

Progetto Modelli dei dati e DBMS di nuova generazione: Gestione di Eventi in Aree Espositive

A.A. 2019/2020

1. Descrizione del dominio applicativo

La Fira s.r.l. è una società che gestisce aree espositive in diverse città europee. Di ogni area espositiva si conosce il nome, la locazione (indirizzo, città e paese) e il numero di visitatori che possono ospitare. Le aree espositive sono formate da slot di esposizione. Gli slot ha un numero identificativo, una dimensione in metri quadri, il prezzo giornaliero di affitto e delle coordinate nell'area espositiva. Gli slot possono essere di due tipi: standard e deluxe. Per gli slot standard è presente anche la lista facilities che sono in esso disponibili (acqua, corrente elettrica, arredi). Per gli slot deluxe è invece presente la percentuale di maggiorazione del prezzo di affitto dello slot.

I clienti della Fira s.r.l. effettuano prenotazioni degli spazi delle varie aree espositive per un dato evento (di cui si conosce il nome, un codice identificativo, il periodo temporale in cui è tenuto e l'area espositiva). Di ogni cliente si tiene traccia della ragione sociale e del numero di telefono e delle aree espositive che ha richiesto in passato e va identificata l'area espositiva preferita. Per ogni prenotazione occorre memorizzare il numero di giorni di affitto e la lista degli slot che sono prenotati (possono essere più di uno ma riguardano tutti la stessa area espositiva). Il prezzo da pagare viene determinato sulla base del periodo, del tipo e numero di slots prenotati. La Fira s.r.l. ha sempre bisogno di sapere per ogni evento quali sono gli slot disponibili e quali sono quelli prenotati in un determinato giorno.

La Fira s.r.l. decide di dotare ogni slot delle sue aree espositive di RFID per tenere traccia dei passaggi dei visitatori nell'ambito di un evento. Ogni visitatore all'ingresso dell'area espositiva fornisce il suo nome, cognome e email e riceve un braccialetto che tiene per tutto l'arco della visita. Nel braccialetto viene inserito un software che permette di tenere traccia del giorno della visita, degli slot visitati e dei percorsi effettuati dal visitatore. Di ogni slot deve essere memorizzato il nome e il tipo (standard e deluxe), il tipo di prodotti esposti, il nome dell'azienda che espone, ed il numero di passaggi che il visitatore ha fatto in quello slot. Inoltre, viene memorizzato il percorso vero e proprio, cioè la sequenza di slot visitati con l'ora di entrata, l'ora di uscita e il tempo trascorso. Inoltre, per ogni visita il braccialetto mette a disposizione sul display con tre faccine attraverso cui il visitatore può esprimere il giudizio sullo slot (mi piace, non mi piace, indifferente). Il giudizio viene attribuito allo slot che sta correntemente visitando. A titolo esemplificativo in Figura 1 viene mostrato un esempio di documento XML.

2. Richieste

Si richiede di svolgere i seguenti punti

1. Effettuare una progettazione concettuale del dominio sopra proposto, andando a creare uno schema UML. Lo schema deve rappresentare nel modo più opportuno i vincoli imposti dal testo. Inoltre, vanno evidenziati tutti i vincoli non esprimibili con lo schema UML in linguaggio naturale. Riportare tutte le assunzioni/semplificazioni che si intendono adottare.
2. sviluppare uno schema XML per gestire i documenti XML da inserire nel braccialetto elettronico. Si faccia in modo che sia presente un solo elemento globale e di utilizzare al meglio il sistema di tipi messo a disposizione da XML schema. Inoltre, non ci possono essere due slots con lo stesso id e nel percorso possono essere inclusi solo slot inclusi nel documento.

```

<?xml version="1.0" encoding="UTF-8"?>
<visit xsi:noNamespaceSchemaLocation="visit.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
id="324" event="EFIORI20" day="4/1/2020" entrance="10:24">
<slots>
  <slot id="s347" rating="like" type="standard" num_visit="3" prod="flower">
  <slot id="s353" rating="unlike" type="deluxe" num_visit="1" prod="bricks">
  <slot id="s447" type="standard" num_visit="5" prod="mirrors">
</slots>
<path>
  <slot ref="s447" length="19">
    <entrance>11:03</entrance>
    <exit>11:24</exit>
  </slot>
  <slot ref="s347" length="2">
    <entrance>11:33</entrance>
    <exit>11:35</exit>
  </slot>
  ...
</path>
</visit>

```

Figure 1 Esempio di documento XML che può essere memorizzato nel braccialetto relativo ad una visita di Mario Rossi per l'evento "Fiori di Primavera" (è solo un esempio e potete modificarlo come ritenete opportuno per renderlo conforme all'intero database)

3. Proporre una traduzione nel modello object-relational di Oracle cercando di sfruttare al meglio le caratteristiche di Oracle nel trattamento degli oggetti e dei documenti XML. Per effettuare la traduzione tenere in considerazione il seguente carico di lavoro:
 - a. Creare un documento XML che riporta gli slot disponibili per un dato evento.
 - b. Determinare l'evento dell'anno. L'evento dell'anno è l'evento che ha portato il maggior incasso possibile
 - c. Determinare lo slot che è maggiormente preferito per un determinato evento con il numero totale di visite ricevute e il numero di like che ha ottenuto. .
 - d. Inserire la prenotazione di un cliente X per un evento Y per i tre slot deluxe liberi che hanno la metratura più alta per un periodo di Z giorni. Dopo questa prenotazione l'area espositiva preferita diventa quella che contiene tali slots. L'operazione deve garantire che la base di dati sia consistente dopo l'operazione.
4. Scrivere gli statement SQL per la creazione della base di dati e per il suo popolamento con sufficienti dati da poter verificare il sistema sviluppato.
5. Implementare in SQL le interrogazioni del punto 3. Utilizzare al meglio il linguaggio SQL/XML e XQuery. Si noti che potrebbe non essere sufficiente una singola interrogazione.

In alternativa, invece del modello object-relational di Oracle è possibile utilizzare MongoDB per la rappresentazione di tutti i dati del dominio applicativo. Funzioni devono essere previste per trasformare i dati da XML in JSON e viceversa. Le interrogazioni sopra descritte devono essere realizzate utilizzando il linguaggio messo a disposizione da MongoDB.

3. Consegna

Il progetto può essere consegnato in un momento qualunque dell'anno accademico e ha validità fino ad Aprile 2021. Dopo tale data occorrerà svolgere un nuovo progetto.

Il progetto deve essere svolto da gruppi di al più **due** persone.

Per gli studenti che consegneranno il progetto entro Febbraio 2020, verrà dato un bonus di 1 punto.
Occorre consegnare:

1. Una documentazione tecnica contenente:
 - a. Il diagramma UML corredato di spiegazioni, motivazioni, vincoli e qualunque altra informazione che permetta di comprendere la progettazione effettuata. Il diagramma deve essere etichettato con il verso di attraversamento delle associazioni che tenga in considerazione il carico di lavoro assegnato.
 - b. La traduzione del modello ad oggetti nel modello object-relational di Oracle (oppure nel modello di MongoDB)
 - c. Le interrogazioni SQL (o le equivalenti in MongoDB) richieste nel carico di lavoro e l'implementazione dei principali vincoli.
2. Uno script per la creazione della base di dati, il suo popolamento, l'implementazione dei principali vincoli e le interrogazioni SQL (o le equivalenti in MongoDB) richieste nel carico di lavoro.