



SafeStreets

Requirements Analysis and Specification Document

Giancarlo Danese - 945265

Davide Savoldelli -

A.Y. 2019/2020 - Prof. Di Nitto Elisabetta

Contents

1	INTRODUCTION	4
1.1	Purpose	4
1.1.1	Goals	4
1.2	Scope	4
1.3	Definitions, Acronyms, Abbreviations	5
1.3.1	Acronyms	5
1.3.2	Abbreviations	5
1.4	Revision history	5
1.5	Reference Documents	5
1.6	Document Structure	5
2	OVERALL DESCRIPTION	6
2.1	Product perspective: here we include further details on the shared phenomena and a domain model (class diagrams and statecharts)	6
2.2	Product functions	6
2.3	User characteristics	7
2.4	Assumptions, dependencies and constraints	7
3	SPECIFIC REQUIREMENTS	8
3.1	External Interface Requirements	8
3.1.1	User Interfaces	8
3.1.2	Hardware Interfaces	9
3.1.3	Software Interfaces	9
3.1.4	Communication Interfaces	9
3.2	Functional Requirements: Definition of use case diagrams, use cases and associated sequence/activity diagrams, and mapping on requirements	9
3.2.1	Scenarios	9
3.3	Performance Requirements	11
3.4	Design Constraints	11
3.4.1	Standards compliance	11
3.4.2	Hardware limitations	11
3.4.3	Any other constraint	11
3.5	Software System Attributes	11
3.5.1	Reliability	11
3.5.2	Availability	11
3.5.3	Security	11
3.5.4	Maintainability	11

3.5.5	Portability	11
4	FORMAL ANALYSIS USING ALLOY: This section should include a brief presentation of the main objectives driving the formal modeling activity, as well as a description of the model itself, what can be proved with it, and why what is proved is important given the problem at hand. To show the soundness and correctness of the model, this section can show some worlds obtained by running it, and/or the results of the checks performed on meaningful assertions	11
5	EFFORT SPENT	11
6	REFERENCES	12

1 INTRODUCTION

1.1 Purpose

This document represents the Requirements Analysis and Specification Document (RASD) for the SafeStreets software project. The goal of this project is to deploy a system that allows users to report traffic/street violations to authorities. Users will have the possibility to send pictures of the violation, together with the type of violation and position of the vehicle. The software focuses particularly on parking violations, including double parking or parking in a disabled-reserved spot.

1.1.1 Goals

- [G1]: Allow users to register to the system with their personal data
- [G2]: Allow users to report traffic/parking violations
- [G3]: Allow users to suggest possible interventions to unsafe areas
- [G4]: Allow authorities to access the violation reports and take measures
- [G5]: Allow authorities to access the Unsafe Areas map
- [G6]: Allow authorities to discard non-valid photos
- [G7]: Retrieve the exact position of the place where the violation is being committed

1.2 Scope

SafeStreets will have an embedded algorithm which will analyze pictures of the vehicle plates sent by the user in order to recognize the vehicle. This information, together with the position of the vehicle and the type of violation that has been committed, will be stored in the software's database. Authorities will have the chance to mine the information retrieved in the database by highlighting the streets/areas in which most of the violations are committed, the type of vehicles which commit most of the violations and which type of violations occur the most, suggesting possible interventions that could be taken.

1.3 Definitions, Acronyms, Abbreviations

- **User Device:** any compatible device with the SafeStreets application, like a smartphone or a computer
- **Personal Information:** information provided by the user during the registration process. It includes name, surname, birth date, address, e-mail address, mobile number.
- **Violation Report:** the act in which users can denounce violations on the streets, by providing the system its position, a photo and by selecting a violation from a precompiled menu
- **Mobile App:** an application that can be run by mobile devices, both smartphones and smartwatches.

1.3.1 Acronyms

- **RASD:** Requirements Analysis and Specification Document.
- **API:** Application Programming Interface.
- **GPS:** Global Positioning System.
- **PRA:** Pubblico Registro Automobilistico
- **AUC:** Authority Unique Code
- **DBP:** Device-bound PIN

1.3.2 Abbreviations

- **App:** application.

1.4 Revision history

1.5 Reference Documents

1.6 Document Structure

This paper refers to the structure suggested by IEEE for RASD documents, with some modifications:

1. **Introduction:** This first section is a general description of the system's scope and goals. It also includes the revision history of the document and its references, definitions, abbreviations and acronyms used along the paper.
2. **Overall Description:** This section includes shared phenomena, requirements and domain assumptions. It also clarifies users' needs.
3. **Specific Requirements:** This section is made up of all the requirements needed for the system, both functional and non functional.
4. **Formal Analysis Using Alloy:** It includes the Alloy model of the described system.
5. **Effort Spent:** In this section you can find information about the hours spent to draft this document.
6. **References:** Here are the references about papers/documents used to support this document.

2 OVERALL DESCRIPTION

2.1 Product perspective: here we include further details on the shared phenomena and a domain model (class diagrams and statecharts)

2.2 Product functions

According to the goals defined in the first section, we can list the most important functions the system needs to work correctly:

- **Registration and Login:** This functionality enables users to register to the system by inserting their personal data, and to login with their chosen username and password. The system must therefore store the data in it's database and retrieve it when queried.
- **Violation Reporting:** The system must provide users with an in-app photo sending functionality where users can send pictures of the vehicles' plate, select the type of violation from the dropdown menu and let their GPS track down their position.
- **Unsafe Areas Map:** The software must cross its reports positions with the municipality to identify unsafe areas and highlight them on a map together with their corresponding violations.

- **Violation Solving:** The system must enable authorities to access the map of violations and as soon as measures are taken, check them as "solved".

2.3 User characteristics

- **Registered User:** A person who registered to SafeStreets, sharing his personal data. He can login to the system by providing credentials and exploit all the services.
- **Authorities:** A Law Enforcement agent who has direct access to the users' reports on their web app and can issue warnings, fines and measures.

2.4 Assumptions, dependencies and constraints

- [D1]: Users' must provide valid credentials when registering to the system
- [D2]: The license plate must be clearly visible in the photo
- [D3]: Users' device must support the GPS technology
- [D4]: Users' device must support Mobile Applications
- [D5]: Users' device must be connected to the internet
- [D6]: Users' device must have a camera
- [D7]: Users' must always take pictures of vehicles only
- [D8]: Users' can't load photos from their device
- [D9]: Authorities must have access to the internet
- [D10]: Authorities must be provided with their authority identification
- [D11]: Authorities must always check the validity of the photos taken

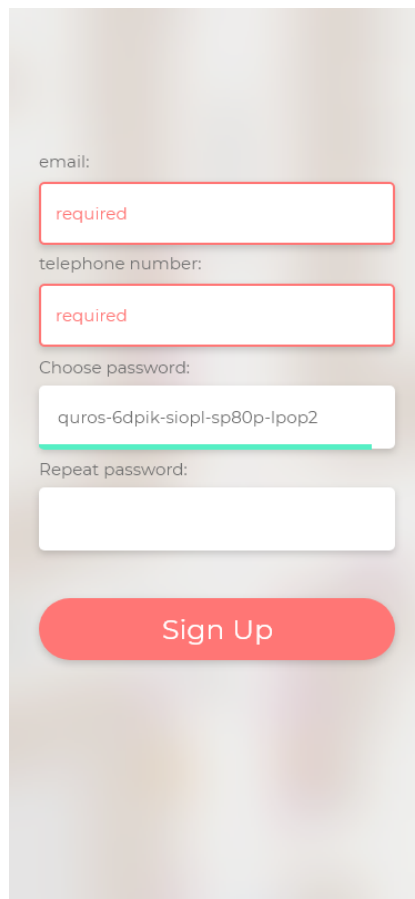
3 SPECIFIC REQUIREMENTS

3.1 External Interface Requirements

3.1.1 User Interfaces

These mockups represent an idea of how the user's interface will be shown and easily accessed, spanning from the simple registration and login to the violation report function.

You can also notice how SafeStreets provides the authorities' with a Web App which is separated both in functions and in design from the normal mobile app destined to users.



email:

telephone number:

Choose password:

Repeat password:

Sign Up

Figure 1: The registration page where you input your credentials

3.1.2 Hardware Interfaces

3.1.3 Software Interfaces

3.1.4 Communication Interfaces

3.2 Functional Requirements: Definition of use case diagrams, use cases and associated sequence/activity diagrams, and mapping on requirements

3.2.1 Scenarios

Scenario 1 - User Registration

Marco decides to contribute to his cities' street security by registering to SafeStreets. He opens the app, and he inserts his personal data, such as name, surname and e-mail. He then decides a username and a password to access the app. He is now ready to report violations and secure the city.

Scenario 2 - Authority Registration

Giovanni is a policeman who wants to register as an authority on the system. He makes a special request, sending his personal information and his authority identification to the system which checks it's validity and sends back to Davide an Authority Unique Code (AUC), which he will need to login, and a Device-bound PIN (DBP), which is bound to special devices in dotation to authorities only. He can now check the users' reports, issue fines and check the Unsafe Areas Map.

Scenario 3 - Violation Report

Roberto is a guy who lives in a street constantly full of double-parked cars. Tired of this situation, he decides to report them to the authorities throughout SafeStreets. He takes a picture of the car with the license plate fully visible, he lets the app track down his position and he selects a type of violation from the dropdown menu.

Scenario 4 - Suggestion Report

Ciro is in his office watching the Unsafe Areas map of his city. In particular, he notices that there's an area in his city full of reports and violations of cars parking in the bike lane and therefore occupying it. So he decides to use the suggestion report function to hint the municipality in placing a protection

barrier parallel to the edge of the bike lane in order to prevent cars from parking on them.

3.3 Performance Requirements

3.4 Design Constraints

3.4.1 Standards compliance

3.4.2 Hardware limitations

3.4.3 Any other constraint

3.5 Software System Attributes

3.5.1 Reliability

3.5.2 Availability

3.5.3 Security

3.5.4 Maintainability

3.5.5 Portability

4 FORMAL ANALYSIS USING ALLOY: This section should include a brief presentation of the main objectives driving the formal modeling activity, as well as a description of the model itself, what can be proved with it, and why what is proved is important given the problem at hand. To show the soundness and correctness of the model, this section can show some worlds obtained by running it, and/or the results of the checks performed on meaningful assertions

5 EFFORT SPENT

- Giancarlo Danese

Day	Subject	Hours
18/10/2019	Purpose, Scope	2
23/10/2019	Domain Assumptions, User Characteristics	1.5
28/10/2019	Definitions, Acronyms, Product Functions	1
30/10/2019	Scenarios, Domains	2

- Davide Savoldelli

Day	Subject	Hours
18/10/2019	Goals, Product Functions	1
23/10/2019	Document Structure	2
28/10/2019	User Interfaces, Mock-ups	3
30/10/2019	Goals, Requirements	1

6 REFERENCES