

Booking Lite App

Author: Abhineet Raman(21f1005523@ds.study.iitm.ac.in)

21f1005523

Description: Divided tasks of different functionality into smaller tasks and constantly worked upon each task. Although every time I encountered a bug, tried to fix it by own. If doesn't able to do so, then taking help of different resources like the documentation, few seniors and course instructor and completed the project.

Database Design:

user			Bookings		
CREATE TABLE "user" ("id" INT			CREATE TABLE "Bookings" ("b_		
id	INTEGER	"id" INTEGER	b_id	INTEGER	"b_id" INTEGER
email	TEXT	"email" TEXT UNIQUE	uname	TEXT	"uname" TEXT
password	TEXT	"password" TEXT NOT NULL	t_name	TEXT	"t_name" TEXT NOT NULL
username	TEXT	"username" TEXT NOT NULL	s_name	TEXT	"s_name" TEXT
accessToken	TEXT	"accessToken" TEXT UNIQUE	timing	TEXT	"timing" TEXT NOT NULL
active	INTEGER	"active" INTEGER	quantity	INTEGER	"quantity" INTEGER NOT NULL
urole	TEXT	"urole" TEXT NOT NULL	date	TEXT	"date" TEXT NOT NULL
Show			rating	TEXT	"rating" TEXT NOT NULL
CREATE TABLE "Show" ("id" INT			Profile		
id	INTEGER	"id" INTEGER	CREATE TABLE "Profile" ("uname"		
name	TEXT	"name" TEXT NOT NULL	uname	TEXT	"uname" TEXT NOT NULL
img_link	TEXT	"img_link" TEXT NOT NULL	fname	TEXT	"fname" TEXT
tags	TEXT	"tags" TEXT NOT NULL	lname	TEXT	"lname" TEXT
price	INTEGER	"price" INTEGER NOT NULL	dob	TEXT	"dob" TEXT
rating	REAL	"rating" REAL	phone	INTEGER	"phone" INTEGER
timing	TEXT	"timing" TEXT NOT NULL	Seats		
t_id	INTEGER	"t_id" INTEGER NOT NULL	CREATE TABLE "Seats" ("sl_no" INT		
seats	INTEGER	"seats" INTEGER	sl_no	INTEGER	"sl_no" INTEGER
Theatre			sid	INTEGER	"sid" INTEGER NOT NULL
CREATE TABLE "Theatre" ("id" INT			sname	TEXT	"sname" TEXT NOT NULL
id	INTEGER	"id" INTEGER	total_seats	INTEGER	"total_seats" INTEGER NOT NULL
name	TEXT	"name" TEXT NOT NULL	booked	INTEGER	"booked" INTEGER
place	TEXT	"place" TEXT NOT NULL	available	INTEGER	"available" INTEGER NOT NULL
capacity	INTEGER	"capacity" INTEGER NOT NULL			
location	TEXT	"location" TEXT NOT NULL			

System Design:

- **OS:** Any OS is compatible
- **Coding Languages used:** HTML, CSS, JavaScript(Vue.js), Python(Flask, Flask-SQLAlchemy, Matplotlib, celery, flask-caching, flask-restful, flask-JWT-Extended, Jinja2, SMTPlib and basic modules)
- **Database:** SQLite
- **Platform:** Any Browser which supports HTML5
- **Features:** This project includes 11 components with two different types of users- Admin and Std. User:
 1. **Login Page:** To login into the account using token-based Authentication
 2. **Signup Page:** To create user in database.

3. **Dashboard:** Two different dashboard for two different types of users for different types of tasks respectively.
4. **Add Venue:** A form which takes some data to create Venue which is done only by Admin.
5. **Add Show:** A form which takes some data to create Show which is done only by Admin.
6. **Edit Show:** A form which takes few data to update Show which is done only by Admin.
7. **Summary:** This component provides summary of the different show tickets sold in graphical chart.
8. **Bookings:** This component displays the bookings done by user and allow them to rate the booked Show.
9. **Book Ticket:** This component allows to book the selected show to the user only.
10. **Search:** This component shows searched keywords dashboard.
11. **Profile:** This component updates/Displays the profile info. Of the users.

API Design: API has been designed on the databases used between frontend and backend endpoints where data has been transmitted in JSON format.

Presentation Link: <https://clipchamp.com/watch/PHpqPAfGii8>