



**POLITECNICO
DI MILANO**

Ingegneria Del Software

Progetto Individuale
Manan Tariq – 775842

Serra Virtuale

○ Analisi dei requisiti : Diagramma i*



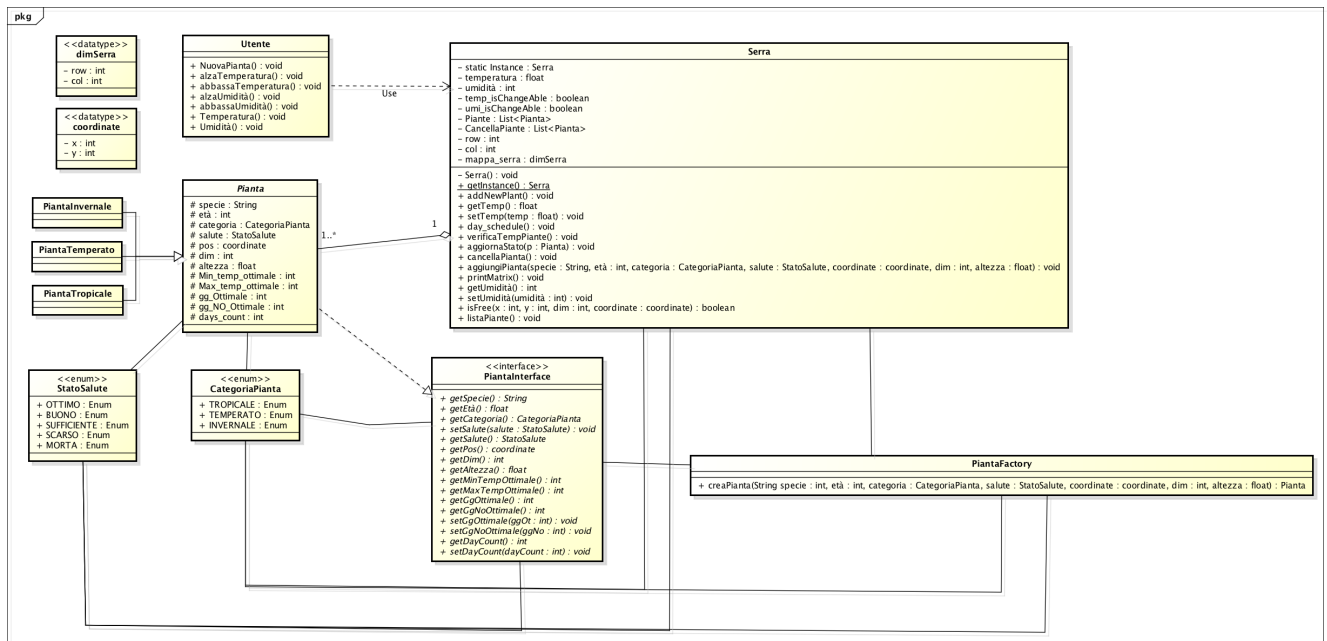
Per l'analisi dei requisiti è stato utilizzato un modello goal-oriented. In questa fase sono stati raccolti tutti i requisiti che sono necessari per soddisfare le richieste del client.

Sono stati individuati due attori principali: l'utente e la serra.

L'utente può posizionare le piante, regolare la temperatura (0.5) e l'umidità (5%) ogni mezz'ora.

La serra virtuale riceve i comandi dall'utente per impostare la temperatura e l'umidità. I task "imposta umidità" e "imposta temperatura" dipendono rispettivamente dalla risorsa "alta/bassa temperatura" e "alta/bassa umidità". Inoltre la serra controlla la temperatura e l'umidità una volta al giorno. Per ogni pianta passati i dieci giorni aggiorna lo stato di salute.

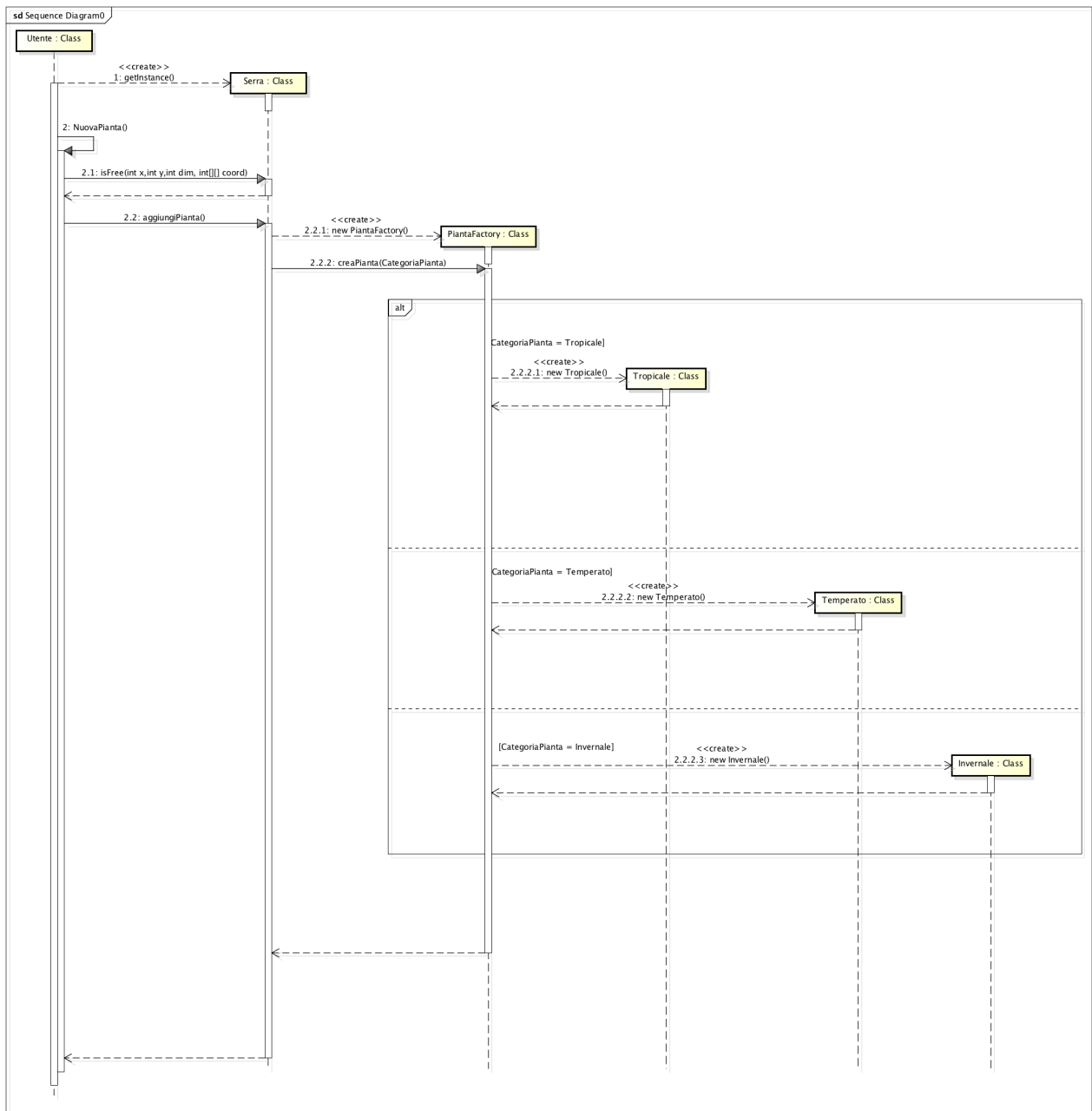
○ Progettazione UML: Class Diagram



Il class diagram fornisce un'idea generale sulle caratteristiche e le relazioni tra le classi da implementare nel progetto. Per facilitare la progettazione sono stati usati i seguenti design patterns: Singleton e Factory.

Il singleton è usato per avere un'unica istanza della classe Serra. La classe utente esegue i comandi usando un'unica istanza. La classe PiantaFactory è un esempio di factory pattern. La classe serra usa il metodo creaPianta di piantaFactory per creare le piante. PiantaFactory è associato all'interfaccia PiantaInterface che espone diversi metodi implementati dalla classe Pianta. La classe Pianta è una classe astratta i suoi metodi vengono implementati dai suoi sotto classi PiantaTropicale, PiantaTemperato e PiantaInvernale.

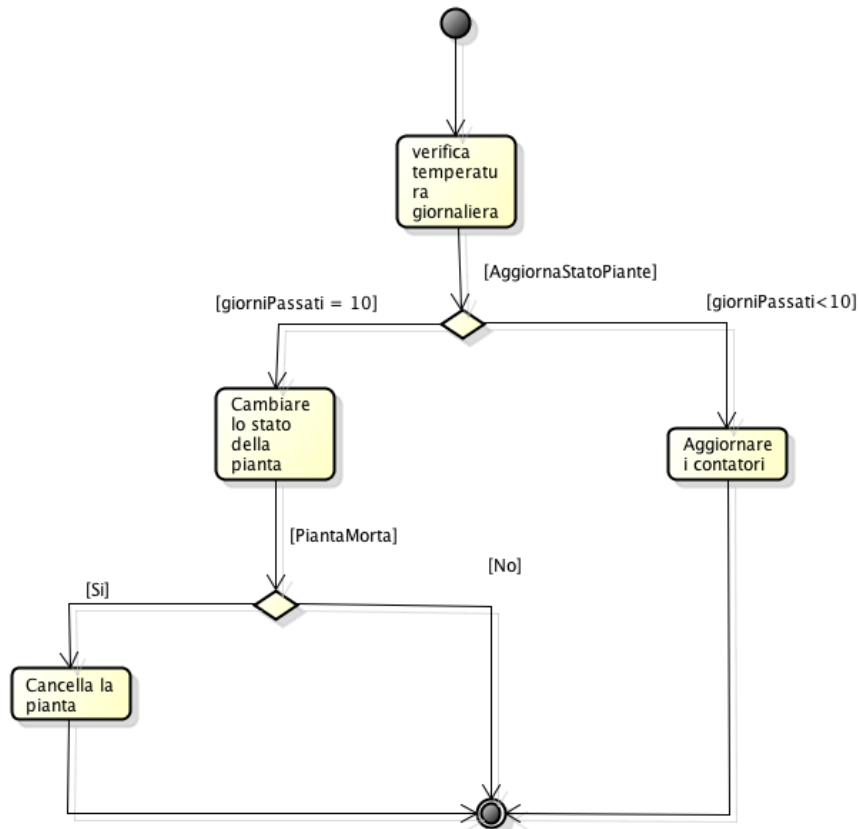
Sequence Diagram:



Il sequence diagram descrive l'andamento nel tempo di una particolare azione. In questo caso è stato utilizzato per descrivere l'aggiunta di nuova pianta.

L'utente sceglie la dimensione e la posizione in cui mettere la pianta. La serra controlla se c'è abbastanza spazio per la pianta con il metodo isfree che ritorna un array con le coordinate delle posizioni. Il client chiama metodo aggiungiPianta della serra a cui passa anche tutti i dati della nuova pianta. Il metodo creaPianta della piantaFactory crea e ritorna l'istanza di una nuova classe a seconda del tipo di pianta richiesta. La classe pianta appena creata viene salvata nella serra.

Activity Diagram:



L'activity diagram descrive il flusso delle attività che si svolgono nel sistema. Nel diagramma sopra viene mostrata l'attività giornaliera del controllo della temperatura della serra e l'aggiornamento degli stati delle piante.

○ Junit Test

Junit test per controllare se i singoli pezzi di codice svolgono le loro funzioni seguendo i requisiti. Sono stati implementati tre metodi di test:

- Temptest() : per controllare se l'incremento della temperatura funziona (da 20 a 20.5) .
- isFreetest() : controlla se lo spazio in cui l'utente vuole aggiungere la nuova pianta è libero o no.
- statoPiantatest(): controlla se passati i dieci giorni lo stato di salute della pianta viene modificata (es. da sufficiente a scarso).