

Tecnologie Web T
21 Giugno 2018 – Compito B

Tempo a disposizione: 3 ore

La soluzione comprende la **consegna elettronica** dei seguenti file mediante l'apposito applicativo Web **esamix** (<http://esamix.labx>):

AcquistiGruppo.zip	file zip contenente il sorgente java/class e pagine Web per punto 1
Supermercato.zip	file zip contenente il sorgente java/class e pagine Web per punto 2
Offerte.zip	file zip contenente il sorgente java/class e pagine Web per punto 3

Ogni file .zip consegnato DEVE CONTENERE TUTTI e SOLI i file creati/modificati e/o ritenuti importanti in generale ai fini della valutazione (ad esempio, descrittori, risorse statiche o dinamiche, codice Java e relativi .class, ecc.) e NON dell'intero progetto

N.B. Per superare la prova scritta di laboratorio ed essere ammessi all'orale, è necessario totalizzare almeno 18 punti (su un totale disponibile di 33), equamente distribuiti sui tre esercizi, ovvero almeno 6 punti sul primo esercizio, 6 punti sul secondo esercizio e 6 punti sul terzo esercizio

Studenti in debito di Tecnologie Web L-A

Viene richiesto lo svolgimento dei soli esercizi 1 (17 punti) e 2 (16 punti). Tempo a disposizione: 2 ore.

I 18 punti necessari per l'ammissione all'orale sono così distribuiti: almeno 10 punti sul primo esercizio e almeno 8 punti sul secondo

ESERCIZIO 1 (11 punti)

Si realizzi un'applicazione Web, principalmente basata su tecnologie JSP, Java servlet e Javascript, per la realizzazione di un sito di e-commerce per la vendita di prodotti a un gruppo di utenti coordinati.

L'applicazione deve consentire ad ogni utente di aprire una propria sessione e di essere riconosciuto come appartenente a un gruppo; come ipotesi semplificativa, non è possibile per uno stesso utente avere più sessioni correntemente attive, ad esempio da dispositivi differenti. Una volta aperta la sua sessione, l'utente deve avere la possibilità di visualizzare il catalogo di prodotti in vendita (realizzato tramite pagina JSP) e per semplicità di aggiungere un solo prodotto al suo carrello (realizzato tramite servlet). Per l'aggiunta di prodotti in carrello, si deve verificare localmente al cliente che il numero di unità di prodotto inserito sia non superiore alla quantità disponibile in catalogo; solo in caso positivo il prodotto sarà aggiunto al carrello.

Differentemente da quanto usuale, un ordine sarà finalizzabile solo dopo che la quantità di prodotto venduta a un gruppo sarà tale da rendere il residuo di magazzino multiplo di una quantità ottimale (prefissata per ogni prodotto e mantenuta come dato lato server); la verifica della condizione suddetta abiliterà, dalla richiesta successiva, un pulsante di finalizzazione ordine normalmente disabilitato. Ci possono essere modifiche concorrenti che possono portare a inconsistenze? Nel caso, descrivere quali sono esattamente come commento nel file sorgente.

Infine, deve essere data la possibilità all'amministratore del sistema (username=admin; password=admin) di visualizzare da quanto tempo tutte le sessioni utente sono attive e di cancellare un carrello esistente selezionato.

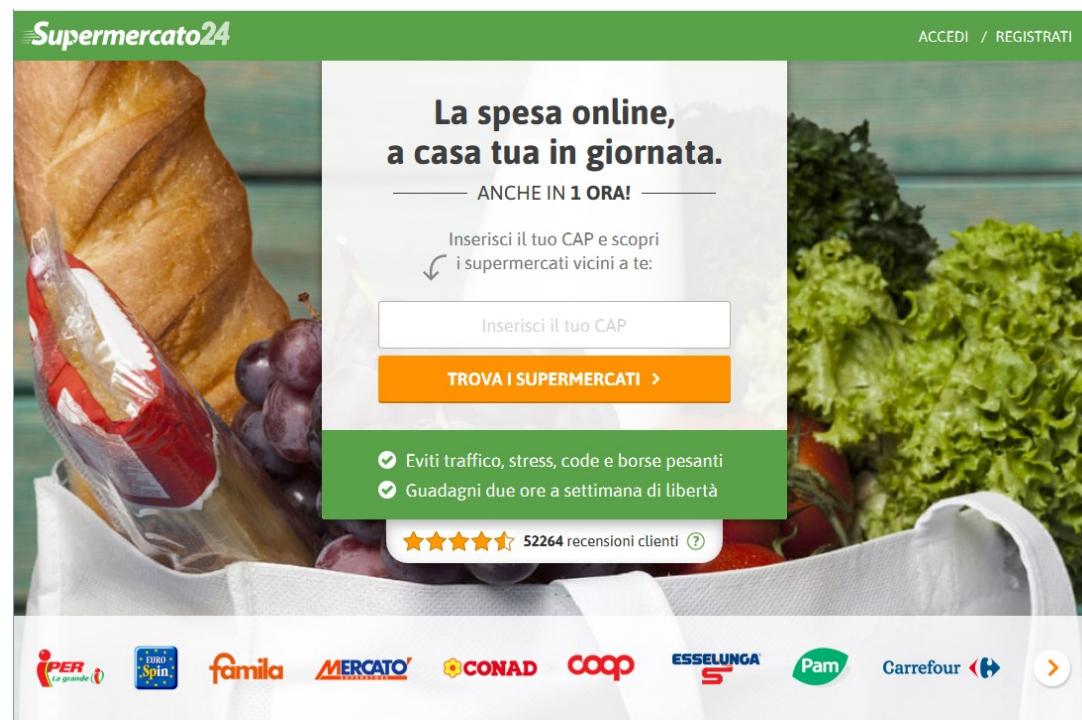
ESERCIZIO 2 (11 punti)

Si realizzino le **pagine Web dinamiche** (basate su **tecnologia HTML, CSS, e Javascript**) in grado di **riprodurre il contenuto e il layout grafico** dello “snapshot” del sito Web **Supermercato** di seguito riportato.

Nel dettaglio lo “snapshot” si compone di tre parti principali:

- 1) barra verde in alto contenente il link “Supermercato” alla “home” e i link “Accedi” / “Registrati” per accedere / registrarsi al servizio di spesa on-line;
- 2) parte centrale, con sfondo fotografia a tema, contenente al centro un form con, oltre alla parte testuale di presentazione da riprodurre come da esempio, il campo di inserimento “Inserisci il tuo CAP” per l’inserimento del proprio CAP e il bottone di ricerca arancione “TROVA I SUPERMERCATI” per trovare i supermercati più vicini ad ognuno. La stringa ammessa per rappresentare il CAP è una stringa formata da 5 numeri, con primo numero diverso da ‘0’. Una volta immesso il valore di un CAP valido, viene mostrata automaticamente nella riga seguente quella contenente il CAP la stringa relativa al quartiere ad esso associato (ad esempio, digitando il CAP “40134”, viene mostrata la stringa “Bologna, Quartiere Saragozza”), stringa che verrà usata come campo di ricerca dal bottone arancione “TROVA I SUPERMERCATI”. Infine, deve essere mostrato il rating delle recensioni dei clienti in termini di migliore punteggio e peggiore punteggio ricevuti, sotto forma di numero di stelle colorate, da 1 a 5 (diversamente da quanto riportato in figura dove invece viene visualizzato il valore medio e il numero totale di recensioni);
- 3) barra bianca in basso contenente lo *slider* delle icone rappresentanti i supermercati on-line che aderiscono al servizio; in particolare, sono mostrate 5 icone alla volta; il passaggio da un insieme di icone al successivo avviene attraverso il bottone “freccia”.

N.B. La soluzione NON deve far uso del costrutto HTML `frame`. Per rappresentare le immagini/icone riportate nello snapshot, si utilizzino figure di esempio a piacere.



Tecnologie Web T
21 Giugno 2018 – Compito B

ESERCIZIO 3 (11 punti)

Si realizzi un'applicazione Web, principalmente basata su tecnologie Javascript, AJAX, JSON e Java servlet per la realizzazione di un efficiente servizio di pubblicità di offerte in corso in alcuni negozi aderenti.

L'applicazione deve mantenere in memoria lato servitore una struttura dati con le offerte più recenti (ultime 24 ore). I dati di ogni offerta sono strutturati in timestamp, nome prodotto, prezzo, negozio, booleano che indica se l'offerta è disponibile per i soli soci del servizio o per tutti.

Quando un utente accede al servizio, viene identificato e associato alla sua utenza (socio del servizio oppure no); come ipotesi semplificativa, non è possibile per uno stesso utente avere più sessioni correntemente attive, ad esempio da terminali differenti. Al primo accesso al servizio, l'utente deve ricevere in formato JSON e avere visualizzate tutte le offerte relative alle ultime 24 ore.

In seguito e fino alla chiusura della sua sessione di interazione, l'applicazione deve scaricare (pre-fetching) lato cliente, ogni 10 minuti, i soli contenuti “nuovi” rispetto a quelli già ottenuti in interazioni precedenti. I contenuti nuovi scaricati non vengono però immediatamente visualizzati ma solo tenuti in cache locale; saranno visualizzati solo come risultato della pressione esplicita del tasto “Visualizza da cache” da parte dell'utente.

La sessione di interazione deve terminare dopo 30 minuti per i clienti “non soci”, dopo un'ora per i clienti soci; inoltre, in caso di numero di clienti superiore a una soglia di alto traffico (100 clienti con sessioni attive), sarà forzata la terminazione di sessione per tutti i clienti “non soci” in modo tale da migliorare la qualità di servizio dei clienti rimanenti.