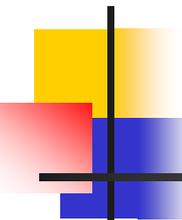


# Microsoft Visio 2002 - UML

---

Sergio Colosio

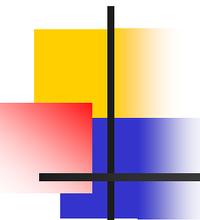


# Microsoft Visio 2002

## UML – Casi d'uso

---

- Uno scenario è una sequenza di passi che descrivono l'interazione tra un utente e il sistema.
- Un caso d'uso è un insieme di scenari legati da un obiettivo comune per l'utente



# Microsoft Visio 2002

## UML – Casi d'uso

---

- La relazione di inclusione si verifica quando un determinato comportamento si ripete in più casi d'uso e non si vuole ripetere la descrizione
- La relazione di generalizzazione si verifica quando un caso d'uso è simile ad un altro, ma fa qualcosa di più.
- La relazione di estensione è simile alla generalizzazione, ma ci sono più regole.

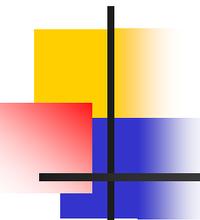
# Microsoft Visio 2002

## UML – Casi d'uso

---

### Esempio 1:

- Attore: cittadino – utente
- Caso d'uso: ricerca di una licenza edilizia
  1. Visualizza la pagina relativa all'interrogazione della banca dati delle licenze edilizie, l'utente digita UID (numero bolletta) e PWD (codice fiscale)
  2. Il sistema visualizza i dati relativi a:
    - numero bolletta
    - data di presentazione
    - progettista
    - descrizione della richiesta
    - località
- Alternativa: numero bolletta non trovata
  - 2a. Il sistema visualizza l'informazione di “bolletta inesistente”
  - 3a. Ritorna a 1
- Alternativa: password non corretta
  - 2a. Il sistema visualizza l'informazione di “password non corretta”
  - 3a. Ritorna a 1



# Microsoft Visio 2002

## UML – Casi d'uso

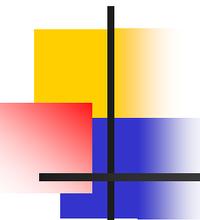
---

### Esempio 2:

- Attore: cittadino – utente

Precondizione: il sistema ha visualizzato i dati relativi alla richiesta

- Caso d'uso: concessione approvata a condizione
  1. <<include>> Ricerca licenze edilizie
  2. Il sistema visualizza i dati relativi a:
    - zona urbanistica
    - tipo intervento
    - posizione pratica
    - parere Commissione Edilizia (approvata a condizione)
    - descrizione integrale del parere della Commissione edilizia



# Microsoft Visio 2002

## UML – Casi d'uso

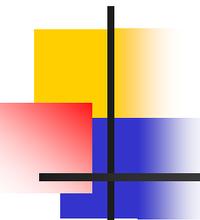
---

### Esempio 3:

- Attore: cittadino – utente

Precondizione: il sistema ha visualizzato i dati relativi alla richiesta

- Caso d'uso: concessione approvata
  1. <<include>> Ricerca licenze edilizie
  2. Il sistema visualizza i dati relativi a:
    - zona urbanistica
    - tipo intervento
    - posizione pratica
    - parere Commissione Edilizia (approvata)
    - descrizione integrale del parere della Commissione edilizia



# Microsoft Visio 2002

## UML – Casi d'uso

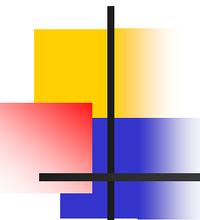
---

Esempio 4:

- Attore: cittadino – utente

Precondizione: il sistema ha visualizzato i dati relativi alla richiesta

- Caso d'uso: concessione non approvata
  1. <<include>> Ricerca licenze edilizie
  2. Il sistema visualizza i dati relativi a:
    - zona urbanistica
    - tipo intervento
    - posizione pratica
    - parere Commissione Edilizia (non approvata)
    - descrizione integrale del parere della Commissione edilizia



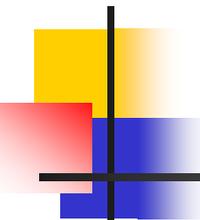
# Microsoft Visio 2002

## UML – Casi d'uso

---

### Esempio 5:

- Attore: navigatore internet
- Caso d'uso: iscrizione a un servizio di profilazione per un portale
  1. Il navigatore accede alla pagina di registrazione
  2. Inserisce i dati richiesti (es. e-mail, password, ecc.)
  3. Il sistema autorizza la registrazione dell'utente
  4. Il sistema invia all'utente un e-mail di conferma
  
- Alternativa: e-mail non valida
  - 2a. Il sistema visualizza l'informazione di e-mail non corretta



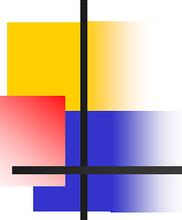
# Microsoft Visio 2002

## UML – Casi d'uso

---

### Esempio 6:

- Attore: navigatore internet
- Caso d'uso: profilazione di un utente
  1. Il navigatore digita UID e PWD
  2. Il sistema riconosce l'utente e invia una conferma di riconoscimento
  3. l'utente naviga nel sito
  4. Il sistema tiene traccia della navigazione
  
- Alternativa: UID e PWD non riconosciute
  - 2a. Il sistema ripropone la Home page di registrazione



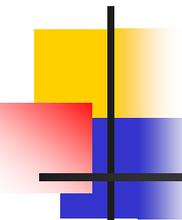
# Microsoft Visio 2002

## UML – Diagramma delle Classi

---

Disegnando un diagramma delle classi si possono adottare tre diversi punti di vista o prospettive.

- Concettuale: da questo punto di vista il diagramma rappresenta i concetti propri del dominio che si sta studiando.
- Specifica: da questo punto di vista il diagramma rappresenta l'interfaccia (software)
- Implementazione: da questo punto di vista si fa esplicitamente riferimento alle classi e si schematizza l'implementazione vera e propria



# Microsoft Visio 2002

## UML – Diagramma delle Classi

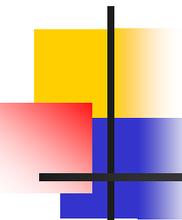
---

Nome del ruolo: etichetta che viene assegnata al capo di un'associazione

Molteplicità: indica quanti oggetti possono prendere parte alla relazione

Navigabilità: si tratta di incrementare il significato di un'associazione attraverso l'utilizzo di frecce.

Si hanno così associazioni unidirezionali se navigabili in un solo verso, bidirezionali se navigabili in entrambe le direzioni.



# Microsoft Visio 2002

## UML – Diagramma delle Classi

---

Per spiegare gli attributi è utile un esempio:

- dal punto di vista concettuale, l'attributo "nome" di un cliente indica che i clienti hanno nomi;
- dal punto di vista della specifica indica che un oggetto cliente può restituire il proprio nome e offre metodi per impostarlo;
- dal punto di vista dell'implementazione un oggetto cliente ha un campo destinato a contenere il nome.

La sintassi UML:

*visibilità nome:tipo = valoreDefault*

- la visibilità può essere + (pubblico), - (privato), \* (protected)

# Microsoft Visio 2002

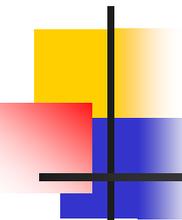
## UML – Diagramma delle Classi

Le operazioni sono processi che una classe sa come effettuare, corrispondono ai metodi della classe.

La sintassi UML:

*visibilità nome (lista parametri): tipi-ritornati {stringa-proprietà}*

- la *visibilità* può essere + (pubblico), - (privato), \* (protected)
- il *nome* è un stringa
- *lista parametri* contiene una serie di parametri separati da virgole, la cui sintassi è simile a quella degli attributi:  
*direzione nome:tipo = valoreDefault*  
l'unica aggiunta è la direzione che indica se il parametro viene usato in input (*in*), output (*out*). O entrambi (*inout*)
- *tipi ritornati* è una lista di tipi di ritorno separati da virgole
- *stringa-proprietà* indica i valori proprietà che si applicano all'operazione data



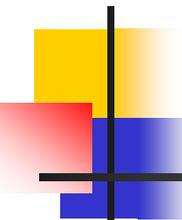
# Microsoft Visio 2002

## UML – Diagramma delle Classi

---

La generalizzazione, in un'ottica di specificità, significa che l'interfaccia del sottotipo deve includere tutti gli elementi presenti nell'interfaccia del tipo padre. Si dice che l'interfaccia del sottotipo è conforme all'interfaccia del supertipo.

Concetto di sostituibilità: dovrebbe essere possibile sostituire un'istanza del sottotipo all'interno di un qualsiasi pezzo di codice che preveda l'esistenza di un'istanza del supertipo e tutto dovrebbe poter continuare a funzionare.



# Microsoft Visio 2002

## UML – Diagramma delle Classi

---

L'UML consente di descrivere i vincoli in qualsiasi modo si desidera, l'unica regola è che questi vengano posti fra parentesi graffe ({}).