BINUS UNIVERSITY

People
Innovation
Excellence

COMP6115
Object Oriented Analysis and Design

Session #4

Business Process and Functional Modeling

BINUS UNIVERSITY

People
Innovation
Excellence

Learning Outcomes

LO1: Identify the basic concept of advance topic in Object Oriented Analysis and Design

LO2 : Use the knowledge to develop documentation for object oriented software analysis and design using Unified Modelling Language

LO3 : Analyze any problem in any software application and find out the alternative solutions using object oriented analysis and design approach

BINUS UNIVERSITY

Chapter 4: Business Process and Functional Modeling

People
Innovation
Excellence

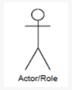
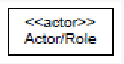
Objectives

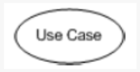
- Understand the process used to identify business processes and use cases.
- Understand the process used to create use-case diagrams
- Understand the process used to model business processes with activity diagrams.
- Understand the rules and style guidelines for activity diagrams.
- Understand the process used to create use case descriptions.
- Understand the rules and style guidelines for use case descriptions.
- Be able to create functional models of business processes using use-case diagrams, activity diagrams, and use case descriptions.

Introduction

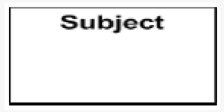
- Now begin the process of turning the requirements into functional models
  - Models are logical; i.e., independent of how they are implemented (manual or computerized)
  - Develop use-cases from the requirements
    - Use-case: how a business system interacts with its environment
    - Includes a diagram and a description to depict the discrete activities that the users perform
  - Develop activity diagrams from the use-cases
    - These model the business processes or how a business operates
    - Used to illustrate the movement of objects (data) between activities

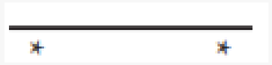
Business Process Identification With Use-Cases

- Elements of Use-Case Diagrams
  - Actors: users  or other interacting systems 

  - Associations: lines to connect actors and use-cases
    - Interactions, inclusions, extensions or generalizations
  - Use-case:  a major process in the system that gives a benefit to the users

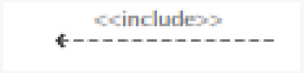
  - Subject boundary: a named box that depicts the scope of the system



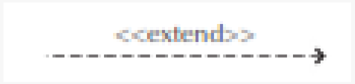
  - An association relationship: links an actor with the use case(s) with which it interacts 

Business Process Identification With Use-Cases(Cont.)

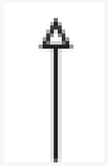
- Elements of Use-Case Diagrams
  - An include relationship: Represents the inclusion of the functionality of one use case within another



  - An extend relationship: Represents the extension of the use case to include optional behavior.



  - A generalization relationship: Represents a specialized use case to a more generalized one.



Identifying Major Use-Cases

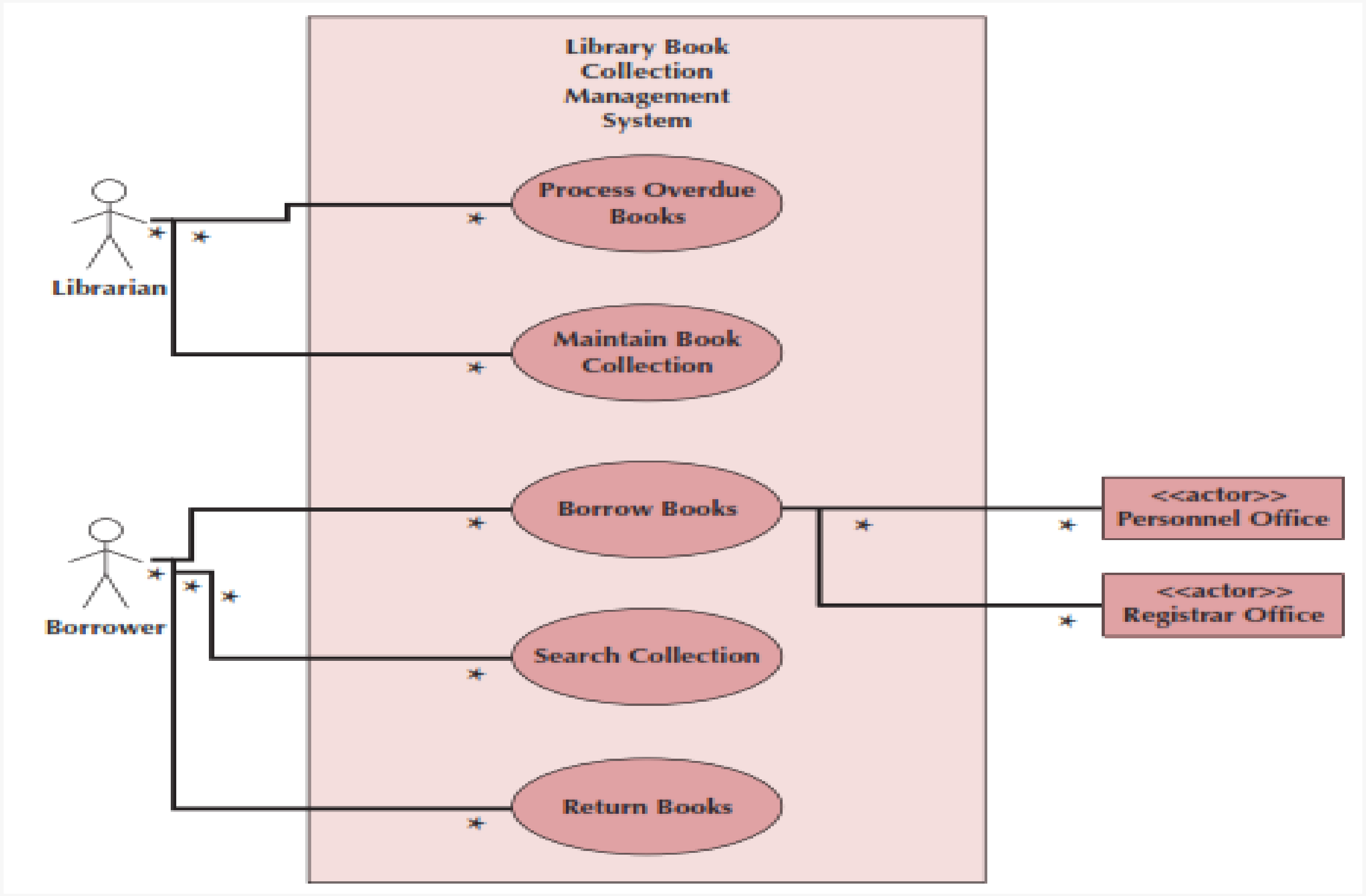
- Review the requirements definition
- Identify the subject's boundaries
- Identify the primary actors and their goals
- Identify the business processes and major use-cases
- Carefully review the current set of use-cases
  - Split or combine some to create the right size
  - Identify additional use-cases

Create a Use-Case Diagram

- Place & draw the use-cases
- Place & draw the actors
- Draw the subject boundary
- Add the associations

Example Use-Case

Library Book Collection Management System
Use Case Diagram



People
Innovation
Excellence

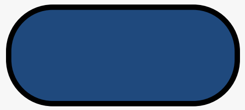
BINUS UNIVERSITY

BPM With Activity Diagrams

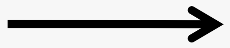
- Business processes consist of a number of activities
- Activity diagrams depict the sequence of these activities
  - Diagrams are abstract and describe processes in general
  - They model behavior independent of objects
  - Can be used for any type of process

Activity Diagram Syntax

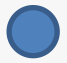
- Action or Activity
  – Represents action or set of actions



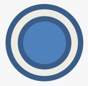
- Control Flow
  – Shows sequence of execution



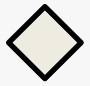
- Initial Node
  – The beginning of a set of actions



- Final Node
  – Stops all flows in an activity



- Decision Node
  – Represents a test condition



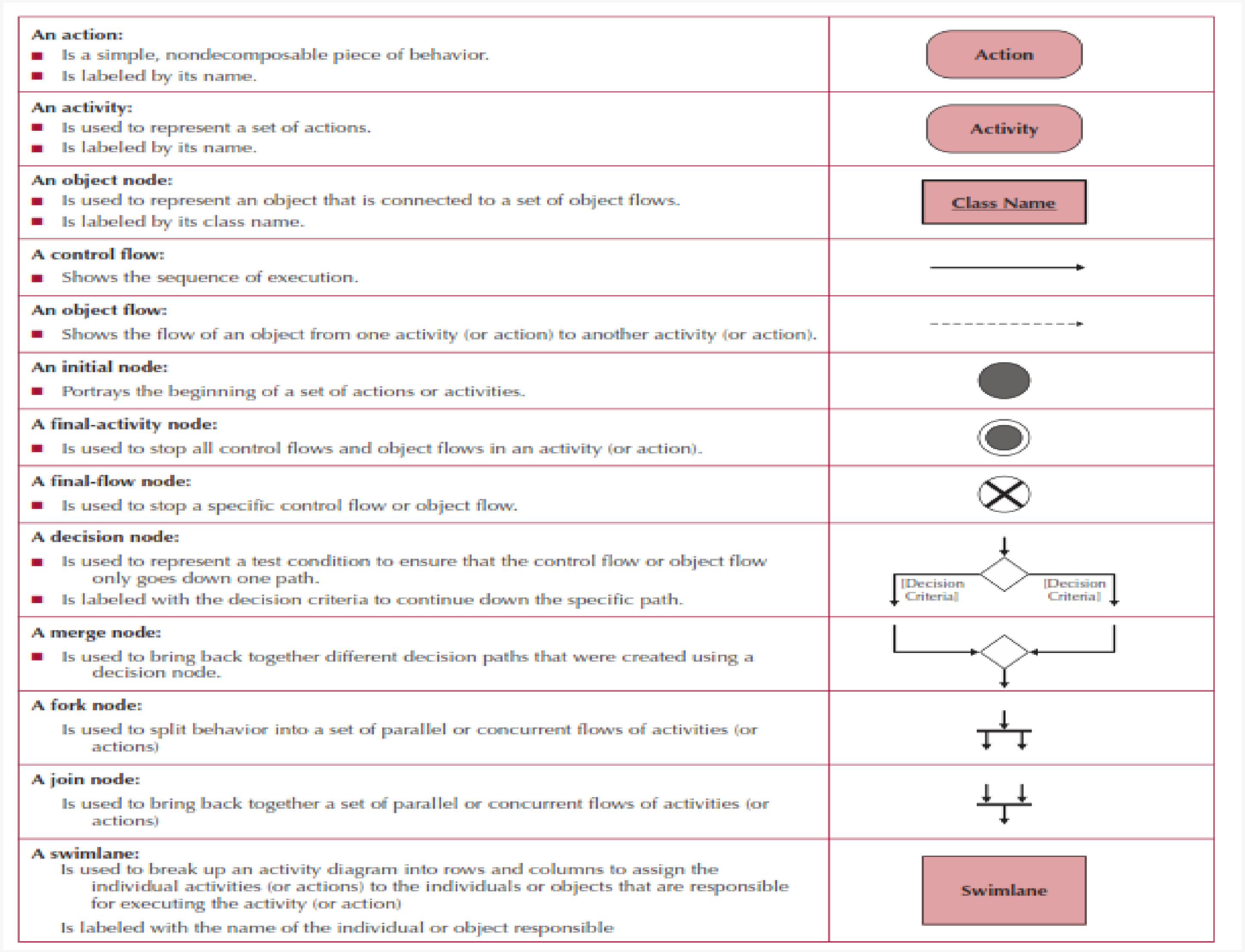
Elements of an Activity Diagram

- Actions & Activities
  - Something performed for some specific business reason
  - Named with a verb and a noun (e.g., Get Patient Information)
  - Activities can be further sub-divided; actions cannot
- Object Nodes: represent the flow of information from one activity to another
- Control Flows: model execution paths
- Object Flows: model the flow of objects
- Control Nodes: 7 types

Control Nodes

- Initial node: the beginning of the set of actions/activities
- Final-activity node: stops all actions/activities
- Final-flow node: stops one execution path but allows others to continue
- Decision node: represents a test to determine which path to use to continue (based on a guard condition)
- Merge node: rejoins mutually exclusive execution paths
- Fork node: separates a single execution path into one or more parallel paths
- Join node: rejoins parallel execution paths

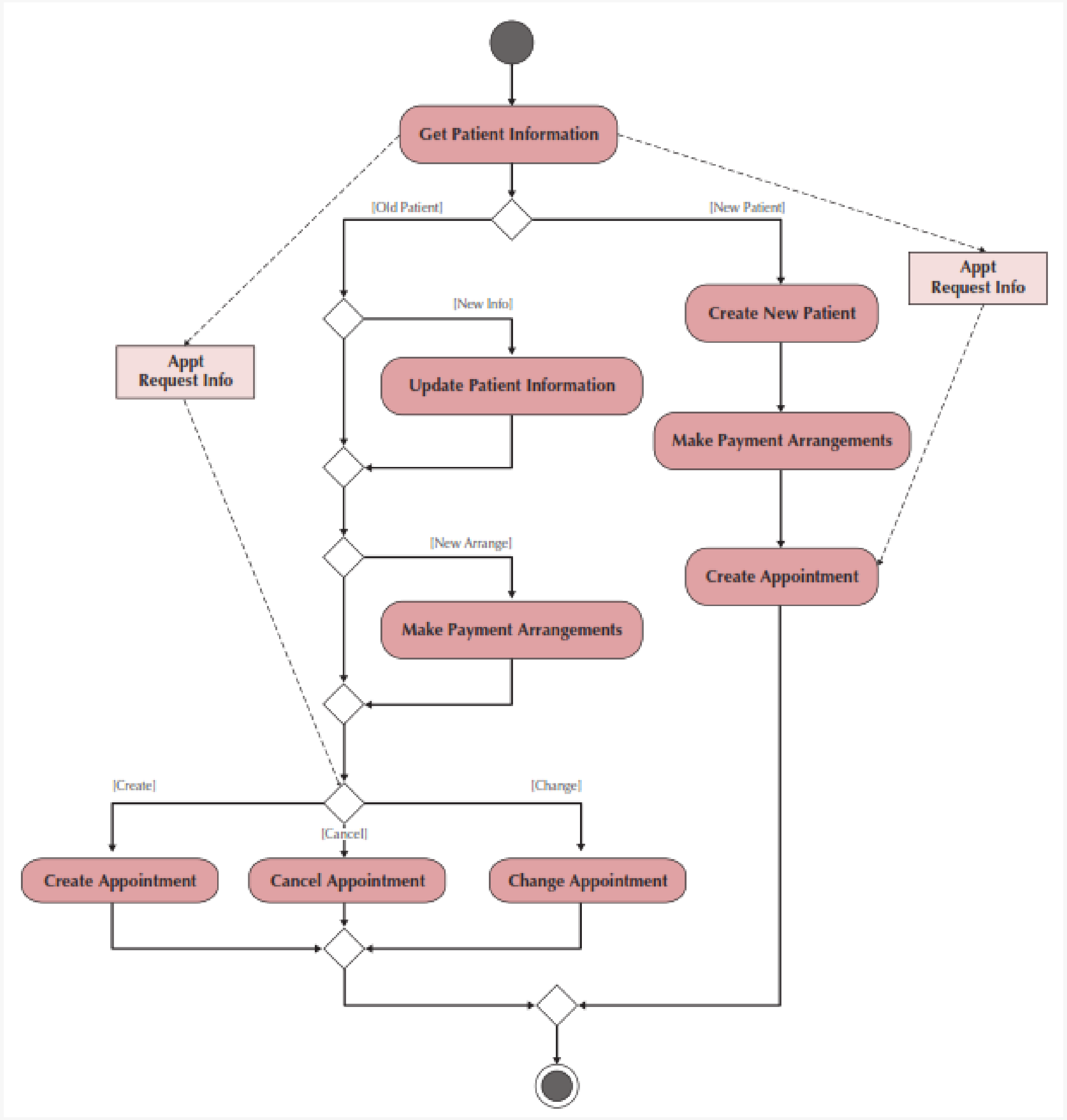
Activity Diagram Symbols



An action: ■ Is a simple, nondecomposable piece of behavior. ■ Is labeled by its name.	Action
An activity: ■ Is used to represent a set of actions. ■ Is labeled by its name.	Activity
An object node: ■ Is used to represent an object that is connected to a set of object flows. ■ Is labeled by its class name.	Class Name
A control flow: ■ Shows the sequence of execution.	→
An object flow: ■ Shows the flow of an object from one activity (or action) to another activity (or action).	- - - - - - →
An initial node: ■ Portrays the beginning of a set of actions or activities.	●
A final-activity node: ■ Is used to stop all control flows and object flows in an activity (or action).	◉
A final-flow node: ■ Is used to stop a specific control flow or object flow.	⊗
A decision node: ■ Is used to represent a test condition to ensure that the control flow or object flow only goes down one path. ■ Is labeled with the decision criteria to continue down the specific path.	◇ [Decision Criteria] [Decision Criteria]
A merge node: ■ Is used to bring back together different decision paths that were created using a decision node.	◇
A fork node: Is used to split behavior into a set of parallel or concurrent flows of activities (or actions)	⊥
A join node: Is used to bring back together a set of parallel or concurrent flows of activities (or actions)	⊤
A swimlane: Is used to break up an activity diagram into rows and columns to assign the individual activities (or actions) to the individuals or objects that are responsible for executing the activity (or action) Is labeled with the name of the individual or object responsible	Swimlane

People
Innovation
Excellence

Sample Activity Diagram



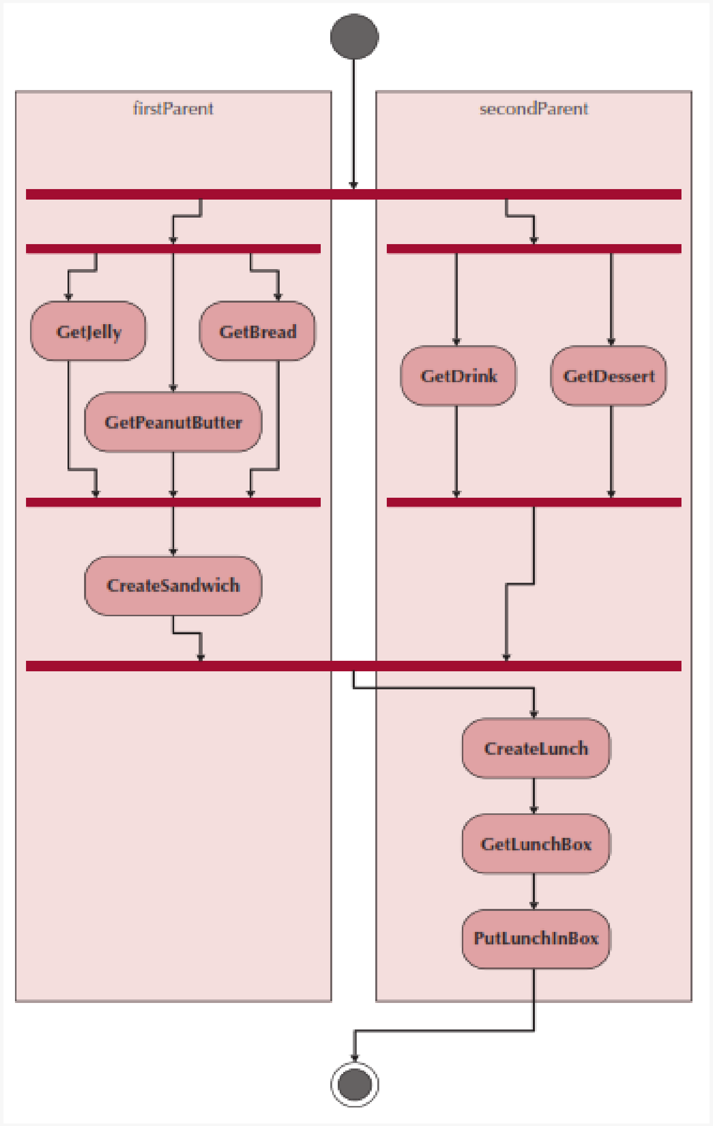
People
Innovation
Excellence

BINUS UNIVERSITY

People
Innovation
Excellence

Swim lanes

- Used to assign responsibility to objects or individuals who actually perform the activity
- Represents a separation of roles among objects
- Can be drawn horizontally or vertically



BINUS UNIVERSITY

Guidelines for Activity Diagrams

1. Set the scope of the activity being modeled
2. Identify the activities; connect them with flows
3. Identify any decisions that must be made
4. Identify potential parallelism in the process
5. Draw the activity diagram

People
Innovation
Excellence

Creating an Activity Diagram

- Choose a business process identified previously
  - Review the requirements definition and use-case diagram
  - Review other documentation collected thus far
- Identify the set of activities used in the business process
- Identify control flows and nodes
- Identify the object flows and nodes
- Lay out & draw the diagram (minimize crossing lines)

People
Innovation
Excellence

Use Cases

- The primary driver for all UML diagramming techniques
- Depicts activities performed by the users
- Describe basic functions of the system:
  - What the user can do
  - How the system responds
- Use cases are building blocks for continued design activities
- Each use-case describes 1 and only 1 function

People
Innovation
Excellence

Types of Use Cases

Purpose		Amount of information	
		Overview	Detail
	Essential	High-level **overview** of issues **essential** to understanding required functionality	**Detailed** description of issues **essential** to understanding required functionality
	Real	High-level **overview** of a specific set of steps performed on the **real** system once implemented	**Detailed** description of a specific set of steps performed on the **real** system once implemented

Elements of a Use Case Description

- Overview:
  - Name, ID Number, Type, Primary Actor, Brief Description, Importance Level, Stakeholder(s), Trigger(s)
- Relationships:
  - Association: Communication between the use case and the actors
  - Extend: Extends the functionality of a use case
  - Include: Includes another use case
  - Generalization: Allows use cases to support inheritance
- Flow of events
  - Normal flow: the usual set of activities
  - Sub-flows: decomposed normal flows to simplify the use-case
  - Alternate or exceptional flows: those not considered the norm
- Optional characteristics (complexity, time, etc.)

Use Case Writing Guidelines

1. Write in the form of subject-verb-direct object
2. Make sure it is clear who the initiator of the step is
3. Write from independent observer's perspective
4. Write at about the same level of abstraction
5. Ensure the use case has a sensible set of steps
6. Apply the KISS principle liberally.
7. Write repeating instructions after the set of steps to be repeated

People
Innovation
Excellence

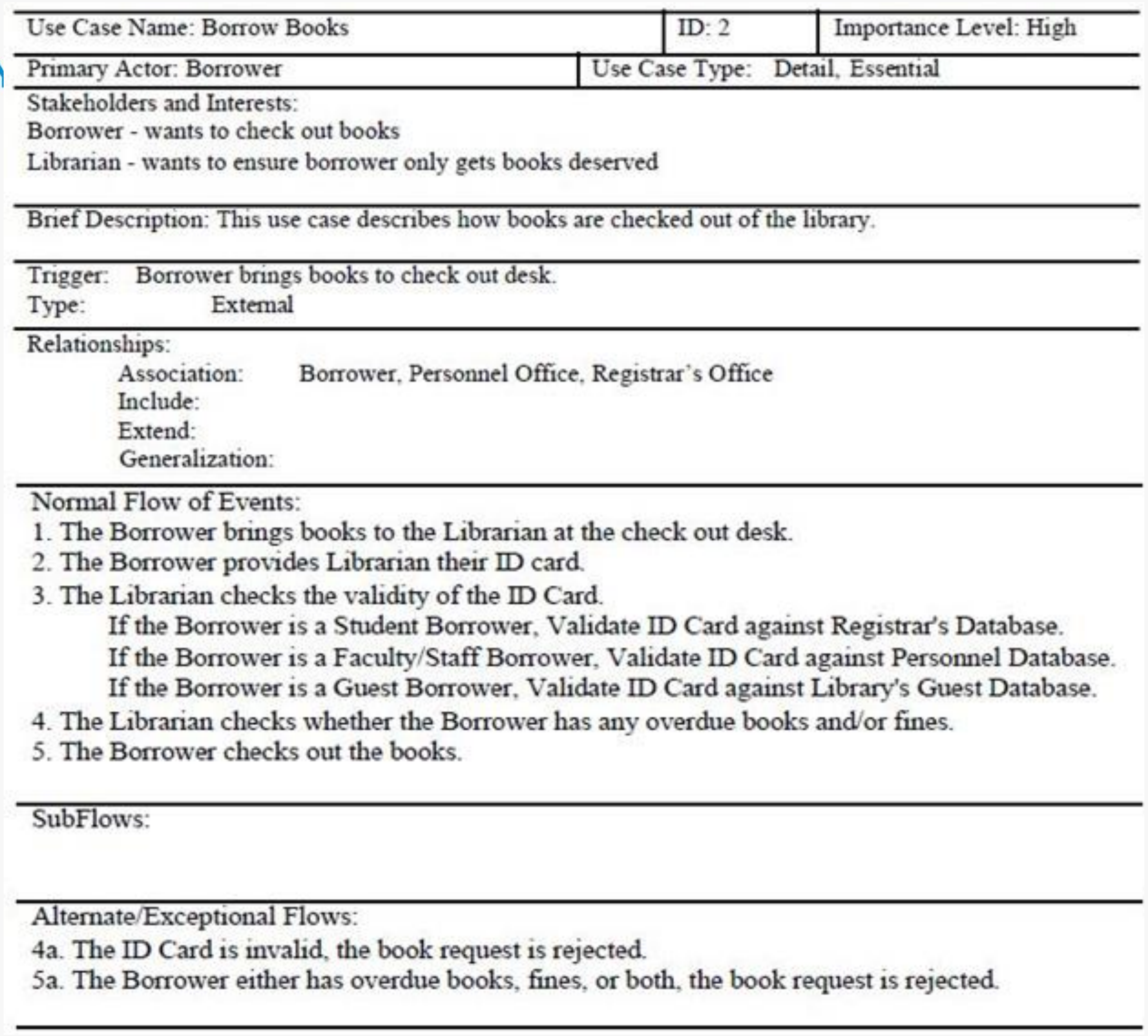
Creating Use-Case Descriptions

1. Pick a high priority use-case and create an overview:
   - List the primary actor
   - Determine its type (overview or detail; essential or real)
   - List all stakeholders and their interests
   - Determine the level of importance of the use-case
   - Briefly describe the use-case
   - List what triggers the use-case
   - List its relationship to other use-cases
2. Fill in the steps of the normal flow of events required to complete the use-case

Creating Use-Case Descriptions (cont.)

3. Ensure that the steps listed are not too complicated or long and are consistent in size with other steps
4. Identify and write the alternate or exceptional flows
5. Carefully review the use-case description and confirm that it is correct
6. Iterate over the entire set of steps again

Example Use-Case Description



Use Case Name: Borrow Books	ID: 2	Importance Level: High
Primary Actor: Borrower	Use Case Type: Detail, Essential	

Stakeholders and Interests:
Borrower - wants to check out books
Librarian - wants to ensure borrower only gets books deserved

Brief Description: This use case describes how books are checked out of the library.

Trigger: Borrower brings books to check out desk.
Type: External

Relationships:
- Association: Borrower, Personnel Office, Registrar's Office
- Include:
- Extend:
- Generalization:

Normal Flow of Events:
1. The Borrower brings books to the Librarian at the check out desk.
2. The Borrower provides Librarian their ID card.
3. The Librarian checks the validity of the ID Card.
   If the Borrower is a Student Borrower, Validate ID Card against Registrar's Database.
   If the Borrower is a Faculty/Staff Borrower, Validate ID Card against Personnel Database.
   If the Borrower is a Guest Borrower, Validate ID Card against Library's Guest Database.
4. The Librarian checks whether the Borrower has any overdue books and/or fines.
5. The Borrower checks out the books.

SubFlows:

Alternate/Exceptional Flows:
4a. The ID Card is invalid, the book request is rejected.
5a. The Borrower either has overdue books, fines, or both, the book request is rejected.

Verifying & Validating a Use-Case

- Use-cases must be verified and validated before beginning structural and behavioral modeling
- Utilize a walkthrough:
  - Perform a review of the models and diagrams created so far
  - Performed by individuals from the development team and the client (very interactive)
    - Facilitator: schedule and set up the meeting
    - Presenter: the one who is responsible for the specific representation being reviewed
    - Recorder (scribe) to take notes and especially to document errors

BINUS UNIVERSITY

People
Innovation
Excellence

Rules for Verification & Validation

1. Ensure one recorded event in the flows of the use-case description for each action/activity on the activity diagram
2. All objects in an activity diagram must be mentioned in an event of the use-case description
3. The sequence of the use-case description should match the sequence in the activity diagram
4. One and only one description for each use-case
5. All actors listed in a use-case description must be shown on the use-case diagram
6. Stakeholders listed in the use-case description may be shown on the use-case diagram (check local policy)
7. All relationships in the use-case description must be depicted on the use-case diagram
8. All diagram-specific rules must be enforced

BINUS UNIVERSITY

People
Innovation
Excellence

Summary

- Presented in this chapter:
  - The identification of business processes using use-case diagrams and descriptions
  - Modeling business processes with activity diagrams
  - How to create the documentation of use-cases and use-case descriptions
  - How to verify and validate the business processes and functional models

People
Innovation
Excellence

References

Denis, Wixom,Tegarden. (2015). Systems Analysis and Design: An Object-Oriented Approach with UML. 5th edition. ISBN: 978-1-118-80467-4, John Wiley & Sons, Inc, Denver (USA)