

ACS2913

Software Requirements Analysis and Design

Instructor: Victor Balogun

USE CASES

Chapter 3 Outline

User Stories and Use Cases

Use Cases and the User Goal Technique

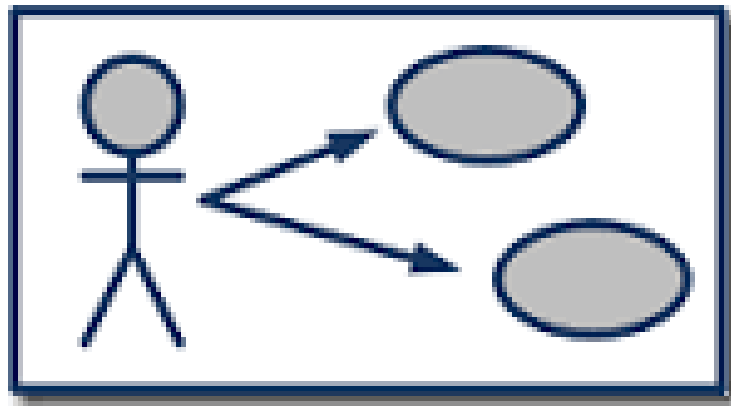
Use Cases and Event Decomposition

Use Cases in the Ridgeline Mountain Outfitters Case

Overview

We've previously provided an overview of systems analysis activities, functional and non-functional requirements, modeling, and information gathering techniques

- This chapter focuses on identifying and modeling the key aspect of functional requirements— **use cases**



User Stories

- ❖ Tools used to capture a description of a software feature from an end-user perspective
 - describes the type of user, what they want and why.
 - helps to create a simplified description of a requirement.
- ❖ One-sentence description of a work-related task done by a user to achieve some goal or result
- ❖ Acceptance Criteria identify the features that must be present at the completion of the task

User Stories

❖ The template for a user story description is:

- *As a <role> I want to <goal> so that <benefit>*

The three elements are important - knowing who the story is for helps ensure we build a useable product, what the functionality is that is needed and the value which will be derived from having that functionality enable us to make good priority decisions.

❖ The sentence for acceptance criteria is:

- *<given><when><then>*

E.g. **Given** the customer has one credit account and one savings account

When they have logged in successfully

Then the two accounts will be listed in account number order (Account no, Name, Balance, Available Funds)

Sample User Story

User Story

As a shipping clerk, I want to ship an order as accurately as possible as soon as the order details are available.

Acceptance Criteria:

- 1. Available order details must pop up on the screen when available.*
- 2. Portable display and scan device would cut time in half.*
- 3. Sort the items by bin location.*
- 4. Indicate number of items in stock for each item and mark backorder for those not available.*
- 5. Recommend shipper based on weight, size, and location.*
- 6. Print out shipping label for selected shipper.*

User Stories

Three aspects being documented on a user story:

1. Who is the functionality for? As a <user role>

◦The more specific, the better the story

2. What should be created? I want to <goal>

◦The user should fully drive this goal

3. Why is it valuable to the user? so that <benefit>

If we don't know the Who, the What and the Why, then there is not proper understanding of the story. If we don't know the story, we should not start working on it.

Use Cases

Use case— an activity that the system performs, usually in response to a request by a user

Use cases define functional requirements

Analysts decompose the system into a set of use cases (functional decomposition)

Two techniques for Identifying use cases

- User goal technique
- Event decomposition technique

Name each use case using *Verb-Noun*

User Goal Technique

This technique is the most common in industry

Simple and effective

Identify all of the potential categories of users of the system

Interview and ask them to describe the tasks the computer can help them with

Probe further to refine the tasks into specific user goals,
“I need to *Ship items, Track a shipment, Create a return*”

User Goal Technique

Some RMO CSMS Users and Goals

User	User goal and resulting use case
Potential customer	Search for item Fill shopping cart View product rating and comments
Marketing manager	Add/update product information Add/update promotion Produce sales history report
Shipping personnel	Ship items Track shipment Create item return

User Goal Technique: Specific Steps

1. Identify all the potential users for the new system
2. Classify the potential users in terms of their functional role (e.g., shipping, marketing, sales)
3. Further classify potential users by organizational level (e.g., operational, management, executive)
4. For each type of user, interview them to find a list of specific goals they will have when using the new system (current goals and innovative functions to add value)
5. Create a list of preliminary use cases organized by type of user
6. Look for duplicates with similar use case names and resolve inconsistencies
7. Identify where different types of users need the same use cases
8. Review the completed list with each type of user and then with interested stakeholders

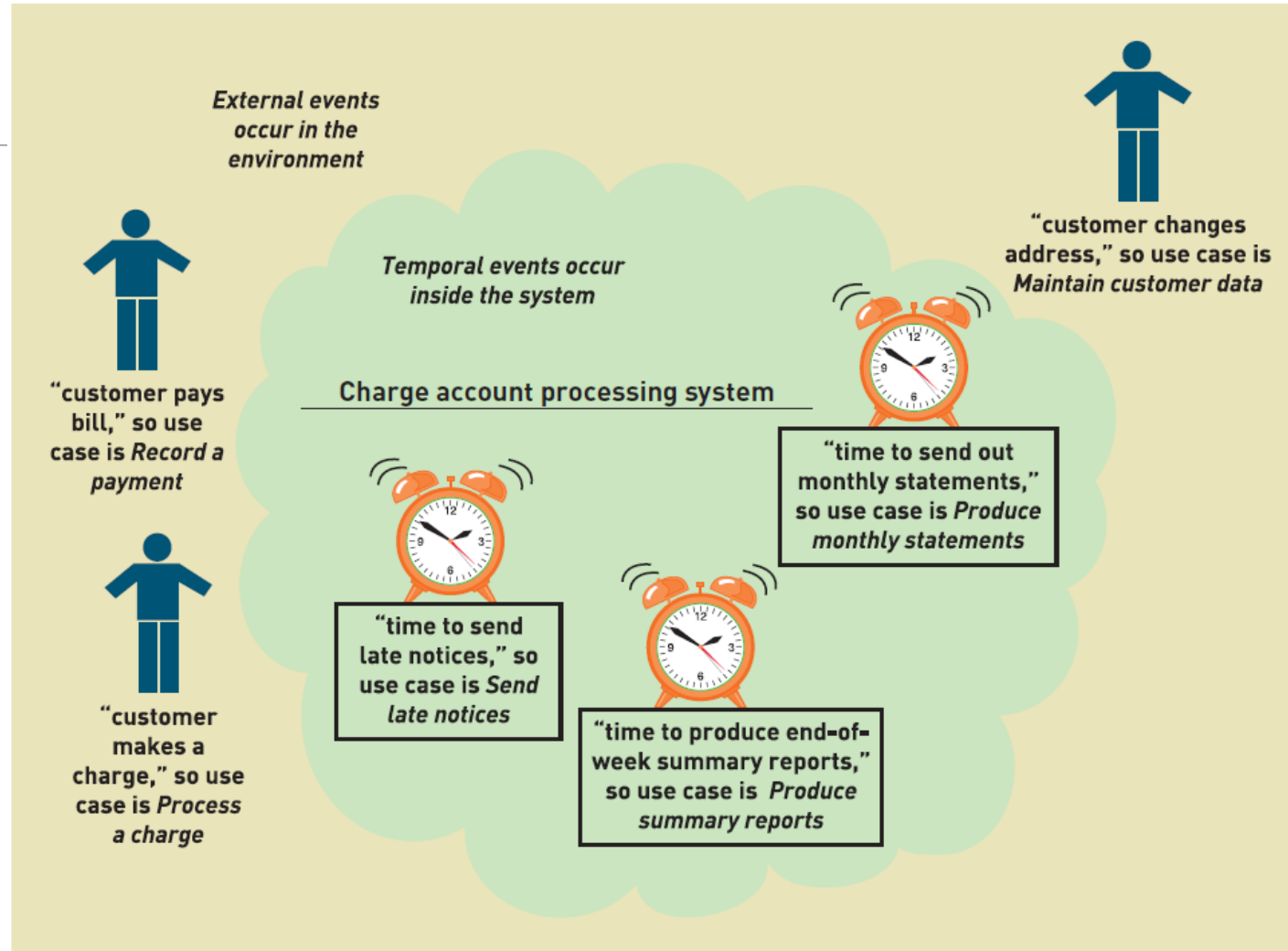
Event Decomposition Technique

More Comprehensive and Complete Technique

- Identify the events that occur to which the system must respond.
- For each event, name a use case (verb-noun) that describes what the system does when the event occurs

Event— something that occurs at a specific time and place, can be described, and should be remembered by the system

Events and Use Cases



Types of Events

External Event

- an event that occurs outside the system, usually initiated by an external agent or actor

Temporal Event

- an event that occurs as a result of reaching a point in time

State Event

- an event that occurs when something happens inside the system that triggers some process
- reorder point is reached for inventory item

External Event Checklist

	Event	Example
✓	External agent or actor wants something resulting in a transaction	Customer buys a product
✓	External agent or actor wants some information	Customer wants to know product details
✓	External data changed and needs to be updated	Customer has new address and phone
✓	Management wants some information	Sales manager wants update on production plans

Temporal Event Checklist

	Event	Examples
✓	Internal outputs needed at points in time	<ul style="list-style-type: none">• Management reports (summary or exception)• Operational reports (detailed transactions)• Internal statements and documents (including payroll)
✓	External outputs needed at points of time	<ul style="list-style-type: none">• Statements, status reports, bills, reminders

Finding the actual event that affects the system



Customer thinks about getting a new shirt



Customer drives to the mall



Customer tries on a shirt at Sears



Customer goes to Walmart

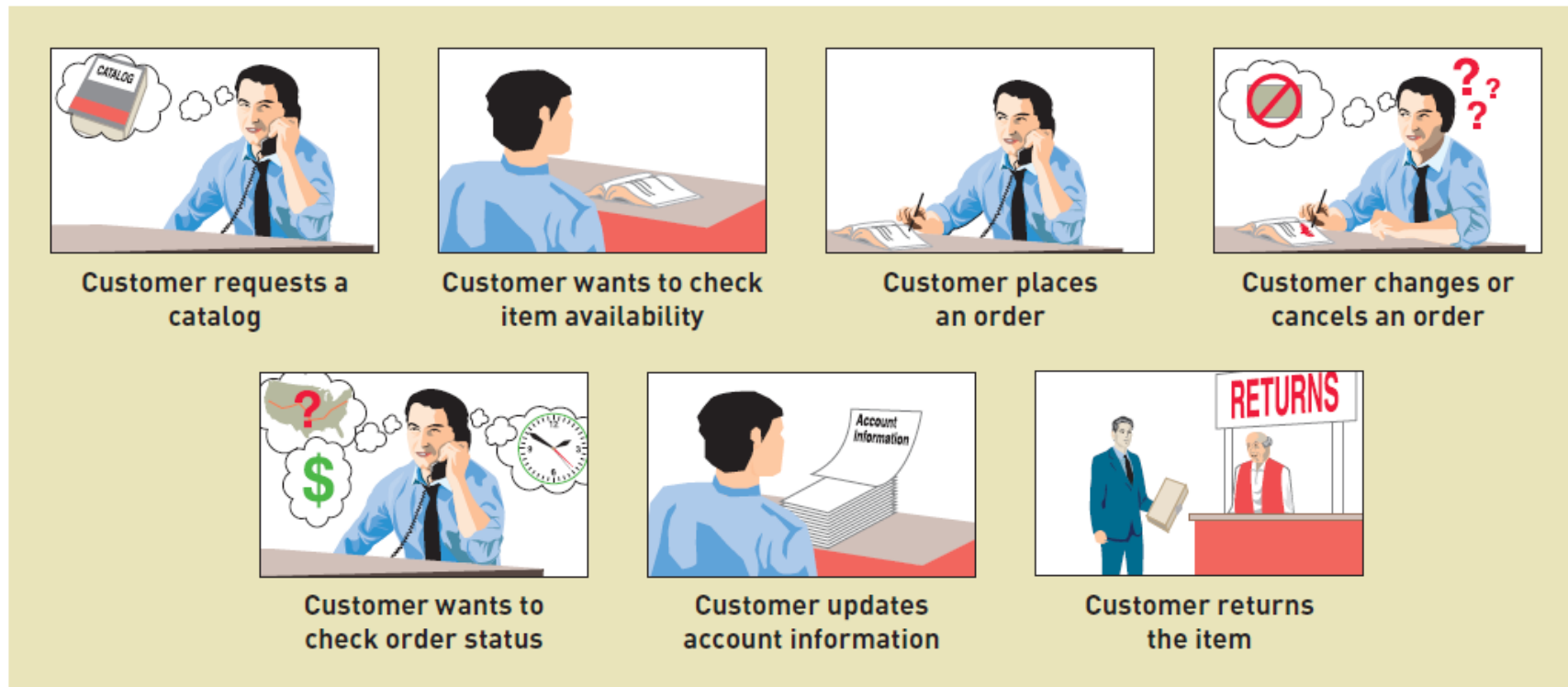


Customer tries on a shirt at Walmart



Customer buys a shirt
(the event that directly affects the system!)

Tracing a sequence of transactions resulting in many events



Event Decomposition Technique: Specific Steps

1. Consider the external events in the system environment that require a response from the system by using the checklist shown in Figure 3-3
2. For each external event, identify and name the use case that the system requires
3. Consider the temporal events that require a response from the system by using the checklist shown in Figure 3-4
4. For each temporal event, identify and name the use case that the system requires and then establish the point of time that will trigger the use case
5. Consider the state events that the system might respond to, particularly if it is a real-time system in which devices or internal state changes trigger use cases.
6. For each state event, identify and name the use case that the system requires and then define the state change.
7. When events and use cases are defined, check to see if they are required by using the perfect technology assumption. Do not include events that involve such system controls as login, logout, change password, and backup or restore the database, as these are put in later.

Event Decomposition: Recruitment system Example

An Applicant (**external**) could trigger the following events:

- Create profile
- Search for vacancies
- Submit application

For the applicant to carry out the above tasks, the system must have the following functionalities (use cases):

- Record applicant profile information
- Record vacancies
- Accept submitted application

Consider events triggered **inside** the system.
Time to update vacancies

- Time to count the number of applications received
- Time to filter applications received
- Time to forward received applications to their respective departments
- Time to produce summary reports.

The system must be able to respond with the following functionalities (use cases):

- Update Vacancies
- Count Applications
- Filter Applications
- Forward Applications
- Produce Summary Report

Event Decomposition Technique: Benefits

Events are broader than user goal: Capture temporal and state events

Help decompose at the right level of analysis: an elementary business process (EBP)

EBP is a fundamental business process performed by one person, in one place, in response to a business event

Uses **perfect technology assumption** to make sure functions that support the users work are identified and not additional functions for security and system controls – *see notes below*

Perfect Technology Assumption

What is the perfect technology assumption?

- The assumption that a system runs under perfect operating and technological conditions.
- In doing this they avoid particular tasks regarding system failure, which they will add controls later in the design process.

What are three examples of events that involve system controls that should not be included initially because of the perfect technology assumption?

- a. Backing up a database
- b. User logging into the system
- c. Restoring the database

Perfect Technology Assumption

Don't worry about functions built into system because of limits in technology and people. Wait until design.

Don't worry much about these until you are considering design issues



User wants to log on to the system



User wants to change the password



User wants to change preference settings



System crash requires database recovery



Time to back up the database



Time to require the user to change the password

Summary

Use cases are the functions identified, the activities the system carries out usually in response to a user request

Two techniques for identifying use cases are the user goal technique and the event decomposition technique

The user goal technique begins by identifying end users called actors and asking what specific goals they have when interacting with the system

The event decomposition technique begins by identifying events that occur that require the system to respond.

Three types of events include external, temporal, and state events