Politecnico di Milano

AA 2018-2019

POLITECNICO
MILANO 1863

Software Engineering 2

RASD

Requirements Analysis and Specification Document

version 2 - 16.12.2018

Matteo Acerbi

Table of Contents

1. Introduction

1.1 Purpose

Personal data, which include examples such as digital informations on the position of users and their health status, but also on telephone call logs or web searches, are undoubtedly the oil of modern data-intensive science and the online economy, and are the basis for apps to provide intelligent services and personalized experiences to each user. The development of databases to manage them is therefore of fundamental importance.
In this context, the purpose of this document is to develop an application called TrackMe, which can be run both on a mobile device and on a PC, that offers an internal service called DATA4Help, designed to be a Personal DataSpace Management System (PDSMS). In our idea, by creating a TrackMe account, each user has the possibility to be associated with a personal repository, managed by DATA4Help, in which to store and handle his personal data, specifying the field of belonging and being able to count on the respect of his privacy, which must necessarily be guaranteed by this service. TrackMe is also intended to give, to registered third parties, the possibility to request access to data of registered users, in order to monitor their location and health status.
Specifically, if a registered third party is interested to receive data of a specific individual, a query will be generated and managed by TrackMe, which will pass it on to the involved user. In this regard, the user will been given the chance to express his will, choosing whether to share his data or not.
Similarly, registered third party can also request TrackMe to receive anonymous data concerning groups (class) of individuals and in this case TrackMe will directly verify whether it is possible or not to make the requested data properly anonymous and then grant or not their transmission.

1.1.2 Goals

[G1] TrackMe allows users and third parties' employees to be recognized by providing a method of identification

[G2] DATA4Help has to work as a cloud datastore service, in which each user can be associated with a private repository prepared to collect user's personal data, in the total respect of the GDPR.

[G3] The user is guaranteed to have access to his stored data at any time, in order to monitor and possibly modify them.

[G4] Each user can express preferences that will be associated with him and stored by DATA4Help in his personal repository. This preferencs may concern the user's will to make his personal data not accessible or only partially accessible or to define certain third parties as not allowed or only partially allowed to access some data.

[G5] TrackMe allows third parties registration

[G6] Third parties can ask for data relating to a specific individual

[G7] Third parties can ask for anonymous data relating to a group (class) of individuals

[G8] When a query to a specific individual is done, the involved user must be enabled by TrackMe to express his will to share his data or not. At the same time, when a query to a group of individuals is done, TrackMe must be able to understand if requested data can be released or not by the system.

1.2 World, Shared and Machine Phenomena

1.2.1 World Phenomena

1. People can use smartwatches or similar devices for their utilities.

2. Each person is associated with personal data, such as data concerning his geographical location and his health status.

1.2.2 Shared Phenomena

1. Each person can use his smartwatch in order to collect and eventually share his personal data with third parties.

2. Third parties may be interested on data concerning a specific individual or a group of individuals and therefore ask for accessing them.

1.2.3 Machine Phenomena

1. Each user can create a Personal Data Space (PDS), that is a repository in wich to collect all his personal data.

2. Each user can specify preferences in which to define which data are accessible and which are not and, where appropriate, to indicate particular third parties that do not have access to certain data. This preferences are also stored in the PDS associated to the user thar has defined them.

3. When a third party asks for data relating to a specific user, the preferences specified by that user are checked and if the third party in question does not appear among those specified by the user as not allowed, the request is passed to the user who can then accept it or reject it.

4. If a third party asks for data related to a group of individuals, the cardinality of this group (class) is checked, in an attempt to verify whether it is possible or to anonymize the requested data. If the number of users belonging to this class is less than 1000, the query is rejected.

1.3 Definitions, Acronyms, Abbreviations

1.3.1 Definitions

- User (Data Owner): specific individual who, by choosing to take advantage of DATA4Help service, accepts that TrackMe accesses his personal data, providing the creation of a repository associated to him, that contains his own data and to which he can access at any time.

- Data: single information concerning a specific user and included by the same user in his associated repository. Data can belong to different fields, but for the purpose of this project we consider for semplicity only two types of data, Health Data and Location Data.

- D4HPreferences: set of wills expressed by the user and associated to his account. Each user can define some data as not accessible by anyone except himself or certain third parties as not allowed to access to some type of data.

- Repository (PDS): digital personal dataspace, stored in the database managed by DATA4Help, that contains all the informations associated with a single user .

- ThirdPartyEmployee (Data Demander): literally a third party's employee, e.g. a person employed by a specific third party in order to perform data queries on its behalf. A third party is an external service or company, which is given the opportunity to use TrackMe in order to access specific data concerning individuals, or anonymous data concerning groups of individuals. The ThirdPartyEmployee, by registering correctly with TrackMe, will be associated with an account that will differ from the normal user's account in some functionalities.

- Query: request provided by a ThirdPartyEmployee of data concerning either a specific individual or a particular group of individuals, like those people belonging to a specific age class or people that live in a particular geographic area.

1.3.2. Acronyms

- RASD: Requirement Analysis and Specification Document

- API: Application Programming Interface

- PDS: Personal Data Space

- PDSMS: Personal DataSpace Management System

- GDPR: General Data Protection Regulation

1.3.3. Abbreviations

- [Gn]: n-th goal
- [Dn]: n-th domain assumption
- [Rn]: n-th functional requirement

1.4 Document Structure

Chapter 1 gives an introduction to the problem, identifying the purpose and the scope of the RASD Document and also defining abbreviations and acronyms in order to give the reader a clear and complete vision of the problem that will be addressed in the following chapters.

Chapter 2 defines the product perspective, by using class diagram in order to identify the subjects involved in the model and state diagram to describe the process of asking for data concerning either a specific user or a class of users. In this chapter are also analyzed product functions, e.g. the main functionlities that TrackMe must ensure, user characteristics and finally domain assumptions

Chapter 3 contains the external interface requirements, including: user interfaces, hardware interfaces and communication interfaces. In this chapter are also defined the functional requirements by using use case and sequence diagram.

Chapter 4 shows the Alloy model and also a world generated by it

Chapter 5 lists all the reference documents used as a bibliographic support to develop the RASD Document.

2. Overall Description

2.1 Product Perspective

As already mentioned above, DATA4Help is a service to be understood as a PDSMS, in which each of the users who is registered, become associated with a repository, i.e. a private space containing his personal data. A third party may have access to this service, by requesting informations about a specific individual or a group of individuals and waiting for a response regarding the possibility or not to allow data transmission. To better understand the structure of the domain model, the following class diagram is shown:

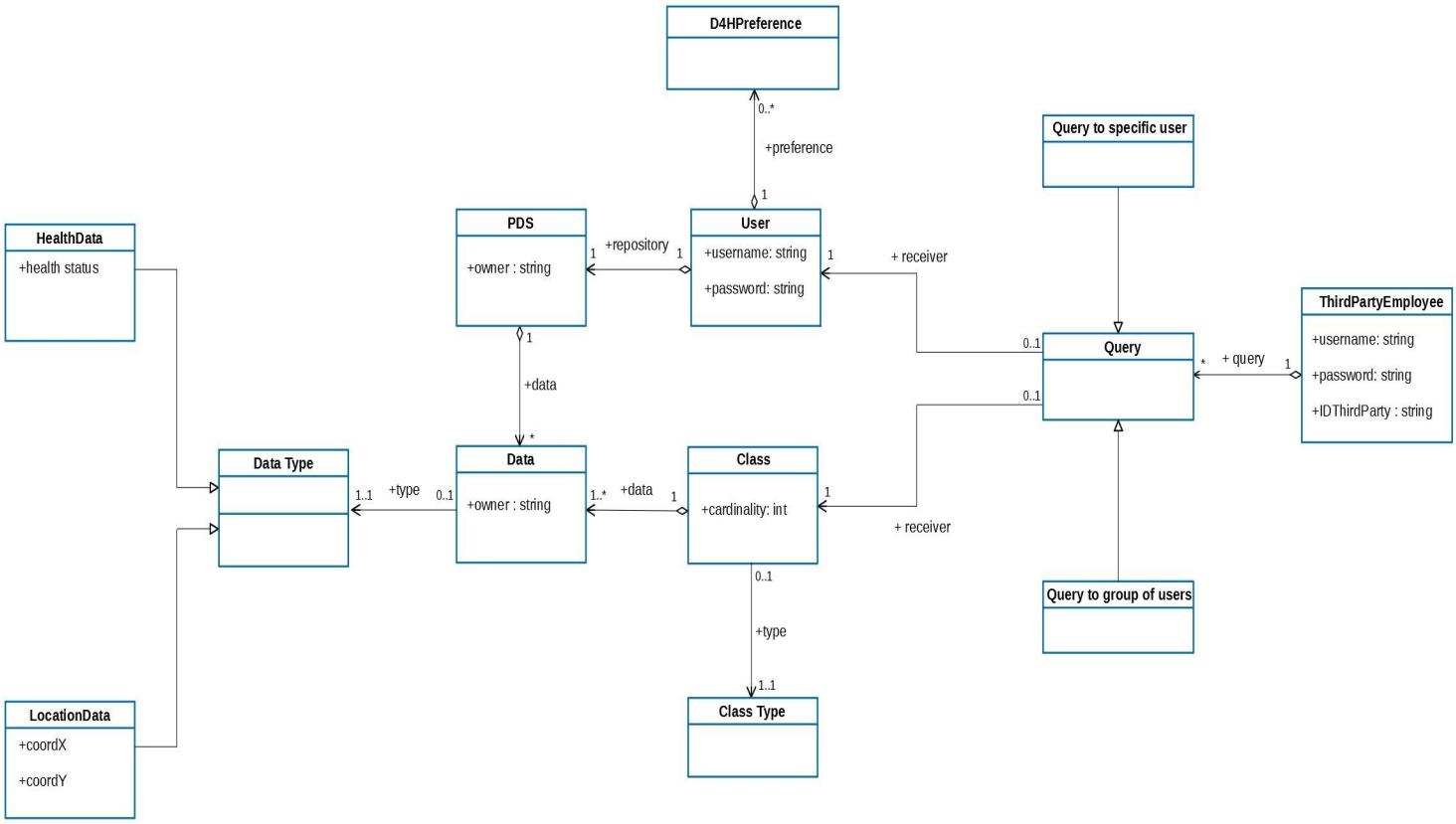

Figura 1: Class Diagram

In addition to the previous diagram, a state diagram is defined below. It rappresents graphically the request provided by a third party, either to a single individual or to a group of individuals:

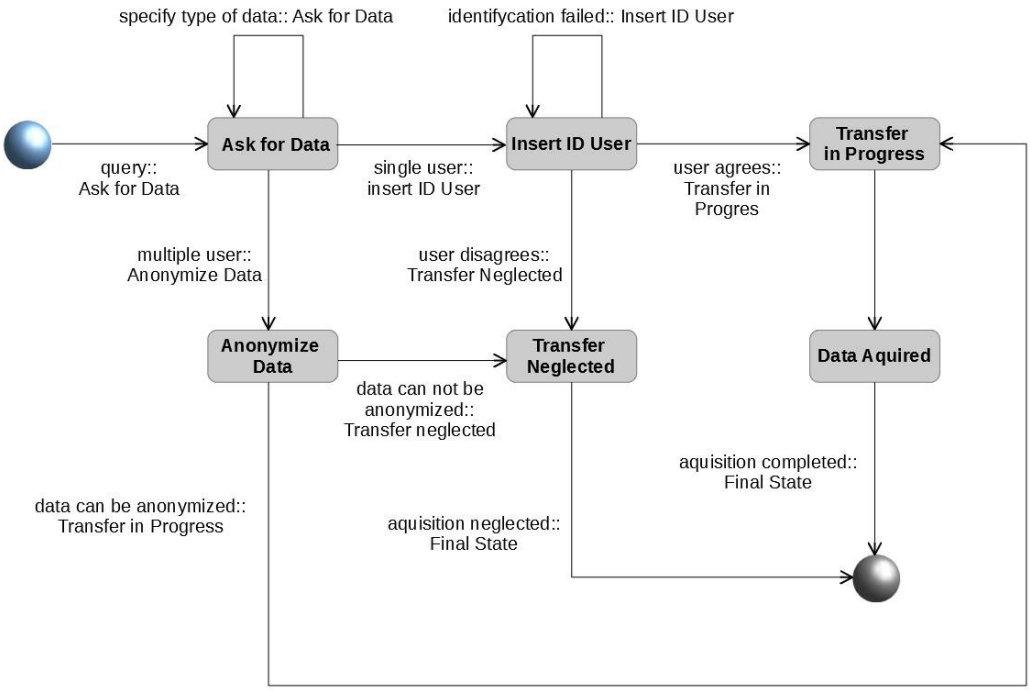

Figura 2: State Diagram

In this diagram all possible situations arising from a third party's query have been taken into account and therefore represented. In particular, when defining the portion of diagram relating to the query for data concerning a single individual, it was assumed that the third party's employee is in possession of an identifier of the individual, such as a security number or a fiscal code in the italian case, that must be provided to TrackMe in order to search for his associated PDS through DATA4Help.

2.2 Product Functions

Referring to the goals reported in section 1.1.2, it is easy to see that the main functions that this service must ensure are essentially 3, listed below:

- Personal Data Collection and Storing
- Preferences Management
- Data Sharing Control

In the following sections these functions will be reviewed and analysed in more detail.

2.2.1 Personal Data Collection and Storing

As specified in [G2] and [G3], DATA4Help must work as a Personal DataSpace Management System, that is as a sort of online store where users's data are permanently collected and to which TrackMe accesses in order to acquire some of them for data sharing, as a virtual space in which the supplydemand forces and the exchange of information co-exist. A repository is set up for each user who registers for this service, to act as a storage space for personal informations, ensuring that these informations will not be lost and that will be accessible for the user at any time. Personal data, the digital record of "everything a person makes and does online or in the real-world", encompasses digital identity; relationships to other people and organizations; real-world and online context activities, interests and behavior; communications data and logs; media produced, consumed and shared; financial data; health data and institutional data. In the specific case of DATA4Help, what we consider as the most relevant are location and health data.

2.2.2 Preferences Management

TrackMe should not only be responsible for the data storage, done through DATA4Help. As already defined in [G7], each individual, by registering for the service and becoming an associated user, has the opportunity to express preferences regarding the accessibility of his personal data. Specifically, the user may express the wish to render unaccessible all his data to any third parties, or define only certain data as accessible. It may also be possible for the user to define certain third parties as not allowed to access to any data or to only certain data. TrackMe must therefore provide, through DATA4Help, a space in which these set of preferences will be stored and it must also ensure that these preferences will be observed and respected from the moment of their definition onwards, in accordance with the wishes of the user.

2.2.3 Data Sharing Control

The main operation that TrackMe has to guarantee is data sharing, i. e. providing the possibility for third parties to register and request the acquisition of data concerning either specific individuals or groups of individuals. In the first case, once the query has been received and the profile associated with the user whose data the third party wants to acquire have been recognized, TrackMe itself must first consult the user's preferences set stored within DATA4Help. If, by checking user's preferences, it emerges that data demanded are not shareble, TrackMe denies the transaction by sending a

warning to the ThirdPartyEmployee, while, if there is no impediment to the transaction, TrackMe will send a notification to the involved user, waiting for his response. In case of query to a group of users, TrackMe will directly determine whether the transaction is possible or not, basing the decision on whether or not is possible to anonymize data required. Also in this caso TrackMe has to access to the database, through DATA4Help, in order to verify the cardinality of the class. If the cardinality is less then 1000, TrackMe will deny the transaction by sending a warning to the ThirdPartyEmployee, otherwise it will acquire data and send them to the ThirdPartyEmployee.

2.3 User Characteristics

The principal characters involved in this model are exentially the sequent:

- User

An individual that is succesfully registered with TrackMe and that has an associated repository stored within DATA4Help, which he can manage at any time. In the domain of a data market he is a data owner who has the total control of his own data.

- ThirdPartyEmployee

A person employed by a specific third party in order to perform data queries on his behalf, concerning either a specific individual or groups of individuals . In the domain of a data market he is a data demander.

2.4 Domain Assumptions

[D1] User's login credentials and ThirdPartyEmployee’s login credentials are unique within the system

[D2] Each repository stored in DATA4Help refers to no more than one user

[D3] Acquisition of an individual's personal data may be made through smartwatches, smartphones or other devices owned by the same individual

[D4] Each individual is associated with an identifier, such as security number, identity card or fiscal code in the italian case, which distinguishes him and which can then allow both TrackMe and third parties to identify him

[D5] There is a reliable Internet connection that allows to use TrackMe app or TrackMe website

3. Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interface

The following project does not define an actual visual implentation of a TrackMe user interface. Anyway, possible ideas to define the structure of this type of interface are reported. As mentioned before, the main idea of this project is that TrackMe is seen as a mobile application, available in an App Store, but also as a website accessible for any user. In this vision DATA4Help is an internal serive offered by TrackMe that works like a Data Space Management System.
Once downloaded from the digital store, the user will be able to run TrackMe, that will first ask him to provide his login credentials in order to log in, while, if the user has not yet been registered, to register for the service.
In order to do this, the user will be shown a first screen in wich there will be a section of Log in, with associated fields in which to enter Username and Password, and a section for the Sign Up prepared for the registration of new users.
At this point we have to make a distinction between the GUI prepared for a normal user and the one prepared for a third party's employee, beacuse, as said previously, they have two distinct roles within the model that we have defined, and TrackMe has to provide different functionalities for each of them.
In the first case, if the normal user wants to be registered on TrackMe, he only has to select the Sign Up section. He will be asked to provide his security number or his fiscal code, a username of his choice and a password that respects the security fees imposed by the system. Once provided valid credentials, the user will be directed to a new window where he will be offered the possibility of being associated to a repository on DATA4Help, containing his personal data, thereby accepting that these data will be acquired by the system.
Once this procedure has been completed, the user is correctly authenticated and is then directed to the screen corresponding to his TrackMe profile. In particular, a notification section will be set up, in order to allow the user to view possible reports of queries for data access provided by third parties. There will be also a section concerning user'preferences, where the user will have the possibility to define certain data as not accessible and also to designate specific third parties as not allowed to access some specific data.
In the latter case, the ThirdPartyEmployee has to register himself with TrackMe, providing an ID of the third party to which he belongs. Once this procedure is done correctly, he has total access to TrackMe functionalities. In particular, the functionalities that TrackMe offers to third parties's employees concern making queries, download data acquired and view notifications (eventual warnings).

3.1.2 Hardware Interfaces

Trackme is meant to be a software application that offers an internal service called DATA4Help, which has the main function to collect and manage personal data, allowing also their transmission to third parties, according to the volunties expressed by data owners. It is not using any hardware interfaces but certainly it requires using a smartphone to run TrackMe app or a personal computer in order to reach TrackMe website. In both cases an internet connection is essential. For the purpose of data acquisition, also smarthwatches can be used.

3.2 Functional Requirements

[G8] When a query to a specific individual is done, the involved user must be enabled by TrackMe to express his will to share his data or not. At the same time, when a query to a group of individuals is done, TrackMe must be able to understand if requested data can be released or not by the system.

[G1] TrackMe allows users and third parties'employees to be recognized by providing a method of identification

[D1] User's login credentials and ThirdPartyEmployee's login credentials are unique within the system.

[R1] Each user and each third party's employee can create an account for the usage of TrackMe, by selecting a username and a password. The username must not be already taken by any other user and the password must respect the security rules imposed by the system.

[G2] DATA4Help has to work as a cloud datastore service, in which each user can be associated with a private repository prepared to collect user's personal data, in the total respect of the GDPR.

[D2] Each repository stored in DATA4Help refers to no more than one user.

[D3] Acquisition of an individual's personal data may happen through smartwatches, smartphones or other devices owned by the same individual

[R2] By only providing the credentials that have been defined during the registration, the user is allowed to create ad manage his associated repository in DATA4Help.

[G3] The user is guaranteed to have access to his stored data at any time, in order to monitor and possibly modify them.

[R3] The user, by providing an email address, can recover login credentials if he loses them or he is unable to enter them.

[R4] Any action or change that a user makes on his PDS is stored and can not be lost

[G4] Each user can express preferences that will be associated with him and stored by DATA4Help in his personal repository. This preferencs may concern the user's will to make his personal data not accessible or only partially accessible or to define certain third parties as not allowed or only partially allowed to access some data.

[R5] A set of preferences cannot be associated with more than one user.

[R6] Only one set of preferences can be associated to each user.

[G5] TrackMe allows third parties registration

[D1] User's login credentials and ThirdPartyEmployee's login credentials are unique within the system.

[R7] Each third party's employee has to provide a valid ID of the third party to which he belongs, in order to register correclty within TrackMe.

[G6] Third parties can ask for data relating to a specific individual

[D4] Each individual is associated with an identifier, such as security number, identity card or fiscal code in the italian case, which distinguishes him and which can then allow both TrackMe and third parties to identify him.

[R8] A third party's employee must provide a valid identifier of the user from whom he wants to receive some data and explicitly declare to the service what type of data the third party is interested about.

[R9] When a third party requests to access the data of a single user, TrackMe checks the preferences defined by the user, stored in his associated PDS. If TrackMe finds that the user have declared in his preferences the will not to allow access to any data, or that the data requested by the third party do not fall within the data defined in the user'preferences as accessible, or that the third party is not allowed to access that particular type of data, TrackMe will immediately notify the third party that it is impossible to access data, by sending an appropriate Warning. Otherwise, TrackMe will send a notification to the user, who may then give his consens or not, and then communicate the outcome to the third party.

[G7] Third parties can ask for anonymous data relating to a group (class) of individuals

[R10] The third party's employee must specify the class type and what data he wants to obtain.

[R11] Queries for anonymous data relating to groups of individuals are handled directly by TrackMe, which must check whether these data can be anonymized or not. If TrackMe can not anonymize those data, the transmission is not allowed.

[R12] If the number of members of the class whose data are requested is less than 1000, TrackMe notifies the third party's employee that it is unable to allow the transmission, by sending him a Warning.

[R13] If the number of members of the class exceeds 1000, TrackMe collects the required data within DATA4Help and sends it to the third party, with a notification of transmission.

3.2.1 Use Case Diagram

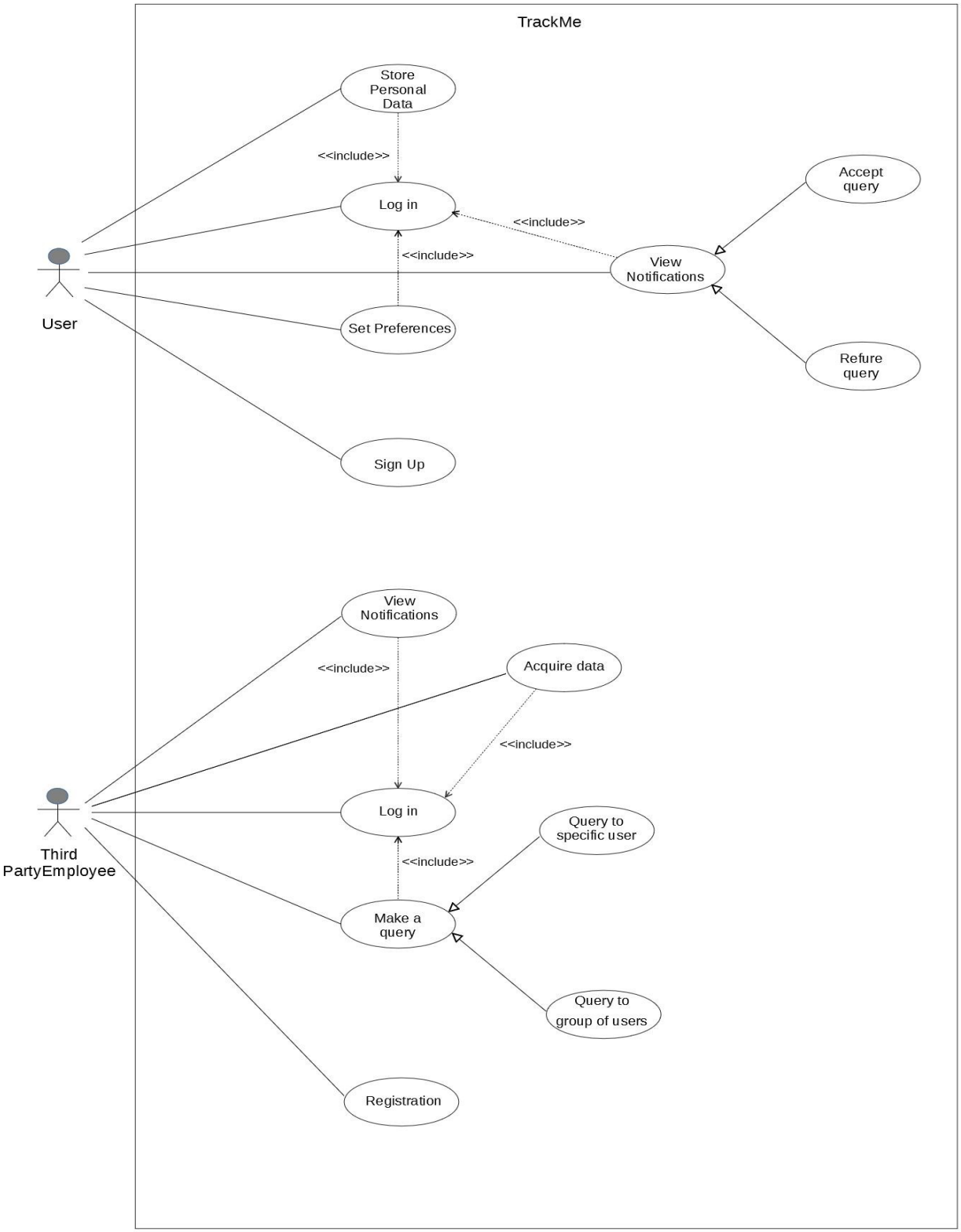

Figura 3: Use Case Diagram

Name	Sign Up
Actor	User
Entry conditions	The user has installed TrackMe application on his/her device or has accessed to TrackMe website.
Events Flow	1. Enter in the “Sign up” field 2. Provide all the necessary informations to create an account, like a valid email address, a password, a security number or a fiscal code and a Username 3. Cilck on “Create a personal PSD in DATA4Help” 4. Agree that TrackMe’ll acquire user’s personal data 5. Click on “Confirm” button 6. The system saves the data
Exit Conditions	A TrackMe account has been succesfully created, with also an associated personal data space registered in DATA4Help
Exceptions	1. The user is already signed up 2. The user didn’t fill all of the mandatory fields with valid data 3. The username is already taken 4. The e-mail is already registered 5. All the exceptions are handled by notifying the user and taking him back to the sign up activity.

Below we report two activity diagrams related to the Login operation and the Notifications checking operation. Because these two operations concern both the normal user and the third party’s employee, we use the more generic term Client to identify both types of users:

Name	Log in
Actors	Client
Entry conditions	The Client has properly registered a TrackMe account associated to him
Events Flow	1. The Client provide the credentials that he has defined during the sign up (registration), like a “Username” and an associated “password” 2. Click on “Access” button
Exit Conditions	The Client can access to his TrackMe account
Exceptions	1. The Client enters invalid Username 2. The Client enters invalid Password 3. All the exceptions are handled by notifying the Client and taking him/her back to the login activity

Name	View Notifications
Actor	Client
Entry conditions	The Client has already logged in
Events Flow	1. The Client accesses the newsletter prepared in his personal account. 2. The Client checks if there are some queries concerning his personal data (if he/she is anormal user) or if there is any communication about outcomes concerning previous queries (if he/she is an employee of a third party).
Exit Conditions	The Client has checked all the notifications sent to his profile.
Exceptions	/

Name	Set Preferences
Actor	User
Entry conditions	The user has already logged in
Events Flow	1. The user chooses setting preferencies. 2. The user specifies if he wishes to render all or some of his data unaccessible to any third party. He also specify if there are some third parties who can not access to some specific data. 3. TrackMe collects the preferences set and store it into the database, through DATA4Help service.
Exit Conditions	The preferences specified by the user are properly registered in his account and stored within his personal repository. This set of preferences will be respected by the system in order to garatee user's privacy
Exceptions	/

Name	Accept query
Actor	User
Entry conditions	The user has already viewed some queries in his newsletter
Events Flow	1. The user answers to the request by writing a message. 2. The user allows the third party to access the data requested. 3. TrackMe takes the message provided by the user and sends a notification to the third party's employee.

Exit Conditions	TrackMe provides the trasmission of user's personal data to the third party, respecting the wishes of the user
Exceptions	1. TrakMe fails sending the message to the third party's employee due to problems with the internet connection. 2. The data transmission fails within DATA4Help.

Name	Refuse query
Actor	User
Entry conditions	The user has already viewed some queries in his newsletter
Events Flow	1. The user answers to the request by writing a message. 2. The user doesn't allow the third party to access the data requested. 3. TrackMe takes the message provided by the user and sends a warning to the third party's employee.
Exit Conditions	TrackMe informs the third party that the user hasn't allowed his data transmission, so the transmission doesn't take place
Exceptions	1. TrakMe fails sending the notiefication to the third party due to problems with the internet connection

Name	Registration
Actor	Third Party's employee
Entry conditions	The third party's employee has already accessed to TrackMe website or to TrackMe applications
Events Flow	1. The third party's employee accesses a specific field dedicated to third party, distinct from the registration field dedicated to normal users. 2. The third party's employee provides all the necessary informations in order to identify the third party to which he/she belongs. 3. The system saves the data.
Exit Conditions	The third party' employee is correctly registered to TrackMe and can take advantage of the services offered by it, including the possibility of requesting access to data concerning specific individuals or groups of individuals
Exceptions	1. The third party's employee is already registered. 2. The third party's employee didn't fill all of the mandatory fields with valid data 3. All the exceptions are handled by notifying the third party and taking it back to the registration activity.

Name	Make a query/ Query to specific User
Actor	Third Party's employee
Entry conditions	The third party's employee has already logged in
Events Flow	1. The third party's employee accessess the account space set up by TrackMe for sending queries concerning data sharing. 2. The third party's employee expressess the particular wish to access some data about a specific individual. 3. The third party's employee provides an identifier of the user to request data from. 4. The third party's employee specifies which type of data is interested to access.
Exit Conditions	All the details specified by the third party's employee have been collected by TrackMe, which will check if possible to share data.
Exceptions	1. By checking the preferences set associated with the user from which the third party would like to obtain certain data, TrackMe has verified that such requested data can not be transmitted to the third party. 2. The indentifier provided by the third party doesn't refer to any user. 3. The user from whom certain data would be obtained has no longer an associaed TrackMe account. 4. TrakMe fails to receive the query due to problems with the internet connection. 5. All the exceptions are handled by notifying the third party's employee and taking him/her back to making queries activity.

Name	Make a query/ Query to group of Users
Actor	Third Party
Entry conditions	The third part has already been correctly registered
Events Flow	1. The third party's employee accessess the account space set up by TrackMe for sending queries concerning data sharing. 2. The third party'employee expresses the particular wish to access some data about a group of individuals. 3. The third party's employee defines what type of group (class) of individuals it is interest about. 4. The third party's employee specifies which type of data is interested to access.
Exit Conditions	This type of query is properly received and directly handled by TrackMe that approves it if it is able to properly anonymize the requested data
Exceptions	1. TrakMe fails to receive the query due to problems with the internet connection. 2. TrackMe finds that the number of individuals whose data satisfy the request is lower than 1000, so it can not accept that request. 3. All the exceptions are handled by notifying the third party and taking it back to making queries activity.

3.2.2 Sequence Diagrams

3.2.2.1 Query for data concerning a specific user

Below is shown a sequence diagram that describes the workflow that corresponds to the principal objective of TrackMe, that is the possibility for third parties to demand data about both a specific individual and a group of individuals. Specifically, the diagram below describes the query for access to personal data concerning a specific individual registered within DATA4Help.

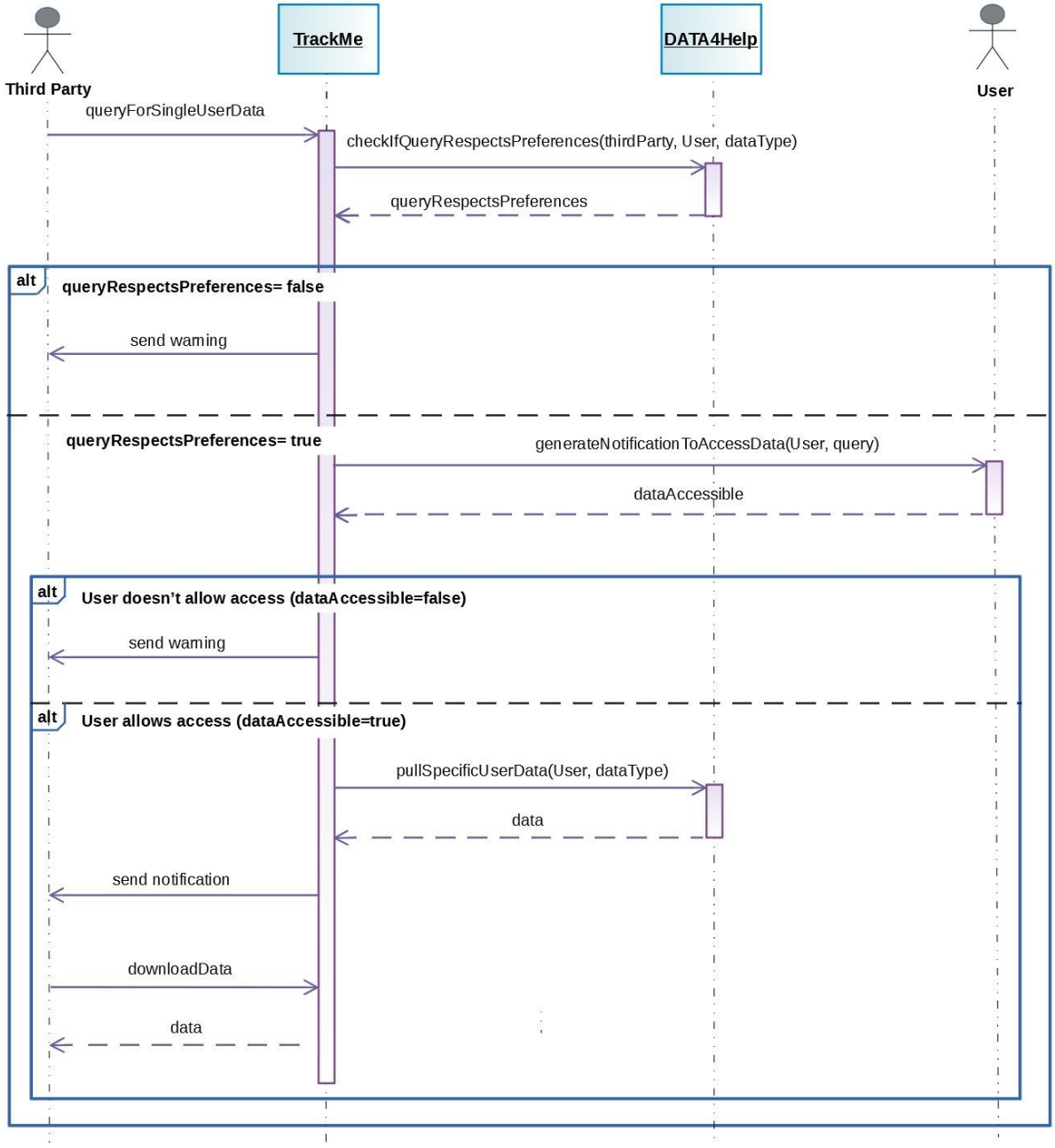

Figura 4: Sequence Diagram - Query for data concerning a specific user

3.2.2.1 Query for data concerning a group of users

The diagram below describes the query for access to anonymize data concerning a specific group (class) of individuals registered within DATA4Help.

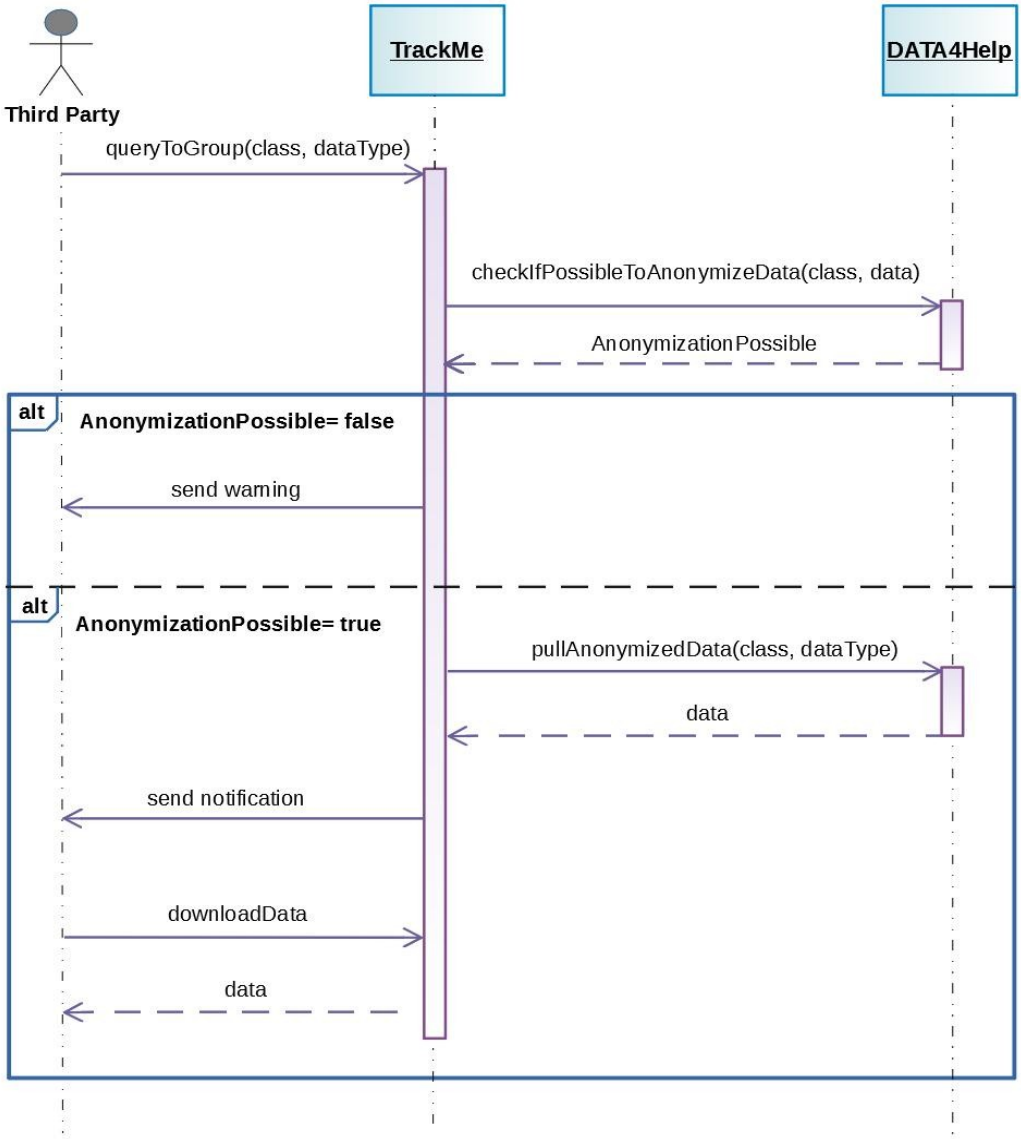

Figura 5: Sequence Diagram - Query for data concerning a group of users

3.3 Performance Requirements

TrackMe must be able to serve fairly great numbers of users simultaneously. In particular, it is required that his internal service DATA4Help, as a DataSpace Management System,will able to manage many repositories, each one associated with a specific user, while constantly ensuring data privacy.
Data available from the participants are heterogeneous and dynamic. Accessing them requires support for multiple Web APIs, formats, and protocols, as well as high tolerance to errors and unknowns.
This service must have available all the informations required to be able to access the data in all the participating data sources, thus relying on the identity management and cataloging services.
When a third party's employee requests data relating to a specific user, TrackMe has to recognize the repository associated to that user through DATA4Help, check the set of preferences saved into it, pull the required data and provide their transmission to the third party's employee. By focusing on accessing data collected in a specific repository, only the user that is the owner of that repository can decide whether or not to allow the access to requested data.
Other requirements that this system must guarantee as its own, are the ability to make anonymous informations about classes of people when possible (class cardinality>1000) and to protect data against all the ecternal services that deliberately try to infer sensitive informations by over-querying a user's PDS.

3.4 Design Constraints

3.4.1 Hardware Limitations

As already mentioned in section 3.1.2, this project aims to propose the development of a TrackMe software-based application which aim to be a data sharing service, characterized by an internal service service DATA4Help that works as a cloud datastore service. so it does not provide for hardware interfaces or even less hardware limitations. As an application or a web site, all that is needed is a smartphone from which to download and use TrackMe app or a PC from which to access the web site associated with TrackMe. Of course, it will also be necessary to have an adequate internet connection.

3.5 Software System Attributes

3.5.1 Reliability

As known by the theory, reliability modelling predicts the probability that a system will not experience unplanned downtime. In order to achieve a good level of reliability we expect that TrackMe will be avilable 24 hours a day, 7 days a week without any interruption or downtime.

3.5.2 Availability

Availability modeling predicts the probability that the system is working properly and, in order to improve it we want to improve reliability of the individual components and ability of TrackMe to respond quickly and effectively to faults. We know that the availability of the entire system is as strong as the weakest component so we have to analyze each component in order to obtain a good score in terms of availability of the system.

3.5.3 Security

As any software application that has to deal with sensitive information concerning its users, even in the case of TrackMe it is essential that proper systems are set up in order to guarantee the most complete data security. In the specific case of TrackMe, in addition to users' login credentials and system data, also personal data concerning users' lives, such as health data and location data, are managed and shared. Therefore, in the case of TrackMe it is even more necessary that such data must be properly preserved through encryption techniques to ensure their protection and thus protect the users themselves.

3.5.4. Maintainability

TrackMe is intended to be flexible, easy to maintain and also capable to facilitate addition of new features and options.

3.5.5. Compatibility

TrackMe is designed to be both a mobile application that can operate both in the Android and iOS environment, and as an online service that can be reached by any device. The goal is indeed to make TrackMe compatible with the largest possible number of devices and therefore available to a wider number of users.

4. Formal Analysis Using Alloy

In this section, the Alloy model is given. In the definition of the model, all the system constraints defined in the previous sections are declared and respected. More precisely, the main constraints that have been made valid for the system are as follows:

- a data belonging to a user cannot be owned by another user at the same time, as it is a personal data related to a specific individual.

- Each user can define a single set of preferences regarding the access of their data and same set of preferences can not be associated with two distinct individuals.

- When a third party requests access to the data of an individual, before transmitting the notification to the user, TrackMe checks the preferences expressed by him and in case it already emerges from these that the transmission is not feasible, no report is sent to the user concerned, while a Warning is sent to the requestig third party.

- If the preferences expressed by the user do not reveal any impediment to trensmission, TrackMe notifies the user of the reguest by sending him a notification. The user can now decide for himself whther or not to allow the transmission of his data.

- If a third party requests data about a group (class) of individuals, TrackMe allows access to such data only if possible to anonymize them, that is, for semplicity, if the number of class members is greater than 1000. However, for semplicity, in the deefinition of this model has been used a threshold of 2 instead of 1000 (this simplification would certainly not lead to a secure system for the privacy of users in reality, but has been placed exclusively to treat the problem more easily).

4.1 ALLOY MODEL

```
open util/integer
abstract sig  Bool {}
sig True extends Bool{}
sig False extends Bool{}


sig ThirdParty {
      queriesToIndividuals: set QueryToSingleUser,
      queriesToGroup: set QueryToGroupOfUsers,
}


sig QueryToGroupOfUsers{
      mittent: one ThirdParty,
      receivers: one Class,
      dataDemanded: set Data,
      accepted: one Bool
}

--a query by a ThirdParty in order to access some data
sig QueryToSingleUser {
      mittent:one ThirdParty,
      dataDemanded: set Data,
      receiver: one User,
      accepted: one Bool,
}


sig Class {
      members: set User,
      classType: one ClassType
}


abstract sig ClassType{}
one sig Over70People extends ClassType{}


sig User {
      personalData: set Data,
      preferences: one D4HPreferences,
      notifications: set UserNotification,
      receivedQueries: set QueryToSingleUser
}


abstract sig Data {
      owner: one User,
      dataType: one DataType,
      accessible: one Bool
}


sig DataType{}
sig HealthData extends DataType {
      healthStatus: one HealthStatus
}
sig LocationData extends DataType {
      locations: one  Location
```

```
}


sig Location {
      -- latitude
      coordX: one Int,
       -- longitude
      coordY: one Int
}

sig  HealthStatus{}

--notification reaches a User, when a ThirdParty asks for his data
sig UserNotification {
      notified: one User,
      queryReceived: one QueryToSingleUser,
      confirmQuery: one Bool
}


sig D4HPreferences {
      user: one User,
      --data defined by the User as accessible by third parties
      accessibleData: set Data,
      --data defined by the User as not accessible by any third party
      unaccessibleData: set Data,
      --third parties which the User defines as allowed to access some data
      allowedThirdParties: set ThirdParty,
      --third parties which the User defines as not allowed to access some data
      notAllowedThirdParties: set ThirdParty
}


fact DataUserConnection {
      all  u: User | all  d: Data | ( d in u.personalData implies  d.owner = u)
and
      (d.owner = u implies  d in u.personalData)
}


fact NoTwoUsersToTheSameData {
      all d1: Data | no disj u1,u2:User | d1.owner= u1 and d1.owner= u2
}


fact UserPreferencesConnection {
      all u1: User | all p1: D4HPreferences | ( u1.preferences= p1 implies
p1.user = u1) and
      (p1.user= u1 implies u1.preferences= p1)
}


fact NoTwoUserToSamePreferences {
      all p1: D4HPreferences | no disj u1, u2: User | u1.preferences=p1 and
u2.preferences=p1
}


fact NoTwoPreferencesToSameUser {
      all u1: User | no disj p1, p2: D4HPreferences | p1.user= u1 and p2.user=u1
}
```

```
fact DataAccessibleD4HPreferencesConnection {
      all u1:User | all p1: D4HPreferences | all d1: Data | (p1= u1.preferences
and d1 in p1.accessibleData) implies
      (d1 in u1.personalData and d1.accessible=True)
}


fact DataUnaccessibleD4HPreferencesConnection {
      all u1:User | all p1: D4HPreferences | all d1: Data | (p1= u1.preferences
and d1 in p1.unaccessibleData) implies
      (d1 in u1.personalData and d1.accessible=False)
}

--each data defined in a D4HPreferences setting can not be owned by two
different users,because each set of preferences refers to only one user and only
to his data
fact DataInPreferencesReferToOnlyOneUser{
      all p1: D4HPreferences| all d1: Data| no disj u1,u2: User |
      (d1 in p1.accessibleData and d1 in u1.personalData and d1 in
u2.personalData)
      or
      (d1 in p1.unaccessibleData and d1 in u1.personalData and d1 in
u2.personalData)
}


fact QueryToSingleUserUserConnection {
      all u1: User | all q1:QueryToSingleUser | (q1.receiver = u1 implies q1 in
u1.receivedQueries) and
      (q1 in u1.receivedQueries implies q1.receiver=u1)
}


fact UnacceptedQueryToSingleUser {
      all q1: QueryToSingleUser | q1.accepted = False iff
      (q1.dataDemanded in q1.receiver.preferences.unaccessibleData or
      q1.mittent in q1.receiver.preferences.notAllowedThirdParties or
      (some n1: UserNotification | n1.notified= q1.receiver and
n1.queryReceived= q1 and n1.confirmQuery = False))
}


fact AcceptedQueryToSingleUser{
      all q1: QueryToSingleUser | q1.accepted = True iff
      (q1.dataDemanded in q1.receiver.preferences.accessibleData and
      (some n1: UserNotification | n1.notified= q1.receiver and
n1.queryReceived= q1 and n1.confirmQuery = True))
}

fact AcceptedQueryToGroupOfUsers {
      all q1: QueryToGroupOfUsers| q1.accepted=True iff
(#q1.receivers.members>1000)
}


fact UnacceptedQueryToGroupOfUsers {
      all q1: QueryToGroupOfUsers| q1.accepted=False iff
(#q1.receivers.members<=1000)
}
```

```
fact DataAccessibleAndNotAccessibleAtTheSameTime {
      all p1: D4HPreferences |no d1: Data |  ((d1 in p1.accessibleData) and (d1
in p1.unaccessibleData))
}


fact QueryToSingleUserGenerateUserNotification {
      all q1: QueryToSingleUser | all u1:User | u1= q1.receiver implies
       (one n1: UserNotification| n1.notified = u1 and n1.queryReceived = q1 and
q1.accepted=n1.confirmQuery)
}


fact NotificationUserConnection {
      all u1: User | all n1: UserNotification | (n1.notified = u1 implies n1 in
u1.notifications) and
      (n1 in u1.notifications implies n1.notified=u1)
}


--if a third party makes a query to a single user,the query is not about data
concerning another user
fact DataDemandedInQueryToSingleUserAreOwnedByTheSameUserWhoReceivesQuery{
      no q1: QueryToSingleUser | q1.dataDemanded.owner != q1.receiver
}


--if a third party makes a query to a class of users,the query is about data
concerning only the data about users who are members of this class
fact DataDemandedInQueryToGroupOfUsersAreOwnedByGroupMembers{
      all q1: QueryToGroupOfUsers | no d1:Data | (d1 in q1.dataDemanded) and
(d1.owner !in q1.receivers.members)
}


pred show () {
      one d1:Data | d1.dataType = HealthData and (one d2:Data |  d2.dataType =
LocationData)
      one d1:Data | d1.accessible = True and (one d2:Data |      d2.accessible=
False)
      one t1: ThirdParty | #t1.queriesToIndividuals=1 and  #t1.queriesToGroup=0
      one c1: Class | one u1: User | one q1: QueryToSingleUser|  c1.members=u1
and q1.receiver!=u1
}


run show for 2 but 1 QueryToGroupOfUsers, 1 QueryToSingleUser, 1 Class, 1
ClassType, 1 Location,1 HealthStatus, 1 UserNotification
```

4.2 World generated by the Alloy Model

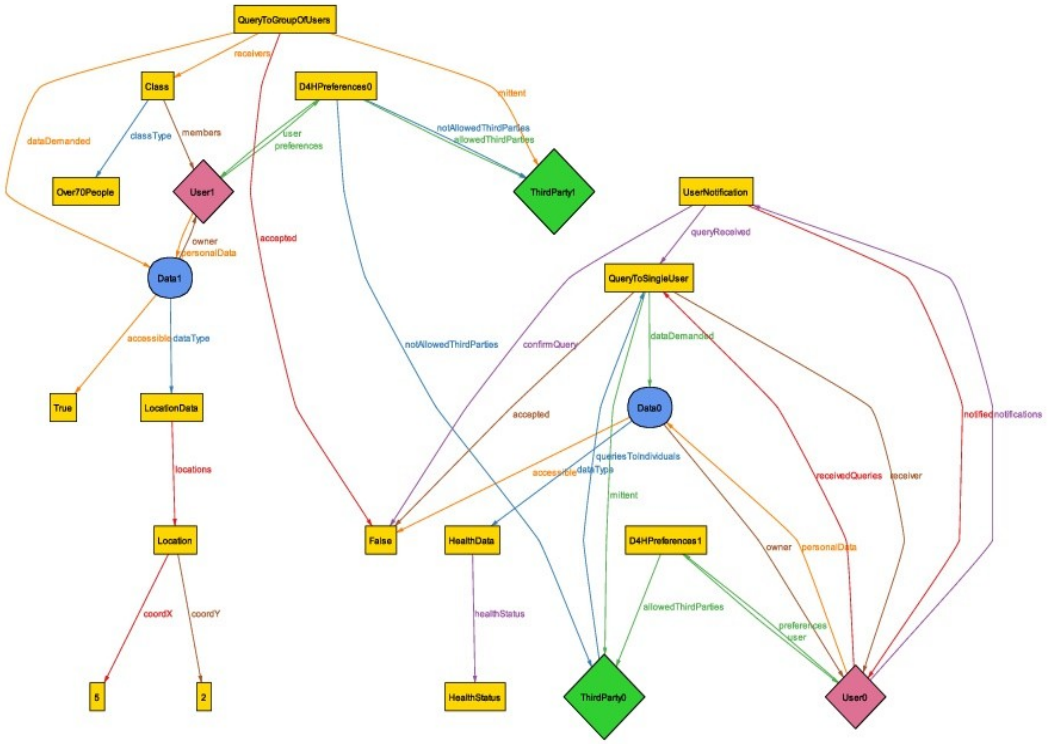

Figura 6: World generated by the alloy model

5. Reference Documents

- Slides - “World and Machine.pdf”
- Slides - “UML for Requirements Engineering.pdf”
- The Unified Model Language (https://www.uml-diagrams.org/)
- De Montjoye Y-A, Shmueli E, Wang SS, Pentland AS (2014) openPDS: *Protecting the Privacy of Metadata through SafeAnswers*. PLoS ONE 9(7): e98790. doi:10.1371/journal.pone.0098790
- Xiangqian Dong, Bing Guo, Xuliang Duan, Yuncheng Shen, Hong Zhang, ,DSPM: *A Platform for Personal Data Share and Privacy Protect Based on Metadata*, College of Computer Science Sichuan University Chengdu, China
- Laura Dragan, Markus Luczak-Roesch, and Nigel Shadbolt: *Understanding Personal Data as a Space - Learning from Dataspaces to Create Linked Personal Data*, University of Southampton, UK
- Slides - “Usage of alloy in RE.pdf”
- A Guide to Alloy (https://www.doc.ic.ac.uk/project/examples/2007/271j/suprema_on_alloy/Web/tutorial0.php)