



FIZZY DRINKS

Michał Mosiński



Przedmiot	Programowanie aplikacji internetowych				
Nazwa zadania	PROJEKT				
Nazwisko imię	Mosiński Michał				
Nr albumu	148590	Grupa	INF:7 D1_S5	Rodzaj studiów (D,Z,W)	dzienne
Rok akademicki	3			Semestr	5
Data wykonania	22 stycznia, 2023 roku				

Tematyką projektu przedstawienie listy napojów, które są przechowywane w bazie danych. Do serwera bazy danych mysql skorzystałem z narzędzia „xampp”. Projekt jest napisany w językach : PHP, HTML, CSS. Do stworzenia niżej opisanej aplikacji skorzystałem z frameworku Laravel. Dodatkowo do systemu logowania skorzystałem z Laravel Breeze. Skorzystałem także z pomocy narzędzia Postman do pobierania zawartości w formacie json przez API. Aby projekt zadziałał musimy mieć zainstalowanego Composera(<https://getcomposer.org/>), środowisko node.js(<https://nodejs.org/en/>), oraz npm(można go zainstalować dzięki komendzie ‘npm install w konsoli).

Proces uruchomienia aplikacji

W tym celu należy mieć uruchomianego xamp-a oraz zawartość z aplikacją przeniesioną do odpowiedniego folderu(na którym xamp działa). Następnie aplikację uruchamiamy przez wpisanie niżej podanych komend do konsoli, jeżeli xxamp jest uruchomiony baza danych mysql powinna się automatycznie stworzyć.

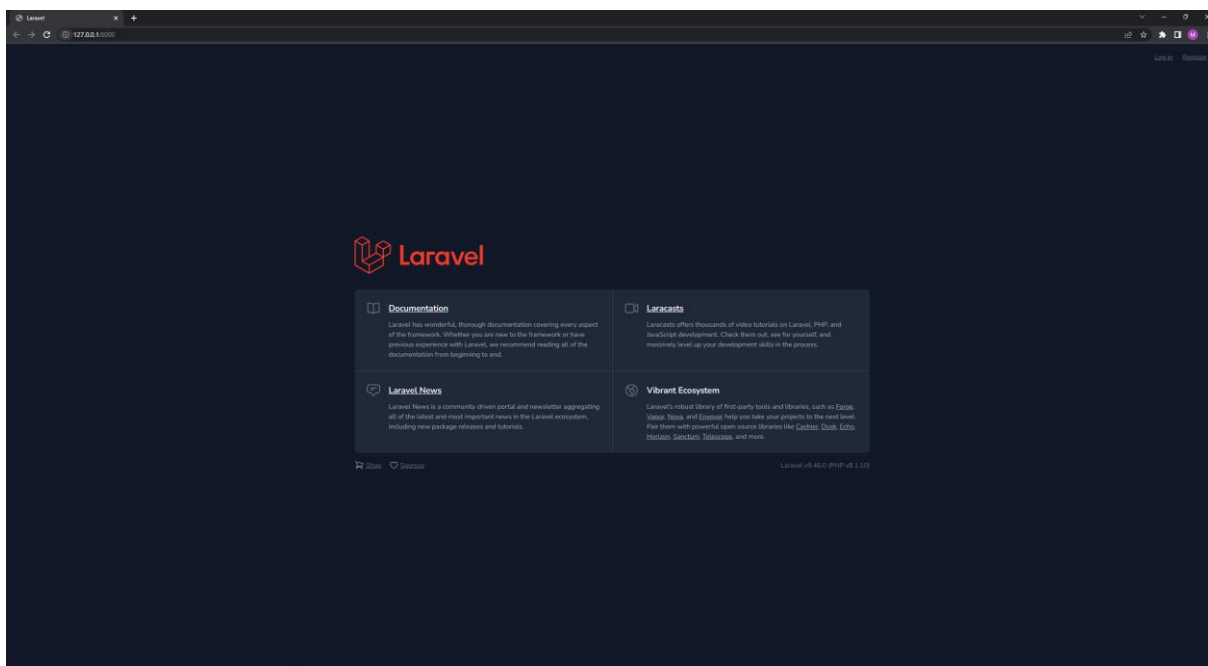
Komendy wpisywane do konsoli :

1. php artisan db:seed
2. php artisan migrate
3. npm run dev
4. php artisan serve

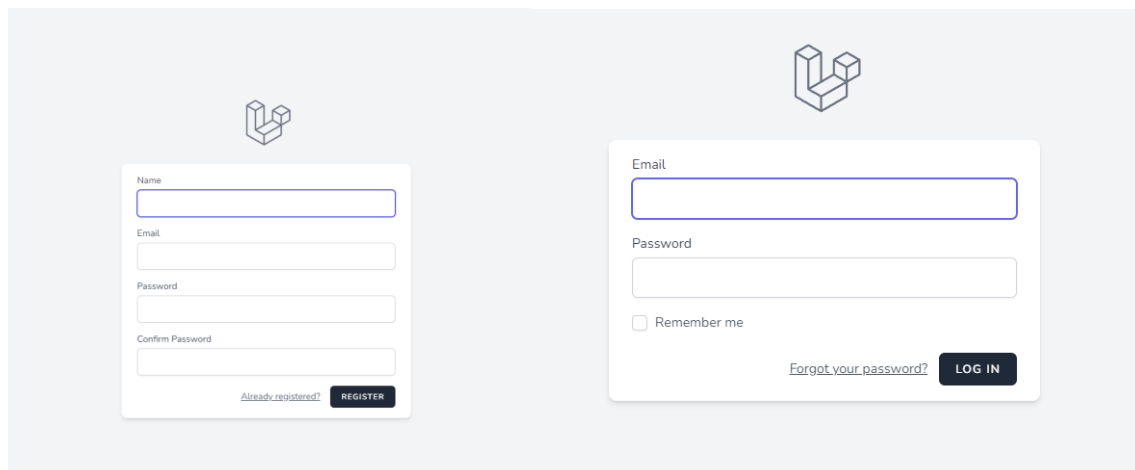
Do uruchomienia aplikacji po raz drugi wystarczy już tylko komendy z punktów : 3,4

Strona startowa aplikacji

Po uruchomieniu aplikacji w terminalu ukazuje się link do aplikacji, gdy w niego wejdziemy zobaczymy stronę startową :



Jest strona główna Laravela, z dodanymi opcjami logowania oraz rejestracji użytkowników. Wyglądają one w następujący sposób :



The image shows two side-by-side forms on a light gray background. Both forms feature a logo consisting of three stacked cubes. The left form is for registration, with fields for Name, Email, Password, and Confirm Password, followed by a 'REGISTER' button and a link for 'Already registered?'. The right form is for login, with fields for Email and Password, a 'Remember me' checkbox, a 'Forgot your password?' link, and a 'LOG IN' button.

Registration Form:

- Name:
- Email:
- Password:
- Confirm Password:
- [Already registered?](#) **REGISTER**

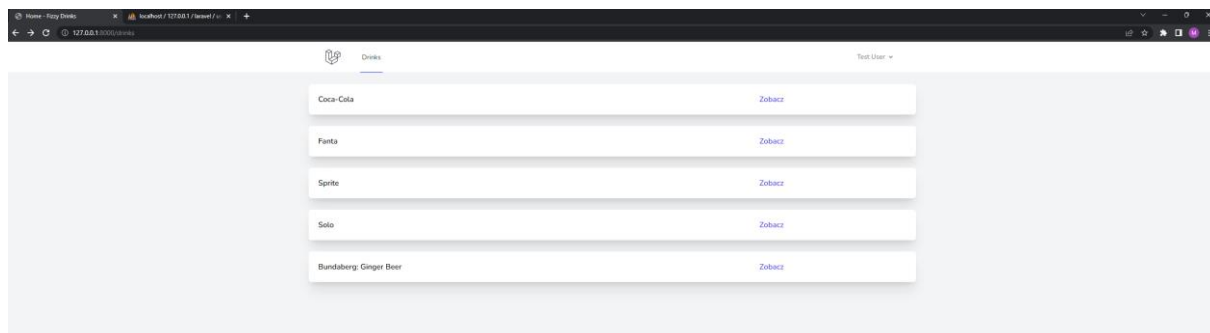
Login Form:

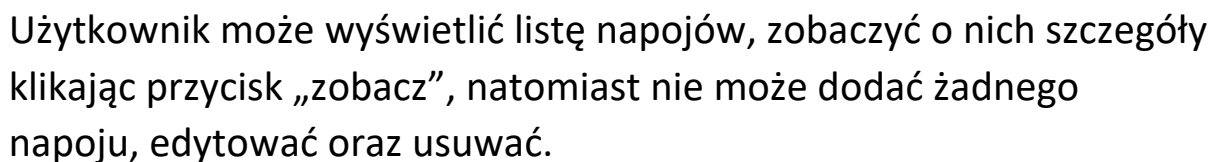
- Email:
- Password:
- ☐ Remember me
- [Forgot your password?](#) **LOG IN**

W osobnym pliku „login.txt” wypisałem dane logowanie do 2 kont, administratora oraz użytkownika

Strona główna aplikacji

Strona główna aplikacji wygląda w następujący sposób :





Fast Data

localhost / 127.0.0.1 / http://localhost:3000/

127.0.0.1:3000/

Test User

Profile

API Token: 3j49avLVrg3CnSUS63zcom1EG90WQ4pntLAYIP6

Profile Information

Update your account's profile information and email address.

Name

Test User

Email

user@example.com

Save

Update Password

Ensure your account is using a long, random password to stay secure.

Current Password

New Password

Confirm Password

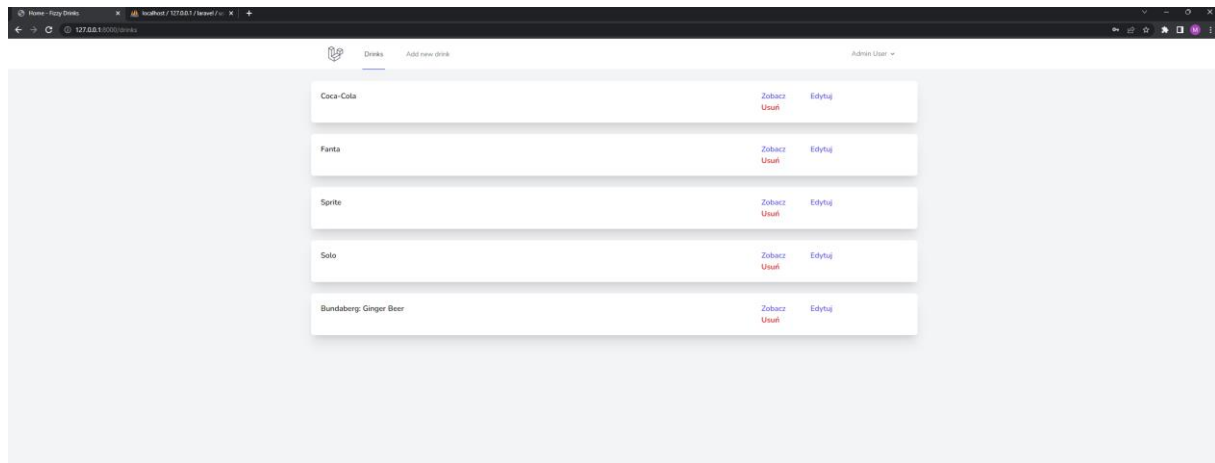
Save

Delete Account

Once your account is deleted, all of its resources and data will be permanently deleted. Before deleting your account, please download any data or information that you wish to retain.

Po zalogowaniu na konto administratora ukazać nam się dodatkowe funkcje na stronie głównej :

Wygląd strony na koncie administratora :

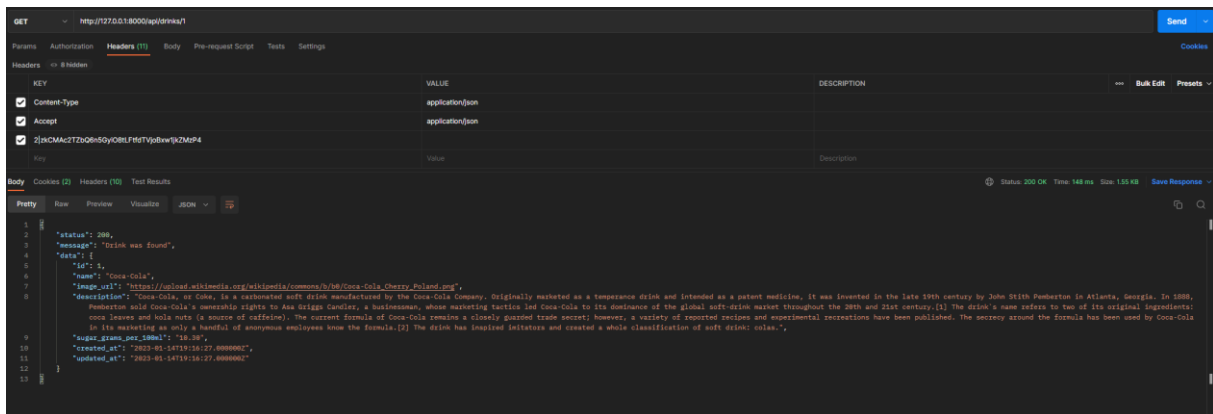


Strona/formularz dodania napoju :

A screenshot of the 'Adding new drink' form in the same admin interface. The browser's address bar shows '127.0.0.1:5173'. The page has a header with a logo, 'Drinks', and a link 'Add new drink'. On the right, it says 'Admin User'. The main content area is titled 'Adding new drink' and contains four input fields: 'Name', 'Description', 'Image URL', and 'Sugar grams per 100ml'. A 'Save' button is located at the bottom right of the form.

Na następnej stronie przedstawię kilka zrzutów ekranu z Postmana, który posłużył do pobrania przez API zawartości w formacie json.

API wyświetlenia wszystkich napojów :



API dodania napoju :

The screenshot displays the Postman interface for a POST request to the endpoint `http://127.0.0.1:8000/api/drinks`. The request is configured with the following headers:

KEY	VALUE	DESCRIPTION
Authorization	<code>Bearer 2j5xCMaG2TZ2Q6n5GyQ8LFR6TVyblwWjKZMP4</code>	
Cookie	<code>XSRF-TOKEN=eyJpdll8VWV3Z2Z0c2MmNmY0E9PSlnZnhiNlpl</code>	
Postman-Token	<code><calculated when request is sent></code>	
Content-Type	<code>application/json</code>	
Content-Length	<code><calculated when request is sent></code>	
Host	<code><calculated when request is sent></code>	
User-Agent	<code>PostmanRuntime/7.30.0</code>	
Accept	<code>*/*</code>	
Accept-Encoding	<code>gzip, deflate, br</code>	
Connection	<code>keep-alive</code>	

The response is a JSON object with the following structure:

```
1 {
2   "status": 201,
3   "message": "Created new drink",
4   "data": {
5     "name": "cola",
6     "description": "cola",
7     "image_url": "cola.jpg",
8     "super_drink_per_100ml": 20,
9     "updated_at": "2023-01-27T18:11:13.600000Z",
10    "created_at": "2023-01-27T18:11:13.600000Z",
11    "id": 58
12  }
13 }
```

The response status is 200 OK, with a time of 140 ms and a size of 536 B.