Requirement Analysis and Specification Document

POLITECNICO
MILANO 1863

Dec 2019

Arash Bahariye 10596377
Niloofar Salimi 10648072

Version 1.1

Deliverable: RASD
Title: Requirement Analysis and Verification Document
Authors: Arash Bahariye, Niloofar Salimi
Version: 1.1
Date: 15-Dec-2019
Download page:
https://github.com/arashbahariye/BahariyeSalimi

Table of Contents

1.Introduction

1.1Purpose

This document is aimed to provide documentation for the process of implementing a system for both common end-user and a third-party authority, in our case municipality of Milano, by which end users will provide the system with violation reports and authorities exploit these reports to extract useful trends out of them. Next step is to get back these trends from municipality and provide part of them as statistics to be shown to common end-user. This app will give every citizen of the Milano an opportunity to become a virtual traffic agent. Also, SafeStreets is expected to be an answer for various limitations faced by the authorities in controlling traffic violations.
The SafeStreets platform provides services in which common end-users can report traffic violations and send the data from their mobile phones and also provides web-based platform for manucipality so that they can exploit reported violations to extract useful trends. This document will try to find the requirements for setting up such a system.

1.2 Scope

1.2.1 Description of the Given Problem

SafeStreets is an application that intends to bring public participation into reducing traffic offences, in particular parking violations. The application allows users to send pictures of violations, including violation's type, location, picture and plate number of violating vehicles
To the system. There are 2 possible services, basic service and advanced service. for the basic service SafeStreets, firstly validate quality of the reported violation in concerns of quality of the pictures taken, filled mandatory fields and so on so that if the report doesn't meet any of these metrics, it will be rejected. Secondly, if reports pass the qualification validation, the system stores the information provided by common end-users into its data bases. In addition, the application allows both end users and authorities to mine the information that has been received as its advanced functionality. For example, by highlighting the streets (or the areas) with the highest frequency of violations, or vehicles that commit the most violations. Of course, different levels of visibility could be offered to different users (by providing paid services and free services). In addition, the municipality (and, in particular, the municipal agent) could offer a service that takes the information about the violations coming from SafeStreets and generates traffic tickets from it. In this case, mechanism should be put in place to ensure that the chain of custody of the information coming from the users is never change, and the information is never altered (e.g., if a manipulation occurs at any point of the image showing the violation, for example to alter the license plate, the application should discard the information). In addition, the information about issued tickets can be used by SafeStreets to build statistics.

1.2.2 Similar Systems

Already there are two applications whose names are TrafficEye and PublicEye that people can report road crashes & traffic violations without disclosing the sender identity.

1.3 World and machine phenomena:

1.3.1 World Phenomena

- The occurrences of Violation
- The report of Violation by common end-user
- Taking photos along with details of Violation

- Upload the photo

1.3.2 Machine phenomena

- Updating the database from common user's reports entry
- creating a new object of class of traffic Violation
- extracting trends from mining data provided by users

1.3.3 Shared phenomena

- Validating common users reports
- Showing statistics

1.4 Goals

[G1] Allow a Visitor to become registered User after providing personal information in sign up form.
[G2] Allow users to report violations.
[G2.1] Getting Access to user's camera on his/her smart phone.
[G2.2] retrieving images must be clear and without noises.
[G2.3] Providing a form for additional information, such as the license plate number, type of violation, details about location and vehicle's position.
[G2.4] information of violation which is sent by users should be represented in a table and all mandatory field must be filled.
[G2.5] Getting access to user's location.
[G2.6] validating user's location. For example, it cannot be on the sea!
[G3] Allow users to get useful information through statistics. For example, highlighting streets with the highest number of parking violations, hence expecting not to find empty parking slot to park the car.
[G4] Give the municipality a panel to have access to the violations and be able to exploit them.
[G5] Retrieving extracted trends and issued tickets from municipality.
[G6] The system must provide a mechanism to reject violation reports if users have altered the plate number.

1.5 Definitions, acronyms and abbreviations

1.5.1 Acronyms

- RASD: Requirement Analysis and Specification Document.
- API: Application Programming Interface

1.5.2 Abbreviations

- [Dn]: n-domain assumption.
- [Gn]: n-goal.
- [Rn]: n-functional requirement.

1.5.3 Definitions

- User: an entity who uses of the application. when it requests to sing up to the application, it should insert the following information: name, surname, telephone number, email address, username and password.
- Traffic Violation: Traffic violation occur when driver violates laws that regulate vehicle operation on streets and in the scope of this application it concerns with parking violations.
- Type of violation: it can be 5 type of parking violation:
    - Double parking: is a traffic offence categorized under parking violations. It occurs when a car parks parallel to another vehicle parked against a curb.
    - Handicap parking: is a traffic offence categorized under parking violations. It occurs when a car parks on that spots are set aside for people with disability.
    - Residential parking: is a traffic offence categorized under parking violations. It occurs when a car parks where parking zones are controlled or designated for use by residents living nearby.
    - Sidewalk parking: is a traffic offence categorized under parking violations. It occurs when a car parks in sidewalks are meant for use by pedestrians.
    - Special parking: is a traffic offence categorized under parking violations. It occurs when a car parks a place that are designated for specific vehicles.
- System: an entity we will create consists of servers and databases.
- API: application programming interface; it is a common way to communicate with other systems.

1.6 Revision History

- Version 1.0: First Release
- Version 1.1:
    - Nov 19, 2019: Proposed goals updated.
    - Nov 28, 2019: Goals, Template for use case table, use case table updated.
    - Dec 1, 2018: States Diagram updated.
    - Dec 5, 2019: Use Case Diagrams, Scenarios, Class diagram updated.
    - Dec 6, 2019: Sequence Diagrams, Requirements Mapping updated.
    - Dec 7, 2019: Diagrams updated.
    - Dec 10, 2019: Scenarios updated.
    - Dec 12, 2019: UI updated and small fixes.
    - Dec 15, 2019: Effort spent added, more user interfaces added, last fixes.

1.7 References

- Specification Document: “Assignments AA 2018-2019.pdf”.
- GPS Performances
- IEEE Std 830-1993 - IEEE Guide to Software Requirements Specifications.

1.8 Document Structure

This RASD is composed by six parts and an appendix:

1. In the first part an introduction to the problem is given listing all the identified goals and providing some base information in order to better understand the other sections of the document.
2. The second part consists of an overall description of the system in which its boundaries are identified, and the actors involved in the system's usage lifecycle are listed. The boundaries are given providing all the necessary assumptions: both the ones required in order to better understand the customer's specifications given and the ones that will hold into the system and now on considered as true.
3. The third part is composed by the specific requirements identified, both functional and non-functional.
4. In the fourth part a list of eight scenarios is provided; each of them describes a particular situation with the system might have to cope with.
5. The fifth part is entirely composed by the UML diagrams that model the system in details.
6. Sixth part is embodied with the Alloy model of the system and includes all the relevant details; a proof of consistency and an example of the generated world are also provided.
7. The last part is accessory and contains a list of the tools used to redact this document and its contents, and a detailed report of the hours spent to do so.

2 Overall Description

2.1 product perspective

The domain model used to describe Safestreets system follows the class diagram present in the figure 1 bellow.

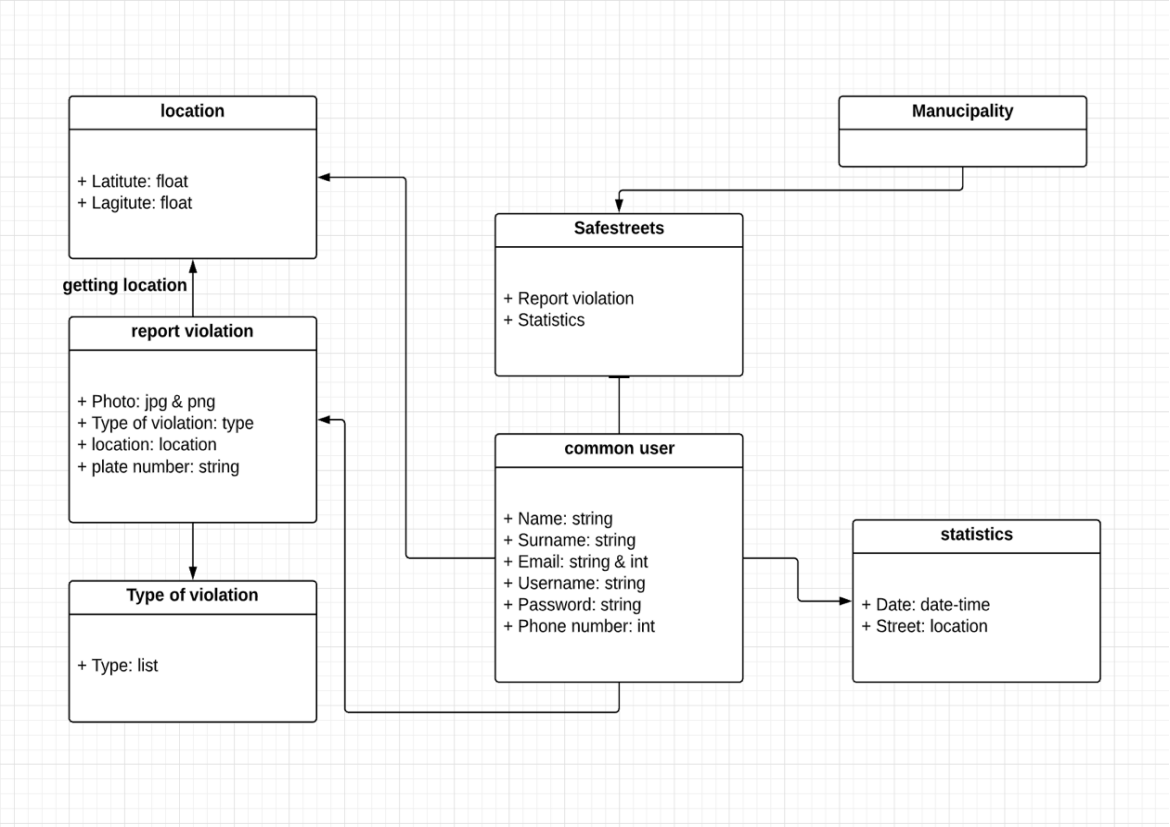


Figure 2.1: Safestreets class diagram

2.2 State Diagrams

In below figures it is possible to view state diagrams of the project:

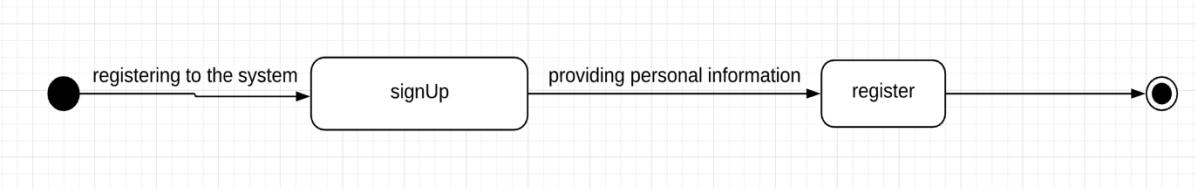


Figure 2.2: common user registration state diagram

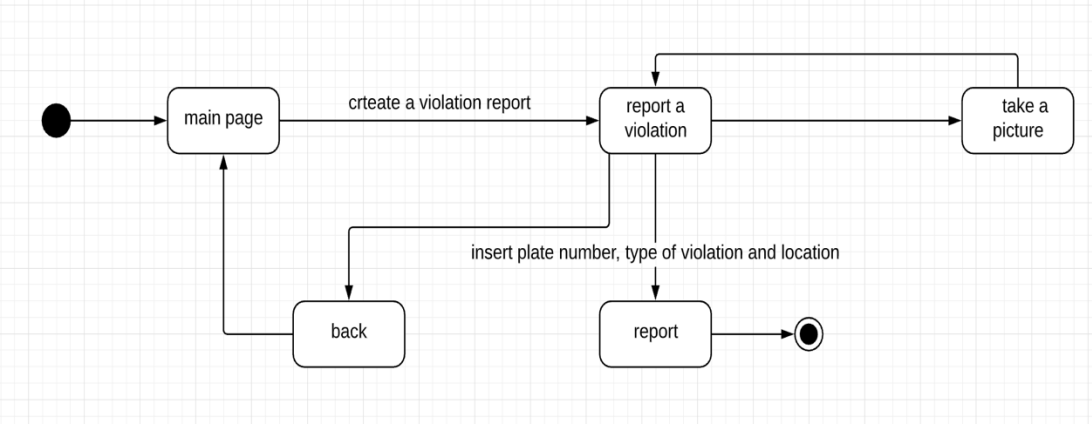


Figure 2.3: Common user reporting violation state diagram

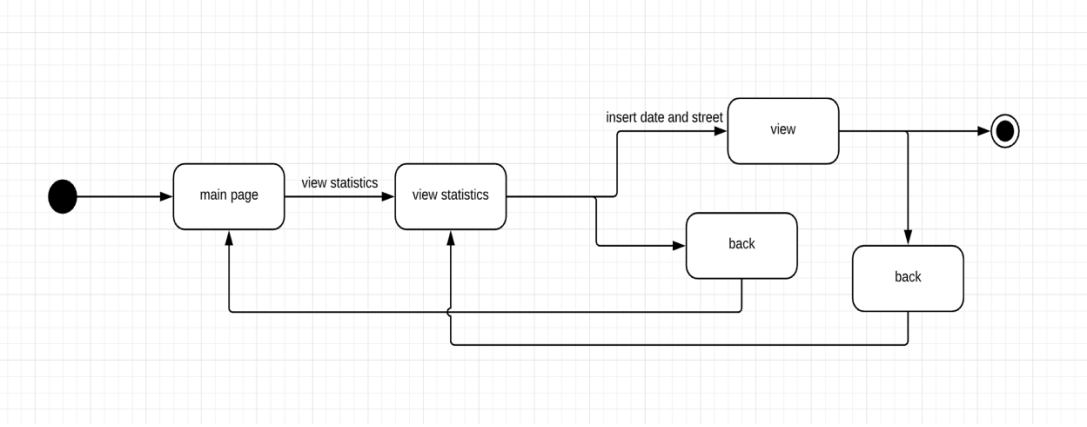


Figure 2.4. Common user view statistics diagram

2.3 Actors

- Common user: a person using Safestreets (through the mobile app). He or she can pass through a successful registration process and now able to use all the safestreet services. He/she can login to the system and, after that, use all the platform's functionalities.
- Municipal: an organization to handle the local traffic and is able to maintain and update the system. Registration for this kind of users is not possible through mobile application and they have to be registered and can benefit from the system through a web platform.

2.4 Domain Assumptions

D1. All users have to provide credentials and be registered in order to benefit from the system. these credentials include username, password, phone number, email address, name and surname.
D2. A No parking area is a special parking area that no one can park the car there. A restricted parking area is a special parking area that people with disabilities can park there.
D3. As soon as the system send violation report the user can create another violation report again.
D4. To report a violation a user has to provide to the system information consist of plate number, type of violation, picture of the violation and location of the violation.
D5. Smartphone of users is equipped with a camera.
D6. The position provided by the user is accurate
D7. The photo of the plate number of the vehicle involved in the violation must not be vivid and can be recognized easily.
D8. The violation reports will be sent directly to Safestreets system.
D9. Location service of the smartphone is active during reporting.
D10. Using the registered plate number of the violating vehicle, the municipality can retrieve details of the vehicle's owner to issue a ticket.
D11. Accurate violation location is located by GPS.
D12. Currently all Italian cars are registrated with plate scheme format: "AA 000 AA"; on left the common EU design with the country code, on right the year of registration and the provincial code.

2.5 Product Functions

[R.1] Individuals and third parties can register to Safestreets services by providing identifiable information.
[R.2] Safestreets should have access to the location of registered individuals from their devices.
[R.3] Municipality can access to the data of violations by a web base platform.
[R.4] Common users are able to exploit the mobile application platform to report a violation.
[R.5] Common user can exploit the mobile application to view desired statistics.
[R.6] Safestreets should validate reports before storing them in concerns of image quality and mandatory fields.
[R.7] Safestreets should have access to the thickest and trends that municipality exploited from the reports.
[R.8] Safestreets should provide a web platform for municipality so that they can perform data mining tasks and use reports to issue traffic thickets to rouge drivers.
[R.9] Safestreets is enabled a mechanism to avoid common users to alter input information in specific plate alteration.

3. Specific Requirements

3.1 Constraints

3.1.1 Regulatory policies

The system will ask for users' permission to have access to its camera and position and safely storing them.

3.1.2 Hardware Limitations

- Smart phones with a camera for common users
- 3G/4G connection
- Up and running GPS
- A PC with a browser for municipality user
- It is just a recommendation for other versions of the software. Municipality should assign each car an RFID which is located inside the car behind the front window. Regarding this, the reporter has to scan the RFID code and attach it to the report. So that the municipality can check if the number of the car is matched with the RFID code or not and find out the validation of the report.

3.1.3 Software Limitations

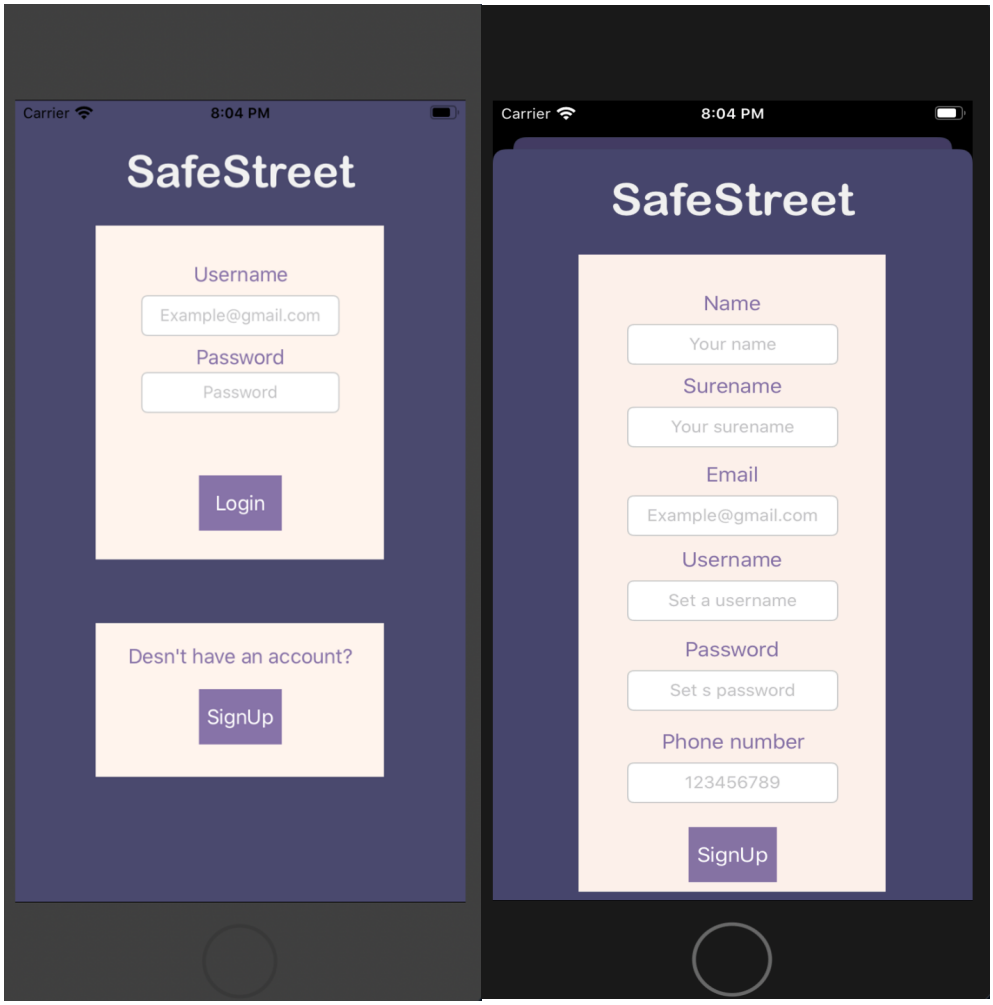
- IOS version 10 or later for common users
- Google chrome browser version 54.0.2840.71or later

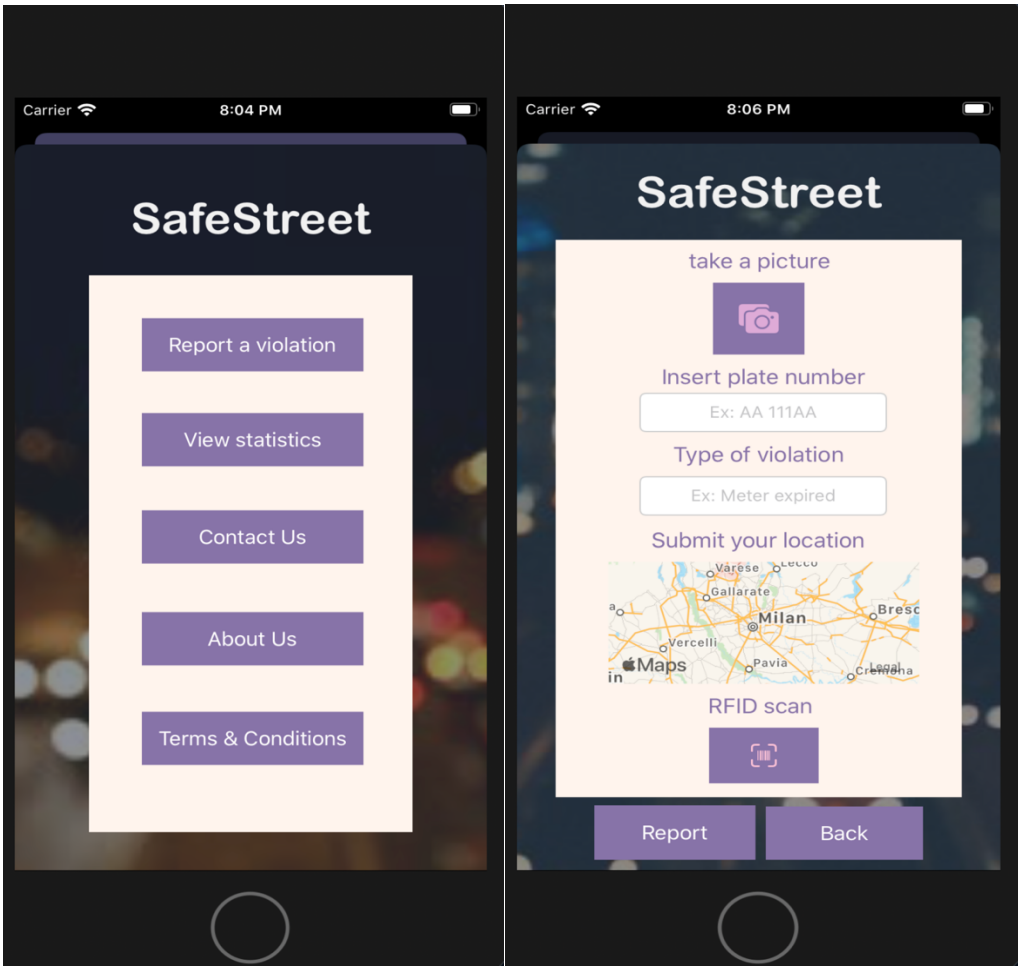
3.1.4 Dependencies

- The safestreet will depend on Map service API's to define the place of violation.
- The app is dependent on the camera on the smartphone.

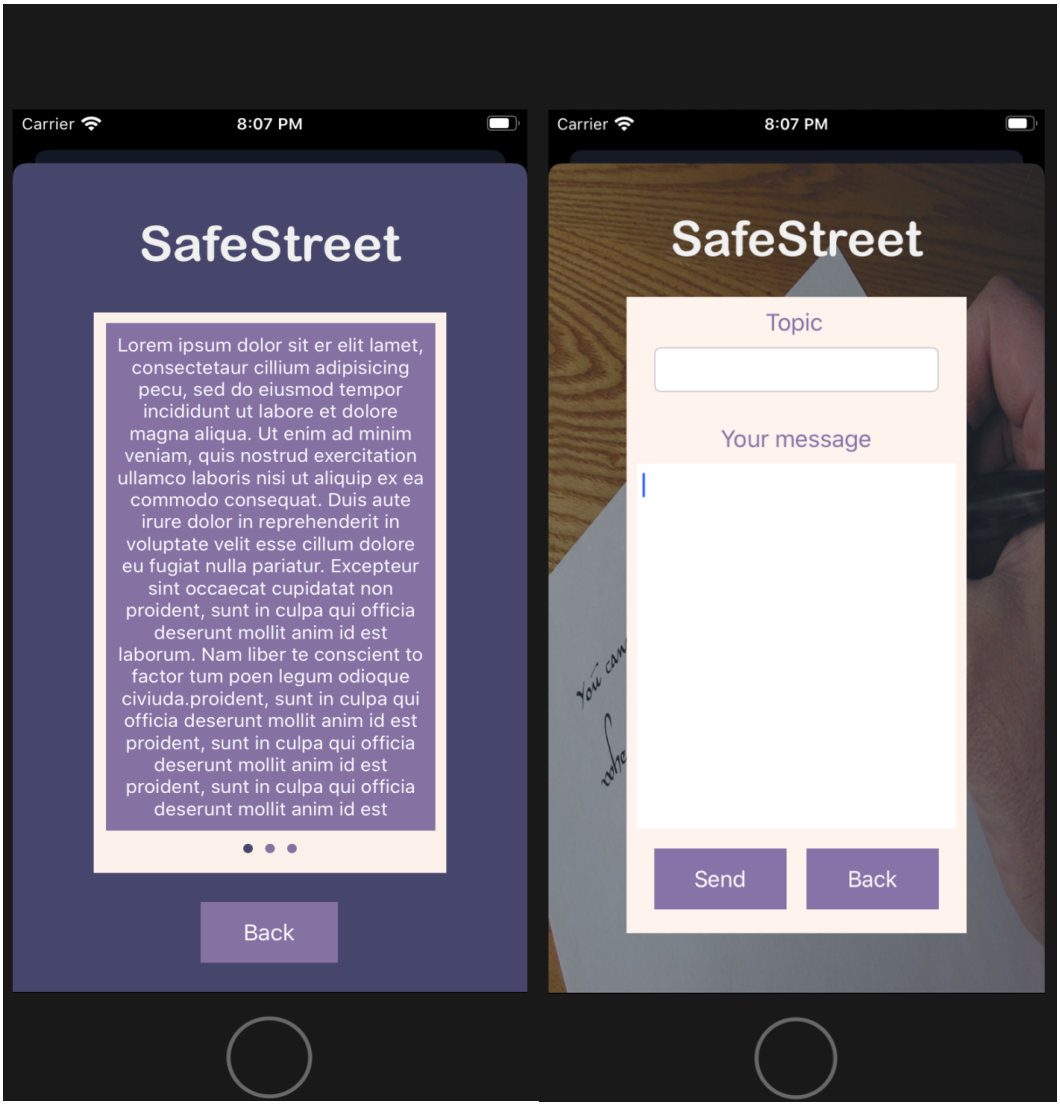
3.2 User Interface

3.2.1 common user




Carrier
8:04 PM
SafeStreet
Report a violation
View statistics
Contact Us
About Us
Terms & Conditions
Carrier
8:06 PM
SafeStreet
take a picture
Insert plate number
Ex: AA 111AA
Type of violation
Ex: Meter expired
Submit your location
Varese
Gallarate
Milan
Vercelli
Pavia
Maps
RFID scan
Report
Back


Carrier
8:06 PM
SafeStreet
Choose date from:
Fri Dec 13 7 05 AM
Today 8 06 PM
Sun Dec 15 9 07
To:
Fri Dec 13 7 05 AM
Today 8 06 PM
Sun Dec 15 9 07
Street name
Ex: viale certosa
View
Back
Carrier
8:06 PM
SafeStreet
Double parking
Handicap parking
Residential parking
Sidewalk parking
Special parking
15.79%
5.26%
21.05%
26.32%
31.58%
Back


Carrier
8:07 PM
SafeStreet
Lorem ipsum dolor sit er elit lamet, consectetaur cillium adipisicing pecu, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Nam liber te conscient to factor tum poen legum odioque civiuda.proident, sunt in culpa qui officia deserunt mollit anim id est proident, sunt in culpa qui officia deserunt mollit anim id est proident, sunt in culpa qui officia deserunt mollit anim id est
Back
Carrier
8:07 PM
SafeStreet
Topic
Your message
Send
Back

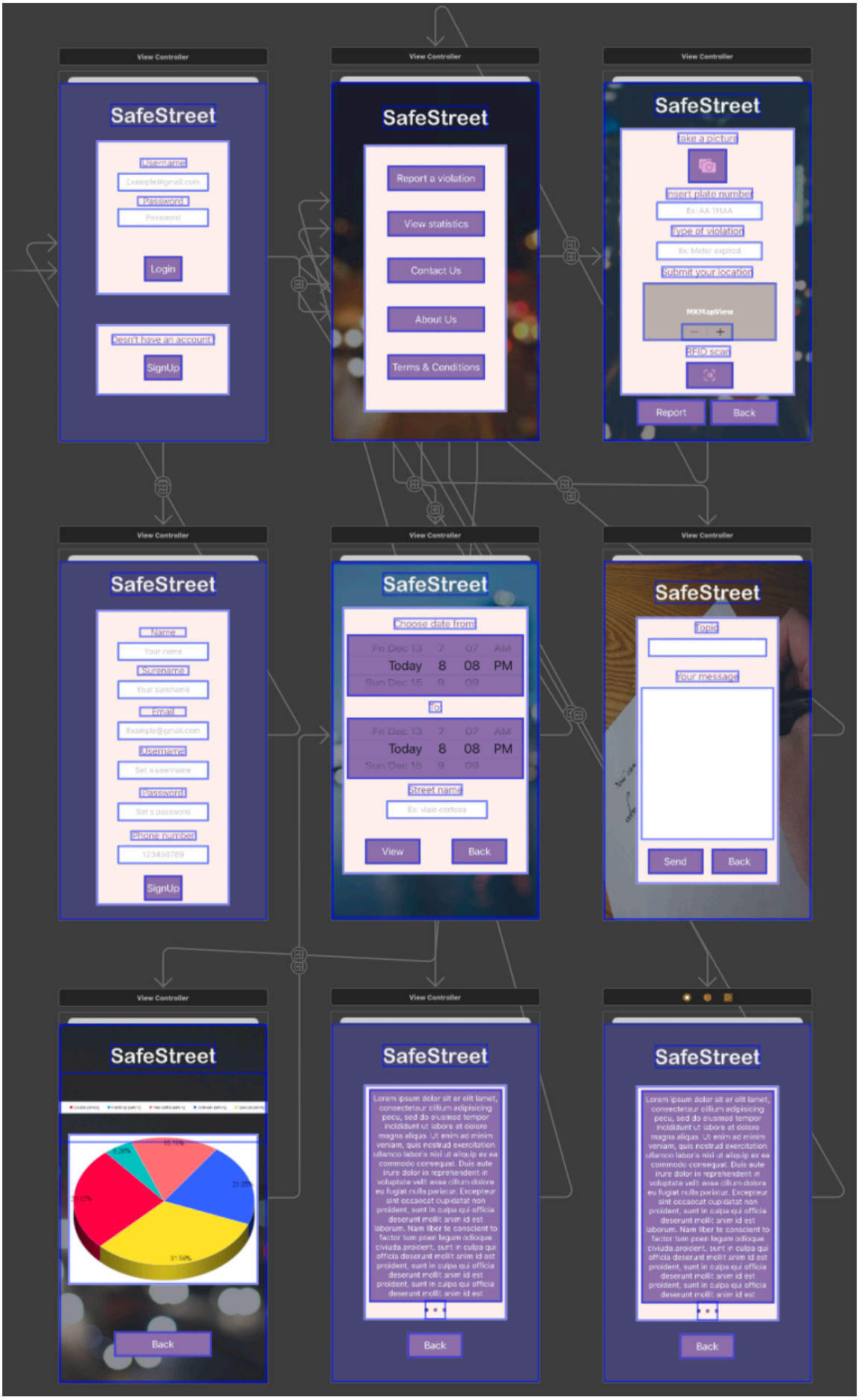

View Controller
SafeStreet
Username
Example@gmail.com
Password
Password
Login
Doesn't have an account?
SignUp
View Controller
SafeStreet
Report a violation
View statistics
Contact Us
About Us
Terms & Conditions
View Controller
SafeStreet
Take a picture
Insert plate number
Ex: AA 111AA
Type of violation
Ex: Meter expired
Submit your location
MKMapView
RFID scan
Report
Back
View Controller
SafeStreet
Name
Your name
Surename
Your surename
Email
Example@gmail.com
Username
Set a username
Password
Set a password
Phone number
123456789
SignUp
View Controller
SafeStreet
Choose date from
Fri Dec 13 7 07 AM
Today 8 08 PM
Sun Dec 15 9 09
To
Fri Dec 13 7 07 AM
Today 8 08 PM
Sun Dec 15 9 09
Street name
Ex: viale certosa
View
Back
View Controller
SafeStreet
Topic
Your message
Send
Back
View Controller
SafeStreet
Back
View Controller
SafeStreet
Lorem ipsum dolor sit er elit lamet, consectetaur cillium adipisicing pecu, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco labor s nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit assa cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Nam liber te conscient to factor tum poen legum odioque civiuda.proident, sunt in culpa qui officia deserunt mollit anim id est proident, sunt in culpa qui officia deserunt mollit anim id est proident, sunt in culpa qui officia deserunt mollit anim id est
Back
SafeStreet
Lorem ipsum dolor sit er elit lamet, consectetaur cillium adipisicing pecu, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco labor s nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit assa cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum. Nam liber te conscient to factor tum poen legum odioque civiuda.proident, sunt in culpa qui officia deserunt mollit anim id est proident, sunt in culpa qui officia deserunt mollit anim id est proident, sunt in culpa qui officia deserunt mollit anim id est
Back

Figure 3.1 common user interface

3.2.2 Municipality

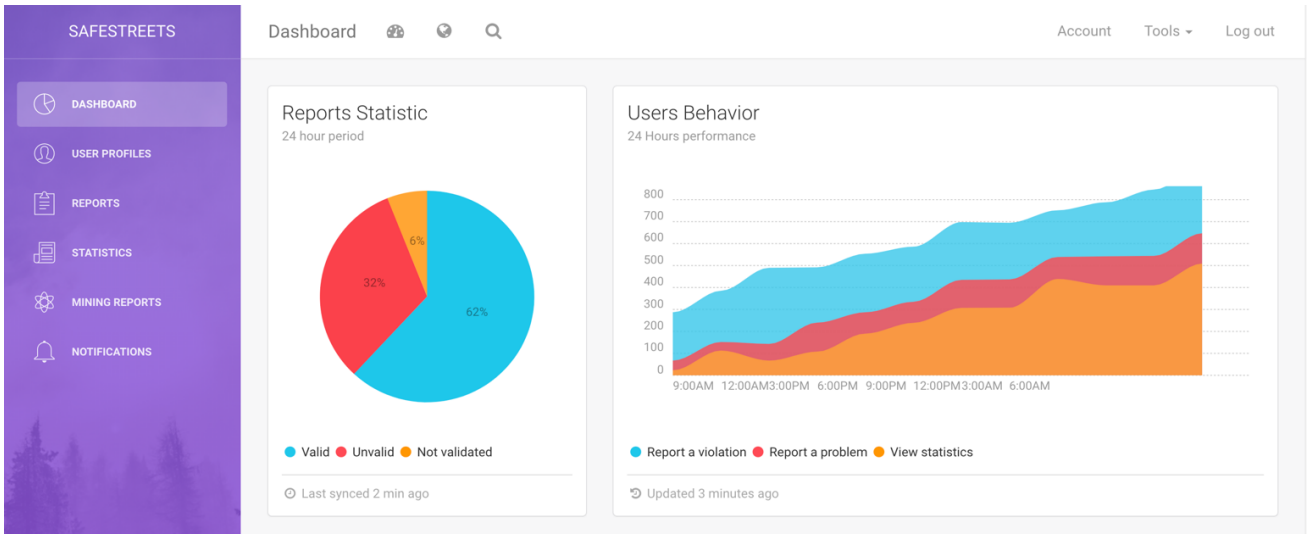


Figure 3.2 Municipality user interface

3.3 Scenarios

Scenario One: Report Validation

Jack is a public transport bus driver. It has occurred a lot when he arrives at a bus station, finding a car parked in that bus station. He has heard about SafeStreet and decides to take advantage of this app to notify authorities about drivers who are not law obedient, so he signs up for the app. While he stops at bus station and waiting for passengers to board, he carelessly takes a picture of a car that parked in that bus station. Since the picture doesn't fulfill the required conditions of what app has specified, jack's report is denied.

Scenario Two: Report Violation

Alex is a handicap driver. It's crucial for him to be able to park his car in handicap parking slots since he has to go to his office by his car every day. Unfortunately, most of the time there are careless drivers who doesn't respect this matter and park their car in those parking slots. So, Alex decides to exploit SafeStreet application do notify authorities about those rough drivers. One day he faces this situation so after Singin he takes a picture of situation along with that car's plate and successfully reports this minor offence.

Scenario Three: Provide web platform to municipality

Sofia is a traffic control operator in municipality of Milan, and she has been tasked to detect streets with highest violation of double-parking cars where it's prohibited so that municipality can plan to place elevator car parking to address the issue. She has heard about SafeStreet application which can indicate such streets by the help of users who report such violations such as double-parking. So, she signs a contract with SafeStreet corporation to receive such information by the help of a web platform which SafeStreet provides.

3.4 Functional Requirements

Relating due to requirements listed in the section 2.5, with the goals defined in section 1.4 and domain assumptions defined in section 2.4, the table 1 below is created as follows

Req_ID	Goal_ID	Domain_ID
R.1	G.1	D.1
R.2	G.2.5, G.2.6	D.7, D.10, D.12
R.3	G.4	D.5, D.7, D.11
R.4	G.2	D.2, D.5, D.6, D.7, D.8, D.12
R.5	G.3	D.11
R.6	G.2.4, G.2.6	D.8
R.7	G.5	D.11
R.8	G.4	D.5, D.7, D.11
R.9	G.6	D.13

Table 1: Mapping of Requirements table

3.5 Use Case Diagram

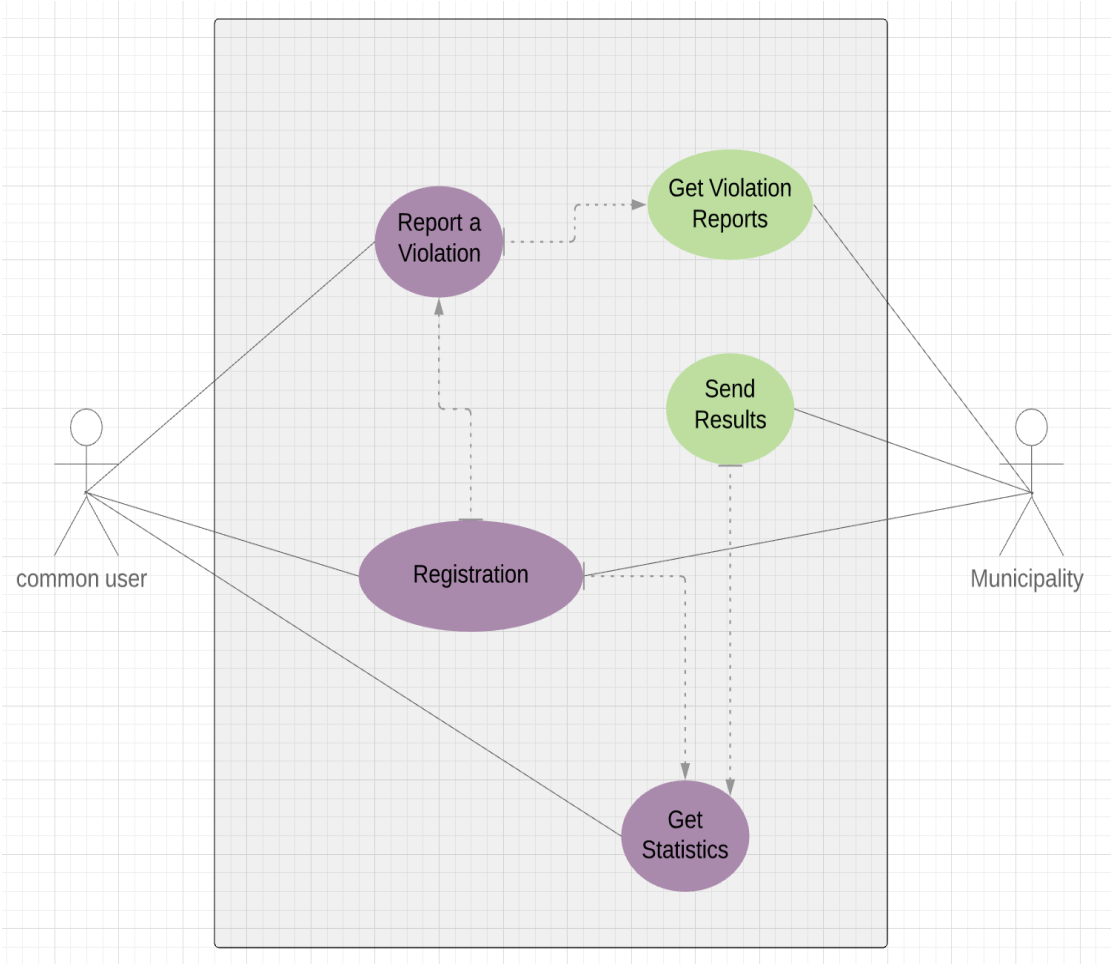


Figure 3.1 Safestreets use case diagram

3.6 Use Case Tables

Name	Registration
Actors	Common user
Entry condition	There are no entry conditions
Event Flow	1. The common user on the home page clicks on the “Sign UP” button to start the registration process. 2. The common user fills all the mandatory fields and provides his/her information. 3. The common user clicks on the “register” button. 4. The system saves the data.
Exit condition	The common user successfully ends the registration process and become a new User. From now on he/she can login to the application providing his/her credentials and start using SafeStreet.
Exceptions	1. The common users can register to the system once. 2. The common user not inserting information in one or more mandatory fields. 3. The common user chooses a username that has already been taken by another user. 4.The Visitor chooses an email that has been associated with another user. All exceptions are handled by notifying the issue to the Visitor and taking back the Event Flow to the point 2.

Table 3.1: Safestreets use case table – Registration

Name	Report a Violation
Actors	Common user
Entry Conditions	The User is already on the home page.
Event Flow	1.The User inserts Login information in the SafeStreets. 2.The User push the “Log In” button in order to access. 3.The User press report a violation button in the home page to report a violation then fills all the mandatory fields and take a picture of the violation. 4.The User clicks on the “report” button. 5.The system first validate the quality of repot then saves the data. 6. The system processes the data and extracts the properties of the car. 7. The system makes a report.

	8.The system sends the report to the municipality.
Exit Conditions	The User is successfully redirected to the Home Page.
Exceptions	1. The User inserts a not valid username. 2. The User inserts a not valid password. 3. quality of the picture doesn't meet the requirement 4. mandatory fields are not filled All exceptions are handled notifying the issue to the Visitor and taking back the Event Flow to the point 2.

Table 3.2: Safestreets use case table – Report a Violation

Name	Get statistict
Actors	Common user
Entry Conditions	1.Municipality have received data from SafeStreet 2.Municipulity considered the data and processed them and issued tickets and sent back to Safestyreets
Event Flow	1. The common user insert login information in the SafeStreets. 2.The User push the "Log In" button in order to have access to home page. 2.The user tap into view statistisc 3.the user chooses the date, street name and push view button 4.common User can see the updated statistics
Exit Conditions	The User is successfully redirected to the Home Page.
Exceptions	1. The process of receiving data from municipality to s fails, In this case the common user redirects to line 2 of event flow.

Table 3.3: Safestreets use case table – get statistics

3.7 Sequence Diagrams

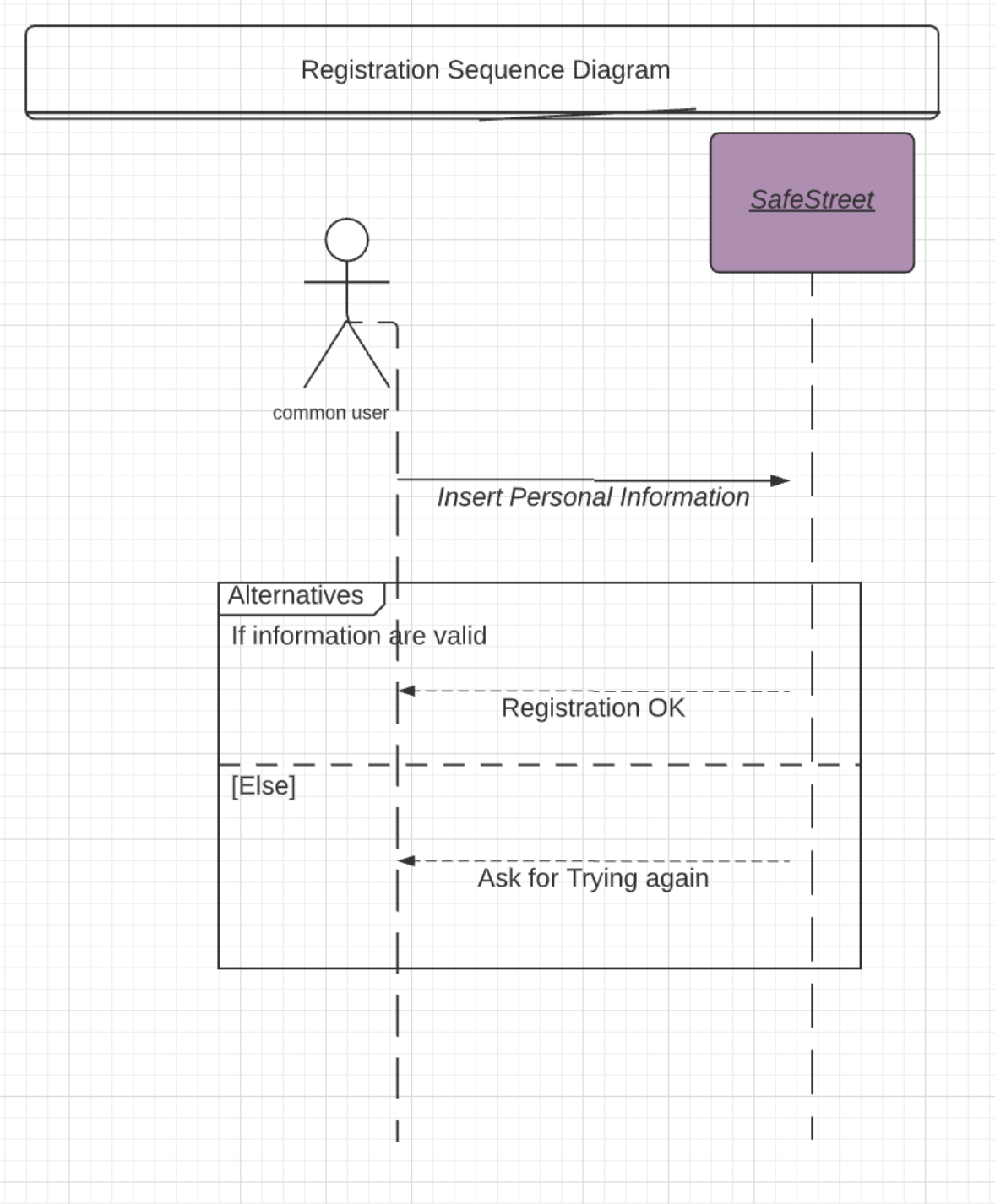


Figure 3.2: Sequence Diagram – Common User Registration

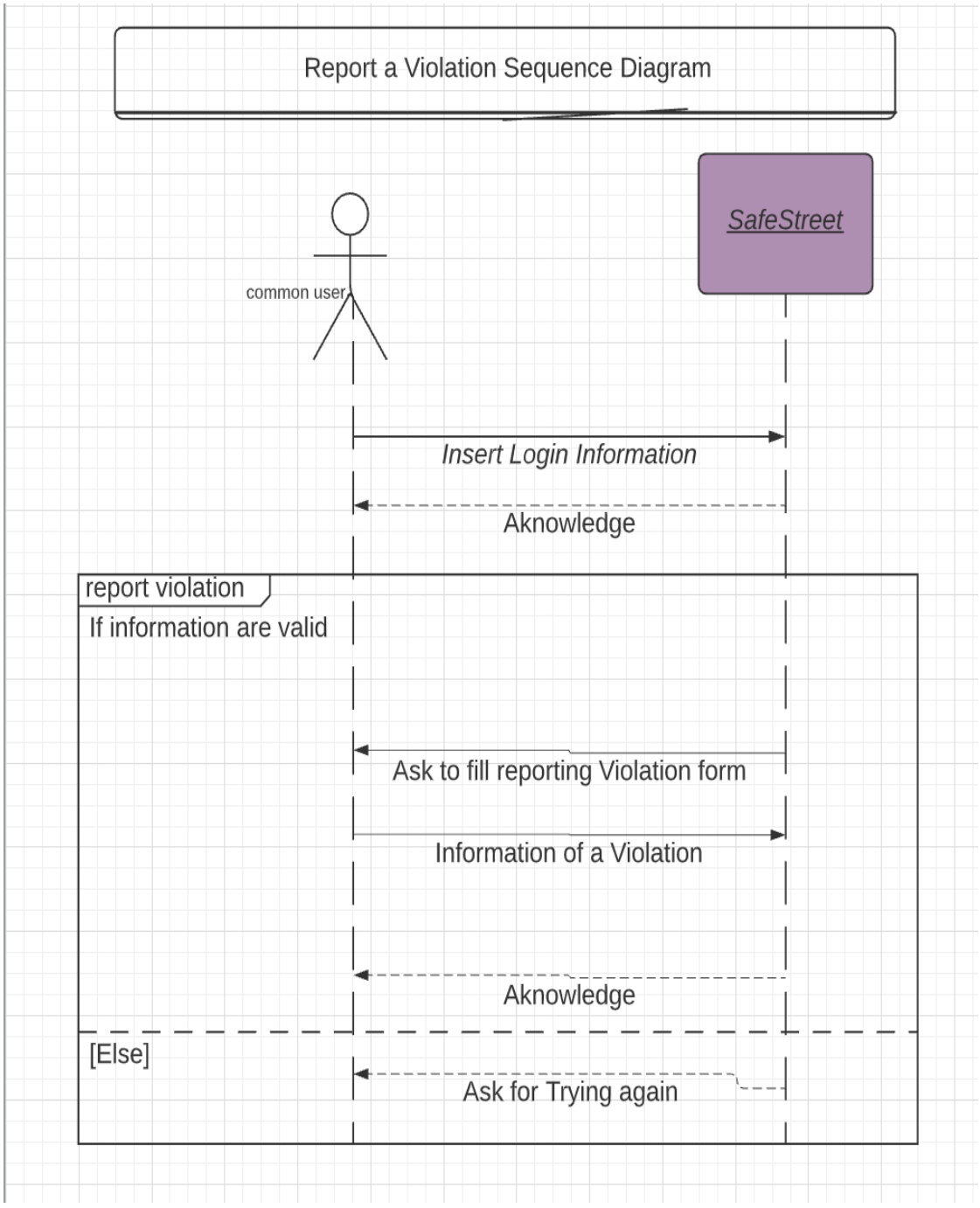


Figure 3.3: Sequence Diagram – Report Violation

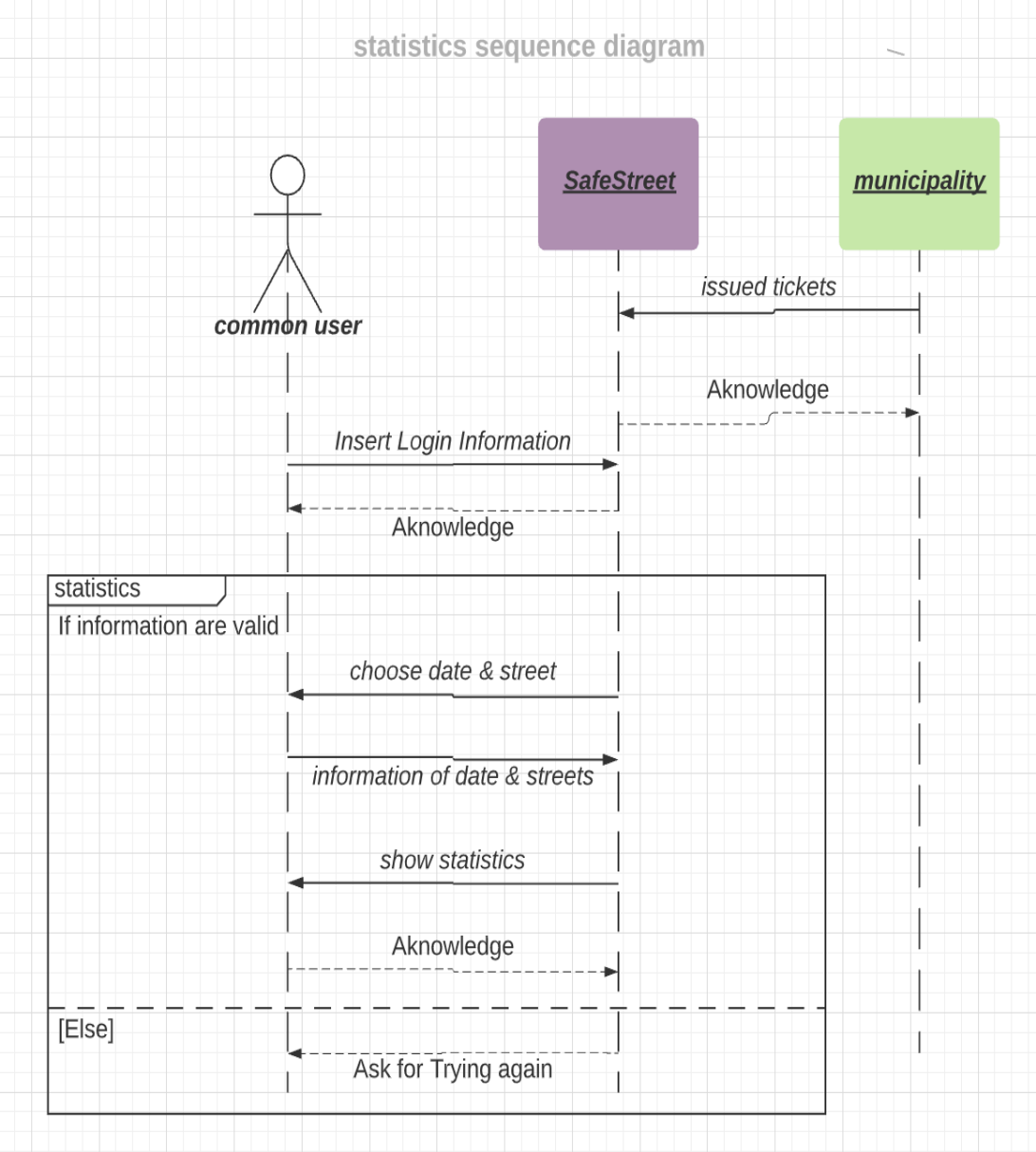


Figure 3.4: Sequence Diagram – Statistics

3.7 Performance Requirements

Statistics service is depending heavily on validated reports so the part of the system which is related to report violation.

3.8 Software System Attributes

3.8.1 Reliability

The application provides a reliable service, as it limits the users to take a picture of the violation at the moment and does not let them to upload pictures from the past so that we are sure the pictures are from the moment and not edited. All the data are sent to the municipality without any additional manipulation so that they can trust the records.

3.8.2 Availability

The availability of the application depends on the device state, whenever it connects to the internet, the reports which are saved on the device are sending to the SafeStreet servers and save in database.

3.8.3 Security

The systems use hash functions to store user's data. In addition, in the reports to the municipality the personal information of the user is not sent.

3.8.4 Maintainability

The application will be implemented and designed with a complete documentary which makes it easier to maintain and understand the application for the future. Additionally, the user interface is designed simply so that every user and third parties in the near future will figure out the usage. Furthermore, as the system sends all the data of the day to the municipalities, not much data will be loss in case the system breaks down. Also, the system takes backups each night so the statistics will be safe again.

3.8.5 Portability

This application is able to run only on iPhone systems. SafeStreet wants to focus on iOS because of its unique features.

4. Formal Analysis Using Alloy

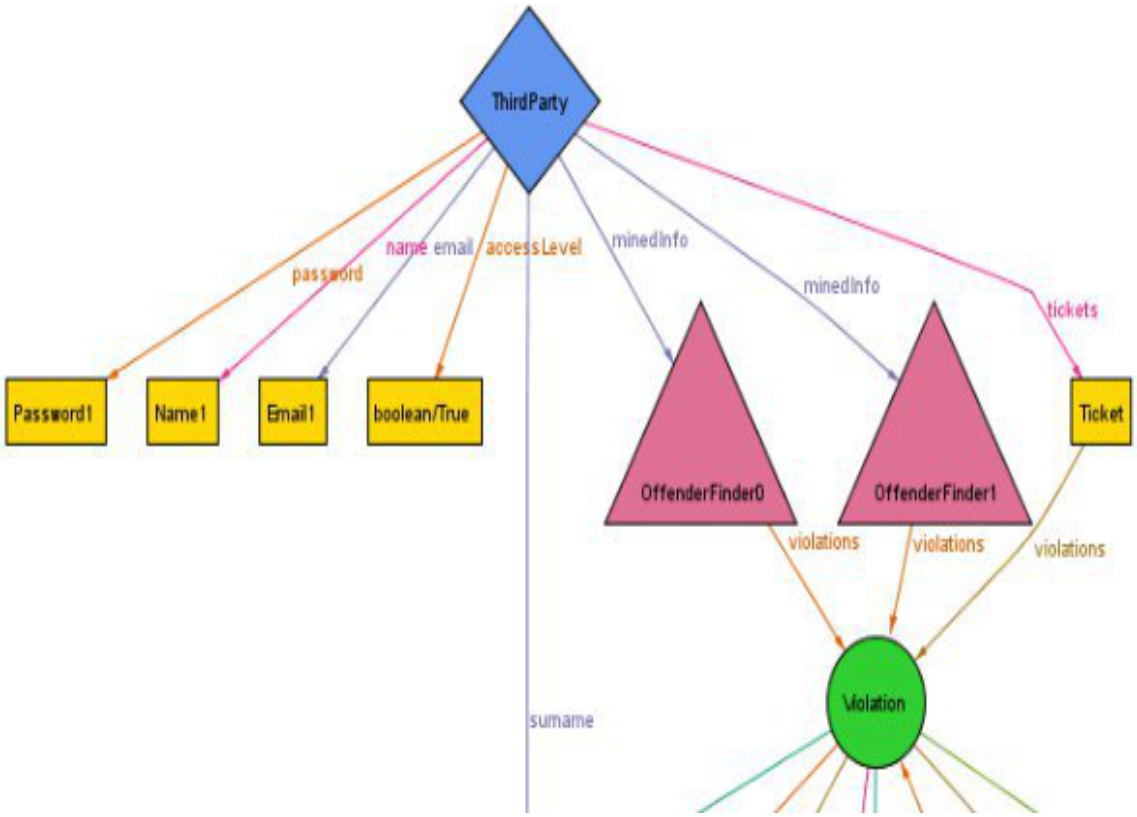
As the project contains two services, it's been decided to integrate the alloy models in order to simplify the analysis and understanding of the worlds and results obtunded.

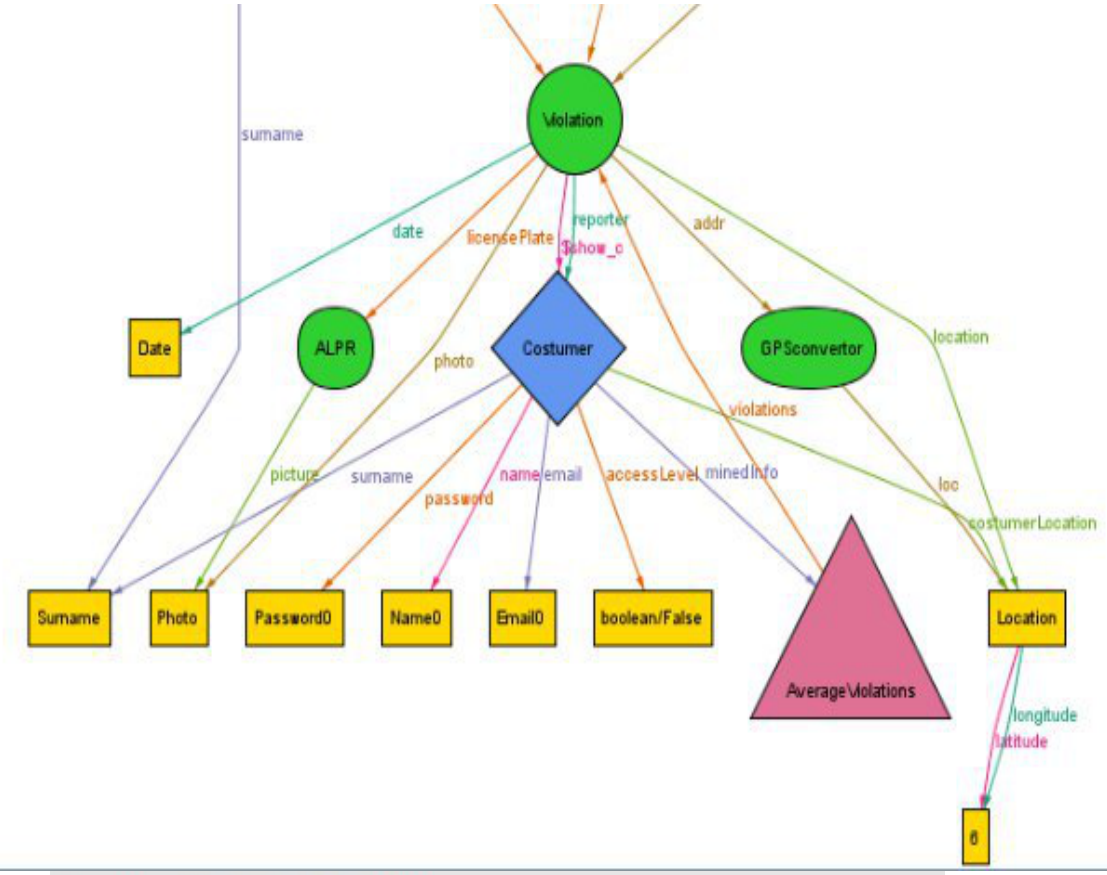
4.1 Alloy Model

```
sig string {}
sig Name, Surname{}
sig Email, Password{}
sig Location { latitude: one Int , longitude:one Int}
sig Photo {}
sig Date {}
// Class User is Interface which includes main attributes of every kinds of User.
abstract sig User {
name: one Name,
surname:  one Surname,
email: one Email,
password: one Password,
accessLevel: one Bool, minedInfo: some MiningModule,}
// Class Costumer which inherits main attributes of User class. also it has Location which
abtained by
// Navigation Unite of his/her Mobile Phone.
sig Costumer extends User{costumerLocation: one Location,}{accessLevel = False}
// Class Costumer which inherits main attributes of User class. also it has 1 to n relation
with tickets.
sig ThirdParty extends User{tickets: some Ticket,}{#tickets >= 0 accessLevel = True}
// Violation has dependent relation with user (here reporter), it must be constructed by only
one Costumer
// it obtains it's location from Navigation Unit's of Costumer's mobile, gives it to
ReverseGioCoding and get
// exact address. In addition to ReverseGioCoding, it has interaction with ALPR which extract
license plate
```

```
// from photo sent by costumer.
sig Violation {location: one Location,addr: some GPSconvertor,reporter: one Costumer,photo: one
Photo,licensePlate: one ALPR,date: one Date}
// ReverseGioCoding and ALPR are modules and we have only one instance of both.
sig GPSconvertor {loc: some Location,}{#GPSconvertor = 1}
sig ALPR {  picture: some Photo}{ #ALPR = 1}
// MiningModule unit with has two childeren, AverageViolation and OffenderFinder, has 1 to n
relation
// with Violation. As statistical rule, it needs all of violation at least for a particular
period.
abstract sig MiningModule {violations: some Violation}
// AverageViolations and OffenderFinder inherit violations from thier parent
sig AverageViolations extends MiningModule{}
sig OffenderFinder extends MiningModule{}
// in contrast with MiningModule entiity, every ticket has 1 to 1 relation with violations.
// if a third party approve validation of a violation, it's ticket issued.
sig Ticket {violations: one Violation}
//////// Facts
// Every Costumer Has Espesific Location
fact {no disjoint c1, c2 : Costumer | c1.costumerLocation = c2.costumerLocation}
// Every User Has Espesific Email
fact {no disjoint u1, u2 : User |u1.email = u2.email}
// Every User Has Specific Password
fact {no disjoint u1, u2 : User |u1.password = u2.password}
// Every User Ordered His/Her Spesific Set of Data Mining Processes
fact {all disjoint u1,u2: User |u1.minedInfo != u2.minedInfo}
// No Same Violation Belongs to Two Reporters
fact { no disjoint v1, v2 : Violation |v1.reporter = v2.reporter}
// as Every Costumer Has Specific Location, The Number of Each One is The Same
fact EqualUserAndLocation{ #Costumer = #Location}
// Enterance Location is the same as User Location, then No Same Location For Different GPS
convertor
fact  { no disjoint revGio1, revGio2: GPSconvertor |revGio1.loc = revGio2.loc}
// No Same Photo For Different Violations
fact {no disjoint v1,v2 : Violation |v1.photo = v2.photo}
// Each Ticket Issued by One Third Party
fact  {all t: Ticket |one tp: ThirdParty | tp.tickets = t}
// Every User Just Can Use No.1 Service Of Mining Module, AverageViolations
fact  {all c: Costumer |c.minedInfo = AverageViolations}
// Each Photo Belongs to Just One Alpr
fact {all ph: Photo|one alpr: ALPR |alpr.picture = ph}
// the Location of Costumer is Sent to GPS convertor. Therefore these are same
fact  { one revGio: GPSconvertor| one c: Costumer |revGio.loc = c.costumerLocation}
// the Location of Costumer is Sent to Violation Unit . Therefore these are same
fact { all viol: Violation |some c:Costumer|viol.location = c.costumerLocation}
// Each Date Belongs to one Violation
fact {all d: Date, viol: Violation |viol.date = d}
// if a ticket exist, it belongs to one violation
fact {one t: Ticket , v:Violation |t.violations = v}
```

```
pred show {} run show for 3
```





Executing "Run show for 3"

Solver=sat4j Bitwidth=4 MaxSeq=3 SkolemDepth=1 Symmetry=20
2803 vars. 312 primary vars. 4923 clauses. 81ms.
Instance found. Predicate is consistent. 55ms.

5- Effort Spent

Niloofar Salimi

Task	Hours
Definitions-Document Structure-Document Structure-Requirements	2.5
Use Case Diagrams-Use Cases-State Diagrams-Scenarios-Sequence Diagrams	5
Alloy	8
Purpose, Scope, Current System-Goals-Domain Assumptions World and Shared Phenomena-	5.5
Use Cases-Software System Attributes-Communication Interfaces-Hardware interface	2
Goals	5
Total	28

Arash Bahariye

Task	Hours
Chapter 1	3.5
Chapter 2	2.5
Chapter 1, 2	1.5
Chapter 3	6.5
Use case Diagram and Scenarios	4.5
Sequence Diagram	5
Completing Chapter 3	3
Integrating Documents	1
Total	28