

Ingegneria del Software T

Analisi orientata agli oggetti

Analisi

☀ Obiettivo

- ☀ Specificare (cioè definire) le proprietà che il sistema dovrà avere senza descrivere una loro possibile realizzazione

☀ Risultato: una serie di documenti

- ☀ contenenti la **descrizione dettagliata dei requisiti**
- ☀ base di partenza della progettazione

☀ Per determinare in dettaglio i requisiti del sistema, è necessario

- ☀ **interagire** il più possibile con l'utente
- ☀ **conoscere** il più possibile l'area applicativa

Analisi

- ★ **Raccolta dei requisiti**

Obiettivo: raccogliere tutte le informazioni su cosa il sistema deve fare secondo le intenzioni del cliente

- ★ **Analisi dei requisiti (funzionali)**

Obiettivo: definire il comportamento del sistema

- ★ **Analisi del dominio**

Obiettivo: definire la porzione del mondo reale, rilevante per il sistema

- ★ **Analisi e gestione dei rischi**

Obiettivo: identificare e gestire i possibili rischi che possono fare fallire o intralciare la realizzazione del sistema

Analisi e gestione dei rischi

- ✦ Analisi sistematica e completa di tutti i possibili rischi che possono fare **fallire o intralciare la realizzazione del sistema** in una qualsiasi fase del processo di sviluppo
- ✦ Ogni rischio presenta due caratteristiche:
 - ✦ **Probabilità che avvenga**
non esistono rischi con una probabilità del 100% (sarebbero vincoli al progetto)
 - ✦ **Costo**
se il rischio si realizza, ne seguono effetti indesiderati e/o perdite

Analisi e gestione dei rischi

- ✦ **Rischi relativi ai requisiti**

i requisiti sono perfettamente noti?

Il rischio maggiore è quello di costruire un sistema che non soddisfa le esigenze del cliente

- ✦ **Rischi relativi alle risorse umane**

è possibile contare sulle persone e sull'esperienza necessarie per lo sviluppo del progetto?

Analisi e gestione dei rischi

- ✦ **Rischi tecnologici**

si sta scegliendo la tecnologia corretta?
si è in grado di aggregare correttamente i vari componenti del progetto (ad es., GUI, DB, ...)?
quali saranno i possibili futuri sviluppi della tecnologia?

- ✦ **Rischi politici**

ci sono delle forze politiche (anche in senso lato) in grado di intralciare lo sviluppo del progetto?

Analisi e gestione dei rischi

- ✦ **Strategia reattiva** o “*alla Indiana Jones*”

- ✦ “Niente paura, troverò una soluzione”

- ✦ **Strategia preventiva**

- ✦ Si mette in moto molto prima che inizi il lavoro tecnico
- ✦ Si individuano i rischi potenziali, se ne valutano le probabilità e gli effetti e si stabilisce un ordine di importanza
- ✦ Si predispongono un piano che permetta di reagire in modo controllato ed efficace
 - Più grande è un rischio
 - Maggiore sarà l'attenzione che bisognerà dedicargli

Analisi

Raccolta dei requisiti

- ☀ **Obiettivo**

raccogliere tutte le informazioni su
cosa il sistema deve fare secondo le intenzioni del cliente

- ☀ **Non prevede passi formali,**

né ha una notazione specifica, perché
dipende moltissimo dal particolare tipo di problema

- ☀ **Risultato**

- ☀ un **documento** (testuale)
 - ☀ scritto dall'analista
 - ☀ discusso e approvato dal cliente
- ☀ un **dizionario** o **glossario** contenente la definizione di tutti i termini e i concetti utilizzati

Analisi

Raccolta dei requisiti

- ✦ **Tipologie di persone coinvolte**
 - ✦ Analista
 - ✦ Utente
 - ✦ Esperto del dominio (non sempre indispensabile)
- ✦ **Metodi utilizzati**
 - ✦ Interviste, questionari
 - ✦ Studio di documenti che esprimono i requisiti in forma testuale
 - ✦ Osservazione passiva o attiva del processo da modellare
 - ✦ Studio di sistemi software esistenti
 - ✦ Prototipi

Analisi

Raccolta dei requisiti

- ✦ La **gestione delle interviste** è molto complessa
 - ▶ i clienti potrebbero
 - ✦ Avere solo una vaga idea dei requisiti
 - ✦ Non essere in grado di esprimere i requisiti in termini comprensibili
 - ✦ Chiedere requisiti non realizzabili o troppo costosi
 - ✦ Fornire requisiti in conflitto
 - ✦ Essere poco disponibili a collaborare

Analisi

Validazione dei requisiti

- ✦ Ogni requisito deve essere **validato** e **negoziato** con i clienti prima di essere riportato nel documento dei requisiti
- ✦ Attività svolta in parallelo alla raccolta
- ✦ **Validità** – il nuovo requisito è inerente il problema da risolvere?
- ✦ **Consistenza** – il nuovo requisito è in sovrapposizione e/o in conflitto con altri requisiti?
- ✦ **Realizzabilità** – il nuovo requisito è realizzabile con le risorse disponibili (hardware, finanziamenti, ...)?
- ✦ **Completezza** – esiste la possibilità che ci siano requisiti rimasti ignorati?

Analisi

Cambiamento dei requisiti

- ✦ È normale che i requisiti subiscano modificazioni ed evolvano nel tempo
 - ✦ Requisiti esistenti possono essere rimossi o modificati
 - ✦ Nuovi requisiti possono essere aggiunti in una fase qualsiasi del ciclo di sviluppo
- ✦ Tali cambiamenti
 - ✦ Sono la norma, non l'eccezione
 - ✦ Possono diventare un grosso problema se non opportunamente gestiti

Analisi

Cambiamento dei requisiti

- ✦ Più lo sviluppo è avanzato, più il cambiamento è costoso
 - ✦ Modificare un requisito appena definito è facile
 - ✦ Modificare lo stesso requisito dopo che è stato implementato nel software potrebbe essere molto costoso
- ✦ Ogni cambiamento deve essere accuratamente analizzato per valutare
 - ✦ La fattibilità tecnica
 - ✦ L'impatto sul resto del sistema
 - ✦ Il costo

Analisi

Cambiamento dei requisiti

- ✦ **Consiglio** – sviluppare sistemi che
 - ✦ Siano il più possibile **resistenti ai cambiamenti dei requisiti**
 - ✦ Inizialmente, eseguano esclusivamente e nel modo migliore i soli compiti richiesti
 - ✦ In seguito, siano in grado di sostenere l'aggiunta di nuove funzionalità senza causare “*disastri*” strutturali e/o comportamentali

Analisi

Analisi dei requisiti

- ✦ **Obiettivo**

definire il **comportamento del sistema** da realizzare

- ✦ **Risultato**

un **modello comportamentale** (o **modello dinamico**)
che descrive in modo chiaro e conciso le **funzionalità** del
sistema

- ✦ che cosa deve fare il sistema per soddisfare il cliente
- ✦ non come il sistema va realizzato

Analisi

Analisi dei requisiti

✦ Strategia

- ✦ **Scomposizione funzionale** (mediante *analisi top-down*) ► identificare le singole funzionalità previste dal sistema
- ✦ **Astrazione procedurale** ► considerare ogni operazione come una singola entità, nonostante tale operazione sia effettivamente realizzata da un insieme di operazioni di più basso livello

✦ Attenzione

La scomposizione in funzioni è **molto volatile** a causa del continuo **cambiamento dei requisiti funzionali**

Analisi

Analisi del dominio

- ☀ **Obiettivo**

definire la **porzione del mondo reale**, rilevante per il sistema da realizzare

- ☀ **Principio fondamentale: Astrazione**

Permette di gestire la complessità intrinseca del mondo reale

- ignorare gli aspetti che non sono importanti per lo scopo attuale
- concentrarsi maggiormente su quelli che lo sono

- ☀ **Risultato**

un **modello dei dati** (o **modello statico**)

che descrive le **entità del mondo reale** rilevanti per il sistema

- su cui si devono mantenere informazioni
- con cui si deve interagire

- ☀ Produce una specifica più resistente ai cambiamenti
gli oggetti sono molto più stabili delle funzioni

Analisi

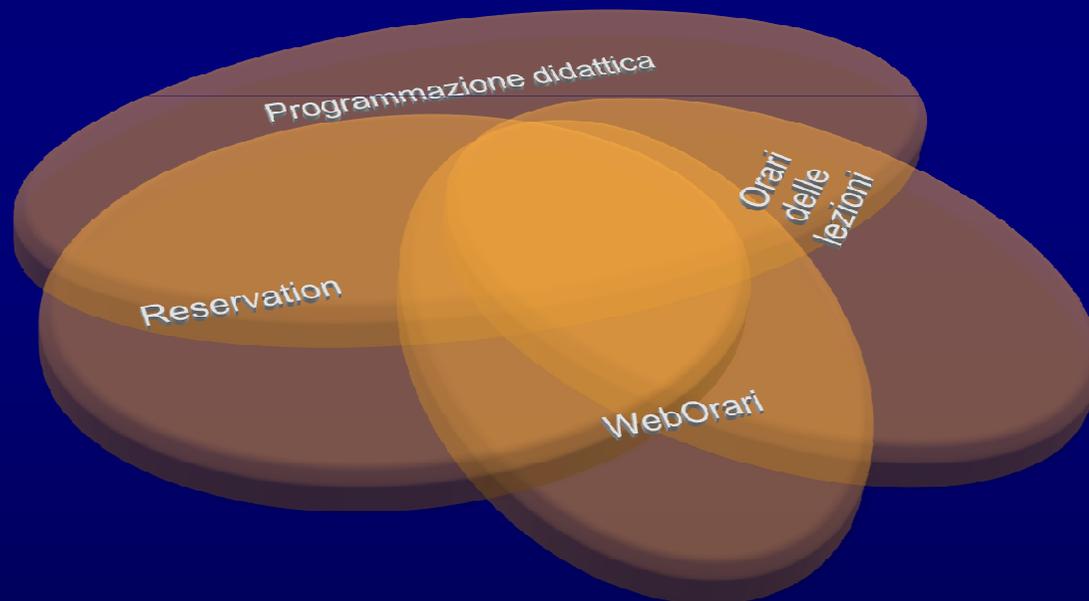
Analisi a livello di area applicativa

- ✦ L'analisi può essere effettuata su un **gruppo di sistemi** afferenti alla **stessa area applicativa**
- ✦ Esempi di aree applicative:
 - ✦ il controllo del traffico aereo
 - ✦ la gestione aziendale
 - ✦ le operazioni bancarie
 - ✦ ...
- ✦ In tal caso, è possibile
 - ✦ **identificare entità e comportamenti comuni** a tutti i sistemi
 - ✦ realizzare **schemi** di progettazione e **componenti** software **riutilizzabili** nei diversi sistemi

Analisi

Analisi a livello di area applicativa

- ✦ Aiuta ad effettuare le analisi dei nuovi sistemi, ed è da queste continuamente migliorata



Esempio: Villaggio Turistico

Raccolta dei Requisiti

- ✱ In un villaggio turistico, gli ospiti fanno spesa nei diversi negozi e pagano i diversi servizi sempre e solo servendosi di una carta (simile a un bancomat) denominata Guest Card
- ✱ La valuta di riferimento è sempre l'euro
- ✱ Al termine della vacanza, ad ogni ospite viene consegnato un estratto conto con la lista delle spese effettuate, nella valuta scelta dal cliente
- ✱ Per ogni spesa, l'elenco deve riportare la data e l'ora, il punto vendita, il tipo di acquisto e l'importo addebitato
- ✱ Al termine di ogni settimana, ad ogni negozio deve essere consegnato l'elenco degli acquisti effettuati presso i vari punti vendita associati

Esempio: Villaggio Turistico

Analisi

- ✦ In un **villaggio turistico**, gli **ospiti** fanno **spesa** nei diversi **negozi** e pagano i diversi **servizi** sempre e solo servendosi di una **carta** (simile a un **bancomat**) denominata **Guest Card**
- ✦ Villaggio Turistico
- ✦ Ospite
- ✦ Spesa
- ✦ Negozio
- ✦ Servizio
- ✦ Carta ► Guest Card

Esempio: Villaggio Turistico

Analisi

- ✦ In un villaggio turistico, gli ospiti **fanno spesa** nei diversi negozi e **pagano** i diversi servizi sempre e solo **servendosi** di una carta (simile a un bancomat) denominata Guest Card
- ✦ Ospite
 - Può acquistare un servizio in un negozio
 - Deve pagare il servizio con la Guest Card
- ✦ Negozio
 - Eroga servizi
 - Incassa il pagamento del servizio mediante Guest Card
- ✦ Servizio
 - Ha un costo
- ✦ Guest Card
 - Unico mezzo per effettuare i pagamenti

Esempio: Villaggio Turistico

Analisi

- ✦ La **valuta di riferimento** è sempre l'**euro**
- ✦ Ospite
 - Può acquistare un servizio in un negozio
 - Deve pagare il servizio con la Guest Card **in euro**
- ✦ Negozio
 - Eroga servizi **il cui costo è in euro**
 - Incassa il pagamento del servizio mediante Guest Card **in euro**
- ✦ Servizio
 - Ha un costo **in euro**
- ✦ Guest Card
 - Permette di effettuare i pagamenti **in euro**
- ✦ Valuta di riferimento
 - Unica in tutto il Villaggio Turistico
 - In euro

Esempio: Villaggio Turistico

Analisi

- ✱ Al termine della vacanza, ad ogni ospite viene consegnato un estratto conto con la lista delle spese effettuate, nella valuta scelta dal cliente
- ✱ Termine della vacanza – evento temporale
- ✱ Estratto conto \equiv lista delle spese effettuate
 - ✱ *report* di stampa
- ✱ Spesa effettuata \equiv Servizio acquistato dall'ospite
- ✱ Cliente \equiv Ospite
- ✱ Valuta scelta dall'ospite
 - ✱ Può essere differente dalla valuta di riferimento

Esempio: Villaggio Turistico

Analisi

- ✦ Al termine della vacanza, ad ogni ospite **viene consegnato** un estratto conto con la lista delle spese effettuate, nella valuta **scelta** dal cliente
- ✦ Ospite
 - ✦ **Deve scegliere la valuta x pagamento finale**
- ✦ Termine della vacanza – evento
 - ✦ **Generazione dell'estratto conto acquisti**
 - ✦ **Consegna all'ospite dell'estratto conto acquisti**
 - ✦ **Pagamento finale nella valuta scelta dall'ospite**
- ✦ **NOTA:** Sarà necessario effettuare conversioni tra valute diverse

Esempio: Villaggio Turistico

Analisi

- ✦ Per ogni **spesa**, l'**elenco** deve riportare la **data** e l'**ora**, il **punto vendita**, il **tipo di acquisto** e l'**importo addebitato**
- ✦ Spesa \equiv Acquisto o Movimento
 - ✦ **Data e ora del movimento**
 - ✦ **Punto di vendita** (NON coincide con Negozio!)
 - ✦ **Tipo di acquisto**
 - ✦ **Importo** in euro
- ✦ Punto Vendita
- ✦ Catena Punti Vendita (ex Negozio)
- ✦ Tipo di Acquisto \equiv Servizio

Esempio: Villaggio Turistico

Analisi

- ✦ Al termine di ogni settimana, ad ogni negozio deve essere consegnato l'elenco degli acquisti effettuati presso i vari punti vendita associati
- ✦ Termine di ogni settimana – evento temporale
 - ✦ Generazione dell'estratto conto vendite x Punto Vendita
 - ✦ Consegna alla Catena Punti Vendita

Analisi

Casi d'uso e scenari

- ✦ I requisiti funzionali descrivono il comportamento del sistema
- ✦ I **casi d'uso** e i relativi **scenari** permettono
 - ✦ Di formalizzare i requisiti funzionali
 - ✦ Di comprendere meglio il funzionamento del sistema (e quindi di metterne in evidenza eventuali carenze)
 - ✦ Di comunicare meglio con il cliente

Analisi

Casi d'uso e scenari

✦ **Caso d'uso**

- ✦ Descrizione di una richiesta fatta al sistema da una qualsiasi entità esterna al sistema stesso (attore)
- ✦ Insieme di scenari legati da un obiettivo comune

✦ **Scenario** – sequenza di passi che descrive

- ✦ sia l'interazione tra l'attore e il sistema
- ✦ sia le elaborazioni necessarie per soddisfare la richiesta dell'attore

Analisi

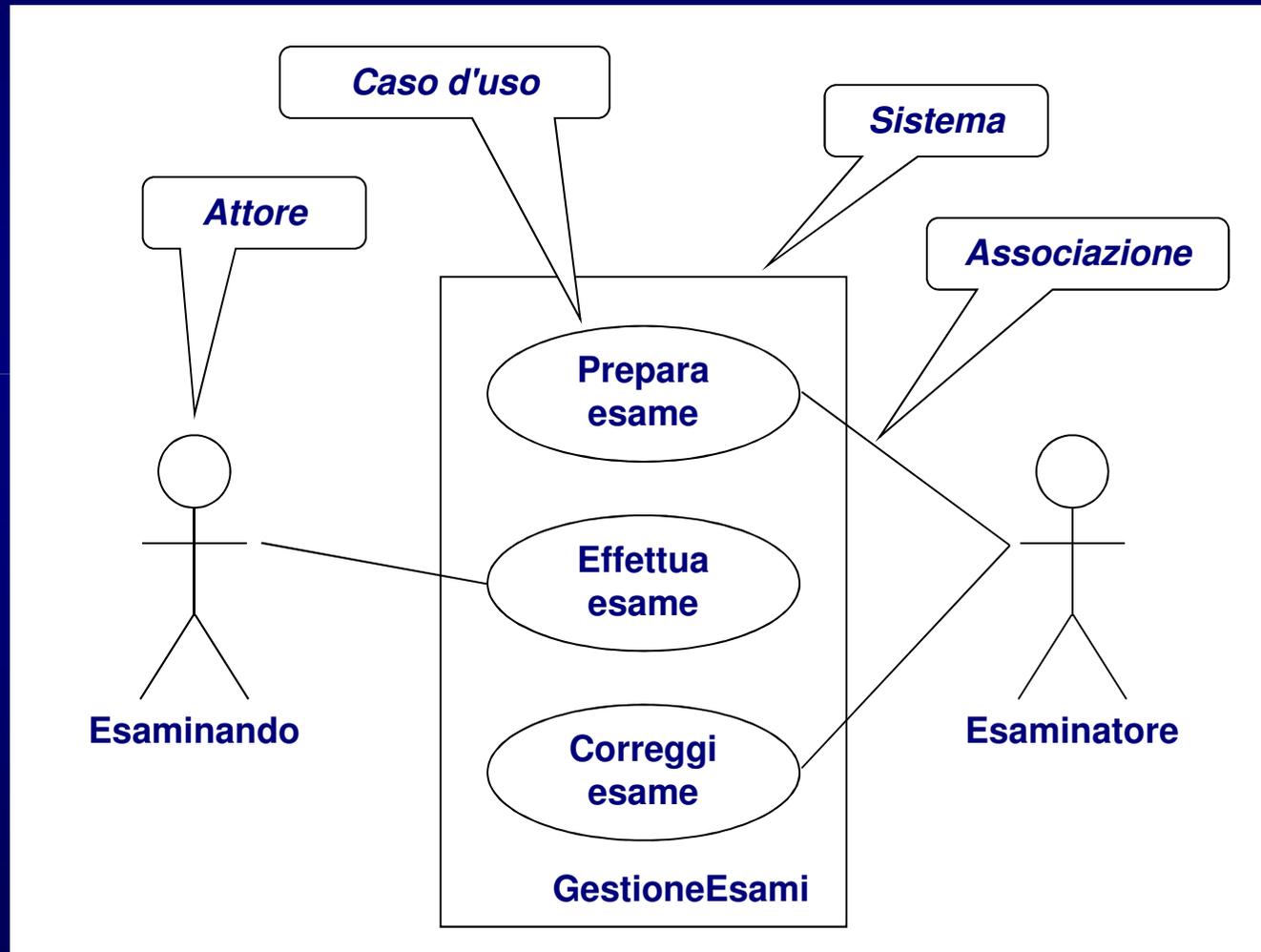
Casi d'uso e scenari

✦ **Passi da intraprendere**

- ✦ Individuare il confine del sistema
- ✦ Individuare gli attori
- ✦ Individuare i casi d'uso
 - Specificare il caso d'uso
 - Specificare gli scenari associati al caso d'uso

- ✦ L'insieme di tutti i casi d'uso costituisce l'immagine del sistema verso l'esterno

Analisi Casi d'uso e scenari



Analisi

Casi d'uso e scenari



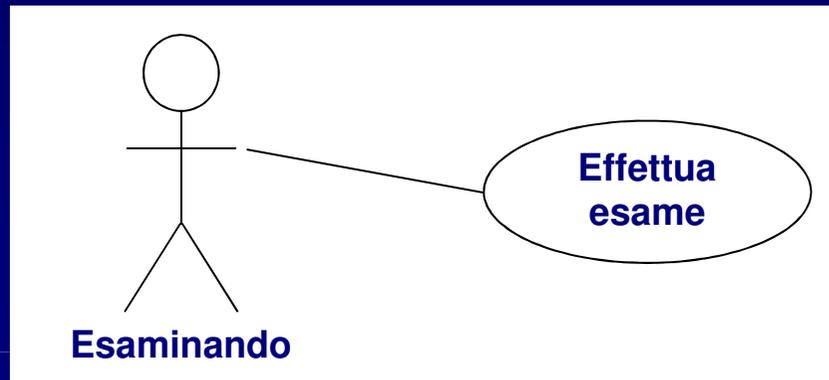
★ **Attore**

ruolo interpretato da un utente (persona o sistema esterno) nei confronti del sistema

- ★ Tutti gli esaminandi interpretano lo stesso ruolo
- ★ Tutti gli esaminatori interpretano lo stesso ruolo

Analisi

Casi d'uso e scenari



Scenario principale
del caso d'uso
“Effettua esame”

1. L'esaminando entra nel sistema (*login*)
2. L'esaminando inizia l'esame
3. L'esaminando naviga tra le domande e risponde
4. L'esaminando termina l'esame
5. L'esaminando esce dal sistema (*logout*)

Analisi

Casi d'uso e scenari

✦ Un caso d'uso

- ✦ viene sempre **avviato** direttamente o indirettamente dall'intervento di un **attore** che si pone un dato **obiettivo**
 - l'esaminando vuole fare l'esame
- ✦ si **conclude con successo** quando l'obiettivo viene raggiunto
 - l'esaminando ha fatto l'esame
- ✦ si **conclude con fallimento** quando l'obiettivo NON viene raggiunto
 - l'esaminando non è riuscito a fare l'esame – ad es. non è riuscito ad effettuare il login (in questo contesto, il fatto che l'attore abbia superato o no l'esame è irrilevante)

Analisi

Casi d'uso e scenari

- ✦ Un caso d'uso viene sempre descritto dal punto di vista di un attore e comprende
 - ✦ **0+ Precondizioni** – condizioni che devono essere tutte verificate prima che il caso d'uso possa essere eseguito – vincoli sullo stato iniziale del sistema
 - ✦ **1+ Sequenze di passi** – cioè **sequenze di interazioni** tra l'attore e il sistema necessarie a raggiungere l'obiettivo richiesto – potrebbero comprendere ramificazioni (**if**) e iterazioni (**for**, **foreach** e **while**)
 - ✦ **0+ Postcondizioni** – condizioni che devono essere tutte vere quando il caso d'uso termina l'esecuzione di norma con successo

Analisi

Casi d'uso e scenari

- ✦ Ogni sequenza di passi deve
 - ✦ essere scritta in una **forma narrativa strutturata**
 - ✦ utilizzare il **vocabolario di dominio**
- ✦ In tal modo, il committente
 - ✦ potrà comprendere facilmente la descrizione dei casi d'uso e di conseguenza
 - ✦ non solo sarà in grado di validare i casi d'uso
 - ✦ ma sarà anche incoraggiato a partecipare attivamente alla loro definizione

Analisi

Casi d'uso e scenari

- ✦ Tipicamente un caso d'uso comprende
 - ✦ uno **scenario principale**
 - ✦ eventuali **scenari alternativi**
 - che rappresentano possibili varianti del flusso
 - che sono fatti “scattare” da opzioni, condizioni d'errore, violazione della sicurezza, ...
- ✦ Ogni **scenario**
 - ✦ potrebbe essere descritto formalmente mediante un **diagramma di sequenza**,
 - ✦ ma viene descritto mediante il **linguaggio naturale**, che è più adeguato nella fase di raccolta dei requisiti

Definizione dei casi d'uso e degli scenari

1. Identificare tutti i **possibili utilizzi** del sistema
2. Definire un **attore** per ogni categoria di utenza e per ogni ruolo che l'utente gioca e che ha rilevanza per il sistema
3. Per ciascun attore identificare tutti gli **obiettivi significativi** che gli utenti si pongono e che il sistema dovrà supportare
4. Per ciascun obiettivo definire un **caso d'uso** e i relativi **scenari**
5. **Ricontrollare e validare** i casi d'uso insieme agli utenti e ai committenti

Analisi

Relazioni tra casi d'uso

✦ **Generalizzazione / Specializzazione**

- ✦ Si utilizza quando un caso d'uso è simile ad un altro, ma fa qualcosa di più
- ✦ È applicabile anche agli attori – un attore può essere la specializzazione di un altro attore

✦ **Inclusione** «include» (o «uses»)

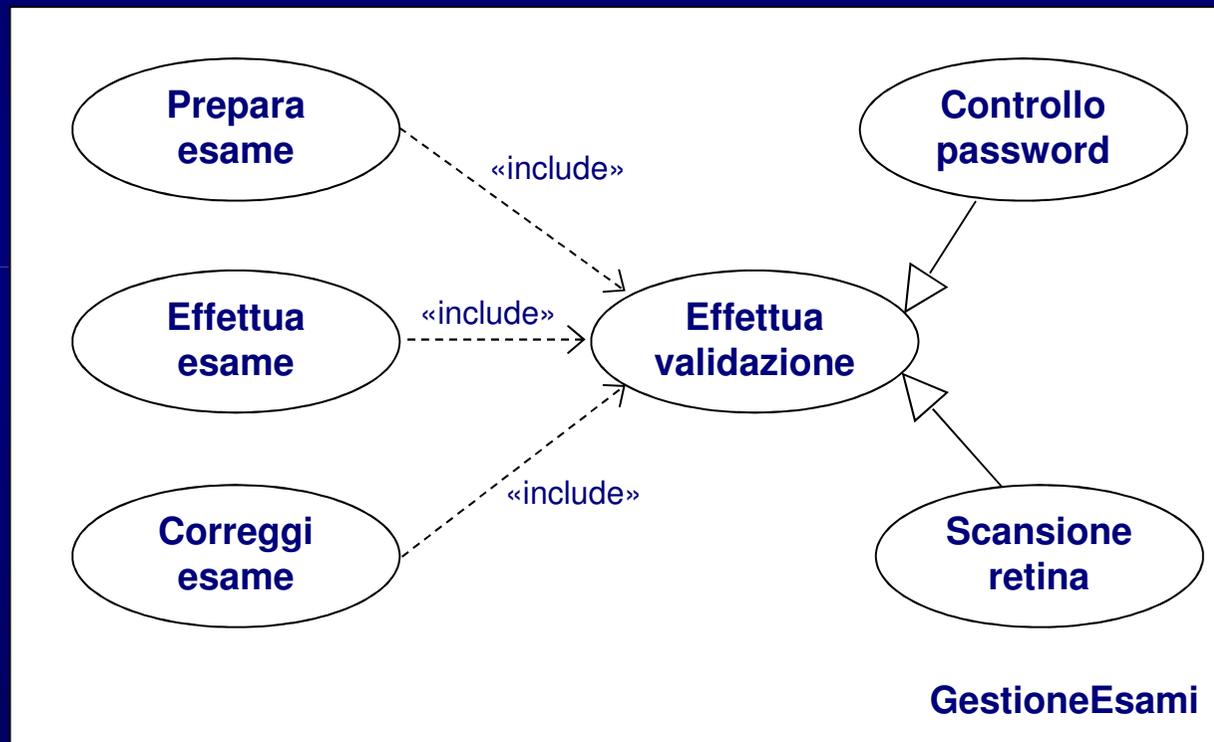
- ✦ Si utilizza quando un caso d'uso “usa” almeno una volta un altro caso d'uso

✦ **Estensione** «extend» (o «extends»)

- ✦ Si utilizza quando è necessario aggiungere un comportamento opzionale a un caso d'uso esistente

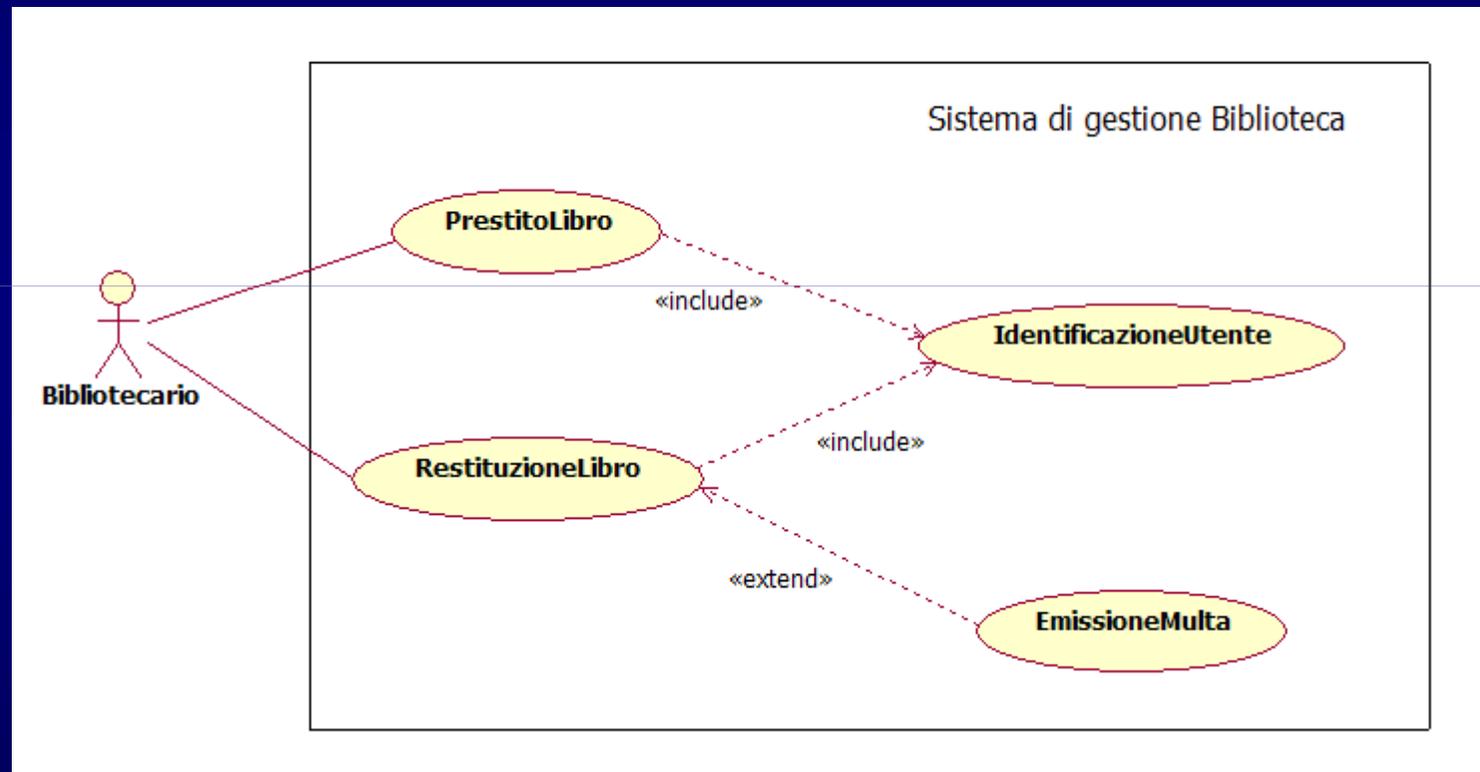
Analisi

Relazioni tra casi d'uso



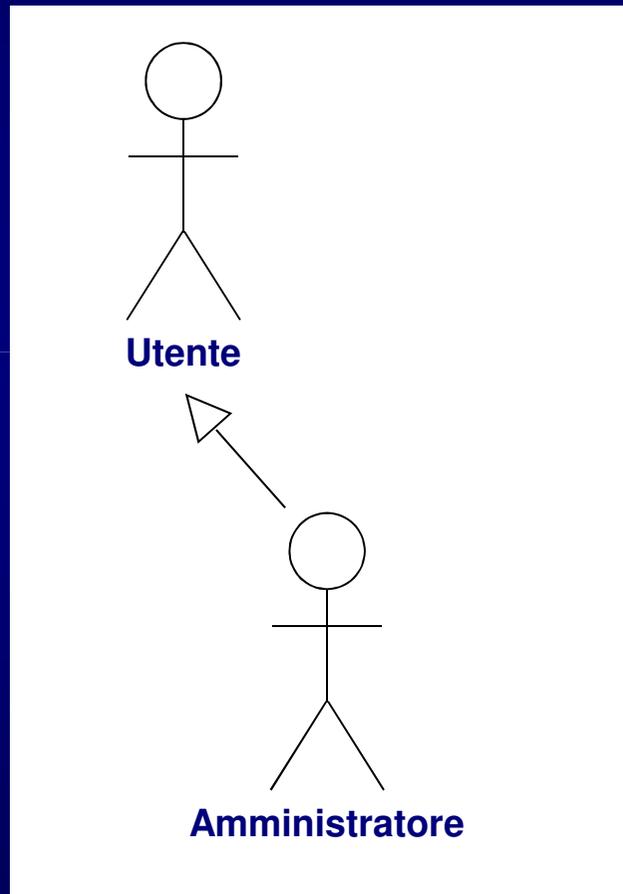
Analisi

Relazioni tra casi d'uso



Analisi

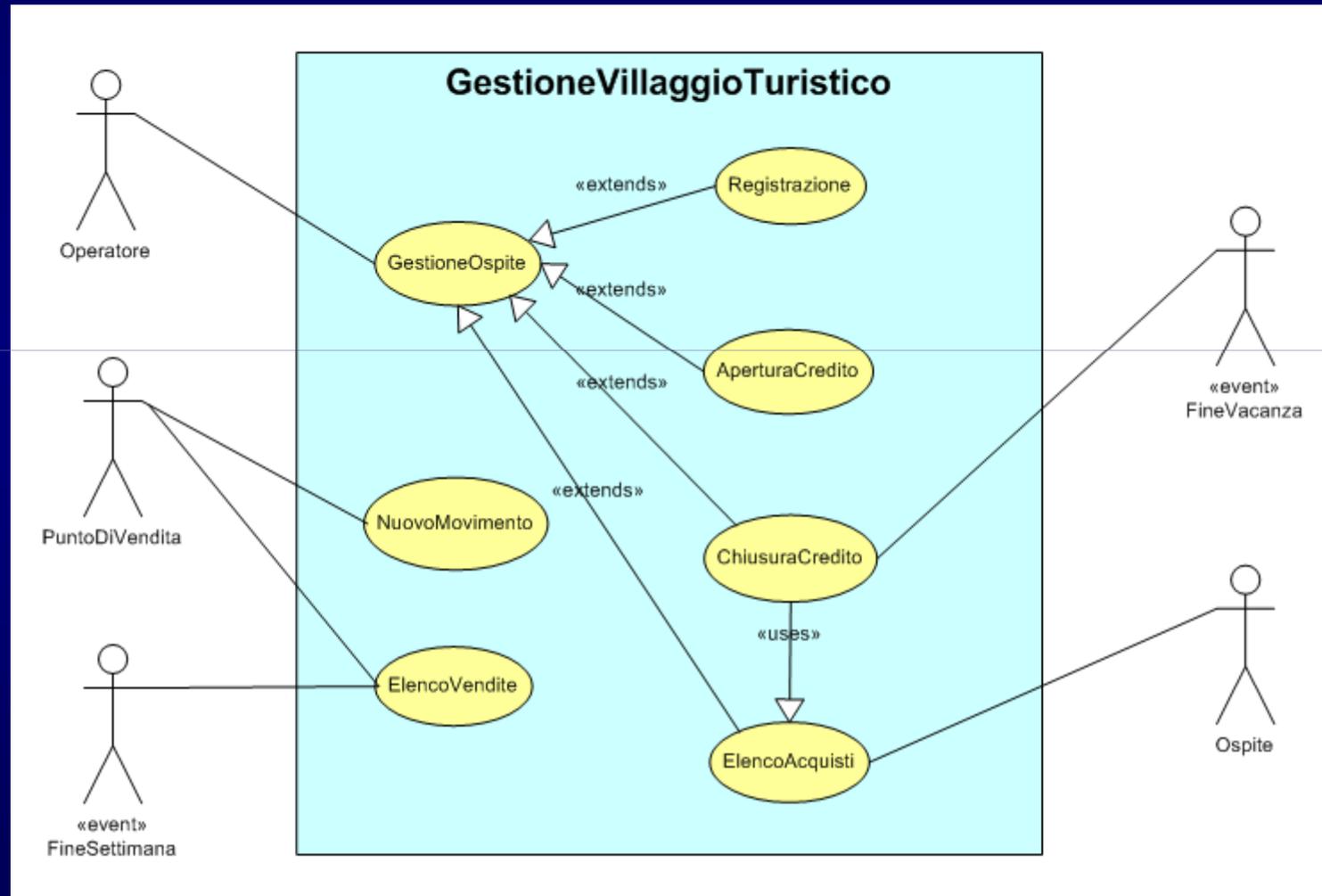
Relazioni tra attori



- ☀ L'attore Amministratore
 - ☀ Eredita tutti i casi d'uso dell'attore Utente
 - ☀ Ha casi d'uso propri

Esempio: Villaggio Turistico

Casi d'uso



Descrizione di un caso d'uso e dei suoi scenari

Titolo	Nome del caso d'uso (CU)
Descrizione	Breve descrizione di CU
Relazioni	Relazioni di CU con altri casi d'uso
Attori	Elenco degli attori che interagiscono con CU
Precondizioni	
Postcondizioni	
Scenario principale	
Scenari alternativi	
Requisiti non funzionali	
Punti aperti	