

The Essence of Use Case Modeling

Dr. Christoph Steindl
Catalysts GmbH

Agenda

- Use Cases
 - What are Use Cases anyway?
 - How do we develop Use Cases?
 - What is the Power of Use Cases?
 - Why do we do Use Cases?

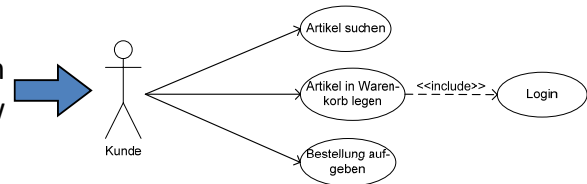
What are Use Cases anyway? Catalysts

- Use Cases define the functional requirements of a system from the user's point of view
 - Which actor does what with the system?
 - How shall the system respond to the actor's requests?
- Use Cases describe something that has value to the user
 - a step in some activity within a business process
 - he/she gets more money if he/she performs it more often
 - he/she is relieved having finished and can go for a coffee / cigarette afterwards
- Use Cases describe computer-aided tasks
 - At one place
 - At one time (1 to 15 minutes)
 - By one person
- They are mainly textual, not just UML Use Case diagrams
- They don't have to be perfect to be very valuable to the project.

Use Case ID and Name	UC-001 register new customer
System Context	Initial use case for the home shopping ordering system
Description	Requires a new customer for access to the home shopping ordering system. The main use case is that the customer can place any order in their own shop. Other orders which require the customer to be registered.
Preconditions	None
Success/ Failure	Customer registered
Priority and Dependencies	Customer of Our Service Catalogue
Main Scenario	1. System prompts for customer selection 2. Customer selects product 3. System determines that customer is within the delivery area for the home shopping ordering system and prompts for customer details 4. Customer supplies name, postal address, telephone number and email address if available 5. System stores the details, issues a unique reference and confirms successful registration 6. Use Case ends successfully
Alternatives	1a. Customer selection is outside delivery area 1a1. System informs customer that they are outside the delivery area, but records details anyway for management information purposes. 1a2. Use Case ends in failure (outside customer) 2a. Customer already registered 2a1. System informs customer that name and address are already registered. 2a2. Use Case ends in failure (already registered) 3a. Customer not located 3a1. System informs customer that registration has been rejected, but registration attempt recorded for all purposes. 3a2. Use Case ends in failure (not located)

← With all the necessary detail

Not just an overview →



How do we develop Use Cases? Catalysts

- Together with the customer in a facilitated workshop.
 1. Identify the actors and their goals → list of actors and list of use cases
 2. Describe the main scenarios that deliver the goals → sketch of use cases
 3. Identify possible failures → list of failures for the whole system and each use case
 4. Describe the alternative scenarios that handle the failures → complete use cases
- In a combination of group activities and individual activities
- Using a slim template (tailored to the needs of the project)
- In parallel to sketching the domain model and the user interface
- Breadth first (all actors and goals), then the details
 - Judiciously decide where to spend your energies
 - Areas of risk, areas of uncertainty
 - Areas of new, visionary functionality, areas of complex functionality
 - Experience steady progress, don't get lost in details, no analysis paralysis

What is the Power of Use Cases? Catalysts

- We **create them jointly with the customer** – in an **iterative and risk-driven** fashion.
- They describe the system's functionality **from the user's point of view**.
- **Customers and developers understand them**, they develop a **shared vision**.
- They focus on **user – system interactions** rather than just telling „what the system shall do“, they **describe the dynamics**.
- They focus on what has **value to the user**, so that we can **build the right system**. We can work on the most valuable functionality first and never do something not worth it.
- We **ask the right questions at the right time**.
- It's easy to get an overview of the system and to find the details.
- It's easy to see whether everything is described at the appropriate level of detail.

The Language of Use Cases Catalysts

- Use Cases suggest **very short sentences**: “subject predicate object”
 - The user
 - enters ...
 - selects ...
 - The system
 - validates the user's input
 - processes the user's input
 - prepares the (user interface for the) next step
- They ask for a **very simple language**.
- They **separate the sun-shine scenario** from the **complexities**.
 - No complications
 - No deviations