**Dokumentáció**

Fejlesztők: Molnár Csenge Anna, Berkovics Gellért

Berkovics Gellért: Asztal és mobil alkalmazás fejlesztés

Molnár Csenge Anna: Frontend tesztelés

Jegyzet:

**Projekt célja**

A hallássérülés meghatározása, kategorizációja

Hallássérülésnek nevezzük a hallószerv valamely részének veleszületett vagy szerzett sérülését, illetve fejlődési rendellenességét, amelynek az éptől eltérő hallásteljesítmény lesza következménye. Hallássérülés esetén a hallásküszöb megemelkedik, a személy csak az erősebb hangokat hallja meg – vagy még azokat sem hallja. Ezt nevezzük hallásveszteségnek. A hallássérülés gyűjtőfogalom. Hallássérült személyek a nagyothallók, a siketek és a cochleáris implantátumot (CI)1 viselők. A beszéd felfogása szempontjából a legfontosabb szerepeta 250 és 4000 Hz közötti hangok érzékelése tölti be (FRERICHS, 1998). A hallássérülés orvosi szempontú meghatározása a gyógypedagógiai megközelítésnél jóval tágabb. Orvosi értelemben hallássérültnek tekinthetők mindazok, akiknek hallásteljesítménye az ép hallástól eltérést mutat. Az egyoldali enyhe fokú hallásveszteség, a kétoldali kismértékű hallásveszteség, a csak az igen magas frekvenciákat érintő hallássérülés, az átmeneti hallásromlás, illetve az időskori halláscsökkenés orvosi értelemben igen, azonban gyógypedagógiai értelemben nem tartoznak a hallási fogyatékosság körébe, mivel nem vagy csak kevéssé érinti a beszédértés, beszédelsajátítás területét. „A hallássérülés gyógypedagógiai fogalma (hallási fogyatékosság) elsősorban a beszédértéshez szükséges hallásterületen közepes vagy annál súlyosabb fokú nagyothallást, siketséggel határos vagy siketségnek diagnosztizált hallásveszteséget jelent. Más megközelítésben a hallássérültek pedagógiája a hallássérült kifejezést olyan halláscsökkenésre alkalmazza, amelynek következményeként a beszédfejlődés nem indul meg, vagy a beszéd oly mértékben sérült, hogy a beszéd megindításához, korrekciójához speciális beszédfejlesztő módszerek alkalmazására van szükség.” (FARKAS és PERLUSZ, 2000) A hangerő csökkent érzékelése szempontjából a következő kategóriák állíthatók fel a jobbik fülön főbb beszédfrekvenciákon mért átlagos hallásveszteséget alapul véve:

• enyhe fokú nagyothallás: 25–40 dB,

• közepes fokú nagyothallás: 40–60,

• súlyos fokú nagyothallás: 60–90 dB,

• átmeneti sáv a súlyos nagyothallás és a siketség között: 90–110 dB,

• siketség: 110 dB felett. (PYTEL, 1996)

Nem funkcionális:

* Angular CLI: 17.2.2
* Angular Material
* Node: 18.18.
* Bootstrap 5
* Bootstrap, 5.3 verzió

**Felhasznált technológiák**

Angular

Bootstrap CSS

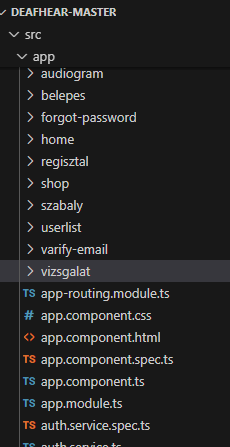
Typescrip

http

Nodejs

Git / GitHub

* Komponensek, funkciók mappákba rendezve funkcionalitás alapján
* Services az implementálásához
* Pipe-ok az adatok formázásához
* Segédfunkciók a közös műveletek elvégzéséhez
* A Firebase konfigurációt és szolgáltatásokat külön modulokba szervezzük az átláthatóság érdekében

/DeafHearShop

/src

/app

/audiogram

/belepes

/forgot-password

/home

/regiszter

/shop

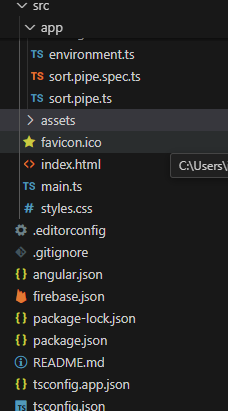
/szabaly(audiogram2)

/userlist

/viszgalat

app.module.ts

app-routing.module.ts

 /assets

/image

/hang

A következő **vizuális komponensek** lettek létrehozva:

* app.component
* nav.component – menüsor, navigációs mező
* home.component – a termékek megjelenítésére szolgáló componens
* információk, illetve a kapcsolatfelvétel érhető el
* register.component - beléptető felület
* user-sign-up.component – regisztrációs felület
* forgotpassword.component- Elfelejtett jelszó igénylésének űrlapja
* verifyemail.component – Tájékoztató felület az elfelejtett jelszó linkjének elküldéséről

A regisztrációnál kötelező megadni a felhasználónevet, e-mail címet, jelszót és további releváns adatokat (például teljes név, telefonszámokt).

* A jelszónak legalább 6 karakter hosszúnak kell lennie, valamint tartalmaznia kell kisbetűt, valamint számok.
* Időpont foglaláshoz bejelentkezés kötelező.
* A galériába való feltöltéshez bejelentkezés kötelező.

**Frontend**

**Molnár Csenge Anna**

Bejelentkezési űrlap (HTML):

* Az <h1> címsor a bejelentkezés oldalának címét jeleníti meg.
* Egy űrlap található, amelyben a felhasználó megadhatja az e-mail címét és a jelszavát.
* Az e-mail cím és jelszó mezők kötelezőek, és megjelenik egy hibaüzenet, ha nem megfelelő formátumban vannak.
* Egy gomb, amely lehetővé teszi az e-mail cím megerősítését.
* Gombok a bejelentkezés, regisztráció és visszalépés funkcióhoz.

Sign up:

Ezek a komponensek lehetővé teszik az új felhasználók regisztrációját az alkalmazásba, ellenőrzik az űrlap kitöltését, és megjelenítik a megfelelő üzeneteket a regisztráció sikerességéről vagy hibáiról.

Regisztrációs űrlap (HTML):

* Egy <h1> címsor jelzi a regisztrációs oldal címét.
* Négy mezőt tartalmaz: teljes név, telefonszám, Lakcím, e-mail cím és jelszó.
* Az e-mail cím, jelszó mezők kötelezőek, és megjelenik egy hibaüzenet, ha a felhasználó nem megfelelően tölti ki ezeket a mezőket.
* Az e-mail cím mező mellett egy kis szöveges üzenet jelenik meg, ha az e-mail cím formátuma nem megfelelő.
* A regisztráció gomb aktív, ha az összes kötelező mező megfelelően van kitöltve.

Regisztrációs komponens (TypeScript):

* Az osztály felelős az új felhasználók regisztrációjáért, az űrlap validálásáért és a szükséges információk kezeléséért..
* Az függvény ellenőrzi, hogy az összes kötelező mező megfelelően van-e kitöltve.
* register() függvény kezeli az űrlap beküldését, elküldi az adatokat a szervernek, és navigál a megfelelő oldalra a regisztráció sikeressége esetén.
* register() függvény megjeleníti a felugró üzeneteket sikeres vagy sikertelen regisztráció esetén.

Verify email:

Ez a komponens lehetővé teszi a felhasználók számára, hogy megerősítsék az e-mail címüket azáltal, hogy megadják a kapott megerősítő kódot, és ezáltal bejelentkezzenek a rendszerbe. A megfelelő visszajelzések megjelenítése segíti a felhasználókat a folyamat során.

HTML kód:

* Egy űrlapot jelenít meg, amely egy megerősítő kód bevitelére szolgál.
* Az input mezőt megfelelő címkével látjuk el, amely leírja, mit kell megadni az input mezőbe.
* A "Küldés" gombra kattintva az űrlapot elküldi, ami a verifyEmail() függvényt hívja meg.

TypeScript kód:

* Az osztály egy code változót tartalmaz, amelybe a felhasználó által megadott megerősítő kódot tároljuk.
* Az email és password változók a bejelentkezéshez használt adatokat tárolják, amelyeket a komponens az URL paramétereiből kap meg.
* A verifyEmail() metódus meghívja az auth.verifyEmail() függvényt, hogy ellenőrizze a megerősítő kódot. Ha az ellenőrzés sikeres, a felhasználót bejelentkezteti a login() függvény segítségével.
* A belepes() metódus bejelentkezteti a felhasználót az auth.belepes() függvény segítségével. Ha a bejelentkezés sikeres, akkor megjeleníti a sikeres üzenetet és átirányítja a felhasználót a főoldalra (/home), ellenkező esetben hibaüzenetet jelenít meg.

A Komponensek az alábbiak:

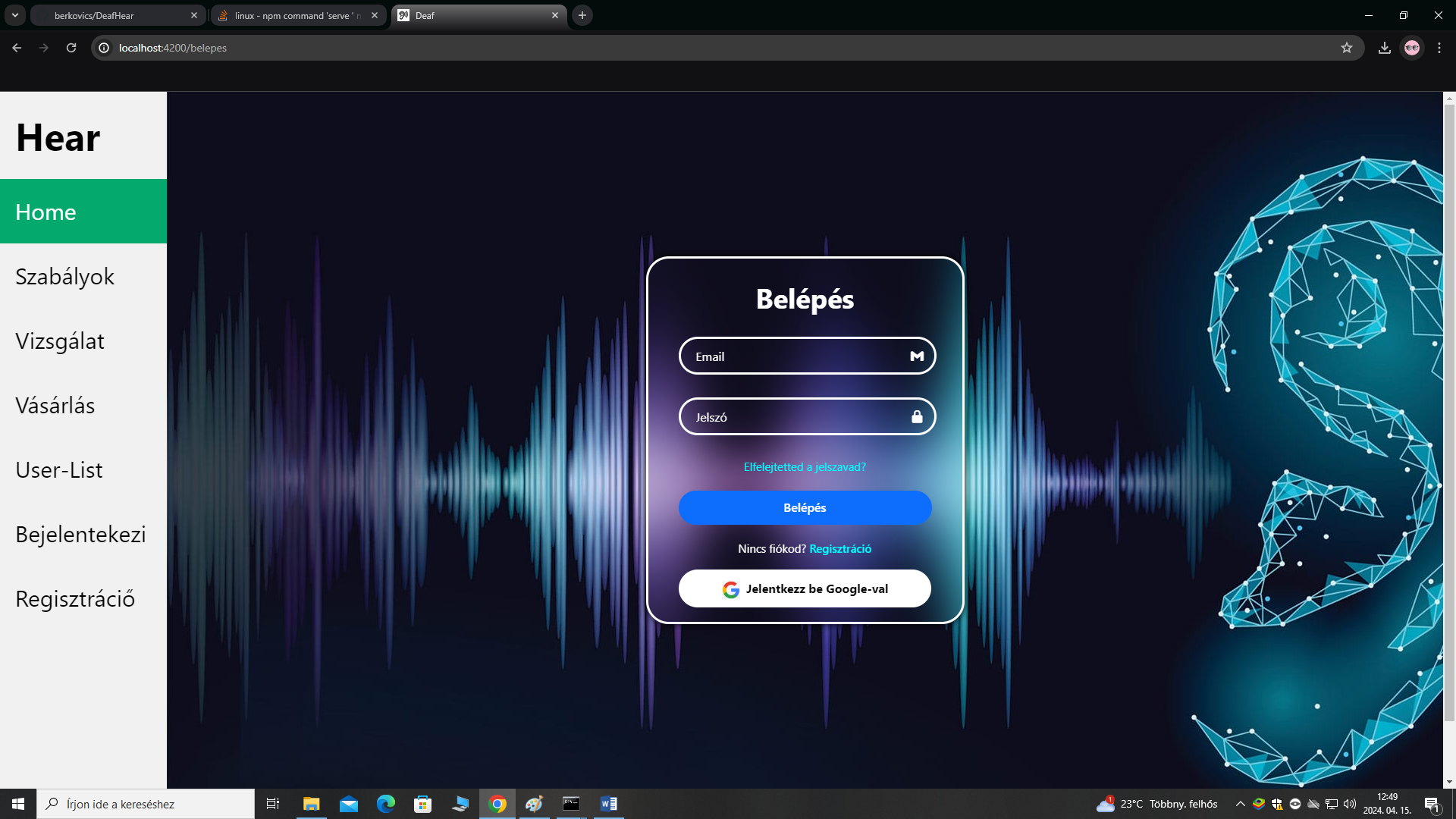
* Belepes
* Register
* Audiogram
* Vizsgalat
* Shop

Végül a Szolgáltatások litsája:

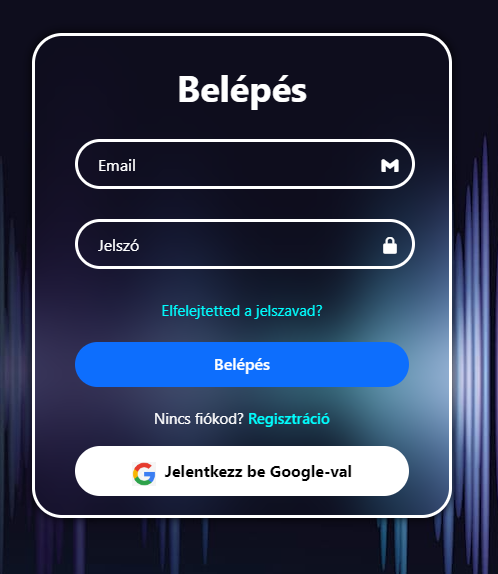
* auth.service:
  + A felhasználó beléptetését végzi, és tárolja el a böngésző lokális tárolójában.
* config.service

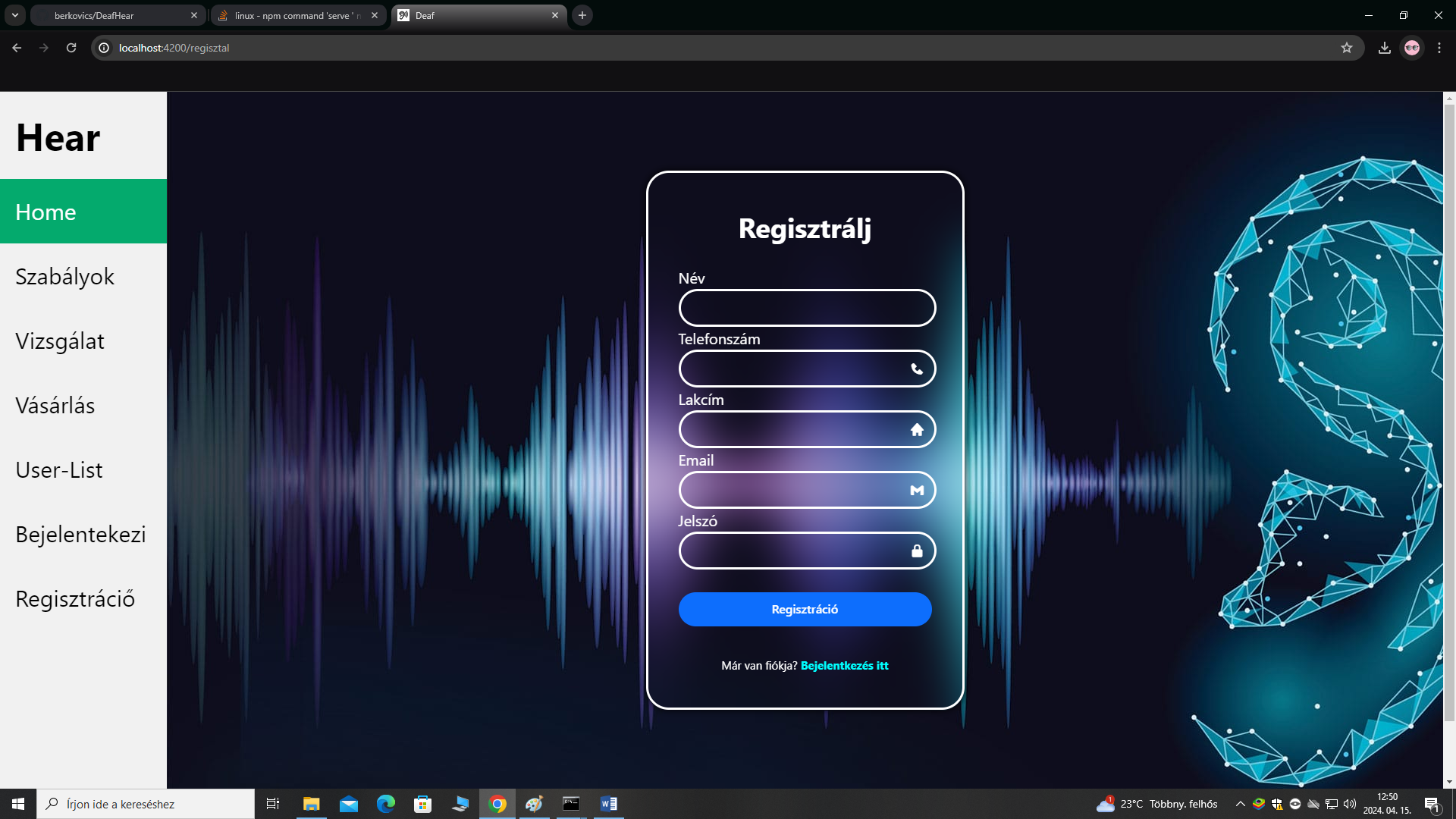
-**Login**: A login komponens felel a bejelentkezésért továbbá innen is lehet tovább lépni a regisztrációra ha esetleg nincs még fiókod

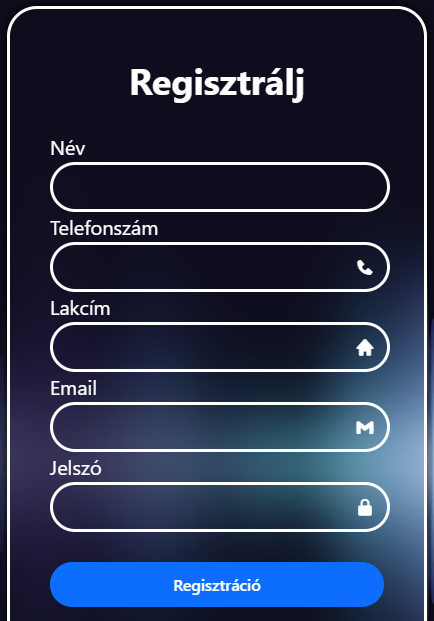
Asztali:



Telefon:



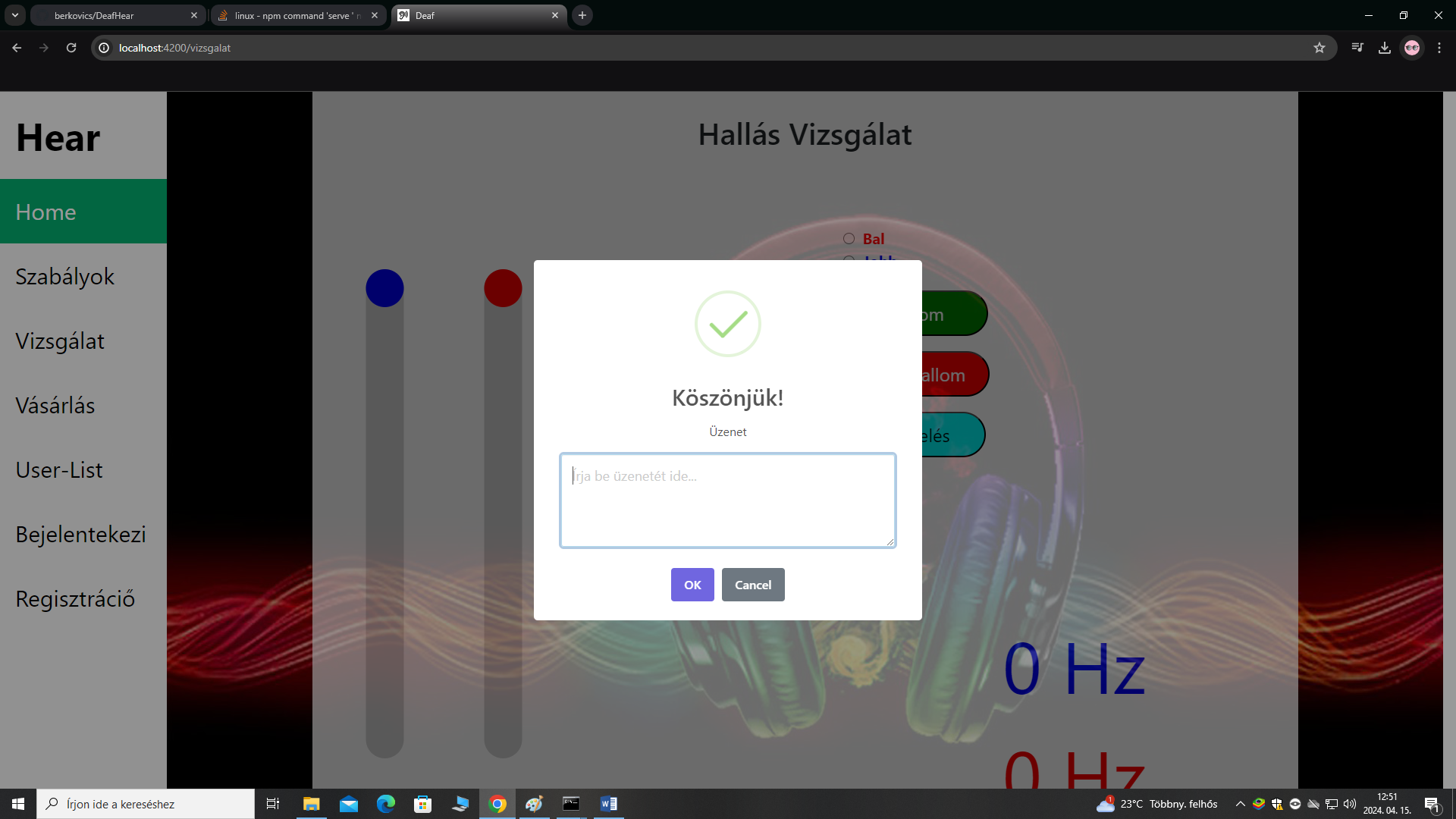
-**Sign-up**: Regisztrációhoz szükséges komponens ahol a szükséges adatok kitöltése után lehet regisztrálni vagy tovább lehet navigálni a bejelentkezésre ha az illetőnek van már fiókja

Telefon:

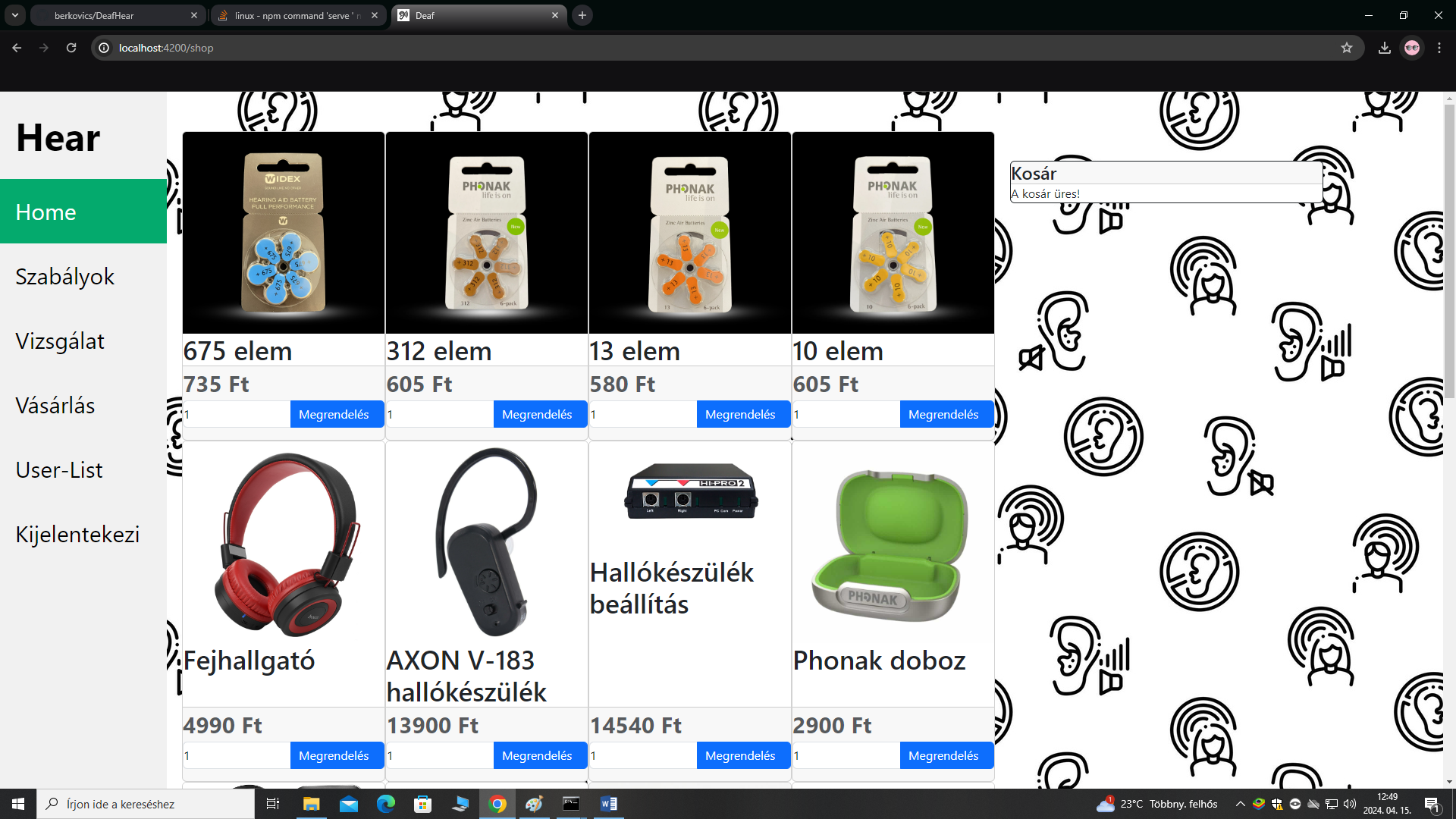
-**Vizsgalat:**

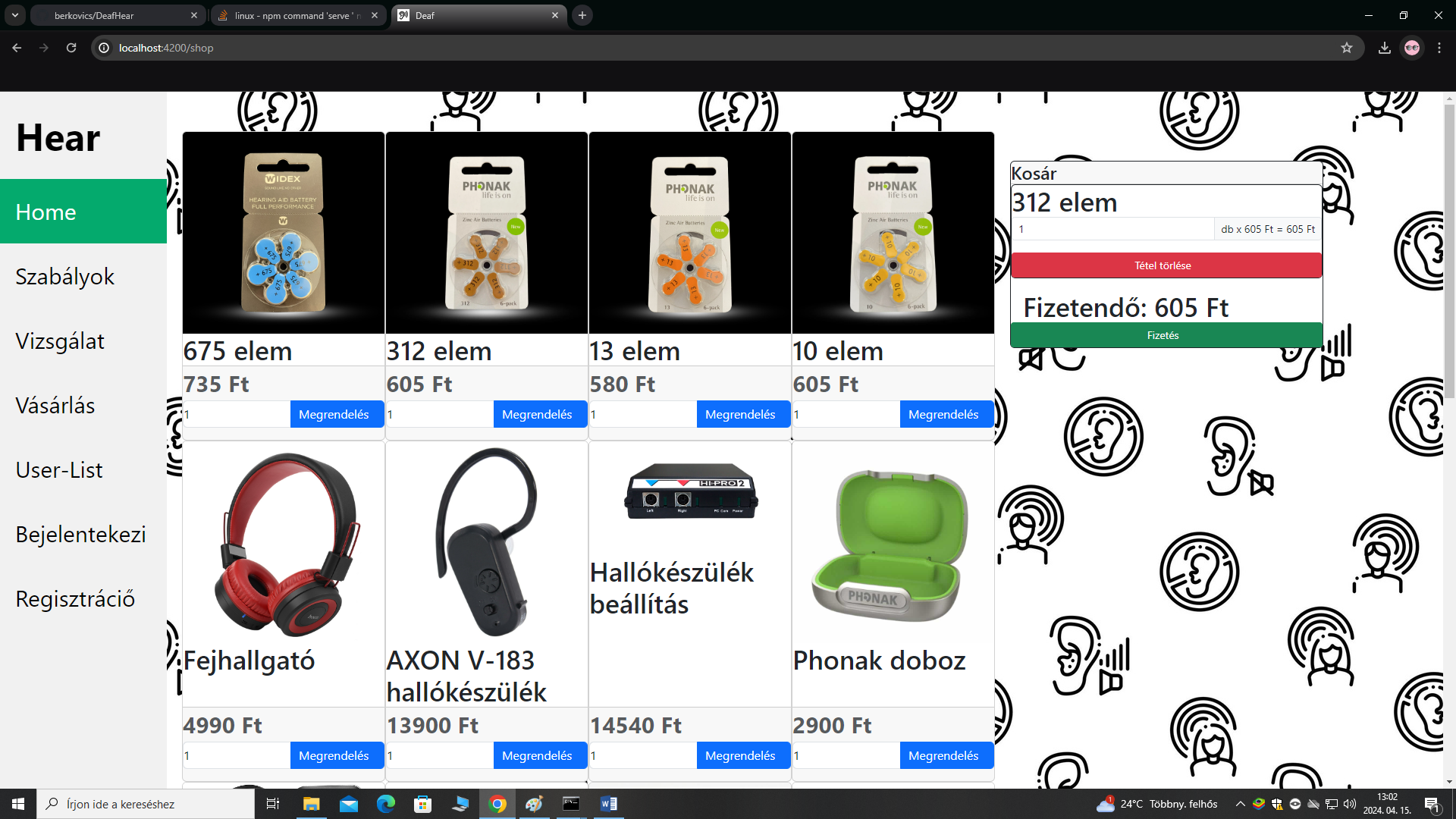
A hallásérzékenységet különböző frekvenciájú és erősségű hangok adásával vizsgáljuk. A felhasználónak reagálnia kell, hogy hallja-e a hangot. A hangok frekvenciája 250 és 4000Hz között változik, az erőssége pedig 20 és 80dB között . A program egy audiogramot rajzol, így a felhasználó képet kap arról, menyire hall jól. Ezután eldöntheti, szüksége van-e egy professzionális hallásvizsgálatra.

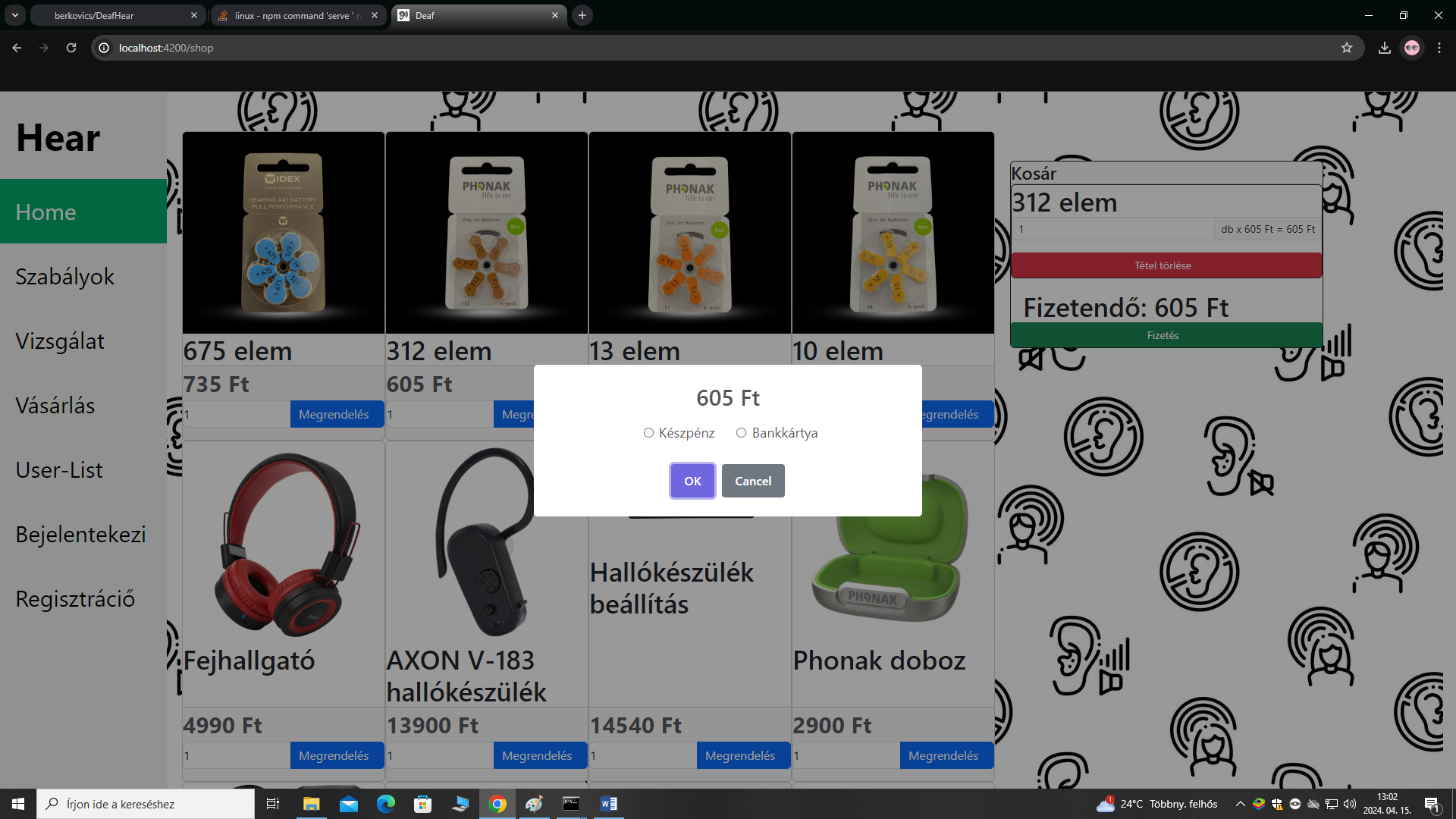




-**Shop:** Itt lehet megvásárolni a kívánt tárgyakat utánvéttel vagy kártyával. Ez természetesen csak bejelentkezés után lehetséges. Valamint a személyes adatokat is csak akkor lehet megnézni, ha az illető rendelkezik már fiókkal.

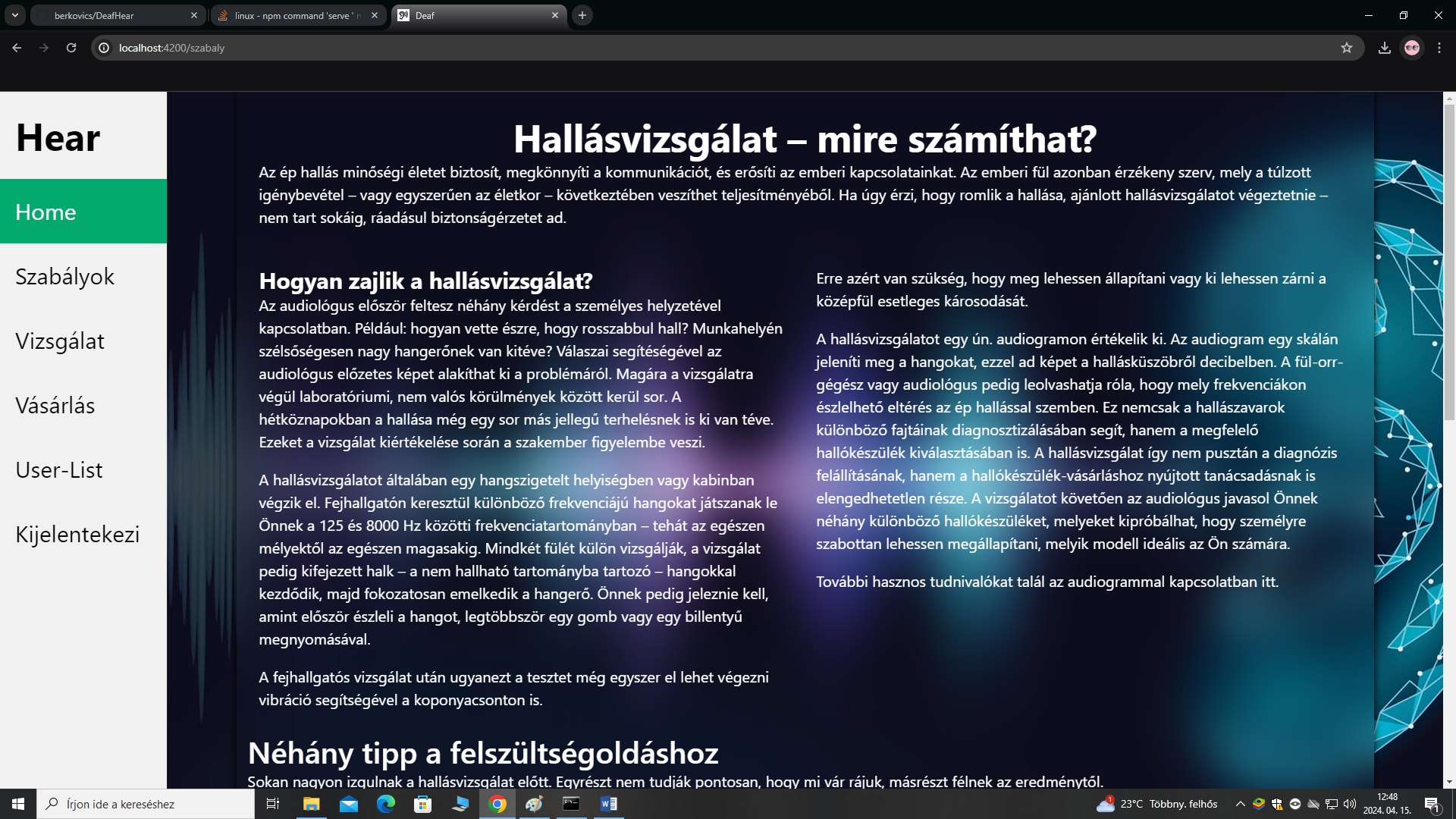






**-Audiogram:**

Hallásvizsgálati görbe, melyre a vizsgáló egyén felvezeti a hallásvizsgálat eredményét, melyet egy koordináta rendszerben jelölnek. A tengelyeken itt a hangok frekvenciája és erőssége szerepel Herzben, illetve decibelben.



**Asztal**

**Berkovics Gellért**

**API útvonalak**

Az **api.php** fájl tartalmazza az alkalmazás API végpontjainak útvonalbeállításait. Ez a dokumentum áttekintést nyújt a elérhető útvonalakról és azok megfelelő funkcionalitásairól.

Felhasznált technologia:

Visual C#

Api

.Net Framework

**Hitelesített útvonalak**

* **/newconfig** (POST): Új konfiguráció létrehozása.
* **/modifyconfig** (PUT): Létező konfiguráció módosítása.
* **/deleteconfig** (DELETE): Konfiguráció törlése.
* **/configs** (GET): Az összes konfiguráció lekérése.

**Hitelesítés nélküli útvonalak**

* **/register** (POST): Új felhasználó regisztrálása.

Model

Ehhez kapcsolódik a **News** **Service**, mely az aktuális hírek megjelenítéséért felelős a főoldalon (*„id”*, *„Name”*, *„*phonenumber*”*, *„Address”*, *„Email”*) amit a backendtől kér el a végpontról egészen a **ApiController**-ből, amit majd későbbi metódusok egészítenek ki;

***get()*** -api/Hallas/5

***post()*** - api/Hallas

***put()-*** *api/Hallas/5*

***delete()*** *-* *api/Hallas/5*

Az értékeket a Hear Modeltől várja, a service belül pedig *param* változóban a *„Id”*, *„Left”*, és a *„Right”* paramétereket kéri

***updateModel()*** –

***deleteModel()*** –

**userCors(**)- builder megjelenik

DeafHearApiContext: megjelent a webek kapcsolodik a backend müködik

**Hiba** komponens: Hibás URL megadására tervezett oldal.

**Swagger Hibás megjelenikpl.error a regisztrácio hiányzás**

**Letöltés a Manage NuGet:**

* Microsoft.EntityFrameworkCore(6.0.23)
* Microsoft.EntityFrameworkCore.SqlServe(6.0.23)
* Microsoft.EntityFrameworkCore.Tools(6.0.23)
* Microsoft.VisualStudio.Web.CodeGeneration.Design(8.0.2)

**Migration:**

Package Manager Console kell megnyitni

hozzáad Add-Migration Initial

utána kell letölteni az Update-Database

**url:**

Ez az oktatóanyag a következő adatmegmaradási szolgáltatások add hozzá az Angular HttpClient segítségével.

launchSettign.json : https://localhost:5291

Angular

HttpClientModule: <http://localhost:7144/api/Registers/>

<http://localhost:7144/api/Hallass/>

Kapcsolodi feltéve, hogy egy nyelv létezik egy nyelv-specifikus Szolgáltatás:

* C# ,
* Api,
* TypeScript,
* HTML,
* CSS

Controller,:

controller létrehozás hogy „add new Scaffolded item „http common „API controller with actions, using Entity Framework”