DD

*Version n°1*

*Software engineering 2*

*2018*

*Mohamed Gawish*

*Emma Bortone*

ABSTRACT:

The design document (DD) provides a functional description of the system

[Attirez l’attention du lecteur avec une citation du document ou utilisez cet espace pour mettre en valeur un point clé. Pour placer cette zone de texte n’importe où sur la page, faites-la simplement glisser.]

Politecnico di Milano

TrackMe Project

TrackMe Project

|  |  |
| --- | --- |
| Image associÃ©e |  |

[Introduction 1](#_Toc532046296)

[Purpose 1](#_Toc532046297)

[Hypotheses on the assignment 2](#_Toc532046298)

[Definitions, Acronyms and abbreviations 3](#_Toc532046299)

[Architectural design 4](#_Toc532046300)

[Overview 4](#_Toc532046301)

[Component view 6](#_Toc532046302)

[Deployment view 11](#_Toc532046303)

[Runtime view 12](#_Toc532046304)

[Component interfaces 21](#_Toc532046305)

[Selected architectural styles and patterns 21](#_Toc532046306)

[User interface design 21](#_Toc532046307)

[Requirements traceability 21](#_Toc532046308)

[Implementation and test plan 22](#_Toc532046309)

[Effort spent 22](#_Toc532046310)

[Mohamed 22](#_Toc532046311)

[Emma 22](#_Toc532046312)

[References 23](#_Toc532046313)

# Introduction

## Purpose

The purpose of this document is to define the requirements analysis and specification document (RASD) of the services Data4Help and AutomatedSOS offered by the company TrackMe. This document contains the requirements of the system to be developed and its application domain. It can be used as a baseline for software evaluation and for charge control.

### Data4Help

Data4Help is a software-based service allowing third parties to monitor the location and health status of individuals. Data4Help supports the registration of individuals and of third parties.

* Individuals who register to Data4Help allow the company TrackMe to acquire their data.
* Third parties who register to Data4Help can access the data acquired by TrackMe by the mean of requests. They can request to :
  + Access to the data of a specific individual by providing a unique identifier. The request is then transferred to the individual who decide to accept or refuse it.
  + Access to anonymized data of groups of individuals. The request is accepted if the number of individuals satisfying it is higher than 1000.

When a request is accepted, third parties must be able to access the previously saved data and to subscribe to the request. By subscribing to a request, third parties will receive the new data corresponding to the request as soon as the data is available and without the need to renew the request process each time. Third parties can make an unlimited number of requests and subscriptions

The goals of the Data4Help service are :

* [G 1] : Third parties must be able to request to access to the data of specific individuals or to anonymized groups of individuals.
* [G 2] : At any time, third parties should never have access to data of specific individuals without their agreement.
* [G 3] : Third parties must have the possibility to subscribe to new data if their request is accepted.
* [G 4] : Individuals must be able to consult their data and accept/refuse requests

### AutomatedSOS

AutomatedSOS is a service build on top of Data4Help, therefore AutomatedSOS must verify all the requirements of the service Data4Help. In addition, AutomatedSOS offers the possibility to monitor the health of the subscribed individuals and to automatically send an ambulance to the location of the individuals if their health parameter are below certain thresholds.

All the goals of Data4Help are also goals of Automated SOS, but AutomatedSOS have the additional goal :

* [G 5] : An SOS is sent to an ambulance service, specifying the position of the individual, with a reaction time below 5 seconds from the time the individual’s health parameters are below threshold.

## Hypotheses on the assignment

To solve the ambiguity and incompleteness of the project, we made some assumptions on the services Data4Help and AutomatedSOS. Those assumptions are listed in the two following paragraphs.

### Data4Help

* The data is collected by the mean of a smartwatch synchronized to a smartphone application. We made this choice because most of the smartwatches currently on the market are aimed to be linked to a smartphone through an application and we want our software to run on as many platforms as possible (design for portability)
* The subscription to a request on a group of individuals is automatically cancelled if the number of individuals whose data satisfy the request goes bellow 1000 at some point.
* The subscription to a request on a specific individual is automatically cancelled if the individual cancels his agreement.
* The individual can see the data collected by the service Data4Help. Indeed, the individual must earn something in exchange for the data he agrees to share. In this case, the individual gains the ability to control his health data.
* Data4Help must respect the General Data Protection Regulation (GDPR)
* Third parties can be compagnies, organizations or persons who need to acquire data (for example students or independent data scientists).
* The Data4Help service take the form of :
  + **A website and a Web API for data request.**

We made this choice because using a website is the more convenient solution for a company or an organization who need to punctually acquire data and a web API is the most convenient solution for third parties who need to acquire data regularly (each hour for example)

* + **A smartphone application for data acquisition.**

We made the choice to consider that the individuals have some piece of Data4Help installed on their personal device which allow Data4Help to acquire their health data. The motivation for the individual to install Data4Help on his smartphone will be to observe his own health data. Other solution would be to gathers data from some other service/system but this would mean to buy the data and the delay might be more important.

### AutomatedSOS

* The AutomatedSOS service is not an independent application. The individuals who want to subscribe to AutomatedSOS will first need to download the application Data4Help on their smartphone to register to the service. We made this choice because the AutomatedSOS service is only an extension of the Data4Help service so the requirements of the Data4Help service are also requirements of the AutomatedSOS service .
* AutomatedSOS notifies the ambulance service that an ambulance needs to be sent to a certain location by sending a SMS.
* Individuals who want to subscribe to AutomatedSOS are required to have a smartwatch and to link it to their Data4Help account. Indeed, a smartphone cannot acquire the pulse rate of the individual and this data is significant for detecting emergencies.
* The thresholds are automatically defined by AutomatedSOS depending on the individual’s characteristics. We choose to prevent individuals to set themselves their thresholds to avoid unnecessary SOS or an urgent situation not being detected.
* Individuals also have the possibility to download an application on their smartwatch, in addition to the app on their smartphone, although this is not mandatory for the data acquisition. This will allow the individual to have an overview of their health data by simply looking at their watch. We made this choice because the individuals who subscribe to AutomatedSOS are elderly people who might not be comfortable with smartphones.

## Definitions, Acronyms and abbreviations

Third party: company or organization or person who need to access data of individuals

Individual: person willing to share his data with a third party and who want to monitor his health.

Smartwatch: device aimed to be worn on the wrist. A smartwatch can collect data, can be linked to a smartphone application and can support wireless technologies like Bluetooth, Wi-Fi, and GPS.

Subscription: an arrangement for automatically receiving data corresponding to a specific accepted request as soon as the data is available and without the need to renew the request process each time.

Request: the act, for a third party, of asking for data. A request can be individual or anonymized.

Individual request: A request that concerns a unique person known by his fiscal code.

Anonymized request: A request that concerns a group of individuals who fit some criteria. The result of this type of request is anonymized.

Anonymize: to remove any information that shows which particular person a data relates to.

SOS: SMS sent by Data4Help to an ambulance service asking to send an ambulance to a specific location. An SOS is sent when the health parameters of a monitored individual go below threshold.

# Architectural design

## Overview

The physical components that are involved in the Data4Help service are:

* Individual’s cell phone
* Individual’s Smartwatches
* Third party’s Computer
* TrackMe system (which contains databases and server)

For the service Automated SOS, the following service must be added :

* Ambulance service

Communications between components:

* Cell phone ↔ Server (The cell phone sends requests to the server and the server answers)
* Computer ↔ Server (The computer sends requests to the server and the server answers)
* Server ↔ Database (The server makes queries and the database answers)
* Smartwatch ↔ Cell phone (The cell phone sends queries to the smartwatch and the smartwatch sends data to the cellphone via Bluetooth)
* Cell phone → Ambulance service (The cell phone sends SMS to an ambulance service)

The server must send data to the different users (individuals or third parties) only when it’s necessary. Because the users use the app through devices that can be offline, the server must only send data when asked by the users. Otherwise the user could never receive the data if his device is offline. For those reasons, an event-based system would not be adapted

We need to design a system which involves many stakeholders such as individuals, third parties, ambulance services, track systems. Moreover, in all the interactions the system is providing a service to the users so we decided to use a client-server architectural approach.

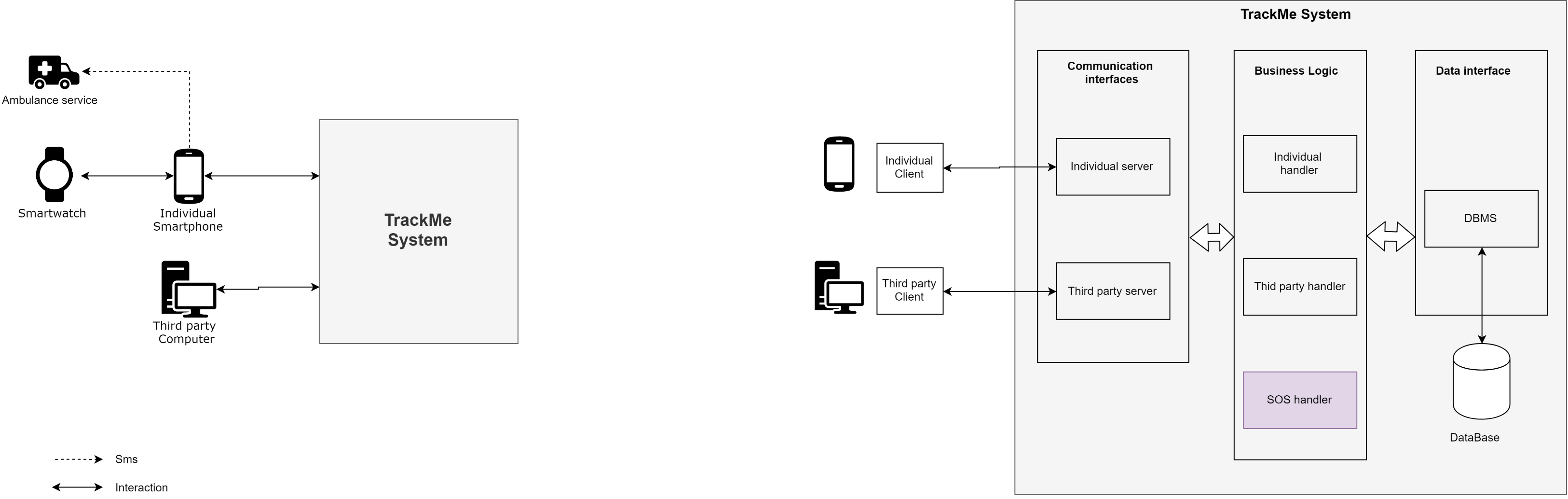


Figure 1 Overview 1

Going deeper in the analysis of the components of the TrackMe system, we are able to identify three different layers :

* **Communication Interfaces :**

This lawyer contains the interfaces components that allow the system to communicate with external agents (individual’s smartphones and third parties’ computers). As TrackMe system interacts with different external agents so it needs to have different communication Interfaces :

* A software module is needed in order to provide functionalities of the system to the individuals on their smartphone. This software module should allow individuals to send requests to the server using an API.
* A software module is needed to provide the functionalities of the system to the third parties. This module includes a website back -end and an API. The website back-end allows third parties to communicate with the server using a website ; the API allows the parties to download the data requested.
* **Business Logic :**

This lawyer focuses on the application logic of the TrackMe system. The business logic manages the individuals data, the third parties requests and their subscriptions ; for each of these functions, several software modules are necessary. Those modules will use the communication interfaces between the end user interface and the database. The business logic module receives from the communication interface orders to do specific actions; then the business logic ask to the data base interface the required data to execute these actions, and finally, returns the result to the communication interface.

* **Data Interface**

This lawyer contains all the modules that allow to store the data produced or retrieved from external resources. These modules allow interaction between the Business Logic modules and the System Databases.

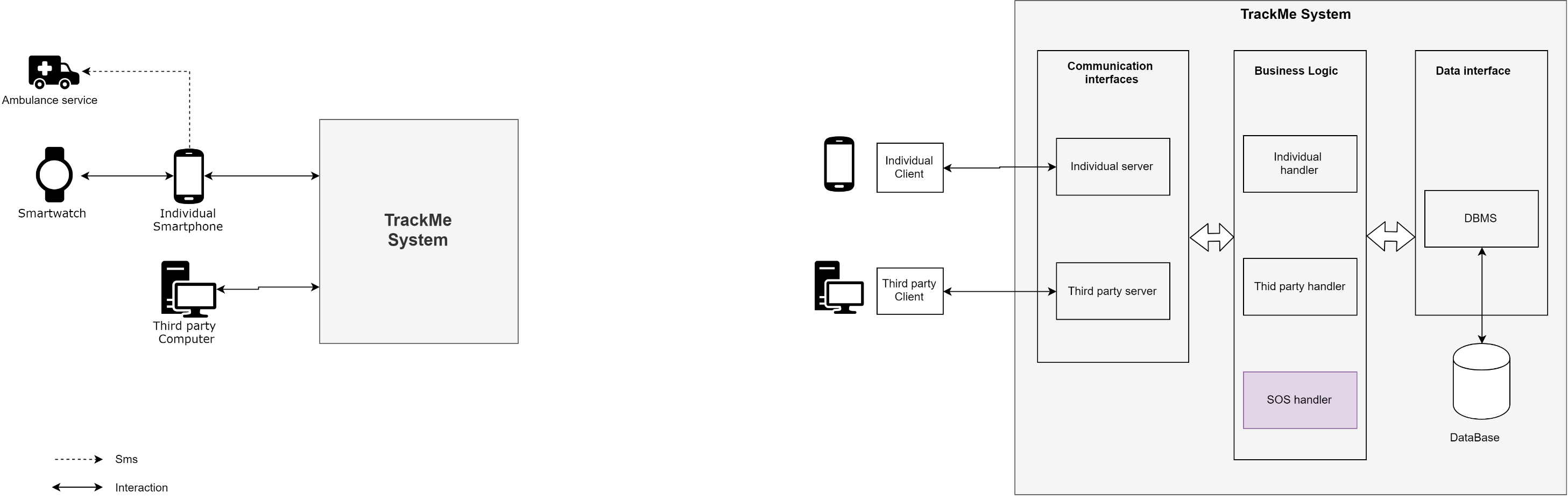


Figure 2 : Overview 2

## Component view

The following diagram, give a high-level representation of the components of the TrackMe system.

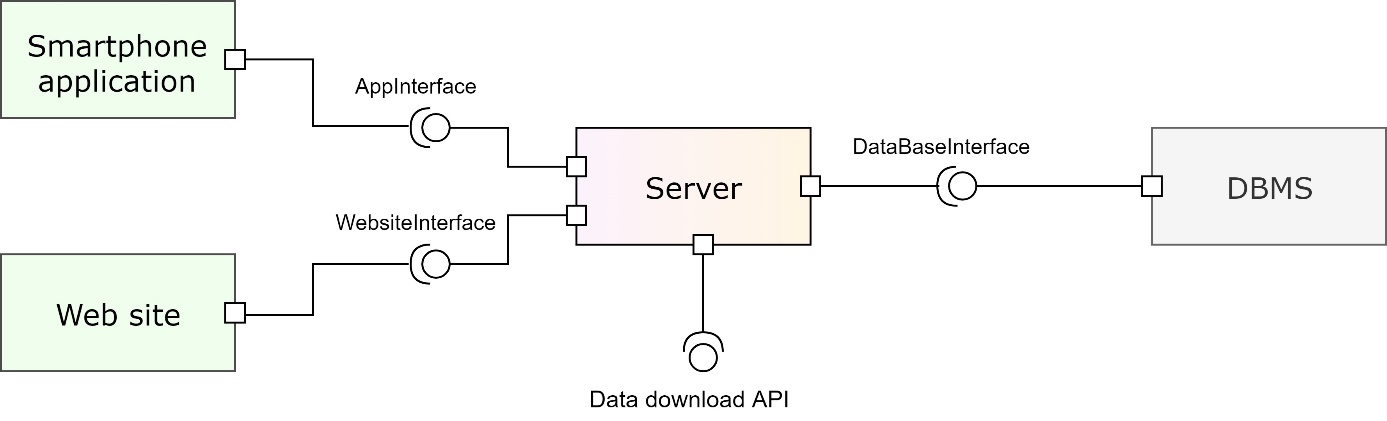


Figure 3: High level representation TrackMe system

As we can see, the system is composed of four main parts : A smartphone application, a Web site, a server and a DBMS. The smartphone application is intended for individuals and the Web site is intended for third parties. The server contains the business logic and make the connection between the others components and the DBMS.

The next component diagram contains a more low-level description of the server component :

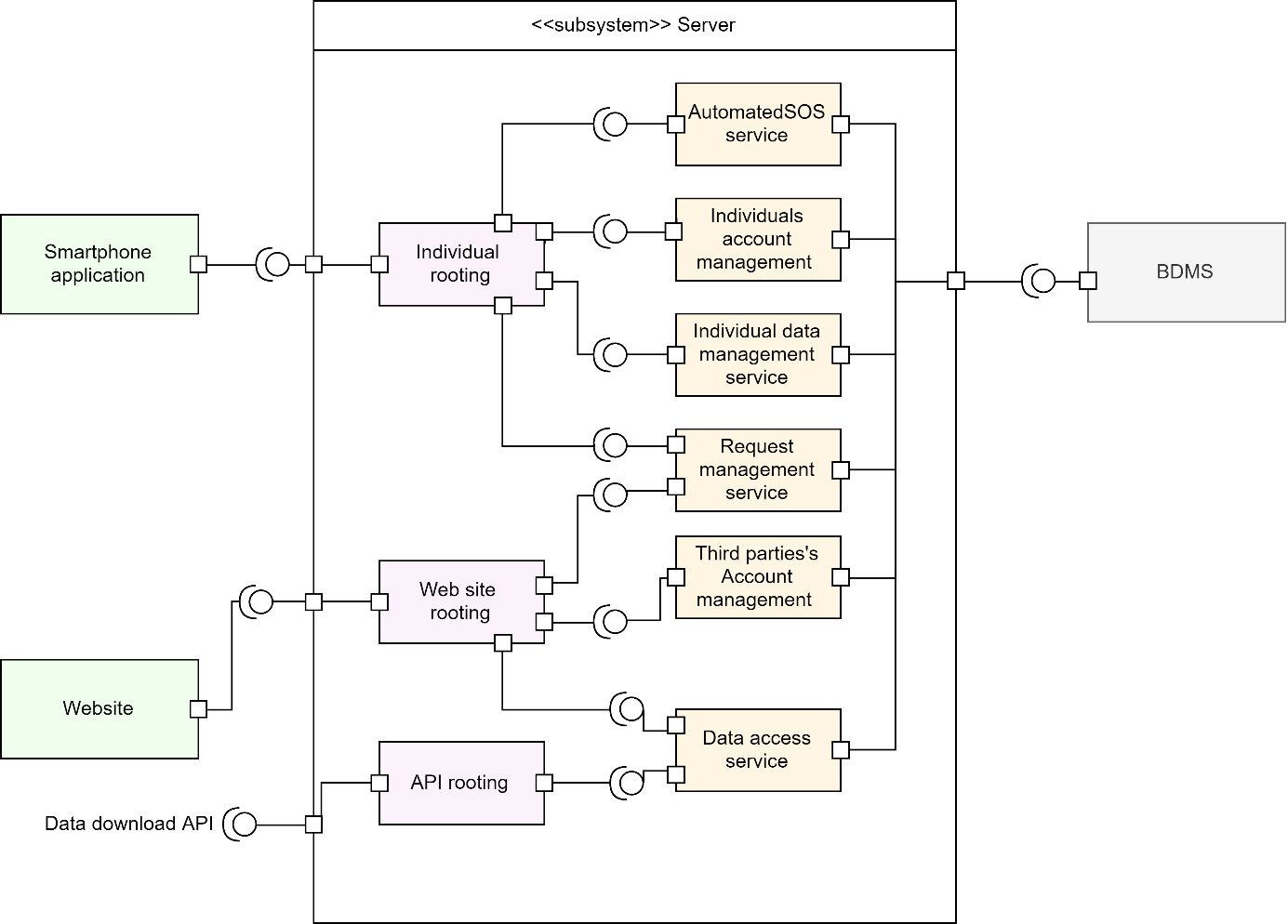


Figure 4: Low level representation trackMe system

As we can see, the server is composed of three main parts : Individual rooting, Web site rooting and API rooting. Those components determine whether a request that is received is valid or not and, if it is, to dispatch it to the relevant service component.

* **Individual rooting.**

This component manages all the functions the smartphone application has to provide to the individuals.

Individual rooting redirects the requests from the application to the corresponding component:

* *AutomatedSOS service* : manages all the functionalities of the AutomatedSOS service
* *Individuals account management* : manages the functionalities that allow individuals to register, login etc..
* *Individuals data management service :* manage the logic of the health and position data of individuals
* *Request management service :* manage the logic of the requests (for individuals : accepting or refusing request)
* **Web site rooting and API rooting.**

As explained in the RASD, the third parties can access all the functions of the Data4Help system through the Website and have also the possibility to download the data from accepted requests through an API. The API is aimed at accelerating the data acquisition process for third parties who need to acquire data regularly.

Web site rooting redirect the requests from the website to the corresponding component:

* *Third parties account management:* manages the functionalities that allow third parties to register, login etc..
* *Data access service:* manages the logic of formatting data from the Database to the third party.
* *Request management service:* manage the logic of the requests (for third parties : submitting individual or anonymized requests, subscribing to accepted requests)

The two following diagrams describe more precisely the business logic components, by showing their interactions with the data model. It is assumed that each component is able to invoke operations on the database

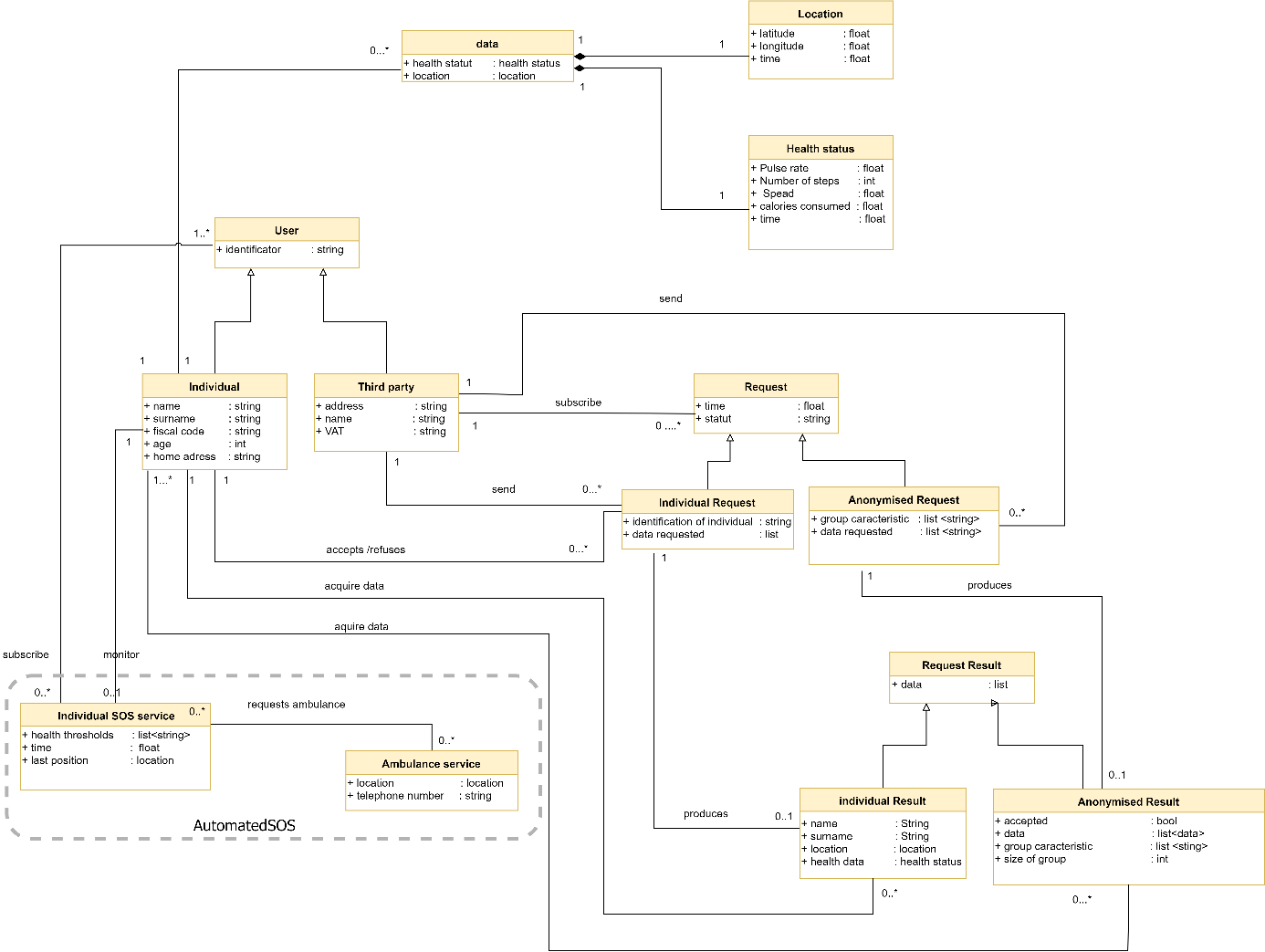


Figure 5: Class diagram of AutomatedSOS system

Some business components contain functions that requires the need to save data in the database. Those functions return a Boolean value : True if the data has been successfully saved, False otherwise. When a function is of this type, the return value is “bool\*”. The symbol \* make the distinction between functions of this type and other Boolean functions.

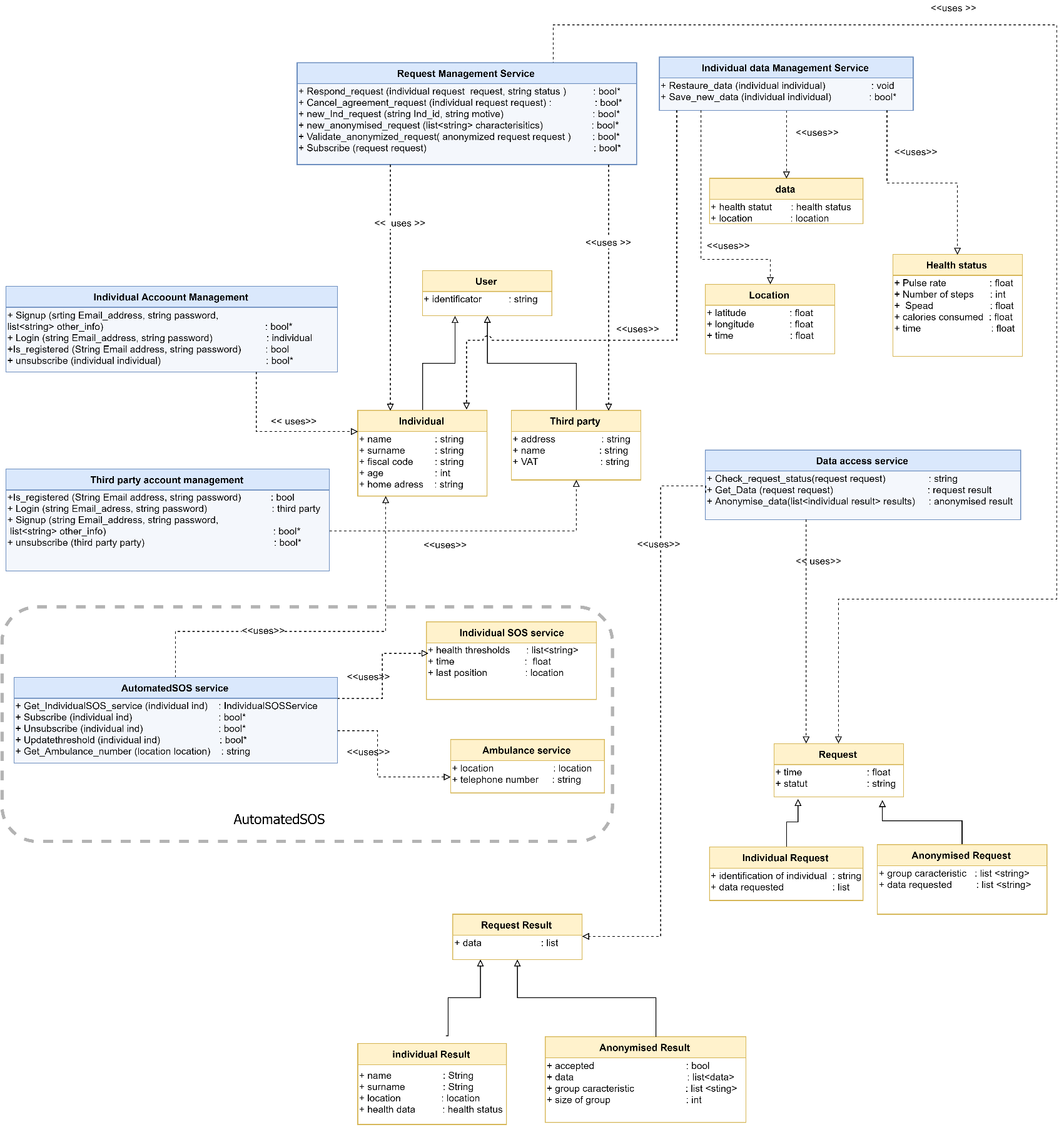


Figure 6: Description of logic components

The following diagram (figure XX) describes in further details the components of the individual’s smartphone. A smartphone contains a Database which stores the last health data of individuals. When an internet connection is available, the collected data is send to the TrackMe server.

Moreover, the thresholds and the phone number of the SOS services are also stored in the smartphone database and updated when an internet connection is available. This allows the app AutomatedSOS to send an SOS (using SMS) even without having an internet connection.

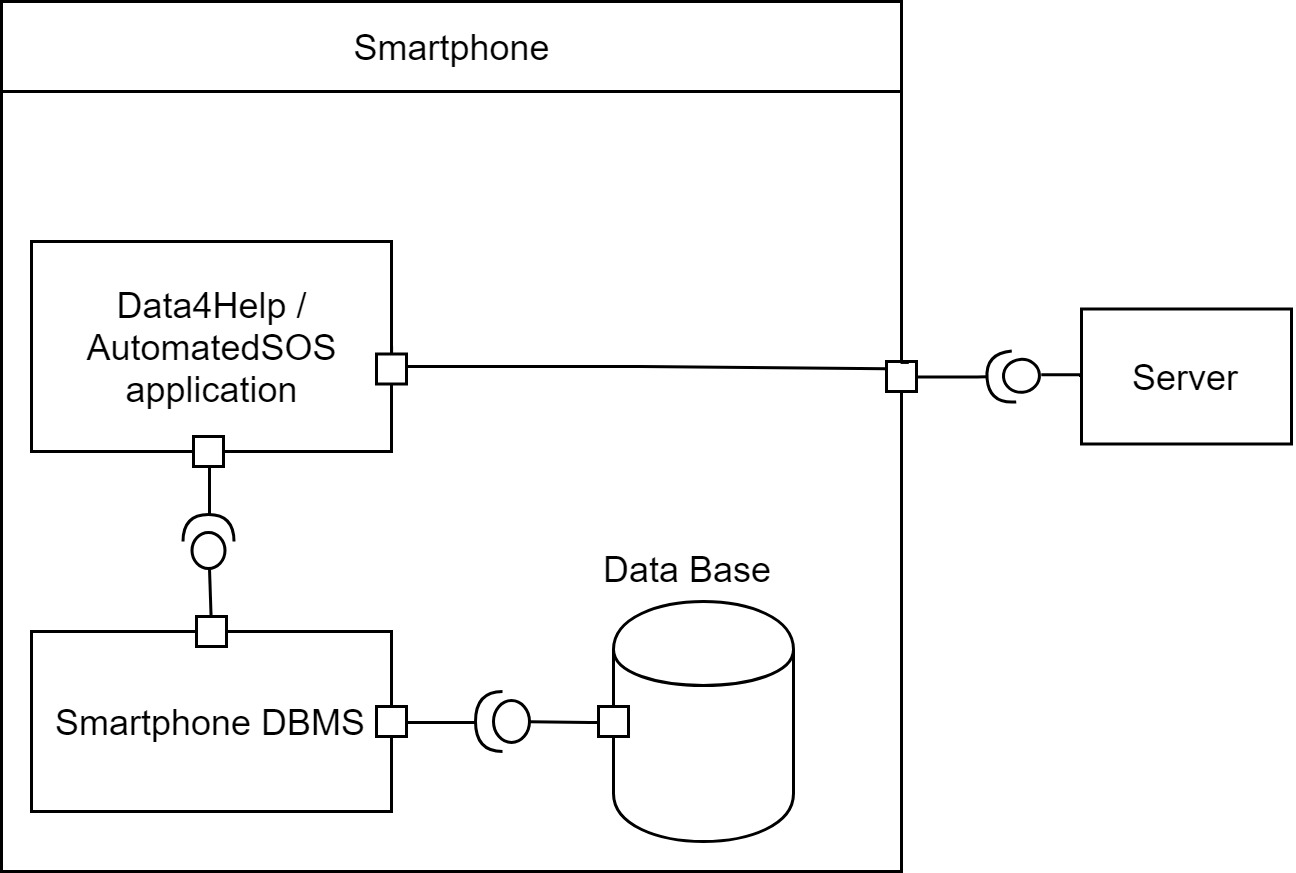


Figure 7: Component view system smartphone

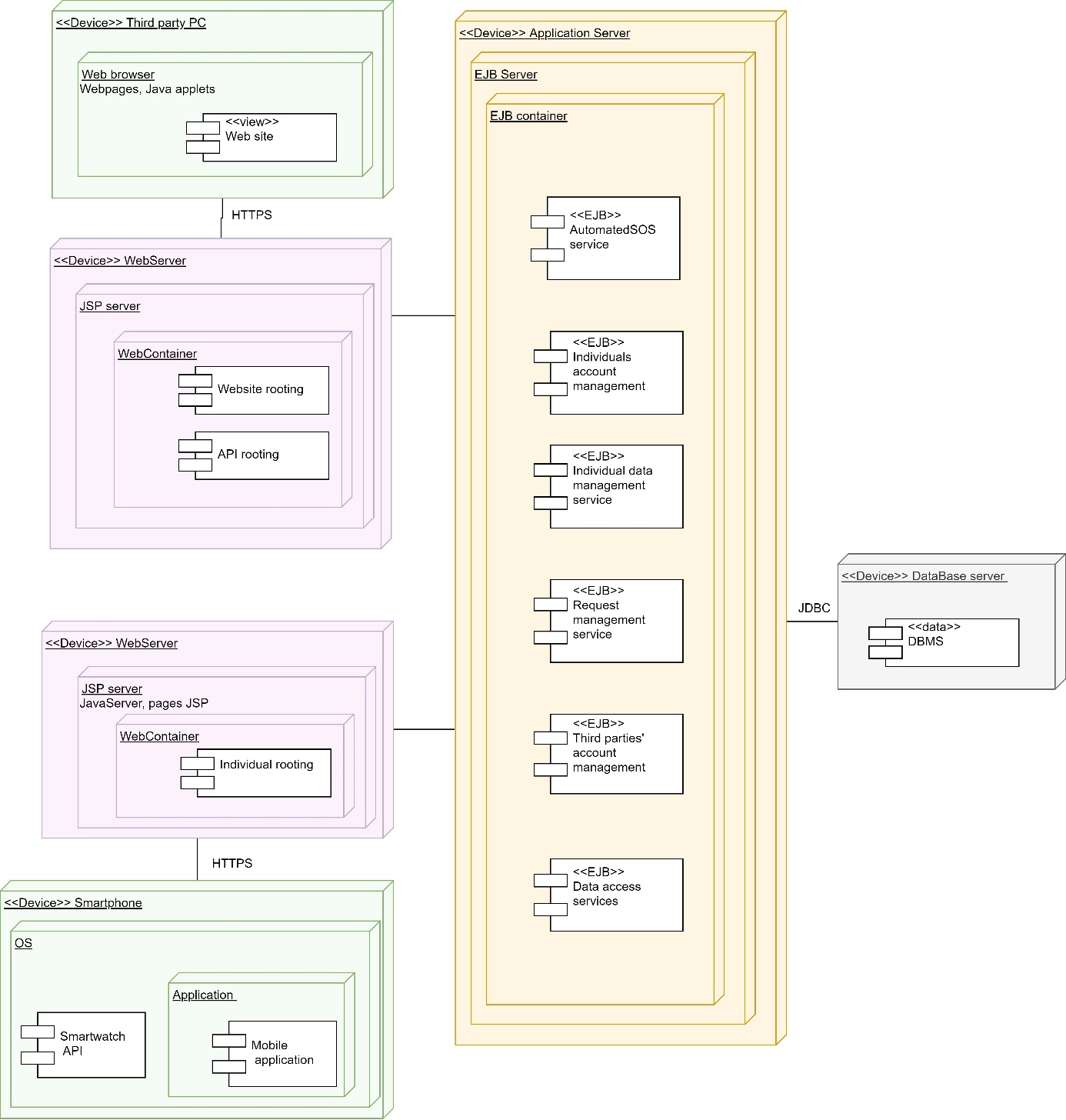
## Deployment view

The following deployment diagram represents the system's use of the physical infrastructure and how the system components are distributed and how they relate to each other.

The application Data4Help and AutomatedSOS require the following devices: A smartphone (for individuals) and a computer (for third parties). Those devices send information to a Webserver. There are two distinct Webservers, one that is in charge of the requests from the smartphone and one that is in charge of the requests from the computer.

The Webserver component communicate with the application server. The application server contains the business logic and communicates with the Webservers and the database server.

The database server stores all the data that is necessary for the Data4Help and AutomatedSOS services. Concerning the data of individuals, the database stores the account information, the history of their health and location data and their thresholds (if they subscribed to AutomatedSOS). Concerning third parties, the database stores their account information, their request and their subscriptions. The database also stores information on ambulance services (phone number and location).



## Runtime view

In this section we represent some runtime views of the interactions between the components of the Data4Help and AutomatedSOS systems.

### A third party makes an anonymized request

Third parties can make individual or anonymized requests. In the case where a third party submit an anonymized request, the validation (or non-validation) of the request can be done immediately as the choice does not involve a response for the individuals concerns.

The third party submits the anonymized request though the website interface. Then, the website rooting service transfers the request to the Request management service that carries out the validation.

The validation function implies an SQL request to the DBMS in order to obtain the number of individuals whose data satisfy the request. Then the status of the requests (validate or refused) is updated in the database and the response is communicated to the third party.

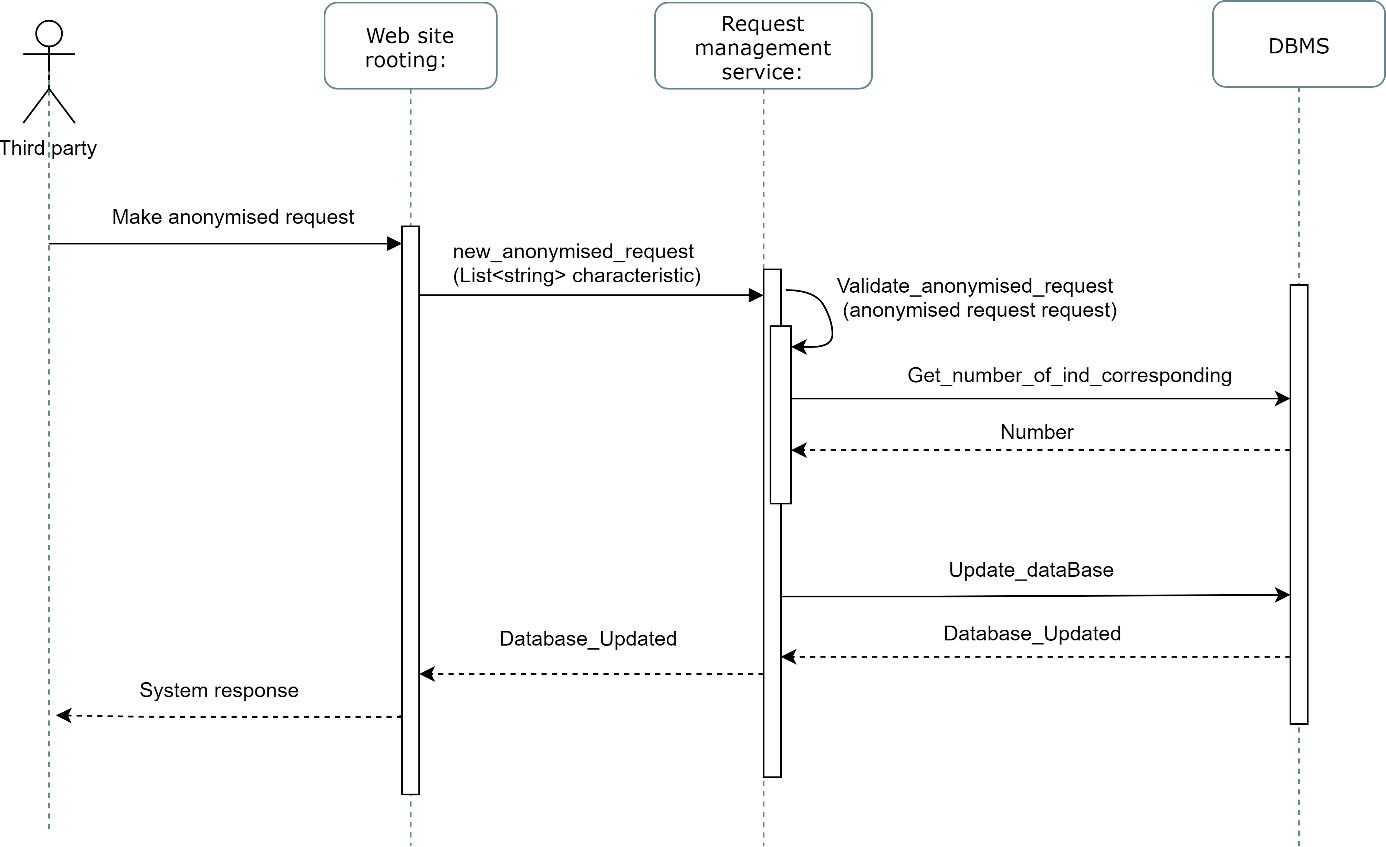


Figure 8: A third party makes an anonymized request

### A third party makes an individual request

Third parties can make individual or anonymized requests. In the case of an individual request the validation of the request is based on the response of the individual concerned.

The third party submits the anonymized request though the website interface. Then, the website rooting service transfers the request to the Request management service that saves it in the database with the status “waiting for validation”. Then the third party receives a confirmation message that his request has been saved.

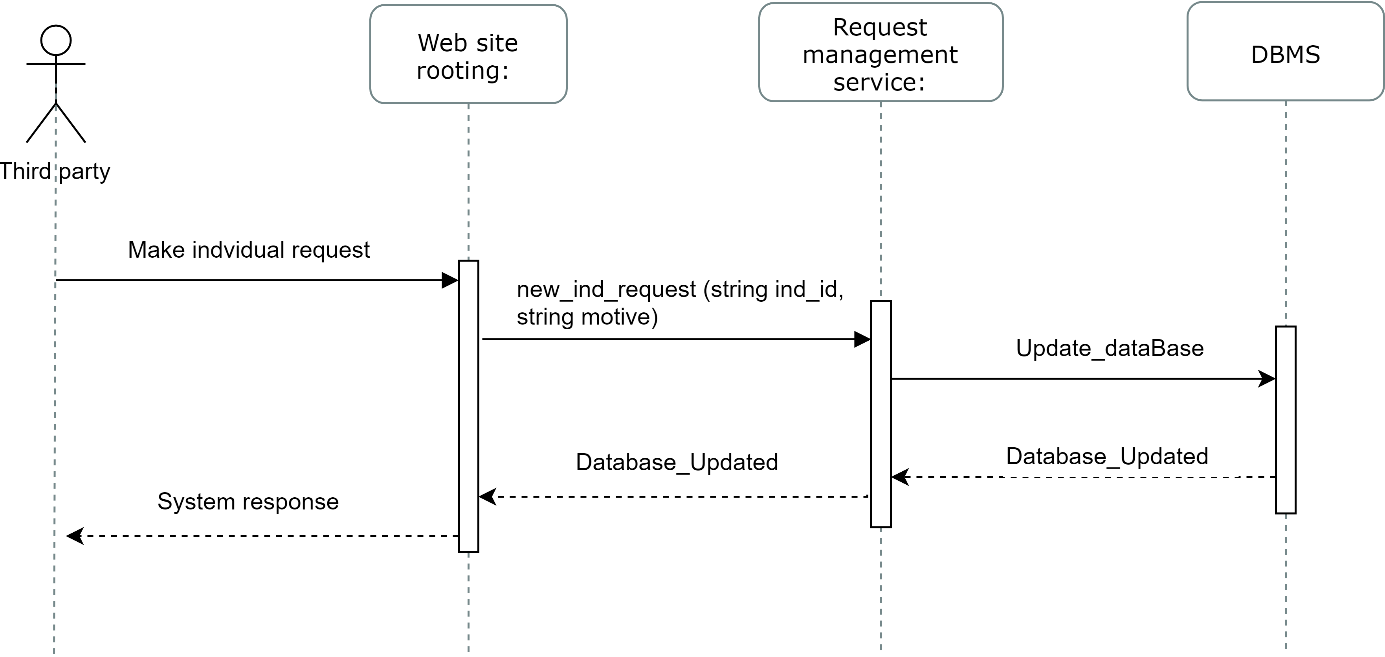


Figure 9: A third party makes an individual request

### An individual responds to a request

Individuals may be the subject of requests from third parties. Whenever an individual logs in on the smartphone application, the application rooting service make a request to the DBMS asking for the unanswered requests. Then the application displays on the screen of the user’s smartphone all the requests that are waiting for a response.

For each unanswered request, the individual can make the choice between accepting or refusing it by pressing a button. The individual’s response is then transferred to the application rooting to the request management service that transfer the modification of the request’s status to the database. Then, a confirmation message is printed to the user’s smartphone screen if the request’s new status is successfully saved in the database

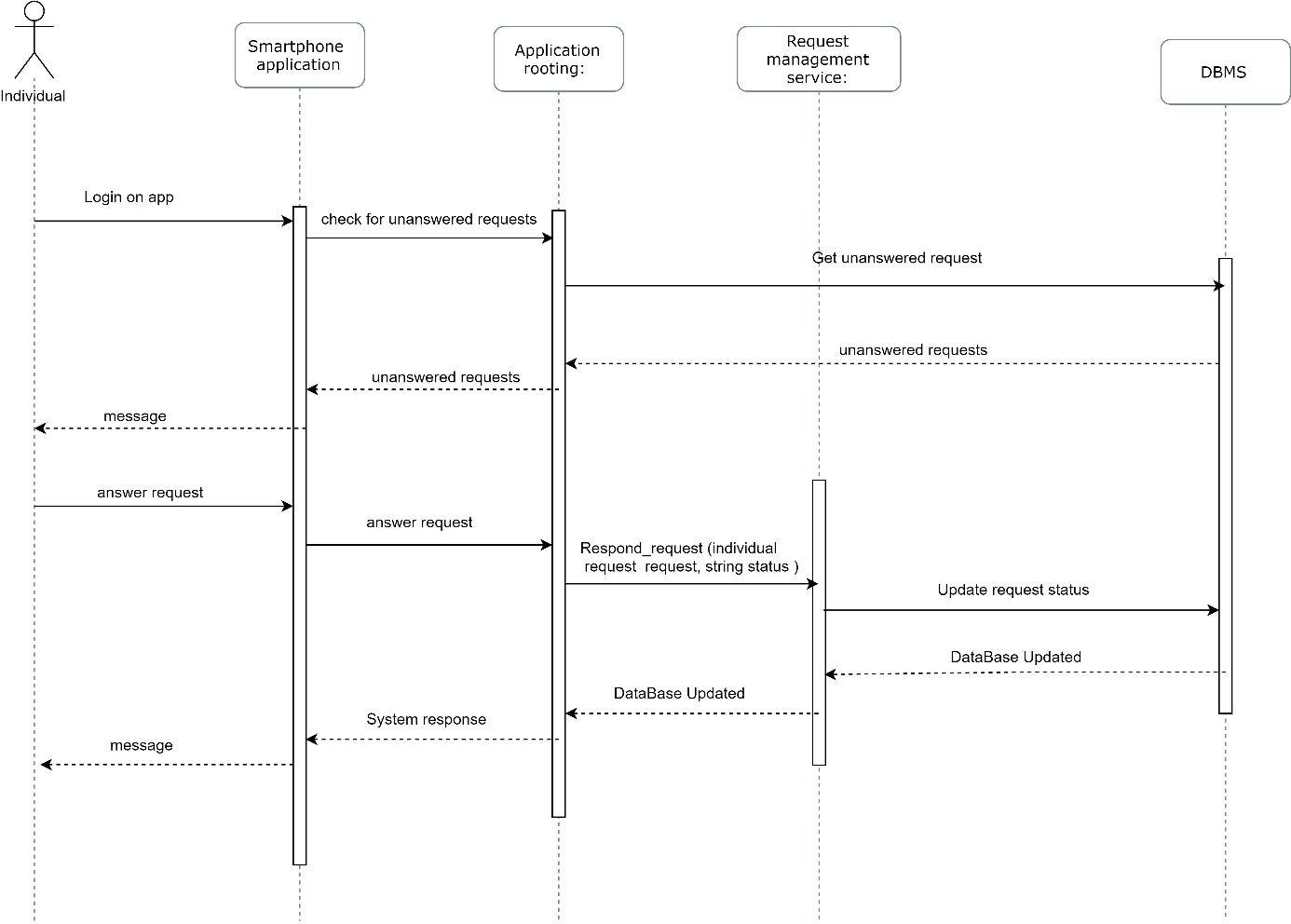


Figure 10: An individual responds to a request

### A third party subscribes to a request

Third parties have the possibility to subscribe to their accepted requests. By subscribing to a request, a third party has the possibility to download the newly available data corresponding to the request without going trough the validation process. When a third party makes request and when that request is accepted, the third party has access to the data until the date on which the request was made. When a third party subscribe to a request, he has access to the up to date data as long as his request stay valid.

When a third party logs in the Data4Help website, if he has an accepted request, he has the possibility to subscribes to it (if a request is not accepted, the “subscribe button” does not appear). When the third party clicks on subscribe, the website rooting transfers the request to the Request management service which transfers it to the data access service that asks to the DBMS the status of the requests.

If the status is accepted, the status of the request is updated in the database and the third party receives a confirmation message.

This double check increases the security and avoid third party being able to access data from unwilling individuals.

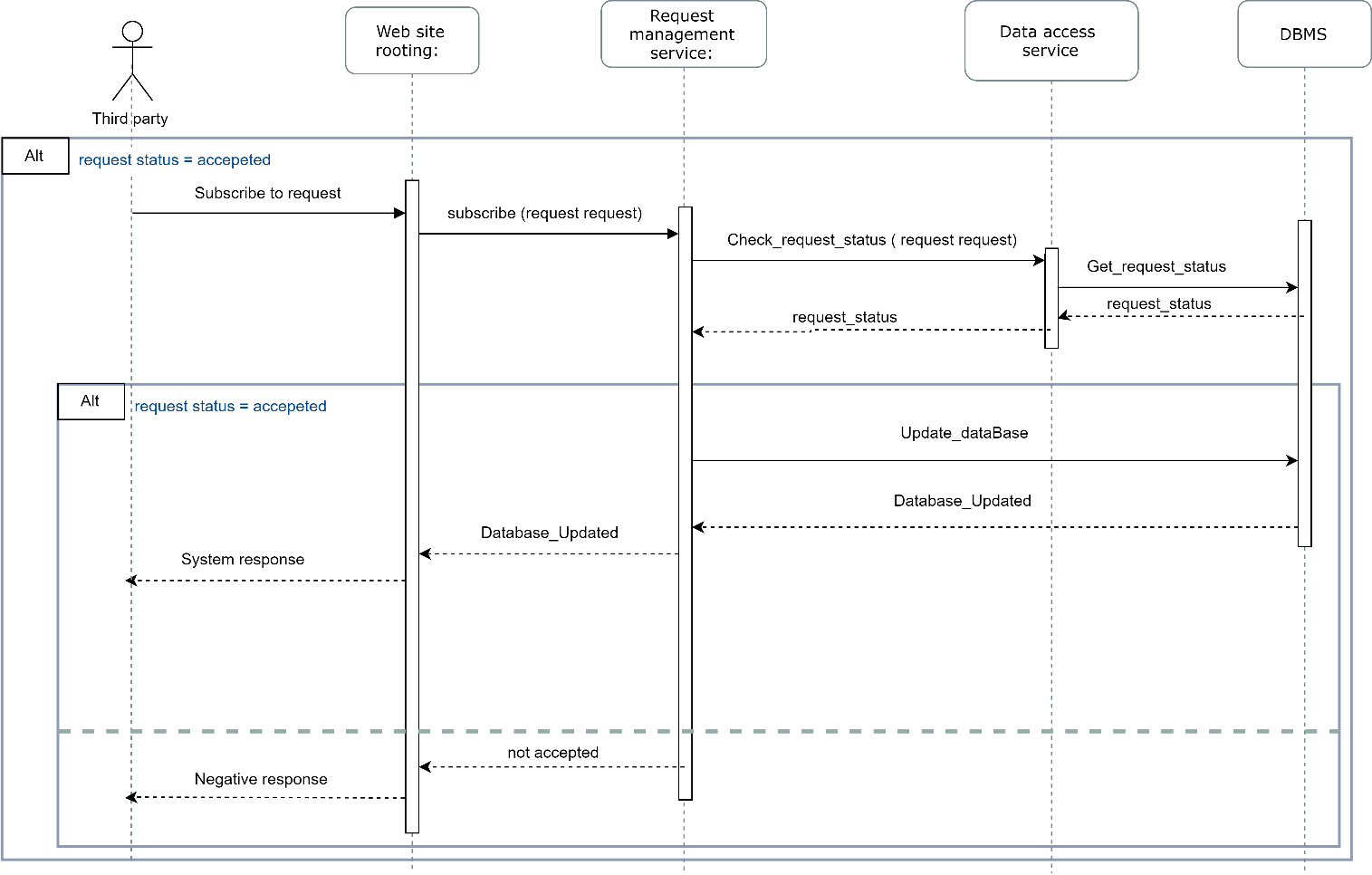


Figure 11: A third party subscribes to a request

### A third party download data from an anonymized request

When a third party logs in the Data4Help website, a request is made to the Website rooting service in order to know what needs to be displayed on the third party’s computer screen. If the third party has some accepted requests, the download button will appear on his screen.

When the third party clicks on the download button, the Website rooting service requests to the Request management service to validate the anonymized request. This requires calling the DBMS to count the number of individuals concerned.

Without this double check, if a third party never refreshed the webpage since the moment his request was accepted (so since the moment the download data button was available) and if during this time the number of individuals concerned by the request drop under 1000, a third party would be able to acquire unauthorized data.

If the request is anonymizable, the Website rooting service asks the DBMS to get the needed data. Then the data access service anonymizes the data and sends the result to the third party.

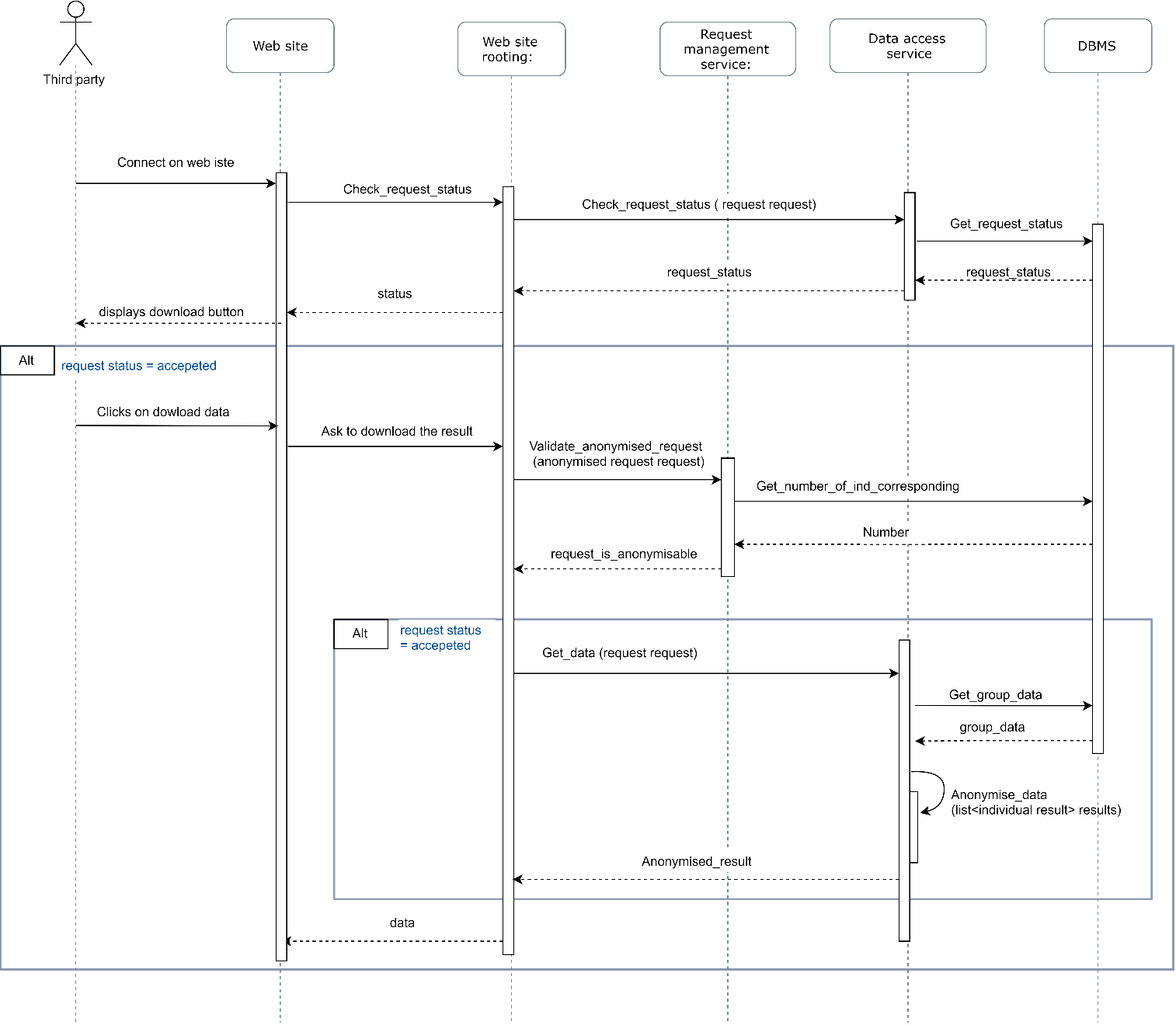


Figure 12: A third party download data from an anonymized request

### A third party download data from an individual request

When a third party logs in the Data4Help website, a request is made to the Website rooting service in order to know what needs to be displayed on the third party’s computer screen. If the third has some accepted requests, the download button will appear on his screen.

When the third party clicks on the download button, the Website rooting service requests to the Data access service to get the required data. To do so, the data access service checks the request status and if the request is accepted, ask the needed data to the DBMS. Then the data is sent to the third party.

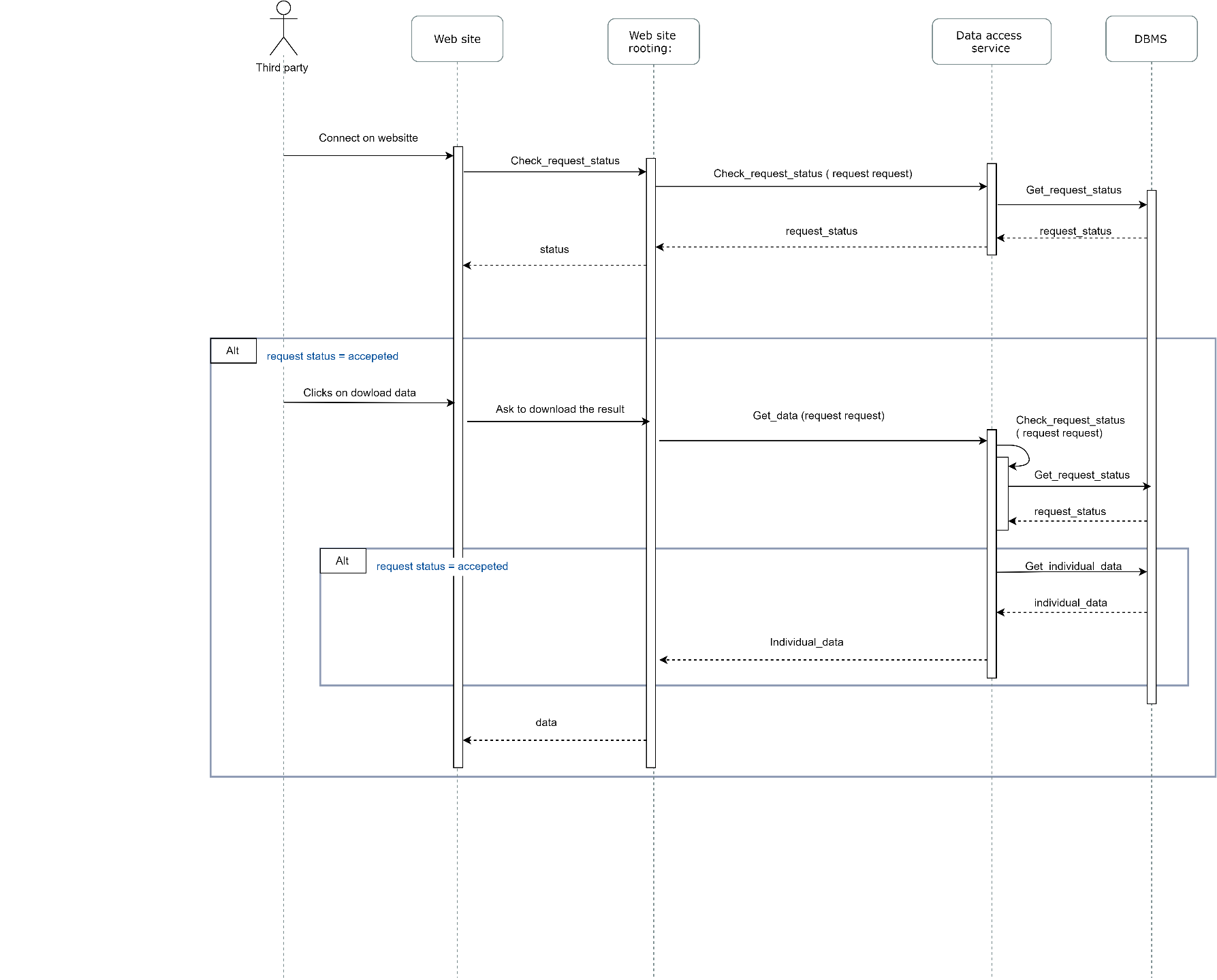


Figure 13: A third party download data from an individual request

### AutomatedSOS

The service AutomatedSOS must send an SOS to an ambulance service when the health parameters of the user are below certain thresholds. The service must therefore be functional even without an internet connection.

To do so, the emergency data (the thresholds and the phone number of the closest emergency service) must be regularly updated and stored in the smartphone internal database when an internet connection is available (every five minutes). This is what is shown on the first sequence diagram (figure XXX).

Therefore the AutomatedSOS service will be able to detect when the individual is bellow threshold and to send the emergency SMS using the data collected when an internet connection was available.

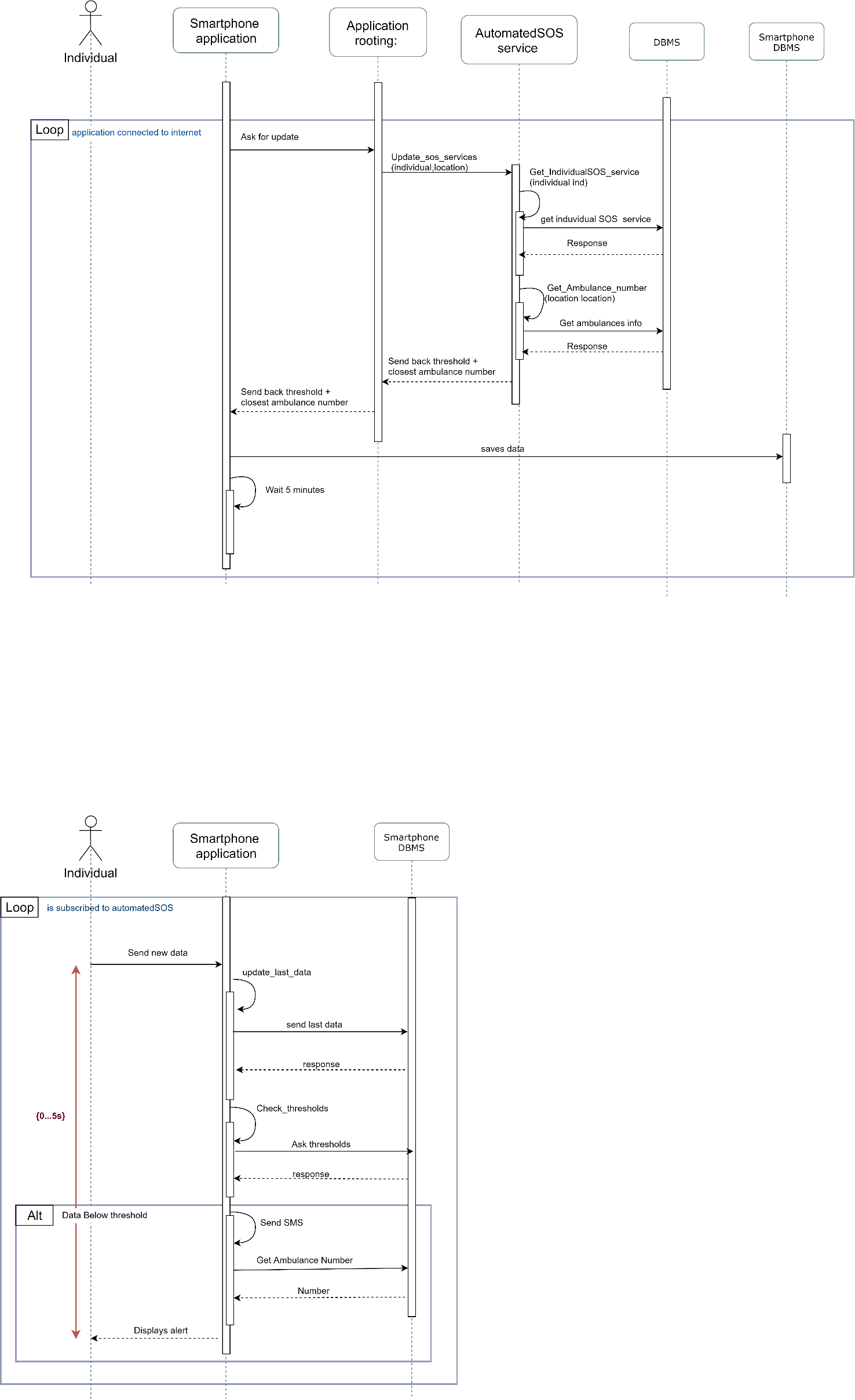


Figure 14: Updating emergency information when online

The following diagram shows the process of sending an emergency SMS. This process does not need any internet connection has it uses only the data stores in the smartphone database.

The smartphone application acquires new data from the users, saves the data in the internal database and check if the thresholds are exceeded. If they are, the smartphone sends an SMS to the closest emergency service. The process from the acquisition of new data to the sent of the SMS must last less than 5 seconds.

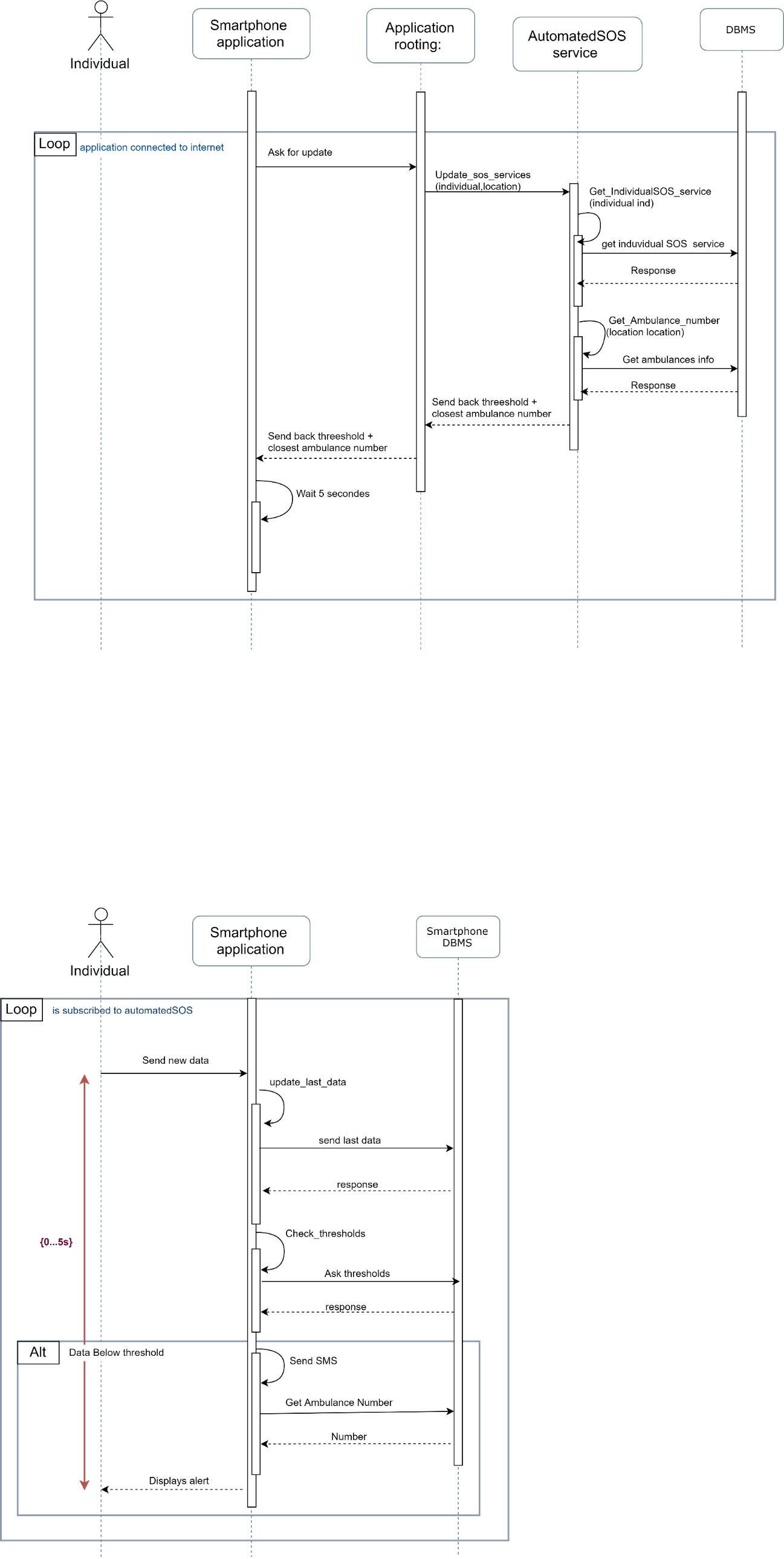


Figure 15: Sending SOS

### The application sends new data to the Data4Help server

The Data4Help application sends the health and location data of the individual from the smartphone data base to the Data4Help server.

The data is retrieved in the smartphone database and is sent by the Application rooting service to the Individual data management service that send the new data to the Data4Help DBMS.

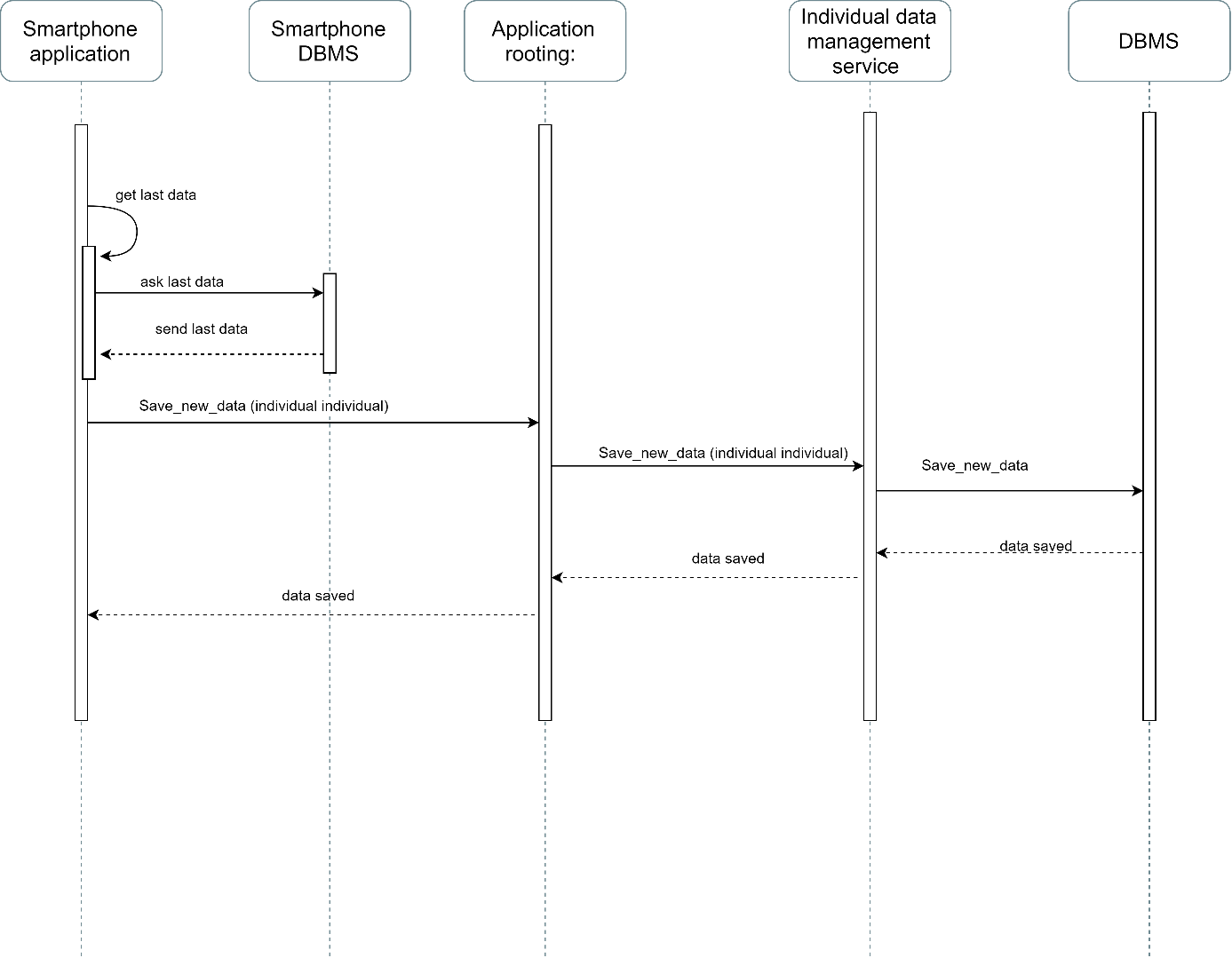


Figure 16: The application sends new data to the Data4Help server

## Component interfaces

ADD PART OF MOHAMED

## Selected architectural styles and patterns

ADD PART OF MOHAMED

# User interface design

The two end users are third parties and individuals. The user interfaces are presented for both types of users in the RASD document.

# Requirements traceability

In the following table presents a mapping correspondence between the requirements defined in the RASD related to each goal and the components identified in the server component diagram.

|  |  |  |
| --- | --- | --- |
| Goal | Requirement | Component |
| [G1] : Third parties must be able to request to access to the data of specific individuals or to anonymized groups of individuals. | R.8 ; R.9 | DBMS, request management service, Data access service, Individual data management service |
| [G2] : At any time, third party should never have access to data of specific individuals without their agreement. | R.10 ; R.11 | DBMS, Data access management system |
| [G3] : Third parties must have the possibility to subscribe to new data if their request is accepted. | R.12 | DBMS, request management system |
| [G 4] : Individuals must be able to consult their data and accept/refuse requests | R.13 | DBMS, request management system, Individual data management service |
| [G1], [G2], [G3] & [G4] | R.1 ; R.2 ; R.3 ; R.4 ;  R.5 ; R.6 ; R.7 | DBMS, Individual account management, Third partie’s account management |
| [G 5] : An ambulance is requested to the location of the individual with a reaction time below 5 seconds from the time the parameters are below threshold | R.14 ; R.15 ; R.15 ; R.17 | DBMS, AutomatedSOS service, Individual account management |

# Implementation and test plan

ADD PART FROM MOHAMED

# Effort spent

## Mohamed

ADD PART FROM MOHAMED

## Emma

|  |  |  |
| --- | --- | --- |
| Date | Work | Hours |
| 18/11/2018 | Writing the main components in draft form | 1 |
| 22/11/2018 | Component view diagrams | 1 |
| 25/11/2018 | Component view diagrams | 1 |
| 27/11/2018 | Component view + deployment view | 2 |
| 1/12/2018 | Component view + deployment view | 2 |
| 2/12/2018 | Deployment view diagrams | 1 |
| 3/12/218 | Added explanation text on Component view | 1 |
| 5/12/2018 | Added explanation text on Deployment view +  Runtime view diagrams | 4 |
| 6/12/2018 | Runtime view diagrams | 3 |
| 7/12/2018 | Runtime view diagrams + Explanation text | 2 |
| 8/12/2018 | Update on runtime view +  Requirements traceability | 3 |

# References

[1] SOMMERVILE, Iam. *Software engineering 9*. International edition.

[2] Assignment *Mandatory Project: goal schedule, and rules*.

[3] Les numériques, *COMPARATIF / Quelle montre connectée choisir ?*

https://www.lesnumeriques.com/montre-connectee/comparatif-montres-connectees-a1781.html