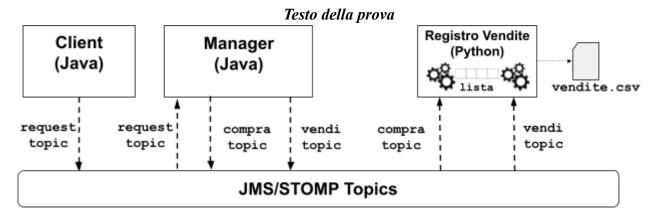
## Università degli Studi di Napoli Federico II Esame di Advanced Computer Programming

Proff. De Simone, Della Corte

## Prova pratica del giorno 17/09/2024 Durata della prova: 120 minuti

Lo studente legga attentamente il testo e produca il programma ed i casi di test necessari per dimostrarne il funzionamento. La mancata compilazione dell'elaborato, la compilazione con errori o l'esecuzione errata daranno luogo alla valutazione come prova non superata.

Al termine della prova lo studente dovrà far verificare il funzionamento del programma ad un membro della Commissione.



Il candidato realizzi un sistema di acquisto biglietti per concerti su **JMS** e **STOMP**. Il sistema include i seguenti componenti:

- Client. Componente <u>Java</u> che genera richieste di tipo *compra* o *vendi*, sotto forma di messaggi JMS. Client attende 2 secondi tra le richieste, e genera un totale di 20 richieste. Per ogni richiesta, Client genera un MapMessage composto da due campi: *tipo*, che conterrà il tipo di richiesta (cioè *compra* o *vendi*), e *valore*, che conterrà
  - La stringa *TARGA\_veicolo-quotazione* (per esempio "*AB000CD-2150*") nel caso di vendita
  - o La String ID\_cliente-budget (per esempio "giggino-2500") nel caso di acquisto Il MapMessage è poi inviato sul topic request. Il tipo di richiesta viene fornito da terminale dall'utente. Ad esempio: java nomepackage.Client compra
- **Manager.** Componente <u>Java</u> che implementa la <u>ricezione asincrona</u> JMS sul topic <u>request</u>. Alla ricezione di ciascun MapMessage, il *listener* JMS estrae i due campi (*tipo* e *valore*) e manda un TextMessage con il valore sul tipo compra o vendi a seconda della tipologia
- **Registro Vendite**. Componente <u>Python</u> che implementa la <u>ricezione asincrona</u> STOMP sul topic *compra* e sul topic *vendi*.
  - Alla ricezione di un TextMessage sul topic compra, il componente avvia un nuovo processo converte la stringa presente nel messaggio (e.g., "giggino-2500") in un dictionary (e.g., {"giggino": 2000}) e inserisce tale elemento in una lista.
  - O Alla ricezione di un TextMessage sul topic vendi, il componente avvia un nuovo processo che consumerà il primo elemento dalla coda per il quale il budget associato ad un cliente (e.g., il budget del cliente giggino è 2500 nell'esempio precedente) è maggiore o uguale alla quotazione dell'auto riferita nel messaggio che ha appena ricevuto (nell'esempio di messagio AB000CD-2150, il budget del client giggino è maggiore o uguale a 2150). Nel caso di avvenuta consumazione, aggiungere ad un file vendite.csv con intestazione uguale a targa,utente,quotazione\_auto le informazioni relative alla vendita. Per esempio il file risultante sarà:

## targa, utente, quotazione\_auto AB000CD, giggino, 2150

## L'accesso alla lista deve essere gestito in mutua esclusione tra i processi generati.

Lo studente dovrà sviluppare le applicazioni **Client**, **Manager** e **Registro Vendite**. Il sistema sarà testato da terminale con 2 Client (uno per *compra* e uno per *vendi*), 1 Manager, e 1 Registro Vendite.