



POLITECNICO
MILANO 1863

Software Engineering 2

A.Y. 2019/2020

SafeStreets

RASD – Requirement Analysis and Specification Document

Version 1.0

Authors:

Daniele Comi 10528029 - 944534

Anton Ghobryal 10501942 - 945577

Professor:

Elisabetta Di Nitto

Contents

1. Introduction	4
1.1 Purpose.....	4
1.1.1 General Purpose	4
1.1.2 Goals	5
1.2 Scope	6
1.2 Definitions, Acronyms, Abbreviations.....	8
1.2.1 Definitions.....	8
1.2.2 Acronyms	9
1.2.3 Abbreviations.....	9
1.4 Revision history	9
1.5 Reference Documents	9
1.6 Document Structure.....	10
2. Overall description	11
2.1 Product perspective.....	11
2.2 Product functions.....	18
2.3 User characteristics.....	19
2.4 Assumptions, dependencies and constraints.....	20
2.4.1 Assumptions.....	20
2.4.2 Dependencies and Constraints	20
3. Specific requirements.....	21
3.1 External Interface Requirements	21
3.1.1 User interfaces.....	21
3.1.2 Hardware Interfaces	25
3.1.3 Software Interfaces	25
3.1.4 Communication Interfaces.....	25
3.2 Functional Requirements	25
3.2.1 User.....	25
3.2.2 Authority.....	39
3.2.3 Requirements.....	51
3.3 Performance Requirements.....	54
3.4 Design Constraints.....	55
3.4.1 Standards compliance	55
3.4.2 Hardware limitations.....	55
3.5 Software System Attributes.....	56

3.5.1 Reliability	56
3.5.2 Availability.....	56
3.5.2 Security.....	56
3.5.3 Maintainability.....	56
3.5.4 Portability	56
4. Formal Analysis using Alloy	57
5. Effort spent.....	69
6. References	70

1. Introduction

1.1 Purpose

1.1.1 General Purpose

SafeStreets is a crowd-sourced application that intends to provide users with the possibility to notify authorities when traffic violations occur, specifically parking violations. The application allows users to send pictures of violations, including their date, time, and position, to authorities. The main purpose of SafeStreets is to reduce the number of accidents that may be caused by certain violations that can be avoided easily. The following list may illustrate and visualize the type of violations, according to the traffic regulation and laws¹⁰, that may be captured and notified to the authorities:

- Double line parking
- Expiry of the parking time limit
- No parking area
- Parking in places reserved to people with disabilities
- Parking in the middle of bike lanes
- Parking near bus stops
- Parking on crosswalk
- Parking on residents reserved spots
- Parking ticket missing
- Possible vehicles damage by third parties (e.g. broken glass)

SafeStreets stores the information provided by the users, completing it with suitable metadata. When it receives a picture, it runs an algorithm to read the licence plate and stores the retrieved information with the violation, including also the type of violation (input by the user) and the name of the street where the violation occurred (which can be retrieved from the geographical position of the violation). In addition, the application allows both end users and authorities to mine the information that has been received, for example by highlighting the streets (or the areas) with the highest frequency of violations, or the vehicles that commit the most violations. Of course, different levels of visibility are offered to different roles, for example the authorities can see the licence plate numbers of the vehicles that commit any violation while the end user cannot see that.

Moreover, there's another functionality that can be provided by SafeStreets. If the municipality offers a service that allows users to retrieve the information about accidents that occur on the territory of the municipality, SafeStreets can cross that information with its own data to identify potentially unsafe areas, hence suggest possible interventions depending of the type of the most committed violation in that area. The following examples show which intervention could be suggested depending on the preceding examples of violations presented earlier in this paragraph:

- Add a barrier between the bike lane and the part of the road for motorized vehicles
- Install a towaway zone sign
- Increase parking slots

- Increase local police controls

The main purpose of this functionality is that SafeStreets also identifies areas with critical number of accidents and reports suggestions as a possible solution as an automatized method to engage with the problem. Thus, it could help the authorities to highlight where the interventions should be provided, and this functionality should make it easier to point out the areas with critical statistics. So, if the municipality provide the needed information, it helps with the traceability of the main problem, therefore handling it providing also a higher measurement on local security.

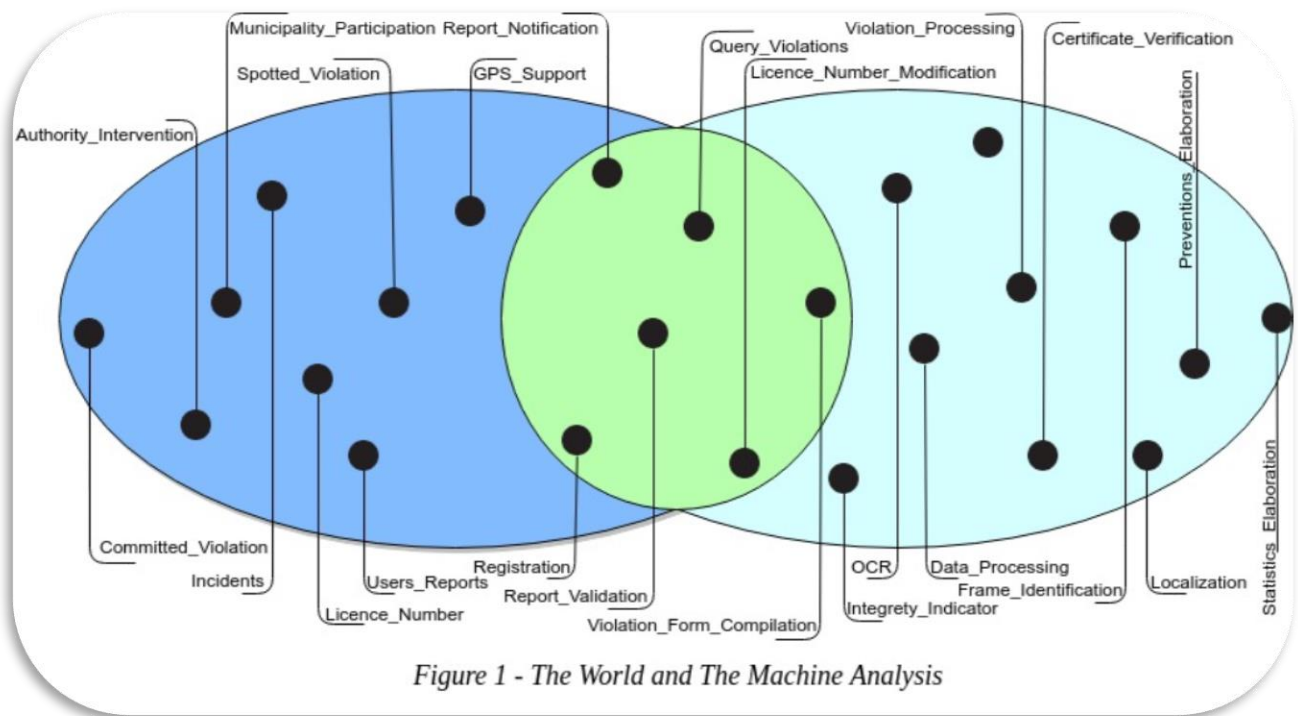
1.1.2 Goals

Taking the abstraction as an outcome of the “real-world” only, we should be able to define the goals as a part of the requirement engineering of an S2B to satisfy the stakeholders’ requests:

- [G1] Every registered user should be able to notify violations
- [G2] Every recognized authority should be able to access the application
- [G3] Every recognized authority should be able to receive any violation that has been pointed out by a registered user
- [G4] Every communication from the user must include a violation that has been committed by a recognizable vehicle
- [G5] Every registered end user should be able to mine general information about the violations committed in a certain area
- [G6] Every recognized authority must be able to verify the notified violations by the registered users
- [G7] Every recognized authority must be able to receive suggestions about improving the local security

Reading these goals, we should acknowledge the fact that the system considers two most end users: the normal user and the authorities. They’ll be defined later.

1.2 Scope



SafeStreets is meant to help authorities to identify some serious violations, traffic and parking violations, that may cause accidents in the future being. Thus, as it's been called, it's intended for making streets safer. Also, this application will increase the efficiency on reporting violations with the help of a common citizen. In order to report a violation, citizens won't have to go to a police station (that might be far from the current position of the violation), they won't even have to search where they are in order to report formally the committed violation. There are also some assumptions made in order to satisfy the goals of the S2B and the fundamental requirements that would help the lower level to easily realize the implementation part without considering the research on some tech already defined and available for use, also for higher level perspectives, for future improvements; thus it will be easier to integrate some new tech inherent to the domain of the application.

SafeStreets allow users to report a violation to the authorities when they spot one. In order to obtain the ability of using SafeStreets the user will have to register himself into the application system. Users have two different modes to register themselves into the system: the first one is the proprietary authentication and the second one consists of SPID⁹ authentication. Generally, they will have to subscribe with their full name and fiscal code since they're mandatory to be able to fill certain reports. Registered Users obtain points that indicate their integrity through their continuous voluntary participation in order to provide the possibility of achieving the goal of making the streets safer. These points are called integrity points. Initially, users, who have registered with SPID⁹, have more integrity points than the proprietary authentication (according to demonstrating more integrity into the society verifying his own identity through a public system of digital authentication). Moreover, when a report is verified by the authorities, integrity points of the notifier increase. Users can see also, through a map, the security level of a zone. Allowing users to mine general information about notified violations doesn't violate the privacy of the reporting user according to the Legislative Decree 196/03¹ and the regulation 2016/679³ given since they aren't authorized to access other users' private information such as fiscal code, name, surname etc.

Security level is calculated being based on the statistics of the types of violations committed in the interested area. Of course, any user will have the possibility to change the password in case it is forgotten through the normal process of password change link sent to their email address.

As it is in the specification of the S2B, Reports are composed of date, time, position, a note (with a maximum fixed number of characters) and a clear picture of the committed violation in which the licence plate should be included, but it isn't a restricted requirement because, in the worst case, there are two possible situations: in the first one the licence plate isn't clear (e.g. poor quality or blurry image) the user is allowed to do one out of two possible actions that consist of re-take the picture of the violation or modify the licence plate number, and if the user chooses to do the second action, the system shall recognize the report as one, instead, with a modified licence plate number and this induces minor level of credibility; instead, in the second situation, if the system doesn't recognize a vehicle in the taken picture it will take an immediate action to discard this picture and it will eventually ask the user to take a new clearer picture to be able to proceed, and that precludes the fact that user might send pictures that are not in accordance with the domain of the application (e.g. photos that don't contain a vehicle such as selfies).

Since the violation must be notified in real-time domain, the user is not allowed to upload a picture at all. So that, situations as creating a false violation or manipulating data of a certain violation. For the same reason the user is not allowed to modify a photo. If the user notices something that should be mentioned, there's a note that he can fill in briefly with possible observations. Also, the user must have a stable active connection to be able to submit the violation.

A report should satisfy the application domain before it becomes in hands of authorities and in order to realize this fact a report should include the preconditions described earlier. When a report is filled in completely the authorities must be able to receive it through the application. Within this context, the authorities are defined as Italy's law enforcement agencies. The authorities, interested in the application willing to use it for increasing local security, must have a valid digital certificate provided by the police forces through the Ministry of the Interior⁵ and the Ministry of the Defense⁶. An authority must register making a formal and direct request to the Certified Email^{7,8} address of SafeStreets through his Certified Email^{7,8} which will give him in a secure way the credentials generated to be able to use the application or by just using SPID with his Certified Email. The login process with the authority credentials requires also a valid digital certificate. Once an authority is registered and SafeStreets has added his credentials in the system, he will be able to receive notifications about the committed violations. Registered authorities have the maximum authorization to access all the data notified by users. They also have access to all normal user functionalities, thus the capability of reporting violations. The authorities can also verify and validate the visualized reports depending on the veracity of the notified violations.

Either the registration process or the reports made and of the user who carried it out are respects the terms established by the Legislative Decree 196/03¹, Legislative Decree 82/05² and the regulation 2016/679³.

SafeStreets offers also the possibility to be an important participant as an independent entity which can provide suggestions to the improvement of a certain area. In order to realize such a functionality, SafeStreets should have access to accident records of the applied areas. Interested municipalities, in order to apply this functionality, must guarantee access to those data records because it helps the application to cross the provided data about accidents with its own data to provide suitable suggestions depending on the identified situation.

1.2 Definitions, Acronyms, Abbreviations

1.2.1 Definitions

- **Violation:** a subset of anything that is classified as a traffic violation by the Traffic regulation and laws document. This subset is composed of:
 - Double line parking
 - Expiry of the parking time limit
 - No parking area
 - Parking in places reserved to people with disabilities
 - Parking in the middle of bike lanes
 - Parking near bus stops
 - Parking on crosswalk
 - Parking on residents reserved spots
 - Parking ticket missing
 - Possible vehicles damage by third parties (e.g. broken glass)
- **Vehicle:** any terrestrial identifiable vehicle subject to Traffic regulation and laws document, like cars, motorbikes, trucks, etc...
- **User:** any citizen registered in the system who is using any of SafeStreets functionalities.
- **Violation report, notification:** acknowledgment in SafeStreets system of a new violation occurred.
- **Authority:** any registered law enforcement using SafeStreets application alongside its authority-restricted functionalities
- **Municipality:** any central administration of a city or a town which may or may not give open access to its incidents data.
- **Reliability score:** score assigned to any user account which gives a sense of how much a user is reliable in giving information regarding violations.
- **Safe area:** a low radius geographical area where violations are lower than a certain threshold or lower than other areas.
- **Suggestion:** an automatically inferred hint given to the authorities by SafeStreets regarding how they could improve, with the help and permission of their municipality, area marked as high-risk area due to a high correlation of violations and incidents reported from the same municipality. Possible suggestions are:
 - Add a barrier between the bike lane and the part of the road for motorized vehicles
 - Install a towaway zone sign
 - Increase parking slots
 - Increase local police controls
- **Galileo:** Global localization system based on a network of 24 satellites commissioned by European Union and ESA (European Space Agency)
- **SPID:** is the unique system of access with digital identity to the online services of the Italian public administration and of private members: citizens and companies can access services with a unique digital identity in a secured way
- **Certified Email:** A certified email is an email that can only be sent using a special Certified Email Account provided by a registered provider. When a certified email is sent, the sender's provider will release a receipt of the successful (or failed) transaction.

This receipt has legal value and it includes precise information about the time the certified email was sent. A certified email account can only handle certified email and can't be used to send regular email.

1.2.2 Acronyms

- **API:** Application Programming Interface
- **D.L.:** Legislative Decree
- **DCPM:** Decree of the President of the Council of Ministers of the Italian Republic
- **DD:** Design Document
- **EEA:** European Economic Area
- **EU:** European Union
- **GDPR:** General Data Protection Regulation
- **GPS:** Global Positioning System
- **IEEE:** Institute of Electrical and Electronics Engineers
- **S2B:** Software to Be
- **SPID:** Public Digital Identity System
- **UI:** User Interface

1.2.3 Abbreviations

- G_n = n^{th} goal
- D_n = n^{th} domain assumption
- R_n = n^{th} requirement

1.4 Revision history

1.5 Reference Documents

- D.L. 196 of 2003 (196/03) <https://www.camera.it/parlam/leggi/deleghe/Testi/03196dl.htm>
- D.L. 82 of 2005 (82/05) <https://docs.italia.it/italia/piano-triennale-ict/codice-amministrazione-digitale-docs/it/v2017-12-13/index.html>
- General Data Protection Regulation (EU) 2016/679 <https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32016R0679>
- IEEE 830-1998 - IEEE Recommended Practice for Software Requirements Specifications <https://standards.ieee.org/standard/830-1998.html>
- IEEE 29148-2018 - ISO/IEC/IEEE International Standard - Systems and software engineering -- Life cycle processes -- Requirements engineering <https://standards.ieee.org/standard/29148-2018.html>
- Specification document "Mandatory Project Assignment AY 2018-2019" https://polimi365-my.sharepoint.com/:b/g/personal/10528029_polimi_it/EXR1gN6gBoxJgMC86Ow45gMBFwZzkRSWuoaf5K7t1wZutA?e=SPnVkl
- Ministry of the Interior and digital certificates released <http://politichepersonale.interno.it/ita/index.php?IdMat=1&IdSot=35&IdNot=386>
- Ministry of the Defence and digital certificates released <http://www.pkiff.difesa.it/#secEN>
- Certified Email <https://www.agid.gov.it/it/piattaforme/posta-elettronica-certificata>

- Certified Email RFC <https://tools.ietf.org/html/rfc6109>
- SPID <https://www.agid.gov.it/it/piattaforme/spid>
- Traffic regulation and laws <http://www.aci.it/i-servizi/normative/codice-della-strada.html>
- Italian license plate verifier <http://www.targa.co.it/data/doc.aspx>
- Police State license plate verifier <https://www.crimnet.dcpc.interno.gov.it/crimnet/ricerca-targhe-telai-rubati-smarriti/FAQ>
- Alloy documentation <http://alloytools.org/documentation.html>

1.6 Document Structure

- Chapter 1 is an introduction: it describes the purpose of the system informally and by making use of the list of goals which the application must reach. Moreover, it defines the scope, where the aim of the project is defined in greater detail and the application domain and the most important shared phenomena are shown.
- Chapter 2 offers an overall description of the project. Here the actors involved in the application's usage lifecycle are identified and the boundaries of the project are defined, listing all the necessary assumptions. Furthermore, a class diagram is provided, aid to better understanding the general structure of the project, with all the related entities. Then some state diagrams are listed to make the evolution of the crucial objects clear. Finally, the functions offered by the system are here more clearly specified, with respect to the previously listed goals.
- Chapter 3 represents the body of the document. It contains the interface requirements, which are: user interfaces, hardware interfaces and software interfaces. It then lists some scenarios to show how the system acts in real world situations, followed by the description of the functional requirements, using use cases and sequence diagrams. All the requirements necessary in order to reach the goals are given, linked with the related domain assumptions. Lastly, the non-functional requirements are defined through performance requirements, design constraints and software system attributes.
- Chapter 4 contains the Alloy model of some critical aspects with all the related comments and documentation in order to show how the project has been modelled and represented through the language.
- Chapter 5 shows the effort which each member of the group spent working on the project.

2. Overall description

2.1 Product perspective

In the below figure 2 is shown the main structure of the SafeStreets application, only the main parts without going too deep in the actual SafeStreets structure which is not a topic for this document. One of the main objectives is security. To offer two type of different authentications, an authentication manager will have as its duty to offer these two authentications as transparent as possible, of course SPID⁹ authentication will have to communicate with different server than the ones of SafeStreets. SPID⁹ servers and its authentication process will follow its documentations. User data has to be secured in the best way possible following directives imposed by D.L. 196/03 and the General Data Protection Regulation 2016/679, so even if not shown, for clarity reasons, in the class diagram data will indeed have to be decrypted to be seen from the application and has to be crypted when memorized in the database through SafeStreets servers to guarantee. A user will not ever be able to access other user restricted information, SafeStreets will grant him only the permission to explore violations signalled. Contraposed to the authority which has the legal rights to see every user data, expect obviously the authentication related ones. An authority to access this data will have to let its previous asked digital certificate be verified in every application session to use even just one of the main three restricted functions.

SafeStreets needs to be as fast as possible, considering the requirement to have always an available Internet connection, SafeStreets computing servers will be used for peculiar sub functionalities like live statistics computation on violations data or like the violation image validity recognizer which will be featured with the use for example of a Convolutional Neural Network helping authorities to have as many as possible valid violations.

Concerning valid violations to also ensure no misuse of any localization spoofing service, to upload to SafeStreets servers a new violation it will be needed a granted, precise and secured positioning system.

SafeStreets needs to be as autonomous as possible to prohibit any misuse of its violations reporting system. Indeed, even when a user needs to modify the license plate autonomously read in case of a wrong reading, the application through Violation Control will have to lower the quality attribute of this violation due to this needed modification made by the user, even if it will result right.

Expect some note that a user can write about some violation and obviously the type of violations, editing the license, at the cost of a notified lower quality, is the only allowed edit for a violation that can be made by the user.

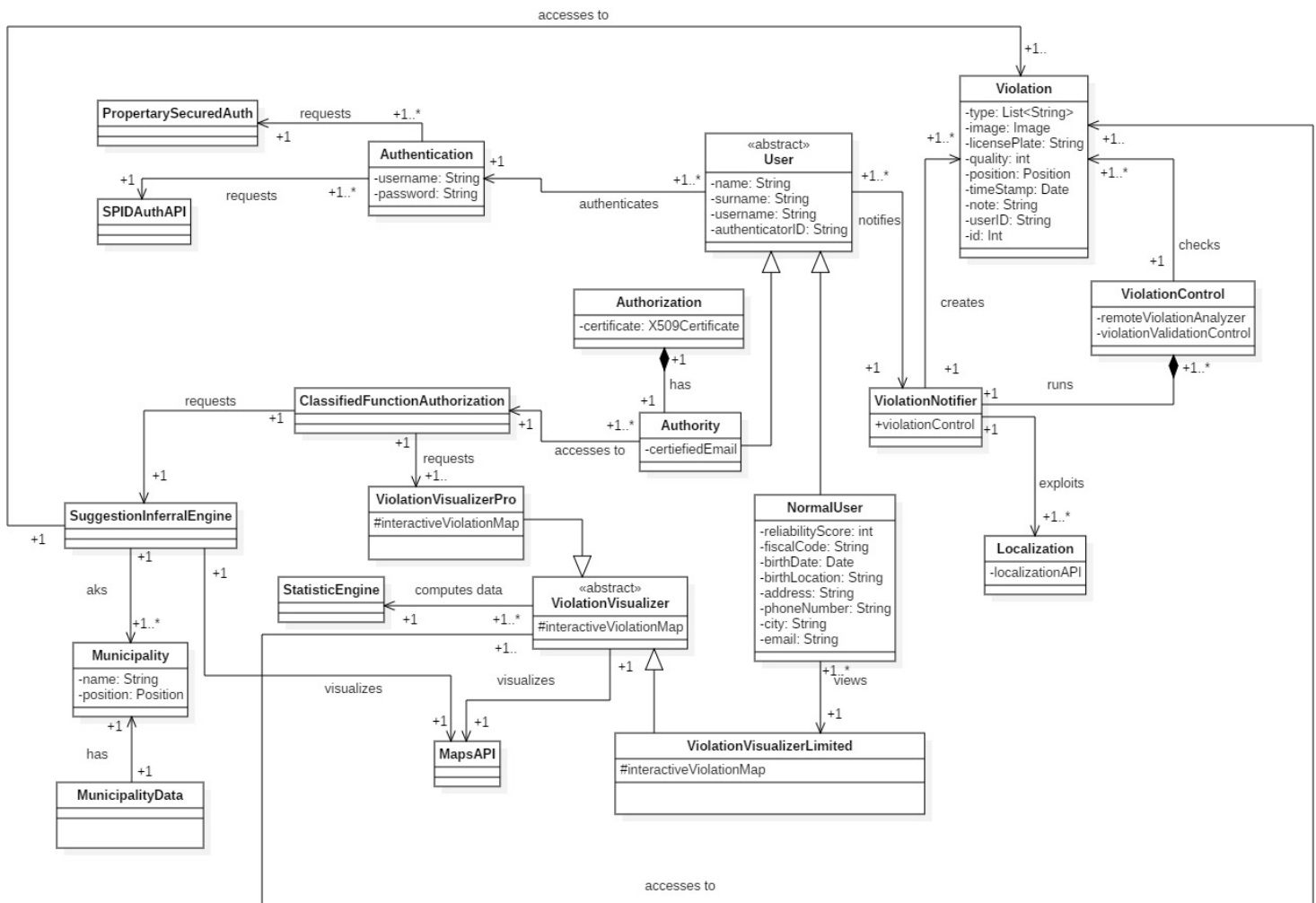


Figure 2 – SafeStreets Descriptive Class Diagram

In the following images we are going to analyse some state diagrams displaying some major functionalities.

Here is shown in Figure 3 the State Diagram regarding the notification of a violation. The starting point as seen is the user starting to create a notification of a violation. At first the user is requested to take a picture which has to be indeed valid, to be valid it need to include a vehicle which is also identifiable through its license plate and will continue if and only if the picture is valid. Data regarding the application which is taken automatically will be consider valid if and only if an internet access is available and a satellite link is present. After inserting furthermore metadata in the notification the notification will be sent if and only if there is an internet access available.

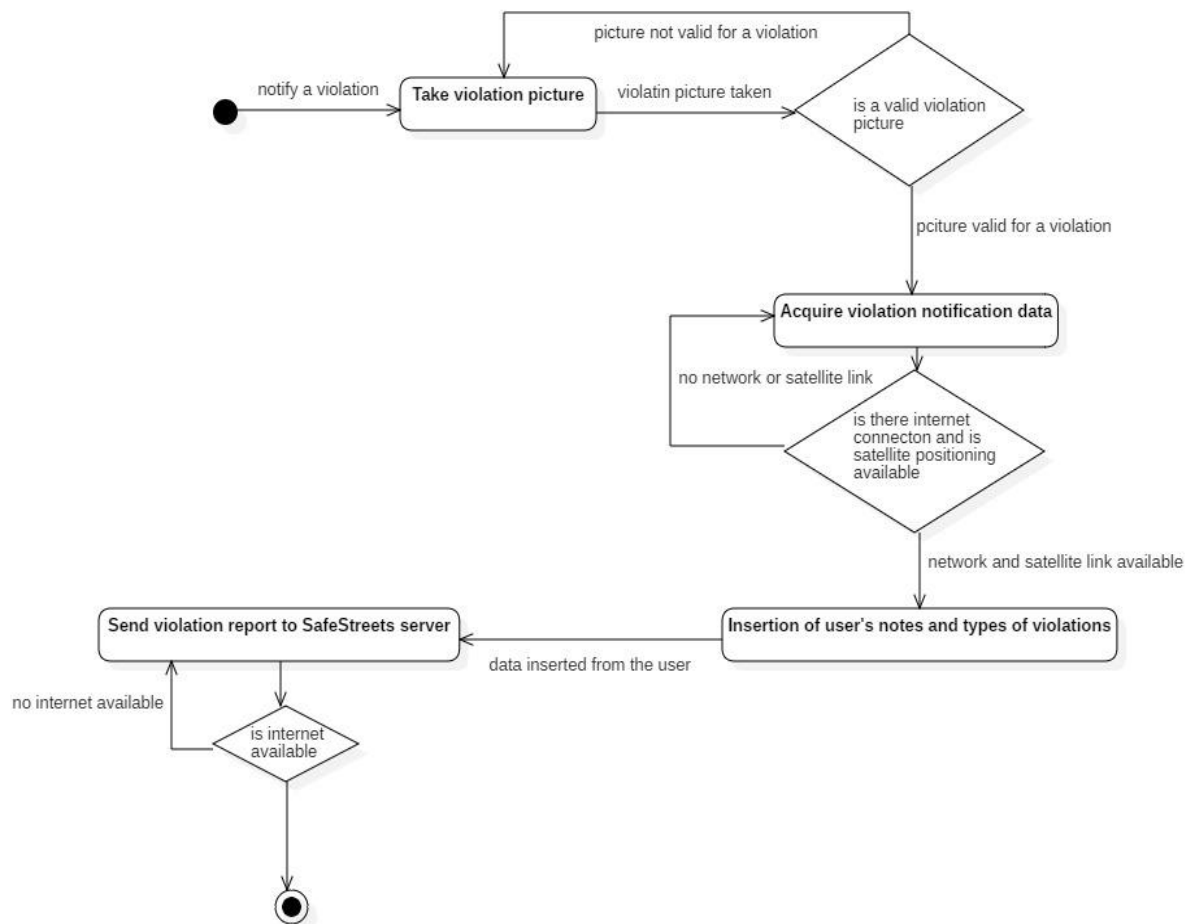


Figure 3 – Notification of a violation State Diagram

Then next image in Figure 4 is instead regarding the process of validation of a violation report by an authority. As can be seen below after receiving a notification from SafeStreets the authority will first check the details of the violations, and after being sure of its validity on his will the report will be set as verified, on the condition that there is an internet connection available. SafeStreets will then automatically increment the user reliability score.

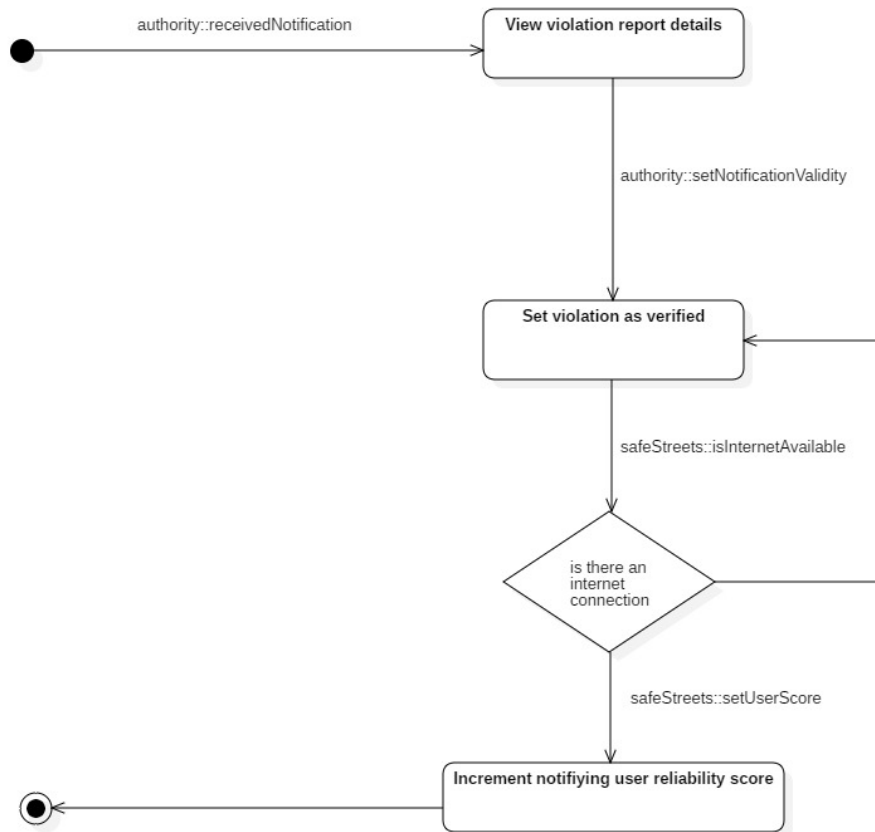


Figure 4 – Authority checking a violation report notified State Diagram

In the State Diagram shown in Figure 5, it is represented how a new suggestion is inferred and then notified to authorities. Periodically SafeStreets will check for new municipality data and will infer new suggestions using SafeStreets data. If a new suggestion is available, it will be notified to the authorities.

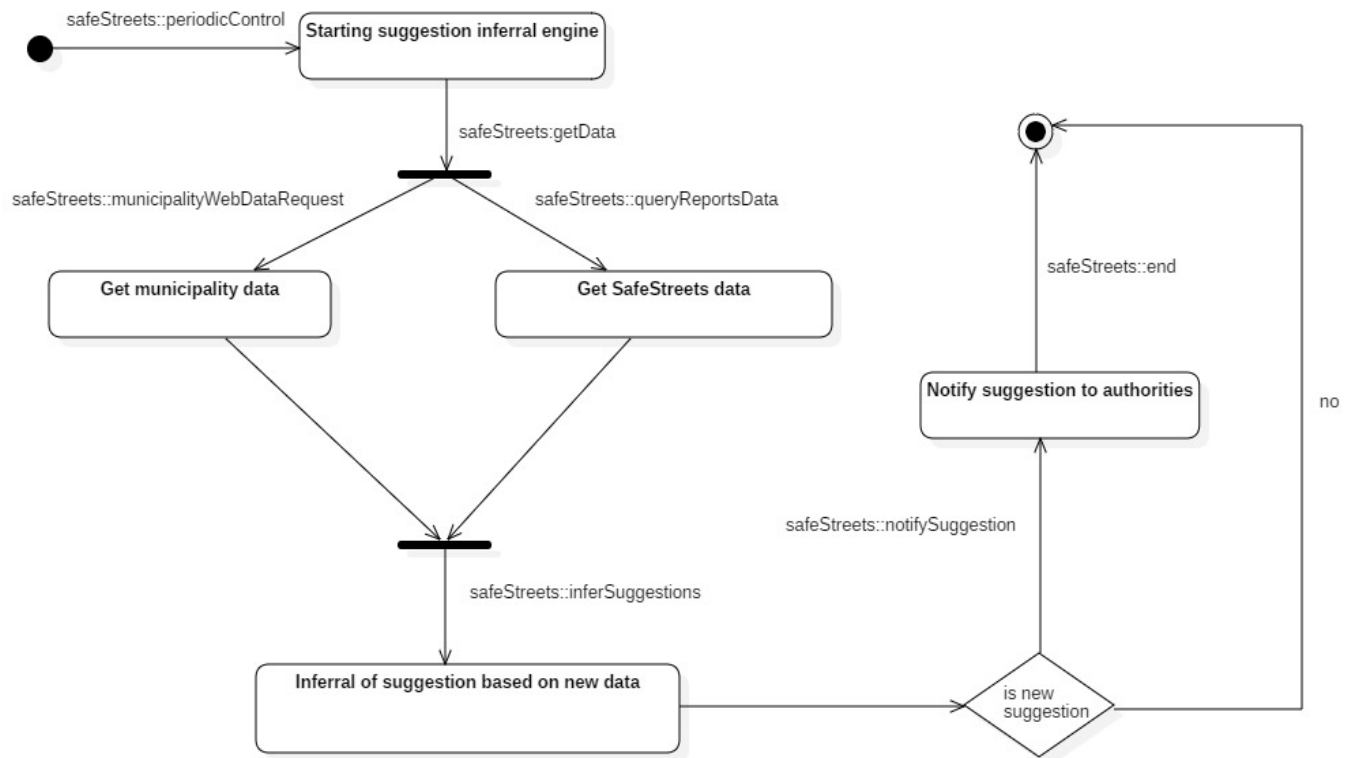


Figure 5 – SafeStreets inferring new suggestion for a municipality to authorities State Diagram

In the last State Diagram shown in Figure 6, it is represented how statistics, upon user request to display them, are computed and then efficiently sent.

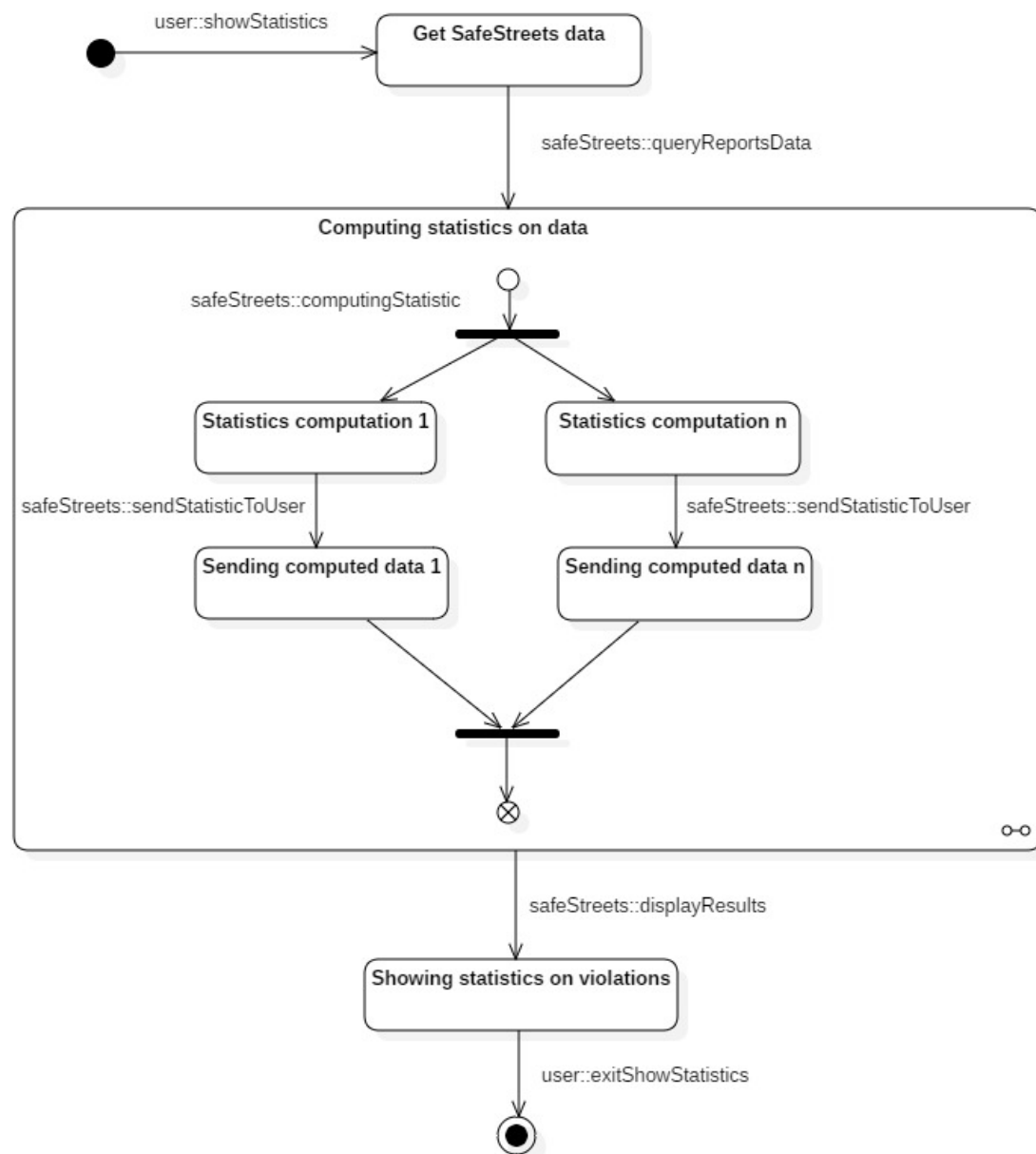


Figure 6 – SafeStreets computing and sending statistics to be shown to the user State Diagram

The following table shows instead the various phenomena and who control them in a shared or not shared way.

Phenomenon	Shared	Who controls it?
User registers to the application	Y	W
User (or authority) accesses the application	N	W
User spots a violation	N	W
User takes a picture of the spotted violation	Y	W
User fills the report and notifies the committed violation	Y	W
The system processes the reported violation	N	M
User modifies the licence number	Y	W
The system refuses the same violation reported by the same user	Y	M
Authority registers through a registration API	Y	W
The system recognizes an authority through a proper digital certificate	Y	M
Authority receives a notification about some violation	N	M
Authority intervenes based on the notified violation	Y	W
The system accepts (or refuses) a photo of a violation taken by a user	Y	M
Communication between user and authority includes only a proper filled report of some violation	N	M
User (or authority) searches general information and statistics about committed violations of some area	Y	W
The system calculates the statistics based on the observed data in its own database	N	M
Authority verifies some user's report	Y	W
The system calculates the user's integrity points	N	M
Authority evaluates user's credibility through his profile integrity points	N	W
Municipality participates in the program	N	W
System crosses its data with the municipality's data to provide suggestions	N	M
Authority receives suggestions about possible improvements	Y	M
Authority intervenes based on the provided suggestions	Y	W
Authority receives a notification	Y	M

2.2 Product functions

In this section the most important functions and requirements are listed and explained with few more details

- **Violations signaller**

This functionality allows the user to create a new violation record to be sent through Internet to SafeStreets server which will safely and securely memorize it in the databases. The violations signaller asks to the user just to take a picture of the identifiable vehicle committing, at most of his knowledge, a violation or more precisely from one to up to three of the violations which are listed and allowed by SafeStreets during the violation notification.

The user will also need to compose a small note containing a text which length is up to 140 characters, its function is to clarify at user's best capabilities the current situation which will eventually be controlled by authorities, just to give them much details and context as possible. Any other data will be automatically filled by SafeStreets application. Like the violation position, its timestamp, vehicle license plate and the violation quality which actually depends on the latter: if a user thinks the automated license plate reading system did a wrong read due to bad angled images or other issues which could compromise OCR accuracy, he will be able to edit the license plate field at the cost of reducing violation quality for security reason.

In order to prevent the same violation to be reported twice by the same user, it will be prevented to any user to send any kind of violation reports involving the same vehicle within at least 20 minutes. Also, it will be in general prohibited to report twice a violation of the same vehicle from the same precise location.

- **Safeness areas map**

This functionality is one of the main reasons SafeStreets idea was born. Allowing users to know how safe a certain geographically area by using every data collected about violations with the help of authority verification. Any user will be able just to open inside SafeStreets this map to have him pinpointed every area in his surroundings with its safety level regarding street traffic. This will allow to make eventually, in a certain time period, streets safer and safer with a decreasing number of incidents caused by traffic violations and a constant increase of citizens happiness regarding how traffic and traffic services are managed by their municipality. Of course, all this can be possible with the intervention of the authorities which will also be discussed in the following functionality how it will be done.

- **Suggestion inferring system**

This functionality is another very important one for SafeStreets: it allows authority to know which solutions they should apply, and propose to municipality council, to reduce the amount of violations committed which led to incidents reported by municipality data. There will be a list of possible suggestions and they will be inferred completely

autonomously by taking in account violations data notified by the users of SafeStreets and, as said before, municipality data regarding incidents. Authority will be able to both check suggestions whenever they want but they will also be notified as soon as a new suggestion has been inferred by SafeStreets.

- **Automated violation validity control**

To help authorities doing a better job and to give them the least work possible regarding violation notified through SafeStreets, an autonomously violation validation system will be present to check if a currently composing violation notification actually a traffic violation and not something else like any other type of uncorrelated violations or anything which does not involve any violations at all, either an error or not done by the user. It will be able to recognize if the image is containing an actual identifiable vehicle and it will be able to read its license plate.

- **User reliability score improvement**

An important functionality to improve SafeStreets efficiency in doing what's was designed for, a reliability score is assigned to each user indicating how much a user is reliable in signalling violations. Authority can increase or decrease this score by verifying it, but it will be also done autonomously in case of multiple consecutive bad composed violation notifications from a user. A better imposed user base will improve SafeStreets purpose fulfilment.

- **Authority violations map**

An important functionality thought for Authorities indeed is the map and list displaying violations notified in a certain area by SafeStreets users. Authority will be able to access every data related to a specific violation: personal data of the user who made it, violations type, license plate, data inferred from the license plate using Police State and Italian license plate verifier API which will even check for stolen vehicles, target location and so on. Violations reports will also be notified to authority as soon as they are reported, each authority will receive notifications regarding only his municipality.

2.3 User characteristics

SafeStreets requires some fundamental actors who could help in order to exploit the possibility to use all the possible functionalities to accomplish what is built for as described earlier. Those actors belong to three different categories: a normal user or a citizen, the authorities (mainly, local police) and municipality. Their characteristics will be described as follows:

- **User:** a person who registers to SafeStreets, a citizen of the community, a passer-by who may spot a violation while he's walking around the streets, having the possibility to report it via the application, simply taking a photo of the violation, and filling in the required data. He may register through the proprietary authentication or SPID⁹. Data as his full name and his fiscal code are restrictedly required. He has access to his own profile on which, in addition to his own data, there will be all the violation notified followed by an integrity score which describes the credibility of the reports provided by him. He's also allowed to make queries about other violations (obviously, without violating other users'

privacy accordingly with the terms established by the Legislative Decree 196/03¹ and the regulation 2016/679³). He's also allowed to visualize, through a map, the security level of a certain area and the statistics on the type committed violations.

- **Authority:** mainly, the authority is represented by the local police (in the domain of the application). Authorities are recognizable through a valid digital certificate which is provided by the police forces through the Ministry of the Interior⁵ and the Ministry of the Defence⁶, usable according to D.P.C.M. of 30.10.2014 N.193 art.21 and art.22⁴. Once they access, they have the maximum authority to access data of the violations made respecting the terms established by the Legislative Decree 196/03¹ and the regulation 2016/679³. They should act based on the information provided by a violation notification. They also receive suggestions on how to decrease the number of violations and, consequently, increase the local security. They also have access to the same functionalities as the normal users.
- **Municipality:** municipalities, in the domain of the application, has a main role as a provider of accidents' records, if those are available. If a municipality is interested in applying the application in order to obtain suggestions about the improvement of local security as mentioned earlier, they must allow SafeStreets to access municipality database on accidents and this is mandatory in order to cross those data with SafeStreets data to provide accurate suggestions about possible improvements to avoid more violations.

2.4 Assumptions, dependencies and constraints

2.4.1 Assumptions

- **D1:** Users can identify a violation.
- **D2:** Authorities possess valid digital certificates provided by the police forces through the Ministry of the Interior⁵ and the Ministry of the Defence⁶.
- **D3:** Users decide to modify the licence number providing the right one if and only if the recognized licence number, through the application, isn't the correct one.
- **D4:** Users have an available internet connection for the whole process of reporting a violation.
- **D5:** Every considered geographical area is covered by a satellite location system (e.g. GPS and Galileo).
- **D6:** Authority acts based on notifications about violations or suggestions about security improvement.
- **D7:** The quantity of both SafeStreets data and those provided by the municipality, if available, is enough to infer suggestions to authorities.

2.4.2 Dependencies and Constraints

There are three main issues that cover different areas of the S2B:

- **Internet connection:** SafeStreets requires a stable internet connection in order to function as it should be. However, areas internet coverage depends on the operator that provides internet services.

- Location System: Geographically, there are areas where it's very hard to obtain an accurate position of the user's device. Thus, SafeStreets will be able to provide accurate position depending on where the committed violation is.
- Data crossing: SafeStreets will be able to provide accurate suggestions depending on the quantity of its own data and the quantity of data provided by the municipality.

Digital certificates, provided by authorities, must be compliant with the D.P.C.M. of 30.10.2014 N.193 art.21 and art.22⁴. Finally, the S2B must be compliant with the GDPR³ normative by the privacy.

3. Specific requirements

3.1 External Interface Requirements

3.1.1 User interfaces

The following design of the smartphone app of SafeStreets for both users and authorities and the design for the desktop application exclusively available for authority are here shown. They are just an intuitive way to lead the design towards this direction but any necessary changes or adjustments or new components can be added to ensure a better user experience and improve both usability and simplicity.

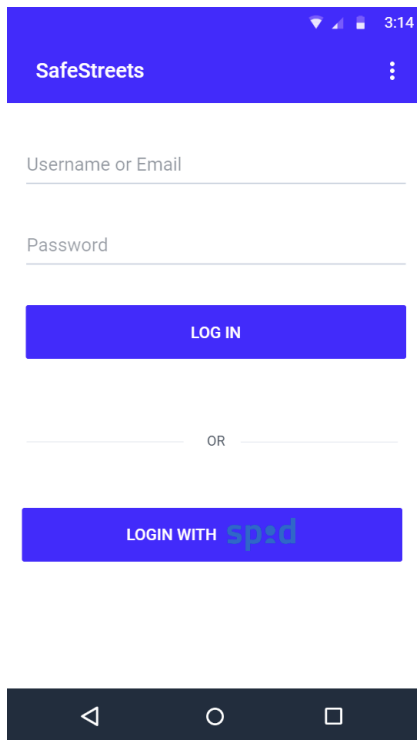


Figure 7 – Login

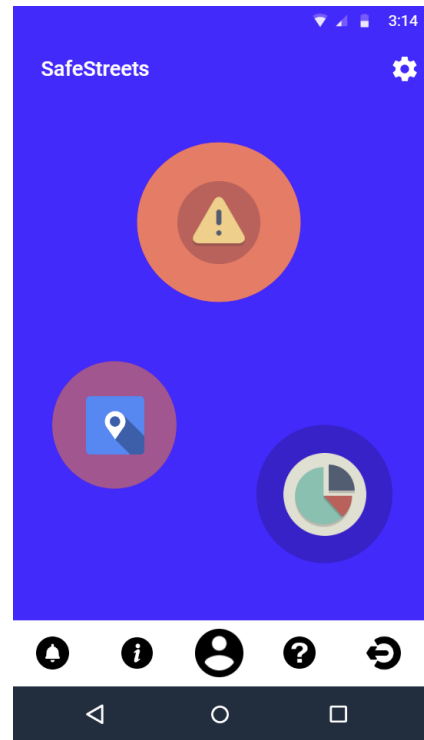


Figure 8 – Home screen

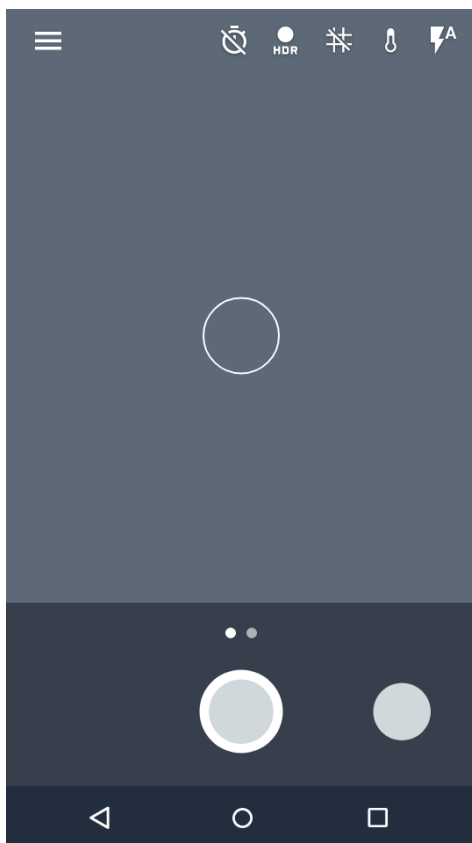


Figure 9 – Camera for violation picture

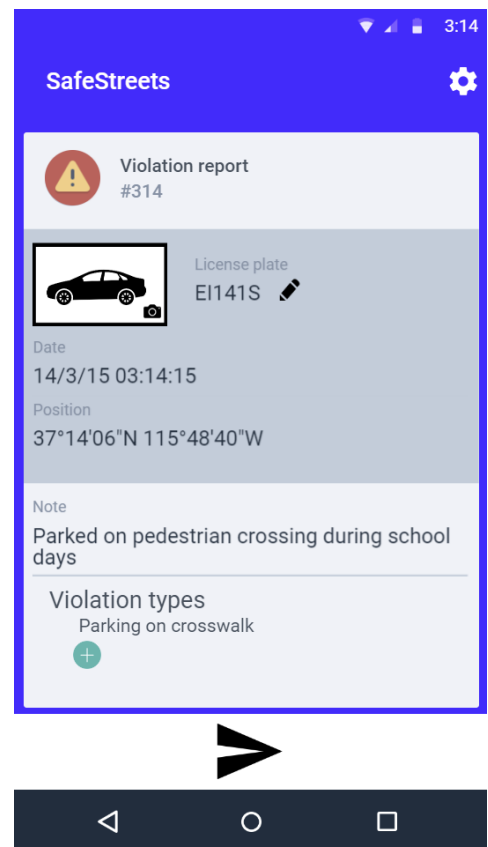


Figure 10 – Violation report UI

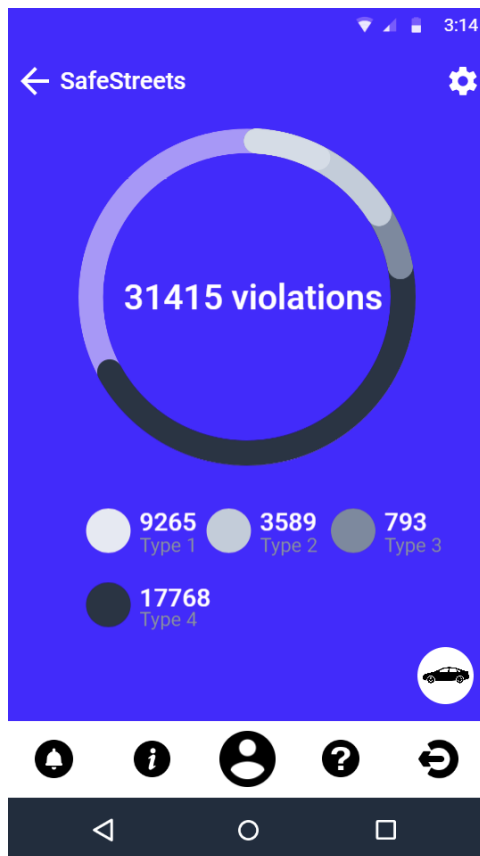


Figure 11 – Statistics



Figure 12 – Safeness Map

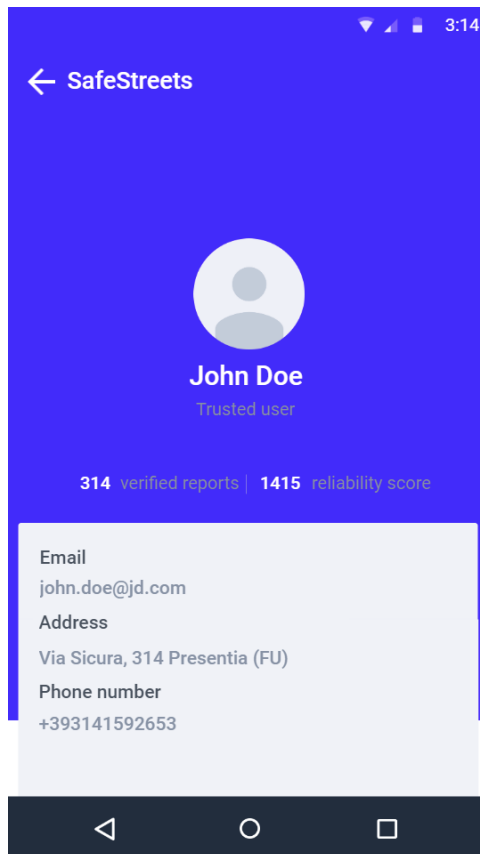


Figure 13 – User account details

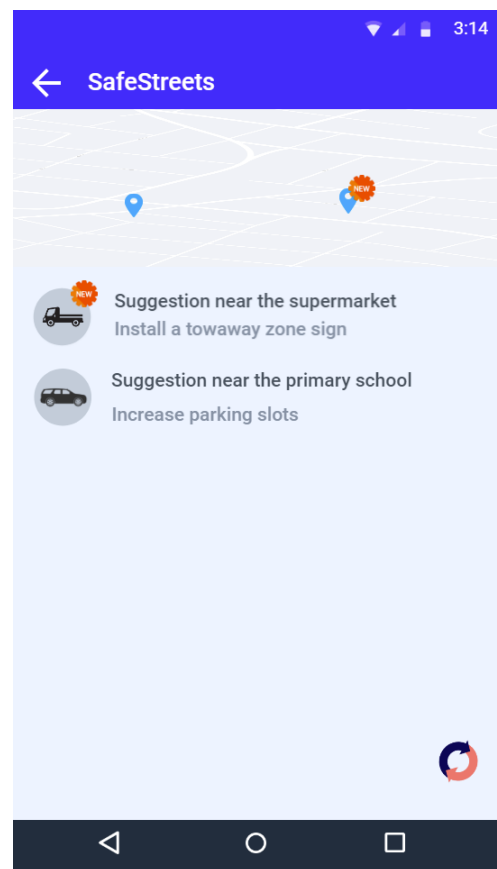


Figure 14 – Inferred suggestions to authorities

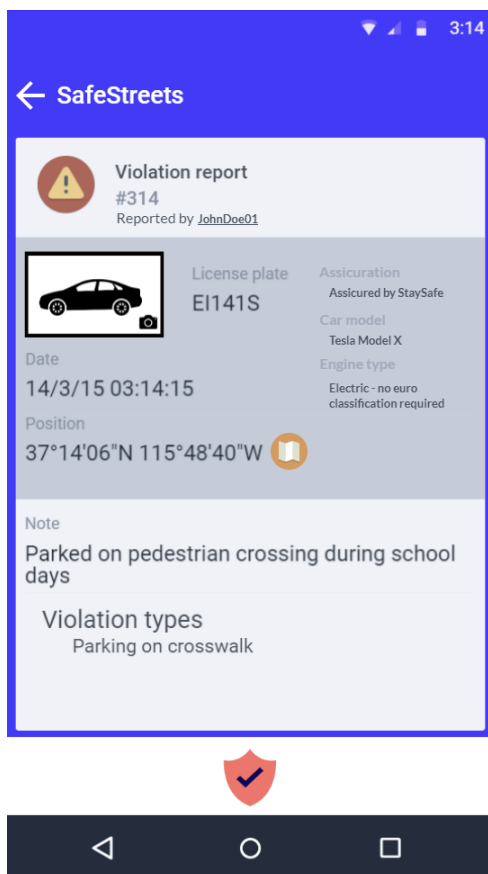


Figure 15 – Violations reported to authorities

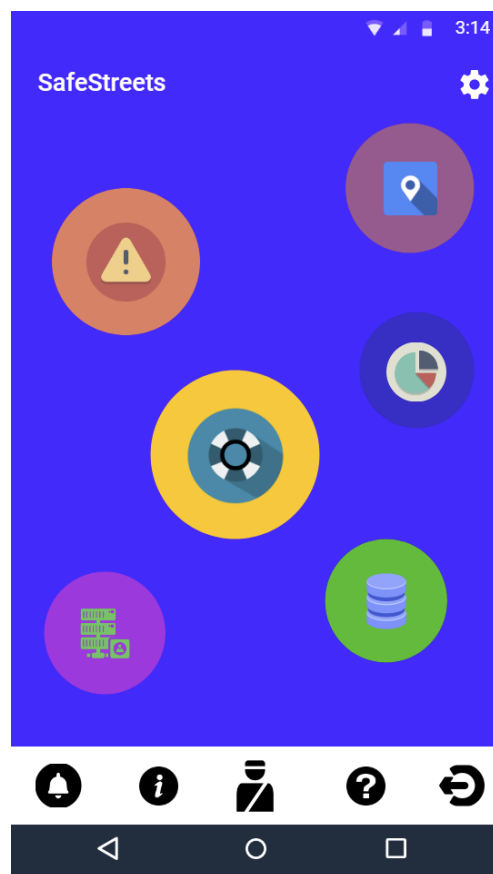


Figure 16 – Authorities home screen

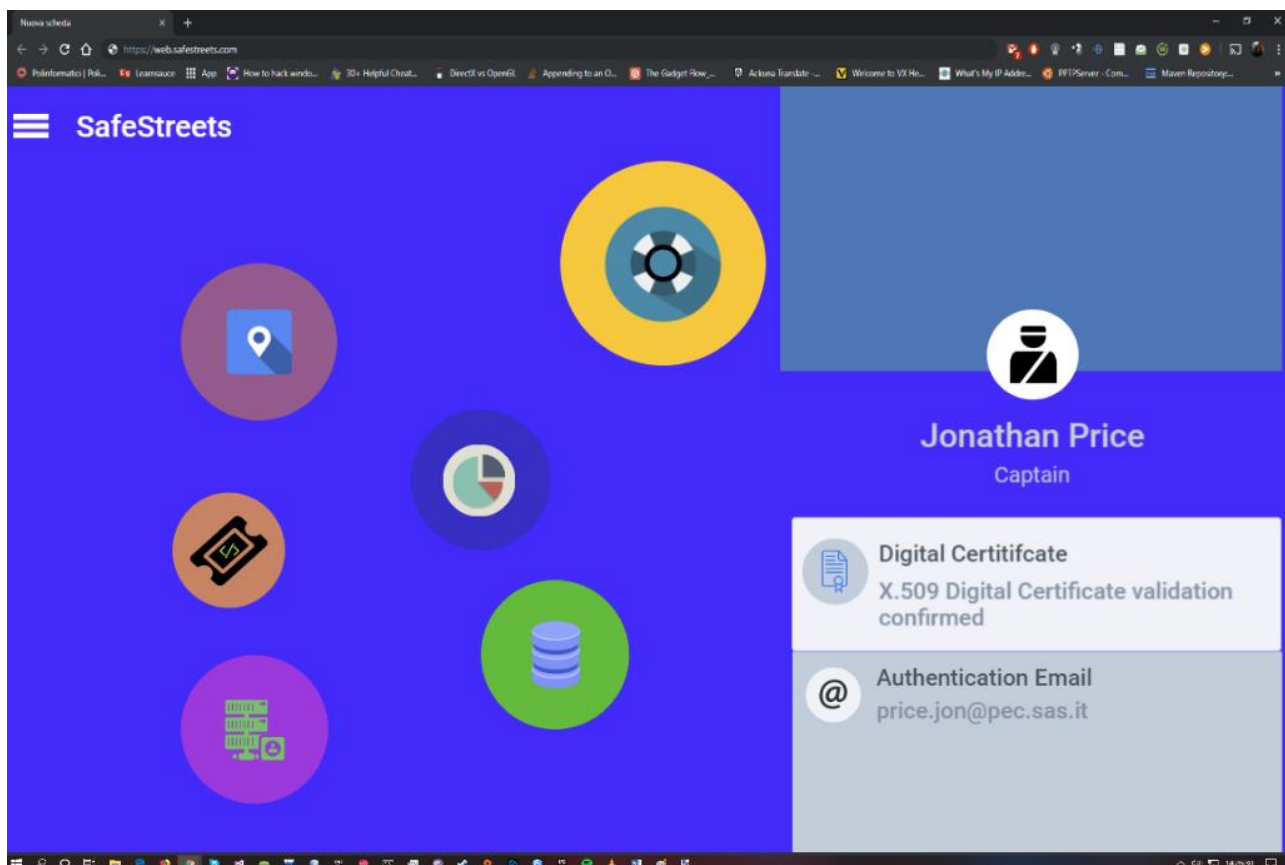


Figure 17 – exclusive authority SafeStreets desktop access through secure SaaS web-app

3.1.2 Hardware Interfaces

The S2B has no hardware interface since it's an application which can be normally installed on user's and authority's smartphones or configured on the authorities' computers (For further information, see the DD).

3.1.3 Software Interfaces

There are two main interfaces to-be implemented in order to satisfy the main functionalities of the application. Here will be introduced only an overview, for further information, see DD.

The first interface is an interface on the authorities' devices in order to interact with the notified violations or suggestions virtually. The second one has to interact with the municipality database in order to allow crossing SafeStreets' data types with municipality's data. The third interface is to interact with the Italian license plate verifier and with State Police license plate verifier.

3.1.4 Communication Interfaces

There's an asynchronous communication interface between user interface and authority interface. For further information see DD.

3.2 Functional Requirements

3.2.1 User

Scenarios:

- Antonio is an employee of a care taking company for disabled people. Every day of the week he helps disabled people by driving them across town to do some shopping, go to the pharmacy, go to the park to take a walk under the sun, etc... Unfortunately, he often finds parking places reserved for disabled people already taken by someone who has no valid permit to park in those reserved areas. After lots of complaints, his company decided to make all its employees use SafeStreets. Antonio can now know every area safeness and their frequencies of violations regarding parking in places reserved for disabled people. Thanks to SafeStreets he can find parking places easier and focus on what really matters in his job.
- Gremilde is an elderly woman living in Milan, still very active. During her days in the week she goes around the city to do shopping, waiting her nephews out of the school, meet with some friends, go to the supermarket to buy what's missing, etc... Unfortunately, she was involved several times in little incidents when going away from her parking place due to someone who did a double lane parking. She indeed didn't have to pay any fee but it's very annoying to her having the car grounded at the mechanic. Her older nephew suggested her SafeStreets while talking during an afternoon, from that moment she discovered where she could park without any worries of double line parking violations and also contributed to improve SafeStreets efficiency by sending a

violation notification of a double line parking and help any other member of SafeStreets community in having a better city, a safer city to live.

Use cases:

In the below Figure 18 is shown the User Use Case Diagram with all its use cases. The use case diagram is showing proprietary and SPID⁹ login/signup fused just for the sake of clarity of the diagram, but they are as shown in this paragraph, two different use cases.

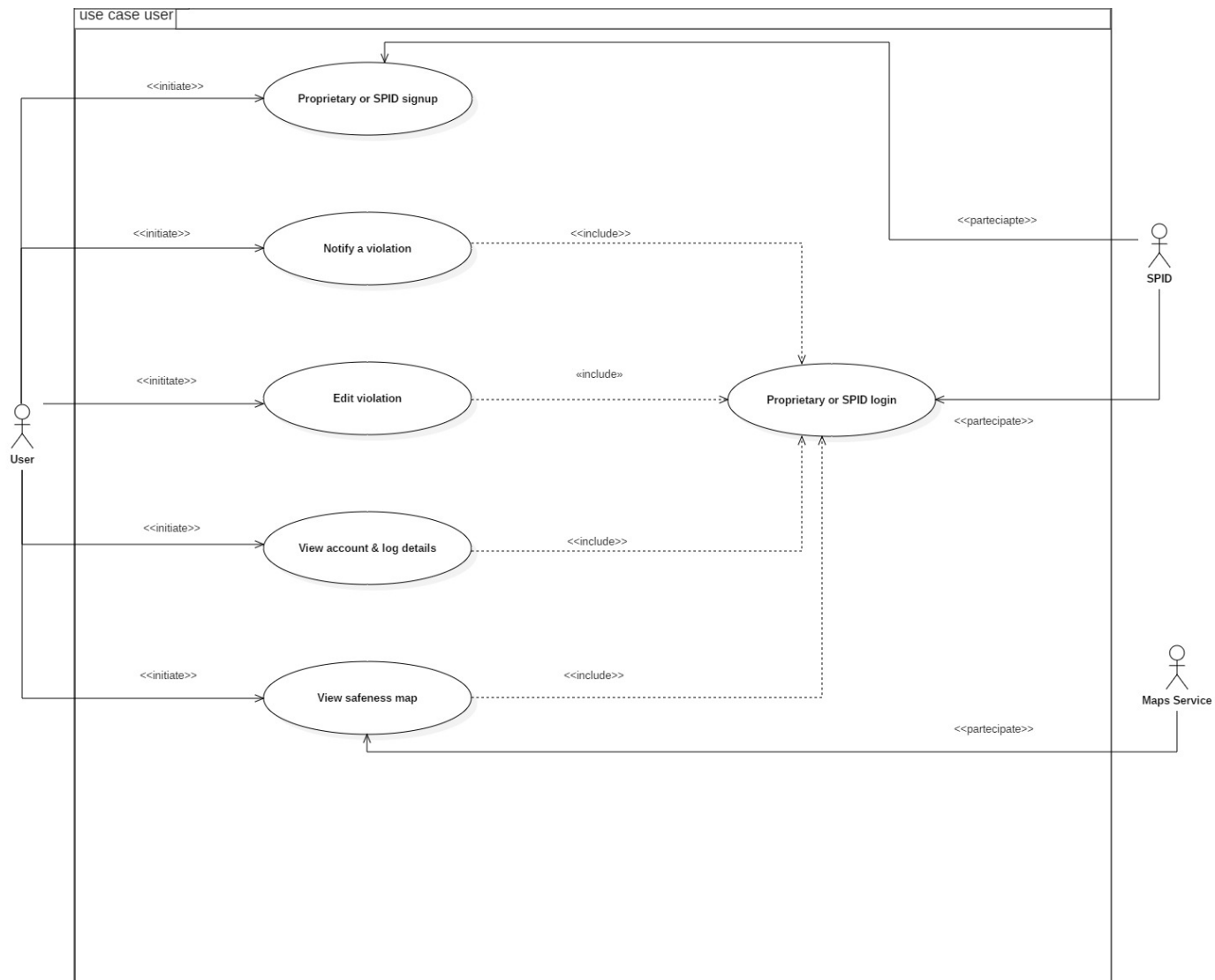


Figure 18 – User Use Case Diagram

Name	SPID signup
Actor	User
Entry condition	The user has opened SafeStreets application after having it downloaded and installed.
Events flow	<ol style="list-style-type: none"> 1) The user chooses at the beginning the proprietary signup option "with SPID". 2) The user is presented SPID authentication form. 3) The user fills his SPID login data. 4) The user authenticates in SPID accepting to share SPID data to SafeStreets. 5) SPID returns to SafeStreets user data. 6) SafeStreets lets the user choose a mandatory username. 7) SafeStreets safely process the user signup request and safely register the user in the system with a higher reliability score.
Exit condition	The user has correctly registered thanks to SafeStreets which saved all his data using SPID.
Exceptions	<ol style="list-style-type: none"> 1) The user has entered wrong data in the SPID authentication form. 2) The user has entered a username already taken by someone else. In this case the user is shown by SafeStreets this error message. 3) SafeStreets app can't register the user due to missing Internet connection. <p>In this case it suggests the user to use the app only when an Internet connection is available.</p>

Name	Proprietary signup
Actor	User
Entry condition	The user has opened SafeStreets application after having it downloaded and installed.
Events flow	<ol style="list-style-type: none"> 1) The user chooses at the beginning the proprietary signup option "Signup in SafeStreets". 2) The user fills every field presents which are all mandatory. 3) The user chooses the confirmation of signup button. 4) SafeStreets safely process the user signup request and safely register the user in the system initializing his reliability score to zero.
Exit conditions	The user has correctly registered thanks to SafeStreets which saved all his data.
Exceptions	<ol style="list-style-type: none"> 1) The user has entered wrong data or inconsistence data or missing data, like a not corresponding fiscal code, birth date, identification document or invalid written email, or invalid password or one or more of these data missing. In these cases, SafeStreets app suggests the user which fields were to be checked due to errors. 2) The user has entered a username already taken by someone else. In this case the user is shown by SafeStreets this error message. 3) The user has entered an email already registered. In this case SafeStreets app suggests the user this error message and invite him to login if that is his email. 4) SafeStreets app can't register the user due to missing Internet connection. In this case it suggests the user to use the app only when an Internet connection is available.

Name	Proprietary login
Actor	User
Entry condition	<ol style="list-style-type: none"> 1) The user has opened SafeStreets application after having it downloaded and installed. 2) The user has already signed up in SafeStreets through proprietary signup.
Events flow	<ol style="list-style-type: none"> 1) The user chooses at the beginning the proprietary login option "Login in SafeStreets". 2) The user fills the authentication form with or the username or the email which used to register and the correlated password. 3) The user chooses the confirmation of login button. 4) SafeStreets process the user login request and login the user in the system.
Exit conditions	The user has correctly logged in.
Exceptions	<ol style="list-style-type: none"> 1) The user has entered a wrong username. In this case SafeStreets app suggests the user to enter the correct login data. 2) The user has entered a wrong email. In this case SafeStreets app suggests the user to enter the correct login data. 3) The user has entered a wrong password. In this cases SafeStreets app suggests the user to enter the correct login data. 4) SafeStreets app can't log in the user due to missing Internet connection from the user side. In this case it suggests the user to use the app only when an Internet connection is available.

Name	SPID login
Actor	User
Entry condition	<ol style="list-style-type: none"> 1) The user has opened SafeStreets application after having it downloaded and installed. 2) The user has already signed up in SafeStreets through SPID.
Events flow	<ol style="list-style-type: none"> 1) The user chooses at the beginning the proprietary login option "Login with SPID". 2) The user fills the SPID authentication form. 3) The user authenticates in SPID. 4) SPID returns to SafeStreets user data which are used to identify the user in SafeStreets. 5) SafeStreets process the user login request and login the user in the system.
Exit conditions	The user has correctly logged in.
Exceptions	<ol style="list-style-type: none"> 1) The user has entered wrong login data in SPID. 2) SafeStreets app can't log in the user due to missing Internet connection from the user side. In this case it suggests the user to use the app only when an Internet connection is available.

Name	Notify a violation
Actor	User
Entry condition	<ol style="list-style-type: none"> 1) The user has opened SafeStreets application after having it downloaded and installed. 2) The user has already logged in in SafeStreets.
Events flow	<ol style="list-style-type: none"> 1) The user chooses from SafeStreets app "notify a violation" button. 2) The user takes a picture of the violation. 3) The system checks for the validity of the violation picture. 4) The system reads the license plate and add its equivalent written text form. 5) The user adds up to three of the allowed types of violations to be notified. 6) SafeStreets app automatically adds other data like date, hour, precise position. 7) The user adds some notes to better explain and to give a better context to the violation. 8) The user taps on the "Send notification" button. 9) The system registers this violation and notify the user of having correctly sent the notification.
Exit conditions	The user correctly sends the notification.

<p>Exceptions</p>	<ol style="list-style-type: none"> 1) The user takes a picture containing no kind of vehicle. In this case SafeStreets notifies the user about this error and asks him to take another picture. 2) The user doesn't select any kind of violations. In this case SafeStreets app suggest the user to fulfil this field. 3) The user doesn't write any note. In this case SafeStreets app suggests the user to fulfil this mandatory field. 4) The user writes a note longer than 140 characters. In this case SafeStreets display a non-error warning and does not allow any more text to be written, it only allows to delete the characters contained in the note. 5) The user takes a picture containing a vehicle which is not identifiable. SafeStreets notifies the user about this error and asks him to take another picture. 6) The user tries to notify a violation which is the same one already made before. SafeStreets display the user this error. 7) SafeStreets app can't send any violation notification due to missing Internet connection from the user side. In this case it suggests the user to use the app only when an Internet connection is available. 8) SafeStreets app can't send any violation notification due to missing satellite link for a correct precise position. In this case it suggests the user to use the app only when a stable satellite link is available.
-------------------	--

Name	Edit violation
Actor	User
Entry condition	<ol style="list-style-type: none"> 1) The user has opened SafeStreets application after having it downloaded and installed. 2) The user has already logged in in SafeStreets. 3) The user has composed a correct violation notification yet to be sent.
Events flow	<ol style="list-style-type: none"> 1) The user notices a wrong license plate read. 2) The user selects the edit license plate button. 3) The user enters the license plate of the identifiable vehicle. 4) The system lowers the quality of the violation notification. 5) The user taps on the "Send notification" button. 6) The system registers this violation and notify the user of having correctly sent the notification.
Exit conditions	The user correctly sends the notification.
Exceptions	<ol style="list-style-type: none"> 1) The user doesn't write any license plate. In this case SafeStreets app suggests the user to fulfil this mandatory field. 2) SafeStreets app can't send any violation notification due to missing Internet connection. In this case it suggests the user to use the app only when an Internet connection is available.

Name	View safeness map
Actor	User
Entry condition	<ol style="list-style-type: none"> 1) The user has opened SafeStreets application after having it downloaded and installed. 2) The user has already logged in in SafeStreets.
Events flow	<ol style="list-style-type: none"> 1) The user enters the "Safeness map" through SafeStreets app. 2) SafeStreets app locates the user. 3) SafeStreets app get violations data and statistics. 4) SafeStreets app shows the user the map centred in his location with the safeness of the areas present in it, with various statistics on vehicles too.
Exit conditions	The user exits the safeness map.
Exceptions	<ol style="list-style-type: none"> 1) SafeStreets app can't display any data due to missing Internet connection from the user side. In this case it suggests the user to use the app only when an Internet connection is available. 2) SafeStreets app can't display any data due to missing satellite link for a correct precise position. In this case it suggests the user to use the app only when a stable satellite link is available.

Name	View account & log details
Actor	User
Entry condition	<ol style="list-style-type: none"> 1) The user has opened SafeStreets application after having it downloaded and installed. 2) The user has already logged in in SafeStreets.
Events flow	<ol style="list-style-type: none"> 1) The user enters the "Account details" section through SafeStreets app. 2) SafeStreets app gets user data. 3) SafeStreets app shows the user details such as: name, notifications of violations made, his reliability score. 4) The user selects one of the notifications of violation made. 5) SafeStreets app gets data about that violation. 6) SafeStreets app shows the user details regarding that notification
Exit conditions	The user exits the account details section.
Exceptions	<ol style="list-style-type: none"> 1) SafeStreets app can't display any data due to missing Internet connection from the user side. In this case it suggests the user to use the app only when an Internet connection is available.

The following sequence diagrams want to focus on the main and critical points of different use cases, for a clarity purpose some trivial conditions are omitted like the fact that there is a need for internet access

to do a successful login, such trivial cases are already covered in the use cases' exceptions list already fully shown above.

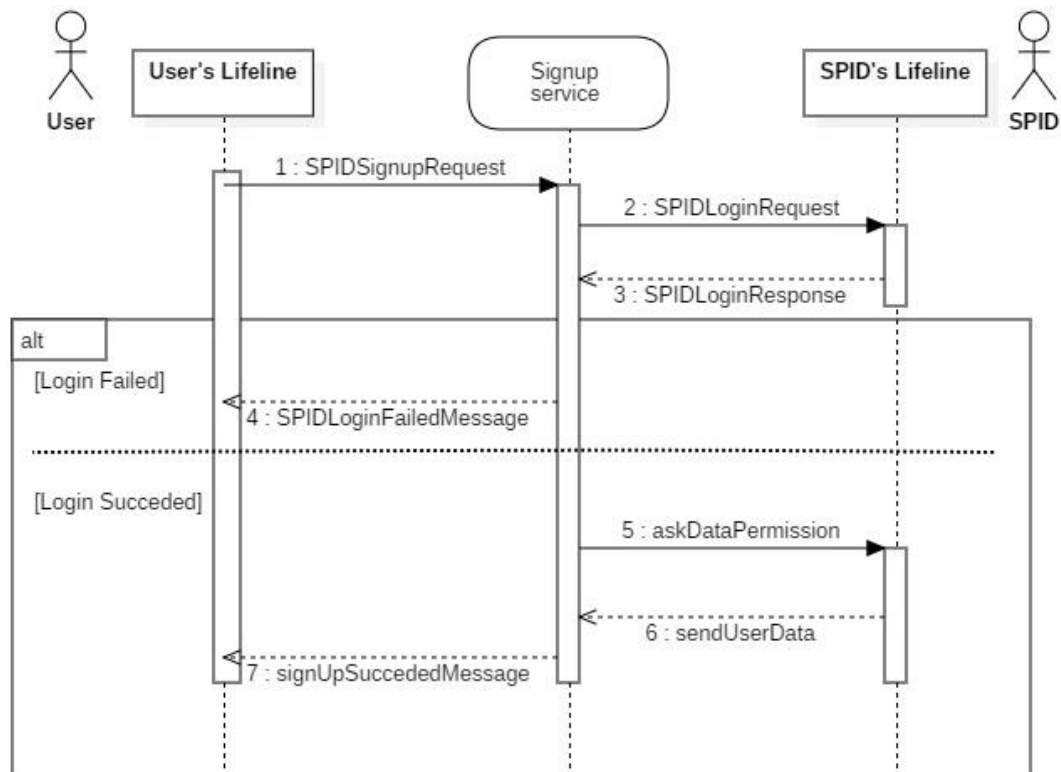


Figure 19 – User SPID Signup Sequence Diagram

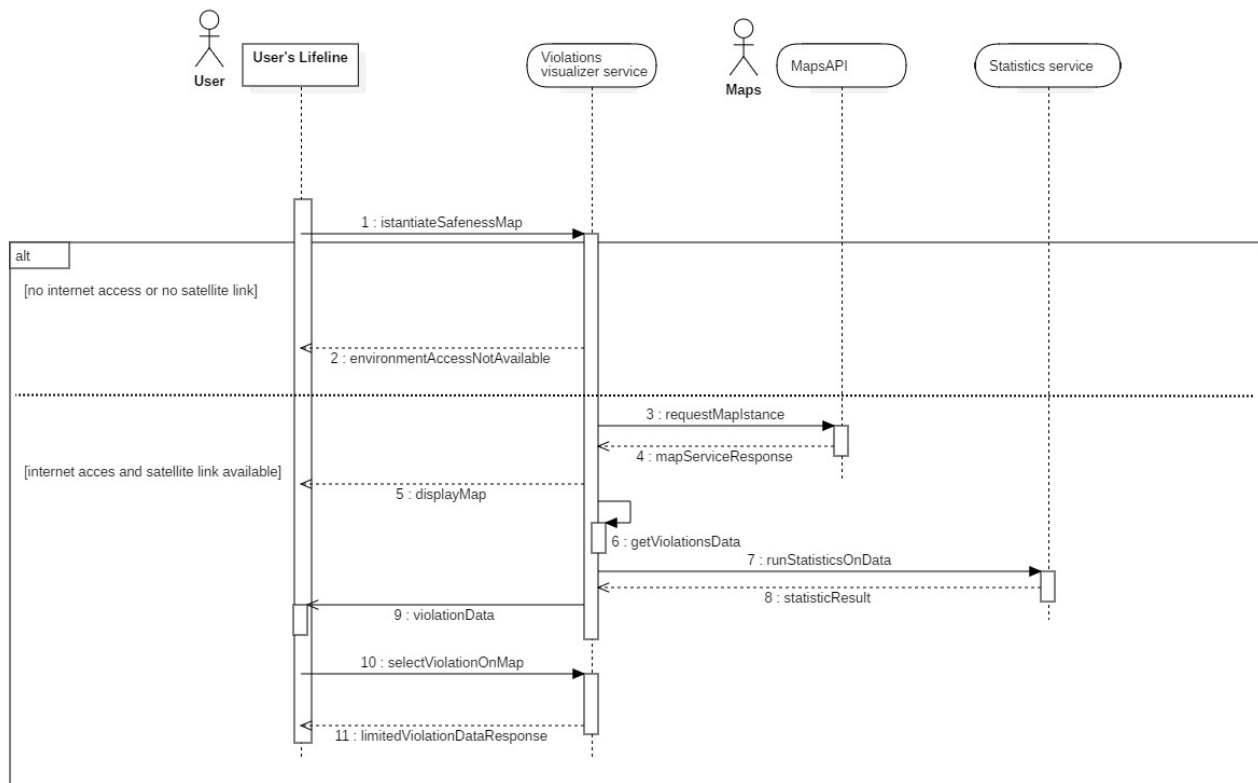


Figure 20 – User accessing Safeness Map Sequence Diagram

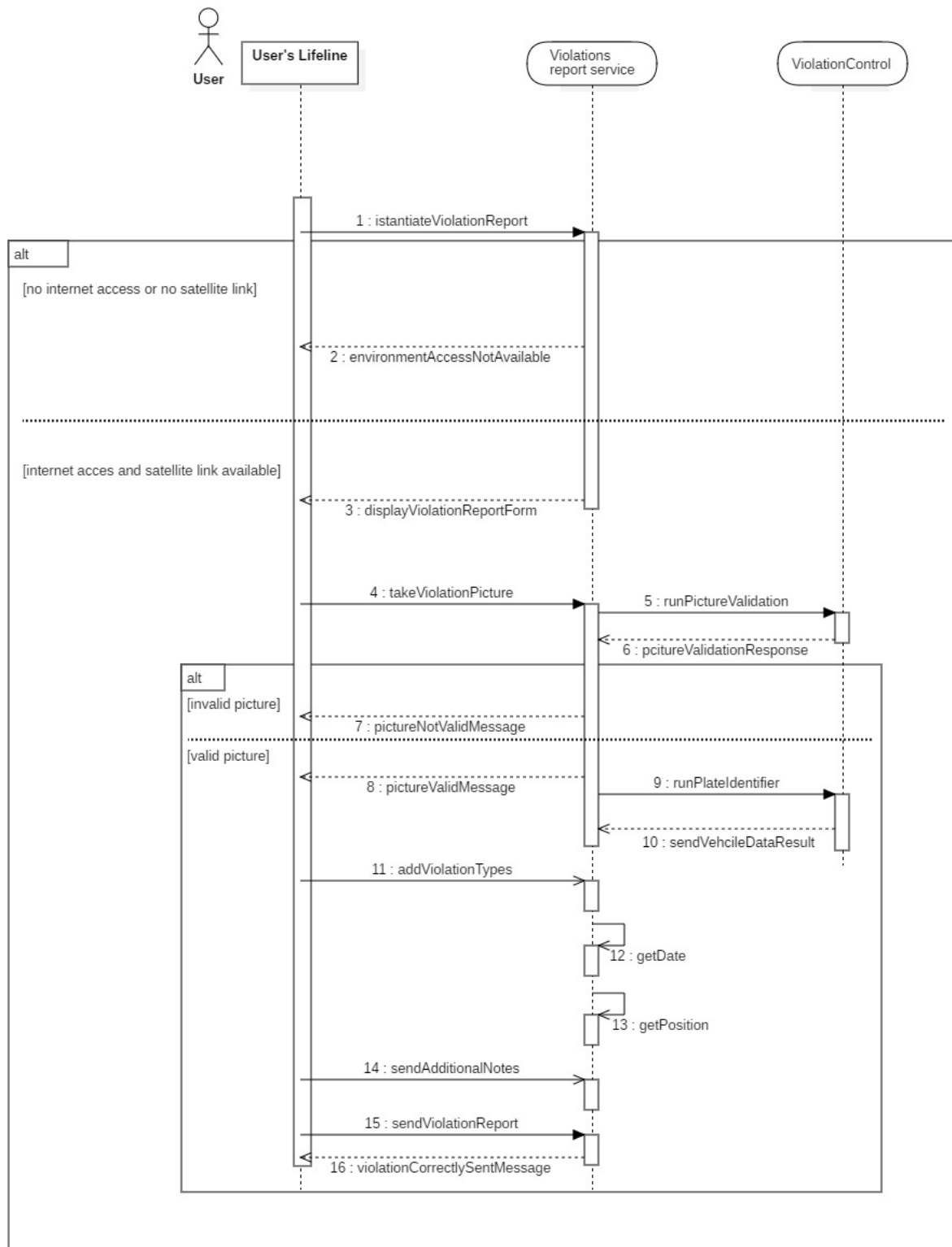


Figure 21 – User reporting a violation Sequence Diagram

3.2.2 Authority

Scenarios:

- Giuseppe is a police officer living in Florence who often receives complaints from citizens, while being on duty in the historic city centre, of various traffic violations committed by tourists which could lead to various little incidents. Unfortunately, police officers can't be everywhere any time due to their large area to cover. When the news of SafeStreets arrives at the police command Giuseppe is very interested in using it. After making a request through the police command to receive credentials from SafeStreets, he logs in and authenticates himself to use SafeStreets authority functionalities by using his provided digital certificate. Few days later he starts receiving various traffic violations notifications, so precise that he can easily go to the target location and can verify that a car was really parking in a reserved place for residents. He immediately writes a fine to the owner of the vehicle. Being very satisfied of this citizen-police cooperation to make Florence a better place that by setting the notification of the violation as certificated, he gives reliability points to the user named Alessandro.
- Camilla is a carabinieri who often receives as duty calls incidents occurred in her near area of duty of Rome. She fortunately usually must manage those incidents which are not so serious but indeed it's quite annoying for everyone involved. While checking her secured email she sees a memorandum of some weeks ago talking about a new application called SafeStreets and its potential applications. She wants to try SafeStreets, it could be very useful to her. After making a request through the police command to receive credentials from SafeStreets, she logs in and authenticates herself to use SafeStreets authority functionalities by using her provided digital certificate from her command centre. The functionality she wants to try first is the Suggestion Infer System, after opening it she can already see the situation in the city and discovers that near Via Santa Domenica Talao, SafeStreets is suggesting to add a towaway zone sign suggest this by showing her that lots of incidents occurred in that same area where lots of notifications of violations regarding vehicle parked in no parking area. She discovers that there were little or no towaway zone sign in that street, by talking at the municipality building of this issue a week later two new towaway zone signs were installed, hoping that those incidents occurring when residents are going out from their home and encounters without seeing a car parked partially in front of their gate, will start to decrease in number anytime soon.

Use cases

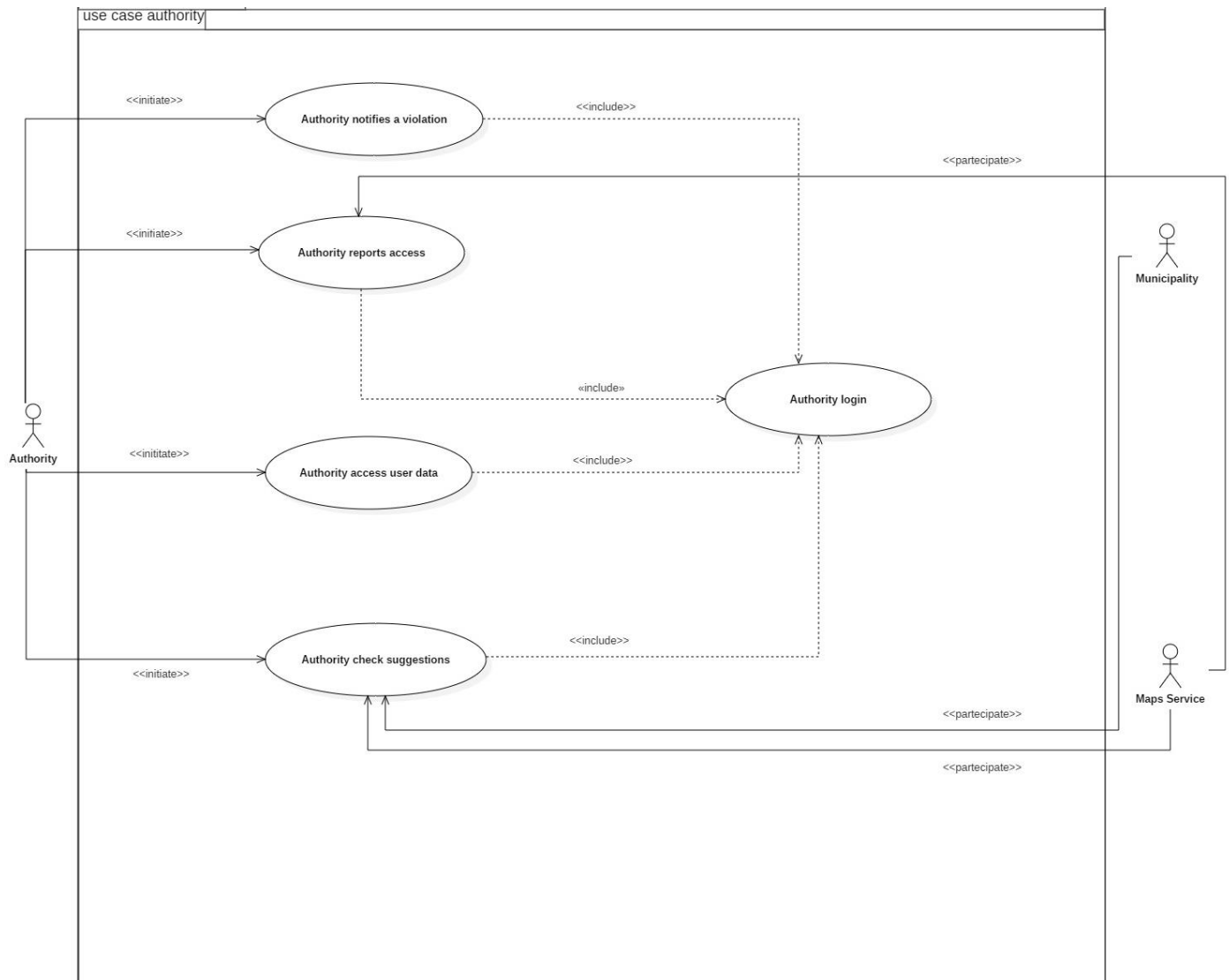


Figure 22 – Authority Use Case Diagram

Name	Authority login
Actor	Authority
Entry condition	Authority has opened SafeStreets application and has received his authentication credentials by SafeStreets after making a formal and direct request.
Events flow	<ol style="list-style-type: none"> 1) The authority chooses at the beginning the proprietary login option "Login in SafeStreets". 2) The authority is presented the SafeStreets authentication form. 3) The authority fills his login data. 4) SafeStreets logs in the authority in the first level of authentication. 5) SafeStreets asks the authority to select his digital certificate. 6) The authority selects his digital certificate. 7) The authority clicks on login confirmation. 8) SafeStreets confirms authority digital certificate. 9) SafeStreets logs in the authority.
Exit condition	The authority has correctly logged in.

Exceptions	<ol style="list-style-type: none"><li data-bbox="1098 181 1559 383">1) The authority has entered a wrong username. In this case SafeStreets app suggests the user to enter the correct login data.<li data-bbox="1098 383 1559 584">2) The authority has entered a wrong email. In this case SafeStreets app suggests the user to enter the correct login data.<li data-bbox="1098 584 1559 786">3) The authority has entered a wrong password. In this cases SafeStreets app suggests the user to enter the correct login data.<li data-bbox="1098 786 1559 987">4) The authority has entered an invalid digital certificate. In this cases SafeStreets app suggests the user to choose the correct digital certificate.<li data-bbox="1098 987 1559 1234">5) SafeStreets app can't log in the authority due to missing Internet connection. In this case it suggests the user to use the app only when an Internet connection is available.
------------	---

Name	Authority reports access
Actor	Authority, User
Entry condition	Authority has opened SafeStreets application, has correctly logged in and has received a notification for new violations notified by SafeStreets users.
Events flow	<ol style="list-style-type: none"> 1) The authority selects the new notification received from SafeStreets. 2) Safestreets gets the new violations and the ones not already verified by an authority. 3) Safestreets presents to the authority the map containing every violation made in the area and below an ordered list of violations. 4) The authority asks to Safestreets to order the list by decreasing date and decreasing reliability score by selecting it in the order type options. 5) Authority selects a violation notification listed and presented in the map by SafeStreets. 6) The authority can see this notification details such as: who made it, picture of the violation, position, license plate, vehicle type and its details such as insurance which is being obtained by external API. 7) The authority verifies this violation notification because he went to the target location. 8) SafeStreets adds reliability score points to the user which notified this violation and notifies him. 9) The authority exits this SafeStreets functionality.
Exit condition	The authority has exited this SafeStreets functionality.

Exceptions	<p>1) SafeStreets app can't access data due to missing Internet connection. In this case it suggests the user to use the app only when an Internet connection is available.</p>
------------	---

Name	Authority check suggestions
Actor	Authority, Municipality
Entry condition	Authority has opened SafeStreets application, has correctly logged.
Events flow	<p>1) The authority chooses "Suggestions" functionality.</p> <p>2) Safestreets check if municipality has new data that it doesn't already downloaded in SafeStreets system before.</p> <p>3) Safestreets app presents to the Authority the result of the inferred data</p> <p>4) The authority asks to Safestreets to order the suggestions by decreasing inferred probability of success.</p> <p>5) The authority exits this SafeStreets functionality.</p>
Exit condition	The authority exits this SafeStreets functionality.
Exceptions	<p>1) SafeStreets app can't access data due to missing Internet connection. In this case it suggests the user to use the app only when an Internet connection is available.</p>

Name	Authority access user data
Actor	Authority, User
Entry condition	Authority has opened SafeStreets application and has been correctly authorized to access the advanced violations reports functionality.
Events flow	<ol style="list-style-type: none"> 1) Authority selects a violation notification listed and presented in the map by SafeStreets. 2) The authority can see this notification details such as: who made it, picture of the violation, position, license plate, vehicle type and its details such as insurance which is being obtained by external API. 3) The authority does not mark as verified this violation notification because he went to the target location. 4) The authority checks user violation notifications history. 5) The authority checks that this user has not done any kind of fake violation notifications in the past. 6) The authority chooses to don't do anything about because this mislead violation notification taken in account as an error due to human nature. 7) The authority exits this functionality.
Exit condition	The authority exits this SafeStreets functionality.
Exceptions	<ol style="list-style-type: none"> 1) SafeStreets app can't access data due to missing Internet connection. In this case it suggests the user to use the app only

	when an Internet connection is available.
--	---

Name	Authority notifies a violation
Actor	Authority
Entry condition	Authority has opened SafeStreets application and has done a secure login using his digital certificate.
Events flow	<ol style="list-style-type: none"> 1) The authority while being on duty find a vehicle in a state of violation. 2) The authority decides to notify the violation to help getting better suggestions regarding possible security measures in his town through inferred suggestion by SafeStreets. 3) The authority chooses from SafeStreets app "notify a violation" button. 4) The authority takes a picture of the violation. 5) The system checks for the validity of the violation picture. 6) The system reads the license plate and add its equivalent written text form. 7) The authority adds up to three of the allowed types of violations to be notified. 8) SafeStreets application will automatically add other data like date, hour, precise position. 9) The authority adds some notes to better explain and to give a better context to the violation. 10) The authority taps on the "Send notification" button. 11) The system registers this violation as made by an authority and notify the authority of having correctly sent the notification.
Exit condition	The authority has exited this SafeStreets functionality.
Exceptions	<ol style="list-style-type: none"> 1) The authority takes a picture containing no kind of vehicle. In this case SafeStreets notifies the authority about this error and asks him to take another picture. 2) The authority doesn't select any kind of violations. In this case SafeStreets app suggest the authority to fulfil this field. 3) The user doesn't write any note. In this case SafeStreets app suggests the user to fulfil this mandatory field. 4) The authority writes a note longer than 140 characters. In this case SafeStreets display a non-error warning and does not allow any

	<p>more text to be written, it only allows to delete the characters contained in the note.</p> <ol style="list-style-type: none"> 5) The authority takes a picture containing a vehicle which is not identifiable. SafeStreets notifies the authority about this error and asks him to take another picture. 6) The user tries to notify a violation which is the same one already made before. SafeStreets display the user this error. 7) SafeStreets app can't send any violation notification due to missing Internet connection from the user side. In this case it suggests the authority to use the app only when an Internet connection is available. 8) SafeStreets app can't send any violation notification due to missing satellite link for a correct precise position. In this case it suggests the authority to use the app only when a stable satellite link is available.
--	--

The following sequence diagrams want to focus on the main and critical points of different use cases, for a clarity purpose some trivial conditions are omitted like the fact that there is a need for internet access to do a successful login or to receive a notification, such cases are already covered in the use cases' exceptions list.

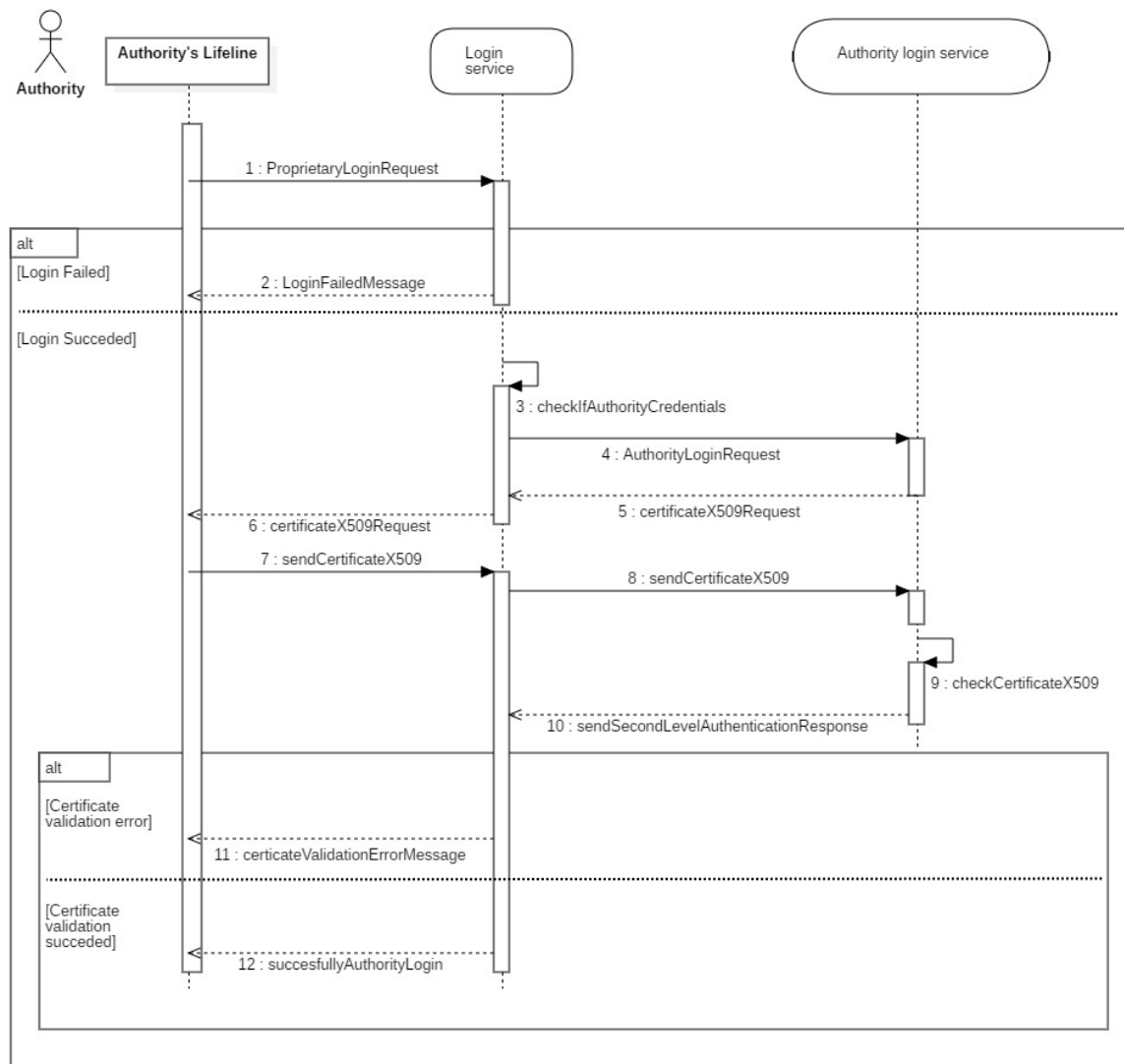


Figure 23 – Authority logging in Sequence Diagram

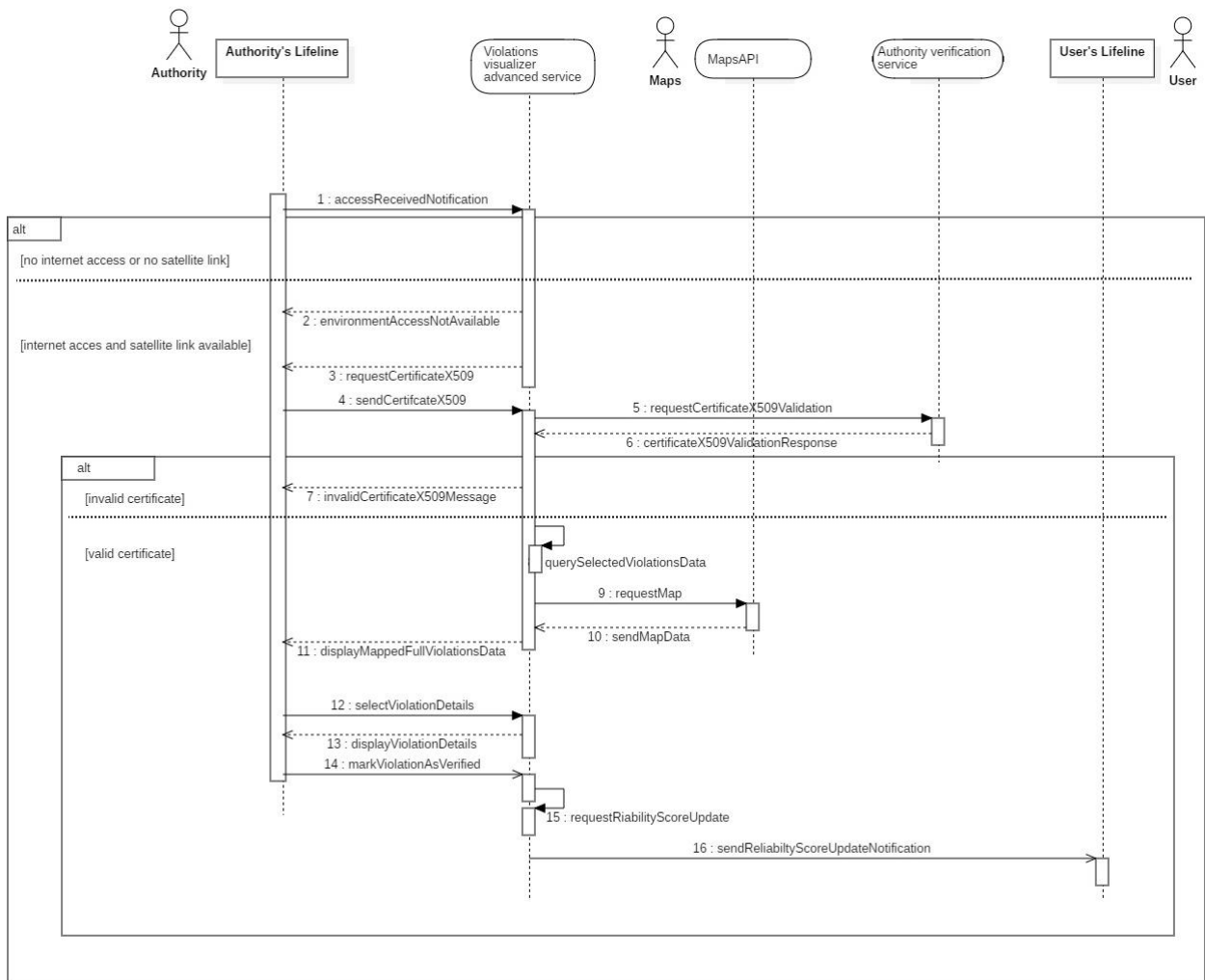


Figure 24 – Authority accessing new reported violations Sequence Diagram

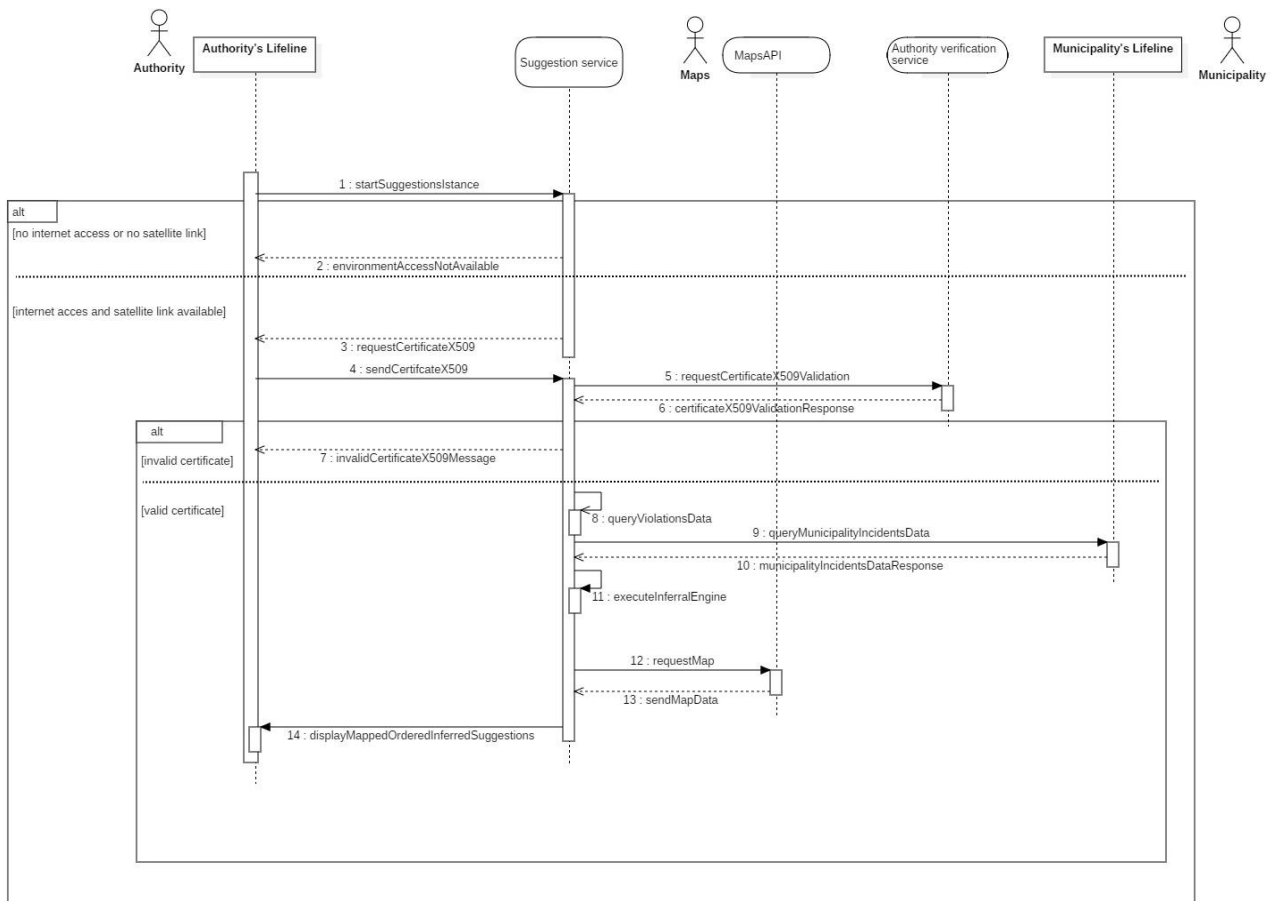


Figure 25 – Authority accessing Suggestions Sequence Diagram

3.2.3 Requirements

[G1] Every registered user should be able to notify violations.

- [R1] The user must be registered to use the application.
- [R2] The user can register, and access, through two different authentication methods: SPID⁹ and proprietary authentication.
- [R3] The user registered with SPID⁹ has a higher initial integrity score than a registered user with proprietary authentication.
- [R4] Each user has an integrity score.
- [R5] Each user can access the details of his own and view his data, integrity score and reports made.
- [R6] Each registration made by a user follows the indications imposed by the Legislative Decree 196/03¹ and the Regulation 2016/679³, which are shown to the user.
- [D1]: Users can identify a violation.
- [D4]: Users have an available internet connection for the whole process of reporting a violation.

[G2] Every recognized authority should be able to access the application.

- [R7] Each authority can access the application through his pre-given credentials and its digital certificate provided by the police forces through the Ministry of the Interior⁵ and the Ministry of the Defense⁶.
- [R8] Each authority can have access to the application features available for users without privileged access.
- [D2]: Authorities possess valid digital certificates provided by the police forces through the Ministry of the Interior⁵ and the Ministry of the Defense⁶.

[G3] Every recognized authority should be able to receive any violation that has been pointed out by a registered user.

- [R9] Each authority has full access to the reports made.
- [R10] Each authority has full access on vehicle information pinpointed by the external Italian license plate verifier and by State Police portal.
- [R11] Each authority can access the details of the report made and of the user who carried it out according to the terms established by the Legislative Decree 196/03¹ and the regulation 2016/679³.
- [R12] Any authority shall receive notifications regarding the municipality in his current position.
- [D2]: Authorities possess valid digital certificates provided by the police forces through the Ministry of the Interior⁵ and the Ministry of the Defense⁶.

[G4] Every user's communication must be a precise violation report that has been committed by a recognizable vehicle.

- [R13] A report must consist of an image, date, time, location and metadata.
- [R14] The metadata of a report is the type of report, the quality of the report and the notes entered by the user.
- [R15] The notes entered by the user cannot be longer than 140 characters.

- [R16] Date, time and location must be added automatically via the Internet and GPS/Galileo satellites.
- [R17] The user can proceed with the violation signalling if the GPS location is precise and obtained securely.
- [R18] It is possible to report in the presence of an Internet connection only.
- [R19] User reporting image is recognized as valid for reporting only if it contains a vehicle that can be identified through the license plate.
- [R20] The system must be able to recognize the vehicle registration number.
- [R21] The user can decide to modify the result of the reading of the license plate made by the system.
- [R22] The system includes for each report a warning if the user has modified the vehicle license plate in which the notification will have reported a lower quality.
- [R23] The user can't send multiple times the same violation report
- [R1] The user must be registered to use the application.
- [R2] The user can register, and access, through two different authentication methods: SPID⁹ and proprietary authentication.
- [D1]: Users can identify a violation.
- [D3]: Users decide to modify the license number providing the right one if and only if the recognized license number, through the application, isn't the correct one.
- [D4]: Users have an available internet connection for the whole process of reporting a violation.
- [D5]: Every considered geographical area is covered by a satellite location system (e.g. GPS and Galileo).

[G5] Every registered end user should be able to mine general information about the violations committed in a certain area.

- [R24] Each user can access a map showing the security level in certain areas.
- [R25] Each user can have limited access to reports by viewing information that does not violate the privacy of the reporting user according to the Legislative Decree 196/03¹ and the regulation 2016/679³.
- [R26] Each user can view statistics based on reports made in certain areas.
- [D4]: Users have an available internet connection for the whole process of reporting a violation.

[G6] Every recognized authority must be able to verify the notified violations by the registered users

- [R27] Authorities can indicate an alert as verified through the application.
- [R28] Each alert verified by an authority will give the user who has indicated it a higher reliability score.
- [D6]: Authority acts based on notifications about violations or suggestions about security improvement.

[G7] Every recognized authority must be able to receive suggestions about improving the local security.

- [R29] The system must be able to access the accident data present in a specific municipal area if present.

- [R30] The system must analyse accidents and violations data to produce a suggestion to be notified to the authority to improve road safety.
- [R31] Any authority shall receive suggestions regarding the municipality in his current position.
- [D6]: Authority acts based on notifications about violations or suggestions about security improvement.
- [D7]: The quantity of both SafeStreets data and those provided by the municipality, if available, is enough to infer suggestions to authorities.

Traceability matrix

In the following table it will be present a mapping between requirements, goals and the use cases shown. This is done to give to the reader a better context to understand which requirements satisfies which goals and which of goals and requirements are shown in the various use cases presented.

Requirement	Goal	Use case
R1	G1	Proprietary login, SPID login, Notify a violation
R2	G1	Proprietary signup, SPID signup, Notify a violation
R3	G1	SPID signup
R4	G1	Proprietary signup, SPID login
R5	G1	View account & log details
R6	G1	View account & log details
R7	G2	Authority login
R8	G2	Authority notifies a violation
R9	G3	Authority access user data, Authority reports access
R10	G3	Authority reports access
R11	G3	Authority access user data, Authority reports access
R12	G3	Authority reports access
R13	G4	Notify a violation
R14	G4	Notify a violation
R15	G4	Notify a violation
R16	G4	Notify a violation
R17	G4	Notify a violation
R18	G4	Notify a violation
R19	G4	Notify a violation
R20	G4	Notify a violation
R21	G4	Edit violation
R22	G4	Edit violation

R23	G4	Notify a violation
R24	G5	View safeness map
R25	G5	View safeness map
R26	G5	View safeness map
R27	G6	Authority reports access
R28	G6	Authority reports access
R29	G7	Authority check suggestions
R30	G7	Authority check suggestions
R31	G7	Authority check suggestions

3.3 Performance Requirements

SafeStreets handles reports' formats and then it sends those reports to authorities. Since it doesn't require any strictly low period for a report to be handed to the authorities, from authorities' view, it mustn't exceed 5 minutes for the report to be notified, instead, from user's view, it mustn't exceed 10 seconds for the process of the report to be sent to the authorities. And the user's view performance is strictly important since it guarantees the usability of the application in terms of satisfaction of the user himself.

3.4 Design Constraints

3.4.1 Standards compliance

SafeStreets provides to each user an integrity score to describe the credibility of violation reports of each user and how much users are using the application in order to increase the safety level of the areas they are interested in.

Regarding the privacy of data, since SafeStreets possesses sensitive information about its users, such as fiscal code and full name, the entire application is subject to the GDPR³, a regulation in EU law on data protection and privacy for all individuals within the EU and the EEA.

Since SafeStreets handles authorities' registration into the application, it must verify that the appliers as authorities are really authorities. Thus, SafeStreets uses a valid digital certificate as a standard for the verification of authorities' identity. In order to identify authorities, SafeStreets requires, only for whom is registered as an authority, a valid digital certificate according to D.P.C.M. of 30.10.2014 N.193 art.21 and art.22⁴.

3.4.2 Hardware limitations

Since SafeStreets is an application that only sends and receives data, it doesn't require a hardware infrastructure. However, as mentioned on the "Dependencies and constraints" section, to be able to run this application, a device must have both internet connection and a location system (for further information, see DD). At the same time this is not a hard constraint because nowadays electronic devices with a defined OS contain all the required characteristics so, these hardware limitations won't create any unavoidable obstacle that could affect the usability of SafeStreets.

3.5 Software System Attributes

3.5.1 Reliability

SafeStreets system must run without any kind of interruptions due to malfunctions or anything

else. To achieving that, it's fundamental to ensure a strong fault tolerance, which can be done by parallelization of duplicated and dislocated resources like main servers, inner processes and by using a cloud infrastructure logic with a dynamically allocated resource pool, also called elastic load balancer.

3.5.2 Availability

Parallely to reliability, Safestreets must guarantee to be continuously available to every user with downtimes resolved in the quickest time possible. It's possible to achieve this with well managed and a not over complex system, maintaining also a good trade off to guarantee reliability. Here are listed the expected availability for the main functionalities of SafeStreets:

- Violations Notifier needs to have an expected availability of 99.99 %, fundamental to notify to authorities the violations occurring in a municipality.
- Area Safeness needs to have an expected availability of 99.999 %, because that's the main reason users will use this app: knowing which area is safer.
- Suggestion Notifier needs to have an expected availability of 99.9%, little downtimes
will have no arm since there will be no expected high or frequently drastic changes in data concerning violations and incidents occurring in a certain municipality.

3.5.2 Security

Security is a very important topic discussed in this document for SafeStreets. Any archived information, communication and exchange of data in the network shall be encrypted in the most secure way to guarantee the highest level of privacy requested by the Legislative Decree 196/03¹, Legislative Decree 82/05² and the regulation 2016/679³. Secret authentication data needs to be archived in such a way that is not possible in any way and for anyone to get those data.

3.5.3 Maintainability

SafeStreets shall be developed in such a way that future additional functionalities or any required fix will be done with the lowest cost possible. This document is providing some hints on the main logic behind the system, but it will be discussed in a further document the better approach and the best design pattern to achieve maintainability.

3.5.4 Portability

SafeStreets shall be developed to be the more portable as possible, which means should work across a vast range of devices such as different smartphone vendors and computers. Achieving this will not be so hard by taking advantages of vendors APIs and for example by utilizing an approach like Software as Service in a cloud infrastructure.

4. Formal Analysis using Alloy

In this section an analysis of some critical aspects of the system is provided exploiting Alloy.

The focus is on some static constraints, in particular:

- It cannot happen that two different users have the same username or that two different users have the same fiscal code and so on for identifying personal information.
- It cannot happen that an authority does not have a digital certificate.
- It cannot happen that a user does two times the same violation

Then, some other necessary, but less important, structural constraints are expressed and properly commented in the Alloy formal notation. These bonds are verified analysing the worlds generated running the model and checking some relevant assertion.

```
--signature to obtain a true and false value
abstract sig Bool{}
sig True extends Bool{}
sig False extends Bool{}

--image of a violation signature
sig Image {
  accepted : one Bool
}

--license plate of a vehicle committing a violation signature
sig LicensePlate{
  modified : one Bool
}

--authentication types for SafeStreets users
abstract sig AuthenticationType {}
sig SPIDAuthentication extends AuthenticationType {}
sig ProprietaryAuthentication extends AuthenticationType {}

--user details signatures
sig Username{}
sig Password{}

--authentication of a specific user signature
sig Authentication {
  authenticationType : one AuthenticationType,
  username : one Username,
  password : one Password
}
```

```

--other user details
sig Name{}
sig Surname{}
sig AuthenticatorID{}
sig FiscalCode{}
sig Address{}
sig PhoneNumber{}
sig Email{}

--location signatures
sig Position{}
sig City{
    position : one Position
}

--abstract user signature containing various details
abstract sig User {
    name : one Name,
    surname : one Surname,
    authenticatorID : one AuthenticatorID,
    authentication : one Authentication,
    reportsMade : set Violation,
    city : one City
}{{#authentication.authenticationType=1 or #authentication.authenticationType=2}}

--normal user signature containing also more personal details
sig NormalUser extends User {
    reliabilityScore : one Int,
    birthDate : one Date,
    fiscalCode : one FiscalCode,
    birthLocation : one City,
    phoneNumber : one PhoneNumber,
    address : one Address,
    email : one Email,
    view : one ViolationVisualizerLimited
}{{reliabilityScore>=0}}

--authority signature which is more restrictive
sig Authority extends User {
    authorization : one DigitalCertificateX509,
    notification : set Violation,
    suggestions : set SuggestionsType,
    email : one CertifiedEmail,
    view : ViolationVisualizerPro
}

--date signature for determining time
sig Date{}

--signatures for creating a digital certificateX509

```

```

sig ID{}
sig DigitalCertificateX509{
  creationDate : one Date,
  dateOfExpiry : one Date,
  id : one ID
} {creationDate!=dateOfExpiry}

--certified email for authorities
sig CertifiedEmail{}

--violation details signatures and violation signature declaration

sig VehicleType{}
sig ViolationType{}
sig Note{}

sig Violation{
  violationID : one Int,
  violationType : some ViolationType,
  position : one Position,
  timeStamp : one Date,
  vehicleType: one VehicleType,
  image : one Image,
  licensePlate : one LicensePlate,
  note : one Note,
  email : one Email,
  authenticatorID : one AuthenticatorID,
  verified : one Bool
} {#violationType = < 3 and violationID >= 0}

--map displaying a certain area signature
sig Map{}

--visualizers signature with different level of visibility
--containing various violations
abstract sig ViolationVisualizer{
  map : one Map,
  violation : some Violation
}
sig ViolationVisualizerLimited extends ViolationVisualizer{}
sig ViolationVisualizerPro extends ViolationVisualizer{}

--a certain municipality data signature
sig MunicipalityData {}

--represents Municipality_i
sig Municipality {
  city : one City,
  incidents : one MunicipalityData
}

```

--inferring system of SafeStreets signature

```
sig SuggestionsType {}
```

```
sig SuggestionInferalEngine {  
  suggestions : some SuggestionsType,  
  municipalities : set Municipality,  
  violations : some Violation  
}
```

--no duplicate username is possible so that a certain username is associated with a specific user

```
fact noDuplicateUsernameNormalUser{  
  all u1, u2: NormalUser | (u1.authentication.username = u2.authentication.username) implies u1 = u2  
}
```

--no duplicate user personal details is possible among different users

```
fact noDuplicateNormalUser {  
  all u1, u2 : NormalUser | (u1.autheticatorID = u2.autheticatorID or u1.fiscalCode = u2.fiscalCode or u1  
.email = u2.email  
    or u1.authentication = u2.authentication or u1.phoneNumber = u2.phoneNumber) implies u1 =  
u2  
}
```

--each authority has his own personal username which is not replicable

```
fact noDuplicateUsernameAuthority{  
  all u1, u2: Authority | (u1.authentication.username = u2.authentication.username) implies u1 = u2  
}
```

--no duplicate authority detailed data is possible among different authorities

```
fact noDuplicateAuthority {  
  all u1, u2 : Authority | u1 = u2  
    iff (u1.autheticatorID = u2.autheticatorID or u1.email = u2.email or u1.authentication = u2.authenti  
cation)  
}
```

--

needed to guarantee there not exist any kind of equivalent authorization among different user of every kind

```
fact noDuplicateAuthorization{  
  all a1, a2 : Authority | a1.authorization = a2.authorization implies a1 = a2  
}
```

--

for obvious, natural reasons, a city has its own position which is not shared along any other different cities

```
fact cityPosition{  
  all c1, c2 : City | c1.position = c2.position implies c1 = c2  
  all u : User | some c : City | c = u.city  
}
```

```

--
digital certificates X509 cannot have any kind of duplicates to be valid and to respect security measures
and constraints
fact noDuplicateDigitalCertificateX509{
    all a1, a2: Authority | a1.authorization.id = a2.authorization.id implies a1.authorization = a2.authorization
    all a : Authority | some c : DigitalCertificateX509 | c = a.authorization
    all c1, c2 : DigitalCertificateX509 | c1 = c2 iff c1.id = c2.id
}

--
needed to guarantee a quite important requirement which is that no user can report multiple times the
same violation
fact noDuplicateViolationsFromUser {
    all v1, v2 : Violation |
        (v1.timeStamp = v2.timeStamp or v1.email = v2.email or v1.autheticatorID = v2.autheticatorID)
        implies v1.violationID = v2.violationID
    all u : User | some v : Violation | v in u.reportsMade
    all v1, v2 : Violation | v1 = v2 iff v1.violationID = v2.violationID
    all u1, u2 : User | no v : Violation | u1 != u2 and v in u1.reportsMade and v in u2.reportsMade
}

fact noDuplicateViolationsPerUser{
    all u : NormalUser | some v : Violation | v in u.reportsMade implies (v.email = u.email
        and (some a : Authority | v in a.notification))
}

--
this fact guarantees that there are no violations where one of them have a violation type list containing
violation type duplicates
fact noDuplicateViolationTypes {
    all vt, vt', vt'' : ViolationType, v: Violation |
        ((#v.violationType=3 and vt in v.violationType and vt' in v.violationType and vt'' in v.violationType)
            implies (vt != vt' and vt' != vt'' and vt != vt'')) or
        ((#v.violationType=2 and vt in v.violationType and vt' in v.violationType) implies (vt != vt'))
}

--this guarantees a better reliability score for users logging in with the SPID authentication system
fact reliabilityScoreInit {
    all u, u' : User | (u!=u' and u.authentication.authenticationType = SPIDAuthentication
        and u'.authentication.authenticationType = ProprietaryAuthentication
        and #u.reportsMade = 0 and #u'.reportsMade = 0) implies (u.reliabilityScore > u'.reliabilityScore)
}

--
this fact is used to check whether a violation has an assigned value of verification which is false as default
fact verificationManagement{
    all v : Violation | v.verified = False iff v.verified != True
}

```

```

--
this guarantees the possibility of an authority to increase a certain violation reporting user reliability score in case of
--his violation was verified by this authority
fact reliabilityScoreManagement {
  all u, u' : User | u = u' and (some x : Int | x > 0 and (u'.reliabilityScore = u.reliabilityScore + x)) iff
    (some v : Violation | v in u.reportsMade and v.verified = True)
}

--notification of violations
fact notificationFromUserToAuthority{
  all u: User | #u.reportsMade>0 iff (some a : Authority | let r = u.reportsMade | r in a.notification)
}

--user accessing service to create new violations
fact violationNotifier{
  all u : User | some v : Violation | v.email = u.email implies v in u.reportsMade
}

--needed to guarantee the autonomous notification system based on authorities current positions
fact notifyWhomManagement{
  all u : User, a : Authority | let r = u.reportsMade | r in a.notification iff (r.position = a.city.position)
}

--this grants the possibility to check on various SafeStreets statistics
fact statisticsVisualization{
  all v : ViolationVisualizer | some u : User | v = ViolationVisualizerPro implies u = Authority
  all v : ViolationVisualizer | some u : User | v = ViolationVisualizerLimited implies u = NormalUser
  all v1, v2 : ViolationVisualizer | v1 = v2 iff v1.map = v2.map
}

fact statistics{
  all v : ViolationVisualizer | #v.violation>0 iff (some u : User | #u.reportsMade>0)
}

--this following facts are needed to guarantee a working suggestion inferring system
fact MunicipalityDataManagement{
  all s : SuggestionInferalEngine | #s.suggestions>0 implies (#s.municipalities>0 and (some v : Violation | #v>0))
  all s : SuggestionInferalEngine | all m : Municipality | m in s.municipalities and #m.incidents>0
    and (all m' : Municipality | (m.city = m'.city or m.incidents = m'.incidents) iff m = m')
  all m1, m2 : Municipality | m1.city = m2.city implies m1 = m2
}

fact suggestionsActivation{
  all s : SuggestionInferalEngine, v : Violation | #s.suggestions>0 implies (#s.municipalities>0 and #v>0)
}

```

```

fact MunicipalityDataProvided{
  all s : SuggestionInferalEngine | (#s.municipalities>0 and (some u : User | #u.reportsMade>0))
  implies (some a : Authority | s.suggestions in a.suggestions)
}

--world regarding SafeStreets accesed by registred authorities and users
pred Registration {
  #NormalUser = 1
  #Authority = 1
}

run Registration for 3 but 0 SuggestionsType, 0 MunicipalityData

--
world regarding the visualization of the violations map with the different levels of visibility depedning o
n the user type
pred VisualizerMap {
  #ViolationVisualizerLimited = 1
  #ViolationVisualizerPro = 1
}

run VisualizerMap for 2 but 0 SuggestionsType, 0 MunicipalityData

--world regarding the new suggestions inferred by SafeStreets by using municipality data
pred Suggestions{
  #SuggestionInferalEngine = 1
  #Municipality = 2
}
run Suggestions for 3

--
really important assert to garantuee authority access to his various assigned special SafeStreets function
alities
assert AuthorityRecognition {
  no a : Authority | no c : DigitalCertificateX509 | a.authorization.id = c.id
}
check AuthorityRecognition

--
The presented Alloy code was made to be as clear as possible considering the respective UML which will
be more detailed for obvious reasons

```

Here are the worlds generated through the predicates. In these first two worlds it was wanted to have a smaller world to better explain the behaviours of the various components.

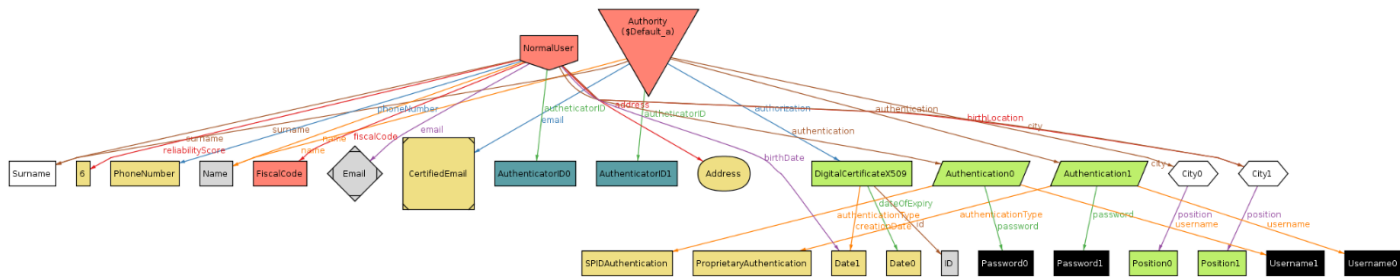


Figure 26 – World 1

From the first world in figure 26 we can easily check how a user and an authority have their various personal details. An authority has its own digital certificate and its type of authentication where it is shown to have its specific details.

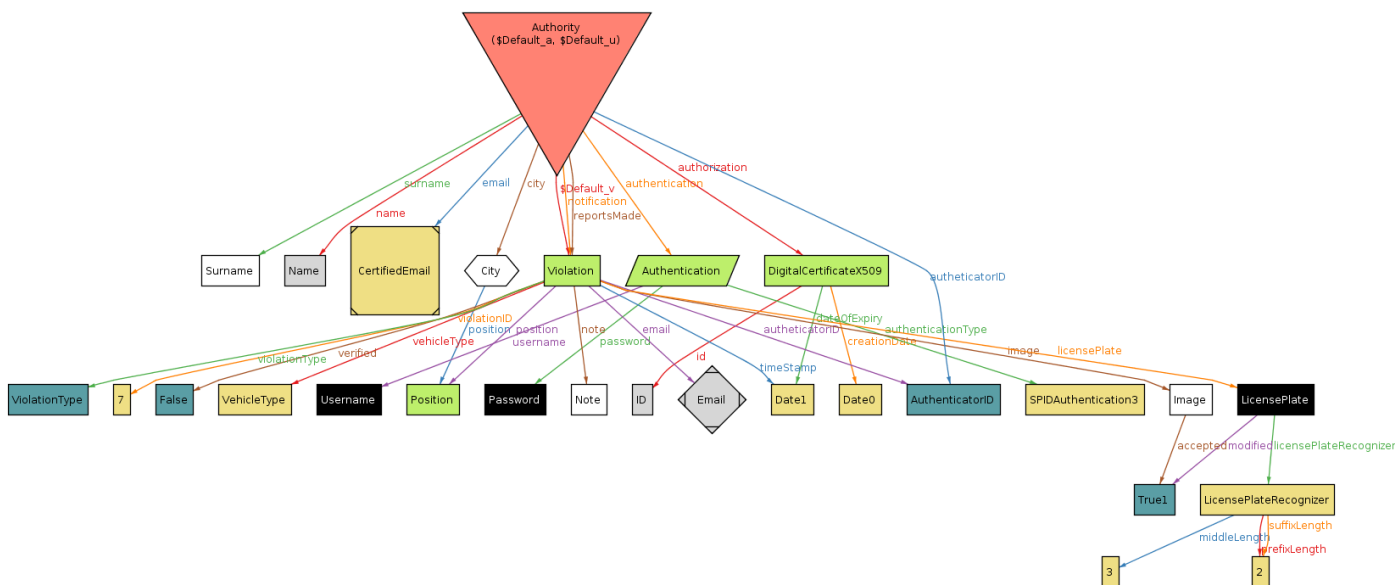


Figure 27 – World 2

In the second world in figure 27 it is instead shown in more details how an authority is composed, including a violation he can have made. A license plate of the vehicle involved in the violation it is shown with its details with the case where a user, where in this case is an authority, has decided to modify it because he believes it was a bad reading. An image of the violation is always present as shown and its status to true represent its final acceptance.

In this fourth world in the figure 29 it is shown how normal users and authorities can interact with the safeness map which has indeed different level of visualization depending on which users is asking this feature to SafeStreets. The authority will interact with the violation visualizer pro, where the normal user will have only access to the violation visualizer limited. They will indeed show various or even same violations depending on the position on which a map inside a certain visualizer is focused to.

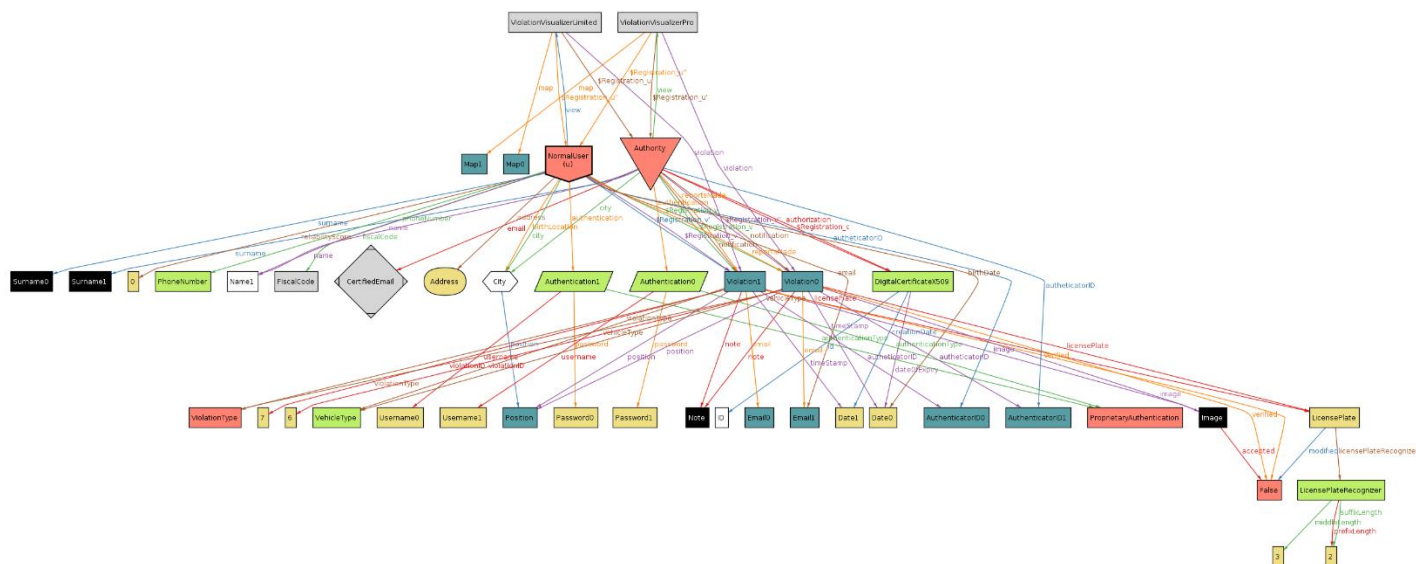


Figure 30 – World 5

In this world fifth represented in figure 30 it was done to better show a version of the world including even more information to comprehend much details as possible such various visualizers, violations made and visualized and so on. Of course, more details at the cost of a less clean diagram for obvious reasons.

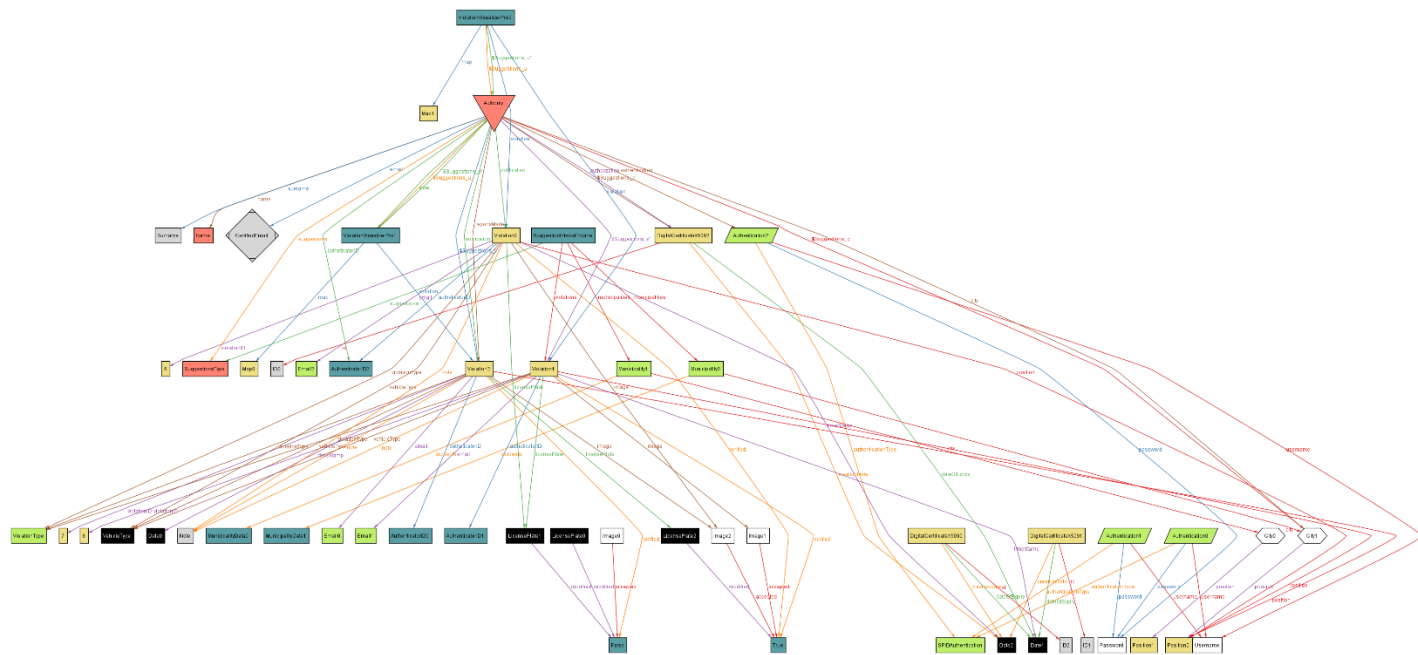


Figure 31 – World 6

In this sixth and last world it is shown the case of a running suggestion inferring system of SafeStreets in which an authority has access to. He will get information inferred by using Municipalities data about incidents occurred in their own area. This computation will eventually result in a specific, possible, suggestion available to the current accessing authority.

It is important to see that each municipality is of course associated with a specific city which then has its municipality data that will be used to infer new suggestions depending on this data changing over time together with the always increasing number of violations reported by various user in this same city where this certain municipality is located.

It is also necessary to show in the end the results of the alloy analyser where worlds are generated, and assertion are correctly verified.

Executing "Run Suggestions for 3 but 0 NormalUser"

Solver=sat4j Bitwidth=4 MaxSeq=3 SkolemDepth=1 Symmetry=20
6119 vars. 528 primary vars. 11669 clauses. 25ms.
Instance found. Predicate is consistent. 19ms.

Executing "Run Registration for 3 but 0 SuggestionsType, 0 MunicipalityData"

Solver=sat4j Bitwidth=4 MaxSeq=3 SkolemDepth=1 Symmetry=20
7709 vars. 618 primary vars. 14714 clauses. 29ms.
Instance found. Predicate is consistent. 24ms.

Executing "Run VisualizerMap for 2 but 0 SuggestionsType, 0 MunicipalityData"

Solver=sat4j Bitwidth=4 MaxSeq=2 SkolemDepth=1 Symmetry=20
3523 vars. 318 primary vars. 6810 clauses. 23ms.
Instance found. Predicate is consistent. 14ms.

Executing "Run Suggestions for 3 but 0 NormalUser"

Solver=sat4j Bitwidth=4 MaxSeq=3 SkolemDepth=1 Symmetry=20
6119 vars. 528 primary vars. 11669 clauses. 18ms.
Instance found. Predicate is consistent. 15ms.

Executing "Check AuthorityRecognition"

Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
8206 vars. 654 primary vars. 15603 clauses. 27ms.
No counterexample found. Assertion may be valid. 12ms.

5. Effort spent

Daniele Comi

Task	Effort spent
Definitions, acronyms, abbreviations	0.5
Reference documents	0.5
References	0.5
Overview	1
Versions	0.5
Product perspective	1
Product functions	2
Functional requirements	10
User interfaces	6
Software system attributes	2
Alloy	2
Goals and requirements	4
UML	13

Anton Ghobryal

Task	Effort spent
Purpose	0.5
Scope	2
User characteristics	1
Constraint	0.5
Assumptions and dependencies	4
External interface requirements	2
Performance requirements	2
Design constraints	2
Alloy	24
Goals and requirements	4
UML	0.5

6. References

1. D.L. 196 of 2003 (196/03) <https://www.camera.it/parlam/leggi/deleghe/Testi/03196dl.htm>
2. D.L. 82 of 2005 (82/05) <https://docs.italia.it/italia/piano-triennale-ict/codice-amministrazione-digitale-docs/it/v2017-12-13/index.html>
3. General Data Protection Regulation (EU) 2016/679 <https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32016R0679>
4. D.P.C.M. 193 of 2014 (193/2014) <http://www.prefettura.it/alessandria/allegati/Download:D.p.c.m. 193 2014 regolamento banca dati-554036.htm>
5. Ministry of the Interior and digital certificates released <http://politichepersonale.interno.it/ita/index.php?IdMat=1&IdSot=35&IdNot=386>
6. Ministry of the Defence and digital certificates released <http://www.pkiff.difesa.it/#secEN>
7. Certified Email <https://www.agid.gov.it/it/piattaforme/posta-elettronica-certificata>
8. Certified Email RFC <https://tools.ietf.org/html/rfc6109>
9. SPID <https://www.agid.gov.it/it/piattaforme/spid>
10. Traffic regulation and laws <http://www.aci.it/i-servizi/normative/codice-della-strada.html>
11. Alloy documentation <http://alloytools.org/documentation.html>