

Prova d'Esame

Cognome e Nome _____ Matricola _____

Quesito svolto base ? intermedio ? avanzato ?

Tempo a Disposizione: 4 ore e mezza

Master Mind

Livello Base

E' necessario scrivere un'applicazione Web che consente di giocare ad una versione semplificata del gioco Master Mind. Il gioco consiste nell'indovinare una combinazione di numeri. L'applicazione deve consentire di eseguire i seguenti casi 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 (**NOTA**: è necessario prevedere tra gli altri un utente “Mario Rossi” con nome utente **utente** e password **utente**)
- Se l'autenticazione ha successo, l'utente accede allo schermo principale del caso d'uso successivo (**gestione.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 inserisce combinazione”

Scenario Principale

- L'utente accede ad uno schermo iniziale (**gestione.jsp**) nel quale sono riassunti i dati dell'utente e viene visualizzata una maschera per inserire i dati della combinazione da inserire, ovvero: (a) il codice della combinazione; (b) le cifre della combinazione. Lo schermo contiene inoltre un collegamento per raggiungere lo schermo finale (**uscita.jsp**).
- Sottomettendo la maschera, l'utente accede ad uno schermo (**conferma.jsp**) nel quale sono riassunti i dati alla combinazione inserita; lo schermo contiene un bottone per confermare la scelta e un collegamento per tornare alla pagina precedente (**gestione.jsp**) e correggere la combinazione fornita.
- Decidendo di confermare la scelta, il sistema registra la combinazione in una tabella **Combinazioni** della base di dati in cui viene memorizzata (a) il codice della combinazione; (b) la combinazione (x) il riferimento all'utente che ha effettuato l'inserimento. L'utente accede allo schermo finale (**uscita.jsp**) nel quale è disponibile un collegamento allo schermo iniziale (**gestione.jsp**) nel caso l'utente voglia inserire una ulteriore combinazione

Scenario alternativo: La stessa combinazione esiste già nella base di dati

- 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

Scrivere l'applicazione Web che effettua le operazioni elencate sopra secondo le seguenti specifiche:

- L'applicazione deve utilizzare pagine JSP e un'architettura di tipo **Modello 1**.
- La grafica deve essere organizzata utilizzando un foglio di stile **CSS**.
- Non è necessario provvedere alla convalida dei dati sottomessi dall'utente.
- Non è necessario provvedere alla convalida del codice HTML prodotto dalle pagine JSP.

Livello Intermedio

E' necessario scrivere un'applicazione Web che consente di giocare ad una versione estremamente semplificata del gioco Master Mind. Il gioco consiste nell'indovinare una combinazione segreta fatta di 4 numeri compresi tra 1 e 6 scelta dal calcolatore. Il giocatore tenta di indovinare la combinazione nascosta effettuando dei tentativi. Ad ogni tentativo il computer fornisce una risposta. La risposta dice quante cifre della combinazione nascosta sono state azzeccate dal giocatore (per semplicità consideriamo solo cifre uguali in uguale posizione). Es: se la combinazione nascosta è 4261, il tentativo 1234 riceve come risposta "1 cifra individuata" (corrispondente al 2); il tentativo 2241 riceve come risposta "2 cifre individuate" (il 2 e l'1); il tentativo 2416 riceve come risposta "0 cifre individuate"). L'applicazione deve consentire di eseguire il seguente caso d'uso.

“Utente indovina combinazione”

Scenario Principale

- L'utente accede ad uno schermo iniziale (**index.jsp**) che contiene una maschera attraverso la quale l'utente fornisce il nome con cui vuole giocare.
- Sottomettendo la maschera, il calcolatore provvede a caricare la combinazione segreta di 4 cifre dalla tabella **Combinazione** della base di dati. Da quel momento in poi, il giocatore può cominciare a giocare effettuando tentativi e verificando le risposte del computer.
- L'utente accede allo schermo iniziale del gioco (**giocata.jsp**) nel quale viene visualizzato (a) il nome del giocatore (b) la lista dei tentativi effettuati fino a quel punto dal giocatore e (c) le relative risposte fornite dal computer. La pagina contiene una maschera attraverso la quale è possibile effettuare un nuovo tentativo. Infine, la pagina contiene un pulsante per uscire dal gioco.
- Sottomettendo la maschera, il tentativo del giocatore viene convalidato per verificare che sia fatto di cifre comprese tra 1 e 6. Se il tentativo è corretto ma la combinazione segreta non è stata indovinata viene visualizzato uno schermo di risposta (**risposta.jsp**) nel quale viene confermato il tentativo effettuato e viene fornita la risposta del calcolatore. La pagina contiene inoltre un collegamento per ritornare allo schermo di gioco (**giocata.jsp**).
- Se il tentativo è corretto e la combinazione segreta è stata indovinata, l'applicazione registra nella tabella **Partite** l'esito della partita, ovvero (a) la combinazione da individuare; (b) il numero di tentativi effettuati; (c) il nome del giocatore. Il giocatore accede allo schermo di successo (**indovinato.jsp**) che contiene un messaggio con cui viene segnalato che il gioco è finito, con un riassunto dei tentativi effettuati e delle risposte. La pagina inoltre contiene un collegamento per cominciare una nuova partita (**index.jsp**) ed uno per uscire dal gioco (**uscita.jsp**).
- Se l'utente decide di uscire dal gioco viene visualizzato lo schermo finale (**uscita.jsp**), che contiene un messaggio di saluto e un collegamento per raggiungere la pagina iniziale dell'applicazione (**index.jsp**) e ricominciare daccapo.

Scenario alternativo: Combinazione scorretta

- L'utente ritorna allo schermo di immissione (**giocata.jsp**) in cui viene visualizzato un messaggio di errore.

Scrivere l'applicazione Web che effettua le operazioni elencate sopra secondo le seguenti specifiche:

- L'applicazione deve utilizzare pagine JSP e un'architettura di tipo **Modello 1** oppure un'architettura di tipo **Modello 2** basata sull'utilizzo del framework didattico **pinco**
- E' necessario provvedere alla **convalida** dei dati sottomessi dall'utente
- La grafica deve essere organizzata utilizzando un foglio di stile **CSS**
- Non è necessario provvedere alla convalida del codice HTML prodotto dalle pagine JSP
- Per la gestione della base di dati è necessario utilizzare **PostgreSQL**, e il relativo driver **JDBC**

Livello avanzato

E' necessario scrivere un'applicazione Web che consente di giocare ad una versione semplificata del gioco Master Mind. Il gioco consiste nell'indovinare una combinazione segreta fatta di numeri compresi tra 1 e 6 scelta dal calcolatore. Per semplicità si supponga che i numeri siano tutti diversi. Il giocatore tenta di indovinare la combinazione nascosta effettuando dei tentativi. Ad ogni tentativo il computer fornisce una risposta. Le risposte sono fatte di pallini neri e pallini bianchi, secondo queste regole: (a) un pallino nero per ciascuna cifra del tentativo che corrisponde in valore e posizione ad una cifra della combinazione nascosta; (b) un pallino bianco per ciascuna cifra del tentativo che corrisponde in valore ma non in posizione ad una cifra della combinazione nascosta (es: se la combinazione nascosta è 4261, il tentativo 1234 riceve come risposta 1 pallino nero (corrispondente al 2) e 2 pallini bianchi (corrispondenti all'1 e al 4); il tentativo 3512 riceve 0 pallini neri e 2 pallini bianchi; il tentativo 4216 riceve 2 pallini neri e 2 pallini bianchi). L'applicazione deve consentire di eseguire il seguente caso d'uso.

“Utente indovina combinazione”

Scenario Principale

- L'utente accede ad uno schermo iniziale (**index.jsp**) che contiene una maschera attraverso la quale l'utente fornisce il nome con cui vuole giocare.
- Sottomettendo la maschera, il calcolatore provvede a generare la combinazione segreta di 4 cifre, facendo in modo che ciascuna cifra compaia una sola volta (es: la combinazione 4261 è accettabile, mentre la combinazione 2113 non lo è). Da quel momento in poi, il giocatore può cominciare a giocare effettuando tentativi e verificando le risposte del computer.
- L'utente accede allo schermo iniziale del gioco (**giocata.jsp**) nel quale viene visualizzato (a) il nome del giocatore (b) la lista dei tentativi effettuati fino a quel punto dal giocatore e (c) le relative risposte fornite dal computer. La pagina contiene una maschera attraverso la quale è possibile effettuare un nuovo tentativo. Infine, la pagina contiene un pulsante per uscire dal gioco. Sottomettendo la maschera, il tentativo del giocatore viene convalidato per verificare che sia fatto di cifre tutte diverse e comprese tra 1 e 6.
- Se il tentativo è corretto ma la combinazione segreta non è stata indovinata viene visualizzato uno schermo di risposta (**risposta.jsp**) nel quale viene confermato il tentativo effettuato e viene fornita la risposta del calcolatore. La pagina contiene inoltre un collegamento per ritornare allo schermo di gioco (**giocata.jsp**).
- Se il tentativo è corretto e la combinazione segreta è stata indovinata, l'applicazione registra nella tabella **Partite** l'esito della partita, ovvero (a) la combinazione da individuare; (b) il numero di tentativi effettuati; (c) il nome del giocatore. Inoltre, registra tutti i tentativi effettuati nella tabella **Tentativi**; per ciascun tentativo memorizza (a) il numero progressivo del tentativo (1, 2, 3, ecc.) (b) la combinazione tentata; (c) i pallini neri; (d) i pallini bianchi; (e) il riferimento alla partita. Il giocatore accede allo schermo di successo (**indovinato.jsp**) che contiene un messaggio con cui viene segnalato che il gioco è finito, con un riassunto dei tentativi effettuati e delle risposte. La pagina inoltre contiene un collegamento per cominciare una nuova partita (**index.jsp**) ed uno per uscire dal gioco (**uscita.jsp**).
- Se l'utente decide di uscire dal gioco viene visualizzato lo schermo finale (**uscita.jsp**), che contiene la combinazione da indovinare, il riassunto dei tentativi effettuati con le relative risposte, un messaggio di saluto e un collegamento per raggiungere la pagina iniziale dell'applicazione (**index.jsp**) e ricominciare daccapo.

Scenario alternativo: Combinazione scorretta

- L'utente ritorna allo schermo di immissione (**giocata.jsp**) in cui viene visualizzato un messaggio di errore.

Scrivere l'applicazione Web che effettua le operazioni elencate sopra secondo le seguenti specifiche:

- L'applicazione deve utilizzare pagine JSP e un'architettura di tipo **Modello 2** basata sul framework didattico **pinco**, ed uno strato di persistenza basato su DAO e gestione delle transazioni
- E' possibile scegliere se utilizzare o meno un pool di connessioni
- E' necessario provvedere alla **convalida** dei dati sottomessi dall'utente
- Il codice HTML prodotto deve essere conforme al **DTD XHTML-Basic**. E' necessario procedere alla convalida del codice prodotto utilizzando il validatore **xmlvalid**
- La grafica deve essere organizzata utilizzando un foglio di stile **CSS**.
- NOTA: per generare un valore casuale, è possibile utilizzare il metodo statico `random()` della classe `java.lang.Math`; per esempio, per generare un numero intero casuale tra 1 e 10, è possibile utilizzare la riga di codice: `int numero = (int)(Math.random() * 10 + 1);`