

Progetto 4

Trasferimento denaro

Fabio Sabbion

10730401

Analisi dei dati

Un'applicazione web consente la gestione di trasferimenti di denaro online da un conto a un altro. L'applicazione supporta registrazione e login mediante una pagina pubblica con opportune form. La registrazione controlla la validità sintattica dell'indirizzo di email e l'uguaglianza tra i campi "password" e "ripeti password". La registrazione controlla l'unicità dello username. Un utente ha un nome, un cognome, uno username e uno o più conti correnti. Un conto ha un codice, un saldo, e i trasferimenti fatti (in uscita) e ricevuti (in ingresso) dal conto. Un trasferimento ha una data, un importo, un conto di origine e un conto di destinazione. Quando l'utente accede all'applicazione appare una pagina LOGIN per la verifica delle credenziali. In seguito all'autenticazione dell'utente appare l'HOME page che mostra l'elenco dei suoi conti. Quando l'utente seleziona un conto, appare una pagina STATO DEL CONTO che mostra i dettagli del conto e la lista dei movimenti in entrata e in uscita, ordinati per data discendente. La pagina contiene anche una form per ordinare un trasferimento. La form contiene i campi: codice utente destinatario, codice conto destinatario, causale e importo. All'invio della form con il bottone INVIA, l'applicazione controlla che il conto di destinazione appartenga all'utente specificato e che il conto origine abbia un saldo superiore o uguale all'importo del trasferimento. In caso di mancanza di anche solo una condizione, l'applicazione mostra una pagina con un avviso di fallimento che spiega il motivo del mancato trasferimento. Nel caso in cui entrambe le condizioni siano soddisfatte, l'applicazione deduce l'importo dal conto di origine, aggiunge l'importo al conto di destinazione e mostra una pagina CONFERMA TRASFERIMENTO che presenta i dati dell'importo trasferito e i dati del conto di origine e di destinazione con i rispettivi saldi precedenti al trasferimento e aggiornati dopo il trasferimento. L'applicazione deve garantire l'atomicità del trasferimento: ogni volta che il conto di destinazione viene addebitato, il conto di origine deve essere accreditato. Ogni pagina contiene un collegamento per tornare alla pagina precedente. L'applicazione consente il logout dell'utente.

Entità, attributi, relazioni

Analisi delle specifiche

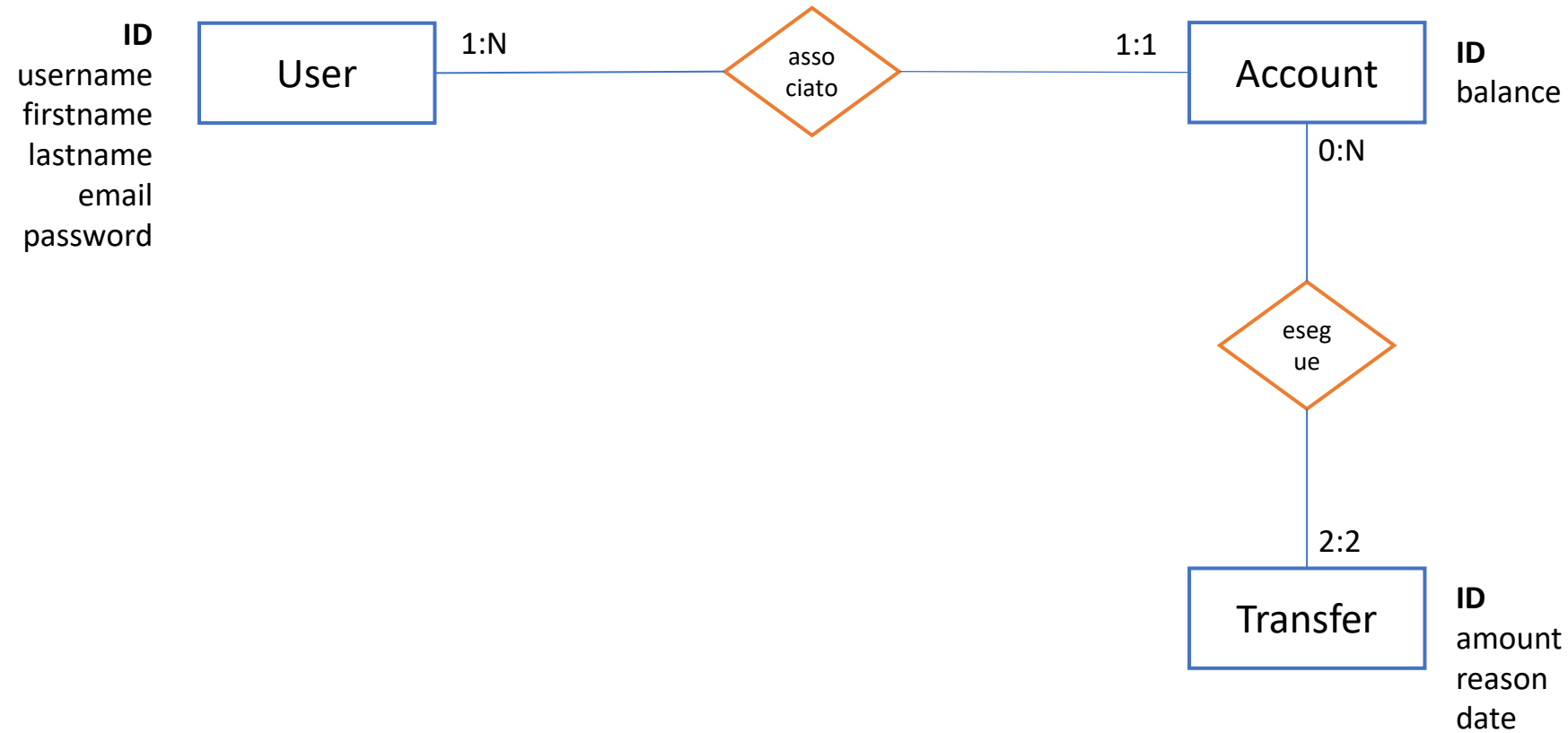
Un'applicazione web consente la gestione di trasferimenti di denaro online da un conto a un altro. L'applicazione supporta registrazione e login mediante una pagina pubblica con opportune form. La registrazione controlla la validità sintattica dell'indirizzo di email e l'uguaglianza tra i campi "password" e "ripeti password". La registrazione controlla l'unicità dello username. Un utente ha un nome, un cognome, uno username e uno o più conti correnti. Un conto ha un codice, un saldo, e i trasferimenti fatti (in uscita) e ricevuti (in ingresso) dal conto. Un trasferimento ha una data, un importo, un conto di origine e un conto di destinazione. Quando l'utente accede all'applicazione appare una pagina LOGIN per la verifica delle credenziali. In seguito all'autenticazione dell'utente appare l'HOME page che mostra l'elenco dei suoi conti. Quando l'utente seleziona un conto, appare una pagina STATO DEL CONTO che mostra i dettagli del conto e la lista dei movimenti in entrata e in uscita, ordinati per data discendente. La pagina contiene anche una form per ordinare un trasferimento. La form contiene i campi: codice utente destinatario, codice conto destinatario, causale e importo. All'invio della form con il bottone INVIA, l'applicazione controlla che il conto di destinazione appartenga all'utente specificato e che il conto origine abbia un saldo superiore o uguale all'importo del trasferimento. In caso di mancanza di anche solo una condizione, l'applicazione mostra una pagina con un avviso di fallimento che spiega il motivo del mancato trasferimento. Nel caso in cui entrambe le condizioni siano soddisfatte, l'applicazione deduce l'importo dal conto di origine, aggiunge l'importo al conto di destinazione e mostra una pagina CONFERMA TRASFERIMENTO che presenta i dati dell'importo trasferito e i dati del conto di origine e di destinazione con i rispettivi saldi precedenti al trasferimento e aggiornati dopo il trasferimento. L'applicazione deve garantire l'atomicità del trasferimento: ogni volta che il conto di destinazione viene addebitato, il conto di origine deve essere accreditato. Ogni pagina contiene un collegamento per tornare alla pagina precedente. L'applicazione consente il logout dell'utente.

Pages (views), view components, events, actions

Completamento delle specifiche

- Al momento della **registrazione**, viene **creato automaticamente un nuovo conto corrente** associato all'utente
- La pagina **HOME** contiene il pulsante **LOGOUT**
- Tutti i campi delle form sono richiesti

Database design



Database schema (1)

```
CREATE TABLE `user` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `username` VARCHAR(45) NOT NULL UNIQUE,  
  `firstname` VARCHAR(45) NOT NULL,  
  `lastname` VARCHAR(45) NOT NULL,  
  `email` VARCHAR(45) NOT NULL,  
  `password` VARCHAR(45) NOT NULL,  
  PRIMARY KEY (`id`)  
);
```

```
CREATE TABLE `account` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `balance` INT NOT NULL,  
  `userid` INT NOT NULL,  
  PRIMARY KEY (`id`),  
  CONSTRAINT `account_user`  
  FOREIGN KEY (`userid`)  
  REFERENCES `user` (`id`)  
  ON DELETE CASCADE  
  ON UPDATE NO ACTION  
);
```

Database schema (2)

```
CREATE TABLE `transfer` (  
  `id` INT NOT NULL AUTO_INCREMENT,  
  `amount` INT NOT NULL,  
  `reason` VARCHAR(250) NOT NULL,  
  `from_account` INT NOT NULL,  
  `to_account` INT NOT NULL,  
  `date` DATETIME NOT NULL DEFAULT CURRENT_TIMESTAMP,  
  PRIMARY KEY (`id`),  
  FOREIGN KEY (`from_account`)  
  REFERENCES `account` (`id`)  
  ON DELETE NO ACTION,  
  FOREIGN KEY (`to_account`)  
  REFERENCES `account` (`id`)  
  ON DELETE NO ACTION  
);
```

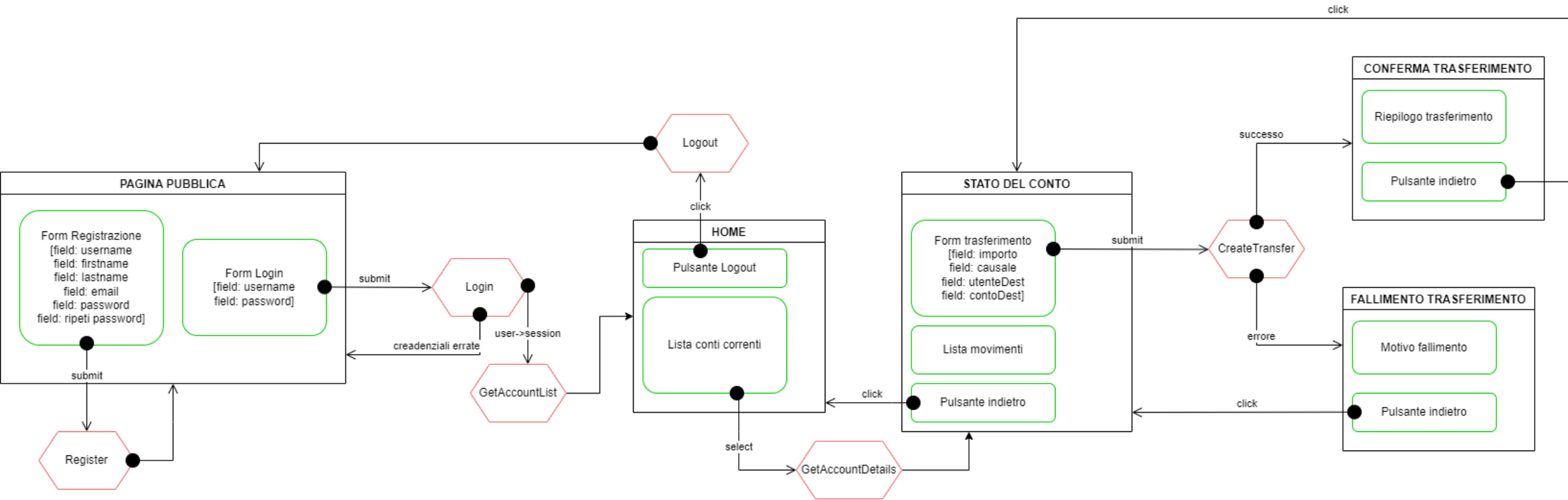
Viste e componenti

- **Pagina pubblica**
 - Form login
 - Form registrazione
- **HOME**
 - Lista conti correnti
 - Pulsante “logout”
- **STATO DEL CONTO**
 - Dettagli conto (codice, saldo)
 - Lista movimenti
 - Form nuovo trasferimento
 - Pulsante “indietro”
- **CONFERMA TRASFERIMENTO**
 - Riepilogo trasferimento
 - Pulsante “indietro”
- **FALLIMENTO TRASFERIMENTO**
 - Motivo del fallimento
 - Pulsante “indietro”

Eventi e azioni

- Login
 - Controllo credenziali
- Registrazione
 - Controllo unicità username
 - Controllo uguaglianza campi “password” e “ripeti password”
 - Controllo validità email
 - Creazione conto corrente
- Selezione conto corrente
 - Mostra pagina con dettagli conto
- Invio form trasferimento
 - Controllo appartenenza del conto di destinazione all’utente indicato
 - Controllo saldo sufficiente
- Logout
 - Logout dell’utente e ritorno alla pagina principale
- Click pulsanti “indietro”
 - Mostra pagina precedente

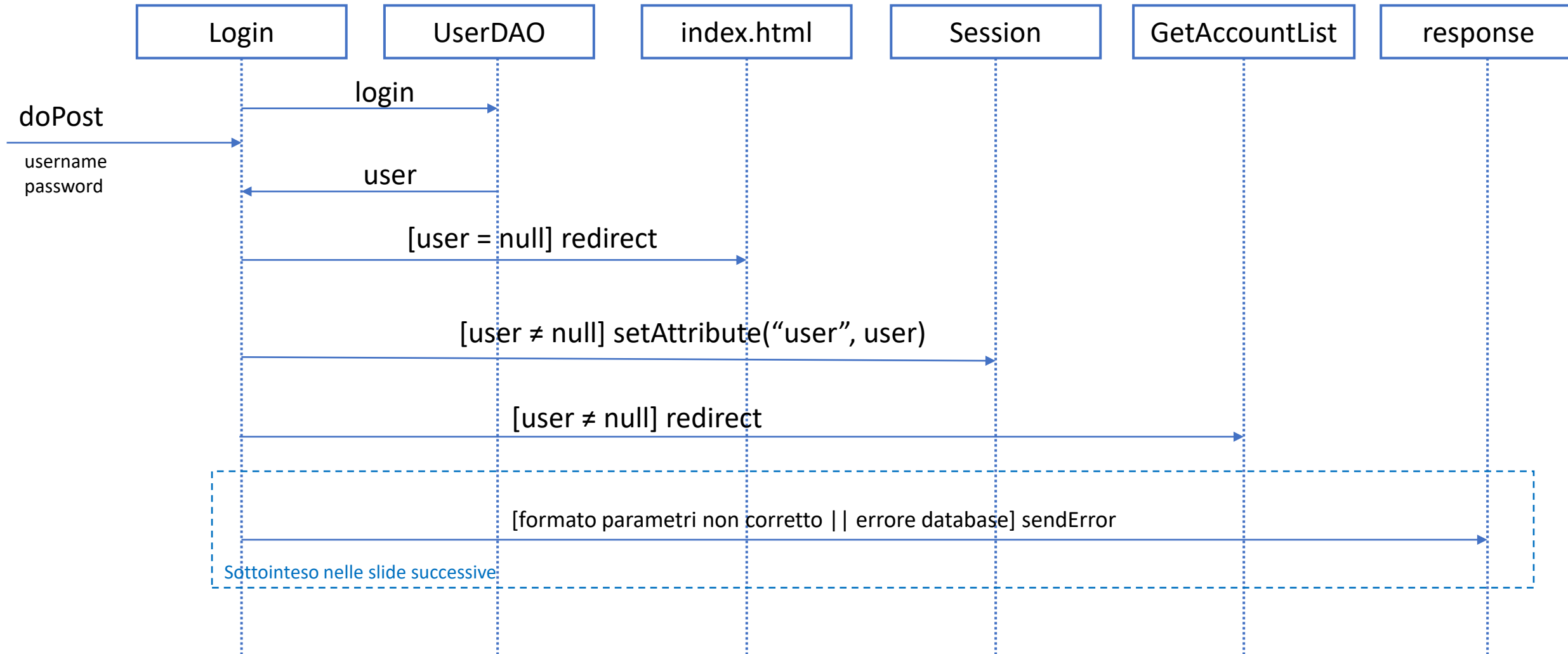
Application design



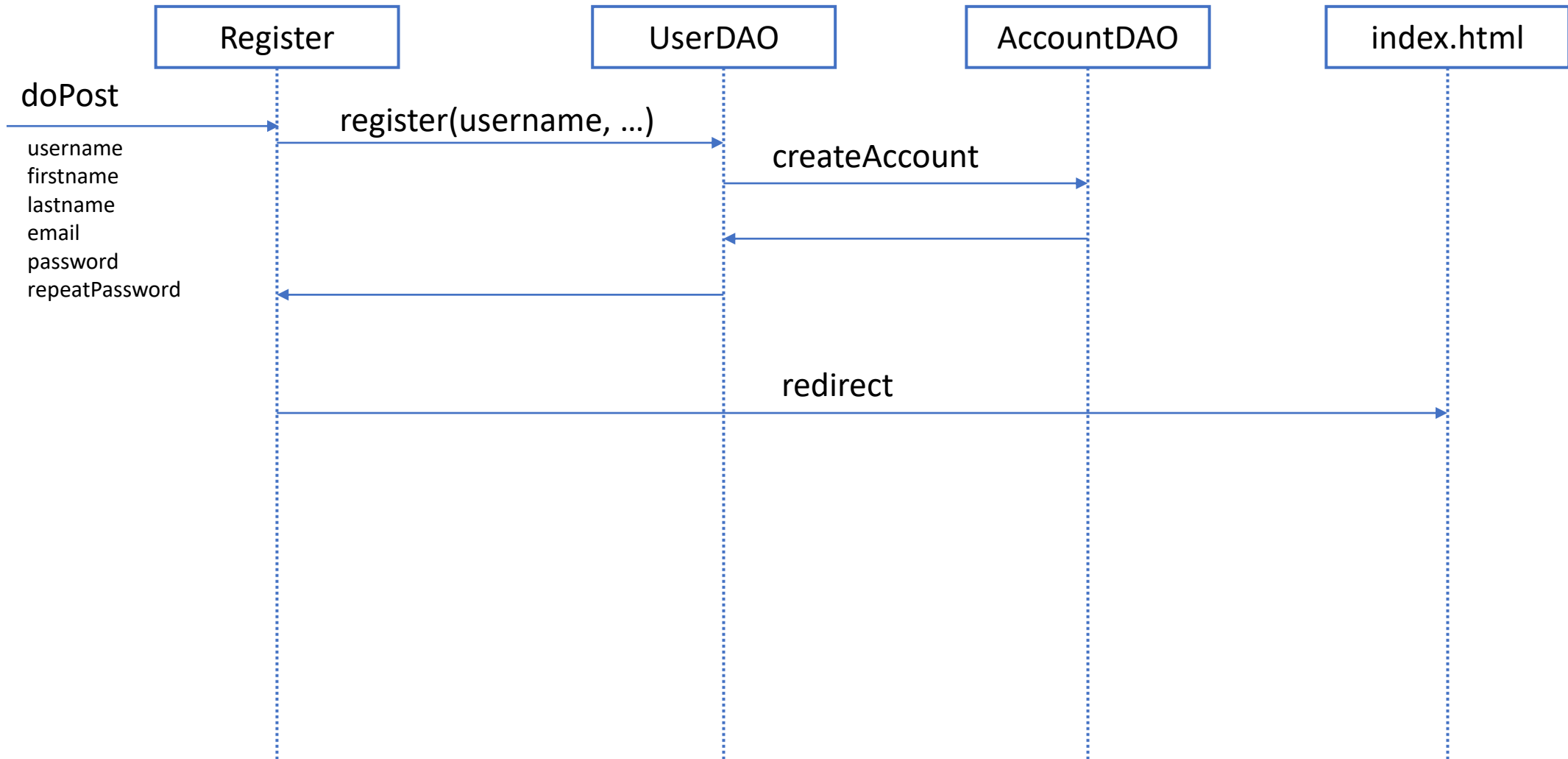
Componenti

- Model objects (Beans)
 - User
 - Account
 - Transfer
 - TransferSummary (include dettagli conti correnti prima e dopo il trasferimento)
- Data Access Objects
 - UserDao
 - login
 - register
 - getIdFromUsername
 - AccountDAO
 - createAccount
 - getAccountDetails
 - getAccountsById
 - TransferDAO
 - createTransfer
 - getTransfersByAccountId
 - getTransferById
- Controllers (Servlet)
 - Login
 - Logout
 - Register
 - GetAccountList
 - GetAccountDetails
 - CreateTransfer
- Views & Components
 - Index.html (pagina pubblica)
 - Form login
 - Form registrazione
 - Homepage.jsp
 - Lista conti correnti
 - Pulsante "Logout"
 - AccountPage.jsp
 - Dettagli conto
 - Lista Movimenti
 - Form nuovo trasferimento
 - Pulsante "indietro"
 - TransferConfirm.jsp
 - Riepilogo trasferimento
 - Pulsante "indietro"
 - TransferError.jsp
 - Motivo fallimento
 - Pulsante "indietro"

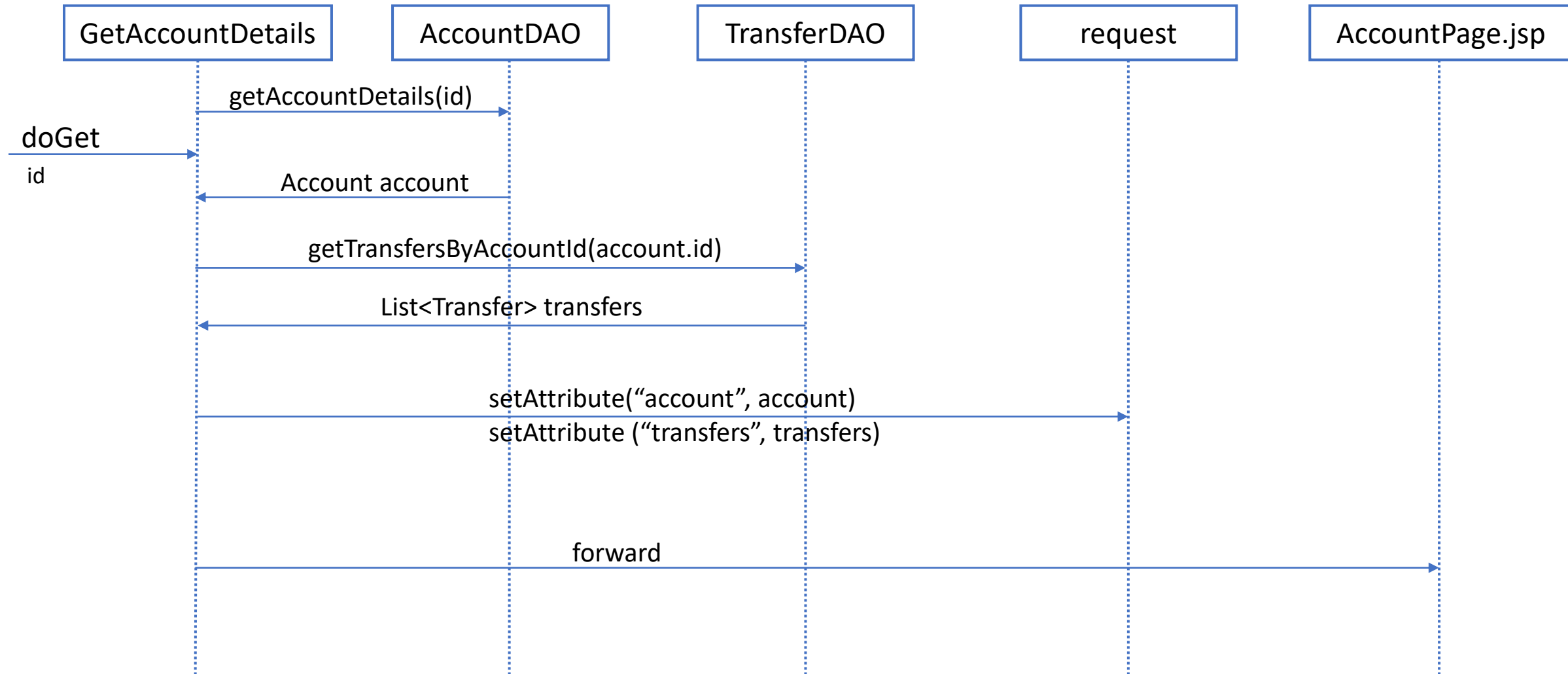
Evento: Login



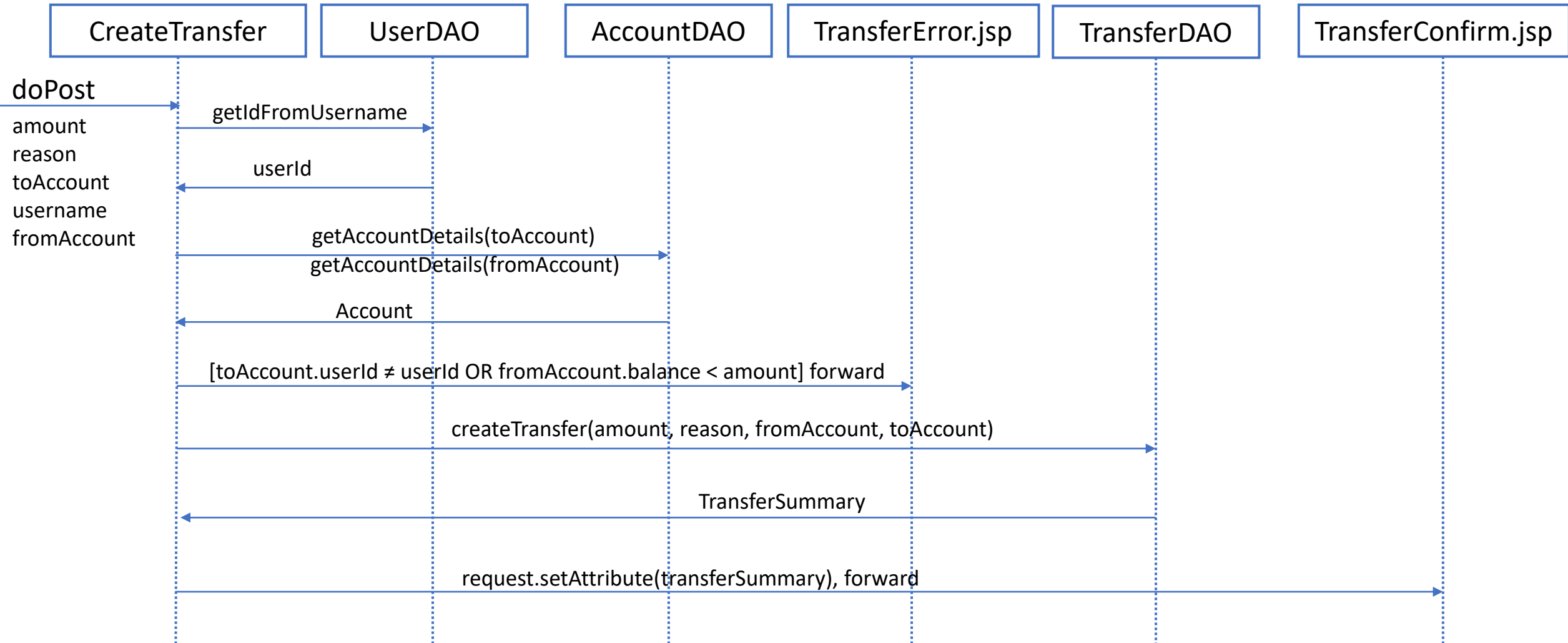
Evento: registrazione



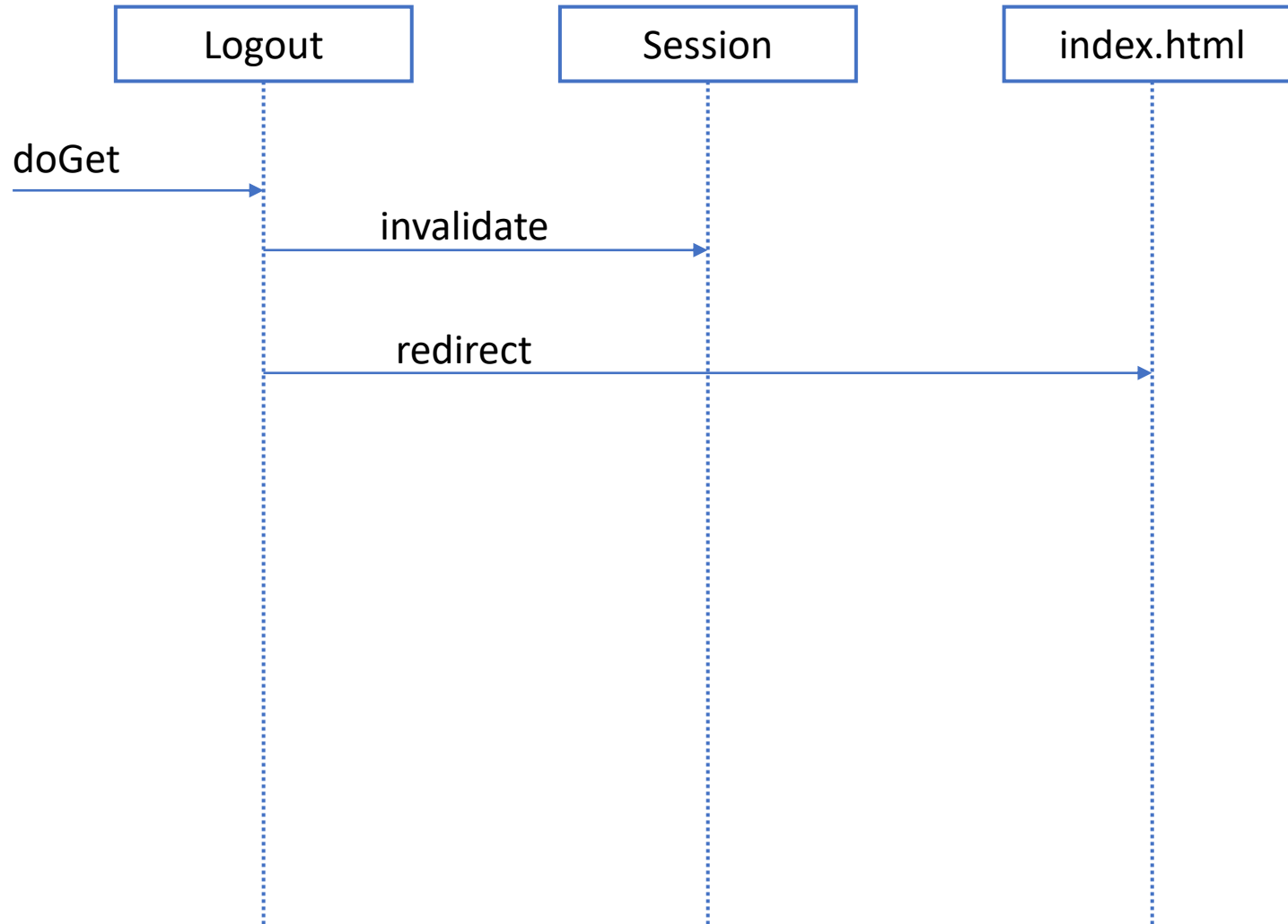
Evento: selezione conto corrente



Evento: invio form trasferimento

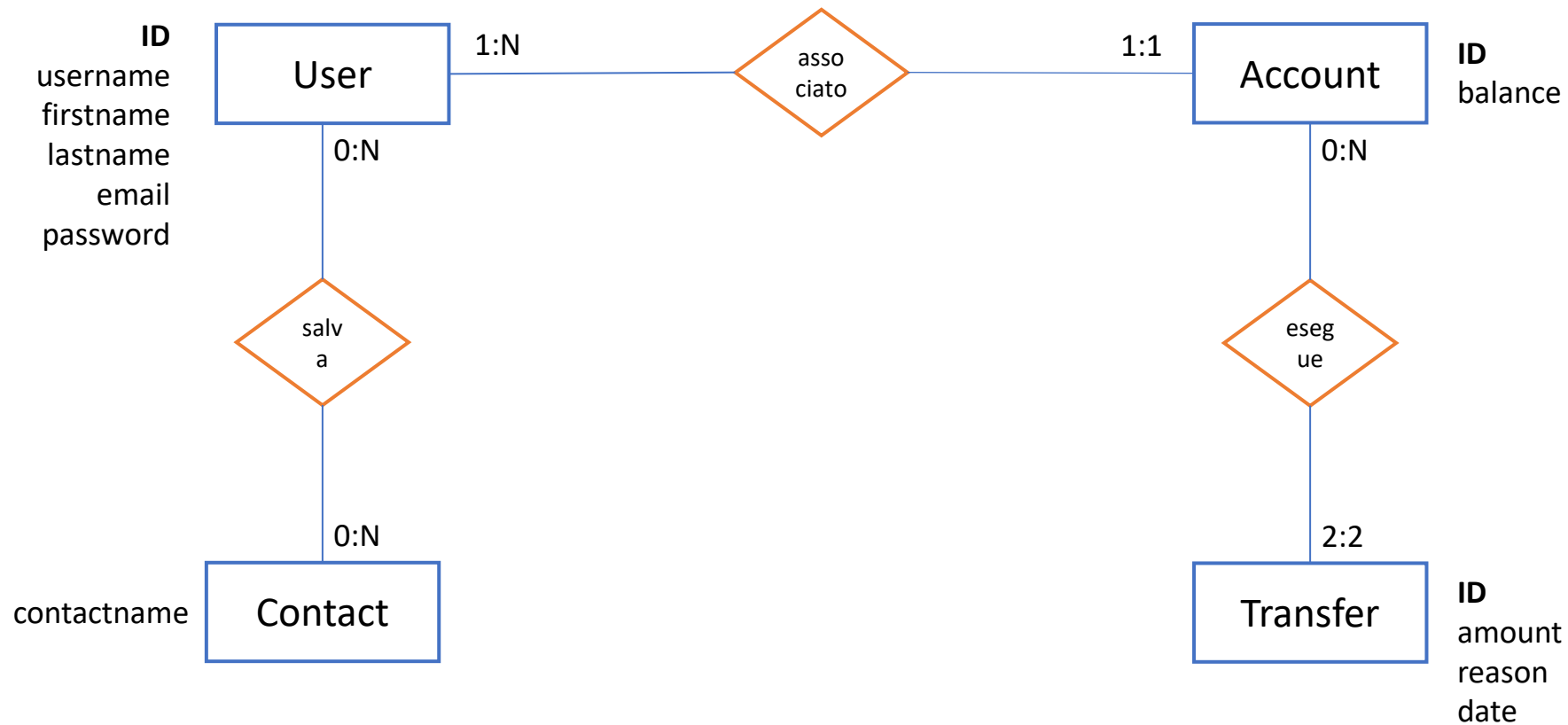


Evento: logout



Versione con JavaScript

Database design



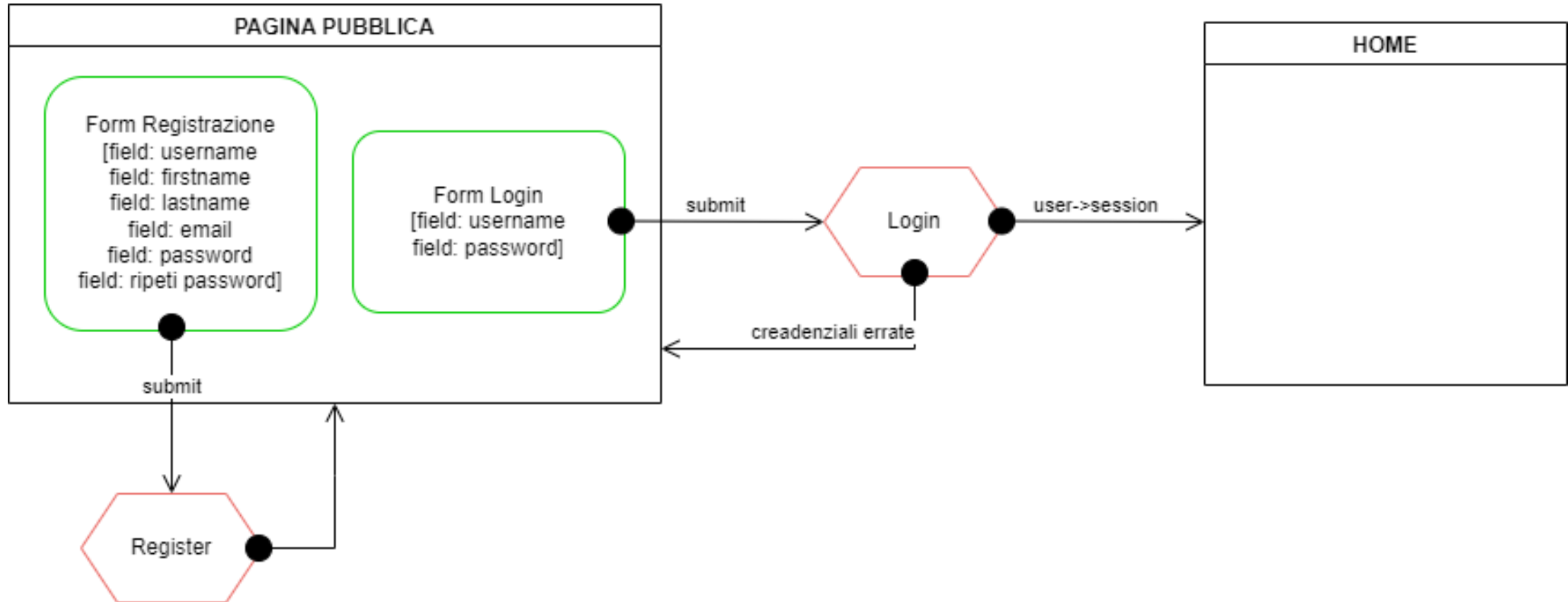
Database schema

```
CREATE TABLE `contact` (  
  `userid` INT NOT NULL,  
  `contactname` VARCHAR(45) NOT NULL,  
  PRIMARY KEY(`userid`, `contactname`),  
  CONSTRAINT `contact_user`  
  FOREIGN KEY (`userid`)  
  REFERENCES `user` (`id`)  
  ON DELETE CASCADE  
);
```

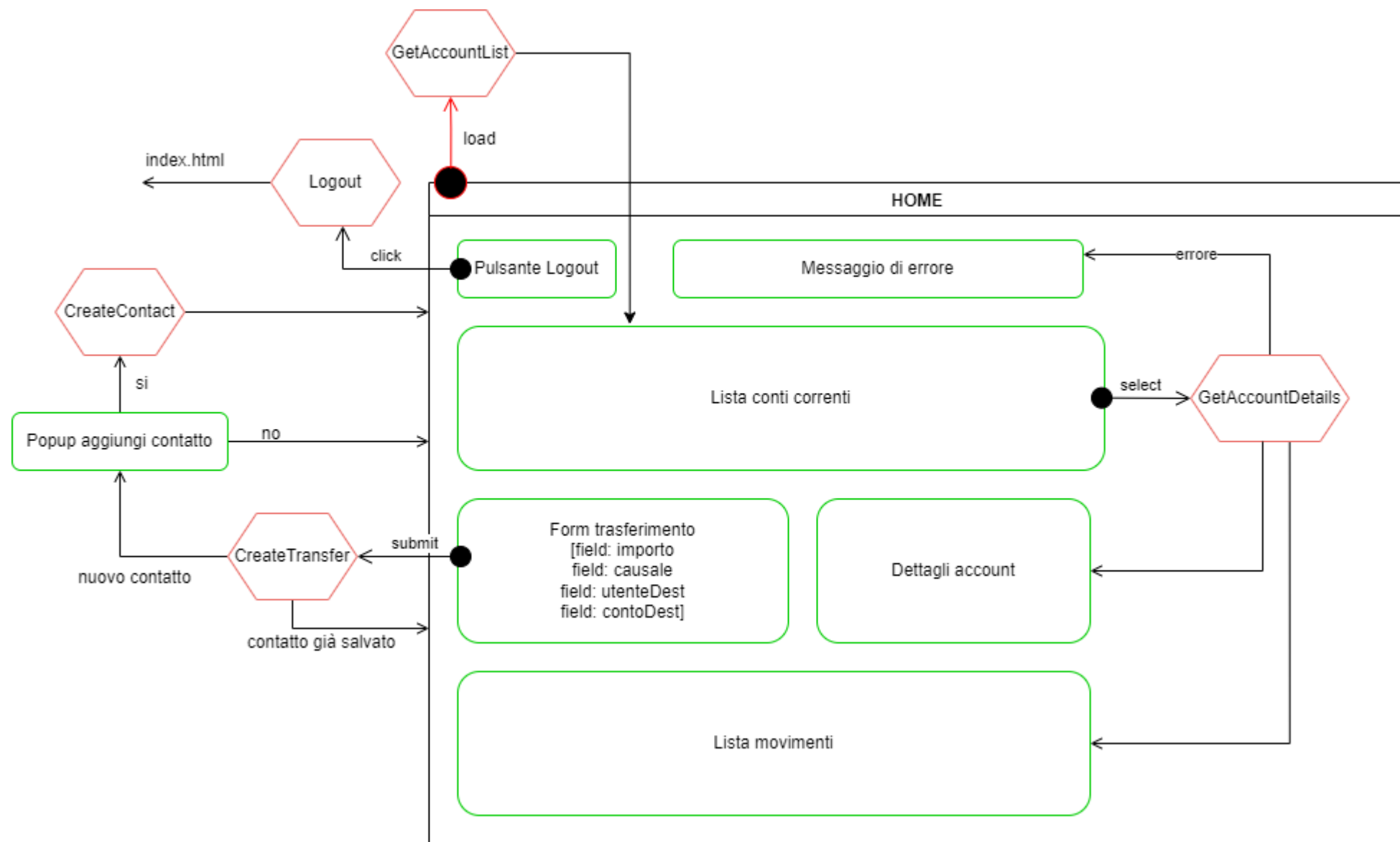
Viste e componenti

- Pagina pubblica
 - Form login
 - Form registrazione
- HOME
 - Lista conti correnti
 - Form trasferimento
 - Dettagli conto corrente
 - Lista movimenti
 - Messaggio di errore
 - Pulsante logout

Application design (1)



Application design (2)



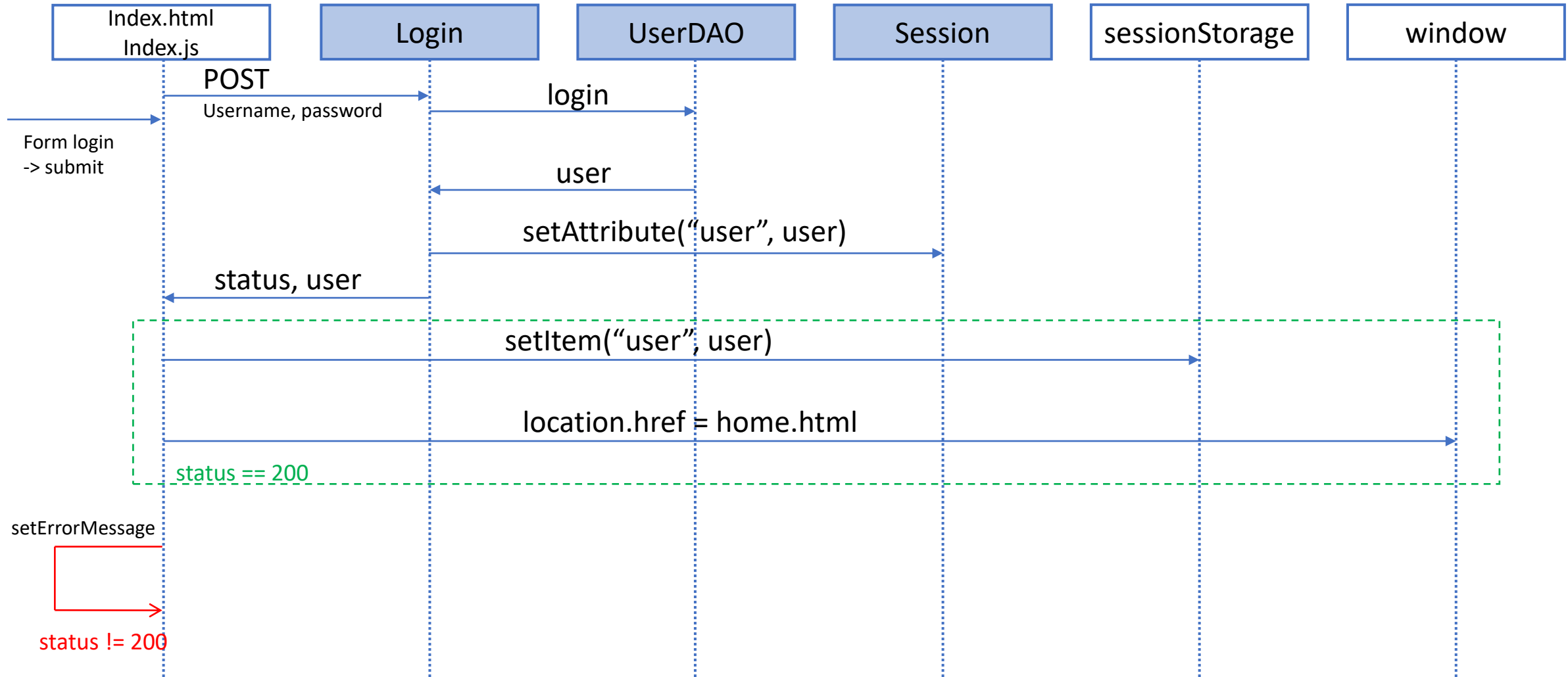
Eventi e azioni

Client side		Server side	
Evento	Azione	Evento	Azione
Index -> form registrazione-> submit	Controllo validità campi	POST /Register Username, firstname, lastname...	Controllo dati, creazione nuovo utente e conto
Index -> form login -> submit	Controllo validità campi	POST /Login Username, password	Controllo credenziali, salvataggio sessione
Home -> load	Aggiorna vista con elenco conti correnti	GET /GetAccountList GET /Contacts	Estrazione lista conti correnti e contatti salvati dell'utente autenticato
Home -> elenco conti -> seleziona conto	Aggiorna vista con dettagli conto e lista trasferimenti	GET /GetAccountDetails?id=...	Controllo parametro, estrazione dettagli conto e lista trasferimenti
Home -> form nuovo trasferimento -> submit	Aggiorna lista trasferimenti con nuovo trasferimento	POST /CreateTrasfer Amount, reason, ...	Controlli, creazione trasferimento in database
Home -> form nuovo trasferimento -> submit -> success	Controlla contatto presente in rubrica, mostra popup di aggiunta contatto	POST /Contacts username	Creazione nuovo contatto in database
Home -> pulsante logout -> click	Torna a index.html	GET /Logout	Invalida sessione

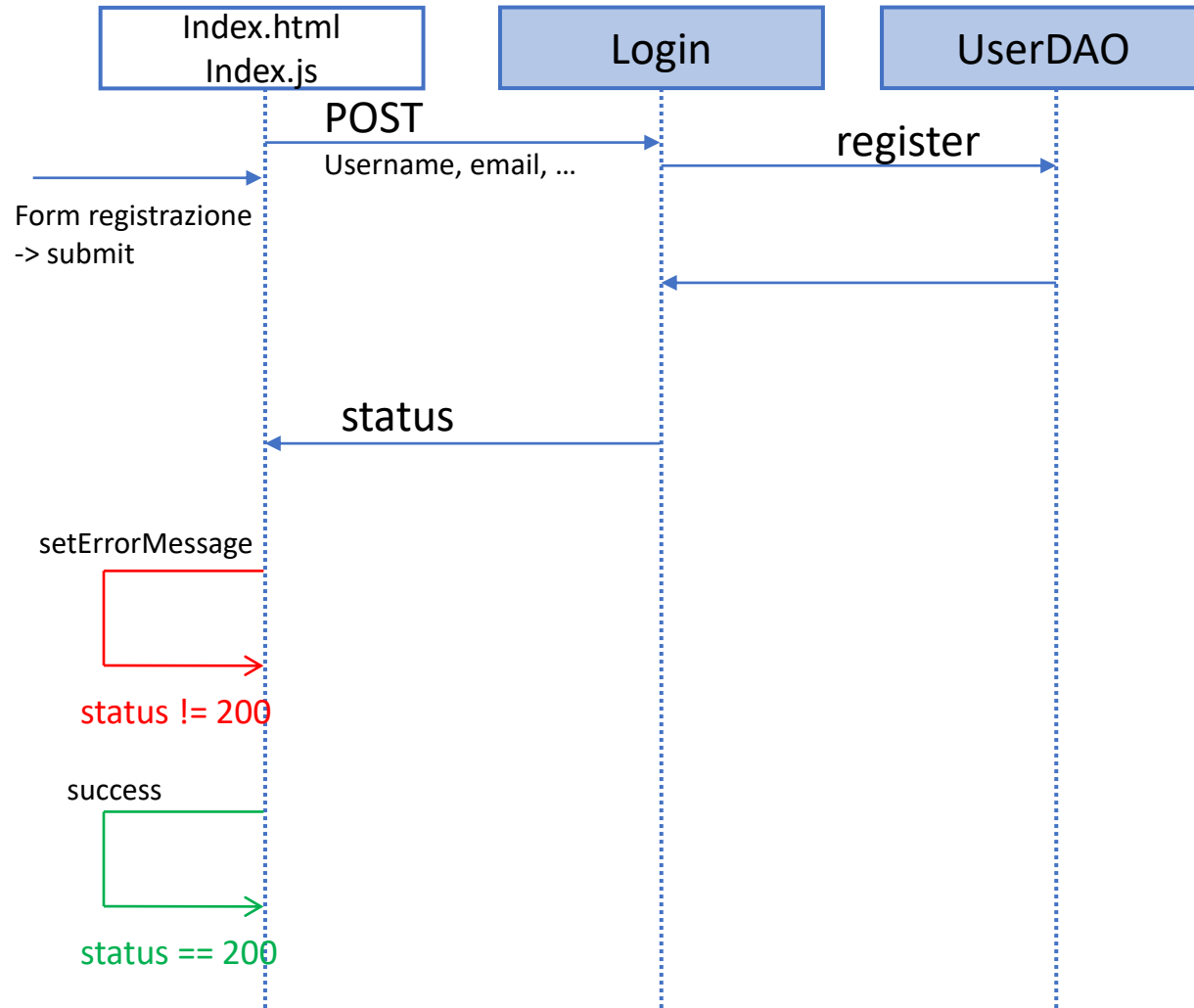
Nuovi componenti

- Model objects (Beans)
 - AccountDetails (include dettagli account e lista trasferimenti)
- Controllers (Servlet)
 - Contacts
- Data Access Objects
 - ContactDAO
 - getContacts
 - createContact

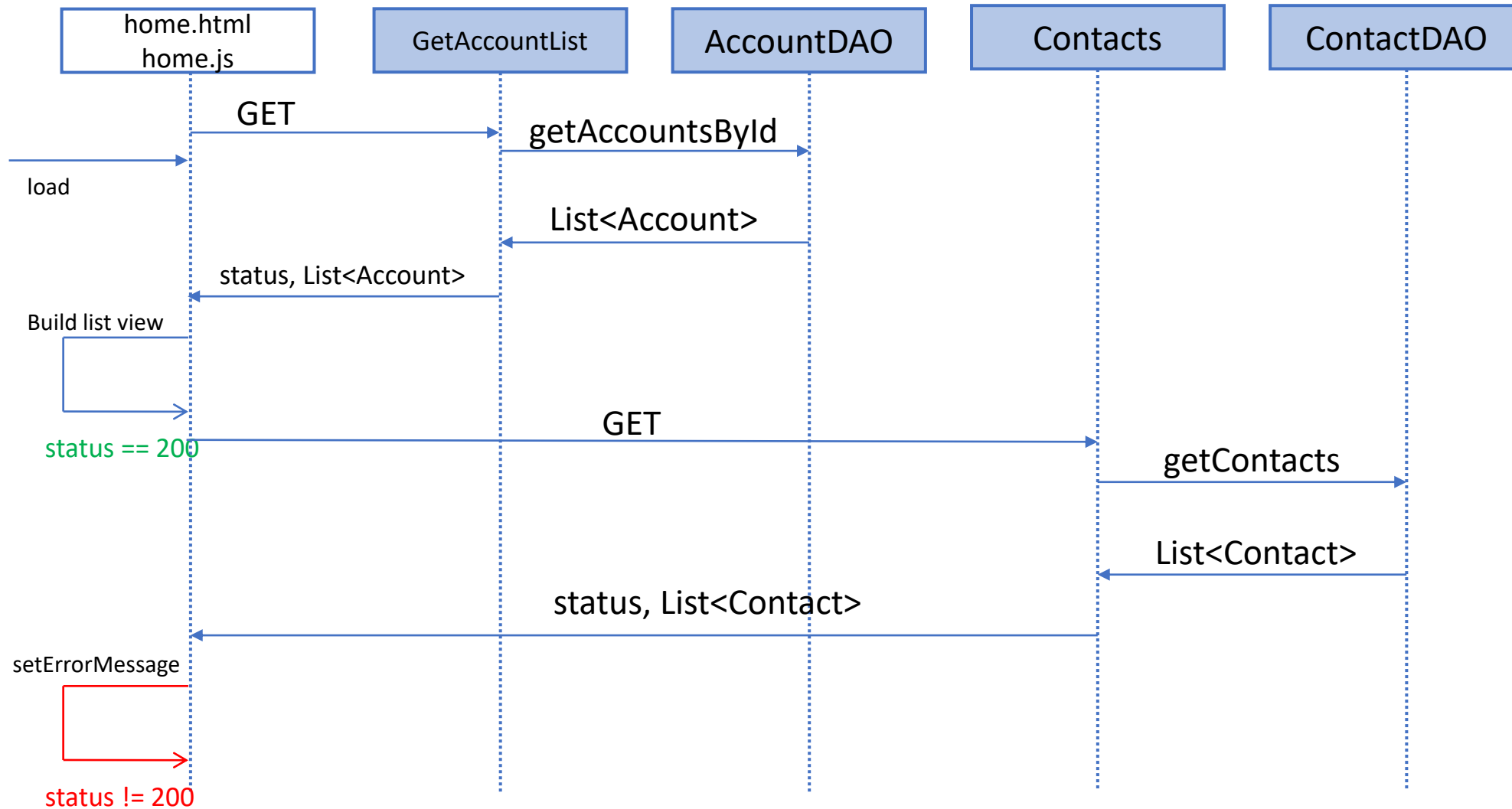
Evento: Login



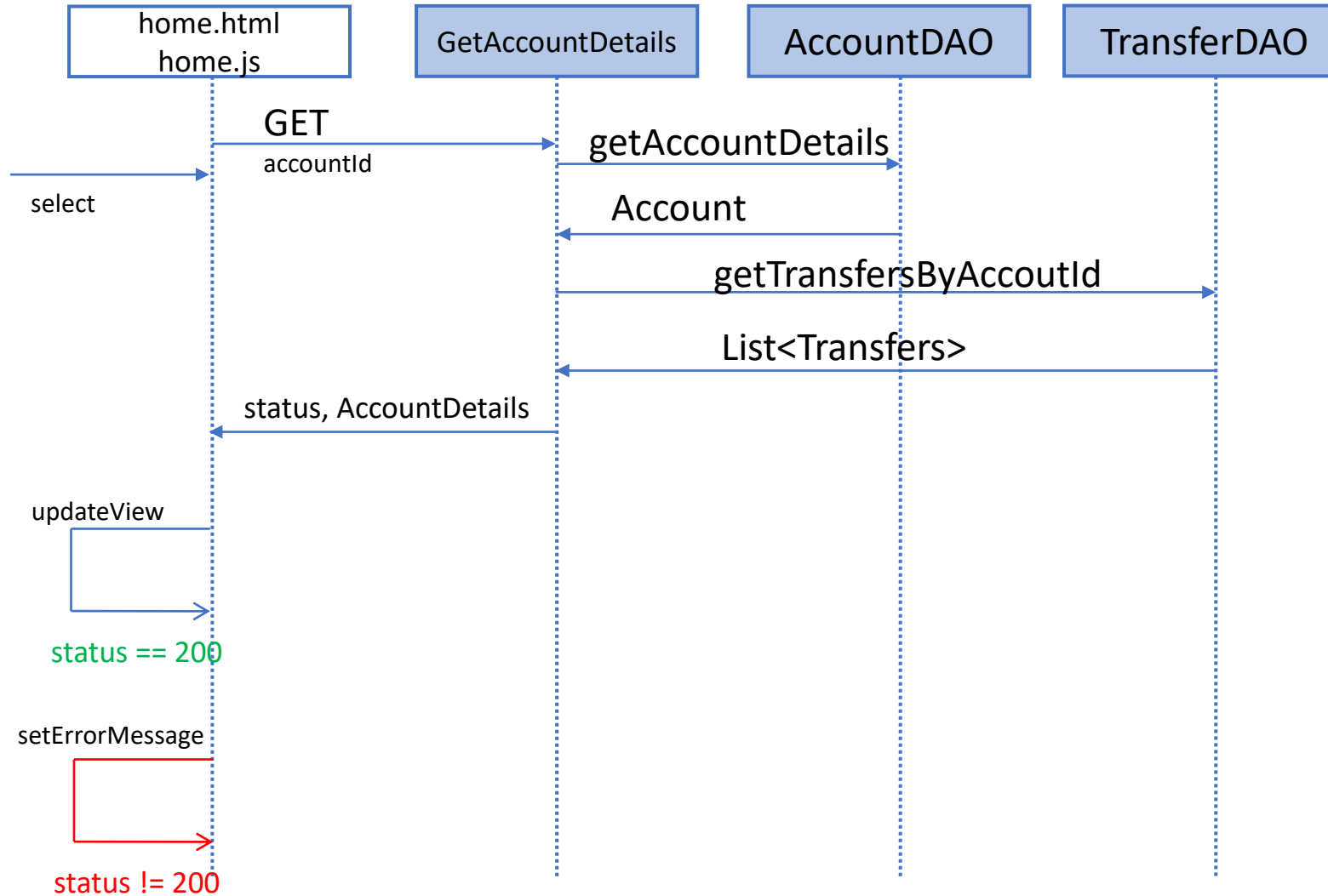
Evento: Registrazione



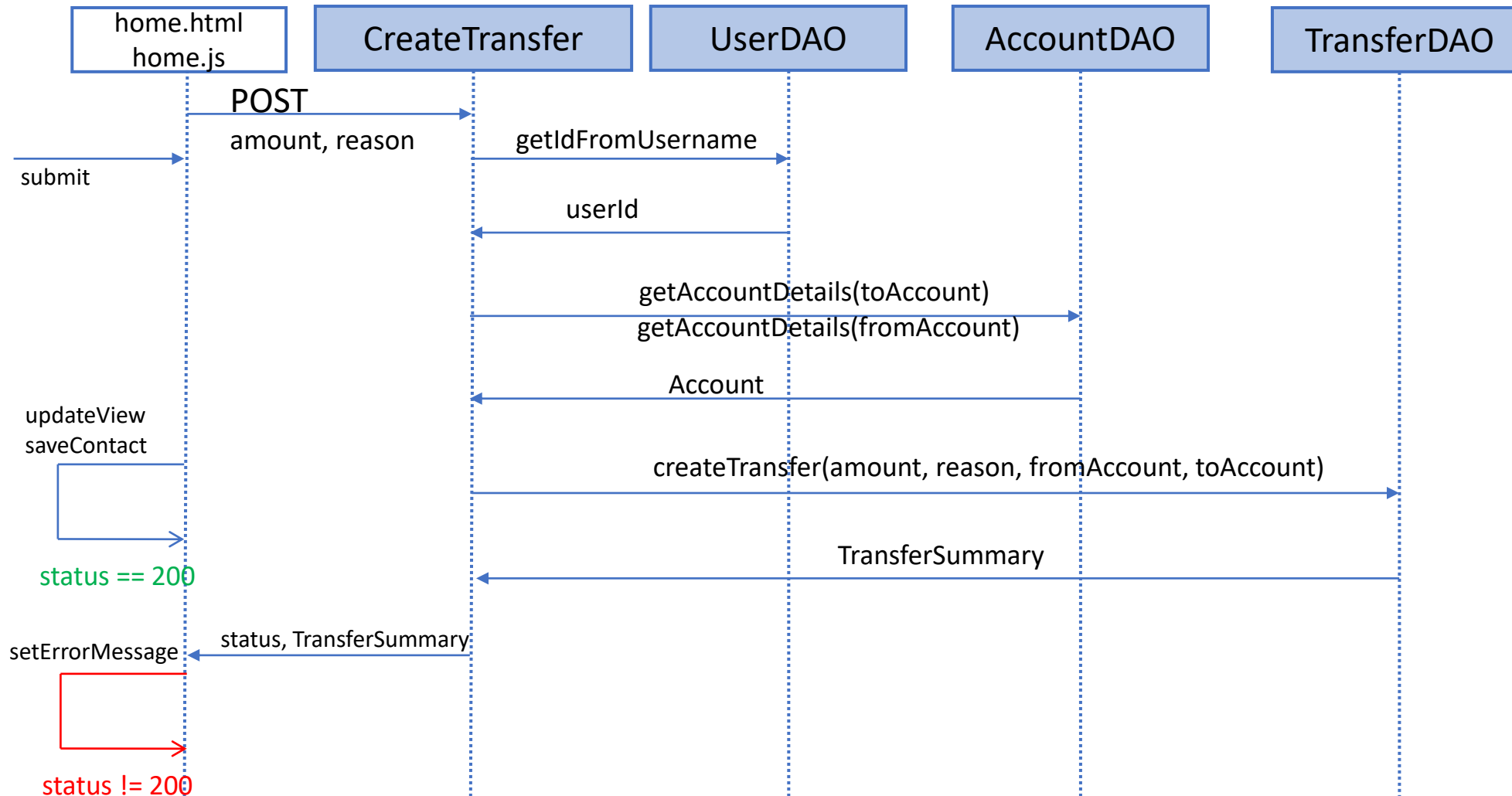
Evento: load Home page



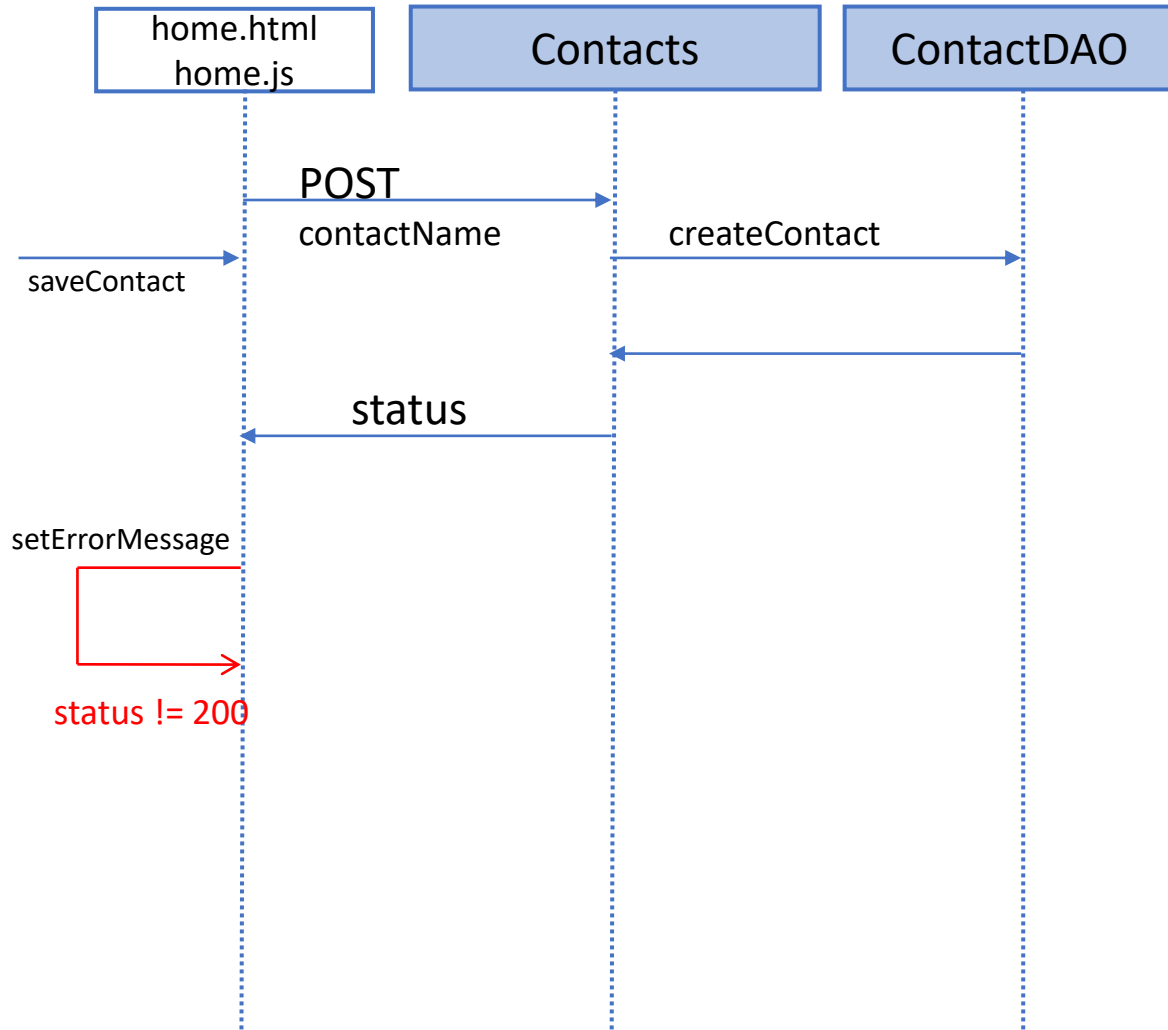
Evento: selezione conto corrente



Evento: invio form trasferimento



Evento: creazione contatto



Evento: logout

