Author

Sriram Kashyap Kurilla
21F1005107
21f1005107@student.onlinedegree.iitm.ac.in
I am a student of BITS PIlani – Hyderabad Campus in my Second Year studying Electronics and Communication Engineering

Description

The project is about creating a Flashcards Application, and designing and implementing the models, views, controllers, and the database used to store the data.

Technologies used

Some of the extensions/modules used:
flask_login – handle and implement login
flask_sqlalchemy – interact with the sql database
flask_restful – interaction with restful api
werkzeug.security – password hashing
datetime – record date and time
Flask – main application code
Bootstrap – Styling
Jinja2
Vuejs
Flask - caching
SMTP servers - fake email hosting

DB Schema Design

Name	Type	Schema
card		CREATE TABLE "card" ("card_id" INTEGER NOT NULL, "front" VARCHAR(512) NOT NULL, "back" VARCHAR(512) NOT NULL, "score" INTEGER, "deck" VARCHAR NOT NULL, PRIMARY KEY("card_id"), FOREIGN KEY("deck") REFERENCES "deck"("deck_name"))
card_id	INTEGER	"card_id" INTEGER NOT NULL
front	VARCHAR(512)	"front" VARCHAR(512) NOT NULL
back	VARCHAR(512)	"back" VARCHAR(512) NOT NULL
score	INTEGER	"score" INTEGER
deck	VARCHAR	"deck" VARCHAR NOT NULL
deck		CREATE TABLE "deck" ("id" INTEGER NOT NULL, "deck_name" VARCHAR(30), "user" VARCHAR NOT NULL, "date_created" DATETIME, "score" INTEGER, "is_public" BOOLEAN, "last_rev" DATETIME, PRIMARY KEY("id"), FOREIGN KEY("user") REFERENCES "user"("username"))
id	INTEGER	"id" INTEGER NOT NULL
deck_name	VARCHAR(30)	"deck_name" VARCHAR(30)
user	VARCHAR	"user" VARCHAR NOT NULL
date_created	DATETIME	"date_created" DATETIME
score	INTEGER	"score" INTEGER
is_public	BOOLEAN	"is_public" BOOLEAN
last_rev	DATETIME	"last_rev" DATETIME
export		CREATE TABLE export (id INTEGER NOT NULL, filename VARCHAR(50), data BLOB, PRIMARY KEY (id))
id	INTEGER	"id" INTEGER NOT NULL
filename	VARCHAR(50)	"filename" VARCHAR(50)
data	BLOB	"data" BLOB
user		CREATE TABLE "user" ("id" INTEGER NOT NULL, "username" VARCHAR(30) NOT NULL UNIQUE, "password" VARCHAR(300) NOT NULL, "date_created" DATETIME, "score" INTEGER, "email" VARCHAR(30) NOT NULL UNIQUE, "last_rev" DATETIME, PRIMARY KEY("id"), UNIQUE("username"))
id	INTEGER	"id" INTEGER NOT NULL
username	VARCHAR(30)	"username" VARCHAR(30) NOT NULL UNIQUE
password	VARCHAR(300)	"password" VARCHAR(300) NOT NULL
date_created	DATETIME	"date_created" DATETIME
score	INTEGER	"score" INTEGER
email	VARCHAR(30)	"email" VARCHAR(30) NOT NULL UNIQUE
last_rev	DATETIME	"last_rev" DATETIME

API Design

The API has been created and is being interacted with using Flask-restful

Cards:

- Create a new card in the deck
- Review card

Decks:

- Create a new deck for the particular user
- Read, update or delete a deck
- Export deck as csv

Users:

- Create a new user where each user is associated with an account

Architecture and Features

The app is started by running run.py which imports create_app from main.py. The APIs are defined inside api.py and the pages are initiated from views.py. The controllers are present inside views.py.

The app opens up into the landing screen. This page gives user to access both login screen For existing users, and signup screen for new users. These have two different html screens login.html and register.html for the same purpose. Some constraints regarding username and password have been provided. Once logged in, the user is redirected into dashboard.html where new decks can be created.

There is a logout function provided on top too.

Clicking the Add cards/Review button, redirects user to review.html screen where user can add cards into the deck. Cards can be reviewed by hovering over them to see the answer and then rating each card according to difficulty.

Additionally, user authentication is also required to run the app, so pasting the dashboard or review URL will not work and will redirect the user to login screen with a login required message. All the templates are in the templates folder from which the flask app uses them. CRUD on decks have been applied. APIs to get scores have also been provided.

A proper login system, validation for username and password is also provided

You can export a deck as CSV by clicking on the export button and the CSV file will be automatically downloaded

Backend jobs for daily reminders and monthly report have also been created

Video

https://drive.google.com/file/d/1K403vjhlTnIFLZ8Uy9CQIDs01j10mH7j/view?usp=sharing