UNIVERSITY OF INFORMATION TECHNOLOGY

Faculty of Information Systems

Chapter 2

Requirements Determination and Analysis

Dr. Cao Thi Nhan

LEARNING OBJECTIVES

1. Understand methods for Determining user requirements.

2. Have ability to use suitable method for specific situations.

3. Have ability to use Business Functional Diagram and Use case model to model business functions and processes.

CONTENT

1. Introduction

2. Methods for Determining User Requirements

3. Business Function Diagram (BFD)

4. Use case diagram

INTRODUCTION

Introduction

- A requirement is simply a statement of what the system must do or what characteristics it needs to have:

  - the business needs (business requirements)

  - the users need to do (user requirements)

  - the software should do (functional requirements);

  - characteristics the system should have (nonfunctional requirements);

  - and how the system should be built (system requirements).

Introduction

- **Functional requirements**: how the system will support the user in completing a task.

  1. Process-oriented: A process the system must/should perform.

     - The system must check incoming customer orders for inventory availability.

     - The system should allow students to view a course schedule while registering for classes

  2. Information-oriented: Information the system must contain.

     - The system must retain customer order history for 3 years

     - The system must include real-time inventory levels at all warehouses.

Introduction

- **Nonfunctional requirements**: include important behavioral properties that the system must have.

  1. Operational: The physical and technical environments in which the system will operate
     - The system can run on handheld devices.
     - The system should be able to work on any Web browser

  2. Performance:
     - Any interaction between the user and the system should not exceed 2 seconds
     - The system should be available for use 24 hours per day, 365 days per year.
     - The system supports 300 simultaneous users from 9–11 A.M.; 150 simultaneous users at all other times

Introduction

- **Nonfunctional requirements**: include important behavioral properties that the system must have

  3. Security:

     - Only direct managers can see personnel records of staff

     - The system includes all available safeguards from viruses, worms, Trojan horses, etc.

  4. Cultural and Political:

     - Company policy is to buy computers only from Dell, send salary to staff just using Vietcombank.

     - Personal information is protected in compliance with the Data Protection Act.

Methods for Determining User Requirements

Methods for Determining User Requirements

1. Interview

2. Questionnaires

3. Analyse documents

4. Observation

5. Joint Application Design (JAD)

6. STROBE

Methods for Determining User Requirements

1. Interview

2. Questionnaires

3. Analyse documents

4. Observation

5. Joint Application Design (JAD)

6. STROBE

Interview

- Interview is an important method for collecting data on human and system information requirements

- Interviews reveal information about:

    - Interviewee opinions

    - Interviewee feelings

    - Goals

    - Key HCI concerns

Interview Preparation

- Reading background material

- Establishing interview objectives

- Deciding whom to interview

- Preparing the interviewee

- Deciding on question types and structure

Question Types

- Open-ended

- Closed

Open-Ended Questions

- Open-ended interview questions allow interviewees to respond how they think, wish…

- Useful when you want to understand a larger process or draw out the interviewee's opinions, attitudes, or suggestions

- Example:

  1. How is this task performed?

  2. Why do you perform the task that way?

  3. There are any ways to improve this task?

  4. What added features would you like to have in the new billing system?

Open-Ended Questions

- Advantages

- Disadvantages

Advantages of Open-Ended Questions

- Puts the interviewee at ease

- Makes phrasing easier for the interviewer

- Provides richness of detail

- Reveals avenues of further questioning that may have gone untapped

- Allows the interviewer to pick up on the interviewee's vocabulary

- Useful if the interviewer is unprepared

Disadvantages of Open-Ended Questions

- May result in too much irrelevant detail

- Possibly losing control of the interview

- May take too much time for the amount of useful information gained

- Potentially seeming that the interviewer is unprepared

Closed Interview Questions

- Closed interview questions limit the number of possible responses

- useful when you want information that is more specific or when you need to verify facts.

Closed Interview Questions

- Example:

1. How many personal computers do you have in this department?

2. Do you review the reports before they are sent out?

3. How many hours of training does a clerk receive?

4. Is the calculation procedure described in the manual?

5. How many customers ordered products from the Web site last month?

Closed Interview Questions

- Advantages

- Disadvantages

Benefits of Closed Interview Questions

- Saving interview time

- Easily comparing interviews

- Keeping control of the interview

- Covering a large area quickly

- Getting to relevant data

Disadvantages of Closed Interview Questions

- Boring for the interviewee. Failing to build rapport between interviewer and interviewee

- Failure to obtain rich detailing

- Missing main ideas

Bipolar Questions

- A special kind of closed question

- Bipolar questions are those that may be answered with a "yes" or "no" or "agree" or "disagree"

- Bipolar questions should be used sparingly

Probes

- May be either open-ended or closed

- Probing questions elicit more detail about previous questions

- The purpose of probing questions is:
    - To get more meaning
    - To clarify
    - To draw out and expand on the interviewee's point

Probes

- Example:

1. Why?

2. You mentioned both an intranet and an extranet solution. Please give an example of how you think each differs.

3. Give an example of how ecommerce has been integrated into your business processes.

4. What makes you feel that way?

5. Tell me step by step what happens after a customer clicks the "Submit" button on the Web registration form.

Closing the Interview

- Always ask "Is there anything else that you would like to add?"

- Summarize and provide feedback on your impressions

- Ask whom you should talk with next

- Set up any future appointments

- Thank them for their time.

After the interview: Interview Report

- Write as soon as possible after the interview
- Provide an initial summary, then more detail
- Review the report with the respondent
- Recorder?

Methods for Determining User Requirements

1. Interview

2. Questionnaires

3. Analyse documents

4. Observation

5. Joint Application Design (JAD)

6. STROBE

Questionnaires

Questionnaires are useful in gathering information from key organization members about:

- Attributes: what people in the organization say they want

- Beliefs—what people think is actually true.

- Behavior—what organizational members do.

- Characteristics—properties of people or things.

Tips

- Simple, Short, Specific
- Technically accurate
- Addressed to those who are knowledgeable
- Appropriate for the reading level of the respondent
- Place most important questions first
- Introduce less controversial questions first
- Cluster items of similar content together
- Easy to analyse data (measurement scales)
- Electronic version / Paper version

Examples

Example Likert Scale

1. Wikipedia has a user friendly interface.

strongly agree — agree (X) — neutral — disagree — strongly disagree
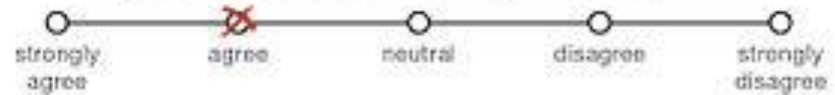
2. Wikipedia is usually my first resource for research.

strongly agree (X) — agree — neutral — disagree — strongly disagree

3. Wikipedia pages generally have good images.

strongly agree — agree — neutral — disagree (X) — strongly disagree
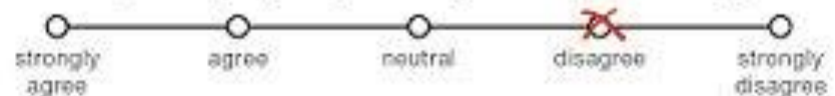
4. Wikipedia allows users to upload pictures easily.

strongly agree — agree — neutral — disagree (X) — strongly disagree

5. Wikipedia has a pleasing color scheme.

strongly agree — agree — neutral (X) — disagree — strongly disagree

Methods for Determining User Requirements

1. Interview

2. Questionnaires

3. Analyse documents

4. Observation

5. Joint Application Design (JAD)

6. STROBE

Analyse various documents

- Internal documents

- Website

- Current Information systems

Methods for Determining User Requirements

1. Interview

2. Questionnaires

3. Analyse documents

4. Observation

5. Joint Application Design (JAD)

6. STROBE

Observation

- Observation provides insight on what organizational members actually do

- Can also reveal important clues regarding HCI concerns

- See firsthand the relationships that exist between decision makers and other organizational members

Methods for Determining User Requirements

1. Interview

2. Questionnaires

3. Analyse documents

4. Observation

5. Joint Application Design (JAD)

6. STROBE

Joint Application Design (JAD)

- Joint Application Design (JAD) can replace a series of interviews with the user community

- JAD is a technique that allows the analyst to accomplish requirements analysis and design the user interface with the users in a group setting

Conditions That Support the Use of JAD

- Users want something new

- The organizational culture supports joint problem-solving behaviors

- Analysts forecast an increase in the number of ideas using JAD

Who Is Involved

- Executive sponsor

- IS analyst

- Users (Customers)

- Session leader

- Observers

- Scribe (secretary)

Who Is Involved
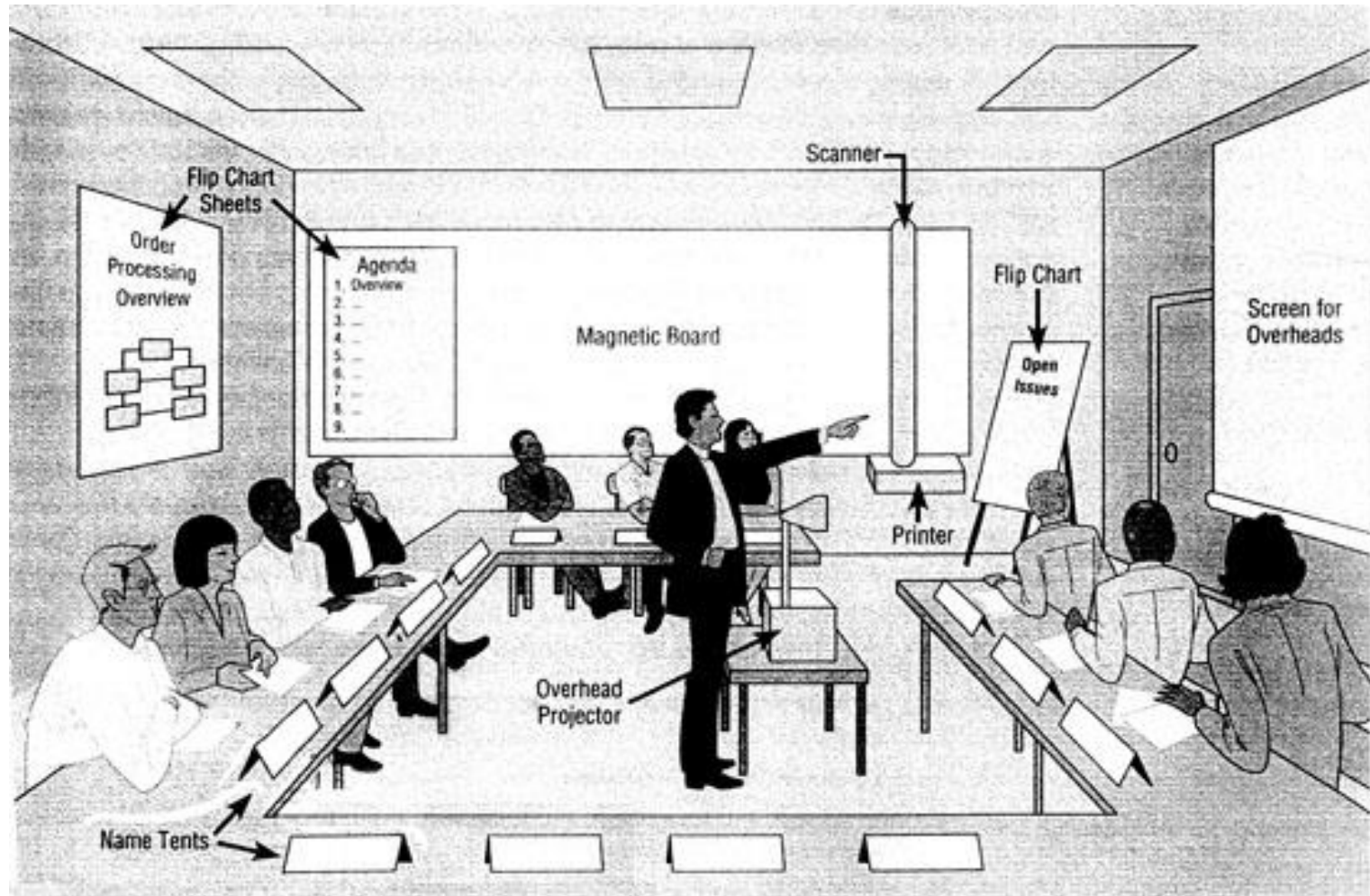
Prentice Hall

Methods for Determining User Requirements

1. Interview

2. Questionnaires

3. Analyse documents

4. Observation

5. Joint Application Design (JAD)

6. STROBE

STROBE

STRuctured **OB**servation of the **E**nvironment — a technique for observing the decision-maker's physical environment

STROBE Elements

- Office location

- Desk placement

- Stationary equipment

- External information sources

- Office lighting and color

- Clothing worn by decision makers

Methods for Determining User Requirements

1. Interview

2. Questionnaires

3. Analyse documents

4. Observation

5. Joint Application Design (JAD)

6. STROBE

How the customer explained it

CONTENT

1. Introduction

2. Methods for Determining User Requirements

3. Business Function Diagram (BFD)

4. Use case analysis

Business Function Diagram (BFD)

Business Function Diagram

- Functional Decomposition Diagrams (FDD)

- A top-down representation of a function or process.

- Using an FDD, an analyst can show business functions and break them down into lower-level functions and processes

- During requirements modeling, analysts use FDDs to model business functions and show how they are organized into lower-level processes

Business Function Diagram
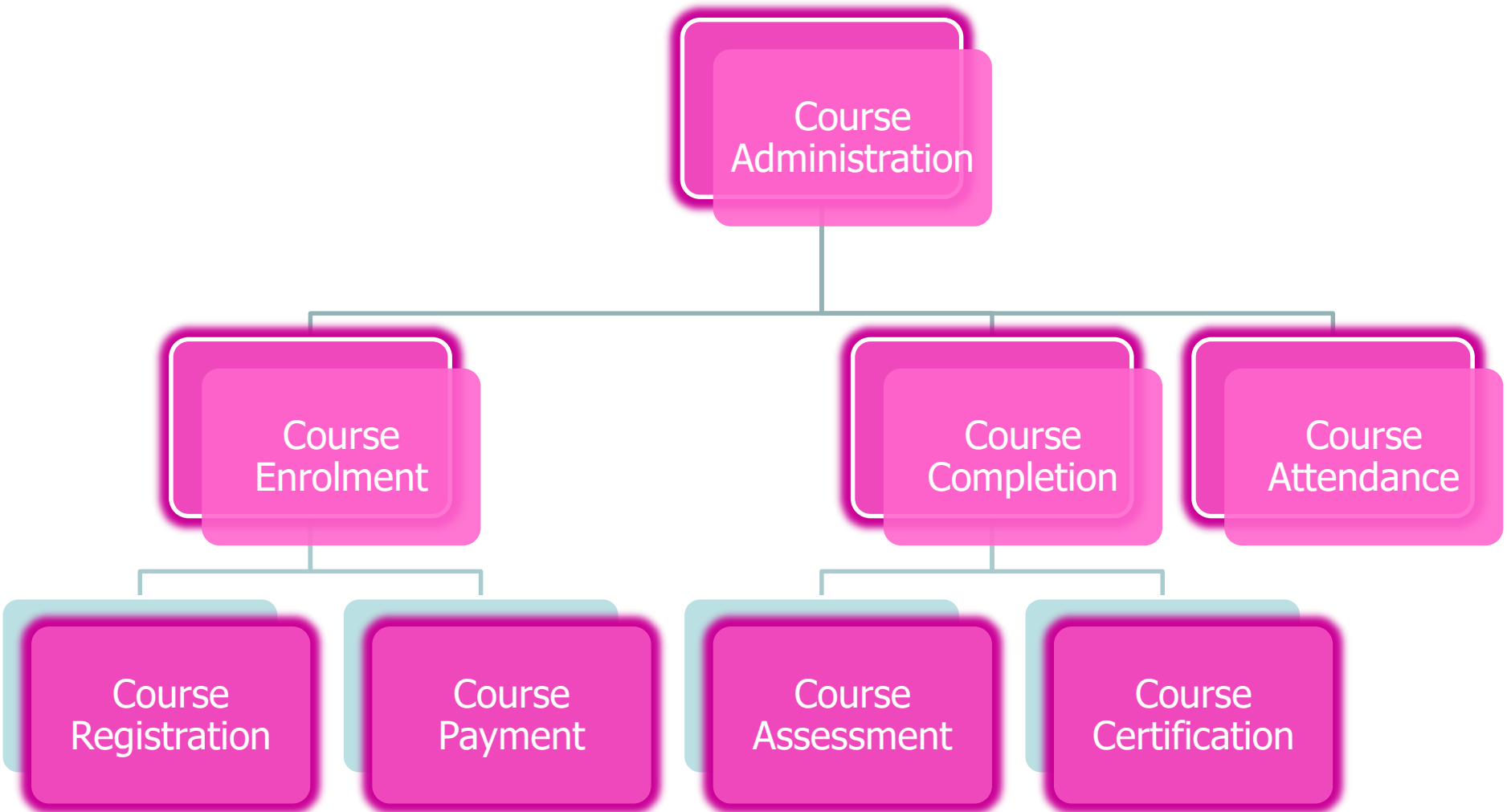
Use case diagram

Use case analysis

- A use case diagram visually represents the interaction between users and the information system

- In a use case diagram, the user becomes an actor, with a specific role that describes how he or she interacts with the system

Actor

- A person / system that derives benefit from and is external to the system

- Is labeled with its role

- Is placed outside the system boundary

- Can be associated with other actors using a specialization / superclass association

Actor/Role   Actor/Role   Customer   Patient   Doctor

Actor

Traveler → Travel Agent → Airline Booking System

Traveler → Airline Booking System
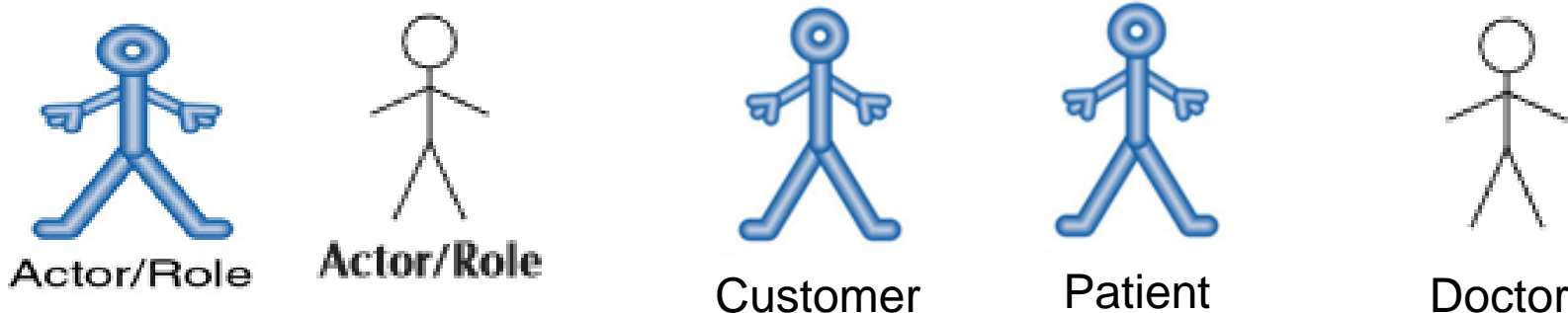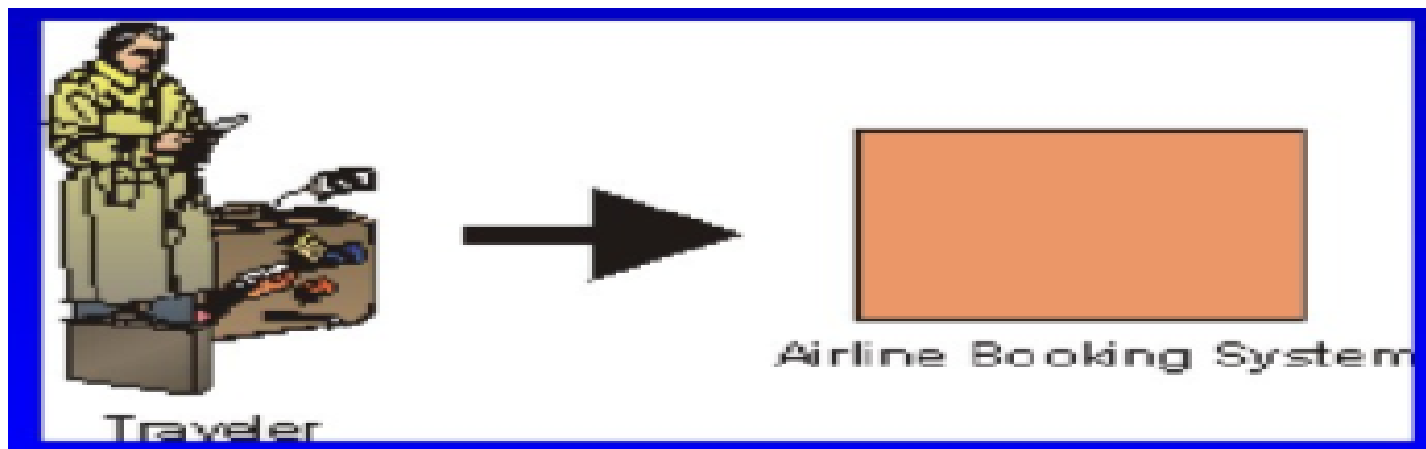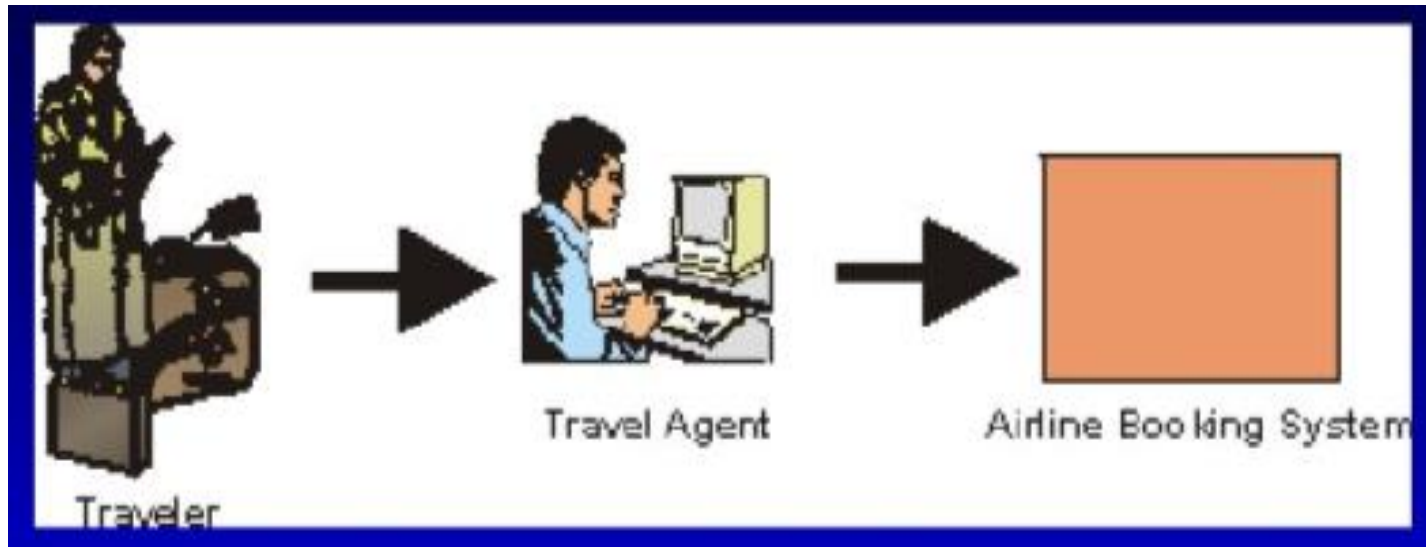
Use case

- Represents a major piece of system functionality

- Can extend another use case

- Can include another use case

- Is placed inside the system boundary
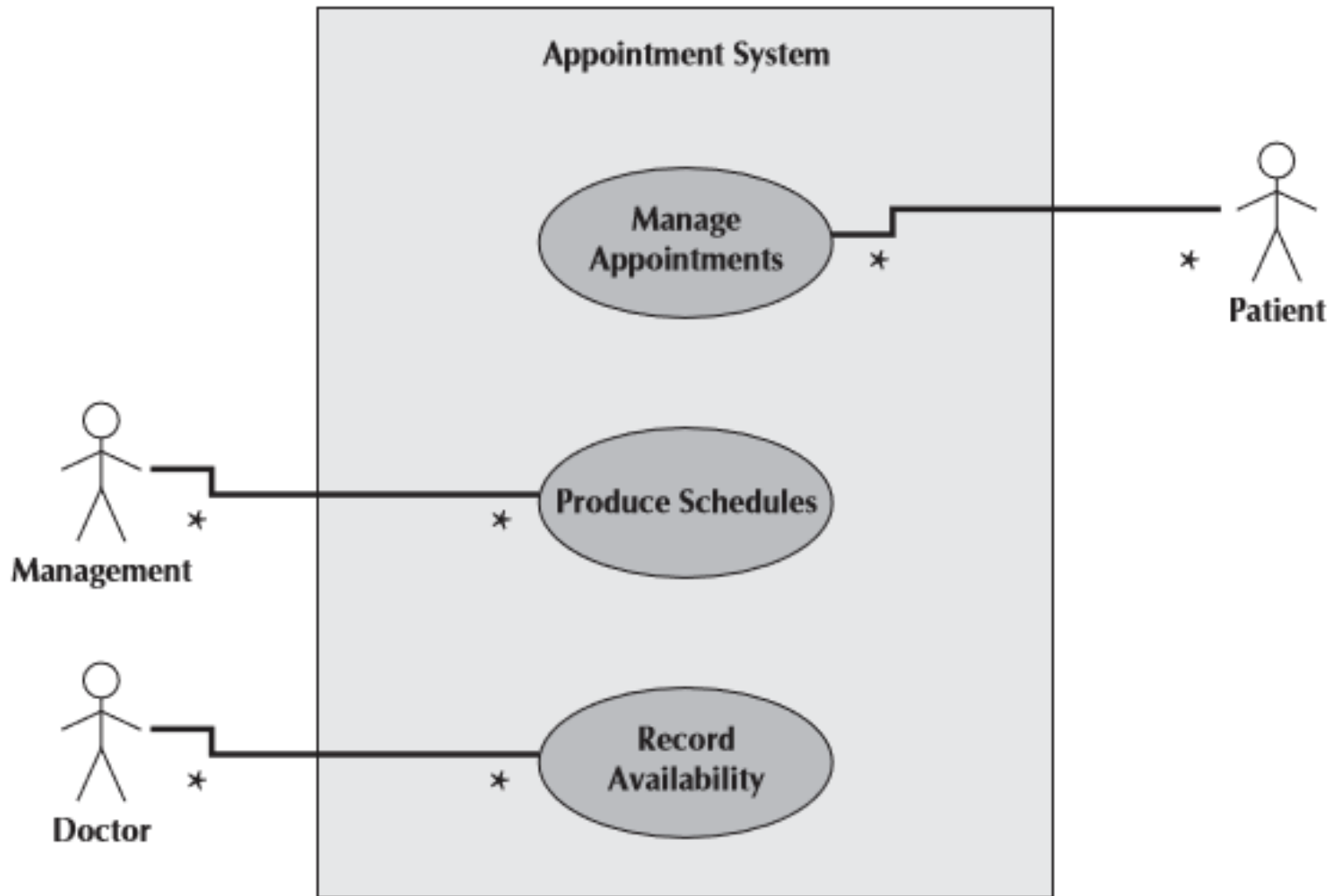
- Is labeled with a descriptive verb-noun phrase

Use Case

Make appointment

Produce schedule information
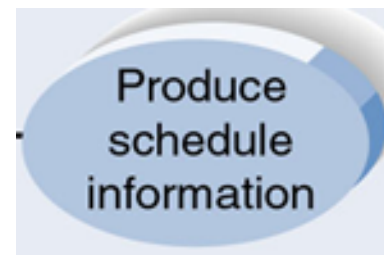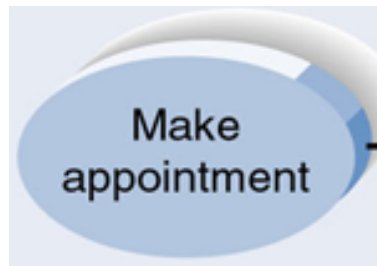
System boundary - Association relationship

- ## System boundary

  - Includes the name of the system inside or on top

  - Represents scope of the system

  ```
  System
  ```
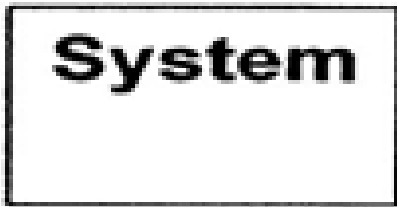
- ## Association relationship
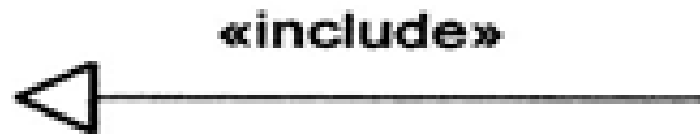
  - Links an actor with the use case
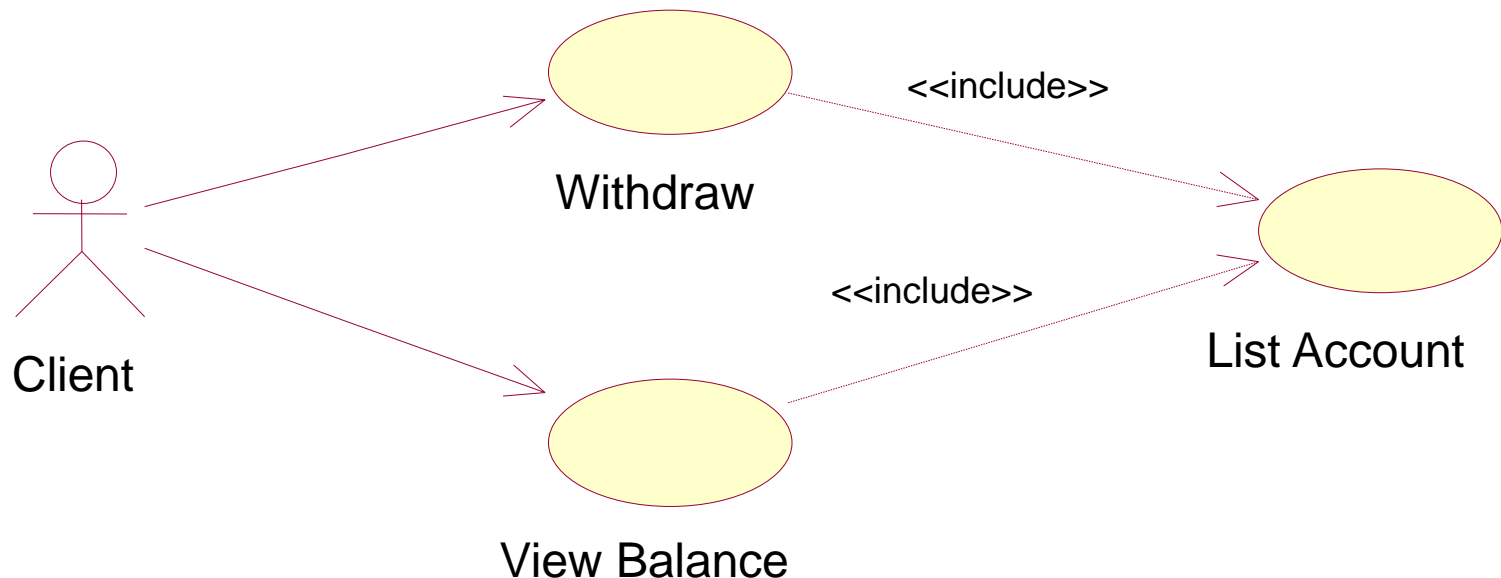
Use case analysis

Include relationship

- Represents the inclusion of the functionality of one use case with in another

- The arrow is drawn from the based use case to the used use case

«include»

<<include>>

Include relationship

- Represents the inclusion of the functionality of one use case with in another

- The arrow is drawn from the based use case to the used use case
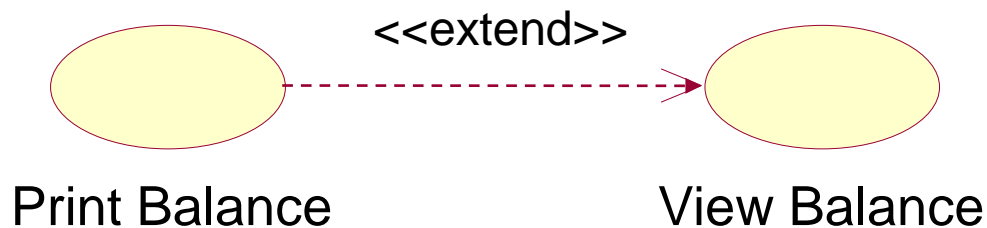
Extend relationship

- Represents the extension of the use case to include optional behavior

- The arrow is drawn from the extension use case to the based use case

«extend»

<<extend>>

Extend relationship

- Represents the extension of the use case to include optional behavior

- The arrow is drawn from the extension use case to the based use case
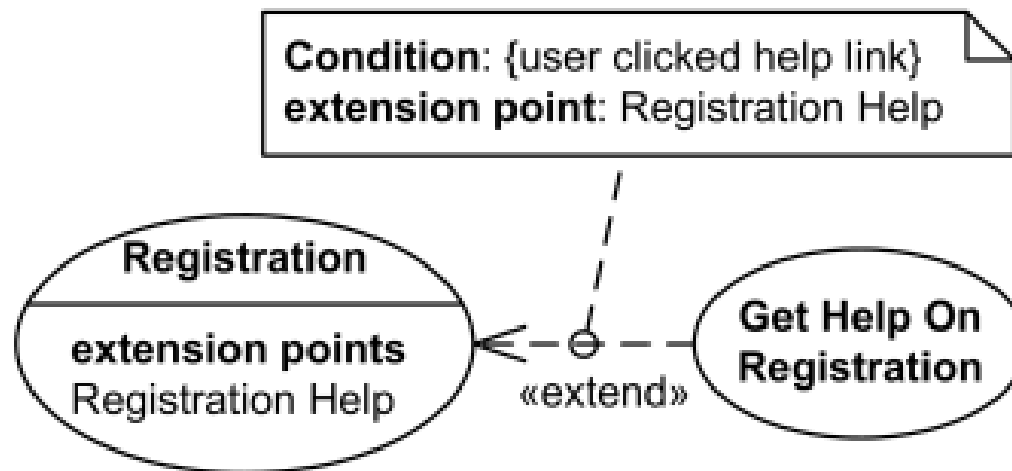


```
          <<extend>>
( Print Balance )  -  -  - >  ( View Balance )
```

Print Balance   View Balance
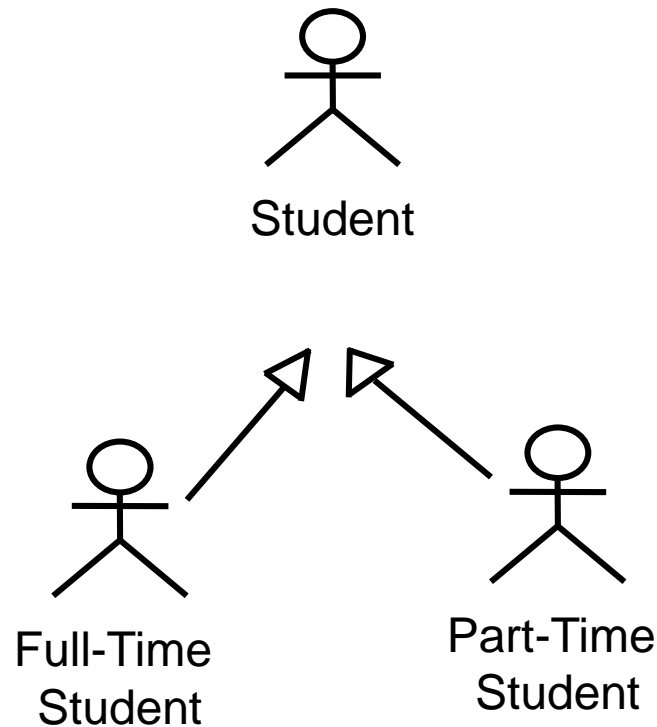
Extend relationship

Registration use case is complete and meaningful on its own.
It could be extended with optional **Get Help On Registration** use case.

Registration use case is conditionally extended by Get Help On Registration use case in extension point Registration Help.
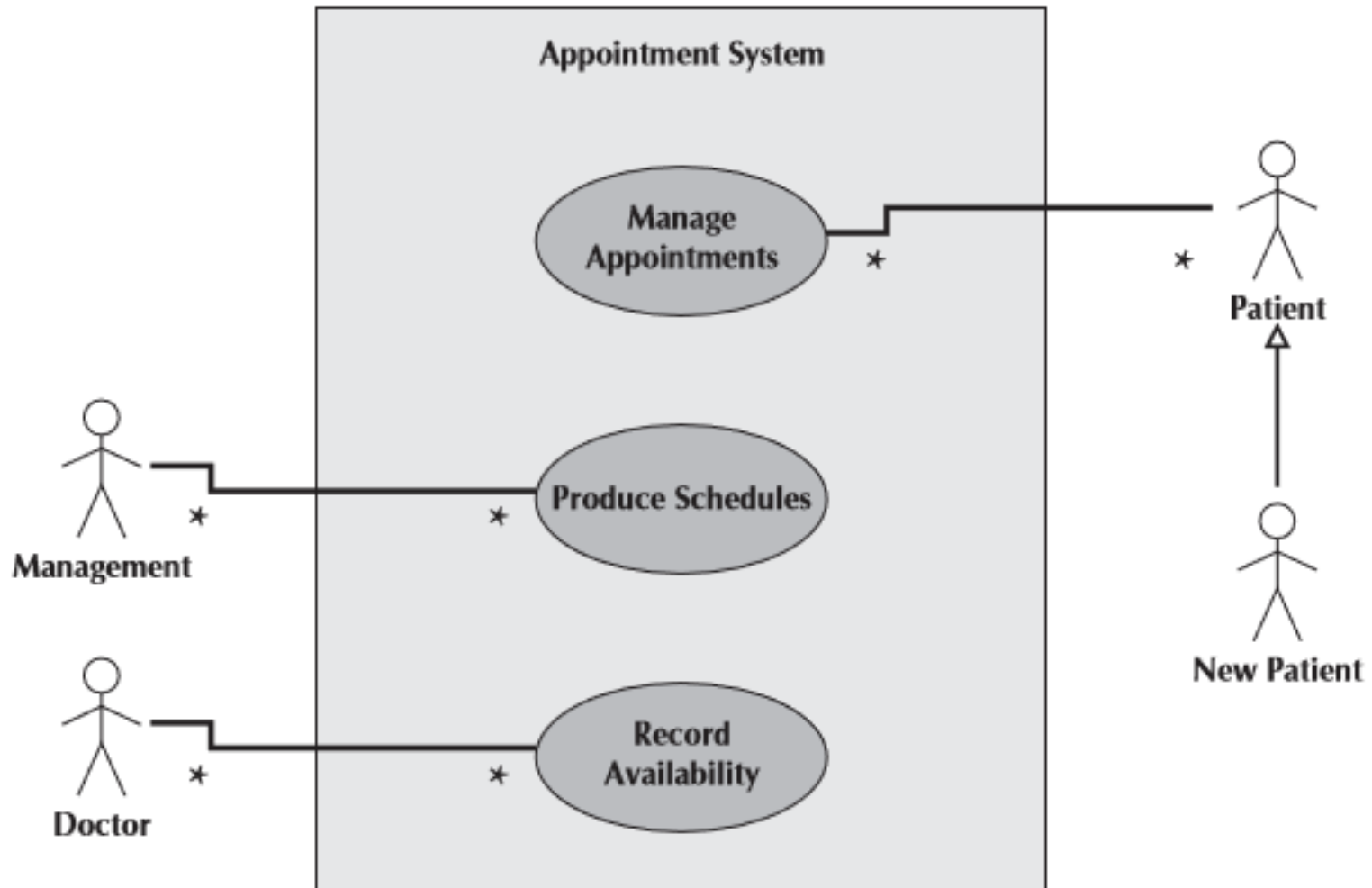
Generation relationship

- Represents a specialized use case to a more gereralized one. ◁──────────────

Student

Full-Time Student

Part-Time Student

Generation relationship

- Use case with specialized actor

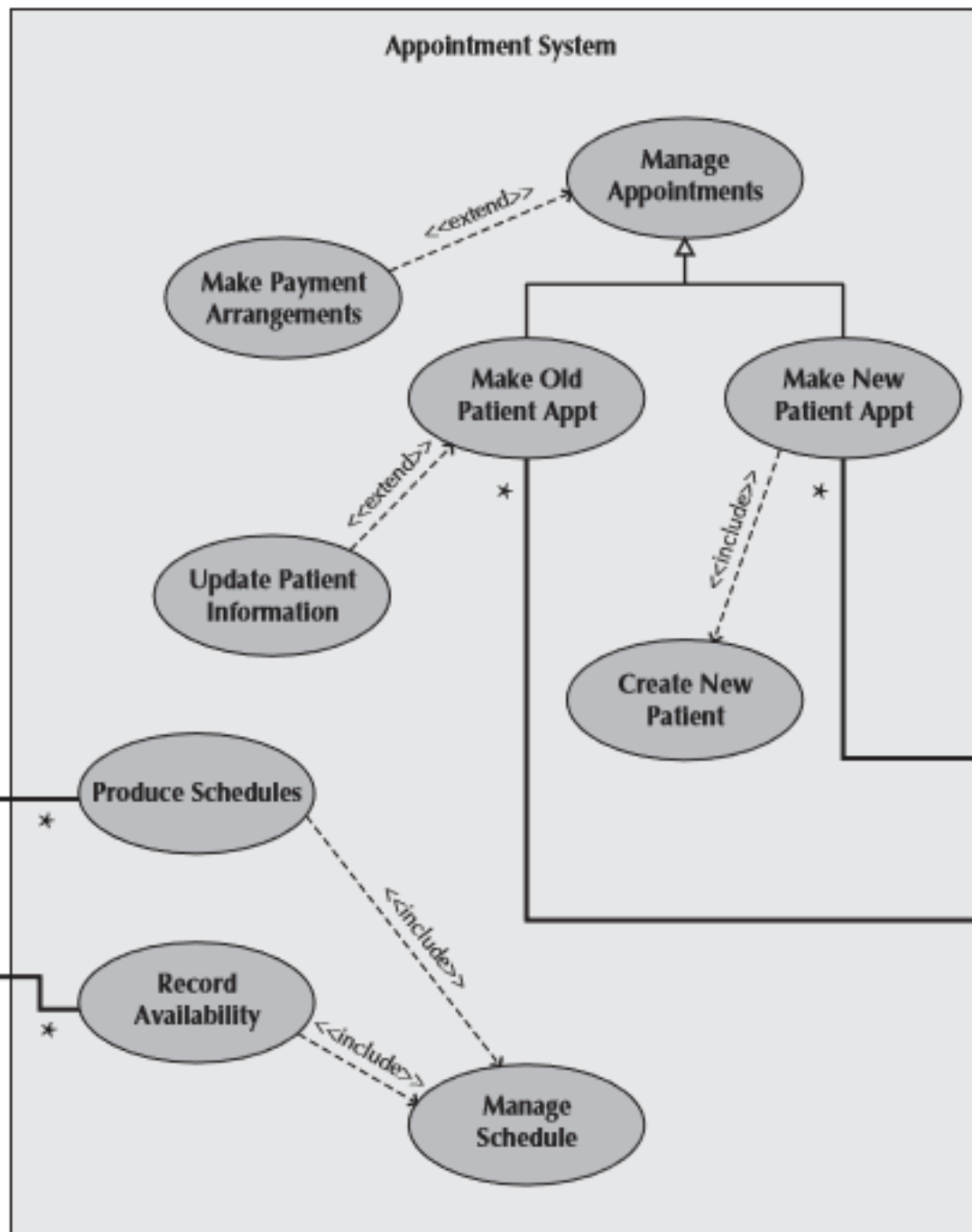# USE CASE MODEL

Use case model

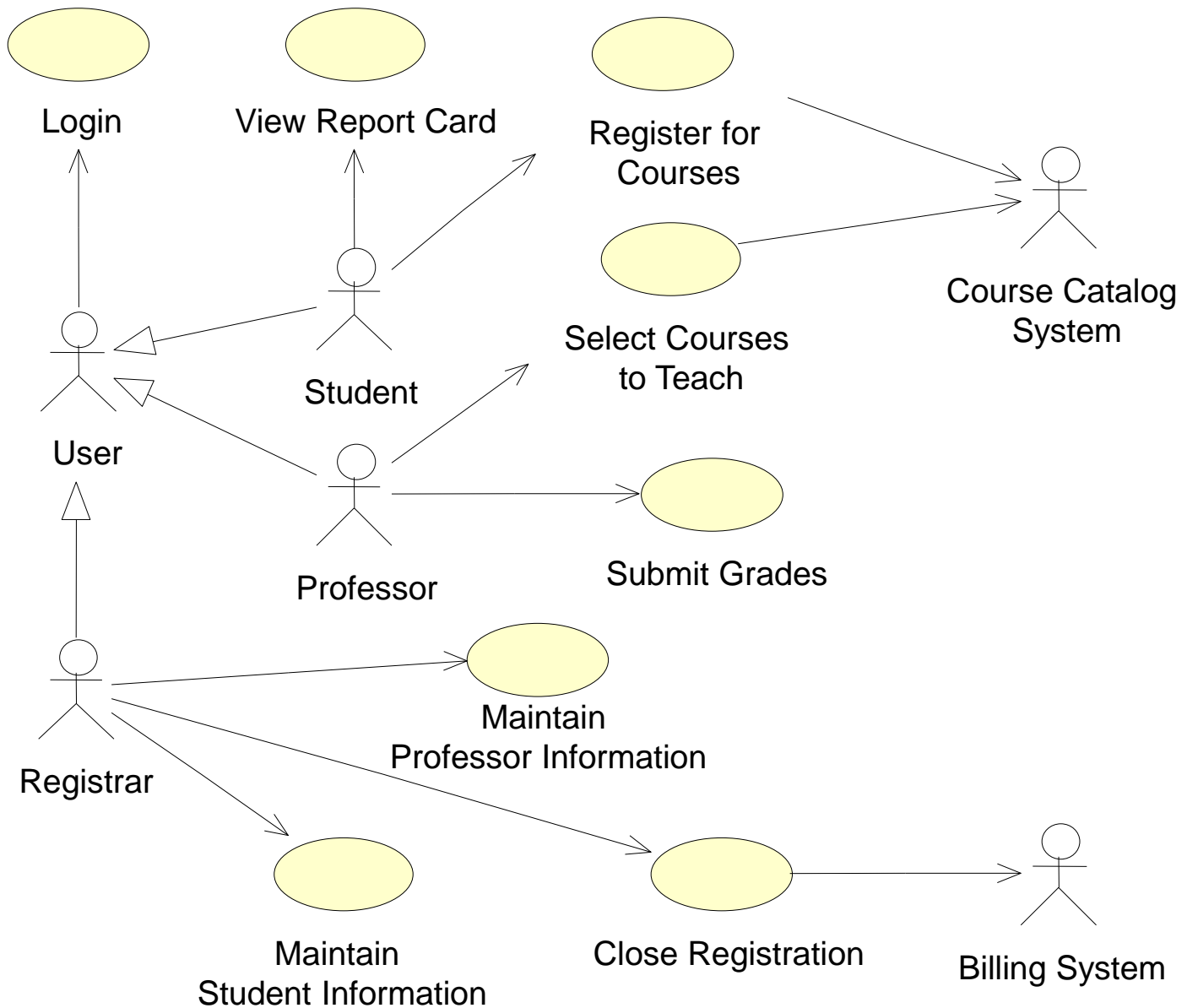- Use-case model / diagrams present a graphical overview of the main functionality of a system.

Appointment System

- Manage Appointments
- Make Payment Arrangements — <<extend>> → Manage Appointments
- Make Old Patient Appt
- Make New Patient Appt
- Update Patient Information — <<extend>> → Make Old Patient Appt
- Create New Patient — <<include>> → Make New Patient Appt
- Produce Schedules
- Record Availability
- Manage Schedule
- Produce Schedules — <<include>> → Manage Schedule
- Record Availability — <<include>> → Manage Schedule

Actors: Management, Doctor, Patient, New Patient, Old Patient

Login

View Report Card

Register for
Courses

Course Catalog
System

Student

Select Courses
to Teach

User

Professor

Submit Grades

Registrar

Maintain
Professor Information

Maintain
Student Information

Close Registration

Billing System

subject, business boundary

business actor

association

business use case

«Business»
Airport

Group Check-In

include relationship

Tour Guide

«include»

generalization between actors

Individual Check-In

extend relationship

1..*

«extend»

business actor

Baggage Check-In

1..*

Passenger
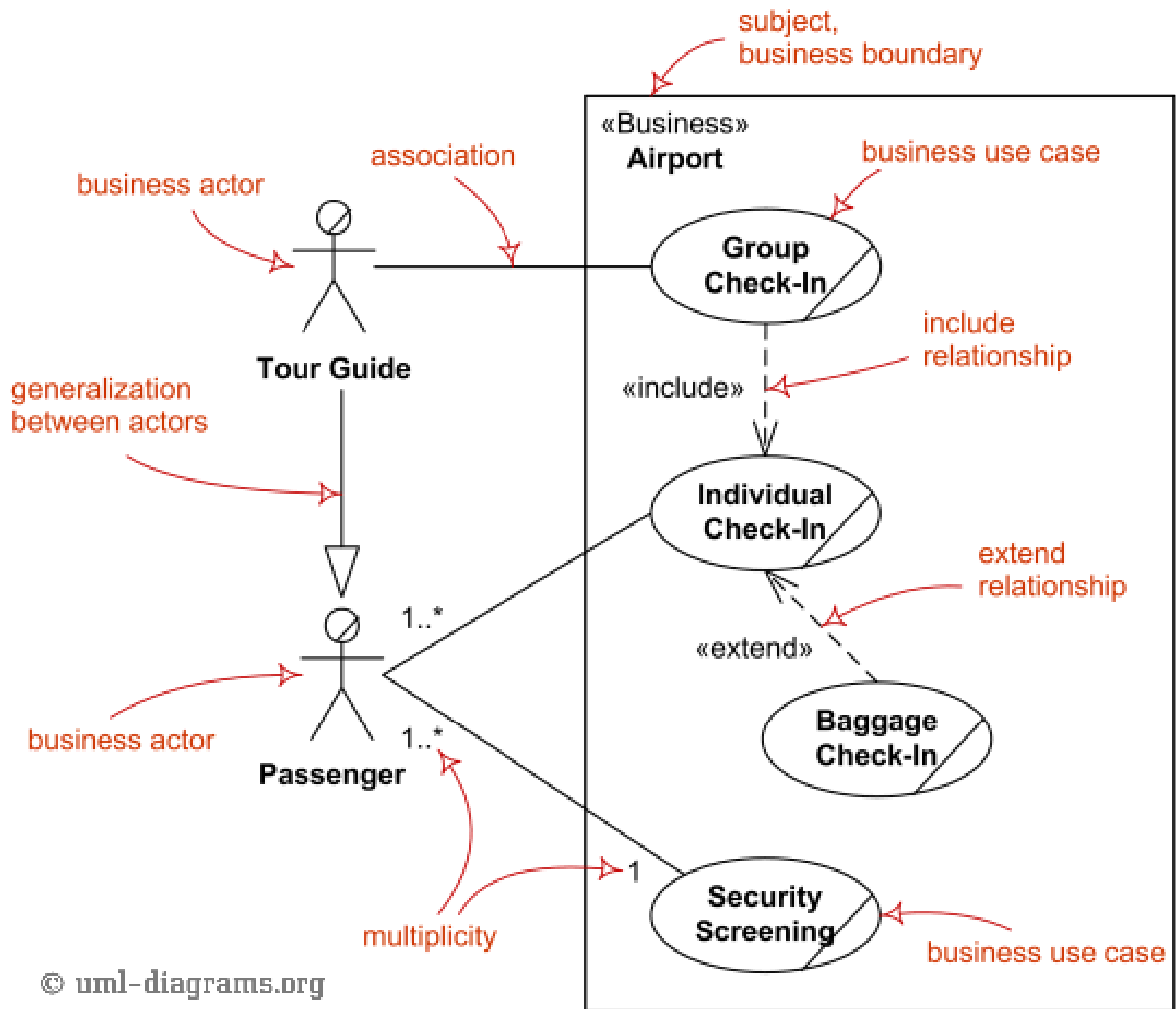
multiplicity

1

Security Screening
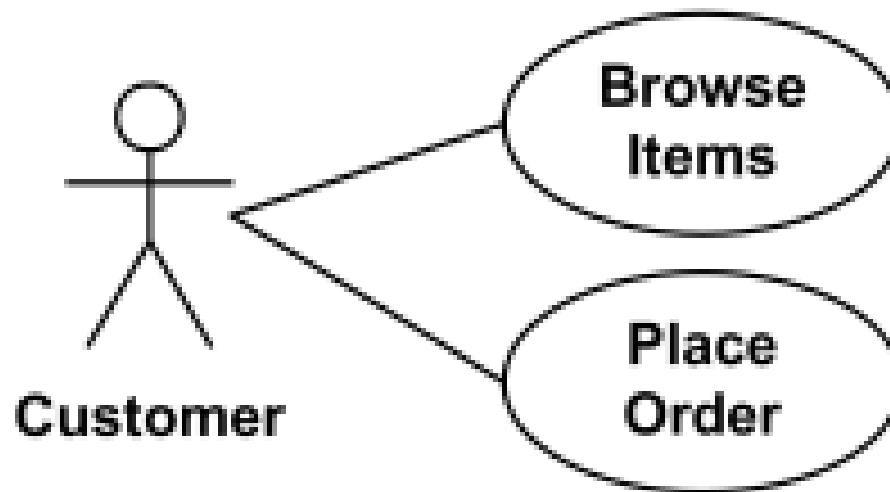
business use case

© uml-diagrams.org

Association Between Actor and Use Case

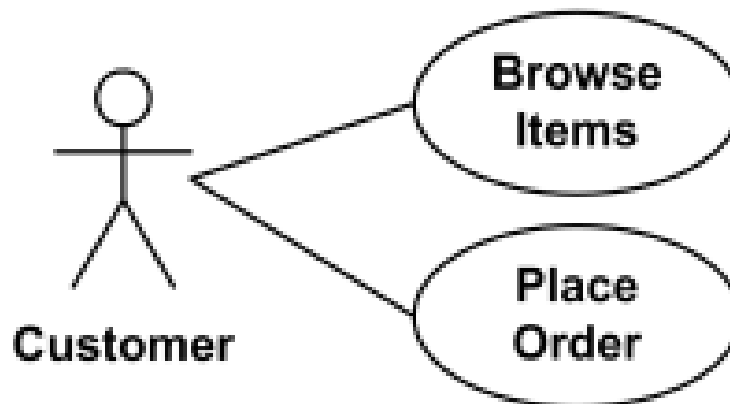- An actor could be associated to one or several use cases.

Association Between Actor and Use Case

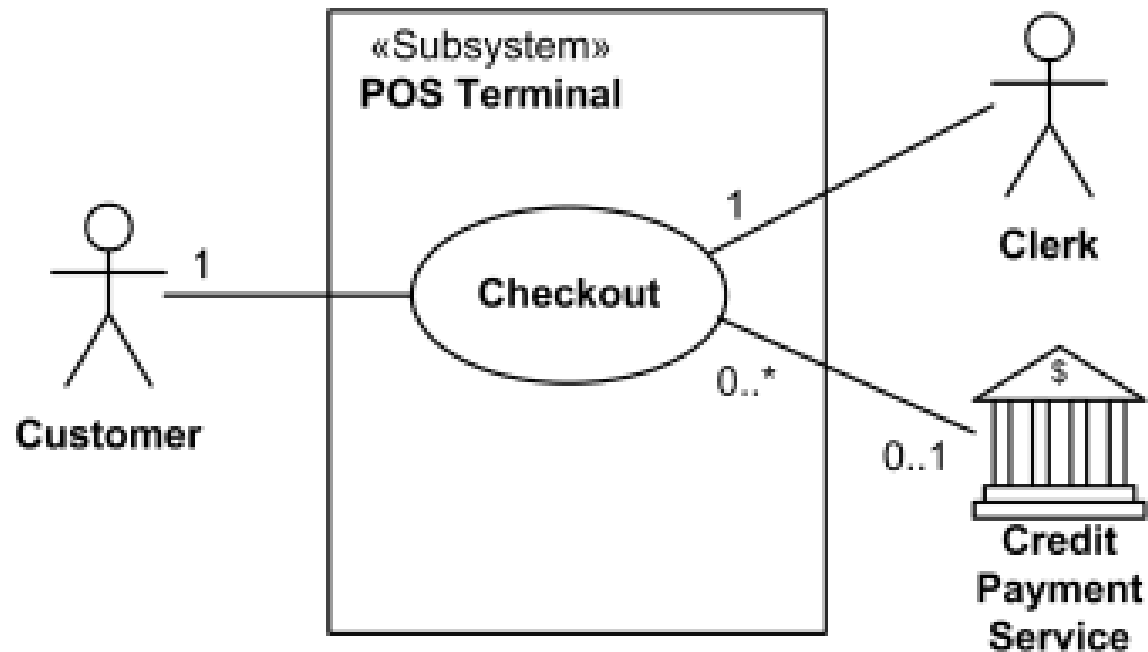- A use case may have one or several associated actors

Association Between Actor and Use Case

- Multiplicity of an Actor, optional
- Value: 0. 1. 2. *. 2..5
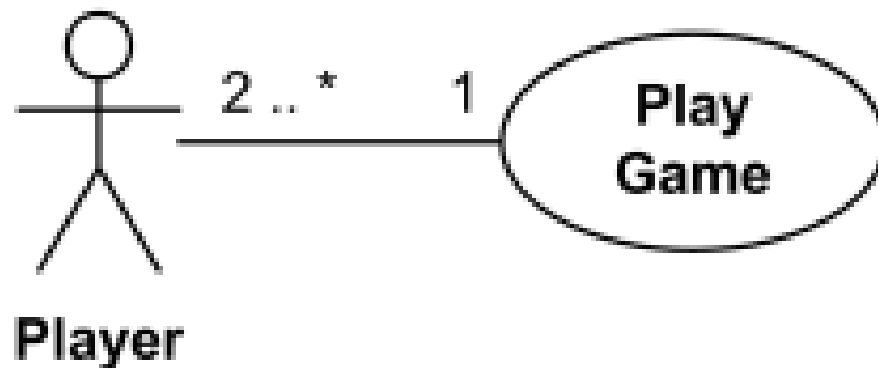
Association Between Actor and Use Case

- Multiplicity of an Actor, optional
- Value: 0, 1, 2, *, 2..5

- *Checkout use case requires Customer actor, hence the 1 multiplicity of Customer.*
- *The use case may not need (Credit Payment Service) thus the 0..1 multiplicity.*

Association Between Actor and Use Case

- Multiplicity of an Actor, optional
- Value: 0, 1, 2, *, 2..5

- *2 actor instances (Players) are involved in the "Play Game" use case.*

The process used to create use cases and use case diagrams

The process used to create use cases and use case diagrams

- ✓ Use-case Descriptions
- ✓ Steps for creating Use-case Diagrams

Use-case Descriptions

1. Use case name
2. Primary actor
3. Stakeholders are interests
4. Brief description
5. Trigger
6. Relationship
7. Normal flow of events
8. Sub-flows
9. Alternate/exceptional flow

Use-case Descriptions Example

1. Use case name: *Manage Appointment*
2. Primary actor: *Patient*
3. Stakeholders are interests:
   - ✓ *Patient – wants to make, change, or cancel an appointment*
   - ✓ *Doctor – wants to ensure patient's needs are met in a timely manner*
4. Brief description: *This use case describes how we make, changing, or cancelling an appointment.*
5. Trigger: *Patient calls and asks for a new appointment, or changing or cancelling an existing appointment*

Use-case Descriptions Example

6. Relationships:
    - ✓ Association: *Patient actor*
    - ✓ Include: *Make Payment Arrangements*
    - ✓ Extend: *Create New Patient*

Use-case Descriptions Example

7. Normal flow of events
   1. The Patient contacts the office regarding an appointment (appt.)
   2. The Patient provides the Receptionist with his or her name, address, phone number.
   3. The Receptionist validates that the Patient exists in the Patient database
   4. The Receptionist executes the Make Payment Arrangements use case
   5. The Receptionist asks Patient if he or she would like to do:
      a. If the Patient wants to make a new appointment: the S-1 (new appointment sub-flow) is performed
      b. If the Patient wants to cancel an existing appointment: the S-2 (cancel appointment sub-flow) is performed
      c. If the Patient wants to change an existing appointment: the S-3 (change appointment sub-flow) is performed

Use-case Descriptions Example

8. Subflows

S-1: New appointment

1. The Receptionist asks the Patient for possible times

2. The Receptionist matches the Patient's desired appointment times with available times and schedules new appointment

S-2: Cancel appointment

1. The Receptionist asks the Patient for the appointment time

2. The Receptionist finds the current appointment time in the database and cancels it

S-3: Change appointment

1. The Receptionist performs S-2: cancel appointment

2. The Receptionist performs S-1: new appointment

Use-case Descriptions Example

9. Alternate or exceptional flow

3a: The Receptionist executes the Create New Patient use case

S-1, 2a1: The Receptionist proposes some alternative appointment times based on what is available in the appointment schedule

S-1, 2a2: The Patient chooses one of the proposed times or decides not to make an appointment

The process used to create use cases and use case diagrams

- ✓ Use-case Descriptions
- ✓ Steps for creating Use-case Diagrams

Steps for creating Use-case Diagrams

- Identify major use cases

  1. Review requirements definition
  2. Identify subject's boundaries
  3. Identify primary actors and goals
  4. Identify business processes
  5. Review current set of use cases

Steps for creating Use-case Diagrams

- Creating a use case diagram

  1. Place and draw use cases

  2. Place and draw actors

  3. Draw subject's boundary

  4. Add associations

The process used to create use cases and use case diagrams

- ✓ Use-case Descriptions
- ✓ Steps for creating Use-case Diagrams