Author

Vetrivel M.D
21F1004921
21f1004921@student.onlinedegree.iitm.ac.in
I am currently pursuing MSc Cybersecurity (Integrated Course) from PSG College of Technology

Description

The final course project is to develop a website where people can create flashcards which can be used for memory training this website should be able to manage group of cards (known as deck) and provide an analysis on how well the user remembers the answers for the created flash cards

Technologies used

flask
Vue.js
Chart.js
flask-jwt-extended
flask-cors
flask-SQLAlchemy
SQLAlchemy
cryptography
flask_restful

Architecture and Features

FRONTEND:
- HTML files are within the folder templates
- JavaScript files are within the folder script
- CSS files are within the folder static
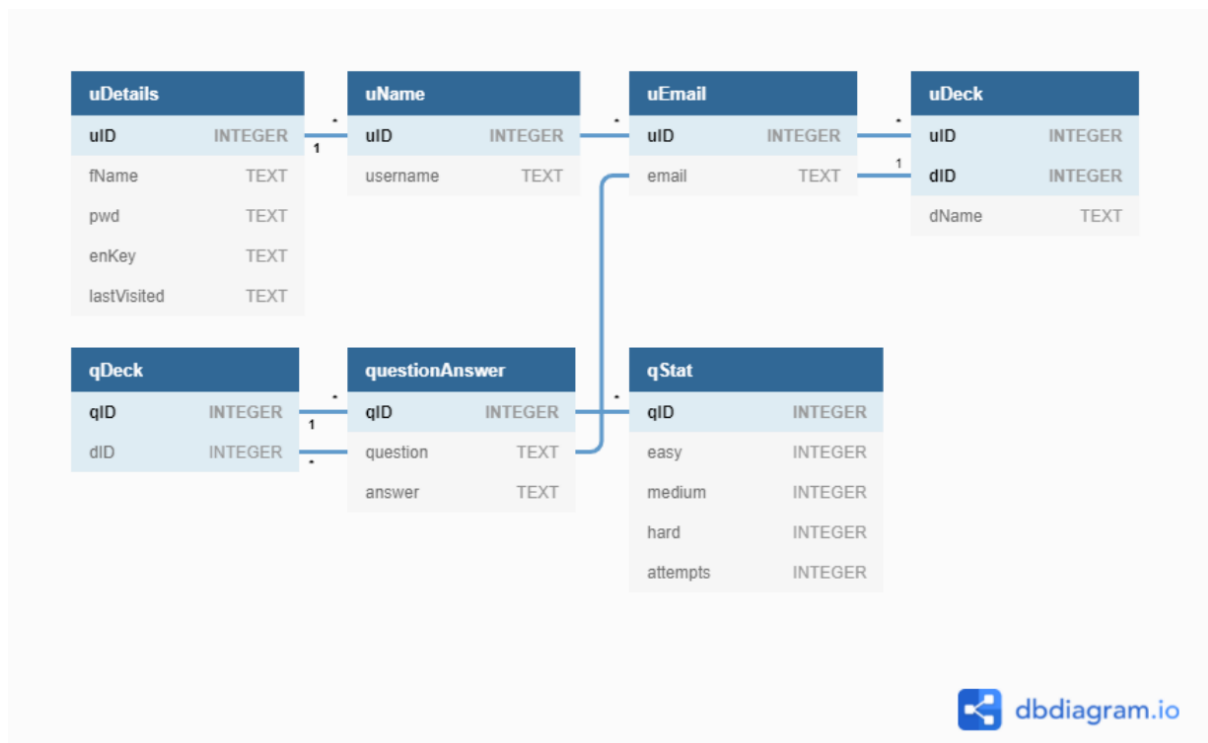- Logo of the website in within the folder logo

BACKEND:
- api.py file contains the api definition
- FlashcardDB is the SQLite database

➔ You can create an account by providing your name, username, email and password (JWT Token based Authentication)
➔ You can create a deck which is a collection of cards
➔ You can add cards to a deck

➔ You can study a deck and provide feedback for each question on how well you remember the answer for the questions.
➔ You can analyse your performance by looking at the graph provided.
➔ You can edit the name of a deck and the question and answers provided in the cards of the deck.
➔ You can delete a deck or a card
➔ You can change your name, username and password

DB Schema Design

API Design

➔ /api/login (POST, GET) – Used for authenticating valid user
➔ /api/signup (POST) – To create a new user
➔ /api/deck (POST, GET, PUT, DELETE) – To perform CRUD operations on decks
➔ /api/card (POST, GET, PUT, DELETE) – To perform CRUD operations on cards
➔ /api/user (GET, PUT) – To get (or) update User details
➔ /api/stat (GET, POST) – To update (or) get a user's statistics
➔ /api/getCard (GET) – To get the current users card.

Video

https://drive.google.com/file/d/19eqfEGHU89OEdSOX8RE1Vw7rdylPJpyC/view?usp=sharing