Use Cases

Prof. Dr. Robert Mertens

SoSe 2020

Introduction and Overview

Introduction

… oder: „What are Use Cases?“

- Stories (text)
  - System use
  - Fulfillment of user goals
- Use Case diagrams
  - Organization tool
  - Easy to learn
  - Text is more important
- Role of case cases
  - Important OOA/D
  - Important for subsequent project phases

„The indispensable first step to getting the things you want out of life: decide what you want.“

- Ben Stein

Motivation

… oder: „Why use Use Cases?“

- User centered
  - Who is using the system?
  - What are typical use scenarios?
  - What are the users‘ goals?
- easy
  - Can be understood by domain experts
  - Can be written by domain experts
- Scalable
  - Subject matter depth
  - Degree of formalism

„Uses cases integrate end users at an early stage of the development process. This is an important success factor in IT projects.“

Scenarios and Alternatives

… or: „How to write down a use case?“

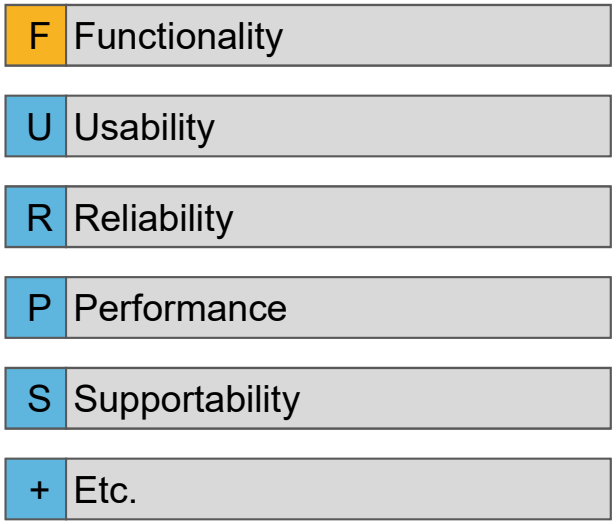
Main success scenario	Alternative I	Alternative II
		
▪ Player rolls dice ▪ Player moves token according to pips ▪ Token is on an empty lot ▪ User buys the lot	▪ User already owns the lot ▪ Users buys a house for the lot	▪ Lot is owned by an opponent ▪ User has to pay rentDas Grundstück gehört einem anderen Spieler. ▪ Der Spieler muss Miete zahlen

„A Secenario is a specific sequence of actions and interactions between the user and the system. Another name for scenarios is use case instances.“

Use Cases und Requirements

… or: „Aren‘t use cases requirements, too?“

- Use cases <u>are</u> requirements
  - But do not comprise all requirements
- Use cases are an important tool
  - To find requirments
  - To define them
- Use cases ↔ feature lists
  - Assigned to stakeholders
  - Easier to check benefits

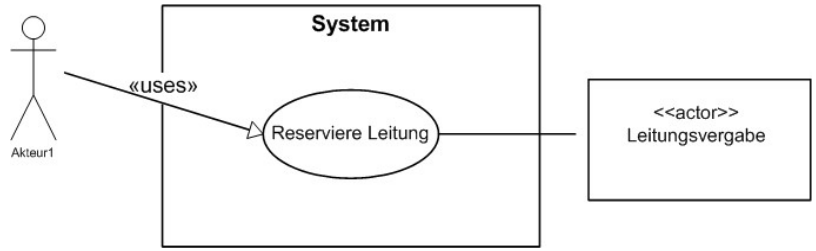


„Use Cases are mainly describing functional requrements.“

Actors

… or: „What roles are there in use cases?“

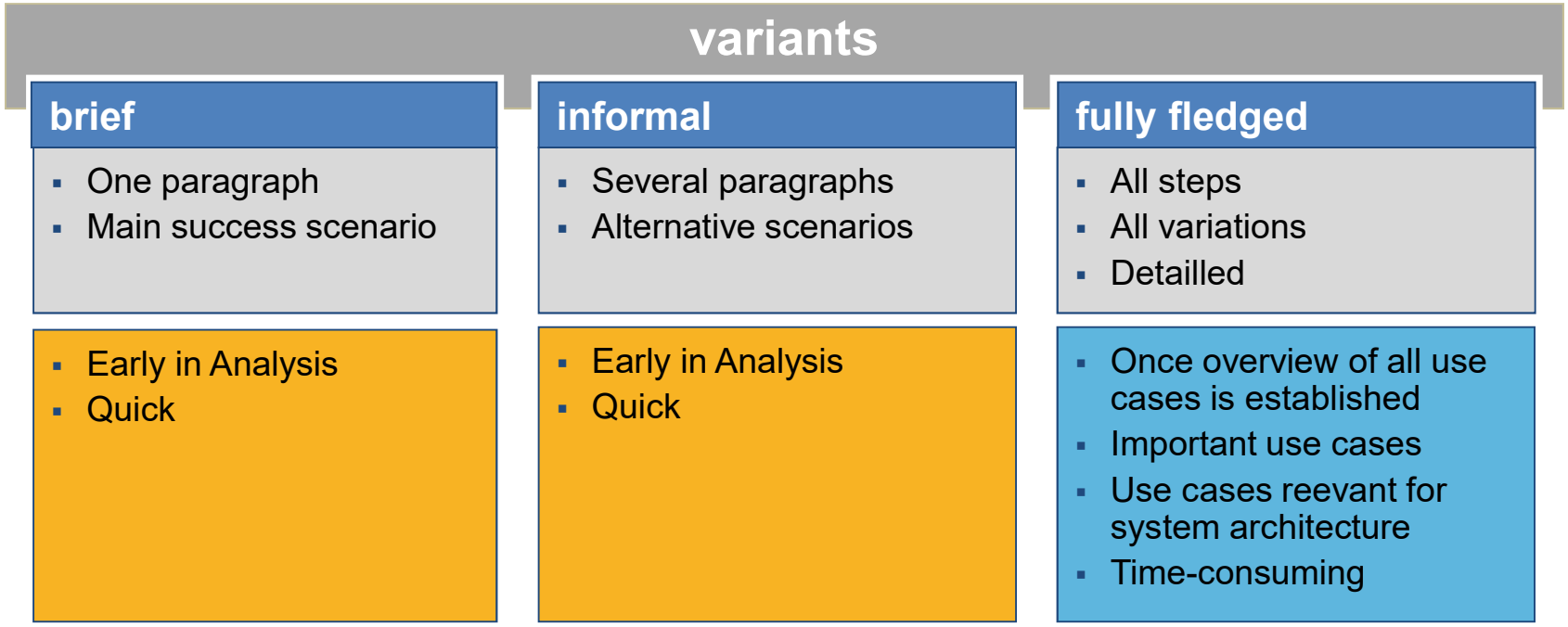
- primary actor
  - Use the system
  - Fulfill own goals (with the system)
  - Only one per use case
- Supporting actors
  - Provide services
  - Provide information
- Offstage Akteure
  - Have an interest in the use case
  - Are neither primary nor supporting
  - Example: IRS



„Everyone and every system except for the one beeing constructed is an actor. In other systems‘ use cases the system can be an actor. For instance when it calls those systems. “

Formats

… oder: „How long does it take to write a use case?“



Structure of a Use Case

Structure

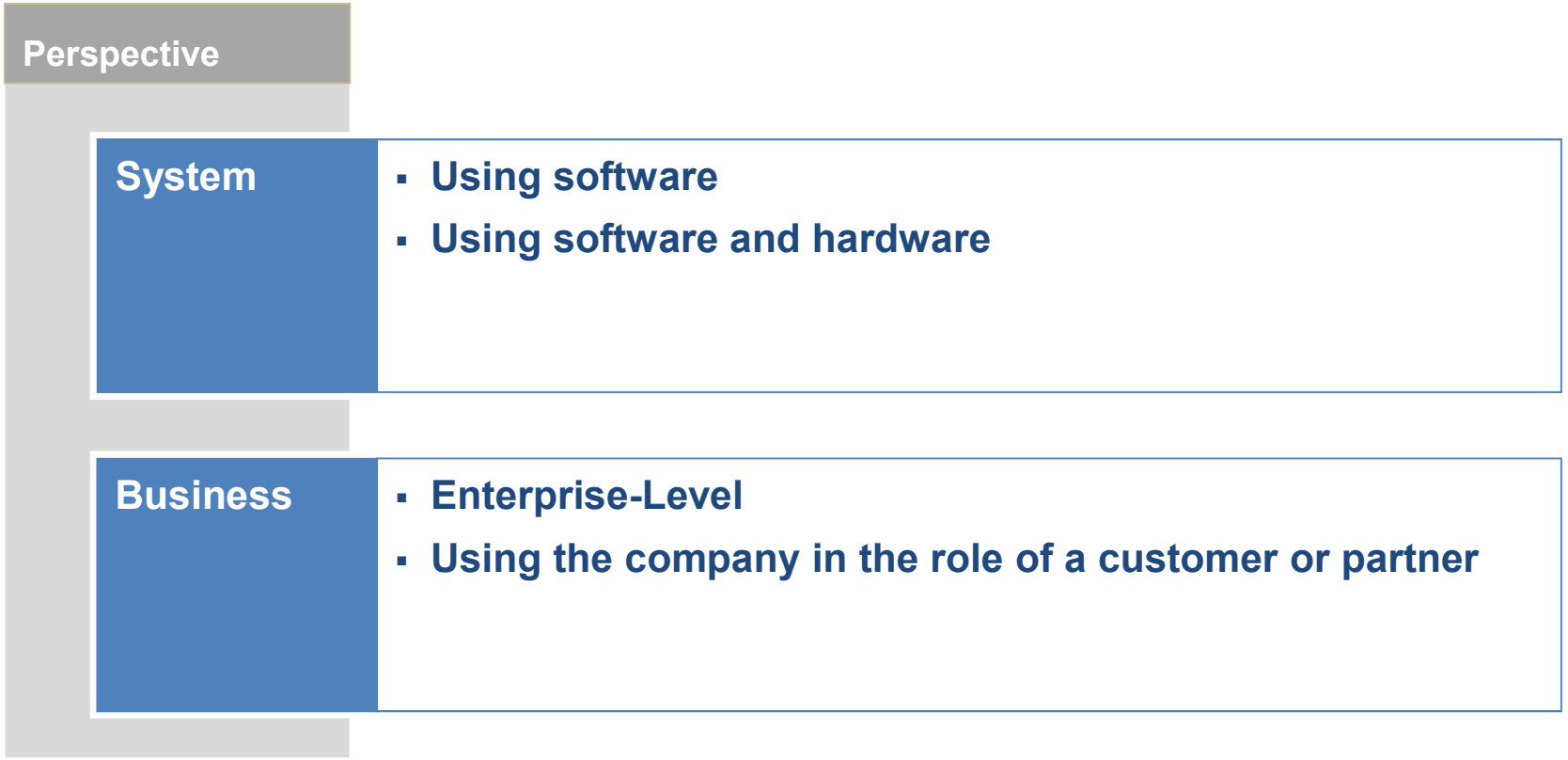
… oder: „A template.“

Name	Name, should start with a verb.
Scope	The system being developed.
Level	„User goal“ or „Sub-function.
Primary actor	Who is calling the use case??
Stakeholders and interests	Who has what interest in the use case?
Preconditions	Which (relevant) conditions must be met before starting the use case?
Postconditions	Which (relevant) conditions must be met after a successful run?
Main success scenario	Typical case leading to success, no branching and conditions.
extensions	Alternative paths to success or failure.
Special requirements	Non functional requirements in the context of the use case.
Technology and data variations	Variations in I/O or data formats.
Frequency of occurence	Influences analysis, test and time of implementation.
Open issues	Open issues.

„The orifginal can be found at alistair.cockburn.us“

Scope

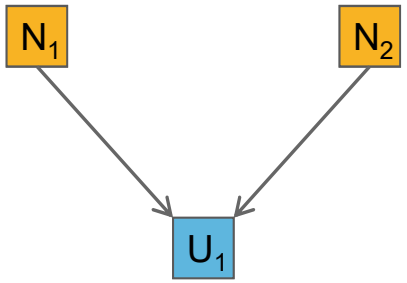
… or: „What to look at and what to ignore?“

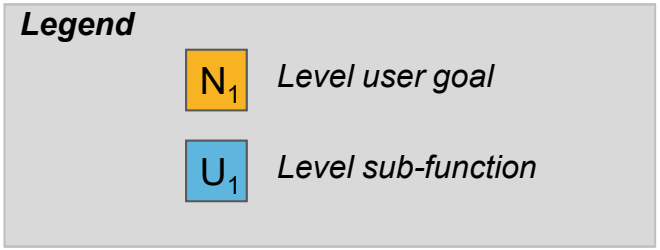


Level

… oder: „Who starts the use case?“

- Level user goal
  - Started by the user to fulfill a goal
- Level sub function
  - Common step im many use cases





Stakeholder and their interests

… or: „Why was the system build in the first place?“

- Stakeholder
  - Some are your customers
  - All are soe kind of actor
- Interests
  - Starting point for use cases
  - Stakeholder list for fining all interests

„Stakeholder influence the ways the system is build. Either directly through use cases or indirectly as offstage actors.“

Pre- and postconditions

… or: „Cause and effect“

- Preconditions
  - Necesarry, not sufficient
  - No noise (plugged in etc.)
- Post conditions
  - Met after a run
  - Might differ for extensions
- Stakeholder Interests
  - Are met in postconditions

„Fish diet 2020: An obvious difference!“

Main Success Scenario

… or: „Life as it should be.“

- Structure
  - No if-then-consrutcts
  - Repetitions are okay
- Step categories
  - Start
  - Interactions between actors
  - Validations/checks
  - State changes of the system

„Main success scenarios can be complux – but they describe the typical way to succeed. “

Extensions/Alternatives

… or: „What can go wrong and what to do then?“

- Describe alternatives
  - Other wasy to succeed
  - Errorhandling anf failures
  - Might be written in use case notations
- Extensions have two parts
  - Conditions (detectable by system)
  - Sequence of actions
- Stakeholder Interests
  - Functional: main success scenario and alternatives
  - Non-functional: in „special requirements“

1. Player rolls dice
2. Player moves token according to number of pips
3. Token is on a free lot
4. Player buys the lot

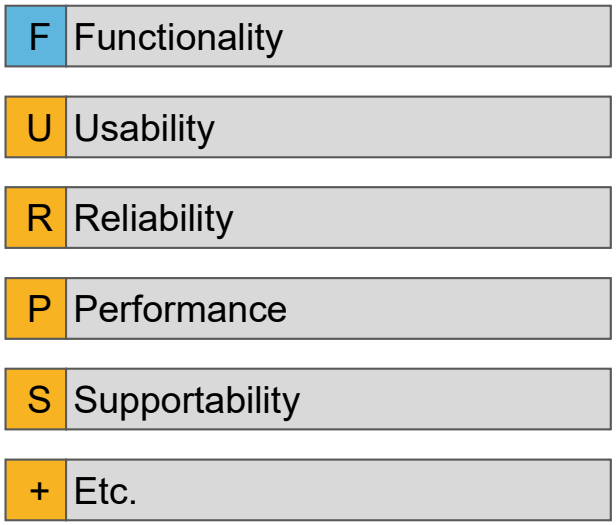
3a. Player already owns the lot:
1. Player buys a house for the lot

„Zur Kennzeichnung des Erweiterungspunktes wird die Schrittnummer durch einen Buchstaben erweitert. Die Nummerierung für die Schritte der Alternative beginnt wieder bei 1.“

Special Requirements

… or: „Did we forget something?“

- Gathered in use cases
  - But often with global effects
- Central gathering
  - Compiled in additional specificatons
  - After use cases are comüpleted

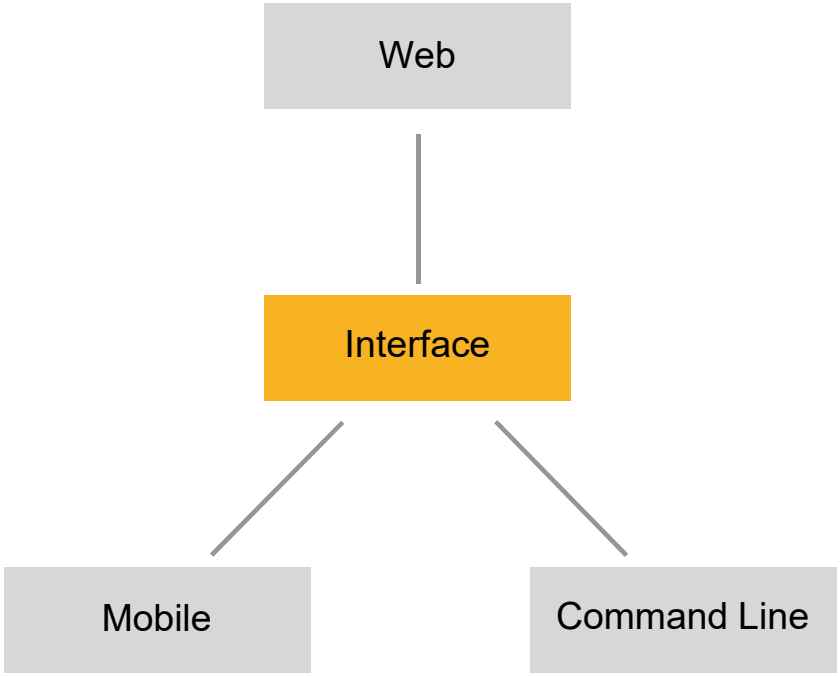


„Use cases are a good tool for finding non-functional requirements as well!“

Technology and Data Variations

… or: „Is there an App for that?“

- Differences in *how*, not in *what*
  - Should be avoides
  - Sometimes necessary
- Example: technical constraints
  - Input devices
  - Data processing rate
- Example: data formats
  - ISBN 10 and ISBN 13
  - SEPA account data

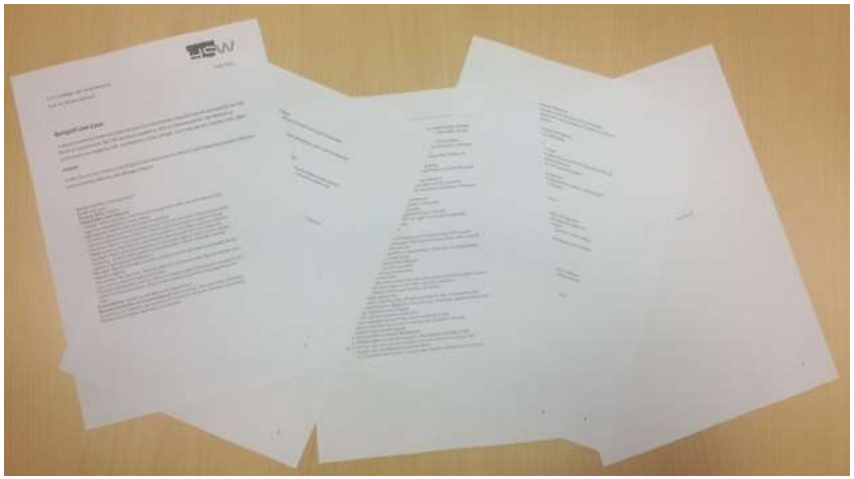


„One and the same use case can be implemented with differnt technologies!“

An Example

… or: „A complete use case description can get long!“

- Example in handout
- Task
  - Find actors
  - Assign categories to actos



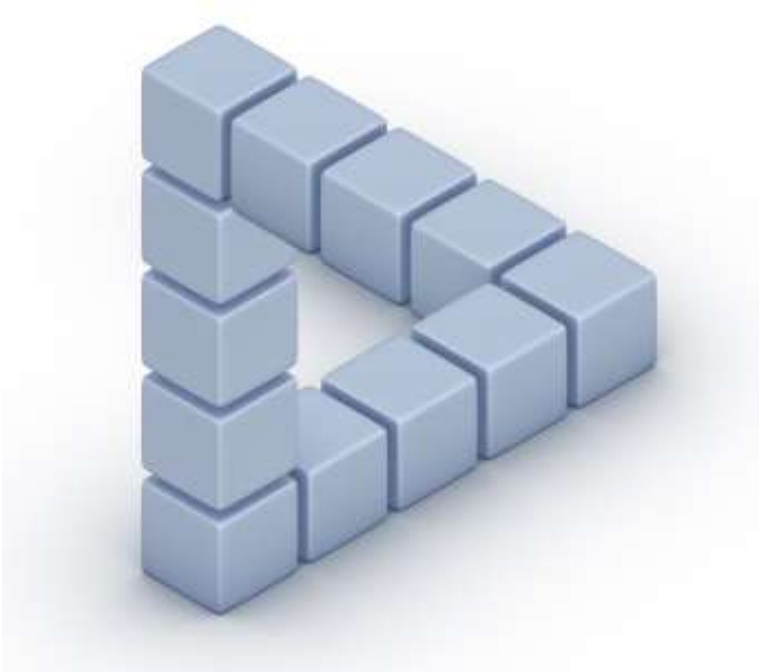
„A complete use case descriptions comprises a number of pages and tackles all relevant exceptions!“

Recommendations and Conventions

Use Case Done – But Wrong

… or: „Not everything that is written down is correct!“

- No Feedback
  - Conceptual errors?
  - Do useres really want the use case this way?
- Reality check needed
  - Runnable code
  - Testing, testing, testing!



„Only because it looks right does not mean ist is! “

Aternative Format: Two Colums

… or: „What happens where?“

Player	System
1. Player initiates turn.	
	2. System generates random number n.
	3. Player moves token forward n steps.
	4. System checks that field is empty and offers it to player.
5. Player buys lot.	
	6. System marks lot as belonging to player..

„The two-column-format is also called conversational format. Ist emphasis is on the inter-action between the actors and the system. It shows which step happens where. “

Recommendation: Ignore User Interfaces

… or: „What does the user really want?“

- Focus on intension, not actions
  - User has intentions
  - System haresponsibilities
- Abstracting from concrete actions
  - „enter login“ → „authenticate“
  - „enter ISBN “ → „select book“
- Root-Goal
  - Gives new perspectives
  - Leads to innovative solutions

„Specific user interfaces and input devices often lead to specific actions. Abstraction allows for finding the actual user goals.“

Recommentation: Keep It Short

… or: „dito.“

- Easier to read
- Gets read
- Easier to find problems

„Too much informations is harder to read and harder to keep up to date!“

Recommendation: Use Cases are Black Boxes

… or: „Look at the whole, not the parts!“

- Fokus on analysis
  - Not „How?“
  - But „What?“
- No design decisions
  - More degrees of freedom for use case
  - Opens new perspectives

„Don‘t keep busy with looking at the details. The solution is not devised bevor you know what it should be able to do.“

Recommendation: Look at Users and Goals

… or: „What is it all about?“

- Target criteria:
  - Measurable
  - Assigned to actors
- Actors as starting point

„A use case should not only realize a goal. That goal should also be perceived useful by the users. “

Recommendation: Four-Phases-Model

… or: „How to find use cases use cases?“

Define system boundaries	Identify primary actors	Goals of primary actors	Define use cases
• What is outside the system? • What is part of the system?	• Who starts and stops the system? • Is „time“ an actor (cron-jobs)? • Who administrates the systen? • Whom are errors communicated to? • Are updates push or pull? • Who reads log data? Internal or external? • Who monitors performance? • Does the system restart automaticallz? • Is the system called by external systems or services?		• Usually one use case per goal. • Should start with a verb.
• When boundaries are not clear, define external actors first	• Both phases are closely linked? • Identifying primary actors first can help (brainstorming).. • Alternative: Look at external event. E.g.: System gets updated: By whom? When?		• Tip: CRUD can aggregate all editing use cases. Names like manage_X are often used in these cases.

Testing Use Cases

… or: „Is this a good use case or not?“

- Tests are only guidelines
  - blurry
  - No knockout criterion
- Exceptions
  - Sub-functions
  - Preparatory steps (like login)
  - Etc.

Boss Test

Boss: *„What have you achived today?“*

Employee: *„I logged in.“*

Würde der Boss die Antwort gutheißen, besteht der Use Case Einloggen den Boss Test.

Elementary Business Process Test

„An elementary business process (EBP) is defined as a task performed by one person in one place at one time, in response to a business event, which adds measurable business value and leaves the data in a consistent state).“

C. Larman, Applying UML and Patterns

Size test

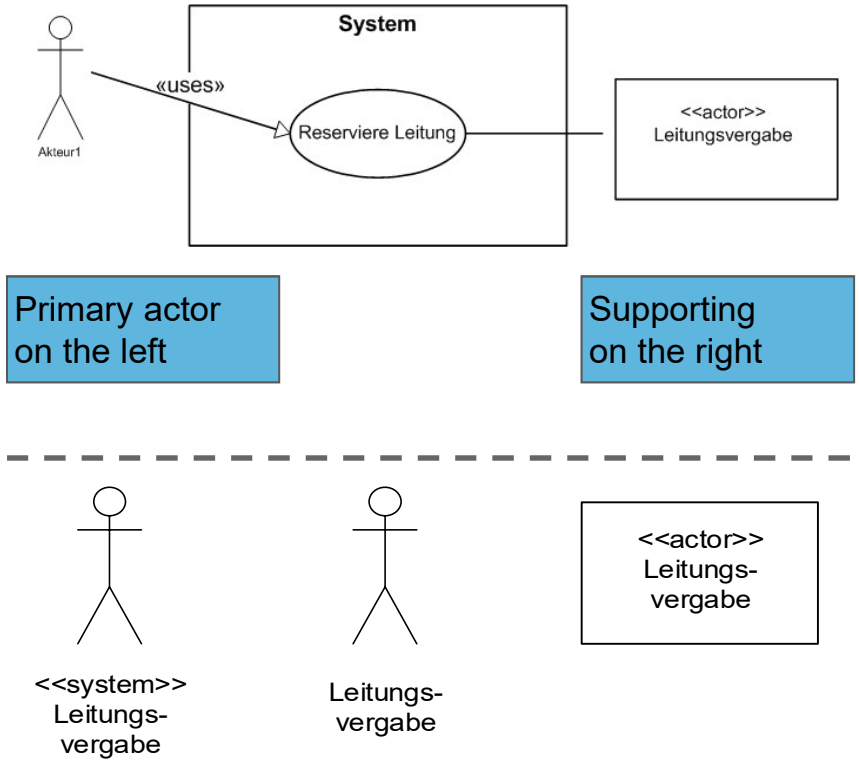
The uses case comprises a number of steps.

A fully fledged use case usually encompasses 3-10 pages.

Recommendtation: Text is King

… or: „The story matters most!“

- Diagrams are just support
  - Overview and orientations
  - Visualization of user goals
- Details and courses of actions
  - As text
  - Activity diagrams help, too
- Design conventions
  - Placement of actors
  - Visual differences possible



„UML comes with three different ways to visualize external actors.“


Thank you for your attention!