

Contents

1	Introduction	2
1.1	Purpose	2
1.2	Scope	2
1.2.1	Description of the given problem	2
1.2.2	Current system	3
1.2.3	Goals	3
2	Overall Description	4
2.1	Product perspective	4
2.2	User Characteristics	4
2.2.1	Actors	4
2.3	Constraints	5
2.3.1	Regulatory policies	5
2.3.2	Hardware limitations	5
2.3.3	Interface to other applications	5
2.4	Assumptions and dependencies	6
2.4.1	Text Assumptions	6
2.4.2	Domain assumptions	6

Introduction

1.1 Purpose

The purpose of this Requirement Analysis and Specification Document (RASD) is to provide a detailed description about the software SafeStreets that we wish to develop. In particular this document is focused on important aspects that are useful during the design of the software architecture like: scope, functional and non-functional requirements, use cases and scenarios, constraints and assumption, class/sequence diagrams, limitation and interfaces with other softwares. Overall the document is a useful guide for the developers that will have to follow and implement all the necessary requirements, nevertheless it's also a document that can be given to potential customers to get them an idea of what the software will be like.

1.2 Scope

1.2.1 Description of the given problem

SafeStreets is a service that aims to improve the safety of the streets via the help of the users. They can notify violations or any illegal behaviour related to driving to authorities. In particular users can interact with the service via an application that can be used to send the violation reports; the latters mainly consist in a picture, taken by the user, of the vehicle responsible of the violation, moreover users can send along with the picture location, date and additional information related to the infringement. The system also provides a Web interface that can be used by the authorities in order to check the violations recieved. It's important to note that the picture sent by the user must contain the car plate in order to let the police officers know which is the real vehicle that committed the violation. Another functionality of the application offers to both users and authorities the

possibility to see areas/streets with the highest violations rates thanks to the data collected over time. As an advanced functionality the application can interact with services offered by the municipality; in particular if a service offers data related to accidents SafeStreets can cross this data with its own data in order to get a better idea of the potentially unsafe areas and therefore suggest some possible interventions. Ultimately the application will have to be scalable and easy to use in order to provide a fast and efficient utilization for users that see a violation and want to immediately report it.

1.2.2 Current system

SafeStreets is a new service that it's entering the market right now, so all the architecture will have to be built from scratch. There aren't any legacy systems that need to be integrated into the application with the exception of the third party services that offer some functionality required by the application (like Maps and plate recognition), those services will be better explained in the following paragraphs.

1.2.3 Goals

- G1 : Allow a person to become a registered User after submitting his credentials for the registration to the service
- G2 : Allow the user to send a violation report consisting in a picture and some metadata
- G3 : Allow the user to watch the history of his reports and their status
- G4 : Allow the user to watch on a map areas and streets with a high number of violations
 - G4.1 : The user gets the latest violations near its current location
 - G4.2 : The user gets the latest violations near the location that he specifies

Overall Description

2.1 Product perspective

TO BE FINISHED

The application is connected to a backend service that saves and handles all the notifications and interacts with some third party services used by the system.

The system offers the possibility for both users and authorities

2.2 User Characteristics

2.2.1 Actors

- *Visitor*: a person without a SafeStreet account. Visitors can only have access to the homepage and the registration form of the mobile application;
- *User/ Mobile user*: a person correctly registered to the SafeStreet account service. Users/Mobile users can perform any of the actions made available by the SafeStreet mobile application;
- *Recognized authority*: a recognized authority (Police station/ municipality) which submitted to SafeStreet and can interact with it through its web application interface;
- *Local system administrator/Police corporal*: a person which belongs to a recognized authority, in charge of dealing police technician accounts and scheduling patrols;
- *Police technician*: a policeman encharged of dealing with the violations report. He/She patrols the unsafe areas and gives fines to the reported cars in case of violations;

- *Third party recognition service*: an image recognition service which allows SafeStreet to extract car plates from the violation report's images.

2.3 Constraints

2.3.1 Regulatory policies

The user information is stored accordingly to the GDPR policies in order to guarantee the privacy of individuals.

- No data is shared with third parties for commercial purposes;
- All reported images and violation data are stored safely through encryption methods. The third party system used for image recognition can not store any information/image which identifies;
- Any additional information, such as GPS position or Camera access, is promptly asked to the user before performing the specific operation, accordingly to the Android/IOS standards.

2.3.2 Hardware limitations

- Mobile applications:
 - Any kind of modern smartphone;
 - Internet connectivity;
 - Camera;
 - GPS.
- Web application:
 - Browser (any HTTP client);
 - Internet connectivity.

2.3.3 Interface to other applications

A certified external image recognition service deals with the car plate recognition through report's images.

Report's information are stored in a cloud database which guarantees data encryption and information retrieving through authentication.

2.4 Assumptions and dependencies

As far as the specification document is concerned, it is necessary to specify some details and to state clearly a few ambiguous points. In order to better clarify those situations the following assumptions are introduced.

2.4.1 Text Assumptions

2.4.2 Domain assumptions