APPLICATION-DOMAIN ANALYSIS - USE

SU:E16:L7

CONTENT

- **Kaj’s car rental: Assignment 1**
- **Application domain analysis: Use**

ASSIGNMENT 1

KAJ'S CARS

THE ASSIGNMENT

- **System definition**

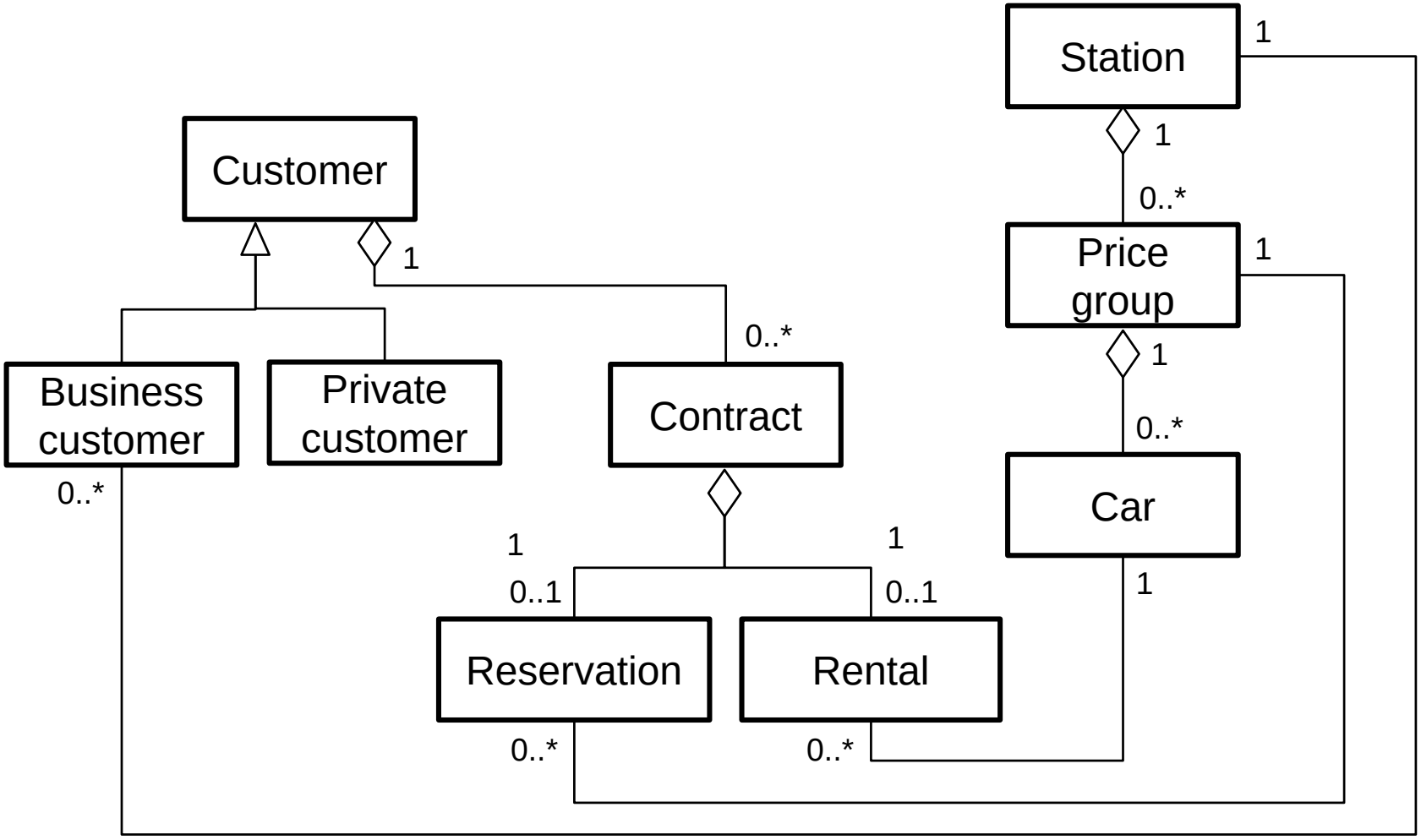
  A computerized system to support work in connection with the rental, booking and maintenance of cars in Kay's Cars. The system is used by a diverse group of people with varying computer skills, and is based on a PC at the counter, as the only access to the system. The system deals with cars in different price groups, private and business customers as well as the lease agreements entered with customers who rent cars. The system is a tool used to create an overview of the stock and handle the details of each rental.

1. Complete the activity ‘Classes' and describe the result in an event table.
2. Complete the activity 'Structure' and describe the result in a class-diagram.

EVENT TABLE

	Customer	Private customer	Business customer	Lease	Reservation	Rental	Station	Price group	Car
Reserved	X	X	X	X	X			X	
Cancelled	X	X	X	X	X			X	
Rented with deposit		X		X	X	X			X
Rented			X	X	X	X			X
Returned	X	X	X	X		X			X
Payed						X			
Bought								X	X
Sold								X	X
Move planned								X	
Move cancelled								X	
Car moved								X	X
Delivered for repairs									X
Returned from repairs									X
Business c. created			X				X		
Business c. closed			X				X		
Price group created							X	X	
Price group closed							X	X	
Station created							X		
Station closed							X		

CLASS DIAGRAM



APPLICATION-DOMAIN ANALYSIS

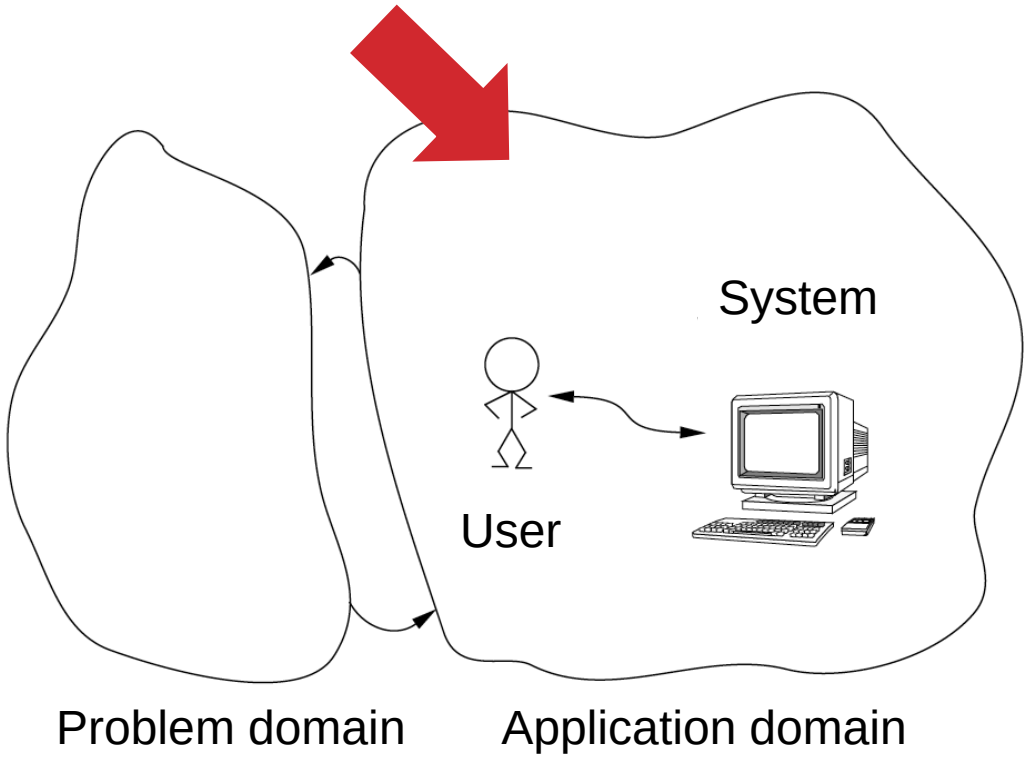
October 18, 2016

MODEL THE CONTEXT

Application domain:
The organization that administrates, monitors, or controls a problem domain.

Focus:

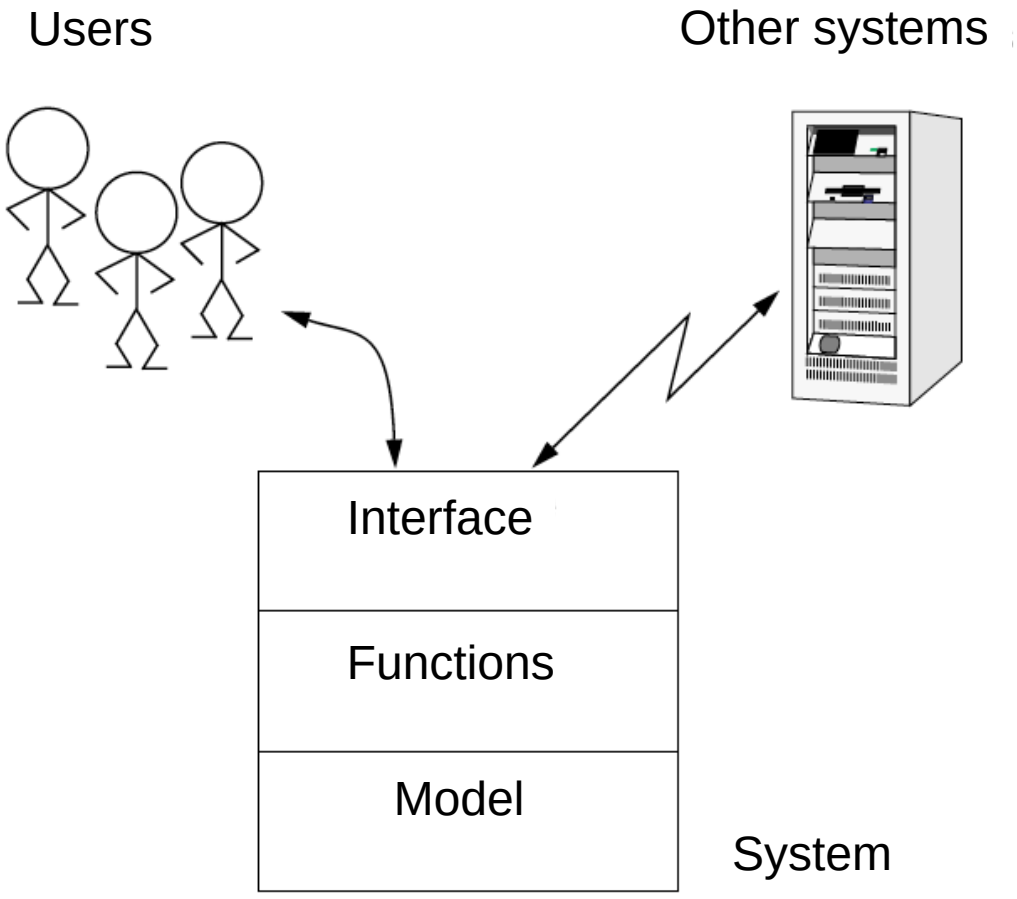
- Actors
- Use cases
- Functions
- Interfaces



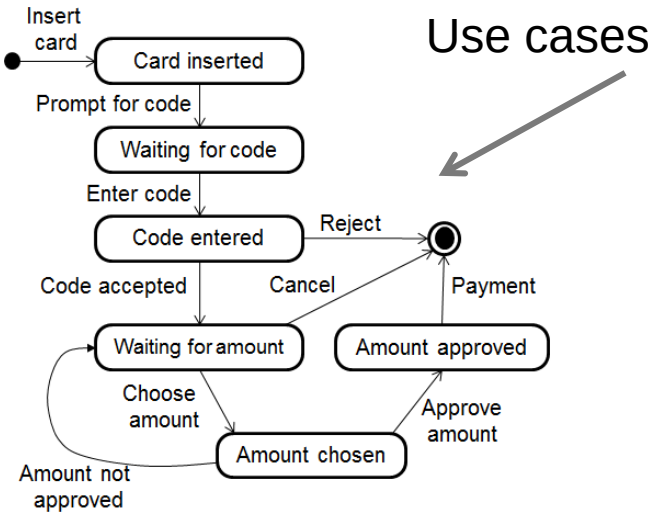
EMPHASIZE THE ARCHITECTURE

Use case: A pattern for interaction between the system and actors in the application domain.

IT-system: A collection of components that implements modeling requirements, functions and interfaces.



APPLICATION-DOMAIN ANALYSIS: RESULT

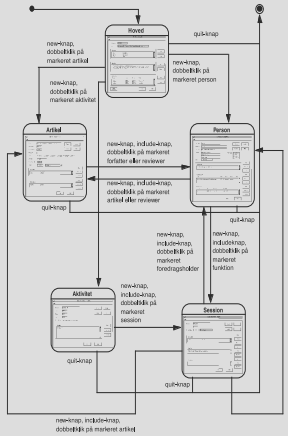


Use cases

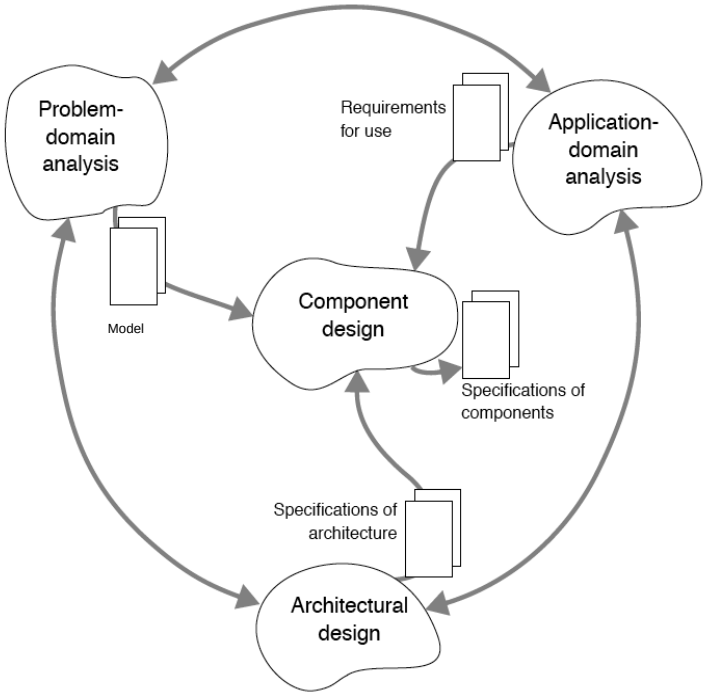
Make schedule	Very complex	Update
Calculate schedule consequences	Complex	Signal
Employment	Simple	Update
Query possible appointments	Complex	Read
Erase schedule	Simple	Update
Make appointment	Medium	Update
. . .	. . .	. . .

List of functions

Interfaces



ACTIVITIES IN APPLICATION-DOMAIN ANALYSIS



Use

- How does the system interact with people and systems in the context?
- Use case and actor

Functions

- What are the system's information processing capabilities?
- Function

Interfaces

- What are the target system's interface requirements?
- Interface, user interface, system interface

Stable properties ⟷ Transient properties

Model — Functions — Interfaces

OVERVIEW OF 'APPLICATION-DOMAIN'

Purpose	• To determine a system's usage requirements.
Concepts	• Application-domain: An organization that administrates, monitors, or controls a problem-domain • Requirements: A system's externally observable behavior.
Principles	• Determine the application domain with use cases. • Collaborate with users.
Result	• A complete list of the systems' overall usage requirements.

USE

October 18, 2016

OVERVIEW OF 'USE'

Purpose	• To determine how actors interact with a system.
Concepts	• Actor: An abstraction of users or other systems that interact with the target system. • Use case: A pattern for interaction between the system and actors in the application domain.
Principles	• Determine the application domain with use cases. • Evaluate use cases in collaboration with users. • Assess social changes in the application domain.
Result	• Descriptions of all use cases and actors.

RESULT(1): ACTOR AND USE CASE OVERVIEW

Use cases	Actors			
	Account owner	Creditor	Administrator	Liquidity monitor
Payment	V	V		
Cash withdrawal	V			
Money transfer	V	V	V	
Account information	V		V	V
Credit information		V	V	
Registration			V	
Monitoring			V	
Error correction			V	

Actor: An abstraction of users or other systems that interact with the target system.

Use case: A pattern for interaction between the system and actors in the application domain.

RESULT(2): ACTOR AND USE CASE SPECIFICATIONS

Account owner

Goal: A person who owns an account. The account owner's basic need is to make payments with their plastic cards

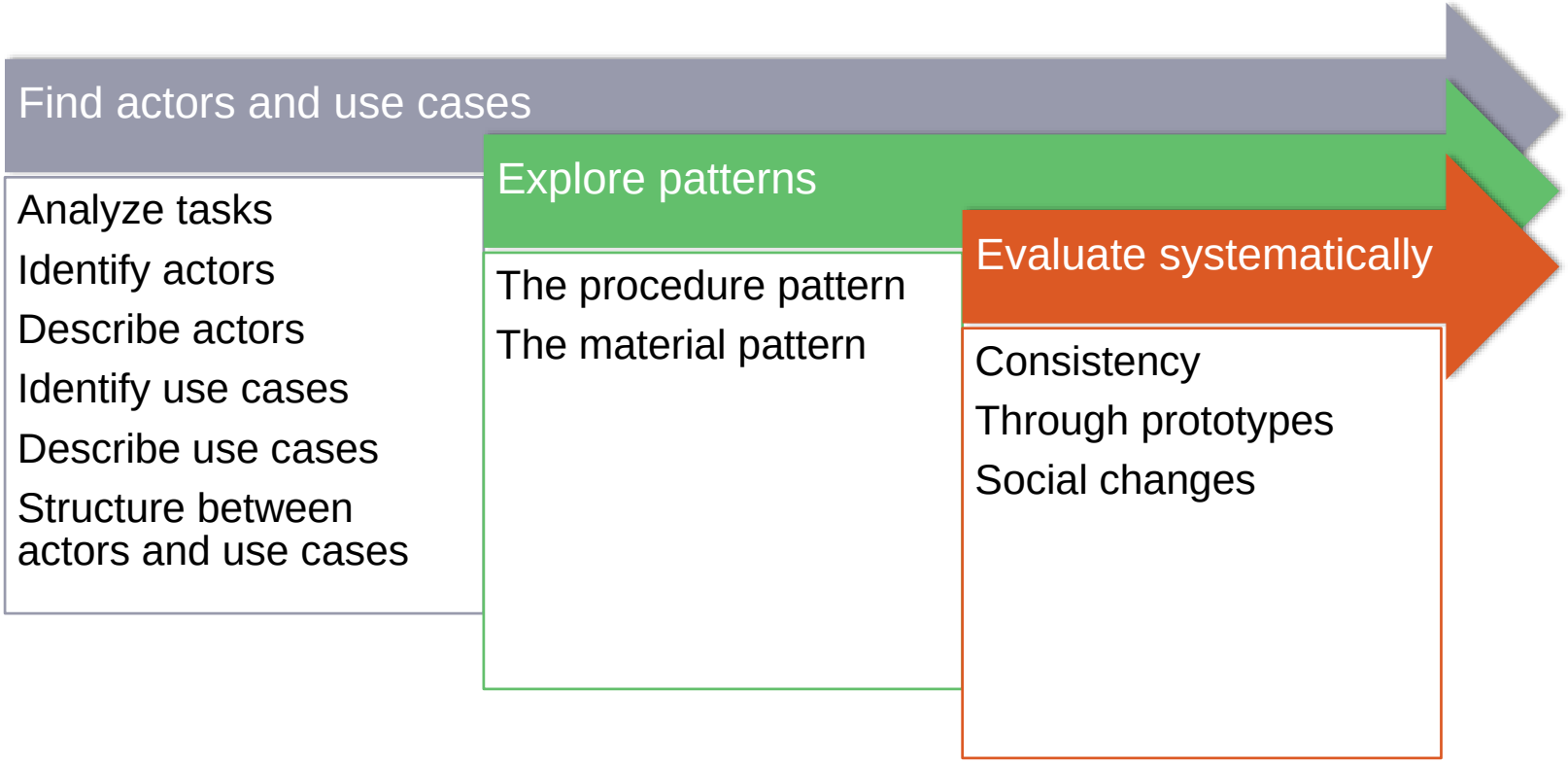
Characteristics: The system's user's include many account owners with different levels of experience and sophistication.

Examples: Account owner A feels insecure about using a plastic card as a form of payment. Owner A originally got a card because it …

Cash withdrawal

Use case: Cash withdrawal is initiated by the account owner when he wants to use his credit card to withdraw cash from an ATM. The account owner inserts his card in the ATM, and is then asked, via the screen, to type in the pin code. The screen will either show a polite denial, in which case the card will be ejected from the ATM and the process will be cancelled; or the screen will show a menu asking the account owner to choose an amount of money by typing on the ATM keyboard. A new screen picture …

ACTIVITIES IN ‘USE’



ANALYZE TASKS (1)

- **Principle:**
  - Determine the application-domain with use cases.
  - Avoid detailed descriptions of “how it used to be”.
- **Goal:**
  - Overview of the work tasks in the application-domain.
  - Especially the division of work and the task boundaries.
- **Information sources:**
  - System definition
  - Rules, procedures, text books . . .
  - Observation, interviews, think-aloud tests, video, self registering, participation in the work of the users, culture analysis, role play, rich pictures.

ANALYZE TASKS: EXAMPLE

- **Which tasks are in the application-domain?**
- **What is the division of work?**
- **What are the task boundaries?**
- **Describe the tasks:**
  - Name and contents
  - Purpose
  - How are the tasks assigned?
  - Who performs the task?
  - Relation to other tasks
  - Result

- **Tasks in the conference system**
- Establishing a new conference
- Detailed planning of a conference
- Administration of participants
- Registration of a person
- Information to the committee
- Information to participants, writers and reviewers

FIND AND DESCRIBE ACTORS

Who will use the system?

Identify actors

- Determine the division of work and the task-related roles in the target system's context.
- Involve human and system actors
- A specific person or system can appear as different actors
- Actors are different, when they act differently in the use cases

Describe actors

- Actor specifications
  - Goal: role in relation to the target system
  - Characteristics: aspects of actor's use of the system
  - Example: general characteristics

Account owner

Goal: A person who owns an account. The account owner's basic need is to make payments with their plastic cards

Characteristics: The system's user's include many account owners with different levels of experience and sophistication.

Examples: Account owner A feels insecure about using a plastic card as a form of payment. Owner A originally got a card because it was the only way to get an ID card for his checks. Owner A only withdraws money from the ATM in emergency situations.

Account owner B is technically curious and she uses the system often, optimally and to the limit of its abilities. B has never had any major problems understanding the potential of the system, and B also examines the opportunities that are not obviously accessible.

GROUP ASSIGNMENT

- **Study the course and semester pages in Moodle**
- **Identify tasks and actors related to collecting, displaying, finding and updating information on the pages.**

IDENTIFY USE CASES

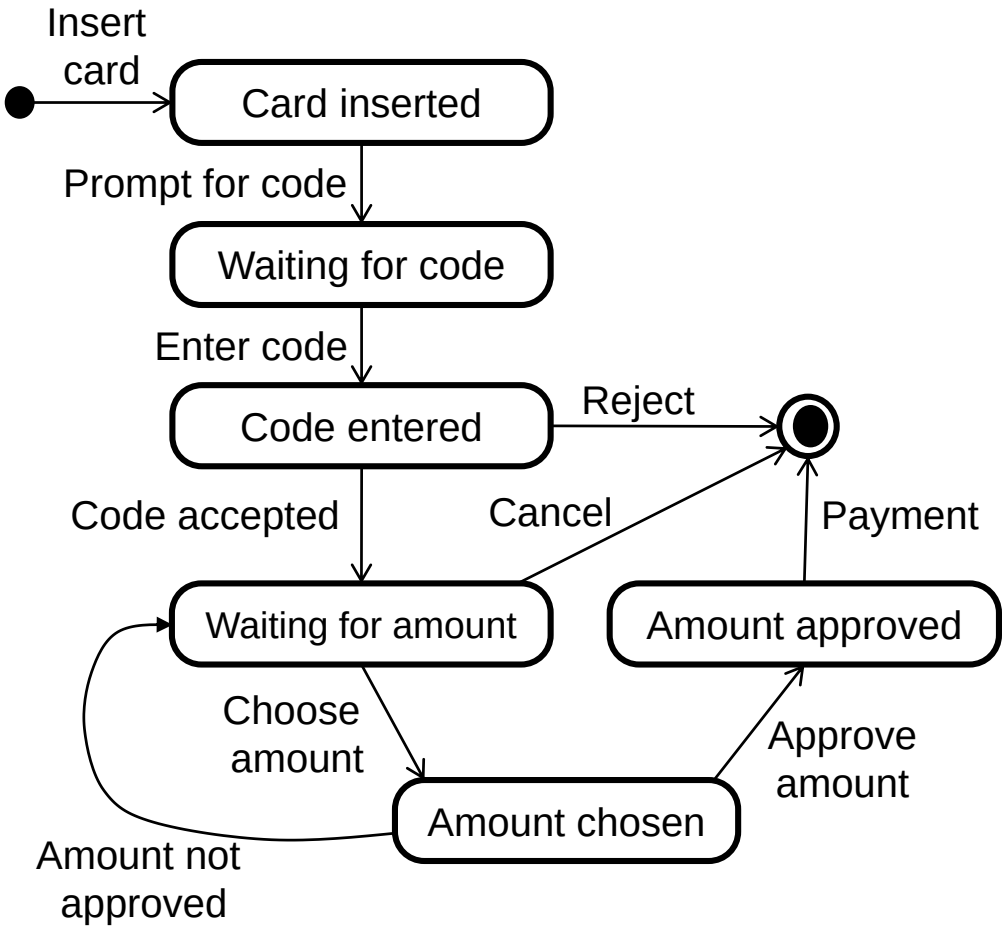
How will the system be used?

- **Use cases provide an overview of the system's requirements**
  - From the user's perspective
  - Foundation for defining and evaluating basic function and interface requirements
- **Identify use cases**
  - Describe how *actors interact* with the *system* to *complete tasks.*
  - Minimize overlap between use-cases
  - Coherent
- **Write scenarios**
  - Examples of execution of tasks with use cases
- **Describe use cases**
  - As text (use case specification)
  - As state chart diagram

DESCRIBE USE CASES

Cash withdrawal

Use case: Cash withdrawal is initiated by the account owner when he wants to use his credit card to withdraw cash from an ATM. The account owner inserts his card in the ATM, and is then asked, via the screen, to type in the pin code. The screen will either show a polite denial, in which case the card will be ejected from the ATM and the process will be cancelled; or the screen will show a menu asking the account owner to choose an amount of money by typing on the ATM keyboard. A new screen picture asks the account owner to approve the transaction. If the transaction is not approved, the account owner is again asked to type in an amount. Otherwise, the use case end by the ejection of the card, and the desired amount of money is paid.
Objects: (to be added)
Features: (to be added)



EXAMPLE: BANK

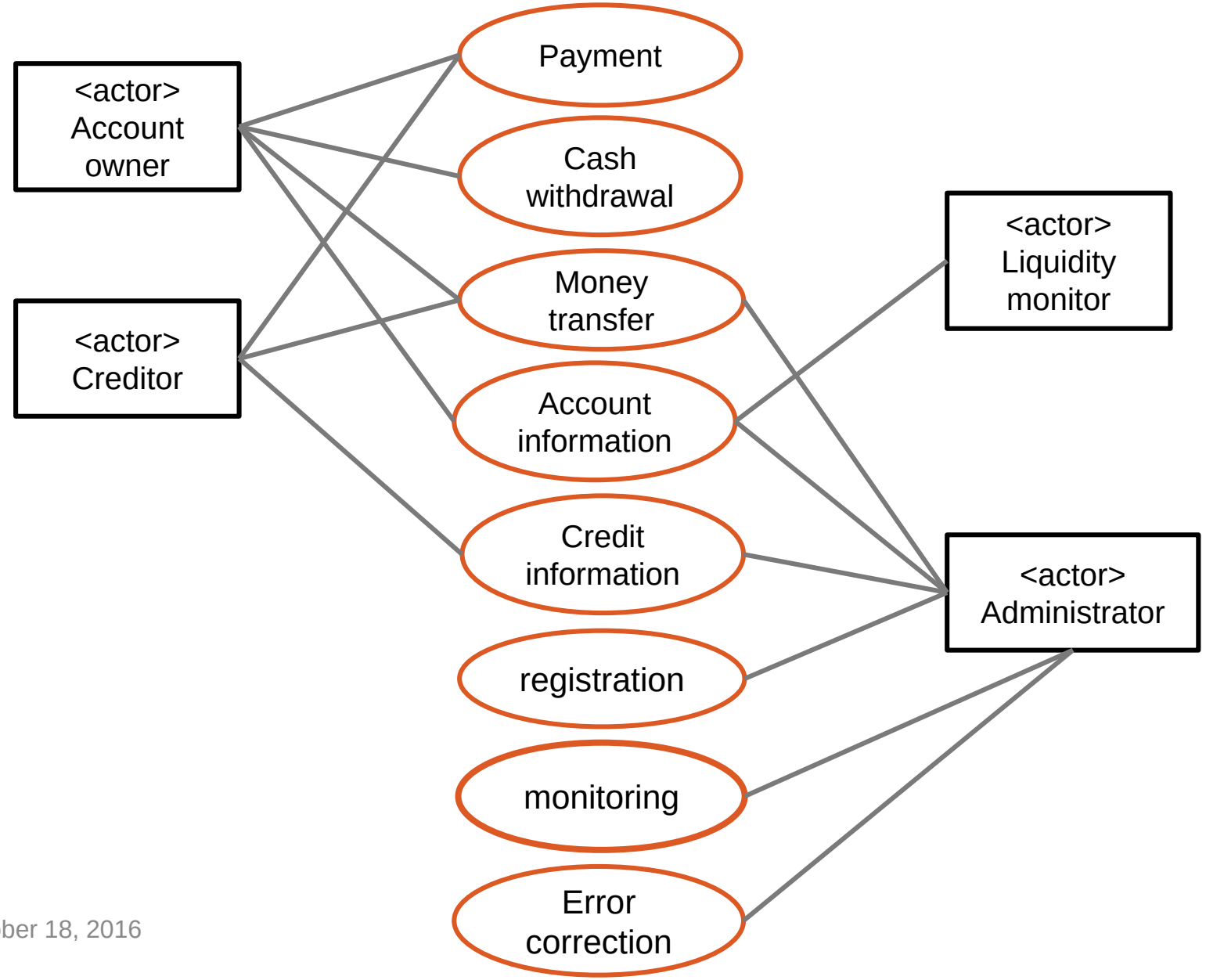
- **Actors are users and other systems interacting with the system**

Example: an automated payment system

- **Four actors:**
  - account owners,
  - creditors,
  - administrators,
  - liquidity monitors

Account information
Use case: Account information is started by the *administrator*, *account owner*, or *liquidity monitor*. In order for an account owner to obtain access to information about an account, he should identify himself using a card. The actor states the account number, as well as what information is desired. The system then responds with the desired information or with a message stating that the information cannot be disclosed.
Objects: Customer, Account

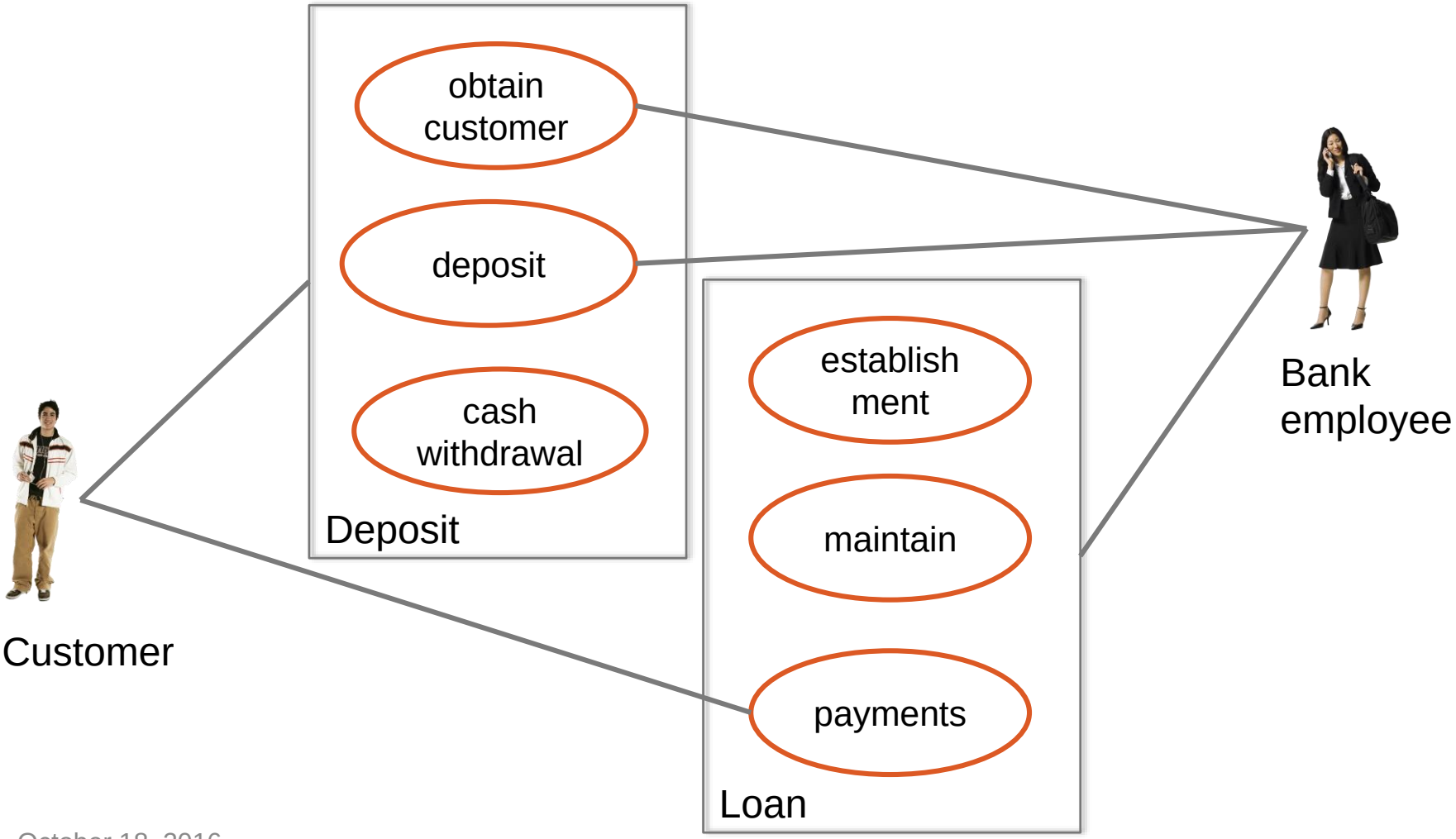
USE CASE OVERVIEW(1): USE CASE DIAGRAM



USE CASE OVERVIEW(2): ACTOR TABLE

Use cases	Actors			
	Account owner	Creditor	Administrator	Liquidity monitor
Payment	V	V		
Cash withdrawal	V			
Money transfer	V	V	V	
Account information	V		V	V
Credit information		V	V	
Registration			V	
Monitoring			V	
Error correction			V	

USE CASE OVERVIEW(3): USE CASE CLUSTERS



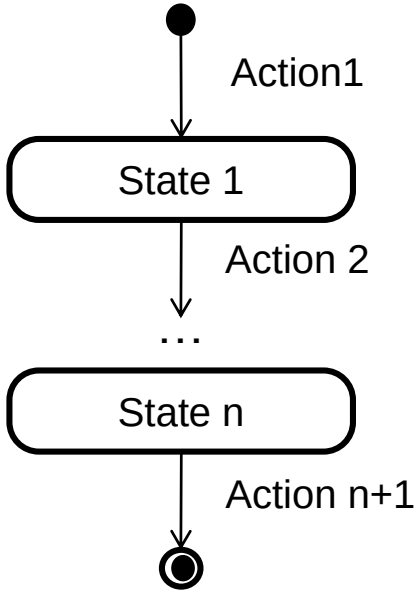
GROUP ASSIGNMENT

- **Course and semester pages in Moodle**
- **Identify tasks and actors related to collecting, displaying, finding and updating information on the pages.**
- **Identify use cases**
- **Make a use case diagram or actor table**

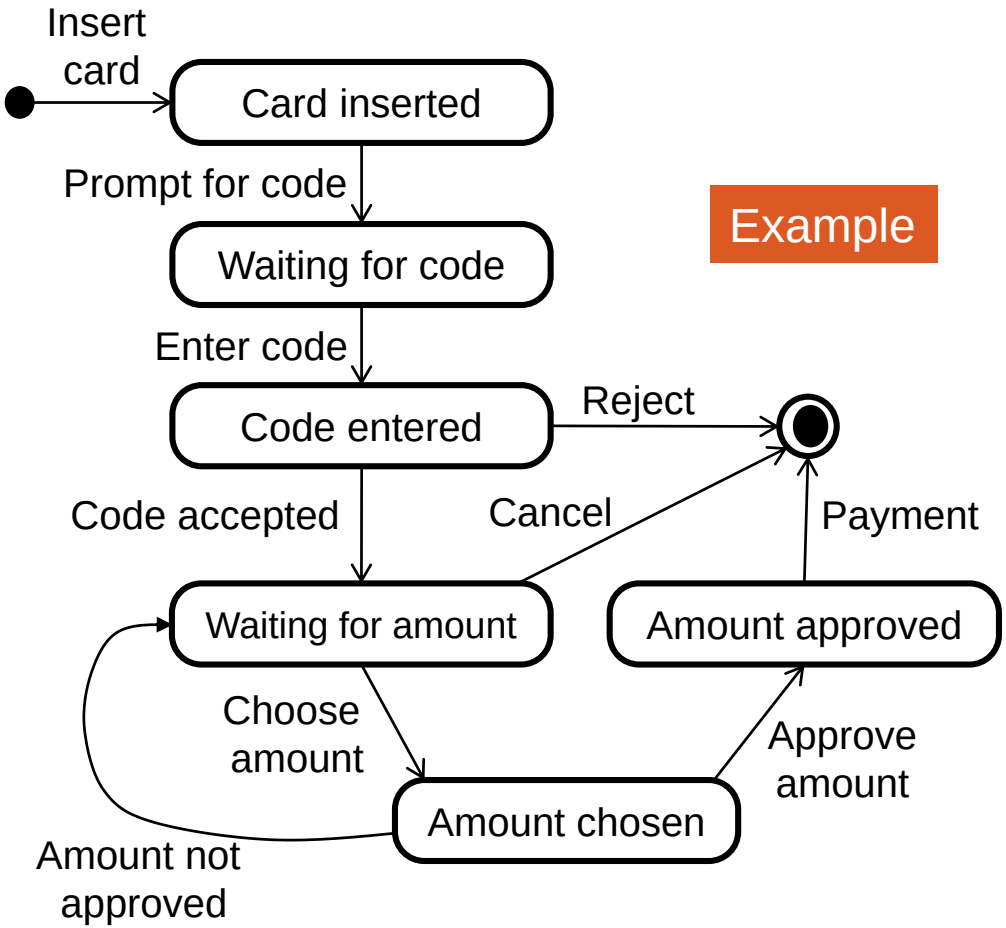
THE PROCEDURAL PATTERN

ensure business rules are observed

Generic



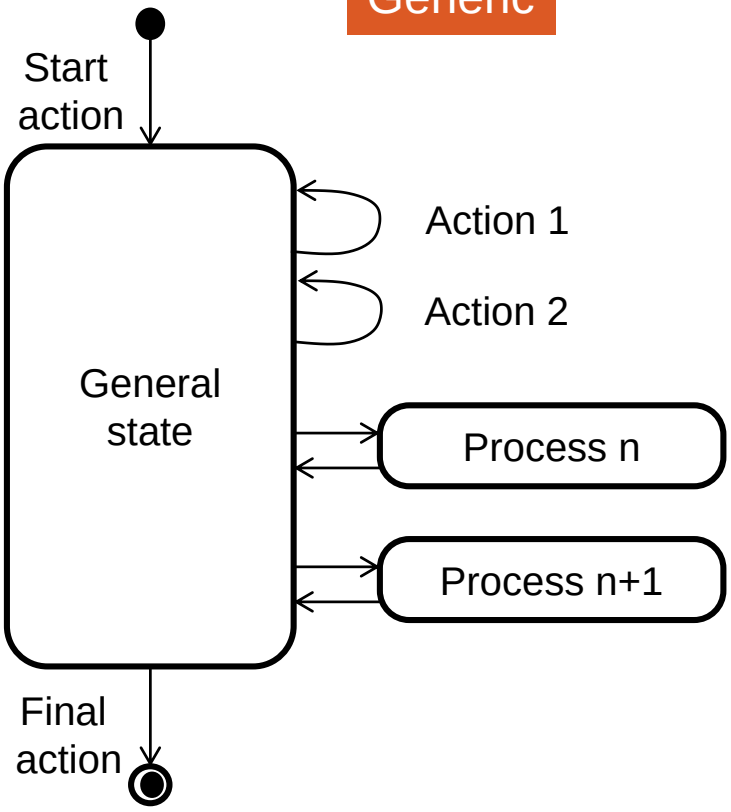
Example



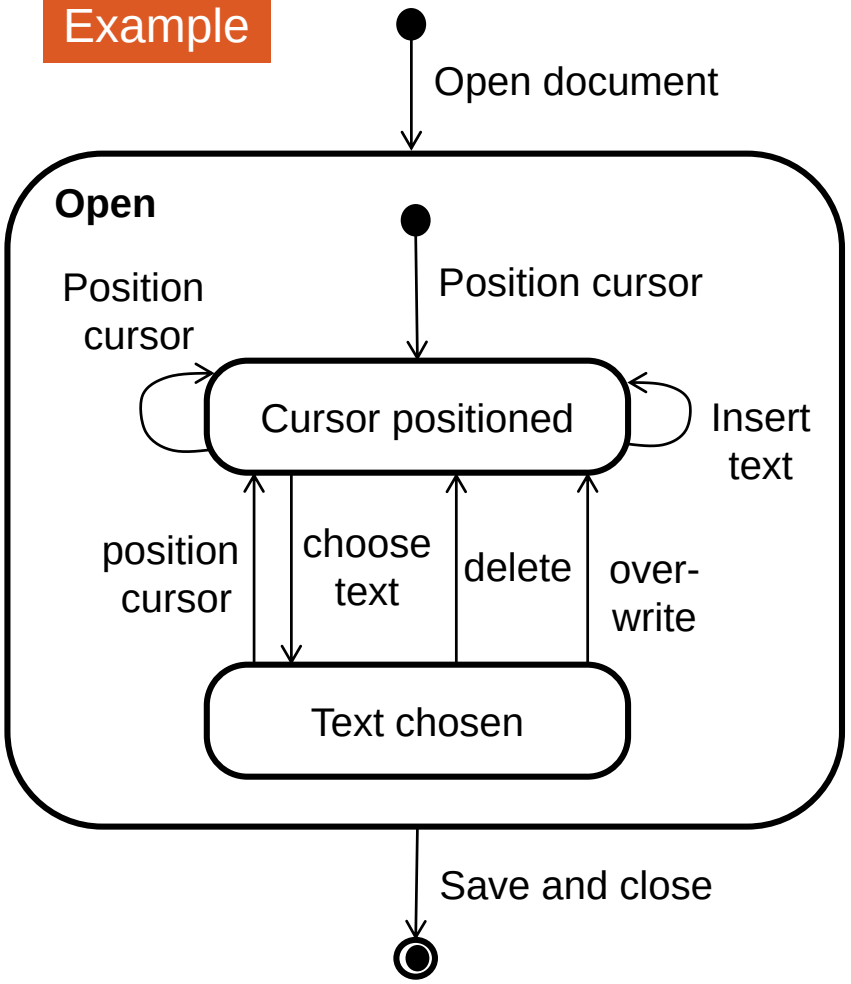
THE MATERIAL PATTERN

The actor can do almost everything in any order

Generic



Example



GROUP ASSIGNMENT

- **Course and semester pages in Moodle**
- **Identify tasks and actors related to collecting, displaying, finding and updating information on the pages.**
- **Identify use cases**
- **Make a use case diagram or actor table**
- **Find examples of the procedure and material patterns**

USE CASES, ACTORS, EVENTS, OBJECTS

Application-domain

Problem domain

Actors

static

Actor: **An abstraction of users or other systems interacting with the target system.**

Use cases

dynamic

Use case**: A pattern for interaction between the system and actors in the application domain.**

Classes

Class: **A description of a collection of objects sharing structure, behavioral pattern, and attributes.**

Behavior

Event**: An instantaneous incident involving one or more objects.**

Example:

A customer enters an order over the Internet

- "Customer“ is both a class and an actor,
- “Order item" is both an event and a use case

EVALUATE SYSTEMATICALLY: CONSISTENCY

- **Each use case should be simple and constitute a coherent whole.**
- **Description of actors and use cases should provide understanding and overview.**
- **Use cases should be described in enough detail to enable identification of functions and interface elements.**
- **Test patterns of use with users.**

EVALUATE SYSTEMATICALLY: PROTOTYPE

- **Use cases are best assessed through planned experiments**

1. Planning
   - Describe the prototype content
2. Development
   - Start with simple prototypes on paper
   - Simple prototypes in for example Power point
   - Functioning prototypes
3. Preparation
4. Test
   - Cooperation
   - Realism
   - Which users
5. Summarizing

EVALUATE SYSTEMATICALLY: SOCIAL CHANGES

Parameters that you can use to evaluate use cases and their social implications.

	The Mechanistic Extreme	The Romantic Extreme
Work content	Specialized job Polarized division of labor Many procedures and rules Regulated by rules	Varied job No division of labor No procdures and rules Regulated by consequences
Autonomy and control	Monitoring Stressful load Little influence on job Low general influence	Self regulation No performance quotas Great influence on own job High general influence
Social relations	No security No self-realization Little social interaction Alienated	Security Self-realization Great social interaction Integrated
Education and development	No education requirements Stagnation	Extensive education requirements Development

OVERVIEW OF 'USE'

Purpose	• To determine how actors interact with a system.
Concepts	• Actor: An abstraction of users or other systems that interact with the target system. • Use case: A pattern for interaction between the system and actors in the application domain.
Principles	• Determine the application domain with use cases. • Evaluate use cases in collaboration with users. • Assess social changes in the application domain.
Result	• Descriptions of all use cases and actors.