1. Introduction
   1. Purpose
      1. General Purpose

This document represents the Requirement Analysis and Specification Document (RASD). In this document we will explain SafeStreets. This will be done by a detailed presentation of the proposed solution and its purpose, listing its goals, and the requirements and assumptions through which they will be achieved.

SafeStreets is a public interface aimed to public-spirit citizens who want to help keeping the streets clear. This S2B intends to provide users with the possibility to notify authorities when traffic violations occur. This materializes using a platform through which users can upload pictures of streets violations, in particular parking violations.

There could also be other types of issues that a customer can report, for example speed violations, accidents, non respected traffic lights and give ways. They’re reported in different ways.

The S2B also has a map, based on Google Maps, on which some areas are highlighted with different colors according to the number and types of violations reported (for example first the user can choose the type of violation, then the map shows different areas with different colors: a red area means that a lot of the chosen violation have occurred, yellow is medium quantity, green one means very few).

Once the violation is sent, its data are stored in SafeStreets center and analyzed by the software, in order to retrieve information to update the map.

The application wants to offer to the users a way to help the municipality.

* + 1. Goals
  1. Scope
  2. Definitions, acronyms, abbreviations
     1. Definitions
     2. Acronyms
     3. Abbreviations
  3. Revision history
  4. Reference documents
  5. Document structure

1. Overall Description
   1. Product perspective
   2. Product functions
   3. User characteristics
   4. Assumptions, dependencies and constraints
2. Specific Requirements
   1. External interface requirements
      1. User interfaces
      2. Hardware interfaces
      3. Software interfaces
   2. Functional Requirements
   3. Performance Requirements
   4. Design constraints
   5. Software system attributes
      1. Reliability
      2. Availability
      3. Security
      4. Maintainability
      5. Compatibility
3. Formal analysis using Alloy
4. Effort spent