



Internetové aplikace
Manažérsky kalendár

Autori: Samuel Oleksák (xoleks00), Michal Findra (xfindr00)
Akademický rok: 2022/23

Obsah

1	Zadanie	2
2	Použité knižnice a nástroje	2
2.1	Odôvodnenie výberu nástrojov	2
2.2	npm run ciele	2
3	Interakcia s backendom	3
3.1	Hooky na vytváranie HTTP žiadostí	3
4	Autentizácia	4
5	Implementáčny detaily	4
5.1	Cesty	4
5.2	Hlavné komponenty	4
5.2.1	Panel nástrojov (Toolbar)	4
5.2.2	Modálne okno udalosti	5
5.2.3	Implementácia pohľadu mesiaca	6
5.2.4	Implementácia pohľadu týždňa	6
5.2.5	Detail udalosti	7
5.2.6	Implementácia správy používateľov	8
5.2.7	Notifikácie	9
5.3	Pomocné súbory	10
6	Testovanie	10

1 Zadanie

Ředitel vaší firmy se na vás obrátil s požadavkem na vývoj aplikace, kterou charakterizoval takto:

Systém bude používat vedení firmy, tj. ředitel a vedoucí oddělení (manažeri) a sekretářky na obou úrovních. Ředitel a manažeri, případně jejich sekretářky budou vkládat informace o plánovaných akcích. Ředitel má možnost vidět kromě svého kalendáře i kalendáře všech manažerů, ti vidí pouze svoje, z ředitelova dostávají pouze informaci o termínech, kdy nebude ředitel přítomen. Ředitel má možnost manažerům sám některé akce plánovat. Systém musí v takovém případě zajistit odeslání zprávy příslušnému manažerovi. Systém také musí poskytovat informace o volných časech vybraných či všech manažerů, aby ředitel mohl plánovat některé společné akce.

2 Použité knižnice a nástroje

Frontend projektu je vypracovaný v jazyku JavaScript s využitím knižnice *React.js* a designového systému *PatternFly*. End-to-end testovanie je uskutočňované pomocou nástroja *Cypress*. Projekt vyžaduje *node.js* verzie 14. Na statickú kontrolu sa využíva knižnica *ESLint*.

2.1 Odôvodnenie výberu nástrojov

Dôvodom výberu knižníc je ich vysoká popularita, čo sa premieta v ich kvalite, množstve funkcionality a množstve dostupných informačných zdrojov. Designový systém *PatternFly* bol zvolený pre veľké množstvo poskytovaných komponentov používateľského rozhrania vhodných pre implementáciu tohto konkrétneho projektu, ako sú napríklad komponenty na výber času a dátumu.

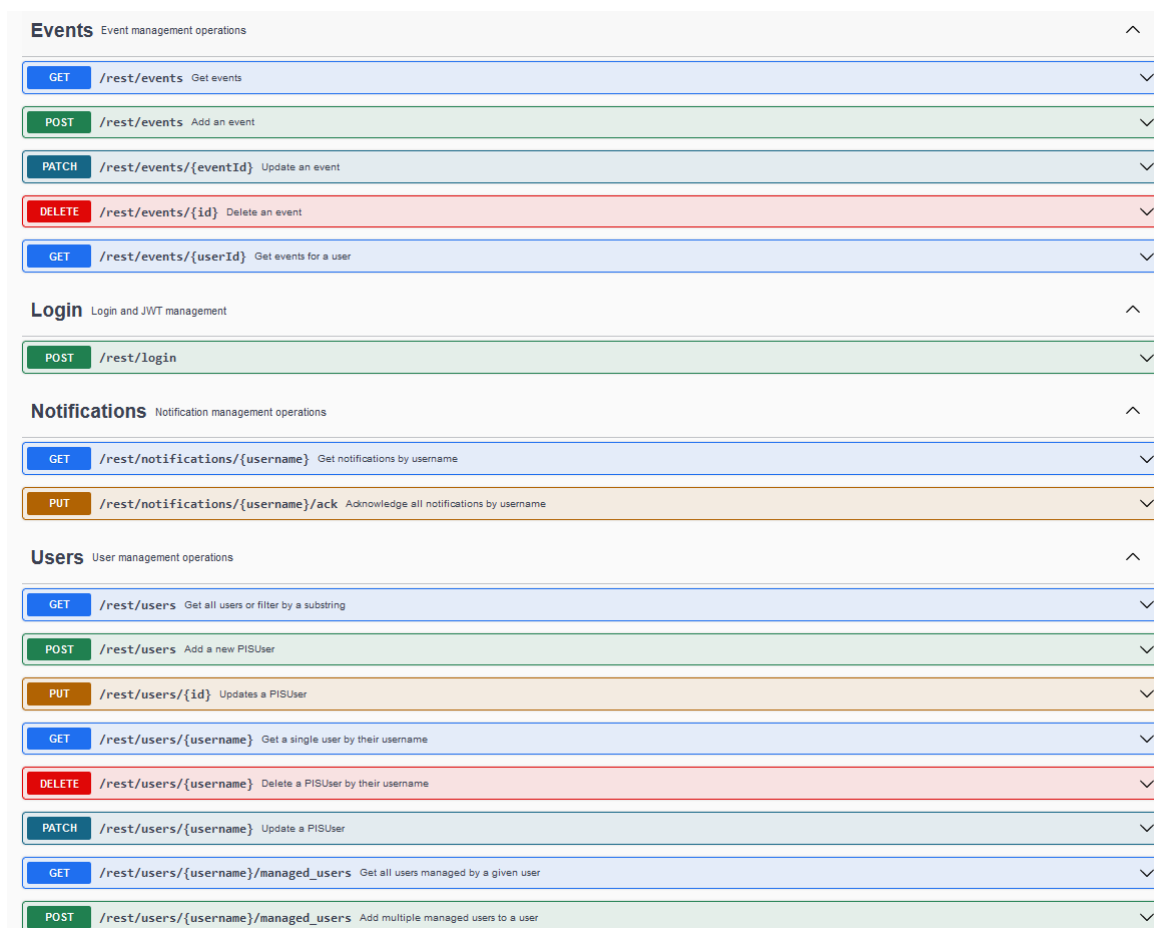
2.2 npm run ciele

Aplikácia definuje nasledujúce `npm run` ciele:

- `npm run start` – spustí aplikáciu,
- `npm run lint` – spustí statickú kontrolu kódu,
- `npm run lint:fix` – spustí statickú kontrolu kódu a pokúsi sa opraviť niektoré chyby,
- `npm run e2e` – spustí testy v interaktívnom režime,
- `npm run test` – spustí testy v terminálovom režime.

3 Interakcia s backendom

Backendová aplikácia poskytuje REST rozhranie. Na komunikáciu sa využíva knižnica *Axios*.



Events Event management operations		^
GET	/rest/events	Get events
POST	/rest/events	Add an event
PATCH	/rest/events/{eventId}	Update an event
DELETE	/rest/events/{id}	Delete an event
GET	/rest/events/{userId}	Get events for a user
Login Login and JWT management		^
POST	/rest/login	
Notifications Notification management operations		^
GET	/rest/notifications/{username}	Get notifications by username
PUT	/rest/notifications/{username}/ack	Acknowledge all notifications by username
Users User management operations		^
GET	/rest/users	Get all users or filter by a substring
POST	/rest/users	Add a new PISUser
PUT	/rest/users/{id}	Updates a PISUser
GET	/rest/users/{username}	Get a single user by their username
DELETE	/rest/users/{username}	Delete a PISUser by their username
PATCH	/rest/users/{username}	Update a PISUser
GET	/rest/users/{username}/managed_users	Get all users managed by a given user
POST	/rest/users/{username}/managed_users	Add multiple managed users to a user

Obr. 1: Zoznam endpointov poskytovaných backendom.

3.1 Hooky na vytváranie HTTP žiadostí

Na vytváranie HTTP žiadostí slúžia hooky `useFetch` a `useAction`. `useFetch` zabaľuje GET žiadosť a vracia premennú s výsledkom žiadosti, pravdivostnú hodnotu udávajúcu, či sa žiadosť už zodpovedala alebo nie a funkciu, ktorej volanie znovu zašle GET žiadosť s novými parametrami. Implementácia obnovovacej funkcie

je nasledovná – obnovovacia funkcia pri svojom volaní inkrementuje čítač. `axios.get` funkcia sa nachádza v `useEffect` funkcií. Čítač sa nachádza v `sensitivity` liste daného `useEffect` hooku a pri každej zmene čítača sa vyšle HTTP žiadosť.

Na vytváranie HTTP žiadostí typu POST, PUT, PATCH a DELETE je použitý hook `useAction`. Zavolanie hooku vráti funkciu, ktorej volanie spustí HTTP žiadosť s vloženými parametrami. Toto umožňuje opakované vytváranie HTTP žiadostí bez nutnosti poskytovať veľké množstvo údajov pri každom a jednom volaní.

4 Autentizácia

Na autentifikáciu sa používa JWT. JWT získaný z `/login` endpointu pomocou prihlasovacieho modálneho okna je uložený v `localStorage`. Pri obnovení stránky sa JWT načíta z `localStorage`, skontroluje sa jeho platnosť a ak je platný, tak sa používa na ďalšie HTTP žiadosti. Počas behu aplikácie je token uložený pomocou `useState` a zasiela sa do hookov na vytváranie HTTP žiadostí. Na dekódovanie JWT je použitá knižnica `jwt-decode`¹.

5 Implementačné detaily

5.1 Cesty

Aplikácia poskytuje dve cesty. Navigáciu medzi nimi zabezpečuje knižnica *React router DOM*, čo umožňuje, aby aplikácia bola SPA. Jedna z ciest slúži na prezera-
nie kalendára a druhá cesta, ktorá je prístupná iba správcovi používateľov, slúži na pridávanie, úpravu a vymazávanie používateľov.

5.2 Hlavné komponenty

5.2.1 Panel nástrojov (Toolbar)

Panel nástrojov umožňuje interakciu používateľa s kalendárom – poskytuje tlačidlá na navigáciu doprava a doľava v kalendári; umožňuje prepínať medzi zobrazením týždňa a mesiaca, zobrazíť kalendáre iných používateľov, vytvoriť novú údalosť a skryť svoj vlastný kalendár (vhodné pri prezeraní kalendárov iných používateľov).

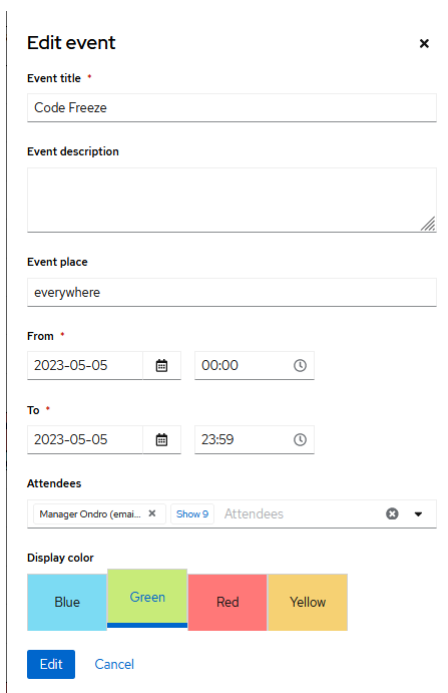
Tlačidlá navigácie zasielajú správy kalendárovým komponentom pomocou inkrementácie čísla uloženého v stave komponentu. Kalendárové komponenty správy zachytávajú s použitím `useEffect` citlivého na dané číslo.

¹<https://www.npmjs.com/package/jwt-decode>

Zobrazovanie kalendárov iných osôb je implementované pomocou komponentu Typeahead, ktorý je podobný multi-select dropdownu, ale umožňuje používateľovi vyhľadávať v položkách. Ostatní zamestnanci sa dajú vyhľadávať podľa vlastného mena, používateľského mena alebo emailovej adresy. Vyhľadávanie v zozname používateľov prebieha na strane frontendu a je realizované pomocou konverzie reťazcov do NFD² a vyhľadávaním podreťazca. Používatelia (ktorí nemajú rolu riaditeľ) si smú zobraziť iba kalendár svojich priradených manažérov, preto sa zoznam všetkých osôb filtruje a zobrazujú sa iba osoby, ktorých kalendár sa dá vidieť.

5.2.2 Modálne okno udalosti

Modálne okno udalosti umožňuje vytvárať nové udalosti alebo upravovať už existujúce udalosti. Modálne okno si vo svojom komponente ukladá pomocou `useState` stav svojich jednotlivých polí.



The image shows a modal window titled "Edit event" with a close button (X) in the top right corner. The form contains the following fields and sections:

- Event title ***: A text input field containing "Code Freeze".
- Event description**: A large text area for description.
- Event place**: A text input field containing "everywhere".
- From ***: A date and time selector showing "2023-05-05" and "00:00".
- To ***: A date and time selector showing "2023-05-05" and "23:59".
- Attendees**: A section with a search bar containing "Manager Ondro (email...)" and a "Show 9 Attendees" button.
- Display color**: A row of four color-coded buttons: "Blue", "Green" (which is highlighted with a blue border), "Red", and "Yellow".
- At the bottom, there are two buttons: "Edit" (in blue) and "Cancel" (in light blue).

Obr. 2: Modálne okno udalosti počas úpravy udalosti.

²https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/String/normalize

5.2.3 Implementácia pohľadu mesiaca

Zobrazenie mesiaca je implementované pomocou HTML tabuľky. Počet riadkov tabuľky je vypočítaný ako počet celých týždňov, do ktorých mesiac zasahuje. Tento počet môže byť podľa konkrétneho mesiaca 4 až 6. Každá udalosť je konvertovaná na jeden (jednodenné udalosti) alebo viac (viacdenné udalosti) prvkov. Udalosti v rámci jedného dňa sú zoradené podľa času začiatku. Dnešný deň je zobrazený s modrou farbou v pozadí. Výška riadku je dynamická a prispôsobí sa počtu udalostí v dňoch v danom týždni.

Mon	Tue	Wed	Thu	Fri	Sat	Sun
01 1 on 1 with Manager Ondro Manager Standup Sprint Planning Check-servers	02 1 on 1 with Manager Juro Manager Standup Check-servers	03 1 on 1 with Manager Samo Manager Standup Home-office Check-servers	04 Home-office 1 on 1 with Manager Michal Manager Standup Check-servers	05 Home-office Code Freeze Manager Standup Company Meeting Check-servers	06 Two-day Spa	07 Two-day Spa
08 1 on 1 with Manager Ondro Manager Standup Check-servers	09 1 on 1 with Manager Juro Manager Standup Check-servers	10 1 on 1 with Manager Samo Manager Standup Home-office Check-servers	11 Home-office 1 on 1 with Manager Michal Manager Standup Check-servers	12 Home-office Code Freeze Manager Standup Company Meeting Check-servers	13 Two-day Spa	14 Two-day Spa
15 1 on 1 with Manager Ondro Manager Standup Sprint Planning Check-servers	16 1 on 1 with Manager Juro Manager Standup Check-servers	17 1 on 1 with Manager Samo Manager Standup Home-office Check-servers	18 Home-office 1 on 1 with Manager Michal Manager Standup Check-servers	19 Home-office Code Freeze Manager Standup Company Meeting Check-servers	20 Two-day Spa	21 Two-day Spa
22 1 on 1 with Manager Ondro Manager Standup Check-servers	23 1 on 1 with Manager Juro Manager Standup Check-servers	24 1 on 1 with Manager Samo Manager Standup Home-office Check-servers	25 Home-office 1 on 1 with Manager Michal Manager Standup Check-servers	26 Home-office Code Freeze Manager Standup Company Meeting Check-servers	27 Two-day Spa	28 Two-day Spa
29 1 on 1 with Manager Ondro Manager Standup Sprint Planning Check-servers	30 1 on 1 with Manager Juro Manager Standup Check-servers	31 1 on 1 with Manager Samo Manager Standup Home-office Check-servers	01 Home-office	02 Home-office	03	04

Obr. 3: Pohľad mesiaca.

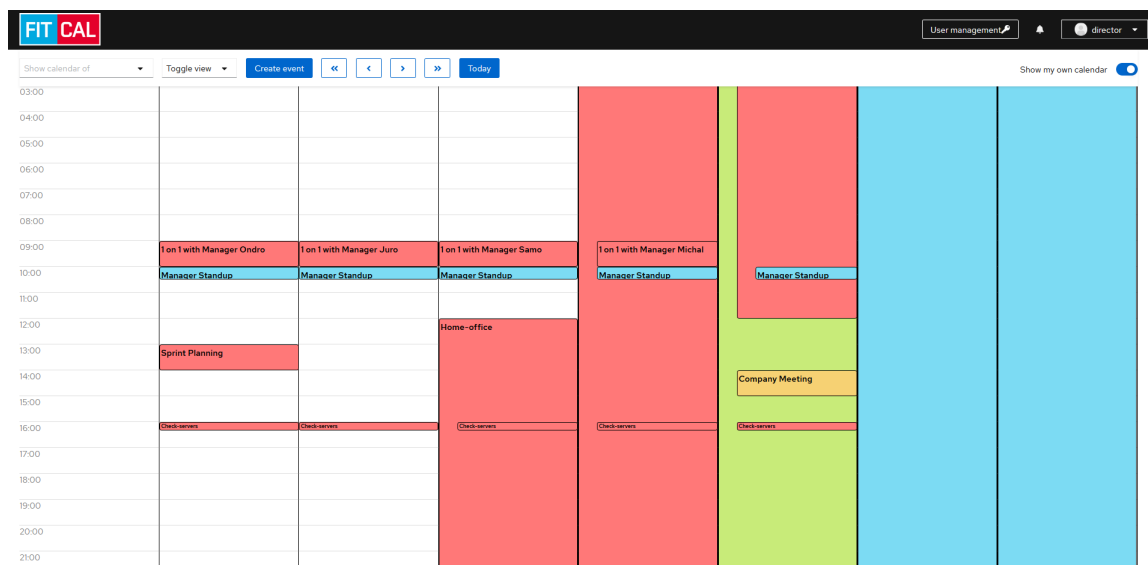
5.2.4 Implementácia pohľadu týždňa

Pohľad týždňa je implementovaný ako 8 stĺpcov zoradených vedľa seba. Prvý stĺpec obsahuje details daného týždňa a značky pre každú hodinu. Ďalších 7 reprezentuje každý deň v týždni. Zobrazovanie udalostí so sebou prináša dva hlavné problémy – ako zobraziť prelínajúce sa udalosti a ako zobraziť viacdenné udalosti.

Pri detekcii prelínajúcich sa udalostí sa kratšej z udalostí nastaví ľavá marža vyššie ako dlhšej udalosti, čo umožní, aby obe udalosti boli viditeľné aj keď majú rovnaký začiatok a koniec. Udalosť, nad ktorou je kurzor je stále zobrazená v popredí (je jej nastavený vysoký *z-index* pri udalosti *hover*).

Viacdenné udalosti sú na začiatku rozdelené na viacero jednodňových udalostí,

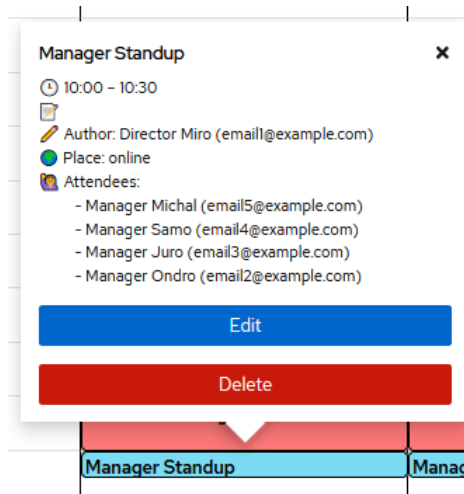
v prvom dni viacdennej udalosti je jej virtuálny koniec pre daný deň nastavený na 24:00; v posledný deň je jej virtuálny začiatok pre daný deň nastavený na 00:00; dni medzi nami majú nastavený virtuálny začiatok 00:00 a koniec 24.00.



Obr. 4: Pohľad týždňa.

5.2.5 Detail udalosti

Pri kliknutí na udalosť sa zobrazí detail udalosti obsahujúci metadáta udalosti. V prípade, že je prihlásený používateľ aj autorom tejto udalosti, dostane možnosť vymazať alebo upraviť túto udalosť pomocou modálneho okna.



Obr. 5: Detail udalosti zobrazený po kliknutí na udalosť.

5.2.6 Implementácia správy používateľov

Správa používateľov prebieha zo samostatnej cesty, ku ktorej majú prístup iba administrátori. Administrátor v tabuľke vidí zoznam používateľov spolu s ich metadátami. Každý používateľ (okrem samotného prihláseného používateľa) sa dá vymazať. Okrem používateľského mena sa každému používateľovi dajú zmeniť všetky jeho atribúty. Na vytvorenie tabuľky bolo nutné získať z backendu informácie o všetkých používateľoch, preto modálne okno na vyplnenie existujúcich polí pri editácii používateľa nepotrebuje zasielať ďalšiu HTTP žiadosť.

Používateľom s rolou sekretárka sa dá meniť zoznam priradených manažérov pomocou samostatného modálneho okna (obr. 7). Modálne okno poskytuje dva zoznamy, jeden obsahuje zoznam manažérov, ktorí priradení sekretárke nie sú a druhý zoznam tých, ktorí jej sú priradení. Používatelia sa pomocou tlačidiel dajú medzi zoznamami presúvať.

FIT CAL

User management

director

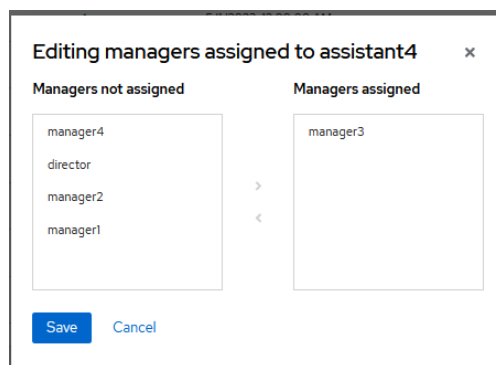
Go back

Add new user

Registered users

User name	Full name	Email address	Role	Can manage us...	Registered	
manager1	Manager Ondro	email2@example.com	Manager	✗	5/1/2023, 12:00:00 AM	<div>EditRemove</div>
assistant5	Assistant Rozalia	email10@example.com	Assistant	✗	5/1/2023, 12:00:00 AM	<div>EditEdit assigned managersRemove</div>
assistant2	Assistant Nadezda	email7@example.com	Assistant	✗	5/1/2023, 12:00:00 AM	<div>EditEdit assigned managersRemove</div>
director	Director Miro	email@example.com	Director	✓	5/1/2023, 12:00:00 AM	<div>Edit</div>
manager3	Manager Samo	email4@example.com	Manager	✗	5/1/2023, 12:00:00 AM	<div>EditRemove</div>
assistant3	Assistant Julia	email8@example.com	Assistant	✗	5/1/2023, 12:00:00 AM	<div>EditEdit assigned managersRemove</div>
assistant1	Assistant Jozefina	email6@example.com	Assistant	✗	5/1/2023, 12:00:00 AM	<div>EditEdit assigned managersRemove</div>
assistant4	Assistant Petronela	email9@example.com	Assistant	✗	5/1/2023, 12:00:00 AM	<div>EditEdit assigned managersRemove</div>
manager2	Manager Juro	email3@example.com	Manager	✗	5/1/2023, 12:00:00 AM	<div>EditRemove</div>
manager4	Manager Michal	email5@example.com	Manager	✗	5/1/2023, 12:00:00 AM	<div>EditRemove</div>

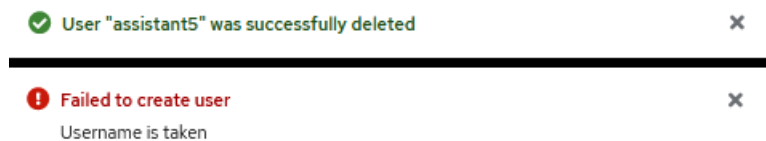
Obr. 6: Stránka správy používateľov.



Obr. 7: Modálne okno na správu priradených manažérov.

5.2.7 Notifikácie

Úspešné vykonané akcie, ako vytváranie používateľa, úprava udalosti atp., zaistia zobrazenie plávajúceho upozornenia (*toast notification*) pomocou funkcie `addToastNotification()`, ktorá je propagovaná pomocou `props` do všetkých komponentov. Neúspešne vykonané akcie taktiež zobrazia notifikáciu v červenej farebnej schéme spolu s dôvodom neúspechu získaného z HTTP odpovede.



Obr. 8: Plávajúce notifikácie indikujúce úspech alebo neúspech vykonanej akcie.

5.3 Pomocné súbory

Medzi pomocné súbory patria:

- `CalendarHelper.js` – implementuje mnoho funkcií na manipuláciu s dátumami a časmi.
- `Constants.js` – izoluje konštanty do samostatného súboru.
- `Hooks.js` – hooky používané na HTTP komunikáciu (popísané v sekcii 3.1).
- `Utils.js` – rôzne pomocné funkcie, ktoré nepatria do žiadnej inej kategórie.
- `Validators.js` – funkcie na validáciu používateľských vstupov rôzneho typu.

6 Testovanie

End-to-end testovanie je implementované pomocou knižnice *Cypress*. Na vytváranie testov bolo použité prehliadačové rozšírenie *Cypress Scenario Recorder*³, ktoré umožňovalo nahrávať kroky používateľa na stránke a preložiť ich do príkazov v *Cypress*. Testy vyžadujú spustený backend, bez žiadnych zmien v databáze.

Zoznam jednotlivých testovaných prípadov:

- Aplikácia sa načíta,
- používateľ sa môže prihlásiť,
- používateľ môže vytvoriť sa vymazať udalosť,
- administrátor môže vytvoriť a vymazať nového používateľa,
- sekretárka si môže pozrieť kalendár priradeného manažéra,
- vytvorenie novej udalosti zašle všetkým účastníkom notifikáciu.

³<https://chrome.google.com/webstore/detail/cypress-scenario-recorder/fmpgoobcionmfneadjapdabmjfkmfkeb>