

This manual provides basic tips how to work with the OpenReq Live environment and thus provides insights into basic processes supported by OpenReq services.

Table of Contents

List of Figures	1
Introduction	2
User Registration	2
Dashboard.....	2
Projects Main Page.....	3
OpenReq Live Project Settings.....	4
Specify Rating Dimensions	5
Import Requirements from Twitter	5
Move Requirements	5
Delete Requirements.....	5
Project Settings	6
Prioritization of Requirements	6
Stakeholder Recommendation	7
Dependency Management in OpenReq Live.....	9
Statistics.....	11

List of Figures

Figure 1 Registration in OpenReq Live	2
Figure 2 Dashboard of OpenReq Live	3
Figure 3 Overview of an example project in Openreq Live.....	3
Figure 4 Settings Overview of OpenReq Live projects.....	4
Figure 5 Advanced options in OpenReq Live.....	4
Figure 6 Definition of requirement rating dimensions.....	5
Figure 7 Import Requirements from Twitter directly into OpenReq Live.	5
Figure 8 Advanced project settings.....	6
Figure 9 Argumentation-based rating interface	7
Figure 10 Stakeholder Assignment.....	8
Figure 11 Dependency management in OpenReq Live.	9
Figure 12 Import of recommended dependencies.....	9
Figure 13 Indication of inconsistencies in OpenReq Live.....	10
Figure 14 Details about the inconsistencies of the current requirement model.	10
Figure 15 Statistics about changes of the requirement's title, description and state.....	11
Figure 16 Indication of the dependencies in OpenReq Live.....	12

Introduction

In order to provide stakeholders a convenient and user-friendly access to the OpenReq services, we have developed an innovative, web-based RE platform called OpenReq Live. OpenReq Live is a freely available collaborative RE platform¹ which fosters the cross-fertilization of ideas between stakeholders and allows them to jointly manage their projects and take full advantage of the recommendation power provided by OpenReq's tool suite. OpenReq Live is capable of tackling all of the everyday RE tasks stakeholders face in their software projects.

From a technical viewpoint, the OpenReq platform represents a central hub which combines the functionality of the most relevant OpenReq services. The platform provides all necessary functionality to create and update the requirements model of a project which includes requirements, releases, dependencies, and further release-related constraints (such as the maximum requirement capacity defined for a release or the release date/deadline) of a project.

User Registration

There are two options to use the system. The *first option* is to use it as a user without registration. In this case, the system generates unique links for the management and participation in the created projects. The *second option* is to sign up for a user account (see Figure 1). Registered users have the advantage of an individual dashboard where all relevant projects can be seen at a glance. During the registration process, users will be asked about their personal data (mainly name and e-mail address) and after providing the information, a user account is "ready to go".

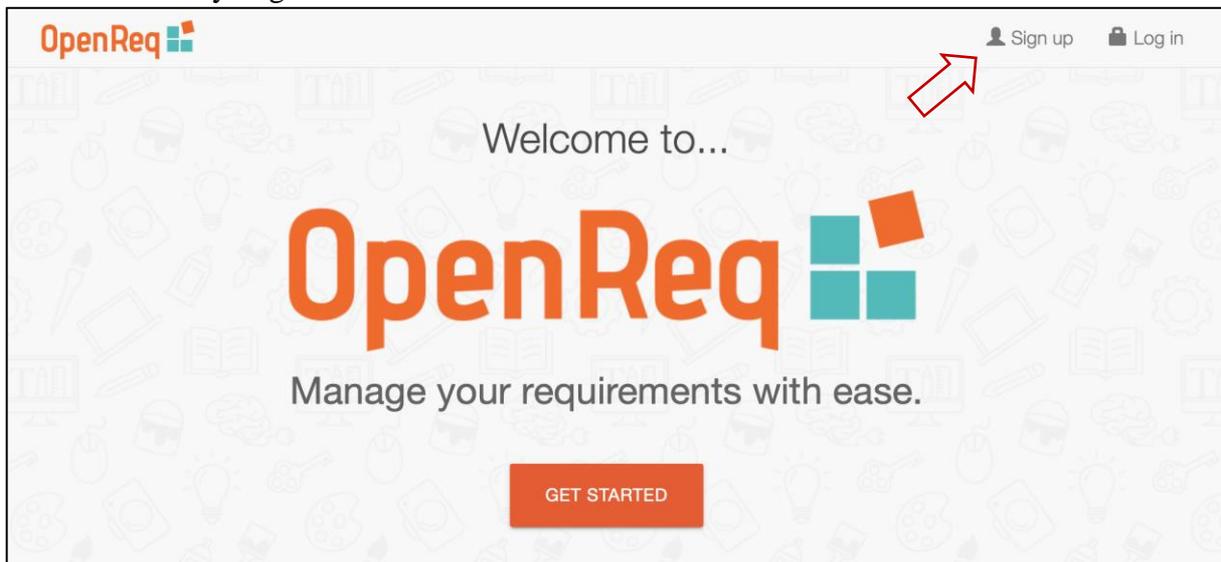


Figure 1 Registration in OpenReq Live

Dashboard

After a user has logged into the system he/she is automatically forwarded to his/her personal dashboard where all projects related to the user are displayed at a glance (projects the user participates in or the user has created him/herself) – see Figure 2.

¹ <https://live.OpenReq.eu>

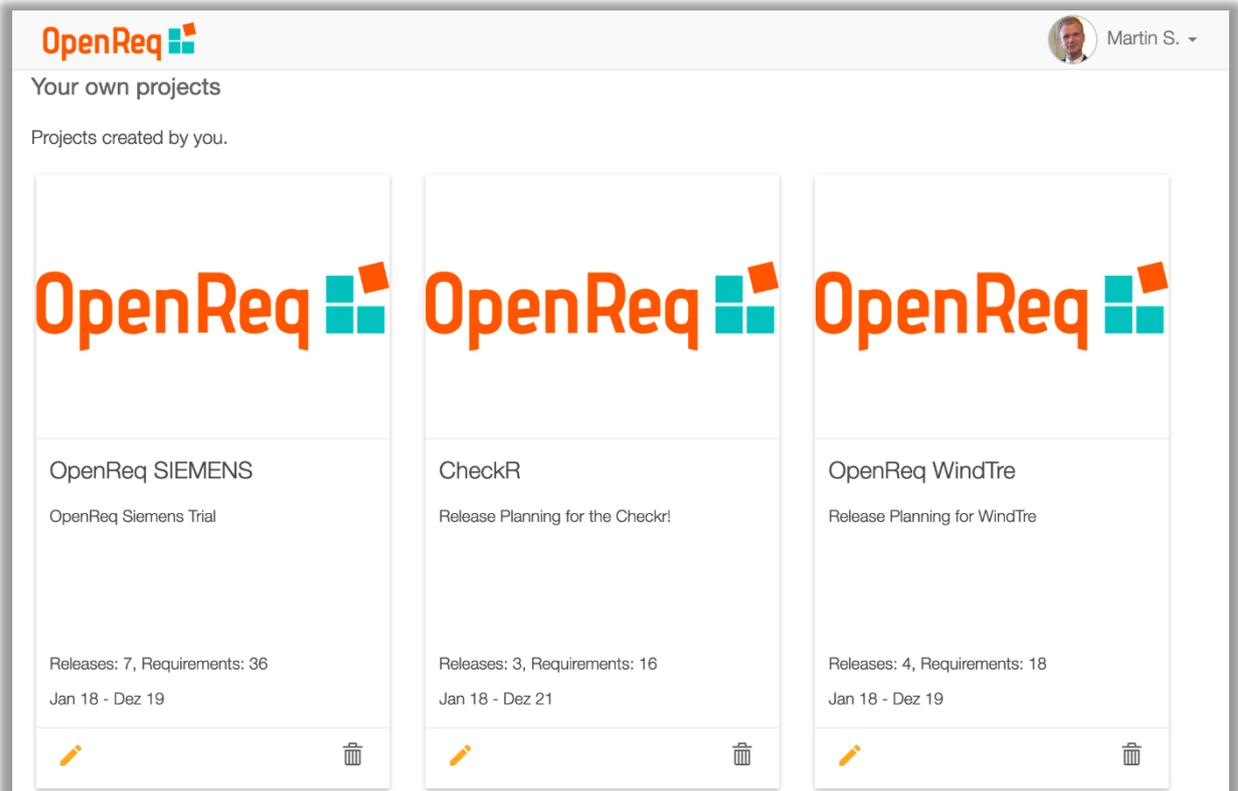


Figure 2 Dashboard of OpenReq Live

Projects Main Page

On a project's main page, the platform presents a compact overview of the project which illustrates the defined project structure at a glance (see Figure 3). The users of the platform can modify and update the requirements and releases directly on this overview page.

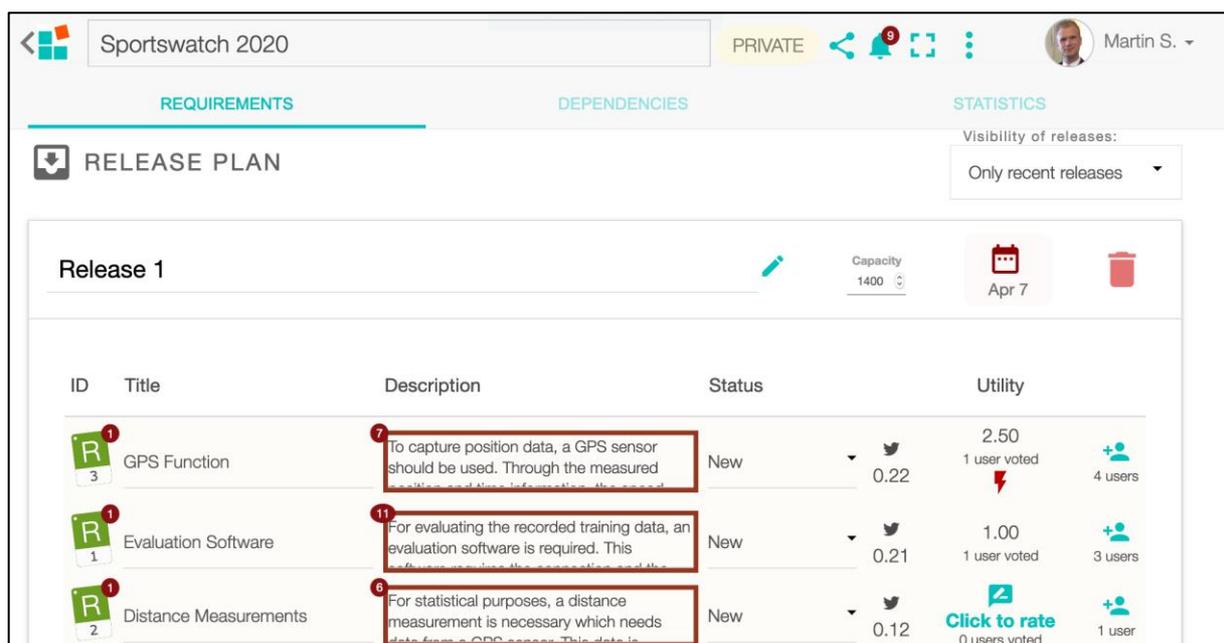


Figure 3 Overview of an example project in Openreq Live

Requirements consisting of a *unique ID*, *title*, *description*, and *status*, are listed on the page and ordered by a utility value. Each release has a deadline and a maximum capacity value (in hours) which limits the possible number of requirements that can be assigned to the release. The OpenReq services have been seamlessly integrated into OpenReq Live's user interface. For example, the numbers labeled red indicate issues (such as requirement duplicates, ambiguities in a requirement's description text, etc.) reported by some of the services.

OpenReq Live Project Settings

Every project can be adjusted according to the individual needs by enabling / disabling OpenReq features individually.



Figure 4 Settings Overview of OpenReq Live projects.

Major settings that can be defined, are the following (see Figure 4):

1. Invite other stakeholders to the project.
2. Individual Notification Center (overview of important messages such as the detection of similar requirements etc.).
3. Enter full screen mode.
4. Advanced Options.

The following screen illustrates the options available in the *advanced options* Section (see 4. in Figure 4).

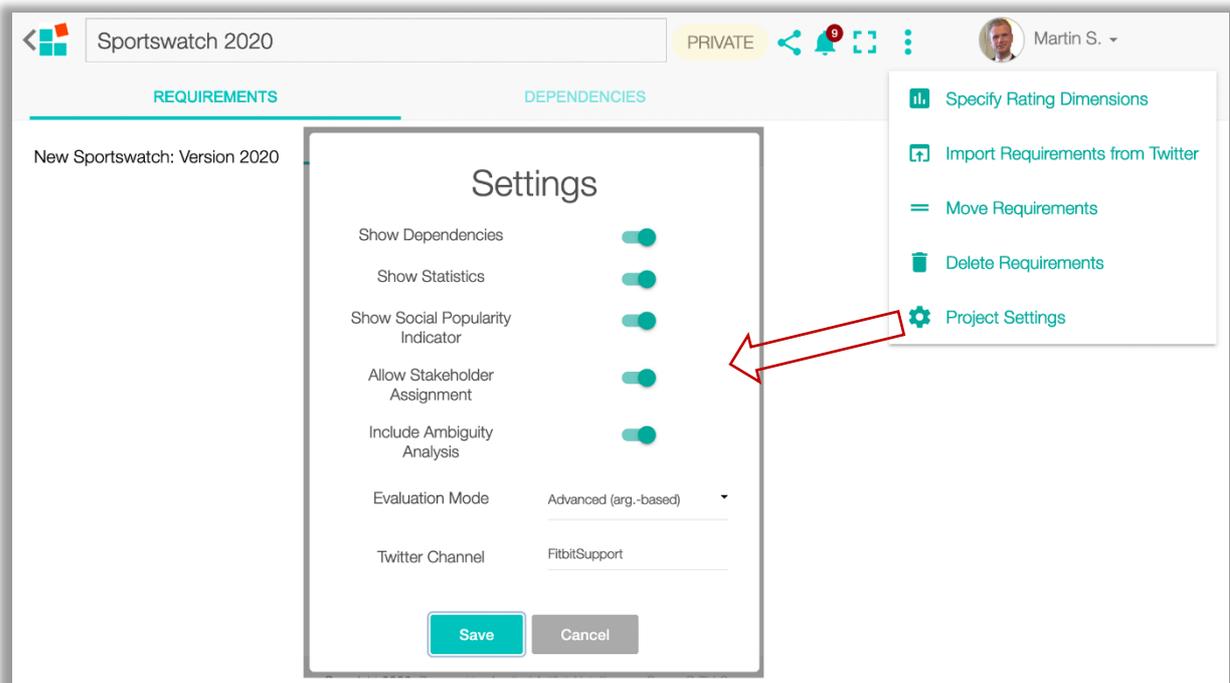


Figure 5 Advanced options in OpenReq Live.

Specify Rating Dimensions

Depending on the project's properties, there is a need for different evaluation criteria regarding requirements. OpenReq Live allows a flexible definition of the rating dimensions of the requirements on a project level (see Figure 6).

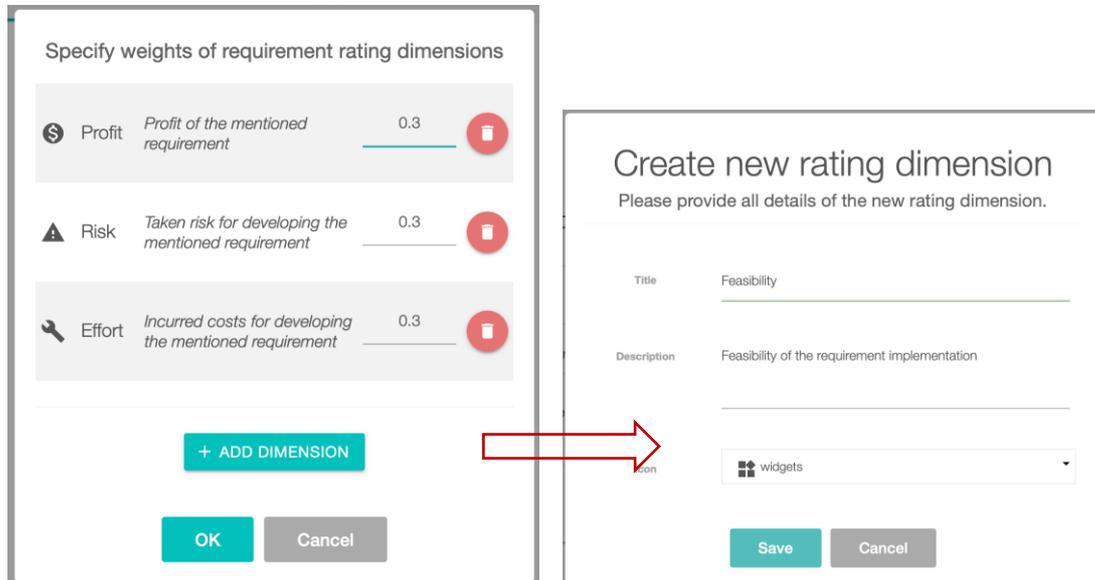


Figure 6 Definition of requirement rating dimensions.

Import Requirements from Twitter

On the basis of the defined Twitter channel in the project settings (see Figure 5), the tweets on this channel are analyzed and classified in three different categories (*Inquiries*, *Problem Reports* and *Irrelevant*). Tweets of all three categories can be directly imported in the projects requirements list.

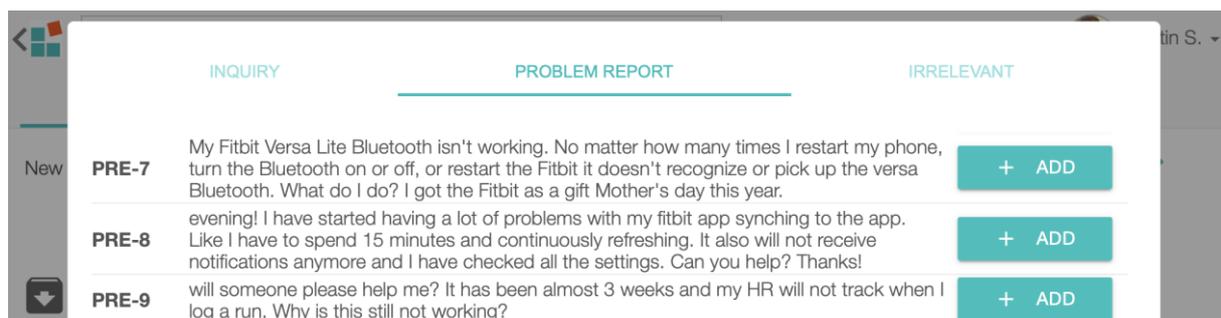


Figure 7 Import Requirements from Twitter directly into OpenReq Live.

Move Requirements

With this option enabled, the requirements can be moved to the defined releases in the release plan via drag and drop.

Delete Requirements

Deletion of project requirements.

Project Settings

To tune OpenReq Live to specific project demands, several OpenReq features and / or services can be selected in the project's settings. Additionally, the *evaluation methodology* can be a *five-star rating scale*, a *utility-based evaluation*, or an advanced evaluation mode based on *argumentations pro and contra a requirement*. Finally, the relevant Twitter channels for the current project can be defined.

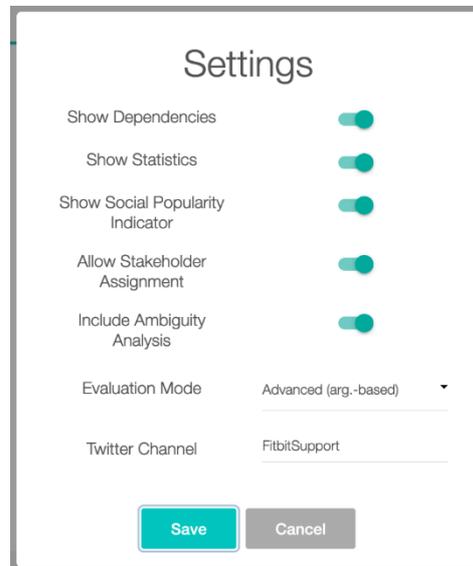


Figure 8 Advanced project settings.

Prioritization of Requirements

The evaluation of the requirements defines the basis for the utility-based prioritization. Figure 9 shows an example of an argumentation-based MAUT rating interface. Stakeholders are asked to provide textual feedback (in terms of arguments) for every requirement. The stakeholders argue on issues related to the requirement and manually classify the sentiment of their arguments (*positive (PRO)*, *neutral (NEUTRAL)*, or *negative (CON)*). Moreover, based on the type of issue, every argument has to be assigned to at least one interest dimension *profit*, *effort*, *risk*. Strongly diverging arguments assigned to the same interest dimension indicate strong disagreement and appear as a conflict in the interface. The involved stakeholders have to discuss the issue and resolve the conflict by reevaluating the requirement (i.e., providing improved arguments based on the outcome of the discussion). This leads to more information exchange between stakeholders and has a positive impact on the quality of the prioritization process. After the elimination of all conflicts, the system computes a utility/priority value for each requirement using MAUT and ranks the requirements based on their priority.

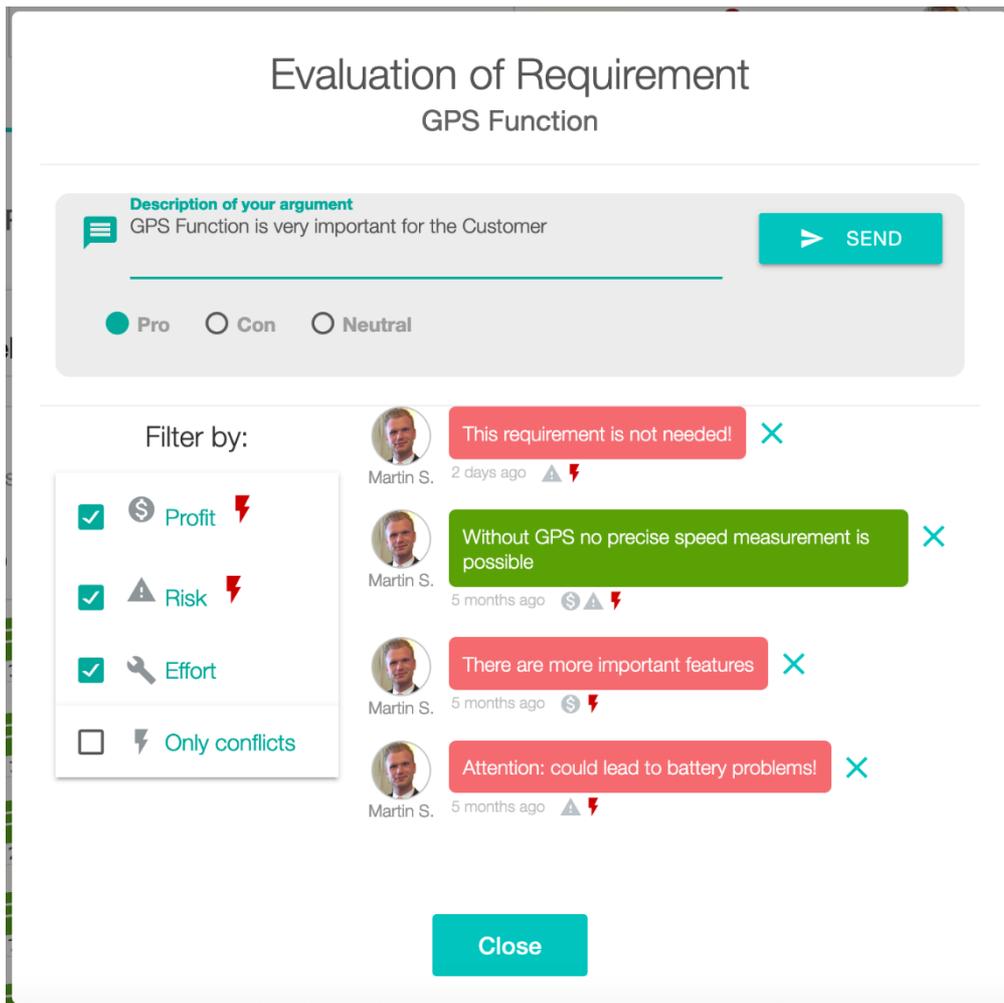


Figure 9 Argumentation-based rating interface

Argumentation-based interfaces allow stakeholders to exchange arguments for/against a requirement. Each argument must be assigned to at least one interest dimension. Negative arguments are highlighted in red, positive ones in green, and neutral arguments in orange.

Stakeholder Recommendation

After clicking on the user icon of a requirement, stakeholders can be assigned to a requirement. Automatic recommendations for appropriate stakeholders are marked with an “AI” tag whereas manual assigned stakeholders are marked with a “ST” tag (see Figure 10). In the case that a stakeholder accepts the assignment he / she will be marked in green.

Propose assignee:

Enter name...

PROPOSE

Proposed assignees:

	✂️ Appropriateness		🕒 Availability		Result	
	Your	Average	Your	Average		
Martin Stettinger (you) AI Accepted	9	3.6	4	7.0	5.3	
Muesluem Atas AI	10	3.3	1	7.8	5.5	
AF ST	2	2.0	9	9.0	5.5	
Lord Root	-	2.8	-	8.3	5.5	

Close

Figure 10 Stakeholder Assignment

Dependency Management in OpenReq Live

Dependencies can be added in different ways in OpenReq Live. First, dependencies can simply be *added manually* by defining the left and the right side of the dependency as well as the type of the dependency (*Implies*, *Requires*, *Excludes*, *Incompatible*) – see Figure 11.

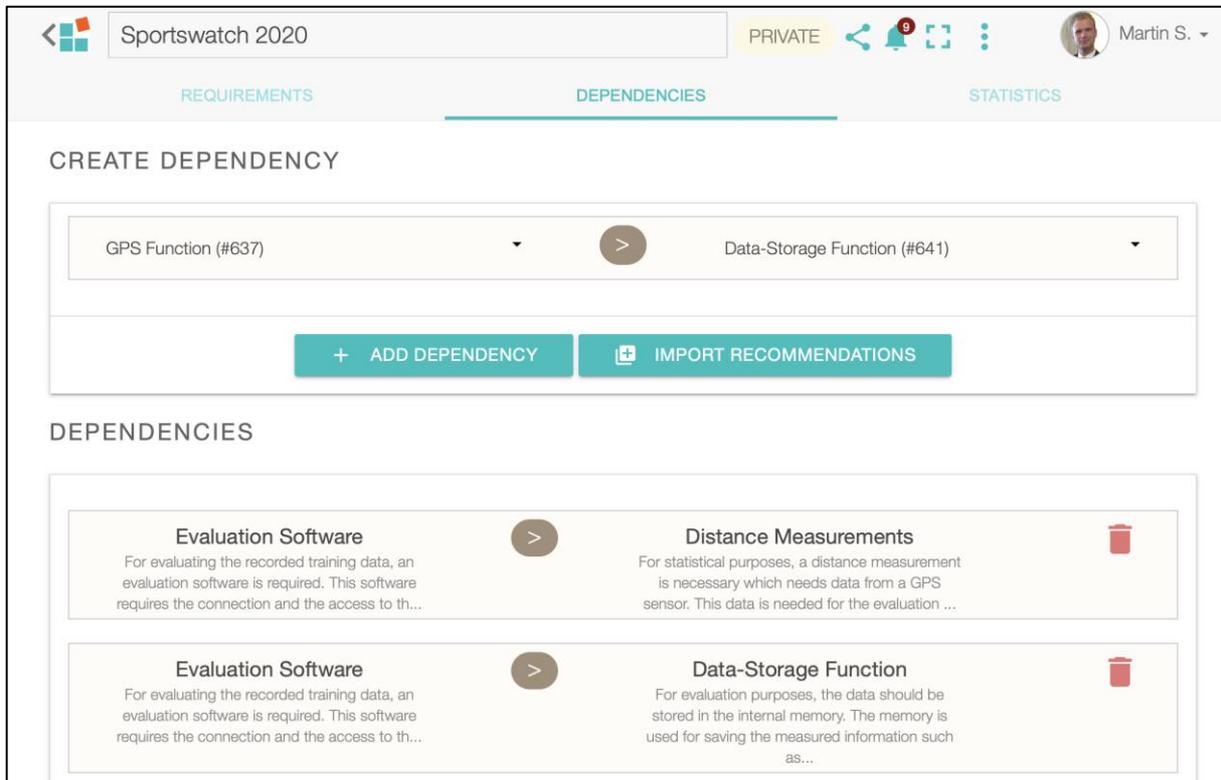


Figure 11 Dependency management in OpenReq Live.

The second option for the definition of dependencies is the import of *recommended dependencies* powered by OpenReq services (see Figure 12).

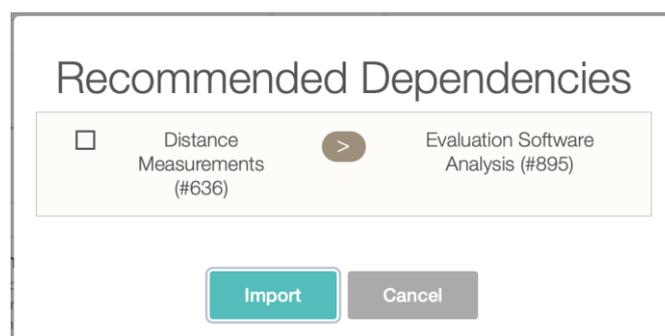


Figure 12 Import of recommended dependencies.

In case of inconsistent dependency definitions, the system informs the user about that state both in the notification center as well as directly in the *dependency* section (see Figure 13).

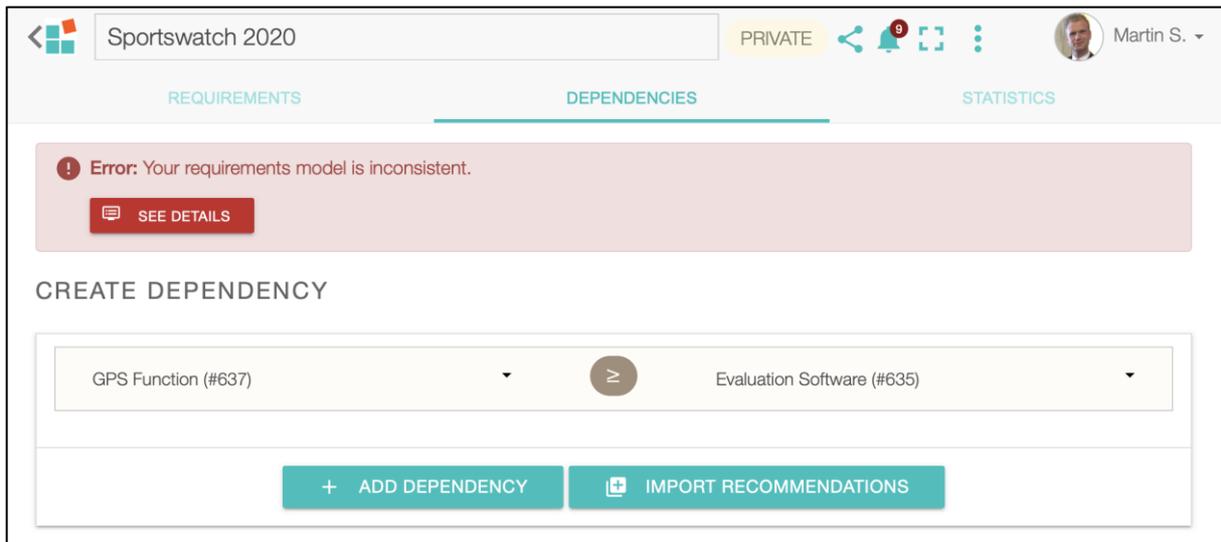


Figure 13 Indication of inconsistencies in OpenReq Live.

OpenReq Live also presents details and root causes of an inconsistency. Additionally, OpenReq Live shows two possible ways of restoring consistency (adaptation of selected requirements and / or adaptation of selected dependencies) – see Figure 14.

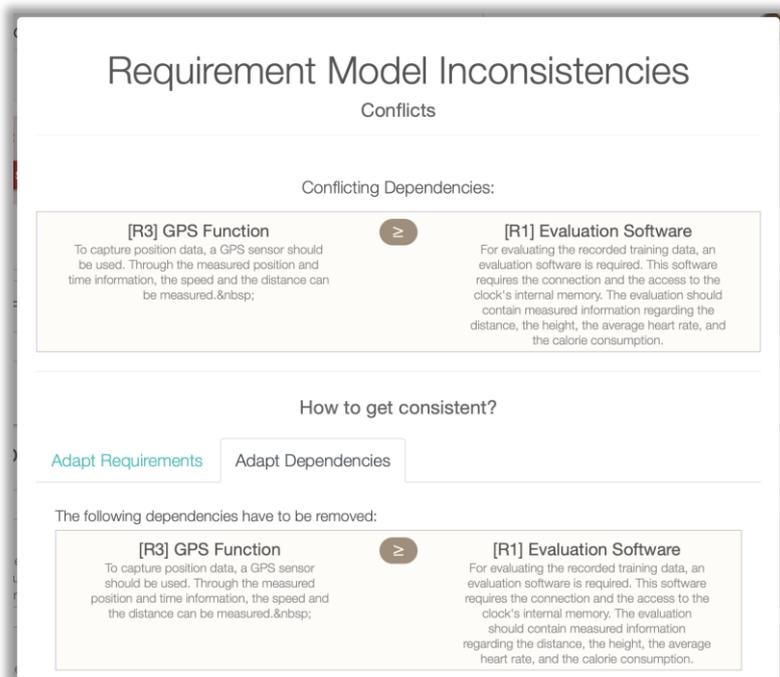


Figure 14 Details about inconsistencies in the current requirements model.

Statistics

In the project's statistics section, two different statistics can be found. First, the user sees details regarding the changes on requirements, for example, how often has a requirements title, description, or state been changed during the release planning process (see Figure 15).

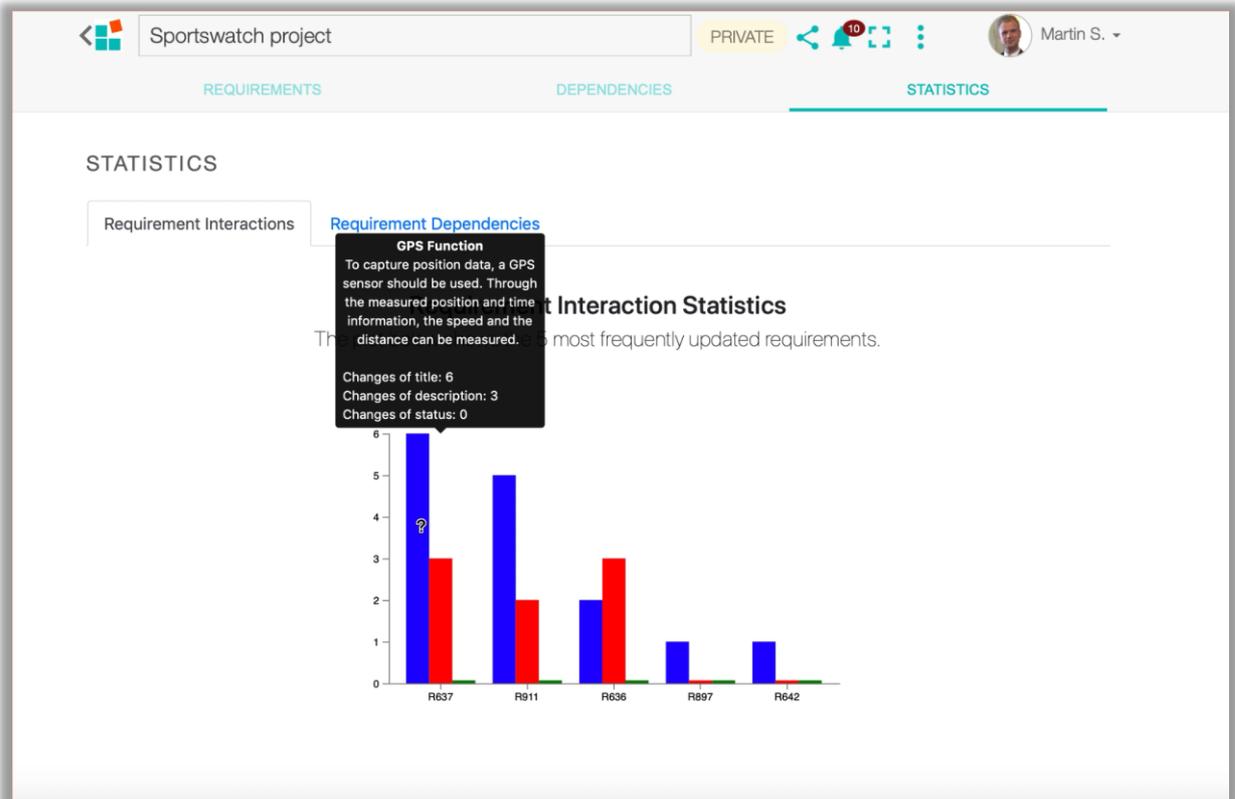


Figure 15 Statistics about changes of the requirement's title, description and state.

Second, the user sees a visualization of all dependencies ("dependency wheel") which shows the relation between requirements in an intuitive fashion (moving the cursor on a requirement highlights the connections to all related requirements on the basis of the available dependencies) – see Figure 16.

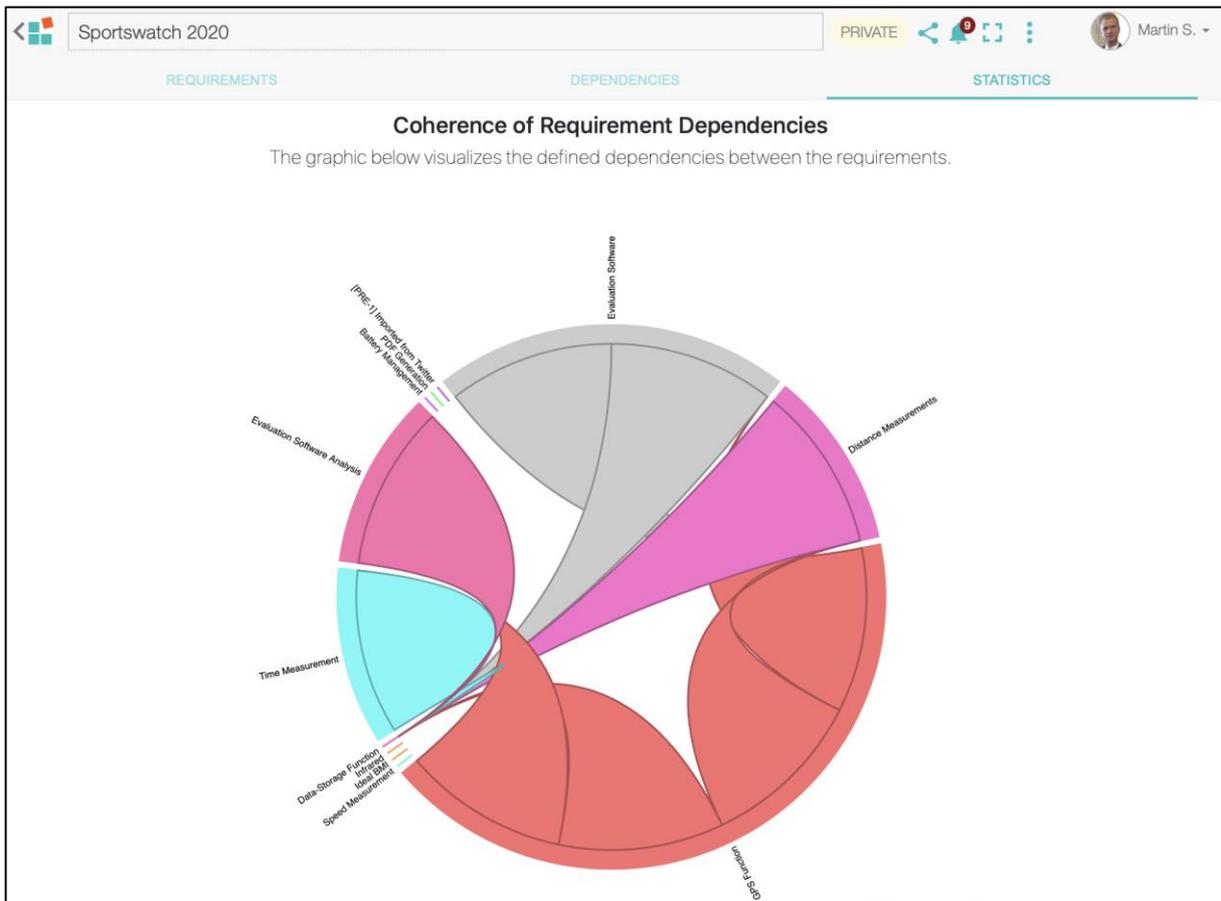


Figure 16 Indication of the dependencies in OpenReq Live.

Further information regarding the OpenReq project can be found here: www.openreq.eu.