

Chapter 13: Building Information Systems

Learning Track 1: Unified Modeling Language

Object-oriented development can be used to improve system quality and flexibility. A number of techniques for the analysis and design of object-oriented systems have been developed, but the Unified Modeling Language (UML) has become the industry standard. UML allows system builders to represent different views of an object-oriented system using various types of graphical diagrams, and the underlying model integrates these views to promote consistency during analysis, design, and implementation.

Table 12-1 provides an overview of UML and its components. “Things” are objects and “structural things” allow system builders to describe objects and their relationships. UML uses two principal types of diagrams: structural diagrams and behavioral diagrams.

TABLE 12-1 An Overall View of UML and Its Components: Things, Relationships, and Diagrams

UML Category	UML Elements	Specific UML Details	
Things	Structural Things	Classes	
		Interfaces	
		Collaborations	
		Use Cases	
		Active Classes	
	Behavioral Things	Components	
		Nodes	
		Interactions	
		State Machines	
		Grouping Things	
Relationships	Annotational Things	Packages	
		Notes	
	Structural Relationships	Dependencies	
		Aggregations	
		Associations	
		Generalizations	
		Behavioral Relationships	
	Diagrams	Structural Diagrams	Includes
			Extends
			Generalizes
Class Diagrams			
Object Diagrams			
Behavioral Diagrams		Component Diagrams	
		Deployment Diagrams	
		Use Case Diagrams	
		Sequence Diagrams	
		Collaboration Diagrams	
		Statechart Diagrams	
		Activity Diagrams	

Structural diagrams are used to describe the relationships between classes. One type of structural diagram is called a class diagram. It shows classes of employees and the relationships between them. The terminators at the end of the relationship lines in this type of diagram indicate the nature of the relationship. Generalization is a relationship between a general kind of thing and a more specific kind of thing. This type of relationship is sometimes described as a “is a relationship.” Generalization relationships are used for modeling class inheritance.

Behavioral diagrams are used to describe interactions in an object-oriented system. Figures 12-1 and 12-2 illustrate two types of behavioral diagrams: a use case diagram and a sequence diagram.

FIGURE 12-1 A UML Use-Case Diagram: Credit Card Processing



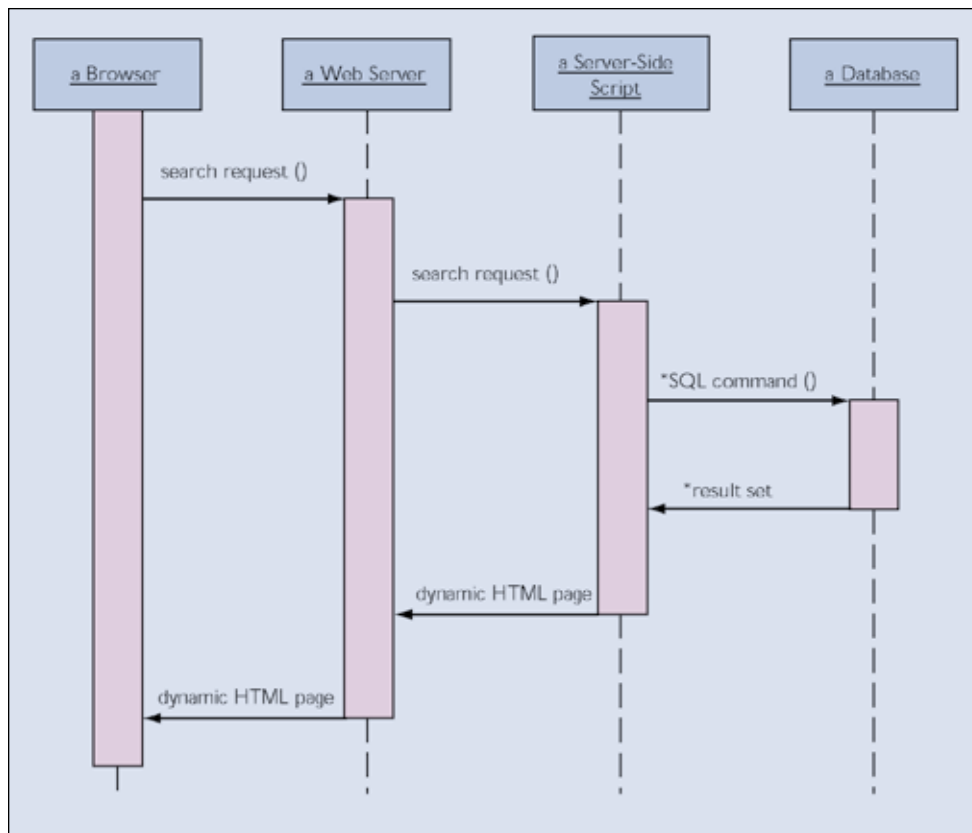
Use case diagrams model the functions of a system, showing how objects interact with each other and with the users of the system. Illustrated here is a use case diagram for credit card processing that was created with SmartDraw software.

A use case diagram shows the relationship between an actor and a system. The actor (represented in the diagram by a stick figure) is an external entity that interacts with the system, and the use case represents a series of related actions initiated by the actor to accomplish a specific goal.

continued

Several interrelated use cases are represented as ovals within a box. Use case modeling specifies the functional requirements of a system, focusing on what the system does rather than how it does it. The system's objects and their interactions with each other, and with the users of the system, are derived from the use case model.

FIGURE 12-2 A UML Sequence Diagram.



Sequence diagrams describe interactions among classes in terms of the communication between objects (messages) during a specified period of time. Illustrated here is a sequence diagram for a Web page interacting with a database that was created with SmartDraw software.

A sequence diagram describes the interactions among objects during a certain period of time. The vertical axis represents time, whereas the horizontal axis represents the participating objects and actors. Boxes along the top of the diagram represent actors and instances of objects. Lateral bars drop down from each box to the bottom of each diagram, with interactions between objects represented by arrows drawn from bar to bar. The sequence of events is displayed from top to bottom, with the first interaction at the top and the last at the bottom of the diagram. Sequence diagrams are used in system design to derive the interactions, relationships, and operations of the objects in the system.

COPYRIGHT NOTICE

Copyright © 2017 Kenneth Laudon and Jane Laudon.

This work is protected by United States copyright laws and is provided solely for the use of instructors in teaching their courses and assessing student learning. Dissemination or sale of any part of this work (including on the World Wide Web) will destroy the integrity of the work and is not permitted. The work and materials from this site should never be made available to students except by instructors using the accompanying text in their classes. All recipients of this work are expected to abide by these restrictions and to honor the intended pedagogical purposes and the needs of other instructors who rely on these materials.