

Budapesti Műszaki Szakképzési Centrum
Bolyai János Műszaki Technikum és Kollégium

DOKUMENTÁCIÓ

SZOFTVERFEJLESZTŐ ÉS -TESZTELŐ SZAKMA

vizsgaremek készítéséhez

A fejlesztő csapat tagjai: Ceglédi Zente Holló és Lénárt Dániel Péter

A dolgozat címe: **FoxHabits**

A dolgozat hivatkozása: <https://foxhabits.com>

A belső konzulens neve: Nagy Ferenc

A feladat leírása: Webalapú alkalmazás, melynek segítségével a felhasználó követheti a mindennapi tevékenységeit, hasznos és ártó szokásait. Az alkalmazás statisztikákon keresztül szemlélteti a felhasználónak a fejlődését.

A dokumentum lezárásának dátuma: 2025. április 14.

Tartalom

Tartalom.....	1
1.Projekt Specifikáció	4
1.1. Projekt rövid leírása	4
1.2. Fejlesztés során felhasznált technológiák	4
1.2.1. Szoftvertechnológiák.....	4
1.2.2. Konkrét szoftverek.....	4
1.3. Szerepkörök	5
1.3.1. Felhasználó	5
1.3.2. Moderátor.....	5
1.3.3. Adminisztrátor.....	5
1.4. Főbb funkciók.....	5
1.4.1. Felhasználók számára	5
1.4.2. Moderátorok számára.....	5
1.4.3. Adminisztrátorok számára	6
1.5 Célok és célkitűzések.....	6
1.5.1. Mit szeretnénk elérni a szoftverrel?	6
1.5.2. Rövid távú célok	6
1.5.3. Hosszú távú célok	7
1.6 Követelmények.....	7
1.6.1. Funkcionális követelmények	7
1.6.2. Nem-funkcionális követelmények	7
1.7. Alkalmazási területek.....	7
1.8. Fejlesztési módszertan.....	8
1.9. Csapat tagjainak szerepkörei.....	8
1.10. Ütemterv	9
2. Adatbázis Dokumentáció.....	10
2.1. Bevezetés	10
2.2. Általános információk	10
2.3. Adatbázis Szerkezet	10
2.3.0. <i>auth.users</i> tábla (a Supabase biztosítja)	10

2.3.1. <i>entries</i> tábla	10
2.3.2. <i>habits</i> tábla	11
2.3.3. <i>habit_names</i> tábla	13
2.3.4. <i>checklists</i> tábla	14
2.3.5. <i>settings</i> tábla.....	15
2.4. Adatbiztonság.....	16
3.1 Általános információk	17
3.2 API-ok.....	17
3.2.1. GET /api/habits ~ Szokások lekérése	17
3.2.2. POST /api/habits ~ Új szokás hozzáadása a rendszerhez	18
3.2.3. GET /api/habits/own ~ Saját szokások lekérése	19
3.2.5. GET /api/habits/[id] ~ A szokás adatainak lekérdezése	21
3.2.6. PATCH /api/habits/[id] ~ Szokás módosítása	22
3.2.7. DELETE /api/habits/[id] ~ A szokás törlése.....	24
3.2.8. GET /api/entries/habit/[id] ~ Egy szokáshoz tartozó bejegyzések lekérdezése	25
3.2.9. POST /api/entries/habit/[id] ~ Egy szokás logolása	26
3.2.10. DELETE /api/entries/[id] ~ Egy bejegyzés törlése.....	28
3.2.11. GET /api/habits/export	28
3.2.12. GET /api/checklists ~ Saját gyorslisták lekérése	29
3.2.13. POST /api/checklists ~ Gyorslista hozzáadása.....	30
3.2.14. GET /api/checklists/[id] ~ Egy gyorslista lekérése	30
3.2.15. PATCH /api/checklists/[id] ~ Egy gyorslista frissítése	31
3.2.16. DELETE /api/checklists/[id] ~ Egy gyorslista törlése.....	33
3.2.17. GET /api/habit-names ~ Lekéri a jóváhagyott szokás neveket.....	33
4. Manuális tesztelési dokumentáció	35
4.1. Cél meghatározása	35
4.2. Tesztterv	35
4.3.1. 1. Tesztelés	36
4.3.2. 2. Tesztelés	39
4.4. Hibák nyomon követése.....	44

4.5. Teszt összegzés	45
5. Automata tesztelési dokumentáció	46
5.1. Cél meghatározása	46
5.2. Tesztterv	46
5.3. Automata tesztek.....	47
5.4. Hibák nyomon követése.....	55
5.5. Teszt összegzés.....	55
6. Felhasználói dokumentáció	56
6.1. Bevezetés	56
6.2. Regisztráció és bejelentkezés	56
6.3. Szokások kezelése	57
6.4. Listák.....	60
6.5. Statisztika.....	61
6.6. Beállítások	61
7. Összegzés.....	63
7.1. Kihívások	63
7.2. Fejlődés.....	63
7.3. Jövőbeli tervek.....	63
7.4. Költségek.....	64
7.5. Forrásmegjelölés.....	64

1. Projekt Specifikáció

1.1. Projekt rövid leírása

A projekt célja egy online szokáskövető platform fejlesztése, amely lehetővé teszi felhasználói számára a napi szokásaik követését, legyen ez káros vagy egészséges vagy akár csak egy egyszerű napi teendő.

1.2. Fejlesztés során felhasznált technológiák

1.2.1. Szoftvertechnológiák

- **Front – és backend:** Next.js – modern, komponens React alapú keretrendszer.
- **Adatbázis:** Supabase – PostgreSQL relációs adatbázis, a gyors és biztonságos RLS a lekérdezésekhez és nagy adatmennyiségek kezeléséhez, jó skálázhatóság miatt.
- **Autentikáció:** Supabase Auth – JWT token alapú – adatbázissal integrált hitelesítés.
- **Hosting:** Vercel - zökkenőmentes CI/CD deployment Github Actions segítségével.
- **Domain bérlés:** Namecheap – Legnépszerűbb választás nemzetközi domain bérlés esetén
- **E-mailek kezelése:** Resend – Jól skálázható és integrálható SMTP szolgáltatás.

1.2.2. Konkrét szoftverek

- **IDE:** Visual Studio Code – könnyen testreszabható fejlesztői környezet, remek bővítményekkel, a Next.js-sel történő fejlesztés elősegítésére.

- **Verziókezelés:** GitHub – kódtár, dokumentumok kezelése, deployment, verziók és hibák nyomon követése.
- **Tesztelés:** Postman (API tesztekhez), Vitest (automata tesztekhez)
- **Kommunikáció:** Discord, Messenger, Teams, Telefon.
- **Tervezés és nyomon követés:** Trello, GitHub Issues, Notion

1.3. Szerepkörök

1.3.1. Felhasználó

- **Funkciók:** Regisztráció, szokások követése, törlése, létrehozása, módosítása, követése, egyszerű statisztikák megtekintése.

1.3.2. Moderátor

- **Funkciók:** Új szokás nevek áttekintése, elfogadása.

1.3.3. Adminisztrátor

- **Funkciók:** Felhasználók kezelése, portál szintű statisztikák elemzése.

1.4. Főbb funkciók

1.4.1. Felhasználók számára

- Szokások nyomon követése, naplózása.
- Szokások történetének megtekintése.
- Gyorslista (egyszerű To Do) használata és kezelése.
- Összesített statisztikák megtekintése a teljes tevékenységükről.

1.4.2. Moderátorok számára

- Minden felhasználó szintű funkció.

- Elfogadott szokásnevek megtekintése.
- Szokásnevek kérelmének elbírálása.

1.4.3. Adminisztrátorok számára

- Minden korábbi szintű funkció.
- Felhasználók és jogosultságaik kezelése.
- Felhasználók kitiltása az oldalról.
- Portálszintű statisztikák megtekintése

1.4.4. Jogosultságkezelés

- A látogatók nem érik el azokat az oldalakat, amelyek esetében bejelentkezés szükséges.
- A felhasználó nem hozhat létre publikus szokást, a Moderátor jóváhagyása nélkül.
- A Moderátor és az Adminisztrátor látja a portálon található összes szokás adatait, valamint azt, hogy azok kikhez tartoznak.
- A felhasználók nem látják egymás szokásainak adatait, továbbá nem látják, hogy a publikus szokásokat ki hozta létre, és ki hagyta jóvá.

1.5 Célok és célkitűzések

1.5.1. Mit szeretnénk elérni a szoftverrel?

- A cél egy felhasználóbarát platform, ahol bárki egyszerűen kezelheti a napi, heti és havi szokásait, teendőit, bevásárlólistáját.

1.5.2. Rövid távú célok

- Egy MVP (Minimum Viable Product) fejlesztése, amely elég funkcióval rendelkezik ahhoz, hogy a korai ügyfelek számára használható legyen,

akik így olyan visszajelzést adhatnak, amelyek felhasználhatóak a jövőbeli fejlesztéshez.

1.5.3. Hosszú távú célok

- Egy kimagasló minőségű szolgáltatás, ami a mindennapjaink hasznos eszközévé tud válni.
- Extra funkciók fejlesztése és bevezetése, például különböző mesterséges intelligencia alapú eszközök, telefonos applikáció, szokás megosztás.

1.6 Követelmények

1.6.1. Funkcionális követelmények

1. Felhasználók regisztrációja és hitelesítése.
2. Szokások létrehozása és naplózása.
3. Szokásnaplók megjelenítése és statisztikák.

1.6.2. Nem-funkcionális követelmények

4. Teljesítmény: A rendszernek képesnek kell lennie egyszerre 300 felhasználót kiszolgálni.
5. Biztonság: Adatok, műveletek titkosítása (pl. HTTPS, SSL, RLS, Secure Action IDs).
6. Skálázhatóság: A rendszernek könnyen bővíthetőnek kell lennie a felhőszolgáltatások bővítésével (Vercel, Supabase).

1.7. Alkalmazási területek

A webalkalmazásunk szinte bárkinak egy hasznos eszköz lehet a mindennapjaiban, mivel mindenkinek vannak szokásai és feladatai. Az egyedi szokások hozzáadása, egyedi ismétlődés-intervallum beállítása jól testre

szabhatóvá teszi a platformunkat. A modern és letisztult felhasználói felület által bármilyen korosztály számára egyszerűen kiismerhető és alkalmazható a rendszerünk.

1.8. Fejlesztési módszertan

Az agilis módszertant használjuk, amely lehetővé teszi az iteratív és rugalmas fejlesztési folyamatokat. Az egyes iterációk (sprintek) során rendszeresen átbeszéltük az elkészült részeket a konzulenssel, ezek alapján finomhangoltuk a fejlesztést a következő szakaszokban.

- Iterációkban történő fejlesztés (2 hetes sprintek).
- Folyamatos visszacsatolás és finomhangolás.

1.9. Csapat tagjainak szerepkörei

- **Frontend:** Lénárt Dániel Péter és Ceglédi Zente Holló
- **Backend:** Ceglédi Zente Holló és Lénárt Dániel Péter
- **Adatbázis és SMTP:** Ceglédi Zente Holló és Lénárt Dániel Péter
- **Tesztelés:** Ceglédi Zente Holló
- **Dokumentálás:** Ceglédi Zente Holló és Lénárt Dániel Péter

1.10. Ütemterv

1. Kezdeti tervezés (1. hónap):

- Részletes specifikációk összeállítása, fejlesztési környezet kialakítása, alkalmazandó technológiai eszközök, keretrendszerek megválasztása, alkalmazás nevének kiválasztása, domain megválasztása.

2. Első prototípus (2-3. hónap):

- Felhasználói felület és alapfunkciók fejlesztése, manuális tesztek futtatása, adatbázis séma létrehozása teszt adatokkal, automatikus deploy beállítása, hosting konfigurálása.

3. Béta verzió (4-5. hónap):

- Teljes funkcionalitás implementálása, automata tesztek integrálása.

4. Élesítés (6. hónap):

- Teljes oldalas manuális tesztek végzése a hibák feltárásának érdekében, hibák javítása.

2. Adatbázis Dokumentáció

2.1. Bevezetés

Ez a dokumentáció a felhasználói regisztrációhoz használt adatbázis szerkezetét és működését ismerteti. Az adatbázis célja a felhasználók adatainak biztonságos tárolása és kezelése.

2.2. Általános információk

- **Adatbázis neve:** FoxHabits
- **Adatbázis-kezelő rendszer (DBMS):** PostgreSQL
- **Verzió:** 15.6.1.139
- **Készítők:** Ceglédi Zente Holló, Lénárt Dániel Péter
- **Dátum:** 2025. február 12.
- **Host:** [Supabase](#)
- **Tervezés:** [DBDiagram](#)
- **Séma modell:** [Prisma](#)

2.3. Adatbázis Szerkezet

2.3.0. *auth.users* tábla (a Supabase biztosítja)

2.3.1. *entries* tábla

Leírás: Az entries tábla a szokások bejegyzéseit tárolja, időponttal és típussal ellátva, valamint a hozzá tartozó szokás azonosítóval.

Mező neve	Típusa	Leírás	Korlátozások
entry_id	INT	A bejegyzés egyedi azonosítója	PRIMARY KEY
time_of_entry	TIMESTAMP	A bejegyzés időpontja	NOT NULL

entry_type	entry_type	A bejegyzés típusa (done/skip)	NOT NULL
habit_id	INT	A kapcsolódó szokás azonosítója	FOREIGN KEY (habits.habit_id), NOT NULL
datetime	TIMESTAMP	A bejegyzés dátuma és ideje	NOT NULL

Példa a tábla létrehozására:

```
model entries {
  entry_id      BigInt      @id @unique @default(autoincrement())
  habit_id      BigInt
  datetime      DateTime    @db.Timestamp(6)
  time_of_entry DateTime    @default(now()) @db.Timestamptz(6)
  entry_type    entry_type @default(done)
  habits        habits      @relation(fields: [habit_id], references:
[habit_id], onDelete: Cascade, onUpdate: NoAction)

  @@schema("public")
}
```

2.3.2. habits tábla

Leírás: A habits tábla a felhasználók szokásainak adatait tárolja, beleértve a szokás nevét, típusát, státuszát és időbeli jellemzőit.

Mező neve	Típusa	Leírás	Korlátozások
habit_id	INT	A szokás egyedi azonosítója	PRIMARY KEY
habit_name_id	INT	A szokás nevének azonosítója	REFERENCES habit_names(habit_name_id)
habit_interval	INT	Két elvégzés közötti intervallum	NOT NULL

habit_internal_type	habit_internal_type	Két elvégzés közötti intervallum típusa (days/weeks/months/years)	NOT NULL
start_date	TIMESTAMP WITH TIME ZONE	A szokás kezdő dátuma	NOT NULL
is_active	BOOLEAN	Aktív-e a szokás	DEFAULT TRUE
created_date	TIMESTAMP WITH TIME ZONE	Létrehozás időpontja	NOT NULL
related_user_id	UUID	A kapcsolódó felhasználó azonosítója	REFERENCES auth.users(id)
habit_type	habit_type	A szokás típusa (normal_habit/ bad_habit)	NOT NULL

Példa a tábla létrehozására:

```

model habits {
  habit_id          BigInt          @id @unique @default(autoincrement())
  related_user_id   String           @db.Uuid
  habit_type        habit_type
  interval          Int              @db.SmallInt
  habit_interval_type interval_type @default(days)
  start_date        DateTime         @default(now()) @db.Timestamptz(6)
  is_active         Boolean          @default(false)
  created_date      DateTime         @default(now()) @db.Timestamptz(6)
  habit_name_id     BigInt
  entries           entries[]
  habit_names       habit_names     @relation(fields: [habit_name_id],
references: [habit_name_id])
  users             users           @relation(fields: [related_user_id],
references: [id], onDelete: Cascade, onUpdate: NoAction)

  @@schema("public")
}

```

2.3.3. habit_names tábla

Leírás: A habit_names tábla a szokások neveit és megjelenítési beállításait tárolja, valamint az első beküldő azonosítóját.

Mező neve	Típusa	Leírás	Korlátozások
habit_name_od	INT	A szokásnév egyedi azonosítója	PRIMARY KEY
habit_name	TEXT	A szokás megnevezése	NOT NULL
habit_name_status	habit_name_status	A szokásnév státusza (new/approved/ rejected/private)	NOT NULL
created_at	TIMESTAMP WITH TIME ZONE	Létrehozás időpontja	-
sender_user_id	UUID	Felhasználó azonosítója	REFERENCES auth.users(id)

Példa a tábla létrehozására:

```
model habit_names {  
  habit_name      String           @unique @db.VarChar  
  habit_name_status habit_names_status  
  sender_user_id  String?          @db.Uuid  
  created_at      DateTime?        @db.Timestamptz(6)  
  habit_name_id   BigInt           @id @unique  
  @default(autoincrement())  
  users           users?           @relation(fields: [sender_user_id],  
references: [id], onUpdate: NoAction)  
  habits          habits[]  
  
  @@schema("public")  
}
```

2.3.4. checklists tábla

Leírás: A checklists tábla a felhasználók által létrehozott gyorslistákat és azok elemeit tárolja.

Mező neve	Típusa	Leírás	Korlátozások
id	INT	A lista egyedi azonosítója	PRIMARY KEY
name	TEXT	A gyorslista neve	-
elements	JSON	A gyorslista elemei	-
created_at	TIMESTAMP WITH TIME ZONE	Létrehozás időpontja	NOT NULL
updated_at	TIMESTAMP WITH TIME ZONE	Frissítés időpontja	-
user_id	UUID	Felhasználó azonosítója	REFERENCES auth.users(id)

Példa a tábla létrehozására:

```
model checklists {
  id          BigInt    @id(map: "checklist_pkey")
  @default(autoincrement())
  user_id     String?   @db.Uuid
  name        String?
  elements    Json?     @db.Json
  created_at  DateTime  @default(now()) @db.Timestamptz(6)
  updated_at  DateTime? @default(now()) @db.Timestamptz(6)
  users       users?    @relation(fields: [user_id], references: [id],
onDelete: Cascade, onUpdate: NoAction, map: "checklist_user_id_fkey")

  @@schema("public")
}
```

2.3.5. settings tábla

Leírás: A settings tábla a felhasználók alkalmazásbeállításait tárolja, beleértve a megjelenítési és értesítési preferenciákat.

Mező neve	Típusa	Leírás	Korlátozások
user_id	UUID	A felhasználó egyedi azonosítója (referencia az auth.users táblára)	PRIMARY KEY
dark_mode	dark_mode	Sötét mód beállítása (dark/light/system)	NOT NULL
localization	localization	Lokalizációs beállítások (en/hu)	--
email_notifications	BOOLEAN	Email értesítések engedélyezése	DEFAULT FALSE

Példa a tábla létrehozására:

```
model settings {
  user_id      String      @id @unique
  @default(dbgenerated("auth.uid()")) @db.Uuid
  dark_mode    dark_mode   @default(system)
  localization localization? @default(hu)
  email_notifications Boolean @default(true)
  users        users       @relation(fields: [user_id],
references: [id], onDelete: Cascade, onUpdate: NoAction)

  @@schema("public")
}
```


2.4. Adatbiztonság

- **Jelszó hashelés:** A jelszavakat a Supabase segítségével bcrypt algoritmussal hasheljük, amely biztonságos és széles körben elfogadott módszer a jelszavak tárolására.
- **Adatvédelem:** Az e-mail címek és felhasználónevek egyedi korlátozása megakadályozza a duplikált adatok tárolását, csökkentve ezzel az adatbázis inkonzisztenciájának kockázatát.
- **E-mail-es megerősítés:** A regisztráció során a felhasználónak meg kell erősítenie az e-mail címét.

3. REST API Dokumentáció

A projekt megvalósítása során egyaránt alkalmaztunk RestAPI megoldásokat, és actions fájlokat is. Azokat a funkciókat, amelyeket a felhasználók elérnek RestAPI kapcsolaton keresztül valósítottuk meg.

3.1 Általános információk

- **Bázis URL:** <https://foxhabits.com/>
- **API lekérések útvonala:** <https://foxhabits.com/api/>
- **Protokoll:** HTTPS
- **Formátum:** JSON
- **Autentikáció:** Bearer Token (JWT)

3.2 API-ok

3.2.1. GET /api/habits ~ Szokások lekérése

Hibakódok

Hibakód	Leírás
200	Sikeres kérés
201	Erőforrás létrehozva
400	Helytelen kérés
401	Jogosultság hiánya
404	Nem található
500	Szerverhiba

Várható bemenetek: **Header:** Authorization: Bearer {token}

Válasz formátuma:

```
[
  {
    "habits": [
      {
        "id": "string",
        "name": "string",
        "description": "string"
      }
    ]
  }
]
```

Hibakezelés:

- **401:** Hitelesítési hiba
- **500:** Szerverhiba

3.2.2. POST /api/habits ~ Új szokás hozzáadása a rendszerhez

Hibakódok

Hibakód	Leírás
200	Sikeres kérés
201	Erőforrás létrehozva
400	Helytelen kérés
401	Jogosultság hiánya
404	Nem található
500	Szerverhiba

Várható bemenetek:

- **Header:** Authorization: Bearer {token}
- **Body:**

```
{
  "habit_names": "string (1-255 karakter)",
  "habit_name_status": "new | pending | approved | ...",
  "habit_type": "normal_habit | bad_habit",
  "interval": number,
  "habit_interval_type": "hours | days | weeks | months | years",
  "start_date": "string (ISO dátum)",
  "is_active": true
}
```

Válasz:

```
{
  "habit_id": "string"
}
```

Hibakezelés:

- **400:** Hibás vagy hiányos bemenet
- **401:** Hitelesítési hiba
- **500:** Szerverhiba

3.2.3. GET /api/habits/own ~ Saját szokások lekérése

Hibakódok

Hibakód	Leírás
200	Sikeres kérés
401	Jogosultság hiánya
500	Szerverhiba

Várható bemenetek:

- **Header:** Authorization: Bearer {token}

Válasz:

```
[
  {
    "habit_id": "string",
    "habit_type": "normal_habit | bad_habit",
    "interval": number,
    "habit_interval_type": "hours | days | weeks | months | years",
    "start_date": "string (ISO dátum)",
    "is_active": boolean,
    "habit_names": {
      "habit_name": "string"
    },
    "created_date": "string (ISO dátum)",
    "entries": [
      {
        "entry_id": "string",
        "time_of_entry": "string",
        "entry_type": "string",
        "datetime": "string (ISO dátum)"
      }
    ]
  }
]
```

vagy

```
{
  "message": "Nincs egyetlen szokásod sem."
}
```

Hibakezelés:

- **401:** Hiányzó Authorization fejléc esetén: "Hiányzó autentikációs token"
- **401:** Érvénytelen vagy lejárt token esetén: "Érvénytelen vagy lejárt token"
- **500:** Bármilyen szerveroldali vagy adatbázis-hiba esetén: "Szerver hiba történt"

3.2.5. GET /api/habits/[id] ~ A szokás adatainak lekérdezése

Hibakódok

Hibakód	Leírás
200	Sikeres kérés
400	Hiányzó vagy érvénytelen szokás azonosító
401	Nincs bejelentkezve a felhasználó
404	A szokás nem található, vagy a felhasználónak nincs jogosultsága megtekinteni
500	Szerverhiba

Várható bemenetek:

- **URL paraméter:** habit_id: a lekérdezni kívánt szokás azonosítója (szám)
- **Header:** Authorization: Bearer {token}

Válasz:

```
{
  "data": {
    "habit_id": "number",
    "related_user_id": "string",
    "habit_type": "normal_habit | bad_habit",
    "interval": "number",
    "habit_interval_type": "hours | days | weeks | months | years",
    "start_date": "string (ISO dátum)",
    "is_active": "boolean",
    "habit_names": {
      "habit_name": "string"
    },
  },
  "created_date": "string (ISO dátum)",
  "entries": [
    {
      "entry_id": "string",
      "time_of_entry": "string",
      "entry_type": "string",
      "datetime": "string (ISO dátum)"
    }
  ]
}
```

```

    }
  ]
}
}

```

Hibakezelés:

- **400:**
 - Ha nincs habit_id az URL-ben: "A szokás azonosítója nem található!"
 - Ha a habit_id nem szám: "A szokás azonosítónak számnak kell lennie!"
- **401:** Be nem jelentkezett felhasználó esetén: "A felhasználó nem található!
Kérlek jelentkezz be."
- **404:** Ha nem található ilyen habit_id, vagy nem a felhasználó saját szokása:
"A szokás nem létezik, vagy nincs jogosultságod a megtekintéséhez."
- **500:** Bármilyen szerveroldali vagy adatbázis-hiba esetén: "Szerver hiba történt"

3.2.6. PATCH /api/habits/[id] ~ Szokás módosítása

Hibakódok

Hibakód	Leírás
200	Sikeres módosítás, visszatér a frissített objektummal
400	Hibás kérés (pl. érvénytelen vagy hiányzó adat, nincs mit frissíteni)
401	Nincs bejelentkezve a felhasználó
403	Nincs jogosultság a szokás módosításához (nem saját, nem admin)
500	Szerverhiba vagy adatbázis hiba

Várható bemenetek:

- **URL paraméter:** habit_id: A frissítendő szokás azonosítója (szám)

- **Header:** Authorization: Bearer {token}

Request body:

```
{
  "habit_name": "string (1-255 karakter, opcionális)",
  "habit_name_status": "new | private | approved | rejected
(opcionális)",
  "habit_type": "normal_habit | bad_habit (opcionális)",
  "interval": number (pozitív, opcionális),
  "habit_interval_type": "hours | days | weeks | months | years
(opcionális)",
  "start_date": "string (ISO dátum, opcionális)",
  "is_active": boolean (opcionális),
  "related_user_id": "string (opcionális - csak admin módosíthatja
másra)"
}
```

Válasz formátuma:

```
{
  "data": {
    "habit_id": number,
    "habit_type": "normal_habit | bad_habit",
    "interval": number,
    "habit_interval_type": "hours | days | weeks | months | years",
    "start_date": "ISO dátum string",
    "is_active": boolean,
    "related_user_id": "string",
    "habit_name_id": number,
    ...
  }
}
```

Hibakezelés:

- **400:**
 - "A szokás azonosítónak számnak kell lennie!"
 - "Nincs frissítendő mező."

- "A szokás nevének státusza csak new, private lehet." (ha nem moderátor vagy admin felhasználó próbálja módosítani)
- **401:** Be nem jelentkezett felhasználó esetén: "A felhasználó nem található!
Kérlek jelentkezz be."
- **403:** "Nincs jogosultságod a szokás módosításához."
- **500:** "Szerverhiba"

3.2.7. DELETE /api/habits/[id] ~ A szokás törlése

Hibakódok

Hibakód	Leírás
200	Sikeres törlés
400	Hibás kérés - hiányzik vagy nem szám a habit_id
401	Nincs bejelentkezve a felhasználó
404	A szokás nem létezik, vagy nincs jogosultság a törléshez

Várható bemenetek:

- **URL paraméter:** habit_id: A törlendő szokás azonosítója (szám)
- **Header:** Authorization: Bearer {token}

Válasz formátuma:

```
{
  "message": "Sikeres törlés!"
}
```

Hibakezelés:

- **400:**
 - "A szokás azonosítónak számnak kell lennie!"
 - "Érvénytelen szokás azonosító! "

- **401:** Be nem jelentkezett felhasználó esetén: "A felhasználó nem található!
Kérlek jelentkezz be."
- **404:** "A szokás nem létezik, vagy nincs jogosultságod a törléshez."

3.2.8. GET /api/entries/habit/[id] ~ Egy szokáshoz tartozó bejegyzések lekérdezése

Lekéri az adott szokáshoz tartozó bejegyzéseket.

Admin esetén: minden bejegyzést visszaad a megadott szokáshoz.

Nem admin felhasználónál: csak akkor ad vissza bejegyzéseket, ha a szokás az adott felhasználóhoz tartozik.

Hibakódok

Hibakód	Leírás
200	Sikeres lekérdezés, a visszatérő adat a szokáshoz tartozó bejegyzések tömbje
400	Hiányzik vagy nem szám a habit_id
401	Nincs bejelentkezve a felhasználó
403	A szokás nem a felhasználóhoz tartozik (ha nem admin)
500	Szerverhiba vagy adatbázis hiba

Várható bemenetek:

- **URL paraméter:** habit_id: A szokás azonosítója (szám)
- **Header:** Authorization: Bearer {token}

Válasz formátuma:

```
[
  {
    "entry_id": 1,
    "habit_id": 123,
    "datetime": "2025-04-14T12:34:56.789Z",
    "time_of_entry": "2025-04-14T12:34:56.789Z",
```

```

    "entry_type": "done" | "skipped"
  },
  ...
]

```

Hibakezelés:

- **400:**
 - "A szokás azonosítója nem található!"
 - "A szokás azonosítónak számnak kell lennie!"
- **401:** "A felhasználó nem található! Kérlek jelentkezz be."
- **403:** "A szokás nem található vagy nem tartozik hozzád!"
- **500:**
 - "Hiba történt a szokás lekérdezése közben."
 - "Hiba történt az admin lekérdezés során."

3.2.9. POST /api/entries/habit/[id] ~ Egy szokás logolása

Új bejegyzés logolása egy szokáshoz ("done" vagy "skipped").

Csak akkor lehet bejegyzést logolni, ha a szokás valóban az adott felhasználóhoz tartozik (vagy admin az illető).

Hibakódok

Hibakód	Leírás
200	Sikeres hozzáadás
400	Hiányzó vagy hibás szokás ID, vagy a beküldött adatok hibásak
401	Nincs bejelentkezve a felhasználó
403	A megadott szokás nem tartozik az adott felhasználóhoz (ha nem admin)
500	Szerverhiba vagy adatbázis hiba

Várható bemenetek:

- **URL paraméter:** habit_id: A szokás azonosítója (szám)

- **Header:** Authorization: Bearer {token}

Request body:

```
{
  "time_of_entry": "2025-04-14T08:00:00.000Z",
  "datetime": "2025-04-14T08:00:00.000Z", // opcionális, default: now
  "entry_type": "done" // vagy "skipped", default: "done"
}
```

Válasz formátuma:

```
{
  "data": [
    {
      "entry_id": 123,
      "habit_id": 456,
      "time_of_entry": "2025-04-14T08:00:00.000Z",
      "datetime": "2025-04-14T08:00:00.000Z",
      "entry_type": "done"
    }
  ],
  "success": true
}
```

Hibakezelés:

- **400:**
 - "A szokás azonosítója nem található!"
 - "A szokás azonosítónak számnak kell lennie!"
- **401:** "A felhasználó nem található! Kérlek jelentkezz be."
- **403:** "A szokás nem található vagy nem tartozik hozzád!"
- **500:**
 - "Hiba történt a bejegyzés mentése közben."
 - "Hiba történt a bejegyzés feldolgozása közben."

3.2.10. DELETE /api/entries/[id] ~ Egy bejegyzés törlése

Csak az admin vagy az adott szokás tulajdonosa törölhet bejegyzést.

Hibakódok

Hibakód	Leírás
200	Sikeres törlés
403	Nincs jogosultság a bejegyzés törléséhez

Várható bemenetek:

- **URL paraméter:** entry_id: A törlendő bejegyzés azonosítója (szám)
- **Header:** Authorization: Bearer {token}

Válasz formátuma:

```
{
  "success": true
}
```

Hibakezelés:

- **403:** Ha nincs a felhasználónak jogosultsága a bejegyzés törlésére, vagy a bejegyzés nem található.

3.2.11. GET /api/habits/export

A bejelentkezett felhasználó összes szokásának exportálása CSV fájlban.

Hibakódok

Hibakód	Leírás
200	Sikeres exportálás, válaszként CSV fájl
401	Nem vagy bejelentkezve
500	Hiba történt a szokások lekérdezése vagy exportálása közben

Várható bemenetek:

- **Header:** Authorization: Bearer {token}

Példa CSV kimenet:

```
habit_id,name,description,related_user_id,created_at
1,"Vízivás","Napi 2 liter","d7c9...","2025-04-10T10:00:00Z"
2,"Olvasás","Min. 10 oldal","d7c9...","2025-04-11T08:30:00Z"
```

3.2.12. GET /api/checklists ~ Saját gyorslisták lekérése

Hibakódok

Hibakód	Leírás
200	Sikeres lekérdezés
401	Nincs bejelentkezve a felhasználó
500	Hiba történt az adatok lekérdezése során

Várható bemenetek:

- **Header:** Authorization: Bearer {token}

Válasz formátuma:

```
[
  {
    "id": 1,
    "user_id": "d7c9...",
    "name": "Utazás előtti teendők",
    "elements":
    { "Alma": "CHECKED", "Krumpli": "CHECKED", "Kolbász": "CHECKED" },
    "created_at": "2025-04-12T09:00:00Z",
    "updated_at": "2025-04-14T14:30:00Z"
  },
  ...
]
```

3.2.13. POST /api/checklists ~ Gyorslista hozzáadása

Hibakódok

Hibakód	Leírás
200	Sikeres checklist létrehozás
400	Érvénytelen bemeneti adat
401	Nem vagy bejelentkezve
500	Szerverhiba vagy adatbázis hiba

Várható bemenetek:

- **Header:** Authorization: Bearer {token}

Request body:

```
{
  "name": "Hétfégi bevásárlás",
  "elements": []
}
```

Válasz formátuma:

```
{
  "id": 5,
  "user_id": "d7c9...",
  "name": "Hétfégi bevásárlás",
  "elements": [],
  "created_at": "2025-04-14T12:34:56.789Z",
  "updated_at": "2025-04-14T12:34:56.789Z"
}
```

3.2.14. GET /api/checklists/[id] ~ Egy gyorslista lekérése

Hibakódok

Hibakód	Leírás
200	Sikeres lekérés
400	Hiányzó vagy érvénytelen azonosító
401	Nincs bejelentkezve a felhasználó
403	Nincs jogosultság az adott checklisthez
404	A checklist nem található
500	Szerverhiba vagy adatbázis hiba

Várható bemenetek:

- **URL paraméter:** id: A checklist azonosítója (egész szám)
- **Header:** Authorization: Bearer {token}

Válasz formátuma:

```
{
  "data": {
    "user_id": "d7c9...",
    "name": "Hétfégi bevásárlás",
    "elements": [
      { "description": "Tej", "status": "CHECKED" },
      { "description": "Tojás", "status": "UNCHECKED" }
    ],
    "created_at": "2025-04-14T12:34:56.789Z",
    "updated_at": "2025-04-14T12:34:56.789Z"
  }
}
```

3.2.15. PATCH /api/checklists/[id] ~ Egy gyorslista frissítése

Hibakódok

Hibakód	Leírás
200	Sikeres frissítés

Hibakód	Leírás
400	Hibás kérés, nincs módosítandó adat, vagy validációs hiba
401	Nincs bejelentkezve a felhasználó
403	Nincs jogosultság a módosításhoz
500	Szerverhiba vagy adatbázis hiba

Várható bemenetek:

- **URL paraméter:** id: A checklist azonosítója (egész szám)
- **Header:** Authorization: Bearer {token}

Request body:

```
{
  "id": 5,
  "user_id": "d7c9...",
  "name": "Frissített bevásárlás",
  "elements": {
    "0": "UNCHECKED",
    "1": "CHECKED"
  },
  "updated_at": "2025-04-14T13:00:00.000Z"
}
```

Válasz formátuma:

```
{
  "id": 5,
  "user_id": "d7c9...",
  "name": "Frissített bevásárlás",
  "elements": {
    "0": "UNCHECKED",
    "1": "CHECKED"
  },
  "updated_at": "2025-04-14T13:00:00.000Z"
}
```

3.2.16. DELETE /api/checklists/[id] ~ Egy gyorslista törlése

Hibakódok

Hibakód	Leírás
200	Sikeres törlés
400	Hibás vagy hiányzó checklist ID
401	Nincs bejelentkezve a felhasználó
500	Szerverhiba vagy adatbázis hiba

Várható bemenetek:

- **URL paraméter:** id: A checklist azonosítója (egész szám)
- **Header:** Authorization: Bearer {token}

Válasz formátuma:

```
{ "success": true }
```

3.2.17. GET /api/habit-names ~ Lekéri a jóváhagyott szokás neveket

Hibakódok

Hibakód	Leírás
200	Sikeres lekérés
401	A felhasználó nincs bejelentkezve
500	Szerverhiba vagy adatbázis hiba

Várható bemenetek:

- **Header:** Authorization: Bearer {token}

Válasz formátuma:

```
[  
  { "habit_name": "Edzés" },  
  { "habit_name": "Fogmosás" },  
  { "habit_name": "Olvasás" }  
]
```

4. Manuális tesztelési dokumentáció

4.1. Cél meghatározása

Ez a dokumentáció a manuális tesztelési folyamatokat foglalja össze. Célja, hogy a tesztesetek és eredmények egyértelműen nyomon követhetők és reprodukálhatók legyenek.

4.2. Tesztterv

- **Tesztelési cél:** A webalkalmazás alapvető autentikációs funkcióinak manuális tesztelése különböző operációs rendszerekkel és különböző böngészőkkel.
- **Tesztelt modulok/funkciók:**
 - Bejelentkezés
 - Regisztráció
 - Elfelejtett jelszó
- **Tesztkörnyezet:**
 - **Operációs rendszerek:**
 - Mac OS 15.1
 - Microsoft Windows [Version 10.0.19045.5131]
 - **Böngészők:**
 - Google Chrome: 131.0.6778.70
 - Safari: 18.1
 - Brave: 1.73.91 Chromium: 131.0.6778.85 (Hivatalos verzió) (64 bites)
- **Tesztelési eszközök:** Manuális tesztelés, tesztelési sablonok.
- **Tesztelők:** Ceglédi Zente Holló és Lénárt Dániel Péter.
- **Tesztelt build-ek (Github commit azonosító):** 268e0ff (fox-habits.vercel.app), 32998e9 (foxhabits.com), 79fcbad (foxhabits.com)

4.3.1. 1. Tesztelés

Az alábbi táblázat tartalmazza a tesztelési eseteket az 1. teszt során:

Teszt Eset ID	Leírás	Lépések	Elvárt eredmény	Eredmény
A001	Bejelentkezés érvényes adatokkal	1. Nyisd meg a bejelentkezési oldalt. 2. Írd be az érvényes felhasználónevet és jelszót 3. Kattints a Bejelentkezés gombra.	A rendszer bejelentkezik, és a kezdőlap megjelenik.	Pass
A002	Bejelentkezés hibás adatokkal	1. Nyisd meg a bejelentkezési oldalt 2. Írd be hibás/nem létező adatokat 3. Kattints a Bejelentkezés gombra.	A rendszer helytelen adatokra figyelmeztető üzenetet jelenít meg.	Pass
A003	Bejelentkezési oldal meghívása bejelentkezve	1. Nyisd meg a bejelentkezési oldalt bejelentkezett sessionnel.	A rendszer átirányít a profil oldalra.	Pass

A004	Regisztráció oldal meghívása bejelentkezve	1. Nyisd meg a regisztráció oldalt bejelentkezett sessionnel.	A rendszer átirányít a profil oldalra.	Pass
A005	Védett oldal megnyitása bejelentkezés nélkül	1. Nyiss meg egy védett oldalt, például a /profil oldalt.	A rendszer átirányít a bejelentkezés oldalra.	Pass
A006	Regisztráció érvényes adatokkal	1. Nyisd meg a regisztráció oldalt 2. Írd be a nevet, email címet, jelszót és megerősítést 3. Kattints a Regisztráció gombra.	A rendszer átirányít a bejelentkezési oldalra, és email érkezik a regisztrációról.	Pass
A007	Gyenge jelszó regisztrációnál	1. Nyisd meg a regisztráció oldalt 2. Írd be a nevet, email címet, gyenge jelszót és megerősítést 3. Kattints a Regisztráció gombra.	A rendszer figyelmeztető üzenetet jelenít meg, hogy gyenge a jelszó, és a regisztráció nem érvényes.	Pass
A008	Eltérő jelszók regisztrációnál	Nyisd meg a regisztráció oldalt, írd be a nevet, email címet, jelszót és eltérő megerősítést, kattints a Regisztráció gombra.	A rendszer figyelmeztető üzenetet jelenít meg, hogy a jelszavak nem egyeznek, és a regisztráció sikertelen.	Pass

A009	Duplikált regisztráció	<p>1. Nyisd meg a regisztráció oldalt</p> <p>2. Írd be egy nevet, egy létező email címet, jelszót és a jelszó megerősítését</p> <p>3. Kattints a Regisztráció gombra.</p>	A rendszer egy kis üzenetben visszajelzi, hogy létezik egy felhasználó ezzel az email címmel, és átirányít a bejelentkezés oldalra.	Pass
A010	Mező kihagyása regisztrációnál	<p>1. Nyisd meg a regisztráció oldalt</p> <p>2. Írd be egy létező email címet, jelszót és a jelszó megerősítését (min. egy mezőt hagyj üresen)</p> <p>3. Kattints a Regisztráció gombra.</p>	A rendszer egy kis üzenetben visszajelzi, hogy nincs teljesen kitöltve a regisztrációs űrlap.	Pass

4.3.2. 2. Tesztelés

Az alábbi táblázat tartalmazza a tesztelési eseteket a 2. teszt során:

Teszt Eset ID	Leírás	Lépések	Teszthez használt dummy adatok	Elvárt eredmény	Eredmény
B001/1	Regisztráció érvényes adatokkal, e-mail címmel	1. Nyisd meg a regisztráció oldalt. 2. Adj meg érvényes nevet, e-mail címet, jelszót, és azonos jelszót <i>a jelszó megerősítése</i> mezőbe 3. Kattints a <i>Regisztráció</i> gombra.	Név: Kovács József E-mail: kamuemail4@zente.org Jelszó: KovacsJozsi12! Jelszó megerősítése: KovacsJozsi12!	A rendszer átdobja a felhasználót a bejelentkezés oldalra. (regisztrálódik a felhasználó az adatbázisba)	Partially Nem kapott értesítést a sikeres regisztrációról, vagy arról, hogy ellenőrző e-mailt küldtek. (ellenőrző mail nincs formázva)
B001/2	Regisztráció érvényes adatokkal, e-mail címmel	1. Kattints az e-mailben kapott linkre.	-	Jóváhagyásra kerül az e-mail cím	Partially Jóváhagyásra kerül az e-mail, de nem kap értesítést a sikeres

					jóváhagyásról az oldalon.
B002/1	Jelszó visszaállítás kérése, jelszó megváltoztatása majd bejelentkezés	1. Nyisd meg az elfelejtett jelszó oldalt. 2. Add meg az e-mail címed 3. Kattints a visszaállítási link küldése gombra.	E-mail: cegledizente@gmail.com	E-mailt kell kapnod a visszaállítási linkkel.	Pass (Nincs formázva a visszaállító e-mail)
B002/2	Jelszó visszaállítás kérése, jelszó megváltoztatása majd bejelentkezés	1. Kattints az e-mailben kapott linkre 2. Adj meg egy új jelszót és erősítsd meg.	-	Az oldalnak be kell jelentkeznie, és át kell dobnia a <i>profil</i> oldalra.	Partially Nincs olyan szigorú jelszó ellenőrzés, mint a létrehozásnál. Például elfogadott jelszó: "Admin12". Átdobás után nem frissül a header
B002/3	Jelszó visszaállítás kérése, jelszó	1. Nyisd meg a bejelentkezés oldalt.	-	Az oldalnak be kell	Fail "Ismeretlen hiba.

	megváltoztatása majd bejelentkezés	2. Jelentkezz be a megváltoztatott jelszóval.		jelentkeztetnie a felhasználót.	<i>Próbálja újra.</i> " (megjegyzés: nem kapott a regisztrációt jóváhagyandó e- mailt)
B003	Bejelentkezés érvényes adatokkal	1. Nyisd meg a bejelentkezési oldalt. 2. Írd be az érvényes felhasználónevet és jelszót 3. Kattints a Bejelentkezés gombra.	-	A rendszer bejelentkezik, a <i>profil</i> oldal megjelenik.	Pass
B004/1	Jelszó visszaállítás kérése, jelszó megváltoztatása majd bejelentkezés (2.)	1. Nyisd meg az elfelejtett jelszó oldalt. 2. Add meg az e-mail címed 3. Kattints a visszaállítási link küldése gombra.	E-mail: zentecegedi@gmail.com	E-mailt kell kapnod a visszaállítási linkkel.	Pass
B004/2	Jelszó visszaállítás kérése, jelszó	1. Kattints az e-mailben kapott linkre	Jelszó: Admin12	Az oldalnak be kell jelentkeznie,	Pass

	megváltoztatása majd bejelentkezés (2.)	2. Adj meg egy új jelszót és erősítsd meg.		és át kell dobnia a <i>profil</i> oldalra.	
B004/3	Jelszó visszaállítás kérése, jelszó megváltoztatása majd bejelentkezés (2.)	1. Nyisd meg a bejelentkezés oldalt. 2. Jelentkezz be a megváltoztatott jelszóval.	-	Az oldalnak be kell jelentkeztetnie a felhasználót.	Pass
B005	Jelszó visszaállítás kérése nem létező felhasználó esetében	1. Nyisd meg a <i>jelszó visszaállítás</i> oldalt. 2. Adj meg, egy az oldalon nem regisztrált e- mail címet.	E-mail: czh@czh.hu (nem regisztrált)	Hibaüzenet visszaadása	Fail Üzenet: "Jelszó visszaállítási link elküldve az alábbi e-mail címre."
B006	Bejelentkezés Google-lel, még nem létező felhasználó esetén.	1. Nyisd meg a bejelentkezés oldalt. 2. Kattints a Google-lel való belépés gombra 3. Csináld végig a Google autentikációját	E-mail (Google): zentehollo@gmail.com	A felhasználó átdobása a <i>profil</i> oldalra a bejelentkezett fiókkal.	Pass
B007	Jelszó visszaállítási linkre való	1. Kattints másodjára is a jelszó visszaállítása e- mailben található linkre.	E-mail: zenteceglei@gmail.com	Hibaüzenet visszaadása	Fail Nincs üzenet

	kattintás másodszorra				
B008	Bejelentkezés Google-lel, már korábban e-mailel regisztrált felhasználó esetén.	1. Nyisd meg a bejelentkezés oldalt. 2. Kattints a Google-lel való belépés gombra 3. Csináld végig a Google autentikációját egy olyan fiókkal, amelyel már korábban regisztráltál e- mail címmel.	E-mail: zentecegedi@gmail.com	A felhasználó átdobása a <i>profil</i> oldalra a bejelentkezett fiókkal.	Pass
B009	Bejelentkezés Google-lel, korábban már Google-lel regisztrált felhasználó esetén.	1. Nyisd meg a bejelentkezés oldalt. 2. Kattints a Google-lel való belépés gombra 3. Csináld végig a Google autentikációját egy olyan fiókkal, amelyel már korábban regisztráltál a Google autentikációval.	E-mail (Google): zentehollo@gmail.com	A felhasználó átdobása a <i>profil</i> oldalra a bejelentkezett fiókkal.	Pass

B010	Regisztráció, helytelen megerősítő jelszóval.	1. Nyisd meg a regisztráció oldalt. 2. Adj meg érvényes nevet, e-mail címet, jelszót és írd meg egy eltérő jelszót a jelszó megerősítése mezőbe. 3. Kattints a Regisztráció gombra.	Név: CZH E-mail: cegledizentehollo@gmail.eu.admin Jelszó: Admin.123 Jelszó megerősítése: Admin.124	<i>A jelszavak nem egyeznek</i> hibaüzenetet kell visszaadnia	Pass
-------------	--	---	--	---	------

4.4. Hibák nyomon követése

A hibák rögzítéséhez és nyomon követéséhez a következő formátumot használjuk:

- Hiba azonosító (Bug ID)
- Hiba leírása
- Lépések a reprodukáláshoz
- Elvárt eredmény
- Aktuális eredmény
- Prioritás
- Teszthez használt fiktív (dummy) adatok

- Hiba forrása/Lehetséges megoldás/Megjegyzés

A hibákat, és azok megoldását egy Excel táblázat segítségével vezettük.

4.5. Teszt összegzés

A manuális tesztelések során az összes kritikus funkció lefedésre került. Az elvárt eredmények és az aktuális eredmények alapján több kritikus hibát sikeresen javítottunk.

5. Automata tesztelési dokumentáció

5.1. Cél meghatározása

Ez a dokumentáció az automata tesztelési folyamatokat foglalja össze. Célja, hogy a tesztesetek, szkriptek és eredmények átlátható módon dokumentálva legyenek, ezáltal megkönnyítve a reprodukálhatóságot és a hibajavítást.

5.2. Tesztterv

- **Tesztelési cél:** A szokásokhoz kapcsolódó funkciók megfelelő működésének ellenőrzése.
- **Tesztelt modulok/funkciók:**
 - Egy adott szokás állapota egy adott pillanatban [habit-utils.ts → calculateHabitStreak()]
 - Sorozat (streak) megfelelő kiszámítása a szokások esetén [habit-utils.ts → isHabitCompletedOnDate()]
- **Tesztkörnyezet:**
 - **Operációs rendszer:** Microsoft Windows [Version 10.0.19045.5737]
 - **Fejlesztői környezet:** Visual Studio Code, Cursor
- **Tesztelési eszköz:** Vitest

5.3. Automata tesztek

5.3.1. isHabitCompletedOnDate() funkció tesztelése

5.3.1.1. A teszteléshez használt fiktív (dummy) adatok

```
let dailyHabitWithoutEntries = {
  habit_id: 30,
  habit_interval_type: 'days' as const,
  interval: 1,
  start_date: '2025-04-07T09:27:00+00:00',
  is_active: true,
  habit_type: 'normal_habit' as const,
  habit_names: { habit_name: 'Fogmosás' },
  entries: []
}

let dailyBadHabitWithEntry = {
  habit_id: 11,
  habit_interval_type: 'days' as const,
  interval: 1,
  start_date: '2025-02-06T17:15:00+00:00',
  is_active: true,
  habit_type: 'bad_habit' as const,
  habit_names: { habit_name: 'Cigaretta lerakása' },
  entries: [
    {
      datetime: '2025-03-02T20:56:04.559',
      entry_id: 12,
```



```

    habit_id: 11,
    entry_type: 'done' as const,
    time_of_entry: '2025-03-02T20:56:04.338+00:00'
  }
]
}

let dailyNormalHabitWithEntry = {
  habit_id: 11,
  habit_interval_type: 'days' as const,
  interval: 1,
  start_date: '2025-02-06T17:15:00+00:00',
  is_active: true,
  habit_type: 'normal_habit' as const,
  habit_names: { habit_name: 'Cigaretta lerakása' },
  entries: [
    {
      datetime: '2025-03-02T20:56:04.559',
      entry_id: 12,
      habit_id: 11,
      entry_type: 'done' as const,
      time_of_entry: '2025-03-02T20:56:04.338+00:00'
    },
    {
      datetime: '2025-03-04T18:20:04.559',
      entry_id: 12,
      habit_id: 11,
      entry_type: 'skipped' as const,
      time_of_entry: '2025-03-02T20:56:04.338+00:00'
    }
  ]
}

```

```

    ]
  }

  let weeklyNormalHabitWithEntry = {
    habit_id: 12,
    habit_interval_type: 'weeks' as const,
    interval: 1,
    start_date: '2025-02-03T17:15:00+00:00',
    is_active: true,
    habit_type: 'normal_habit' as const,
    habit_names: { habit_name: 'Heti edzés' },
    entries: [
      {
        datetime: '2025-02-05T20:56:04.559',
        entry_id: 13,
        habit_id: 12,
        entry_type: 'done' as const,
        time_of_entry: '2025-03-02T20:56:04.338+00:00'
      },
      {
        datetime: '2025-02-14T18:20:04.559',
        entry_id: 14,
        habit_id: 12,
        entry_type: 'skipped' as const,
        time_of_entry: '2025-03-04T18:20:04.338+00:00'
      }
    ]
  }
}

```

5.3.1.2. A tesztesetek

Automata Teszt ID	Leírás	Input Adatok	Elvárt eredmény	Eredmény (Pass/Fail)
A/A001	Napi szokás tesztelése bejegyzés nélkül (megadott dátummal)	dailyHabitWithoutEntries, 2025-04-07T09:27:00+00:00	isCompleted: false, isSkipped: false	Pass
A/A002	Napi szokás tesztelése bejegyzés nélkül (aktuális dátummal)	dailyHabitWithoutEntries, Date()	isCompleted: false, isSkipped: false	Pass
A/A003	Napi szokás tesztelése elvégzett bejegyzéssel	dailyNormalHabitWithEntry, 2025-03-02T09:27:00+00:00	isCompleted: true, isSkipped: false	Pass
A/A004	Napi szokás tesztelése elvégzett bejegyzéssel nap elején	dailyNormalHabitWithEntry, 2025-03-02T00:00:00+00:00	isCompleted: true, isSkipped: false	Pass
A/A005	Napi szokás tesztelése kihagyott bejegyzéssel	dailyNormalHabitWithEntry, 2025-03-04T09:27:00+00:00	isCompleted: false, isSkipped: true	Pass

A/A006	Napi szokás tesztelése nem elvégzett és nem kihagyott bejegyzéssel	dailyNormalHabitWithEntry, 2025-03-07T09:27:00+00:00	isCompleted: false, isSkipped: false	Pass
A/A007	Rossz napi szokás tesztelése nem elvégzett és nem kihagyott bejegyzéssel	dailyBadHabitWithEntry, 2025-03-07T09:27:00+00:00	isCompleted: true	Pass
A/A008	Rossz napi szokás tesztelése elvégzett bejegyzéssel	dailyBadHabitWithEntry, 2025-03-02T09:27:00+00:00	isCompleted: false	Pass
A/A009	Heti szokás tesztelése nem elvégzett bejegyzéssel	weeklyNormalHabitWithEntry, 2025-03-02T09:27:00+00:00	isCompleted: false, isSkipped: false	Pass
A/A010	Heti szokás tesztelése adott héten elvégzett bejegyzéssel	weeklyNormalHabitWithEntry, 2025-02-04T12:30:00+00:00	isCompleted: true, isSkipped: false	Pass

A/A011	Heti szokás tesztelése adott héten elvégzett bejegyzéssel (második példa)	weeklyNormalHabitWithEntry, 2025-02-09T12:30:00+00:00	isCompleted: true, isSkipped: false	Pass
--------	--	--	--	------

5.3.1. isHabitCompletedOnDate() funkció tesztelése

5.3.1.1. A teszteléshez használt fiktív (dummy) adatok

```
let habitWithStreak = {
  habit_id: 16,
  habit_interval_type: 'days' as const,
  interval: 1,
  start_date: '2025-02-03T17:15:00+00:00',
  is_active: true,
  habit_type: 'normal_habit' as const,
  habit_names: { habit_name: 'Fogmosás' },
  entries: [
    {
      datetime: '2025-02-05T20:56:04.559',
      entry_id: 13,
      habit_id: 16,
      entry_type: 'done' as const,
      time_of_entry: '2025-03-02T20:56:04.338+00:00'
    },
    {
      datetime: '2025-02-06T20:56:04.559',
```

```

    entry_id: 14,
    habit_id: 16,
    entry_type: 'done' as const,
    time_of_entry: '2025-03-02T20:56:04.338+00:00'
  },
  {
    datetime: '2025-02-07T20:56:04.559',
    entry_id: 15,
    habit_id: 16,
    entry_type: 'done' as const,
    time_of_entry: '2025-03-02T20:56:04.338+00:00'
  }
]
}

```

Automata Teszt ID	Leírás	Input Adatok	Elvárt eredmény	Eredmény (Pass/Fail)
A/B001	Szokás ellenőrzése 3 napos sorozattal (streak- kel), az utolsó bejegyzés napján	habitWithStreak, 2025-02-07T01:00:00+00:00	streak: 3	Pass

A/B002	Szokás ellenőrzése 3 napos sorozattal (streak- kel), az utolsó nap utáni napon	habitWithStreak, 2025-02-08T01:00:00+00:00	streak: 3	Pass
A/B003	Szokás ellenőrzése 3 napos sorozattal (streak- kel), az utolsó nap utáni második napon	habitWithStreak, 2025-02-09T01:00:00+00:00	streak: 0	Pass

5.4. Hibák nyomon követése

Az automata tesztek során észlelt hibák rögzítéséhez a következő formátumot alkalmazzuk:

- Hiba azonosító (Bug ID)
- Hiba leírása
- Lépések a reprodukáláshoz
- Teszt Szkript címe
- Elvárt eredmény
- Aktuális eredmény
- Prioritás
- Státusz (nyitott, javítva, lezárva)

5.5. Teszt összegzés

Az automata tesztelés során a szokásokhoz kapcsolódó összes kritikus funkció lefedésre került. Az automata tesztek futtatása és a kapott eredmények alapján a szoftver megfelelő működése biztosított.

6. Felhasználói dokumentáció

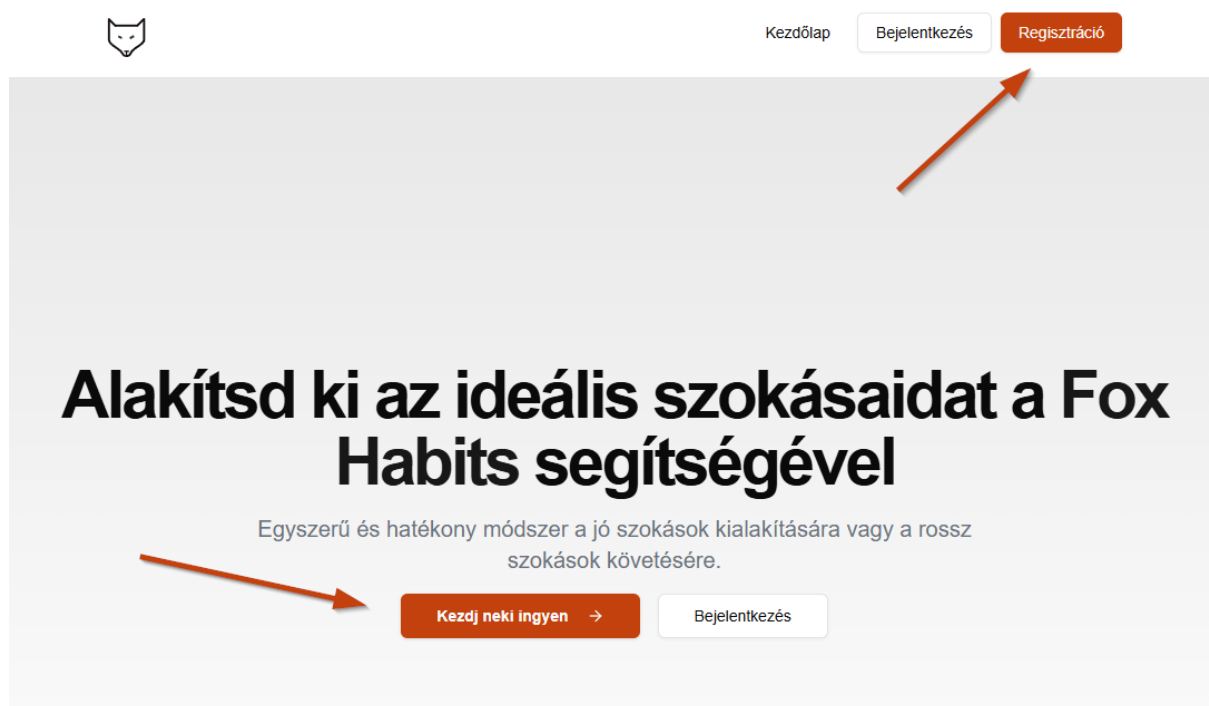
6.1. Bevezetés

A **FoxHabits** célja, hogy segítsen a felhasználók felhasználói számára a mindennapi szokásaik és feladataik követését. A platform továbbá segíti a fejlődést különböző kimutatások készítésével és értesítések küldésével.

6.2. Regisztráció és bejelentkezés

1. Keresd fel a FoxHabits weboldalát a <https://foxhabits.com/> domain-en

2. Kattints a „Kezdj neki ingyen!” vagy a „Regisztráció” gombra.



3. Add meg a neved, e-mail címed és a jelszavadat, vagy kattints egyszerűen a „Google” gombra a regisztrációs űrlapnál.

3.1. Amennyiben a Google-lel történő regisztrációt választottad, válaszd ki a megfelelő Google fiókot, majd fogadd el a felhasználói feltételeket és az adatvédelmi szabályzatot. Ebben az esetben az oldal automatikusan be is jelentkezett.

Regisztráció

Regisztrálj egyszerűen a neveddel és email címeddel

Név


Email

Jelszó


Jelszó megerősítése

Regisztráció

— vagy —


 Google


Már van fiókod? [Bejelentkezés](#)


 Bejelentkezés Google-fiókkal

Fiókválasztás

a(z) `rffkwoefcijnkadminbc.supabase.co` alkalmazásba való továbblépéshez


@gmail.com


@gmail.com

 **Másik fiók használata**

Az alkalmazás használata előtt tekintse át a(z) `rffkwoefcijnkadminbc.supabase.co` **adattvédelmi irányelveit** és **általános szerződési feltételeit**.

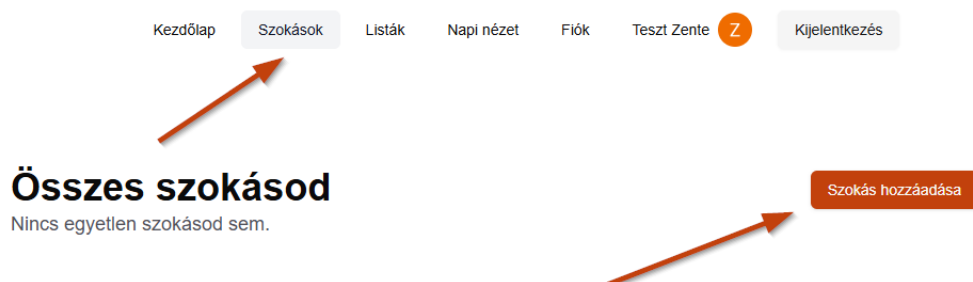
4. A regisztráció után rendszerünk megerősítő e-mail-t küld a korábban megadott e-mail címre, amelyre rá kell kattintanod, hogy véglegesítsd a regisztrációd.

5. Az e-mailben kapott linkre kattintva az oldal átirányít a bejelentkezés oldalra, ahol a már korábban megadott e-mail címmel és jelszóval bejelentkezhetsz az oldalra.

6.3. Szokások kezelése

Felhasználóként szokásokat hozhatsz létre és kezelhetsz.

1. Navigálj a navigációs menün keresztül a „Szokások” aloldalra.
2. Kattints a „Szokás hozzáadása” gombra az első szokásod létrehozásához.



3. Add meg az új szokásod nevét, típusát, intervallumát és a kezdő dátumot.

A szokás típusának kiválasztásánál dönthetsz a normál és a káros szokás kiválasztása között. Normál szokás esetében a cél, hogy az adott szokást minél hosszabb időn keresztül fenntartsd, káros szokás esetén a cél, hogy minél hosszabb ideig mellőzd az adott szokást.

Az intervallum esetében a kiválasztott számmal adható meg, hogy egy adott időszakon keresztül (naponta, hetente, havonta, évente) hányszor kell az adott szokást elvégezni.

A kezdő dátumot akkor érdemes módosítani, ha egy szokást később szeretne a felhasználó elkezdni vagy ha például a hetente elvégzendő szokást hétfőn szeretné kezdeni a felhasználó.

Például: Spanyol nyelv gyakorlása hetente kétszer a mai naptól kezdve.

Új szokás létrehozása

Add meg az új szokásod részleteit alább.

Szokás neve

Spanyol nyelv gyakorlása

Szokás típusa

Normál szokás

Intervallum

2

Intervallum típusa

hetente

Kezdő dátum

2025. 03. 23. 18:44

Szokás létrehozása

4. Kattints a „Szokás létrehozása” gombra.

A szokás létrehozása után a szokások aloldalon megtalálható lesz az új szokás.

Itt van lehetőségünk a létrehozott szokásokat:

- szerkeszteni, a ceruza ikonra történő kattintással
- törölni, a szemetes ikonra történő kattintással

Spanyol ...

Aktív

Szokás

Minden 2. héten

Elkezdve: 2025 March 23.

Létrehozva: 2025 March 23.

Spanyol ...

Aktív

Szokás

Minden 2. héten

Elkezdve: 2025 March 23.

Elvégezve: 2025 March 23. 18:55

Létrehozva: 2025 March 23.

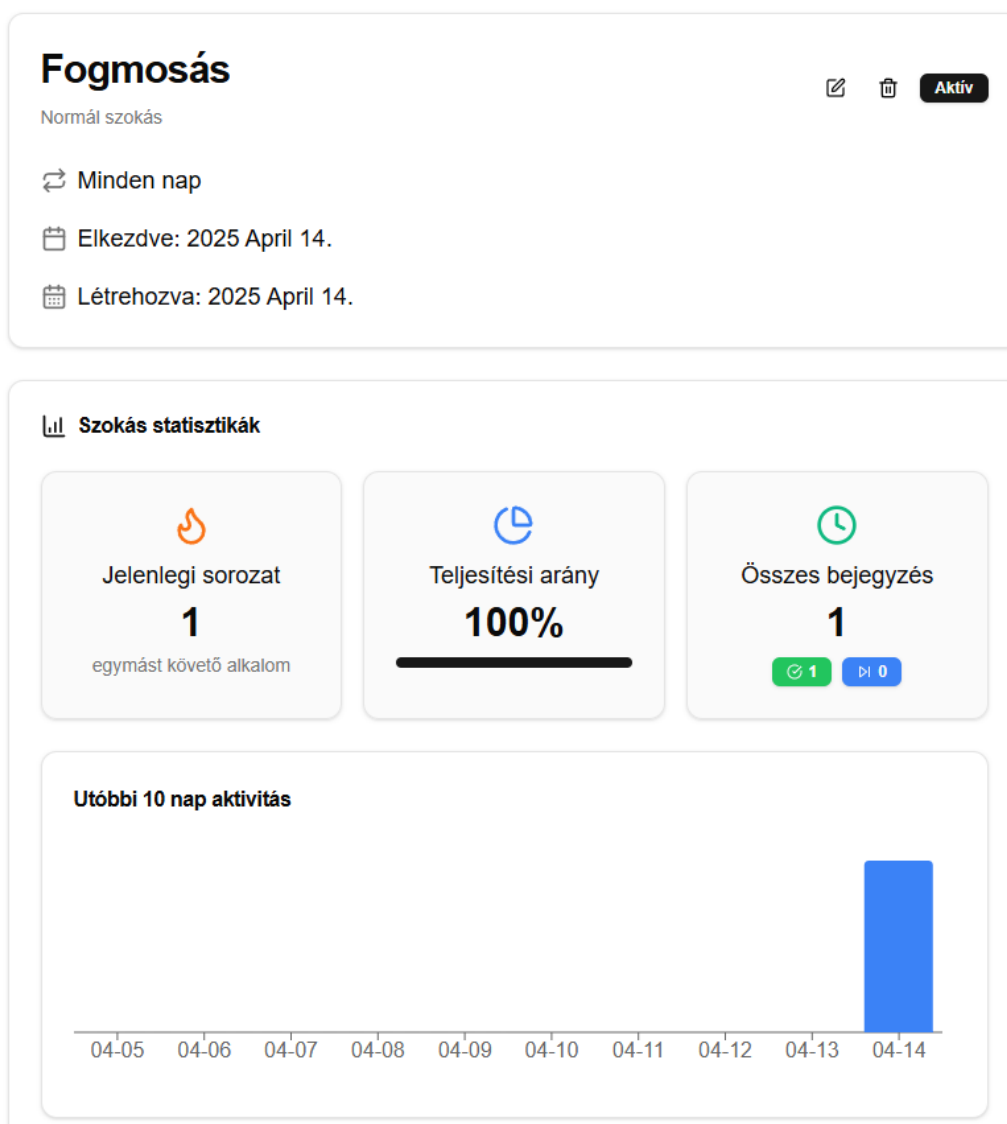
5. Amikor a felhasználó elvégezte a szokását, azt a „Szokások” oldalon késznek jelölheti a pipa ikon segítségével.

Amennyiben a szokás nem került elvégzésre, de az indokkal történt (például betegség miatt nem teljesíthető) azt a felhasználó a jobbra mutató nyíl segítségével kihagyottnak jelölheti meg, ebben az esetben a sorozat nem kerül megszakításra, csupán nem növekszik.

Ha a felhasználó nem végzi el a szokást, vagy nem jelöli kihagyottnak, abban az esetben a rendszer nem teljesítettként kezeli, ez így jelenik meg a statisztikákban is.

6. Van lehetősége a felhasználónak az adott szokás adatait megtekinteni, ha az adott szokáshoz tartozó kártyára kattint a „Szokások” oldalon.

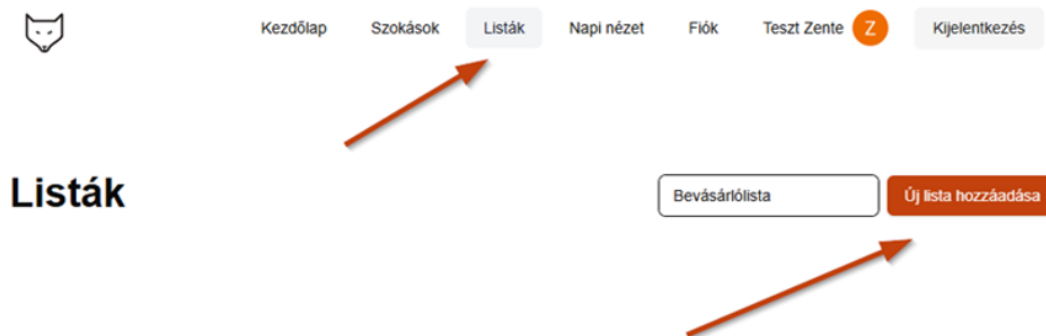
Ezen az oldalon megtalálhatóak az adott szokáshoz tartozó korábbi naplózások, a leghosszabb és jelenlegi egybefüggő sorozat (streak) adatai, és egyéb, az elvégzéshez kapcsolódó statisztikák.



6.4. Listák

A felhasználóknak lehetőségük van listákat (egyszerű To Do) létrehozni és kezelni. Ennek célja, hogy az alkalmazásban a felhasználó tudja vezetni a bevásárlólistáját, teendőit.

1. Navigálj, a navigációs menün keresztül a „Listák” aloldalra.
2. Az oldal jobb felső oldalán írd be az új lista nevét, majd kattints az „Új lista hozzáadása” gombra, vagy üsd le az ENTER billentyűt.



Az újonnan létrehozott listák között a listák navigációs sorában található gombokkal lehet váltani.

A lista nevét szerkeszteni tudjuk a ceruza ikonnal, törölni pedig a kuka ikonnal tudjuk a listát.

3. A létrehozott listába elemeket lehet felvinni, ezt az „Új elem” mezőbe történő írással, majd az ENTER billentyű lenyomásával vagy a „Hozzáadás gomb” megnyomásával tehetjük meg.

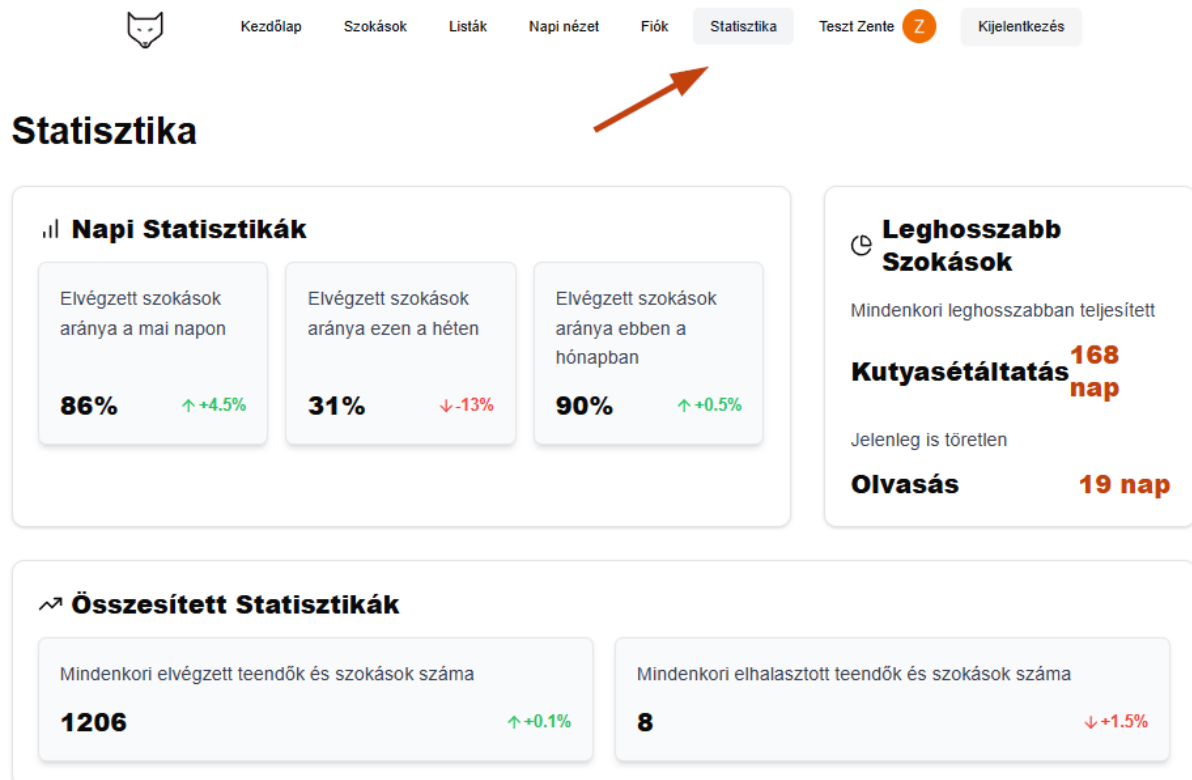
Az elemet teljesítettként / elvégzettként jelölhetjük meg a listaelem bal oldalán található jelölőnégyzettel. Az elem nevét szerkeszthetjük a ceruza ikonnal, vagy törölhetjük a kuka ikonra történő kattintással.



6.5. Statisztika

Felhasználóként van lehetőség a teljes portálra vonatkozó statisztikákat megtekinteni a saját szokásairól összesítve.

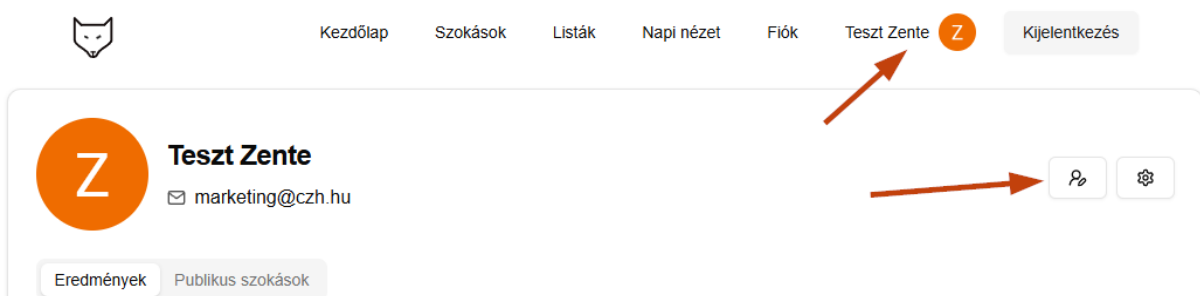
1. Navigálj a „Statisztikák” gombra kattintva a navigációs soron keresztül.
2. Tekintsd meg a statisztikáidat.



6.6. Beállítások

A felhasználónak van lehetősége a saját kijelzett nevét szerkeszteni.

1. Navigálj a saját nevedre kattintva a navigációs menün keresztül.
2. A felhasználó és ceruza ikonra kattintva tudod módosítani a nevedet az alkalmazáson belül.



3. A fogaskerék ikonra kattintva módosíthatod, hogy szeretnél-e e-mailes értesítéseket kapni, beállíthatod, hogy milyen témával szeretnéd használni az alkalmazást, valamint exportálhatod a szokásaidat és a log-jaidat (szokás bejegyzéseidet).

Beállítások

Általános beállítások

Személyre szabhatod a szokáskövető alkalmazást

Téma

Világos



Értesítési beállítások

Beállíthatod, hogyan szeretnéd megkapni az értesítéseket

Email értesítések



Adatkezelés

Exportálhatod a szokásaidat és a hozzájuk tartozó adataidat.

Adatok exportálása

Beállítások mentése

7. Összegzés

7.1. Kihívások

A projekt során több kihívással is szembesültünk.

- Munka felosztása – Kezdetben úgy döntöttünk felosztjuk egymás között a frontend-en és a backend-en végzendő feladatokat, ám a megvalósítás közben sokkal jobbnak láttuk, ha inkább adott funkciókat csak az egyikünk dolgozza ki teljes mértékben, így úgy láttuk, hogy sokkal hatékonyabban sikerült dolgoznunk.
- Új technológiák – A projekt fejlesztése során számos új technológiával találkoztunk, amelyeket eleinte nehezen sajátítottunk el, ám idővel sikerült ezeket megértenünk és átlátnunk.
 - Next.js sajátosságai (pl. prefetching) – Az alkalmazás fejlesztése során olyan kihívásokkal szembesültünk, amelyek más Node.js-alapú webes keretrendszereknél korábban nem merültek fel, így ezekre új megoldásokat kellett találnunk.
- Időmenedzsment – Az egyik legnagyobb probléma számunkra a projekt fejlesztése során a megfelelő időbeosztás volt. A hét hónapos periódus során az első hónapokban nagyon gyorsan haladtunk, majd lelassult a fejlesztés az időszak közepén, így az utolsó hónapra jó néhány feladatunk maradt.

7.2. Fejlődés

A vizsgaremek elkészítése során jelentős fejlődést értünk el a Next.js és a React használatában. Emellett fejlődöttünk a csapatmunkában, megtanultuk, hogy miként tudjuk a feladatokat hatékonyan felosztani egymás között, fejlődött az időmenedzsmentünk, valamint a kommunikációs készségeink is.

7.3. Jövőbeli tervek

Az alkalmazással kapcsolatban számos ötletünk felmerült a továbbfejlesztésre:

- Mobilalkalmazás
- Megosztott szokások
- AI-funkciók implementálása
- Értesítésrendszer fejlesztése
- Statisztikák kibővítése

Szeretnénk az alkalmazásunkat továbbfejleszteni egy olyan szintre, hogy a piacon is megállja a helyét.

7.4. Költségek

Domain vásárlás (érvényesség: 2024. december 3. – 2025. december 3.)

TYPE	NAME	QTY	DURATION	PRICE	SUB TOTAL
PURCHASE	PremiumDNS	1	1 year	\$2.98	\$2.98
REGISTER	Domain Registration	1	1 year	\$6.49	\$6.49
	ICANN Fee	-	-	\$0.18	\$0.18
	TOTAL	-	-	-	\$9.65 ~ 3 827,62 Ft

7.5. Forrásmegjelölés

- Frontend
 - <https://ui.shadcn.com/>
 - <http://tailwindcss.com/>
 - <https://daisyui.com/components/timeline/>
- Backend
 - <https://supabase.com/docs/>
 - <https://nextjs.org/docs/>
- Inspiráció
 - <https://www.habitify.me/>
 - <https://www.todoist.com/>
 - <https://notion.so/>
- Egyéb
 - <https://stackoverflow.com/>
 - <https://chatgpt.com/>
 - <https://claude.ai/new>
 - <https://www.perplexity.ai/>
 - <https://chat.deepseek.com/>