

DEBRECENI SZC BAROSS GÁBOR TECHNIKUM, SZAKKÉPZŐ ISKOLA ÉS  
KOLLÉGIUM

# Reent it

SZAKDOLGOZAT

KÉSZÍTETTE:

Marinka Szilveszter Richárd  
Hagyó Csaba  
Cseh Márton Dávid



Debrecen  
2025

# Tartalomjegyzék

## Tartalomjegyzék

Bevezető.....	1
Milyen keretrendszereket használunk? .....	4
Szerveroldali .....	4
MongoDB.....	4
Prisma .....	5
Kliensoldali .....	6
Figma .....	6
TailwindCSS .....	7
Multilanguage.....	7
Egyéb döntések .....	9
Külső könyvtárak használata.....	9
Arculat .....	10

# Bevezető

Mindhárman egy fontos élethelyzet előtt állunk: egyetemre készülünk, még hozzá Budapestre. Az oda költözéshez elengedhetetlen, hogy találunk egy megfelelő albérletet, de amikor nekiláttunk a keresésnek, gyorsan rájöttünk, hogy a jelenlegi albérletkereső oldalak több szempontból sem ideálisak.

Az egyik legnagyobb probléma, hogy ezek az oldalak nem nyújtanak elegendő információt ahhoz, hogy megbízható döntést hozhassunk. Például nem lehet tudni, milyen értékeléseket kapnak a lakást kiadó tulajdonosok, ami kockázatos helyzetekhez vezethet. Emellett az ingatlanok pontos elhelyezkedése sem mindig elérhető, így nehéz megítélni, hogy egy adott lakás mennyire felel meg a közlekedés szempontjainknak. A reklámok jelenléte tovább rontja a felhasználói élményt, és a legtöbb oldal kezelőfelülete messze nem mondható felhasználóbarátnak.

Ezeket a problémákat látva úgy döntöttünk, hogy elkészítjük a saját albérletkereső oldalunkat, amely a szoftverfejlesztői ismereteinket kihasználva igyekszik megoldani ezeket a hiányosságokat.

A platformunk célja, hogy egyszerű és intuitív legyen, ugyanakkor pontos és megbízható adatokat nyújtson a felhasználóknak. Az oldalunk lehetővé teszi a tulajdonosok értékelését, hogy a bérlők biztosabb döntést hozhassanak, pontos térképes megjelenítést biztosít az ingatlanok elhelyezkedéséről, teljesen mentes a zavaró reklámoktól, a modern felhasználói felület (UI) és a zökkenőmentes felhasználói élmény (UX) érdekében készült.

A fejlesztés során kiemelt figyelmet fordítunk a tervezésre és az egyszerű kezelhetőségre, hiszen az a célunk, hogy az oldal használata ne csak praktikus, hanem élmény dús is legyen. Ez a projekt nemcsak a saját albérletkeresési tapasztalatainkat alapozza meg, hanem egy olyan eszközt hoz létre, amely szélesebb körben is hasznos lehet.

A projekt fejlesztése során több fontos lépést tettünk meg annak érdekében, hogy az általunk tervezett albérletkereső oldal a felhasználói igényeknek maximálisan megfeleljen. A kezdeti szakaszban alapos piackutatást végeztünk, amely során számos meglévő albérletkereső platformot elemeztünk, hogy megértsük, mi az ami működik, és mi az ami nem. A kutatás eredményei alapján egyértelművé vált, hogy a felhasználók leginkább az alábbi három szempontot tartják fontosnak: megbízhatóság, könnyű használhatóság és részletes információk.

Ezeket a szempontokat figyelembe véve kezdtük el a weboldal tervezését. A felhasználói felület (UI) kialakításánál nagy hangsúlyt fektettünk a letisztult és modern dizájnrá. Célunk, hogy a felhasználók könnyedén navigálhassanak az oldalon, és gyorsan rátaláljanak a keresett ingatlanokra. Az oldalra látogatva a felhasználók azonnal rálátnak a legfontosabb információkra: az ingatlan elhelyezkedésére, ár-érték arányára, a bérlői és tulajdonosi véleményekre, és az ingatlan állapotára. A térképes megjelenítés a pontos elhelyezkedést biztosítja, így a bérlők könnyen megítélhetik, hogy az adott ingatlan mennyire van közel az egyetemhez, a közlekedési csomópontokhoz, vagy éppen a szükséges üzletekhez.

A felhasználói élmény (UX) javítását célzó tervezési döntések között szerepelt a gyors keresés és szűrési lehetőségek beépítése is. A bérlők szűrhetik az ingatlanokat több szempont szerint is: ár, elhelyezkedés, szoba szám, felszereltség, és egyéb fontos paraméterek. Ezen kívül az oldal folyamatosan frissíti a hirdetéseket, így a felhasználók biztosak lehetnek abban, hogy mindig naprakész információkat látnak.

A "clean code" elvét követve a fejlesztés során arra törekszünk, hogy a kód átlátható, könnyen érthető és könnyen módosítható legyen. A változó- és függvényneveket egyértelműen választjuk, hogy bárki könnyen megértse, mit csinálnak. A kódot egyszerűsítjük és rendszerezzük, hogy a későbbi fejlesztések és javítások könnyebben elvégezhetők legyenek. Így biztosítjuk, hogy a projekt hosszú távon is jól működjön. Emellett a kód strukturálásával és tisztaságával csökkentjük a hibák lehetőségét, és gyorsabbá tesszük a munkát. A fejlesztők számára könnyen érthető és módosítható kód biztosítja a projekt hatékony kezelését a jövőben is.

```
import { createClient } from "@supabase/supabase-js";

const supabase = createClient(process.env.NEXT_PUBLIC_SUPABASE_URL,
process.env.NEXT_PUBLIC_SUPABASE_ANON_KEY);

export default supabase;
```

A projekt egyik legnagyobb kihívása volt, hogy a program asztali számítógépen kívül minden platformon, így mobiltelefonokon és táblagépeken is zökkenőmentesen működjön. Ehhez rezponzív dizájnt alkalmaztunk, amely automatikusan igazodik a képernyőmérethez, így a felhasználói élmény nem csökken a különböző eszközök használata során.

A fejlesztés során végig tartottuk szem előtt, hogy az oldalunk ne csupán a saját albérléskeresési tapasztalatainkra épüljön, hanem egy olyan eszközként is szolgáljon, amely szélesebb körben hasznos lehet más, hasonló helyzetben lévő felhasználók számára is. Mivel mindhárunk számára fontos, hogy a projekt végén egy jól működő, megbízható platformot adjunk át, ezért folyamatos tesztelést végeztünk, hogy biztosítsuk a hibamentes működést.

Ebben a dokumentációban bemutatjuk az oldalunk fejlesztésének hátterét, a megvalósítás folyamatát, valamint azokat a szoftverfejlesztési döntéseket, amelyek meghatározzák a projektsikerét.

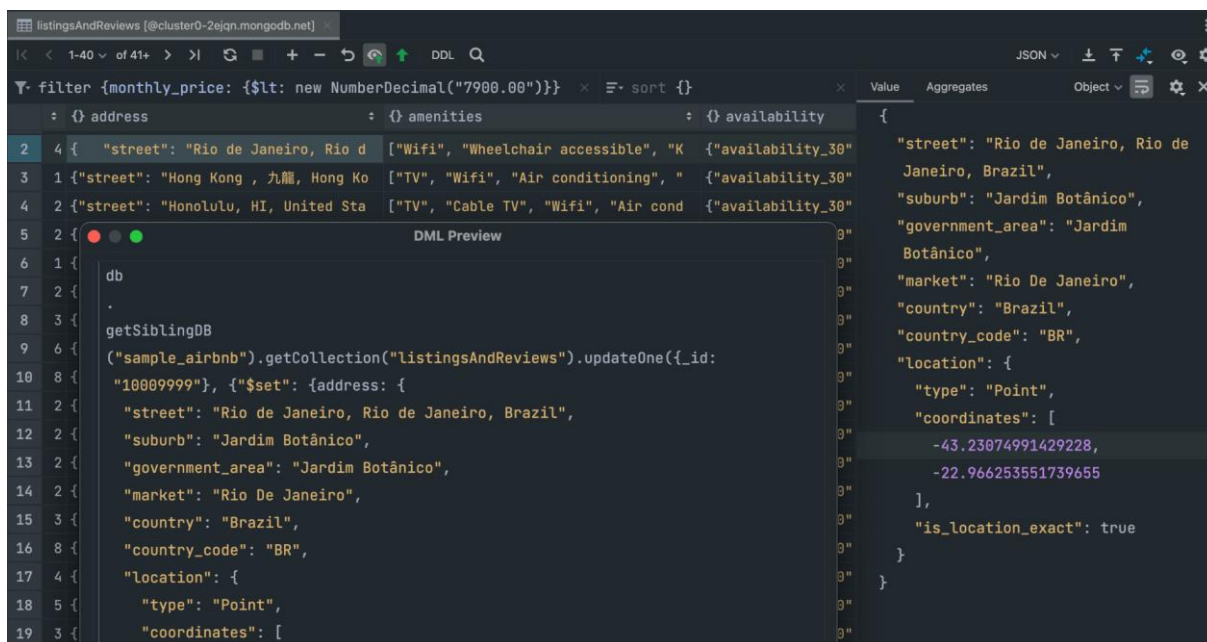
# Milyen keretrendszereket használunk?

Projekt fejlesztése során törekedünk arra, hogy minél kevesebb harmadik féltől származó könyvtárat használjunk, ezáltal pár Utils saját fejlesztésű, és saját szerveren futtatható, így teljes befolyásunk van a működésére.

## Szerveroldali

### MongoDB

A projektünk kezdetekor el kellett döntenünk, hogy az adatbázis kezeléshez használt programmal kapcsolatban maradunk-e a hagyományos MySQL-nél, amit az elmúlt években tanítottak nekünk vagy megpróbálunk valamit, ami nem szokványos, de a munkánk annál inkább felgyorsíthatja, könnyebbé teheti. Ekkor jutottunk a döntésre, hogy az eddig megszokott program helyett MongoDB-t használunk, ami jobban megfelel az igényeinknek. A MongoDB a MySQL-hez képest rugalmasabb adatkezelő, a programunk gyakran változó adat szerkezetait egyszerűbben kezeli. A MongoDB használata lehetővé teszi számunkra, hogy gyorsabban és hatékonyabban fejlesszük alkalmazásunkat. Az adatmodell könnyen módosítható, ami csökkenti a fejlesztési időt. Ezenkívül a MongoDB skálázhatósága is kiemelkedő, mivel képes nagy mennyiségű adat kezelésére.

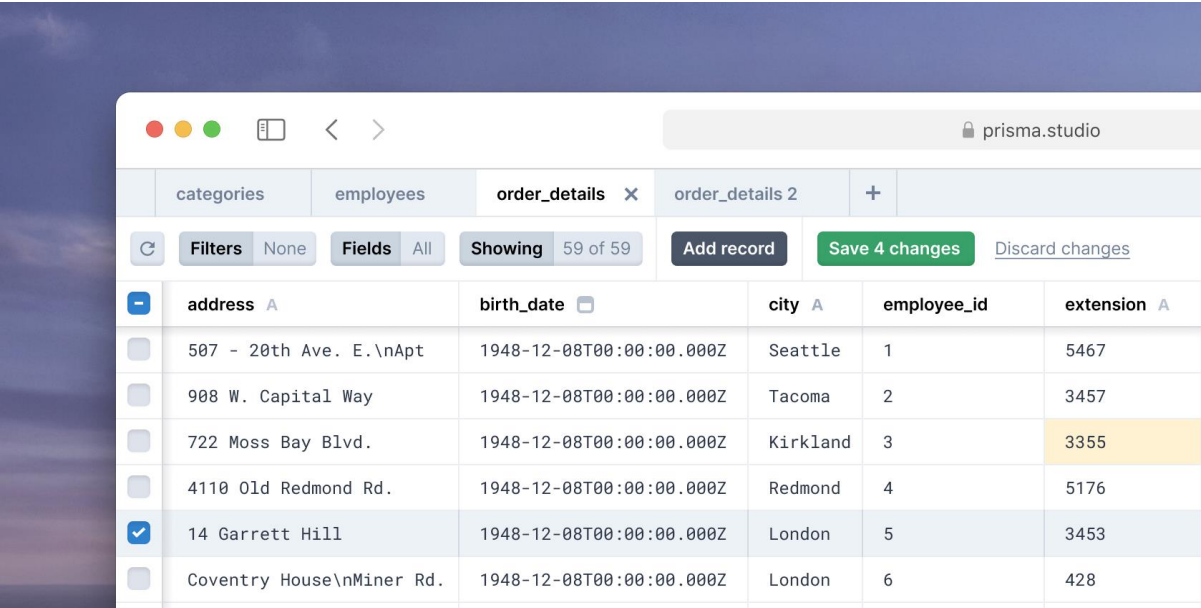


1. ábra

Példa a MongoDB felhasználói felületére és kódjára

## Prisma

Az adatbázis kezeléshez Prisma ORM-et használunk a MongoDB-vel egyhuzamban. A prisma schema-kat létrehozva jelentősen felgyorsítja a munka folyamatát sima sql kódhoz vagy akár más keretrendszerhez képest. Az eszköz automatikusan generál egy típusbiztos lekérdező klienst ami azt teszi lehetővé, hogy közvetlen az adatbázis objectumaira hivatkozhatunk programunkból, ez jelentősen felgyorsítja a fejlesztést és minimalizálja a potenciális hibákat. A különböző adatbázis kezelők támogatása is egy nagy előny hisz a sima MySQL-től eltérve a MongoDB-re való támogatás hatalmas megkönnyebbülés. A Prisma nagyban hozzájárult a zökkenőmentes kezeléséhez adatbázisunkban.



The screenshot shows the Prisma Studio web application interface. At the top, there's a browser address bar with 'prisma.studio'. Below it, a navigation bar shows tabs for 'categories', 'employees', 'order\_details', and 'order\_details 2'. The 'order\_details' tab is active. Below the tabs, there's a toolbar with buttons for 'Filters', 'Fields', 'Showing 59 of 59', 'Add record', 'Save 4 changes', and 'Discard changes'. The main area displays a table with columns: 'address', 'birth\_date', 'city', 'employee\_id', and 'extension'. The table contains six rows of data. The row with '14 Garrett Hill' is selected, indicated by a blue checkmark in the first column.

	address A	birth_date	city A	employee_id	extension A
<input type="checkbox"/>	507 - 20th Ave. E.\nApt	1948-12-08T00:00:00.000Z	Seattle	1	5467
<input type="checkbox"/>	908 W. Capital Way	1948-12-08T00:00:00.000Z	Tacoma	2	3457
<input type="checkbox"/>	722 Moss Bay Blvd.	1948-12-08T00:00:00.000Z	Kirkland	3	3355
<input type="checkbox"/>	4110 Old Redmond Rd.	1948-12-08T00:00:00.000Z	Redmond	4	5176
<input checked="" type="checkbox"/>	14 Garrett Hill	1948-12-08T00:00:00.000Z	London	5	3453
<input type="checkbox"/>	Coventry House\nMiner Rd.	1948-12-08T00:00:00.000Z	London	6	428

2. ábra

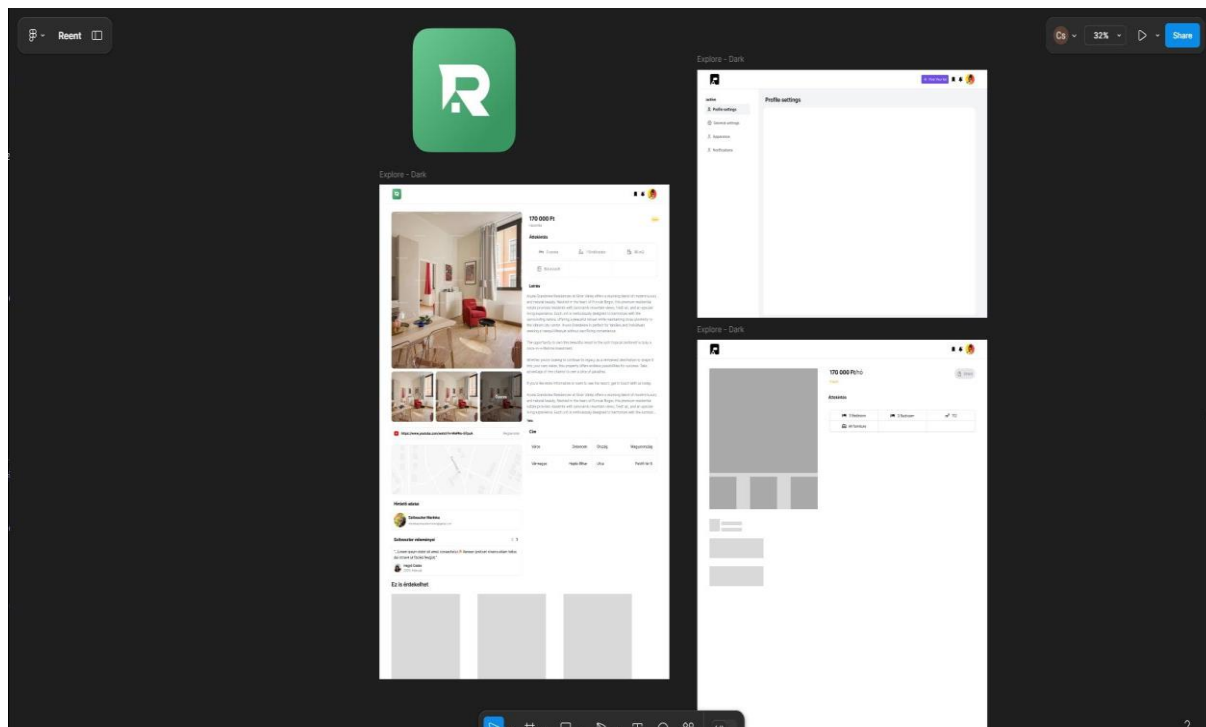
Példa a prisma studio felhasználó felületéről

# Kliensoldali

## Figma

Feladat elkezdése előtt nyilvánvaló lett számunkra, hogy a programozás egyik alap lépését, a tervezést nem hagyhatjuk ki. Erre nyújt megoldást a Figma, mint modern tervezőeszköz. A felhasználói felület, a Frontend egyik legnagyobb kihívása a Figma használatával kifejezetten egyszerűbb és élménydúsabb lett. A program egyik legnagyobb előnye, hogy a valós idejű együttműködést lehetővé teszi ezáltal a csapatunk tagjai egyszerre dolgozhatnak ugyanazon a terven, akárhol is helyezkedhettek el, bármilyen időben. A tervezési folyamat időigénye ennek hatására egekbe is képes nőni. Jóval megkönnyítette a kommunikációt és a közös munkát is.

A kommunikáció további fontos tényezője, hogy a Figma böngésző alapú, így külön telepítésre nincs szükség, bárhol bármilyen eszközről el lehet érni az oldalunk egyik legfontosabb területét, akár mobiltelefonról is. A dizájn integrálása más keretrendszerbe egy nagyon fontos funkció, hisz sokszor, több időt igénylő kód megvalósítása sose volt ilyen zökkenőmentes. A projektünk kinézete a modern webdizájnból nyer ihletet, szeretné replikázni és ennek megvalósításához elengedhetetlen eszköz volt a Figma.



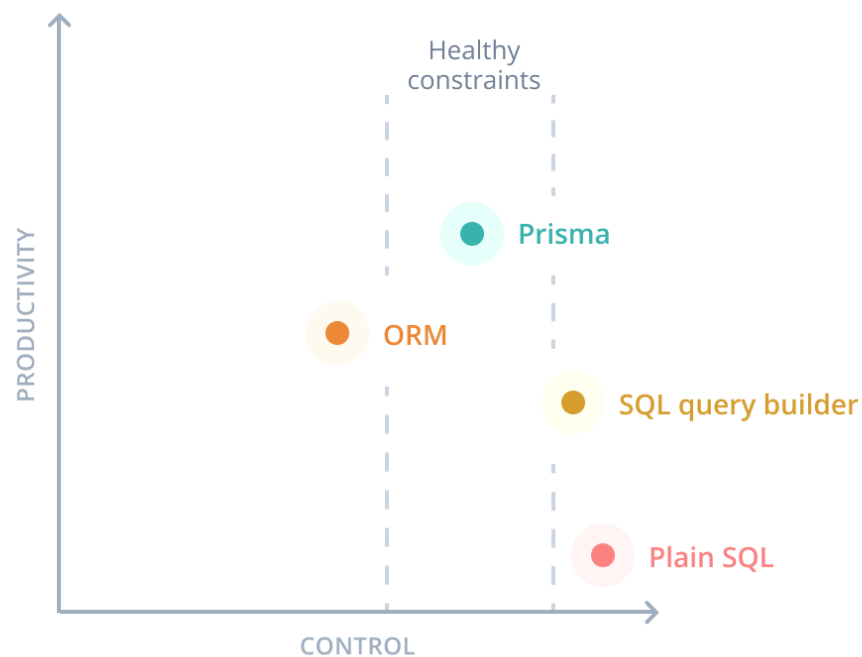
3. ábra

Figma felhasználó felület



## TailwindCSS

A frontend programozás egyik legnagyobb része a dizájn kifejlesztése és a stílusos megoldások megvalósítása. Erre a célra sok fajta megoldás létezik, de mi egy keretrendszer használatával fejlesztjük projektünket. A TailwindCSS egy modern, könnyen használható keretrendszer, ami előre definiált osztályokat biztosít a stílusok közvetlen alkalmazásához a HTML elemekre. Egy stílusos, felhasználóbarát weboldal ki alakításához elengedhetetlen egy megbízható keretrendszer alkalmazása, ami nem korlátoz előre meghatározott komponensekre így lehetővé teszi az egyedi és reszponzív felhasználói felület létrehozását. A keretrendszer integrálása projektünkbe zökkenőmentes, és gyors folyamat, ami lehetővé teszi, hogy a fontos dolgokra koncentráljunk

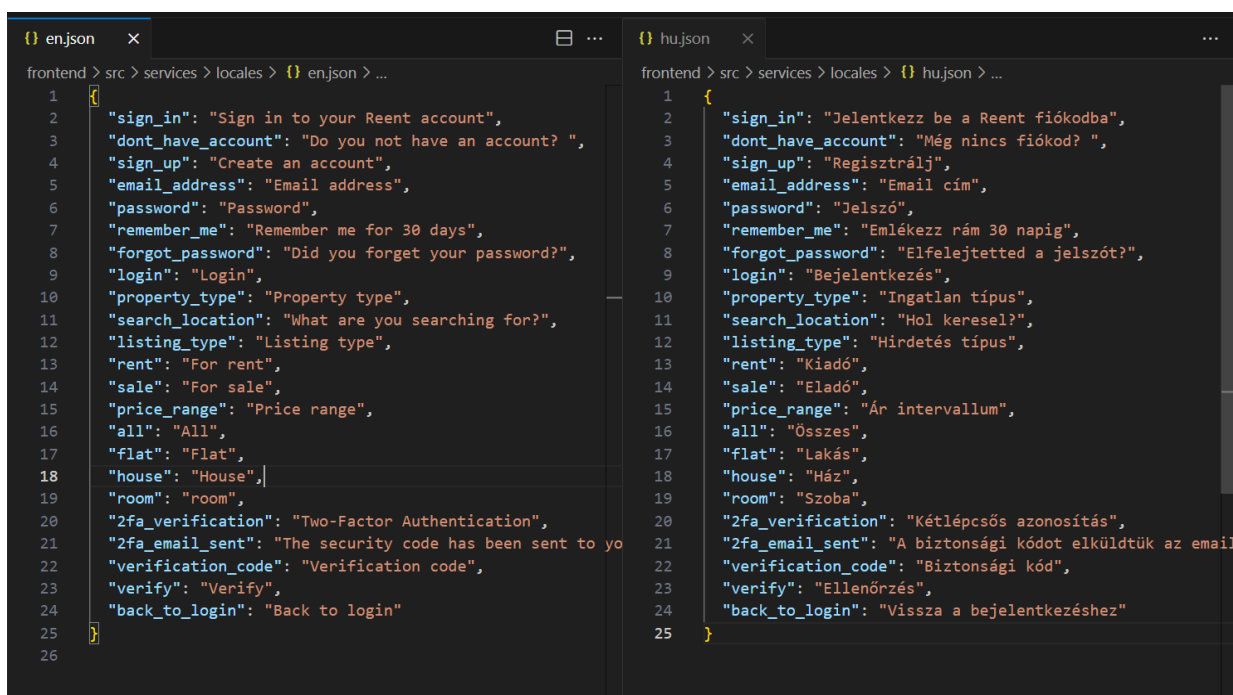


4. ábra

Prisma teljesítménye többi keretrendszerhez képest

## Multilanguage

A projektünk fejlesztése során nagy figyelmet kap a nyelvi helyesség, annak esztétikája és az elérhetőség ami lehetővé teszi, hogy ne csak magyar anyanyelvűek élvezhessék weboldalunk. Ez a döntés nagyban javítja a felhasználói élményt és nagyobb közönség elérését teszi lehetővé. A többnyelvűség megvalósítása érdekében el kellett döntenünk milyen nyelveket szeretnénk az oldalunkon, a döntés nem volt nehéz hisz a magyar nyelv adott volt a feladat követelményekbe, az angol pedig az internet nyelve és a világon legtöbb ember által beszélt nyelvek egyike. Mindkét nyelvhez külön fordítási fájlokat hozunk létre amelyek tartalmazzák az adott nyelvhez tartozó szövegeket és kifejezéseket. Először elkészítettük a magyar nyelv fájlt majd azt fordítottuk angolra nagy odafigyeléssel a helyesírásra, a nyelvi normákra és a kulturális sajátosságokra. Elengedhetetlen feltétele a nyelv integrálásának az oldalunkban, hogy a felhasználótól elvárt bevitel a váltás között minél érthetőbb és felhasználó barátabb legyen, ezért törekedtünk egy letisztult megoldást nyújtani. Összességében a Multilanguage funkció nagyban hozzájárul a felhasználói elégedettséghez, külföldi látogatóink által.



5. ábra

Angol és magyar verziói ugyanannak az oldalnak

# Egyéb döntések

## Külső könyvtárak használata

Projectünk során törekedünk minimalizálni a külső könyvtárak használatát, és inkább saját megoldásokat dolgozunk ki. Ez a döntés hosszútávon nagy előnyökkel jár a munka folyamatában, a végeredményben és a felhasználói élményben. A saját kód írása fejleszti a programozási, probléma megoldó és általánosságban szakmai képességünk/tudásunk. Mélyebb megértést szereztünk a programozási alapelvekről és a komplikáltabb algoritmusok működéséről is. Alaposan utána kell járnunk és megértenünk a megoldandó problémát, hisz nem támaszkodtunk előre elkészített megoldásokra. A saját kód írásával nagyban növeljük a programunk flexibilitását, rugalmasságát. E döntés által nem vagyunk kötve külső könyvtárakhoz, vagy előre specifikált struktúrákhoz amik nagyban akadályoznák a munka előre haladását. A különböző függőségek száma is minimalizálva volt, e döntés által, nem kell aggódnunk más kódjának hibáival programunkban.

```
frontend > src > pages > UploadPage.jsx > UploadPage > validateStep
54   if (!formData.deposit || formData.deposit < 0) newErrors.deposit = "Érvénytelen kaució";
55   if (!formData.description?.trim()) newErrors.description = "Kötelező mező";
56   if (!formData.city?.trim()) newErrors.city = "Kötelező mező";
57   if (!formData.county?.trim()) newErrors.county = "Kötelező mező";
58   if (!formData.yard) newErrors.yard = "Kötelező mező";
59
60
61   setErrors(newErrors);
62   return Object.keys(newErrors).length === 0;
63 };
64
65 const nextStep = () => {
66   if (!validateStep(step)) return;
67   setStep((prev) => Math.min(prev + 1, 4));
68 };
69
70 const prevStep = () => setStep((prev) => Math.max(prev - 1, 1));
71
72 const handleImageUpload = (files) => {
73   const newImages = files.map((file) => ({
74     url: URL.createObjectURL(file),
75     name: file.name,
76     file,
77   }));
78
79   setFormData((prev) => ({
80     ...prev,
81     images: [...prev.images, ...newImages].slice(0, 10),
82   }));
83 };
84
85 const goToStep = (stepNumber) => {
```

6. ábra

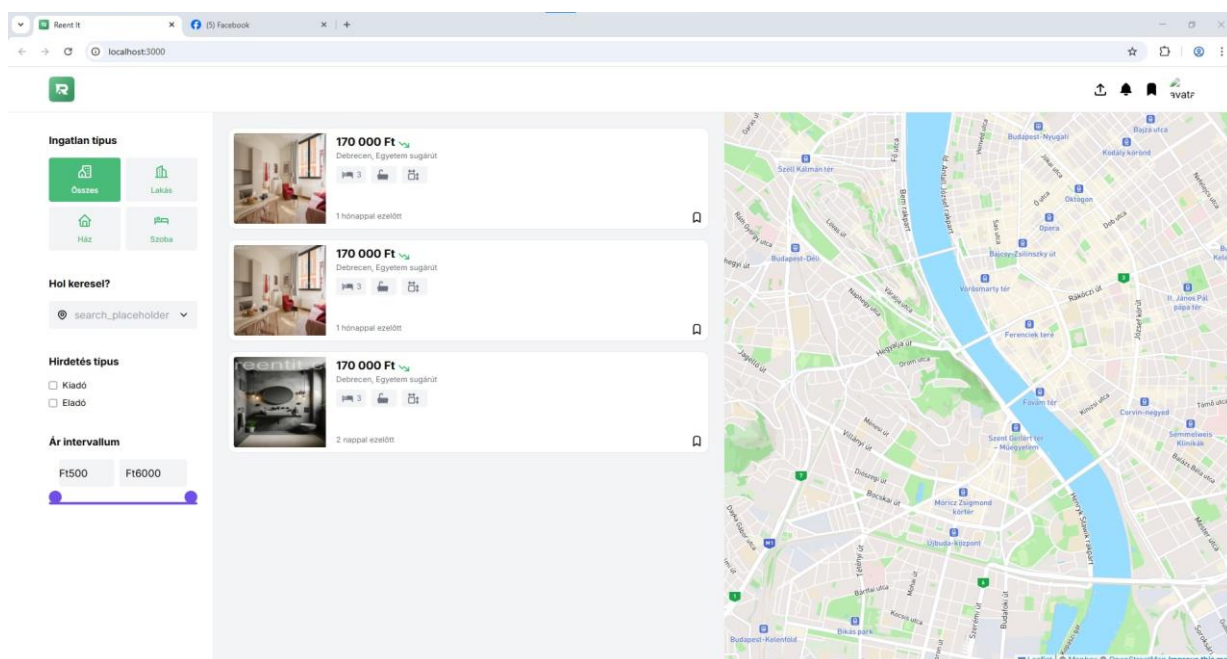
Példa egy személyesen írt algoritmusra

# Arculat

A projektünk professzionális megjelenéséhez fontos az arculat részletes és előre megtervezett kialakítása. A weboldal megtervezésekor végcélnak egy modern stílusú, egyedi és saját értékeinket tükröző megoldást alakítunk ki, és valósítunk meg, saját képességeink szerint. Az arculattervezés legfőbb pontjai a következők:

- Színséma
- Logó
- Betűtípus
- Layout dizájn

A színséma feladata, hogy életet adjon az oldalba, nagyon fontos a felhasználókban keltett érzés a weboldal megnyitásakor. A csapatunk által használt színárnyalat az Óceánzöld (#48BB78). Az Óceánzöld egy komplex árnyalata a zöldnek, ami összekeveri a Jászöld luxust, és szofisztikáltságot sugárzó tulajdonságait, egy kis sárgával ami energiát és odaadást sugall. A zöld az üzlet színe ami egy megbízható érzést ad.



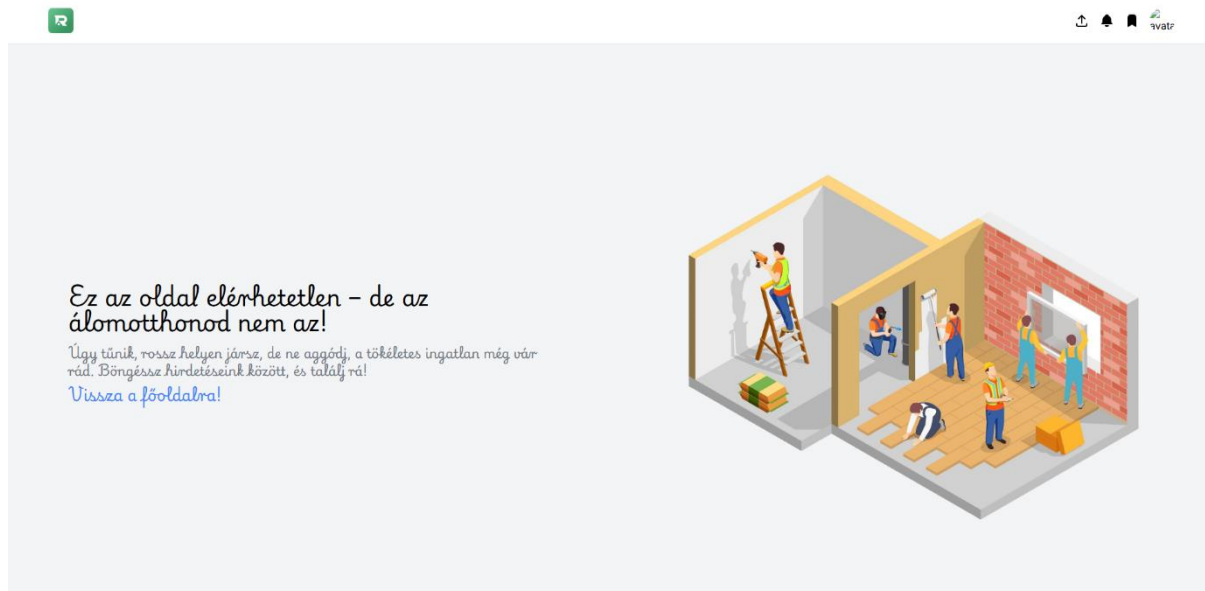
7. ábra

Példa az Óceánzöld használatáról weboldalunkban

A logó készítésekor a prioritásunk a felismerhetőség, letisztult dizájn és az oldalunkat látogató felhasználón hatott pszichológiai hatás. Fő a letisztultság ezért az oldalunk nevének kezdőbetűjbe építetjük az webhely célját szimbolizáló házat ami a megbízható lakási lehetőséget jelképezi.

Ha betűtípusról van szó az emberi agy legnagyobb követelménye az egyszerűség és kényelem. Szeretünk olyan betűket olvasni már gyerekkortól amit könnyen megértünk és nem kell energiát fektetnünk a megértésébe. A csapatunk intenzív kutatást folytatott a legjobb betűtípus megtalálásához, és a végén az Inter nevű betűtípusra döntöttünk. Egy nagyon megbízható és kényelmesen kialakított font amit sokan használnak a dizájn világában.

A layout dizájn nagyon fontos része proramunknak. Törekedtünk a modern dizájnrá, hisz a mai világban az elavult rendszerek nem felvállalhatóak, gyorsan fejlődő világban fontos a felszínre úszni és fenntartani a tempót. Erre törekedtünk programunk fő dizájnával. Minél kevesebb elterelés a figyelem megőrzése céljából, reklám mentesség, kerekített kialakítás a főbb jellemzője ennek a stílusnak.



8. ábra

példa az oldaldizájnrá

# Befejezés

Összegzőképpen, arra törekedtünk, hogy e 3 fős baráti társasággal egy működőképes, stílusos és jól működő albérléskereső weboldalt hozzunk létre, ami megfelel az elvárt követelményeknek és a mi igényeinknek, a kezünkből kiadott munka nem csak egy jegy, hanem egy tükre a személyiségünknek, a magunkhoz címzett elvárásoknak. Nagyon fontos tényező, hogy az elkészített program akár a munka világában is megállja a helyét. Természetesen nem szeretnénk lealacsonyítani a professzionális munkákat a dolgozó világban, de törekedünk minél nagyobb minőségben kötelességeinket elkészíteni, ezzel felkészülni a jövőre. Nélkülözhetetlen számunkra a mindent beleadó feladvány elkészítés, hisz a 3 tagjából a csapatnak 3 megy egyetemre, 2 pedig informatikai szakra. Reméljük, hogy ezzel a projectmunkával kitudunk törekedni és tiszta, vagy akár annál is jobb lapokkal kezdhessük elkövetkező tanulmányi éveinket. Megvoltak az ellentétek az év eleje óta tartó projekt munka során, vannak nem megvalósított, elkészült ötleteink de összességében elégedettek vagyunk a minőségével weboldalunknak.