Chapter 1

Requirements Traceability

R_1 At sign up, *User* must provide email and password.

R_2 At sign up, *User* must accept *Terms and conditions*, including the *Privacy statement.*

- **Authentication Manager:** handles the registration process for each *User.*

R_3 Identify a *User* by his/her *User identifier.*

R_{13} Query SafeStreets Database for a *User* by his/her *User identifier.*

R_{29} Identify *Municipality* by its *Reference code.*

- **Authentication Manager:** allows Registered Entities to be identified by their identifier and password that are stored in Authentication Database through Authentication DAO.

R_4 Receive *User picture.*

R_5 Receive *User* choice for the type of *Traffic violation.*

R_6 Receive *User position.*

R_9 Create *User report.*

R_{10} Store *User report* in SafeStreets Database.

R_{14} Retrieve specific *User reports* by querying SafeStreets Database.

R_{20} Allow *Users* to request all their *User report* and the related *Ticket feedback* stored in SafeStreets Database at any time.

R_{21} Send to a specific *User* his/her stored *User reports.*

R_{22} Send to a specific *User* the *Ticket feedback* related to his/her *User report.*

- **User reports Manager:** contains all the business logic in order to send and receive *User report* data by Mobile App Interface, loading and storing them in Reports Database through Reports DAO.

R_7 Send *User picture* to the *License plate recognition service.*

R_8 Receive *Recognized license plate* from the *License plate recognition service.*

R_{26} Allow *User* to re-take the *User picture.*

- **User reports Manager:** manages the transfer of *User picture* to the third party service responsible of recognizing the License plate number and sends it back to the client.

R_{12} Store *Ticket feedback* in SafeStreets Database.

R_{25} Receive *Ticket feedback* from *Municipality.*

- **User reports Manager:** handles the *Ticket Feedback* issuing confirmation sent by the *Municipality* and updates its status in Reports Database through Reports DAO.

R_{11} Send *User report notification* to *Municipality.*

- **Notification Manager:** for each new violation reported by *Users*, it is responsible of sending a notification to the concerned *Municipality.*

R_{15} Validate the data which constitutes the *User report.*

- **Authentication Manager:** guarantees a chain of custody matching *User reports*' data and *Users' identifiers* in a digital signature.

R_{16} Generate *Possible interventions* crossing *Municipality Accidents* data with SafeStreets Database data.

R_{17} Store generated *Possible interventions* for the *Municipality.*

- **User report Manager:** allows specific *User reports'* information retrieval based on report position and timestamp.
- **Accidents Manager:** periodically, it retrieves latest accidents data from *Municipality's* database, stores them in Accidents database and collects accidents basing on particular parameters.
- **Interventions Manager:** contains the logic able to cross *User reports'* and *Accidents'* data in order to generate *Possible interventions* to suggest to *Municipality.*

R_{18} Allow *Municipality* to set specific constraints to define *Detailed statistics.*

R_{19} Send *Detailed statistics* to the requesting *Municipality.*

R_{24} Send *Public statistics* to the requesting *User.*

R_{23} Allow *User* to choose among filters which define *Public statistics*.

- **Statistics Manager:** contains all the business logic in order to send *Public* and *Detailed Statistics* based on particular filters.

R_{27} Request *Accidents* data to *Municipality*.

R_{28} Receive *Accidents* data from *Municipality*.

- **Accidents Manager:** contains all the business logic in order to send and receive *Accidents* data from *Municipality* IT, loading and storing them in Accidents Database through Accidents DAO.

In the provided mapping the ”Router” component is not mentioned for simplicity reasons, but clearly it is directly or indirectly connected to the fulfillment of the majority of the system functionalities, as it routes to the right component every message coming from the Web Application side.