Politecnico di Milano

DIGITAL INNOVATION LAB

FINAL DELIVERY

PUBLIC TRASPORTATION

Disabled people: a challenging discussion

***Professor*:**

Marco Brambilla

***Authors*:**

Omar Abdrabou
Simone Bianchera
Leonardo Mandruzzato
Cristiano Serafini

Sommario

1. Introduction

1.1 Context

Different people use public transportation (PT) daily because, especially in the big cities, it represents the best solution for moving around, saving time and money, and possibly reducing pollution.

PT is part of the daily life of many people. For this reason, it is always a focus of debate about possible innovations and improvements. Unfortunately, in Italian's big cities (e.g., Milan, Turin, and Rome), public transportation is also often a source of complaints regarding the intra-city movement due to the accessibility, availability, and timetable precision of the fleet.

1.2 Instantiation details

The project focuses its attention on the northern Italian regions for many reasons. Since the whole group had the opportunity to experience daily the functioning of this specific public transportation system, we will have a more detailed overview of the problem.

Going a step further in the instantiation phase, let's specify the cities the project will consider. The targets are Milan because it is the city we are studying in and, thus, we will be honored to improve its public service. And then Turin, because it already has a platform that may represent a starting point for the project.

Problem Description

In the last few years, the topic concerning the accessibility of the PT for disabled people has become central in the public and political discussion. The project aims to improve the quality of the information provided to these subjects and, consequently, their public transportation approachability.

Today, transport companies and municipalities are making great efforts to improve the accessibility for people with a handicap from a structural standpoint. All these efforts can lose their meaning if not supported by the right information system. Without specific information, disabled people can incur different problems.

For example, in those cities where not all the buses of a line are equipped with a wheelchair access platform *(figure 1)*, it could be possible that a disabled person who is waiting for a bus notices that she/he cannot physically enter only once the bus is in front of her/him. This unpleasant situation forces the customer to wait for the following vehicle.

Another issue that reduces the service's efficiency is the lack of information about the possible occupation of the disabled reserved seats *(figure 2)*. Again, if a person in a wheelchair would discover that her/his "assigned" place is already taken from others, she/he would be obliged to get on the following bus.

Both the two use cases do not only depend on structural limits but also informational ones. The service perceived by the disabled customers would be much better if there would be a way to know in advance if the buses that will pass will be suitable for wheelchair access and whether the seats reserved for them are or not occupied.

Apart from this specific information regarding the sphere of tools related to people with disabilities, also other kinds of info can be useful to these subjects and sensitive individuals in general. For example, the number of people already present on a specific vehicle is a piece of data not provided to the public. Claustrophobic people, pregnant women, people with mobility difficulties (e.g., disabled people and elderly) would benefit from a platform that distributes this data in advance.

To conclude, the project goal is to fill this lack of information and make them available in the best and more intuitive way possible.

Figure 1: this picture, taken in Milan, shows the platform that most trams have to pick up disabled passengers.

Figure 2: this photo, taken in Turin, shows the place reserved for people who have to stay in a wheelchair.

1.3 Glossary

- PT: Public Transportations
- CRM: Customer Relationship Management
- RS: Recommender System
- Hand: Name of the application
- UI: User Interface
- UX: User Experience
- IFML:
- BPMN: Business Modeling Process Notation
- DB: Data Base
- REST: REpresentational State Transfer
- RASD: Requirement Analysis & Specification Document
- DD: Design Document
- API: Application Programing Interface
- CXJM: Customer Experience Journey Mapping
- BU: Business Unit
- ER: Entity Relational

1.4 Stakeholders

Firstly, the main stakeholders (i.e., the final users) are the people suffering from disabilities, as they would be interested in our proposition to move around daily in an easier way.

Secondly, the various public transportation entities may wish to collaborate on this platform to create a more functional offer to their customers. The project could improve their services, but, at the same time, it would require necessary information in their possession. Thus, there could be a synergy between these companies and the team behind the platform. Another possible idea may be to sell them a sort of turnkey solution. In this way, they would not share reserved information with an external entity, but still, they would benefit from the functionalities the service offers.

Finally, the public and private entities that offer assistance to the disabled would also benefit from the info the project aims to provide.

We have identified three major groups of stakeholders, based on the reasons they are interested in the designed offer:

- Direct use of the product to improve the quality of life of the affected people
- Providers of the information required
- Introducing the services to the assisted persons and using them to better the assistance they already offer

Stakeholders Group	Stakeholders Subgroup	Motive of interest
People affected with disabilities, their partners, and families.	People inconvenienced from disabilities.	Direct use of the possible solution to solve some of the issues related to public transportation.
	Immediate families of disabled people.	Use of the possible solution as a means to provide better help in everyday transportation with public means.
	Partners of disabled people.	
Public entities offering transportations means.	Train companies	Implementation of a few sensors and sharing of some detail about the vehicles to permit the gathering of the necessary information.
	Bus companies	
	Streetcar companies	
Public and private entities offering some type of assistance to people with disabilities.	Public, no-profit associations	Acting as an intermediary to introduce the product to the assisted people and usage of it in some of their activities.
Other types of entities which include people affected with disabilities in their personnel.	Local companies	Relative usage as helper in some internal events.
	Schools	
	Entertainment activities	

1.5 References

- "Come risolvere le difficoltà di trasporto disabili a Milano" – Autoability https://www.autoability.it/trasporto-disabili-a-milano-2
- "l trasporto dei disabili tra servizio pubblico e servizio privato" – GTP Consorzio Trasporti alternativi https://www.trasportodisabiligtp.it/il-trasporto-dei-disabili-tra-servizio-pubblico-e-servizio-privato/
- "La disabilità e il trasporto pubblico locale" – Giampaolo Lavezzo http://www.superando.it/2016/01/28/la-disabilita-e-il-trasporto-pubblico-locale/
- "Il disabile deve poter accedere liberamente al trasporto pubblico" – Lucia Izzo https://www.studiocataldi.it/articoli/27595-il-disabile-deve-poter-accedere-liberamente-al-trasporto-pubblico.asp
- "La mobilità e il trasporto secondo la legge 104/92" – Disabili.com https://www.disabili.com/legge-e-fisco/speciali-legge-a-fisco/legge-104-disabili/22431-legge-104-la-mobilita-e-i-trasporti

- “Autobus, treni e trasporti più semplici per persone con disabilità?” – Disabili.com https://www.disabili.com/mobilita-auto/articoli-mobilita-a-auto/autobus-treni-e-trasporti-pubblici-locali-piu-semplici-per-le-persone-con-disabilita

2. Personas

With the personas, we are trying to figure out the main final targets of the project and the principal entities involved in the functioning of the application.

2.1 Persona radar

We designed the persona radar based on the focus the project will have on the stakeholders chosen. The solutions conceived will focus mainly on people in wheelchairs but who can still lead a fairly independent life (e.g., people who have lost the use of their legs). Furthermore, the project will take care also of other sensible subjects as people with walking difficulties (e.g., people with crutches) and pregnant women. Also, relatives and, in general, people who look after a disabled person will benefit from the solution the project aims to propose. A relevant persona is the driver since it will have an active role. The project does not provide a service for the driver but rather charges the latter for some tasks fundamental for the functioning of the application.

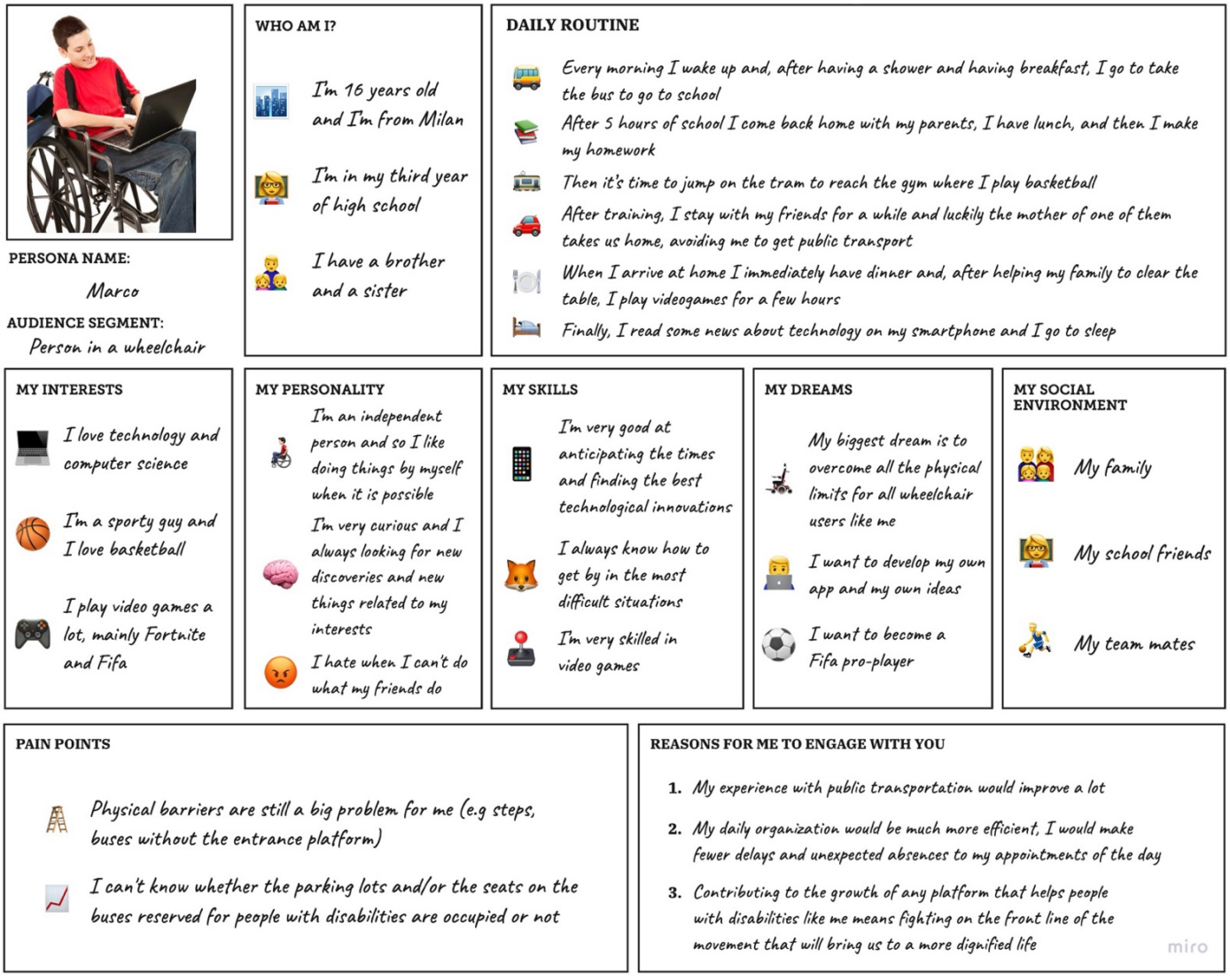
2.2 *Primary persona – Person in a wheelchair*

As anticipated in the section related to the radar, the focus will be on those people in a wheelchair who can lead a pretty independent life. Generally, they do not need accompaniers and are used to move around the city where they live autonomously. Since the disability they have is irreversible (or almost impossible to cure), they will often incur specific and recurring problems. For example, physical barriers are a big issue, and for this reason, they are one of the pain points for such subjects. In addition to physical barriers, they often lack information. For example, they will be late to appointments, not because of their poor organization, but because they cannot know if someone has already occupied the parking lot reserved for them. The same thing can happen when their movements are not made by car, but by public transport. It may happen that they cannot jump on a bus because there is no more physical space for their wheelchair (i.e., the reserved seat/place is not more available), messing around their daily plan.

With our solution, their experience with PT would improve a lot. Consequently, also the organizational capabilities of the subjects will exploit the platform would do the same.

Finally, the project targets should be available to adopt new technologies to improve their lives in the foreground and possibly that of others with the same difficulties.

For the reasons listed above, the target should be like Marco.



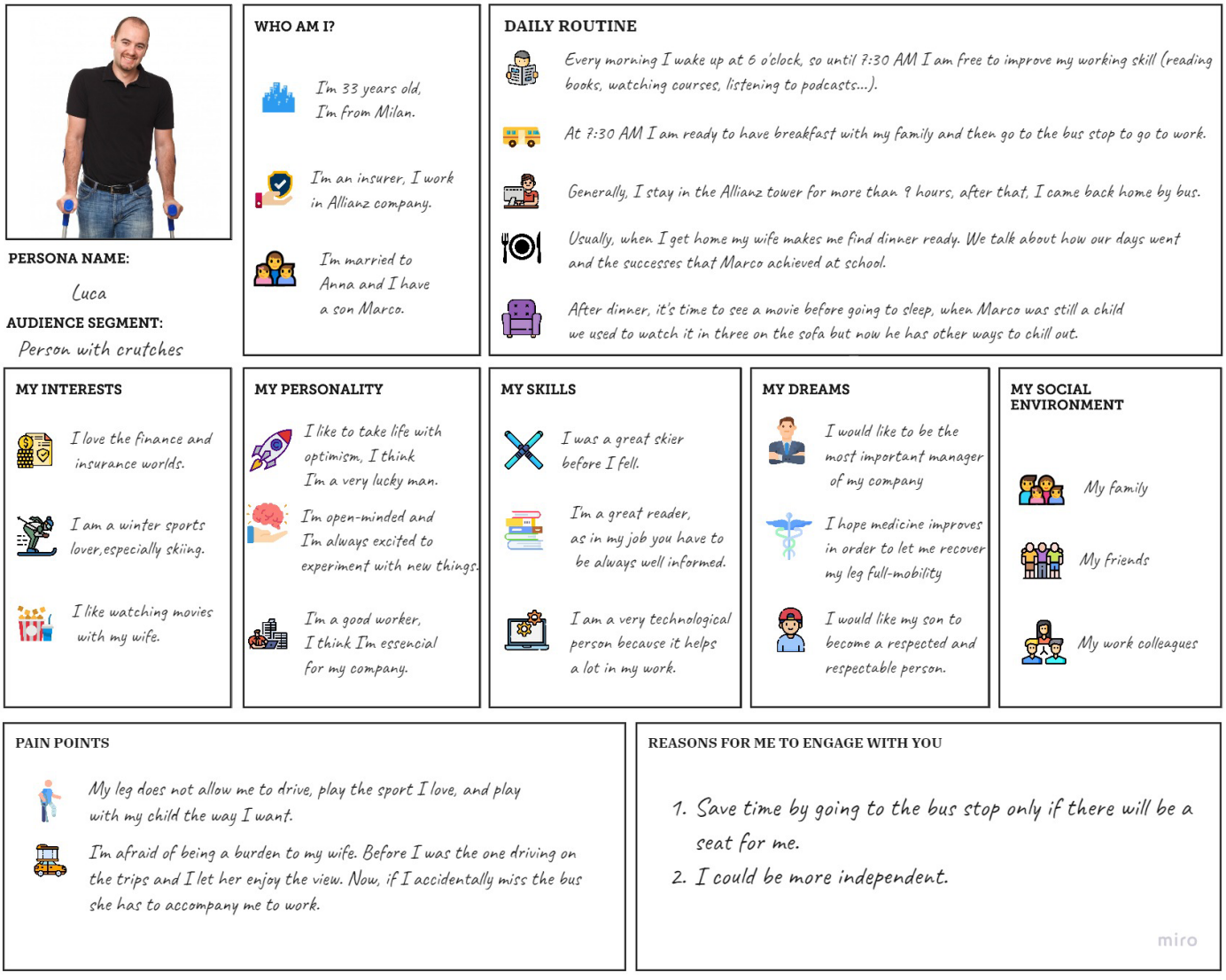
2.3 Secondary personas

In this paragraph, the secondary characters are reported and described. They are not primary maybe because they are not directly affected by the problem, or they will not make use of our application for a long time.

2.3.1 Person with crutches

The person with crutches or, more generally, people who have some minor injury suffer from the problem that we would like to solve in our project.

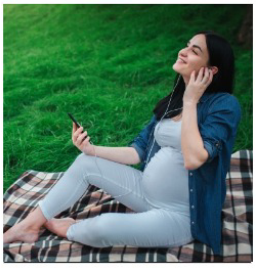
The personas below show Luca, a businessman who seriously injured his leg while skiing. Luca is forced to take the bus every morning to go to work and he never knows if there will be a seat for him.



2.3.2 Pregnant woman

The pregnant woman is one of the subjects interested in the problem raised in the previous document. Knowing if there will be her reserved place in public transportation could be helpful to manage the difficulty of the pregnancy.

She will be a temporary user due to her transitory condition, but she could be also a way to reach other mothers and they will be always present in society.



PERSONA NAME:

Claudia

AUDIENCE SEGMENT:

Pregnant woman

WHO AM I?

- I'm 31-years-old and I live in Redecesio, near Milano
- I'm director's assistant in "Principe di Savoia", in the center of Mlan
- I'm pregnant with two incoming babies, 5-months pregnancy

DAILY ROUTINE

- I wake up at 7 and I make breakfast with a Kinder Bueno and a coffee
- I take the bus in the front of my house and then travel to Lambrate. Then I use the tram to reach my work place
- After 8 hours of solving problems and adjusting accounting, finally I can return home
- Returned home, I chill out with my boyfriend and my dog
- Before dinner I walk out in the neighbourhood
- After dinner I usually read books or listen music of new unknown groups

MY INTERESTS

- Reading books
- Listening to music
- Walking in the neighbourhood with my dog

MY PERSONALITY

- I love planning all the things in my life
- I'm very shy when I don't know people
- I love travelling and knowing new places and new cultures

MY SKILLS

- Good at math
- Good money management
- Fast in answering the messages on the phone
- Optimum experience in planning and organizing

MY DREAMS

- Having a girl and a boy
- Becoming the director of the hotel
- Buy a villa

MY SOCIAL ENVIRONMENT

- Family
- Work-mates
- People in the neighbourhood with dogs

PAIN POINTS

- My back hurts if I stay too much time standing up
- I often get tired due to the pregnancy, so I must rest even for a short time

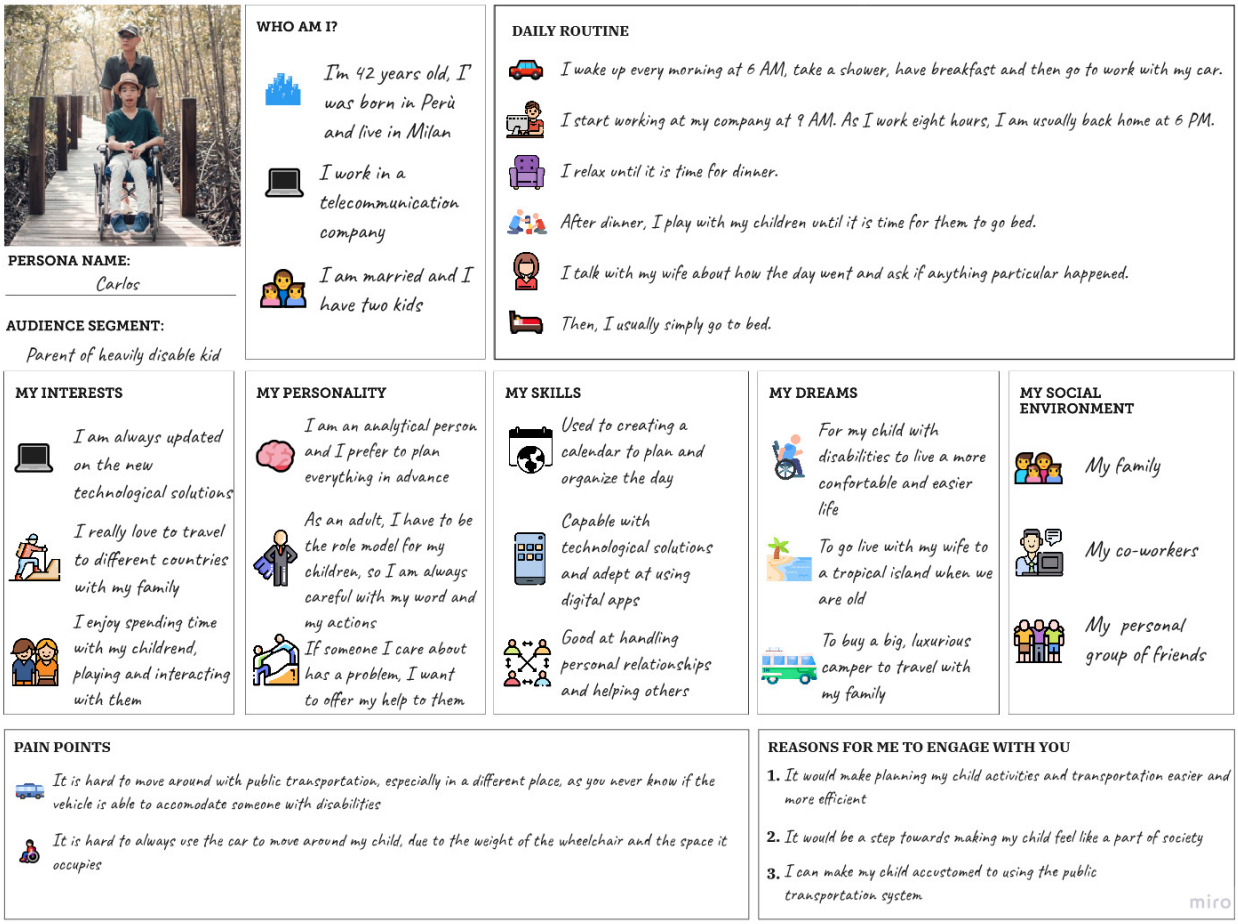
REASONS FOR ME TO ENGAGE WITH YOU

1. It would be easier planning my movements
2. It would be easier knowing when I can leave my work in order to take a seat returing home
3. I could manage unexpected transfers with no fear of standing up all time and arriving without remaining energy

2.3.3 Parent of heavily disable kid

The third secondary persona we chose is a hypothetical parent that is taking care of his child affected with disabilities.
The reason we wanted to bring focus on this segment is due to the high amount of population that can be considered part of this group, from close relatives to workers that are tasked with taking care of these kinds of people.

As such, our effort in designing the landscape of our solution will also take into consideration people in these circumstances.

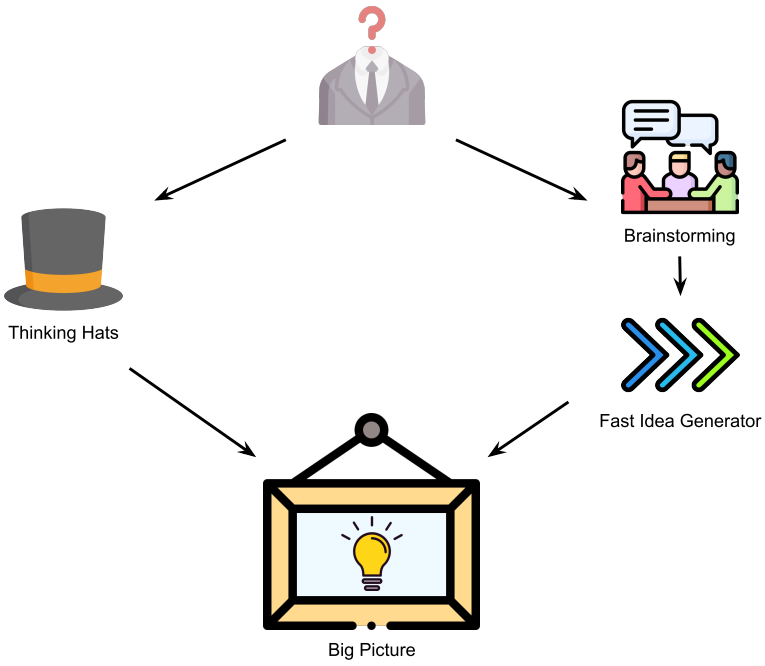


3. Idea Generation

In this macro-section, we will explain in detail how we come up with some of the ideas we had and how they evolved during our meetings. In general, we used three main approaches/tools in parallel to generate and discuss about them:

1. Thinking hats
2. Brainstorming
3. Fast Idea Generator

Then, we grouped and reviewed the ideas to obtain a more orderly and precise general picture to analyze and evaluate in the following steps.



3.1 Thinking Hats

3.1.1 Approach description

The problems discussed during our sessions regard the circumstances surrounding the relationship between people affected with physical disabilities and the usage of the public transportation system.

To have a complete view of the situation, we gathered information and data (from the internet and directly from a concerned individual) to deeply delve into the matter and distinguish the issues that people with disabilities are facing and analyze their impact.

By using the data and the *Personas* we previously created as a basis, we were able to define four different situations to be discussed:

1. Retrieving information on the reserved place for wheelchairs.
2. Better handling reserved seats occupation.
3. Calculating the current number of people on transportation means.
4. Additional features:
    - Showing the transportation means' real-time position.
    - Adding a reporting system for the stops.
    - Showing additional information regarding the access on the vehicle.

Our strategy in analyzing these four focal points made use of the Thinking Hats methodology, allocating four rounds of discussion to each of them.

In the first two rounds, all the hats, apart from yellow and black, were assigned, forcing us to have multiple standpoints when discussing the topics, looking at them from different angles.

In the third round, we all impersonated the yellow hat, while in the fourth we all used the black one.

As such, we divided into two distinct phases the work done:

1. During the first two discussion sessions, ideas were formed and discussed from different points of view.
2. Then, the advantages and disadvantages of each proposal were identified, to determine their validity.

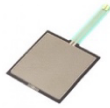
This modus operandi will be reported in the figures below, where the hats used in the first two sessions are placed on the left and the flow of thought is numbered. While the ideas from the last two sessions are on the right side, not numbered because we think of it like a pro and cons of the various ideas.

3.1.2 Retrieving information on the reserved place for wheelchairs

The first possible solution implies weight sensors on the platform. Since a person in a wheelchair must use the latter to get on the bus, this could be the element to exploit. Some issues can emerge when the places for wheelchairs in the bus are more than one. It would be impossible to keep the count of how many disabled people are actually in the vehicle.

The second idea refers to a specific set of operations everyone in a wheelchair should do to enter on a bus. Unfortunately, an informant who leads this kind of life in person told us that no one usually follows this procedure precisely.

The last solution, the most shared in the group, involves the bus driver. Among her/his tasks, she/he must also take care of the entrance and the exit of each disabled person (i.e., activating/extracting the platform and helping the concerned person with some specific logistic moves). Thus, she/he would not have problems to report an entrance/exit of a customer in a wheelchair. On the internet there are many articles highlighting that drivers are not particularly stressed and in fact there are many cases of people on the phone driving. So, a new task to perform would not have an impact that could compromise their work.

Possible Solution	Creative	Process	Facts	Feelings	Cautions	Benefits
 Weight sensors on the platform	2. Weight sensor on the platforms		1. To enter a disable must use the platform 4. Some buses have two places reserved for the wheelchairs' customers	3. Impossible to distinguish between an entrance and an exit if there is more than one person with the wheelchair on the bus	• This solution would not be accurate at all	• This solution would be pretty automatic
 Exploit the steps of the entrance procedure	6. Exploit the actions performed during the procedure (seat belt, button pressing, etc.)	8. We need to understand if this procedure is actually followed	5. People in a wheelchair should follow a procedure to get on the bus 9. Nobody actually follows the procedure	7. It could be a procedure that nobody really follows	• Putting sensors on platforms, seat belts, and/or buttons would have big costs (installation, testing, maintenance) • The results with sensors would be very approximative • Sensors would require strict adherence to the rules (the routine)	• Using this solution combined with the previous one (double check) would completely automatize the functioning
 Counter function performed by the bus driver	10. Driver with a counter function	11. Too much impact on the responsibilities of the bus driver	12. Drivers do not have lots of tasks apart from driving		• Delegating the solution to the drivers requires their availability to assume more responsibilities and carry out more tasks	• The solution delegated to the driver is the most accurate • Delegating the task to the driver do not have costs apart from the development of the interface used by her/him to report an entrance or an exit

3.1.3 Better handling reserved seats occupation

The following sub-problem is about the reserved seat for people with temporal injuries, or in general without the need for a wheelchair and pregnant women.

The table is organized into 4 main possible solutions:

- Cameras linked to a neural network.
- Mini gates in reserved seats' proximity.
- A button that informs the presence of the category of people written above.
- An app with which the driver will report the current situation as concern those seats.

In particular, during the discussion about the second possible solution, a crucial issue emerged: informing the user of a possible change in public transportation becomes critical and expensive (e.g., through a communication campaign). Furthermore, distributing the cards that unlock these mini gates and building the infrastructure that checks whether the user has any disabilities that make her/him eligible to use the concerned seats could create difficulties in implementing such an idea.

Possible Solution	Creative	Process	Facts	Feelings	Cautions	Benefits
 Camera with neural network	1. Cameras linked to a neural network to distinguish people	4. How to convince the company to give us the access to the cameras?	2. Cameras are already on the vehicle	3. Not good for privacy	• In-house development and testing (no open solutions in the market) • A huge dataset must be found • We must have the access to the cameras • Privacy issues • We don't know the positions of the cameras and their cover view of the bus	• Common infrastructures for other sub-problems • Infrastructure already exists • More precision than the actual implemented system
 Mini-gate for reserved seats	5. Mini-gate to block reserved seats, only unlockable with a card 8. Not all reserved seats, only a partial number 10. To let the people adapt to the new solution 14. The verify is done with video autenthication and medical documents	7. We have to convince the company to adopt only our idea 11. A communication campaign must be organized 13. How to validate the card for temporal users?	12. It is needed both an app and a physical card, so also the elders can use it	6. Blocking a seat is not a good idea and it seems expensive 9. What is the gain blocking only a partial number? 15. Too many users' sensitive data	• Need of control infrastructures • Additional maintenance cost • Worse user-experience • Reduction of total all-available seats • People must be advised of this solution	• Seats become totally reserved • It can be used also to count specific categories of people • Easy physical implementation
 Button for presence	16. There are buttons through this category of people can advise of their presence 20. Having the driver near the buttons, he/she must check them	17. Where must the buttons be placed? 22. Users must be informed	18. If the buttons would be placed far by the driver, every guy will touch them	19. The affected people will enter in the first door, near the reserved seats 21. The driver won't check them	• The driver can't check them in any situations • Not user-oriented • Reliability given to the users	• Easy implementation • The driver's responsability doesn't increase a lot
 App for presence	23. Driver must report their presence through an app	26. A training to the drivers must be provided	24. Giving more responsibility, their salary will grow	25. It will take times during the stops, increasing the delay	• More responsibility to the driver • If the bus is full, he/she couldn't be able to control • The same problem if a lot of people enter in the bus • Possible additional delay	• The user experience doesn't change • It will fit with other possible solutions in the remaining sub-problems • Counting target people precise enough

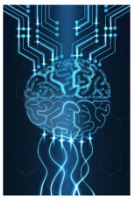
3.1.4 Calculating the current number of people on transportation means

Four possible solutions emerged from the discussion. The first concerns the use of photocells to count people. It would be a more expensive, but also a more precise solution than the one already in place (AMI Ferrara). The sensors should be placed near the doors, perhaps protected by some wrapping to avoid vandalism.

The second is very similar to the one above but has a less accurate estimate and lower price. This type of sensor will be placed on the vehicle ceiling, which is more difficult to vandalize.

The last two ideas are also useful for solving the above problem (Retrieving wheelchair seat information, better management of reserved seat occupancy). The one relating to the driver in charge of counting the people in this problem is the least precise because it is based on the subjectivity of the driver herself. As for the machine learning solution, it is not that difficult to implement to solve this problem. Basically, you should implement an algorithm that, given a photo, counts the people in it.

The image below shows the flow behind all these considerations.

Possible Solution	Creative	Process	Facts	Feelings	Cautions	Benefits
 People counter photocell	1 A photocell could be placed at each entrance to the bus or tram to count people entering and exiting.	4 How much would the photocell cost?	2 A similar solution is the one used by AMI Ferrara, devices with internet access in means of transport are counted. 5 Their cost is around 150/200 euros.	3 Surely there would be a better person count than the current one.	• Quite an expensive solution. • Need for additional and periodic maintenance. • The sensors may not work properly if the vehicle is full. • Easy to break by vandalizing.	• Accurate estimation of the number of people inside. • Simple installation. • The user experience is not affected as the user does not have to provide feedback. • Available on the market.
 Proximity sensor	6 It is possible to use proximity sensors placed on the ceiling to have an estimation of the passengers inside.	7 How much would these sensor cost?	8 The cost of these sensors is around 60 euros.	9 Should be taken into consideration people's build.	• We have an indicative, not precise estimation. • It should be understood in which places put the sensors and in which quantity.	• Simple installation. • Automatic system. • The user experience is not affected as the user does not have to provide feedback. • Available on the market.
 Driver in charge of counting the people	10 We could leave it to the driver to determine if full or not.	12 Should we raise drivers' salaries?	13 Estimation is influenced by subjectivity. 14 People entering or leaving are not considered.	11 The driver who has been doing his job for some time will know immediately if the bus is full or not.	• Very subjective solution, the estimate is left to the driver. • It could be an extra burden that the driver has to deal with instead of just focusing on driving.	• Economical hardware solution. • The user experience is not affected as the user does not have to provide feedback.
 Machine learning algorithm for people counting	15 A machine-learning algorithm based on the photos taken by the cameras inside the vehicle could be used to count people.	17 Adding this new technology to our core resources could benefit us greatly. 18 Better a make or buy solution?	19 Almost all of the company's transportation vehicles have built-in cameras.	16 Wow, super technological means of transport.	• Solution development costs (not open solution found). • The company must allow us access to the cameras (privacy issues). • We should understand if the cameras already installed are positioned favorably for our needs or if we need to install others.	• We could use the same infrastructure to solve the other problems of our project as well. • Certainly more accurate than current methods.

3.1.5 Additional features

During our analysis, three optional functionalities that would improve the user experience were identified.

The first feature would show to users the position of vehicles on the map in real-time, by gathering data from a GPS that is installed on each one of them.

The second function is about the creation of a system that would allow users to interact with it in more ways, for example:

- To report the presence of physical barriers that would impede some people with disabilities to get onboard.
- To give a rating to the stops at different times, so that users can see if, for example, there are too many people to get onboard at a specific stop.

The last function regards showing users information about the public vehicles, like if a certain bus or tram line has elevation platforms to accommodate wheelchairs users or not.

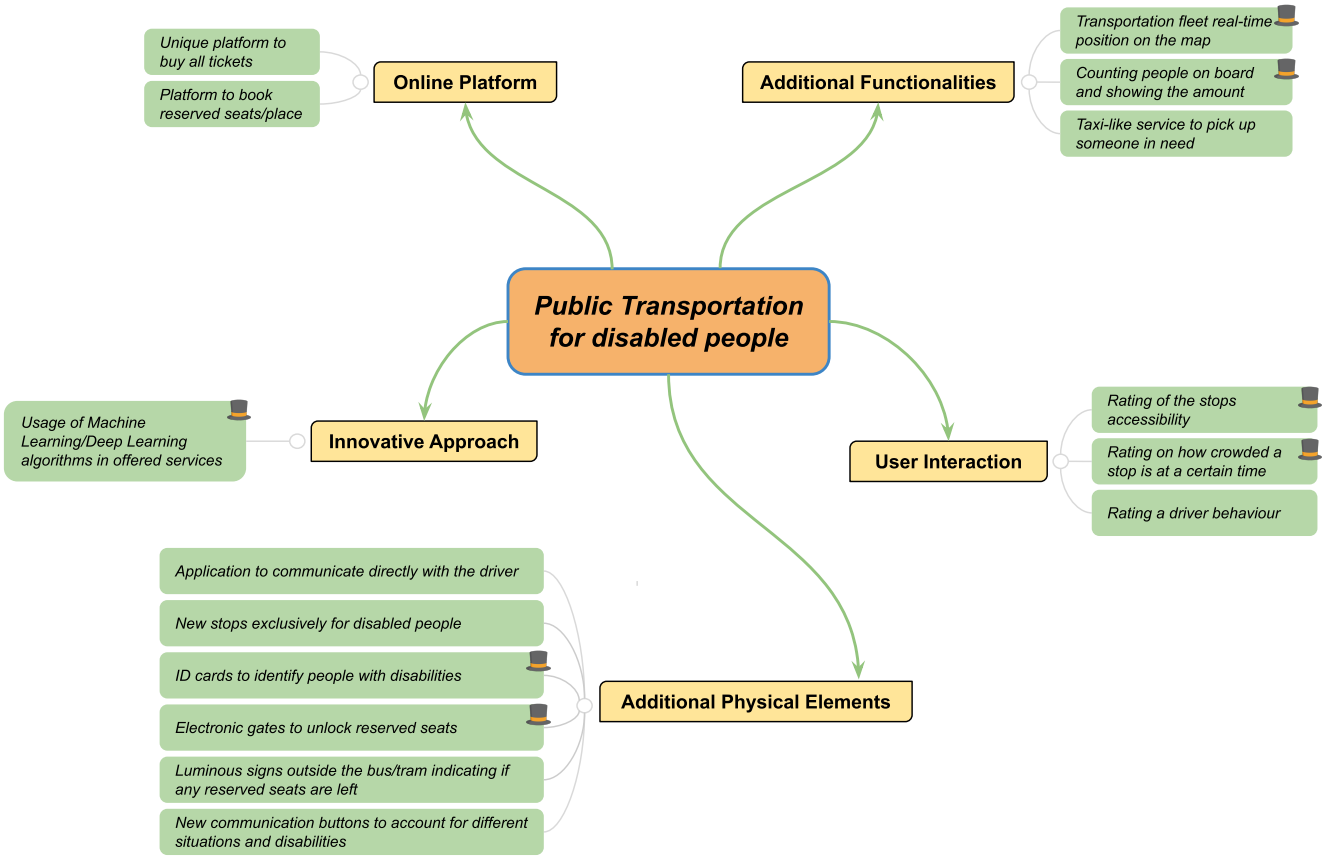
It would work very well together with the real-time tracking, providing users an engaging interface that would require no effort to be understood and used properly.

Possible Solution	Creative	Process	Facts	Feelings	Cautions	Benefits
 Transportation means real-time tracking	1. Transportation means' position broadcasted in real-time with GPS on a map.	4. Why don't we use the already existing funcion?	3. Turin has already implemented this feature.	2. Really useful function.	• Not a necessary function, only secondary in importance.	• Really useful for planning one's movements. • The interface would show information that is immediately useful and be more engaging. • GPS are already instanlled on the majority of transportation means. • Easy and intuitive function to use.
 Reporting system	5. Add a system to report if a stop has physical barriers that are an issue for people with disabilites or if there are too many people at a certain time to get onboard.	6. How would you handle the rating? 8. Would we need to make deals also with the Municipalities?	7. Some companies have a system that is similar to this one.	5. We should add a rating	• Not a necessary function, only secondary in importance. • The power would be all in the hands of the users, who could give bad rating on purposes. • Bad ratings could tarnish the reputation of the authorities we work with, making them less prone to collaborate with us.	• It would make choosing a stop that accomodates one's special needs possible, by consulting this information. • If a stop has bad rating, it could be a signal to make the compent authorities fix it.
 Vehicles additional information	9. Show information about the vehicles to ease access.		10. In Turin, if you access Google Maps, this information is already available.	11. It does not sound that useful	• Not a necessary function, only secondary in importance.	• Really useful for planning one's movements. • The interface would show information that is immediately useful and be more engaging. • Easy and intuitive function to use.

3.2 Brainstorming

The brainstorming aims to generate new ideas regarding the possible solutions to the problems the project is facing. Since the group had already had the *thinking hats* session, during these meetings, we have tried to propose different ideas from the ones previously obtained. In a nutshell, the brainstorming goal was to generate new proposals to add to the ones already in our hands.

3.2.1 Ideas



3.2.2 Feedback

We initially named a facilitator to run the brainstorming. Once chosen, he introduced the various issues the project aims to solve in the most comprehensive way possible. The main problem was divided into many sub-parts to conduct a well-structured discussion. Starting from the issues related to the primary persona (i.e., the wheelchair user), the facilitator captured all the proposals made by the 4 of us. We then moved toward the radar edges analyzing all the issues related to the secondary/tertiary personas.

Since the group had already got to generate new ideas project-related (i.e., the *thinking hats* sessions) and widely discussed some possible solutions, carrying on the meeting was easy for the facilitator due to a shared understanding of all the situations.

Everything went smooth since everybody constructively approached the brainstorming, accepting all the ideas coming from the others with no judgments and trying to add something to them, if necessary and possible.

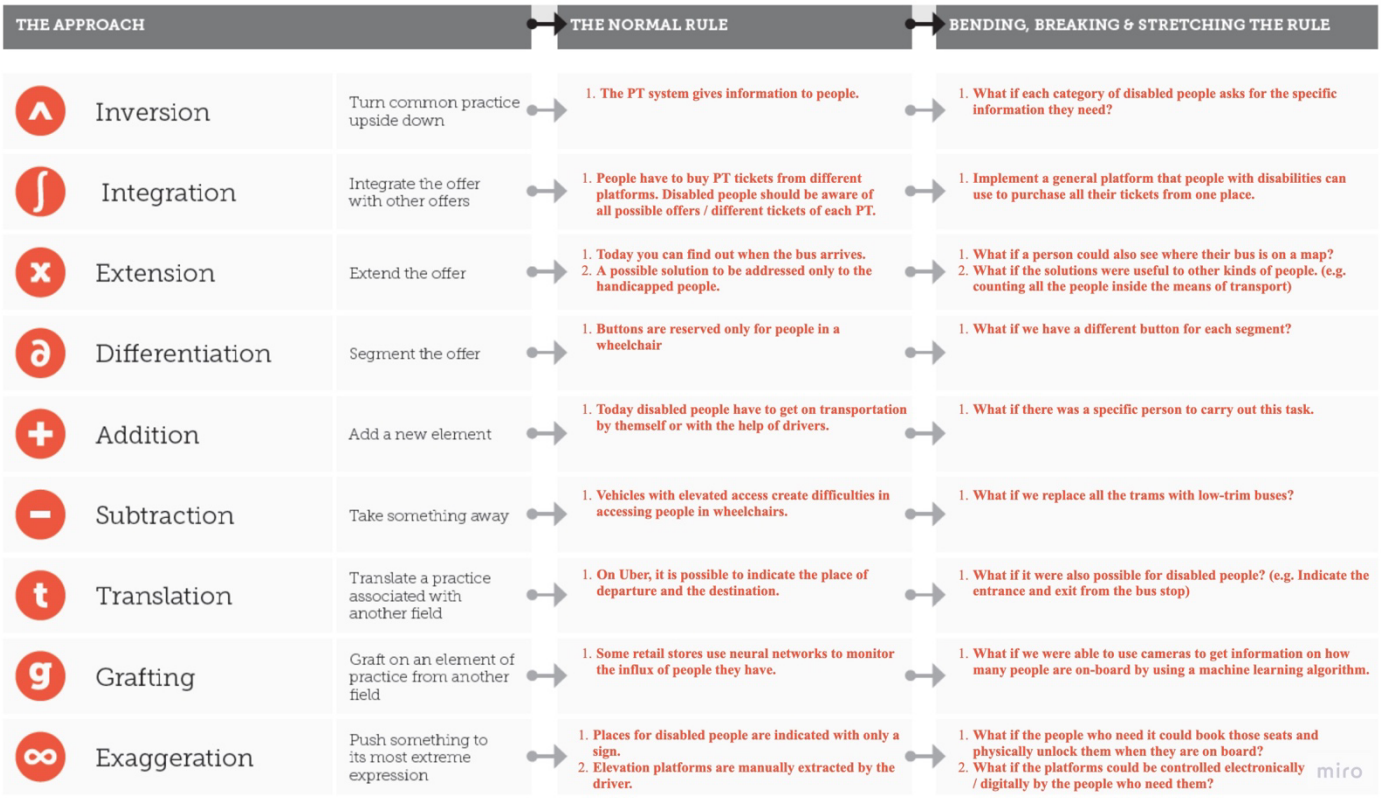
We almost did not know each other before this experience, and the first calls were a little bit cumbersome. The comprehension of the way of thinking of the other members is improving meeting after meeting. With this comprehension, also productivity is bettering, and we can now converge to shared ideas faster and faster.

Each of us has his way of running discussions and approaching new topics. We consider this an added value for this kind of project since it gives different flavors to what we create.

In conclusion, we are satisfied with how we carried out the meetings until now, and we are also proud and propositive with the ideas that came from them.

3.3 *Fast Idea Generator*

In order to generate even more ideas, the ones that emerged from the brainstorming have been stretched exploiting the *Fast Idea Generator* tool.



3.4 *Idea Grouping & Review*

Several ideas emerged from our discussions. We were able to group them into three possible macro solutions:

1) Machine Learning and Deep Learning solution
2) Driver Mobile Application solution
3) Physical solution: the introduction of photocells and proximity sensors for counting people, mini gate for closing reserved seats, and weight sensors on the platform for counting people in a wheelchair.

Making a careful review only the last solution was discarded as too expensive and imprecise.

The other two were worthy of consideration. In particular, the first has attracted our attention as it is based on an increasingly widespread technology, a solution that is certainly expensive but enough efficient, adopting this solution could be an investment for the future.

The second instead presents a more standard solution where the task is entrusted to the driver, certainly less expensive and now more efficient than the rival. We will analyze these two solutions in more detail in the SWOT analysis.

4. Idea Evaluation

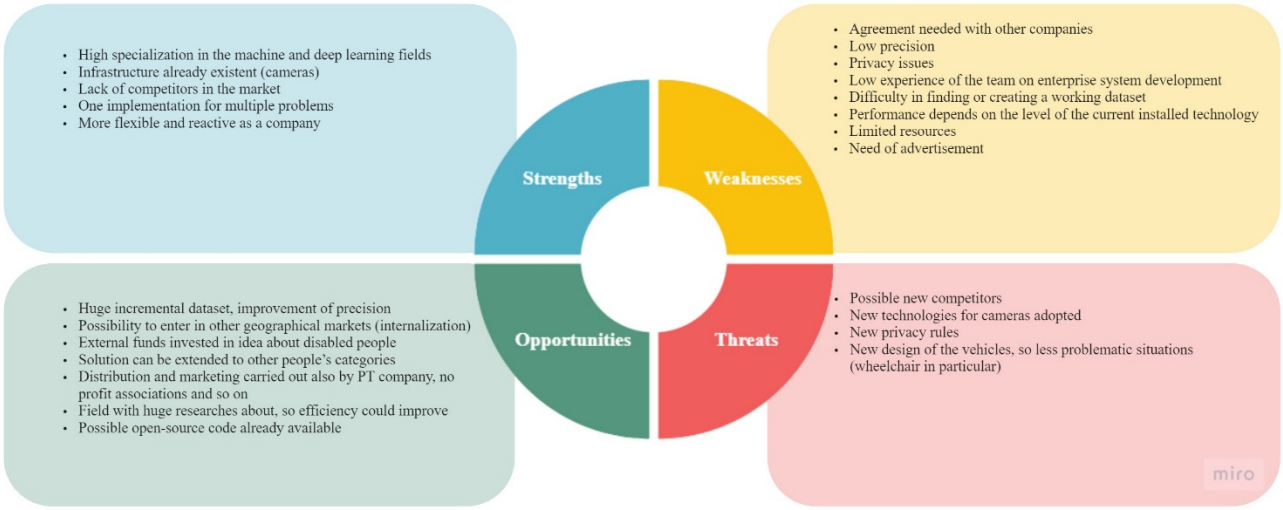
4.1 SWOT analysis

The SWOT analysis shown below was carried out assuming that our standpoint is that of a startup. Then they highlight Strengths, Weaknesses, Opportunities, and Threats related both to the business model of our organization and to the proposed solution itself.

4.1.1 Machine and Deep Learning SWOT

This first SWOT analysis deals with the solution related to the implementation of a deep learning / machine learning algorithm. With today's technology, it does not have a high accuracy (e.g., the model can classify an overweight woman as a pregnant one).

This solution has a particular weakness: the transport company must permit us access to their cameras. Moreover, the image feed received should be of sufficient quality to reach the best precision possible. As such, this solution would be expensive from the point of view of the company that decides to adopt it, and from the developing team side, that should be very prepared to provide an excellent algorithm with little funds. Bearing in mind that we are a start-up and not an already known company on the market.

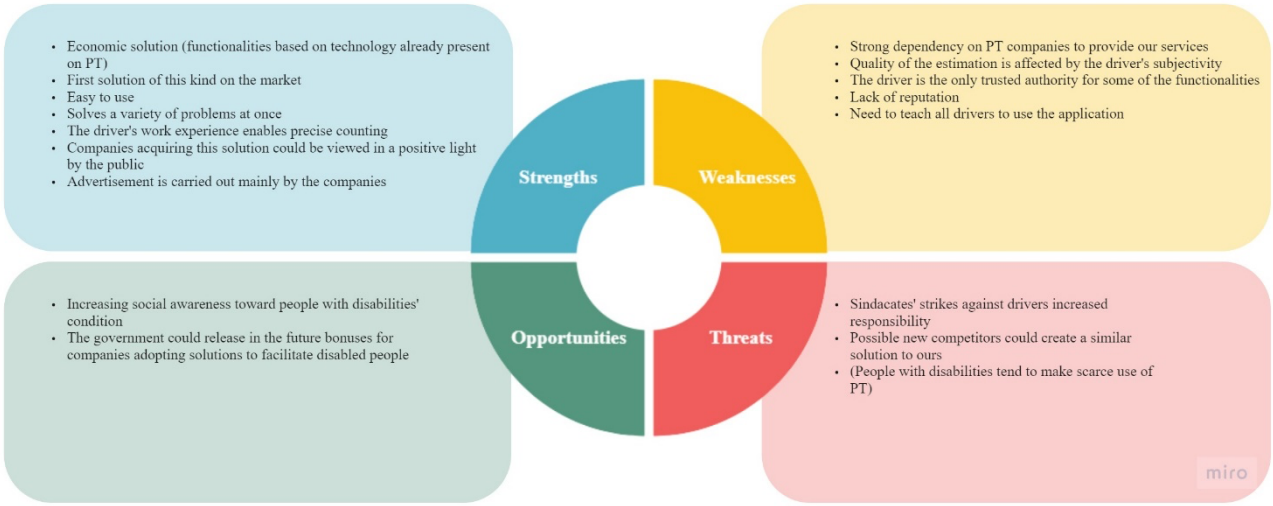


4.1.2 Driver Mobile Application SWOT

This SWOT deals with an app that could allow the driver to estimate the number of disabled and non-disabled people within the means of transport.

At the implementation level, this is a less expensive solution with respect to the previous one, and it requires less in-depth knowledge from the team. It is not based on specific algorithms but only on the expertise and effort of the driver, who should be trained to carry out this operation in the best possible way.

An important concept to notice is the implementation simplicity of this kind of solution, and this means that it is easy to copy for other competitors in the sector. So, the team should periodically release updates that improve the app and make it harder and harder to replicate.

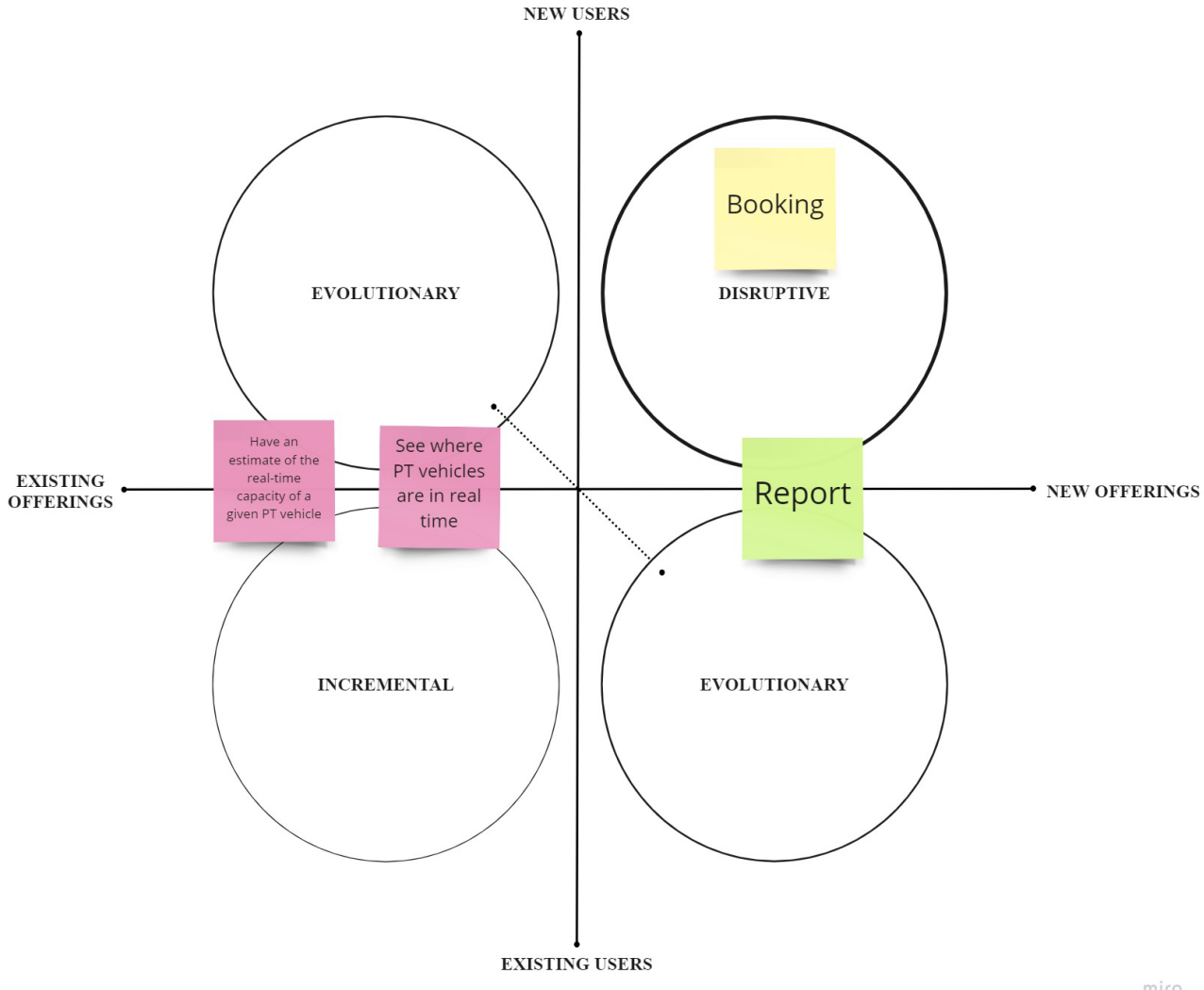


4.2 Promises & Potential Map

The graph below shows the main features that our application should have:

- Booking: Allows disabled users to reserve a seat in a PT vehicle. It could be a destructive feature because it targets a new service to users who are not the typical customers of a PT company.
- Report: Allows users to report a malfunction due to PT company. This feature is positioned halfway between disruptive and evolutionary because it certainly constitutes a new offer designed for the disabled but also normal users who are already customers of the company can use it.
- See where PT vehicles are in real-time: This functionality can be placed between incremental and evolutionary. It is aimed at all users of the app who, as previously mentioned, is a mix of old and new customers for the PT company. However, this functionality is already present in the Turin transport app, so it is an already existing offer.
- Have an estimate of the real-time capacity of a given PT vehicle: It is halfway between evolutionary and incremental for the same reasons as the functionality above. The only difference is that this offer is already implemented in the Ferrara PT company but in a different way.

It is important to note that the colors in the chart have no meaning but have been added to give more color to the picture.



5. Customer Journey

5.1 Chosen solution

After the SWOT analysis faced in the previous section, we decided to adopt the driver mobile application solution.

It means that there will be two applications. Bus/tram drivers will completely control the first one, and they will collect all the necessary information the project aims to provide to its users. On the other hand, the final customers will use the second app to visualize all the data gathered.

We based our decision on efficiency and cost-effectiveness. We based our decision on efficiency and cost-effectiveness. Furthermore, we believe that this solution will be more easily adopted by the PT company with respect to the one related to the machine learning solution because it is less invasive as regards the customer's privacy.

5.2 Customer persona selection

We decided to design the *customer journey* of the *primary persona* of the project. Thus, we will describe the *Customer Experience Journey Mapping (CXJM)* of a person in a wheelchair in the following pages. This kind of target is the one that would benefit the most from a solution like the one the project aims to provide.

The ideas that emerged in the previous sessions mainly focus on the "own" part of the service lifecycle. Therefore, the behavioral line designed starts from the steps immediately following the purchase and registration phase and finishes to the experience evaluation stage.

5.3 Behavioral line

To draw the below behavioral line, we put ourselves in the shoes of our primary persona Luca, the guy in the wheelchair. It is relevant to notice that color plays a fundamental role in the following graphical representation. The main steps are the green ones, while the orange one is the meeting point of the "buy" and "own" parts of the service lifecycle. Finally, the yellow post-it represents something to add or a detailed explanation of the previous green one.

After the registration process, a tutorial starts, and it will be essential for users to deeply understand the various features our application provides.

The first main step of our application consists of setting customer preferences such as the choice of lines and stops. It is relevant to underline that this information is not required during registration because it is not something fixed, and the user can change it whenever he wants.

Now our customer is ready to use the platform. He can select the line, the stops and check whether the selected ones are free from physical barriers. Then he can choose a specific means of transport and get more details about it. In this way, he can access information such as the current number of people inside, whether the vehicle is wheelchair accessible or not, and whether the route he wants to do is available and not yet booked. After analyzing this data, he can decide whether to go to the bus stop and jump into the vehicle or not.

Once at the stop, the user may find his seat occupied because a person who does not have the app is sitting there, and the driver has not promptly reported it.

At this point, the user should send us feedback in which he explains the situation. After the eventual report verification (by the driver), the user will be rewarded to apologize for the inconvenience.


Registration
Application's usage tutorial
First application start
The application asks about the user's usage preferences
Choice of preferred transportation lines
Choice of preferred PT stops
The user checks the bus current capacity.
The user selects the bus
The user selects the stop where he will get on
The user checks which are the stops without physical barriers where he can get on
The user selects the line
The user wants to plan the route
The users checks the statistics of the stop to see how many people will likely get onboard
The user checks and selects the stop where he will get off
The user books the reserved space to signal his presence
The user goes to the stop
The user finds out that the space is occupied by another wheelchair user
The user notifies the situation with the reporting system
The report gets verified
If the report gets validated, the user will receive some kind of compensation

5.4 Customer Journey Map

In this part, we summarize the customer journey from the awareness phase to the retain one.

In particular, we want to enlighten the following aspects: the challenges we have to focus on, how to measure the impact through metrics described, and, finally, where or how to achieve new improvement opportunities.

Since we provide a turnkey solution, it could be hard to fill in the awareness part of the map because the communication and marketing campaign is delegated to the buyers (i.e., the PT companies). They have the responsibility to make people aware of the new service they will provide. For this reason, we approached this part from their standpoint.

Since customers could use the application without registration (e.g., to see the nearest buses to a specific stop), we decided to consider the "download" and the "registration" as a single stage.

Moreover, we introduced the "verified user" concept. It is a person with disabilities who compiles a specific form to prove her/his condition. The goal is to distinguish our main targets from the other personas that could download it, since some services, such as booking the reserved seat, must be dedicated only to them.

Finally, we also included a new key idea that will improve the probability to receive feedback and reports: the "benefit" system. This system will reward those users who constructively send us feedbacks and reports.

To better understand the relevance of each touchpoint we marked them with a colored spot in the following way:

- Red spot 🔴: the most relevant touchpoints
- Yellow spot 🟡: medium relevance touchpoints
- Green spot 🟢: the least relevant touchpoints

LIFECYCLE STAGE	AWARENESS	DOWNLOAD & REGISTRATION	ONBOARDING	CONVERT & RETAIN
USER EXPERIENCE				
TOUCH POINTS	• No-profit events • Site design, content Articles on newspapers • Dedicated pages on no-profit associations websites	• Download app button • Registration from on the app • Upgrade form for verified accounts	• Application tutorial • Guided preferences setup • Intuitive UI	• Support through different channels • Reporting system • Benefits to the users if they are active on reporting/reviewing
CHALLENGES	• make agreements with no-profit associations • understand where website visitors are coming from • get medias to talk as much as possible about this new service	• design a registration form fast to fill • obtain reviews on the digital stores • find a way to verify a person's disability directly from the user ID	• adding more features and presenting them to users in a clear and effective way • handling the growing number of partner companies that will implement the application	• differentiate the support channels • plan the benefit system • develop a well functioning reporting system
RECOMMENDATION				
METRICS	• number articles mentioning our product • bounce rate • time between entering the product page and clicking on the download button • percentages related to where people discovered the service	• download percentage • registration percentage • registration time • verification percentage	• monthly number of users online • weekly number of seats booked • time needed to choose book a seat	• churn rate • customer rate actually using benefits • number of customer problems solved by support systems • report rate verified correctly • report rate solved
IMPROVEMENTS & OPPORTUNITIES	• create ad-hoc content on real customer experiences • create a communication campaign (social pages, testimonials, advertising) • create a website section dedicated to the articles the product is mentioned on	• ask for reviews during usage • use unicode for disability	• possibility of rearranging the UI regarding the user preferences • implementation of a notification system to keep users to alert them of incoming events • showing to users their most taken route, to enable a faster booking	• add other support channels (e.g., chat on the app) • add an app section where the customer can monitor its report progress • add a classification system/ranking based on the number of suggestions/reviews/ verified reports

6. Goal Posters

After designing the structure of the customer journey, the focus was shifted towards analyzing in detail two specific use-cases that are close to our users and are deeply involved with the core functionalities of our proposed solution.

The methodical process that has been executed, highlighted:

- The personas objective in the use-cases.
- The temporal horizon these use-cases take place in.
- The questions that the personas ask when the use-cases take place.
- The relevant metrics used to collect data regarding what was asked.
- The actions that are carried out to answer the questions they are linked to by using the defined metrics as a base.

As a result, the two use-cases were structured with more precision, and the focal points are displayed to convey important information in a clean and easy to understand manner.

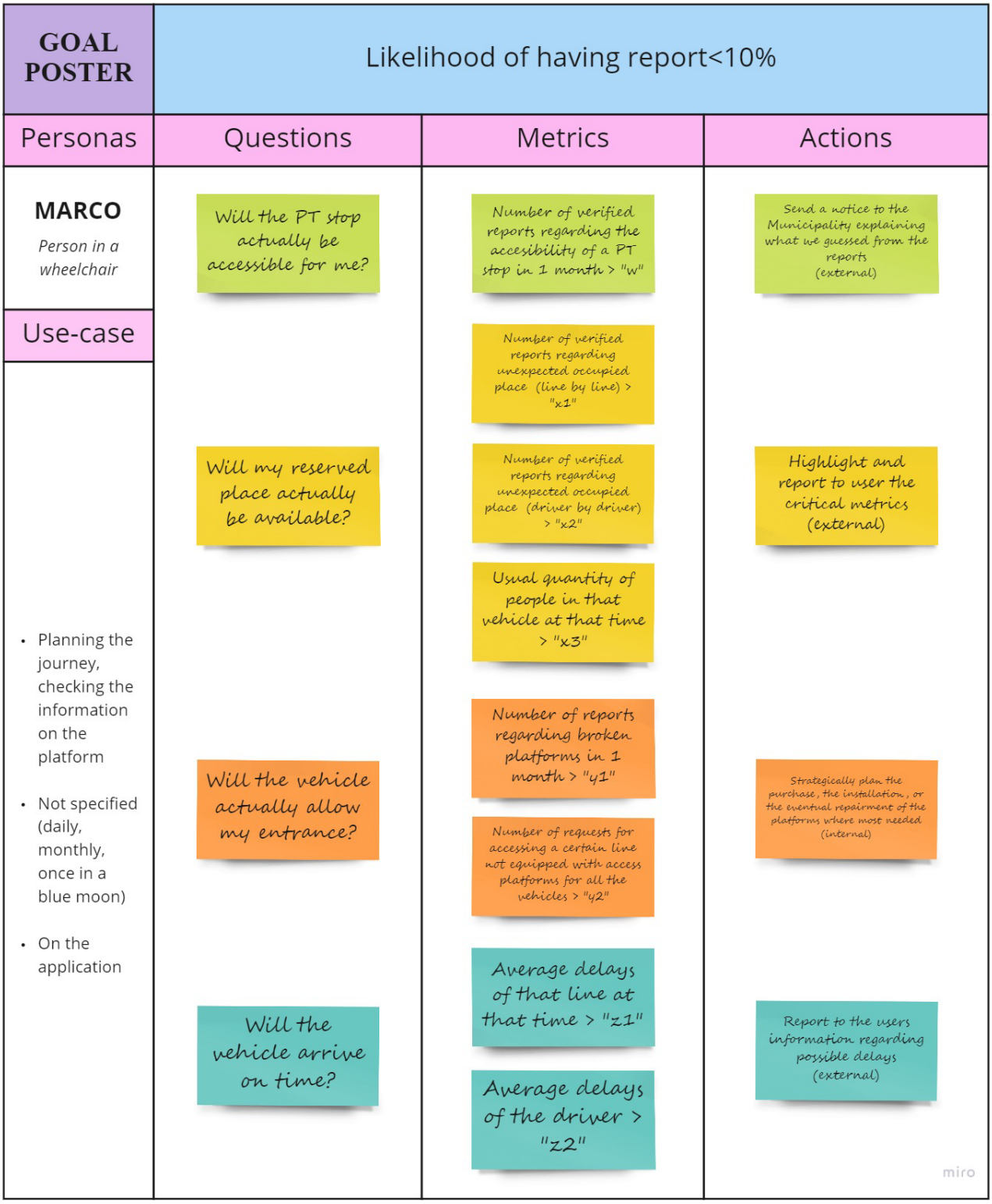
The questions, relevant attached metrics and correlated actions are grouped together with different colors to highlight the various course of actions.

The first use-case goes over the doubts that a person on a wheelchair has when using the application's functions, like questioning the actual availability of a reserved seat and the presence of physical barriers that are not notified.
As these doubts are about trust issues that every user with disabilities is likely to have at first, we identified this use case as the most important one. Every potential problem raised by the persona is linked to one to three metrics that converge in an action each for category, to present the practical answer to the question. It was also taken into consideration that each company adopting our solution might use different standards to evaluate each situation, by the usage of generic parameters in our metrics, when referring to thresholds. Also, as the action regarding reports notifying problems with PT stops involves a third party, the municipality, the validity of these reports as a whole is taken into consideration only if at least a certain number of different users signals the same issue.

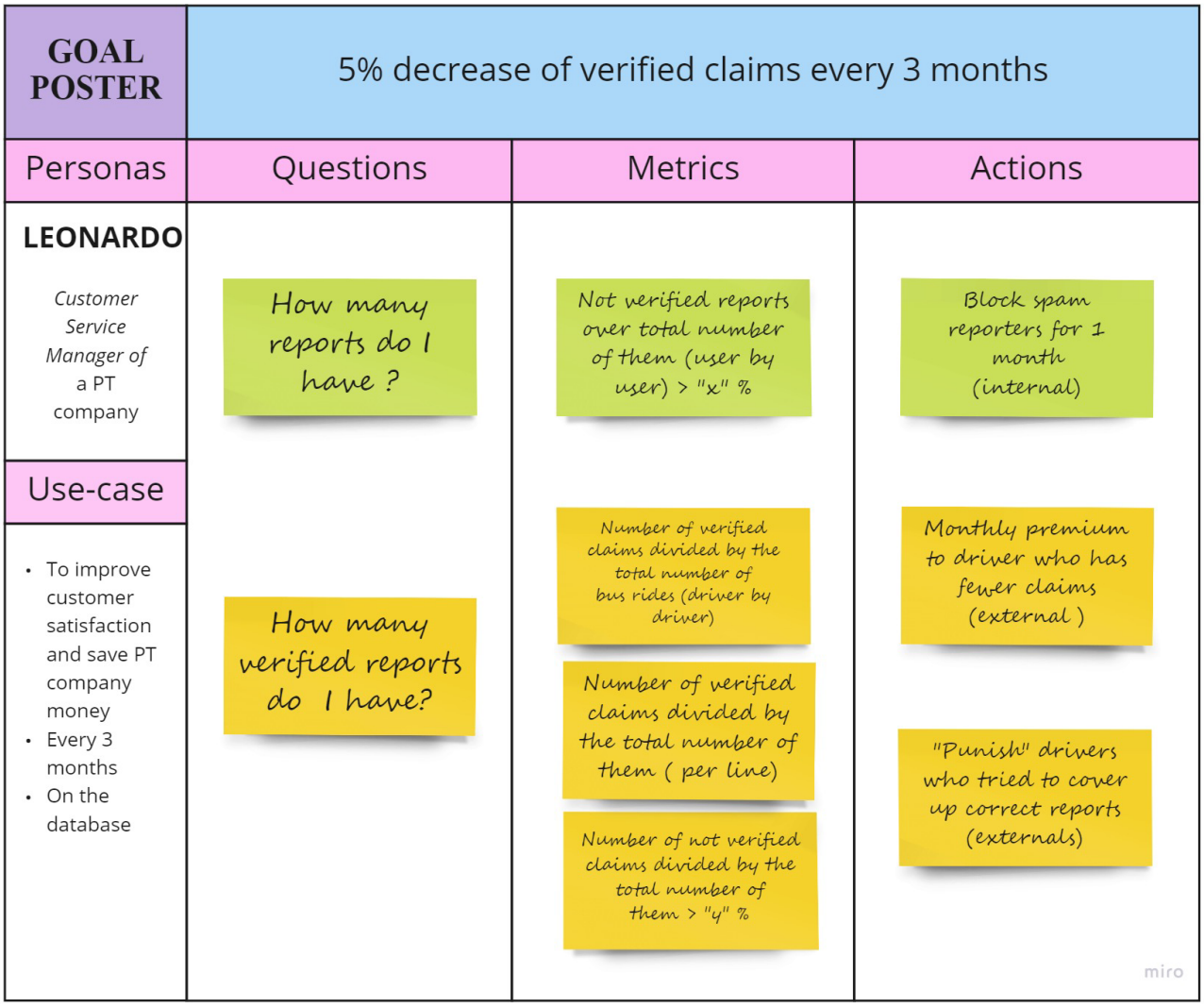
The second use-case explores the situation of companies that decided to integrate our application into their business logic. By focusing on the total amount of reports received and on analyzing their validity, the company can monitor the overall effectiveness of the solution and take appropriate action.

As this use-case is more business-driven than the precedent, a possible timeframe of three months at a time was chosen to conduct all the relevant analysis on the date gathered.

6.1 *Primary Persona Goal Poster*



6.2 *Customer Service Manager Goal Poster*



7. BPMN

The project proceeds with the definition and explanation of some fundamental processes the users will perform to use the app or the stakeholders would have to carry out to keep the correct functioning of the platform.

To obtain more details about these processes and have their graphical representation, we exploited the Business Process Model & Notation. We designed three different models:

1. Booking BPMN
2. Report BPMN
3. Driver BPMN

The first design concerns how the system responds when a user perform a booking on the platform. The second representation should clarify how the digital app deals with customers that try to create reports and ask for the related rewards. Finally, the last one explains how the platform manages any possible interaction that occurs when a disabled person gets on the bus or when someone who has booked does not show up at the stop. This last BPMN deeply involves the driver, who has an active and fundamental role inside the process.

We decided to analyze the first two processes because we consider them fundamental since they give a detailed view on how to deal with the two core functions performed by the users.

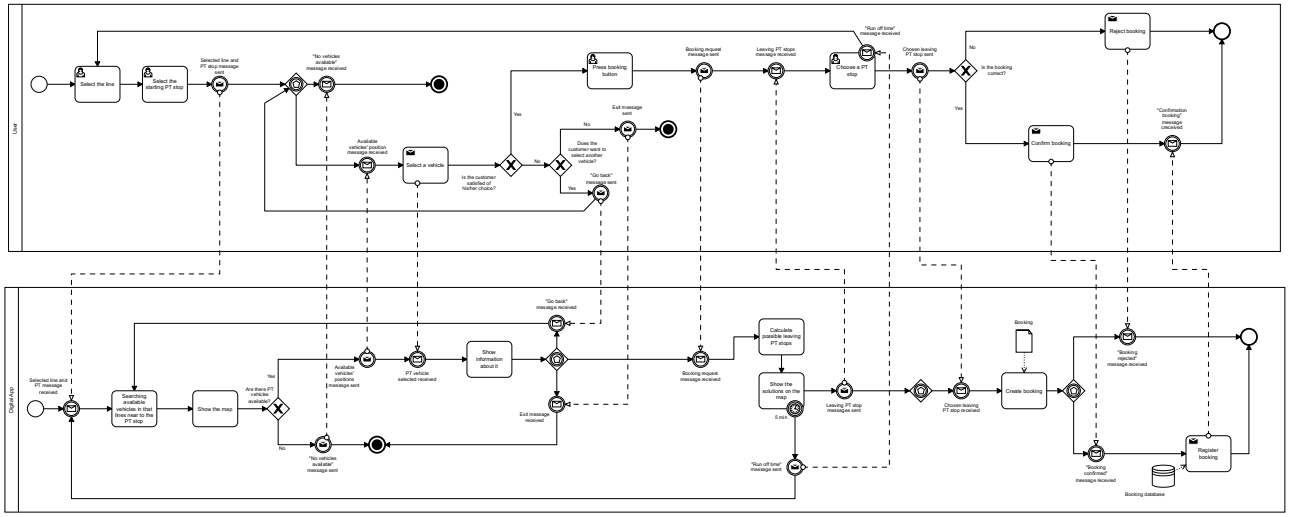
On the other side, we opted to analyze the third situation to answer some questions about the behavior of the driver in standard situations and borderline cases.

7.1 Booking BPMN

The process starts with the digital app that waits for the users' line and bus stop choices. Once obtained, it can search for possible travel solutions and eventually show the map with the available results (if there are any). These results are in the form of buses/trams icons moving on the map. They are somehow illuminated if bookable and blacked out if not available.

If a user clicks on a highlighted vehicle icon, she/he can have some further information about it (e.g., how crowded it is, the reliability of posted the arrival time, etc.).

At this point, the customer can book the solution that better suits its needs, and only here she/he can choose the arrival PT stop. The availability of the latter will be bound by the other bookings made on the platform.

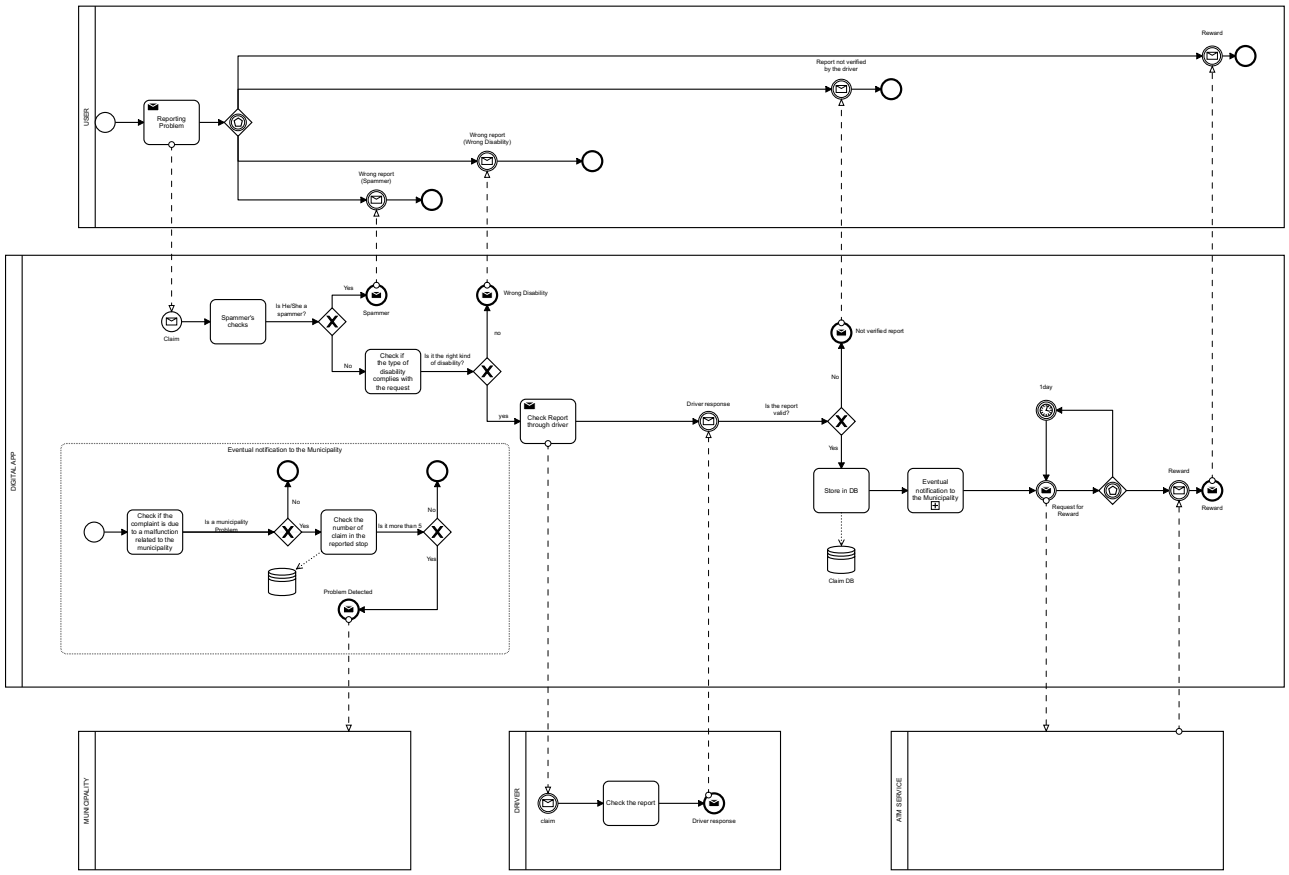


7.2 Report BPMN

This process starts with the user who reports a problem to the system. From that moment on, the digital application applies some filters to the report to check if it is valid and coherent. Most of these filtering actions are carried out directly from the app. However, the driver executes the last validation step. If the driver confirms what was reported by the user, the system stores the verified report in the DB.

If the report includes a claim regarding something strictly linked with the municipality, and a certain number of similar complaints are already present in the DB, the system can send a message to inform those in charge about this issue.

Finally, the following steps of the process regard reward management. If the system verifies the report, it can then ask the ATM/GTT service for the reward to give to the user. Since rewards have a fundamental role in the project, in case of no responses by the PT company, the system would make multiple requests to it (one per day) until an answer occurs.



7.3 Driver BPMN

The following process tries to answer questions like:

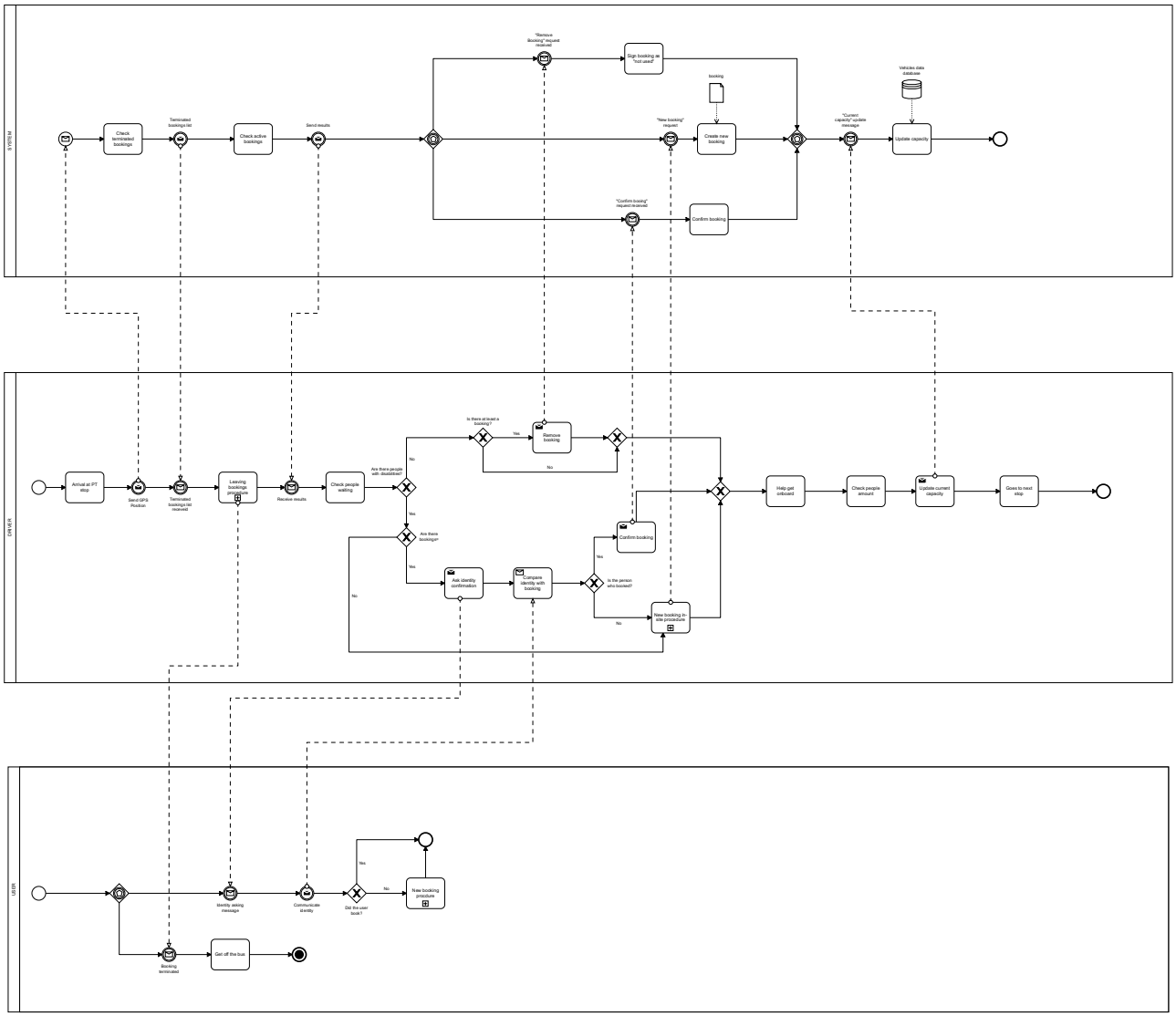
- What would happen if a person in a wheelchair who does not use the app gets on a bus?
- What would happen if a person who booked the place does not get on the reserved bus?

Only with the help and work of the driver, the system can correctly manage these kinds of situations.

The driver, interacting with the digital app or using its intuition and experience, has to perform the following actions:

- Ask the system if there is a reservation at the current stop.
- Try to understand if, at the stop where she/he is passing, there may or may not be a disabled person who will get on the bus/tram.
- Ask the possible new passenger if it uses or not the digital app.
- Compare the identity of who booked the place with the one who is trying to get on the bus/tram.
- Eventually, make new reservations or delete old/expired ones.
- Inform the passenger who did not book the seat but who is getting on the bus/tram the longest route she/he can have with that specific vehicle without overlapping with an eventual already booked from another user bus ride.
- Help the passengers in a wheelchair to get on/off to/from the bus/tram (task already performed by drivers).
- Update the current capacity of the vehicle.

These actions must be carried out in the order described by the following BPMN to let the digital app work also in borderline cases.


PUBLIC
Check terminated bookings
Terminated bookings list
Check active bookings
Send results
"Remove Booking" request received
Sign booking as "not used"
booking
"New booking" request
Create new booking
"Current capacity" update message
Vehicles data database
Update capacity
"Confirm booking" request received
Confirm booking
DRIVER
Arrival at PT stop
Send GPS Position
Terminated bookings list received
Leaving bookings procedure
Receive results
Check people waiting
Are there people with disabilities?
Is there at least a booking?
Yes
Remove booking
No
Are there bookings?
Yes
Ask identity confirmation
Compare identity with booking
Is the person who booked?
Yes
Confirm booking
No
New booking in-site procedure
No
Help get onboard
Check people amount
Update current capacity
Goes to next stop
USER
Identity asking message
Communicate identity
Did the user book?
Yes
No
New booking procedure
Booking terminated
Get off the bus

8. UI/UX

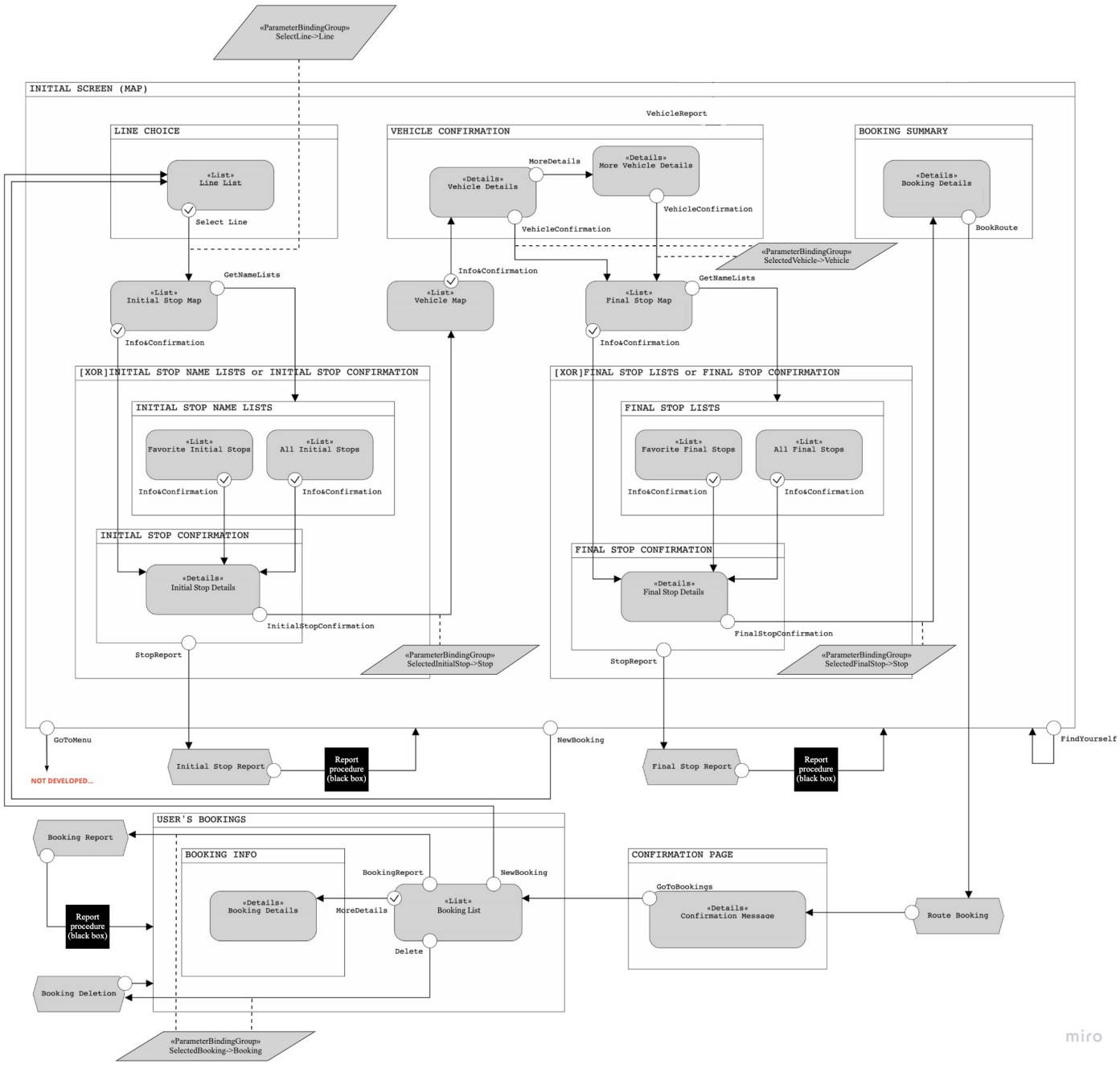
After delineating the BPMN concerning the booking and report processes and related to the driver's actions when someone gets on board, we proceeded by defining the IFML model of few interactions, and the Wireframes and Mockups of some of the screens that users will directly see when using the application.

The IFML model enables us to cover in detail the content of the interfaces that will be presented on-screen and the behavior that people can apply over them, in terms of events that can be triggered and, consequentially, the target results of the interfaces' status.

The Wireframes and Mockups were then defined to give a graphical representation of what was described, showing how the different pages are connected with one another and how the content is organized within them.

8.1 IFML

The model represents the interactions that occur when a user tries to book a seat on a public transportation mean, going over all the choices that are required to finalize the booking.



8.2 Wireframe

By analyzing the entirety of our application structure, we identified two sections that required particular attention in the organization of the contents inside.

The first represented screen is the real-time map, as it constitutes the homepage of the application and shows the movement of the transportation means and the various stops.

The second represented screen is the page containing all the past and present bookings the user has done. Each of them is shown in a compacted form, presenting only essential information and the button to report an abnormality, but by clicking on them, more details are displayed.

8.3 Mockup

Mockups are based on the wireframe of the map page, showing the evolution of the application's content when a user interacts with it to get information about PT stops, transportation means, and their real-time position.

In particular, the mockups follow a specific order. They must be read right to left and up to down.

Figures 1 and 2 are not explanatory as regards the booking process. Their scope is to introduce the name and the logo of the application.

The first screen the user bumps into shows a map, a sort of loading bar in the upper part, and three buttons in the bottom one (Figure 3).

The loading bar will fill up as the booking is completed. The button in the bottom-left corner, if pressed, will show the main menu of the application. The opposite button, the one in the bottom-right corner, has the functionality to zoom in or zoom out the map to let the user have a better view of the stops or the vehicles. The last button, the most important one (i.e., "Prenota ora") triggers the booking process.

After the selection of the line through a list (Figure 4) and its confirmation (Figure 5), all the PT stops associated with that line will appear on the map (Figure 6). Now, the user can choose the stop directly from the map with a tap on the related pin. If she/he is not aware of the map of Milan, he can ask the application for the names of the stops (Figure 7). Whether the user chooses directly from the map or using the names list, she/he must then confirm it to move on to the next step (Figure 8).

The latter consists of the vehicle choice. The system will show to the user some of the vehicles belonging to that line and arriving at the chosen stop to let her/him choose the one that fits better with her/his needs (Figure 9). Even in this case, it is possible to ask for the written list (Figure 10). From both ways to visualize the vehicles, the user can guess only some information regarding them (i.e., the current quantity of people on the bus, whether the place for the wheelchair is available or not). If the user is interested in a specific vehicle, she/he can tap on one of them to have more information (Figure 11). Once the user decides, it can confirm the selection (Figure 11).

The last relevant step of the booking process consists of selecting the final stop. The steps are very similar to those already seen for the initial stop selection (Figure 12, 13, and 14).

Finally, the last operation is to control the booking summary and confirm it (Figure 15). If the system manages to make the reservation without errors, the user is notified through a specific message and redirected to his reservations (Figure 16).

In all three confirmation containers (Figure 8, 11, and 14), there is a specific button in the upper-right corner that triggers the reporting process.

Figure 1: Home Screen

Figure 2: Logo Screen

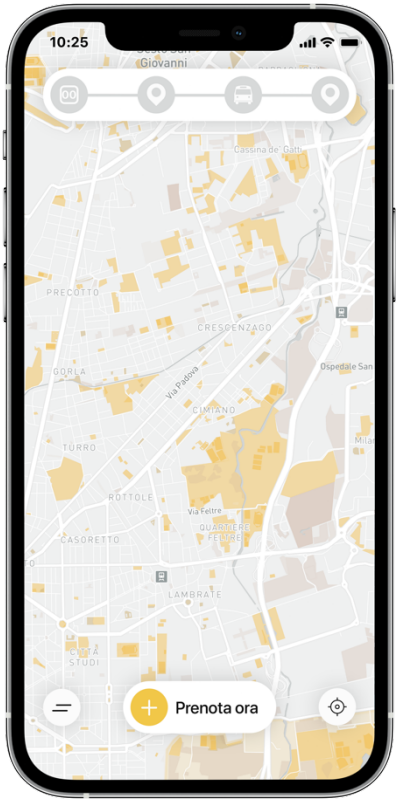


Figure 3: Initial Screen

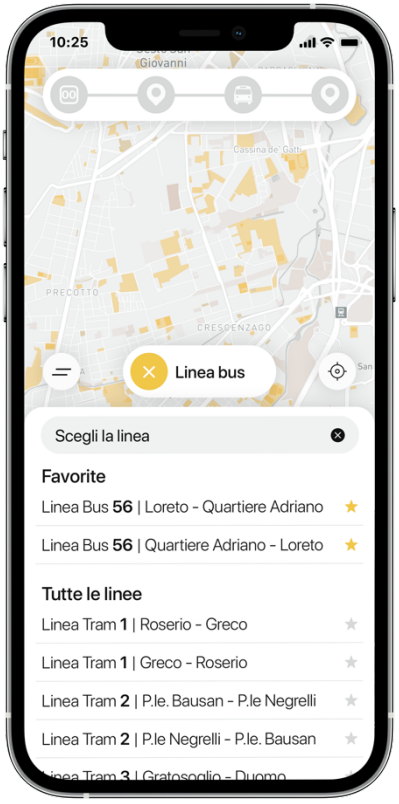


Figure 4: Line Selection

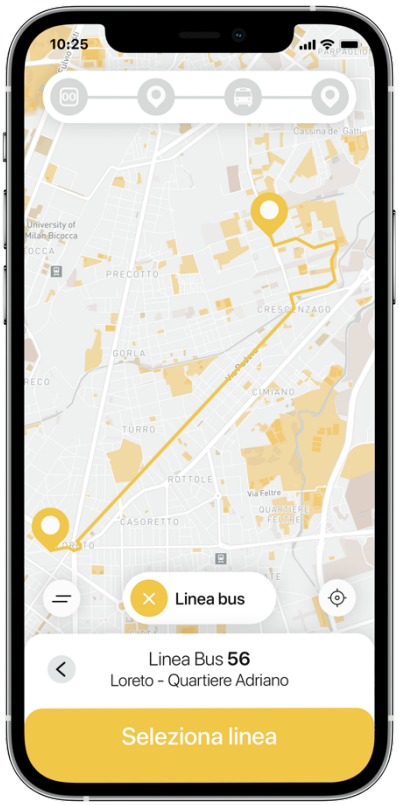


Figure 5: Line Confirmation

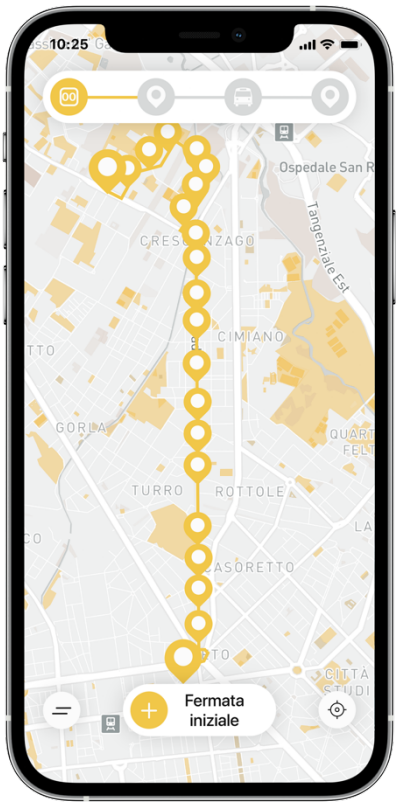


Figure 6: Initial Stop Selection (map)

Figure 7: Initial Stop Selection (name list)

Figure 8: Initial Stop Confirmation

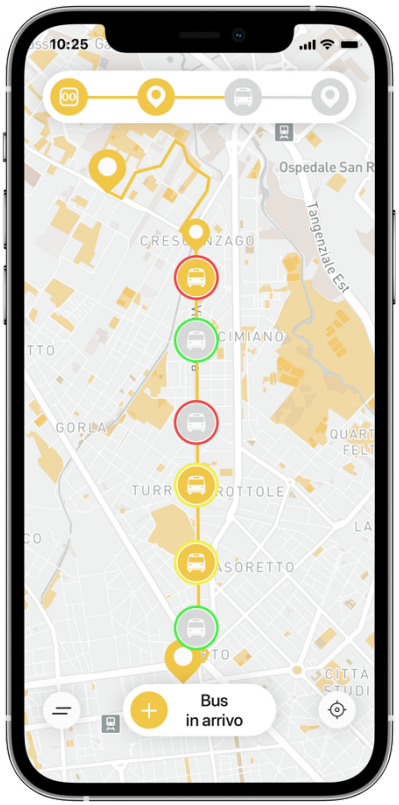


Figure 9: Bus Selection

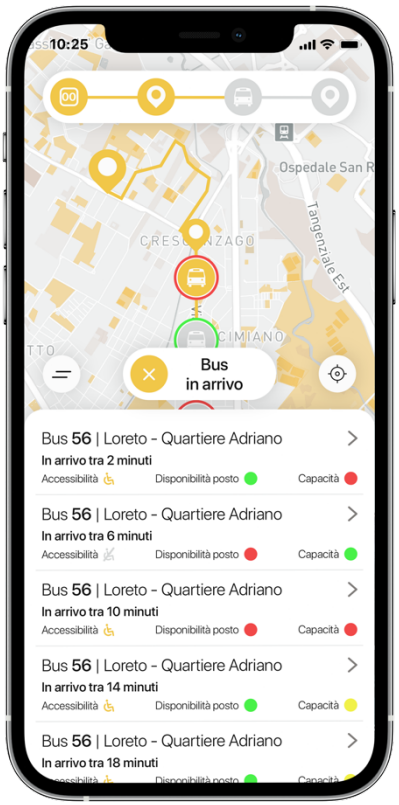


Figure 10: Bus Information

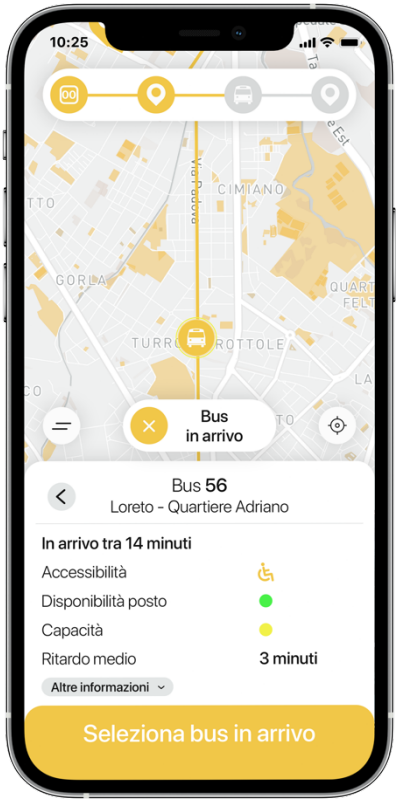


Figure 11: Bus Confirmation

Figure 12: Final Stop Selection (map)

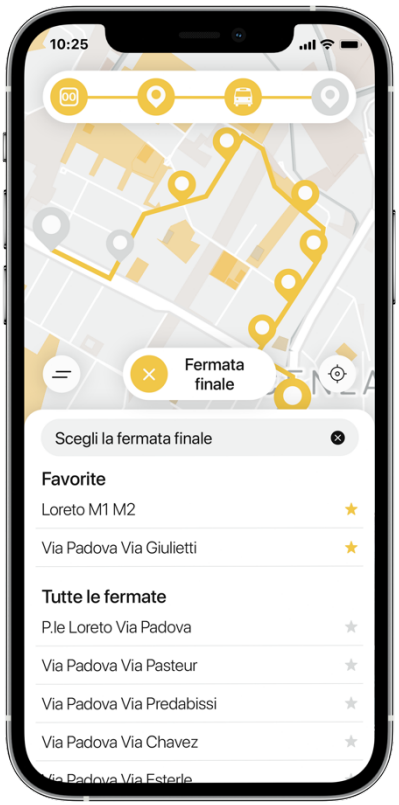


Figure 13: Final Stop Selection (name list)

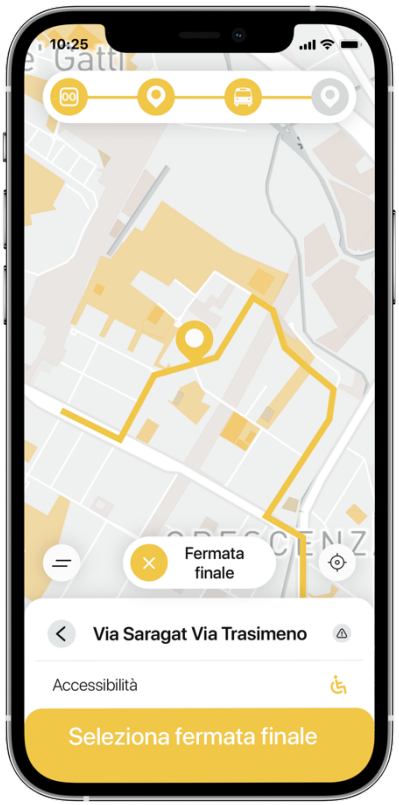


Figure 14: Final Stop Confirmation

Figure 15: Booking Confirmation

Figure 16: Confirmation Screen

9. INFORMATION SYSTEM and ENTERPRISE ARCHITECTURE

9.1 Data-driven decision

In this section, the types of collected data will be analyzed, with particular attention on the different report kinds, and which decision or action they can lead to through a data-driven decision approach.

Particularly, this part will be divided into two macro-categories: our company's perspective and third-party one.

Each type of data will be explained with a short introductive table that will follow this below:

Data	Type	Decision	Action

OUR COMPANY PERSPECTIVE

Interaction through the screen (touches)	Prescriptive (Decision Support)	Redesign the user interface	Change entities position Set different first type of view
Visit website	Prescriptive (Decision Support)	Redesign the website interface	Highlight the most accessed information Change pages position

The focus of the company's attention is to improve the user experience and the product's awareness. The touches can help us to recognize which parts the user usually utilizes, such as the list of the PT stops instead of the map. In addition, the number of people who download the application must be as wide as possible, to guarantee efficient service. Given the B2B model, the website must be as effective as possible to attract new potential company that would like to power up their PT services.

THIRD PARTY PERSPECTIVE

Registration data	Prescriptive (Decision Support)	Increase the users' awareness	Segment the new target and create a new communication campaign

Using the registration data, the PT company could understand which is a typical user, especially in terms of age. The system will work at best when the user type will be "liquid" as possible, therefore avoiding too many bookings created by the driver. The main idea is to attract every kind of person and, to do this, an effective communication campaign must be designed.

Report concerning the driver + verified report declared as false	Diagnostic	Discourage to not respect the guidelines	Punish the driver Review the process
Report concerning the driver (amount)	Prescriptive (Decision Automation)	Encourage to follow guidelines	Reward the drivers with less claims

In the first two rows, the drivers' management is the key area of interest in both positive and negative situations. In the first case, the data must raise also different questions, such as: is it only a driver's fault or it is the process incomplete, too complex, or too slow? For this reason, it must be only diagnostic, instead of prescriptive. There is no possibility to immediately reach a decision, even with the system support, the "why" must be found out first.

The second situation could be prescriptive with decision automation, it is just calculating which driver receives the least verified reports.

Report on wrong information given by the system	Prescriptive (Decision Automation)	Fix the error	Change the information

In this situation, the system contains wrong information, such as a PT stop that was considered accessible when it is not or the same thing about a vehicle. In this case, the issue does not need a particular human decision, it must be checked and then corrected. The control can be done through the Municipality (in case of PT stops) or by an employee in the deposit.

Feedback about PT stops	Prescriptive (Decision Automation)	Report the PT stops	Contact the Municipality
Feedback about inaccessible vehicle	Predictive	Modify the float	Depending on budget: 1) Add the platform in the vehicle 2) Redistribute the float

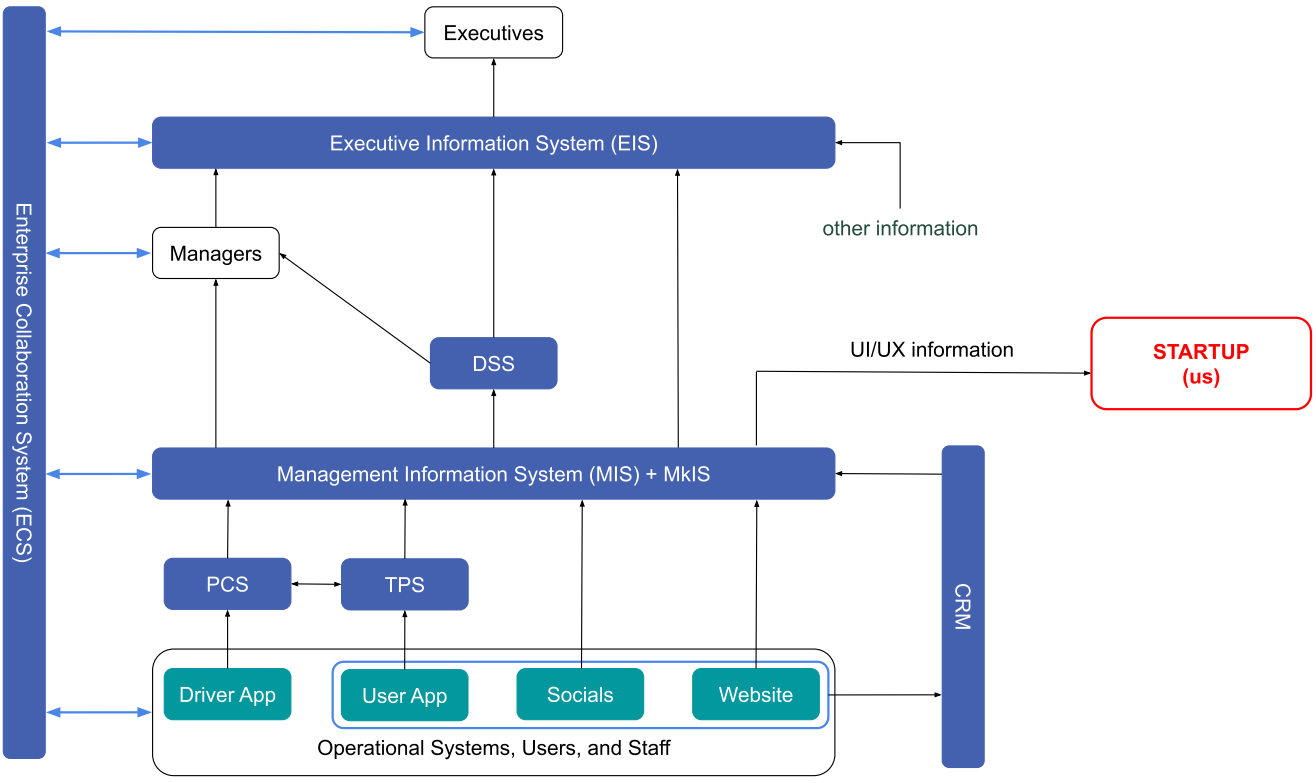
In the application, the user can suggest some improvements regarding the buses that he/she cannot take due to vehicle inaccessibility or low capacity. These data can be used to, depending on budget and the other data inside the company regarding different type of users, redesign the float in terms of tool or daily

distribution. Therefore, the type cannot be prescriptive, indeed the budget and other situations' constraints imply the need of taking the decision also according to that.

Moreover, the inaccessible PT stops could be also a subject of the feedback written above. If the number of these is high, the company can contact the Municipality.

9.2 Information flow scheme

To develop the information flow scheme, we looked at it from our future client's standpoint (i.e., the PT company). So, apart from the red rectangle, every block of the following scheme will belong to the information system of a public transportation company. In the "Operational System, Users, and Staff" macro-block, there are 2 elements the client will buy from us (i.e., user and driver mobile applications) and two elements they already have (i.e., socials and website). We used the most classical scheme to represents the information flow of the client, and we suggested what to add at each block to integrate our solution with its system.



SUGGESTIONS:

PCS	TPS	MIS	DSS	ECS
• Driver system • Report validation system	• Booking system • Report generation system • Registration system	• Marketing information • Report & feedback • Bookings • App usage information	• Marketing parameters • Fleet simulation • Reward system • Municipality communication • App usage parameters	• E-Mail • Chat • File sharing

In a nutshell, to deal with the data produced by our application, the client should be able to add the systems indicated above (e.g., driver system, booking system, etc.) or to make capable the parts of the infrastructure

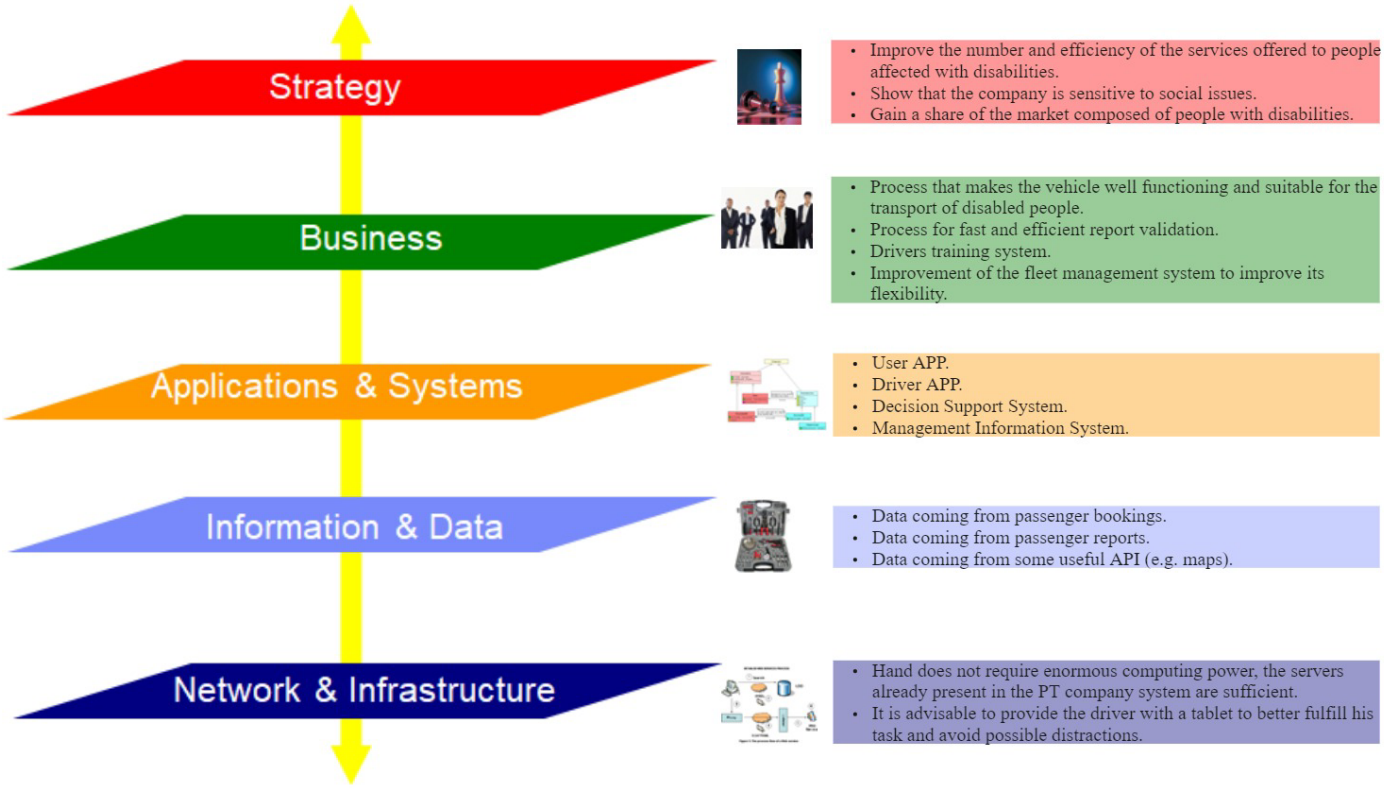
they already have of dealing with the kinds of information we indicated in the list (e.g., marketing parameters, app usage information, etc.).

9.3 Enterprise Architecture domain

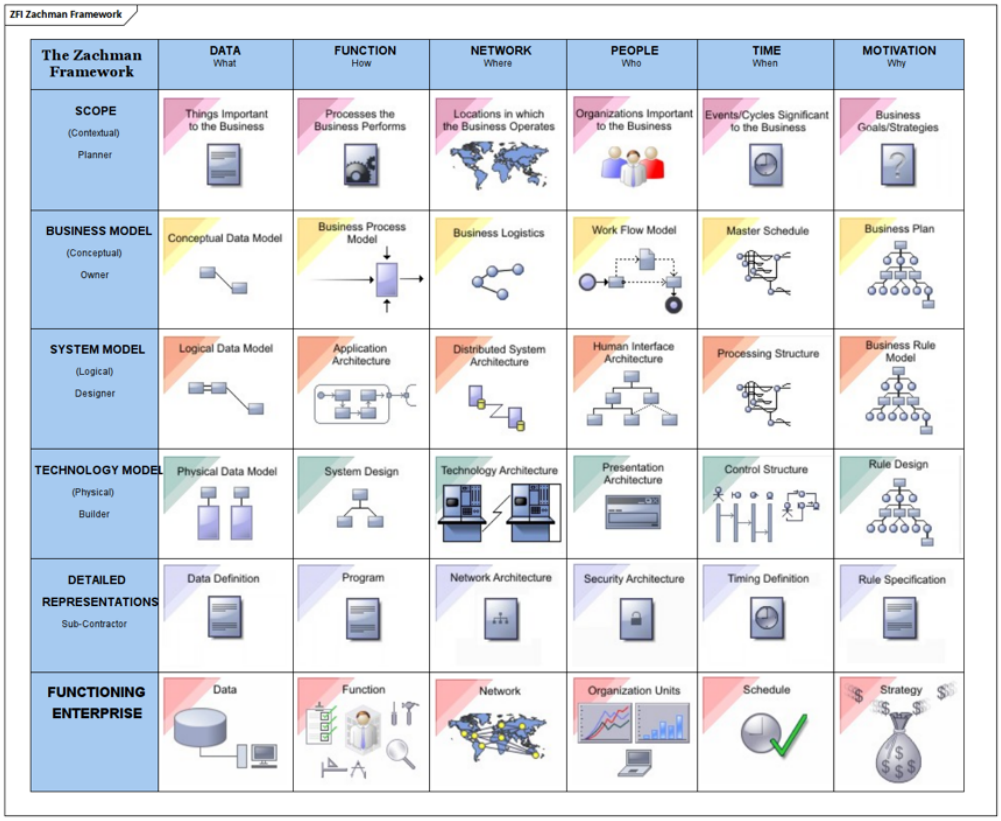
The picture below shows the various strategic advantages our customers could have. Following the scheme, after the strategic part for each section, some key points have been written to follow.

The most important to underline is what we find in the business section under the heading "Driver training system".

Since our app focuses on the work done by the driver, we believe that it is not enough to follow the tutorial dedicated to him but that the PT company must develop a training system to ensure that the driver performs his tasks in the best possible way. Since our app is focused on the work done by the driver, we believe that it is not enough to follow the tutorial dedicated to him but that the PT company must develop a training system to ensure that the driver performs his duties in the best possible way.



9.4 *Zachman Framework*



The figure above represents the classical Zachman framework. We will define in detail the first three levels of it in the following pages. We think that Planner, Owner, and Designer levels can be enough to describer our startup, and that further levels may be too difficult to implement at this point in the project.

9.4.1 Planner (Objective View)

1. What:
    a. PT companies' details (name, address, place)
    b. Statistics on report generated by companies
    c. PT companies' feedbacks
    d. Application performance data (UI and UX)

2. How:
    a. Data collection (usage & diagnostic)
    b. Software development
    c. Service level agreement (SLA) with buyers
    d. Support and maintenance

3. Where:
    a. Milan
    b. Turin

4. When:
    a. Starting beta-testing phase in August
    b. Official launch on the market in September
    c. Disabled people international day on December 3rd

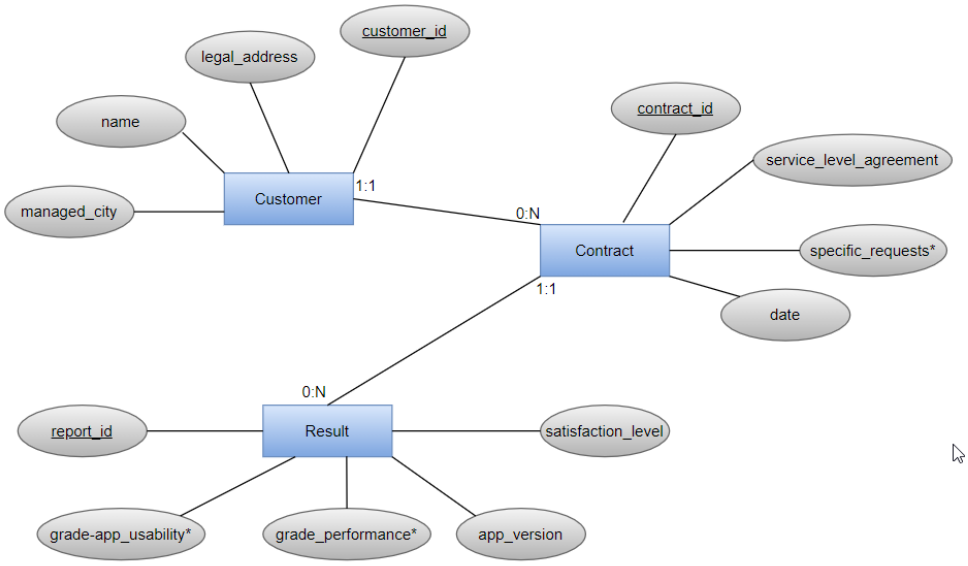
5. Who:
    a. Software development & technical support BUs
    b. Sales BU

6. Why:
    a. Create a brand image
    b. Guarantee a sufficient service to targeted users
    c. Capitalize on big cities' PT issue regarding targeted users

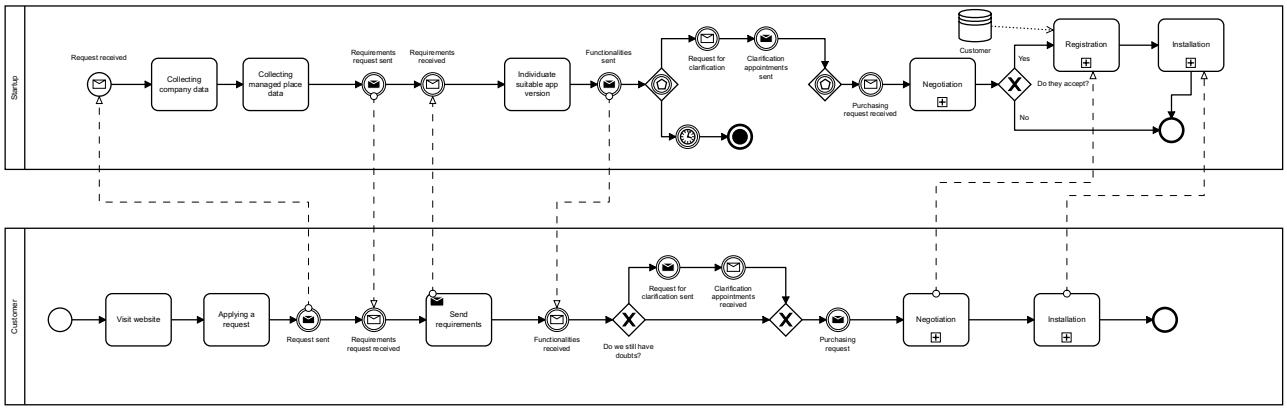
The most important data needed by our startup consists in the application's performance and satisfaction of our clients.
In the actual situation, the main objectives are Milan and Turin due to the potential tourism and an efficient PT system already implemented.
The events reported in the Zachman Framework are an obvious estimate and, the last one suggests a possible creation of an event to collect user feedback to promote our start-up to other cities (also European).

9.4.2 Owner (Conceptual View)

- What:



- How:



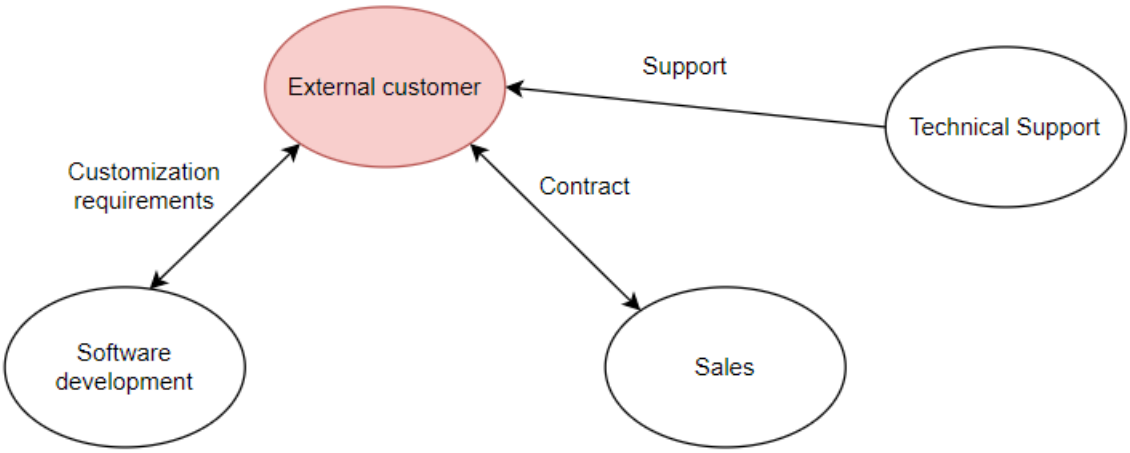
- Where:
  - Milan

- When:
  - The best way to represents the master schedule would be with a Gantt scheme. It would be quite complicated to design it already at this point in the project. However, we can already distinguish the macro phases to be scheduled:
    - Requirements collection → it implies the collection from all the stakeholders of the requirements needed for the developers
    - Project scope definition → it regards the definition of what will be part of the project and what it will not
    - Installation activities definition → it includes the definition of what the installation activities actually are, and the allocation of the time required by them

- After-sales support definition → it concerns the definition of what all the post-sales support activities will be and how they will be addressed by our startup

  We can now estimate how long these phases would last (the estimates are very approximate):
- Requirements collection: 1/2 weeks
- Project scope definition: 1/2 weeks
- Installation activities definition: 1/2 weeks
- After-sales support definition: 1/2 weeks

- Who:



- Why:
  - The idea that of the proposed solution is to find a way to answer all the needs that people with disabilities have to face if they want to use public transportation, a piece of the market that is currently untouched.

    The reasoning behind it is trying to make use of approaches that are already widely spread with similar applications, and cater them to the needs of this particular group of people, implementing useful features and revising existing processes.

    The choice of going for partnerships with PT companies to sell them our solution, instead of implementing it ourselves, is due to the lack of fundamental resources such an approach would take at infrastructural level, vehicles needed and staff to manage such a start-up correctly, all elements established PT companies already have.

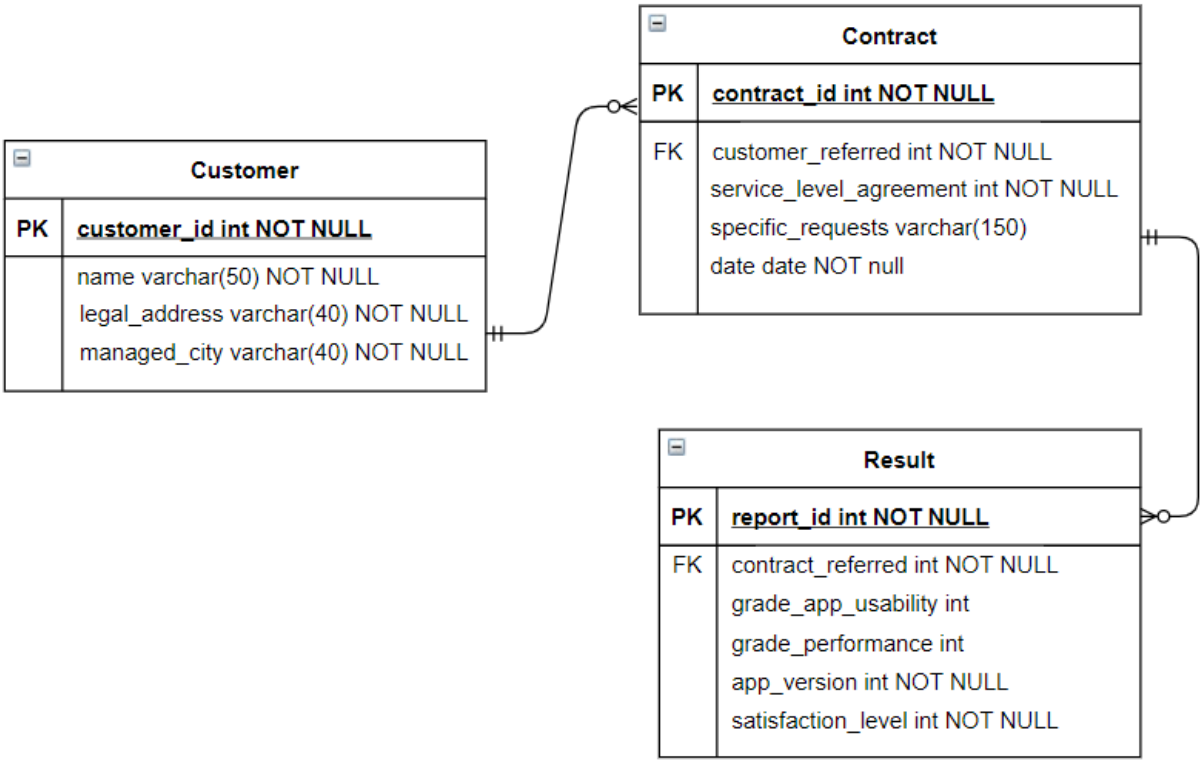
Being a startup, our database won't contain too many objects, indeed the most important records we need is about Pt companies' data, the stipulated contract, and the result associated with that one. In particular, each developed version of the application can achieve different results in the same agreement.

In the BPMN the process represents a possible first approach, avoiding going deeply for what concerns the installation and the negotiation.

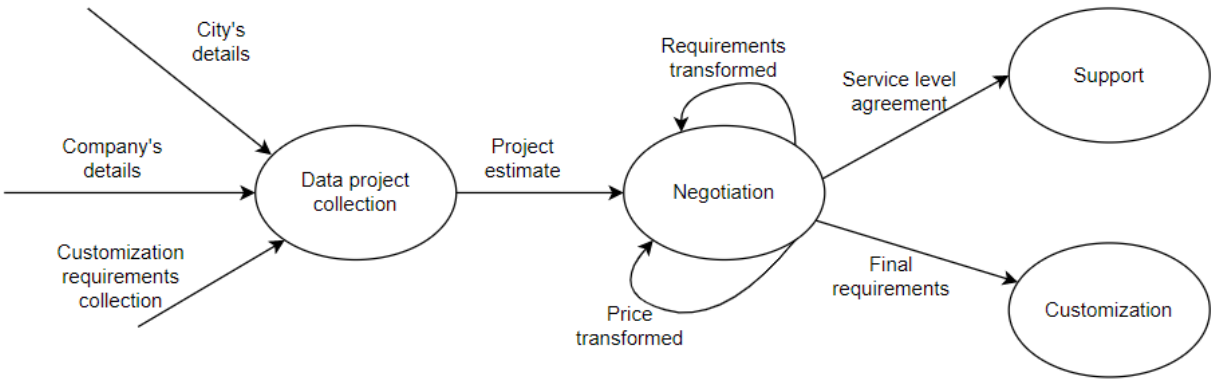
Moreover, it is suggested a representation of the responsibility for each business unit.

9.4.3 Designer (Logical View)

- What:



- How:

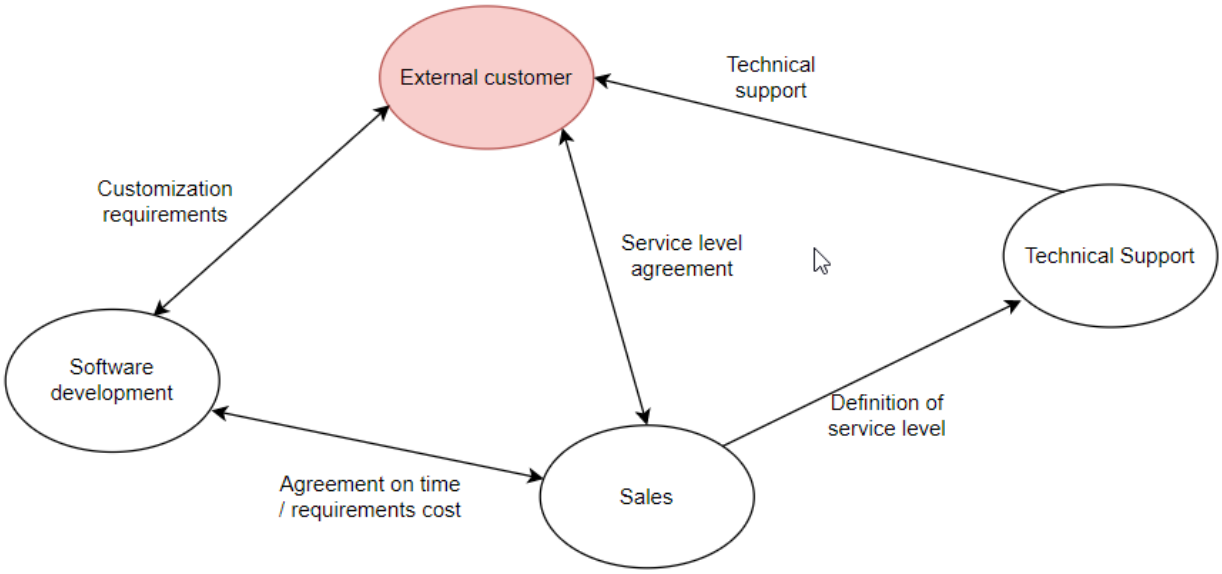


- Where:
  - Milan

- When:
  - Below are rough estimates that may be useful for our application. Everything will depend on the skills of the developers and those of the leaders who will take care of the operations. As for the testing part, we have sections with a short time interval because most of the tests will be done hand in hand with the writing of the code.

- Requirement: Translation of all business requirements into logical ones. Write the RASD document. Estimated time 1 month.
- Design: Write the DD document. Estimated time 1 month.
- Software: Development Application development, frontend, and backend. Estimated time 4 months.
- System testing: Testing Back End backend test paying attention to the security protocols used to avoid any attacks on the PT company system through our app. Estimated time 3 weeks.
- Front End testing: Testing user interface testing to ensure better usability. Estimated time 1 week.

Who:



- Why:
  - On a technological point of view, integration with the target infrastructure does not require particular changes, as the solution runs on MongoDB, which has little impact with components already operating. The choice of using an open-source technology adapts itself well to selling a single solution that has to be implemented with multiple systems, achieving high-integrability.
  - Only users can create a report
  - Only the driver can verify a report
  - Both users and drivers can book a seat
  - The same seat cannot be booked twice

10. CRM and Recommender System

In this section, we will focus our attention on the tools exploited to extract value from the data collected and which actions/decisions this mined information can lead us to.

Companies usually adopt at least two different systems to analyse and exploit owned data: CRM and recommender. Both would be integrated to the clients' information systems and they would have completely different impacts and utility degrees.

The CRM would be helpful for clients (i.e., PT companies) to retrieve information on our solution's performance, especially in terms of processes. Moreover, it would improve the final customer (i.e., PT users) experience.

On the other hand, it may be too much to adopt a recommender system. Probably, the benefits it would bring would not be as much as the costs suffered to develop it. Adopting it could imply a loss instead of a profit. This due to many reasons:

- Difficult to find a recommender system that is suitable for the activities our solution carries on
- Difficult to collect the right amount of data to apply machine learning algorithms
- The number of customers it would target is very low with respect to the total number of final customers a PT company has

So, in the following section, we will go into details as concerns for the CRM because we believe it would add value to our solution.

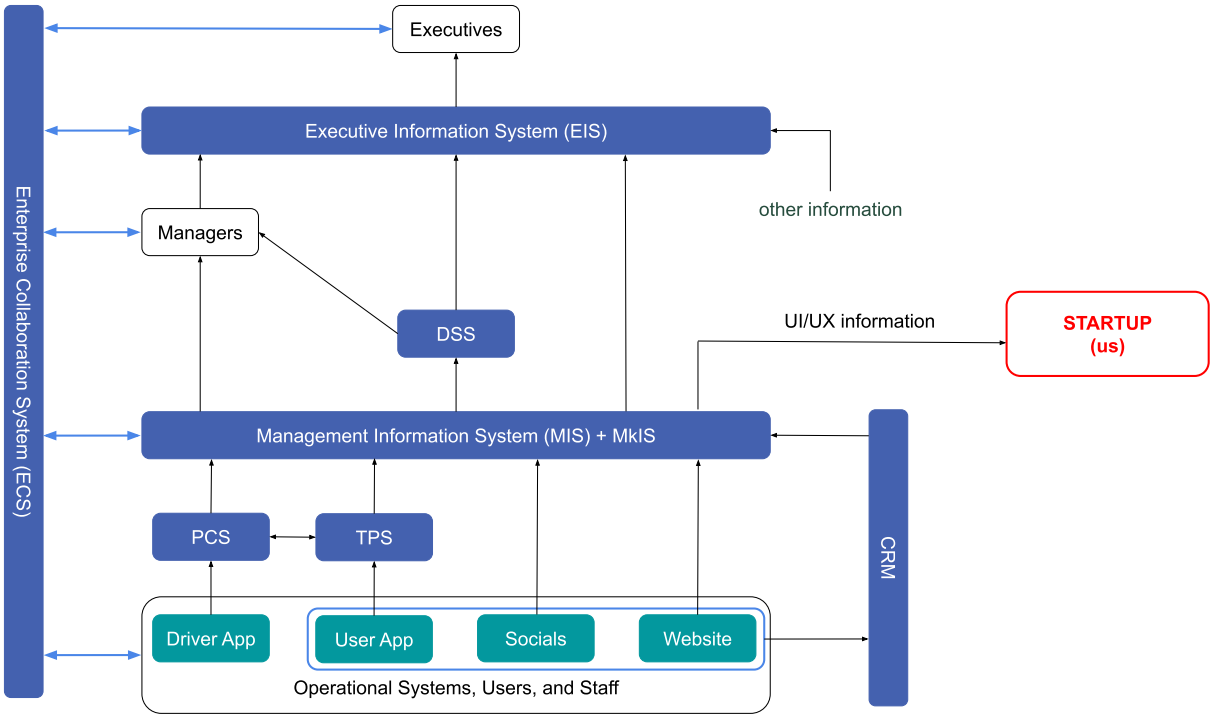
As for the recommender, we will make a proposal that could remotely fit with our goals and objectives, but we will not bring in this document full explanation of its possible implementation.

10.1 CRM

10.1.1 CRM Information System

In the previous section, we specified what to add to each component of the public transportation company's information system in order to integrate it in the best way possible with our turnkey solution. Now, let's focus on a specific block we have not analysed yet: the Customer Relationship Management system.

The approach with this system will remain the same. We will try to design a CRM that should be added to the PT company's current one to deal with the new kinds of data our app will generate and provide.



As the picture shows, the CRM of the PT company will receive information from all the touchpoints it will have with customers. So, the CRM's task is to collect data from the *user application*, the *social pages*, and the *website*. In this way, it can obtain different kinds of information and consequently improve different nuances of the relationship with customers.

It will deal with this list of data:

- bookings
- reports
- registrations
- usage data on the website and socials
- support requests

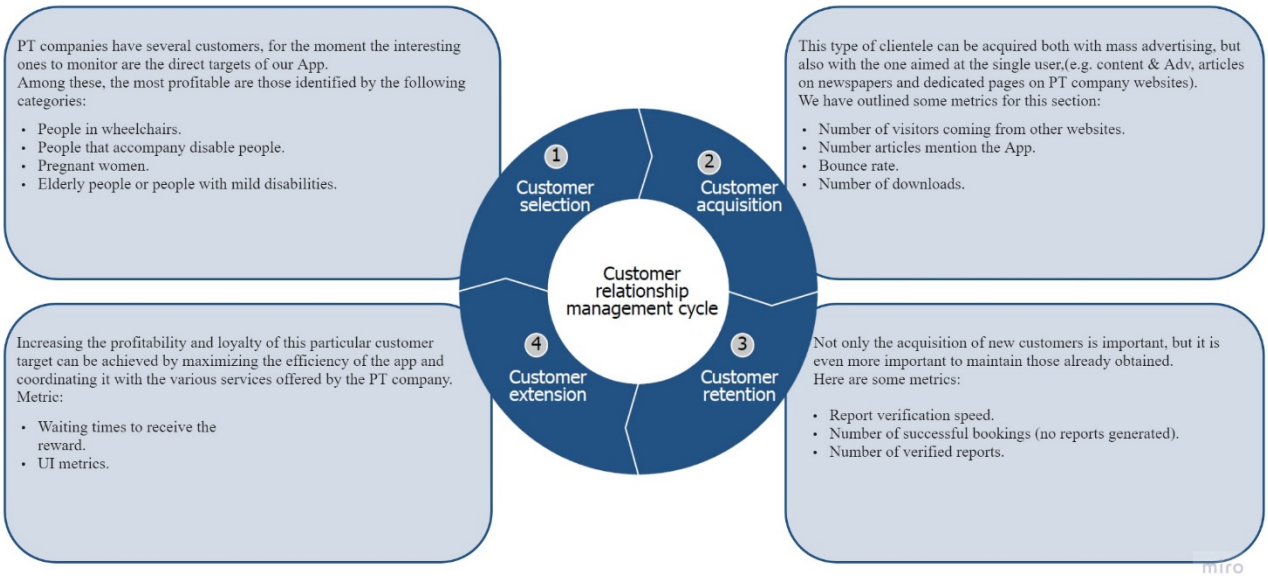
The CRM is not only an information collector, but it also has analytical functionalities to extract value from data. From an analytical standpoint, the CRM will have these two main functionalities:

- customer profiling → the CRM can profile clients by looking at the most used routes, habits, frequented places. So, it can make assumptions about the client's lifestyle and thus target other similar profiles, who are not users yet, exploiting this knowledge.

- marketing campaign management → the CRM can guess the communication campaign efficiency by analysing registrations and interaction with socials and website. Integrating this information, in some cases, the CRM can also suggest how to change the approach for the promotion.

10.1.2 CRM Cycle

Below there is a diagram that outlines some important metrics or categories for each stage of the CRM cycle. It is important to specify well two sentences of this scheme.
In the first phase, it is important to underline that the last category, the one made up of the elderly, could be a bit borderline, however over time even the elderly generation will be able to use smartphones properly and therefore will become a possible target.
In the last step, the phrase "UI metrics" indicates a group of UI-related metrics such as the time it takes to click on a certain position on the screen or to complete a booking or report request.



10.2 Recommender System

As suggested in the introduction of the section 10, the recommender system will lead more costs than benefit. For this reason, it won't be inserted in the scheme of the information system.

In case of development, it will use the user application's data and it will be directly linked to the Management Information System.

Given the nature of the project, the recommendations useful for the user could be a limited set. The most effective one is on an individual level.

When the user selects a vehicle and presses the booking button and if the vehicle prevision probably leads to a worse user experience, the system will suggest another available vehicle in the same line after the chosen one in terms of arrival time. To do that, the RS is configured as a Context-Aware type, where the base ICM (Item-Content Matrix) will be formed by the vehicles and the "availability of the seat" or "the crowdedness" will be the characteristics. The context will be divided into:

- Days of the week/month
- Time slot

The main difference from a classical Context-Aware is that the data will change, indeed the two "features" (not in the strict sense of the RS literature) aren't static, but they can evolve. To clarify better, the vehicles' number can't be used to identify the rows, otherwise, the switching of them could generate a large amount of forecast/suggestion errors.

Considering a multi-variable level, the possible suggestion could be the best path in order to go from a certain initial PT stop to a final one. However, this solution will rise up the complexity and the cost, trying to make better than an existent application (Google Maps) that can use a bigger amount of data.

In substance, the recommender system will suggest only which vehicle fits the user need given a time range, so possible departure to maximum arrive, and given a line's path, so initial and final stop.

11. Data Driven Decisions

11.1 Data Features

The solution that was designed is aimed at solving organizational problems that people with disabilities have when taking public transportation.

When analyzing in detail the actual numbers of this group, we have that the nearly 3 million people in Italy suffer from disabilities, around 5,2% of the entire population.

Of this number, 123.000 resides in Lombardy, being 4,1% of the total, and less than 4% are in Piedmont. Only 14,4% of people with disabilities makes frequent use of public transportation, against the 25,5% of the rest of the population.

By analyzing the schema of our solution and the processes that are to be carried out, four primary data sources have been identified:

1. Booking requests made by a user on the application.
2. Reports made by a user on the application.
3. Information provided by the driver of each PT's vehicle, like updates of the current capacity.

Considering the input data that is to be managed, the variety of the data that our solution will ingest, store, and analyze is also limited.

The data stream related to the first source is influenced only by people with disabilities and the limited number of drivers, as the others cannot book a reserved seat, so its volume will be relatively low at the start, and then it will plateau after the application has spread enough.

On the other hand, the second input stream is influenced by any type of user, as there are no requirements to submit a report. Considering the stability and expertise of the PT companies, the total volume of reports should be fairly low, with a decreasing trend as time passes.

For the third input stream, only users recognized as drivers are able to provide this kind of data, but they will be provided every time a vehicle reaches a new stop in their determined route, being a data stream with a higher frequency one than the others.

Taking all this into account, we can affirm that our solution:

- Will not have to manage huge amounts of data at start, with a growing trend depending on the overall usage. (*Medium Volume*)
- Will have a limited frequency of ingestion of most of the data. (*Relatively low Velocity*)
- Will have to manage input data represented in a single format. (*Low Variety*)
- Will be able to run analyses and create statistics based on data that has low need to be cleaned or wrangled. (*High Veracity*)
- Will enable companies to tap into an untouched portion of the market, solving the current issues by restructuring the way their services are offered. (*High Value*)

Regarding the type of operations on data that are to be carried out by our solution, both the ingestion and CRUD operations will not be done with neither a *batch approach* nor a *real-time approach*, as there is no need take immediate, real-time decisions.

11.2 NoSQL Choice

To make a decision about the technology to use to store the data and base all the operations on, we took into careful consideration not only the characteristics of our data, but also of our potential clients, the public transportation companies.

- *Graph DBs*: The flexibility and ease in representing relationships between entities of this technology would not be too useful in this specific business case.
  Also, the 'restriction' of having to use a completely new and rapidly evolving query language like GraphQL, could be an issue for technicians, that would have to learn and manage a new structure, unsimilar to others they might have used.
- *Columnar DBs:* In this business case, the majority of operations on the database would be inserts, which are not the strongest point of this kind of technology. Also, when evaluating the type and frequency of queries that could be carried out, there is no urgent need for an approach that aims at maximizing speed.
- *Key-Value DBs*: This technology is highly scalable and is optimized for data with single key and value, but a parser is required to store multiple values and is not as performant for lookup operations, as it requires scanning the whole collection or creating separate index values.

In the end, our choice was to implement a *document-based* NoSQL database, namely **MongoDB**, due to some of its characteristics that we found crucial to best valorize our data and carry out operations as easily as possible.

The deciding factors were:

- The flexibility in adding or removing a field even after the creation of a document (entity)
- The ease in scaling a model based on this technology
- The possibility of interlinking documents by adding a reference within the schema
- The Open-Source nature of the technology

The only constraints are on the single size of a document being 16 MB at most and not having a support for transactions. Considering the structure of our entities and our operations, we judged these factors be acceptable.

11.3 E.R. Conceptual Data Model

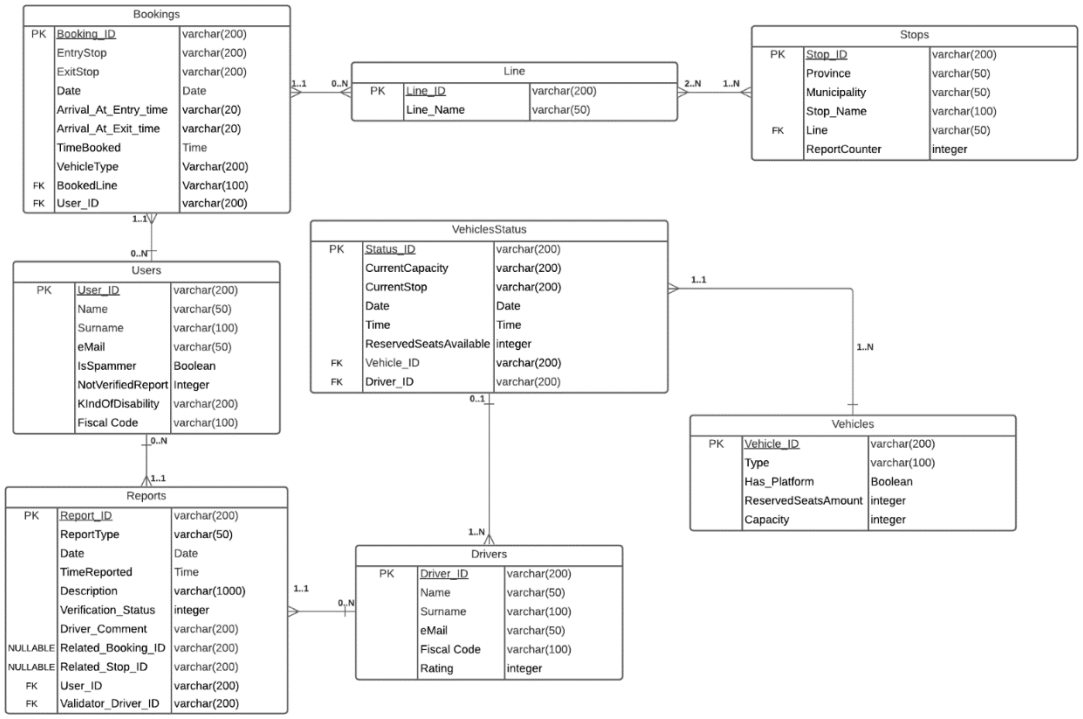
The following schema shows the general structure of the data entities that our application creates and manages to operate and carry out the established functionalities.

As it still is in the initial implementation phase, the structure of the listed entities represents only a general interpretation

As this structure is created on the system of the PT companies that bought our solution, only what were determined being the most relevant entities for companies are listed below.

- The User entity contains information about the registered users that ever booked a seat on a vehicle belonging to that specific PT company.
- The Booking entity stores information about the all the bookings made by users.
- The Report entity contains information about all the reports that users have made about an issue with a past booking or accessibility issues. For this reason, the field that can contain the related booking ID is set to be nullable.
- The Driver and Vehicle entities that contains the list of a company's drivers and vehicles available.
- The VehicleStatus entity, used to keep track of the live status of the vehicles currently running and updated after reaching each stop with new data. As such, it also acts as source of data to run some analysis to create statistics.

It is very important to specify that both the user and the driver can make a reservation (the driver can do it for a user who does not have the app), but the drivers' table is not linked to the reservation table because the request has been made by the latter will be passed as a request by a fictitious user.



11.4 Document Structure

Below are listed the document structures of the entities that are represented in the above E/R model.

A bit of pre-processing is needed before storing the data of a booking or a report, due to the way Mongo natively stores Date and Time, different from what is normally seen as value from these kinds of fields.

i. Users

```
{
    _id: ObjectId('60b34e04688c7523647a3a9e'),
    Name: 'Mark',
    Surname: 'White',
    Age: 41,
    Fiscal_Code: 'WHTMRK80A01F205M',
    Email: 'mark.white80@gmail.com',
    IsSpammer: false,
    NotVerifiedReports: 0,
    KindOfDisability: null
}

{
    _id: ObjectId('60b34e04688c7523647a3a9f'),
    Name: 'Antonio',
    Surname: 'Catalano',
    Age: 31,
```

```
    Fiscal_Code: 'CTLNNT80E10L219O',
    Email: 'antonio.catalano90@gmail.com',
    IsSpammer: true,
    NotVerifiedReports: 5,
    KindOfDisability: null
}
```

ii. Drivers

```
{
    _id: ObjectId('60b34e04688c7523647a3aa0'),
    Rating: 4,
    Related_User_Account: ObjectId('60b34e04688c7523647a3a9e')
}
```

iii. Bookings

```
{
    _id: ObjectId('60b3604aa6cdbd0008d11683'),
    Entry_Stop: 'Loreto M1 M2',
    Exit_Stop: 'Via Padova Via Pasteur',
    Date: '22/05/2021',
    Arrival_At_Entry_Time: '18:05:00',
    Arrival_At_Exit_Time: '18:25:00',
    Confirmed: true,
    Concluded: true,
    VehicleType: 'Autobus',
    BookedLine: '56',
    Booking_User_ID: '57506d62f57802807471dd41',
    Vehicle_ID: '60b36006a6cdbd0008d11681'
}
```

iv. Reports

```
{
    _id: ObjectId('60b36387a6cdbd0008d11685'),
    ReportType: 'Elevation platform missing/broken',
    Date: '31/05/2021',
    Time_Reported: '18:35:00',
    Description: 'The bus\' elevation platform was broken and I couldn\'t get onboard',
    Driver_Comment: 'The platform was functional, but the passenger refused to get on it',
    Related_Booking_ID: '5a934e000102030405000000',
    Verification_Status: 1,
    Verification_Date: '23/05/2021',
    Reporting_User_ID: '57506d62f57802807471dd41',
    Vehicle_ID: '5a934e000102030405000003',
```

```
    Driver_Validating_ID: '507f1f77bcf86cd799439011'
}
```

v. Vehicles

```
{
    _id: ObjectId('60b36006a6cdbd0008d11681'),
    Type: 'Autobus',
    Has_Platform: true,
    Reserved_Seats_Amount: 2,
    Capacity: 30,
    VehicleLine: '56'
}
```

vi. VehiclesStatus

```
{
    _id: ObjectId('60b36440a6cdbd0008d11687'),
    Vehicle_Status_ID: '507f191e810c19729de860ea',
    Vehicle_ID: '5a934e000102030405000003',
    Date: '22/05/2021',
    Status_Time: '18:30:00',
    Current_Capacity: 'Yellow',
    Current_Stop: '60b35ee2a6cdbd0008d1167e',
    Reserved_Seats_Available: 1,
    Current_Driver_ID: '507f1f77bcf86cd799439011'
}
```

vii. Lines

```
{
    _id: ObjectId('60b35ee2a6cdbd0008d1167c'),
    Line_Name: '56',
    Stops: [
        {
            _id: ObjectId('60b35ee2a6cdbd0008d1167d'),
            Stop_Name: 'Loreto M1 M2',
            Province: 'Milan',
            Municipality: 'Milan',
            Report_Counter: 1
        },
        {
            _id: ObjectId('60b35ee2a6cdbd0008d1167e'),
            Stop_Name: 'P.le Loreto Via Padova',
            Province: 'Milan',
            Municipality: 'Milan',
```

```
        Report_Counter: 0
      },
      {
        _id: ObjectId('60b35ee2a6cdbd0008d1167f'),
        Stop_Name: 'Via Padova Via Pasteur',
        Province: 'Milan',
        Municipality: 'Milan',
        Report_Counter: 0
      }
    ]
}
```

One point that deserves a bit more attention is the Driver's schema: due to being 'special users', hence sharing the majority of their fields, we decided to separate the two and put a reference of the related User account in the Driver's schema.

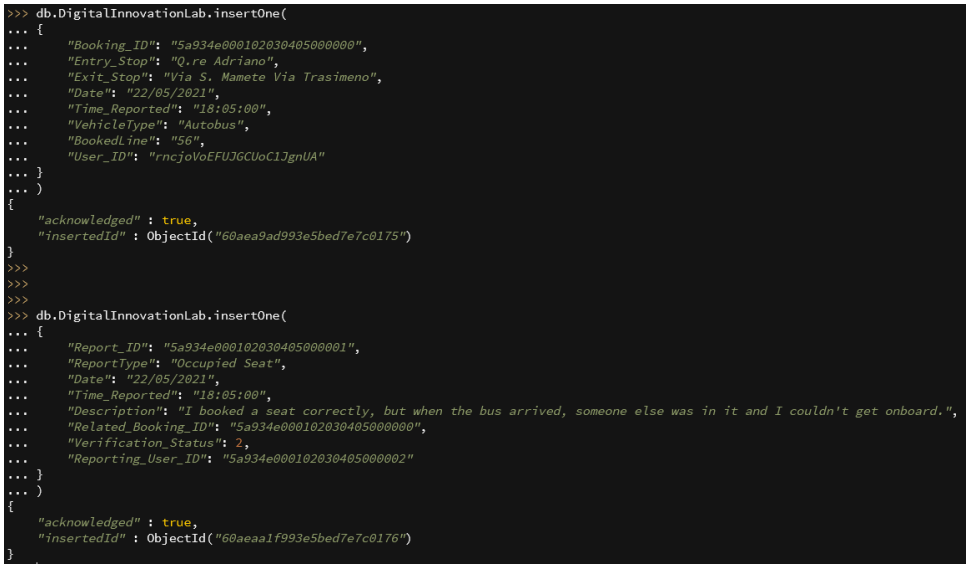
One other factor that supported this decision, is how all the queries that might use the Drivers' rating as a parameter, now do not have to access the linked document, as the information is ready to be used.

11.5 MongoDB Queries

Based on the documents' structure defined above, here are the command used on a MongoDB system to create the collections and insert the documents in them.

```
MongoDB Web Shell

>>> use DigitalInnovationLab;
switched to db DigitalInnovationLab
>>> db.createCollection("Users");
...
... db.createCollection("Drivers");
...
... db.createCollection("Vehicles");
...
... db.createCollection("VehicleStatus");
...
... db.createCollection("Bookings");
...
... db.createCollection("Reports");
{ "ok" : 1 }
>>>
>>> show collections;
Bookings
Drivers
Reports
Users
VehicleStatus
Vehicles
```



Also, here are some interesting the queries that could potentially be run against the database:

- **Find all the vehicles of a certain line that have an elevation platform and currently have any reserved seat available**
    - *db.Vehicles.find({ VehicleLine: "56" ,Has_Platform: true, Vehicle_ID: db.VehicleStatus.find({Reserved_Seats_Available: { $gt: 0 }, Date: {$eq: new Date("2021-06-02T00:00:00.000Z")} }).Vehicle_ID});*

- **Find all the Users that are currently classified as spammers with at least 5 reports Users that have or have 3 or more reports**
    - *db.User.find({ NotVerifiedReports: { $gte: 3 }, $or: [{ IsSpammer: true }, { NotVerifiedReports: { $gte: 3 } }] });*

12. Prototype

12.1 System Architecture

The implementation of the proposed solution goes over all the steps of the process where a user sends a report to our application, checks on its status and a driver validates the report's result.
The architecture of our system is composed of three fundamental technologies:

- Docker engine
- MongoDB with the MongoExpress GUI
- Spring Boot framework

12.2 Docker Engine

Docker is an open-source platform for building, deploying, and managing containerized applications. A Docker container is a lightweight, standalone, executable unit of software that includes everything needed to run an application, including libraries, system tools, code, and runtime.

Both our MongoDB and the attached GUI MongoExpress run inside the Docker ecosystem as containers. The reason behind choosing to implement these services as containers is due to some advantages of this approach, as:

- Ease in Docker installation
- Pre-made images for the desired services
- Consistent behavior across different machines
- Possibility of sharing data by using volumes

These reasons made the container approach optimal for our use-case.

12.2.1 Docker Containers

Container Image Name	Version
mongo	4.4.6
mongo-express	0.54.0

12.2.2 Docker Containers Configuration

Inside the docker-compose.yml file are contained the instances of the MongoDB services and its GUI, with the environment variables and all other properties that characterize their behaviour.

MongoDB Configuration	
Container Name	mongodb
Restart Option	always
Hostname	mongodb
Data Directory	./DIL-project/mongodb/data:/data/db - ./DIL-project/mongodb/conf:/data/configdb
Port Mapping	27017:27017
MONGO_INITDB_ROOT_USERNAME	dil
MONGO_INITDB_ROOT_PASSWORD	dil
MONGO_INITDB_DATABASE	admin
Network	mongo-compose-network

MongoExpress Configuration	
Container Name	mongo-express
Restart Option	always
Hostname	mongo-express
Port Mapping	8081:8081
ME_CONFIG_BASICAUTH_USERNAME	test
ME_CONFIG_BASICAUTH_PASSWORD	test

ME_CONFIG_MONGODB_ADMINUSERNAME	dil
ME_CONFIG_MONGODB_ADMINPASSWORD	dil
ME_CONFIG_MONGODB_SERVER	mongodb
ME_CONFIG_OPTIONS_EDITORTHEME	ambiance
ME_CONFIG_REQUEST_SIZE	100kb
Network	mongo-compose-network
Depends On	mongodb

12.2.3 Docker Commands

The services are started and stopped by running the following commands:

1. *docker-compose up*
   - To start all the services contained in that Yaml file.
2. *docker-compose down*
   - To stop all the services contained in that Yaml file.

12.3 MongoDB

MongoDB is a document-oriented NoSQL database used for high volume data storage that, instead of tables and rows, makes use of collections and documents. Documents are store in BSON format and consist of key-value pairs which are the basic unit of data in MongoDB, very similar to the well-known JSON format. Collections contain sets of documents and function which is the equivalent of relational database tables.

Our choice was to implement a *document-based* NoSQL database was due to some of its characteristics that we found crucial to best valorize our data and carry out operations as easily as possible.

The deciding factors were:

- The flexibility in adding or removing a field even after the creation of a document (entity)
- The ease in scaling a model based on this technology

- The possibility of interlinking documents by adding a reference within the schema
- The Open-Source nature of the technology

12.4 MongoExpress

MongoExpress is a web-based MongoDB admin interface written in Node.js, Express.js, and Bootstrap3. While not being a component that carries out any important task in our system, the implementation of a web-based GUI made it easier to conduct our tests and check whatever the functions behaved as expected, due to its visual nature.

12.5 REST API

Since we want to let the client and the server talk through the internet, we adopted one of the most used and spread approach: the REST APIs.

REpresentational State Transfer (REST) is an approach to Web Services design inspired by the architectural principles typical of the Web. It focuses on the description of the resources, on how to identify them on the Web, and on how to transfer them from one machine to another. It can be also seen as a software architectural style which uses a subset of HTTP. However, “architectural style” is not entirely appropriate. Since REST defines a detailed set of architectural principles, it can be considered as a real standard. These pillars on which the standard is based on are:

1. Resource identification (through the endpoint addresses)
2. Use of HTTP verbs, methods, and response codes
3. Auto-descriptive resources
4. Links between resources
5. Stateless communication

With these APIs, we were able to make both the clients (i.e., driver and user mobile apps) and the server communicate through HTTP methods by exchanging messages written in JSON.

12.6 Spring Boot

The Spring Framework is an application framework and inversion of control container for the Java platform. The framework's core features can be used by any Java application, but there are extensions for building web applications on top of the Java EE (Enterprise Edition) platform.

Having few days for the implementation and the goal to produce a full report process, we decided to adopt the Springboot framework.

The key advantages are:

- Reduction of time spent in development
- Possibility to create a component architecture with a managed container
- Management of the HTTP request
- Possibility to build REST APIs
- Connection with NoSQL database, especially with the MongoDb through the library mongo.client
- Possibility to expand easily the current prototype, also in term of security through Spring Security

12.7 Implementation

In the prototype we provided:

- a CLI for the User application
- a CLI for the Driver application
- Backend server for the reporting process
- MongoDB database contained in a Docker container
- MongoDB GUI

The process is described as follow:

1. The user will check his/her bookings
2. He/She will select one of them and then make a report, inserting the report type, a brief comment, and if regarding the entry stop or the exit stop
3. The driver will check if there are any unconfirmed reports
4. After finding one, he/she will decide to confirm it or reject it
5. The user will check the current verification status and finding out what the driver will have said

The video shows the process.

The architecture of the prototype contains two main components: Booking Service and Report Service. Both of them can handle the many requests coming from the requests.

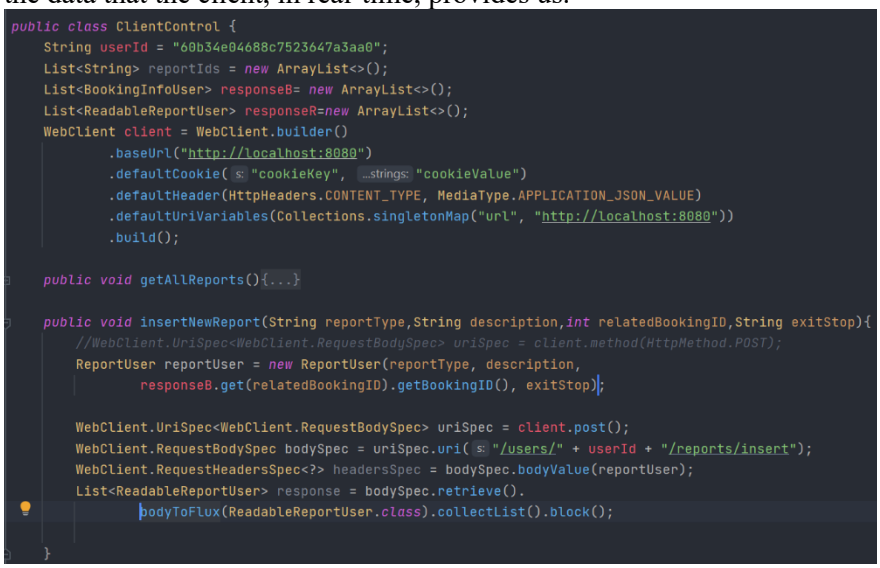
As suggested by the name, the first one is in charge of the registration of the booking, the sending to the correct user and driver. Only the second action is implemented in the prototype.

The Report Service has the responsibility to insert, reject or confirm a report, other than sending it to the corresponding user and driver.

This choice allows the system to manage the requests coming from both user and driver, moreover, the component architecture permits handling different requests.

Here same useful screen of our code:

1. This first screen shows the "ControlClient" class. This is the class in charge of doing all the requests for the client. Here we can see the method "insertNewReport". This method performs a POST request with the data that the client, in real-time, provides us.



```
public class ClientControl {
    String userId = "60b34e04688c7523647a3aa0";
    List<String> reportIds = new ArrayList<>();
    List<BookingInfoUser> responseB= new ArrayList<>();
    List<ReadableReportUser> responseR=new ArrayList<>();
    WebClient client = WebClient.builder()
            .baseUrl("http://localhost:8080")
            .defaultCookie( s: "cookieKey",  ...strings: "cookieValue")
            .defaultHeader(HttpHeaders.CONTENT_TYPE, MediaType.APPLICATION_JSON_VALUE)
            .defaultUriVariables(Collections.singletonMap("url", "http://localhost:8080"))
            .build();

    public void getAllReports(){...}

    public void insertNewReport(String reportType,String description,int relatedBookingID,String exitStop){
        //WebClient.UriSpec<WebClient.RequestBodySpec> uriSpec = client.method(HttpMethod.POST);
        ReportUser reportUser = new ReportUser(reportType, description,
                responseB.get(relatedBookingID).getBookingID(), exitStop);

        WebClient.UriSpec<WebClient.RequestBodySpec> uriSpec = client.post();
        WebClient.RequestBodySpec bodySpec = uriSpec.uri( s: "/users/" + userId + "/reports/insert");
        WebClient.RequestHeadersSpec<?> headersSpec = bodySpec.bodyValue(reportUser);
        List<ReadableReportUser> response = bodySpec.retrieve().
                bodyToFlux(ReadableReportUser.class).collectList().block();

    }
```

2. This screen shows the method "getAllReports". The method performs a GET request asking the server all the reports related to a particular user, this request will be forwarded to Mongo DB that, and the response will be print in the client CLI.

```
public void getAllReports(){
    WebClient.UriSpec<WebClient.RequestBodySpec> uriSpec = client.method(HttpMethod.GET);
    WebClient.RequestBodySpec bodySpec = uriSpec.uri( s: "/users/" + userId + "/reports/receive");
    responseR = bodySpec.retrieve().bodyToFlux(ReadableReportUser.class).collectList().block();
    int j=0;
    for (ReadableReportUser Info: Objects.requireNonNull(responseR)) {
        System.out.println("This is your report number: "+j);
        System.out.println("ReportType: "+Info.getReportType());

        switch (Info.getVerificationStatus()){
            case 0:
                System.out.println("VerificationStatus: WAITING");break;
            case 1:
                System.out.println("VerificationStatus: REJECT");break;
            case 2:
                System.out.println("VerificationStatus: CONFIRMED");break;
        }
        System.out.println("DateReport: "+Info.getDateReport());
        System.out.println("BookedLine: "+Info.getBookedLine());
        j++;
    }
}
```

3. This last screen shows the method "verifyReport". This is the class in charge of doing all the requests for the driver. This method performs a PUT request with the data that the driver, in real-time, provides us. We need a PUT because the driver cannot store another report but only verify the existing one.

```
    private void showFirstDetails(){...}

    public boolean showMoreDetails(int position){...}

    public boolean numberCorrectReport(int position) { return position < readableReportDrivers.size(); }

    public void verifyReport(int result, int report_related_position, String comment){
        ProcessedReport processedReport = new ProcessedReport(readableReportDrivers.get(report_related_position-1)
                .getReportID(),comment,String.valueOf(result));
        System.out.println(processedReport.getVerificationStatus());
        WebClient.UriSpec<WebClient.RequestBodySpec> uriSpec = client.put();
        WebClient.RequestBodySpec bodySpec = uriSpec.uri( s: "/drivers/" + driverId + "/reports/verify");
        WebClient.RequestHeadersSpec<?> headersSpec = bodySpec.bodyValue(processedReport);
        headersSpec.retrieve().bodyToMono(String.class).block();
    }


}
```

[GitHub Repository]