Nama Peserta : Yosianus Antonio

Kode Peserta : FSDO002ONL007

Final Project Web Api Service, Infrastructure:

1. Language = C# dotnet, EntityFrameWokCore
2. Database = MySQL
3. Hosting = Heroku + Docker (Environment C# dotnet 5.0)
4. Security = JWT with Refresh token
5. Builpackage = https://buildpackregistry.s3.amazonaws.com/buildpacks/jincod/dotnetcore.tgz

Payment Api adalah sebuah web service API yang menyediakan layanan untuk payment anda menggunakan JWT authentication, untuk membuild app.

langkah yang perlu di ketahui untuk menggunakan service ini menggunakan Postman adalah:

1. register untuk user baru untuk dapat login dan mendapatkan token dan refreh token.

Graphical user interface, text, application, email

Description automatically generated - register: [POST]<https://payment-a.herokuapp.com/api/AuthManagement/Register>

Pada service register sedikit ada perubahan pada respon, saya beri pemberitahuan bahwa register sukses dan haruskan untuk login jika ingin mendapat token.

return new OkObjectResult(new { Register = "Success, Please Login", Validation = true });

Graphical user interface, text, application, email

Description automatically generated - login: [POST]<https://payment-a.herokuapp.com/api/AuthManagement/Login>

Pada Login kita masukan 2 parameter yaitu email dan password yang jika berhasil akan memberi informasi login success, token & juga RefreshTokennya

2. simpan masukan token kedalam Authorization untuk mengakses data dari pembayaran

Pada Route: /api/payment kita perlu memasukan Token ke dalam header dengan key Authorization yang berada pada header postman dengan Bearer [Token]

Graphical user interface, text, application, email

Description automatically generated a. Get data All : [GET]<https://payment-a.herokuapp.com/api/payment>

Graphical user interface, text, application, email

Description automatically generatedb. Post data: [POST]<https://payment-a.herokuapp.com/api/payment>

pada menu push data kita perlu memasukan semua data kecuali paymendDetailId dan saya menambahkan content untuk response feedback

{

  "cardOwnerName": "Andrea",

  "cardNumber": "453452",

  "expirationDate": "2023-12-01",

  "securityCode": "asdkad"

}

Tambahan untuk informasi saat response ok:

return new OkObjectResult(new { Success = "Insert Success", Validation = true, existItem });

Graphical user interface, text, application, email

Description automatically generatedc. Update Data: [PUT] <https://payment-a.herokuapp.com/api/payment/3>

pada method put akan mengedit data dalam database dengan memberikan inputan yang akan diubah sesuai dengan id pada data dan url

{

            "paymentDetailId": 3,

            "cardOwnerName": "Andrea",

            "cardNumber": "453452",

            "expirationDate": "2023-12-01T00:00:00",

            "securityCode": "apa aja"

        }

Tambahan untuk informasi saat response ok:

return new OkObjectResult(new { Success = $"Updated data with id {item.paymentDetailId}", Validation = true, item });

Graphical user interface, text, application, email

Description automatically generatedd. Get Data by Id: [GET]<https://payment-a.herokuapp.com/api/payment/3>

Pada method Get by

{

            "paymentDetailId": 3,

            "cardOwnerName": "Andrea",

            "cardNumber": "453452",

            "expirationDate": "2023-12-01T00:00:00",

            "securityCode": "apa aja"

        }

Tambahan untuk informasi saat response ok:

return new OkObjectResult(new { Success = $"Get data with id {item.paymentDetailId}", Validation = true, item });

Graphical user interface, text, application, email

Description automatically generatedd. Delete Data: [DELETE]<https://payment-a.herokuapp.com/api/payment/4>

Method delete akan menghapus sesuai parameter id yang kita masukan pada url di atas

3. Jika token exp maka masukan RefreshToken yang di berikan saat login untuk mendapat token terbaru.

Graphical user interface, text, application, email

Description automatically generated URL: []<https://payment-a.herokuapp.com/api/AuthManagement/RefreshToken>

Refresh Token digunakan untuk memperbarui token yang telah habis expirationnya untuk menjaga keamanan data dan akases semakin singkat semakin aman.

Parameter input dari Refresh Token adalah :

1. Previous Token
2. Refresh Token (Saat Login)

Selamat mengguanakan :D !!!!