Table of Contents

[I. Introduction 2](#_Toc23320610)

[I.1 Purpose 2](#_Toc23320611)

[I.1.1 General Purpose 2](#_Toc23320612)

[I.1.2 Goals 2](#_Toc23320613)

[I.2 Scope 3](#_Toc23320614)

[I.3 Definitions, Acronyms, Abbreviations 3](#_Toc23320615)

[I.4 Revision History 4](#_Toc23320616)

[I.5 Reference Documents 4](#_Toc23320617)

[I.6 Document Structure 4](#_Toc23320618)

[II. Overall Description 5](#_Toc23320619)

[II.1 Product Perspective 5](#_Toc23320620)

[II.2 Product Functions 5](#_Toc23320621)

[II.3 User Characteristics 5](#_Toc23320622)

[II.4 Assumptions, Dependencies and Constraints 5](#_Toc23320623)

[III. Specific Requirements 5](#_Toc23320624)

[III.1 External Interface Requirements 5](#_Toc23320625)

[III.1.1 User Interfaces 5](#_Toc23320626)

[III.1.2 Hardware Interfaces 5](#_Toc23320627)

[III.1.3 Software Interfaces 5](#_Toc23320628)

[III.1.4 Communication Interfaces 5](#_Toc23320629)

[III.2 Functional Requirements 5](#_Toc23320630)

[III.3 Performance Requirements 5](#_Toc23320631)

[III.4 Design Constraints 5](#_Toc23320632)

[III.4.1 Standards Compliance 5](#_Toc23320633)

[III.4.2 Hardware Limitations 5](#_Toc23320634)

[III.4.3 Any Other Constraint 5](#_Toc23320635)

[III.5 Software System Attributes 5](#_Toc23320636)

[III.5.1 Reliability 5](#_Toc23320637)

[III.5.2 Availability 5](#_Toc23320638)

[III.5.3 Security 5](#_Toc23320639)

[III.5.4 Maintainability 5](#_Toc23320640)

[III.5.5 Portability 5](#_Toc23320641)

[IV. Formal Analysis Using Alloy 5](#_Toc23320642)

[V. Effort Spent 5](#_Toc23320643)

[VI. References 5](#_Toc23320644)

1. Introduction

## Purpose

### General Purpose

Safestreet is a crowd-sourced application that intends to provide users with the possibility to notify authorities when traffic violations occur. In particular the application allow users to send picture of traffic violation with their geographical position and a description of the violation. The violation are send by the application to the authorities that take the right measures. Thanks to the system the work of the authorities is simplified because common user add valuable information such as:

* Geographical position
* Plate number (not mandatory)
* Type of violation
* Date and time

In addition the application permit, for all type of users, the mining of the information such as:

* Area in which there is currently a car accident
* Area with high traffic density

Authorities can also access to more detailed data such as:

* Number of violation committed by a car
* Area with high frequency of violations

### Goals

Goal 1: Users has the possibility to notify traffic violations to the authorities.

Goal 2: The application must register information about the violations and send them to the authorities.

Goal 3: Users and authorities to access to the information collected from all the users.

Goal 4: The system must allow access to different types of users (with different roles) and give them different levels of visibility.(It is the same of 3?)

Goal 5: The application allows communication with the municipality services.

If we choose ADVF1:

Goal 6: The application must elaborate the information provided by the municipality to identify unsafe areas and suggest interventions.

If we choose ADVF2:

Goal 6: The system must ensure safety in the chain of custody of the information coming from the users; they mustn’t be corrupted, manipulated or broken.

Goal 7: The application must elaborate the information provided by the municipality (about issued tickets) to build statistics.

## Scope

The safestreet service is offered to common users that wants to help the authorities reporting traffic violations (type of violation and location). It is thought for the authorities control all the violations that occurs and take actions, so they need to be supported by the user through the service, that stands in the middle.

A picture containing vector graphics

Description automatically generated  

The software to be (from now on S2B) give the user the possibility to send pictures of traffic violations, and other data such as date and time, car’s plate (not mandatory) and geographical position. The system will also allow the user to choose which data to send and which not. Safestreet offers, also, to the user to consult their reports and highlights on traffic and car accident. It is assumed that the user has a device with the capacity to acquire the necessary data (camera + GPS); if some of this device are not present, the data cannot be acquire. The authorities can access to all the reports and highlights on violation density logging in and querying the Safestreet platform (both users and authorities have to register to the system first). The system relies on the fact that all the users can be identified with an unique key (their fiscal code) and so the authorities can know who sent the violations. Authorities can request to the system to retrieve violations or ask for aggregate data on the base of reported violations. The aggregate requests are handled directly by Safestreet. **to be check and make additions…**

## Definitions, Acronyms, Abbreviations

1. Definitions

* **Users**: the “normal” customer of the application that exploits the application only to send traffic violations and to retrieve information from it.
* **Authorities**: the customer of the application that exploits it to monitor the violations and take adequate measures.
* **Customer:** general safestreet customer, can be an authority or a user
* **Report:** a module reporting a traffic violations with useful data

1. Acronyms

* API = Application Programming Interface
* GPS = Global Positioning System
* UI = User Interface
* S2B = Software To Be

1. Abbreviations

* Gn = nth goal
* Dn = nth domain assumption
* Rn = nth requirement

## Revision History

* Version 1.0:
  + First release

## Reference Documents

* IEEE std 830-1998 IEEE Recommended Practice for Software Requirements Specifications
* Specification document: Safestreet Mandatory project Assignment
* UML diagrams: <https://www.uml-diagrams.org/>
* Alloy doc: <http://alloy.lcs.mit.edu/alloy/documentation.html>

## Document Structure

The RASD document is composed of five chapters, as outlined below:

**Chapter 1** contains an introduction, that describes the purpose of the system informally. Furthermore there is a list of goals that the application has to reach. It is also defined the scope and the aim of the system in great detail. Moreover it show a clear report of the world and shared phenomena (**To do W&S…**)

**Chapter 2** (**To do…**)

**Chapter 3** (**To do…**)

**Chapter 4** (**To do…**)

**Chapter 5** (**To do…**)

1. Overall Description

## Product Perspective

## Product Functions

## User Characteristics

## Assumptions, Dependencies and Constraints

1. Specific Requirements

## External Interface Requirements

### User Interfaces

### Hardware Interfaces

### Software Interfaces

### Communication Interfaces

## Functional Requirements

## Performance Requirements

## Design Constraints

### Standards Compliance

### Hardware Limitations

### Any Other Constraint

## Software System Attributes

### Reliability

### Availability

### Security

### Maintainability

### Portability

1. Formal Analysis Using Alloy
2. Effort Spent
3. References