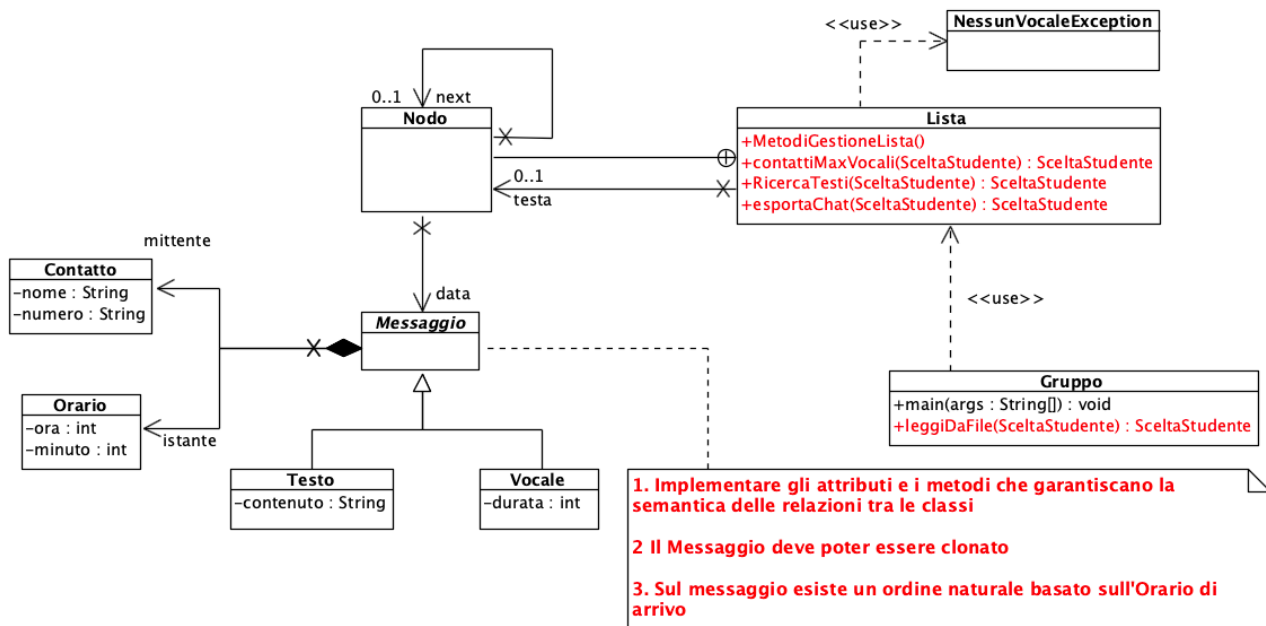


# Corso di Programmazione A.A. 2024/2025

## Esercitazione del 13 Giugno 2025

### Prof. Domenico Amalfitano

Si vuole implementare un'applicazione software per la gestione delle chat di gruppo. Il software è modellato secondo il diagramma UML riportato di seguito. Lo studente è tenuto a leggere e interpretare correttamente il diagramma per implementare le classi richieste.



### Requisiti del software.

- Ogni gruppo gestisce la chat tramite una lista concatenata di messaggi.  
**N.B. La lista dei messaggi è automaticamente ordinata in base all'orario di pubblicazione.** Lo studente non effettui controlli di consistenza sui valori degli attributi `ora` e `minuto` della classe `Orario`.
- La lista concatenata implementa la logica di ordinamento crescente per l'inserimento dei messaggi. Lo studente implementi i metodi richiesti per garantire la corretta gestione della lista ordinata.
- Lo studente implementi in modo opportuno, tenendo conto della semantica, le relazioni tra le classi `Messaggio`, `Testo`, `Vocale`, `Orario` e `Contatto`.
  - Ogni messaggio si specializza in `Testo` e `Vocale`.
    - Un messaggio di tipo `testo` ha come attributo il contenuto testuale del messaggio.
    - Un messaggio di tipo `vocale` ha come attributo un valore intero che rappresenta la durata del messaggio. Due messaggi sono uguali quando lo sono i rispettivi mittenti e istanti di pubblicazione in chat.
- Due messaggi sono considerati uguali se coincidono il mittente e l'orario di pubblicazione.
- Nel diagramma UML mancano i costruttori, il metodo `toString()` e i metodi di `get` e `set` degli attributi. Lo studente implementi i metodi che ritiene necessari tenendo anche conto della semantica delle relazioni.

6. Il metodo `contattiMaxVocali` restituisce le caratteristiche del contatto o dei contatti che hanno almeno un messaggio vocale della durata pari alla durata massima di tutti i vocali della chat di gruppo. Il metodo solleva un'eccezione (`NessunVocaleException`) sempre gestita nel caso in cui non siano presenti messaggi vocali nella chat di gruppo.
7. Il metodo `ricercaTesti` stampa a video il contenuto di tutti i messaggi testuali che contengono una stringa ricercata dall'utente e inserita da tastiera.
8. La classe `Lista` implementa un metodo `esportaChat`, che restituisce una collezione contenente la **clonazione di tutti i messaggi presenti nella chat**, mantenendone l'ordine. Lo studente implementi il metodo facendo uso del meccanismo di clonazione (`clone`) nelle classi opportune.
9. È ammesso l'uso dei container Java a supporto dell'implementazione dei metodi `contattiMaxVocali`, `ricercaTesti` e `esportaChat`.
10. Si implementi la classe utente `Gruppo` per testare le funzionalità del sistema, tenendo conto anche di diversi scenari d'uso.
11. La classe `Gruppo` carica i messaggi della chat da un file di testo opportunamente formattato dallo studente.