



POLITECNICO

MILANO 1863

Data4Help

Lorenzo, Molteni, Negri

ATD - Acceptance Test Deliverable

January 20, 2019

Contents

1	Introduction	2
1.1	Purpose and Scope	2
1.2	References	2
1.3	Project analysed	2
1.4	Overview	2
2	Installation Setup	3
2.1	Server	3
2.2	Third Party client	3
2.3	Individual client	3
3	Acceptance Test Cases	4
3.1	Third Party - Registration	4
3.2	Individual - Registration	5
3.3	Login	5
3.4	Third Party - Settings Management	6
3.5	Individual - Settings Management	7
3.6	Individual - AutomatedSOS Service	7
3.7	Individual - Data Acquisition	8
3.8	Data Management	8
3.9	Individual Requests	9
3.10	Anonymous Requests	10
4	Summary	12

1 Introduction

1.1 Purpose and Scope

The purpose of this document is to validate the assigned project with regards to the installation procedure, the testing of implemented requirements, and the overall coherence and quality of documentation.

1.2 References

The documents used to derive all Acceptance Test cases, and perform installation, are:

- RASD: Requirements Analysis and Specification Document
- DD: Design Document
- ITD: Implementation and Testing Document

1.3 Project analysed

The authors of the project we had to review are:

- Paolo Romeo: paolo1.romeo@mail.polimi.it
- Andrea Scotti: andrea5.scotti@mail.polimi.it
- Francesco Staccone: francesco.staccone@mail.polimi.it

Their GitHub repository can be found at: <https://github.com/dev-strenuus/RomeoScottiStaccone>.

1.4 Overview

The remaining parts of the document are organised as follows:

- **Installation Setup:** explains what we did in order to install the System and run it properly
- **Acceptance Test Cases:** contains the acceptance test cases defined and their outcomes
- **Summary:** is an overall consideration regarding the quality of the documentation and coherence with the implementation

2 Installation Setup

Installation has been performed on MacOS Mojave 10.14.2 by all members of our group.

2.1 Server

To install the server we unzipped the executable folder and ran the command provided in the ITD to execute the jar.

Due to LaTeX changing the “ into ”, copy-paste of the command to run the server on the Terminal doesn’t work, so we had to insert the correct quote character. Apart from this issue caused by LaTeX, the server is immediate to setup and run.

2.2 Third Party client

To access the interface of a Third Party, we tried using Safari, Chrome and Firefox to connect to the provided ip (with *ipserver:port* set to localhost:3000). No issues were encountered.

2.3 Individual client

Node.js was required to access the interface of an *Individual*, but its installation was not necessary because all members of our group already have it installed, along with NPM.

To install Cordova, Ionic and Phonegap we used the provided command, but had to add “sudo” to give permissions to install the dependencies globally, as most Macs are not setup with super user permissions.

For what concerns Phonegap, the iOS application is no longer available on the AppStore, but we managed to borrow an Android device to try the mobile experience. Nonetheless, after contacting the team, they also recommended us to try a browser as well, without having to launch Phonegap.

Safari does not display a date picker, so we could not use it for signup. We instead used Chrome, and had to disable Web Security by running it through the Terminal: “open -a Google Chrome -args -disable-web-security -user-data-dir”. After signup was successful, we could switch back to Safari, without having further issues.

For the Android application we carefully followed the installation guide provided in the ITD document without any relevant issue. The interface and functionalities are exactly the same available for the web version.

3 Acceptance Test Cases

We tested the functionality of each implemented requirement in the ITD, both with positive and negative outcomes. For each test, we provide a brief description of how it was performed, its desired outcome and its actual outcome. HTTP codes are retrieved using Chrome Developer Tools.

3.1 Third Party - Registration

- **R15)** Two different users cannot have the same username.
- **R16)** The system must allow the third party to register to the application, by specifying its VAT registration number, name and password.

Test description	Desired outcome	Test outcome
Register a user providing correct informations	Registration is successful, user is brought back to login page	HTTP 200; Test passed
Register a user providing a VAT already used	Registration fails, user is prompted to insert a different VAT	HTTP 409; Test passed
Register a user providing a Name already used	Registration fails, user is prompted to insert a different Name	HTTP 409; Test passed
Register a user without accepting the agreement	Registration fails, user is prompted to accept the agreement	Request is not sent; Test passed
Register a user with a VAT not long 11 characters	Registration fails, user is prompted to insert a valid VAT	Request is not sent; Test passed
Register a user with an empty Name or Password	Registration fails, user is prompted to insert a valid Name or Password	Request is not sent; Test passed

Table 1: Third Party Registration suite

For what concerns R13, the VAT has to be unique for every Third Party, thus being the "username" mentioned in the requirement. As we can see though, the Name of the Third Party must also be unique, but this is not specified in the documentation, and can be confusing since for Individuals the (Name, Surname) couple doesn't have to be unique, but only its Fiscal Code does.

3.2 Individual - Registration

- **R15)** Two different users cannot have the same username.
- **R13)** The system must allow the individual to register to the application by selecting a password and providing his/her data, fiscal code included.

Test description	Desired outcome	Test outcome
Register a user providing correct informations	Registration is successful, user is brought back to login page	HTTP 200; Test passed
Register a user providing a Fiscal Code already used	Registration fails, user is prompted to insert a different Fiscal Code	HTTP 409; Test passed
Register a user without accepting the agreement	Registration fails, user is prompted to accept the agreement	Request is not sent; Test passed
Register a user with a Fiscal Code not long 16 characters	Registration fails, user is prompted to insert a valid Fiscal Code	Request is not sent; Test passed
Register a user with an empty Name, Surname or Password	Registration fails, user is prompted to insert a valid Name, Surname or Password	Request is not sent; Test passed
Register a user with a Birthdate not matching dd/mm/yyyy	Registration fails, user is prompted to insert a valid Birthdate	HTTP 400; Test partially failed : the check should be done without sending the request
Register a user with an empty Latitude or Longitude	Registration fails, user is prompted to insert a valid Latitude or Longitude	Request is not sent; Test passed

Table 2: Third Party Registration suite

3.3 Login

- **R17)** The system must allow the third party to log in to the application by providing the combination of a VAT registration number and a password that match an account.
- **R14)** The system must allow the individual to log in to the application by providing the combination of a fiscal code and a password that matches an account.

Test description	Desired outcome	Test outcome
Log in a third party account with valid credentials	Login is successful, user is brought to the third party user home page	HTTP 200; Test passed
Log in the third party website with malformed VAT	Login fails	Request not sent; Test passed
Log in the third party website with credentials not matching an account	Login fails	HTTP 401; Test passed
Log in an individual account with valid credentials	Login is successful, user is brought to the individual user home page	HTTP 200; Test passed
Log in the individual website with malformed fiscal code	Login fails	Request not sent; Test passed
Log in the individual website with credentials not matching an account	Login fails	HTTP 401; Test passed

Table 3: Login suite

3.4 Third Party - Settings Management

- **R33)** The system must allow the Third party to change its password.

Test description	Desired outcome	Test outcome
Change password with a new one of long length (13 characters)	Change password successful	HTTP 200; Test passed
Change password with a new one of short length (4 characters)	Change of password successful	HTTP 422; Test failed: password not changed

Change password providing wrong old password	Change password fails	HTTP 422; Test passed
--	-----------------------	------------------------------

Table 4: Third Party Settings Management suite

3.5 Individual - Settings Management

- **R31)** The system must allow the user to change his/her personal info.
- **R32)** The system must allow the individual to change his/her password.
password.

Test description	Desired outcome	Test outcome
Change password with a new one of long length (13 characters)	Change of password successful	HTTP 200; Test passed
Change password with a new one of short length (4 characters)	Change of password successful	HTTP 422; Test failed: password not changed
Change password providing wrong old password	Change of password fails	HTTP 422; Test passed
Change coordinates with valid new ones	Change of coordinates successful	HTTP 200; Test passed
Change coordinates with invalid new ones (e.g. latitude 300)	Change of coordinates not successful	HTTP 200; Test failed: the request should not go through

Table 5: Individual Settings Management suite

You can insert invalid values for Latitude and Longitude, but this means creating garbage data.

3.6 Individual - AutomatedSOS Service

- **R25)** The AutomatedSOS service must be enabled.
- **R18)** When the health status values go below the threshold, the system

must send an SOS within 5 seconds.

- **R26)** The system must allow the user to enable/disable the Automated-SOS service at any time. password.

Test description	Desired outcome	Test outcome
Enable automated-SOS	Enable automatedSOS successful	HTTP 200; Test passed
Disable automated-SOS	Disable automatedSOS successful	HTTP 200; Test passed

Table 6: Individual AutomatedSOS Service suite

3.7 Individual - Data Acquisition

- **R27)** The system must be able to store data retrieved from registered users.
- **R35)** The system must be able to aggregate data based on the location of the individuals residence. password.

Test description	Desired outcome	Test outcome
Try to access health data of a group of individuals that does not match the location filters specified in the anonymous request.	No data are shown	No data are shown Test passed

Table 7: Data acquisition suite

3.8 Data Management

- **R11)** The system is optimized to send the data received from the mobile application to the third parties as soon as possible.
- **R9)** The third party is not allowed to access the users data until he/she accepts the request.
- **R2)** The system must be able to provide to the third party the location and the health status of individuals.

- **R1)** The users must have given the consensus to the treatment of their information to the third party. password.

Notice that the following tests have been performed on dummy data given that, as expressed in the ITD, R3 and R35 have not been implemented yet. Also the last part of R2 has not been implemented and as a consequence location data will not be considered during the tests.

Test description	Desired outcome	Test outcome
Try to access health data of a pending single request	Data are unavailable	Function unavailable; Test passed
Try to access health data of an accepted single request	Data are available	Health data are shown after pressing the "watch" button in the notification Tab; Test passed
Try to access health data of a refused single request	Data are unavailable	Function unavailable; Test passed
The users must have given the consensus to the treatment of their information to third parties	User consensus is explicitly asked	User consensus is given in the agreement accepted during the signup phase and indirectly when accepting a request Test passed

Table 8: Data Management suite

3.9 Individual Requests

- **R28)** The user must have an active subscription to stop it.
- **R29)** The system must be able to allow the user to unsubscribe to the third party and to stop the transmission of his/her data.
- **R7)** The system must be able to forward the requests from the third party to the user;
- **R8)** The system must save the preference of the user.
- **R37)** The system must allow the third party to make individual requests.

Test description	Desired outcome	Test outcome
------------------	-----------------	--------------

Send individual request with invalid fiscal code	Send individual request fails	HTTP 409; Test passed
Send individual request with valid fiscal code	Individual request sent and received by individual user	HTTP 200; Test passed
Accept individual request from third party	Individual request accepted and third party is notified	HTTP 200; Test passed
Refuse individual request from third party	Individual request refused and third party is notified	HTTP 200; Test passed
Send individual request to individual who has refused a previous request sent by the same third party	Individual request is sent	HTTP 409; Test failed: the request is not sent and the user is prompted to check if the fiscal code is correct
Send individual request to individual who has refused a previous request sent by a different third party	Individual request is sent	HTTP 200; Test passed

Table 9: Individual Requests suite

Even though we could not import real data from a device for an Individual account, we were able to test Individual requests using randomly created data through an ad hoc button in the Individual home page.

3.10 Anonymous Requests

- **R5)** The system must be able to provide to the third party the health status of individuals in an anonymous way.
- **R4)** The groups must be composed at least by 1000 individuals.
- **R6)** The system must be able to aggregate the data of the individuals, as requested by the third party.
- **R36)** The system must allow the third party to make group requests.

Note that to make the testing phase more accessible the developers added a command line parameter, "*DanonymousSize*", to set the minimum number of individuals needed to satisfy an anonymous request. While performing the tests this parameter has been set to 2.

Test description	Desired outcome	Test outcome
Send anonymous request with valid filters	Anonymous request sent successfully	HTTP 200; Test passed
Send anonymous request with invalid filters (negative values for age, values out of range for latitude and longitude)	Anonymous request sent successfully	HTTP 200; Test failed: user should be informed of malformed filters
Try to access health data related to an anonymous request concerning more than the number of individuals specified in the params	Health data shown successfully	Health data are shown after pressing the "watch" button in the notification Tab; Test passed
Try to access health data related to an anonymous request concerning less than the number of individuals specified in the params	Health data shown successfully	No health data are shown Test passed

Table 10: Anonymous Requests suite

R4 is not satisfied by the system. We have verified this after sending an anonymous request with malformed filters (successfully sent as mentioned in the above table). The response contained an empty data set, thus showing that the check on the 1000 individuals was not performed correctly.

4 Summary

Overall we reckon that the application works as expected and the implemented requirements and goals have been almost completely reached. In the remaining part of this section we will highlight some of the features that we think should be improved as well as what are the good elements of the project.

When performing testing, we noted that most fields (except for VAT and Fiscal Code) do not have explicit range limits. A name, for instance, may have a length that ranges from 1 to the maximum value possible for that data type. This may cause security issues for the company (database attacks).

We did not count this as an error, since the documentation does not specify constraints on those variable, and implementing them is quite straightforward. On the other hand, a password has hidden constraints, since it cannot be of arbitrary length, but the user is not informed of this and it's not written in the documentation.

The user interface is simple, responsive and presents its functionalities in an ordered and structured manner. However, even though the outcome of some tests is correct, we found the lack of feedback confusing, as when you press a button sending a HTTP request you don't know if it was successful or if it failed (and why).

As a final note we wanted to highlight the fact that the developing team has been always available to provide answer and support to any question.