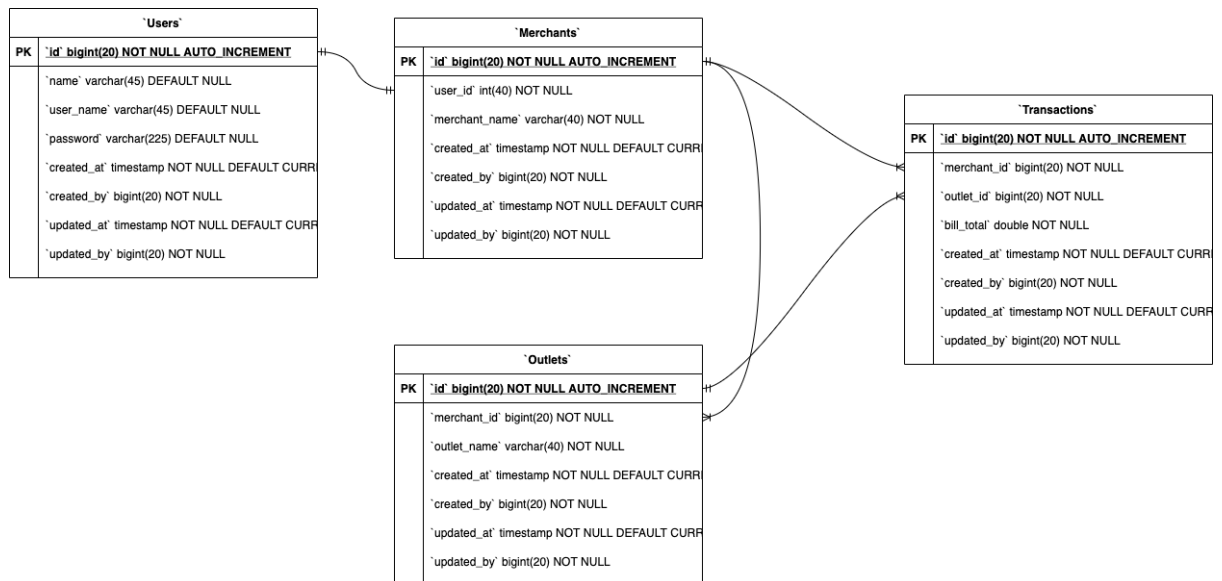


Test Case : Backend

1. Terdapat ERD seperti berikut:



Dalam team ini anda ditugaskan membuat API fungsi-fungsi berikut:

- Membuat fungsi login (5 point)
- Untuk authorization pada point A gunakan JWT (6 point)
- Laporan **nama merchant, omzet** per hari dalam pada bulan november mulai tanggal 1 sampai dengan tanggal 30 dengan pagination. Apabila tidak ada transaksi pada tanggal itu **omzet** akan bernilai 0 (6 point)
- API untuk menampilkan laporan **nama merchant, nama outlet, omzet** per hari pada bulan november mulai tanggal 1 sampai dengan tanggal 30 dengan pagination. Apabila tidak ada transaksi pada tanggal itu **omzet** akan bernilai 0 (6 point)
- Pada poin C pastikan user tidak bisa melakukan akses pada merchant_id yang bukan miliknya (10 point)
- Pada poin D pastikan user tidak bisa melakukan akses laporan pada outlet_id yang bukan miliknya (5 point)
- Dari test case pada point C dan point D, apakah struktur ERD yang dibentuk sudah optimal ? berikan penjelasannya (9 point)
- Dokumen teknis Data Manipulation Language (DML) (3 point)

Ketentuan Coding pekerjaan

- Dikerjakan dengan menggunakan Golang (*Recommended*)

- b. Untuk PHP menggunakan framework CodeIgniter/Laravel
- c. Gunakan best practice Rest API untuk menentukan Response Code

Keterangan :

- Omzet adalah jumlah dari bill total
- Download architecture diatas di [pastebin](#)

2. Terdapat block code sbb:

Terdapat model area dengan struct dibawah ini

```
type (
    Area struct {
        ID int64          `gorm:"column:id;primaryKey;"`
        AreaValue int64    `gorm:"column:area_value"`
        AreaType string    `gorm:"column:type"`
    }
)
```

Handler memiliki function seperti dibawah ini

```
func (_r *AreaRepository) InsertArea(param1 int32, param2 int64, type []string, ar *Model.Area)
(err error) {
    inst := _r.DB.Model(ar)
    Var area int
    area = 0
    switch type {
        case 'persegi panjang':
            var area := param1 * param2
            ar.AreaValue = area
            ar.AreaType = 'persegi panjang'
            err = _r.DB.create(&ar).Error
            if err != nil {
                return err
            }
        case 'persegi':
            var area = param1 * param2
            ar.AreaValue = area
            ar.AreaType = 'persegi'
            err = _r.DB.create(&ar).Error
            if err != nil {
                return err
            }
        case segitiga:
```

```

        area = 0.5 * (param1 * param2)
        ar.AreaValue = area
        ar.AreaType = 'segitiga'
        err = _r.DB.create(&ar).Error
        if err != nil {
            return err
        }
    default:
        ar.AreaValue = 0
        ar.AreaType = 'undefined data'
        err = _r.DB.create(&ar).Error
        if err != nil {
            return err
        }
    }
}

```

Dan dipanggil dengan cara berikut

```

err = _u.repository.InsertArea(10, 10, 'persegi')
if err != nil {
    log.Error().Msg(err.Error())
    err = errors.New(en.ERROR_DATABASE)
    return err
}

```

Dari keterangan diatas, apa saja yang harus diperbaiki dan berikan contoh konkritnya (25 point)

3. Buatkan flowchart dan pseudocodenya untuk menampilkan deret bilangan dengan inputan deret pertama, deret kedua dan nilai bilangan ke x (15 point)

Contoh kasus pertama:

Inputan deret pertama = 2

Inputan deret kedua = 4

Value x = 5

Maka akan menghasilkan deret angka 2,4,6,8,10

Contoh kasus kedua:

Inputan deret pertama = 5

Inputan deret kedua = 8

Value x = 7

Maka akan menghasilkan deret angka 5,8,11,14,17,20,23

4. Terdapat variabel array dengan nilai [4, -7, -5, 3, 3.3, 9, 0, 10, 0.2]. Buatlah flowchart dan pseudocode untuk melakukan logical sorting variabel tersebut baik secara ascending maupun descending (10 point)

Hasil yang diharapkan ketika sorting secara ascending

[-7, -5, 0, 0.2, 3, 3.3, 4, 9, 10]

Hasil yang diharapkan ketika sorting secara descending

[10, 9, 4, 3.3, 3, 0.2, 0, -5, -7]

Ketentuan Test Case

- Lama Pengerjaan : 48 jam dan kandidat diwajibkan mengirimkan hasil test case maksimal 49 jam setelah email test case dikirim, Kandidat akan mendapatkan nilai lebih jika mengirimkan sebelum batas waktu maksimal.
- Kandidat mengirim hasil Test Case Backend ke email join@majoo.id dengan subject Jawaban Test Case Backend majoo_(Nama) cc email ke sasmito@majoo.id, bayu.angga@majoo.id, keni@majoo.id, danny@majoo.id, yoka@majoo.id dan indriarko@majoo.id

SELAMAT MENGERJAKAN