**Versi Dokumen**

| **Versi** | **Tanggal** | **Penulis** | **Pemeriksa** | **Deskripsi Perubahan** |
| --- | --- | --- | --- | --- |
| 1.0 | 7 Juni 2022 | Yuli  Andi Sugito  Ruslan Efendi Nasution |  | Initial Version |

**DAFTAR ISI**

[APLIKASI 3](#_Toc105499891)

[GLOBAL 3](#_Toc105499892)

[IDE 3](#_Toc105499893)

[Berlaku umum 3](#_Toc105499894)

[Function 3](#_Toc105499895)

[File name 3](#_Toc105499896)

[Loging 4](#_Toc105499897)

[Login Aplikasi 4](#_Toc105499898)

[Enkripsi 4](#_Toc105499899)

[Send Email 4](#_Toc105499900)

[File Upload 5](#_Toc105499901)

[PHP 5](#_Toc105499902)

[GLOBAL 5](#_Toc105499903)

[Upload 5](#_Toc105499904)

[Installer PHP dan Apache 6](#_Toc105499905)

[Node js 6](#_Toc105499906)

[Template API 6](#_Toc105499907)

[Struktur Template 6](#_Toc105499908)

[Struktur Framework Service API 10](#_Toc105499909)

[DATABASE 11](#_Toc105499910)

[ORACLE 11](#_Toc105499911)

[SQL SERVER 13](#_Toc105499912)

[NAMING CONVENTION 14](#_Toc105499913)

# APLIKASI

## Global

### IDE

* Visual Studio Code
* Notepad ++
* Dreamweaver

### Berlaku umum

|  |  |  |
| --- | --- | --- |
| **Tipe** | **Naming** | **Keterangan** |
| String | StrVariable (StrNama) |  |
| Array | ArrVariable |  |
| Number | NumVariable |  |
| Boolean | BolVariable |  |
| Integer | IntVariable |  |

|  |  |  |
| --- | --- | --- |
| **Tipe** | **Naming** | **Keterangan** |
| String | ConsStrVariable |  |
| Array | ConsArrVariable |  |
| Number | ConsNumVariable |  |
| Boolean | ConsBolVariable |  |
| Integer | ConsIntVariable |  |

### Function

Nama Function Camel Case dan mencerminkan proses nya

- Function harus memiliki 1 atau lebih input parameter

- Function harus memiliki return value

Contoh Function :

*ProsesInsertData(ArrData, StrNoPermohonan){*

*Query = "Insert Into Data "*

*Return Query;*

*}*

### File name

* Penamaan nama file menggunakan Bahasa Indonesia dan menjelaskan isi dari file, CamelCase untuk nama File

Contoh :  Report Absensi – ReportAbsensi.php

### Loging

* Logging ditulis dalam file
* Penamaan file log sesuai dengan tanggal penulisan log [Log\_YYYYMMDD]
* Ukuran file log dibatasi perfile  10MB
* Jika ukuran file melebihi batas makan akan dibuatkan file log baru dengan akhiran sekuensial number

Contoh

* Log\_20220410\_01.txt
* Log\_20220410\_02.txt

Next :

* Format Log
* Issue /ping dan /metrics yang akan terus masuk log sedangkan ini tidak harus di masukkan log
* Masa retensi karena pengaruh ke loging () \* cek aturan di prodsup ?
* Masa retensi usahakan bisa parameter

### Login Aplikasi

* Untuk pengambangan aplikasi internal baru diharuskan melakukan autentikasi  menggunakan API LDAP
* Pengembangan aplikasi yang tidak menggunakan API LDAP, maka untuk informasi password harus di simpan dalam bentuk enkripsi

### Enkripsi

* Password  yang disimpan pada database Oracle di encypt menggunakan fucntion DB F\_SC\_ENCRYPT dan  F\_SC\_DECRYPT

* Enkripsi pada aplikasi menggunakan algoritma AES 256 dengan tambahan CRYPTO\_KEY dan CRYPTO\_IV

### Send Email

* Pengiriman Email untuk setiap aplikasi diarahkan menggunakan API atau Database ADMIN\_MESSAGE

### Send SMS

* Pengiriman SMS untuk setiap aplikasi diarahkan menggunakan API atau Database ADMIN\_MESSAGE

### File Upload

File yang diupload ke dalam server harus dipisahkan folder upload dari folder aplikasi

Akses Open dan Download harus menggunakan perantara script aplikasi (controller)

Harus di standarisasi kapan upload ke S3, Lokal Folder Server dan BLOB DB

* S3 peruntukan nya untuk sharing file antara aplikasi contoh BFLASH dan BLS

Contoh :

HTDOCS

* BLS
* Upload

View\_file.php

* UPLOAD\_BLS

## PHP

### Global

* Php minimum untuk new aplikasi 7.4 apahce 2.4
* Standarisasi penggunaan framework (laravel)
* Pembatasan penggunaan global variable
* Login Aplikasi menggunakan API LDAP
* Session timeout ( dafault  24 menit dan di setup pada php.ini)
* Max execution time php (default 60 second dan di setup pada php.ini), untuk kondisi tertentu misalkan upload dapat merubah max execution time dengan cara :

ini\_set('max\_execution\_time', '300'); //300 seconds = 5 minutes

ini\_set('max\_execution\_time', '0'); // for infinite time of execution

* Logging untuk aplikasi (waktu simpan, ratation file, size)
* Melakukan encypt untuk setting username dan pass Database, Akses API dan lain lain
* Terdapat autentication page sesuai dengan role dan menu step
* Penyeragaman Datagtid dan paging

### Upload

Proses upload harus validasi mime type, filesize, path lokasi file

if(!in\_array(mime\_content\_type($fileUpload), array(‘application/pdf’,’image/jpg’,’image/png’)))

{

echo  "Tipe file tidak diijinkan.";

}

List Mime Type :

[Common MIME types - HTTP | MDN (mozilla.org)](https://developer.mozilla.org/en-US/docs/Web/HTTP/Basics_of_HTTP/MIME_types/Common_types)

$file\_path = '/user01/work';

if (!file\_exists($file\_path)) {

    mkdir($file\_path, 0777, true);

}

### Installer PHP dan Apache

Menggunakan installer php dan apache yang standart

[https://bhinnekalife.sharepoint.com/:f:/s/ITSD-CDC/EvLFz2j7BtdLjaJUQtVY6ukBrfi\_a0ShM2\_VhVsr31hWkQ?e=tutkmB](https://bhinnekalife.sharepoint.com/:f:/s/ITSD-CDC/EvLFz2j7BtdLjaJUQtVY6ukBrfi_a0ShM2_VhVsr31hWkQ?e=tutkmB )

## **Node js**

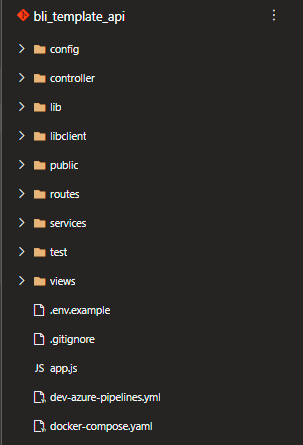
### Template API

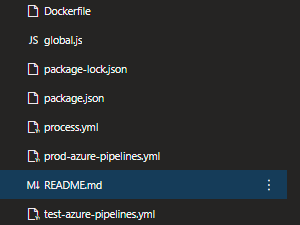
Template API dapat didownload dari link berikut :

<https://dev.azure.com/bli-org01/SOLDER/_git/bli_template_api>

### Struktur Template

Link Template : **https://dev.azure.com/bli-org01/SOLDER/\_git/bli\_template\_api**





| **No** | **Nama File / Folder** | **Keterangan** |
| --- | --- | --- |
| 1 | Config | * Folder untuk menyimpan konfigurasi yang dibutuhkan oleh service   Ex : Konfigurasi Aplikasi, Konfigurasi Database   * Penulisan nama file menggunakan camel case * Setiap filename mewakili 1 konfigurasi |
| 2 | Controller | * Folder untuk menyimpan logic controller endpoint api * Penulisan nama file menggunakan camel case * Setiap filename mewakili 1 fungsi logic controller   Contoh :  > SubmitDataController ( Untuk melakukan submit / insert data )  > EditDataController ( Untuk melakukan update data )  > DeleteDataController ( Untuk melakukan delete data ) |
| 3 | Lib | * Folder untuk menyimpan fungsi / konfigurasi yang dibutuhkan oleh Logic Controller * Penulisan nama file menggunakan camel case * **Config.js,**sebagai file konfigurasi untuk melakukan enkripsi value login BLAT * **CryptoBlat.js,**sebagai file yang berisi fungsi untuk melakukan enkripsi login BLAT * **JwtAuth.js,**sebagai file yang berisi fungsi middleware yang mencatat log ketika mengtrigger ke url endpoint api lainnya * **Logger.js**, sebagai file yang berisi fungsi untuk melakukan konfigurasi log dan mencatat log service aplikasi. Pada proses yang membutuhkan pencatatan log cukup memanggil fungsi yang sudah disiapkan   **Info(message)  -->**fungsi yang dipanggil sebagai log info  **Error(message) -->**fungsi yang dipanggil sebagai log error   * **Response.js,** sebagai file yang berisi fungsi untuk memberikan response api |
| 4 | Libclient | * Folder untuk menyimpan installer oracle client jika ada kebutuhan untuk mengakses database oracle * Installer ini dapat dihapus jika selama pengembangan tidak mengakses ke database oracle * Jika ada kebutuhan untuk mengakses database lainnya perlu disesuaikan kembali sesuai kebutuhan |
| 5 | Public | * Folder untuk menyimpan file css untuk kebutuhan response dalam html code * Folder ini belum digunakan untuk service yang dikembangkan saat ini |
| 6 | Routes | * Folder untuk menyimpan routing endpoint terhadap api yang akan digunakan. * Lakukan perubahan pada file Router.js untuk menambahkan logic controller yang baru |
| 7 | Services | * Folder untuk menyimpan service untuk menjalankan api pun database * Tidak perlu dilakukan perubahan jika tidak diperlukan |
| 8 | Tests | * Folder untuk menyimplan fungsi unit test jika ada * Tidak wajib ditambahkan jika tidak digunakan |
| 9 | Views | * Folder untuk meyimpan respose error * File ini belum digunakan untuk service yang pernah dikembangkan saat ini |
| 10 | .env.example | * File yang dibutuhkan untuk melakukan deployment pada local komputer sebagai template contoh terhadap env yang digunakan pada aplikasi * Pada local computer cukup ubah file .env.example menjadi .env |
| 11 | .gitignore | * File yang dibutuhkan untuk ignore file – file ataupun folder yang tidak dibutuhkan saat source code naik   Ex : folder logs |
| 12 | app.js | * File utama yang digunakan untuk menjalankan startup service |
| 13 | dev-azure-pipeline.yml | * File yang dibutuhkan untuk CI CD pada envi development. Yang perlu disesuaikan adalah   1. Trigger Branch        * 1. Service Connection Container Registry |
| 14 | docker-compose.yml | * FIle template contoh yang dibutuhkan untuk menjalankan service melalui docker-compose * Jika tidak menjalankan service melalui docker-compose maka tidak perlu dilakukan perubahan |
| 15 | Dockerfile | * File yang dibutuhkan saat proses CI DevOps Azure untuk build code sebagai docker |
| 16 | global.js | * File yang digunakan untuk menyimpan parameter global yang digunakan oleh aplikasi |
| 17 | package-lock.json | * File yang berisi daftar library beserta versi yang digunakan ketika melakukan development. * File ini tidak wajib diupload |
| 18 | package.json | * File yang berisi daftar library beserta versi yang digunakan ketika melakukan deployment |
| 19 | process.yml | * File yang berisi variable service ketika menjalankan docker melalui pm2 * Lakukan update untuk menyesuaikan nama service |
| 20 | prod-azure-pipeline.yml | * File yang dibutuhkan untuk CI CD pada envi production. Yang perlu disesuaikan adalah   1. Trigger Branch        * 1. Service Connection Container Registry |
| 21 | README.md | * File yang dapat diubah untuk memberikan informasi detail terhadap project, dapat berisi juga history development, sumber data jika dibutuhkan |
| 22 | test-azure-pipeline.yml | * File yang dibutuhkan untuk CI CD pada envi development. Yang perlu disesuaikan adalah   1. Trigger Branch        * 1. Service Connection Container Registry |

### Struktur Framework Service API

* Language Node Js dengan menggunakan library framework express
* Setiap service yang dikembangkan harus dapat menyimpan file log untuk kebutuhan analisa jika terjadi issue.
* Build dan deploy dalam bentuk docker
* Struktur MPC ( pemisahan logic controller, fungsi/Proses ataupun model database )
* Mengimplementasikan Unit Test pada pengembangan source ( nice to have )
* Parameter yang bersifat rahasia harus di enkripsi
* Menjalankan review code melalui sonarqube / sonarcloud.
* Mengimplementasikan Swagger ( nice to have )
* Pattern Routing API
  + /api/v1/[Aplikasi]/[EndPoint]

[Aplikasi] --> BLS / BAI / BDS / BSTAR / BLAT

[EndPoint] --> endpoint yang menggambarkan fungsi yang di panggil

* \*\*Setiap Fungsi pake logging ex: try catch\*\*
* Security API

# DATABASE

## ORACLE

### Global

* Bahasa Indonesia
* NAMA Snake case dipisahkan oleh under score (\_)
* Huruf besar semua
  + Contoh: TELEPON\_KLIEN
* Boleh menggunakan singkatan
* Boleh menggunakan istilah standard
* Harus menyertakan nama SCHEMA pada nama object (DDL)
* Harus ada initial / alias table jika menggunakan > 1 table  pada Query

Contoh Select \* from Klien **A**

* Insert into dan select statement harus menyebutkan semua kolom yang digunakan (tidak boleh menggunakan select \*)

### Constraint

Format nama

* PRIMERY KEY = [NAMA\_TABEL]\_PK
  + Contoh: APLIKASI\_PK

* FOREING KEY = [NAMA\_TABEL\_UTAMA]\_[NAMA\_TABEL REFERENCE]\_FK
  + Contoh: APLIKASI\_PRODUK\_FK
  + Nama tabel utama dan tabel reference bisa di singkat
    - Contoh (1):
      * Tabel utama: APLIKASI\_KUESIONER\_MEDIS
      * Tabel reference: KUESIONER\_MEDIS

APLKUEMED\_KUEMED\_FK

* + - Contoh (2):
      * Tabel utama: AKSEPTASI
      * Tabel reference: APLIKASI

AKS\_APL\_FK

### Funtion

* Format nama [NAMA\_SCHEMA].F\_[NAMA\_FUNCTION]
* Object di dalam nya (table, function, view) tidak menggunakan NAMA\_SCHEMA jika dalam satu schema dengan function tersebut.
* Jika berbeda schema dengan schema function,wajib menggunakan NAMA\_SCHEMA
* Setiap set variable dari select wajib di bungkus dengan BEGIN EXECPTION END
* EXCEPTION

### Index

* Format nama  [NAMA\_SCHEMA].IDX[XX][NAMA\_TABLE]
* XX  number incremental
* NAMA\_TABLE  nama table
  + Contoh: IDX01\_APLIKASI
* Harus menyertakan nama SCHEMA pada table yang di index
  + Contoh: CREATE INDEX ADMIN\_BPLINK.IDX01APLIKASI ON **ADMIN\_BPLINK**.APLIKASI

### Job

* Format nama [NAMA\_SCHEMA].JOB\_[NAMA\_PROCEDURE]
* JOB memanggil 1 Procedure
* Informasi INTERVAL JOB

|  |  |
| --- | --- |
| **INTERVAL JOB** | **Keterangan** |
| SYSDATE+120/1440 | dijalankan setiap 2 jam |
| SYSDATE+360/1440 | dijalankan setiap 6 jam |
| SYSDATE+60/1440 | dijalankan setiap jam |
| SYSDATE+1800/86400 | dijalankan setiap setengah jam |
| TRUNC(SYSDATE+1)+6/24 | dijalankan setiap jam 6 pagi 1 kali sehari |
| TRUNC(SYSDATE+1)+1/24 | dijalankan setiap jam 1 pagi 1 kali sehari |
| TRUNC(LAST\_DAY(SYSDATE)) + 1 | Awal bulan di bulan berikutnya akan jalan |

### Procedure

* Format nama [NAMA\_SCHEMA].P\_[NAMA\_PROSES]
* Object di dalam nya (table, function, view) tidak menggunakan NAMA\_SCHEMA jika dalam satu schema dengan tersebut.
* Jika berbeda schema dengan schema procedure,wajib menggunakan NAMA\_SCHEMA
* Declare variable hanya yang digunakan dalam Procedure
* Setiap set variable dari select wajib di bungkus dengan BEGIN EXCEPTION END
* Wajib EXCEPTION
* Untuk SP yang di Panggil JOB harus masuk ke table log (JOB\_LOGS):

- TGL\_PROSES = SYSDATE

- TGL\_FINISH =SYSDATE

|  |  |  |  |
| --- | --- | --- | --- |
| ID | NAMA\_PROSES | ST\_PROSES | KETERANGAN\_PROSES |
| Diisi oleh sequence JOB\_LOG\_SEQ | [NAMA  PROCEDURE/JOB ] | 1= Proses sedang berlangsung  2= Proses selesai dan sukses  3= Proses error | 1= Proses mulai  2= Proses selesai  3= SUBSTR('Proses Gagal :' || '[' || SQLCODE || ']' || SQLERRM, 1, 4000) |

### Sequence

Format nama [NAMA\_SCHEMA].SEQ\_[NAMA\_TABEL]

### Table

* Format nama [NAMA\_SCHEMA].[NAMA\_TABEL]
* Harus memiliki PK
* Untuk table transaksi harus memiliki kolom uniq ID yang diisi oleh sequence dan trigger
* Pengelompokan table
* Seluruh table transaksi memiliki kolom diletakkan di depan setelah unik id:

|  |  |
| --- | --- |
| **Nama Kolom** | **Tipe Data** |
| USER\_RECORD | VARCHAR2(20 BYTE) |
| TGL\_RECORD | DATE |
| USER\_UPDATE | VARCHAR2(20 BYTE) |
| TGL\_UPDATE | DATE |

### Trigger

* Format nama TRG\_[NAMA\_PROSES]\_[NAMA\_TABLE]

|  |  |  |
| --- | --- | --- |
| **Nama** | **Keterangan** | **Contoh** |
| BINS | Before insert | TRG\_BINS\_APLIKASI |
| BUPD | Before update | TRG\_BUPD\_APLIKASI |
| BDEL | Before delete | TRG\_BDEL\_APLIKASI |
| AINS | After insert | TRG\_AINS\_APLIKASI |
| AUPD | After update | TRG\_AUPD\_APLIKASI |
| ADEL | After delete | TRG\_ADEL\_APLIKASI |

* Menyertakan SCHEMA pada table
* Setiap set variable dari select wajib di bungkus dengan BEGIN EXECPTION END;
* EXCEPTION

### View

* Format nama VW\_[NAMA\_TABLE/NAMA PROSES]
* Define field yang akan di tampilkan
* Tidak boleh select \* untuk sub query dan patikan select untuk data yang akan ditampilkan
* Harus menggunakan alias tabel
* Tidak menggunakan nama SCHEMA pada query didalam view

### Variable

* Format nama variable:

|  |  |  |  |
| --- | --- | --- | --- |
| **Tipe** | **Tipe data** | **Format** | **Contoh** |
| Variabel inputan | Varchar/string/char/text | S\_[NAMA VARIABLE] | S\_USER\_RECORD |
| Variabel inputan | Integer/number/decimal | I\_[NAMA VARIABLE] | I\_CEK\_LIBUR |
| Variabel inputan | Date | D\_[NAMA VARIABLE] | D\_CEK\_LIBUR |
| Variabel lokal | Varchar/string/char/text | LS\_[NAMA VARIABLE] | LS\_USER\_RECORD |
| Variabel lokal | Integer/number/decimal | LI\_[NAMA VARIABLE] | LI\_CEK\_LIBUR |
| Variabel inputan | Date | LD\_[NAMA VARIABLE] | LD\_CEK\_LIBUR |

## SQL SERVER

### Global

* Bahasa Indonesia
* NAMA Camel case, huruf  pertama menggunakan huruf besar
* Huruf besar semua
  + Contoh: TeleponKlien
* Boleh menggunakan singkatan
* Boleh menggunakan istilah standard
* Harus menyertakan nama SCHEMA pada nama object (DDL)
* Harus ada initial table jika menggunakan > 1 table
* Insert into dan select statement harus menyebutkan semua kolom yang digunakan (tidak boleh menggunakan select \*)

| No | Kategori/ Mengandung Kata | Nama | | | Contoh |
| --- | --- | --- | --- | --- | --- |
| Depan | Tengah | Belakang |
| 1 | Transaksi | Trx |  |  | TrxPolis |
| 2 | Master | Ms |  |  | MsKlien |
| 3 | Reporting | Lap | Jenis report | Nama laporan | LapKlaimMeninggal  LapCollactionNewBusinsess  LapCollactionRenewalJatuhTempo |
| 4 | Parameter | Param |  |  | ParamRate |
| 5 | Konfigurasi |  |  | Config | ETLConfig  ReportConfig  TabelConfig  PackageConfig |
| 6 | Log |  |  | Log | ETLLog |

### Constraint

Format nama

* PRIMERY KEY = [NamaTabel]PK
  + Contoh: AplikasiPK

* FOREING KEY = [NamaTabelUtama][NamaTabelReferenceFK
  + Contoh: AplikasiProdukFK
  + Boleh di singkat

### Funtion

1. Menggunakan Standarisasi Camel Case

* [*String*](https://id.wikipedia.org/wiki/String)*diawali dengan huruf besar.*
* *String* tidak memiliki spasi di antaranya.
* *String* memiliki paling sedikit dua huruf kapital yang dipisahkan oleh paling sedikit sebuah huruf kecil

1. Nama diawali Fn[Nama Function]

Contoh :

* FnCekHariLibur

1. Object di dalam nya (table, function, view, triger) tidak menggunakan NamaShcema jika dalam satu schema dengan tersebut.
2. Jika berbeda schema dengan schema procedure,wajib menggunakan NamaShcema

### Index

* Format nama  [NamaSchema].IDX[XX][NamaTabel]
* XX  number incremental
* NamaTabel  nama table
  + Contoh: IDX01Aplikasi
* Harus menyertakan nama SCHEMA pada table yang di index
  + Contoh: CREATE INDEX ADMIN\_BPLINK.IDX01Aplikasi  ON **ADMIN\_BPLINK**.APLIKASI

### Job

1. Menggunakan Standarisasi Camel Case

* [*String*](https://id.wikipedia.org/wiki/String)*diawali dengan huruf besar.*
* *String* tidak memiliki spasi di antaranya.
* *String* memiliki paling sedikit dua huruf kapital yang dipisahkan oleh paling sedikit sebuah huruf kecil

1. Nama diawali Job[NamaProses]

Contoh :

* JobHariLibur

### Procedure

1. Menggunakan Standarisasi Camel Case

* [*String*](https://id.wikipedia.org/wiki/String)*diawali dengan huruf besar.*
* *String* tidak memiliki spasi di antaranya.
* *String* memiliki paling sedikit dua huruf kapital yang dipisahkan oleh paling sedikit sebuah huruf kecil

1. Format nama [NamaSchema].SP[NamaProses]

Contoh :

* SPHariLibur

1. Object di dalam nya (table, function, view, triger) tidak menggunakan NamaShcema jika dalam satu schema dengan tersebut.
2. Jika berbeda schema dengan schema procedure,wajib menggunakan NamaShcema
3. Declare variable hanya yang digunakan dalam Procedure
4. **Setiap set variable dari select wajib di bungkus dengan BEGIN EXECPTION END**
5. **Wajib EXCEPTION**
6. Harus masuk ke table log (JobLogs):

- TglProses = SYSDATE

- TglFinish =SYSDATE

|  |  |  |  |
| --- | --- | --- | --- |
| Id | NamaProses | STProses | KeteranganProses |
| incremental | [NAMA  PROCEDURE/JOB ] | 1= Proses sedang berlangsung  2= Proses selesai dan sukses  3= Proses error | 1= Proses mulai  2= Proses selesai  3= SUBSTR('Proses Gagal :' || '[' || SQLCODE || ']' || SQLERRM, 1, 4000) |

### Table

1. Menggunakan Standarisasi Camel Case

* [*String*](https://id.wikipedia.org/wiki/String)*diawali dengan huruf besar.*
* *String* tidak memiliki spasi di antaranya.
* *String* memiliki paling sedikit dua huruf kapital yang dipisahkan oleh paling sedikit sebuah huruf kecil

Contoh :

* AlamatLengkap
* JenisKelamin
* StatusPerkawinan

1. Untuk table **master data** gunakan prefix **Rf**diawal pembentukan nama table

Contoh :

* RfStatusPerkawinan

1. Untuk table **transaksi**gunakan prefix **Tr**diawal pembentukan nama table

Contoh :

* TrRiwayatPolis

1. Untuk table **transaksi** wajib memiliki Primary Key ( BigDecimal )

### Trigger

1. Menggunakan Standarisasi Camel Case

* [*String*](https://id.wikipedia.org/wiki/String)*diawali dengan huruf besar.*
* *String* tidak memiliki spasi di antaranya.
* *String* memiliki paling sedikit dua huruf kapital yang dipisahkan oleh paling sedikit sebuah huruf kecil

1. Nama diawali Trg[NamaTriger]

Contoh :

* TrgTrxPolisAmtPremi

1. Object di dalam nya (table, function, view, triger) tidak menggunakan NamaShcema jika dalam satu schema dengan tersebut.
2. Jika berbeda schema dengan schema procedure,wajib menggunakan NamaShcema

### View

1. Menggunakan Standarisasi Camel Case

* [*String*](https://id.wikipedia.org/wiki/String)*diawali dengan huruf besar.*
* *String* tidak memiliki spasi di antaranya.
* *String* memiliki paling sedikit dua huruf kapital yang dipisahkan oleh paling sedikit sebuah huruf kecil

1. Nama diawali Vw[NamaView]

Contoh :

* VwHariLibur

1. Object di dalam nya (table, function, view, triger) tidak menggunakan NamaShcema jika dalam satu schema dengan tersebut.
2. Jika berbeda schema dengan schema procedure, wajib menggunakan NamaShcema