

Node.js alapú backend fejlesztés

Gál Mihály

vizsgafeladata

2022

<https://github.com/gal-mihaly/webuniNodeJs.git>

1.Feladat leírása

Twitter klón alkalmazás készítése. Az alkalmazásba felhasználók jelentkezhetnek be és szöveges alapú twiteket írhatnak, azokat módosíthatják és akár törölhetik is. Az összes felhasználó egy lista oldalon kerül megjelenítésre és ezek alapján a hozzátartozó twitek megjeleníthetők.

2. Szereplők és igények

A rendszer első verziója kétféle szereplőt különböztet meg.

A **látogató** egy olyan felhasználó akinek még nincs regisztrációja a rendszerbe, ezért nem képes twiteket készíteni.

A **felhasználó** már képes bejelentkezni a rendszerbe, így már saját twiteket hozhat létre, azokat módosíthatja és akár törölheti is.

A rendszerben listázott twitek időrendben csökkenő listázásban kerülnek megjelenítésre, azaz a legfrissebb twit lesz legfelül látható.

Az első verzióban a regisztrációt csak a rendszergazdák képesek elvégezni, akiknek a regisztrációs szándékról e-mailt kell küldeni és a felhasználókat manuálisan fogják regisztrálni. Az így elkészített regisztrációról e-mail üzenetet küldenek, ami tartalmazza a belépési jelszavat.

Az első verzióban a felhasználók nem képesek módosítani a jelszavukat vagy elfelejtett jelszavat kérni, azt a rendszergazdától emailben kérhetik.

3. Use-case funkció lista

- látogatók képesek szöveges tweeteket olvasni. A tweetek csökkenő időrendi sorrendben kerülnek listázásra.
- felhasználók képesek regisztrálni a rendszerbe (emailcím és jelszó megadásával). A rendszer email üzeneteket küld a sikeres regisztrációról.
- felhasználó képes egy felhasználó név / vagy email párossal bejelentkezni a rendszerbe.
- a felhasználó képes az jelszó emlékeztetőt küldetni a rendszerrel amennyiben azt elfelejti. A rendszer email üzenetben küldi ki az aktuális jelszavát a felhasználónak.
- felhasználó képes kijelentkezni a rendszerből
- egy bejelentkezett felhasználó képes tweeteket olvasni, a tweetek csökkenő időrendi sorrendben kerülnek listázásra
- bejelentkezett felhasználók képesek szöveges tweeteket készíteni
- bejelentkezett felhasználó képes egy saját tweetjét módosítani
- bejelentkezett felhasználó képes egy saját tweetjét törölni
- egy bejelentkezett felhasználó képes a többi felhasználót egy listában böngészni, aminek során egy felhasználóra kattintva egy új oldalon keresztül a kiválasztott felhasználó valamennyi tweetjét olvasni.

4. Magas szintű rendszerterv

Az alkalmazás **server side rendering** technológiával **ejs template** használatával kerül fejlesztésre.

Az alkalmazásban használt adatok **adatbázis**ba kerülnek perzisztálásra, az alkalmazás ezen verziójában a **lokijs** adatbáziskezelő kerül használatra. A rendszer ezen verziója a jelszavakat plaintext módon közvetlenül az adatbázisban tárolja.

A bejelentkezett felhasználók kezelése **memória alapú session storage** használatával lesz megvalósítva.


Az email küldő service jelen alkalmazás verzióban csupán egy console logon keresztül valósul meg.

A fejlesztés során a fejlesztők munkája **nodemon** npm package használatával, azonban ez csak a development verzióban lesz elérhető. (-- save -D)

Az alkalmazás a jelszavak kezelését jelen verzióban sima **plain text**-ként kezeli. (nem kerül titkosításra)

5. Screenek

Bejelentkezési oldal



A login screen with a light green background. The title "LOGIN" is centered at the top in a large, dark grey font. Below the title, there are two input fields: "E-Mail Address" and "Password". The "E-Mail Address" field is a white rectangle with a thin black border. The "Password" field is a white rectangle with a thin black border. To the right of the "Password" field, there is a link "Forgot Password?" in a small, blue font. Below the input fields, there is a blue button with the text "Login" in white. At the bottom, there is a link "Don't have an account? Sign Up" in a small, blue font.

LOGIN

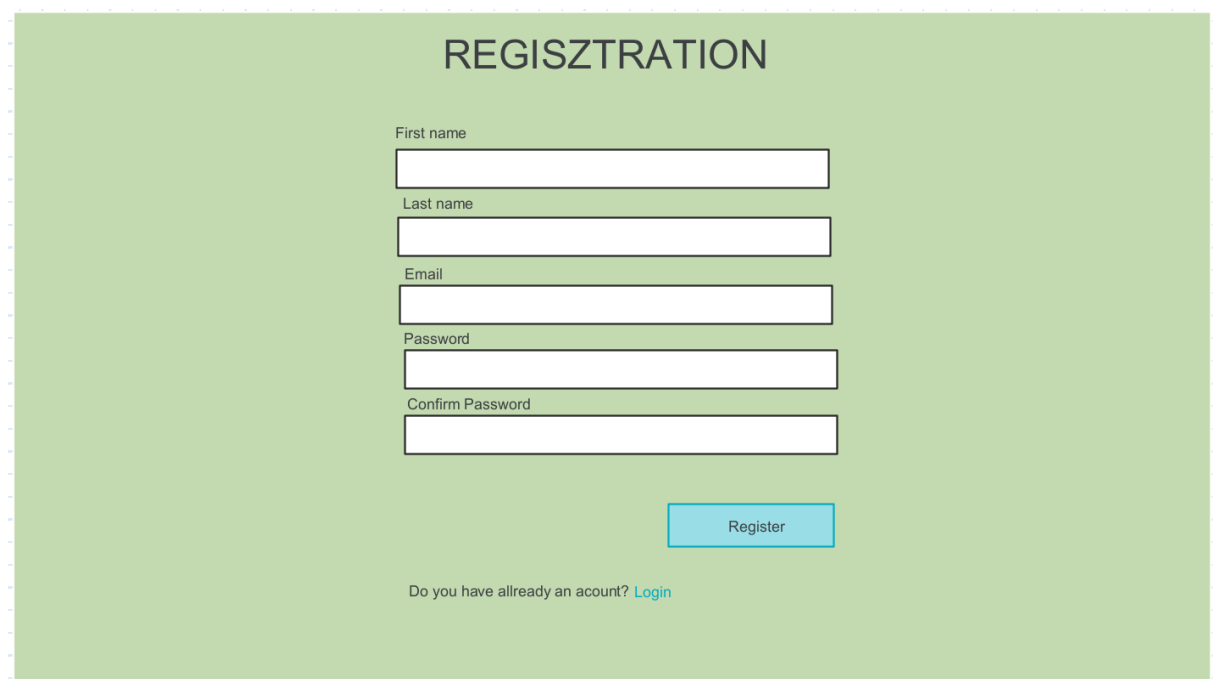
E-Mail Address

Password [Forgot Password?](#)

Login

Don't have an account? [Sign Up](#)

Regisztrációs oldal



A registration screen with a light green background. The title "REGISZTRATION" is centered at the top in a large, dark grey font. Below the title, there are five input fields: "First name", "Last name", "Email", "Password", and "Confirm Password". Each field is a white rectangle with a thin black border. Below the input fields, there is a blue button with the text "Register" in white. At the bottom, there is a link "Do you have allready an account? Login" in a small, blue font.

REGISZTRATION

First name

Last name

Email

Password

Confirm Password

Register

Do you have allready an account? [Login](#)

Elfelejtett jelszó oldal

Forgot your password?

Please, type in your email address for resending your password.

E-Mail Address

Request

Do you have an account? [Login](#)

Home oldal

LOGO MyTwitter

Home [Twitters](#) [User](#)

Write your twite!

Title

Your twite

Twite it!

user@gg.aa

My Firest twite
Hi everyone! This my first twite.

2022.12.00 00:00

Twite szerkesztés oldal

You are editing your twite

Title

Your twite

Twittererek lista oldal

LOGO

MyTwitter

Home [Twitters](#) [User](#)

LIST OF TWITTERS

[user@gg.aa](#)
[user2@gg.aa](#)
[user3@gg.aa](#)
[user4@gg.aa](#)

Twitter posztjai oldal

LOGO

MyTwitter

Home Twitters User

Twites of user 1

user1@gg.aa

My newest twite

The weather is so cooled!

2022.12.00 01:00

user1@gg.aa

My Firest twite

Hi everyone! This my first twite.

2022.12.00 00:00

6. Routing

Ebben a fejezetben az alkalmazásban használt routok kerülnek ismertetésre

GET	/home authMW homeMW	Főoldal
GET	/signup signupMW	Regisztrációs oldal megtekintése
POST	/signup signuppostMW	Regisztrációs adatok átadása backend-nek
GET	/login loginMW	Bejelentkezési oldal megtekintése
POST	/login loginpostMW	Bejelentkezési adatátadás backend-nek
GET	/forgot forgotMW	Elfelejtett jelszó oldal megtekintése
POST	/forgot forgotpostMW	Elfelejtett jelszó adatok átadása backend-nek
GET	/logout authMW logoutMW	Felhasználó kijelentkezésének kérése a backend-től
POST	/twite authMW twitepostMW	Új twite adatainak átadása backend-nek
GET	/twites/edit/:twiteld authMW editTwiteMW	Twite szerkesztési oldal megtekintése
POST	/twites/edit/:twiteld authMW twitePostMW	Twite szerkesztési adatok átadása backend-nek

GET	/twites/delete/:twiteld <i>authMW</i> <i>delTwiteMW</i>	Twite törlés kérése backend-től
GET	/twitters <i>authMw</i> <i>twittersMW</i>	Twitterek lista oldal megtekintése
GET	/twitter:twitterId <i>authMW</i> <i>twitterMW</i>	Egy twiter twite-inek megjelenítőoldal megtekintése
GET	/500 <i>internalServerErrorMW</i>	Internal Server Error oldal megtekintése.
GET	/404 <i>pageNotFoundMW</i>	Not Found oldal megtekintése.
GET	/	Átirányítás a <i>/home</i> oldalra

7. Middlewarek

Ebben a fejezetben az alkalmazásban használt middlewerek kerülnek ismertetésre.

authMW

Session alapján ellenőrzésre kerül hogy a felhasználó bejelentkezett-e már. Amennyiben igen akkor adatbázisból ellenőrzi user adatokat, amiket a `res.locals.user` -ben továbbít.

signupMW

Regisztrációs oldal renderelését végzi.

signuppostMW

Ezen keresztül kerül felküldésre a backendnek a regisztrációs kérelem. Ellenőrzésre kerül, hogy az email címmel már korábban került-e regisztráció. Validálást is végez a post adatokra, amennyiben validációs hiba van, akkor rendereli a signup képernyőt a validációs error-ok kijelzésével.

loginMW

Bejelentkezési oldal renderelését végzi.

loginpostMW

Ezen keresztül kerül felküldésre a backend-nek a bejelentkezési adatok. Az adatokra validációs ellenőrzést végez, ezt követően pedig autorizációs ellenőrzést végez. Amennyiben validációs hiba, vagy autorizációs hiba jelentkezik, akkor újraprendelni a bejelentkezési oldalt ezen error-ok kijelzésével.

logoutMW

A bejelentkezett felhasználó kijelentkezési kérelmét kezeli.

forgotMW

Elfelejtett jelszó oldal rendelelrsét végzi.

forgotpostMW

A felhasználó elfelejtett jelszó kérelmét kezeli. Amennyiben a rendszerben az email cím létezik, akkor email üzenetben elküldésre kerül az aktuális jelszó. Amennyiben az email cím a rendszer által nem ismert, akkor figyelmen kívül hagyódik a kérelem, hibaüzenet erről nem keletkezik.

twiteposttMW

Bejelentkezett felhasználó új twite írási kérelmét kezeli. Fogadja az adatokat, azon validációt végez. Amennyiben validációs hiba keletkezik, akkor újraprendereli az oldalt

a validációs error-ok kijelzésével. Amennyiben nem keletkezett validációs hiba, akkor a felhasználó új twite-je mentésre kerül az adatbázisba, majd a főoldalra átirányítást történik.

editTwiteMW

Felhasználó twite szerkesztésének oldalát rendeleleri.

edittwitePostMW

Felhasználó twite szerkesztési kérelmét kezeli. Validálást végez a post adatokra. Amennyiben vlidációs error keletkezik akkor újrarendelelreésre kerül az oldal a validációs hibák kijelzésével. Amennyiben nincs validációs hiba, akkor a felhasználó twite-jét módosítja az adatbázisban, majd a főoldalra irányít.

delTwiteMW

A felhasználó egy twite-jének törlését végzi az adatbázisban. Törlést követően a főoldalra irányít.

twittersMW

Twitter-ek listázás oldalát rendeleleri.

twitterMW

Egy kiválasztott Twitter-hez tartozó twite-k oldalát rendeleri.

internalServerErrorMW

Internal Server Error oldal rendelerését végzi.Szerver oldali hiba kijelzésére..

pageNotFoundMW

Page Not Found oldal rendelerését végzi. Kliens oldali hiba jelzésére.

8. Modellek

Users

- ID
 - firstName
 - lastName
 - email
 - password
- regisztrált felhasználók
 - UUID, egyedi azonosító
 - string, keresztnév
 - string, családnév
 - string, e-mail cím
 - string, jelszó

Twits

- ID
 - title
 - content
 - twitedBy
 - time
- posztok
 - UUID, egyedi azonosító
 - string, cím
 - string, tartalom
 - string, postoló
 - date, posztolás ideje