

# Ingegneria del Software A.A. 2018/2019

## Esame 2019-07-15

---

### Esercizio 1 (6 punti)

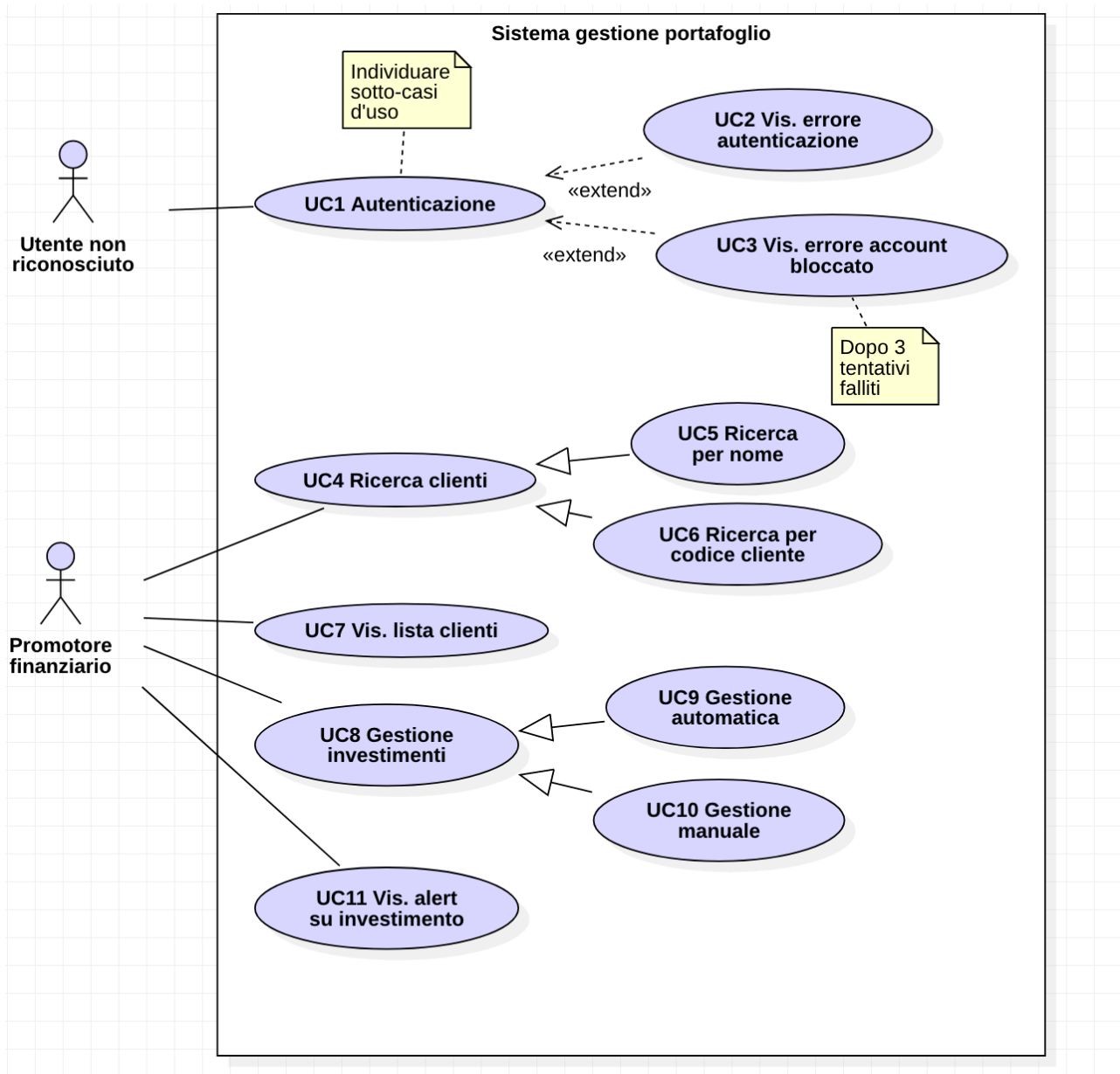
#### Descrizione

La piattaforma di investimenti della banca *SmartBank* permette ai promotori finanziari di proporre ai propri clienti soluzioni diversificate e altamente personalizzabili. La piattaforma prevede autenticazione tramite numero unico di matricola (codice cliente) e *password* associata. L'errore nell'inserimento della *password* porta alla visualizzazione di un errore. Al terzo errore di immissione *password*, il sistema blocca l'utente, impedendogli di autenticarsi. Ogni promotore può effettuare ricerche all'interno del proprio portafoglio clienti, per nome o codice cliente. Per ciascun cliente, la lista visualizza il nominativo (nome e cognome), il profilo di rischio associato (alto/medio/basso), e il capitale totale gestito dalla banca. Il dettaglio della posizione del cliente permette la gestione dell'insieme di fondi associati agli investimenti. Il promotore può scegliere fra due tipologie di investimenti: manuale, e automatico. Nel caso dell'investimento manuale, il promotore deve inserire il nome dell'azione da comprare, scegliendola fra quelle disponibili, e l'importo massimo da impiegare. Nel caso dell'investimento automatico, il promotore deve solo specificare quanto il cliente possa rischiare, mentre il sistema opera in autonomia comprando o liquidando diverse quantità di azioni da diversi fondi, fino al raggiungimento del *plafond* corrispondente al livello di rischio fissato. Qualora il capitale di un cliente sia stato intaccato per più del 30% rispetto all'investimento iniziale, il sistema genera un *alert*, visualizzabile all'interno dell'applicazione.

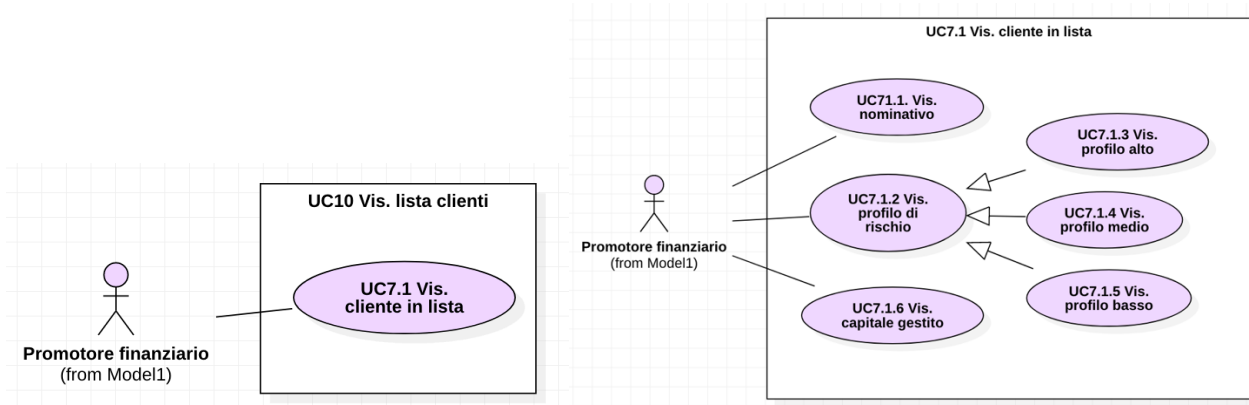
Si utilizzino i diagrammi dei casi d'uso per modellare gli scenari sopra descritti. Non ne è richiesta la descrizione testuale.

#### Soluzione

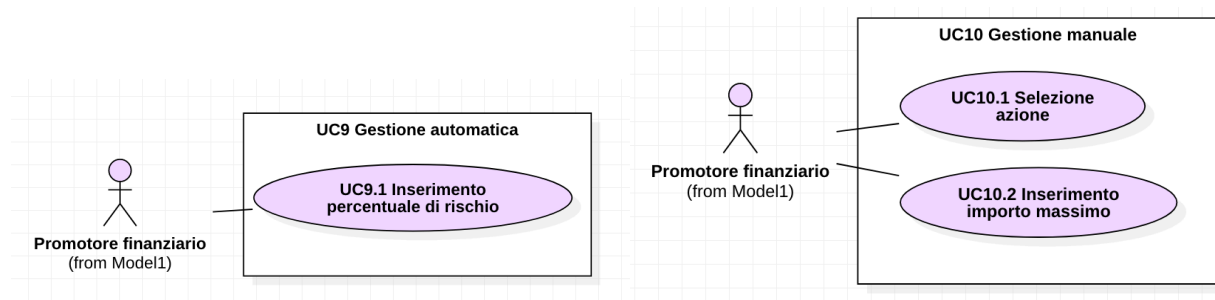
Un possibile diagramma dei casi d'uso associato al suddetto testo è il seguente.



La gestione della visualizzazione delle informazioni dei singoli clienti è modellata dai seguenti casi d'uso.



Infine la gestione degli investimenti può essere modellata come segue.



## Esercizio 2 (7 punti)

### Descrizione

L'applicazione per gli investimenti della *SmartBank* è tecnologicamente all'avanguardia. Il motore di investimento tiene monitorate costantemente diverse fonti di informazione digitali esterne al sistema, come ad esempio Twitter. Restando in ascolto sui *tweet* di diversi profili influenti sul mercato finanziario, il sistema valuta ogni nuovo post pubblicato. Per questioni di proprietà intellettuale, il sistema non può usare le API fornite da Twitter. Il testo recuperato dalla fonte Twitter viene trasformato in *token* (o *term*). Successivamente, usando un algoritmo di apprendimento automatico, il sistema calcola una misura di quanto il *tweet* possa influire negativamente o positivamente su un determinato titolo azionario. Tale algoritmo è alla base di tutta la piattaforma e pertanto nuove versioni migliorative di esso vengono rilasciate regolarmente. Sulla base della misura di influenza determinata dall'algoritmo, il sistema scorre il portafoglio di azioni di ogni singolo cliente, utilizzando un criterio di attraversamento basato sul valore delle azioni. Le azioni vengono quindi vendute o comprate di conseguenza.

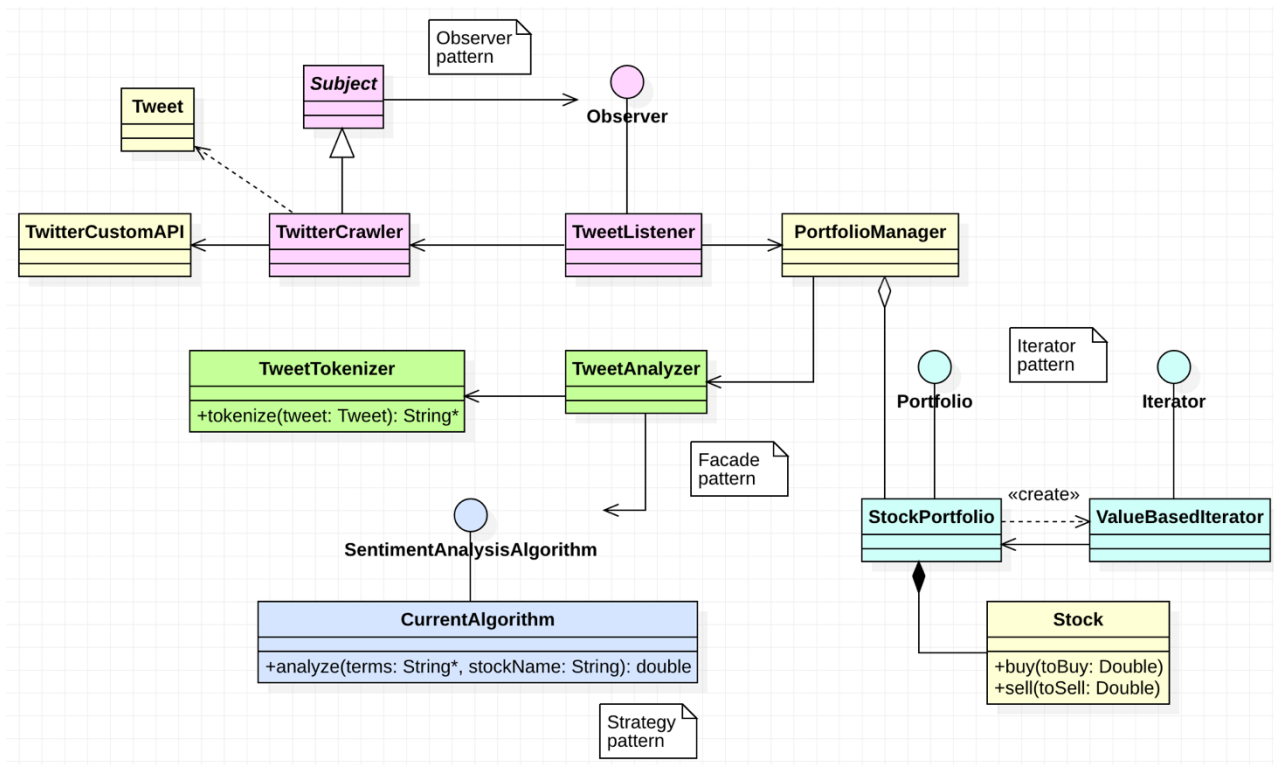
Si modelli il sistema sopra descritto mediante un diagramma delle classi e i *design pattern* a esso pertinenti. Immaginando un nuovo *tweet* sul profilo della CNN, si utilizzi un diagramma di sequenza per descrivere l'interazione fra le componenti individuate, immaginando una vendita di azioni Apple e un acquisto di azioni Google all'interno di un portafoglio cliente.

### Soluzione

Una possibile soluzione prevede l'utilizzo dei seguenti *design pattern*:

- Observer pattern
- Strategy pattern
- Iterator pattern
- Facade pattern

Il diagramma delle classi corrispondente è il seguente.



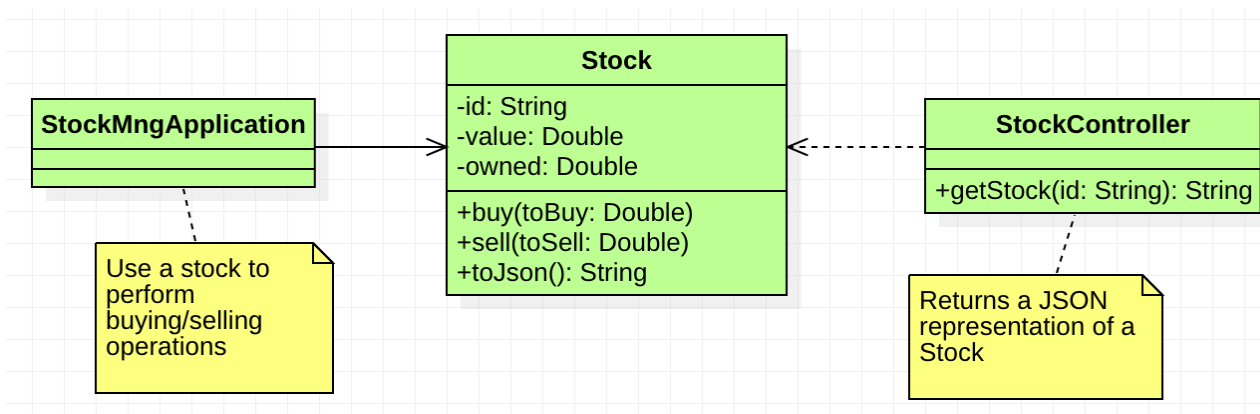
Il diagramma di sequenza deriva linearmente dal suddetto diagramma.

### Esercizio 3 (3 punti)

#### Descrizione

La classe `Stock` è utilizzata per gestire oggetti che rappresentano azioni finanziarie, sia come parte del processo di compravendita, che per esporre tali informazioni in formato JSON tramite un servizio REST.

Si modifichi il diagramma delle classi riportato in figura in modo da farlo aderire al “Single Responsibility Principle”.



#### Soluzione

L’applicazione del “Single Responsibility Principle” permette di isolare le dipendenze rispetto ai client della classe `Stock`, che deve essere pertanto suddivisa in due tipi distinti.

