

Money Transfer Project

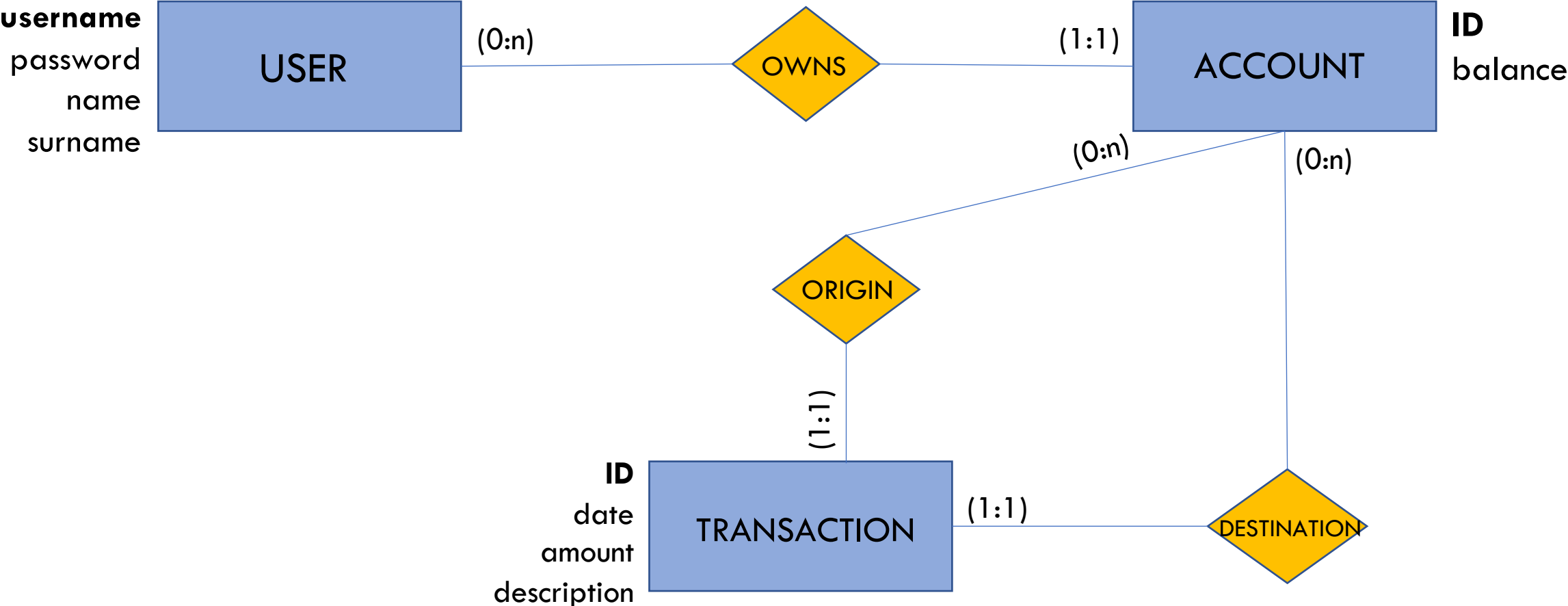
Web Technologies

Data Analysis

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.

Entities, **attributes**, **relationships**

Database Design

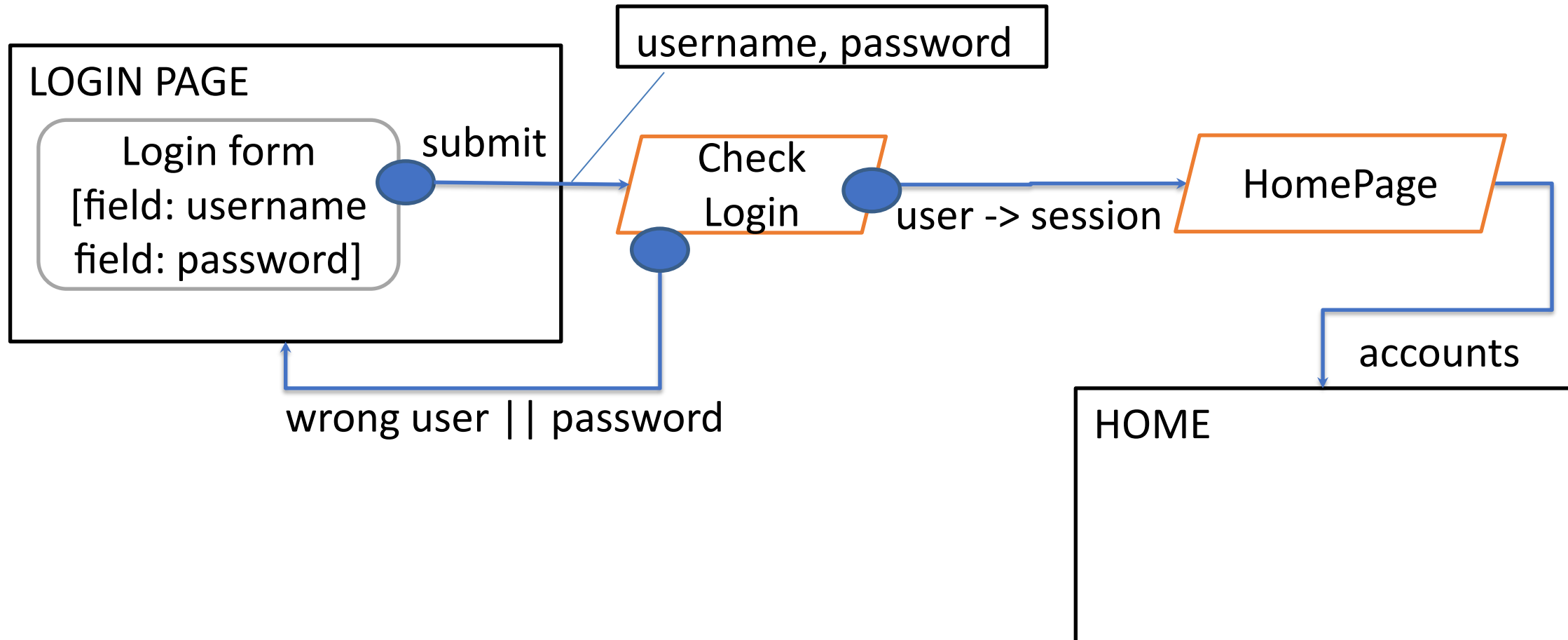


Application Requirement Analysis

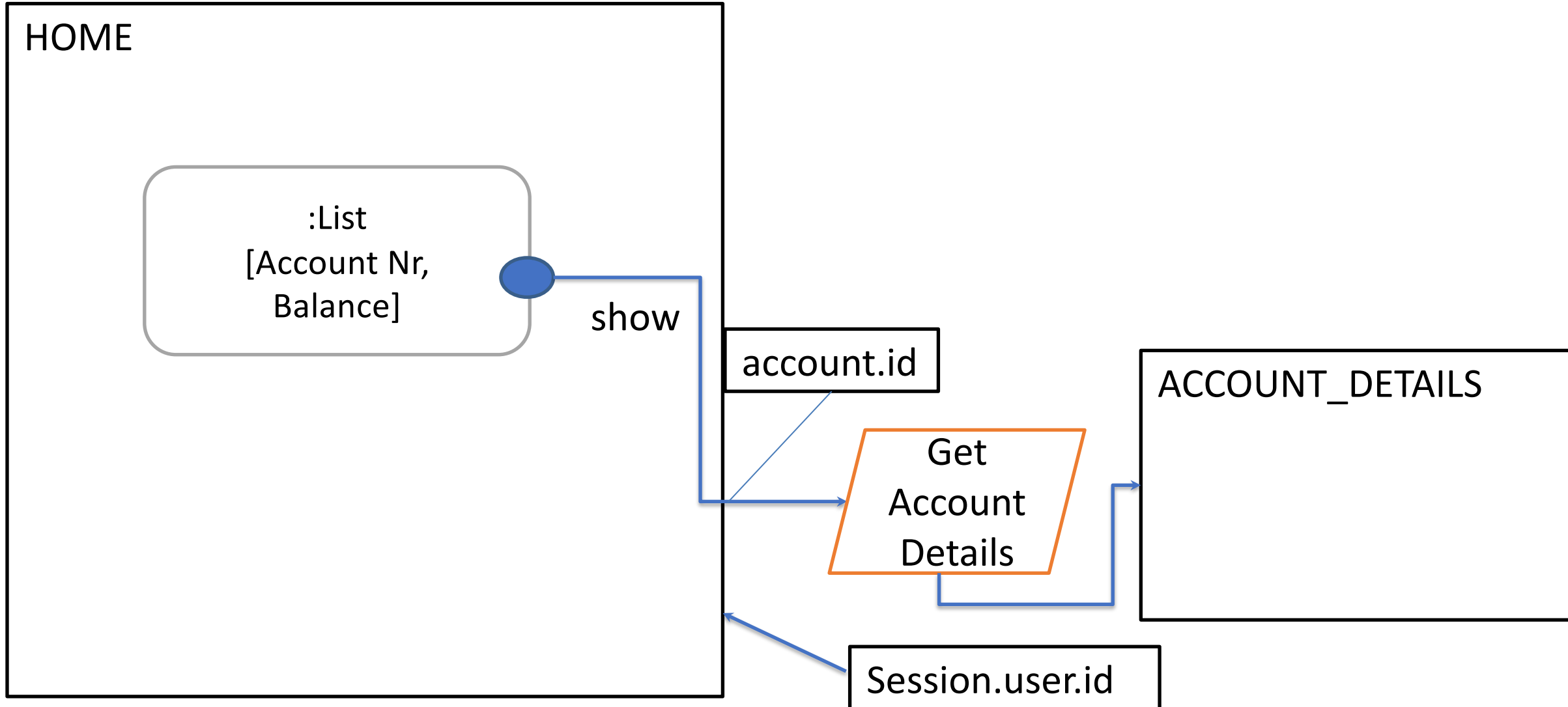
Un'applicazione web consente la gestione di trasferimenti di denaro online da un conto a un altro. Un utente ha un nome, un codice e uno o più conti correnti. Un conto ha un codice, un saldo, e i trasferimenti fatti (in uscita) e ricevuti (in ingresso) dal conto. Un trasferimento ha una data, un importo, 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 **botone 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.

Pages (views), view components, events, actions

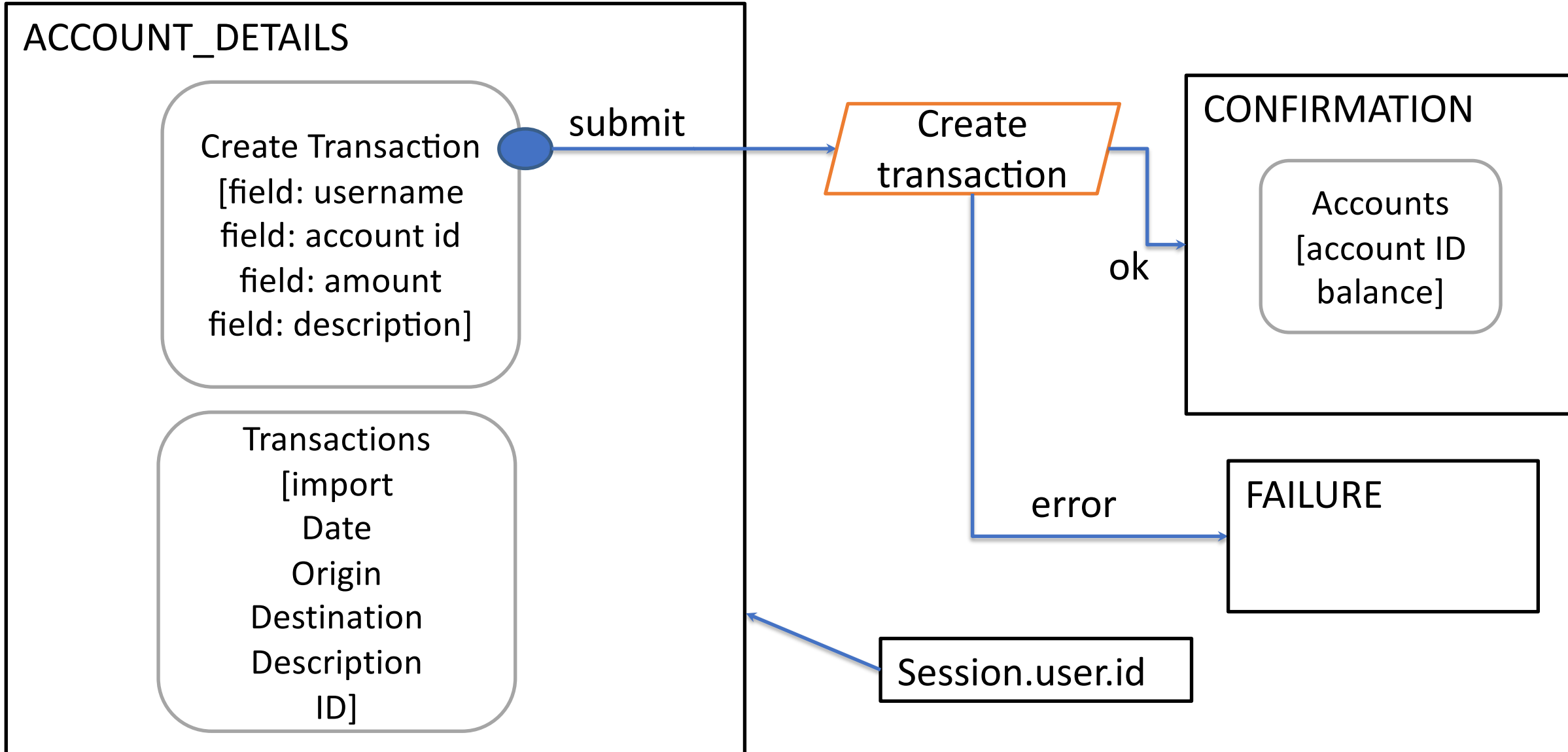
Application Design - Login



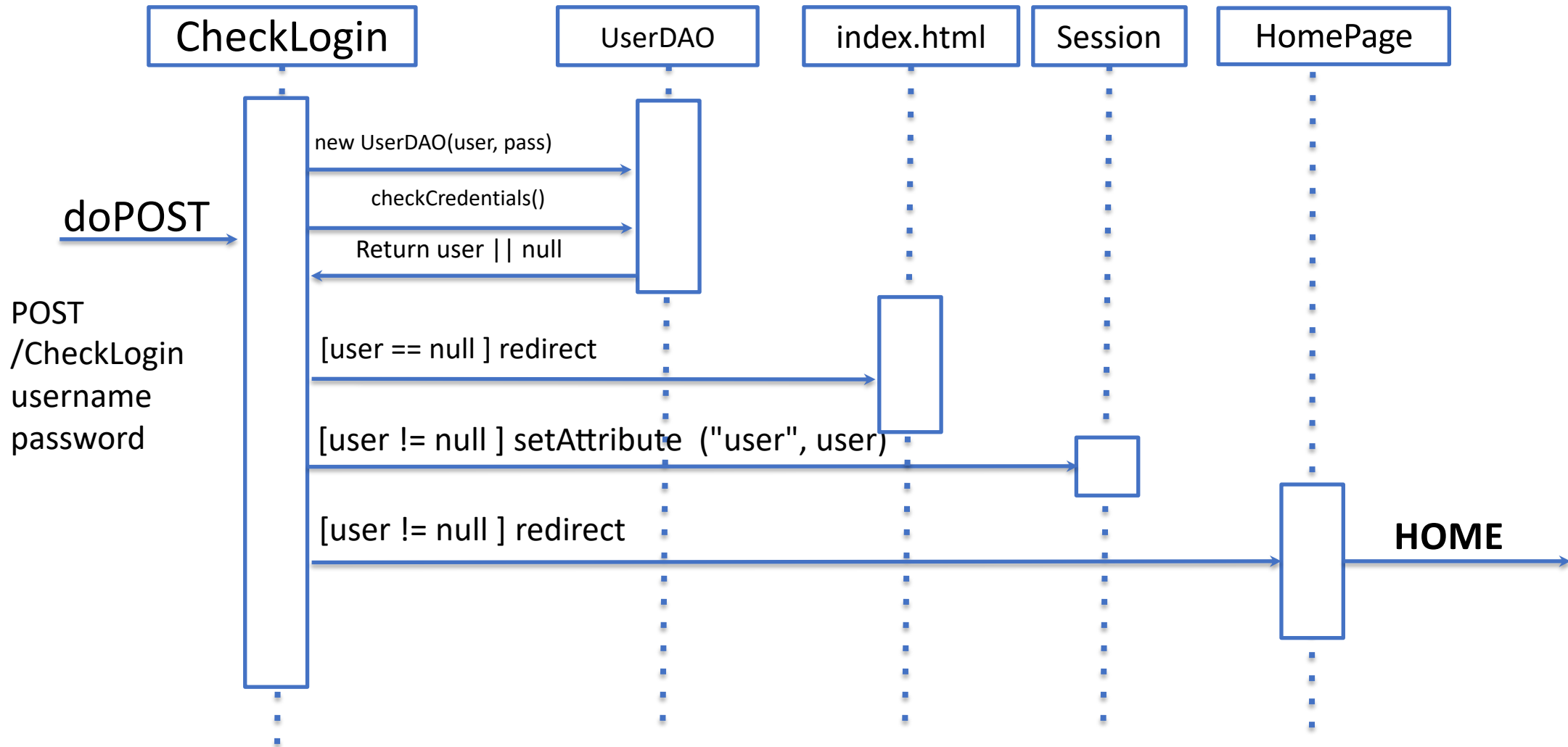
Application Design - Home



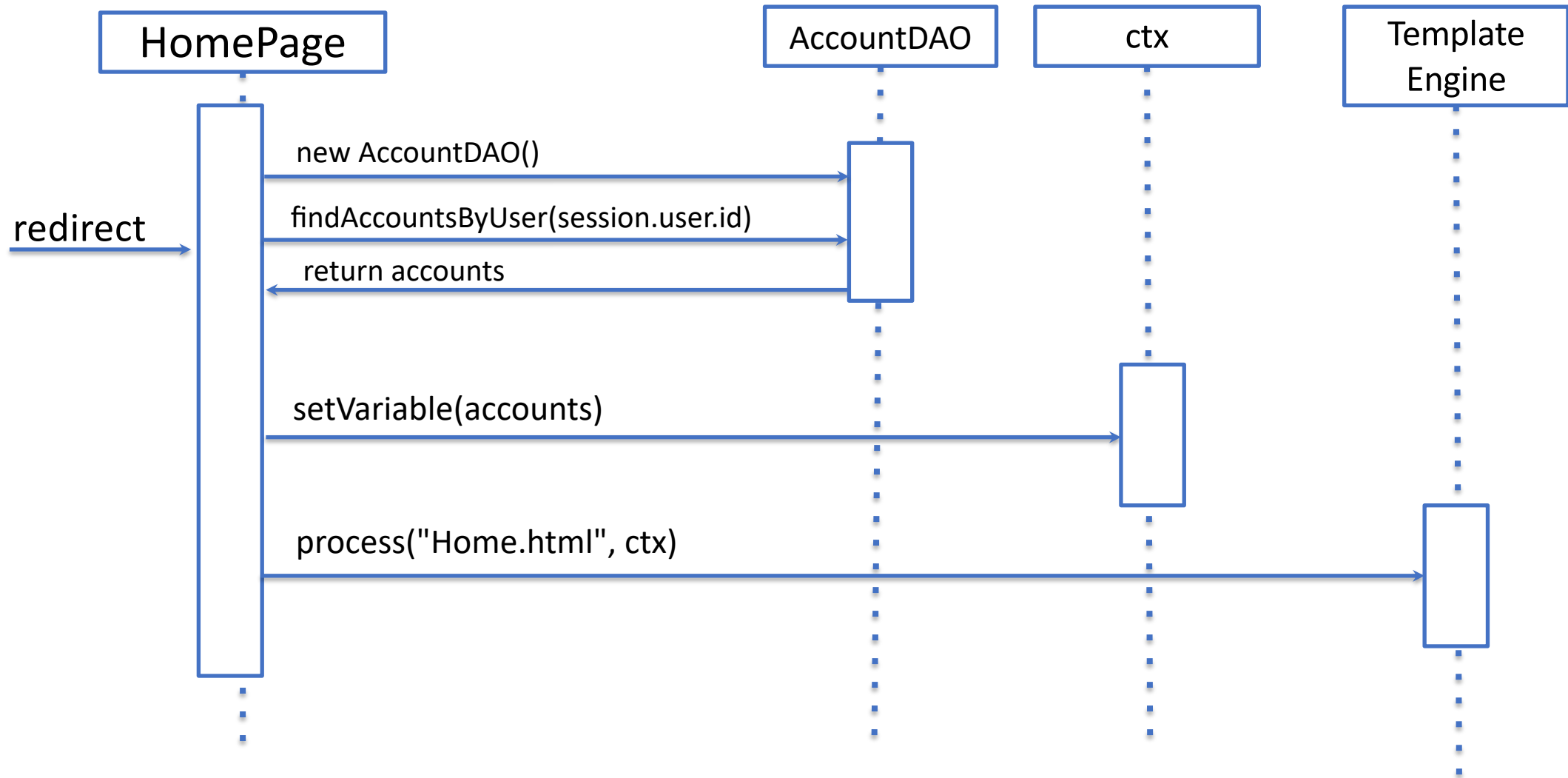
Application Design – Account Details



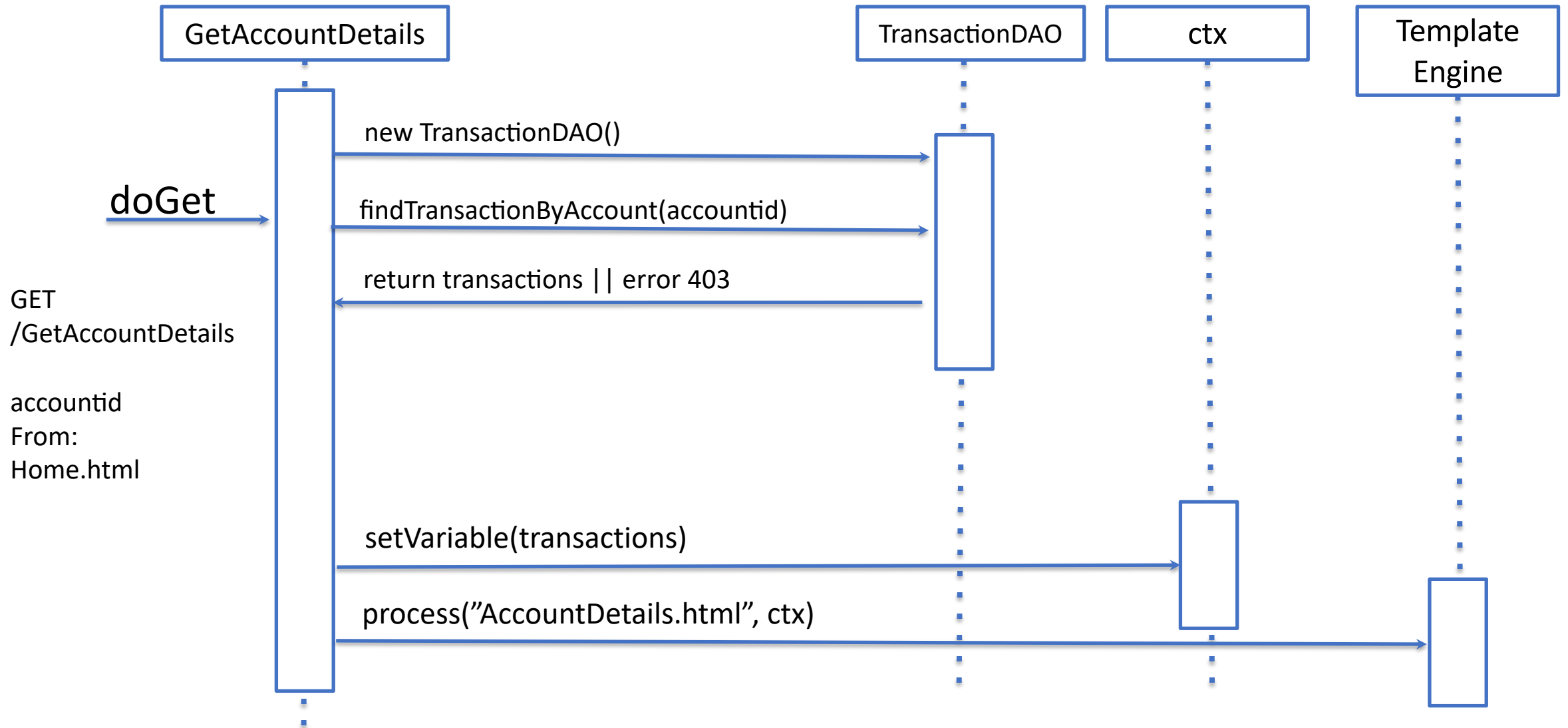
Login Event



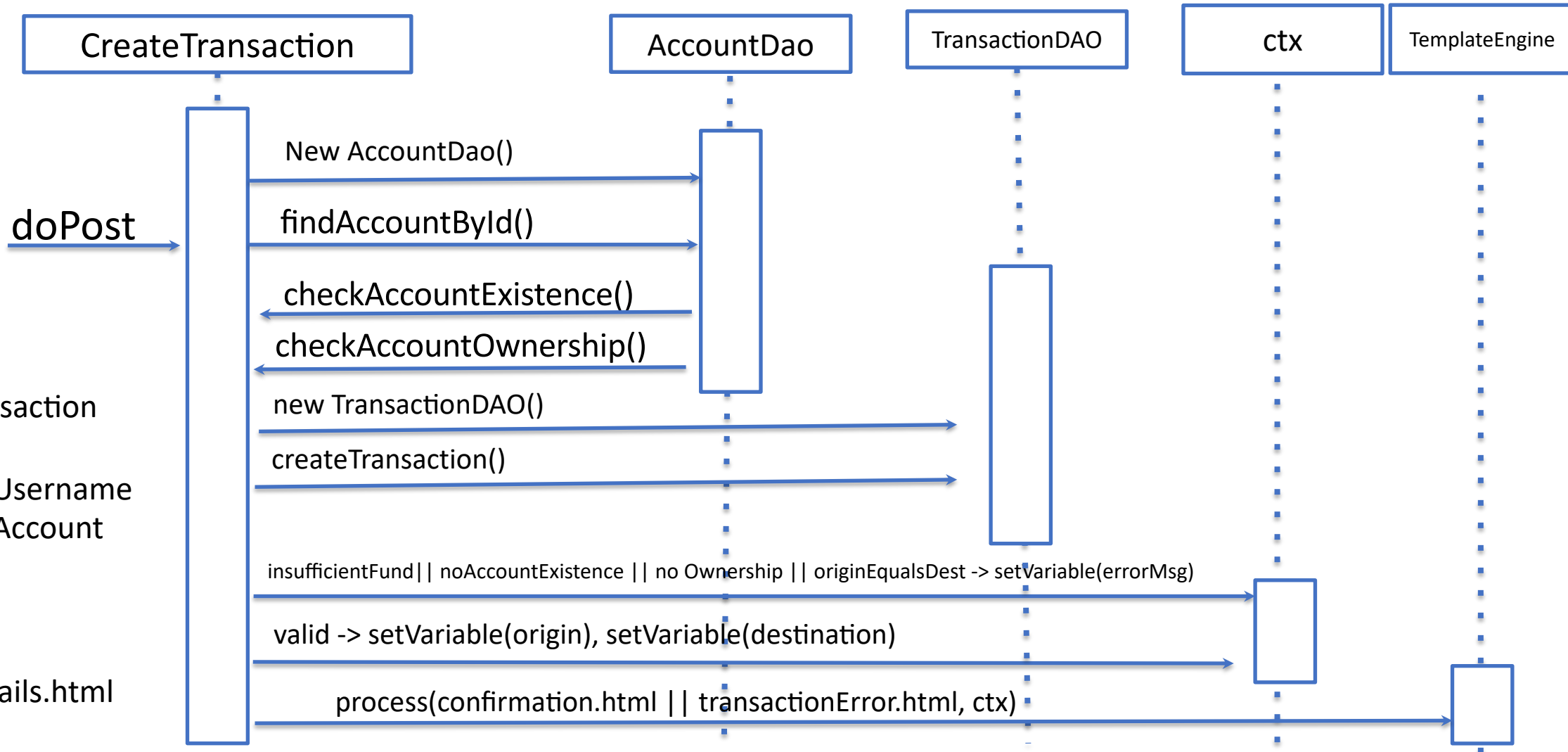
Event: HomePage



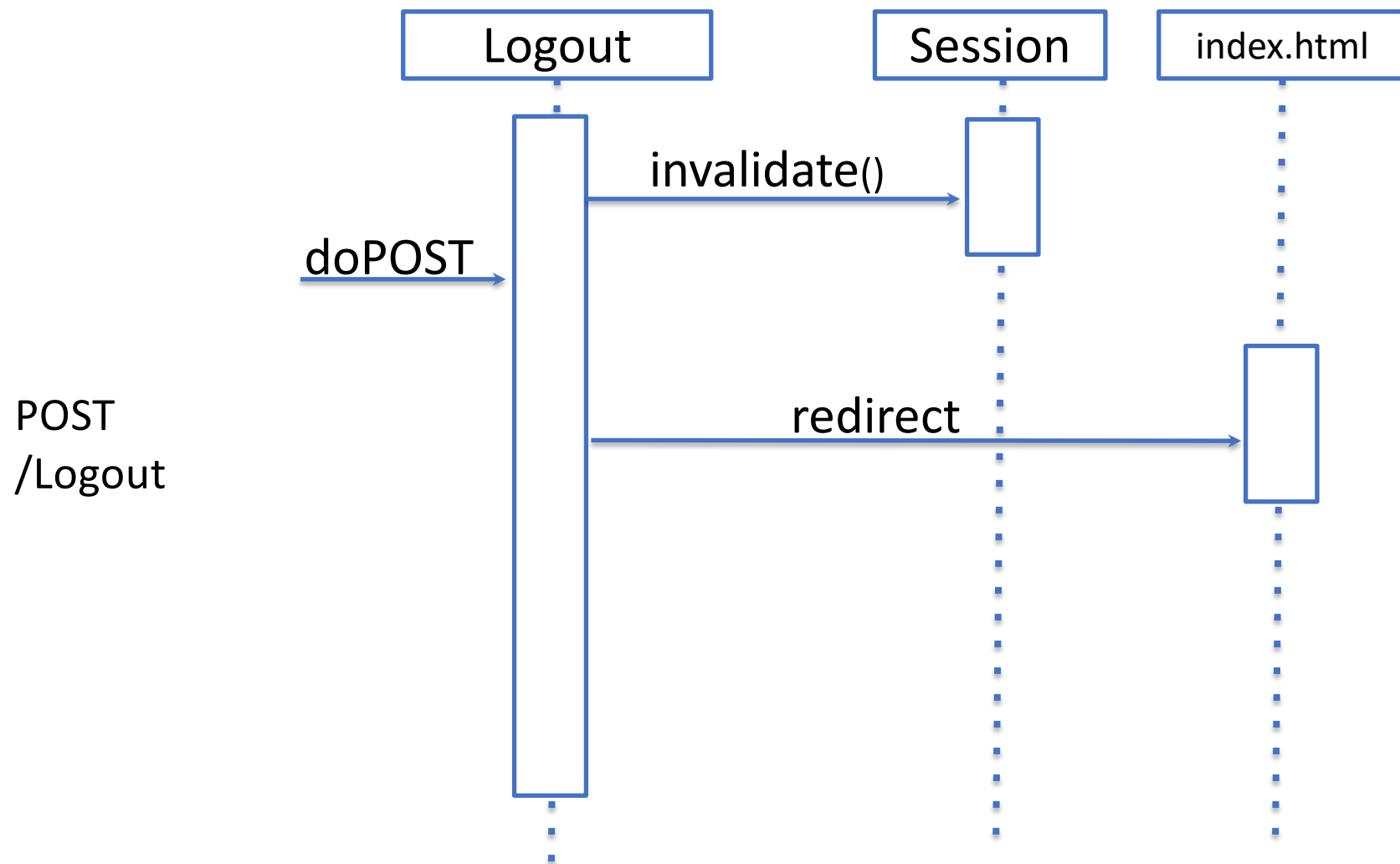
Event: GetAccountDetails



Event: CreateTransaction



Event: Logout



Event: sort Accounts list

