

Seleksi Beasiswa Bootcamp dan Kerja di Sanbercode

Technical Skill Assessment: Project Backend dengan Laravel dan Frontend dengan Nextjs

Catatan Penting:

- Project wajib disimpan dalam repository github yang visibilty-nya **PRIVATE**. Jika tidak **private** maka akan **didiskualifikasi**
- Invite **abdulalim110** sebagai collaborator dalam github
- **Repository Project wajib ada dua** yaitu repo untuk **backend** dan repo untuk **frontend**
- Aktivitas **commit** akan menjadi nilai lebih
- Projectnya **wajib terdeploy** ke server (bisa menggunakan vercel, netlify, railway dsb.)

Kerjakanlah sesuai dengan kemampuan Anda saat ini semampu Anda sebuah project Project Backend dengan Laravel dan Frontend dengan Nextjs sebagaimana panduan berikut:

1. CRUD Book & Category

Buatlah CRUD Category dengan penamaan pathnya adalah categories

Dengan field pada tabel/data collection category adalah

- id (tipe data int)
- name(tipe data string)
- created_at (tipe data time)
- updated_at (tipe data time)

list path

berikut list path dari CRUD category(portnya tidak harus 8080):

- GET localhost:8080/categories (untuk menampilkan seluruh category)
- POST localhost:8080/categories(untuk membuat category baru)
- PATCH localhost:8080/categories/:id(untuk update category yang sudah ada berdasarkan id category)
- DELETE localhost:8080/categories/:id(untuk menghapus data category)
- GET localhost:8080/categories/:id/books(untuk menampilkan seluruh data books berdasarkan id category)

Buatlah CRUD Books dengan penamaan pathnya adalah books

Dengan field pada tabel/data collection book adalah

- id (tipe data int)
- title(tipe data string)
- description (tipe data string)
- image_url(tipe data string)
- release_year(tipe data int)
- price (tipe data string)
- total_page(tipe data int)
- thickness(tipe data string)
- created_at (tipe data time)
- updated_at (tipe data time)
- category_id (tipe data int - terhubung dengan category)

Untuk method POST dan PATCH user hanya bisa input data:

- title(tipe data string)
- description (tipe data string)
- image(tipe data string)
- release_year(tipe data int)
- price (tipe data string)
- total_page(tipe data int)
- category_id (tipe data int)

Validasi

Untuk release_year dibuatkan validasi bahwa inputan minimal 1980 dan maksimal adalah 2021

Konversi

thickness diinput berdasarkan total_page

Jika total_page kurang dari sama dengan 100 maka thickness-nya "tipis"

Jika total_page diantara 101 dan 200 maka thickness-nya "sedang"

Jika total_page lebih dari atau sama dengan 201 maka thickness "tebal"

List path books

berikut list path dari CRUD books(portnya tidak harus 8080):

- GET localhost:8080/books (untuk menampilkan seluruh book)
- POST localhost:8080/books (untuk membuat book baru)
- PATCH localhost:8080/books/:id (untuk update book yang sudah ada berdasarkan id book)
- DELETE localhost:8080/books/:id (untuk menghapus data book)

data harus tersimpan jadi ketika direfresh data masih ada,

display data bisa berupa tabel ataupun card list

2. Filter Book dan category book

Buatlah fitur filter pada path books dan categories/:id/books

Contoh path:

localhost:8080/books?sortByTitle=asc&minYear=2000&maxPage=150

localhost:8080/books?title=algoritma&minYear=2000&maxYear=2010

localhost:8080/categories/1/books?sortByTitle=desc&minYear=2000&maxPage=150

Parameter yang digunakan untuk filter adalah

- title(search berdasarkan title tidak perlu case sensitif)
- minYear (membuat filter minimum tahun)
- maxYear (membuat filter maksimum tahun)
- minPage (membuat filter minimum halaman buku)
- maxPage (membuat filter maksimum halaman buku)
- sortByTitle (untuk mengurutkan antara descending atau ascending berdasarkan title)

Perlu diperhatikan semua filter diatas bisa digunakan semua ataupun tidak, contoh jika minYear saja yang ada berarti di kondisikan untuk mencari buku dengan minimal tahun yang di cari dan sebagainya.

3. Authentication

buatlah authentication untuk login (jika ada register maka akan ada nilai plus). user yang tidak login tidak bisa create, delete ataupun update.

4. State Management

Pada Project Frontend Wajib menggunakan state management

5. Share Project

- Project wajib disimpan dalam repository github yang visibilty-nya **PRIVATE**. Jika tidak **private maka akan didiskualifikasi**
- Invite **abdulalim110** sebagai collaborator dalam github
- **Repository Project wajib ada dua** yaitu repo untuk **backend** dan repo untuk **frontend**
- Aktivitas **commit** akan menjadi nilai plus



Untuk Diperhatikan

Ketidaksesuaian dengan panduan berikut dapat menyebabkan diskualifikasi

1. Kerjakan test yang tersedia dalam link berikut hingga tenggat waktu yang telah ditentukan
 - Kerjakan sesuai perintah dan bahasa pemrograman yang diminta dalam tugas
 - Tugas tidak harus diselesaikan seluruhnya
 - Kerjakan sesuai kemampuan terjauh yang kamu kuasai saat ini
2. Apabila telah menyelesaikan task, silakan melaporkan melalui link berikut:
<https://bit.ly/testsbstanberLN1>
3. Projectnya wajib terdeploy ke server (bisa menggunakan vercel, netlify, railway dsb.)