1 INTRODUCTION

1.1 Purpose

This document aims to serve as a basis for the organization of the project that will lead to the development of SafeStreets by presenting the requirements. Secondly, once the product has been completed, it will be used to evaluate the final result and judge its loyalty with respect to what is set in these pages. In conclusion, the document is intended for those who actually will contribute to the creation of the software (Project Managers, System and Requirement Analysts, Developers, Testers and so on) but also to users who will benefit from it.

1.2 Scope

1.2.1 Description of the given problem

Nowadays the number of cars on the streets of our cities is constantly increasing and for this reason it is necessary to find new solutions in order to guarantee that they can be parked legally. Nevertheless, it often happens, due to lack of parking zones or laziness of drivers, to find cars parked in an unacceptable way and the situation sometimes becomes difficult to manage by the authorities. In this context SafeStreets operates: the software permits to share with authorities photos of parking violations but this is not all. In addition, SafeStreets mixes the information given by its users with some data computed by the system in order to have a complete description of the violation, in fact the report is accompanied by the licence plate, the location, the date and the type of violation. The retrieval of the first two data is managed by the software while the third is entered manually by the user and the result is saved in the database and can be visualized by municipality.

Moreover, the product has another interesting feature that gives the opportunity to users to navigate through data stored anonymously in the system visualizing correlations between places and the number of violations through maps, diagrams and other representations.

Obviously not everything is accessible by anyone but in some cases permission is required to access certain data.

In conclusion there is a very innovative function, possible thanks to the collaboration of the municipality, that allows SafeStreets to propose possible solutions to solve problems due to repeated violations that occur in certain areas of the city that for these reasons are considered dangerous.

1.2.2 Goals

- G1: Users and municipality must be able to visualize on a map the zones where violations occur more frequently.
- G2: Users and municipality must be able to visualize statistics regarding violations.
- G3: The municipality must be able to confirm or mark the reports as fake
- G4: Municipality should receive solutions that suggest how to solve the problems of violations in problematic zones.
- G5: Users should be able to report violations anonymously.
- G6: Municipality should be able to have access to all the anonymous segnalations sent to SafeStreets.
- G7: Each user should be able to visualize the list of segnalations sent by him.

1.2.3 The world, the machine and shared phenomena

World phenomena

- Collection of municipality data into database
- Traffic violations
- A user encounters a parking violation

Machine phenomena

- Traffic plate recognition
- Position retrieval from GPS
- Collection of data into database
- Data anonymization
- Retrieval of date and time of the segnalation
- Processing of data to be presented to users

Shared phenomena

- Violation reported to SafeStreets using the mobile app via a picture (W)
- Data mining and visualization by users (M)
- Suggestions of interventions (M)
- Linking SafeStreets and municipality data (M)

The reader can find between brackets a letter “M” if the shared phenomena is controlled by the machine and observed by the world and a letter “W” in the opposite situation.

1.3 Definitions, Acronyms, Abbreviations

1.3.1 Definitions

- Driver: the one who makes the violation.
- User: who benefits from the service offered by SafeStreets by sending photos of the violations or accessing the data saved in the system.
- Municipality: public authority who take advantage of the service offered by SafeStreets and eventually collaborate with it providing supplementary information and receiving possible solutions to reduce the number of violations.
- Segnalation: report sent by a user. It is composed by some elements like a photo in which the violation figures, the type of violation, date, time and the street where the violation occurred.
- Authority: used as a synonym for municipality.
- Violation: corresponds to parking in an illegal way, there are many types of different parking violations: parking where it's not permitted (on a crosswalk, near intersections, on sidewalks, on bus stops, on a handicapped zone without permissions, parking in time slots where it is not allowed to stop on that specific street, double parking,).
- Solution: report sent by SafeStreets to the municipality after an accurate evaluation of violations occurred in a certain area. Its aim is to reduce the number of these types of events.
- Zone: the basic unit in which it is possible to divide a city.
- Anonymous segnalation: segnalation saved in the database which presents no reference to the specific user who sent the same notification when the municipality visualizes it.
- Statistic data: representations of the data that SafeSteets stores. Users can visualize them.
- Coordinates: latitude and longitude of the position in which the user takes the photo that will attach to the report.
- Solution table: It corresponds to a map that assigns to each possible parking violation a solution to propose. It is regularly updated because the system evaluates the impact that certain solutions have had in specific zones of the city and whether these have actually reduced the number of violations.
- System: This word is used to refer to SafeStreets and its components such as the server that allows communications with users, the database and all the others that allow data to be computed and solutions to be offered to its users.

1.3.2 Acronyms

- RASD: Requirement Analysis and Specification Document
- DB: DataBase
- GPS: Global Positioning System
- OCR: Optical Character Recognition

1.3.3 Abbreviations

- Gn: nth goal
- Rn: nth requirement
- Dn: nth domain assumption

1.4 REFERENCE DOCUMENTS

- Specification document: “SafeStreets Mandatory Project Assignment”
- Alloy documentation: http://alloy.lcs.mit.edu/alloy/documentation.html
- 830-1993 - IEEE Recommended Practice for Software Requirements Specifications: https://ieeexplore.ieee.org/document/392555

1.5 DOCUMENT STRUCTURE

This RASD is divided into five chapters and each one of these have a specific function that is briefly described below:
The first chapter gives an overview of the problem indicating what the document proposes to do describing in principle the operation of the software, listing the goals to be reached and finally clarifying some details that allow a better understanding. This is meant to be the entry point for developers and other technical people, while users can match their expectations with the presented goals.
The second chapter explains in an accurate way important details that permit to understand better the overall functioning of the system. Here the reader can also find the descriptions of the main functions of the software, the actors and the list of assumptions, dependencies and constraints to be considered. The latter ones serve as limitations for the developer’s options.
The third chapter contains very important details for the development team, in fact it describes the functioning of all the interfaces of the system. Moreover, it proceeds by defining the functional requirements using the use case and sequence diagrams. Finally, non-functional requirements are listed through Performance Requirements, Design Constraints and Software System Attributes.

In the fourth chapter the results obtained through the formal analysis using Alloy are reported but first the modeling is justified specifying what it demonstrates and why it is important for this specific problem.
Finally, the fifth chapter shows schematically the effort spent for the realization of this document.

2 OVERALL DESCRIPTION

2.1 Product perspective

The product is not totally independent but it does not rely on any previous structure. It just cooperates with municipality when it shares with SafeStreets its data regarding accidents and solutions adopted in the past.

Moreover, SafeStreets exploits the services offered by an API which permits to retrieve, using GPS coordinates, the name of the street from where the user is sending the segnalation.

In addition, the system uses an OCR API that can transform the licence plate, which captured in the photo taken by the user, in a string easily manageable.

The service offered by SafeStreet is designed to be accessible both from a mobile app and through a website. The only difference is that from the smartphone the user can either send reports or consult data while in the web version only the latter option will be accessible.

In order to have a better view of the system as a whole, the following is a very simple class diagram that represents a model of the application domain. Note that the classes do not report all the attributes that characterize them but only the main ones that favor understanding.

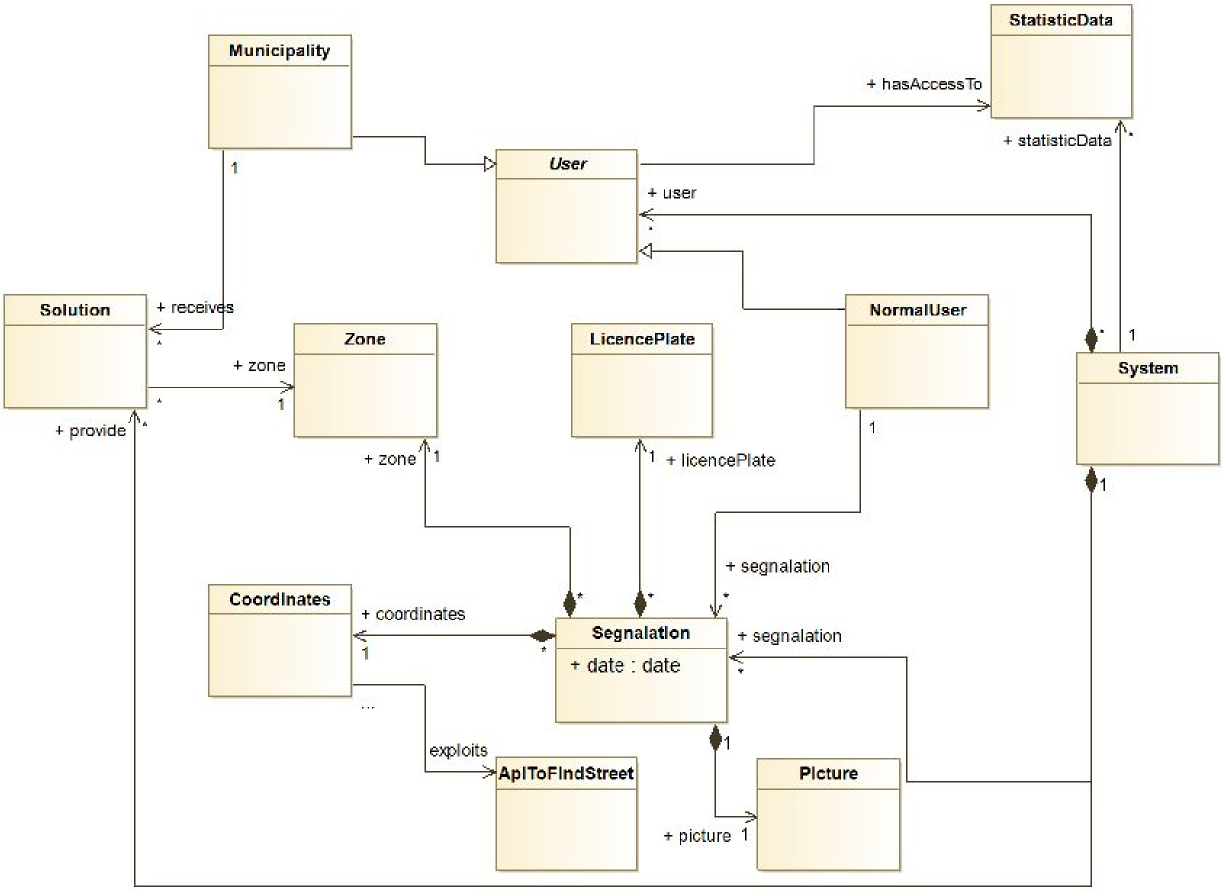


Figure 1 - Class diagram

In the class diagram over here the class “StatisticData” represents also the maps that can be visualized by users. On the other hand here it’s represented the fact that different users have different permissions over data; Class diagrams in fact are not the right tool to show these kind of relations but how users can access data will be clarified in other sections of this document.

Below the reader can find some state diagrams which show how states change in particular operations that are really important for the system.

The first diagram represents the various states of a segnalation while the user fulfil its mandatory requests and SafeStreets compute the missing parts.

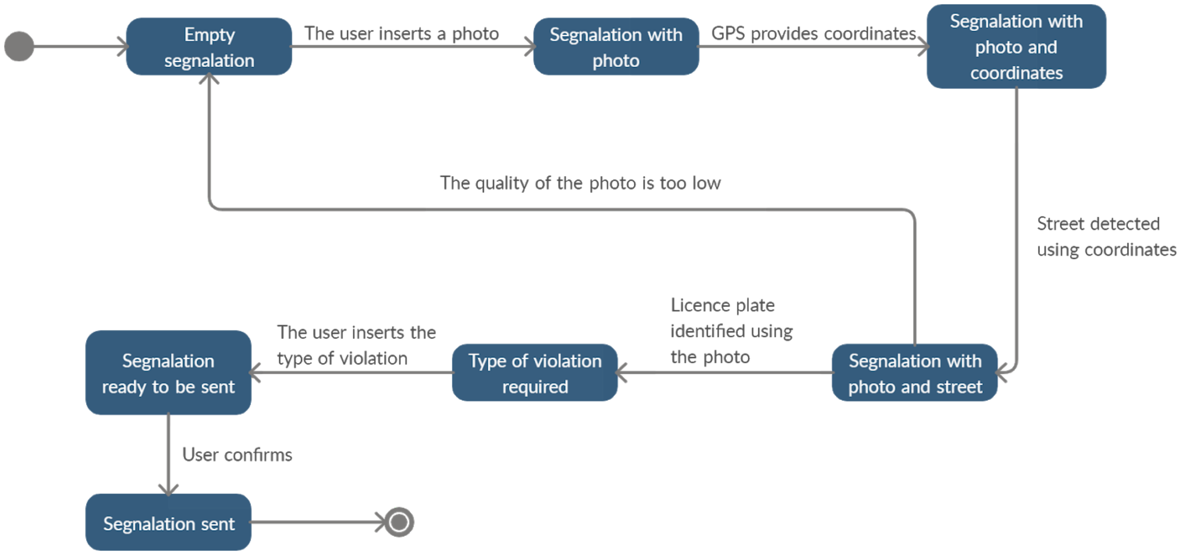

Empty segnalation
The user inserts a photo
Segnalation with photo
GPS provides coordinates
Segnalation with photo and coordinates
The quality of the photo is too low
Street detected using coordinates
Licence plate identified using the photo
Segnalation with photo and street
Type of violation required
The user inserts the type of violation
Segnalation ready to be sent
User confirms
Segnalation sent

Figure 2 - First state diagram

The second one proceeds in the description of the flow of the state of the segnalation once it is received by SafeStreets till the moment in which it is definitely stored in the database.

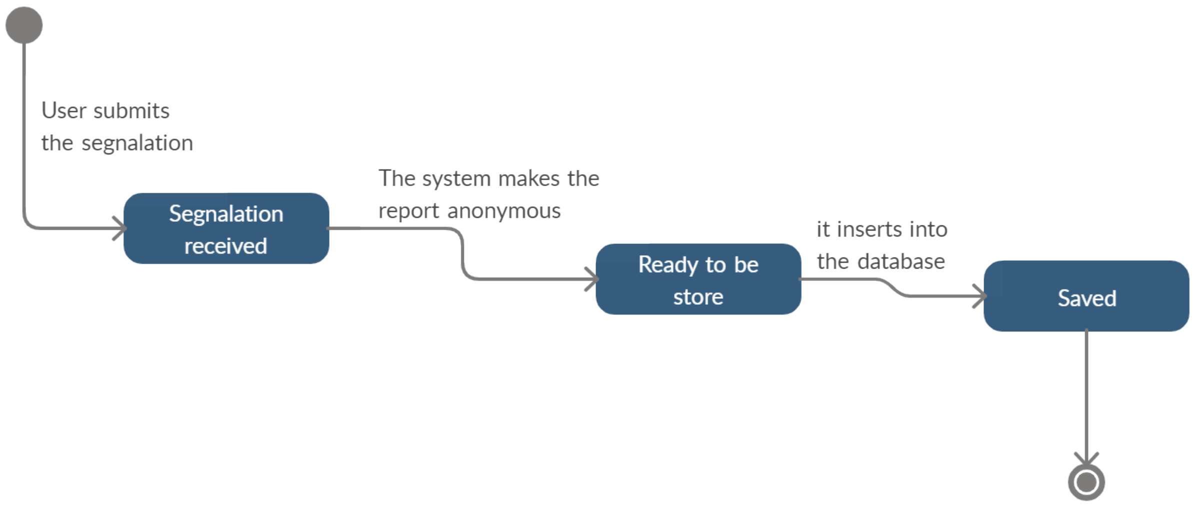

User submits the segnalation
Segnalation received
The system makes the report anonymous
Ready to be store
it inserts into the database
Saved

Figure 3 - Second state diagram

In the third and last case shown by state diagrams the reader can see what happens to a specific zone when SafeStreets receives a large number of segnalations indicating that many violations occur there.

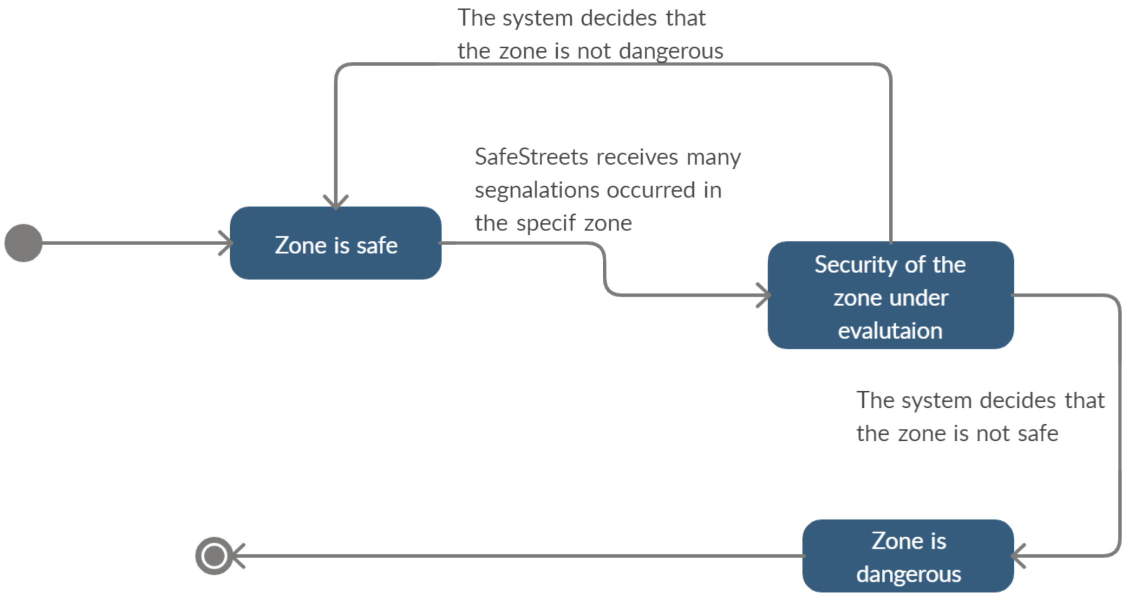


Figure 4 - Third state diagram

2.2 Product Functions

Based on the presented goals, we can identify three product functions that the system is going to provide. It’s worth mentioning that SafeStreets can do its best when the municipality cooperates with it.

2.2.1 Collection of user data

This is the crucial function of SafeStreets: the system lets a user make a segnalation about a violation. The only data the user is asked to enter is the type of the violation and a photo which captures the car parked. The GPS of the device will be used to infer the street where the picture is taken, the licence plate is obtained through an OCR algorithm while the date is obtained by the system. All the segnalations will only store these information and any user sensitive data will not be saved. This ensures that the DB is freely explorable so that everyone with the right permissions can download the dataset and work with it to extract whatever information they care about. SafeStreets must also ensure that no fake data are sent by users, it achieves this by allowing to take pictures only by an embedded camera feature who makes impossible to modify the photo or to upload already existing photos from the gallery.

2.2.2 Visualizing statistics

This product function is in charge of mining the DB in order to provide insightful statistics to better understand how much spread out are certain types of violations. In particular, it will be possible to visualize on a map the zones more prone to parking violations highlighting them with different colors that represent different degrees of frequency. In addition, after a click, performed by the user on one of them, a list of all the violations occurred in that specific zone will be displayed.

Moreover, SafeStreets will provide specific data for each zone that show in which days the number of violations in higher and in which hour of the same days. A ranking concerning the vehicles which committed the most violations will also be available.

2.2.3 Offering solutions

This product function must be implemented if and only if the Municipality allows users to retrieve information about accidents. In this case SafeStreets will cross these data with its own. Exploiting the interventions Municipality applied in the past, the system will recommend possible solutions to fight violations and prevent accidents. Based on the impact that the proposed solution had, SafeStreets will use this feedback to suggest better and better solutions in the future. The procedure that leads to the proposal of a solution to the municipality works in this way: the system realizes that a solution for a certain area must be proposed when certain types of parking violations occur regularly. At this point, SafeStreets actually verifies which are the specific violations that characterize the zone, then consults the solution table and check which solution corresponds to the specific accident in the current state of the table. Finally, prepare the solution to propose to the municipality by attaching the area and the type of accident to which reference is made.

2.3 User Characteristics

Here we better define what the users of the service are.

A portion of the collected data is available to everyone, so each citizen interested on getting a grasp of the situation of the traffic violations in its territory can enjoy the statistics computed by SafeStreets. However, more active citizens can take part in collecting data by providing segnalations. A particular subclass of user is the one represented by municipality, which in addition can provide its own data in order to get offered solutions to avoid accidents and violations. Moreover, these special users have access to the list of all the segnalations sent by normal user, in this way the can use them in the way they prefer.

Obviously during the registration it is not immediate to sign-up an account with these additional permissions: those who actually represent the authorities or work for them must explicitly make a request to SafeStreets in the registration. They also must specify which is their area of competence, in this way they can have access to the material of their interest and not to data of other cities. Finally, then the system verifies that the person who sent the request actually has the right and if the verification is successful, the user of the municipality finally receives the credentials.

2.4 Assumptions, dependencies and constraints

We assume that the following domain assumptions hold:

- D1: The user’s smartphone GPS will provide a position with an error within 20 meters.
- D2: The picture must be taken with a camera that has at least 5 Mpx to allow the OCR algorithm to retrieve the licence plate.
- D3: The user’s Internet Connection works properly during the upload of the segnalation.
- D4: The data provided by Municipality are correct.
- D5: SafeStreets has a mapping that associates every possible type of violation with a possible solution to fight the specific problem.
- D6: Each user has a specific username that is unique.
- D7: SafeStreets has access to the GPS and camera of the device that is running the app.

SafeStreets rely on two external tools:

1. External library that implements the OCR algorithm, to be chosen by the development team.
2. An API to retrieve the location.

3 SPECIFIC REQUIREMENTS

3.1 External Interface Requirements

3.1.1 User Interfaces

The following mockups give a suggestion on how the mobile and web app should look like.

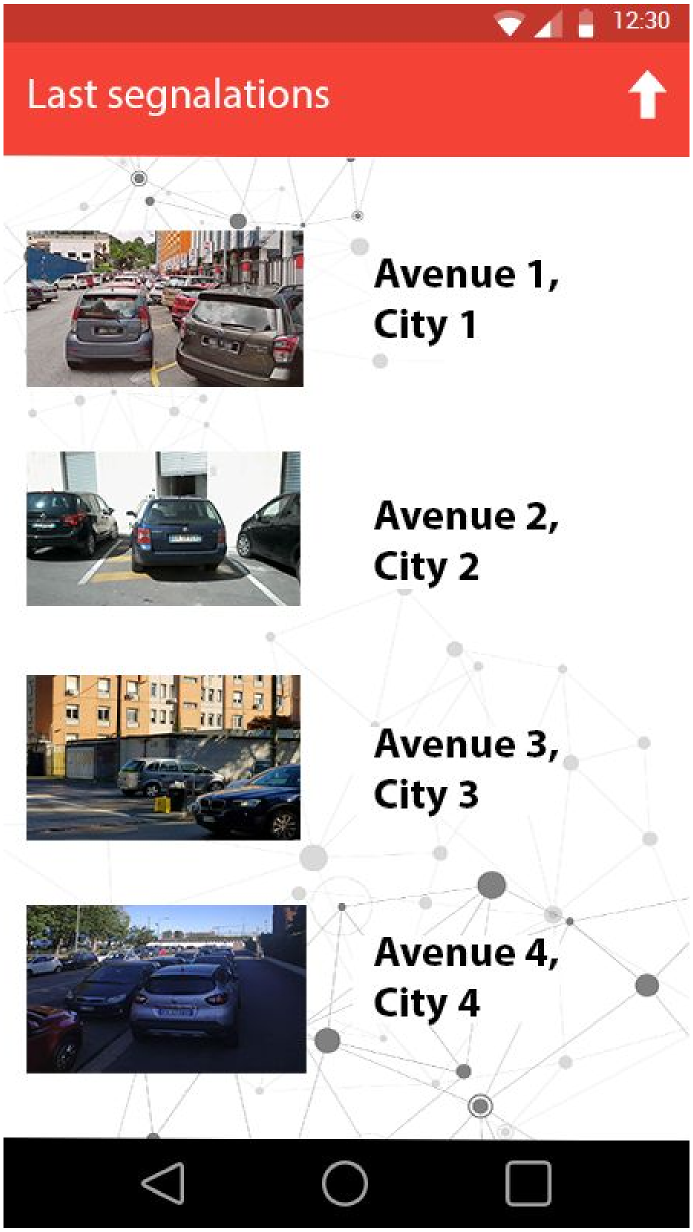


Figure 5 - Home Page

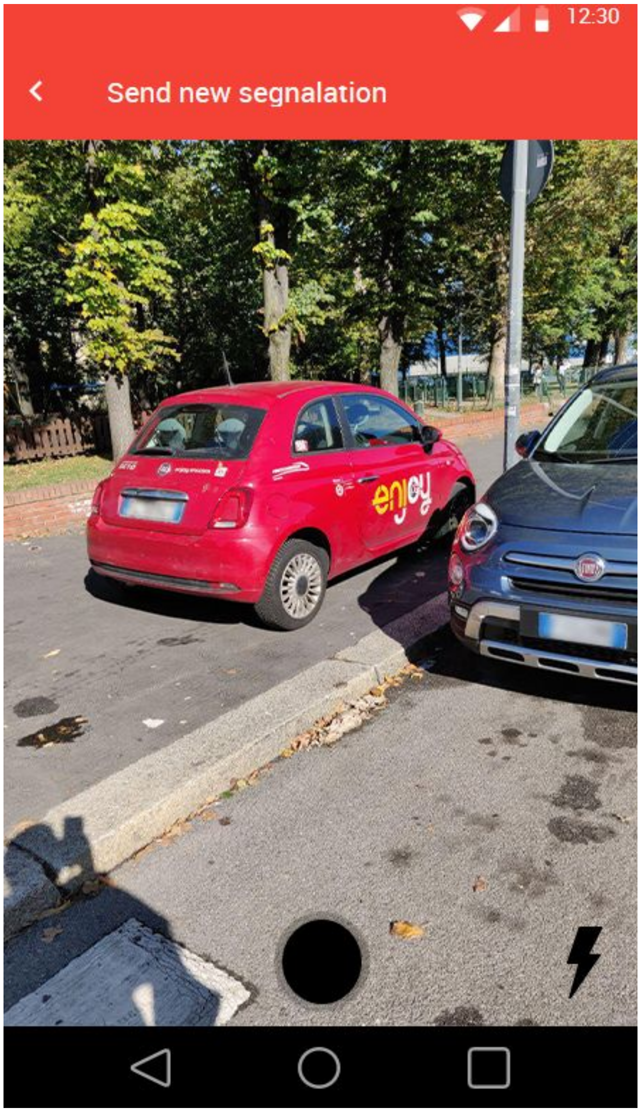


Figure 6 - Segnalation report

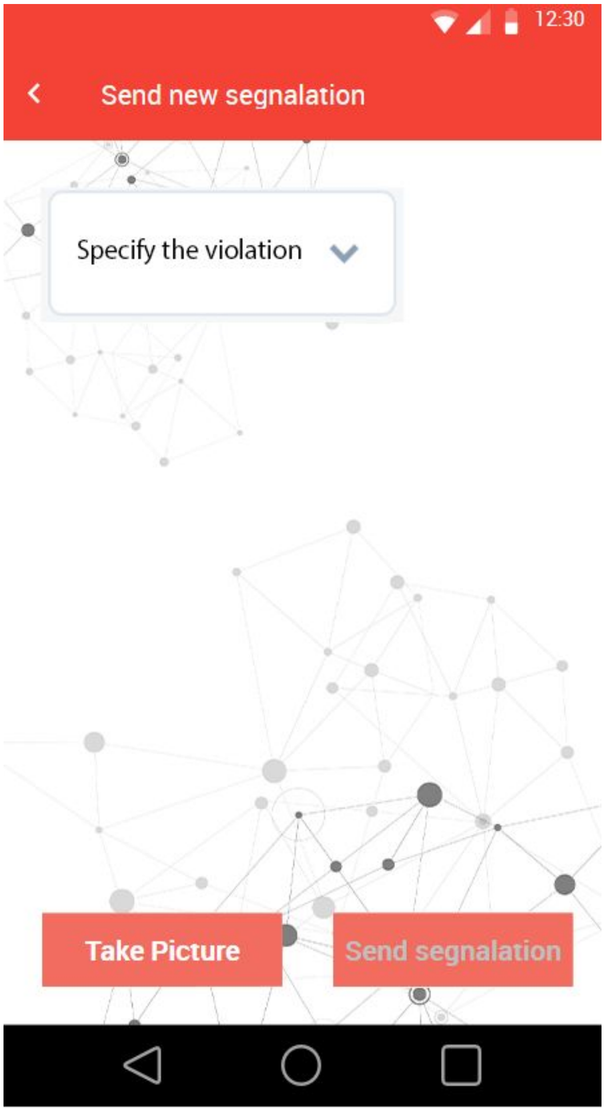


Figure 7 - Type of violation

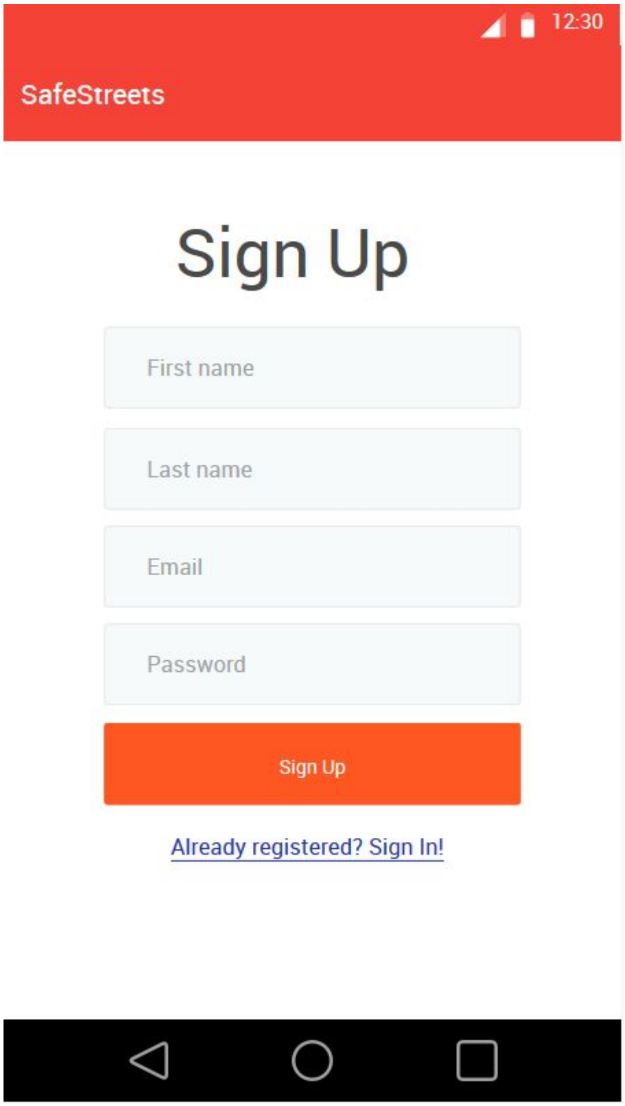


Figure 8 - SignUp

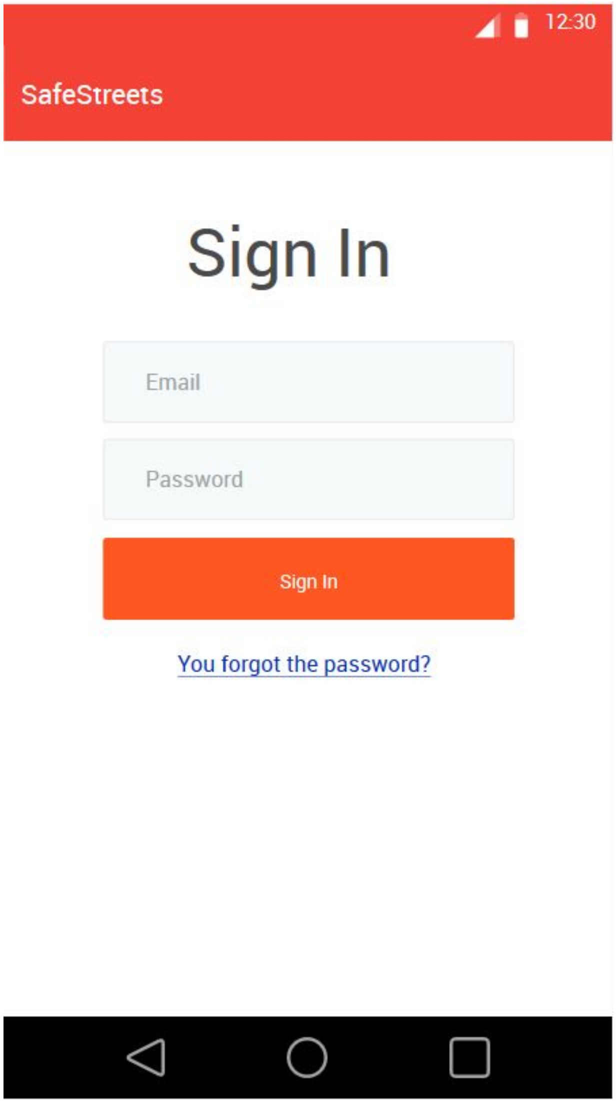


Figure 9 - SignIn

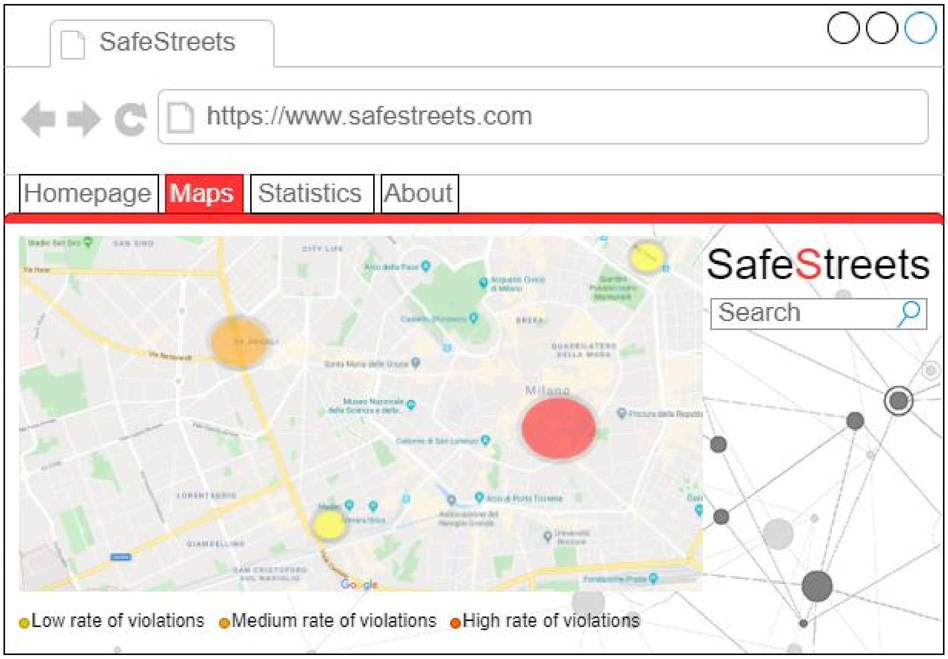
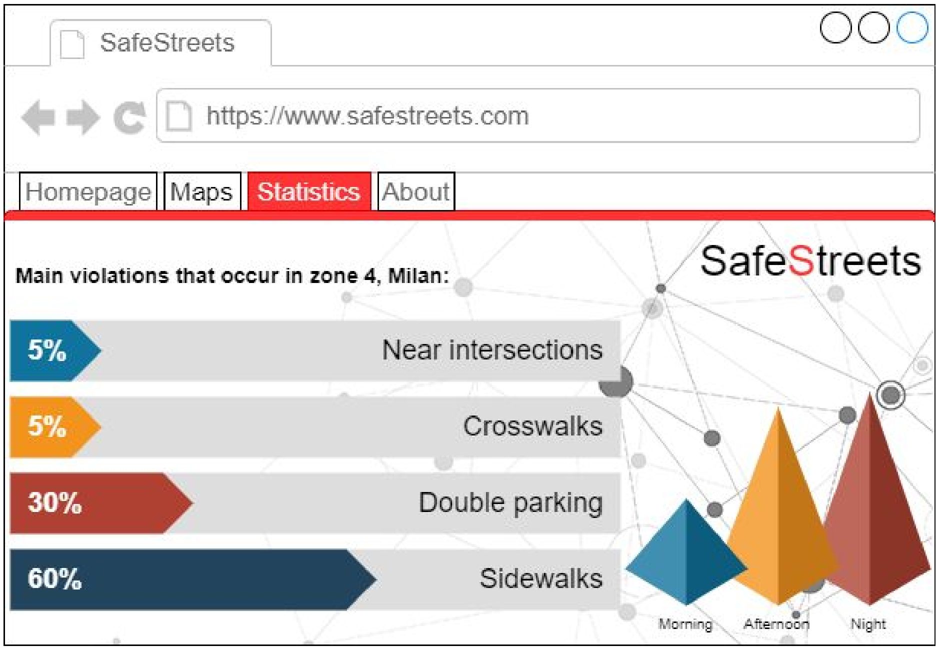


Figure 10 - Web App visualizing a map which represents the rate of violations



Main violations that occur in zone 4, Milan:

5%	Near intersections
5%	Crosswalks
30%	Double parking
60%	Sidewalks

Figure 11 - Web App visualizing statistics

3.1.2 Hardware Interfaces

This application doesn't have hardware interfaces, however it requires to the device of the user to have an internet connection to access and visualize data, and a smartphone equipped with GPS to send segnalations.

3.1.3 Software Interfaces

The application doesn't provide any API to external applications.
It provides access to the DB, though, and it does so according to the permission each user has.
Another API is used to retrieve the location and the name of the street.
An external library will provide the OCR algorithm to read the licence plate.

3.1.4 Communication Interfaces

The system sends and receives data over HTTPS.

3.2 Functional Requirements

3.2.1 Normal user

Scenarios

Scenario 1

Francesco recently obtained the driving license despite his certified disabilities that granted him the right to park in parking lots reserved for disabled people. This is very important for him because walking for long distances is often a problem since he uses crutches. Fortunately, some of the parking lots in front of the building where he lives are reserved for those like him who have the special permission shown on the dashboard of the car, in this way Francesco is able to go home without any problem. But unfortunately his neighbor, despite not having any health problem and absolutely no particular permission, insists on parking his car in the parking lots of the disabled and sometimes because of this Francesco is forced to park elsewhere and this causes him many difficulties. In order to try to solve the problem, Francesco downloaded SafeStreets on his smartphone and whenever this unpleasant situation occurs he sends a report through it hoping that the authorities can intervene. Indeed, because of the numerous reports received for the frequent recurrence of the same violation the authorities decided to intensify controls in his neighborhood and several times they gave a ticket to the rude lady who finally stopped parking in illegal way.

Scenario 2

Federico is a civil engineering student very fascinated by the management of roads within cities and in his free time he likes to study the various choices that administrations adopt to solve numerous problems and sometimes try to send letters to his municipality where he presents cases he studied with several reflections regarding the same ones up to the point of proposing real solutions that. Specifically Federico, to start his research or prove that certain problems are real, uses the maps and statistics that SafeStreets makes available to all registered users. In fact through the first he chooses which areas to focus his interests on, while thanks to the latter he has a clearer vision of what really happens within each area.

Scenario 3

Giuseppina, an old woman with physical disabilities, is used to park her car nearby her house so that she doesn't need to walk much. Unluckily for her, a popular restaurant is just around the corner and it attracts many customers. As a consequence her spot is very often already taken by some non-disabled person. Thanks to SafeStreets she was able to report once for all this violation and now she can again enjoy parking where she was used to.

Use case diagram

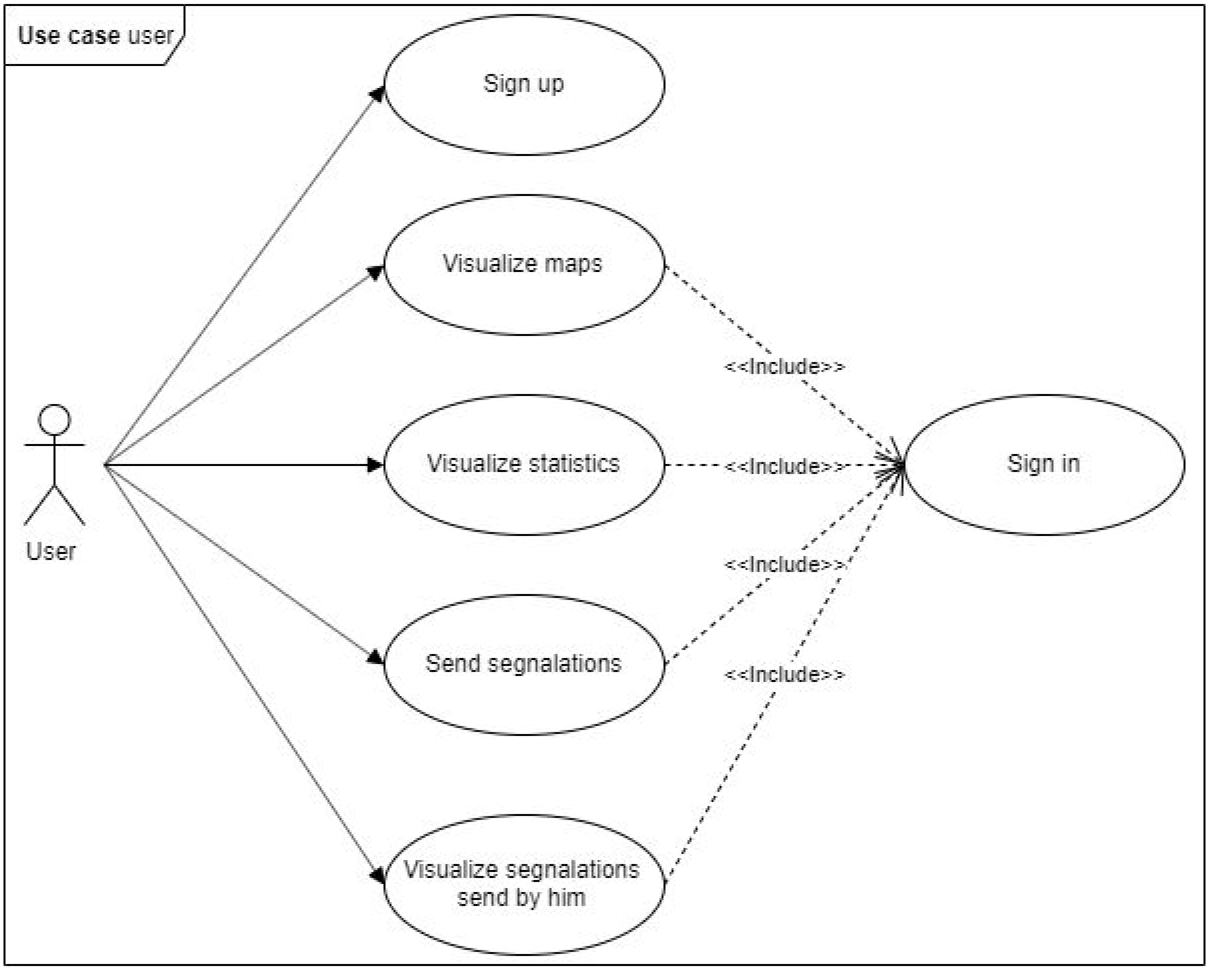


Figure 12 - Use case diagram for user

Use cases

Name	Send new segnalation
Actor	User
Entry condition	The user is already logged in. The user has opened the mobile app and in the home page tapped the upload button in the top right corner.
Events flow	1. The user specify the type of the violation via the combobox. 2. The user presses the “take picture” button. 3. The user take the picture via the SafeStreets integrated camera

	feature. 4. The user confirms his segnalation by pressing the now activated button “send segnalation”. 5. Safestreets analyzes the picture to obtain the licence plate. 6. SafeStreets obtains the location using the GPS. 7. SafeStreets saves the data into the DB together with the current date and time.
Exit condition	The segnalation has been sent successfully to SafeStreets.
Exceptions	1. The OCR algorithm couldn’t infer the licence plate. In this case the user is asked to take the picture again.

Name	Sign Up
Actor	User
Entry condition	The user opens the SafeStreets app or web site for the first time or the user doesn’t have an account.
Events flow	1. The SignUp page shows up. 2. The user chooses if he wants to become a normal user or a municipality user. 3. The user fills all the fields. 4. The user presses the “SignUp” button. 5. If the user had decided to become a municipality user then SafeStreets checks that he has the requirements to become one of them. 6. SafeStreets stores the new user into the DB and send to the user a verification email. 7. The user verifies his account.
Exit condition	The user is registered in the SafeStreets DB.
Exceptions	1. A user with the same email is already registered. SafeStreets will show up

	the “SignIn” page. 2. The user does not verify his account within a week since registration. Then the system removes its profile from the DB.

Name	Sign In
Actor	User
Entry condition	The user has successfully signed up.
Events flow	1. The SignUp page shows up. 2. The user clicks on “Already registered? Sign In!”. 3. The SignIn page shows up. 4. The user fills all the fields. 5. The user presses the “SignIn” button. 6. The HomePage is displayed.
Exit condition	The user is signed in.
Exceptions	1. No match Email/Password has been found in the DB. An error message is shown and the user can try again.

Name	Visualize statistics
Actor	User
Entry condition	The user has opened the SafeStreets App and he is already logged in.
Events flow	1. The user enters in the area where he can visualize statistics. 2. The user chooses which type of statistics he wants to visualize. 3. In the first case, a map is shown with zones highlighted according to the rate of violations characterizing those zones. In the second case the user can see a percentage of the types of violations that most occur. The

	selected statistics show up.
Exit condition	The user successfully views the statistics.
Exceptions	No exceptions

Name	Visualize maps
Actor	User
Entry condition	The user has opened the Web App and he is already logged in.
Events flow	1. The user enters in the area where he can visualize maps. 2. The user chooses the area that he wants to visualize. 3. The requested map shows up.
Exit condition	The user successfully views maps.
Exceptions	SafeStreets can’t find the requested zone and asks the user to insert a different input.

Sequence diagram

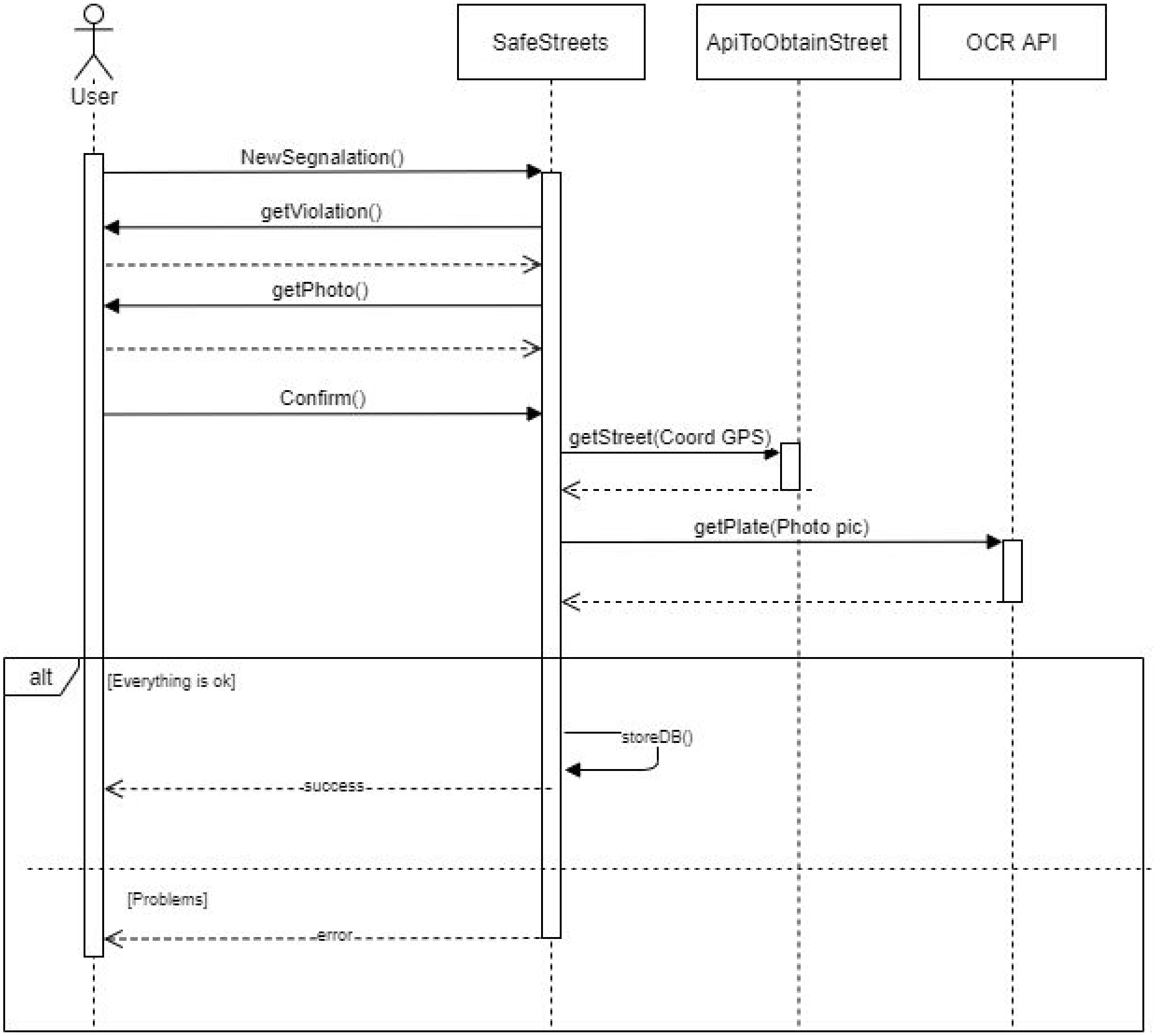


Figure 13 -Sequence diagram: send new segnalation

3.2.2 Municipality user

Scenarios

Scenario 3

Marco works as a traffic policeman in a small town where every Thursday morning there is a market in the main square. It happens that due to this event, many citizens go to the center of the town every week and it often happens that they park outside the parking areas because they are full or too far from the desired destination. Because of what has just been described, SafeStreets receives many reports of parking violations coming from the zones near the city center. Marco on that day then checks the list of reports received from the areas of his competence and hangs around that streets to fine these vehicles and all the others for which the signal has not arrived.

Scenario 4

Monica is an employee of the municipality of her city responsible for the administration of the roads and managing the problems related to it. It has often solved problems due to the occurrence of an excessive number of parking violations in different areas through disparate solutions, notifying each time SafeStreets of the remedy adopted, but this does not always happen since it is not easy to have continuous control of each area of the city. It happens in fact that one day SafeStreets realizes that in the area near the elementary school of the city often occur too many parking violations related to the fact that vehicles park over sidewalks and after checking the table of solutions notifies the municipality advising to install some poles to separate the sidewalk from the street as it has been noticed in the past that this is the most effective solution. Monica, since she has an account with all the additional permissions granted to the members of the municipality, receives this notification and put into practice the proposed solution, effectively obtaining a remarkable decrease in the number of violations.

Use case diagram

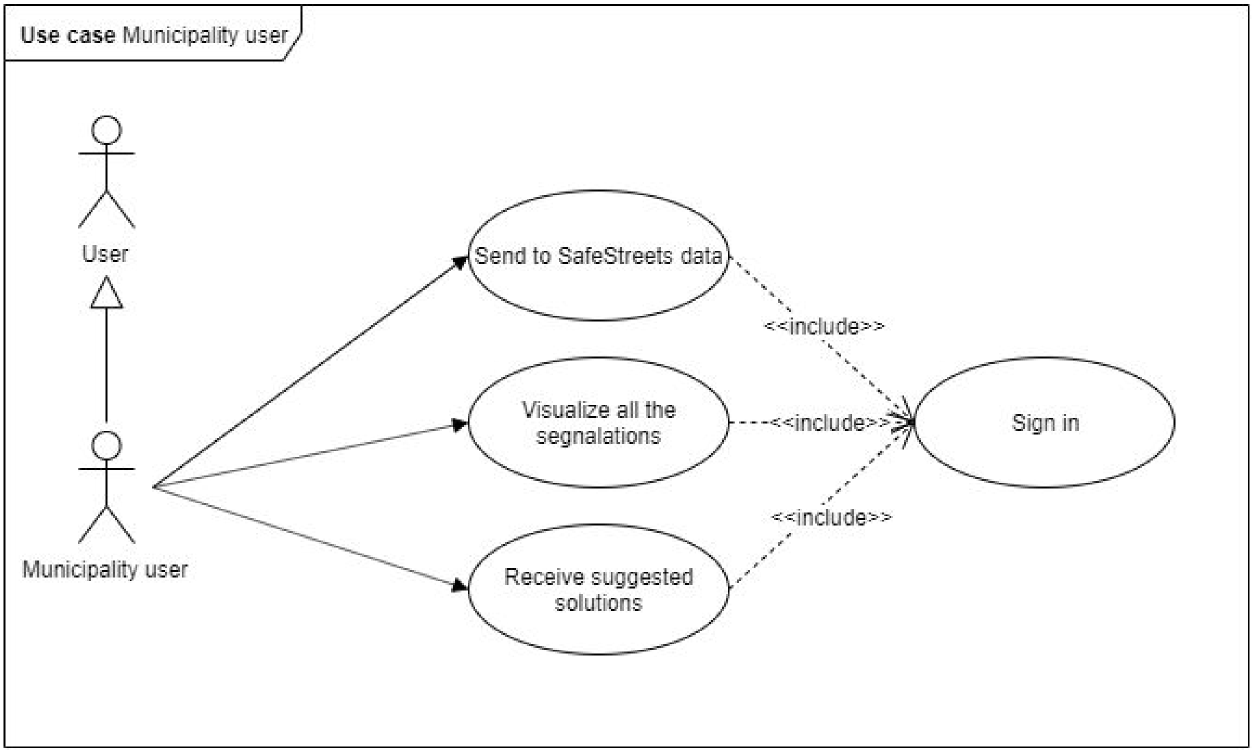


Figure 14 - Use case diagram for municipality user

Use cases

Name	Receive suggested solutions
Actor	Municipality user
Entry condition	SafeStreets recognizes that a particular kind of violation occurs frequently in a given zone.
Events flow	1. SafeStreets checks the solution table looking for that kind of violation. 2. SafeStreets extracts the subset of possible interventions to tackle the problem. 3. SafeStreets sends the possible solutions to the interessed municipality users mentioning the violation and the zone where the solution should be applied. 4. If the Municipality apply the solution, SafeStreets will update its solution

	table based on the feedback the solution had.
Exit condition	The Municipality receives the possible solutions proposed by SafeStreets.
Exceptions	1. SafeStreets couldn't find any solutions to the particular violation in the solution table. The report sent to the Municipality user is solution free.
Special Requirement	SafeStreets must provide its solution to the Municipality within a day from since the problem is recognized.

Name	Visualize all the segnalations
Actor	Municipality user
Entry condition	The user is already logged in.
Events flow	1. The user is in the homepage and chooses to visualize all the segnalations of the zones of his city. 2. A list of all the segnalations is shown. They are sorted by date and zone. 3. The user may want to know more info about a specific segnalation so he clicks on the desiderd one and in this way he can see all the non sensible data of the report.
Exit condition	The Municipality user visualize all the segnalations of its city.
Exceptions	//
Special Requirement	SafeStreets must provide all the segnalations of at least the last 3 months.

There is an exception that is common to all use cases, that is the one in which the user doesn't have a stable internet connection. In this case the web app or mobile app warns you.

Sequence diagram

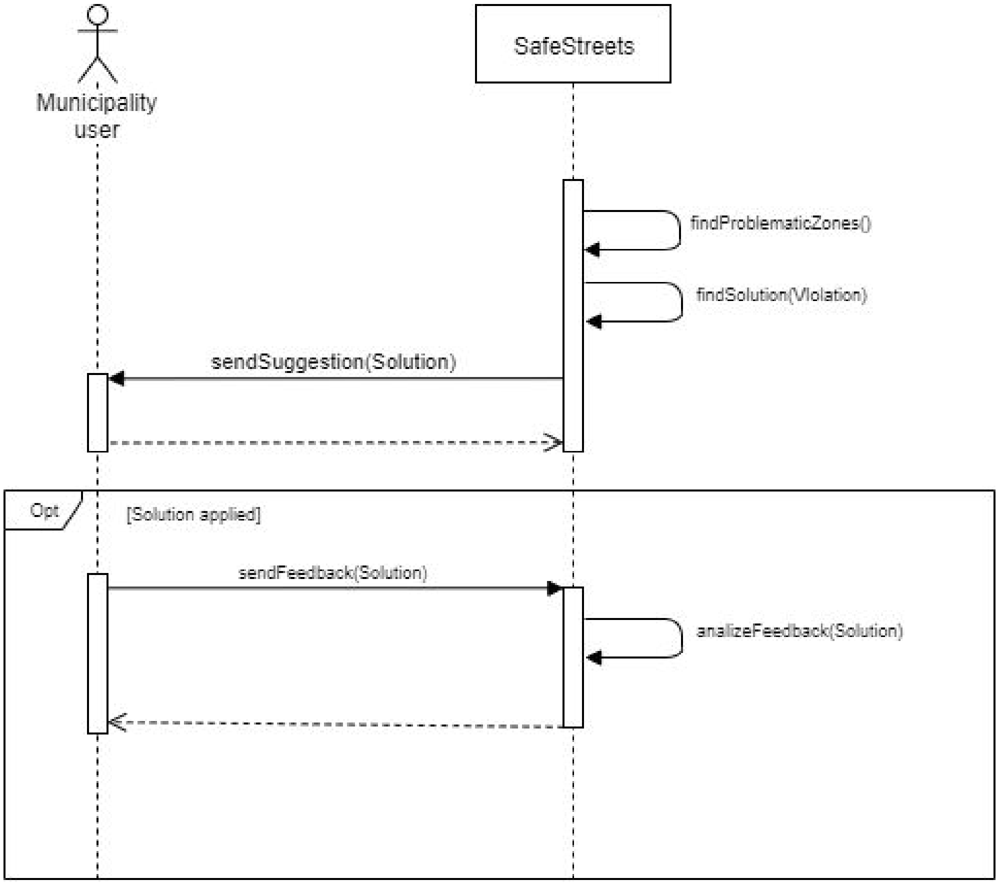


Figure 15 -Sequence diagram: send new segnalation

3.2.3 Requirements

G1: Users must be able to visualize on a map the zones where violations occur more frequently.

- **R1**: The system must allow the user to sign-up.
- **R2**: The system must allow the user to sign-in.
- **R3**: The system must provide the requested data if and only if the user has the permission to visualize them.
- **R4**: The system must store the segnalations sent by users in order to collect data of parking violations and where they occur.
- **R5**: The system must allow the user to choose which statistics visualize between those available.
- **D4**: The data provided by Municipality are correct.
- **D6**: Each user has a specific username that is unique.

G2: Users must be able to visualize statistics regarding the vehicles that commit the most violations.

- **R1**: The system must allow the user to sign-up.
- **R2**: The system must allow the user to sign-in.
- **R3**: The system must provide the requested data if and only if the user has the permission to visualize them.
- **R4**: The system must store the segnalations sent by users in order to collect data of parking violations and where they occur.
- **R5**: The system must allow the user to choose which statistics visualize between those available.
- **R11**: Municipality should be able to provide its own data to SafeStreets.
- **D2**: The picture must be taken with a camera that has at least 5 Mpx to allow the OCR algorithm to retrieve the licence plate.
- **D3**: The user's Internet Connection works properly during the upload of the segnalation.
- **D4**: The data provided by Municipality are correct.
- **D7**: SafeStreets has access to the GPS and camera of the device that is running the app.

G3: Users must be able to visualize statistics regarding the days and time at which certain zones are more exposed to violations.

- **R1**: The system must allow the user to sign-up.
- **R2**: The system must allow the user to sign-in.
- **R3**: The system must provide the requested data if and only if the user has the permission to visualize them.
- **R4**: The system must store the segnalations sent by users in order to collect data of parking violations and where they occur.
- **R5**: The system must allow the user to choose which statistics visualize between those available.
- **R8**: The system must complete the segnalation with the correct date, time, the licence plate and the name of the street.
- **R9**: Before accepting the user's report, the system must check that the data entered are correct and that the segnalation is complete in each field.
- **R11**: Municipality should be able to provide its own data to SafeStreets.
- **D3**: The user's Internet Connection works properly during the upload of the segnalation.
- **D4**: The data provided by Municipality are correct.
- **D7**: SafeStreets has access to the GPS and camera of the device that is running the app.

G5: Municipality should receive solutions that suggest how to solve the problems of violations in problematic zones.

- **R1**: The system must allow the user to sign-up.
- **R2**: The system must allow the user to sign-in.
  (Secondo te ci vanno anche r7, r8, r9, r10 qui?)

- **R4**: The system must store the segnalations sent by users in order to collect data of parking violations and where they occur.
- **R11**: Municipality should be able to provide its own data to SafeStreets.
- **R12**: The system should recognize zones where the occurrence of parking violations can be reduced and find a solution to propose.
- **R13**: The system must update its solution table, evaluating if the occurrence of violations has decreased, whenever the municipality reports that a solution has been adopted.
- **D4**: The data provided by Municipality are correct.
- **D5**: SafeStreets has a mapping that associates every possible type of violation with a possible solution to fight the specific problem.

G6: Users should be able to report violations anonymously.

- **R1**: The system must allow the user to sign-up.
- **R2**: The system must allow the user to sign-in.
- **R4**: The system must store the segnalations sent by users in order to collect data of parking violations and where they occur.
- **R6**: The system must allow the user to attach to the segnalation a photo just taken through the app.
- **R7**: The system must allow the user to specify the type of parking violation that just occurred.
- **R8**: The system must complete the segnalation with the correct date, time, the licence plate and the name of the street.
- **R9**: Before accepting the user's report, the system must check that the data entered are correct and that the segnalation is complete in each field.
- **R10**: The system, after having accepted the user's report, must take care of removing all sensitive user data from the segnalation before inserting it in the database.
- **D1**: The user’s smartphone GPS will provide a position with an error within 20 meters.
- **D2**: The picture must be taken with a camera that has at least 5 Mpx to allow the OCR algorithm to retrieve the licence plate.
- **D3**: The user’s Internet Connection works properly during the upload of the segnalation.
- **D6**: Each user has a specific username that is unique.
- **D7**: SafeStreets has access to the GPS and camera of the device that is running the app.

G7: Municipality should be able to have access to all the anonymous segnalations sent to SafeStreets.

- **R1**: The system must allow the user to sign-up.
- **R2**: The system must allow the user to sign-in.
- **R3**: The system must provide the requested data if and only if the user has the permission to visualize them.
- **R4**: The system must store the segnalations sent by users in order to collect data of parking violations and where they occur.

- **R5**: The system must allow the user to choose which statistics visualize between those available.
- **R6**: The system must allow the user to attach to the segnalation a photo just taken through the app.
- **R7**: The system must allow the user to specify the type of parking violation that just occurred.
- **R9**: Before accepting the user's report, the system must check that the data entered are correct and that the segnalation is complete in each field.
- **R10**: The system, after having accepted the user's report, must take care of removing all sensitive user data from the segnalation before inserting it in the database.
- **D1**: The user’s smartphone GPS will provide a position with an error within 20 meters.
- **D2**: The picture must be taken with a camera that has at least 5 Mpx to allow the OCR algorithm to retrieve the licence plate.
- **D3**: The user’s Internet Connection works properly during the upload of the segnalation.
- **D7**: SafeStreets has access to the GPS and camera of the device that is running the app.

G9: Each user should be able to visualize the list of segnalations sent by him.

- **R1**: The system must allow the user to sign-up.
- **R2**: The system must allow the user to sign-in.
- **R3**: The system must provide the requested data if and only if the user has the permission to visualize them.
- **R4**: The system must store the segnalations sent by users in order to collect data of parking violations and where they occur.
- **D3**: The user’s Internet Connection works properly during the upload of the segnalation.

Traceability matrix

The following table shows for each requirement the relations that it has with respect to use cases. Here are listed only the use cases that are strictly linked with the corresponding requirement in order to avoid a huge number of repetitions.

R1	● Sign In
R2	● visualize maps ● visualize statistics ● send segnalations ● visualize segnalations send by him ● send to SafeStreets data ● Visualize all the segnalations ● Receive suggested solutions

R3	● Visualize statistics ● visualize segnalations send by him ● Visualize all the segnalations
R4	● Visualize statistics ● visualize maps ● visualize segnalations send by him ● Visualize all the segnalations ● Receive suggested solutions
R5	● Visualize statistics
R6	● Send new segnalation
R7	● Send new segnalation
R8	● Send new segnalation
R9	● Send new segnalation
R10	● Send new segnalation ● Visualize all the segnalations
R11	● Visualize statistics ● visualize maps ● Receive suggested solutions
R12	● Receive suggested solutions
R13	● Receive suggested solutions

3.3 Performance requirements

SafeStreets will reasonably have a large user base and a large amount of information within its database, so it must be able to manage everything ensuring a pleasant user experience for anyone who interacts with the system without any particular slowdown.

3.4 Design constraints

3.4.1 Hardware limitations

SafeStreets doesn’t have hardware limitations difficult to satisfy, in fact the only ones requested exist only to guarantee the correct functioning of the system and the services offered by it. Below there is a brief outline that lists the minimum requests that the device used by the user must satisfy:

- Internet connection
- GPS
- Camera (if the user wants to send reports)

3.4.2 Any other constraint

As previously mentioned SafeStreets wants to protect the privacy of its users and for this reason, although it allows each of them to view the reports they have sent in the past, when it offers to the municipality the reports saved in its database does not allow to trace the author of the report .

3.5 Software system attributes

3.5.1 Reliability

In order to guarantee and improve reliability of the software some tools of Statistical Modeling and Estimation of Reliability Functions are used; in addition, downtimes, and how they recur at certain time intervals, are continuously monitored. In this way it is possible to forecast the reliability through testing phase; when the results obtained are not good, a special team is responsible for reinforcing the system in order to make it more robust and more fault tolerant.

3.5.2 Availability

SafeStreets wants to achieve the highest level of availability through the implementation of individual components characterized by reliability and a complex system capable of responding quickly to errors. In any case, the goal is for the system to be available at least 99.9% of the time

3.5.3 Security

Security is a fundamental aspect of every software and the same can be said for SafeStreets that wants its users' identity and privacy to be safe as well as the confidential data that the municipality sends to the system. This is why the information in the database are encrypted.

3.5.4 Maintainability

Nowadays maintainability is a prerequisite if you want to offer a software product that can always be modern and up to date. In fact, through this attribute it can evolve continuously limiting the efforts required and the associated expenses. All this is ensured by the use of design patterns and following the design principles.

3.5.5 Portability

SafeStreets can work in different environments even if not all the functionalities are available on all the platforms because of some hardware limitations. However, it is possible to access the services offered by SafeSteets using a special application or the website through devices that have installed the main OS of smartphones and computers.

HOURS OF WORK

De Vita Simone

15/10/2019: 3h
17/10/2019: 30min
18/10/2019: 45min
20/10/2019: 5h
23/10/2019: 1h
24/10/2019: 30 min
27/10/2019: 2h

Fabio Fontana

15/10/2019: 3h
17/10/2019: 45min
18/10/2019: 1h 30min
19/10/2019: 4h
21/10/2019: 45min
23/10/2019: 1h
26/10/2019: 4h
28/10/2019: 30min
29/10/2019: 2h
30/10/2019: 2h