Vendéglátói minta projekt bemutatása!

2021.szept-2022.ápr.-ig tartott.

A program vendéglátói egységeknek készült. Szerintem könnyebb lenne (és persze gyorsabb is), ha felváltaná az elektronika a papírt és a ceruzát. Gyorsabb és könnyebb egy ”tablet”/”telefon” a kézben és ott megrendelni az ételeket, italokat, ami rögtön megjelenik a konyhán.

A fő cél, a vendéglátó egységek könnyítése. Vendéglátói végzetségem van (szakács) és sokszor tapasztaltam már, hogy a papír és a toll/ceruza ideje lejárt. „Haladni kell a korral”. A programunk ezt az űrt próbálja betölteni. Mivel a pincéreknek sokat kell mászkálniuk (rendelés felvétel, étel/ital kihordás, stb…) ezért ezzel a kis alkalmazással igyekszünk nekik és persze a többieknek is segíteni. A vendégnek is könnyebb talán, mert ott helyben mondja mit szeretne enni-inni, a pincér felveszi a rendelést és már azonnal tovább is tudja küldeni a konyhának és a söntésnek (pult, jobb megnevezés). A konyha már neki is tud fogni elkészíteni és így a vendég hamarabb kapja meg a rendelését, hamarabb is élvezheti az élvezeti cikkeket. Az üzletnek sem hátrány a szoftver, mert így kevesebb az elírás, az olvashatatlanság. Pontosabb a megrendelés és ezáltal kevesebb a rontási esély felesleges idegeskedés, selejtezés. A vendéglátó egység még annak is örül, hogy ilyen gyorsan kapja meg a vendég a kért ételét-italát, mert hamarabb kapja meg és ezáltal hamarabb is távozhat, ami azt követi, hogy több lehet a vendége, ezáltal több a bevétele is. A projektet nem kell telepíteni sehova mivel HTML alapú. Mindenhol azonnal elérhető. Könnyen megtanulható, nem bonyolult applikáció. Manapság már mindenkinek van otthon számítógépe, okos telefonja (főleg a fiataloknak), internete. Nem kell telepíteni. Én a BACKEND-el foglalkoztam.

***Felhasznált technológiák***

A háttért PHP-val valósítottam meg, mivel számomra ez volt a legegyszerűbb. Igyekeztem a legbiztonságosabb és a legegyszerűbb módot alkalmazni. Mint biztonság és technika a PHP PDO-t használtam szinte mindenhol. Visual Studio Code-ot, Git-Hub-ot, Mysql-t és insomniát preferáltam előtérben.

***Fejlesztői környezet***

* *Szoftverek amiket használtam!*
* *Visual Studio Code*

A kód megírásához a „ Visual Studio Code”-ot használtam, mert az egyik legegyszerűbb, legnépszerűbb app. Támogatja a hibakeresőket, valamint beépített [Git](https://hu.wikipedia.org/wiki/Git" \o "Git) támogatással rendelkezik, továbbá képes az intelligens kódkezelésre az [IntelliSense](https://hu.wikipedia.org/w/index.php?title=IntelliSense&action=edit&redlink=1" \o "IntelliSense (a lap nem létezik)) segítségével. Ezen felül testre szabható, így a felhasználók megváltoztathatják a kinézetet (témát), megváltoztathatják a szerkesztő gyorsbillentyű-kiosztását, az alapértelmezett beállításokat és még sok egyebet. A Visual Studio Code az Electron nevű keretrendszeren alapszik, amellyel asztali környezetben futtatható [Node.js](https://hu.wikipedia.org/wiki/Node.js) alkalmazások fejleszthetőek.

* *MySql*

A MySQL az egyik legelterjedtebb adatbázis-kezelő, teljesen [nyílt forráskódú](https://hu.wikipedia.org/wiki/Ny%C3%ADlt_forr%C3%A1sk%C3%B3d) [LAMP](https://hu.wikipedia.org/wiki/LAMP_(szoftvercsomag)) ([Linux](https://hu.wikipedia.org/wiki/Linux)–[Apache](https://hu.wikipedia.org/wiki/Apache_HTTP_Server" \o "Apache HTTP Server)–**MySQL**–[PHP](https://hu.wikipedia.org/wiki/PHP)) összeállítás részeként költséghatékony és egyszerűen beállítható megoldást ad [dinamikus](https://hu.wikipedia.org/wiki/Dinamikus_weboldal" \o "Dinamikus weboldal) webhelyek szolgáltatására. A MySQL adatbázisok adminisztrációjára a mellékelt [parancssori](https://hu.wikipedia.org/wiki/Parancssor) eszközöket (*mysql* és *mysqladmin*) használhatjuk. Széles körben elterjedt és népszerű adminisztrációs eszköz a PHP nyelven írt, nyitott forráskódú *[phpMyAdmin](https://hu.wikipedia.org/wiki/PhpMyAdmin" \o "PhpMyAdmin)*. A *phpMyBackupPro* (amelyet szintén PHP-ban írtak) adatbázisok (akár időzített, ismétlődő) mentésére szolgál eszközül.

* *PHP*

A **PHP** egy általános [szerveroldali](https://hu.wikipedia.org/wiki/Szerver) [szkriptnyelv](https://hu.wikipedia.org/wiki/Szkriptnyelv" \o "Szkriptnyelv) dinamikus weblapok készítésére. Az első szkriptnyelvek egyike, amely külső fájl használata helyett HTML oldalba ágyazható. A kódot a [webszerver](https://hu.wikipedia.org/wiki/Webszerver) PHP feldolgozómodulja értelmezi, ezzel dinamikus weboldalakat hozva létre. A PHP nyelv népszerűsége képességeinek bővülésével folyamatosan nőtt. A hagyományos [HTML](https://hu.wikipedia.org/wiki/HTML) lapokkal ellentétben a kiszolgáló a PHP-kódot nem küldi el az ügyfélnek, hanem a kiszolgáló oldalán a PHP-értelmező motor dolgozza fel azt. A programokban lévő HTML elemek érintetlenül maradnak, de a PHP kódok lefutnak. A kódok végezhetnek [adatbázis](https://hu.wikipedia.org/wiki/Adatb%C3%A1zis)-lekérdezéseket, létrehozhatnak képeket, [fájlokat](https://hu.wikipedia.org/wiki/F%C3%A1jl) olvashatnak és írhatnak, kapcsolatot létesíthetnek távoli kiszolgálókkal. A PHP-kódok kimenete a megadott HTML elemekkel együtt kerül az ügyfélhez. A PHP-t parancssori alkalmazásként is telepítik, így alkalmas parancsfájlok készítésére.

* *Insomnia*

Kiváló minőségű API-k biztosítása szabványok és az Insomnia API tervezési platformmal való együttműködés révén.  Egyszerűsített API kliensünkkel gyorsan és egyszerűen küldhet REST, SOAP, GraphQL és GRPC kéréseket közvetlenül az Insomnián belül. API-k tervezése OpenAPI használatával.

* *Git-Hub*

Ahhoz, hogy fel tudjak tölteni az oldalra, regisztráció szükséges, de a nyilvános tárolókat bárki szabadon böngészheti és letöltheti. Egy regisztrált felhasználóval lehetőség nyílik megbeszélésekre, tárolók kezelésére, mások tárolóihoz való hozzájárulásra és a kódban történt változások megtekintésére.

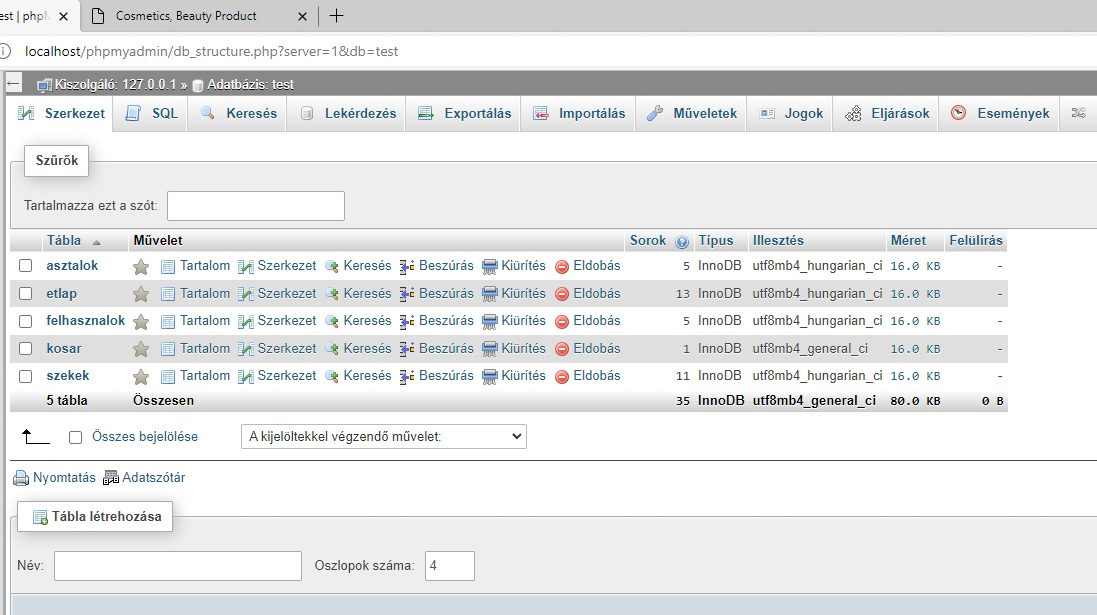
* *Hardver, amiben dolgoztam!*

Amd A8-as, 3000Mhz-en működő processzor, 16Gb RAM, 120Gb-os SSD és egy AMD Radeon R7 200 Series (2Gb)-os videokártya. Win 10-es operációs rendszerben. A monitor egy 24”-os Samsung. 16:9-es képarányban.

* *Kialakított adatszerkezet*

Az adatbázist MySql-lel végeztem. Több táblát és táblákon keresztül több mezőt készítettem. Mindegyik táblához saját kódot írtam. A kódokat igyekeztem a lehető legbiztonságosabbá tenni. PHP PDO a választásom.

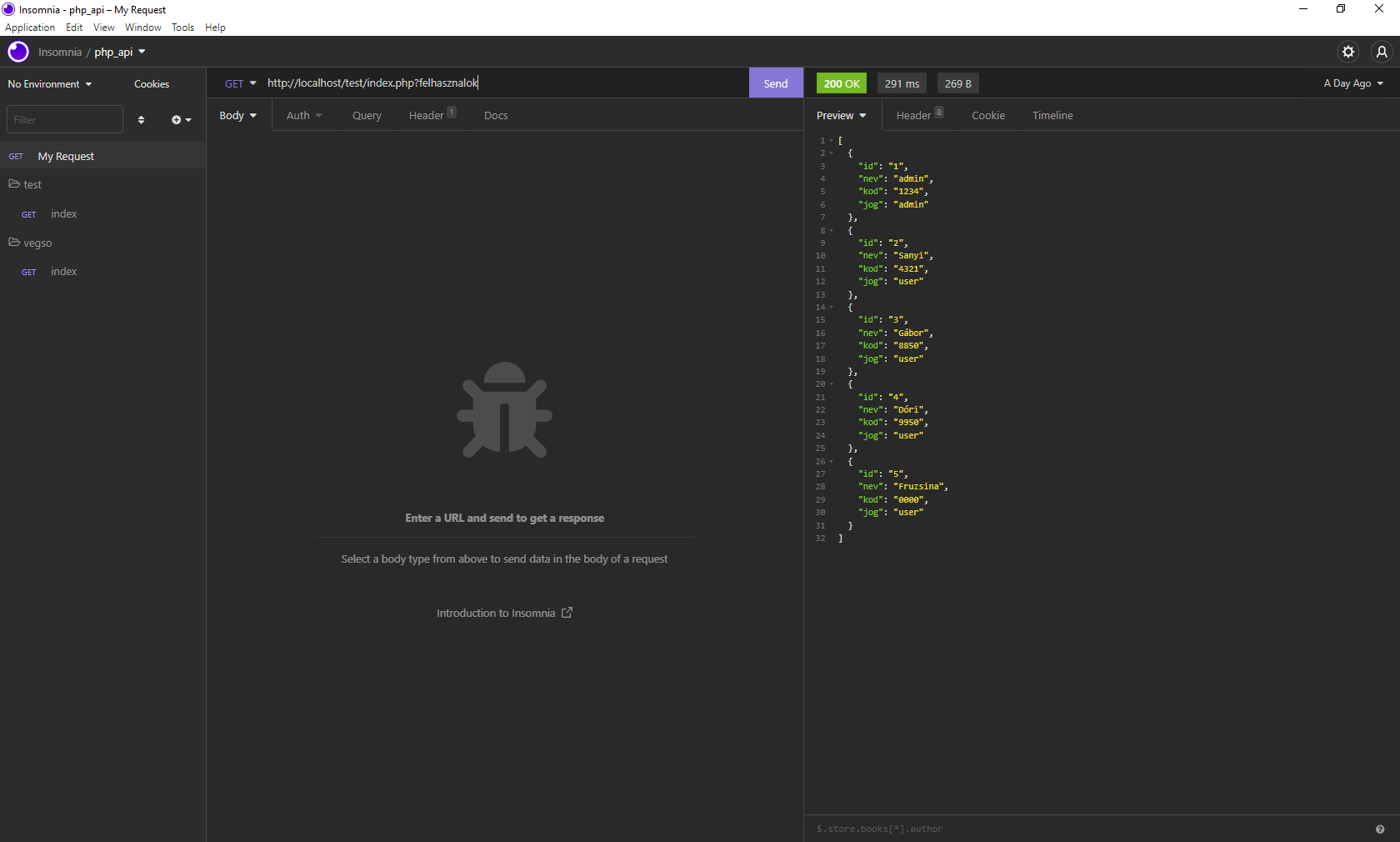
* *Adatbázis:*



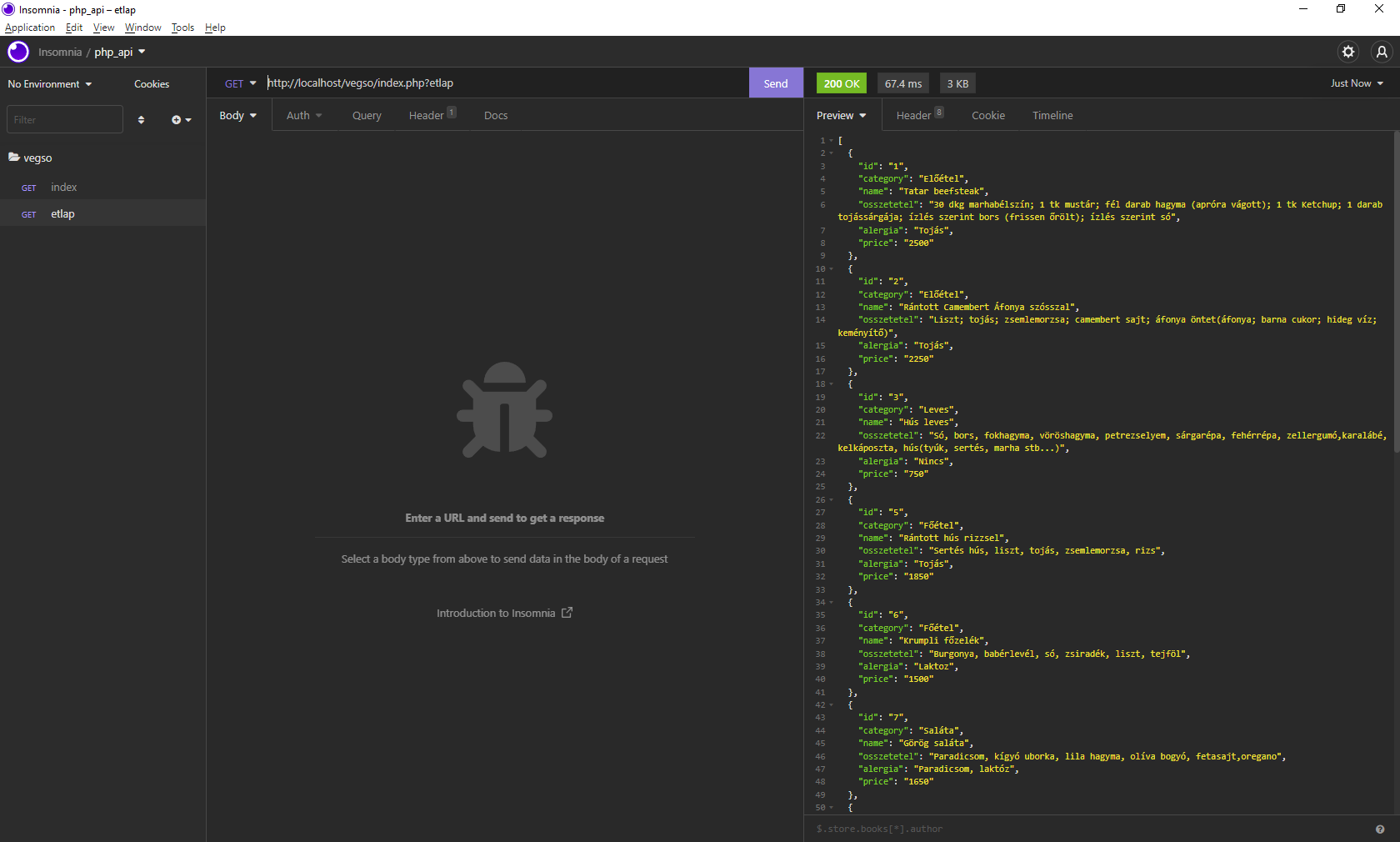
Az adatbázisban van öt tábla. Név szerint:

1. asztalok
2. etlap
3. felhasznalok
4. kosar
5. szekek

Az adatbázist localhost segítségével érem el insomnián keresztül, hogy Json formátumot kapjak vissza, és azt is tudjak visszaadni. Azt tudom, hogy van még egy jó program (ami talán a legjobb) postman.

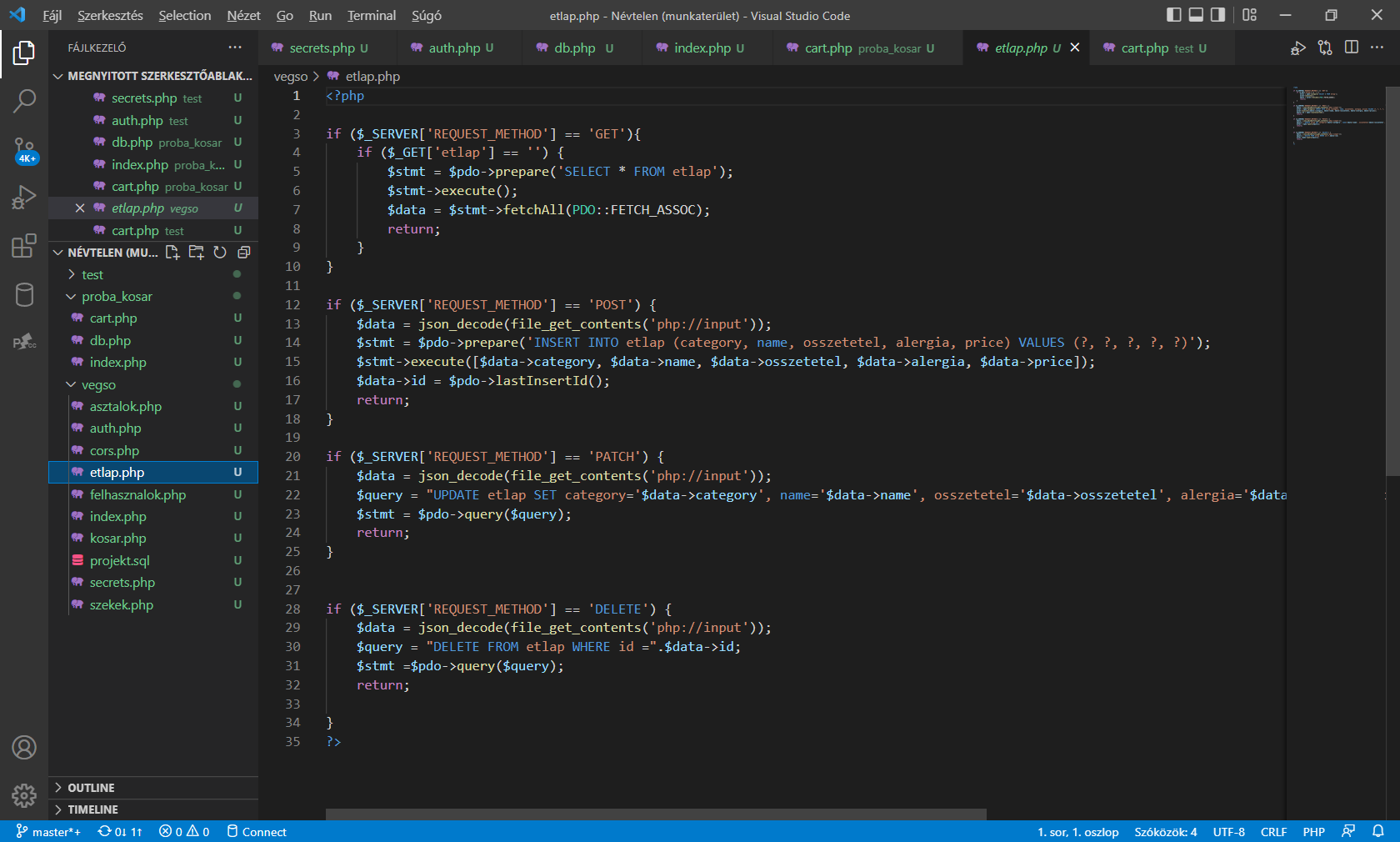


-***etlap***



Insomniában, Json formátumnak köszönhetően gyorsan és átlátható módon mutatja meg.

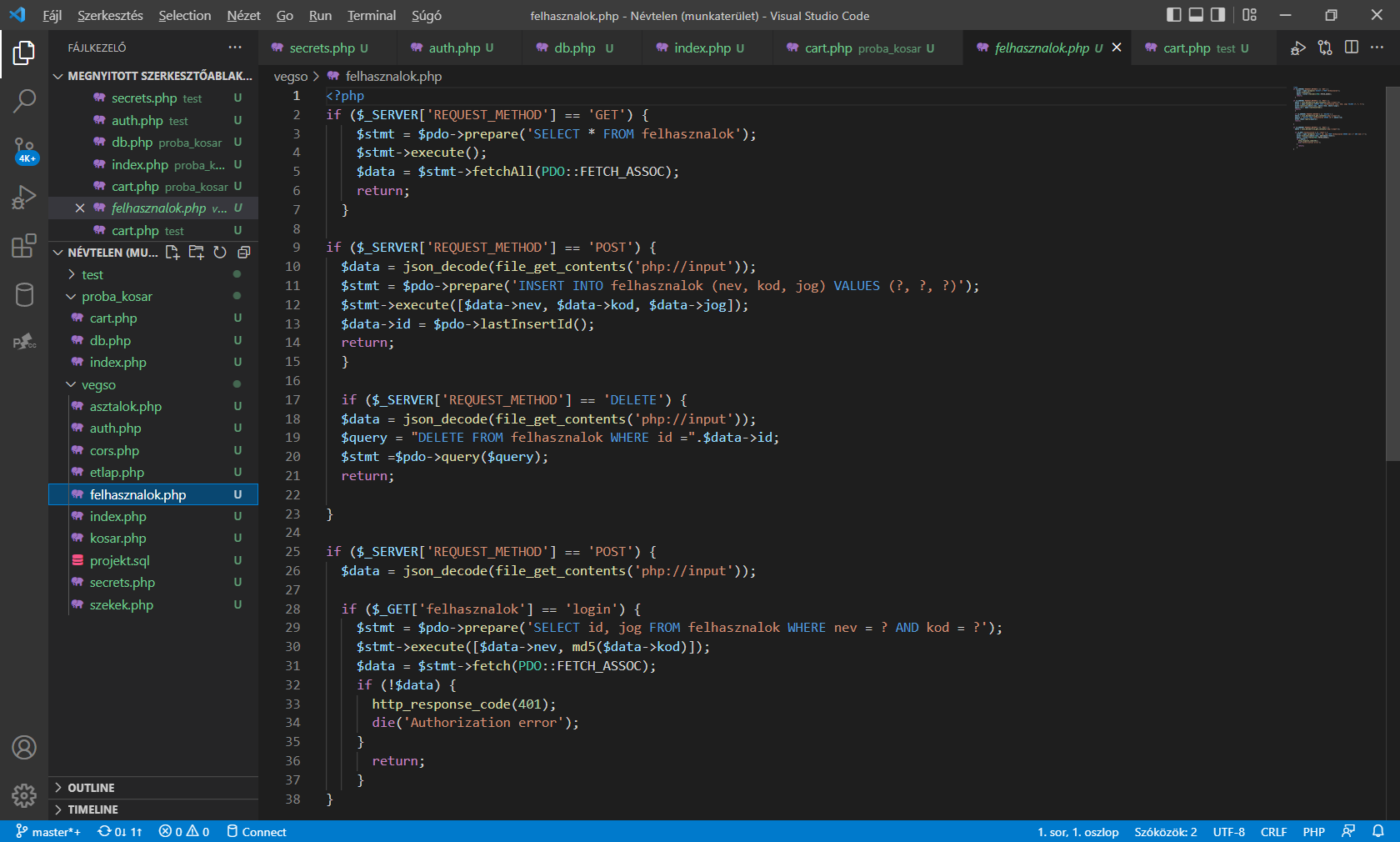
Az étlapban vannak azok a mezők amik a legfontosabbak (ételek, italok).



Itt mindegyik method-us megjelenik. GET-POST-PUT-DELET!

-*felhasznalok:*

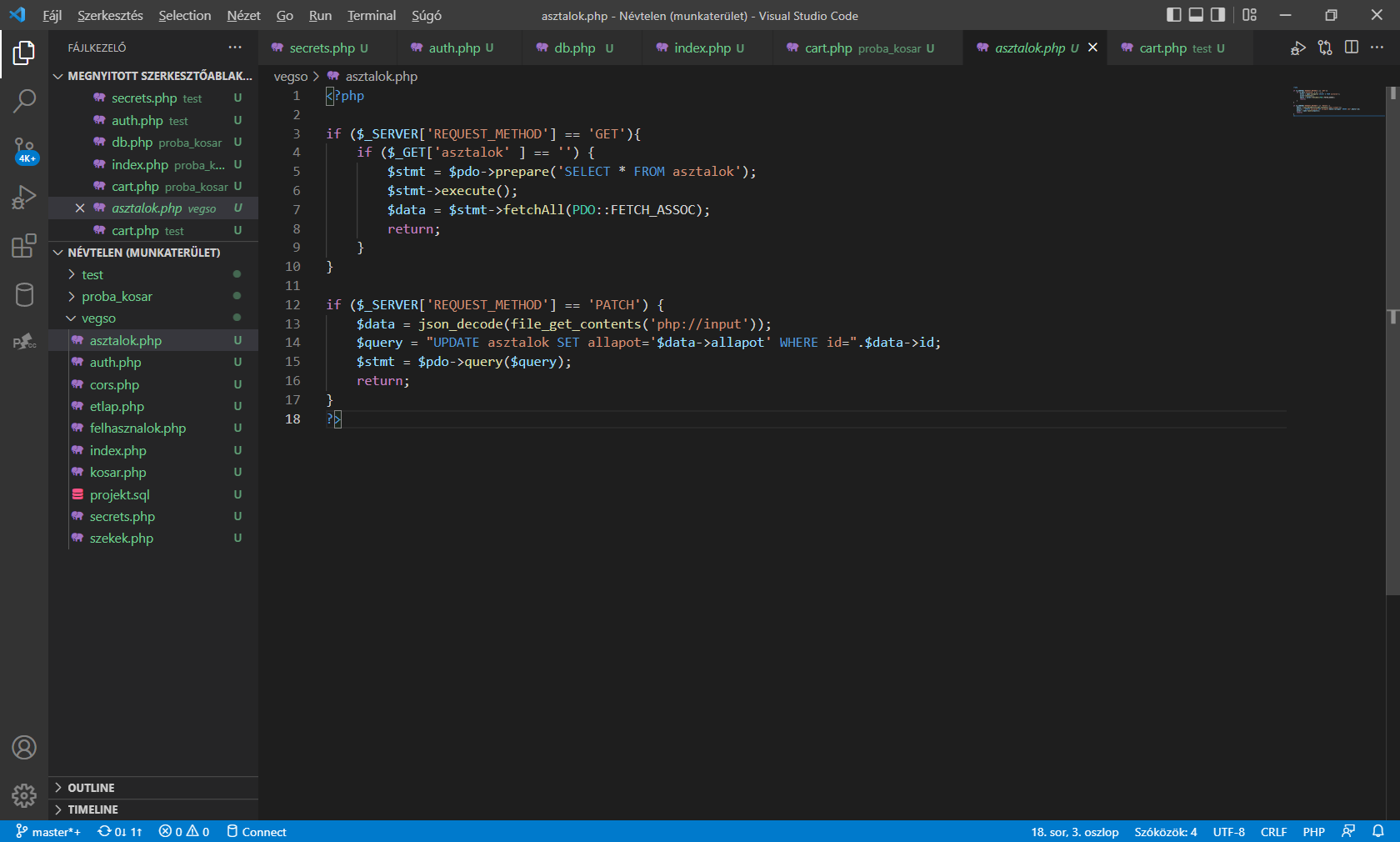
A felhasználóknál nem csak az összes method-ot raktam bele, hanem még egy pluszt is. Kódot.

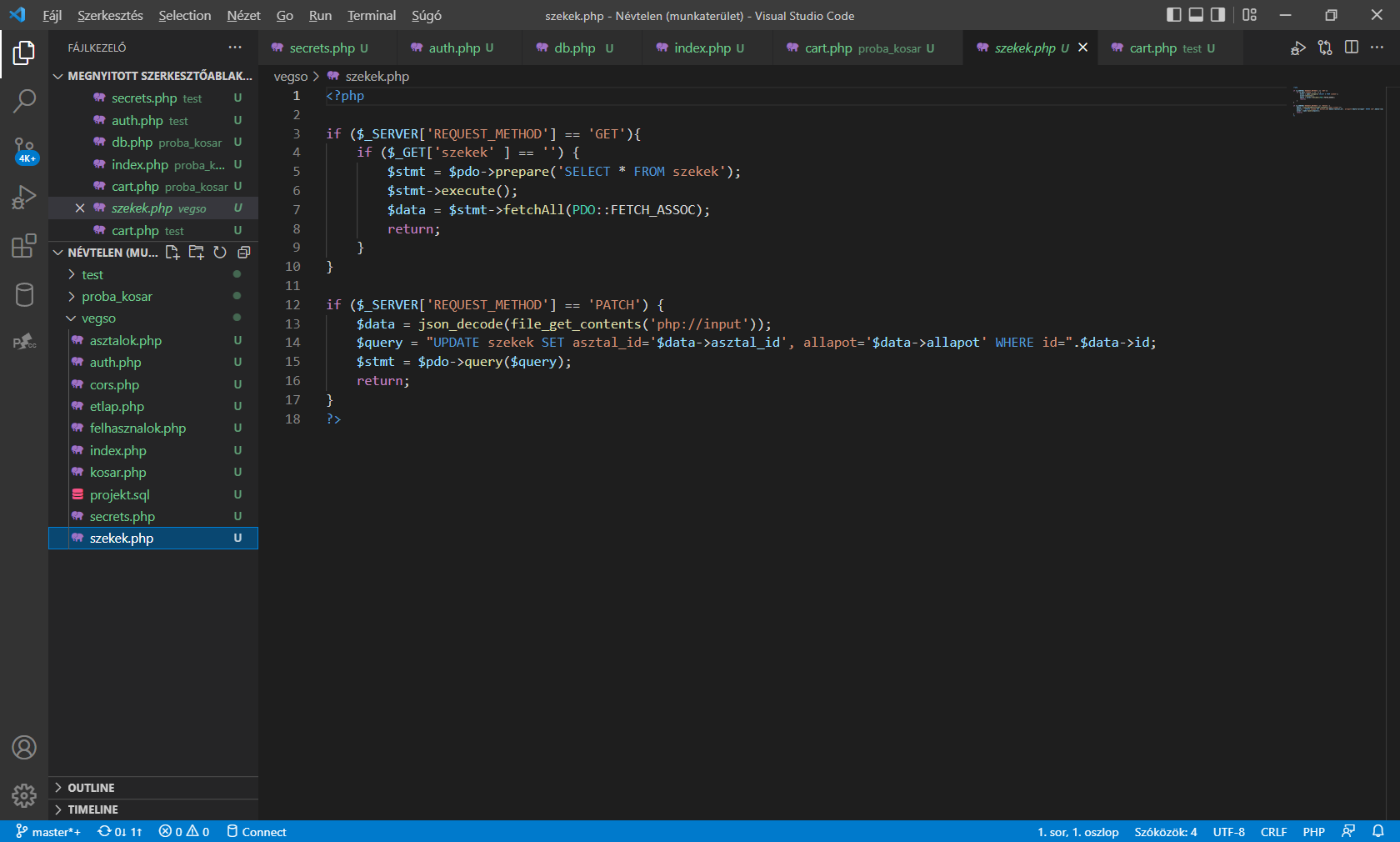


Ez a kód (név és egy hozzá tartozó egyedi kód) egy biztonsági rendszer, hogy idegen NE tudjon belepiszkálni.

*-Szekek, asztalok:*

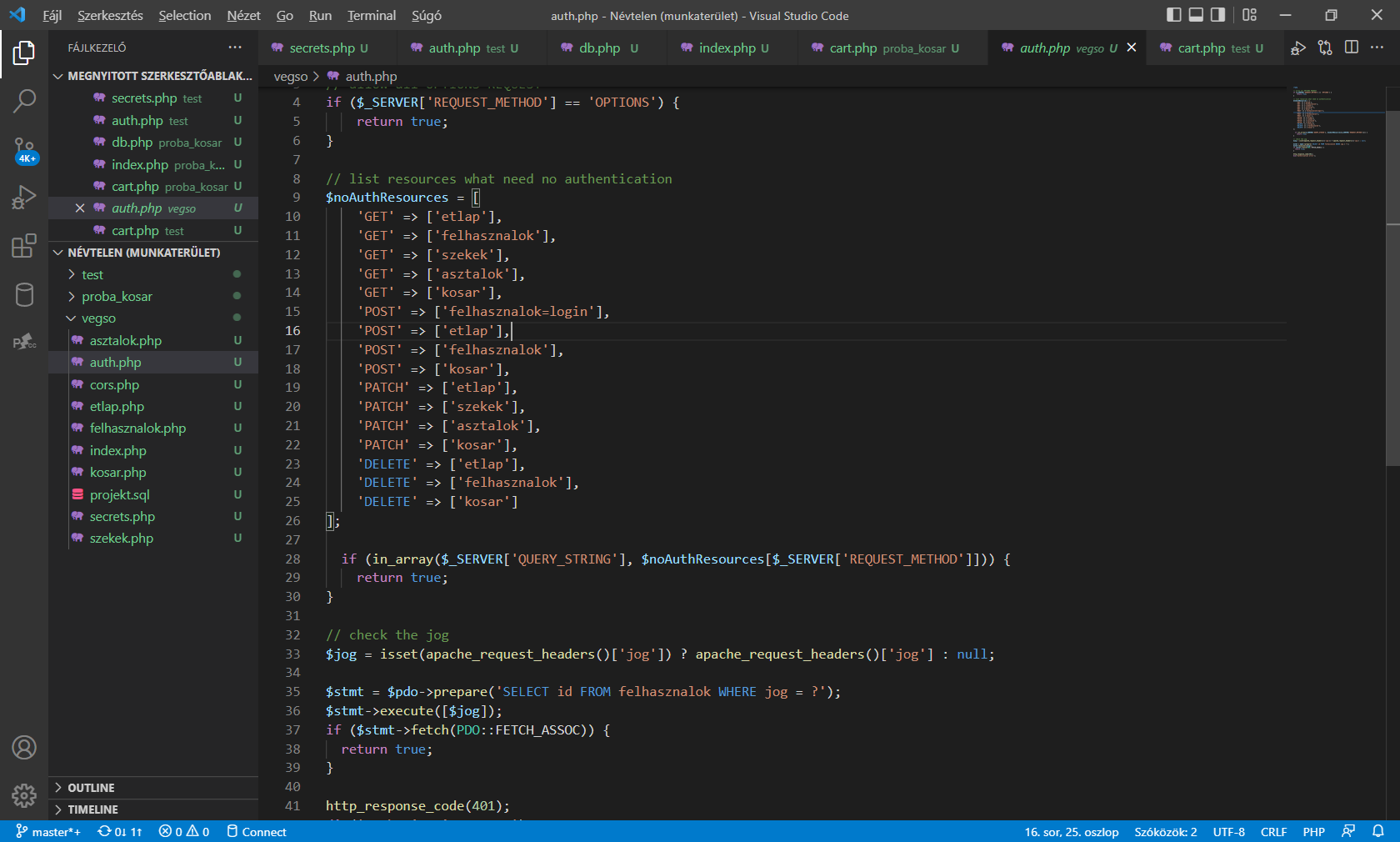
Mindkettőnél csak lekérés és változtatás van.





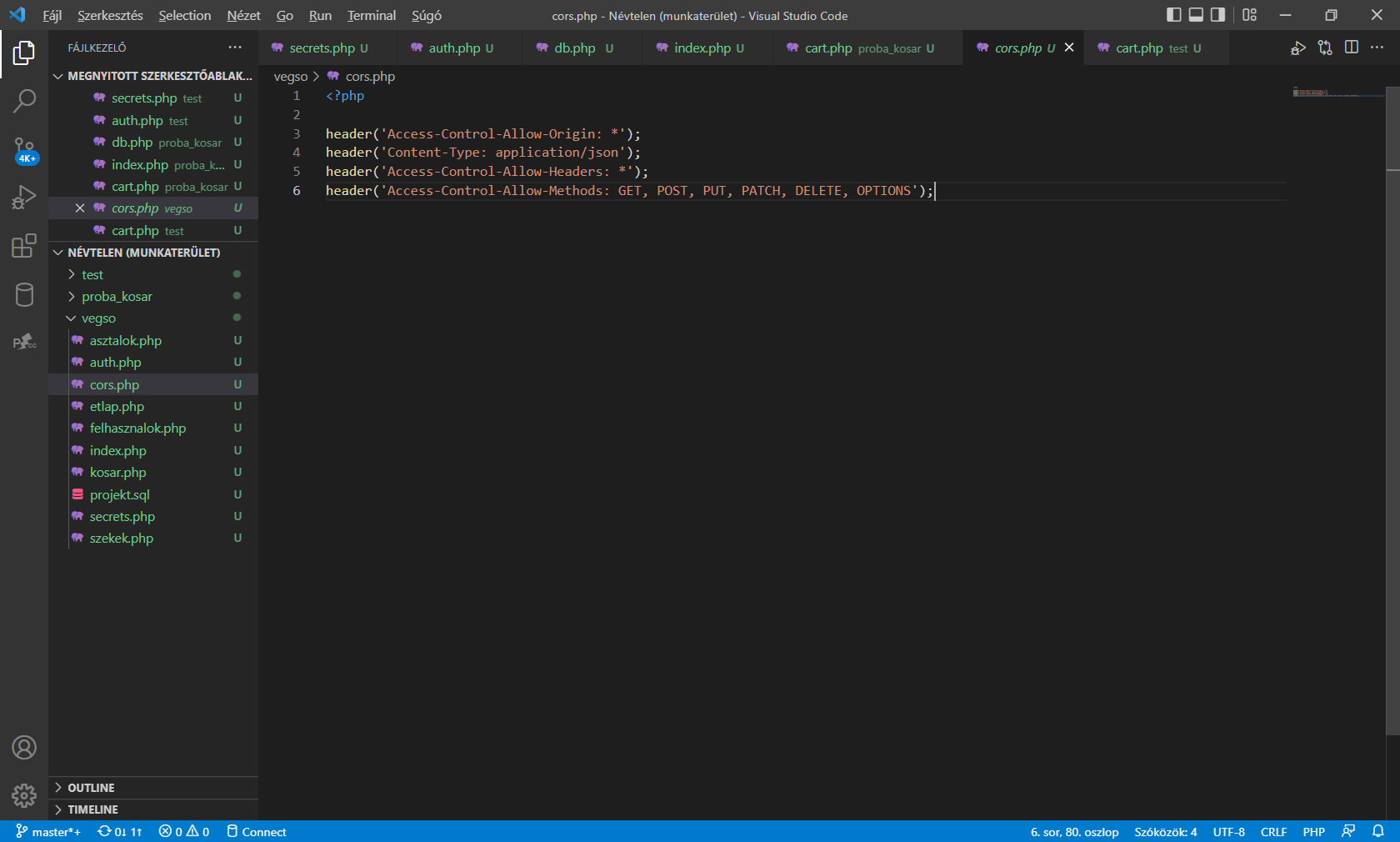
*-auth:*

Az auth-ban vannak a method-usok megjelölése (mivel mit lehet csinálni), illetve ebben van még a jogosultság egy része is. Csak akkor lehet bármit is csinálni, ha be van jelentkezve az illető.



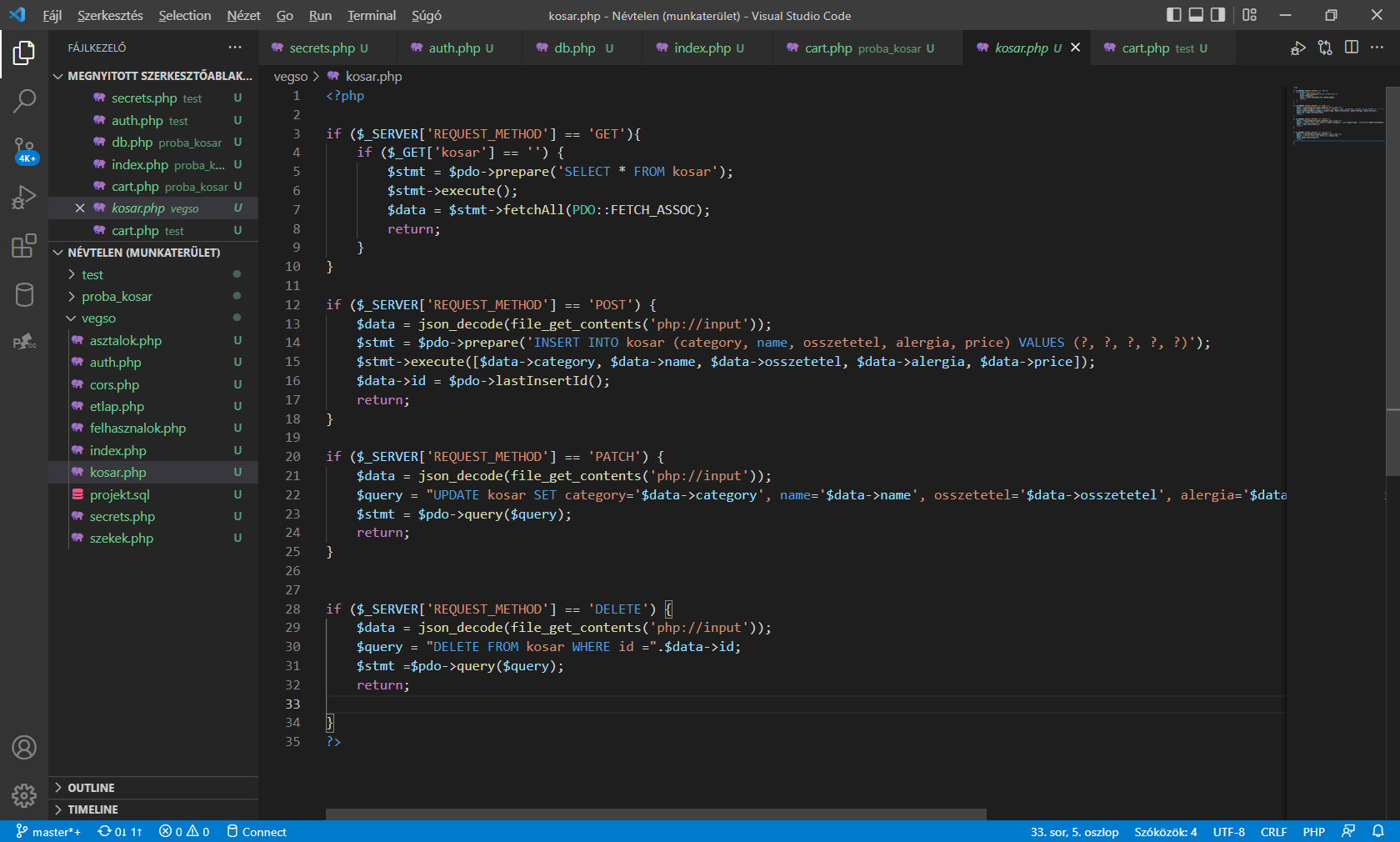
*-cors:*

Ebben a részben mondom ki, hogy mit szeretnék tenni a projektel.



*-kosar:*

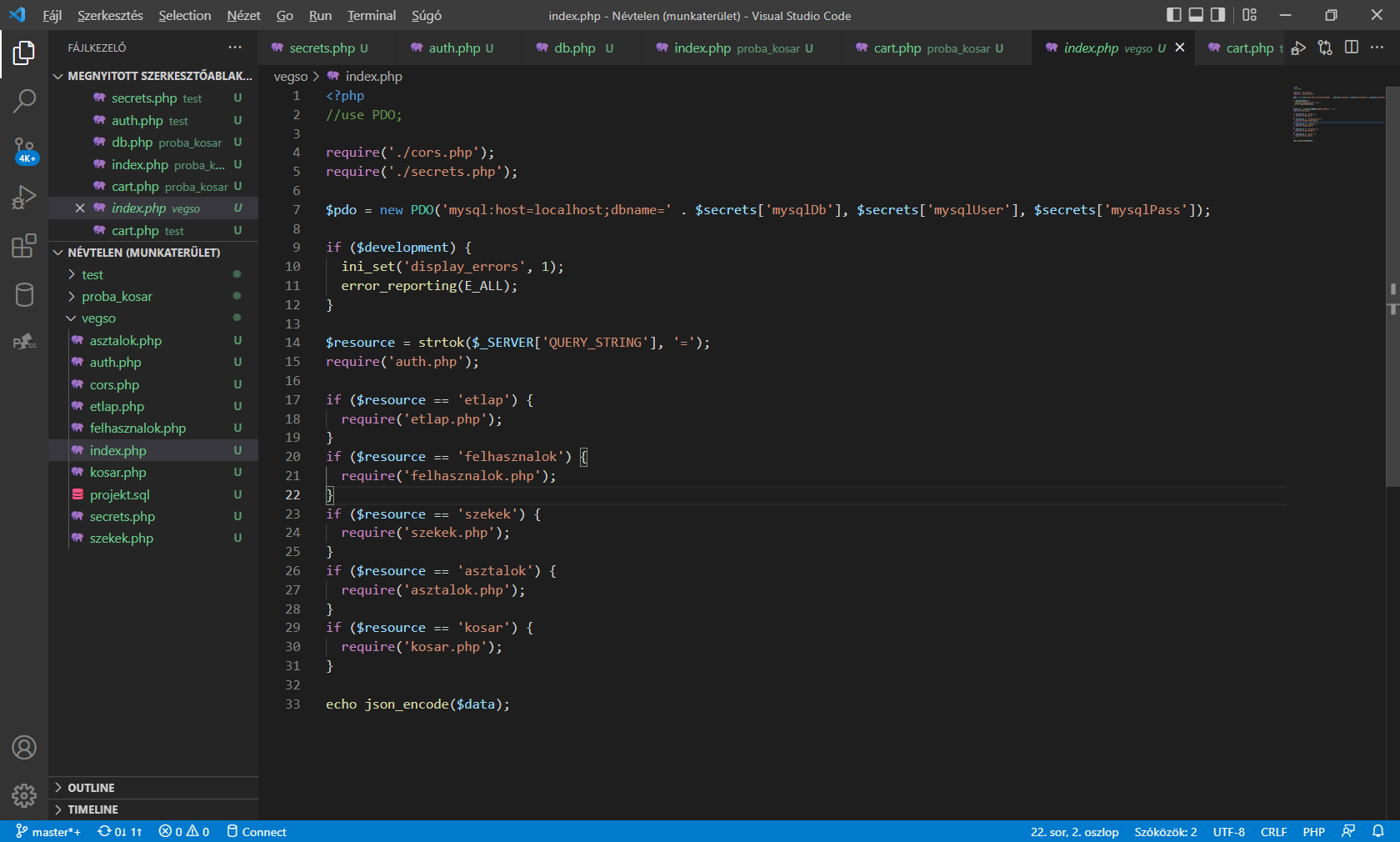
A kosár az a rész ahol a rendelések összegyűlnek. Ezzel a kóddal is lehet mindent tenni. Lekérni, Törölni. Módosítani, stb….



*-index:*

A legutolsó (de nem a „legutolsó”) programrész. Itt kapcsolódok hozzá az adatbázishoz, itt jelentem ki, hogy mi jelenjen meg és hol, innen importálom be a dolgokat. Gondolok ilyenre „kapcsolat, engedélyek, stb…”. És végül itt mondom meg, hogy Json formátum legyen.

echo json\_encode($data);



A fejlesztés viszont folyamatosan működik. Össze rakni asztalokat-székeket, esetleg képeket is hozzá rakni és még sok egyéb más funkcióval bővíteni.

***Felhasználóknak***

Ez a program vendéglátóknak készült. A papírt és a tollat/ceruzát váltja ki. Tudja, hogy a mai helyzetben milyen rohanó világ van. A vendégek rohannak, minden azonnal szeretnének megkapni. A pincér folyamatosan csak rohangál. Tehát ez a kis app ezt próbálja normalizálni. Nem igényel nagy szakértelmet. Internet van mindenkinek (99%-ban), esetleg okostelefonja, tablet-je (99%-ban). Telepíteni sem kell mert internet alapú. A programba behelyezik (szakemberek) az ön éttermének a másolatát. Asztalok, székek, étlap stb…. Önnek csak el kell indítania. Az app igen-igen egyszerű. A tablet-tet a kezébe veszi, bekapcsolja és már nézi is a felületet.

***Összefoglalás***

A programot viszonylag sikerült az elvárásoknak megfelelőnek elkészíteni. Alaposan ki kellett tanulmányozni a Backend részét (egyébként nagyon megtetszett), mivel rengeteg lehetőséget látok benne. Szeretném jobban kitanulmányozni, hogy minél jobban és erősebben legyen megírva a program. Gondolok itt a hacker-kedésre. Ami számomra nehéz volt először is megérteni, hogy mi mit csinál, miért azt csinálja, De a végére egészen jól belejöttem. Különböző kereteket fogok használni, hogy lássam, melyik lenne a legoptimálisabb, legbiztonságosabb és persze a legegyszerűbb kód. Később szeretnék egy telepítő részt is hozzá rakni, hogy kisebb legyen a hacker-kedés.

**Szerző**:

Koza Gábor