



WEB TECHNOLOGIES EXAM

A.A. 2019-2020

Marco Petri

marco.petri@mail.polimi.it

Gruppo 62, Esercizio 4

Analisi dei requisiti sui dati

Un'applicazione web consente la gestione di trasferimenti di denaro online da un conto a un altro. Un **utente** ha un **nome**, un **codice** 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. In caso di verifica di entrambe le condizioni, l'applicazione deduce l'importo dal conto origine, aggiunge l'importo al conto destinazione e mostra una pagina CONFERMA TRASFERIMENTO che presenta i dati del conto di origine e destinazione, con i rispettivi saldi aggiornati. L'applicazione deve garantire l'atomicità del trasferimento: ogni volta che il conto di destinazione viene addebitato il conto di origine deve essere accreditato e viceversa.

Entità: **rosso**
Attributi: **verde**
Relazioni: **ciano**

Un primo sguardo ai requisiti presenti nel testo porta all'osservazione della presenza di diverse entità, mostrate a fianco in colore rosso.

L'informazione sui trasferimenti può essere progettata come entità debole, questa entità è infatti strettamente dipendente dal conto di origine e dal conto di destinazione, in ogni caso più di un trasferimento potrebbe avere lo stesso conto di origine e di destinazione nella stessa data. Per questa ragione l'identificazione della chiave primaria deve essere scelta di conseguenza, scegliendo ad esempio una chiave numerica per identificare un trasferimento.

Analisi dei requisiti applicativi

Un'applicazione web consente la gestione di trasferimenti di denaro online da un conto a un altro. Un utente ha un nome, un codice 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. In caso di verifica di entrambe le condizioni, l'applicazione deduce l'importo dal conto origine, aggiunge l'importo al conto destinazione e mostra una **pagina CONFERMA TRASFERIMENTO** che presenta i **dati del conto di origine e destinazione**, con i rispettivi saldi aggiornati. L'applicazione deve garantire l'atomicità del trasferimento: ogni volta che il conto di destinazione viene addebitato il conto di origine deve essere accreditato e viceversa.

Viste: **rosso**

Componenti visuali: **verde**

Eventi: **ciano**

Azioni: **viola**

Completamento della specifica

Gli utenti sono caratterizzati da un codice univoco autogenerato che li identifica all'interno del sito web per trasferimenti di denaro, questo codice univoco viene utilizzato anche per effettuare il login.

Un trasferimento è caratterizzato da una data dell'anno solare e deve controllare che la data inserita dall'utente sia corretta per evitare che quest'ultimo inserisca valori non validi.

Esiste una pagina comune senza necessità di autenticazione accessibile a chiunque che permette agli utenti registrati di effettuare l'autenticazione e accedere ai contenuti riservati.

Esiste una pagina comune senza necessità di autenticazione accessibile a chiunque che permette agli non utenti registrati di effettuare la registrazione al sito web e successivamente autenticarsi.

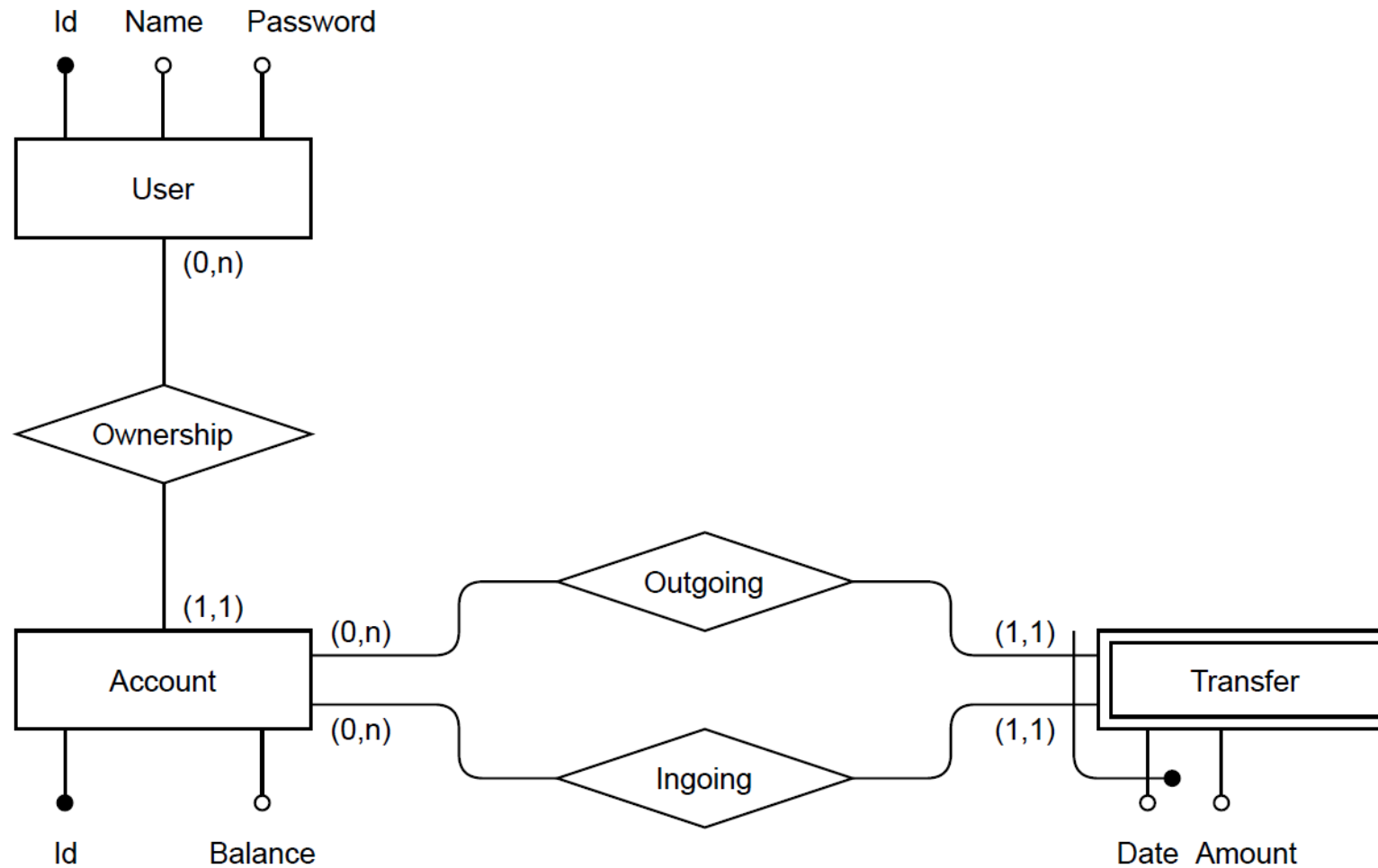
Esiste una pagina principale (home) che mostra l'elenco dei conti posseduti dall'utente autenticato che la visualizza, l'elenco è cliccabile e al click su uno dei conti viene aperta una pagina di stato del conto.

Esiste una pagina di stato del conto che mostra dettagli del conto e lista dei movimenti, essa contiene inoltre un form per l'ordinazione di trasferimenti monetari verso altri conti che invia una richiesta POST che viene verificata lato server.

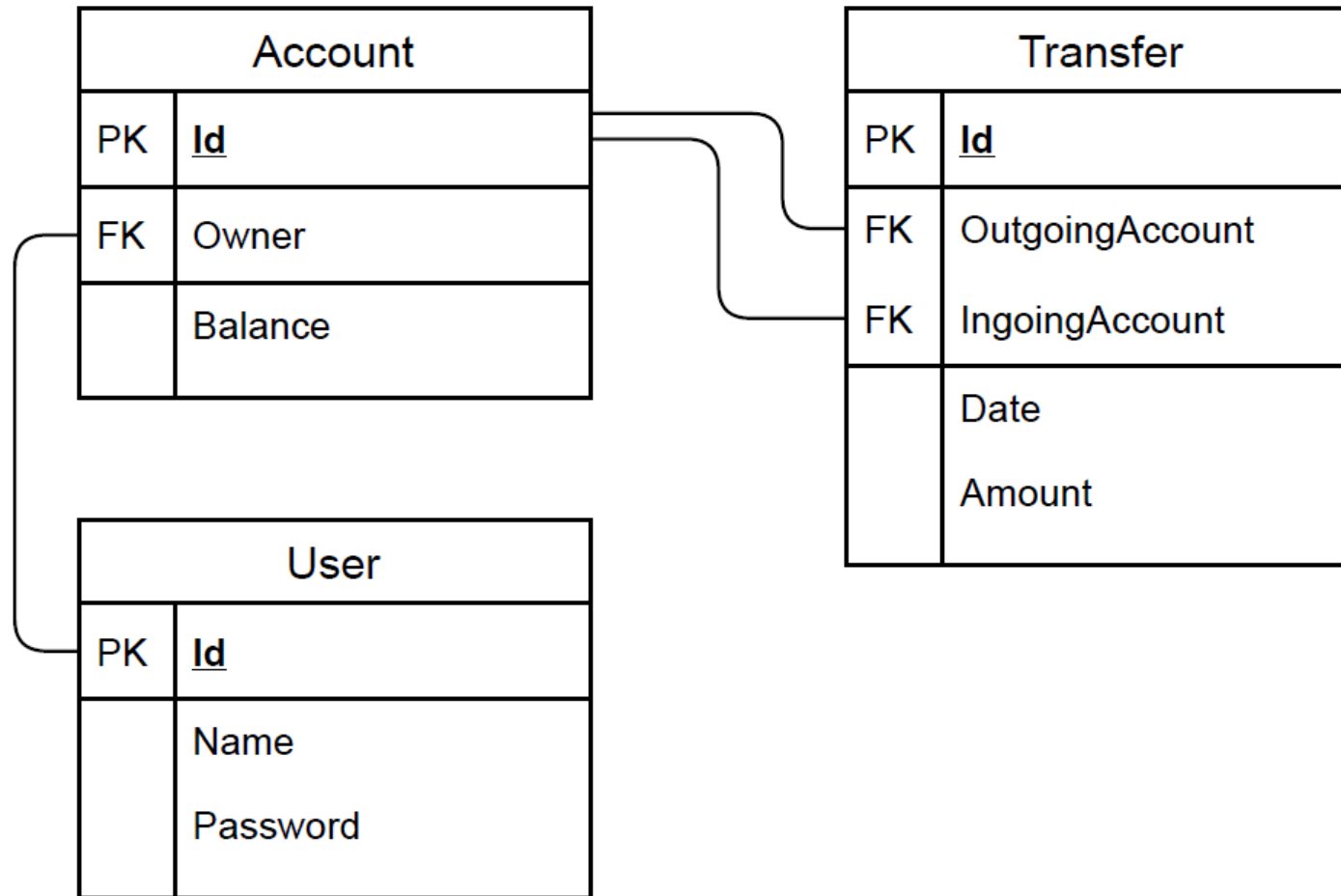
Esiste una pagina di fallimento che mostra le ragioni del fallimento dell'ordinazione del conto se questo non fosse ordinabile.

Esiste una pagina di conferma del trasferimento che riporta i dati dei conti di origine e destinazione con i rispettivi saldi aggiornati.

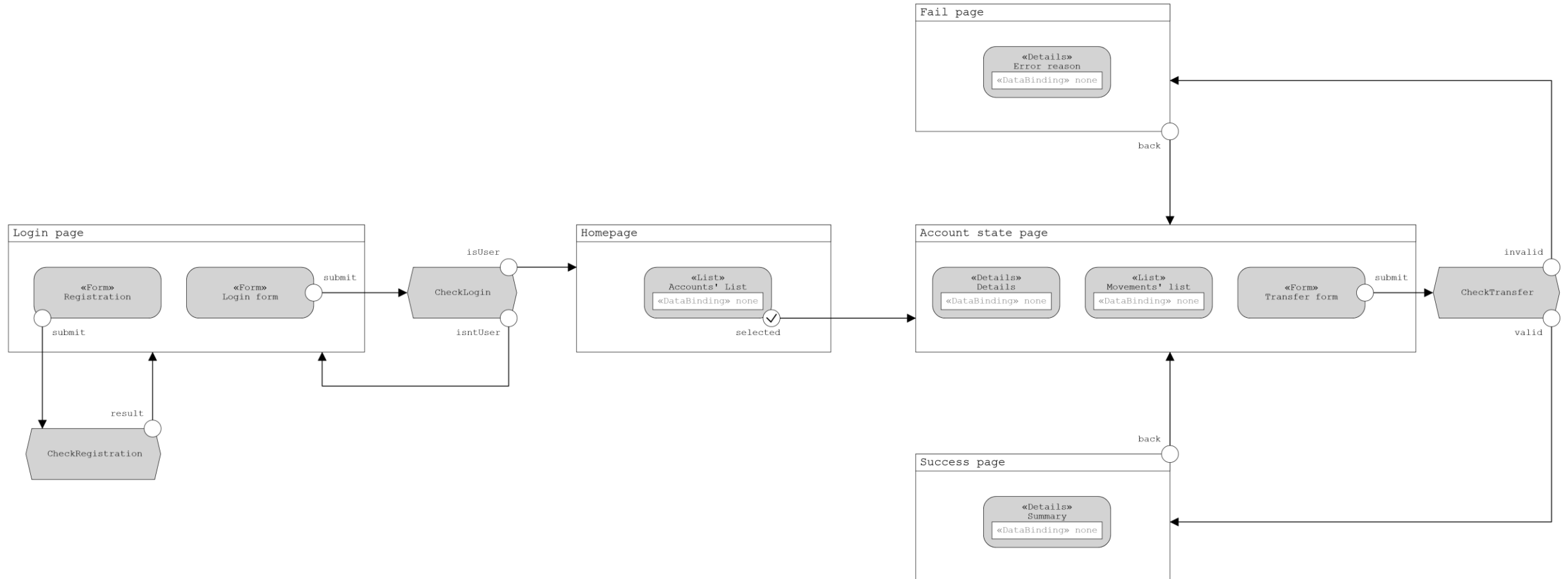
Diagramma ER del database



Progetto logico del database



Schema IFML dell'applicazione PURE HTML



Schema IFML dell'applicazione RIA

Componenti applicazione PURE HTML

Beans:

- User
- Account
- Transfer

DAO:

- UserDAO
 - + findUser
 - + findAccounts
 - + createUser
 - + modifyPassword
- AccountDAO
 - + findAccount
 - + findOutTransfers
 - + findInTransfers
 - + sumAmount
 - + subAmount

TransferDAO

- + findTransfer
- + createTransfer

Controllers:

- CheckRegistration
- CheckLogin
- CheckTransfer
- GetLogin
- GetHomepage
- GetAccountState
- GetSuccess
- GetFail

Filters:

- UserFilter
- AccountFilter

Views:

- Login page
- Home page
- Account state page
- Success page
- Fail page

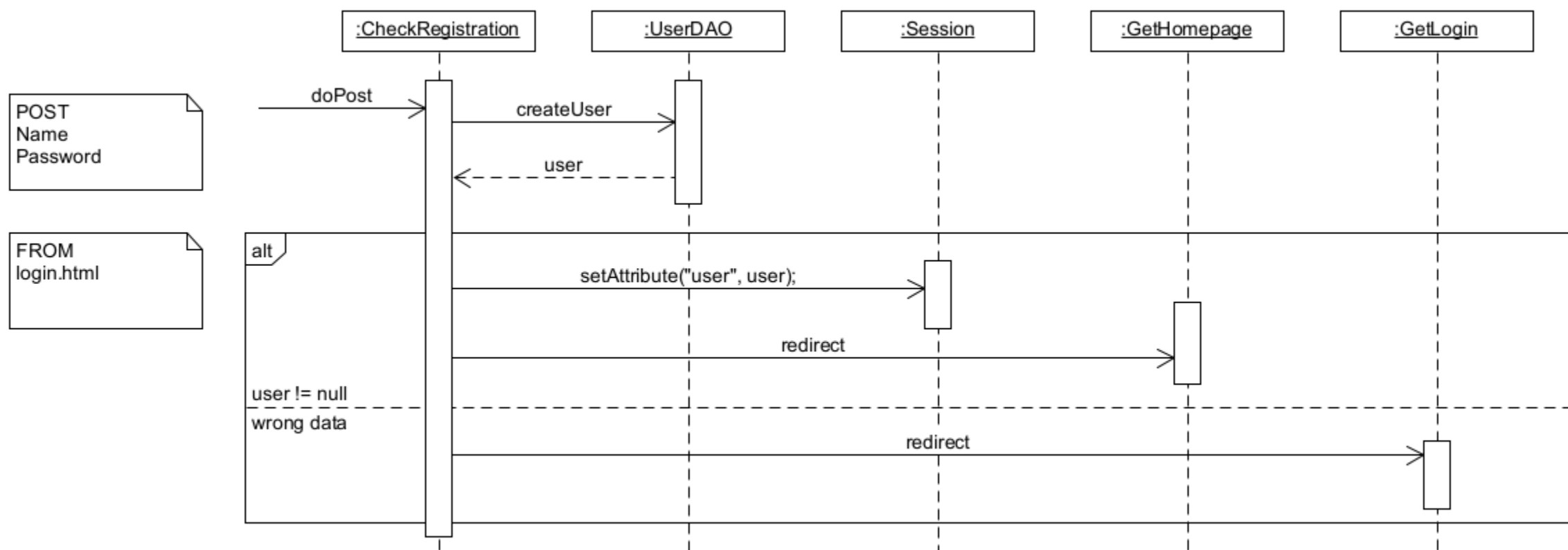


Componenti applicazione RIA

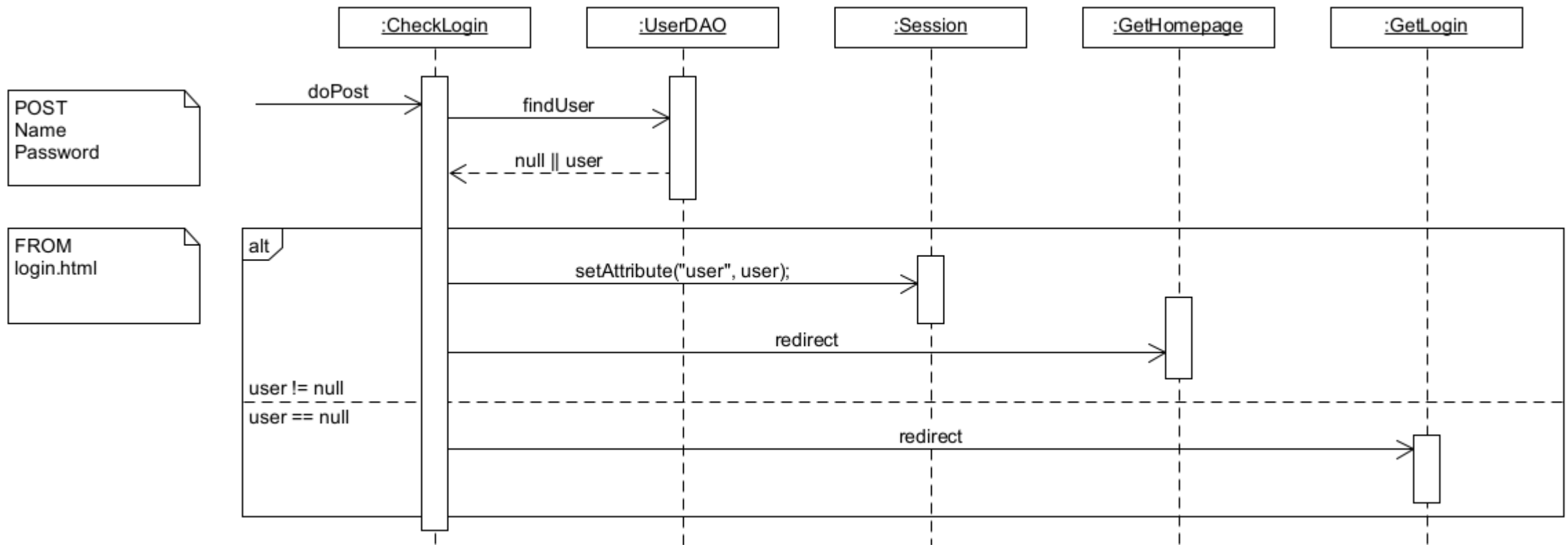


Diagrammi di sequenza PURE HTML

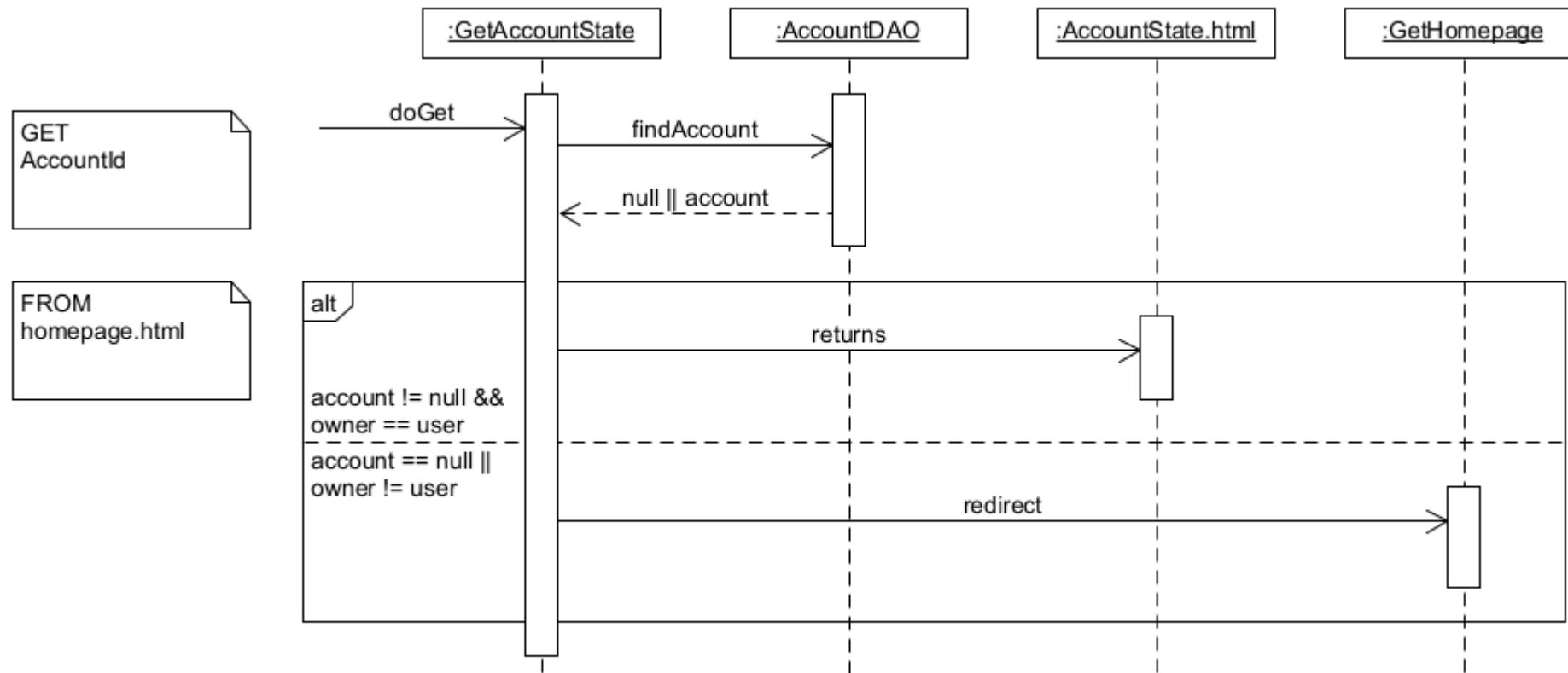
Registrazione



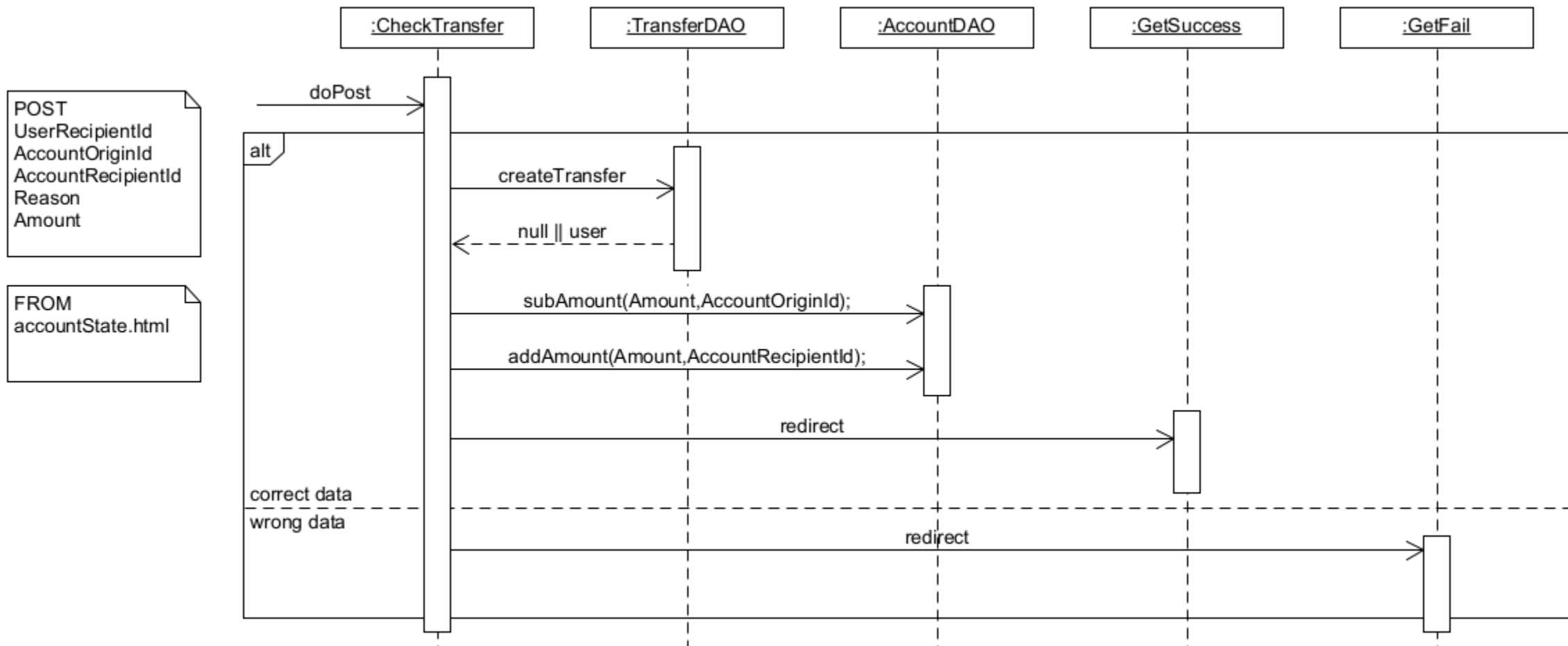
Accesso



Selezione di un conto



Ordinazione trasferimento



The image features decorative curved lines in the top-left and top-right corners. These lines are composed of multiple overlapping layers in shades of light green and light blue, creating a sense of depth and movement.

Diagrammi di sequenza RIA



Componenti applicazione RIA