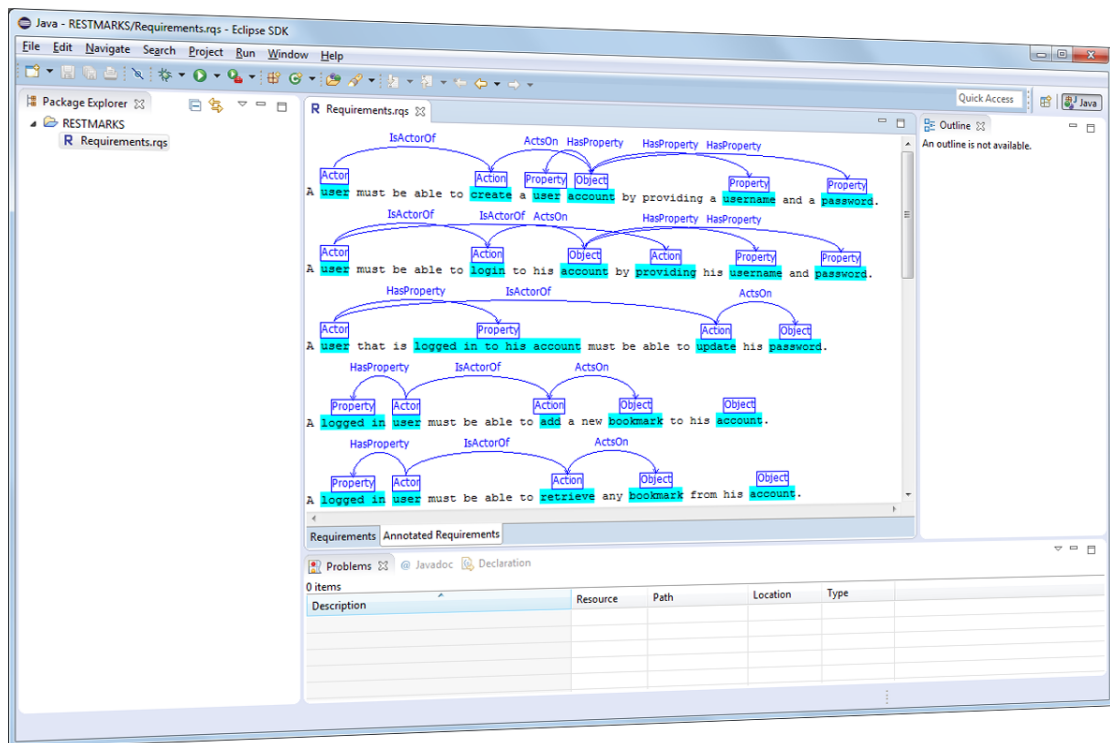


Requirements Editor[©]

User Manual of the Requirements Editor[©] Eclipse Plugin



Contents

1. Introduction	3
2. Installing the Requirements Editor [®]	4
3. Updating the Requirements Editor [®]	7
4. Using the Requirements Editor [®]	8
4.1. Overview	8
4.2. Creating Requirement Annotations	9
4.2.1. Creating Projects and Requirement Annotations Files.....	9
4.2.2. Adding Requirements	10
4.2.3. Viewing and Modifying Annotations	11
4.2.4. Automatic Annotation	13
4.3. Importing/Exporting Requirements Annotations Files.....	14
4.3.1. Importing Requirements Annotations Files.....	14
4.3.2. Exporting Requirements Annotations Files	15
References	17

1. Introduction

Requirements Editor[®] is an Eclipse plugin designed and developed by the Aristotle University of Thessaloniki as part of the S-CASE project [1]. The aim of this tool is to help developers express the static view of software projects using functional requirements.

The diagram tool is built using the Plug-in Development Environment (PDE) of Eclipse [2]. The tool's update site is:

http://s-case.github.io/s-case_update_site/

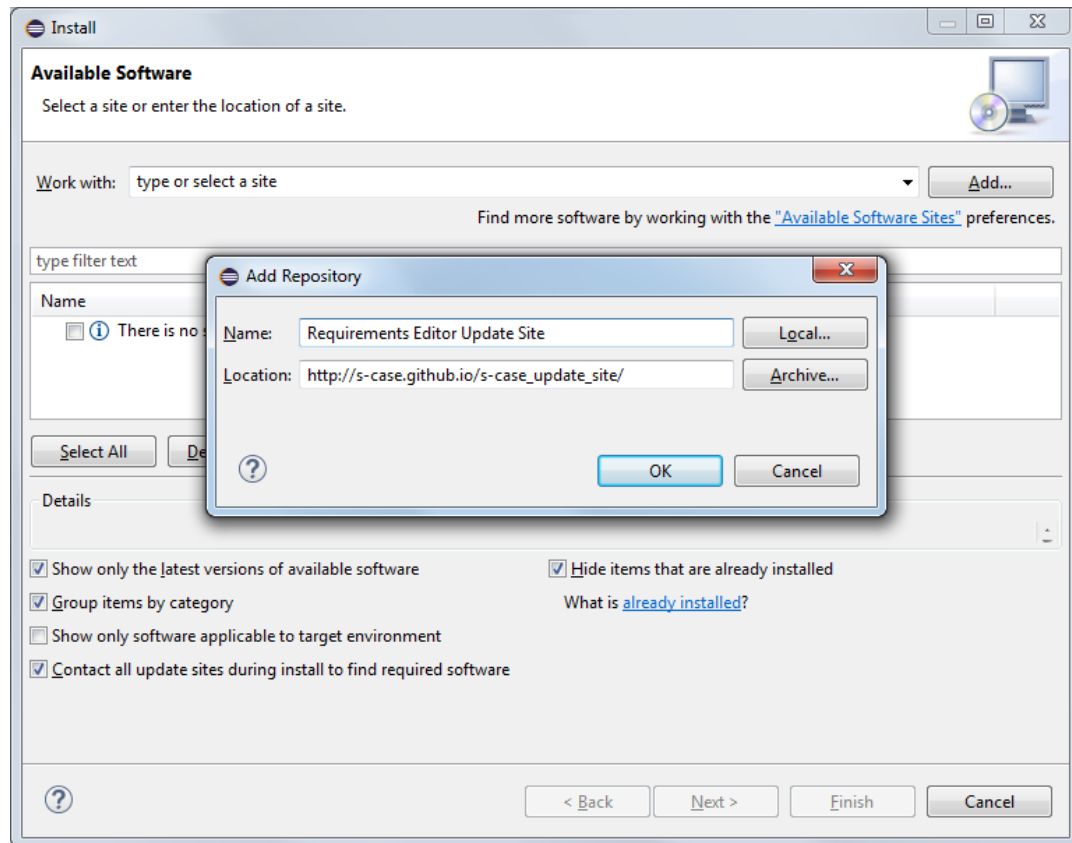
The tool is licensed under version 1.0 of the Eclipse Public License (EPL) [3].

2. Installing the Requirements Editor[®]

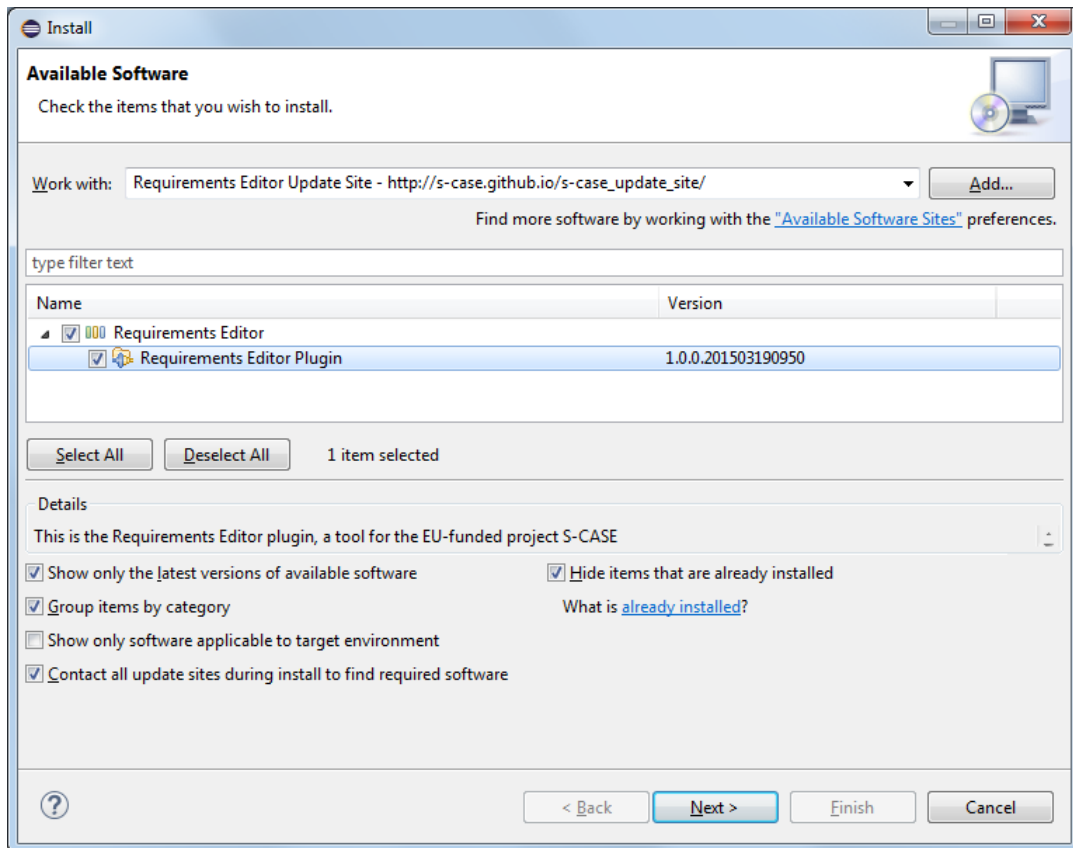
The installation procedure of the Requirements Editor[®] is simple. The easiest way to install the editor is by using the Eclipse plugin update site. In Eclipse, you can select the *Help* menu and click on the option *Install new software*. After that, click on the *Add...* button on the right to add a new repository. In the dialog box that opens insert the address of the Requirements Editor[®] update site

`http://s-case.github.io/s-case_update_site/`

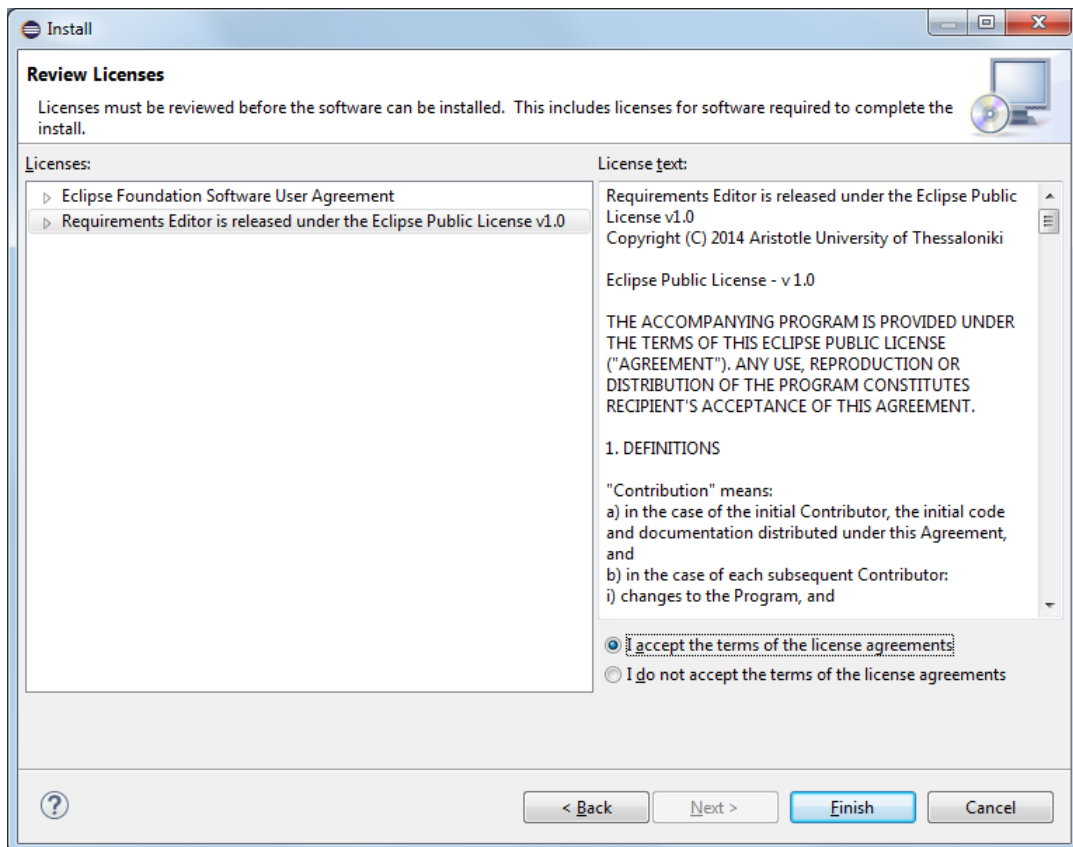
in the location field and fill in a name for the new repository.



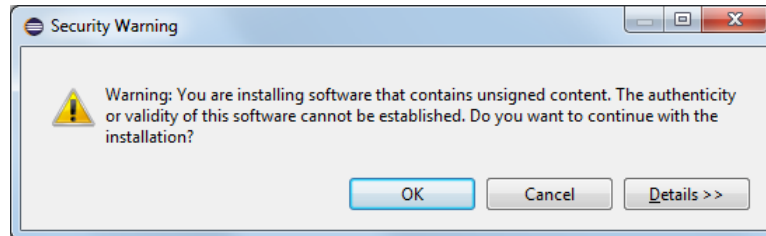
After that the new repository should be selected in the *Work with* field. Tick the *Requirements Editor Plugin* and press *Next* in order to start the installation.



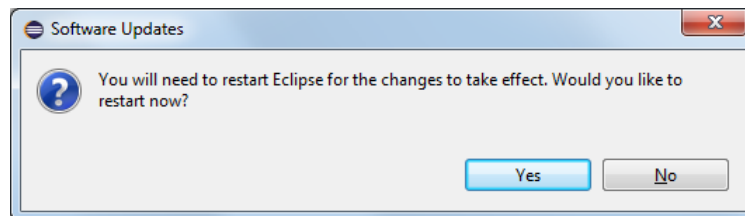
Upon pressing *Next* once again, the install shield shows the product's license.



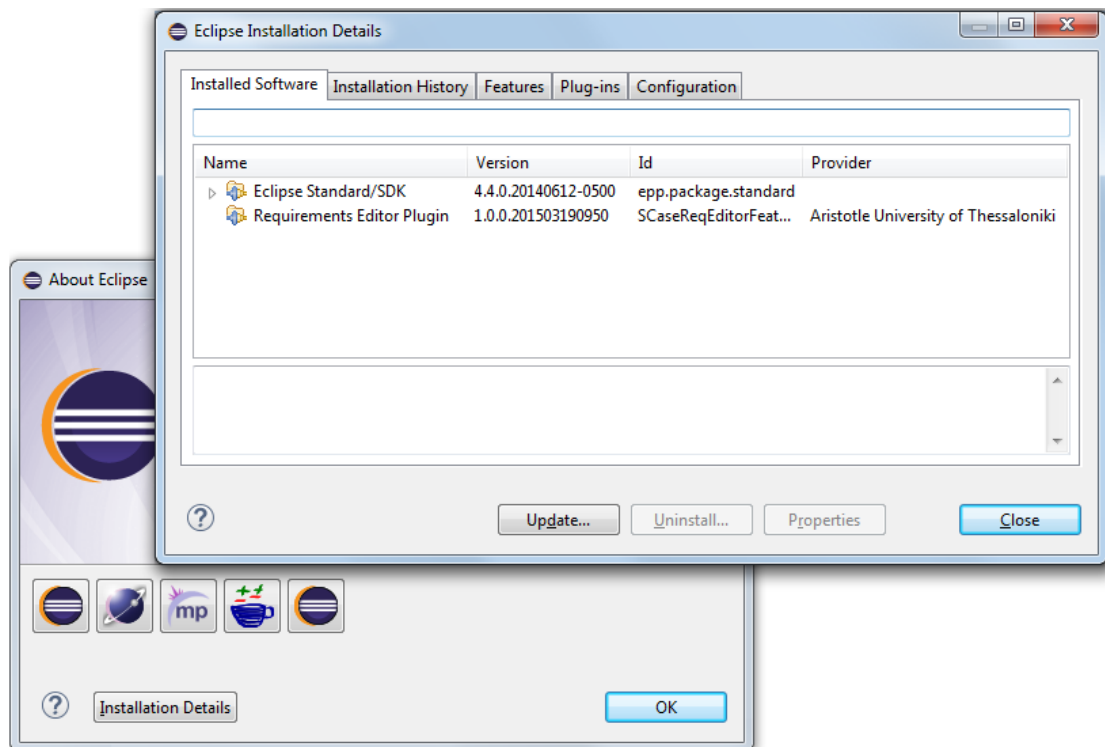
If you agree with the terms of the license agreements, select the radio button / *accept the terms of the license agreements* and press *Finish*. The Requirements Editor[®] shall start installing. Note that Eclipse may issue a security warning since the plugin is not signed with a certificate. You can safely ignore this warning by pressing *OK*.



Finally Eclipse will continue with the installation. Upon completion, you have to press *Yes* in the prompt box that asks you to restart Eclipse.

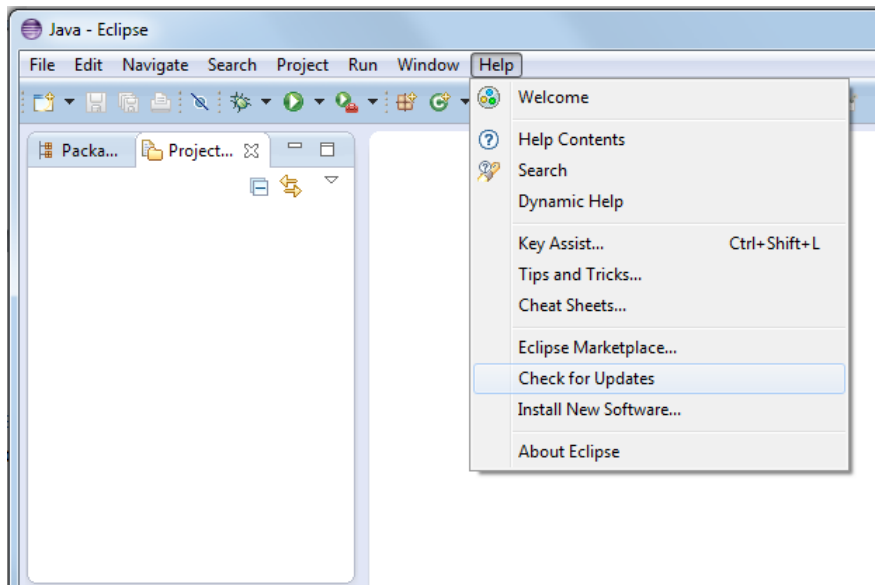


You can check that Requirements Editor[®] is installed by selecting the option *About Eclipse* of the *Help* menu and pressing *Installation Details*.

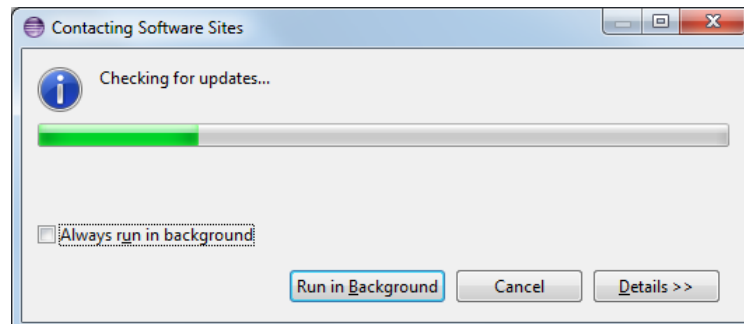


3. Updating the Requirements Editor[®]

Updating the tool is quite easy. The update is initialized by selecting *Help* and *Check for Updates*.



Since the tool is installed, Eclipse will contact the software site and will prompt for updating it if there is some newer version.



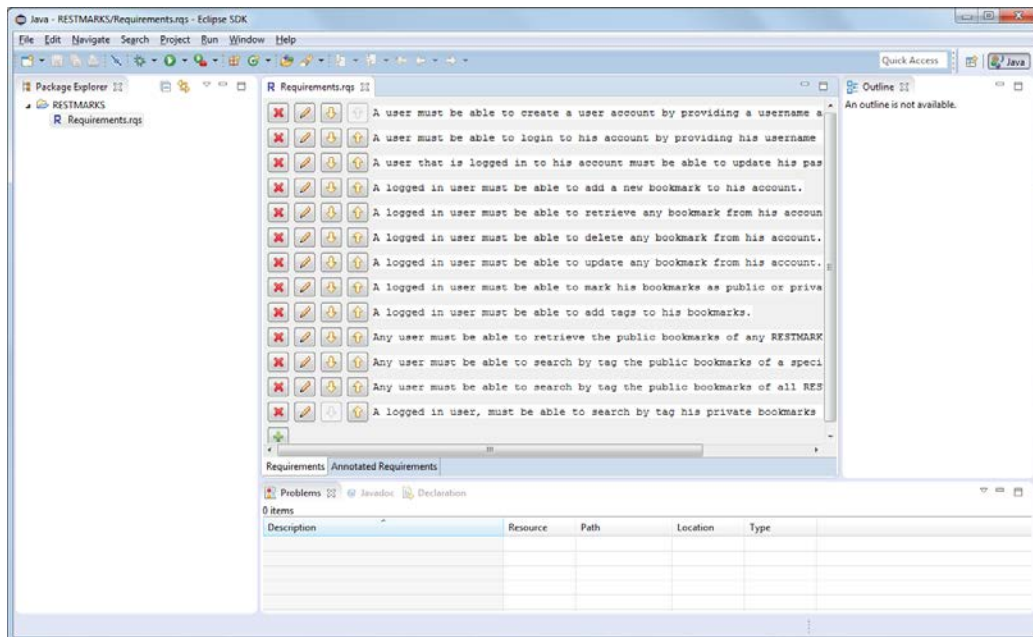
Upon prompting the user, Eclipse will start the update procedure of the plugin which is similar to the Install procedure of the previous Section.

4. Using the Requirements Editor[®]

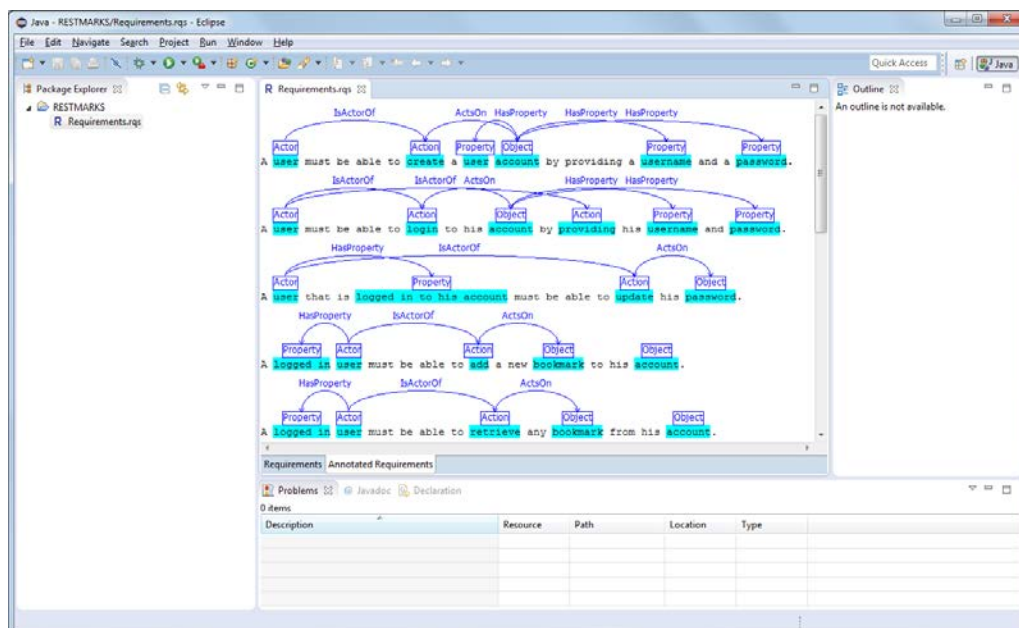
This Section covers the usage of Requirements Editor[®]. Initially, an overview of the main screens of the tool is shown and after that the procedure of adding and annotating requirements and importing/exporting them is demonstrated.

4.1. Overview

Requirements Editor[®] is a multi-page editor of Eclipse. The first page of the editor is shown in the following screenshot.



The first page is an editor containing user requirements. The second page, which is a UI annotations editor, is shown in the following screenshot.

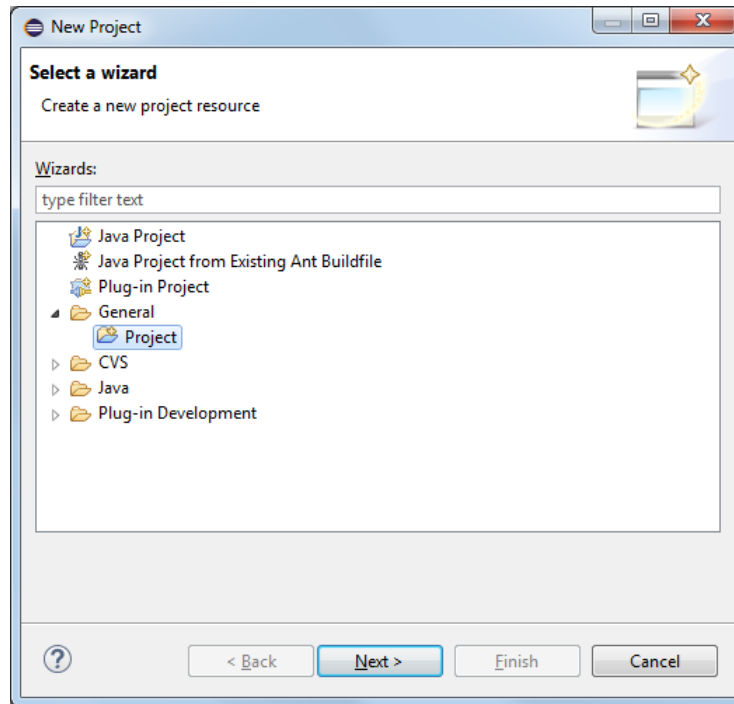


Note that the user is able to change the size and the position of the editor as normally allowed by the Eclipse IDE.

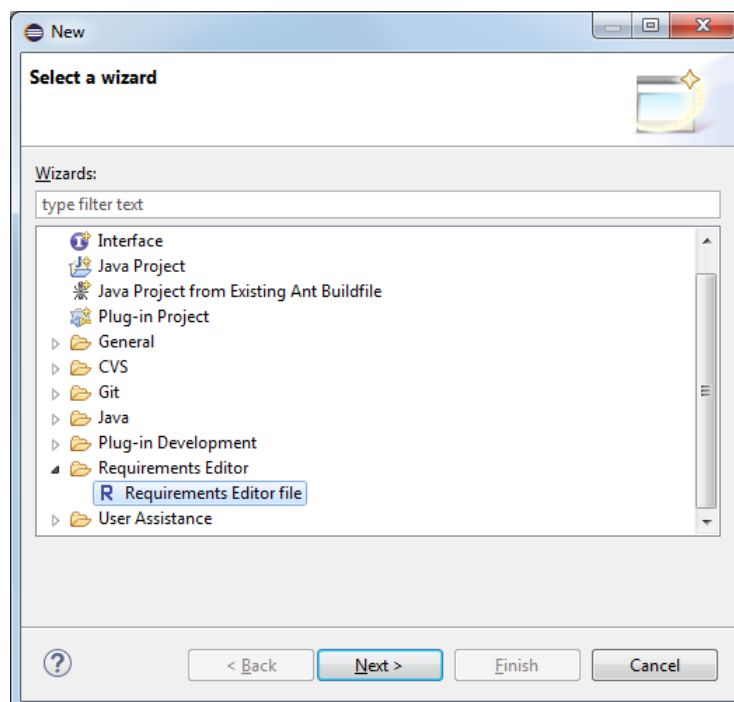
4.2. Creating Requirement Annotations

4.2.1. Creating Projects and Requirement Annotations Files

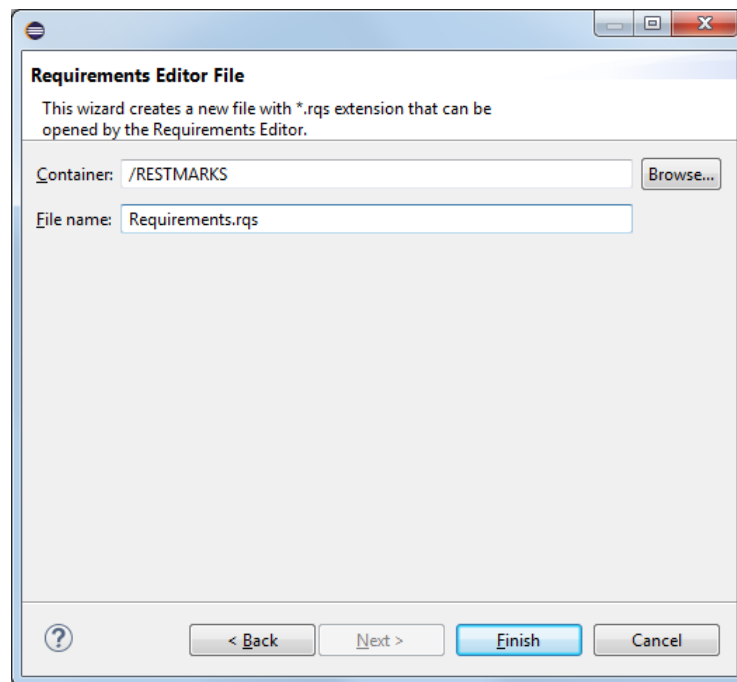
Requirement Annotations stored in Eclipse projects. So at first, one has to create a project by selecting the option *File* and then *New* and *Project...*. After that, the general type of project has to be selected.



After that, the user is asked to give a name for the project and press *Finish*. The process for creating a requirements file is similar. Upon selecting the option *File* and then *New* and *Project...* the user is presented with the following screen.



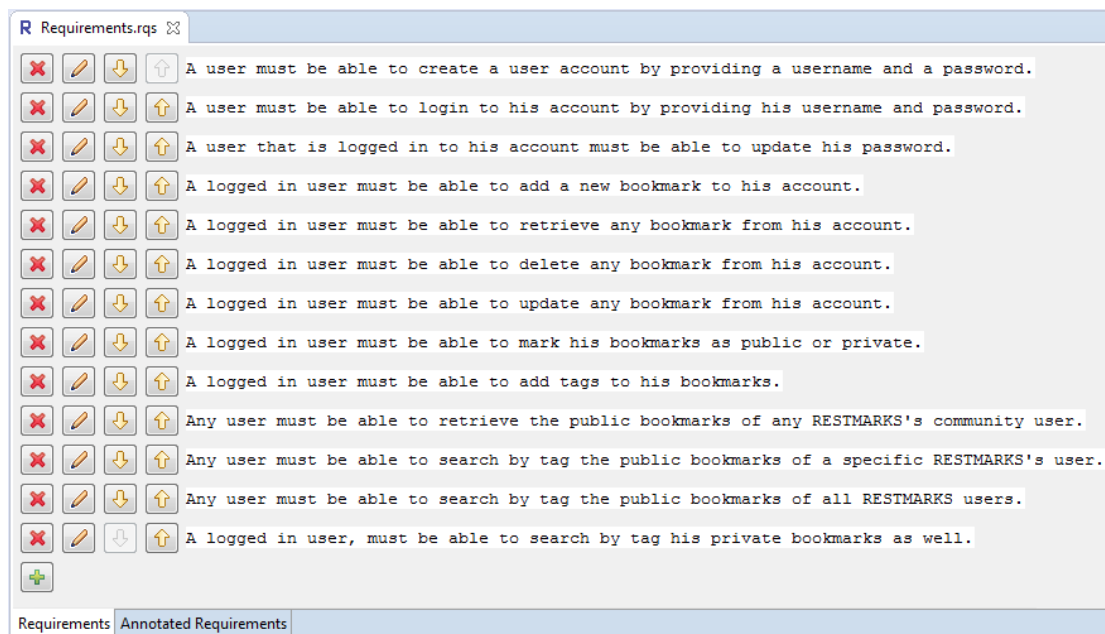
After clicking on *Next*, the next screen asks for a name for the newly created Requirements file.





Note that the name must have the extension rq5.



4.2.2. Adding and Modifying Requirements

The user may add the functional requirements in text form in the first page of the Requirements Editor[©], as shown in the following screenshot.



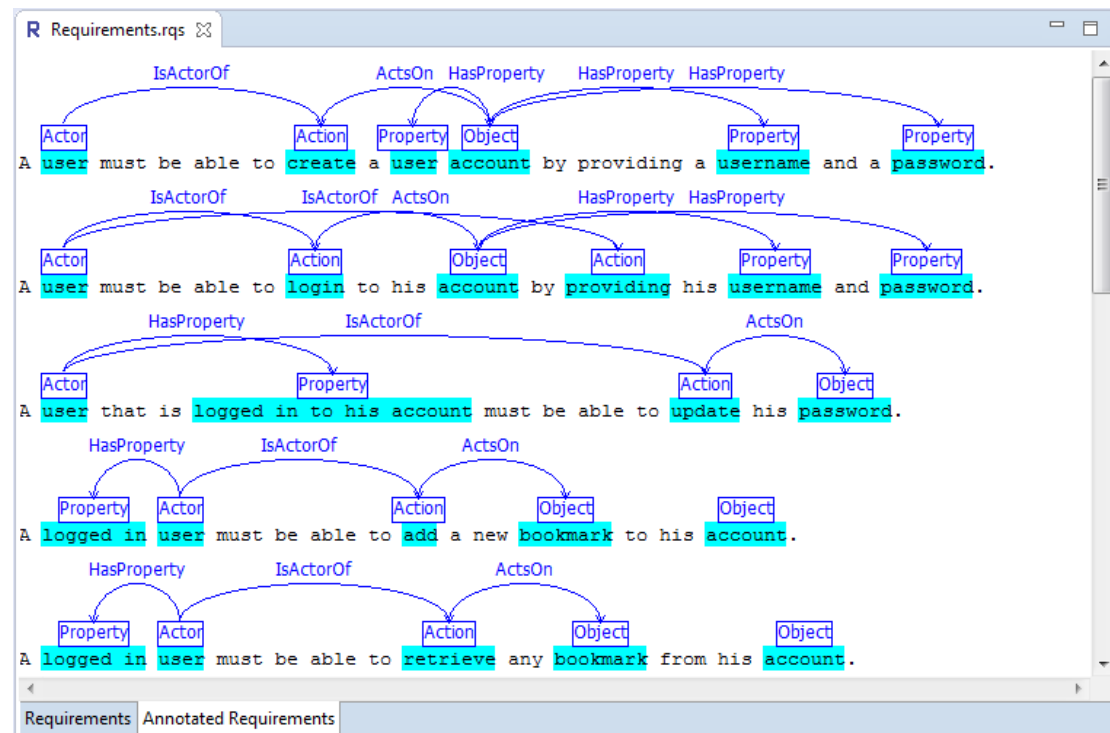
The user can add a new requirement by pressing the  button.

Requirements can also be deleted, using the  button, or modified, using the  button. Note that if a requirement is deleted or modified, the corresponding annotations of the requirement are also deleted.

Finally, requirements can be moved down or up using the  and  buttons.

4.2.3. Viewing and Modifying Annotations

Annotations can be viewed and modified in the second page of the Requirements Editor[®]. An example showing an annotated requirements file is shown in the following screenshot.



As shown in this screenshot, there are two types of annotations, *entities* and *associations*. Entities declare the type of each word (or phrase), and they can be one of:

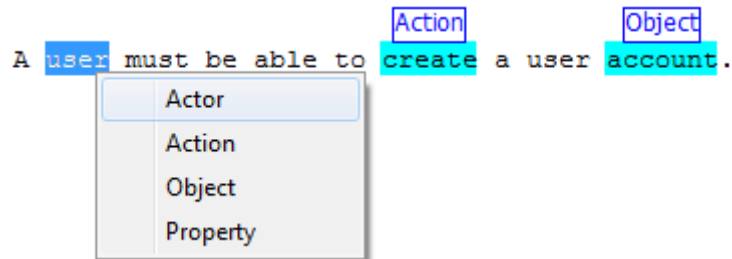
- Actor
- Action
- Object
- Property

Associations declare the relations between different entities. The following relations are supported:

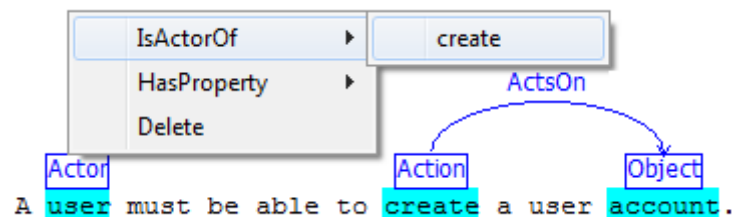
- IsActorOf, which is defined from an Actor to an Action;
- ActsOn, which is defined from an Action to an Object or from an Action to a Property;
- HasProperty, which is defined from an Actor to a Property or from an Action to a Property or from an Object to a Property.

Annotations are added and deleted using the right click context menu, while any modification of annotations is automatically saved to the file. Undo/redo operations are not supported.

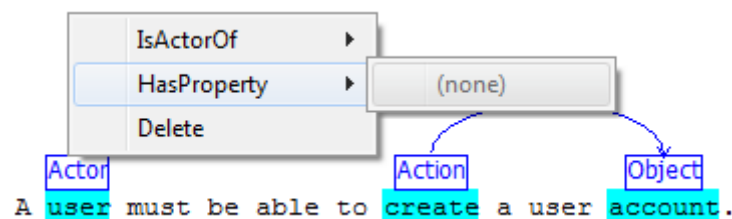
An entity annotation can be created by selecting a word or phrase and right clicking on the selected word. After that the user can select the appropriate entity from the context menu.



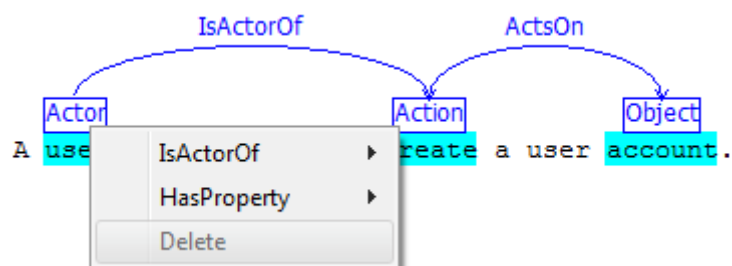
Associations are created by right clicking on existing entity annotations. The Requirements Editor[®] automatically determines all possible associations from the currently selected entity and groups them in categories according to their type.



When no association is possible from a selected entity or any possible target entities are already selected, the tool does not allow selecting the association.

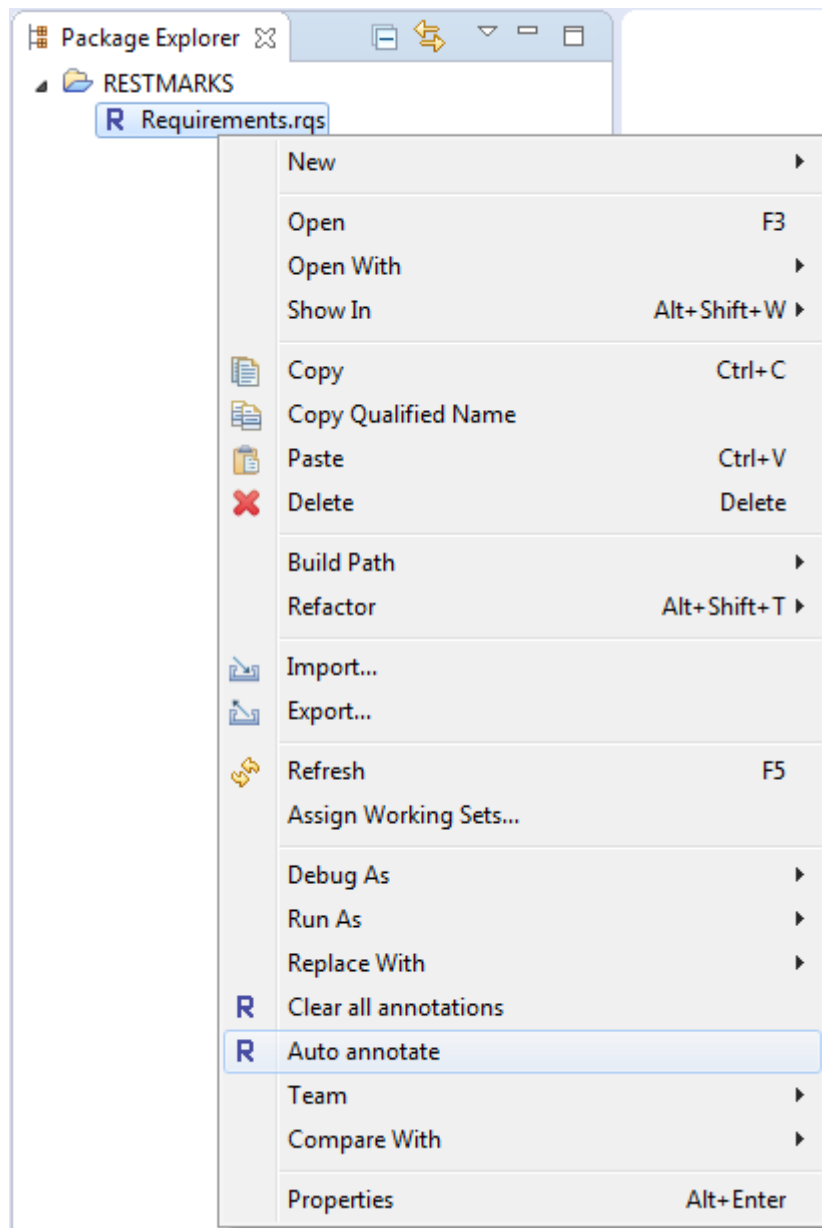


Finally, deleting an annotations is simple. One can delete an annotation by right clicking on it and selecting the option *Delete*. Note, however that entity annotations cannot be deleted if the related associations are not deleted first.



4.2.4. Automatic Annotation

The Requirements Editor[®] also supports automatically annotating software requirements using an NLP parser. This is performed by right clicking on the rqs file and selecting the option *Auto annotate*.



Note that when selecting this option all annotations are removed. Additionally, as shown in the above screenshot, the user can select to clear the annotations of a file using the menu option *Clear all annotations*. This command is particularly useful if the user wants to modify the requirements in the first page of the editor.

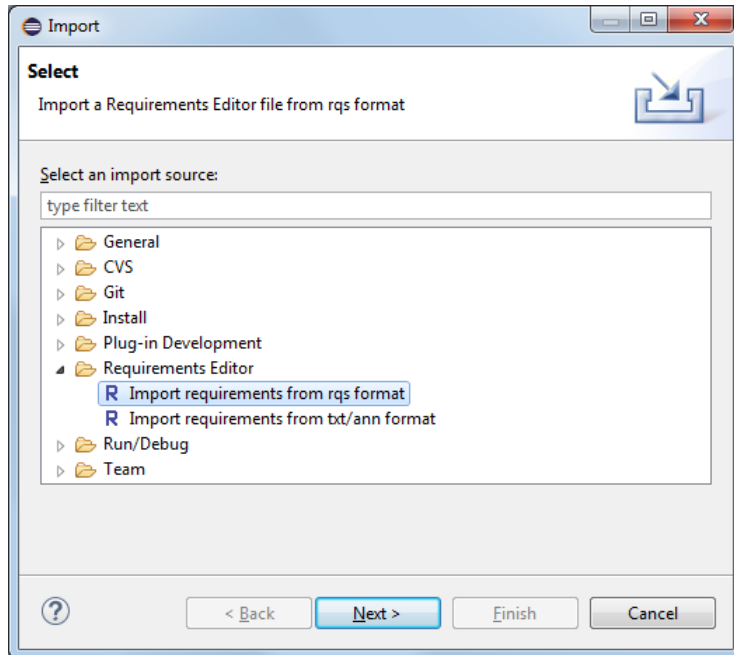
Finally, note that automatic annotation should be used as a helpful complement; however it is important that the user inspects the annotations and modifies them if required.

4.3. Importing/Exporting Requirements Annotations Files

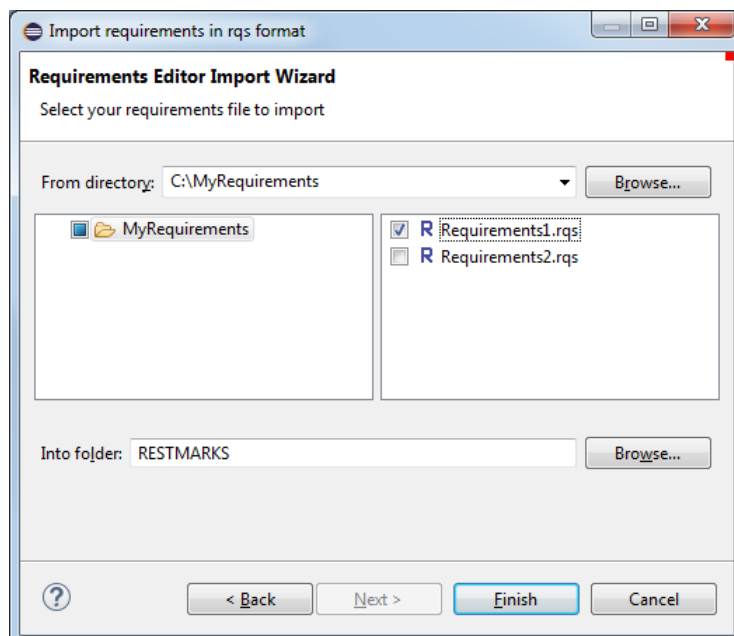
The Requirements Editor[®] can import and export the requirements and the annotations of a project using the functionality of the Eclipse IDE.

4.3.1. Importing Requirements Annotations Files

Requirements Annotations Files are imported into projects from two formats, the *rqs* format and the *txt/ann* format. After right clicking on a project and selecting the *Import...* option in the context menu, the following screen is shown.



Then, the user can select either to import a final rqs file or to import a text file which includes the requirements and (optionally) an ann file which includes the respective annotations. Selecting the *Import requirements from rqs format* and clicking *Next* shows the following screen.

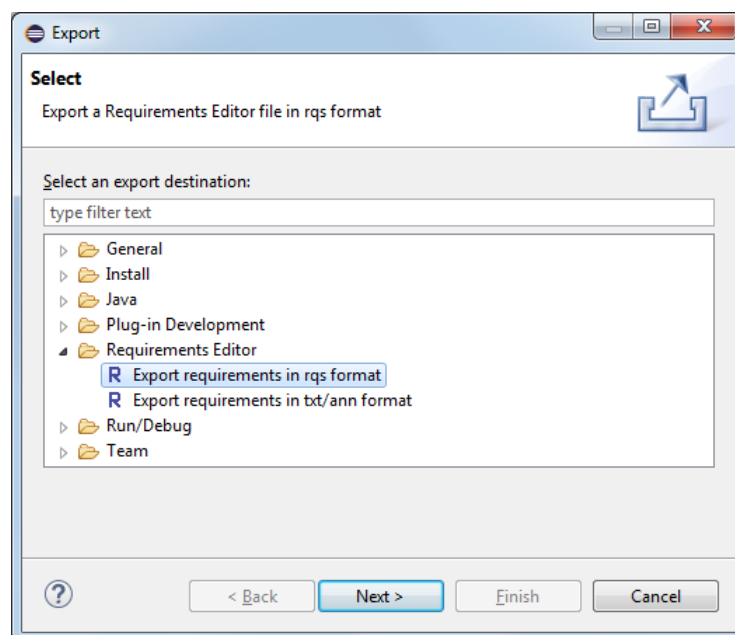


As shown in the above Figure, the user has to select the directory where the requirements files to be imported reside. After that, the user is presented with a list of requirements files on the right of the screen. Note that only files with extension rqs are accepted. In the bottom of the window, the user may select the project that will have the newly imported requirements files.

The procedure for importing txt and ann files is similar. Note however that importing an ann file without importing the respective txt file is not allowed (whereas the opposite is allowed).

4.3.2. Exporting Requirements Annotations Files

Exporting requirements and annotations is also a quite simple procedure. As in the import operations, one can export them in rqs or in txt/ann format by initially selecting the option *Export...* from the right click context menu of a project.

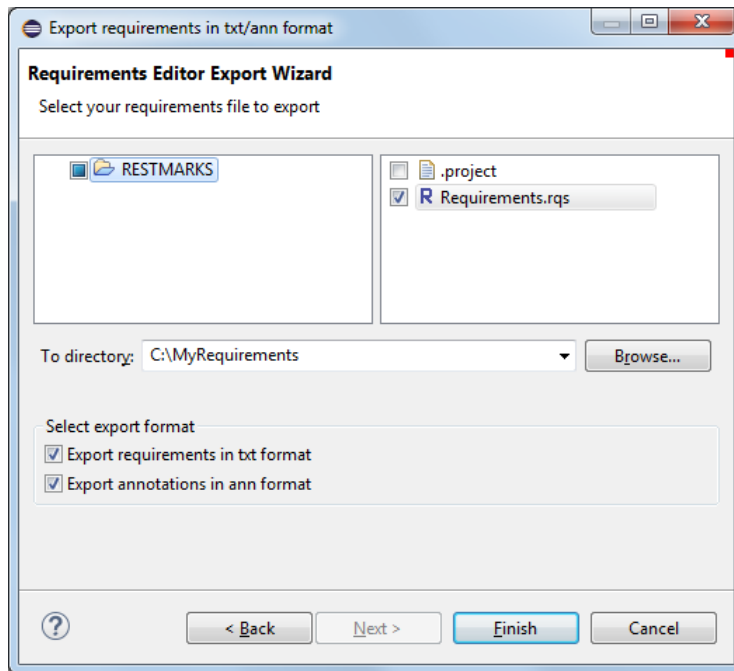


After that, the user selects the appropriate export wizard with regard to the format of the file that the file will be exported.

Similarly to the import option, the *Export requirements in rqs format* option is simple. The user selects the file to be exported and the directory to export the file to.

The *Export requirements in txt/ann format* option is also quite simple, however in this case the user has to select the export format. This is accomplished by ticking the options buttons after selecting the file to be exported and the export directory. Note that in this case it is possible to export either of the two formats without necessarily exporting the other one.

A screenshot of the txt/ann export wizard is shown in the following screen.



Note that trying to import or export any file with no rqs or txt/ann extension will cause Requirements Editor[®] to display error messages.

References

- [1] S-CASE project, FP7, project webpage: <http://www.scasefp7.eu/>
- [2] Eclipse Plug-in Development Environment (PDE), available online: <https://eclipse.org/pde/>
- [3] Eclipse Public License (EPL), version 1.0, available online: <https://www.eclipse.org/legal/epl-v10.html>