

# TECNOLOGIE INFORMATICHE PER IL WEB

Carlo Sgaravatti – 10660072

Esercizio 3 – Gestione documenti

# Analisi requisiti per DB

Un'applicazione web consente la gestione di **cartelle**, **sottocartelle** e **documenti** online. L'applicazione supporta registrazione e login di **utenti** mediante una pagina pubblica con opportune form. La registrazione controlla l'unicità dello **username**. Una **cartella** ha un **proprietario**, un **nome** e una **data di creazione** e **può contenere (solo) sottocartelle**. Una **sottocartella** **può contenere (solo) dei documenti**. Un **documento** ha un **proprietario**, **nome**, una **data di creazione**, un **sommario** e un **tipo**. Quando l'utente accede all'applicazione appare una HOME PAGE che contiene un albero delle proprie cartelle e delle sottocartelle.

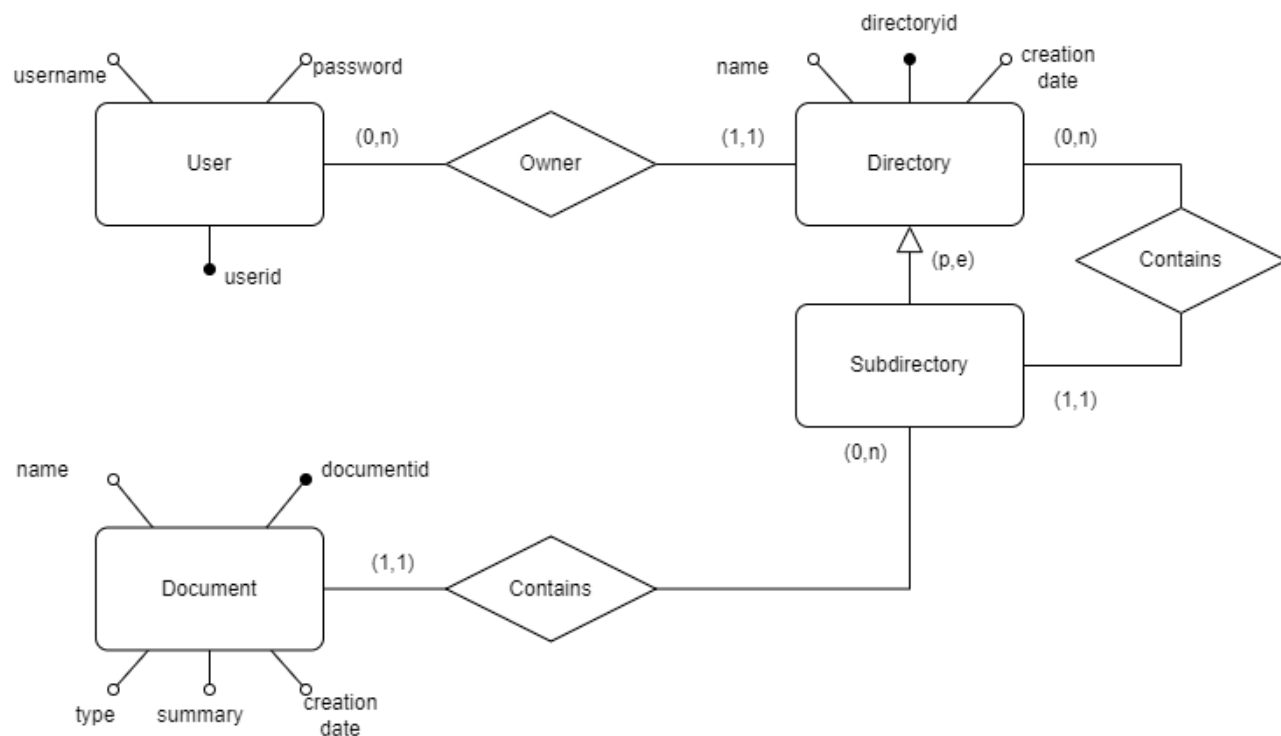
Nell'HOME PAGE l'utente può selezionare una sottocartella e accedere a una pagina DOCUMENTI che mostra l'elenco dei documenti di una sottocartella. Ogni documento in elenco ha due link: *accedi* e *sposta*. Quando l'utente seleziona il link *accedi*, appare una pagina DOCUMENTO che mostra tutti i dati del documento selezionato. Quando l'utente seleziona il link *sposta*, appare la HOME PAGE con l'albero delle cartelle e delle sottocartelle; in questo caso la pagina mostra il messaggio "*Stai spostando il documento X dalla sottocartella Y. Scegli la sottocartella di destinazione*", la sottocartella a cui appartiene il documento da spostare NON è selezionabile e il suo nome è evidenziato (per esempio con un colore diverso). Quando l'utente seleziona la sottocartella di destinazione, il documento è spostato dalla sottocartella di origine a quella di destinazione e appare la pagina DOCUMENTI che mostra il contenuto aggiornato della sottocartella di destinazione. Ogni pagina, tranne la HOME PAGE, contiene un collegamento per tornare alla pagina precedente. L'applicazione consente il logout dell'utente. Una pagina GESTIONE CONTENUTI raggiungibile dalla HOME PAGE permette all'utente di creare una cartella, una sottocartella di una cartella esistente e un documento all'interno di una sottocartella.

**Entities** - **relationships** - **attributes**

# Diagramma E/R

Ipotesi aggiuntive fatte:

- Un documento può essere contenuto solo in una sottocartella
- Un documento può essere contenuto solo in una sottocartella che è a sua volta contenuta in una cartella avente lo stesso proprietario del documento



# Modello relazionale

```
CREATE TABLE `directory` (  
  `directoryid` int NOT NULL AUTO_INCREMENT,  
  `name` varchar(50) NOT NULL,  
  `creation_date` date NOT NULL,  
  `is_subdirectory` tinyint NOT NULL DEFAULT '0',  
  `father_directory` int DEFAULT NULL,  
  `user` int NOT NULL,  
  PRIMARY KEY (`directoryid`),  
  UNIQUE KEY `user_name_unique` (`name`,`user`,`father_directory`),  
  KEY `user_idx` (`user`),  
  KEY `father_idx` (`father_directory`),  
  CONSTRAINT `father` FOREIGN KEY (`father_directory`) REFERENCES `directory` (`directoryid`) ON DELETE CASCADE ON UPDATE CASCADE,  
  CONSTRAINT `user` FOREIGN KEY (`user`) REFERENCES `user` (`userid`) ON DELETE CASCADE ON UPDATE CASCADE );
```

```
CREATE TABLE `user` (  
  `userid` int NOT NULL AUTO_INCREMENT,  
  `username` varchar(30) NOT NULL,  
  `password` varchar(20) NOT NULL,  
  `email` varchar(50) NOT NULL,  
  PRIMARY KEY (`userid`),  
  UNIQUE KEY `username_UNIQUE` (`username`));
```

```
CREATE TABLE `document` (  
  `documentid` int NOT NULL AUTO_INCREMENT,  
  `name` varchar(50) NOT NULL,  
  `creation_date` date NOT NULL,  
  `summary` text NOT NULL,  
  `subdirectory` int NOT NULL,  
  `type` varchar(20) NOT NULL DEFAULT '.txt',  
  PRIMARY KEY (`documentid`),  
  UNIQUE KEY `name_subdirectory_unique` (`name`, `subdirectory`),  
  KEY `subdirectory_idx` (`subdirectory`),  
  CONSTRAINT `subdirectory` FOREIGN KEY (`subdirectory`) REFERENCES `directory` (`directoryid`) ON DELETE CASCADE ON UPDATE CASCADE);
```

VERSIONE PURE HTML

# Analisi requisiti applicazione – Pure HTML

Un'applicazione web consente la gestione di cartelle, sottocartelle e documenti online. L'applicazione supporta **registrazione** e **login** di utenti mediante una **pagina pubblica** con **opportune form**. La **registrazione** controlla l'unicità dello username. Una cartella ha un proprietario, un nome e una data di creazione e può contenere (solo) sottocartelle. Una sottocartella può contenere (solo) dei documenti. Un documento ha un proprietario, nome, una data di creazione, un sommario e un tipo. Quando **l'utente accede** all'applicazione appare una **HOME PAGE** che contiene un **albero delle proprie cartelle e delle sottocartelle**.

Nell'HOME PAGE l'utente può selezionare una sottocartella e accedere a una pagina **DOCUMENTI** che mostra l'**elenco dei documenti di una sottocartella**. Ogni documento in elenco ha due link: **accedi** e **sposta**. Quando l'utente **seleziona il link accedi**, appare una pagina **DOCUMENTO** che mostra **tutti i dati del documento selezionato**. Quando l'utente **seleziona il link sposta**, appare la HOME PAGE con l'albero delle cartelle e delle sottocartelle; in questo caso la pagina mostra il **messaggio "Stai spostando il documento X dalla sottocartella Y. Scegli la sottocartella di destinazione"**, la sottocartella a cui appartiene il documento da spostare NON è selezionabile e **il suo nome è evidenziato** (per esempio con un colore diverso). Quando l'utente **seleziona la sottocartella di destinazione**, il **documento è spostato dalla sottocartella di origine a quella di destinazione** e appare la pagina DOCUMENTI che mostra il contenuto aggiornato della sottocartella di destinazione. Ogni pagina, tranne la HOME PAGE, contiene un **collegamento per tornare alla pagina precedente**. L'applicazione consente il **logout** dell'utente. Una pagina **GESTIONE CONTENUTI** raggiungibile dalla **HOME PAGE** permette all'utente di **creare una cartella, una sottocartella di una cartella esistente e un documento all'interno di una sottocartella**.

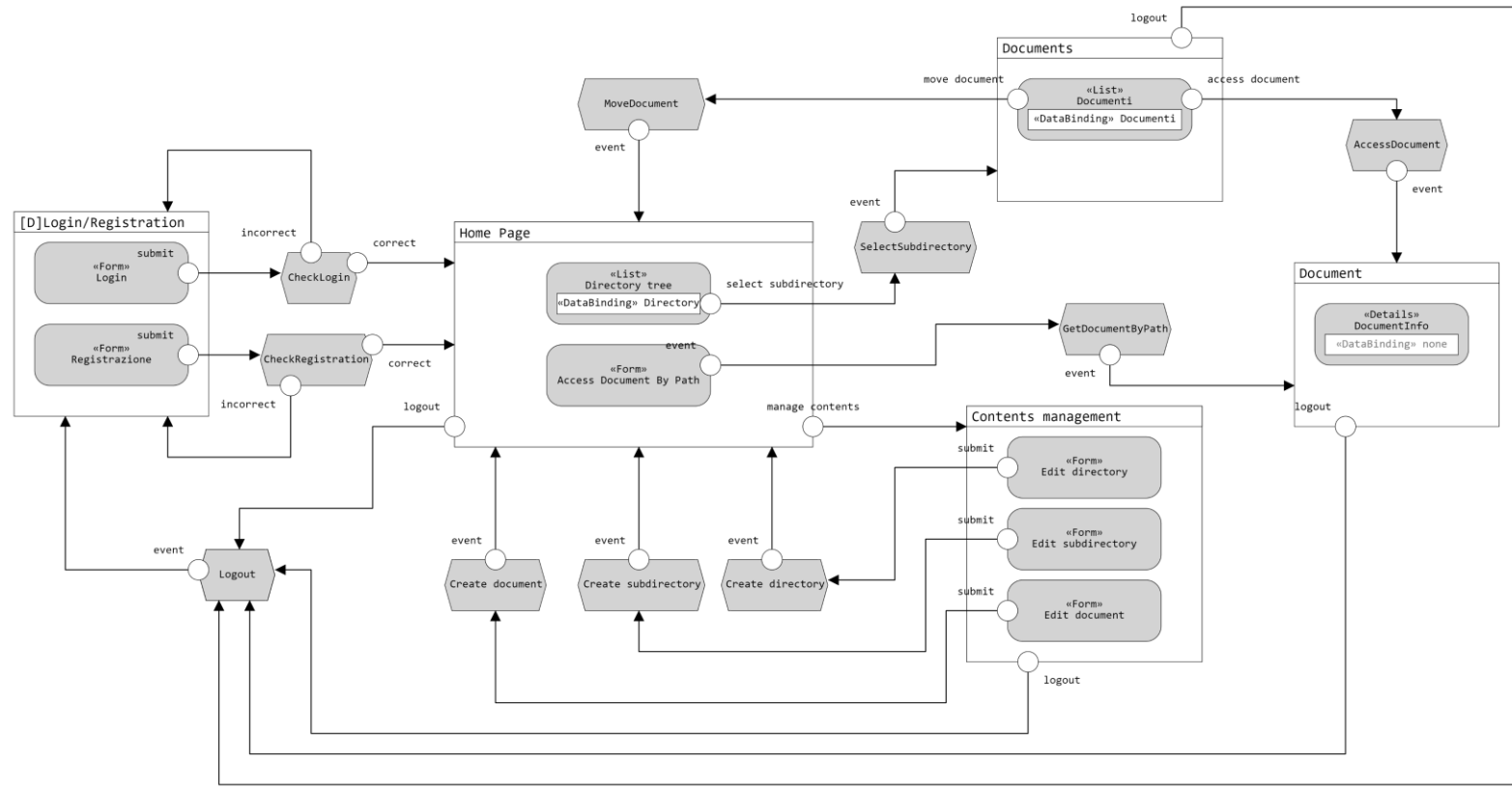
**Page(views)** - **view components** - **events** - **actions**

# Completamento delle specifiche

- In tutte le pagine, tranne la pagina pubblica di default, è presente un collegamento per fare il logout
- In tutte le pagine, tranne la HOME PAGE e la pagina di default, è presente un collegamento per tornare alla home (oltre a quello per tornare alla pagina precedente)
- Quando viene creato un nuovo contenuto (cartella, sottocartella o documento), l'applicazione ritorna alla HOME PAGE
- La pagina GESTIONE CONTENUTI è raggiungibile anche quando si sta spostando un documento, tuttavia questo ne annulla l'operazione
- Si è permesso all'utente di selezionare la lingua dell'interfaccia utente (italiano o inglese) tramite dei collegamenti posti in ogni pagina (tranne la pagina di default)
- L'accesso al documento è stato reso possibile anche dalla HOME PAGE inserendo in un'opportuna form il percorso del documento (del tipo: Cartella/Sottocartella/Documento)



# IFML



# Componenti (1/2)

## Model Objects (Beans)

- User
- Directory
- Subdirectory (extends Directory)
- Document

## Views (Thymeleaf Templates)

- index.html
- home.html
- documents.html
- document.html
- contentCreation.html
- navbar.html (fragment)

## Data Access Objects

- UserDao
  - createUser
  - checkCredentials
- DirectoryDao
  - createDirectory
  - createSubdirectory
  - findDirectoryById
  - findDirectoriesAndSubdirectoriesOfUser
  - getAllDirectoriesOfUser
  - getAllSubdirectoriesOfUser
  - findDirectoryByNameAndUser
  - findSubdirectoryAndFatherDirectory
- DocumentDao
  - createDocument
  - findDocument
  - findDocumentAndSubdirectory
  - moveDocument
  - findAllDocumentsOfSubdirectory
  - findDocumentFromNameAndSubdirectory

# Componenti (2/2)

## Controllers (Servlets)

- AccessDocument
- ChangeLanguage
- CreateDirectory
- CreateDocument
- CreateSubdirectory
- GetDocumentByPath
- GoToHome
- GoToPreviousPage
- Login
- Logout
- MoveDocument
- Registration
- SelectSubdirectory
- Welcome

## Filters

- LoginFilter
- AddingDateFilter
- PreviousPageFilter

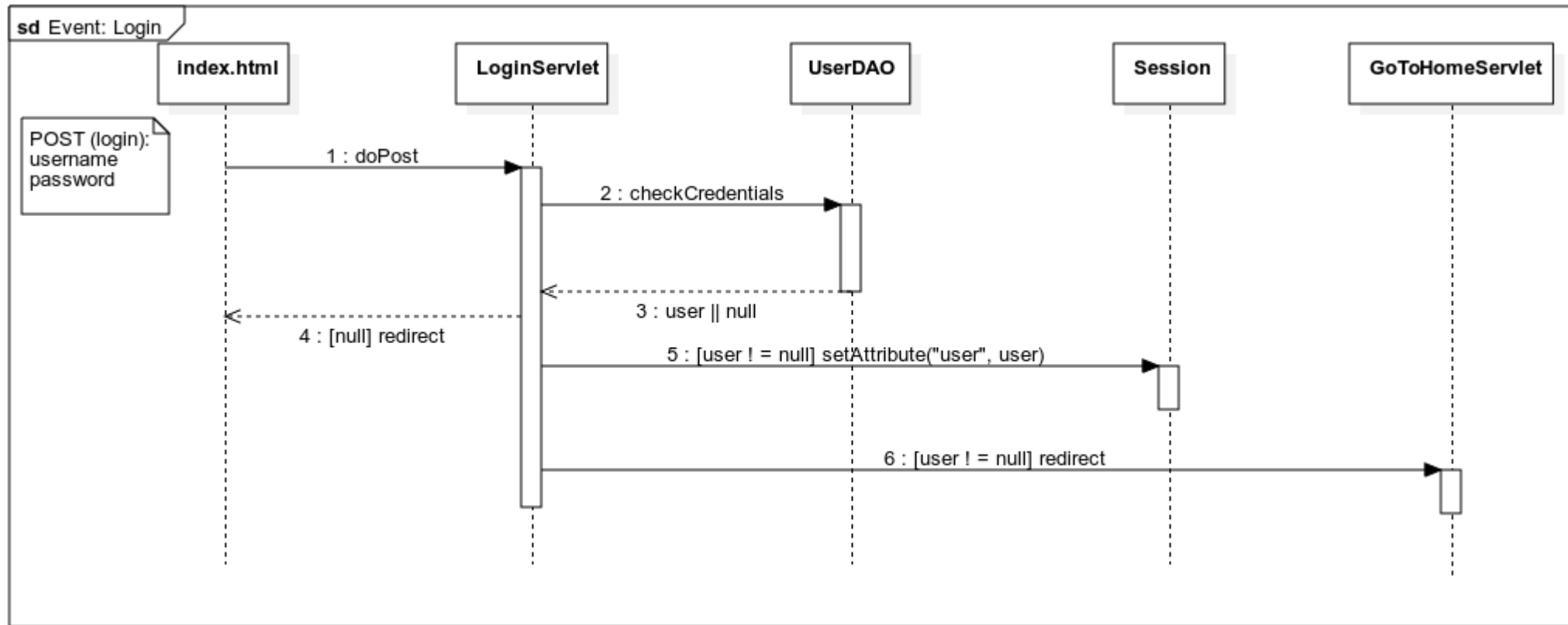
## Utils

- ConnectionCreator
- ThymeleafTemplateEngineCreator
- Pair<T, V>

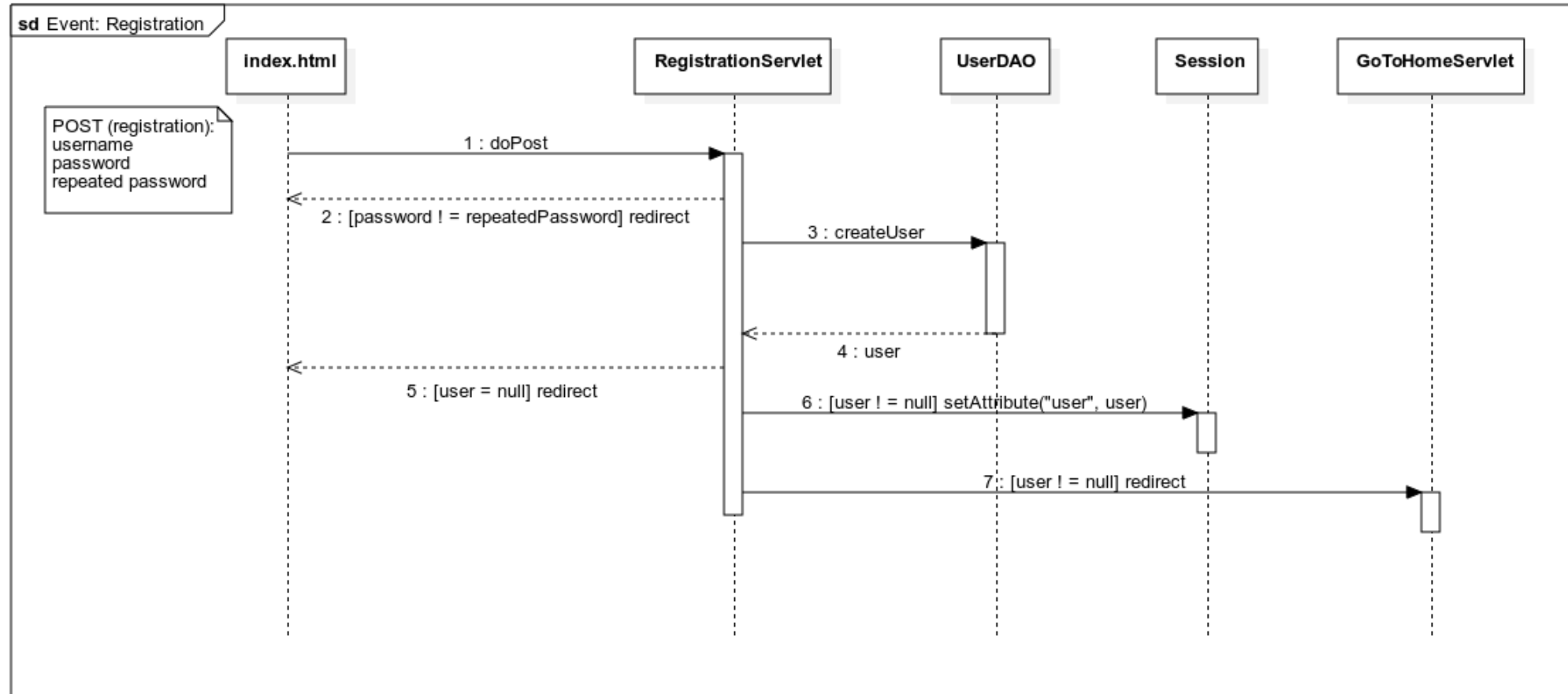
# Sequence Diagrams

- I seguenti sequence diagram mostrano alcuni degli eventi più significativi
- I diagrammi di alcuni eventi non sono stati riportati perché molto simili a quelli di altri eventi (ad esempio la creazione di una sottocartella e di una cartella sono molto simili alla creazione di un documento)
- Alcuni dettagli (come la gestione di alcuni errori nel caso di invio di parametri scorretti da parte dell'utente) sono stati omessi per rendere i diagrammi più leggibili

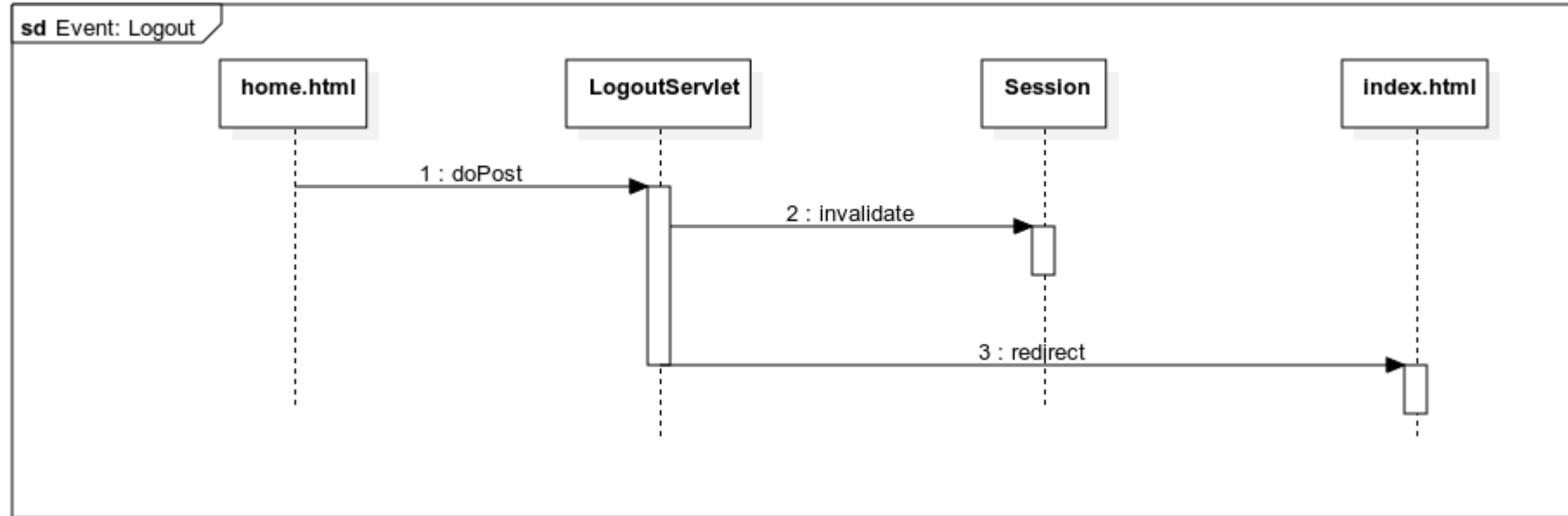
# Login



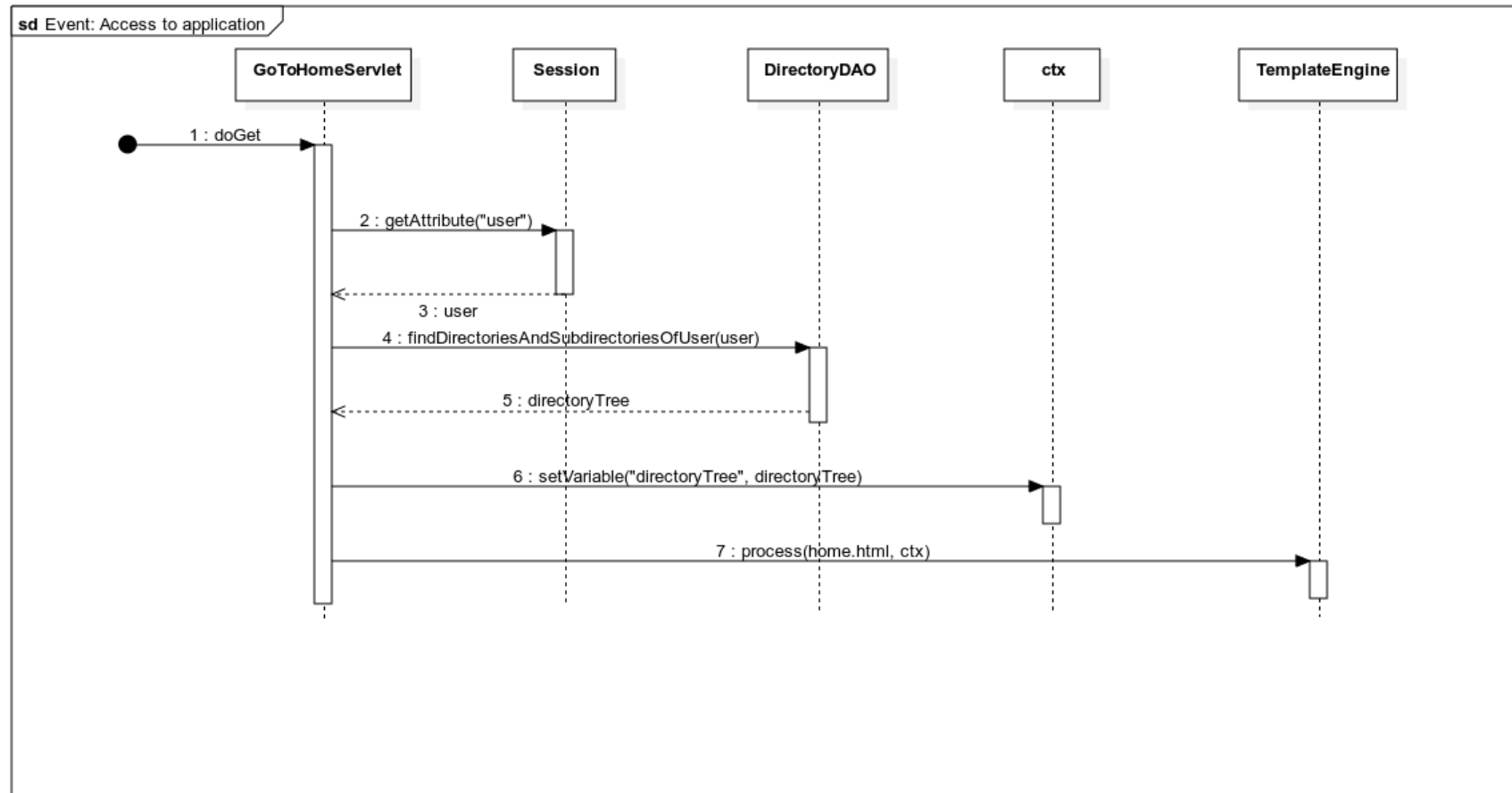
# Registrazione



# Logout

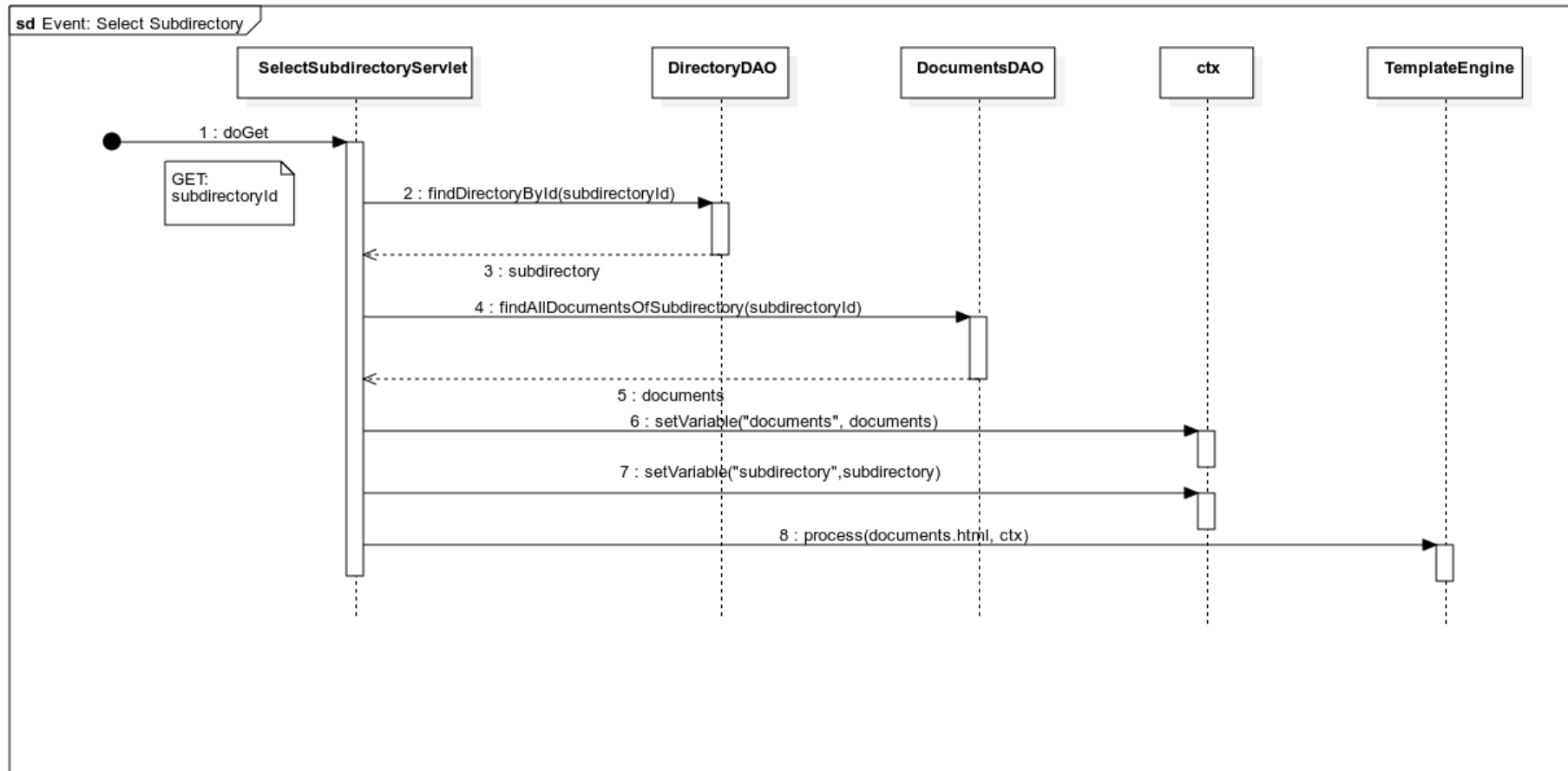


# Accesso all'applicazione

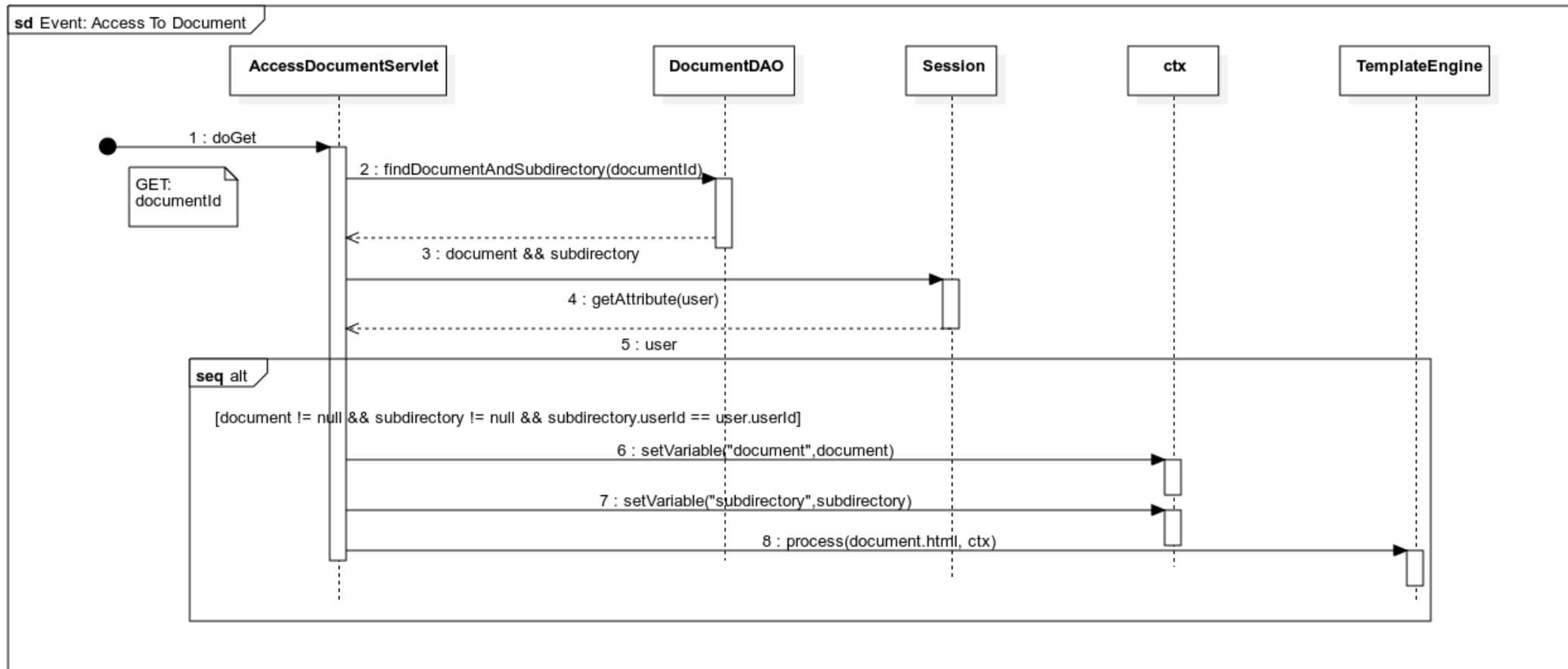




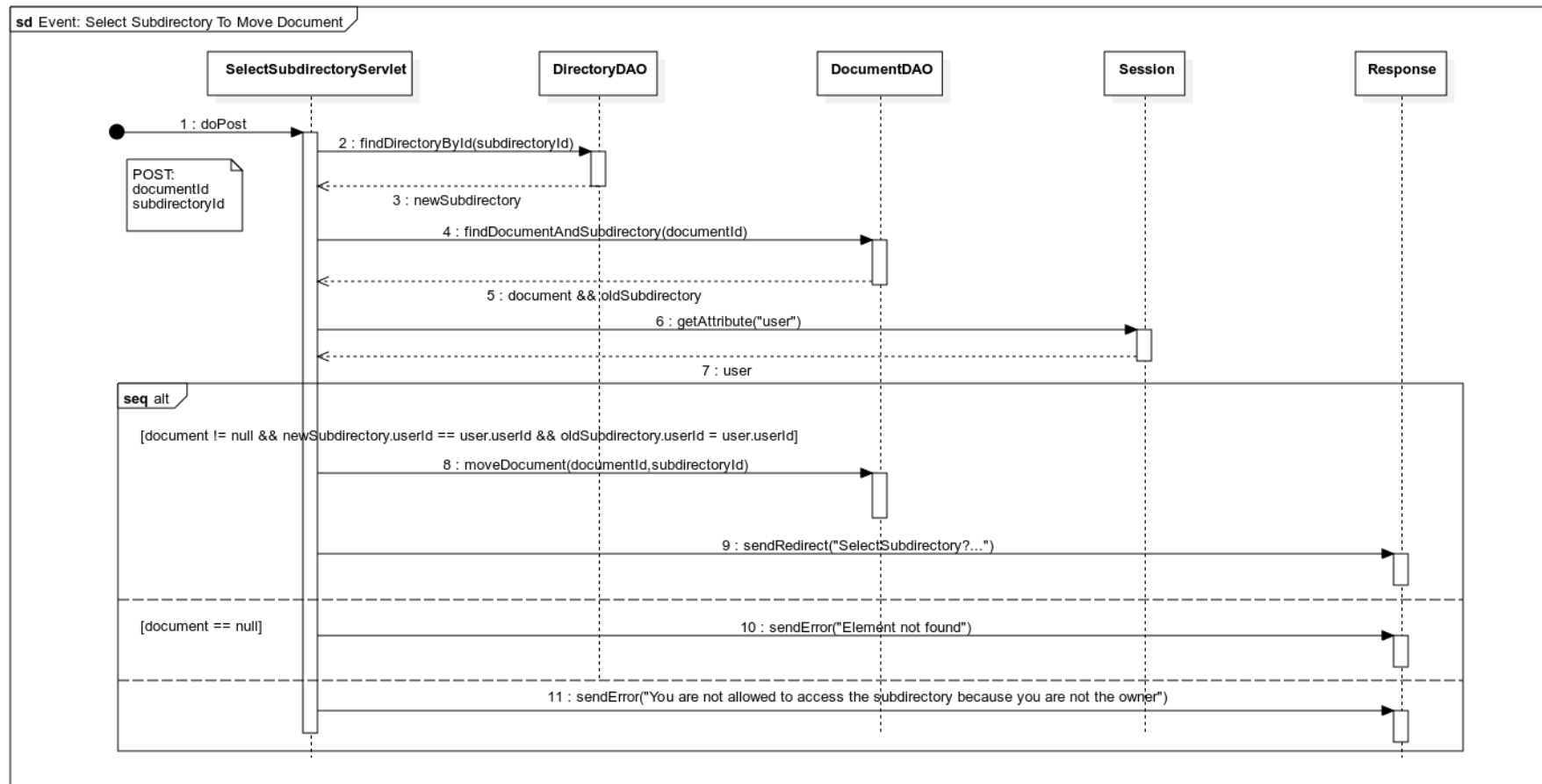
# Selezione di una sottocartella



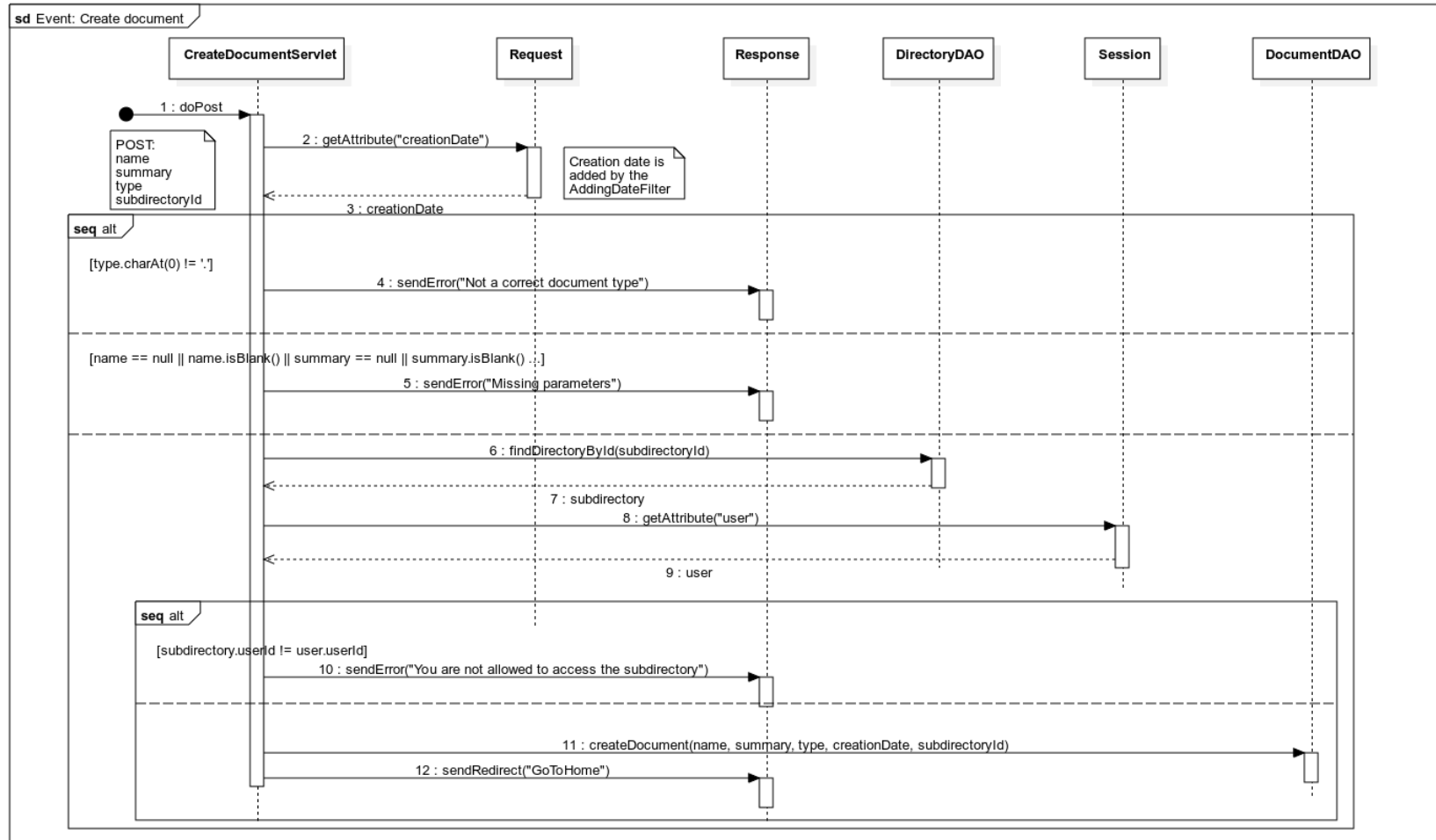
# Accesso ad un documento



# Spostamento di un documento (dalla home)



# Creazione di un documento



VERSIONE RIA

# Analisi requisiti applicazione - RIA

Si realizzi un'applicazione client server web che modifica le specifiche precedenti come segue:

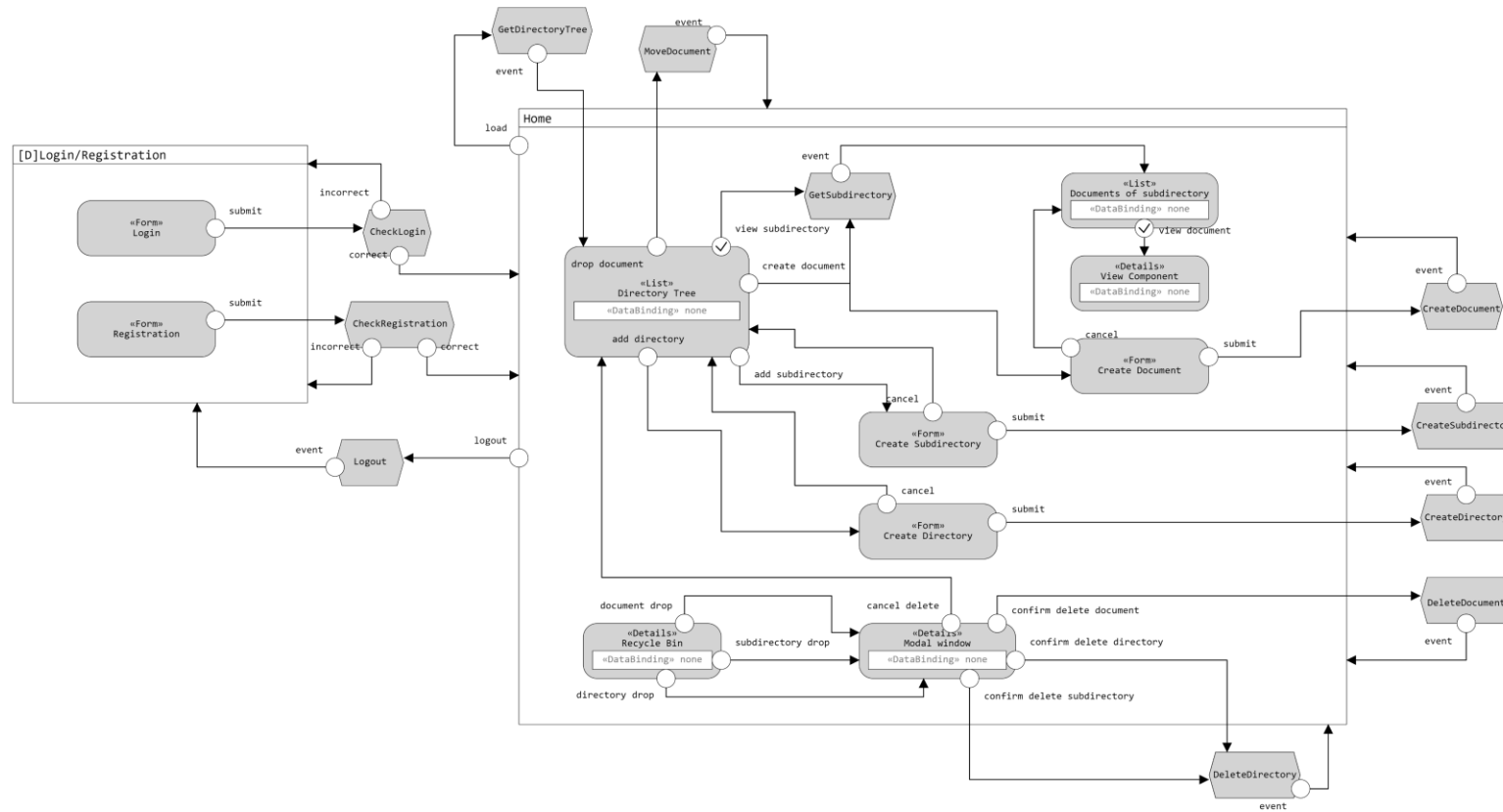
- La registrazione controlla la validità sintattica dell'indirizzo di email e l'uguaglianza tra i campi "password" e "ripeti password", anche a lato client.
- Dopo il login, l'intera applicazione è realizzata con un'unica pagina
- Ogni **interazione dell'utente** è gestita senza ricaricare completamente la pagina, ma produce **l'invocazione asincrona del server** e l'eventuale modifica del contenuto da aggiornare a seguito dell'evento.
- **Errori a lato server** devono essere **segnalati** mediante un **messaggio di allerta** all'interno della pagina.
- La funzione di spostamento di un documento è realizzata mediante drag and drop.
- La funzione di **creazione di una sottocartella** è realizzata nella **pagina HOME** mediante un **bottone AGGIUNGI SOTTOCARTELLA** posto di fianco ad ogni cartella. La **pressione del bottone** fa apparire un **campo di input** per l'inserimento del nome della sottocartella.
- La funzione di **creazione di un documento** è realizzata nella pagina HOME mediante un **bottone AGGIUNGI DOCUMENTO** posto di fianco ad ogni sottocartella. La **pressione del bottone** fa apparire una **form di input** per l'inserimento dei dati del documento.
- Si aggiunge una **cartella denominata "cestino"**. Il **drag & drop di un documento o di una cartella nel cestino** comporta la **cancellazione**. Prima di inviare il comando di cancellazione al server l'utente vede una **finestra modale di conferma** e può decidere se **annullare l'operazione** o **procedere**.

**Page(views)** - **view components** - **events** - **actions**

# Completamento delle specifiche

- Nella home è presente anche un bottone AGGIUNGI CARTELLA che permette di creare una nuova cartella
- La cancellazione di una sottocartella comporta l'eliminazione di tutti i documenti contenuti in essa
- Il drag and drop di una cartella nel cestino consente di eliminare una cartella, il che comporta l'eliminazione di tutte le sue sottocartelle (e di tutti i documenti contenuti nelle sottocartelle)
- Nella home è presente un bottone che permette di tornare indietro allo stato precedente della pagina
- La conferma o l'annullamento di un'operazione di cancellazione avviene tramite due bottoni all'interno della finestra modale
- I messaggi di errore (lato server o lato client) durante il login/registrazione vengono mostrati all'interno della pagina di login

# IFML





# Componenti lato server

## Controllers (Servlets)

- CreateDirectory
- CreateDocument
- CreateSubdirectory
- DeleteDocument
- DeleteDirectory
- GetDirectoryTree
- GetSubdirectory
- Login
- Logout
- MoveDocument
- Registration

## Filters

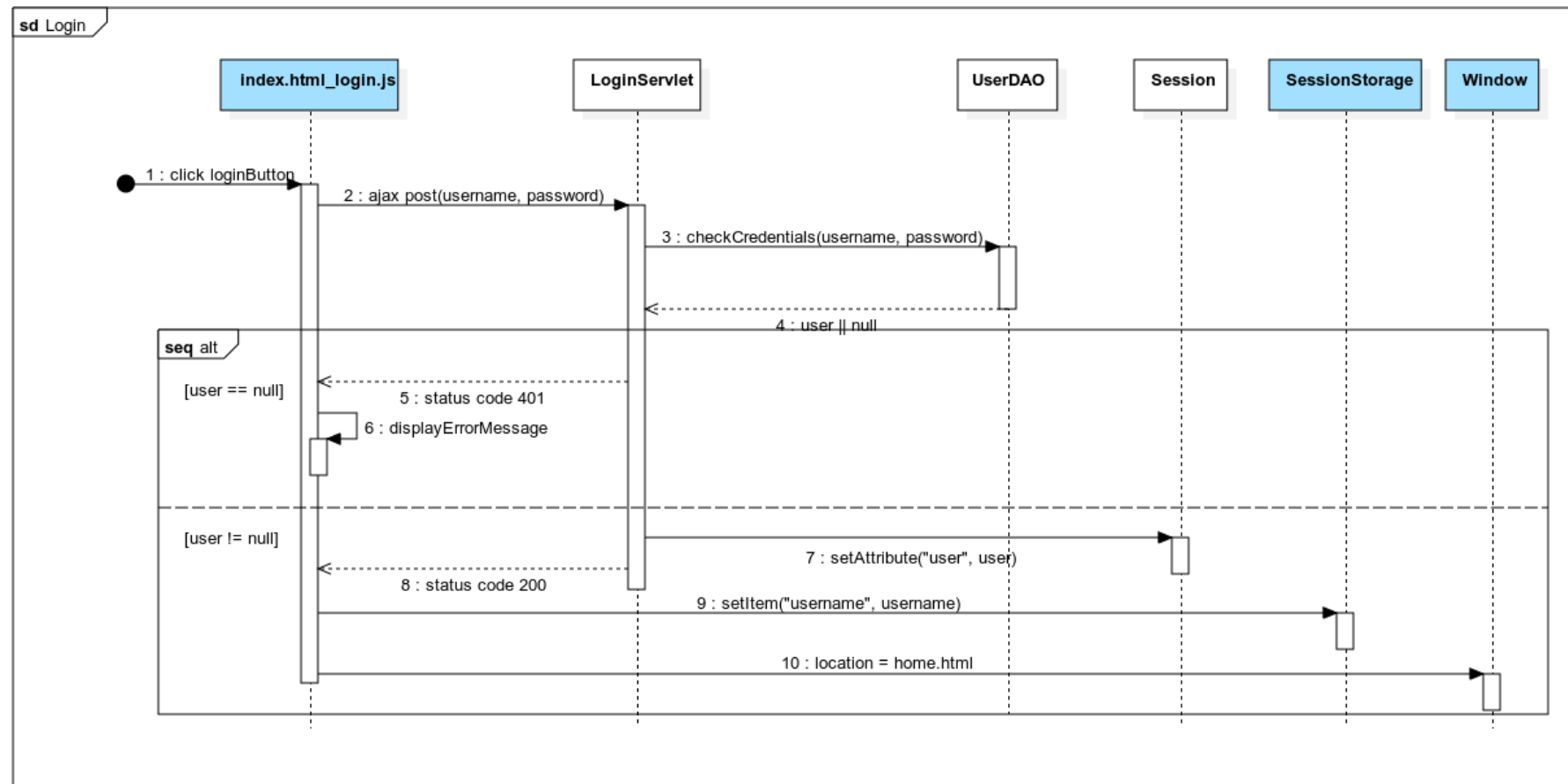
- LoginFilter
- AddingDateFilter

I Beans e i DAO sono gli stessi della versione pure HTML.

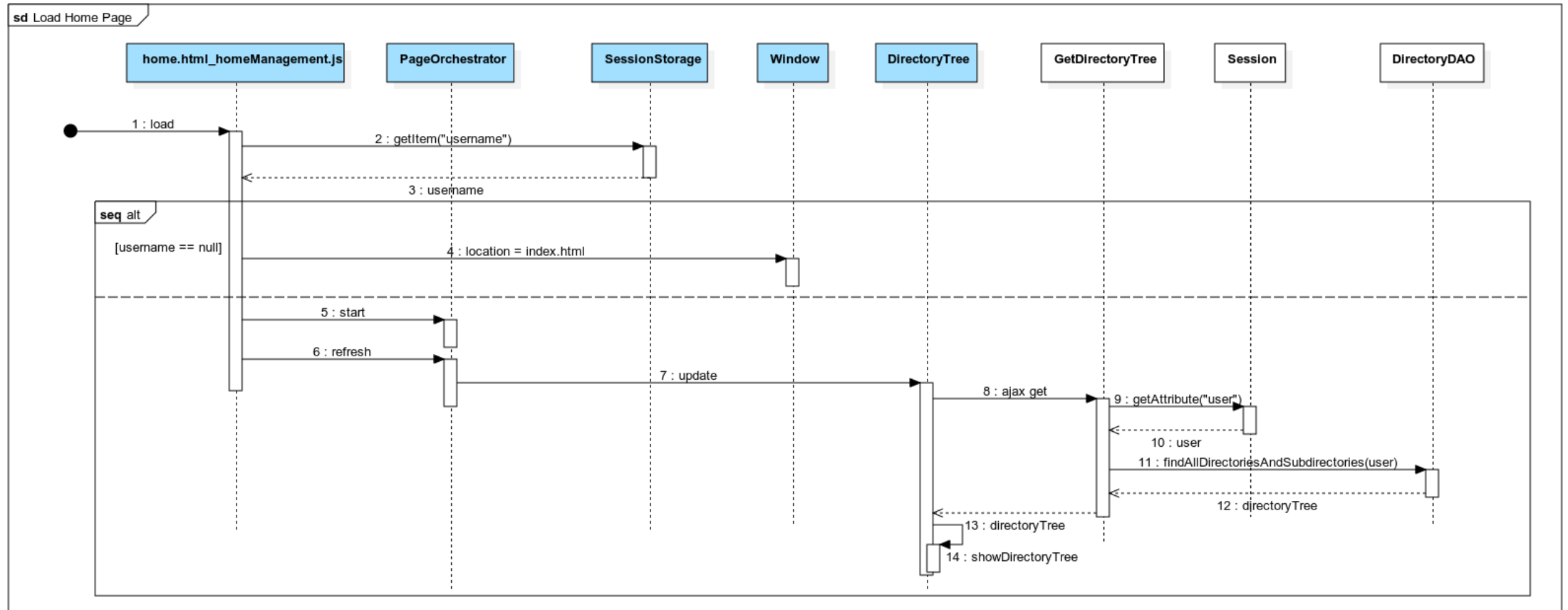
# Componenti lato client

- PageOrchestrator:
  - inizializza i componenti della pagina
  - ricarica la pagina o parti della pagina a seguito di modifiche
- MessageDisplayer:
  - mostra messaggi di errore all'interno della pagina
- FormsManager:
  - gestisce l'apparizione/la scomparsa di form in seguito a click di bottoni
  - gestisce gli eventi in seguito alla compilazione delle form
- ContentModifier:
  - invia le richieste asincrone al server nel caso di modifica dei dati (creazione o cancellazione di contenuti, spostamento di documenti)
- ModalDisplayer:
  - mostra la finestra modale di conferma in seguito al drag and drop nel cestino di un documento/cartella/sottocartella
- DragAndDropManager:
  - gestisce tutti gli eventi riguardanti il drag and drop di cartelle/sottocartelle/documenti
- SubdirectoryView:
  - aggiorna la parte di pagina che mostra il contenuto di una sottocartella (dopo averne richiesto al server in maniera asincrona l'elenco dei documenti)
- DirectoryTree:
  - mostra l'albero delle cartelle e sottocartelle dell'utente

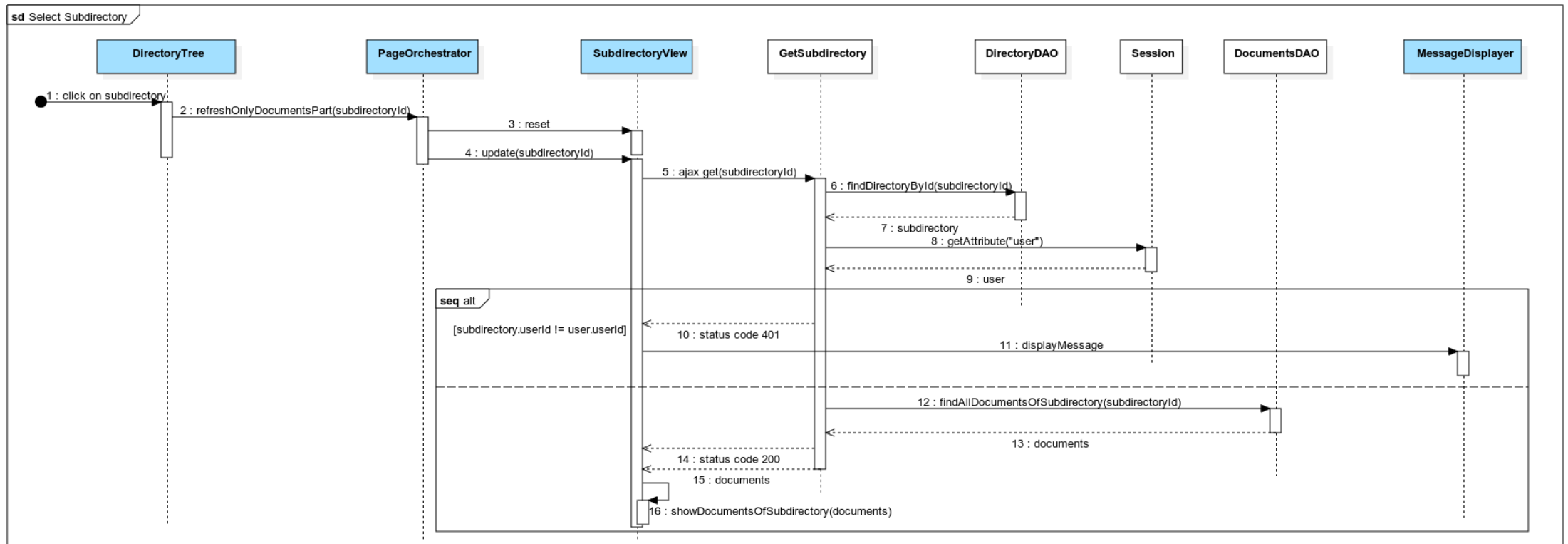
# Login



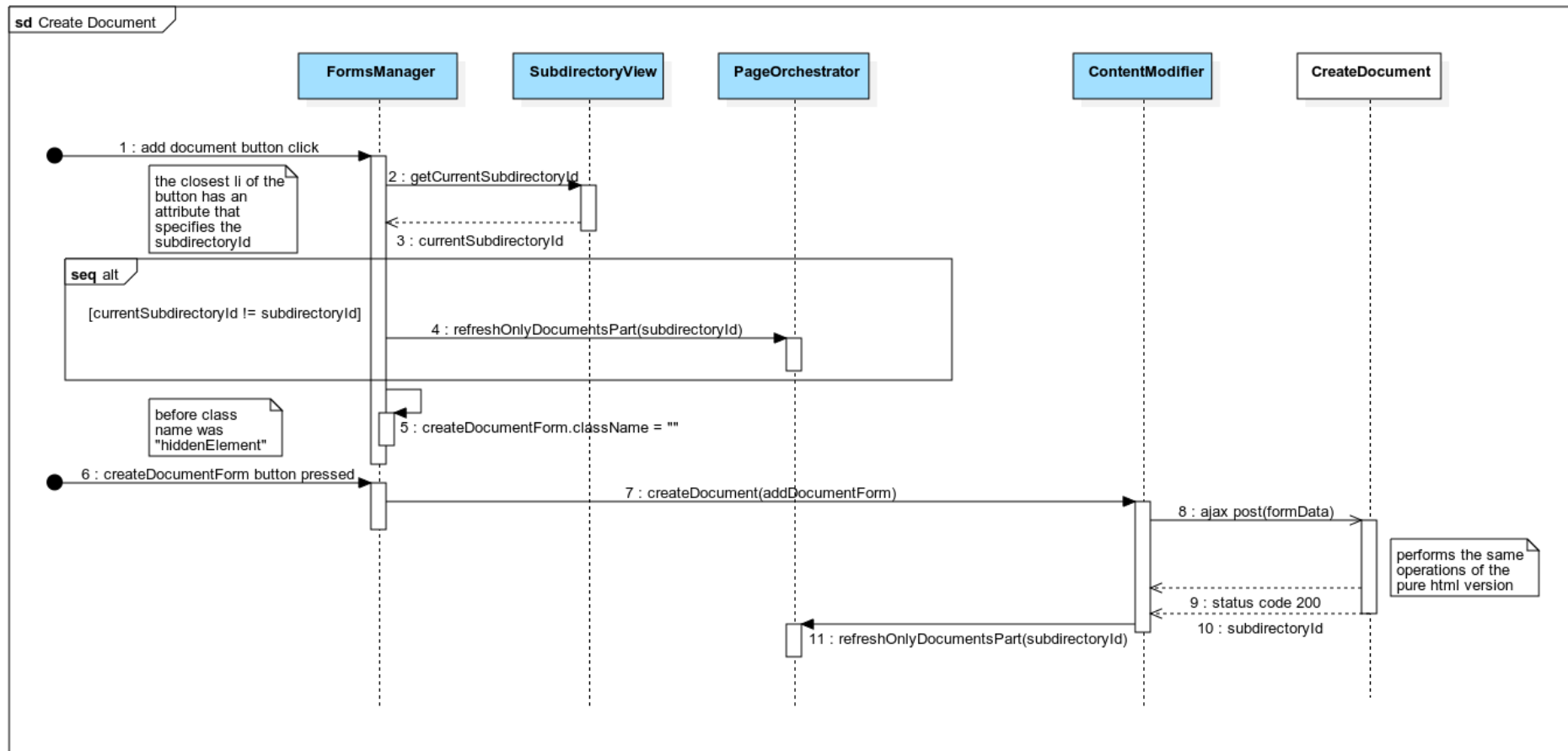
# Caricamento della home page



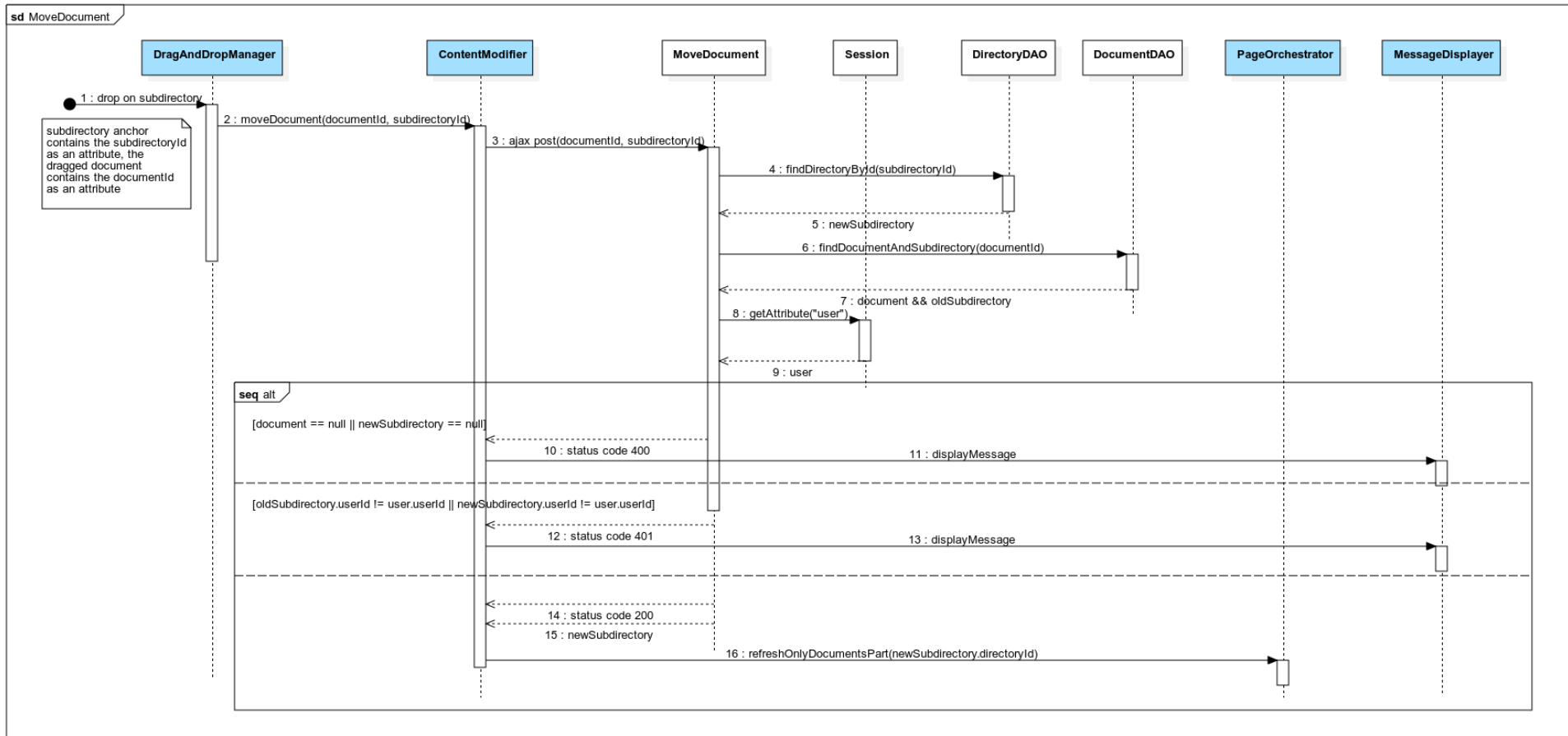
# Selezione di una sottocartella



# Creazione di un documento



# Spostamento di un documento



# Cancellazione di un documento

