

Simple Flashcards
Vývojářská dokumentace

Jan Jeníček

Obsah

1	O aplikaci	3
2	Frontend (simpleflashcards-client)	3
2.1	Struktura kódu	3
2.2	Základní knihovny	4
2.3	Instalace a úprava aplikace	4
2.4	Použité knihovny	4
3	Backend (firebase-functions)	5
3.1	Struktura kódu	5
3.2	Databáze	6
3.3	Struktura databáze	6
3.3.1	collections	6
3.3.2	decks	7
3.3.3	users	8
3.3.4	deckProgress	8
3.4	Uživatelské účty	9
3.4.1	Registrace	9
3.4.2	Přihlašování a odhlašování	9
3.5	Instalace a úprava funkcí	9
3.6	Použité knihovny	9
4	Odkazy	10
4.1	Simple Flashcards	10
4.2	Využité služby	10

1 O aplikaci

Simple Flashcards je webová aplikace určená k jednoduchému učení pojmů (např. slovíček cizího jazyka).

Projekt je rozdělen do dvou částí:

- **firebase-functions** - API, které bylo vytvořeno pro komunikaci mezi Cloud Firestore databází a klientem.
- **simpleflashcards-client** - Samotná Simple Flashcards webová aplikace.

Tato dokumentace je zaměřena na popis struktury projektu. Podrobný popis jednotlivých funkcí lze najít na webových stránkách této dokumentace (viz sekce Odkazy).

2 Frontend (simpleflashcards-client)

Tato část dokumentace je zaměřena na samotnou webovou aplikaci a popis jednotlivých komponentů a funkcí, které komunikují s API serveru. Samotná aplikace běží na hostingu služby Firebase Hosting.

2.1 Struktura kódu

Webová aplikace je rozdělena do následujících částí:

- **Components** - Zde můžete najít jednotlivé komponenty této aplikace.
- **Pages** - Zde můžete najít komponenty aplikace, které tvoří dané stránky. K tomu využívají komponenty ze složky components.
- **ReduxActions** - Zde můžete najít veškeré funkce, které komunikují s API serveru a s Redux reducery (čímž mění hodnoty ve Redux store).
- **ReduxReducers** - Zde můžete najít všechny Redux reducery.
- **Util** - Zde můžete najít Javascriptové moduly, které jsou používány ostatními částmi aplikace.
- **Ostatní** - Zde jsou ostatní moduly aplikace.

2.2 Základní knihovny

Tento projekt je vytvořen pomocí těchto nejpodstatnějších knihoven: React, Redux, Axios, Material UI. Od toho se odvíjí struktura celé aplikace.

React

Aplikace je sestavena z jednotlivých komponentů. Popis každého komponentu aplikace můžete najít v této dokumentaci v sekci Pages nebo Components.

Redux

Tato knihovna je klíčovou součástí aplikace. Umožňuje přehledné předávání informací mezi danými komponenty aplikace. Díky této knihovně je vytvořen tzv. store. Na tomto místě jsou uložena data aplikace. Když potřebuje např. nějaký komponent tato data získat, stačí mu přistoupit do storu a nemusí je tak získávat od ostatních komponentů. Jednotlivé funkce můžete najít v této dokumentaci v sekci ReduxActions a ReduxReducers.

Axios

Axios je knihovna, která zajišťuje komunikaci mezi klientem a serverem. Využívají ji funkce této aplikace pro komunikaci s API serveru. Tyto funkce můžete najít v sekci ReduxActions.

Material UI

Samotný vzhled aplikace je zajištěn pomocí knihovny Material UI. Tato knihovna umožňuje rychlé a přehledné vytváření komponentů s jednotným vzhledem.

2.3 Instalace a úprava aplikace

Pro vlastní úpravu aplikace stačí stáhnout složku simpleflashcards-client z Githubu. V té je pak třeba spustit příkaz `npm install`. Pro spuštění aplikace lokálně se pak dá použít příkaz `npm start` a pro sestavení aplikace příkaz `npm run build`.

2.4 Použité knihovny

- @material-ui/core
- @material-ui/icons

- axios
- jwt-decode
- material-table
- query-string
- react
- react-dom
- react-redux
- react-router-dom
- react-scripts
- redux
- redux-thunk
- better-docs
- jsdoc

3 Backend (firebase-functions)

Tato část dokumentace je zaměřena na Cloud Functions, tedy na funkce, které tvoří API projektu a manipulují s daty v Cloud Firestore databázi na základě požadavků webové aplikace. Funkce běží na serverech služby Firebase.

3.1 Struktura kódu

Funkce

- **collection** - Funkce, které se starají o vytváření, upravování, připínání a odepínání kolekcí a získávání dat kolekcí.
- **collectionCards** - Funkce, které získávají z databáze karty na učení z dané kolekce a ukládají pokrok uživatele.
- **deck** - Funkce, které se starají o vytváření, upravování, připínání a odepínání balíčků a získávání dat balíčků.

- **deckCards** - Funkce, které získávají z databáze karty na učení z daného balíčku a ukládají pokrok uživatele.
- **functions** - Funkce používané dalšími moduly.
- **user** - Funkce spravující data uživatelů.

Ostatní

- **index** - Javascript modul, v němž se z funkcí vytvoří API.

Util

- **admin** - Javascript modul pro spuštění Firebase Admin SDK.
- **authMiddleware** - Funkce, která slouží k ověření uživatele.
- **other** - Konstanty používané ve funkcích.

3.2 Databáze

Tento projekt využívá jako databázi službu Cloud Firestore. Jedná se o NoSQL databázi. Projekt využívá také službu Cloud Functions. Tato služba umožňuje vytvořit API a naprogramovat funkce, díky kterým se dá do databáze přistupovat. V této dokumentaci je zaznamenána struktura databáze a popis jednotlivých funkcí.

3.3 Struktura databáze

Databáze je rozdělena do 3 částí:

- **collections** - Dokumenty jednotlivých kolekcí.
- **decks** - Dokumenty jednotlivých balíčků.
- **users** - Dokumenty uživatelů a dokumenty se zaznamenanými pokroky u daných balíčků.

3.3.1 collections

V této části databáze jsou uloženy všechny kolekce. Každý dokument obsahuje data jedné kolekce a má název podle ID dané kolekce. Tyto ID jsou automaticky generovány funkcemi z Cloud Firestore. V dokumentech jsou uloženy následující informace:

- [string] colName - Název kolekce.
- [string] colDescription - Popis kolekce, který uživatel zadal. Pokud uživatel nezadal popis kolekce, má hodnotu null.
- [string] creatorId - ID uživatele, který kolekci vytvořil.
- [boolean] private - Informace, která udává, jestli je kolekce veřejná nebo soukromá.
- [Array<string>] deckArray - Pole, které obsahuje ID balíčků, které jsou obsaženy v dané kolekci.

3.3.2 decks

V této části databáze jsou uloženy všechny balíčky. Každý dokument obsahuje data jednoho balíčku a má název podle ID daného balíčku. Tyto ID jsou automaticky generovány funkcemi z Cloud Firestore. V dokumentech jsou uloženy následující informace:

- [string] deckName - Název balíčku.
- [string] deckDescription - Popis daného balíčku, který uživatel zadal. Pokud uživatel nezadal popis balíčku, má hodnotu null.
- [string] deckImage - Adresa obrázku daného balíčku. Pokud uživatel žádný balíček nenahrál, je zde zapsána adresa pro výchozí obrázek.
- [string] creatorId - ID uživatele, který balíček vytvořil.
- [boolean] private - Informace, která udává, jestli je balíček veřejný nebo soukromý.
- [Array<Object>] cardArray - Pole, které obsahuje objekty jednotlivých karet balíčku. Každý objekt obsahuje následující informace:
 - [string] cardId - ID dané kartičky, které bylo vygenerováno funkcí newId.
 - [string] body1 - Text přední strany kartičky.
 - [string] body2 - Text zadní strany kartičky.

3.3.3 users

V této části jsou uloženy všechny informace o uživateli a jejich pokroku v učení. Každý dokument obsahuje data jednoho uživatele a má název podle ID daného uživatele. ID jsou automaticky generovány službou Firebase Authentication při registraci uživatele. V jednotlivých dokumentech této části databáze jsou pak uloženy následující informace:

- [string] username - Uživatelské jméno.
- [string] bio - Popis uživatelského profilu.
- [string] email - Emailová adresa uživatele.
- [Array<string>] createdDecks - Pole, které obsahuje ID všech balíčků, které uživatel vytvořil.
- [Array<string>] pinnedDecks - Pole, které obsahuje ID všech balíčků, které uživatel připnul.
- [Array<string>] createdCollections - Pole, které obsahuje ID všech kolekcí, které uživatel vytvořil.
- [Array<string>] pinnedCollections - Pole, které obsahuje ID všech kolekcí, které uživatel připnul.

3.3.4 deckProgress

Ke každému souboru v této části databáze (users) je přiřazena skupina dokumentů, která se nazývá deckProgress. Dokumenty těchto skupin jsou určeny k uchovávání pokroku uživatele u balíčků a mají název podle ID daného balíčku. V každém dokumentu je pak uložen seznam karet balíčku, které se uživatel učil, společně se zaznamenaným pokrokem u každé karty. V každém dokumentu jsou tedy uloženy následující informace:

- [string] deckId - ID daného balíčku (Je identické s názvem dokumentu. V dokumentu je zapsáno, aby bylo možné dokumenty hromadně prohledávat v databázi.).
- [Array<Object>] cardArray - Pole, které obsahuje objekty karet balíčku, které se uživatel učil. Ty obsahují následující informace:
 - [string] cardId - ID dané kartičky.
 - [number] understandingLevel - Číslo, které udává, jak dobře uživatel rozumí dané kartičce.

3.4 Uživatelské účty

3.4.1 Registrace

Když se uživatel zaregistruje do aplikace, je zaregistrován do služby Firebase Authentication. Zde jsou pak uloženy např. přihlašovací údaje uživatele (email a heslo) nebo ID uživatele. Zároveň je při registraci vytvořen dokument v databázi v části users. V tomto dokumentu se pak ukládají ostatní informace o uživateli (viz výše).

3.4.2 Přihlašování a odhlašování

Přihlášení uživatele probíhá takto: Klient zažádá o přihlášení. Cloud funkce získá ze služby Firebase Authentication idToken uživatele a ten pak odešle zpět klientovi. Tam se pak idToken uloží do localStorage a je přidán do headeru dalších požadavků na server. Při zpracovávání těchto požadavků se pak vždy ověří platnost daného idTokenu. To zajišťuje funkce authMiddleware.

Při odhlášení uživatele je idToken odstraněn z localStorage klienta a z headeru požadavků.

3.5 Instalace a úprava funkcí

Pro užití funkcí pro vlastní projekt je třeba vytvořit a nastavit nový Firebase projekt (více zde).

Po vytvoření projektu příkazem `firebase init functions` stačí do adresáře projektu přidat složku functions a v této složce pak spustit příkaz `npm install`. Po případné úpravě kódu daných funkcí je třeba soubory nahrát na server příkazem `firebase deploy`.

3.6 Použité knihovny

- busboy
- cors
- express
- firebase
- firebase-admin
- firebase-functions
- better-docs

- `firebase-functions-test`
- `jsdoc`

4 Odkazy

4.1 Simple Flashcards

- **Simple Flashcards** - <https://simpleflashcards-4aea0.firebaseio.com>
- **Stránka o aplikaci** - <https://morcinus.github.io/SimpleFlashCards/>
- **Uživatelská dokumentace** - <https://morcinus.github.io/simpleflashcards-user-docs/>
- **Vývojářská dokumentace pro front-end (simpleflashcards-client)** - <https://morcinus.github.io/simpleflashcards-client-docs/>
- **Vývojářská dokumentace pro back-end (firebase-functions)** - <https://morcinus.github.io/simpleflashcards-fb-functions-docs/>
- **Github** - <https://github.com/Morcinus/SimpleFlashCards>

4.2 Využité služby

- **Cloud Firestore** - NoSQL databáze, ve které jsou uložena všechna data aplikace.
- **Firebase Cloud Functions** - Služba, ve které jsou uloženy funkce, které manipulují s daty v Cloud Firestore databázi.
- **Firebase Storage** - Místo, na které se ukládají všechny obrázky, které uživatelé nahrají do aplikace.
- **Firebase Authentication** - Služba, která spravuje uživatelské účty této aplikace.
- **Firebase Hosting** - Hosting, na kterém běží tato aplikace.