

Esami di stato per l'abilitazione all'esercizio della professione di Ingegnere I Sessione 2014

Sezione A Prova progettuale – Settore Informatica

Al candidato è richiesto di svolgere uno dei seguenti temi proposti.

TEMA N. 1

Si progetti un sistema informativo per la gestione del servizio mensa di un grande campus universitario. Il campus dispone di diverse mense dove gli studenti possono recarsi a colazione, a pranzo, e a cena. Non tutte le mense sono aperte a cena e ognuna ha un proprio giorno di chiusura. Ogni mensa ha orari diversi, un certo numero di posti a sedere, un numero massimo di coperti per pasto. Ogni mensa ha un giorno della settimana in cui i pasti costano 25% in meno. Alcune mense servono pasti specifici per vegetariani, della cucina internazionale e della cucina mediterranea.

Ogni studente è dotato di una tessera mensa, e di una carta a punti prepagata per comprare i pasti. Ogni mensa richiede una “quantità minima di punti” (per l’acquisto del pasto più economico) diversa. Uno studente può mangiare ad una certa mensa se la sua carta ha punti a sufficienza. L’ingresso è vietato anche quando una mensa è completa: il numero di persone nella mensa è pari al numero di posti a sedere. L’addetto deve avvisare lo studente ogni volta che i punti residui sulla tessera non sono più sufficienti a consumare un pasto, neppure il più economico. Le tessere possono essere ricaricate. Il minimo sono 50 punti, il massimo 1000 punti.

Il sistema deve essere in grado di monitorare l’occupazione delle mense in “tempo reale”. A tale scopo dei monitor informativi dell’università devono poter guidare gli studenti indicando le mense meno affollate e visualizzando altre informazioni utili sulle mense, quali, ad esempio, le mense più convenienti perché applicano per quel giorno lo sconto sui pasti, le mense ancora aperte, la “quantità minima di punti”, ecc. Nell’accezione più ampia, le informazioni sulle mense devono poter essere visualizzate anche da dispositivi mobili, quindi viene definita un’interfaccia tipo web service per accedervi. Alla fine della giornata il sistema deve anche produrre un consuntivo sull’uso delle diverse mense al fine di consentire il monitoraggio e il miglioramento della qualità del servizio.

Si svolgano i seguenti punti.

- Progettare l’architettura del sistema tramite la definizione di un diagramma delle componenti UML o un diagramma a blocchi facendo tutte le ulteriori ipotesi necessarie per la modellizzazione delle varie entità/dispositivi coinvolti. Fornire se si ritiene utile anche altri

diagrammi in UML o notazioni simili (macchine di stato, reti di Petri, ecc.) per modellare il comportamento di una o più componenti critiche del sistema e le loro interazioni.

- Si effettui la progettazione concettuale dei dati che il sistema deve gestire (quelli relativi alle mense, pasti, punti, sconti, orari apertura/chiusura, ecc.) mediante un modello Entità-Relazione o mediante un diagramma delle classi UML.
- Si definisca lo schema logico relazionale del database di supporto del sistema, usando il linguaggio SQL per definire le tabelle e i vincoli. Scrivere, in particolare, le seguenti interrogazioni in algebra relazionale e/o SQL:
 - a. Trovare le mense aperte ad una certa ora in una certa data.
 - b. Contare il numero di posti disponibili in una mensa ad una certa ora in una certa data.
 - c. Trovare le mense che servono il pasto più economico del giorno (a pranzo o a cena).
- Si definiscano i messaggi essenziali che devono essere scambiati tra il vari client e server (si definiscano sia i messaggi di richiesta che quelli di risposta) per la visualizzazione di un certo numero di informazioni sulle mense ritenute dal candidato essenziali. Usare a tale scopo, un IDL (Interface Description Language) di propria scelta, ad esempio: RSDL (RESTful Service Description Language) per REST web services, WSDL, XML DTD o schema, Java interface, UML interface e sequence diagram, ecc.
- Scrivere il codice Java-like o pseudocodice del lato server che gestisce i gruppi di messaggi (interagendo con il database per prelevare/aggiornare i dati) per la visualizzazione delle informazioni sui vari dispositivi (monitor e dispositivi mobili).

TEMA N. 2

Una compagnia aerea vuole estendere il suo portale web per inserire la gestione di un programma fedeltà. Chi si iscrive al programma, ogni volta che vola con la compagnia, accumula punti (miglia) che danno diritto a dei premi. Ad esempio, bisogna volare per almeno 25.000 miglia per avere diritto ad un volo gratuito in Europa; occorrono 65.000 miglia per un volo negli Stati Uniti; bastano 5.000 miglia per un buono acquisto in un negozio convenzionato.

Il nuovo portale deve gestire i clienti della compagnia che partecipano al programma. I partecipanti sono organizzati in tre fasce di merito in funzione delle miglia volate durante un anno solare. Inizialmente tutti appartengono al primo livello; se si volano 35.000 miglia si passa al secondo livello; si accede al terzo livello con 100.000 miglia volate in un anno. I tre livelli danno diritto a facilitazioni e premi differenziati. Oltre ai clienti, il portale deve gestire i tipi di premi (volo gratuito, soggiorno gratuito, buono sconto), il numero di miglia necessarie per ogni premio particolare (un volo gratuito a New York richiede più miglia di un volo per Roma) e lo storico dei clienti: quanti voli ha effettuato ogni cliente, quante miglia ha guadagnato, quali premi ha già riscosso e quante miglia gli restano da “spendere”. Il portale non deve gestire l'intero piano di volo della compagnia aerea per sapere se la richiesta di un biglietto premio per una certa data può essere evasa. Semplicemente, se il cliente ha un numero sufficiente di miglia, viene erogato un voucher che poi il cliente utilizzerà nel momento della presentazione.

Il portale deve essere in grado di aggiornare la posizione di ogni cliente in funzione di ogni volo effettuato e di ogni premio richiesto. Occorre tenere in considerazione anche che le miglia accumulate scadono dopo 5 anni dal momento in cui sono state acquisite (cioè dalla data del volo).

Nell'accezione più ampia, il portale web deve poter essere consultato anche da applicazioni per dispositivi mobili, quindi viene definita un'interfaccia tipo web service per accedervi.

Si consideri di memorizzare i premi, i voli e l'anagrafica clienti in un nuovo database o nell'attuale database di supporto del portale. Si consideri inoltre la comunicazione client-server che riguarda la registrazione di un nuovo utente, l'autenticazione di un utente con il servizio, la consultazione della propria posizione (numero di miglia) rispetto al programma fedeltà, la richiesta di un premio, ecc. In particolare, si svolgano i seguenti punti:

- Si effettui la progettazione concettuale dei nuovi dati da gestire (quelli relativi ai voli, premi, e clienti) mediante un modello Entità-Relazione o mediante un diagramma delle classi UML.
- Si definisca lo schema logico relazionale del database del server, usando il linguaggio SQL per definire le tabelle e i vincoli. Scrivere quindi le seguenti interrogazioni in algebra relazionale e/o SQL:
 - a. Trovare il numero di miglia accumulate da un cliente.
 - b. Trovare i possibili premi che un utente, in base alle miglia accumulate, può riscuotere.

- c. Trovare i clienti che hanno diritto ad un certo tipo di premio.
- d. Contare il numero di clienti che hanno riscosso un certo tipo di premio.
- e. Contare i clienti che hanno diritto (in base alle miglia accumulate) a dei premi ma che non hanno riscosso alcun premio.
- Si definiscano i messaggi che devono essere scambiati tra il client e il server (si definiscano sia i messaggi di richiesta che quelli di risposta) usando un IDL (Interface Description Language) di propria scelta, ad esempio: RSDL (RESTful Service Description Language) per REST web services, WSDL, XML DTD o schema, Java interface, UML interface e sequence diagram, ecc.
- Scrivere il codice Java-like o pseudocodice del lato server che gestisce i gruppi di messaggi (interagendo con il database per prelevare/aggiornare i dati) per erogare alcuni servizi essenziali.
- Progettare l'interfaccia di una ipotetica applicazione per dispositivo mobile che si interfaccia con il server (flusso di navigazione da una schermata all'altra). Dopo aver chiaramente definito le funzionalità ritenute essenziali, il candidato disegni dei mockup funzionali di base delle schermate previste e descriva la modalità di navigazione.