

OBIETTIVO

Lo scopo del progetto è creare un'applicazione web con la quale gli utenti potranno acquistare biglietti aerei. Il sistema si basa su un database relazionale con le sue funzioni di gestione voli e prenotazione. Avremo un database che supporterà la prenotazione su molteplici tratte in tutto il mondo, centinaia di voli per varie compagnie aeree.

FASE 1 - DATABASE

Vincoli:

- Database Oracle

Il database deve essere progettato e sviluppato per supportare:

- **Dettagli volo:** include il terminal del volo di origine e il terminal di destinazione, insieme agli scali intermedi, il numero di posti prenotati/posti disponibili tra due destinazioni, costo base.
- **Dettagli compagnia aerea:** Nome compagnia, sede.
- **Dettaglio aereoporto:** Città, Nome
- **Dettagli aereo:** Codice aereo, codice compagnia, modello, stato (funzionante/non funzionante), note
- **Dettaglio cliente:** include codice cliente, nome, indirizzo e numero di telefono, età. Queste informazioni possono essere utilizzate per conservare i registri del cliente per qualsiasi emergenza o per qualsiasi altro tipo di informazione.
- **Dettaglio prenotazione:** include il codice cliente, il numero del volo, la data della prenotazione, la data del viaggio, costo fatturato. Il prodotto presenta le principali caratteristiche del sistema di database delle compagnie aeree, come mostrato nel diagramma del flusso di dati sottostante del sistema di database delle compagnie aeree
- **Tabella statistiche** (id volo, numero prenotazioni, età media passeggeri, numero adulti[>18], numero bambini[<18], totale euro fatturato)
- **Classe e caratteristiche dell'utente:** gli utenti del sistema dovrebbero essere in grado di recuperare dal database le informazioni sui voli tra due determinate città con date/orari di viaggio specificati.
Esisteranno utenti AMMINISTRATORI in grado di effettuare prenotazioni, inserimento compagnie aeree, ecc., e utenti READER in grado di visualizzare soltanto le informazioni

Implementare i seguenti TRIGGER:

- Al cambio di stato di un AEREO da funzionante a non funzionante, modifichi automaticamente tutti i VOLI collegati all'aereo in questione, e imposti ai VOLI in primo aereo utile al posto di quello non funzionante
- All'inserimento di un nuovo VOLO, verificare che per l'aereo assegnatoli sia il primo volo, in questo caso sarà necessario aggiornare il campo note di AEREO, con la dicitura "*Volo inaugurale in data [data del volo inserito]*"
- All'inserimento di una prenotazione aggiornare con un trigger tutte le informazioni della tabella STATISTICHE

NOTA: Sono previsti voli con SCALI.

Creare:

- Diagramma Entità-Relazione
- Modello relazionale (tabelle normalizzate derivanti dal diagramma ER)
- Implementare in database progettato nei passi precedenti

FASE 2 – COSTRUZIONE DEL BACKEND IN JAVA

Ai fini dello sviluppo dell'applicazione, risulterà necessario la creazione di una batteria di WebServices (REST) che permettano di eseguire le seguenti operazioni:

- Visualizzare i dettagli di una singola prenotazione
- Inserire/Modificare una prenotazione

- Visualizzare i dettagli di un singolo cliente
- Inserire/Modificare un cliente (senza necessariamente una prenotazione associata)
- Visualizzare i dettagli di una singola compagnia aerea
- Inserire/Modificare una compagnia aerea
- Visualizzare i dettagli di un aeroporto
- Inserire/Modificare un aeroporto

Implementare la seguente logica per stabilire il costo di un biglietto

Calcolo il costo del biglietto del volo sulla base del costo base in modo tale che aumenti del +10% ogni volta che si supera uno scaglione di posti prenotati rispetto ai posti totali di cui dispone l'aereo.

Il numero di scaglioni è 5.

Es:

prezzo base = 100

Numero posti totali = 50

I primi 10 pagheranno 100

I successivi 10 pagheranno 110

I successivi 10 pagheranno 121

Una volta ottenuto il costo a seconda dello scaglione, se la prenotazione cade in un giorno di festa domenica/natale/ecc, si applica un ulteriore +10% al costo.

Altro

Utilizzare il framework SWAGGER per poter creare documentazione e client web per i WebServices da creare.

Esempio da poter seguire: <https://www.theserverside.com/video/Step-by-step-RESTful-web-service-example-in-Java-using-Eclipse>

FASE 2 – COSTRUZIONE DEL FRONTEND IN ANGULAR

Realizzare un'applicativo di FrontEnd in Angular che faccia uso dei WebServices realizzati al punto precedente.

L'applicativo disporrà delle seguenti funzionalità:

- Dettaglio prenotazione
- Inserimento di una prenotazione

- Modifica di una prenotazione
- Inserimento aereo, volo, cliente, utente

Per accedere all'applicativo bisognerà loggarsi.

Solo gli utenti autorizzati dovranno poter accedere all'applicativo.

Possibile esempio da cui partire: <https://medium.com/nerd-for-tech/angular-api-tutorial-d87f538ac33c>