✅ Backend Tesztelői Dokumentáció

# 1. Általános információk

- Technológia: Next.js API route, Prisma ORM, MongoDB  
- Tesztelendő részek: API végpontok, adatvalidáció, hibakezelés  
- Tesztelési módszerek: Manuális API tesztelés (Postman) , Egység- és integrációs tesztek (Jest)

# 2. Környezet beállítása

Fejlesztői környezet indítása:

npm install  
npx prisma generate  
npm run dev

Tesztadatbázis beállítása (`.env`):

DATABASE\_URL="mongodb+srv://test:test@user.0z2pk.mongodb.net/test "

# 3. Tesztelési forgatókönyvek (manuális tesztek)

**### Felhasználói modul**

1. **\*\*Felhasználó sikeres létrehozása\*\***

   - Végpont: `POST /user`

   - Bemenet:

     ```json

     {

       "firstName": "John",

       "lastName": "Doe",

       "email": "john.doe@example.com",

       "password": "StrongP@ssw0rd!"

     }

     ```

   - Elvárt kimenet: A létrehozott felhasználó adatai, beleértve az `id`-t.

2. **\*\*Bejelentkezés érvényes adatokkal\*\***

   - Végpont: `POST /user/login/token`

   - Bemenet:

     ```json

     {

       "email": "john.doe@example.com",

       "password": "StrongP@ssw0rd!"

     }

     ```

   - Elvárt kimenet: Felhasználói adatok és egy JWT token.

3. **\*\*Bejelentkezés érvénytelen jelszóval\*\***

   - Végpont: `POST /user/login/token`

   - Bemenet:

     ```json

     {

       "email": "john.doe@example.com",

       "password": "WrongPassword"

     }

     ```

   - Elvárt kimenet: `401 Unauthorized` hiba.

4. **\*\*Felhasználó adatainak lekérése azonosító alapján\*\***

   - Végpont: `GET /user/:id`

   - Bemenet: Az URL-ben: `id = "user123"`

   - Elvárt kimenet: A felhasználó adatai.

5. **\*\*Felhasználó banki adatokkal történő lekérése\*\***

   - Végpont: `GET /user/userbank/:id`

   - Bemenet: Az URL-ben: `id = "user123"`

   - Elvárt kimenet: A felhasználó adatai, beleértve a kapcsolódó bankszámlákat.

6. **\*\*Felhasználó törlése\*\***

   - Végpont: `DELETE /user/:id`

   - Bemenet: Az URL-ben: `id = "user123"`

   - Elvárt kimenet: A felhasználó sikeres törlése.

---

**### Bevételi modul**

1. **\*\*Bevétel sikeres hozzáadása\*\***

   - Végpont: `POST /Income`

   - Bemenet:

     ```json

     {

       "total": 500,

       "category": "Salary",

       "description": "Monthly salary",

       "userId": "user123",

       "bankAccountId": "account123"

     }

     ```

   - Elvárt kimenet: A létrehozott bevétel adatai.

2. **\*\*Bevétel lekérése azonosító alapján\*\***

   - Végpont: `GET /Income/:id`

   - Bemenet: Az URL-ben: `id = "income123"`

   - Elvárt kimenet: A bevétel adatai.

3. **\*\*Bevétel frissítése\*\***

   - Végpont: `PATCH /Income/:id`

   - Bemenet:

     ```json

     {

       "total": 600,

       "category": "Bonus",

       "description": "Year-end bonus",

       "userId": "user123",

       "bankAccountId": "account123"

     }

     ```

   - Elvárt kimenet: A frissített bevétel adatai.

4. **\*\*Bevétel törlése\*\***

   - Végpont: `DELETE /Income/:id`

   - Bemenet: Az URL-ben: `id = "income123"`

   - Elvárt kimenet: A bevétel sikeres törlése.

---

**### Kiadási modul**

1. **\*\*Kiadás sikeres hozzáadása\*\***

   - Végpont: `POST /expense`

   - Bemenet:

     ```json

     {

       "total": 100,

       "category": "Rent",

       "description": "Monthly rent payment",

       "userId": "user123",

       "bankAccountId": "account123"

     }

     ```

   - Elvárt kimenet: A létrehozott kiadás adatai.

2. **\*\*Kiadás lekérése azonosító alapján\*\***

   - Végpont: `GET /expense/:id`

   - Bemenet: Az URL-ben: `id = "expense123"`

   - Elvárt kimenet: A kiadás adatai.

3. **\*\*Kiadás frissítése\*\***

   - Végpont: `PATCH /expense/:id`

   - Bemenet:

     ```json

     {

       "total": 120,

       "category": "Utilities",

       "description": "Electricity bill",

       "userId": "user123",

       "bankAccountId": "account123"

     }

     ```

   - Elvárt kimenet: A frissített kiadás adatai.

4. **\*\*Kiadás törlése\*\***

   - Végpont: `DELETE /expense/:id`

   - Bemenet: Az URL-ben: `id = "expense123"`

   - Elvárt kimenet: A kiadás sikeres törlése.

---

**### Bankszámla modul**

1. **\*\*Új bankszámla létrehozása\*\***

   - Végpont: `POST /accounts`

   - Bemenet:

     ```json

     {

       "userId": "user123",

       "currency": "HUF",

       "ownerName": "John Doe"

     }

     ```

   - Elvárt kimenet: A létrehozott bankszámla adatai.

2. **\*\*Bankszámla adatainak lekérése azonosító alapján\*\***

   - Végpont: `GET /accounts/:id`

   - Bemenet: Az URL-ben: `id = "account123"`

   - Elvárt kimenet: A bankszámla adatai.

3. **\*\*Bankszámlához kapcsolódó kiadások lekérése\*\***

   - Végpont: `GET /accounts/allex/:id`

   - Bemenet: Az URL-ben: `id = "account123"`

   - Elvárt kimenet: A bankszámlához kapcsolódó kiadások listája.

4. **\*\*Bankszámlához kapcsolódó bevételek lekérése\*\***

   - Végpont: `GET /accounts/allin/:id`

   - Bemenet: Az URL-ben: `id = "account123"`

   - Elvárt kimenet: A bankszámlához kapcsolódó bevételek listája.

5. **\*\*Tranzakció két bankszámla között\*\***

   - Végpont: `PATCH /accounts/transfer`

   - Bemenet:

     ```json

     {

       "accountfrom": "account123",

       "accountto": "account456",

       "amount": 1000

     }

     ```

   - Elvárt kimenet: Az összeg levonása az egyik számláról és hozzáadása a másikhoz.

6. **\*\*Bankszámla törlése\*\***

   - Végpont: `DELETE /accounts/:id`

   - Bemenet: Az URL-ben: `id = "account123"`

   - Elvárt kimenet: A bankszámla sikeres törlése.

---

**### Ismétlődő tranzakció modul**

1. **\*\*Ismétlődő tranzakció létrehozása\*\***

   - Végpont: `POST /repeatabletransaction`

   - Bemenet:

     ```json

     {

       "name": "Rent Payment",

       "total": 500,

       "category": "Rent",

       "repeatAmount": 1,

       "repeatMetric": "Month",

       "repeatStart": "2025-01-01T00:00:00Z",

       "repeatEnd": "2025-12-31T00:00:00Z",

       "accountId": "account123",

       "userId": "user123"

     }

     ```

   - Elvárt kimenet: Az ismétlődő tranzakció adatai.

2. **\*\*Ismétlődő tranzakció lekérése azonosító alapján\*\***

   - Végpont: `GET /repeatabletransaction/:id`

   - Bemenet: Az URL-ben: `id = "transaction123"`

   - Elvárt kimenet: Az ismétlődő tranzakció adatai.

3. **\*\*Ismétlődő tranzakcióhoz kapcsolódó kiadások lekérése\*\***

   - Végpont: `GET /repeatabletransaction/expense/:id`

   - Bemenet: Az URL-ben: `id = "transaction123"`

   - Elvárt kimenet: Az ismétlődő tranzakcióhoz kapcsolódó kiadások listája.

4. **\*\*Ismétlődő tranzakció frissítése\*\***

   - Végpont: `PATCH /repeatabletransaction/:id`

   - Bemenet:

     ```json

     {

       "name": "Updated Rent Payment",

       "total": 600,

       "category": "Rent",

       "repeatAmount": 1,

       "repeatMetric": "Month",

       "repeatStart": "2025-01-01T00:00:00Z",

       "repeatEnd": "2025-12-31T00:00:00Z"

     }

     ```

   - Elvárt kimenet: A frissített tranzakció adatai.

5. **\*\*Ismétlődő tranzakció leállítása\*\***

   - Végpont: `DELETE /repeatabletransaction/stop/:id`

   - Bemenet: Az URL-ben: `id = "transaction123"`

   - Elvárt kimenet: Az ismétlődő tranzakció sikeres törlése.

# 4. Swagger használata teszteléshez Online tesztelés: 1. Indísd el a szervert (npm run start:dev) 2. Add meg az URL-t a keresőben: http://localhost:3000/docs

# 5. Teszteléshez használt eszközök

- ✅ Postman (manuális API hívásokhoz)  
- ✅ Jest (automata teszteléshez)  
- ✅ Swagger UI (dokumentált API interakció)