



Rendszerterv

Időseket segítő intelligens közösségi platform

Szoftverarchitektúrák tárgy házi feladat

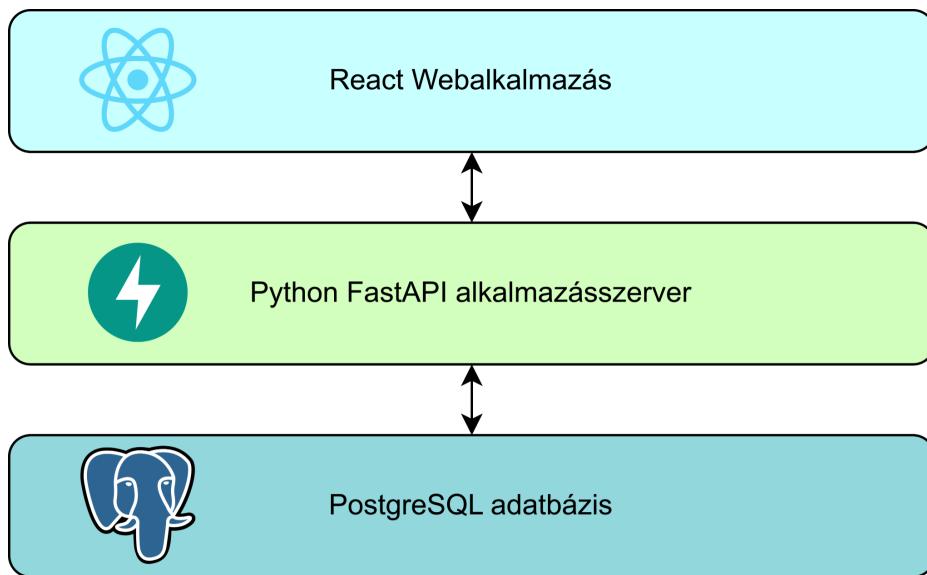
2025

Tartalomjegyzék

Kliensoldali alkalmazás.....	3
Grafikus felhasználói felület.....	3
Célja.....	3
Bejelentkezés és regisztráció.....	3
Navigáció és profilkezelés.....	4
Segélykérő felülete (Help Seeker).....	7
Kérések listázása.....	7
Új kérés létrehozása.....	8
Önkéntes felülete (Volunteer).....	9
Térképes és listás nézet.....	9
Kérés részletei és jelentkezés.....	11
Értékelési rendszer.....	12
Technológiai háttér és megvalósítás.....	13
Alkalmazott technológiák.....	14
Architektúra és működés.....	14
Szerveroldali Alkalmazás.....	15
Célja.....	15
Architektúra.....	15
Adatbázis réteg.....	15
Adatelérési Réteg, Üzleti Logika Rétege.....	17
Autentikáció és Autorizáció.....	19
Jelszókezelés és regisztráció.....	19
Token-alapú hitelesítés (JWT).....	19
Munkamenet-kezelés és token rotáció.....	20
Jogosultságkezelés (RBAC).....	20
REST API Interfész.....	20
Hitelesítés (Auth).....	20
Általános Funkciók (Common).....	21
Segélykérő Végpontjai (Help Seeker).....	21
Önkéntes Végpontjai (Volunteer).....	21
Gamifikáció (Quests).....	22

Bevezetés

Az általunk készített alkalmazás három fő rétegből (komponensből) áll. Ez az architektúra lehetővé teszi, hogy az alkalmazás funkcionális részei logikailag és fizikailag elkülönüljenek, ezáltal biztosítva a karbantarthatóságot, skálázhatóságot és a könnyebb tovább fejleszthetőséget. A komponensek közötti kommunikáció világosan definiált interfészeken keresztül történik.



Ez a három komponens szorosan együttműködik annak érdekében, hogy az alkalmazás teljes funkcionalitása biztosított legyen, ugyanakkor szigorúan elkülönített funkciókat látnak el. Ezen felépítés lehetővé teszi, hogy az alkalmazás moduláris, könnyen karbantartható és technológiailag rugalmas legyen.

Kliensoldali alkalmazás

Grafikus felhasználói felület

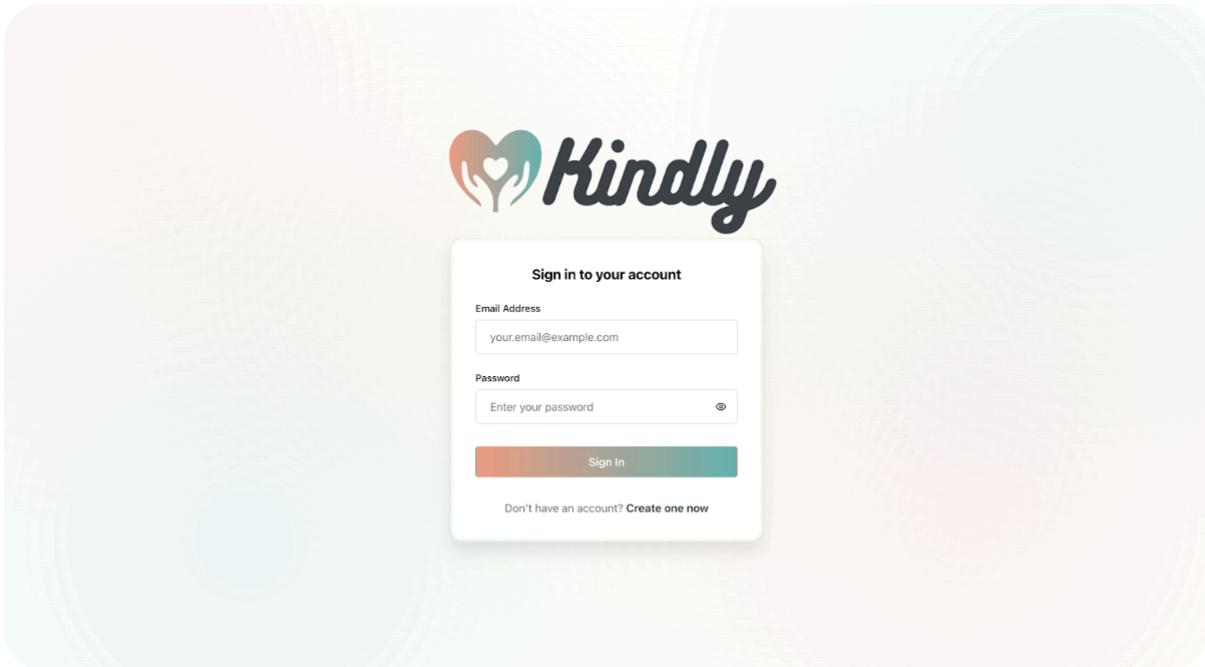
Célja

A rendszer felhasználói felületének célja, hogy letisztult, átlátható és könnyen kezelhető környezetet biztosítson minden idősebb korosztályhoz tartozó segélykérők, minden digitálisan jártas önkéntesek számára. A felületet reszponzív webes technológiákkal készítettük el, biztosítva a könnyű hozzáférést.

A felhasználói interakciók a szerepköröktől (Segélykérő vagy Önkéntes) függően változnak, így az alábbiakban funkcionális csoportokra bontva mutatjuk be a felületet.

Bejelentkezés és regisztráció

A program indításakor a felhasználót a bejelentkezési képernyő fogadja. Itt e-mail cím és jelszó megadásával léphetnek be a rendszerbe.

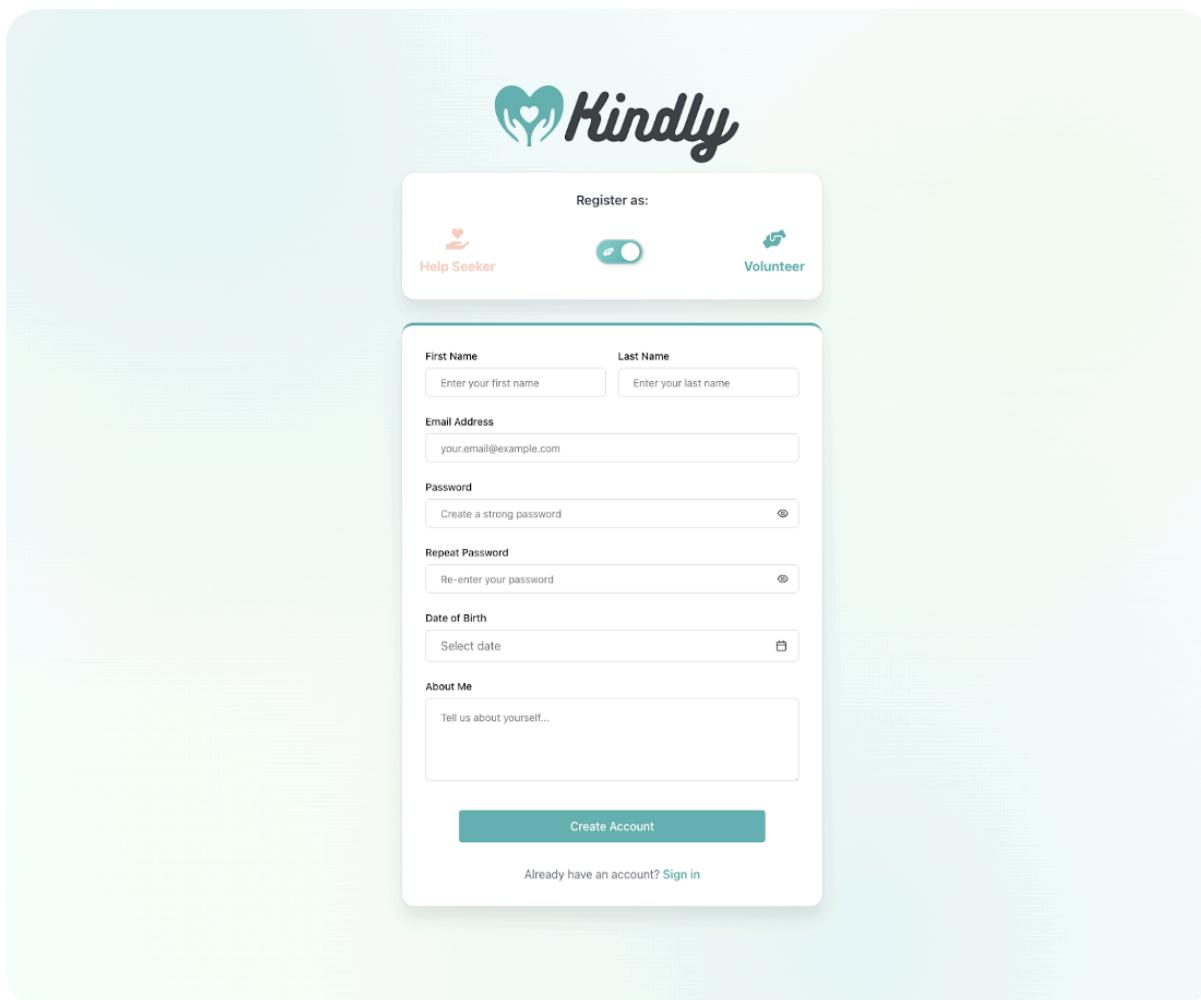


Bejelentkezási felület

Amennyiben a felhasználó még nem rendelkezik fiókkal, átnavigálhat a regisztrációs oldalra. A regisztrációs felületen egy interaktív kapcsoló segítségével választható ki a kívánt szerepkör:

- Segélykérő (Help Seeker): Segítségre szoruló felhasználó.
- Önkéntes (Volunteer): Segítséget nyújtó felhasználó.

Ez a választás meghatározza a későbbi jogosultságokat és az elérhető funkciókat.

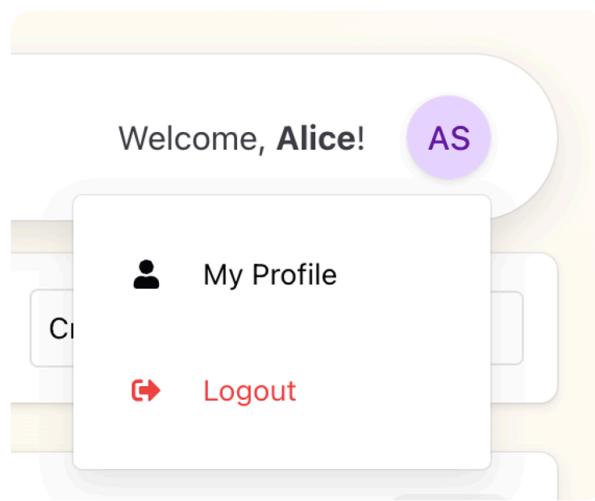


Regisztrációs felület szerepkörválasztással

Navigáció és profilkezelés

A rendszerbe való belépés után a képernyő jobb felső sarkában található Avatar ikon biztosítja a felhasználói fiókhoz kapcsolódó műveleteket. Az ikonra kattintva egy lenyíló menü jelenik meg, amely lehetőséget biztosít:

- A profil oldalra való navigálásra.
- A rendszerből való kijelentkezésre.



Felhasználói fiókhoz kapcsolódó műveletek

A profil oldalon a felhasználó áttekintheti saját vagy mások adatlapját, ahol a legfontosabb információk, mint a név, profilkép, szerepkör és az átlagos értékelés azonnal láthatóak. A gamifikációs elemek hangsúlyosak: nyomon követhető az aktuális szint, a megszerzett tapasztalati pontok (XP) és a következő szint eléréséhez szükséges haladás.

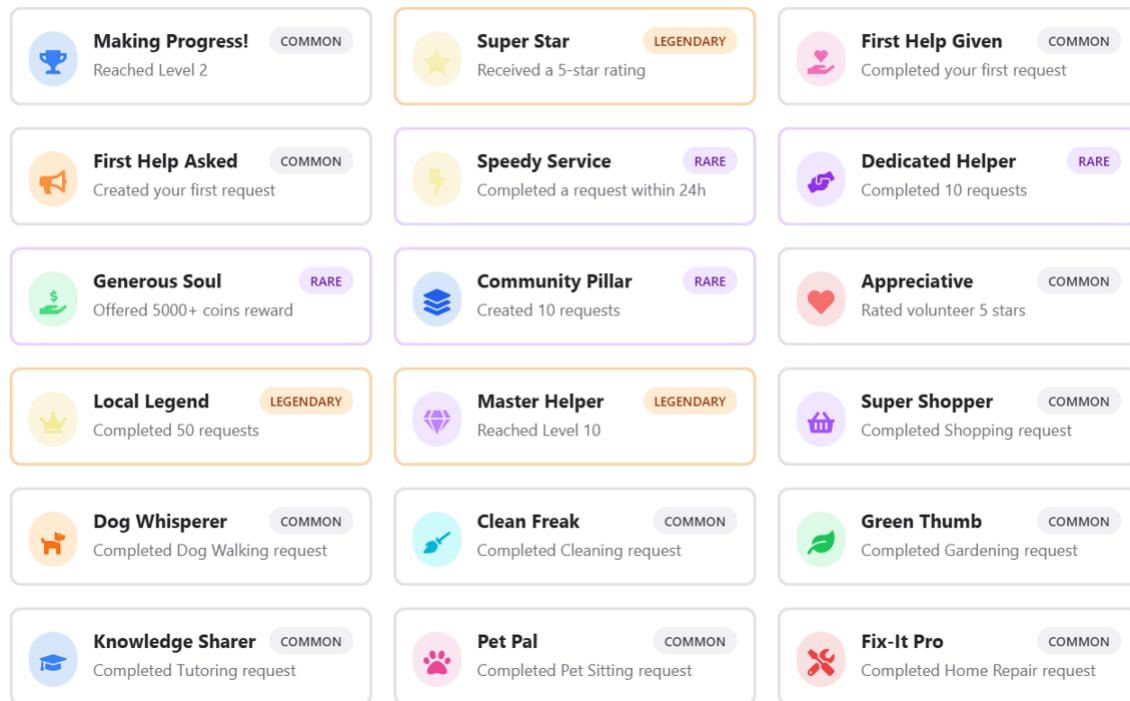
The screenshot shows a user profile for "Alice Smith". At the top, there's a navigation bar with "Back to Requests" on the left, the Kindly logo in the center, and "Welcome, Alice!" with a purple "AS" badge on the right. Below the navigation is a large profile card for Alice Smith. The card features a purple circular icon with "AS", a "Help Seeker" status, a 5.0 rating, and a progress bar indicating she is an "Enthusiast" at level 4, with 340 / 400 XP. The card also shows her date of birth as Jan 1, 1990. Under the "About Me" section, it says "A friendly help seeker.". The "Badges" section displays three achievements: "Making Progress!" (Common, Reached Level 2), "Appreciative" (Common, Rated volunteer 5 stars), and "Super Star" (Legendary, Received a 5-star rating).

Felhasználó profilja, jelvényekkel

A személyes adatok alatt a megszerzett jelvények (Badges) sorakoznak, amelyek ritkaság szerint (Common, Rare, Legendary) kerülnek megjelenítésre. minden egyes jelvény egyedi mérföldkövet vagy különleges teljesítményt szimbolizál, legyen szó egy feladat rekord idő alatti elvégzéséről vagy nagy értékű jutalmak felajánlásáról. A kártyák vizuális kialakítása a ritkaságot jelző színes keretek és feliratok azonnal tükrözi az elért eredmény presztízsét, így

a profil látogatói könnyen felmérhetik a felhasználó elkötelezettségét és a közösségen betöltött szerepét.

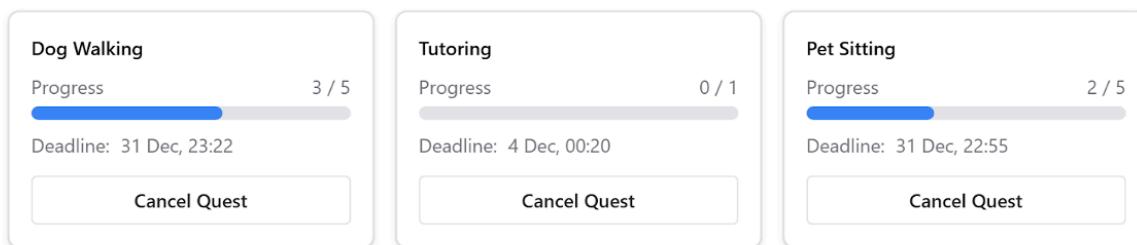
🏆 Badges



Jelvények listája

Saját profil esetén a felhasználó itt tekintheti meg továbbá az aktív küldetéseit is (Quests). Jelenleg a küldetések csak az önkéntesek számára elérhetőek. Céljuk a folyamatos aktivitás ösztönzése. Ezek a kihívások automatikusan követik a felhasználó tevékenységét. Például bizonyos típusú, kategóriájú kérések teljesítését, és valós időben frissülnek a haladási számon. A küldetések sikeres teljesítése tapasztalati pontokkal (XP) jár, amelyek elengedhetetlenek a szintlépéshez és a magasabb rang eléréséhez, ezáltal motiválva a közösség tagjait a rendszeres részvételre.

📋 My Quests



Küldetések áttekintése

A jobb felső sarokban található “Edit Profile” gomb segítségével a felhasználók megtekinthetik, valamint módosíthatják személyes adataikat, így a nevüket, e-mail címüket, születési évüket, valamint az “About me” szekcióban rövid bemutatkozást írhatnak magukról.

Welcome, Alice! AS

Edit Profile

First Name: Alice

Last Name: Smith

Email Address: alice@example.com

Date of Birth: January 1, 1990

About Me: A friendly help seeker.

Cancel Save Changes

Profil szerkesztésének a felülete

Segélykérő felülete (Help Seeker)

A segélykérők számára a fő nézet a saját kéréseik menedzselésére szolgál.

Kérések listázása

A főoldalon a segélykérők listázva láthatják az általuk meghirdetett feladatokat. A lista áttekinthető formában tartalmazza:

- A kérés címét és részleteit.
- A feladat típusát és időpontját.
- Az aktuális státuszt.
- A jelentkezők számát.

A lista támogatja a lapozást, valamint lehetőséget biztosít a kérések rendezésére és szűrésére, segítve a gyors áttekintést.

Welcome, Alice! AS

My Requests

Create New Request

Sort by: Created Date

Filters

All Statuses All Types

Status

All Open Completed

Dusting and Vacuuming Apartment Open

Friendly help needed for general dusting and vacuuming of a small apartment; takes about 1-2 hours.

Cleaning

Nov 28, 2025, 10:37 AM No applicants

SAT Math Tutoring Open

In search of a tutor with experience in prepping students for SAT math. Materials can be provided.

Tutoring

Nov 28, 2025, 10:37 AM No applicants

Grocery Shopping for Senior Open

Looking for a kind volunteer to do weekly grocery runs for a senior citizen.

Shopping

Nov 27, 2025, 10:37 AM No applicants

Light Bulb Replacement Open

Struggling to replace several high-ceiling bulbs. Would appreciate someone with a ladder to help!

Home Repair

Nov 23, 2025, 10:37 AM No applicants

Help with Planting Flowers Open

Would love some help planting new flowers and arranging pots in my garden.

Gardening

Nov 26, 2025, 10:37 AM No applicants

Temporary Pet Sitting Needed Open

Need someone to feed and walk my dog for three days while I am away for work.

Pet Sitting

Nov 27, 2025, 10:37 AM No applicants

Saját kérések listázása és szűrése

Új kérés létrehozása

Új feladat meghirdetésekor egy űrlapot kell kitölteni, ahol megadható a név, leírás, időintervallum (kezdeti és végdátum/idő), valamint az opcionális jutalom mértéke.

A helyszín megadásakor a rendszer segítséget nyújt: a cím gépelésekor egy listából választható ki a validált cím.

The screenshot shows the 'Create Request' interface. At the top, there's a 'Request Name' field with a placeholder 'Enter a descriptive name for your request'. Below it is a 'Description' field with a placeholder 'Provide details about what you need help with'. Under 'Request Types', there's a button labeled 'Generate with AI' which has triggered an AI suggestion for 'Dog Walking'. There's also a dropdown menu labeled 'Select types...'. The 'Location' field contains a placeholder 'Search for an address'. The 'Start Date & Time' and 'End Date & Time' fields both show 'November 21, 2025 at 11:40 AM' and 'November 21, 2025 at 12:40 PM' respectively. The 'Reward (\$)' field contains the value '0'. At the bottom right are two buttons: 'Discard' and 'Create Request'.

Új kérés létrehozása

A feladat típusának meghatározásához a rendszer mesterséges intelligencia támogatást is nyújt. A cím és a leírás alapján az MI ajánlatokat tesz a kategóriára egy felugró ablakban. A felhasználó ezt elfogadhatja, vagy manuálisan módosíthatja, illetve el is utasíthatja az ajánlást.

This is a close-up view of the 'Request Types' section. It shows the 'Generate with AI' button which has suggested 'Dog Walking'. Below it is a button labeled 'Dog Walking X'. Further down is a dropdown menu with the placeholder 'Select types...'. The background of this section is light gray.

Mesterséges intelligencia által javasolt kategóriák

Amennyiben egy feladatra önkéntesek jelentkeznek, a segélykérő a segélykérés részletes nézetén tekintheti meg a jelentkezőket, majd egy dialógus ablakban választhatja ki azt a személyt, aki elvégzi a feladatot. A munka végeztével a segélykérő ugyanitt jelölheti késznek a feladatot. Késznek jelölt feladatok esetén lehetősége van értékelni az önkéntes munkáját. A kéréseket szerkeszteni is tudja mindenkor, amíg nem jelentkezett önkéntes az adott kérésre. A rendszer a feladat lezárását követően a megadott jutalomösszeg 10%-át automatikusan jóváírja az önkéntes és a segélykérő tapasztalati pontjai (XP) között, így is buzdítva a felhasználókat az aktivitásra.

The screenshot shows a detailed request card for an apartment cleaning task. At the top, there's a navigation bar with 'Back to Requests' and the Kindly logo. On the right, it says 'Welcome, Alice!' and has a purple profile icon. The main card has a title 'Apartment cleaning' with an 'Edit' button. It includes sections for 'Description' (looking for someone to clean the apartment), 'Location' (123 Main St, Anytown), 'Schedule' (Start: November 29, 2025 at 10:37 AM; End: November 29, 2025 at 12:37 PM), 'Reward' (\$932), 'Request Types' (Cleaning), and 'Applicants (1)' (Bob Johnson). Bob Johnson is listed with a green profile picture, 'No ratings yet', and a green checkmark. At the bottom of the card is a 'Mark as completed' button.

Segélykérés részletes nézete

Önkéntes felülete (Volunteer)

Térképes és listás nézet

Az önkéntesek kétféle módon böngészhetik feladatokat:

- Térképes nézet:** A kérések földrajzi elhelyezkedés alapján, marker pontokként jelennek meg a térképen. A markerre kattintva egy információs ablakban megjelennek a legfontosabb adatok, ahonnan közvetlenül a részletes nézetre navigálhatnak.

The screenshot shows the Kindly app's user interface. At the top, there's a header with the Kindly logo, a search bar, and a "Browse Requests" button. On the right, it says "Welcome, Bob!" and has a profile icon. Below the header is a sidebar titled "Filters" with sections for "All Statuses" and "All Types". It includes a "Price Range" slider set to "All prices", a "Status" section with "All" selected, and a "Type" dropdown set to "Select types...". The main area is a map of Budapest with several green location pins. A modal window is open in the center, titled "Grocery shopping", which reads: "Looking for a volunteer to pick up groceries and deliver them to my home. List will be provided." Below this is a timestamp "\$1056 Dec 1, 2025, 10:37 AM" and a "View Details" button. The map also features a legend in the bottom-left corner: "Legend" with "Open Requests" (green dot) and "Completed" (grey dot).

Feladatok megjelenítése térképen

2. **Listás nézet:** A feladatok hagyományos listában is böngészhetők. Itt fejlett szűrési lehetőségek állnak rendelkezésre:
 - a. Ár/Jutalom tartomány.
 - b. Státusz.
 - c. Feladat típusa.

The screenshot shows the Kindly app's "Browse Requests" page. At the top, there are filters for "All Statuses" and "All Types". A "Price Range" slider is set to "All prices". Under "Status", "All" is selected. In the "Type" section, there is a dropdown menu with "Select types..." and a "Type" button. The main area displays a grid of 10 request cards:

- General Gardening Help** (Open): Need a hand weeding flower beds and trimming shrubs in my back yard. **Gardening**. \$1108 applied 0. Date: Nov 22, 2025, 10:37 AM.
- English Tutoring for Adult** (Open): Looking for an English tutor to help me improve my conversational skills. Flexible schedule. **Tutoring**. \$1125 applied 0. Date: Nov 23, 2025, 10:37 AM.
- Light Bulb Replacement** (Open): Struggling to replace several high-ceiling bulbs. Would appreciate someone with a ladder to help! **Home Repair**. \$945 applied 0. Date: Nov 23, 2025, 10:37 AM.
- Deep Clean Bathroom & Kitchen** (Open): Looking for detailed cleaning of bathroom and kitchen; disinfecting and organizing needed. Must like pets. **Cleaning**. \$972 applied 0. Date: Nov 25, 2025, 10:37 AM.
- Pet Sitting Over the Weekend** (Open): Need someone to look after my two cats from Friday evening to Sunday night, including feeding and playtime. **Pet Sitting**. \$1008 applied 0. Date: Nov 26, 2025, 10:37 AM.
- Help with Planting Flowers** (Open): Would love some help planting new flowers and arranging pots in my garden. **Gardening**. \$1068 applied 0. Date: Nov 26, 2025, 10:37 AM.
- Lawn Mowing Help Needed** (Applied): Need assistance mowing the front and back lawns. Preferably bring your own mower. **Gardening**. \$1002 applied 1. Date: Nov 27, 2025, 10:37 AM.
- Temporary Pet Sitting Needed** (Open): Need someone to feed and walk my dog for three days while I am away for work. **Pet Sitting**. \$1024 applied 0. Date: Nov 27, 2025, 10:37 AM.
- Grocery Shopping for Senior** (Open): Looking for a kind volunteer to do weekly grocery runs for a senior citizen. **Shopping**. \$1108 applied 0. Date: Nov 27, 2025, 10:37 AM.

At the bottom, it says "Showing 1 - 9 of 17 requests" with navigation arrows and a page number 1.

Feladatok megjelenítése lista nézetben

Kérés részletei és jelentkezés

A részletes nézetben az önkéntes minden információt lát a feladatról: ki hirdette meg, pontos leírás, helyszín, időpont, jutalom és a jelentkezők száma.

A részvételi szándékot az “*Apply to help*” gomb megnyomásával jelezhetik. Ha már jelentkeztek, a felületen visszajelzést kapnak a jelentkezés tényéről. Amennyiben a hirdető elfogadja a jelentkezésüket, szintén itt láthatják a visszajelzést.

The screenshot shows a detailed request card on the Kindly platform. At the top right, there's a welcome message "Welcome, Bob!" and a green circular badge with the letters "BJ". The main title of the request is "Lawn Mowing Help Needed". To the right of the title, it says "Alice Smith" and "★ 5.0".
Description: Need assistance mowing the front and back lawns. Preferably bring your own mower.
Location: 123 Main St, Anytown
Schedule: Start: November 27, 2025 at 10:37 AM; End: November 27, 2025 at 12:37 PM
Reward: \$1002
Request Types: Gardening
Applicants: No applicants yet
Buttons: A large teal button at the bottom center labeled "Apply to Help" with a person icon.

Részletes nézet és jelentkezés gomb

Értékelési rendszer

A bizalom építése érdekében a rendszer kétirányú értékelést tesz lehetővé. Amint a segélykérő "Completed" (Befejezett) státuszba állít egy kérést:

1. A részletes nézeten megjelenik a véleményezés lehetősége.

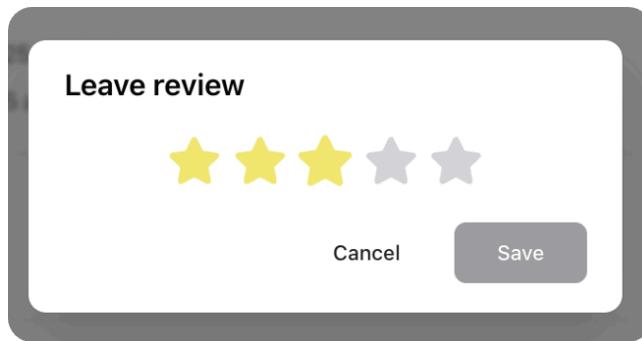
The screenshot shows a task request for "Apartment cleaning". The request details are as follows:

- Description:** Looking for someone to clean my apartment, including floors, kitchen, and bathroom. Must bring your own supplies.
- Location:** 123 Main St, Anytown
- Schedule:**
 - Start: November 29, 2025 at 10:37 AM
 - End: November 29, 2025 at 12:37 PM
- Reward:** \$932
- Request Types:** Cleaning
- Applicants:** 1 applicant

A green button at the bottom of the request card says "This request has been completed".

Lezárt feladat és értékelés leadása gomb

2. Egy felugró dialógusablakban a felek 1-től 5 csillagig terjedő skálán értékelhetik a másik fél együttműködését.



Értékelés leadása

Az így kapott értékelés tízszeres szorzóval kerül jóváírásra tapasztalati pontokban (XP) minden fél számára, így ösztönözve a feladatok lelkismeretes elvégzését.

Technológiai háttér és megvalósítás

A felhasználói felület modern, komponens-alapú architektúrára épül, amely biztosítja a gyors működést, a skálázhatóságot és a könnyű karbantarthatóságot. A fejlesztés során a legfrissebb iparági sztenderdeket követtük.

Alkalmazott technológiák

A frontend alkalmazás alapját a *React* keretrendszer (v19) adja, amely lehetővé teszi a dinamikus és interaktív felületek létrehozását. A fejlesztői környezet és a build folyamatok gyorsaságát a *Vite* biztosítja. A kódminőség és a típusbiztonság érdekében *TypeScript* nyelvet használtunk.

A rendszer főbb építőelemei:

- **Stílus és UI keretrendszer:** *Chakra UI* (v3) - Ez biztosítja a reszponzív, akadálymentes és egységes megjelenést. A dizájnrendszer testreszabott témát használ, amely definiálja az alkalmazás színpalettáját és tipográfiáját.
- **Útvonalválasztás (Routing):** *React Router* (v7) - Felelős az oldalak közötti navigációért és a védett útvonalak kezeléséért, biztosítva, hogy illetéktelen felhasználók ne érhessék el a belső funkciókat.
- **Állapotkezelés:** A globális állapotokat (pl. bejelentkezett felhasználó adatai) a *React Context API* kezeli.
- **Ürlapkezelés:** *React Hook Form* és *Zod* - Ezek kombinációja teszi lehetővé a komplex ürlapok (pl. regisztráció, kérés létrehozása) hatékony kezelését és validációját.
- **Térképszolgáltatás:** *Mapbox GL* - Interaktív térképes nézetek megjelenítésére és címkeresésre szolgál.

Architektúra és működés

Az alkalmazás forráskódja funkcionális egységekre tagolódik, ami elősegíti az átláthatóságot.

1. Kommunikáció a szerverrel (API réteg)

A backend rendszerrel való kommunikációt az *Axios* könyvtár végzi. A biztonságos adatcsere érdekében egy központi API szolgáltatást alakítottunk ki, amely automatikusan kezeli:

- A hitelesítési tokenek (JWT) csatolását minden kéréshez.
- A lejárt tokenek csendes frissítését (Silent Refresh) a háttérben, így a felhasználónak nem kell folyamatosan újra bejelentkeznie.
- A hibák egységes kezelését és a felhasználó átirányítását szükség esetén.

2. Biztonság és Hitelesítés

A bejelentkezési folyamat és a jogosultságok kezelése központosítottan történik. A *ProtectedRoute* komponens gondoskodik arról, hogy a csak regisztrált felhasználók számára elérhető oldalak (pl. /requests, /profile) védve legyenek. A rendszer különbséget tesz a szerepkörök között (Segélykérő vs. Önkéntes), és ennek megfelelően jeleníti meg vagy rejt el az egyes funkciókat.

3. Felhasználói Élmény (UX)

A felület tervezésekor kiemelt figyelmet fordítottunk a visszajelzésekre:

- Töltési állapotok: Az adatok betöltése közben vizuális indikátorok jelzik a folyamatot.
- Értesítések: A felhasználói műveletek sikerességéről vagy sikertelenségről felugró üzenetek tájékoztatnak.

Szerveroldali Alkalmazás

Célja

A szerveroldali alkalmazás alapvető célja, hogy megbízható, biztonságos és skálázható háttérteret biztosítson a közösségi platform működéséhez, központi egységként kezelve a rendszer üzleti logikáját és adatforgalmát. Legfontosabb feladatai közé tartozik a felhasználók, a meghirdetett feladatok és a beérkező értékelések teljes életciklusának menedzselése, valamint az ezekhez tartozó adatok központi tárolása és kezelése.

A rendszer felel a biztonságos működésért: elvégzi a felhasználók hitelesítését, gondoskodik a jelszavak védett tárolásáról, és szigorú jogosultságkezeléssel biztosítja, hogy a segélykérők és önkéntesek csak a számukra engedélyezett funkciókhoz és adatokhoz férjenek hozzá. Emellett a szerver biztosítja a folyamatos kommunikációt a felhasználói felülettel, fogadja és feldolgozza a kliensoldalról érkező kéréseket, valamint kiszolgálja az intelligens funkciókat, például a feladatok kategóriájának automatikus, mesterséges intelligencia általi meghatározását.

Architektúra

A Kindly szolgáltatás szoftverarchitektúrája a klasszikus N-rétegű tervezési minta alapelveire épül, amely három elkülöníthető modulra tagolódik:

- adatbázisréteg (Database Layer, DB)
- adatelérési réteg (Data Access Layer, DAL)
- üzleti logika rétege (Business Logic Layer, BLL)

Tekintettel arra, hogy a rendszer megvalósítása során az adatelérési és az üzleti logikai réteg funkcionálisan szorosan kapcsolódik egymáshoz, a dokumentáció a továbbiakban ezeket nem különálló egységekként, hanem összevontan tárgyalja. A következő fejezetek ezen architektúrális rétegek részletes bemutatását tartalmazzák.

Adatbázis réteg

Az adatbázis réteg feladata az adatok perzisztens kezelése. Ebben a fejezetben ennek részleteit fogjuk bemutatni és tervezési lépéseit. A rendszer alapjául a PostgreSQL relációs adatbázis-kezelő rendszert választottuk a PostGIS kiegészítővel amely térinformatikai képességekkel bővíti ki az adatbázis kezelő rendszert. Így az adatbázis réteg megbízhatósága és fejlett adattípusai révén kiválóan alkalmas a komplex üzleti logika és a térinformatikai adatok kiszolgálására.

A fizikai adatmodell tervezése során a normalizálás elveit követtük a redundancia minimalizálása és az adatintegritás biztosítása érdekében. A központi entitás a felhasználó (`user`), amely egyetlen táblában egyesíti a rendszer összes szereplőjét. A specifikációban meghatározott szerepkörök (Segélykérő és Önkéntes) megkülönböztetése az `is_volunteer` logikai mező segítségével történik. Ez az egységes megközelítés

egyszerűsíti a hitelesítési folyamatokat, miközben a profilhoz kapcsolódó egyéb adatok – mint az elérhetőség (email), a bemutatkozás (about_me), vagy a rendszer által számított megbízhatósági szint (avg_rating, level, experience), valamint a megszerzett elismeréseket tartalmazó badges mező – mindenki számára elérhetők. A biztonságos hitelesítést a jelszavak mellett a különálló refresh_token tábla támogatja, amelynek szükségességét és fontosságát egy későbbi fejezetben tárgyaljuk.



Az adatbázis egyed-kapcsolat diagramja.

A rendszer legfontosabb funkcionális elemét a kérések (request) tábla valósítja meg, amely tárolja a segélykérés minden lényeges paraméterét: a címet, leírást, a felkínált jutalmat (reward), valamint az időbeli korlátokat (start, end). Kiemelt jelentőséggel bír a térképes funkciók támogatása, amelyet a táblában tárolt földrajzi koordináták (longitude, latitude), valamint a speciális location mező amely kombinálja a hosszúsági és szélességi fokokat egy egyedi adattípusba, lehetővé téve a hatékony térbeli lekérdezéseket a „közelben lévő” feladatokhoz. A kérések életciklusát a status mező követi nyomon, míg a lekérdezések optimalizálása érdekében az application_count mező denormalizált formában tárolja az aktuális jelentkezők számát. A kérések kategorizálása érdekében egy kapcsolótáblát (type_of) iktattunk be, amely több-a-több (N:M) kapcsolatot valósít meg a kérések és a rögzített kategóriák (request_type) között, így egy feladat több típusba is tartozhat egyszerre.

A felhasználók közötti interakciót az application tábla rögzíti, amely összeköti az önkéntest a kiválasztott feladattal. Ez a tábla nem csupán a jelentkezés tényét tárolja, hanem annak teljes életciklusát is követi a status mező segítségével (pl. jelentkezett,

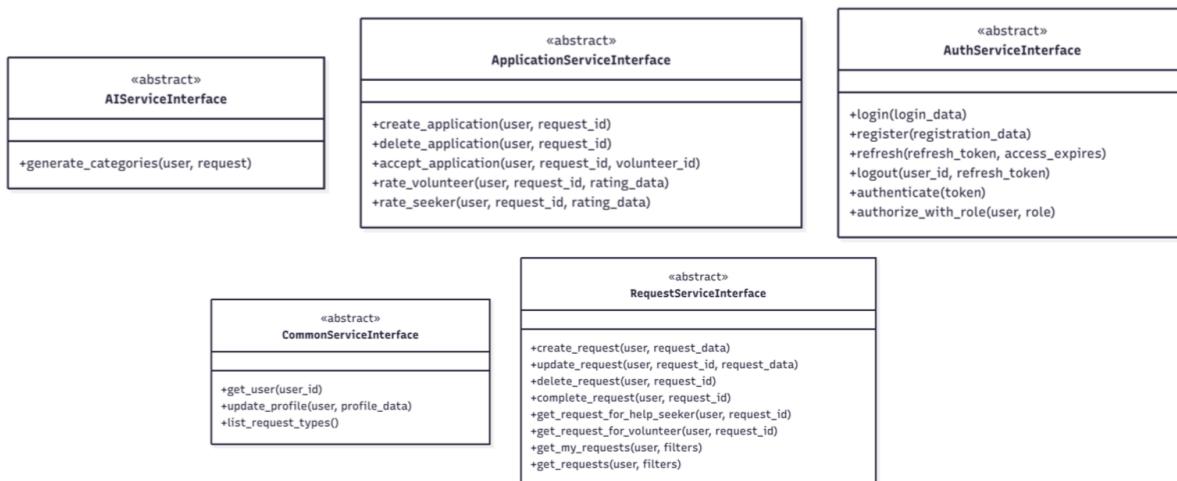
elfogadva). Szintén itt kerülnek tárolásra a feladat lezárását követő kölcsönös értékelések (`volunteer_rating`, `help_seeker_rating`), amelyek alapjául szolgálnak a bizalmi indexek számításának.

A rendszer egyik gamifikációs rétegét a küldetések (quest) tábla valósítja meg, amely a felhasználók aktivitását hivatott ösztönözni. Ez az entitás összeköti a felhasználót (`user_id`) egy specifikus feladattípussal (`request_type_id`), miközben definiálja a teljesítendő célt (`target_count`), követi az aktuális haladást (`current_count`), és rögzíti a teljesítés határidejét (`deadline`).

Az adatok konziszenciáját a táblák között definiált idegen kulcsok (Foreign Keys) és a szigorú típusosság (pl. `timestamptz` az időzónák kezelésére, `serial4` az egyedi azonosítóhoz) garantálja.

Az alkalmazás üzleti logikája és a fizikai adatbázis közötti kapcsolatot, valamint az objektum-relációs leképezést (ORM) az SQLAlchemy könyvtár biztosítja. A megvalósítás során a relációs séma minden egyes tábláját egy-egy dedikált Python osztály reprezentálja a forráskódban. Ezeket a modelosztályokat olyan speciális metaadatokkal láttuk el, amelyek pontosan definiálják az adatbázistáblákra való leképezés szabályait, így teremtve összhangot az objektumorientált programkód és az adattárolási réteg között.

Adatelérési Réteg, Üzleti Logika Rétege



Az egyes szolgáltatások által definiált interfészek.

Az alkalmazás üzleti logikáját (BLL) és az adatelérési funkciókat (DAL) szorosan együttműködő, moduláris szolgáltatások (Service-ek) valósítják meg. A rétegek közötti kommunikációt absztrakt interfészök definálják, amelyek biztosítják a kód tesztelhetőségét és a függőségek egyértelmű kezelését. A rendszer főbb üzleti folyamatai az alábbi szolgáltatások köré csoportosulnak:

- **AuthServiceInterface – Hitelesítés és Jogosultságkezelés** A rendszer biztonsági kapuőre, amely a felhasználók beléptetését és azonosítását végzi.

- Kezeli a regisztrációt (register) és a bejelentkezés folyamatát (login), JWT (JSON Web Token) alapú hitelesítést biztosítva.
- Felelős a munkamenetek fenntartásáért a tokenek frissítésével (refresh), valamint a kijelentkeztetésért (logout).
- Implementálja a szerepkör-alapú védelmet (authorize_with_role), garantálva, hogy a segélykérők és önkéntesek szigorúan elkülönített funkciókat érjenek el.
- **RequestServiceInterface – Segélykérések Menedzszerese** A platform központi elemét, a feladatok teljes életciklusát kezelő modul.
 - Biztosítja a kérések létrehozását (create_request), módosítását és törlését, validálva, hogy ezeket csak a jogosult segélykérő végezhesse el.
 - Szerepkör-specifikus nézeteket nyújt: a segélykérők számára elérhetővé teszi a jelentkezők adatait (get_request_for_help_seeker), míg az önkéntesek számára a feladat publikus részleteit (get_request_for_volunteer).
 - Kezeli a feladatlisták szűrését és lapozását (get_requests), támogatva a keresést státusz, típus vagy jatalom alapján.
 - Lehetővé teszi a feladatok lezárását (complete_request) a munka elvégzése után.
- **ApplicationServiceInterface – Jelentkezés és Interakció** Ez a szolgáltatás köti össze a felhasználókat a feladatokkal.
 - Kezeli az önkéntesek jelentkezését egy adott feladatra (create_application), valamint a visszavonást.
 - Támogatja a kiválasztási folyamatot, ahol a segélykérő elfogadhatja a jelentkezőt (accept_application), ezzel hozzárendelve a feladatot az önkénteshez.
 - Implementálja a kétirányú értékelési rendszert (rate_volunteer, rate_seeker), amely lehetővé teszi a felek számára a kölcsönös visszajelzést a munka lezárása után.
- **CommonServiceInterface – Általános Felhasználói Funkciók** A felhasználók személyes adatainak és a rendszer törzsadatainak kezelője.
 - Lehetőséget biztosít a profiladatok lekérdezésére (get_user) és módosítására (update_profile), beleértve a bemutatkozást is.
 - Szolgáltatja a rendszerben elérhető feladatkategóriák listáját (list_request_types) a kérések létrehozásához és szűréséhez.
- **AIServiceInterface – Intelligens Támogatás** A rendszer mesterséges intelligencia funkcióit magába foglaló modul.
 - A generate_categories metóduson keresztül elemzi a feladat címét és leírását, majd automatikusan javaslatokat tesz a legmegfelelőbb kategóriákra, ezzel segítve a felhasználót a kitöltésben.

- **QuestServiceInterface – Gamifikáció és Ösztönzés** A felhasználói elköteleződés növelését és a közösségi aktivitás fenntartását célzó gamifikációs réteg kezelője.
 - Nyomon követi a felhasználókhoz rendelt küldetések (quests) állapotát és a kitűzött célok (pl. meghatározott számú segítségnyújtás) teljesülését.
 - Vezérli a jutalmazási rendszert: automatikusan oszt ki jelvényeket (badges) és elismeréseket mérföldkövek elérésekor (pl. „First Help Asked”, „Community Pillar”), amelyeket a RequestService és AuthService eseményei válthatnak ki.
 - Kezeli a tapasztalati pontok (XP) és a felhasználói szintek számítását, közvetlen visszajelzést adva a felhasználóknak a platformon végzett tevékenységek értékéről.

Autentikáció és Autorizáció

A rendszer biztonságának gerincét az AuthService komponens alkotja, amely a modern webes alkalmazások elvárásainak megfelelően, teljes mértékben JWT-alapú, állapotmentes hitelesítési és szerepkör-alapú jogosultságkezelési rendszert valósít meg. Ennek középpontjában a felhasználók megbízható azonosítása és a védett erőforrásokhoz való hozzáférés precíz szabályozása áll.

Jelszókezelés és regisztráció

A felhasználói fiókok védelme érdekében a jelszavak soha nem kerülnek tárolásra nyílt szöveges formában. A regisztráció során a rendszer a pwdlib könyvtár Argon2id algoritmusát használja – amely jelenleg a jelszótitkosítás egyik legbiztonságosabb és leginkább ajánlott megoldása – a jelszavak egyirányú hash-elésére. Az így generált hash automatikusan egyedi, véletlenszerű salt-tal együtt kerül mentésre, ezzel gyakorlatilag lehetetlenne téve a jelszavak visszafejtését vagy szótáras támadásokkal történő feltörését.

Token-alapú hitelesítés (JWT)

A bejelentkezés és a későbbi azonosítás teljes mértékben állapotmentes, HS256 algoritmussal aláírt JSON Web Tokenekre épül. A rendszer kettős token stratégiát követ:

1. **Access Token** – Rövid élettartamú (általában 5–15 perc), a kliensoldali kérések azonnali hitelesítésére szolgál. Payloadjában szerepel a felhasználó egyedi azonosítója (id), e-mail címe és szerepköre (is_volunteer), így a kérések feldolgozása során nincs szükség adatbázis-lekérdezésre a jogosultság ellenőrzéséhez.
2. **Refresh Token** – Hosszabb élettartamú token, amelynek hash-elt lenyomatát a rendszer külön RefreshToken táblában tárolja. Ez lehetővé teszi a tokenek egyedi visszavonását biztonsági incidens esetén: csak azok a refresh tokenek tekinthetők érvényesnek, amelyek szerepelnek az adatbázisban.

A hitelesítési folyamat során a beérkező token a szerveroldali titkos kulccsal (JWT_SECRET) kerül dekódolásra és ellenőrzésre: aláírás, lejárat idő (exp), kiállító (iss) és egyebek.

Munkamenet-kezelés és token rotáció

A felhasználói munkamenet folytonosságát a refresh végpont biztosítja, amely nem csupán új access tokent állít ki, hanem biztonsági okokból rotálja a refresh tokent is: a régi token azonnal érvénytelenné válik, helyette egy teljesen új kerül kiadásra és mentésre. Ez jelentősen csökkenti egy esetlegesen ellenőrzésre kerülő refresh token felhasználhatóságának időablakát.

A kijelentkezés (logout) szerveroldali művelet: a hozzá tartozó refresh token fizikailag törlésre kerül az adatbázisból, ezzel azonnal és visszavonhatatlanul megszakítva a munkamenet megújításának lehetőségét.

Jogosultságkezelés (RBAC)

A rendszer szerepkör-alapú hozzáférés-szabályozást (Role-Based Access Control) alkalmaz. A szükséges jogosultsági információ – hogy a felhasználó önkéntes vagy segélykérő – már az access token payloadjában megtalálható (is_volunteer flag), így az authorize_with_role metódus adatbázis-lekérdezés nélkül, azonnal eldöntheti, hogy a kérés jogosult-e az adott erőforrás elérésére. Amennyiben a felhasználó szerepköre nem felel meg az endpoint által megkövetelt minimális szintnek, a rendszer 403 Forbidden válassza megfelelő kivétellel tagadja meg a hozzáférést.

REST API Interfész

A rendszer a külvilág felé egy jól strukturált, verziót tartalmazó RESTful API-t biztosít, amely a /api/v1 prefix alatt érhető el. A végpontok tervezésekor az erőforrás-orientált megközelítést alkalmaztuk, ahol az URL-ek az entitásokat (pl. felhasználók, kérések), a HTTP igék pedig a műveleteket (GET, POST, PUT, DELETE, PATCH) jelölik. Az alábbiakban funkcionális csoportonként ismertetjük az interfészt.

Hitelesítés (Auth)

A rendszerbe való belépés és a munkamenetek kezelése a /api/v1/auth útvonalcsoporthoz kapcsolódik.

- **POST /login:** Felhasználó bejelentkeztetése e-mail és jelszó alapján. Sikeres azonosítás esetén a rendszer Access és Refresh tokeneket ad vissza.
- **POST /register:** Új felhasználói fiók létrehozása. A regisztráció során a felhasználó megadja szerepkörét (Segélykérő vagy Önkéntes).
- **POST /refresh:** A lejáró Access token megújítása érvényes Refresh token segítségével, biztosítva a folyamatos munkamenetet.
- **POST /logout:** Kijelentkezés, amely érvényteleníti a Refresh tokenet szerveroldalon.

- **POST /login-form:** Alternatív bejelentkezési végpont, amely támogatja a application/x-www-form-urlencoded formátumot (pl. OpenAPI UI-hoz).

Általános Funkciók (Common)

A minden bejelentkezett felhasználó számára elérhető, közös funkciók a /api/v1/common alatt találhatók.

- **GET/PUT /profile:** A bejelentkezett felhasználó saját profiladatainak (név, bemutatkozás) lekérdezése és módosítása.
- **GET /users/{user_id}:** Más felhasználók publikus profiladatainak megtekintése (pl. önkéntes adatlapja).
- **GET /request-types:** A rendszerben definiált feladatkategóriák (pl. Bevásárlás, Kertészkedés) listázása, amely szükséges a kérések létrehozásához és szűréséhez.

Segélykérő Végpontjai (Help Seeker)

A segélykérők speciális üzleti folyamatait kiszolgáló végpontok a /api/v1/help-seeker névtérben kaptak helyet. Ezekhez csak a megfelelő szerepkörrel rendelkező felhasználók férhetnek hozzá.

- **Feladatok Kezelése:**
 - **POST /requests/:** Új segítségkérés meghirdetése.
 - **GET /requests/:** A saját, meghirdetett kérések listázása, szűrési lehetőségekkel.
 - **GET/PUT/DELETE /requests/{request_id}:** Egy konkrét kérés részleteinek megtekintése, adatainak módosítása vagy törlése (amíg nincs jelentkező).
- **Interakciók és Életciklus:**
 - **POST /requests/generate-categories:** AI támogatás igénybevétele a kategóriák automatikus meghatározásához cím és leírás alapján.
 - **PATCH /requests/{request_id}/applications/{volunteer_id}/accept:** Jelentkező elfogadása, amellyel a feladat hozzárendelésre kerül az önkénteshez.
 - **PATCH /requests/{request_id}/complete:** A feladat manuális lezárása „Teljesített” állapotba.
 - **POST /requests/{request_id}/rate-volunteer:** A segítő önkéntes értékelése a munka végeztével.

Önkéntes Végpontjai (Volunteer)

Az önkéntesek számára dedikált funkciók a /api/v1/volunteer útvonalon érhetők el.

- **Böngészés:**

- **GET /requests/**: Az összes elérhető, nyitott feladat listázása és keresése szűrők (pl. távolság, típus) segítségével.
- **GET /requests/{request_id}**: Egy kiválasztott feladat publikus részleteinek megtekintése.
- **Jelentkezés és Értékelés:**
 - **POST /requests/{request_id}/application**: Jelentkezés leadása egy kiválasztott feladatra.
 - **DELETE /requests/{request_id}/application**: Korábbi jelentkezés visszavonása.
 - **POST /requests/{request_id}/rate-seeker**: A segélykérő értékelése a közös munka lezárása után.

Gamifikáció (Quests)

A felhasználók motiválását szolgáló gamifikációs funkciók a /api/v1/quests alatt érhetők el.

- **GET /quests**: A bejelentkezett felhasználó aktív küldetéseinek listázása. A válasz tartalmazza a küldetés típusát, a célértéket (target_count), a jelenlegi haladást (current_count) és a határidőt.
- **POST /quests/{quest_id}/cancel**: Lehetőséget ad egy folyamatban lévő küldetés megszakítására, amennyiben a felhasználó nem kívánja azt teljesíteni.