

Opdracht: Bouw een Evenementen-API

Doel van de Opdracht

Het doel van deze opdracht is om een RESTful API te bouwen waarmee gebruikers evenementen kunnen beheren. Denk aan evenementen zoals workshops, concerten, of sportwedstrijden. Je werkt met Node.js, Express en Mongoose. De projectstructuur is georganiseerd in **models**, **controllers** en **routes**, en de API wordt gedocumenteerd met **Swagger**. Tot slot **deploy** je de API op [Render.com](https://render.com), zodat deze publiek toegankelijk is.

Opdrachtvereisten

1. Functionaliteiten van de API

- **GET** /events : Haal alle evenementen op.
- **POST** /events : Voeg een nieuw evenement toe.
- **GET** /events/:id : Haal de details van een specifiek evenement op aan de hand van een ID.
- **PUT** /events/:id : Werk een evenement bij.
- **DELETE** /events/:id : Verwijder een evenement.

2. Projectstructuur Gebruik de volgende structuur:

```
├── models/  
│   └── Event.ts  
├── controllers/  
│   └── eventController.ts  
├── routes/  
│   └── eventRoutes.ts  
├── server.ts  
├── package.json  
└── swagger.ts
```

3. Databasemodel Maak een Mongoose-model voor een "Event" met de volgende velden:

- **name** (String, verplicht): Naam van het evenement.
- **date** (Date, verplicht): Datum van het evenement.
- **location** (String, verplicht): Locatie van het evenement.
- **description** (String, optioneel): Beschrijving van het evenement.
- **isFree** (Boolean, standaard: false): Of het evenement gratis is.

4. Swagger-documentatie

- Documenteer alle API-endpoints met Swagger.
- Maak een Swagger-document dat beschikbaar is via /api-docs .

5. Extra Vereisten

- Implementeer foutafhandeling voor niet-bestaande routes en validatie (bijvoorbeeld verplicht veld ontbreekt).
- Gebruik environment-variabelen voor database-instellingen en andere configuraties (bijvoorbeeld met dotenv).

6. Deployment

- **Deploy de API op [Render.com](https://render.com)**, zodat deze toegankelijk is via een publieke URL.
 - Gebruik **Render Secrets** om je environment-variabelen (zoals database-URL en andere configuraties) veilig op te slaan.
-

Eindproduct

Het eindproduct moet:

- Alle genoemde API-functionaliteiten bevatten.
 - Georganiseerd zijn volgens de gestructureerde opzet.
 - Beschikbare Swagger-documentatie hebben op `/api-docs`.
 - Werkende foutafhandeling en nette code bevatten.
 - Correct gedeployed zijn op [Render.com](https://render.com), met een werkende publieke URL.
-