

	<b>BRIEF DOCUMENT</b>	Document No.
		N/A

Project	Golang	Date	N/A
Revision No.	N/A	Revision Date	N/A
Reference No.	N/A	Assignee	N/A
Mandays	24 Hours	Target Release	N/A

Brief Description
<p>Mini Project:</p> <ol style="list-style-type: none"> <li>1. Membuat sebuah project untuk input data buku dan kategorinya</li> <li>2. Database yang digunakan adalah relational database diantaranya mysql atau postgresql</li> <li>3. Deploy production project di railway</li> </ol> <p>Mohon diperhatikan untuk mengikuti instruksi yang diberikan.</p>

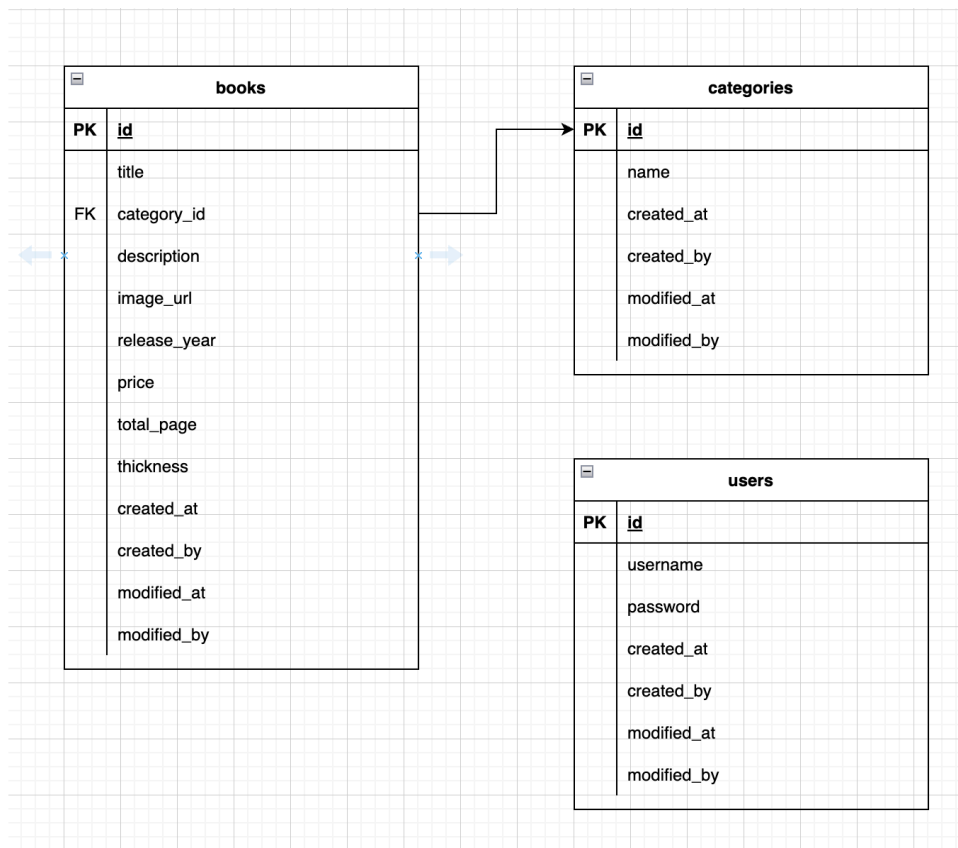
Setup Project
<p>Kerjakan dalam repository git (github) dengan ketentuan:</p> <ol style="list-style-type: none"> <li>1. Boleh menggunakan repository sebelumnya yg digunakan untuk tugas harian</li> <li>2. Boleh membuat repository baru untuk quiz</li> </ol> <p>Struktur folder diserahkan ke peserta dengan ketentuan:</p> <ol style="list-style-type: none"> <li>1. Wajib menerapkan package dan import</li> </ol> <p>Package yang disarankan:</p> <ol style="list-style-type: none"> <li>1. <a href="https://github.com/gin-gonic/gin">https://github.com/gin-gonic/gin</a></li> <li>2. <a href="https://github.com/lib/pq">https://github.com/lib/pq</a></li> </ol>

## Setup Project

3. <https://github.com/rubenv/sql-migrate>

Lalu init project dengan go modules.

## Database Design



Tabel Buku

Name	Data Type
id	integer
title	varchar
description	varchar
image_url	varchar

## Database Design

release_year	integer
price	integer
total_page	integer
thickness	varchar
category_id	integer
created_at	timestamp
created_by	varchar
modified_at	timestamp
modified_by	timestamp

Tabel Kategori

Name	Data Type
id	integer
name	varchar
created_at	timestamp
created_by	varchar
modified_at	timestamp
modified_by	timestamp

Tabel User

Name	Data Type
id	integer
username	varchar
password	varchar
created_at	timestamp
created_by	varchar
modified_at	timestamp

## Database Design

modified\_by

timestamp

## Detail Feature

### Authentication

Buatlah salah satu authentication untuk mengakses api dari project ini mengikuti ketentuan berikut.

1. Middleware Basic Auth
2. Middleware JSON Web Token

Untuk yang memilih menggunakan JSON Web Token, perlu membuat api untuk user login dengan path "api/users/login"

## Detail Feature

### Kategori

Buatlah api untuk mengelola kategori buku mengikuti ketentuan berikut ini.

1. Penamaan path nya menggunakan "api/categories"

Detail path:

#### 1. **localhost:8080/api/categories**

Method yang digunakan adalah get.

Api ini digunakan untuk menampilkan seluruh kategori

#### 2. **localhost:8080/api/categories**

Method yang digunakan adalah post.

Api ini digunakan untuk menambahkan kategori.

#### 3. **localhost:8080/api/categories/:id**

Method yang digunakan adalah get.

Api ini digunakan untuk menampilkan detail kategori.

### Detail Feature

#### Kategori

#### 4. **localhost:8080/api/categories/:id**

Method yang digunakan adalah delete.

Api ini digunakan untuk menghapus kategori.

#### 5. **localhost:8080/api/categories/:id/books**

Method yang digunakan adalah get.

Api ini digunakan untuk menampilkan buku yang tersedia berdasarkan kategori tertentu.

Buatlah validasi untuk api kategori mengikuti ketentuan berikut.

1. Gunakan middleware basic auth / jwt untuk mengakses api kategori.
2. Berikan message error jika menghapus data kategori yang tidak tersedia.
3. Berikan message error jika mengupdate data kategori yang tidak tersedia.

### Detail Feature

#### Buku

Buatlah api untuk mengelola buku mengikuti ketentuan berikut ini.

1. Penamaan path nya menggunakan "api/books"

Detail path:

#### 1. **localhost:8080/api/books**

Method yang digunakan adalah get.

Api ini digunakan untuk menampilkan seluruh buku

#### 2. **localhost:8080/api/books**

Method yang digunakan adalah post.

Api ini digunakan untuk menambahkan buku.

#### 3. **localhost:8080/api/books/:id**

Method yang digunakan adalah get.

Api ini digunakan untuk menampilkan detail buku..

## Detail Feature

### Buku

#### 4. `localhost:8080/api/books/:id`

Method yang digunakan adalah delete.

Api ini digunakan untuk menghapus buku.

Buatlah validasi untuk api buku mengikuti ketentuan berikut.

1. Gunakan middleware basic auth / jwt untuk mengakses api buku..
2. Berikan message error jika menghapus data buku yang tidak tersedia.
3. Berikan message error jika mengupdate data buku yang tidak tersedia.
4. Berikan validasi untuk memastikan bahwa `release_year` hanya berisi inputan minimal 1980 dan maksimal 2024.

Buatlah logic konversi saat input data buku mengikuti ketentuan berikut.

1. Tidak semua yang ada di kolom tabel buku diinputkan melalui object json, ada beberapa kolom yang harus dikonversi terlebih dahulu.
2. Untuk kolom **thickness** didapatkan melalui konversi berdasarkan total page. Jika total page nya melebihi 100 maka kolom thickness diisi "tebal". Jika total page nya kurang dari 100 maka kolom thickness diisi "tipis".

## Deployment

Deploy project ke railway.

## Documentation

Dokumentasikan cara penggunaan, kegunaan dan list path yang tersedia di project ini dengan memodifikasi file `readme.md` dan jika dirasa perlu, bisa ditambahkan swagger docs.