

## VAS VÁRMEGYEI SZAKKÉPZÉSI CENTRUM HORVÁTH BOLDIZSÁR KÖZGAZDASÁGI ÉS INFORMATIKAI TECHNIKUM

A 5 0613 12 03 számú Szoftverfejlesztő és –tesztelő vizsgaremek

# A szoftveralkalmazás dokumentációja

Készítették:

LAKI ZSOMBOR MOLNÁR BENEDEK SPAITS RICHÁRD

**SZOMBATHELY** 

2025

## 1. Bevezetés

Ez a dokumentáció a Szoftverfejlesztő és -tesztelő képzés keretében készített vizsgamunkánkat mutatja be. Egy ügyfél- és feladatkezelő szoftvert hoztunk létre, amely a kis- és magánvállalkozások számára egyszerűbbé és hatékonyabbá teszi az ügyfeleik és teendőik számontartását. Ebben a dokumentációban megpróbáljuk minél alaposabban ismertetni a fejlesztési folyamatot, az alkalmazás működését és a fejlesztéshez használt eszközöket. Emellett részletes leírásokkal bemutatjuk a program funkcióit és segítséget nyújtunk az oldal használatában.

## 1.1 Szoftver célja

A szoftverünk célja egy olyan letisztult, szabványoknak megfelelő, felhasználóbarát felületet biztosító alkalmazás kis- és magánvállalkozóknak, amellyel gyorsan és egyszerűen tudják vezetni és irányítani a vállalkozásukat. Sok magánvállalkozó tevékenységében előfordulhat az a probléma, hogy nem tudják ügyfeleiket hatékonyan nyilvántartani. Erre a problémára akkor figyeltünk fel, mikor megtudtuk, hogy egyik csapattagunk családtagjának vállalkozásában az ügyfelek adatait Excel táblában tartották nyilván. Ezért célunk, hogy a cégmenedzseléssel járó feladatokat egyszerűen tudják a felhasználóink elvégezni és egy átlátható képet kapjanak a cégük működéséről. Egy cég vezetése, a feladatok és ügyfelek számontartása egy megterhelő és időigényes feladat. Egy magánvállalkozó vagy egy kisvállalkozó számára nem áll rendelkezésre annyi pénz és erőforrás, hogy egy asszisztenst tudjon alkalmazni, aki beosztja az idejét és jegyzi a szükséges teendőket, emellett a vállalkozó szerepe elengedhetetlen a kevés alkalmazott miatt, emiatt rendkívül fontos, hogy az idejét az igazán fontos teendőkre tudja fordítani, ne pedig az adminisztrációs munkák és a számontartás akadályozzák a cég előrehaladását. Ezeket a problémákat szoftverünk egyszerű ügyfél-, alkalmazottés feladatbejegyzéssel, módosítással és átlátható statisztika monitorozással kívánja kiküszöbölni.

### 1.2 Felhasználói felületek

A projektünket úgy terveztük, hogy reszponzív legyen, azaz mind mobilon, tableten, asztali számítógépen vagy laptopon egy letisztult felhasználói élményt nyújtson. Figyeltünk arra, hogy az oldal egyes részei ne lógjanak ki a képből, ne takarják el egymást akkor sem, ha képernyőméret csökken. Ügyeltünk arra is, hogy az elemek kisebb képernyőkön is jól láthatóak legyenek és a mobil kijelzőkön is felhasználóbarát legyen az oldal.

## 1.3 Betűszavak és meghatározások

Felhasználó – A szoftver használója, a cég adminisztrátora. Bejegyzi a feladatokat és az ügyfeleket és kezeli azokat.

Ügyfél – A cég kliense. A cégen kívül eső személy, de együttműködik, üzletel a céggel közös feladatok esetében.

Alkalmazott – A cég dolgozója. Önállóan vagy ügyfelekkel együttműködve végez feladatokat.

Feladat – Egy alkalmazott vagy a felhasználó által elvégzendő tevékenység. Lehetnek önállóak, vagy csatlakozhatnak hozzá ügyfelek, ha szolgáltatásokról van szó. Határidejük van, túllépésük esetén lejárttá válnak.

Cégforma/Cégtípus – a cég vállalkozási formája (pl. kft, bt, zrt)

## 2. Készítési folyamat

### 2.1 Feladatok felosztása

A szoftver elkészítése egy hosszú és több lépésből álló folyamat, amely során számos problémát kell megoldani. A feladatokat az egyenlő munkamegosztás érdekében felosztottuk magunk között. A feladatokat a projekt céljai és követelményei alapján határoztuk meg, egy professzionális projekt létrehozásának lépéseit követve.

A feladatok felosztása során figyelembe vettük, hogy melyik feladat mennyire időigényes és mennyire bonyolult a megoldása, törekedtünk az egyenlő munkamegosztásra. Emellett arra is ügyeltünk, hogy a feladatok a csapattagok szakértelmének megfelelően legyenek kiosztva. A feladatok kiosztása mellett felelősöket is neveztünk ki, hogy tudjuk egymás munkáját ellenőrizni.

| Feladat                             | Készítő                  | Felelős           |
|-------------------------------------|--------------------------|-------------------|
| Adatbázis – Egyedek összegyűjtése,  | Felosztva a tagok között | Felosztva a tagok |
| ismertetése                         |                          | között            |
| Adatbázis – ER Modell               | Molnár Benedek           | Spaits Richárd    |
| Adatbázis - Relációs modell         | Molnár Benedek           | Spaits Richárd    |
| Adatbázis – Modellezés              | Spaits Richárd           | Laki Zsombor      |
| Adatbázis – Mintaadatok             | Molnár Benedek           | Spaits Richárd    |
| Backend – Végpontok                 | Spaits Richárd           | Molnár Benedek    |
| Backend – Modellek                  | Spaits Richárd           | Molnár Benedek    |
| Backend – Egység tesztelés          | Spaits Richárd           | Laki Zsombor      |
| Frontend – Bejelentkező/regisztráló | Laki Zsombor             | Molnár Benedek    |
| felület                             |                          |                   |
| Frontend – Homepage/vezérlő felület | Molnár Benedek           | Laki Zsombor      |
| Frontend – Statisztika oldal        | Laki Zsombor             | Molnár Benedek    |
| Frontend – Naptár oldal             | Laki Zsombor             | Molnár Benedek    |
| Frontend – Ügyfelek oldal           | Laki Zsombor             | Molnár Benedek    |
| Frontend – Dizájn                   | Laki Zsombor             | Spaits Richárd    |
| Frontend – Manuális tesztelés       | Laki Zsombor, Molnár Be- | Spaits Richárd    |
|                                     | nedek                    |                   |
| Koordinátor                         | Molnár Benedek           | Laki Zsombor      |

## 2.2 Az adatbázis megtervezése

#### 2.2.1 Az ER-modell megtervezése

Az adatbázis megtervezésének első lépése az ER-modell, vagyis az egyed-kapcsolat modell megtervezése volt. Az ER-modellben meghatároztuk azokat az egyedeket, amiket fontosnak tartottunk ötletünk megvalósításához. A legfőbbek, amik az alapkövei az adatbázisunknak az a személy, cég és feladat. Vannak a cégek, a cégekhez tartoznak emberek különböző szerepkörökkel, ezek a felhasználó, az alkalmazott és az ügyfél. Az oldalt csak felhasználó típusú emberek tudják használni, ők vezetik a cég adminisztrációját. Az alkalmazottak a cégen belül dolgozó emberek, az ügyfelek pedig a cégen kívüli kliensek, akikkel a cég együttműködik vagy szolgáltatást végez nekik. A személyekhez tartoznak feladatok. Ezeket azok, amiket mi alapvetőnek tartottunk az egész program működéséhez, ezek jelentik a magját az adatbázisunknak. Ezekhez az egyedekhez kapcsolódik a többi egyed. A céghez kapcsolódik egy cégformák és egy cím nevű egyed. A feladatokhoz tartozik egy szín nevű egyed, ami a feladatok színkódja miatt fontos, mivel azt szerettük volna, hogy legyen a felhasználónak lehetősége színek alapján beazonosítani a feladatokat vagy azok típusát. A színek külön táblát kaptak az adatbázisban. Ezt azért terveztük így, mert a színeket nem kód, hanem a felhasználóbarátság érdekében név alapján válogathatja a felhasználó.

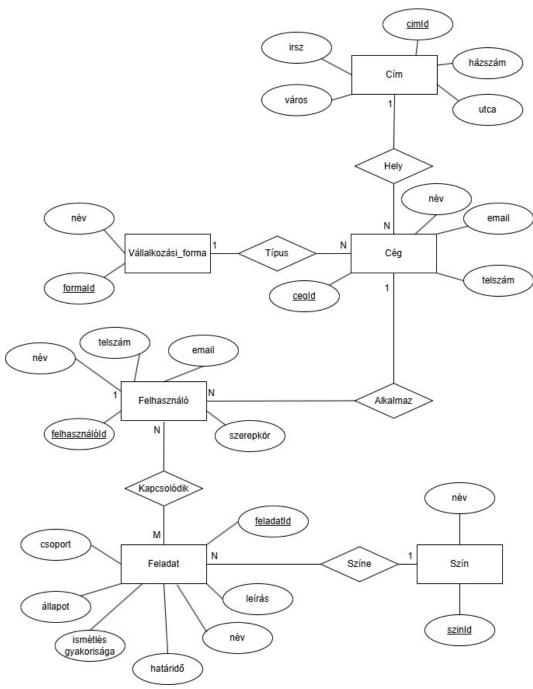
### Egyedek:

- Feladatok (Név, Leírás, Ismétlés gyakorisága, Állapot, Szín, Határidő)
- Felhasználók (Név, Telefonszám, Email, Jelszó (titkosított), Szerepkör)
- Cég (Név, Telefonszám, Email)
- Formák (Név)
- Címek (Házszám, Utca, Településnév, Irányítószám)
- Színek (Név, Színkód)

#### Egyed kapcsolatok:

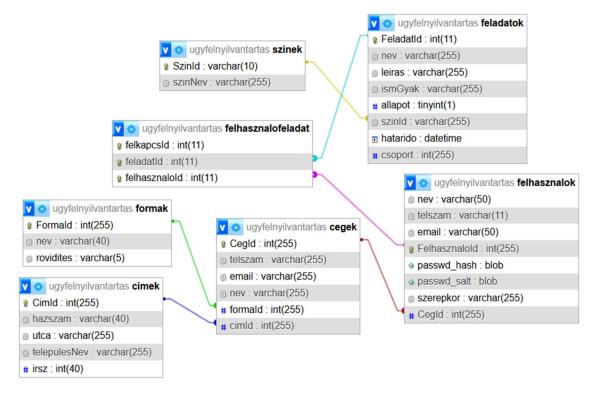
- A Feladatok tábla több-több kapcsolatban áll a Felhasználók táblával, egy feladathoz több felhasználó is kapcsolódhat, ugyanis a felhasználó táblában ügyfeleket és alkalmazottakat is tárolunk. Egy felhasználónak több feladata is lehet.
- Egy feladatnak egy színe lehet, viszont egy szín több feladathoz is kapcsolódhat.
- Egy Felhasználónak egy Cége lehet, viszont egy Cégben több Felhasználó is lehet. Úgy lehet meghatározni, hogy melyik ügyfél melyik felhasználóhoz tartozik, hogy az ügyfélnek ugyan az a cégId-je, mint a felhasználónak. A cég alatt vannak az alkalmazottak is, szintén a cégId dönti el melyik Felhasználóhoz tartozik.

- Egy Cégnek egy Cégformája van, viszont egy Cégforma több céghez is kapcsolódhat.
- A Cégnek van címe



2.2.3 ER modell

#### 2.2.2 A relációs modell



2.2.4 Relációs modell

## 2.3 A felhasználói felület megtervezése

#### 2.3.1 A felhasználói felületek felosztása

A felhasználói felületet két részre bontottuk. Az első rész, amivel egy új felhasználó találkozik, az a főoldal. Ezen az oldalon van lehetőség regisztrálni, vagy ha van már a felhasználónak fiókja, akkor beléphet oda a belépés menüponttal.

Ha megtörtént a belépés, akkor kerülünk be az szoftver másik részére, ahol már tudunk ténylegesen tevékenykedni.

#### 2.3.2 A felhasználói felület kinézete

A felhasználói felülettel törekedtünk a modernségre és az egyszerűségre. A mai weboldalak szabványai alapján terveztük meg és készítettük el az oldalt. Próbáltuk azt elérni, hogy egy új felhasználó számára is intuitív legyen a weboldal használata és az oldal vizuális megjelenése is vonzó legyen, professzionalizmust sugározzon.

A legszembetűnőbb dolog egy weboldalon az az elemek és a háttér színe, illetve a betűtípus. Az elmúlt években népszerűvé váltak a sötét témájú oldalak, a legtöbb ember ezeket preferálja, mivel a szemnek nem annyira fárasztó és megterhelő, főleg egy kevésbé megvilágított szobában. Ezen okok miatt választottuk a sötétkéket az oldal fő színének. Ez a szín profeszszionalizmust és megbízhatóságot sugall, emiatt úgy gondoljuk, hogy a kék szín illik az oldal adminisztrációs jellegéhez. A legtöbb elem a sötétkék valamely árnyalata, ahol viszont szükséges, ott eltérünk ettől.

```
export const colors = {
   background: "#152238",
   button: "#2a3b5f",
   card: "#1c2e4a",
   text: "#c5cacd",
   buttonHover: "#3b4e7a",
   secondaryBackground: "#f5f5f5",
   completed: "#4caf50",
   uncompleted: "#f44336",
};
```

2.4.2 - Használt színek

Ezen kívül még a feladatoknál is fontos szerepet játszanak a színek, mert a feladatok háttérszínét a felhasználó választhatja ki. Ez akkor hasznos, ha valamilyen szempont alapján csoportosítani szeretné a feladatokat, mivel így ránézésre könnyen meg lehet különböztetni a különböző csoportú feladatokat.

## 3. Komponensek technikai leírása

#### 3.1 Frontend

A frontend React.js keretrendszer 18.3.1-es verziója segítségével készült a Vite 6.2.3-as verzióján belül, a komponensekre bontott oldalrendezés és a beépített eszközök előnyeit kihasználva. Ennek köszönhetően könnyen átlátható, és egyszerűen bővíthető plusz funkciókkal. Kifejezetten hasznos volt a fejléc és a lábléc elhelyezésében, ugyanis nem kellett minden oldalon újra létrehozni ezeket, hanem csak a szükséges komponenseket helyeztük el a kódban. A szoftver elkészítésében a React saját modulkönyvtára, illetve külső modulok is nagy szerepet játszottak. Meghívott modulok:

#### • react-dom:

A React hivatalos csomagja, a React komponensek DOM-ba (Document Object Model) való rendereléséért felel.

#### • react-router-dom

Lehetővé teszi, hogy többoldalas alkamazások készítését navigációval, útvonalkezeléssel és dinamikus URL-ekkel.

#### react-toastify

Modern letisztult felugró ablakok készítésére alkalmas.

#### • is-cookie

Sütik olvasására, írására, törlésére használatos.

- @mui/material
- @mui/x-charts
- @emotion-react
- @emotion/styled

Az egyszerű oldalszerkezet és modern megjelenésért felelős könyvtárak.

A modern és egységes kinézetet a Material UI komponenskönyvtár és a CSS (Cascading Style Sheets) stílusleíró nyelv biztosítja.

#### 3.2 Backend

A backend C# nyelvben, ASP.NET webalkalmazás-keretrendszerrel lett megírva. A backend fontos szerepet játszik a felhasználó autentikációban, a bevitt adatok vizsgálásában és az ismétlődő feladatok kezelésében is. A backend RESTful API-t valósít meg, amelyek az ügyfél-, feladat- és céges adatok kezelését végzik. Az adatbázis-kapcsolatot Entity Framework Core segítségével kezeljük.

A backend fő elemei a Controllerek, melyek bizonyos metódusokat (végpontok) tartalmaznak. Ezek a metódusok végzik el az összekötött adatbázissal a műveleteket, ami tartalmazza az adatok lekérdezését, módosítását. Minden végpontnak van egy elérési útvonala, ami segítségével tudja a Frontend az adott függvényeket meghívni és a Frontend változóit használva paramétereket megadni. Az adatbázisból szerzett adatokat modellek segítségével használjuk fel, melyek speciális osztályok, amelyek az adatbázis tábláira hivatkoznak. A modellek mezőiként adjuk meg az adatbázis oszlopainak megfelelő változók neveit és típusait. Ezeket a modelleket a végpontok használják arra, hogy a metódusnak paraméterként megadott objektumaiból, és azok adataiból egy példányt készítve tudják a kapott értékeket feldolgozni.

Minden metódus a futtatásuk után egy HTTP státusz kódot (pl.: 200 OK, 404 Not Found), és ha szükséges, az adatbázisból lekért adatokat adja vissza, amelyeket a Frontend fel tud használni. A HTTP kódok biztosítják, hogy a weboldalon, hiba esetén is megkapja a szükséges információkat a felhasználó. Legtöbb esetben a felhasználó követ el hibát adatbevitelkor (pl.: Bejelentkezés során hibás jelszót ad meg), így létfontosságú, hogy az ebből adódó hibák kezelve vannak.

A jelszavak ellenőrzése a következő elven történik: Regisztráció során a megadott jelszóból egy titkosított, byte típusú változót töltünk fel az adatbázisba. A bejelentkezéskor megadott jelszót ugyanazzal a titkosítási módszerrel átalakítjuk és összehasonlítjuk az adatbázisban szereplővel. Amennyiben a kettő megegyezik, a jelszó helyes.

#### 3.3 Adatbázis

A szoftver adatbázisban tárolja a bejegyzett ügyfeleket, létrehozott felhasználókat és a feladatokat. Projektünkben MariaDb-t használtunk, ami egy SQL alapú relációadatbázis-kezelő szerver.

## 4. A program felhasználói felületének bemutatása

#### 4.1 Oldalak

#### 4.1.1 Főoldal

A főoldal az első oldal, amit egy új felhasználó lát. A főoldalon olvasható néhány bekezdésnyi szöveg arról, hogy mit ajánl a mi szoftverünk, illetve néhány pontban felsoroltuk, miért érdemes minket választani, milyen funkciókkal áll a szoftverünk felhasználóink rendelkezésére.

A főoldalon megtalálható egy "Tervezz!" felirat gomb is, ami a regisztrációs felületre visz, illetve a fejlécben el van helyezve, egy bejelentkezés gomb is a visszatérő felhasználóink számára.

### 4.1.2 Bejelentkezés

A bejelentkező felületen a már regisztrált és létező fiókkal rendelkező felhasználók léphetnek be. Ehhez az email címüket és a jelszavukat kell megadni. Ezután a "Bejelentkezés" gomb megnyomásával sikeres bejelentkezés esetén kerülnek be az irányító felületre.

Ha hibás email címet vagy hibás jelszót adnak meg, akkor sikertelen lesz a bejelentkezés és nem fognak tudni belépni.

Az oldalon még található egy "Még nincs fiókod?" felirat is, amely mellett lévő "Regisztráció" szövegre kattintva a regisztrációs felületre kerül a felhasználó.

#### 4.1.3 Regisztráció

A regisztrációs oldal kinézetben hasonlít a bejelentkező oldalra. Itt a felhasználónak az email címét, nevét, telefonszámát és egy jelszót kell megadnia.

A jelszónak legalább 8 karakternek, legfeljebb 20 karakternek kell lennie, tartalmaznia kell kis- és nagybetűket, legalább egy számjegyet és nem tartalmazhat különleges karakterek. A jelszót az elírások elkerülése miatt kétszer kell megadni. Az email cím formátumának meg kell felelnie a szabványnak, illetve nem lehet olyan email címet megadni, amivel már korábban regisztráltak fiókot. A telefonszámnak pedig meg kell felelnie a magyar telefonszám szabványnak.

Ha minden mezőt kitöltött a felhasználó és az összes adat megfelelően van megadva és a jelszavak is megegyeznek, akkor a felhasználó átkerül a regisztrációs folyamat második csőfokára, a cégének a regisztrálására. Ha valamilyen adatot helytelenül adott meg a felhasználó, akkor jelzi neki az oldal és módosítás után újra próbálkozhat a regisztrációval.

Az oldalon még található egy "Már van fiókod?" felirat is, amely mellett lévő "Bejelentkezés" szövegre kattintva a bejelentkező felületre kerül a felhasználó.

### 4.1.4 Cég regisztrálása

A cég regisztrálása a második része a regisztrációs folyamatnak. Ezen az oldal a céggel kapcsolatos adatokat kell megadni, amik a cég email címe, a cég neve, a cég telefonszáma, földrajzi elhelyezkedése és a cégnek a típusa.

Az email címnek meg kell felelnie a szabványnak és nem lehet olyan címet adni a cégnek, amivel már van cég regisztrálva. A cég nevének legalább 4 karakter hosszúságúnak kell lennie. A cég telefonszámának, csak úgy, mint a felhasználó telefonszámának, meg kell felelnie a magyar szabványnak.

A földrajzi elhelyezkedés 4 mezőből áll, ezek az irányítószám, város, utca, házszám. Az irányítószám beírásával automatikusan kitölti a weboldal a város mezőt. Ha az irányítószám nem a beírt városra mutat, vagy a beírt városhoz nem a megadott irányítószám tartozik, akkor a regisztrációnál nem fogja elfogadni az oldal.

A cég típusát egy legördülő listából kell kiválasztani.

Ha minden adat megfelelő, akkor a cég létrehozása gomb megnyomásával a felhasználó és a cég regisztrációja is sikeres. Ezután visszakerül a felhasználó a bejelentkező oldalra, ahol a létrehozott fiók adataival bejelentkezve kap teljes hozzáférést az oldal mindegyik funkciójához.

#### 4.1.5 Feladataid

A bejelentkezés után az első oldal, amit a felhasználó lát. Egy üdvözlő szöveg alatt találhatóak az aktuális hónap feladatai. Ha a felhasználó először lépett be, egy felugró ablak meg fogja kérdezni, hogy szeretne-e az oldal használatával kapcsolatos tippeket látni. Ha igenre nyom, a kurzor húzásával súgóüzenetek jelennek meg. Erről a funkcióról részletesebb leírás a 4.2.7-es pontban található A feladatokat hónapokra lehet szűrni az év és hónapválasztó legördülőlistákkal. A legördülőlisták mellett egy kijelölődoboz helyezkedik el, ami megnyomásával csak a teljesítetlen állapotú feladatokat jeleníti meg a rendszer. Ebben a sorban helyezkedik el az új feladat hozzáadása is.

## 4.1.6 Ügyfelek

Az Ügyfelek oldal a felhasználók számára az általuk rögzített ügyfelek kezelésére szolgál. A felület átlátható módon támogatja az ügyfelek böngészését, felvitelét, szerkesztését és eltávolítását. A weblapon látható az ügyféllista, ahol egy névre kattintva megjelennek az adott ügyfél adatai, valamint az őt érintő feladatok. Egy feladatra kattintva felugró ablakban tekinthetők meg annak részletei, amelyek szükség esetén módosíthatók. A kiválasztott ügyfél adatait felugró ablakban lehet szerkeszteni.

#### 4.1.7 Céged

Ez az oldal szolgál a cég alkalmazottainak kezelésére. Ugyanúgy működik, mint az ügyfelek oldal, viszont itt a cég alkalmazottjait és feladataikat lehet kezelni.

#### 4.1.8 Statisztika

A statisztika oldalon az ügyfelek és az alkalmazottak produktivitásáról kapunk egy átfogó képet. A megvizsgálni személy kiválasztása után átfogó adatokat kapunk az éves teljesítményéről. Látható, hogy mennyi feladatban vett rész és milyen arányban teljesítette ezeket. A grafikon tetején lévő nyilakkal lehet az évet módosítani.

## 4.1.9 Profil

A profil oldal a fiókhoz tartozó adatokat jeleníti meg. A módosítás gombra nyomva van lehetőségünk az adatokat módosítani. A módosítani nem kívánt adatokat üresen hagyhatjuk, ebben az esetben az adott adat változatlan marad. A mentés gombra kattintva felülírhatjuk korábbi adatainkat. Az oldalon az adatmódosítás mellett lehetőség van átlépni a cég profiljára, illetve a kijelentkezésre.

## 4.1.10 Cégprofil

A cégprofil a céghez tartozó adatokat jeleníti meg. A cégprofil szerkezete és használata megegyezik a profil oldallal. Az oldal a cég adatinak megváltoztatására szolgál.

#### 4.2 Az oldal funkciói

#### 4.2.1 Felugró ablakok

A feladatok, ügyfelek és alkalmazottak hozzáadása és szerkesztése felugró ablakokban történik. Ez a legnagyobb előnyt a feladatoknál jelenti, mivel több oldalon is szerepel és van, hogy más környezetben, ezért más értékekkel nyílik meg.

Például a feladat hozzáadása 3 oldalon is lehetséges és egy kicsit mindegyik különbözik, de ezt könnyen lehet kezelni a felugró ablaknak elküldött változókkal. A *postToUser* nevű változóban mindig az a céges dolgozó van eltárolva, akihez kapcsolódik a feladat. Az *ugyfel* nevű változónak csak akkor van megadva érték, hogyha az Ügyfelek oldalon van egy kiválasztott ügyfél és úgy adunk hozzá feladatot. Ilyenkor alapértelmezettként ki van választva az ügyfél. Lehet az *ugyfel* változó üres is, ilyenkor pedig opcionálisan lehet egy lenyíló listából kiválasztani az ügyfeleket.

```
export default function NewTaskPopup({
   trigger,
   closePopup,
   refreshData,
   postToUser,
   ugyfel,
}) {
```

4.2.1 A NewTaskPopup dinamikus tulajdonságai

## 4.2.2 Új feladat hozzáadása

Az oldal legfontosabb funkciója. A Feladataid, Ügyfelek és Céged oldalokon elérhető. A feladataid oldalon egy kör alapon plusszal jelzett gombbal hozható elő. A feladat címe és határideje megadása kötelező, míg a többi mező kitöltése opcionális, amiket üresen hagyunk, alapértelmezett értéket kapnak. Ez az ismétlődésnél alapértelmezett az "Egyszeri", a szín esetében pedig a "Zöld". A határidő nem lehet korábbi az aktuális dátumnál, és az évnek 2101-nél kisebbnek kell lennie. A rendszer megakadályozza az érvénytelen dátumok bevitelét: például, ha a felhasználó a hónap mezőbe "4"-et ír, a rendszer automatikusan a nap mezőre vált, mivel a 40 nem érvényes hónap. A nap mezőbe, ha 30 vagy 31-nél többet írunk, a beviteli mező fókuszváltásánál visszaugrik a maximum lehetséges értékre. A feladatot a /api/FelhasznaloFeladat/PostFeladatWithFelhasznalo backend metódussal küldjük, ami lehetővé teszi, hogy a több-több kapcsolat (FelhasznaloFeladat) táblában is megjelenjenek az összekötő adatok. Ha a feladatrögzítés sikertelen, egy react-toastify felugró ablakban jelzi azt az oldal, az aktuális hibaüzenettel. Sikeres feladatrögzítés esetén szintén felugró ablakban jelenik meg az erre utaló üzenet.

#### 4.2.3 Feladat módosítása

A felugró ablak szerkezete megegyezik a feladat hozzáadásához használt felugró ablakkal. A feladat fejlécében megjelenik a felhasználó, az ügyfél és az alkalmazott. A feladat részleteinek módosítása alapjáraton le van tiltva, ez a lila ceruza ikonnal lehet feloldani. A letiltás feloldása után a felhasználó kedve szerint változtathatja a feladat részleteit: A feladat nevét, leírását, határidejét és idejét, ismétlődését és színét. Ezeken kívül az állapotát is tudja változtatni "Teljesített" vagy "Nincs teljesítve" közül választva egy kapcsolóval. A feladatot törölni is lehet a kuka ikonnal. Mentésre vagy törlésre kattintás után az oldal megkérdezi, hogy az összes ilyen feladatot, vagy csak ezt az egyet szeretné módosítani, törölni. Erről a funkcióról részletesebben a <u>4.2.4</u>-es pontban lesz szó. A "Csak ezt a feladatot" lehetőség kiválasztása után törlés esetén, az oldal a /api/Feladatok DELETE metódusával törli az adatbázisból a feladat rekordját, és a (FelhasznaloFeladat) kapcsolótáblában hozzátartozó értékeket. Mentés esetén a /api/Feladatok PUT metódusával változtatja meg a rekord részleteit.

#### 4.2.4 Feladatok ismétlése

A szoftver egyik leglényegesebb funkciója a feladatok ismétlése. Ez a /api/Feladatok/Repeat metódus minden alkalommal lefut, amikor az oldal frissül. Ha egy feladat határideje lejárt, és az állapota "Teljesített", a backend metódus újra rögzíti az adatbázisba, a feladat ismétlődésének megfelelően: például, ha az ismétlődés "Havi", a határidő már lejárt 2025-04-12-én, és "Teljesített", a feladat újra megjelenik az adatbázisban 2025-05-12 határidővel.

Ezek a feladatok egy "csoportba" tartoznak. Ezeket a feladatokat egybe lehet változtatni és törölni is akár. A módosításnál az aktuális szerkesztett feladat id-jét a /api/Feladatok/PutAll/id metódussal küldjük a backendnek, ahol a feladat csoport mezőjét megvizsgálja, és összegyűjti az összes azonos csoport mezővel rendelkező feladatot. Ezek után az összesnek megváltoztatja az értékeit. A törlés hasonló logika alapján működik, a /api/Feladatok/DeleteAll/id metódus összegyűjti az összes azonos csoport mezővel rendelkező feladatot, majd törli azokat és a FelhasznaloFeladat táblában talált értékeit.

Backenden ez a következőképpen zajlik: A feladat táblából kiválogatjuk azokat a rekordokat, melyeknek a határideje a mai dátumhoz képest lejárt és az állapotuk kész (boolean true érték), és egy listában tároljuk őket. A lista elemein végigjárva, az elemek értékeiből létrehozunk egy új objektumot, melynek az új határidejét az ismétlés gyakorisága határoz meg. Így egy heti feladat új határideje 7 nappal később lesz, az eredetihez képest, és az állapota "Nincs kész" lesz. Ezeket az ismétlődő feladatokat egy Csoport mező azonosítja, ami a feladat létrehozásakor megegyezik az Id-jével, ismétléskor pedig az ismételt feladat Csoport-jával, ami lehetővé teszi a 4.2.3-ban említett csoportos vagy egyedi módosítást.

4.2.4 Ismétlés napjainak meghatározása

## 4.2.5 Ügyfelek és alkalmazottak bejegyzése

Ügyfeleket és alkalmazottakat külön-külön felugró ablakban lehet hozzáadni a céghez. Az létrehozott személyekhez nem tartozik jelszó, így ezekbe a fiókokba nem lehet bejelentkezni. A szerepkör mellett még abban is különbözik a két hozzáadás egymástól, hogy az ügyfél email címének csak a cégen belül kell egyedinek lennie, míg az alkalmazotténak az egész adatbázisban. Ez azért szükséges, mert egy ügyfél több céggel is kapcsolatban állhat, viszont egy alkalmazott egyszerre csak egy céghez tartozhat.

### 4.2.6 Ügyfelek és alkalmazottak oldalak sajátosságai

Az ügyfelek és alkalmazottak betöltése után a TAB billentyűvel előre, a Shift + TAB-bal pedig hátra lehet lépkedni a személyek között. A kijelölést az ügyfelek kiválasztása melletti gombbal lehet megszűntetni.

### 4.2.7 Súgóüzenetek

A weboldal felhasználóbarát felületet biztosít a felhasználóknak, és ebben a súgóüzenetek nagy szerepet játszanak. Ha egy böngészőből először indul az oldal, bejelentkezés után egy felugró ablakban megkérdezi a felhasználótól, hogy szeretne-e tippeket látni. A felhasználó válaszával megegyező "true" vagy "false" értéket ment el a böngésző a sütik közé, ami 365 nap után lejár, és törlődik a rendszerből. A válasz megadását következően nem fog megjelenni többet a felugró ablak egészen a süti lejárásáig. Ezt az értéket a Profil oldalon lehet változtatni egy kapcsolóval. Igen válasz esetén az oldal komponensein "Hover" cselekvéssel lehet ezeket a súgóablakokat előhívni. Nem válasz esetén a súgóüzenetek nem jelennek meg.

#### 4.2.8 Statisztika oldal működése

Az oldalon a megjelenő személyek egyikének kiválasztása esetén az az oszlopdiagrammon látható lesz hónapokra lebontva a feladatainak száma. A diagram alatt megjelenik egy éves összegzés is az összes és a teljesített feladatok számáról. Az oszlopdiagram mobilos, vagyis vertikális nézetben vízszintes oszlopokban, míg asztali nézetben függőleges oszlopokban jeleníti meg az adatokat, így könnyebben olvashatóak. Egy oszlopra kattintva az adott hónap kerül kiválasztásra. Újbóli kattintással megszűnik a kijelölés.

Az oszlopdiagram alatt egy kördiagramban a kiválasztott személy teljesített és nem teljesített feladatainak aránya látható a kijelölt hónapra vonatkozóan. Ha nincs hónap kiválasztva, a diagram az egész éves adatokat jeleníti meg. A személyek listáját szerepkör szerint lehet szűrni.

#### 4.2.9 Felhasználó adatainak módosítása

A felhasználó adatait a profil oldalon lehet módosítani. Az adatok módosításhoz szükséges a jelszó megadása is. A mentés gomb megnyomása után helyes jelszó esetén a régi adatokat felülírják az új adatok.

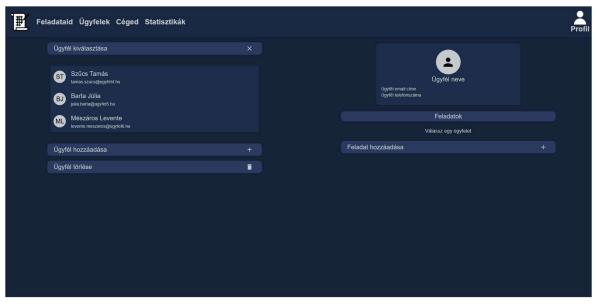
A jelszó módosítása gombra nyomva egy felugró ablakban lehet a jelenlegi jelszót megváltoztatni.

Ebben az ablakban meg kell adni a régi jelszót, az új jelszót és az új jelszó ismétlését. Ha a régi jelszó helyes, az új jelszó megegyezik az ismétléssel és megfelel a követelményeknek, akkor a jelszó megváltozik, titkosítva kerül az adatbázisba.

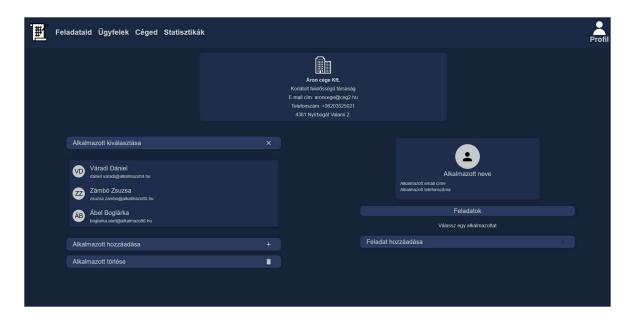
## 4.3 Monitorképek



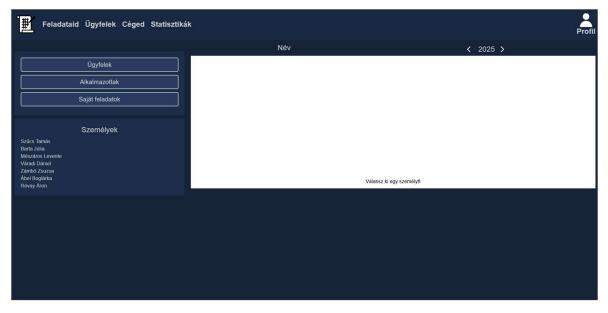
4.3.1 Feladataid oldal



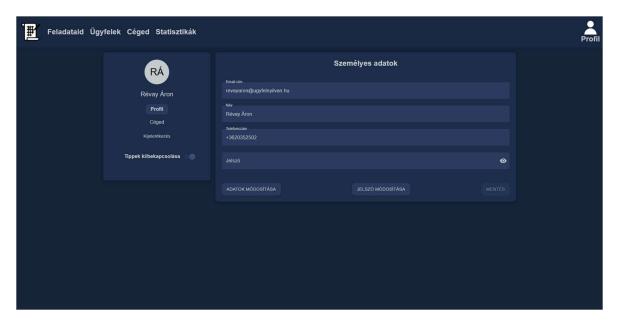
4.3.2 Ügyfelek oldal



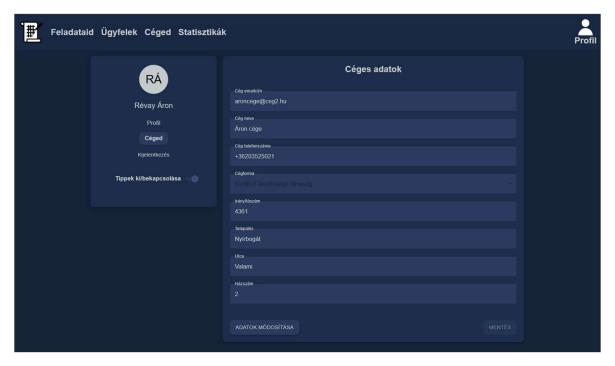
4.3.3 Céged oldal



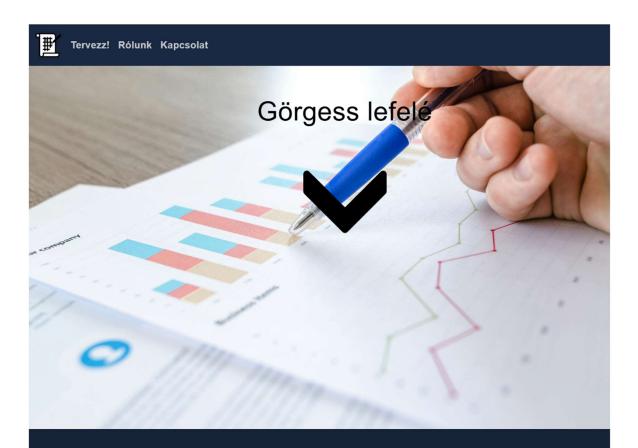
4.3.4 Statisztika oldal



4.3.5 Profil oldal



4.3.6 Cégprofil oldal





# TERVEZZ ÚGY ELŐRE, MINT MÉG SOHA

A vállalkozások hatékony működéséhez elengedhetetlen a jól szervezett ügyfél- és alkalmazottkezelés, valamint a feladatok precíz nyomon követése. Platformunk lehetővé teszi, hogy egy helyen kezelje cége minden fontos adatát – ügyfeleit, munkatársait és teendőit. Egyszerűen regisztrálhat vállalkozást, hozzárendelhet alkalmazottakat és ügyfeleket, és könnyedén rögzítheti, nyomon követheti a feladatokat.

Szolgáltatásunk intuitív felületet biztosít a hatékony munkavégzéshez. A beépített statisztikai modul segítségével valós időben elemezheti cége teljesítményét, így gyorsan azonosíthatja a fejlődési lehetőségeket. Automatizált rendszereinkkel minimalizálhatja az adminisztrációs terheket, miközben növeli az üzleti hatékonyságot és átláthatóságot.

Válasszon minket, ha egy megbízható, könnyen kezelhető és átfogó üzleti megoldást keres, amely segíti vállalkozása gördülékeny működését és növekedését!

# MIÉRT VÁLASSZ MINKET?



Határidők

Állítsd be munkáid határidejeit, és



Ügyfelek

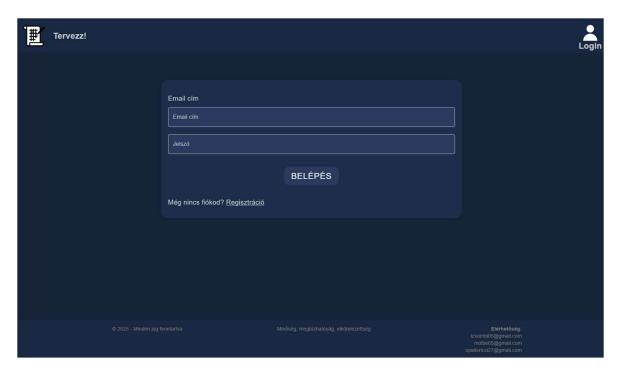
Mentsd el és szerkeszd ügyfeleid és



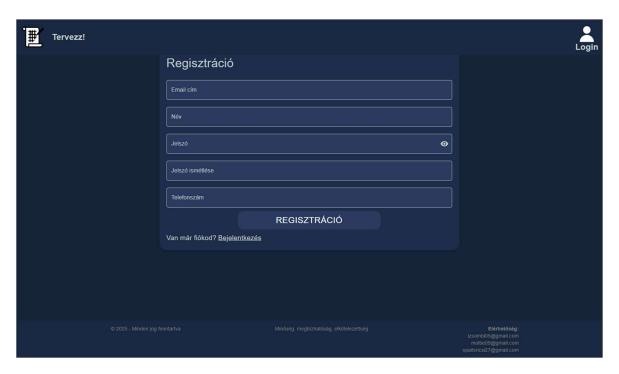
**Produktivítás** 

Figyeld meg, ahogy egyre hatékonyabban yégzed a n

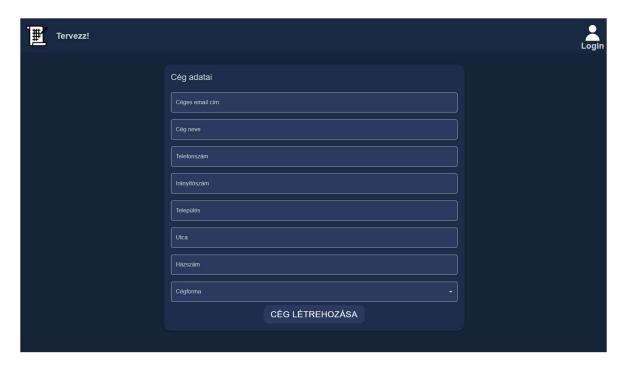
4.3.7 Kezdőoldal



4.3.8 Bejelentkezés oldal



4.3.9 Regisztráció oldal



4.3.10 Cég regisztráció oldal

## 5. A szoftver működésének műszaki feltételei

A program futtatásához az alábbi alkalmazások kellenek:

- MySql
- Visual Studio 2019
- NodeJs v20.8.1

Műszaki és rendszerkövetelmények (minimum):

• Processzor: Intel Pentium G4400 2 Core 3.3GHz

• Videókártya: integrált

• Memória: 4GB

### 6. Tesztelés

## 6.1 Egységtesztelés (Backend)

A Backend helyes működését egységtesztek (Unit Test) ellenőrzik. Az egységtesztek a kód egyes részeit tesztelik zárt környezetben, jelen esetben a feljebb leírt végpontokat.

A tesztek egy tesztadatbázissal dolgoznak, ami a rendes adatbázis szerkezetével megegyezik, azonban csak ideiglenesen vannak tárolva. A teszt során, a tesztadatbázison lefutatott metódusok sikerességét vizsgáljuk, biztosítva, hogy éles futtatásnál valóban az elvárt eredményeket kapjuk meg.

## 6.1.1 Példa és magyarázat

```
[TestMethod]
0 references
public void Get_ShouldReturnSelectedTask()
{
    var ctx = new TestNyilvanContext();
    FillTestDatabase(ctx);

    var controller = new FeladatokController(ctx);
    var result = controller.Get(1) as OkNegotiatedContentResult<Feladatok>;

    Assert.IsNotNull(result);
    Assert.AreEqual("TesztFel2",result.Content.Nev);
}
```

6.1.1 Példateszt

A ctx nevű változó a tesztadatbázis, melynek 3 feladat típusú objektumot adunk, minden kötelező adatot megadva. Ezen a tesztadatbázison lefuttatjuk a FeladatokController paraméteres Get metódusát 1-es FeladatId-t keresve, melyet a result változó tárol. Mivel a kapott objektum név értéke megegyezik a tesztadatbázisban lévővel, a teszt sikeresnek minősül.

## **6.2 Manuális tesztelés (Frontend)**

A projekt fejlesztése során manuális teszttel ellenőriztük a felhasználó felület funkcióit. Ennek célja, hogy a hibákat felfedjük a szoftverben, és a probléma okait megszüntessük, ezzel biztosítva az alkalmazás helyes és logikus működését. A tesztelés során több különböző eszközön, és képernyőméreten is kipróbáltuk weboldalunkat, ezzel ellenőrizve az oldal reszponzív megjelenését és hibamentes használatot.

#### 6.2.1 Tesztelési szempontok és eredményük

Az oldal tesztelésénél ezeket az eseteket használtuk, amik az esetleges hibákat felfedhetik:

|                 | Teszteset               | Várt ered-     | Kapott eredmény       | Sike-    |
|-----------------|-------------------------|----------------|-----------------------|----------|
|                 |                         | mény           |                       | res-     |
|                 |                         |                |                       | ség      |
| Regisztráció    | Email: gipszjakab@      | Hibás ema-     | Hibaüzenet:           | ✓        |
|                 |                         | ilcím          | Helytelen formá-      |          |
|                 |                         |                | tumú emailcím!        |          |
| Regisztráció    | Jelszó: Gipszjakab123   | A két jelszó-  | Hibaüzenet:           | <b>√</b> |
|                 | Jelszó újra:            | nak meg kell   | A jelszavak nem       |          |
|                 | Gipszjakab124           | egyeznie       | egyeznek              |          |
| Cég regisztrá-  | Irányítószám: 109       | Hibás          | Hibaüzenet: Hibás     | <b>√</b> |
| ció             |                         | irányítószám   | irányítószám          |          |
| Feladat hozzá-  | Feladat címe: (üres)    | Nincs feladat- | Hibaüzenet: A fel-    | <b>√</b> |
| adás (Felada-   |                         | cím            | adat címe kötelező    |          |
| taid oldal)     |                         |                |                       |          |
| Feladat hozzá-  | Ismétlés: (üres)        | Alapértelme-   | Ismétlés: Egyszeri    | <b>√</b> |
| adás (Felada-   |                         | zett ismétlés  |                       |          |
| taid oldal)     |                         |                |                       |          |
| Feladat hozzá-  | Szín: (üres)            | Alapértelme-   | Szín: Zöld            | <b>√</b> |
| adás (Felada-   |                         | zett szín      |                       |          |
| taid oldal)     |                         |                |                       |          |
| Ügyfél hozzá-   | Név: Barta Júlia        | Sikeres rögzí- | Üzenet: Sikeres rög-  | <b>✓</b> |
| adása (Ügyfe-   | Emailcím:               | tés            | zítés.                |          |
| lek oldal)      | bartajulia34@ugyfel.hu  |                |                       |          |
|                 | Telefonszám:            |                |                       |          |
|                 | +36706665555            |                |                       |          |
| Feladat törlés  | 3 db egy csoportba tar- | Sikeres törlés | Üzenet: Sikeres tör-  | ✓        |
| (csoportos)     | tozó feladat törlése    |                | lés.                  |          |
| Feladat módo-   | Határidő:               | Nem enged ir-  | 0302-re állítja, majd | ✓        |
| sítás           | 2025-04-20→30213-       | reális dátumot | a fókusz a hónapra    |          |
|                 | 04-20                   | beírni         | esik                  |          |
| Statisztika ol- | Mobil nézetben elindít- | A grafikon     | A grafikon vertiká-   | <b>✓</b> |
| dal reszponzi-  | juk az oldalt           | vertikális né- | lis nézetbe ugrik     |          |
| vitása          |                         | zetbe ugrik    |                       |          |

7. A szoftver használatának bemutatása

A felhasználónak először regisztrálnia kell, a saját, illetve a cége adatai megadásával. Beje-

lentkezés után az oldal funkciói elérhetővé válnak. A Feladataid oldalra kerül először a fel-

használó, ami átlátható felületet biztosít a feladatok szűrésére évre, hónapra, esetleg állapot

alapján. A fejlécben más oldalra vezető linkek vannak. Itt lehet eljutni az ügyfelek és alkal-

mazottak kezelésére, valamint a teljesítmény monitorozására alkalmas felületekre.

7.1 Futtatás

7.1.1 Adatbázis:

Az sql alapú adatbázisnak alapértelmezés szerint a 3307-os porton kell futnia. Ezt a backend

web.config fájljában lehet módosítani. Importálja az ADATBÁZIS.sql fájlt.

7.1.2 Frontend:

A frontend alapértelmezetten az 5173-as porton fut, a backend innét fogad kéréseket. A

CORS beállítás megváltoztatásával elérhetjük, hogy más portokról is fogadjon kéréseket.

Nyissa meg a vizsgaremek forráskódjának mappájában talált telepites.bat fájlt a függőségek

telepítéséhez, majd miután lefutott, futtassa indit.bat fájlt a frontend webszerverének elin-

dításához. Írja böngészőjének keresőjébe a konzolon megjelent címet.

Ha a töltés megakad a telepítés közben, a konzolt zárja be, majd indítsa újra.

7.1.3 Backend:

Nyissa meg a vizsgaremek forráskódjának mappájában talált **ugyfelek** mappát, majd az ott

talált ugyfelek.sln-t futtassa. A projekt betöltése után futtassa kívánt böngészőjében a

Swagger-t.

7.2 Bejelentkezés

Egy előre elkészített felhasználó bejelentkezési adatai:

Kovács Viktor:

Email cím: ko

kovacsviktor1@ugyfelnyilvan.hu

Jelszó:

Kvcsvtkr11

Révay Áron:

• Email cím:

revayaron@ugyfelnyilvan.hu

Jelszó:

Revayaron11

27

#### 7.3 Használata

## 7.3.1 Új feladat hozzáadása

- 1. Lépjen a "Feladataid" oldalra a fejlécben található link segítségével.
- 2. Kattintson a feladatok fejlécében található pluszjelre.
- 3. Töltse ki a mezőket, és válassza ki az ismétlés gyakoriságát, majd a feladat színét.
- 4. Kattintson a "Feladat hozzáadása" gombra.
- 5. Sikeres rögzítés esetén egy felugró ablakban jelzi azt a weboldal.

#### 7.3.2 Feladat módosítása

- 1. Kattintson a módosítani kívánt feladatra.
- 2. Írja át a szükséges mezőket.
- 3. Kattintson a "Mentés" gombra.
- 4. Válassza ki, hogy erre vagy az összes ebbe a csoportba tartozó feladatot szeretné módosítani.
- 5. Kattintson a "Mentem" gombra.
- 6. Sikeres módosítás esetén egy felugró ablakban jelzi azt az weboldal.

### 7.3.3 Ügyfél bejegyzése

- 1. Lépjen az "Ügyfelek" oldalra a fejlécben található link segítségével.
- 2. Kattintson az "Ügyfél hozzáadása" melletti pluszjelre.
- 3. Töltse ki a mezőket az ügyfél adataival.
- 4. Kattintson a "Mentés" gombra.

#### 7.3.4 Ügyfél szerkesztése

- 1. Lépjen az "Ügyfelek" oldalra a fejlécben található link segítségével.
- 2. Válassza ki a módosítani kívánt ügyfelet.
- 3. Kattintson az ügyfél kártyáján megtalálható ceruza jelre.
- 4. Kattintson a felugró ablak jobb alsó sarkában lévő ceruza jelre.
- 5. Módosítsa a kívánt adatokat.
- 6. Kattintson a "Mentés" gombra.

### 7.3.5 Ügyfél törlése

- 1. Lépjen az "Ügyfelek" oldalra a fejlécben található link segítségével.
- 2. Válassza ki a törölni kívánt ügyfelet.
- 3. Kattintson az "Ügyfél törlése" melletti gombra.

#### 7.3.6 Alkalmazott bejegyzése

- 1. Lépjen az "Céged" oldalra a fejlécben található link segítségével.
- 2. Kattintson az "Alkalmazott hozzáadása" melletti pluszjelre.
- 3. Töltse ki a mezőket az alkalmazott adataival.

4. Kattintson a "Mentés" gombra.

#### 7.3.7 Alkalmazott szerkesztése

- 1. Lépjen az "Céged" oldalra a fejlécben található link segítségével.
- 2. Válassza ki a módosítani kívánt alkalmazottat.
- 3. Kattintson az alkalmazott kártyáján megtalálható ceruza jelre.
- 4. Kattintson a felugró ablak jobb alsó sarkában lévő ceruza jelre.
- 5. Módosítsa a kívánt adatokat.
- 6. Kattintson a "Mentés" gombra.

#### 7.3.8 Alkalmazott törlése

- 1. Lépjen az "Céged" oldalra a fejlécben található link segítségével.
- 2. Válassza ki a törölni kívánt alkalmazottat.
- 3. Kattintson az "Alkalmazott törlése" melletti gombra.

## 8. Továbbfejlesztési lehetőségek

A szoftverünk jelenlegi változata is támogatja a kis- és magánvállalkozók adminisztratív feladatait, azonban új funkciók bevezetésével a felhasználók munkalendületét többszörösére lendítené fel. Ebben a fejezetben ilyen fejlesztésekre térünk ki, amik a jövőben az alkalmazás használatát megkönnyítik.

## 8.1 Kollaboratív és szerepkör-alapú hozzáférési struktúra

Ezzel a fejlesztés bevezetésével az alkalmazottak és ügyfelek is bejelentkezhetnének a rendszerbe, és szerepkörüknek megfelelő jogosultságokkal hozzáférhetnének a releváns funkciókhoz és adatokhoz, így növelve a cégben való együttműködés hatékonyságát és megsegítve a cég alatt dolgozók munkáját is. A kommunikációt egy szoftverbe épített üzenetküldési lehetőség könnyítené. Az oldal jobb alsó sarkában megjelenne egy fül, amit, ha kinyit a felhasználó, a munkatársaival való beszélgetései megjelennek. Ha üzenet érkezett egy piros kör jelenik meg a kinyitatlan fülön, megnyitásra egyből az üzenetre dobja a felhasználót.

## 8.2 Naptár integráció és szinkronizáció

Weboldalunk jelenleg határidőket kezel, viszont nincs kifejezett naptár nézete. Egy naptár integrációs funkció egyszerűbbé tenné a feladatok átláthatóságát, időbeli vizualizálását és már népszerű alkalmazásokkal való szinkronizálását, például az Outlook vagy a Google Naptár. Ez lehetővé teszi a felhasználó személyes és céges naptárának összehasonlítását, biztosítva azt, hogy két esemény nem keresztezi egymást.

## 8.3 Céged oldal cég névjegyének való nyomtatása

Egy olyan funkció is hasznos lehetne a cégek számára, ha a Céged oldalon található kártyát személyre tudnák szabni (pl. a háttérszínét, betűstílust stb.) és akár ki is tudnák nyomtatni, hogy névjegykártyaként fel tudják használni.

# Tartalom

| 1. | Bevezetés                                     | 1    |   |
|----|---|------|---|
|    | 1.1 Szoftver célja                            | 1    |   |
|    | 1.2 Felhasználói felületek                    | 2    | , |
|    | 1.3 Betűszavak és meghatározások              | 2    | , |
| 2. | Készítési folyamat                            | 3    |   |
|    | 2.1 Feladatok felosztása                      | 3    |   |
|    | 2.2 Az adatbázis megtervezése                 | 4    |   |
|    | 2.2.1 Az ER-modell megtervezése               | 4    |   |
|    | 2.2.2 A relációs modell                       | 6    |   |
|    | 2.3 A felhasználói felület megtervezése       | 7    | , |
|    | 2.3.1 A felhasználói felületek felosztása     | 7    | , |
|    | 2.3.2 A felhasználói felület kinézete         | 7    | , |
| 3. | Komponensek technikai leírása                 | 8    |   |
|    | 3.1 Frontend                                  | 8    |   |
|    | 3.2 Backend                                   | 9    |   |
|    | 3.3 Adatbázis                                 | 9    | 1 |
| 4. | A program felhasználói felületének bemutatása | . 10 | ) |
|    | 4.1 Oldalak                                   | . 10 | ) |
|    | 4.1.1 Főoldal                                 | . 10 | ) |
|    | 4.1.2 Bejelentkezés                           | . 10 | ) |
|    | 4.1.3 Regisztráció                            | . 10 | ) |
|    | 4.1.4 Cég regisztrálása                       | . 11 |   |
|    | 4.1.5 Feladataid                              | . 11 |   |
|    | 4.1.6 Ügyfelek                                | . 11 |   |
|    | 4 1 7 Céged                                   | 12   |   |

|    | 4.1.8 Statisztika                                    | 12 |
|----|--|----|
|    | 4.1.9 Profil   | 12 |
|    | 4.1.10 Cégprofil                                     | 12 |
|    | 4.2 Az oldal funkciói                                | 12 |
|    | 4.2.1 Felugró ablakok                                | 12 |
|    | 4.2.2 Új feladat hozzáadása                          | 14 |
|    | 4.2.3 Feladat módosítása                             | 14 |
|    | 4.2.4 Feladatok ismétlése                            | 15 |
|    | 4.2.5 Ügyfelek és alkalmazottak bejegyzése           | 16 |
|    | 4.2.6 Ügyfelek és alkalmazottak oldalak sajátosságai | 16 |
|    | 4.2.7 Súgóüzenetek                                   | 16 |
|    | 4.2.8 Statisztika oldal működése                     | 17 |
|    | 4.2.9 Felhasználó adatainak módosítása               | 17 |
|    | 4.3 Monitorképek                                     | 18 |
| 5. | A szoftver működésének műszaki feltételei            | 24 |
| 6. | Tesztelés  | 25 |
|    | 6.1 Egységtesztelés (Backend)                        | 25 |
|    | 6.1.1 Példa és magyarázat                            | 25 |
|    | 6.2 Manuális tesztelés (Frontend)                    | 25 |
|    | 6.2.1 Tesztelési szempontok és eredményük            | 25 |
| 7. | A szoftver használatának bemutatása                  | 27 |
|    | 7.1 Futtatás   | 27 |
|    | 7.1.1 Adatbázis:                                     | 27 |
|    | 7.1.2 Frontend:                                      | 27 |
|    | 7.1.3 Backend:                                       | 27 |
|    | 7.2 Bejelentkezés                                    | 27 |
|    | 7.3 Használata                                       | 28 |

|    | 7.3.1 Új feladat hozzáadása                               | 28 |
|----|---|----|
|    | 7.3.2 Feladat módosítása                                  | 28 |
|    | 7.3.3 Ügyfél bejegyzése                                   | 28 |
|    | 7.3.4 Ügyfél szerkesztése                                 | 28 |
|    | 7.3.5 Ügyfél törlése                                      | 28 |
|    | 7.3.6 Alkalmazott bejegyzése                              | 28 |
|    | 7.3.7 Alkalmazott szerkesztése                            | 29 |
|    | 7.3.8 Alkalmazott törlése                                 | 29 |
| 8. | . Továbbfejlesztési lehetőségek                           | 29 |
|    | 8.1 Kollaboratív és szerepkör-alapú hozzáférési struktúra | 29 |
|    | 8.2 Naptár integráció és szinkronizáció                   | 29 |
|    | 8.3 Céged oldal cég néviegyének való nyomtatása           | 30 |