



D6.3-E Executable Process framework for certification

Author(s): **Naia Arana** **Tecnalia-ESI**
 Huascar Espinoza **Tecnalia-ESI**
 Adrián Noguero **Tecnalia-ESI**

Issue Date	31 January 2011 (m24)
Deliverable Number	D6.3-E
WP Number	WP6: Compositional Verification, Validation and
Status	Delivered

Dissemination level	
X	PU = Public
	PP = Restricted to other programme participants (including the JU)
	RE = Restricted to a group specified by the consortium (including the JU)
	CO = Confidential, only for members of the consortium (including the JU)

Document history			
V	Date	Author	Description
<i>0.1</i>	<i>2011-01-10</i>	<i>NAO, HEO</i>	<i>ToC and first content</i>
<i>0.2</i>	<i>2011-01-26</i>	<i>NAO</i>	<i>Content</i>
<i>0.3</i>	<i>2011-01-31</i>	<i>ANM- HEO</i>	<i>Review</i>
<i>1.0</i>	<i>2011-01-31</i>	<i>NAO</i>	<i>Final version</i>

Disclaimer

The information in this document is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

The document reflects only the author's views and the Community is not liable for any use that may be made of the information contained therein.

Summary

The D6.3-E Executable Process framework for certification is a public document delivered in the context of WP6, Task 6.3: Specification of a Certification Metamodel for Energy Management Deployments with regard to how has been modelled the process for certification and how it is supported by the final prototype.

This document describes the prototype developed in this task from the point of view of the final user who will take advantage of it while developing his product/system focusing on its certification.

Contents

SUMMARY	3
CONTENTS	4
ABBREVIATIONS	5
1. INTRODUCTION	6
2. OVERVIEW	7
2.1 PREREQUISITES.....	7
3. THE TOOL ENVIRONMENT	8
3.1 THE eDIANA CERTIFICATION TOOL PERSPECTIVE.....	8
3.2 PREFERENCES.....	10
4. GUIDE FOR USING: QUALIFICATION REFERENCES MODULE	12
4.1 REFERENCE.....	13
4.2 LEVEL.....	17
4.3 REQUIREMENT.....	18
4.4 ANNEX.....	23
5. GUIDE FOR USING: DEVELOPMENT PROJECT MODULE	25
5.1 PROJECT.....	25
5.2 REQUIREMENT.....	30
5.3 EVIDENCE.....	31
6. GUIDE FOR USING DESCRIPTION OF ASSESSMENT	35
6.1 PROJECT.....	35
6.2 REQUIREMENT EVALUATION.....	37
CONCLUSION	39
ACKNOWLEDGEMENTS	40
REFERENCES	40

Abbreviations

eDIANA	Embedded Systems for Energy Efficient Buildings
CDC	Cell Device Concentrator
iEi	intelligent Embedded interface

1. Introduction

This document is the fifth deliverable of Task 6.3, "Executable Process framework for certification" (D6.3-E), in WP6, "Compositional Verification, Validation and Certification". While tasks 6.1 and 6.2 focus on Verification and Validation issues, Task 6.3 deals with Certification aspects.

This document goes with the final prototype developed in this task. The implementation of the prototype has been based on previous works presented in this task. It started with the analysis of several standards, best practices and guides that were relevant to eDIANA components and platform. This analysis provides eDIANA WP6 partners a wide knowledge about how different types of standards could be, and the difficulties to adopt a common metamodel that would gather or combine most of them.

Deliverable D6.3-B was the first result of this analysis and the basis to create the final certification metamodel that was described in deliverable D6.3-D. This deliverable also provides full details about the metamodel use cases. Furthermore, deliverable D6.3-C has defined the first analysis of assesses process languages that will establish the basis of future extensions of the Certification metamodel and the prototype that is accompanied by this document.

This deliverable can be considered as the user manual of the delivered prototype implementation. In this document the user can find the pre-requirements of the prototype, the environment that it presents to the user, besides a guide of use, starting from the creation of a Qualification Reference or standard, the definition of the evidences that this standard has been accomplished during the project and the final assessment.

2. Overview

This section will describe the infrastructure that has been used to develop the Certification Framework and that it is needed to use it.

2.1 Prerequisites

To use the eDIANA Certification Framework tool (GEMDE Certification), some preconditions need to be fulfilled:

1. Eclipse is an open source Integrated Development Environment written in Java almost completely. It is used to develop applications and projects in Java, but not exclusively because ADA, C, C++, Cobol, Perl, PHP, Python, Ruby, Scala and Scheme can be use by means of plugins. The used version of Eclipse is Galileo 3.5, so it will be mandatory having this version of Eclipse installed in the computer. The Eclipse platform comprises next components:
 - a. Equinox OSGi: this is a standard bundling framework
 - b. Core platform: this component enables booting Eclipse, running the plug-ins.
 - c. Standard Widget Toolkit (SWT): it is a widget set that allows to build graphic user interfaces.
 - d. JFace: viewer classes to bring model view controller programming to SWT, file buffers, text handling, text editors...
 - e. Eclipse Workbench: the workbench provides the user views, editors, perspectives, wizards...
2. Plugins: these are the mechanisms to provide functionalities to the Rich Client Platform (RCP). Added to the plugins of Eclipse Galileo 3.5, the platform needs the plugins of next projects:
 - a. Eclipse Modeling Framework Project (EMF) 2.5.0v200906151043.
 - b. Extended Editing Framework (EEF) 0.8.0v201006010538.
 - c. Acceleo 3.0.0v201006010437.

The eDIANA Certification Framework has been developed as Eclipse Platform plugins that has to be added to the Eclipse plugins. Next plugins provides the functionalities of the Certification Framework:

1. Certification Qualification References: three plugins.
2. Certification Development Project: three plugins.
3. Certification Assessment Project: three plugins.
4. Certification Framework: one plugin.

5. Certification Utils: one plugin.
3. The user needs to get a minimum understanding of the eDIANA Certification Metamodel described in deliverable D6.3-D Software Engineering Process for Certification Metamodel, where the last version of the Metamodel is included.

3. The Tool Environment

3.1 The eDIANA Certification Tool Perspective

The eDIANA Certification perspective is organized in the following way; the user will work over 5 zones or views (Figure 1):

- Menu and Toolbars
- Navigation zone
- Tree view
- Properties zone
- Perspective selection

All the zones and views can be modified in size and position, but the schema that the GEMDE perspective provides is the most suitable to work with the Certification Framework.

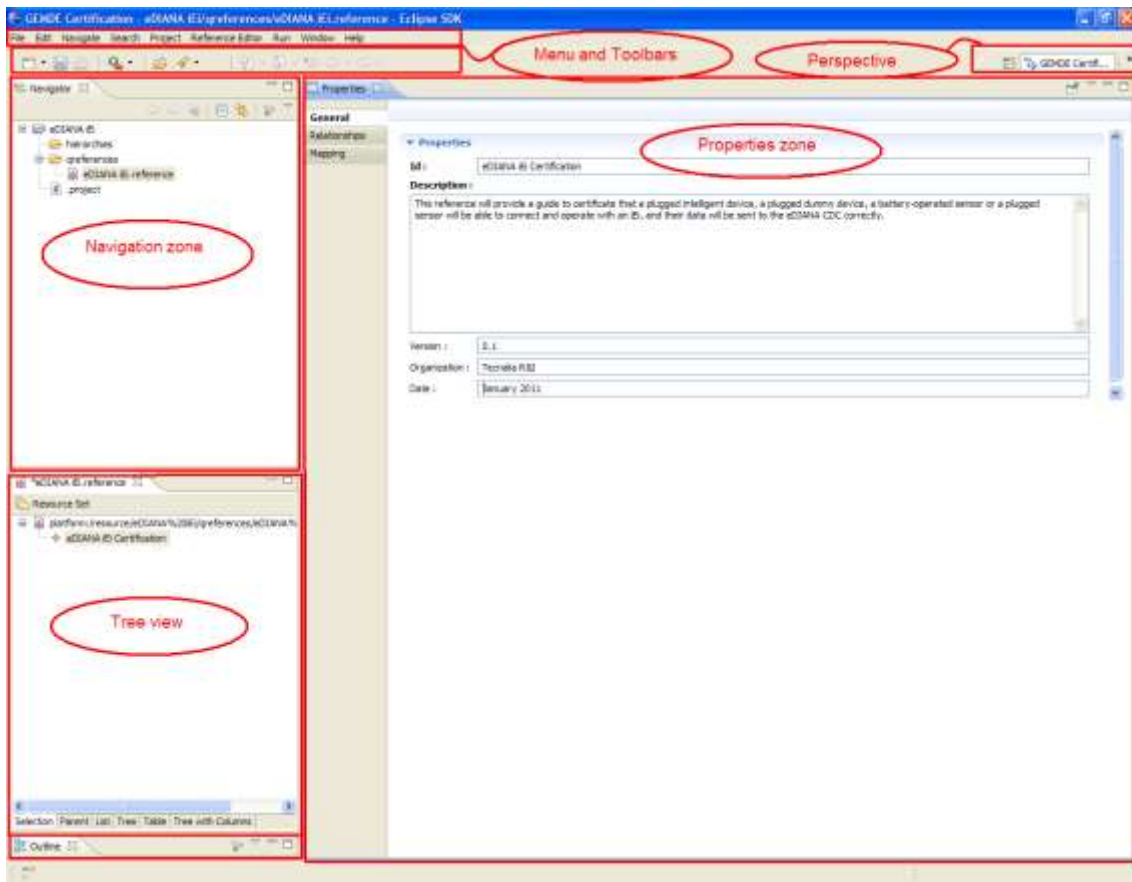


Figure 1 Certification Framework Perspective

Menu and Toolbars

This zone provide access to all the functionalities the Eclipse Platform offers through its menu: File, Edit, Navigation, Search, Project, Run, Window and Help, plus those the Certification Framework provides: Reference Editor, Dproject Editor and Assessment Editor.

The toolbars are the same as Eclipse provides.

Perspective selection

Eclipse allows the selection of the perspective in the upper right zone. The selected perspective must be GEMDE Certification perspective.

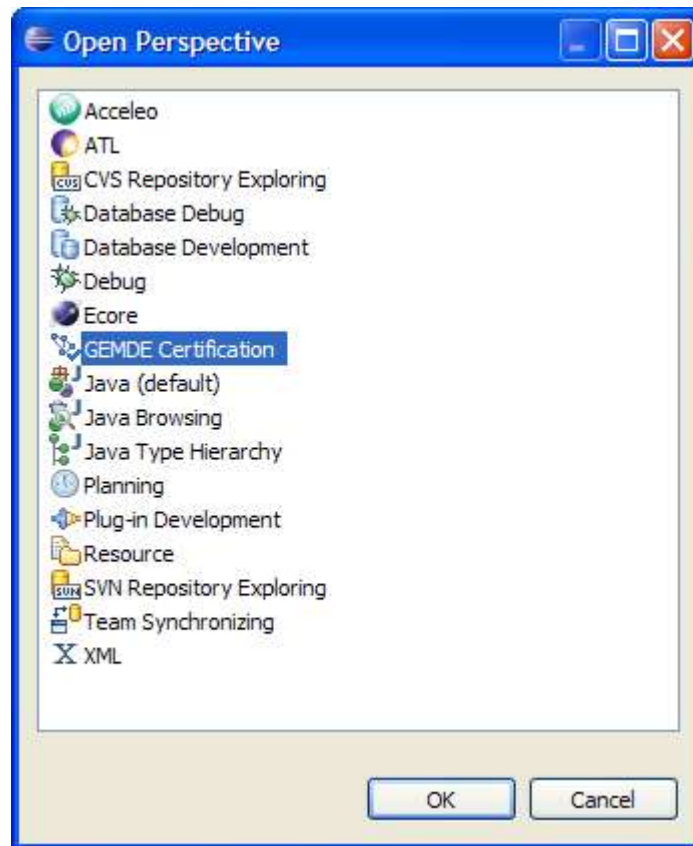


Figure 2 Open perspective

Navigation zone

This area allows the user to navigate through the projects and select the files. The creation of new files, directories will be shown here.

Tree view

This zone provides a tree view of the generated projects and elements: references, requirements, evidences, evaluations...

A hierarchical view of the elements is shown in this view.

Properties zone

This zone provides information of the selected element in the Tree View. It is organized in tabs to build a friendlier user interface.

3.2 Preferences

The Certification Framework makes a contribution to the Eclipse Platform preferences. Selecting in the menu Window-Preferences, the user can access to the preferences dialog of the entire environment.

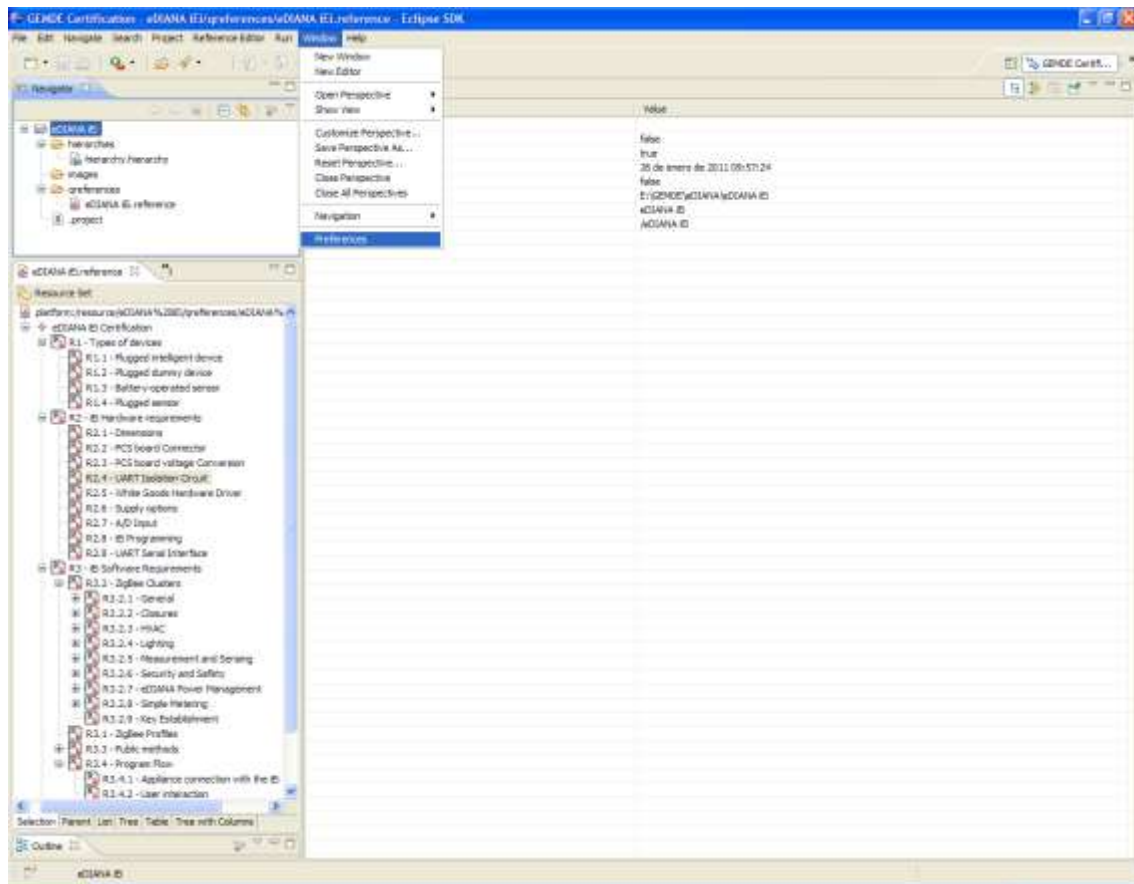


Figure 3 Preferences menu

In those preferences the Certification Framework has an entry to define the directory where the Qualification References or standards are stored. The framework will use this directory to present the user the available standards that can be associated to a Development project.

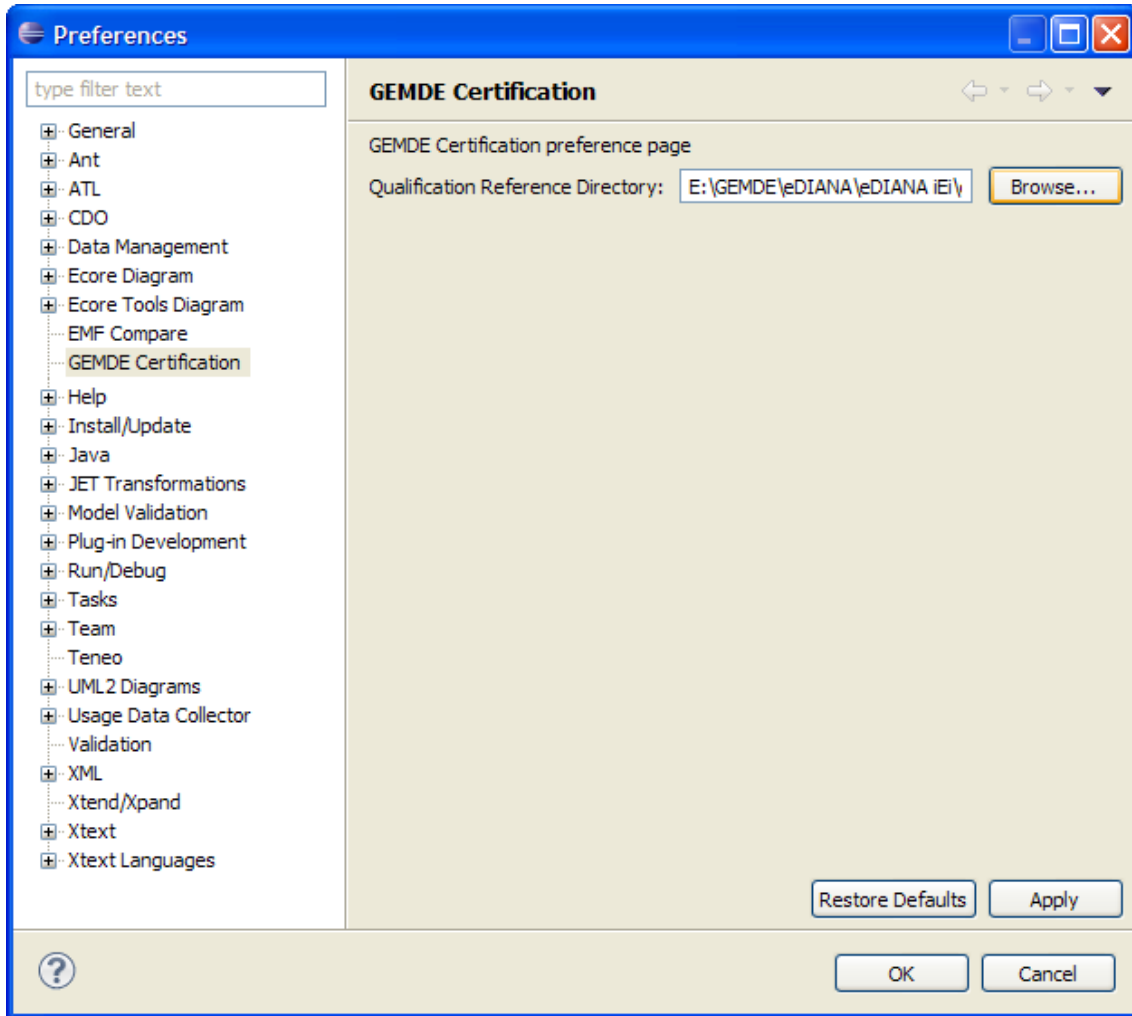


Figure 4 Preferences dialog

4. Guide for using: Qualification References module

This section will describe the functionality to create Qualification References or standards. To describe the different modules of the Certification Framework it has been defined a eDIANA Certification project that covers the creation of an standard, a development project and an assessment to evaluate the developed project against the standard.

Inside the eDIANA architecture, there is a component that has been developed at Cell level: the iEi. This component covers the communication of each component of the Cell level to the CDC. In task 3.2 a hardware and a software has been developed to cover this functionality, but as there is an undetermined number of devices that can be connected to the CDC, the task 3.2 will provide a hardware and a software that takes on the functionalities of communication with the CDC. At the same time, this created iEi will offer an open interface to connect to different types of devices (appliances, sensor, etc.).

Taking into account the documents provided by task 3.2 and task 5.1 (integration), it has been created a standard or reference that any manufacturer that wants to introduce a device in the eDIANA Platform at Cell level and wants to use this iEi, must fulfil. This standard covers hardware, software and development requirements. The description and the guide of the Certification Framework will be done attending to this iEi Certification Standard and examples of device connection implementations.

4.1 Reference

The creation of a new Qualification Reference must be made through the creation of a new project of type QReference. To generate a new project the user can select the entry of the menu File-New-Other or the same entry of the contextual menu in the Navigation zone (Figure 5).

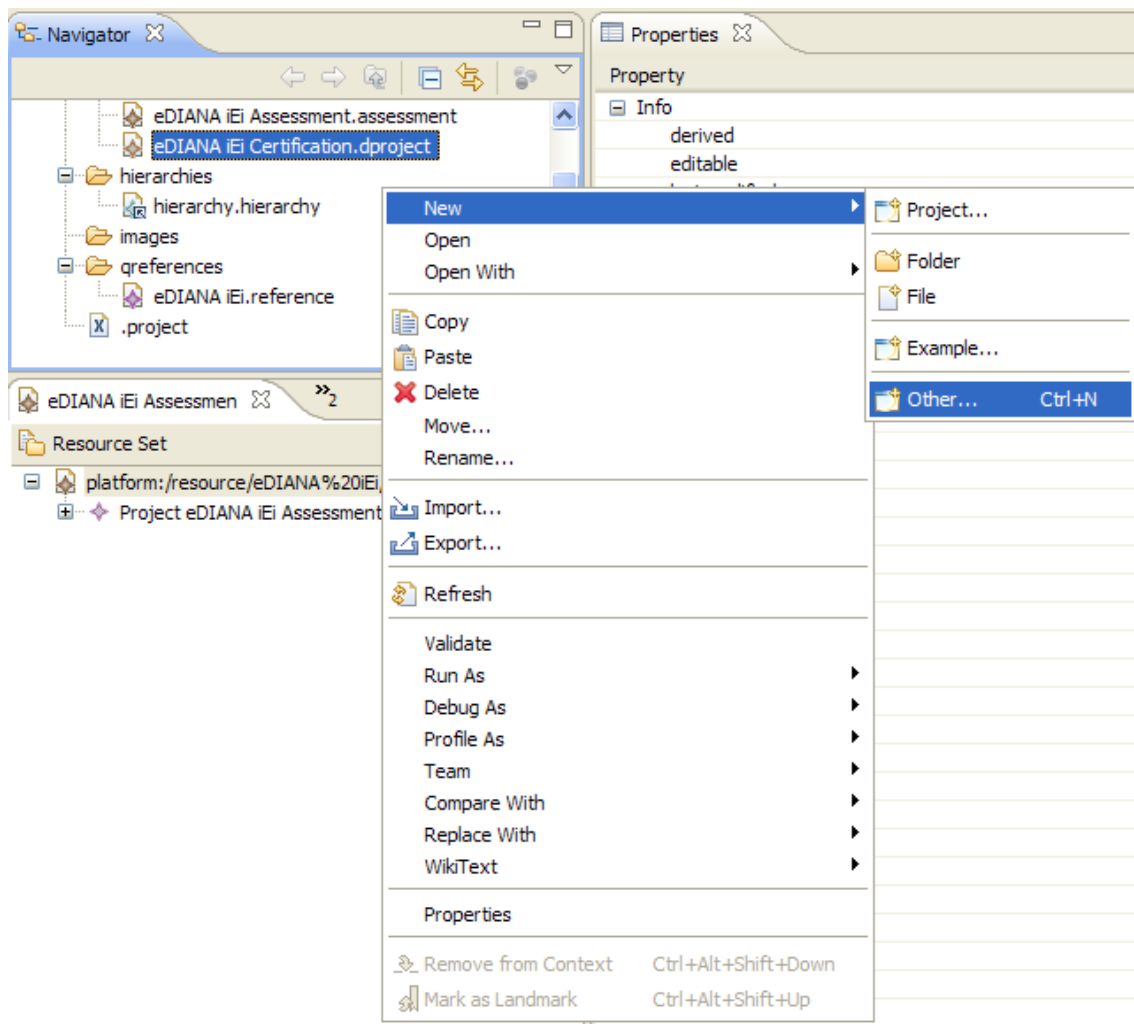


Figure 5 New Project contextual menu

This selection leads the user to a modified New Project wizard in which QReference Project must be selected:

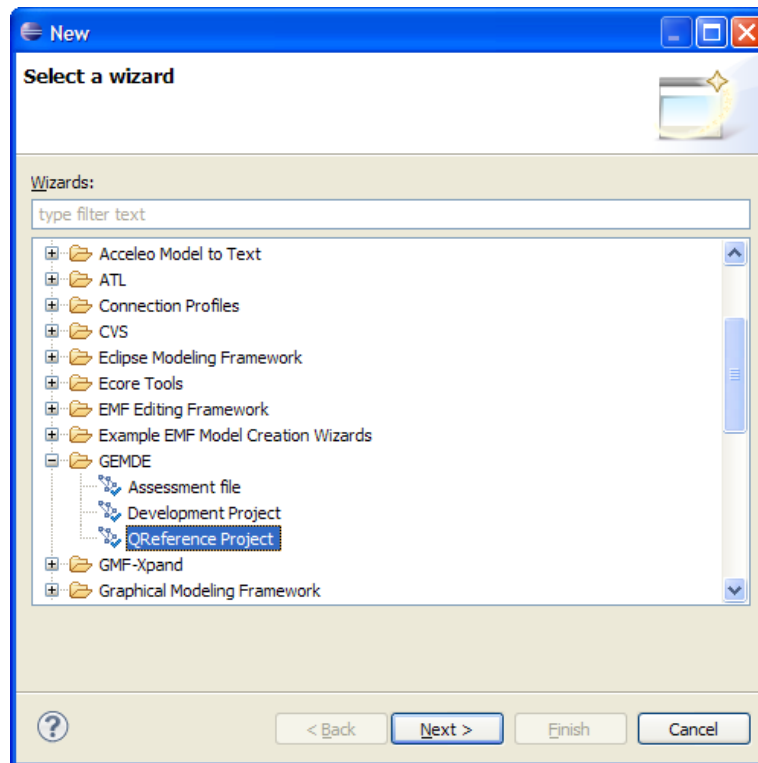


Figure 6 New QReference Project I

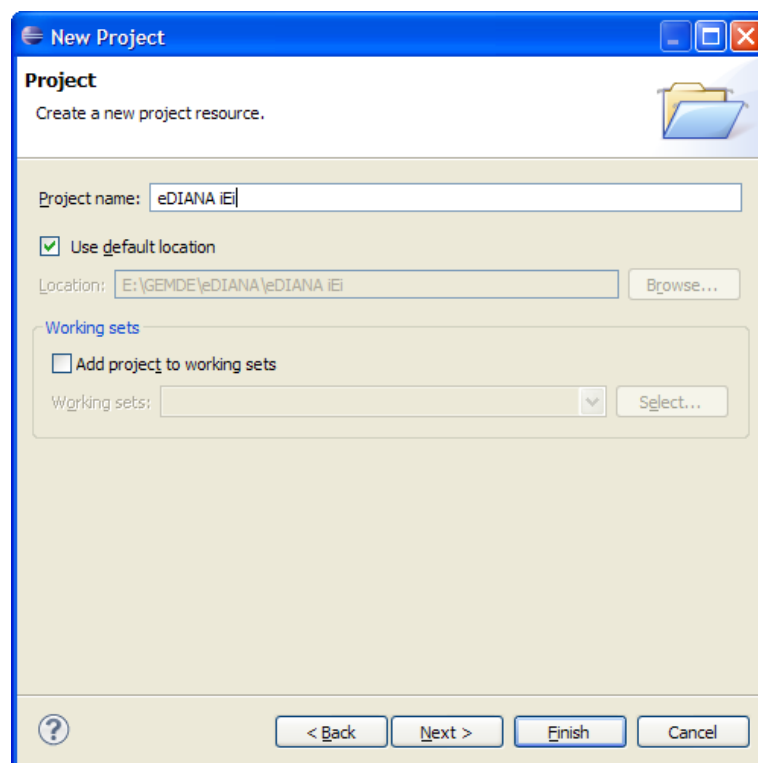


Figure 7 New QReference Project II

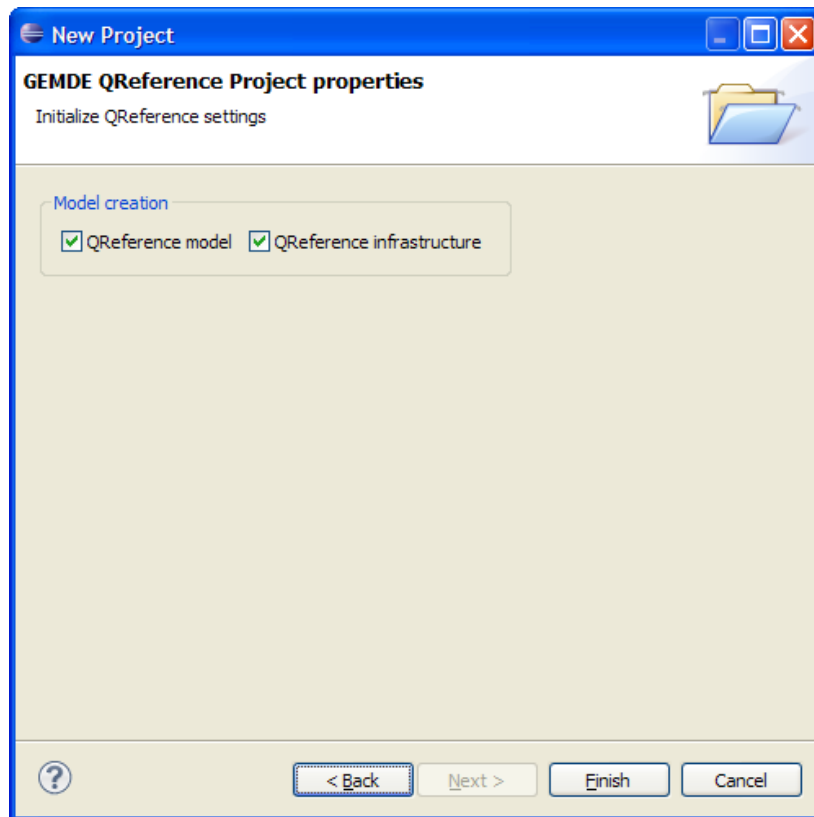


Figure 8 Qualification Reference Infrastructures

The last page of this wizard allows the user to generate automatically a directory infrastructure hanging from the project (QReference infrastructure checkbox):

- Hierarchies: directory where the hierarchy maps are stored.
- Qreferences: directory where the new Qualification Reference will be stored.
- .project: Eclipse file.

If this option is not selected, the Qualification Reference will be created and stored in the directory or project selected when the contextual menu was shown or in the project itself if it has been created from the menu.

If the checkbox QReference model is not checked, the model containing the Qualification Reference will not be created.

Once the Qualification Reference has been created, the first item is presented to the user.

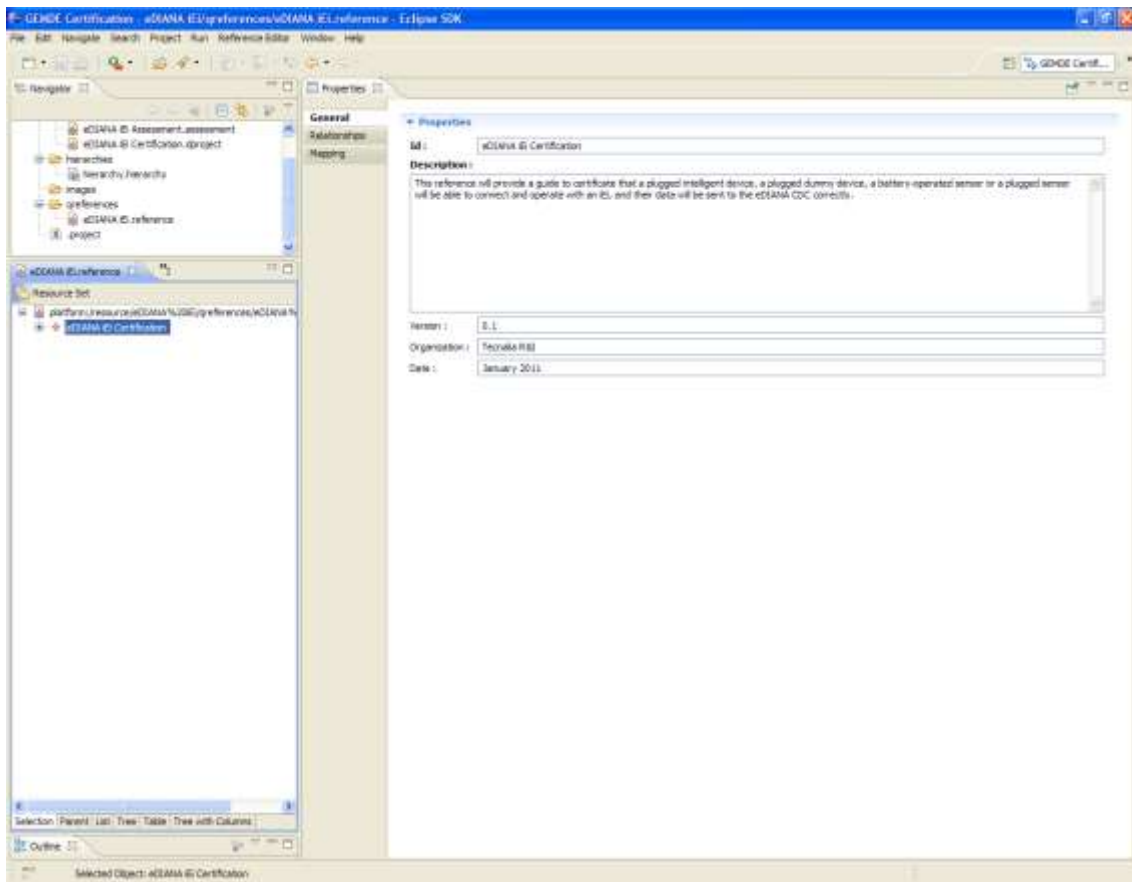


Figure 9 Qualification Reference

In the zone of properties the framework presents several fields to describe the Qualification Reference divided in tabs:

- General
 - ID: Identifier or name of the Qualification Reference.
 - Description: Description of the Qualification Reference.
 - Version: Version of the Qualification Reference.
 - Organization: Organization that has created or supports the Qualification Reference.
 - Date: Date of the version of the Qualification Reference.
- Relationships
 - Has Subsets: references to subsets or groups of requirements with a own entity. For example, the IEC 61508 has several parts which each one makes up a standard.

- Has Levels: references to grades or levels of fulfilment of the Qualification Reference or standard.
- Has Requirements: references to the requirements that form the Qualification Reference or standard.
- Has Annex: references to tables or different entries that are not consider as requirements.
- Mapping: The last tab of the properties enable the mapping of the standard to an ontology described in directory hierarchy. This mapping will allow the comparison of standards and evidences, (not implemented yet).
 - Mapping ID: Identifier of the mapping.
 - Mapping Version: version of the mapping.
 - Mapping: root element of the ontology which the standard is mapped to.

To illustrate this part of the prototype, the figures show the definition of the eDIANA iEi Certification Reference that will help to the certification of a new component that wants to be connected at Cell level with the CDC through the iEi. As it is previously said, it will be based on results of D3.2-B.

4.2 Level

The entities of class Level show different grades of fulfilment of the Qualification References. To generate a new Level, it can be made through the contextual menu, after selecting a Qualification Reference and choosing New Child-Qualification Level.

In the properties zone, there are two fields that must be completed:

- Name: Name of the level
- Description: Description of the level

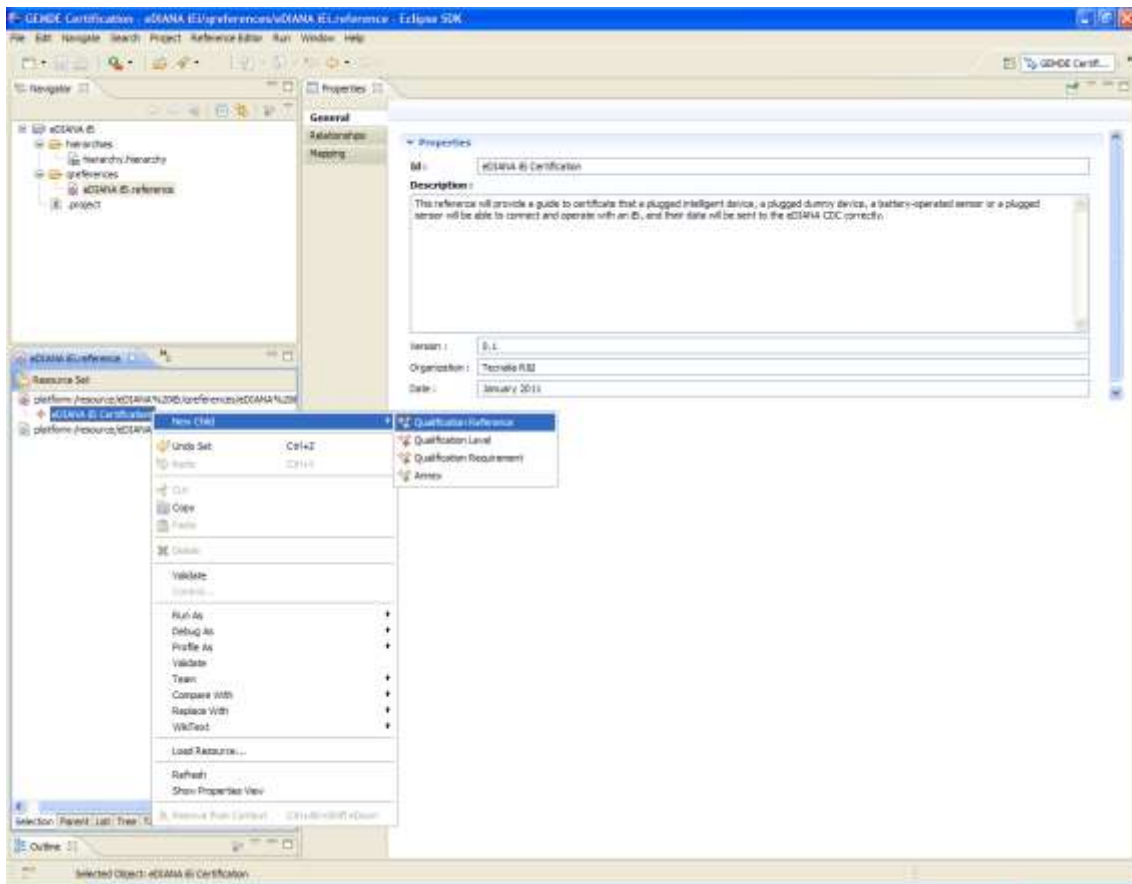


Figure 10 Qualification Reference's children

4.3 Requirement

The Qualification Reference is composed of requirements that describe the issues that must be fulfilled. The creation of new requirements is similar to the creation of levels.

The requirements can be composed of more detailed requirements; the user will define the hierarchy that the final standard will show in the framework.

Next Figure 11 shows the tabs and fields that must be defined in the requirement.

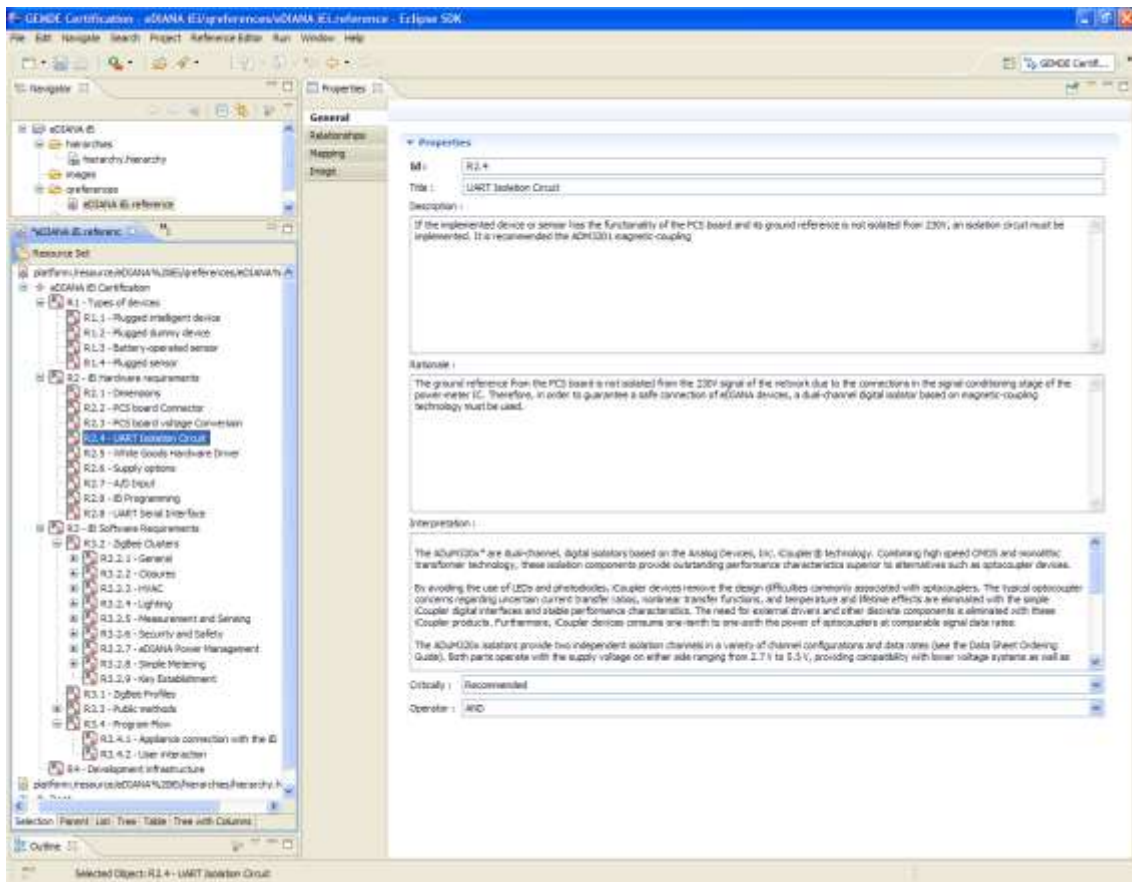


Figure 11 Qualification Requirements

- General
 - Id: Unique identifier of the requirement
 - Title: Title of the requirement. In the Tree View, the requirement will be show as the joint of id and title.
 - Description: Description of the requirement
 - Rationale: R of the existence of the requirement
 - Interpretation: possible interpretation of the requirement, or some help to implement the requirement
 - Critically:
 - Not applicable
 - Prohibited
 - Recommended
 - Mandatory

- Operator: If the requirement is under another requirement, this field shows if all the requirements of the same level of the tree must be fulfilled (AND), or if it is only one the requirement of the same level that must be fulfilled (OR).
- Relationships
 - Belongs to: indicates if the requirement is associated to a Qualification Level, it must be fulfilled to accomplish this level.
 - Depends on: references to the requirements that have a dependence relationship with this requirement.
 - Refined by: references to the requirements that define deeper this one
 - Related To: references to the entries and tables of the annex that provide more extra information about this requirement or how to fulfil it.

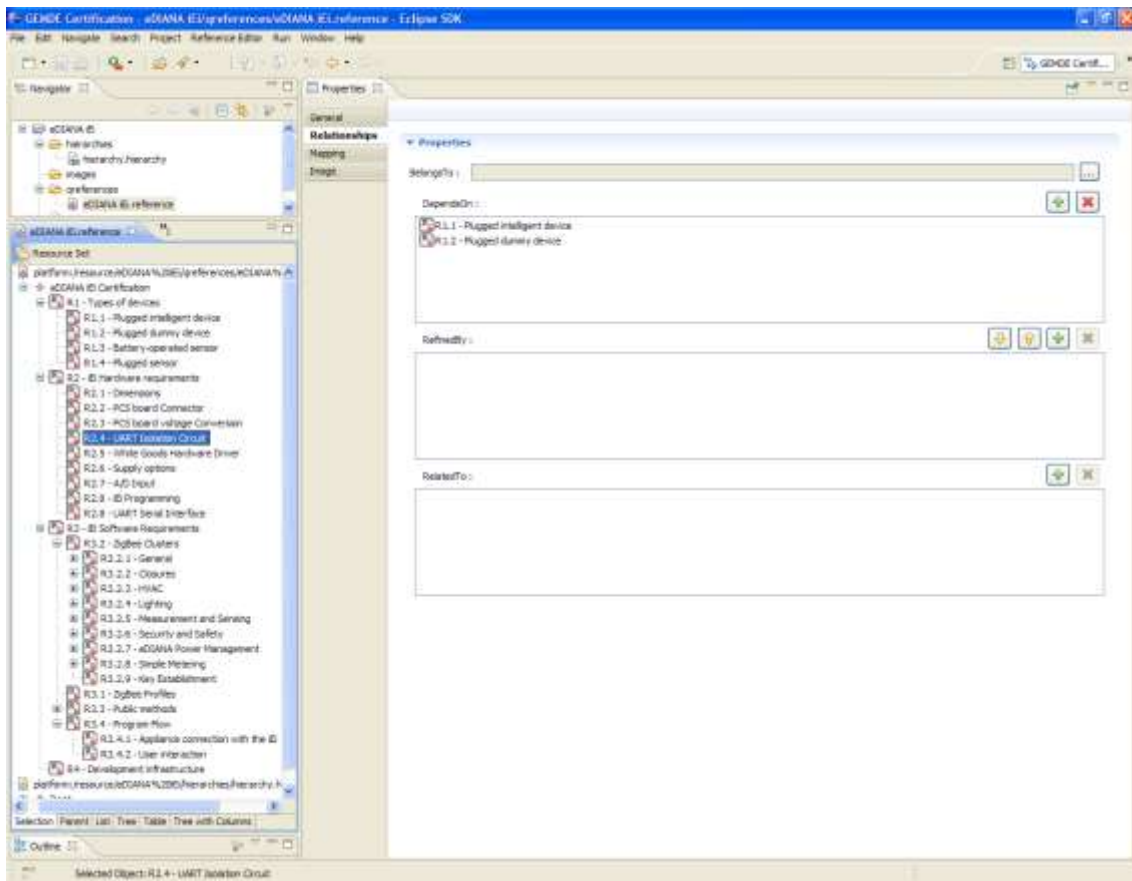


Figure 12 Requirement relationships

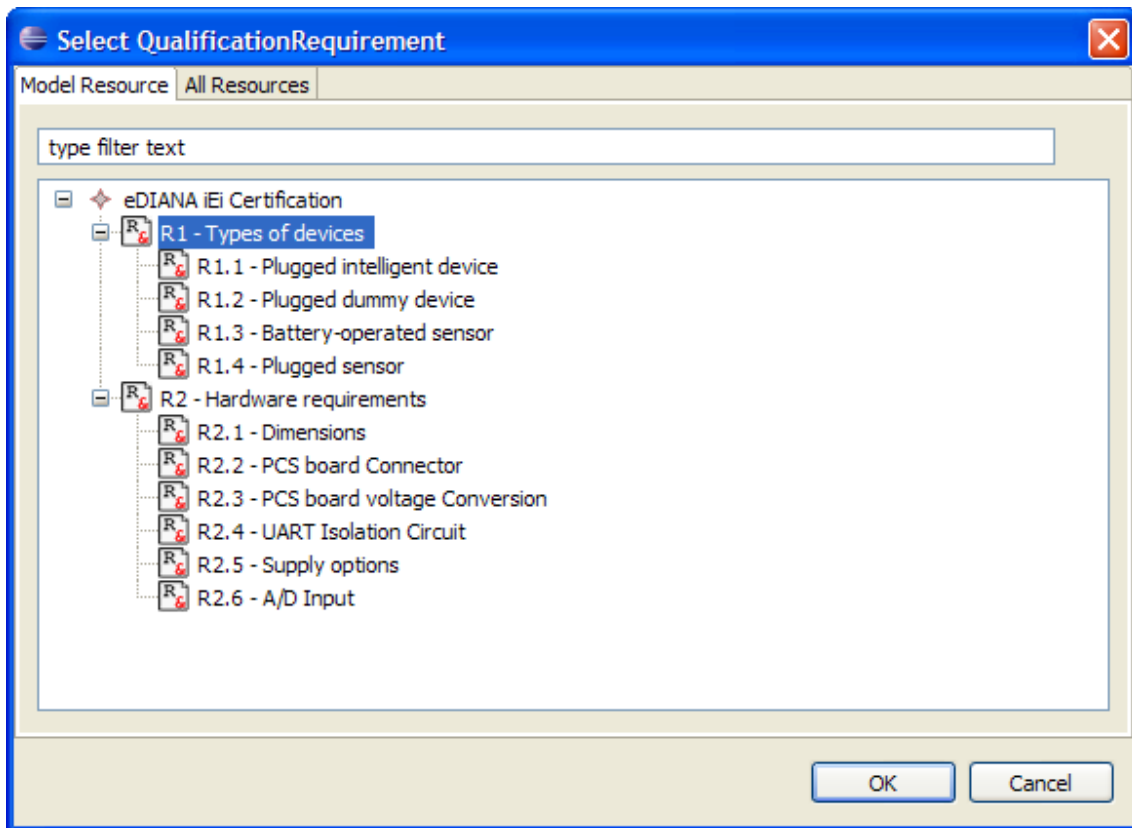


Figure 13 Selection of dependant requirements

- Mapping: references to the items of the ontology that describes the requirement.
- Image: images that help the understanding of the requirement.

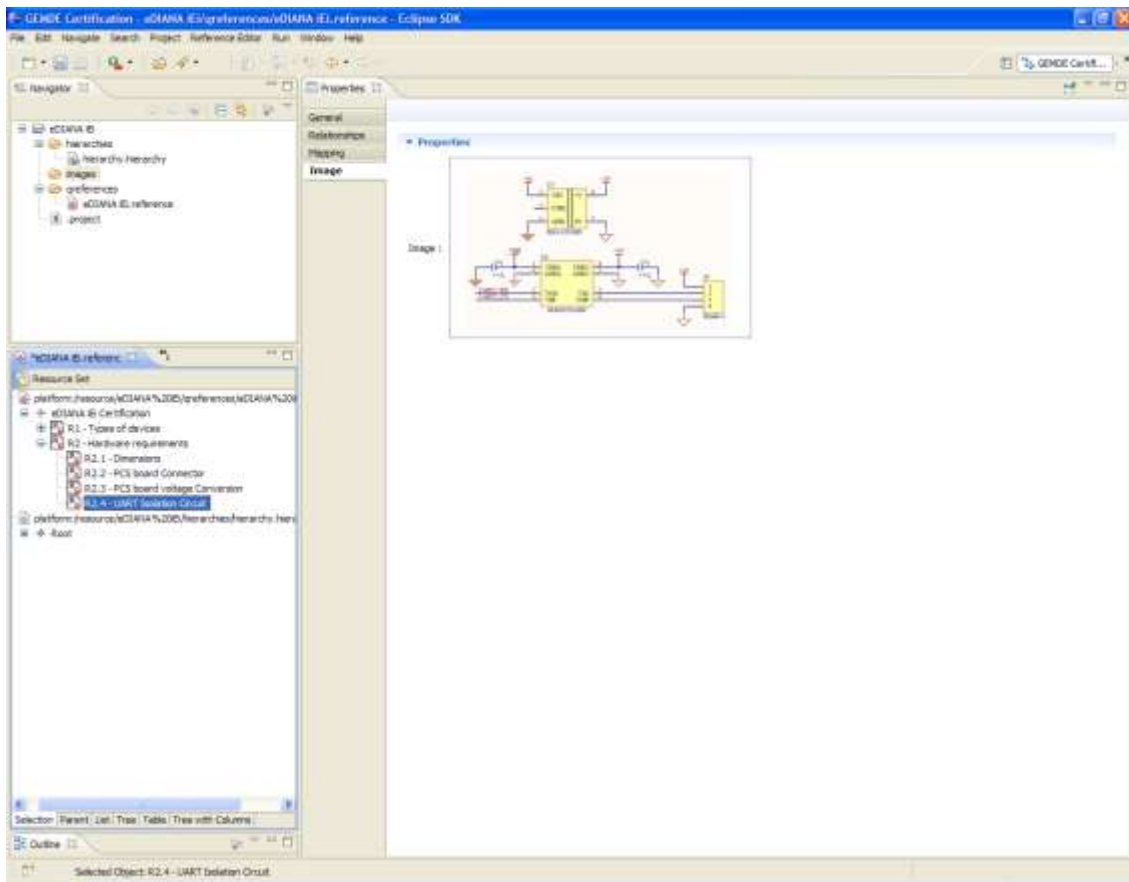


Figure 14 Requirement image

The eDIANA iEi reference is composed of several requirements grouped in the next areas:

- Types of devices. These requirements defines the types of devices that can be connected to the eDIANA iEi, as well as the modules of the iEi that must use for a right implementation
- iEi HW requirements. These are the HW requirements that the connected device must fulfil to achieve a complete integration with the iEi.
- iEi SW requirements. These are the SW requirements that the connected device must fulfil to achieve a complete integration with the iEi.
- Development infrastructure. This requirement is about the needed infrastructure to develop the HW and especially the SW of the iEi that will allow the connection of the device to the CDC.

4.4 Annex

Most of the standards are assisted by tables to explain the requirements that must be fulfilled. The class Annex allows the addition of entries, some standards like IEC 61508 provides techniques and measures that help to achieve the accomplishments of the requirements; the Annex entities make possible to associate those entries with Qualification Levels and recommendations.

The Annex has the next fields:

- Name: Name of the annex.
- Description: Description of the annex.
- Has Table: References to the tables of the annex.
- Has Entries: References to the entries of the annex.

Inside an Annex the user can have Entries and Tables. The entries can be techniques, methods, measures, etc., ultimately, any help to achieve a requirement. The entries are organized hierarchically in the tree and have the next fields to introduce information:

- ID: Unique identifier of the entry.
- Title: Title or name of the entry.
- Description: Detailed description of the entry.
- Operator: The value OR indicates that the entry is an equivalent entry to other entries at the same tree level with OR operator, so only one of the alternate or equivalent entry has to be taken into account or satisfied. The value AND shows that it has not equivalence and should be taken into account.
- Composed by: References to the entries that explain in more detail the actual entry.

The tables have the next fields:

- ID: Unique identifier of the table.
- Title: Title or name of the table.
- Description: Detailed description of the table.
- Has Registers: References to the registers of the table.

The table is composed of registers; those registers make the association between the entry, the recommendation and the Qualification Level. The fields that can be completed are:

- Recommendation: Short description of the recommendation that the standard gives for the entry to achieve the Qualification Level.
- Description: Description of the recommendation
- Comment.
- Relation X: Qualification Level associated to the recommendation
- Relation Y: It can be a table or an entry.
- Group: When several entries are equivalent, and only one of the equivalent entries should be selected, this field allows the user grouping several equivalent entries. "0" is used for non equivalent entries, other numbers determine the group of the equivalent entries.
- Operator: "AND" marks the entry as non equivalent, "OR" mark it as equivalent.

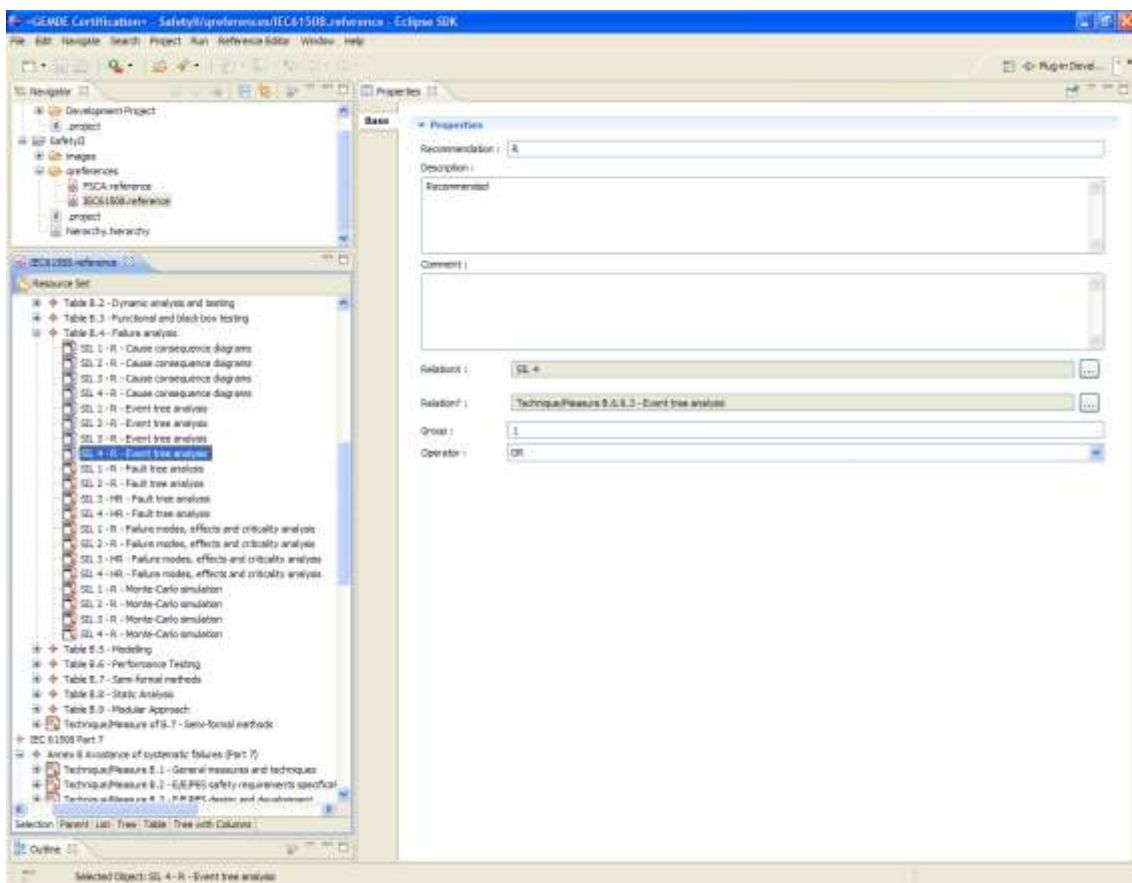


Figure 15 Annex

5. Guide for using: Development Project module

This section will describe the functionality to create Development Projects, this project provide support to define the evidences that the developed project accomplish the Qualification Reference.

Going on with the example that we are showing in the figures, it has been created a Development Project that will cover the requirements of the eDIANA iEi standard needed to implement an iEi that would be connected to a washing-machine.

5.1 Project

The generation of a new Development Project is similar to the generation of a new Qualification Reference plus the selection of the associated reference. It can be made through the menu or the contextual menu, and selecting "Development Project" in GEMDE option.

The first steps of the modified new project wizard are quite similar, but this wizard includes a page that allow the user to select the standard and the requirements that he wants to implement in the development of the product.

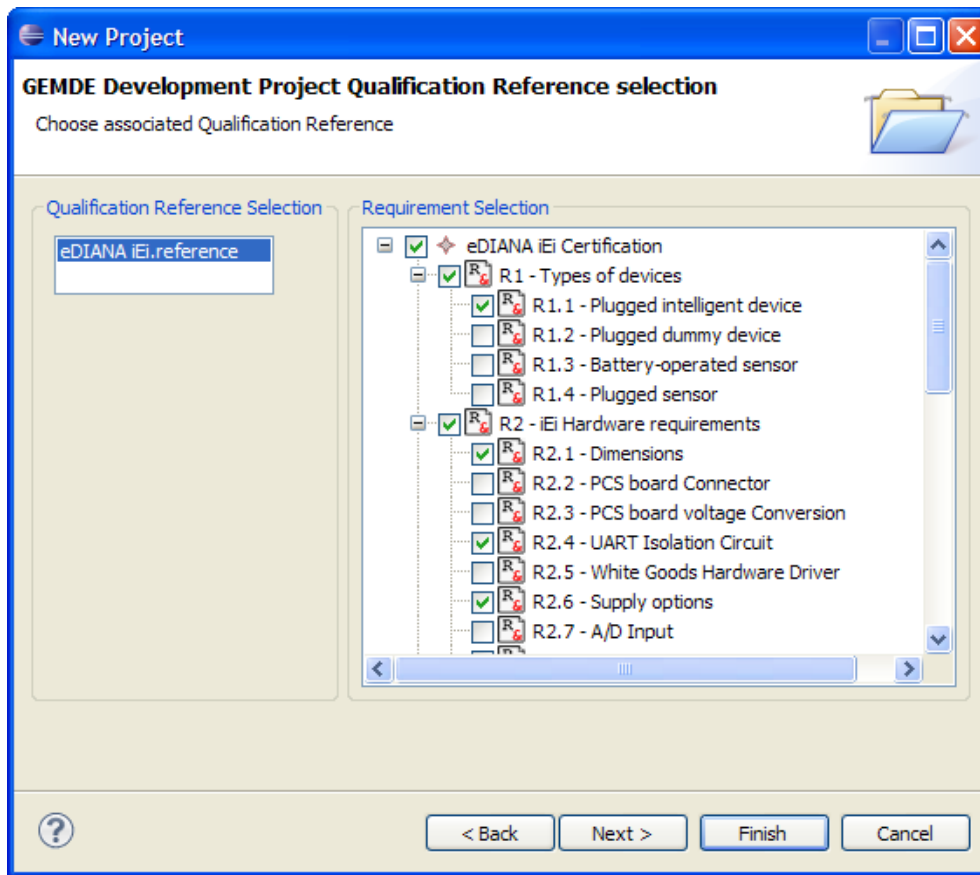


Figure 16 Reference requirements selection

The wizard presents a list box with the available references. They have to be stored in the directory established in preferences. Once the user selects the desired reference, a checkboxtree with the requirements of the reference is shown to the user. It is possible a partial selection of requirements.

This wizard presents a page that allows the user to select the creation of a directory infrastructure. If the checkbox of the infrastructure is set, the framework creates a new Eclipse project and a directory "Development Project" and generates there the file of the project. If the checkbox is not set, the framework generates the file inside the directory "Development Project" of the actual Eclipse project

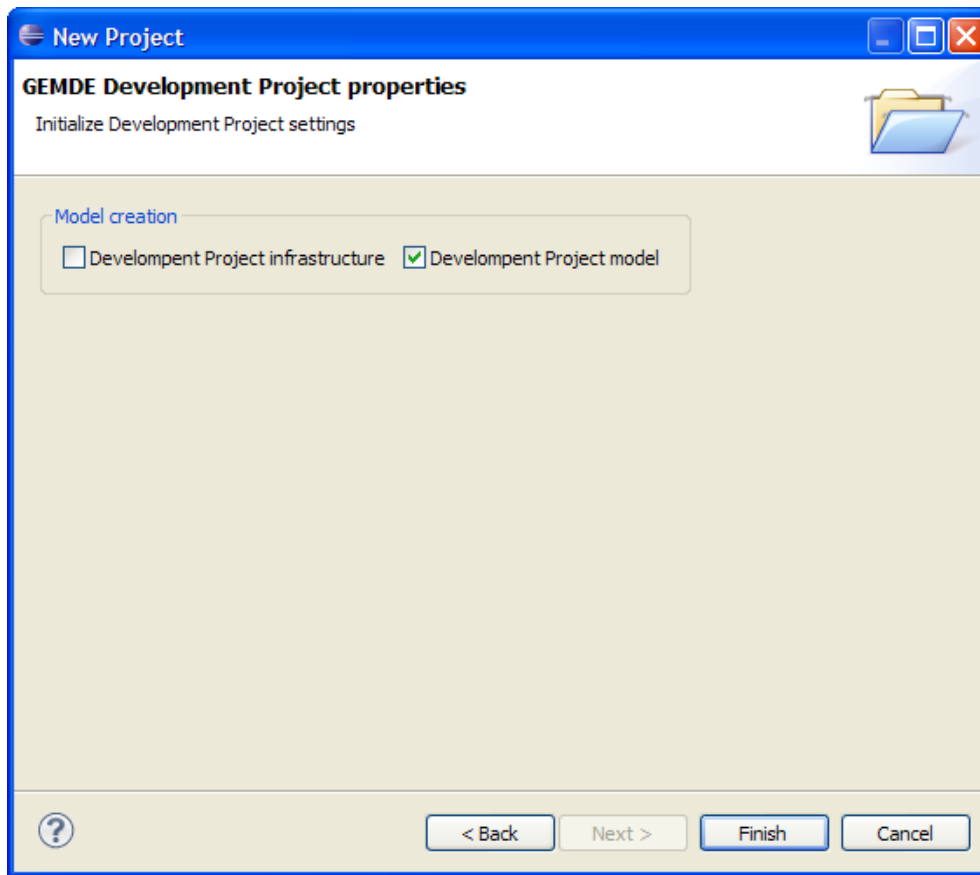


Figure 17 Development Project infrastructure

The wizard generates a new development requirement per each standard requirement; it associates them, besides associating the development project to the standard or reference.

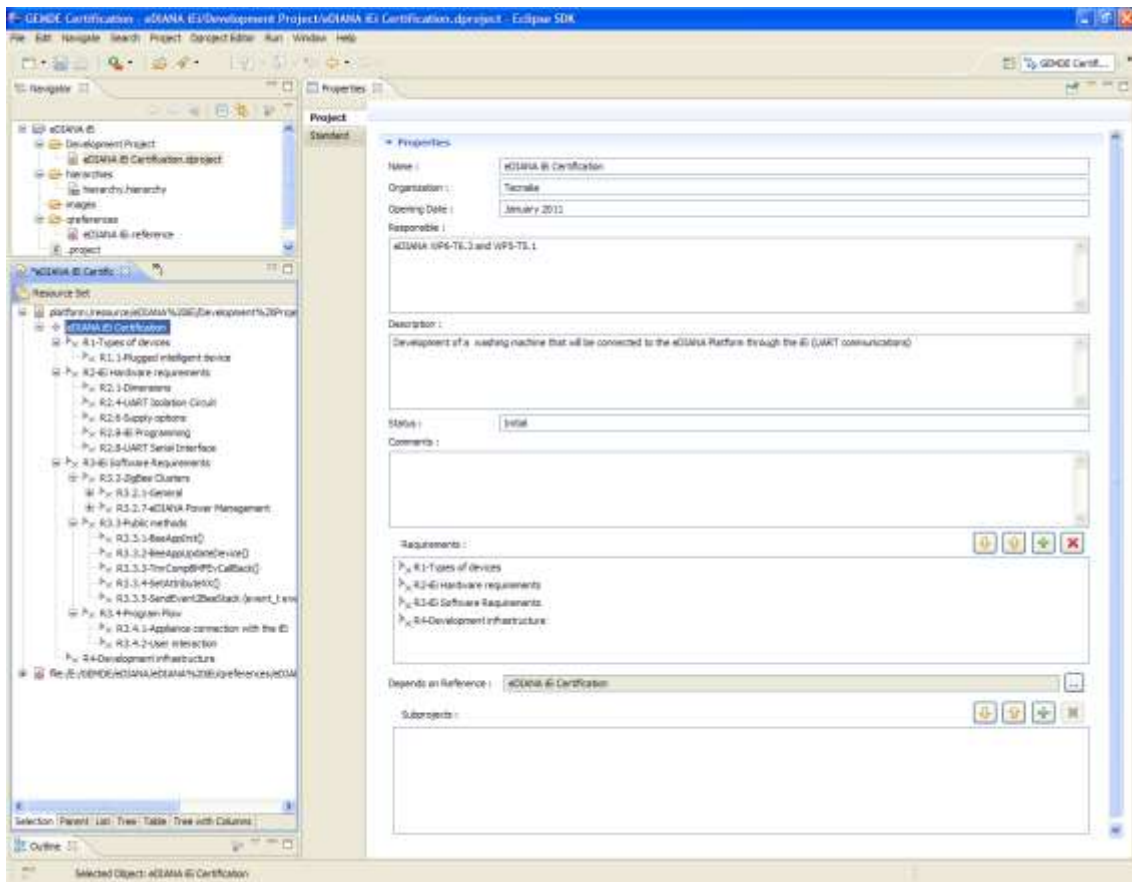


Figure 18 Development project view

This view is organized in tabs, the tab Project shows information about the development project, and the tab Standard shows information about the associated Qualification Reference.

The fields of tab Project are:

- Name: Name of the project. The framework set it by default.
- Organisation: Name of the organisation that will implement the project.
- Opening Date: Project start date.
- Responsible: Name and title of the responsible of the project.
- Description: Description of the project.
- Status: Status of the project.
- Comments.
- Requirements: references to the requirements of the project.

- Depends on Reference: Qualification reference to which the project is associated. This field is completed by the framework.
- Subprojects: Reference to Development projects. The framework generates the same structure that is in the Qualification Reference. If the selected requirements are part of defined subsets, the subprojects are generated.

The information provided in tab Standard is obtained from the associated Qualification Reference:

- Reference Name.
- Reference Description.

The implementation of the eDIANA iEi reference, the eDIANA iEi Certification project will not have all the requirements provides in the reference, there must be selected those ones that affect the new component that will be attached to the iEi. For example, in the definition of a project to integrate the iEi inside a washing-machine, the next requirements and group of requirements would be selected:

- R1 Types of devices
 - R1.1 Plugged intelligent device
- R2 HW requirements
 - R2.1 Dimensions
 - R2.4 UART isolation Circuit
 - R2.6 Supply options
 - R2.8 iEi Programming
 - R2.9 UART Serial Interface
- R3 iEi SW requirements
 - R3.2 ZigBee Clusters
 - R3.3 Public methods
 - R3.4 Program flow
- R4 Development infrastructure

Evidences should be provided to every requirement or group of requirements.

5.2 Requirement

The requirements of the Development project are generated by the framework. It associates them to the corresponding requirements of the standard.

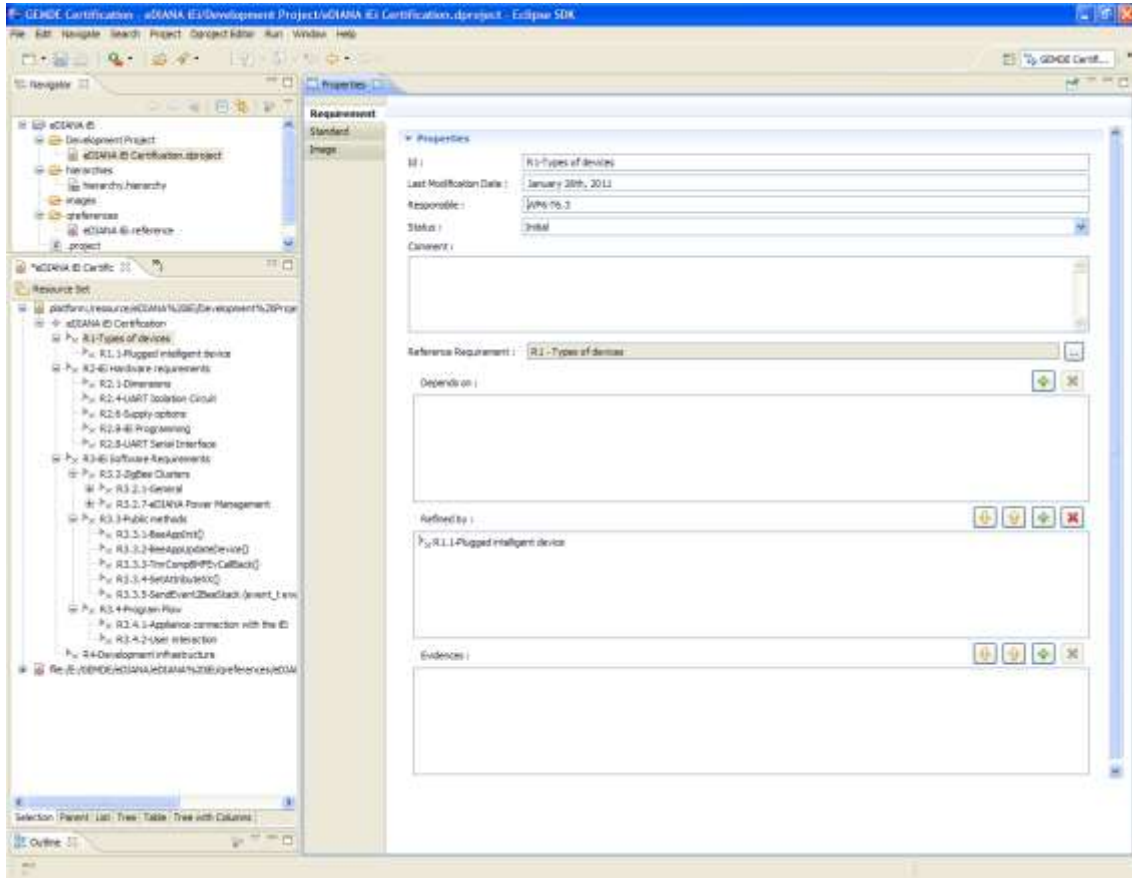


Figure 19 Development project requirement

The framework completes some fields by default with information of the associated reference requirement.

The fields of tab Requirement are:

- **Id:** Identifier of the requirement. The framework completes this field with information of the reference requirement, although the user can change it.
- **Last Modification Date:** Date of the last modification of this requirement.
- **Responsible:** Person in charge of establishing the evidences of the requirement.
- **Status:** Status of the requirement indicating if it has been completed, if it is ongoing, etc.

- Comment.
- Reference Requirement: associated requirement of the Qualification Reference or standard.
- Depends on: References to requirements on which it depends.
- Refined by: References to requirements that define it more explicitly.
- Evidences: References to the evidences that show the accomplishment of the requirement.

Other tabs of this view show different information of the associated Qualification Requirement. This information can not be modified by the user.

5.3 Evidence

A requirement can have several evidences attached to a status and a date. Furthermore, evidence attached to a specific status and date can have several documents that show the accomplishment of the requirement.

The evidences can be created at any level of the requirements tree. If the evidence is attached to a requirement that is refined by several requirements, it is supposed that the evidence will cover all those requirements.

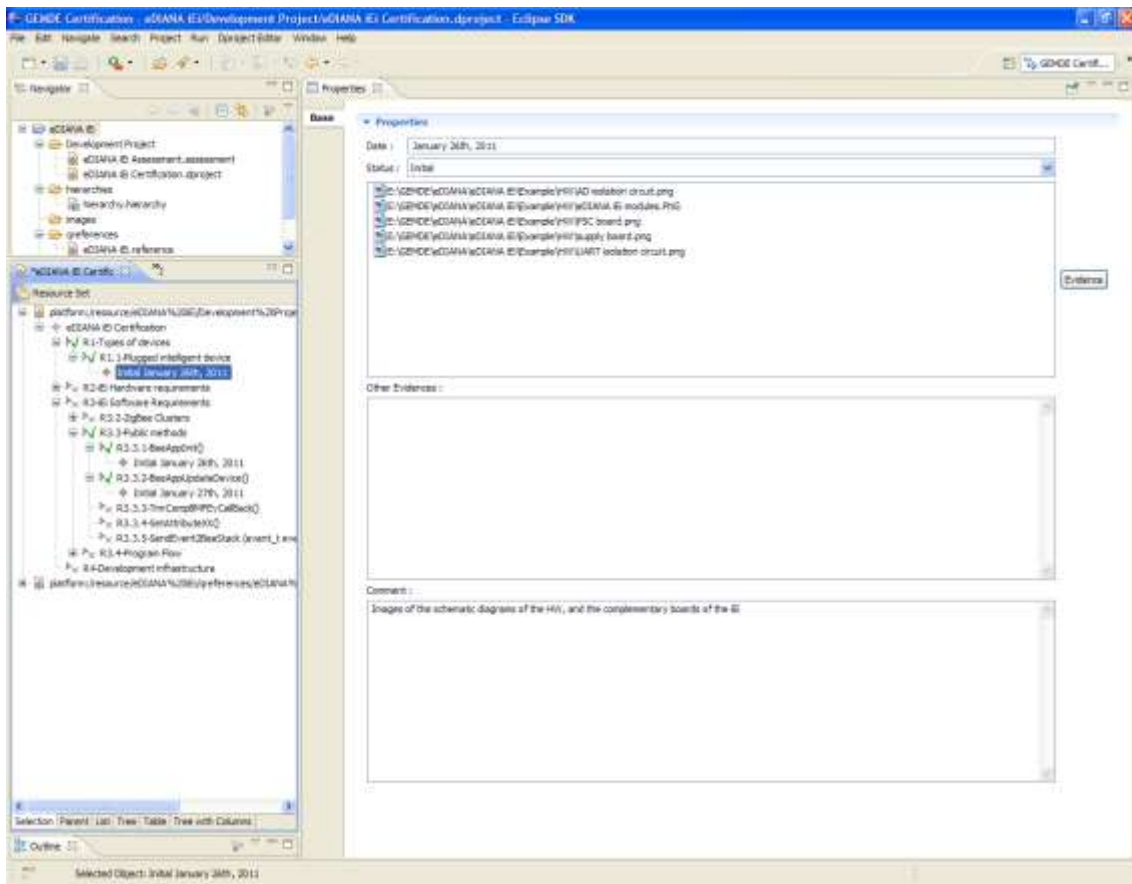


Figure 20 Evidence

The fields of the evidences are:

- Date: Date when the evidence has been defined.
- Status: Sentence that indicates if the evidence has been finished or not.
- Evidence: Documents that show how the requirement has been achieved.
- Other Evidence: If the requirement can not be accomplished by means of a digital document, this text will demonstrate or indicate how the requirement has been fulfilled.
- Comment.

To select the documents that will form the evidence, the framework offers next dialog.

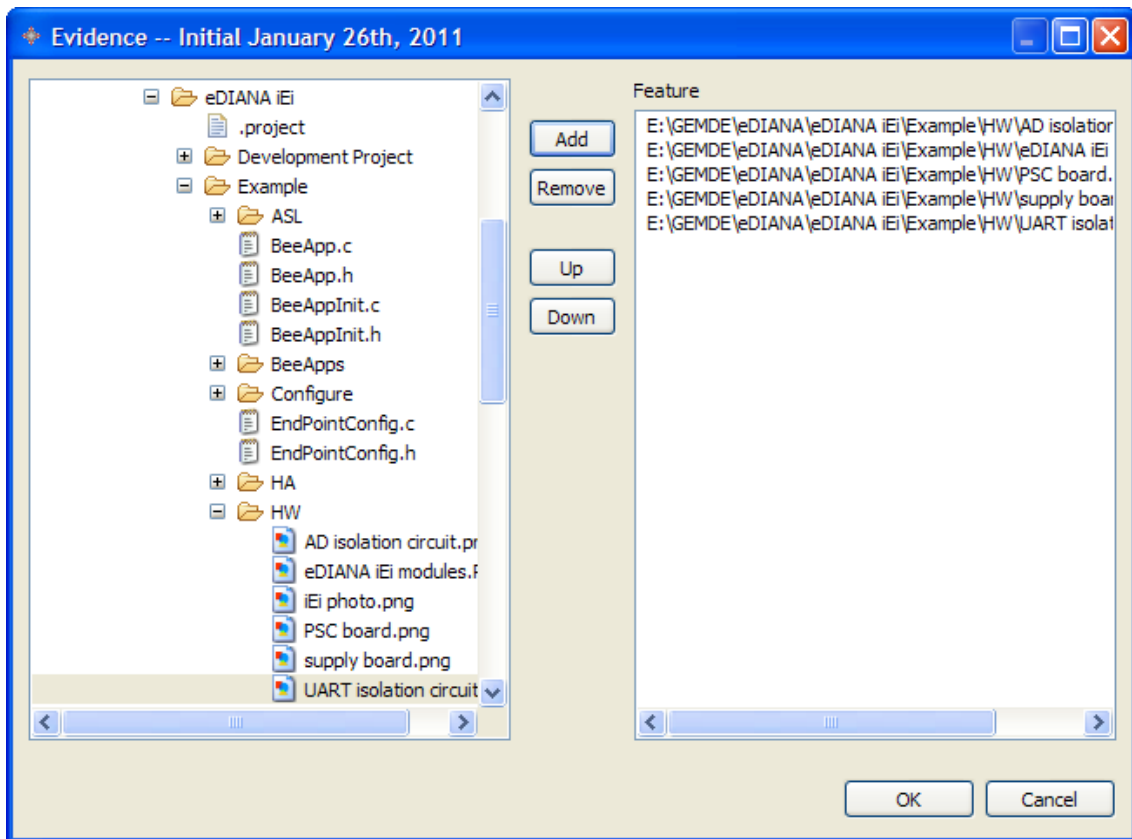


Figure 21 Evidence dialog

This dialog has a system browser on the left side that allows the user selecting the files that will be added to the evidence.

Double clicking twice on evidence, the framework will launch the corresponding application to open the file.

Executable Process framework for certification

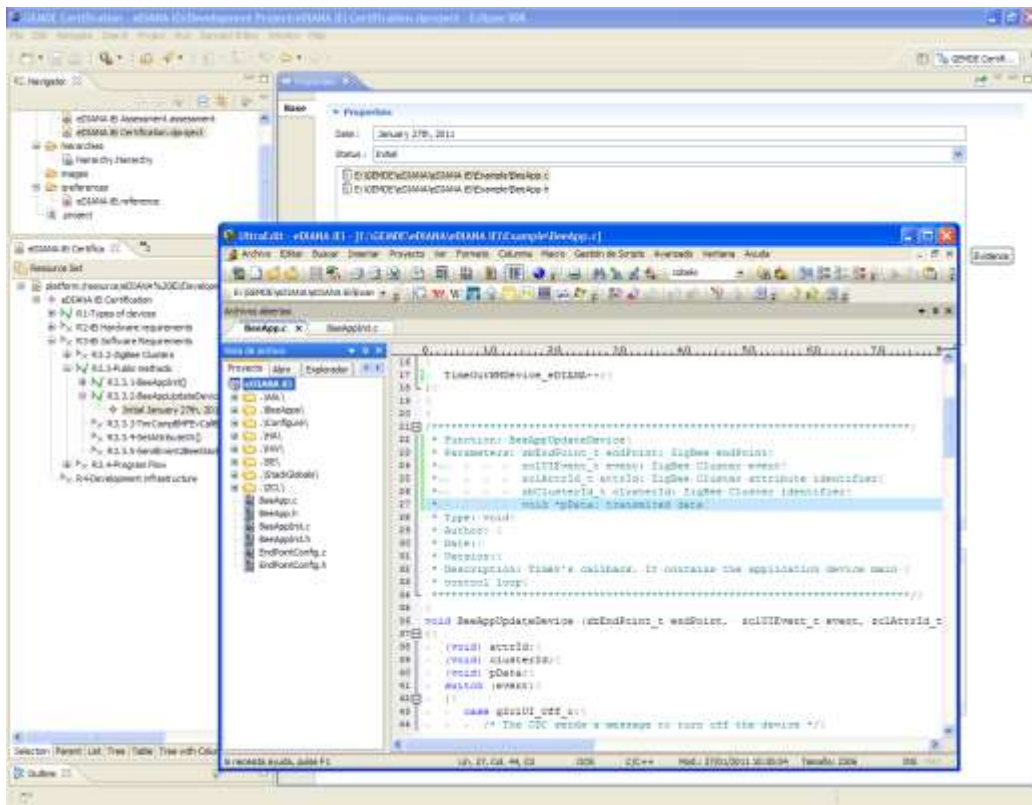


Figure 22 Double clicking on evidence I

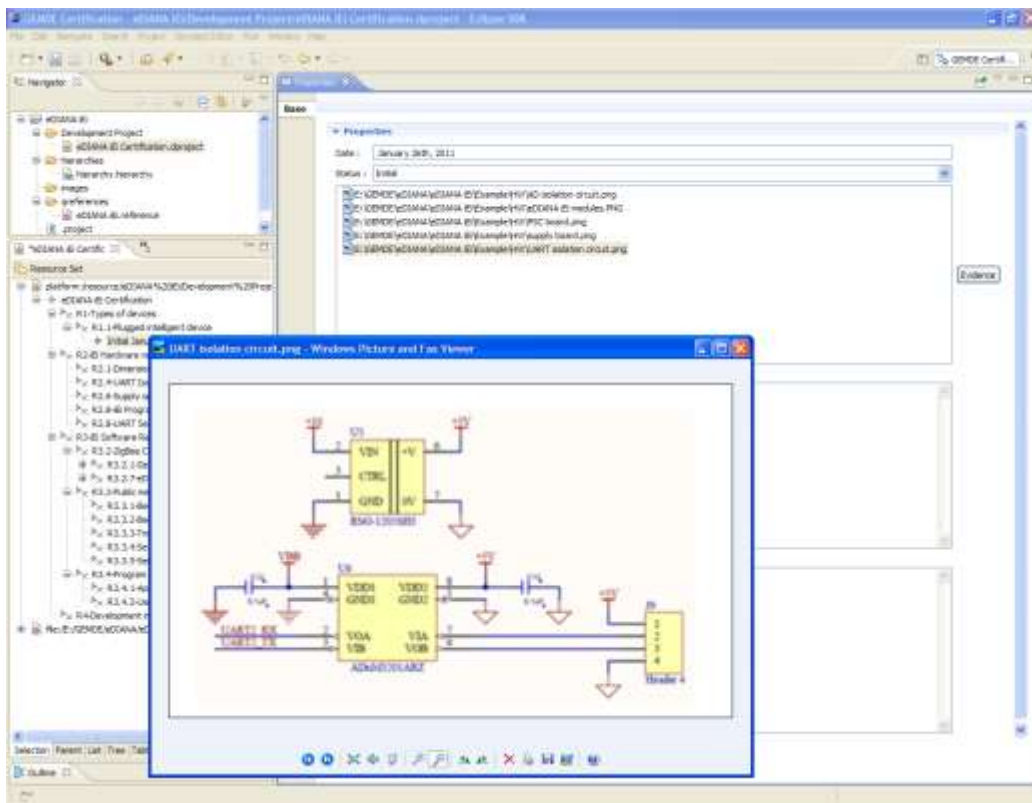


Figure 23 Double clicking on evidence II

6. Guide for using Description of Assessment

Once the evidences have been defined for each requirement, the user can generate the Assessment project that will help to the final assessor to establish if the defined evidences effectively accomplish the requirements.

6.1 Project

The generation of a new Assessment Project can be made through the menu or the contextual menu: New-Other, and selecting Assessment File.

The wizard asks the user to select the associated Development Project as well as the name of the assessment.

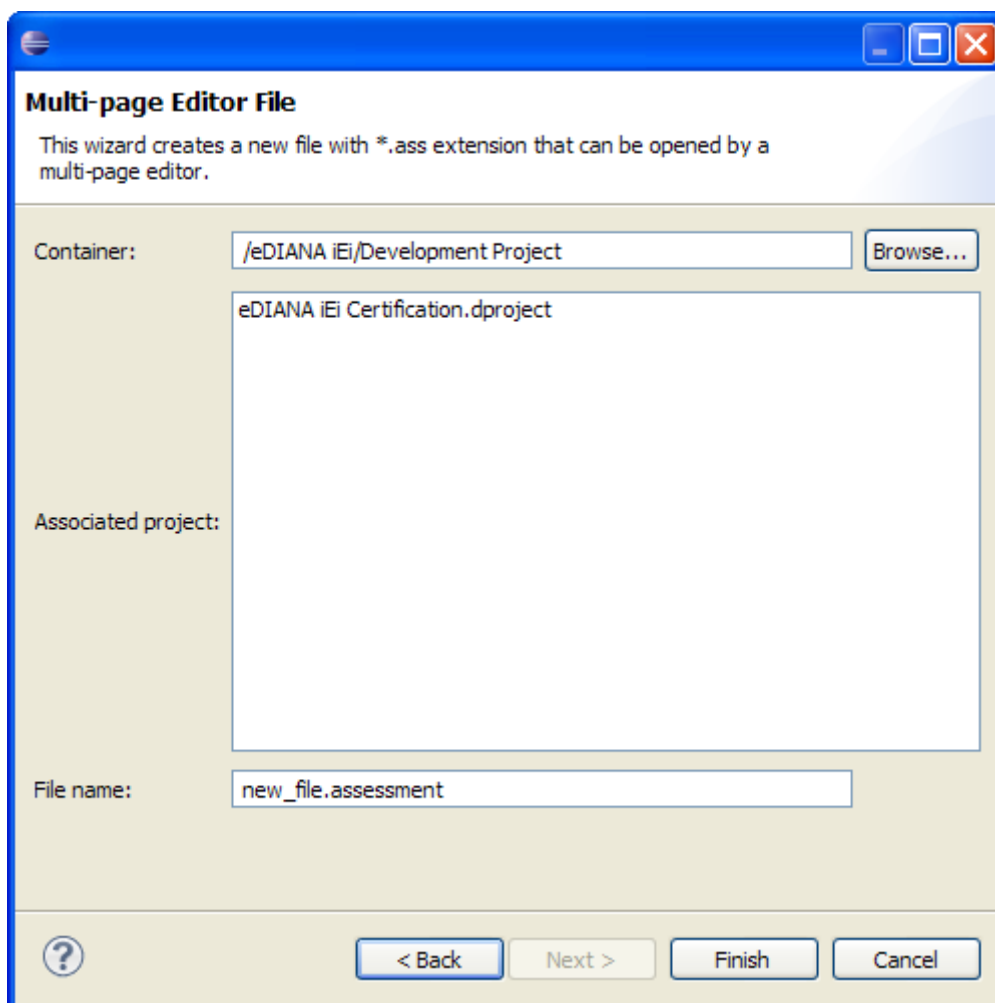


Figure 24 Assessment wizard

Once selected the Development project, the framework replicate its structure to generate the Assessment project and its requirements evaluations.

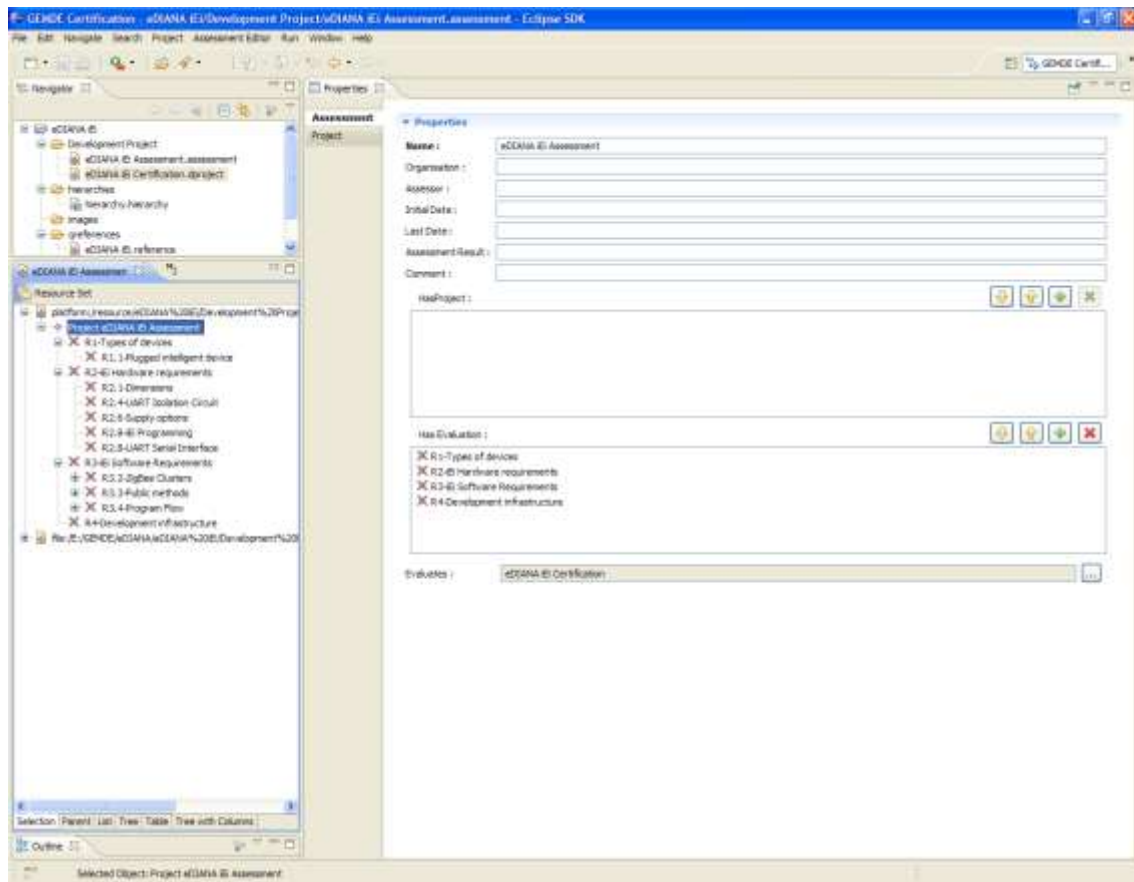


Figure 25 Assessment project

The fields of the Assessment are:

- Name: Name of the assessment, provided by default by the framework.
- Organisation: Name of the organisation that will perform the assessment.
- Assessor: Name of the people in charge of performing the assessment.
- Initial Date: Date when the assessment starts.
- Last Date: Date when the assessment finishes.
- Assessment Result.
- Comment.
- Has Project: If the associated development project has subprojects, they are replicated in this structure. This fields shows the references to this subprojects.
- Has Evaluation: References to the requirements evaluations.
- Evaluates: Identifier of the associated development project.

The tab Project provides information of the associated Development Project its related Qualification Reference.

6.2 Requirement Evaluation

The requirement evaluation allows the assessor to define if the requirement has been fulfilled, in which grade and possible comments to the evaluation.

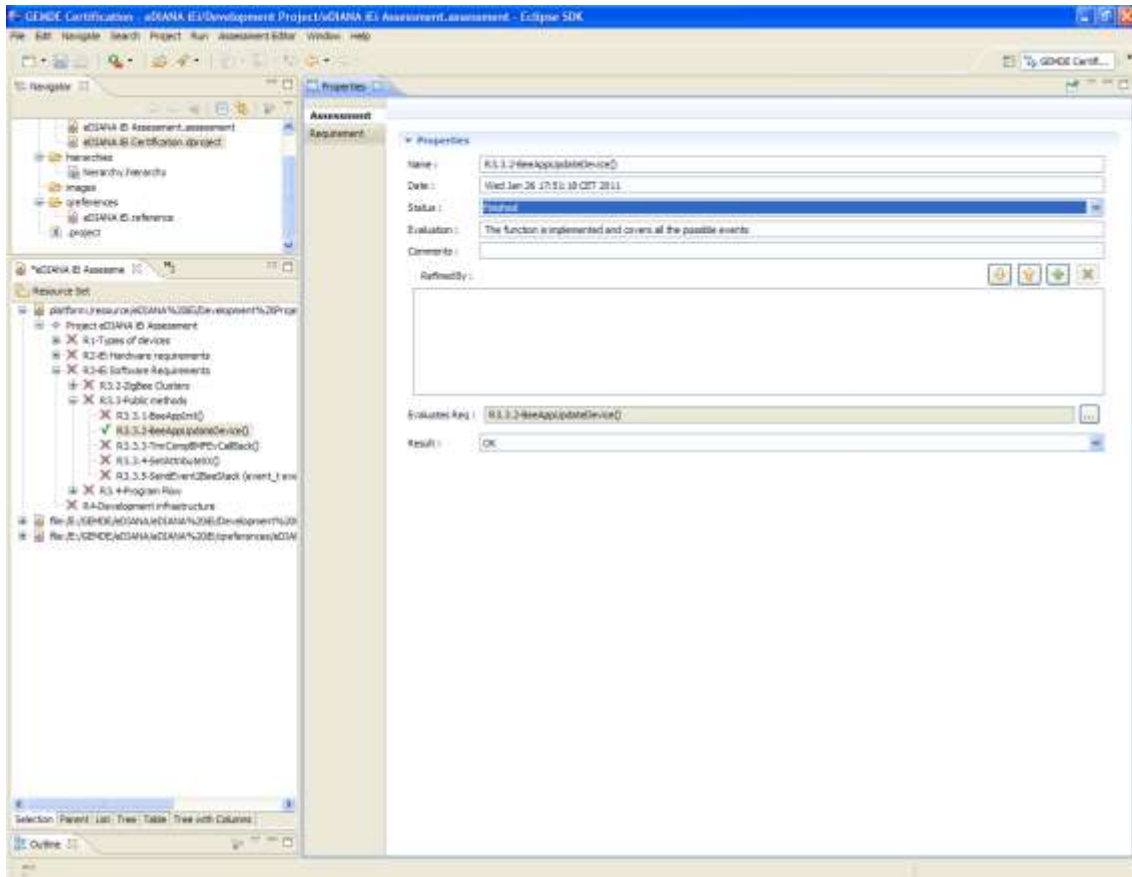


Figure 26 Requirement Evaluation

This task is supported by the next fields of tab Assessment:

- Name: Name or title of the requirement evaluation. The framework completes it by default with the identifier of the requirement.
- Date: Date when the assessment is performed.
- Status:
 - Initial
 - Ongoing
 - Finished

- Other
 - Evaluation: Final evaluation of the requirement. The assessor should explain here if the requirement has been fulfilled correctly with the evidences that have been provided.
 - Comments.
 - Refined by: Evaluations of refined requirements.
 - Evaluates: Associated requirement that is assessed.
 - Result: Short description of the result of the assessment: OK, KO, Undetermined.

The tab Requirements provides information of the associated project requirement, and the standard requirement that is being evaluated.

Conclusion

This deliverable, together with the prototype, finishes the Task 6.3 Specification of a Certification Metamodel for Energy Management Deployments. The last work performed in this task has consisted in the implementation of a Certification Framework that will support certification tasks against different standards or practices during the development of a project, product or component. The present document acts as a user manual of this prototype.

The definition and implementation of this prototype is the final result of all the previous work done in this task, besides contributions of other WPs (specifically T3.2 and T5.1). The analysis of standards allowed the identification of main issues and terms related to standards and best practices. This work followed in the second deliverable of the task that provided the first iteration of what would be the Certification metamodel. Deliverable 6.3-C made the first analysis that showed which process modelling notations and metamodels could be used to extend the prototype.

Finally, the last part of the work done in this task has been the finalization of the prototype, as well as the definition of the first iteration of the eDIANA iEi reference that has been created to illustrate the management of the prototype. This reference is an example of how different standards or best practices can be defined with the help of this prototype; and how it can be used to support the certification of projects that accomplish it. To complete it, it has been completely essential the interaction with WP3 and WP5, specifically with the work done in task 3.2 and 5.1. The first task has been responsible of the definition and implementation of the iEi, whereas the task 5.1 is responsible of the integration of the iEis.

Future work in this area will be the extension of this framework to cover the modelling of the business process and integrating it through the certification process. Unfortunately, this work will be performed beyond the scope of this project.

Acknowledgements

The eDIANA Consortium would like to acknowledge the financial support of the European Commission and National Public Authorities from Spain, Netherlands, Germany, Finland and Italy under the ARTEMIS Joint Technology Initiative.

References

- Arriola, Aitor; Ormazabal, Gotzon; Zubia, Cristina; Marcos, José Manuel; Atutxa, Txomin. Intelligeng Embedded Interface (iEi)-System Release.eDIANA Project 2010.
- Arana, Naia; Espinoza, Huascar; Tellado, Borja; Rodríguez, Francisco; Sánchez, Víctor; Uriarte, Amaia; Martínez, Luis; López, José Antonio. Certification Standards Analysis. eDIANA Project 2009.
- Arana, Naia; Etxeberria, Leire; Specifications of Certification Metamodel. eDIANA Project 2010.
- Benedicto, Jesús; Rosenberg, Igor; Soler, Ignacio; Arana, Naia; Espinoza, Huascar. Analysis of Standard Process Models. eDIANA Project 2010.