FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Motivational and Trust Mechanisms to Leverage Shared Mobility

Bruno Miguel Costa Barros

Mestrado Integrado em Engenharia Informática e Computação

Supervisor: Rosaldo J. F. Rossetti, PhD

July 25, 2019

Motivational and Trust Mechanisms to Leverage Shared Mobility

Bruno Miguel Costa Barros

Mestrado Integrado em Engenharia Informática e Computação

Approved in oral examination by the committee:

Chair: Ana Paula Rocha, PhD
External Examiner: João Emílio Almeida, PhD
Supervisor: Rosaldo J. F. Rossetti, PhD

July 25, 2019

Abstract

Nowadays, the concept of a sharing economy is a key aspect for the sustainability of a city. Smart Cities are becoming more and more a reality today and sustainability becomes a real concern. Society has access to its own vehicles which contribute in a less positive way to the environment and to the economy, undermining the current and the future generations, compromising their sustainability.

A solution to this problem seems to be very simple, for example, more public transport and less private vehicles. But it is not always possible to have access to public transport to all of the user's destinations, and at the times the user intends to travel.

The solution found by many for this problem focuses on shared mobility. The well-known Carpooling and Carsharing concepts consist in sharing or renting the user's own vehicle in order to decrease the number of cars on the streets, through favouring a higher occupancy and avoiding vehicles to circulate with a single passenger: the driver. This solution has everything to be successful, but also has disadvantages.

The fact that we live in a world of technology, it is possible to know that the interest of the users in a new application, in a new piece of technology is volatile and can be lost very quickly if the user does not feel motivated to keep on using it. Another disadvantage of shared mobility is the lack of trust in other persons with little acquaintance while one opts for such platforms. People very unlikely will have enough trust in other users they barely know, compromising the efficiency of such systems at least as a daily commuting solution.

The main objective of this dissertation is to devise a solution capable of surpassing the three barriers defined before, namely mobility, trust, and motivation, in order to keep users contributing to the sustainability of generations.

With all the analyzed state of the art, it was possible to come up with a solution for each of the aforementioned problems. The problem of mobility can be tackled by the use of shared mobility, the problem of trust can be tackled by the use of a social network mechanism, and finally the motivational issues can be addressed by solutions implementing gamification techniques. The final product of this dissertation ended up being a mobile application leveraged on a social network for ridesharing that uses gamification techniques, making users more aware of how their mobility decisions might impact the environment.

The resulting app was tested in terms of the implemented features in a closed community, by some volunteer Users. Users needed to follow a script of tasks to perform in the application and then answer a questionnaire. The answers to the questionnaire were very positive, with users having liked the application and seeing this as a promising solution to overcome the mobility concerns presented.

Key Words: Smart City, Sustainability, Gamification, Shared Mobility, Shared Economy, Carsharing, Closed Communities, Social Network, Trust

Resumo

Atualmente, o conceito de economia partilhada é um aspecto fundamental para a sustentabilidade de uma cidade. As cidades inteligentes estão a tornar-se cada vez mais uma realidade e a sustentabilidade torna-se uma preocupação real. A sociedade tem acesso a veículos próprios que contribuem de forma menos positiva para o meio ambiente e para a economia, prejudicando as gerações atuais e futuras, comprometendo a sua sustentabilidade.

Uma solução para este problema parece ser muito simples, por exemplo, mais transportes públicos e menos veículos particulares. Mas nem sempre é possível ter acesso a transportes públicos para todos os destinos do utilizador e nos horários em que este pretende viajar.

A solução encontrada por muitos para este problema concentra-se na mobilidade partilhada. Os conhecidos conceitos Carpooling e Carsharing, que consistem em dividir ou alugar o próprio veículo do utilizador para diminuir o número de carros em uso, favorecendo uma maior ocupação e evitando que veículos circulem com um único passageiro: o motorista. Esta solução tem tudo para ser bem sucedida, mas também tem desvantagens.

O facto de que vivemos num mundo de tecnologia, é possível saber que o interesse dos utilizadores numa nova aplicação, numa nova peça de tecnologia é volátil e pode ser perdido muito rapidamente se o utilizador não se sentir motivado para manter o uso. Outra desvantagem da mobilidade partilhada é a falta de confiança em noutras pessoas com pouco conhecimento enquanto se opta por tais plataformas. As pessoas muito provavelmente não terão confiança suficiente nos outros utilizadores que mal conhecem, comprometendo a eficiência de tais sistemas pelo menos como uma solução de comunicação diária.

O principal objectivo desta dissertação é conceber uma solução capaz de superar as três barreiras definidas antes, ou seja, mobilidade, confiança e motivação, a fim de manter os utilizadores contribuindo para a sustentabilidade de todas as gerações.

Com todo o estado da arte analisado, foi possível chegar a uma solução para cada um dos problemas acima mencionados. O problema da mobilidade pode ser resolvido pelo uso da mobilidade partilhada, o problema da confiança pode ser enfrentado através do uso de um mecanismo de rede social e, finalmente, a questão motivacional pode ser abordada por soluções que implementam técnicas de gamificação. Esta dissertação acabou por ter como produto final uma aplicação móvel que se baseia numa rede social para partilha de viagens que usa técnicas de gamificação, tornando os utilizadores mais conscientes de como suas decisões de mobilidade podem ter impacto no meio ambiente.

A aplicação resultante foi testada em termos dos recursos implementados numa comunidade fechada, por alguns utilizadores voluntários. Os utilizadores precisavam de seguir uma lista de tarefas para executar na aplicação e depois responder a um questionário. As respostas ao questionário foram muito positivas, com os utilizadores a gostar da aplicação e vendo esta como uma solução promissora para superar as preocupações de mobilidade apresentadas.

Palavras-chave: Cidade Inteligente, Sustentabilidade, Gamificação, Mobilidade Partilhada, Economia Partilhada, Partilha de Carros, Comunidades Fechadas, Rede Social, Confiança

Acknowledgements

I would like to thank my supervisor Professor Rosaldo Rossetti for all the support during the realization of this dissertation.

I also Would like to thank all my family to the love and support they gave me during all these years, not only in the process of the execution of this dissertation but also my entire life. Without their support, I would not be capable to achieve what I've had already achieved. A really big "thank you" to my girlfriend, that is my best friend since we were together.

Finally a big word of appreciation to my friends, especially *random.org #PT Long Live Tugalândia E Os Preços Baixos, it would not be the same without them.

Bruno Miguel Costa Barros

"When something is important enough,
you do it even if the odds are not in your favor."

Elon Musk

Contents

List of Figures

LIST OF FIGURES

List of Tables

Abbreviations

FWC First World Countries

Chapter 1

Introduction

1.1 Scope

During all these years the population of first world countries has been thinking in ways to keep improving the sustainability of future generations. That sustainability is a pivotal point in the creation and evolution of smart cities. One of the major improvements in the sustainability of the current and future generations is the way they improve the environmental conditions to keep the city as smart and green as possible so that people can perceive quality of life.

One of the main obstacles is the mobility of the population. The excessive use of private means of transport in these areas has a big and very negative impact upon the environment, jeopardizing sustainability. One of the major steps, in order to combat this trend, is the use of public transportation but that is not always the best mean of for different reasons. The solution that have proven more consistent over the years generally rely on the so-called shared mobility.

Shared mobility consists of sharing private means of transportation, which allows users to access other types of mobility services. There are some types of shared mobility services, such as Carsharing, Bikesharing, and Ridesharing. This not only provides a good impact upon the environment and consequently leveraging sustainability in general, but it also provides users with a way so that they can waste less money on their trips since they are sharing a vehicle or making money just to "rent" their own vehicle. This kind of behavior falls withing a type of economy, namely the sharing economy.

The sharing economy is an economic model that presents new consumption habits and new business models, where the assets that are usually little used are profitable therefore opening up new opportunities of market models to be explored. In a shared economy, services and labor by people to people are allowed, usually by using options provided by online platforms, which can be more easily accepted and paid with the benefit of reduced transaction costs.

1.2 Problem Statement and Motivation

The use of private transportation, for example commuting to work, can be quite disadvantageous in both monetary and environmental terms since most of the people that go to work on their own cars usually travel alone. The use of public transportation can be a solution but both the schedules and destinations may not always coincide with those of the user.

A solution that is more frequent nowadays is the use of shared mobility, where persons can share trips with other persons in order to prevent the use of an unnecessary amount of vehicles circulating on the streets. An already known disadvantage of this solution is the lack of trust of some persons in the use of such platforms, basically because people very unlikely have enough trust in users of a system they do not know well.

The solution developed for these problems aims to be inserted in the context of a Conference, whose participants form up a very well-defined community, generally attending the event in consecutive editions for long times. The idea is thus to profit from a community of people that are familiar with each other, minimizing the impact of issues concerning trustworthiness and maximizing the potential of attendees gathering to share trips.

1.3 Aim and Goals

In order to improve the referred problems, the aim of this project is to build a system that offers solutions for those big three problems, namely the motivation, trust, and mobility. In order to perform a solution for these three main problems, a study in the areas of the shared mobility, closed communities, and social networks, as well as in the area of gamification will be performed.

The main goal is to design a platform for travel sharing solutions using a social network system in order to improve the trust factor and using gamification techniques as a motivational factor.

1.4 Document Outline

Besides the introduction, this dissertation contains five more chapters. The Literature Review is described in chapter 2 in order to position this work amongst other similar efforts and to understand what has already been done in this direction. Chapter 3 presents the methodological approach used for building the platform, a high-end approach to the methods used to implement it, and an strategy for the evaluation of the resulting solution. Then, in Chapter 4 the proposed architecture for the application is explained in detail including all the layers that compose the its underlying structure. A thorough evaluation of the system with a detailed discussion of the main results are presented in Chapter 5. Finally, Chapter 6 concludes this report, emphasizing on the main contributions of this dissertation, and presenting prospects for further development and future research directions.

Chapter 2

Literature Review

In this Chapter the actual state of the art is described, all the articles and experiments that were formulated around the topics covered, topics that were taken into account for the realization of this dissertation. Here are defined the main topics of study in order to formulate the bigger problem.

The first section is addressed to Shared Mobility. Firstly, it is introduced the definition of Shared Mobility, meaning and nowadays' position. Then, it is approached the theme of the Sharing Economy and the impact that shared mobility has on this type of economy. Next, are specified the two most relevant techniques of shared mobility for this dissertation, the Carsharing, and the Ridesharing.

The second section is responsible for the definition of trust, especially in the context of closed and open communities, in the reality of the internet in social networks.

In the third section, it is approached the Gamification theme. Firstly, it is introduced the definition of Gamification. Then it is specified the types of Gamification and how it can be divided into different categories. Finally, it is approached the motivational factor of these techniques.

After all the definition and contextualization in the areas of urban mobility, more specifically in the area of the shared mobility and the gamification, it is approached the conjugation of both terms and is evaluated all the state of the art in the Gamification of Urban Mobility. First the Gamification in the Carsharing and Ridesharing area and then in the Parking area.

After all the definition and examples, there is a space for exposition of all the existing Frameworks and Models that are designated to these areas.

Finally a quick summary of all the terms that were addressed.

2.1 Shared Mobility

Shared Mobility consists of the share of any mean of transportation with other users on a need basis [SCBC15]. The sharing strategy, that has been growing in Urban areas [MA15], allows all the users to have access to a totally different range of possible transportation services. Shared

Mobility represents a variety of alternative transportation modes and the most relevant as said in [CK14] are Carsharing, Bikesharing, and Ridesharing. Possible schemes of shared mobility, for instance, are reported in [SKC+16]. Also, ways of analyzing such schemes are suggested elsewhere [CSR+18, SRKP16].

A great part of the evolution of technology is the formation and support of Smart Cities. A Smart City can be described as "A city well performing in a forward-looking way in economy, people, governance, mobility, environment, and living, built on the smart combination of endowments and activities of self-decisive, independent and aware citizens" by [HO07]. One of the most important aspects for the good functioning of a Smart City is the **Sustainability** of all the generations and nowadays the urban mobility system is experiencing issues in terms of long-term Sustainability in environmental, social and economic areas [ACZ+13].

One of the most important factors of that sustainability is the environment. Among the recent years the growth of population in the urban areas have highly increased and the prospection is to reach even higher values [CDBN11]. That growth of the population has a big impact in their mobility, this problem comes mostly from most of the people having their own car, which increases the number of vehicles that circulate on the roads. That excessive number of vehicles produce a higher quantity of heavy metals [MDARM+15] and greenhouse gas [Zyg13] emissions to the atmosphere and that behavior is a bad indicator of the sustainability of the generations. It is central to achieve the Sustainability of the present generation and, more importantly, ensure the Sustainability of future generations.

Shared Mobility affects other areas as the Economy and the Environment. The use of Shared Mobility opens the possibility for users to invest in the Shared Economy while improving environmental sustainability.

2.1.1 Shared Economy

Shared Economy is defined, by [HSU16], as a peer-to-peer type Economy where the users change services, products, etc. and the transactions are usually via community-based online services. The use of Shared Economy is directly involved with the growth of Smart cities [GPS15] as is more often seen the exchange of services or products from one user to another. All of the communication and the transactions are performed with a web basis.

Shared Mobility has a big importance in the Shared Economy. The use of technologically based transactions as a mean of payment for Shared Mobility improves the Shared Economy.

2.1.2 CarSharing

The technological evolution of the world with the creation of Smart Cities gave the possibility of the users to be more aware of the environment that surrounds them, the needs and all the offers provided by all the others. The technology made a huge impact on how the user can make a profit with his vehicle that he is currently not using.

Carsharing is a model of vehicle sharing, and as said in [KQWW13], is a car rental, attractive to customers who make only occasional use of a car on demand. A car can be sent on an as-needed basis by the owner to the requester of the vehicle. This is a model in growth since the initial of the 2000s and already having a good acceptance community [Kat03].

There are two possibilities for trips, the round trip where the user that picks the car needs to get the car back to the same place he picked up and the One-way trip where the user can pick the car in one place and let the car in another place previously defined.

As said before this is one of the most used way of Shared Mobility, having an already having a good acceptance community, having multiple examples of apps like Whim[1], Migo[2], Yego[3], DriveNow[4], Emov[5] and others.

2.1.3 RideSharing

As was mentioned before, the technology helped in the development of all the economy necessary for the transactions of services between users in today's Smart Cities. The solutions offered by some public means of transportation are not always that much practical. Some users have the private mean of transportation and usually, do not travel with the car full. The idea of sharing a ride between two persons with identical origin and destinations can be very useful and profitable as some articles have proved [SRS+14].

Ridesharing such as Carsharing is a model of vehicle sharing, instead of renting a vehicle for another user riding the car, is a mode of transportation in which individual travelers share a vehicle for a trip and split travel costs such as gas, toll, and parking fees with others that have similar itineraries and time schedules [FDO+13]. Ridesharing became really popular when companies started encouraging simple users to give a ride to another user in exchange for monetary retrieval. One of the main examples is Uber[6], with this company being present not only in America but also in many European countries.

Some other examples of applications that follows this pattern of shared mobility are Whim[7], BlaBlaCar[8], Caronetas[9] and others.

2.2 Trust

Along with the modernization of the World, we can see a visual growth in Technology, from the evolution of systems, especially in the industrial societies after their maturation, had in technology one of the "prime movers and determinants of their general development" [SM92]. This growth

[1] https://whimapp.com/
[2] https://www.getmigo.com/
[3] https://www.getyugo.com/
[4] https://www.drive-now.com/
[5] https://www.emov.eco/
[6] https://uberportugal.pt/
[7] https://whimapp.com/
[8] https://www.blablacar.pt/
[9] http://www.caronetas.com.br/

in technology means a lot of people connected to the internet doing different types of things, but especially connecting with each other. The use of social Networks has increased in the last decade and with that millions of users using social networks like Facebook all day and a lot of time, it became a usual practice to verify their social network feed multiple times a day [BML10].

With the increased usage of Social Networks, people started to create their circle of contacts, a network created by Users, where they recreate a 'neighborhood' in their connections in order to make the relation between each other closer and increasing the trust between them [Sco88]. These connections are a big factor in the relation between users. Connections in social networks not always represent a real-world connection. Users that do not know each other in real life might end up connecting to each other for multiple reasons. Most Social Networks have matching algorithms in order to suggest connections to the user. Those algorithms find characteristics in users that might cause interest in others, a similar interest in some area or subject [TLT+11]. Profile Similarities can be a major boost in relationship strength. These connections can be categorized with a level of relationship strength, which can be defined as "a hidden effect of nodal profile similarities" [XNR10].

That said, social networks are open communities that facilitate communication with people that we don't know yet, and "the stability of a community depends on the right balance of trust and distrust" [ARH00]. That stability can only be achieved with a strong relationship between the connections and the user and a big trust factor of the User in the community.

The Trust factor is a big concern, and in Smart Cities, this aspect is taken even more into account. The creation of a social network helps the fact that a person can build his own network of contacts possibly using matching mechanisms and improving the probability of establishing a connection [EW09].

2.3 Gamification

Technology influenced all of Humans lives until the actual days. The evolution made people more interested and, above all, more dependent on technology. But all this interest is very volatile and can be lost very easily if the user loses interest. Gamification appears as a solution for these sorts of problems.

Gamification, as said in the following articles [GVRM+17], [NTS+15], [DDKN11], was once defined as "the use of game design elements in non-game contexts". These techniques are used to stimulate specific usage patterns by users, merging gaming mechanisms and game concepts with the system in question, even if the system was not originally designed with playful intentions. One of the most important elements of games that make people enjoying it is making small steps visible and consequently this approach make the development look greater [KS15]. Whereas digital games have already been suggested and used in the domain of transportation [RAKG13], the use of gamification in this domain is still to be explored to its full potential.

As it was possible to see, techniques of gamification are made to interact with the user, being gamification responsible for intervening in three major areas of human perception [LH11].

- **Cognitive** — Gamification can provide a more complex system of rules in which players have to explore through active experimentation and discovery;
- **Emotional** — In the realization of the Game the user can feel emotions such as curiosity, joy or frustration;
- **Social** — Games provide to the user the possibility to experience new roles and new identities and supplies to them the possibility to interact with other users;

Consequently, is possible to build a gamification technique surrounding a system and make a user involved in a game without noticing and, more important, making the users addicted to that game.

2.3.1 Types of Gamification Techniques

There are some different ways of Gamifying a system. The way of how the techniques reward the users for their interaction with the system can be different, as [RBK+17] says that game-based interventions should attribute relatively small rewards for the users' behaviors.

There were defined some Gamification techniques over the years, as it can be found in [MHWA17] and also in [ZC11]. They are usually divided into two categories, Simple and Advanced.

2.3.1.1 Simple

In the Gamification area, the simple techniques are those that do not require much thinking by the user, are usually more intuitive. These techniques are the most simple and easy to understand but also could be less effective.

Techniques as a **Point System** are an easy way to gamify a System, for each input in the System the user is rewarded with a number of points and different type of tasks lead to a different number of points.

Another simple technique of Gamification is a **Badges System**, where the user can achieve different badges from different tasks. This technique can be allied with the first one, when a user has a specific number of points gets a different badge.

The last example of a simple technique of Gamification is the **Leaderboards**. This is the most versatile technique presented in the simple category because it can be allied with both of last ones. This technique has the benefits of not being too melancholic and enables a more direct competition with other users.

2.3.1.2 Advanced

In the Advanced area of Gamification, it is possible to see more complex techniques, a big part of them not being only user dependent. These techniques are more complex and can be much more effective than simple ones.

The **Level System** is considered an advanced technique of Gamification, it involves a much more prepared side of the system and constant update, either by the creation of new levels or by the factor of innovation.

Another technique is the **Challenges/Missions** technique. This technique offers the user different types of challenges or missions for him to complete in order to be rewarded. This technique involves an innovative and constant update of Challenges/Missions, making each one different from the previous ones, making the user addicted by not experiencing the same behavior.

A technique that is also very effective is the **Feedback** technique. This technique is largely used in gamified applications and systems nowadays and consists in the other user giving feedback to the first one by a service that the user has provided.

2.3.2 Motivation

The Gamification techniques are beginning to be used in the most diverse systems of today, making the gamification a must use framework. This phenomenon of growth happens because these techniques have a motivational factor between the users. Those techniques leverages fun, competition, rewards and game mechanics in different contexts. As elements of improvement of the motivation of the user, there are some principles which according to [FMLR16] must be followed.

- **Relatedness** — The interaction and connection to other Users;
- **Competence** — The effectiveness in the resolution of a problem in a given environment;
- **Autonomy** — The capacity of the user in controlling his own life;

This can modify the users' approach in respect to the environment or the economy making the User more self-aware of his behavior.

2.4 Gamification in Urban Mobility

Urban Mobility is one of the areas that have improve the most with the advance in Smart Cities and Technology. There are a lot of technology investments in the urban mobility area, a lot of applications, a lot of smart vehicles, in general, a big improvement.

There are a variety of applications introduced in the theme of Smart Cities, applications for everything that can be done in a vehicle. Applications like GPS, applications for public transportation, application for specific transportation in the case of Carsharing and Ridesharing talked above. The majority of these applications have a technique that helps to improve the app, helps to improve the user interaction with the app and sometimes helps to improve the user behavior with the environment, this technique called Gamification. Instill good behaviors in the society through motivational factors will improve Sustainability, as said in [SB15], "persuading citizens to make more sustainable transport choices could help address the negative impact cars have on cities."

As an example of a Gamification application in the urban area, the Waze app [10]. Waze is a "social community-based traffic and navigation application", a type of GPS that is content-updated by its community. "The application informs the users about the traffic, construction, accidents, the police, helps to find the cheapest gas along the route, and, in case of a traffic jam, it helps to change the route to find the most efficient route". These are some of the main functionalities of this application, but this application has a gamification factor, encouraging users to help build its crowdsourced maps and offering points, for example, for driving around with Waze and validating the application's directions. At regular intervals, the application provides to the user different games, in which the drivers collect items while driving which provides the user the possibility of getting extra points, by the resolution of map errors or by notifying the authors about road problems and others.

Another example of a Mobility application in urban mobility is the Metropia app [11]. Metropia is, as Waze, an application for the Urban Mobility, Metropia is an "advanced platform that allows travelers to optimize their travel, Metropia uses real-time data, and predictive algorithms to predict traffic and provide the best route to a certain location". It is used "behavioral economics strategies and gamification elements to reward drivers to shift their travel to off-peak times, to take under-utilized streets, and opt for alternative means of transportation". This culminates in points earned to unlock some discounts in local stores, restaurants, and some other places.

2.4.1 Gamification in Carsharing and Ridesharing

Carsharing and Ridesharing are one big part of this dissertation and these two categories are one of the main focus of Urban Mobility.

The [OPT18] article observed and made a study in the urban mobility area, especially in the shared mobility area with a more deep study in the area of carpooling and how a gamified app in the city of Warsaw could impact the drivers and surroundings in the environment and the other people in a Simulated ambient.

To start was made a study in how was the interest of carpooling in the city of Warsaw. They end up with 337 respondents, in which only 45 of them were not interested in carpooling, neither in being a passenger or a driver. The 292 other respondents were mildly or high interested in carpooling being drivers or passengers.

After some surveys surrounding the questions and what parameters most influence the Carpooling mechanism so that could be gamified. They end up with many answers and after analyzing all the answers they came up with a solution on what type of user characteristics should be verified by the mechanism for a complete carpooling gamified application.

The parameters that ended up as particularly essential for the drivers and passengers in a Carpool System are:

- **Impatience**

[10] https://www.waze.com
[11] https://www.metropia.com

The impatience is an important factor for the persons that use the carpool system since an impatient person is not so good as a passenger that is as a driver.

- **Kindness**

  The kindness is a crucial factor in the driver and the passengers of a carpool trip for the trip to happen at his best.

- **Allowable duration of the auction**

  The allowable duration of the auction is important to make the best negotiation for the whole participants.

- **Dependency of price on the number of passengers**

  The dependency of price on the number of passengers helps the negotiation in making the best prices for the trips with more passengers, benefiting them.

After the research, the authors formulated a prototype of simulation of an urban game that simulated the interest in carpooling. Game participants in the simulation whom the authors named "orcs" compete for “transport routes” by forging “alliances” through carpooling. The drivers and passengers with all the trips that they are involved in collect “trophies”, namely points that place the user in the hierarchy, a leaderboard. Drivers and passengers compete for different leaderboards with a point attribution different for the two cases, but there is also a competition for those "alliances" mentioned above and is a competition between “clans” (companies and institutions), which results in winning a weekly virtual title.

For the simulation to be the most precise and realistic possible the authors needed to predict the behavior of the users of the carpool system and make that representative in the agent system. For that, the authors implemented game mechanisms for the agents to decide and execute actions to manage the carpool activity. The game mechanisms that were used will be described below.

Competing for limited goods, that is representative of the space in a car, with time and space limits as this mechanism, is responsible for the restriction in the time and space available for the trip. **Cooperation**, that is representative of the creation of collectives who plan they journey together, this parameter will restrict the trip for a determined group of agents that have a coincident path. Finally, the **Immediate Feedback** on the offer, representative of the result is the acceptance or the rejection of the offer, where the agents will be capable to represent the interest on the trip that was proposed. The game has defined rules and those can convert parameters of agents such as the price of the offer, kindness, capacity, region into results to end up with a conclusive result.

The results of the simulation incorporating the usage of a gamification-based carpooling model described above were extremely interesting, ending up with a big reduction of the cars in the street.

No.	Variation Name	Number of Cars	Li/L0(%)
1	Referential	43855	100,0%
2	Auction, Constant Price	23854	54,4%
3	Price decreases exponentially	22330	50,9%
4	Price decreases proportionally	23280	53,1%
5	Impatient agents, price decreases exponentially	28850	65,8%
6	Impatient agents, price decreases proportionally	29830	68,0%
7	Patient agents, price decreases exponentially	16290	37,1%
8	Patient agents, price decreases proportionally	17490	39,9%

Table 2.1: Results of the Simulation Experiments, adapted from [OPT18]

As it is possible to see in Table 2.1 the number of cars has significantly been reduced. Even with the use of impatient agents and having the price decrease proportionately, which are both the worst cases, the experiment ends up being successful presenting an improvement of cars present in the trips in the order of 32%.

By the end of the simulation the authors concluded that "the cost of developing, testing, and parameterizing a mobile application that facilitates carpooling using gamification is relatively small" and "the effectiveness of stimulation of residents by using an aptly constructed gamification system and urban serious games is an incredibly interesting idea, including from a scientific standpoint".

2.4.2 Gamification in Parking

Parking is one of the most important parts of urban mobility. As said in [DGA16], the parking system plays a very important role in the urban traffic system and has a close relation with accidents and environmental pollution and represents an example of a smart mobility service [SZ14].

Nowadays, the usage of cars has increased a lot and the number of park slots has not matched this growth. The imbalance between the parking supply and the parking demand, the space available for parking, is very less compared to vehicle usage. People are always late to go somewhere, to do something and the car is the best way to get to the destination in the quickest way possible. Sometimes there are no available parking slots or even a reduced amount of them and usually the driver parks the car in the second row. Drivers don't always have time to properly park the car and the car is parked occupying more than one slot which invalidates the other parking slot for another driver to park his car.

The size of the vehicle, the design, and production of vehicles are done in such a way that they occupy a lot of space. Size of the vehicle has become a matter of luxury and status, as big cars represent bigger roles in today's society. Those bigger cars are a real concern since they occupy more space and then reducing the efficiency usage of parking slots.

In order to keep the parking experience in a Smart City a not so stressful experience, where the driver wastes his time and founds by an excessive waste of oil, can be created a parking management application. This would help the driver to see all the available spots to park his car before he gets there and to rent that place for a limited time to grant that place when he gets

there. Overall is a good idea but to make that idea better and to keep the drivers using properly the parking slots and to make them think in the other drivers, a gamification technique can be included.

In the same article referenced above, the authors proposed an application for drivers parking their cars in the parking slots all over the city following certain types of rules and that will reward them points. The methodology used was the smart parking method, in which the user can book their parking slots before going there just to save their time, this will help not only in searching and pre-booking parking slots but also can be used to identify the parked vehicle.

A gamified approach defined for this application was defined having in the count a point and leaderboard strategy, as this gamified version of parking can create a process of visual learning to the drivers. To gamify the parking process the concept is introduced for providing points to drivers in each of their parking activities. With a greater number of points, the more will be the chances of winning and climbing up on the leaderboard.

The gamification technique developed had well-defined point criteria that were based on categories:

- **Parking Levels**

  The types of parking and the locals are a very important restriction for parking, the difference between parking in parks categorized with different labels are rewarded with a different number of points. The parks are labeled with 3 different levels 1, 2 and 3 and the points that the user earn are 2, 3 and 5 points respectively.

- **Proper Parking**

  The way the user parks his car is important to maintain the sustainability of their application purpose. The users shall park their cars in a responsible way to not interfering with the parking slots near. The correct parking is rewarded with 5 points and the not so correct parking is rewarded with 3 points.

- **Driving Skills**

  In a way to protect the environment the time needed to park the car is really important. The CO2 emission is a real concern and the motor emissions of CO2 when the driver is parking is high, in order to improve this concern the time spent to park the car in a correct way is measured to reward the drivers that emit fewer pollutants into the atmosphere. The time needed to achieve the maximum points are the 2 minutes or less, that is rewarded with 10 points. Then, drivers that take 2-5 minutes to park their cars are rewarded with 8 points. Finally, the drivers that don't have that major parking skills and take more than 5 minutes are rewarded with just 5 points.

In the final, the drivers with the most amount of points are rewarded. The use of a leaderboard to show the drivers with the most points is a gamification technique to incentive the drivers to compete between them for a major spot in the leaderboard.

In summary, this approach of gamification uses the point system to reward the drivers. This approach makes drivers having more concern in parking the cars in the correct places (Parking Levels) in order to make better usage of the parks that are available. This approach also verifies if the drivers park correctly their cars (Proper Parking), in order to not disturb other parking slots that are near the parking slot used by the driver and makes the parking system a better system in always having all the slots available. Finally, in order to keep a sustainable environment is important to control the emission of pollutant gases to the atmosphere (Driving Skills) in order to preserve the environment is important that the drivers do not waste to much time in parking their cars to emit fewer pollutant gases.

The introduction of points into the parking process will motivate people to park their vehicles in a systematic way and will eliminate the problem of wrong parking. Finally, this opens up a wide range of possibilities for incentive mechanisms to be applied to transportation with various purposes [KMR+14].

2.5 Models and Frameworks

This section is meant to represent the already build **Frameworks** and **Models** that can help at the start of a new gamified application or system. In order to start a project in the area of gamification, is important to have a structured base of a gamified system. for that reason, there are some frameworks that help developers in building a good and structured gamification system.

Very articles defined frameworks and strategies to gamify a system or an application. In the following will be shown some of the strategies mentioned in the articles.

In the [MHWA17] the authors defined a seven-phase method to gamify a system in the proper way.

- **Project preparation** — This phase is defined as preparation as it is the phase where is defined the objectives list and defined all the requirements for the system and its gamification;
- **Analysis** — This phase is where is defined the context and its success metrics and is defined also the target audience, by identifying their needs and motivations;
- **Ideation** — This phase is responsible for the idealization of the ideas, the brainstorm, and consolidation;
- **Design** — This phase is the creation phase where all the ideas of the previous phase give rise to prototypes and where these prototypes are evaluated;
- **Implementation** — This phase uses the prototypes to reach a proper development and is where is implemented the gamification approach;
- **Evaluation** — This phase is responsible for the evaluation of the gamification approach developed in the previous phase;

- **Monitoring** — This phase is the continuous phase after the release is important to keep the system gamification technique updated and it is in this phase that this is done.

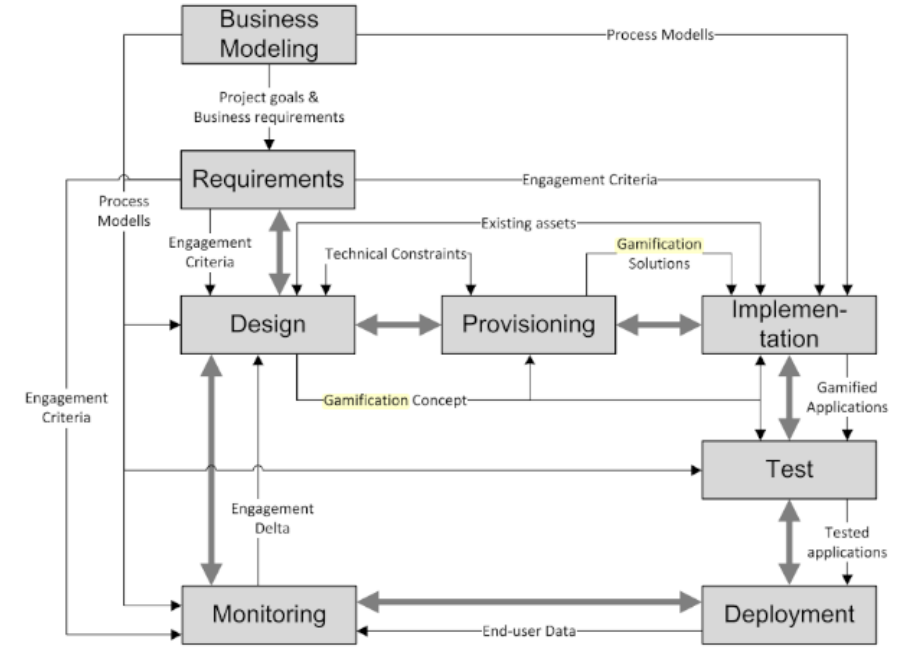


Figure 2.1: Gamification Model, adepted from [RW15]

As a validation of the model mentioned above, the book [RW15] confirmed the basic phases as it is possible to see in the figure 2.1 the phases of Requirements, Design, Implement, Evaluate/Test, Monitorize are present in both models.

Starts with a **Business Modeling** where are defined the models for the Business and the general objectives of the project, needed for the next step. The next phase is the **Requirements** phase where are defined all the requirements for the Gamification purpose, an analysis of the end user is made in order to achieve a list with reasons for and against the process as well as motivating factors. In the **Design** phase it is supposed to end up with a concept of the gamification technique to be implemented, this technique is playtested with already some end users. The **Provisioning** phase is responsible for analyzing the IT infrastructures needed to better implement the gamification designed in the previous phase. The **Implementation** phase is the phase where are assembled, implemented and integrated all components for the final gamified application, it is here that the gamified application or system is built. The **Test** phase it is like the name itself says the testing phase where the application or the system is tested. The **Deployment** phase is the phase after the test where the application or the system is sent to the end users and it is really used. The **Monitoring** phase is the phase of the continuous development of the system, where the improvements of the system are made and when it needed some improvement on the technique, is in this phase that is detected.

This model ends up to work the same way as the previous one but with a more visual solution.

After defining a method for the approach of the implementation of a Gamification technique is important that this gamification has a structure and an already defined model of idealization next is described a framework for a Gamification technique in a system.

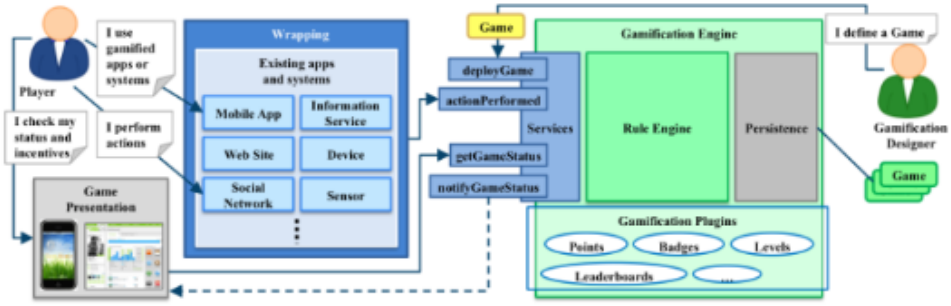


Figure 2.2: Gamification Framework, adapted from [KMP+15].

In Figure 2.2, from article [KMP+15] it is possible to see the example of a framework for the construction of a gamified system. The framework is composed of four big layers. Each of the layers represents a different state for the implementation of the framework. The four main layers are the Gamification Engine, the Wrapping layer, the Game Presentation layer the Gamification Plugins layer.

The **Gamification Engine** layer is the most important one, is the layer responsible for the instantiation and execution of the games related to the Gamification technique. The **Wrapping Layer** that is the layer responsible for all the communication between the Gamification Engine referenced above and all the engines as applications and systems responsible for the monitorization of the behavior of the user. The **Game Presentation Layer** is the layer responsible for the interaction with the user, the UI of the application and systems belong to this layer. The **Game Plugin Layer** is the layer responsible for the rewards of the users' behavior, is in this layer that is defined all the reward system and the gamification technique applied to the game, be it a scoring system, a leaderboard or a feedback system.

2.6 Summary

In order to conclude this Chapter, this section aims to show to the reader the main summary of the sections in the whole Literature Review. The Chapter contains three main sections, the first one reflects the Shared Mobility, the second one the Gamification and the third one is the section that converges both of the previous terms in the Gamification in Urban Mobility. The fourth section reflects the already established models and frameworks in the area of Gamification.

The **Shared Mobility** is a growing strategy emergent in today's Smart Cities being responsible for a great impact in the **Shared Economy**. **Shared Economy** is defined as a peer-to-peer type Economy where the users change services, products, etc. and the transactions are usually via community-based online services, being the **Shared Mobility** a great driving force in this type

of Economy. Shared Mobility is defined as the share of any mean of transportation with other users and has three big alternative transportation in **Carsharing**, **Ridesharing** and **Bikesharing**. The two means of alternative transportation approached in this dissertation are the **Carshring** and **Ridesharing**.

Carsharing is a car rental, attractive to customers who make only occasional use of a car on demand and the best characteristic is that the user can rent a car on an as-needed basis. **Rideshraing** is a mode of transportation in which users share their own vehicle for a trip in order to split travel costs like gas, or parking fees with other users which have similar itineraries and time schedules.

Having in count the most popular RideSharing applications presented in the 2.2.

Table 2.2: RideSharing Applications Analysis

System	Plataform	Driver	Passanger	Communities
Uber	- Web - Mobile	- Car Share 1. Make the car available 2. Act as a driver	- Search for the nearby driver 1. Define the Origin and Destination 2. Matching of Driver	- Open Community
Whim	- Mobile	- Trip Share 1. Define the planned trip 2. Make the trip available 3. Accept the other Users	- Search for the best match for the planned trip 1. Define the Origin and Destination 2. Match the possible trips	- Open Community
BlaBlaCar	- Web - Mobile	- Trip Share 1. Define the planned trip 2. Make the trip available 3. Accept the other Users	- Search for the best match for the planned trip 1. Define the Origin and Destination 2. Match the possible trips	- Open Community
Caronetas	- Web	- Trip Share 1. Define the planned trip 2. Make the trip available 3. Accept the other Users	- Search for the best match for the planned trip 1. Define the Origin and Destination 2. Match the possible trips	- Closed Communities - Creation of Groups - Share of Trips inside Groups

These strategies are great drivers of sustainability in Smart Cities, developing not only the Economy but slowing down some traffic problems, the ones that are caused by the excess of

vehicles in the roads or the ones caused by the difficult access to public or private transportation vehicles.

Users are inserted in communities nowadays, with the use of Social Networks and connection between the users of those systems. Users build a relationship by connecting between them, this relationship has a strength that can be measured by a lot of factors, including the common interests. That strength will help the users having an increased trust in the relation, consequently a better trust in the connection and the community. Increased trust in the community will make the user having more confidence using a system.

The **Gamification** strategy has seen its use increased in various systems and applications these days. **Gamification** can be defined as "the use of game design elements in non-game contexts". There are various types of Gamification, there is not a really well-defined categorization of them but Gamification can be divided into two types, the **Simple** and the **Advanced**.

These techniques have a great effect on the users, being used as a **Motivational Factor** making the user in some ways addicted to the non-game system.

The partnership between **Gamification** and **Urban Mobility** is not at all unknown, being present in some articles already published and in some application and systems already built. There are already built techniques for the Carsharing and Ridesharing system that were tested in a simulation of the city of Warsaw. A game technique meant to the drivers and passengers where the number of trips and quality of join the passengers rewarded the users. The Gamification of parking is an idea already built, with a game built for the users to rent their parking slots and gaining points for the choice of the correct park, the quality of parking and the quickness of parking.

Finally, the technique of Gamification of systems is being developed through the years and well-known people in this area have already built **Models** and **Frameworks** for the Gamification of systems. There were presented two **Models** which are concordant and divide the strategy of Gamification in seven/eight parts.

After the Model is important to define a **Framework** to a complete gamified system.

With the review of all these definitions, models, frameworks, etc. is possible to define the actual state of the art.

Chapter 3

Methodological Approach

In this Chapter the methodological approach is described in order to reach the solution for the problem itself.

It is also described the problem with more detail along with the proposed solution to solve this problem based on the Literature Review done in the last Chapter.

It starts with the **Problem Formalization**, explaining the reasons for the formulation and proposal of this dissertation. The problem is decomposed and divided into sub-problems for a better understanding of the main problem.

Then, a list of the **Solution Requirements** is shown containing all the requirements the solution needs in order to perform well.

Next, the **Solution Architecture** is presented, which will represent the approach used for the successful accomplishment of the task and the proposed architecture for that solution.

Almost ending, it is explained the evaluation methodology in the **Evaluation Approach**.

In the end, a brief **Summary** of all the sections described in this Chapter is presented.

3.1 Problem Statement

After the review of the literature, with the state of the art is possible to formulate a problem.

With the Literature Review is possible to see the actual state of the Shared Mobility, all the application that already exists prepared to give the user the solution for the mobility problems. There are a lot of applications that offer the user the possibility of sharing a ride with other users or seen from the other perspective, the possibility to join a ride another user proposed. Is also possible to rent your car to another user and thus create a possible source of income, while the user does not want to use his car. From another perspective, a user can rent a car for possibly a lower price and in a need basis.

These are all good solutions for mobility problems that were mentioned, but there is a factor that is not so well addressed the trust factor. Since all the applications and systems shown before

that try to break the barriers of mobility are for open communities, the end user doesn't always know who is the user that is on the other side. This lack of trust on the other users can cause mistrust in the system and make the user avoid to the maximum to use this type of mobility.

In order to fulfill this gap, a solution for this trusty users' problems is defined.

3.2 Solution Requirements

In order to reach a good solution is necessary to formalize a list of requirements that the application needs to achieve for good functionality, taking into account the proposed objective. The following list contains the main requirements for the system.

- **Create a Profile**

  As a possible User, the person needs to create an account in order to set his definitions and have all the information relative to his trips and connections.

- **Log In**

  The users that already have an account needs the possibility to log in into their accounts in order to manage all their information and to perform all the activities the system will provide.

- **See Profile**

  The User needs to see his profile in order to check if every information is correct and to see what the other users can see.

- **Create a Trip**

  The main objective of this system is to share trips so the User needs the possibility to create his own trip that he can share with other Users and wait for other User to request rides to that Trip that he can accept or reject.

- **Request Trip**

  There is the case that the User may not have the possibility to make the trip, so the User needs the possibility to request a Trip that he wants to perform and wait for the other User to answer his request with a ride to a Trip.

- **Edit Trip**

  The User needs the possibility to edit the Trip in order to edit some field that has changed like the price or the date or time.

- **Search Trip**

  One of the main features is the possibility of the User to Search Trips, offers or requests, in order for the User to answer to that Trips, requesting a ride or answering to a request offering a ride.

- **Request Ride**

  As said before the system needs to give the possibility to the User to ask for a Ride in a Trip that he wants to perform. The Rider of that Trip may or may not accept the ride request.

- **Accept Ride Request**

  In order to define the participants of the Trip, the Rider of the Trip can accept the ride requests that were sent.

- **Offer Ride**

  It was seen before that the User can request a trip, so in order to answer that requests, the User needs to offer a ride to User requesting, by creating or selecting an existing trip.

- **Accept Ride Offer**

  If an answer to the Trip request is made the User needs the possibility to accept the best offer, so the user can accept the trip that he wants the most.

- **See Itinerary**

  All of those Trips and Rides need to be together in order for the User to keep a track on all the Trips he is in.

- **See Carbon Footprint**

  One of the main topics of this dissertation is the Sustainability so the User needs the possibility to check his Carbon footprint in order to understand the impact of that shared trips.

- **Search People**

  As one of the main objectives of this dissertation is to create a Social Network, the User needs to search for people in order to create his connections.

- **See People Profiles**

  As the User is searching for other Users, the possibility of having more information offers the User more credibility of the other User.

- **Request Connection**

  If the User finds another User that he wants to have in his connections the system needs to provide the possibility of the User connect them.

- **Accept Connection Request**

  The Connections need acceptance for both parts, so the User that received the connection requests needs a way to accept or decline the request.

- **See all the Connections**

  All the connections of the User must be visualized in order to the User see the individual itinerary of that user or just see his profile.

- **See Connections Trips**

  As said in the previous point, the User may want to see the individual itinerary of a User in order to check a specific trip done by that user.

3.3 Solution Architecture

As said in the last sections, the major problems for the already built solutions for the mobility problems in the shared mobility area lie in the lack of trust and motivation by the users in the community using this type of systems.

In order to help the users to get a more reliable opinion on these systems and making them not always concerned with the risks of using this type of approach of mobility, a solution is needed. The solution that this dissertation aims to purpose is based on the communities, open and close, in the Internet field, using Social Network mechanisms.

This solution consists in differentiating the choice of the users based on the trust of the user in the other users. The creation of a social network that allows the Users to create their own network of contacts aims to give to the users the solution for the trust factor. The sharing of the itinerary of the users only with the user's connections restricts the recipients of the users' information of mobility, making him more at-eased with the system.

Another concern that this dissertation aims to overcome is the motivational factor, this factor is approached in the Literature Review in the figure of the Gamification techniques. As said, Gamification has an impact in the user in areas of Human perception, making the user feel in some way connected to that system. The Cognitive, Emotional and Social perceptions are the areas that by being stimulated, can cause a motivational factor for the person to use the system in a continuous way.

The approach of the solution consists of a gamified social network using a mobile application for shared mobility, in order to aggregate three important factors:

- **RideSharing** — In order to help the user in the mobility problem;

- **Conenctions' Network** — In order to solve the trust factor, not so addressed by other solutions;

- **Gamification** — In order to give a motivational factor.

In order to provide all the communications between Users and to have all the information not saved in each device, the Architecture of the System built based in Client/Server Architectures [NCG11].

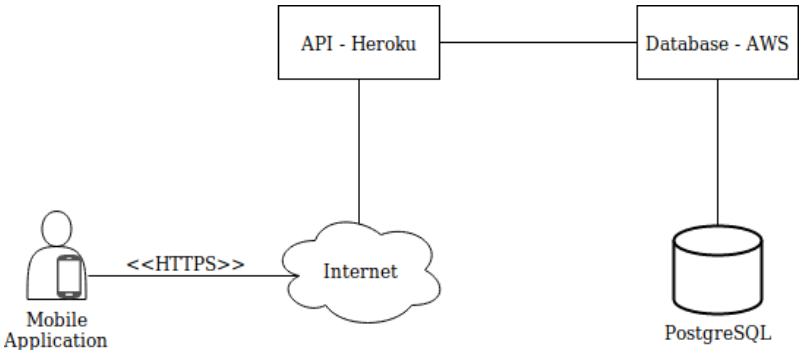


Figure 3.1: System Architecture

The proposed Solution 3.1 is divided into 3 main components. This components are representative of the Client/Server System and are briefly described below.

- **Mobile Application**

  This component is representative of the interface and is responsible for the communication of the System with the User. The User will interact only with the mobile application in order to use the system.

- **API**

  All the middle interactions between the data and the mobile application are handled by this component. Is responsible for handling the requests done by the User through the mobile application and then manage the data. Is also responsible for the collection of the data and consequent response for the mobile application.

- **Database**

  This is the component responsible for the large volumes of data storage and retrieval related to the System. All the information inserted by the User is stored and changed in here and then retrieved when needed.

3.4 Evaluation Approach

In order to verify the correct implementation of the Human-Computer Interaction, a set of rules is defined in [GT+04] for the interface design of mobile applications. The strategy used by the author was based on an older technique named Shneiderman's "Golden Rules of Interface Design" [SPC+16]. The author redefined some of the rules in order to better fit the needs of mobile applications.

Shneiderman's "Golden Rules of Interface Design" consists of eight rules:

1. **Enable frequent users to use shortcuts**

   Time is a critical aspect in people nowadays, saving them by reducing the number of operations the user needs to perform a specific task is an advantage.

2. **Offer informative feedback**

   As a user performs an action, there should be some system feedback, substantial and understandable feedback like error messages.

3. **Design dialogues to yield closure**

   In order to perform a task, is important to sequentially the steps to perform the action. The sequencing needs to have a beginning, middle, and end.

4. **Support internal locus of control**

   A very crucial point of a mobile application is that the User feels that can control the application and the application does what the user has asked for.

5. **Consistency**

   In order for the User to get the most complete experience, he needs to feel the app is robust. That offers consistency, that the app is capable to get and to retrieve all the information.

6. **Reversal of action**

   A mobile platform that is internet connection dependent in order to perform well needs a good connection to the internet but despite that, the application should rely the less possible on the internet connection.

7. **Error prevention and simple error handling**

   Mobile applications are often used because of the lack of time for the user to use a computer or other system in order to perform the action must quickly as possible, in order to that action be the quickest possible the user errors must be prevented and when they exist, they must be reversible.

8. **Reduce short-term memory load**

   User's short term memory is an issue. Users don't always memorize all the steps to complete an action, so the interface should help the user to better perform and remember the steps to perform the action. Intuitive application design will help the user to better perform.

9. **Design for Enjoyment**

   The author despite agreeing with those rules found necessary to define some other guidelines that could be useful and for one of those revealed critical for the core of this system.

Despite the most important part of a mobile application being the functionality and usability, there are some other factors that may contribute to the influence of the system in the user. The aesthetics of the application is a crucial factor for the user to like and enjoy using the application. It even creates a "positive affective response from the user". The color scheme and the overall design can also create an attraction to the application on the user.

3.5 Summary

In order to answer the problems defined in the previous chapter, this chapter aims to create a methodology that will be followed to better implement the solution.

In summary, the approach consists of a system, in this case, a mobile application. That mobile application will provide the users the possibility for the Users to create an account and to connect between them, like in a Social Network. Users will have the possibility to create trips they will perform and share them with other users, giving other users the possibility to request rides, that may be accepted. Users will also be able to request for specific trips, that requests can be answered with trips by other users offering rides to their trips.

The Solution will have three main components, Mobile Application, API and Database that will be responsible for the interaction with User, the intermediary between the interface and the data and the storage and retrieval of the data, respectively.

The application will need to be tested by some Users in order to have some feedback. The metrics that will be used to measure the quality of the application are the ones referred to in the previous subsection.

With all of those aspects of the methodology, is possible to implement the solution.

Chapter 4

Implementation

After the Methodological Approach defined in the previous Chapter, in order to achieve the complete solution was needed an implementation of the proposed solution.

In order to better understand this system, it had to be divided into several sections. Each one of the sections is responsible for something related to the Application, and besides working independently, in order for the application to proper work, all of them need to be working.

The Implementation is described in this Chapter in a more detailed way, taking into account all the methods referred to in the Methodological Approach Chapter.

Firstly, a description of the **Mobile Application** is presented, more specifically the main functionalities.

Then, the **API** is explained in a more detailed way, showing all the responsibilities.

Almost ending, the **Database** is described, all the components and data structures are detailed.

Finally, a brief **Summary** of all the components and characteristics addressed in the previous subsections are presented.

4.1 Mobile Application

This component is the core of the Application and has all the features implemented, represents the Application itself.

This component was built using React Native as this framework has proven to be one of the most used for native application in recent years. React Native uses native methods while the developer is using Web development tools. Development of mobile applications using React Native has advantages as the developer experience a cross-platform development [Eis15].

This component is responsible for all the communication between the User and the System, working as the Visual Interface for all the inputs of the Users and the retrieving of all the responses. This component has an enormity of features implemented that can be seen bellow.

The User will use his credentials in order to Sign In in the Screen in Figure 4.1, or if he doesn't have an account, he needs to proceed to all the steps in figures 4.2 in order to Sign Up. In the first step the User needs to define his personal data and credentials in the Screen in Figure 4.2a, then is required that the User defines at least one of the social networks in the Screen in Figure 4.2b, then the User chooses one or more areas of interest in the Screen in Figure 4.2c, finally the user can define the information relative to the conference in the Screen in Figure 4.2d.

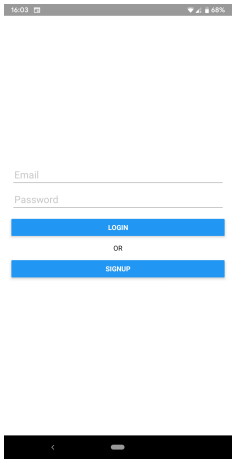
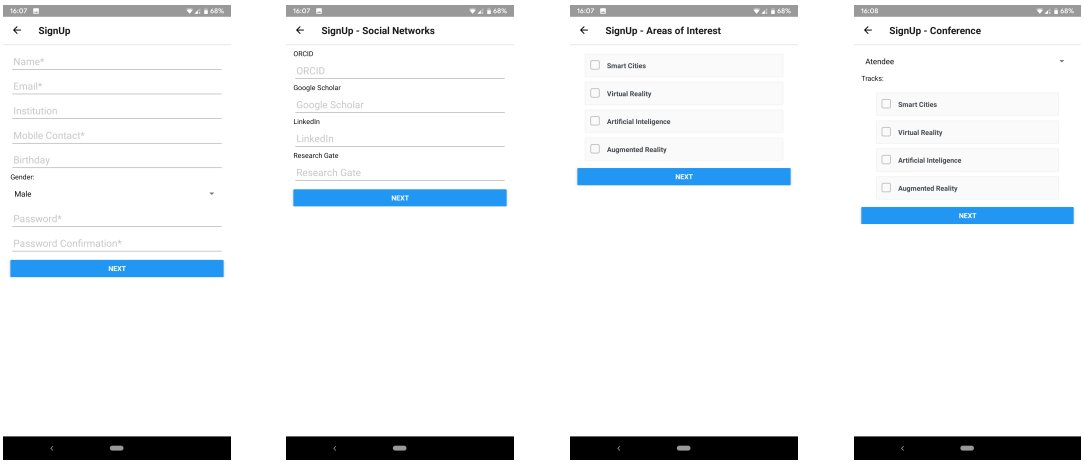


Figure 4.1: Sign In Screen



(a) Step 1 (b) Step 2 (c) Step 3 (d) Step 4

Figure 4.2: Sign Up Screens

After the Sign In or the Sign-Up, the User is redirected to the Main Menu in the Screen in Figure 4.4a. Here in this are present all the possibilities of actions the User can perform. Allied to this menu, there is a Side Menu in Figure 4.4b. Is from these menus that the User can perform all the activities in the application. This screen also allows the User to see his own Profile Screen in Figure 4.3 and perform the log out.

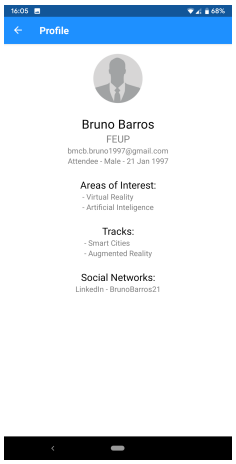
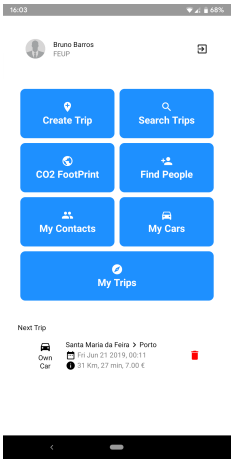
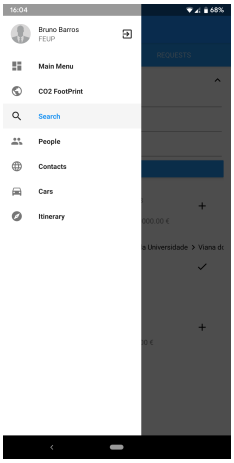


Figure 4.3: Profile Screen



(a) Main Menu Screen

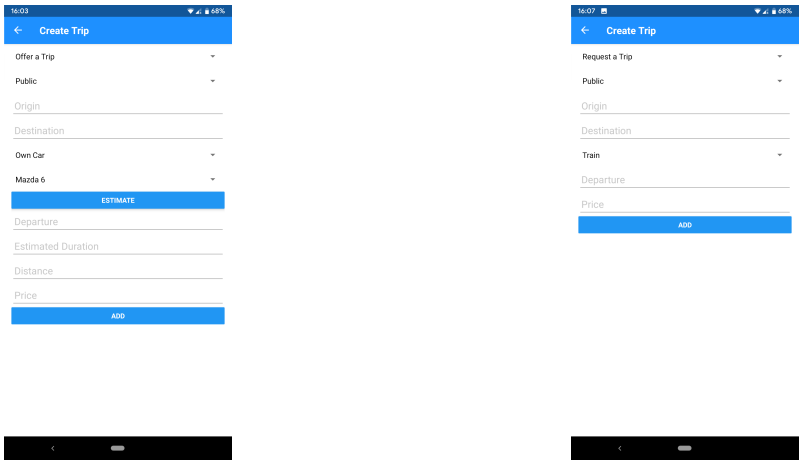


(b) Side Menu

Figure 4.4: Menu Screens

- **Create a Trip**

  Is in the Screen in Figure 4.5 that the User can Create, in Figure 4.5a, or Request, in Figure 4.5b, any Trip that he wants to perform. The User needs to fill all the fields in the forms, in order to create or request the Trip. The User needs to define first, the privacy of the Trip, then, define the Origin and the Destination, alongside with the mean of transportation that he plans to use. In the case of the Creation of a Trip, the user can estimate the distance and the time needed for the travel, but then if the User doesn't agree with the estimation, he can change the values. Finally, the user can define the cost of the Trip and Departure date and time.

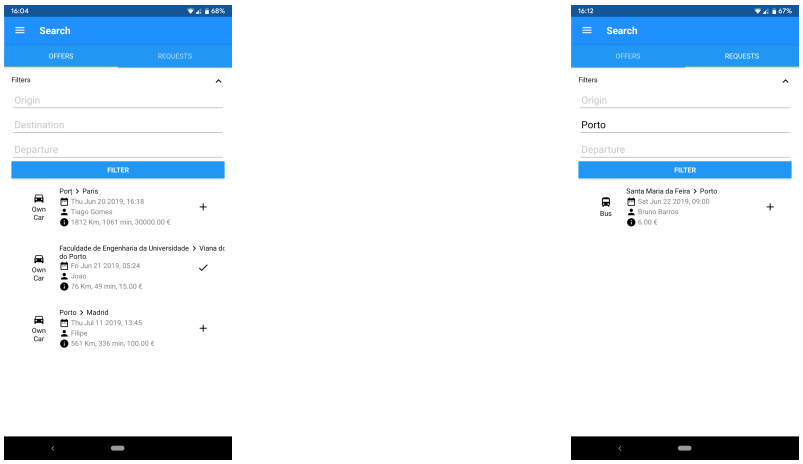


(a) Create Trip (b) Request Trip

Figure 4.5: Create and Request Trip Screens

- **Search Trips**

  In the Screen in Figure 4.6 the User can search for Trips. Trips that can be offered by other Users, in Figure 4.6a, or Trips the other Users are requesting, in Figure 4.6b. In both of the tabs of the screen is possible for the User to filter all the Trips by the Origin, Destination and Date and Time. In each of the options, the User can interact with the trip by requesting a ride in the Screen of Figure 4.6a or answering with a Trip in the Screen of Figure 4.6b.



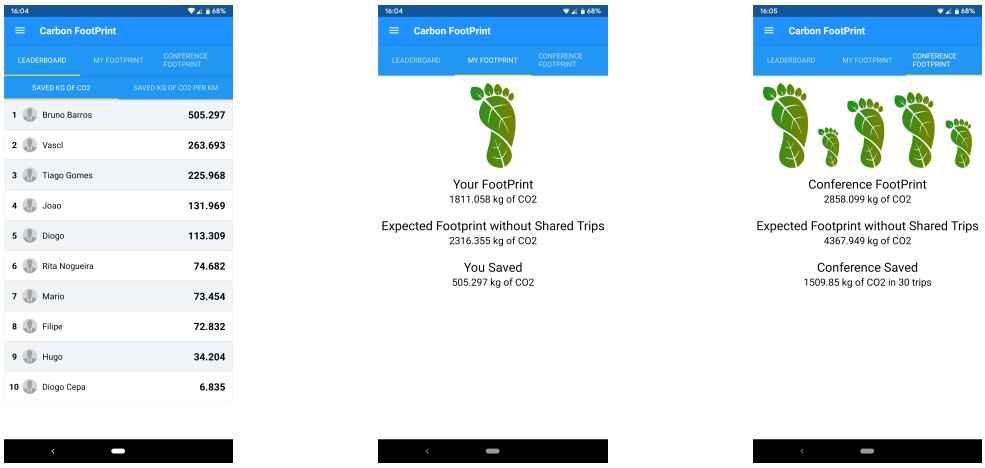
(a) Create Trip (b) Request Trip

Figure 4.6: Search Screen

- **CO2 Footprint**

  The User in the Screen in Figure 4.7 can visualize three possible tabs. The User can visualize a leader-board in the Screen of Figure 4.7a in which is present competition between the User and its Connections to see who saved more Carbon emissions to the atmosphere. The User can also visualize his own Carbon Footprint in the Screen of Figure 4.7b where the User

sees the total oh his emissions and the saved emissions. The same metrics but referent to the Conference the User can see in the Screen of Figure 4.7c.

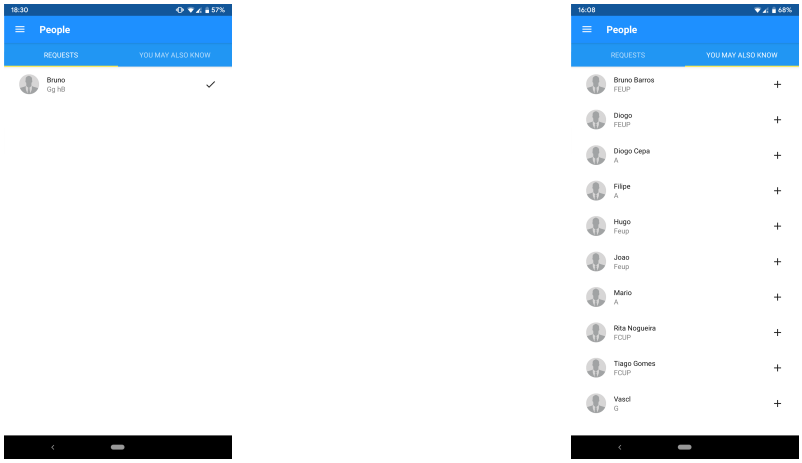


(a) LeaderBoard (b) Individual Footprint (c) Conference Footprint

Figure 4.7: CO2 Footprint

- **Find People**

The Screen in Figure 4.8 allows the User to manage the Requests of other Users. The User can answer other Users' request connections in the Screen in Figure 4.8a or searching Users to add to his connections in 4.8b. This screen allows the User to see other Users profile page in the Screen in Figure 4.3 in order to know more about each user.



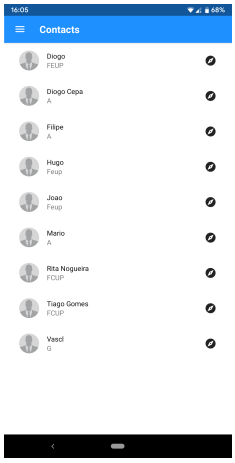
(a) Requests (b) You May Also Know

Figure 4.8: People Screen

- **Contacts**

In the Screen in Figure 4.9 the User can access the Users in his network. The User can see the other User profiles in the Screen in Figure 4.3 and also can see the User Itinerary in the

Screen in Figure 4.9b, where the User can see the other User's itinerary as well as interact with the Trips, allowing him to request a ride for a trip.



(a) Contacts Screen

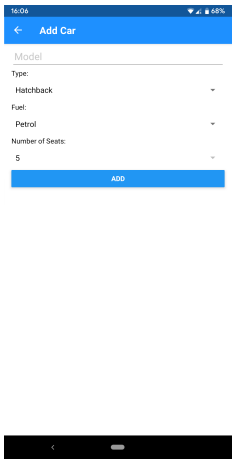
(b) User Itinerary

Figure 4.9: People Screen

- **Cars**

  The Screen in Figure 4.10 allows the User to manage his garage, more concretely manage his Cars available in the application. The User can see a list of all his cars in the application in the Screen in Figure 4.10a. The User can add more cars to his garage by completing the form present in the Screen in Figure 4.10b, filling the info in all the fields like the model, the type, the fuel and the number of seats. The cars can after be used in the creation of a trip in the Screen in Figure 4.5a, by choosing the option "Own Car" and selecting the specific car.

(a) Cars List

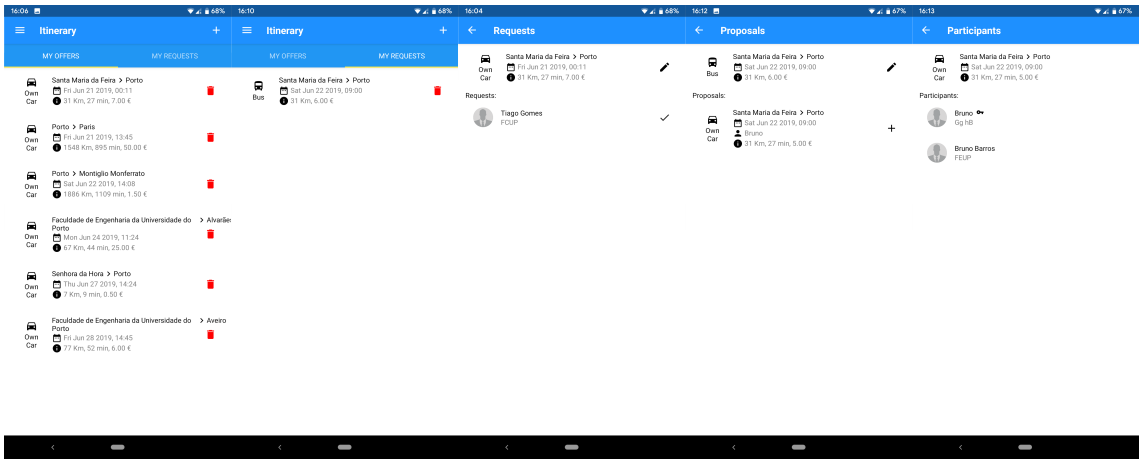


(b) Create Car

Figure 4.10: Cars Screen

- **Itinerary/Trips**

Is in this Screen in Figure 4.11 that the User can manage all of his Trips. The User can see his trips in the Screen in Figure 4.11a and his requests of trips in the Screen in Figure 4.11b. Here in this two screens, the User can see all the Trips he is in, the Trips that he offered and the Trips that he was accepted as Rider in the Screen in Figure 4.11a. User can access his Trips, he can see all the requests of the other Users to that trip in the Screen in Figure 4.11c, where the User can accept other Users to have a ride with him. If the User is having a ride with another User, he can see the participants of that trip in the Screen in Figure 4.11e. The user also can see all the proposals of other Users to his requested Trips in the Screen in Figure 4.11d, here the User can accept a proposal and then the Trip goes the User Itinerary in the Screen in Figure 4.11a.



(a) Offers (b) Requests (c) Ride Requests (d) Trip Proposals (e) Trip Participants

Figure 4.11: Itinerary Screen

4.2 API

This component is responsible for the communication between the Mobile Application and the Database. This component was built using NodeJS and ExpressJS as these frameworks are currently used in Web development for the creation of full-stack, middleware and RESTful API components [Bro14]. The use of these frameworks has increased in recent years, by the increasing of the use of JavaScript. The use of these frameworks alongside with the React Native in the native application was selected by the cross-platform support and the fact that the basis of all of them being JavaScript.

In this component is handled all the definitions of the Database, how it is modeled. When a function is triggered in the Mobile Application it is redirected to the API so the API triggers a function to execute the task that the user requested in the Application. The API has routes that can be accessed, all HTTP methods, GET, POST, PUT and DELETE. Those routes call functions that will trigger alterations in the database, for all the tables defined in the next section, like the creation of a new entry, an update or a delete of row.

All of those routes and the individual explanation can be accessed in the appendix A.

4.3 Database

This is the component responsible for the large volumes of data storage and retrieval related with the System, is here that all the data format is defined stored, retrieved and filtered and how all the data is interconnected. This Component was built using PostgreSQL by being one of the most advanced open source database server used nowadays [Mom01].

For this case was requested to create a data structure that was capable for the User to define his Itinerary, sharing trips, having Cars and more.

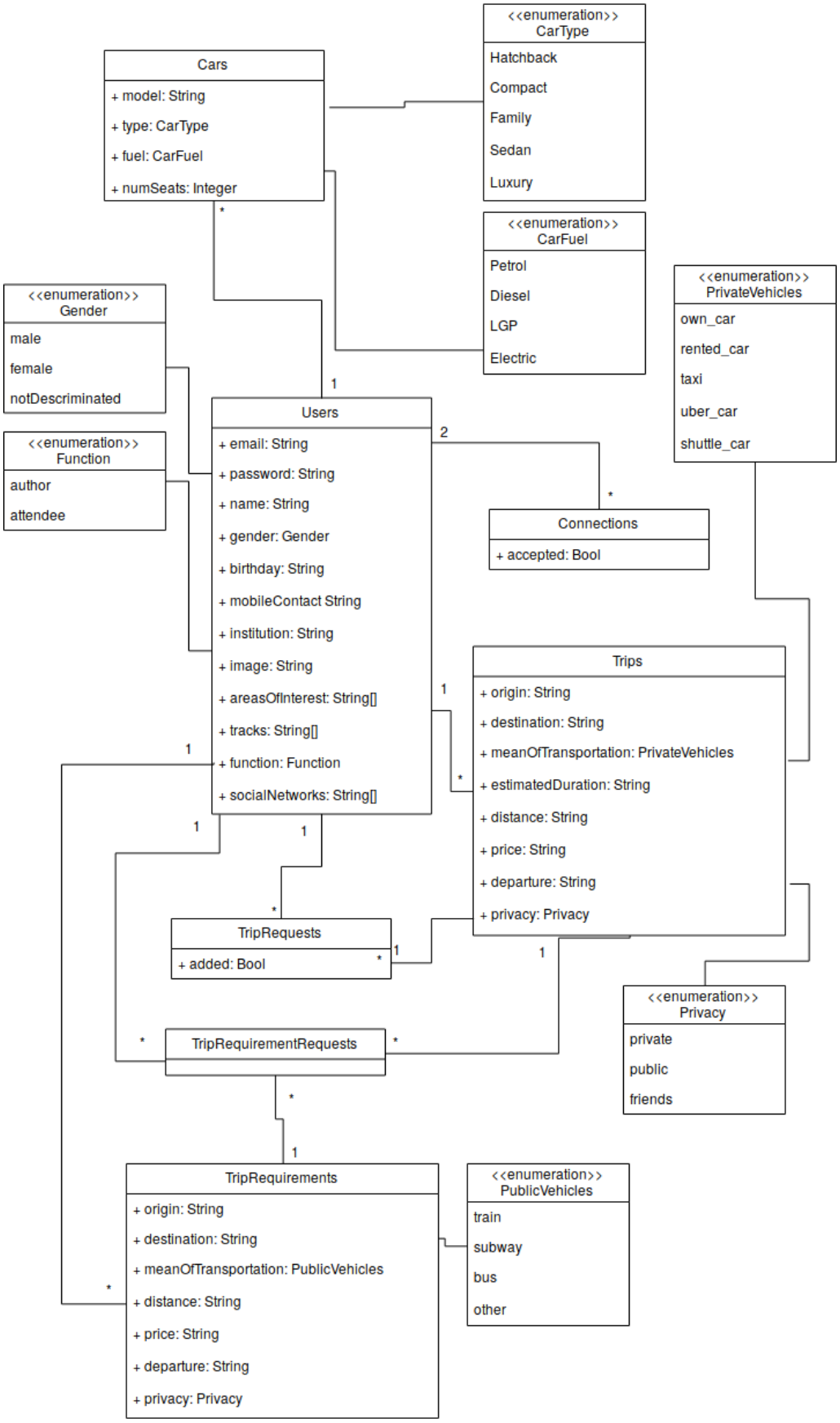


Figure 4.12: Database Architecture.

4.3.1 Data Structure

In Figure 4.12 is possible to see the Data Structure that was built in order to answer the Requirements defined before. The Data Structure is composed of 6 Tables, that defines the "objects" that are needed for the application to run. The abstraction of these "objects" defines their characteristics.

1. **Users**

   This Class is responsible to identify the User, containing all the information relative to the User in order to other Users and the System can recognize the person.

   Each User has to identify his personal characteristics like the name, email, gender, birthday and mobile contact. Each User also needs to create his own credentials in the password. Users define also more technical information in the institution that they come from, their areas of interest and their social networks. In order to inform the other Users and for a specific reason the User must fill the conference related fields as the function and define the tracks that he pretends to attend at the conference.

2. **Connections**

   This Class is responsible for the connection between Users in order to the Users creating their network.

   For each connection, there are two Users, user1 and user2 and an accepted field for the requests that are accepted. For each connection that is accepted, there is another one with the users switched.

3. **Trips**

   This Class is responsible for the representation of a Trip in a User's Itinerary, each Trip has relative information to the Trip in order to inform the other Users about the Trip.

   Each Trip has associated an origin and a destination in order to understand the trajectory of the trip, as well as the mean of transportation, the estimated duration, distance, price and departure date and hour for the other Users get to know what are the conditions that the Rider is offering. Each trip has privacy associated.

   Each Trip has a CO2 emission related, that is used to the calculation of each User Carbon Footprint and helps in the use of gamification using the CO2 emissions saved as a value to a leaderboard in order to motivate the User to reduce the Carbon emissions.

4. **Trip Requests**

   This Class is responsible for the representation of a request of a ride to another User's trip.

   For each Trip Request is associated the trip that the requester wants to participate in, the user in order to identify the User that is requesting the ride and finally a property added that represents the answer of the rider to the request, initially is false but if the rider accepts, the value turns true and the trip is added to the requester's itinerary.

5. **Trip Requirements**

   This Class is responsible for the representation of a public transportation Trip in a User's Itinerary, this trip is called a requirement for the instance of the User expecting a ride from the origin to the destination.

   Each Trip Requirement, like the Trip, has associated a origin and a destination in order to understand the trajectory of the public transportation trip, as well as the public mean of transportation that the user is expecting to use, the distance, price and departure date and hour for the other Users to know more about that trip and inform the conditions that he needs to get a ride. Each trip Requirement has privacy associated.

   Each Trip Requirement has a CO2 emission related, that is used to the calculation of each User Carbon Footprint and helps in the use of gamification using the CO2 emissions saved as a value to a leaderboard in order to motivate the User to reduce the Carbon emissions.

6. **Trip Requirement Requests**

   This Class is responsible for the representation of the responses to the Trip Requirements with trips, in order to fulfill the needs of the Users.

   For each Trip Requirement Request is associated the trip requirement that represents the public transportation trip of the requester, the user in order to identify the User that is offering a ride and finally the trip in order to show the requester the trip conditions the rider is offering. One Request can be accepted, once is done, the Trip Requirement is deleted and a Trip Request already accepted is created for the Trip.

7. **Car**

   This Class is responsible for the representation of the User's vehicles, in order to better calculate the Carbon Footprint of the Trip. A car can be associated twith a trip.

   Each Car has associated a model, a type, fuel and the number of seats in order to calculate better the Carbon Footprint.

4.3.2 Enumeration Structures

Attached to this model of data are associated with some Enumerations. This Enumeration represents a data type not standard defined that are the values correspondent to that category.

1. **Gender**

   This data structure represents the gender of the user and could be male, female or not discriminated.

2. **Function**

   The Function is the data structure that defines the function of the user in the conference, being an attendee or an author.

3. **Private Vehicles**

   This data structure contains all the vehicles that can be used on private trips. The vehicles are their own car, a rented car, a taxi, an uber or a shuttle.

4. **Public Vehicles**

   This data structure contains all the vehicles that can be used in public trips. The vehicles are train, subway, bus or other.

5. **Privacy**

   This data structure represents the privacy of trips, filtering the receptors. The options are private that meaning only the user can see, public for all the users of the application to see the trip or friends for only the users added to the user's network seeing.

6. **Car Type**

   This data structure represents the type of cars the user can choose for his car. The types are hatchback, compact, family, sedan or luxury.

7. **Car Fuel**

   In order to represent the type of fuel of the car was created this data structure. The fuels available to choose are petrol, diesel, LGP or electric.

4.4 Summary

In Summary, the system is composed of three elements in the Application, the API and the Database. The Application, built in React Native responsible to all the interactions with the User. The API, built in NodeJS and Express, is responsible for the communication between the application and the database. The Database, built in PostgreSQL, is responsible for the storage and retrieval of all the data of the system.

All of these three components represent all the system.

Chapter 5

Results and Analysis

In order to verify if the solution is good as well as if the implementation of the system has been successful is important to test the system. It is important to test the quality of the implementation of the application, in order to verify if the functionalities were all implemented and are working as it was supposed to be. It is also important to know if there are some features that are missing and or if some are not so well implemented. This section will then be responsible for showing if the implemented solution accomplished the features proposed in Chapter 4.

Firstly, all the information about the technique used to test the system is introduced, in this case using questionnaires, and all the information about the content and the formulation of the questionnaires.

Then, all the results of the questionnaires and the conclusions associated with each result are shown and what the meaning of each result.

Finally, a brief summary of all chapter and the test results is presented.

5.1 Questionnaires

There are some ways to see if the system checks all the points defined before. The strategy selected was using inquiries, present in Appendix B. These inquiries were done to 23 participants with ages ranging from 18 to 32 and were used to measure some satisfaction metrics. The users have experienced the application and answered some questions related to it and to the features it offers. This questionnaires also had the function of measuring the interest and the at-ease of persons in RideSharing.

The inquiry, initially, is divided into two parts:

1. **Before testing the application**

   The user answers some questions before testing the application in order to measure the users' interest and at-ease with Ridesharing and the actual platforms and social networks designed for that ending.

2. **After testing the application**

   This part is dedicated to the user answering some questions about the application itself, after testing the system.

The user after answering the first questions starts to test the application, following a task script where he has to guide himself. The task script has the following steps:

1. Sign up in the application
2. Offer a Trip
3. Connect with another person
4. Request a Ride of that User's Itinerary
5. Verify the CO2 Footprint
6. Search for a Trip
7. Request a Ride
8. Create a Trip Request
9. Accept an offer to the Trip Request
10. Create a Car
11. Associate the Car with the Trip
12. Verify again the CO2 Footprint

After completing these tasks the User needs to answer some questions about the experience in using the application.

This part of the Questionnaire is dived in seven sub-sections:

1. **Simplicity -** This section is dedicated to the User to answer some question about the simplicity of the achievement of the tasks performed in the experience.
2. **Perceptibility -** In this section, the User answers the same questions but related to the User perception of the achievement of the tasks.
3. **Informative Feedback -** The questions in this section are relative to the information retrieved when carrying out the tasks described above.
4. **Sequencing -** Likely as before in this section, the User answers questions about his perception of the sequencing of the achievement of the tasks.
5. **Error Handling -** Relative to error handling the User answers questions about the perception and the reversibility of the errors during the tasks.

6. **Design -** In order to evaluate the Design the User rates the design of the application relative to the Simplicity, Overall Appearance, and the Color Selection. The User can leave tips to improve the design.

7. **Features -** Finally, this section is an open response where the User can leave the information of the features that he liked the most, the least liked and what he thinks they may be missing.

The first question is repeated at the end in order to see if the application had an impact in the User's decisions.

After the execution of a reasonable number of tests is possible to reach some conclusions about Satisfaction Metrics.

5.2 Satisfaction Metrics

After the execution of the tests and the Users answered the inquiries, there were some conclusions that were possible to take.

As explained before the inquiry about the application was divided into seven sections. In order to verify all the User's satisfaction metric in all of the sections there were three types of question:

1. **Ranking** - Where the User ranked the task in a 1-5 metric, where 1 was very negative and 5 was very positive.

2. **Yes or No** - Where the User replied yes or no to the question

3. **Open Answer** - Where the User could write any response.

These questions were applied to the sections motioned above following the following criteria.

5.2.1 Simplicity

In this category, the questions refer to the simplicity of the execution of some of the tasks proposed in the task script, in this case, the creation. The user categorized the simplicity of the execution of the actions related to every task on a 1-5 scale.

The average User classification for this category is present in the Graphic 5.1.

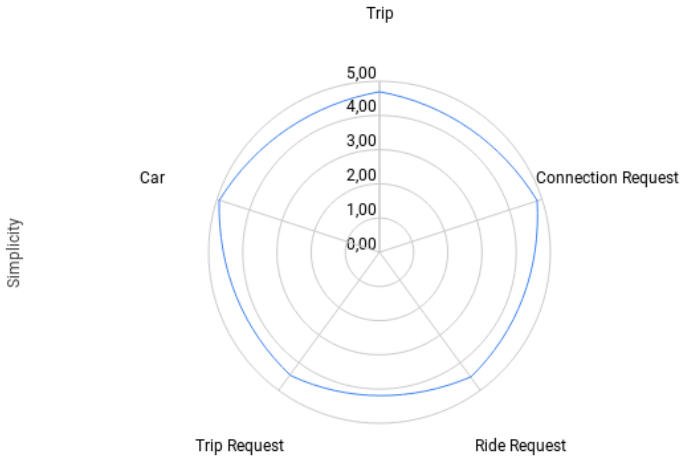


Figure 5.1: Ranking of Simplicity

Having in count the results is possible to assure that, as the results are close to the maximum value in less than one unit of measurement, the simplicity in execution of the creation of the tasks was very simple.

5.2.2 Perceptibility

In the category of Perceptibility, the questions refer to the perceptibility of the user when is performing the task of creation. The user categorized the perceptibility of the execution of the actions related to every task on a 1-5 scale.

The average User classification for this category is present in Graphic 5.2.

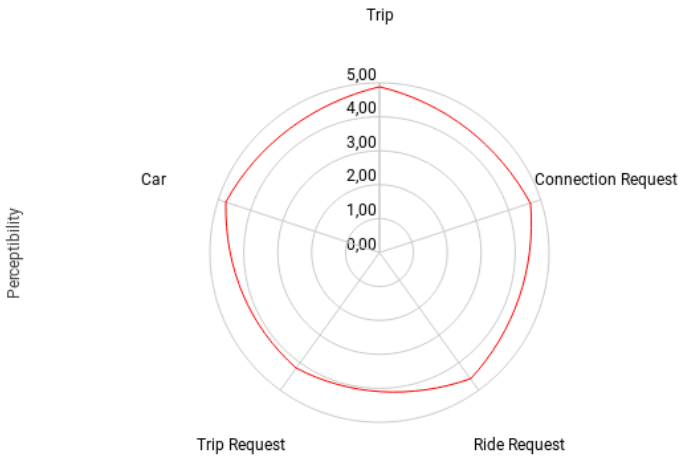


Figure 5.2: Ranking of Perceptibility

Looking at the results is possible to affirm that, as the results are close to the maximum value in less than one unit of measurement used, the users' perceptibility in the execution of the creation of the tasks is clear.

5.2.3 Informative Feedback

Relative to Informative Feedback, it was asked to the Users if after the execution of the realization of a task if the User received enough informative feedback by the application. The user answered Yes or No to questions related to the informative feedback of the execution of the actions.

The percentage results of positive questions for this category are present in Graphic 5.3.

Figure 5.3: Percentage of Positive Answers of Informative Feedback

After the look at those results, is possible to see that the results are not equally good. Despite almost all the values being relatively good, the value of the Ride Request is bellow relative to the others, this meaning the feedback the user gets after the realization of a Ride Request may not having enough feedback.

5.2.4 Sequencing

For the Sequencing category, the Users were asked if the sequencing of the realization of tasks has a logical sequencing. The user answered Yes or No to questions related to the sequence of the execution of the actions.

The percentage results of positive questions for this category are present in Graphic 5.4.

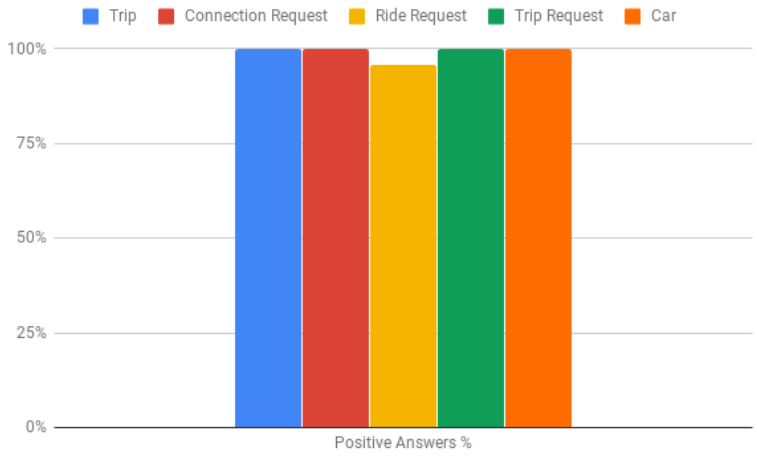


Figure 5.4: Percentage of Positive Answers of the Sequence of actions

By looking at those results is possible to affirm that the sequencing was successfully hit and the Users find the actions Sequenced in a logical order and with logical steps.

5.2.5 Error Handling

In the Error Handling category, two subcategories were had in count the perceptibility of the User's error by the own User and the ability to reverse the error himself. The user answered Yes or No to questions related to the perceptibility and the reversibility of the errors in the execution of the actions.

The percentage results of positive questions for this both categories of Errors are present in Graphic 5.5.

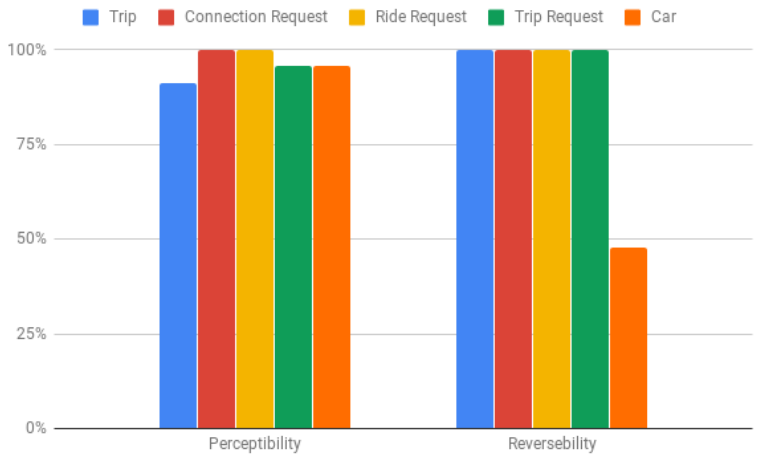


Figure 5.5: Percentage of Positive Answers of Erros Perceptibility and Reversibility

Is possible to come up with a conclusion by looking at those results. The ability for the Users to realize and to reverse an error is good, except for the reversibility of the Car, by the fact that a car is not editable.

5.2.6 Design

For the Design category, the User was asked about the opinion on the Design of the application. The user categorized the application in the different categories related to the design on a 1-5 scale and answered to a question related to the design about anything that he wanted to get better.

The average User classification for this category is present at the Graphic 5.6.

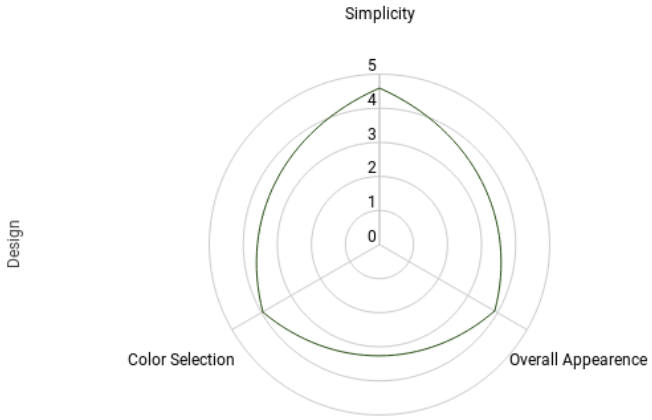


Figure 5.6: Ranking of Design

By the results at the graph is possible to see that the Users found the Design of the application simple, but that it was possible to improve in the categories of the overall appearance and the color selection, making the application more attractive and more relatable to the environment.

Users were asked to give some suggestions to improve the Design and the most used solutions were:

1. More relatable color (55%)
2. Dark mode (27%)

Those suggestions indicate mostly the same conclusions on the rate question, being a more relatable color the most suggested alteration and a dark mode in order to a better appearance.

5.2.7 Features

In the Features category, the questions were open and it gave the User the ability to choose and to justify their choice relative to the features. The user answered three questions, two related to the features implemented in the application, one about a feature that he liked to be implemented.

The most common answers in each of the categories were:

Category	Feature	Percentage
Most Valuable	1. Trip Request	48%
	2. Ride Request	35%
Least Valuable	1. Car	42%
	2. CO2 Footprint	33%
Missing	1. Notifications	44%

Table 5.1: Results of Categorization of Features

After looking at the table 5.1 presented above is possible to end up with the conclusion that the Trip Request and Ride Request are the User's favorites. The Users like the possibility to ask for a trip from a certain place to another and also the possibility to ask for a ride in an already defined trip.

In terms of the least appreciated features, the Car and the CO2 Footprint were the most voted. The Users don't find it necessary to define their own car in order to better calculate their carbon footprint mostly because a great part of the users doesn't find the carbon footprint feature a positive feature.

As an addition to the already implemented features, Users would like to have a notification system, in order to receive more feedback from the tasks that were performed.

5.2.8 Interest in RideSharing - Before vs After

This section is relative to the conjugation of two questions. Initially, Users were asked to answer three questions before using the application and after that experiment, they needed to answer the same three questions. The three questions were relative to the interest of the User in the categories of:

- Offering Trips
- Requesting Rides
- Sharing Itinerary

This questions were asked before and after, with the purpose of understand the effect that the application had in the Users in those areas, in order to see if the application and the social network system helped. The user categorized his interest in the RideSharing categories on a 1-5 scale.

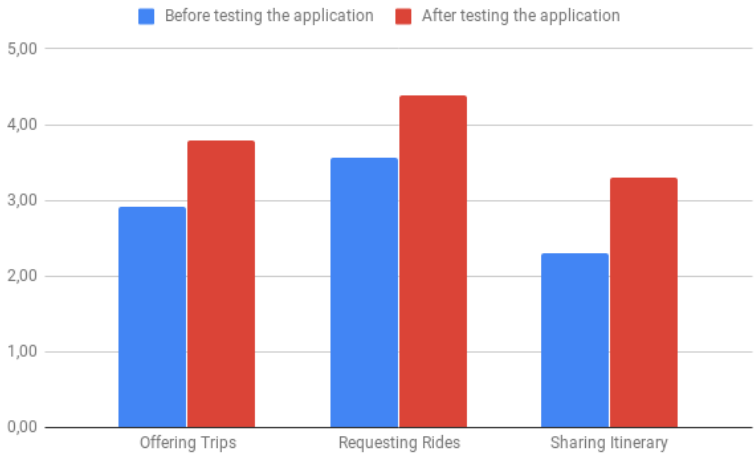


Figure 5.7: Average Responses of Users interest in RideSharing before and after testing the Application.

By looking at the graphic in Figure 5.7 is possible to see an increase in the average responses in nearly 1 value, after the use of the application in all of the three categories. This is indicative that users found in the application enough arguments to increase their trust and at-ease in the performance of Ridesharing. That said, is visibly understandable that the application and social network system improved the Users' trust and confidence with shared mobility.

5.3 Summary

After all the tests on real users resulted in positive feedback by them. The Users found the features of the application positive in the shared mobility scenario. The majority of the features implemented had positive feedback and led the users to a comfortable situation in sharing or requesting a trip or a ride.

The questionnaires helped also in the detection of not so good, not so well implemented or not so helpful features in the application, they also helped in finding some bugs and some situations in that the application needed to perform better.

After all the tests it was perceptible that the application is a strong help in the trust field as well as a big help in the ridesharing and shared mobility field. Helps in the share of information alongside a social network with user connections in order to improve the trust factor in the system.

Overall, the application had positive feedback and was considered by the users a good solution despite the need of some improvements in some areas and the addition of some new features. All of that being described in the future work section.

Chapter 6

Conclusions

In this Chapter, the reader is presented with a Summary of all the Chapters and Sections shown previously including some conclusions and observations. Firstly, an **Overview** of all the work done is presented. Then, the **Main Contributions** of this thesis are highlighted towards the solution of the problems previously identified. Finally, suggestions of what can be done as resulting research streams of this project are presented in the **Future Work** so as to make the system better, more useful, and extensible to other application domains.

6.1 Overview

In conclusion, this dissertation provided a mobile application. As mentioned before this application aimed to serve as a possible solution to the three aforementioned problems.

The Social Network provided the users with the possibility of creating their own network of contacts, making them more comfortable with potential companions of their trips and itineraries. The application is focused on the sharing of trips and itineraries performed, restricting the users that use the system so as to offer a pool of reliable users to form up groups of riders that share some comon characteristics. The Gamification Techniques aimed to motivate users not only to use the application but also to behave in a more sustainable manner.

The project associated with this dissertation resulted in a mobile application capable of meeting all characteristics identified as requirements for the devised solution. It was possible to see in the Results and Analysis that the application had a good feedback from the testers, that found the application useful and usable, despite some aspects that needing certainly to be further improved. In summary, the main three points and objectives with the application were achieved in this prototype version, ready to be deployed and tested in a real-world environment.

6.2 Contributions

In the end, the main contributions of the system were focused on the improvement of trustworthiness of users, the mobility, and the motivation. In summary, the main contributions resulting from

the implemented application can be summarized as follows:

- **Social Network:** the application allows users to form up a social network in the mobility domain, in order to find other users that they want to connect with and the ones that they know. This particularity aims to instill enough trust to support users' decision-making regarding their williness to share their itinerary.

- **Shared Mobility / RideSharing:** in order to help out torwards improving mobility systems, this application is used as a social network where users can share and request trips. They can also request rides and share their own itinerary with other users, from their connections or with other people.

- **Gamification:** to illustrate a possible solution for the motivational issues that the users face nowadays while using any kind of application or system, this application aims to offer the user a technique that not only provides the user with a motivational aspect but also represents a means to induce a more sustainable behavior of the user.

6.3 Future Work

Although the project of this dissertation ended up in a usable application, there are some aspects certainly offering room for further improvement and additional features to be implemented. Some of the improvements and needed features to be considered discussed and detailed below, which were mainly identified from the results of the questionnaires as previously presented in the Results and Analysis Chapter. Additionally to further improvements, there are some other features requiring additional investigation on more effective ways and approaches to be implemented. Some of such features and improvements are listed below.

- **Notifications:** the implementation of a notification system is one of the most important features to be added to the application. Not only the notification system in mobile devices but also the notification via e-mail from riders to the requesters, and vice-versa, in order to facilitate the initiation of the conversations between the two parts.

- **Chat:** as said in the previous point the conversations between the two parts in a shared trip is really important, so a chat feature that creates a chat room for the participants of a certain trip would be a very positive characteristic allowing for an easy-to-plan environment where riders can plan their itinerary together.

- **Matching Mechanism:** user's connection is a big part of the solution leveraging trust in the proposed solution. Having a matching mechanism that allow the user to know people that have the same kind of interests or if there is some other user from the same institution is also a desired future feature.

- **Payment Mechanism:** one of the hardest parts in the ridesharing mechanism is the payment; indeed, different methods of payment, using coins can be disadvantageous. Adding a payment method such as PayPal could be an improvement to take into consideration.
- **Map visualization:** a map visualization of the proposed trip can be an important feature to be added, the users can define more concretely the trip they want to perform and the other users can see what path the driver of their choice will follow for the trip.
- **Multiple stops trip:** in association with the previous feature a trip with multiple stops can be very advantageous as well. Other users can do half of the itinerary, or even request trips from/to a middle-way origin/destination on the itinerary.

Supported by the fact that all platforms, application, and API/Server are done in the same language, despite the fact of using different frameworks, it facilitates the implementation of the previously defined features. The solutions presented can also be implemented in different contexts, allowing for a more flexible and extensible platform.

- **Garbage Collection:** in order to keep sustainability as a pivotal idea, the implementation of the same solution in a garbage collection context would help to improve the sustainability of the environment, as well as communities in general. One of the metrics for the gamification of the system could take into account the number and the type of places selected for the collection and also the quantity of garbage collected by each individual.
- **Physical Well-Being:** the sustainability of the population does not only depend on the environment, but the physical well-being of people is also a great factor for the population to unfold and cultivate sustainable behaviors. The possibility of the creation of sporting events and share them with other users is a possible implementation to have into account. The metrics of gamification to account for could be the hours of physical exercise of each individual amongst a group of friends, for instance.

References

[ACZ+13] Marika Arena, Federico Cheli, Dario Zaninelli, Alfonso Capasso, Regina Lamedica, and Antonio Piccolo. Smart mobility for sustainability. In *AEIT Annual Conference, 2013*, pages 1–6. IEEE, 2013.

[ARH00] Alfarez Abdul-Rahman and Stephen Hailes. Supporting trust in virtual communities. In *Proceedings of the 33rd annual Hawaii international conference on system sciences*, pages 9–pp. IEEE, 2000.

[BML10] Moira Burke, Cameron Marlow, and Thomas Lento. Social network activity and social well-being. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1909–1912. ACM, 2010.

[Bro14] Ethan Brown. *Web development with node and express: leveraging the JavaScript stack*. " O'Reilly Media, Inc.", 2014.

[CDBN11] Andrea Caragliu, Chiara Del Bo, and Peter Nijkamp. Smart cities in europe. *Journal of urban technology*, 18(2):65–82, 2011.

[CK14] Boyd Cohen and Jan Kietzmann. Ride on! mobility business models for the sharing economy. *Organization & Environment*, 27(3):279–296, 2014.

[CSR+18] J. Cruz, E. Silva, R. J. F. Rossetti, D. C. Silva, E. C. Oliveira, and J. Neto. Application of multi-agent systems to shared transport services: A review. In *2018 13th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–6, June 2018.

[DDKN11] Sebastian Deterding, Dan Dixon, Rilla Khaled, and Lennart Nacke. From game design elements to gamefulness: defining gamification. In *Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments*, pages 9–15. ACM, 2011.

[DGA16] Kavisha Duggal, Lovi Raj Gupta, and Sravani Ammu. Gamification leads to smart parking. *International Journal of Computer Applications*, 144(10), 2016.

[Eis15] Bonnie Eisenman. *Learning react native: building native mobile apps with javascript*. " O'Reilly Media, Inc.", 2015.

[EW09] M Max Evans and Anthony KP Wensley. Predicting the influence of network structure on trust in knowledge communities: Addressing the interconnectedness of four network principles and trust. *Electronic Journal of Knowledge Management*, 7(1), 2009.

[FDO+13] Masabumi Furuhata, Maged Dessouky, Fernando Ordóñez, Marc-Etienne Brunet, Xiaoqing Wang, and Sven Koenig. Ridesharing: The state-of-the-art and future directions. *Transportation Research Part B: Methodological*, 57:28–46, 2013.

[FMLR16] Panagiotis Fotaris, Theodoros Mastoras, Richard Leinfellner, and Yasmine Rosunally. Climbing up the leaderboard: An empirical study of applying gamification techniques to a computer programming class. *Electronic Journal of e-learning*, 14(2):94–110, 2016.

[GPS15] Paula Gori, Pier Luigi Parcu, and Maria Luisa Stasi. Smart cities and sharing economy. 2015.

[GT+04] Jun Gong, Peter Tarasewich, et al. Guidelines for handheld mobile device interface design. In *Proceedings of DSI 2004 Annual Meeting*, pages 3751–3756, 2004.

[GVRM+17] Mario García-Valdez, José-C Romero, Alejandra Mancilla, Juan-J Merelo, and Francisco Fernández-de Vega. Gamification techniques in collaborative interactive evolutionary computation. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pages 87–88. ACM, 2017.

[HO07] Urban Housing and Mobility Studies OTB. Smart cities ranking of european medium-sized cities. 2007.

[HSU16] Juho Hamari, Mimmi Sjöklint, and Antti Ukkonen. The sharing economy: Why people participate in collaborative consumption. *Journal of the association for information science and technology*, 67(9):2047–2059, 2016.

[Kat03] Richard Katzev. Car sharing: A new approach to urban transportation problems. *Analyses of Social Issues and Public Policy*, 3(1):65–86, 2003.

[KMP+15] Raman Kazhamiakin, Annapaola Marconi, Mirko Perillo, Marco Pistore, Giuseppe Valetto, Luca Piras, Francesco Avesani, and Nicola Perri. Using gamification to incentivize sustainable urban mobility. In *Smart Cities Conference (ISC2), 2015 IEEE First International*, pages 1–6. IEEE, 2015.

[KMR+14] Z. Kokkinogenis, N. Monteiro, R. J. F. Rossetti, A. L. C. Bazzan, and P. Campos. Policy and incentive designs evaluation: A social-oriented framework for artificial transportation systems. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 151–156, Oct 2014.

[KQWW13] Sven O Krumke, Alain Quilliot, Annegret K Wagler, and Jan-Thierry Wegener. Models and algorithms for carsharing systems and related problems. *Electronic Notes in Discrete Mathematics*, 44:201–206, 2013.

[KS15] Oliver Korn and Albrecht Schmidt. Gamification of business processes: Redesigning work in production and service industry. *Procedia Manufacturing*, 3:3424–3431, 2015.

[LH11] Joey J Lee and Jessica Hammer. Gamification in education: What, how, why bother? *Academic exchange quarterly*, 15(2):146, 2011.

[MA15] Duncan McLaren and Julian Agyeman. *Sharing cities: a case for truly smart and sustainable cities*. MIT Press, 2015.

[MDARM+15] JA Rodríguez Martín, C De Arana, JJ Ramos-Miras, C Gil, and R Boluda. Impact of 70 years urban growth associated with heavy metal pollution. *Environmental Pollution*, 196:156–163, 2015.

[MHWA17] Benedikt Morschheuser, Juho Hamari, Karl Werder, and Julian Abe. How to gamify? a method for designing gamification. 2017.

[Mom01] Bruce Momjian. *PostgreSQL: introduction and concepts*, volume 192. Addison-Wesley New York, 2001.

[NCG11] Antonio Nasuto, Giuseppe Cassone, and Danilo Gotta. Method and system for monitoring performance of a client-server architecture, April 26 2011. US Patent 7,933,988.

[NTS+15] Adina Letiţia Negruşa, Valentin Toader, Aurelian Sofică, Mihaela Filofteia Tutunea, and Rozalia Veronica Rus. Exploring gamification techniques and applications for sustainable tourism. *Sustainability*, 7(8):11160–11189, 2015.

[OPT18] Robert Olszewski, Piotr Pałka, and Agnieszka Turek. Solving "smart city" transport problems by designing carpooling gamification schemes with multi-agent systems: The case of the so-called "mordor of warsaw". *Sensors*, 18(1):141, 2018.

[RAKG13] R. J. F. Rossetti, J. E. Almeida, Z. Kokkinogenis, and J. Gonçalves. Playing transportation seriously: Applications of serious games to artificial transportation systems. *IEEE Intelligent Systems*, 28(4):107–112, July 2013.

[RBK+17] Michael Ro, Markus Brauer, Kathy Kuntz, Raj Shukla, and Ingo Bensch. Making cool choices for sustainability: Testing the effectiveness of a game-based approach to promoting pro-environmental behaviors. *Journal of Environmental Psychology*, 53:20–30, 2017.

[RW15] Torsten Reiners and Lincoln C Wood. *Gamification in Education and Business*. Springer, 2015.

[SB15] Tom Saunders and Peter Baeck. Rethinking smart cities from the ground up. *London: Nesta*, 2015.

[SCBC15] Susan Shaheen, Nelson Chan, Apaar Bansal, and Adam Cohen. Shared mobility: A sustainability & technologies workshop: definitions, industry developments, and early understanding. 2015.

[Sco88] John Scott. Social network analysis. *Sociology*, 22(1):109–127, 1988.

[SKC+16] E. Silva, Z. Kokkinogenis, Á. Câmara, J. Ulisses, J. Urbano, D. C. Silva, E. Oliveira, and R. J. F. Rossetti. An exploratory study of taxi sharing schemas. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pages 247–252, Nov 2016.

[SM92] Gert Spaargaren and Arthur PJ Mol. Sociology, environment, and modernity: Ecological modernization as a theory of social change. *Society & natural resources*, 5(4):323–344, 1992.

[SPC+16] Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs, Niklas Elmqvist, and Nicholas Diakopoulos. *Designing the user interface: strategies for effective human-computer interaction*. Pearson, 2016.

[SRKP16] E. Silva, R. J. F. Rossetti, Z. Kokkinogenis, and J. Pinto. A systematic review protocol on shared transportation. In *2016 11th Iberian Conference on Information Systems and Technologies (CISTI)*, pages 1–4, June 2016.

[SRS+14] Paolo Santi, Giovanni Resta, Michael Szell, Stanislav Sobolevsky, Steven H Strogatz, and Carlo Ratti. Quantifying the benefits of vehicle pooling with shareability networks. *Proceedings of the National Academy of Sciences*, 111(37):13290–13294, 2014.

[SZ14] Andrea Sassi and Franco Zambonelli. Towards an agent coordination framework for smart mobility services. In *8th International workshop on agents in traffic and transportation (May 2014)*, 2014.

[TLT+11] Chenhao Tan, Lillian Lee, Jie Tang, Long Jiang, Ming Zhou, and Ping Li. User-level sentiment analysis incorporating social networks. In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1397–1405. ACM, 2011.

[XNR10] Rongjing Xiang, Jennifer Neville, and Monica Rogati. Modeling relationship strength in online social networks. In *Proceedings of the 19th international conference on World wide web*, pages 981–990. ACM, 2010.

[ZC11] Gabe Zichermann and Christopher Cunningham. *Gamification by design: Implementing game mechanics in web and mobile apps*. " O'Reilly Media, Inc.", 2011.

[Zyg13] Sotiris Zygiaris. Smart city reference model: Assisting planners to conceptualize the building of smart city innovation ecosystems. *Journal of the Knowledge Economy*, 4(2):217–231, 2013.

Appendix A

API Routes

In this appendix is present all the routes of the API accompanied by an explanation of the function.

A.1 User

A.1.1 Create User

Route to create an User.

- **URL:**

```
/users/create
```

- **Method:**

```
POST
```

- **Params:**

```
No params
```

- **Body:**

```
{
    name: STRING,
    email: STRING,
    password: STRING,
    passwordConfirmation: STRING,
    institution: STRING,
```

```
        gender: STRING,
        birthday: STRING,
        mobileContact: STRING,
        function: STRING,
        areasOfInterest: [STRING],
        tracks: [STRING],
        socialNetworks: [STRING]
     }
```

- **Response:**

```
   {
        id: INTEGER,
        email: STRING,
        name: STRING,
        institution: STRING,
        image: STRING,
        areasOfInterest: [STRING],
        gender: STRING,
        birthday: STRING,
        mobileContact: STRING,
        tracks: [STRING],
        function: STRING,
        socialNetworks: [STRING],
        token: STRING
  }
```

A.1.2 User Information

Route to get information about the User.

- **URL:**

```
   /users/:id
```

- **Method:**

```
   GET
```

- **Params:**

```
   id: INTEGER
```

- **Body:**

```
No Body
```

- **Response:**

```
{
    id: INTEGER,
    email: STRING,
    name: STRING,
    institution: STRING,
    image: STRING,
    areasOfInterest: [STRING],
    gender: STRING,
    birthday: STRING,
    mobileContact: STRING,
    tracks: [STRING],
    function: STRING,
    socialNetworks: [STRING],
    token: STRING
}
```

A.1.3 Email Existence

Route to verify existence of the email.

- **URL:**

```
/users/verify/:email
```

- **Method:**

```
GET
```

- **Params:**

```
email: STRING
```

- **Body:**

```
No Body.
```

- **Response:**

```
{
    message: 'Email free'
}
```

or

```
{
    message: 'Email already in use'
}
```

A.1.4 Log In

Route to perform the login.

- **URL:**

```
/login
```

- **Method:**

```
POST
```

- **Params:**

```
No Params.
```

- **Body:**

```
{
    email: STRING,
    password: STRING
}
```

- **Response:**

```
{
    id: INTEGER,
    email: STRING,
    name: STRING,
    institution: STRING,
    image: STRING,
    areasOfInterest: [STRING],
    gender: STRING,
    birthday: STRING,
    mobileContact: STRING,
    tracks: [STRING],
    function: STRING,
    socialNetworks: [STRING],
    token: STRING
}
```

A.2 Connections

A.2.1 Create Connection

Route to create a connection between two Users.

- **URL:**

```
/connections/create
```

- **Method:**

```
POST
```

- **Params:**

```
No Params.
```

- **Body:**

```
{
    user1: INTEGER,
    user2: INTEGER
```

```
    }
```

- **Response:**

```
    {
        user1: INTEGER,
        user2: INTEGER,
        accepted: BOOLEAN
    }
```

A.2.2 Connection Acceptance

Route to accept a Connection Request.

- **URL:**

```
    /connections/accept
```

- **Method:**

```
    POST
```

- **Params:**

```
    No Params.
```

- **Body:**

```
    {
        user1: INTEGER,
        user2: INTEGER
    }
```

- **Response:**

```
    {
        user1: INTEGER,
        user2: INTEGER,
```

```
        accepted: BOOLEAN
    }
```

A.2.3 User Connections

Route top get all the Users in the User's Connections.

- **URL:**

```
    /connections/:id
```

- **Method:**

```
    GET
```

- **Params:**

```
    id: INTEGER
```

- **Body:**

```
    No Body.
```

- **Response:**

```
    [
        {
            id: INTEGER,
            email: STRING,
            name: STRING,
            institution: STRING,
            image: STRING,
            areasOfInterest: [STRING],
            gender: STRING,
            birthday: STRING,
            mobileContact: STRING,
            tracks: [STRING],
            function: STRING,
            socialNetworks: [STRING]
        },
```

```
        ...
    ]
```

A.2.4 Connection Requests

Route to get all the Users who sent Connection Requests.

- **URL:**

```
    /connections/requests/:id
```

- **Method:**

```
    GET
```

- **Params:**

```
    id: INTEGER
```

- **Body:**

```
    No Body.
```

- **Response:**

```
    [
        {
            id: INTEGER,
            email: STRING,
            name: STRING,
            institution: STRING,
            image: STRING,
            areasOfInterest: [STRING],
            gender: STRING,
            birthday: STRING,
            mobileContact: STRING,
            tracks: [STRING],
            function: STRING,
            socialNetworks: [STRING]
        },
```

```
        ...
    ]
```

A.2.5 Find People

Route to show Users that the User may want to add to the Connections.

- **URL:**

```
    /people/:id
```

- **Method:**

```
    GET
```

- **Params:**

```
    id: INTEGER
```

- **Body:**

```
    No Body.
```

- **Response:**

```
    [
        {
            id: INTEGER,
            email: STRING,
            name: STRING,
            institution: STRING,
            image: STRING,
            areasOfInterest: [STRING],
            gender: STRING,
            birthday: STRING,
            mobileContact: STRING,
            tracks: [STRING],
            function: STRING,
            socialNetworks: [STRING]
        },
```

```
        ...
    ]
```

A.3 Trips

A.3.1 Add Trip

Route to add a Trip to the Itinerary.

- **URL:**

```
    /trips/:id/add
```

- **Method:**

```
    POST
```

- **Params:**

```
    id: INTEGER
```

- **Body:**

```
    {
        origin: STRING,
        destination: STRING,
        image: STRING,
        estimatedDuration: STRING,
        distance: STRING,
        price: SRING,
        departure: STRING,
        arrival: STRING,
        privacy: STRING,
        car: INTEGER
    }
```

- **Response:**

```
    {
```

```
        id: INTEGER,
        origin: STRING,
        destination: STRING,
        image: STRING,
        estimatedDuration: STRING,
        distance: STRING,
        price: SRING,
        departure: STRING,
        arrival: STRING,
        privacy: STRING,
        car: INTEGER
    }
```

A.3.2 Edit Trip

Route to edit the Trip.

- **URL:**

```
    /trips/:user_id/:id/edit
```

- **Method:**

```
    PUT
```

- **Params:**

```
    user_id: INTEGER
    id: INTEGER
```

- **Body:**

```
    {
        origin: STRING,
        destination: STRING,
        image: STRING,
        estimatedDuration: STRING,
        distance: STRING,
        price: SRING,
        departure: STRING,
        arrival: STRING,
        privacy: STRING,
```

```
        car: INTEGER
    }
```

- **Response:**

```
    {
        id: INTEGER,
        origin: STRING,
        destination: STRING,
        image: STRING,
        estimatedDuration: STRING,
        distance: STRING,
        price: SRING,
        departure: STRING,
        arrival: STRING,
        privacy: STRING,
        car: INTEGER
    }
```

A.3.3 Trip Information

Route to get the Trip Information.

- **URL:**

```
    /trips/:id
```

- **Method:**

```
    GET
```

- **Params:**

```
    id: INTEGER
```

- **Body:**

```
    No Body.
```

- **Response:**

```
{
    id: INTEGER,
    origin: STRING,
    destination: STRING,
    image: STRING,
    estimatedDuration: STRING,
    distance: STRING,
    price: SRING,
    departure: STRING,
    arrival: STRING,
    privacy: STRING,
    car: INTEGER
}
```

A.3.4 Delete Trip

Route to delete a Trip from User Itinerary.

- **URL:**

```
/trips/:id/:user
```

- **Method:**

```
DELETE
```

- **Params:**

```
id: INTEGER
user: INTEGER
```

- **Body:**

```
No Body.
```

- **Response:**

```
{
    message: 'done'
}
```

A.3.5 Itinerary

Route to get User Itinerary.

- **URL:**

```
/itinerary/:id
```

- **Method:**

```
GET
```

- **Params:**

```
id: INTEGER
```

- **Body:**

```
No Body.
```

- **Response:**

```
[
    {
        id: INTEGER,
        origin: STRING,
        destination: STRING,
        image: STRING,
        estimatedDuration: STRING,
        distance: STRING,
        price: SRING,
        departure: STRING,
        arrival: STRING,
        privacy: STRING,
        car: INTEGER
    },
```

```
        ...
    ]
```

A.4 Ride Requests

A.4.1 Create Request

Route to create a Ride Request.

- **URL:**

```
    /trips/:trip/request
```

- **Method:**

```
    POST
```

- **Params:**

```
    trip: INTEGER
```

- **Body:**

```
    {
        user: INTEGER
    }
```

- **Response:**

```
    {
        trip: INTEGER,
        user: INTEGER,
        added: BOOLEAN
    }
```

A.4.2 All Requests

Route to get all Ride Request.

- **URL:**

```
/trips/:trip/requests
```

- **Method:**

```
GET
```

- **Params:**

```
trip: INTEGER
```

- **Body:**

```
No Body.
```

- **Response:**

```
[
    {
        user: INTEGER,
        name: STRING,
        image: STRING,
        institution: STRING,
        trip: INTEGER,
        added: BOOLEAN
    },
    ...
]
```

A.4.3 All Accepted Requests

Route to get all accepted Ride Request.

- **URL:**

```
/trips/:trip/requests/accepted
```

- **Method:**

```
GET
```

- **Params:**

```
trip: INTEGER
```

- **Body:**

```
No Body.
```

- **Response:**

```
[
    {
        user: INTEGER,
        name: STRING,
        image: STRING,
        institution: STRING,
        trip: INTEGER,
        added: BOOLEAN
    },
    ...
]
```

A.4.4 Update Value Request

Route to update the value of acceptance of the Ride Request.

- **URL:**

```
/trips/:trip/update
```

- **Method:**

```
PUT
```

- **Params:**

```
trip: INTEGER
```

- **Body:**

```
{
    id: INTEGER,
    value: BOOLEAN
}
```

- **Response:**

```
{
    trip: INTEGER,
    user: INTEGER,
    added: BOOLEAN
}
```

A.4.5 Itinerary Relative to User

Route to get another User's Itinerary with the User's requests.

- **URL:**

```
/trips/:rider/:requester
```

- **Method:**

```
GET
```

- **Params:**

```
rider: INTEGER
requester: INTEGER
```

- **Body:**

```
No Body.
```

- **Response:**

```
[
    {
        id: INTEGER,
        origin: STRING,
        destination: STRING,
        image: STRING,
        estimatedDuration: STRING,
        distance: STRING,
        price: STRING,
        departure: STRING,
        arrival: STRING,
        requested: BOOLEAN,
        added: BOOLEAN,
        car: INTEGER,
        privacy: STRING
    },
    ...
]
```

A.4.6 Delete Request

Route to delete a Request to the Trip.

- **URL:**

```
/trips/:trip/request/:id
```

- **Method:**

```
DELETE
```

- **Params:**

```
trip: INTEGER
id: INTEGER
```

- **Body:**

```
No Body.
```

- **Response:**

```
{
    message: 'done'
}
```

A.5 Trip Request

A.5.1 Search Trips

Route to get all the Trips filtered by the Search.

- **URL:**

```
/feed/:id/itinerary
```

- **Method:**

```
GET
```

- **Params:**

```
id: INTEGER
```

- **Body:**

```
No Body.
```

- **Response:**

```
[
    {
        id: INTEGER,
        origin: STRING,
        destination: STRING,
        image: STRING,
        estimatedDuration: STRING,
        distance: STRING,
        price: STRING,
```

```
            departure: STRING,
            arrival: STRING,
            requested: BOOLEAN,
            added: BOOLEAN,
            car: INTEGER,
            privacy: STRING
        },
        ...
    ]
```

A.5.2 Trip Request Information

Route to get the information of the Trip Request.

- **URL:**

```
/requirement/trip/:id
```

- **Method:**

```
GET
```

- **Params:**

```
id: INTEGER
```

- **Body:**

```
No Body.
```

- **Response:**

```
{
    id: INTEGER,
    origin: STRING,
    destination: STRING,
    image: STRING,
    distance: STRING,
    price: STRING,
    departure: STRING,
```

```
        privacy: STRING,
    }
```

A.5.3 All Trip Requests

Route to get All the Trip Requests of a User.

- **URL:**

```
    /requirement/:id
```

- **Method:**

```
    GET
```

- **Params:**

```
    id: INTEGER
```

- **Body:**

```
    No Body.
```

- **Response:**

```
    [
        {
            id: INTEGER,
            origin: STRING,
            destination: STRING,
            image: STRING,
            distance: STRING,
            price: STRING,
            departure: STRING,
            privacy: STRING,
        },
        ...
    ]
```

A.5.4 Create Trip Request

Route to create a Trip Request.

- **URL:**

```
/requirement/:id
```

- **Method:**

```
POST
```

- **Params:**

```
id: INTEGER
```

- **Body:**

```
{
    origin: STRING,
    destination: STRING,
    meanOfTransportation: STRING,
    distance: STRING,
    price: STRING,
    departure: STRING,
    privacy: STRING,
}
```

- **Response:**

```
{
    id: INTEGER,
    origin: STRING,
    destination: STRING,
    image: STRING,
    distance: STRING,
    price: STRING,
    departure: STRING,
    privacy: STRING,
}
```

A.5.5 Edit Trip Request

Route to edit a Trip Request.

- **URL:**

```
/requirement/:id/edit
```

- **Method:**

```
PUT
```

- **Params:**

```
id: INTEGER
```

- **Body:**

```
{
    origin: STRING,
    destination: STRING,
    meanOfTransportation: STRING,
    distance: STRING,
    price: STRING,
    departure: STRING,
    privacy: STRING,
}
```

- **Response:**

```
{
    id: INTEGER,
    origin: STRING,
    destination: STRING,
    image: STRING,
    distance: STRING,
    price: STRING,
    departure: STRING,
    privacy: STRING,
}
```

A.5.6 Delete Trip Request

Route to delete a Trip Request.

- **URL:**

```
/requirement/:id
```

- **Method:**

```
DELETE
```

- **Params:**

```
id: INTEGER
```

- **Body:**

```
No Body.
```

- **Response:**

```
{
    message: 'done'
}
```

A.6 Trip Request Responses

A.6.1 Search Requests

Route to get all the Requests filtered by the Search.

- **URL:**

```
/feed/:id/requirements
```

- **Method:**

```
GET
```

- **Params:**

```
id: INTEGER
```

- **Body:**

```
No Body.
```

- **Response:**

```
[
    {
        id: INTEGER,
        user: STRING,
        origin: STRING,
        destination: STRING,
        image: STRING,
        price: STRING,
        departure: STRING,
        privacy: STRING,
        requested: BOOLEAN,
    }
]
```

A.6.2 Responses Trip Request

Route to get all the responses to Trip Requests.

- **URL:**

```
/requirement/:tr/requests
```

- **Method:**

```
GET
```

- **Params:**

```
tr: INTEGER
```

- **Body:**

```
No Body.
```

- **Response:**

```
[
    {
        id: INTEGER,
        user: STRING,
        origin: STRING,
        destination: STRING,
        image: STRING,
        estimatedDuration: STRING,
        distance: STRING,
        price: STRING,
        departure: STRING,
        privacy: STRING,
        arrival: STRING
    },
    ...
]
```

A.6.3 Create Response Trip Request

Route to create a Response to a Trip Request.

- **URL:**

```
/requirement/:tr/:id/add
```

- **Method:**

```
POST
```

- **Params:**

```
    tr: INTEGER
    id: INTEGER
```

- **Body:**

```
    {
        trip: INTEGER
    }
```

- **Response:**

```
    {
        tripRequirement: INTEGER,
        trip: INTEGER,
        user: INTEGER
    }
```

A.6.4 Accept Response

Route to accept the response to the Trip Request.

- **URL:**

```
    /requirement/:tr/accept/:trip
```

- **Method:**

```
    POST
```

- **Params:**

```
    tr: INTEGER
    trip: INTEGER
```

- **Body:**

```
    No Body.
```

- **Response:**

```
{
    message: 'done'
}
```

A.6.5 All Responses

Route to get all responses to the Trip Request.

- **URL:**

```
/requirement/:requester/:rider
```

- **Method:**

```
GET
```

- **Params:**

```
requester: INTEGER
rider: INTEGER
```

- **Body:**

```
No Body.
```

- **Response:**

```
[
    {
        id: INTEGER,
        origin: STRING,
        destination: STRING,
        departure: STRING,
        image: STRING,
        price: STRING,
        privacy: STRING,
        requested: BOOLEAN,
```

```
        },
        ...
    ]
```

A.6.6 Delete Response

Route to delete a Response to a Trip Request.

- **URL:**

```
    /requirement/request/:tr/:id
```

- **Method:**

```
    DELETE
```

- **Params:**

```
    tr: INTEGER
    id: INTEGER
```

- **Body:**

```
    No Body.
```

- **Response:**

```
    {
        message: 'done'
    }
```

A.7 Cars

A.7.1 All Cars

Route to get all the Cars.

- **URL:**

```
/cars/:id
```

- **Method:**

```
GET
```

- **Params:**

```
id: INTEGER
```

- **Body:**

```
No Body.
```

- **Response:**

```
[
    {
        user: INTEGER,
        model: STRING,
        type: STRING,
        fuel: STRING,
        numSeats: INTEGER
    },
    ...
]
```

A.7.2 Create Car

Route to create a new Car.

- **URL:**

```
/cars/:id
```

- **Method:**

```
POST
```

- **Params:**

```
id: INTEGER
```

- **Body:**

```
{
    model: STRING,
    type: STRING,
    fuel: STRING,
    numSeats: INTEGER
}
```

- **Response:**

```
{
    user: INTEGER,
    model: STRING,
    type: STRING,
    fuel: STRING,
    numSeats: INTEGER
}
```

A.7.3 Delete Car

Route to delete the Car.

- **URL:**

```
/cars/:user_id/:car_id
```

- **Method:**

```
DELETE
```

- **Params:**

```
user_id: INTEGER
car_id: INTEGER
```

- **Body:**

```
No Body.
```

- **Response:**

```
{
    message: 'done'
}
```

A.8 Footprint

A.8.1 Places

Route to get the places with an input string.

- **URL:**

```
/places/:place
```

- **Method:**

```
GET
```

- **Params:**

```
place: STRING
```

- **Body:**

```
No Body.
```

- **Response:**

```
[
    {
        name: STRING,
```

```
            lat: STRING,
            long: STRING
        },
        ...
    ]
```

A.8.2 Calculate Trip Info

Route to calculate the trip information.

- **URL:**

```
    /places/calculate
```

- **Method:**

```
    POST
```

- **Params:**

```
    No Params.
```

- **Body:**

```
    {
        location1: {
            name: STRING,
            lat: STRING,
            long: STRING
        },
        location2: {
            name: STRING,
            lat: STRING,
            long: STRING
        }
    }
```

- **Response:**

```
    {
```

```
        origin: STRING,
        destination: STRING,
        dist: INTEGER,
        time: INTEGER
    }
```

A.8.3 Carbon Footprint

Route to get User Carbon Footprint.

- **URL:**

```
    /footprint/:id
```

- **Method:**

```
    GET
```

- **Params:**

```
    id: INTEGER
```

- **Body:**

```
    No Body.
```

- **Response:**

```
    {
        carbonFootPrint: FLOAT,
        carbonFootPrintWithoutSharedTrips: FLOAT,
        effectInCarbonFootprint: FLOAT
    }
```

A.8.4 Conference Carbon Footprint

Route to get the Conference Carbon Footprint.

- **URL:**

```
/footprint/conference
```

- **Method:**

```
GET
```

- **Params:**

```
No Params.
```

- **Body:**

```
No Body.
```

- **Response:**

```
{
    carbonFootPrint: FLOAT,
    carbonFootPrintWithoutSharedTrips: FLOAT,
    effectInCarbonFootprint: FLOAT,
    numberOfTrips: INTEGER
}
```

A.8.5 LeaderBoard

Route to get the User LeaderBoard with the Connections.

- **URL:**

```
/leaderboard/:id
```

- **Method:**

```
GET
```

- **Params:**

```
    id: INTEGER
```

- **Body:**

```
    No Body.
```

- **Response:**

```
    [
        {
            userid: INTGER,
            username: STRING,
            userimage: STRING,
            savedCO2: FLOAT,
            savedCO2perKm: FLOAT,
        }
    ]
```

Appendix B

Questionnaire

In this appendix is present the Questionnaire used in the test phase.

RideSharing App

In this questionnaire, it would be required that you have tried the Application and followed all the steps in the script.
After that, you can answer the questions below.

*Obrigatório

Interest in RideSharing

Before testing the Application, how do you assess your interest in RideSharing?

1. **How are you interested in Ridesharing? ***
   Marcar apenas uma oval por linha.

	1	2	3	4	5
In Offering Trips?					
In Requesting Rides?					
In Sharing your Itinerary?					

Simplicity

2. **How simple is to create ... ***
   Marcar apenas uma oval por linha.

	1	2	3	4	5
... a Trip					
... a Connection Request					
... a Trip Participation Request					
... a Trip Request					
... a Car					

Perceptibility

3. **How perceptible is to create ...** *
Marcar apenas uma oval por linha.

	1	2	3	4	5
... a Trip					
... a Connection Request					
... a Trip Participation Request					
... a Trip Request					
... a Car					

Informative Feedback

Information that you get after completing a task

4. **You get enough informative feedback when you create ...** *
Marcar apenas uma oval por linha.

	Yes	No
... a Trip		
... a Connection Request		
... a Trip Participation Request		
... a Trip Request		
... a Car		

Sequentialization

Taks sequetialization have a beginning, a middle, and an end

5. **Did you find the tasks to be sequentialized when you create ...** *
Marcar apenas uma oval por linha.

	Yes	No
... a Trip		
... a Connection Request		
... a Trip Participation Request		
... a Trip Request		
... a Car		

Error Handling

Handling of the errors by the application

6. **Did you find your errors perceptible when you create ... ***
Marcar apenas uma oval por linha.

	Yes	No
... a Trip		
... a Connection Request		
... a Trip Participation Request		
... a Trip Request		
... a Car		

7. **Did you find your errors reversible when you create ... ***
Marcar apenas uma oval por linha.

	Yes	No
... a Trip		
... a Connection Request		
... a Trip Participation Request		
... a Trip Request		
... a Car		

Design

8. **How would you rate the design of the application in terms of ... ***
Marcar apenas uma oval por linha.

	1	2	3	4	5
... Simplicity					
... Overall Appearence					
... Color Selection					

9. **What would you change in the design of the application?**

Features

10. **What is the Feature that you found to be the most valuable?**

11. **What is the Feature that you found to be the least necessary?**

12. **What is the Feature that you found to be missing?**

Interest in RideSharing

After testing the Application how assess your interest in RideSharing?

13. **How is now your interest in Ridesharing?** *
Marcar apenas uma oval por linha.

	1	2	3	4	5
In Offering Trips?					
In Requesting Rides?					
In Sharing your Itinerary?					