1. Adatbázis

Az Adatbázis a weboldalunk alapvető és kritikus eleme, amely két fő célra használatos. Elsődlegesen szolgál a felhasználók hitelesítésére, ugyanis itt tároljuk azokat az adatokat, amelyek az azonosításhoz és az egyedi felhasználói profilok létrehozásához szükségesek, mint a felhasználónevek, jelszavak, jogosultságok és id. Másrészt az adatbázis tartalmazza a feladat eredetileg meghatározott tábláit és az importálást követően a letöltött adatokat.

A felhasználóknak lehetőségük van üzenetet küldeni az üzemeltetőnek. Ezek az üzenetek szintén az adatbázisban vannak tárolva. Egy-egy rekordon, a felhasználó azonosítójával megállapítható ki küldte az üzenetet.

Miután bejelentkezés nélkül is van lehetőség üzenetküldésre, ilyenkor pedig nincs azonosítható felhasználó, null értékkel kerülnek rögzítésre az üzenetek, megjelenítéskor ezek a rekordok a „Vendég” megnevezést kapják.

* 1. Adatbázis táblák
     1. anyag tábla:

A tábla a korlátozott darabszámban kiadott magyar forint érmékhez használt nyersanyagokat tartalmazza. Minden rekord egy konkrét fémet reprezentál. A femid mező, a tábla elsődleges kulcsa, mely azonosítja a fémeket.

A képen szöveg, képernyőkép, Betűtípus látható

Automatikusan generált leírás

* + 1. tervezo tábla:

A képen szöveg, képernyőkép, Betűtípus látható

Automatikusan generált leírásA tervezo tábla a forint érmék tervezőinek adatait tárolja. Minden sor egy konkrét tervezőt reprezentál, beleértve a tervező egyedi azonosítóját és nevét. Ez a tábla a tervezők azonosítására szolgál, amely lehetővé teszi az egyes érmék készítőinek nyomon követését.

* + 1. erme tábla:

Az erme tábla a limitált darabszámban kiadott magyar forint érmék alapvető adatait tárolja. Minden sor egy érmeadatot reprezentál, annak címletével, tömegével, kiadott darabszámával, valamint a kibocsátás és bevonás dátumával. Ez a tábla az érmék részleteinek, információinak tárolására szolgál.

A képen szöveg, képernyőkép, Betűtípus, szám látható

Automatikusan generált leírás

* + 1. akod tábla:

Az akod tábla egy kapcsolótábla. Az alapanyagoknak használt fémek, azaz anyag tábla és az érmék adatait tartalmazó erme adatbázistábla adainak kapcsolatait tartalmazza. A kapcsolótáblára azért volt szükség, mert az anyagok és érmék kapcsolata több a többhöz, egy érméhez több alapanyagot is használhattak, valamint egy alapanyagot több érme anyagaként is felhasználtak.

A képen szöveg, képernyőkép, Betűtípus, sor látható

Automatikusan generált leírás

* + 1. tkod tábla:

A tkod tábla szintén egy kapcsolótábla, ami a tervezők és érmék kapcsolatait reprezentálja. Az érmék és tervezőik közt szintén több a többhöz a kapcsolat, több tervező is közreműködhetett egy érme megalkotásában, valamint egy-egy tervező több érménél is szerepet játszott.

A képen szöveg, képernyőkép, Betűtípus, sor látható

Automatikusan generált leírás

Az adatbázis elkészülte után a Laravelben a megfelelő modelleket valósítottuk meg, amelyek közvetlen kapcsolatot biztosítanak az adatbázis és az alkalmazás logikája között. Ezek segítségével könnyen kezelhetők az adatbázis műveletek, mint például az adatok lekérdezése, beszúrása, frissítése vagy törlése. A modellek reprezentálják az egyes táblákat, könnyítik az adatokkal való munkát, támogatják a kapcsolatok kezelését és ezek az üzleti logika központjai.

Az adatbázis tartalmaz a fent felsorol táblákon kívül több adattáblát is, mint a user, sessions, cache táblák. Ezek nem szerepeltek az importálandó állományban, viszont a Laravel automatikusan létrehozta ezeket a megfelelő működés miatt. Ezek a táblák az alkalmazás működéséhez szükségesek és bizonyos komponensek alapértelmezett konfigurációjával együtt kerülnek létrehozásra.

1. REST végpontok

A REST hívások HTTP protokollon keresztül történnek. Általában négy fő műveletet használunk, GET az adatok lekérésére, POST új adatok rögzítéséhez, PUT az adatok frissítésére, módosítására szolgál, végül a DELETE végpontok adatok törlésére használtak.

A kérések törzs adata (amennyiben van), többnyire xml vagy json formában kerül küldésre, valamint jellemzően ugyan így kapja vissza a választ a kliens, kiegészülve még egy lehetőséggel, a html-el.

A REST végpontokhoz Basic authentikáció szükséges. Az oldalon regisztrációkor megadott e-mail cím és jelszó használható a végpontok elérésére.

A routes/api.php fájlban definiáltuk a Laravel alkalmazás API végpontjait, amelyek RESTful kéréseket fogadnak. Ez a fájl külön van választva a webes útvonalaktól, hogy elkülönítse az API-funkcionalitást. Itt határozzuk meg, hogy milyen HTTP metódusokat használhatnak a kliensek, és melyik controller kezelje ezeket a kéréseket.

Az API útvonalak automatikusan a /api prefixel vannak ellátva, például egy érme adatainak lekérése a GET /api/erme végponton történhet. Ez a fájl tehát az API-központú forgalom irányításának egyik legfőbb része.

Fejlesztés során Postmannel teszteltük a végpontokat, ezen kérésekről, egy importálható Postman collection elérhető a GIT Repositoryban, minden a következőkben felsorolt végponthoz.

Az elkészült végpontokat cUrl-el is teszteltük, itt például a 25-ös azonosítójú érme törlése látható.



* 1. Végpontok

Az Adatbázis mind az 5, a feladat által meghatározott adattáblája lekérhető GET metódussal, valamint az érme táblához kapcsolódóan megvalósításra kerültek POST, PUT és DELETE végpontok is.

* + 1. A képen szöveg, képernyőkép, Betűtípus látható

       Automatikusan generált leírás/api/anyag GET végpont

Az anyag tábla az érmék alapanyagaival elérhető, http GET kérést indítva a következő címen: <http://katonazsoltweb2.testhosting.hu/api/anyag> . Az adathalmaz szűkítésére femid query paraméter használható. A képen a válasz json formátuma látható.

* + 1. A képen szöveg, képernyőkép, Betűtípus látható

       Automatikusan generált leírás/api/tervezo GET végpont

A tervezők adatainak eléréséhez http GET kérést szükséges indítani a következő címre: <http://katonazsoltweb2.testhosting.hu/api/tervezo> Az adathalmaz szűkítésére a tid query paraméter használható. A képen a válasz json formátuma látható.

* + 1. A képen szöveg, képernyőkép, Betűtípus látható

       Automatikusan generált leírás/api/akod GET végpont

Az anyag-érme táblák kapcsolótáblája a <http://katonazsoltweb2.testhosting.hu/api/akod> címre indított http GET kéréssel érhető el. Az adathalmaz szűkítésére query paraméterként a femid megadható. A képen a válasz json formátuma látható.

* + 1. A képen szöveg, képernyőkép, Betűtípus látható

       Automatikusan generált leírás/api/tkod GET végpont

Az érme-tervező kapcsolótábla adatai szintén elérhetők, ehhez http GET kérést kell indítani a <http://katonazsoltweb2.testhosting.hu/api/tkod> címre. A tervezoid query paraméterrel szűrhető az adathalmaz. A képen a válasz json formátuma látható.

* + 1. /api/erme végpontok

A képen szöveg, képernyőkép, Betűtípus látható

Automatikusan generált leírásA program lehetőséget nyújt új érme felvételére, meglévő módosítására és akár egy érme törlésére is az adatbázisból.

Érme adatok lekérése az előző végpontokhoz hasonlóan történik. GET http kérést kell indítani a <http://katonazsoltweb2.testhosting.hu/api/erme> címre, melyet az ermeid query paraméterrel szűrhetünk. A képen a válasz json formátuma látható.

Új adat rögzítésekor a használt Url nem különbözik a GET kérésnél használttól, csupán annak metódusát szükséges módosítani.

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Automatikusan generált leírás

Egy ilyen kérésnél a küldött adatok vizsgálata elhagyhatatlan, így az ErmeControllerhez kapcsolódóan létrehoztunk egy ErmeRequest osztályt, mely az adatok validációs feladatait látja el. Ez ellenőrzi a bejövő kérést, megvizsgálja a mezők értékét, típusát, kötelezőségét és ha minden rendben talál, visszaadja a vezérlést az ErmeControllernek és folytatódhat az adatrögzítés, módosítás. Ellenkező esetben, a probléma okával tér vissza.

Már meglévő rekord módosításakor az url-ben szerepelnie kell az érme azonosítójának a kérést törzsében pedig a többi adatnak a POST kéréshez hasonlóan. Módosításkor az url-ben megadott id-val rendelkező érme adatait módosítjuk.

A képen szöveg, Betűtípus, képernyőkép látható

Automatikusan generált leírásTörlés szintén lehetséges, erről korábban cUrl-t is csatoltunk. Egy érme törléséhez http DELETE kérést kell indítani a /api/erme végpontra. Amennyiben a megadott azonosító nem létezik, a kérést elutasítjuk 404-es státuszkóddal.

1. MNB SOAP kliens

A Laravel nem tartalmaz alapértelmezetten SOAP implementációt, miután ez a REST-nél egy régebbi, mára kevésbé népszerű protokoll. A feladat elvégzésében a Laminas keretrendszert használtuk, ami a számos összetevője közt a SOAP szerverek kezelésében is segítséget nyújt.

Az MNBService osztály egy szolgáltatásosztály, mely az MNBController-t szolgálja ki és SOAP kliens segítségével kommunikál a Magyar Nemzeti Bank (MNB) hivatalos árfolyamadatokat nyújtó webszolgáltatásával. Az osztály felelős az MNB SOAP végpontjainak eléréséért, az árfolyamadatok lekérdezéséért és azok feldolgozásáért.

A getWsdl() metódus a WSDL fájl tartalmát tölti le, mely alapján létrehozható volt a többi metódus olyan célokra mint az elérhető devizanemek letöltése, a napi árfolyamok letöltése, havi árfolyamok letöltése.

Az MNBService egy robusztus SOAP kliens, amely egyszerűsíti az MNB webszolgáltatásaival való integrációt.

Az MNBController osztály a Magyar Nemzeti Bank SOAP szolgáltatásához készült kliensfukciókat biztosítja az alkalmazás nézetei számára.

A devizanemeket több lenyíló lista is megjeleníti, melyeket adatbázisban nem tárolunk külön, de az oldal megfelelő sebessége érdekében ezeket gyorsítótárazzuk. A devizanemeket a rendszer a Laravel Cache rendszerében tárolja.

Az MNB szolgáltatásokhoz kapcsolódó útvonalak, a web.php-ban kerültek meghatározásra. Az összes végpont egy /mnb prefixel van ellátva, így külön csoportosítva az alkalmazás egyéb végpontjaitól.

A nézeteket, vagyis a weboldalon történő megjelenítést a resources/views/MNB és a resources/view/livewire mappában található php fájlok valósítják meg, utóbbi a dinamikus tartalommegjelenítésben nyújt segítséget.