

Author

Hakash.D

21f2000582

21f2000582@ds.study.iitm.ac.in

I love exploring new ideas and if interest seeks me i'll get well focussed

Description

A ticket booking app with two ends. One for admin and other for the users. Admin can create,delete or update venues with shows for users to book after logging in. Users on other hand can book their tickets via logging in.

Technologies used

Flask, Flask-Sqlalchemy, Flask-login, flask-bcrypt

Flask- Framework and the core

Flask-sqlalchemy - To help us query data from our database(controller)

Flask_JWT - Authentication

Flask-bcrypt - generate hashed password

Redis_Caching - Cache data

Celery - Schedule batch jobs

DB Schema Design

DB MODELS: User, Admin, Show, Venue, Userbooking, Rating

User columns : user_no , username,id,password,admin_no

user_no : Unique index for each user.

username: Username for each user.

id : Unique user id for each user.

password : Password to authenticate users.

timestamp : The time the user logged in.

Admin columns: admin-no,admin_id, password

admin_no : Unique index for each admin.

admin_id : Unique admin id for each admin.

password : Password to authenticate admins.

Show columns: show_no,name,tags,start_time,

end_time,price,venue_no,admin_no,admin_rating

show_no : Unique index for each show.

name : Name of the show.

tags: Show tags.

start_time: start time of the show.

end_time: end time of the show.

admin_rating: The rating given by admin.
admin_no : Admin associated with the show.

Venue columns: venue_no, name, place, location, capacity, admin_no
venue_no : Unique index for each venue.
name : Name of the venue.
place: Sub location.
location : Location of the venue.
capacity: capacity of the venue.
admin_no : Admin associated with the venue.

Userbooking columns: book_id, user_no, show_no, quantity
book_id : Unique index for each booking.
user_no : User who booked the show.
show_no : Show booked by user.
quantity: Number of tickets booked by the user.
timestamp : The time the booking created.

Rating columns: rate_id, user_no, show_no, rating
rate_id : Unique index for each rating.
user_no : User who rated the show.
show_no : Show rated by user.
rating: Rating given by the user.

I have implemented unique indices for each column to easily retrieve them.
Each column has relationships for better querying

API Design
NULL

Architecture and Features

application folder consists of controllers.py, models.py, config.py, database.py
My-custom directory which consists of vue files for frontend inside src folder. The
app.py is stored in root directory

USER END: A search bar based on Preferences, List of Shows and Venues with
housefull and book buttons. Booking page with price and number of tickets as
input and a confirmation page. A page for rating the bookings.

ADMIN END: Various options to delete, update and create shows and venues.

Video

https://drive.google.com/file/d/116BFcocJ8p-8qfTTtVPZ0g96rSy1BawF/view?usp=drive_link