

# Ingegneria del Software A.A. 2018/2019

## Esame 2019-09-10

---

### Esercizio 1 (6 punti)

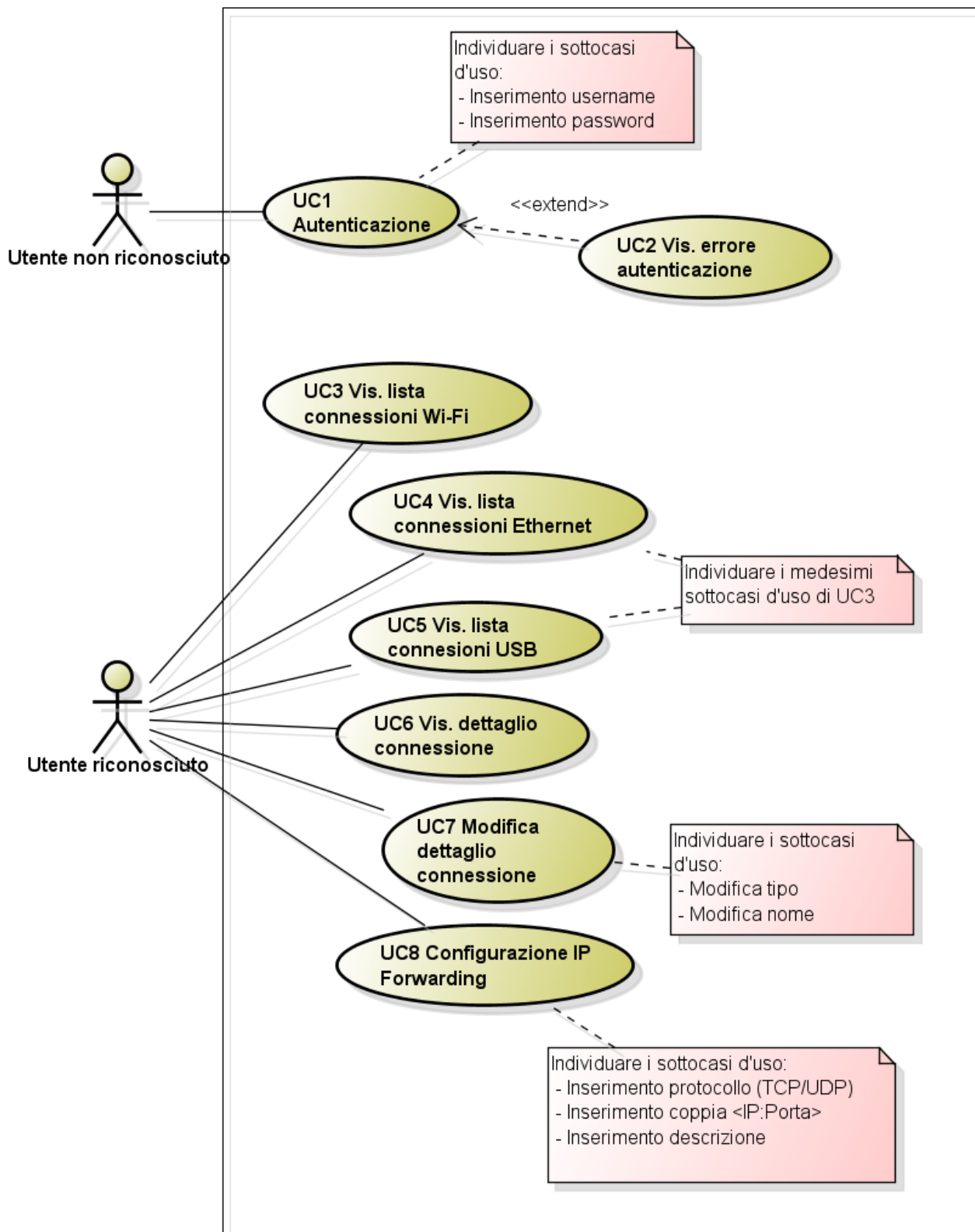
#### Descrizione

I *router* moderni spesso forniscono una interfaccia web dedicata alla configurazione e disponibile su un indirizzo di rete locale, per esempio 192.168.1.1. Per accedere alla configurazione, è necessario autenticarsi, fornendo *username* e *password*. La *landing* page raggiunta dopo l'autenticazione presenta la situazione delle connessioni di rete attualmente attive per quel *router*, suddivise per tipologia: wi-fi, Ethernet, USB. Ogni lista visualizza un identificativo del dispositivo connesso, e i suoi indirizzi MAC e IPv4. La selezione del singolo dispositivo visualizza informazioni di dettaglio, come lo stato (Connesso/Non connesso), il tipo (Generico/Computer desktop/Computer laptop/ Telefono), e un nome descrittivo. Le ultime due voci possono essere modificate. È inoltre possibile configurare il *port forwarding*, ossia la redirectione di una coppia verso un'altra coppia. Per tale operazione occorre specificare il tipo di protocollo (TCP/UDP), le due coppie, e una breve descrizione testuale.

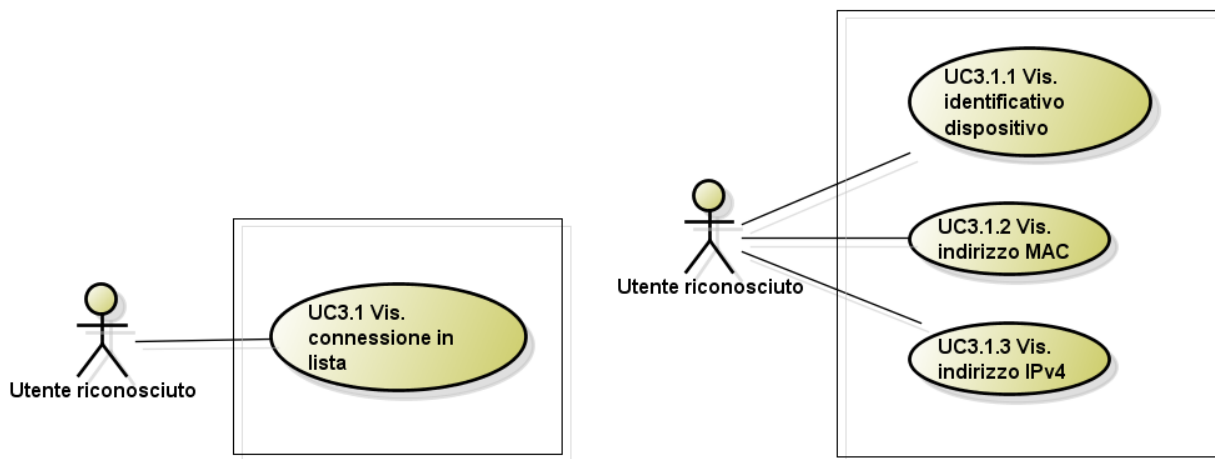
Si utilizzino i diagrammi dei casi d'uso per modellare gli scenari sopra descritti. Non ne è richiesta la descrizione testuale.

#### Soluzione

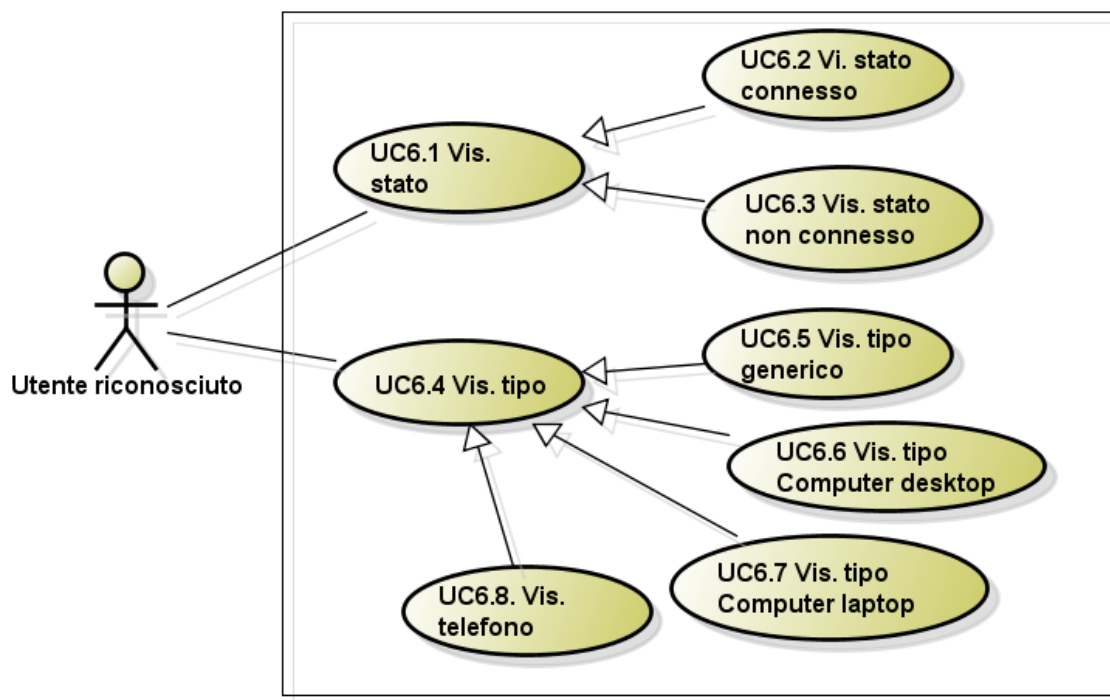
Una possibile soluzione è la seguente.



I sottocasi individuabili sono i seguenti.



E i seguenti.



## Esercizio 2 (7 punti)

### Descrizione

Una *start-up* innovativa sta sviluppando una stampante che utilizza inchiostri naturali, non inquinanti, per la riproduzione di testo e immagini. Rispetto alla stampa industriale, la particolarità del progetto sta nell'applicazione di differenti algoritmi, flessibilmente adattabili, per la composizione dei colori. Alla base della progettazione software, sta il tipo `Printer` che ha la responsabilità di inviare istruzioni di lavoro alle testine della stampante. Poiché l'algoritmo di selezione dei colori è attualmente ancora in stadio sperimentale, i progettisti dell'azienda hanno necessità di adottare una architettura che ne faciliti l'evoluzione. Gli oggetti da inviare alla stampa sono inizializzati dalle informazioni provenienti da due *listener*, che ascoltano due sonde, poste rispettivamente sulla porta USB e sul connettore wi-fi della stampante. Il tipo di oggetto da stampare viene specificato dall'esterno, e i *listener* semplicemente

acquisiscono e inoltrano tale informazione. La trasformazione degli oggetti in istruzioni di lavoro per le testine avviene utilizzando un componente dedicato. Parte di tali istruzioni sono comuni per i diversi tipi di dato (testo o immagine), parte invece dipende da esso. L'architettura *software* prevede poi un componente che, ricevuto dai *listener* un comando speciale, invia alle testine le istruzioni per effettuare una stampa di prova di un'immagine precaricata nella memoria SSD della stampante.

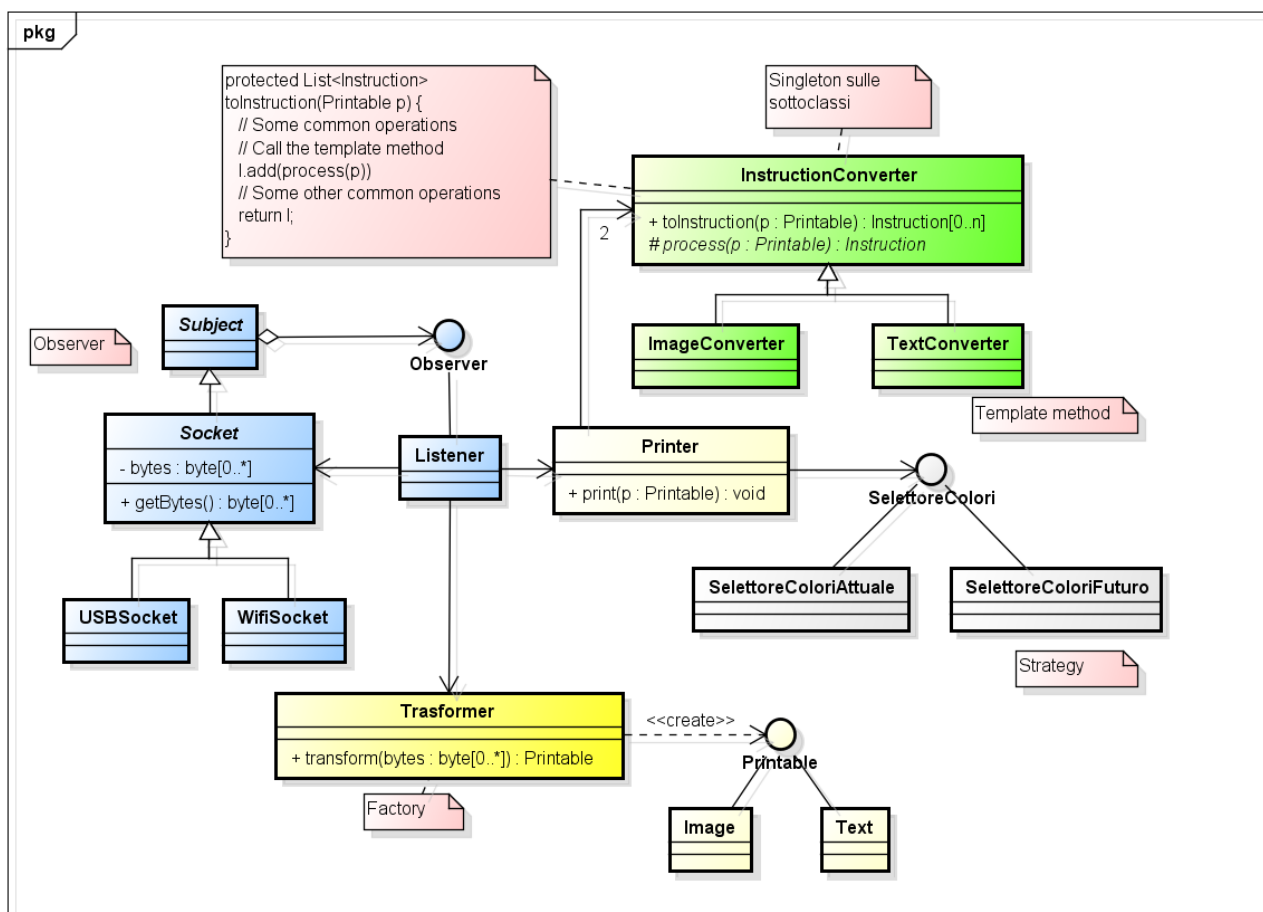
Si modelli tale sistema mediante un diagramma delle classi, comprensivo dei *design pattern* a esso pertinenti. Utilizzando un diagramma di sequenza, si descriva poi la stampa di una immagine.

## Soluzione

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

- Observer
- Strategy
- Template Method
- Factory
- Singleton

Un possibile diagramma delle classi è il seguente.



Il diagramma di sequenza corrispondente è facilmente individuabile.

### Esercizio 3 (3 punti)

#### Descrizione

Il *framework* delle *Collection* in Java permette di ordinare una lista in modo arbitrario utilizzando il metodo della classe `Collections`, `static void sort(List list, Comparator c)`. L'interfaccia `Comparator` contiene un unico metodo `int compare(T o1, T o2)`. Tale *framework* utilizza un noto *design pattern* della GoF.

Si fornisca il diagramma delle classi che lo contestualizza, per una lista di oggetti della classe `Person`, utilizzando un comparatore che ordina per `surname`.

#### Soluzione

Il *design pattern* utilizzato è lo *Strategy Pattern*. Di seguito si riporta il diagramma delle classi richiesto.

