***Funkciók és szolgáltatások bővítése***

## 1.1 Dinamikus árfigyelés

### Célkitűzés

A dinamikus árfigyelés funkció elsődleges célja, hogy a felhasználók számára folyamatosan frissülő, megbízható és naprakész információkat biztosítson az utazással kapcsolatos költségek alakulásáról. Az utazás tervezése során az egyik legfontosabb szempont a költségek optimalizálása, különösen akkor, ha a felhasználó rugalmasan tudja alakítani az indulás időpontját, az úti célt vagy a programokat. A rendszer úgy lett kialakítva, hogy automatikusan nyomon kövesse a különböző utazási elemek – például repülőjegyek, szállások és programok – árváltozásait, és ezekről időben, jól érthető formában értesítést küldjön a felhasználónak.

Ez a funkció nem csupán kényelmi szolgáltatásként működik, hanem aktívan hozzájárul a költséghatékony utazástervezéshez. A felhasználók számára lehetőséget biztosít arra, hogy az árak alakulását figyelembe véve hozzanak döntéseket, például mikor érdemes lefoglalni egy szállást vagy repülőjegyet. A rendszer által küldött értesítések segítenek abban, hogy a felhasználó ne maradjon le egy kedvező ajánlatról, és ne kelljen naponta manuálisan ellenőriznie az árakat. Ezáltal a döntéshozatal nemcsak gyorsabbá, hanem megalapozottabbá is válik.

A dinamikus árfigyelés tehát egy olyan intelligens modul, amely az utazási költségek optimalizálásán túl hozzájárul a felhasználói élmény növeléséhez, az időgazdálkodás hatékonyságához, valamint a tudatos tervezéshez. A funkció különösen hasznos lehet azok számára, akik csoportosan utaznak, vagy több alternatívát mérlegelnek az utazás során.

### Funkcionális leírás

A dinamikus árfigyelés modul több egymással összefüggő funkcióból áll, amelyek együttesen biztosítják a rendszer teljes körű működését. Az alábbiakban részletezve bemutatjuk a modul főbb funkcióit:

* **Árfigyelés repülőjegyekre** A rendszer közvetlenül kapcsolódik különböző repülőjegy-szolgáltatókhoz, például légitársaságokhoz és aggregátor platformokhoz, API-n keresztül. A felhasználó által megadott útvonalakra és időpontokra vonatkozóan a rendszer folyamatosan monitorozza az aktuális árakat, és rögzíti az esetleges változásokat. Ez lehetővé teszi, hogy a felhasználó az optimális ár eléréséhez időben tudjon foglalni. A funkció támogatja az egyirányú, retúr és több szakaszos útvonalakat is, valamint képes figyelembe venni az átszállások számát és időtartamát.
* **Szállásárak követése** A modul integrálva van olyan népszerű szállásfoglaló platformokkal, mint például a Booking.com vagy az Airbnb. A felhasználó által kiválasztott szálláshelyek árváltozásait a rendszer automatikusan rögzíti, és összehasonlítja a korábbi árakkal. Ez segít abban, hogy a felhasználó a legkedvezőbb időpontban tudja lefoglalni a szállást. A rendszer képes megkülönböztetni a különböző szobatípusokat, szolgáltatásokat és lemondási feltételeket, így a kalkulált árak pontosan tükrözik a választott opciókat.
* **Programok és belépők figyelése** A rendszer képes nyomon követni különféle programok – például kulturális események, múzeumi belépők, túrák – jegyáraiban bekövetkező változásokat. Ez különösen hasznos lehet azok számára, akik előre szeretnék megtervezni programjaikat, és szeretnék elkerülni a drágább időszakokat. A funkció támogatja a helyszín szerinti szűrést, az esemény típusát, valamint a jegytípusok (pl. felnőtt, diák, csoportos) árainak összehasonlítását is.
* **Értesítési rendszer** A rendszer többféle értesítési formát támogat, beleértve az e-mailes értesítéseket, push üzeneteket mobil eszközökre, valamint webes felugró ablakokat. Az értesítések célja, hogy a felhasználó azonnal tudomást szerezzen az árak csökkenéséről vagy emelkedéséről, így időben tud reagálni. A felhasználó beállíthatja, milyen gyakorisággal szeretne értesítést kapni, és milyen árküszöb esetén legyen figyelmeztetve. Az értesítések tartalmazzák az aktuális árat, az előző árhoz viszonyított változást, valamint a foglalásra mutató hivatkozást.
* **Árdiagram és előzmények** A rendszer grafikus formában jeleníti meg az árak alakulását, amely lehetővé teszi az ártrendek vizuális követését. A felhasználó visszamenőleg is megtekintheti, hogyan változtak az árak az adott szolgáltatás esetében, ami segíti a döntéshozatalt. A diagramok interaktívak, lehetőség van időintervallum kiválasztására, valamint az egyes szolgáltatások összehasonlítására. Az előzmények exportálhatók, és a felhasználó saját profiljához menthetők, így később is visszakereshetők.

### Felhasználói felület és dizájn

A dinamikus árfigyeléshez tartozó felhasználói felület kialakításánál elsődleges szempont volt a letisztultság, az áttekinthetőség és a könnyű kezelhetőség. A cél az volt, hogy a felhasználók gyorsan és intuitív módon hozzáférjenek az árakkal kapcsolatos információkhoz, és könnyedén értelmezhessék azok változásait. A felület vizuális megjelenése és funkcionális elrendezése egyaránt támogatja a hatékony döntéshozatalt, miközben esztétikailag is illeszkedik az utazástervező weboldal egységes arculatához.

A főbb jellemzők az alábbiak szerint részletezhetők:

* **Színvilág** A rendszer ármozgásokat sötétkék és világoskék színkódokkal jelöli. A sötétkék árnyalat a stabil vagy csökkenő árakat reprezentálja, míg a világoskék az emelkedő árakat emeli ki. Ez a színpaletta nemcsak vizuálisan kellemes, hanem funkcionálisan is jól használható, mivel segíti a felhasználót az árak gyors értelmezésében. A színek kontrasztja biztosítja, hogy a fontos információk kiemelkedjenek, ugyanakkor nem zavarják meg a felhasználói élményt. A színvilág harmonizál a weboldal többi moduljával, így egységes, professzionális megjelenést eredményez.
* **Interaktív grafikonok** A felhasználó számára lehetőség nyílik különböző időintervallumok kiválasztására, amelyek alapján megtekintheti az árak alakulását. A grafikonok dinamikusan frissülnek, és lehetőséget biztosítanak az ártrendek részletes elemzésére. A diagramok interaktívak: az egérrel vagy érintéssel történő mozgás során megjelennek az adott időponthoz tartozó konkrét árértékek, valamint az előző időszakhoz viszonyított változások. A grafikonok típusai között választható vonaldiagram, oszlopdiagram és területdiagram, attól függően, hogy a felhasználó milyen vizuális megjelenítést preferál.
* **Kártyás elrendezés** Minden figyelt tétel – legyen az repülőjegy, szállás vagy program – külön kártyán jelenik meg. A kártyák tartalmazzák az aktuális árat, az előző árhoz viszonyított változást, valamint egy foglalásra irányuló gombot, amely közvetlenül elérhetővé teszi a szolgáltatást. A kártyák elrendezése rácsos vagy listás formában is megjeleníthető, a felhasználó preferenciái szerint. A kártyákon belül ikonok, színkódok és rövid szöveges összefoglalók segítik az információ gyors feldolgozását.
* **Mobilbarát kialakítás** A felület teljes mértékben reszponzív, azaz különböző képernyőméretekhez automatikusan alkalmazkodik. Mobil eszközökön is gyorsan betöltődik, és az értesítések integrálhatók a rendszer értesítési sávjába, így a felhasználó bárhol és bármikor hozzáférhet az információkhoz. A mobilos nézet optimalizált gombméreteket, egyszerűsített menüstruktúrát és érintésre érzékeny elemeket tartalmaz, amelyek megkönnyítik a navigációt. A mobilos felület támogatja az értesítések közvetlen megnyitását, valamint a foglalási folyamat elindítását egyetlen érintéssel.

### Technikai megvalósítás

A dinamikus árfigyelés modul technikai háttere korszerű, megbízható és skálázható megoldásokra épül, amelyek biztosítják a rendszer stabil működését, a gyors adatfeldolgozást és a biztonságos felhasználói élményt. A fejlesztés során figyelembe vettük a modularitást, az integrálhatóságot és a jövőbeli bővíthetőséget is.

A technikai komponensek az alábbiak szerint épülnek fel:

* **Backend** A szerveroldali működést egy Node.js alapú rendszer biztosítja, amely REST API-n keresztül kommunikál az árfigyelő szolgáltatásokkal. A backend felelős az adatok lekéréséért, feldolgozásáért, tárolásáért és továbbításáért a frontend felé. A REST API struktúrája jól dokumentált, és támogatja az aszinkron adatkezelést, amely lehetővé teszi a gyors válaszidőt és a skálázhatóságot. A szerveroldali logika tartalmazza az értesítési szabályokat, az árküszöbök kezelését, valamint az adatnaplózást.
* **Frontend** A felhasználói felület React technológiával készült, amely lehetővé teszi a dinamikus tartalomkezelést és a gyors felhasználói interakciókat. A React komponensalapú felépítése elősegíti a moduláris fejlesztést, a könnyű karbantartást és az új funkciók gyors integrálását. A frontend felelős az árak megjelenítéséért, a grafikonok rendereléséért, az értesítések fogadásáért, valamint a felhasználói beállítások kezeléséért. A felület támogatja a többnyelvű megjelenítést, valamint a sötét és világos témák közötti váltást.
* **Adatbázis** A rendszer MongoDB-t használ az adatok tárolására, amely rugalmasan kezeli az árfigyelési előzményeket, a felhasználói beállításokat és az értesítési naplókat. A NoSQL struktúra lehetővé teszi a gyors lekérdezéseket, az adatok hatékony indexelését és a skálázható adattárolást. Az adatbázisban tárolt információk titkosítva kerülnek mentésre, és a hozzáférés jogosultságokhoz kötött, így biztosítva az adatbiztonságot.
* **API integrációk**

### Használhatóság és működés

A dinamikus árfigyelés modul használata során a felhasználó egyszerűen és intuitív módon tudja beállítani, hogy mely utazási elemeket szeretné figyelemmel kísérni. Ez lehet például egy konkrét repülőjárat, egy kiválasztott szálláshely vagy egy adott program, esemény. A rendszer lehetőséget biztosít arra, hogy a felhasználó egyéni preferenciái alapján válassza ki azokat a tételeket, amelyek számára relevánsak, és amelyek árának alakulása befolyásolhatja az utazási döntéseit.

Miután a felhasználó elvégezte a kívánt beállításokat, a rendszer automatikusan elindítja az árfigyelési folyamatot. Ez azt jelenti, hogy a háttérben folyamatosan monitorozza az adott szolgáltatás árát, és összeveti azt az előzőleg rögzített értékekkel. Amennyiben az ár eléri vagy meghaladja a felhasználó által beállított küszöbértéket – például egy meghatározott maximum vagy minimum ár –, a rendszer értesítést küld. Ez az értesítés lehet e-mail formájában, push üzenetként mobil eszközön, vagy akár webes felugró ablakon keresztül.

A felhasználó bármikor módosíthatja a figyelési paramétereket, például megváltoztathatja az árküszöböt, új elemeket adhat hozzá a figyelési listához, vagy eltávolíthatja a már nem releváns tételeket. Emellett lehetősége van arra is, hogy ideiglenesen szüneteltesse az értesítéseket, vagy teljesen kikapcsolja azokat, ha már nem kívánja követni az adott ármozgásokat. A rendszer minden módosítást azonnal érvényesít, és ennek megfelelően frissíti a figyelési logikát.

A modul működése során kiemelt figyelmet fordít a felhasználói élményre: a beállítási folyamat egyszerű, lépésről lépésre vezetett, és nem igényel technikai előképzettséget. A felület gyorsan betöltődik, az interakciók gördülékenyek, és a rendszer visszajelzései (pl. sikeres mentés, figyelmeztetés, hibaüzenet) egyértelműek és informatívak. A cél az, hogy a felhasználó teljes kontrollt gyakorolhasson az árfigyelés felett, miközben a rendszer a háttérben megbízhatóan végzi a feladatát.

### Tesztelés és validálás

A dinamikus árfigyelés funkció megbízható működésének biztosítása érdekében elengedhetetlen a rendszer alapos tesztelése és validálása. A tesztelési folyamat célja, hogy feltárja az esetleges hibákat, ellenőrizze a rendszer stabilitását, valamint igazolja, hogy az implementált megoldások megfelelnek a specifikációban rögzített elvárásoknak. A validálás során azt vizsgáljuk, hogy a rendszer valóban azt nyújtja-e, amit a felhasználók elvárnak tőle, és hogy a felhasználói élmény megfelel-e a tervezett szintnek.

A funkció tesztelése során az alábbi szempontokat vizsgáljuk:

* **Árfrissítések pontossága** A rendszer által lekért áraknak pontosan meg kell egyezniük a szolgáltatók aktuális kínálatával. Ennek érdekében összehasonlító teszteket végzünk, ahol a lekért adatok valós idejű árakkal kerülnek összevetésre. Vizsgáljuk, hogy az árak frissítése megfelelő időközönként történik-e, és hogy a rendszer képes-e kezelni a hirtelen árváltozásokat, például akciók vagy szezonális kiugrások esetén.
* **Értesítések működése** Az értesítési rendszer működésének tesztelése során ellenőrizzük, hogy az értesítések időben, megfelelő formában és tartalommal érkeznek-e. Vizsgáljuk, hogy a felhasználó által beállított küszöbértékek alapján történik-e az értesítés generálása, valamint hogy az értesítések kikapcsolása, szüneteltetése és újraaktiválása hibamentesen működik-e. Tesztfiókokkal szimuláljuk a különböző értesítési beállításokat, és figyeljük, hogy a rendszer megfelelően reagál-e az ármozgásokra.
* **Felhasználói interakciók** A felület intuitív használhatóságát manuális teszteléssel ellenőrizzük. Vizsgáljuk, hogy a felhasználói műveletek – például beállítások módosítása, új figyelési elem hozzáadása, értesítések kezelése – gyorsan és hibamentesen végrehajthatók-e. A tesztelés során különböző felhasználói profilokat hozunk létre, és szimuláljuk a tipikus használati eseteket, hogy feltárjuk az esetleges UX problémákat.
* **Mobiltesztelés** A rendszer mobil eszközökön történő működését külön teszteljük Android és iOS platformokon. Vizsgáljuk a felület reszponzivitását, az értesítések megjelenését és kezelhetőségét, valamint az interakciók (pl. érintés, görgetés, zoom) megfelelő működését. A cél az, hogy a funkció mobilon is teljes funkcionalitással, gyorsan és hibamentesen működjön.

### Fejlesztési környezet

A dinamikus árfigyelés modul fejlesztése során korszerű, iparági szinten elismert eszközöket és technológiákat alkalmaztunk, amelyek biztosítják a hatékony munkavégzést, a kód minőségét, valamint a projekt átláthatóságát és fenntarthatóságát.

* **Fejlesztői eszközök** A fejlesztéshez használt főbb eszközök közé tartozik a Visual Studio Code, amely modern, testreszabható fejlesztői környezetet biztosít.
* **Verziókezelés** A projekt Git alapú verziókövetést használ, amely lehetővé teszi a fejlesztési folyamat strukturált nyomon követését. Külön ágakat hoztunk létre a frontend és backend fejlesztéshez, valamint a teszteléshez, így biztosítva a párhuzamos munkavégzést és a hibamentes integrációt. A commit üzenetek szabványos formátumban készülnek, és minden változtatás dokumentálva van.
* **Tesztelési keretrendszer**

## 1.2 Időjárás-előrejelzés integráció

### Célkitűzés

Az időjárás-előrejelzés integráció célja, hogy az utazástervező rendszer felhasználói az utazás előkészítése és lebonyolítása során megbízható, valós idejű és előrejelzett meteorológiai információkhoz jussanak az általuk kiválasztott úti célokra vonatkozóan. Az időjárási viszonyok jelentős hatással lehetnek az utazás élményére, a programok megvalósíthatóságára, valamint a csomagolás és öltözködés tervezésére. Éppen ezért elengedhetetlen, hogy a felhasználók már az utazás megtervezésekor hozzáférjenek az aktuális és várható időjárási adatokhoz.

A funkció lehetővé teszi, hogy az utazók az időjárási körülmények figyelembevételével alakítsák ki napi programjaikat, kiválasszák a megfelelő ruházatot és felszerelést, valamint eldöntsék, hogy az adott időszak alkalmas-e az utazásra. Például egy tengerparti nyaralás esetén fontos tudni, hogy várható-e eső vagy vihar, míg egy hegyi túra esetén a hőmérséklet és a szélsebesség lehet döntő tényező. A rendszer által biztosított információk segítik a felhasználókat abban, hogy elkerüljék a kellemetlen meglepetéseket, és az időjárásnak megfelelően alakítsák ki utazási terveiket.

Az integráció célja továbbá az is, hogy a felhasználók ne külön alkalmazásokból vagy weboldalakról kelljen, hogy beszerezzék az időjárási adatokat, hanem közvetlenül az utazástervező felületen, egy helyen, összefüggésében láthassák azokat. Ez nemcsak kényelmesebbé teszi a tervezést, hanem növeli a rendszer használhatóságát és komplexitását is.

### Funkcionális leírás

Az időjárás-előrejelzés modul több egymással összefüggő funkciót tartalmaz, amelyek együttesen biztosítják a felhasználók számára a teljes körű meteorológiai tájékoztatást. A funkciók célja, hogy az időjárási információk ne csupán statikus adatokként jelenjenek meg, hanem aktívan támogassák az utazási döntéshozatalt.

* **Aktuális időjárási adatok megjelenítése** A rendszer valós időben jeleníti meg az adott úti cél aktuális időjárási viszonyait. Az adatok között szerepel a hőmérséklet (°C), a páratartalom (%), a szélsebesség (km/h), az UV-index (skálán), valamint a látási viszonyok (pl. köd, tiszta égbolt). Ezek az információk segítenek a felhasználónak abban, hogy az adott pillanatban milyen körülményekre számíthat, és ennek megfelelően alakítsa ki napi programját vagy öltözetét.
* **Rövid távú előrejelzés (1–3 nap)** A rendszer óránkénti bontásban biztosítja az időjárási előrejelzést a következő 72 órára vonatkozóan. Ez különösen hasznos lehet azok számára, akik rövid utazást terveznek, vagy akik napi szinten szeretnék optimalizálni programjaikat. Az előrejelzés tartalmazza a várható hőmérsékletet, csapadék valószínűségét, szélirányt és sebességet, valamint az UV-indexet. Az adatok vizuálisan is megjelennek, például grafikonokon vagy idővonalakon, amelyek segítik az értelmezést.
* **Hosszú távú előrejelzés (4–10 nap)** A rendszer napi bontásban biztosítja az időjárási előrejelzést a következő 4–10 napra. Az adatok között szerepelnek a napi átlaghőmérséklet, a várható csapadék mennyisége és formája (pl. eső, hó), valamint a hőmérsékleti ingadozások. Ez a funkció különösen hasznos azok számára, akik hosszabb utazást terveznek, és szeretnék előre látni, hogy milyen időjárási viszonyokra számíthatnak az egyes napokon. A rendszer figyelembe veszi a regionális időjárási sajátosságokat is, így pontosabb előrejelzést nyújt.
* **Időjárási figyelmeztetések** A rendszer automatikusan értesítést küld a felhasználónak, ha az adott úti célon extrém időjárási jelenség várható. Ilyen lehet például vihar, heves esőzés, extrém hőség, havazás, jégeső vagy erős szél. Az értesítések célja, hogy a felhasználó időben tudomást szerezzen a potenciálisan veszélyes körülményekről, és ennek megfelelően módosíthassa terveit. Az értesítések formája lehet e-mail, push üzenet vagy webes figyelmeztetés, és tartalmazzák az esemény típusát, várható időpontját és javasolt teendőket.
* **Időjárás alapú ajánlások** A rendszer nemcsak adatokat szolgáltat, hanem intelligens ajánlásokat is tesz a felhasználó számára. Például ha eső várható, a rendszer javasolhat beltéri programokat, múzeumlátogatást vagy wellness szolgáltatásokat. Ha napsütéses idő várható, akkor szabadtéri tevékenységeket, túrákat vagy strandolást ajánlhat. Az ajánlások figyelembe veszik a felhasználó korábbi preferenciáit, az úti cél adottságait, valamint a programok elérhetőségét. Ezáltal az időjárás nem akadály, hanem tervezési szempontként jelenik meg az utazás során.

### Felhasználói felület és dizájn

Az időjárás-előrejelzés modul felhasználói felülete úgy lett kialakítva, hogy vizuálisan letisztult, informatív és könnyen értelmezhető legyen minden korosztály és technikai szintű felhasználó számára. A cél az volt, hogy az időjárási adatok ne csupán száraz információként jelenjenek meg, hanem intuitív, vizuálisan jól strukturált formában, amely segíti a gyors döntéshozatalt és a programtervezést.

A felület kialakításánál figyelembe vettük a modern webes dizájn irányelveket, a felhasználói szokásokat, valamint a mobilos és asztali eszközök közötti különbségeket. A modul minden eleme reszponzív, azaz automatikusan alkalmazkodik a képernyőmérethez, így biztosítva a zökkenőmentes használatot bármilyen eszközön.

A főbb vizuális és funkcionális jellemzők az alábbiak szerint részletezhetők:

* **Színvilág** A felület színpalettája sötétkék és világoskék árnyalatokat használ, amelyek nemcsak esztétikailag harmonikusak, hanem funkcionálisan is jól elkülönítik az időjárási típusokat. A sötétkék háttér nyugalmat, stabilitást és megbízhatóságot sugall, míg a világoskék elemek kiemelik az aktuális, releváns adatokat. A színkódolás segíti a felhasználót abban, hogy első pillantásra felismerje az időjárási helyzetet, például a viharos időt vagy a napsütéses periódusokat. A színek kontrasztja biztosítja a jó olvashatóságot, még gyengébb fényviszonyok mellett is.
* **Ikonrendszer** A modul minden időjárási jelenséghez egyértelmű, grafikus ikont rendel, amelyek vizuálisan is segítik az információ gyors értelmezését. Például napsütés esetén egy stilizált nap ikon jelenik meg, felhős időnél felhő, eső esetén csepp vagy esőfelhő, havazásnál hópehely. Az ikonok egységes stílusban készültek, és illeszkednek a weboldal általános arculatához. A grafikus elemek mérete és elhelyezése optimalizált, így nem zavarják a szöveges tartalmat, hanem kiegészítik azt.
* **Interaktív térkép** A felhasználó az úti célra kattintva megtekintheti a helyi időjárást egy interaktív térképen, amely lehetőséget biztosít a nagyításra, mozgatásra és részletesebb adatok megjelenítésére. A térkép rétegei között váltható a hőmérséklet, csapadék, szélirány, UV-index és egyéb meteorológiai paraméterek. A térkép integrálva van a többi modulba, így például az útvonaltervezővel is összekapcsolható, és megjeleníthető, hogy az adott útvonal mentén milyen időjárási viszonyokra lehet számítani.
* **Mobilbarát megjelenés** A modul teljes mértékben reszponzív kialakítású, azaz mobil eszközökön is hibamentesen működik. A grafikonok, ikonok és szöveges elemek automatikusan igazodnak a képernyőmérethez, így biztosítva a jó olvashatóságot és kezelhetőséget. A mobilos nézetben a menük egyszerűsítettek, a gombok nagyobb méretűek, és az érintésre optimalizált interakciók révén a felhasználó könnyedén navigálhat az adatok között. A rendszer támogatja a mobilos értesítéseket is, így a felhasználó azonnal értesülhet az időjárási figyelmeztetésekről, akár utazás közben is.

### Technikai megvalósítás

Az időjárás-előrejelzés modul technikai háttere korszerű, megbízható és skálázható megoldásokra épül, amelyek biztosítják a rendszer stabil működését, a gyors adatfeldolgozást és a biztonságos felhasználói élményt. A fejlesztés során kiemelt figyelmet fordítottunk az adatbiztonságra, a modularitásra és az integrálhatóságra, hogy a rendszer hosszú távon is fenntartható és bővíthető legyen.

A technikai komponensek az alábbiak szerint épülnek fel:

* **Backend** A szerveroldali működést egy Node.js alapú rendszer biztosítja, amely REST API-n keresztül kommunikál az időjárási szolgáltatókkal. A backend felelős az adatok lekéréséért, feldolgozásáért, tárolásáért és továbbításáért a frontend felé. A REST API struktúrája jól dokumentált, és támogatja az aszinkron adatkezelést, amely lehetővé teszi a gyors válaszidőt és a skálázhatóságot. A szerveroldali logika tartalmazza az értesítési szabályokat, az időjárási figyelmeztetések kezelését, valamint az adatnaplózást.
* **Frontend** A felhasználói felület React technológiával készült, amely lehetővé teszi a dinamikus tartalomkezelést és a gyors felhasználói interakciókat. A React komponensalapú felépítése elősegíti a moduláris fejlesztést, a könnyű karbantartást és az új funkciók gyors integrálását. A frontend felelős az időjárási adatok megjelenítéséért, a grafikonok rendereléséért, az értesítések fogadásáért, valamint a felhasználói beállítások kezeléséért. A felület támogatja a többnyelvű megjelenítést, valamint a sötét és világos témák közötti váltást.
* **Adatbázis** A rendszer MongoDB-t használ az adatok tárolására, amely rugalmasan kezeli a lekérdezett időjárási előzményeket, a felhasználói beállításokat és az értesítési naplókat. A NoSQL struktúra lehetővé teszi a gyors lekérdezéseket, az adatok hatékony indexelését és a skálázható adattárolást. Az adatbázisban tárolt információk titkosítva kerülnek mentésre, és a hozzáférés jogosultságokhoz kötött, így biztosítva az adatbiztonságot.
* **API integráció** A rendszer több külső szolgáltatással van összekötve, például az OpenWeatherMap, WeatherAPI vagy Meteomatics platformokkal. Ezekből származnak az időjárási adatok, amelyeket a rendszer feldolgoz és megjelenít. Az API-k integrációja során figyelembe vettük a szolgáltatók által biztosított adatfrissítési gyakoriságot, a válaszidőt, valamint az adatstruktúrák kompatibilitását. A rendszer képes kezelni az API-k esetleges hibáit, és alternatív adatforrásokat használ, ha szükséges.
* **Cache-elés** A rendszer ideiglenesen tárolja az időjárási adatokat, hogy csökkentse az API-hívások számát, gyorsítsa a betöltést, és biztosítsa a felhasználói élmény folyamatosságát. A cache-elés időalapú szabályok szerint működik, és automatikusan frissül, ha az adatok elavulnak. Ez különösen fontos mobilos használat esetén, ahol az adatkapcsolat korlátozott lehet.

### Használhatóság és működés

Az időjárás-előrejelzés integráció modul használata során a felhasználó egyszerűen és intuitív módon választhatja ki az utazás tervezésekor releváns úti célt. A rendszer ezt követően automatikusan megjeleníti az adott helyszínhez tartozó aktuális meteorológiai adatokat, valamint a rövid és hosszú távú időjárási előrejelzéseket. A felhasználó számára ez a funkció különösen hasznos, mivel az időjárási viszonyok jelentős hatással lehetnek az utazás élményére, a programok megvalósíthatóságára, valamint a csomagolás és öltözködés tervezésére.

A rendszer lehetőséget biztosít arra, hogy a felhasználó egyéni preferenciái alapján beállítsa, milyen típusú időjárási eseményekről szeretne értesítést kapni. Például választhatja, hogy csak az extrém hőmérsékleti viszonyokról, viharokról, esőzésekről vagy havazásról kapjon figyelmeztetést. Ezeket a beállításokat a felhasználói profilban lehet elvégezni, és a rendszer ezeket automatikusan alkalmazza minden új úti cél kiválasztásakor.

Az értesítések célja, hogy a felhasználó időben tudomást szerezzen az esetleges időjárási változásokról, és ennek megfelelően módosíthassa utazási terveit. A rendszer az értesítéseket több csatornán keresztül képes továbbítani: e-mail formájában, push üzenetként mobil eszközökre, valamint webes felugró ablakokként a felületen belül. Az értesítések tartalmazzák az esemény típusát, várható időpontját, valamint javasolt teendőket (pl. program módosítása, csomagolási tanácsok).

A modul működése során kiemelt figyelmet fordít a felhasználói élményre. A felület gyorsan betöltődik, az interakciók gördülékenyek, és a rendszer visszajelzései (pl. sikeres mentés, figyelmeztetés, hibaüzenet) egyértelműek és informatívak. A cél az, hogy a felhasználó teljes kontrollt gyakorolhasson az időjárási információk felett, miközben a rendszer a háttérben megbízhatóan végzi a feladatát.

### Tesztelés és validálás

Az időjárás-előrejelzés modul megbízható működésének biztosítása érdekében elengedhetetlen a funkciók alapos tesztelése és validálása. A tesztelési folyamat célja, hogy feltárja az esetleges hibákat, ellenőrizze a rendszer stabilitását, valamint igazolja, hogy az implementált megoldások megfelelnek a specifikációban rögzített elvárásoknak. A validálás során azt vizsgáljuk, hogy a rendszer valóban azt nyújtja-e, amit a felhasználók elvárnak tőle, és hogy a felhasználói élmény megfelel-e a tervezett szintnek.

A funkció tesztelése során az alábbi szempontokat vizsgáljuk:

* **Adatok pontossága** A rendszer által lekért időjárási adatoknak pontosan meg kell egyezniük a valós meteorológiai viszonyokkal. Ennek érdekében összehasonlító teszteket végzünk, ahol a lekért információkat hivatalos meteorológiai szolgálatok adataival vetjük össze. Vizsgáljuk, hogy az adatok frissítése megfelelő időközönként történik-e, és hogy a rendszer képes-e kezelni a hirtelen időjárási változásokat.
* **Frissítési gyakoriság** A rendszernek biztosítania kell, hogy az időjárási adatok rendszeresen és megbízhatóan frissüljenek. A tesztelés során ellenőrizzük, hogy az API-hívások megfelelő időközönként történnek, és hogy az adatok naprakészen jelennek meg a felhasználói felületen. A frissítési ciklusok optimalizálása kulcsfontosságú a rendszer hatékonysága szempontjából.
* **Felhasználói interakciók** A felület intuitív használhatóságát manuális teszteléssel ellenőrizzük. Vizsgáljuk, hogy a felhasználói műveletek – például úti cél kiválasztása, értesítési beállítások módosítása, térkép használata – gyorsan és hibamentesen végrehajthatók-e. A tesztelés során különböző felhasználói profilokat hozunk létre, és szimuláljuk a tipikus használati eseteket, hogy feltárjuk az esetleges UX problémákat.
* **Mobiltesztelés** A rendszer mobil eszközökön történő működését külön teszteljük Android és iOS platformokon. Vizsgáljuk a felület reszponzivitását, az értesítések megjelenését és kezelhetőségét, valamint az interakciók (pl. érintés, görgetés, zoom) megfelelő működését. A cél az, hogy a funkció mobilon is teljes funkcionalitással, gyorsan és hibamentesen működjön.
* **Értesítési rendszer** Az értesítési rendszer működésének tesztelése során ellenőrizzük, hogy az értesítések időben, megfelelő formában és tartalommal érkeznek-e. Vizsgáljuk, hogy a felhasználó által beállított eseménytípusok alapján történik-e az értesítés generálása, valamint hogy az értesítések kikapcsolása, szüneteltetése és újraaktiválása hibamentesen működik-e. Tesztfiókokkal szimuláljuk a különböző értesítési beállításokat, és figyeljük, hogy a rendszer megfelelően reagál-e az időjárási eseményekre.

### Fejlesztési környezet

Az időjárás-előrejelzés modul fejlesztése során korszerű, iparági szinten elismert eszközöket és technológiákat alkalmaztunk, amelyek biztosítják a hatékony munkavégzést, a kód minőségét, valamint a projekt átláthatóságát és fenntarthatóságát.

* **Fejlesztői eszközök** A fejlesztéshez használt főbb eszközök közé tartozik a Visual Studio Code, amely modern, testreszabható fejlesztői környezetet biztosít.
* **Verziókezelés** A projekt Git alapú verziókövetést használ, amely lehetővé teszi a fejlesztési folyamat strukturált nyomon követését. Külön ágakat hoztunk létre a frontend és backend fejlesztéshez, valamint a teszteléshez, így biztosítva a párhuzamos munkavégzést és a hibamentes integrációt. A commit üzenetek szabványos formátumban készülnek, és minden változtatás dokumentálva van.

## 1.3 Költségmegosztás funkció

### Célkitűzés

A költségmegosztás funkció célja, hogy csoportos utazások során biztosítsa a kiadások átlátható, igazságos és egyszerű elosztását a résztvevők között. Az utazás során gyakran előfordul, hogy a résztvevők közösen fizetnek különböző szolgáltatásokért – például szállásért, étkezésért, belépőkért vagy közlekedésért –, és ezeknek a költségeknek az utólagos elszámolása sokszor nehézkes, időigényes és konfliktusforrás lehet.

A rendszer célja, hogy ezt a folyamatot digitalizálja, automatizálja és átláthatóvá tegye. A modul lehetőséget nyújt a közös költségek tételes rögzítésére, azok automatikus szétosztására a résztvevők között, valamint a pénzügyi egyenleg folyamatos nyomon követésére. A felhasználók pontosan láthatják, hogy mennyit fizettek be, mennyi a rájuk eső rész, és mennyi az esetleges visszajáró összeg vagy tartozás.

Ezáltal elkerülhetők a félreértések, a pontatlan elszámolások, és minden utazó számára egyértelművé válik, hogy milyen mértékben járult hozzá az utazás költségeihez. A funkció nemcsak pénzügyi szempontból hasznos, hanem hozzájárul a csoportos utazások gördülékenyebb lebonyolításához, a bizalom fenntartásához és a közös élmények zavartalan megéléséhez.

A modul különösen hasznos lehet baráti társaságok, családok, céges csapatépítők vagy akár szervezett túrák esetén, ahol több résztvevő közösen vesz részt a költségek viselésében. A digitális elszámolás nemcsak gyorsabb, hanem pontosabb is, mint a hagyományos, papíralapú vagy szóbeli megállapodások.

### Funkcionális leírás

A költségmegosztás modul több egymással összefüggő funkciót tartalmaz, amelyek együttesen biztosítják a pénzügyi elszámolás teljes körű kezelését. A rendszer célja, hogy minden résztvevő számára egyértelmű, testreszabható és könnyen kezelhető megoldást nyújtson a közös kiadások kezelésére.

* **Költségek rögzítése** A felhasználók tételesen rögzíthetik az utazás során felmerülő kiadásokat, például szállásfoglalás, étkezés, belépők, közlekedés, programok vagy egyéb szolgáltatások. Minden költséghez megadható a fizető személy, az összeg, a dátum, a költség típusa, valamint a résztvevők, akikre az adott tétel vonatkozik. A rendszer lehetőséget biztosít a költségek kategorizálására, így könnyen áttekinthetővé válik, hogy mely területeken merültek fel kiadások.
* **Automatikus megosztás** A rendszer képes automatikusan szétosztani a rögzített költségeket a résztvevők között. Alapértelmezés szerint egyenlő arányban történik a megosztás, de lehetőség van egyéni beállításokra is, például ha valaki nem vett részt egy adott programon, vagy ha eltérő mértékben járult hozzá egy szolgáltatás igénybevételéhez. A megosztás logikája rugalmasan testreszabható, és a rendszer azonnal frissíti az egyenlegeket a módosítások alapján.
* **Egyéni hozzájárulások kezelése** A modul támogatja az egyéni hozzájárulások kezelését is, ami különösen fontos fakultatív programok esetén. Például ha egy városnéző túrán csak néhány utazó vesz részt, akkor a költség csak közöttük oszlik meg. A rendszer lehetővé teszi, hogy minden költségtételhez külön meg lehessen adni, kik az érintett utazók, így az elszámolás pontos és igazságos marad.
* **Egyenlegkövetés** Minden felhasználó számára elérhető egy egyenlegnézet, amely megmutatja, hogy mennyit fizetett be, mennyi a rá eső költség, és mennyi az aktuális egyenlege. Az egyenleg lehet pozitív (túlfizetés), negatív (tartozás) vagy nulla (kiegyenlített állapot). A rendszer folyamatosan frissíti az egyenlegeket, így a felhasználók mindig naprakész információval rendelkeznek.
* **Kifizetések rögzítése** A rendszer nyilvántartja, ha valaki kiegyenlítette a tartozását, például készpénzes fizetéssel vagy banki átutalással. A kifizetések rögzítése után az egyenlegek automatikusan frissülnek, és a rendszer jelzi, ha minden utazó elszámolt a költségekkel. Ez különösen hasznos az utazás végén, amikor a végső elszámolás történik.
* **Exportálási lehetőség** A költséglista és az egyenlegek letölthetők PDF formátumban, amely megkönnyíti az elszámolást, különösen akkor, ha hivatalos dokumentációra van szükség (pl. céges utazás esetén). Az exportált dokumentum tartalmazza a költségek részleteit, a megosztási arányokat, az egyenlegeket és a kifizetési státuszokat. A PDF formátum könnyen megosztható, nyomtatható és archiválható.

### Felhasználói felület és dizájn

A költségmegosztás modul felhasználói felülete intuitív, logikusan felépített és könnyen kezelhető, különösen csoportos használatra optimalizálva. A cél az volt, hogy a felhasználók gyorsan és egyszerűen tudják rögzíteni a közös kiadásokat, nyomon követni az egyenlegeket, valamint áttekinteni a teljes pénzügyi helyzetet az utazás során. A felület kialakítása során figyelembe vettük a modern webes dizájn elveit, a felhasználói szokásokat, valamint a mobilos és asztali eszközök közötti különbségeket.

A modul vizuális megjelenése és funkcionális elrendezése egyaránt támogatja a hatékony adatkezelést, miközben esztétikailag is illeszkedik az utazástervező rendszer egységes arculatához. A felület minden eleme reszponzív, azaz automatikusan alkalmazkodik a képernyőmérethez, így biztosítva a zökkenőmentes használatot bármilyen eszközön.

A főbb vizuális és funkcionális jellemzők az alábbiak szerint részletezhetők:

* **Színvilág** A modul sötétkék háttérrel és világoskék kiemelésekkel jelenik meg, amely segíti a vizuális tagolást és a pénzügyi adatok gyors értelmezését. A sötétkék szín nyugalmat, stabilitást és megbízhatóságot sugall, míg a világoskék elemek kiemelik az aktuális, releváns információkat. A színpaletta kontrasztos, jól olvasható, és segíti a felhasználót abban, hogy első pillantásra felismerje a fontos adatokat, például a tartozásokat vagy a túlfizetéseket.
* **Tétellista nézet** Minden költség külön sorban jelenik meg, megjelölve a fizető személyt, az összeget, a költség típusát, valamint az érintett utazókat. A lista rendezhető dátum, összeg vagy kategória szerint, és lehetőség van szűrésre is, például csak egy adott utazóhoz tartozó tételek megjelenítésére. A tétellista interaktív: a felhasználó módosíthatja, törölheti vagy új költséget adhat hozzá egyetlen kattintással.
* **Grafikus egyenlegdiagram** A rendszer kördiagram vagy oszlopdiagram formájában jeleníti meg az egyéni hozzájárulásokat és tartozásokat. A diagramok színkódoltak, így könnyen értelmezhetők: például zöld szín jelzi a túlfizetést, piros a tartozást, kék az egyenleg kiegyenlítettségét. A diagramok interaktívak, az egérrel vagy érintéssel történő mozgás során megjelennek az adott utazóhoz tartozó pontos értékek. A grafikus megjelenítés segíti a csoportos elszámolás átláthatóságát, és vizuálisan is megerősíti az egyenlegek állapotát.
* **Csoportosítási lehetőség** A költségek kategóriák szerint csoportosíthatók, például étkezés, szállás, közlekedés, programok, egyéb. Ez a funkció lehetővé teszi, hogy a felhasználók könnyen áttekintsék, mely területeken merültek fel a legnagyobb kiadások, és hol lehet esetleg optimalizálni. A kategóriák színkódoltak, és a rendszer automatikusan javaslatot tesz a kategóriákra a költségtípus alapján, de a felhasználó manuálisan is módosíthatja azokat.
* **Mobilbarát kialakítás** A felület teljes mértékben reszponzív, azaz mobil eszközökön is gyorsan betölthető, és lehetővé teszi az adatok gyors rögzítését akár utazás közben is. A mobilos nézet optimalizált gombméreteket, egyszerűsített menüstruktúrát és érintésre érzékeny elemeket tartalmaz, amelyek megkönnyítik a navigációt. A rendszer támogatja a mobilos értesítéseket is, így a felhasználó azonnal értesülhet, ha új költség került rögzítésre, vagy ha változás történt az egyenlegében.

### Technikai megvalósítás

A költségmegosztás modul technikai háttere korszerű, megbízható és skálázható megoldásokra épül, amelyek biztosítják a rendszer stabil működését, a gyors adatfeldolgozást és a biztonságos felhasználói élményt. A fejlesztés során kiemelt figyelmet fordítottunk az adatbiztonságra, a modularitásra és az integrálhatóságra, hogy a rendszer hosszú távon is fenntartható és bővíthető legyen.

A technikai komponensek az alábbiak szerint épülnek fel:

* **Backend** A szerveroldali működést egy Node.js alapú rendszer biztosítja, amely REST API-n keresztül kezeli a költségadatokat, felhasználói csoportokat és egyenlegeket. A backend felelős az adatok lekéréséért, feldolgozásáért, tárolásáért és továbbításáért a frontend felé. A REST API struktúrája jól dokumentált, és támogatja az aszinkron adatkezelést, amely lehetővé teszi a gyors válaszidőt és a skálázhatóságot.
* **Frontend** A felhasználói felület React technológiával készült, amely lehetővé teszi a dinamikus tartalomkezelést és a gyors felhasználói interakciókat. A React komponensalapú felépítése elősegíti a moduláris fejlesztést, a könnyű karbantartást és az új funkciók gyors integrálását. A frontend felelős a költséglista megjelenítéséért, az egyenlegdiagramok rendereléséért, az interakciók kezeléséért, valamint a felhasználói beállítások mentéséért.
* **Adatbázis** A rendszer MongoDB-t használ az adatok tárolására, amely rugalmasan kezeli a költségek részleteit, a felhasználók közötti megosztási arányokat és a kifizetési státuszokat. A NoSQL struktúra lehetővé teszi a gyors lekérdezéseket, az adatok hatékony indexelését és a skálázható adattárolást. Az adatbázisban tárolt információk titkosítva kerülnek mentésre, és a hozzáférés jogosultságokhoz kötött, így biztosítva az adatbiztonságot.
* **Hitelesítés** A felhasználók biztonságos bejelentkezéssel (OAuth2) csatlakozhatnak a csoporthoz, és csak az általuk létrehozott vagy meghívott utazásokhoz férnek hozzá. A hitelesítési rendszer támogatja a többfaktoros azonosítást, a jelszókezelést, valamint az adminisztrátori jogosultságok kezelését is. Ez biztosítja, hogy az adatokhoz csak az arra jogosult személyek férjenek hozzá, és minden művelet naplózásra kerüljön.

### Használhatóság és működés

A költségmegosztás modul használata során a felhasználók számára biztosított funkcionalitás lehetővé teszi, hogy az utazás bármely szakaszában – legyen az előkészület, aktív utazás vagy utólagos elszámolás – egyszerűen és gyorsan rögzítsenek új költségeket. A rendszer úgy lett kialakítva, hogy a felhasználók egyetlen felületen kezelhessék az összes pénzügyi adatot, miközben a háttérben automatikusan történik az egyenlegek kiszámítása és frissítése.

A költségek rögzítése során a felhasználó megadhatja a tétel pontos megnevezését, összegét, a fizető személyt, valamint azt, hogy kiket érint az adott kiadás. Ez lehetővé teszi, hogy a rendszer pontosan meghatározza, milyen arányban kell megosztani a költséget a résztvevők között. A felhasználó választhat egyenlő megosztást, vagy egyéni arányokat is beállíthat, például ha valaki nem vett részt egy adott programon, vagy ha eltérő mértékben járult hozzá a kiadáshoz.

A modul további lehetőséget biztosít arra is, hogy a csoport tagjai egymásnak kifizetéseket rögzítsenek. Ez különösen hasznos az utazás végén, amikor a résztvevők elszámolnak egymással, például készpénzes visszafizetések vagy banki átutalások formájában. A rendszer minden kifizetést naplóz, és automatikusan frissíti az egyenlegeket, így a pénzügyi állapot mindig naprakészen jelenik meg.

A felhasználói élmény szempontjából kiemelten fontos, hogy a modul gyorsan reagáljon, könnyen kezelhető legyen, és ne igényeljen technikai előképzettséget. A felület logikusan felépített, a műveletek egyértelműek, és a rendszer visszajelzései (pl. sikeres mentés, hibajelzés, egyenlegváltozás) azonnal megjelennek. A mobilbarát kialakításnak köszönhetően a felhasználók akár utazás közben, menet közben is rögzíthetik a költségeket, így nem kell utólag rekonstruálni a kiadásokat.

### Tesztelés és validálás

A költségmegosztás funkció megbízható működésének biztosítása érdekében elengedhetetlen a rendszer alapos tesztelése és validálása. A tesztelési folyamat célja, hogy feltárja az esetleges hibákat, ellenőrizze a számítási algoritmusok pontosságát, valamint igazolja, hogy a felhasználói interakciók zökkenőmentesen és hibamentesen zajlanak.

A funkció tesztelése során az alábbi szempontokat vizsgáljuk:

* **Költségelosztás pontossága** A rendszernek pontosan kell kiszámítania az egyenlegeket, figyelembe véve az egyéni megosztási arányokat, a részleges részvételt, valamint a kifizetések időzítését. A tesztelés során különböző szimulált utazási forgatókönyveket alkalmazunk, amelyekben eltérő számú résztvevő, különböző típusú költségek és egyéni hozzájárulások szerepelnek. A cél az, hogy minden esetben helyes egyenlegszámítás történjen, és ne forduljon elő számítási hiba.
* **Felhasználói interakciók** A költségek rögzítése, módosítása és törlése során a rendszernek gyorsan és hibamentesen kell működnie. A tesztelés során ellenőrizzük, hogy a felhasználói műveletek logikusan követik egymást, a felület megfelelő visszajelzést ad, és a mentések azonnal érvényesülnek. A tesztelés kiterjed a különböző eszközökön történő használatra, valamint a különböző böngészők kompatibilitására is.
* **Adatbiztonság** A pénzügyi adatok védelme kiemelten fontos, ezért a rendszer titkosítással és jogosultságkezeléssel biztosítja, hogy az adatokhoz csak az arra jogosult felhasználók férjenek hozzá. A tesztelés során ellenőrizzük az adatátvitel biztonságát (HTTPS), a felhasználói hitelesítés működését (OAuth2), valamint a hozzáférési szintek helyes kezelését. A rendszer naplózza a műveleteket, így visszakövethető minden változtatás.
* **Mobiltesztelés** A funkció mobil eszközökön is teljes funkcionalitással kell, hogy működjön. A tesztelés során Android és iOS platformokon ellenőrizzük a felület reszponzivitását, az interakciók gördülékenységét, valamint az értesítések megjelenését. A cél az, hogy a felhasználók mobilon is ugyanolyan hatékonyan tudják kezelni a költségeket, mint asztali gépen.
* **Exportálás tesztelése** A PDF exportálás során a rendszernek pontosan kell tükröznie a rendszerben tárolt adatokat. A tesztelés során ellenőrizzük, hogy a letöltött dokumentum tartalmazza a költségek részleteit, a megosztási arányokat, az egyenlegeket és a kifizetési státuszokat. A PDF formátumnak jól strukturáltnak, olvashatónak és nyomtathatónak kell lennie.

### Fejlesztési környezet

A költségmegosztás modul fejlesztése során korszerű, iparági szinten elismert eszközöket és technológiákat alkalmaztunk, amelyek biztosítják a hatékony munkavégzést, a kód minőségét, valamint a projekt átláthatóságát és fenntarthatóságát.

* **Fejlesztői eszközök** A fejlesztéshez használt főbb eszközök közé tartozik a Visual Studio Code, amely modern, testreszabható fejlesztői környezetet biztosít.
* **Verziókezelés** A projekt Git alapú verziókövetést használ, amely lehetővé teszi a fejlesztési folyamat strukturált nyomon követését. Külön ágakat hoztunk létre a frontend és backend fejlesztéshez, valamint a teszteléshez, így biztosítva a párhuzamos munkavégzést és a hibamentes integrációt. A commit üzenetek szabványos formátumban készülnek, és minden változtatás dokumentálva van.

## 1.4 Fenntarthatósági kalkulátor

### Célkitűzés

A fenntarthatósági kalkulátor célja, hogy az utazástervezés során a felhasználók tudatos, környezetbarát döntéseket hozhassanak, figyelembe véve az utazásuk által okozott környezeti hatásokat. A modern turizmus egyik legnagyobb kihívása a fenntarthatóság biztosítása, mivel az utazások során jelentős mennyiségű szén-dioxid-kibocsátás és egyéb környezeti terhelés keletkezik. A kalkulátor célja, hogy ezt a láthatatlan hatást láthatóvá tegye, és a felhasználók számára kézzelfogható adatokat szolgáltasson az utazásuk ökológiai lábnyomáról.

A rendszer képes becslést adni az utazás karbonlábnyomára, azaz a közlekedési eszközök, szálláshelyek és választott programok alapján kiszámítja, hogy mekkora CO₂-kibocsátással jár az adott út. Ezen túlmenően a kalkulátor nemcsak informál, hanem aktívan segít a környezetbarát alternatívák kiválasztásában. A felhasználó javaslatokat kap arra vonatkozóan, hogyan csökkentheti az utazás környezeti terhelését – például választhat alacsony kibocsátású közlekedési módot, fenntartható szálláshelyet vagy természetközeli programokat.

A funkció célja tehát kettős: egyrészt edukálja a felhasználót a fenntartható utazás lehetőségeiről, másrészt konkrét eszközöket biztosít a környezeti hatás csökkentésére. A kalkulátor elősegíti a fenntartható turizmus elterjedését, és ösztönzi a felhasználókat arra, hogy felelős utazási szokásokat alakítsanak ki. Ez nemcsak a környezet védelme szempontjából fontos, hanem hozzájárulhat a helyi közösségek támogatásához, a természeti értékek megőrzéséhez, valamint a hosszú távon fenntartható turisztikai infrastruktúra kialakításához.

### Funkcionális leírás

A fenntarthatósági kalkulátor modul több egymással összefüggő funkciót tartalmaz, amelyek együttesen biztosítják a környezeti hatások átfogó értékelését és a fenntartható alternatívák bemutatását. A rendszer célja, hogy a felhasználó ne csupán információt kapjon, hanem aktívan részt vehessen a környezetbarát döntéshozatalban.

* **Karbonlábnyom kalkuláció** A kalkulátor kiszámítja az utazás során keletkező becsült szén-dioxid-kibocsátást (CO₂), figyelembe véve a közlekedési eszközök típusát (pl. repülő, vonat, autó, busz), a szálláshelyek energiafelhasználását és fenntarthatósági minősítését, valamint a választott programok környezeti terhelését. A számítás alapjául nemzetközi szabványokon alapuló kibocsátási értékek szolgálnak, amelyeket a rendszer automatikusan frissít az aktuális adatok alapján. A kalkuláció eredménye egy összesített érték, amely megmutatja az utazás teljes karbonlábnyomát kilogrammban vagy tonnában kifejezve.
* **Közlekedési alternatívák összehasonlítása** A rendszer lehetőséget biztosít arra, hogy a felhasználó különböző közlekedési módok között válasszon, és összehasonlítsa azok környezeti hatását. Például egy repülőút kibocsátása jelentősen magasabb lehet, mint egy vonatos vagy buszos utazásé. A kalkulátor vizuálisan is megjeleníti az egyes opciók közötti különbségeket, és javaslatot tesz a leginkább környezetbarát megoldásra. A rendszer figyelembe veszi az utazás távolságát, az utasok számát, valamint az adott közlekedési eszköz energiahatékonyságát.
* **Szállásminősítés fenntarthatóság szerint** A kalkulátor kiemeli azokat a szálláshelyeket, amelyek zöld minősítéssel rendelkeznek, például energiatakarékos működésűek, helyi termékeket használnak, vagy környezetbarát szolgáltatásokat kínálnak. A rendszer integrálva van szállásfoglaló platformokkal, így a felhasználó közvetlenül láthatja, mely szálláshelyek felelnek meg a fenntarthatósági kritériumoknak. A minősítés alapjául nemzetközi tanúsítványok (pl. Green Key, EU Ecolabel) szolgálnak, és a rendszer részletes információt nyújt az egyes szálláshelyek környezeti teljesítményéről.
* **Programajánlás környezetbarát szempontok szerint** A kalkulátor olyan programokat ajánl, amelyek alacsony környezeti terheléssel járnak, például gyalogtúrák, kerékpáros városnézések, helyi természeti értékek felfedezése, vagy kulturális események, amelyek nem igényelnek nagy energiafelhasználást. A rendszer figyelembe veszi az úti cél adottságait, a programok elérhetőségét, valamint a felhasználó érdeklődési körét. Az ajánlások célja, hogy a felhasználó ne mondjon le az élményekről, hanem környezetbarát módon élvezhesse az utazást.
* **Összesített környezeti hatás megjelenítése**

### Felhasználói felület és dizájn

A fenntarthatósági kalkulátor felhasználói felülete vizuálisan letisztult, átgondoltan strukturált és környezettudatos megjelenésű, amely már első ránézésre is ösztönzi a felhasználót a zöldebb, felelősebb döntések meghozatalára. A dizájn célja, hogy a környezetbarát szemlélet ne csupán tartalmi szinten jelenjen meg, hanem vizuálisan is tükröződjön a felület minden elemében.

A felület kialakítása során kiemelt figyelmet kapott a színvilág, az interaktivitás, az információk vizuális megjelenítése, valamint a mobilos használhatóság. A modul minden eleme reszponzív, azaz automatikusan alkalmazkodik a különböző képernyőméretekhez, így biztosítva a zökkenőmentes használatot asztali gépen, táblagépen és okostelefonon egyaránt.

A főbb vizuális és funkcionális jellemzők az alábbiak szerint részletezhetők:

* **Színvilág** A modul sötétkék háttérrel és világoskék, valamint zöld kiemelésekkel jelenik meg. A sötétkék háttér nyugalmat, stabilitást és megbízhatóságot sugall, míg a világoskék árnyalatok a semleges információkat emelik ki. A zöld szín a fenntartható ajánlásokat jelöli, például alacsony kibocsátású közlekedési módokat vagy környezetbarát szálláshelyeket. A színpaletta kontrasztos, jól olvasható, és segíti a felhasználót abban, hogy gyorsan felismerje a környezetbarát lehetőségeket. A színek harmonizálnak a modul céljával, és vizuálisan is megerősítik a fenntarthatósági üzenetet.
* **Interaktív űrlap** A felhasználó egy könnyen kezelhető, lépésről lépésre vezetett űrlapon keresztül adhatja meg az utazás paramétereit. Ezek közé tartozik az indulási hely, a célállomás, a választott közlekedési mód, a szállás típusa, valamint a tervezett programok jellege. Az űrlap dinamikusan reagál a megadott adatokra, és azonnal frissíti a kalkulációs eredményeket. A beviteli mezők logikusan csoportosítottak, az egyes lépésekhez súgó szövegek és ikonok tartoznak, amelyek segítik az eligazodást. Az űrlap mobilon is könnyen kezelhető, az érintésre optimalizált elemek révén.
* **Karbonlábnyom diagram** A kalkulátor az egyes komponensek – például közlekedés, szállás, programok – környezeti hatását grafikus formában jeleníti meg. A diagram lehet oszlopdiagram, kördiagram vagy területdiagram, attól függően, hogy a felhasználó milyen vizuális megjelenítést preferál. A diagramok színkódoltak, interaktívak, és lehetőséget biztosítanak az egyes elemek részletes megtekintésére. Az adatok mellett rövid szöveges magyarázat is megjelenik, amely segíti az értelmezést. A vizualizáció célja, hogy a felhasználó könnyen átlássa, mely tényezők járulnak hozzá leginkább az utazás karbonlábnyomához.
* **Ajánlási panel** A kalkulátor egy külön ajánlási panelen keresztül alternatív lehetőségeket kínál, amelyekkel csökkenthető az utazás környezeti terhelése. Például javasolhatja a repülő helyett a vonat vagy busz használatát, a hagyományos hotel helyett eco-hostel választását, vagy a városi programok helyett természetközeli tevékenységeket. Az ajánlások személyre szabottak, figyelembe veszik a felhasználó által megadott paramétereket, valamint a célállomás adottságait. Az ajánlási panel vizuálisan elkülönül, zöld színnel kiemelve jelenik meg, és tartalmazza az alternatíva környezeti előnyeit is.
* **Mobilbarát kialakítás** A felület teljes mértékben reszponzív, azaz mobil eszközökön is hibamentesen működik. Az űrlapok, diagramok és ajánlások automatikusan igazodnak a képernyőmérethez, így biztosítva a jó olvashatóságot és kezelhetőséget. A mobilos nézetben a menük egyszerűsítettek, a gombok nagyobb méretűek, és az érintésre optimalizált interakciók révén a felhasználó könnyedén navigálhat az adatok között. A rendszer támogatja a mobilos értesítéseket is, így a felhasználó azonnal értesülhet, ha új ajánlás érhető el vagy ha változás történt a kalkulációban.

### Technikai megvalósítás

A fenntarthatósági kalkulátor technikai háttere korszerű, megbízható és skálázható megoldásokra épül, amelyek biztosítják a rendszer stabil működését, a gyors adatfeldolgozást és a biztonságos felhasználói élményt. A fejlesztés során kiemelt figyelmet fordítottunk az adatbiztonságra, a modularitásra és az integrálhatóságra, hogy a rendszer hosszú távon is fenntartható és bővíthető legyen.

A technikai komponensek az alábbiak szerint épülnek fel:

* **Backend** A szerveroldali működést egy Node.js alapú rendszer biztosítja, amely REST API-n keresztül végzi a számításokat és ajánlásokat. A backend felelős az adatok lekéréséért, feldolgozásáért, tárolásáért és továbbításáért a frontend felé. A REST API struktúrája jól dokumentált, és támogatja az aszinkron adatkezelést, amely lehetővé teszi a gyors válaszidőt és a skálázhatóságot. A szerveroldali logika tartalmazza a karbonlábnyom kalkulációs algoritmusokat, az ajánlási szabályokat, valamint az adatnaplózást.
* **Frontend** A felhasználói felület React technológiával készült, amely lehetővé teszi a dinamikus tartalomkezelést és a gyors felhasználói interakciókat. A React komponensalapú felépítése elősegíti a moduláris fejlesztést, a könnyű karbantartást és az új funkciók gyors integrálását. A frontend felelős az űrlapok megjelenítéséért, a diagramok rendereléséért, az ajánlások vizuális megjelenítéséért, valamint a felhasználói beállítások mentéséért.
* **Adatbázis** A rendszer MongoDB-t használ az adatok tárolására, amely rugalmasan kezeli a felhasználói beállításokat, kalkulációs előzményeket és ajánlási statisztikákat. A NoSQL struktúra lehetővé teszi a gyors lekérdezéseket, az adatok hatékony indexelését és a skálázható adattárolást. Az adatbázisban tárolt információk titkosítva kerülnek mentésre, és a hozzáférés jogosultságokhoz kötött, így biztosítva az adatbiztonságot.
* **Külső API integrációk** A rendszer több külső szolgáltatással van összekötve, amelyek biztosítják a kalkulációk és ajánlások pontosságát:
*  **Carbon Interface API**: Ez az API szolgáltatja a CO₂-kibocsátás számításához szükséges alapértékeket különböző közlekedési módokra, szállástípusokra és tevékenységekre vonatkozóan. A rendszer az API segítségével valós idejű adatokat kér le, és ezek alapján végzi el a karbonlábnyom kalkulációt.
*  **Green Travel Index**: Egy fenntarthatósági adatbázis, amely zöld minősítéssel rendelkező szálláshelyeket, programokat és szolgáltatásokat tartalmaz. A kalkulátor ebből az adatbázisból nyeri azokat az ajánlásokat, amelyek környezetbarát alternatívaként jelennek meg a felhasználó számára.
*  **OpenStreetMap**: A közlekedési útvonalak és távolságok lekérdezéséhez használt térképes adatforrás. A rendszer az indulási és célállomás alapján kiszámítja az utazás távolságát, és ennek megfelelően becsüli meg a közlekedési kibocsátást. Az OpenStreetMap integráció lehetővé teszi az útvonalak vizuális megjelenítését is, amely segíti a felhasználót az alternatívák összehasonlításában.

### Használhatóság és működés

A fenntarthatósági kalkulátor használata során a felhasználó az utazás megtervezésének kezdetén megadja azokat az alapvető paramétereket, amelyek az utazás környezeti hatásának kiszámításához szükségesek. Ezek a paraméterek magukban foglalják az indulási helyet, a célállomást, a választott közlekedési módot (pl. repülő, vonat, autó, busz), a szállás típusát (pl. hotel, apartman, eco-hostel), valamint a tervezett programok jellegét (pl. városnézés, túrázás, kulturális események).

A rendszer ezeket az adatokat automatikusan feldolgozza, és a háttérben lefuttatja a karbonlábnyom kalkulációs algoritmust, amely a megadott információk alapján becslést ad az utazás teljes CO₂-kibocsátására. A kalkuláció eredménye egy összesített érték, amelyet a rendszer vizuálisan is megjelenít, például kördiagram vagy oszlopdiagram formájában. A felhasználó így könnyen átláthatja, hogy mely tényezők járulnak hozzá leginkább az utazás környezeti terheléséhez.

A kalkulátor ezután javaslatokat tesz a környezetbarát alternatívákra. Ezek az ajánlások személyre szabottak, figyelembe veszik a felhasználó által megadott paramétereket, valamint a célállomás földrajzi és infrastrukturális adottságait. Például ha a felhasználó repülőutat tervez, a rendszer javasolhatja a vonat vagy busz használatát, ha elérhető. Ha a szállás hagyományos hotel, a kalkulátor ajánlhat eco-hostelt vagy zöld minősítéssel rendelkező szálláshelyet. A programok esetében a rendszer előnyben részesíti a természetközeli, alacsony kibocsátású tevékenységeket.

A felhasználó számára lehetőség nyílik arra is, hogy a kalkulátor által javasolt alternatívák alapján módosítsa az útvonalat vagy a választott szolgáltatásokat. A rendszer azonnal újraszámolja a karbonlábnyomot, és megmutatja, hogyan változik az utazás környezeti hatása a módosítások hatására. Ez a visszacsatolás segíti a tudatos döntéshozatalt, és ösztönzi a felhasználót a fenntarthatóbb opciók választására.

### Tesztelés és validálás

A fenntarthatósági kalkulátor megbízható működésének biztosítása érdekében a fejlesztési folyamat során kiemelt figyelmet fordítottunk a funkciók alapos tesztelésére és validálására. A cél az volt, hogy a rendszer pontos, releváns és felhasználóbarát módon működjön, miközben megfelel a nemzetközi környezetvédelmi elvárásoknak.

A tesztelés során az alábbi szempontokat vizsgáljuk:

* **Számítási pontosság** A kalkulált értékeknek meg kell felelniük a nemzetközi szabványoknak, valamint a külső API-k által szolgáltatott adatoknak. A tesztelés során összehasonlító elemzéseket végzünk, ahol a rendszer által generált karbonlábnyom értékeket összevetjük hiteles forrásokkal. A cél az, hogy a kalkulációk ne csak becslések legyenek, hanem valós, megbízható adatokon alapuló eredmények.
* **Ajánlások relevanciája** A rendszer által kínált környezetbarát alternatíváknak valóban relevánsnak és megvalósíthatónak kell lenniük. A tesztelés során ellenőrizzük, hogy az ajánlások figyelembe veszik-e a felhasználó preferenciáit, az úti cél adottságait, valamint a szolgáltatások elérhetőségét. A cél az, hogy a javaslatok ne legyenek általánosak, hanem valóban segítő szándékúak és személyre szabottak.
* **Felhasználói interakciók** Az űrlapok kitöltése során a rendszernek gyorsan és hibamentesen kell működnie. A tesztelés során vizsgáljuk, hogy a beviteli mezők logikusan vannak-e elrendezve, a súgók és ikonok segítik-e az eligazodást, valamint hogy a diagramok jól értelmezhetők-e. A cél az, hogy a felhasználó technikai előképzettségtől függetlenül könnyedén tudja használni a kalkulátort.
* **Mobiltesztelés** A funkció mobil eszközökön is teljes funkcionalitással kell, hogy működjön. A tesztelés során Android és iOS platformokon ellenőrizzük a felület reszponzivitását, az interakciók gördülékenységét, valamint az értesítések megjelenését. A cél az, hogy a felhasználók mobilon is ugyanolyan hatékonyan tudják használni a kalkulátort, mint asztali gépen.
* **Adatbiztonság** A felhasználói adatok védelme kiemelten fontos, ezért a rendszer titkosítással és jogosultságkezeléssel biztosítja, hogy az adatokhoz csak az arra jogosult felhasználók férjenek hozzá. A tesztelés során ellenőrizzük az adatátvitel biztonságát (HTTPS), a felhasználói hitelesítés működését, valamint a hozzáférési szintek helyes kezelését. A rendszer naplózza a műveleteket, így visszakövethető minden változtatás.

### Fejlesztési környezet

A fenntarthatósági kalkulátor fejlesztése során korszerű, iparági szinten elismert eszközöket és technológiákat alkalmaztunk, amelyek biztosítják a hatékony munkavégzést, a kód minőségét, valamint a projekt átláthatóságát és fenntarthatóságát.

* **Fejlesztői eszközök** A fejlesztéshez használt főbb eszközök közé tartozik a Visual Studio Code, amely modern, testreszabható fejlesztői környezetet biztosít.
* **Verziókezelés** A projekt Git alapú verziókövetést használ, amely lehetővé teszi a fejlesztési folyamat strukturált nyomon követését. Külön ágakat hoztunk létre a frontend és backend fejlesztéshez, valamint a teszteléshez, így biztosítva a párhuzamos munkavégzést és a hibamentes integrációt. A commit üzenetek szabványos formátumban készülnek, és minden változtatás dokumentálva van.