1. **Introduction**
2. **Purpose**

This document is the Requirements Analysis Specification Document (RASD) for TrackMe’s systems Data4Help, AutomatedSOS and Track4Run. Its purpose is to provide a detailed description of these 3 systems. It is addressed to be read by the customers, users, systems analysts, requirements analysts, developers, programmers, testers and project managers.

Data4Help is designed as a software application used to allow third parties to monitor the location and the health status of individuals. Once registered, third parties can request data both of specific users and of groups of individuals. In the former case, an individual can accept or refuse the request; in the latter case, data will be provided only if they can be properly anonymized by the application. If the request is approved, Data4Help allows third parties to subscribe to new data and to receive them as soon as they are produced.

AutomatedSOS is a service designed to allow elderly people to receive help if their health parameters, monitored by Data4Help, go below certain thresholds.

Finally, Track4Run is designed to allow third parties to create runs, individuals to both enroll them and see the position of all runners on a map.

1. **Goals**

Data4Help:

* [G1] - Allows individuals to make available their position
* [G2] - Allows individuals to make available their health status
* [G3] - Allows third parties to request data of some specific individuals
* [G4] - Allows third parties to request access to anonymized data of groups of individuals
* [G5] - Allows individuals to choose whether to accept or not the request for sharing data required by third parties
* [G6] - Allows third parties to be able to see saved data as soon as a request is approved by the individual.
* [G7] - Allows third parties to have access to new data as soon as they are produced.
* [G8] - Allows third parties to be notified with the user’s response

AutomatedSOS:

* [G9] - Allows individuals to receive help if their health parameters go below certain thresholds

Track4Run:

* [G10] - Allows third parties to create athletic runs
* [G11] - Allows individuals to enroll to a run
* [G12] - Allows individuals to see the position of the runners on the map during a run

1. **Scope**

Data4Help is a useful application to monitor individuals’ data. These data are acquired by a smartwatch on which the application is downloaded. Using AutomatedSOS functionalities, one can be sure of receiving help if his/her health parameters go below certain thresholds. Finally, Track4Run provides interesting features for all those interested in a run. Focusing on the relevant phenomena for the system to be developed, it is possible to distinguish the world ones and the shared ones.

1. **World phenomena:**
2. **Shared phenomena:**

1. **Definitions, acronyms, abbreviations**
2. **Definitions**

* Users:  people registered on Data4Help, AutomatedSOS and Track4Run
* Individuals: people registered on Data4Help, AutomatedSOS and Track4Run that allow the application to acquire their data
* Group: set of individuals
* Third parties: people registered on Data4Help that can request data
* Participants: individuals enrolling a run
* Spectators: individuals that see a run
* Organizers: third parties that organize runs
* Data: information about health parameters and location of an individual or a group of individuals
* Help: ambulance
* Health parameters: pression, heartbeat and quality of sleep
* Devices: electronic device on which Data4Help can be installed
* System: the whole software system to be developed, comprehensive of all its parts and modules
* Node: place from which a run passes
* External partners: operators of a public service for urgent emergencies

1. **Acronyms**

* RASD:  Requirements Analysis Specification Document
* GPS: Global Positioning System
* HTTPS: Hypertext Transfer Protocol Secure
* TLS: Transport Layer Security

1. **Abbreviations**

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

1. **Revision History**
2. **Reference Documents**

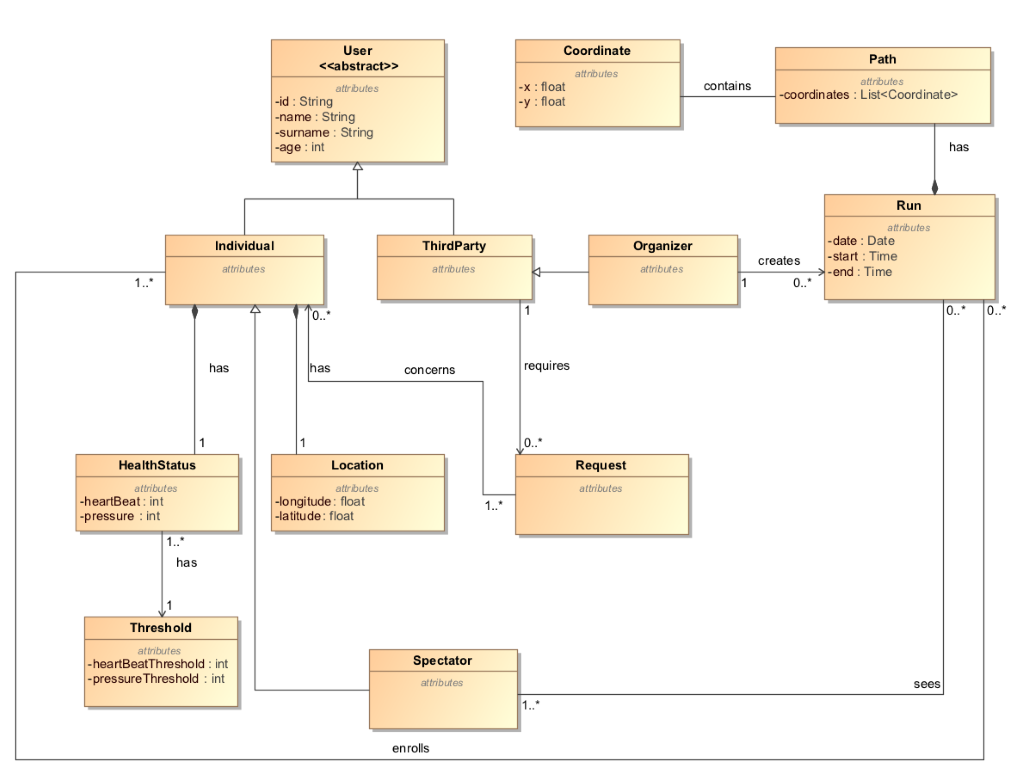
* Specification document “Mandatory Project Assignment AY 2018-2019”
* “Requirements Engineering Part II” from Beep

1. **Overview**

The document is divided in six parts. The first one includes introductory information to give a view of what this document is about. Indeed, it describes the purpose of the system to be developed by listing its goals and it describes also the scope of the application, by listing the world and the machine phenomena. The second part contains (to be completed). The third part contains (to be completed). The fourth part contains the alloy model

1. **Overall Description**
2. **Product perspective**

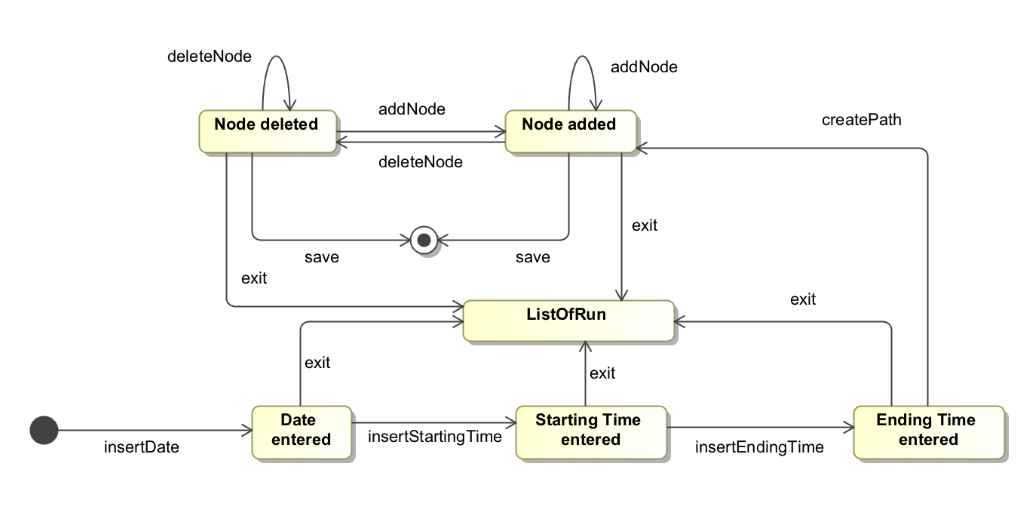
Data4Help is a system designed to provide the functionalities described in section 2.2 Product function. AutomatedSOS uses data provided by Data4Help and relies on external partners to send ambulances when necessary. Also, Track4Run uses data provided by Data4Help, furthermore it implements external APIs to offer maps services. To better represent the structure of the system, look at the class diagram shown in the figure below.



Aggiungere attributi in HealthStatus, magari cambiare “Coordinate” con “Node”

It is possible to observe that the system has two types of users: individuals and third parties. The former have a location and a health status, because Data4Help has permission to acquire these data. Furthermore, individuals can also enroll to a specific run and retrieve information about participants during it, thus becoming spectators. The latter can retrieve data of both individuals and groups of individuals through a specific request and can organize runs, thus becoming organizers.

The processes of creating a new run is shown in the state diagram below. (secondo me dovremo mettere lo state chart per fare una richiesta, perchè alla fine è il cuore di Data4Help, che è usata da tutte le altre applicazioni)



“ListOfRun” deve essere ListOfRuns”

It is possible to observe that creating a run implies the definition of the date, the starting time, the ending time and the path. A path is made of nodes and different nodes can be added or deleted during the creation of it. If the third parties click the “Exit” button during the creation of a run, they are shown the page containing the list of all runs and their run is not saved. Otherwise, when they finish inserting the nodes, the run is correctly created and saved.

1. **Product functions**

This section includes the most important requirements with respect to the already mentioned goals of the system.

**Request for data:**

Third parties can request for data of both individuals and groups of individuals. In the former case, it is necessary for the third parties to know the unique ID of the person they want to send the request to. Indeed, third parties are asked to enter the ID in order to send the request. The individual receives the request through a notification and can choose whether to accept it or not. Since Data4Help is able to retrieve data directly from an individual ‘s smartwatch, if the request is accepted, third parties are shown requested data on their devices. If the request is refused, third parties receive a message in which they are warned that the individual has chosen not to share his/her data. In the latter case, instead, third parties are asked to enter specific information in order to send the request. Indeed, third parties could be interested in data about people of a specific age range or living in a specific geographical area. Once the form is filled, the request is directly handled by the system: if the number of individuals whose data satisfy the request is higher than 1000, the request is accepted. In this case, indeed, the system is able to properly anonymize the requested data that are shown to third parties on their device. Furthermore, Data4Help sends them a message asking if they want to subscribe to the data they’ve asked for, in order to receive new values as soon as they’re produced. This request can be either accepted or refused. If the constraint is not satisfied, third parties receive a message informing them that the requested data cannot be made available for security reasons.

**Monitor health status:**

Individuals already signed up to Data4Help can use the functionality provided by AutomatedSOS service: by comparing health parameters with certain thresholds, the system is able to provide help to the individuals. In particular, AutomatedSOS sends, in less than 5 seconds, the external partner a message containing the location of the person who feels bad and its health parameters. The external partner is an operator of a public service for urgent emergencies and is in charge of sending an ambulance. AutomatedSOS is automatically activated by Data4Help if an individual is over 70 years old, otherwise it must be manually activated by the interested person.

**Create a run:**

Third parties can create a run by using the functionality provided by Track4Run. They have to insert the date of the run, its starting and ending time and also its path. While selecting the nodes, third party can either delete a previously inserted node or add new nodes, as shown in state diagram in section 2.1 Product Perspective. Track4Run effectively creates the run and adds it to the list of all runs if and only if all the fields are filled with valid data. The created run can be seen on the application both by third parties and by individuals.

**Enroll to a run:**

Individuals can see on their devices the list of all upcoming runs and they can also decide to enroll to one of them (il secondo “to” ci va?). In this case, since Track4Run exploits Data4Help data, there’s no need for the individuals to enter their name, surname and age: they only have to enter their password to actually enroll the selected run. Once the enrollment is completed, the individuals are provided with their bib number.

See runners’ position:

Individuals can see on their devices the list of all runs taking place and they can decide to become spectators of a run. This means that Track4Run allows them to see on a map the position of all runners. The position is known because Track4Run exploits Data4Help measurements. (quest’ultima frase non so se metterla, perchè penso che sarà detto anche in altre parti del documento)

1. **User characteristics**
2. **Assumptions, dependencies and constraints**

* [D1] - The individuals’ devices are able to measure all the health parameters needed by Data4Help
* [D2] - The individuals’ devices are able to provide an accurate detection of the health parameters
* [D3] - The GPS must be active on the individuals’ devices
* [D4] - The individuals’ devices are able to provide an accurate enough current location
* [D5] - The external partner takes charge of the help request
* [D6] - An ambulance arrives to the user’s location
* [D7] - A run present in the list of all runs is actually held
* [D8] - The third parties know the ID of the individual they’ve asked data for
* [D9] - An internet connection must be active
* [D10] - The individuals are always with the device when they are using Data4Help

1. **Specific Requirements**
2. **External Interface Requirements**
3. **User Interfaces**

The following mockups represent a basic idea of what the application will look like in the first release.

DA METTERE MOCKUPS

1. **Hardware Interfaces**

1. **Software Interfaces**

The system will need:

* A server database where it will store users’ data such as location and health parameters.

If Track4Run is used to monitors participants position during a run, the application requires the use of a map. External services such as Google Maps can be used. (da finire e mettere in elenco puntato)

1. **Communication Interfaces**

The clients communicate with the server via HTTPS requests (port 443), and TLS protocols guarantees communication security. There will be the need to build an API that lets the application to store and/or retrieve data from the database, which is located server-side.

1. **Functional Requirements**

The following list contains the functional requirements of the system to be developed:

* [R1] - Individuals’ health status is provided to Data4Help by their smartwatches
* [R2] - Individuals’ location is provided to Data4Help by their smartwatches
* [R3] - Third parties can identify a specific individual entering on Data4Help his/her unique ID
* [R4] - Third parties can send a specific individual the request for his/her data through Data4Help
* [R5] - Third parties can send a request for anonymized data of groups of individuals through Data4Help
* [R6] - Third parties’ request is sent to the interested user by Data4Help
* [R7] - Individuals can view on Data4Help the request they’ve received
* [R8] - Individuals can accept or refuse the request using Data4Help
* [R9] - Third parties can view on their devices data provided by Data4Help
* [R10] - Data4Help has access to the requested data
* [R11] - Data4Help must take into account individuals’ response
* [R12] - Third parties can view requested data on their devices
* [R13] - The application sends the users’ data to the external partner
* [R14] - The users are shown the map
* [R15] - Organizers can create a run by entering the date, the starting and ending time and the path
* [R16] - Individuals are shown a list of athletic runs
* [R17] - Individuals can select a specific run from the list
* [R18] - Track4Run shows the location only of the individuals who actually enrolled the run
* [R19] - Users can create a Data4Help account
* [R20] - Users can log in to the application by providing the combination of a username and a password that match an account

These requirements ensure the satisfaction of the goals in the context of the domain assumptions.

* [G1] - Allows individuals to make available their position
* [D3] - The GPS must be active on the individuals’ devices
* [D4] - The individuals’ devices are able to provide an accurate enough current location
* [D9] - An internet connection must be active
* [D10] - The individuals are always with the device when they are using Data4Help
* [R2] - Individuals’ location is provided to Data4Help by their smartwatches
* [G2] - Allows individuals to make available their health status
* [D1] - The individuals’ devices are able to measure all the health parameters needed by Data4Help
* [D2] - The individuals’ devices are able to provide an accurate detection of the health parameters
* [D9] - An internet connection must be active
* [D10] - The individuals are always with the device when they are using Data4Help
* [R1] - Individuals’ health status is provided to Data4Help by their smartwatches
* [G3] - Allows third parties to request data of some specific individuals
* [D8] - The third parties know the ID of the individual they’ve asked data for
* [D9] - An internet connection must be active
* [R3] - Third parties can identify a specific individual entering on Data4Help his/her unique ID
* [R4] - Third parties can send a specific individual the request for his/her data through Data4Help
* [G4] - Allows third parties to request access to anonymized data of groups of individuals
* [D9] - An internet connection must be active
* [R5] - Third parties can send a request for anonymized data of groups of individuals through Data4Help
* [G5] - Allows individuals to choose whether to accept or not the request for sharing data required by third parties
* [D9] - An internet connection must be active
* [R6] - Third parties’ request is sent to the interested user by Data4Help
* [R7] - Individuals can view on Data4Help the request they’ve received
* [R8] - Individuals can accept or refuse the request using Data4Help
* [G6] - Allows third parties to be able to see saved data as soon as a request is approved by the individual.
* [D9] - An internet connection must be active
* [R9] - Third parties can view on their devices data provided by Data4Help
* [R10] - Data4Help has access to the requested data
* [G7] - Allows third parties to have access to new data as soon as they are produced.
* [D9] - An internet connection must be active
* [R1] - Individuals’ health status is provided to Data4Help by their smartwatches
* [R2] - Individuals’ location is provided to Data4Help by their smartwatches
* [R9] - Third parties can view on their devices data provided by Data4Help
* [R10] - Data4Help has access to the requested data
* [G8] - Allows third parties to be notified with the user’s response
* [D9] - An internet connection must be active
* [R11] - Data4Help must take into account individuals’ response
* [R12] - Third parties can view requested data on their devices
* [G9] - Allows individuals to receive help if their health parameters go below certain thresholds
* [D2] - The individuals’ devices are able to provide an accurate detection of the health parameters
* [D3] - The GPS must be active on the individuals’ devices
* [D4] - The individuals’ devices are able to provide an accurate enough current location
* [D5] - The external partner takes charge of the help request
* [D6] - An ambulance arrives to the user’s location
* [R1] - Individuals’ health status is provided to Data4Help by their smartwatches
* [R2] - Individuals’ location is provided to Data4Help by their smartwatches
* [R13] - The application sends the users’ data to the external partner
* [G10] - Allows third parties to create athletic runs
* [D9] - An internet connection must be active
* [R14] - The users are shown the map
* [R15] - Organizers can create a run by entering the date, the starting and ending time and the path
* [G11] - Allows individuals to enroll to a run
* [D9] - An internet connection must be active
* [R16] - Individuals are shown a list of athletic runs
* [R17] - Individuals can select a specific run from the list
* [G12] - Allows individuals to see the position of the runners on the map during a run
* [D3] - The GPS must be active on the individuals’ devices
* [D4] - The individuals’ devices are able to provide an accurate enough current location
* [D7] - A run present in the list of all runs is actually held
* [D9] - An internet connection must be active
* [R2] - Individuals’ location is provided to Data4Help by their smartwatches
* [R14] - The users are shown the map
* [R16] - Individuals are shown a list of athletic runs
* [R18] - Track4Run shows the location only of the individuals who actually enrolled the run

1. **Scenarios**

**Scenario 1:**

Andrea secretly skips classes because of a test for which he isn’t well prepared. He decides to take a walk in Milan. While doing breakfast and reading the oroscope he receives a request asking for his current location. Andrea doesn’t want his data to be available to third parties because he is afraid his mother will find out he’s not at school. Therefore, he refuses the request and orders another muffin.

**Scenario 2:**

While running along Naviglio river, Lucio receives a request from a company asking him to share his current data. He accepts the request and the company is able to find out what Lucio is doing. As soon as Lucio accepts, the company receives previously saved data from TrackMe and it is asked whether to subscribe to new Lucio’s data or not.

**Scenario 3:**

Shebi is a company that wants to make a survey to compare engineering and philosophy students’ quality of sleep. It uses Data4Help and by filling the form in the request section, acquires all the needed informations because TrackMe is able to properly anonymize the requested data. By comparing the average of the values of the two, Shebi finds out that engineer students sleep less.

**Scenario 4:**

The pharmacy of Sedriano, a small city in Milan province, wants to know if it has to increase the purchase of Norvasc, a medicine for the hypertension treatment. For doing so, it asks Data4Help the pressure data about people living in Sedriano with age between 60 and 80 years. Since TrackMe hasn’t enough registered users to properly anonymize data, it is not able to provide the pharmacy with the requested information.

**Scenario 5:**

Martina is 78 years old and she is celebrating Christmas’ Eve with her family. One of the gifts she receives is a smartwatch. Her niece, Giorgia, suggests her to download Data4Help on the new device. Martina signs up as an individual providing her e-mail, password, fiscal code and her personal informations. Because of her age, AutomatedSOS service is automatically activated, in order to monitor her health status and to guarantee help in case of need.

After a week Martina is walking to the supermarket with her smartwatch tied to her wrist. Suddenly she feels a little weak. As soon as her parameters go below certain thresholds, the smartwatch starts ringing and AutomatedSOS sends Martina’s data to an external partner. She faints and when she wakes up, she is on the ambulance trying to figure out what happened.

**Scenario 6:**

Politecnico Di Milano wants to organize a run called PolimiRun. For doing so, it decides to use a new service in Data4Help application called Track4Run. The organizers enter the date, the starting and ending time and also the path of the run. The event is visible to all the users of the application.

**Scenario 7:**

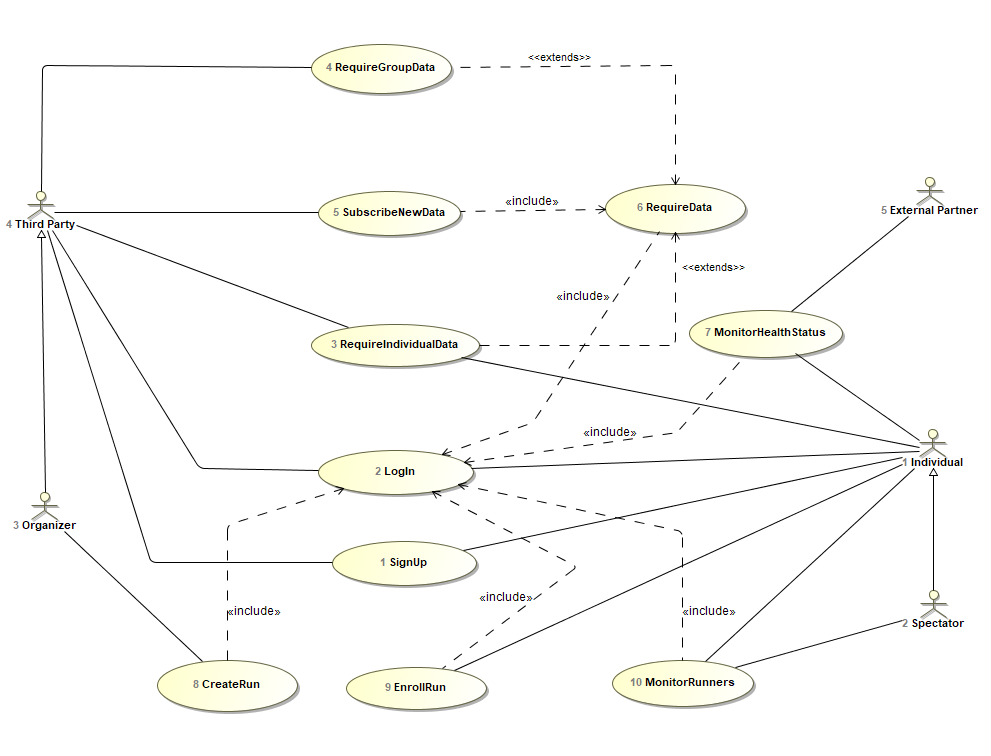
Stefano is a Computer Science and Engineering student who wants to attend PolimiRun. He has Track4Run installed on his smartwatch and decides to use for the first time this application to enroll the run. He doesn’t need to insert his data because Data4Help provides Track4Run all the personal information. Therefore, he receives his bib number.

**Scenario 8:**

Mariangela has two sons attending PolimiRun. During the run she decides to use the Track4Run functionality to see on a map the position of her sons. Since the application exploits the features offered by Data4Help, including the users’ location, Mariangela finds out that the youngest son has already finished instead the eldest one is still missing 3 km to the finish line.

1. **Use cases**

The use case diagram is represented in the image below.



A detailed description of the use cases is now provided.

Use case 1:

|  |  |
| --- | --- |
| Name | SignUp |
| Actor | Individual  Third party |
| Entry conditions | The user has installed Data4Help on his/her device |
| Events flow | 1. The user clicks on “Sign up” button 2. The user chooses whether to sign up as a third party or an individual 3. The user fills all the mandatory fields and provides the necessary information 4. The user clicks on “Confirm” button |
| Exit conditions | The user has successfully registered and now is able to use the application |
| Exceptions | 1. The user is already signed up 2. The user didn’t fill all the mandatory fields with valid data 3. The username is already taken 4. The e-mail is already registered |
| Constraints and nonfunctional requirements | / |

Use case 2:

|  |  |
| --- | --- |
| Name | LogIn |
| Actor | Individual  Third party |
| Entry conditions | The user has already signed up |
| Events flow | 1. The user opens the application on his/her device 2. The user enters his/her username and password 3. The user clicks on the “Log in” button |
| Exit conditions | The user is successfully logged in |
| Exceptions | 1. The user enters invalid username 2. The user enters invalid password |
| Constraints and nonfunctional requirements | / |

Use case 3:

|  |  |
| --- | --- |
| Name | RequireIndividualData |
| Actors | Individual  Third Party |
| Entry conditions | Users have already logged in |
| Events flow | 1. Third party clicks the “send individual request” button 2. Third party inserts the individual’s ID 3. Third party selects the individual’s parameters he is interested in 4. Third party sends the request to the individual 5. The individual chooses whether to accept or not the request 6. Third party is notified with the response and he is provided with the saved data if and only if it is positive. |
| Exit conditions | Third party can view the response on his device (non so se mettere il punto 5 del flow event o questa exit condition) |
| Exceptions | 1. Individual’s ID inserted by the third party doesn’t exist 2. Third party doesn’t select any parameter |

Use case 4:

|  |  |
| --- | --- |
| Name | RequireGroupData |
| Actor | Third Party |
| Entry conditions | Third party has already logged in |
| Events flow | 1. Third party clicks the “send group request” button 2. Third party fills the form with the required data 3. Third party sends the request 4. Third party is notified with the response and is provided with the required information if and only if TrackMe is able to properly anonymize data |
| Exit conditions | Third party can view the response on his device |
| Exceptions | Third party doesn’t fill all the mandatory fields with valid data |
| Constraints and nonfunctional requirements | The number of users satisfying third party’s request must be higher than 1000 |

Use case 5:

|  |  |
| --- | --- |
| Name | SubscribeNewData |
| Actor | Third party |
| Entry conditions | Third party has already received a positive response to his request |
| Events flow | 1. Third party is asked if he wants to subscribe to new data and to receive them as soon as they are produced 2. Third party chooses whether to subscribe or not to new data |
| Exit conditions | TrackMe is notified with the third party’s response |
| Exceptions | / |

Use case 7:

|  |  |
| --- | --- |
| Name | MonitorHealthStatus |
| Actors | External partner  Individual(E’ DA TOGLIERE?? Perchè Data4Help acquisisce I dati dell’utente automaticamente. In tal caso va modificato lo useCaseDiagram) |
| Entry conditions | Users has already logged in |
| Events flow | 1. Data4Help sends AutomatedSOS an individual’s health parameters 2. If these parameters go below certain thresholds, AutomatedSOS notifies the external partner, otherwise Data4Help repeats the measurements |
| Exit conditions | External partner is notified with all the information needed |
| Exceptions | / |
| Constraints and nonfunctional requirements | The reaction time from the moment the parameters go below the thresholds must be less than 5 seconds |

Use case 8:

|  |  |
| --- | --- |
| Name | CreateRun |
| Actor | Organizer |
| Entry conditions | Organizer has already logged in |
| Events flow | 1. Organizer enters the list of all runs 2. Organizer click the “+” button 3. Organizer fills the form with all required information (date, starting time and ending time) 4. Organizer defines the path on the map 5. Organizer click the button “create run” |
| Exit conditions | The run has been correctly created |
| Exceptions | Organizer doesn’t fill all the mandatory fields with valid data |
| Constraints and nonfunctional requirements | / |

Use case 9:

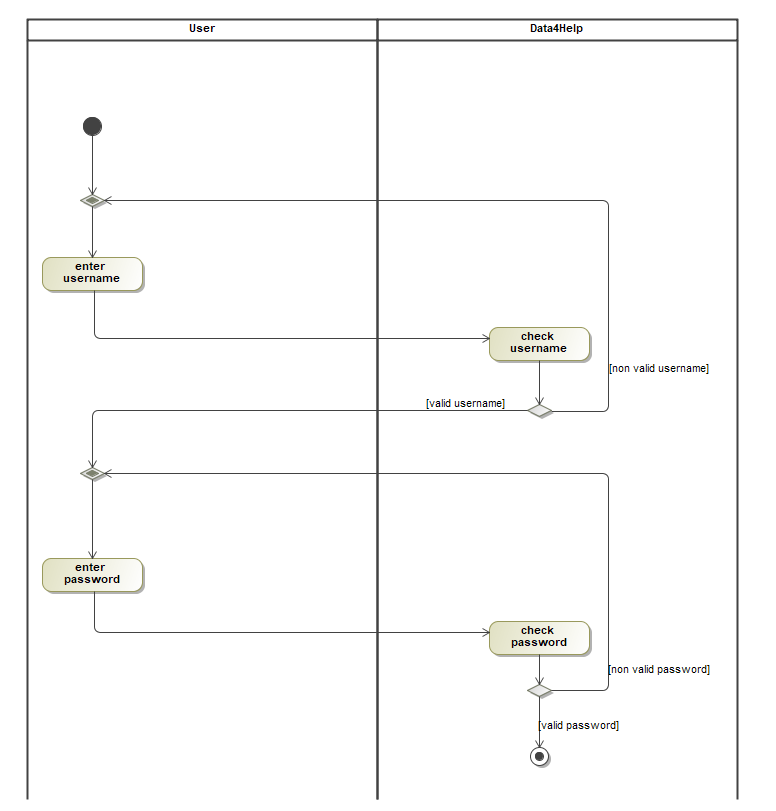
|  |  |
| --- | --- |
| Name | EnrollRun |
| Actor | Individual |
| Entry conditions | Individual has already logged in |
| Events flow | 1. Individual sees the list of all runs 2. Individual select the run he wants to enroll 3. Individual enters his password |
| Exit conditions | Individual has correctly enrolled to the run |
| Exceptions | / |
| Constraints and nonfunctional requirements | / |

Use case 10:

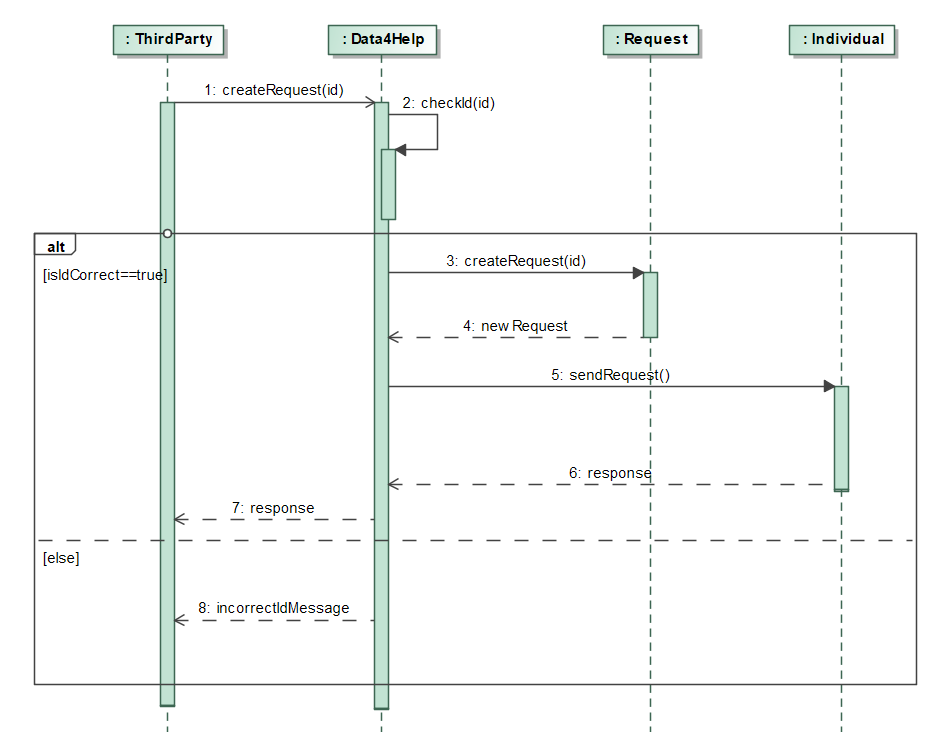
|  |  |
| --- | --- |
| Name | MonitorRunners |
| Actors | Spectator  Individual |
| Entry conditions | The users have already logged in  Individual has enrolled to the run |
| Events flow | 1. Spectator sees the list of all runs 2. Spectator selects the run he wants to view 3. Track4Run returns the spectator a map showing all the location of all the participants |
| Exit conditions | Spectator can see the position of all runners on his device |
| Exceptions | / |
| Constraints and nonfunctional requirements | / |

1. **Diagrams**

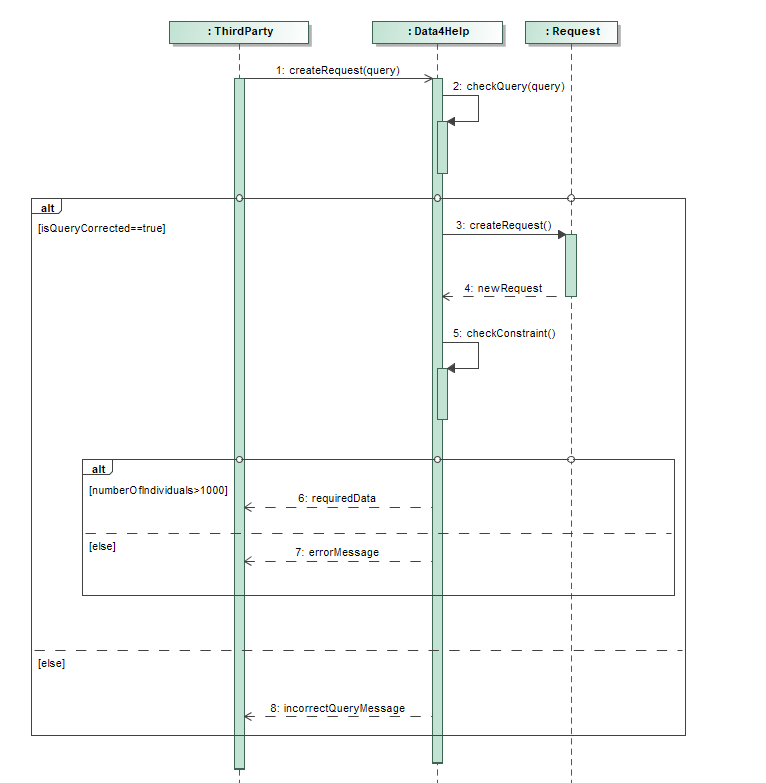
The activity diagram of use case 2 is represented below.



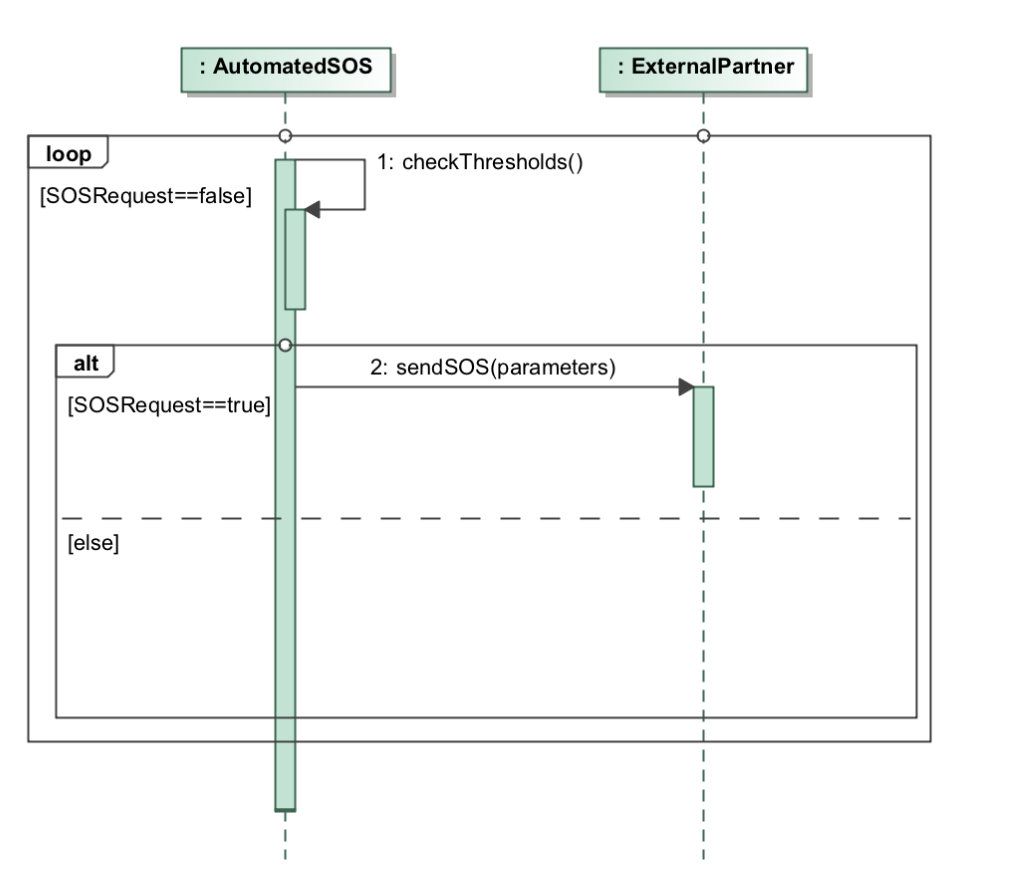
The sequence diagram of use case 3 is represented below.



The sequence diagram of use case 4 is represented below.



The sequence diagram of use case 7 is represented below.



1. **Performance Requirements**
2. **Design Constraints**
3. **Standards compliance**
4. **Hardware limitations**
5. **Any other constraint**
6. **Software System Attributes**
7. **Reliability**
8. **Availability**
9. **Security**
10. **Maintainability**
11. **Portability**
12. **Formal Analysis Using Alloy**
13. **Effort Spent**
14. **References**