

Software Engineering

Chapter 2: Requirements using UML

الباب 2 : المتطلبات بإستعمال لغة النمذجة الموحدة

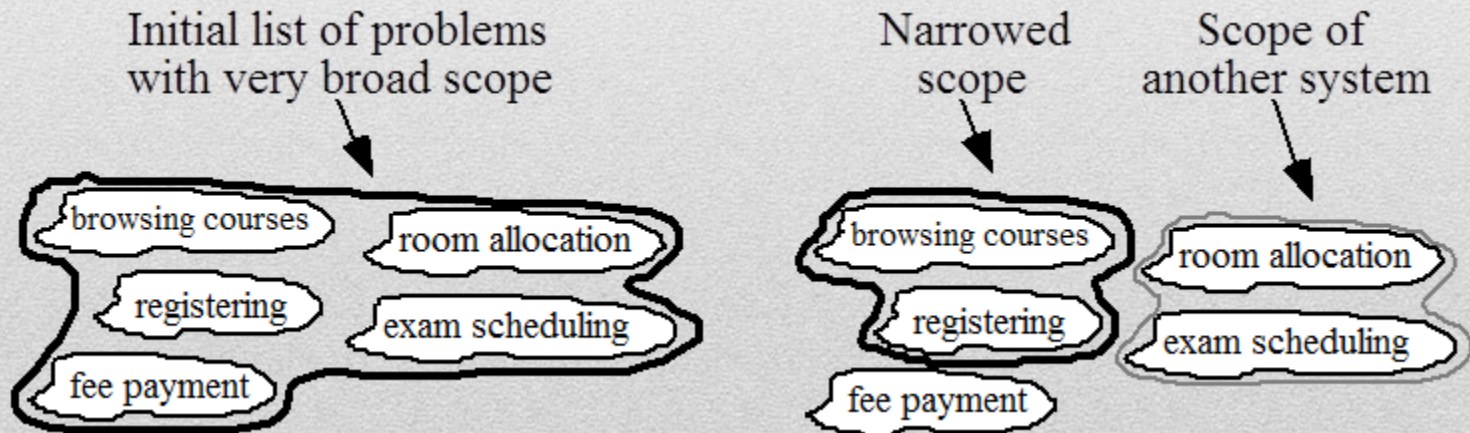
1

تحديد المجال

❖ تضيق مجال المشكلة عن طريق تحديد أكثر دقة

❖ قائمة جميع الأشياء التي قد يتصور النظام القيام بها

❖ Example: A university registration system



ما هي المتطلبات؟

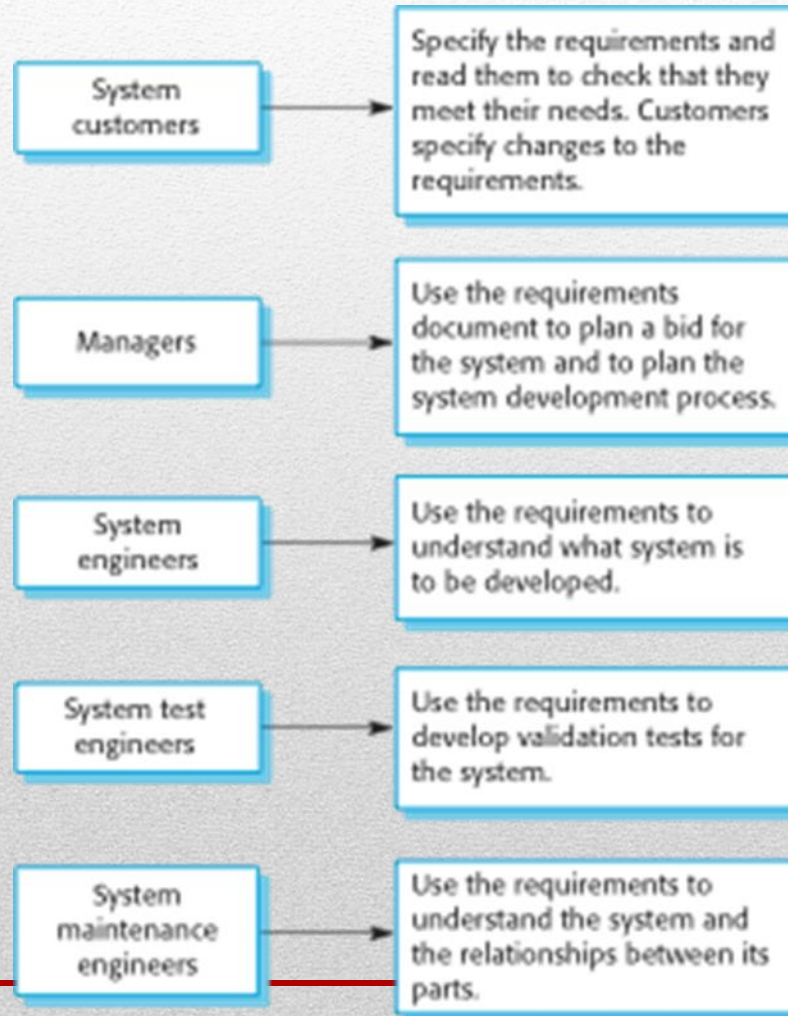
- ❖ المتطلبات بيان تصف إما :
 - (1) جانب من جوانب النظام المقترح، ما يجب القيام به، أو
 - (2) قيود موجودة على تنمية النظام.
- ❖ في كلتا الحالتين يجب أن تساهم في ايجاد طريق نحو حل مشكلة الزبون.
- ❖ تمثل مجموعة المتطلبات اتفاق تفاوضي بين أصحاب المصلحة.
- ❖ تجمع مجموعة المتطلبات في وثيقة المتطلبات (*document requirements*).

وثيقة المتطلبات

- ❖ وثيقة المتطلبات (the software requirements document) هو بيان رسمي لما هو مطلوب من مطوري النظام.
- ❖ تشمل كلا من تعريف متطلبات المستخدم ومواصفات متطلبات النظام.
- ❖ هي ليست وثيقة تصميم، تحدد ما يجب القيام به للنظام بدلا من كيف.

Users of a requirements document

مستخدمي وثيقة المتطلبات



محتوى و ثيقة المتطلبات (1)

القراء المتوقعين من الوثيقة ، تاريخ النسخ ، ملخص التغييرات التي أدخلت على كل إصدار	تمهيد (preface)
تعريف الوظائف العامة للنظام ، العلاقات مع الأنظمة الأخرى، الأهداف،	المقدمة introduction
قائمة المصطلحات التقنية المستعملة	قاموس المصطلحات (glossary)
قائمة الخدمات المقدمة ، لمحة عامة عن بنية النظام،	تعريف متطلبات المستخدمين
تبين توزيع الوظائف عبر وحدات النظام	تصميم النظام system architecture

محتوى و ثقة المتطلبات (2)

تعريف المتطلبات الوظيفية و الغير وظيفية، بطريقة مفصلة	تعريف متطلبات النظام
نماذج النظام التخطيطية التي تبين العلاقة بين مكونات النظام الداخلية والنظام وبيئته	نماذج النظام system models
التغيرات المحتملة (تغيرات تقنية، وظيفية، ...)	تعريف نمو النظام (system evolution)
تفاصيل الإحتياجات الضرورية ، مثال تفاصيل الأجهزة و قواعد البيانات، ...	الملحقات (annexes)
كالفهرسة الأبجدية للوثيقة، قائمة الأشكال، قائمة الجداول	الفهرسة (index)

Types of Requirements

- ❖ **Functional requirements**
 - Describe *what* the system should do

- ❖ **Non Functional requirements**
 - Quality requirements
 - ✓ *Constraints* on the design to meet specified levels of quality
 - Platform requirements
 - ✓ *Constraints* on the environment and technology of the system
 - Process requirements
 - ✓ *Constraints* on the project plan and development methods

Functional Requirements

- What inputs the system should accept
- What outputs the system should produce
- What data the system should store that other systems might use
- What computations the system should perform
- The timing and synchronization of the above

Non-functional requirements

❖ ثلاثة أنواع رئيسية

❖ 1. متطلبات الجودة:

- سهولة الاستخدام والكفاءة ،
- زمن الاستجابة
- الإنتاجية
- الموثوقية (Reliability)
- التوافرية أو التواجدية (Availability)
- الإسترداد من الأضرار (recovery from failure)
- القابلية للصيانة والتعزيز (maintainability)
- القابلية لإعادة الاستخدام (reusability)

Non-functional requirements

2. Platform requirements (متطلبات البيئة):

the environment and technology of the system.

- ✓ Platform
- ✓ Technology to be used

3. Process requirements: (متطلبات الإجرائية)

the project plan and development methods

- ✓ Development process (methodology) to be used
- ✓ Cost and delivery date
 - Often put in the contract or the project plan

تحديد المقاييس للمتطلبات الغير وظيفية

وحدة القياس	الخاصية
عدد المعاملات المعالجة / الثانية زمن الاستجابة	السرعة Speed
Mbytes , Gbytes	الحجم size
فترة التدريب عدد الإطارات المساعدة	سهولة الاستخدام
(Mean time to failure) (Probability of unavailability) (Rate of failure occurrence)	الموثوقية Reliability
(Time to restart after failure) (Percentage of events causing العطل في المتسببة للأحداث المتسببة في العطل failure) (Probability of data corruption on failure)	الصلابة Robustness
	الوقت لإعادة تشغيل بعد العطل النسبة المئوية للأحداث المتسببة في العطل احتمال تلف البيانات مع العطل

Use-Cases: describing how the user will use the system

- ❖ A *use case* is a typical sequence of actions that a user performs in order to complete a given task
 - The objective of *use case analysis* is to model the system
 - how users interact with this system when trying to achieve their objectives.

It is one of the key activities in requirements analysis

- A *use case model* consists of
 - ✓ a set of use cases
 - ✓ an optional description or diagram indicating how they are related

Use cases

❖ A use case should

- Cover the *full sequence of steps* from the beginning of a task until the end.
- Describe the *user's interaction* with the system ...
 - ✓ (Not the computations the system performs).
 -
- Only include actions in which the actor interacts with the computer.
 - ✓ Not actions a user does manually

Scenarios

السيناريو

❖ السيناريو هو مثال لحالة استخدام
حدوث محدد لحالة استخدام

✓ يمكن أن يكون لحالة استخدام عدة سيناريوات :

- من طرف فاعل محدد...
- في وقت محدد ...
- مع بيانات محددة.

Example

- ❖ List a minimal set of use cases for the following actors in a library system: Borrower, Checkout clerk, Librarian, Accounting system
- ❖ Borrower (المقترض) :
 - Search for an item by title, by author, by subject
 - Place a book on hold if it is on loan to somebody else
(ضع كتابا في الانتظار إذا كان على سبيل الاعارة لشخص آخر)
- ❖ Checkout Clerk (موظف المكتبة):
 - Check out an item for a borrower
 - Check in an item that has been returned
 - Renew an item
 - Add a new borrower
 - Update a borrower's personal information (address, telephone, ...)

Example

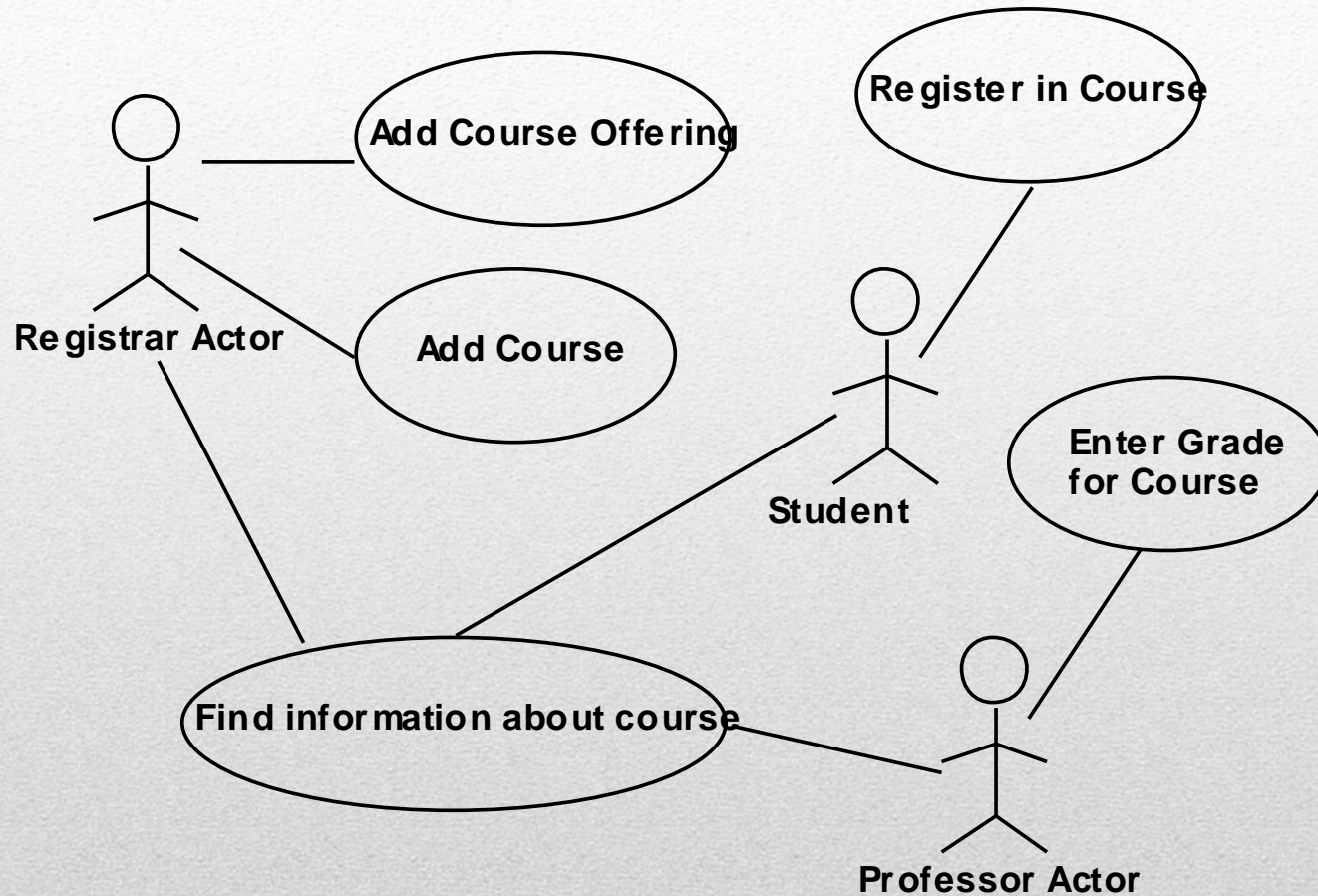
- ❖ **Librarian (أمين المكتبة) :**
 - All the borrower and checkout clerk use cases, plus,
 - Add a new item to the collection
 - Delete an item from the collection
 - Change the information the system has recorded for an item
- ❖ **Accounting system (نظام المحاسبة) :**
 - Obtain the amount of overdue fines paid by borrowers

وصف حالة استخدام

- ❖ **A. Name:** Give a short, descriptive name to the use case.
- ❖ **B. Actors (الجهات الفاعلة):** List the actors who can perform this use case.
- ❖ **C. Goals:** Explain what the actor or actors are trying to achieve.
- ❖ **D. Pre-conditions:** State of the system before the use case.
- ❖ **E. Summary:** Give a short informal description.
- ❖ **F. Related use cases.**
- ❖ **G. Steps:** Describe each step using a 2-column format.
- ❖ **H. Post-conditions:** State of the system in following completion.

- ❖ A and G are the most important

Use case diagrams



Example

- ❖ **Use case:** Check out an item for a borrower
- ❖ **Actors:** Checkout clerk (regularly), chief librarian (occasionally)
- ❖ **Goals:** To help the borrower to borrow the item if they are allowed, and to ensure a proper record is entered of the loan
- ❖ **Preconditions:** The borrower must have a valid card and not owe any fines. The item must have a valid barcode and not be reserved.

- ❖ **Steps:**

- Actors actions

1. Scan item's barcode and barcode of borrower's card.
3. Stamp item with due date
4. Confirm that the loan is to be initiated.

- System response

2. Display confirmation that the loan is allowed.
5. Display confirmation that the loan has been recorded.

Postconditions: The system has a record of the fact that the item is borrowed and the date is due.

Extensions – Generalizations - Inclusions

❖ Extensions (الإمتداد)

- تستخدم لجعل التفاعلات الإختيارية (optional interactions) مبينة (explicit) أو التعامل مع حالات استثنائية.

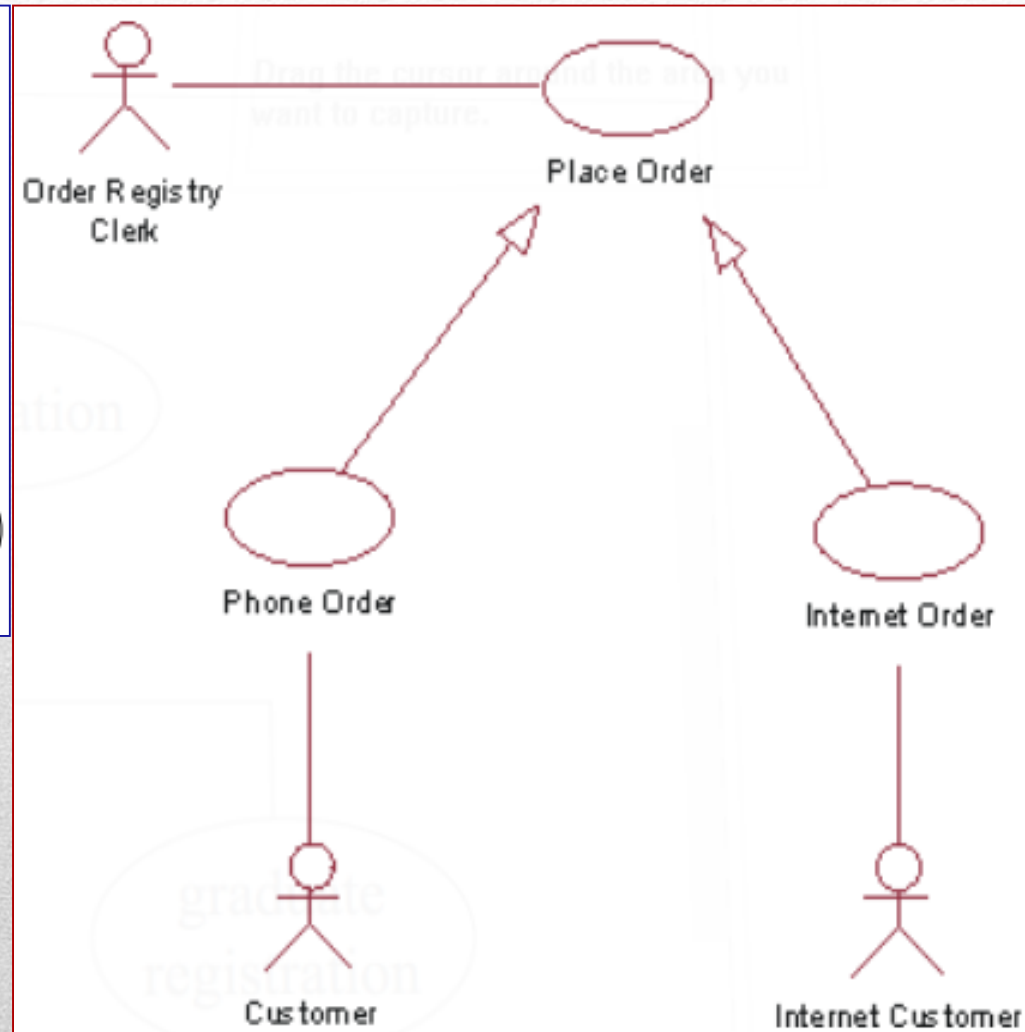
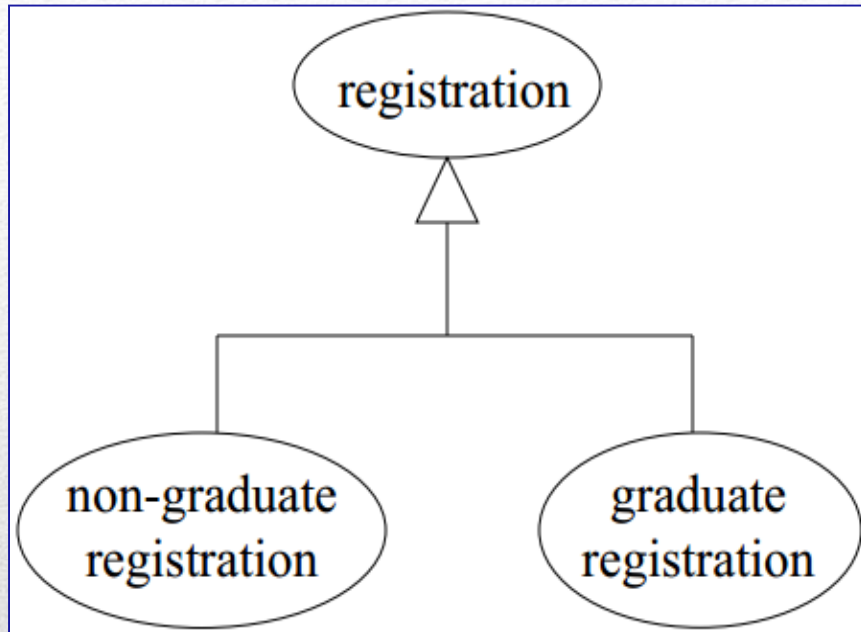
❖ Generalizations (التعميمات)

- هناك حالة استخدام معممة تمثل العديد من حالات استخدام متماثلة.
- عدة حالات إستخدام يمكنها أن تظهر كحالة إستخدام مشتركة.
- The child use case inherits the behavior and meaning of the parent use case.
- The child may add or override the behavior of its parent.

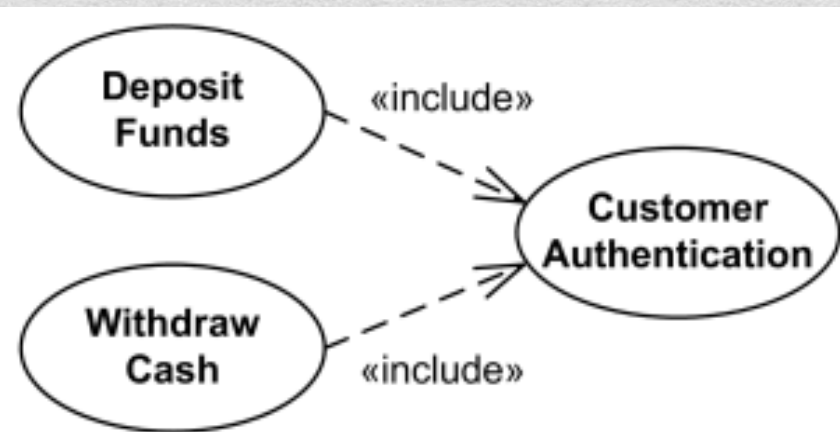
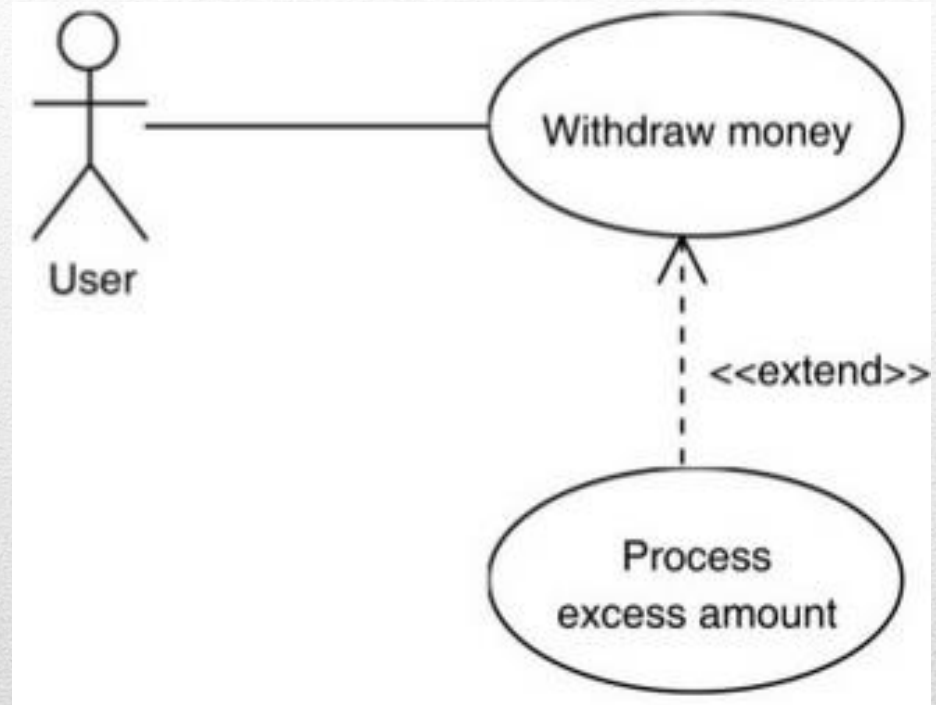
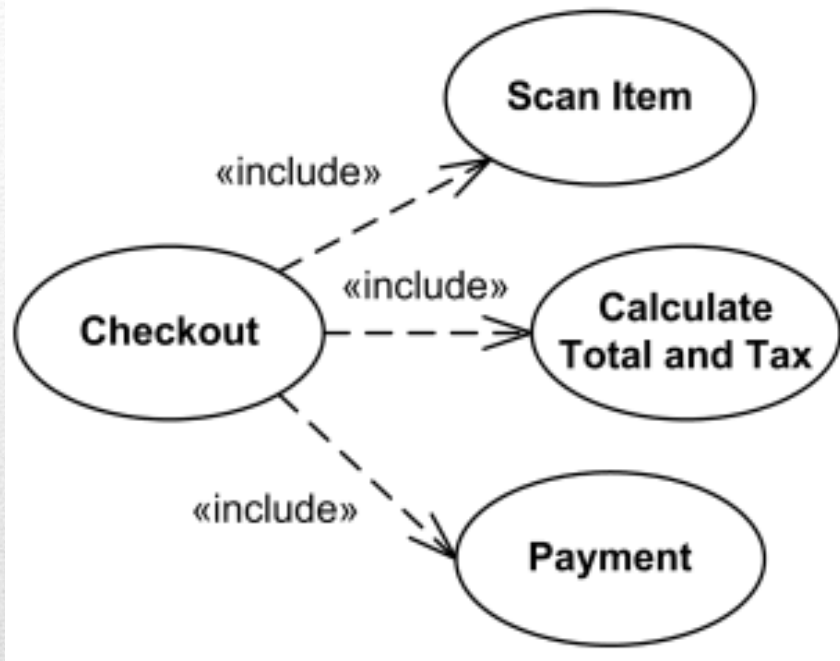
❖ Inclusions (المشتملات)

- تسمح للتعبير عن مكونات حالة إستخدام لتمكين تعيين القواسم المشتركة بين عدة حالات إستخدام.
- ✓ يمكن لحالات إستخدام مختلفة تشترك في سلسلة من الإجراءات.
- ✓ تمكنك من تجنب تكرار التفاصيل في حالات إستخدام متعددة .

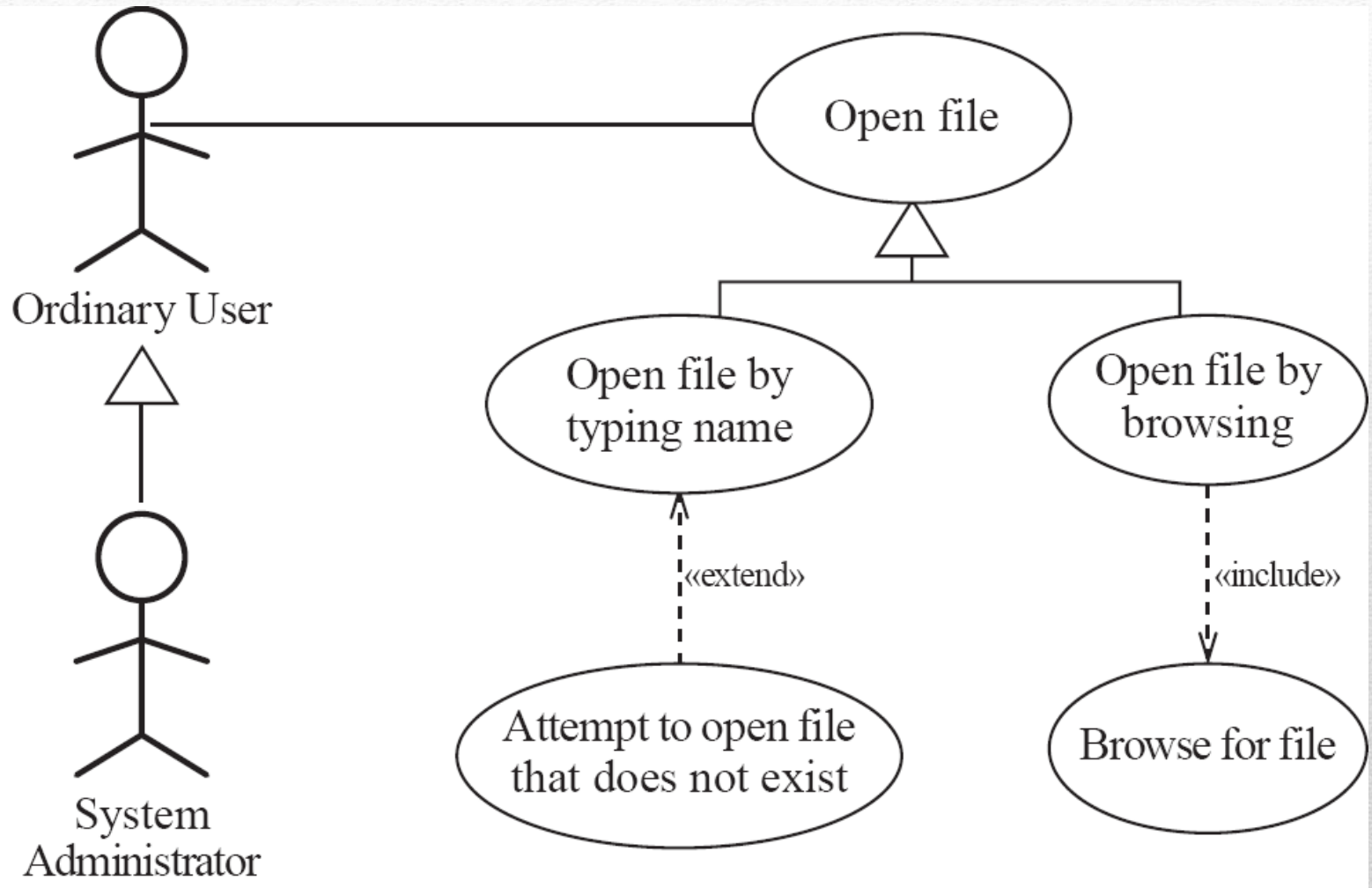
generalization



Inclusions , Extensions



Example of generalization, extension and inclusion



Example description of a use case

Use case: Open file

Related use cases:

Generalization of:

- Open file by typing name
- Open file by browsing

Steps:

Actor actions

1. Choose 'Open...' command
3. Specify filename
4. Confirm selection

System responses

2. File open dialog appears
5. Dialog disappears

Example (continued)

Use case: Open file by typing name

Related use cases:

Specialization of: Open file

Steps:

Actor actions

1. Choose 'Open...' command
- 3a. Select text field
- 3b. Type file name
4. Click 'Open'

System responses

2. File open dialog appears
5. Dialog disappears

Example (continued)

Use case: Open file by browsing

Related use cases:

Specialization of: Open file

Includes: Browse for file

Steps:

Actor actions

1. Choose 'Open...' command
3. Browse for file
4. Confirm selection

System responses

2. File open dialog appears
5. Dialog disappears

Example (continued)

Use case: Attempt to open file that does not exist

Related use cases:

Extension of: Open file by typing name

Actor actions

1. Choose 'Open...' command
- 3a. Select text field
- 3b. Type file name
4. Click 'Open'

6. Correct the file name
7. Click 'Open'

System responses

2. File open dialog appears

5. System indicates that file does not exist

- 8 Dialog disappears

Example (continued)

Use case: Browse for file (inclusion)

Steps:

Actor actions

1. If the desired file is not displayed, select a directory
3. Repeat step 1 until the desired file is displayed
4. Select a file

System responses

2. Contents of directory is displayed

The benefits of basing software development on use cases

- تساعد على تحديد مجال النظام.
- أن تستخدم للتخطيط لعملية التطوير.
- أن تستخدم في التطوير والتحقق من صحة المتطلبات.
- تشكل أساسا لتعريف حالات الاختبار.
- تستخدم لهيكل دليل المستخدم

❖ They can

- Help to define the *scope* of the system
- Be used to *plan* the development process
- Be used to both develop and validate the requirements
- Form the basis for the definition of test cases
- Be used to structure user manuals