Ketentuan dan persyaratan :

1. Waktu Pengerjaan 120 Menit

2. Hasil jawaban di simpan di repository online pilih salah satu diantara github, gitlab atau bitbucket. (hanya kirimkan link repository nya saja).
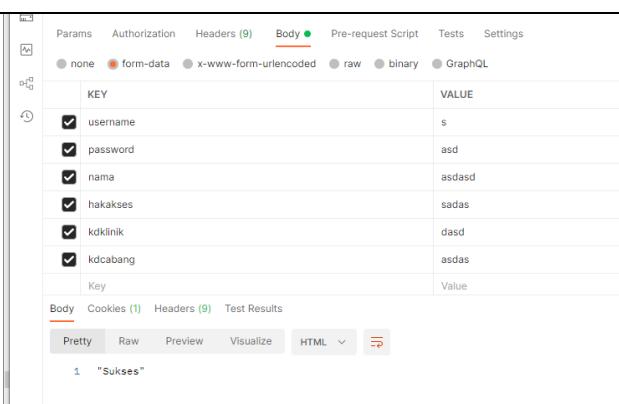
A. SOAL BACKEND DENGAN MENGGUNAKAN PHP (LARAVEL) / NodeJS
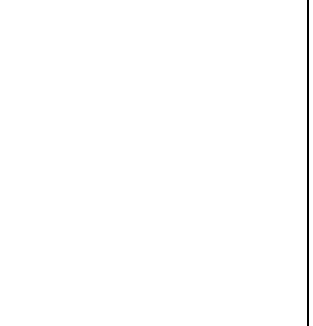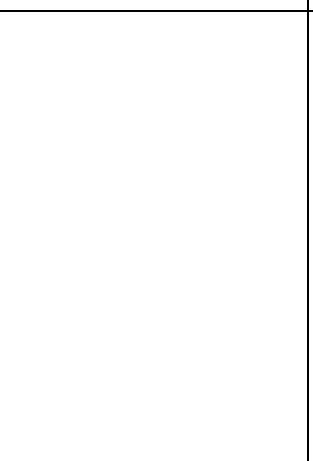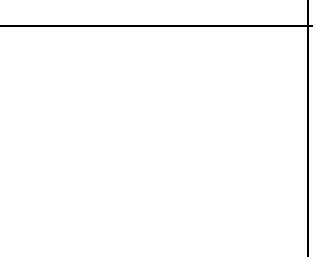
1. Buat table (nama bebas)

2. Buat Rest API

- (Metode Testing Pada Postman atau yang lain, disertakan dengan sample pengujian)

JENIS	JSON RESPONE
INSERT	

UPDATE	
DELETE	
READ	

3. Soal Ekripsi Data Backend

Buatkan login dengan table yang sudah di buat dengan hasil yang di ekripsi aes-256-CBC Dan token jwt yang di hasilkan sudah tersimpan di cokiee yang tersimpan 1 hari

Untuk melihat hasil enkripsi silahkan buatkan api untuk decrypt

POST ∨ | http://localhost:3000/api/decrypt | **Send** ∨

Params Authorization Headers (10) Body ● Scripts Settings   **Cookies**
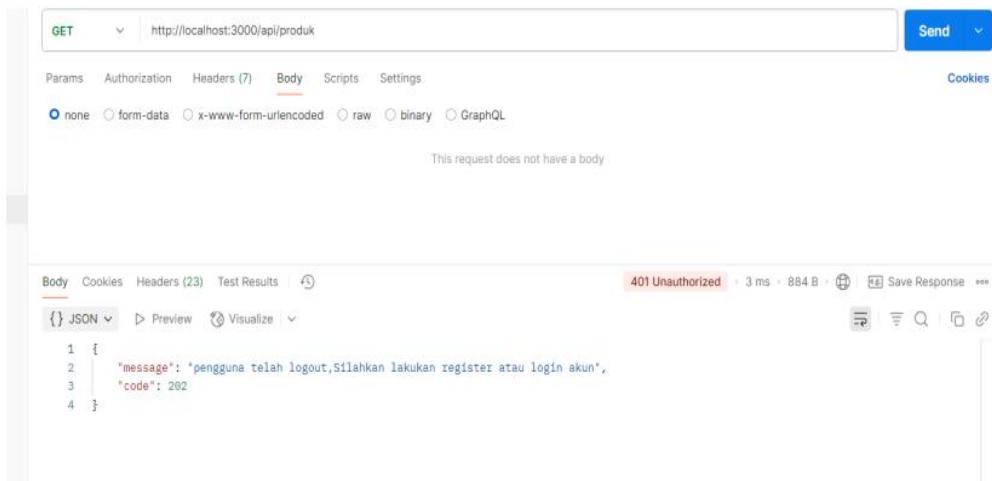
○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON ∨   **Beautify**

1 {"encryptedData":"84e2039eb26d87c577b44f62a8fc39e8:1ff3a663c4f3fd7833d7ab471c64ee33effe4bc82eecebeece4f8a4bc6b6e0d7b0969f490842f3aa7dbbec14a5
   e6c5716196d3ef217d1e69c64525ab15ae3313430cf6ce2d4f091b9fe483f87895d8dc99907f8bdbd8c8f404e1c26cb6ea04076a871a45f4693cef2b43e56d04f7bdfe19b
   5f293e81418e54e3d933ad5d94bb1da9992d140e39bbb8eaea1c62b878ab2d53f04fe4f72c69712224d0937d96611a35fd5299e35ea7eaab8a825d13da42608a0e4dd77f2
   785f446412552ba36e785b281b7c498ff0b46129117e5c9cdff45d337d8289e2fb417b08e05171984c9e31e75991c8e428f33641d96e02552b94b0292d10b3bdacca7252b
   21474b1a535fd36b3e873e0f08c386fb28e2dc6413c419d83e71a6c87d8809ea502e6171897"}

Body Cookies (1) Headers (23) Test Results ⟲   200 OK · 5 ms · 1.06 KB · ⊕ · ⟦⟧ Save Response ···

{} JSON ∨   ▷ Preview   ⟲ Visualize ∨   ⇥ ⇥ Q ⟦ ⧉

```
 1  {
 2      "data": [
 3          {
 4              "iduser": 18,
 5              "username": "johndoe",
 6              "nama": "John Doe",
 7              "email": "johndoe@example.com",
 8              "hp": "081234567890",
 9              "alamat": "Jl. Sudirman No.10, Jakarta",
10              "password": "$2a$10$BigbLaohJx671mrEvZ8b9u5Ui1C3l9pqCJu9BUBAXKILRTA3yaxzC",
11              "aktif": 1,
12              "statususer": "1",
13              "foto": "https://example.com/profile.jpg"
14          }
15      ]
16  }
```

3. Memasang midlleware validasi Ketika user sudah logout

GET ∨ | http://localhost:3000/api/produk | **Send** ∨

Params Authorization Headers (7) Body Scripts Settings   **Cookies**

● none ○ form-data ○ x-www-form-urlencoded ○ raw ○ binary ○ GraphQL

This request does not have a body

Body Cookies Headers (23) Test Results ⟲   401 Unauthorized · 3 ms · 884 B · ⊕ · ⟦⟧ Save Response ···

{} JSON ∨   ▷ Preview   ⟲ Visualize ∨   ⇥ ⇥ Q ⟦ ⧉

```
 1  {
 2      "message": "pengguna telah logout,Silahkan lakukan register atau login akun",
 3      "code": 202
 4  }
```

Menampilkan data dengan token yang sudah berhasil ada di cookie

GET   http://localhost:3000/api/produk   Send

Params Authorization Headers (8) Body Scripts Settings   Cookie
Headers   8 hidden

Key	Value	Description
Key	Value	Description

Body Cookies (1) Headers (23) Test Results   200 OK · 5 ms · 2.38 KB · Save Response

{} JSON ∨ ▷ Preview Visualize ∨

1 {
2   "data":
      "f1c59878e6e0a6ce997290b9eabde495:72d9ac16649e9c801114838b5bfe5e1b372690f1ef3209e671b6a94a5c45de4ee421547b63f7c165631bcf7f989689aed
      b0c9c25d59dbd05e3e25278b3c19fb8b2d7debfff0cb67b6f82f7bbabbd5e7e7c07ecfa25d893bedb92616c726b3b668f1a15bead76018e1cf6d0da6fca8b0fb3ec
      a834f818f17d126a09621d6a889c5d35384d4ec19e8d26cb1c02a1ab5600d3891e7846d4693a934038ae669bc08b21a7489c819901bb47b3b1361e2d9a22115852d
      641d9655787946b6fb9305a434957dcbb425ecd72ef5e6cf47b6df9a482336a808c0bd2537b2c2ad494c353bb1bfafe44605fa37ed6fed0b16eab45498c736401b1
      99153de83cdab3cf7e27dd59ae23703af5a433394a7e639d3b768902f7d4831e89f550230baba0d35717ff0b35c14acd1508931e252bdb7fbc4b2878f8fb6249df2
      ba292184d2e75329bb29f631e01326285fab9f726e03dfb0b77a36be7d374ff9f9aeae82d494ba17d90ba96b2c4a584d350398762105c64dd8b45c01813ddabb270
      d7dd9b49df5391c247aebbbc8d2187ea8acd120a3252ceb902200d51db264b2641074514d19c9ce0ca51d159bd2d3388f1eff20466b0df7590a9ace4348456364dc
      8eba7322d01e71e58be00905c5938b19f0b862289e4564804afae7114e8ac6240b458bd48c5ae652e85d7294d667954ccb72aee2733d447917ca2195fd467774f91
      814b744d6689d5c53c1f20c86f63bfdb6ae7694313a4f1f8c01d3b0cce487f1041c2474664374de0583a661d98d1e69d36370d5b0915b47b079714ffde9241464e2
      c89b1e2a74b2456553c3ead577f8650ffee9f023119ea170e321eabcccdbd3e6ea5a5c0334838325bf939566aeeb405cafbac732a49229d54a69cfc5650009ff9d8
      8976348413d1563c400a24a43773e379cb1f04a2501cb54394140d5ba9d84284b27a5e054c901eae2b82e7828bc6354e6b7b8dca754af544bb79852b24a1a60978c
      ecbadc4718842818dda6586088b6df74ac5c4de648e92eb842a1279d229addf4669db38cfba1245f6221f9dfdd4cc2c92978ef92c17d6089e3855d0a682b387b7ef
      5713b12f6843f958ba83dbad9d22caf9769a57d632238efc46155d695c08e6"
3 }

POST   http://localhost:3000/api/decrypt   Send

Params Authorization Headers (10) Body ● Scripts Settings   Co
○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON ∨   Bea

1 {"encryptedData":"f1c59878e6e0a6ce997290b9eabde495:72d9ac16649e9c801114838b5bfe5e1b372690f1ef3209e671b6a94a5c45de4ee421547b63f7c165631bcf7
  9689aedb0c9c25d59dbd05e3e25278b3c19fb8b2d7debfff0cb67b6f82f7bbabbd5e7e7c07ecfa25d893bedb92616c726b3b668f1a15bead76018e1cf6d0da6fca8b0f
  ca834f818f17d126a89621d6a889c5d35384d4ec19e8d26cb1c02a1ab5600d3891e7846d4693a934038ae669c08b21a7489c819901bb47b3b1361e2d9a22115852d64
  65578794b6fb9305a434957dcbb425ecd72ef5e6cf47b6df9a482336a808c0bd2537b2c2ad494c353bb1bfafe44605fa37ed6fed0b16eab45498c736401b199153de8
  ab3cf7e27dd59ae23703af5a433394a7e639d3b768902f7d4831e89f550230baba0d35717ff0b35c14acd1508931e252bdb7fbc4b2878f8fb6249df2ba292184d2e757
  b29f631e01326285fab9f726e03dfb0b77a36be7d374ff9f9aeae82d494ba17d90ba96b2c4a584d350398762105c64dd8b45c01813ddabb270d7dd9b49df5391c247ae
  c8d2187ea8acd120a3252ceb902200d51db264b2641074514d19c9ce0ca51d159bd2d3388f1eff20466b0df7590a9ace4348456364dc8eba7322d01e71e58be00905c5
  b19f0b862289e4564804afae7114e8ac6240b458bd48c5ae652e85d7294d667954ccb72aee2733d447917ca2195fd467774f91814b744d6689d5c53c1f20c86f63bfdb
  7694313a4f1f8c01d3b0cce487f1041c2474664374de0583a661d98d1e69d36370d5b0915b47b079714ffde9241464e2c89b1e2a74b2456553c3ead577f8650ffee9f0
  19ea170e321eabcccdbd3e6ea5a5c0334838325bf939566aeeb405cafbac732a49229d54a69cfc5650009ff9d88976348413d1563c400a24a43773e379cb1f04a2501c
  39414d05ba9d84284b27a5e054c901eae2b82e7828bc6354e6b7b8dca754af544bb79852b24a1a60978cecbadc4718842818dda6586088b6df74ac5c4de648e92eb842
  79d229addf4669db38cfba1245f6221f9dfdd4cc2c92978ef92c17d6089e3855d0a682b387b7ef5713b12f6843f958ba83dbad9d22caf9769a57d632238efc46155d69
  8e6"}

Body Cookies (1) Headers (23) Test Results   200 OK · 4 ms · 1.55 KB · Save Respons

{} JSON ∨ ▷ Preview Visualize ∨

1 {
2   "data": {
3   "message": "Data berhasil di tampilkan",
4   "data": [
5   {
6   "idproduk": 2,
7   "nama_produk": "E-Book Bisnis",
8   "detail_produk": "Panduan bisnis sukses",
9   "harga_produk": "75000.00",
10   "keunggulan produk": "Materi eksklusif",

4. Array1 = [2, 5, 8, 9]

   Array2 = [1, 2, 3, 4, 5, 6, 7]

   Buatlah script dengan menggunakan array diatas agar supaya di ubah menjadi

   Array3 = [1, 3, 4, 6, 7]

5. Buatlah script untuk menyelesaikan contoh soal berikut

String Random : PT.AbadI*perKASa@BeRsAmA-DIGItAL#SolUTiONs

Menjadi : PT. Abadi Perkasa Bersama Digital Solutions

6. Buat konversi nominal angka menjadi bilangan

Nominal : Rp.10.113.199,50

Terbilang : "Sepuluh Juta Seratus Tiga Belas Ribu Seratus Sembilan Puluh Sembilan Koma Lima Puluh Rupiah"

B. FRONT END

1. Buat CRUD dari soal nomor 1 dan 2 menggunakan Angular 12 secara visual

C. MOBILE DEV

1. Buat CRUD dari soal nomor 1 dan 2 menggunakan Java Kotlin Secara Visual (merupakan nilai tambah jika menggunakan salah satu antara state management BLOC, Provider, GetX atau RXDart)