

EBUY

Analisi concettuale

1.1 Raffinamento dei requisiti:

1. Degli utenti interessa:

- 1.1. Username (univoco)
- 1.2. Istante di registrazione (DataOra)
- 1.3. I post che hanno pubblicato (v.req. 3)
- 1.4. I post CompraSubito che hanno acquistato (v.req. 3.5.)
- 1.5. Le aste a cui hanno inviato bid (v.req. 3.4)
- 1.6. Se sono utenti privati:
 - 1.6.1. I bid proposti (v.req. 4)
 - 1.6.2. E gli oggetti acquistati in modalita "compralo subito" (v.req. 3.5)
- 1.7. Se sono venditori professionali:
 - 1.7.1. I loro post (v.req. 3)
 - 1.7.2. La URL della loro vetrina
 - 1.7.3. La popolarita calcolata come:

- La somma tra il numero di utenti che hanno fatto bid ai loro post e il numero di utenti che hanno acquistato in modalita "compralo subito" (tutto negli ultimi 12 mesi)

- Se < 50 è bassa
- Se è tra 50 e 300 è media
- Se > 300 è alta

1.7.4. L'affidabilita, calcolata:

- $m \cdot (1 - z) / 5$
- "m" è la media aritmetica di tutti i feedback ricevuti
- $z = FN / FT$, dove FN è il numero di feedback negativi e FT è il

numero di feedback totali

- I feedback negativi sono quelli che hanno voto ≤ 2

2. Degli oggetti in vendita interessa:

2.1. Se nuovo:

- 2.1.1. Il numero di anni di garanzia (minimo 2)

2.2. Se usato:

- 2.2.1. Il numero di anni di garanzia (minimo 0)
- 2.2.2. Le condizioni (ottimo, buono, discreto, da sistemare)

3. Dei post ci interessa:

- 3.1. Descrizione
- 3.2. Delle categorie interessa:

- 3.2.1. Le sottocategorie
- 3.2.2. Calcolare le dirette sottocategorie
- 3.3. Metodi di pagamento (bonifico o carta di credito)
- 3.4. Se è un'asta interessa:
 - 3.4.1. Prezzo iniziale (reale ≥ 0)
 - 3.4.2. Prezzo dei rialzi (reale > 0)
 - 3.4.3. Scadenza (DataOra)
 - 3.4.4. I bid (v.req. 4)
 - 3.4.5. Il prezzo in un dato momento (calcolabile)
 - 3.4.6. L'ultimo bid (v.req. 4)
 - 3.4.7. L'utente che si è aggiudicato l'asta (v.req. 1)
- 3.5. Se è "Compralo Subito":
 - 3.5.1. Prezzo di vendita (reale > 0)
 - 3.5.2. L'acquirente (v.req. 1), cioè colui che ha acquistato per primo
- 3.6. L'utente che l'ha pubblicato (v.req. 1)
- 3.7. L'oggetto (v.req. 2)
- 3.8. I feedback di cui interessa:
 - 3.8.1. Il voto (intero tra 0 e 5)
 - 3.8.2. Un commento testuale (opzionale)

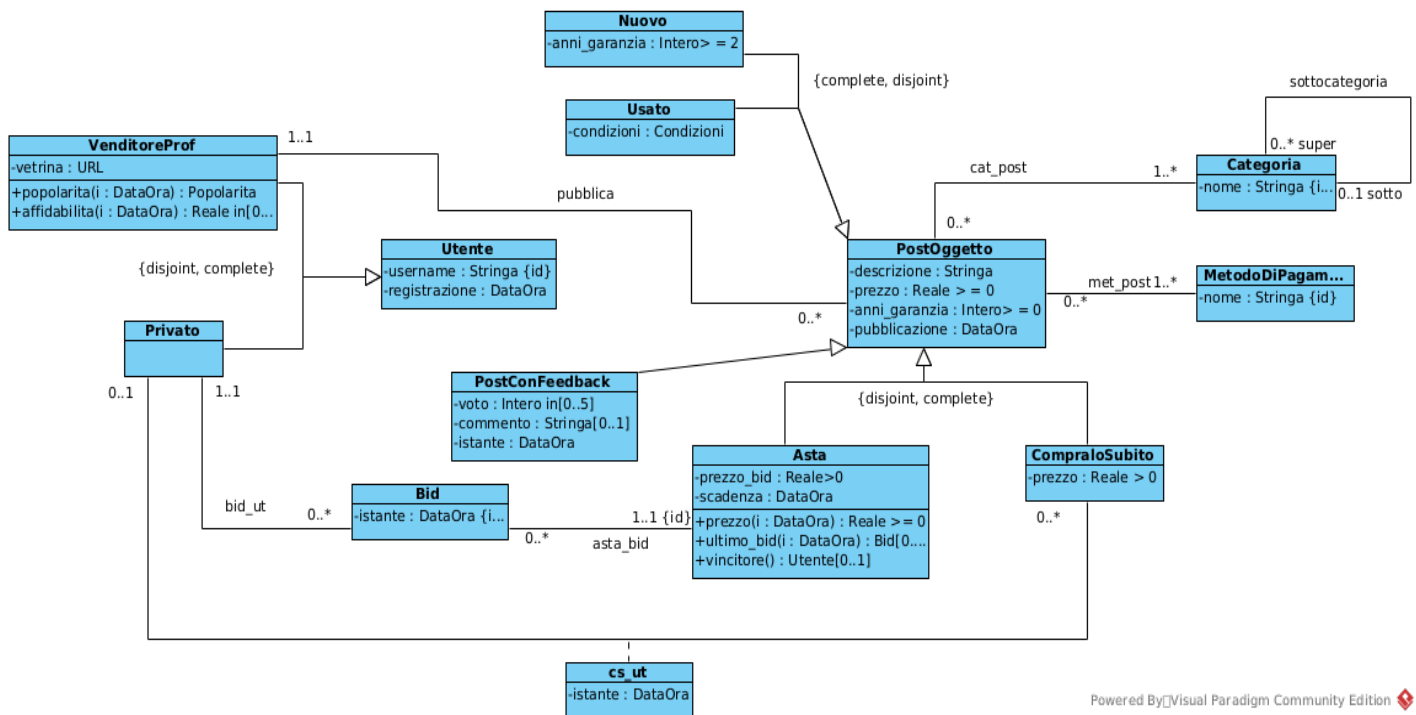
4. Dei bid ci interessa:

- 4.1. Istante (DataOra)
- 4.2. Il bidder (v.req. 1)

999. Il sistema deve permettere:

- x999.1. Ai venditori di pubblicare post per oggetti in vendita con o senza asta.
- x999.2. Ai privati di proporre un nuovo bid per un oggetto in vendita tramite asta.
- x999.3. Ai privati di procedere all'acquisto di un oggetto messo in vendita con la modalità "compralo subito".
- x999.4. Agli ospiti di registrarsi come venditori professionali o utenti privati.
- x999.5. Ai privati di lasciare feedback a post acquistati in modalita "Compralo Subito" o ad aste aggiudicate.
- x999.6. Agli amministratori di creare nuove categorie.
- x999.7. Agli amministratori di creare nuovi metodi di pagamento.
- x999.8. Ai venditori di cancellare i loro post.
- x999.9. Ai moderatori di cancellare qualsiasi post.
- x999.10. Calcolare la popolarita.
- x999.11. Calcolare l'affidibailita.

1.2 Diagramma UML



1.2.1 Specifica dei tipi di dato

Condizioni: {ottimo, buono, discreto, da sistemare}

URL: Stringa secondo standard

Popolarita: {bassa, media, alta}

1.2.2 Specifica delle classi

1.2.2.1 Specifica della classe Utente

affidabilita(i: DataOra): Reale in 0..1

precondizioni:

- Sia PF l'insieme dei pf:PostConFeedback tali che sono coinvolti con this nel link di associazione pubblica oppure priv_ou e che hanno pf.istante <= i. PF deve avere almeno un elemento.

postcondizioni:

- Non ha side effect.
- Sia S la somma dei pf.voto tali che pf in PF.
- Sia FT la cardinalità di PF.

- $m = S/FT$
- Sia PFN l'insieme dei pf in PF tali che $pf.voto \leq 2$.
- Sia FN la cardinalità PFN.
- $z = FN/FT$.
- $result = m*(1-z)/5$

popolarita(i: DataOra): Popolarita

precondizioni:

postcondizioni:

- Sia P l'insieme dei post pubblicati da this, formalmente:
tutti i $p:PostOggetto$ che partecipano a pubblica o priv_ou insieme a this.
- Sia Ucs l'insieme degli $u:Privato$ tali che partecipano a cs_ut insieme ai p in P dove l'istante di cs_ut è compreso tra $i - 12$ e i .
- Sia Ua l'insieme degli $u:Privato$, sia $b:Bid$ tale che $(b,u):bid_ut$, deve essere che $(b,p):asta_bid$ dove p in P e $i - 12 \leq b.istante \leq i$.
- $result = |Ucs| + |Ua|$

Specifica della classe Categoria

[V.Categoria.noCicli] Le sottocategorie non possono formare dei cicli.

Sia $c[0]...c[n]:Categoria$ tale che:

- $(c[0],c[1]):sottocategoria$
- $(c[1],c[2]):sottocategoria$
- ...
- $(c[n-1],c[n]):sottocategoria$

Non può essere che $(c[n],c[0]):sottocategoria$.

E

Non esiste $c:Categoria$ tale che $c':Categoria$ dove $c = c'$ e $(c,c'):sottocategoria$.

Specifica della classe Asta

[V.Asta.scadenzaDopoPubblicazione] La scadenza dell'asta deve essere dopo la sua pubblicazione.

Per ogni oggetto della classe Asta deve essere vero che $pubblicazione < scadenza$.

Più formalmente:

Per ogni $a:Asta$ deve essere $a.pubblicazione < a.scadenza$.

Equivalentemente:

Non esiste $a:Asta$ tale che $a.pubblicazione > a.scadenza$.

prezzo(i: DataOra): Reale ≥ 0

precondizioni: nessuna.

postcondizioni:

- L'operazione non modifica il livello estensionale dei dati
- Sia B l'insieme dei bid fatti a this fino ad i , formalmente:
 B è l'insieme dei $b:Bid$ tali che $(this,b):asta_bid$ e $b.istante \leq i$.
- Sia N il numero di elementi di B (formalmente, la cardinalità, $|B|$).
- $result = this.prezzo + N * this.prezzo_bid$

ultimo_bid(i: DataOra): Bid[0..1]

precondizioni: nessuna.

postcondizioni:

- L'operazione non modifica il livello estensionale
- Sia B l'insieme dei $b:Bid$ coinvolti in un link $asta_bid$ insieme a this e che hanno come valore dell'attributo istante minore o uguale a i , formalmente:
Sia B l'insieme dei $b:Bid$ tali che $(this,b):asta_bid$ e $b.istante \leq i$.
- $result$ è il b in B con valore dell'attributo istante maggiore (o massimo). Formalmente:
 $result$ è definito come il b' in B tale che non esiste un b'' dove $b'.istante < b''.istante$.

Specifica della classe VenditoreProf

popolarita(i:DataOra):Popolarità

Restituisce un valore qualitativo ("bassa", "media", "alta") che rappresenta la popolarità del venditore negli ultimi 12 mesi, in base al numero di utenti distinti che hanno:

1) fatto un bid su un suo post, oppure

2) acquistato un suo post in modalità "compralo subito"

La somma tra il numero di utenti che hanno fatto bid ai loro post e il numero di utenti che hanno acquistato in modalità "compralo subito" (tutto negli ultimi 12 mesi)

Precondizioni:

-nessuna precondizione

Postcondizioni:

- l'operazione non modifica il livello estensionale dei dati;

- sia p:Privato e b:Bid che deve essere (p,b):bid_ut;

- sia p:Privato e cs:CompraloSubito che deve essere (p,cs):cs_ut;

il risultato è uguale alla cardinalità di p che ha almeno un link(p,cs) o (p,b) con i:adesso
< privati.bid.totale < i:adesso-12 mesi

- Se < 50 result = bassa

- Se è tra 50 e 300 result = media

- Se > 300 result = è alta

affidabilità (i:DataOra):Reale in[0..1]

calcola l'affidabilità del venditore, restituendo un numero reale compreso tra 0 e 1, in base ai feedback ricevuti prima della data specificata.

Precondizione:

PostCondizioni:

- l'operazione non modifica il livello estensionale dei dati;

- il risultato è calcolato con la seguente formula: $m \cdot (1-z)/5$, di cui:

- m è la media espressa calcolando SV/VT , di cui:

-SV è la somma di ogni istanza pf.voto, con pf.istante < this.istante;

-VT è la cardinalità di tutte le istanze con pf, con pf.istante < this.istante;

- z è espressa con la seguente formula: FN/FT , di cui:

-FN è la cardinalità di pf.voto che sia ≤ 2 , con pf.istante < this.istante;

-FT è la cardinalità di tutte le istanze con pf, con pf.istante < this.istante;

Specifica della classe Bid

[V.Bid.istanteMaggioreDiRegistrazione] Gli utenti devono registrarsi prima di proporre bid.

Per ogni b:Bid sia u:Utente tale che (b,u):bid_ut allora deve essere che b.istante > u.registrazione.

[V.Bid.istanteTraPubblicazioneScadenza] L'istante dei bid devono essere tra la pubblicazione e la scadenza dell'asta.

Per ogni $b: \text{Bid}$ sia $a: \text{Asta}$ tale che $(a,b): \text{asta_bid}$ deve essere che $a.\text{pubblicazione} < b.\text{istante} < a.\text{scadenza}$.

Specifica delle Operazioni

1. prezzo($i: \text{DataOra}$): Reale ≥ 0)

Ritorna il prezzo corrente dell'asta all'istante i . Se non ci sono offerte fino a quell'istante, restituisce il prezzo di base (prezzo_bid iniziale).

Precondizioni:

i deve essere una data antecedente o uguale al tempo corrente.

Postcondizioni:

Il valore restituito è il prezzo più alto offerto fino all'istante i , o il prezzo di base se non ci sono offerte.

Specifica della classe PostOggetto

[V.PostOggetto.pubblicazioneMinoreRegistrazione] Un post deve essere pubblicato da un utente già registrato.

Per ogni $p: \text{PostOggetto}$ sia $u: \text{Utente}$ tale che $(u,p): \text{pubblica}$ deve essere che $p.\text{pubblicazione} > u.\text{registrazione}$.

Specifica dei vincoli esterni

[V.PrivatoBid.istanteMaggioreDiRegistrazione] Gli utenti devono registrarsi prima di proporre bid.

Per ogni $b: \text{Bid}$ sia $u: \text{Privato}$ tale che $(b,u): \text{bid_ut}$ allora deve essere che $b.\text{istante} > u.\text{registrazione}$.

[V.AstaBid.istanteTraPubblicazioneScadenza] L'istante dei bid devono essere tra la pubblicazione e la scadenza dell'asta.

Per ogni $b:Bid$ sia $a:Asta$ tale che $(a,b):asta_bid$ deve essere che $a.pubblicazione < b.istante < a.scadenza$.

[V.PostOggettoUtente.pubblicazioneMinoreRegistrazione] Un post deve essere pubblicato da un utente già registrato.

Per ogni $p:PostOggetto$ sia $u:VenditoreProf$ tale che $(u,p):pubblica$ deve essere che $p.pubblicazione > u.registrazione$.

[V.PostOggettoVenditoreProf.nonPreRegistrazione] I venditori non possono pubblicare post prima di essere registrati.

Per ogni $VenditoreProf$ e $PostOggetto$ coinvolti in un link di associazione pubblica deve essere che $registrazione$ di $VenditoreProf < pubblicazione$ di $PostOggetto$.

Per ogni istanza v di $VenditoreProf$ e di p $PostOggetto$ coinvolti in un link di associazione pubblica deve essere che $v.registrazione < p.pubblicazione$.

Per ogni $v:VenditoreProf$ sia $p:PostOggetto$ tale che $(v,p):pubblica$ allora deve essere che $v.registrazione < p.pubblicazione$.

[V.CompraloSubitoPrivato.acquistoMaggiorePubblicazione] L'istante di acquisto di un post in modalità "compralo subito" deve essere maggiore della pubblicazione del post.

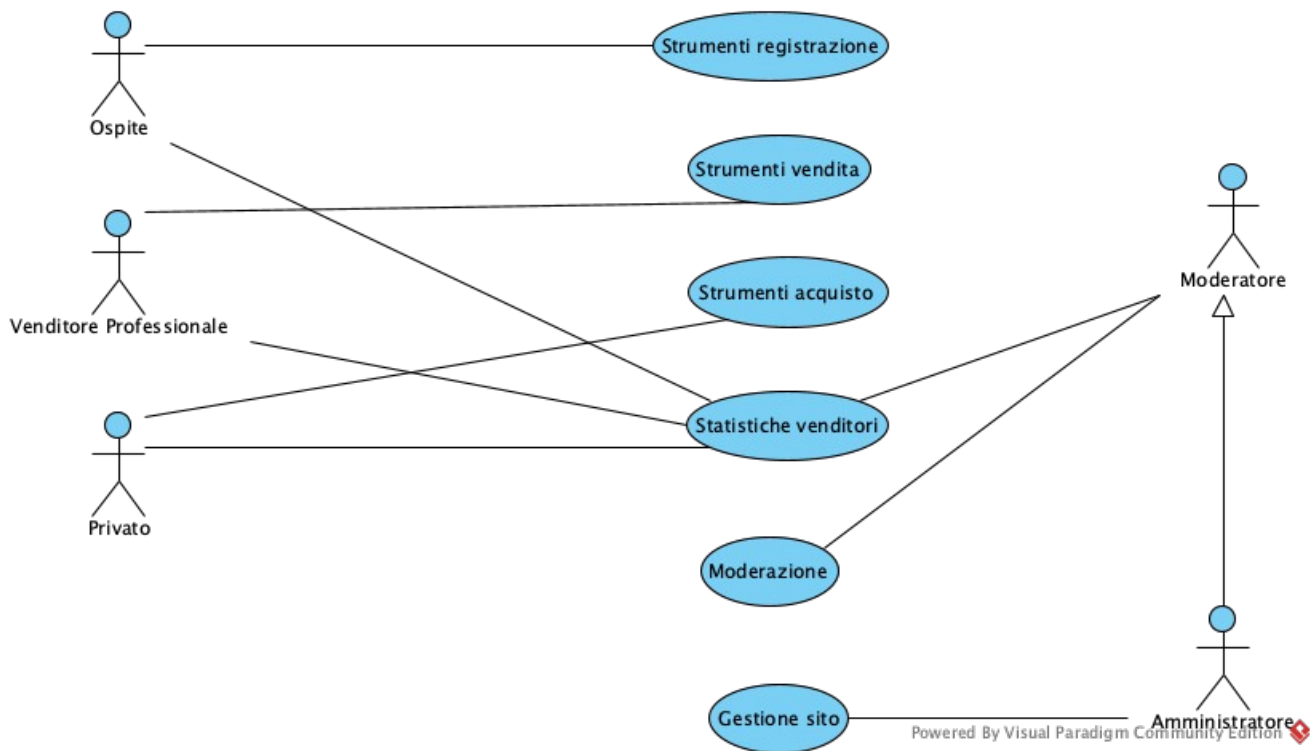
Per ogni $p:CompraloSubito$ sia $ut:Privato$ tale che $(p,ut):cs_ut$ deve essere vero che $(p,ut).istante > p.pubblicazione$.

[V.PostConFeedbackAstaCompraloSubito.feedbackValido] I post possono avere feedback solo dopo che il post in modalità "compralo subito" è stato acquistato oppure, se è un'asta, dopo che questa è scaduta ed è stata vinta da qualcuno (cioè, se c'è almeno un bid).

Per ogni $pf:PostConFeedback$:

- Se $pf:CompraloSubito$, sia $u:Privato$ deve essere $(pf,u):cs_ut$ tale che $(pf,u).istante < pf.istante$.
- Se $pf:Asta$, deve essere $pf.scadenza < adesso$ e esiste $b:Bid$ tale che $(pf,b):asta_bid$.

1.3 Diagramma degli Use-Case



1.3.1 Specifiche degli Use-Case

Specifiche dello Use-Case Strumenti Registrazione

Registra VenditoreProf(username:stringa) vetrina(username:url):VenditoreProf

Precondizione: Non esiste u:Utente con u.username = username

Postcondizione:

Viene creato e restituito l'oggetto v.venditoreProf dove :

v.username = username

v.registrazione = adesso

v.vetrina = vetrina

Registra_Privato(username:stringa):Privato

Precondizione: Non esiste u:Utente con u.username = username

Postcondizione:

Viene creato e restituito l'oggetto p.Privato dove :
p.username = username
p.registrazione = adesso

Specifica dello use case GestioneSito

creazione_pagamento(nome:String):MetodoPagamento

precondizioni:

se non esiste m:metodo con m:nome = nome

postcondizioni:

viene creato un oggetto m:metodoPagamento dove mp.nome = nome

creazione_categoria(nome:String):Categoria

precondizioni:

se non esiste c:categoria con c:nome = nome

postcondizioni:

viene creato un oggetto c:Categoria dove c.nome = nome

Specifica dello use case pubblica Asta

**pubblica_post_asta(
 descrizione: Stringa,
 condizioni: Condizioni [0..1],
 prezzo: Reale > 0,
 anni_garanzia: Intero >= 0,
 scadenza: DataOra,
 prezzo_bid: Reale > 0,
 venditore: Venditore,
 categoria: Categoria [1..*],
 metodo_pagamento: MetodoDiPagamento [1..*]
): Asta**

- precondizioni:

- se condizioni non è valorizzato allora anni_garanzia deve essere >= 2
- scadenza deve essere >= di adesso

- postcondizioni:
 - viene creato e restituito l'oggetto v:VenditoreProf dove:
 - a.descrizione = descrizione
 - a.anni_garanzia = anni_garanzia
 - a.scadenza = scadenza
 - a.prezzo_bid = prezzo_bid
 - si crea il link (v, a):pubblica dove v è un oggetto della classe VenditoreProf associato all'attore che invoca l'operazione
 - crea a: Usato se condizioni è valorizzato
 - a.condizioni = condizioni
 - crea a: Nuovo se cindizioni non è valorizzato
 - per c ogni in categoria:
 - si crea il link (c, a):cat_pos
 - per ogni mp in metodo di pagamento:
 - si crea il link (mp, a):met_pos

Specifica dello use case pubblica CompraSubito

pubblica_post_cs(
descrizione: Stringa,
condizioni: Condizioni [0..1],
prezzo: Reale > 0,
anni_garanzia: Intero >= 0,
venditore: Venditore,
categoria: Categoria [1..*],
metodo_pagamento: MetodoDiPagamento [1..*]
) : CompraSubito

- preconditioni:
 - se condizioni non è valorizzato allora anni_garanzia deve essere >= 2
- postcondizioni:
 - viene creato e restituito l'oggetto v:VenditoreProf dove:
 - cs.descrizione = descrizione
 - cs.anni_garanzia = anni_garanzia
 - si crea il link (v, cs):pubblica dove v è un oggetto della classe VenditoreProf associato all'attore che invoca l'operazione
 - crea cs: Usato se condizioni è valorizzato
 - cs.condizioni = consizioni
 - crea cs: Nuovo se cindizioni non è valorizzato
 - per c ogni in categoria:
 - si crea il link (c, cs):cat_pos
 - per ogni mp in metodo di pagamento:
 - si crea il link (mp, cs):met_pos
 - si crea il link (mp, cs):met_pos

Specifica dello Use Case Gestione Sito

privato_feedback(post: PostOggetto, voto: Intero in[0,5], commento:Stringa[0..1]): PostConFeedback:

- precondizioni:
 - Se post:Asta allora:
 - Se post.vincitore() ritorna un ut:Utente, deve essere che ut è associato all'attore che invoca l'operazione
 - Altrimenti post:CompraloSubito:
 - Deve esistere (u, cs):cs_ut, dove u è un oggetto della classe u:Privato associato all'attore che invoca l'operazione
- postcondizioni:
 - post diventa PostConFeedback, dove gli attributi di post rimangono invariati e si aggiungono:
 - post.voto = voto
 - post.commento = commento
 - post.istante = adesso

privato_bid(asta: Asta): Bid:

- precondizioni:
 - Sia B l'insieme di tutti i b:Bid, non deve esistere un altro b' tale che b'.istante = adesso e b' è coinvolto in un link (b', asta):asta_bid
- postcondizioni
 - Si crea e restituito l'oggetto b:Bid, dove:
 - b.istante = adesso
 - Si crea il link (b, ut):bid_ut, dove ut è un oggetto della classe ut:Privato associato all'attore che invoca l'operazione
 - Si crea il link (asta, b):asta_bid

privato_compralosubito(cs: CompraloSubito): cs_ut:

- precondizioni:
 - Nessuna

- postcondizioni:
- viene creato e restituito il link (ut, cs):cs_ut, dove ut è un oggetto della classe ut:Privato associato all'attore che invoca l'operazione

Specifica dello Use Case Strumenti Venditori

ottieni_popolarita(vp:VenditoreProf): Popolarita:

- precondizioni:
 - Nessuna
- postcondizioni:
 - calcola e ritorna vp.popolarita(i = adesso)

ottieni_affidabilita(vp: VenditoreProf): Reale in[0..1]:

- precondizioni:
 - Sia po:PostConFeedback con valore per l'attributo po.publicazione <= adesso, deve esistere almeno un link dell'associazione (vp, po):pubblica
- postcondizioni:
 - calcola e ritorna vp.affidabilita(i = adesso)

Specifica dello Use Case Moderazione

cancellaPost(p:PostOggetto): None

precondizioni:

se esiste un link(cs,u) allora non può essere invocata dall'attore

se p:asta:

se a.scadenza è <= istante l'operazione non può essere invocata

dall'attore

postcondizioni:

p:Post viene eliminato il link (v,p):pubblica

per ogni C in categoria:

viene eliminato ogni link (c,p):cat_post

per ogni m in mp:

viene eliminato ogni link (m,p):met_post

se p:asta:

se presenti link(b,a):asta_bid allora viene eliminato ogni link

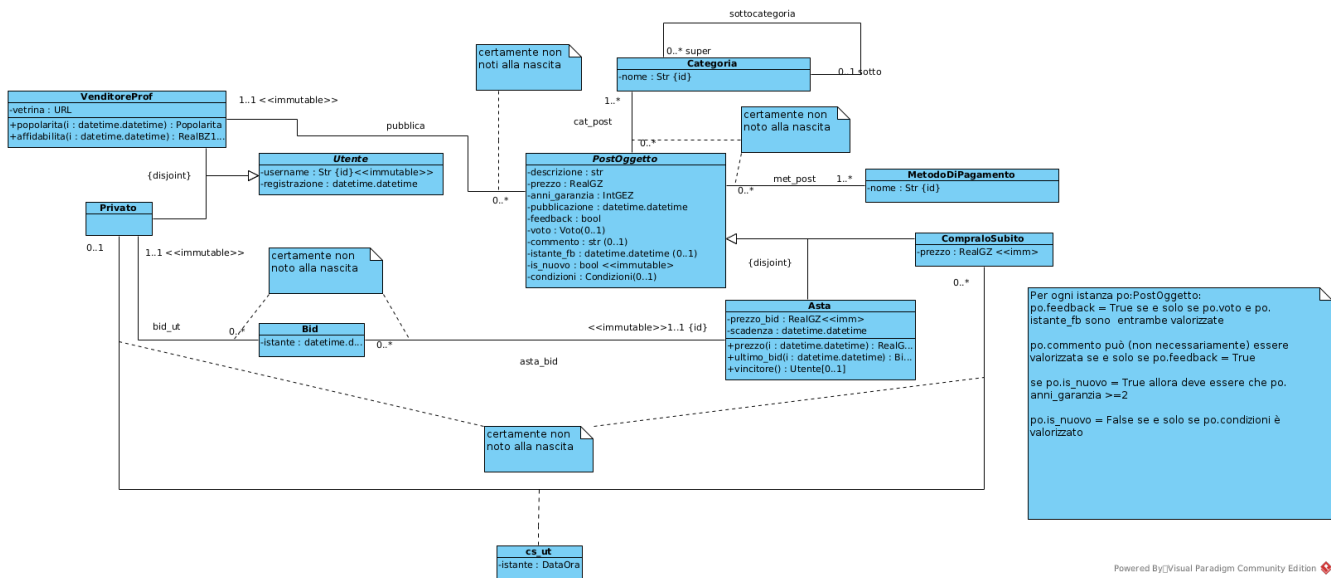
(b,a):asta_bid

se p:compraloSubito:

se presenti link(cs,u):cs_ut allora viene eliminato ogni link

(cs,u):cs_ut

1.4 Diagramma Ristrutturato



Specifiche della classe PostOggetto

Per ogni istanza po:PostOggetto:

po.feedback = True se e solo se po.voto e po.istante_fb sono entrambe valorizzate

po.commento può (non necessariamente) essere valorizzata se e solo se po.feedback = True

se po.is_nuovo = True allora deve essere che po.anni_garanzia >=2

po.is_nuovo = False se e solo se po.condizioni è valorizzato