



POLITECNICO MILANO 1863

TrackMe
Software Engineering 2 Project
ATD Document

Stefano Martina, Alessandro Nichelini, Francesco Peressini

A.Y. 2018/2019
Version 1.0.0

January 20, 2019

Contents

1	Introduction	3
1.1	Purpose and Scope	3
1.2	Reference	3
1.3	Overview	3
1.4	Revision History	3
2	Project	4
2.1	Project info	4
2.1.1	Authors	4
2.1.2	Links	4
3	Installation Setup	5
3.1	Backend	5
3.1.1	Installation and launch	5
3.1.2	Conclusion	5
3.2	Frontend	6
3.2.1	Installation and launch	6
3.2.2	Conclusion	6
4	Acceptance Tests	7
4.1	Tests	7
4.2	Actor registration	7
4.3	Actor Authentication	7
4.4	Individuals Management	7
4.5	Data management	8
4.6	Query management	8
4.6.1	Issues	8
4.7	Race management	9
4.7.1	Issues	9
4.8	Users Spectating Race	9
4.8.1	Issues	9
5	Conclusion	10
6	Effort spent	11

1 Introduction

1.1 Purpose and Scope

The Acceptance Test Document has the purpose of evaluating the adherence of the implementation with respect to the documents previously delivered.

To do so, we have considered the three documents available in the provided GitHub repository.

1.2 Reference

As said above, the three documents analysed are:

- **ITD Document (v.1.0):** [link](#)
- **DD Document (v.1.1):** [link](#)
- **RASD Document (v.1.2):** [link](#)

1.3 Overview

The document is organised in this way:

- **Project:** the identification of the project we have analysed;
- **Installation Setup:** explains what we did to install the prototype of the tested project as well as any problem or incoherency we have found following the provided documentation;
- **Acceptance Tests:** the acceptance test cases we have applied and the corresponding outcome.

1.4 Revision History

- 1.0.0 - Initial version (20/01/2019)

2 Project

2.1 Project info

In this section we present the details about the project we have tested.

2.1.1 Authors

- Daniele Montesi - 912980
- Nicola Fossati - 915244
- Francesco Sgherzi - 915377

2.1.2 Links

- **GitHub repository:** [link](#)

3 Installation Setup

3.1 Backend

3.1.1 Installation and launch

We have installed the backend as explained in the ITD Document, with the only exception of the particular system-dependent command (we tested the project on macOS).

We installed the following module with *"brew"* command:

- **PostgreSQL**: an open source object-relational database system;
 - installed with the command: *brew install postgresql*
- **NodeJS**: an asynchronous event driven JavaScript runtime.
 - installed with the command: *brew install nodejs*

After the launch of the postgre service (with the command *brew services start postgresql*), we created a new database and a new role for the admin user. Then, with the new role just created, we imported the dump of the database provided in the Implementation folder of the previously mentioned repository. As last step we have configured the *"start.sh"* file with the following configuration and launched it with the command *"node app.js"*.



```
TEST_API="enabled"  
DATABASE_URL="postgres://admin:password@localhost:5432/test"  
JWT_SECRET="E9ql4cmZzDNG9qL8xh6F"  
MAIL_PROVIDER="gmail"  
MAIL_ADDR="mail@prova.it"  
MAIL_PASSWD="pass"  
LOCAL="enabled"  
PORT=12345  
HOST="localhost:${PORT}/v1"  
MIN_USER_NUMBER=2
```

Figure 1: Error in the database configuration

3.1.2 Conclusion

We didn't encounter particular issues, the guide lines are clear and enough explicative.

The only thing which makes sense to be mentioned is one error encountered during the database connection. It always responds with *"message: role //password// does not exist"*. This is due to a small error in the configuration file, probably to the fact that is a machine dependant parameter (view image 1).

3.2 Frontend

3.2.1 Installation and launch

Regarding the Android mobile application, we have installed the provided APK file on an Android Smartphone. We also tried to install it on a few simulators.

For what concerns the iOS app, we managed to install Flutter SDK and to build the application, however we would have preferred to have a more precise and coincide guide in the ITD document such the one provided for the Android environment.

Lastly, regarding the web-app, we launched it using the command *"python3 -m http.server"*, as explained in the guidelines.

3.2.2 Conclusion

Following the instructions on the ITD Document, on an Android smartphone everything works correctly. Doing the same steps on a simulator, the application crashes, probably due to some incompatibilities with the running version of the simulator itself.

iOS app correctly built and run on simulator.

The web-app works correctly.

4 Acceptance Tests

4.1 Tests

For the acceptance tests, we worked on the mobile Android application but the results can be considered valid also for the iOS application since, as explained in RASD and DD documents, both the applications share the same codebase. Moreover, the tested project doesn't use platform-dependent frameworks or technologies (with the exception of the Google Maps framework that has been reported as a known issue); all the results are actually valid for both platforms.

We proceeded testing each requirement that has been declared as "implemented" in the ITD document.

4.2 Actor registration

- [RM_M]: *OK*. Users can register providing information required, but registration process succeed also without an associated smartwatch.
- [R2_M]: *OK*. Implementation works: registration process is successful if there is NOT another user with the same email/fiscal code. However, errors-handling is not able to distinguish between a duplicate email or a duplicate fiscal code.
- [R11_W]: *OK*. Organisers can successfully register.
- [R1_W]: *OK*. Companies can successfully register.
- [R14_C]: *OK*. Information such as birthday and fiscal code are validated through UI form.

4.3 Actor Authentication

- [R1_M]: *OK*. Users can successfully log into the application.
- [R12_M]: *OK*. Organisers can successfully log into the application.
- [R2_W]: *OK*. Companies can successfully log into the application.

4.4 Individuals Management

- [R6_M]: *OK*. Requests appear in the right section of the mobile app. Users can actually accept or decline them.
- [R2_C]: *OK*. Users received notification about new requests at the email address provided during the registration phase.

4.5 Data management

- [R4_C]: *OK*. We can't directly test this feature, but it seems that all data loaded into the application is also available after a logout.
- [R5_C]: *OK*. Same as the previous one.
- [R2_S]: *OK*. Assuming the requirement [R2_S] as "*App can send data registered locally to Data4Help Web App*", the mobile application can actually send data to the Web App.

4.6 Query management

- [R6_W]: *OK*. Logged-in companies can actually query on some group of individuals data providing location, heart rate and accelerometer information.
- [R7_W]: *OK*. Companies can actually request data of individuals.
- [R8_W]: *OK*. Companies can access to individuals data through the web app.
- [R9_W]: *OK*. The web app actually provides the data download functionality.
- [R6_C]: *OK*. After the acceptance by the user, third parties can retrieve data from the individual correctly.
- [R6bis_W]: *OK*. The web app implement the subscription functionality through a slider. The company actually receives a notification when new data is available.
- [R7_C]: *KO*. Check the "Issues" section.

4.6.1 Issues

During the testing of requirements [R7_W] and [R8_W], we have found the following issue:

- the process fails if requests are made by a company that has the same email address of a user.
The error message is the following: "*Error: Unauthorised. Retry*" and it appears on the website as an alert. However, since it's unlikely that a company uses the same address of a user, we still marked the requirement as "OK".

4.7 Race management

- [R8_M]: *OK*. Registration is possible through the track4run tab at anytime.
- [R9_M]: *KO*. Check the "Issues" section.
- [R13_M]: *OK*. Organisers actually have all the necessary information to create a path.
- [R14_M]: *OK*. Organisers can define the path of the run and the starting/ending time and date.

4.7.1 Issues

Concerning requirement [R9_M], the application only displays the name of the run, its status and (partially) the position of participants. Information about the starting point, the ending point, the precise path and the run length are not available.

4.8 Users Spectating Race

- [R10_M]: *KO*. Check the "Issues" section.
- [R13_C]: *KO*. Check the "Issues" section.

4.8.1 Issues

Concerning the requirements [R10_M] and [R13_C], the application only displays a static pin on a Google Maps interface for each participants of the run.

We were not able to test if the application actually track the position of the runners, also considering that the authorisation to access to the actual location of the runners were never asked at no time.

5 Conclusion

Apart from the few issues described above and in the ITD document provided to us by the developers, the application is working as expected: all the major features described by the ITD document are present and ready to be used by a final user.

The user interface is pretty intuitive and easy to user, even at a first start; the initial set-up is super easy concerning the installation of the application for Android, a little bit difficult concerning the installation on an iOS device, also due to the strictly policy imposed by Apple.

Future update should focus on providing new features to the final user such as AutomatedSOS, for which is already present a section in the application.

6 Effort spent

- Stefano Martina: 5.00h
- Alessandro Nichelini: 5.00h
- Francesco Peressini: 5.00h