



SafeStreets

Design Document

Giancarlo Danese - 945265

Davide Savoldelli - 928676

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

Contents

1	INTRODUCTION	3
1.1	Purpose	3
1.2	Scope	3
1.3	Definitions, Acronyms, Abbreviations	3
1.3.1	Acronyms	4
1.4	Revision History	4
1.5	Reference Documents	4
1.6	Document Structure	4
2	ARCHITECTURAL DESIGN	4
2.1	Overview: High-level components and their interaction	4
2.2	Component view	6
2.3	Deployment view	6
2.4	Runtime view: You can use sequence diagrams to describe the way components interact to accomplish specific tasks typically related to your use cases	6
2.5	Component interfaces	6
2.6	Selected architectural styles and patterns: Please explain which styles/patterns you used, why, and how	6
2.7	Other design decisions	6
3	USER INTERFACE DESIGN: Provide an overview on how the user interface(s) of your system will look like; if you have included this part in the RASD, you can simply refer to what you have already done, possibly, providing here some extensions if applicable.	6
4	REQUIREMENTS TRACEABILITY: Explain how the requirements you have defined in the RASD map to the design elements that you have defined in this document.	6
5	IMPLEMENTATION, INTEGRATION AND TEST PLAN: Identify here the order in which you plan to implement the subcomponents of your system and the order in which you plan to integrate such subcomponents and test the integration.	6
6	EFFORT SPENT	6
7	REFERENCES	6

1 INTRODUCTION

1.1 Purpose

This document represents the Design Document (DD) for SafeStreets software and contains a functional description of the system. Since we provide an overall guide of the systems' architecture, it's addressed to the software development team

1.2 Scope

SafeStreets will have an embedded algorithm which will analyze pictures of the vehicle plates sent by the user in order to recognize the vehicle. This information, together with the position of the vehicle and the type of violation that has been committed, will be stored in the software's database.

Authorities will have the chance to mine the information retrieved in the database by highlighting the streets/areas in which most of the violations are committed, the type of vehicles which commit most of the violations and which type of violations occur the most, suggesting possible interventions that could be taken.

1.3 Definitions, Acronyms, Abbreviations

- **User Device:** any compatible device with the SafeStreets application, like a smartphone or a computer
- **Personal Information:** information provided by the user during the registration process. It includes name, surname, birth date, address, e-mail address, mobile number.
- **Violation Report:** the act in which users can denounce violations on the streets, by providing the system its position, a photo and by selecting a violation from a precompiled menu
- **Mobile App:** an application that can be run by mobile devices, both smartphones and smartwatches.
- **Violations Map:** a map, accessible only by authorities, which contains notifications and alerts about all the unsafe areas where several violations are committed

1.3.1 Acronyms

- RASD: Requirements Analysis and Specification Document.
- DD: Design Document
- API: Application Programming Interface.
- GPS: Global Positioning System.
- PRA: Pubblico Registro Automobilistico
- AUC: Authority Unique Code
- DBP: Device-bound PIN

1.4 Revision History

1.5 Reference Documents

1.6 Document Structure

2 ARCHITECTURAL DESIGN

2.1 Overview: High-level components and their interaction

Our system can be summarized into 3 logical layers:

- **Presentation Layer**
This layer is divided into the Client tier (Mobile App for users) and the Web tier (Web App for authorities). From here, users and authorities can access the softwares' functionality
- **Application Layer**
Users and authorities logged into the software can exploit all the system's functionalities and are directly linked to a Demilitarized Zone (DMZ), through which it communicates with the Application Server
- **Data Layer**
Finally, here we have the Database Server and the Database itself, where all the data about the users, the authorities and the violations reported are stored and managed

A third-party ERP solution is how the System Administration is implemented, therefore we cannot provide information about its implementation

- 2.2 Component view
- 2.3 Deployment view
- 2.4 Runtime view: You can use sequence diagrams to describe the way components interact to accomplish specific tasks typically related to your use cases
- 2.5 Component interfaces
- 2.6 Selected architectural styles and patterns: Please explain which styles/patterns you used, why, and how
- 2.7 Other design decisions
- 3 **USER INTERFACE DESIGN:** Provide an overview on how the user interface(s) of your system will look like; if you have included this part in the RASD, you can simply refer to what you have already done, possibly, providing here some extensions if applicable.
- 4 **REQUIREMENTS TRACEABILITY:** Explain how the requirements you have defined in the RASD map to the design elements that you have defined in this document.
- 5 **IMPLEMENTATION, INTEGRATION AND TEST PLAN:** Identify here the order in which you plan to implement the subcomponents of your system and the order in which you plan to integrate such subcomponents and test the integration.
- 6 **EFFORT SPENT**
- 7 **REFERENCES**