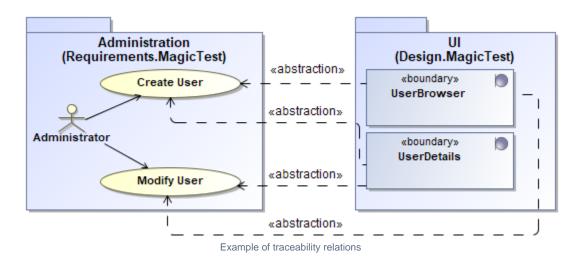
## Analyzing traceability relations

You can perform the impact and/or gap analysis in your project using the Dependency Matrix feature, which is a powerful way for representing traceability relations between multiple elements from different packages, levels of abstraction, views, or other relations that cannot be represented on diagrams, for example, relations through UML tags. You can create your own dependency matrices or even custom dependency matrix types for visualization of various traceability relations. You only need to define a relevant traceability property as **dependency criteria** for this.

The following figure depicts an example of traceability relations between several model elements.



These relations can be represented by the following traceability properties:

- Realizing relations (use cases classes):
  - Realizing Class
  - Realizing Element
  - All Realizing Elements
- Specifying relations (classes use cases):
  - Specifying Use Case
    - Specifying Element
    - All Specifying Elements

You can create a dependency matrix to visualize these traceability relations. The following steps will show you how to create a dependency matrix for the Realizing Class predefined property.

To create a dependency matrix for the Realizing Class predefined property:

1.	Create	a D	epend	ency	Matrix	diagram
----	--------	-----	-------	------	--------	---------

- 2. Define row and column type element types as follows:
  - a. For the Row Element Type, select UseCase
  - b. For the Column Element Type, select Class
- 3. Define row and column scopes as follows:
  - a. For the Row Scope, click **Administration**.
  - b. For the column scope, click **Design > MagicTest** and select **UI**.
- 4. Define the **Realizing Class** property as Dependency Criteria.
- 5. Rebuild the matrix (click *in the diagram toolbar*).

Row Element Type:	UseCase	 Column Element Type:	Class		
Row Scope:	Administration 849	 Column Scope:	U		{}*y
Dependency Criteria:	Realizing Class	 Direction: Both	$\sim$	Show Elements: All	~

Defining criteria for visualization of traceability relationships that can be represented by Realizing Class predefined property

The following figure depicts the created Dependency Matrix showing the traceability relationships between use cases and design classes they realize:

Legend		· 🛅	UI											
		ClassDetails	CourseBrowser	CourseDetails	InstructorProfile	Login	LoginDialog	QuestionDetails	😴 Student UI Navigati-⊞	StudentProfile	TestAssessmentWir-E	TestDetails	UserBrowser	UserDetails
🖃 💼 Administration		3	3	3	1					1			3	3
Create Class	1	7												
🔘 Create Course	2		7	7										
Create User	2												7	7
Modify Class	1	7												
Modify Course	2		7	7										
O Modify User	2												7	7
Remove Class	1	7												
Remove Course	2		7	7										
Remove User	2												7	7
🔿 View Instructor Profile	1				7									
🔿 View Student Profile										7				

Visualization of Realizing Class traceability relations on Dependency Matrix

## **Related Pages**

- Traceability

  - Creating traceability relations
    Traceability relations representation
    Navigating between different levels of abstraction
    Analyzing traceability relations
    Predefined traceability rules
    Custom traceability rules