



# QuickFlash

A fast and minimal flashcard application

## CONTACT INFORMATION

Sheikh Uzair Hussain

Roll Number: 21f1001254

Email: [21f1001254@student.onlinedegree.iitm.ac.in](mailto:21f1001254@student.onlinedegree.iitm.ac.in)

## ABOUT THE AUTHOR

I am a 21-year old undergrad student passionate about leveraging technology to solve hard problems, empower people, and make the world a better place. Apart from this BSc degree, I am pursuing a BTech in Computer Science & Engineering (2nd year).



## Description

QuickFlash is a modern and minimal flashcard app that enables users to train their memory and track their progress over time. Users can add decks (with cards inside them) and take reviews for each deck. It features a slick user interface and experience that makes using this application a breeze.

### Dummy User:

Email: uzair.hussain10@gmail.com

Password: test@123



## Technologies used

QuickFlash uses the following tools and technologies on the back end:

1. Python 3: Back-end scripting language.
2. SQLite: Database Management System.
3. Flask: Back-end framework.
4. Flask-SQLAlchemy: ORM/DB library.
5. Flask-Login: Authentication and user management.
6. Flask-Mail: SMTP library for Flask.
7. Celery: Async and scheduled job handling library.
8. Redis: Job broker for Celery.

And on the front end, it uses

1. HTML, SASS, CSS, JavaScript: Front end languages.
2. VueJS: Front-end framework.
3. Bootstrap 5 + Bootstrap Icons: Front-end components.
4. Vuex: App-wide state management library.
5. Vue Router: Routing library for Vue.
6. Vite: ESBuild-based bundler.
7. PapaParse: CSV parser in JavaScript.
8. Date-FNS: Date manipulation library in JavaScript.

# DB Schema Design

## users

id integer (PK)	name varchar(100)	email varchar(100) (unique)	password char(72)
1	Uzair Hussain	uzair@gmail.com	sha256\$AXe24cNTv...
2	आयुष	aayush@gmail.com	sha256\$AXe24cNTv...

## decks (belongs to user)

id - integer (PK)	user_id - integer (FK on users.id)	name - varchar(100)
1	1	Uzair's Animal Deck
2	2	आयुष के कार्ड डेक

## cards (belongs to deck)

id - integer (PK)	deck_id- integer (FK on decks.id)	front varchar(100)	back varchar(100)	points integer
1	1	Dog	कुट्टा	5
2	1	Cat	बिल्ली	7

## reviews (belongs to deck and user)

id - integer (PK)	user_id - integer (FK on users.id)	deck_id- integer (FK on decks.id)	taken_at datetime
1	1	1	2021-11-28T03:29:24.817263
2	1	1	2021-11-28T03:29:24.817263

## review\_cards (belongs to review and card)

id - integer (PK)	review_id - integer (FK on reviews.id)	card_id- integer (FK on cards.id)	difficulty integer	score integer
1	1	1	1	5
2	1	1	5	25



## API Design

QuickFlash has a RESTful API built with Flask. It allows the JavaScript frontend to communicate with the application. It uses a simple bearer token system for authentication. The associated file has been included in the ZIP file containing the project.



## Architecture and features

The back application logic is broken down into four entities with corresponding backend and frontend modules:

1. Authentication  
**Model:** models.py → User  
**Views:** client/src/views/auth/  
**Controller:** auth.py
2. Dashboard  
**Model:** None  
**Views:** client/src/views/dashboard/  
**Controller:** dashboard.py
3. Deck  
**Model:** models.py → Deck, Card  
**Views:** client/src/views/decks/  
**Controller:** deck.py
4. Review  
**Model:** models.py → Review, ReviewCard  
**Views:** client/src/views/reviews/  
**Controller:** review.py

### Authentication:

The authentication controller handles things like logging in, logging out, and user registration. It provides an **auth** blueprint containing the relevant routes.

### Dashboard:

The dashboard controller only has a single route i.e. the actual dashboard. It returns the data for the dashboard view (decks and reviews). It provides a **dashboard** blueprint containing the relevant routes.

### Deck:

The deck controller provides a CRUD-ish interface for the card decks added by the user. It allows the users to create, import, list, view, export, update and delete a deck and its cards. It provides a **deck** blueprint containing the relevant routes.

### Review:

The review controller provides a partial CRUD interface to take a review of any deck added by the user. It allows the users to take a review, view detailed review reports, and delete a review. It provides a **review** blueprint containing the relevant routes.

Apart from these entities, there are the usual Flask files and a few more files for miscellaneous tasks such as Celery jobs. The database file is named **quickflash.sqlite3** and is an SQLite database file that is generated automatically if it's missing, with a dummy user.



## Video

[https://drive.google.com/drive/folders/1JrRsVdHIk2NCZEjbfA1afeJe2zuxtwSr?usp=s\\_haring](https://drive.google.com/drive/folders/1JrRsVdHIk2NCZEjbfA1afeJe2zuxtwSr?usp=s_haring)



## Git Repository

<https://github.com/sheikhuzairhussain/quickflash>