feature of the platform.

The first benefit is the increase in mutual trust between users. Secondly, by creating a community around the platform users can create events, which involve other people, even strangers, but with whom they share the same interests. As an example, a user may share with the community the intention of reaching the city centre for a concert. By sharing rides, users not only save money, but also enlarge their social network by reaching people they shares interests with and spend quality time with. Tight integration with social networks allows the carpooling platform to add social interactions to the list of environmental and financial benefits. In addition, government institutions and retail companies can benefit from the use of our system and social networks. By promoting events and activities, they can reward users to reach their facilities by ridesharing. Institutions and companies would benefit from users actively promoting them through the social network. Users would benefit by gaining extra virtual credits that he/she can spend within the market.

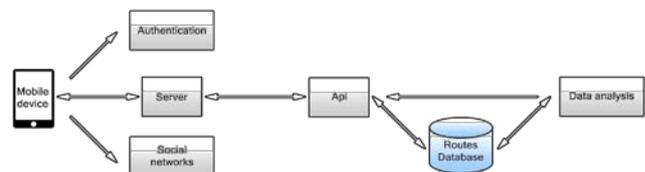


Figure 1. Application Architecture-Components of the CORSA platform

The implementation of such system is very simple. By accessing a dedicated interface, institutions and retail companies can create promotional campaigns with associated locations. These locations, called hot spots, can be markers or polygons that represent a place of interest. When users check-in or checkout their position is controlled by the server. If there is an associated

promotional campaign to their location, they receive the reward.

The most interesting aspect of the carpooling is that it allows improving transportation and environmental sustainability by reducing traffic. Micro-carpooling is especially interesting since cities are highly polluted areas, due to the high traffic density.

CORSA is a valuable tool to address such problems. By creating campaigns and associated hotspots it allows institutions to influence urban mobility in a way that is engaging and fun for the end user. Secondly, by analysing long term data, government institutions could study accurate commuter flows allowing them to create effective campaigns or redefine parts of the public transport network.

3 CORSA Architecture

The real time factor points towards the direction of a mobile solution as a main form of interaction with the system.

A key step in mobile development is deciding whether to opt for a native implementation or a hybrid solution. Both solutions have strengths and weaknesses with the main trade off being between performance and platform coverage. Both are key aspects for our platform, but a lot of the performance concerns are tied to network communication and server interaction. Going native would only speed up part of the process. On the other hand, fast prototyping and spread of the applications are key aspects we did not want to compromise on.

The system is composed of three main components (see fig 1): a mobile frontend, a server backend with real time bidirectional communication capabilities and a path management system accessed through an API.

The app and the real time server communicate through web sockets, while uses http protocol to calls the path management API. In both cases, data is transferred using JSON format.

A common problem in developing mobile apps that require heavy interaction with a server is to establish whether first developing the app and then the server or conversely developing both of them side by side. What emerged quickly was that usability was so important that use cases would need to be tested for usability constraints on a device or emulator. Graphical mockups did not aid the purpose, since the lack of navigation structures in mobile apps is substituted by gestures. Because of all of this, we decided to use a web application as a mockup.

To compensate the absence of the API we created static JSON files to inject data into the app. At the end a testing stage where we evaluated usability of the UI and use cases, the static JSON files actually described what data the path management API had to provide and its structure. With a well-defined structure, we were able to concentrate on algorithms and query optimization during the API implementation stage.

All details about (1) the mobile application that has been developed, (2) the management of paths and (3) server and application data can be found in [8].

4 Final remarks

In this paper we introduced CORSA, an open source solution for a real time ride sharing (RTRS) with high accessibility, high usability, and effectiveness and efficiency in finding a riding solution for users. The challenges a RTRS platform should cope with was also discussed, from both a functional and a technical point of view; the overall architecture was also outlined.

Future issues to be investigated are (1) how the use of social network can improve the proposed carpooling service (2) to discover and exploit hidden relationships, for instance communities among users sharing trips and (3) to gather data on users and rides for further analysis.

Acknowledgements

This work was developed under the projecty "S.R.S. - Progetto di formazione integrato SINERGREEN (Smart Intelligent Energy Green), RES-NOVAE, SEM" supported by MIUR (Minister of Education, University and Research).

References

1. Fagin, R. & Williams, J.H., A fair carpool scheduling algorithm. *IBM J Res Dev*, 27(2), pp. 133–139, 1983.
2. Teal, R.F., Carpooling: Who, how and why. *Transportation Research Part A: General*, 21(3), pp. 203 – 214, 1987.
3. Golbeck, J., The dynamics of web-based social networks: Membership, relationships, and change. *First Monday*, 12(11), 2007.
4. Carchiolo, V., Longheu, A., Malgeri, M. & Mangioni, G., Trust assessment: A personalized, distributed, and secure approach. *Concurr Comput : Pract Exper*, 24(6), pp. 605–617, 2012.
5. Carchiolo, V., Longheu, A., Malgeri, M. & Mangioni, G., Users attachment in trust networks: Reputation vs. effort. *IJBIC*, 5(4), pp. 199–209, 2013.
6. Carchiolo, V., Longheu, A., Malgeri, M. & Mangioni, G., Search for overlapped communities by parallel genetic algorithms. *IJCSIS*, 6, no.2, 2009.
7. Leskovec, J., Adamic, L.A. & Huberman, B.A., The dynamics of viral mar-keting. *ACM Trans Web*, 1(1), 2007.
8. [Loria, M. & Bonarrigo, S., Rtrs: a social oriented proposal. Technical report, DIEEI - Universita' di Catania, 2014.