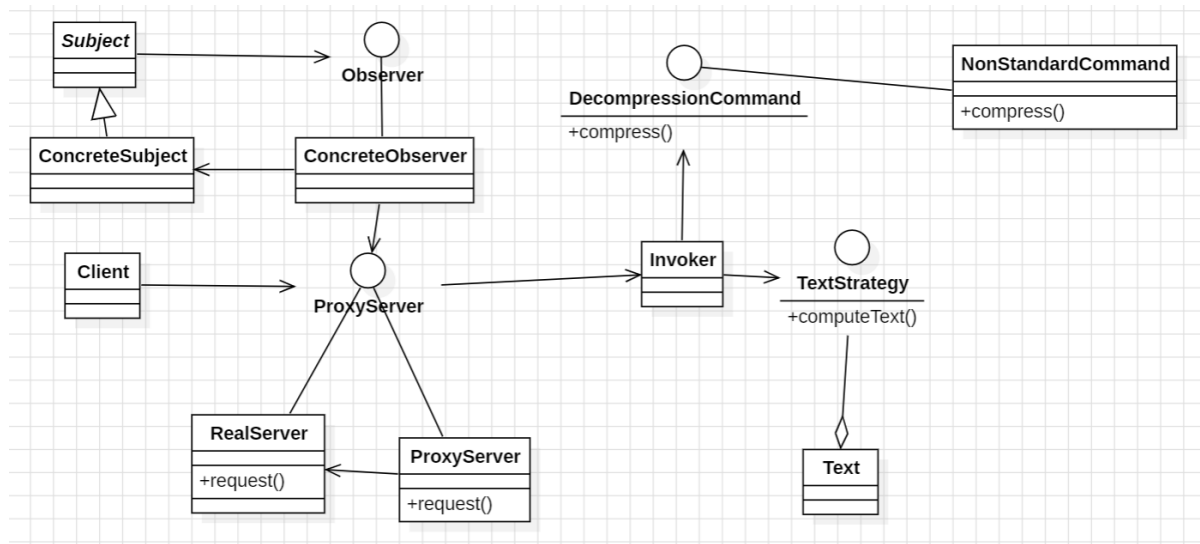


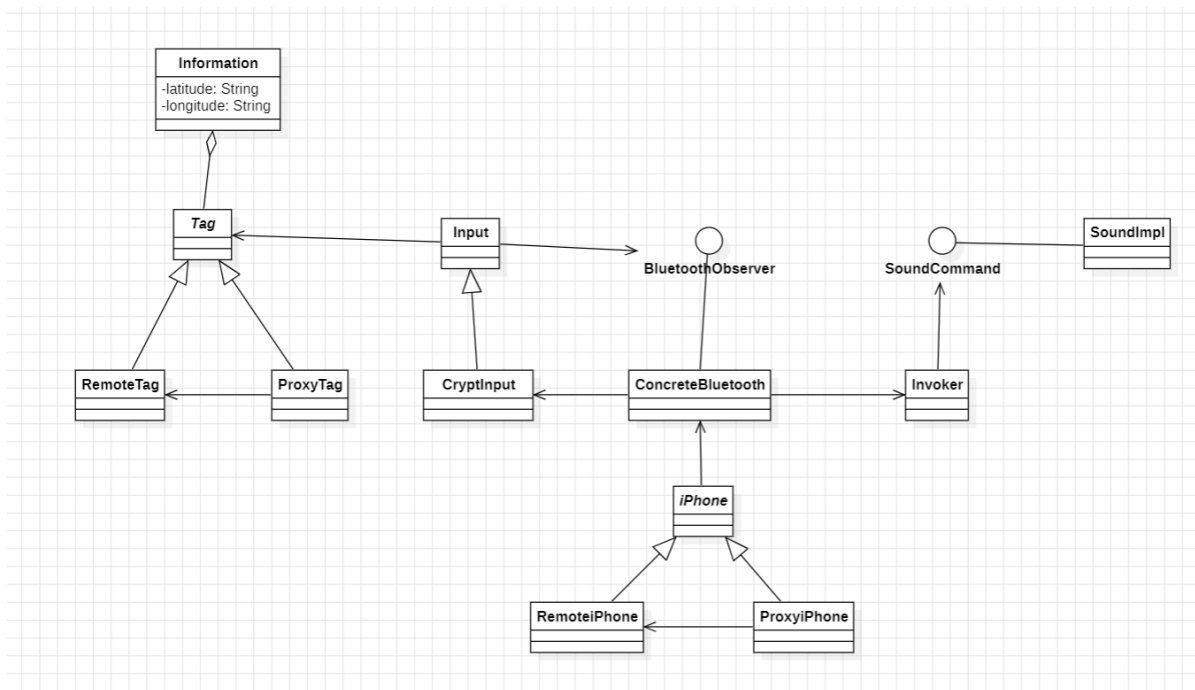
I *chatbot* intelligenti, come ChatGPT o Google Bard, sono uno degli argomenti più discussi negli ultimi mesi. Una *startup* italiana sta cercando di svilupparne una propria versione, chiamata *Dante*. Il sistema è composto di due componenti, una *client* e una *server*. La prima, si occupa di pre-elaborare il testo inserito dall'utente, dividendolo dapprima in termini (*terms*), successivamente costruendo una struttura ad albero che rappresenta semanticamente il testo, e infine effettuando una compressione degli stessi con un algoritmo proprietario. Fatto questo, il *client* invia le informazioni compresse al *server*. In questa interazione, *client* e *server* operano come se entrambi fossero localizzati sulla medesima macchina, ma la loro comunicazione è *asincrona*. Il *server*, decomprime quanto ricevuto e lo interpreta, usando un algoritmo la cui implementazione richiede lo scorrimento della struttura ad albero in modo non *standard*. Appena pronto, il testo risultante, contenente la risposta a quanto richiesto dall'utente, viene ritornato al *client*.

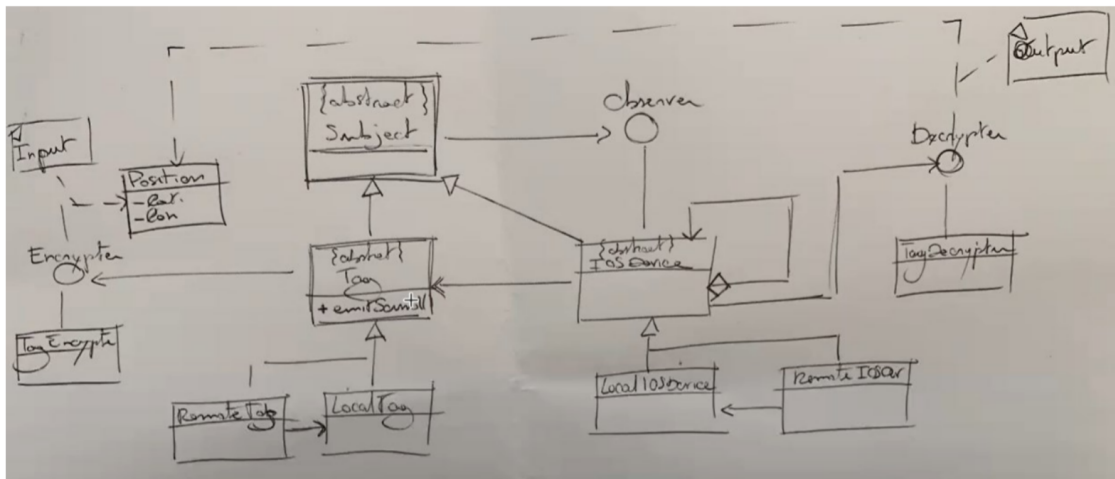
Si modella tale sistema mediante un diagramma delle classi, comprensivo dei *design pattern* a esso pertinenti.



Apple ha recentemente lanciato sul mercato la sua versione dei *tag bluetooth*, chiamata AirTag. Ogni *tag* è un dispositivo fisico capace di connettersi a dispositivi Apple, come iPhone, iPad e MacBook, utilizzando il canale di comunicazione *bluetooth*. A intervalli regolari, ogni *tag* invia un segnale crittato contenente la sua posizione (latitudine e longitudine). Un iPhone che entra nel raggio di azione del tag, e che precedentemente era stato abbinato al tag stesso, rileva l'informazione con il proprio sensore in ascolto sul canale *bluetooth*. Per facilitarne la gestione, il tag, pur non eseguendo fisicamente sul sistema operativo del telefono, è visto come locale ad esso. L'iPhone, quindi, invia a sua volta la posizione del tag e il suo identificativo a tutti i dispositivi che si trovino nel suo raggio di azione *bluetooth*. In questo modo, tali telefoni formano una rete estesa, capace di rilevare tag anche molto distanti dai telefoni associati. Anche in questo caso, gli iPhone dialogano tra loro come se fossero locali. Infine, un telefono associato a un tag può richiedere a quest'ultimo di emettere un suono, utilizzando un opportuno comando, se collegato direttamente alla medesima rete *bluetooth*.

Si modella tale sistema mediante un diagramma delle classi, comprensivo dei *design pattern* a esso pertinenti.



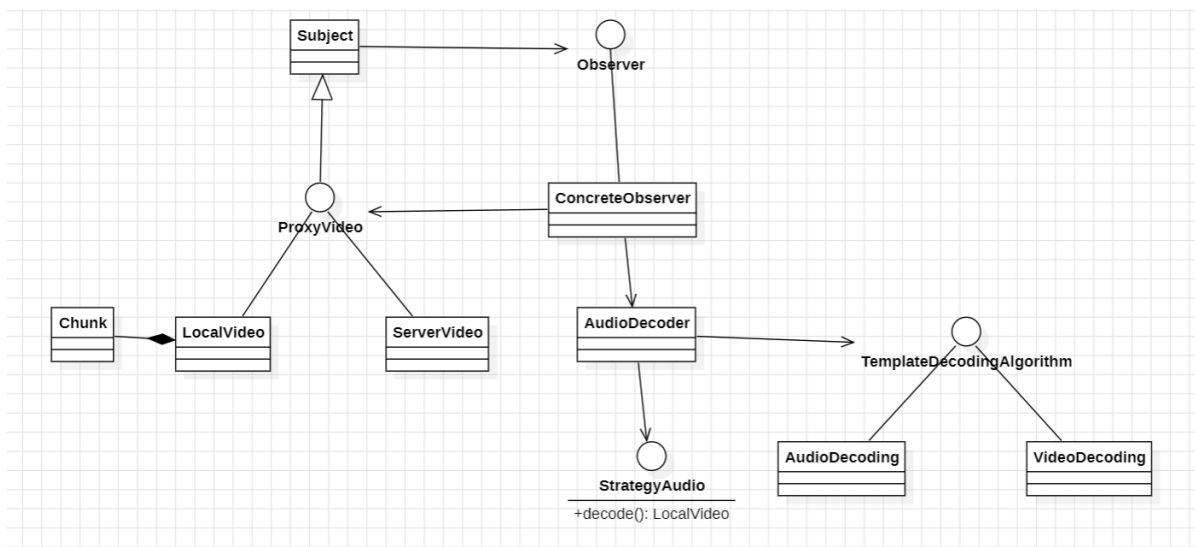


Esercizio 2 (7 punti)

Descrizione

Netmus era un sistema molto avanzato per il suo tempo dal punto di vista architetturale. Per limitare l'utilizzo di risorse, anticipando i tempi, faceva uso della *reactive programming*. In particolare, la riproduzione di un brano utilizzando YouTube avveniva attraverso un *listener* sulla disponibilità di nuove informazioni provenienti dal sito di streaming video. Ogni video veniva suddiviso in *chunk*, e la disponibilità di un nuovo *chunk* immediatamente notificata al riproduttore musicale. Questo, avvalendosi di un algoritmo di decodifica audio/video, separava i due canali, restituendo unicamente quello audio. L'algoritmo era in continuo miglioramento, quindi la struttura pensata per la decodifica permetteva la sua estensione in modo agevole. Per semplicità, un brano musicale era comunque visto come un file completo localmente, contenendo l'informazione audio, il titolo e l'immagine della copertina dell'album associato.

Si modelli tale sistema mediante un diagramma delle classi e i *design pattern* a esso pertinenti.

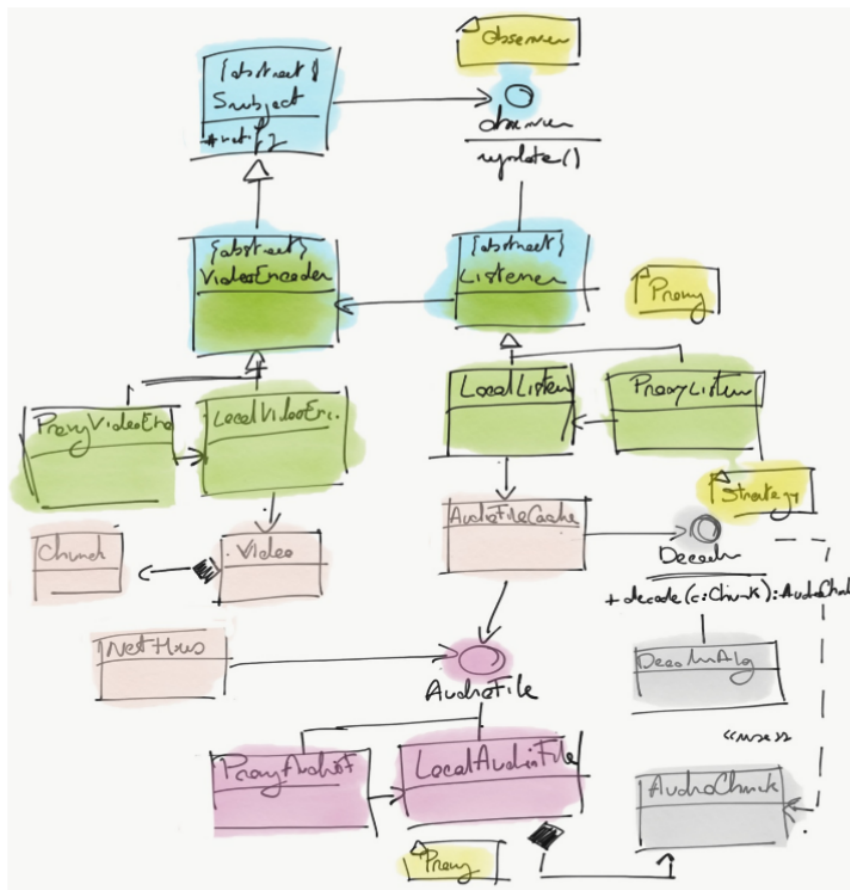


Soluzione

I pattern individuati sono i seguenti:

- Observer
- Remote proxy
- Virtual proxy
- Strategy

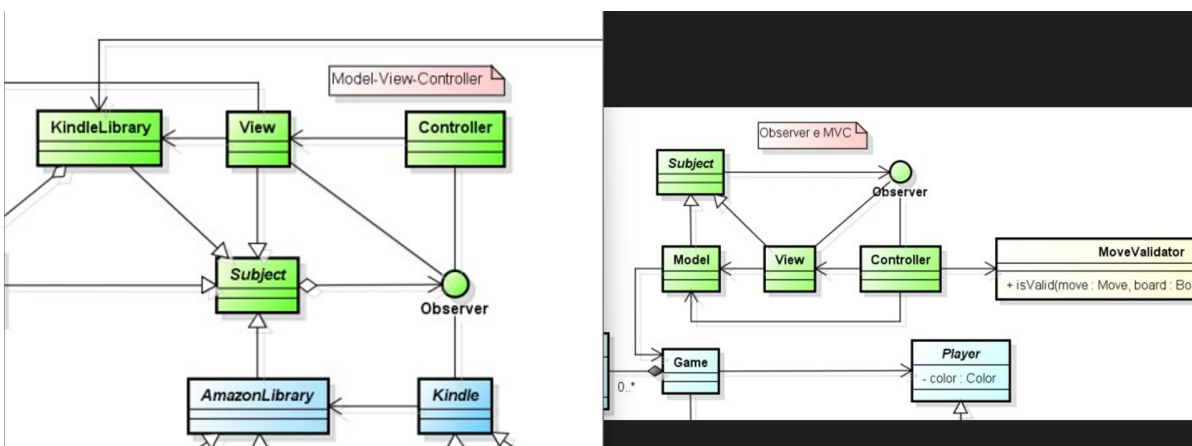
Il diagramma delle classi è il seguente.



Domanda 1/3 (punti 6/30)

Il Sudoku è uno dei giochi enigmistici più popolari al mondo. Molti sono i risolutori *online* di Sudoku, ma pochi di essi, dato uno schema iniziale, consentono al giocatore di risolverlo passo per passo con l'aiuto di suggerimenti diretti. Un'azienda giapponese sta sviluppando una nuova applicazione *desktop* proprio con quelle caratteristiche. L'applicazione presenta una vista interattiva nella quale il giocatore può posizionare i numeri di partenza dello schema. Da quel punto in avanti, il giocatore può richiedere il prossimo suggerimento e fare la propria mossa. L'architettura dell'applicazione adotta il *pattern* MVC. L'algoritmo di calcolo della prossima mossa, elemento centrale del sistema, è oggetto di miglioramenti continui, e di esso esistono più versioni. Una di esse richiede la mossa a un servizio remoto esterno, adattando la chiamata interna all'interfaccia esposta da quel servizio, e accedendovi come se fosse locale all'applicazione *desktop*. Una volta ricevuta, la mossa suggerita viene visualizzata opportunamente sullo schermo.

Si modelli tale sistema mediante un diagramma delle classi, comprensivo dei *design pattern* a esso pertinenti.



Il Sudoku è uno dei giochi enigmistici più popolari al mondo. Molti sono i risolutori *online* di Sudoku, ma pochi di essi, dato uno schema iniziale, consentono al giocatore di risolverlo passo per passo con l'aiuto di suggerimenti diretti. Un'azienda giapponese sta sviluppando una nuova applicazione *desktop* proprio con quelle caratteristiche. L'applicazione presenta una vista interattiva nella quale il giocatore può posizionare i numeri di partenza dello schema. Da quel punto in avanti, il giocatore può richiedere il prossimo suggerimento e fare la propria mossa. L'architettura dell'applicazione adotta il *pattern* MVC. L'algoritmo di calcolo della prossima mossa, elemento centrale del sistema, è oggetto di miglioramenti continui, e di esso esistono più versioni. Una di esse richiede la mossa a un servizio remoto esterno, adattando la chiamata interna all'interfaccia esposta da quel servizio, e accedendovi come se fosse locale all'applicazione *desktop*. Una volta ricevuta, la mossa suggerita viene visualizzata opportunamente sullo schermo.

