2023.01.29 – Cats\_MVC\_model\_documentation

docker elindítás

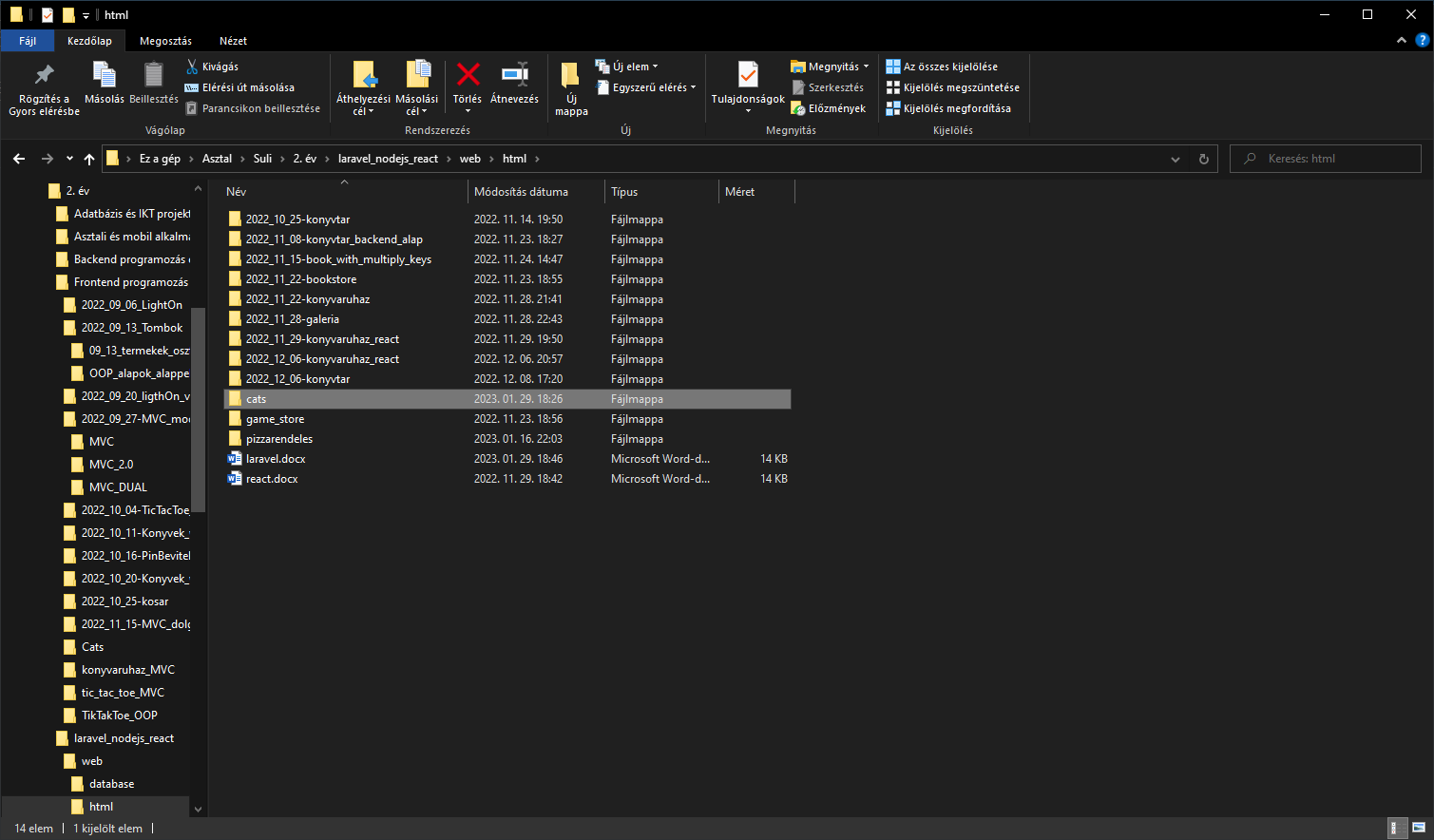
vs code terminal használata

docker exec -it (egyéni docker útvonal) 58537be3e7f7 /bin/bash

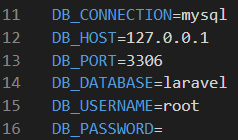
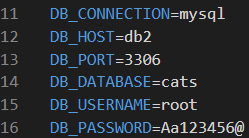
composer create-project laravel/laravel cats

miután kész a projekt

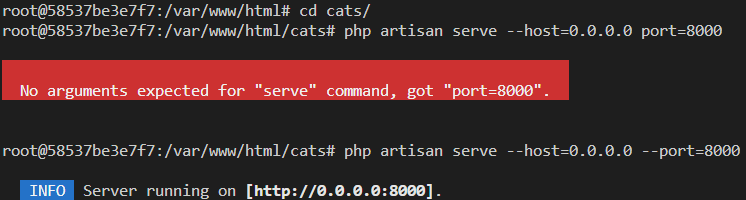
az adott fájlba hozzuk létre a projektet



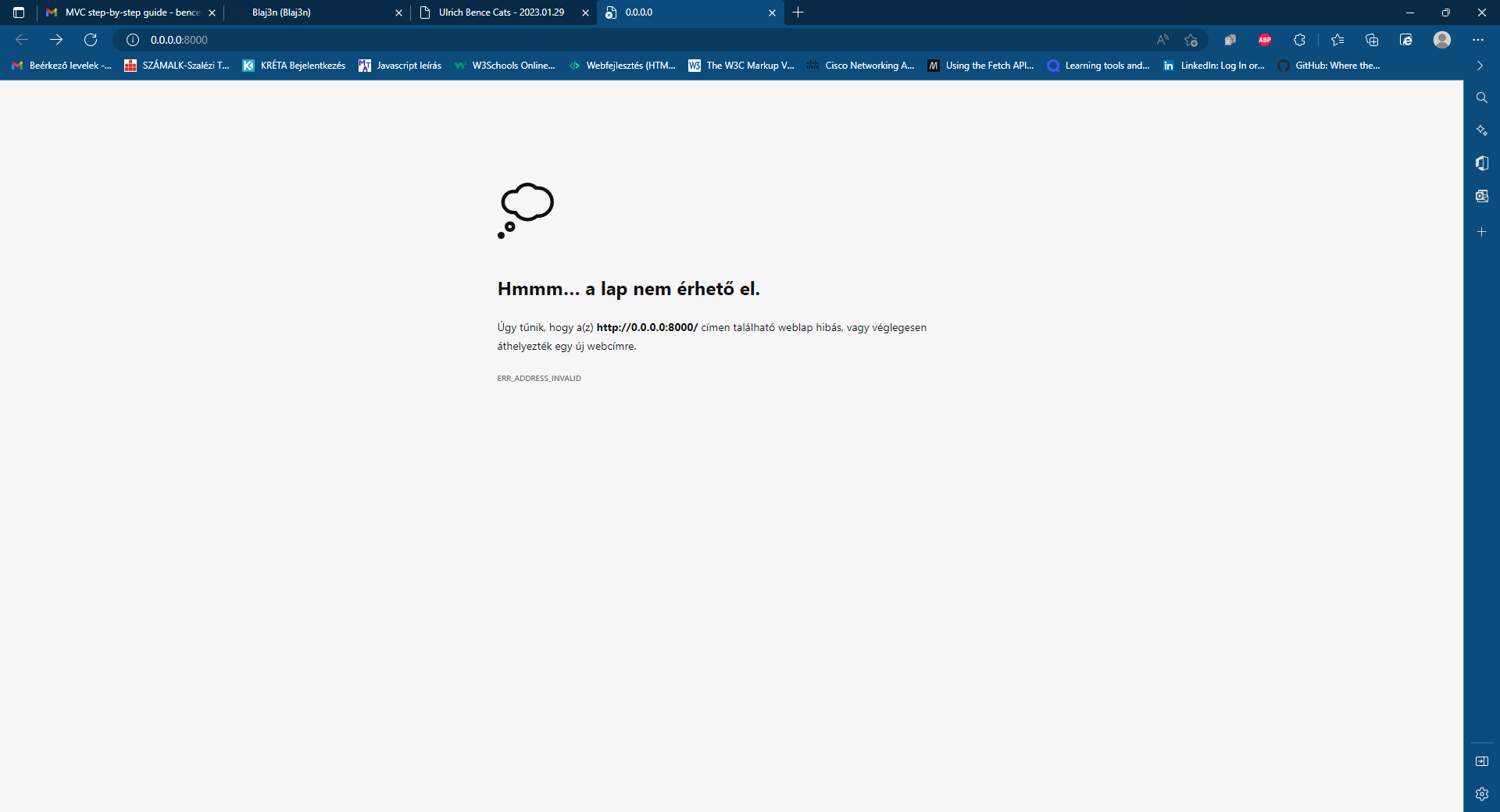
a .env fájlban írjuk át az

 -ra

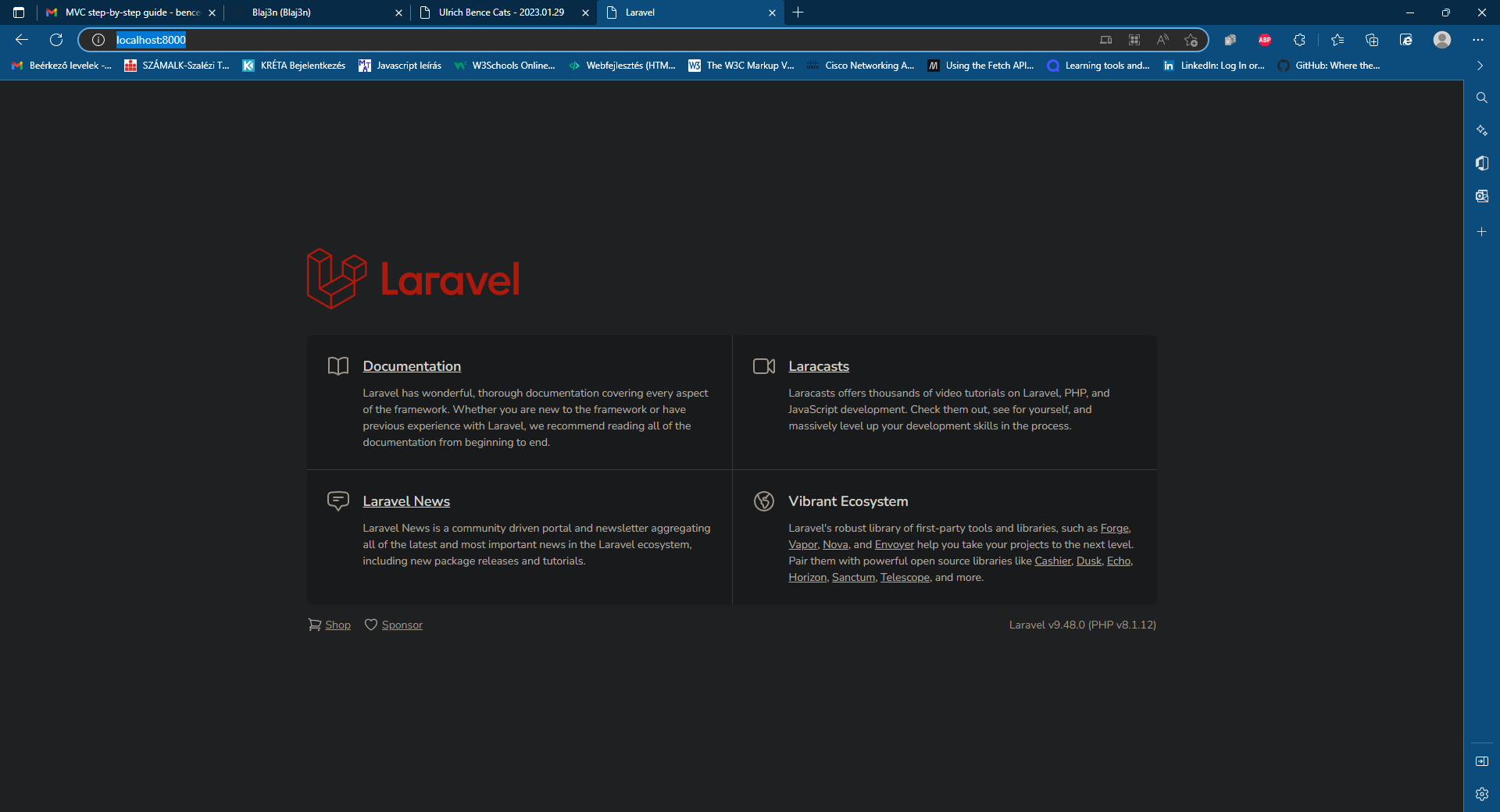
miután a terminalban beléptünk a mappánkba



a fent látható parancsokat beírjuk (helyesen) és a linkre ctrl+bal klikk rákattintunk



[0.0.0.0](http://0.0.0.0:8000/):8000 🡪 ezt a linket [localhost:8000-re](http://localhost:8000/) írjuk át



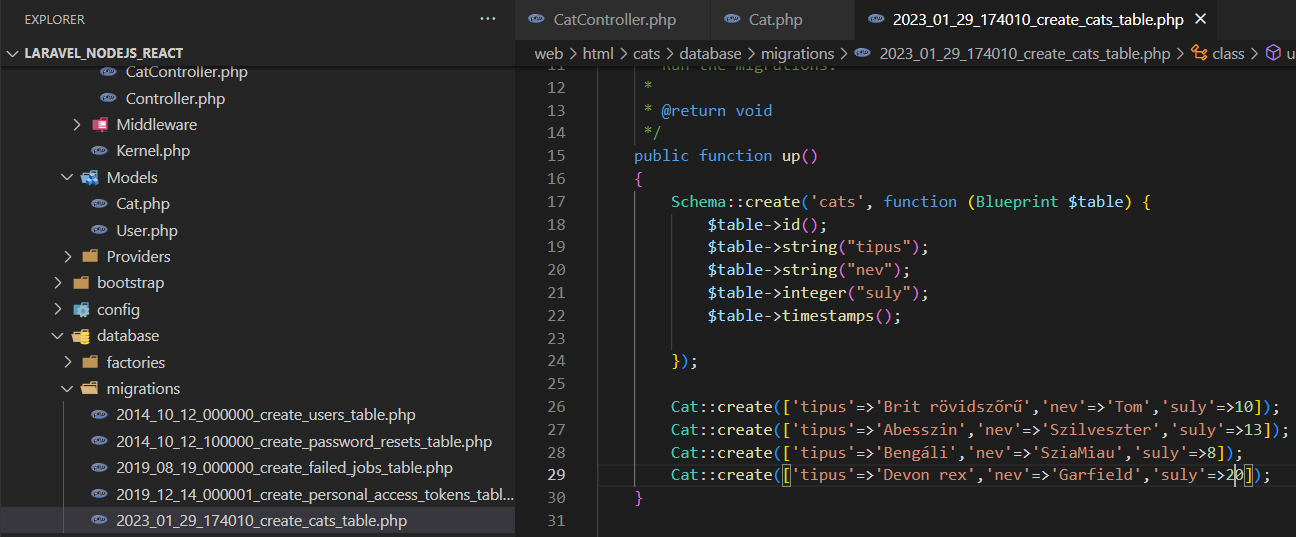
és ezt az oldalt kell kapnunk

majd a vs code-ban a terminálon létrehozunk egy modelt a következő parancssal

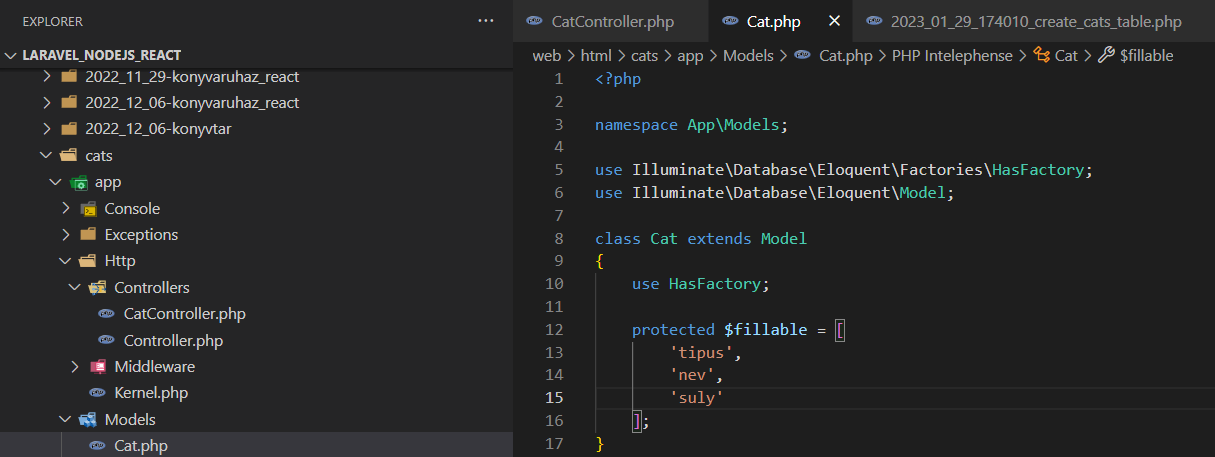
php artisan make:model Cat -m 🡪 Model mappában létrejön a Cat.php fájl

php artisan make:controller CatController 🡪 Controller mappában létrejön a CatController.php fájl

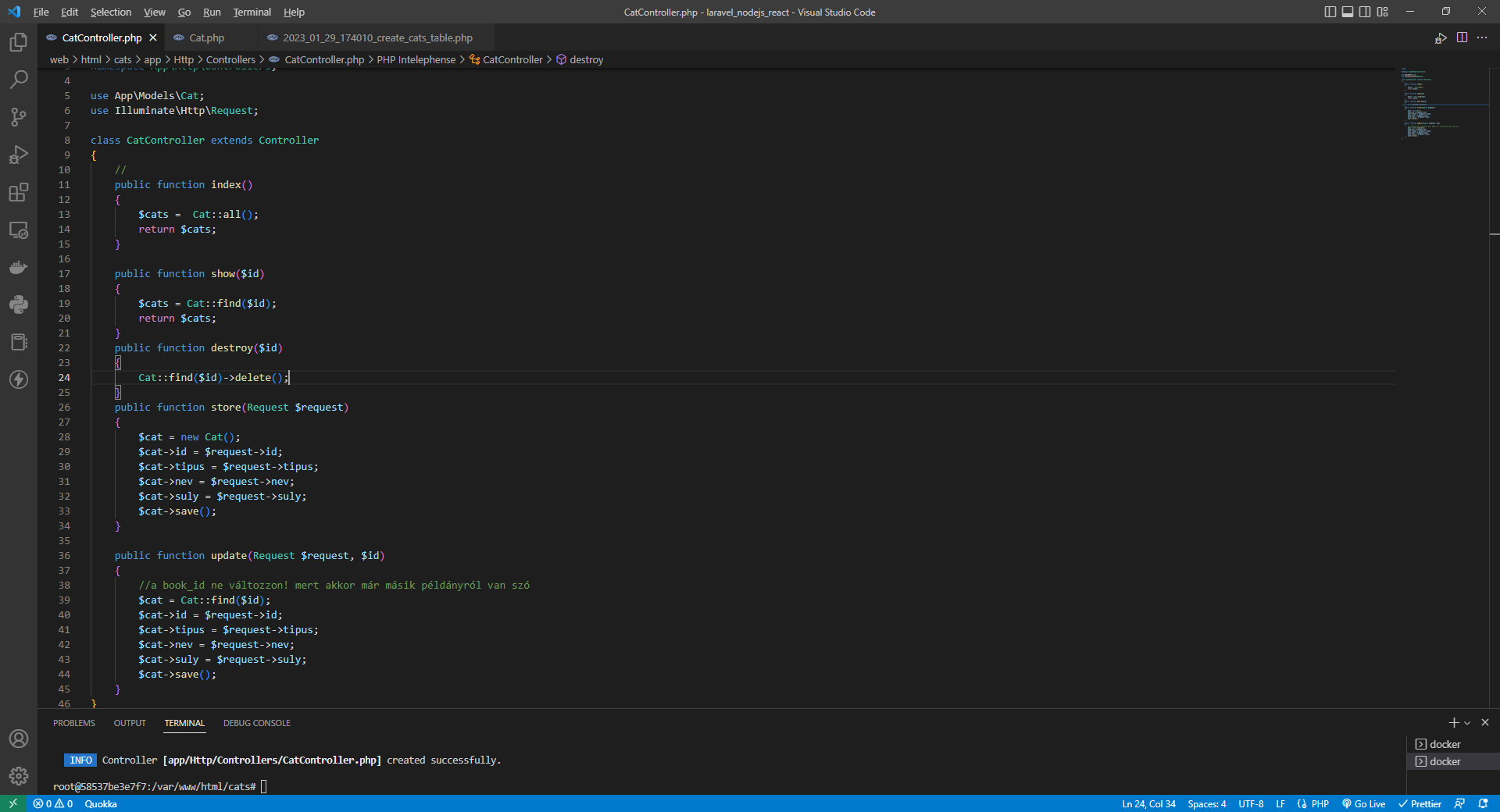
Majd a database/migrations mappában létrehozzuk a tábla oszlopokat, illetve feltöltjük azt adatokkal



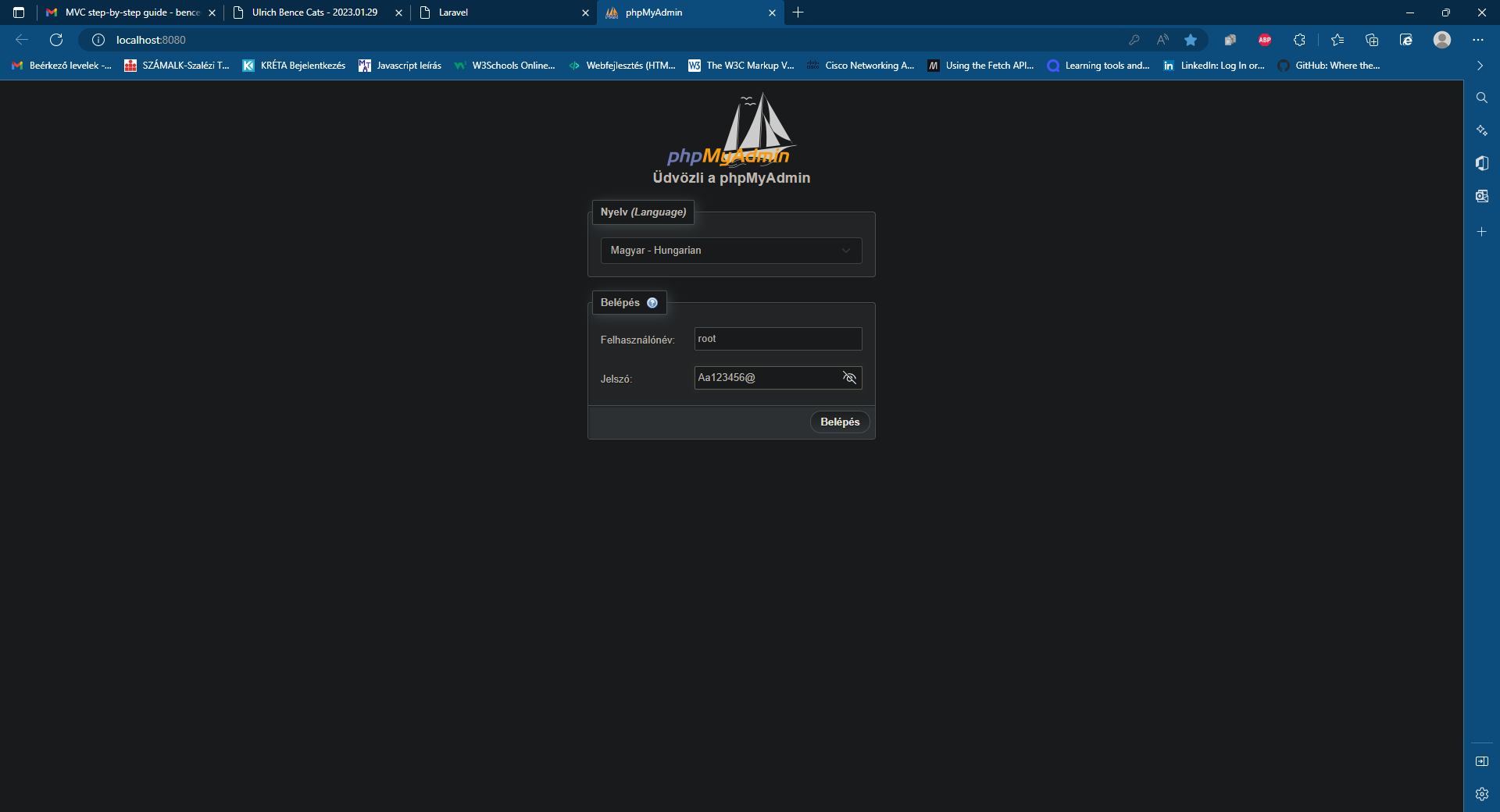
Az app/Models/Cat.php fájl



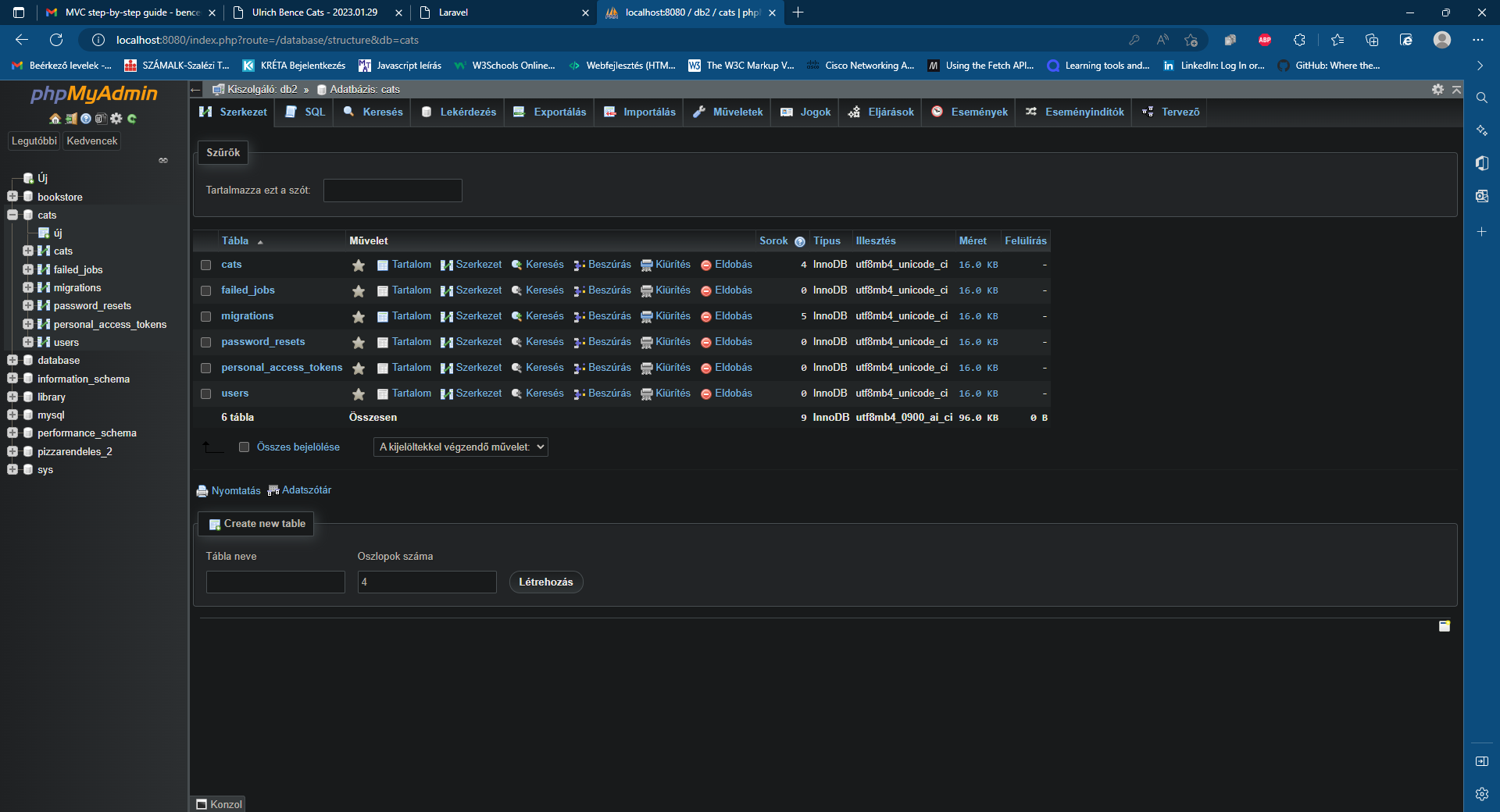
app/http/Controllers/CatController.php



miután ezekkel megvagyunk a terminalba beírjuk, hogy php artisan migrate és ha minden sikerül akkor

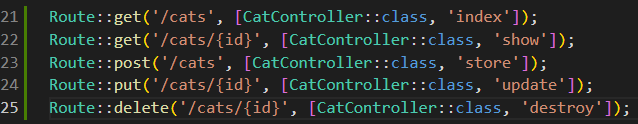


Ezen a felületen belépve

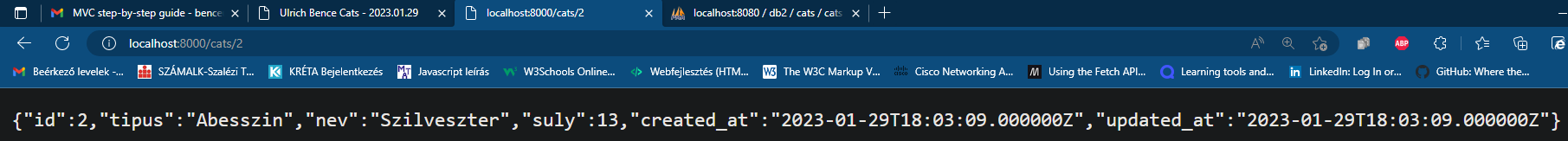


Ezt láthatjuk

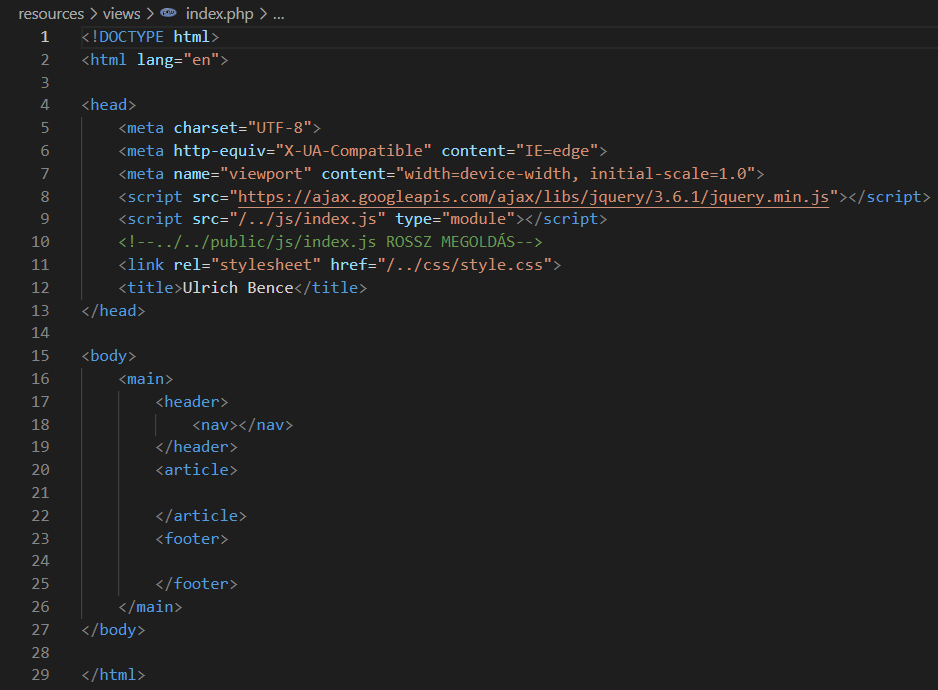
app/routes/web.php



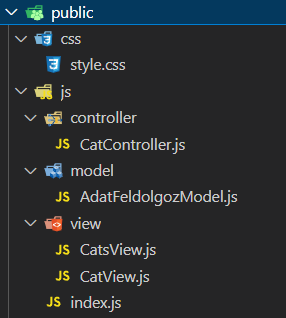
megírtuk az 5 api végpontot, illetve azokat le is teszteltük



app/resources/views mappa, és azon belül létrehozunk egy index.php-t

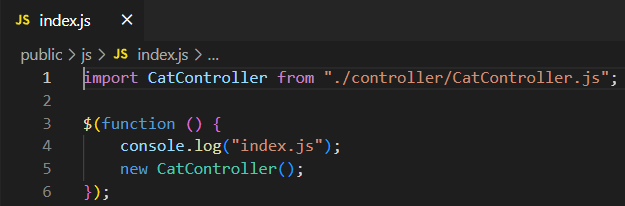


Ezt követően létrehoztuk a public mappában a fájl struktúrát

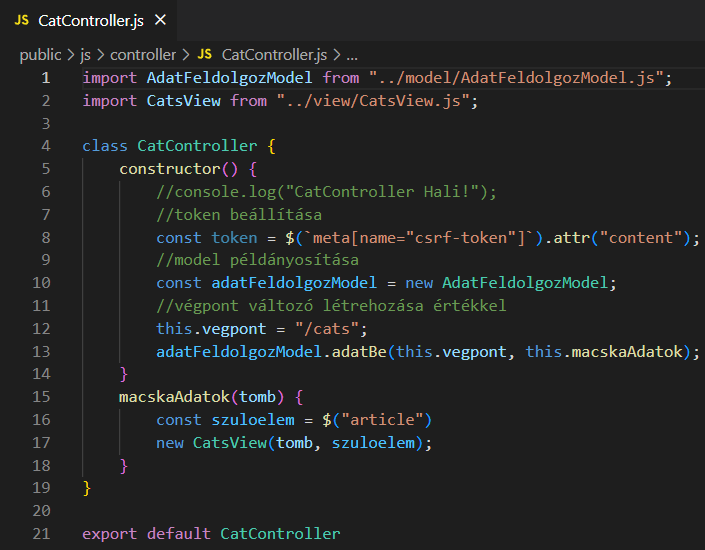


Utána a következő fájlokat hoztuk létre:

1. index.js



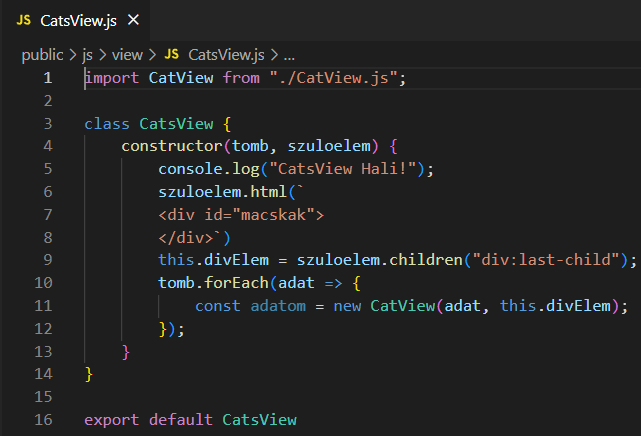
1. CatController.js



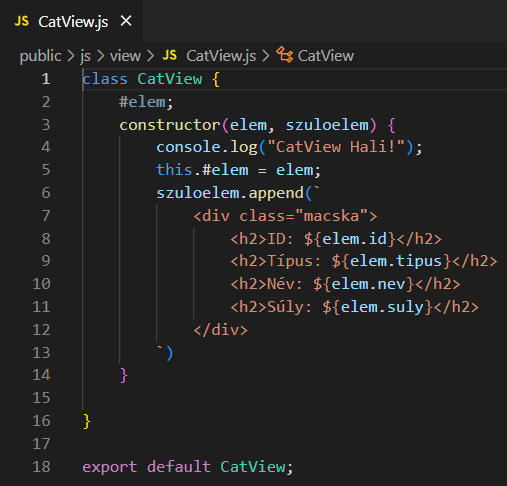
1. AdatFeldolgozModel.js



1. CatView.js



1. CatView.js



1. style.css

