Traccia-Askos

Il progetto prevede la realizzazione di un sistema informativo completo per la gestione moderna ed efficiente di un tour operator privato, come ad esempio Askos Tour. Lo scopo è quello di sostituire l'attuale gestione parzialmente manuale con una piattaforma digitale centralizzata, basata su un database relazionale SQL, in grado di organizzare, controllare e monitorare tutte le attività operative, i tour, le guide, i clienti e le prenotazioni.

Il cuore del sistema sarà un catalogo strutturato di tour, ciascuno identificato da un codice univoco, con informazioni dettagliate come titolo, durata, data, Prezzo, numero partecipanti attualmente prenotati e lingua in cui viene offerto. Ogni tour sarà assegnato a una guida turistica qualificata, selezionata anche in base alla conoscenza linguistica necessaria per il pubblico previsto.

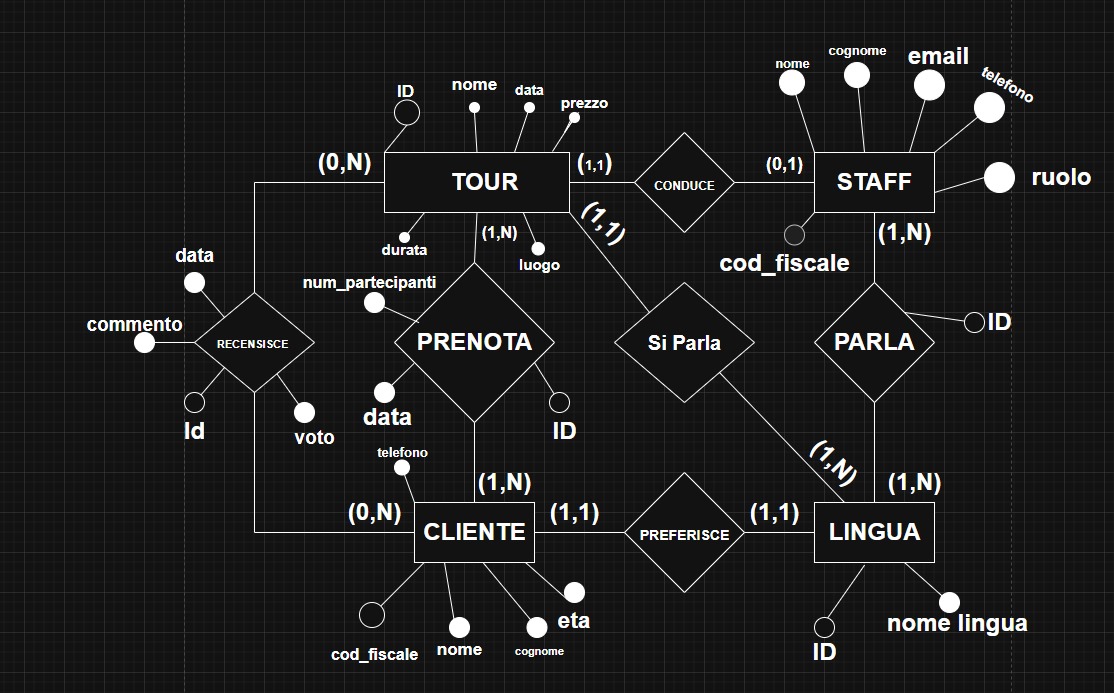
Le guide turistiche registrate nel sistema disporranno di un profilo completo, che includerà dati anagrafici, codice fiscale, contatti e l'elenco delle lingue conosciute.

I clienti, suddivisi per lingua , avranno a disposizione profili personali con dati anagrafici, contatti e lingua preferita, utile per suggerire i tour più adatti. Ogni cliente potrà effettuare prenotazioni onlne e monitorando lo storico delle proprie attività.

La gestione delle prenotazioni costituirà una delle funzionalità centrali del sistema: ogni prenotazione sarà visible da. Il database manterrà uno storico completo delle prenotazioni per ogni cliente, utile sia a fini gestionali che per eventuali analisi statistiche.

Infine, il sistema disporrà di un'interfaccia di amministrazione centralizzata, utile per creare tour, gestire guide, clienti, lingue e attività operative, garantendo così un controllo completo e una visione integrata di tutte le risorse e i processi aziendali.

1. **Analisi e Progettazione Concettuale**

****

L’obiettivo del progetto è la realizzazione di un sistema informativo per la gestione dei tour offerti da Askos Tours. Il sistema copre tutte le funzionalità principali relative alla prenotazione, gestione e svolgimento dei tour, oltre alla gestione del personale (guida e driver) e alla raccolta di recensioni da parte dei clienti.  
  
Il dominio è stato analizzato identificando le seguenti entità principali:  
  
- Cliente, con attributi come codice fiscale, nome, cognome, telefono, email e lingua preferita.  
- Lingua, per gestire le lingue in cui vengono condotti i tour.  
- Staff, che include sia guide che driver, con i relativi dati anagrafici e ruolo.  
- Tour, che rappresenta ogni attività turistica con data, durata, prezzo, lingua e personale assegnato.  
- Prenotazione, che collega un cliente a un tour specifico, con il numero di partecipanti.  
- Recensione, che viene registrata da un cliente solo dopo aver partecipato al tour.  
  
Relazione di specializzazione  
  
Una delle scelte progettuali chiave è l’introduzione della specializzazione dell'entità generale User (fornita da Django) in due entità specifiche: Cliente e Staff.  
  
- Cliente estende User aggiungendo dati come codice fiscale, telefono, e lingua preferita.  
- Staff invece rappresenta una persona che lavora per l’azienda con un ruolo ben definito (guida o driver), e può parlare più lingue.  
  
Questo approccio è stato scelto per mantenere una coerenza con il sistema di autenticazione di Django (che usa il modello User), permettendo login e registrazione separati per clienti e staff, e una gestione chiara dei permessi e delle funzionalità.  
  
Relazioni tra entità  
  
- Ogni Cliente può prenotare più tour, ma ogni prenotazione fa riferimento a un solo cliente e un solo tour.  
- Ogni Tour può ricevere più recensioni, ma ogni cliente può recensire un determinato tour una sola volta (vincolo unique\_together).  
- Lo Staff può parlare più lingue (relazione N:N mediata da una tabella Staff\_Lingua).  
- Inizialmente, si era previsto che un Tour potesse essere condotto da più membri dello staff (guida e driver), con una relazione N:N Tour\_Staff. Tuttavia, nella versione finale si è semplificato il modello, distinguendo direttamente nei campi del modello Tour i due ruoli:  
  
 - guide → ManyToMany con filtro ruolo='guida'  
 - driver → ManyToMany con filtro ruolo='driver'  
  
Questa scelta è motivata dal fatto che il numero di guide e driver per tour è limitato e ben definito, e i ruoli sono mutuamente esclusivi. Questo ha permesso una gestione più chiara nei template e nelle interfacce di amministrazione, pur mantenendo una flessibilità nel numero di assegnazioni.

**2. Progettazione Logica**

Dal modello concettuale è stato derivato il seguente modello logico relazionale:  
  
CLIENTE  
PK: cod\_fiscale  
Attributi: nome, cognome, telefono, email  
FK: lingua\_preferita → LINGUA.id  
  
LINGUA  
PK: id  
Attributi: nome\_lingua  
  
STAFF  
PK: cod\_fiscale  
Attributi: nome, cognome, email, telefono, ruolo  
  
STAFF\_LINGUA (relazione PARLA N:N)  
PK: id  
FK: staff\_cod\_fiscale → STAFF.cod\_fiscale  
FK: lingua\_id → LINGUA.id  
  
TOUR  
PK: id  
Attributi: nome, data, durata, prezzo, luogo  
FK: lingua\_id → LINGUA.id  
  
PRENOTAZIONE (relazione PRENOTA)  
PK: id  
FK: cliente\_cod\_fiscale → CLIENTE.cod\_fiscale  
FK: tour\_id → TOUR.id  
Attributi: data, numero\_partecipanti  
  
RECENSIONE (relazione RECENSISCE)  
PK: id  
FK: cliente\_cod\_fiscale → CLIENTE.cod\_fiscale  
FK: tour\_id → TOUR.id  
Attributi: voto, commento, data  
Vincolo: unique(cliente\_cod\_fiscale, tour\_id)  
  
TOUR\_GUIDA / TOUR\_DRIVER (gestite nel modello Tour tramite ManyToMany)  
Relazione N:N implicita filtrata per ruolo  
  
Vincoli  
  
- Un cliente non può recensire un tour più di una volta  
- Un tour può essere prenotato solo se la data è futura  
- Le cancellazioni sono permesse solo prima della data del tour  
- Le recensioni sono consentite solo dopo aver partecipato al tour (data passata e prenotazione esistente)  
- Le assegnazioni staff ai tour rispettano il ruolo previsto

**3. Implementazione del Sistema Informativo**

L'applicazione è stata realizzata con il framework Django, utilizzando:  
  
- Modelli (models.py) per strutturare il database secondo lo schema logico  
- Template HTML per l’interfaccia utente (con Bootstrap)  
- View personalizzate per la logica  
- Form e autenticazione custom per clienti e staff  
- Admin area per la gestione dei dati da parte degli amministratori  
  
Funzionalità implementate  
  
1. Registrazione e autenticazione differenziata  
 - I clienti si registrano con codice fiscale, password e lingua preferita.  
 - Lo staff si registra specificando il ruolo e le lingue parlate.  
  
2. Prenotazione dei tour  
 - I clienti possono prenotare i tour futuri con filtro per lingua preferita.  
 - Ogni prenotazione è visibile nella sezione “I miei tour”.  
  
3. Recensioni dei tour  
 - Dopo aver svolto un tour (data passata), un cliente può lasciare una recensione.  
 - Le recensioni sono pubbliche e visibili a tutti tramite una pagina dedicata.  
  
4. Visualizzazione tour assegnati per staff  
 - Guide e driver vedono solo i tour assegnati a loro, con esclusione di quelli già svolti.  
 - Assegnazioni distinte in base al ruolo.  
  
Estensioni facultative svolte  
  
- Filtro dinamico per lingua nei tour disponibili  
- Conteggio partecipanti visibile per ogni tour  
- Messaggi personalizzati al login, logout e in caso di errore  
- Dashboard utente (tour prenotati, recensioni lasciate)