



Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica

Laboratorio di
Ingegneria del Software
a.a. 2013-2014

LEZIONE 4 - USE CASE DIAGRAM

Catia Trubiani

Dipartimento di Ingegneria e Scienze dell'Informazione e
Matematica (DISIM) - Università degli Studi dell'Aquila

catia.trubiani@univaq.it

Use Case Diagram

Gli use case diagrams servono per modellare le funzionalità e i modi d'uso del sistema

- come le funzionalità sono percepite dall'utente
- quali sono gli scenari tipici

Un caso d'uso modella l'intero processo di esecuzione di una funzionalità del sistema:

- dal momento in cui viene espressa l'intenzione di utilizzo da parte dell'utente del sistema
- al momento in cui il sistema processa la richiesta e torna un valore (o un ack) all'utente

Obiettivi degli use cases

1- Per decidere e descrivere i **requisiti funzionali** del sistema, risultante in un accordo tra le parti interessate e gli sviluppatori che lo implementano

2- Per dare una descrizione chiara e coerente di ciò che il sistema dovrebbe fare in modo che il modello può essere utilizzato in tutto il processo di sviluppo per **comunicare** a tutti gli sviluppatori tali requisiti

3- Per fornire una base per l'esecuzione di **test di sistema** che verificano che il sistema funzioni in modo appropriato e convalidarlo. Il sistema finale esegue effettivamente le funzionalità inizialmente richieste?

4- Per fornire la capacità di **rintracciare** i requisiti funzionali nelle effettive classi ed operazioni del sistema. Per semplificare le modifiche ed estensioni al sistema modificando il caso d'uso e quindi l'analisi dei casi d'uso interessati nella progettazione e implementazione del sistema.

3

Use case

L'approccio alla modellazione a casi d'uso è black box: viene specificato quali sono le funzionalità ma non sono forniti i dettagli implementativi.

Non è importante:

- come i casi d'uso sono implementati nel sistema
- come il sistema lavora internamente

4

... ma perché e/o per chi?

Quali sono le funzionalità del sistema? Possono essere specificate da più utenti:

- committenti (stakeholders)
- project managers
- progettisti

Quali sono i requisiti del sistema? Bisogna definire tecniche di modellazione adeguate:

- progettisti
- sviluppatori

Dare indicazioni per il test di sistema

- testers
- project managers

5

Use case diagram

Il **sistema** rappresenta l'oggetto della modellazione

Cosa serve per definire un use case diagram:

- **attori**: ruolo o utente della funzionalità (spesso è una persona fisica)
- **caso d'uso**: descrive un'intera unità funzionale
- **relazioni tra le entità**: specificata attraverso un segmento che collega due o più entità
 - associazione, generalizzazione, estensione, inclusione

Ogni relazione ha una sua notazione specifica (sintassi)

Ogni relazione ha una sua specifica astratta (semantica)

6

Use case diagram -attore-

Un attore specifica un ruolo di interazione con il sistema

Un attore può rappresentare un essere umano, un device, un software, etc.

In generale un attore può essere una qualsiasi entità esterna al sistema considerato

Table 16.1 - Graphic nodes included in use case diagrams

Node Type	Notation	Reference
Actor (default)	 Customer	See "Actor (from UseCases)" on page 586.
Actor (optional user-defined icon - example)		

7

Use case diagram -use case- (1/3)

Uno use case è

- (normalmente) incluso in un sistema
- correlato con attori tramite relazioni (di uso, associazioni, generalizzazioni, etc.)
- ogni relazione sottintende ad una specifica semantica
- un'unità funzionale autonoma quindi non è associabile ad altri casi di uso ma solo ad attori

8

Use case diagram -use case- (2/3)

Uno use case :

- è sempre attivato da un attore
 - un attore richiede (direttamente/indirettamente) al sistema di eseguire un caso d'uso
- deve produrre un qualche effetto visibile a uno o più attori
 - in genere "ritorna un valore" all'attore che lo attiva
- deve essere specificato in modo completo
 - uno use case è completamente specificato se a seguito della sua esecuzione, il sistema risulta in uno stato per cui non sono attesi ulteriori input/azioni per completare la funzionalità che esso rappresenta

9

Use case diagram -use case- (3/3)

La parte più difficile è decidere il livello di specificazione delle funzionalità del sistema

Errori comuni:

- *micro* use case che non hanno una valenza significativa dal punto di vista delle funzionalità offerte dal sistema
- *macro* use case che raggruppano funzionalità diverse

10

Relazione di -associazione-

Definisce quale attore (i.e. chi) è coinvolto in quale caso d'uso (i.e. che cosa) del sistema

Specifica il solo coinvolgimento nel caso d'uso, non chi compie/subisce azioni sul sistema

E' graficamente rappresentata con una linea

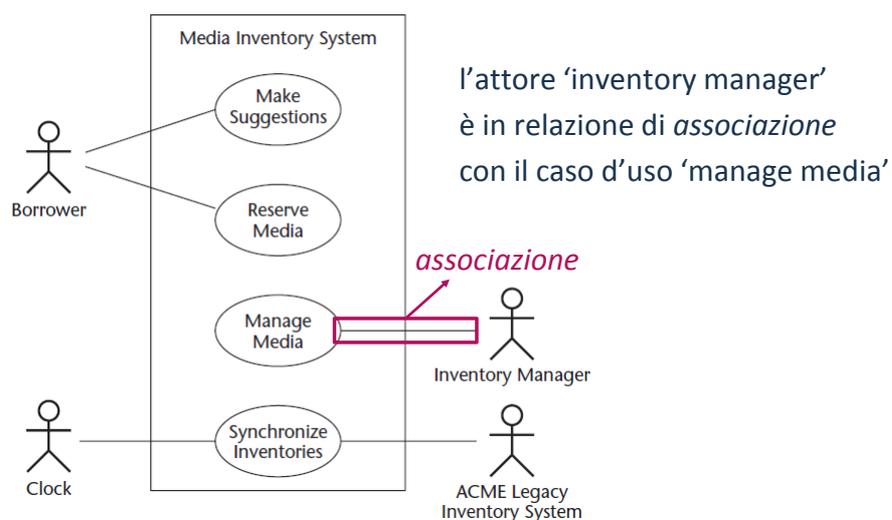
Attori associati a use case con cardinalità maggiore di 1

- un attore è coinvolto in più use cases di quel tipo
- non è specificato se l'interazione è concorrente o mutuamente esclusiva rispetto al tempo

E' la relazione più usata

11

Un esempio di -associazione-



12

Relazione di -generalizzazione-

E' associata ad un concetto molto importante e che viene riusato in molti diagrammi

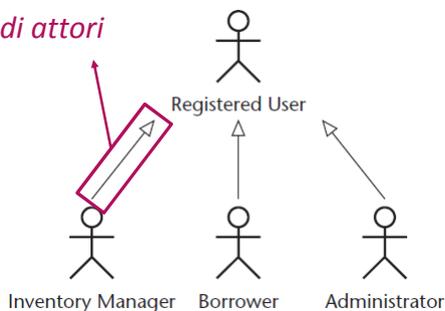
E' una relazioni tra due entità : "padre" e "figlio", vale a dire che specifica come un "figlio" estende (specializza) le caratteristiche del "padre"

Le caratteristiche ed i vincoli specificate nel padre sono implicitamente applicati al figlio

13

Un esempio di -generalizzazione- (1/2)

*generalizzazione
di attori*



Quando diversi attori, come parte dei loro ruoli, svolgono anche un ruolo più generale, allora si usa la relazione di generalizzazione.

Il comportamento del ruolo generale è descritto in una superclasse attore.

Gli attori specializzati ereditano il comportamento della superclasse ed estendono tale comportamento in qualche modo.

14

Relazione di -generalizzazione-

Vincolo di relazione tra due use cases

- generic use case: contenente il comportamento comune
- specific use case: aggiunge e/o modifica il comportamento nel caso generale

Specifica come un “figlio” specializza le caratteristiche del “padre”

Le caratteristiche ed i vincoli specificate nel padre sono implicitamente applicati al figlio

L’esecuzione di use case “fratelli” è esclusiva

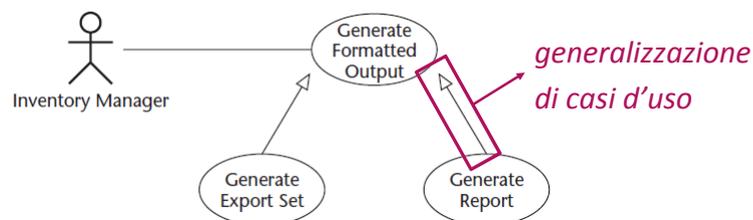
Table 7.3 - Graphic paths included in structure diagrams

<i>PATH TYPE</i>	<i>NOTATION</i>	<i>REFERENCE</i>
Generalization		See “Generalization (from Kernel, PowerTypes)” on page 71.

15

Un esempio di -generalizzazione- (2/2)

I casi d’uso «figli» non si limitano a riutilizzare le stesse procedure del padre (nel qual caso si avrebbe la relazione di -inclusione-), ma modificano il comportamento in qualche modo



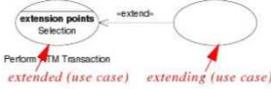
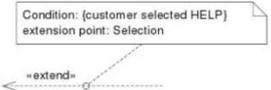
16

Relazione di -estensione-

L'estensione può avvenire in uno o più punti:

- specifica come il comportamento dei casi d'uso che estendono è inserito nel comportamento base
- comportamenti vincolati a particolari condizionali o casi d'uso del sistema

Table 16.1 - Graphic nodes included in use case diagrams

Node Type	Notation	Reference
Extend	 <p>extension points Selection Perform ATM Transaction <i>extended (use case)</i> <i>extending (use case)</i></p>	See "Extend (from UseCases)" on page 589.
Extend (with Condition)	 <p>Condition: [customer selected HELP] extension point: Selection =extend=</p>	

17

Relazione di -estensione-

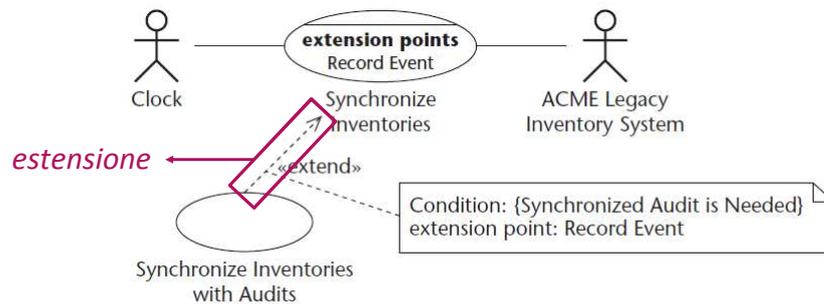
L'estensione è utile nei seguenti scenari:

- il sistema che si sta modellando potrà essere istanziato con molteplici configurazioni opzionali
- l'istanza di un caso d'uso del sistema si compone di sotto-funzionalità che possono essere non completamente specificate
- il caso d'uso base è definito indipendentemente dai casi d'uso che lo estendono
- lo use case base modella una interazione "completa" tra sistema ed attori, il caso d'uso estendente può non avere senso se privato del caso d'uso base
- lo use case estendente può modificare le proprietà dello use case base, mentre lo use case base no

18

Un esempio di -estensione-

Il caso d'uso base descrive il comportamento essenziale e non è a conoscenza di alcun caso d'uso specifico che lo estende, tuttavia c'è la possibilità di essere esteso.



19

Relazione di -inclusione-

Lo scopo primario della relazione «include» è il riuso di parti comuni

Il comportamento dello use case dipende dal comportamento (esterno) del use case incluso

Lo use case incluso è sempre richiesto

Concettualmente è simile alla chiamata di funzioni

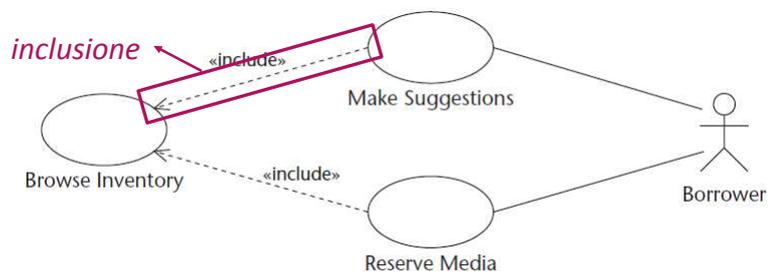
Table 16.1 - Graphic nodes included in use case diagrams

Node Type	Notation	Reference
Include		See "Include (from UseCases)" on page 592.

20

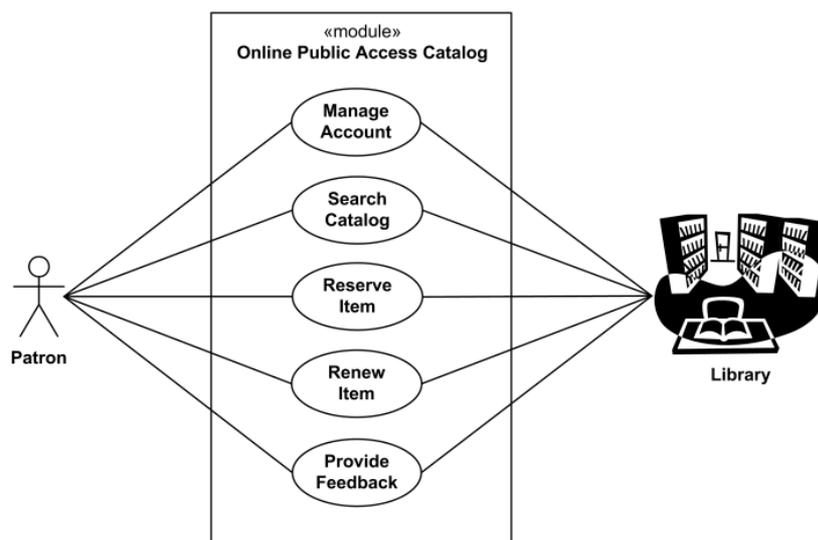
Un esempio di -inclusione-

L'idea della relazione di -inclusione- è quella di modellare il comportamento comune, come ad esempio informazioni di login, che molti casi d'uso dipendono o comprendono altri casi d'uso. Questa relazione incoraggia il **riutilizzo** e il progettista dovrebbe tenere questo in mente quando utilizza questa relazione.



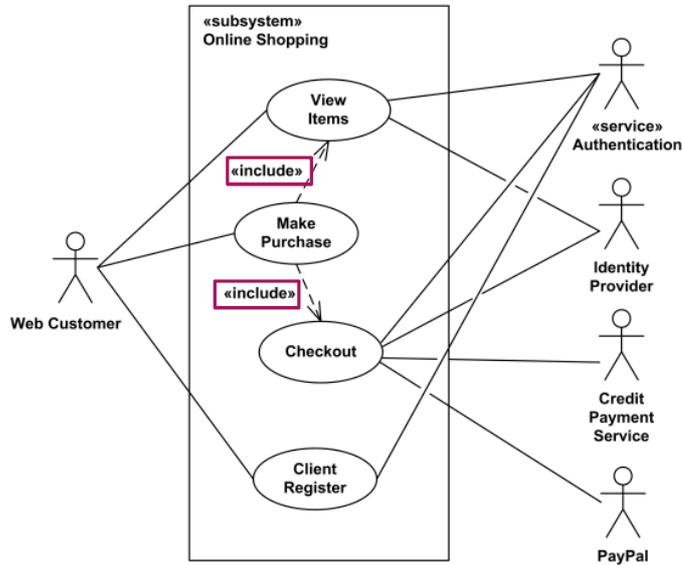
21

Un esempio di Use case diagram



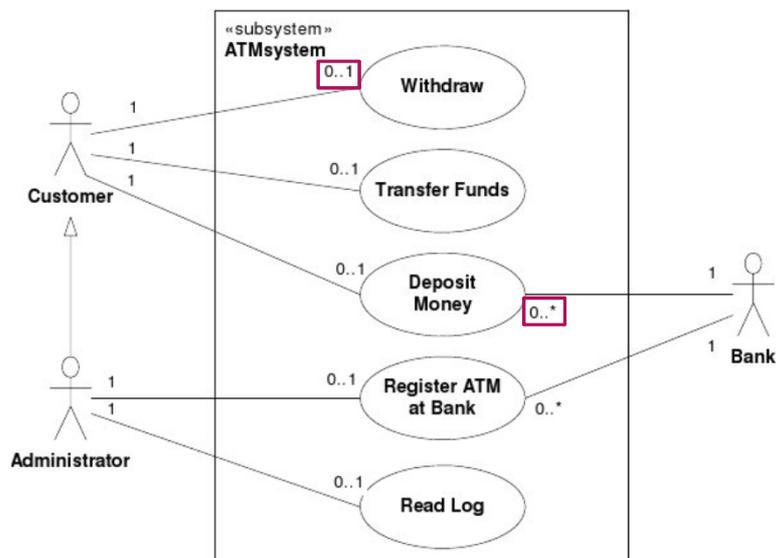
22

Un esempio di Use case diagram



23

Un esempio di Use case diagram



24

Un esempio di Use case diagram

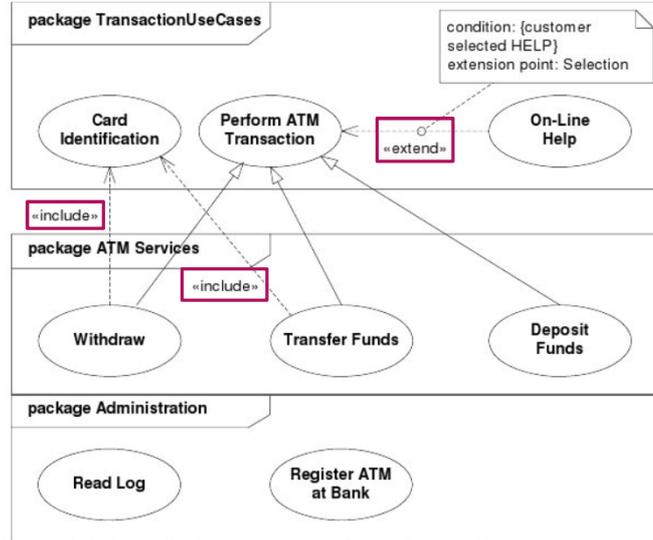
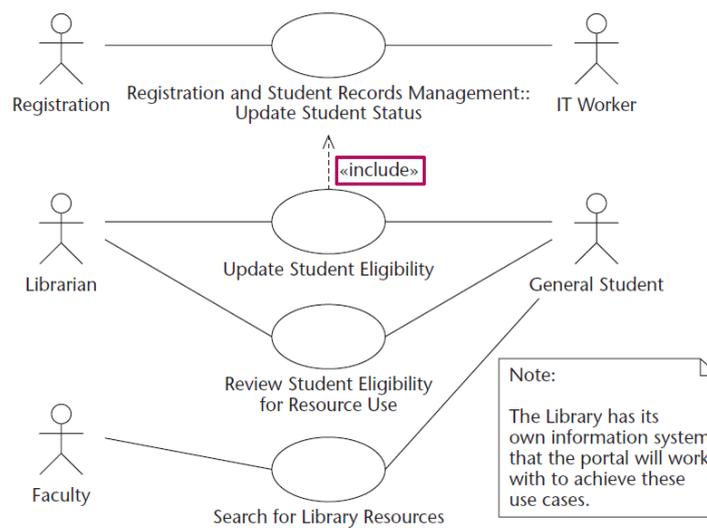


Figure 16.7 - Example of use cases owned by various packages

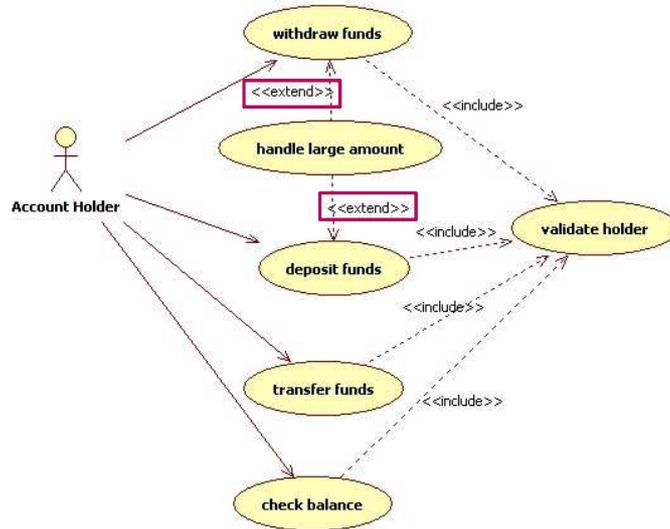
25

Un esempio di Use case diagram



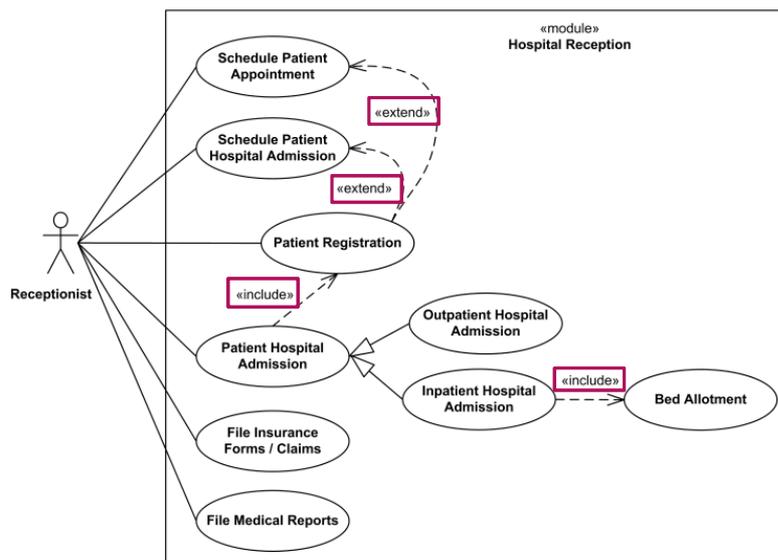
26

Un esempio di Use case diagram



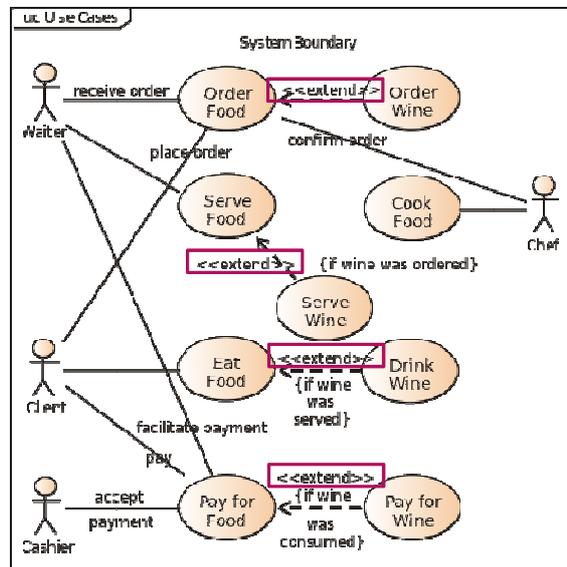
27

Un esempio di Use case diagram



28

Un esempio di Use case diagram



29

Use case ... proviamoci!

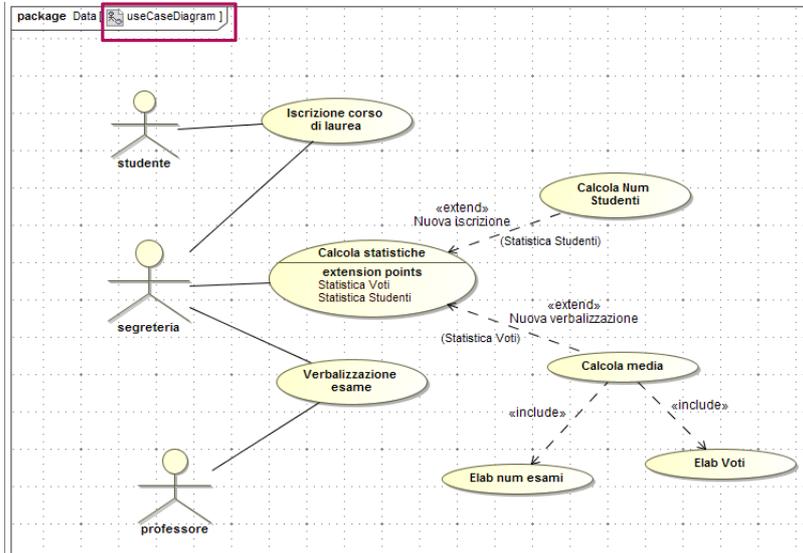
Si vogliono modellare gli studenti (con nome, cognome, numero di matricola, età), il corso di laurea in cui sono iscritti, ed i corsi di cui hanno sostenuto l'esame, con il professore che ha verbalizzato l'esame, ed il voto conseguito. Di ogni corso di laurea interessa il codice e il nome. Di ogni corso 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. Dopo che uno studente ha sostenuto un esame, il professore comunica l'avvenuta verbalizzazione con i dati relativi (studente, corso, voto).

La segreteria vuole periodicamente fare delle statistiche. In particolare, ogni volta che uno studente verbalizza un nuovo esame, vuole calcolare la media dei voti e del numero di esami sostenuti dallo studente. Inoltre, non appena nel sistema una nuova iscrizione viene effettuata, la segreteria vuole aggiornare il numero di studenti di un corso di laurea.

30

Una possibile soluzione



31

Questions?



catia.trubiani@univaq.it

32