Lecture 7

INCEPTION

Questions?

- What is the vision and business case for this project?
- Feasible?
- Buy and/or build?
- Rough estimate of cost: Is it $10K-100K or in the millions?
- Should we proceed or stop?

What Artifacts May Start in Inception?

Artifact[1]	Comment
Vision and Business Case	Describes the high-level goals and constraints, the business case, and provides an executive summary.
Use-Case Model	Describes the functional requirements, and related non-functional requirements.
Supplementary Specification	Describes other requirements.
Glossary	Key domain terminology.
Risk List & Risk Management Plan	Describes the business, technical, resource, schedule risks, and ideas for their mitigation or response.
Prototypes and proof-of-concepts	To clarify the vision, and validate technical ideas.
Iteration Plan	Describes what to do in the first elaboration iteration.
Phase Plan & Software Development Plan	Low-precision guess for elaboration phase duration and effort. Tools, people, education, and other resources.
Development Case	A description of the customized UP steps and artifacts for this project. In the UP, one always customizes it for the project.

Use Case Modeling

Practical Way of Doing

Agenda

- Identify and write use cases.
- Relate use cases to user goals and elementary business processes.
- Use the brief, casual, and fully dressed formats, in an essential style.
- Relate use case work to iterative development.

Introduction

- The UP defines the **Use-Case Model within the Requirements discipline.**
- Essentially, this is the set of all use cases; it is a model of the system's functionality and environment

Use Case Diagrams

A formal way of representing how a business system interacts with its environment

Illustrates the activities that are performed by the users of the system

A scenario-based technique in the UML

A sequence of actions a system performs that yields a valuable result for a particular actor

Use Case Analysis

- What is an Actor?
  - A user or outside system that interacts with the system being designed in order to obtain some value from that interaction
- Use Cases describe scenarios that describe the interaction between users of the system (the actor) and the system itself.

Use Cases

- **Use case diagrams** describe what a system does from the standpoint of an external observer. The emphasis is on *what* a system does rather than *how.*

- Use case diagrams are closely connected to scenarios. A **scenario** is an example of what happens when someone interacts with the system.

Use Cases

- Here is a scenario for a medical clinic.

- A *patient calls the clinic to make an appointment for a yearly checkup. The receptionist finds the nearest empty time slot in the appointment book and schedules the appointment for that time slot. "*

- We want to write a use case for this scenario.
- Remember: A **use case** is a summary of scenarios for a single task or goal.

Use Cases

- Step 1 Identify the actors
- As we read the scenario, define those people or systems that are going to interact with the scenario.

- A *patient calls the clinic to make an appointment for a yearly checkup. The receptionist finds the nearest empty time slot in the appointment book and schedules the appointment for that time slot. "*

Use Cases

- Step 1 Identify the actors
- As we read the scenario, define those people or systems that are going to interact with the scenario.

- A *patient calls the clinic to make an appointment for a yearly checkup. The receptionist finds the nearest empty time slot in the appointment book and schedules the appointment for that time slot. "*

Questions for Identifying People Actors

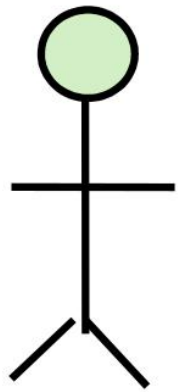
- Who is interested in the scenario/system?
- Where in the organization is the scenario/system be used?
- Who will benefit from the use of the scenario/system?
- Who will supply the scenario/system with this information use this information, and remove this information?
- Does one person play several different roles?
- Do several people play the same role?

Questions for Identifying Other Actors

- What other entity is interested in the scenario/system?
- What other entity will supply the scenario/system with this information, use this information, and remove this information?
- Does the system use an external resource?
- Does the system interact with a legacy system?

Actors

- An Actor is outside or external the system.
- It can be a:
  - Human
  - Peripheral device (hardware)
  - External system or subsystem
  - Time or time-based event
- Represented by stick figure



Use Cases

- A **use case** is a summary of scenarios for a single task or goal.
- An **actor** is who or what initiates the events involved in the task of the use case. Actors are simply roles that people or objects play.
- So as we read our scenario, what or who is the actor????

Use Cases

- So as we read our scenario, what or who is the actor????

- *A patient calls the clinic to make an appointment for a yearly checkup. The receptionist finds the nearest empty time slot in the appointment book and schedules the appointment for that time slot. "*

- The actor is a **Patient**.

Use Cases

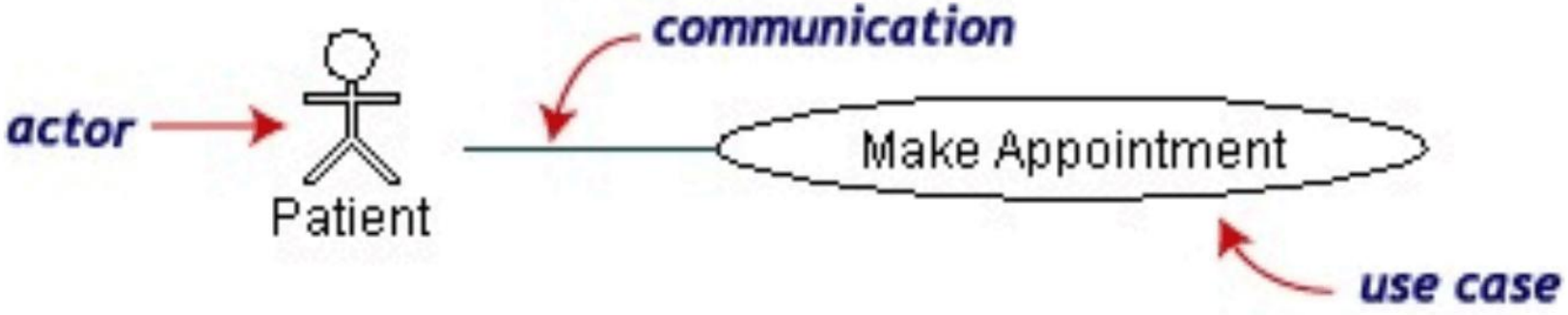
- The **use case** is a summary of scenarios for a single task or goal.

- So What is the Use Case????

- The Use Case is **Make Appointment.**
- It is a use case for the medical clinic.

Use Cases

- The picture below is a **Make Appointment** use case for the medical clinic.
- The actor is a **Patient**. The connection between actor and use case is a **communication association** (or **communication** for short).



Actors are stick figures. Use cases are ovals. Communications are lines that link actors to use cases.

Use Case Componentss

- The use case has three components.
- The **use case** task referred to as the use case that represents a feature needed in a software system.
- The **actor(s)** who trigger the use case to activate.
- The **communication** line to show how the actors communicate with the use case.

Use Case Diagram - Use Case

- A major process performed by the system that benefits an actor(s) in some way
- Models a dialogue between an actor and the system
- Represents the functionality provided by the system

Use Case

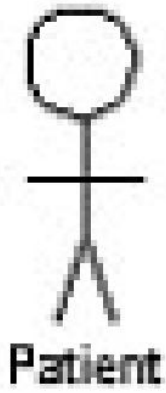
- Each use case in a use case diagram describes one and only one *function* in which users interact with the system
  - May contain several “paths” that a user can take while interacting with the system
  - Each path is referred to as a scenario

Use Case

- Labelled using a descriptive verb-noun phrase
- Represented by an oval

Use Case - Actor

- Labelled using a descriptive noun or phrase
- Represented by a stick character



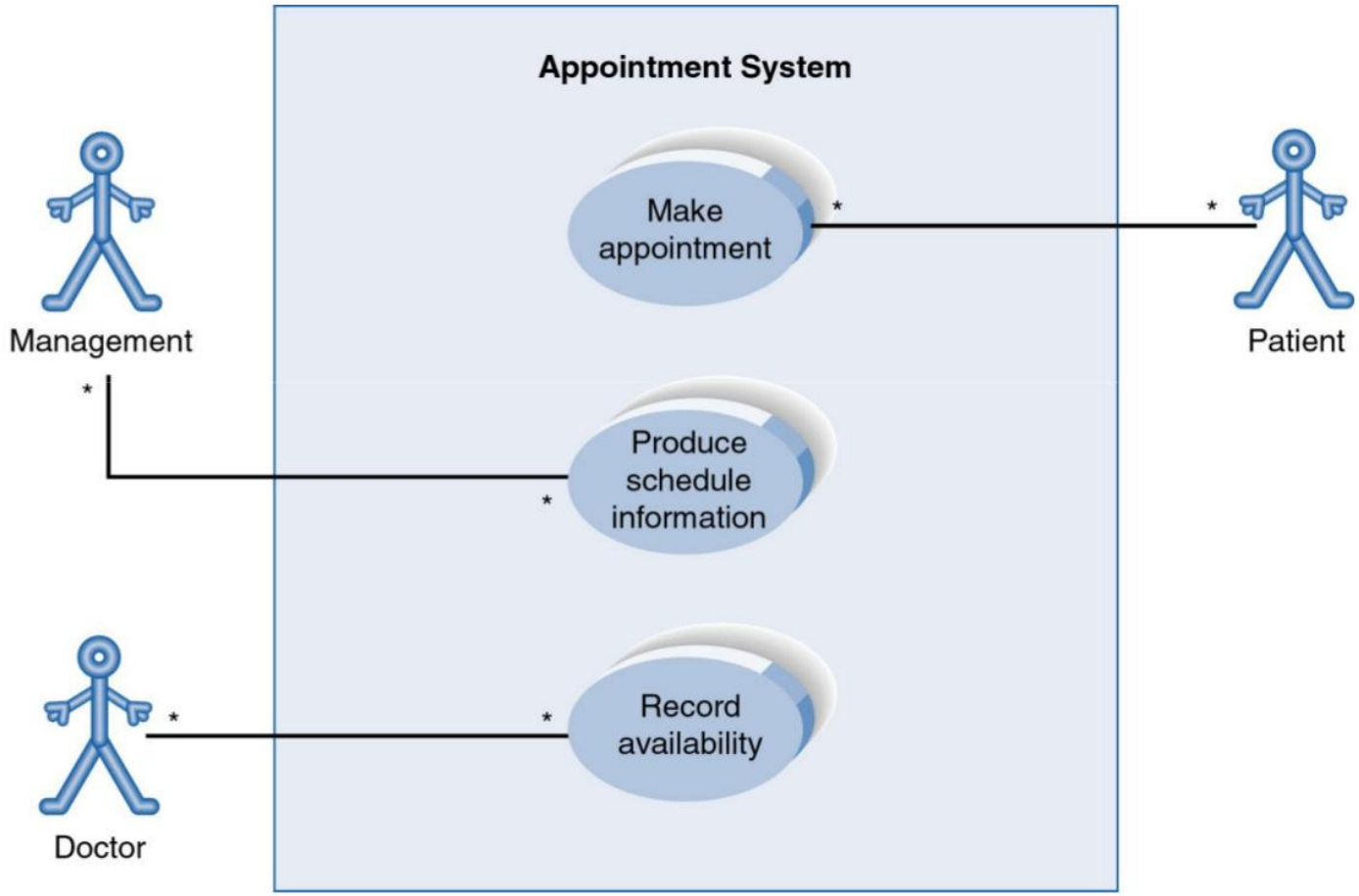
Use Case - Relationships

- Relationships
  - Represent communication between actor and use case
  - Depicted by line or double-headed arrow line
  - Also called association relationship

Use Case - Relationships

- Boundary
  - A boundary rectangle is placed around the perimeter of the system to show how the actors communicate with the system.

Use-Case Diagram



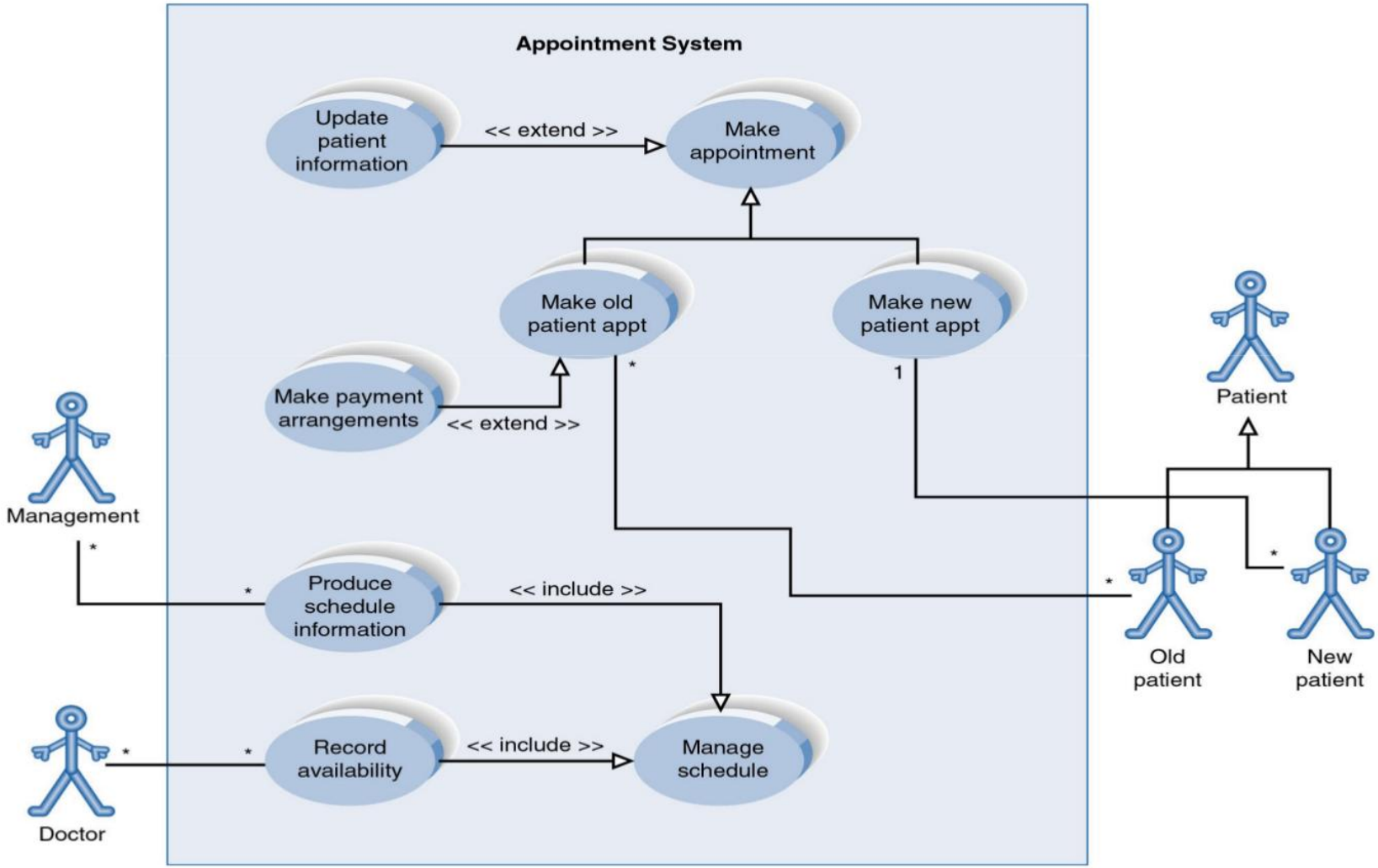
Use Case Diagram

- Other Types of Relationships for Use Cases
  - Generalization
  - Include
  - Extend

Components of Use Case Diagram

- Generalization Relationship
  - Represented by a line and a hollow arrow
    - From child to parent

Example of Relationships



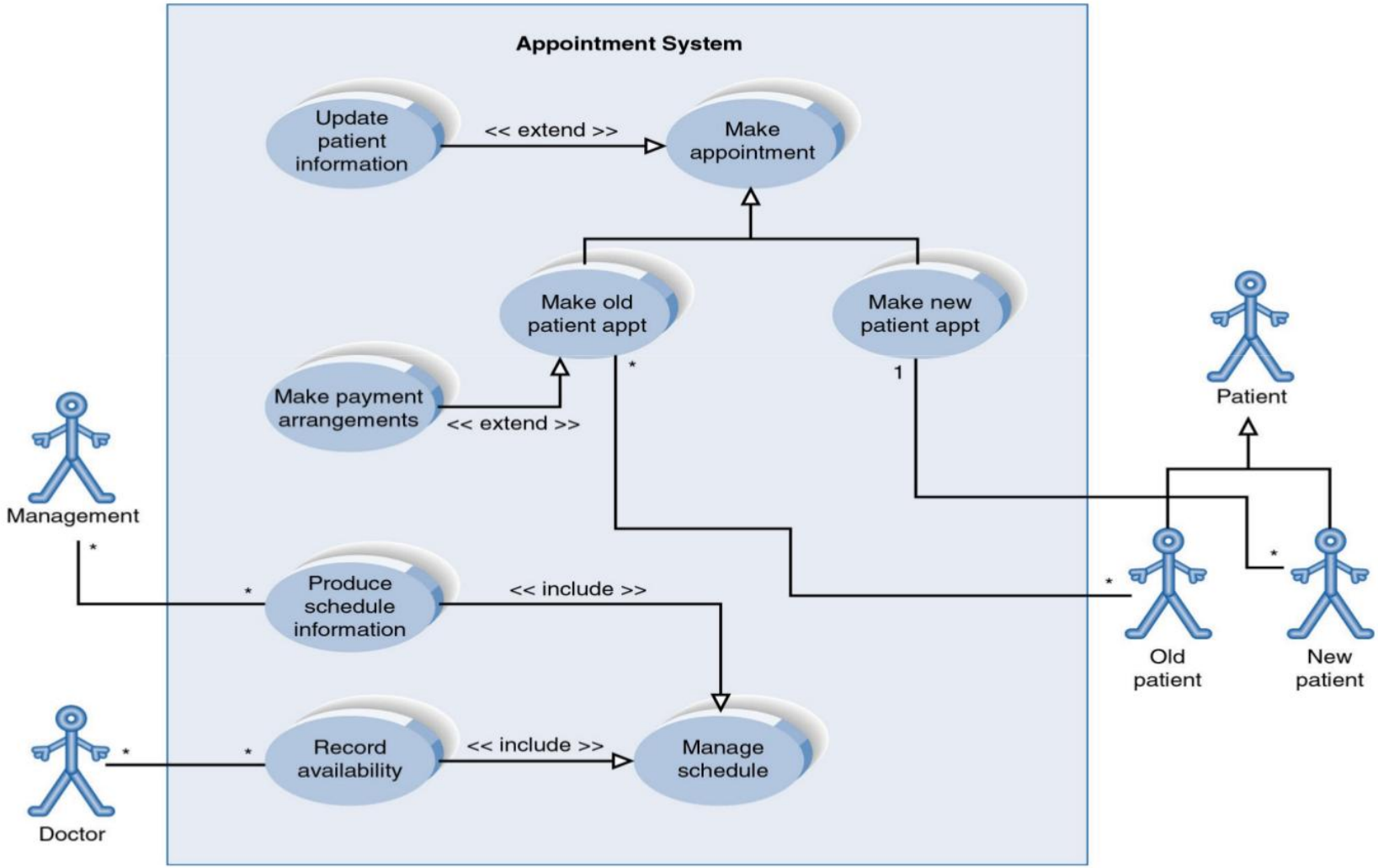
Use Case Diagram

- Include Relationship
  - Represents the inclusion of the functionality of one use case within another
  - Arrow is drawn from the base use case to the used use case
  - Write << include >> above arrowhead line

Use Case Diagram

- Extend relationship
  - Represents the extension of the use case to include optional functionality
  - Arrow is drawn from the extension use case to the base use case
  - Write << extend >> above arrowhead line

Example of Relationships



Use Case Relationships

- Pro:
  - Reduces redundancy in use cases
  - Reduces complexity within a use case
- Con
  - May introduce complexity to use case diagram

Benefits of Use Cases

- Use cases are the primary vehicle for requirements capture in UP
- Use cases are described using the language of the customer (language of the domain which is defined in the glossary)
- Use cases provide a contractual delivery process (UP is Use Case Driven)
- Use cases provide an easily-understood communication mechanism
- When requirements are traced, they make it difficult for requirements to fall through the cracks
- Use cases provide a concise summary of what the system should do at an abstract (low modification cost) level.

Difficulties with Use Cases

- As functional decompositions, it is often difficult to make the transition from functional description to object description to class design
- Reuse at the class level can be hindered by each developer “taking a Use Case and running with it”. Since UCs do not talk about classes, developers often wind up in a vacuum during object analysis, and can often wind up doing things their own way, making reuse difficult
- Use Cases make stating non-functional requirements difficult (where do you say that X must execute at Y/sec?)
- Testing functionality is straightforward, but unit testing the particular implementations and non-functional requirements is not obvious

Use Case Model Survey

- The Use Case Model Survey is to illustrate, in graphical form, the universe of Use Cases that the system is contracted to deliver.
- Each Use Case in the system appears in the Survey with a short description of its main function.
  - Participants:
    - Domain Expert
    - Architect
    - Analyst/Designer (Use Case author)
    - Testing Engineer