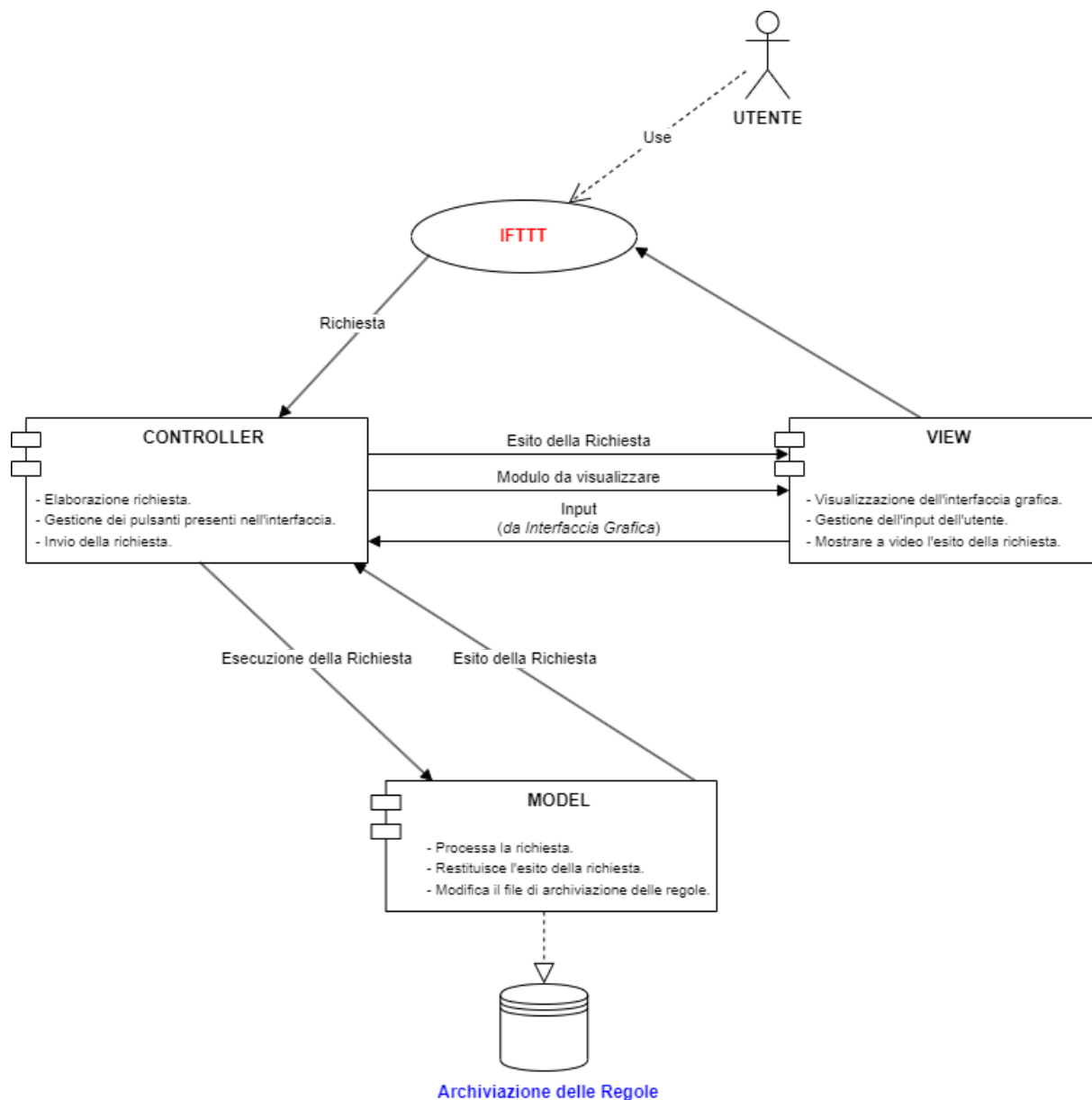


ARCHITETTURA DEL SOFTWARE



Per l'implementazione dell'applicativo abbiamo scelto di utilizzare l'architettura MVC (Model-View-Controller). Si tratta di un design pattern ampiamente utilizzato per organizzare e separare le responsabilità all'interno di un'applicazione. Migliora la modularità e la manutenibilità del codice.

Il **Model** rappresenta i dati e la logica di business dell'applicazione. È responsabile della gestione, dell'aggiornamento e dell'accesso ai dati. Il Model non è coinvolto direttamente nella gestione dell'interfaccia utente, ma si concentra sulla manipolazione dei dati sottostanti.

Il **View** è responsabile della presentazione dei dati all'utente e della gestione dell'interfaccia utente. Estrae i dati dal Model e li visualizza in un formato comprensibile per l'utente.

Esso non si occupa della logica di business o della gestione dei dati, ma si concentra sulla presentazione delle informazioni.

Il **Controller** funge da intermediario tra il Model ed il View. Gestisce gli input dell'utente, interpreta le azioni richieste e coordina le interazioni tra Model e View. Si occupa della logica di controllo e delle decisioni basate sugli input utente. Quando un utente interagisce con l'interfaccia utente, il Controller decide come gestire l'input, potenzialmente aggiornando il Model e notificando al View eventuali cambiamenti.

Con il termine “Richiesta” si intende tutto ciò che l’utente può fare, ovvero la creazione, la modifica e l’eliminazione di una regola, la sua attivazione o disattivazione, la creazione, la modifica e l’eliminazione di un contatore, eseguire varie operazioni su un file, come l’aggiunta di una stringa, la copia, lo spostamento e l’eliminazione.

Invece, la modifica del file di archiviazione delle regole comprende l’aggiunta di una nuova regola, la modifica o l’eliminazione di una regola già esistente.