

Descrizione progetto:

Un'asta online dove il server è il BANDITORE (accoglie le offerte, se sono accettabili o meno, comunica la migliore offerta); i clienti partecipano all'asta e devono avere le informazioni sull'offerta migliore (e di chi) e possono rilanciare se il budget (da loro inseriti) lo consente. I clienti hanno interfaccia grafica.

Banditore (server):

- Continua comunicazione dell'offerta migliore con il nome dell'offerente.
- Aspettare n° secondi dopo la comunica e conferma dell'offerta.
- In caso di termine dei secondi, comunica il vincitore e continuare con nuovo oggetto.
- Deve inserire oggetti per iniziare l'asta.

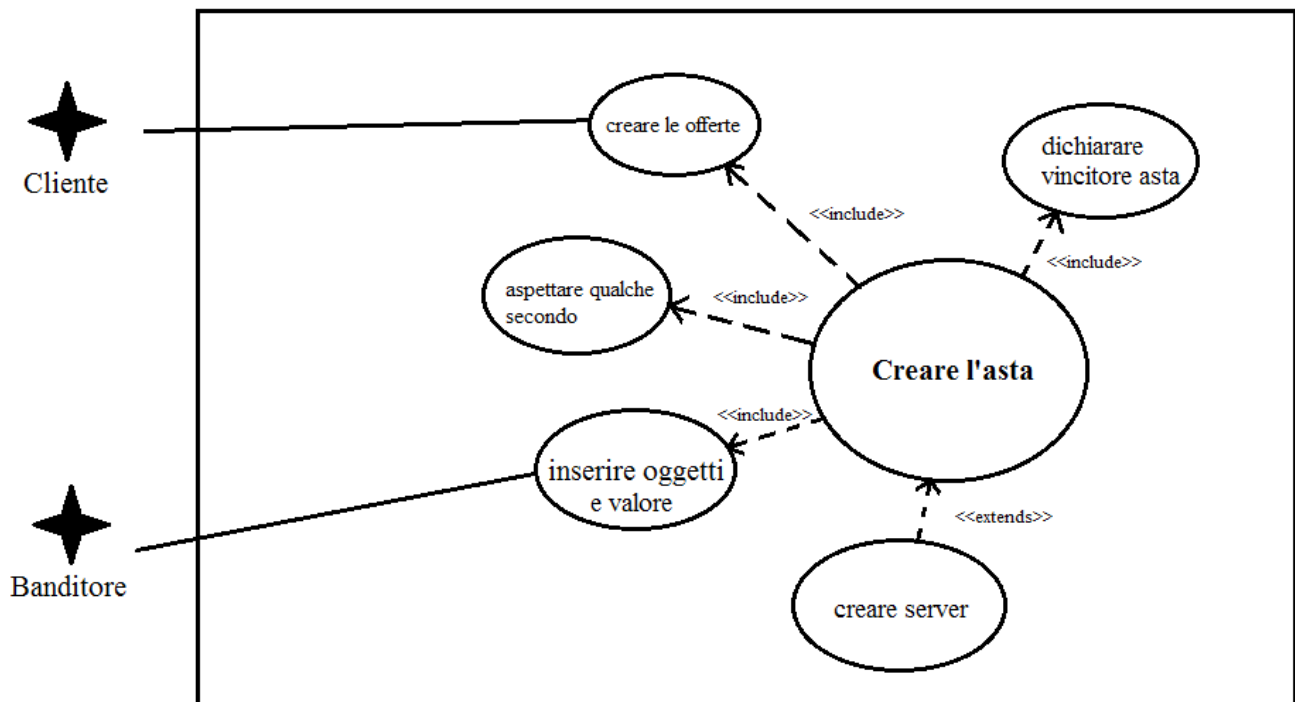
Cliente:

- Prima di iniziare l'asta, deve comunicare un budget iniziale.
- Durante l'asta, può rilanciare un offerta (deve essere $>$ dell'offerta migliore e $<$ del budget disponibile).

Specifiche di requisiti

N°	Specifiche	Tipo	Priorità
1	Il Banditore ha interfaccia grafica	Funzionale	May
2	Il Cliente ha interfaccia grafica	Funzionale	Must
3	Comunicazione continua dell'offerta migliore e nome dell'offerente	Funzionale	Must
4	Attesa dopo la comunica dell'offerta migliore	Funzionale	Must
5	Quando è finito l'attesa, comunica il vincitore e consegna l'oggetto	Funzionale	Must
6	I clienti devono essere in 12 (24)	Non Funzionale	Should
7	Il Cliente deve inserire un oggetto e il suo valore iniziale	Funzionale	Should
8	Il banditore inserisce gli oggetti	Funzionale	Must

Diagramma dei casi d'uso



UML delle classi

Percorso: Progetto 16-12\UMLDiagrams\Class Diagrams\AstaOnline.cdg

^^^^(usare NetBeans)^^^^

Diagrammi di sequenza