POLITECNICO DI MILANO

DATA4HELP PRO by TRACKME

Requirement Analysis and Specification Document

Paolo Saccani
Diego Riva
Matteo Rubiu

A.Y. 2018-19 Software Engineering 2

Abstract

It’s recognized that in the IT field changes are almost a daily issue that have a heavy impact on companies’ resources. Based on this, ***TrackMe*** has chosen its strategy for the upcoming years: a great challenge is to be as "open" to input as possible and being ready to face new opportunity quickly.

It is also clear, by now, that ductility is the most difficult business’ property to chase when a company grows, along with the increased costs of maintenance. The vision of ***TrackMe*** is to develop modular software components, with strong and reliable "core" foundation, and disrupt the market as a SaaS (Software-as-a-Service) provider. The services, in this way, can follow the market and satisfy the demand faster than a traditional "vertical" software solution.

This document is meant to be helpful, in the same manner, for stakeholders and developers of ***TrackMe***. The first ones acknowledged as company's investors, third parties that are interested in the services and final users.

INDEX OF CONTENTS

SECTION 1: Introduction

1.1 Purpose

Nowadays healthcare's technologies give people the opportunity to monitor their own health status parameters. People can use software and apps which help them interpret data on their own. In this way they can be aware of the progresses they're making and the effects of a care management program on their health.

There are so many devices able to observe and interpret people's data in real time (for instance Apple Watch, Samsung Gear, Xiaomi Mi Band, and so on), and this, joint to an increasing interest in fitness and healthcare, educates users not only to understand the importance of those topics, but also to participate proactively in a care management program. Added to this, the possibility of seeing the consequences of their actions in a clear and visual manner, through data representation, can influence their own health decisions.

In this document ***TrackMe*** wants to introduce and analyse its new service aggregator, called ***Data4Help Pro***. It is based on ***Data4Help***, a service that will allow 3rd parties, through data monitoring, to better show correlation between health status parameters, personal information and location: with the goal of support them making daily choices and reaching their goals. On top of that a company could use these stored data to do researches and pattern recognition on groups of people: a main example could be analysing healthcare trends for helping then the whole population.
Naturally privacy is a major concern at ***TrackMe*** so if a user does not authorize the specific use of his/her personal position and data, he/she will not be identifiable thanks to the establishment of a minimum threshold in the number of a group participants involved in a query.

User's health is a main priority for ***TrackMe***, so we decided to introduce another new service called ***AutomatedSOS***: the idea is to give a company in the field of elderly care the chance to offer a service to their clients, that consists in monitoring if their vital parameters drop below a specific value and immediately send an ambulance request for the user's position to an ambulance dispatcher.

As mentioned before, the popularity of fitness keeps growing year by year and, as a company, we are looking at it with lots of interest with the targets of expanding our business and bringing people closer to sport. This led us to imagine another service called ***Track4Run***: the aim is to offer to run organizers the tools to create events and then define the path on a map. Moreover, it will give to spectators the possibility to see the position of athletes participating in the run "live" on the map, exploiting their GPS data.

1.1.1 Goals List

To better understand the problem, are defined the following goals:

[G1]: A user can register to the service;

[G2]: A user can access its own data;

[G3]: A user can authorize the use of personal data;

[G4]: A 3rd party can register to the service;

[G5]: A 3rd party can perform queries over group of user's data;

[G6]: A 3rd party can access a specific user's data;

[G7]: A user, allowed by a 3rd party, can benefit of AutomatedSOS service.

[G8]: A 3rd party can organize running competitions;

- ⇒ [G8.1]: A 3rd party can organize run event;
- ⇒ [G8.2]: A user can enrol to the run;
- ⇒ [G8.3]: A user can see the position of a runner in the map during the run;

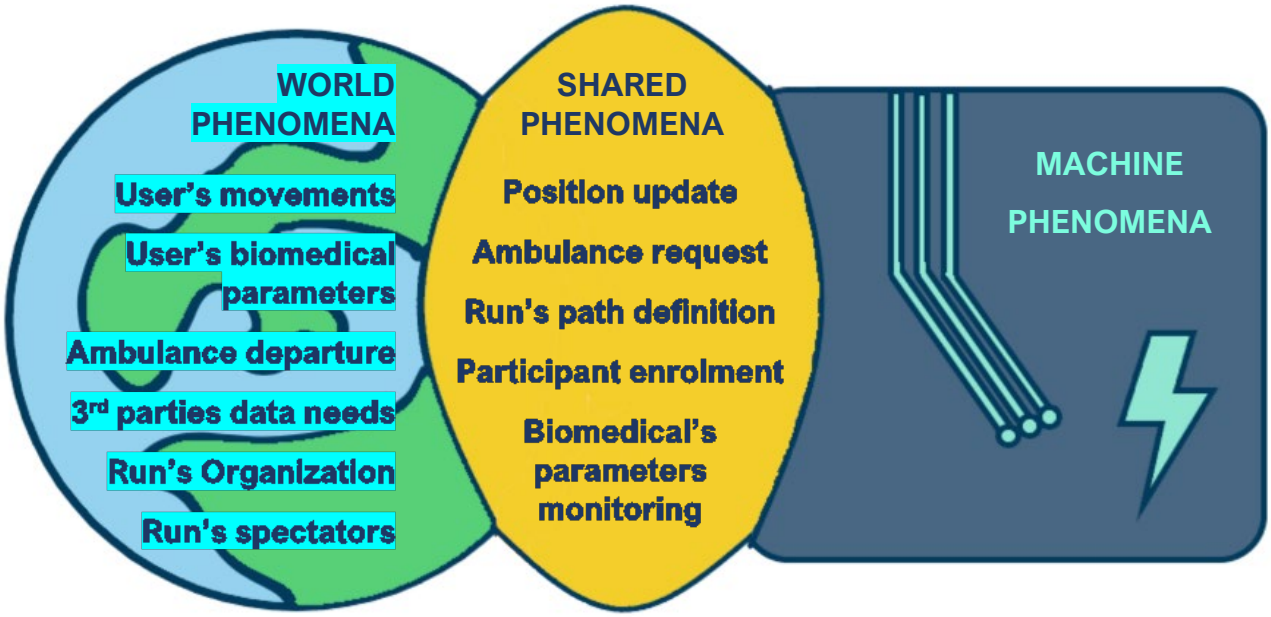


Figure 1.1: Diagram of Phenomena

1.2 Scope

To truly understand the domain in which ***TrackMe*** will operate we need to observe the phenomena that belongs to the real-world and the ones that are shared with the machine, as synthetized in *figure 1.1*.

World Phenomena:

- **User's Movement:** It is important for us to follow user's moves by using the GPS signal on the user's device.
- **User's Biomedical Parameters:** People's health status nowadays can be analysed by many devices, which can monitor the most relevant parameters.
- **Ambulance departure:** In case of health problems it may be necessary the intervention of an ambulance, reaching the location of the sick person.
- **3RD parties data needs:** Our core business is not only dealing with single individuals, but mainly with other companies that are constantly looking for information regarding people's health and geo-localization in the world.
- **Run's Organization:** 3rd parties may need to organize run events in which users can take part.
- **Run's spectators:** Spectators want to follow runners on the map for a better experience of the event.

Shared Phenomena:

- **Position Update:** The system is constantly updating the actual user's position.
- **Biomedical parameters monitoring:** Biomedical parameters are recorded and stored constantly by our system to analyse and study their connection to personal health status.
- **Ambulance request:** When user's biomedical parameters are below a certain threshold, an ambulance request is immediately sent to the dispatcher.
- **Run's path definition:** When 3rd parties organize a run competition, they need to plan it through our system, defining the path in which participants will run.
- **Participant enrolment:** Participants will be able to join a run, organized by 3rd parties, by enrolling through our system.

1.3 Definitions, Acronyms, Abbreviations

Definitions:

User: A person that registered to our service, mobile side, with his credentials.
3rd party: A company, an association or an organization that benefits of our services.
Run event: A run competition in which participants will run on a defined path.
System: All the components that permit to realize the services and satisfy the requirements.
Service Provider API: An API to embed in our system an external service, provided by a 3rd party.
Visitor: A person/company that is interested in our services, but he's not registered yet.
Client: An unidentified utilizer of the system (user or 3rd party).

Acronyms:

API: Application Programming Interface.
GPS: Global Positioning System.
RASD: Requirement Analysis and Specification Document.
VAT: Identification number for a company

Abbreviations:

[DX]: Domain assumption number x

[RX]: Requirement number x

[GX]: Goal number x

1.4 Revision History

1.5 Reference Documents

- Specification document: “Mandatory Project Assignment AY 2018-2019.pdf”
- RASD example from previous year: "RASD to be analyzed.pdf"
- Course material

1.6 Document Structure

ABSTRACT

CHAPTER 1: It present a brief introduction to the given problem, definition of goals the product must accomplish, clarification of the scopes of the phenomena and other information needed to understand completely the document

CHAPTER 2: It describes more in-depth the behaviour of the final product by using state-charts and class diagrams and define then the domain assumptions and constrains as much as target clients.

CHAPTER 3: Here are described mainly functional (and non) requirements, an initial approach to the design of user interface and its constrains, investigation of use cases and scenarios (using also sequence diagrams).

CHAPTER 4: World modelling and analysis using Alloy, it contains also a proof of consistency and an example of the generated world.

CHAPTER 5: Deepening of the effort spent by the group

SECTION 2: Overall Description

2.1 Product perspective

The software we are going to release should be fully integrated with existing platform and devices (by OEMs) through proprietary API. The core idea for this platform is to be modular, the foundation will be ***Data4Help.***

Our product, mobile side, will not have a complex user interface: mainly it is intended to be a visually pleasant aggregator of personal health data. The idea is that the individuals, after the appropriate registration (exploited further on) can:

- Synchronize a wearable device.
- See his/her profile with personal information.
- Edit his/her personal information.
- Respond to data access request by 3rd parties.
- Confirm/revoke his/her choices for privacy authorizations.
- See if the ***AutomatedSOS*** service is active for his/her account.
- See his/her biomedical and health parameters, like weight or heartrate.
- Enter ***Track4Run*** subsection and:
    - Choose to enrol to a run event from a list of possible ones.
    - Be a spectator and see, on a map, the "live" position of athletes.

The dynamic of this possible interactions is exploited more clearly in *figure 2.1*.

On the opposite if, during the registration, the 3rd party's side is chosen, then the options are the following, exploited in *figure 2.2.* Therefore, they can:

- Perform group queries, setting the desired filters and see/download the results.
    - The system will accept these if the privacy of the users is guaranteed, any request that will end with a number of users fewer than 1000 will be automatically rejected.
- Manage the list of users that have positively responded to the access request for the goal of using their data for studies or researches.

- See and edit the information on their profile.
- Observe and download single user data.
    - If he/she is qualified for the service, enable or disable AutomatedSOS.
- If interested, create a run event and then define the path on map, selecting from the accessible roads/streets.

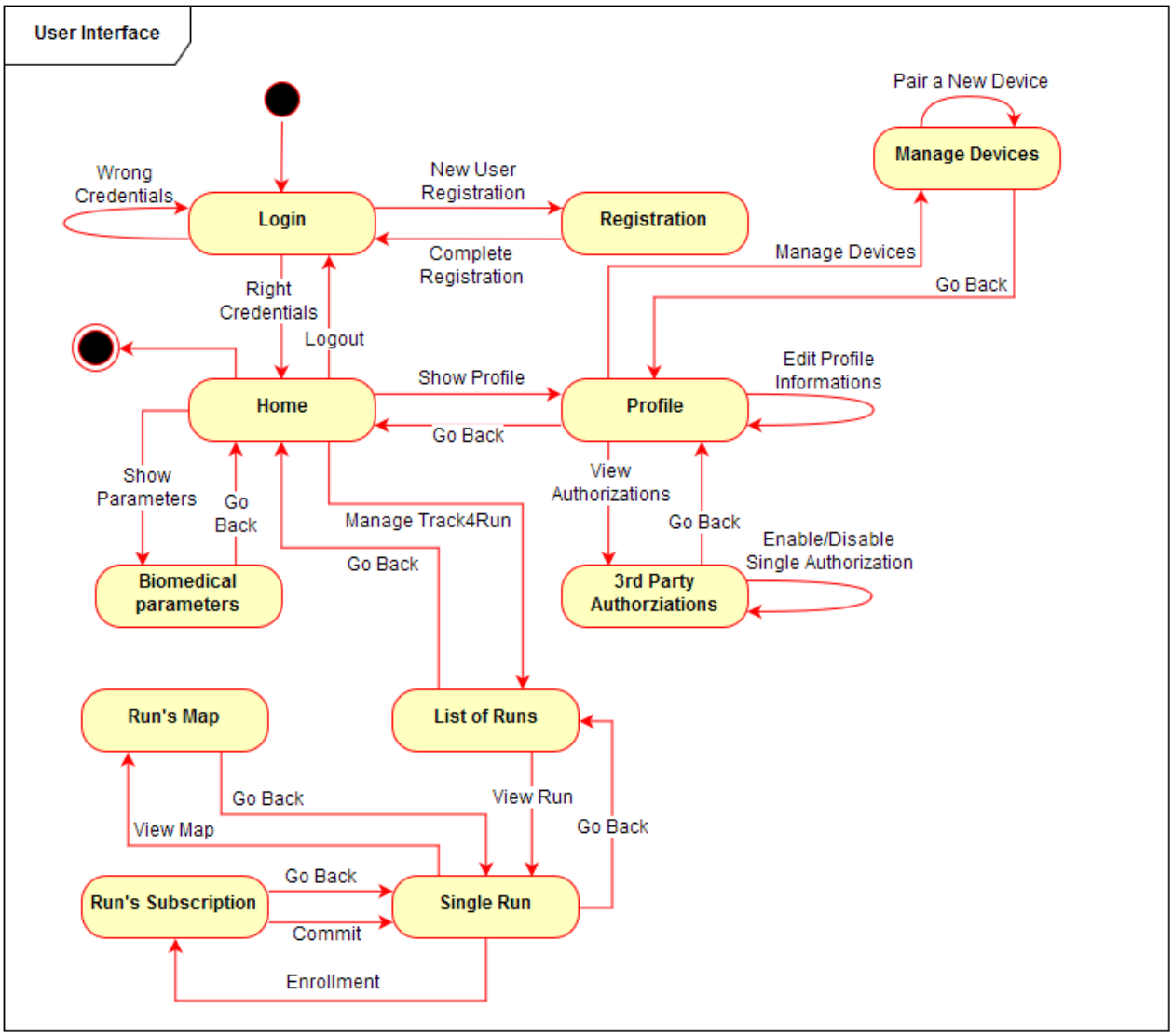


Figure 2.1: state diagram for the user

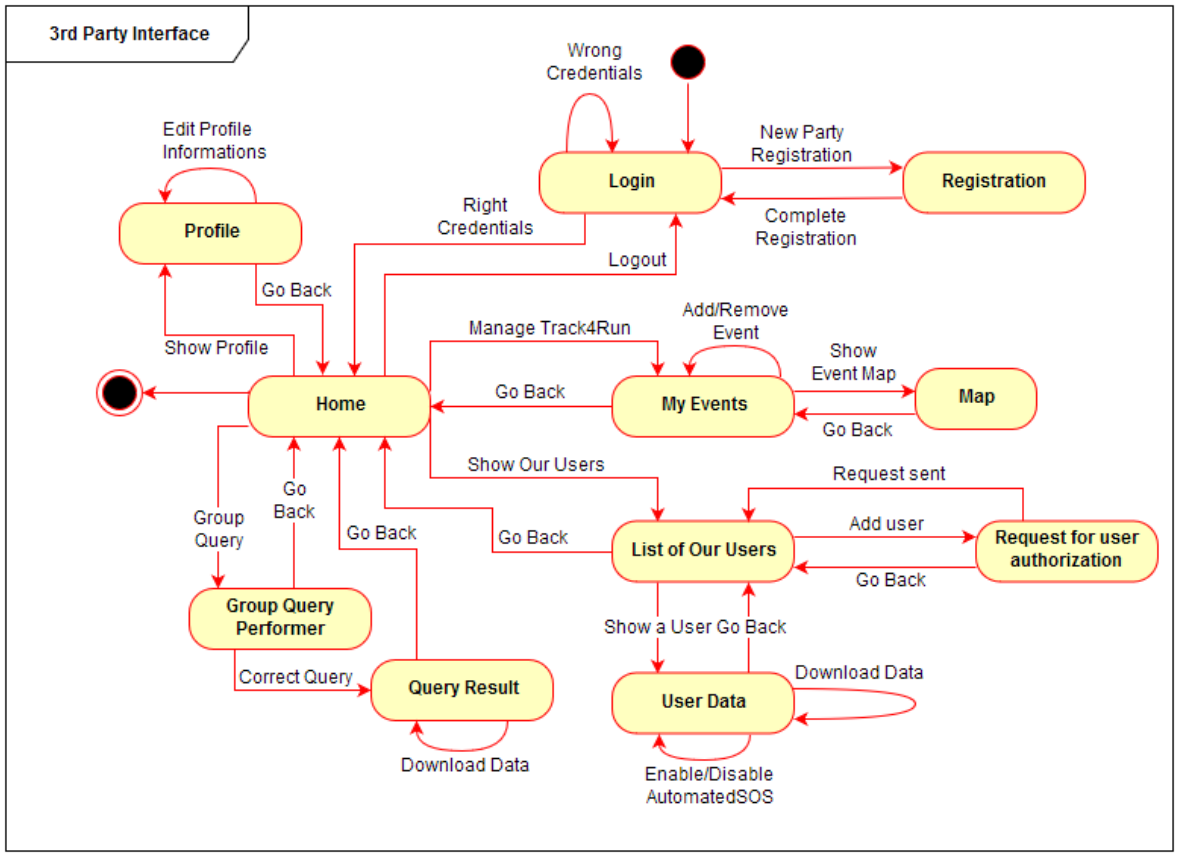


Figure 2.2: state diagram for the 3rd parties

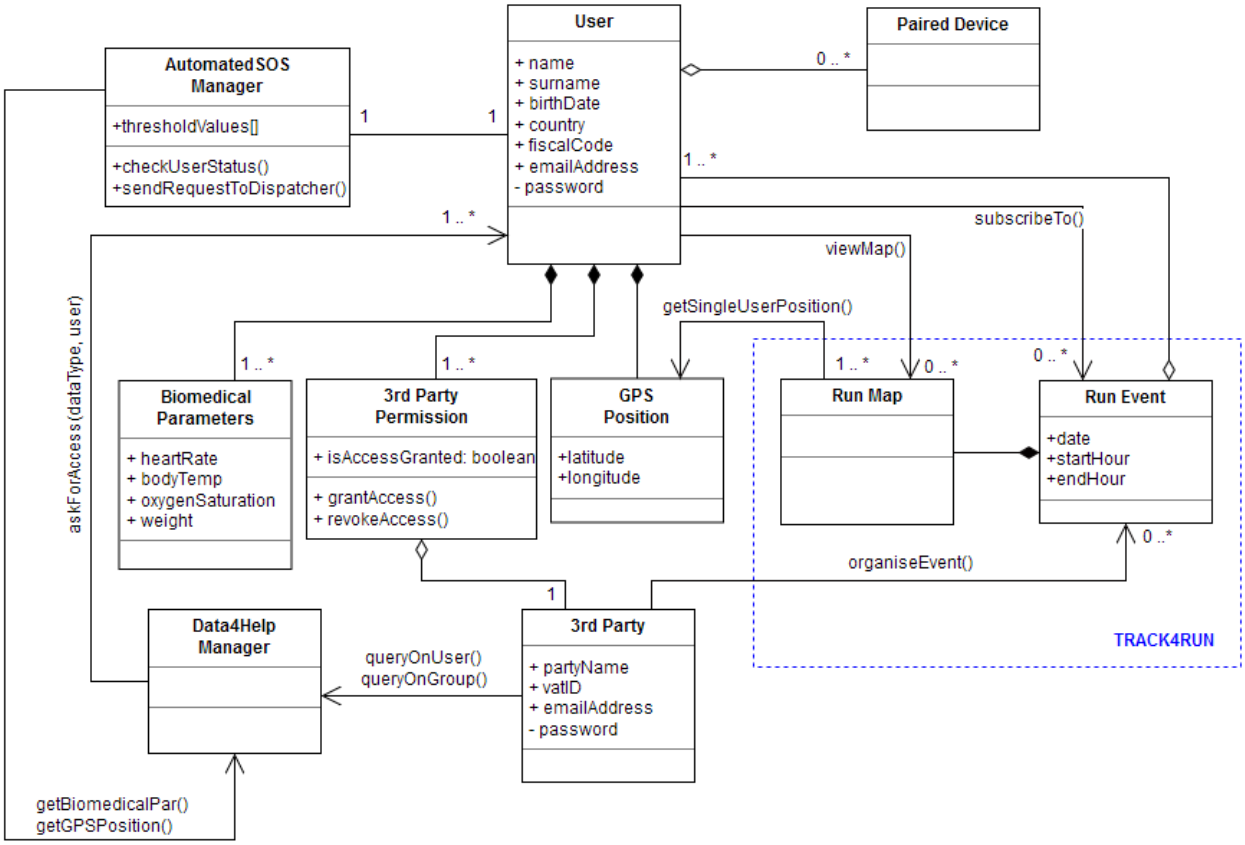


Figure 2.3: general class diagram for the system

In *figure 2.3* we represent the general structure of our system, using a class diagrams, this must be considered a high level of abstraction. Mainly it is useful to understand the interactions and the hierarchy details that have to be maintained in the real implementation. The main entities found in the diagram are the following ones:

- **User**: class that represents a person that registered to our service with his/her credentials whom is also composed of:
    - Paired device/s.
    - Biomedical parameters.
    - 3rd party permissions: each 3rd party that ask for user's data is then connected to him/her and has a Boolean access "flag".
    - GPS position: latitude and longitude catered by the smartphone.
- **3rd party**: class that represents the company able to perform operations on the users' specific/group data and organize a run event, as previously exploited.
- **Data4Help Manager**: to retrieve only the allowed data is needed a class that act as an access controller.
- **AutomatedSOS manager**: class that runs the continue checks on user's parameters, like heartrate, to immediately extrapolate GPS position and send a help request, if needed.
- **Run event** and **Run Map** utilized in the management of ***Track4Run*** service and quite self-explicative.

2.2 Product functions

The system offers to the user a great integration with existing devices and the missing ring between data interpreter/visualizer and a purely human care program. The most important requirements to achieve are:

- To guarantee the un-traceability of a user in a group query.
- To offer access to well categorized user data to 3rd parties.
- To show a user his data in a well assembled way.
- To manage a run event from enrolment to path definition.
- To "stream" a run on a map exploiting GPS data from the runners.

2.3 User characteristics

The intended customers for these services are essentially split in private users and 3rd party companies interested in generating business or knowledge from the data provided by ***TrackMe***.

The single user must be able to use a smart-phone (GPS catered), have access to the internet and have at least one wearable device that collect bio-medical data.

In a more in-depth view 3rd parties will have different characteristics based on what service they are interested:

- companies related to the bio-medical or health sector, for researches and studies
- companies in the field of elderly care, for ***AutomatedSOS***
- Especially associations dedicated to organization of sporting events for ***Track4Run***

2.4 Assumptions, Dependencies and Constraints

In this first stage of development, we make the following assumptions on the world to concentrate first on the most important aspect of the system:

[D1]: Every user has at least an internet connected device.

[D2]: Every final user has at least a GPS equipped device.

[D3]: Every user has at least one device able to catch biological data.

[D4]: Data that come from the user is assumed to be correct over a fair percentage.

[D5]: A request on group data is delivered correctly, when processed by the system.

[D6]: If an ambulance request is sent, it is always received by the dispatcher.

[D7]: An ambulance always reaches the correct location, provided by the patient.

[D8]: An organizer of a run competition can create a path of only accessible roads/streets.

[D9]: If a user enrols in a run competition, he will participate at it.

[D10]: Any runner of a specific event will follow the defined path.

[D11] Information provided by a client, during the registration, is assumed to be correct.

[D12] All policies about privacy and authorizations are compliant with GDPR regulation

SECTION 3: Specific Requirements

3.1 External interface requirements

3.1.1 User Interfaces

Figure 3.1: mobile and web-based login page

Since there are two very different type of customers for our services, respectively accessible via mobile or web application, different interfaces are needed.

Here we introduce the basic concepts for those interfaces with tentative mock-ups of the principal pages, we will then exploit the design further in the Design Document.

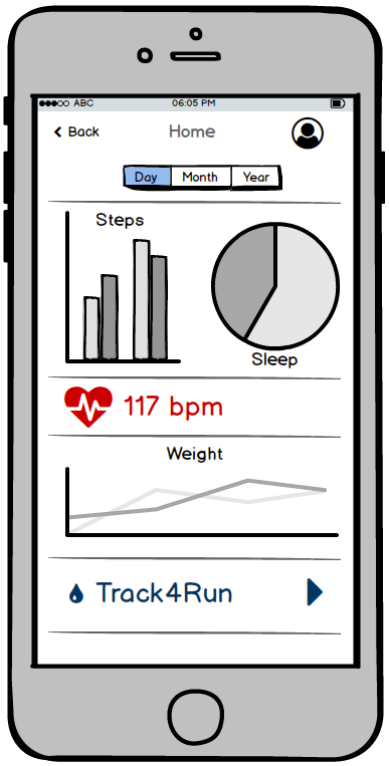


Figure 3.2: home page

In the mobile app, once signed up and paired a device, the user finds itself in the Home (figure 3.2) where he/she can see:

- A visual synthesis of the biomedical data observed by the device/s.
- The link to the Track4Run subsection.

In all the pages it is present a top bar with a representative title for better orientation through the app.

Figure 3.3: profile page

Excluding the home page, it is present on the left a back button that links the user to the previous visited page; on the right hand instead, there is the user icon that brought the user to the profile page (figure 3.3) where he/she can:

- Watch and manage his/her personal information.
- Check connected device/s status/es.
- Reach the Third Parties Authorizations page.

This last one, as it is exposed in *figure 3.4* on the right, consists in an extensible list of 3rd parties that requested the access to the user's data: the first time this happens it is notified to the user, which is then able to confirm or revoke this access to a specific company via a switch button any moment.

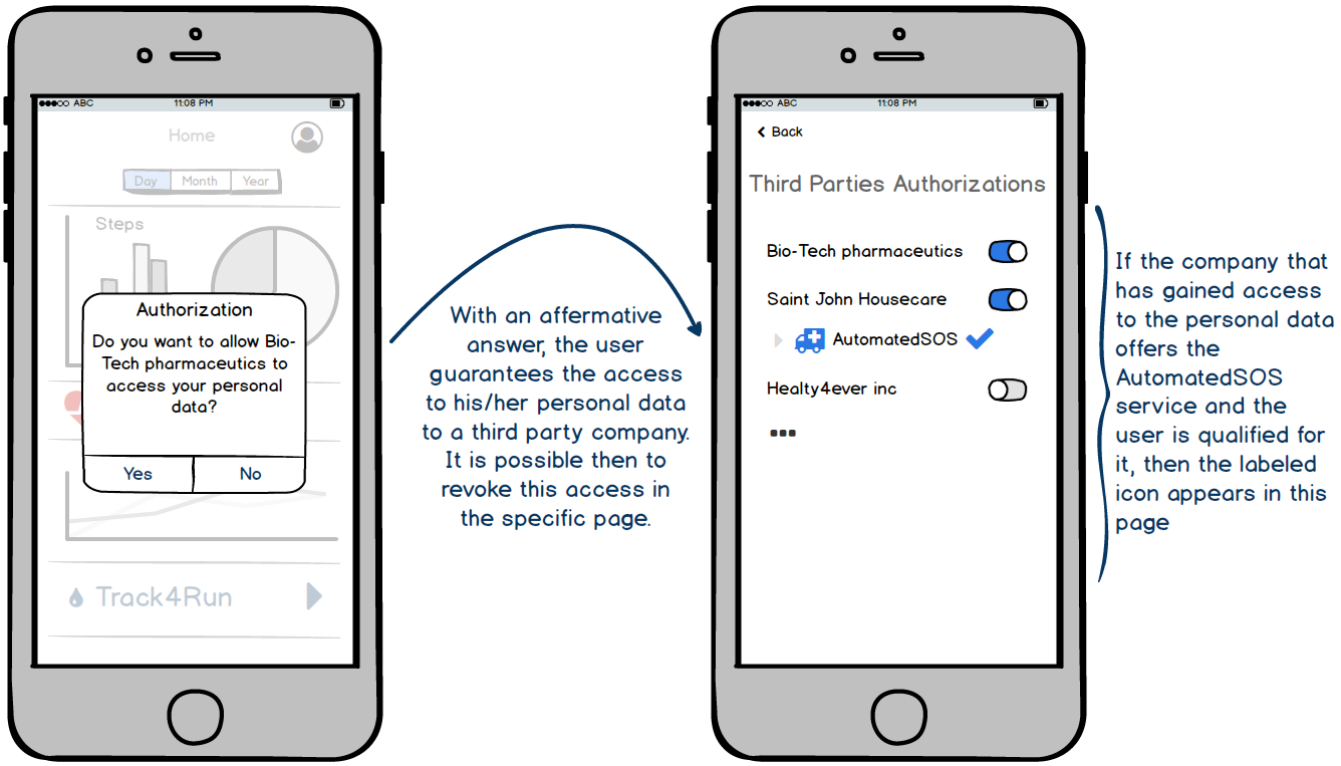


Figure 3.4: notification for 3rd party request on data access and authorizations' page

There is then the section dedicated to Track4Run where the user will find a list of nearby run events to which he/she can subscribe. If the user wants, he/she can be also a spectator in an ongoing run and watch on a map the runners' progress and the live ranking, clicking on the event, like in *figure 3.5.*

Figure 3.5: run spectator page

The 3rd party interfaces, on the opposite side, are web-oriented and reachable through any modern browser that supports security encryption.

After the registration, that it is elaborated further on, the company operator finds an intuitive home page where are situated the links to:

- The section in which can be set the filters to perform group queries, in form of checkboxes, sliders or input areas (figure 3.6).

- The section in which are visible all the singular users, whom confirmed the specific access to his/her data, as a scrolling list.
- The track4Run section for the management of run events.

In the upper left corner, it is always visible the menu icon, that any moment can bring the operator to the home.

Exploding, for example, the track4Run section we can analyse the page where it is possible to define the path of a run (figure 3.7), once created the event.

On the left there is a map in which it is possible to select the roads for the run, meanwhile on the right are shown more in-depth information of them and if they are picked or not (starting from the nearest to the first selection).

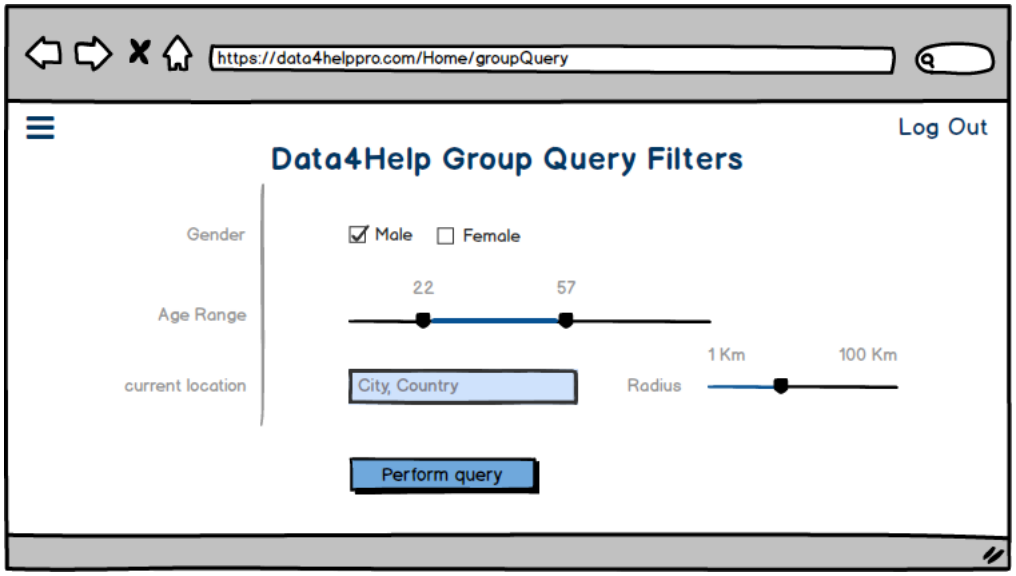


Figure 3.6: query filters setting page

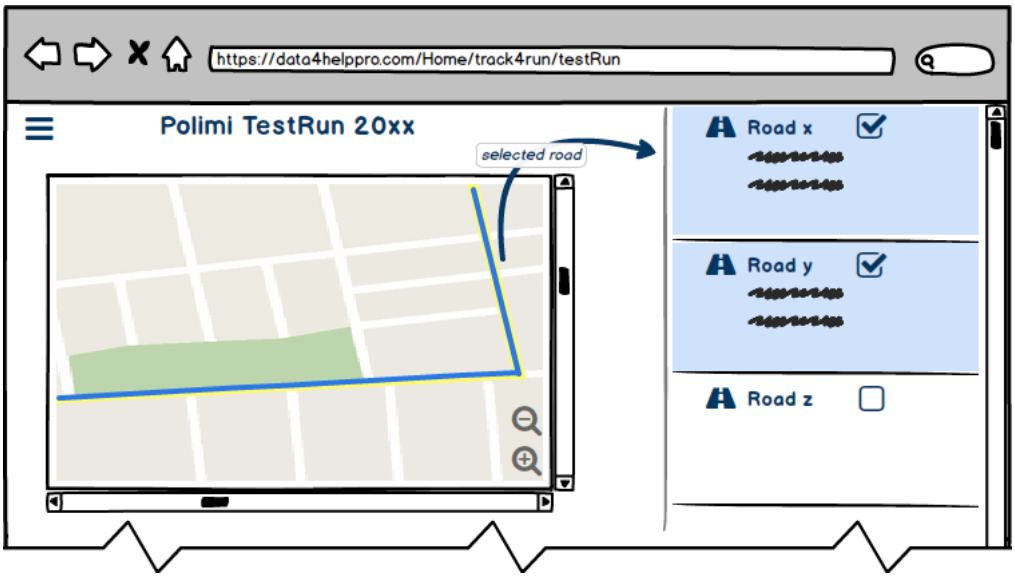


Figure 3.7: path definition page

3.1.2 Software and Communication Interfaces

The mobile side of our application must be cross-platform, capable of reading and interpreting data from the major wearable manufacturers, through their API.
The same goes for the map service utilised in Track4Run, it will be chosen the most open provider, with API well documented and possibly ready to provide the tools for this project, limiting the adaptation to the problems.
It will have to establish secure connection from the smartphone app or the web application to the system's servers, via a reliable protocol like HTTPS.

3.2 Functional requirements

3.2.1 Goals satisfiability

[G1]: A user can register to the service.
[R1]: User must fill up the "user's registration form" providing required data.
[R2]: Fiscal code and synchronization with the first device are required for registration.
[R3]: The app saves user's personal information.
[R4]: The credentials that a user has to provide consist in: e-mail and password.
[R5]: Once received user's registration, the system must send a verification mail at the provided e-mail.
[D1]: Every user has at least an internet connected device.
[D11] Information provided by a client, during the registration, is assumed to be correct.

[G2]: A user can access its own data.
[R6]: The user's app reads data from wearables and interprets them correctly through their API.
[R7]: The data-types should be correctly tagged and identified (on a smartphone).
[R8]: The system must be accessible via mobile app.
[R9]: The app must communicate and share information with the system.
[D1]: Every user has at least an internet connected device.

[D3]: Every user has at least one device able to catch biological data.

[D4]: Data that come from the user is assumed to be correct over a fair percentage.

[D11] Information provided by user, during the registration, is assumed to be correct.

[G3]: A user can authorize the use of personal data.

[R10]: The mobile app must save user's personal information and 3rd party authorizations.

[R11]: Whenever a 3rd party wants to access personal data, the specified user must receive on his app a pop up notification that asks him to grant the authorization.

[R12]: Whenever a third party wants to access personal data, the company is added in the "3rd Party Authorization" list.

[R13]: The system allows anyone to access a specific user's data only if it is granted in his "3rd Party Authorization" list.

[D12] All policies about privacy and authorizations are compliant with GDPR regulation

[G4]: A 3rd party can register to the service.

[R14]: The system must be accessible to 3rd parties via web application.

[R15]: User must fill up the "3rd parties registration form" providing the required data.

[R16]: A 3rd party must obtain access to all ***TrackMe*** services.

[R17]: VAT number is required for registration.

[R18]: The credentials that a 3rd party has to provide consist in: e-mail and password.

[R19]: Once received 3rd party registration, the system must send a verification mail at the provided e-mail.

[D11] Information provided by a client, during the registration, is assumed to be correct.

[G5]: A 3rd party can perform queries over group of users' data.

[R20]: Any request must be sent to ***TrackMe*** servers to be processed.

[R21]: The system must provide the correct data.

[R22]: The system will accept any request for which the number of individuals, whose data satisfy the request, is higher than 1000.

[R23]: If the request is invalid the system must not approve it.

[D4]: Data that come from the user is assumed to be correct over a fair percentage.

[D5]: A request on group data is delivered correctly, when processed by the system.

[G6]: A 3rd party can access a specific user's data.

[R24]: A 3rd party must provide the fiscal code of the user of whom they are interested in accessing the data.

[R25]: The access is granted only if the specific user accepts the authorization.

[R26]: Once obtained the permission, 3rd parties can observe data until the user revoke the authorization.

[D1]: Every user has at least an internet connected device.

[D2]: Every final user has at least a GPS equipped device.

[D3]: Every user has at least one device able to catch biological data.

[D4]: Data that come from the user is assumed to be correct over a fair percentage

[G7]: A user, allowed by a 3rd party, can benefit of *AutomatedSOS* service.

[R27]: The system provides to the qualified user the ***AutomatedSOS*** services.

[R28]: The system must be able to evaluate if the user's parameters are below a minimum health threshold.

[R29]: The system sends the GPS position of the user to an ambulance dispatcher in less than 5 seconds.

[D1]: Every user has at least an internet connected device.

[D2]: Every final user has at least a GPS equipped device.

[D3]: Every user has at least one device able to catch biological data.

[D4]: Data that come from the user is assumed to be correct over a fair percentage.

[D6]: If an ambulance request is sent, it is always received by the dispatcher.

[D7]: An ambulance always reaches the correct location, provided by the patient.

[G8]: A 3rd party can organize a run event;

[R30]: A 3rd party must be able to utilize a map, provided through external API.

⇒ [G8.1]: A 3rd party define the path of the run;

[R31]: The system must provide a map of selectable roads

[D8]: An organizer of a run competition can create a path of only accessible roads/streets.

⇒ [G8.2]: A user can enrol to the run.

[R32]: The system must provide a map of selectable roads.

[D9]: If a user enrols in a run event, he will participate at it.

⇒ **[G8.3]: A user can see the position of a runner in the map during the run.**

[R33]: The application must show correct path of a selected run.

[D2]: Every final user has at least a GPS equipped device.

[D10]: Any runner of a specific event will follow the defined path.

3.2.2 Scenarios

SCENARIO 1.

John decided to use an app that would help him to control his parameters' trend after a training session at the gym. He needs an application that can store data provided by its synchronized devices and can make those data well readable through chart and tables.

For those reason John opted to install ***Data4HelpPro*** that is a valid solution for his needs: the application support data from the most common devices offering a great visual analysis to the user.

In this way John is now able to control his parameters and to manage more personalized training sessions.

SCENARIO 2.

A hospice intends to rely on a company able to acquire basics biological parameters through devices, process and send them at the hospice's system. The goal is to give to elderly patients the opportunity to live their life at home, without continues controls and calls for general analysis.

In order to guarantee the best support in case of problems, the hospice wants to offer at patients a service that is able to control their data and determine if, for example, heartrate goes below a certain threshold.

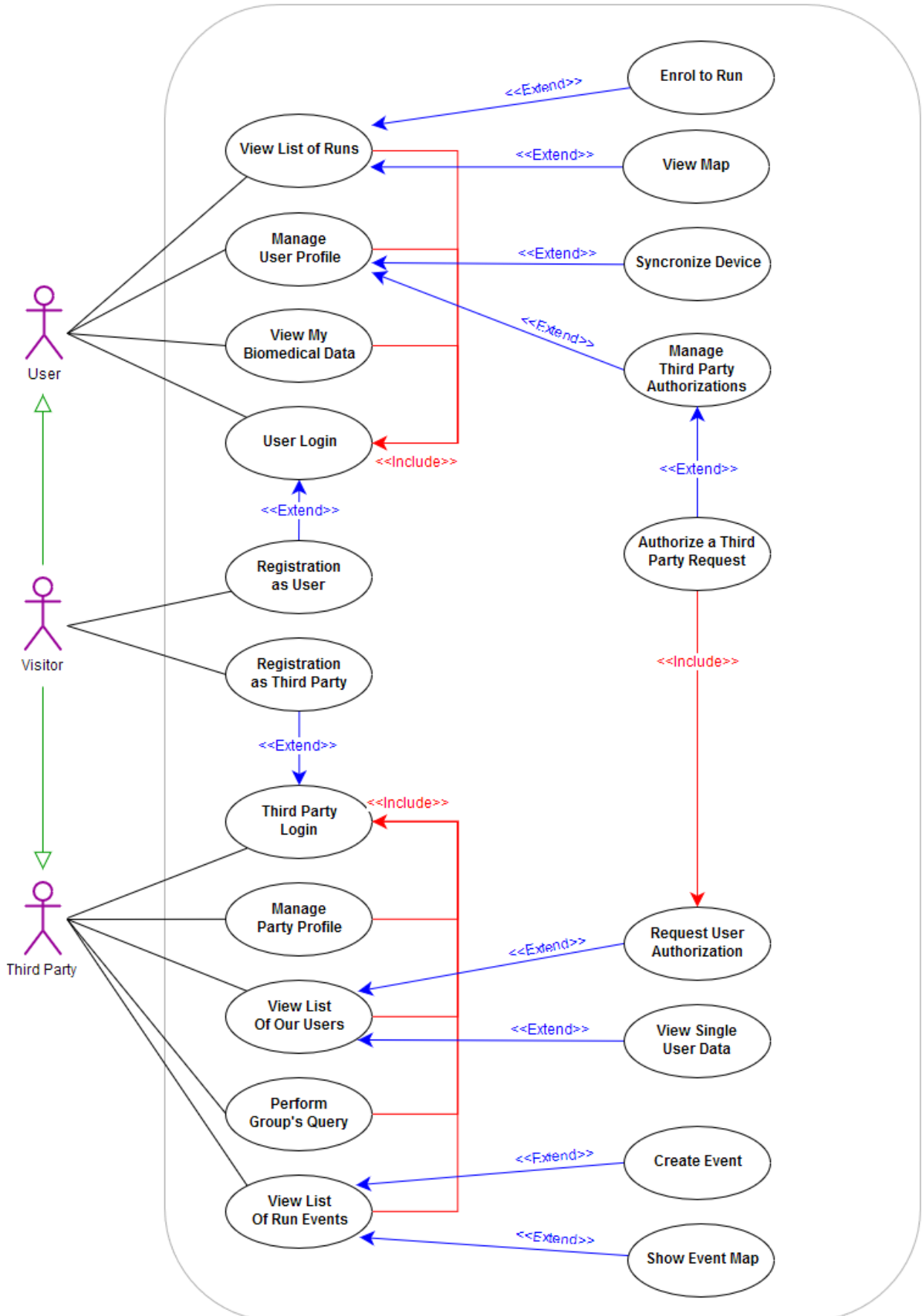
For those reason the hospice decided to adopt the ***Data4Help Pro*** solution that provide them the opportunity to access patients' data and launch an SOS request to an ambulance dispatcher which will receive the position of the person in less than 5 seconds.

SCENARIO 3.

An emergent brand in sportswear wants to organize run events for publicize his business and attract people to live together a day devoted to sport. The company needs also to access at participators' data to study how runners are distributed at the arrival of the race.

Then they decided to register to ***Data4Help Pro*** and create a new event for the successive week. In this way all the users of the app will be able to enrol or even follow runners, through their GPS signal, on a map while organizers can catch data live during the run, analysing biological parameters of participators eventually.

3.2.3 Use case diagram



3.2.4 Use case tables

REGISTRATION

ACTORS	Visitor (single user/company)
INPUT CONDITIONS	The visitor open app/browser application
FLOW OF EVENTS	1. The visitor taps on the registration button; 2. The visitor provides an e-mail and a password; 3. The visitor enters the required user/3rd party account information values and requests that the system saves (fiscal code or VAT). 3.1 If the visitor is signing up as user, the system asks him to connect his first device; 4. The system saves the data; 5. The system sends an e-mail to the visitor with a verification link;
OUTPUT CONDITIONS	Once certificated, any visitor, from now on, can log in to the application providing e-mail and password
EXCEPTIONS	• The visitor is already a client • The visitor inserts not valid information • The visitor chooses an existing email

LOG IN

ACTORS	Client (single user/3rd party)
INPUT CONDITIONS	The client is correctly registered
FLOW OF EVENTS	1. The system requests that client enters e-mail and password; 2. The client provides an email and a password; 3. The system validates credentials
OUTPUT CONDITIONS	Client is redirected to his homepage
EXCEPTIONS	• Invalid credentials

MANAGE PROFILE

ACTORS	• User • 3rd party
INPUT CONDITIONS	The client is already logged to the system
FLOW OF EVENTS	1. The client tap on “Profile” button; 2. The system provide a new page with a list of data saved on the system during registration (pt. 3), the possibility to change credentials (pt. 4) a section linked to “3rd party authorizations” (use case “3RD PARTY AUTHORIZATIONS” only for users) 3. Client chooses which information he wants to modify and then taps on “edit profile information” button and change values; 4. To change credentials Client taps on “modify password” (pt. 4.1) or “modify e-mail” (pt. 4.2) 4.1. The system provides a new page in which the client has to commit his e-mail, his old password and his new one twice. Then he will receive a validation e-mail to complete the change. 4.2. The system provides a new page in which the client has to insert current e-mail, the new one twice and his password. Then he will receive a verification e-mail on the new one. 5. At the end of any modification the client can save changes clicking on “save changes” button to update the system and will be finally redirect to the homepage
OUTPUT CONDITIONS	• Client will successfully change his profile or credentials • Client exits form “manage profile” coming back to homepage • The system provides a pop-up warning that changes have not been saved
EXCEPTIONS	• Some sections are not filled up • The client inserts not valid information • Changes have not been saved

THIRD PARTY AUTHORIZATIONS

ACTORS	User
INPUT CONDITIONS	a) The user has accessed “profile” page b) The user receives a notification of a new 3rd party request
FLOW OF EVENTS	a) 1. User taps on “3rd party authorizations” to enter in his privacy section; 2. The system provide a new page with a list of requests and the status of AutomatedSOS, activated or not; 3. The user can mark or unmark the agreement to process personal data by any single 3rd party; b) 1. The system provides a pop-up on the current page requesting the authorization to access at personal data from 3rd parties; 2. User pick “yes” to authorize the process of data, “no” to refuse the consensus 3. The system adds the request into the list in the “3rd party authorization” page
OUTPUT CONDITIONS	• User has successfully changed his agreements • User exit from “3rd party authotization” page coming back to the homepage • Pop-up closes automatically, and user continue in the current page
EXCEPTIONS	

VIEW BIOLOGICAL DATA

ACTORS	User
INPUT CONDITIONS	User is already logged in
FLOW OF EVENTS	1. User taps on “View Biometrics Data” to enter in the corresponding section; 2. The system provides a new page with a list of biological data acquired by user’s devices;
OUTPUT CONDITIONS	User exit from “Biological Data” page coming back to the homepage
EXCEPTIONS	

MANAGE CONNECTED DEVICES

ACTORS	User
INPUT CONDITIONS	• User is already logged in • The user is accessed in “profile” page
FLOW OF EVENTS	1. User taps on “Paired Devices” from “Profile” page to enter in the corresponding section; 2. The system provides a new page with a list of devices that have been connected at least one time. At the top of the list there are devices currently synchronized; 3. The system give the opportunity to add (pt.4-5) and delete (pt. 6) devices in the list; 4. The user taps on “add new device” button to search new devices that will be shown at the end of the list and select the chosen one; 5. The system synchronizes with the device and add it in the list 6. The user taps on “remove” icon next to a listed device to delete it from there.
OUTPUT CONDITIONS	User correctly synchronized his new device
EXCEPTIONS	Synchronization failure

TRACK4RUN (user)

ACTORS	User
INPUT CONDITIONS	User is already logged in
FLOW OF EVENTS	1. User taps on “Track4Run” from the homepage to enter in the corresponding section; 2. The system provide a new page with a list of current runs and a list of planned events; 3. The user can click on a specific ongoing run (pt.4) or on a specific event (pt.5); 4. The system opens a new page with a map of the path, the current position and generalities of each runner; 5. The system opens a new page with information about the selected event and the possibility to join it tapping on “enrol” button

OUTPUT CONDITIONS	• User exit from the map coming back to “Track4Run” page • User correctly enrol a run
EXCEPTIONS	

VIEW USER DATA

ACTORS	3rd party
INPUT CONDITIONS	• 3rd party is already logged in • At least one user has given his authorization
FLOW OF EVENTS	1. 3rd party tips on “view user data” form the homepage; 2. The system provides a new page with a list of users that have granted the processing of personal data. The 3rd party can view single user’s data (pt. 3-5) or request a new user’s authorization (pt. 6-9) 3. The 3rd party can click on a specific user; 4. The system opens a new page with all data provided by the selected user; 5. The 3rd party can finally download those data clicking on the corresponding button; 6. 3rd party tips on “new request” button; 7. The system opens a new page with the condition of the service and a blank where the 3rd party has to insert the fiscal code of the chosen user; 8. 3rd party sends the request clicking on “Accept and send request” 9. In case of positive answer, 3rd party can decide to enable “AutomatedSOS” service to that user.
OUTPUT CONDITIONS	• Data successfully downloaded • 3rd party exits from page coming back to the previous page • Request successfully sent • Operation aborted
EXCEPTIONS	• Provided fiscal code doesn’t exists in system’s DBMS

PERFORM GROUP QUERIES

ACTORS	3rd party
INPUT CONDITIONS	3rd party is already logged in
FLOW OF EVENTS	1. 3rd party tips on “compile new group query” button; 2. The system provide a new page with a list of filters usable for queries (age range, current location, country); 3. 3rd party disposes his filters and commit the request;
OUTPUT CONDITIONS	• Requested is accepted and user is added in the list • Request is refused • 3rd party comes back to the list of users aborting the operation
EXCEPTIONS	

TRACK4RUN (3rd party)

ACTORS	3rd party
INPUT CONDITIONS	3rd party is already logged in
FLOW OF EVENTS	1. 3rd party taps on “Track4Run” from the homepage to enter in the corresponding section; 2. The system provides a new page with a list of current runs and a list of planned events; 3. The user can click on a specific run (pt.4) or on the “create a new event” button (pt.5-6); 4. The system opens a new window with a map of the path, the current position and generalities of runners; 5. The system opens a new page providing a form with a map where organizers can specify data, location and the path of the event; 6. Once completed the form, 3rd party has to confirm the event;
OUTPUT CONDITIONS	• Exit from the map, back to “Track4run” page • An event has been correctly created
EXCEPTIONS	• Invalid form compilation

3.2.4 Sequence diagrams

The following are the high-level sequence diagrams for the main Use Cases. They show the interaction between the methods and the objects related to the Class Diagram. To make the dynamic even clearer we exploited some of the internal component of the system. The purpose is to better show how the components interact.

3.3 Performance requirements

3.3.1 Hardware limitations

The mobile side application to function correctly must satisfy the following requirements:

- Touchscreen or generic input device
- GPS
- Internet connection

3.4 Design constraints

- The bio-medical data shown in the mobile app are divided in multiple views: daily, monthly and yearly. In this way the user can have different type of deepening into his/her progresses or habits.

3.5 Software system attributes

3.5.1 Reliability and Availability

The system must guarantee a 24/7 service to the user because with the ***AutomatedSOS*** service there are lives that we can save with a reliable system. If we are going to borrow the servers from a cloud services provider, we must strongly check if they will support redundancy and clustering.

3.5.2 Security

At ***TrackMe*** security has to be a primary concern since we operate with a large amount of personal data that could expose our user to real threats. The data storage should take place through state of the art DBMS or through best-in-business cloud services provider.

SECTION 4: Formal analysis using Alloy

4.1 Introduction

In this section we we'll analyse a subset of our system: ***Track4Run.***

We have modelized the interactions between the users, the 3rd parties and ***Track4Run.***

In particular we focused on modelling the interactions between the following entities: Coordinates, Positions, Dates, Users, Parties, Run Events and Paths.

To simulate the model in an acceptable time we made some simplifications with Integers:

- Data Format: we avoided the "month" attribute and instead we used just a "day" attribute that simply represents the day number in a year *(0 < day < 365).*
- Coordinates: longitude and latitude are modelled as floats with an integer and a decimal part (both are modelled as Integers, but the decimal part is limited).

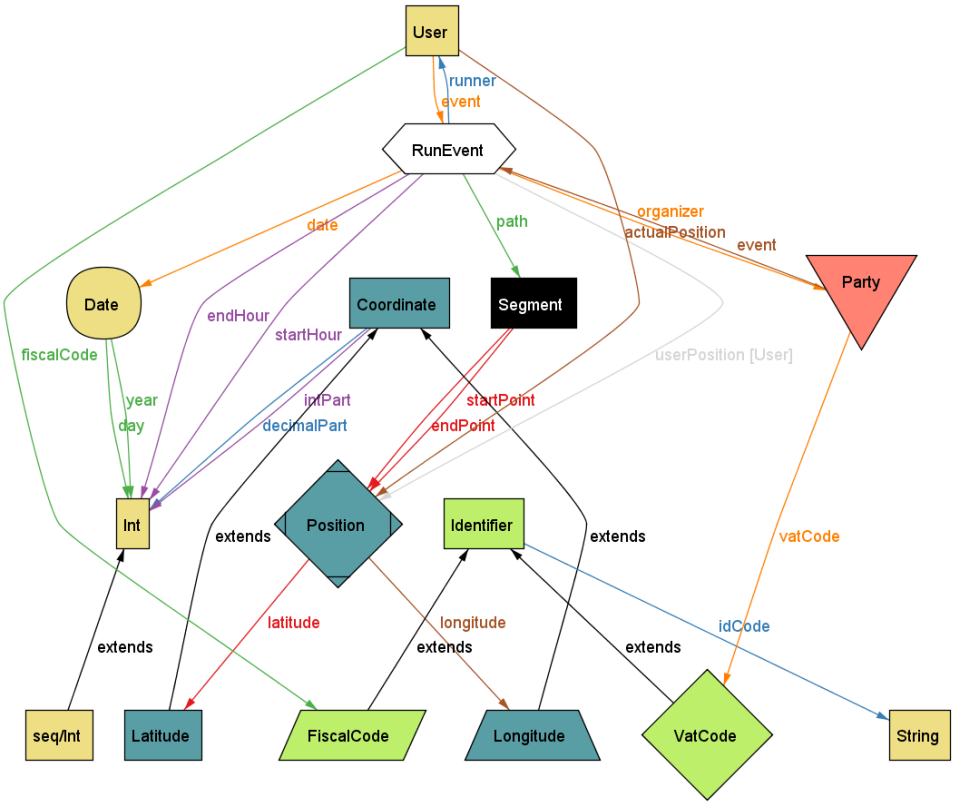


Figure 4.1 Metamodel Diagram

4.2 Signatures

```
//SIGNATURES

abstract sig Coordinate{
	intPart: one Int,
	decimalPart: one Int
}{ decimalPart >= 0 }

sig Latitude extends Coordinate {}
{ intPart >= -90 and intPart <= 90 }

sig Longitude extends Coordinate {}
{ intPart >= -180 and intPart <= 180}

sig Position{
	latitude: one Latitude,
	longitude: one Longitude,
}

sig Date{
	day: one Int,
	year: one Int
}
{ day > 0 and day <= 255 and year > 0}
--{ day > 0 and day <= 365 and year > 2000}
--Since 2000 is too big for our simu

abstract sig Identifier{
	idCode: String
}

sig FiscalCode extends Identifier{
}

sig VatCode extends Identifier{
}

sig User{
	fiscalCode: one FiscalCode,
	actualPosition: one Position,
	event: set RunEvent
}

sig Party{
	event: set RunEvent,
	vatCode: one VatCode
}

sig RunEvent{
	date: one Date,
	startHour: one Int,
	endHour: one Int,
	organizer: one Party,
	runner: some User,
	userPosition: User -> one Position,
	path: some Segment
}{ startHour >= 0 and endHour < 24 and startHour < endHour }

sig Segment{
	startPoint: Position,
	endPoint: Position
}{ startPoint != endPoint }
```

4.3 Facts

```
//FACTS

fact pathDefinitionConstraints{
	all re: RunEvent, s: Segment | s in re.path implies (
		lone s1,s2: Segment | s1 in re.path and s2 in re.path and s2.startPoint = s.endPoint and  s1.endPoint = s.startPoint
	)
}

fact uniquePosition{
	all disj p1, p2: Position | p1.longitude != p2.longitude or p1.latitude != p2.latitude
}

fact uniqueIdentifierCode{
	all disj i1, i2: Identifier | i1.idCode != i2.idCode
}

fact uniqueUser{
	all disj u1, u2: User | u1.fiscalCode != u2.fiscalCode
}

fact uniqueFiscalCodeForUser{
	all u: User | one fc: FiscalCode | u.fiscalCode = fc
}

fact uniqueParty{
	all disj p1, p2: Party | p1.vatCode != p2.vatCode
}

fact uniqueVatCodeForParty{
	all p: Party | one vc: VatCode | p.vatCode = vc
}

fact uniqueEventOrganizer{
	all re: RunEvent, p: Party | re.organizer = p implies re in p.event and
		(no p1: Party | re in p1.event)
}

fact eventsOverlap{
	all disj r1, r2: RunEvent |
		r1.date = r2.date and ((r1.path & r2.path) != none) implies (r2.startHour > r1.endHour or r1.startHour > r2.endHour)
}
```

4.4 Predicates

```
//PREDICATES

pred partyAddRunEvent[p: Party, re: RunEvent ]{
	!(re in p.event) implies p.event = p.event + re
}

pred userEnrolToRunEvent[u: User, re:RunEvent]{
	!(re in u.event) implies u.event = u.event + re
}

pred isUserCurrentlyRunningInEvent[u: User, re:RunEvent]{
	one s: Segment | s in re.path and (u.actualPosition = s.startPoint or u.actualPosition = s.endPoint)
}

pred show{}
```

SECTION 5: Effort Spent

Tools Used

The tools used for the realization of this document are the following:

- **Microsoft Word**: text editor with Live Collaboration feature
- **Microsoft OneDrive**: cloud space for synchronizing files between us
- **Draw.io**: uml class diagram, state, usecase and sequence diagrams
- **Alloy 5.0.0.1**: for realizing and analysing our model

NB: we haven't done Github commits because we preferred keeping the document on our shared OneDrive in order to edit it live between different

Hours Spent

The time we spent is equally divided between us:

- Paolo Saccani: 30 hours;
- Diego Riva: 30 hours;
- Matteo Rubiu: 30 hours.