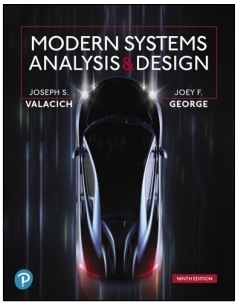
Modern Systems Analysis and Design

Ninth Edition



Appendix 7A

Object-Oriented Analysis and Design: Use Cases

Pearson

1

Learning Objectives

7A.1 explain use cases and use case diagrams and how they can be used to model system functionality

7A.2 present the basic aspects of how to create written use cases

7A.3 discuss process modeling with use cases for electronic commerce applications

Pearson

2

Introduction

- Use cases are a different way to model the functionality of a business process

Pearson

3

Use Cases (1 of 2)

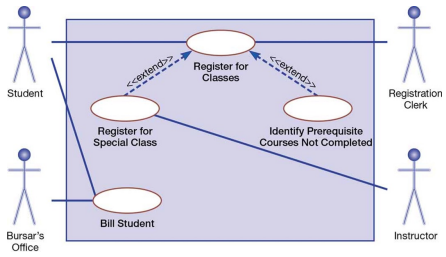
7A.1 Explain use cases and use case diagrams and how they can be used to model system functionality

- **Use case** – depiction of a system's behavior or functionality under various conditions as the system responds to requests from users
- **Actor** – external entity that interacts with a system

Pearson

4

Figure 7-26: A Use Case Diagram for a University Registration System



Pearson

5

Use Cases (2 of 2)

7A.1 Explain use cases and use case diagrams and how they can be used to model system functionality

- A use case:
  - Can be stated as a present-tense verb
  - Describes the behavior of a system in response to requests from actors
  - Do not reflect all system requirements
  - Consists of actors and use cases
  - To identify a use case, ask the following questions:
    - What are the main task performed by each actor?
    - Will the actor read or update any information in the system?
    - Will the actor have to inform the systems about changes outside the system?
    - Does the actor have to be informed of unexpected changes?

Pearson

6

Use Case Diagrams

7A.1 Explain use cases and use case diagrams and how they can be used to model system functionality

- **Use case diagram** – picture showing system behavior, along with the key actors that interact with the system
  - Initiated by an actor
  - Represents complete functionality
- **Abstract use case** – use case that is initiated by another use case

Pearson

7

Definitions and Symbols (1 of 2)

7A.1 Explain use cases and use case diagrams and how they can be used to model system functionality

- **Actor** is a role, not an individual represented by stick figures
- **Use case** is represented by an ellipse and represents a single system function
- **System boundary**, represented by a box, includes all the relevant use cases (actors are outside the system boundary)
- **Connections** are an association between an actor and a use case and is represented by arrows

Pearson

8

Definitions and Symbols (2 of 2)

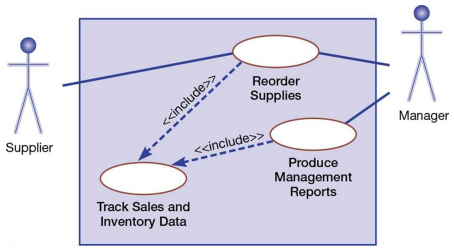
7A.1 Explain use cases and use case diagrams and how they can be used to model system functionality

- **Extend relationship** – association between two use cases where one adds new behaviors or actions to the other
- **Include relationship** – association between two use cases where one use case uses the functionality contained in the other

Pearson

9

Figure 7-27: A Use Case Diagram Featuring an Include Relationship



Pearson

10

Figure 7-29: A Template for Writing Use Cases

Use Case Title:
Primary Actor:
Level:
Stakeholders:
Precondition:
Minimal Guarantee:
Success Guarantee:
Trigger:
Main Success Scenario:
Extensions:

(**Source:** Cockburn, A. (2001). **Writing effective use cases** (1st ed.). Reprinted and Electronically reproduced by permission of Pearson Education, Inc. Upper Saddle River, New Jersey.)

Pearson

11

Written Use Cases (1 of 2)

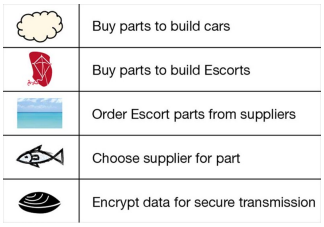
7A.2 Present the basic aspects of how to create written use cases

- **Level** – perspective from which a use case description is written, typically ranging from high level to extremely detailed
- Five suggested levels (Cockburn)
  - White: as seen from the clouds
  - Kite: still in air, but more detailed than at cloud level
  - Blue: sea level view
  - Fish: below sea level, detail increases deeper down
  - Black: bottom of the sea, maximum detail provided

Pearson

12

Figure 7-30: Use Case Levels and Detail When Moving From Top to Bottom



(**Source:** George, J. F., Batra, D., Valacich, J. S., Hoffer, J. A. (2007). **Object-oriented systems analysis and design** (2nd ed.) (pp. 174, 168, 175, 172, 176, 177).Reprinted and Electronically reproduced by permission of Pearson Education, Inc., New York, NY.)

Pearson

13

Written Use Cases (2 of 2)

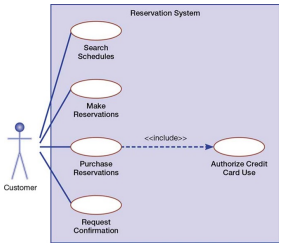
7A.2 Present the basic aspects of how to create written use cases

- **Stakeholder** – people who have a vested interest in the system being developed
- **Preconditions** – things that must be true before a use case can start
- **Minimal guarantee** – least amount promised to the stakeholder by a use case
- **Success guarantee** – what a use case must do effectively in order to satisfy stakeholders
- **Trigger** – event that initiates a use case
- **Extension** – set of behaviors or functions in a use case that follow exceptions to the main success scenario

Pearson

14

Figure 7-31: A Use Case Diagram for a Reservation System

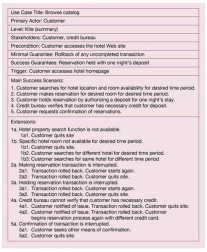


(**Source:** George, J. F., Batra, D., Valacich, J. S., & Hoffer, J. A. (2007). **Object oriented systems analysis and design** (2nd Ed.) (pp. 174, 168, 175, 172, 176, 177). Reprinted and Electronically reproduced by permission of Pearson Education, Inc., New York, NY.)

Pearson

15

Figure 7-32: Kite Level Written Use Case for Making a Hotel Room Reservation



Use Case Title: Browse catalog
Primary Actor: Customer
Level: Kite (summary)
Stakeholders: Customer, credit bureau
Precondition: Customer accesses the hotel Web site
Minimal Guarantee: Rollback of any uncompleted transaction
Success Guarantee: Reservation held with one night's deposit
Trigger: Customer accesses hotel homepage

Main Success Scenario:
1. Customer searches for hotel location and room availability for desired time period.
2. Customer makes reservation for desired room for desired time period.
3. Customer holds reservation by authorizing a deposit for one night's stay.
4. Credit bureau verifies that customer has necessary credit for deposit.
5. Customer requests confirmation of reservations.

Extensions:
1a. Hotel property search function is not available.
1a1. Customer quits site
1b. Specific hotel room not available for desired time period.
1b1. Customer quits site.
1b2. Customer searches for different hotel for desired time period.
1b3. Customer searches for same hotel for different time period
2a. Making reservation transaction is interrupted.
2a1. Transaction rolled back. Customer starts again.
2a2. Transaction rolled back. Customer quits site.
3a. Holding reservation transaction is interrupted.
3a1. Transaction rolled back. Customer starts again.
3a2. Transaction rolled back. Customer quits site.
4a. Credit bureau cannot verify that customer has necessary credit.
4a1. Customer notified of issue. Transaction rolled back. Customer quits site.
4a2. Customer notified of issue. Transaction rolled back. Customer begins reservation process again with different credit card.
5a. Confirmation of transaction is interrupted.
5a1. Customer seeks other means of confirmation.
5a2. Customer quits site.

(**Source:** George, J. F., Batra, D., Valacich, J. S., & Hoffer, J. A. (2007). **Object oriented systems analysis and design** (2nd Ed.) (pp. 174, 168, 175, 172, 176, 177). Reprinted and Electronically reproduced by permission of Pearson Education, Inc., New York, NY.)

Pearson

16

Summary

- In Appendix A you learned how to:
  - Explain use cases and use case diagrams and how they can be used to model system functionality
  - Present the basic aspects of how to create written use cases
  - Discuss process modeling with use cases for electronic commerce applications

Pearson

17

Copyright

Pearson

18