

Tecnologie Informatiche per il Web - AA. 2020-21

La presentazione è individuale e richiede:

- L'invio almeno tre giorni prima della data di presentazione della documentazione del progetto
- L'effettuazione di una dimostrazione del progetto
- La discussione del codice con eventuali domande sulle motivazioni della progettazione e della codifica

La dimostrazione richiede la preparazione di una base di dati con contenuto sufficiente alla dimostrazione di TUTTI i possibili scenari di uso dell'applicazione.

Commenti generali:

- La registrazione degli utenti via web non è richiesta
- Eventuali funzioni non richieste di gestione dei dati (es, modifica o cancellazione) possono essere realizzate se comode per il testing ma sono opzionali e non valutate

Esercizio 1: aste online

Versione HTML pura

Un'applicazione web consente la gestione di aste online. Gli utenti accedono tramite login e possono vendere e acquistare all'asta. La HOME page contiene due link, uno per accedere alla pagina VENDO e uno per accedere alla pagina ACQUISTO. La pagina VENDO mostra una lista delle aste create dall'utente e non ancora chiuse, una lista delle aste da lui create e chiuse e una form per creare un nuovo articolo e una nuova asta per venderlo. L'asta comprende l'articolo da mettere in vendita (codice, nome, descrizione, immagine), prezzo iniziale, rialzo minimo di ogni offerta (espresso come un numero intero di euro) e una scadenza (data e ora, es 19-04-2021 alle 24:00). La lista delle aste è ordinata per data+ora crescente. L'elenco riporta: codice e nome dell'articolo, offerta massima, tempo mancante (numero di giorni e ore) tra il momento (data ora) del login e la data e ora di chiusura dell'asta. Cliccando su un'asta compare una pagina DETTAGLIO ASTA che riporta per un'asta aperta i dati dell'asta e la lista delle offerte (nome utente, prezzo offerto, data e ora dell'offerta) ordinata per data+ora decrescente. Un bottone CHIUDI permette all'utente di chiudere l'asta se è giunta l'ora della scadenza (si ignori il caso di aste scadute ma non chiuse dall'utente). Se l'asta è chiusa, la pagina riporta i dati dell'asta, il nome dell'aggiudicatario, il prezzo finale e l'indirizzo (fisso) di spedizione dell'utente. La pagina ACQUISTO contiene una form di ricerca per parola chiave. Quando l'acquirente invia una parola chiave la pagina ACQUISTO è aggiornata e mostra un elenco di aste aperte (la cui scadenza è posteriore alla data e ora dell'invio) il cui articolo contiene la parola chiave nel nome o nella descrizione. La lista è ordinata in modo decrescente in base al tempo (numero di giorni e ore) mancante alla chiusura. Cliccando su un'asta aperta compare la pagina OFFERTA che mostra i dati dell'articolo, l'elenco delle offerte pervenute in ordine di data+ora decrescente e un campo di input per inserire la propria offerta, che deve essere superiore all'offerta massima corrente di un importo pari almeno al rialzo minimo. Dopo l'invio dell'offerta la pagina OFFERTA mostra l'elenco delle offerte aggiornate. La pagina ACQUISTO contiene anche un elenco delle offerte aggiudicate all'utente con i dati dell'articolo e il prezzo finale.

Versione con JavaScript

Si realizzi un'applicazione client server web che estende e/o modifica le specifiche precedenti come segue:

- Dopo il login, l'intera applicazione è realizzata con un'unica pagina.

Esercizio 3: catalogazione di immagini

Versione HTML pura

Un'applicazione permette all'utente (ad esempio il curatore di un catalogo online di immagini) di gestire una tassonomia di classificazione utile per etichettare immagini allo scopo di consentire la ricerca in base alla categoria.

Dopo il login, l'utente accede a una pagina HOME in cui compare un albero gerarchico di categorie. Le categorie non dipendono dall'utente e sono in comune tra tutti gli utenti. Un esempio di un ramo dell'albero è il seguente:

- 9 Mitologia classica e storia antica >>sposta
- 91 Divinità della mitologia classica >>sposta
- 911 Divinità del cielo >>sposta
- 9111 Giove >>sposta
- 91111 Attributi di Giove >>sposta
- 9112 Giunone >>sposta
- 912 Divinità degli inferi >>sposta
- 9121 Plutone >>sposta
- 9122 Ecate >>sposta

Le categorie hanno nomi distinti. L'utente può inserire una nuova categoria nell'albero. Per fare ciò usa una form nella pagina HOME in cui specifica il nome della nuova categoria e sceglie la categoria padre. L'invio della nuova categoria comporta l'aggiornamento dell'albero: la nuova categoria è appesa alla categoria padre come ultimo sottoelemento. Alla nuova categoria viene assegnato un codice numerico che ne riflette la posizione (ad esempio, la nuova categoria Mercurio, figlia della categoria "911 Divinità del cielo" assume il codice 9113). Per semplicità si ipotizzi che per ogni categoria il numero massimo di sottocategorie sia 9, numerate da 1 a 9. Dopo la creazione di una categoria, la pagina HOME mostra l'albero aggiornato. L'utente può spostare di posizione una categoria: per fare ciò clicca sul link "sposta" associato alla categoria da spostare. A seguito di tale azione l'applicazione mostra, sempre nella HOME page, l'albero con evidenziato il sotto albero attestato sulla categoria da spostare: tutte le altre categorie hanno un link "sposta qui". Ad esempio, a seguito del click sul link "sposta" associato alla categoria "9111 Giove" l'applicazione visualizza l'albero come segue:

- 9 Mitologia classica e storia antica >>sposta qui
- 91 Divinità della mitologia classica >>sposta qui
- 911 Divinità del cielo >>sposta qui
- 9111 Giove**
- 91111 Attributi di Giove**
- 9112 Giunone >>sposta qui
- 912 Divinità degli inferi >>sposta qui
- 9121 Plutone >>sposta qui
- 9122 Ecate >>sposta qui

La selezione di un link "sposta qui" comporta l'inserimento della categoria da spostare come ultimo figlio della categoria destinazione. Ad esempio, la selezione del link "sposta qui" della categoria "912 Divinità degli inferi" comporta la seguente modifica dell'albero:

- 9 Mitologia classica e storia antica >>sposta
- 91 Divinità della mitologia classica >>sposta
- 911 Divinità del cielo >>sposta
- 9111 Giunone >>sposta
- 912 Divinità degli inferi >>sposta
- 9121 Plutone >>sposta
- 9122 Ecate >>sposta
- 9123 Giove >>sposta
- 91231 Attributi di Giove >>sposta

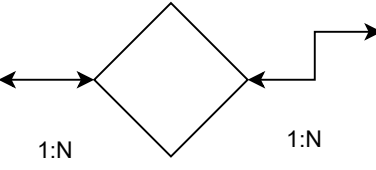
Le modifiche effettuate da un utente e salvate nella base di dati diventano visibili agli altri utenti. Lo spostamento potrebbe creare un vuoto nella numerazione delle categorie figlie dello stesso padre. La funzione di ricalcolo della numerazione delle categorie figlie dello stesso padre di quella spostata non è richiesta ma è lasciata come opzionale.

Versione con JavaScript

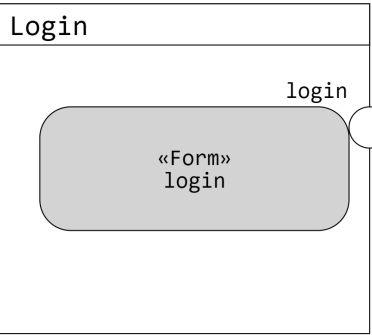
Si realizzi un'applicazione client server web che estende e/o modifica le specifiche precedenti come segue:

- Dopo il login dell'utente, l'intera applicazione è realizzata con un'unica pagina.
- Ogni interazione dell'utente è gestita senza ricaricare completamente la pagina, ma produce l'invocazione asincrona del server e l'eventuale modifica del contenuto da aggiornare a seguito dell'evento.
- La funzione di spostamento di una categoria è realizzata mediante drag & drop.
- A seguito del drop della categoria da spostare compare una finestra di dialogo con cui l'utente può confermare o cancellare lo spostamento. La conferma produce l'aggiornamento a lato client dell'albero.
- L'utente realizza spostamenti anche multipli a lato client. A seguito del primo spostamento compare un bottone SALVA la cui pressione provoca l'invio al server dell'elenco degli spostamenti realizzati (NON dell'intero albero). L'invio degli spostamenti produce l'aggiornamento dell'albero nella base dei dati e la comparsa di un messaggio di conferma dell'avvenuto salvataggio.

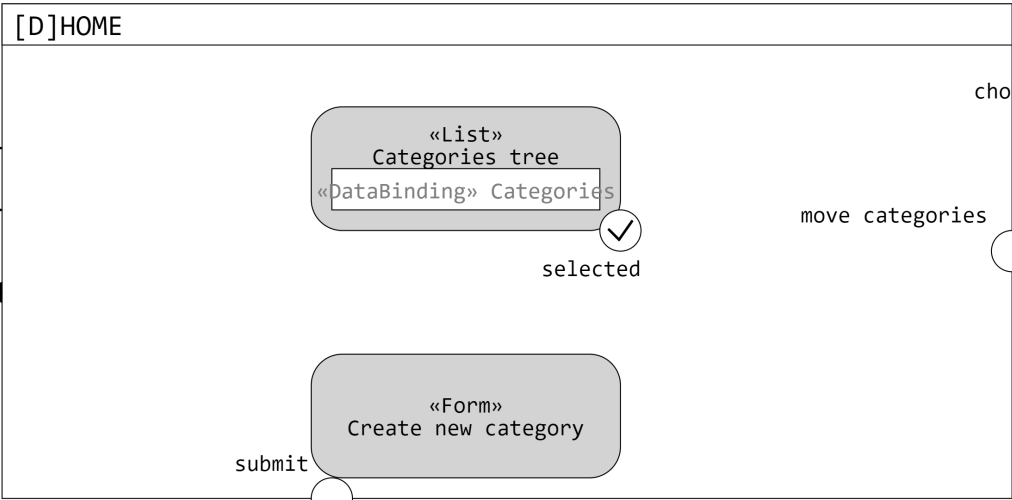
Users	
PK	<u>id int NOT NULL AI</u>
	customer_username char(50) NOT NULL
	customer_psw char(50) NOT NULL



Categories	
PK	<u>id int NOT NULL AI</u>
	<u>number int NOT NULL</u>
	<u>name char(50) NOT NULL</u>
	parent_number int NOT NULL



login



«List» Categories tree
«DataBinding» Categories
selected

«Form» Create new category
submit

display updated tree

add new category

choose new father category
move categories
display highlighted categories

Components

Model objects (beans):

- Users:
- Categories

Data Access Objects (Classes):

CategoriesDAO:

- findAllCategories
- findCategoryById
- findCategoryByNumber

UsersDAO:

- findUserByUsername

Controllers:

- GetCategories [GET]
- AddCategory [POST]
- UpdateCategory [POST]

View:

- LoginPage
- Home