

Kodolányi János Egyetem

SZAKDOLGOZAT

MOLNÁR CSANÁD IVÁN
ÜZEMMÉRNÖK-INFORMATIKUS
ALAPKÉPZÉSI SZAK

Budapest
2023.11.20.

Kodolányi János Egyetem
Informatika Tanszék

Online Dietetikus Szakértő
Mesterséges Intelligencia segítségével
Online Dietetic Expert System with AI

Konzulens: Dr. Pitlik László

Készítette: Molnár Csanád Iván
ÜZEMMÉRNÖK-INFORMATIKUS
ALAPKÉPZÉSI SZAK

Budapest
2023.11.20.

Tartalomjegyzék

1.	Bevezetés	3
1.1.	Kiindulási helyzet	3
1.2.	Motiváció.....	3
1.3.	Rövid feladat ismertetés	4
1.4.	Célok.....	4
1.5.	Hasznosság.....	5
1.6.	A dolgozat szerkezetéről	5
1.6.1.	Feldolgozott szempontok	6
1.6.2.	Említett témák helyzete a dolgozatban.....	7
1.6.3.	Nem feldolgozott szempontok	8
2.	Szakirodalmi háttér	9
2.1.	A szakdolgozati téma és a szak tantárgyainak kapcsolata	9
2.2.	A külföldi, nem magyar nyelvű szakirodalom megjelenése a dolgozatban	15
2.3.	A magyar nyelvű szakirodalom megjelenése a dolgozatban	15
3.	Szerver	16
3.1.	Szerver felállítása/konfigurálása.....	17
3.2.	Programok telepítésének bemutatása.....	20
3.2.1.	Apache2 telepítése	21
3.2.2.	Tűzfal telepítése	22
3.2.3.	POSTGRESQL telepítése	22
3.2.4.	PHP telepítése:	24
3.2.5.	Routing megoldások	25
3.3.	Adatbázis	25
4.	Weblap működésének és megírt kódnak a bemutatása	27
4.1.	Request Handler.....	27
4.2.	Smarty template engine.....	28
4.3.	Skeleton felépítése	29
4.4.	Stíluselemek	30
4.5.	Trans Handler, kétnyelvűség megoldása.....	30
4.6.	JavaScript/Ajax funkciók	31
5.	Elemzés.....	32
5.1.	Elemzés bemutatása naiv és optimalizált módon	32
5.2.	COCO bemutatása és az adatcsere megoldásai	35

5.3.	A végtermék, miképp jelenítjük meg a kezelése után (adatvizualizáció)	36
6.	Chat GPT – dolgozat	37
6.1.	Adatgyűjtés	38
6.2.	Optimalizálás	39
6.3.	Hibaelhárítás	40
6.4.	Kód Értelmezés	40
6.5.	Újabb ötletek kitalálásához	42
7.	Vita és konklúziók	44
8.	Összegzés	44
8.1.	HU-magyar Összegzés	44
8.2.	EN-English Summary	46
9.	Mellékletek	47
9.1.	Kivonat	47
9.2.	Abstract	48
9.3.	Ábrajegyzék	49
9.4.	Hivatkozásjegyzék	51
9.5.	Rövidítések jegyzéke	55

1. Bevezetés

1.1. Kiindulási helyzet

A 2023 tavaszi félévében kezdődött, ekkor került kitalálásra a lehetőség, hogy milyen élethelyzetben lévő embereknek javasoljuk a tervezett szolgáltatást (vö. ODESSAI = Online Dietetic Expert System with AI). Több megoldás kivitelezése szóba jött az online megoldásokon keresztül a lokálisan futtatott programokig. Élelmiszer-terméknek IT-szolgáltatása, lehetett volna a fő prioritásunk pl. a valamilyen betegséggel rendelkező felhasználó, aki nem szakértő. Ilyen esetben kellett volna generálni az adatbázishoz és az ahhoz tartozó lekérdezéshez támogatásképp egy funkciót, melyet kiválasztva a felhasználó lehetőséget kap a saját betegségének beállításával, ezzel támogatva az adott betegségeket, allergiákat. (pl.: cukorbetegség, alacsony vérnyomás, magas koleszterinszint, terhesség, lisztérzékenység)

1.2. Motiváció

Az ODESSAI szolgáltatás alap képe az volt, hogy lássunk el egy humán szakértőt olyan adatbázis háttérrel, amely segítséget tud nyújtani a humán szakértő (dietetikus) számára a mindennapi munkában, a betegek ellátásában.

„A táplálkozási intervenció (Nutrition Care Process Modell, NCPM) folyamata: a táplálkozási anamnézis felvétele, dietetikai diagnózis felállítása, táplálásterápiás/táplálkozási intervenciók terv meghatározása, a táplálkozási intervenció/táplálásterápia, szükség esetén a klinikai táplálás kivitelezése, monitorozása, elemzése.” (Forrás:

http://real.mtak.hu/164865/1/Taplalasterapias_dois.pdf)

Lehetőséget biztosít továbbá a paraméterek pl. protein, zsírok kiválasztása esetén a szakértőnek, hogy az élelmiszer-termék, amit keres az ügyfele számára (jelen példában legyen sajt), az minél több zsírt tartalmazzon és minél kevesebb proteint, pontos kitöltése alapján a dietetikus felhasználónk elé áll egy optimalizált kiértékelés a sajtok csoportjában.

A megvalósítása ennek az élelmiszer-termék szolgáltatásnak motivációval töltött volt, hiszen a piacon hasonló adottságokkal rendelkező program, ami támogatna magyar nyelvezetet is, nem létezik (2023.11.01).

1.3. Rövid feladat ismertetés

Az online szakértői rendszerszolgáltatás megkövetelte, hogy egy keretrendszer vegye körül. A jelenleg is használt szerveren alapuló weblapszolgáltatás (ODESSAI), amelyen postgresql adatbázis alapon történik a meghívások jelentős része (a táblázatok html-ben kezeljük, azoknak a feltöltéséért felel a postgresql), ez lehetett volna lokálisan futtatott program, amelyet mi tervezünk meg.

Első gondolatnak felmerült a Python nyelven íródott program, amelyben a lehetőségeket ismervén a felhasználó elé kínálva pipálható, választási lehetőségeket terjesztettünk volna kiíratásra. A lokálisan való futtatás elvette volna az ODESSAI-tól a lehetőséget, miszerint bárhol használható, gyors elérésű optimalizált választ felkínáló lehetőség.

A választott programozási nyelv eldöntése azonban nem történt meg, mivel a lehetőségeket feltárva elgondolkodtunk a felhő alapú megoldáson, ezzel az ötlettel és az utána történő kereséssel bukkantunk rá az Amazon által biztosított szerverre, az AWS (Amazon Web Services, vö. 3.1. fejezet).

1.4. Célok

Meggyorsítsuk a szokványos szakkönyvi/online alapon történő kereséshez képest a mindennapos folyamatot, egy egyszerűen használható, megbízható adatbázissal rendelkezünk, amiben a tévesztés, mint például nem megfelelő adatok tárolása, létezik a táblában 0 szám érték, amennyiben az adattárban az az érték nem is szerepelne, illetve a pontosítás érdekében a tizedesjegyek kerekítését sem engedjük.

A hasznossága abban rejlik, hogy optimalizáltan kapja meg a felhasználó kérdéseire (pl. Melyik sajt az, amelyikben a fehérje mennyisége kiemelkedő, azonban kevés hamut tartalmaz) a választ, amennyiben szeretne keresni, az optimalizálás megtörténtével nem egyenesen a 2 lehetőség megvizsgálásával ér véget.

1.5. Hasznosság

A projekt alap ötlete abból indult ki, hogy az ikertestvérem, aki szintén a KJE-n végzett turisztikán, 2023 tavaszán, cukorbeteg. Szüksége volt egy olyan szolgáltatásra (vö. ODESSAI), amit fel tud használni, mint páciens.

Optimális lett volna számára egy olyan lehetőség, amely a felhasználói felületet felbővítve, olyan lehetőséget nyújtott volna, amiben ki tudja választani a betegségét, ezzel állítva a szűréshez használt adatokat, minél kevesebb szénhidrátot keresve, mivel ez a betegségének a legfontosabb prioritása (minél kevesebb annál jobb).

A szoftveren (vö. ODESSAI) belüli felhasználói beállítások arra szolgáltak, hogy bármilyen ételmiszer-termékre saját betegségét ismerve tudjon keresni a beteg és/vagy dietetikus tanácsadó, úgy hogy mind megfeleljen a korábban beállított feltételeknek.

Korai lépéseket tettünk abba az irányba, hogy a szolgáltatásfejlesztése és annak lépései feltölthetőek legyenek a tudományos forrásokkal teletöltött MIAÚ-ra (Magyar Internetes Alkalmazott/Agrár Informatikai Újság).

Ezt a lehetőséget egy hónap használat és próbálkozás után elvetettük, mivel rendszer fejlesztése és üzemeltetése szempontjából nem optimális a régi verziókat használni, illetve a piacon racionális kihívás egy új modult egy meg lévő keretrendszer korlátai között létrehozni.

Amennyiben a megadott dokumentáció alapján valaki létrehozná az ODESSAI újkori verzióját, a legfrissebb verzió számokkal mindig a legoptimálisabb dolgozni, mivel ahhoz található a legtöbb forrás és segítség, amennyiben elakadás történik.

1.6. A dolgozat szerkezetéről

A dokumentációban található hivatkozások, amelyeknél külön megjelölésre került a pontos helyzete, amelyet Ódor Márk dolgozatában foglalnak el, pontosan összefüggnek, mivel a felépítés ketté szedése megtörtént az ODESSAI kitalálásakor.

Továbbá fontos megemlíteni, hogy szerzői döntést követően szövegközi hivatkozások kerültek elhelyezésre, hiszen a gyors elérésű adatok a dolgozat megírásánál és reprodukálhatóságánál fontos szerepet játszanak. Az összegzésben elsődlegesen a felhasználó orientáltság szempontjából került összefoglalásra a dolgozat, nem pedig technológia orientáltan, mert az röviden nehezen összefoglalható, míg a felhasználóbarátság tetszőlegesen röviden hozható létre. Én egy olyan hivatkozásjegyzéket hoztam létre, ahol az ugrópont van, elől, mert a hipertextes navigációt ez támogatja elsődlegesen, minden más leíró adat, amelyből nem

következik a továbbiakban kattintás, tehát az olvasót úgy orientálok, hogy elsődlegesen a tovább lépésre sarkallom.

1.6.1. Feldolgozott szempontok

A dolgozat tartalmazza az alábbi szempontokat, folyamatokat, melyeken a készítése alatt szó esett, illetve megvalósításra került.

- Az Amazon által kínált lehetőségek bemutatása, a szerverünk kiválasztása
- Egy webszerver felépítése és a funkcióinak ellátása, MIAÚ-hoz hasonlóság előállítása (Magyar Internetes Alkalmazott/Agrár Informatikai Újság) (URL: <https://miau.my-x.hu/miau2009/index.php3>).
- A projekthez szükséges programok telepítése (Composer, Smarty, PHP, Apache2, Winscp és annak beállításai, Pgadmin4)
- A megfelelő adattár megtalálása (adatbázisok és adatbázis kezelő programok)
- Háttértartalom optimalizálása, SQL (Structured Query Language) funkciók és Pgadmin4 bemutatása
- A szerveren futtatott weblap titkosításának megoldása és hozzá tartozó hiteles tanúsítvány
- Anti-diszkriminatív modellezése egy adattárnak, optimalizált kimutatás
- 2 nyelvűség támogatása (mind weblapon, mind pedig az adattáblában)
- A lekérdezések limitációi (ki legyen a felhasználó kérdés)
- A leírt kódnak fontosabb részei kiemelve, magyarázva (ebbe tartoznak: „Trans Handler”, „Request Handler”, Routing megoldások, Smarty Template Engine)
- ChatGPT a dolgozat viszonyában
- Adatvizualizáció megjelenése a dolgozatban, folyamatára a kiértékelés folyamatáról és annak bemutatása
- Szakirodalmi források, megjelenések a dolgozatban
- COCO (Component-based Object Comparison for Objectivity) Y0 modellező algoritmus rövid bemutatása, ismertetése

1.6.2. Említett témák helyzete a dolgozatban

A szolgáltatás kitalálásához felderített szakirodalom megjelenése a dolgozat részleteiben a 2. fejezetben olvasható, együtt a tantárgyi megfelelésekkel, amelyek a dolgozat kapcsán előkerültek.

A dolgozat tartalmazza a környezet felépítését, amelyben az egész szoftver megvalósult. Ennek további lehetőségeiről és a téma kapcsán előforduló kérdésekről olvashat a 3. fejezetben.

A 3.1-es alfejezet részletezi a szerver felépítését alapjaiban.

A telepítési lépések pontos leírása és azok a futó programok, amelyeket használ a szolgáltatás, elérhetőek a 3. fejezet, 2. alfejezetének alfejezeteiben (3.2.1 - 3.2.5).

Az adatbázis megszerzése, részletességének problémái, alternatívák és implementációs megoldásai, olvashatóak a 4. fejezetben.

A weblapon megjelenő kód és azon belüli funkciók ellátása pontosabb bemutatásra kerül, a kétnyelvűséget támogató funkciót ellátó programkóddal együtt (PHP, Smarty, JavaScript/ Ajax, Html, MDBootstrap) a 4. fejezet, 1.-től 6.-ig alfejezeteiben.

Az elemzési, kiértékelési folyamat és benne felmerülő kérdések, olvashatóak az 5. fejezet 1. alfejezetében.

Az ODESSAI során segédkezet nyújtó mesterséges intelligencia bemutatása (COCO-Component-based Object Comparison for Objectivity) és a megoldása az állományküldésnek és kezelésnek olvasható az 5. fejezet 2. alfejezetében.

Az adatvizualizáció, illetve a végeredménye a kiértékelésnek és az ehhez tartozó formaságok olvashatóak az 5. fejezet 3. alfejezetében.

A dolgozat és a ChatGPT közreműködése benne, további részekre van bontva, példákkal ellátva, hogy mire is lehetett használni az ODESSAI létrehozásakor. Olvasható a 6. fejezet, 1.-től 5.-ig alfejezeteiben.

A kétnyelvű összegzés olvasható (magyar és angol nyelven) a 7. fejezet, 1. és 2. alfejezetében.

Az ábrák jegyzéke a 8. fejezetben található.

A források megjelölése a 9. fejezetben található.

A felhasznált tartalmak elérési útvonalai megtalálhatóak a 10. fejezetben, együtt a kivonattal a dolgozatról.

1.6.3. Nem feldolgozott szempontok

A dolgozat nem tér ki bővebben a következő dolgokra, mivel az idő és a dolgozat terjedelmi korlátai limitáló tényezők voltak, így választani kellett, hogy mi férjen bele ebbe a keretbe. Az alábbi felsorolás azokat a gondolatokat tartalmazza, amely számításba került, majd ezt követően mérlegelve az idő és a befektetett energia mennyiségét, elvetődött. Ez által, csak röviden lesz a felsorolásban szereplő témákra kitérés a dolgozatban.

- További adattárak használata, azokra való beállítása az optimalizáláshoz
- A másik szempont támogatása, miszerint nem a dietetikus kezébe adnánk a terméket, hanem bárki által használható volna (ennek a limitációit meg kell szabni, a felhasználót meghatározni)
- Optimalizálni a lekérdezési lehetőségeket betegségekre (pl.: cukorbetegség, alacsony vérnyomás, magas koleszterinszint, terhesség, lisztérzékenység)
- PHP oldalon, a Laravel használata felmerült (egy magasabb szinten használt PHP keretrendszer, amelyben a teljes projektet lehetett volna megépíteni), azonban a szakértés ehhez szükségeltetett, amely nem volt meg, ennek a megtanulása időigényes lett volna, ezért nem ezt választottuk
- Étlap és diéta összerakó funkció
- Készételek elemzése, szétbontása étlap alapján
- Árral ellátása a termékeknek (vö. Ár/teljesítmény-elemzés)
- Testi adottságok beállítása és mentése
- Bejelentkezési lehetőség és adatok mentése profilonként
- Szakértő támogatás kérése, szakvéleménnyel ellátása és folyamatos együttműködés

2. Szakirodalmi háttér

A szakirodalmak megjelölés és a tantárgyakhoz való kapcsolata feltüntetésre kerül az első említésnél és magyarázatot kap, hogy miként volt rá szükség a dolgozat szempontjából.

(Sorrend: forrás, szerző, kiadás ideje, kiadási intézmény)

2.1. A szakdolgozati téma és a szak tantárgyainak kapcsolata

A Kodolányi János Egyetem által biztosított tantárgyak megjelenése a dolgozatban olvasható az alábbi pontokban. A szolgáltatás kitalálásánál felhasznált információk és idézetek is a felsorolásban olvashatóak.

A tantárgyi megfelelőként használt Idézetek és azokhoz tartozó dokumentumok, felmutathatók egyaránt a magyar és nem magyar nyelvű források témájában. A redundanciát kikerülve, hogy a dolgozat olvasó szempontjából ne tűnjön egyhangúnak, a tantárgyaknál megjelölt idézetek és hozzá tartozó források nem kerültek megemlítésre a magyar és nem magyar szakirodalom megjelenése pontban.

- **Algoritmusok és adatszerkezetek** megjelenése az Ajax funkciók által ellátott kiválasztásakor történik, az SQL adatbázis számára pontos query létrehozása a PHP fájlokban történik (get_items.php, query_database.php, selections.php, selections2.php). *„Az algoritmusokat az emberiség már jóval azelőtt ismerte és használta, mielőtt az őket végrehajtani tudó számítógépeket megépítette”* (Forrás: http://www.informatom.hu/sze/01/LGB_SZ001/Algoritmusok_%C3%A9s_adatstrukt%C3%BAr%C3%A1k.pdf, Pusztai Pál, 2006, Széchenyi István Egyetem).
- **A hálózatok és számítógép architektúrák** tantárgy megfeleltetése a dolgozatban az Amazon szerver felállítása és a hozzá tartozó beállításokban található meg. *„Ma egyre többen tapasztaljuk meg azt, hogy a számítógép-hálózatok megjelennek a társadalom szinte minden szférájában. Nagyon fontos és reméljük megbízható építő kövei lesznek az **"Információs Szupersztrádák"**-nak és segítik a hatékony **"Információs Társadalom"** építését. Továbbá megtapasztalhatjuk a felelős közösségi munka szükségességét. Ez talán lehet egy új szín a viszonylag hosszú hazai **"PC korszak"** atomizáltsága után az együttműködésben.”* (Forrás: <https://www.inf.u-szeged.hu/~bohus/oktseg.html>, Dr. Bohus Mihály, 1995, Szegedi Egyetem).

- A **programozási alapelvek és módszertanok** előkerülnek a következő programozási nyelvezetekben, amelyek szerver részét képezték a szolgáltatásnak PHP, JavaScript, Smarty, Html kódok megjelenése. *„Egy univerzális programozási nyelv ismeretében algoritmusaink programmá írhatók, kipróbálhatók, hiszen a „visszacsatolás”, a megoldás helyességének számítógéppel történő ellenőrzése a tanulási folyamat szerves része* ” (Forrás: http://www.informatom.hu/sze/01/LGB_SZ001/Algoritmusok_%C3%A9s_adatstrukt%C3%BAr%C3%A1k.pdf, Pusztai Pál 2006, Széchenyi István Egyetem).
- **Rendszermodellezés** megjelenése fedezhető fel a szolgáltatás kitalálásának idején, felépítésének alapjaiban, előre tervezéseknél a fejlesztésekkel kapcsolatban és a célcsoport meghatározásánál. *”Mind a természetben, mind az ember alkotta rendszerekben fellelhetők bizonyos szabályszerűségek: egyesek a rendszer elemei közötti kapcsolatokat, míg mások magukat az elemeket jellemzik. Bár sok mindenben eltér egy közlekedési hálózat, egy számítógép-hálózat, egy épület vagy egy város felépítése, a strukturális modellezés eszköztárával olyan „nyelvet” kapunk, amivel ezeket hasonlóan modellezhetjük”* (Forrás: <https://ftsrg.mit.bme.hu/remo-jegyzet/rendszermodellezes.pdf>, Bergmann Gábor, Darvas Dániel, Molnár Vince, Szárnyas Gábor, Tóth Tamás, 2023.06.02, Budapesti Műszaki és Gazdaságtudományi Egyetem).
- A **felhasználói interfészek és vizualizáció** tantárgy megjelenése a dolgozatban a programkódban megjelenő CSS és Html elemek, amelyek a képét adják a felhasználónak a végtermék formájában. A projekt megtervezésekor figyelembe vettük a következő idézetet: *” Az emberek általában 7 információs egységet tudnak fejben tartani. Ha ennél többet ajánlunk fel, akkor nagyobb eséllyel vétének hibát”* (Forrás: https://dcs.uni-pannon.hu/files/docs/users/csertangyorgy/swtech/16_UI.pdf, Dr. Csertán György, Ismeretlen keletkezési idejű, Pannon Egyetem).
- **Rendszertervezésnek** tekinthető, rendszermodellezés tantárggyal összefüggően, az ODESSAI megtervezésekor felmerülő gondolat, amely az előre gondolkodást jelentette, hiszen meg lett tervezve a teljes program, mielőtt bármilyen kódsor került volna leírásra. *”Ha egy üzleti vállalkozás fő vonásait akarjuk szemügyre venni, annak sajátosságaira való tekintettel, akkor a küldetés, a célok és fő célkitűzések, valamint az üzleti környezet lesznek számunkra az elsődlegesek.”* (Forrás: <http://mek.oszk.hu/14300/14321/pdf/14321.pdf>, Harsáczki András, Nagy Dénes, 2013,

Eszterházy Károly Főiskola). Az alapvető létrehozási elvek szempontjai közé tartoztak a dokumentumban található alapelvek.

- **Informatikai védelem és biztonság:** SQL injection védelem, titkosítás megoldásai, belépési lehetőség a weblapon (felhasználónév/jelszó használata). „*A biztonság tehát a minőség és a megbízhatóság mellett a harmadik olyan követelmény, amelyet figyelembe kell venni a hosszútávú működés fenntartása szempontjából*” (Forrás: https://nki.gov.hu/wp-content/uploads/2019/03/NJSZT_IT_Biztonsag_kozerthetoen_v3.pdf, Erdősi Péter Máté, Solymos Ákos, 2018, Neumann János Számítógép-tudományi Társaság).
- **Adatbázisok 1 és 2** tantárgy elemei felfedezhetők a különböző felsorolt adatbázisok keresésénél és a megfelelő megtalálásánál, illetve annak az elemzésénél. „*Ma, amikor az információ és a tudás a legfontosabb erőforrássá kezd válni, különösen felértékelődik az a képesség, amely (nagy) adatbázisok tervezésére, megvalósítására és/vagy hatékony lekérdezésére tesz valakit alkalmassá.*” (Forrás: <https://db.bme.hu/~gajdos/Adatbazisok2019.pdf>, Gajdos Sándor, 2019, Budapesti Műszaki és Gazdaságtudományi Egyetem).
- A **Machine Learning és mesterséges intelligencia** tantárgyak megjelenése a mesterséges intelligencia használatán alapult különböző módokon kérések, ötletszerzés, megoldáskeresés, problémafeltárás. „*Over the past two decades Machine Learning has become one of the main stays of information technology and with that, a rather central, albeit usually hidden, part of our life. With the ever increasing amounts of data becoming available there is good reason to believe that smart data analysis will become even more pervasive as a necessary ingredient for technological progress.*” A Machine Learning fontosságát bemutató dokumentumból ötletet merítettem a szolgáltatás kialakításához (Forrás: <https://alex.smola.org/drafts/thebook.pdf>, Alex Smola, S.V.N. Vishwanathan, 2008, Cambridge University/Cambridge Egyetem)
- **Operációs rendszerek** tantárgy részleteibe menően fellelhető a szerveren megtalálható futó operációs rendszer témájában, telepítési lépéseiben. „*Egy jó módszer az operációs rendszer definiálására, ha megragadjuk kettős természetét: az operációs rendszer egyrészt virtuális gép, másrészt erőforrás menedzser. (Néha így definiálják: válaszoló gép)*”.(Forrás: <https://users.iit.uni-miskolc.hu/~vadasz/GEIAL302B/GEIAL202-Operacios-rendszerek-jegyzet.pdf>, Dr. Vadász Dénes, 2002, Miskolci Egyetem).

- **A programozás 1, 2 és 3** tantárgy a PHP, JavaScript, Smarty és Html során került elő, továbbá fellelhető még a táblázatok előkészítő nézetében, amely a kódolást megelőzte, hogy megbizonyosodjunk arról, hogy annak, amit csinálunk, van értelme. (Forrás: <https://m2.mtmt.hu/gui2/?mode=browse¶ms=publication;34071055>, Rikk János, Pitlik László 2023, Kodolányi János Egyetem)
- **Szoftverüzemeltetés** megjelenik a szolgáltatás fenntartásánál és fejlesztésénél.
- *„Tesztelésre azért van szükség, hogy a szoftver termékben meglévő hibákat még az üzembe helyezés előtt megtaláljuk, ezzel növeljük a termék minőségét, megbízhatóságát.”* (Forrás: <https://aries.ektf.hu/~gkusper/SzoftverTeszteles.pdf>, Ficsor Lajos, Dr. Kovács László, Krizsán Zoltán, Dr. Kusper Gábor, ismeretlen keletkezési idejű, Eszterházy Károly Katolikus Egyetem) A **Szoftvertesztelés** megtörtént, mivel a szolgáltatás, annak létrehozása előtt, amikor még csak a koncepció volt készen, felkerestem a testvéremet, aki cukorbetegséggel küzd és felkértem, hogy tesztelje le, hogy vajon a szolgáltatásnak van-e létjogosultsága.
- *„Szoftver architektúrának nevezzük a szoftver fejlesztése során meghozott elsődleges tervezési döntések halmazát”* (Forrás: https://people.inf.elte.hu/groberto/elte_kasz/eloadas_anyagok/elte_kasz_ea08_eml.pdf, Giachetta Roberto, Ismeretlen keletkezési idejű, Eötvös Loránd Tudományegyetem) **Szoftverarchitektúrák** tantárgy részletekbe menően megtalálható a dolgozatban, a szerver felépítésében és azt követő, a szolgáltatás felállításához szükséges lépésekben.
- *„A projekt szó a XXI. század egyik legalapvetőbb kifejezése, használata a köznyelvben csaknem olyan gyakori, mint a gazdaság és a menedzsment szakterületein. Azonban felmerül a kérdés – vajon minden esetben igazunk van-e, amikor azt mondjuk: „... belefogtam egy új projektbe...”?”* (Forrás: https://innovativ-tudasvaros-efop361.unimiskolc.hu/files/1043/Projektmenedzsment_tananyag.pdf, Tóthné Kiss Anett, 2018, Miskolci Egyetem) A **fejlesztések minőség- és projektmenedzsmentje** tantárgy elemei fellelhetőek a dolgozat teljes tartalmában, hiszen több mint egy féléven keresztül átívelő szolgáltatásnak a létrehozásáról szól és szorosan egybefügg a manapság használatos „projekt” szó használatával, amelyet a dolgozat folyamán többször alkalmaztam.

- „Az infokommunikációs technológiák változása új távlatokat nyit meg az állampolgárok és a kormányzati szektor közötti kommunikációban, együttműködésben, az információ és a tudásmegosztás területein; lehetőséget adva az állampolgári visszajelzésekre, a véleménynyilvánításra, ezen keresztül erősítve a részvételt a szakpolitikák tervezésében és a döntéshozatali folyamatokban.” (Forrás: https://nkerepo.uni-nke.hu/xmlui/bitstream/handle/123456789/14286/Innovativ_infokommunikacios_tech_nikak_az_elektronikus_kormanyzati_megoldasokban_2019.pdf;jsessionid=43F449A4B6738A469510D7C86432DE6B?sequence=1, Kő Andrea, 2018, Nemzeti Közszerződési egyetem). Az **innovatív és kommunikációs technológiák** megjelenése a dokumentumban a projekt alap ötletében is, emellett a projekt teljes egészére kiterjedően fellelhető, hiszen egy, jelenleg a piacon nem létező szolgáltatást hozunk létre az ODESSAI-val és a legújabb technológiák kerültek felhasználásra, mint például a mesterséges intelligencia, a chatGPT és a chatPDF. Ezen felül a szolgáltatás amelyet nyújtunk a humán szakértő és a páciense közötti kommunikációt segíti és optimalizálja, gyorsítja fel.
- „Az emberi elme produktumainak összessége, a kultúra – a specifikusan emberi funkcióknak, az emlékezetnek és a nyelvnek köszönhetően – állandó kölcsönhatásban áll az elmével, az egyes emberi énnel, valójában képes formálni, alakítani azt.” (Forrás: <https://mek.oszk.hu/08200/08200/08200.pdf>, Szirmai Éva, Újvári Edit, 2009, Szegedi Tudományegyetem) A **komplex társadalomtudományi ismeretek, az emberi viselkedés és kommunikáció és az emberi civilizáció és identitás** tantárgy az Európában már jelen lévő gondolatot támogatva jelenik meg, miszerint egy „jó” diéta sokat számít egy ember életében és egészségében, összefüggésben van az elhízással és a betegségekkel egyaránt. Ezért fontos kiemelni, hogy a szolgáltatás megtervezésénél fontos szerepet játszott a realitás, miszerint a válasz amelyet egy ehhez hasonló szolgáltatástól várna az ember, befolyásolhatja a mentális és fizikai egészségét is, mind jó és rossz irányba is.
- „A szükséglet és az igény szoros kapcsolatban van egymással. A fogyasztói igény jellemzője, hogy a gazdaság szereplői nem a saját szervezetükön belüli munkával és nem közösségi intézményeken keresztül elégítik ki, hanem a piacon keresztül. Az igényt magát többféleképpen tudjuk csoportosítani, például mennyiség vagy jelleg szerint.” (Forrás: <https://edit.elte.hu/xmlui/bitstream/handle/10831/85759/V%C3%A1llalati%20Gazdas>

[%C3%A1gtan_Holl%C3%B3sy_Kiss-Jegyzet_.pdf?sequence=1](#), Hollósy Zsolt, Kiss Livia Benita, 2023, Eötvös Lóránd Tudományegyetem). **A vezetési és vállalkozási ismeretek, illetve a vállalatgazdaságtan** megjelenik a szolgáltatás létrehozásakor kitalált tervvel együtt, hiszen ezt egy piacképes termékként fejlesztettük Ódor Márk hallgatótársammal és ahhoz, hogy ez szalonképesse váljon szükség volt ezekre az ismeretekre és a piaci rést kihasználva létrehoztuk az igényeknek megfelelő szolgáltatást.

- **Az elektronikus áramkörök és az elektronika fizikai alapjai** tantárgyak összefüggésbe köthetők az említett korábbi feladattal, amiben a Raspberry Pi konfigurációjáról tanultam és a tudást, amit akkor szereztem, most felhasználhattam. Megismertem továbbá a hozzá tartozó elektronika részt is, közelebb kerültem a nyákok és az azon történő forrasztásokhoz, kábelezésekhez.
- *„A természet világa nem csupán létezők és erők pállott kotyvaléka, hanem csodálatosan leleményes, egységes matematikai rendszer.”* (Forrás: https://www.citatum.hu/konyv/Isten_gondolatai, Paul Davies, 1983-magyar fordítás 2000, Arizona State University/Arizonai Egyetem). **A matematika**, amelyet az egyetemen tanultunk segítséget nyújtott egy ilyen összetett feladat létrehozásában és az abban történő számolások, programozás létrehozásában.
- *„A biztonság tehát a minőség és a megbízhatóság mellett a harmadik olyan követelmény, amelyet figyelembe kell venni a hosszútávú működés fenntartása szempontjából”.* **A szakterületi jogi ismeretek és az IT biztonság tantárgy, az informatikai védelem és biztonság:** megismerése és a dolgozatban előkerülése a biztonsági réteg létrehozásakor került elő, amikor a felhasználó adatairól beszéltem és az abból származó veszélyekről, továbbá a hackelés lehetőségének kikerülése is lényeges volt, ezért nem található meg a weblapon külön lehetőség adatot bevinni kézzel (SQL injection). Mivel a szakterület kiterjed az élelmiszeriparra, a dietetikusokra és az informatikára ezért a tantárgy lényeges szerepet játszott a dolgozat létrehozásában. (Forrás: https://nki.gov.hu/wp-content/uploads/2019/03/NJSZT_IT_Biztonsag_kozerthetoen_v3.pdf, Erdősi Péter Máté, Solymos Ákos, 2018, Neumann János Számítógép-tudományi Társaság).

2.2. A külföldi, nem magyar nyelvű szakirodalom megjelenése a dolgozatban

<https://extension.psu.edu/selecting-cheese-for-health> (Selecting Cheese for Health, 2018, Szerző(k): Stacy Reed, Pennsylvanai Egyetem, angol nyelvű forrás, 2023.11.17) → Segítséget nyújtott az alábbi Stacy Reed által írt forrás, specifikusan a sajtokra szűrt keresés során, amely felmerült az adatbázisok kapcsán. A nézet amely megalapozta a projektet. Részletességében alul múlta a szükséges mennyiséget, viszont a megjelenése és az adatok rendszerezése példa értékű volt.

<https://www.bda.uk.com/about-dietetics/what-is-dietitian/dietitian-or-nutritionist.html>

(Dietitian or nutritionist?, 2023, Szerző(k): domain mögötti tulajdonos/szerkesztő, angol nyelvű forrás, 2023.11.17). A dietetikus, humán szakértő alapvető bemutatása és a munkakörének leírása innen származik a dolgozatban, egyfajta munkaköri leírást adott hozzá a forrás, amely körvonalazta a szakmaiságát a szolgáltatásunk célcsoportjának.

<https://doi.org/10.1016/j.tifs.2016.08.005> (Role of food processing in food and nutrition security, 2016.10.01, Szerző(k): Mary Ann Augustin, angol nyelvű forrás, 2023.11.17). Élelmiszerbiztonság szempontjai kerültek elő a szolgáltatás kapcsán és jövő képek felmutatásával bizonyította Mary Ann Augustin a feldolgozott ételekről.

Kihívásnak tekinti az írásában a megfelelő táplálkozásra törekedést és annak nehézségeit figyelembe véve próbál megoldást találni a problémára, ennek a mintájára és megoldására készült az ODESSAI.

2.3. A magyar nyelvű szakirodalom megjelenése a dolgozatban

<https://miau.my-x.hu/miau/10/tqm22.htm> (A minőség fogalma az élelmiszeriparban, 1999, Szerző(k): Lorbert Zsuzsanna, magyar nyelvű forrás, 2023.11.17). A termékek minőségileg megfelelő kialakítását szereztük ebből a forrásból. Szó van az élvezeti értékről benne, amely a táplálkozás megfelelő arányaira tér ki. Az élelmezés-egészségügyi biztonságáról részről kivett fontos gondolat a „*az élelmiszerek a fogyasztó egészségét nem károsíthatják*”.

<https://www.rackhost.hu/tudasbazis/informatikai-alapok/hogyan-lehet-egy-jelszot-feltorni/>

(Hogyan lehet egy jelszót feltörni?, 2011, Szerző(k): a domain mögötti tulajdonos/szerkesztő, magyar nyelvű forrás, 2023.11.17). Az SQL injection és az etikus hackelés bemutatása a fontos

fókuszra ennek a dokumentumnak, ebből kiemelten fontos megemlíteni információ biztonság tekintetéből az SQL injection-t, hiszen bármely adat manipulálása és azok bármely módon lekérdezése, módosítása megtörténhet, ha ez az eshetőség megtörténik.

http://real.mtak.hu/164865/1/Taplalasterapias_dois.pdf (TáplálásTerápiás konszenzus 2022, 2023.01. Szerző(k): Dr. Pálfi Erzsébet, Dr. Molnár Andrea, Kovács Andrea, Havasi Anikó, Szűcs Zsuzsanna, magyar nyelvű forrás, 2023.11.17). A dietetikus, humán szakértő további körvonalazása történik ebben a dokumentumban, védelmezi a végzettséggel rendelkező, a szakmában dolgozó szakembert. A dietetikus működését bemutatja, ezzel segítve a szolgáltatás képének kialakulását.

„A dietetika fő feladata, célja az egészséges és beteg emberek segítése, táplálkozással kapcsolatos állapotok gyógyítása, kezelése, az egészségnevelés, valamint a táplálkozással és életmóddal összefüggő betegségek megelőzése”. (Forrás:

https://www.globemedicalcenter.hu/dietetika/?gclid=Cj0KCQiApOyqBhDlARIsAGfnyMrNWZejMaRqxroVt8Lvb-LZqXQwiQll50z4XS7Uog2H7yrRL5-4WuAaAkTsEALw_wcB,

Szerző(k): Pongrácz Adrienn, 2021). Az idézet meghatározza az értékrendjét a dietetikának, továbbá körül írja, hogy mivel is foglalkozik a tudományág. Ennek a felhasználásával történt a szolgáltatásunk kialakulása.

„Az emelkedő élelmiszer-infláció nemcsak az általános inflációt növeli, hanem bizonytalanságot is teremt, ami nagyobb inflációs várakozásokhoz vezet”,(Forrás: <https://doi.org/10.20311/stat2023.04.hu0354>, Bareith Tibor, Fertő Imre 2023.04.12). Eme gondolat került elő amikor az élelmiszer termékek ár-érték arányát vettük figyelembe, felfedezhető volt, hogy az adatok évről évre exponenciálisan növekednek, amikor az értékükről beszélünk.

3. Szerver

A saját munka több önálló fejezetre tagolódik (vö. 3.1-3.2-3.3).

Elsőként az ODESSAI működéséhez szükséges hardver-alapok kialakítása kerül bemutatásra (vö. 3. fejezet):

Minden, ami az alábbi fejezetekben olvasható, amennyiben ez nincsen feltüntetve külön, egy időben is történhetett volna, párhuzamosságra képes a haladás szempontjából.

A prioritási lista felállítása a saját szempontok és haladás kialakulása alapján történt.

3.1. Szerver felállítása/konfigurálása

Itt kezdésképp elindultam a virtualbox-apache szerver irányába az elején, hiszen ebben a témában már korábbi tantárgyak esetében történt tapasztalatszerzés, azonban ennek megvoltak a hátrányai, többek között, hogy nem tudtam rendszeresen futtatni, ezért nem biztosított teljes elérést.

Akadály volt továbbá az elérhetőség változása minden indításnál, mivel a helyi hálózat nem engedélyezte a konfigurációját ennek.

Ezen tapasztalatokat követően döntöttem úgy, hogy az Amazon által biztosított interneten elérhető szervert fogjuk alkalmazni, amely a Linux alapú Debian operációs rendszerrel van ellátva.

Rendkívül sok lehetőség lett volna ár kategóriában az 1\$/óra bérezett szervertől, egészen az ingyenes kivitelezésig létezik minden az Amazonnál.

Értelemszerűen a felhasználó minél többet fektet ebbe bele annál nagyobb erőforrásokkal lesz ellátva.

Összesen ilyen szolgáltatás típusból 345 fajtát kínálnak fel (forrás: <https://aws.amazon.com/ec2/instance-types/>), melyeket csoportokba sorolnak teljesítmény (több pénz=nagyobb teljesítmény), ebből a csoportosításból választhat a felhasználó különböző szempontok szerint, van nagy tárhely kevés teljesítménnyel, vagy nagy számítási kapacitással bíró, kevés tárhellyel rendelkező is.

Ebből kiválasztásra került a t3 (forrás: <https://aws.amazon.com/ec2/instance-types/t3/>) csoport (t=burstable performance / robbanásszerű teljesítményt jelent a t), azon belül is az alsóbb pénzügyi osztályba tartozó lehetőségek közül választottunk. Megvizsgálva a feltételeket, hogy valaki ilyet használjon, kínálnak egy kedvező lehetőséget, miszerint 1 évig fizetés nélkül is használható, biztosítanak számunkra 750 óra elérést egy hónapra, minden hónapban újakezdve a számlálást. (Forrás: https://aws.amazon.com/free/?all-free-tier.sort-by=item.additionalFields.SortRank&all-free-tier.sort-order=asc&awsf.Free%20Tier%20Types=*all&awsf.Free%20Tier%20Categories=*all)

Típusát tekintve t3.micro (vö. 1. ábra): 1 gigabyte memória, 8 gigabyte fűzőhellyel rendelkező, amennyiben a határérték meghatározásokat betartja a felhasználó, tehát nincsen nagyobb adatmozgás rajta naponta mint 10 gigabyte, addig ingyenes hozzáférésű, a nap minden pontján elérhetőséget biztosító szerver. (lásd 1. ábra).

☐	Instance type ▾	vCPUs ▾	Architecture ▾	Memory (GiB) ▾	Storage (GB) ▾	Storage type ▾	Network performance ▾	On-Demand Windows pricing ▾	On-Demand Linux pricing
☐	t3.nano	2	x86_64	0.5	-	-	Up to 5 Gigabit	0.01 USD per Hour	0.0054 USD per Hour
☐	t3.micro	2	x86_64	1	-	-	Up to 5 Gigabit	0.02 USD per Hour	0.0108 USD per Hour
☐	t3.small	2	x86_64	2	-	-	Up to 5 Gigabit	0.04 USD per Hour	0.0216 USD per Hour
☐	t3.medium	2	x86_64	4	-	-	Up to 5 Gigabit	0.0616 USD per Hour	0.0432 USD per Hour
☐	t3.large	2	x86_64	8	-	-	Up to 5 Gigabit	0.114 USD per Hour	0.0864 USD per Hour
☐	t3.xlarge	4	x86_64	16	-	-	Up to 5 Gigabit	0.2464 USD per Hour	0.1728 USD per Hour
☐	t3.2xlarge	8	x86_64	32	-	-	Up to 5 Gigabit	0.4928 USD per Hour	0.3456 USD per Hour

(1. ábra: Futtatható szerver lehetőségek a T3-as csoportban az Amazonnál, Forrás: <https://eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#InstanceTypes>)

Erre a szerverre Putty-on (Putty=szabad és nyílt forráskódú terminálemulátor Forrás: <https://www.putty.org/>) keresztül SSH (Secure Shell, biztosított kapcsolat 2 elérési pont között, TCP 22-es port) segítségével lehetett kapcsolódni egy kulcs használatának segítségével, amely beengedte a felhasználót (ez a szerver által kibocsátott titkosításának kulcsa volt). Innentől kezdve a kapcsolat létrejött a szerverrel, el lehetett kezdeni a konfigurációját a szervernek. Külső elérés biztosítása érdekében ki kellett nyitni a 22-es, a 80-as és a 443-as látható portokat, amelyeken elérhetővé vált a szerver az interneten (vö. 2. ábra).

Instance summary for i-0f613e8bd9ab97135 (miau-clone) [Info](#)

Updated less than a minute ago

Refresh

Connect

Instance state ▾

Actions ▾

Instance ID

i-0f613e8bd9ab97135 (miau-clone)

Public IPv4 address

13.51.177.80 [Open address](#)

Private IPv4 addresses

172.31.37.185

IPv6 address

–

Instance state

Running

Public IPv4 DNS

ec2-13-51-177-80.eu-north-1.compute.amazonaws.com [Open address](#)

Hostname type

IP name: ip-172-31-37-185.eu-north-1.compute.internal

Private IP DNS name (IPv4 only)

ip-172-31-37-185.eu-north-1.compute.internal

Elastic IP addresses

–

Answer private resource DNS name

IPv4 (A)

Auto-assigned IP address

13.51.177.80 [Public IP]

AWS Compute Optimizer finding

Opt-in to AWS Compute Optimizer for recommendations. [Learn more](#)

IAM Role

–

VPC ID

vpc-0db12f68da17034c5 [Open](#)

Auto Scaling Group name

–

Subnet ID

subnet-0c2532780680990b3 [Open](#)

Launch time

Sun Jul 30 2023 12:55:20 GMT+0200 (közép-európai nyári idő)

Details

Security

Networking

Storage

Status checks

Monitoring

Tags

▼ Security details

IAM Role

–

Owner ID

508243260110

Security groups

sg-00434d76cfadc1a74 (launch-wizard-1)

▼ Inbound rules

Filter rules

1

>

Name	Security group rule ID	Port range	Protocol	Source	Security groups	Description
–	sgr-0686b77c3499193fb	443	TCP	0.0.0.0/0	launch-wizard-1	–
–	sgr-025cc7cbad43fa17a	80	TCP	0.0.0.0/0	launch-wizard-1	–
–	sgr-08f6c9256ab590365	22	TCP	0.0.0.0/0	launch-wizard-1	–

(2. ábra: AWS konfigurációk, Portok kinyitása, Forrás: <https://eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#SecurityGroup:groupId=sg-00434d76cfadc1a74>)

A típus kiválasztását követte az operációs rendszer kiválasztása.

Itt volt lehetőségünk az igénylésnél választani az operációs rendszerek közül, a felsorolások között szerepelt:

- Windows 7
- Windows Vista
- Windows 8
- Windows 10
- Windows 11

Ezen felül jelen volt még a macOS:

- 12 Monterey (2021)
- 13 Ventura (2022)
- 14 Sonoma (2023)

A felsorolás további része a következő oldalon található, megszakítást nem szerettem volna beletenni a felsorolásba, így külön oldalra került a Linuxra eső része a listának.

A választás ezeken kívül esett a Linuxra, ahol szintén nagy volt a választható lista mérete:

- CentOS
- Debian
- Fedora
- RHEL (Red Hat Enterprise Linux)
- SUSE (Software und System-Entwicklung)
- Ubuntu
- Oracle
- Bitnami
- Rocky Linux.

Ezek közül a választás a Debian-ra (Debian 5.10.179-2 verzió, 2023.07.14) esett. A választás azért erre a verzióra esett, mert 2023.06.25-én. ez volt a legfrissebb elérhető verzió, amelyet az Amazon biztosított.

Korábban volt tapasztalatunk a Debian rendszerekkel, Rikk János tanár úr kérésére telepíteni és használatba helyezni kellett, egy általa biztosított Raspberry Pi-t szerverként.

A szerverként futó Raspberry Pi-t arra használtuk, hogy a tanév tananyagát adatmegőrzés és védelem érdekében duplikáltunk (2021.05.08).

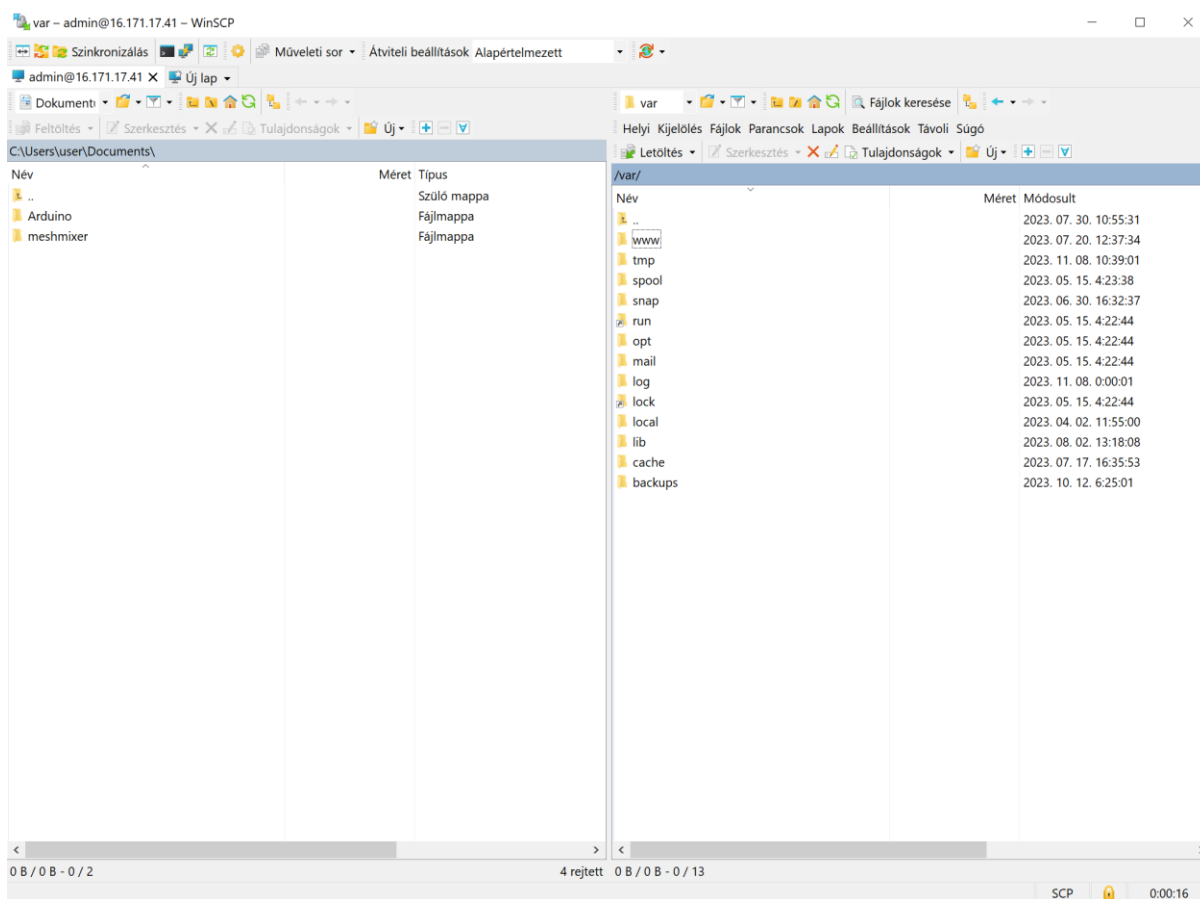
A konfigurációja hasonló volt a két operációs rendszernek (Raspberry Pi Debian, „Közönséges” Debian azonos alapúak).

A telepítés részletei bővebben Ódor Márk szakdolgozatában a 3. fejezetben olvashatóak.

3.2. Programok telepítésének bemutatása

Terminálból, a következő parancsok használatával a sorban meghatározott programok telepítését engedélyezte számunkra. A szerveren szükséges erőforrások és programok a következőképp kerültek fel telepítésre.

Ennek a kivitelezése PUTTY-on keresztül történt, amely egy szabad és nyílt forráskódú terminálemulátor, soros konzol és hálózati fájlátviteli alkalmazás. A választás azért a PUTTY-ra esett, mivel korábbi használatból eredően volt tapasztalat a használatában. Továbbá külső program használata történt még a szerver mappa rendszer elérésénél, a WinSCP (Szabad és nyílt forráskódú SFTP (SSH File Transfer Protocol), FTP (File Transfer Protocol), WebDAV (Web Distributed Authoring and Versioning) és SCP (Secure Copy Protocol) kliens a Microsoft Windows-ra.), a program segítségével elérhetővé vált a szerver, mint a Windows-os fájlkezelő. (vö. 3. ábra)



(3. ábra Mappa-szerű elérés, Forrás: WinSCP-n keresztüli elérése a szervernek)

3.2.1. Apache2 telepítése

apt update → *Általános frissítés frissítés*

apt install apache2 → *Apache2 telepítése*

a2enmod ssl → *SSL engedélyezése*

systemctl restart apache2 → *Apache újraindítása*

systemctl status apache2 → *Apache helyzetjelentés kérése*

A saját aláírt tanúsítványon lévő aláírás védi a tanúsítványon belüli információ sértetlenségét, de nem garantálja az információk hitelességét.

A titkosítás további részleteiről és a tanúsítvány kialakításáról Ódor Márk dokumentálásában olvashat, az 5. fejezetben olvashat.

openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout → *Titkosítás beállítása*

/etc/ssl/private/apache-selfsigned.key -out → *A kulcs létrehozása (saját aláírású)*

/etc/ssl/certs/apache-selfsigned.crt → *A tanúsítvány létrehozása (saját aláírású)*

vim /etc/apache2/sites-enabled/000-default.conf → *Default beállítások megnyitása*

A "000-default.conf" fájl megnyitását követően a 443-as portot ki kell nyitni, mivel alapértelmezetten a portok kinyitása attól még, hogy próbálunk rajtuk kommunikálni nem történik meg. (vö. 4. ábra)

A „vim” parancsokkal működő terminál-sori kommunikációk mindegyike a Vim-et () nyitja meg, amely egy több platformon is futó szövegszerkesztő (Linuxon a legelterjedtebb) hasonlóképp, mint a Windows bármelyikében a Notepad („.txt” kiterjesztésű fájlok szerkesztésénél alapértelmezett).

```
<VirtualHost *:443>
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
    SSLEngine on
    SSLCertificateFile /etc/apache2/certs/apache.crt
    SSLCertificateKeyFile /etc/apache2/certs/apache.key
</VirtualHost>
# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

(4. ábra vim /etc/apache2/sites-enabled/000-default.conf kiegészítése, 443-as port konfigurálása, Forrás: szerveren: /etc/apache2/sites-enabled/000-default.conf)

A szerver további beállításához tartozó lépések:

apache2ctl configtest → *Beállítások ellenőrzése az apache2 programnál*

systemctl reload apache2 → *Apache2 újratöltése, újraindítása*

3.2.2. Tűzfal telepítése

apt install ufw → *Uncomplicated firewall telepítése (tűzfal)*
ufw app list → *Kilistázza az átengedett programokat*
ufw default deny incoming → *Tiltja a beérkező kéréseket*
ufw default allow outgoing → *Engedélyezi a kimenő kéréseket*
ufw allow ssh → *Engedélyezi a „secure shell”(SSH) kapcsolatot*
ufw allow 'WWW Full' → *Engedélyezi a www elérését*
ufw enable → *Bekapcsolja a tűzfalat*
ufw status verbose → *Az aktív szűrési szabályait megtekintésre bocsátja*

3.2.3. POSTGRESQL telepítése

(postgresql; <https://www.postgresql.org/>):

1. ' wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key add -' → *Hozzáférést a postgresql-hez*
- 2.1. ' echo "deb http://apt.postgresql.org/pub/repos/apt/ \$(lsb_release -cs)-pgdg main"' → *Az „echo” paranccsal kiírja a szöveget, megadja a postgresql-hez tartozó legfrissebb csomagokat tartalmazó mappának az elérési útját. (deb=binary package, -pgdg=kódból kinyerve készíti el a mappa nevet, main=specifikáltan azt a komponenst keressük)*
- 2.2. ' | sudo tee /etc/apt/sources.list.d/postgresql-pgdg.list > /dev/null ' → *Ez a parancs az előző echo alapját véve írja bele egy file-ba ami az adott útvonalon elérhető a fájl.*
- Összességében az 2.1. és a 2.2. pont arra szolgál, hogy amennyiben van package amit le szeretnénk szedni a postgresql-hez ez sikeres legyen és alaphoz ide tegye a telepítés során őket.
3. 'sudo apt-get update -y -' → *Bármely frissítés van, szedje le*
4. 'sudo apt-get install postgresql' → *Telepítése a postgresql-nek*
5. 'sudo netstat -tunlp | grep 5432' → *Direkt címzést ír nem pedig bármely fedést (pl. port/username lehet fedés), figyeli a kommunikációt, ID-t ad a felhasználó programoknak (egyéni szám-címzés)*

6. 'sudo psql --version' → *Verziószám lekérése*
7. Ezek után az elérése a mappákban: /lib/postgresql/15/bin.
8. 'adduser postgres' → *Postgres felhasználó hozzáadása*
9. 'mkdir -p /usr/local/pgsql/data' → *Mappa létrehozása*
10. 'chown postgres /usr/local/pgsql/data' → *Hozzáférhetőség beállítása a mappán, hozzárendelés a postgres-hez*
11. 'su - postgres' → *Postgres felhasználóként bejelentkezés*
12. '/usr/local/pgsql/bin/createdb usda' → *Az „usda” nevezetű táblázat létrehozása*
13. '/usr/local/pgsql/bin/psql usda' → *Adatbázis megnyitása*

A hozzáférések beállítása a postgresql-nél, ekkor van meghatározva, hogy miként kommunikál a szerver komponenseivel, localhostot hoz létre és azon keresztül történik az adatbázisok elérése, pgadmin4 segítségével.

```
# DO NOT DISABLE!
# If you change this first entry you will need to make sure that the
# database superuser can access the database using some other method.
# Noninteractive access to all databases is required during automatic
# maintenance (custom daily cronjobs, replication, and similar tasks).
#
# Database administrative login by Unix domain socket
local    all             postgres                                md5

# TYPE      DATABASE      USER      ADDRESS              METHOD

# "local" is for Unix domain socket connections only
local    all             all                                trust
# IPv4 local connections:
host     all             all        127.0.0.1/32         trust
# IPv6 local connections:
host     all             all        ::1/128              trust
# Allow replication connections from localhost, by a user with the
# replication privilege.
local    replication     all                                trust
host     replication     all        127.0.0.1/32         trust
host     replication     all        ::1/128              trust
```

(5. ábra Postgresql konfigurációs fájl, amelybe a csatlakozásokat és információközléseket lehet nyitni, vagy limitálni. Forrás: /etc/postgresql/15/main/pg_hba.conf szerver mapparendszere)

A szerkesztés eleinte hibásan történt, hiszen a portok megnyitása nélkül próbálkoztam elérni és ezért a kapcsolat semmiféle képpen nem akart felállni.

3.2.4. PHP telepítése:

A műveletek elvégzéséhez PHP-t használtam, ennek a telepítése az alábbi pontok alatt

1.: 'sudo apt -y install php php-common' → *Feltelepíti a php-t*

2.: 'php -v' → *Verziószám ellenőrzés*

3.: 'sudo apt -y install php-cli php-fpm php-json php-pdo php-mysql php-zip php-gd php-mbstring php-curl php-xml php-pear php-bcmath' → *A legtöbb hozzátartozó kiegészítő csomag telepítése*

4.: 'sudo apt-get install php8.2-curl' → *A curl hozzátartozó kiegészítő csomag telepítése*

A csomagok telepítését követően, ahhoz hogy működjön a továbbiakban a "curl" meghívás ahhoz szükségünk van a php.ini fájlban az átírására, meg kell keresni benne a ";extension=curl" sort és kivenni a sor elejétől a pontosvesszőt, ezzel engedélyezni a bővítményt. (vö. 6.ábra)

```
extension=curl
;extension=ffi
;extension=ftp
;extension=fileinfo
;extension=gd
;extension=gettext
;extension=gmp
;extension=intl
;extension=imap
;extension=mbstring
;extension=exif      ; Must be after mbstring as it depends on it
;extension=mysqli
;extension=oci8_12c  ; Use with Oracle Database 12c Instant Client
;extension=oci8_19  ; Use with Oracle Database 19 Instant Client
;extension=odbc
;extension=openssl
;extension=pdo_firebird
;extension=pdo_mysql
;extension=pdo_oci
;extension=pdo_odbc
;extension=pdo_pgsql
;extension=pdo_sqlite
;extension=pgsql
```

(6. ábra PHP bővítmények engedélyezései, Forrás: szerver, /etc/php/8.2/apache2/php.ini)

3.2.5. Routing megoldások

Az alapvető működési módszere, amivel az apache2 üzemelt, az, hogy egy alapértelmezett HTML fájl kerül lefuttatásra, ennek az átirányításához használjuk a „.htaccess” fájlt, beleírásra kerül az átirányítás (vö. 7. ábra).

```
RewriteEngine On
RewriteBase /
RewriteCond %{REQUEST_FILENAME} !-d
RewriteCond %{REQUEST_FILENAME} !-f
RewriteRule ^(.+)$ index.php [QSA,L]
```

(7. ábra. htaccess beltartalma, Forrás: szerver, /var/www/html/.htaccess)

3.3. Adatbázis

Adatbázis után kutatva találtunk rengeteg forrásból származó nutríciós adatot, eleinte nem foglalkoztunk adott témával/termékkel specifikusan, mint sajt vagy magvak. Rengeteg helyről lehetett volna különböző forrás helyek megjelölésével összehozni a táblázatunkat, amely az adatagyont képezi. Vizsgálatnak indultunk eleinte a NÉBIH adattárával (Nemzeti Élelmiszerlánc-biztonsági Hivatal Forrás: <https://portal.nebih.gov.hu/adatbazisok-elelmiszer>) azonban nem volt kielégítő mennyiségű adat benne, ezért kerestünk egy nagyobb adatmennyiséggel rendelkező adatbázist. Ennek az adattárnak a limitációja az import és export termékek átfutása az országon, illetve tartalmaz még tejpiaci jelentést is, ez nem termékenkénti pontos lebontás, amelyet kerestünk is volna.

További keresés történt még adatbázis után, amelyet számításba vettünk még, FAO (Food and Agricultural Organization of the United Nations, Forrás: <https://www.fao.org/faostat/en/#home>).

A keresési feltételek bejelölése után, tudtunk szűrni pontos és naprakész adatok után, azonban a szűrési feltételek korlátozták a termékekről nyert adatokat, a felhasználásukat és a mozgatásukat (import és export) elérhettük, azonban a kiválasztott termékek listáján belül a termékleírások, amelyek a nutríciós adatokat tartalmazzák, nem voltak elérhetőek, ezért nem erre esett a választás.

A termékek egyedi kiválasztása (sajtok, húsok és magvak), kerestünk egyesével adattárakat a termékekre. Ekkor találtuk a Philadelphiában található Pennsylvanai Egyetem (Forrás: <https://extension.psu.edu/selecting-cheese-for-health>) által biztosított táblázatot, amely ellátta

volna a feladatát egyes termékek adattárát. A gond a rendelkezésre álló adat mennyiségével volt, a sajtok kiválasztása során csupán 18 különböző sajt volt elérhető.

Azonban a részletessége az Egyetem adattáblájának kiemelkedő volt, hiszen az oszlopok mennyisége, a táblázat „szélessége” megfelelő lett volna az ODESSAI létrehozásához.

Ekkor döntöttük el, hogy amíg megtaláljuk a megfelelő adatbázist, amely egy helyről szolgáltatja a megfelelő mennyiségű és minőségű adatot (minőség = részletesség, hitelesség) a sajtok, húsok és magvak csoportjában, addig nem szabad abbahagyni a keresést.

Ehhez hasonló (egyed termékek adatrészletességét tekintve), adatbázisként kezelő szolgáltatást, amely nagy háttér tartalommal rendelkezik találtam, mégpedig a Kalóriabázist. (vö. 8. ábra)

A 8. ábrán egy kalóriatáblázat látható, amely a Kalóriabázis egyik nézete, ellát diétás étrend segítségnyújtást, illetve lehetőséget ad felvinni a fogyasztott termékek adatait, hogy figyelhető legyen a napi kalóriabevitel, egészség szempontjából pedig a vitaminok és egyéb hasznos tápanyagok.

KALÓRIATÁBLÁZAT: Összes kategória

OKOS KALÓRIATÁBLÁZAT: összeválogattuk a leggyakoribb 1000 ételt és alapanyagot, hogy egy minden szempontból rendezhető, interaktív kalóriatáblázat álljon rendelkezésedre! Szűrhetesz kategóriákra a képekre kattintva, tényezők szerint növekvő és csökkenő sorrendbe rendezhetsz a nyilakkal és minden tényezőt az egyes rubrikák bepipálásával tovább szűrhetesz (lenti rubrika alacsony értékűek, fentiek a magas). Például meg tudod csinálni, hogy rendezed FEHÉRJE szerint azokat a TEJTERMÉKEKET, melyek KÁLCIUMTARTALMA közepesen és nagyon magas, de VASTARTALMA minimális. A kérdőjeleket igyekezzünk kitölteni. Segíteni így tudsz: a keresőből megnyitva ("I" gomb), beírhatók a hiányzó adatok (segítségére lehetnek a címkézőtők).

Összes kategória	Kalória	Zsír	Szénhidrát	Fehérje	Rost	Gyümölcs	Cukor	Kalcium	Vas	Magnézium	Fosfor	Kálium	Nátrium	Cink	C-vitamin
Banán	89	0.33	22.8	1.09	2.6	1	18.2	6	0.26	60	94	480	1	0.15	8.7
Paradicsom	18	0.2	3.9	0.88	2.6	?	2.63	10	0.27	20	24	294	5	0.17	19
Alma	52	0.17	13.8	0.26	1.3	100	10.4	6	0.12	5	11	144	1	0.04	4.6
Tehéntej (1,5%)	44	1.5	4.6	3.1	0	?	4.7	125	0	15	112	182	52.1	0.41	2.5
Fehér kenyér	270	3	52.3	8.3	2.3	?	5	35	4.89	26	103	127	478	0.95	0
Fehér rizs (főtt)	130	0.3	28.2	2.7	1	?	0.05	2	0.14	5	8	10	5	0.41	0
Tojás	143	9.51	0.72	12.6	0	?	0.37	56	1.75	12	198	138	142	1.29	0
Trappista sajt, Trappista sajt	381	28.1	1.6	27.7	0	?	0.57	650	0.43	24	360	136	534	2.6	0
Kígyóborka	12	0.2	1.9	0.7	0.5	?	1.67	16	0.28	13	24	147	2	0.2	2.8
Főtt tojás, Lágytojás	155	10.6	1.12	12.6	0	?	0.37	9	1	14	130	161	75	0.74	0

(8. ábra: Kalóriabázis, “Kalóriatáblázat” nézete, Forrás: <https://kaloriabazis.hu/kaloriatablazat>)

További keresést követően találtam rá az USDA (United States Department of Agriculture, <https://fdc.nal.usda.gov/>) táblázatára, amely rendelkezik mérésekkel a kiválasztott termékek kapcsán, mint a magvak, húsok, illetve a sajtok.

Az itt fellelhető adatok elérhetőek voltak adatbázis formátumban, minden termék egy állományban, amely megkönnyítette az adatok begyűjtését, ezek után kellett formálni, hogy számunkra feldolgozható formában érjük el.

Szétszedésre kerültek az általunk kiválasztott élelmiszer-termékek, sajtok, húsok és magvak, ezeket megszűrve, ha bárhol tartalmazott duplikációt, akkor azokat leszedtük, amennyiben volt benne hiányos adat, azt vagy megkerestük a hozzá tartozó adattáblában, vagy kiszedtük azt a terméket, amely hiányos volt, hogy ne okozzon gondot a humán szakértőnek.

4. Weblap működésének és megírt kódnak a bemutatása

A weblap egy Amazon Web Services által biztosított t3.micro szerveren van, amelyen Debian Linux fut. A szerverre van telepítve az Apache2, amelynek a telepítési részleteit, illetve beállítását a 3. fejezet 2. alfejezetében olvashatta.

A kódok létrehozási, időrendi sorrendben kerültek listázásra.

4.1. Request Handler

PHP-ban létrehozott segédfájl, amely egyszeri hozzárendelésében, az index.php fájlban importálásra kerül (require_once parancs felhasználással).

Működésének alapja a bejövő kommunikációnak a kezelése, "Request Handler" az általam írt fájl neve, ebben minden, ami az URL címünk mögé kerül tehát www.dieteticexpert.hu/OLAP, jelen esetben az "OLAP" áll mögötte, mint „POST” utasítás, amely azt jelenti, hogy a szerver külső forrásból kapott egy kérelmet, hogy lássa el a feladatát, ami jelenlegi helyzetben a feltételek vizsgálata, hogy létezik-e a beírt paraméter. Ebben az esetben a kód mely ezt vezérel, megkapja, hogy „POST” a metodika, amelyet végre kell, hajtson, megnézi, hogy előre lett-e konfigurálva már olyan, amely ennek a szó megfelelőnek van deklarálva, amely a „/” jelet követi.

POST, GET, DELETE, PUT parancsokat, a továbbiakban "\$method" néven érem el, ezzel utalva a meghívás módszerének. Amennyiben van ilyen, a ciklus végrehajtása folytatódik. Ezek után belép, végrehajtja az adott paraméterhez csatolt kikötéseket, adatok lekezelését és hozzáfűzi a megfelelő ágon található "tpl" kiterjesztésű fájlokat.

4.2. Smarty template engine

Smarty (Forrás: https://www.smarty.net/quick_install szerveren: /var/www/html/smarty/templates)

A Smarty egy úgy nevezett „template engine”, amely segíti az általános fájlok közötti ugrálást, változó átadást, meghívásokat és lehetőséget biztosít ahhoz, hogy megfelelően működjön a rendszer, egy szerteágazó meghívás sorozatban is.

A “tpl” kiterjesztés a “Smarty PHP template engine”-hez tartozik, ezzel kötöm össze az egymáshoz szükséges, műveletekben összefüggő fájlokat.

Amennyiben nincsen paraméter megadva, automatikusan a “home”, azaz a kezdőlapot hozza be, ezzel is egyszerűsítve az “error”, azaz hiba oldalak megszűnésével a megjelenítést.

A “tpl” kiterjesztésű fájlok segítségével történik a “skeleton” felépítése, amely a vázat jelenti, bármilyen összeépítés könnyedén kikapcsolható, hozzáadható, mivel a meghívások gyűjtője egy helyen van (vö. 9. ábra). (\$request=POST/GET jelen esetben)

(skeleton=a teljes váz, amely a meghívásokat tartalmazza adott kérésekre, URL címben történő keresésre választ ad, pl. www.dieteticexpert.hu/OLAP).

Ehhez kellett telepíteni az elején egy Composert, terminálon keresztül, ezzel tudtuk elérni a legtöbb PHP kiegészítést, amelyek közé tartozik a Smarty is.

```
{include file='navigation.tpl'}
<section>
    {if $request eq 'home'}
        {include file='home.tpl'}
    {/if}
    {if $request eq 'menu2'}
        {include file='olap2.tpl'}
    {/if}
    {if $request eq 'menu1'}
        {include file='dummy.tpl'}
    {/if}
    {if $request eq 'about'}
        {include file='about.tpl'}
    {/if}
    {if $request eq 'OLAP'}
        {include file='try.tpl'}
    {/if}
</section>
{include file='footer.tpl'}
```

(9. ábra Index.tpl beltartalma, Forrás: /var/www/html/smarty/templates/index.tpl)

A fő működési elve a template, azaz “tpl” kiterjesztésű fájlokban rejlik, ezek tartalma lehet HTML-ként üzemelő meghívás, azonban ezekbe ékelve a “{}” zárójelek használatával lehetőség nyílik arra, hogy az ember használja a PHP megszólítását is.

```
<div id="result"></div>
<center><h3>{$trans['OLAP1']}</h3><a href="OLAP">{$trans['back']}</a></center>
</div>
<center>{$trans['usda']}<a href="https://fdc.nal.usda.gov/">USDA</a>
```

(10. ábra Meghívási szintaktika bemutatása, forrás /var/www/html/smarty/templates/try.tpl)

4.3. Skeleton felépítése

A “skeleton”, más nevén a csontváz, amire felépül az ODESSAI, Smarty template fájlok összefűzésén alapszik. Beérkezik a meghívás, kiválasztódik a fájl, az a fájl betöltésre kerül, mint tartalom.

A fő lényege és előnye, hogy a projekt részletekből áll, könnyebb a változtatásokat implementálni (pl. új funkció ellátása, egy szimpla “include” paranccsal megoldódik).

Több fájl szükséges hozzá, hogy a váz elkészülhessen, az ODESSAI esetében ez az “index.php”, amiben az útvonalakhoz történő elválasztás és a funkciók összehívása történik, illetve az “index.tpl”, ahol a PHP funkcióból kinyert és átküldött változók meghívásra kerülnek, amelyek meghatározzák, hogy melyik további fájlok szükségesek, hogy a tartalom teljes legyen. Például JavaScript meghívása (vö. 11. ábra), vagy a stíluselemek meghívása (vö. 12. ábra)

```
<script>
var lng="{$language}"</script>
<script src="./assets/js/js.js"></script>
```

(11. ábra JavaScript meghívása, Forrás: /var/www/html/smarty/templates/index.tpl)

```
<link href="./assets/css/bootstrap.min.css" rel="stylesheet">
<link href="./assets/css/mdb.min.css" rel="stylesheet" />
```

(12. ábra Stíluselemek meghívása, Forrás: /var/www/html/smarty/templates/index.tpl)

4.4. Stíluselemek

A HTML megformázása, stílus elemekkel való ellátása és az MDBootstrap funkcióinak ellátása bővebben olvasható Ódor Márk dolgozatában. A 4. fejezet 4. alfejezetében olvasható.

Tartalmazza az ODESSAI, a stílus elemek között a font-awesome-on (Forrás: <https://fontawesome.com/icons/categories/coding>) található, az oldal navigációs sorában elhelyezkedő, egészségügyi ikonok között található “pulzáló szív”, ennek a megtalálása az én feladatom volt. (Forrás: <https://fontawesome.com/icons/heart-pulse?f=classic&s=solid>)

4.5. Trans Handler, kétnyelvűség megoldása

Implementálni kellett a 2 nyelvűséget is ide, amely a szavak yaml-ként mentését jelenti, ebben a fájlban található az összes magyar-angol megfelelője az adott változónak.

Mindezt egy transhandler.inc.php nevezetű PHP kiterjesztésű fájlal kezelek, mely a beérkező request-be megkapja, hogy melyik nyelv a jelenleg kiválasztott és ehhez tölti be a változókat a hozzátartozó yaml fájlokból.

Amennyiben nem létezik a hozzátartozó nyelvi mappa (pl en.yaml, hu.yaml) akkor saját magának generál egyet.

Ezt a mappát ellátja a linux paraméterezése szerint olvasási, szerkesztési és futtatási joggal (0777-> rwxrwxrwx, read/write/execute).

Az akkor használt változókat beletölti, ekkor fordítás nélkül.

(ezt a felhasználó tovább folytathatja, azzal, hogy beírja a megfelelő jelentését a szavaknak).

Ennek az alkalmazása a programkód elején egy „require_once” paranccsal fut, amely akár hányszor kerül említésre, csak egyszer fut le, ezzel optimalizálva az erőforráshasználatot.

```
require_once __DIR__.' /inc/requesthandler.inc.php';  
require_once __DIR__.' /inc/transhandler.inc.php';  
require_once __DIR__.' /inc/smarty.inc.php';
```

(13. ábra Importálás bemutatása, Forrás: /var/www/html/index.php a szerver mapparendszerében)

Az importálás bemutatása után pedig, amennyiben a fordítást szeretné használni a felhasználó akkor a „Trans Handler” PHP meghívása a 14. ábrán látható.

```
<label for="category">{$trans['nameoftable']}</label><br>
<select class="form-control" id="category" name="category"><p>{$trans['tables']}</p>
  <option value="">{$trans["Select Item"]}</option>
  <option value="public.usdae">{$trans['sajt']}</option>
  <option value="hus">{$trans['hus']}</option>
  <option value="magvak">{$trans['mag']}</option>
</select>
```

(14. ábra fordítás meghívása, Forrás: /var/www/html/smarty/templates)

A jelenleg aktív “translations”-t hívja meg és annak a tartalmában megkeresi a ‘Content’ névre hallgató változót.

Ezt tölti bele a feljebb meghívást mutató változóba, minden yaml fájlban használnia kell a ‘Content’ szót, ebben az esetben, mert ha az üres lesz, az hibát okoz, ha nyelvet vált a felhasználó (vö. 15. ábra).

```
1  Language: Magyar
2  PageTitle: Online Dietetikus Szakértő
3  Greetings: Csókolom!
4
5  home: Kezdőlap
6  menu1: Menü 1
7  menu2: OLAP_nyers
8  about: Rólunk
9  Info: Információ
```

(15. ábra részlet a Trans Handler által gyártott hu.yaml fájlból, Forrás: /var/www/html/ a szerver mappa rendszerében)

Részlet a hu.yaml fájlból, hasolóképp kell megadni az összes változót, azonban befogadja a html szintaktikai megoldásait is (például a mondatba szúrva egy
).

4.6. JavaScript/Ajax funkciók

Létrehoztam a JavaScript megismerését követően egy nyilat, amely az oldal „footer” részében helyezkedik el (lábjegyzet része az oldalnak), ennek a funkciója az oldal tetejére vezéreli az olvasót.

A weblapon, a “/OLAP” címzés alatt található menüpontoknál, a kiválasztás Ajax funkciókkal történik. Ezek azt takarják, hogy amennyiben bárki megváltoztatja a választott terméket (alap esetben üres, mivel nincs kiválasztás), a funkció lefut, ezzel a választáshoz állítja a kijelzett információkat/adatokat.

Ezen funkciók közé tartozik például az élelmiszer-termék választása, amennyiben a felhasználó kiválasztja a sajtokat, mint élelmiszer-termék, abban a pillanatban a sajtok táblának az oszlopfejlécei megjelennek a második legördülő menüpontban, amelyre további funkciók épülnek.

Az első kiválasztást követően, amennyiben nem üresek a további menüpontok, akkor törlődnek a belső tartalmaik, hogy tiszta legyen, és újra lehessen tölteni választáshoz megfelelően.

A második választást követően, a harmadik választható menücsoporthoz a másodikban kiválasztottak alapján töltődik fel, ezzel hordozva a „csomagot”, amelyet a választásnál megadott a felhasználó (csomag=kiválasztott elemek listája).

Ezen felül a negyedik választható legördülő menücsoporthoz hasonlóan, mint a harmadik, az előtte kiválasztott elemek alapján töltődik fel, ezzel lehetőséget adva a felhasználónak arra, hogy beállítsa a személyes preferenciáit, amelyre szűrni szeretne.

5. Elemzés

Ebben a fejezetben az alábbi alfejezetek kerülnek bemutatásra: Elemzés bemutatása naiv és optimalizált módon (vö. 5.1), COCO bemutatása és az adatcsere megoldásai (vö. 5.2), a végtermék, miképp jelenítjük meg a kezelés után (adatvizualizáció) (vö. 5.3)

A sorrend párhuzamosan is érthető lehet, viszont a vizsgálatok felépítése, kezdve a naiv móddal majd pedig az optimalizáció, fontos a megértés szempontjából, hiszen erre épül fel az ODESSAI. A terjedelme az 5. fejezet alfejezeteinek (vö. 5.1-5.2-5.3) a dolgozat terjedelmi korlátai és a rendelkezésre álló idő tekintetében arányosan csökkentett, az elemzés értelmezhetőségén azonban nem ront.

5.1. Elemzés bemutatása naiv és optimalizált módon

A kiválasztott adatokat oszloponként ranglistázzuk, így mindenki kap egy pontozást, az adott darabszámhoz megfelelően (pl. 100 sajtónál 1-100ig van a ranglista).

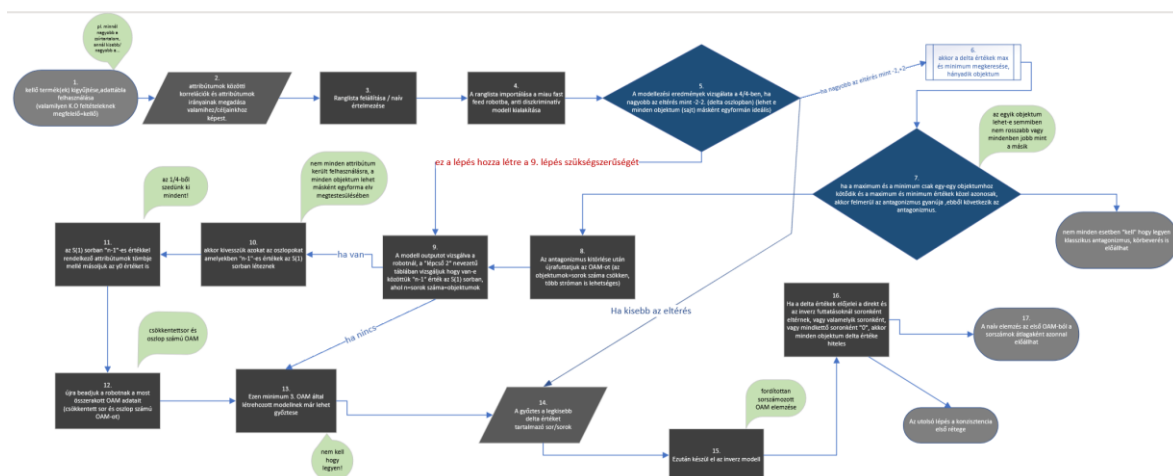
Az adatok irányának meghatározásáról létrehozunk egy külön sort, amely egy 0 vagy 1 értéket tartalmaz aszerint, hogy a felhasználó mit szeretne az adott attribútummal, tehát, a felhasználó milyen attribútumból szeretne többet, vagy épp kevesebbet (attribútum = oszlopfejléc=pl. protein, érték 0 vagy 1, alapértelmezetten 0, tehát nem történik vele semmi, számolás szempontjából irreleváns).

A táblázat sorait összegezve létrehozunk a táblázatban egy ranglista oszlopot és megadjuk a ranglista elemei közül, hogy melyik lenne a legjobb az alapján, hogy a korábban felállított oszlop alapú ranglistáknak mi az összértéke. (protein 12, szénhidrát 35, hamu, 52, ranglista értéke ennek az átlaga: 33). Ezt az értéket a többi sorral összevetve ranglistába állítjuk őket, majd az 1. helyezett lesz a naiv vizsgálatunk “győztese”. Ez a számolás pontatlanságot szül, mivel az attribútumok közötti összefüggések nincsenek a számításban figyelembe véve. Korai verziója az adatvizualizációhoz tartozik, mivel kimutatásunk során használtunk “hő térképet” az adatok értelmezéséhez.(vö. 16. ábra). A 16. ábrán látható Excelen felhasznált színskála a minél több annál pirosabb, illetve minél kevesebb annál zöldbe és ennek a megfordítása lett felhasználva.

Sajt 100gramm	Energia (kcal)	Fehérje (g)	Zsír (g)	Szénhidrát (g)	Rost	Cukor (g)	Hamu (g)	Kalcium; CA(g)	Vas; Fe (g)	Telített zsír (g)	Sodium (g)	Ár /100g (HUF)	Links for prices	links for meta data	Sum
Brie	334.0	20.8	27.7	0.5	0.0	0.5	2.7	0.0	0.0	17.4	0.1	850.0	https://bevasarlas.	https://fdc.nal	1253.6
Camembert	300.0	19.8	24.3	0.5	0.0	0.5	0.0	0.0	0.0	15.3	0.1	432.0	https://bevasarlas.	https://fdc.nal	792.4
Cheddar	409.0	23.3	34.0	2.4	0.0	0.3	3.7	0.1	0.0	19.2	0.1	1029.0	https://bevasarlas.	https://fdc.nal	1521.1
Túró	103.0	11.6	4.2	4.6	0.0	0.0	1.5	0.0	0.0	2.6	0.0	288.0	https://bevasarlas.	https://fdc.nal	415.5
Krémsajt	343.0	5.8	33.5	4.6	0.0	0.0	1.4	0.0	0.0	19.7	0.0	473.0	https://bevasarlas.	https://fdc.nal	881.0
Feta	265.0	14.2	21.5	3.9	0.0	0.0	5.2	0.0	0.0	13.3	1.1	520.0	https://bevasarlas.	https://fdc.nal	844.3
Kecskesajt	364.0	21.6	29.8	0.1	0.0	0.1	0.0	0.0	0.0	20.6	0.0	1099.0	https://bevasarlas.	https://fdc.nal	1535.3
Gouda	356.0	24.9	27.4	2.2	0.0	2.2	3.9	0.1	0.0	17.6	0.1	831.0	https://bevasarlas.	https://fdc.nal	1265.4
Monterey Jack	373.0	24.5	30.3	0.7	0.0	0.5	0.0	0.1	0.0	19.1	0.1	350.0	https://www.tesco	https://fdc.nal	798.2
Mozzarella	296.0	23.7	20.4	4.4	0.0	1.8	0.0	0.1	0.0	11.7	0.1	575.0	https://bevasarlas.	https://fdc.nal	933.2
Muenster	368.0	23.4	30.0	1.1	0.0	1.1	3.7	0.1	0.0	19.1	0.1	910.0	https://www.tesco	https://fdc.nal	1356.5
Parmesan	420.0	29.6	28.0	12.4	0.0	1.1	7.2	0.1	0.0	15.5	1.8	1459.0	https://bevasarlas.	https://fdc.nal	1973.6
Swiss	393.0	27.0	31.0	1.4	0.0	0.0	3.0	0.1	0.0	18.2	0.0	1466.0	https://bevasarlas.	https://fdc.nal	1939.7
Mascarpone	429.0	3.6	42.9	3.6	0.0	4.6	0.0	0.0	0.0	30.4	0.0	508.0	https://bevasarlas.	https://fdc.nal	1022.1
Kéksajt	353.0	21.4	28.7	2.3	0.0	0.5	5.1	0.1	0.0	18.7	1.2	969.0	https://bevasarlas.	https://fdc.nal	1400.0
	sok jó	sok jó	kevés jó	kevés jó	sok jó	kevés jó		sok jó	sok jó	kevés jó		kevés jó			SZUM

(16. ábra Kezdetleges elemzés-bemutató Forrás: <https://docs.google.com/spreadsheets/d/1XqJrxpjm6h3pVhhXeejOPKWFQd7YlVNf/edit#gid=1990324278>)

Naiv vizsgálatunkból kiderült, hogy az alap döntések meghozása, gyakran helytelen eredményt szül, mivel nem vesszük figyelembe az attribútumok összefüggését egymással, ekkor jött szóba a robot által kivizsgálata.



(17. ábra a folyamatára bemutatása, Forrás: saját készítés)

Magyarázat: Zöld buborékok a kommentek, mellékes információk, az adat folyamatát nem befolyásoló tényező

- A kék négyszögek a választókövek, amelyeknél több irányba ágazik a folyamatára, a magyarázatai a döntéseknek benne találhatóak.
- Az ellipszis, hasonlóan mint a fekete négyszögek a lépések magyarázatát mutatják, a különbség a mintában, az elhelyezkedése, az elipszisek kezdő és végpontot jelölnek.
- A paralelogramma a 2. lépésben a változót jelenti, amennyiben mást jelöl a felhasználó akkor más lesz a betáplált adat.

A COCO (Component-based Object Comparison for Objectivity), amely egy kiértékelő robot, amely elvégzi az anti-diszkriminatív kiértékelést amennyiben pontos adatokat adunk neki.

- A táblázat elemeit ellátjuk a korábban megadott irányok alapján rangsorszámmal, amelyet felhasználva számoltatunk a robottal, ezeknek az értékei a kiválasztott menüpontok alapján történik, a Több/Kevesebb rész alatt, amikor a kiválasztást követően az adatbázisból történő lekérdezés, a megadott irányok alapján kérdezi le a táblázatot.
- Y0 érték hozzáfűzésével, minden adatot egységesen kezelve 1 millió jószág pontról indít, soronként egy oszlop hozzáadása történik és ezt számításba véve a korábban megadott adatok irányával, amellyel meghatároztuk, hogy milyen attribútumból mennyit szeretnénk (kevés/sok) számolja ki, a “győztes” terméket az összes közül, amennyiben nincsen eltérés a visszkapott fájl („output.txt”) tartalmában, akkor a lehet minden másként egyforma válasz adható a kérdésre
- Amennyiben az érték eltér bármikor, ezeket a lépéseket fontos addig ismételni, amíg a robot által visszaadott táblázatokból, a 4. táblát vizsgálva, a delta oszlopban, amelyben az eltérések vannak a maximális érték maximum 2 és minimum -2, ezzel tehető fel a mondat: Lehet-e minden “kiválasztott termék” másként egyformán ideális. Ennek a vizsgálatnak, ha az eredménye azt adja vissza, hogy több ilyen kiemelkedő értékkel rendelkező van, akkor további vizsgálat szükséges.
- Amennyiben a maximum és a minimum csak egy-egy objektumhoz kötődik és a maximum és minimum értékek közel azonosak, akkor felmerül az antagonizmus gyanúja, ebből következik az antagonizmus. Azonban nem minden esetben "kell" hogy legyen klasszikus antagonizmus, körbeverés is előállhat.
- Az értékek, amelyek eltértek, törlése után újra futtatjuk az OAM (Objektum Attribútum Mátrix)-ot (az objektumok=sorok száma, csökken, több eltért érték is lehetséges)

- A modell kimenetet vizsgálva a robotnál, a "lépcső 2" nevezetű táblában vizsgáljuk, hogy van-e közöttük "n-1" érték az S (1) sorban, ahol n=sorok száma=objektumok
- Kivesszük azokat az oszlopokat amelyekben "n-1"-es értékek az S (1) sorban léteznek
- S (1) sorban "n-1"(ahol n=adatok száma)-es értékkel rendelkező attribútumok tömbje mellé másoljuk az Y0 értéket is
- Ezt követően újra beadjuk a robotnak a most összerakott OAM adatait (csökkentett sor és oszlop számú OAM-ot)
- Ezen minimum 3. OAM által létrehozott modellnek már lehet győztese
- A győztes a legkisebb delta értéket tartalmazó sor/sorok
- Ezután készül el az inverz modell
- Ha a delta értékek előjelei a direkt és az inverz futtatásoknál soronként eltérnek, vagy valamelyik soronként, vagy mindkettő soronként "0", akkor minden objektum delta értéke hiteles

5.2. COCO bemutatása és az adatcsere megoldásai

A COCO (Component-based Object Comparison for Objectivity)-val történő kommunikáció, egy szerver-szerver közötti kommunikáció. Ehhez, hogy a kommunikáció működhessen, szükséges az SSL mindkét oldalról (Secure Sockets Layer, titkosított adatfolyam, adatvédelem).

Ezen felül létre kellett hozni egy kapcsolatot létrehozó PHP fájlt. (Forrás: https://miau.myx.hu/myx-free/coco/engine3_curl.php). A keresési adatok alapján betöltődik a kiválasztott, irányba állított oszlop ranglistázott számozása, majd hozzá kell csatolni egy Y0 oszlopot is harmadikként, amely 1.000.000 "jószág" ponttal látja el az összes sort, amelyet a kiértékelő algoritmus felfelé és lefelé torzít a megoldás érdekében.

Amennyiben lehet minden másként egyforma, akkor az értékek szorosan egymáshoz közel, a megadott értéket megtartva maradnak, akkor mondható, hogy a kiválasztott lehetőségek közül a kettő, amire a választás esett, kioltja egymást, így ebben az esetben lehet minden saját másként egyforma. Abban az esetben, hogy ezek az értékek eltérnek egymástól, felfelé és lefelé, akkor van szó arról, hogy nem lehet minden saját másként egyforma, jelen van az antagonizmus. Ebben az esetben kezdődik a további vizsgálat, amely a kitérő értékeket összehasonlítja és kiveszi a minden tekintetben rosszabbat, majd újra lefuttatja a vizsgálatot, hasonló szempontokkal.

5.3. A végtermék, miképp jelenítjük meg a kezelése után (adatvizualizáció)

A végtermék adatvizualizációs megjelenítése a következőképp folyik:

Az oldal megnyitása után a felhasználó a funkciókat ellátó www.dieteticexpert.hu/OLAP elérésen tudja használni az ODESSAI-t.

Itt található 4 választható legördülő menü, amelyek balról jobbra-egymásra épülnek. (vö. 17. ábra)

The screenshot shows a web interface for data visualization. At the top, it says 'USDA forrásai alapján készült élelmiszer-attribútum táblázat Forrás: USDA'. Below this are four dropdown menus arranged horizontally: 'Válassz ételcsoportot', 'Válassz Kategória', 'Válassz Több', and 'Válassz Kevesebb'. Below the 'Válassz Kategória' menu is a 'KERES' button.

(18. ábra /OLAP elérhetőségen a megjelenő lehetőségek, Forrás: www.dieteticexpert.hu/OLAP)

Itt a kiválasztást sorba követve a felhasználó meghatározhatja az élelmiszer-termék csoportját, amiből választani szeretne.

A következő menüpontban (2. menüpont, Kategória-címezésű) ezek lesznek láthatóak, ezekből kiválasztva töltődik fel a 3. és a 4. menüpont (Több/Kevesebb), amely az attribútumoknak, jellemzőknek lehetőséget ad irányt hozzáadni, amelyet a felhasználó szeretne (több/kevés fehérjét szeretne, és/vagy több/kevés zsírt, és/vagy több/kevés hamut, stb..).

Ezek meghatározása után a „Keres” gomb megnyomásával, a kiválasztott, feltételeknek szűrt táblázatot és a hozzá tartozó ranglistát, amelyet felállítottunk korábban az adatbázis létrehozásával, megjelenítésre küldjük a felhasználónak.

Párhuzamosan a megjelenítéssel, egy „CURL” parancs segítségével a COCO kiértékelésére küldjük az input2.txt nevezetű fájlunkat, utána az onnan megkapott, kész kimeneti adatot (output.txt) formázzuk és a felhasználó elé bocsájtjuk táblázatként formázva, ahol a táblázat elemei oszlopfejléccel ellátva (tápérték-adatok), amennyiben pedig létezne közöttük nem minden másként egyforma kérdésre választ adó, akkor további vizsgálat indul, amely ezt a kitérést az Y0 értékben vizsgálja.

A megjelenítés egy üres <div>-ben történik, amely egy HTML elem. Egy JavaScript (js.js) tölti meg, egy PHP fájl (query_database.php) közreműködésével, amely átveszi a kikeresett adatokat, azokat a kiválasztott adatok iránya szerint beleveszi a számításba (a táblázat

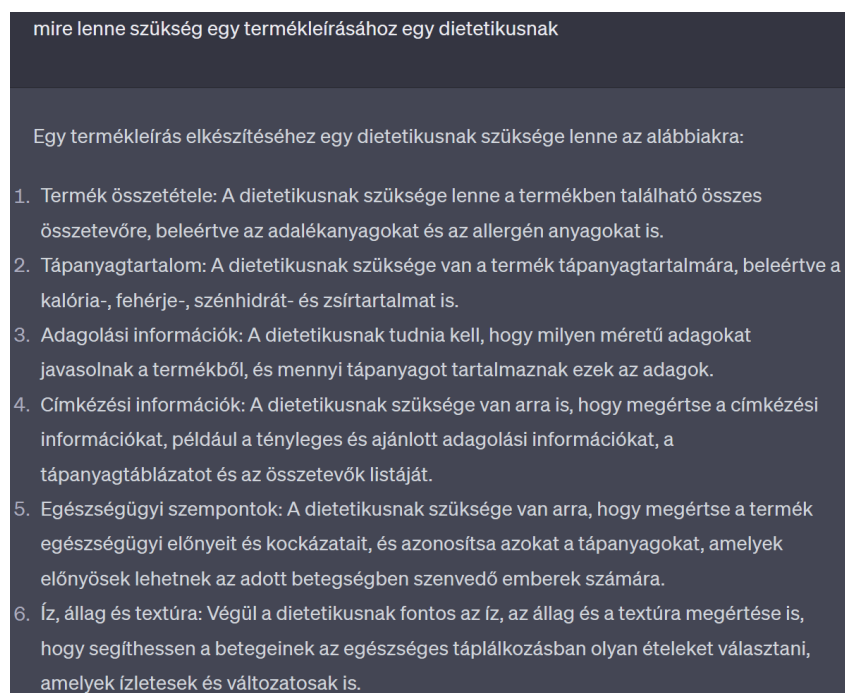
tartalmazza a 0 és 1 iránnyal rendelkező ranglistázott oszlopokat is (0=minél több annál jobb, 1=minél kevesebb annál jobb).

Majd ezeket küldi input2.txt-ként tovább cURL meghívással, ebből érkezik egy vissza-adat, amely az output.txt-ben található minden futást követően.

Az output.txt-t ezek után ellenőrizzük, hogy létezik-e a megadott $Y_0=1.000.000$ értéktől eltérő. (Ha igen, vizsgáljuk tovább, ha nem akkor pedig kiiratjuk)

6. Chat GPT – dolgozat

Ebben a fejezetben a chatGPT kötelező használatát előíró intézményi elvárásnak való megfelelő tételes bemutatása következik az alábbi alfejezetekre bontva: adatgyűjtés (vö. 6.1), optimalizálás (vö. 6.2), hibaelhárítás (vö. 6.3), kód-értelmezés (vö. 6.4), újabb ötletek kitalálásához felhasználása (vö. 6.5).



(19. ábra “Mire lenne szüksége egy dietetikusnak?”, Forrás: <https://chat.openai.com/c/5de85f4b-2078-4964-ae9e-c870885a4354>)

A megjelölt források, amelyek a chatGPT ábráihoz tartoznak, pontos meghívásra mutatnak a mesterséges intelligenciával való beszélgetésben, ezért ezek csak a beszélgető fél részére elérhetőek ezek a linkek.

A dolgozatban több helyen használatba vettük a chatGPT-t, az ötletelés során megkérdeztük, hogy “mire lenne szükség egy termékleíráshoz egy dietetikusnak?”

Erre a kérdésre visszatértünk még egyszer egy hónap múlva, hogy megnézzük, változott-e a válaszadás. A válasz mindkét alkalommal megegyezett, így megerősítést nyertünk, hogy nem csupán kitaláció, amit élénk vetett a mesterséges intelligencia.

6.1. Adatgyűjtés

Használtuk továbbá adatok gyűjtésére, hogy segítsen kiválasztott termékcsoporthoz összerakni egy táblázatot. (Ábrán látható kérdés: A korábban bemásolt táblázat részletét kérem egységesített mértékegységekkel)

Természetesen, az alábbi táblázatban az adatokat egységesített mértékegységekkel találod: 

Sajt típusa	Kalória (kcal/100 g)	Fehérje (g/100 g)	Zsír (g/100 g)	Szénhidrát (g/100 g)	Kalcium (mg/100 g)	Nátrium (mg/100 g)	Foszfor (mg/100 g)
Cheddar	403	25,4	33,1	1,3	721	621	510
Parmezán	431	41,6	29,1	3,2	1184	1603	734
Mozarella	280	22,2	20,3	2,2	505	481	369
Kék sajt	353	21,4	28,7	2,3	528	1377	421
Feta	264	14,2	21,3	4,1	493	1116	165
Brie	334	20,3	27,7	0,1	184	629	168
Camembert	300	19,8	24,0	0,5	388	842	266
Gouda	356	24,9	27,9	2,2	739	819	473
Roquefort	369	18,7	31,2	2,1	376	1590	387
Edámi	357	24,1	27,4	1,4	738	810	463
Gorgonzola	360	19,0	30,5	2,2	567	1305	292
Ricotta	174	11,3	13,1	3,1	187	87	163
Cottage cheese	98	11,1	4,3	2,4	83	364	123
Halloumi	316	20,8	26,0	2,2	655	1068	

(20. ábra Mértékegység egyesítés, Forrás: <https://chat.openai.com/c/5de85f4b-2078-4964-ae9e-c870885a4354>)

6.2. Optimalizálás

A kódoptimalizálás előkerült, mivel kezdetekben, amikor egy lekérdezés lefutott, akkor az egész adatbázisnak a kiválasztott táblájából minden elemét meghívtam (SELECT * FROM x, x=tábla neve). Ez időben és erőforrás felhasználásban sem volt optimális, mivel a táblázatban, amit lekérdeztünk egy kis részlete volt fontos a felhasználónak, amelyet PHP meghívások segítségével kezelt és formázott.

1. **Error Handling:** It's essential to add error handling to your AJAX requests. Handle possible errors like network issues, server errors, or unexpected responses.
2. **Input Validation:** Ensure that the input data is validated and sanitized before sending it to the server to prevent security vulnerabilities.
3. **Optimize DOM Manipulation:** If you're frequently updating the DOM with many elements, consider using document fragments or optimizing the way you add elements to the page for better performance.
4. **Separation of Concerns:** Make sure that your JavaScript and HTML are properly separated. Avoid inline JavaScript and consider using unobtrusive JavaScript techniques.
5. **Security:** Be aware of potential security risks, especially when dealing with user input. Ensure that your PHP scripts validate and sanitize data before using it in SQL queries to prevent SQL injection attacks.
6. **Accessibility:** Ensure that your dynamic UI elements are accessible to all users, including those who rely on screen readers or keyboard navigation.
7. **Caching:** Consider adding caching mechanisms for items retrieved from the server to improve performance and reduce the number of requests.

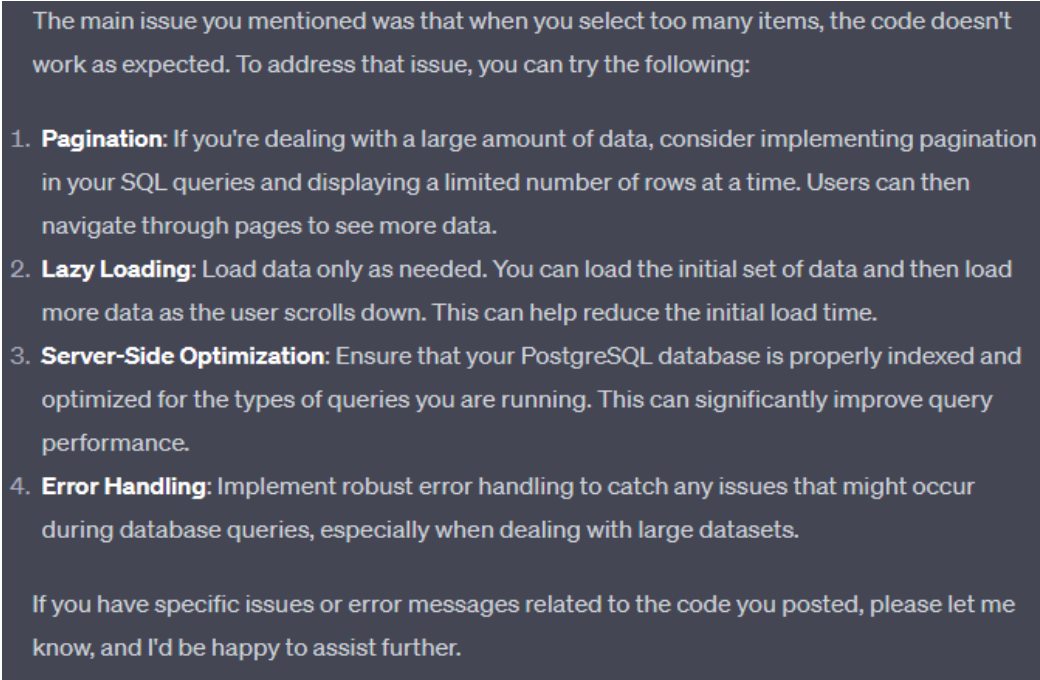
These are some general improvements and suggestions for your code. If you have specific questions or need further assistance with any part of your code, please let me know.

(21. ábra Optimalizálás kivitelezése, Forrás: <https://chat.openai.com/c/21ce8c0b-d4fa-4f7a-bafb-abd19eac9de2>)

6.3. Hibaelhárítás

Optimalizálás mellett szükség volt hibaelhárításra is (vö. 22. ábra)

Ekkor jöttek elő az erőforrás-optimalizált kódsorok, amelyek felesleges futtatást és adathasználatot gátolnak meg.



The main issue you mentioned was that when you select too many items, the code doesn't work as expected. To address that issue, you can try the following:

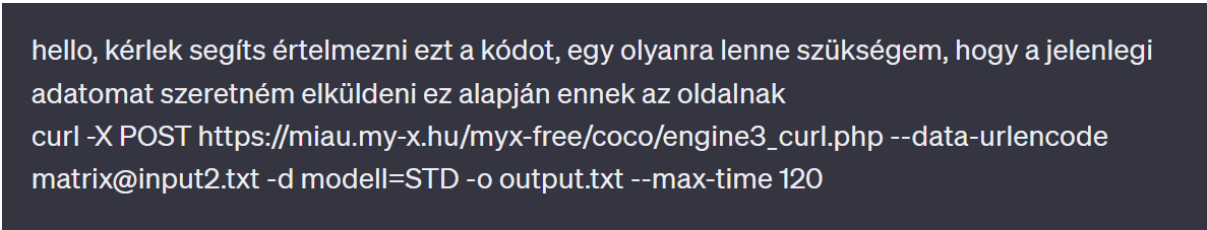
1. **Pagination:** If you're dealing with a large amount of data, consider implementing pagination in your SQL queries and displaying a limited number of rows at a time. Users can then navigate through pages to see more data.
2. **Lazy Loading:** Load data only as needed. You can load the initial set of data and then load more data as the user scrolls down. This can help reduce the initial load time.
3. **Server-Side Optimization:** Ensure that your PostgreSQL database is properly indexed and optimized for the types of queries you are running. This can significantly improve query performance.
4. **Error Handling:** Implement robust error handling to catch any issues that might occur during database queries, especially when dealing with large datasets.

If you have specific issues or error messages related to the code you posted, please let me know, and I'd be happy to assist further.

(22. ábra Hibaelhárítás/Debug, Forrás: <https://chat.openai.com/c/0c125452-40dc-4e5e-a76a-e78c616f2ddf>)

6.4. Kód Értelmezés

Használtuk számunkra ismeretlen kód darabokon, részleteken pl.: Pitlik László segítségével küldött egy meghíváshoz tartozó kódrészletet, amivel lehet információt küldeni és fogadni a COCO -tól (Component-based Object Comparison for Objectivity) (vö. 23. ábra, 24. ábra)



hello, kérlek segíts értelmezni ezt a kódot, egy olyanra lenne szükségem, hogy a jelenlegi adatomat szeretném elküldeni ez alapján ennek az oldalnak

```
curl -X POST https://miau.my-x.hu/myx-free/coco/engine3_curl.php --data-urlencode matrix@input2.txt -d modell=STD -o output.txt --max-time 120
```

(23. ábra Kóddal kapcsolatos kérdés, Forrás: <https://chat.openai.com/c/9322952f-4e7b-4993-8115-468bcbea7cca>)

A 23. ábra kérdésére a következő válasz érkezett:

1. ``curl``: Ez a parancs neve, és egy eszköz, amely lehetővé teszi a HTTP kérések küldését és fogadását a parancssorból vagy szkriptből.
2. ``-X POST``: Ez a parancssori opció azt jelzi, hogy a ``curl`` egy POST kérést fog küldeni a szervernek. A POST kérés használható adatok küldésére a szerver felé.
3. ``https://miau.my-x.hu/myx-free/coco/engine3_curl.php``: Ez a URL, ahova a POST kérés küldve lesz. Ez egy távoli szerver címe.
4. ``--data-urlencode matrix@input2.txt``: Ezzel az opcióval a ``curl`` elküldi a ``matrix`` nevű adatot a ``input2.txt`` nevű fájlból. Az adatot a POST kérés testében küldi el.
5. ``-d modell=STD``: Ezzel az opcióval egy további adatot küld el a szervernek, ahol a ``modell`` kulcshez tartozik az érték ``STD``.
6. ``-o output.txt``: Ez az opció meghatározza, hogy a választ, amit a szerver visszaküld, egy ``output.txt`` nevű fájlba mentse el.
7. ``--max-time 120``: Ezzel az opcióval beállítható egy időkorlát (másodpercekben), ameddig a ``curl`` megpróbálja elérni a szerveret. Ha a szerver nem válaszol ennyi időn belül, a ``curl`` megszakítja a kérést.

Ez a ``curl`` parancs tehát egy POST kérést indít a megadott szerver felé, adatokat küld a ``matrix`` változóból és a ``modell`` kulcs értékével, majd a szerver válaszát elmenti az ``output.txt`` nevű fájlba. Az egész műveletnek maximum 120 másodperce van befejeződni, különben megszakad.

(24. ábra Kóddal kapcsolatos válasz, Forrás: <https://chat.openai.com/c/9322952f-4e7b-4993-8115-468bcbea7cca>)

A 23. és a 24. ábra kérdésével kapcsolatban, a megfelelő fájl, amely ellátja a funkciót, nem volt egyszerű lefutású probléma. A kódok létrehozásával járó akadályt több napos próbálkozás, JavaScripttel, külső PHP próbálkozással, terminálból való cURL meghívással követett. A próbálkozások közül a terminálból való működése rögtön ment, a PHP, illetve JavaScript megoldásai lépésenként csökkentették a lehetséges problémák számát, először meg kellett bizonyosodni róla, hogy a próbálkozások alatt, amikor a szerveren nem volt még megfelelő tanúsítvány a biztonságot illetően, akkor működhet-e egyáltalán a kommunikáció. Ezt követte az egyszerű cURL PHP létrehozása, amely egy egyszerű szöveget jelenített meg, amennyiben megfelelően történt a meghívás. Következő lépésként a MIAÚ-val történő kommunikációval kellett megpróbálkozni, majd a pontos, általunk szükséges meghívást kellett reprodukálni, először külön PHP fájlban, hogy megbizonyosodjunk a kapcsolat fennállásáról. Ezt követően létrehoztam egy JavaScriptet, amely a már korábban létrehozott „Keres” gomb megnyomására

Ajax funkcióként lefutott és elküldte a „curlrequest.php” fájlak a megfelelő adatokat, amelyeket előtte feltöltött az „input2.txt”-be.

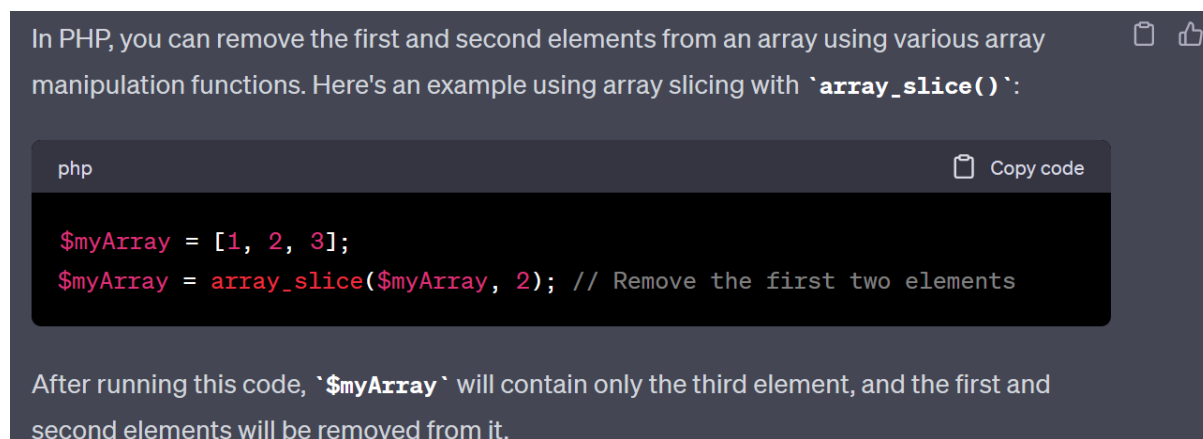
6.5. Újabb ötletek kitalálásához

Többször előfordult, hogy egy téma kapcsán egyetlen irányba vitt el a gondolat, ekkor kellett egy „külső szempont”. Tanácsadó szerepet is betöltött a dolgozat kapcsán a chatGPT, hiszen egy elakadásnál (vö.

Kérdés lefordítva, „Hogyan törölnéd az 1. és a 2. elemet a 3 vagy több elemes táblázatból?” (természetesen PHP funkció ellátásra vonatkozott a kérdés, kontextusból nem derül ki)

how would you remove 2 elements from a 3 element array, the first and second is to be removed, the third should be left intact

(25. ábra Ötletelő kérdés, Forrás: <https://chat.openai.com/c/a3306933-dc10-4a5e-94d9-d19ea91c15f4>)



The screenshot shows a chat window with a dark background. At the top, there's a question in white text: "In PHP, you can remove the first and second elements from an array using various array manipulation functions. Here's an example using array slicing with `array\_slice()`:". Below the question is a code block with a dark background and light-colored text. The code is:

```
$myArray = [1, 2, 3];  
$myArray = array_slice($myArray, 2); // Remove the first two elements
```

 Below the code block, there's an answer in white text: "After running this code, `\$myArray` will contain only the third element, and the first and second elements will be removed from it." In the top right corner of the chat window, there are icons for a document and a thumbs up.

```
In PHP, you can remove the first and second elements from an array using various array manipulation functions. Here's an example using array slicing with `array_slice()`:
```

```
php
```

```
$myArray = [1, 2, 3];  
$myArray = array_slice($myArray, 2); // Remove the first two elements
```

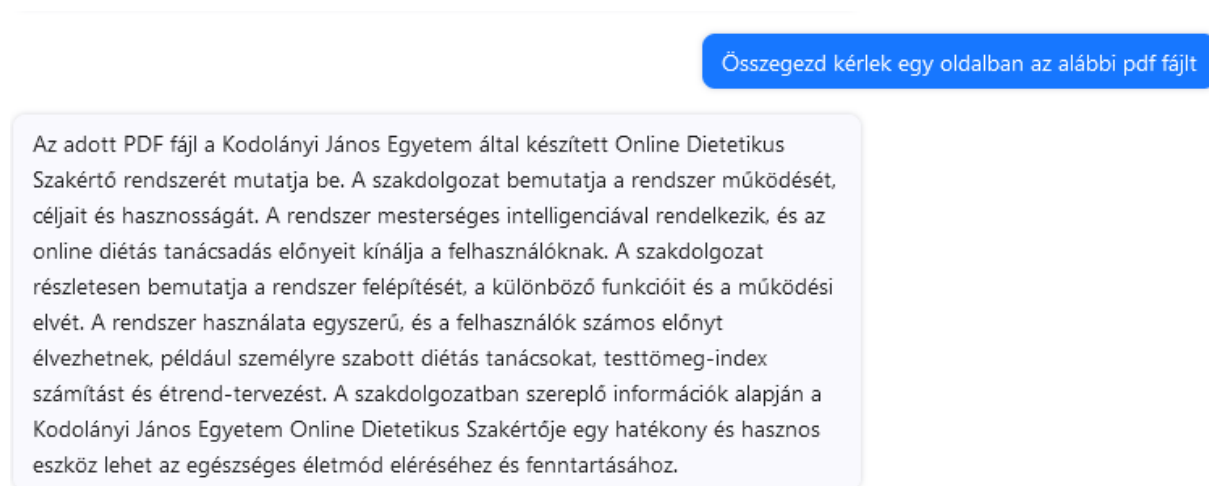
```
After running this code, `$myArray` will contain only the third element, and the first and second elements will be removed from it.
```

(26. ábra Ötletelő válasz, Forrás: <https://chat.openai.com/c/a3306933-dc10-4a5e-94d9-d19ea91c15f4>)

Élve a mesterséges intelligencia által biztosított lehetőséggel, meg kellett próbálkozni a kihasználásával. Ajánlást kaptam, hogy létezik egy MI (mesterséges intelligencia), ami képes beolvasni egy teljes PDF kiterjesztésű állományt, feltöltés formájában és arról kérdésekre tud válaszolni. (Forrás: chatpdf.com)

Az első próbálkozás egy alapvetően egyszerű 1 oldalas dokumentumról szólt, amelyben a dolgozat bemutatását írtam röviden. A válaszáda megfelelőnek bizonyult, ezért került sor a nagyobb mennyiségű adat kezelésére, a teljes dolgozat átadása egyetlen állományban.

Az összegzés második próbálkozása a chatpdf.com mesterséges intelligencia segítségével történt, az eredménye sajnálatosan sikertelen próbálkozás, mivel a teljes PDF fájl (a dokumentáció WORDX formátuma átalakítva PDF-be) beolvasása után, arra a kérdésre, hogy összegezze nekem egy oldalban, a kapott fájlt, magyarul, egy pontatlan, nem a témába illő szöveget sikerült produkálnia, amelyet így nem tudtam használni. (vö. 27. ábra)



(27. ábra Mesterséges intelligencia segítség, Forrás: <https://www.chatpdf.com/c/MdKQgxMGA1hMlHgOOGPYw>)

7. Vita és konklúziók

A dolgozat készítése közben felmerülő kérdések közül szeretnék kiemelni egyet, amely befolyásolta a választást az adatbázis tekintetében. A probléma, amely fenn állt, a sajtók táblázatnál jött elő először, amely több forrást megvizsgálva, ahogyan azt említettem a dolgozat adatbázis részében. Az antagonizmus megjelenése ebben a témakörben az attribútumok mennyiségében található, amennyiben sok található attribútumból és kevés objektumból, akkor nem mondható ki, hogy lehet-e másként egyforma. Abban az esetben, ha az objektumok száma több és az attribútumok száma kevesebb, akkor egy hasonló szituáció keletkezik, amely hasonlóan az előző példához, nem ad választ a kérdésre.

8. Összegzés

Az alábbiakban a dolgozat eredményei kerülnek összefoglalásra magyar és angol nyelven (vö. 8.1-8.2).

8.1. HU-magyar Összegzés

Az ODESSAI létrehozása a dietetikus szakértők számára történt meg, ezzel segítve a munkájukat, hogy a pácienseik számára minél gyorsabb megoldást találhassanak a problémáikra és optimalizált adatmennyiséggel tudjanak dolgozni.

Az Online Dietetikus Szakértő egy mesterséges intelligenciával ellátott online szakértőrendszer.

A rendszer célja, hogy segítséget nyújtson az embereknek közvetett módon, a dietetikus számára nyújt segítséget, az egészséges életmód kialakításában és a megfelelő táplálkozás elérésében.

Működése alapján a felhasználók személyre szabott diétás tanácsokat kaphatnak, amelyek figyelembe veszik az egyéni igényeket és a testi adottságokat. Az online szakértőrendszer előnyei közé tartozik, hogy bármikor és bárhol elérhető, és a felhasználók személyes adat megadása nélkül tudják használni a szolgáltatást. Elkészítéséhez szükség volt a PHP, JavaScript, Html, Smarty Template Engine használatára.

Használatba került továbbá a mesterséges intelligencia több formája, mint például a chatGPT, illetve a chatPDF. Az elemzéshez felhasználtuk a MIAÚ által létrehozott és publikussá tett, számításokat elvégző robotot, a COCO-t (Component-based Object Comparison for Objectivity).

Az Online Dietetikus Szakértő használata egyszerű és könnyen áttekinthető és a felhasználók számára számos előnyt kínál, mint például a testsúlycsökkentés, a jobb egészség és a magasabb életminőség előidézését egy szakértő segítségével keresztül. Összességében az Online Dietetikus Szakértő egy hatékony és innovatív megoldás az egészséges életmód eléréséhez és a megfelelő táplálkozás kialakításához egy humán szakértő segítségével.

8.2. EN-English Summary

The thesis is an online dietetic expert system with AI a.k.a ODESSAI. The basic setup for this software, is that a dietician is in need of a database which helps and accelerates the usual workflow, (e.g. getting data from different online datasheets or technical books). This eliminates that part of the process by providing the database needed for the dietician, with detailed food-product databases and the help of the COCO (Component-based Object Comparison for Objectivity) modelling AI, the dietician gets an optimized result based on the given search factors (e.g. cheese products which contain lesser amount of protein and major amount of fat). Other solution would be to simply trust a naive optimalization, which is not mathematically proper nor exact.

The software provides the view of the optimal choice based on the search factors.

ODESSAI runs on an Amazon Web Services server, t3. micro.

Using PHP, Smarty, Html and postgresql with pgadmin4.

This document contains the details which were needed for the thesis, going through each aspect we have encountered during the making of ODESSAI.

9. Mellékletek

A 9. fejezet tartalmazza: kivonat (vö. 9.1), abstract-angol nyelvű összegzés (vö. 9.2), ábrajegyzék (vö. 9.3), hivatkozásjegyzék (vö. 9.4), rövidítésjegyzék (vö. 9.5).

9.1. Kivonat

Az Online Dietetikus Szakértő ötletének kitalálásáról és teljes felépítéséről szól az alábbi dokumentum. Részletezve külön szedett pontokban olvasható a szolgáltatás felépítése, sorrendiségnek megfelelően, időrendben. A szempontok, amikre kitértem a dolgozat írása közben, felsorolásra kerültek indoklással ellátva, illetve a felmerülő problémák, amelyek a fejlesztése közben előkerültek. Mivel a szolgáltatás létrehozásával felmerült a túlságosan nagy méret, szükségünk volt a két rész szétszedésére, ezzel került a szakdolgozat teljes egésze ketté szedésre, Ódor Márk hallgatótársammal együtt készítettem a szolgáltatást.

A kettőnk dolgozatának egymásra való hivatkozása a külön kezelt feladatokat mutatja be, ezzel teljes terméket kínálva, amennyiben az olvasó összeköti a kettőt.

Az élelmiszer-termék szolgáltatás létrehozásának legfőbb szempontja a piaci rés, amely a termék jelenlétének hiányát jelenti a piacon, ezért is történtek meg az alábbi dokumentációban olvasható lépések.

Létrehozásának limitációit firtatva, több nekiindulás történt, ezekre külön kitér a dolgozat, illetve, hogy miként lehetett volna további fejlesztést végezni.

A saját részem leginkább a belső felépítésére fókuszál, míg Márk dolgozata a megjelenést és az ahhoz tartozó szükségleteket helyezi előnybe.

9.2. Abstract

The following document is about the invention and complete structure of the idea of the Online Dietitian Expert. The structure of the service can be read in detail in separate points, according to the sequence, in chronological order. The aspects that I discussed while writing the thesis were listed with justification, as well as the problems that arose during its development. Since the creation of the service was too large, we needed to separate the two parts, thus the entire thesis was divided into two parts. I created the service together with my fellow student Márk Ódor.

The cross-referencing of our two papers shows the tasks handled separately, thus offering a complete product if the reader connects the two.

The main aspect of creating a food product service is the market gap, which means the absence of the product's presence on the market, which is why the steps described in the following documentation were taken.

Examining the limitations of its creation, several initiatives were taken, which are discussed separately in the thesis, as well as how further development could have been carried out.

My own part mostly focuses on its internal structure, while Márk's thesis prioritizes the appearance and the needs associated with it.

9.3. Ábrajegyzék

1. ábra: Futtatható szerver lehetőségek a T3-as csoportban az Amazonnál, Forrás: <https://eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#InstanceTypes>:
2. ábra: AWS konfigurációk, Portok kinyitása, Forrás: <https://eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#SecurityGroup:groupId=sg-00434d76cfadc1a74>
3. ábra Mappa-szerű elérés, Forrás: Winscp-n keresztüli elérése a szervernek
4. ábra vim /etc/apache2/sites-enabled/000-default.conf kiegészítése, 443-as port
5. ábra Postgresql konfigurációs fájl, amelybe a csatlakozásokat és információközléseket lehet nyitni, vagy limitálni. Forrás: /etc/postgresql/15/main/pg_hba.conf szerver mapparendszere
6. ábra PHP bővítmények engedélyezései, Forrás: szerver, /etc/php/8.2/apache2/php.ini
7. ábra. htaccess beltartalma, Forrás: szerver, /var/www/html/.htaccess
8. ábra: Kalóriabázis, “Kalóriatáblázat” nézete, Forrás: <https://kaloriabazis.hu/kaloriatablazat>
9. ábra Index.tpl beltartalma, Forrás: /var/www/html/smarty/templates/index.tpl
10. ábra JavaScript meghívása, Forrás: /var/www/html/smarty/templates/index.tpl
12. ábra Stíluselemek meghívása, Forrás: /var/www/html/smarty/templates/index.tpl
13. ábra Importálás bemutatása, Forrás: /var/www/html/index.php a szerver mapparendszerében
14. ábra fordítás meghívása, Forrás: /var/www/html/smarty/templates
15. ábra részlet a Trans Handler által gyártott hu.yaml fájlból, Forrás: /var/www/html/ a szerver mappa rendszerében
16. ábra Kezdetleges elemzés-bemutatás Forrás: <https://docs.google.com/spreadsheets/d/1XqJrxpjm6h3pVhhXeejOPKWFQd7YlVNf/edit#gid=1990324278>
17. ábra a folyamatábra bemutatása Forrás: saját készítés
18. ábra /OLAP elérhetőségen a megjelenő lehetőségek, Forrás: www.dieteticexpert.hu/OLAP
19. ábra “Mire lenne szüksége egy dietetikusnak?”, Forrás: <https://chat.openai.com/c/5de85f4b-2078-4964-ae9e-c870885a4354>
20. ábra Mértékegység egyesítés, Forrás: <https://chat.openai.com/c/5de85f4b-2078-4964-ae9e-c870885a4354>
21. ábra Optimalizálás kivitelezése, Forrás: <https://chat.openai.com/c/21ce8c0b-d4fa-4f7a-bafb-abd19eac9de2>

22. ábra Hibaelhárítás/Debug, Forrás: <https://chat.openai.com/c/0c125452-40dc-4e5e-a76a-e78c616f2ddf>
23. ábra Kóddal kapcsolatos kérdés, Forrás: <https://chat.openai.com/c/9322952f-4e7b-4993-8115-468bcbea7cca>
24. ábra Kóddal kapcsolatos válasz, Forrás: <https://chat.openai.com/c/9322952f-4e7b-4993-8115-468bcbea7cca>
25. ábra Ötletelő kérdés, Forrás: <https://chat.openai.com/c/a3306933-dc10-4a5e-94d9-d19ea91c15f4>
26. ábra Ötletelő válasz, Forrás: <https://chat.openai.com/c/a3306933-dc10-4a5e-94d9-d19ea91c15f4>
27. ábra Mesterséges intelligencia segítség, Forrás: <https://www.chatpdf.com/c/MdKQgxMGA1hMlHgOOGPYw>

9.4. Hivatkozásjegyzék

Amely forrásoknál nem található „Első kiadása” címpont, abban az esetben a kiadásának dátuma ismeretlen.

<https://www.putty.org/> (Putty, Felhasználva: 2023.11.04, Első kiadása: 1998.11.19)

<https://laravel.com/> (Laravel, Felhasználva: 2023.11.04 Első kiadása:2011.06.09)

<https://kaloriabazis.hu/> (Kalóriabázis, Felhasználva: 2023.11.04, Első kiadása: 2012.06.06)

<https://www.postgresql.org> (Postgresql, Felhasználva: 2023.11.04, Első kiadása: 1995)

https://www.smarty.net/quick_install (Smarty, Felhasználva: 2023.11.04)

<https://fontawesome.com/icons/categories/coding> (Fontawesome, Felhasználva: 2023.11.04)

<https://fontawesome.com/icons/heart-pulse?f=classic&s=solid> (Fontawesome, Felhasználva: 2023.11.04)

<https://docs.google.com/spreadsheets/d/1XqJrxpjm6h3pVhhXeejOPKWFQd7YlVNf/edit#gid=1990324278> (Online Dietetic Expert.xlsx, Szerző: Molnár Csanád, Ódor Márk, Felhasználva: 2023.11.04, Első kiadás: 2023.02.11)

https://miau.my-x.hu/myx-free/coco/engine3_curl.php (COCO cURL meghívásának címe, Felhasználva: 2023.11.04, Szerző(k): a domain mögötti tulajdonos/szerkesztő)

<https://www.kalóriaguru.hu/kaloriatablázat/trappista-sajt-kaloria.php> (Online Táplálkozási Napló, Felhasználva: 2023.11.04, Első kiadása: 2013.10.13, Szerző(k): a domain mögötti tulajdonos/szerkesztő)

https://www.fogyas.info/tapanyag-kaloria-energia-tablázat_kcal-kalkulator.php?#Feta%20sajt (Tápanyagtáblázat – Kalóriatáblázat, Felhasználva: 2023.11.04, Első kiadása 2013.10.01, Szerző(k): a domain mögötti tulajdonos/szerkesztő)

<https://www.google.com/search?q=kal%C3%B3ria+feh%C3%A9j+zs%C3%ADr+sz%C3%A9nhidr%C3%A1t+t%C3%A1bl%C3%A1zat+%3Ctable+%3Ctable%3E> (Google keresési adatok, Felhasználva: 2023.11.04, Szerző(k): Molnár Csanád)

https://hu.wikipedia.org/wiki/Secure_Shell (Secure Shell, Felhasználva: 2023.11.04, Első kiadása: 2006.06.24 Szerző(k): wikipédia szerzői)

https://eu-north-1.console.aws.amazon.com/console/home?nc2=h_ct®ion=eu-north-1&src=header-signin# (Amazon Web Services, Felhasználva: 2023.11.04, Első kiadása: 2002.07)

<https://eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#Home> (Amazon Web Services, Felhasználva: 2023.11.04, Első kiadása: 2002.07)

<https://fdc.nal.usda.gov/> (U.S Department Of Agriculture, Agricultural Research Service, Felhasználva: 2023.11.04, Első kiadása: 2015)

<https://portal.nebih.gov.hu/adatbazisok-elelmiszer> (A Nemzeti Élelmiszerlánc-biztonsági Hivatal Felhasználva: 2023.11.04, Első kiadása: 2012.03.15)

https://miau.my-x.hu/myx-free/coco/beker_y0.php (COCO Y0 modellező algoritmus, Felhasználva: 2023.11.04, Első kiadása: 2012.10.25, Szerző(k): a domain mögötti tulajdonos/szerkesztő)

<https://support.microsoft.com/hu-hu/office/az-olap-online-analytical-processing-%C3%A1ttekint%C3%A9se-15d2cdde-f70b-4277-b009-ed732b75fdd6> (Az OLAP (Online Analytical Processing) áttekintése, Felhasználva: 2023.11.04, Első kiadása: 2006, Szerző(k): Microsoft)

<https://www.fao.org/faostat/en/#compare> (FAOSTAT, Felhasználva: 2023.11.04, Első kiadása: 2011.04, Szerző(k): a domain mögötti tulajdonos/szerkesztő)

https://en.wikipedia.org/wiki/SQL_injection (SQL injection, Felhasználva: 2023.11.04, Első kiadása: 1998, Szerző(k): wikipédia szerzői)

<https://www.inf.u-szeged.hu/~bohus/oktseg.html> (Felhasználva: 2023.11.17, Első kiadása: 1995, Szerző(k): Dr. Bohus Mihály, Szegedi Egyetem)

<https://ftsrc.mit.bme.hu/remo-jegyzet/rendszermodellezes.pdf> (Felhasználva: 2023.11.17, Első kiadása: 2023.06.02, Szerző(k): Bergmann Gábor, Darvas Dániel, Molnár Vince, Szárnyas Gábor, Tóth Tamás)

https://dcs.uni-pannon.hu/files/docs/users/csertangyorgy/swtech/16_UI.pdf, (Felhasználva: 2023.11.17, Első kiadása: Ismeretlen keletkezési idejű, Szerző(k): Dr. Csertán György)

<http://mek.oszk.hu/14300/14321/pdf/14321.pdf>, (Felhasználva: 2023.11.17, Első kiadása: 2013, Szerző(k): Harsáczki András, Nagy Dénes)

https://nki.gov.hu/wp-content/uploads/2019/03/NJSZT_IT_Biztonsag_kozerthetoen_v3.pdf (Felhasználva: 2023.11.17, Első kiadása: 2018.08, Szerző(k): Erdősi Péter Máté, Solymos Ákos)

<https://db.bme.hu/~gajdos/Adatbazisok2019.pdf>, (Felhasználva: 2023.11.17, Első kiadása: 2019, Szerző(k): Gajdos Sándor)

<https://alex.smola.org/drafts/thebook.pdf> (Felhasználva: 2023.11.17, Első kiadása: 2010.10.01, Szerző(k): Alex Smola, S.V.N. Vishwanathan)

<https://users.iit.uni-miskolc.hu/~vadasz/GEIAL302B/GEIAL202-Operacios-rendszerek-jegyzet.pdf> (Felhasználva: 2023.11.17, Első kiadása: 2002.09, Szerző(k): Dr. Vadász Dénes)

http://www.informatom.hu/sze/01/LGB_SZ001/Algoritmusok_%C3%A9s_adatstrukt%C3%BAr%C3%A1k.pdf (Felhasználva: 2023.11.17, Első kiadása: 2006, Szerző(k): Pusztai Pál)

<https://m2.mtmt.hu/gui2/?mode=browse¶ms=publication;34071055>, (Felhasználva: 2023.11.17, Első kiadása: 2023, Szerző(k): Rikk János, Pitlik László)

<https://aries.ektf.hu/~gkusper/SzoftverTeszteles.pdf>, (Felhasználva: 2023.11.17, Első kiadása: ismeretlen keletkezési idejű, Szerző(k): Ficsor Lajos, Dr. Kovács László, Krizsán Zoltán, Dr. Kusper Gábor)

https://people.inf.elte.hu/groberto/elte_kasz/eloadas_anyagok/elte_kasz_ea08_eml.pdf, (Felhasználva: 2023.11.17, Első kiadása: Ismeretlen keletkezési idejű, Szerző(k): Giachetta Roberto)

https://innovativ-tudasvaros-efop361.uni-miskolc.hu/files/1043/Projektmenedzsment_tananyag.pdf, (Felhasználva: 2023.11.17, Első kiadása: 2018 Szerző(k): Tóthné Kiss Anett)

https://nkerepo.uni-nke.hu/xmlui/bitstream/handle/123456789/14286/Innovativ_infokommunikacios_technikak_a_z_elektronikus_kormanyzati_megoldasokban_2019.pdf;jsessionid=43F449A4B6738A469510D7C86432DE6B?sequence=1 (Felhasználva: 2023.11.17, Első kiadása: 2018, Szerző(k): Kő Andrea)

<https://mek.oszk.hu/08200/08200/08200.pdf>, (Felhasználva: 2023.11.17, Első kiadása: 2009, Szerző(k): Szirmai Éva, Újvári Edit)

https://edit.elte.hu/xmlui/bitstream/handle/10831/85759/V%C3%A1llalati%20Gazdas%C3%A1gtan_Holl%C3%B3sy_Kiss-Jegyzet_.pdf?sequence=1, (Felhasználva: 2023.11.17, Első kiadása: 2023, Szerző(k): Hollósy Zsolt, Kiss Livia Benita)

https://nki.gov.hu/wp-content/uploads/2019/03/NJSZT_IT_Biztonsag_kozerthetoen_v3.pdf (Felhasználva: 2023.11.17, Első kiadása: 2018, Szerző(k): Erdősi Péter Máté, Solymos Ákos)

https://www.citatum.hu/konyv/Isten_gondolatai, (Felhasználva: 2023.11.17, Első kiadása: 1983, Szerző(k): Paul Davies).

<https://www.bda.uk.com/about-dietetics/what-is-dietitian/dietitian-or-nutritionist.html> (Felhasználva: 2023.11.17, Első kiadása: 2023, Szerző(k): domain mögötti tulajdonos/szerkesztő)

<https://doi.org/10.1016/j.tifs.2016.08.005> (Felhasználva: 2023.11.17, Első kiadása: 2016.10.01, Szerző(k): Mary Ann Augustin)

<https://extension.psu.edu/selecting-cheese-for-health> (Felhasználva: 2023.11.17, Első kiadása: 2018, Szerző(k): Stacy Reed)

<https://miau.my-x.hu/miau/10/tqm22.htm> (Felhasználva: 2023.11.17, Első kiadása: 1999, Szerző(k): Lorbert Zsuzsanna)

<https://www.rackhost.hu/tudasbazis/informatikai-alapok/hogyan-lehet-egy-jelszot-feltorni/> (Felhasználva: 2023.11.17, Első kiadása: 2011 Szerző(k): a domain mögötti tulajdonos/szerkesztő)

http://real.mtak.hu/164865/1/Taplalasterapias_dois.pdf (Felhasználva: 2023.11.17, Első kiadása: 2023.01, Szerző(k): Dr. Pálfi Erzsébet, Dr. Molnár Andrea, Kovács Andrea, Havasi Anikó, Szűcs Zsuzsanna)

<https://doi.org/10.20311/stat2023.04.hu0354>, (Felhasználva: 2023.11.17, Első kiadása: 2023.04.12, Szerző(k): Bareith Tibor, Fertő Imre)

https://aws.amazon.com/free/?all-free-tier.sort-by=item.additionalFields.SortRank&all-free-tier.sort-order=asc&awsf.Free%20Tier%20Types=*all&awsf.Free%20Tier%20Categories=*all (Amazon Web Services, Felhasználva: 2023.11.17, Első kiadása: 2002.07. Szerző(k): a domain mögötti tulajdonos/szerkesztő)

<https://eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#InstanceTypes> (Amazon Web Services, Felhasználva: 2023.11.17, Első kiadása: 2002.07. Szerző(k): a domain mögötti tulajdonos/szerkesztő)

<https://eu-north-1.console.aws.amazon.com/ec2/home?region=eu-north-1#SecurityGroup:groupId=sg-00434d76cfadc1a74> (Amazon Web Services, Felhasználva: 2023.11.17, Első kiadása: 2002.07. Szerző(k): a domain mögötti tulajdonos/szerkesztő)

<https://www.chatpdf.com/c/MdKQgxMGA1hMIHgooGPyW> (chatPDF, Felhasználva: 2023.11.17)

<https://chat.openai.com/c/5de85f4b-2078-4964-ae9e-c870885a4354> (chatGPT, Felhasználva: 2023.11.17)

<https://chat.openai.com/c/21ce8c0b-d4fa-4f7a-bafb-abd19eac9de2> (chatGPT, Felhasználva: 2023.11.17)

<https://chat.openai.com/c/0c125452-40dc-4e5e-a76a-e78c616f2ddf> (chatGPT, Felhasználva: 2023.11.17)

<https://chat.openai.com/c/9322952f-4e7b-4993-8115-468bcbea7cca> (chatGPT, Felhasználva: 2023.11.17)

<https://chat.openai.com/c/a3306933-dc10-4a5e-94d9-d19ea91c15f4> (chatGPT, Felhasználva: 2023.11.17)

9.5. Rövidítések jegyzéke

MIAÚ (Magyar Internetes Alkalmazott/Agrár Informatikai Újság)

COCO (Component-based Object Comparison for Objectivity)

MI (Mesterséges intelligencia)

SSH (Secure Shell) → Egy szabványcsalád, és egyben egy protokoll is, amit egy helyi és egy távoli számítógép közötti biztonságos csatorna kiépítésére fejlesztettek ki.

USDA (United States Department of Agriculture) → USA mezőgazdasági minisztériuma

NÉBIH (Nemzeti Élelmiszerlánc-biztonsági Hivatal) → A magyar élelmiszerlánc biztonságának felügyeletéért felelős országos hatáskörű állami szervezet.

PUTTY → Egy szabad és nyílt forráskódú terminálemulátor, soros konzol és hálózati fájlátviteli alkalmazás.

OLAP (Online Analytical Processing) → Egy olyan technológia, amely nagyvállalati adatbázisok rendszerezését és az üzleti intelligencia támogatását használja.

AWS (Amazon Web Services) → Amazon.com által az interneten kínált szolgáltatások gyűjteménye, amelyek egy felhő alapú platformot alkotnak.

FAO (Food and Agricultural Organization of the United Nations) → Az Egyesült Nemzetek Szervezetének Élelmezésügyi és Mezőgazdasági Szervezete az ENSZ szakosított szervezete, amelynek fő feladata az élelmezésbiztonság megvalósítása.

PDF (Portable Document Format) → Egy hordozható, kompakt fájlformátum, amely lehetőséget nyújt teljes dokumentumok kisebbé tételére (tömörít)

ODESSAI (Online Dietetic Expert System with AI) → Online Dietetikus Szakértő mesterséges intelligenciával

ChatGPT(Generative Pre-trained Transformer) → A ChatGPT az OpenAI mesterséges intelligencia kutató laboratórium által kifejlesztett chatbot, mely a felhasználókkal való folyamatos kommunikáció automatizálása során értelmezőmodelleket használ, melyek segítségével a bevitt információkat azonnal interaktívan kezeli.

SQL → Structured Query Language (strukturált lekérdezőnyelv) relációsadatbázis-kezelők lekérdezési nyelve. Angol nyelvterületen 'eszjkuel' a kiejtése.

SSL mindkét oldalról (Secure Sockets Layer, titkosított adatfolyam, adatvédelem).