Professional Elective 2

EIT Elective 2 – 1

WTh 8:00 am – 9:30 am

Group no. 7

Chapter 4: Use Case Analysis

Alvarez, Ansherina

De Villar, Vince Anfernee

Dela Pena, Angeline Kassandra

Fidelino, Julian Marcus

Maramara, Sophia

Molina, Atasha Rich

Torralba, Alexander James

Prof KT V. Fortuny

Professor

Introduction

Use cases assist us in comprehending and clarifying the users' required interactions with the system and can reveal most, if not all, of the new system's functional requirements. As a result, use cases are extensively used in the analysis phase when working with users in interviews or workshop settings to discover user and functional requirements.

Use Cases

- Use cases originated as a part of the object-oriented development world but have been accepted as a useful tool regardless of the development methodology in use.
- Use cases describes what the system will do from the user's perspective.
- Use cases depicts a set of activities performed to produce some output result.
- Creation of use cases is often done as a part of interview sessions with users and as a part of JAD (Joint Application Development) sessions.

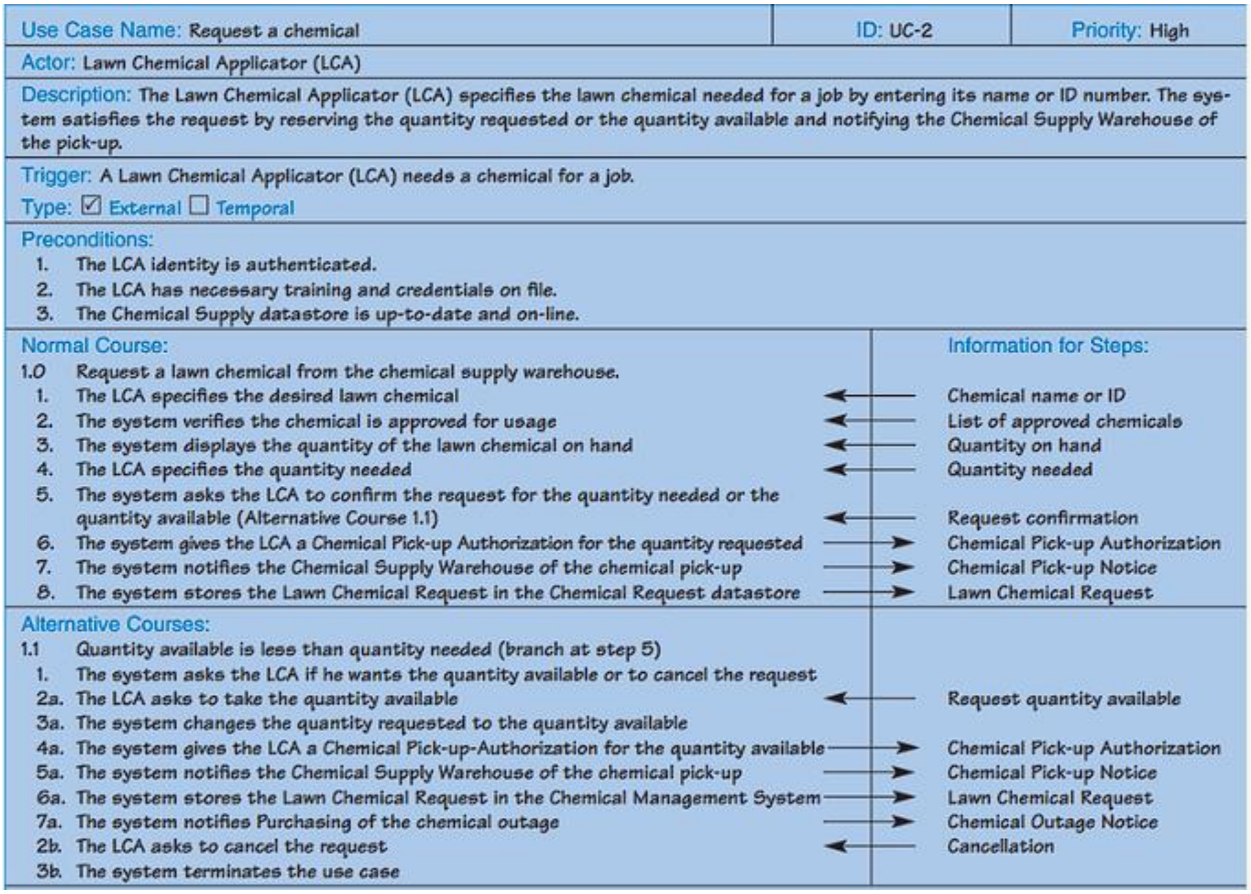
Characteristics of Use Cases

- Organize functional requirements.
- Models the goals of user-system interactions.
- Records paths called scenarios from trigger events to goals.
- Describe one main flow of events and various alternate flows.
- Multi-level.

Elements of a Use Case

- Basic Information
    - Each use case has a name and a number assigned to it. The name should be as brief as possible the number is merely a numerical sequence that is used to identify each use case.
        - Priority
        - Actor
        - Trigger

- Preconditions
  - Define the state the system must be in before the use case commences.
- Normal Course
  - The normal course lists the steps that are performed when everything flows smoothly in the system. This is sometimes called the “happy path”.
- Alternative Courses
  - Alternative courses are included to depict branches in logic that also will lead to a successful conclusion of the use case.



Use Case Name: Request a chemical	ID: UC-2	Priority: High

Actor: Lawn Chemical Applicator (LCA)

Description: The Lawn Chemical Applicator (LCA) specifies the lawn chemical needed for a job by entering its name or ID number. The system satisfies the request by reserving the quantity requested or the quantity available and notifying the Chemical Supply Warehouse of the pick-up.

Trigger: A Lawn Chemical Applicator (LCA) needs a chemical for a job.

Type: ☑ External ☐ Temporal

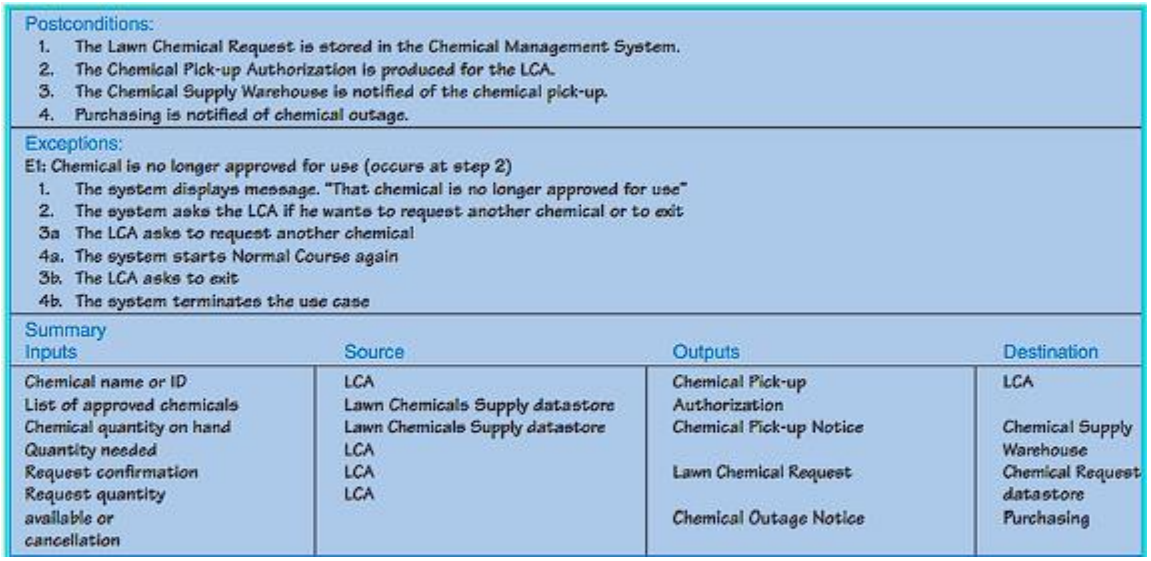
Preconditions:

1. The LCA identity is authenticated.
2. The LCA has necessary training and credentials on file.
3. The Chemical Supply datastore is up-to-date and on-line.

Normal Course:		Information for Steps:
1.0 Request a lawn chemical from the chemical supply warehouse.		
1. The LCA specifies the desired lawn chemical	←	Chemical name or ID
2. The system verifies the chemical is approved for usage	←	List of approved chemicals
3. The system displays the quantity of the lawn chemical on hand	←	Quantity on hand
4. The LCA specifies the quantity needed	←	Quantity needed
5. The system asks the LCA to confirm the request for the quantity needed or the quantity available (Alternative Course 1.1)	←	Request confirmation
6. The system gives the LCA a Chemical Pick-up Authorization for the quantity requested	→	Chemical Pick-up Authorization
7. The system notifies the Chemical Supply Warehouse of the chemical pick-up	→	Chemical Pick-up Notice
8. The system stores the Lawn Chemical Request in the Chemical Request datastore	→	Lawn Chemical Request

Alternative Courses:		
1.1 Quantity available is less than quantity needed (branch at step 5)		
1. The system asks the LCA if he wants the quantity available or to cancel the request		
2a. The LCA asks to take the quantity available	←	Request quantity available
3a. The system changes the quantity requested to the quantity available		
4a. The system gives the LCA a Chemical Pick-up-Authorization for the quantity available	→	Chemical Pick-up Authorization
5a. The system notifies the Chemical Supply Warehouse of the chemical pick-up	→	Chemical Pick-up Notice
6a. The system stores the Lawn Chemical Request in the Chemical Management System	→	Lawn Chemical Request
7a. The system notifies Purchasing of the chemical outage	→	Chemical Outage Notice
2b. The LCA asks to cancel the request	←	Cancellation
3b. The system terminates the use case		

- Postconditions
    - Post-conditions specify what the use case must do before it can be terminated. These activities may occasionally be part of the normal and exceptional event flows.
- Exceptions
    - Are error conditions encountered while performing use case steps.
- Summary Inputs and Outputs
    - The use case's final section summarizes the major inputs and outputs to the use case's steps. Each of the use case's major inputs and outputs, as well as their source and destination, are listed.
- Additional Use Case Issues
    - Additional sections on use case forms may be included by some organizations. It may be useful to include sections devoted to:
        - Frequency of use
        - Business Rules
        - Special Requirements
        - Assumptions
        - Notes and Issues



Postconditions:

1. The Lawn Chemical Request is stored in the Chemical Management System.
2. The Chemical Pick-up Authorization is produced for the LCA.
3. The Chemical Supply Warehouse is notified of the chemical pick-up.
4. Purchasing is notified of chemical outage.

Exceptions:

E1: Chemical is no longer approved for use (occurs at step 2)

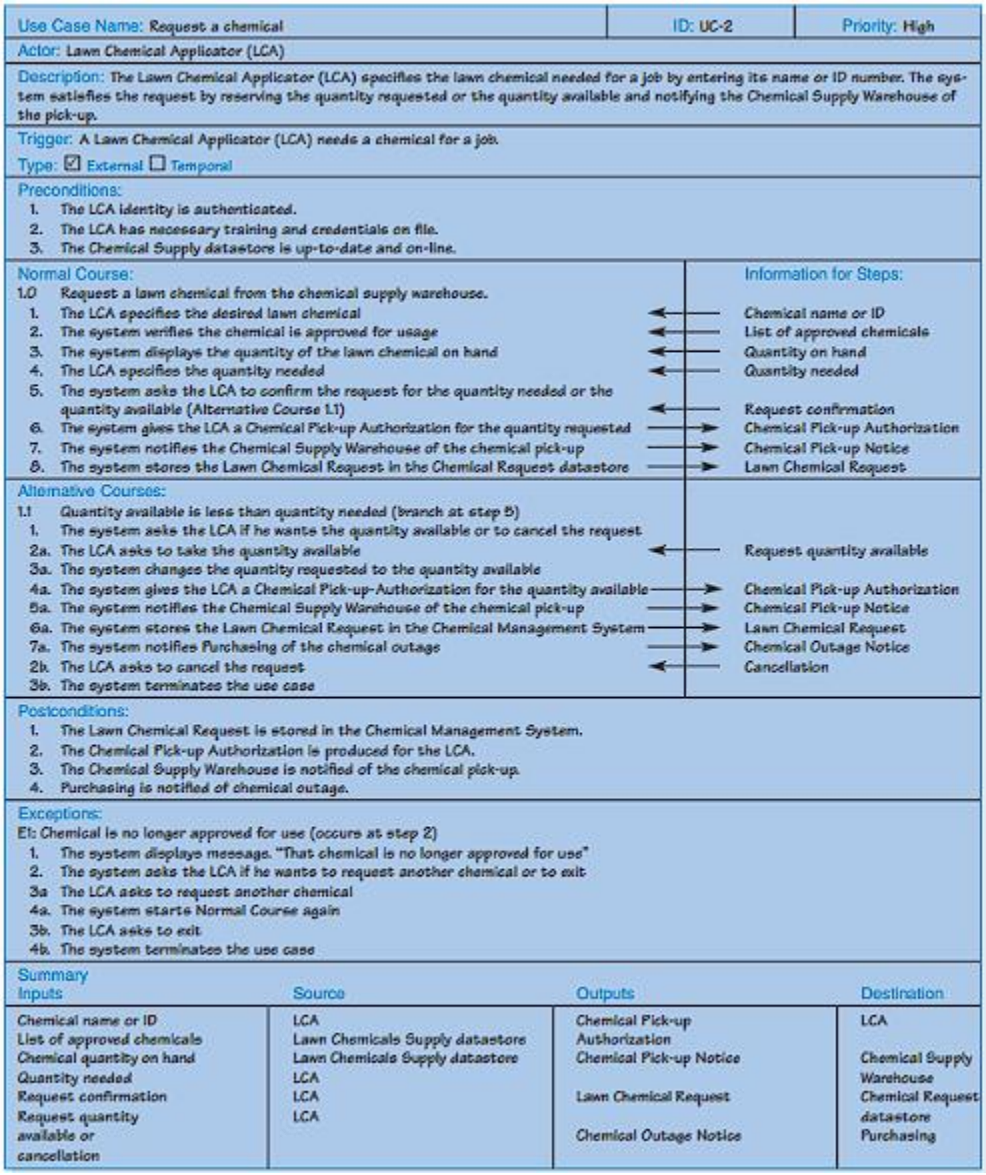
1. The system displays message. "That chemical is no longer approved for use"
2. The system asks the LCA if he wants to request another chemical or to exit

3a The LCA asks to request another chemical
4a. The system starts Normal Course again
3b. The LCA asks to exit
4b. The system terminates the use case

Summary Inputs	Source	Outputs	Destination
Chemical name or ID	LCA	Chemical Pick-up Authorization	LCA
List of approved chemicals	Lawn Chemicals Supply datastore		
Chemical quantity on hand	Lawn Chemicals Supply datastore	Chemical Pick-up Notice	Chemical Supply Warehouse
Quantity needed	LCA		
Request confirmation	LCA	Lawn Chemical Request	Chemical Request datastore
Request quantity available or cancellation	LCA	Chemical Outage Notice	Purchasing

Alternative Use Case Formats

- Request a Chemical Use Case
  - a **fully dressed use case**
  - very thorough, detailed, and highly structured
  - written as an essential use case
  - it depicts the user-system interactions as abstract, technology-independent steps



Use Case Name: Request a chemical	ID: UC-2	Priority: High

Actor: Lawn Chemical Applicator (LCA)

Description: The Lawn Chemical Applicator (LCA) specifies the lawn chemical needed for a job by entering its name or ID number. The system satisfies the request by reserving the quantity requested or the quantity available and notifying the Chemical Supply Warehouse of the pick-up.

Trigger: A Lawn Chemical Applicator (LCA) needs a chemical for a job.

Type: ☑ External ☐ Temporal

Preconditions:
1. The LCA identity is authenticated.
2. The LCA has necessary training and credentials on file.
3. The Chemical Supply datastore is up-to-date and on-line.

Normal Course:		Information for Steps:
1.0 Request a lawn chemical from the chemical supply warehouse.		
1. The LCA specifies the desired lawn chemical	←	Chemical name or ID
2. The system verifies the chemical is approved for usage	←	List of approved chemicals
3. The system displays the quantity of the lawn chemical on hand	←	Quantity on hand
4. The LCA specifies the quantity needed	←	Quantity needed
5. The system asks the LCA to confirm the request for the quantity needed or the quantity available (Alternative Course 1.1)	←	Request confirmation
6. The system gives the LCA a Chemical Pick-up Authorization for the quantity requested	→	Chemical Pick-up Authorization
7. The system notifies the Chemical Supply Warehouse of the chemical pick-up	→	Chemical Pick-up Notice
8. The system stores the Lawn Chemical Request in the Chemical Request datastore	→	Lawn Chemical Request
Alternative Courses:		
1.1 Quantity available is less than quantity needed (branch at step 5)		
1. The system asks the LCA if he wants the quantity available or to cancel the request		
2a. The LCA asks to take the quantity available	←	Request quantity available
3a. The system changes the quantity requested to the quantity available		
4a. The system gives the LCA a Chemical Pick-up-Authorization for the quantity available	→	Chemical Pick-up Authorization
5a. The system notifies the Chemical Supply Warehouse of the chemical pick-up	→	Chemical Pick-up Notice
6a. The system stores the Lawn Chemical Request in the Chemical Management System	→	Lawn Chemical Request
7a. The system notifies Purchasing of the chemical outage	→	Chemical Outage Notice
2b. The LCA asks to cancel the request	←	Cancellation
3b. The system terminates the use case		

Postconditions:
1. The Lawn Chemical Request is stored in the Chemical Management System.
2. The Chemical Pick-up Authorization is produced for the LCA.
3. The Chemical Supply Warehouse is notified of the chemical pick-up.
4. Purchasing is notified of chemical outage.

Exceptions:

E1: Chemical is no longer approved for use (occurs at step 2)

1. The system displays message. "That chemical is no longer approved for use"
2. The system asks the LCA if he wants to request another chemical or to exit

3a The LCA asks to request another chemical

4a. The system starts Normal Course again

3b. The LCA asks to exit

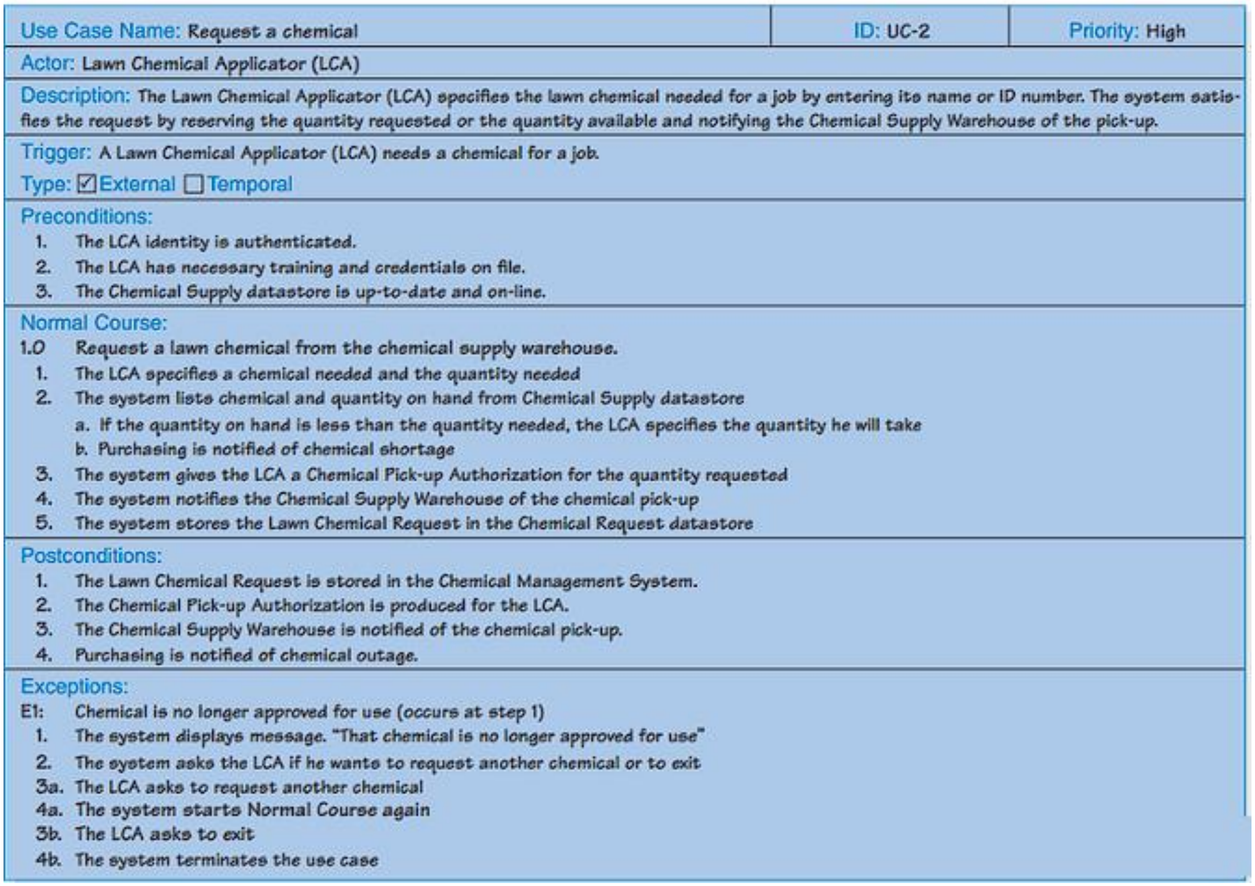
4b. The system terminates the use case

Summary

Inputs	Source	Outputs	Destination
Chemical name or ID	LCA	Chemical Pick-up Authorization	LCA
List of approved chemicals	Lawn Chemicals Supply datastore	Chemical Pick-up Notice	Chemical Supply Warehouse
Chemical quantity on hand	Lawn Chemicals Supply datastore	Lawn Chemical Request	Chemical Request datastore
Quantity needed	LCA	Chemical Outage Notice	Purchasing
Request confirmation	LCA		
Request quantity available or cancellation	LCA		

- Fully Dressed Use Case
    - Is not always required but does provide value in certain circumstances. Fully dressed use cases are especially valuable when:
        - User representatives are not closely engaged with the development team throughout the project.
        - The application is complex and has a high risk associated with system failures.
        - Comprehensive test cases will be based on the user requirements.
        - Collaborating remote teams need a detailed, shared understanding of the user requirements.

Casual Use Case Format



Use Case Name: Request a chemical	ID: UC-2	Priority: High

Actor: Lawn Chemical Applicator (LCA)

Description: The Lawn Chemical Applicator (LCA) specifies the lawn chemical needed for a job by entering its name or ID number. The system satisfies the request by reserving the quantity requested or the quantity available and notifying the Chemical Supply Warehouse of the pick-up.

Trigger: A Lawn Chemical Applicator (LCA) needs a chemical for a job.

Type: ☑ External ☐ Temporal

Preconditions:
1. The LCA identity is authenticated.
2. The LCA has necessary training and credentials on file.
3. The Chemical Supply datastore is up-to-date and on-line.

Normal Course:
1.0 Request a lawn chemical from the chemical supply warehouse.
1. The LCA specifies a chemical needed and the quantity needed
2. The system lists chemical and quantity on hand from Chemical Supply datastore
    a. If the quantity on hand is less than the quantity needed, the LCA specifies the quantity he will take
    b. Purchasing is notified of chemical shortage
3. The system gives the LCA a Chemical Pick-up Authorization for the quantity requested
4. The system notifies the Chemical Supply Warehouse of the chemical pick-up
5. The system stores the Lawn Chemical Request in the Chemical Request datastore

Postconditions:
1. The Lawn Chemical Request is stored in the Chemical Management System.
2. The Chemical Pick-up Authorization is produced for the LCA.
3. The Chemical Supply Warehouse is notified of the chemical pick-up.
4. Purchasing is notified of chemical outage.

Exceptions:
E1: Chemical is no longer approved for use (occurs at step 1)
1. The system displays message. "That chemical is no longer approved for use"
2. The system asks the LCA if he wants to request another chemical or to exit
3a. The LCA asks to request another chemical
4a. The system starts Normal Course again
3b. The LCA asks to exit
4b. The system terminates the use case

Use Cases and the Functional Requirements

- Functional Requirements
    - DOs and DONTs of the system
    - It is the things that are placed concisely that aims to know what the system should provide to accomplish the user's tasks.

Where can we obtain the data for the functional requirements?

ANSWER: **based on the stories or observations of the potential users of the system**

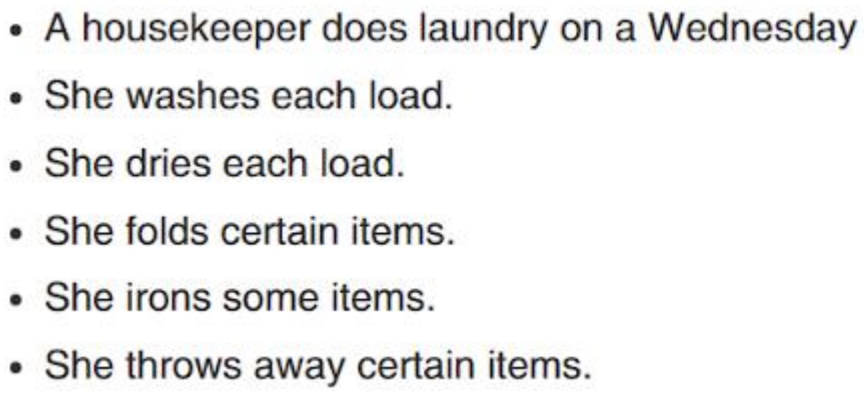
How can we get stories or observations from the potential users of the system?

ANSWER: **Use Cases**

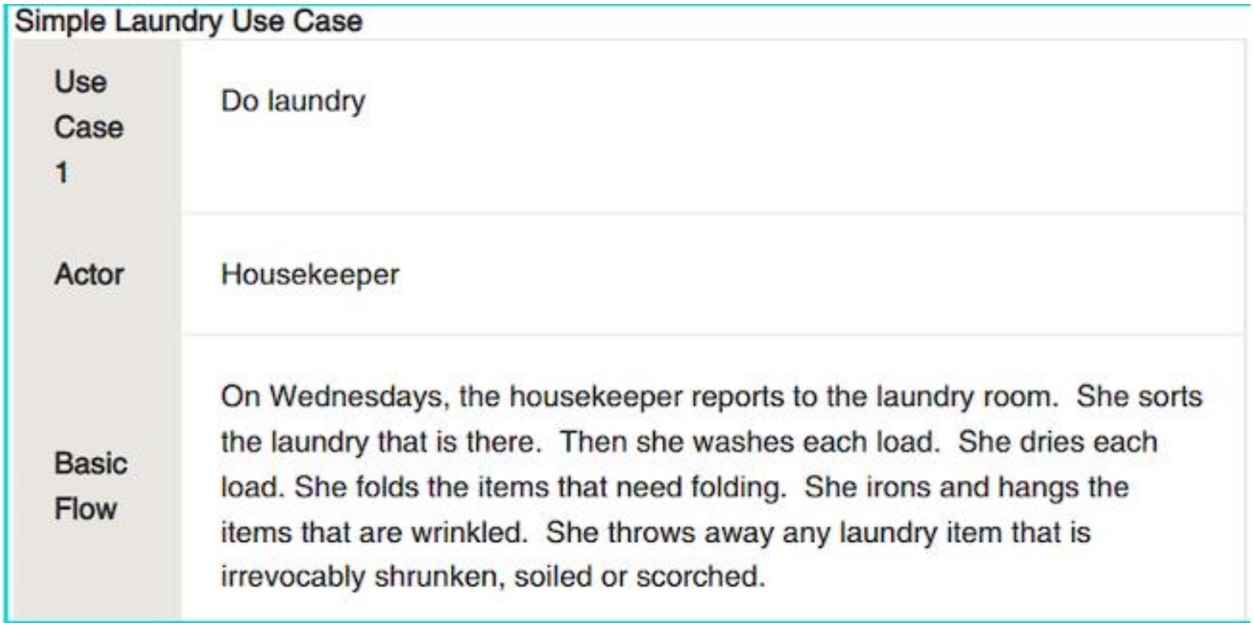
Why are functional requirements essential?

- Stakeholders **have a single source of truth**
- High predictability = saves time and resources
- Helps developers see if the system caters all the specific functionalities

Examples:

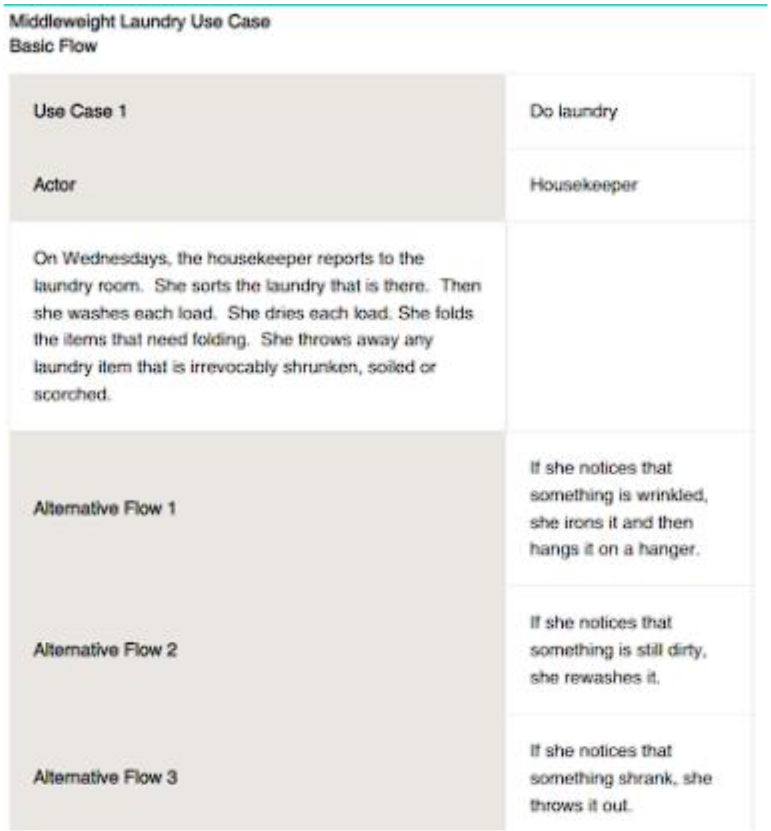


- A housekeeper does laundry on a Wednesday
- She washes each load.
- She dries each load.
- She folds certain items.
- She irons some items.
- She throws away certain items.



Simple Laundry Use Case

Use Case 1	Do laundry
Actor	Housekeeper
Basic Flow	On Wednesdays, the housekeeper reports to the laundry room. She sorts the laundry that is there. Then she washes each load. She dries each load. She folds the items that need folding. She irons and hangs the items that are wrinkled. She throws away any laundry item that is irrevocably shrunken, soiled or scorched.



Middleweight Laundry Use Case
Basic Flow

Use Case 1	Do laundry
Actor	Housekeeper
On Wednesdays, the housekeeper reports to the laundry room. She sorts the laundry that is there. Then she washes each load. She dries each load. She folds the items that need folding. She throws away any laundry item that is irrevocably shrunken, soiled or scorched.	
Alternative Flow 1	If she notices that something is wrinkled, she irons it and then hangs it on a hanger.
Alternative Flow 2	If she notices that something is still dirty, she rewashes it.
Alternative Flow 3	If she notices that something shrank, she throws it out.

Use Cases and Testing

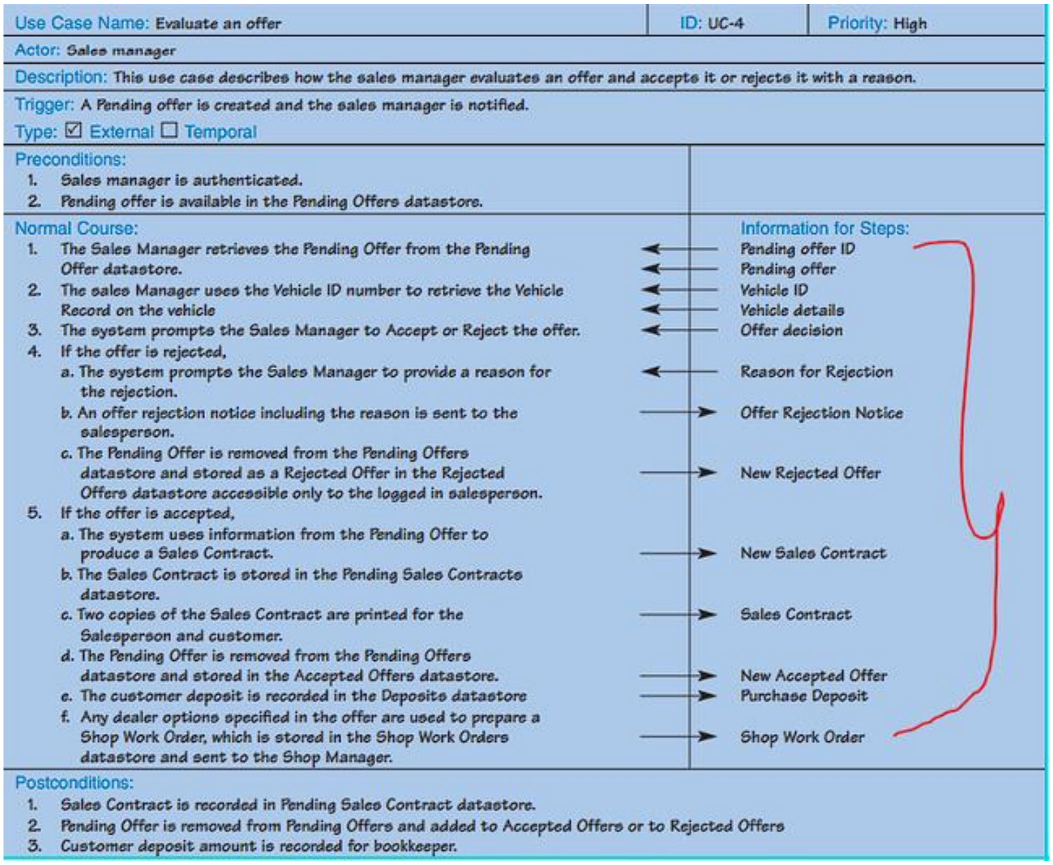
A technique for identifying test cases that cover the entire system from start to finish on a transaction-by-transaction basis. Many organizations develop test plans early in the development process. This strategy gives the testing/quality assurance personnel an early understanding of the system underdevelopment. It also allows them to readily identify elements of the tests they will want to perform when the system is put through its paces.

Building Use Cases

- Identify the Major Use Cases
    - begins with the requirement definition:
        - **Process-oriented functional requirements** - things the system must do
        - **Information-oriented functional requirements -** content the system must have
    - Activities
        - Start a use case report form for each use case by filling in the name, description, and trigger.
        - If there are more than nine use cases, group them into packages.
    - Typical Questions Asked
        - Ask who, what, when, and where about the use cases (or tasks)
        - What are the major tasks that are performed?
        - What triggers this task? What tells you to perform this task?
- Identify the Major Steps for Each Use Case
    - Here are the following steps:
        - **Complete the main body of the use case form -** users and analysts work together to **describe the envisioned interactions between the user and the system** to complete the response to the event.
        - **The user-system interactions should be outlined as a series of steps in the Normal Course section of the form -** focus on what an independent observer would see the user and system do in response to the event

- Identify Elements withing Steps
  - The goal at this point is to identify the major inputs and outputs for each step.
  - Identify detailed information for the steps but provide only general categories in the summary area of the use case form.

Examples of elements within steps:



Use Case Name: Evaluate an offer	ID: UC-4	Priority: High

Actor: Sales manager

Description: This use case describes how the sales manager evaluates an offer and accepts it or rejects it with a reason.

Trigger: A Pending offer is created and the sales manager is notified.

Type: ☑ External □ Temporal

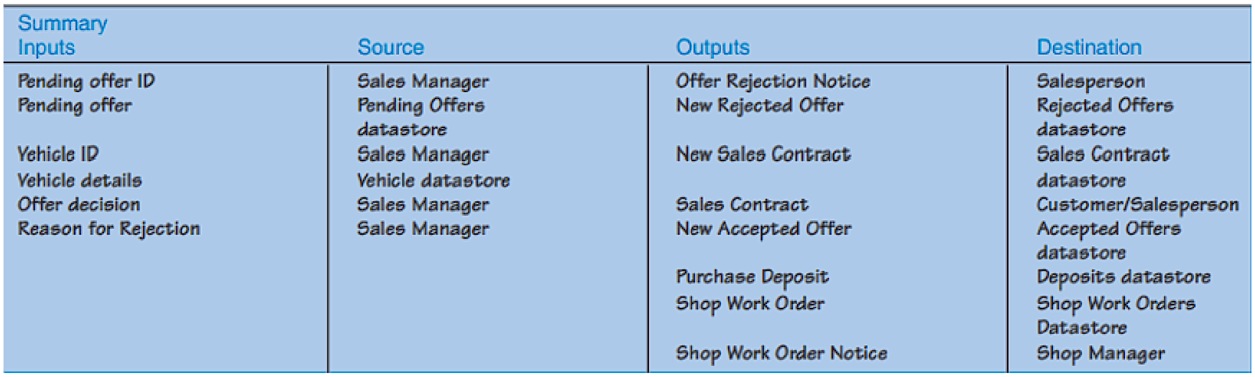
Preconditions:

1. Sales manager is authenticated.
2. Pending offer is available in the Pending Offers datastore.

Normal Course:		Information for Steps:
1. The Sales Manager retrieves the Pending Offer from the Pending Offer datastore.	←	Pending offer ID
	←	Pending offer
2. The sales Manager uses the Vehicle ID number to retrieve the Vehicle Record on the vehicle	←	Vehicle ID
	←	Vehicle details
3. The system prompts the Sales Manager to Accept or Reject the offer.	←	Offer decision
4. If the offer is rejected,		
a. The system prompts the Sales Manager to provide a reason for the rejection.	←	Reason for Rejection
b. An offer rejection notice including the reason is sent to the salesperson.	→	Offer Rejection Notice
c. The Pending Offer is removed from the Pending Offers datastore and stored as a Rejected Offer in the Rejected Offers datastore accessible only to the logged in salesperson.	→	New Rejected Offer
5. If the offer is accepted,		
a. The system uses information from the Pending Offer to produce a Sales Contract.	→	New Sales Contract
b. The Sales Contract is stored in the Pending Sales Contracts datastore.		
c. Two copies of the Sales Contract are printed for the Salesperson and customer.	→	Sales Contract
d. The Pending Offer is removed from the Pending Offers datastore and stored in the Accepted Offers datastore.	→	New Accepted Offer
e. The customer deposit is recorded in the Deposits datastore	→	Purchase Deposit
f. Any dealer options specified in the offer are used to prepare a Shop Work Order, which is stored in the Shop Work Orders datastore and sent to the Shop Manager.	→	Shop Work Order

Postconditions:

1. Sales Contract is recorded in Pending Sales Contract datastore.
2. Pending Offer is removed from Pending Offers and added to Accepted Offers or to Rejected Offers
3. Customer deposit amount is recorded for bookkeeper.

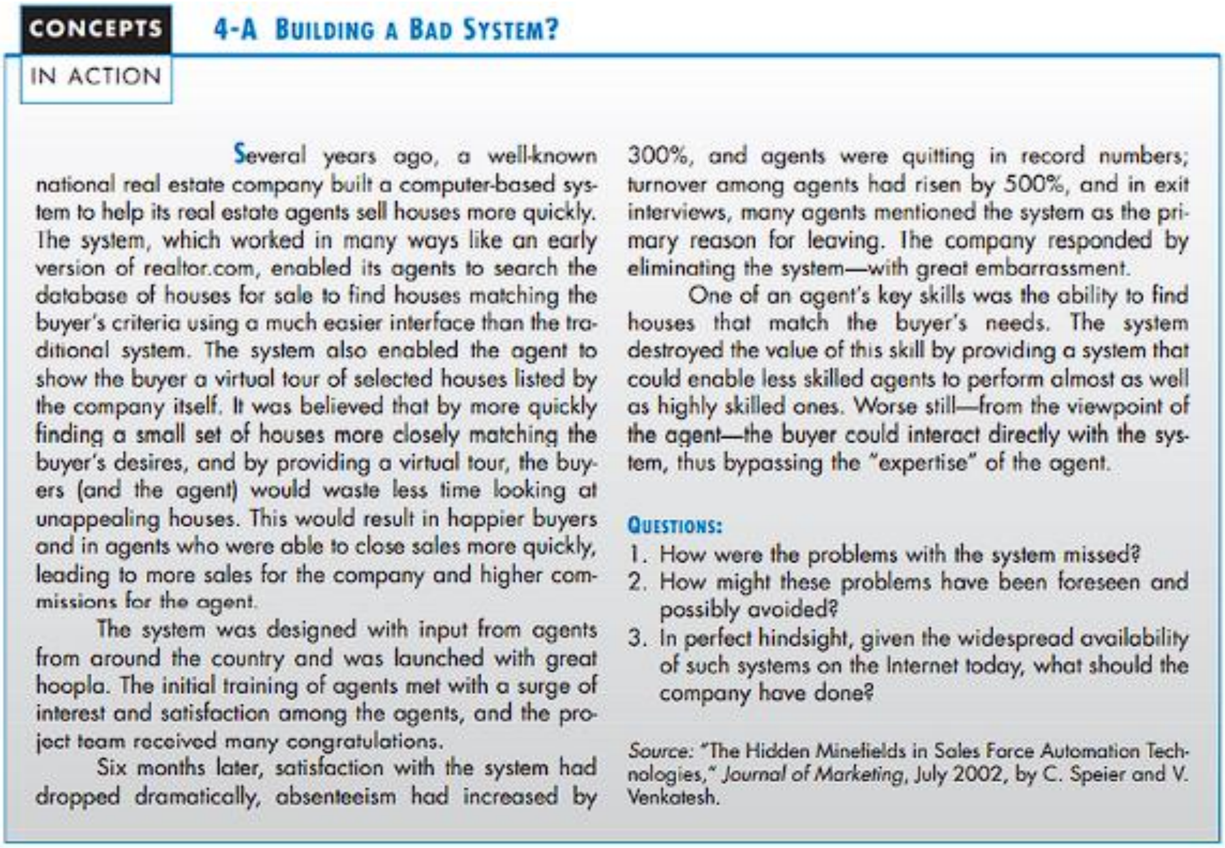


Summary Inputs	Source	Outputs	Destination
Pending offer ID	Sales Manager	Offer Rejection Notice	Salesperson
Pending offer	Pending Offers datastore	New Rejected Offer	Rejected Offers datastore
Vehicle ID	Sales Manager	New Sales Contract	Sales Contract datastore
Vehicle details	Vehicle datastore		
Offer decision	Sales Manager	Sales Contract	Customer/Salesperson
Reason for Rejection	Sales Manager	New Accepted Offer	Accepted Offers datastore
		Purchase Deposit	Deposits datastore
		Shop Work Order	Shop Work Orders Datastore
		Shop Work Order Notice	Shop Manager

Confirm the Use Case

- Confirm that the use case is correct as written
- Review the use case with the users to make sure that each step and each **input and output are correct** and that the final result of the use case is **consistent**.
- Ask the user to **role-play, or execute the use case** by using the written steps in the use case.

The user follows the written steps like a recipe to make sure that those steps and inputs really can produce the outputs and final result defined for the use case.



CONCEPTS IN ACTION

4-A Building a Bad System?

Several years ago, a well-known national real estate company built a computer-based system to help its real estate agents sell houses more quickly. The system, which worked in many ways like an early version of realtor.com, enabled its agents to search the database of houses for sale to find houses matching the buyer's criteria using a much easier interface than the traditional system. The system also enabled the agent to show the buyer a virtual tour of selected houses listed by the company itself. It was believed that by more quickly finding a small set of houses more closely matching the buyer's desires, and by providing a virtual tour, the buyers (and the agent) would waste less time looking at unappealing houses. This would result in happier buyers and in agents who were able to close sales more quickly, leading to more sales for the company and higher commissions for the agent.

The system was designed with input from agents from around the country and was launched with great hoopla. The initial training of agents met with a surge of interest and satisfaction among the agents, and the project team received many congratulations.

Six months later, satisfaction with the system had dropped dramatically, absenteeism had increased by 300%, and agents were quitting in record numbers; turnover among agents had risen by 500%, and in exit interviews, many agents mentioned the system as the primary reason for leaving. The company responded by eliminating the system—with great embarrassment.

One of an agent's key skills was the ability to find houses that match the buyer's needs. The system destroyed the value of this skill by providing a system that could enable less skilled agents to perform almost as well as highly skilled ones. Worse still—from the viewpoint of the agent—the buyer could interact directly with the system, thus bypassing the "expertise" of the agent.

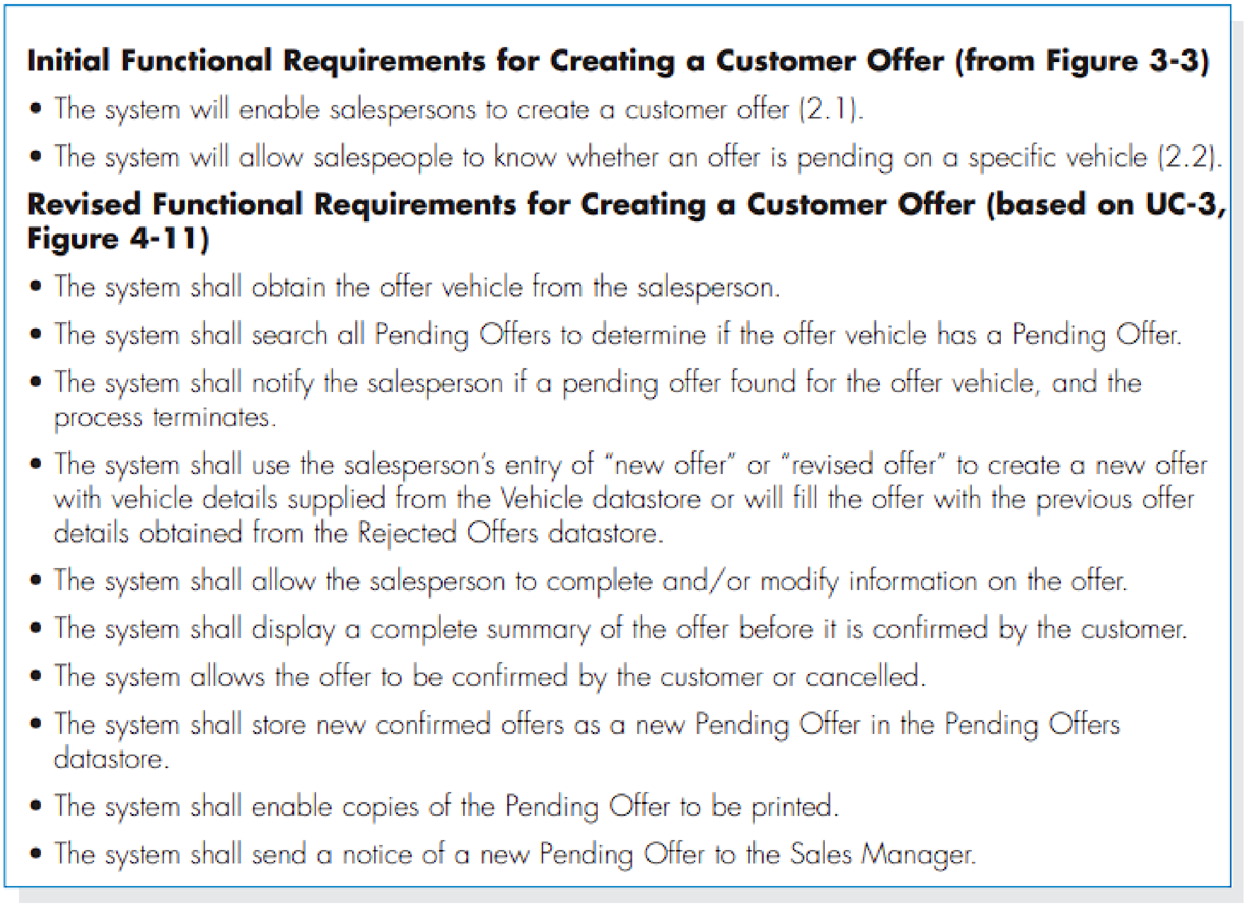
Questions:

1. How were the problems with the system missed?
2. How might these problems have been foreseen and possibly avoided?
3. In perfect hindsight, given the widespread availability of such systems on the Internet today, what should the company have done?

Source: "The Hidden Minefields in Sales Force Automation Technologies," *Journal of Marketing*, July 2002, by C. Speier and V. Venkatesh.

Revise Functional Requirements Based on Use Cases

The functional requirements in the requirements definition may be modified to reflect this more detailed understanding and to provide insight to the development team on some “back-end” processing that will be needed that may not be obvious from the use cases alone.



Initial Functional Requirements for Creating a Customer Offer (from Figure 3-3)

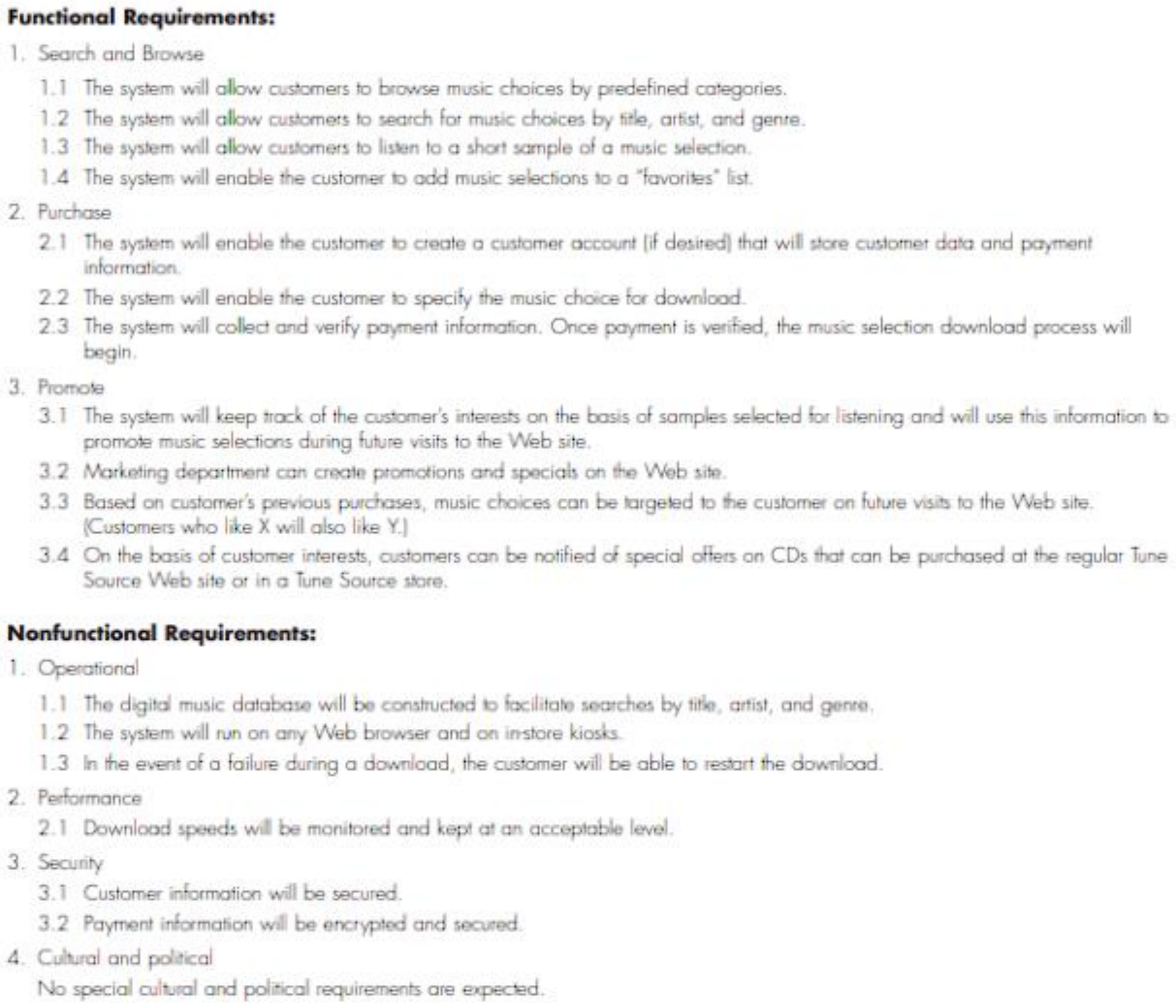
- The system will enable salespersons to create a customer offer (2.1).
- The system will allow salespeople to know whether an offer is pending on a specific vehicle (2.2).

Revised Functional Requirements for Creating a Customer Offer (based on UC-3, Figure 4-11)

- The system shall obtain the offer vehicle from the salesperson.
- The system shall search all Pending Offers to determine if the offer vehicle has a Pending Offer.
- The system shall notify the salesperson if a pending offer found for the offer vehicle, and the process terminates.
- The system shall use the salesperson’s entry of “new offer” or “revised offer” to create a new offer with vehicle details supplied from the Vehicle datastore or will fill the offer with the previous offer details obtained from the Rejected Offers datastore.
- The system shall allow the salesperson to complete and/or modify information on the offer.
- The system shall display a complete summary of the offer before it is confirmed by the customer.
- The system allows the offer to be confirmed by the customer or cancelled.
- The system shall store new confirmed offers as a new Pending Offer in the Pending Offers datastore.
- The system shall enable copies of the Pending Offer to be printed.
- The system shall send a notice of a new Pending Offer to the Sales Manager.

Applying the Concepts at Tune Source

The first step in creating the use cases is to identify the major use cases according to the requirements definition, which was developed in the last chapter. Take a minute and carefully read the requirements definition. Identify the major use cases that you think need additional definition before you continue reading. The information in the functional requirements definition sometimes just flows into the use cases, but it usually requires some thought as to how to structure the use cases. After you read the requirements definition, you may be tempted to identify use cases that correspond directly to the requirement categories, such as (1) search and browse, (2) purchase, and (3) promote.



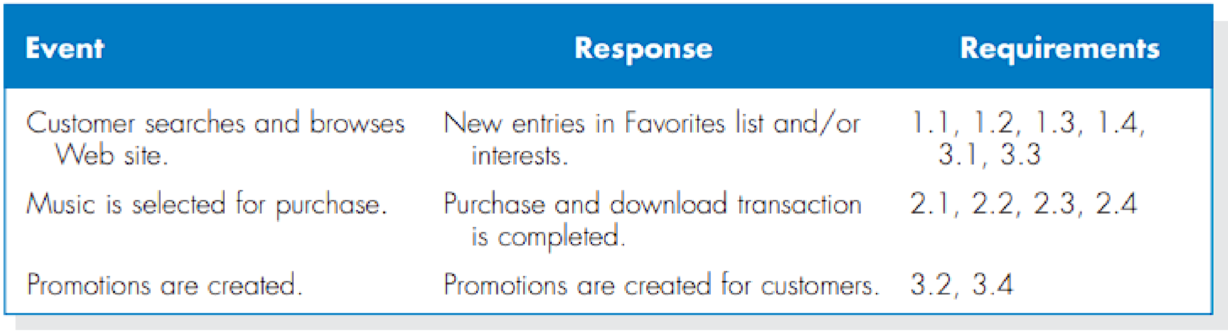
Functional Requirements:

1. Search and Browse
   - 1.1 The system will allow customers to browse music choices by predefined categories.
   - 1.2 The system will allow customers to search for music choices by title, artist, and genre.
   - 1.3 The system will allow customers to listen to a short sample of a music selection.
   - 1.4 The system will enable the customer to add music selections to a "favorites" list.
2. Purchase
   - 2.1 The system will enable the customer to create a customer account (if desired) that will store customer data and payment information.
   - 2.2 The system will enable the customer to specify the music choice for download.
   - 2.3 The system will collect and verify payment information. Once payment is verified, the music selection download process will begin.
3. Promote
   - 3.1 The system will keep track of the customer's interests on the basis of samples selected for listening and will use this information to promote music selections during future visits to the Web site.
   - 3.2 Marketing department can create promotions and specials on the Web site.
   - 3.3 Based on customer's previous purchases, music choices can be targeted to the customer on future visits to the Web site. (Customers who like X will also like Y.)
   - 3.4 On the basis of customer interests, customers can be notified of special offers on CDs that can be purchased at the regular Tune Source Web site or in a Tune Source store.

Nonfunctional Requirements:

1. Operational
   - 1.1 The digital music database will be constructed to facilitate searches by title, artist, and genre.
   - 1.2 The system will run on any Web browser and on in-store kiosks.
   - 1.3 In the event of a failure during a download, the customer will be able to restart the download.
2. Performance
   - 2.1 Download speeds will be monitored and kept at an acceptable level.
3. Security
   - 3.1 Customer information will be secured.
   - 3.2 Payment information will be encrypted and secured.
4. Cultural and political

   No special cultural and political requirements are expected.

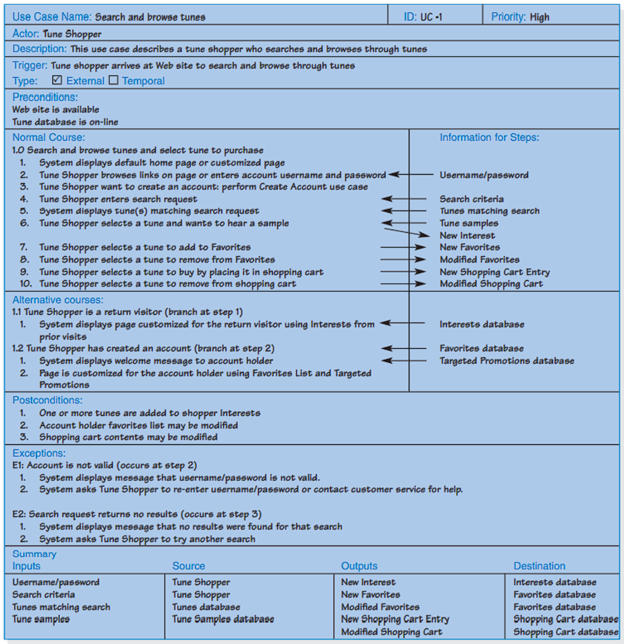


Event	Response	Requirements
Customer searches and browses Web site.	New entries in Favorites list and/or interests.	1.1, 1.2, 1.3, 1.4, 3.1, 3.3
Music is selected for purchase.	Purchase and download transaction is completed.	2.1, 2.2, 2.3, 2.4
Promotions are created.	Promotions are created for customers.	3.2, 3.4

Elaborating on the Use Cases

The goal at this point is to describe how the use case operates. The best way to begin to understand these use cases is to visualize yourself browsing a sales-oriented Web site, searching for particular items, investigating specific items further, finally making a decision to buy, and completing the purchase. The techniques of visualizing your interaction with the process and thinking about how other systems work (informal benchmarking) are important techniques that help analysts and users understand how processes work and how to write the use cases.

- Search and Browse Tunes
    - The Tune shopper is intended to look around the tunes website but not necessarily includes for the purpose of purchasing any of the tunes.



Use Case Name: Search and browse tunes	ID: UC -1	Priority: High

Actor: Tune Shopper

Description: This use case describes a tune shopper who searches and browses through tunes

Trigger: Tune shopper arrives at Web site to search and browse through tunes

Type: ☑ External ☐ Temporal

Preconditions:
Web site is available
Tune database is on-line

Normal Course:	Information for Steps:
1.0 Search and browse tunes and select tune to purchase	
1. System displays default home page or customized page	
2. Tune Shopper browses links on page or enters account username and password	Username/password
3. Tune Shopper want to create an account: perform Create Account use case	
4. Tune Shopper enters search request	Search criteria
5. System displays tune(s) matching search request	Tunes matching search
6. Tune Shopper selects a tune and wants to hear a sample	Tune samples New Interest
7. Tune Shopper selects a tune to add to Favorites	New Favorites
8. Tune Shopper selects a tune to remove from Favorites	Modified Favorites
9. Tune Shopper selects a tune to buy by placing it in shopping cart	New Shopping Cart Entry
10. Tune Shopper selects a tune to remove from shopping cart	Modified Shopping Cart
Alternative courses:	
1.1 Tune Shopper is a return visitor (branch at step 1)	
1. System displays page customized for the return visitor using Interests from prior visits	Interests database
1.2 Tune Shopper has created an account (branch at step 2)	Favorites database
1. System displays welcome message to account holder	Targeted Promotions database
2. Page is customized for the account holder using Favorites List and Targeted Promotions	

Postconditions:
1. One or more tunes are added to shopper Interests
2. Account holder favorites list may be modified
3. Shopping cart contents may be modified

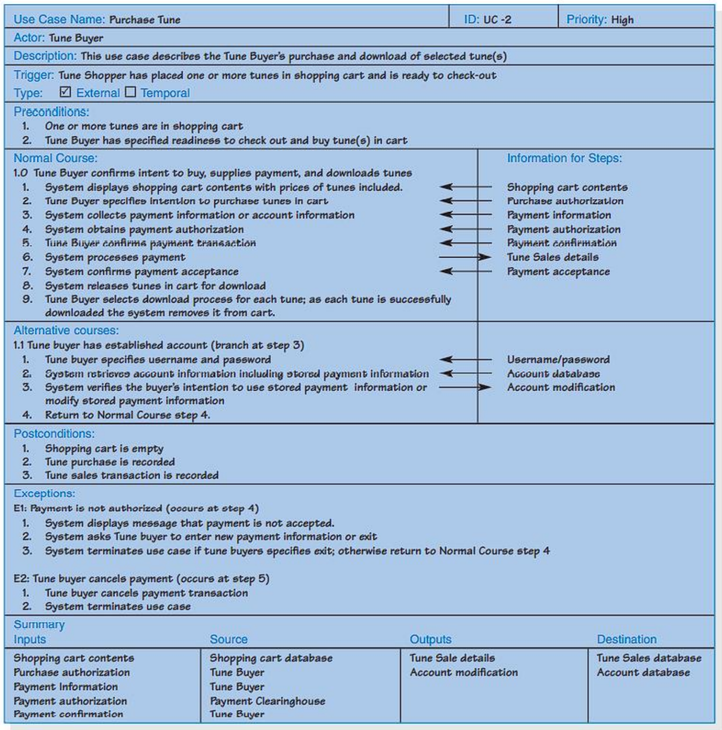
Exceptions:
E1: Account is not valid (occurs at step 2)
1. System displays message that username/password is not valid.
2. System asks Tune Shopper to re-enter username/password or contact customer service for help.

E2: Search request returns no results (occurs at step 3)
1. System displays message that no results were found for that search
2. System asks Tune Shopper to try another search

Summary

Inputs	Source	Outputs	Destination
Username/password	Tune Shopper	New Interest	Interests database
Search criteria	Tune Shopper	New Favorites	Favorites database
Tunes matching search	Tunes database	Modified Favorites	Favorites database
Tune samples	Tune Samples database	New Shopping Cart Entry	Shopping Cart database
		Modified Shopping Cart	Shopping Cart database

- Purchase Tunes
    - The Tune Buyer actor is specified for the Purchase Tunes use case. This designation is given because the website user has indicated a desire to purchase the item(s) in the cart.



Use Case Name: Purchase Tune	ID: UC -2	Priority: High

Actor: Tune Buyer

Description: This use case describes the Tune Buyer's purchase and download of selected tune(s)

Trigger: Tune Shopper has placed one or more tunes in shopping cart and is ready to check-out

Type: ☑ External □ Temporal

Preconditions:

1. One or more tunes are in shopping cart
2. Tune Buyer has specified readiness to check out and buy tune(s) in cart

Normal Course:		Information for Steps:
1.0 Tune Buyer confirms intent to buy, supplies payment, and downloads tunes		
1. System displays shopping cart contents with prices of tunes included.	←	Shopping cart contents
2. Tune Buyer specifies intention to purchase tunes in cart	←	Purchase authorization
3. System collects payment information or account information	←	Payment information
4. System obtains payment authorization	←	Payment authorization
5. Tune Buyer confirms payment transaction	←	Payment confirmation
6. System processes payment	→	Tune Sales details
7. System confirms payment acceptance	←	Payment acceptance
8. System releases tunes in cart for download		
9. Tune Buyer selects download process for each tune; as each tune is successfully downloaded the system removes it from cart.		

Alternative courses:		
1.1 Tune buyer has established account (branch at step 3)		
1. Tune buyer specifies username and password	←	Username/password
2. System retrieves account information including stored payment information	←	Account database
3. System verifies the buyer's intention to use stored payment information or modify stored payment information	→	Account modification
4. Return to Normal Course step 4.		

Postconditions:

1. Shopping cart is empty
2. Tune purchase is recorded
3. Tune sales transaction is recorded

Exceptions:

E1: Payment is not authorized (occurs at step 4)

1. System displays message that payment is not accepted.
2. System asks Tune buyer to enter new payment information or exit
3. System terminates use case if tune buyers specifies exit; otherwise return to Normal Course step 4

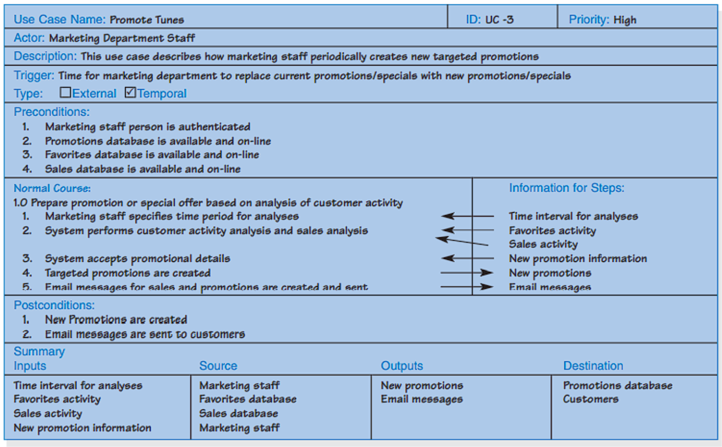
E2: Tune buyer cancels payment (occurs at step 5)

1. Tune buyer cancels payment transaction
2. System terminates use case

Summary

Inputs	Source	Outputs	Destination
Shopping cart contents Purchase authorization Payment Information Payment authorization Payment confirmation	Shopping cart database Tune Buyer Tune Buyer Payment Clearinghouse Tune Buyer	Tune Sale details Account modification	Tune Sales database Account database

- Promote Tunes
    - As for the Promote Tunes use case, the marketing team analyzes the files of recent customer purchases and additions to the customer favorites list on a regular basis.



Use Case Name: Promote Tunes	ID: UC -3	Priority: High

Actor: Marketing Department Staff

Description: This use case describes how marketing staff periodically creates new targeted promotions

Trigger: Time for marketing department to replace current promotions/specials with new promotions/specials

Type: □External ☑Temporal

Preconditions:

1. Marketing staff person is authenticated
2. Promotions database is available and on-line
3. Favorites database is available and on-line
4. Sales database is available and on-line

Normal Course:		Information for Steps:
1.0 Prepare promotion or special offer based on analysis of customer activity		
1. Marketing staff specifies time period for analyses	←	Time interval for analyses
2. System performs customer activity analysis and sales analysis	←	Favorites activity
	←	Sales activity
3. System accepts promotional details	←	New promotion information
4. Targeted promotions are created	→	New promotions
5. Email messages for sales and promotions are created and sent	→	Email messages

Postconditions:

1. New Promotions are created
2. Email messages are sent to customers

Summary

Inputs	Source	Outputs	Destination
Time interval for analyses Favorites activity Sales activity New promotion information	Marketing staff Favorites database Sales database Marketing staff	New promotions Email messages	Promotions database Customers