Brno University of technology
Faculty of Information Technology

Technical report

Authors:
Kozhevnikov Dmitrii (xkozhe00)
Pastushenko Vladislav (xpastu04)
Tomason Viktoryia (xtomas34)

Brno
05.12.2021

Obsah

Project objective

The goal of our project was to create a website for a transport company (as an analogue of DPMB) that would be user-friendly. Our project was based on a website *dpmb.cz* that needed changes. In the course of studying the content of the site, our team realized that using this site can be inconvenient for both an ordinary user and the administrator who serves it.

Therefore, we have set the following tasks:

- Development of a new design for a website
- Creating a simple and intuitive user interface
- Creating a convenient panel for the administration of this site
- Writing the backend and its connection with the frontend
- Adapting information for the user
- Testing

Design, tools and implementation

K zadaní projektu byl dan virtuální stroj s referenčním klientem a serverem. Bylo nutné vytvořit spojení mezi klientem a serverem a poté pomocí Wiresharku zachytit tuto komunikaci a analyzovat ji. Výsledek zachycené komunikace je v souboru ***isa.pcap***.

Byl nám chycen neznámý protokol TCP, který obsahoval pakety, které měly nešifrované data jak při odesílání na server, tak při přijímání odpovědi z neho.

Backend

In this project, the backend was executed on ExpressJS to the MVC pattern.

In the root folder *api* there is a file **.env** in which there is configuration data, such as passwords, URL, etc.

The main implementation files are located in the folder **server** The folder *server* contains a file **app.js** with the configuration settings for ExpressJS. The file **mysql.js** contains a class *MySQL* that is used to connect to the database. It contains a constructor in which we use the *bookshelf* to connect to the database and interact with it, and we also use *knex*.

The file routes.js contains the main routes of our backend available for external connection. These routes are divided into groups according to the models that are used in the database.

There are also folders *controllers, model, utils*.

The folder *controller* contains all controllers for each database entity. The controller is called from the routes and then processes the received data and sends a request to the model and sends a request to the model, which are located in the folder *model*.

In the *model* folder there is a file **baseModel.js** in which represented the parent class *BaseModel*. This class has basic functions that are often used by other models. The other files contain the implementation of independent modules in which queries to the database are generated.

There is also a folder *public* containing all public pages.

Frontend

In this project, the backend was executed on ReactJS with using the Sass preprocessor.

In the root folder *client* contains the main file **Api.js** a file **style.sass** that contains the main styles for the main page. Also here are the folders *components*, *pages*, *public*, *store*.

The folder *components* contains the main components that the user sees. *Basement* contains footer, *Dashboard* - Admin panel, *Navigator* - Navigator, *Searcher* - route search form and *Userboard* which contains user interface. These folders contain subfolders that contain specific elements and their corresponding styles.

The folder *pages* contains main pages, a folder *styles* with styles and a folder *user* that contains a token for user verification using **Cookies** The file **_app.js** contains the main function *App* that builds the pages.

The folder *public* contains the main files for design solutions.

The folder *store* contains subfolders for elements containing files that link the frontend to the backend.

During the implementation, we used graphic libraries *Ant Design* and *MUI*.

Pages

index.js - the main page
dashboard.js - administrative panel, available only for users with the appropriate roles
login.js - user registration form available only for administration
userboard.js - a page with a user interface containing his current trips and settings
server-error.js - the error page, if the backend does not respond
contact-us.js page for contact with the site administrator
contact.js – main information about company
faq.js - the page with the most frequent questions and answers to them
documents.js - page with links to documents provided by the company
lodni-doprava.js - page with information and schedule of the ship
transport.js - the page with the scheme of urban transport
vacancies.js - current job offers in DPMB
pricelist.js - page with price information for advertisers
lost-and-found.js - information about lost and found items

Folders with components

Basement - footer
Dashboard - components for the administrative panel
Faqs - components to display FAQs
Navigator - navigator
News – carousel news
Searcher - route search form
Trams – animated tram cars
TransportTables - components for displaying tables with transport data

Chips

- Use of cookies
- Footer and navigator for moving the user around the site
- Special design
- Using video for the background and animations for various elements of the site
- New panel for site administration

How to use it?

User

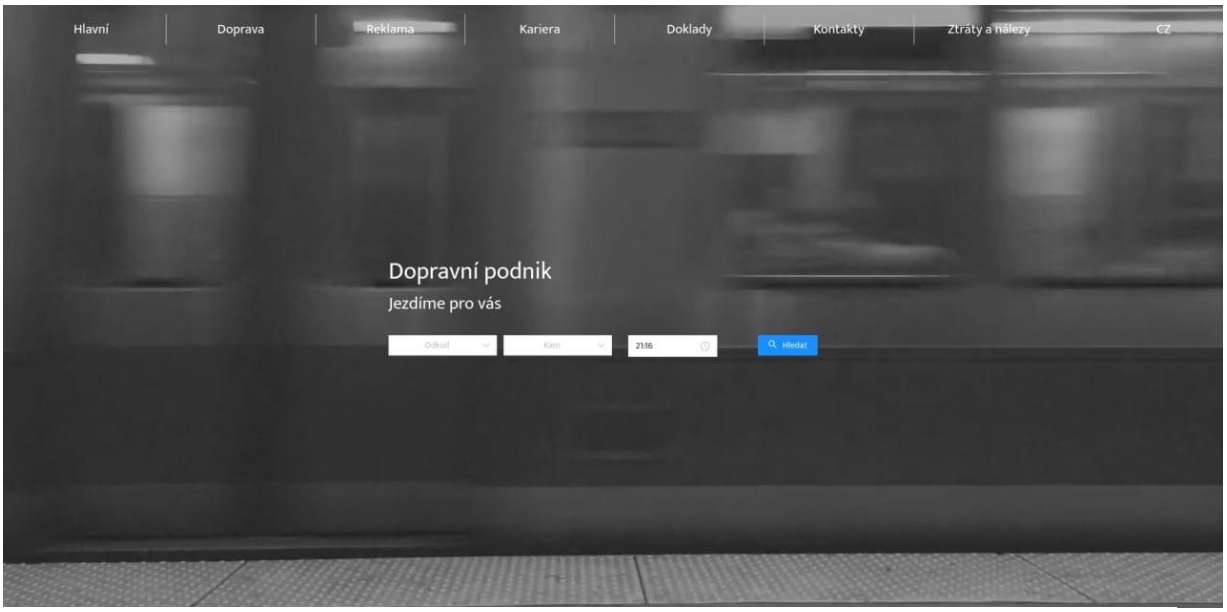
In our project, pages are implemented for ordinary users and for administrators.

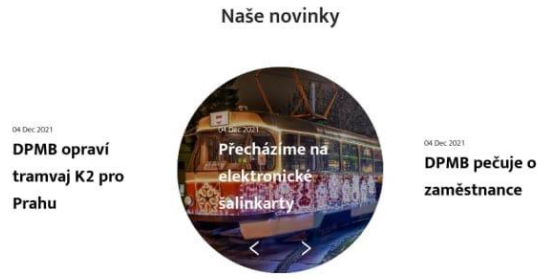
An ordinary user has the ability to move freely around the site and does not have access to the admin panel. The user can use the navigator and footer to navigate through the pages. There are also interactive buttons on the pages that also provide the user with some information. On the main page there is a route search form in which the user can set his route and departure time.

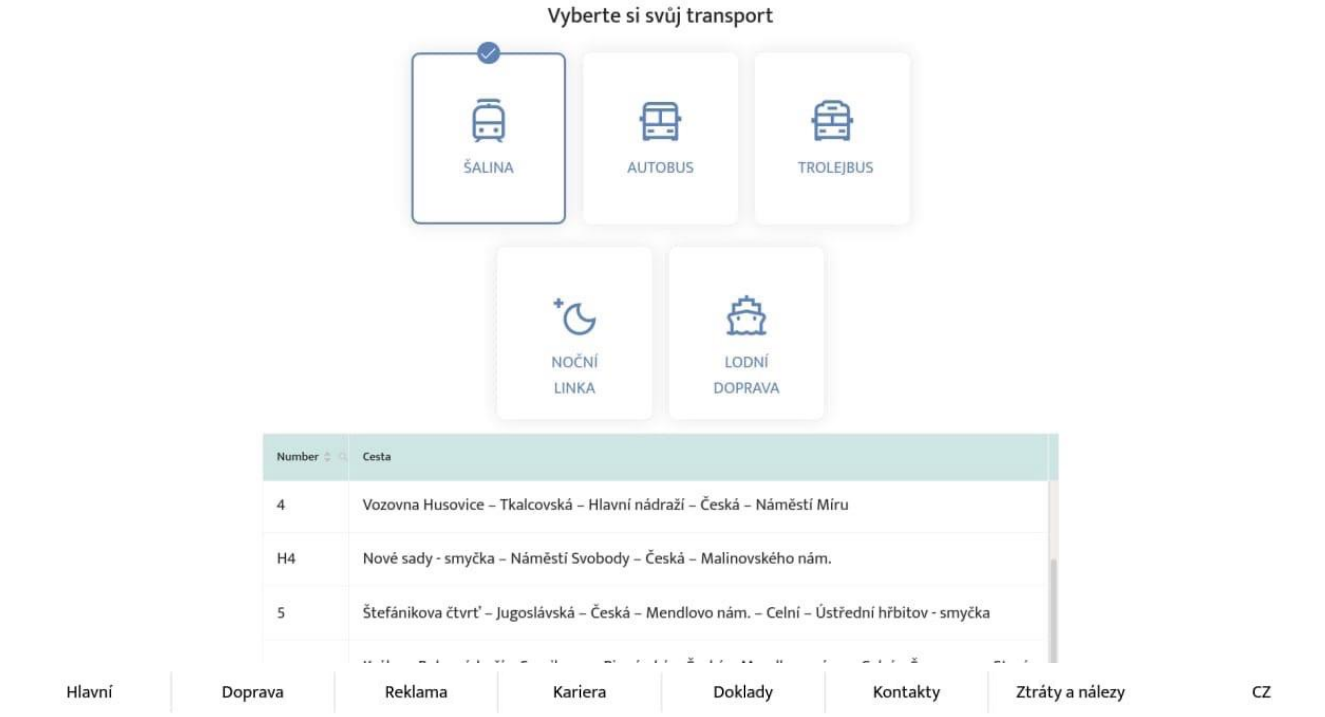
Administrator and personnel

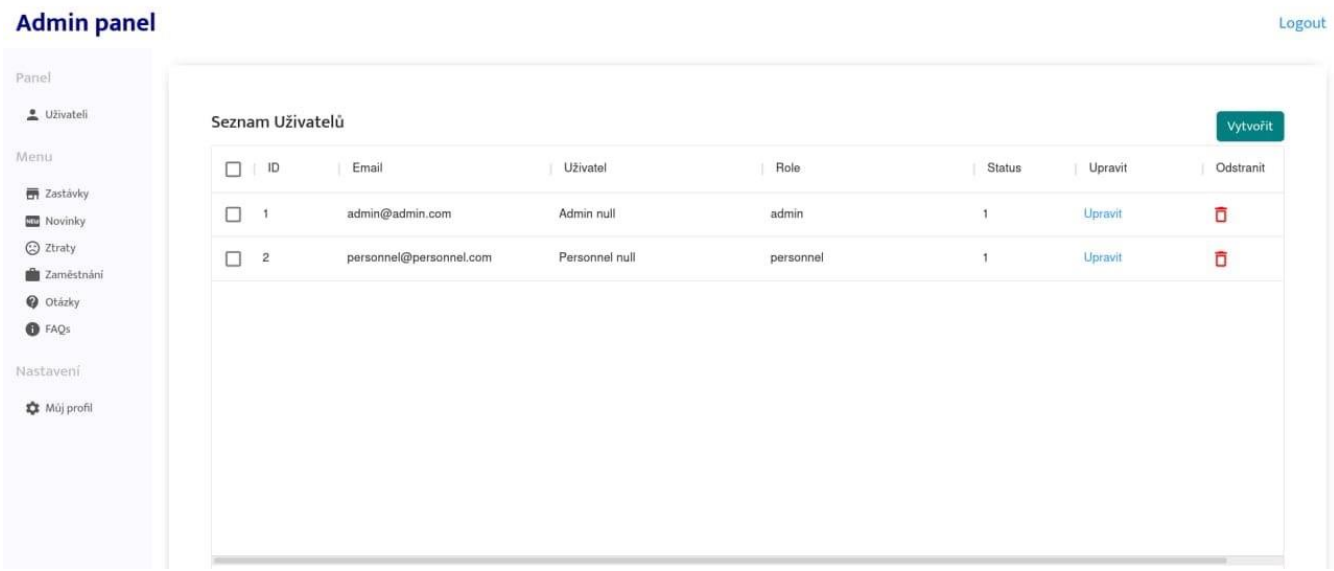
A special panel has been developed for site employees to interact with the data that will be displayed on the site. Only a user who is registered in the system can get to this page. Registration of the user is carried out by the site administrator, who is already specified in the database (for testing **login**: admin@admin.com, **password**: Admin123). You can get to the administrative panel by clicking the button *Admin* in the footer. On the administrative panel, the user sees on the left the main items that interact with the database: users, stops, lost items, the most frequent questions, vacancies, news and a list of questions that users send. The administrator can add new items and delete old ones, as well as edit users.

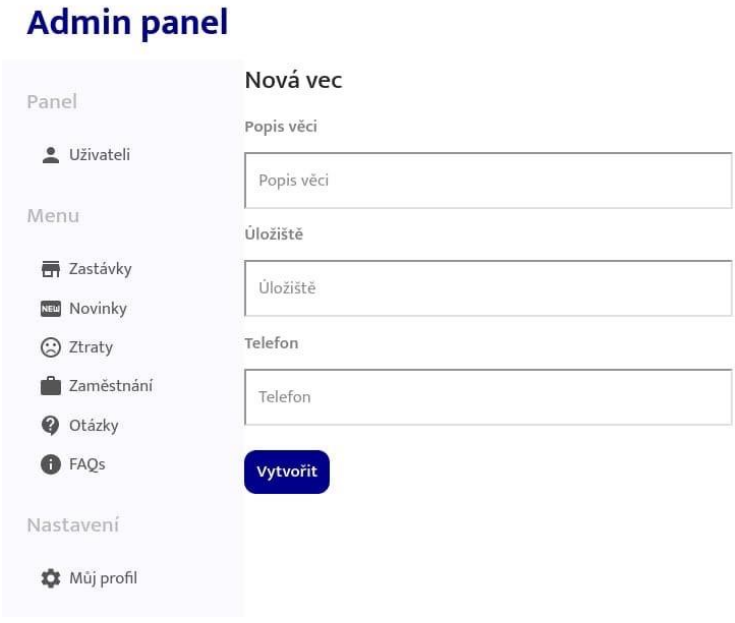
Accomplished work






Vyberte si svůj transport
ŠALINA
AUTOBUS
TROLEJBUS
NOČNÍ LINKA
LODNÍ DOPRAVA
Number
Cesta
4
Vozovna Husovice – Tkalcovská – Hlavní nádraží – Česká – Náměstí Míru
H4
Nové sady - smyčka – Náměstí Svobody – Česká – Malinovského nám.
5
Štefánikova čtvrť – Jugoslávská – Česká – Mendlovo nám. – Celní – Ústřední hřbitov - smyčka
Hlavní
Doprava
Reklama
Kariera
Doklady
Kontakty
Ztráty a nálezy
CZ


Admin panel
Logout
Panel
Uživateli
Menu
Zastávky
Novinky
Ztraty
Zaměstnání
Otázky
FAQs
Nastavení
Můj profil
Seznam Uživatelů
Vytvořit
ID
Email
Uživatel
Role
Status
Upravit
Odstranit
1
admin@admin.com
Admin null
admin
1
Upravit
2
personnel@personnel.com
Personnel null
personnel
1
Upravit


Admin panel
Panel
Uživateli
Menu
Zastávky
Novinky
Ztraty
Zaměstnání
Otázky
FAQs
Nastavení
Můj profil
Nová vec
Popis věci
Popis věci
Úložiště
Úložiště
Telefon
Telefon
Vytvořit

Instalation

- From root folder of project
- (cd client; sudo npm install &); (cd api; sudo npm install &);
- (cd client; sudo npm run start &); (cd api; sudo npm run start &);
- FE: localhost, BE: localhost:8000

Literature

Materials from lectures were used, as well as documentation for the libraries that we used