

Use cases



Version 2.3 – 20 October 2013




© Maurizio Morisio, Marco Torchiano, 2012



Attribution–NonCommercial–NoDerivs 2.5

- You are free: to copy, distribute, display, and perform the work

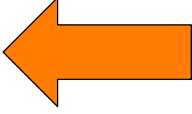
Under the following conditions:

-  **Attribution.** You must attribute the work in the manner specified by the author or licensor.
-  **Noncommercial.** You may not use this work for commercial purposes.
-  **No Derivative Works.** You may not alter, transform, or build upon this work.
 - For any reuse or distribution, you must make clear to others the license terms of this work.
 - Any of these conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

This is a human-readable summary of the Legal Code (the full license) found at the end of this document

Requirements Document

1. Purpose and scope
2. The terms used / Glossary
3. **The use cases** 
4. The technology to be used
5. Other various requirements
6. Human backup, legal, political, organizational issues

History

- Introduced by Ivar Jacobson in 1992
 - ◆ codified a visual modelin technique
- Found widespread adoption in the 1990's
 - ◆ initially in the Object-Oriented community

Use case

- Captures a contract between the stakeholders of a system about its behavior.
- Describes the system's behavior under various conditions as it responds to a request
 - ♦ from a stakeholder, the *primary actor*.
- The primary actor initiates an interaction with the system to accomplish some goal.
- The system responds, protecting the interests of all the stakeholders.

Purpose

- Describes business' work process,
- Focus discussion about up coming software system requirements,
 - ♦ not a requirements description substitute
- Functional requirements for a system
- Document the design of the system

Writing use cases

- Might be written
 - ♦ in a small, close-knit group, or
 - ♦ in a formal setting, or
 - ♦ in a large or distributed group
- Each situation lends to a different writing style suitable for the purpose
 - ♦ Fully dressed
 - ♦ Casual

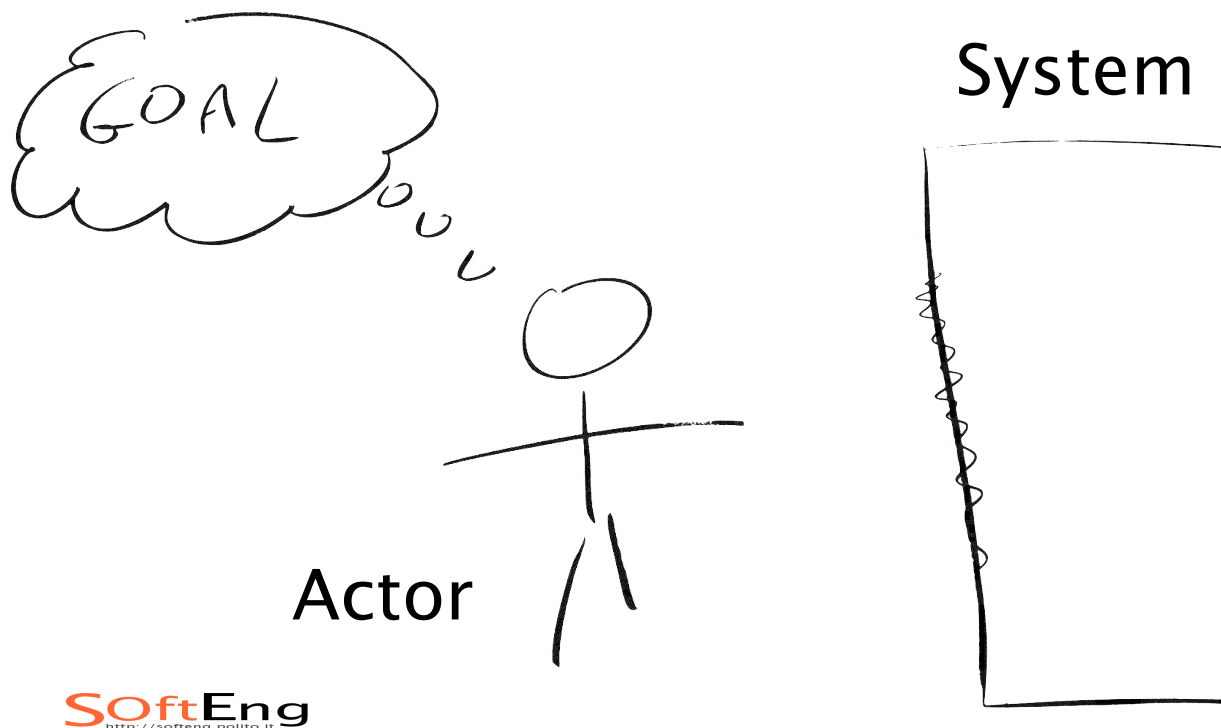
Requirements and Use Cases

- They can represent the functional (behavioral) requirements
 - ♦ If properly written
- They miss
 - ♦ External interfaces
 - ♦ Data formats
 - ♦ Business rules

Definition

A use case describes a **goal-oriented** set of **interactions** between external actors and the system under consideration.

Key elements



Key elements

- The **actor** involved
 - ♦ type of user that interacts with the system
- The **system** being used
 - ♦ treated as a black-box
- The functional **goal** that the actor achieves using the system
 - ♦ the reason for using the system

Description

- Use cases describe the sequence of interactions between actors and the system
 - ♦ to deliver the service that satisfies the goal
- They also include possible variants of this sequence
 - ♦ either successful or failing

Actor

- Actors are parties **outside the system** that interact with the system.
- An actor may be a class of users, roles users can play, or other systems.
 - ♦ A **primary actor** is one having a goal requiring the assistance of the system.
 - ♦ A **secondary actor** is one from which the system needs assistance.

Actor

- External entities:
 - ♦ Which persons interact with the system (directly or indirectly)?
 - Don't forget maintenance staff!
 - ♦ Will the system need to interact with other systems or existing legacy systems?
 - ♦ Are there any other hardware or software devices that interact with the system?
 - ♦ Are there any reporting interfaces or system administrative interfaces?

Stakeholders

- Someone having a vested interest in the behavior of the system (use case)
 - ♦ E.g. owner, board of directors
 - ♦ Offstage actors

Goals

- The use case cares only what is the relationship of the actor to the system
- The goal must be of value to the (primary) actor:
 - ♦ “Enter PIN code” is not
 - ♦ “Withdraw cash” is

Stakeholders and interests

- The system under design operates a contract between stakeholders
 - ♦ Use cases detail the behavioral part
- Use case captures all and only the behaviors related to satisfy the stakeholders' interests
 - ♦ See Guarantees section

Scope

- The extent of what we consider to be designed by us
 - ♦ Not already existing
 - ♦ Not someone else's design job
- Defined by
 - ♦ In-out list
 - ♦ Actor-Goal list
 - ♦ Use case briefs

Scope

- Enterprise
 - ♦ Business use cases
- Software System
 - ♦ System use cases
- Software Component

In-out list

- What lies in or out of scope

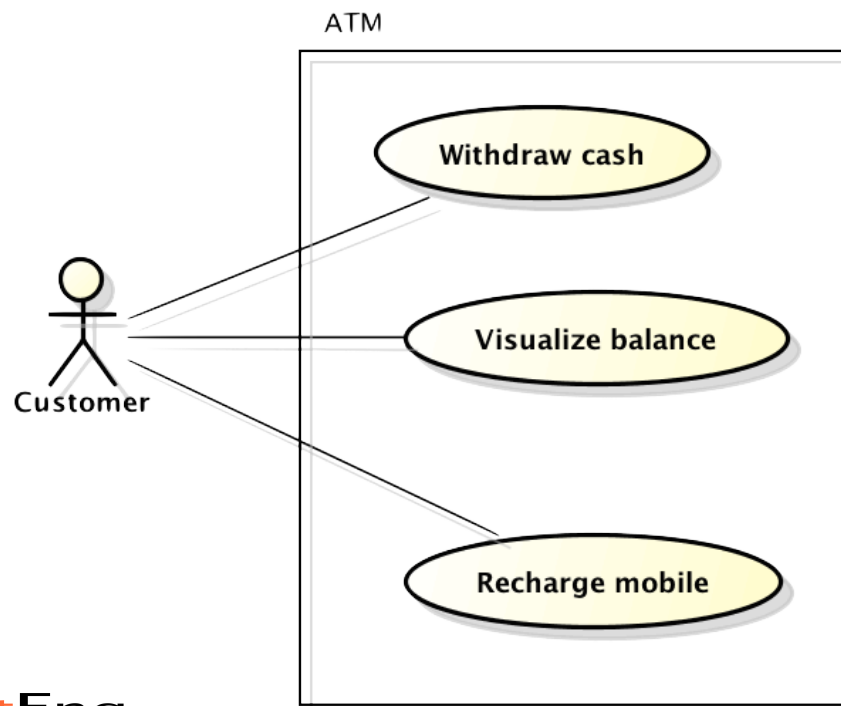
Topic	In	Out
Invoicing in any form		Out
Producing reports about requests, e.g. by vendor, by part, by person	In	
Merging requests to one PO	In	
Partial deliveries, late deliveries, wrong deliveries	In	
All new system services, software	In	
Any non-software parts of the system		Out
Identification of any pre-existing software that can be used	In	
Requisitions	In	

Actor-goal list

- Actors and corresponding goals + Priority
 - ♦ See UML Use Case Diagrams

Actor	Task-level Goal	Priority
Any	Check on requests	1
Authorizer	Change authorizations	2
Buyer	Change vendor contacts	3
Requestor	Initiate an request	1
"	Change a request	1
"	Cancel a request	4
"	Mark request delivered	4
"	Refuse delivered goods	4
Approver	Complete request for submission	2
Buyer	Complete request for ordering	1
"	Initiate PO with vendor	1
"	Alert of non-delivery	4
Authorizer	Validate Approver's signature	3
Receiver	Register delivery	1

Actor-goals as UC Diagram



Use case briefs

- Summary consisting of 2–6 sentences
 - ♦ What is going on
 - ♦ Most significant activities

Actor	Goal	Brief
Production Staff	Modify the administrative area lattice	Production staff add admin area metadata (administrative hierarchy, currency, language code, street types, etc.) to reference database and contact info for source data is cataloged. This is a special case of updating reference data.
Production Staff	Prepare digital cartographic source data	Production staff convert external digital data to a standard format, validate and correct it in preparation for merging with an operational database. The data is catalogued and stored in a digital source library.
Production & Field staff	Commit update transactions of a shared checkout to an operational database	Staff apply accumulated update transactions to an operational database. Non-conflicting transactions committed to operational database. Application context synchronized with operational database. Committed transactions cleared from application context. Leaves operational database consistent, with conflicting transactions available for manual/interactive resolution.



Essential use cases

- Intended to capture the essence of problems through technology-free, idealized, and abstract description
- Independent of the technical implementation of the system.
 - ♦ Not “the user presses the enter button”
 - ♦ Better “the user confirms his choice”

Conventional use case

User action	System response
insert card	read magnetic strip request PIN
enter PIN	verify PIN display transaction option menu
press key	display account menu
press key	prompt for amount
enter amount	display amount
press key	return card
take card	dispense cash
take cash	

25

Essential use case

User action	System response
identify self	verify identity offer choices
choose	dispense cash
take cash	

Precision

- How much detail is provided
- Brief
 - ◆ few sentences summarizing the use case
- Casual
 - ◆ few paragraphs of text elaborating the use case in the form of a summary or story
- Detailed
 - ◆ formal document based on a long template with fields for various sections

Precision stages

1. Actors and Goals
 - ◆ Prioritized list of actors and their goals
2. Main success scenario
 - ◆ Trigger and main success scenario
 - ◆ Check meeting stakeholders interests
3. Failure conditions
 - ◆ List of all possible failures
4. Failure handling
 - ◆ How to respond to each failure
 - ◆ Potentially revealing

Granularity

- **Summary level** is the 50,000 feet perspective,
- **User-goal level** is the sea-level perspective,
- **Subfunction** is the underwater perspective.

Summary level

- Summary level use cases:
 - ◆ are large grain use cases that encompass multiple lower-level use cases; they provide the context (lifecycle) for those lower-level use cases.
 - ◆ they can act as a table of contents for user goal level use cases.

Summary level

- Use Case:
 - ♦ Manage Funds By Bank Account
- Scope:
 - ♦ Bank Accounts and Transactions System
- Intention in Context:
 - ♦ The intention of the Client is to manage his/her funds by way of a bank account. Clients do not interact with the System directly; instead all interactions go through either: a Teller, a Web Client, or an ATM, which one depends also on the service.
- Primary Actor: Client
- Main Success Scenario:
 - ♦ 1. Client opens an account.
 - ♦ Step 2 can be repeated according to the intent of the Client
 - ♦ 2. Client performs task on account.
 - ♦ 3. Client closes his/her account

User Goal Level

- User-goal level use cases:
 - ♦ describe the goal that a primary actor or user has in trying to get work done or in using the system.
 - ♦ *are usually done by one person, in one place, at one time; the (primary) actor can normally go away happy as soon as this goal is completed.*

User Goal Level

- Use Case: Deposit Money
- Scope: Bank Accounts and Transactions System
- Intention in Context:
 - ♦ The intention of the Client is to deposit money on an account. Clients do not interact with the System directly; instead, for this use case, a client interacts via a Teller. Many Clients may be performing transactions and queries at any one time.
- Primary Actor: Client
- Main Success Scenario:
 - ♦ 1. Client requests Teller to deposit money on an account, providing sum of money.
 - ♦ 2. Teller requests System to perform a deposit, providing deposit transaction details*.
 - ♦ 3. System validates the deposit, credits account for the amount, records details of the transaction, and informs Teller.

User Goal Level

- Extensions:
 - ♦ 2a. Client requests Teller to cancel deposit: use case ends in failure.
 - ♦ 3a. System ascertains that it was given incorrect information:
 - 3a.1. System informs Teller; use case continues at step 2.
 - ♦ 3b. System ascertains that it was given insufficient information to perform deposit:
 - 3b.1. System informs Teller; use case continues at step 2.
 - ♦ 3c. System is not capable of depositing (e.g. transaction monitor of System is down)**:
 - 3c.1. System informs Teller; use case ends in failure.
- Notes:
 - * a hyperlink to a document that contains data details and formats.
 - ** this is an example of an IT infrastructure failure, we only write it in a use case if there is a corresponding project constraint that states a physical separation, e.g., transaction supported by a legacy system.

Subfunction level

- Subfunction level use cases
 - ♦ provide “execution support” for user–goal level use cases; they are low–level and need to be justified, either for reasons of reuse or necessary detail

Subfunction level

- Use Case: Identify Client
- Scope: Automatic Teller Machine (ATM for short)
- Intention in Context:
 - ♦ The intention of the Client is to identify him/herself to the System. A project (operational) constraint states that identification is made with a card and a personal identification number (PIN).
- Primary Actor: Client
- Main Success Scenario:
 - ♦ 1. Client provides Card Reader with card; Card Reader informs System of card details*.
 - ♦ 2. System validates card type.
 - ♦ 3. Client provides PIN to System.
 - ♦ 4. System requests BAT System to verify identification information*.
 - ♦ 5. BAT System informs System that identification information is valid, and System informs Client.

Right level

- Too-much levels
 - ♦ Very summary, cloud level
 - ♦ Sub-subfunctions (black)
- Tendency to be too specific
 - ♦ What does the primary actor really want?
 - ♦ Why is this actor doing this?

Exercise

"Jenny is standing in front of her bank's ATM. It is dark. She has entered her PIN, and is looking for the 'Enter' button."

- Name a summary, a user and a subfunction goal for Jenny.

Answers

- ◆ Very Summary: Take some special one out for dinner.

- Summary: Use the ATM
- User goal: Get money from the ATM.
- Subfunction: Enter PIN
- - ◆ Sub-Subfunction: Find the Enter button.

Template

- SWEED template
 - ◆ One column (no table)
 - ◆ Sequenced: Numbered steps (Dewey decimal numbering system) and extensions to main scenario use alphabetic letters to differentiate from main steps

Template

- Use Case:
 - ♦ Name of the use case. This is the goal stated by a short active verb phrase.
- Scope:
 - ♦ The scope of the “system” being considered (black-/white- box and enterprise/system/component).
- Level:
 - ♦ Summary, User-goal, or Subfunction

Template

- Intention in Context:
 - ♦ A statement of the primary actors intention and the context within which it is performed.
- Primary Actor:
 - ♦ The primary actor of the use case.
- *Stakeholders' Interests:
 - ♦ The list of stakeholders and their key interests in the use case.
- *Precondition:
 - ♦ What we can assume about the current state of the system and environment.

Template

- *Minimum Guarantees:
 - ◆ How the interests of the stakeholders are protected in all circumstances.
- *Success Guarantees:
 - ◆ The state of the system and environment if the goal of the primary actor succeeds.
- *Trigger:
 - ◆ What event starts the use case.

Template

- Main Success Scenario:
 - ◆ `<step_number> "." <action_description>`
The numbered steps of the scenario, from trigger to goal delivery, followed by any clean-up.
 - ◆ Conditions and alternatives are shown in the extension part.
- *Extensions:
 - ◆ `<step_altered> <condition> ":" <action_description> or <sub-use_case>`

Template

- Steps can be nested. Dewey numbers are then used, e.g. 3a.1
- An extension always refers to a step of the Main Success Scenario.
- An extension step takes place in addition to the respective main step, notation: 2 ||,
- or as an alternative, notation: 2a.
- An extension might correspond to regular behavior, exceptional behavior that is recoverable, or unrecoverable erroneous behavior.

Main Success Scenario: Steps

- An interaction between two entities
 - ♦ Customer enters address
- A validation step to protect an interest of a stakeholder
 - ♦ System validate PIN code
- An internal change to satisfy a stakeholder
 - ♦ System deducts amount from balance

Actions steps

- Use a simple grammar
 - ♦ *<Subject> <verb> <direct object>*
<prepositional phrase>
- Show clearly “who has the ball”
- Written from a bird’s eye pov
- Shows the process progressing
- Shows intent not movement
- Contains a reasonable set of actions
- Doesn’t “check whether”, “validates”

Action steps idioms

- Timing
 - ♦ “At any time...” or
 - ♦ “As soon as..”
- Trigger interaction with other system
 - ♦ “User has System A kick System B”
- Repeat
 - ♦ “Do steps x-y until condition”
- Parallel
 - ♦ “Steps x-y can happen in any order”

Exercise

Mary, taking her two daughters to the day care on the way to work, drives up to the ATM, runs her card across the card reader, enters her PIN code, selects FAST CASH, and enters \$35 as the amount. The ATM issues a \$20 and three \$5 bills, plus a receipt showing her account balance after the \$35 is debited. The ATM resets its screens after each transaction with FAST CASH, so that Mary can drive away and not worry that the next driver will have access to her account.

- Write the main success scenario for the task-level use case "Withdraw money using FASTCASH option"

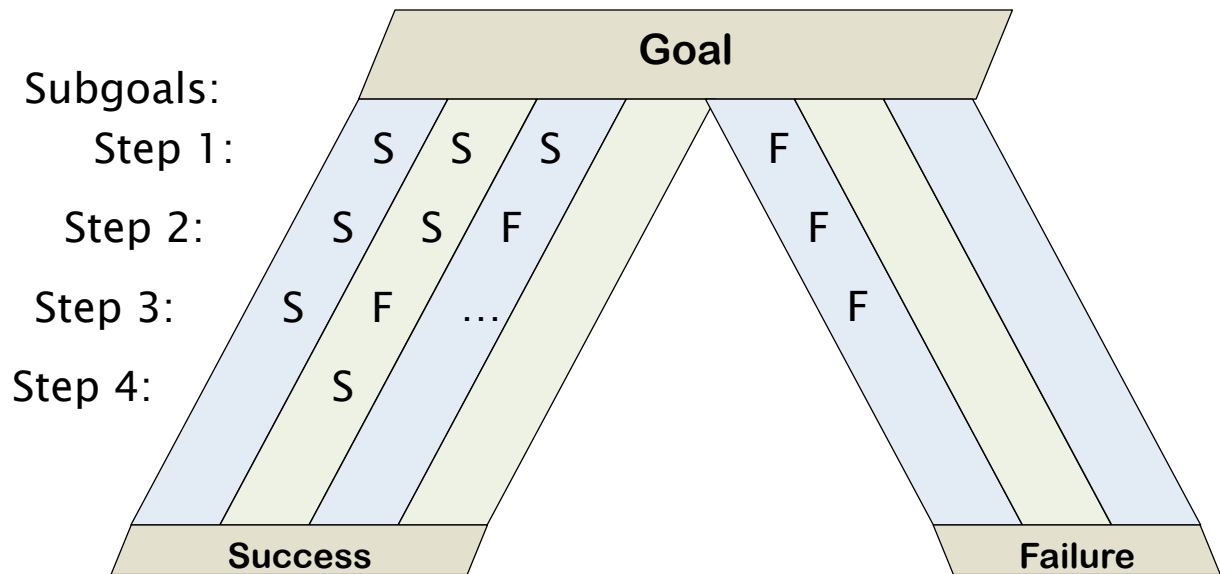
SoftEng
<http://softeng.polito.it>

Answer

1. Customer runs ATM card through the card reader.
2. ATM reads the bank id and account number from the card, validates them with the main computer.
3. Customer enters PIN.
4. ATM validates PIN.
5. Customer selects FASTCASH and withdrawal amount, a multiple of \$5.
6. ATM notifies main banking system of customer account, amount being withdrawn, and receives back acknowledgement plus the new balance.
7. ATM delivers the cash, card and a receipt showing the new balance.
8. ATM logs the transaction.

SoftEng
<http://softeng.polito.it>

Alternatives



Extensions

- Brainstorm and include every possibility
 - ◆ Alternate success path
 - ◆ Primary actor behaves incorrectly
 - ◆ Inaction by the primary actor
 - ◆ Negative outcome of validation
 - ◆ Internal failure (expectable)
 - ◆ Unexpected or abnormal failure
 - ◆ Critical performance failure

Extensions

- The condition expresses what was detected
 - Not what happened
 - “PIN entry time-out” **OK**
 - “Customers forgets PIN” **Wrong**
- The system must be able to detect the condition
- Merge equivalent conditions

Exercise

- Brainstorm and list the things that could go wrong during the operation of an ATM.

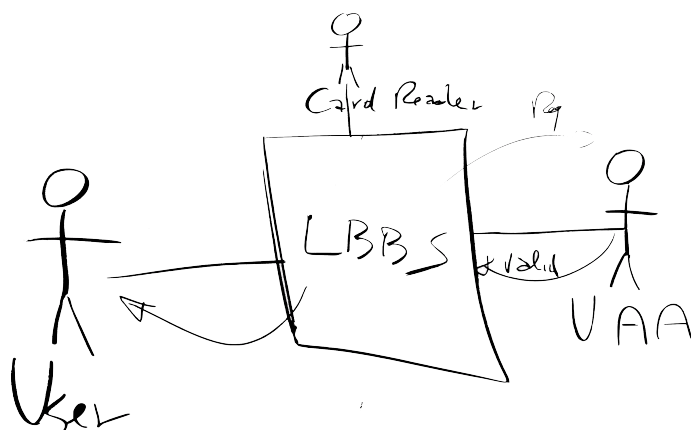
Answer

- Card reader broken or card scratched
- Card for an ineligible bank
- Incorrect PIN
- Customer does not enter PIN in time
- ATM is down
- Host computer is down, or network is down
- Insufficient money in account
- Customer does not enter amount in time
- Not a multiple of \$5
- Amount requested is too large
- Network or host goes down during transaction
- Insufficient cash in dispenser
- Cash gets jammed during dispensing
- Receipt paper runs out, or gets jammed
- Customer does not take the money from the dispenser

SoftEng
http://softeng.polito.it

Example: LBBS

- An automated library book borrowing system (LBB system). The goal is to relieve the librarians from processing book loans.



SoftEng
http://softeng.polito.it

Example: Check out books

- **Scope:** Library Book Borrowing System
- **Level:** User Goal
- **Intention in Context:**
 - ◆ The intention of the User is to check-out books from a library. Only one User can check-out books at any one time.
- **Primary Actor:**
 - ◆ User (becomes Member once s/he has identified him/herself with the System)

Example: Check out books

- **Main Success Scenario:**
 1. User requests System to check-out books.
 2. User identifies him/herself to System.
 - Step 3 is repeated for each book that is checked-out by Member.*
 3. Member registers book with System.
 4. Member indicates to System that s/he has finished checking out books.
 5. System records all books registered by Member as on loan, requests Printer to print out a receipt* for the session, and puts itself in a state to receive the next User.

Example: Check out books

- Extensions:
 - 2a. User fails to identifies him/herself with System: use case ends in failures.
 - 3a. System informs Member that s/he has reached his/her maximum number of books allowed on loan; use case continues at step 4.
 - 3||a. Member requests System to remove book from the books that are checked out:
 - 3||a.1. Member identifies book to System.
 - 3||a.2a. System identifies book and removes it from the list of books registered by Member; use case continues from where it was interrupted.
 - 3||a.2b. System fails to identify book; use case continues from where it was interrupted.

Example: Check out books

- (3-4)a. Member requests System to cancel check-out.
- (3-4)a.1. System removes all books that were registered by Member, and puts itself in a state to receive the next User; use case ends in failure.
- (3-4)b. System times-out waiting for input from Member:
 - (3-4)b.1. System removes all books that were registered by Member, and puts itself in a state to receive the next User; use case ends in failure.

Structure – inclusion

- An include relationship between two use cases means that the sequence of behavior described in the included (or sub) use case is included in the sequence of the base (including) use case.
 - ♦ Including a use case is thus analogous to the notion of calling a subroutine

Structure – extension

- The extends relationship provides a way of capturing a variant to a use case.
 - ♦ Extensions are not true use cases but changes to steps in an existing use case.
- The extends relationship includes
 - ♦ the condition that must be satisfied if the extension is to take place, and
 - ♦ references to the extension points defined in the extended use case

Applying use cases

- Define the scope
 - ♦ What's in and what's out
- Identify your actors:
 - ♦ who will be using the system?
- Identify their goals:
 - ♦ what will they be using the system to do?
- Write a use case brief
 - ♦ 2-6 sentences
- Detail the steps in main success scenario
 - ♦ 3-9 steps
- Identify failure conditions
 - ♦ Which conditions the system can detect?
- Define failure handling

Warnings

- Use case alone are not the requirements
- Use cases should be based on some form of conceptual model or glossary
- Use cases should not include details about the user interface

Typical errors

- Undefined or inconsistent system boundary
- Take System's viewpoint instead of Actor's
- Inconsistent Actor names
- Use cases refer to too many actors
 - ♦ Spider's web

Typical errors

- Too long specifications
- Confused specifications
 - ♦ Use of conditional logic
 - ♦ Attempt to describe algorithms
- Actor non fully entitled
- Customer not understanding
- Never ending use cases
- Dialog descriptions
 - ♦ Long, over-constrained, brittle

USE CASE DIAGRAMS

Use case diagrams

- Use cases can be (collectively) represented using UML use-case diagrams
- Often the “brief” level of detail can be useful in this context

Elements of use case diagrams



Actor

- Someone (user) or something (external system, hardware) that
 - ♦ Exchanges information with the system
 - ♦ Supplies input to the system, or receives output from the system



Use Case

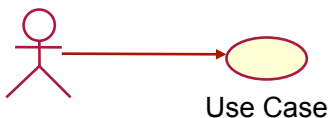
- A functional unit (functionality) part of the system

Relations



Actor

Use Case



Actor

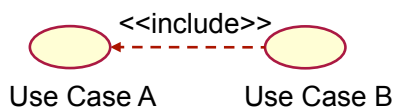
Use Case



Actor

Use Case

- Association models:
 - ♦ Which actors participate in a use case
 - ♦ Where execution starts
 - ♦ Adornments (e.g. multiplicity, direction) allowed

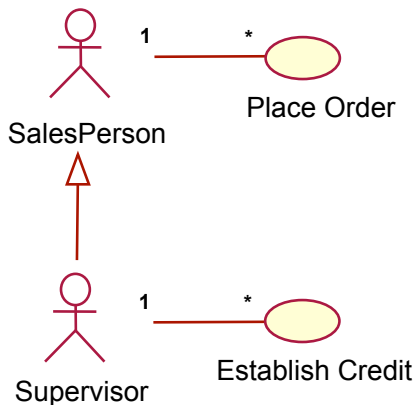
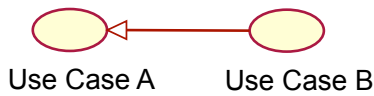


Use Case A

Use Case B

- Include
 - ♦ Models that functionality A is used in the context of functionality B (one is a phase of the other)

Relations



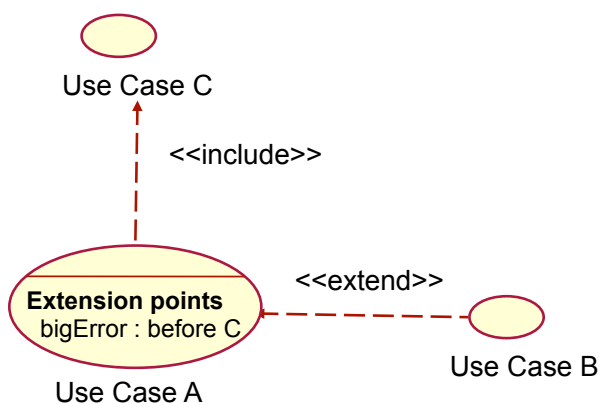
Generalization

- ◆ Defines functionality B as a specialization of functionality A (e.g. a special case)

Generalization

- ◆ A generalization from an actor B to an actor A indicates that an instance of B can communicate with the same kinds of use-case instances as an instance of A

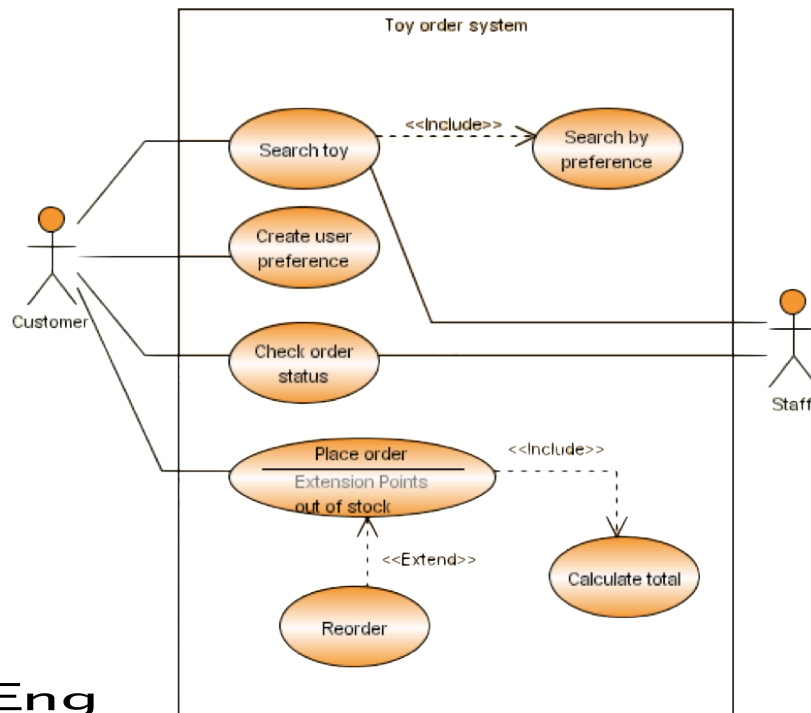
Relations



Extension

- ◆ An extend relationship from use case A to use case B indicates that an instance of use case B may be augmented by the behavior specified by A
- ◆ The behavior is inserted at the location defined by the extension point (name : where) in B, which is referenced by the extend relationship

Use case – Example



SUMMARY

Pros

- Use case modeling is generally regarded as an excellent technique for capturing the functional requirements of a system.
- Use cases discourage premature design
- Use cases are traceable.
- Use cases can serve as the basis for the estimating, scheduling, and validating effort.

Pros

- Use cases are reusable within a project.
 - ♦ The use case can evolve at each iteration from a method of capturing requirements, to development guidelines to programmers, to a test case and finally into user documentation.
- Use case alternative paths capture additional behaviour that can improve system robustness.

Pros

- Use cases are useful for scoping.
 - ◆ Use cases make it easy to take a staged delivery approach to projects; they can be relatively easily added and removed from a software project as priorities change.
- Use cases have proved to be easily understandable by business users, and so have proven an excellent bridge between software developers and end users.

Pros

- Use case specifications don't use a special language. They can be written in a variety of styles to suit the particular needs of the project.
- Use cases are concerned with the interactions between the user and the system.
 - ◆ UI designers can get involved in the development process either before or in parallel with software developers.

Pros

- Use cases put requirements in context, they are clearly described in relationship to business tasks.
- Use case diagrams help stakeholders to understand the nature and scope of the business area or the system under development.

Pros

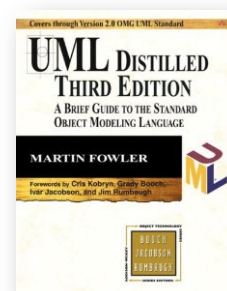
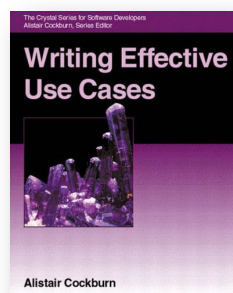
- Test Cases (System, User Acceptance and Functional) can be directly derived from the use cases
- Use cases are critical for the effective execution of Performance Engineering

Cons

- Use case flows are not well suited to easily capturing
 - ♦ non–interaction based requirements of a system (such as algorithm or mathematical requirements) or
 - ♦ non–functional requirements (such as platform, performance, timing, or safety–critical aspects)
- Use cases templates do not automatically ensure clarity. Clarity depends on the skill of the writer(s).

References

- A.Cockburn, “Writing Effective Use Cases”. Addison–Wesley, 2000.
- M.Fowler, “UML Distilled” IV Edition. Addison–Wesley, 2010.



License (1)

- THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.
- BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.
- **1. Definitions**
 - **"Collective Work"** means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.
 - **"Derivative Work"** means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.
 - **"Licensor"** means the individual or entity that offers the Work under the terms of this License.
 - **"Original Author"** means the individual or entity who created the Work.
 - **"Work"** means the copyrightable work of authorship offered under the terms of this License.
 - **"You"** means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.
- **2. Fair Use Rights.** Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.
- **3. License Grant.** Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:
 - a. to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
 - b. to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;The above rights may be exercised in all media and formats whether now known or hereafter devised. The above rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats, but otherwise you have no rights to make Derivative Works. All rights not expressly granted by Licensor are hereby reserved, including but not limited to the rights set forth in Sections 4(d) and 4(e).

83

License (2)

- **4. Restrictions.** The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:
 - a. You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(c), as requested.
 - b. You may not exercise any of the rights granted to You in Section 3 above in any manner that is primarily intended for or directed toward commercial advantage or private monetary compensation. The exchange of the Work for other copyrighted works by means of digital file-sharing or otherwise shall not be considered to be intended for or directed toward commercial advantage or private monetary compensation, provided there is no payment of any monetary compensation in connection with the exchange of copyrighted works.
 - c. If you distribute, publicly display, publicly perform, or publicly digitally perform the Work, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied, and to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work. Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.
 - d. For the avoidance of doubt, where the Work is a musical composition:
 - i. **Performance Royalties Under Blanket Licenses.** Licensor reserves the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work if that performance is primarily intended for or directed toward commercial advantage or private monetary compensation.
 - ii. **Mechanical Rights and Statutory Royalties.** Licensor reserves the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions), if Your distribution of such cover version is primarily intended for or directed toward commercial advantage or private monetary compensation.
 - **Webcasting Rights and Statutory Royalties.** For the avoidance of doubt, where the Work is a sound recording, Licensor reserves the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions), if Your public digital performance is primarily intended for or directed toward commercial advantage or private monetary compensation.

License (3)

- **5. Representations, Warranties and Disclaimer**
 - UNLESS OTHERWISE MUTUALLY AGREED BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.
 - **6. Limitation on Liability.** EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
 - **7. Termination**
 - a. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.
 - b. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.
 - **8. Miscellaneous**
 - a. Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.
 - b. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
 - c. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.
 - d. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.
-