



POLITECNICO DI MILANO

MASTER'S DEGREE IN
COMPUTER SCIENCE AND ENGINEERING

SOFTWARE ENGINEERING 2

TrackMe

Requirements Analysis and Specification Document

Authors

Alberto ARCHETTI
Fabio CARMINATI

Reference professor

Elisabetta DI NITTO

v.0 - October 26, 2018

Contents

1	Introduction	1
1.1	Purpose	1
1.1.1	Project description	1
1.1.2	Goals	1
1.2	Scope	2
1.2.1	World	2
1.2.2	Shared phenomena	2
1.3	Definitions, acronyms, abbreviations	3
1.3.1	Definitions	3
1.3.2	Acronyms	3
1.3.3	Abbreviations	3
1.4	Revision history	3
1.5	Reference documents	3
1.6	Document structure	3
2	Overall description	5
2.1	Product perspective	5
2.2	Product functions	5
2.3	User characteristics	5
2.3.1	Actors	5
2.4	Assumptions, dependencies, constraints	5
2.4.1	Domain assumptions	5
3	Specific requirements	6
3.1	Scenarios:	6
3.2	External interface requirements	6
3.2.1	User interfaces	6
3.2.2	Hardware interfaces	6
3.2.3	Software interfaces	6
3.2.4	Communication interfaces	6
3.3	Functional requirements	6
3.4	Performance requirements	6
3.5	Design constraints	6
3.5.1	Standards compliance	6
3.5.2	Hardware limitations	6
3.5.3	Any other constraint	6

3.6	Software system attributes	6
3.6.1	Reliability	6
3.6.2	Availability	6
3.6.3	Security	6
3.6.4	Maintainability	6
3.6.5	Portability	6
4	Formal analysis using Alloy	7
5	Effort spent	8

1 Introduction

1.1 Purpose

1.1.1 Project description

TrackMe wants to develop a software-based service that allows individual users to collect health data, called **Data4Help**. This data can be retrieved from the system and visualized according to different filters by a user interface. The system allows third parties registration. Third parties can request access to users' collected data in two ways:

Single-person data After the third party makes a request to the system for a single user data sharing (providing for example the fiscal code of the user), the system asks the user for authorization; if positively provided, the third party is granted access to the user's data

Anonymous-group data Third parties can be interested in big amounts of data, but not in who are the people that are providing it; the system, once the request by the third party is sent, checks if the data can be effectively anonymized (it must find at least 1000 people that match the third party request) and, if positively evaluated, grants access to the anonymized data to the third party that requested it

Third parties can subscribe to new data and receive it as soon as it is collected by the system.

Another service that TrackMe wants to develop is **AutomatedSOS**, built on **Data4Help**. This service analyzes users' data and calls a SOS whenever data exceeds the basic health parameters. For this particular purpose, system performances will be a critical aspect to be taken into account, because even the slightest delay matters in critical health situations.

1.1.2 Goals

Here we present the goals that will be reached once the project is completed:

G.U1 Users can collect, store and manage their health data

G.U2 Users can choose to have their health monitored; if their health is critical, an ambulance will be dispatched

- G.T1** Third parties can ask single users for their health data sharing
- G.T2** Third parties can request access to anonymized data that comes from groups of people
- G.T3** Third parties can subscribe to new data and receive it as soon it is produced

1.2 Scope

1.2.1 World

Our *world* is composed of two main actors: users and third parties. Users are interested in monitoring their health parameters and third parties are interested in developing services or researches that exploit the data gathered by the users. **Data4Help** is the service that acts as a bridge between these actors' needs.

Phenomena that occur in the *world* and are related to our application domain are

- physical conditions of our users
- third parties projects
- SOS system that is able to dispatch ambulances

These phenomena exist in the *world*, but cannot be observed directly by our system.

1.2.2 Shared phenomena

In order to communicate with the *world*, our system needs to share some aspects with it. We will list the aspects controlled by the world, but observable by the machine:

- physical parameters of the users, gathered through sensors on wearable devices
- third parties requests to the system for the data they need
- users' location, acquired through GPS signals

On the other hand, the aspects that occur in the machine, but are observable by the world are

- interfaces that organize the gathered data that can be filtered according to time or type of data
- messages for the SOS system, that are sent in case of critical health of a user

1.3 Definitions, acronyms, abbreviations

1.3.1 Definitions

1.3.2 Acronyms

1.3.3 Abbreviations

1.4 Revision history

Version	Log
v.0	First version of RASD completed

1.5 Reference documents

See References for details on the consulted documents.

1.6 Document structure

This document uses the IEEE standards for requirement analysis documents as a guideline towards a clear and logical explanation of its contents:

- Section 1 gives a brief introduction on the project to be developed and adds notes on references and revisions

- Section 2 describes the world and the shared phenomena, by defining assumptions and constraints; it identifies also the goals and the main functions of the project
- Section 3, as the main part of this document, is about requirement analysis; it has also sections about interfaces of the system and software attributes
- Section 4 contains the **Alloy** model that certifies correctness of goals implication by requirements and domain assumptions
- Section 5 lists the overall modifications and additions to this document, ordered by date, as the hour counter of effort spent by each group member

2 Overall description

2.1 Product perspective

2.2 Product functions

2.3 User characteristics

2.3.1 Actors

User Person that has successfully created an account of TrackMe. She or He can exploit all the functionalities of the application

Third Party Entity that can request to Data4Help the access to either individual or group DataSets

2.4 Assumptions, dependencies, constraints

2.4.1 Domain assumptions

D1 da 1

D2 da 2

3 Specific requirements

3.1 Scenarios:

3.2 External interface requirements

3.2.1 User interfaces

3.2.2 Hardware interfaces

3.2.3 Software interfaces

3.2.4 Communication interfaces

3.3 Functional requirements

3.4 Performance requirements

3.5 Design constraints

3.5.1 Standards compliance

3.5.2 Hardware limitations

3.5.3 Any other constraint

3.6 Software system attributes

3.6.1 Reliability

3.6.2 Availability

3.6.3 Security

3.6.4 Maintainability

3.6.5 Portability

4 Formal analysis using Alloy

5 Effort spent

References

- [1] Mandatory Project Assignment AY 2018-2019
- [2] IEEE 830-1993 - IEEE Recommended Practice for Software Requirements Specifications
- [3] ISO/IEC/IEEE 29148 - Systems and software engineering — Life cycle processes — Requirements engineering
- [4] Collection and Processing of Data from Wrist Wearable Devices in Heterogeneous and Multiple-User Scenarios
<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5038811/>
- [5] Google Fit API
<https://developers.google.com/fit/overview>
- [6] Slides of the course by Prof. Di Nitto
<https://beep.metid.polimi.it/>
- [7] L^AT_EXtemplates
<http://www.latextemplates.com/>
<http://www.overleaf.com/latex/examples/title-page-with-logo/hrskypjpkrrpd>