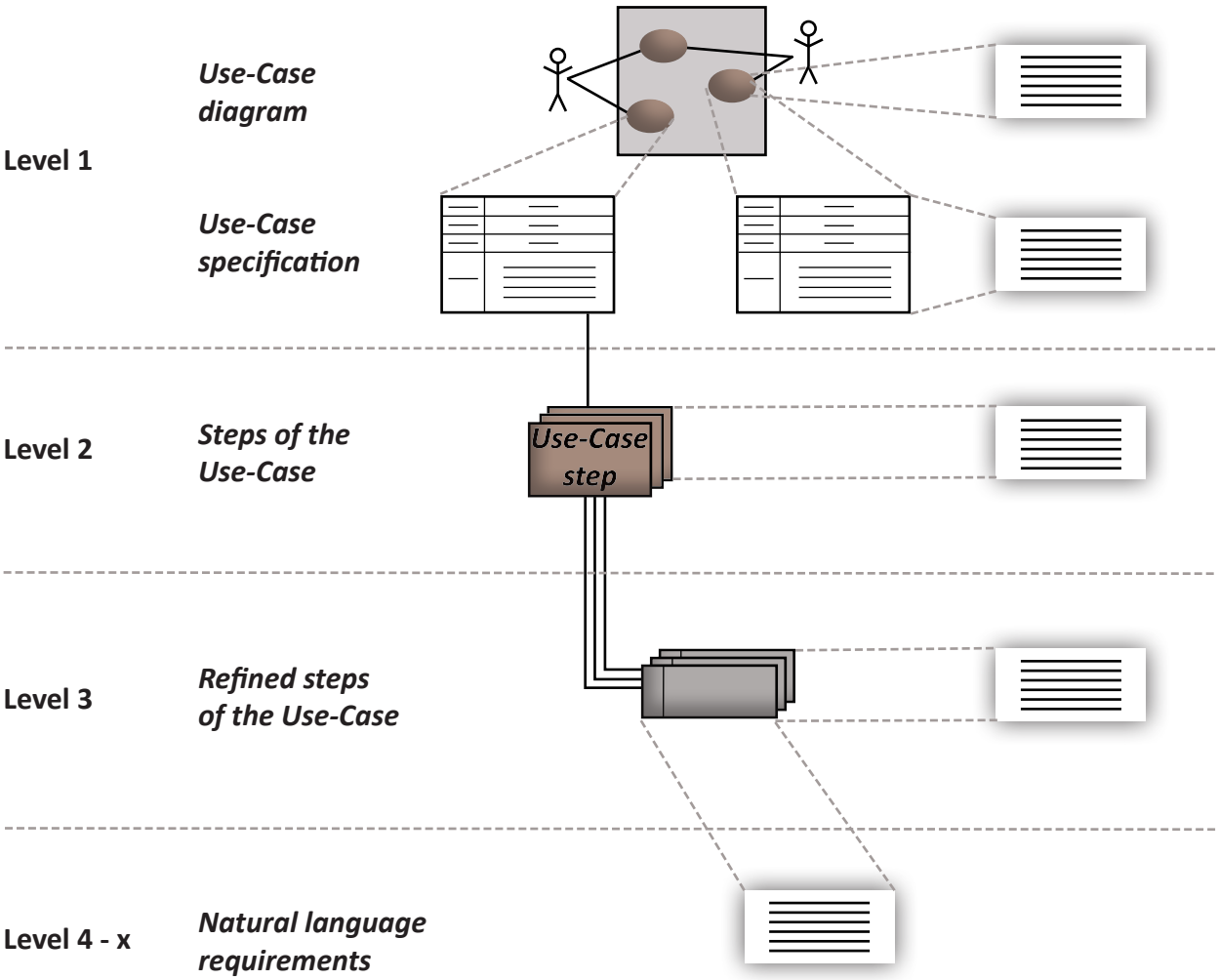
Die SOPHISTen

Handout Stable

1. Overview - An Examplary Illustration



STABLE is a systematic approach for structuring requirements specifications on the basis of Use-Case analysis. In this approach, the analyzed system (or partial system) is seperated in behavior-oriented elements, starting from the rough requirements and working towards the refined requirements.

Starting point is a collection of use cases, which describe the system from a user`s perspective. These Use-Cases are refined during the system analysis, resulting in more detailed Use-Cases, activities or states. Often, these refined artifacts are also documented with UML-diagrams.

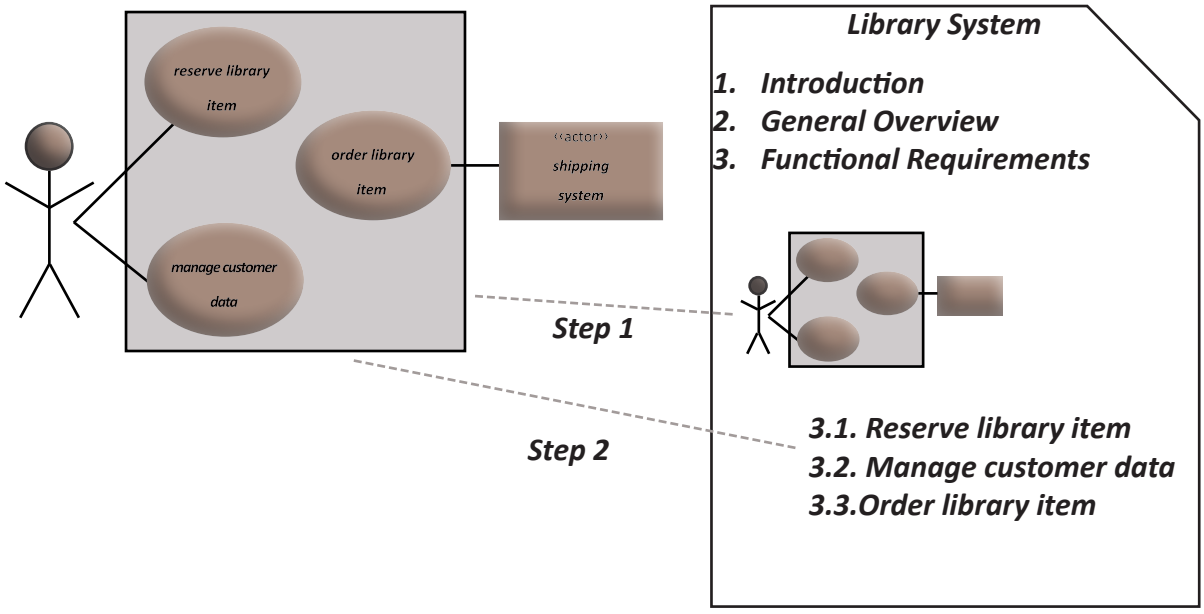
In a nutshell, the models and artifacts that are created during Use-Cases analysis are transfered 1:1 into a requirements document, e.g. a standardized document structure, according to the STABLE regulations. The structure of the models thus becomes a document structure for the natural-language requirements of the requirements specification.

2. Examplary implementation of STABLE (Level 1)

STABLE is an iterative approach that starts with the roughest requirements and can be applied for each level of refinement until the desired granuality for the document structure has been reached. In the following, Level 1 is described on the basis of Use-Case diagrams and accompanying Use-Case descriptions. It is also possible to use activity diagrams or state diagrams instead of use case descriptions, depending on documentation procedures in the project.

Use-Case Diagram

The system Use-Case diagram documents the essential functions of the system, i.e. the highest level of system functions and serves as an overview for the functional requirements of your system. For each Use-Case, a sub-chapter is created, where more detailed requirments for the respective part of the system functionality can be documented in.



Step 1:

Document the Use-Case diagram of the whole world system at the start of your chapter „functional requirements“

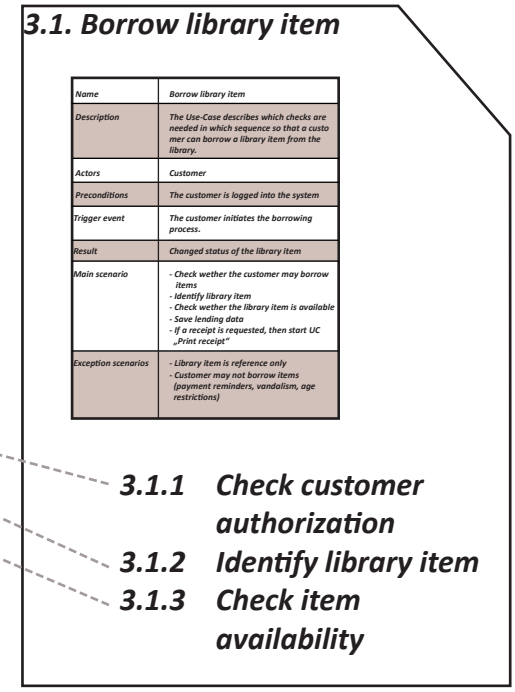
Step 2:

Create a sub-chapter „<name_of_use_case>“ for each Use-Case.

Use-Case Specifications

Use-Case specifications contain additional information about individual Use-Case, including a main scenario with the essential steps of the described system functionality. These steps become sub-headings for your chapter about the functionality of the respective Use-Case.

Name	*Borrow library item*
Description	*The Use-Case describes which checks are needed in which sequence so that a custo mer can borrow a library item from the library.*
Actors	*Customer*
Preconditions	*The customer is logged into the system*
Trigger event	*The customer initiates the borrowing process.*
Result	*Changed status of the library item*
Main scenario	*- Check wether the customer may borrow items* *- Identify library item* *- Check wether the library item is available* *- Save lending data* *- If a receipt is requested, then start UC „Print receipt“*
Exception scenarios	*- Library item is reference only* *- Customer may not borrow items (payment reminders, vandalism, age restrictions)*



Step 1:

Document the Use-Case specifications of your Use-Case at the start of your chapter „Detailed functional requirements“.

Step 2:

Create a sub-heading „<name_of_use_case_step>“ for each step of your Use-Case`s scenario.