



Task and User Analysis

Dr Mahmood Niazi

Objectives

- To understand the relationship between goals, tasks and actions.
- To understand different techniques for task analysis.
- To understand the needs of users of different experience levels
 - beginners, intermediates, and expert.

Goals, tasks, and actions

- A **goal** is an end result to be achieved.
- A **task** is a structured set of related activities that are undertaken in some sequence.
- An **action** is an individual operation or step that needs to be undertaken as part of the task.

Example

- Goal

- ☐ Software Requirements Specification

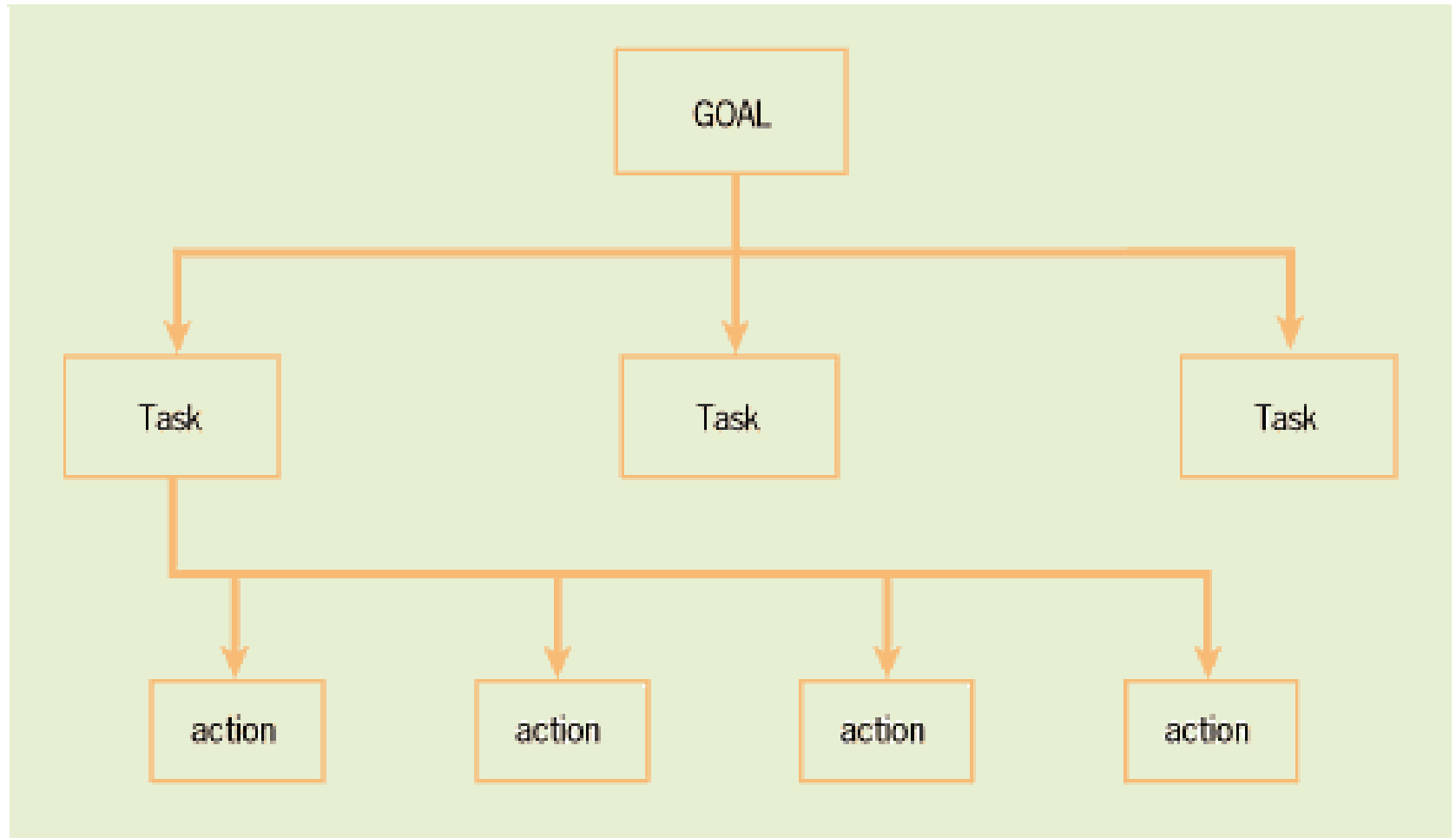
- Task

- ☐ Create use case diagrams, Identify functional and non-functional requirements

- Actions

- ☐ Write use case diagrams
- ☐ Write functional and non-functional requirements
- ☐ ...

Relationship between goals, tasks, and actions



Some task are decomposable into subtasks before the action level is reached.

Task characteristics for “withdrawing money” from an ATM

Does the task vary from one occasion to the next?

No.

How frequently is the task carried out?

May be daily, weekly, or less frequently.

What kinds of skills or knowledge are needed?

Must remember PIN to access machine.

Is the task affected by the environment?

Weather conditions could affect use of machine (e.g., the user may be wearing gloves in winter, it may be raining, bright sunlight may make reading the display difficult).

Is the task time critical?

Users may be in a hurry when using the ATM, since ATMs are often used for their speed and convenience.

Task characteristics for “withdrawing money” from an ATM *(Cont’d)*

Are there any safety or security hazards?

There are no safety hazards in the use of the ATM itself. However, the users’ personal safety in relation to onlookers and the safeguarding of their PINs and the cash withdrawn are considerations.

Will the work be done alone or with others?

The work will be done alone.

Will the users normally be switching between several tasks?

Many users will check their balance before withdrawing money. The users will not switch between tasks when withdrawing money, but external factors (like children) may divert their attention.

Task sequences

- Example: different task sequences for “sending a letter”

Sequence 1	Sequence 2
Write the letter.	Get an envelope.
Get an envelope.	Address the envelope.
Address the envelope.	Write the letter.
Put a stamp on the envelope.	Put the finished letter in the envelope.
Put the finished letter in the envelope.	Put a stamp on the envelope.

- UI should support different possible task sequences, and be flexible to accommodate them.

Task analysis

- It is the process of examining the way in which people perform their task.
 - It involves taking an in-depth look at the tasks and actions a person undertakes, along with the knowledge he needs to perform those tasks and reach a goal.
- Techniques for task analysis
 - Techniques that describe the steps required to complete a task
 - Use case diagrams

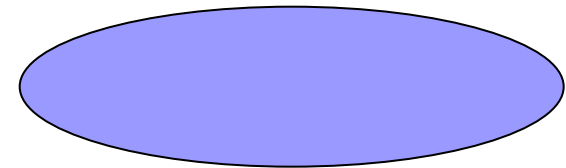
Use cases

- Use-cases are a scenario based technique in the UML which identify the actors in an interaction and which describe the interaction itself
- A set of use cases should describe all possible interactions with the system
- Sequence diagrams may be used to add detail to use-cases by showing the sequence of event processing in the system

Use Case modeling and documenting

A **use case** is a behaviorally related sequence of steps (a scenario), both automated and manual for the purpose of completing a single business task. A use case represents steps in a specific business process

- Submit change of address
- Maintain member order
- Make purchase inquiry



Make appointment

Use Case modeling and documenting

An **actor** represents anything that needs to interact with the system to exchange information. An actor is a user, a role, which could be an external system as well as a person.

- Club member
- Doctor
- Patient



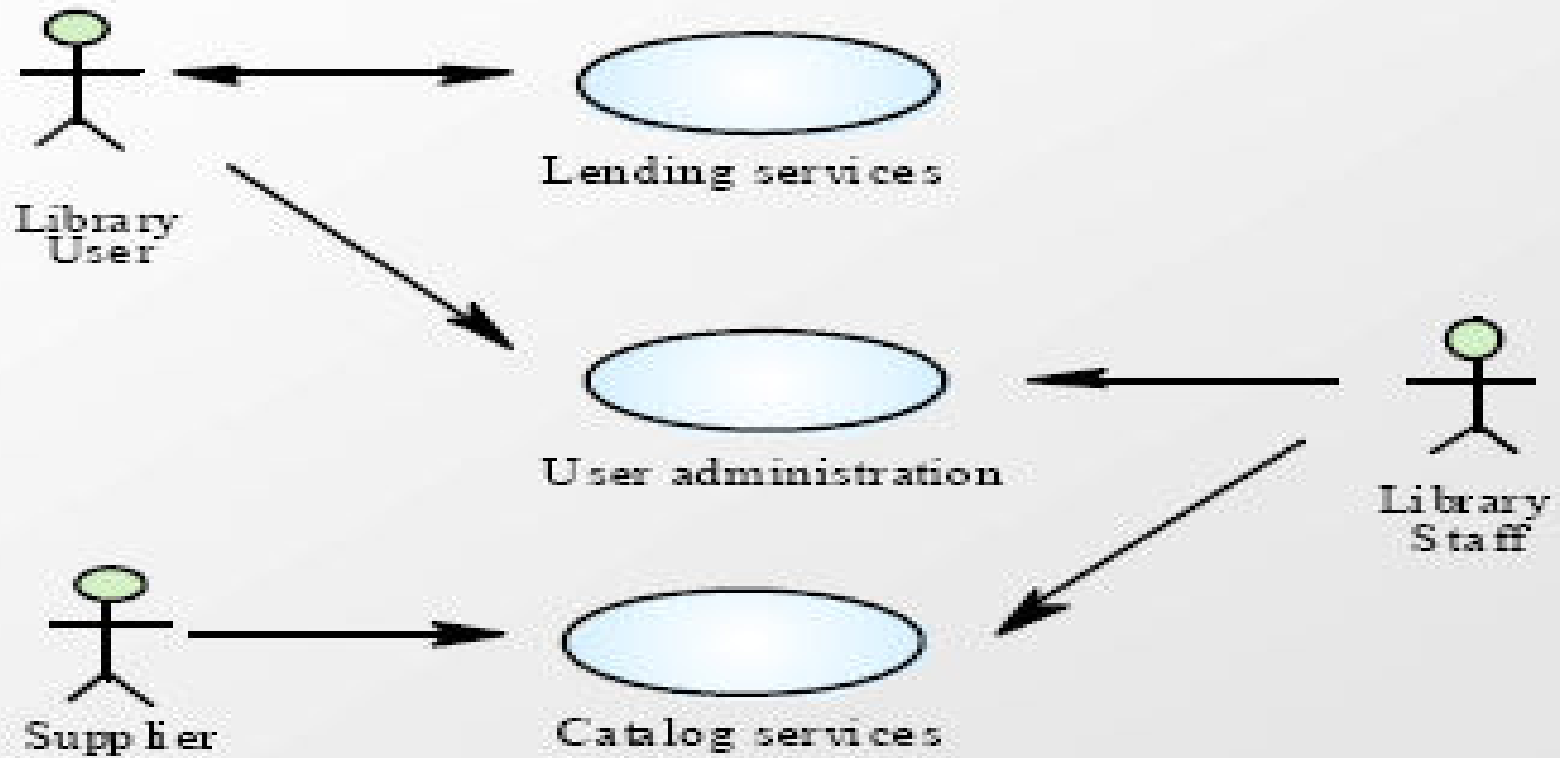
An actor

□ Communication



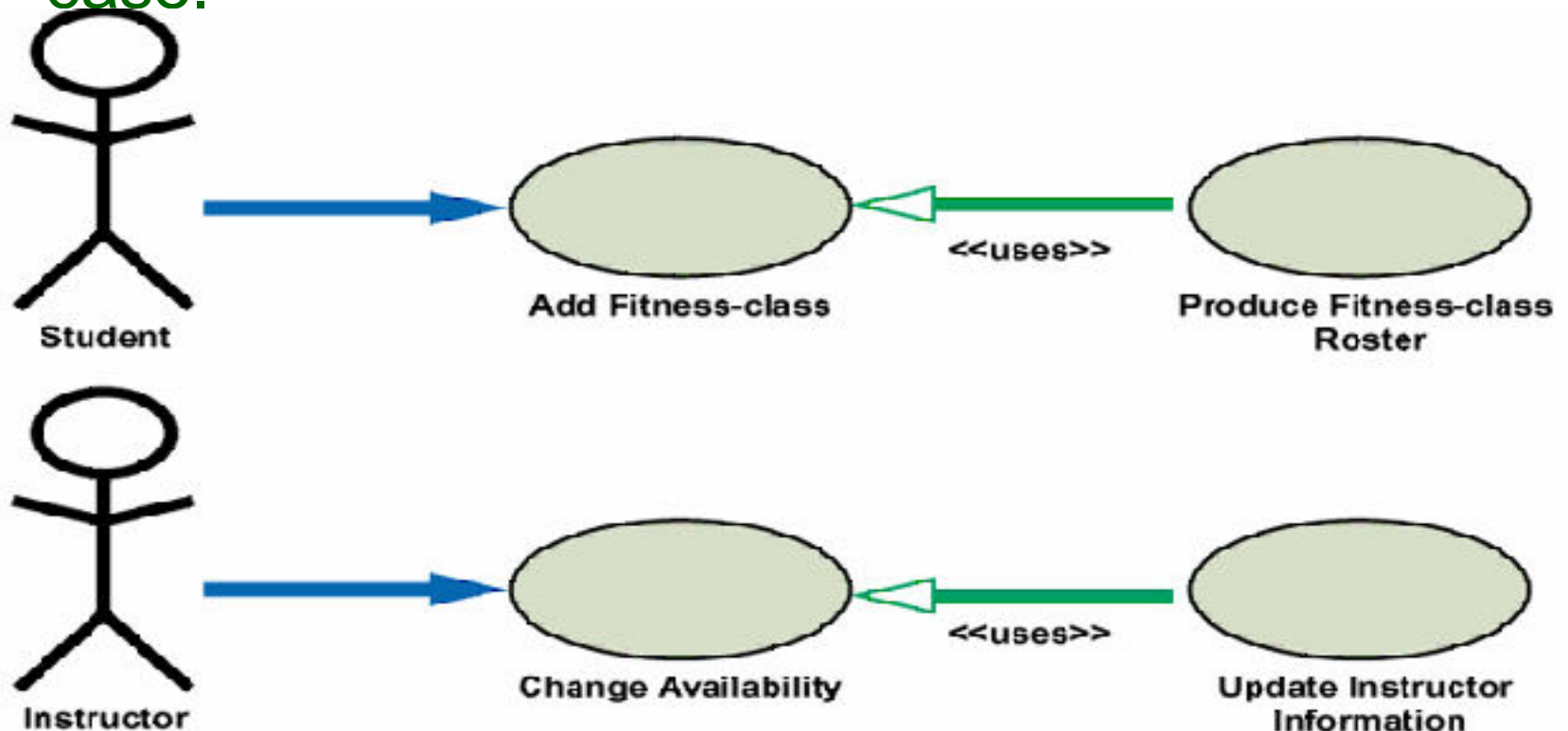
Use Case modeling and documenting

Library use-cases



Use Case modeling and documenting

Use case can also interact with other use case.



Use Case modeling and documenting

Step 1: Identifying Actors and Use Cases

Step 2: Constructing a Use Case Model Diagram

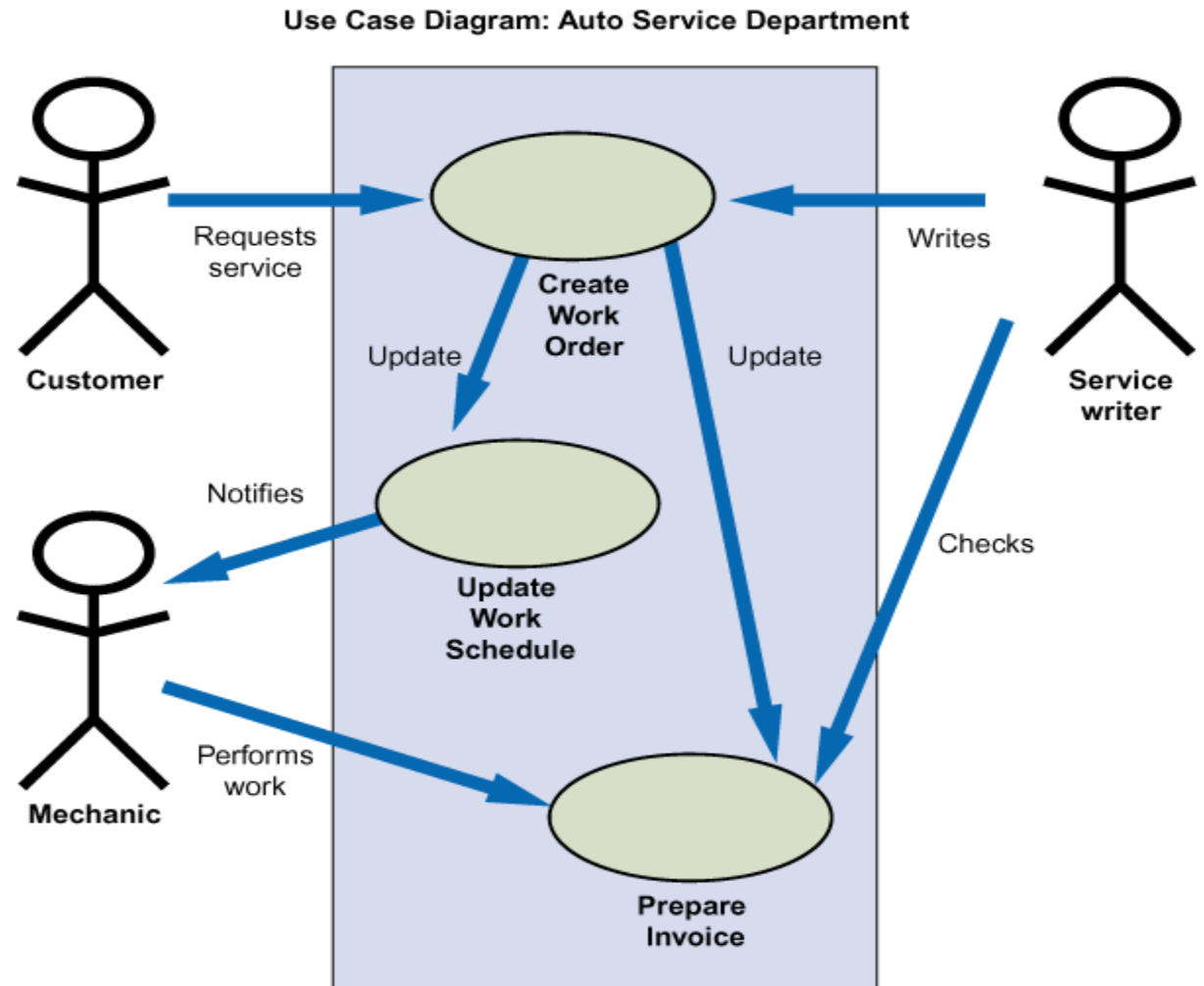
Step 3: Document the Use Case Course of Events

Step 1: Identifying Actors and Use Cases

- Task and user analysis
- Interviews
- Working with the users and identifying the primary inputs and outputs of the system and the external parties that receive and submit them.

Step 2: Constructing a Use case Model diagram

A use case model diagram is a visual summary of several related use cases within a system or subsystem.



Step 3: Documenting the Use Case



Add New Student Use Case

Add New Student

Name: Add New Student

Actor: Student/Manager

Description: Describes the process used to add a student to a fitness-class

Successful completion:

1. Manager checks FITNESS-CLASS SCHEDULE object for availability
2. Manager notifies student
3. Fitness-class is open and student pays fee
4. Manager registers student

Alternative:

1. Manager checks FITNESS-CLASS SCHEDULE object for availability
2. Fitness-class is full
3. Manager notifies student

Precondition: Student requests fitness-class

Postcondition: Student is enrolled in fitness-class and fees have been paid

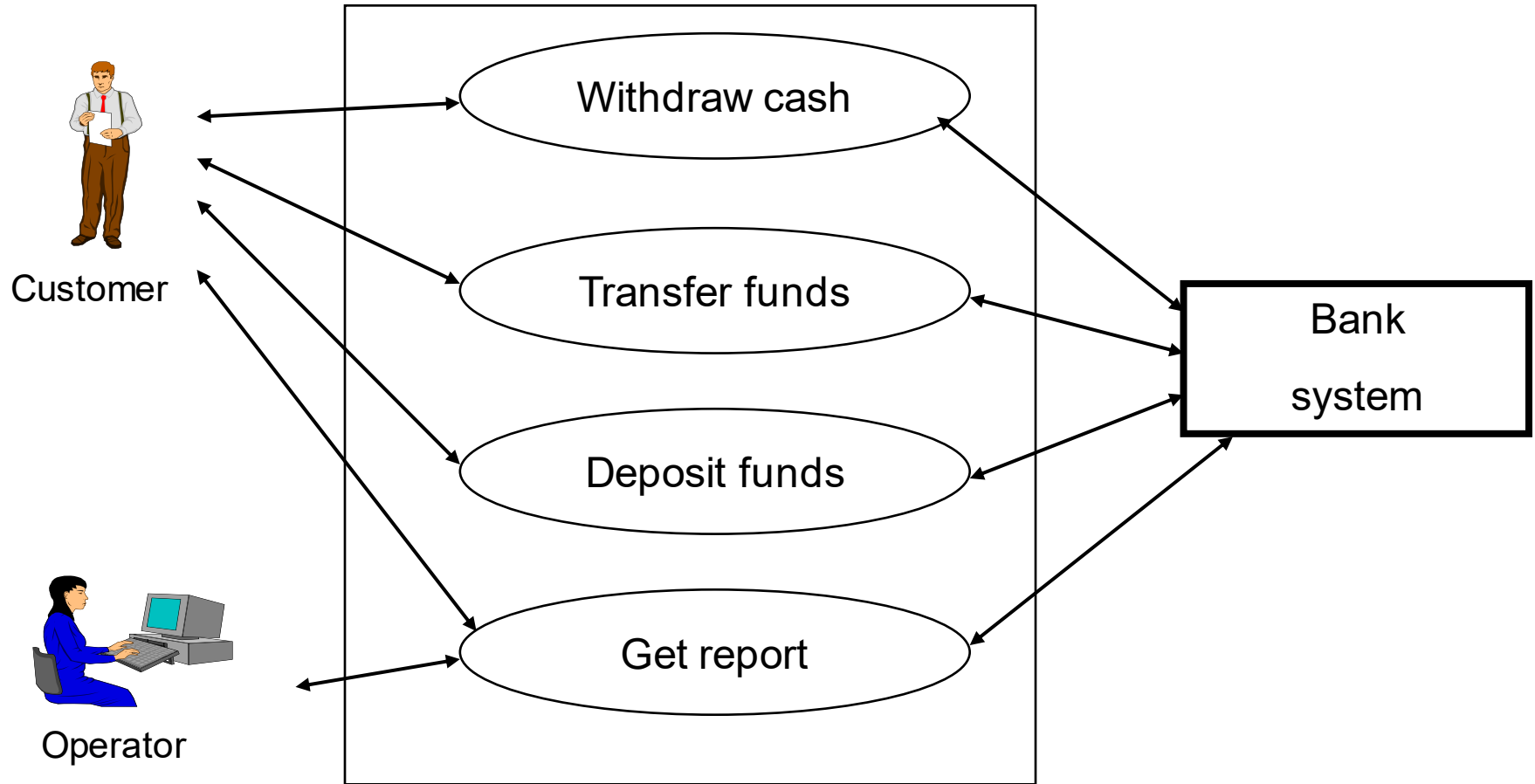
Assumptions: None

Benefits of Using Use Cases

- Facilitates user involvement.
- A view of the desired system's functionality from an external person's viewpoint.
- An effective tool for validating requirements.
- An effective communication tool.

Example Use Cases

Cash withdrawal from an ATM



Example Use Cases

Cash withdrawal from an ATM

NORMAL COURSE

A greeting message is waiting

The customer inserts card into machine

If card is acceptable ask for PIN

The customer enters PIN

If PIN is correct display customer accounts and request selection

The customer selects an account

The ATM displays the account limits

The customer enters the withdrawal amount

The bills are dispensed

The card is ejected

The receipt is printed out

Example Use Cases

Cash withdrawal from an ATM

ALTERNATE COURSE

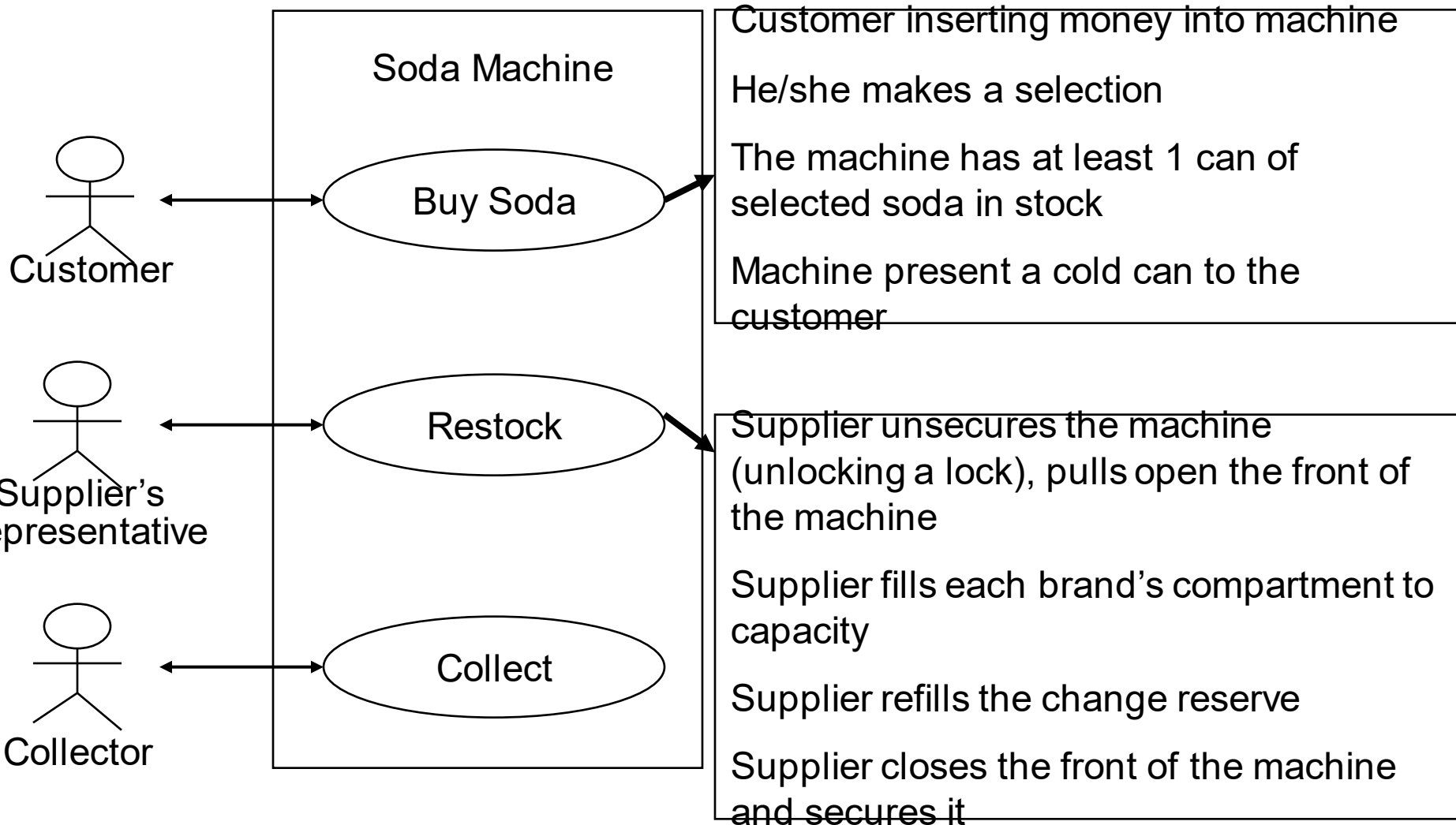
Card is not acceptable

Incorrect PIN code

Withdrawal request too large

Customer cancels

Example Use Cases



Example Use Cases: Class activity

Consider typical post office and the processes involved in selling stamps, renting post office boxes and delivering mail to postal customers.

Identify possible actors and use cases involved in post office functions

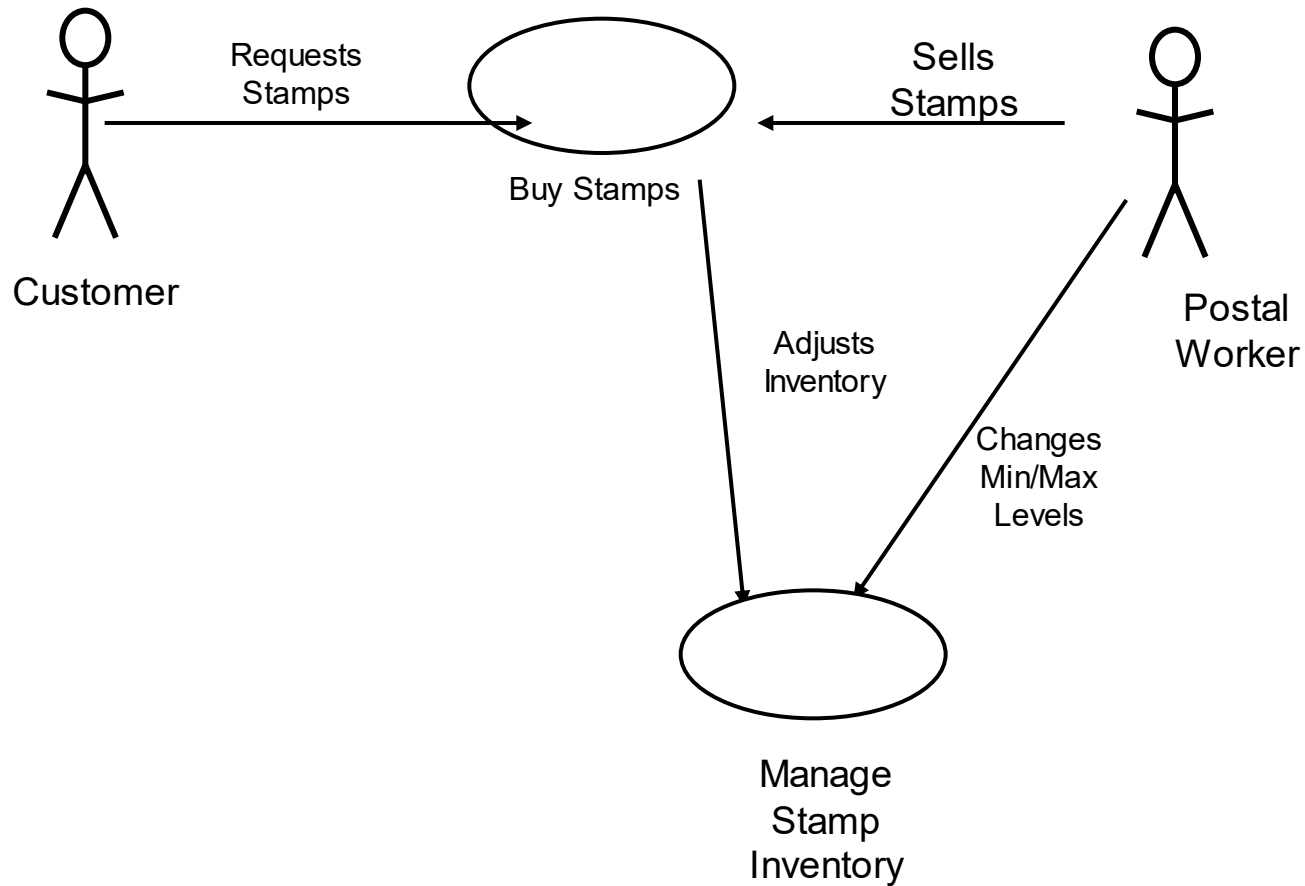
Create a use case diagram for any two use cases of the post office system

Complete a use case description document for any one use case

Actors might include Customer, Postal Worker, and Route Carrier.

Use cases might include Buy Stamps, Manage Stamp Inventory, Rent P.O. Box, Certify Mail, Deliver Mail, Sort Incoming Mail.

Example Use Cases diagram



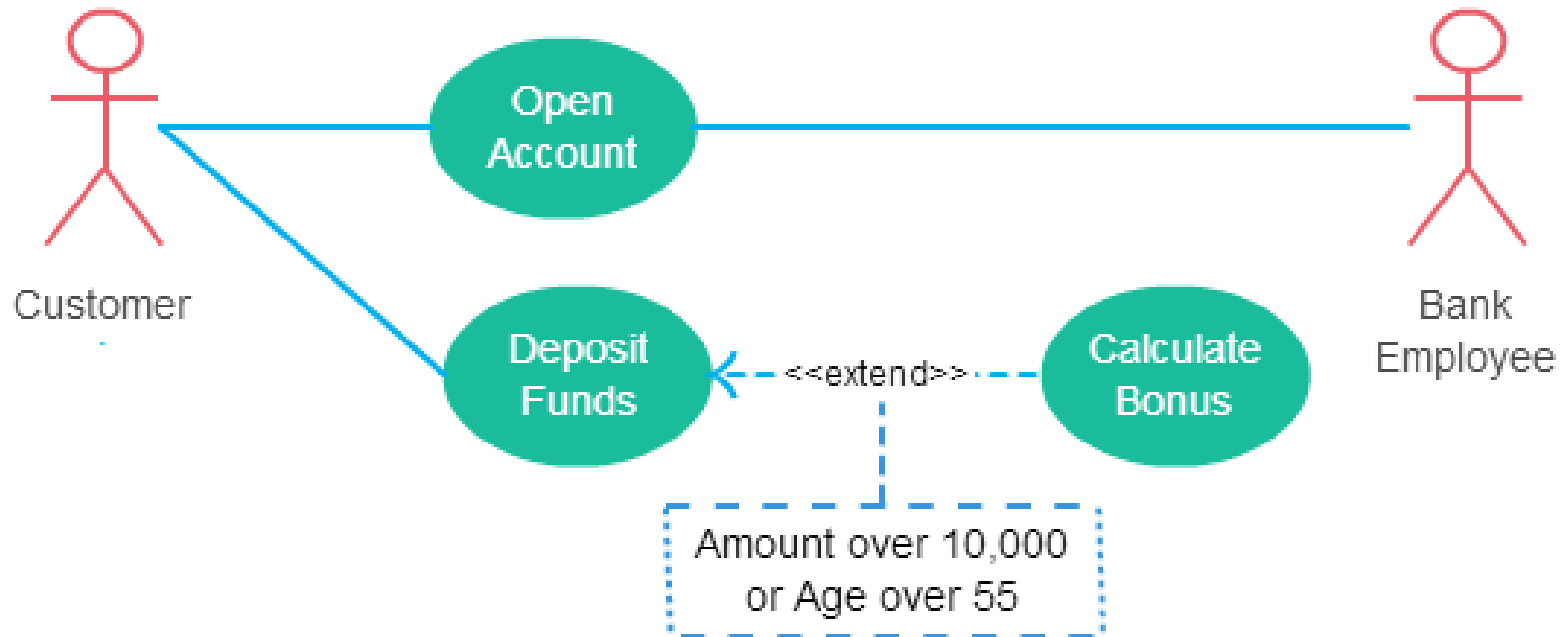
Example Use Case description document

<i>Name:</i>	<i>Buy Stamps</i>
<i>Actor:</i>	<i>Customer/Postal Worker</i>
<i>Description:</i>	<i>This use case describes the process used to buy stamps at the post office</i>
<i>Successful Completion</i>	<ol style="list-style-type: none"><i>1. Customer requests stamps</i><i>2. Postal Worker checks on availability of stamps</i><i>3. Stamps are available and customer pays fee</i><i>4. Customer receives stamps and stock is adjusted</i>
<i>Alternative:</i>	<ol style="list-style-type: none"><i>1. Customer requests stamps</i><i>2. Postal Worker checks on availability of stamps</i><i>3. Stamps are not available and customer selects alternate stamps or no stamps</i><i>4. Customer receives stamps and stock is adjusted (if alternate stamps selected) or customer leaves with no stamps</i>
<i>Pre-condition:</i>	<i>Customer wants to purchase stamps</i>
<i>Post-Condition:</i>	<i>None</i>
<i>Assumptions:</i>	<i>Customer has enough money</i>

Example Use Case description document

Use Case Number			
Use Case Name			
Author/Source			
Date of Creation			
Precondition(s)			
Successful Post Condition			
Actors			
Priority			
Related Use Cases			
Flow of Events			
Main Flow			
User Action		System Response	
UA1		SR1	
UA2		SR2	
Alternative 1: Title			
User Action		System Response	
A1.UA1		A1.SR1	
A1.UA2		A1.SR2	
Alternative 2: Title			
User Action		System Response	
A2.UA1		A2.SR1	
A2.UA2		A2.SR2	

Extend Relationship Between Two Use Cases

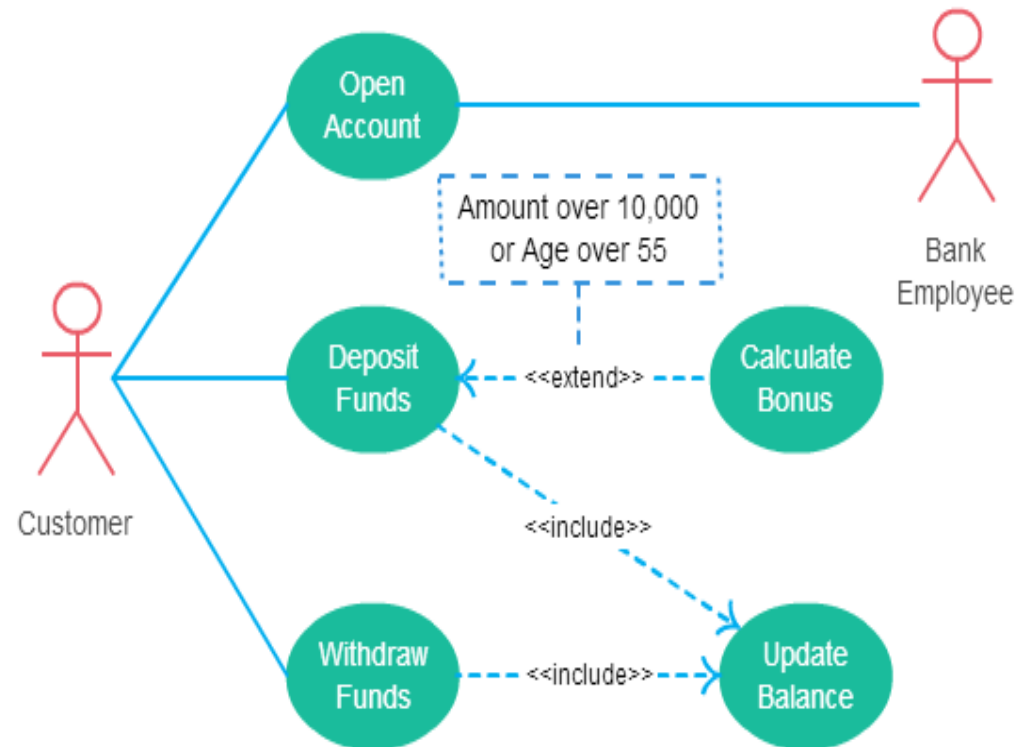


Extend Relationship Between Two Use Cases

- Many people confuse the extend relationship in use cases. As the name implies it extends the base use case and adds more functionality to the system. Here are few things to consider when using the <<**extend**>> relationship.
- **The extending use case is dependent on the base use case.** In the diagram the “Calculate Bonus” use case doesn’t make much sense without the “Deposit Funds” (base) use case.
- **The extending use case is usually optional** and can be triggered conditionally. In the diagram you can see that the extending use case is triggered only for deposits over 10,000 or when the age is over 55.
- **The base use case must be meaningful on its own.** This means it should be independent and must not rely on the behavior of the extending use case.

Include Relationship Between Two Use Cases

- Include relationship show that the behavior of the included use case is part of the base use case. The main reason for this is to reuse the common actions across multiple use cases. In some situations this is done to simplify complex behaviors.
- The base use case is incomplete without the included use case.
- The included use case is mandatory and not optional.





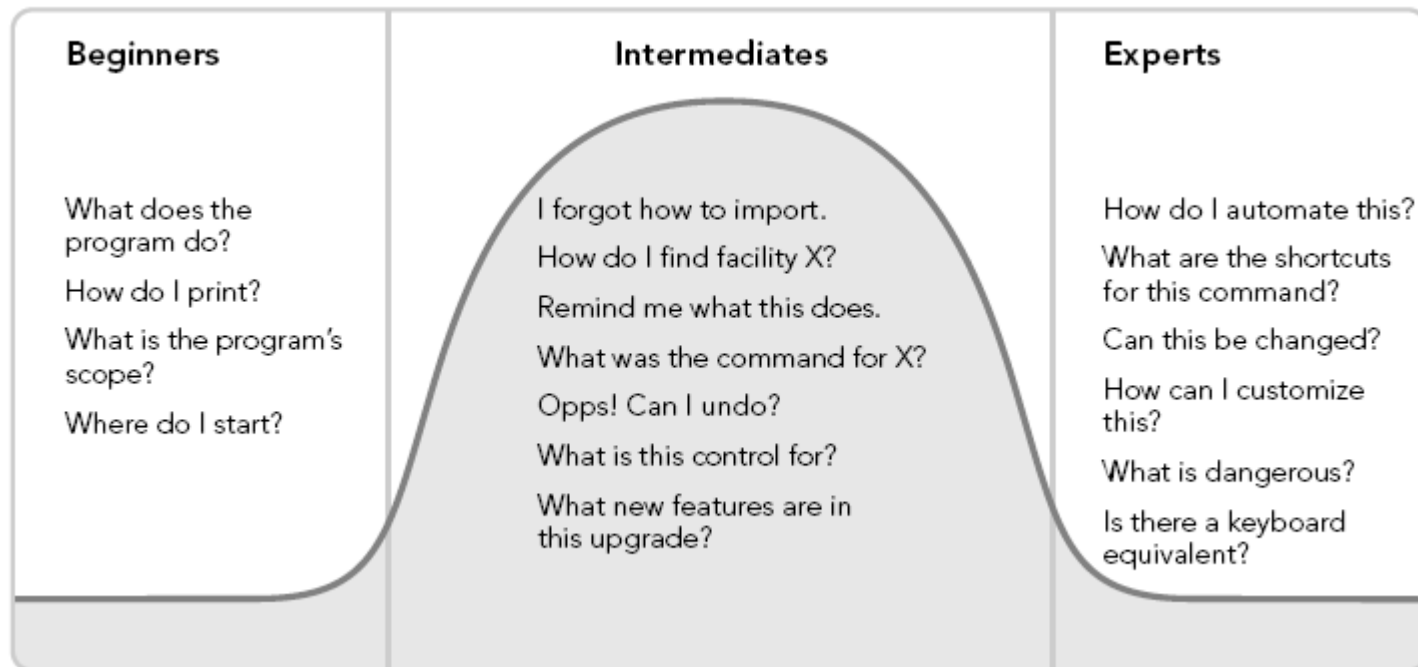
User Analysis

Area of investigation and information gathered for UI design

Focus of investigation	Information gathered
The domain	Wider specialist knowledge Specific knowledge for a computer system
The users	Who they are; focuses on the <i>real</i> (primary) users, but also considers other stakeholders (secondary users)
Characteristics of the users	Age, sex, culture, physical abilities and physical disabilities, educational background, computer/IT experience, motivation, attitude, enjoyment, satisfaction
Characteristics of the tasks	Are the tasks easy, complex, novel, variable, repetitive, frequent or infrequent, single tasks or multitasking, time critical, requiring individual or collaborative working? Are there safety issues in relation to the work?
Physical environment	Noise, stress, comfort, dirt, dust, heating, lighting, ventilation, furniture, working space, individual offices, open-plan areas, equipment layout, hazards in the workplace
Social environment	Pressure of work, individual or collaborative working, individual offices or open-plan areas

Beginners, intermediates, and expert

- Most users are neither beginners nor experts; instead, they are *intermediates*.



Designing for different experience levels

- A well-balanced user interface ...
 - Does not cater to the beginner or to the expert, but rather devotes the bulk of its efforts to satisfying all type of users.
- To rapidly and painlessly get beginners into intermediacy.
- To avoid putting obstacles in the way of those intermediates who want to become experts.
- To keep intermediates happy as they stay firmly in the middle of the skill spectrum.

What beginners need

- Nobody wants to remain a beginner.
- They need some instruction, but not very much, and the process has to be rapid and targeted.
- Beginners don't need reference information; they need overview information, such as a guided tour.
 - Standard online help is a poor tool for them.
- Beginners also rely heavily upon menus to learn and execute commands

What intermediates need

- Intermediates need access to tools.
- They don't need scope and purpose explained to them because they already know these things.
- Intermediates know how to use reference materials.
 - Online help

What experts need

- Experts want shortcuts to everything.
- Expert users constantly, aggressively seek to learn more and to see more connections between their actions and the product's behavior and representation.
- Experts appreciate new, powerful features.

UI strategies for beginners, intermediates, and expert

■ Beginners

- ☐ Few and simple features
- ☐ Lots of feedback and confirmation
- ☐ Tutorials

■ Intermediates

- ☐ Help remembering
- ☐ Easy to recognize, consistent
- ☐ On-line help with search

■ Experts

- ☐ Shortcuts
- ☐ Customization
- ☐ Little or no feedback (distracting)

User characteristics relevant to UI design

- UI should match the **characteristics** of the intended real users.
- In UI design, users (or user groups) are described in relation to their characteristics. A **profile** for each user (or user group) is created, usually through questionnaire, that includes:
 - ☐ Age
 - ☐ Sex
 - ☐ Culture
 - ☐ Physical abilities and disabilities
 - ☐ Education background
 - ☐ Computer/IT experience
 - ☐ Motivation
 - ☐ Attitude

Translating user characteristics into UI design requirements: ATM example

User characteristics	ATM UI Requirements
Age range from 12 to 80+	ATM screen height needs to accommodate users of varying height.
May be fully able-bodied or may have some physical limitations	ATM screen height needs to accommodate able-bodied users as well as users with walking sticks or those who use wheelchairs.
May have some physical limitations in relation to hearing	All user inputs should have both visual and auditory feedback.
May have some physical limitations in relation to sight	Screen text should be of a reasonably large font.
May have some physical limitations in relation to use of hands	Touchscreens, if used, should have target areas that are large enough to locate. Touchscreens, if used, should be sensitive enough to respond to users with decreased strength in fingers or hands.
Little or no experience of computer/IT use	The application should be easy to use and easy to learn.

Summary

- To understand the relationship between goals, tasks and actions.
- To understand different techniques for task analysis.
- To understand the needs of users of different experience levels
 - beginners, intermediates, and expert.