POLITECNICO
MILANO 1863

SafeStreets

Requirements Analysis and Specification Document

Davide Cocco - 944122
Marco Gasperini - 944922

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

Contents

1 INTRODUCTION

1.1 Purpose

This document will contain the `Requirements Analysis and Specification Document` for SafeStreets, a crowd-sourced application that will allow users to notify traffic violations to authorities, and authorities themselves to assist them in enforcing laws for these offences through a data mining system. The software will not be designed to substitute the existing system, but rather to improve its efficiency.

1.2 Scope

The software will be dedicated to civilians and authorities: through a mobile application, civilians will have the possibility to report a violation by submitting pictures, position, date and the license plate of the offender to the authorities in an effort to improve their communities, while officers can use the data mining system, which updates a map periodically with new reports highlights areas with high violation frequency, to facilitate the process of law enforcement by locating offences and releasing tickets. Authorities will be supplied with a web application with the ability to add new officers, and will also have access to the aforementioned system. A data warehouse will be thus implemented, and repeated reports of violations could also lead to automated traffic tickets being released at the discretion of the authorities. A two factor authentication will be used to verify the officers' identity, and a unique code will be required for the releasing of tickets.

1.2.1 Goals

- Allow future users to easily register and login.
- Allow users to notify authorities of traffic violations through the use of the camera.
- Store relevant info about the violation in the data warehouse.
- Allow only authorities to visualize relevant data for law enforcement purposes.
- Assist authorities in the process of law enforcement:
  - implement togglable functionality for automated traffic ticket compilation and sending;

  - the officer's client GUI must show reports in real time and high violation frequency areas;

- Guarantee security:
  - allow discarding of invalid reports;
  - avoid unauthentic officer registrations by design;
  - block people who came in possession of an officer's device to cause any damage;
  - only allow authorities to visualize processed data helpful for law enforcement;
  - ban users who abuse of the violation reporting system.
- Guarantee privacy to both civilians and authorities.

1.3 Definitions, Acronyms, Abbreviations

1.3.1 Definitions

- `Violation | Offence:` we will strictly refer to any kind of static traffic violation, especially parking violations.
- `Authority:` a law enforcement authority which is manages traffic violations, it could be the local police, the traffic wardens etc. We will refer with this term also to the personnel which operates the web application in the authorities' headquarters.

1.3.2 Acronyms

- `HFVZ:` high violation frequency zone.
- `GPS:` global positioning system.

1.3.3 Abbreviations

- [Gn]: n-goal.
- [Rn]: n-requirment.
- [Dn]: n-domain assumption
- App: application.

1.4 Revision history

1.5 Reference Documents

- Specification Document ”Mandatory Project Assignment A.Y. 2019/2020”
- 830-1993 - IEEE Recommended Practice for Software Requirements
- ”Appunti di Sistemi Informativi per il Settore dell’Informazione” A.Y. 2018/2019
- Alloy Language Reference ”http://alloytools.org/download/alloy-language-reference.pdf”
- UML diagrams: https://www.uml-Âdiagrams.org/

1.6 Document Structure

1. `Introduction`: the first section is a general description of the system’s scope and its goals. It also includes the revision history of the document and its references. Definitions and abbreviations used along the paper will be provided.
2. `Overall Description`: this section includes shared phenomena, requirements and domain assumptions. It also clarifies users’ needs.
3. `Specific Requirements`: this section includes all the requirements, both functional and non functional.
4. `Formal Analysis Using Alloy`: it includes the Alloy model of the described system.
5. `Effort Spent`: this section includes information about the hours spent to compile this document.
6. `References`: this section includes 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

The goals to be accomplished require the following functions:

- `Registration and login management:` while civilians will be able to register and login in standard fashion through email and password, officers will be added by authorities through their client: an automatically generated password will be sent to their email, and they will be able to access through their badge ID and such password. This way it is impossible for civilians to log in as an officer, as they would have to know both the badge number and the password contained in the officer's personal email.

- `Violation reporting:` allow users (both civilians and officers) to choose between a set of offences and send various pictures representing it, with the corresponding date and location (via GPS and system timestamping) to the authority which will process them and discard incoherent or wrong ones. Photos can only be taken in real time and not uploaded to avoid manipulations.

- `Ban requesting and issuing:` if authorities receive undescriptive photos or in any shape or form useless reports, they can send a request from the authorities' GUI to the SafeStreets HQ's server to ban the abusing user.

- `Data storage, mining and visualization:` store all the relevent data from the reporting procedure to process it through ETL and present it to the authority to assist them in law enforcement. The most important functionality will be periodically updating the map in the officers' and authorities' GUIs with new violations and hightlightings of HVFZs, and also listing repetead offenders.

- `Automated ticket emission:` officers can locate violations through the map, and through a unique code sent to the authority, sign an automatically compilated ticket which will be emitted and sent via email to the Vehicle Licensing Agency. The code is to make sure that even if the officer's mobile is stolen while he's logged in, the thief cannot emit tickets.

2.3 User characteristics

- `Visitors:` users who haven't yet registered or logged in, they are only able to register as a civilian or log as a civilian (through email and password) or as an officer (through Badge ID and password issued by the authority).

- `Civilians:` users unaffiliated to law enforcement authorities who are interested in improving their community, they are only able to report violations but don't have access to mined data (except the updated map with HVFZs). They're required to login so they can eventually be excluded in case of security violations.

- `Officers:` they're registered through their authority listing by their badge ID them through the web application and receiving a password. They're able to see the mined data, report violations, and sign the automated tickets.

- `Users:` we will use this name only as a general way to refer to both registered officers and civilians when we're describing actions they are both able to perform in the same way (like compiling violation reports).

- `Authority:` they're manually registered by SafeStreets personnel, and their work is managed in office through an officer using a web application. They can add new officers or confirm self-registered ones, and they're able to see the processed data, enable automatic ticket emissions, receive violation reports and thus discriminate between valid and invalid ones. They can request bans which the SafeStreets server will take care of issuing and can see every officer connected to the application.

2.4 Domain assumptions

- [**D1**]: The officer's Badge ID is assumed to be unique.

- [**D2**]: Users are assumed to provide a valid email.

- [**D3**]: At least one of the city's authorities' server is always online to process reports.

- [**D4**]: Users' devices support the Mobile Application.
- [**D5**]: Users' devices are connected to the Internet, their camera and GPS work correctly.
- [**D6**]: Email from the authority containing the officer's password is always received correctly.
- [**D7**]: Authorities' devices support the web application.
- [**D8**]: Authorities correctly receive the great majority of reports.
- [**D9**]: The web application at the authorities' location is operated by an assigned officer most of the time.
- [**D10**]: Officers not on another duty tend to verify the validity of reports and sign tickets.
- [**D11**]: Officers only sign correct tickets.
- [**D12**]: The users compile reports correctly most of the times (photo and type of offense).
- [**D13**]: Timestamps and location are compiled correctly by the device with very good accuracy.
- [**D14**]: Only the officer knows his personal pin that allows him to proceed in ticket compilation.

3 SPECIFIC REQUIREMENTS: Here we include more details on all aspects in Section 2 if they can be useful for the development team

3.1 External Interface Requirements

3.1.1 User Interfaces

3.1.2 Hardware Interfaces

3.1.3 Software Interfaces

3.1.4 Communication Interfaces

3.2 Scenarios

This section describes some of the scenarios in which SafeStreets may be used. Scenarios omit steps already described in previous ones:

3.2.1 User registration

Giancarlo saw the advertisement of SafeStreets and decided to download the mobile application to report a car that always parks on the sidewalk outside his house and blocks the passage of pedestrians. After downloading and launching the new app, he is asked to fill a registration form in which he must insert his email address and a password. After clicking the ”Sign In” button he receives an email with a link to confirm his registration, which he clicks to finalize his registration. From now on he can login and report offences with the ”Report” button in the main screen.

3.2.2 Correct violation reporting

Mary just logged in as a civilian to report a double-parked car. Since it’s her first use of the application, she’s asked to allow SafeStreets to access her GPS and camera, to which she complies. She clicks ”Report” on his main screen and chooses a violation from the modal which pops up. The app then opens her camera asking to take a photo of the vehicle showing the offence being committed. Then she’s asked to frame the license plate with her camera, and the supervised learning algorithm scans the various frames looking for an interpretation for the image. The app then associates the report with a timestamp and a GPS position, and sends it to one of the authorities' servers. After a couple of minutes, Mary receives a notification in the app's notification section informing her her report was correctly accepted and processed.

3.2.3 Wrong violation reporting

Simon decides to use the app to report his neighbour's vehicle because he doesn't like him. He sends a photo which in fact doesn't show any violation being committed. Arthur, a policeman who's working with the SafeStreets web application at his barracks, receives the report and notices there's clearly no violation being committed in the photo, and clicks on the "Warn" button which alerts the SafeStreets server of an invalid report, and discards the report immediately after. After a couple of minutes, Simon receives a notification telling him he's been forbidden from using the reporting function for a month.

3.2.4 Officer registration

Billy is a policeman whose barrack has just installed the SafeStreets application and data server. He received an email notifying him that he may be enabled to utilize the service. After a while, he receives an email from his barrack's SafeStreet client containing an automatically generated code. By entering his badge ID and this code in the officer login section of the mobile application, Billy is now able to login when he desires by using the code as password. Once logged in, Billy will have to enter a 5-digit pin that will allow him to make a traffic ticket signed by him when he receives a report. Bill now sees the map of his surroundings, periodically updated with new traffic offence reports and with HFVZ highlighted.

3.2.5 Correct Ticket Emission

John was on patrol this morning and checked on the SafeStreets App if there was any report in his vicinity. As soon as he logged in he noticed a reported violation in the map, about 250 meters from his position. John clicks the "Accept" button on the tooltip to confirm to other active officers he's checking the authenticity of the violation, and avoiding any other one of replicating this task. He now sees the photos of the reporting and the type of offence. Once he's near the vehicle, a "Confirm" button pops up. John checks that the vehicle is effectively committing a violation and the license plate matches the one in his screen. Once John clicks it and enters his secret 5 digit pin his authority's server will be able to place his signature into an automatically generated ticket that will be sent to the Vehicle Licensing

Authority. Now the violation tooltip can disappear and John goes on with his duties.

3.2.6 Wrong report discarded by an officer

Gabriele, who is an officer, is walking towards an offence reported from SafeStreets. As soon as he arrives there, he realizes that the report sent involved a car parked on the pedestrian crossing which is no longer there. So he decides click on the "Delete" button which subsequently communicates to the authorities' server to remove it from the map.

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

3.4 Use cases

3.4.1 Civilian Visitor Registration

NAME	Civilian registration
ACTOR	Visitor
GOALS	[G1]
ENTRY CONDITIONS	The Visitor has installed the Application on his/her Mobile device
EVENTS FLOW	1. Fill all mandatory fields in the Civilian Registration Form 2. The visitor clicks on "Register" button 3. The system checks data has not been used already 4. The system sends an email as confirmation 5. The system saves Users' data
EXIT CONDITIONS	The Visitor has clicked in the email's link
EXCEPTIONS	1. The email is already registered 2. Some mandatory fields are not filled

3.4.2 Civilian Login

NAME	Civilian Login
ACTOR	Civilian
GOALS	[G1]
ENTRY CONDITIONS	The civilian is in the Civilian Login page
EVENTS FLOW	1. The civilian enters his email 2. The civilian enters the password 3. The civilian clicks on ”Login” button 4. The system checks the correctness of the credentials
EXIT CONDITIONS	The civilian has successfully logged in
EXCEPTIONS	1. The civilian is not registered 2. The email is wrong 3. The password is wrong 4. Some mandatory fields are not filled

3.4.3 Officer Registration

NAME	Register to SafeStreets
ACTOR	Authority
GOALS	[G1]
ENTRY CONDITIONS	1.The authority has installed the web application 2. The authority has a correctly configured server
EVENTS FLOW	1. Launch the web application 2. The authority personnel clicks on the ”Register new officer” button 3. The authority adds the officer’s badge ID and email 4. The server checks the data has not been used already 5. The server sends an email with an automatically generated password to the officer 6. The system saves the officer’s data
EXIT CONDITIONS	The email has successfully been sent to the officer
EXCEPTIONS	1. The badge is already registered 2. The email is already registered 3. Some mandatory fields are not filled

3.4.4 Officer Login

NAME	Officer Login
ACTOR	Officer
GOALS	[G1] , [G.6.2]
ENTRY CONDITIONS	1. The officer has downloaded the mobile application 2.The officer has seen the password in his email 3.The officer is in the Officer Login page
EVENTS FLOW	1. The officer enters his badge ID 2. The officer enters his password 3. The officer clicks on "Login" button 4. The system checks the correctness of the credentials 5. The officer chooses a 5 digit pin to emit tickets
EXIT CONDITIONS	The officer has successfully logged in
EXCEPTIONS	1. The officer is not registered 2. The badge ID is wrong 3. The password is wrong 4. Some mandatory fields are not filled

3.4.5 Violation reporting

NAME	Violation reporting
ACTOR	User
GOALS	[G2] ,[G5], [G.6.1]
ENTRY CONDITIONS	1.The user is logged in 2. The user has allowed the app to use the camera and GPS
EVENTS FLOW	1. The user clicks on "Report" 2. The user chooses a violation from the dropdown menu 3. The user takes a pic of the vehicle and the violation being committed
	4. The user frames the license plate 5. The system computes the plate number 6. The system adds date and timestamp and sends the report to the authority
EXIT CONDITIONS	The report has successfully been sent
EXCEPTIONS	1. The plate can't be recognised 2. GPS is turned off 3. Camera can't be used for any reason

3.4.6 Ticket confirmation

NAME	Ticket confirmation
ACTOR	Officer
GOALS	[G2] ,[G5], [G.6.3]
ENTRY CONDITIONS	1. The Officer is logged in 2. The officer is near a reported violation 3. The plate number matches the one contained in the report
EVENTS FLOW	1. The officer clicks on the "Confirm" button 2. The officer inputs his pin 3. The system checks the pin's correctness 4. The system sends a notice to the authority 5. The system deletes the report from the map
EXIT CONDITIONS	The confirmation has successfully been sent
EXCEPTIONS	The pin is wrong

3.4.7 Report validation

NAME	Report invalidation
ACTOR	Authority
GOALS	[G2] ,[G3], [G.6.1]
ENTRY CONDITIONS	1. The Authority is in the main menu 2. The authority received a violation report
EVENTS FLOW	1. The authority checks the photo correctness 2. The authority clicks "Validate" 3. The system adds the report to the data warehouse 4. The system adds the report to the map 5. The server sends notifications to active officers.
EXIT CONDITIONS	The report has been validated
EXCEPTIONS	None

3.4.8 Report invalidation with warning

NAME	Report invalidation
ACTOR	Authority
GOALS	[G2] ,[G3], [G.6.1]
ENTRY CONDITIONS	1. The Authority is in the main menu 2. The authority received a violation report
EVENTS FLOW	1. The authority checks the photo correctness 2. The authority clicks ”Validate ” 3. The authority clicks on ”Warn” 4. The server sends a ban request to the SafeStreets server
EXIT CONDITIONS	The report has been invalidated
EXCEPTIONS	None

3.5 Performance Requirements

3.6 Design Constraints

3.6.1 Standards compliance

3.6.2 Hardware limitations

3.6.3 Any other constraint

3.7 Software System Attributes

3.7.1 Reliability

The system should always be able to process reports and store them in the data warehouse, but according to G1 the priority is assisting law enforcers and not substituting the existing procedures, thus transient and occasional deviations can be accepted. Reports can be processed by multiple authorities' servers if the previous one fail (e.g. every police barrack has a server), so replication will be used to guarantee both availability and reliability.

3.7.2 Availability

The system must fullfil all the registration, login requests and reports whenever needed, only a small percentage of requests missed (around 0.1%) can be tolerated.

3.7.3 Security

Officers information is stored in the authorities' data server and not in SafeStreets', which only hosts civilian accounts. Through the badge-password authentication and the fact that only authorities can add officers, it will be impossible for civilians to log in as officers, and even if in some way someone who is not an officer is able to access to an officer account, he will not be able to do any substantial harm because of the pin required to sign tickets. A timeout will also automatically log the officer out. Authorities can see in real time every active officer through their client, so they also may be able to notice inconsistencies. Invalid reports will be discarded at the authority's client, and users will not be allowed to upload images to make sure every report photo is taken in real time and not manipulated. Similarly, the license plate will be scanned through an accurate supervised learning algorith run on the client so that it can not be input by a fraudolent user: officers will have the duty to later check the correctness of the plate number when signing the ticket. The system will also guarantee to not spread any kind of data to third parties to accomplish goal G7.

3.7.4 Maintainability

Best practices and design patterns of Object Oriented Programming along with documentation upkeep will be thoroughly followed to make sure the application is easily customizable and modifiable by SafeStreets' developers.

3.7.5 Scalability

The architecture must be simply scalable as the number of users grows over time. Except civilian account management, which is handled by SafeStreets' server and should not require too much complexity, functions are mostly provided by the authorities' servers which grow in number with the population of the considered city (e.g. the bigger the city the more numerous the barracks, and therefore the number of servers), thus the architecture will be scalable by design. In any case, the number of reports per day should never exceed a few thousands even in heavily populated cities.

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

- Davide Cocco

Day	Subject
15/10/2019	Purpose, Scope, Goals, Product functions, User charateristics
18/10/2019	Domain assumpions, Scenarios, Software attributes, Functional requirement
Total	

- Marco Gasperini

Day	Subject
15/10/2019	Purpose, Scope, Goals, Product functions, User charateristics
18/10/2019	Domain assumpions, Scenarios, Software attributes, Functional requirement
Total	

6 REFERENCES