

S



**SAFESTREETS**

Requirement analysis and specification Document

S

edoardo putti

10 November 2019

Table of Contents

[**1.** **Introduction** 3](#_Toc23538903)

[1.1. Purpose 3](#_Toc23538904)

[1.2. Scope 3](#_Toc23538905)

[1.2.1. Goals 3](#_Toc23538906)

[1.3. Definitions, Acronyms, Abbreviations 3](#_Toc23538907)

[1.3.1. Definitions 3](#_Toc23538908)

[1.3.2. Acronyms 3](#_Toc23538909)

[1.3.3. Abbreviations 3](#_Toc23538910)

[1.4. Revision history 3](#_Toc23538911)

[1.5. Document Structure 3](#_Toc23538912)

[**2.** **Overall Description** 3](#_Toc23538913)

[2.1. Product perspective 3](#_Toc23538914)

[2.2. Product Functions 3](#_Toc23538915)

[2.2.1. Scenario 1 3](#_Toc23538916)

[2.2.2. Scenario 2 3](#_Toc23538917)

[2.2.3. Scenario 3 3](#_Toc23538918)

[2.2.4. Scenario 4 3](#_Toc23538919)

[2.2.5. Scenario 5 3](#_Toc23538920)

[2.3. User characteristics 3](#_Toc23538921)

[2.4. Assumptions, dependencies and constrains 3](#_Toc23538922)

[2.4.1. Domain Assumptions 3](#_Toc23538923)

[**3.** **Specific Requirements** 3](#_Toc23538924)

[3.1. External Interface Requirements 3](#_Toc23538925)

[3.1.1. User Interfaces 4](#_Toc23538926)

[3.1.2. Hardware Interfaces 4](#_Toc23538927)

[3.1.3. Software Interfaces 4](#_Toc23538928)

[3.1.4. Communication Interfaces 4](#_Toc23538929)

[3.2. UML modeling 4](#_Toc23538930)

[3.2.1. Use case diagrams 4](#_Toc23538931)

[3.2.2. Class diagrams 4](#_Toc23538932)

[3.2.3. Activity diagrams 4](#_Toc23538933)

[3.3. Functional Requirement 4](#_Toc23538934)

[3.4. Performance Requirement 4](#_Toc23538935)

[3.5. Design Constrains 4](#_Toc23538936)

[3.5.1. Standard Compliance 4](#_Toc23538937)

[3.5.2. Hardware limitations 4](#_Toc23538938)

[3.5.3. Any other constraint 4](#_Toc23538939)

[3.6. Software System Attributes 4](#_Toc23538940)

[3.6.1. Reliability 4](#_Toc23538941)

[3.6.2. Availability 4](#_Toc23538942)

[3.6.3. Security 4](#_Toc23538943)

[3.6.4. Maintainability 4](#_Toc23538944)

[3.6.5. Portability 4](#_Toc23538945)

[**4.** **Formal Analysis Using Alloy** 4](#_Toc23538946)

[4.1. Alloy Code 4](#_Toc23538947)

[4.2. Results of Alloy Analysis 4](#_Toc23538948)

[4.3. Alloy Model 5](#_Toc23538949)

[**5.** **Effort Spent** 5](#_Toc23538950)

1. **Introduction**
   1. Purpose

This document is the Requirements Analysis and specification Document (RASD) for the Safestreets application offers, about requirements and goals that the system must present. This document offers also an analysis of the world and shared phenomena regarding Safestreets. RASD contains class diagrams to show domain models and other diagrams which illustrates, with more details, transaction of the functionalities of the application.

* 1. Scope

Safestreets is a crowd-sourced application that allows users to send reports about traffic violations. A person become a user of Safestreets by registering himself into the application. After this phase, user can start to use the basic functionalities of the app (e.g. sending reports and retrieving traffic information). Safestreets allows, also, to authorities to register, log-in and use advance functionalities of the app.

The app allows users to send reports to the authorities and querying the app for traffic and violations highlights. The goal of Safestreets is to send the reports, made by the users, to the authorities helping and facilitating with their job.

When a user creates a new report, he can add a picture of the car plate so that the app can extrapolate the car plate id automatically, user can also add the geographical position of the violation and of course the type and timing of the traffic violation. Moreover users are allowed to visualize traffic highlights using the app.

The app allows authorities to querying all the reports and also to extrapolate useful information and statistic from the app.

The system interfaces with other firms (e.g. municipality services, territory maps companies) to offer a more comprehensive costumer experience.

* + 1. Goals

**[G1]:** Users should be able to send reports regarding traffic violations.

[G1]#1 specify traffic violation

[G1]#2 attach a picture of the car plate

[G1]#3 include the geographical position

**[G2]:** Authorities should be able to access to all the reports

**[G3]:** Costumers should be able to access to different kind of information depending on their role

[G3]#1 users can access only to traffic highlights, and their reports

[G3]#2 authorities can access to traffic highlights, violation statistics and all the reports

**[G4]:** The application allows communication with external services

* 1. Definitions, Acronyms, Abbreviations
     1. Definitions

Here is provided a list of definitions of words and expressions used in the document

* **Users:** the “normal” costumer 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 reports and take adequate measures.
* **Customers:** people that uses Safestreets, can be an authority or a common user
* **Report:** a module reporting a traffic violations containing useful data
* **Traffic violations:** violation of the laws that regulate vehicle operation on streets and highways.
  + 1. Acronyms
* API = Application Programming Interface
* GPS = Global Positioning System
* UI = User Interface
* S2B = Software To Be
  + 1. Abbreviations
* G*n = nth* goal
* D*n = nth* domain assumption
* R*n = nth* requirement
  1. Revision history
  2. Document Structure

After purpose and scope, used to briefly introduce the topic, are delineated the goals that the S2B should achieve coupled with a list of useful definitions and acronyms. Subsequentially, the text proceeds with an analysis of the functions that the app should provide. The analysis starts with a general exposition of the scenarios and becomes gradually more detailed passing through the analysis of the actors that will interact with the S2B and the statements of the domain assumptions. After that, the specific requirements are exposed focusing firstly on the external interfaces and then providing the models used to highlights the relations between the actors and S2B and describe the internal structure of the latter. After that, Functional and non-Functional requirements are sequentially discussed. Before ending with the effort spent the references is provided a formal analysis performed with alloy.

1. **Overall Description**
   1. Product perspective
   2. Product Functions
      1. Scenario 1
      2. Scenario 2
      3. Scenario 3
      4. Scenario 4
      5. Scenario 5
   3. User characteristics
   4. Assumptions, dependencies and constrains
      1. Domain Assumptions
2. **Specific Requirements**
   1. External Interface Requirements
      1. User Interfaces
      2. Hardware Interfaces
      3. Software Interfaces
      4. Communication Interfaces
   2. UML modeling
      1. Use case diagrams
      2. Class diagrams
      3. Activity diagrams
   3. Functional Requirement
   4. Performance Requirement
   5. Design Constrains
      1. Standard Compliance
      2. Hardware limitations
      3. Any other constraint
   6. Software System Attributes
      1. Reliability
      2. Availability
      3. Security
      4. Maintainability
      5. Portability
3. **Formal Analysis Using Alloy**
   1. Alloy Code
   2. Results of Alloy Analysis
   3. Alloy Model
4. **Effort Spent**