**Dokumentáció**

Fejlesztők: Móra Barna, Szabó Szebasztián, Kovács Attila

Móra Barna: Backend

Kovács Attila: Frontend

Szabó Szebasztián: Frontend TS

Jegyzet:

Projekt célja

Az ötlet az egy valós problémából indult ki. Ugyanis Pomázon élek viszont semmilyen eseményt nem tudok nyomon követni, valamint haszontalannak és használhatatlannak tartom az önkormányzati oldalt. Megnéztem, hogy hátha más önkormányzati oldalak jobbak vagy használhatóbbak és azt vettem észre sok városnak még ennél is rosszabbul kezelhető oldala van. Ráadásul ezeket elsősorban idősebb korosztály használja főleg számítógépen. A telefonos nézet az végképp használhatatlannak bizonyult. A közösség elsősorban a Facebook csoportban ír és ott lehet olvasni minden városhoz köthető eseményt viszont ezek sajnos nyomon követhetetlenek, egyik értesítés a másik után. Erre jött a megoldás, hogy egy városi fórum/önkormányzati oldalt hozzunk létre. Ez egy sablon amit más önkormányzatok használhatnak és ez sokkal olcsóbb költségvetésből létre tud jönni. Így a városban zajló események a városhoz tartozó oldalon jelennek meg. Ezentúl híreket lokális vagy országos híreket olvashatunk vagy bejelentéseket tehetünk vagy akár tárgyakat hirdethetünk meg amik számunkra már nem kellenek. Így elkerülve a szemetelést is akár.

Felhasznált technológiák

Angular

Bootstrap CSS

Tailwind CSS

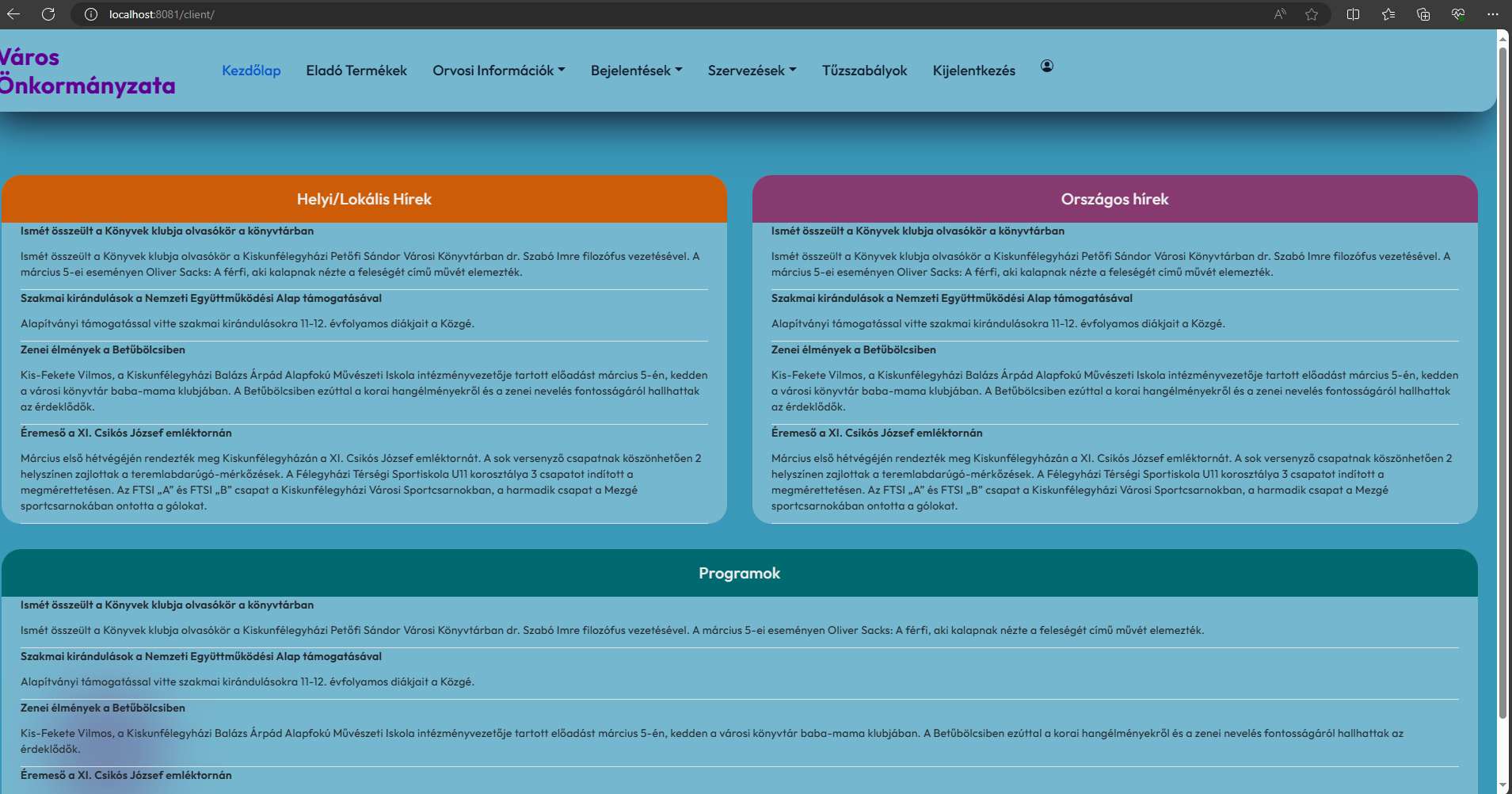
Springboot

Frontend

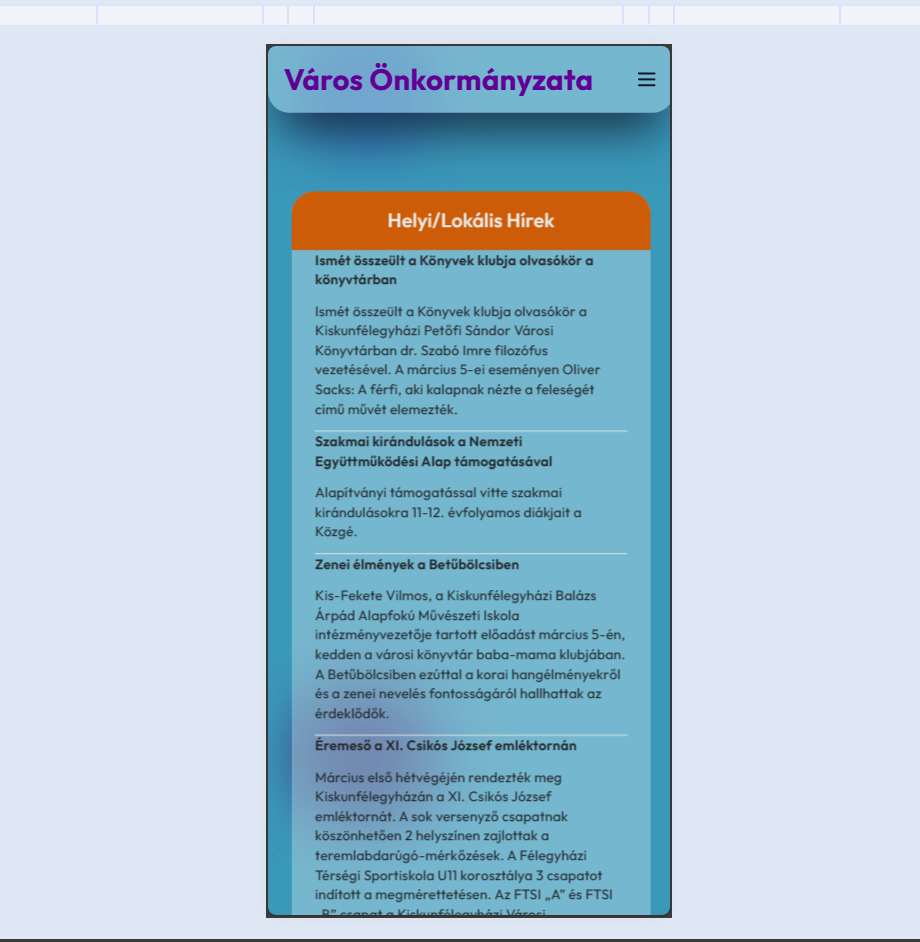
Kovács Attila

Komponensek felhasználói szemszögből:

-**Body komponens**: Ez a főoldal itt jelennek meg a hírek valamint a programok. Ezentúl az adminnak van lehetősége ezeket szerkeszteni vagy kiegészíteni.

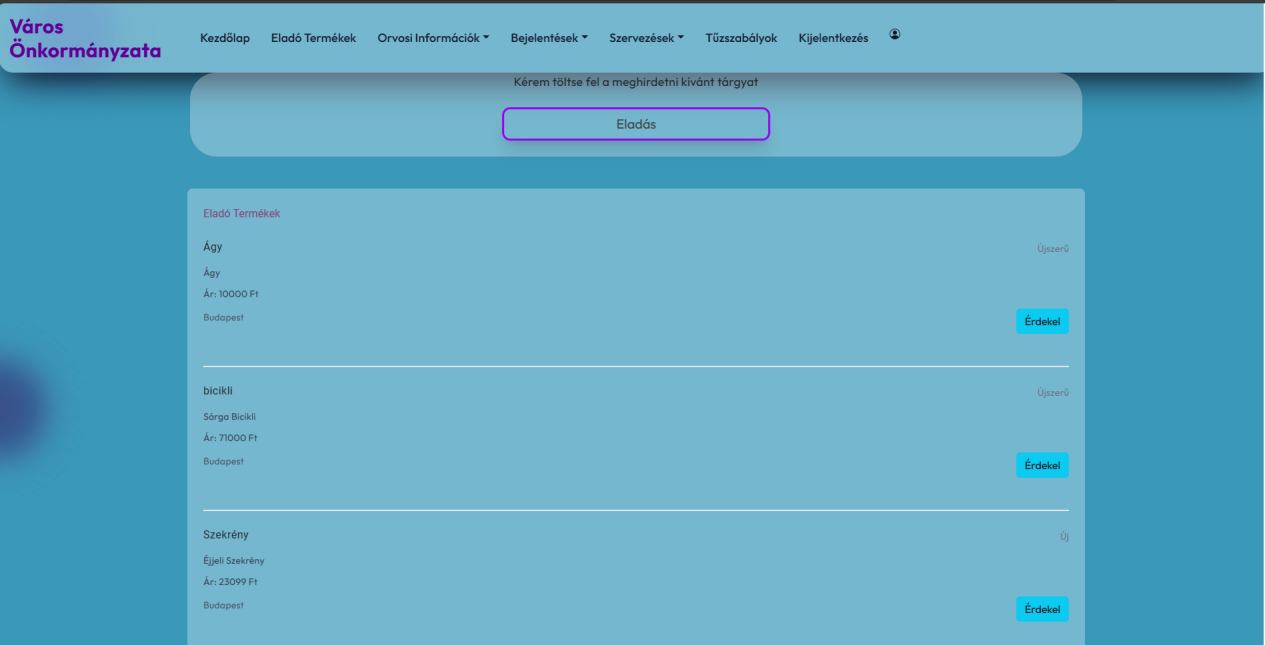
Asztali 

Telefon:

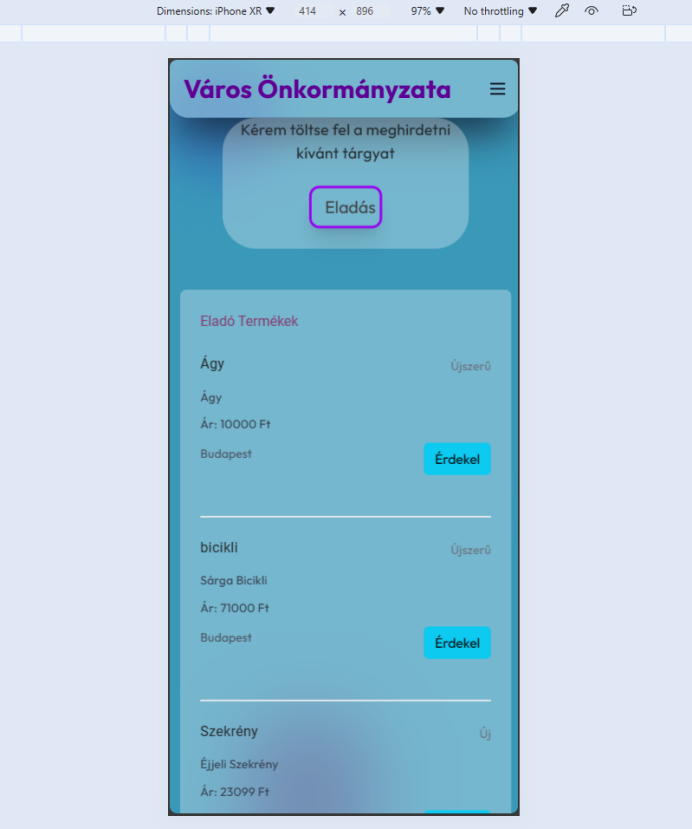


-**Elado-termek komponens:** Ide lehet feltölteni a meghirdetni kívánt tárgyakat pénzért vagy akár ingyen ha számodra nincs értéke a tárgynak. Ez természetesen csak bejelentkezés után lehetséges. Valamint a személyes adatokat is csak akkor lehet megnézni ha az illető rendelkezik már fiókkal.

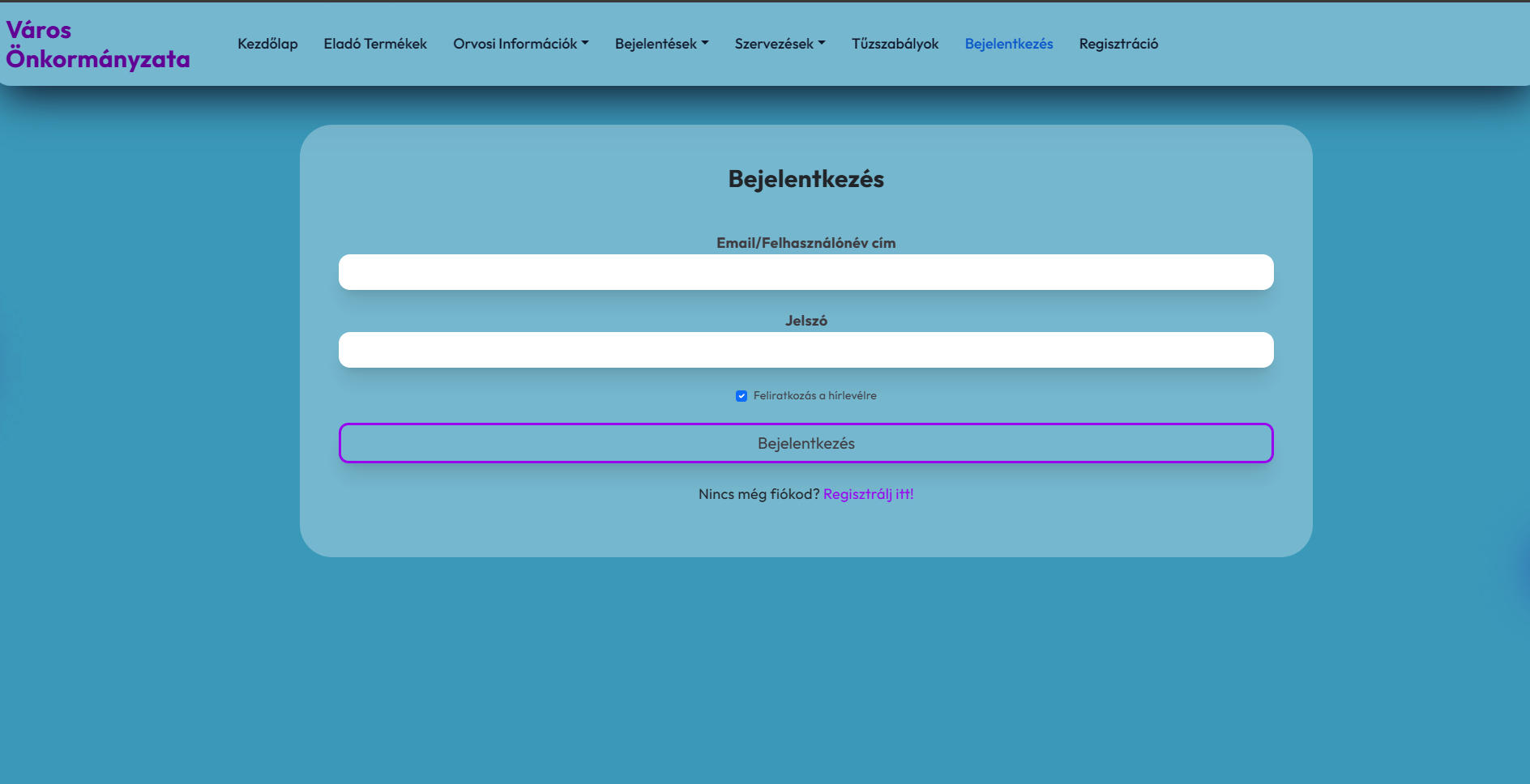
Asztali:



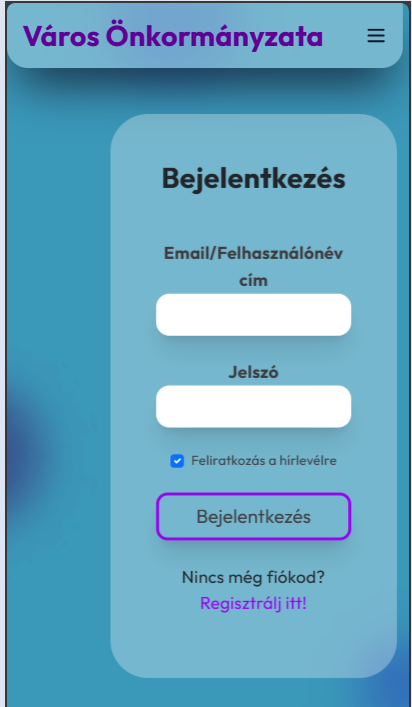
Telefon:

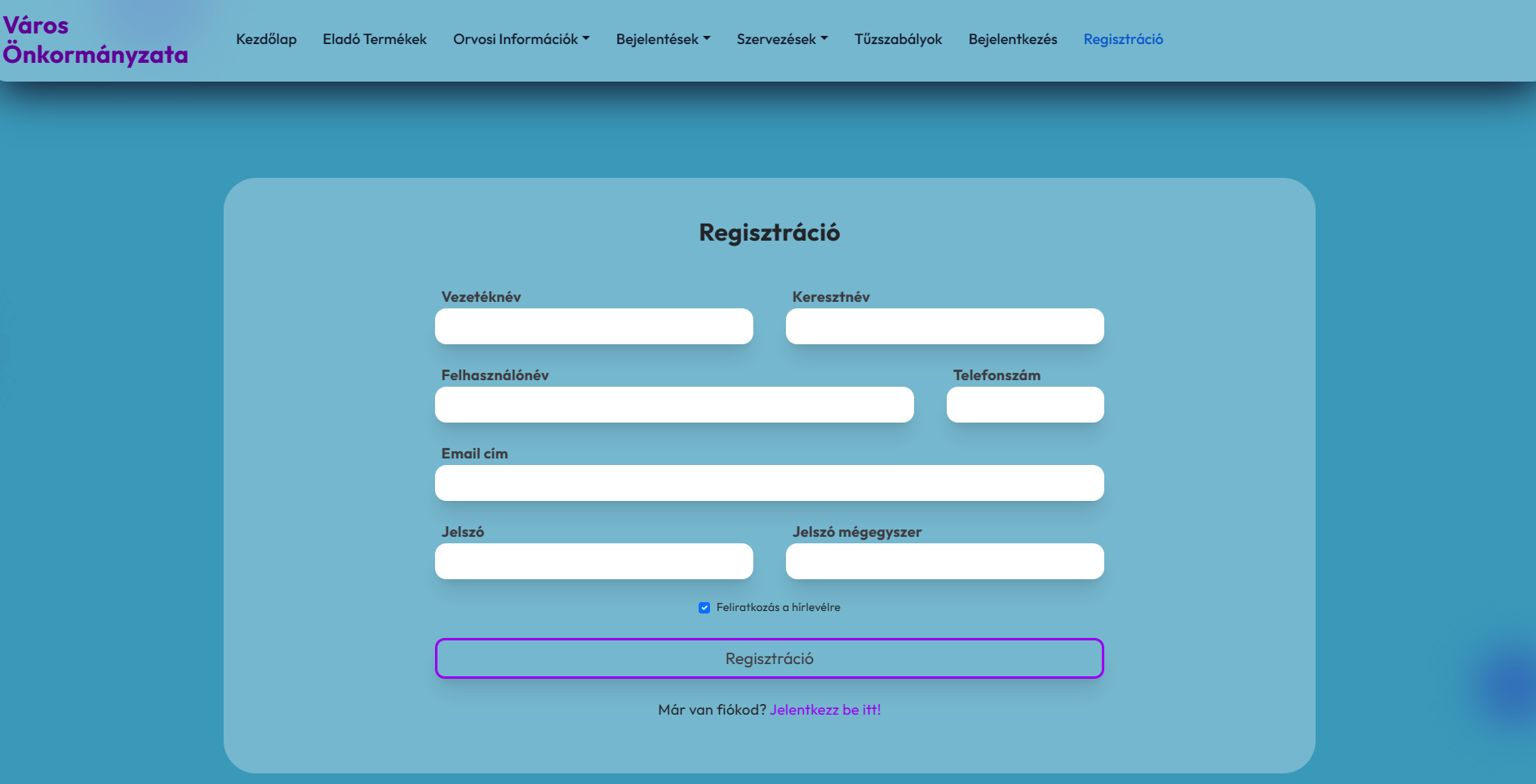


-**Login komponens**: A login komponens felel a bejelentkezésért továbbá innen is lehet tovább lépni a regisztrációra ha esetleg nincs még fiókod

Asztali:

Telefon:



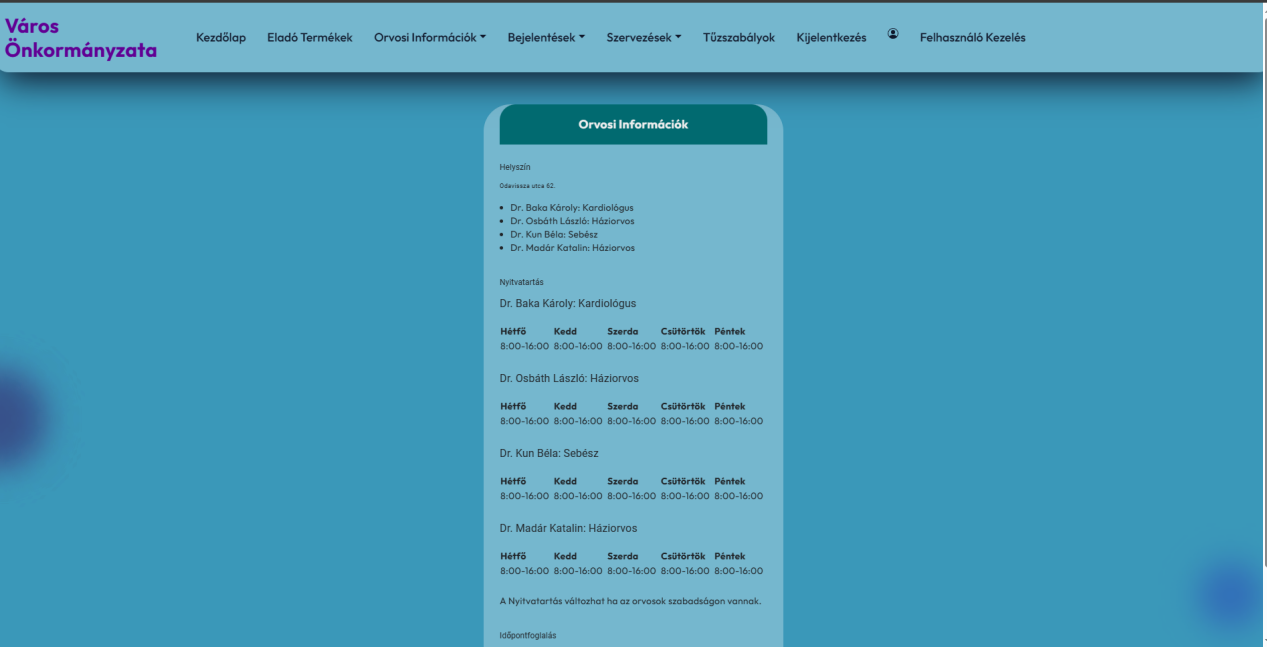
-**Sign-up komponens**: Regisztrácóhoz szükséges komponens ahol a szükséges adatok kitöltése után lehet regisztrálni vagy tovább lehet navigálni a bejelentkezésre ha az illetőnek van már fiókja.

Telefonos nézet:

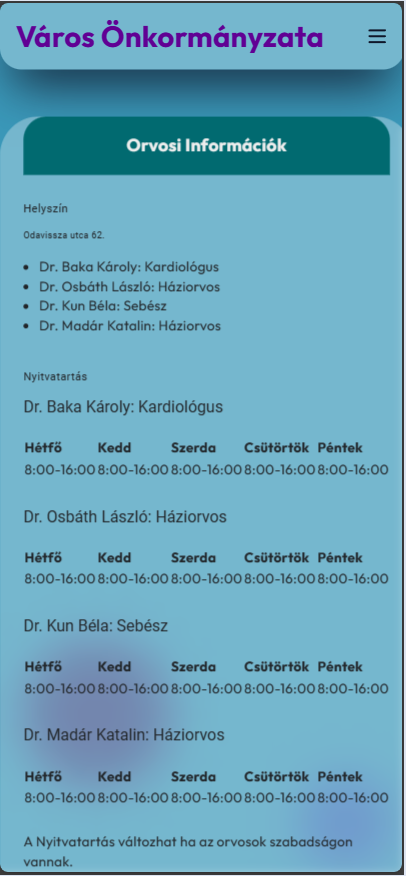


-**Orvos-egyeb-ugyek komponens:** Itt az orvosok listáját valamint a hozzájuk tartozó rendelési időt lehet megnézni továbbá a lap alján lehet jelentkezni időpontra.

Asztali

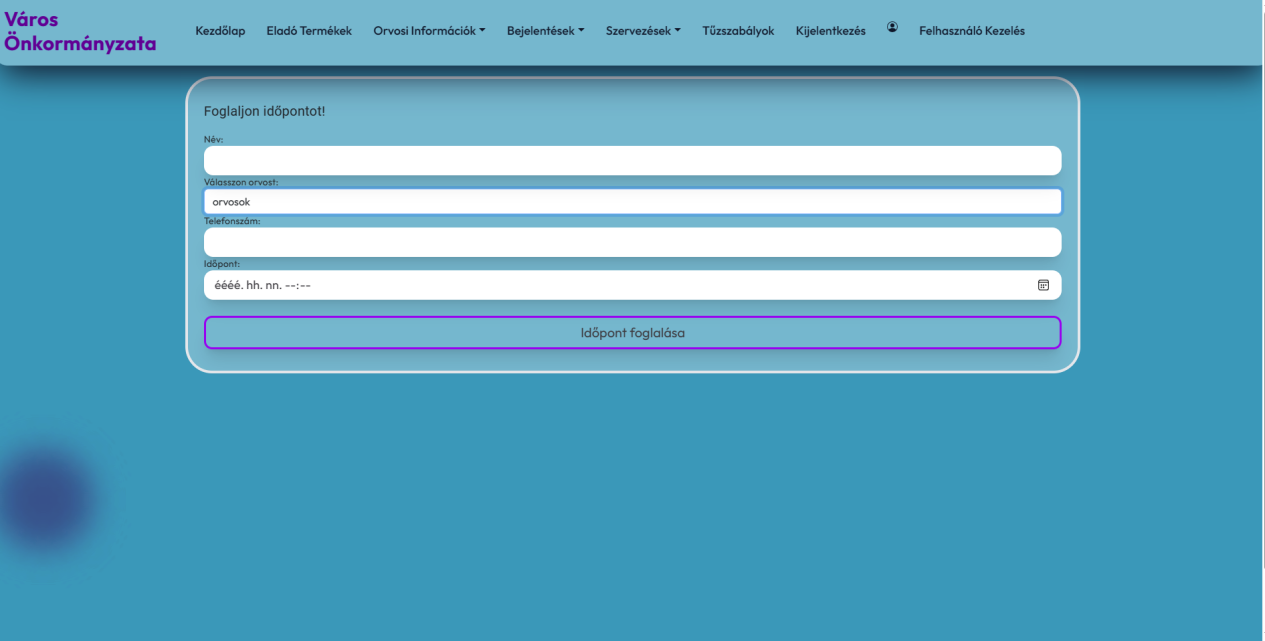


Telefon:



-**Orvos-idopont-foglalas komponens:** Itt lehet időpontot foglalni a kiválasztott orvoshoz.

Asztali

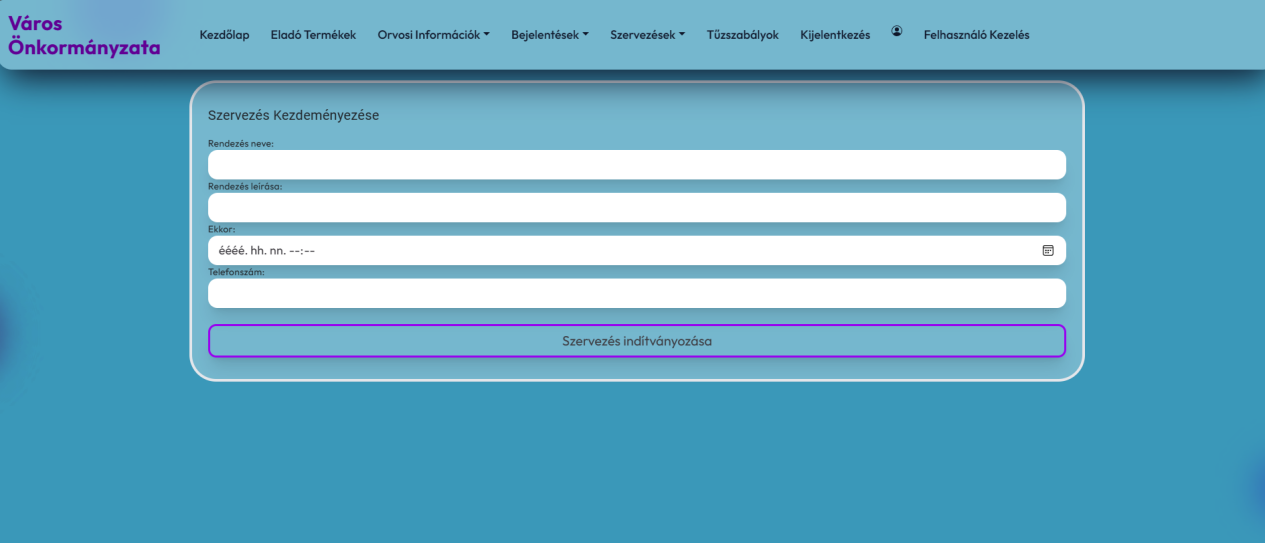


Telefon:

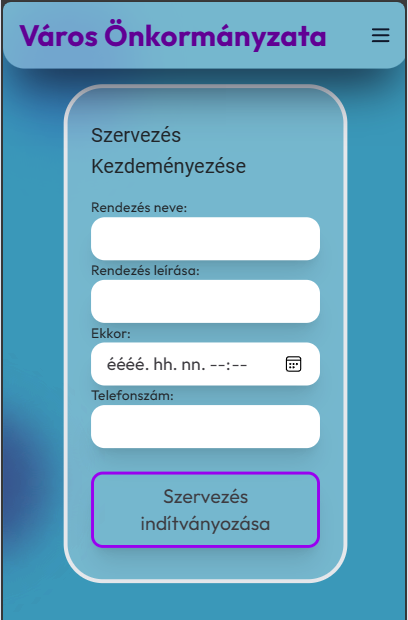


-**Szervezeskezdemeny komponens:** itt kizárólag csak a cégek tudnak rendezvényeket kiírni. Egy példa ha egy spar vagy más üzletlánc szeretné magát promotálni akár létrehozhat egy kezdeményezést hogy spar futás ehhez pontos dátumot és rendezvényleírást kell adnia utána az önkormányzat bírálja el hogy engedélyezi vagy sem és utána a programok résznél a body komponensnél jelenítik meg

Asztali:

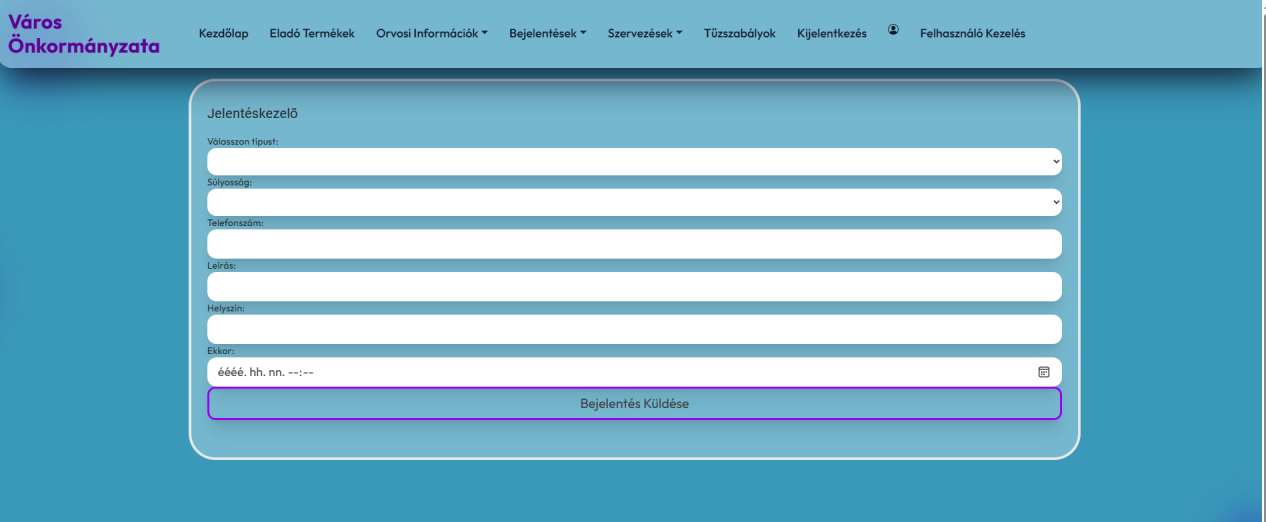


Telefon:



-**jelenteskezelo komponens:** A jelentéskezelő egy olyan felület ahol problémát lehet bejelenteni. Többféle típus van, ezek közül lehet választani és egy súlyosság választó gomb is van benne. Ez annyit takar, hogy ha csőtörés van azonnali 5-ös súlyosságú elem viszont ha szemetet raktak le valahol akkor az 3-as súlyosságú.

Asztali:



Telefon:



Frontend

TS

Szabó Szebasztián

Magába foglalva a különböző **serviceket**, **modelleket**, illetve a **komponens ts fájljait**;

-**Body** **komponens**: A Body komponens különböző híreket és szervezéseket lehet találni; A hírek testre szabhatóan akár helyi („*LOCAL*”), akár nemzeti („*NATIONAL*”), és akár nemzetközi („*INTERNATIONAL*”) híreket is meglehet jeleníteni rajta, melyeket az ORG\_ADMIN szerkeszthet.

Ehhez kapcsolódik a **News** **Service**, mely az aktuális hírek megjelenítéséért felelős a főoldalon (*„id”*, *„orgId”*, *„orgName”*, *„userId”*, *„userName”*, *„title”*, *„content”*, *„type”*) amit a backendtől kér el a *„/resource/”* végpontról egészen a **UserController**-ből, amit majd későbbi metódusok egészítenek ki;

***getNews()*** – *„news/search”*,

***postNews()*** – *„news/new”*

Az értékeket a **Hir** Modeltől várja, a service ts-en belül pedig *param* változóban a *„value”*, *„pageNum”*, és a *„category”* paramétereket kéri

-**Elado-Termekek komponens**: Termékek feltöltéséért és meghirdetéséért felelős komponens; Feltölteni, letörölni, vagy szerkeszteni lehet az adott hirdetéseket, különböző megadható tulajdonsággal, legyen az használtság (*„new”*, *„newish”*, *„used”*) és ár *(„termekek”*, vagy *„termekekFree”*, ha ingyenes).

Ehhez a komponenshez társítva van a **TermekkezeloService**, az értékeket az **EladoTermekekAdatok** **Model**-től kapja (*„userId”*, *„name”*, *„email”*, *„description”*, *„condition”*, *„location”*, *„phone”*, *„price”*), a service ts-en belül pedig „*param”* változóban a *„value”*, *„pageNum”*, és a *„category”* paramétereket kéri.

A backendhez a „*/resource/*” végponton keresztül irányul át az **ItemController-**be, amit majd későbbi metódusok egészítenek ki;

***getTermekek()*** – *„items/search/price”*,

***getTermekekFree()*** – *„items/search/price”*,

***postTermek()*** – *„items/new”*

**FilePickerDirective**: Az eladó termékhez van használva, egy, vagy több fájl kiválasztásáért felel.

-**Felhaszkeres** **komponens**: Felhasználók kilistázására, szerkesztésére (***updateUser()***) és felfüggesztésére (***deleteUser*()**) fejlesztett komponens. **AdminAdatok** **Model** alapján ellenőriz (*„id”*, *„email”*, *„userName”*, *„firstName”*, *„lastName”*, *„phone”*). Ezt az oldalt csak az ADMIN érheti el.

Ez a komponens a **felhaszKeresService**-t használja ami a „/resource/user/” végponton keresztül a backendben lévő **UserController**-el áll kapcsolatban.

***updateUser()*** – *„resource/user”*,

***deleteUser()*** – *„resource/user/del”*

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

-**Jelenteskezelo** **komponens**: 1-5-ig terjedő skálán megadható különböző kategóriájú hibák.

Ehhez a **JelenteskezeloService** kapcsolódik, ami a **BejelentesAdatok**-tól kéri el (*„userId”*, *„type”*, „urgency*”*, *„description”*, *„location”*, *„phone”*, *„date”*) a szükséges paramétereket, melyet aztán a *„resource/notice/new*” végpontról továbbad a backendnek.

-**Orvos-idopont-foglalas komponens**: Kiválasztott egészségügyi ágon való időpontfoglalás;

Ezt a kompnenst követi az **IdopontService**, ami az **IdopontAdatok**-tól kéri el a szükséges paramétereket (*„userId”*, *„orgId”*, *„name”*, *„email”*, *„phone”*, *„startDate”*, *„endDate”*), melyet aztán a *„resource/notice/new*” végpontról továbbad a backendnek. Hasonlóan működik, mint a **Jelentéskezelő Komponens**.

-**Profil komponens**: Megjeleníti a „Profil” felületet, ennek adatait a **ProfilAdatok Model**-ből hívja elő (*„email”*, *„nev”*, *„telefon”*, *„lakhely”*, *„profileImage”*).

-**Sign-up komponens**: Profil regisztrálása, a *„/resource/user”*-en megy keresztül a backendnek, adatit pedig a **regisztacio Model**-től kapja (*„userName”*, „f*irstName”*, *„lastName”*, *„email”*, *„password01”*, *„password02”*, *„phone”*), jelszó duplán beírása kötelező

-**Szervezeskezdemeny komponens**: Cégek által rendezett események, a **szervezesService** viszi tovább a backendnek az adatokat a **SzervezesAdatok** által (*„name”*, „description*”*, *„location”*, *„userId”*, *„orgId”*, *„startDate”*, *„endDate”*, *„phone”, „email”*)