Flash Card Application

Author

Sourabh Warrier
21F1002852
21f1002852@student.onlinedegree.iitm.ac.in
Math enthusiast | Interested in Machine Learning and Quantum Computing

Description

The objective of this project is to build a flash card application using the frameworks and concepts learned in the **Modern Application Development -1** course. The name of the project is **FlashStudy** inspired by a similar android app called Kanji Study.

Technologies used

The main technologies used in this application are flask-restful APIs, flask-sqlalchemy, **SQLite**, and **Python**. **Flask-Sqlalchemy** was chosen to interact with the underlying **SQLite** database which makes this application portable. **Flask-Restful** was used in order to deploy simple REST APIs used to handle user inputs. **Jinja2** was used for templating the user interface.

DB Schema Design

This application uses **8 tables** in an **SQLite** database named **database.sqlite3** found in the **data** folder to keep track of and manage various forms of data. An excerpt of the schema is given below :
The full schema is available in the root folder of the application as a file named **schema.pdf**. It can also be accessed at this url :
https://drive.google.com/file/d/1AL1YJMA00t5sHnubng6onS6mPDDb7PEq/view?usp=sharing

Cards Table

Name	Type	Schema
cards		`CREATE TABLE "cards" ( "card_id" INTEGER, "question" TEXT NOT NULL, "answer" TEXT NOT NULL, "deck_id" INTEGER NOT NULL, PRIMARY KEY("card_id" AUTOINCREMENT), FOREIGN KEY("deck_id") REFERENCES "decks"("deck id") )`
card_id	INTEGER	"card_id" INTEGER
question	TEXT	"question" TEXT NOT NULL
answer	TEXT	"answer" TEXT NOT NULL
deck_id	INTEGER	"deck_id" INTEGER NOT NULL

API Design

This application uses 7 REST APIs in total to collect and process information submitted by the user. All of these are setup such that only post methods are defined for them. The reason for this choice is that html forms used in this application do not support **put** and **delete** requests. Hence All unsupported requests are treated as **post** requests and handled by these APIs.

Architecture and Features

The entry point is the shell script **start.sh** that is located in the root folder of the application (the folder called **final-project**). The README file contains information on how to start the application. Interactions with the database are handled using models and functions. The models are given inside the **models** folder in **models.py** and the functions are given in the **controllers** folder in **functions.py**. The basic setup and configuration files reside in the **application** and **db** folders. Some miscellaneous data and the sqlite3 database reside in the **data** folder. All the APIs reside in the **api** folder in **api.py.** The html and css files reside in the **templates** and **static** folders respectively.

This application has a **quiz** feature where users can take a quiz on the flash cards by choosing which deck to be quizzed on. The scores they obtain will be displayed on a results page along with recommendations as to which cards to study again. The cumulative scores of the user are maintained deckwise and displayed on the dashboard along wth the average rating of deck that the user has previously studied. A graph representing the user's performance on each card of each deck is also displayed with the deck on the dashboard.

The application can be accessed at this url : https://flashstudy.21f1002852.repl.co/

Video

This is a link to the presentation video :
https://drive.google.com/file/d/1vhmuoBNKzuIk3ZENrpZApT9lIhwTKs4G/view?usp=sharing