2023.04.28.

Megyeri Márk Máté

TreCorte

Fejlesztői dokumentáció

TRECORTE

Megyeri Márk Máté

**Cél**

Egy étteremláncnak a nyilvántartás készítése, mely biztosítja az étterem és a vendégek számára a zökkenőmentes adminisztrációs munkákat.

A backend-t Laravel php keretendszerben, az adatbázist MYSQL szerver biztosítja.

**Felhasznált technológiák**

Backend

Laravel API(php keretrendszer)

• laravel -sanctum (Autentikációs kiegészítő csomag)

**Frontend**

**Mobil alkalmazás**

Megyeri Márk Máté

**Kódolási konvenciók**

A kódot git verziókezelővel használjuk.

**Alapkönyvtárak**

• api

• doc

• insomnia

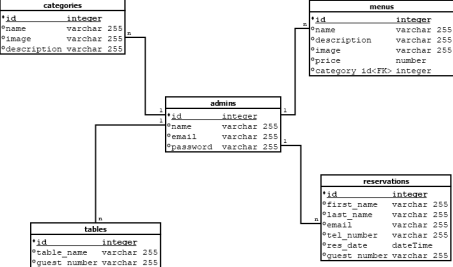
• mobil

• web

Az api backendet, a web a webes frontend, doc a dokumentációt, az insomnia pedig a REST API lekérdezéséhez tartalmazó segítséget tartalmazza

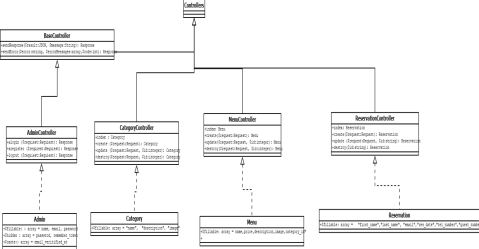
**Felülettervek**

Adatmodell



Megyeri Márk Máté

API UML terv



Megyeri Márk Máté

**Végpontok**

|  |  |  |  |
| --- | --- | --- | --- |
| **Végpontok** | **Metódus** | **Azonosítás** | **Leírás** |
| /st | post | igen | Kategória felvétel (admin) |
| /categories/{id} | put | igen | kategória frissítése |
| /destroyCategory/{id} | delete | igen | kategória törlése |
| /menuscreate | post | igen | menü létrehozása |
| /destroyMenu/{id} | delete | igen | menü törlése |
| /menus/{id} | put | igen | menü frissítése |
| /reservations | get | igen | foglalások (csoportos lekérdezés) |
| /reservations-create | post | nem | foglalás igénylése |
| /reservation/{id} | put | igen | foglalás frissítése (ha a vendégnek megváltott az adata) |
| /destroyReservation/{id} | delete | igen | foglalás törlése |
| /register | post | nem | regisztráció(admin) |
| /login | post | nem | belépés |
| /categories | get | nem | összes felvett kategória(csoportos lekérdezés) |
| /menus | get | nem | összes felvett menü (csoportos lekérdezés) |
| /logout | post | nem | admin kijelentkezés |

**Általános működés:**

A REST API http kéréseket fogad, melyek tartalmazzák a műveletekhez szükséges megfelelő adatokat. A titkos műveletek végpontjai védett, autentikáció szükséges a használatához.

Ilyenek a menük és a kategóriák létrehozása-módosítása-törlése. A felhasználók elérhetik a menük, kategóriákat , mivel nincs szükség autentikációra, mivel ezeknek a végpontjaik publikusak.

Az adatokat JSON formátumban fogadja és dolgozza fel. A vezérlést kontrollerek valósítják meg, minden adatkezelési csoportnak (autentikáció, kategória, menük,foglalások) külön kontrollerük van és itt történik meg az adatfeldolgozás.

A kontrollerek modellekkel vannak kapcsolatban, amelyek az adatkezelésért felelősek. Minden adatkezelési csoportnak (

Megyeri Márk Máté

admin, menük, kategóriák, foglalások) külön modellük van, itt történik az adatok adatbázisból kiolvasása, illetve az adatok kiírása adatbázisba.

A modellek adatbázis táblákkal vannak kapcsolatban, melyek az adatok tárolásáért felelősek.

Az adatbázis táblák adatait a modellek kezelik.

Megyeri Márk Máté

**Osztályok**

**Admincontroller**

Feladata egy új admin felvétele, admin autentikációja, admin kijelentkeztetése.

**Metódusok**

**1. alogin()**

➢ Feladata a felhasználó azonosítása név és jelszó alapján. Sikeres autentikáció esetén generál

➢ egy tokent a felhasználó számára és bejegyzi a personal\_access\_tokens adatbázis tábla

➢ megfelelő mezőjébe, majd átadja a BaseController sendResponse() metódusának a saját üzenettel együtt.

➢ *bejövő paraméterek*: $request (a bejelentkezéshez szükséges adatok a kérésben, name, password ).

➢ *kimenő adatok*: token, saját üzenet.

**2. logout()**

➢ Feladata a felhasználó kijelentkeztetése és a token törlése az adatbázis táblából.

➢ *bejövő paraméterek:* $request (a kijelentkezéshez szükséges adatok a kérésben, token)

➢ *kimenő adatok:* saját üzenet.

**3. aregister()**

➢ *bejövő paraméterek:* $request (a regisztrációhoz szükséges adatok a kérésben, name, email, password, confirm\_password) ➢ Az adatokat validálja, majd sikeres érvényesítés után bejegyzi az admins adatbázis tábla megfelelő mezőibe.

➢ *kimenő adatok:* saját üzenet.

Megyeri Márk Máté

**BaseController**

Feladata a bejelentkezési műveletek válaszainak küldése. Sikeres művelet esetén a generált azonosító tokent visszaküldi, sikertelen művelet esetén hibaüzenetet küld.

**Metódusok:**

***1. sendResponse()***

➢ *bejövő paraméterek:* $result (A generált token), $message (A felhasználónak szánt saját üzenet)

➢ kimenő adatok: Response (sikeres bejelentkezés esetén a token).

***2. sendError()***

➢ *bejövő paraméterek:* $error (A php által generált hibaüzenet), $errorMessages (saját hibaüzenet), $code (A válaszban küldendő http kód).

➢ *kimenő adatok:* Response (sikertelen bejelentkezés esetén hibaüzenet és a hiba kódja).

Megyeri Márk Máté

**CategoryController**

Feladata egy új kategória felvétele, meglévő kategóriák adatainak módosítása, meglévő kategóriák megjelenítése, meglévő kategóriák törlése.

**Metódusok:**

**1. index()**

➢ Visszaadja az adatbázisban szereplő kategóriák listáját csoportonként.

➢ *bejövő paraméterek:* NINCS

➢ *kimenő adatai:* Category (kategória modell amely tartalmazza a kategóriák adatbázisból lekért adatait.

**2. create()**

➢ Eltárol az adatbázisban egy új kategóriát.

➢ *bejövő paraméterek:* request (Kategóriák adatait tartalmazó kérés).

➢ *kimenő adatok*: Category (A kategóriák adatait tartalmazó modell) , egyéni üzenet.

**3. update()**

➢ Frissíti az adatbázisban tárolt egy adott kategória adatait. ➢ *bejövő paraméterek:* $request, $id (a kategóriák adatait tartalmazó kérés és a kategóriák azonosítója ).

➢ *kimenő adatok:* Category ( a kategóriák adatait tartalmazó modell) , egyéni üzenet.

**4. destroy()**

➢ Feladata a kiválasztott kategória törlése az adatbázisból. ➢ *bejövő paraméterek:* $id (A kiválasztott kategória azonosítója ). ➢ *kimenő adatok:* Category (A kiválasztott kategória adatait tartalmazó modell) , egyéni üzenet.

Megyeri Márk Máté

**MenuController**

Feladata egy új menü felvétele, meglévő menük adatainak módosítása, meglévő menük megjelenítése, meglévő menük törlése.

**Metódusok:**

**1. index ()**

➢ Visszaadja az adatbázisban szereplő menük listáját csoportonként. ➢ *bejövő paraméterek:* NINCS

➢ *kimenő adatai:* DB ( osztály, mely segítségével különböző adatbázis-műveleteket hajthatunk végre)

**2. create()**

➢ Eltárol az adatbázisban egy új menüt.

➢ *bejövő paraméterek:* request (Menük adatait tartalmazó kérés). ➢ *kimenő adatok*: Menu (A menük adatait tartalmazó modell), egyéni üzenet.

**3. update()**

➢ Frissíti az adatbázisban tárolt egy adott menü adatait.

➢ *bejövő paraméterek:* $request, $id ( a menük adatait tartalmazó kérés és a menük azonosítója ).

➢ *kimenő adatok:* Menu ( a menük adatait tartalmazó modell ), egyéni üzenet.

**4. destroy()**

➢ Feladata a kiválasztott menü törlése az adatbázisból.

➢ *bejövő paraméterek:* $id (A kiválasztott menü azonosítója ). ➢ *kimenő adatok:* Menu (A kiválasztott menü adatait tartalmazó modell) , egyéni üzenet.

Megyeri Márk Máté

**ReservationController**

Feladata egy új foglalás felvétele, meglévő foglalások adatainak módosítása, meglévő foglalások megjelenítése, meglévő foglalások törlése.

**Metódusok:**

**1. index ()**

➢ Visszaadja az adatbázisban szereplő foglalások listáját csoportonként.

➢ *bejövő paraméterek:* NINCS

➢ *kimenő adatai:* Reservation (foglalások modell amely tartalmazza a foglalások adatbázisból lekért adatait.

**2. create()**

➢ Eltárol az adatbázisban egy új foglalást.

➢ *bejövő paraméterek:* $request (Foglalások adatait tartalmazó kérés).

➢ *kimenő adatok*: Reservation (A foglalások adatait tartalmazó modell) egyéni üzenet.

**3. update()**

➢ Frissíti az adatbázisban tárolt egy adott foglalás adatait. ➢ *bejövő paraméterek:* $request, $id a foglalások adatait tartalmazó kérés és a foglalások azonosítója).

➢ *kimenő adatok:* Reservation (a foglalások adatait tartalmazó modell).

**4. destroy()**

➢ Feladata a kiválasztott foglalás törlése az adatbázisból. ➢ *bejövő paraméterek:* $id (A kiválasztott foglalás azonosítója. ➢ *kimenő adatok:* Reservation (A kiválasztott menü adatait tartalmazó modell) egyéni üzenet.

Megyeri Márk Máté

**Modellek**

**1. Admin**

A regisztrált felhasználók adatait, illetve az azonosító tokent amelynek a generálása a felhasználóhoz kapcsolódik, írja az adatbázis megfelelő táblájába, valamint onnan szolgáltatja szükség szerint. Az Admin modell az admins adatbázis táblával van kapcsolatban az adatokat innen olvassa és ide írja. Az osztályt a Laravel projekt generálja, használata a Sanctum autentikációs kiegészítő csomagon keresztül történik.

**2. Category**

Az adatbázisban felvett kategóriák adatait kezeli, a CategoryController megfelelő metódusai által kért adatot adja vissza az adatbázis megfelelő táblájából, illetve a kontrollerből érkező adatokat írja ki. Az adatbázis categories táblájával van kapcsolatban, az adatokat innen olvassa és ide írja .

Mezők:

$fillable ( array ) elemei: name, image, description.

A fillable mező elemei egy – egy mező a categories adatbázis táblában. Biztonsági okokból csak a felsorolt mezőket írhatja a modell az adatbázisban.

**3. Menu**

Az adatbázisban felvett menük adatait kezeli, a MenuController megfelelő metódusai által kért adatot adja vissza az adatbázis megfelelő táblájából, illetve a kontrollerből érkező adatokat írja ki. Az adatbázis menus táblájával van kapcsolatban, az adatokat innen olvassa és ide írja.

Mezők:

$fillable ( array ) elemei: name, image, description, price, category\_id. A fillable mező elemei egy – egy mező a menus adatbázis táblában. Biztonsági okokból csak a felsorolt mezőket írhatja a modell az adatbázisban.

Megyeri Márk Máté

**4. Reservation**

Az adatbázisban felvett foglalások adatait kezeli, a ReservationController megfelelő metódusai által kért adatot adja vissza az adatbázis megfelelő táblájából, illetve a kontrollerből érkező adatokat írja ki. Az adatbázis reservations táblájával van kapcsolatban, az adatokat innen olvassa és ide írja.

Mezők:

$fillable ( array ) elemei: first\_name, last\_name, email,res\_date, tel\_number, guset\_number. A fillable mező elemei egy – egy mező a reservations adatbázis táblában. Biztonsági okokból csak a felsorolt mezőket írhatja a modell az adatbázisban.

Megyeri Márk Máté

**Mobil alkalmazás felülete**

Megyeri Márk Máté

**Fejlesztői környezet**

Pl.:

XAMPP

Adatbázis kezelése a mysql nevű klienssel. PHP

Visual Studio Code

Laravel

.

.

Megyeri Márk Máté

**Ismert hibák**

Ha frissíteni (updatelni) szeretnénk a menüt, vagy a kategóriákat nem sikerül a felvett kép miatt.

Megyeri Márk Máté

**Fejlesztési lehetőség**

o felhasználó regisztrálása, bejelentkeztetése.

o online ételrendelés.

o a képpel való frissítés a menük és a kategóriáknál.

o automatizált email a feliratkozott vendégeknek az aktuális akciókról, napi menükről, rendezvényekről.

o mobilapplikáció