

# eBuy

## 1 Requisiti

### 1. Utente

- 1.1 nome
- 1.2 data registrazione
- 1.3 bid fatte

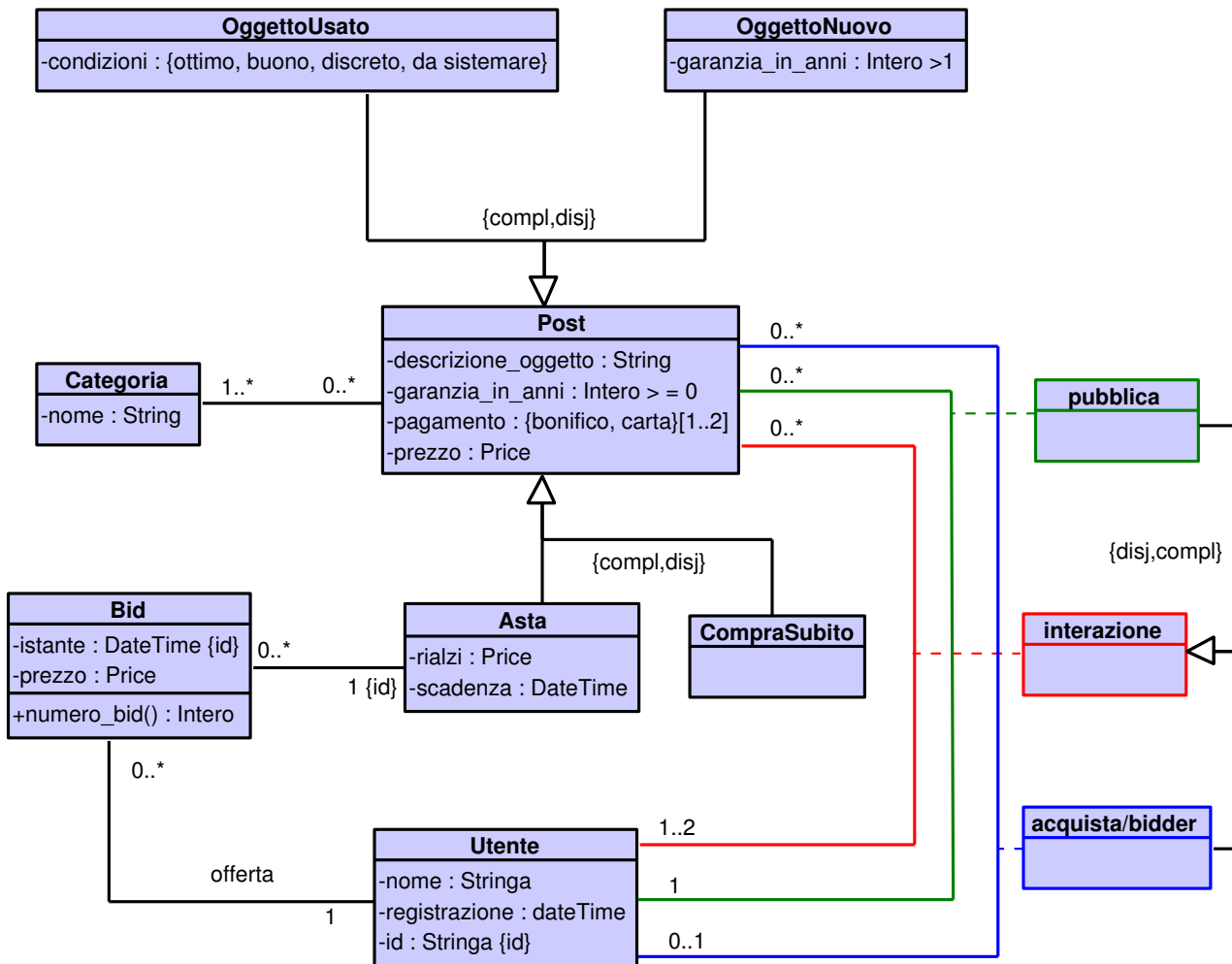
### 2. Post (annuncio)

- 2.1 descrizione oggetto
- 2.2 categoria (fai classe categoria)
- 2.3 garanziaInAnni  $\text{int} \geq 0$
- 2.4 pagabile in (implementato con enum)
  - 2.4.1 bonifico
  - 2.4.2 carta di credito
- 2.5 usato? (implementato con enum)
  - 2.5.1 nuovo
    - 2.5.1.1 garanziaInAnni  $\text{int} > 1$  (attributo specializzato)
  - 2.5.2 usato
    - 2.5.2.1 condizioni ottimo, buono, discreto, da sistemare
- 2.6 asta o no? (disjoint.complete)
  - 2.6.1 post con asta
    - 2.6.1.1 prezzo iniziale in (euro, centesimi)
    - 2.6.1.2 prezzo rialzi in (euro, centesimi)
    - 2.6.1.3 istante scadenza asta (date time)
    - 2.6.1.4 insieme di Bid (offerte)
  - 2.6.2 post senza asta
    - 2.6.2.1 prezzo in (euro, centesimi)
    - 2.6.2.1 utente che ha effettuato l'acquisto

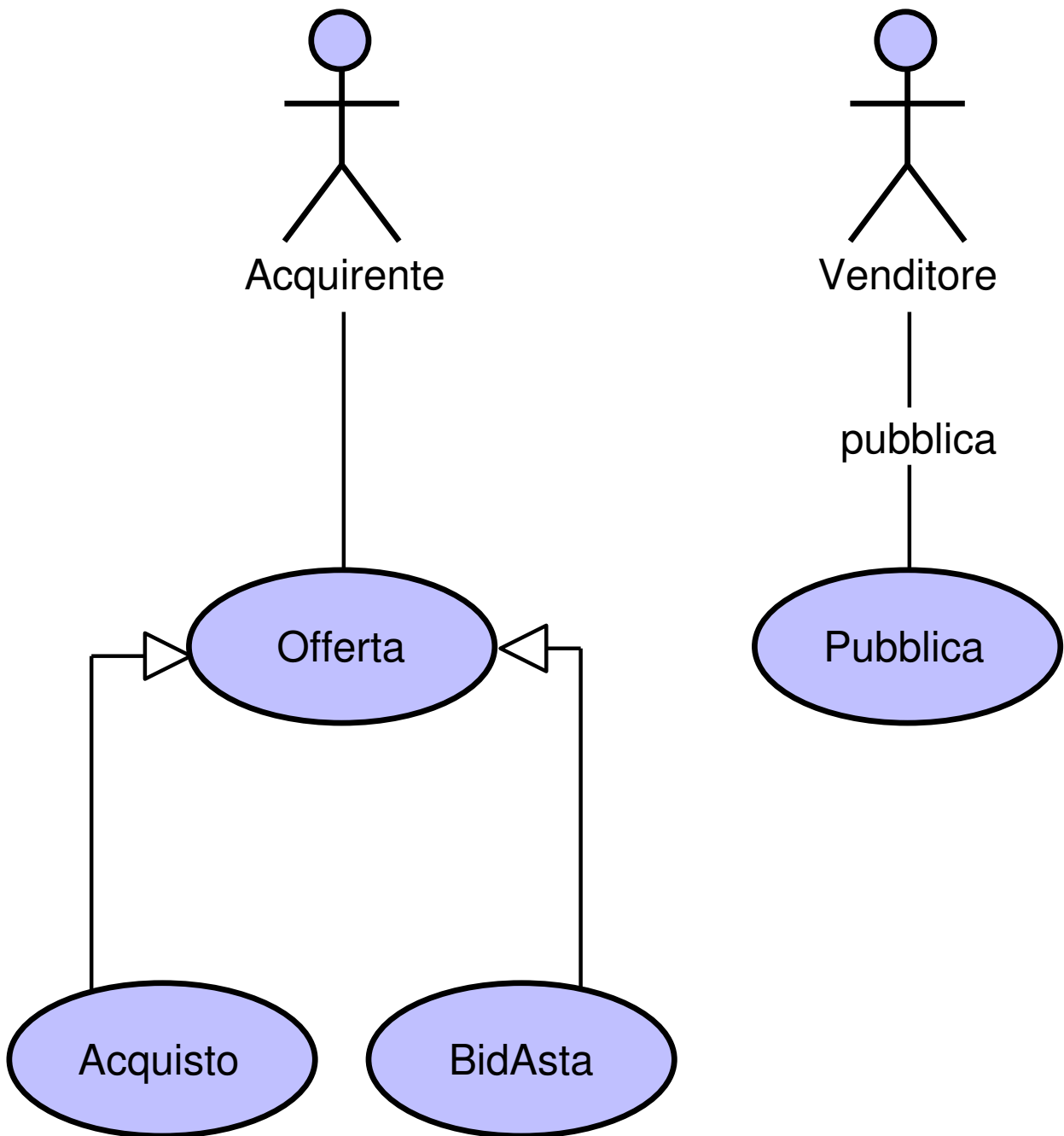
### 3. Bid (offerta)

- 3.1 Utente che ha fatto l'offerta
- 3.2 Post in questione
- 3.3 Istante offerta
- 3.4 ordine nell'offerta, n se è l'n-esimo utente che fa l'offerta
- 3.5 prezzo offerta =  $n \cdot \text{Post.rialzo} + \text{prezzo iniziale}$
- 3.6 bid vincente o no?

## 2 Diagramma UML



### 3 Diagramma Use-Case



## 4 Specifiche

### 4.1 Specifica dei tipi di dato

Price = (euro : Int > 0, cent : [0..99] )

### 4.2 Specifica delle classi

#### 4.2.1 Bid

numero\_bid () : Intero

- *pre-condizioni* : Nessuna
- *post-condizioni* : Non modifica il livello degli oggetti.  
Sia  $a : Asta$  l'oggetto per cui esiste il link ( $this, a$ )  
Sia  $B$  l'insieme di tutti gli oggetti  $x : Bid$  per cui  $\exists(x, a) \wedge x.istante < this.istante$   
 $result = |B| + 1$

### 4.3 Specifica dei vincoli esterni

[V.Bid.istante\_offerta] :  $\forall b : Bid$  e  $\forall a : Asta$  per cui  $\exists(a, b)$ , deve essere vero che  $b.istante \leq a.scadenza$ .

[V.Bid.istante\_reg\_utente] :  $\forall b : Bid$  e  $\forall u : Utente$  per cui  $\exists(u, b)$ , deve essere vero che  $b.istante \geq u.registrazione$ .

[V.Utente.scadenza\_aste] :  $\forall u : Utente$ , sia  $P$  l'insieme degli oggetti  $p : Asta$  tale che  $\exists(p, u) : vende$ .  $\forall p \in P$  deve essere vero che  $p.scadenza \geq u.registrazione$ .

## 4.4 Specifica degli use-case

### 4.4.1 Utente

`offerta (a:Asta, u:Utente) : Bid`

- *pre-condizioni* : Non deve esistere  $(u, a) : pubblica$ .
- *post-condizioni* : Viene creato un oggetto  $b : Bid$  tale che:  
 $b.istante = now$   
 $\exists(u, b) : offerta$   
 $\exists(a, b)$   
Sia  $r = a.rialzi$   
Sia  $price = a.prezzo$   
 $b.prezzo = price + r \cdot (\text{this.numero\_bid}() - 1)$   
Viene creato  $(u, a) : acquista/bidder$

`compraOggetto (c:CompraSubito, u:Utente)`

- *pre-condizioni* : Non deve esistere  $(u, c) : pubblica$ , non deve esistere un link di tipo *acquista* in cui è coinvolto  $c$ .
- *post-condizioni* : Viene creato un link di tipo  $(u, c) : scquista/bidder$ .

`creaAstaNuovo ( u : Utente, prezzoIniziale : Price, desc :Stringa, gar : Intero>1,  
pag : {Bonifico,Carta}[1..2], rialzi : Price, scad : DateTime> now,  
cat : Categoria[1..*])`

- *pre-condizioni* : Deve esistere almeno un oggetto di tipo Categoria.
- *post-condizioni* : Viene creato un oggetto  $a : Asta$  tale che  
 $a.Price = prezzoIniziale$   
 $a.descrizione\_oggetto = desc$   
 $a.garanzia\_in\_anni = gar$   
 $a.pagamento = pag$   
 $\forall c \in cat$ , crea un link  $(a, c)$   
 $a.scadenza = sca$   
 $a.rialzi = rialzi$   
Viene creato un link  $(u, a) : pubblica$ .

Simile ed analogo per i metodi relativi al:

- Creare un post (compra subito) per un oggetto usato
- Creare un post (compra subito) per un oggetto nuovo
- Creare un asta per un oggetto usato