

# Scalable Web Service With Golang

Sesi 13 : Live Coding dan Coaching Session

**ALBERT MARIO - 20 MARET 2024**

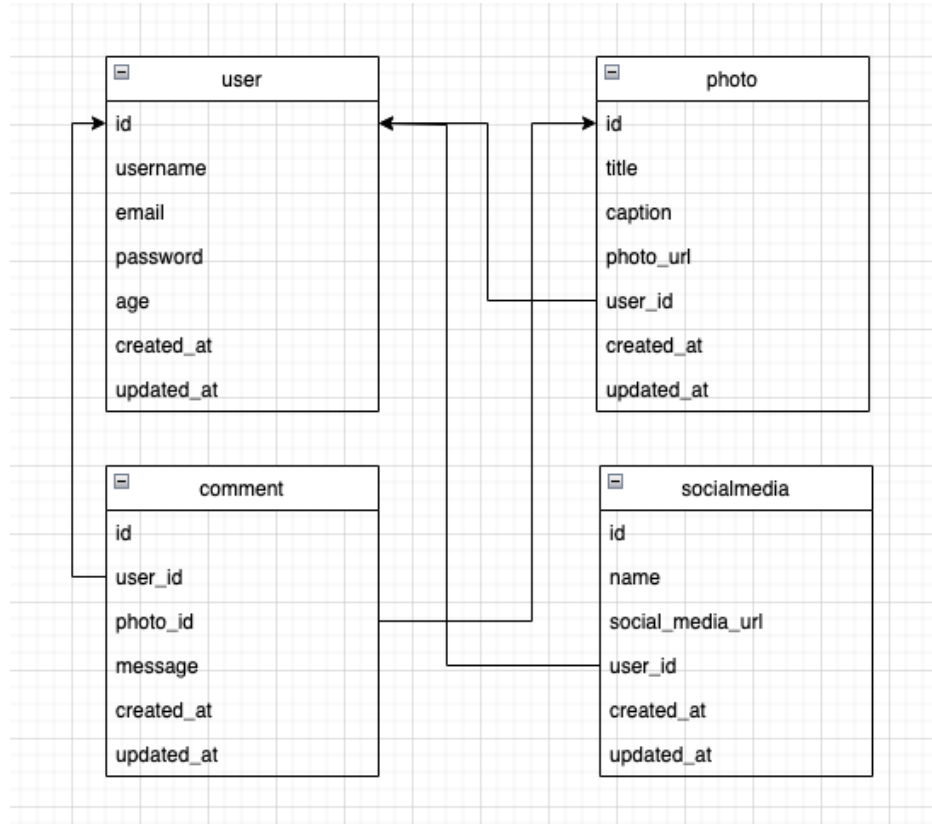


# Final Project

Pada final project ini, kalian akan diminta untuk membuat suatu aplikasi bernama MyGram, yang dimana pada aplikasi ini kalian dapat menyimpan foto maupun membuat comment untuk foto orang lain. Aplikasi ini akan dilengkapi dengan proses CRUD dengan table dan alur yang akan dijelaskan berikut ini:

1. Project ini bebas dikerjakan dengan library apapun. Namun agar proses pengerjaannya lebih cepat dan mudah, disarankan untuk menggunakan framework Gin Gonic dan orm Gorm.
2. Berikut merupakan library/package yang wajib digunakan
  - [github.com/dgrijalva/jwt-go](https://github.com/dgrijalva/jwt-go)
  - [golang.org/x/crypto](https://golang.org/x/crypto)
1. Dalam project ini akan memerlukan 4 table. Berikut merupakan table-table dan field-field yang diperlukan dalam project ini (detail table pada slide berikutnya).

# Project MyGram



Keempat table tersebut harus mempunyai validasi-validasi pada tiap field-field nya. Validasi boleh dibuat sendiri ataupun menggunakan package seperti Go Validator. Berikut merupakan penjelasan validasinya :

1. Field email

- Validasi pengecekan format email yang valid
- Validasi agar dapat menjadi unique index
- Validasi agar field email tidak boleh kosong atau harus terisi

2. Field username

- Validasi agar dapat menjadi unique index
- Validasi agar field username tidak boleh kosong atau harus terisi

3. Field password

- Validasi agar field password tidak boleh kosong
- Validasi agar field password minimal memiliki panjang sebanyak 6 karakter

4. Field age

- Validasi agar field age tidak boleh kosong atau harus terisi
- Validasi agar field age minimal memiliki nilai diatas 8

Keempat table tersebut harus mempunyai validasi-validasi pada tiap field-field nya. Validasi boleh dibuat sendiri ataupun menggunakan package seperti Go Validator. Berikut merupakan penjelasan validasinya :

### **A. Validasi untuk table User :**

1. Field email
  - Validasi pengecekan format email yang valid
  - Validasi agar dapat menjadi unique index
  - Validasi agar field email tidak boleh kosong atau harus terisi
2. Field username
  - Validasi agar dapat menjadi unique index
  - Validasi agar field username tidak boleh kosong atau harus terisi
3. Field password
  - Validasi agar field password tidak boleh kosong
  - Validasi agar field password minimal memiliki panjang sebanyak 6 karakter
4. Field age
  - Validasi agar field age tidak boleh kosong atau harus terisi
  - Validasi agar field age minimal memiliki nilai diatas 8

**B. Validasi untuk  
table photo :**

1. Field title
  - Validasi agar field title tidak boleh kosong
2. Field photo\_url
  - Validasi agar field photo\_url tidak boleh kosong atau harus terisi

**B. Validasi untuk  
table social  
media :**

1. Field name
  - Validasi agar field name tidak boleh kosong
2. Field social\_media\_url
  - Validasi agar field social\_media\_url tidak boleh kosong atau harus terisi

## *Final Project*

# **Project MyGram**



### **B. Validasi untuk table comment :**

1. Field message
  - Validasi agar field message tidak boleh kosong

### **Berikut merupakan alur dari aplikasinya:**

- Endpoint-endpoint untuk mengakses data pada table SocialMedia, Photo, dan Comment harus melalui proses autentikasi terlebih dahulu, dan proses autentitasinya wajib menggunakan JsonWebToken.
- Untuk endpoint-endpoint yang berguna untuk memodifikasi data kepemilikan seperti **Update** atau **delete** maka harus melalui proses otorisasi.



Endpoint-endpoint yang harus dibuat :

1. User :

- Register [POST]
- Login [POST]

1. Photo :

- GetAll [GET]
- GetOne [GET]
- CreatePhoto [POST]
- UpdatePhoto [PUT]
- DeletePhoto [DELETE]

1. Comment :

- GetAll [GET]
- GetOne [GET]
- CreateComment [POST]
- UpdateComment [PUT]
- DeleteComment [DELETE]

4. Social Media :

- GetAll [GET]
- GetOne [GET]
- CreateSocialMedia [POST]
- UpdateSocialMedia [PUT]
- DeleteSocialMedia [DELETE]

## *Final Project*

# **Project MyGram**



Seluruh routing endpoint diatas harus diikuti dengan betul

Seluruh request body, headers maupun request params harus diikuti dengan betul.

Seluruh response status, maupun response data nya harus diikuti dengan betul.

Mohon untuk diperhatikan notes-notes yang telah diberikan diatas seperti endpoint-endpoint yang harus melewati proses autentikasi dan yang tidak. Begitu juga dengan proses autorisasinya.

Perlu diingat disini bahwa proses autorisasi dilakukan setelah proses autentikasi, bukan sebaliknya.

- **Tidak ada materi khusus hari ini.**
- **Silahkan teman-teman melanjutkan mengerjakan final projectnya.**
- **Jika ada pertanyaan mengenai apa pun (final project, assignment lain, golang, software engineering, atau tech career) bisa langsung bertanya saja.**

Tips mengerjakan final project

1. Pahami konteks bisnisnya.
2. Buat model databasenya, bagaimana interaksi antar entity dalam sistem.
  - a. (Opsional )Buat model data untuk masing - masing request dan response agar lebih mudah
3. Buat repository layer dan query ke database.
4. Buat controller layer untuk user.
5. Buat mekanisme autentikasi.
  - a. Tetapkan data payload / claims yang akan diletakkan.
  - b. Middleware
6. Buat controller layer untuk flow lainnya.
  - a. Pastikan autentifikasi dan otorisasinya diletakkan di flow yang tepat
7. Buat validasi request.



# Thank You

---

**PT Hacktivate Teknologi Indonesia**

Gedung Aquarius Pondok Indah  
Jalan Sultan Iskandar Muda No.7  
Kebayoran Lama, Jakarta Selatan

*[www.hacktiv8.com](http://www.hacktiv8.com)*

