Moving Towards Specifications

- What functions will the new system provide?
  - How will people interact with it?
  - Describe functions from a user's perspective
- UML Use Cases
  - Used to show:
    - the functions to be provided by the system
    - which actors will use which functions

UML Use Case Diagrams

Capture the relationships between actors and use cases.

Notation for Use Case Diagrams

Use cases and Actors

- Use case:
  - a pattern of behavior that the new system is required to exhibit
  - a sequence of related actions performed by an actor and the system via a dialogue.
- Actor:
  - anything that needs to interact with the system:
    - a person
    - a role that different people may play
    - another (external) system.

Example: Staff Management
When one use case adds behaviour to a base case
- used to model a part of a use case that the user may see as optional system behavior;
- also models a separate sub-case which is executed conditionally.

One use case invokes another (like a procedure call);
- used to avoid describing the same flow of events several times
- puts the common behavior in a use case of its own.

Example: Car

Example: Meeting Scheduler

Identifying Actors

- Look for:
  - the users who directly use the system
  - also others who need services from the system

- To find actors that are people/roles ask:
  - Who will be a primary user of the system? (primary actor)
  - Who will need support from the system to do her daily tasks?
  - Who will maintain, administrate, keep the system working? (secondary actor)
  - Who or what has an interest in the results that the system produces?

- To find actors that are external systems ask:
  - Which hardware devices does the system need?
  - With which other systems does the system need to interact with?

Finding Use Cases

- For each actor, ask the following questions:
  - Which functions does the actor require from the system?
  - What does the actor need to do?
  - Does the actor need to read, create, destroy, modify, or store some kinds of information in the system?
  - Does the actor have to be notified about events in the system?
  - Does the actor need to notify the system about something?
  - What do those events require in terms of system functionality?
  - Could the actor’s daily work be simplified or made more efficient through new functions provided by the system?
Generalizations

- **Actor classes**
  - It's sometimes useful to identify classes of actor
  - E.g. where several actors belong to a single class
  - Some use cases are needed by all members in the class
  - Other use cases are only needed by some members of the class
  - Actors inherit use cases from the class

- **Use Case classes**
  - Sometimes useful to identify a generalization of several use cases

Exercise: Online Order System

Prepare a use case diagram for the online order system:

- Using the online order system, the customer places his order (by adding items, possibly removing items, and then submitting the order).
- The order clerk retrieves the order from the system and assigns it to a delivery person.
- The delivery person delivers the order to the customer.

UML Sequence Diagrams

- **Describe a Use Case using Sequence Diagrams**
  - Sequence diagrams show step-by-step what's involved in a use case
  - Which objects are relevant to the use case
  - How those objects participate in the function
  - You may need several sequence diagrams to describe a single use case.
  - Each sequence diagram describes one possible scenario for the use case
  - Sequence diagrams…
  - ...should remain easy to read and understand.
  - ...do not include complex control logic

Example: Place book order

Example: Calculate staff bonuses

Example: Add an advertisement
Modelling Sequences of Events

- Objects “own” information and behaviour
  - Objects don’t “know” about other objects’ information, but can ask for it.
  - To carry out business processes, objects have to collaborate.
    - ...by sending messages to one another to invoke each others’ operations
  - Objects can only send messages to one another if they “know” each other
    - I.e. if there is an association between them.

Exercise: Online Order System

- Prepare a sequence diagram for the Assign Driver use case:

The order clerk requests a list of drivers from the system. Using the system, he selects a driver and the system checks the driver’s availability. This is repeated until a driver has been selected (until an available driver is found). Then the order clerk assigns the driver using the system, and the system notifies the delivery person.

References
