**Kodok**

**Dokumentasi v 1.0**

***Engineer***



**Oktober 2022**

# Daftar Isi

[Daftar Isi 2](#_Toc117070830)

[Words From Previous Developers To Our Successor 4](#_Toc117070831)

[1. Database 5](#_Toc117070832)

[a. Users 5](#_Toc117070833)

[b. Users\_Levels 5](#_Toc117070834)

[c. Projects 6](#_Toc117070835)

[d. Projects\_Types 6](#_Toc117070836)

[e. Projects\_Stats 6](#_Toc117070837)

[f. Projects\_Keterangans 6](#_Toc117070838)

[g. Mitras 6](#_Toc117070839)

[h. Products 6](#_Toc117070840)

[i. Projects\_Handovers 6](#_Toc117070841)

[j. Documents 7](#_Toc117070842)

[k. Documents\_Type 7](#_Toc117070843)

[l. Migrations 7](#_Toc117070844)

[2. MVC (Model, View, Controller) 7](#_Toc117070845)

[a. Model 7](#_Toc117070846)

[b. View 8](#_Toc117070847)

[c. Controller 8](#_Toc117070848)

[I. Login 9](#_Toc117070849)

[II. Admin 10](#_Toc117070850)

[ Controller\_AdminApprovalDocument 10](#_Toc117070851)

[ Controller\_AdminExportData 10](#_Toc117070852)

[ Controller\_ListProject 10](#_Toc117070853)

[ Controller\_AdminMitra 10](#_Toc117070854)

[ Controller\_AdminProducts 10](#_Toc117070855)

[ Controller\_AdminUsers 11](#_Toc117070856)

[III. Engineer 11](#_Toc117070857)

[ Controller\_EngineerDocumentsTools 11](#_Toc117070858)

[ Controller\_EngineerHandoverProjects 11](#_Toc117070859)

[ Controller\_EngineerHistoryHandover 11](#_Toc117070860)

[ Controller\_EngineerHistoryProjects 11](#_Toc117070861)

[ Controller\_EngineerListProjects dan Controller\_EngineerUploadDocument 11](#_Toc117070862)

[ Controller\_EngineerYourProjects 12](#_Toc117070863)

[IV. Guest 12](#_Toc117070864)

[V. Manager 12](#_Toc117070865)

[ Controller\_ManagerApprovalProjects 12](#_Toc117070866)

[ Controller\_ManagerAssignProjects 12](#_Toc117070867)

[ Controller\_ManagerHome 13](#_Toc117070868)

[ Controller\_ManagerListProjects 13](#_Toc117070869)

[VI. User Selain Admin, Manager, Engineer, dan Guest 13](#_Toc117070870)

[3. File Pendukung Lainnya 14](#_Toc117070871)

[a. script.js 14](#_Toc117070872)

[b. .env 14](#_Toc117070873)

[c. web.php 14](#_Toc117070874)

[d. Export 14](#_Toc117070875)

[e. Import 15](#_Toc117070876)

[4. Interaksi Antar File 15](#_Toc117070877)

[5. Struktur Organisasi 15](#_Toc117070878)

[6. Life Cycle 16](#_Toc117070879)

# Words From Previous Developers To Our Successor

Halo developer Kodok, selamat datang di dokumentasi aplikasi Kodok. Semoga hari kalian indah dan selalu diberi keberkahan oleh yang Maha Kuasa. Amin. Sebelumnya kami ucapkan selamat dan sukses dalam mengembangkan aplikasi Kodok agar lebih baik kedepannya. Semoga aplikasi ini dapat digunakan oleh seluruh staff SDTL secara rutin hingga akhir hayat Artajasa dan bisa bermanfaat untuk memanage projek – projek yang ada di SDTL sebagaimana rencana awalnya aplikasi ini dibuat.

Di dalam dokumentasi ini, akan dijelaskan mengenai struktur database Kodok dan source code yang membuat aplikasi ini bisa berjalan. Mulai dari kegunaan masing – masing bagian dan flow dari masing – masing fitur. Ingat, yang ada di dalam dokumentasi ini hanyalah yang berkaitan dengan Kodok, bukan Laravel itu sendiri. Jadi untuk hal - hal menyangkut cara kerja Laravel, bisa dilihat langsung di dokumentasi Laravel di website resminya.

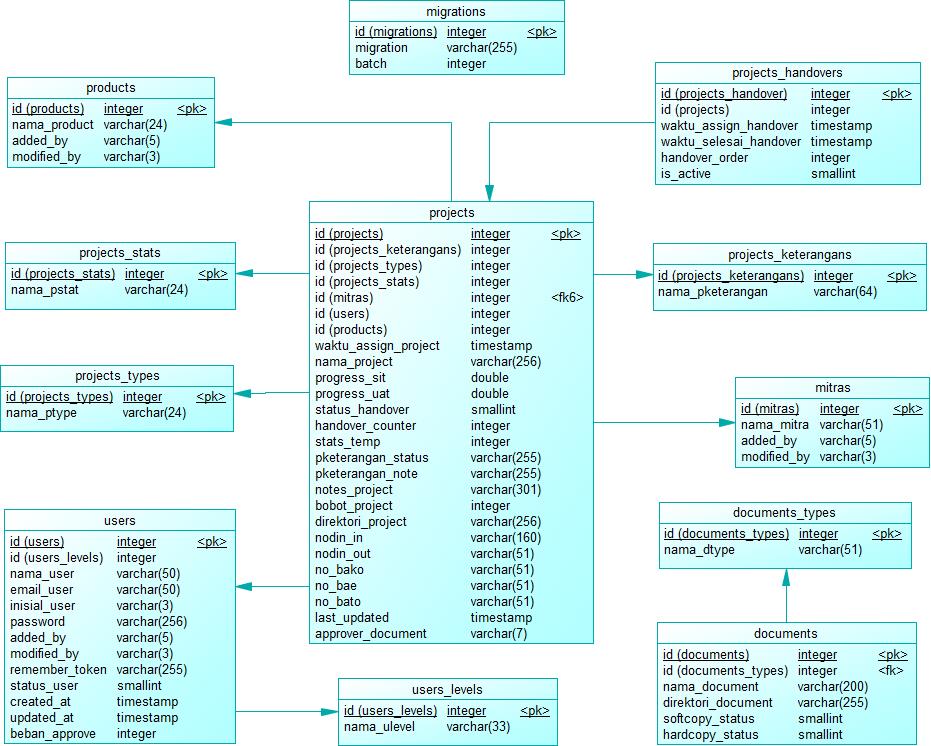
Semoga dokumentasi ini bisa menjadi petunjuk yang cukup jelas untuk siapa pun yang akan melanjutkan development aplikasi Kodok. Dan kami ucapkan juga selamat melanjutkan perjuangan kami para pendahulu kalian yang sudah mengerahkan segenap usaha untuk aplikasi ini. Jika mungkin ada kesalahan dan sesuatu yang kurang dari dokumentasi ini atau kodingannya itu sendiri, kami ucapkan maaf yang sebesar – besarnya. Dan bagi siapa pun yang ingin mengupdate bagian ini, bisa diubah sesuka hati kalian. Mohon dokumentasi ini bisa terus diupdate tiap kali ada update pada aplikasi Kodok.

Sekian, Terima kasih. Salam hangat dari kami yang pernah singgah



# Database

Struktur database Kodok seperti gambar di bawah:



Gambar 1: Struktur database website Kodok

## Users

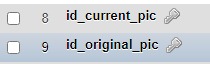
Tabel Users megandung data user berupa nama, inisial, dan email beserta rolenya yang direference dari tabel Users\_Levels menggunakan idnya. Kolom password menyimpan password dalam bentuk hash. Remember\_token merupakan kolom default dari Laravel. Added\_by dan modified\_by berisi inisial user Admin yang menambah / mengubah data user, sedangkan created\_at dan updated\_at berisi waktu data user ditambah / diubah. Dan beban\_approve digunakan sebagai flag pada sistem pembagian approval dokumen untuk user Admin.

## Users\_Levels

Tabel Users\_Levels digunakan untuk menyimpan role user. Users\_Levels sendiri akan menentukan hak akses dari masing – masing user. Tabel ini hanya berisikan id dan nama dari level usernya.

## Projects

Tabel Projects berisikan seluruh data sebuah projek, mulai dari nama projek, waktu assign, progress pengujian (SIT dan UAT), nomor dokumen (BAKO, BATO, BAE, Nodin In dan Nodin Out), mitra, produk, keterangan projek, status projek, tipe projek, pic utama maupun pengganti. Selain itu, user bisa menambahkan catatan, Kolom stats\_temp digunakan dalam sistem perubahan status. Status\_handover dan handover\_counter digunakan pada sistem handover project. Kolom last\_updated digunakan untuk menyimpan waktu terakhir user melakukan perubahan pada data di tabel projects. Dan approver\_document berisikan inisial user admin yang melakukan approval dokumen.

Pada implementasinya, id user di dalam satu data Projects direference di dua kolom berbeda, yaitu pada kolom id\_current\_pic dan id\_original\_pic. Id\_current\_pic menyimpan id user yang memegang projek tersebut saat ini, sedangkan id\_original\_pic menyimpan id user yang asli.

Gambar 2: PIC pada tabel Projects

## Projects\_Types

Tabel ini menyimpan jenis projek seperti Internal Test, Sertifikasi, dll.

## Projects\_Stats

Tabel ini menyimpan status sebuah projek seperti On Progress, Pengujian Done, Projek Done, dll.

## Projects\_Keterangans

Tabel ini menyimpan keterangan approval projek, seperti Approved by Manager, Declined by Manager, dll.

## Mitras

Tabel Mitras menyimpan data mitra berupa id dan nama mitra. Selain itu tabel ini juga menyimpan user yang menambahkan ke database dan yang mengubah datanya.

## Products

Tabel Products menyimpan data produk yang ada di Artajasa seperti ATM Bersama, Payment, ATM Bersama QR, dll.

## Projects\_Handovers

Tabel Projects\_Handovers menyimpan data ketika sebuah projek dihandover ke orang lain. Tabel ini menyimpan id projek yg di handover, beserta waktu handover mulai dan selesai. Selain itu tabel ini juga menyimpan urutan handovernya, dan apakah handover ini msh berjalan atau tidak.

## Documents

Tabel ini menyimpan data dokumen pengujian seperti direktori tempat dokumen disimpan, nama dokumennya, jenis dokumen, dan status apakah dokumen tersebut sudah disubmit atau belum. Walaupun tabel ini sudah ada di database, namun secara fungsional tabel ini belum diimplementasikan ke dalam sistem Kodok.

## Documents\_Type

Tabel ini menyimpan tipe dokumen, seperti MOM, BATO, BAKO, Nodin, dll. Walaupun tabel ini sudah ada di database, namun secara fungsional tabel ini belum diimplementasikan ke dalam sistem Kodok.

## Migrations

Tabel ini merupakan tabel bawaan dari Laravel yang digunakan untuk mencatat history migration database. Karena pada projek ini kita menggunakan migration, maka tabel ini otomatis degenerate oleh Laravel.

# MVC (Model, View, Controller)

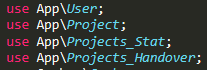
Secara struktur, Laravel merupakan framework yang mengimplementasikan MVC (Model, View, Controller). Secara umum, Model berfungsi untuk menghandle database, View menghandle tampilan / front end, dan Controller menghandle back end. Untuk format penulisan function pada website Kodok, kita menggunakan camel case dan untuk variabel menggunakan lowercase.

Pada bab ini akan menjelaskan secara umum kegunaan dari masing – masing file di website Kodok dan flow dari sebuah action. Untuk penjelasan lebih detail per line bisa dilihat di comment di dalam filenya masing - masing.

## Model

Model merupakan file yang menyambungkan projek Laravel dengan database dan bisa ditemukan di direktori Kodok/apps dengan format nama file [Nama Tabel Tanpa Akhiran “s”].php. Contohnya, tabel projects pada database akan dihandle oleh model Project.php. Sebenarnya penamaan nama model bisa dilakukan secara bebas, namun ada step tambahan yaitu menambahkan line “protected $table = ‘[Nama Tabel]’;”, maka dari itu untuk aplikasi Kodok kita menggunakan format penamaan yang standard.

Model sendiri dapat digunakan dengan mengimportnya terlebih dahulu di bagian atas source code seperti gambar dibawah.



Jika model sudah diimport seperti contoh pada Gambar 6, maka tabel users, projects, projects\_stats, dan projects\_handovers di database bisa digunakan untuk query builder pada controller tersebut. Kita bisa menambahkan model apa saja yang diperlukan pada controller sesuai kebutuhan halamannya.

Model sendiri berisi rules – rules untuk sebuah tabel. Function protected $attributes berfungsi untuk menentukan default value dari sebuah kolom. Lalu function protected $fillable berfungsi kolom mana saja yang dapat diubah dari aplikasi. Dan function public $timestamp = false digunakan untuk tabel – tabel yang memiliki data type timestamp supaya data baru bisa dimasukkan.

## View

View merupakan file yang menghandle tampilan / front end. View sendiri bisa ditemukan di direktori Kodok/resources/views. Di aplikasi Kodok ini, view kita bagi ke dalam beberapa folder yaitu Layouts, Mails, Pages, dan Templates.

## Controller

Controller merupakan file yang menghandle bagaimana sebuah halaman dalam web bekerja atau back end. Di dalam file ini berisi function – function yang membuat sebuah halaman bekerja sebagaimana mestinya.

Pada direktori controller, terdapat satu nama controller yang dimiliki oleh semua user, yaitu Controller\_[User]SearchDocuments. Controller ini dibuat untuk menghandle halaman yang bisa mencari dokumen – dokumen yang sudah diupload. Namun fitur ini belum diimplementasikan ke aplikasi Kodok sehingga isinya hanya ada function untuk membuka halaman dan mengenerate tabel. Selain controller tersebut, ada juga beberapa controller yang sudah ada namun belum diimplementasi di aplikasi Kodok dengan harapan kedepannya akan diselesaikan fitur tersebut.

File – file controller terletak di direktori Kodok/app/Http/Controllers. Satu Controller memegang satu halaman pada web Kodok. Untuk sistem penamaan controller, kita menggunakan format Controller\_[User][Nama Halaman].php, contoh: Controller\_EngineerListProjects.php.

Ada pula function yang berada di tiap controller seperti function untuk membuka halaman dan function dataTable yang berfungsi untuk mengenerate tabel di halaman tersebut. Untuk function dataTable hanya dimiliki oleh halaman – halaman yang memiliki tabel di dalamnya dan isinya berbeda tiap halaman tergantung data apa saja yang perlu ditampilkan di halaman tersebut.

Di tiap function yang bersifat action, yaitu function yang harus ditrigger oleh user agar function tersebut dijalankan, terdapat authorizatrion bawaan Laravel untuk membatasi action yang dapat dilakukan oleh user.

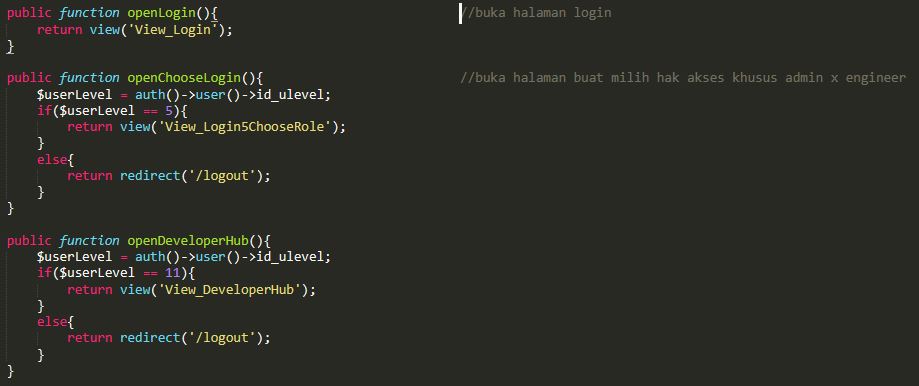


Seperti yang bisa dilihat pada Gambar 4, merupakan contoh authorization yang disediakan oleh Laravel. ‘isAdmin’ pada Gambar 4 menandakan bahwa hanya user Admin yang dapat mengakses function tersebut. Untuk menambah jenis hak akses, bisa dengan menambahkan jenis baru pada file AccessPolicy.php yang terletak di direktori Kodok/app/Policies.

Hal lain yang perlu diperhatikan juga yaitu setiap function berupa action yang mengubah data pada sebuah tabel, Kodok mengimplementasikan AJAX agar setiap kali sebuah data ditambah, diedit, atau dihapus maka tabel akan langsung direfresh dan perubahan data langsung ditampilkan. File javascript yang menghandle ini ada di direktori Kodok/public/assets/dist/js di file script.js. Selain itu, file script.js juga menghandle proses menampilkan modal seperti form atau detail data projek. Dan untuk semua button submit, script.js juga lah yang mengatur agar menampilkan Swal berupa notifikasi data berhasil / gagal diubah.

### Login

Controller yang menghandle proses login merupakan Controller\_Login. Controller ini menghandle semua kegiatan login pada website Kodok. Di dalam controller ini terdapat function openLogin, openChooseLogin, authenticate, dan logout.

Function openLogin() dan openChooseLogin() berfungsi untuk membuka halaman login dan halaman memilih role khusus untuk user dengan role Admin x Engineer.

Function authenticate berfungsi untuk mengautentikasi data yang dimasukkan user pada halaman. Secara flow, pertama data divalidasi. Jika domain email tidak sesuai atau salah satu box tidak diisi maka Kodok akan mengeluarkan warning. Jika data sudah sesuai formatnya, email dan password akan dilempar ke database CIS Artajasa untuk diverifikasi. Jika data yang dimasukkan salah maka Kodok akan meredirect ke halaman login, namun jika data sudah benar, maka Kodok akan mengenerate password dalam bentuk hash. Setelah itu, Kodok akan memeriksa email yang dimasukkan apakah sudah terdaftar di DB Kodok atau belum. Jika belum, maka user akan didaftarkan secara otomatis dengan role guest. Dan jika sudah, maka password user di DB akan diupdate. Setelah itu, Kodok akan menjalankan autentikasi data kembali ke DB Kodok, lalu berdasarkan rolenya maka user akan diredirect ke halamannya masing – masing.

Function logout digunakan untuk menjalankan fungsi logout dari website Kodok.

### Admin

#### Controller\_AdminApprovalDocument

Controller ini berfungsi untuk mengatur function yang ada di halaman Approval Document untuk user Admin. Di controller ini hanya ada satu function action yaitu function untuk approve kelengkapan dokumen. Jika dokumen sudah dianggap lengkap oleh admin, maka data projek tersebut akan dioper ke manager untuk diapprove oleh manager. Jika projek direject oleh user Admin, maka Admin diberikan kemampuan untuk menulis pesan singkat untuk user Engineer kenapa projek tersebut direject / tidak diapprove.

#### Controller\_AdminExportData

Controller ini berisikan function untuk export data per tabel yang sudah dipisahkan masing – masing functionnya, dan function “store” yang digunakan untuk mengimport data. Namun secara frontend, untuk import data didisable karena mengakibatkan “waktu\_assign\_project” pada tabel project menjadi waktu dan tanggal data diimport, bukan data aslinya.

#### Controller\_ListProject

Controller ini menghandle function yang ada pada halaman List Project pada user Admin. Di halaman ini terdapat function export yang sudah diset data apa saja yang akan ditampilkan pada excel sebagai ouputnya yang dapat dilihat di file AdminProjectExport.php pada direktori Kodok/app/Exports.

Lalu ada juga function “detail” yang akan meload detail dari sebuah projek. Function ini menjalankan query yang disimpan juga querynya pada file controller ini juga, lalu melempar data – data tersebut ke frontend untuk ditampilkan.

#### Controller\_AdminMitra

Controller ini menyimpan function untuk memanage data – data mitra mulai dari menambah, mengedit, dan menghapus.

Proses tambah data mitra dibagi menjadi dua function, yaitu “create” dan “store”. Function create berfungsi untuk menyiapkan model Mitra di form yang nantinya akan diisi oleh user dengan data mitra baru. Lalu function store yang akan menyimpan data yang sudah diisi ke database.

Proses edit data mitra juga dibagi menjadi dua function, yaitu function “edit” dan “update”. Function edit berfungsi untuk mengambil data mitra yang ingin diubah, lalu ditampilkan ke dalam form edit mitra. Sedangkan function update berfungsi untuk mengambil data yang sudah diedit lalu menyimpannya ke database.

#### Controller\_AdminProducts

Controller ini memiliki function – function yang mengatur halaman Product pada user Admin. Function yang ada pada controller ini sama dengan yang ada di Controller\_AdminMitra, namun data yang dihandle merupakan data product.

#### Controller\_AdminUsers

Controller ini mengatur function yang ada pada halaman Users pada user Admin. Secara function mirip dengan halaman Products dan Mitra, namun yang membedakan ada pada function “destroy”, function untuk menghapus data.

Ketika admin menghapus data user, yang terjadi adalah flag “status\_user” yang ada pada tabel Users diubah nilainya dari satu menjadi nol. Hal ini dilakukan karena id user menjadi foreign key pada tabel projects, yang menyebabkan data user tidak bisa dihapus begitu saja. Melainkan harus mengganti semua id yang masih menjadi foreign key di tabel projects, baru data user tersebut bisa dihapus. Selain itu hal ini juga dilakukan agar apabila suatu saat user tersebut kembali ke SDTL, maka flag “status\_user” bisa dikembalikan nilai menjadi satu.

### Engineer

#### Controller\_EngineerDocumentsTools

Controller ini merupakan controller yang menhandle halaman Documents Tools. Namun fitur ini belum diimplementasi di aplikasi Kodok.

#### Controller\_EngineerHandoverProjects

Controller ini mengatur halaman Project On Going, Handover Project pada user Engineer. Halaman ini memiliki function untuk menampilkan projek apa saja yang sedang dihandover ke user tersebut. Lalu action yang dapat dilakukan oleh engineer pada halaman ini ialah mengubah status handover menjadi done ketika projek yang dihandover sudah selesai masa handovernya dan ingin dikembalikan kepada pemilik aslinya.

#### Controller\_EngineerHistoryHandover

Controller ini berisikan function untuk halaman Project Done, Handover Project. Pada halaman ini hanya ada function untuk menampilkan data – data projek yang telah selesai dihandover ke user tersebut.

#### Controller\_EngineerHistoryProjects

Controller ini mirip seperti halaman Project Done, Handover Project, namun yang berbeda adalah halaman ini menampilkan data – data projek yang sudah berstatus Projek Done atau Drop. Halamanya ada di Project Done, Your Project.

#### Controller\_EngineerListProjects dan Controller\_EngineerUploadDocument

Controller\_EngineerListProjects menghandle function pada halaman List Project di user Engineer. Untuk halaman ini di user Engineer blm diimplementasikan, jadi baru ada controllernya tinggal dibuat viewnya saja jika jadi ingin diimplementasikan.

Sedangkan Controller\_EngineerUploadDocument berfungsi untuk menghandle halaman Upload Document. Namun fitur ini juga belum diimplementasikan di aplikasi Kodok.

#### Controller\_EngineerYourProjects

Controller ini mengatur backend pada halaman Project On Going, Your Project. Di halaman ini, user Engineer dapat mengubah status pekerjaan dan mengubah data – data tambahan seperti progress SIT UAT, PIC diluar SDTL yaitu AM, PM, dan Produk, dan menambahkan notes jika terdapat kendala pada pengujian supaya manager bisa mengetahui lebih detail bagaimana kondisi sebuah projek.

Untuk perubahan status ke “Pengujian Done” dan “Projek Done”, maka projek tersebut harus diapprove terlebih dahulu oleh user Admin dan Manager. Sebelum diapprove, maka statusnya akan tetap “On Progress”. Apabila user Admin atau Manager melakukan reject pada projek tersebut, maka akan ditulis juga alasan kenapa direject agar user bisa mengetahui apa Langkah yang harus dilakukan agar projeknya bisa diapprove.

### Guest

Sejauh ini, user Guest hanya memiliki satu controller yaitu Controller\_GuestSearchDocuments yang akan menghandle halaman Search Documents. User ini dibuat untuk semua user di Artajasa yang bukan dari SDTL dan bukan VP SDTL yang saat ini dipegang oleh VP ITO 1. Diawal, user ini dikhususkan untuk kegiatan audit, sehingga halaman yang dibuat untuk user ini adalah halaman untuk pencarian dokumen agar bisa dicek kelengkapan dokumen yang diupload untuk semua projek di SDTL.

### Manager

#### Controller\_ManagerApprovalProjects

Controller ini berfungsi untuk memanage halaman Approcal Projects untuk user Manager. Di halaman ini, user Manager bisa melakukan approval terhadap projek dengan status “Pengujian Done” dan “Projek Done”. Namun sebelum data projek masuk ke halaman ini, projek tersebut harus sudah diapprove oleh user Admin terkait kelengkapan dokumen projek. Jika user Manager melakukan reject terhadap sebuah projek, maka user Manager diberikan kemampuan untuk menulis pesan singkat untuk user Engineer kenapa projek tersebut direject / tidak diapprove.

#### Controller\_ManagerAssignProjects

Controller ini menghandle function untuk halaman Assign Project pada user Manager. Di halaman ini, manager bisa mengassign projek baru atau menghandover projek kepada user Engineer.

Pada saat mengassign projek baru, maka function yang digunakan ialah function “storeNew”. Seperti function penambahan data lainnya, function ini menerima input user lalu menambahkannya ke dalam tabel projects.

Lalu pada saat menghandover sebuah projek, ketika projek yang ingin dihandover ternyata masih dipegang oleh seorang user, maka user Manager berhak memindahkannya handover tersebut ke user baru dan status handover pada user handover yang lama menjadi otomatis done sehingga handovernya tidak dobel dipegang oleh 2 user.

#### Controller\_ManagerHome

Controller ini merupakan controller yang mengatur halaman Home pada user Manager. Controller ini hanya melakukan query terhadap data – data yang diperlukan untuk mengisi grafik lalu melempar data tersebut ke view untuk selanjutnya diproses disana. Data yang diambil merupakan data – data tahun ini. Dan sebenarnya, diiawal pembuatan Kodok, halaman ini direncanakan memiliki filter data tahun berapa yang ingin dimunculkan, namun sampai saat ini filter tersebut belum juga diimplementasikan.

#### Controller\_ManagerListProjects

Controller ini berisikan function untuk memanage halaman List Project pada user Manager. Pada halaman ini, manager dapat mengedit informasi pada projek, menghapus data projek, dan melihat detail projek yang diisi oleh user Engineer.

Selain itu user Manager juga mampu mengekspor data projek sesuai dengan format yang sudah ditentukan di file ManagerProjectExport.php yang terletak di direktori Kodok/app/Exports.

Pada halaman ini juga terdapat filter untuk menampilkan tertentu saja yang ingin ditampilkan pada tabel.

### User Selain Admin, Manager, Engineer, dan Guest

Mungkin kalian bertanya – tanya, kenapa controllernya hanya ada untuk user Admin, Manager, Engineer, dan Guest padahal level usernya ada lebih dari itu?

Jawabannya adalah karena ada user yang belum dibuat / direncanakan halaman khususnya seperti user AM, PM dan Product. Maka dari itu di controller login, user – user tersebut akan langsung diarahkan untuk logout dari aplikasi Kodok. Kalau memang belum direncanakan, kenapa dibuat levelnya? Karena user – user tersebut perlu dibedakan saat mengisi data PIC bisnis pada projek. Namun jika kedepannya ingin ditambahkan halaman khusus maka akan lebih baik.

Dan ada juga user yang merupakan varian dari user yang sudah ada yaitu user Admin x Engineer, POH, dan Eksekutif. User Admin x Engineer merupakan varian gabungan dari user Admin dan Engineer, makanya mereka mendapat halaman chooseLogin saat pertama kali login ke aplikasi Kodok. User POH merupakan varian gabungan dari user Engineer dan setengah Manager, namun aksesnya hanya sekedar melihat List Project. Dan user Eksekutif merupakan varian user Manager yang hanya dapat melihat List Project dan Home.

Selain itu, keempat user yang memiliki controller sendiri merupakan empat user pertama yang ada di

# File Pendukung Lainnya

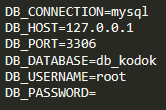
Di luar file – file yang sudah disebutkan sebelumnya, ada juga file yang tidak termasuk ke dalam kategori MVC yang perlu dijelaskan kegunaannya.

## script.js

File script.js merupakan file javascript yang menghandle proses javascript dan AJAX pada aplikasi Kodok. Semua perubahan data yang memerlukan update tabel secara langsung diproses di file ini menggunakan AJAX, yaitu delete data, proses approve dan decline, dan mengganti status projek. Selain AJAX, file ini juga mengatur respons saat user menekan tombol submit dan menampilkan modal ketika dibutuhkan seperti saat melihat detail projek. Untuk penjelasan lebih lanjut mengenai flownya akan dibahas di bab selanjutnya.

## .env

Untuk saat ini, file .env digunakan untuk mensetting database apa yang akan digunakan. Di development, database harus diset ke database local masing – masing developer sedangkan di production database harus diarahkan ke database yang ada di server production. Untuk melakukan setting database, tinggal mengisi data yang ada pada line berikut.



Seperti yang bisa dilihat pada Gambar, kita tinggal menyesuaikan variabel – variabel tersebut. DB\_HOST berisikan IP database, DB\_PORT ini biasanya port bebas, DB\_DATABASE merupakan nama databasenya, DB\_USERNAME dan DB\_PASSWORD adalah username password dari databasenya. Secara default isinya adalah username “root” dan password kosong.

## web.php

web.php merupakan file yang mengatur routing url di aplikasi Kodok. Semua url beserta methodnya disimpan di file ini.

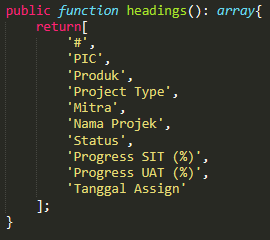


Gambar diatas merupakan salah satu contoh inisialisasi route pada file web.php. Route::get berarti url tersebut menggunakan method get. Lalu ‘/admin/approval/table’ adalah urlnya. ‘Controller\_AdminApprovalDocuments@dataTable’ adalah controller dan function apa yang harus dieksekusi saat mengakses url tersebut. Lalu ->name(‘approval.table’) merupakan nama yang diberikan untuk route tersebut. Kegunaan dari memberi nama supaya di view url tersebut bisa langsung dipanggil.

## Export

Export merupakan file atau kumpulan file yang mengatur format output ketika user melakukan export data dari aplikasi Kodok. Di Kodok sendiri kita menggunakan output berupa file excel. File – file export bisa ditemukan di direktori Kodok/app/Exports. File export sendiri merupakan library dari Maatwebsite.

Di aplikasi ini, file export memiliki 3 function. Yang pertama adalah function query() yang berisikan query yang akan dijalankan. Lalu ada function headings() yang mengenerate heading di file excelnya yang bisa dilihat pada Gambar dan Gambar. Dan function registerEvents() yang mengatur tabel pada excel, misalnya apakah tulisannya bold atau tidak, warna background, dan masih banyak lagi yang bisa dilakukan oleh function ini tergantung kebutuhannya.





## Import

Mirip dengan Export, Import merupakan file atau kumpulan file yang mengatur proses import data ke database. File – file import bisa ditemukan di direktori Kodok/app/Imports. File import juga library dari Maatwebsite.

Di dalam file import berisi satu function yaitu function model() dengan input array yang berisi data – data per baris. Lalu di dalamnya dicocokan heading pada excel dimasukkan ke kolom tabel yang mana.

# Interaksi Antar File

# Struktur Organisasi

Untuk saat ini, anggota tim Kodok dibagi menjadi tiga role, yaitu PM (Project Manager), Front End Developer dan Back End Developer.

PM bertugas untuk melakukan diskusi dengan manager terkait fitur – fitur apa saja yang dirasa dibutuhkan lalu menyampaikannya kepada programmer. Setelah itu melakukan penjadwalan dengan programmer mengenai fitur – fitur apa saja yang bisa dilakukan dalam satu sprint. Selain itu tugas PM adalah memastikan dalam satu sprint, fitur yang dikerjakan bisa selesai tepat waktu sesuai jadwal. Dan yang terkahir, PM juga bertugas untuk mengupdate user guide Kodok dan melakukan presentasi kepada seluruh anggota SDTL ketika fitur baru telah diimplementasikan. Selanjutnya PM membuat dokumentasi terkait fitur-fitur yang akan ditambah pada sprint bulan tersebut yang di tanda tangani oleh manager, senior pendamping, PM, dan programmer

Front End dan Back End Developer seperti namanya, bertanggung jawab atas pembangunan aplikasi Kodok di ranahnya masing – masing.

# Life Cycle

Proses migrasi dari environment development ke production Kodok berbeda ketika melakukan development yang sudah menggunakan git sebagai sarana versioningnya. Migrasi aplikasi Kodok dilakukan secara manual dengan membuat backup versi production sekarang menjadi file backup dan mengcopy folder Kodok terbaru dari laptop ke server (saat ini ada di server 29). Hal ini dilakukan karena github, tempat kita menyimpan repository aplikasi ini, tidak bisa diakses dari server Artajasa sehingga tidak bisa langsung pull dari versi yang paling baru. Namun sekarang repository sudah dipindah ke gitlab Artajasa, yang sepertinya memungkinkan proses migrasi dilakukan dengan cara melakukan pull. Namun cara ini hingga penulis membuat dokumentasi ini belum dicoba, jadi belum bisa dimasukkan ke versi dokumentasi yang sekarang.

Aplikasi Kodok sendiri berjalan pada sprint dengan rentang waktu satu bulan atau lebih tergantung fitur yang akan dikerjakan. Diawal bulan PM akan melakukan penjaringan aspirasi ke manager terkait fitur apa yang perlu ditambah. Lalu PM dan programmer melakukan konsultasi ke senior pendamping terkait fitur – fitur baru. Kemudian PM dan programmer mengadakan penjadwalan dan fitur apa saja yang akan dikembangkan pada sprint bulan tersebut. Setelah semua sudah ok, maka sprint sudah bisa berjalan dan di minggu terakhir bulan tersebut, diadakan migrasi. PM juga membuat dokumentasi yang ditanda tangani oleh manager, senior pendamping, PM, dan programmer terkait fitur apa saja yang benar-benar telah dimigrasikan