

Progettazione del Software

Emiliano Casalicchio

*Dipartimento di Informatica e Sistemistica
SAPIENZA Università di Roma – Sede di Rieti*

<http://www.ce.uniroma2.it/courses/PSW>

(Basato su materiale didattico di G.De Giacomo)

La fase di analisi

- Cosa è l'analisi
- Gli Use Case: definizione e utilizzo

Cos'è l'analisi

- L'analisi è la fase del ciclo di sviluppo del software caratterizzata da:

INPUT: requisiti raccolti

OUTPUT: **schema concettuale** (anche detto modello di analisi) dell'applicazione

OBIETTIVO:

- costruire un modello dell'applicazione che sia **completo, preciso e rigoroso** ma anche **leggibile**, indipendente da linguaggi di programmazione e **traducibile** in un programma
- concentrarsi su **cosa**, e non su come (indipendenza da aspetti realizzativi/tecnologici)

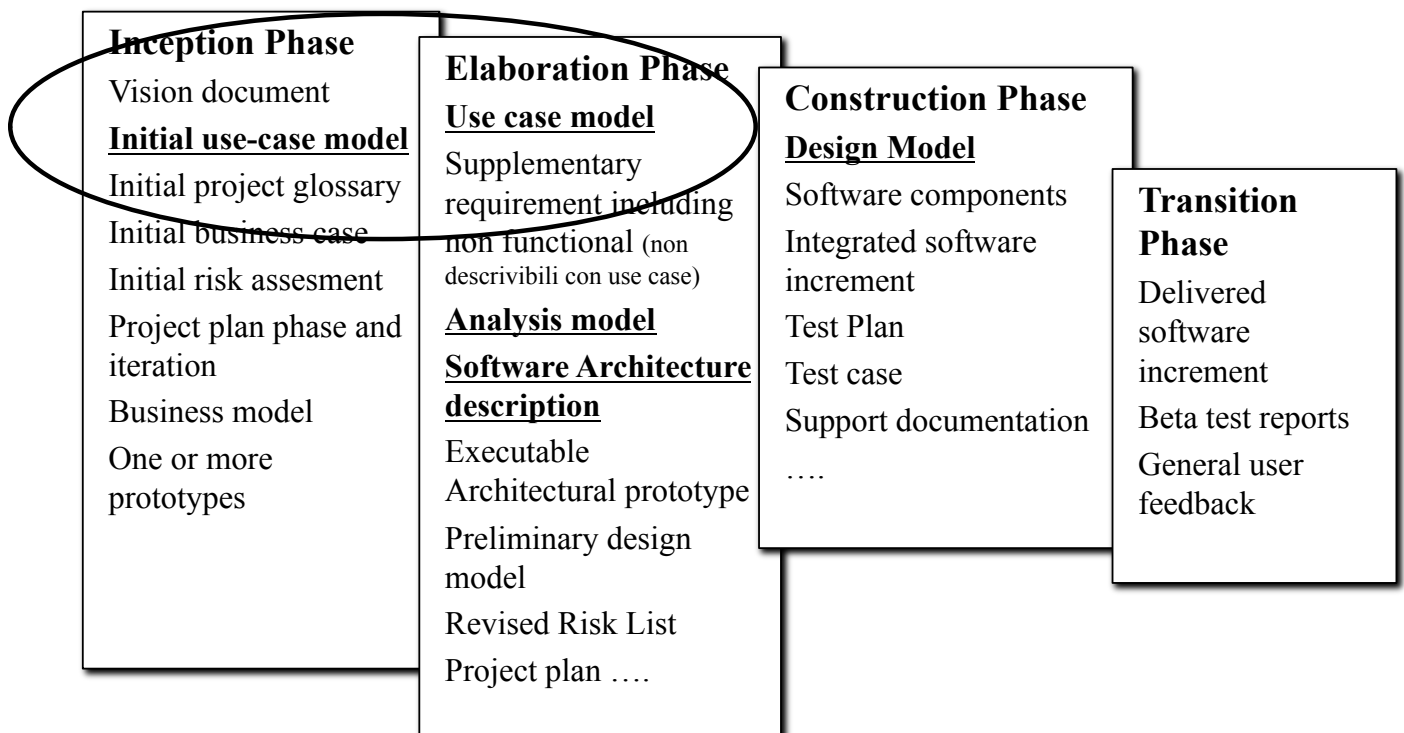
A cosa serve l'analisi

- Analizzare i requisiti:
 - coglie le loro implicazioni,
 - li specifica con l'obiettivo di formalizzarli e di eliminare incompletezze, inconsistenze e ambiguità
- Crea un modello (**schema concettuale**) che sarà un riferimento per tutte le fasi successive del ciclo di vita del software
- Verifica i requisiti con l'utente finale
- Prende decisioni fondamentali sulla strutturazione e sulla modularizzazione del software
- Fornisce la specifica delle funzionalità da realizzare

Che cosa è lo schema concettuale

- Lo schema concettuale è costituito da:
 - Il **diagramma degli use-case**
 - Descrive le funzionalità fondamentali che il sistema deve realizzare, in termini di scenari di utilizzo del sistema
 - Il **diagramma delle attività**
 - descrive il processo
 - Il **diagramma degli stati e delle transizioni (State Machine Diagram)**
 - Descrive, per le classi significative, il tipico ciclo di vita delle sue istanze
 - Il **diagramma delle classi e degli oggetti**
 - Descrive le classi dell'applicazione e le loro proprietà; descrive anche gli oggetti particolarmente significativi
 - I **documenti di specifica**
 - Descrivono con precisione quali condizioni devono soddisfare i programmi che realizzano il sistema
 - Viene prodotto un documento di specifica per ogni classe, ed un documento di specifica per ogni use case

UP, UML e Analisi



Seconda Parte

La fase di analisi

- Cosa è l'analisi
- Gli Use Case: definizione e utilizzo
- I Class e Object diagram: definizione e utilizzo

Il diagramma degli use case

Il **diagramma degli use-case** descrive le funzionalità fondamentali che il sistema deve realizzare, in termini di scenari di utilizzo del sistema

Use case

- Il diagramma degli use-case descrive le funzionalità fondamentali che il sistema deve realizzare, in termini di scenari di utilizzo del sistema
- Un use case rappresenta una **tipica interazione** tra un utente ed il sistema software da realizzare.
- Un use case cattura una qualche **funzione visibile dall'utente**, e la sua descrizione si ottiene attraverso l'interazione tra analista ed utente in fase di analisi.
- In altre parole, un Use Case definisce un particolare modo di utilizzare il sistema, il quale offre servizi e funzionalità in risposta a eventi prodotti da attori esterni.

Use case

Un use case modella un processo (o un insieme di processi) che è **trasversale** rispetto alle classi, cioè **coinvolge più classi allo stesso livello**, e sarebbe una forzatura modellarlo come una operazione di una singola classe

Un use case è in genere composto da diverse **operazioni**, che non vengono definite in modo dettagliato nel diagramma. Vedremo, quando parleremo di “specificazione di un use case”, come queste operazioni vengono definite

Use Case: Actors

Un **use case** è formulato sulla base delle **funzionalità** offerte dal sistema così come sono percepite dagli utenti.

Oltre agli use case, un altro componente fondamentale del diagramma degli use case è l'attore. Un **actor** è un ruolo che un utente (una persona o un sistema esterno) gioca interagendo con il sistema.

- Lo stesso utente può essere rappresentato da più di un attore (può giocare più ruoli).
- Più utenti possono essere rappresentati dallo stesso attore.

Use case: diagramma

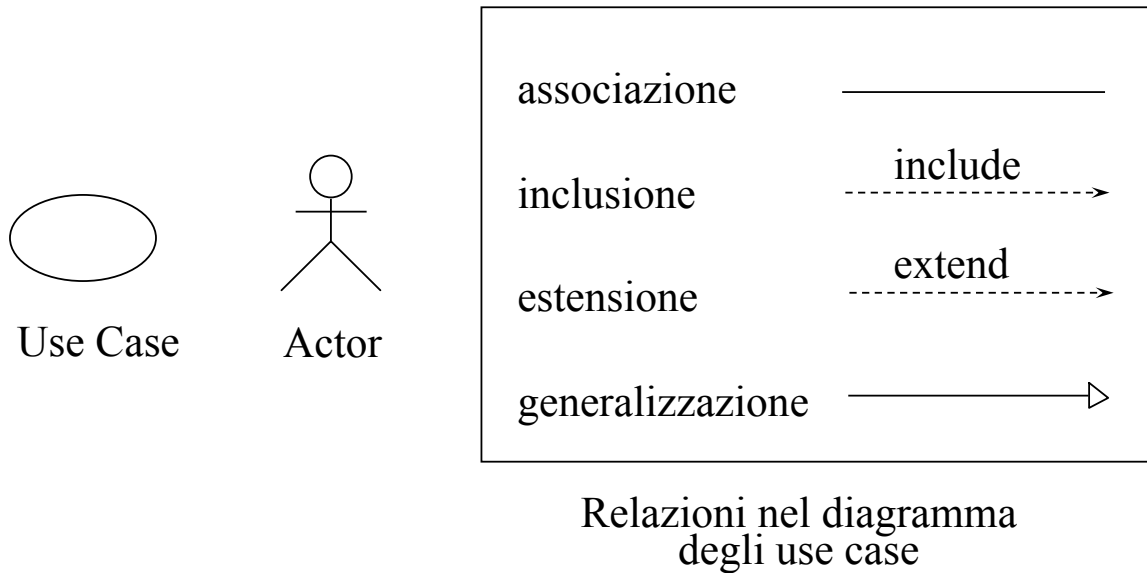
Un **diagramma degli use case** è un **grafo** i cui **nodi** possono essere

- Attori
- Use Case

mentre gli **archi** rappresentano

- la comunicazione tra gli attori e gli use case,
- i legami d'uso tra use case
- l'estensione di uno use case da parte di un altro
- la generalizzazione di un use case da parte di un altro

Componenti di un diagramma degli use case



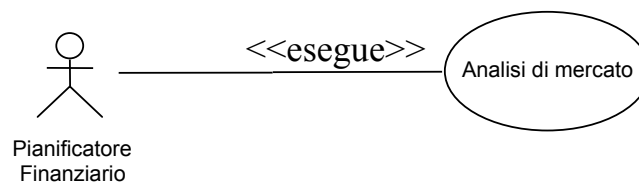
Stereotype

- Indica un'uso speciale o intento d'uso e può essere applicato a quasi tutti gli elementi UML
- Uno stereotype
 - modifica il significato di un elemento UML
 - descrive il ruolo di un elemento nel modello
- Ad esempio vedremo <<extend>> e <<include>> che modificano il significato di relazione tra use case

Associazione

La partecipazione di un attore ad uno Use Case è rappresentata da un arco di **associazione** (con un nome) tra il simbolo dell'attore e il simbolo di Use Case. In generale, questo significa che l'attore "comunica" con lo Use Case.

Il nome dell'associazione non è necessario.



Esempio Guida

Si vuole modellare un sistema di gestione delle iscrizioni ad un corso di laurea. Si vogliono modellare gli studenti (con nome, cognome, numero di matricola, età), il corso di laurea in cui sono iscritti, ed i corsi di cui hanno sostenuto l'esame, con il professore che ha verbalizzato l'esame, ed il voto conseguito. Di ogni corso di laurea interessa il codice e il nome. Di ogni corso interessa il nome e la disciplina a cui appartiene (ad esempio: matematica, fisica, informatica, ecc.). Di ogni professore interessa codice ed età.

Al momento dell'iscrizione, lo studente specifica il corso di laurea a cui si iscrive.

Al momento dell'iscrizione lo studente paga la retta di iscrizione e riceverà regolare fattura emessa dalla segreteria.

Dopo l'effettuazione di un esame, il professore comunica l'avvenuta verbalizzazione dell'esame con i dati relativi (studente, corso, voto).

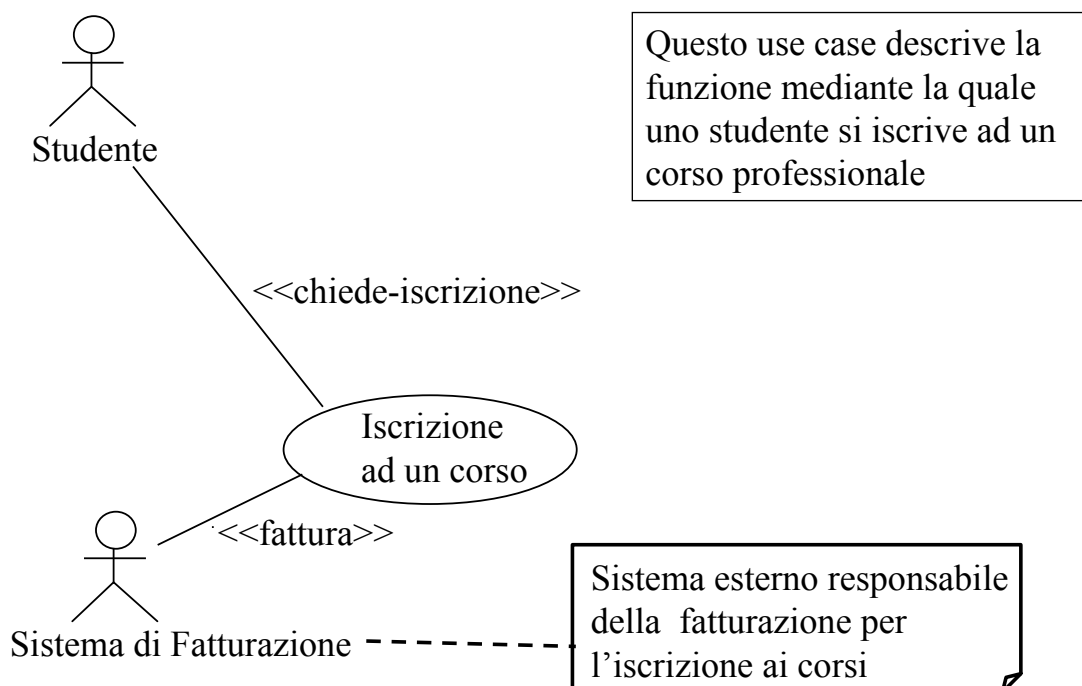
La segreteria vuole periodicamente calcolare la media dei voti di uno studente, e il numero di studenti di un corso di laurea.

Esempio Guida

Si vuole modellare un sistema di gestione delle iscrizioni ad un corso di laurea.

- *Al momento dell'iscrizione, lo studente specifica il corso di laurea a cui si iscrive.*
- *Al momento dell'iscrizione lo studente paga la retta di iscrizione e riceverà regolare fattura emessa dalla segreteria.*

Esempio di diagramma degli use case



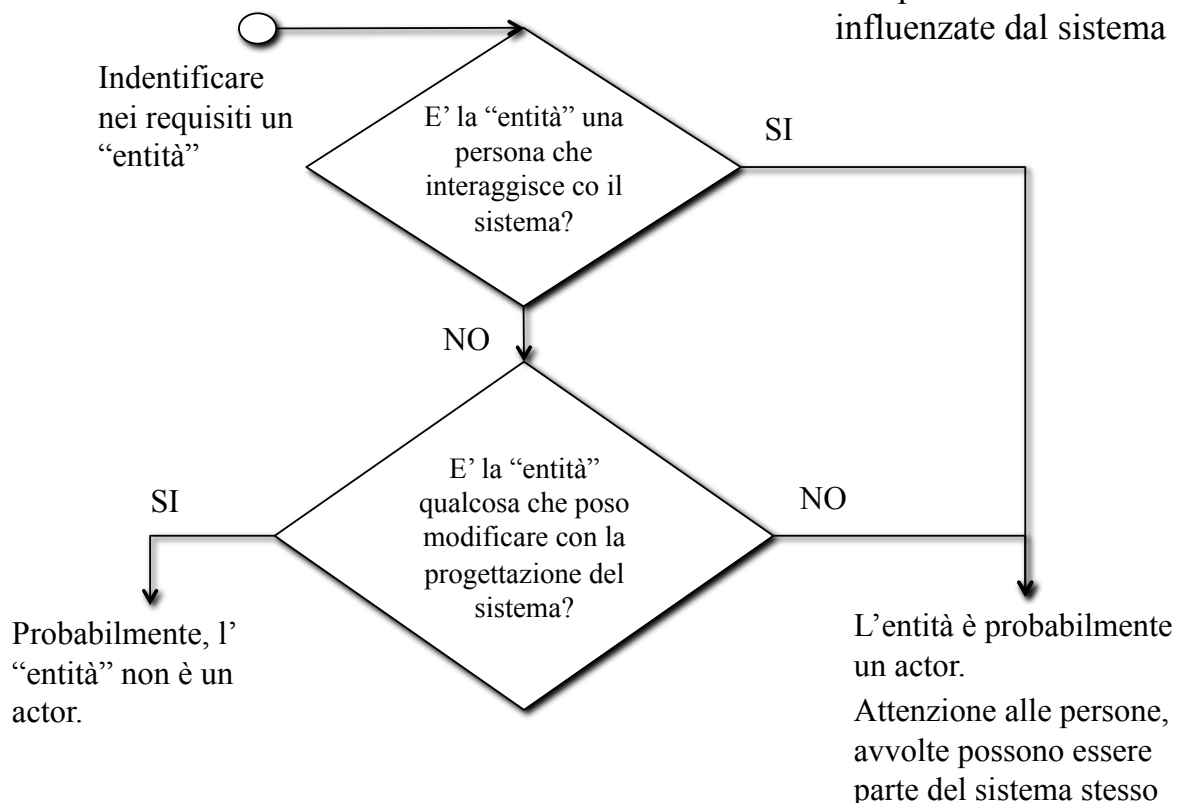
Come ottenere un Use Case diagram?

Lo use case definisce una “storia” stilizzata di come un end-user interagisce con il sistema.

1. Definire gli attori
2. Definire gli use case (principali domande a cui rispondere)
 1. Chi sono gli actor principali e secondari
 2. Quali sono gli obiettivi degli actor
 3. Quali sono le precondizioni (non rappresentate graficamente)
 4. Quali sono le principali azioni richieste/svolte dagli actor
 5. Quali eccezioni vanno considerate nello svolgimento della storia
 6. Quali variazioni vanno considerate nelle interazioni con gli actor
 7. Quali sono le informazioni acquisite, prodotte e modificate dagli attori

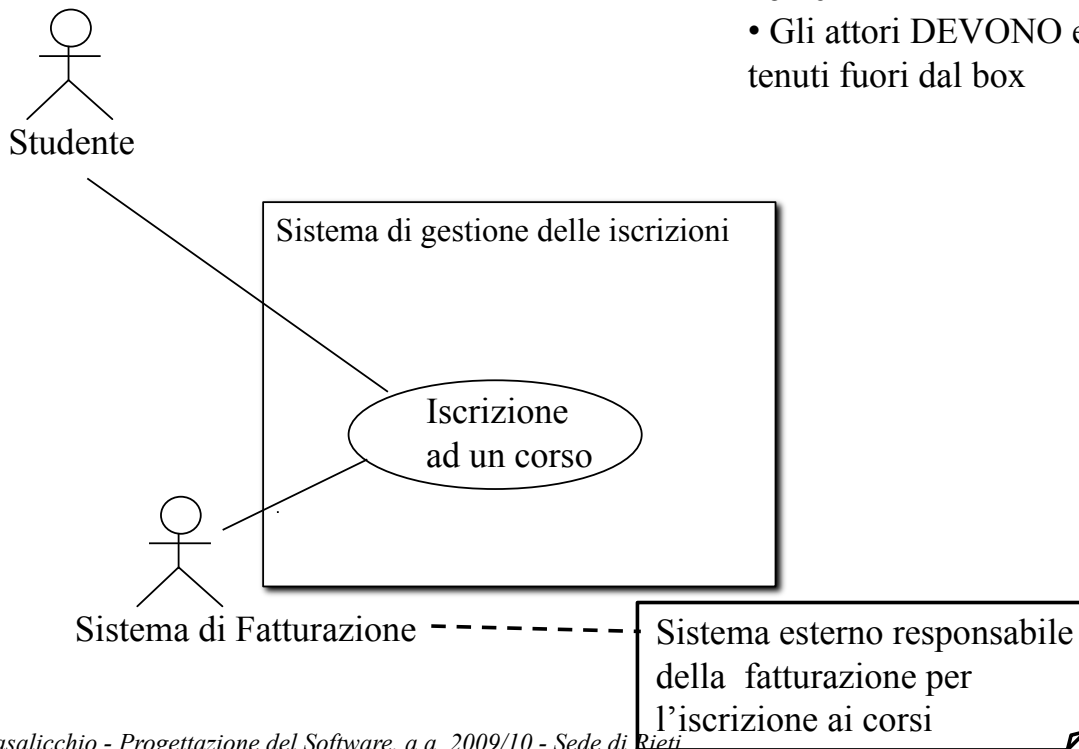
Come riconoscere gli attori

- Tipicamente gli attori sono “cose”/“entità” che non possono essere influenzate dal sistema



Confini di un Use Case

- Sono impliciti, ma possono essere evidenziati con un box al quale viene assegnato un nome
- Gli attori DEVONO essere tenuti fuori dal box

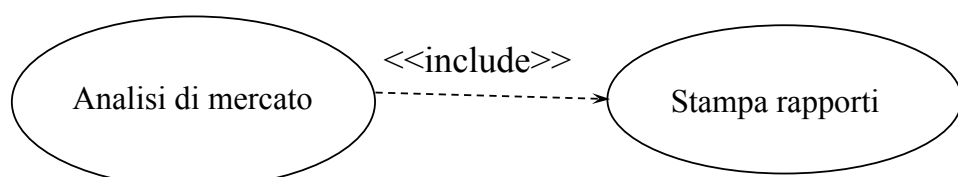


E.Casalicchio - Progettazione del Software, a.a. 2009/10 - Sede di Rieti

23

Inclusione

- Una relazione d'inclusione tra use case è rappresentata da una **freccia** tra lo use case che include e quello che è incluso.
- La freccia è etichettata con **<<include>>**.
- Una relazione d'inclusione da un Use Case *A* ad un Use Cases *B*, indica che **ogni istanza dello Use Case *A* includerà anche il comportamento specificato per lo Use Case *B***.
- In altre parole, qualche funzionalità di *A* richiede di **servirsi** di qualche funzionalità di *B*.



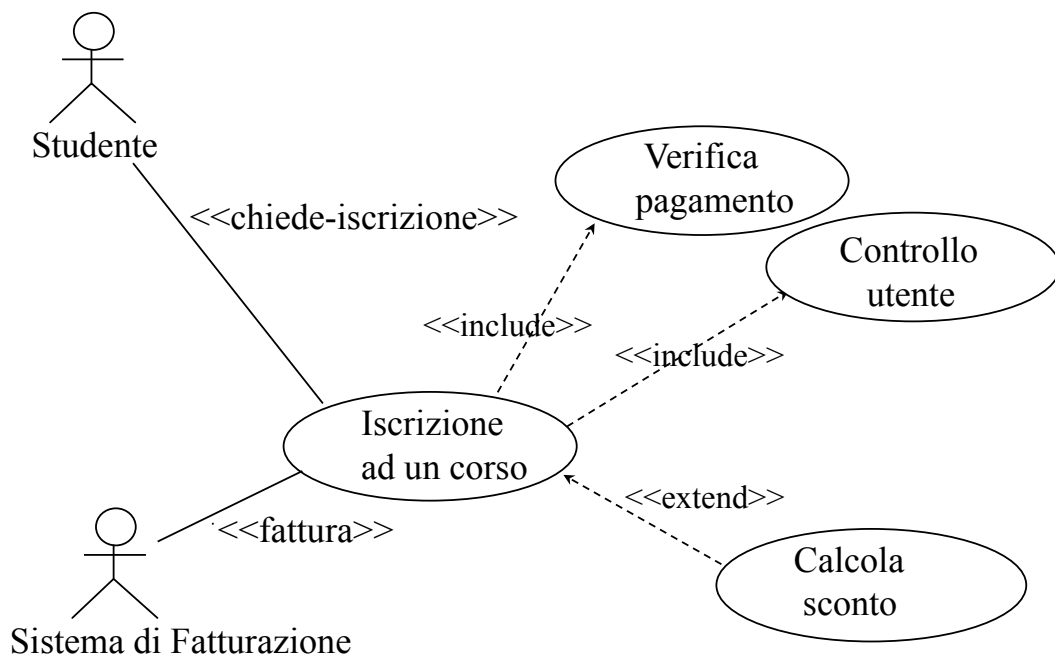
E.Casalicchio - Progettazione del Software, a.a. 2009/10 - Sede di Rieti

24

Estensione

- La relazione “estende” tra use case è rappresentata da una **freccia** etichettata con “**extend**” dallo Use Case che definisce l’estensione allo Use Case base.
- La relazione “extend” tra un Use Case *A* ed un Use Case *B* indica che **ogni istanza di B, in condizioni particolari, viene esteso con le funzionalità di una istanza di A.**
- Per uno stesso Use Case, i comportamenti definiti attraverso diverse estensioni possono occorrere all’interno di ogni singola sua istanza.

Esempi di extend e include



Generalizzazione

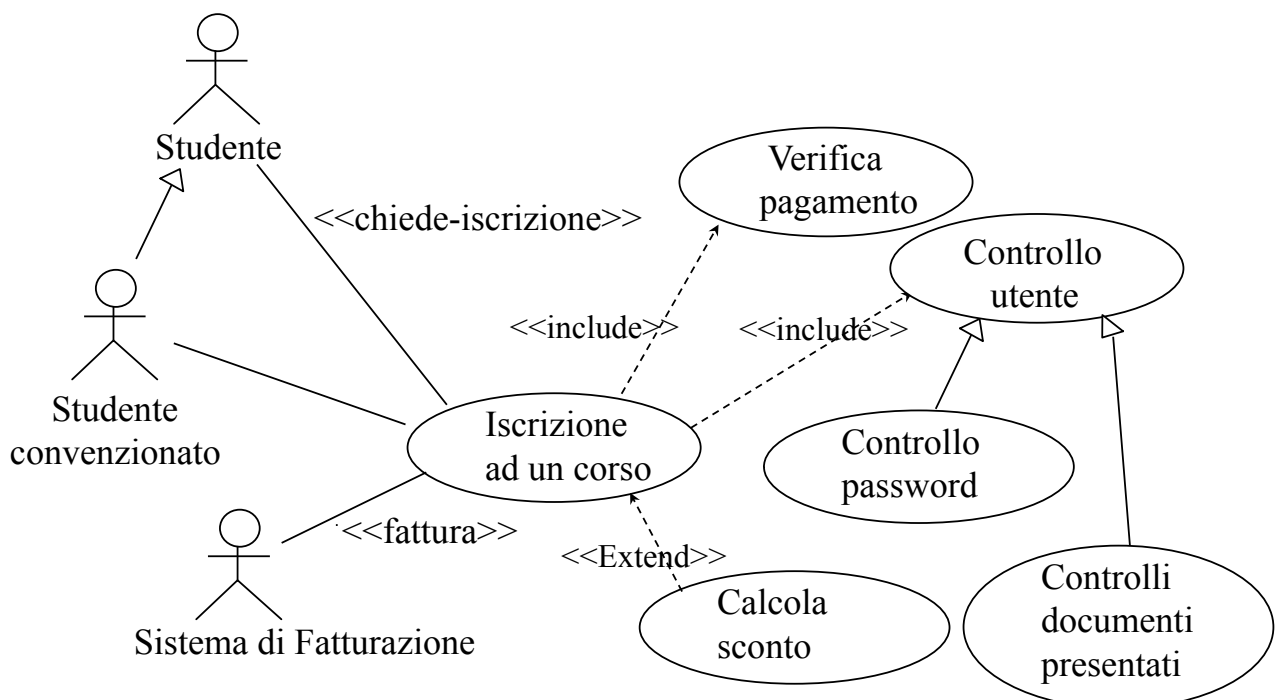
La generalizzazione si applica sia ad attori sia a use case.

Un attore A è generalizzazione di un attore B quando B è visto come un caso particolare di A .

Analogamente, **un Use Case A è generalizzazione di un Use Case B quando B è visto come un caso particolare di A .**

Il concetto di generalizzazione è simile al concetto di ereditarietà nella OOP.

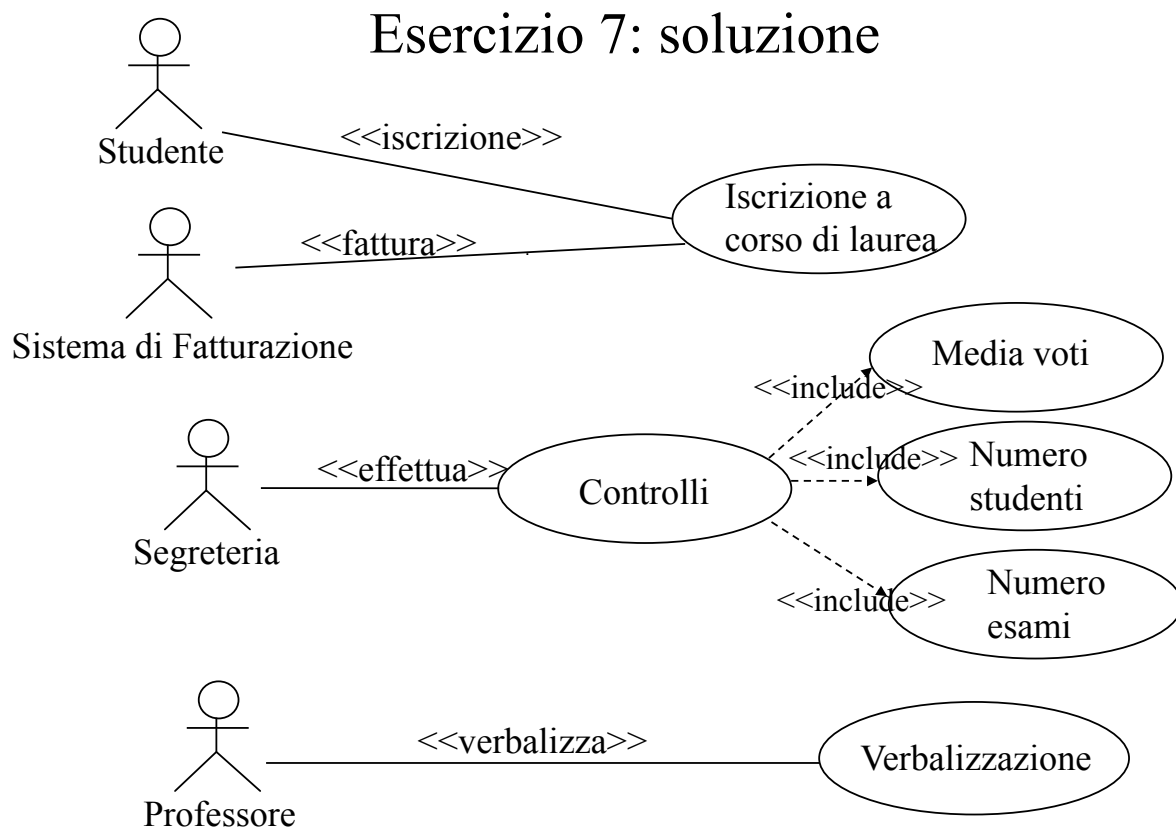
Esempi di extend, include e generalizzazione



Esercizio 7

Si vuole modellare un sistema di gestione delle iscrizioni ad un corso di laurea.

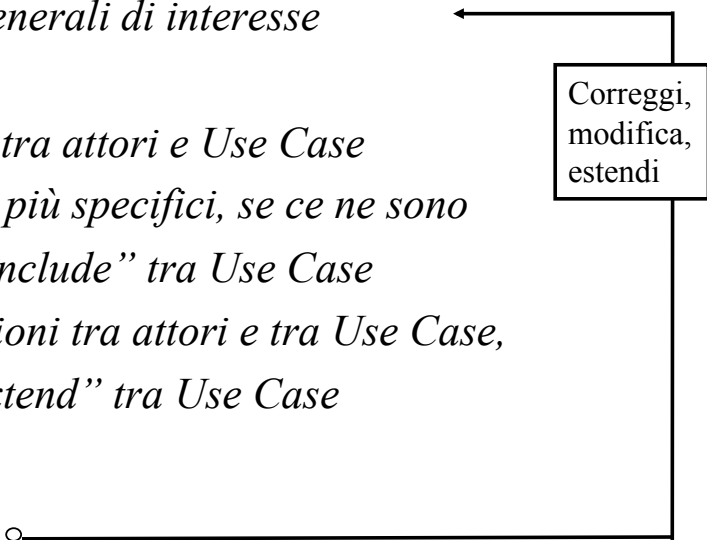
- Al momento dell'iscrizione, lo studente specifica il corso di laurea a cui si iscrive.
- Al momento dell'iscrizione lo studente paga la retta di iscrizione e riceverà regolare fattura emessa dalla segreteria.
- Dopo l'effettuazione di un esame, il professore comunica l'avvenuta verbalizzazione dell'esame con i dati relativi (studente, corso, voto).
- La segreteria vuole periodicamente calcolare la media dei voti di uno studente, e il numero di studenti di un corso di laurea.



Aspetti metodologici nella costruzione del diagramma degli Use Case

Un metodo comunemente usato per costruire il diagramma degli Use Case prevede i seguenti passi

- *Individua gli Use Case generali di interesse*
- *Individua gli attori*
- *Individua le associazioni tra attori e Use Case*
- *Individua altri Use Case, più specifici, se ce ne sono*
- *Determina le relazioni “include” tra Use Case*
- *Individua le generalizzazioni tra attori e tra Use Case,*
- *Individua le relazioni “extend” tra Use Case*
- *Controllo di qualità*



Correggi,
modifica,
estendi

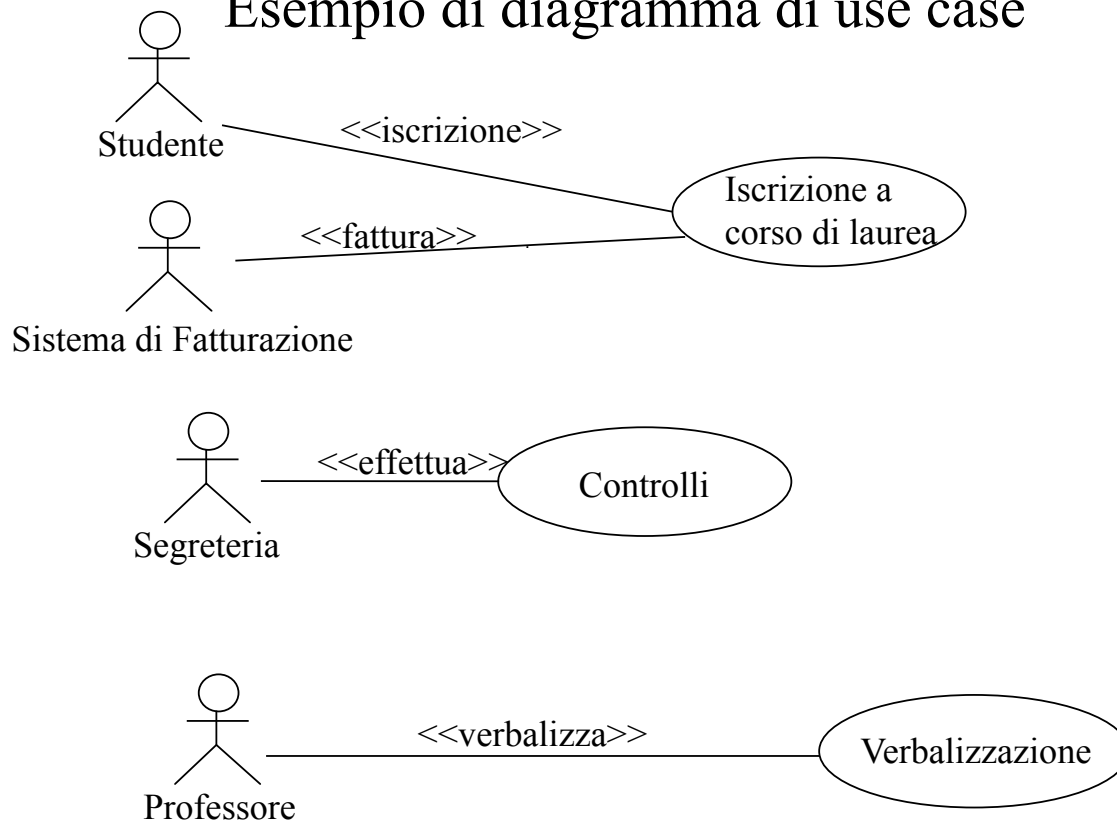
Controllo di qualità del diagramma degli Use Case

- *Sono stati colti tutti gli aspetti insiti nei requisiti?*
- *Sono state individuate tutte le associazioni e tutte le generalizzazioni?*
- *Ci sono ridondanze nel diagramma?*
- *È alta la coesione dei singoli Use Case?*
- *È basso l'accoppiamento tra diversi Use Case?*

Descrizione Use case

Use case name	Iscrizione	
Related Requirement		
Goal In context		
Precondition		
Successful end condition		
Failed end condition		
Primary actors		
Secondary actors		
Trigger		
Main flow	Step	Action
	1	A1
	2	A2
	3	A3
Extentions	Step	Branching Action
	3.1	A3.1

Esempio di diagramma di use case



Use Case name	Iscrizione a corso di laurea	
Related Requirement	Al momento dell'iscrizione, lo studente specifica il corso di laurea a cui si iscrive. Al momento dell'iscrizione lo studente paga la retta di iscrizione e riceverà regolare fattura emessa dalla segreteria.	
Goal In context	Uno studente richiede di iscriversi ad un corso di laurea	
Precondition	Nessuna	
Successful end condition	Viene creato un record per lo studente; viene emessa fattura	
Failed end condition	Non viene emessa fattura; non viene creato record studente; viene creato record studente con dati parziali	
Primary actors	Studente	
Secondary actors	Sistema di fatturazione	
Trigger	Lo studente attiva richiesta di iscrizione mediante pagina web	
Main flow	Step	Action
	1	L'utente accede alla pagina web delle iscrizioni
	2	L'utente seleziona il corso di laurea prescelto
	3	L'utente immette i propri dati
Extentions	Step	Branching Action
	2.1	L'utente richiede un percorso specifico

La specifica

Lo schema concettuale viene alla fine corredato da una **specifica** per ogni **Use Case**

La specifica di un Use Case ha lo scopo di definire precisamente il comportamento di ogni operazione di cui lo Use Case è costituito

Specifica di un Use Case

La specifica di un Use Case si fornisce facendo la lista delle operazioni (una o più) che costituiscono lo Use Case stesso, e fornendo poi la specifica di ogni operazione. La specifica di un Use Case D ha la seguente forma:

InizioSpecificaUseCase D
Specifica della operazione 1
...
Specifica della operazione N
FineSpecifica

Specifica di una operazione

Che sia una operazione di una classe o una operazione di un Use Case, la specifica di una operazione ha la seguente forma:

alfa (X: T1, ... , Xn: Tn): T
pre: *condizione*
post: *condizione*

- alfa (X: T1, ... , Xn: Tn): T è la **segnatura** dell'operazione (T può mancare),
- pre rappresenta la **precondizione** dell'operazione, cioè l'insieme delle condizioni (a parte quelle già stabilite dalla segnatura) che devono valere **prima** di ogni esecuzione della operazione
- post rappresenta le **postcondizioni** della operazione, cioè l'insieme delle condizioni che devono valere **alla fine** di ogni esecuzione della operazione

Esempio di specifica di use case

InizioSpecificaUseCase Controlli

MediaVoti(s: Studente): real

pre : s.NumEsami > 0

post : result = s.MediaVoti() // invoca operazione di Studente

NumeroStudenti(c: CorsoDiLaurea): int

pre : nessuna

post : result = c.NumeroStud() //invoca operazione di CorsoDiLaurea

MediaEsami(c: CorsoDiLaurea): real

pre : c ha almeno uno studente iscritto

post : result è la media del numero di esami sostenuti dagli studenti iscritti al corso di laurea c

FineSpecifica

Specifica mediante una notazione formale

InizioSpecificaUseCase Controlli

MediaVoti(s: Studente): real

pre : $(\exists c \mid c \in \text{CorsoDiLaurea} \wedge \langle s, c \rangle \in \text{iscritto}) \wedge$
s.NumEsami > 0

post : result = s.MediaVoti()

NumeroStudenti(c: CorsoDiLaurea): int

pre : true

post : result = c.NumeroStud()

...

FineSpecifica