POLITECNICO
MILANO 1863

Academic Year 2019 / 2020

Safe Streets

Cappelletti Andrea, Maglione Sandro

Requirement Analysis and Specification Document (RASD)

Deliverable:	RASD
Title:	Requirement Analysis and Verification Document
Authors:	Cappelletti Andrea, Maglione Sandro
Version:	1.0
Date:	10 November 2019
Download page:	https://github.com/andreacappelletti97/CappellettiMaglione
Copyright:	Copyright © 2019, Cappelletti Andrea, Maglione Sandro – All rights reserved

x

Contents

1 Introduction

1.1 Purpose

The following document represents the Requirements Analysis and Specification Document (RASD) for the crowd-sourced application *SafeStreets*. The goals of this document are:

- to describe the full application domain;
- to explore the real needs of the stakeholders in order to model the system in accordance to their necessities;
- to understand and communicate the requirements for the system to be developed;
- to assess the constraints and limits of the system;
- to clarify assumptions and dependencies;
- to present the main scenarios and use cases of the end-product;
- to present formal and semi-formal models required to provide a better understanding of the system;

The intent of this document is to support the developers who will implement and validate the requirements. It may also be used as a contractual basis.

1.2 Scope

1.2.1 Description of the given problem

SafeStreets is a crowd-sourced application that intends to provide users with the possibility to notify authorities when traffic violations occur, and in particular parking violations. The application allows users to send pictures of violations, including their date, time, and position, to authorities. Examples of violations are vehicles parked in the middle of bike lanes or in places reserved for people with disabilities, double parking, and so on.

1.2.2 Description of the requirements

SafeStreets stores the information provided by users, completing it with suitable meta- data. In particular, when it receives a picture, it runs an algorithm to read the license plate (one can also think of mechanisms with which the user can help with the recognition), and stores the retrieved information with the violation, including also the type of the 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 could be offered to different roles.

If the municipality offers a service that allows users to retrieve the information about the accidents that occur on the territory of the municipality, SafeStreets can cross this information with its own data to identify potentially unsafe areas, and suggest possible interventions (e.g., add a barrier between the bike lane and the part of the road for motorized vehicles to prevent unsafe parking).

1.2.3 System to be

The SafeStreets application is completely new and therefore it will be built from scratch. SafeStreets aims to utilize a crowd-sourced approch to solve the problem of parking violations. Every user will we able to report violations by sending a picture and some additional information, namely date, time, position, and license plate. SafeStreets will then store the data that will support the authorities.

SafeStreets will integrate accidents information from the municipality, if available, to identify and suggest possible interventions in unsafe areas.

1.2.4 Goals

In this section we list the goals of the system. We divided the goals related to the authorities from the goals of normal users.

1.2.5 Normal users

- **[G1]** Allow user to report a violation
  - **[G1.1]** Allow user to send a picture
  - **[G1.2]** Allow user to select type of violation
  - **[G1.3]** Allow user to input the license plate
- **[G2]** Allow user to save their settings and activity in a private account
  - **[G2.1]** Allow user to sign up by providing his/her name, surname, email and password
  - **[G2.2]** Allow user to sign in using his/her email and password
- **[G3]** Allow user to visualize information about violations in a specific area
  - **[G3.1]** Allow user to visualize potentially unsafe areas
  - **[G3.2]** Allow user to visualize the violations' frequency of an area
- **[G4]** Allow user to visualize all report that he/she sent to the system

1.2.6 Authorities

- **[G5]** Allow authority to visualize information about violations
  - **[G5.1]** Allow authority to search information of a specific license plate
  - **[G5.2]** Allow authority to visualize a list of the cars associated with most violations reported
  - **[G5.3]** Allow authority to visualize a list of violations in a specific area
- **[G6]** Allow authority to receive information about violations report in real-time
  - **[G6.1]** Allow authority to subscribe to a specific area to be notified about violations
  - **[G6.2]** Allow authority to subscribe to a specific license plate to be notified about violations
  - **[G6.3]** Allow authority to be notified about violations nearby his/her current position
  - **[G6.4]** Allow authority to validate violations through a notification
- **[G7]** Allow authority to receive recommendations about possible interventions

1.3 Definitions, Acronyms, Abbreviations

1.3.1 Definitions

To better understand the overall document we provide a set of definitions.

- **System:** The software to be developed;
- **User:** A person that uses the system;
- **Normal user:** A person that uses the system which is not responsible for the streets' security;
- **Authority:** A person employed by the municipality who is responsible for the security of the streets' network;
- **Municipality:** A clearly defined territory where all the people living on that territory share one common *local* government;
- **Position:** The current phisical location of the mobile device of a User in terms of Latitude and Longitude;
- **Location:** Another term to refer to 'Position';
- **Area:** A district of a city consisting in a collection of streets which are connected together and which have an horizontal extension similar to the vertical extension;
- **UI:** The visual design of a screen of the application;
- **Real-time:** An update is in real-time if it contains data that has been downloaded right after it has been correctly reported by another user and stored to the system.

1.3.2 Acronyms

- **API:** Application Program Interface;
- **ML:** Machine Learning;
- **UI:** User Interface;
- **GPS:** Global Positioning System.

1.3.3 Abbreviation

- **GN:** Goal number N;
- **DN:** Domain assumption number N;
- **RN:** Requirement number N.

1.4 Document Structure

This specification document follows the standard suggested by IEEE and therefore is composed of different sections.

Section 1: in which the purpose of the document is explained, followed by the scope, which provides the description of the problem, the application domain and the system to be, and the goals. Moreover, the terminology used across the document is explained to prevent misunderstandings and clarify unambiguity.

Section 2: in which the context of the product is clearly outlined. Firstly, the main interactions with the outside world, in terms of world and shared phenomena, are defined. The product funtionalities are then discussed in relation the the user characteristics and interactions with the service. Finally, constraints, assumptions, and dependencies are specified.

Section 3: in which the requirements of the application are stated. The first section presents an overview of the external interface requirements. A list of the functional requirements is then provided, followed by some clarifications about further constraints, performance requirements, and finally a description of the software system attributes in terms of non-functional requirements.

Section 4: in which some scenarios are presented. Each section represents a possible use case of the application in a real life situation. Each scenarios is focused on a specific functionality of the System.

Section 5: in which semi-formal UML diagrams are presented to graphically describe the functionality and various interactions of the System. Those diagrams are not intended to model any design specification. They aim to provide a general overview of the application, the actors involved and their concrete interaction with the System. In particular, we present some use case, statechart, class, and sequence diagrams, which we believe are appropriate for model requirement specifications.

Section 6: in which a formal Alloy analysis of the System is presented and commented. We model the System using Alloy, defining the proper signatures, facts, and assertions. We then run the model to test its formal consistency. We also present and discuss some world generated by the tool.

Section 7: in which a list of the tools used and the hours of work is outlined. The hours of work are presented for each memeber of the team. Each working session has been registered; we provide the date, a description of the work, the total amount of time and the start and end time for each session.

2 Overall Description

2.1 Product perspective

In this section, in order to provide an high-level description of the features of SafeStreets, we outline the World and Shared phenomena that concern our application. Particularly we consider world phenomena, that are relevant for the system but they are also out of control of our application and shared phenomena, that can be observed and controlled by the application.

- **World Phenomenon:** A User wants to report a violation.
  Shared Phenomenon (controlled by the world): A User send a picture of the violation and inputs the type through the application. The license plate will also be request if the software is unable to recognize it from the picture.
- **World Phenomenon:** A User wants to consult his/her report.
  Shared Phenomenon (controlled by the world): A User, after performing a log in operation, is redirected to his/her reports' history.
- **World Phenomenon:** A User location changes.
  Shared Phenomenon (controlled by the world): A new User location is detected by the User device through GPS.
- **World Phenomenon:** A User wants to know the safetiness of an area.
  Shared Phenomenon (controlled by the world): A User inputs the area of interest and the application displays the information requested
- **World Phenomenon:** A User wants to know the safetiness of an area.
  Shared Phenomenon (controlled by the world): A User inputs the area of interest and the application displays the information requested

- **World Phenomenon:** An Authority wants to a list of the violations of an license plate.
  Shared Phenomenon (controlled by the world): An Authority inputs the license plate of interest and the application displays the list requested
- **World Phenomenon:** An Authority wants to a be notified about real-time violations.
  Shared Phenomenon (controlled by the world): An Authority activated the notification system and the application starts listening to real-time updates

2.2 Product functions

In this section we list the main features that will be available in the system. The provided list is intended as a breif explaination about some of the main features, more details will be provided in the following sections of the document.

- **User account management:** In order to be granted access to the system, each user must have registered his/her own account. The system provides a form for the registration, which collects the user name, surname, email, and password. The user will then validate his/her credentials and he/she will then be allow to access the service by providing the correct email and password to another dedicated form;
- **Violation reporting:** The System allows the user to user to take a picture of the violation. It then attaches information about date and time (timestamp), the position (through the GPS), the type of violation (input by the user), the license plate (input by the user or recognized by the System), and it stores the complete report;

- **Visualize violation data:** The System retrieves the data stored about the violation, it organizes it and displays it in a human-readable UI, suited for the specific needs or role of the user. The data retrieved can be specified by the authority through a search system;
- **Real-time notifications:** The System sends real-time notifications to authorities which have activated the notification service.
- **Recommendations:** The System will integrate municipality's data about accidents, if present, in order to identify potentially unsafe areas and suggest possible interventions to authorities.

2.3 User characteristics

The user will interact with the system for two main reasons: send reports and visualize information about violations. In this context, the user could be any person who lives in the municipality of interest. We make a general distinction in two categories, normal users and authorities:

- Normal user: Interested in the well-being and security of the streets. He/She supports the municipality by sending reports of violations in his/her area. He/She can then consult data about violations collected with the help of other citizens and provided by the system;
- Authority: Person whose job is to assure the security of the streets. He/She exploits the system by searching and analyizing data about violations, in order to be able to predict possible accidents and highlight potentially unsafe areas. An authority has access also to the functionalities provided to normal users.

2.4 Constraints

- The system must have access to the GPS position, the internet connection, and the camera of the user device.
  - None of those services will be exploited for any other purpose outside the normal application functionalities as stated;
  - The service in its entirity cannot be provided if an internet connection is missing;
  - The report service will not be available in case the system will not have access to either the GPS or the camera.
- Personal user information (name, surname, email, license plate) will not be used for commercial purposes;
- Every institution who is allowed to search and mine data about violations (i.e. authorities) must have a formal authorization from the municipality. Furthermore, anonymity is always guaranteed in all the services provided by the application.

2.5 Assumptions and dependencies

- **[D1]:** The user sends the report right after having taken the picture. Therefore, the GPS position represents the correct violation's location
- **[D2]:** When a user sends a report, he/she is reporting a real violation
- **[D3]:** Every account email is unique
- **[D4]:** Every real-time violation update is not older than 5 minutes

- **[D5]:** Every user that has access to the System has an account registered and he/she is currently logged in
- **[D6]:** The report is stored into the system after the end of sending procedure in less than 10 seconds
- **[D7]:** An accurate location can be retrived through the GPS when it is activated (by market standard: <= 10 meters)
- **[D8]:** The camera of the device is available when requested, otherwise the operating system will signal an error message
- **[D9]:** User's device interfaces correctly with the network after a connection has been established
- **[D10]:** Internet connection permission has to be available in all the system components
- **[D11]:** Data are correctly transferred after a request is sent, otherwise an error message is returned
- **[D12]:** Municipality data about accidents, when available, is correct
- **[D13]:** When a PEC email address is recognized, the User is upgrated to Authority and the System is allowed to grant the authorities' functionalities safetly
- **[D14]:** An area with more violations and accidents is considered less safe
- **[D15]:** Every User's device that interacts with the System has a GPS, a camera, and an internet connection available
- **[D16]:** Every Authority has an advanced knowledge about traffic laws and violations, therefore they can validate violations' pictures.

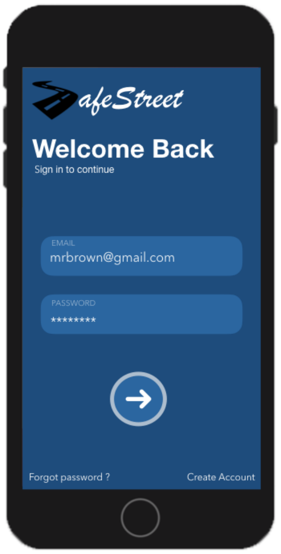
3 Specific Requirements

In this section we present all the requirements.

3.1 External interface requirements

3.1.1 SafeStreet Mobile Application

This section contains some of the SafeStreet mobile application mock-ups.
Their aim is to show a visual representation of how the final UI should look like.

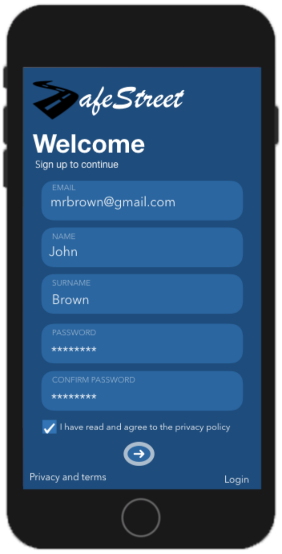


Welcome Back
Sign in to continue

EMAIL mrbrown@gmail.com

PASSWORD ********

Figure 1: SafeStreets login page



Welcome
Sign up to continue

EMAIL mrbrown@gmail.com

NAME John

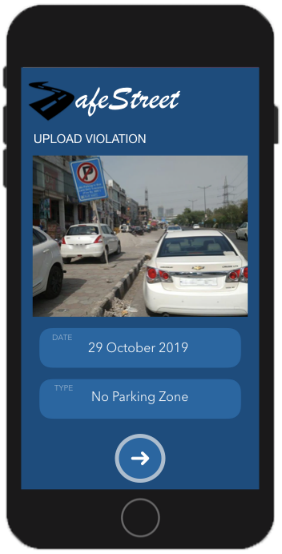
SURNAME Brown

PASSWORD ********

CONFIRM PASSWORD ********

☑ I have read and agree to the privacy policy

Figure 2: SafeStreets registration page



UPLOAD VIOLATION

DATE 29 October 2019

TYPE No Parking Zone

Figure 3: SafeStreets violation report upload

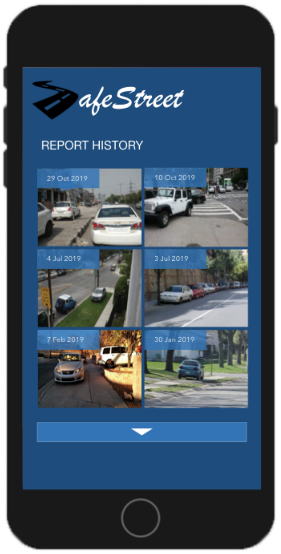
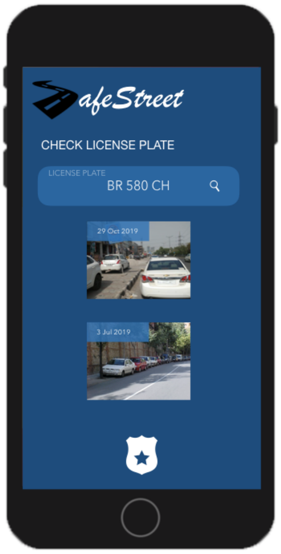
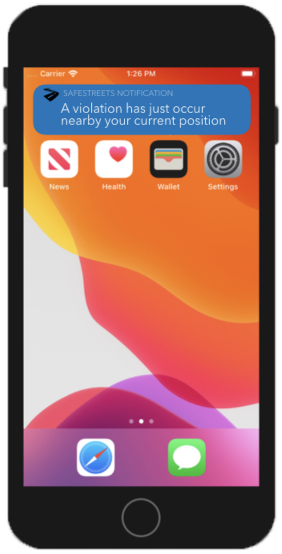


Figure 4: SafeStreets report history



LICENSE PLATE BR 580 CH

Figure 5: SafeStreets license plate research



A violation has just occur nearby your current position

Figure 6: SafeStreets notification to Authority

3.2 Functional requirements

3.2.1 Functional requirements list

- **[R1]:** After a Visitor correctly fills and sends the registration form, he/she is allowed to sign in to the System as a User
- **[R2]:** A User is upgraded to Authority if the System recognizes a PEC email address
- **[R3]:** If the User credentials are correct, he/she is logged in by the System
- **[R4]:** The System visualizes the user's previous sent reports when requested
- **[R5]:** A User can access the camera to take a picture of the violation

- **[R6]:** After a picture has been taken, the System provides a form to the user to insert the type of violation
- **[R7]:** The System displays the safetiness of an area when requested
- **[R8]:** The System displays the frequency of the violations of an area when requested
- **[R9]:** When a report has been stored, the System associates the report to the User who sent it
- **[R10]:** A User can select the area that he/she is interested in
- **[R11]:** An Authority can input a license plate
- **[R12]:** The System attaches the GPS position when a report is sent
- **[R13]:** The Authority can toggle the notification service
- **[R14]:** The System periodically checks the GPS position of an Authority if he/she has the notification service enabled
- **[R15]:** If a violation is detect nearby the Authority position and he/she has the notification service activated, the System sends a notification
- **[R16]:** An Authority is periodically updated about violation in an area if he/she has activated the notification service in that area
- **[R17]:** When information about a license plate is available, the System returns it
- **[R18]:** An account is registered if and only if the email provided is not already present into the System
- **[R19]:** The System suggests possible interventions in an area when requested by an Authority
- **[R20]:** The User must insert the license plate if the System is unable to recognize it from the picture
- **[R21]:** If a User reports more than one violation of the same license plate in the space of 24 hours, only one violation will be considered valid, thus uploaded.
- **[R22]:** A notification is sent to an authority every time a license plate reaches ten not yet validated violation reports.

3.2.2 Goals, requirements, assumptions

- **[G1]:** Allow user to report a violation
  - **[R5]:** A User can access the camera to take a picture of the violation
  - **[R6]:** After a picture has been taken, the System provides a form to the user to insert the type of violation
  - **[R9]:** When a report has been stored, the System associates the report to the User who sent it
  - **[R12]:** The System attaches the GPS position when a report is sent
  - **[R20]:** The User must insert the license plate if the System is unable to recognize it from the picture
  - **[R21]:** If a User reports more than one violation of the same license plate in the space of 24 hours, only one violation will be considered valid, thus uploaded.

  - **[D1]:** The user sends the report right after having taken the picture. Therefore, the GPS position represents the correct violation's location
  - **[D2]:** When a user sends a report, he/she is reporting a real violation
  - **[D5]:** Every user that has access to the System has an account registered and he/she is currently logged in
  - **[D6]:** The report is stored into the system after the end of sending procedure in less than 10 seconds
  - **[D7]:** An accurate location can be retrived through the GPS when it is activated (by market standard: <= 10 meters)
  - **[D8]:** The camera of the device is available when requested, otherwise the operating system will signal an error message
  - **[D9]:** User's device interfaces correctly with the network after a connection has been established
  - **[D10]:** Internet connection permission has to be available in all the system component
  - **[D11]:** Data are correctly transferred after a request is sent, otherwise an error message is returned
  - **[D15]:** Every User's device that interacts with the System has a GPS, a camera, and an internet connection available

- **[G2]:** Allow user to save their settings and activity in a private account

  - **[R1]:** After a Visitor correctly fills and sends the registration form, he/she is allowed to sign in to the System as a User
  - **[R2]:** A User is upgraded to Authority if the System recognizes a PEC email address
  - **[R3]:** If the User credentials are correct, he/she is logged in by the System
  - **[R18]:** An account is registered if and only if the email provided is not already present into the System

  - **[D3]:** Every account email is unique
  - **[D9]:** User's device interfaces correctly with the network after a connection has been established
  - **[D10]:** Internet connection permission has to be available in all the system components
  - **[D13]:** When a PEC email address is recognized, the User is upgrated to Authority and the System is allowed to grant the authorities' functionalities safetly

- **[G3]:** Allow user to visualize information about violations in a specific area

  - **[R7]:** The System displays the safetiness of an area when requested
  - **[R8]:** The System displays the frequency of the violations of an area when requested
  - **[R10]:** A User can select the area that he/she is interested in

  - **[D5]:** Every user that has access to the System has an account registered and he/she is currently logged in
  - **[D9]:** User's device interfaces correctly with the network after a connection has been established
  - **[D10]:** Internet connection permission has to be available in all the system components

  - **[D11]:** Data are correctly transferred after a request is sent, otherwise an error message is returned
  - **[D14]:** An area with more violations and accidents is considered less safe
  - **[D15]:** Every User's device that interacts with the System has a GPS, a camera, and an internet connection available

- **[G4]:** Allow user to visualize all report that he/she sent to the system

  - **[R4]:** The System visualizes the user's previous sent reports when requested

  - **[D5]:** Every user that has access to the System has an account registered and he/she is currently logged in
  - **[D9]:** User's device interfaces correctly with the network after a connection has been established
  - **[D10]:** Internet connection permission has to be available in all the system components
  - **[D11]:** Data are correctly transferred after a request is sent, otherwise an error message is returned
  - **[D13]:** When a PEC email address is recognized, the User is upgrated to Authority and the System is allowed to grant the authorities' functionalities safetly
  - **[D15]:** Every User's device that interacts with the System has a GPS, a camera, and an internet connection available

- **[G5]:** Allow authority to visualize information about violations

  - **[R2]:** A User is upgraded to Authority if the System recognizes a PEC email address
  - **[R11]:** An Authority can input a license plate
  - **[R17]:** When information about a license plate is available, the System returns it

  - **[D5]:** Every user that has access to the System has an account registered and he/she is currently logged in
  - **[D9]:** User's device interfaces correctly with the network after a connection has been established
  - **[D10]:** Internet connection permission has to be available in all the system components
  - **[D11]:** Data are correctly transferred after a request is sent, otherwise an error message is returned
  - **[D13]:** When a PEC email address is recognized, the User is upgrated to Authority and the System is allowed to grant the authorities' functionalities safetly
  - **[D15]:** Every User's device that interacts with the System has a GPS, a camera, and an internet connection available

- **[G6]:** Allow authority to receive information about violations report in real-time

  - **[R2]:** A User is upgraded to Authority if the System recognizes a PEC email address
  - **[R13]:** The Authority can toggle the notification service
  - **[R14]:** The System periodically checks the GPS position of an Authority if he/she has the notification service enabled
  - **[R15]:** If a violation is detect nearby the Authority position and he/she has the notification service activated, the System sends a notification

  - **[R16]:** An Authority is periodically updated about violation in an area if he/she has activated the notification service in that area
  - **[R22]:** A notification is sent to an authority every time a license plate reaches ten not yet validated violation reports.

  - **[D4]:** Every real-time violation update is not older than 5 minutes
  - **[D5]:** Every user that has access to the System has an account registered and he/she is currently logged in
  - **[D9]:** User's device interfaces correctly with the network after a connection has been established
  - **[D10]:** Internet connection permission has to be available in all the system components
  - **[D11]:** Data are correctly transferred after a request is sent, otherwise an error message is returned
  - **[D13]:** When a PEC email address is recognized, the User is upgrated to Authority and the System is allowed to grant the authorities' functionalities safetly
  - **[D15]:** Every User's device that interacts with the System has a GPS, a camera, and an internet connection available
  - **[D16]:** Every Authority has an advanced knowledge about traffic laws and violations, therefore they can validate violations' pictures.

- **[G7]:** Allow authority to receive recommendations about possible interventions
  - **[R2]:** A User is upgraded to Authority if the System recognizes a PEC email address
  - **[R19]:** The System suggests possible interventions in an area when requested by an Authority

  - **[D5]:** Every user that has access to the System has an account registered and he/she is currently logged in
  - **[D9]:** User's device interfaces correctly with the network after a connection has been established
  - **[D10]:** Internet connection permission has to be available in all the system components
  - **[D11]:** Data are correctly transferred after a request is sent, otherwise an error message is returned
  - **[D12]:** Municipality data about accidents, when available, is correct
  - **[D13]:** When a PEC email address is recognized, the User is upgrated to Authority and the System is allowed to grant the authorities' functionalities safetly
  - **[D15]:** Every User's device that interacts with the System has a GPS, a camera, and an internet connection available

3.3 Design constraints

3.3.1 Regulatory policies

The system deals with sensible information about the user. The system must be granted access to the GPS and the camera of the device. Furthermore, the system also collects some information about each user (name, surname, email). Therefore, the application must comply to the a set of regulations about privacy and security which are enforced by the government. Those regulations might limit the scope of the service or expand the required functionalities.

Moreover, the System stores information about license plates. These data is inserted by the User or recognized from the picture of the violation automatically. No further information or personal data will be associated with the owner of the vehicle's license plate. The license plate will be used to store a unique reference to the vehicle in order to display a list of violation related to the same anonymous individual. Nonetheless, only certified authorities will be granted access to license plates.

3.3.2 Hardware and software limitations

The system relies heavily on the data supplied by the device sensors (GPS, camera), therefore the following hardware limitations must be taken into account:

- The location detected by the GPS may differ from the exact position of the user and the violation, while being within 10 meters in most cases
- The GPS module may not be able to detect the correct location at all times
- The camera may not be avaliable or be compromised

3.4 Performace requirements

The system must support up to 100'000 users registered, 30'000 daily active user which amount for 120'000 daily requests. The latency time to retrieve data about violations must be lower than 10 seconds for every user request. The latency time to store the report must be lower than 5 seconds.

3.4.1 Scalability

The system must be scalable 10 times the original workload, reaching 1'000'000 users registered, within 1 year from deployment. The response time requirements must still be met by the new workload scale.

3.5 Software system attributes

In this section we define some further requirements not directly related to functional behaviors but still relevant to the success of the overall user experience. Therefore, for each of the following categories, we define some minimal requirements that the system should meet.

3.5.1 Reliability

The system must ideally guarantee the full avaliability of the service at any time, aiming to reach an up time above 99% and as close as possible to 100%, with only sporadic connection errors permitted (1 in 100'000 requests).

3.5.2 Security

The system must guarantee the complete anonymity of every user that sends a request, without exception. No details about violations must be granted to normal users outside of an overview of unsafe areas and the number of reported violations. No data can be accessed by unregistered users.

3.5.3 Accuracy

The GPS must provide position's data with an error lower than 10 meters. Furthermore, if the service fails to detect the device position in less than 30 seconds, the report is aborted to avoid to compromise the correctness of the violation position.

3.5.4 Maintainability

The system must be designed to be flexible, modular and easy to maintain. Those requirements will be necessary to allow the system to be expanded easily with the addition of new features without compromising the existing functionalities. Furthermore, since the system relies heavily on the data sent by the users, the database system must be flexible and consistent. Therefore, specific strategies should be put in place to assure the correctness of the data provided, as well as the possibility of expanding the amount of information avaliable for each report to hone the quality of the service in future versions of the application.

3.5.5 Portability

The system is designed to be used on a mobile device. Therefore, the client side part of the application should work as long as the device on which is installed has a functioning network connection, GPS system, and camera. The server side part of the application can be moved to another server as long as the server configuration is duplicated on the new machine.

4 Scenarios

In this section we provide some context-related examples to explain the feature of SafeStreets' system. This does not intend to be a complete overview of all the features.

4.1 User first access and registration

Mr Brown can not bear parking violations and he wants to help authorities to fix this problem. Surfing the web, he finds out SafeStreets and decides to download the mobile application and use it. This is the very first time he uses the application, so he has to fill the registration form providing name, suername, email and password. Before completing the registration, he must agree the policy about data acquisition of SafeStreets, otherwise he can not go on with the process. Once accepted, he can submit the registration. A confirmation link is sent to his email address. His account will be created only after he confirms the link.

4.2 User has been promoted to Authority

Emma's dream as always been to help the municiplity and to work as a policeman. When she found out about SafeStreets, she registered as a Normal User. One day she finally becomes a policeman in her municiplity. Now she would like to access more data from SafeStreets to be able to do her job at best. So, she asks the municiplity for a PEC email address to allow her account to be promoted as an Authority. She then registers with her PEC email to SafeStreets which, after having recognized the email, promotes her account to Authority.

4.3 User reports a violation

Marco is out for a walk. Unfortunately a car is parked again on the sidewalk, and he is unable to move on without stepping into the cars lane. He is really fed up of those situations, and he would like to help the authorities to solve this problem. He picks up his phone and signs in to SafeStreets. From the app he can access the camera and take a picture of the violation. He then adds the type of violation as 'Parked on Grass/Sidewalk', he confirms the license plate recognized by the application, and sends the report.

4.4 User wants to see the history of is reports

Giorgia is commited to help the authorities with violations. She is therefore an usual user of SafeStreets, and she has a long report history. Giorgia believes someone keeps parking in a red zone nearby here house. She also suspect that it is always the same car, and she also remembers to have seen that car in other red zones in the city. Therefore, she opens the application and checks her report history. She in fact recognizes the car in multiple reports she previously sent.

4.5 Authority wants to have information about a license plate

Gianni works as policeman in the municiplity of Milano. He is not used to file parking violation fines, because he believes that his role is to teach the right behaviour and not to penalize distracted people for a one-time violation. Nonetheless, he cannot excuse people that commit always the same violation. Some collegues advised him to try SafeStreets. At the next violation that Gianni sees, he opens the application and asks for data about the license plate of the car. The system reports multiple violation from this car, so Gianni is happy to leave a ticket.

4.6 Authority is notified about violations nearby his position

The municiplity of Como is promoting the service of SafeStreets to his employees. Mario is intrigued by the possibility that the application can offer. He decides to try the real-time reporting service. He access his account and activates the notifications. On his next turn, he is driving through Via Manzoni. A notification from his phone calls his attention. He parks the car and checks the application. A violation has been report some seconds ago nearby is current position. He goes in the location provided and finds the car parked in a red zone. He files a ticket.

4.7 Authority wants to inspect a list of violations

A new research is being done at Politecnico di Milano about street safety and violations. The researchers needs data to feed to a ML model to advance the research. No dataset is available online, but their searches brings them to SafeStreets. They talk with the municiplity of Milan, which grantes them a permission to study the data collected. So, the municiplity secretary accesses the application as an Authority. She inputs the area requested as 'Piola' and she is allowed to visualize a list of all the violations. She then takes the pictures and the types of violations and she sends them to the researchers.

4.8 Authority validates the reliability of violations

Jenna is a police woman working in the department of Turin. At some point someone uploads on SafeStreets a violation of a license plate that is already associated with nine others violations. The latter reaches ten violations, so the system is asking to an authority a check of reliability. Jenna receives a notification, she has to answer if the ten violations' pictures stored so far are valid or not. She is able to do this because she has an advanced knowledge of the traffic laws, like every other authority.

5 UML modeling

5.1 Use case descriptions

Name	Registration as a Normal User to SafeStreets
Actor	Normal User
Entry condition	The Normal User has opened the mobile application and clicked on the "Create account" button
Event flow	1. The System shows the registration page. 2. The Normal User inputs the following data: email, name, surname, password and confirmation of the password. 3. The Normal User agrees to the privacy policy of SafeStreets. 4. The Normal User confirms and sends the form. 5. The System computes and stores the Normal User information. 6. The System sends a confirmation email to the email provided. 7. The Normal User accesses his/her email account and clicks on the confirmation link. 8. The System activates the account to the User.
Exit condition	A new user account is created. If there already exists an account created with the User email address, the System shows an error message.
Exception	The passwords do not match. In this case, the System displays an error and asks the User to reinput the passwords. The System is unable to process and store the information. In this case an error message is displayed, asking the User to try again. The internet connection is lost. In this case an error message is displayed, asking the User to connect the device in order to be able to send the report.

Name	Violation report sent by User to SafeStreets
Actor	User
Entry condition	The User has opened the mobile application and signed in into his/her account successfully. A internet connection is available.
Event flow	1. The User selectes the camera 2. The System shows the camera screen to the User 3. The User clicks the button or screen to take a picture of the violation 4. The System runs a model to recognize the license plate from the picture 5. The System attaches the license plate code to the report 6. The System attaches the picture to the report 7. The System shows the form to add the type of violation 8. The User selectes the most appropriate type of violation 9. The User confirms the license plate recognized by the System 10. The User clicks the buttons to send the report 11. The System stores the violation information 12. The System shows the home screen of the application
Exit condition	A new violation report is stored successfully and the User is back to the home page of the application. If some problem occurs, the System displays an error message
Exception	The device does not have a working camera. In this case the System warns the User that the violation cannot be reported without a picture. The System is not allowed to access the camera. In this case a permission request is displayed to the User for the System to gain access to the camera. The System is not able to access the camera. In this case a error message is displayed, asking the User to try again later. The User cannot find a correct type of violation. In this case, since no report can be send without this information, the User is required to choose the most suitable type for the situation. The System is unable to process and store the report. In this case an error message is displayed, asking the User to try again. The System is unable to recognize the license plate. In this situation, the System asks the User the option to input the license plate manually or to take another picture. The internet connection is lost. In this case an error message is displayed, asking the User to connect the device in order to be able to send the report.

Name	History report requested by User
Actor	User
Entry condition	The User has opened the mobile application and signed in into his/her account successfully. A internet connection is available.
Event flow	1. The User selectes the history report list 2. The System retrieves all the reports associated with the User 3. The System displays the report list to the User
Exit condition	The screen containing the list of reports of the User has been displayed. If some problem occurs, the System displays an error message
Exception	The System is not able to retrieve the list of reports. In this case, a 'try later' error message is displayed. The User never sent any report. In this case, a message is displayed, telling the User that no report has ever been sent. The internet connection is lost. In this case an error message is displayed, asking the User to connect the device.

Name	Checking license plate violations by an Authority
Actor	Authority
Entry condition	The Authority has opened the mobile application and signed in into his/her account successfully. A internet connection is available.
Event flow	1. The Authority inputs the license plate code into the application 2. The System retrieves all the reports associated with the license plate 3. The System displays the report list to the Authority in order of date starting from the most recent one
Exit condition	The screen containing the list of reports of the license plate has been displayed. If some problem occurs, the System displays an error message
Exception	The license plate is not present in the database, either because the Authority mistyed it or because the license plate is missing or counterfeit. In this case, an error message is displayed, informing the Authority that the license plate is not present into the application. The System is not able to retrieve the list of reports. In this case, a 'try later' error message is displayed. The internet connection is lost. In this case an error message is displayed, asking the User to connect the device.

Name	Real-time notification about nearby violation for an Authority
Actor	Authority
Entry condition	The Authority is registered and signed in into his/her account. The Authority has the notification service activated. A new violation is reported nearby the Authority GPS position. A internet connection is available.
Event flow	1. The System checks the database for real-time reports nearby the GPS position of the device 2. The System finds a recent report and retrieves it 3. The System sends a notification to the Authority device 4. The Authority clicks on the notification message 5. The System displays the report, providing the exact location to the Authority
Exit condition	The System displayed the report information. If some problem occurs, the System displays an error message
Exception	The System does not find any report. In this case, no notification is sent.

Name	List of violations in an area for an Authority
Actor	Authority
Entry condition	The Authority is registered and signed in into his/her account. A internet connection is available.
Event flow	1. The Authority inputs the area of interest 2. The System retrieves the list of violations in that area 3. The System displays the list of violations requested
Exit condition	The System displayed the list of violations. If some problem occurs, the System displays an error message
Exception	The System does not find any report. In this case, a message is displayed, telling the User that no report has ever been sent. The System is not able to retrieve the list of reports. In this case, a ’try later’ error message is displayed. The internet connection is lost. In this case an error message is displayed, asking the User to connect the device.

5.2 Use case diagrams

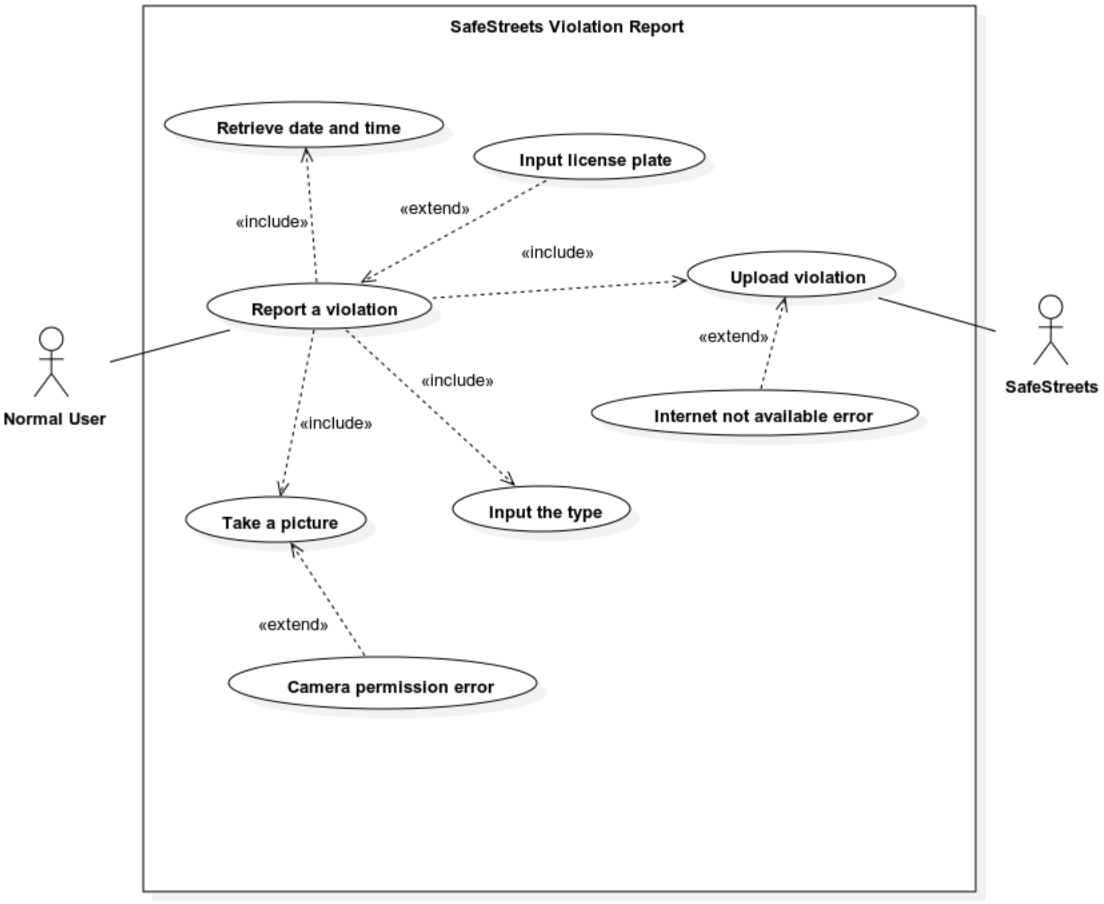


Figure 7: This Use Case diagram shows the main functionality of SafeStreets, particularly the report of a violation by a Normal User.

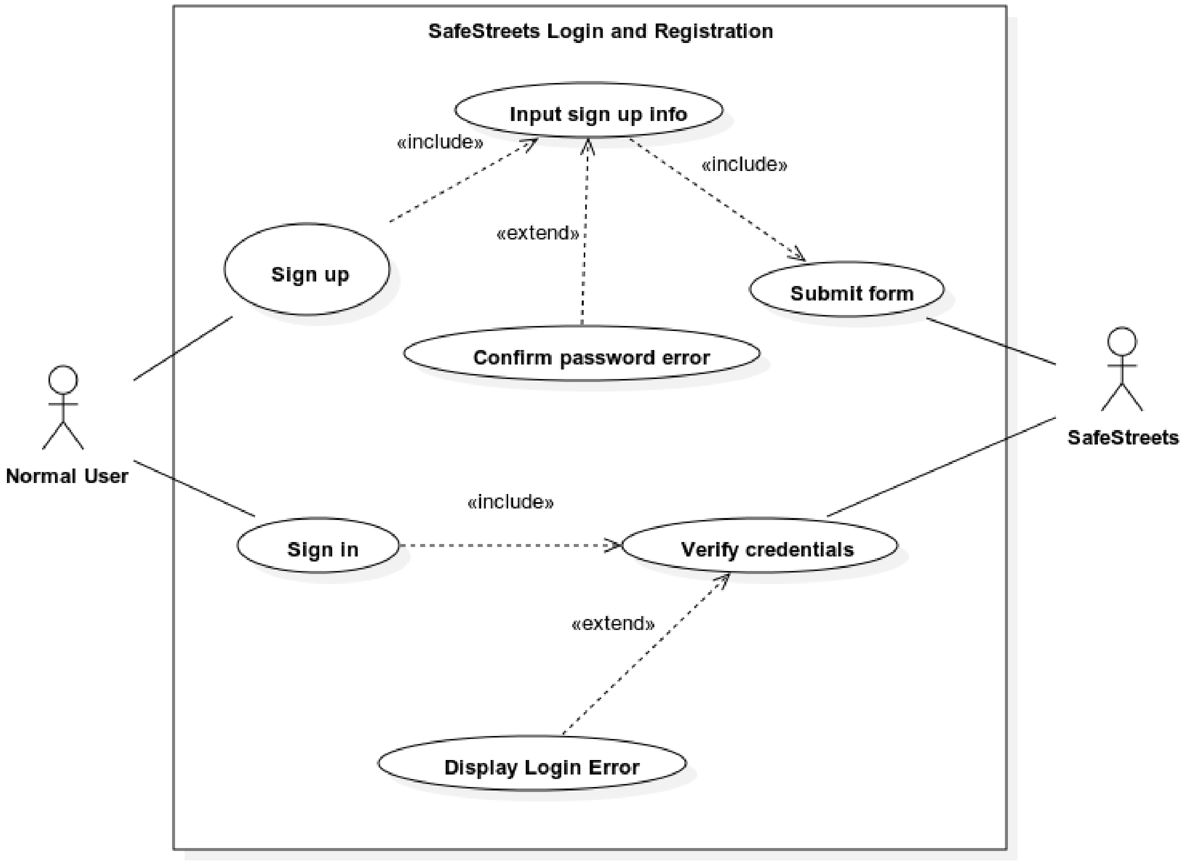


Figure 8: This Use Case diagram shows the login and registration handling.

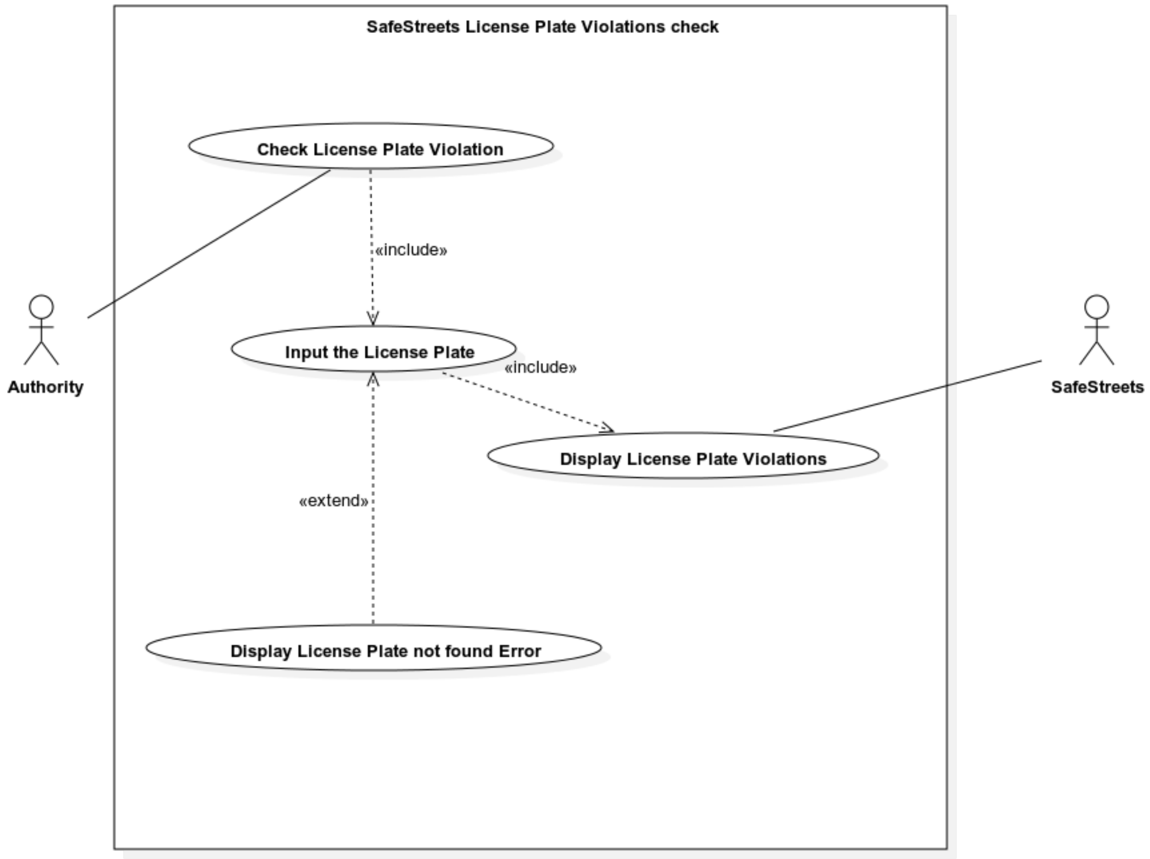
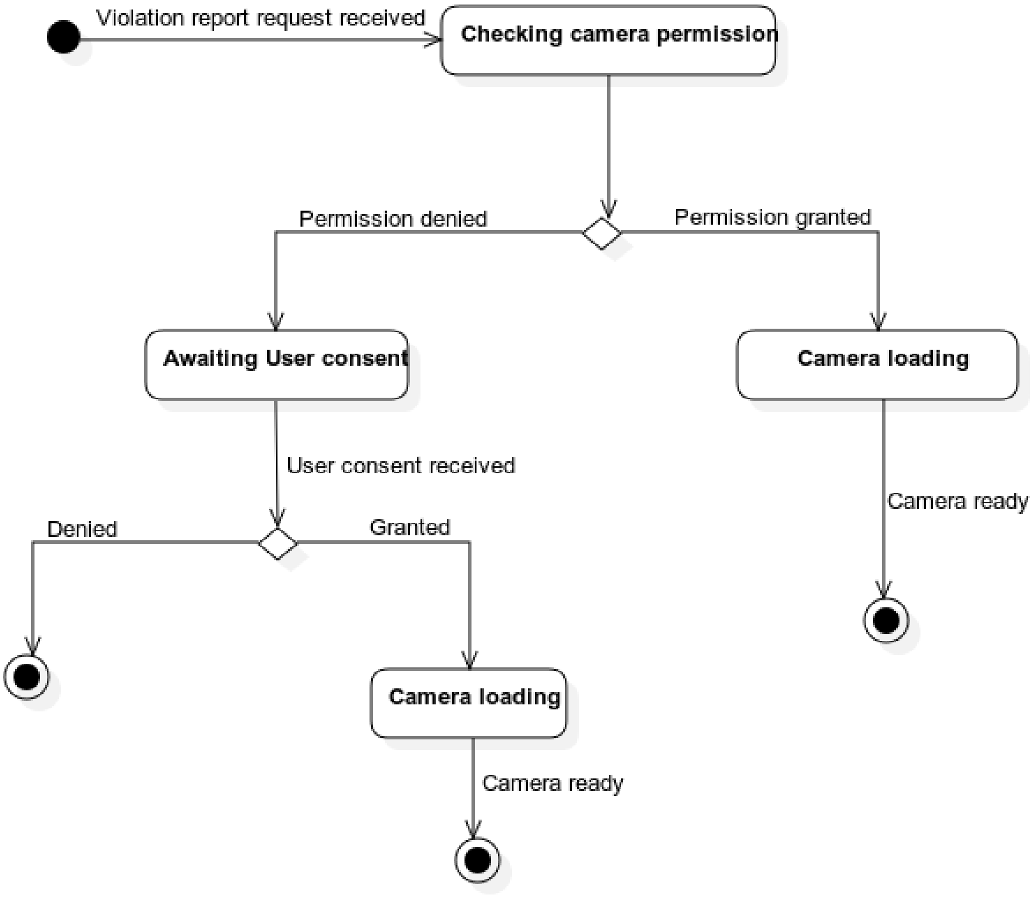

SafeStreets License Plate Violations check
Check License Plate Violation
«include»
Input the License Plate
«include»
Display License Plate Violations
«extend»
Display License Plate not found Error
Authority
SafeStreets

Figure 9: This Use Case diagram shows the license plate check by the Authorities.

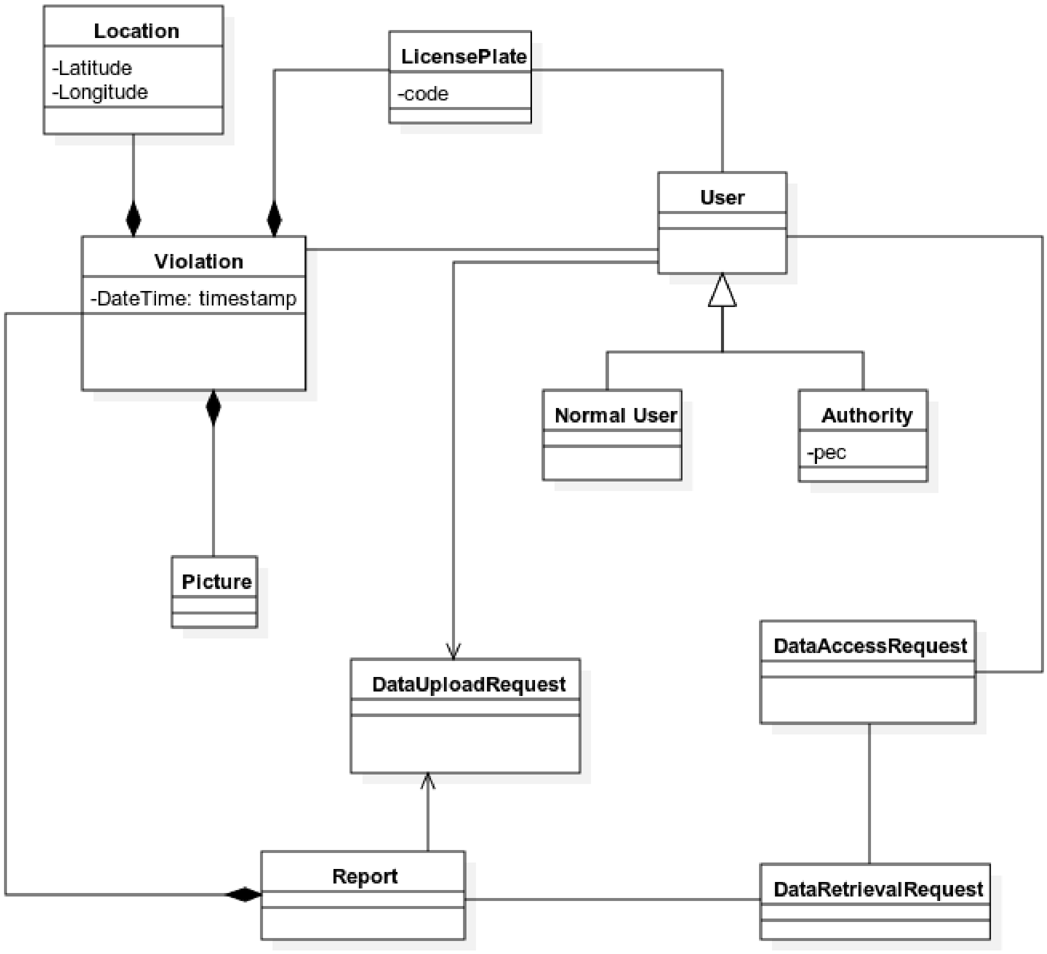
5.3 Statechart Model Diagrams

In this section the Statechart diagram models the authorization process related to the camera of the mobile device. The diagram describes the states the system pass throught when a User try to access the camera reporting a violation.



5.4 Class diagrams

In this section we provide a class diagram of the main part of the system. The purpose of the diagram is not to provide an implementation or design aspect of the system to be, therefore there are not completely modelled attributes, visibility of attributes, types and naming conventions. This kind of detail will be pursued during the design phase. Read the diagram with this assuntion in mind.



5.5 Activity and Sequence diagrams

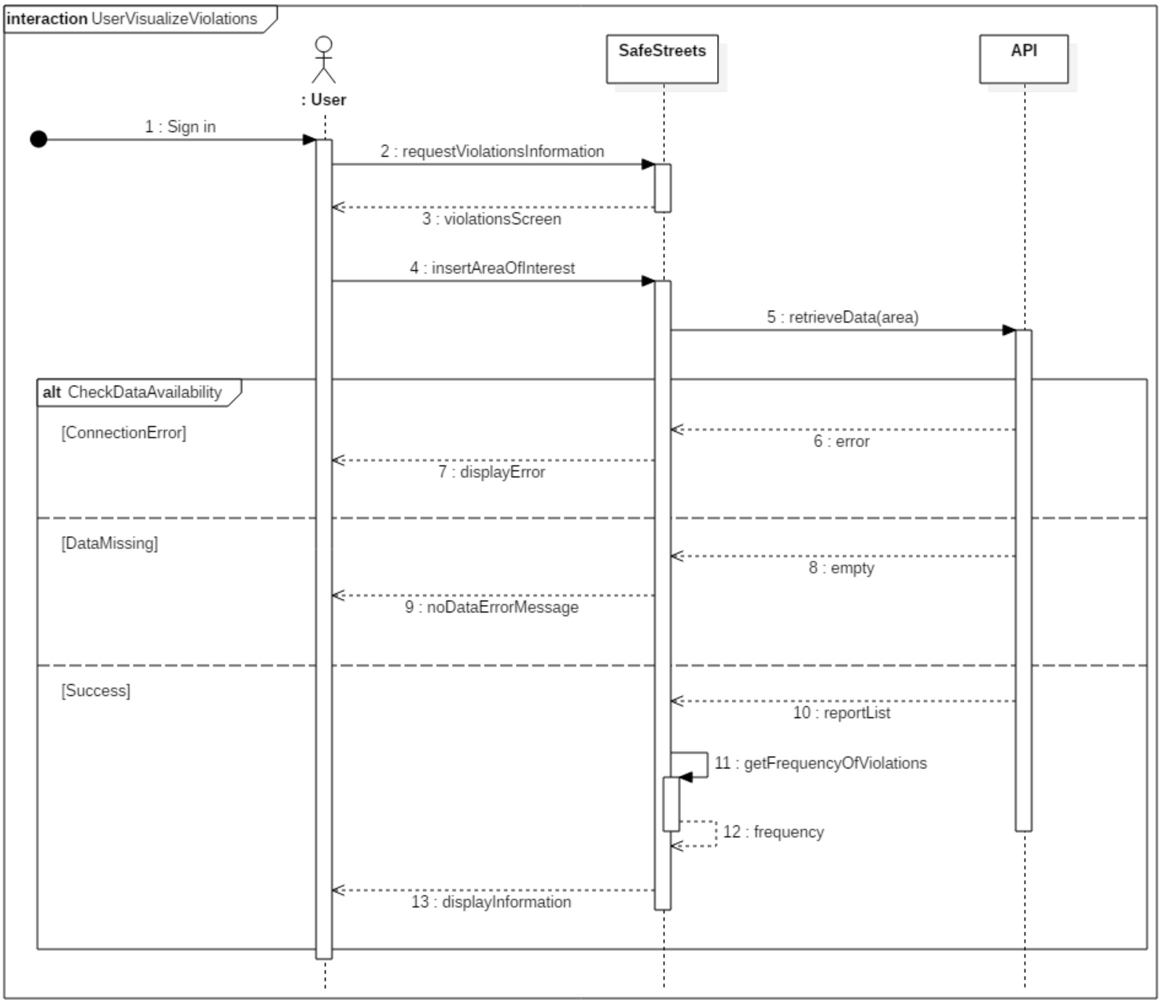


Figure 10: This sequence diagram shows how the User can request and visualize the frequency of violations in a specific area. The User inputs the area of interest and the System connects to the API to retrieve the data.

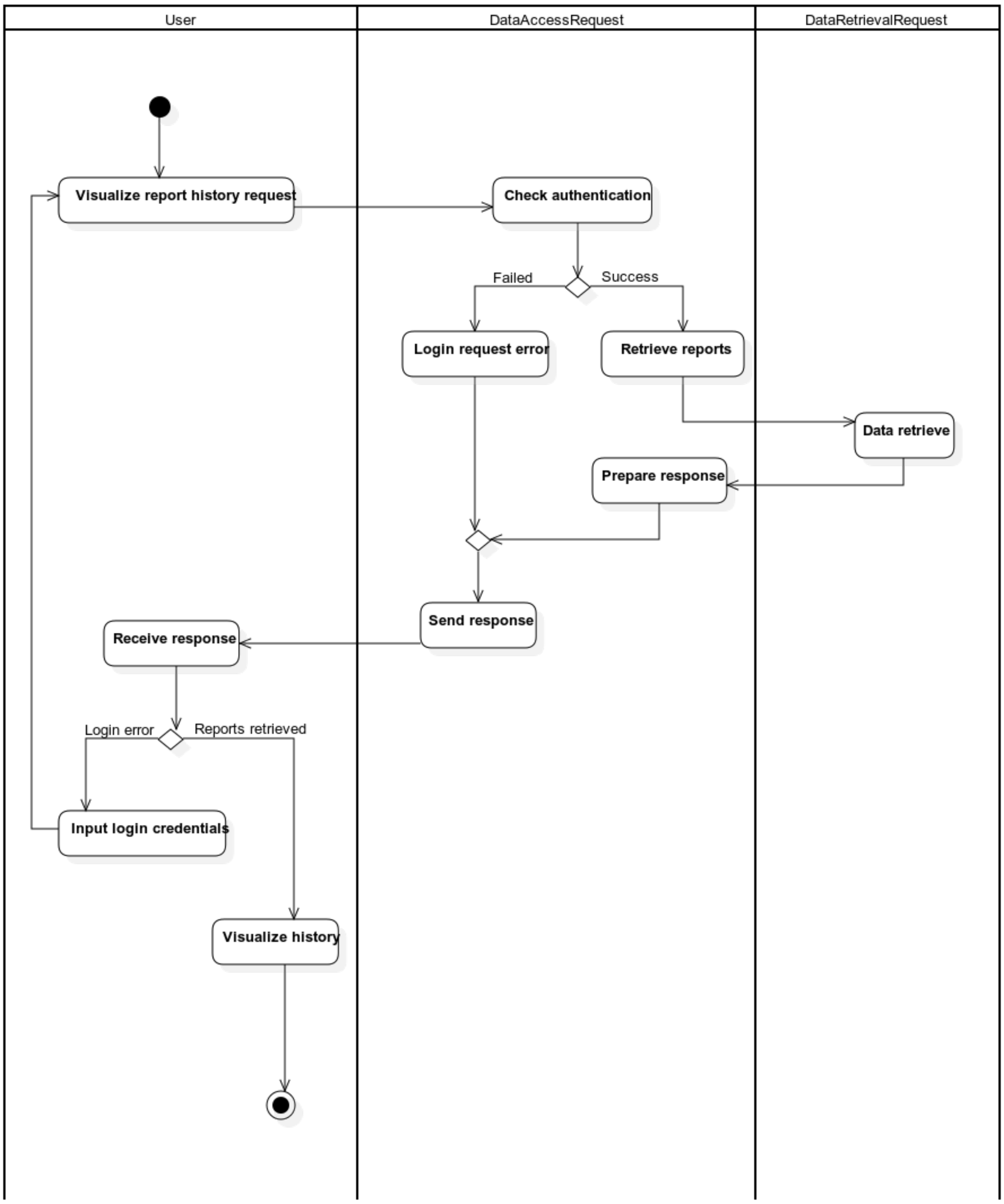


Figure 11: This activity diagram shows the actions performed in order to get the history of the reports sent by a User. It takes in consideration also the authentication process, particularly DataAccessRequest checks if the User authentication is valid, if so DataRetrievalRequest get all the reports and returns them.

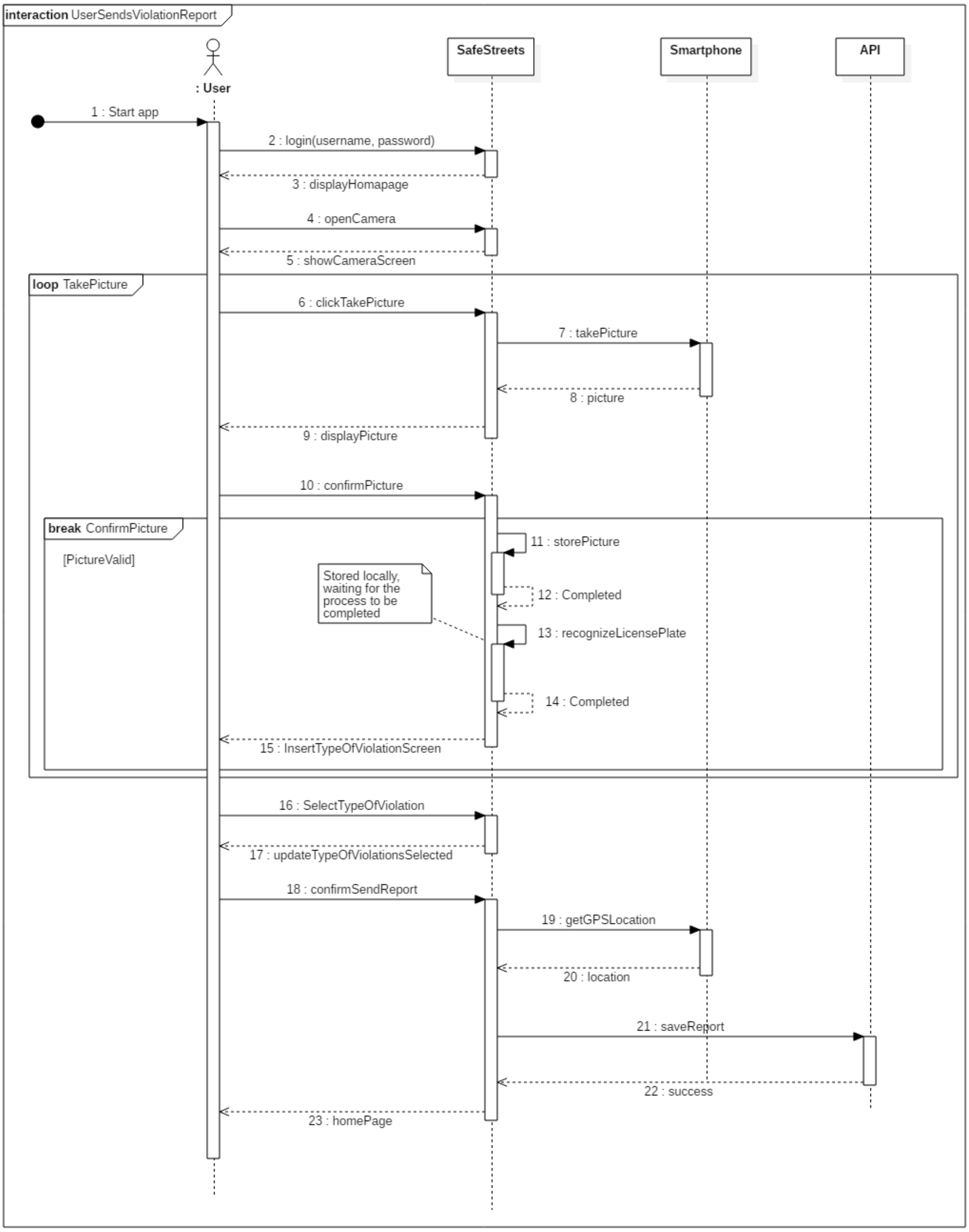


Figure 12: This sequence diagram shows the interaction between User and System in the process of a violation report. After the User opens the app, he/she requests the camera and takes the picture until he/she is satisfied of the result. Then the User adds the type of violation and confirms. The System saves the report and displays the homepage.

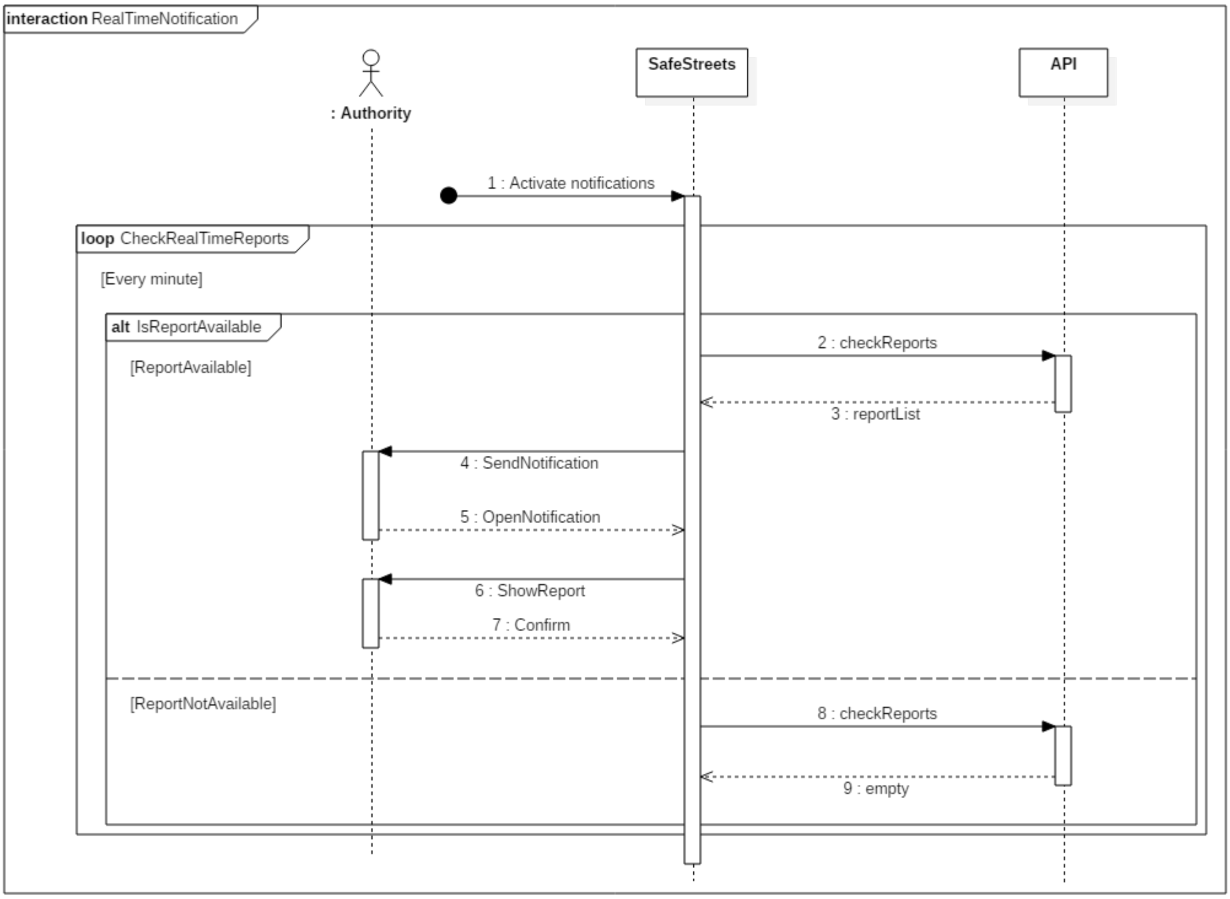


Figure 13: This sequence diagram shows the real-time notification service. After the Authority activates the notification service, the application start to listen for real-time reports. If the System finds reports, it sends a notification to the Authority.

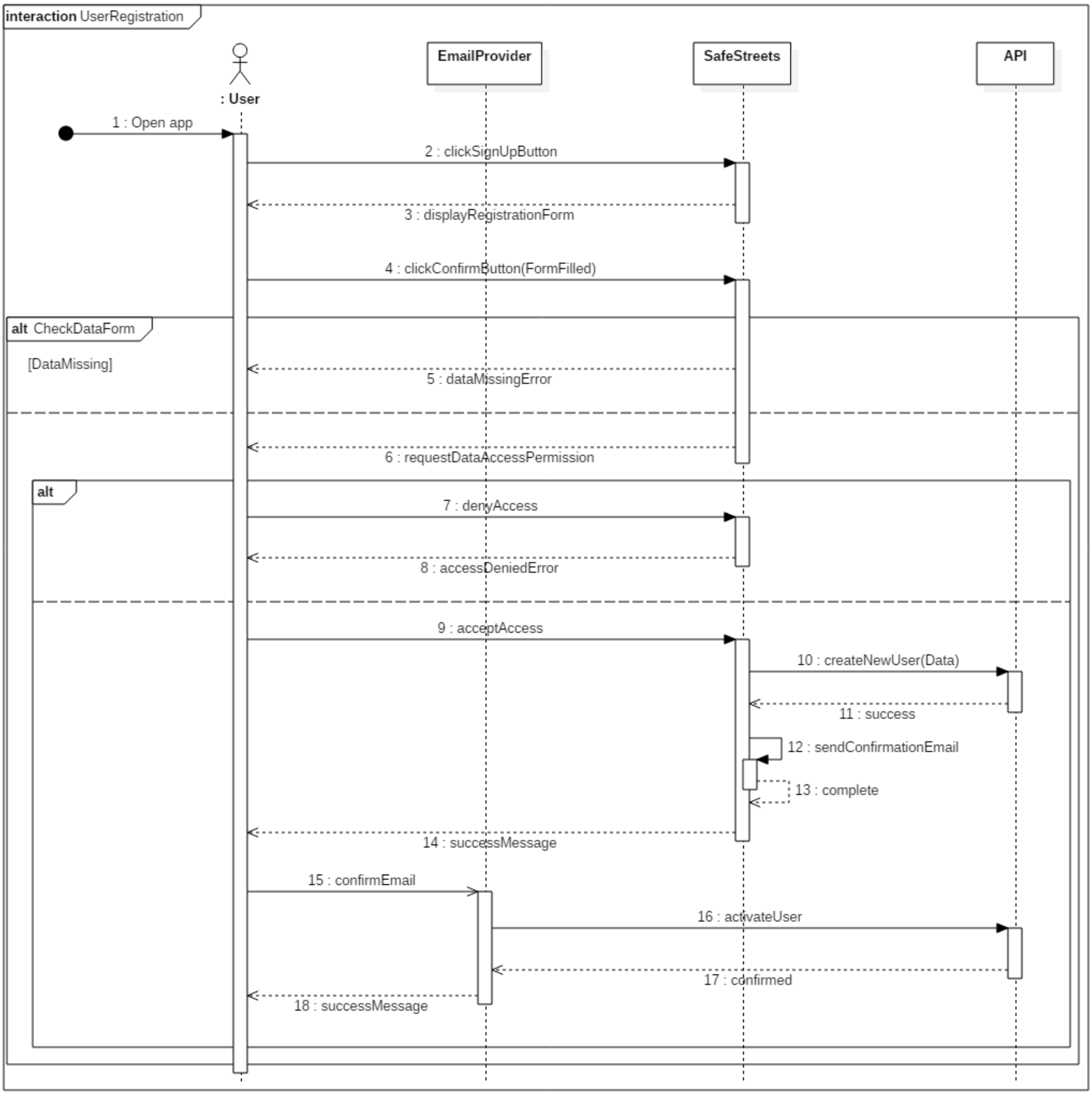


Figure 14: This sequence diagram shows the steps required for a New User to register to SafeStreets. The User must fill the form, accept the usage permissions, and confirm his/her identity via email.

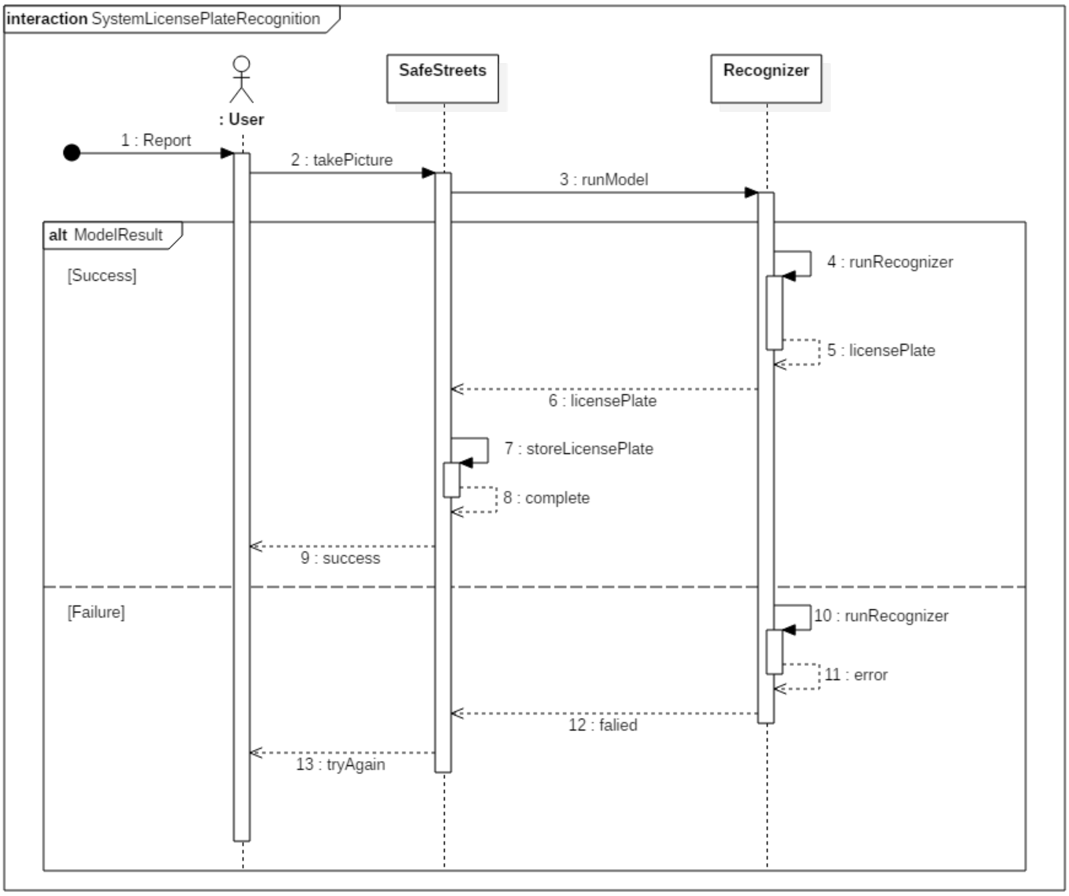
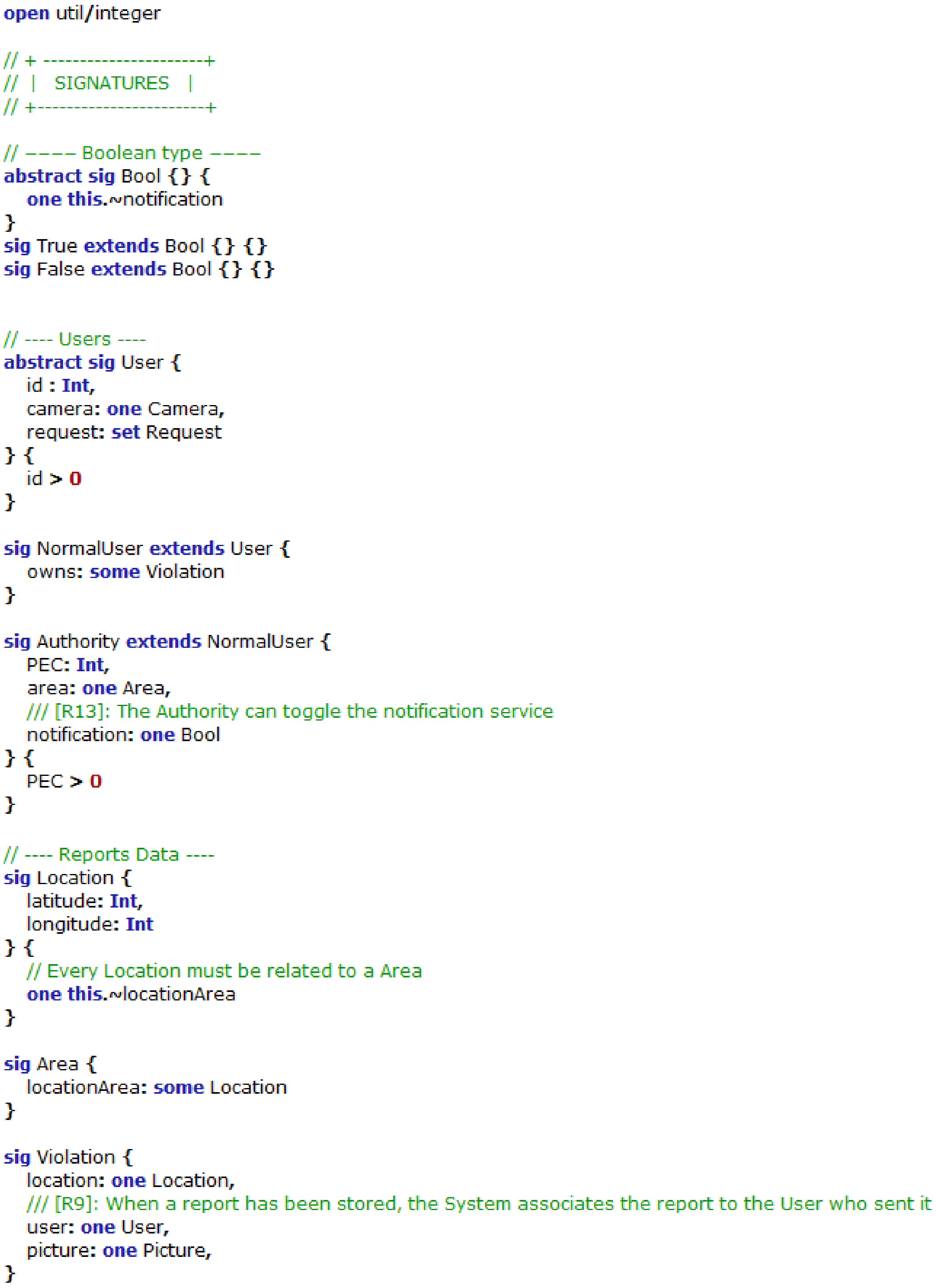


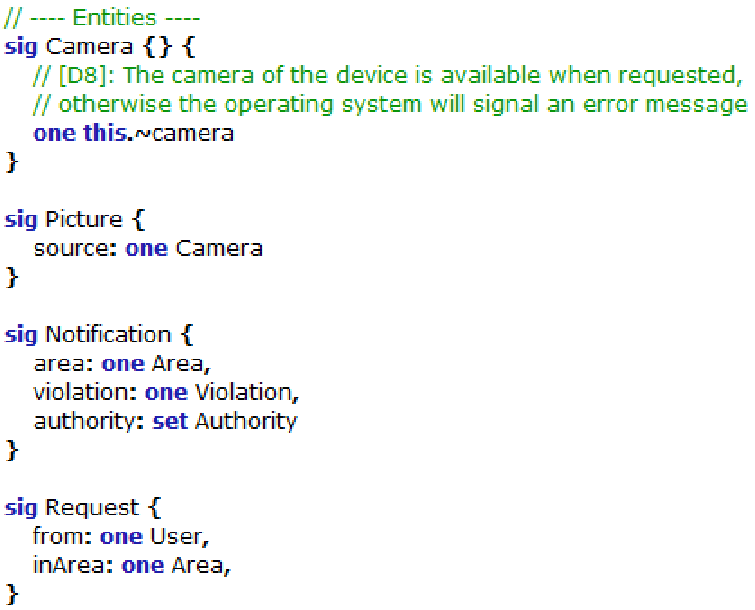
Figure 15: This sequence diagram shows the license plate recognition process. When reporting a violation, a User takes a picture of the violation and the System tries to recognize the license plate. In case of failure, an error is reported.

6 Formal Analysis Using Alloy

6.1 Signatures

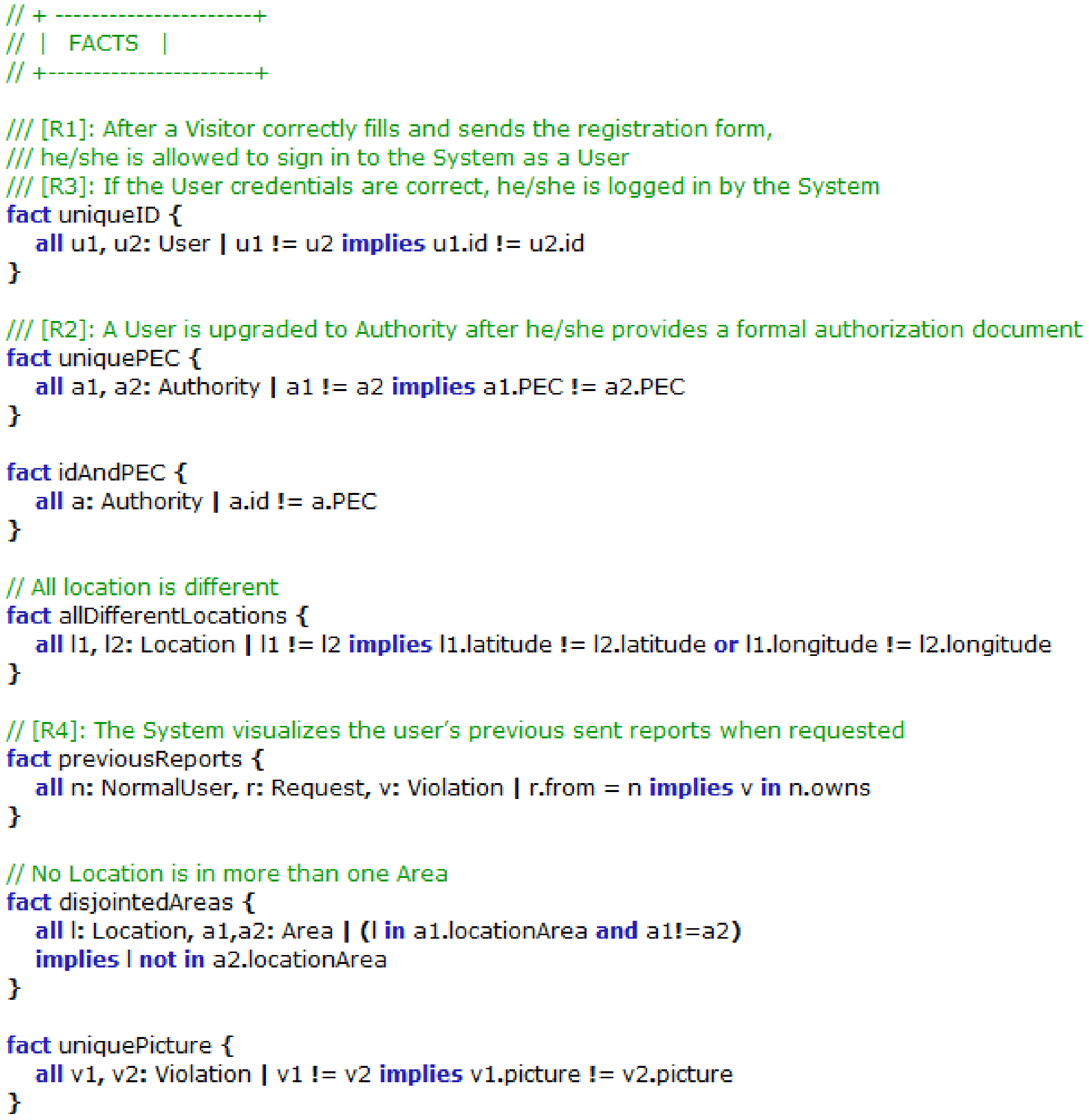


```
open util/integer

// + ----------------------+
// |   SIGNATURES   |
// +------------------------+

// ---- Boolean type ----
abstract sig Bool {} {
    one this.~notification
}
sig True extends Bool {} {}
sig False extends Bool {} {}


// ---- Users ----
abstract sig User {
    id : Int,
    camera: one Camera,
    request: set Request
} {
    id > 0
}

sig NormalUser extends User {
    owns: some Violation
}

sig Authority extends NormalUser {
    PEC: Int,
    area: one Area,
    /// [R13]: The Authority can toggle the notification service
    notification: one Bool
} {
    PEC > 0
}

// ---- Reports Data ----
sig Location {
    latitude: Int,
    longitude: Int
} {
    // Every Location must be related to a Area
    one this.~locationArea
}

sig Area {
    locationArea: some Location
}

sig Violation {
    location: one Location,
    /// [R9]: When a report has been stored, the System associates the report to the User who sent it
    user: one User,
    picture: one Picture,
}
```



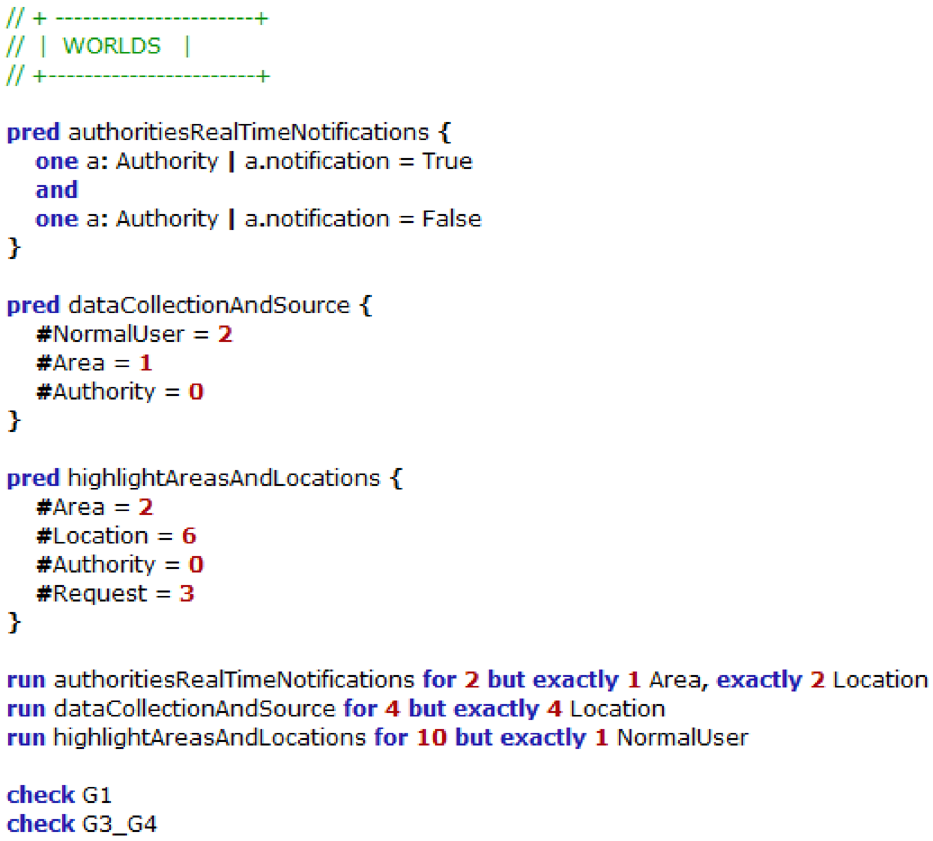
```
// ---- Entities ----
sig Camera {} {
    // [D8]: The camera of the device is available when requested,
    // otherwise the operating system will signal an error message
    one this.~camera
}

sig Picture {
    source: one Camera
}

sig Notification {
    area: one Area,
    violation: one Violation,
    authority: set Authority
}

sig Request {
    from: one User,
    inArea: one Area,
}
```

6.2 Model



```
// + ----------------------+
// |   FACTS   |
// +-----------------------+

/// [R1]: After a Visitor correctly fills and sends the registration form,
/// he/she is allowed to sign in to the System as a User
/// [R3]: If the User credentials are correct, he/she is logged in by the System
fact uniqueID {
    all u1, u2: User | u1 != u2 implies u1.id != u2.id
}

/// [R2]: A User is upgraded to Authority after he/she provides a formal authorization document
fact uniquePEC {
    all a1, a2: Authority | a1 != a2 implies a1.PEC != a2.PEC
}

fact idAndPEC {
    all a: Authority | a.id != a.PEC
}

// All location is different
fact allDifferentLocations {
    all l1, l2: Location | l1 != l2 implies l1.latitude != l2.latitude or l1.longitude != l2.longitude
}

// [R4]: The System visualizes the user's previous sent reports when requested
fact previousReports {
    all n: NormalUser, r: Request, v: Violation | r.from = n implies v in n.owns
}

// No Location is in more than one Area
fact disjointedAreas {
    all l: Location, a1,a2: Area | (l in a1.locationArea and a1!=a2)
    implies l not in a2.locationArea
}

fact uniquePicture {
    all v1, v2: Violation | v1 != v2 implies v1.picture != v2.picture
}
```

```
/// [R5]: A User can access the camera to take a picture of the violation
fact disjointedCameras {
    all c: Camera, u1,u2: User | (c in u1.camera and u1!=u2)
    implies c not in u2.camera
}

// The Picture of the Violation has been taken from the Camera of the User who reported it
fact pictureViolationUserTakenFromOwnCamera {
    all p: Picture, u: User, v: Violation | p.source = u.camera and v.user = u implies v.picture = p
}

// User owns all own violations
fact violationOwner {
    all u : NormalUser, v: Violation | v in u.owns implies v.user = u
}

/// [R14]: The System periodically checks the GPS position of an Authority
/// if he/she has the notification service enabled
/// [R15]: If a violation is detect nearby the Authority position and
/// he/she has the notification service activated, the System sends a notification
/// [R16]: An Authority is periodically updated about violation in an area
/// if he/she has activated the notification service in that area
fact notificationRetrieve {
    all a: Authority, n: Notification |
    (a.area = n.area and a.notification = True and a != n.violation.user)
    iff a in n.authority
}


// + ---------------------+
// |   GOALS   |
// +----------------------+

/// [G1] Allow user to report a violation
assert G1 {
    all u: User, c: Camera |
    u.camera = c implies #u.owns > 0
}

/// [G3] Allow user to visualize information about violations in a specific area
/// [G4] Allow user to visualize all report that he/she sent to the system
assert G3_G4 {
    all u: NormalUser, v: Violation, r: Request, a: Area |
    (r.from = u and r.inArea = a and v.location in a.locationArea) implies v in u.owns
}
```

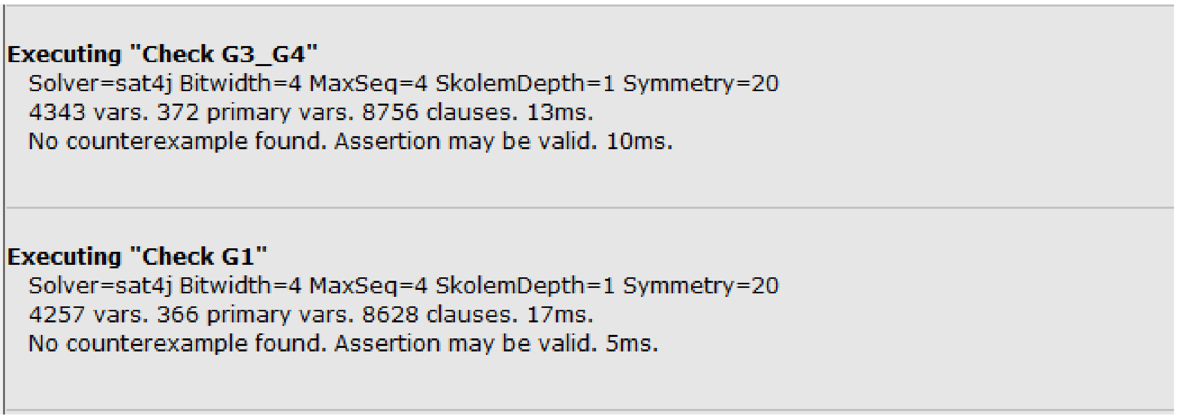


```
// + ----------------------+
// |  WORLDS   |
// +-----------------------+

pred authoritiesRealTimeNotifications {
    one a: Authority | a.notification = True
    and
    one a: Authority | a.notification = False
}

pred dataCollectionAndSource {
    #NormalUser = 2
    #Area = 1
    #Authority = 0
}

pred highlightAreasAndLocations {
    #Area = 2
    #Location = 6
    #Authority = 0
    #Request = 3
}

run authoritiesRealTimeNotifications for 2 but exactly 1 Area, exactly 2 Location
run dataCollectionAndSource for 4 but exactly 4 Location
run highlightAreasAndLocations for 10 but exactly 1 NormalUser

check G1
check G3_G4
```

6.3 Results

The Alloy code models the System, giving specific emphasis to the relationships between NormalUser, Auhtority, Violation, and Request. Every User has a unique id, and he/she has a Camera available to take Picture(s). Those Picture are attached to a Violation. Every Violation also has a Location (from the GPS) and it is associated with the User who took it. Each User can fill a Request with different levels of granularity based on his/her role as NormalUser or Auhtority. Each Request is made by a User for a specific Area. Each Area is composed of a series of Location. Every Location is unique for each Area, in such a way that no Area is intersected with any other. A Notification is sent to every Auhtority which has activated the real-time service (through the True signature) and who's Location is in the same Area as the Location of the Violation.

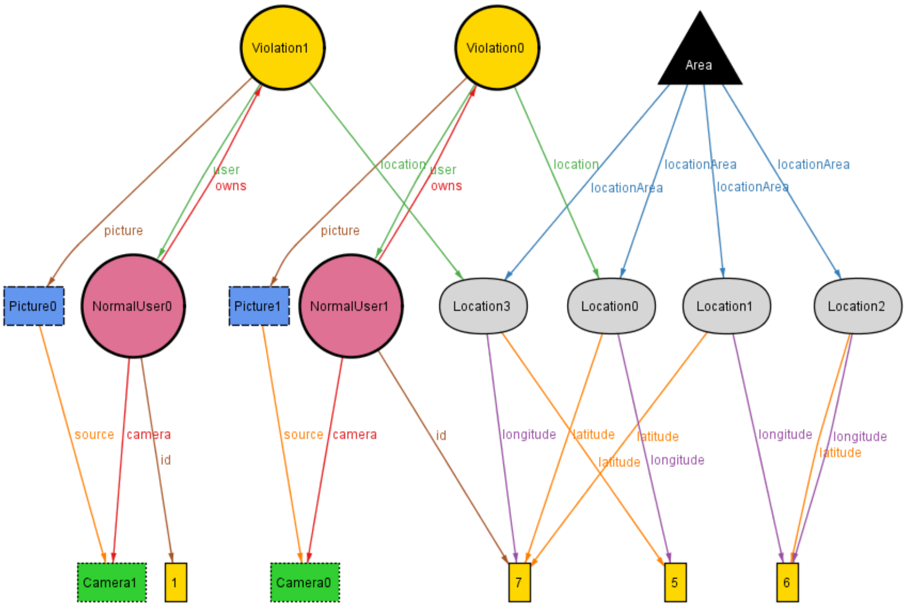
6.3.1 Proof of consistency



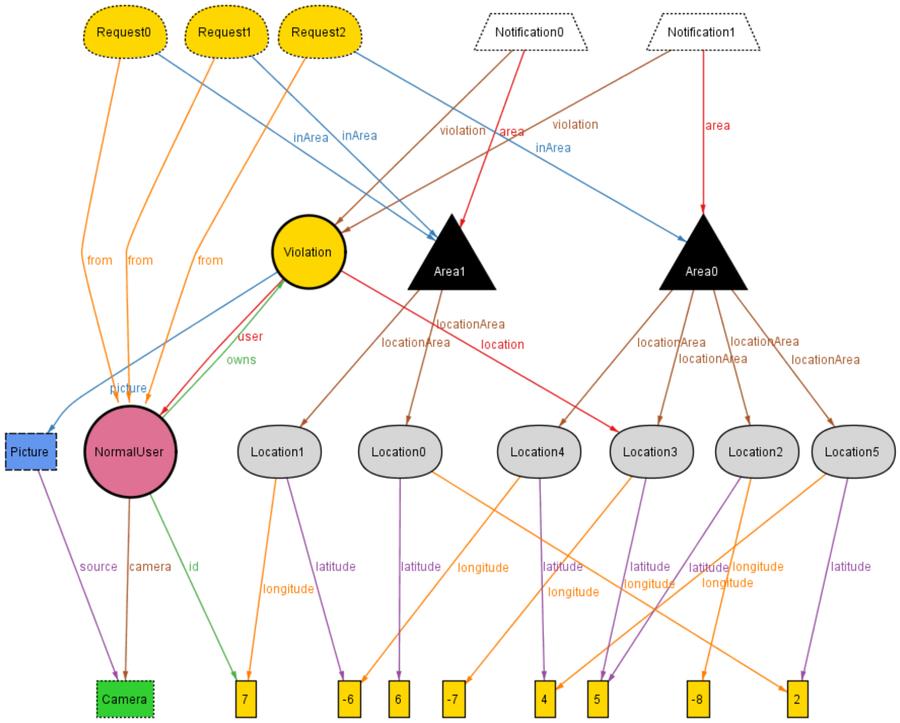
```
Executing "Check G3_G4"
   Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
   4343 vars. 372 primary vars. 8756 clauses. 13ms.
   No counterexample found. Assertion may be valid. 10ms.

Executing "Check G1"
   Solver=sat4j Bitwidth=4 MaxSeq=4 SkolemDepth=1 Symmetry=20
   4257 vars. 366 primary vars. 8628 clauses. 17ms.
   No counterexample found. Assertion may be valid. 5ms.
```

We highlighted for each component of the model how it is related to requirements, assumptions, and goals. We checked the consistency of G1 and G3, G4 (see section 1.2.4).

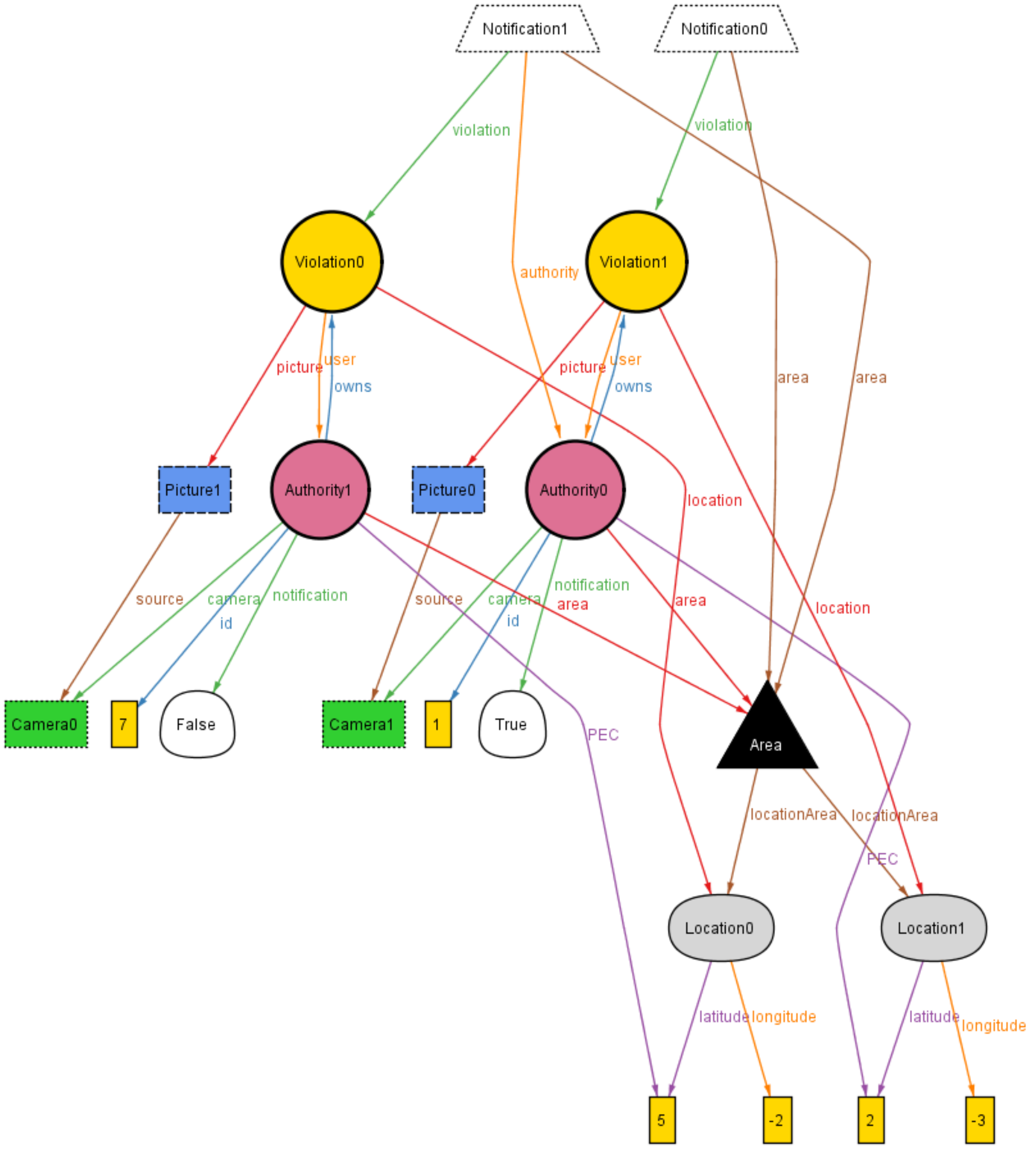
6.3.2 Generated worlds



In this generated world we can clearly see how a NormalUser interacts with the System. Each NormalUser owns a Violation, which has a Picture which was taken by the User's Camera. In this example there exists only one Area with encompasses all the Location available. Each Violation is connected to the Location from where it has been reported. Each NormalUser has an unique id and every Location is made of two coordinates (latitude, longitude).



In this generated world we highlight how a single NormalUser interacts and sends Request to the System. The world has two Area, which are made of different Location. The NormalUser sent a Violation report from his/her Camera. The NormalUser also sent three Request, two for Area1 and one for Area0. Two Notification would be available, but a NormalUser has no access to the notification service, which is reserved for Auhtorities.



In this generated world we focus on the interaction of the Auhtority with the System, as well as the real-time Notification service. Two Auhtorities are present in the world. Every Auhtority is also allowed to send Violation, and in this example each of the two Auhtority sent a Violation. They therefore have a Camera with which they have taken a Picture. Auhtority0 has activated the notification service as we can see from its connection with the entity True. Two Violation have been reported in the same Area in which the Auhtority is currently situated (Area0). However, one of those has been reported by the Auhtority0 itself. Therefore, the Auhtority is notified only from Notification1, which was sent by Auhtority1, that a Violation has been reported nearby his/her Area. Every Auhtority also has a PEC identification number.

7 Appendix

7.1 Used tools

- Visual Studio Code: Editing LaTeX files, pdf generation, and Github repository management - https://code.visualstudio.com/

- Alloy 4.2: IDE tool for modeling requirements - https://alloytools.org/

- LaTeX: For the production of technical and scientific documentation - https://www.latex-project.org/

7.2 Hours of work

7.2.1 Cappelletti Andrea

Date	Task	Hours	Start-End time
16/10/2019	Review assignment document	1h 0min	11:30 - 12:30
16/10/2019	Asking questions and clarifications	30min	14:40 - 15:00
19/10/2019	Setup project structure and environment	2h 30min	15:00 - 17:30
21/10/2019	Learning LaTeX syntax and structure	1h 15min	11:00 - 12:15
22/10/2019	Definitions, Abbreviations and References	45min	11:00 - 11:45
23/10/2019	Product perspective	30min	11:00 - 11:30
23/10/2019	Product perspective and assumptions	50min	14:00 - 14:50
24/10/2019	RASD analysis review	1h 15min	18:30 - 19:45
25/10/2019	Scenarios and iOS mock-ups	2h	12:00 - 14:00
26/10/2019	UML diagrams	2h	15:30 - 17:30
29/10/2019	Mock-ups and UML diagrams	1h 20min	12:15 - 13:35
30/10/2019	General review and UML diagrams	2h 00min	12:50 - 14:50
31/10/2019	UML and mock-ups	3h 00min	9:00 - 12:00
01/11/2019	Review and mock-ups upload	35min	17:20 - 17:55
04/11/2019	Alloy	1h 30min	12:00 - 13:30
05/11/2019	Alloy	2h 30min	11:30 - 14:00
06/11/2019	Alloy	2h 30min	10:30 - 13:00
08/11/2019	Activity diagram and review	2h	15:30 - 17:30
09/11/2019	Complete document review	2h 30min	14:45 - 17:15
Total		**30h 30min**	16/10 to 10/11

7.2.2 Maglione Sandro

Date	**Task**	**Hours**	**Start-End time**
16/10/2019	Review assignment document	1h 0min	11:30 - 12:30
16/10/2019	Asking questions and clarifications	30min	14:40 - 15:00
19/10/2019	Setup project structure and environment	2h 30min	15:00 - 17:30
20/10/2019	Document and files initial LaTeX structure	2h 10min	15:15 - 17:25
21/10/2019	Introduction: purpose and scope	1h 0min	11:20 - 12:20
22/10/2019	Introduction: goals	30min	5:15 - 5:45
22/10/2019	Introduction and Product functions	45min	11:45 - 12:30
23/10/2019	Product functions, User characteristics	1h	11:50 - 12:50
23/10/2019	Constraints, non-functional requirements	2h	14:45 - 16:45
24/10/2019	Performance, attributes, constraints	1h	6:30 - 7:30
25/10/2019	Review after RASD analysis in class	50min	6:30 - 7:20
26/10/2019	Setup Dia UML, requirements	1h	14:30 - 15:30
26/10/2019	Requirements, goals, assumptions	2h	15:30 - 17:30
27/10/2019	Review requirements/assumptions	2h	15:30 - 17:30
29/10/2019	Scenarios and use cases	1h 45min	10:30 - 12:15
30/10/2019	Scenarios, use cases, sequence diagrams	1h 45min	15:15 - 17:00
31/10/2019	Sequence diagrams	1h	6:15 - 7:15
01/11/2019	General review, sequence diagram	1h	14:45 - 15:45
04/11/2019	Alloy	1h 30min	12:00 - 13:30
05/11/2019	Alloy	2h 30min	11:30 - 14:00
06/11/2019	Alloy	2h 30min	10:30 - 13:00
07/11/2019	Alloy	1h 30min	6:30 - 8:00
08/11/2019	Alloy and review	1h	8:30 - 9:30
09/11/2019	Complete document review	2h 30min	14:45 - 17:15
Total		**35h 15min**	16/10 to 10/11

References

The following resources were read before the writing of the RASD document.

- Specification document "SafeStreets Mandatory Project Assignment"
- Slides - "5.b Structure of RASD"
- http://homepage.cs.uiowa.edu/ tinelli/classes/181/Fall17/Notes/alloy-cheatsheet.pdf