

Università degli Studi di Roma “La Sapienza”
Facoltà di Ingegneria – Corso di Laurea in Ingegneria Gestionale
Corso di Progettazione del Software
Proff. Toni Mancini e Monica Scannapieco

Progetto **PI.20070402**, passo **A.3**

versione del 10 aprile 2007

Si vuole progettare e realizzare *eBuy*, un sistema informatico per la gestione di aste on-line e di attività di commercio elettronico. In particolare, il sistema deve permettere, agli utenti registrati, di pubblicare annunci per la vendita di oggetti e di gestire aste al rialzo per la loro aggiudicazione. Il sistema deve anche consentire di mettere in vendita oggetti senza l'effettuazione di alcun'asta (formula “compralo subito”).

Si richiede di proseguire la fase di Analisi, estendendo lo schema concettuale dell'applicazione in modo da modellare i seguenti requisiti aggiuntivi.

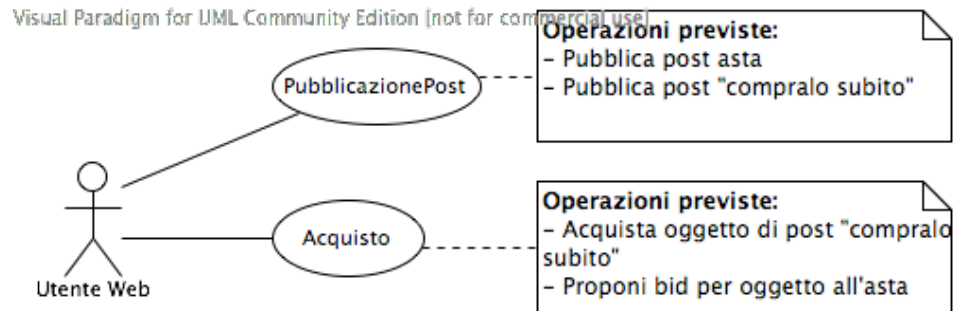
Requisiti

Per favorire l'esclusione dal sistema dei venditori scorretti, si vuole estendere il sistema permettendo, agli utenti che si aggiudicano un'asta e a quelli che acquistano oggetti secondo la modalità “compralo subito”, di esprimere un giudizio sulla qualità della transazione e sul loro grado generale di soddisfazione. Tali giudizi sono denominati *feedback* e consistono in un voto numerico da 0 (non soddisfatto) a 5 (molto soddisfatto), al quale può aggiungersi un commento testuale.

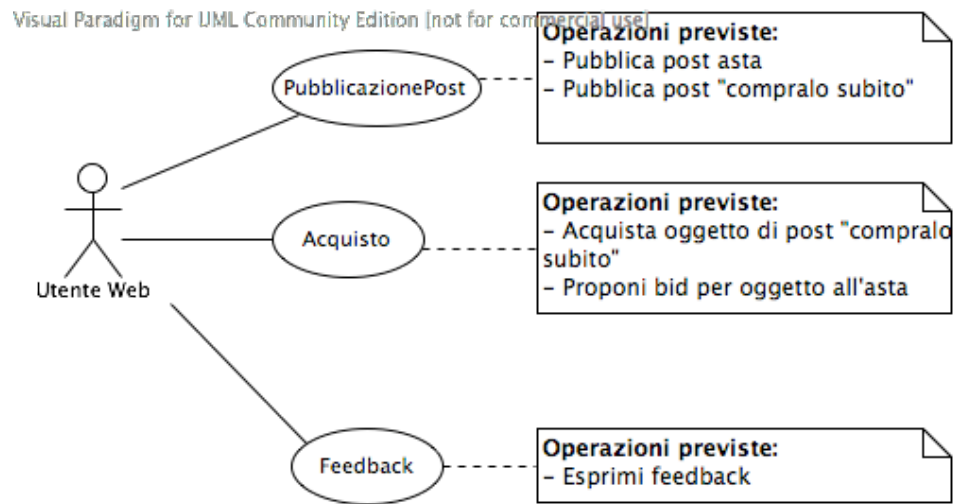
Per ogni utente, il sistema deve permettere di calcolarne l'*affidabilità*, che si calcola a partire dai feedback che ha ricevuto per gli oggetti che ha messo in vendita. In particolare, l'affidabilità di un venditore è data dalla media aritmetica m di tutti i feedback che ha ricevuto, diminuita di un fattore che è tanto più grande quanto maggiore è il numero di feedback negativi (ovvero ≤ 2). Tecnicamente, detta z la frazione dei feedback negativi rispetto ai feedback totali, l'affidabilità di un venditore è data da $m(1 - z)/5$. L'affidabilità è quindi sempre un reale tra 0 e 1 (dato che è pari ad $m/5 \in [0, 1]$ se non ci sono feedback negativi ($z = 0$), mentre scende verso 0 all'aumentare della percentuale z di questi ultimi) che, per ragioni di leggibilità si vuole mostrare come valore percentuale (tra 0 e 100).

1 Fase di Analisi

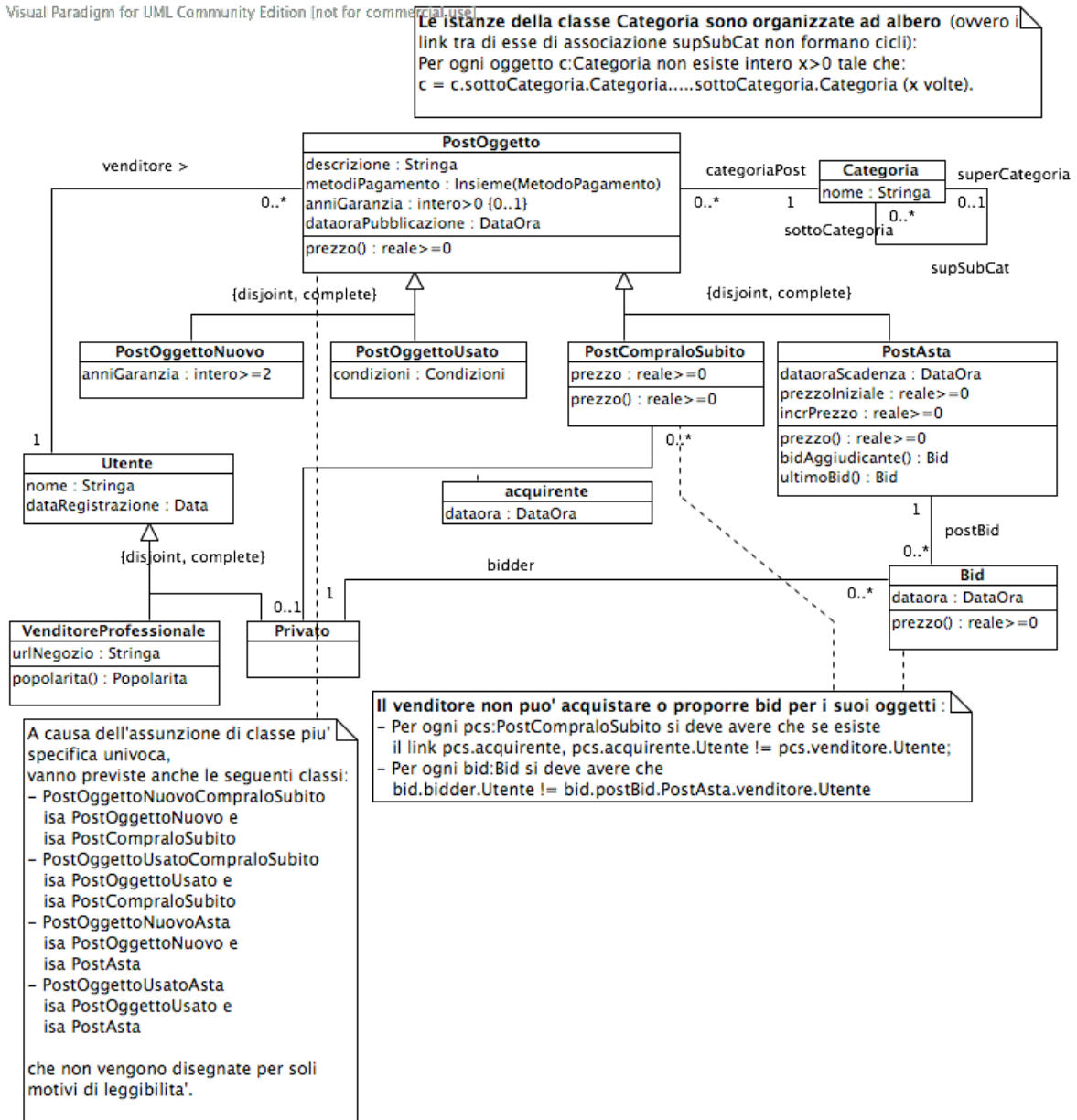
1.1 Diagramma degli Use Case del passi A.1 e A.2



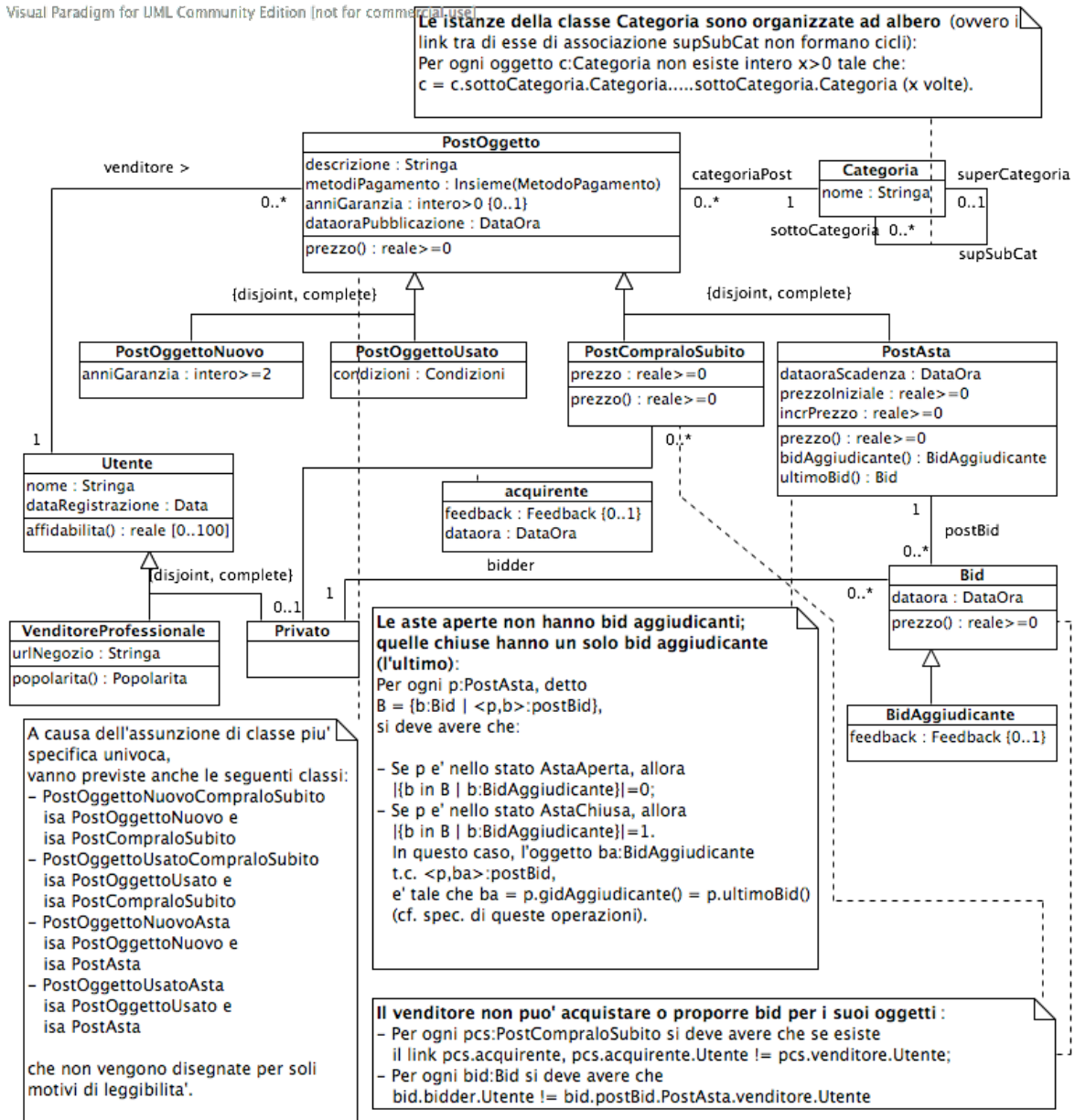
1.2 Diagramma degli Use Case



1.3 Diagramma delle classi UML del passo A.2



1.4 Diagramma delle classi UML



1.5 Specifica dei nuovi tipi di dato

```
SpecificaTipoDiDato Feedback
  attributi
    voto: 0..5;
    commento: Stringa {0,1}
FineSpecifica
```

1.6 Specifica del nuovo use case

```
SpecificaUseCase Feedback
  esprimiFeedbackAsta(u:Privato, p:PostAsta, f:Feedback)
  pre:
    - p e' nello stato 'AstaChiusa',
    - u = p.bidAggiudicante().bidder.Privato (u e' l'utente che si e' aggiudicato l'asta)
    - p.bidAggiudicante().feedback = indefinito

  post: p.bidAggiudicante().feedback = f;

  esprimiFeedbackCompraloSubito(u:Privato, p:PostCompraloSubito, f:Feedback)
  pre: <u,p>:acquirente e <u,p>.feedback = indefinito
  post: <u,p>.feedback = f

FineSpecifica
```

1.7 Specifica delle classi e diagrammi degli stati e transizioni

La specifica di alcune classi viene modificata come segue:

La classe PostAsta

```
SpecificaClasse PostAsta
  prezzo(): reale >= 0
  pre: nessuna
  post: se |this.postBid| = 0 result = this.prezzoIniziale
        altrimenti result = this.ultimoBid().prezzo()

  ultimoBid() : Bid
```

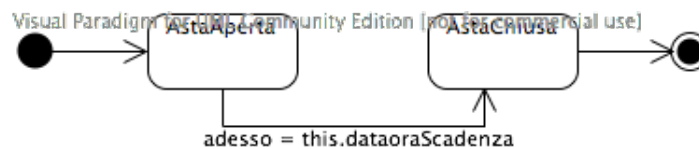
```
pre: |this.postBid| > 0
post: result e' pari all'oggetto b:Bid tale che:
- <b,this>:postBid
- per ogni altro b':Bid tale che <b',this>:postBid, si ha che b.dataora > b'.dataora
```

```
bidAggiudicante(): BidAggiudicante
```

```
pre: this e' nello stato "AstaChiusa" e le prec. di this.ultimoBid() sono soddisfatte
post: result = this.ultimoBid()
```

FineSpecifica

Gli oggetti di questa classe evolvono secondo le regole imposte dal seguente diagramma degli stati e transizioni:



La classe Bid Gli oggetti di questa classe evolvono secondo le regole imposte dal seguente diagramma degli stati e transizioni:



La classe Utente

Specificazione Classe Utente

```
affidabilita(): reale [0..100]
```

```
pre: nessuna
```

```
post: Sia P = {p:PostOggetto | p.venditore.Utente = this} l'insieme dei post pubblicati dall'utente this.
```

Siano

```
- PA = { p:P | p e' di classe PostAsta,
```

p e' nello stato "AstaChiusa" e
p.bidAggiudicante().feedback e' definito }
il sottoinsieme di P dei post in modalita' 'asta' ormai chiusi per i quali
l'aggiudicante ha espresso un feedback, e

- PC = { p:P | p e' di classe PostCompraloSubito,
p e' nello stato "Venduto" e
p.acquirente.feedback e' definito }
quello dei post in modalita' 'compralo subito' di cui si ha un
acquirente che ha espresso un feedback.

(Gli altri post o non hanno feedback, oppure sono ancora "attivi").

Sia m la media dei feedback dei post in PA unione PC:

$$m = \frac{\sum_{p \in PA} (p.bidAggiudicante().feedback.voto) + \sum_{p \in PC} (p.acquirente.feedback.voto)}{|PA| + |PC|}$$

e sia z la frazione dei feedback negativi (≤ 2) in PA unione PC:

$$z = (|\{p:PA \text{ t.c. } p.bidAggiudicante().feedback.voto \leq 2\}| + |\{p:PC \text{ t.c. } p.acquirente.feedback.voto \leq 2\}|) / (|PA| + |PC|)$$

result = m*(1-z)/5.

FineSpecifica