

Gruppuppgift - API-utveckling med säker hantering av användare

Deadline: Fredag den 16/5 kl 23:59

Grupp: 2-3 medlemmar.

Inlämning: Github repo länk som skall innehålla

- Export fil på er MongoDB tabeller,
- Express API:et
- Klienten.
- Dagbok om medlemmarnas ansvarsområden och hur det gick i grupparbetet

Github repot skall vara privat, och ni skall bjuda in mig som en collaborator via mitt github alias: **sibala**

Uppgiftsbeskrivning och krav

Ni skall skapa ett Book API som delvis är lösenordsskyddad.

1 - Tabeller i MongoDB

Ansvarsområde 1 - Tabellen "users" skall innehålla följande information

- username: String
- password: String
- is_admin: Boolean
- created_at: Date

Ansvarsområde 2 - Tabellen "books" skall innehålla följande information

- title: String
- description: String
- author: String
- genres: Array
- image: String
- published_year: Number
- reviews: [{ type: mongoose.Schema.Types.ObjectId, ref: "Reviews" }] // Se **kursvecka 8**
-> **föreläsning 17** med todos ihoplänkad med subtasks

Ansvarsområde 3 - Tabellen "reviews" skall innehålla följande information

- name: String
- content: String
- rating: Number // 1-5
- created_at: Date

2 - API endpoints

Ansvarsområde 1 - Skapa CRUD och inloggning för tabellen "users", med följande endpoints

- Hämta alla användare GET: <http://localhost:3000/users>
- Hämta enskild användare GET: <http://localhost:3000/users/:id>
- Uppdatera befintlig användare med PATCH: <http://localhost:3000/users/:id> (Token)
- Radera befintlig användare med DELETE: <http://localhost:3000/users/:id> (Token)

Inloggning:

- Logga in användare med POST: <http://localhost:3000/auth/login>
- Logga ut användare med POST: <http://localhost:3000/auth/logout>
- Registrera ny användare med POST: <http://localhost:3000/auth/register>

Ansvarsområde 2 - Skapa CRUD för tabellen "books", med följande endpoints

- Hämta alla böcker med GET: <http://localhost:3000/books>
- Hämta enskilt inlägg med tillhörande reviews, med GET: <http://localhost:3000/books/:id>
- Skapa nytt inlägg med POST: <http://localhost:3000/books> (Token)
- Uppdatera befintligt inlägg med PATCH: <http://localhost:3000/books/:id> (Token)
- Radera befintligt inlägg med DELETE: <http://localhost:3000/books/:id> (Token)

Ansvarsområde 3 - Skapa CRUD för tabellen "reviews", med följande endpoints

- Hämta alla reviews GET: <http://localhost:3000/users>
- Hämta enskild review GET: <http://localhost:3000/users/:id>
- Skapa ny review med POST: <http://localhost:3000/users>
- Uppdatera befintlig review med PATCH: <http://localhost:3000/users/:id> (Token)
- Radera befintlig review med DELETE: <http://localhost:3000/users/:id> (Token)

3 - Klientens olika sidor

Ansvarsområde 1 - Skall skapa följande sidor

- Registrera användare. Formulär med följande fält
 - username TEXTFÄLT
 - lösenord TEXTFÄLT
- Logga in användare. Formulär med följande fält:
 - username TEXTFÄLT
 - lösenord TEXTFÄLT
- I lösenordsskyddad adminpanel - Lista alla användare i tabellform
 - username, password, is_admin, created_at

Ansvarsområde 2 - Skall skapa följande sidor

- För besökare - Boklista som listar böcker med följande
 - Mindre Bild
 - Författare
 - År
 - genres
 - Titel
 - Länk navigerar till specifik bok sida
- I lösenordsskyddad adminpanel - Lista alla böcker i tabellform
 - title, author, genres, created_at
- I lösenordsskyddad adminpanel - Skapa ny bok med formulär som inkluderar alla fält för **books** i mongoDB

Ansvarsområde 3 - Skall skapa följande sidor

- Specifik bok sida. Listar följande
 - Större Bild
 - Författare
 - publiceringsår
 - titel
 - beskrivning
 - genres
 - Tillbaka - länk
- Specifik bok sida (samma sida som ovan, fast med en ny sektion under)
 - Skapa ny review tillhörande bok, via formulär som inkluderar alla fält för **reviews** i mongoDB
- Specifik bok sida (samma sida som ovan, fast med en ny sektion under)
 - Lista alla reviews tillhörande specifik bok, med följande data:
 - content, rating, name, created_at

Betyg (G/IG)

För G skall gruppen ha uppfyllt alla punkter i valda ansvarsområden.

Om ni är 3 medlemmar, då skall ni uppfylla alla 3 ansvarsområden.

Om ni endast är 2 medlemmar, då skall ni uppfylla alla punkter på ansvarsområde 1 & 2 (Ni får inte välja ansvarsområde 3 framför de andra ansvarsområdena).

Primärt betygsätts ni som en grupp. Ni får hjälpa med varandras ansvarsområden. Dock måste det framgå att alla i gruppen har varit aktiva via bidrag i github och dagbok.

Om samarbetet inte funkar väl av olika anledningar, kan ni få betygsättas individuellt efter vald ansvarsområde istället. G uppnås om man utfört kravet på vald ansvarsområde, vad gäller:

- Tabell skapandet
- Endpoints
- Klient

OBS 1: Det måste framgå i github vem som kodat vad. Med andra ord är det en förutsättning för alla att versionshantera korrekt, för att bli betygsatta.

OBS 2: Tänk på att jag skall kunna ladda ner allt i en och samma github repo och testköra er applikation med min egen MongoDB. Har ni skapat databasen och applikationen enligt uppgiften, då skall det funka utan problem. Om jag inte kan testköra er applikation får ni komplettering.