**Baranya Megyei SZC Simonyi Károly Technikum**

**és Szakképző Iskola**

Vizsgaremek

|  |  |
| --- | --- |
| Készítet-  TÉK: | ***Hermann Máté***  ***Gál Martin***  ***Pancza Milán*** |

**PÉCS 2025**

*Baranya Megyei SZC Simonyi Károly Technikum és Szakképző Iskola*

*Szakma megnevezése: Szoftverfejlesztő és –tesztelő*

*A szakma azonosító száma: 5 0613 12 03*

***Záródolgozat***

***Cím: Webshop Hálózata***

*Készítette: Név:* Hermann Máté

*Név:* Gál Martin

*Név:* Pancza Milán

*Pécs 2025*

***Nyilatkozat***

Aluírott, büntetőjogi felelősségem tudatában nyilatkozom és aláírásommal igazolom, hogy a benyújtott záródolgozatom saját, önálló munkám. Az abban hivatkozott nyomtatott és elektronikus szakirodalom felhasználása a szerzői jogok szabályainak megfele- lően készült. Tudomásul veszem, hogy záródolgozat esetén plágiumnak számít:

* szószerinti, vagy attól kismértékben eltérő idézet közlése idézőjel és hivatkozás megjelölése nélkül.
* tartalmi idézet hivatkozás megjelölése nélkül
* más publikált gondolatainak (cikk, dolgozat) sajátomként való feltünte-

tése.

Alulírott kijelentem, hogy a plágium fogalmát megismertem, és tudomásul veszem, hogy plágium esetén záródolgozatom visszautasításra kerül.

Kelt.: Pécs, 2025

................................................

Tanuló

................................................

Tanuló

................................................

Tanuló

Tartalomjegyzék

[1. PROJEKT BEMUTATÁSA 5](#_Toc194321972)

[2. Figma Desing Látványterv 7](#_Toc194321973)

[3. Adatbázis 8](#_Toc194321974)

[4. Backend 10](#_Toc194321975)

[5. Postman 11](#_Toc194321976)

[6. Frontend 13](#_Toc194321977)

[7. Angular 19](#_Toc194321978)

[8. Chatgpt/Mesterséges Intelligencia 23](#_Toc194321979)

# PROJEKT BEMUTATÁSA

A projekt a következő publikus GitHub linken tekinthető meg:

<https://github.com/hermannmate420/ProjectFeladat>

A vizsgaremekünk témája egy weboldalt hoz létre, ami a Retro Vintage fantázianévre hallgat. Ez a weboldal arra szolgál a kliens megtekintők számára, hogy igazi ereklyéket, „kincseket” tudjanak vásárolni interneten keresztül. Oldalunk lehetőséget ad különböző termékek megszerzésére, vásárlására, emellett más gyűjtőkkel való beszélgetésre, esetleges adok-kapok cserék lebonyolítására.

Kifejezetten hasonló projektet nem találtunk az adatok gyűjtése közben, így úgymond minimálisan eredeti ötlettel állhattunk elő.

A weboldal ugyebár 4 részből áll össze.

1. Első körben egy Figma design-t hoztunk létre az általános színek kiválasztására, a megadott terv elrendezésére, és a hivatkozások megtervezésére, annak érdekében hogy tudjuk, melyik gomb melyik oldalhoz fog kötődni.
2. Tartalmaz egy adatbázist, amelyet a MAMP nevű programban és PHPMyAdmin-ban hoztunk létre.
3. Készült egy Backend felület ami a Java programban készült el.
4. Tartalmaz egy Frontend-et, ami első körben Visual Studio Code-ban készült el mint alap vázlat. Majd ezt követően az Angular keretrendszert használtuk fel a teljes projekt külső megjelenítésére és tesztelésére.

A csapatmunka bemutatása:

* + - 1. Hermann Máté – Gál Martin:
         1. Elkészítették az adatbázist, töltötték fel tárolt eljárásokkal, emellett töltötték fel adatokkal.
      2. Hermann Máté:
         1. Készítette el a Backend kódot a Java programban. Például a Login-hoz tartozó kódot is.
      3. Gál Martin – Pancza Milán:
         1. Elkészítették a Frontend kódot első körben Visual Studio Code-ban majd pedig annak mintájára Angular keretrendszert használva fejlesztették tovább, alakították át a desing-t.
      4. Pancza Milán:
         1. Készítette el a dokumentációt, emellett pedig a Figma programot felhasználva készítette el a látványtervet.

# Figma Desing Látványterv

A **Figma** egy népszerű online design és prototípus-készítő eszköz, amelyet főként UI/UX tervezéshez használnak.

* **Mire jó?**
  + - Web- és mobilalkalmazások felhasználói felületének tervezésére
    - Prototípusok és interaktív modellek készítésére
    - Együttműködésre a csapatok között valós időben
* **Mire használható?**
  + - Grafikai tervezésre és vektorgrafikák létrehozására
    - Drótvázak (wireframe-ek) és vizuális koncepciók készítésére
    - Design rendszerek és UI komponensek kezelésére

Fő előnye, hogy böngészőből fut, így nincs szükség telepítésre, és könnyen megoszthatók a projektek másokkal.

A képen képernyőkép, szöveg, Multimédiás szoftver, szoftver látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

1. ábra: Figma Desing látványterv

A színpaletta kiválasztásakor próbáltunk olyan színeket keresni amik többé kevésbé ténylegen a Retro stílust hozzák vissza a klienseink számára. Az általunk választott paletta például inkább a ’70-es éveket idézi a színeivel.

A képen képernyőkép, pálmafa, szöveg, Multimédiás szoftver látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

2. ábra: Figma Desing szín paletta

# Adatbázis

Az adatbáziskezelésre mi a MAMP-ot használtuk. Ennek a programnak az előnyeit, és hogy mire használjuk itt olvashatják:

A **MAMP** egy lokális szerverkörnyezet, amely segítségével fejlesztők könnyen futtathatnak és tesztelhetnek PHP-alapú weboldalakat a saját gépükön.

* **Mire jó?**
  + Apache, MySQL és PHP egyetlen csomagban történő telepítésére
  + Webfejlesztési projektek tesztelésére internetkapcsolat nélkül
* **Mire használható?**
  + WordPress és más PHP-alapú rendszerek futtatására
  + Adatbázis-kezelésre a **phpMyAdmin** segítségével
  + Webalkalmazások fejlesztésére és hibakeresésére

A **phpMyAdmin** egy böngészőalapú eszköz, amely megkönnyíti a MySQL-adatbázisok kezelését grafikus felületen keresztül.

A következő ábrán látható lesz, hogy milyen táblákat hoztunk létre a weboldalhoz.

A képen szöveg, képernyőkép, Betűtípus, szám látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

3. ábra:Az adatbázisban elkészült táblák

Mint ahogy a képen is látható, létrehoztunk összesen 8 db táblát, név szerint (kategória, rendelés, rendelt termékek, fizetés, termékek, szállítási cím, teszt visszajelzés, felhasználók)

Ezeken a táblákon felül a Procedures alatt létrehoztunk tárolt eljárásokat is.

A képen szöveg, képernyőkép, szám, Betűtípus látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

4. ábra: Tárolt eljárások

Ezeken felül pedig feltöltöttük a tábláinkat adatokkal, amiket később a weboldalon meg fogunk jeleníteni. A projekt haladásával a képen látható termékek bővültek, ennek megtörténéséhez igénybe vettük a mesterséges intelligenciát, ami mindegyik kategóriánkhoz generált számunkra 10-10 terméket. Ennek következményeképpen a termékeink száma több mint 200-ra nőtt.

A képen szöveg, képernyőkép, szám, Betűtípus látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

5. ábra: Kategóriák tábla feltöltött formában

# Backend

# Postman

**Mi az a Postman?**

A **Postman** egy fejlesztői eszköz, amit elsősorban API-k (alkalmazásprogramozási interfészek) tesztelésére, dokumentálására és fejlesztésére használnak. Különösen népszerű backend-fejlesztők, tesztelők, DevOps szakemberek és szoftverfejlesztők körében.

**Előnyei**

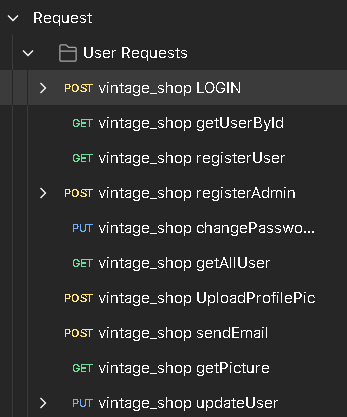
1. **Felhasználóbarát felület**: Könnyen átlátható GUI, még azoknak is, akik nem szuper jártasak az API-k világában.
2. **Könnyű API-tesztelés**: Egyszerűen lehet GET, POST, PUT, DELETE stb. kéréseket küldeni, megnézni a válaszokat és hibakeresni.
3. **Környezeti változók kezelése**: Többféle környezetet (pl. teszt, staging, éles) lehet definiálni változókkal.
4. **Automatizálható tesztek**: JavaScript-ben írt teszt szkriptekkel ellenőrizhető, hogy az API válasza megfelelő-e.
5. **Csapatmunka támogatása**: Megoszthatók a kollekciók, dokumentációk és workspace-ek egy csapaton belül.
6. **Mock szerverek létrehozása**: Tesztelhető egy API akkor is, ha a backend még nem készült el.
7. **Dokumentáció generálás**: Automatikusan generálható és frissíthető API dokumentáció.

**Hátrányai**

1. **Erőforrásigényes lehet**: Főleg nagy kollekciók vagy gyenge gépeken lassulhat.
2. **Ingyenes verzió korlátai**: Néhány fejlettebb funkció (pl. együttműködés, analitika) csak a fizetős verzióban érhető el.
3. **Nagy tanulási görbe haladó szinten**: A basic funkciók egyszerűek, de a pre-request szkriptek, chaining és komplex környezetek időigényesebbek lehetnek.
4. **Offline használat korlátozott**: Egyes funkciók internetkapcsolatot igényelnek, főleg csapatmunkánál.

**Mire használják az emberek?**

* **API-k tesztelése**: Kérések küldése REST, SOAP vagy GraphQL API-khoz.
* **Fejlesztés közbeni ellenőrzés**: Backend endpointok tesztelése frontend implementáció előtt.
* **Automatizált tesztek futtatása**: Validálás, hogy egy API mindig a kívánt formátumban válaszol.
* **API dokumentáció készítése és megosztása**: Könnyen olvasható és megosztható doksik a csapat számára.
* **Mock API szerverek futtatása**: Hasznos, ha a kliens fejlesztése párhuzamosan történik a backenddel.
* **Monitoring**: Bizonyos időközönként automatikus lekérdezések küldése API-khoz.



6. ábra: A Postman-ben készült kérések

A képen látható kérések tartalmazzák a weboldalunk regisztrációját, bejelentkezését, jelszó változtatás lehetőségét, a profil adatok megváltoztatásának lehetőségét, a regisztráció utáni e-mail küldést.

# Frontend

El is érkeztünk a Frontendhez, ezt első körben normál Visual Studio Code-ban fejlesztettük le, bármiféle fajta keretrendszer használata nélkül. Majd ezt követően vittük át az Angular keretrendszerbe, és kezdtük el a korábbi kód mintájára alakítani a weboldalt.

Egy pár szó a VS Code-ról:

A **Visual Studio Code (VS Code)** egy ingyenes, könnyű, mégis erőteljes forráskódszerkesztő, amelyet a Microsoft fejlesztett.

* **Mire jó?**
  + Különböző programozási nyelvekben való fejlesztésre (pl. JavaScript, Python, PHP, C++ stb.).
  + Kódszerkesztésre, hibakeresésre és verziókezelésre.
  + Bővítményekkel testre szabható fejlesztői környezet létrehozására.
* **Előnyei:**
  + Ingyenes és platformfüggetlen (Windows, macOS, Linux).
  + Rengeteg bővítmény érhető el hozzá.
  + Beépített terminál és Git-integráció.
  + Intelligens kódkiegészítés (IntelliSense).
* **Hátrányai:**
  + Nagy projektek esetén több memóriát fogyaszthat.
  + Néha lassabb lehet a sok bővítmény miatt.

Főként web- és szoftverfejlesztők használják, mert gyors, rugalmas és könnyen testre szabható.

Ezt követően, essen egy pár szó az Angular keretrendszerről is.

Elsődleges feladatnak tartottuk, hogy ne azonnal Angularban fejlesszünk mert amikor a projektet elkezdtük, még éppen csak láttunk egy keveset az Angular-ból, így jött az alternatíva, hogy kódoljunk normál VS Code-ban. Ennek az alap szerkezetét láthatjuk a következő ábrán:

A képen szöveg, képernyőkép, szoftver látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

7. ábra: VS Code alapszerkezete

Ebben az alap szerkezetben/vázban létrehoztunk egy kezdetleges felületet, ahol mind az elrendezést, mind a színek párosítását meg tudtuk oldani. Első körben elkészültek a megadott oldalak mappái, majd azon belül mindegyik kapott egy index.html-t és egy css fájlt. Ezek ugye arra szükségesek, hogy meg tudjuk írni a kódot majd pedig képesek legyünk formázni, design-olni őket. Egy két mappa kapott 1-1 javascript fájlt is. Ennek az oka az volt, hogy tartalmaz valamit az oldal, amihez elengedhetetlen volt egy minimális javascript kód megírása.

Ezt követően megkezdődött a tényleges kód írása.

A képen szöveg, képernyőkép, szoftver látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

8. ábra: A Login page kódolt formája index.html-ben

Itt pedig a kezdetlegesen, de elkészült oldalt láthatjuk.

A képen képernyőkép látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

9. ábra: Az elkészült Login page VS Code-ban

Mivel ez egy teljesen kezdetleges alapnak készült, ezért nem is fektettünk bele hatalmas nagy energiát. Ellenben mondjuk a Home page-el.

Első részben létrehoztunk egy navigációs sávot (nav-bar-t) ami mindegyik oldalon megegyezik.



10. ábra: Navigációs sáv

Ehhez a Bootsrap 5.0-ás verzióját használtuk fel először, majd pedig cseréltük ki a saját magunk által megadott nevek gombjaira.

A képen szöveg, képernyőkép látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

11. ábra: A navigációs sáv kódja

Annak érdekében, hogy a nav-bar úgy nézzen ki és működjön egy ilyen kódot kellett létrehozni.

Ezt követte a body, ami a weboldal központját alkotja, ebben elhelyeztünk carouselt is, kiemeltünk egy pár termékcsaládot amik a legkellendőbbek, ellenben mindent beégetett képekkel és szöveggel oldottunk meg.

A képen szöveg, Grafikus tervezés, képernyőkép, poszter látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

12. ábra: A Home Body kezdetleges kinézete

Az itt lévő Carouselekhez kellett a js fájl ugyanis a termékek a megadott időkorláton túl automatikusan váltják magukat. Ez mindhárom kártyán 5 másodperc, emellett pedig 5-5 termék váltja egymást.

A képen szöveg, képernyőkép, képernyő, Betűtípus látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

13. ábra: A Carouselek automatikus váltása

Ezeken felül, minden oldal kapott egy Footert ahol egy pár szóban a készítőkről és a információkról, kapcsolattartásról stb.

A képen szöveg, képernyőkép, szoftver, Betűtípus látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

14. ábra: A Footer kialakítása, elrendezése

A képen szöveg, képernyőkép látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

15. ábra: A Footer kódolása

# Angular

A **Angular** egy népszerű, nyílt forráskódú JavaScript-alapú keretrendszer, amelyet a Google fejlesztett.

* **Mire jó?**
  + Nagy és összetett webalkalmazások fejlesztésére.
  + Egyoldalas alkalmazások (**SPA - Single Page Application**) készítésére.
  + Dinamikus, reszponzív és moduláris weboldalak létrehozására.
* **Előnyei:**
  + Strukturált és jól szervezett fejlesztési környezet.
  + Kétirányú adatkapcsolat (**two-way data binding**), amely megkönnyíti az adatok

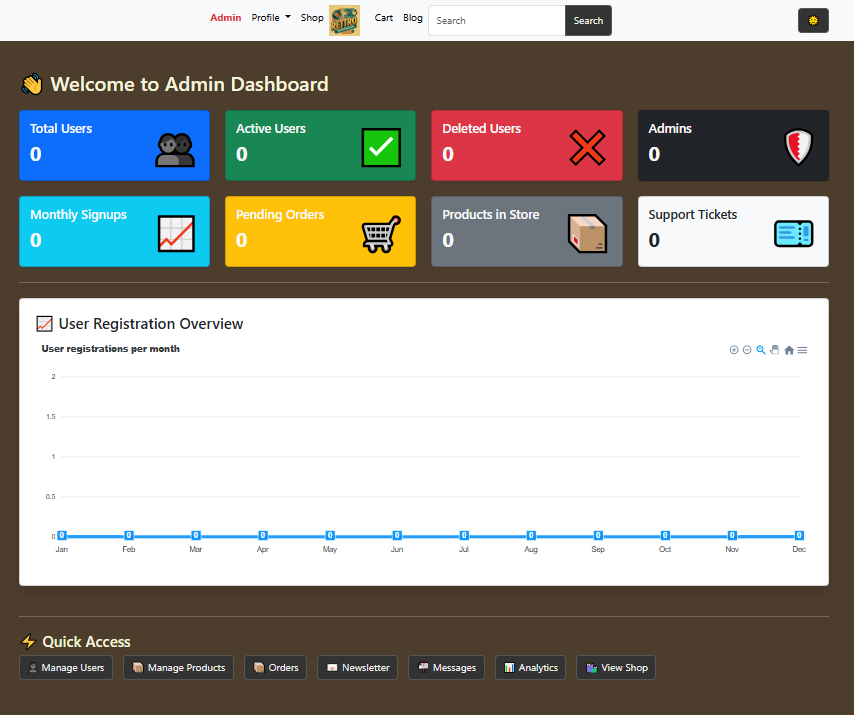
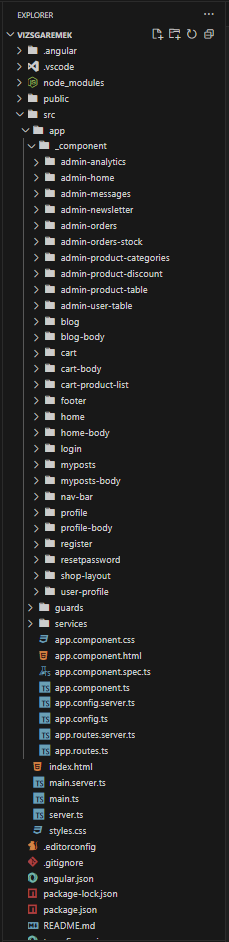
kezelését.

* + Beépített eszközök és funkcionalitások (pl. Dependency Injection, Routing).
  + Nagy közösségi támogatás és hosszú távú Google-támogatás.
* **Hátrányai:**
  + Meredek tanulási görbe a komplexitása miatt.
  + Nagyobb teljesítményigényű, mint más könnyebb keretrendszerek (pl. React, Vue).

Az Angular főként nagyobb vállalati szintű projektekhez ideális, ahol a skálázhatóság és a robusztus szerkezet kiemelten fontos.

Miután a projekt már „kinőtte” a normál VS Code-ot és a tanulmányaink is elérték a megfelelő szintet, elkezdtük az Angulart használni a projekthez mint megadott keretrendszer. Első körben létrehoztuk a megfelelő komponenseket amikre szükségünk volt ahhoz, hogy minden oldal megfelelően átlátható legyen, annak érdekében, hogy ne keveredjünk benne össze, és ne vétsünk olyan hibát amit a későbbiekben esetleg nem tudunk korrigálni.

16. ábra: Az Angularban létrehozott mappaszerkezet, és az Adminnak létrehozott funkciók



Az oldalon megjelenített képen látható a mappa szerkezetünk, ami az alap, kezdő Angular mappákon és fájlokon kívül lettek elkészítve. Ezeket mind egy főbb mappába helyeztük el szintén az átláthatóság kedvéért, amit jelen helyzetben \_component néven mentettünk ugyanis a benne elhelyezett további mappák és fájlok mind különálló komponenseket tartalmaznak.

A képen szöveg, képernyőkép, Betűtípus látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.Nagy energiát és fejlesztést szenteltünk az Admin oldalunknak annak érdekében, hogy az üzemeltető ténylegesen mindent képes legyen koordinálni, ne lehessen megkerülni, minden történés az ő fennhatósága alatt történhessen.

Sikeresen készítettünk valós idejű statisztikát a regisztrációkról, emellett egy jó pár lehetőséggel bővítettük az Admin funkciókat is.

A képen szöveg, képernyőkép, szoftver, Számítógépes ikon látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

17. ábra: Az Admin oldalon követhető statisztikák

Elkészült a normál profilokhoz tartozó change password lehetőség is annak érdekében, ha valaki szeretné megváltoztatni a jelszavát, emellett a teljes profilban lévő adatokat is képes a felhasználó megváltoztatni azért, hogyha esetleg a regisztrációkor valamit hibásan adott volna meg, vagy esetleg a későbbiekben bármi okból kifolyólag szeretne rajta változtatni.

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

18. ábra: A profil adatok megváltoztatása

A képen szöveg, képernyőkép, szoftver, Multimédiás szoftver látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

19. ábra: A jelszó változtatás

# Chatgpt/Mesterséges Intelligencia

A mesterséges intelligencia használata, előnyei, hátrányai:

* Használata:
  + **Írás**: Esszék, e-mailek, novellák, versek, hivatalos dokumentumok írására vagy szerkesztésére.
  + **Programozás**: Kódírás, hibakeresés, algoritmusok megértése.
  + **Tanulás**: Fogalmak elmagyarázása, vizsgafelkészülés, jegyzetelés.
  + **Kreatív ötletelés**: Karakternevek, sztorik, kampányötletek, játékötletek.
  + **Nyelvi segítség**: Fordítás, nyelvtani magyarázat, szókincsfejlesztés.
  + **Mindennapi kérdések**: Tippek életmódhoz, utazáshoz, receptek, ajándékötletek.
  + **Szórakozás**: Viccek, találós kérdések, játékötletek, akár szerepjáték is.
* Előnyei:
  + **Gyors és rendelkezésre állok**: Napi 24 órában elérhető vagyok, nincs alvás vagy ebédszünet.
  + **Sokféle témában tudok segíteni**: Legyen szó írástechnikáról, kódolásról, receptötletekről vagy akár érvelésről egy esszéhez – sok mindenhez értek.
  + **Nyelvtudás**: Több nyelven is kommunikálok, köztük magyarul is elég jól.
  + **Nem ítélkezem**: Bármit kérdezhetsz, nem foglak elítélni vagy kinevetni.
  + **Kreatív vagyok**: Tudok verset írni, mesét kitalálni, vicceket mondani, vagy akár neveket adni karaktereidnek.
  + **Rugalmas stílusban írok**: Lehetek formális, laza, humoros, érzelmes – ahogy szeretnéd.
* Hátrányai:
  + **Nem vagyok ember**: Hiányzik az emberi tapasztalat, empátia mélysége, vagy az igazi "megérzés".
  + **Nem tévedhetetlen**: Néha pontatlan vagy elavult információt adhatok, főleg ha friss hírekről van szó.
  + **Nem vagyok tudatában semminek**: Nem emlékszem rád automatikusan, hacsak te nem engedélyezed a memóriát a beállításokban.
  + **Nem tudok hozzáférni mindenhez**: Nem látok rá privát adatbázisokra, zárt weboldalakra, vagy titkos információkra.
  + **Néha túl „okoskodó” vagy „általános”**: Egyes válaszaim túl általánosak lehetnek, ha nem kapok elég konkrét kontextust.

A mesterséges intelligenciát, avagy a konkrétan a ChatGpt-t főként a problémáink megoldásához, avagy az újabb ismeretek gyűjtéséhez használtuk. Ezen felül segítségünkre volt az adatbázisunk feltöltésében is ahol termékeket generált számunkra, majd a későbbiekben a termékekhez képet is generáltunk vele.

Mint ahogy a képen is látható, a mi általunk megadott prompt-ok alapján generált számunkra megfelelő termékeket:

A képen szöveg, elektronika, képernyőkép, szoftver látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

20. ábra: Kérésünkre kért termékek generálása

Majd miután minden terméket legenerált, kértük hogy készítsen hozzájuk képeket is annak érdekében, hogy a termékeknek legyen látszata is, ne pedig csak egy rövid leírást kapjanak a „vevők”.

A képen óra, karóra, Zsebóra látható

Előfordulhat, hogy a mesterséges intelligencia által létrehozott tartalom helytelen.

21. ábra: A megadott termékekhez generált képek