

# Tecnologie informatiche per il web

## Progetto 1 - Galleria Immagini

Moretti Lorenzo - 10671362

Mormile Matteo - 10666565

17 Giugno, 2022

## Indice

<b>1</b>	<b>Versione PureHTML</b>	<b>3</b>
1.1	Progettazione della base di dati	3
1.1.1	Analisi del testo	3
1.1.2	Completamento delle specifiche della base di dati	3
1.1.3	Diagramma E/R della base di dati	3
1.1.4	Schema logico della base di dati	4
1.2	Definizione dei requisiti applicativi	5
1.2.1	Analisi del testo	5
1.2.2	Completamento delle specifiche applicative	5
1.2.3	Application design	6
1.2.4	Componenti del progetto	7
1.3	Sequence diagram	8
1.3.1	Evento di login	8
1.3.2	Visualizza HomePage	9
1.3.3	Creazione album	9
1.3.4	Caricamento immagine	10
1.3.5	Mostra immagini di un album	11
1.3.6	Mostra dettagli immagine di un album	12
1.3.7	Mostra blocco di immagini successivo/precedente	12
1.3.8	Creazione di un commento	13
1.3.9	Filter Login	13
<b>2</b>	<b>Versione RIA</b>	<b>14</b>
2.1	Definizione dei requisiti applicativi	14
2.1.1	Modifiche rispetto alla versione PureHTML	14
2.1.2	Componenti del progetto	14
2.1.3	Application design	15
2.2	Tabella eventi e azioni Client/Server side	17
2.3	Tabella eventi e controllori Client/Server side	18
2.4	Componenti del progetto Server Side	19
2.4.1	Model Object (Beans)	19
2.4.2	Data Access Object (Classes)	19

2.4.3	Controllers (Servlets).....	19
2.5	Componenti del progetto Client Side.....	20
2.5.1	View & view components .....	20
2.6	Sequence diagram.....	21
2.6.1	Evento di login.....	21
2.6.2	Caricamento della Home .....	22
2.6.3	Creazione di un album .....	23
2.6.4	Caricamento immagine .....	24
2.6.5	Mostra immagini di un album .....	25
2.6.6	Mostra blocco di immagini successivo/precedente.....	25
2.6.7	Mostra dettagli immagine di un album .....	26
2.6.8	Creazione di un commento.....	27
2.6.9	Riordinamento dei propri albums .....	28
2.6.10	Salvataggio ordine personalizzato albums.....	29
2.6.11	Logout.....	29

## 2 Versione RIA

### 2.1 Definizione dei requisiti applicativi

#### 2.1.1 Modifiche rispetto alla versione PureHTML

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

- Dopo il login dell'utente, 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.
- L'evento di visualizzazione del blocco precedente/successivo d'immagini di un album è gestito a lato client senza generare una richiesta al server.
- Quando l'utente passa con il mouse su una miniatura, l'applicazione mostra una finestra modale con tutte le informazioni dell'immagine, tra cui la stessa a grandezza naturale, i commenti eventualmente presenti e la form per inserire un commento.
- L'applicazione controlla anche a lato client che non si invii un commento vuoto.
- Errori a lato server devono essere segnalati mediante un messaggio di allerta all'interno della pagina.
- Si deve consentire all'utente di riordinare l'elenco dei propri album con un criterio diverso da quello di default (data decrescente). L'utente trascina il titolo di un album nell'elenco e lo colloca in una posizione diversa per realizzare l'ordinamento che desidera, senza invocare il server. Quando l'utente ha raggiunto l'ordinamento desiderato, clicca un bottone "salva ordinamento", per memorizzare la sequenza sul server. Ai successivi accessi, l'ordinamento personalizzato è usato al posto di quello di default.

Pages (views), View components, Events, Actions

#### 2.1.2 Componenti del progetto

##### Viste e componenti

###### Index.html (pagina di login)

- Form di login
- Form di registrazione

###### Home.html

- Ancora per il logout
- Ancora per visualizzare la lista degli album
- Ancora per creare album
- Ancora per caricare immagini
- Tabella per mostrare 5 immagini
- Bottoni per scorrere la galleria
- Form per postare un commento
- Dettaglio immagine selezionata

## Eventi e azioni

### Login/Creazione utente

- Verifica delle credenziali

### Hover dell'utente sul menù di creazione dell'immagine

- Mostra form upload immagine

### Hover dell'utente sul menù di creazione dell'album

- Mostra form creazione album

### Hover dell'utente sul menù per visualizzare l'elenco degli album

Mostra tabella contenente elenco degli album

### Hover sulla foto

- Mostra la foto a dimensioni naturali e i suoi dettagli

### Click sul bottone next

- Mostra le 5 immagini successive

### Click sul bottone prev

- Mostra le 5 immagini precedenti

### Click sul bottone carica

- Carica l'immagine

### Click sul bottone crea

- Crea il nuovo album

### Click sul bottone salva ordinamento

- Salva ordinamento della lista degli album

### Click sul bottone salva commento

- Salva il commento relativo alla foto

## 2.1.3 Application design

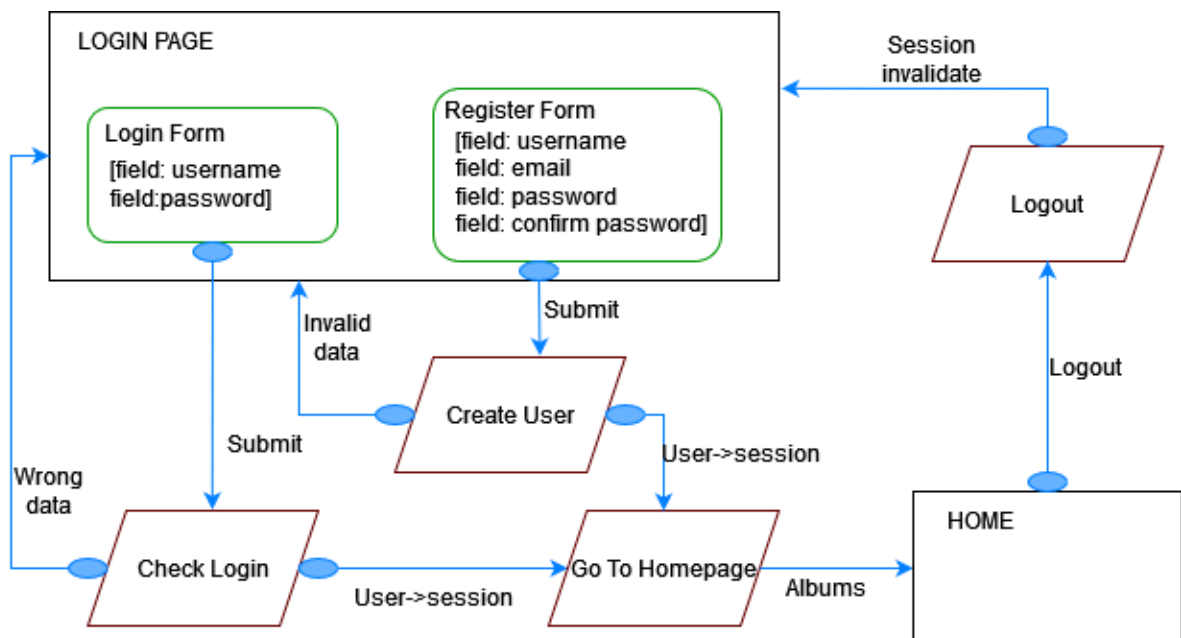


Figura 13: Application design della pagina di login

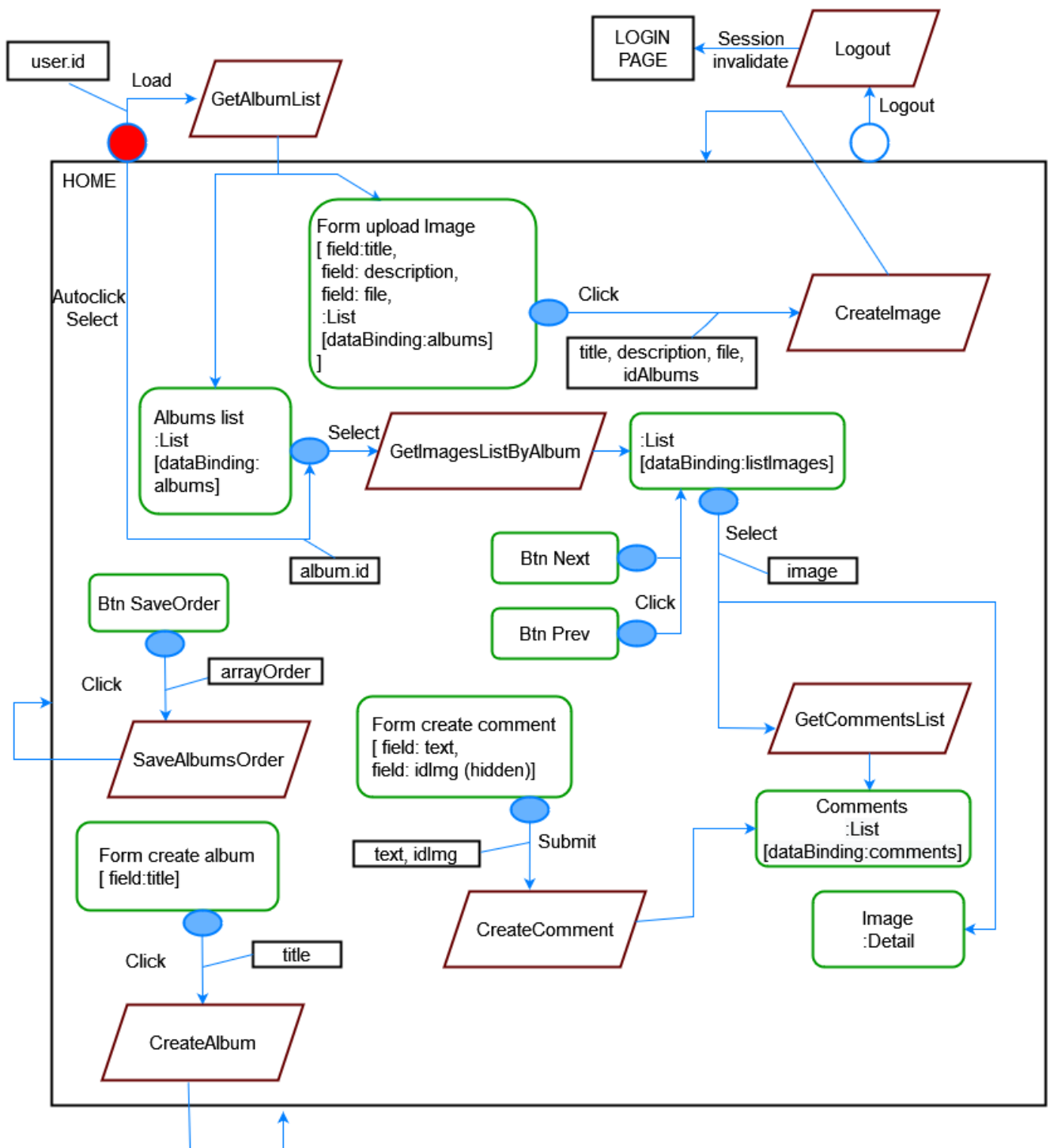


Figura 14: Application design della home page

## 2.2 Tabella eventi e azioni Client/Server side

Client Side		Server Side	
Evento	Azione	Evento	Azione
index->login form ->submit	Controllo dati	POST (username, password)	Controllo credenziali
Home->load	Aggiorna view con componenti	GET GET(idAlbum)	Estrazione della lista degli album e della lista delle immagini del primo in elenco
Home->elenco album ->hover	Mostra la lista degli album Aggiorna componenti visive	-	-
Home->creazione album ->hover	Mostra la form per inserire un nuovo album	-	-
Home->caricamento immagine->hover	Mostra la form per caricare una nuova foto	-	-
Home->logout	-	GET	Terminazione della sessione
Home->elenco album ->select	Aggiorna componente della vista che mostra la lista delle immagini	GET(idAlbum)	Recupero della lista delle immagini dell'album selezionato
Home->creazione album ->submit	Creo un nuovo album	POST(title)	Inserisco un nuovo album nel DB
Home->caricamento immagine->submit	Carico una nuova foto	POST (title,description,file,idAlbums)	Inserisco una nuova foto all'interno di uno o più album
Home->lista immagini ->immagine->hover	Aggiorno componenti visive mostrando immagini in dimensioni reali, dettagli e commenti	-	-
Home->btnNext/BtnPrev ->Click	Aggiorno componenti visive della lista delle immagini	-	-
Home->form commento ->submit	Creo un nuovo commento	POST(text,idImg)	Inserisco un nuovo commento nel DB

## 2.3 Tabella eventi e controllori Client/Server side

Client side		Server side	
Evento	Controllore	Evento	Controllore
Index->login form ->submit	Function make call	POST (username, password)	CheckLogin (Servlet)
Home->load	Function pageOrchestrator	GET GET(idAlbum)	GetAlbumList (Servlet) GetImagesListByAlbum (Servlet)
Home->elenco album ->hover	Hover CSS	-	-
Home-> elenco album ->select	Function AlbumsList.show Make call	GET(idAlbum)	GetImagesListByAlbum (Servlet)
Home->crea album ->hover	Hover CSS	-	-
Home->crea album ->submit	Function CreateAlbum Make call	POST(title)	CreateAlbum (Servlet)
Logout		GET	Logout (Servlet)
Home->Carica immagine ->hover	Hover CSS	-	-
Home->Carica immagine ->submit	Function UploadImage Make call	POST (title, file, description, idAlbums)	CreateImage (Servlet)
Home->Lista immagini ->hover	Function ImageDetails.update	GET(idImage)	GetCommentsList(Servlet)
Home->btnNext->Click	Function imagesList.update	-	-
Home->btnPrev->Click	Function imagesList.update	-	-
Home->Inserisci commento ->submit	Function ImageDetails. updateComments Make Call	POST(text, idImg)	CreateComment (Servlet)

## **2.4 Componenti del progetto Server Side**

### **2.4.1 Model Object (Beans)**

- Album
- Comment
- Image
- User

### **2.4.2 Data Access Object (Classes)**

- AlbumDAO
- AlbumImagesDAO
- CommentDAO
- ImageDAO
- UserDAO

### **2.4.3 Controllers (Servlets)**

- CheckLogin
- CreateAlbum
- CreateComment
- CreateImage
- CreateUser
- GetAlbumsList
- GetCommentsList
- GetImagesByAlbumId
- SaveAlbumsOrder
- Logout
- GetImage\*



## 2.5 Componenti del progetto Client Side

### 2.5.1 View & view components

#### Index

#### Login form

- Gestione errori e submit

#### Home

#### ImageDetails

- Update: aggiorna componenti visive mostrando l'immagine scelta a dimensione reale e i suoi dettagli
- ShowComments: recupera i commenti dal DB relativi a una determinata immagine
- UpdateComments: mostra i commenti relativi all'immagine scelta
- Reset: imposta le condizioni di iniziali visibilità
- RegisterEvent: associa al componente la funzione per la creazione di un commento

#### UploadImage

- UpdateCheckBox: crea le checkbox da inserire nella form di caricamento immagini per selezionare gli albums in cui inserire l'immagine
- RegisterEvent: associa al componente la funzione per il caricamento di una immagine
- Reset: imposta le condizioni iniziali di visibilità

#### ImageList

- Show: richiede al DB la lista delle immagini dell'album selezionato
- Update: mostra al massimo 5 immagini dell'album aggiornando le componenti della vista
- RegisterEvent: registra gli eventi sui pulsanti per la navigazione delle immagini
- Reset: imposta le condizioni iniziali di visibilità

#### CreateAlbum

- RegisterEvent: associa al componente la funzione per il caricamento di un album
- Reset: imposta le condizioni iniziali di visibilità

#### AlbumList

- Show: richiede al DB la lista degli album
- Update: Aggiorna le componenti visive
- CreateTitle / CreateItem: creano i componenti visivi per la lista degli album
- DragStart/DragOver/DragLeave/Drop: permette di gestire l'evento di drag drop sulla lista dei propri album
- unselectRows: sistema la vista della lista degli album
- getAlbumsOrder: restituisce l'elenco in ordine di apparizione degli album
- Autoclick: simula il click sul primo album della lista

## 2.6 Sequence diagram

Di seguito vengono mostrati i sequence diagram dei possibili eventi. Si è scelto di indicare in **azzurro** le componenti lato server (Servlets e DAO) e in bianco quelle lato client (Javascript).

### 2.6.1 Evento di login

Il **click del bottone di login** genera una chiamata AJAX POST alla servlet **CheckLogin** che dopo aver controllato le credenziali tramite interrogazione al DB, salva in sessione lo username e reindirizza alla pagina home.html

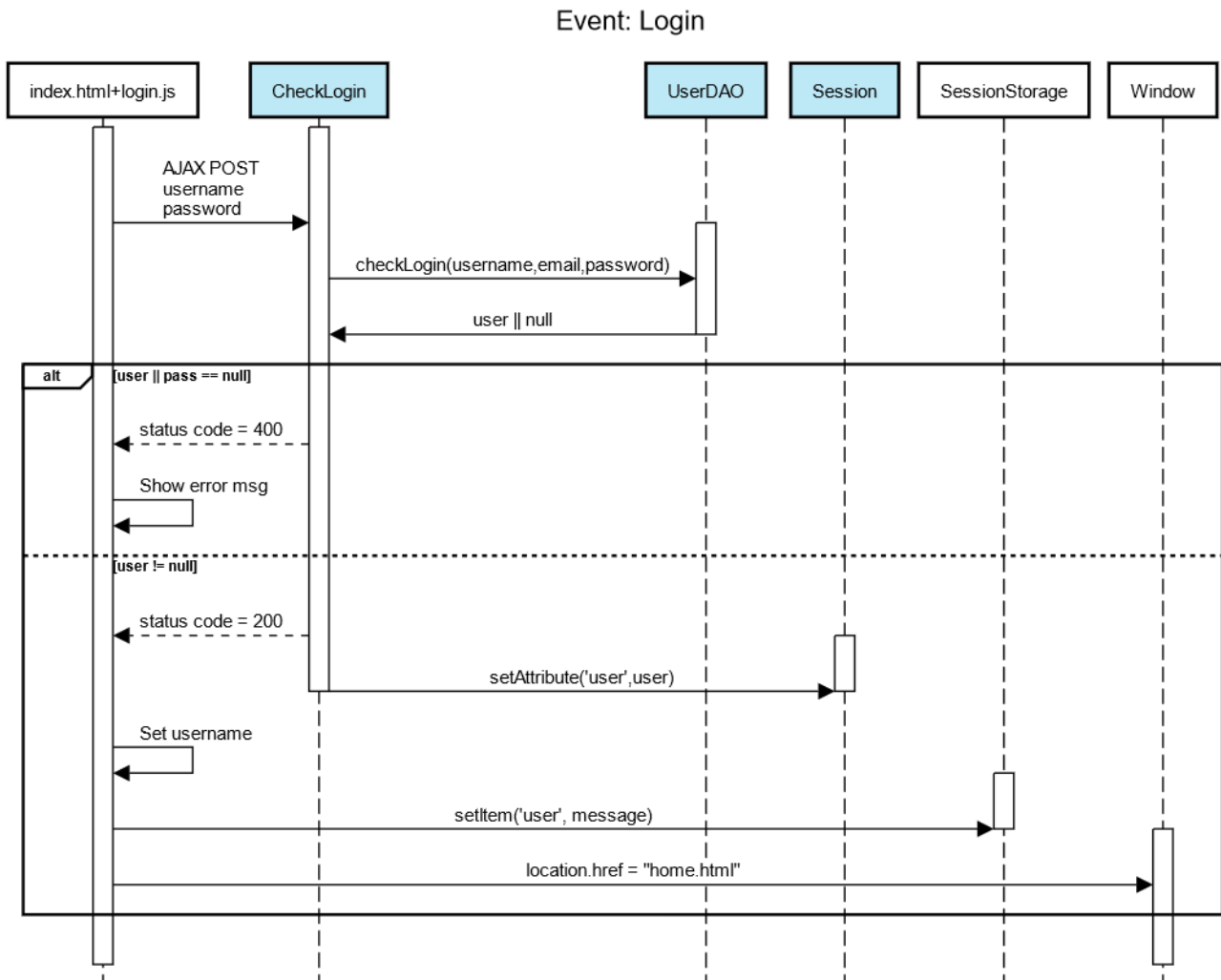


Figura 15: Sequence diagram evento login

### 2.6.2 Caricamento della Home

Al **caricamento della homepage** viene creato un oggetto **PageOrchestrator** che si occupa di creare e resettare gli oggetti che si occupano di eseguire singolarmente le chiamate AJAX per la visualizzazione e la modifica del contenuto della pagina home. Il PageOrchestrator si occupa anche di chiamare la funzione show di **AlbumsList** per mostrare in home l'elenco degli album ed effettua un **autoclick** sul primo album della lista per mostrarlo.

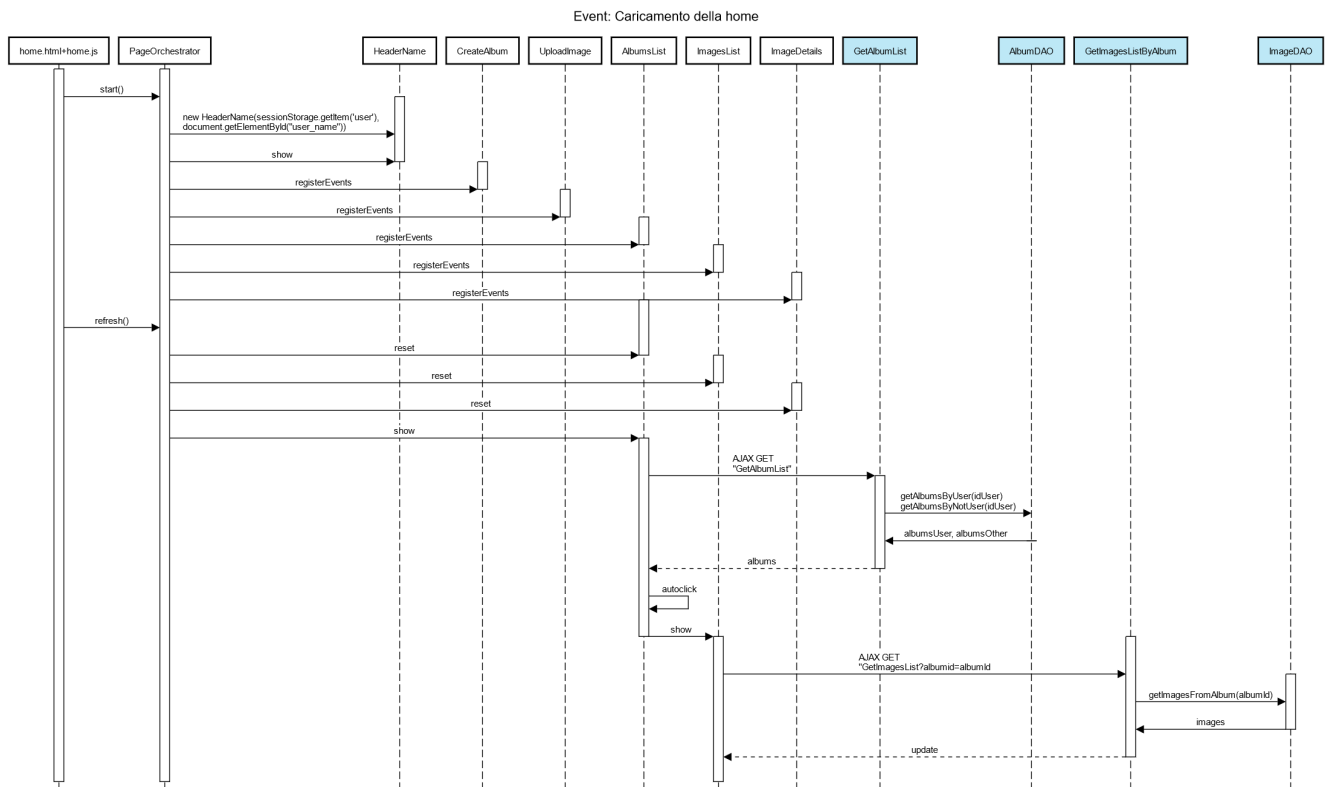


Figura 16: Sequence diagram caricamento homepage

### 2.6.3 Creazione di un album

Il click del bottone di creazione di un album, associato all'oggetto **CreateAlbum (Client)** genera una chiamata AJAX POST alla servlet **CreateAlbum (Server)** che dopo aver controllato la validità del titolo, crea l'album nella base di dati e invoca un refresh della pagina tramite **PageOrchestrator** per mostrare il nuovo album.

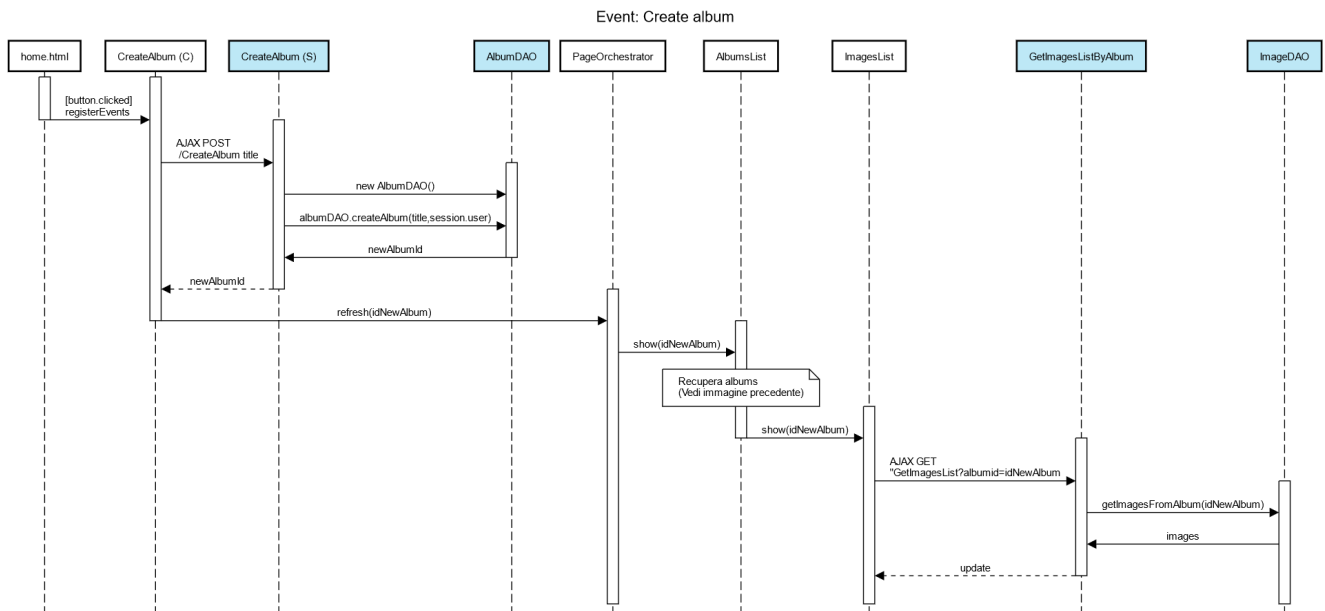


Figura 17: Sequence diagram creazione album

## 2.6.4 Caricamento immagine

Il click del bottone di caricamento di una immagine, associato all'oggetto **UploadImage**, genera una chiamata AJAX POST alla servlet **CreateImage** che controlla la validità del titolo, della descrizione e degli albums a cui si vuole aggiungere l'immagine. In caso di correttezza di questi, crea l'immagine e risponde alla chiamata AJAX in modo affermativo, altrimenti restituisce messaggio e codice di errore.

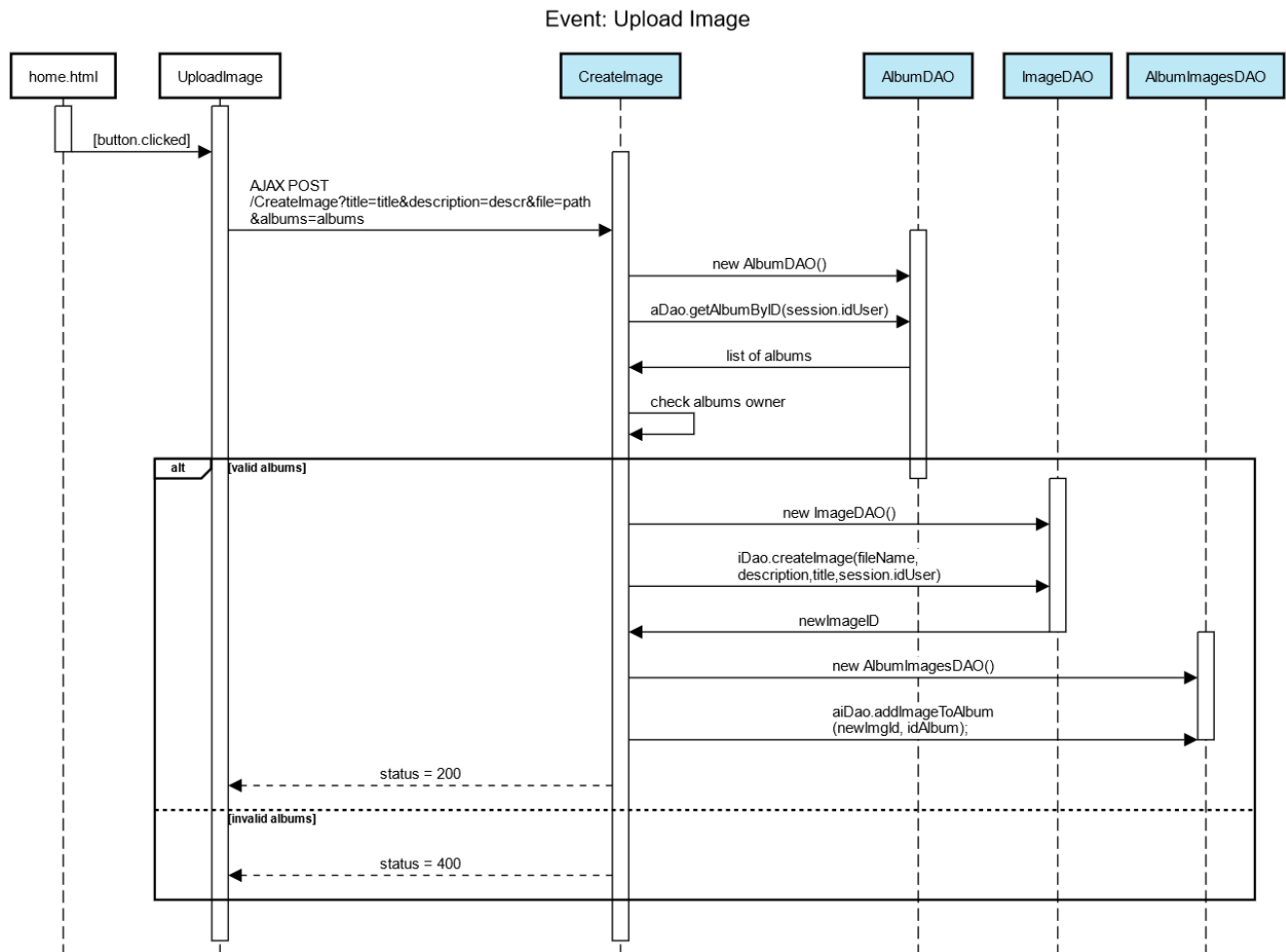


Figura 18: Sequence diagram caricamento immagine

### 2.6.5 Mostra immagini di un album

Il **click sul nome di un album dell'elenco** chiama la funzione `show` dell'oggetto **ImagesList**, che a sua volta genera una chiamata AJAX GET alla servlet **GetImagesListByAlbum** che recupera dal database l'elenco delle immagini dell'album selezionato e le restituisce all'oggetto javascript. Questo si occuperà di selezionare e mostrare solo le prime 5 (o il numero massimo se inferiore) salvandosi una copia dell'intero album in locale.

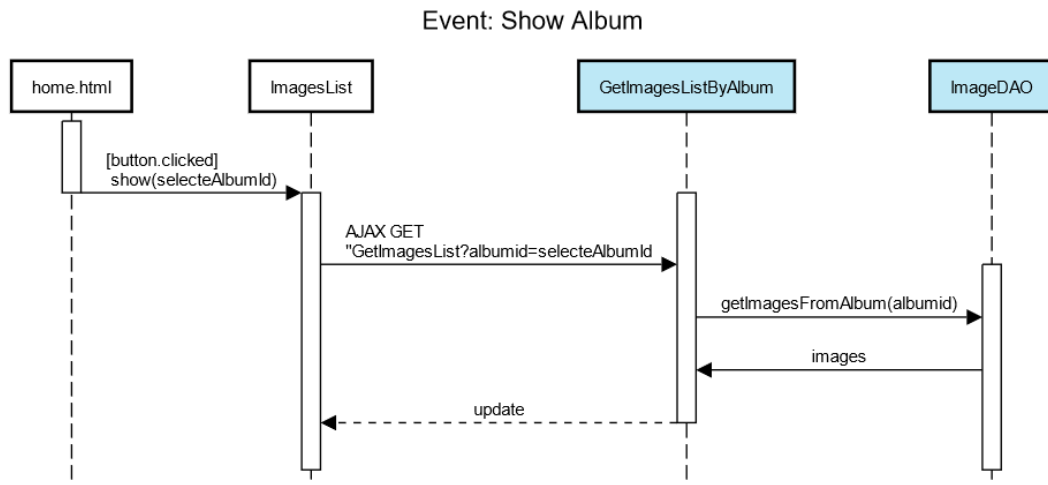


Figura 19: Sequence diagram click mostra album

### 2.6.6 Mostra blocco di immagini successivo/precedente

Il **click del bottone per la navigazione delle immagini** è associato all'oggetto **ImageList** che si occuperà di aggiornare la variabile indice (che tiene traccia della porzione di immagini che viene mostrata) e aggiornare le immagini mostrate.

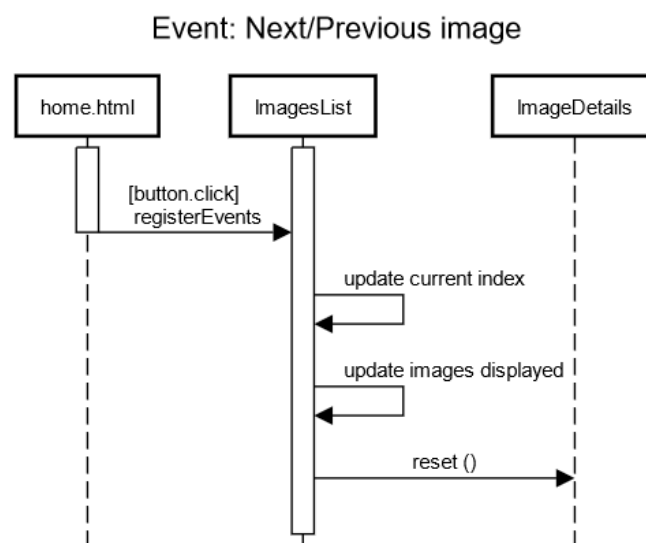


Figura 20: Sequence diagram navigazione album

### 2.6.7 Mostra dettagli immagine di un album

L' **hover con il mouse su una immagine** chiama la funzione `update` dell'oggetto **ImageDetails**, che oltre a recuperare le informazioni dell'immagine dalla copia locale dell'album, effettua una chiamata AJAX GET alla servlet **GetCommentsList** che restituirà l'elenco dei commenti associati all'immagine presenti nel database.

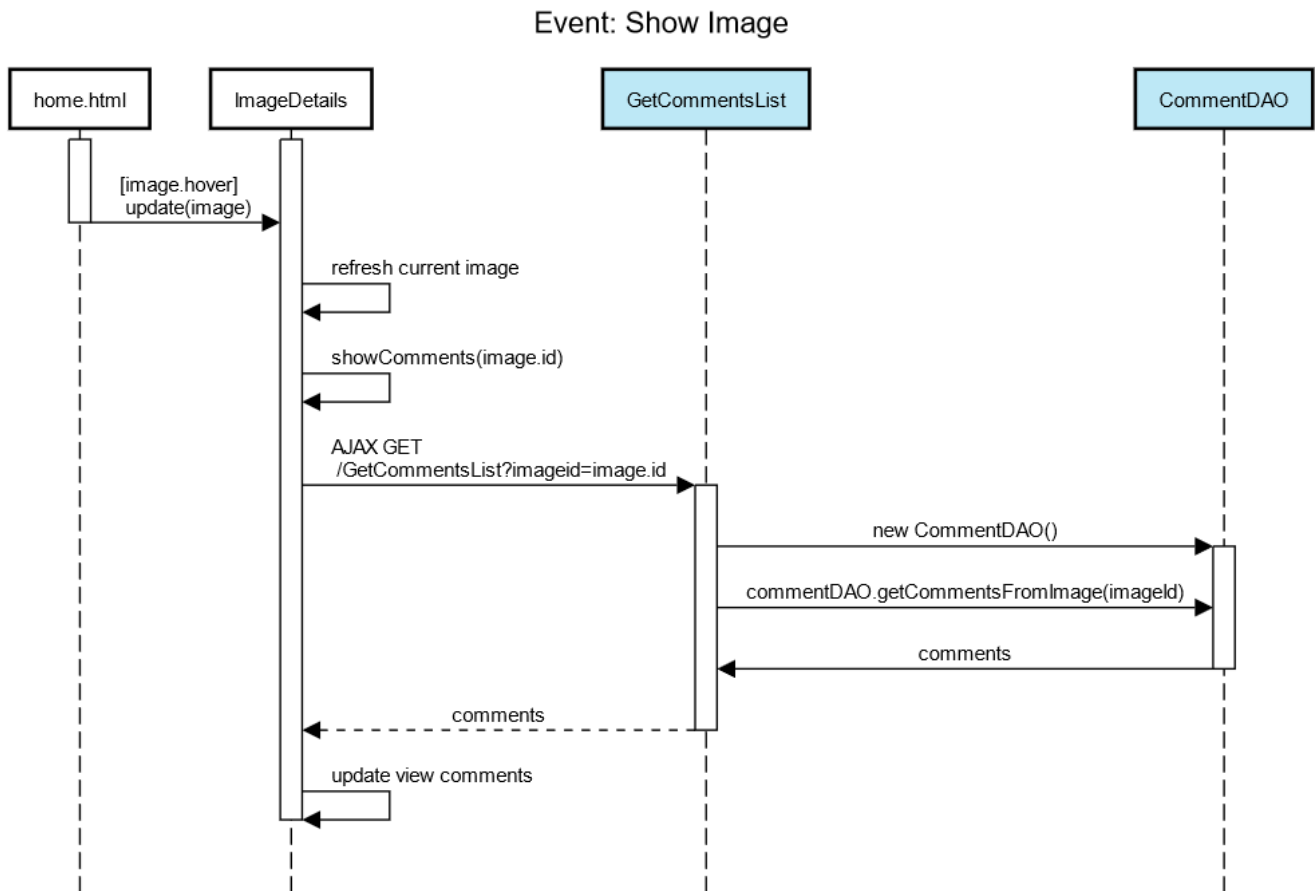


Figura 21: Sequence diagram hover immagine

### 2.6.8 Creazione di un commento

Il click del bottone di creazione di un commento, associato all'oggetto **ImageDetails** genera una chiamata AJAX POST alla servlet **CreateComment** che dopo aver controllato la validità del testo e dell'immagine a cui si vuole associare, crea il commento e restituisce una risposta positiva. L'oggetto **ImageDetails** quindi invoca la servlet **GetCommentsList** per aggiornare i commenti dell'immagine, nostrando anche quello appena creato.

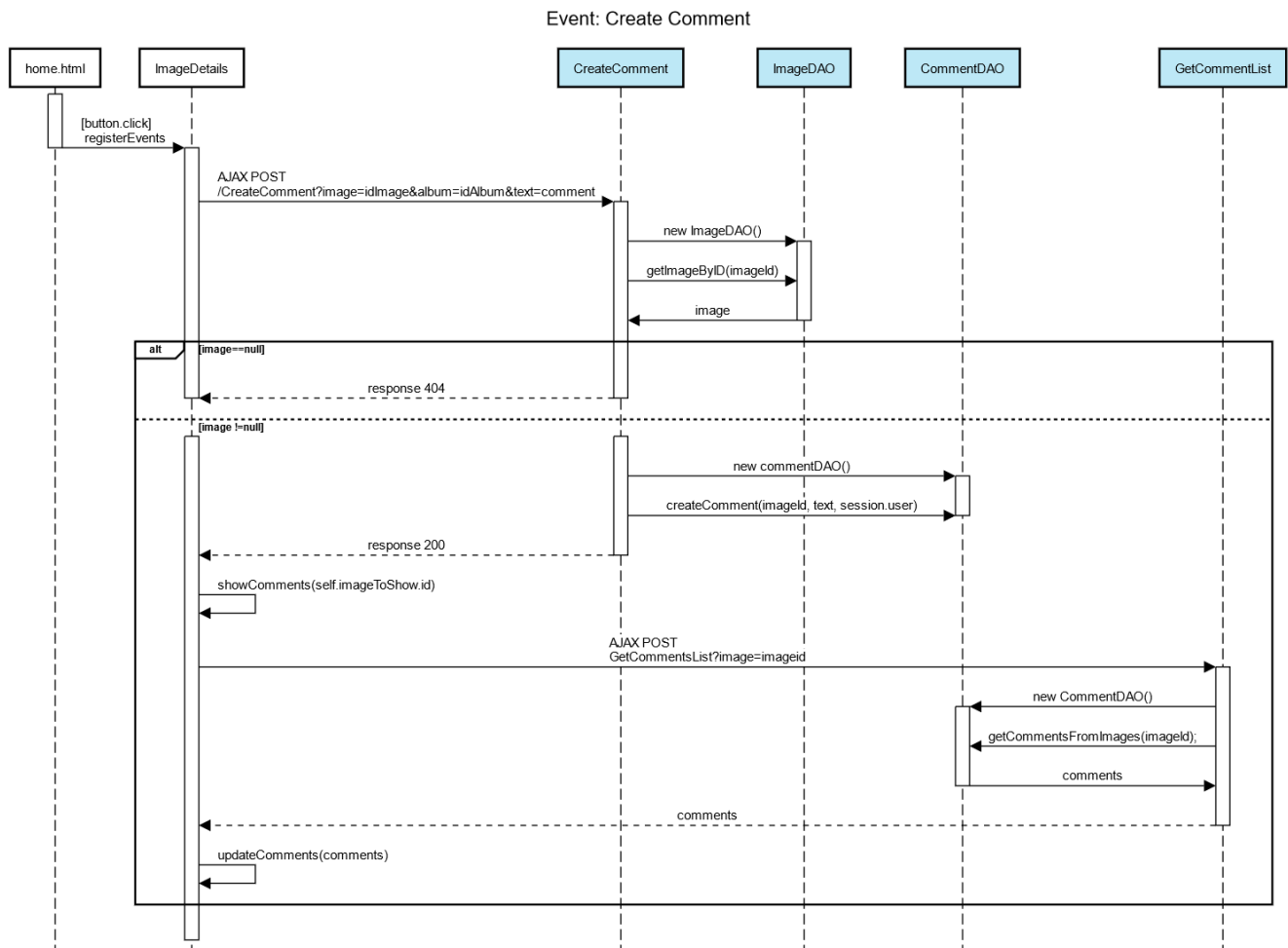


Figura 22: Sequence diagram creazione di un commento



### 2.6.9 Riordinamento dei propri albums

Il **riordinamento dei propri album** genera, a seconda del tipo di evento, una chiamata alla specifica funzione dell'oggetto **AlbumsList** che si occupa di tenere aggiornata la lista visualizzata.

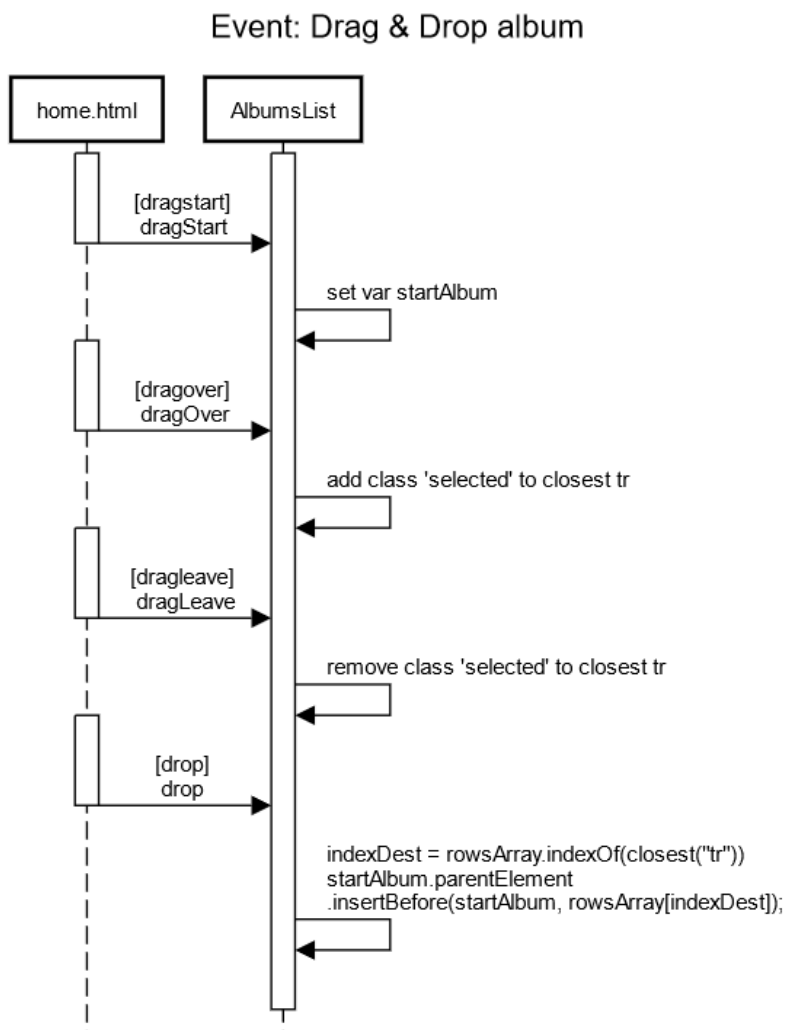


Figura 23: Sequence diagram riordinamento albums

### 2.6.10 Salvataggio ordine personalizzato albums

Il click del bottone di salvataggio dell'ordine degli albums invoca una funzione dell'oggetto **AlbumsList** che dopo aver recuperato l'ordine degli id degli albums, effettua una chiamata AJAX POST alla servlet **SaveAlbumOrder** che dopo opportuni controlli, salverà nella base di dati i nuovi indici di ordinamento.

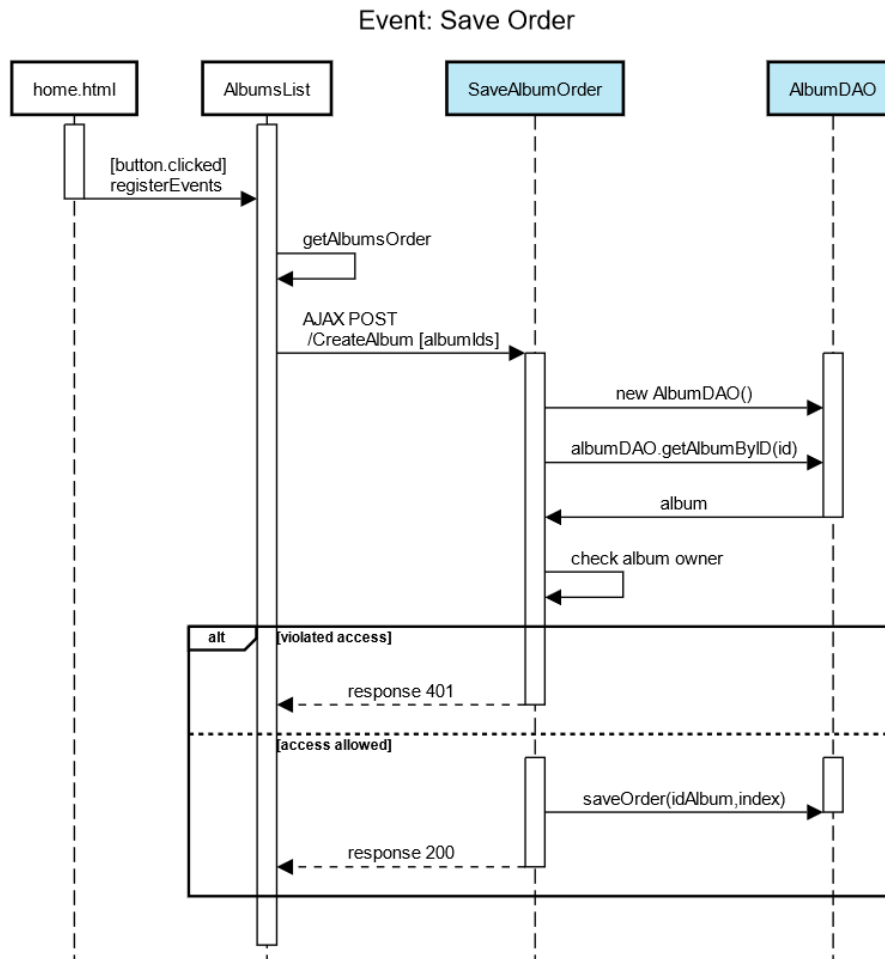


Figura 24: Sequence diagram salvataggio ordine personalizzato albums

### 2.6.11 Logout

Il click del pulsante di logout rimuove dalla sessione l'utente e reindirizza alla pagina di login.

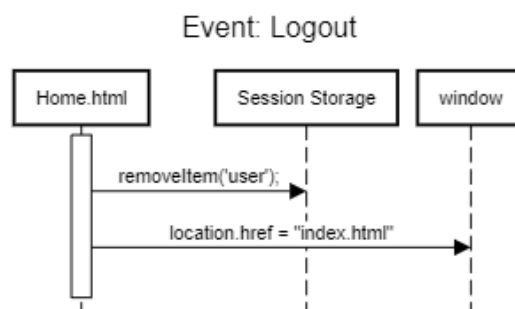


Figura 25: Sequence diagram logout