

# Tehnologii Web – Proiect

---

## Cuprins

1. Descrierea generala a proiectului.....	1
2. Tehnologiile utilizate.....	2
3. Structura datelor.....	3
4. Utilizarea inteligenței artificiale.....	4
5. Concluzii.....	5

## 1. Descrierea generala a proiectului

Proiectul urmareste crearea unei aplicații web destinata gestionării și vizualizării unei baze de date primita, cea din tema primita fiind Disney Characters (Tema 12). Aceasta oferă utilizatorului o platforma pentru a naviga prin personajele Disney, a căuta informații specifice și a filtra după filmele sau seriarele TV în care apar, iar utilizatorii, dupa crearea unui cont, pot adăuga, edita sau șterge înregistrări din baza de date.

Asadar, proiectul contine:

- Vizualizare dinamică: Afișarea unei liste complete de personaje Disney cu imagini și detalii
- Căutare și filtrare: Permitearea utilizatorilor să găsească personaje după nume, film sau serial TV
- Paginare, cu posibilitatea de a selecta numărul de personaje afișate pe pagină (5, 10, 15, 20 sau 30)
- Gestionare administrativă: Panou de control pentru adăugarea, editarea și ștergerea personajelor
- Securitate: Sistem de autentificare cu conturi utilizator protejate prin criptare

Pagina de Acasa (Homepage) are grid responsive și “cards” pentru personaje, funcție de căutare după nume, selectoare de filtru pentru filme și seriale TV si selector de pagină cu opțiuni pentru dimensiunea listei.

Pagina de detalii ale unui personaj contine informații complete despre un personaj, incluzand imaginea si lista de filme si seriale TV in care acesta apare, provenite din baza de date primita.

Panoul Administrativ (Admin Panel) contine un tabel cu toate personajele din baza de date, unde acestea pot fi cautate dupa nume, butoane pentru editare și ștergere și un form pentru adăugarea de noi personaje.

Sistemul de autentificare contine: pagina de logare și pagina de inregistrare. In plus, a fost implementat si un sistem de protecție cu sesiuni de 24 de ore.

## 2. Tehnologiile utilizate

### 2.1 Frontend (Interfața Utilizator)

#### Next.js 16.1.1 (Framework Principal)

Next.js este un framework React care face mai ușoară crearea aplicațiilor web moderne și performante. Aceasta tehnologie a fost aleasa pentru urmatoarele avantaje pe care le ofera: rutare pe bază de fișiere (file-based routing, nu trebuie configurat manual), suportă rendering pe server (SSR) pentru o performanță mai bună, include API Routes pentru backend direct în aplicație si are suport built-in pentru imagini optimizate și CSS Modules.

#### React 19.2.3

React este o bibliotecă JavaScript pentru crearea interfețelor interactive. Acesta permite crearea componentelor reutilizabile și gestionează starea aplicației eficient

#### CSS Modules

Acest sistem de stilizare CSS limitează domeniul de aplicare al stilurilor la o singură componentă, deci evită conflictele de clase CSS și permite organizarea codului.

### 2.2 Backend (Server și Logică)

#### Next.js API Routes

Sunt rute speciale din Next.js care permit crearea API-urilor direct în proiect

- Fișiere Principale:
  - app/api/characters/route.js - Gestionarea personajelor (GET, POST)
  - app/api/characters/[id]/route.js - Operații pe personaje specifice (GET, PUT, DELETE)
  - app/api/media/route.js - Obținerea listei de filme și seriale unice
  - app/api/auth/login/route.js - Autentificare utilizatori
  - app/api/auth/register/route.js - Înregistrare conturi noi
  - app/api/auth/logout/route.js - Logout utilizatori

## Middleware de Securitate

Fişier: middleware.js (interceptează fiecare request şi verifică dacă utilizatorul este autentificat)

- Permite acces la paginile publice (login, register)
- Redirecţionează utilizatorii neautentificaţi la login
- Protejează rutele administrative

## 2.3 Baza de Date

### SQLite cu better-sqlite3

Acesta face legatura, fără a necesita un server separat (nu necesită instalare sau configurarea a unui server de bază de date), stochează totul într-un singur fişier şi suportă toate operaţiile SQL standard.

Driver: better-sqlite3 - versiunea Node.js pentru SQLite

## 2.4 Autentificare şi Securitate

### bcryptjs

Este o bibliotecă pentru hashing (criptare unidirecţionată) a parolelor, astfel încât să se poată asigura un grad ridicat de securitate pentru utilizatorii aplicaţiei web.

### Session Cookies

- Durata: 24 de ore
- Securitate: HTTP-only cookies (nu pot fi accesate din JavaScript)
- Menţine utilizatorul logat pe toată sesiunea

## 3. Structura datelor

### 3.1 Tabelul Characters (Personaje)

Baza de date conţine un tabel characters cu următoarea structură:

Câmp	Tip	Descriere
id	INTEGER PRIMARY KEY	Identificator unic pentru fiecare personaj
name	TEXT	Numele personajului (ex: "Mickey Mouse")
description	TEXT	Descriere detaliată a personajului
imageUrl	TEXT	URL-ul imaginii de profil a personajului

<b>films</b>	TEXT (JSON)	Lista filmelor în care a apărut (format JSON)
<b>tvShows</b>	TEXT (JSON)	Lista serialelor TV în care a apărut (format JSON)
<b>createdAt</b>	DATETIME	Data și ora creării înregistrării
<b>updatedAt</b>	DATETIME	Data și ora ultimei modificări

### 3.2 Exemplu de Înregistrare

- {"id": 1, "name": "Mickey Mouse", "description": "Cel mai iconic personaj Disney, gânditor pozitiv și plin de energie", "imageUrl": "https://example.com/mickey.jpg", "films": ["Fantasia", "The Sorcerer's Apprentice"], "tvShows": ["Mickey Mouse Clubhouse"]}

### 3.3 Tabelul Users (Utilizatori)

Câmp	Tip	Descriere
<b>id</b>	INTEGER PRIMARY KEY	Identificator unic
<b>username</b>	TEXT UNIQUE	Nume de utilizator pentru login
<b>passwordHash</b>	TEXT	Hash bcrypt al parolei
<b>createdAt</b>	DATETIME	Data creării contului

## 4. Utilizarea inteligenței artificiale

Inteligența artificială sub forma agentului GitHub Copilot a ajutat în realizarea unui roadmap general al proiectului și în optimizarea sa. Pentru Roadmap-ul urmat, a propus o structură de proiect scalabilă folosind Next.js, a sugerat utilizarea SQLite pentru simplitate și performanța sa și a ajutat la alegerea tehnologiilor potrivite pentru proiectul dat. La sistemul de autentificare, a ajutat în criptarea parolelor cu bcryptjs, gestionarea sesiunilor cu cookie-uri și la middleware-ul de protecție a rutelor. De asemenea, a ajutat în gestionarea erorilor aparute pe parcurs, și a propus sugestii de optimizare care să aducă îmbunătățiri ale performanței.

## 5. Concluzii

Proiectul este o demonstrație practică a modului în care tehnologiile moderne pot fi combinate pentru a crea o aplicație web funcțională. Utilizarea Next.js, SQLite și CSS Modules au fost o alegere din punct de vedere al scalării și al performanței. Printre problemele întâmpinate se regăsesc următoarele:

- Dropdown-urile de filtrare apăreau în spatele card-urilor, rezolvată prin setarea z-index la 99999
- Filmele și seriile TV erau stocate ca JSON stringuri, rezolvată prin utilizarea `JSON.parse()` și filtrarea cu `.includes()` (structurile de date complexe necesită manipulare specială)

Folosind Next.js, SQLite și designul tematic, am construit o platformă care este atât funcțională și ușor de înțeles pentru utilizator. Colaborarea cu AI (GitHub Copilot) a ajutat dezvoltarea și a ajutat la identificarea de soluțiilor.