POLITECNICO
MILANO 1863

Requirements Analysis and Specifications Document

Software Engineering II Project - A.Y. 2019-2020

SafeStreets

Authors	ID Numbers
Andrea Furlan	XXXXXX
Cosimo Russo	XXXXXX
Giorgio Ughini	XXXXXX

October 31, 2019 Version 1.0

Contents

1 Introduction

1.1 Purpose

The main purpose of SafeStreets is to create a software that provides users the possibility to notify authorities when parking violations occur providing some useful features such as finding the most unsafe areas around them and proposing suggestions to the municipality. In addition, SafeStreets will enable the Local Police to generate traffic tickets from it and to cross all the information it owns with the data of the accidents happened.
Specifically, we want to realize a product which is able to:

- Retrieve pictures uploaded by users of parking violations with possible attached information such as license plate position in the image, type of violation and GPS meta-data.
- Automatically complete the data of a reported violation running a recognition algorithm able to read license plate text.
- Highlight to users the areas with the highest frequency of violations and information about vehicles that commit most violations.
- Automatically identify potentially unsafe areas crossing SafeStreets' information with accident datas from the Local Police, possibly suggesting possible interventions.
- Send violations data to the Local Police to automatically create new traffic tickets if it can be proved that the chain of custody of the information coming from the users is never broken.
- Generate statistics related to ticket emissions to inform users about how effective SafeStreets is.

On the other hand, the purpose of this paper is to define in a detailed way all the functions and requirements of the application.
In doing this, we start focusing on a brief overview to characterize the product with relevance to its interaction with the world, then we will proceed deeply in analysing which functions are relevant and should be provided, and which requirements are needed to the stakeholders.

1.1.1 Goals

- [**G1**] Notifing officers when particular parking violations occur.
- [**G2**] Permitting both users and officers to learn which areas have the highest frequency of violations.
- [**G3**] Permitting both end users and officers to learn which vehicles commit the most violations.
- [**G4**] Suggesting possible interventions to potentially unsafe areas.
- [**G5**] Allowing the local police to generate traffic tickets from SafeStreets data.
- [**G6**] Building and exhibiting statistics.

1.1.2 Traceability Matrix

Since goals, functions and constraints are related to each other a traceability matrix is provided in order to enlight the various relationships among them.

Goal ID	Functions ID	Constraints ID	UseCases ID	Other references
G1	-	C1, C2, C5	-	-
G2	-	C3	-	-
G3	-	C4	-	-
G4	-	C1	-	-
G5	-	C5	-	-
G6	-	C2	-	-

Table 1: Traceability Matrix

1.2 Scope

As our software needs to be compliant with different laws and as it needs to interact with the Local Police, initially, SafeStreets will have a restricted geographic domain coincident with the Italian city of Milan.
Indeed, in order to provide the most complete service, SafeStreets will require the access the Local Police web application to be able to process traffic tickets.
It goes without saying that to organize this kind of service in the most effective way we must experiment first this activity in a internationally-visible city, then applying that to anyone who will demand.

1.2.1 The world-machine phenomena

The first model of our system to be presented is the model "The world and the machine" by M. Jackson and P. Zave. This model highlights the division between phenomena that happen entirely either in the world or in the machine, and those that are shared between the two of them.

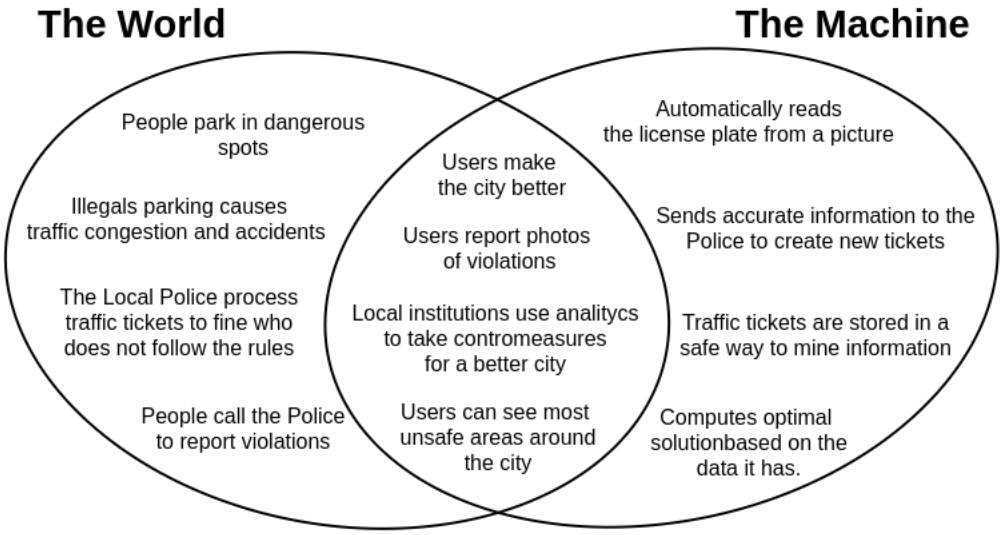

Figure 1: The world-machine phenomena chart.

1.3 Definition and Acronyms

1.3.1 Definitions

1.3.2 Acronyms

1.4 Revision

1.5 Actors

- *Guest*: This actor plays the role of a person who is not registered and thus logged in.
- *User*: This actor refers to the condition of a normal person (not an officer) already signed up and logged.
- *Officer*: This actor represents a public officer that interacts with SafeStreets in some ways.

1.6 References

- The 2019-2020 Software Engineering 2 Project Assignment document
- The IEEE Standard for RASD

1.7 Document Structure

This document is divided in four parts:

- **Introduction**: a description about the goals of SafeStreets and the context in which it will be implemented is provided. Subsections dedicated to the understaing of some acronyms and definitions are also present.
- **Overall Description**: gives an overall description of SafeStreets, focusing on the domain assumptions and the constraints of the application. This section also aims to provide a context to the whole project and to show its integration with the real world. It also shows the possible interactions between the world and the users of SafeStreets.
- **Specific Requirements**: the software requirements, explained in a sufficiently detailed manner to design a system that satisfy them, and the testers to test said requirements are provided. It is also present a detailed description of the possible interactions that can occur between the world and the system, followed with a series of simulations and previews about the above mentioned interactions.
- **Formal Analysis using Alloy**: the requirements are expressed through the Alloy model, with which is possible, since it is a declarative specification language, to define the functions, the constraints and the interactions of SafeStreets.

In the last part of the document a short note that summarize the effort spent in producing this RASD by its authors is shown.

2 Overall description

2.1 Product perspective

The idea is to create an application to allow users to report parking violations without taking much time to their daily life. According to this intention, we would like to realize an extremely friendly user interface and a lightweight software in order to make SafeStreets affordable to many people as possible and runnable by many devices.
Users will certainly be able to exploit the advanced functions of SafeStreets such as charts and analitycs, but as those functions rely over data, the basic violations reporting function will be the core one.

Since a small downtime of SafeStreets is not going to cause damage to anyone, it will be tolerated without much thoughts. On the other hand, as our software is going to run some kind of OCR and AI recognition algorithm that will probably be expensive in terms of resources, it should be very dynamic to support different queries in a few seconds.
In addition, our software is going to process very specific data that could potentially lead someone to be fined, hence it should ensure that the chain of custody is never broken and the images are never altered.
To upload a new picture on SafeStreets or to view charts about violations, it is obviously required an active and functional internet connection. But as said, as data are the core business of SafeStreets, there will be put in place a mechanism such that a user can insert all the information needed to report someone on his mobile application, then those information will be sent as soon as the internet connection is restored.

Concerning the hardware, we intend to have a database which contains all the historical information about reports made by the people. This database will allow both users and officers to see both aggregated and detailed information that require an huge amount of data to be processed. Hence, the internal database engineering should take this into consideration.

2.2 Product functionalities

This section provides an abstract of the main functions of the application. To be able to use any of the given functionalities, the user must first register and then login to the application by providing a valid email and a password.

2.2.1 Notification of Violations

The base function of the application is the possibility to signal a traffic violation.

The user must send one or more picture(s) of the car in which both the violation and the license plate are clearly visible.

The application will try to automatically get the user position using its GPS system, and will notify the user in case of failure so that it can enter it manually.

The users will then send the following information to the system:

- The pictures selected by the user
- The position of the user
- The current date and time
- The type of violation (to be picked from a pre-defined list)
- An optional comment inserted by the user

The information sent by the user will be stored on persistent storage on the server and the police will be able to see it on their clients.

2.2.2 Data Mining

The system will allow the users to extract statistics about violations in the various areas/streets of the cities in the system, for example a user can find the areas in which most segnalations occurred in the last 3 months.

Data mining must take into account the privacy of the users. To guarantee an acceptable level of privacy, different roles are given to the users and the officers.

In particular, a user will only be able to see statistics provided by aggregated data, never he will see the absoulte numbers but only percentages.

Officers, instead, will have the finest granularity: they will see all the information enriched by the actual number of violations and can drill down to the specific licence plates which committed the violations. They will also have more filters available with respect to the users, for example the possibility to see which cars committed most violations in a given period.

2.2.3 Request for interventions

The system will get information from the local police systems about incidents, including the location, the licence plates of the cars involved, and the infractions committed.

By crossing the data about the incidents with the segnalations from its users, it will be able to find unsafe areas and also to suppose a reason for it and make suggestions. For example, if a road has many cyclists hit and many signals of cars parked on the bike lane, it can suggest to add a barrier between the parking lane and the road. The correlations between the infractions found on the police system and the ones on the SafeStreet system, along with the possible solutions, must be done by hand by some human parties. An artificial intelligence can then help to calibrate when the system should launch a warning, training on the approval/rejection of the previous signals.

The officers responsible for handling these recommendations will see on their clients all the data about the signal, including the number of incidents and signals, and can decide to discard it or approve it, thus keeping it in the system for future reference. All the approved signals will be reachable by the officers, once they have been resolved they can be deleted from the list but will remain in an archive available for the AI.

2.2.4 Automatic Tickets

The user will be able to give tickets when he spots a violation. This works by adding to the functionality "Notification of Violations" the possibility to confirm that he is sure that he is signalling an actual violation and wants to give a ticket. In this case the system reads the licence plate and uses the police system to automatically issue a ticket.

Security is a key aspect of this functionality. The system must be sure that the signal was sent by the user and that it has not been modified. To achieve this, the user must insert a passcode every time he wants to issue a ticket to confirm that he is the actual owner of the account. Also, all

messages are encrypted with TLS and an hash of every picture is included in the message to make sure that it has not been modified.

The system will also be able to build statistics using the ticket system. These data will represent the amount of tickets given in a certain street/area in a given period, the most egregious offenders, and the change in the number of tickets in a certain place during time. Just like in the Data Mining section, users will not be able to obtain information on single people, they will only see aggregated data and percentages instead of total numbers. The officers will instead have the possibility to see the actual numbers and to get information about specific license plates.

2.3 User characteristics

The users of SafeStreets can be both males or females of any age with no particular limitations. Of course, said users should have at least a basic knowledge of smartphones and electronic devices in general, especially on how to make photos and videos. Users should also have an e-mail address, primarily used to register and authenticate themselves.
Officers instead can also be both males or females but they need to be actual public officers enrolled in the Local Police. They should have a medium knowledge of networks, softwares and hardwares as well as a fully understanding of how traffic tickets laws can be applied, and they should be capable of using municipality tools and softwares fluently.

2.4 Assumptions and dependencies

2.4.1 Domain Assumptions

- A user should input only correct data when reporting a violation, for example the license plate position should be correct
- The image and picture meta-data aren't altered by the user who first submit the report
- The municipality will provide provide correct and complete information about the accidents
- The municipality services are supposed to be functional during the uptime of SafeStreets.

2.4.2 General Assumptions

- A user cannot access any of the functionality of the application if he does not register first. Since security in the application is a fundamental part and users must be accountable for their signals, the registration approach will be the one used by car sharing services: the user will provide an email, a password, an identity document and a selfie of himself holding such document in which his face is clearly visible. Also, he will provide a 4-digits code that he will have to use every time he chooses to send a ticket, even if he is already logged in. All these data will be approved by hand by an operator during the registration phase.
- Since users are personally accountable for what they do with the app, we suppose that they will use it in the most responsible way. For this reason, all tickets that are inserted correctly will be automatically approved.

2.4.3 Constraints

- **[C1]** It is impossible to send a report without attaching a picture, a description or selecting any reason.
- **[C2]** It is impossible cancel an already sent ticket.
- **[C3]** It is impossible to attach a non-valid address to a picture.
- **[C4]** Is prevented to send a report with a picture that contains non-readable license plate.
- **[C5]** The chain of custody of the information coming from the users shouldn't be broken.

3 Specific requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

3.1.2 Software Interfaces

Considering the domain of our application, we decided to integrate in our project some software components to create an easier-to-use product.
In order to provide users the ability to automatically insert location info into their violation pictures meta-data, we chose to adopt a location provider service. That supplier should expose an API to retrieve the user exact address by his or her GPS position coordinates.
In addition, APIs provided by the municipality are required to retrieve real time information about the accidents that occurred on the territory of the municipality to allow information mining and suggestion proposing.
Finally, the Local Police is required to provide an API to retrieve information about violation coming from SafeStreets users to create and process traffic ticket from it.

3.2 Non functional requirements

3.2.1 Performance

The application should be able to deliver the violations (including the automatic tickets) in an acceptable time, which means at most 15 seconds with a 4G connection, but should average to about 5 seconds. This is not a strict requirement, but it improves a lot the user experience and the possibility that he does not decide to give up with the upload.

The data mining functionality is not as trivial as the delivery of the violations, but keeping in mind the effectiveness of the user experience it should still be able to deliver results with an average waiting time of 5-10 seconds, with an upper bound of 20 seconds.

The request for interventions functionality does not have any time constraints, it should just end. It can be run at any time and its results will be stored on persistent storage, without the need to be used immediately. This functionality does however consume a lot of computer resources, since it must cross a large quantity of data from different sources and it will also

use an AI. For this reason, it will have to run either when the load on the system is low or on a completely separate hardware.

All the other operations that require an internet connection (login, logout...) should be fast enough to not become frustrating for the user to use the application, indicatively they should take 5 seconds at most.

3.2.2 Reliability

The system has no reasons to be highly available except for providing a great user experience. It will have to guarantee a 99.99% availability

3.2.3 Security

Since users will be able to give tickets automatically, security is a key aspect of the application. The information contained in the report of a violation must never be altered, and the login information must be kept safe to prevent hackers from giving false tickets using stolen identities.

Users also give their personal data when registering, and their privacy must be guaranteed.

3.2.4 Scalability

This application will be a pioneer in this field, so it is difficult to foresee how many users will use it and how. A high degree of scalability is therefore required.

3.2.5 Accuracy

The maximium error that the GPS system should commit for the application to accept a position is 15 meters, to let an eventual officer near the violation find the car.

3.3 Scenarios

3.3.1 First scenario

	Creation of a report
Actors	Mario Rossi and Luca Neri: habitual users Maurizio Verdi: a car owner
Flow of events	Maurizio is driving his car to reach his girlfriend house. Once arrived, he leaves his car double parked because he don't want to spend much time looking for a park. Mario is - as always - taking his dog out before dinner and is directed trough the park situated five minute by food. While walking, he notices that Maurizio's car is double parked and is slowing down all the cars running in the street. His dog is making more than an effort to go to the park and so Mario wouldn't have much time to call the Police to report this violation. Meanwhile, Luca is - as always - getting back home by car. He lives outside the city and he thinks the train is not trustworthy so he chose the car. But getting out of the city at 7PM is always very slow because of the many dribbling the cars should do to avoid other cars parked on the lane and that said, he's late for the dinner with his wife. Mario, still in a hurry, takes out his mobile phone and opens SafeStreets app. From the homepage he create a new report and the process didn't take much time: only what is required to shoot a photo and select the double park infraction from the menu. So, after a few seconds, he was again able to walk with his dog trough the park. That day, Luca ended up arriving late at home and eating alone his cold meal. As soon as Maurizio receives the ticket from the Local Police he realize that, now that the world has SafeStreets, it's worth the time to seek a legal parking to avoid being fined again. As days go by, more car owner realize the same thing and all this ends up in Luca being able to come back home with less pain after his long days of work.

Table 2: Scenario 1

3.3.2 Second scenario

	Data mining and suggestion proposal
Actors	Giovanni Ciano: head officer of the Local Police
Flow of events	As situations like double parking keeps happening in the same area as in Table 2, many car owner are getting more polite and respectful, but it's still not enough. Indeed, one day, Giovanni receives a letter from the mayor to find a way to help solving the city mobility problem in that area. It's a difficult problem to handle by his own and he don't have so much data and suggestion from his officers because it seems that they can't find a pattern. Giovanni is desperate but then the idea: he opens his SafeStreets control panel and see that there are suggestion not still examined. The software has detected the same double parking violation keep occurring and so it alerts Giovanni to proceed to the automatic detection of parking infractions. The automatic detection of those infraction is a mechanism, adopted by the Local Police, that consists in putting a special video camera on a Police car and then driving along the streets to automatically fine all car owners that are not compliant with the local laws. Giovanni then schedule that on next Friday, two officers would proceed this way. After that raid, fifteen different parking violations were discovered all in the same street! Without SafeStreets, Giovanni wouldn't ever imagined that such a disease was taking place in that street, but thanks to the suggestion he received, now car owners would certainly respect the law, avoiding slowing down the traffic in that street.

Table 3: Scenario 2

3.3.3 Third scenario

	Data mining and suggestion proposal
Actors	Matteo Neri: officer of the Local Police
Flow of events	Summer is a great season, there are less workers and students in the city and many of them use bikes or go to work by foot. Every autumn, instead, people in the city start to leave their bike at home and begin to go to work by car. Matteo is perfectly aware of this situation because as always, on October his head officer sends him and his colleagues to a few intensive sessions of heavy fining to car parked in the wrong way. Matteo thinks he could do more for the city than fining people all day, and it is always bored of doing so. From when SafeStreets was introduced, citizen and normal users has becoming impressive reporter of violations and this behaving keeps growing. Having taken consciousness of this, now Matteo and his colleagues can be assigned to more meaningful tasks, for example they could be put to handle the traffic near the schools when it starts or ends. Matteo is now more happy of working and he thinks he's contributing to his city in a better way, improving the kids and their parents quality of life.

Table 4: Scenario 3

3.4 Use Cases

This section contains all the use cases initially described with the use cases UML model, then the most important Use Case have their own table which provide further details such as: involved actors, entry conditions, flow of events, exit conditions and exceptional conditions.

3.4.1 User Page use cases

Figure 2: Use cases relative to the user registration and authentication

3.4.2 Sign up

Name	Sign up
Actors	Guest
Entry conditions	None
Flow of events	• The guest reaches the registration page containing the relative form • The guest fills up the form and clicks on "Sign up" to complete the process • The system redirects the user to his profile page.
Exit conditions	The guest has successfully registered into SafeStreets.
Exceptions	The guest left an empty field or typed something invalid: an error message is displayed and the user is asked to fill the form again.

Table 5: Use Case table - Sign up

3.4.3 Login

Name	Login
Actors	User
Entry conditions	The user has already registered in the past.
Flow of events	• The user reaches the login page containing the relative form • The user types the username and password in the login form and click on the "Login" button. • The system redirects the user to the application homepage.
Exit conditions	The user has access to the application functionalities.
Exceptions	Username and password didn't correspond or the username didn't exist: an error message is displayed and the user is asked to fill the login form again.

Table 6: Login Use Case table

3.4.4 Password Recovery

Name	Recover Password
Actors	User
Entry conditions	The user has already registered in the past.
Flow of events	• The user reaches the login page containing the relative form • The user doesn't remember his password so he clicks on "Password recovery" button and is redirected to the password recovery page. • The user inserts his email and clicks on "reset password". • The system sends an email to the user with a link and instruction to reset the password. • The user chooses and types a new password and confirms. • The system redirects the user to the login page.
Exit conditions	The user has request an email to reset his password
Exceptions	The inserted email doesn't match any user in the database, it is displayed an error message and the user is asked to retype a valid email.

Table 7: Recover password Use Case table