

Home Banking

Livello Base

Scrivere un'applicazione Web che consente di gestire un servizio di "home banking".

L'applicazione deve consentire di eseguire il seguente caso d'uso.

"Utente effettua versamento"

Scenario Principale

- L'utente accede ad uno schermo iniziale (index.jsp) nel quale viene visualizzata una maschera per specificare i dati del conto corrente. Nella maschera è necessario specificare il codice del conto corrente (un numero intero)
- Il sistema verifica che nella base di dati esista un conto corrente con il codice specificato. I conti correnti sono memorizzati in una tabella ContiCorrenti della base di dati che contiene il numero del conto, il nome e cognome dell'utente, il saldo in Euro.
- Se il conto esiste, l'utente accede ad uno schermo (versamento.jsp) nel quale sono riassunti i dati del conto corrente; lo schermo contiene inoltre una maschera attraverso la quale l'utente fornisce l'importo da versare
- Il sistema aggiorna il saldo del conto effettuando il versamento
- L'utente accede ad uno schermo (conferma.jsp) nel quale vengono riassunti i risultati dell'operazione (dati del conto ed importo versato); lo schermo contiene un collegamento allo schermo iniziale nel caso l'utente voglia eseguire una ulteriore operazione

Scenario alternativo: Il conto corrente non esiste

- Se, viceversa, il conto corrente non esiste, l'utente accede ad uno schermo (errore.jsp) in cui viene visualizzato un messaggio di errore ed un collegamento allo schermo principale per effettuare una nuova operazione

Livello Intermedio

Scrivere un'applicazione Web che consente di gestire un servizio di "home banking".

L'applicazione deve consentire di eseguire i seguenti caso d'uso.

"Utente effettua login"

Scenario Principale

- L'utente accede ad uno schermo iniziale (index.jsp) nel quale viene visualizzata una maschera per fornire nome utente e password

- Il sistema autentica l'utente accedendo ad una tabella Utenti della base di dati, che contiene il nome utente, la password, il nome e cognome dell'utente e il suo codice di conto corrente (NOTA: per semplicità si suppone che ciascun utente possa avere al massimo un conto corrente)
- Se il conto esiste, l'utente accede allo schermo principale del caso d'uso successivo (versamento.jsp)

Scenario alternativo: nome utente o password scorretta

- Se l'autorizzazione fallisce, l'utente ritorna allo schermo iniziale (index.jsp) in cui viene visualizzato un messaggio di errore

"Utente effettua versamento"

Scenario Principale

- L'utente accede ad uno schermo (versamento.jsp) nel quale sono riassunti i dati dell'utente (nome, cognome, numero di conto corrente, saldo attuale). I dati relativi ai conti correnti sono memorizzati in una tabella ContiCorrenti della base di dati che contiene (a) il codice del conto; (b) il saldo in Euro. Lo schermo contiene una maschera attraverso cui l'utente fornisce l'importo in Euro da versare; l'importo minimo che è possibile versare è di 100 Euro
- Il sistema convalida i dati forniti; se i dati sono corretti il sistema aggiorna il saldo del conto effettuando il versamento
- L'utente accede ad uno schermo (conferma.jsp) nel quale vengono riassunti i risultati dell'operazione (dati del conto aggiornati ed importo versato); lo schermo contiene inoltre (a) un collegamento allo schermo iniziale del caso d'uso (versamento.jsp) nel caso l'utente voglia eseguire una ulteriore operazione; (b) un collegamento ad uno schermo di uscita (uscita.jsp) attraverso il quale l'utente può abbandonare l'applicazione

Scenario alternativo: importo scorretto

- Se la convalida fallisce, l'utente ritorna allo schermo di immissione (versamento.jsp) in cui viene visualizzato un messaggio di errore

Livello avanzato

Scrivere un'applicazione Web che consente di gestire un servizio di "home banking".

L'applicazione deve consentire di eseguire i seguenti caso d'uso.

"Utente effettua login"

Scenario Principale

- L'utente accede ad uno schermo iniziale (index.jsp) nel quale viene visualizzata una maschera per fornire nome utente e password
- Il sistema autentica l'utente accedendo ad una tabella Utenti della base di dati, che contiene il nome utente, la password, il nome e cognome dell'utente e il suo codice di conto corrente (NOTA: per semplicità si suppone che ciascun utente possa avere al massimo un conto corrente)

- Se il conto esiste, l'utente accede allo schermo principale del caso d'uso successivo (bonifico.jsp)

Scenario alternativo: dati scorretti

- Se l'autorizzazione fallisce, l'utente ritorna allo schermo iniziale (index.jsp) in cui viene visualizzato un messaggio di errore

"Utente effettua bonifico"

Scenario Principale

- L'utente accede ad uno schermo (bonifico.jsp) nel quale sono riassunti i dati dell'utente (nome, cognome, numero di conto corrente, saldo attuale). I dati relativi ai conti correnti sono memorizzati in una tabella ContiCorrenti della base di dati che contiene (a) il codice del conto; (b) il saldo in Euro. Lo schermo contiene una maschera attraverso cui l'utente fornisce i dati del bonifico da effettuare, ovvero (a) codice del conto corrente di destinazione (un numero intero); NOTA: il conto corrente deve appartenere ad un altro utente della stessa banca; (b) importo da bonificare in Euro
- Il sistema convalida i dati forniti per verificare che il bonifico sia possibile; in questo caso registra il bonifico nella base di dati; i bonifici sono memorizzati nella tabella Bonifici che contiene il codice del conto di partenza, il codice del conto di destinazione e l'importo del bonifico. Inoltre, il sistema aggiorna i saldi dei due conti
- L'utente accede ad uno schermo (conferma.jsp) nel quale vengono riassunti i risultati dell'operazione (dati dell'utente e dati del bonifico effettuato); lo schermo contiene inoltre (a) un collegamento allo schermo iniziale del caso d'uso (bonifico.jsp) nel caso l'utente voglia eseguire una ulteriore operazione; (b) un collegamento ad uno schermo di uscita (uscita.jsp) attraverso il quale l'utente può abbandonare l'applicazione

Scenario alternativo: dati scorretti per il bonifico

- Se la convalida fallisce, l'utente ritorna allo schermo di immissione (bonifico.jsp) in cui vengono visualizzati i necessari messaggi di errore