Politecnico di Milano
AA 2018-2019

POLITECNICO
MILANO 1863

Computer Science and Engineering
Software engineering 2

TrackMe RASD

Requirements Analysis and Specification Document
Version 1.0
24/10/18

Diego Piccinotti, Umberto Pietroni, Loris Rossi

Table of Contents

1 Introduction

1.1 Purpose

The purpose of this document is to analyze and illustrate specifications for the TrackMe system. This will be accomplished through a detailed analysis of the solution proposed and of the goals, assumptions and requirements that are necessary to realize it.
This document is intended to be used by the clients requiring the development of the system, its end users and all those who are involved in the development process, mainly project managers, analysts and development teams.

Data4Help is designed as a platform to enable third parties to access its users' health and location data. The system continuously collects these data through wearable devices and stores them to enable later access and/or aggregated statistics analysis.
Third parties can request data of some specific individuals, whose collection has to be approved by the user (which grants access to their data from this moment on), or aggregated statistical data on a group of user which must be larger than 1000 individuals to allow for proper anonymization of the data.
Data4Help is then extended in its functionality by **AutomatedSOS**, an additional service built on top of the existing data collection platform, which monitors elderly users' health parameters and automatically requests an ambulance to their location if they exceed a risk threshold determined by a preventive medical checkup.
Finally, Data4Help platform can be used also in run competitions to collect runners' statistics and enable their registration to the runs through another additional service, **Track4Run**, which also enables the run organizers to manage runs' participants and broadcast the live coverage to spectators through a map showing the live position of the runners.

1.2 Scope

In this section we describe the boundaries between the environment and the system to be.
In particular, we distinguish between world, shared and machine phenomena, which we further discuss in paragraph 2.1.
Moreover, we define the goals of the system in order to satisfy the stakeholders' needs.

1.2.1 World, shared and machine phenomena

World phenomena are the ones which the machine cannot observe.
In our context, world phenomena are, for instance:
- Movements of a user from a position to another one;
- The change of a user's health status
- An organizer plans a new run to be held

Phenomena entirely occurring in the machine are considered machine phenomena.
For instance, data manipulation from the database or checking for the proper anonymization of a group of data shall be done by the system to be.

Regarding shared phenomena, which involve both the environment and the system, we have events like:
- A third part requests for a specific user's data
- A user signs up in the system
- The system asks for an ambulance dispatch to the external provider

Further shared phenomena are discussed in the Product Perspective paragraph (2.1).

1.2.2 Goals

G1: The user can be recognized by providing a form of identification
G2: Allow third parties to monitor data about location and health status of individuals.
G3: Allow third parties to access data relative to specific users
G4: Allow third parties to access anonymized data of groups of users
G5: Allow third parties to offer a personalized and non-intrusive SOS service to elderly people so that an ambulance arrives to the location of the customer in case of emergency.
G6: Allow athletes to enroll in a run
G7: Allow organizers to manage runs
G8: Allow spectators to see on a map the position of all runners during the run

1.3 Definitions, acronyms, abbreviations

1.3.1 Definitions

- **User**: individual who accepts to give his location and health status data to Data4Help
- **Third party**: individual or organization registered to Data4Help which can request users' data
- **Data collection**: gathering of users' data through a wearable device

- **Anonymized data**: data about 1000 or more users whose personal information has been previously removed so that they are not directly relatable to the system's users.
- **Risk threshold**: Set of boundary health parameters defined for each AutomatedSOS user. If monitored values of the user's health parameters exceed these boundaries, an ambulance is dispatched to the user's location.
- **Athlete**: user who participates to a run.
- **Run organizer**: third party who can arrange a new run for TrackMe users.
- **Spectator**: public individual who follows a run through a map with runners' positions

1.3.2 Acronyms

- **API**: Application Programming Interface
- **GPS**: Global Positioning System

1.3.3 Abbreviations

- **Gn**: n-goal
- **Dn**: n-domain assumption
- **Rn**: n-functional requirement
- **Rn-NF**: n-non functional requirement

1.4 Reference Documents

- Specifications document: "Mandatory Project Assignment AY 2018-2019"
- IEEE Standard on Requirement Engineering (*ISO/IEC/IEEE 29148, Dec 2011*)

1.5 Document Structure

(to be written when the document is almost complete)

2 Overall Description

2.1 Product perspective

here we include further details on the shared phenomena and a domain model (class diagrams and state diagrams)

Describes external interfaces: system, user, hardware, software; also operations and site adaptation, and hardware constraints

2.1.1 Phenomena

World: (*world phenomena the machine cannot observe)*

- Users move (their position changes)
- Users' health parameters vary
- Organizers plan runs
- The external provider dispatches an ambulance

Shared: (*controlled by the world and observed by the machine or controlled by the machine and observed by the world)*

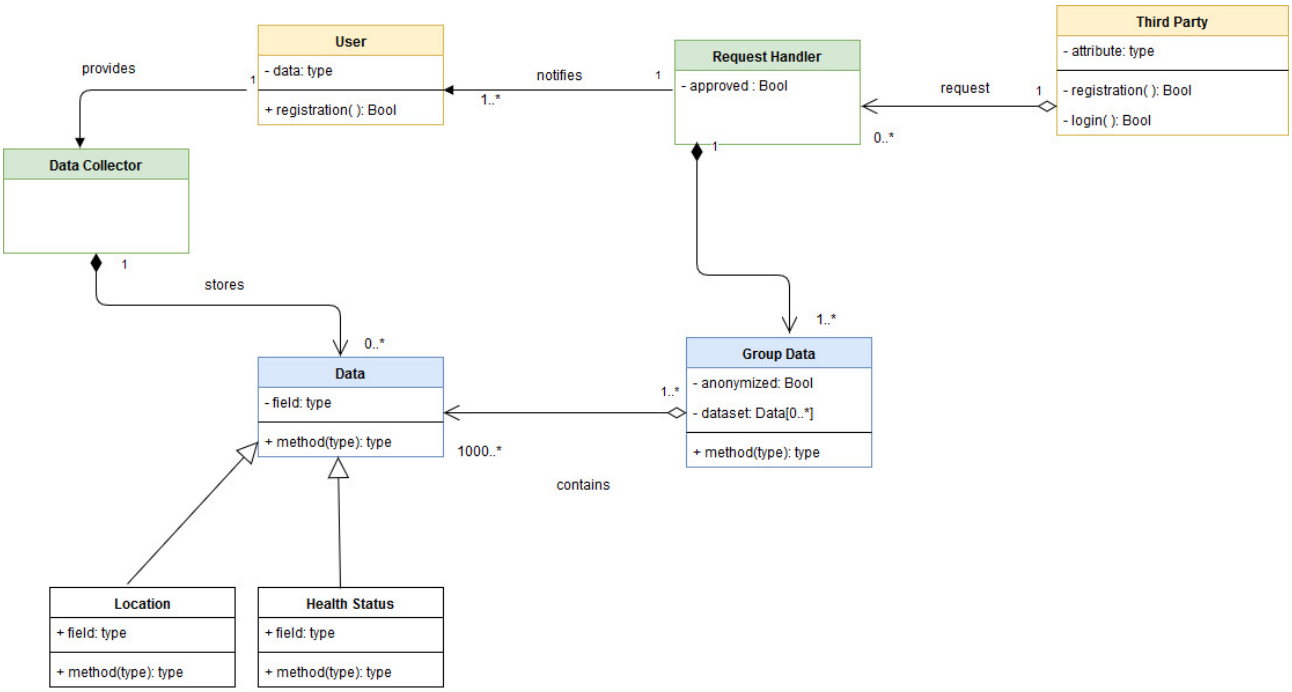
- User and third parties can sign up and login to the system
- User data collection by the system: the data is collected from the sensor but is "inside" the user, thus the phenomenon being shared
- Data requests from third parties: the system acts as an interface between user data and third parties
- System sends data to the third parties
- Data collection acceptance/denial by the user: again, the system acts as an interface between two world entities and knows what is happening
- The system asks for an ambulance dispatch to the external provider: the system produces an effect on the world, the dispatching, so the phenomenon is shared
- Position update on map for runners
- Organizers input runs data to the system
- Athletes enroll to runs available in the system: this phenomenon is shared because we assume that enrollment to runs happens only through the system and is not a copy of a physical form.

- Spectators connect to the system to watch runs: spectators are able to follow the live coverage through the system displaying the map, so the phenomenon is shared

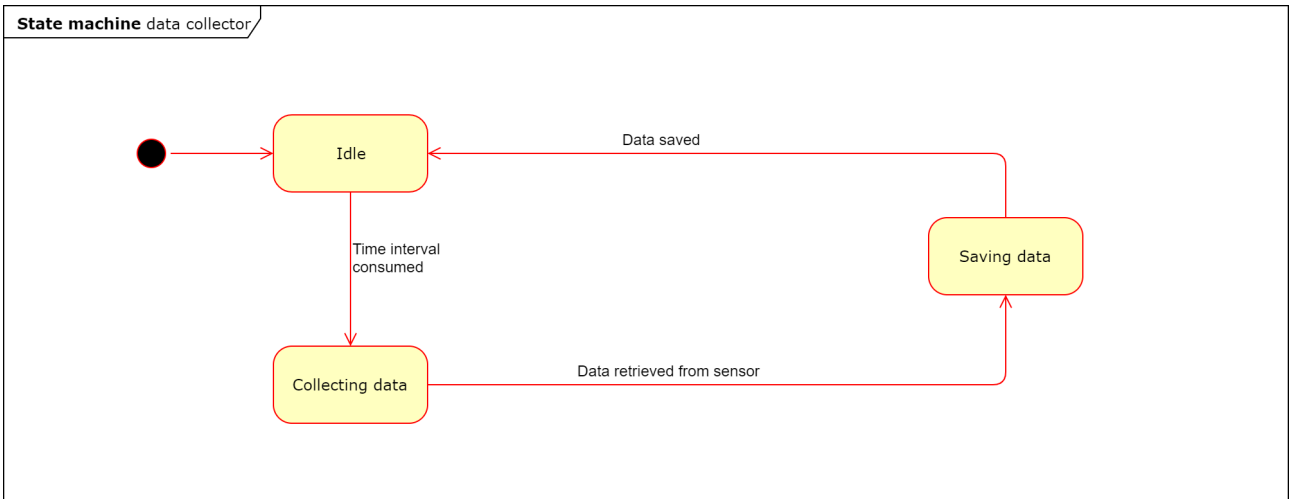
Machine: (*phenomena located entirely in the machine)*

- Database queries
- Anonymization check
- Check of health parameters against risk threshold
- System enforces unicity of usernames

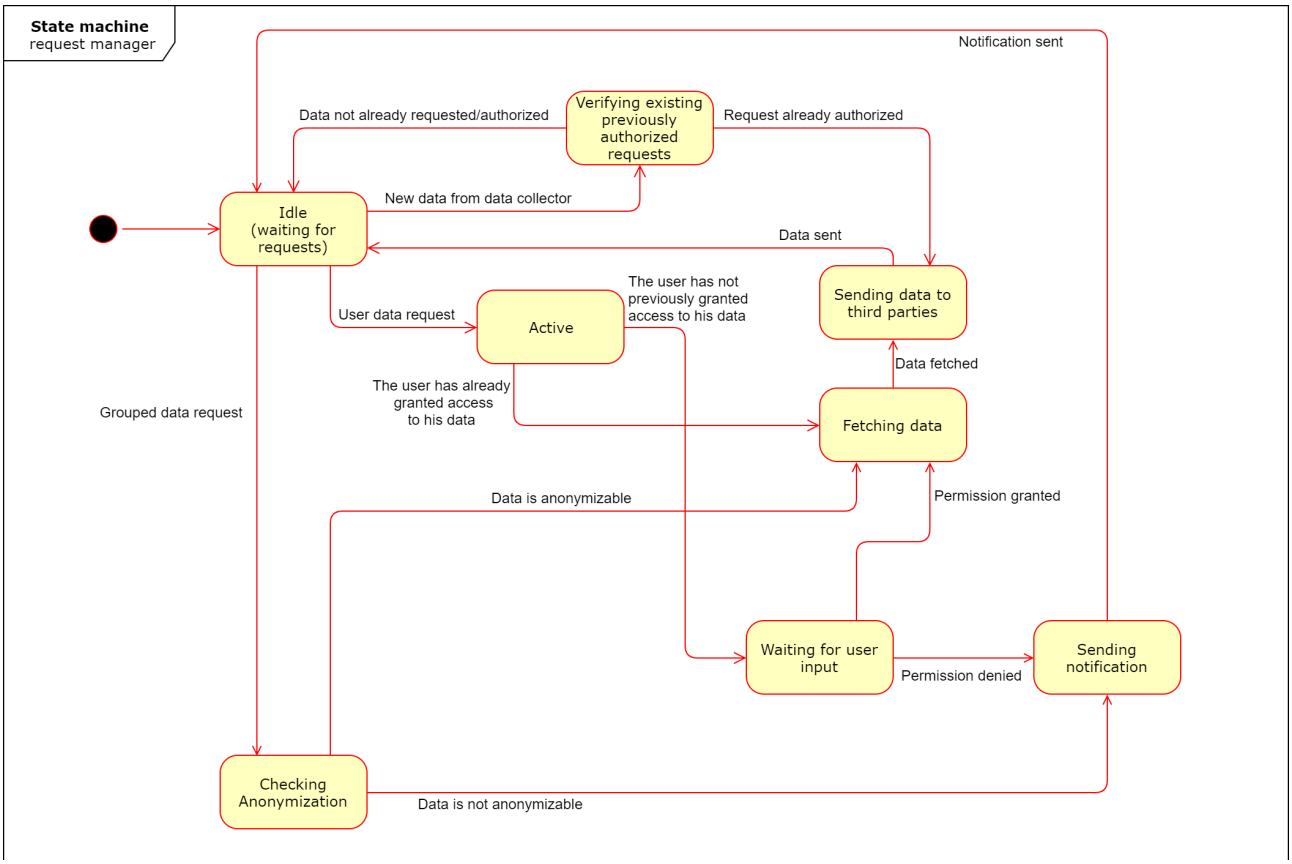
2.1.2 Diagrams



The class diagram shows how the system is structured. The main classes are User and Third Parties, which respectively represent the provider and consumer of data. There are two main different classes of data: Data corresponds to the information related to a specific individual while Group Data is a collection of anonymized Data requested by a Third Party. Each data request belongs to a Data Request class.
World Phenomena: User and Third Party
Shared Phenomena: Data Request
Machine Phenomena: Data classes



This state diagram shows the behavior of the *Data Collector* class, whose duty is to collect data from the sensors placed on the users' wearable devices. The class exhibits a cyclic behavior: starting from an idle state it awakens after a sampling interval time is consumed, then it fetches the data from the user's wearable device and proceeds to save it in the system. Once the save is completed, the class returns idle.



The request manager class, as the name suggests, manages data requests from third parties and fulfils them when data is available. The availability of the data depends on different conditions regarding the nature of the request:

- If the data request concerns a *single user's data*, the class shall verify if the user has previously granted the permission to allow access to their data from that specific

third party. If the permission was not already granted, the class shall proceed to ask the user for permission and fulfil the request if they grant it. Otherwise it shall notify the third party of the unavailability of the data.
- If the request is concerning grouped data, the class shall verify that the data can be anonymized (1000 or more users) and, if this is possible, send the data. Otherwise it shall notify the third party of the unavailability of such data.

Finally, the class is able to manage incoming new data from Data Collector class and dispatch it to the third parties, if they already placed an authorized request for such data.

2.2 Product Functions

(here we include the most important requirements - Summary of major functions)

2.3 User characteristics

(Anything that is relevant to clarify their needs)

2.4 Constraints

(Anything that will limit the developer's options (e.g. regulations, reliability, criticality, hardware limitations, parallelism, etc.)

2.5 Assumptions and Dependencies

Domain Assumptions

D1: Users are uniquely identified by their ID number or fiscal code. [G1, G2]
D2: Information provided by the user during the registration process are assumed to be true. [G1, G2, G3]
D3: User's position is available through GPS. [G1]
D4: User's health related data (heart rate and blood pressure) values are available through a wearable personal device [G1]
D5: A partner of TrackMe provides an ambulance service 24/7. [G4]
D6: The risk threshold for each user is obtained through a preventive hospital check. [G5]
D7: There is an external provider offering a map service [G8]

3 Specific Requirements

(the IEEE standard suggests 8 different templates for this section, we may have a look at them –
All the requirements go in here)

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 Functional requirements

(Taken from Prof document:
Definition of use case diagram, use cases and associated sequence/activity diagrams, and mapping on requirements. Scenarios)
It's important to keep track of the relation between use cases and requirements –
Traceability Matrix

3.2.1 Data4Help

Users: source users and third-party services.

G1: The user can be recognized by providing a form of identification
[D1]: Users are uniquely identified by their ID number or fiscal code.
[D2]: Information provided by the user during the registration process are assumed to be true
[R1]: The system shall allow registration of individuals through the creation of a username and a password.
[R2]: The system shall guarantee the unicity of usernames.
[R3]: The system shall ask the user to provide his personal data (birthdate, gender, residency address, ID number / fiscal code).
[R4]: The system shall ask the user to agree to a policy that specifies that, by registering, users agree that TrackMe acquires their data.

G2: Allow third parties to monitor data about location and health status of individuals.
[D3]: User's position is available through GPS.
[D4]: User's health data (heart rate and blood pressure) are available through a wearable personal device
[R5]: The system shall store past position and health data of every single user.
[R6]: The system shall support the registration of third parties.
[R7]: Third parties shall be allowed to subscribe to new data and the system sends data as soon as they are produced. *[also in G3, G4]*

G3: Allow third parties to access data relative to specific individuals
[D2]: Users are uniquely identified by their ID number or fiscal code.

[R7]: see above
[R8]: The third parties shall be able to request a specific individual's data through his ID number or fiscal code.
[R9]: Upon every data collection request, the system shall ask permission to the user, who can also deny it.

G4: Allow third parties to access anonymized data of groups of individuals
[R7]: see above
[R25]: The system shall allow third parties to search for the desired group of individuals.
[R10]: The system shall accept a grouped data collection request only if the data can be properly anonymized. Anonymization is considered proper if the number of people involved in the request is greater than 1000. *(Same comment as R9)*

3.2.2 AutomatedSOS

Users: elderly people and third parties.

G5: Allow third parties to offer a personalized and non-intrusive SOS service to elderly people so that an ambulance arrives to the location of the customer in case of emergency.
[D6]: The risk threshold for each user is obtained through a preventive hospital check.
[R11]: Frequently enough, health parameters are monitored by the system and compared against the threshold to detect risk situations.

3.2.3 Track4Run

Users: Runners, organizers and spectators

G6: Allow athletes to enroll in a run
[R13]: The system shall allow participants to register to the system.
[R14]: The system shall allow participants to check a list of available runs.
[R15]: The system shall provide the ability to enroll to the desired run only to registered athletes.
[R16]: The system shall allow enrolling only if the user has already agreed to share publicly his location for the duration of the run.

G7: Allow organizers to manage runs
[R17] The system shall allow organizers to register to Track4Run platform.
[R18] The system shall allow organizers to create and delete races.
[R19] The system shall allow organizers to add a path for the run.
[R20] The system shall allow organizers to check a participants list.

[R21] The system shall allow organizers to add or remove participants before the start of the race.

G8: Allow spectators to see on a map the position of all runners during the run
[D7] There is an external provider offering a map service
[R22] The system shall provide a public list of live runs
[R23] The system shall allow the spectator to see a map of the desired run, with live participants' position
[R24] The system shall update the positions of the runners on the map as soon as new data is received

3.3 Performance requirements

G5: Allow third parties to offer a personalized and non-intrusive SOS service to elderly people so that an ambulance arrives to the location of the customer in case of emergency.
[R1-NF]: A reaction time of less than 5 seconds from the time the health parameters are below the threshold must be guaranteed by the system.

3.4 Design constraints

3.4.1 Standards Compliance

3.4.2 Hardware limitations

3.4.3 Any other constraint

3.5 Software System Attributes

3.5.1 Reliability

3.5.2 Availability

3.5.3 Security

3.5.4 Maintainability

3.5.5 Portability

4 Formal Analysis using Alloy

(*In this section you will include your Alloy model. We require you to comment on the model by discussing the purpose of the model, what you can prove with it and why what you prove is important given the problem at hand. You are also required to show one or more worlds obtained by running your model)*

5 Effort Spent

5.1 Piccinotti Diego

Description of the task	Hours
Purpose, Scope, Definition	5
Product Perspective	6
Product Functions	
User Characteristics	
Domain Assumptions	3
Functional Requirements	3
Non-functional Requirements	
Formal Analysis Using Alloy	

5.2 Pietroni Umberto

Description of the task	Hours

Purpose, Scope, Definition	5
Product Perspective	3
Product Functions	
User Characteristics	
Domain Assumptions	3
Functional Requirements	3
Non-functional Requirements	
Formal Analysis Using Alloy	

5.3 Rossi Loris

Description of the task	**Hours**
Purpose, Scope, Definition	6.5
Product Perspective	1
Product Functions	
User Characteristics	3
Domain Assumptions	3
Functional Requirements	
Non-functional Requirements	
Formal Analysis Using Alloy	

6 References

WORD SHARED DOCUMENT