

**Rancang Bangun Aplikasi *Report Management* Berbasis Web**

**Dinas Komunikasi dan Informatika (Kominfo) Kota Madiun**

**Jalan Pahlawan No. 37 Madiun**

**Periode: 08 Juli – 08 Agustus 2019**

Oleh:

|  |  |
| --- | --- |
| Marde Fasma Ul’aza  Ulfatuz Zahroh | 05111640000046  05111640007004 |
|  |  |

Pembimbing Jurusan

Hadziq Fabroyir, S.Kom., Ph.D.

Pembimbing Lapangan

Noor Aflah, S.Kom.

DEPARTEMEN INFORMATIKA

Fakultas Teknologi Informasi dan Komunikasi

Institut Teknologi Sepuluh Nopember

Surabaya 2019

KERJA PRAKTIK – IF184981

*[Halaman ini sengaja dikosongkan]*



KERJA PRAKTIK – IF184981

**Rancang Bangun Aplikasi *Report Management* Berbasis Web**

**Dinas Komunikasi dan Informatika (Kominfo) Kota Madiun**

**Jalan Pahlawan No. 37 Madiun**

**Periode: 08 Juli – 08 Agustus 2019**

Oleh:

|  |  |
| --- | --- |
| Marde Fasma Ul’aza  Ulfatuz Zahroh | 05111640000046  05111640007004 |
|  |  |

Pembimbing Jurusan

Hadziq Fabroyir, S.Kom., Ph.D.

Pembimbing Lapangan

Noor Aflah, S.Kom.

DEPARTEMEN INFORMATIKA

Fakultas Teknologi Informasi dan Komunikasi

Institut Teknologi Sepuluh Nopember

Surabaya 2019

*[Halaman ini sengaja dikosongkan]*

# LEMBAR PENGESAHAN

**KERJA PRAKTIK**

**Rancang Bangun Aplikasi *Report Management* Berbasis Web**

Oleh:

|  |  |
| --- | --- |
| **Marde Fasma Ul’aza**  **Ulfatuz Zahroh** | 05111640000046  05111640007004 |
|  |  |

Mengetahui,

|  |  |
| --- | --- |
| Kepala Dinas Kominfo Kota Madiun,  **Subakri, S.Sos., M.Si.**  NIP. 19680904 198903 1 004 | Pembimbing Kerja Praktik,  **Noor Aflah, S.Kom.**  NIP. 19760907 200312 1 007 |

Menyetujui,

Dosen Pembimbing Kerja Praktik,

**Hadziq Fabroyir, S. Kom., Ph.D.**

NIP. 198602272019031006

**SURABAYA**

**SEPTEMBER, 2019**

*[Halaman ini sengaja dikosongkan]*

**Rancang Bangun Aplikasi *Report Management* Berbasis Web**

**Nama Mahasiswa : Marde Fasma Ul’aza**

**NRP : 05111640000046**

**Nama Mahasiswa : Ulfatuz Zahroh**

**NRP : 05111640007004**

**Jurusan : Informatika FTIK-ITS**

**Pembimbing Jurusan : Hadziq Fabroyir, S.Kom., Ph.D.**

**Pembimbing Lapangan: Noor Aflah, S.Kom.**

# ABSTRAK

Setiap Organisasi Perangkat Daerah (OPD) Kota Madiun mempunyai format yang berbeda dalam pembuatan laporan. Namun, tujuan pembuatan laporan tersebut sama, oleh karena itu dibutuhkan sistem *report management* untuk menyamakan format laporan di tiap OPD Kota Madiun, serta memudahkan OPD Kota Madiun dalam membuat laporan. OPD Kota Madiun berjumlah 59, namun dalam hal riset untuk pembuatan sistem *report management* ini digunakan 5 OPD, yaitu Bagian Administrasi Pembangunan (Adbang), Bagian Organisasi, Badan Perencanaan Pembangunan Daerah (Bappeda), Badan Pengelolaan Keuangan dan Aset Daerah (BPKAD), dan Inspektorat. Dilakukan analisis terkait basis data dan kebutuhan sistem dari laporan yang diberikan oleh 5 OPD tersebut. Kemudian, sistem *report management* berbasis web diimplementasikan menggunakan *framework* CodeIgniter.

**Kata kunci: *Report management,* OPD, Kota Madiun**

*[Halaman ini sengaja dikosongkan]*

# KATA PENGANTAR

Puji syukur kami haturkan kepada Allah SWT karena berkat rahmat-Nya kami dapat menyelesaikan salah satu mata kuliah wajib di Departemen Informatika, Institut Teknologi Sepuluh Nopember (ITS). Laporan kerja praktek ini disusun sebagai pelengkap kerja praktik yang telah dilaksanakan di Dinas Komunikasi dan Informatika (Kominfo) Kota Madiun selama 1 bulan.

Dengan selesainya laporan kerja praktik ini, tidak terlepas dari bantuan banyak pihak yang telah memberikan bantuan dan bimbingan. Kami menyampaikan terima kasih kepada pihak-pihak yang telah membantu dalam pelaksanaan kerja praktik, yaitu:

1. Kedua orang tua penulis.
2. Bapak Hadziq Fabroyir, S.Kom., Ph.D., selaku dosen pembimbing kerja praktik.
3. Bapak Radityo Anggoro, S.Kom., M.Sc., Dr.Eng., selaku koordinator Kerja Praktik.
4. Bapak Noor Aflah, S.Kom., selaku pembimbing lapangan kami di Dinas Komunikasi dan Informatika (Kominfo) Kota Madiun.
5. Seluruh staf di Dinas Kominfo serta staf di berbagai OPD di Kota Madiun yang membantu kelancaran Kerja Praktik.
6. Teman-teman kerja praktik dari berbagai perguruan tinggi.

Kami menyadari masih banyak kekurangan baik dalam pelaksanaan kerja praktik maupun penyusunan buku laporan ini. Namun, kami berharap buku laporan ini dapat menambaha wawasan pembaca dan dapat menjadi sumber referensi. Kami mengharapkan kritik dan saran yang membangun dari pembaca untuk kesempurnaan buku laporan ini.

Surabaya, September 2019

Marde Fasma Ul’aza – Ulfatuz Zahroh

# Daftar Isi

[LEMBAR PENGESAHAN v](#_Toc17577962)

[ABSTRAK 8](#_Toc17577963)

[KATA PENGANTAR 10](#_Toc17577964)

[Daftar Isi 11](#_Toc17577965)

[Daftar Gambar 14](#_Toc17577966)

[Daftar Tabel 16](#_Toc17577967)

[Daftar Kode Sumber 19](#_Toc17577968)

[BAB I PENDAHULUAN 23](#_Toc17577969)

[1.1 Latar Belakang 23](#_Toc17577970)

[1.2 Tujuan 23](#_Toc17577971)

[1.3 Manfaat 24](#_Toc17577972)

[1.4 Rumusan Masalah 24](#_Toc17577973)

[1.5 Lokasi dan Waktu Kerja Praktik 24](#_Toc17577974)

[1.6 Metodologi Kerja Praktik 25](#_Toc17577975)

[1.7 Sistematika Laporan 26](#_Toc17577976)

[BAB II PROFIL DINAS KOMINFO KOTA MADIUN 29](#_Toc17577977)

[2.1 Sejarah Dinas Kominfo Kota Madiun 29](#_Toc17577978)

[2.2 Tugas Pokok dan Fungsi Kominfo Kota Madiun 29](#_Toc17577979)

[2.3 Profil Bidang Pengelolaan Teknologi Informasi dan Komunikasi (TIK) 30](#_Toc17577980)

[2.4 Profil Seksi Layanan Aplikasi dan Tata Kelola Pemerintah Elektronik 31](#_Toc17577981)

[2.5 Struktur Organisasi Dinas Kominfo Kota Madiun 32](#_Toc17577982)

[BAB III TINJAUAN PUSTAKA 35](#_Toc17577983)

[3.1 Pemrograman Web 35](#_Toc17577984)

[BAB IV ANALISIS DAN PERANCANGAN SISTEM 39](#_Toc17577985)

[4.1 Analisis Sistem 39](#_Toc17577986)

[4.2 Perancangan Sistem 55](#_Toc17577987)

[BAB V IMPLEMENTASI SISTEM 94](#_Toc17577988)

[5.1 Implementasi Model 94](#_Toc17577989)

[5.2 Implementasi *Controller* 162](#_Toc17577990)

[5.3 Implementasi Antarmuka Pengguna 227](#_Toc17577991)

[BAB VI PENGUJIAN DAN EVALUASI 236](#_Toc17577992)

[6.1 Tujuan Pengujian 236](#_Toc17577993)

[6.2 Lingkungan Pengujian 236](#_Toc17577994)

[6.3 Identifikasi dan Rencana Pengujian 236](#_Toc17577995)

[6.4 Deskripsi dan Hasil Uji 240](#_Toc17577996)

[BAB VII KESIMPULAN dan SARAN 255](#_Toc17577997)

[7.1 Kesimpulan 255](#_Toc17577998)

[7.2 Saran 255](#_Toc17577999)

[DAFTAR PUSTAKA 257](#_Toc17578000)

[BIODATA PENULIS I 258](#_Toc17578001)

[BIODATA PENULIS II 260](#_Toc17578002)

*[Halaman ini sengaja dikosongkan]*

# Daftar Gambar

[Gambar 1 Struktur Organisasi Dinas Kominfo Kota Madiun 33](#_Toc18032265)

[Gambar 2 Sitemap User OPD 39](#_Toc18032266)

[Gambar 3 Use Case Diagram *Report Management* 40](#_Toc18032267)

[Gambar 4 Spesifikasi Use Case Melihat Data OPD 41](#_Toc18032268)

[Gambar 5 Spesifikasi Use Case Melihat Data User OPD 42](#_Toc18032269)

[Gambar 6 Spesifikasi Use Case Melihat Data Tipe Laporan OPD 44](#_Toc18032270)

[Gambar 7 Spesifikasi Use Case Menampilkan Log 46](#_Toc18032271)

[Gambar 8 Spesifikasi Use Case Melihat Laporan 48](#_Toc18032272)

[Gambar 9 Diagram Komponen *Report Management* 55](#_Toc18032273)

[Gambar 10 Diagram Deployment *Report Management* 56](#_Toc18032274)

[Gambar 11 Subsistem Model 56](#_Toc18032275)

[Gambar 12 Subsistem Controller 57](#_Toc18032276)

[Gambar 13 Subsistem View 57](#_Toc18032277)

[Gambar 14 PDM *Report Management* 58](#_Toc18032278)

[Gambar 15 Halaman Login 227](#_Toc18032279)

[Gambar 16 Halaman Home Admin 228](#_Toc18032280)

[Gambar 17 Halaman Melihat Data OPD 228](#_Toc18032281)

[Gambar 18 Halaman Melihat Data User OPD 229](#_Toc18032282)

[Gambar 19 Halaman Menambah Data User 230](#_Toc18032283)

[Gambar 20 Halaman Melihat Tipe Laporan OPD 230](#_Toc18032284)

[Gambar 21 Halaman Edit Akses User OPD 231](#_Toc18032285)

[Gambar 22 Halaman Melihat Log Aktifitas User OPD 232](#_Toc18032286)

[Gambar 23 Halaman Home User 232](#_Toc18032287)

[Gambar 24 Halaman Melihat Laporan 233](#_Toc18032288)

[Gambar 25 Halaman Mengedit Laporan 234](#_Toc18032289)

*[Halaman ini sengaja dikosongkan]*

# Daftar Tabel

[Tabel 1 Deskripsi Use Case 41](#_Toc18032318)

[Tabel 2 Kebutuhan Fungsional 54](#_Toc18032319)

[Tabel 3 Kebutuhan Non Fungsional 55](#_Toc18032320)

[Tabel 4 Tabel OPD 59](#_Toc18032321)

[Tabel 5 Tabel User 59](#_Toc18032322)

[Tabel 6 Tabel Log 60](#_Toc18032323)

[Tabel 7 Tabel Tipe Laporan 60](#_Toc18032324)

[Tabel 8 Tabel Tipe Laporan OPD 61](#_Toc18032325)

[Tabel 9 Tabel Laporan 61](#_Toc18032326)

[Tabel 10 Tabel Rekap Tender 62](#_Toc18032327)

[Tabel 11 Tabel Detail Rekap Tender 62](#_Toc18032328)

[Tabel 12 Tabel Rekap Pokja 63](#_Toc18032329)

[Tabel 13 Tabel Detail Rekap Pokja 63](#_Toc18032330)

[Tabel 14 Tabel Pegawai 64](#_Toc18032331)

[Tabel 15 Tabel Laporan Kinerja Triwulan 64](#_Toc18032332)

[Tabel 16 Tabel Detail Laporan Kinerja Triwulan 65](#_Toc18032333)

[Tabel 17 Tabel Pemantauan Tindak Lanjut 66](#_Toc18032334)

[Tabel 18 Tabel Temuan 67](#_Toc18032335)

[Tabel 19 Tabel Hasil Temuan 67](#_Toc18032336)

[Tabel 20 Tabel Realisasi Fisik 68](#_Toc18032337)

[Tabel 21 Tabel Program 69](#_Toc18032338)

[Tabel 22 Tabel Kegiatan 71](#_Toc18032339)

[Tabel 23 Tabel Jadwal Pelaksanaan 71](#_Toc18032340)

[Tabel 24 Tabel Jadwal Pelaksanaan OPD 72](#_Toc18032341)

[Tabel 25 Tabel Auditor 73](#_Toc18032342)

[Tabel 26 Tabel SOTK 73](#_Toc18032343)

[Tabel 27 Tabel SOTK OPD 74](#_Toc18032344)

[Tabel 28 Tabel IKM 74](#_Toc18032345)

[Tabel 29 Tabel IKM OPD 75](#_Toc18032346)

[Tabel 30 Tabel Pelayanan Publik 75](#_Toc18032347)

[Tabel 31 Tabel Pelayanan Publik OPD 76](#_Toc18032348)

[Tabel 32 Tabel Tatalaksana 76](#_Toc18032349)

[Tabel 33 Tabel Tatalaksana OPD 77](#_Toc18032350)

[Tabel 34 Tabel Monitoring Kelembagaan 78](#_Toc18032351)

[Tabel 35 Tabel Permasalahan Kelembagaan 78](#_Toc18032352)

[Tabel 36 Tabel Laporan RB Area Perubahan 79](#_Toc18032353)

[Tabel 37 Tabel RB Area Perubahan 80](#_Toc18032354)

[Tabel 38 Tabel RB Area Perubahan Program 80](#_Toc18032355)

[Tabel 39 Tabel RB Area Perubahan Kegiatan 81](#_Toc18032356)

[Tabel 40 Tabel Laporan RB Prioritas 82](#_Toc18032357)

[Tabel 41 Tabel RB Prioritas 82](#_Toc18032358)

[Tabel 42 Tabel RB Prioritas Sasaran 83](#_Toc18032359)

[Tabel 43 Tabel RB Prioritas Sasaran 84](#_Toc18032360)

[Tabel 44 Tabel Laporan RB Fokus 84](#_Toc18032361)

[Tabel 45 Tabel RB Fokus 85](#_Toc18032362)

[Tabel 46 Tabel RB Fokus Sasaran 86](#_Toc18032363)

[Tabel 47 Tabel RB Fokus Kegiatan 87](#_Toc18032364)

[Tabel 48 Tabel Laporan ZI WBK 87](#_Toc18032365)

[Tabel 49 Tabel RB ZI WBK Sasaran 88](#_Toc18032366)

[Tabel 50 Tabel RB ZI WBK Kegiatan 89](#_Toc18032367)

[Tabel 51 Tabel Laporan RB Quick Wins 89](#_Toc18032368)

[Tabel 52 Tabel RB Quick Wins 90](#_Toc18032369)

[Tabel 53 Tabel RB Quick Wins Sasaran 91](#_Toc18032370)

[Tabel 54 Tabel RB Quick Wins Kegiatan 92](#_Toc18032371)

[Tabel 55 Lingkungan Pengujian 236](#_Toc18032372)

[Tabel 56 Identifikasi dan Rencana Pengujian 238](#_Toc18032373)

[Tabel 57 Data Test Pencarian Data 239](#_Toc18032374)

[Tabel 58 Data Test Penambahan Data 239](#_Toc18032375)

[Tabel 59 Data Test Perubahan Data 239](#_Toc18032376)

[Tabel 60 Melihat Data OPD 240](#_Toc18032377)

[Tabel 61 Melihat Data User OPD 241](#_Toc18032378)

[Tabel 62 Melihat Tipe Laporan OPD 241](#_Toc18032379)

[Tabel 63 Melihat Log Aktifitas User OPD 242](#_Toc18032380)

[Tabel 64 Melihat Laporan 242](#_Toc18032381)

[Tabel 65 Pencarian Data OPD 243](#_Toc18032382)

[Tabel 66 Pencarian Data User OPD 244](#_Toc18032383)

[Tabel 67 Pencarian Data Tipe Laporan OPD 245](#_Toc18032384)

[Tabel 68 Pencarian Data Log Aktifitas User OPD 246](#_Toc18032385)

[Tabel 69 Pencarian Riwayat Laporan 247](#_Toc18032386)

[Tabel 70 Penambahan User OPD 248](#_Toc18032387)

[Tabel 71 Pembuatan Laporan 249](#_Toc18032388)

[Tabel 72 Melakukan Reset Password OPD 249](#_Toc18032389)

[Tabel 73 Melakukan Edit Akses OPD Terhadap Laporan 250](#_Toc18032390)

[Tabel 74 Melakukan Edit Laporan 251](#_Toc18032391)

[Tabel 75 Menghapus User OPD 252](#_Toc18032392)

[Tabel 76 Menghapus Laporan 252](#_Toc18032393)

[Tabel 77 Mencetak Laporan 253](#_Toc18032394)

*[Halaman ini sengaja dikosongkan]*

# Daftar Kode Sumber

[Kode Sumber 1 Model Activitylog 94](#_Toc18032120)

[Kode Sumber 2 Model Api\_ekin 95](#_Toc18032121)

[Kode Sumber 3 Model Api\_inaproc\_rup 97](#_Toc18032122)

[Kode Sumber 4 Model Api\_sikd 99](#_Toc18032123)

[Kode Sumber 5 Model Api\_sipp 103](#_Toc18032124)

[Kode Sumber 6 Model Dbbackup 104](#_Toc18032125)

[Kode Sumber 7 Model Dbf 106](#_Toc18032126)

[Kode Sumber 8 Model Laporan 108](#_Toc18032127)

[Kode Sumber 9 Model Opd 110](#_Toc18032128)

[Kode Sumber 10 Model Tipelaporan 111](#_Toc18032129)

[Kode Sumber 11 Model Tipelaporanopd 112](#_Toc18032130)

[Kode Sumber 12 Model User 113](#_Toc18032131)

[Kode Sumber 13 Model Ikm 115](#_Toc18032132)

[Kode Sumber 14 Model Jadwalpelaksanaan 119](#_Toc18032133)

[Kode Sumber 15 Model Laporankinerjatriwulan 121](#_Toc18032134)

[Kode Sumber 16 Model Laporanrbareaperubahan 125](#_Toc18032135)

[Kode Sumber 17 Model Laporanrbfokus 129](#_Toc18032136)

[Kode Sumber 18 Model Laporanrbprioritas 133](#_Toc18032137)

[Kode Sumber 19 Model Laporanrbquickwins 137](#_Toc18032138)

[Kode Sumber 20 Model Laporanrbziwbk 141](#_Toc18032139)

[Kode Sumber 21Model Monitoringkelembagaan 144](#_Toc18032140)

[Kode Sumber 22 Model Pelayananpublik 145](#_Toc18032141)

[Kode Sumber 23 Model Pemantauantindaklanjut 149](#_Toc18032142)

[Kode Sumber 24 Model Realisasifisik 152](#_Toc18032143)

[Kode Sumber 25 Modelrekappokja 155](#_Toc18032144)

[Kode Sumber 26 Model Rekaptender 158](#_Toc18032145)

[Kode Sumber 27 Model Sotk 160](#_Toc18032146)

[Kode Sumber 28 Model Tatalaksana 162](#_Toc18032147)

[Kode Sumber 29 Controller Admin 165](#_Toc18032148)

[Kode Sumber 30 Controller Apicheck 166](#_Toc18032149)

[Kode Sumber 31 Controller Auth 168](#_Toc18032150)

[Kode Sumber 32 Controller Database 169](#_Toc18032151)

[Kode Sumber 33 Controller Opd 173](#_Toc18032152)

[Kode Sumber 34 Controller Ikm 175](#_Toc18032153)

[Kode Sumber 35 Controller Jadwal\_pelaksanaan 178](#_Toc18032154)

[Kode Sumber 36 Controller Laporan\_kinerja\_triwulan 181](#_Toc18032155)

[Kode Sumber 37 Controller Laporan\_rb\_area\_perubahan 185](#_Toc18032156)

[Kode Sumber 38 Controller Laporan\_rb\_fokus 189](#_Toc18032157)

[Kode Sumber 39 Controller Laporan\_rb\_prioritas 194](#_Toc18032158)

[Kode Sumber 40 Controller Laporan\_rb\_quick\_wins 199](#_Toc18032159)

[Kode Sumber 41Controller Laporan\_rb\_zi\_wbk 203](#_Toc18032160)

[Kode Sumber 42 Controller Monitoring\_kelembagaan 205](#_Toc18032161)

[Kode Sumber 43 Controller Pelayanan\_publik 208](#_Toc18032162)

[Kode Sumber 44 Controller Pemantauan\_tindak\_lanjut 211](#_Toc18032163)

[Kode Sumber 45 Controller Realisasi\_fisik 216](#_Toc18032164)

[Kode Sumber 46 Controller Rekap\_pokja 219](#_Toc18032165)

[Kode Sumber 47 Controller Rekap\_tender 222](#_Toc18032166)

[Kode Sumber 48 Controller Sotk 224](#_Toc18032167)

[Kode Sumber 49 Controller Tatalaksana 226](#_Toc18032168)

*[Halaman ini sengaja dikosongkan]*

# BAB I PENDAHULUAN

## Latar Belakang

Perkembangan teknologi informasi semakin membantu manusia dalam menjalankan aktifitas sehari-hari. Saat ini, hamper seluruh kegiatan manusia tidak terlepas dari peran teknologi informasi. Seiring berkembangnya teknologi informasi, kebutuhan akan pembuatan aplikasi terus meningkat, salah satunya di Dinas Komunikasi dan Informatika (Kominfo) Kota Madiun.

Dinas Komunikasi dan Informatika (Kominfo) Kota Madiun adalah salah satu Organisasi Perangkat Daerah (OPD) yang ada di Kota Madiun. Dinas Komunikasi dan Informatika (Kominfo) Kota Madiun dibagi menjadi 2 bidang, yaitu Bidang Pengelolaan Informasi dan Komunikasi Publik dan Bidang Pengelolaan Teknologi Informasi dan Komunikasi (TIK).

Bagian Pengelolaan Informasi dan Komunikasi (TIK) yang mengurus masalah OPD yang berhubungan dengan teknologi informasi. Salah satu masalah OPD yaitu perbedaan format dalam pembuatan laporan. Masalah ini muncul karena laporan tiap OPD masih dibuat secara manual, sehingga masih ada kemungkinan *human error.*

Untuk mengatasi permasalahan tersebut, maka perlu dibuatkan sistem *report management* untuk menyamakan format pembuatan laporan seluruh OPD di Kota Madiun, serta memudahkan OPD dalam pembuatan laporan.

## Tujuan

Tujuan dari pembuatan sistem *report management* dalam pelaksanaan kerja praktik ini yaitu:

1. Untuk menyamakan format laporan seluruh OPD di Kota Madiun.
2. Untuk memudahkan OPD dalam pembuatan laporan.
3. Optimasi sistem dari cara manual menjadi terkomputasi dan terpusat.

## Manfaat

Manfaat dari pembuatan sistem *report management* dalam pelaksanaan kerja praktik ini adalah:

1. Format laporan di setiap OPD di Kota Madiun sama.
2. Dapat mempermudah OPD dalam pembuatan laporan.
3. Lebih mudah dalam mengakses laporan yang dibuat OPD.
4. Dapat memudahkan pengelolaan data dari aplikasi yang digunakan OPD.

## Rumusan Masalah

Berikut ini adalah rumusan masalah pada kerja praktik pembuatan aplikasi *report management*:

1. Bagaimana membuat sistem *report management* bisa diakses dimana saja?
2. Bagaimana keterkaitan sistem *report management* dengan sistem yang sudah ada di Dinas Komunikasi dan Informatika (Kominfo) Kota Madiun?

## Lokasi dan Waktu Kerja Praktik

Pelaksanaan tugas yang diberikan pada saat kerja praktik ini dilakukan di kantor pada saat jam kerja dengan rincian waktu dan tempat sebagai berikut:

Lokasi :Dinas Komunikasi dan Informatika (Kominfo) Kota Madiun

Tugas :Melaksanakan kebijakan, evaluasi, dan bimbingan teknis di Bidang Komunikasi dan Informatika sesuai dengan peraturan perundang-undangan.

Alamat : Jalan Pahlawan No. 37 Madiun, Jawa TImur, 63116

Tanggal : 08 Juli – 08 Agustus 2019

Hari Kerja : Senin – Jumat

Jam Kerja : 07.00 - 15.30 WIB

## Metodologi Kerja Praktik

Tahapan pengerjaan kerja praktik ini dapat dijabarkan sebagai berikut:

1. **Perumusan Masalah**

Penjelasan mengenai masalah diberikan oleh pembimbing lapangan kerja praktik, yaitu sistem *report management*. Dari penjelasan tersebut dihasilkan catatan-catatan penting mengenai gambaran sistem berbasis web yang akan dibuat. Dari catatan-catatan tersebut, kami memutuskan untuk membuat web dengan *framework* CodeIgniter. Selain itu, sebelum dibuatnya sistem ini, sudah ada beberapa sistem yang digunakan Pemerintah Kota Madiun, sehingga kami juga menyesuaikan dengan sistem yang sudah ada.

1. **Studi Literatur**

Setelah menentukan *framework* yang digunakan, kami melakukan studi literatur mengenai cara implementasi sistem sesuai dengan kebutuhan. Proses analisa kebutuhan ini dengan meminta contoh laporan pada tiap OPD di Kota Madiun.

1. **Analisis dan Perancangan Sistem**

Tahap ini mendefinisikan alur dari implementasi. Langkah-langkah yang dikerjakan juga didefinisikan ditahap ini, seperti:

1. Menganalisis kebutuhan sistem *report management.*
2. Menganalisis basis data sistem *report management* dari data laporan yang diberikan OPD.
3. Memodelkan basis data sistem *report management*
4. Menganalisis keterkaitan sistem *report management* dengan sistem yang sudah digunakan di Pemerintah Kota Madiun*.*
5. **Implementasi Sistem**

Tahap implementasi sistem didasarkan pada perancangan dan analisis sebelumnya. Dalam mengimplementasikan sistem *report management,* digunakan *framework* CodeIgniter. Pada tahap ini ada 2 pekerjaan utama, yaitu membuat *front-end ­*dan *back-end.* Pengerjaan web dilakukan di computer local, lalu di *deploy* ke server yang disediakan oleh Dinas Kominfo Kota Madiun.

1. **Pengujian dan Evaluasi**

Pengujian dilakukan dengan menguji fitur-fitur yang telah dibuat. Kesesuaian sistem dengan kebutuhan akan menentukan keberhasilan dalam pengujian. Hal ini akan menentukan apakah hasil evaluasi terhadap sistem sudah sesuai dengan tujuan dan kebutuhan sistem yang sudah di analisis.

1. **Kesimpulan dan Saran**

Kesimpulan yang kami dapatkan adalah masalah perbedaan format laporan tiap OPD di Kota Madiun yang mengurangi efisiensi pekerjaan dalam pengelolaan laporan tiap OPD. Serta masalah banyaknya penggunaan aplikasi yang berbeda menyebabkan melakukan pengentrian data berulang kali. Masalah tersebut dapat terselesaikan dengan adanya sistem *report management.*

## Sistematika Laporan

Laporan kerja praktik ini terdiri dari tujuh bab, dengan rincian sebagai berikut:

1. **Bab I Pendahuluan**

Bab ini menjelaskan tentang latar belakang permasalahan, tujuan, manfaat, rumusan masalah, lokasi dan waktu kerja praktik, metodologi kerja praktik, serta sistematika pengerjaan laporan kerja praktik.

1. **Bab II Profil Instansi Pemerintahan**

Bab ini menjelaskan secara rinci tentang profil instansi pemerintahan tempat kami melaksanakan kerja praktik, yakni Dinas Komunikasi dan Informatika (Kominfo) Kota Madiun.

1. **Bab III Tinjauan Pustaka**

Bab ini menjelaskan mengenai tinjauan pustaka dan literatur yang digunakan dalam penyelesaian kerja praktik.

1. **Bab IV Analisis dan Perancangan Sistem**

Bab ini menjelaskan hasil pembelajaran atau analisa hal apa saja yang diperlukan dan harus diperhatikan dalam kerja praktik ini.

1. **Bab V Implementasi Sistem**

Bab ini menjelaskan tahap-tahap yang dilakukan untuk proses implementasi sistem.

1. **Bab VI Pengujian dan Evaluasi**

Bab ini menjelaskan tentang hasil pengujian dan evaluasi dari sistem yang telah dikembangkan selama pelaksanaan kerja praktik.

1. **Bab VII Kesimpulan dan Saran**

Bab ini mempaparkan kesimpulan yang dapat diambil dan juga saran selama pengerjaan kerja praktik.

*[Halaman sengaja dikosongkan]*

# BAB II PROFIL DINAS KOMINFO KOTA MADIUN

## Sejarah Dinas Kominfo Kota Madiun

Dinas Komunikasi dan Informatika Kota Madiun dulu bergabung dengan Dinas Perhubungan Kota Madiun yang bernama Dinas Perhubungan Komunikasi dan Informatika (Dishubkominfo) Kota Madiun yang berlokasi di Jalan Hayam Wuruk No. 62 Madiun. Berdasarkan penataan Organisasi Perangkat Daerah (OPD) pada tahun 2016, Dinas Komunikasi dan Informatika Kota Madiun dipisah dengan Dinas Perhubungan (Dishub) Kota Madiun. Lokasi kantor Dinas Komunikasi dan Informatika (Kominfo) Kota Madiun di Jalan Pahlawan No. 37 Madiun.

Dinas Komunikasi dan Informatika (Kominfo) Kota Madiun adalah salah satu bagian dari beberapa dinas yang ada di Kota Madiun. Dinas Kominfo dibagi menjadi 2 bidang, yaitu Bidang Pengelolaan Informasi dan Pelayanan Publik dan Bidang Pengelolaan Teknologi Informasi dan Komunikasi (TIK). Bidang Pengelolaan Informasi dan Komunikasi Publik dibagi menjadi 3 seksi, yaitu: Seksi Pengelolaan Informasi dan Statistik, Seksi Pengelolaan Komunikasi Publik, dan Seksi Pengelolaan Media Publik. Bidang TIK dibagi menjadi 3 seksi, yaitu: Seksi Layanan Aplikasi dan Tata Kelola Pemerintahan Elektronik, Seksi Pengelolaan Infrastruktur dan Persandian, dan Seksi Pengembangan Sumber Daya TIK.

## Tugas Pokok dan Fungsi Kominfo Kota Madiun

### Tugas Kominfo Kota Madiun

Tugas Pokok Kominfo Kota Madiun adalah melaksanakan kebijakan, evaluasi, dan bimbingan teknis di Bidang Komunikasi dan Informatika sesuai dengan peraturan perundang-undangan.

### Fungsi Kominfo Kota Madiun

Berikut adalah fungsi-fungsi Kominfo Kota Madiun:

1. Penyusunan rumusan kebijakan teknis di bidang urusan komunikasi dan informatika, statistik, dan persandian berdasarkan peraturan perundang-undangan.
2. Penyelenggaraan urusan pemerintahan dan pelayanan umum di bidang komunikasi dan informatika, statistik, dan persandian.
3. Pembinaan dan pelaksanaan tugas di bidang urusan komunikasi dan informatika, statistic, dan persandian.
4. Pelaksanaan evaluasi dan pelaporan di bidang urusan komunikasi dan informatika, statistic, dan persandian sesuai peraturan perundang-undangan.
5. Pelaksanaan administrasi Dinas.
6. Pelaksanaan tugas dinas lain yang diberikan oleh Walikota.

## Profil Bidang Pengelolaan Teknologi Informasi dan Komunikasi (TIK)

Tugas Bidang Pengelolaan Teknologi Informasi dan Komunikasi (TIK) adalah penyiapan perumusan dan pelaksanaan kebijakan, penyusunan norma, standar, prosedur dan kriteria, dan pemberian bimbingan teknis dan supervisi, serta pemantauan, evaluasi, dan pelaporan di bidang TIK.

Fungsi Bidang Pengelolaan Teknologi Informasi dan Komunikasi (TIK) adalah:

1. Penyusunan kebijakan teknis, perencanaan, program kerja dan evaluasi pelaksanaan tugas-tugas pada Bidang Pengelolaan TIK.
2. Pengelolaan layanan infrastruktur dasar data center, *disaster recovery center* dan Teknologi Informasi dan Komunikasi.
3. Pengelolaan layanan pengembangan intranet dan penggunaan akses internet.
4. Pengelolaan layanan sistem komunikasi intra Pemerintah Kota, layanan keamanan informasi *e-Government.*
5. Pengelolaan layanan manajemen data dan informasi *e-Government.*
6. Pengelolaan layanan pengembangan dan pengelolaan aplikasi generic dan spesifik dan suplemen yang terintegrasi, integrasi layanan publik dan kepemerintahan.
7. Penyelenggaraan ekosistem TIK *Smart City.*
8. Penyelenggaraan *Government Chief Information Officer* (GCIO) Pemerintah Kota.
9. Pengembangan sumber daya TIK Pemerintah Daerah dan masyarakat.
10. Pengelolaan layanan nama domain dan sub domain bagi lembaga, pelayanan publik dan kegiatan.
11. Penyelanggaraan Persandian untuk pengamanan informasi Pemerintah Daerah.
12. Melakukan tugas dinas lain yang diberikan oleh Kepala Dinas.

## Profil Seksi Layanan Aplikasi dan Tata Kelola Pemerintah Elektronik

Tugas Seksi Layanan Aplikasi dan Tata Kelola Pemerintah Elektronik adalah:

1. Melakukan penyusunan perencanaan program, pelaksanaan kebijakan, evaluasi dan pelaporan tugas-tugas pada Seksi Layanan Aplikasi dan Tata Kelola Pemerintah Elektronik.
2. Menyiapkan bahan perumusan, penyusunan norma, standar, prosedur dan kriteria, dan pemberian bimbingan teknis dan supervisi pada Seksi Aplikasi dan Tata Kelola Pemerintah Elektronik.
3. Menyiapkan bahan penyusunan standar format data dan informasi, walikota dan kebijakan.
4. Melaksanakan layanan *recovery* datan dan informasi.
5. Melaksanakan layanan pengelolaan data elektronik Pemerintahan.
6. Melaksanakan layanan pengembangan dan pemeliharaan aplikasi Pemerintahan dan pelayanan publik yang terintegrasi.
7. Melaksanakan layanan Pusat *Aplication Program Interface* (API) daerah.
8. Melaksanakan layanan pengembangan *Business Process Reengineering* pelayanan untuk mendukung layanan interoperabilitas dan interkonektivitas layanan publik di lingkungan Pemerintahan dan non Pemerintah (*Stakeholder Smart City*).
9. Melaksanakan layanan Sistem Informasi *Smart City.*
10. Melaksanakan layanan interaktif Pemerintah Daerah dan Pemerintah Pusat.
11. Melaksanakan tugas dinas lain yang diberikan oleh Kepala Bidang Pengelolaan TIK.

## Struktur Organisasi Dinas Kominfo Kota Madiun

Dinas Kominfo dibagi menjadi 2 bidang, yaitu Bidang Pengelolaan Informasi dan Pelayanan Publik dan Bidang Pengelolaan Teknologi Informasi dan Komunikasi (TIK). Bidang Pengelolaan Informasi dan Komunikasi Publik dibagi menjadi 3 seksi, yaitu: Seksi Pengelolaan Informasi dan Statistik, Seksi Pengelolaan Komunikasi Publik, dan Seksi Pengelolaan Media Publik. Bidang TIK dibagi menjadi 3 seksi, yaitu: Seksi Layanan Aplikasi dan Tata Kelola Pemerintahan Elektronik, Seksi Pengelolaan Infrastruktur dan Persandian, dan Seksi Pengembangan Sumber Daya TIK.



**Gambar 1 Struktur Organisasi Dinas Kominfo Kota Madiun**

*[Halaman ini sengaja dikosongkan]*

# BAB III TINJAUAN PUSTAKA

## Pemrograman Web

Aplikasi web adalah aplikasi yang mengunakan teknologi *browser* untuk menjalankannya dan yang diakses melalui jaringan komputer (Remick, 2011). Sedangkan pemrograman web adalah pemrograman yang digunakan untuk membangun aplikasi web.

* + 1. **Visual Studio Code**

Visual Studio Code adalah penyunting teks yang dirancang untuk mengolah potongan-potongan kode, *plugin*, dan *markup*. Visual Studio Code dapat digunakan sebagai *text editor* untuk berbagai bahasa pemrograman: mulai dari C, Java, Javascript, dan lain-lain. Visual Studio Code memiliki tampilan yang minimalis namun elegan. Terlepas dari sisi tampilan, Visual Studio Code diboboti dengan keakuratan dalam penulisan yang mengikuti algoritma dan aturan setiap bahasa pemrograman yang digunakan. Pada kerja praktik ini, Visual Studio Code digunakan untuk melakukan penyuntingan kode pada aplikasi web.

* + 1. **CSS (*Cascading Style Sheets*)**

CSS adalah kepanjangan dari "Cascading Style Sheets", yaitu bahasa yang digunakan untuk memberi sentuhan gaya atau corak tampilan (*style*) pada sebuah *element* atau struktur halaman web yang dibuat oleh HTML. CSS dikembangkan oleh komunitas internasional yang disebut World Wide Web Consortium (W3C). Komunitas tersebut berkontribusi memberi masukan, menulis dan mengembangan spesifikasi CSS untuk mendefinisikan standar bahasa CSS yang akan digunakan oleh semua orang. Spesifikasi CSS pertama kali diperkenalkan pada tahun 1996.

* + 1. **HTML (*HyperText Markup Language*)**

HTML adalah sebuah bahasa *markup* yang digunakan untuk membuat sebuah halaman web dan menampilkannya ke dalam sebuah *browser*.

* + 1. **PHP (PHP: *Hypertext Prepocessor)***

PHP adalah bahasa pemrograman yang sering disisipkan ke dalam HTML. PHP sendiri berasal dari kata *Hypertext Preprocessor.* Sejarah PHP pada awalnya merupakan kependekan dari *Personal Home Page.* PHP pertaman kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama Form Interpreted (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web.

Bahasa pemrograman ini menggunakan sistem server-side. *Server-side programming* adalah jenis Bahasa pemrograman yang nantinya *script* atau program dijalankan atau di proses oleh server. Kelebihannya adalah mudah digunakan, sederhana, dan mudah untuk dimengerti dan dipelajari.

* + 1. **Codeigniter**

CodeIgniter adalah sebuah web application network yang bersifat *open source* yang digunakan untuk membangun aplikasi PHP dinamis. CodeIgniter menjadi sebuah framework PHP dengan model MVC (Model, View, Controller) untuk membangun website dinamis dengan menggunakan PHP yang dapat mempercepat pengembangan untuk membuat sebuah aplikasi web. Selain ringan dan cepat, CodeIgniter juga memiliki dokumentasi yang lengkap disertai dengan contoh implementasi kodenya. Dokumentasi yang lengkap inilah yang menjadi salah satu alasan kenapa kita memilih menggunakan CodeIgniter.

* + 1. **Javascript**

JavaScript adalah bahasa pemrograman web yang bersifat *Client Side Programming Language*. *Client Side Programming Language* adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*. Aplikasi *client* yang dimaksud merujuk kepada *web browser* seperti Google Chrome atau Mozilla Firefox.

* + 1. **MySQL**

MySQL adalah sebuah *database management system* menggunakan perintah dasar SQL (*Structured Query language*) yang terkenal. MySQL sendiri merupakan DBMS yang *open source* dengan dua bentuk lisensi, yaitu Free Software (perangkat lunak bebas) dan Shareware (perangkat lunak berpemilik yang penggunaannya terbatas). Jadi MySQL adalah database server yang gratis dengan lisensi GNU General Public License (GPL) sehinngga bisa digunakan tanpa membayar.

* + 1. **Git**

Git adalah *version control system* yang digunakan developer untuk mengembangkan software secara bersama-sama. Fungsi utama git yaitu mengatur versi dari *source code* program anda dengan mengasih tanda baris dan code mana yang ditambah atau diganti. Selain itu, dengan git kita tidak perlu khawatir code yang kita kerjakan bentrok, karena setiap developer bisa membuat branch sebagai workspacenya.

*[Halaman ini sengaja dikosongkan]*

# BAB IV ANALISIS DAN PERANCANGAN SISTEM

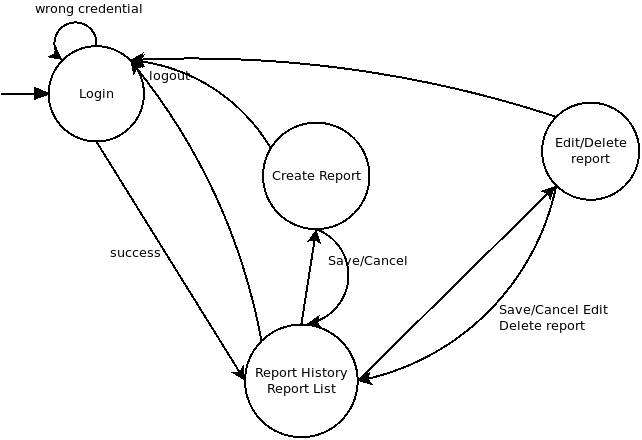
## Analisis Sistem

Bab ini menjelaskan tahapan pembangunan aplikasi *report management*. Tahapan yang dimaksud melingkupi analisa sistem yang akan dibangun. Penjelasan dibagi menjadi dua: definisi umum aplikasi dan analisis kebutuhan fungsional.

* + 1. **Definisi Umum Aplikasi**

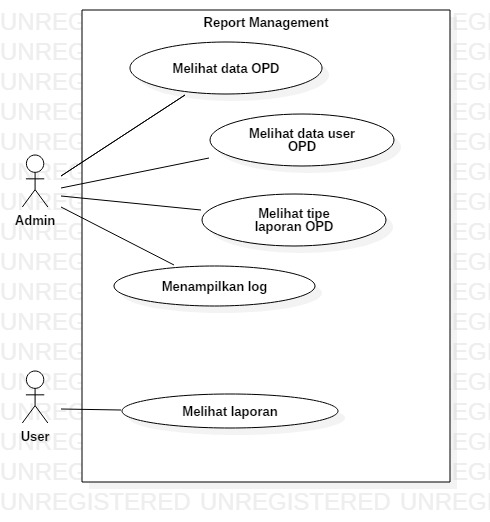
Secara umum, sistem *report management* merupakan sebuah sistem berbasis web yang dapat digunakan oleh OPD di Kota Madiun untuk membuat laporan. Sistem *report management* bisa diakses secara *online* melalui *browser.* Sistem ini dibuat untuk menyamakan format laporan yang dibuat oleh OPD di Kota Madiun, serta memudahkan OPD dalam pembuatan laporan.

Sistem *report management* mempunyai 2 tipe pengguna, *user* OPD dan admin. *User* OPD dapat membuat laporan, mengubah laporan, menghapus laporan, melihat riwayat laporan, dan meng-*export* laporan. Admin dapat melihat historis pembuatan laporan yang dibuat OPD, mengelola *user* OPD, dan mengelola jenis laporan yang dibuat oleh OPD.



**Gambar 2 Sitemap User OPD**

* + 1. **Analisis Kebutuhan**
       1. **Use Case Diagram**



**Gambar 3 Use Case Diagram *Report Management***

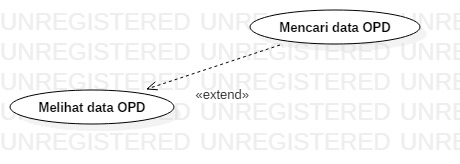
|  |  |  |
| --- | --- | --- |
| **ID** | **Nama Use Case** | **Deskripsi** |
| UC01 | Melihat data OPD | Admin dapat melihat data OPD |
| UC02 | Melihat data User OPD | Admin dapat melihat data User OPD |
| UC03 | Melihat tipe laporan OPD | Admin dapat melihat tipe laporan OPD |
| UC04 | Menampilkan log | Admin dapat melihat log aktifitas User OPD |
| UC05 | Melihat laporan | User OPD dapat melihat laporan |

**Tabel 1 Deskripsi Use Case**

* + - 1. **Spesifikasi Use Case**

1. **Melihat data OPD**
   1. Deskripsi

Kasus penggunaan ini mendeskripsikan Admin dapat melihat data Organisasi Perangkat Daerah (OPD) Kota Madiun.



**Gambar 4 Spesifikasi Use Case Melihat Data OPD**

* 1. Alur
     1. Alur Utama

1. Pada halaman utama, Admin klik menu Database.
2. Sistem menampilkan *dropdown* submenu.
3. Admin klik submenu OPD.
4. Sistem menampilkan data OPD.
   * 1. Alur Alternatif

-

* + 1. Eksepsi

-

* 1. *Special Requirements*

1. Hanya Admin yang telah terautentikasi dan terautorisasi yang dapat mengakses.
2. Data OPD terdiri dari nama OPD, kode user e-budgeting, dan kode user e-kinerja.
   1. Kondisi Awal

-

* 1. Kondisi Akhir

-

* 1. Ekstensi

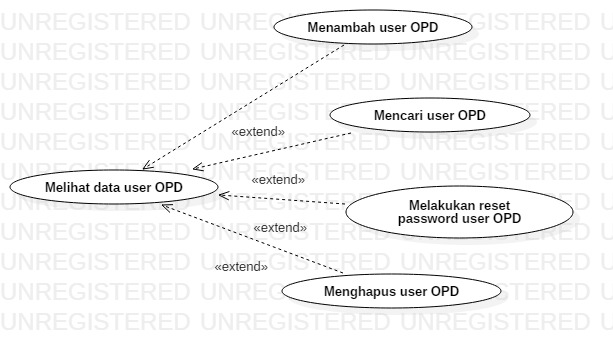
4a Admin dapat **Mencari data OPD**

4a.1 Admin menulis kata yang akan dicari pada kolom Search.

4a.2 Sistem menampilkan data OPD sesuai kata yang ditulis Admin.

1. **Melihat data User OPD**
   1. Deskripsi

Kasus penggunaan ini mendeskripsikan Admin dapat melihat data user Organisasi Perangkat Daerah (OPD) Kota Madiun.



**Gambar 5 Spesifikasi Use Case Melihat Data User OPD**

* 1. Alur
     1. Alur Utama

1. Pada halaman utama, Admin klik menu Database.
2. Sistem menampilkan *dropdown* submenu.
3. Admin klik submenu User.
4. Sistem menampilkan data user OPD.
   * 1. Alur Alternatif

-

* + 1. Eksepsi

-

* 1. *Special Requirements*

1. Hanya Admin yang telah terautentikasi dan terautorisasi yang dapat mengakses.
2. Data user OPD terdiri dari username, nama OPD, dan terakhir login.
   1. Kondisi Awal

-

* 1. Kondisi Akhir

-

* 1. Ekstensi

4a Admin dapat **Menambah user OPD**

4a.1 Admin klik tombol Tambah User.

4a.2 Sistem menampilkan form pengisian data User OPD.

4a.3 Admin mengisi data User OPD.

4a.4 Admin memilih tombol Submit.

4a.5 Sistem menampilkan notifikasi “Tambah user berhasil”.

4b Admin dapat **Mencari user OPD**

4b.1 Admin menulis kata yang akan dicari pada kolom Search.

4b.2 Sistem menampilkan data user OPD sesuai kata yang ditulis Admin.

4c Admin dapat **Melakukan reset password user OPD**

4c.1 Admin klik tombol Reset Pass.

4c.2 Sistem menampilkan notifikasi “Reset password berhasil”.

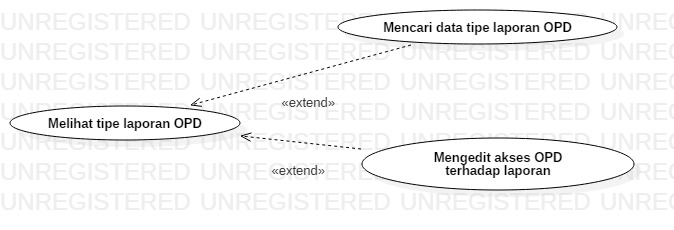
4d Admin dapat **Menghapus user OPD**

4d.1 Admin klik tombol Hapus.

4d.2 Sistem menampilkan notifikasi “Data telah dihapus”.

1. **Melihat tipe laporan OPD**
   1. Deskripsi

Kasus penggunaan ini mendeskripsikan Admin dapat melihat tipe laporan yang bisa dibuat oleh user Organisasi Perangkat Daerah (OPD) Kota Madiun.



**Gambar 6 Spesifikasi Use Case Melihat Data Tipe Laporan OPD**

* 1. Alur
     1. Alur Utama

1. Pada halaman utama, Admin klik menu Database.
2. Sistem menampilkan *dropdown* submenu.
3. Admin klik submenu Tipe OPD.
4. Sistem menampilkan data tipe laporan yang bisa dibuat oleh OPD.
   * 1. Alur Alternatif

-

* + 1. Eksepsi

-

* 1. *Special Requirements*

1. Hanya Admin yang telah terautentikasi dan terautorisasi yang dapat mengakses.
   1. Kondisi Awal

-

* 1. Kondisi Akhir

-

* 1. Ekstensi

4a Admin dapat **Mencari data tipe laporan OPD**

4a.1 Admin menulis kata yang akan dicari pada kolom Search.

4a.2 Sistem menampilkan data tipe laporan OPD sesuai kata yang ditulis Admin.

4b Admin dapat **Mengedit akses OPD terhadap laporan**

4b.1 Admin klik tombol simbol tambah pada kolom ID OPD.

4b.2 Sistem menampilkan data laporan yang bisa diakses oleh OPD.

4b.3 Admin klik tombol Edit Akses.

4b.4 Sistem menampilkan tipe laporan.

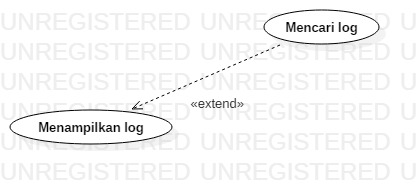
4b.5 Admin memilih laporan yang akan di *enable* akses untuk OPD.

4b.6 Admin klik tombol Simpan.

4b.7 Sistem menampilkan notifikasi “Update akses OPD berhasil”.

1. **Menampilkan log**
   1. Deskripsi

Kasus penggunaan ini mendeskripsikan Admin dapat melihat log aktifitas yang dilakukan oleh user Organisasi Perangkat Daerah (OPD) Kota Madiun.



**Gambar 7 Spesifikasi Use Case Menampilkan Log**

* 1. Alur
     1. Alur Utama

1. Pada halaman utama, Admin klik menu Query Log.
2. Sistem menampilkan *dropdown* submenu.
3. Admin klik submenu Query Log.
4. Sistem menampilkan data log aktifitas user OPD.
   * 1. Alur Alternatif

-

* + 1. Eksepsi

-

* 1. *Special Requirements*

1. Hanya Admin yang telah terautentikasi dan terautorisasi yang dapat mengakses.
2. Data log terdiri dari username, nama OPD, aksi, dan waktu.
   1. Kondisi Awal

-

* 1. Kondisi Akhir

-

* 1. Ekstensi

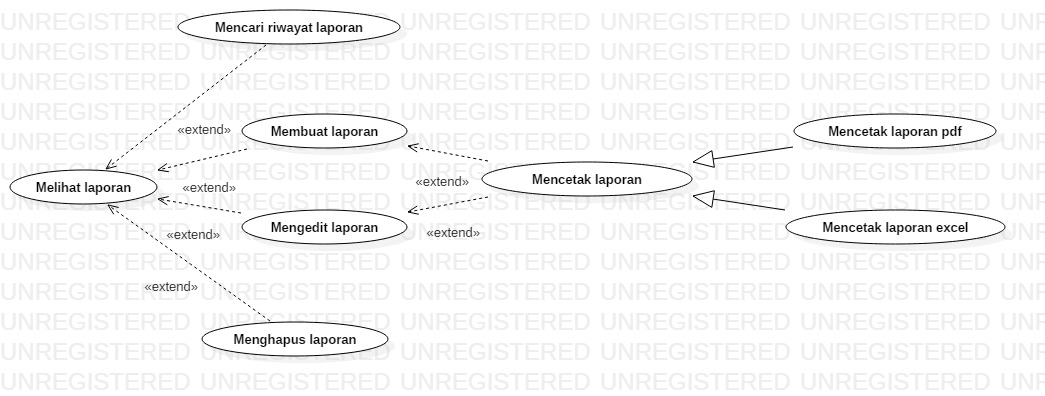
4a Admin dapat **Mencari log**

4a.1 Admin menulis kata yang akan dicari pada kolom Search.

4a.2 Sistem menampilkan log sesuai kata yang ditulis Admin.

1. **Melihat laporan**
   1. Deskripsi

Kasus penggunaan ini mendeskripsikan User Organisasi Perangkat Daerah (OPD) Kota Madiun dapat melihat laporan.



**Gambar 8 Spesifikasi Use Case Melihat Laporan**

* 1. Alur
     1. Alur Utama

1. Pada halaman utama, User OPD klik menu Laporan.
2. Sistem menampilkan *dropdown* submenu.
3. Admin klik salah satu submenu laporan.
4. Sistem menampilkan data laporan yang dipilih User OPD.
   * 1. Alur Alternatif

4b.3a-a User OPD **Mencetak laporan pdf**

4b.3a-a.1 User OPD klik tombol Print.

4b.3a-a.2 Sistem menampilkan *preview* laporan yang akan dicetak.

4b.3a-a.3 User OPD dapat mengatur destination, pages, layout, paper size, pages per sheet, margins, scale, dan options.

4b.3a-a.4 User OPD klik Save.

4b.3a-a.5 Sistem menampilkan *dialog box* Save As.

4b.3a-a.6 User OPD memilih lokasi penyimpanan file. Sistem hanya menampilkan file yang memiliki tipe file yang sama dengan tipe file hasil ekspor.

4b.3a-a.7 User OPD memilih file destinasi yang sudah tersedia di folder yang dipilih atau dapat mengetik nama file yang akan dibuat pada *text box.*

4b.3a-a.8 User OPD klik Save.

4b.3a-a.9 Sistem menyimpan file laporan pada lokasi yang dipilih User OPD.

4b.3a-b User OPD **Mencetak laporan excel**

4b.3a-b.1 User OPD klik tombol Excel.

4b.3a-b.2 Sistem menampilkan *dialog box* Save As.

4b.3a-b.3 User OPD memilih lokasi penyimpanan file. Sistem hanya menampilkan file yang memiliki tipe file yang sama dengan tipe file hasil ekspor.

4b.3a-b.4 User OPD memilih file destinasi yang sudah tersedia di folder yang dipilih atau dapat mengetik nama file yang akan dibuat pada *text box.*

4b.3a-b.5 User OPD klik Save.

4b.3a-b.6 Sistem menyimpan file laporan pada lokasi yang dipilih User OPD.

4c.2a-a User OPD **Mencetak laporan pdf**

4c.2a-a.1 User OPD klik tombol Print.

4c.2a-a.2 Sistem menampilkan *preview* laporan yang akan dicetak.

4c.2a-a.3 User OPD dapat mengatur destination, pages, layout, paper size, pages per sheet, margins, scale, dan options.

4c.2a-a.4 User OPD klik Save.

4c.2a-a.5 Sistem menampilkan *dialog box* Save As.

4c.2a-a.6 User OPD memilih lokasi penyimpanan file. Sistem hanya menampilkan file yang memiliki tipe file yang sama dengan tipe file hasil ekspor.

4c.2a-a.7 User OPD memilih file destinasi yang sudah tersedia di folder yang dipilih atau dapat mengetik nama file yang akan dibuat pada *text box.*

4c.2a-a.8 User OPD klik Save.

4c.2a-a.9 Sistem menyimpan file laporan pada lokasi yang dipilih User OPD.

4c.2a-b User OPD **Mencetak laporan excel**

4c.2a-b.1 User OPD klik tombol Excel.

4c.2a-b.2 Sistem menampilkan *dialog box* Save As.

4c.2a-b.3 User OPD memilih lokasi penyimpanan file. Sistem hanya menampilkan file yang memiliki tipe file yang sama dengan tipe file hasil ekspor.

4c.2a-b.4 User OPD memilih file destinasi yang sudah tersedia di folder yang dipilih atau dapat mengetik nama file yang akan dibuat pada *text box.*

4c.2a-b.5 User OPD klik Save.

4c.2a-b.6 Sistem menyimpan file laporan pada lokasi yang dipilih User OPD.

* + 1. Eksepsi

4c.3a Jika User OPD tidak klik Submit, maka sistem tidak menyimpan data apapun.

4c.3b Jika User OPD klik Hapus, maka data yang sudah diinput akan dihapus.

* 1. *Special Requirements*

1. Hanya User OPD yang telah terautentikasi dan terautorisasi yang dapat mengakses.
2. File hasil cetak laporan berekstensi xlsx dan pdf.
3. File hasil cetak laporan hanya bisa disimpan pada perangkat yang digunakan atau media penyimpanan eksternal yang terhubung dengan perangkat yang sama.
   1. Kondisi Awal
4. Edit dan hapus laporan hanya bisa dilakukan ketika laporan sudah dibuat.
5. File destinasi untuk cetak file tidak dalam keadaan terbuka di program lain.
   1. Kondisi Akhir
6. Setelah mencetak file, file destinasi berisi data laporan yang dicetak.
   1. Ekstensi

4a User OPD dapat **Mencari riwayat laporan**

4a.1 User OPD melakukan set rentang tanggal yang pencarian laporan.

4a.2 User OPD klik tombol Cari.

4a.3 Sistem menampilkan laporan dalam rentang waktu yang diinput.

4b User OPD dapat **Membuat laporan**

4b.1 User OPD melakukan set tanggal pembuatan laporan.

4b.2 User OPD klik tombol buat laporan.

4b.3 Sistem menampilkan laporan yang dibuat.

4b.3a User OPD dapat **Mencetak laporan**

4c User OPD dapat **Mengedit laporan**

4c.1 User OPD klik Edit.

4c.2 Sistem menampilkan data laporan yang akan di edit.

4c.3 User OPD mengedit laporan.

4c.4 User OPD klik tombol Submit.

4c.2a User OPD dapat **Mencetak laporan**

4d User OPD dapat **Menghapus laporan**

4d.1 User OPD klik Hapus.

4d.2 Sistem menghapus laporan dari dalam database.

* + - 1. **Kebutuhan Fungsional**

Kebutuhan fungsional dalam aplikasi menjelaskan bagaimana sistem itu bekerja. Kebutuhan fungsional dari aplikasi *Report Management* yaitu:

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Deskripsi** | **UC** | **Prioritas** |
| F01 | Admin dapat melihat data OPD | UC01 | Must |
| F02 | Admin dapat mencari data OPD | UC01 | Optional |
| F03 | Sistem menampilkan *dropdown* submenu | UC01, UC02, UC03, UC04, UC05 | Must |
| F04 | Sistem menampilkan data dalam database | UC01, UC02, UC03, UC04, UC05 | Must |
| F05 | Admin dapat melihat data User OPD | UC02 | Must |
| F06 | Admin dapat menambah User OPD | UC02 | Must |
| F07 | Admin dapat mencari User OPD | UC02 | Optional |
| F08 | Admin dapat melakukan reset password User OPD | UC02 | Must |
| F09 | Admin dapat menghapus User OPD | UC02 | Must |
| F10 | Sistem menampilkan form pengisian data | UC02, UC05 | Must |
| F11 | Sistem menampilkan notifikasi aktifitas selesai dilakukan | UC02, UC03 | Optional |
| F12 | Admin dapat melihat tipe laporan OPD | UC03 | Must |
| F13 | Admin dapat mencari data tipe laporan OPD | UC03 | Must |
| F14 | Admin dapat mengedit akses OPD terhadap laporan | UC03 | Must |
| F15 | Admin dapat melihat log aktifitas User OPD | UC04 | Must |
| F16 | Admin dapat mencari data log aktifitas User OPD | UC04 | Optional |
| F17 | User OPD dapat melihat laporan | UC05 | Must |
| F18 | User OPD dapat mencari riwayat laporan | UC05 | Must |
| F19 | User OPD dapat membuat laporan | UC05 | Must |
| F20 | User OPD dapat mengedit laporan | UC05 | Must |
| F21 | User OPD dapat menghapus laporan | UC05 | Must |
| F22 | User OPD dapat mencetak laporan | UC05 | Must |
| F23 | Sistem menampilkan *preview* file laporan yang akan dicetak | UC05 | Must |
| F24 | Sistem menampilkan *dialog box* Save As | UC05 | Must |
| F25 | Sistem menyimpan file laporan pada lokasi yang dipilih User OPD | UC05 | Must |
| F26 | Sistem dapat menghapus data dalam database | UC02, UC03, UC05 | Must |

**Tabel 2 Kebutuhan Fungsional**

* + - 1. **Kebutuhan Non-Fungsional**

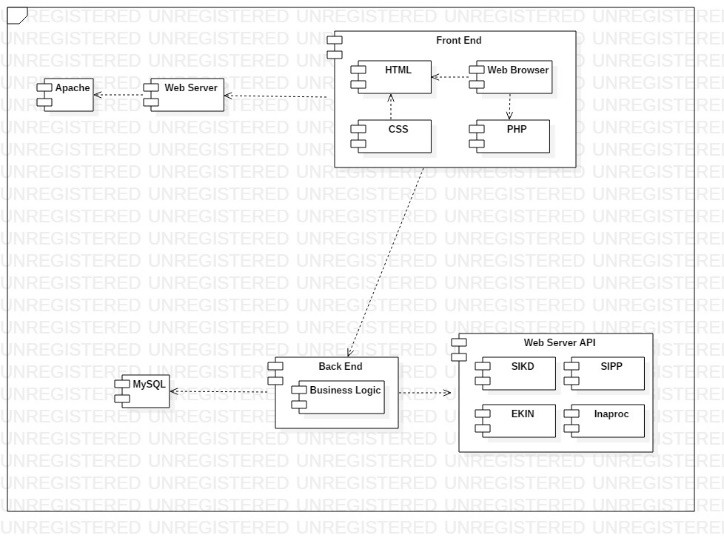
Kebutuhan non-fungsional adalah kebutuhan pengguna untuk mendefinisikan bagaimana batasan dan karakteristik dari sebuah sistem. Kebutuhan non-fungsional dari *Report Management* yaitu,

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Deskripsi** | **UC** | **Kualitas** | **Prioritas** |
| NF01 | Hanya Admin yang terautentikasi dan terautorisasi yang dapat mengakses. | UC01, UC02, UC03, UC04 | Security  (Authenticity, Integrity, Confidentiality) | Must |
| NF02 | Data OPD terdiri dari nama OPD, kode user e-budgeting, dank ode user e-kinerja. | UC01 | Performance Efficiency  (Resource Utilization) | Must |
| NF03 | Data User OPD terdiri dari username, nama OPD, dan terakhir login. | UC02 | Performance Efficiency  (Resource Utilization) | Must |
| NF04 | Data log terdiri dari username, nama OPD, aksi, dan waktu. | UC03 | Performance Efficiency  (Resource Utilization) | Must |
| NF05 | Hanya User OPD yang terautentikasi dan terautorisasi yang dapat mengakses. | UC05 | Security  (Authenticity, Integrity, Confidentiality) | Must |
| NF06 | File hasil cetak laporan berekstensi xlsx dan pdf. | UC05 | Portability  (Adaptability) | Must |
| NF07 | File hasil cetak laporan hanya bisa disimpan pada perangkat yang digunakan atau media penyimpanan eksternal yang terhubung dengan perangkat yang sama. | UC05 | Portability  (Adaptability) | Must |

**Tabel 3 Kebutuhan Non Fungsional**

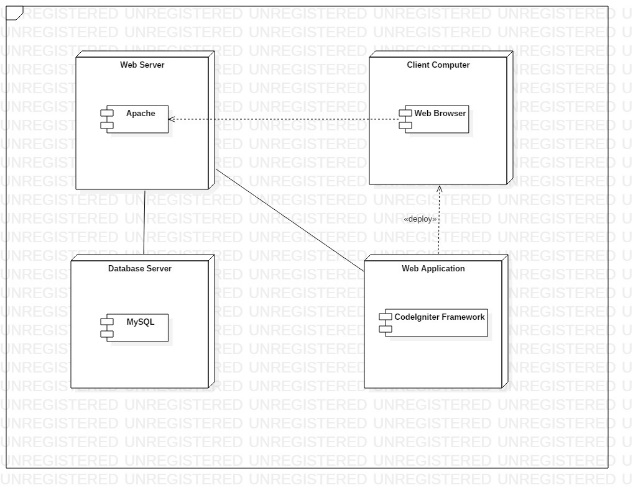
## Perancangan Sistem

* + 1. **Arsitektur Sistem**
       1. **Diagram Komponen**

****

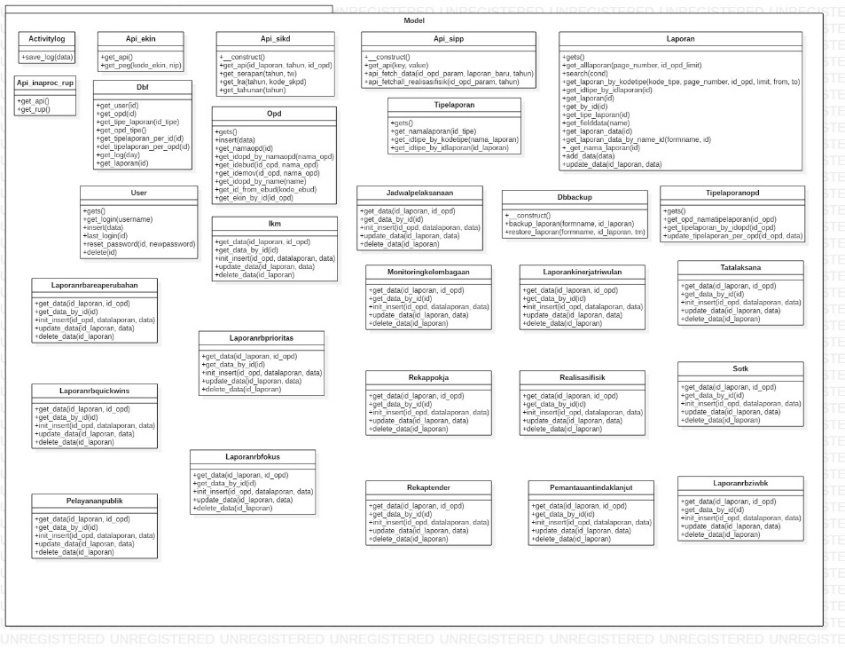
**Gambar 9 Diagram Komponen *Report Management***

* + - 1. **Diagram Deployment**

****

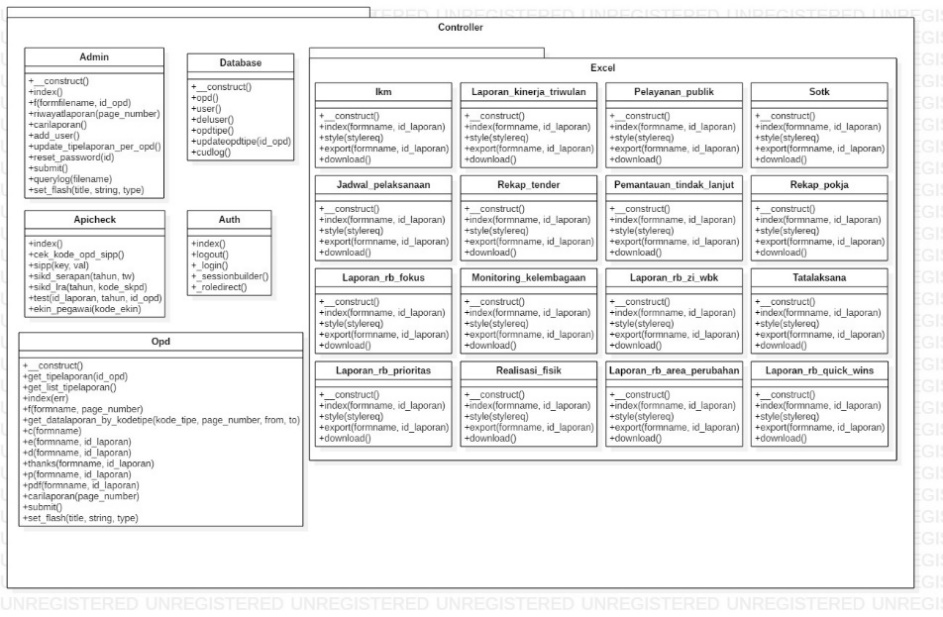
**Gambar 10 Diagram Deployment *Report Management***

* + 1. **Dekomposisi Sistem**
       1. **Subsistem 1: Model**

****

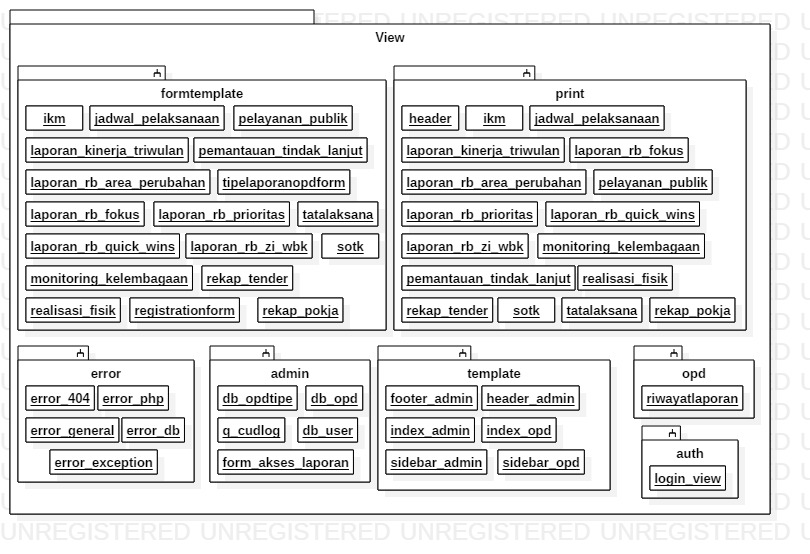
**Gambar 11 Subsistem Model**

* + - 1. **Subsistem 2: Controller**

****

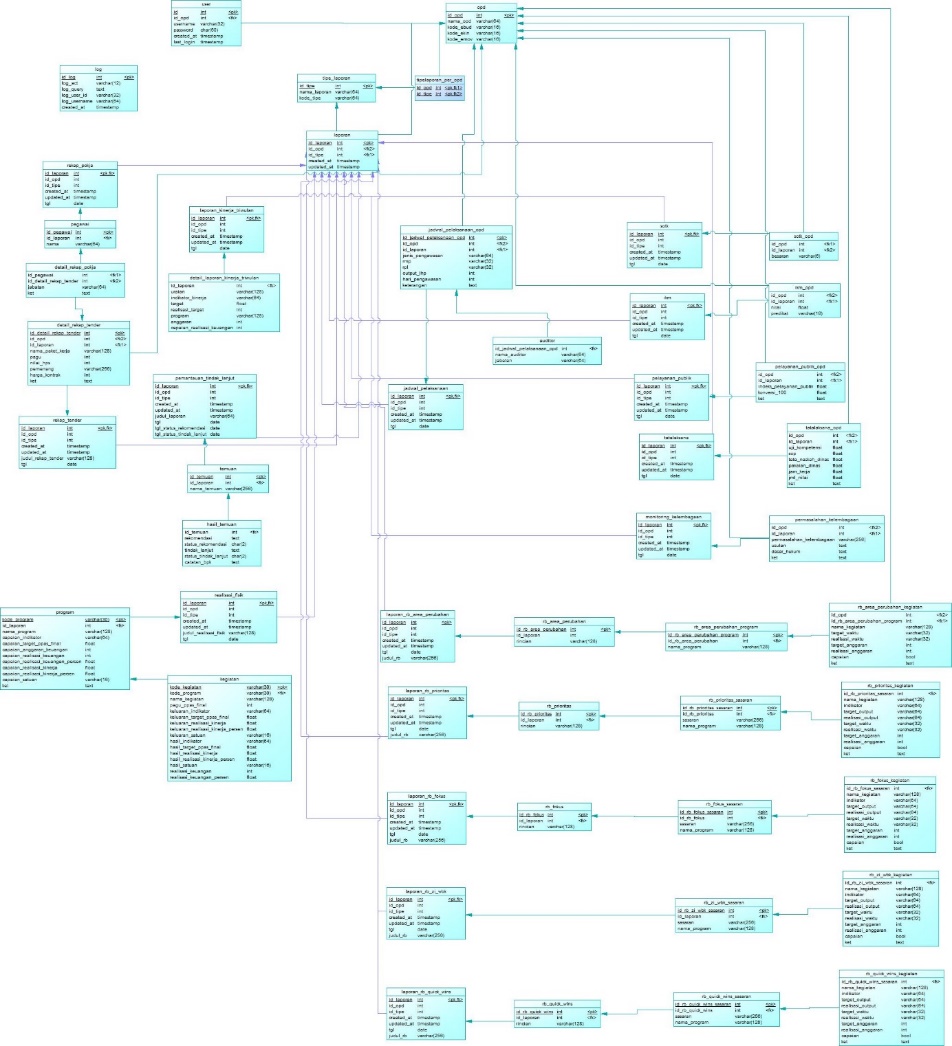
**Gambar 12 Subsistem Controller**

* + - 1. **Subsistem 3: View**

****

**Gambar 13 Subsistem View**

* + 1. ***Physical Data Model***



**Gambar 14 PDM *Report Management***

* + 1. **Daftar Tabel Aplikasi**
       1. **Tabel OPD**

Tabel untuk menyimpan data Organisasi Perangkat Daerah (OPD) Kota Madiun.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_opd | Variable untuk menyimpan id opd | INTEGER | NO | Primary key |
| nama\_opd | Variable untuk menyimpan nama OPD | VARCHAR(64) |  |  |
| kode\_ebud | Variable untuk menyimpan kode e-budgeting | VARCHAR(16) |  |  |
| kode\_ekin | Variable untuk menyimpan kode e-kinerja | VARCHAR(16) |  |  |
| kode\_emov | Variable untuk menyimpan kode e-monev | VARCHAR(16) |  |  |

**Tabel 4 Tabel OPD**

* + - 1. **Tabel User**

Tabel untuk menyimpan data pengguna.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id | Variable untuk menyimpan id pengguna | INTEGER | NO | Primary key |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | Foreign key dari tabel opd |
| username | Variable untuk menyimpan username pengguna | VARCHAR(32) |  |  |
| password | Variable untuk menyimpan password pengguna | CHAR(60) |  |  |
| created\_at | Variable untuk menyimpan tanggal pendaftaran pengguna | TIMESTAMP |  |  |
| last\_login | Variable untuk menyimpan tanggal terakhir pengguna melakukan login | TIMESTAMP |  |  |

**Tabel 5 Tabel User**

* + - 1. **Tabel Log**

Tabel untuk menyimpan data log aktifitas User OPD.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_log | Variable untuk menyimpan id log | INTEGER | NO | Primary key |
| log\_act | Variable untuk menyimpan aktifitas yang dilakukan oleh User OPD | VARCHAR(12) |  |  |
| log\_query | Variable untuk menyimpan detail aktifitas User OPD dalam query sql | TEXT |  |  |
| log\_user\_id | Variable untuk menyimpan log user id | VARCHAR(32) |  |  |
| log\_username | Variable untuk menyimpan username pengguna yang melakukan aktifitas | VARCHAR(64) |  |  |
| created\_at | Variable untuk menyimpan tanggal aktifitas dilakukan | TIMESTAMP |  |  |

**Tabel 6 Tabel Log**

* + - 1. **Tabel Tipe Laporan**

Tabel untuk menyimpan tipe laporan yang ada dalam sistem.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | Primary key |
| nama\_laporan | Variable untuk menyimpan nama laporan | VARCHAR(64) |  |  |
| kode\_tipe | Variable untuk menyimpan kode tipe laporan | VARCHAR(64) |  |  |

**Tabel 7 Tabel Tipe Laporan**

* + - 1. **Tabel Tipe Laporan Per OPD**

Tabel untuk menyimpan tipe laporan yang bisa diakses oleh OPD.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | Foreign key dari tabel OPD |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | Foreign key dari tabel Tipe Laporan |

**Tabel 8 Tabel Tipe Laporan OPD**

* + - 1. **Tabel Laporan**

Tabel untuk menyimpan data laporan yang dibuat.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | Primary key |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | Foreign key dari tabel OPD |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | Foreign key dari tabel Tipe Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  |  |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  |  |

**Tabel 9 Tabel Laporan**

* + - 1. **Tabel Rekap Tender**

Tabel untuk menyimpan data laporan rekap tender.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| judul\_rekap\_tender | Variable untuk menyimpan judul laporan rekap tender | VARCHAR(128) |  |  |
| tgl | Variable untuk menyimpan tanggal laporan | DATE |  |  |

**Tabel 10 Tabel Rekap Tender**

* + - 1. **Tabel Detail Tekap Tender**

Tabel untuk menyimpan detail laporan rekap tender.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_detail\_rekap\_tender | Variable untuk menyimpan id detail rekap tender | INTEGER | NO | Primary key |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | Foreign key dari tabel OPD |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | Foreign key dari tabel Laporan |
| nama\_paket\_kerja | Variable untuk menyimpan nama paket pekerjaan yang ditenderkan | VARCHAR(128) |  |  |
| pagu | Variable untuk menyimpan anggaran paket kerja yang ditenderkan | INTEGER |  |  |
| nilai\_hps | Variable untuk menyimpan nilai hps dari paket kerja yang ditenderkan | INTEGER |  |  |
| pemenang | Variable untuk menyimpan pemenang tender | VARCHAR(256) |  |  |
| harga\_kontrak | Variable untuk menyimpan harga kontrak tender | INTEGER |  |  |
| ket | Variable untuk menyimpan keterangan hasil tender | TEXT |  |  |

**Tabel 11 Tabel Detail Rekap Tender**

* + - 1. **Tabel Rekap Pokja**

Tabel untuk menyimpan data laporan rekap kelompok kerja (pokja).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| tgl | Variable untuk menyimpan tanggal laporan | DATE |  |  |

**Tabel 12 Tabel Rekap Pokja**

* + - 1. **Tabel Detail Rekap Pokja**

Tabel untuk menyimpan detail dari laporan rekap pokja.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_pegawai | Variable untuk menyimpan id pegawai | INTEGER | NO | Foreign key dari tabel Pegawai |
| id\_detail\_rekap\_tender | Variable untuk menyimpan id detail rekap tender | INTEGER | NO | Foreign key dari tabel Detail Rekap Tender |
| Jabatan | Variable untuk menyimpan jabatan dalam kelompok kerja | VARCHAR(64) |  |  |
| Ket | Variable untuk menyimpan keterangan laporan rekap pokja | TEXT |  |  |

**Tabel 13 Tabel Detail Rekap Pokja**

* + - 1. **Tabel Pegawai**

Tabel untuk menyimpan nama pegawai yang bertugas dalam tender yang diatur dalam rekap pokja.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_pegawai | Variable untuk menyimpan id OPD | INTEGER | NO | Primary key |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | Foreign key dari tabel Laporan |
| nama | Variable untuk menyimpan nama pegawai | VARCHAR(64) |  |  |

**Tabel 14 Tabel Pegawai**

* + - 1. **Tabel Laporan Kinerja Triwulan**

Tabel untuk menyimpan data laporan kinerja triwulan.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| tgl | Variable untuk menyimpan tanggal laporan | DATE |  |  |

**Tabel 15 Tabel Laporan Kinerja Triwulan**

* + - 1. **Tabel Detail Laporan Kinerja Triwulan**

Tabel untuk menyimpan detail laporan kinerja triwulan.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | Foreign key dari tabel Laporan |
| uraian | Variable untuk menyimpan uraian program yang dilaporkan pada laporan kinerja triwulan | VARCHAR(128) |  |  |
| indikator\_kinerja | Variable untuk menyimpan indikator kinerja program yang dilaporkan pada laporan kinerja triwulan | VARCHAR(64) |  |  |
| target | Variable untuk menyimpan target program yang dilaporkan pada laporan kinerja triwulan | FLOAT |  |  |
| realisasi\_target | Variable untuk menyimpan realisasi target program yang dilaporkan pada laporan kinerja triwulan | INTEGER |  |  |
| program | Variable untuk menyimpan nama program yang dilaporkan pada laporan kinerja triwulan | VARCHAR(128) |  |  |
| anggaran | Variable untuk menyimpan anggaran program yang dilaporkan pada laporan kinerja triwulan | INTEGER |  |  |
| capaian\_realisasi\_keuangan | Variable untuk menyimpan realisasi anggaran program yang dilaporkan pada laporan kinerja triwulan | INTEGER |  |  |

**Tabel 16 Tabel Detail Laporan Kinerja Triwulan**

* + - 1. **Tabel Pemantauan Tindak Lanjut**

Tabel untuk menyimpan data laporan pemantauan tindak lanjut.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| tgl | Variable untuk menyimpan tanggal laporan | DATE |  |  |
| tgl\_status\_rekomendasi | Variable untuk menyimpan tanggal rekomendasi laporan pemantauan tindak lanjut | DATE |  |  |
| tgl\_tindak\_lanjut | Variable untuk menyimpan tanggal tindak lanjut laporan pemantauan tindak lanjut | DATE |  |  |

**Tabel 17 Tabel Pemantauan Tindak Lanjut**

* + - 1. **Tabel Temuan**

Tabel untuk menyimpan data temuan pada laporan pemantauan tindak lanjut.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_temuan | Variable untuk menyimpan id temuan | INTEGER | NO | Primary key |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | Foreign key dari tabel Laporan |
| nama\_temuan | Variable untuk menyimpan nama temuan laporan pemantauan tindak lanjut | VARCHAR(256) |  |  |

**Tabel 18 Tabel Temuan**

* + - 1. **Tabel Hasil Temuan**

Tabel untuk menyimpan detail kelanjutan dari temuan pada laporan pemantauan tindak lanjut.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_temuan | Variable untuk menyimpan id temuan | INTEGER | NO | Foreign key dari tabel Pegawai |
| rekomendasi | Variable untuk menyimpan rekomendasi temuan pada laporan pemantauan tindak lanjut | TEXT |  |  |
| status\_rekomendasi | Variable untuk menyimpan status rekomendasi temuan pada laporan pemantauan tindak lanjut | CHAR(2) |  |  |
| tindak\_lanjut | Variable untuk menyimpan tindak lanjut temuan pada laporan pemantauan tindak lanjut | TEXT |  |  |
| status\_tindak\_lanjut | Variable untuk menyimpan status tindak lanjut temuan pada laporan pemantauan tindak lanjut | CHAR(2) |  |  |
| catatan\_bpk | Variable untuk menyimpan catatan bpk temuan pada laporan pemantauan tindak lanjut | TEXT |  |  |

**Tabel 19 Tabel Hasil Temuan**

* + - 1. **Tabel Realisasi Fisik**

Tabel untuk menyimpan data laporan realisasi fisik.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| judul\_realisasi\_fisik | Variable untuk menyimpan judul laporan realisasi fisik | VARCHAR(128) |  |  |
| tgl | Variable untuk menyimpan tanggal laporan | DATE |  |  |

**Tabel 20 Tabel Realisasi Fisik**

* + - 1. **Tabel Program**

Tabel untuk menyimpan data program pada laporan realisasi fisik.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| kode\_program | Variable untuk menyimpan kode program pada laporan realisasi fisik | VARCHAR(30) | NO | Primary key |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | Foreign key dari tabel Laporan |
| nama\_program | Variable untuk menyimpan nama program pada laporan realisasi fisik | VARCHAR(128) |  |  |
| capaian\_indikator | Variable untuk menyimpan indikator program pada laporan realisasi fisik | VARCHAR(64) |  |  |
| capaian\_target\_ppas\_final | Variable untuk menyimpan target program pada laporan realisasi fisik | FLOAT |  |  |
| capaian\_anggaran\_keuangan | Variable untuk menyimpan anggaran program pada laporan realisasi fisik | INTEGER |  |  |
| capaian\_realisasi\_keuangan | Variable untuk menyimpan realisasi keuangan program pada laporan realisasi fisik | INTEGER |  |  |
| capaian\_realisasi\_keuangan\_persen | Variable untuk menyimpan realisasi keuangan program dalam persen pada laporan realisasi fisik | FLOAT |  |  |
| capaian\_realisasi\_kinerja | Variable untuk menyimpan realisasi kinerja program pada laporan realisasi fisik | FLOAT |  |  |
| capaian\_realisasi\_kinerja\_persen | Variable untuk menyimpan realisasi kinerja program dalam persen pada laporan realisasi fisik | FLOAT |  |  |
| capaian\_satuan | Variable untuk menyimpan satuan | VARCHAR(16) |  |  |
| ket | Variable untuk menyimpan keterangan program pada laporan realisasi fisik | TEXT |  |  |

**Tabel 21 Tabel Program**

* + - 1. **Tabel Kegiatan**

Tabel untuk menyimpan data kegiatan pada laporan realisasi fisik.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| kode\_kegiatan | Variable untuk menyimpan kode kegiatan pada laporan realisasi fisik | VARCHAR(30) | NO | Primary key |
| kode\_program | Variable untuk menyimpan kode program | VARCHAR(30) | NO | Foreign key dari tabel Program |
| nama\_kegiatan | Variable untuk menyimpan nama kegiatan pada laporan realisasi fisik | VARCHAR(128) |  |  |
| pagu\_ppas\_final | Variable untuk menyimpan anggaran kegiatan pada realisasi fisik | INTEGER |  |  |
| keluaran\_indikator | Variable untuk menyimpan indikator output kegiatan pada laporan realisasi fisik | VARCHAR(64) |  |  |
| keluaran\_target\_ppas\_final | Variable untuk menyimpan target output kegiatan pada laporan realisasi fisik | FLOAT |  |  |
| keluaran\_realisasi\_kinerja | Variable untuk menyimpan output realisasi kinerja kegiatan pada laporan realisasi fisik | FLOAT |  |  |
| keluaran\_realisasi\_kinerja\_persen | Variable untuk menyimpan output realisasi kinerja kegiatan dalam persen pada laporan realisasi fisik | FLOAT |  |  |
| keluaran\_satuan | Variable untuk menyimpan satuan output | VARCHAR(16) |  |  |
| hasil\_indikator | Variable untuk menyimpan indikator outcome kegiatan pada laporan realisasi fisik | VARCHAR(64) |  |  |
| hasil\_target\_ppas\_final | Variable untuk menyimpan target outcome kegiatan pada laporan realisasi fisik | FLOAT |  |  |
| hasil\_realisasi\_kinerja | Variable untuk menyimpan outcome realisasi kinerja kegiatan pada laporan realisasi fisik | FLOAT |  |  |
| hasil\_realisasi\_kinerja\_persen | Variable untuk menyimpan outcome realisasi kinerja kegiatan dalam persen pada laporan realisasi fisik | FLOAT |  |  |
| hasil\_satuan | Variable untuk menyimpan satuan outcome | VARCHAR(16) |  |  |
| realisasi\_keuangan | Variable untuk menyimpan realisasi keuangan kegiatan pada laporan realisasi fisik | INTEGER |  |  |
| realisasi\_keuangan\_persen | Variable untuk menyimpan realisasi keuangan kegiatan dalam persen pada laporan realisasi fisik | FLOAT |  |  |

**Tabel 22 Tabel Kegiatan**

* + - 1. **Tabel Jadwal Pelaksanaan**

Tabel untuk menyimpan jadwal rencana pelaksanaan program kerja pengawasan tahunan Kota Madiun.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| judul\_realisasi\_fisik | Variable untuk menyimpan judul laporan realisasi fisik | VARCHAR(128) |  |  |
| tgl | Variable untuk menyimpan tanggal laporan | DATE |  |  |

**Tabel 23 Tabel Jadwal Pelaksanaan**

* + - 1. **Tabel Jadwal Pelaksanaan OPD**

Tabel untuk menyimpan data jadwal pelaksanaan program kerja pengawasan tahunan Kota Madiun untuk tiap OPD.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_jadwal\_pelaksanaan\_opd | Variable untuk menyimpan id jadwal pelaksanaan tiap OPD | INTEGER | NO | Primary key |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | Foreign key dari tabel OPD |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | Foreign key dari tabel Laporan |
| jenis\_pengawasan | Variable untuk menyimpan jenis pengawasan | VARCHAR(64) |  |  |
| rmp | Variable untuk menyimpan rmp | VARCHAR(32) |  |  |
| rpl | Variable untuk menyimpan rpl | VARCHAR(32) |  |  |
| output\_lhp | Variable untuk menyimpan nilai output lhp | INTEGER |  |  |
| hari\_pengawasan | Variable untuk menyimpan nilai hari pengawasan | INTEGER |  |  |
| keterangan | Variable untuk menyimpan keterangan | TEXT |  |  |

**Tabel 24 Tabel Jadwal Pelaksanaan OPD**

* + - 1. **Tabel Auditor**

Tabel untuk menyimpan data auditor pada pelaksanaan program kerja pengawasan tahunan Kota Madiun.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_jadwal\_pelaksanaan\_opd | Variable untuk menyimpan id jadwal pelaksanaan OPD | INTEGER | NO | Foreign key dari tabel Jadwal Pelaksanaan OPD |
| nama\_auditor | Variable untuk menyimpan nama auditor | VARCHAR(64) |  |  |
| jabatan | Variable untuk menyimpan jabatan auditor | VARCHAR(64) |  |  |

**Tabel 25 Tabel Auditor**

* + - 1. **Tabel SOTK**

Tabel untuk menyimpan data laporan SOTK.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| tgl | Variable untuk menyimpan tanggal laporan | DATE |  |  |

**Tabel 26 Tabel SOTK**

* + - 1. **Tabel SOTK OPD**

Tabel untuk menyimpan data SOTK tiap OPD.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | Foreign key dari tabel OPD |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | Foreign key dari tabel Laporan |
| besaran | Variable untuk menyimpan besaran tiap OPD | VARCHAR(6) |  |  |

**Tabel 27 Tabel SOTK OPD**

* + - 1. **Tabel IKM**

Tabel untuk menyimpan data laporan nilai Indeks Kepuasan Masyarakat (IKM).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| tgl | Variable untuk menyimpan tanggal laporan | DATE |  |  |

**Tabel 28 Tabel IKM**

* + - 1. **Tabel IKM OPD**

Tabel untuk menyimpan data nilai laporan Indeks Kepuasan Masyarakat (IKM) tiap OPD.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | Foreign key dari tabel OPD |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | Foreign key dari tabel Laporan |
| nilai | Variable untuk menyimpan nilai IKM tiap OPD | FLOAT |  |  |
| predikat | Variable untuk menyimpan predikat tiap OPD | VARCHAR(16) |  |  |

**Tabel 29 Tabel IKM OPD**

* + - 1. **Tabel Pelayanan Publik**

Tabel untuk menyimpan data laporan indeks pelayanan publik.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| tgl | Variable untuk menyimpan tanggal laporan | DATE |  |  |

**Tabel 30 Tabel Pelayanan Publik**

* + - 1. **Tabel Pelayanan Publik OPD**

Tabel untuk menyimpan data nilai indeks pelayanan publik tiap OPD.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | Foreign key dari tabel OPD |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | Foreign key dari tabel Laporan |
| indeks\_pelayanan\_publik | Variable untuk menyimpan nilai indeks pelayanan publik tiap OPD | FLOAT |  |  |
| konversi\_100 | Variable untuk menyimpan nilai indeks pelayanan publik di konversi ke dalam 100 | FLOAT |  |  |
| Ket | Variable untuk menyimpan keterangan | TEXT |  |  |

**Tabel 31 Tabel Pelayanan Publik OPD**

* + - 1. **Tabel Tatalaksana**

Tabel untuk menyimpan data laporan hasil pengamatan tatalaksana.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| tgl | Variable untuk menyimpan tanggal laporan | DATE |  |  |

**Tabel 32 Tabel Tatalaksana**

* + - 1. **Tabel Tatalaksana OPD**

Tabel untuk menyimpan data nilai hasil pengamatan tatalaksana tiap OPD.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | Foreign key dari tabel OPD |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | Foreign key dari tabel Laporan |
| uji\_kompetensi | Variable untuk menyimpan nilai uji kompetensi tiap OPD | FLOAT |  |  |
| sop | Variable untuk menyimpan nilai sop tiap OPD | FLOAT |  |  |
| tata\_naskah\_dinas | Variable untuk menyimpan nilai tata naskah dinas tiap OPD | FLOAT |  |  |
| pakaian\_dinas | Variable untuk menyimpan nilai pakaian dinas tiap OPD | FLOAT |  |  |
| jam\_kerja | Variable untuk menyimpan nilai hari dan jam kerja tiap OPD | FLOAT |  |  |
| jml\_nilai | Variable untuk menyimpan jumlah nilai tiap OPD | FLOAT |  |  |
| Ket | Variable untuk menyimpan keterangan | TEXT |  |  |

**Tabel 33 Tabel Tatalaksana OPD**

* + - 1. **Tabel Monitoring Kelembagaan**

Tabel untuk menyimpan data laporan inventarisasi data permasalahan kelembagaan pada SKPD di lingkungan Pemerintah Kota Madiun.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| tgl | Variable untuk menyimpan tanggal laporan | DATE |  |  |

**Tabel 34 Tabel Monitoring Kelembagaan**

* + - 1. **Tabel Permasalahan Kelembagaan**

Tabel untuk menyimpan data permasalahan kelembagaan tiap OPD.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | Foreign key dari tabel OPD |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | Foreign key dari tabel Laporan |
| permasalahan\_kelembagaan | Variable untuk menyimpan permasalahan kelembagaan tiap OPD | VARCHAR(256) |  |  |
| usulan | Variable untuk menyimpan usulan dari permasalahan kelembagaan tiap OPD | TEXT |  |  |
| dasar\_hukum | Variable untuk menyimpan dasar hukum dari permasalahan kelembagaan tiap OPD | TEXT |  |  |
| ket | Variable untuk menyimpan keterangan | TEXT |  |  |

**Tabel 35 Tabel Permasalahan Kelembagaan**

* + - 1. **Tabel Laporan RB Area Perubahan**

Tabel untuk menyimpan data laporan rencana aksi reformasi birokrasi pemerintah daerah pada 8 area perubahan.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| tgl | Variable untuk menyimpan tanggal laporan | DATE |  |  |
| judul\_rb | Variable untuk menyimpan judul laporan RB | VARCHAR(256) |  |  |

**Tabel 36 Tabel Laporan RB Area Perubahan**

* + - 1. **Tabel RB Area Perubahan**

Tabel untuk menyimpan rincian aspek laporan rencana aksi reformasi birokrasi pemerintah daerah pada 8 area perubahan.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_rb\_area\_perubahan | Variable untuk menyimpan id rb area perubahan | INTEGER | NO | Primary key |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | Foreign key dari tabel Laporan |
| rincian | Variable untuk menyimpan rincian aspek laporan rencana aksi reformasi birokrasi pemerintah daerah pada 8 area perubahan | VARCHAR(128) |  |  |

**Tabel 37 Tabel RB Area Perubahan**

* + - 1. **Tabel RB Area Perubahan Program**

Tabel untuk menyimpan data program pada laporan rencana aksi reformasi birokrasi pemerintah daerah pada 8 area perubahan.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_rb\_area\_perubahan\_program | Variable untuk menyimpan id rb area perubahan program | INTEGER | NO | Primary key |
| id\_rb\_area\_perubahan | Variable untuk menyimpan id rb area perubahan | INTEGER | NO | Foreign key dari tabel RB Area Perubahan |
| nama\_program | Variable untuk menyimpan nama program pada laporan rencana aksi reformasi birokrasi pemerintah daerah area perubahan | VARCHAR(128) |  |  |

**Tabel 38 Tabel RB Area Perubahan Program**

* + - 1. **Tabel RB Area Perubahan Kegiatan**

Tabel untuk menyimpan data kegiatan pada laporan rencana aksi reformasi birokrasi pemerintah daerah pada 8 area perubahan.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | Foreign key dari tabel OPD |
| id\_rb\_area\_perubahan\_program | Variable untuk menyimpan id rb area perubahan program | INTEGER | NO | Foreign key dari tabel RB Area Perubahan Program |
| nama\_kegiatan | Variable untuk menyimpan nama kegiatan pada laporan rb area perubahan | VARCHAR(128) |  |  |
| target\_waktu | Variable untuk menyimpan target pelaksanaan kegiatan | VARCHAR(32) |  |  |
| realisasi\_waktu | Variable untuk menyimpan realisasi pelaksanaan kegiatan | VARCHAR(32) |  |  |
| target\_anggaran | Variable untuk menyimpan anggaran kegiatan | INTEGER |  |  |
| realisasi\_anggaran | Variable untuk menyimpan realisasi keuangan kegiatan | INTEGER |  |  |
| capaian | Variable untuk menyimpan capaian kegiatan | BOOLEAN |  |  |
| ket | Variable untuk menyimpan keterangan kegiatan | TEXT |  |  |

**Tabel 39 Tabel RB Area Perubahan Kegiatan**

* + - 1. **Tabel Laporan RB Prioritas**

Tabel untuk menyimpan data laporan rencana aksi reformasi birokrasi pemerintah daerah pada prioritas yang harus dipelihara.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| tgl | Variable untuk menyimpan tanggal laporan | DATE |  |  |
| judul\_rb | Variable untuk menyimpan judul laporan RB | VARCHAR(256) |  |  |

**Tabel 40 Tabel Laporan RB Prioritas**

* + - 1. **Tabel RB Prioritas**

Tabel untuk menyimpan data rincian aspek laporan rencana aksi reformasi birokrasi pemerintah daerah pada prioritas yang harus dipelihara.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_rb\_prioritas | Variable untuk menyimpan id rb prioritas | INTEGER | NO |  |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | Foreign key dari tabel Laporan |
| rincian | Variable untuk menyimpan rincian aspek laporan rencana aksi reformasi birokrasi pemerintah daerah pada prioritas yang harus dipelihara | VARCHAR(128) |  |  |

**Tabel 41 Tabel RB Prioritas**

* + - 1. **Tabel RB Prioritas Sasaran**

Tabel untuk menyimpan data sasaran pada laporan rencana aksi reformasi birokrasi pemerintah daerah pada prioritas yang harus dipelihara.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_rb\_prioritas\_sasaran | Variable untuk menyimpan id rb prioritas sasaran | INTEGER | NO | Primary key |
| id\_rb\_prioritas | Variable untuk menyimpan id rb prioritas | INTEGER | NO | Foreign key dari tabel RB Prioritas |
| sasaran | Variable untuk menyimpan sasaran pada laporan RB pada prioritas yang harus dipelihara | VARCHAR(256) |  |  |
| nama\_program | Variable untuk menyimpan nama program pada laporan RB pada prioritas yang harus dipelihara | VARCHAR(128) |  |  |

**Tabel 42 Tabel RB Prioritas Sasaran**

* + - 1. **Tabel RB Prioritas Kegiatan**

Tabel untuk menyimpan data kegiatan pada laporan rencana aksi reformasi birokrasi pada prioritas yang harus dipelihara.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_rb\_prioritas\_sasaran | Variable untuk menyimpan id rb prioritas sasaran | INTEGER | NO | Foreign key dari tabel RB Prioritas Sasaran |
| nama\_kegiatan | Variable untuk menyimpan nama kegiatan pada laporan rb prioritas | VARCHAR(128) |  |  |
| indikator | Variable untuk menyimpan indikator kegiatan pada laporan rb prioritas | VARCHAR(64) |  |  |
| target\_output | Variable untuk menyimpan target output kegiatan pada laporan rb prioritas | VARCHAR(64) |  |  |
| realisasi\_output | Variable untuk menyimpan realisasi output kegiatan pada laporan rb prioritas | VARCHAR(64) |  |  |
| target\_waktu | Variable untuk menyimpan target pelaksanaan kegiatan | VARCHAR(32) |  |  |
| realisasi\_waktu | Variable untuk menyimpan realisasi pelaksanaan kegiatan | VARCHAR(32) |  |  |
| target\_anggaran | Variable untuk menyimpan anggaran kegiatan | INTEGER |  |  |
| realisasi\_anggaran | Variable untuk menyimpan realisasi keuangan kegiatan | INTEGER |  |  |
| capaian | Variable untuk menyimpan capaian kegiatan | BOOLEAN |  |  |
| ket | Variable untuk menyimpan keterangan kegiatan | TEXT |  |  |

**Tabel 43 Tabel RB Prioritas Sasaran**

* + - 1. **Tabel Laporan RB Fokus**

Tabel untuk menyimpan data laporan rencana aksi reformasi birokrasi pemerintah daerah pada prioritas yang terkait dengan peningkatan kualitas pelayanan fokus pelayanan sector tertentu.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| tgl | Variable untuk menyimpan tanggal laporan | DATE |  |  |
| judul\_rb | Variable untuk menyimpan judul laporan RB | VARCHAR(256) |  |  |

**Tabel 44 Tabel Laporan RB Fokus**

* + - 1. **Tabel RB Fokus**

Tabel untuk menyimpan data rincian aspek laporan rencana aksi reformasi birokrasi pemerintah daerah pada pada prioritas yang terkait dengan peningkatan kualitas pelayanan fokus pelayanan sektor tertentu.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_rb\_fokus | Variable untuk menyimpan id rb focus | INTEGER | NO | Primary key |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | Foreign key dari tabel Laporan |
| rincian | Variable untuk menyimpan rincian aspek laporan rencana aksi reformasi birokrasi pemerintah daerah pada prioritas yang terkait dengan peningkatan kualitas pelayanan fokus pelayanan sektor tertentu | VARCHAR(128) |  |  |

**Tabel 45 Tabel RB Fokus**

* + - 1. **Tabel RB Fokus Sasaran**

Tabel untuk menyimpan data sasaran pada laporan rencana aksi reformasi birokrasi pemerintah daerah pada pada prioritas yang terkait dengan peningkatan kualitas pelayanan fokus pelayanan sektor tertentu.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_rb\_fokus\_sasaran | Variable untuk menyimpan id rb fokus sasaran | INTEGER | NO | Primary key |
| id\_rb\_fokus | Variable untuk menyimpan id rb fokus | INTEGER | NO | Foreign key dari tabel RB Fokus |
| sasaran | Variable untuk menyimpan sasaran pada laporan RB pada pada prioritas yang terkait dengan peningkatan kualitas pelayanan fokus pelayanan sektor tertentu | VARCHAR(256) |  |  |
| nama\_program | Variable untuk menyimpan nama program pada laporan RB pada prioritas yang terkait dengan peningkatan kualitas pelayanan fokus pelayanan sektor tertentu | VARCHAR(128) |  |  |

**Tabel 46 Tabel RB Fokus Sasaran**

* + - 1. **Tabel RB Fokus Kegiatan**

Tabel untuk menyimpan kegiatan pada laporan rencana aksi reformasi birokrasi pemerintah daerah pada pada prioritas yang terkait dengan peningkatan kualitas pelayanan fokus pelayanan sektor tertentu.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_rb\_fokus\_sasaran | Variable untuk menyimpan id rb fokus sasaran | INTEGER | NO | Foreign key dari tabel RB Fokus Sasaran |
| nama\_kegiatan | Variable untuk menyimpan nama kegiatan pada laporan rb fokus | VARCHAR(128) |  |  |
| indikator | Variable untuk menyimpan indikator kegiatan pada laporan rb fokus | VARCHAR(64) |  |  |
| target\_output | Variable untuk menyimpan target output kegiatan pada laporan rb fokus | VARCHAR(64) |  |  |
| realisasi\_output | Variable untuk menyimpan realisasi output kegiatan pada laporan rb fokus | VARCHAR(64) |  |  |
| target\_waktu | Variable untuk menyimpan target pelaksanaan kegiatan | VARCHAR(32) |  |  |
| realisasi\_waktu | Variable untuk menyimpan realisasi pelaksanaan kegiatan | VARCHAR(32) |  |  |
| target\_anggaran | Variable untuk menyimpan anggaran kegiatan | INTEGER |  |  |
| realisasi\_anggaran | Variable untuk menyimpan realisasi keuangan kegiatan | INTEGER |  |  |
| capaian | Variable untuk menyimpan capaian kegiatan | BOOLEAN |  |  |
| ket | Variable untuk menyimpan keterangan kegiatan | TEXT |  |  |

**Tabel 47 Tabel RB Fokus Kegiatan**

* + - 1. **Tabel Laporan ZI WBK**

Tabel untuk menyimpan data laporan rencana aksi birokrasi pemerintah daerah pada prioritas zona integritas menuju wilayah bebas korupsi.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| tgl | Variable untuk menyimpan tanggal laporan | DATE |  |  |
| judul\_rb | Variable untuk menyimpan judul laporan RB | VARCHAR(256) |  |  |

**Tabel 48 Tabel Laporan ZI WBK**

* + - 1. **Tabel ZI WBK Sasaran**

Tabel untuk menyimpan sasaran pada laporan rencana aksi birokrasi pemerintah daerah pada prioritas zona integritas menuju wilayah bebas korupsi.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_rb\_zi\_wbk\_sasaran | Variable untuk menyimpan id rb zi wbk sasaran | INTEGER | NO | Primary key |
| id\_rb\_zi\_wbk | Variable untuk menyimpan id rb zi wbk | INTEGER | NO | Foreign key dari tabel RB ZI WBK |
| sasaran | Variable untuk menyimpan sasaran pada laporan RB ZI WBK | VARCHAR(256) |  |  |
| nama\_program | Variable untuk menyimpan nama program pada laporan RB ZI WBK | VARCHAR(128) |  |  |

**Tabel 49 Tabel RB ZI WBK Sasaran**

* + - 1. **Tabel RB ZI WBK Kegiatan**

Tabel untuk menyimpan data kegiatan pada laporan rencana aksi birokrasi pemerintah daerah pada prioritas zona integritas menuju wilayah bebas korupsi.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_rb\_zi\_wbk\_sasaran | Variable untuk menyimpan id rb fokus sasaran | INTEGER | NO | Foreign key dari tabel RB ZI WBK Sasaran |
| nama\_kegiatan | Variable untuk menyimpan nama kegiatan pada laporan rb fokus | VARCHAR(128) |  |  |
| indikator | Variable untuk menyimpan indikator kegiatan pada laporan rb fokus | VARCHAR(64) |  |  |
| target\_output | Variable untuk menyimpan target output kegiatan pada laporan rb fokus | VARCHAR(64) |  |  |
| realisasi\_output | Variable untuk menyimpan realisasi output kegiatan pada laporan rb fokus | VARCHAR(64) |  |  |
| target\_waktu | Variable untuk menyimpan target pelaksanaan kegiatan | VARCHAR(32) |  |  |
| realisasi\_waktu | Variable untuk menyimpan realisasi pelaksanaan kegiatan | VARCHAR(32) |  |  |
| target\_anggaran | Variable untuk menyimpan anggaran kegiatan | INTEGER |  |  |
| realisasi\_anggaran | Variable untuk menyimpan realisasi keuangan kegiatan | INTEGER |  |  |
| capaian | Variable untuk menyimpan capaian kegiatan | BOOLEAN |  |  |
| ket | Variable untuk menyimpan keterangan kegiatan | TEXT |  |  |

**Tabel 50 Tabel RB ZI WBK Kegiatan**

* + - 1. **Tabel Laporan RB Quick Wins**

Tabel untuk menyimpan data laporan rencana aksi reformasi birokrasi pemerintah daerah pada prioritas yang terkait dengan peningkatan kualitas layanan fokus pelayanan quick wins.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_opd | Variable untuk menyimpan id OPD | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| id\_tipe | Variable untuk menyimpan id tipe laporan | INTEGER | NO | *Inherrit attribute* dari tabel Laporan |
| created\_at | Variable untuk menyimpan tanggal pembuatan laporan | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| updated\_at | Variable untuk menyimpan tanggal laporan di update | TIMESTAMP |  | *Inherrit attribute* dari tabel Laporan |
| tgl | Variable untuk menyimpan tanggal laporan | DATE |  |  |
| judul\_rb | Variable untuk menyimpan judul laporan RB | VARCHAR(256) |  |  |

**Tabel 51 Tabel Laporan RB Quick Wins**

* + - 1. **Tabel RB Quick Wins**

Tabel untuk menyimpan data rincian aspek laporan rencana aksi reformasi birokrasi pemerintah daerah pada prioritas yang terkait dengan peningkatan kualitas layanan fokus pelayanan quick wins.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_rb\_quick\_wins | Variable untuk menyimpan id rb quick wins | INTEGER | NO | Primary key |
| id\_laporan | Variable untuk menyimpan id laporan | INTEGER | NO | Foreign key dari tabel Laporan |
| Rincian | Variable untuk menyimpan rincian aspek laporan rencana aksi reformasi birokrasi pemerintah daerah pada prioritas yang terkait dengan peningkatan kualitas pelayanan fokus pelayanan quick wins | VARCHAR(128) |  |  |

**Tabel 52 Tabel RB Quick Wins**

* + - 1. **Tabel RB Quick Wins Sasaran**

Tabel untuk menyimpan data sasaran pada laporan rencana aksi reformasi birokrasi pemerintah daerah pada prioritas yang terkait dengan peningkatan kualitas layanan fokus pelayanan quick wins.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_rb\_quick\_wins\_sasaran | Variable untuk menyimpan id rb quick wins sasaran | INTEGER | NO | Primary key |
| id\_rb\_quick\_wins | Variable untuk menyimpan id rb quick wins | INTEGER | NO | Foreign key dari tabel RB Quick Wins |
| sasaran | Variable untuk menyimpan sasaran pada laporan RB Quick Wins | VARCHAR(256) |  |  |
| nama\_program | Variable untuk menyimpan nama program pada laporan RB Quick Wins | VARCHAR(128) |  |  |

**Tabel 53 Tabel RB Quick Wins Sasaran**

* + - 1. **Tabel RB Quick Wins Kegiatan**

Tabel untuk menyimpan data kegiatan pada laporan rencana aksi reformasi birokrasi pemerintah daerah pada prioritas yang terkait dengan peningkatan kualitas layanan fokus pelayanan quick wins.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID Field** | **Deskripsi** | **Tipe & Length** | **Boleh NULL** | **Keterangan** |
| id\_rb\_quick\_wins\_sasaran | Variable untuk menyimpan id RB Quick Wins | INTEGER | NO | Foreign key dari tabel RB Quick Wins Sasaran |
| nama\_kegiatan | Variable untuk menyimpan nama kegiatan pada laporan RB Quick Wins | VARCHAR(128) |  |  |
| indikator | Variable untuk menyimpan indikator kegiatan pada laporan RB Quick Wins | VARCHAR(64) |  |  |
| target\_output | Variable untuk menyimpan target output kegiatan pada laporan RB Quick Wins | VARCHAR(64) |  |  |
| realisasi\_output | Variable untuk menyimpan realisasi output kegiatan pada laporan RB Quick Wins | VARCHAR(64) |  |  |
| target\_waktu | Variable untuk menyimpan target pelaksanaan kegiatan | VARCHAR(32) |  |  |
| realisasi\_waktu | Variable untuk menyimpan realisasi pelaksanaan kegiatan | VARCHAR(32) |  |  |
| target\_anggaran | Variable untuk menyimpan anggaran kegiatan | INTEGER |  |  |
| realisasi\_anggaran | Variable untuk menyimpan realisasi keuangan kegiatan | INTEGER |  |  |
| capaian | Variable untuk menyimpan capaian kegiatan | BOOLEAN |  |  |
| ket | Variable untuk menyimpan keterangan kegiatan | TEXT |  |  |

**Tabel 54 Tabel RB Quick Wins Kegiatan**

*[Halaman ini sengaja dikosongkan]*

# BAB V IMPLEMENTASI SISTEM

Bab ini membahas tentang implementasi dari sistem *report management.* Sistem ini dibuat menggunakan metode MVC (Model View Controller).

Tautan implementasi sistem bisa diakses di alamat repositori berikut: github.com/rtejakusuma/elaporan

## Implementasi Model

* + 1. **Model Activitylog**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Activitylog\_model extends CI\_Model

{

public function save\_log($data)

{

$this->db->insert('log', $data);

return $this->db->affected\_rows();

}

}

**Kode Sumber 1 Model Activitylog**

* + 1. **Model Api\_ekin**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Api\_ekin\_model extends CI\_Model

{

public function get\_api()

{

$url = "https://ekinerja.madiunkota.go.id/sync/sync\_pegawai.php?user=sync\_ekin&pass=sync\_ekin\_hamdiramadhan";

$curl = curl\_init();

curl\_setopt\_array($curl, [

CURLOPT\_URL => $url,

CURLOPT\_RETURNTRANSFER => true,

CURLOPT\_ENCODING => "",

CURLOPT\_MAXREDIRS => 10,

CURLOPT\_TIMEOUT => 30,

**Kode Sumber 2 Model Api\_ekin**

CURLOPT\_HTTP\_VERSION => CURL\_HTTP\_VERSION\_1\_1,

CURLOPT\_CUSTOMREQUEST => "GET",

CURLOPT\_SSL\_VERIFYPEER => false,

CURLOPT\_HTTPHEADER => array(

"cache-control: no-cache",

"pass: sync\_ekin\_hamdiramadhan",

"postman-token: a35767f8-489f-2c32-e382-5b76496a0821",

"user: sync\_ekin"

),

]);

$response = curl\_exec($curl);

$err = curl\_error($curl);

if ($err) {

// API GAGAL

return NULL;

} else {

// API BERHASIL

$arr = json\_decode($response, true);

// $arr = $response;

return $arr;

}

}

public function get\_peg($kode\_ekin, $nip = NULL)

{

$data = $this->get\_api();

if ($nip) {

foreach ($data as $key => $row) {

if ($row['nip18'] == $nip && $row['kolok'] == $kode\_ekin && ($row['jabatan'] != '' || $row['jabatan'] != NULL) && $row['unit\_kerja']) {

return $row;

}

}

} else {

$response = array();

foreach ($data as $key => $row) {

if ($row['kolok'] == $kode\_ekin && ($row['jabatan'] != '' || $row['jabatan'] != NULL) && $row['unit\_kerja']) {

array\_push($response, $row);

}

}

return $response;

}

}

}

* + 1. **Model Api\_inaproc\_rup**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Api\_inaproc\_rup\_model extends CI\_Model

{

public function get\_api()

{

$url = "https://inaproc.lkpp.go.id/isb/api/9e80f42a-7850-403f-b619-e33105f1e1a7/json/17864547/pengumumanruptahunan2018/tipe/4:12/parameter/2019:D179";

$curl = curl\_init();

curl\_setopt\_array($curl, [

CURLOPT\_URL => $url,

CURLOPT\_RETURNTRANSFER => true,

CURLOPT\_ENCODING => "",

CURLOPT\_MAXREDIRS => 10,

CURLOPT\_TIMEOUT => 120,

CURLOPT\_HTTP\_VERSION => CURL\_HTTP\_VERSION\_1\_1,

CURLOPT\_CUSTOMREQUEST => "GET",

CURLOPT\_SSL\_VERIFYPEER => false,

CURLOPT\_HTTPHEADER => array(

"cache-control: no-cache"

),

]);

$response = curl\_exec($curl);

$err = curl\_error($curl);

if ($err) {

// API GAGAL

return NULL;

} else {

// API BERHASIL

$arr = json\_decode($response, true);

// $arr = $response;

return $arr;

}

}

public function get\_rup()

{

$data = $this->get\_api();

$this->load->model('opd\_model', 'opd');

$ret = array();

foreach($data as $rup){

$id\_opd = $this->opd->get\_idopd\_by\_namaopd($rup['nama\_satker']);

if($id\_opd != NULL){

array\_push($ret, array(

'id\_opd' => $id\_opd,

'nama\_opd' => $rup['nama\_satker'],

'nama\_paket\_kerja' => $rup['nama\_paket'],

'pagu' => $rup['pagu\_rup']

));

}

}

**Kode Sumber 3 Model Api\_inaproc\_rup**

return $ret;

}

}

* + 1. **Model Api\_sikd**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Api\_sikd\_model extends CI\_Model

{

public function \_\_construct()

{

parent::\_\_construct();

$this->load->helper('url');

$this->load->library('xmlrpc');

$this->load->library('xmlrpcs');

}

public function get\_api($id\_laporan, $tahun, $id\_opd)

{

$this->load->model('opd\_model', 'opd');

$kode\_skpd = $this->opd->get\_idemov($id\_opd);

if ($kode\_skpd == NULL) return NULL;

$fet = $this->get\_lra($tahun, $kode\_skpd);

printf("<pre>%s</pre>", json\_encode($fet, JSON\_PRETTY\_PRINT));

die();

$ret = array();

if ($fet != NULL) {

foreach ($fet as $f) {

array\_push($ret, array(

'kode\_kegiatan' => $id\_laporan . '-' . $f['KODE\_PROGRAM'] . '-' . $f['KODE\_KEGIATAN'],

'realisasi' => $f['REALISASI']

));

}

return $ret;

} else return NULL;

}

public function get\_serapan($tahun = '2019', $tw = '1')

{

$url = 'sikd.madiunkota.net/serapan/serapan' . $tahun . '/api/' . $tw;

$curl = curl\_init();

curl\_setopt\_array($curl, [

CURLOPT\_URL => $url,

CURLOPT\_RETURNTRANSFER => true,

CURLOPT\_ENCODING => '',

CURLOPT\_MAXREDIRS => 10,

CURLOPT\_TIMEOUT => 60,

CURLOPT\_HTTP\_VERSION => CURL\_HTTP\_VERSION\_1\_1,

CURLOPT\_CUSTOMREQUEST => "GET",

]);

$response = curl\_exec($curl);

$err = curl\_error($curl);

if ($err) {

// API GAGAL

return NULL;

} else {

// API BERHASIL

$arr = json\_decode($response, true);

return ['arr' => $arr, 'url' => $url];

}

}

public function get\_lra($tahun, $kode\_skpd)

{

$url = 'sikd.madiunkota.net/lra' . $tahun . '/lra/api/' . $kode\_skpd;

$curl = curl\_init();

curl\_setopt\_array($curl, [

CURLOPT\_URL => $url,

CURLOPT\_RETURNTRANSFER => true,

CURLOPT\_ENCODING => '',

CURLOPT\_MAXREDIRS => 10,

CURLOPT\_TIMEOUT => 180,

CURLOPT\_HTTP\_VERSION => CURL\_HTTP\_VERSION\_1\_1,

CURLOPT\_CUSTOMREQUEST => "GET",

]);

$response = curl\_exec($curl);

$err = curl\_error($curl);

if ($err) {

// API GAGAL

var\_dump($url, $err);

return NULL;

} else {

// API BERHASIL

$arr = json\_decode($response, true);

return ['arr' => $arr, 'url' => $url];

}

public function get\_tahunan($tahun = '2019')

{

//init

$rawdata = $this->get\_api($tahun, '1');

//init data

$data = array();

$idx = 0;

foreach ($rawdata['data'] as $row) {

array\_push($data, ['kode\_skpd' => $row['KODE\_SKPD'], 'pagu' => $row['Pagu 1 Tahun']]);

$data[$idx]['tw'] = array();

array\_push($data[$idx]['tw'], ['realisasi' => $row['realisasi sd Tw 1'], 'target' => $row['target sd Tw 1'], 'prosentase' => $row['prosentase']]);

$idx++;

}

// ini tw2/3/4

for ($i = 2; $i < 5; $i++) {

$rawdata = $this->get\_api($tahun, $i);

**Kode Sumber 4 Model Api\_sikd**

$idx = 0;

foreach ($rawdata['data'] as $row) {

array\_push($data[$idx]['tw'], ['realisasi' => $row['realisasi sd Tw ' . $i], 'target' => $row['target sd Tw ' . $i], 'prosentase' => $row['prosentase']]);

$idx++;

}

}

return $data;

}

}

* + 1. **Model Api\_sipp**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Api\_sipp\_model extends CI\_Model

{

public function \_\_construct()

{

parent::\_\_construct();

$this->load->helper('url');

$this->load->library('xmlrpc');

$this->load->library('xmlrpcs');

}

public function get\_api($key, $value)

{

$key = 'b64149c3ea867959207e933bb686c9ab41d3370b';

$url = "http://eplanning.madiunkota.go.id/api/ws/ppas/final?$key=$value";

$curl = curl\_init();

curl\_setopt\_array($curl, [

CURLOPT\_URL => $url,

CURLOPT\_RETURNTRANSFER => true,

CURLOPT\_ENCODING => '',

CURLOPT\_MAXREDIRS => 10,

CURLOPT\_TIMEOUT => 60,

CURLOPT\_HTTP\_VERSION => CURL\_HTTP\_VERSION\_1\_1,

CURLOPT\_CUSTOMREQUEST => "GET",

CURLOPT\_HTTPHEADER => [

"X-API-KEY: b64149c3ea867959207e933bb686c9ab41d3370b"

]

]);

$response = curl\_exec($curl);

$err = curl\_error($curl);

if ($err) {

// API GAGAL

return NULL;

} else {

// API BERHASIL

$arr = json\_decode($response, true);

return $arr;

}

}

public function api\_fetch\_data($id\_opd\_param, $laporan\_baru, $tahun)

{

if ($laporan\_baru == NULL) {

return NULL;

}

$this->load->model('opd\_model', 'opd');

$idebud = $this->opd->get\_idebud($id\_opd\_param);

$data = $this->get\_api("kode\_opd", $idebud);

if (!isset($data['data'])) {

return NULL;

}

$data = $data['data'];

$this->load->model('laporan\_model', 'lp');

$this->load->model('tipelaporan\_model', 'tl');

$fet = array();

foreach ($data as $row) {

if ($row['tahun'] != $tahun)

continue;

if (!isset($fet[$row['nama\_opd']]))

$fet[$row['nama\_opd']] = array();

array\_push($fet[$row['nama\_opd']], $row);

unset($data);

$data\_prog;

$data\_kg;

foreach ($fet as $dataperopd) {

unset($data\_prog, $data\_kg);

$data\_kg = [];

$data\_prog = [];

$id\_opd = $this->opd->get\_id\_from\_ebud($dataperopd[0]['kode\_opd']);

// unregistered/differ kode\_ebud/ekin/emonev

if ($id\_opd == NULL) {

return NULL;

}

// tabel program + kegiatan

foreach ($dataperopd as $d) {

if (!isset($data\_prog[$laporan\_baru['id\_laporan'] . "-" . $d['kode\_program']])) {

$data\_prog[$laporan\_baru['id\_laporan'] . "-" . $d['kode\_program']] =

array(

'id\_laporan' => $laporan\_baru['id\_laporan'],

'kode\_program' => $laporan\_baru['id\_laporan'] . "-" . $d['kode\_program'],

'nama\_program' => $d['nama\_program'],

'capaian\_indikator' => reset($d['capaian'])['indikator'],

'capaian\_target\_ppas\_final' => reset($d['capaian'])['target\_ppas\_final'],

'capaian\_satuan' => reset($d['capaian'])['satuan'],

);

}

if (!isset($data\_kg[$laporan\_baru['id\_laporan'] . "-" . $d['kode\_program'] . "-" . $d['kode\_kegiatan']])) {

$data\_kg[$laporan\_baru['id\_laporan'] . "-" . $d['kode\_program'] . "-" . $d['kode\_kegiatan']] =

array(

'kode\_kegiatan' => $laporan\_baru['id\_laporan'] . "-" . $d['kode\_program'] . "-" . $d['kode\_kegiatan'],

'kode\_program' => $laporan\_baru['id\_laporan'] . "-" . $d['kode\_program'],

'nama\_kegiatan' => $d['nama\_kegiatan'],

'pagu\_ppas\_final' => $d['pagu\_ppas\_final'],

'keluaran\_indikator' => reset($d['keluaran'])['indikator'],

'keluaran\_target\_ppas\_final' => reset($d['keluaran'])['target\_ppas\_final'],

'keluaran\_satuan' => reset($d['keluaran'])['satuan'],

'hasil\_indikator' => reset($d['hasil'])['indikator'],

'hasil\_target\_ppas\_final' => reset($d['hasil'])['target\_ppas\_final'],

'hasil\_satuan' => reset($d['hasil'])['satuan'],

);

}

}

return array('prog' => $data\_prog, 'kg' => $data\_kg);

}

}

public function api\_fetchall\_realisasifisik($id\_opd\_param = NULL, $tahun = "2020")

{

$this->load->model('opd\_model', 'opd');

$idebud = NULL;

if ($id\_opd\_param != NULL) {

$idebud = $this->opd->get\_idebud($id\_opd\_param);

}

$data = $this->get\_api("tahun", $tahun);

if (!isset($data['data'])) {

echo "TIMEOUT<br/>";

return;

}

$data = $data['data'];

$this->load->model('tipelaporan/realisasifisik\_model', 'rf');

$this->load->model('laporan\_model', 'lp');

$this->load->model('tipelaporan\_model', 'tl');

$fet = array();

foreach ($data as $row) {

if ($id\_opd == NULL) {

if (!isset($fet[$row['nama\_opd']]))

$fet[$row['nama\_opd']] = array();

array\_push($fet[$row['nama\_opd']], $row);

} else {

if ($row['kode\_opd'] != $idebud) continue;

$fet[$row['nama\_opd']] = array();

array\_push($fet[$row['nama\_opd']], $row);

}

}

unset($data);

$data\_lp;

$data\_rf;

$data\_prog;

$data\_kg;

// asumsi 1 opd 1 laporan per tahun => butuh id dari API kalo mau multi laporan

foreach ($fet as $dataperopd) {

unset($data\_lp, $data\_rf, $data\_prog, $data\_kg);

$data\_kg = [];

$data\_prog = [];

$id\_opd = $this->opd->get\_id\_from\_ebud($dataperopd[0]['kode\_opd']);

// unregistered/differ kode\_ebud/ekin/emonev

if ($id\_opd == NULL) {

$data = array([

'nama\_opd' => strtoupper($dataperopd[0]['nama\_opd']),

'kode\_ebud' => $dataperopd[0]['kode\_opd']

]);

$this->opd->insert($data);

continue;

}

$id\_tipe = $this->tl->get\_idtipe\_by\_kodetipe('realisasi\_fisik');

$data\_lp = array(

'id\_tipe' => $id\_tipe,

);

// add to laporan

$laporan\_baru = $this->lp->add\_data($data\_lp);

//add to realisasi\_fisik

$data\_rf = array(

'id\_laporan' => $laporan\_baru['id\_laporan'],

'id\_opd' => $laporan\_baru['id\_opd'],

'id\_tipe' => $laporan\_baru['id\_tipe'],

'created\_at' => $laporan\_baru['created\_at'],

'updated\_at' => $laporan\_baru['updated\_at'],

);

// tabel program + kegiatan

foreach ($dataperopd as $d) {

if (!isset($data\_prog[$laporan\_baru['id\_laporan'] . $d['kode\_program']])) {

$data\_prog[$laporan\_baru['id\_laporan'] . $d['kode\_program']] =

array(

'id\_laporan' => $laporan\_baru['id\_laporan'],

'kode\_program' => $laporan\_baru['id\_laporan'] . $d['kode\_program'],

'nama\_program' => $d['nama\_program'],

'capaian\_indikator' => reset($d['capaian'])['indikator'],

'capaian\_target' => reset($d['capaian'])['target'],

'capaian\_target\_rkpd' => reset($d['capaian'])['target\_rkpd'],

'capaian\_target\_ppas\_draft' => reset($d['capaian'])['target\_ppas\_draft'],

'capaian\_target\_ppas\_final' => reset($d['capaian'])['target\_ppas\_final'],

'capaian\_satuan' => reset($d['capaian'])['satuan'],

);

}

if (!isset($data\_kg[$laporan\_baru['id\_laporan'] . $d['kode\_program'] . $d['kode\_kegiatan']])) {

$data\_kg[$laporan\_baru['id\_laporan'] . $d['kode\_program'] . $d['kode\_kegiatan']] =

array(

'kode\_kegiatan' => $laporan\_baru['id\_laporan'] . $d['kode\_program'] . $d['kode\_kegiatan'],

'kode\_program' => $laporan\_baru['id\_laporan'] . $d['kode\_program'],

**Kode Sumber 5 Model Api\_sipp**

'nama\_kegiatan' => $d['nama\_kegiatan'],

'pagu\_renja' => $d['pagu\_renja'],

'pagu\_rkpd' => $d['pagu\_rkpd'],

'pagu\_ppas\_draft' => $d['pagu\_ppas\_draft'],

'pagu\_ppas\_final' => $d['pagu\_ppas\_final'],

'keluaran\_indikator' => reset($d['keluaran'])['indikator'],

'keluaran\_target' => reset($d['keluaran'])['target'],

'keluaran\_target\_rkpd' => reset($d['keluaran'])['target\_rkpd'],

'keluaran\_target\_ppas\_draft' => reset($d['keluaran'])['target\_ppas\_draft'],

'keluaran\_target\_ppas\_final' => reset($d['keluaran'])['target\_ppas\_final'],

'keluaran\_satuan' => reset($d['keluaran'])['satuan'],

'hasil\_indikator' => reset($d['hasil'])['indikator'],

'hasil\_target' => reset($d['hasil'])['target'],

'hasil\_target\_rkpd' => reset($d['hasil'])['target\_rkpd'],

'hasil\_target\_ppas\_draft' => reset($d['hasil'])['target\_ppas\_draft'],

'hasil\_target\_ppas\_final' => reset($d['hasil'])['target\_ppas\_final'],

'hasil\_satuan' => reset($d['hasil'])['satuan'],

);

}

}

$this->rf->insert\_fetch($data\_rf, $data\_prog, $data\_kg);

}

}

}

* + 1. **Model Dbbackup**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Dbbackup\_model extends CI\_Model

{

public function \_\_construct()

{

parent::\_\_construct();

$this->load->dbutil();

}

public function backup\_laporan($formname, $id\_laporan)

{

$prefs;

if ($formname == "realisasi\_fisik") {

$prefs = array(

'tables' => array('realisasi\_fisik', 'program', 'kegiatan'),

'ignore' => array(),

'format' => 'txt',

'add\_drop' => FALSE,

'add\_insert' => TRUE,

'foreign\_key\_checks' => FALSE

);

}

**Kode Sumber 6 Model Dbbackup**

$backup = $this->dbutil->backup($prefs);

$towrite = "SET foreign\_key\_checks = 0;";

foreach (explode(';', $backup) as $query) {

if (strpos($query, 'INSERT INTO') !== false) {

$towrite = $towrite . str\_replace('INSERT INTO', 'INSERT IGNORE INTO', $query) . ';';

}

}

$towrite = $towrite . "SET foreign\_key\_checks = 1;";

echo write\_file('./migration/' . $formname . '/' . $id\_laporan . '\_' . strval(date('Y-m-d\_H:i:s', time())) . '.sql', $towrite);

}

public function restore\_laporan($formname, $id\_laporan, $tm)

{

$query = file\_get\_contents(APPPATH . "../migration/$formname/$tm.sql");

$this->db->delete('realisasi\_fisik', "id\_laporan = $id\_laporan");

foreach (explode(';', $query) as $q) {

if ($q == '' || $q == NULL) continue;

echo $this->db->query($q);

}

}

}

* + 1. **Model Dbf**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Dbf\_model extends CI\_Model

{

// USER

public function get\_user($id = NULL)

{

$this->db->from('user');

$this->db->join('opd', 'opd.id\_opd = user.id\_opd', 'left');

$this->db->where('user.id != 1');

if ($id) {

$this->db->where('id', $id);

return $this->db->get()->row\_array();

}

return $this->db->get()->result\_array();

}

// OPD

public function get\_opd($id = NULL)

{

$this->db->from('opd');

$this->db->where('opd.id\_opd != 1');

if ($id) {

$this->db->where('opd.id\_opd = ', $id);

return $this->db->get()->row\_array();

}

return $this->db->get()->result\_array(); }

// TIPE LAPORAN

public function get\_tipe\_laporan($id\_tipe = NULL)

{

$this->db->from('tipe\_laporan');

if ($id\_tipe) {

$this->db->where('tipe\_laporan.id\_tipe = ', $id\_tipe);

return $this->db->get()->row\_array();

}

return $this->db->get()->result\_array();}

}

// TIPELAPORAN PER OPD

public function get\_opd\_tipe()

{

$data = $this->get\_opd();

$i = 0;

foreach ($data as $row) {

$tipe = $this->get\_tipelaporan\_per\_opd($row['id\_opd']);

$data[$i]['tipe\_laporan'] = array();

foreach ($tipe as $push) {

array\_push($data[$i]['tipe\_laporan'], $push);

}

$tipe = NULL;

$i++;

}

return $data;

}

public function get\_tipelaporan\_per\_opd($id = NULL)

{

$this->db->from('tipelaporan\_per\_opd');

$this->db->join('opd', 'tipelaporan\_per\_opd.id\_opd = opd.id\_opd', 'left');

$this->db->join('tipe\_laporan', 'tipe\_laporan.id\_tipe = tipelaporan\_per\_opd.id\_tipe', 'left');

if ($id) {

$this->db->where('tipelaporan\_per\_opd.id\_opd', $id);

}

return $this->db->get()->result\_array();

}

public function del\_tipelaporan\_per\_opd($id)

{

foreach ($id as $id\_tipe) {

$this->db->or\_where('tipelaporan\_per\_opd.id\_tipe !=', $id\_tipe);

}

$this->db->delete('tipelaporan\_per\_opd');

activity\_log();

}

public function get\_log($day = NULL)

{

$this->db->from('log');

$this->db->join('user', 'log.log\_user\_id = user.id', 'left');

$this->db->join('opd', 'opd.id\_opd = user.id\_opd', 'left');

$this->db->select('user.username, opd.nama\_opd, log.log\_act, log.created\_at, log.log\_query');

if ($day) {

**Kode Sumber 7 Model Dbf**

$this->db->where('DATE(log.created\_at)', $day);

}

return $this->db->get()->result\_array();

}

// LAPORAN

public function get\_laporan($id = NULL)

{

$this->db->from('laporan');

$this->db->join('opd', 'laporan.id\_opd = opd.id\_opd', 'left');

$this->db->join('tipe\_laporan', 'laporan.id\_tipe = tipe\_laporan.id\_tipe', 'left');

if ($id) {

$this->db->where('laporan.id\_laporan = ', $id);

return $this->db->get()->row\_array();

}

return $this->db->get()->result\_array();

}

}

* + 1. **Model Laporan**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Laporan\_model extends CI\_Model

{

public function gets()

{

return $this->db->select('laporan')->result\_array();

}

public function get\_alllaporan($page\_number, $id\_opd = NULL, $limit = 20)

{

if ($id\_opd == NULL || $id\_opd == '1') {

$this->db->select('id\_laporan, nama\_opd, nama\_laporan, created\_at')

->from('laporan')

->join('tipe\_laporan', 'laporan.id\_tipe = tipe\_laporan.id\_tipe')

->join('opd', 'laporan.id\_opd = opd.id\_opd')

->order\_by('created\_at', 'desc')

->limit($limit, ($page\_number - 1) \* $limit);

} else {

$this->db->select('id\_laporan, nama\_opd, nama\_laporan, created\_at')

->from('laporan')

->where('laporan.id\_opd = ' . $id\_opd)

->join('tipe\_laporan', 'laporan.id\_tipe = tipe\_laporan.id\_tipe')

->join('opd', 'laporan.id\_opd = opd.id\_opd')

->order\_by('created\_at', 'desc')

->limit($limit, ($page\_number - 1) \* $limit);

}

return $this->db->get()->result\_array();

}

public function search($cond)

{

$this->db->from('laporan');

foreach ($cond as $key => $value) {

if ($key != "start\_date" && $key != "end\_date" && $value != NULL)

$this->db->where("laporan.$key = $value");

}

// check date

if (strtotime($cond['start\_date']) > strtotime($cond['end\_date'])) {

$tmp = $cond['start\_date'];

$cond['start\_date'] = $cond['end\_date'];

$cond['end\_date'] = $tmp;

}

$this->db->where("created\_at >= ", $cond['start\_date'] . ' 00:00:00')

->where("created\_at <= ", $cond['end\_date'] . ' 23:59:59');

$this->db->join('tipe\_laporan', 'tipe\_laporan.id\_tipe = laporan.id\_tipe');

return $this->db->get()->result\_array();

}

public function get\_laporan\_by\_kodetipe($kode\_tipe, $page\_number = 10, $id\_opd, $limit = 20, $from = NULL, $to = NULL)

{

$this->db->select("laporan.id\_laporan,$kode\_tipe.\*")

->from('laporan')

->join($kode\_tipe, "laporan.id\_laporan = $kode\_tipe.id\_laporan")

->where("laporan.id\_opd", $id\_opd)

->limit($limit);

if ($from && $to) {

$this->db->where('DATE(laporan.created\_at) >=', $from);

$this->db->where('DATE(laporan.created\_at) <=', $to);

}

$res = $this->db->get();

return $res->result\_array();

}

public function get\_idtipe\_by\_idlaporan($id)

{

$ret = $this->db->get\_where('laporan', array('id\_laporan' => $id))->result();

if ($ret != NULL)

return $ret[0]->id\_tipe;

return NULL;

}

public function get\_laporan($id) // get laporan berdasarkan ID laporan

{

$table = $this->get\_tipe\_laporan($id);

$temp = $this->db->get($table)->row\_array();

if ($temp == NULL) {

return -1; // error code

}

return $temp;

}

public function get\_by\_id($id)

{

return $this->db->get\_where('laporan', array('id\_laporan' => $id))->result()[0]->id\_tipe;

}

public function get\_tipe\_laporan($id) {

**Kode Sumber 8 Model Laporan**

$res = $this->get\_by\_id($id);

return $this->\_get\_nama\_laporan($res);

}

public function get\_fielddata($name)

{

return $this->db->field\_data($name);

}

public function get\_laporan\_data($id)

{

$tipe = $this->get\_tipe\_laporan($id);

return $this->db->get\_where($tipe, array('id\_laporan' => $id))->result[0];

}

public function get\_laporan\_data\_by\_name\_id($formname, $id)

{

$tipe = $this->load->model("tipelaporan/" . str\_replace('\_', '', strtolower($formname)) . "\_model", 'tl');

return $this->tl->get\_data\_by\_id($id);

}

private function \_get\_nama\_laporan($id)

{

$temp = $this->db->get\_where('tipe\_laporan', array('id\_tipe' => $id))->result();

return $temp[0]->kode\_tipe;

}

public function add\_data($data)

{

$this->db->insert(

'laporan',

[

'id\_opd' => $data['id\_opd'],

'id\_tipe' => $data['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

$this->db->order\_by('updated\_at', 'DESC');

return $this->db->get\_where('laporan', ['id\_opd' => $data['id\_opd'], 'id\_tipe' => $data['id\_tipe'],])->result\_array()[0];

}

public function update\_data($id\_laporan, $data)

{

$namalaporan = $this->\_get\_nama\_laporan($id\_laporan);

$this->db->update($namalaporan, $data, array('id\_laporan' => $id\_laporan));

activity\_log();

}

}

* + 1. **Model Opd**

<?php

use phpDocumentor\Reflection\Types\Null\_;

defined('BASEPATH') or exit('No direct script access allowed');

class Opd\_model extends CI\_Model

{

public function gets()

{

$this->db->from('opd')->order\_by('nama\_opd', 'ASC');

return $this->db->get()->result\_array();

}

public function insert($data)

{

$this->db->insert\_batch('opd', $data);

activity\_log();

}

public function get\_namaopd($id)

{

$ret = $this->db->get\_where('opd', array('id\_opd' => $id))->result();

if ($ret != NULL)

return $ret[0]->nama\_opd;

else return NULL;

}

public function get\_idopd\_by\_namaopd($nama\_opd)

{

$ret = $this->db->select('id\_opd')->from('opd')

->where('nama\_opd', ucwords($nama\_opd))->get()->result\_array();

if ($ret != NULL) return $ret[0]['id\_opd'];

else return NULL;

}

public function get\_idebud($id\_opd = NULL, $nama\_opd = NULL)

{

if ($id\_opd) {

$ret = $this->db->get\_where('opd', ['id\_opd' => $id\_opd])->result\_array();

if ($ret) return $ret[0]['kode\_ebud'];

else return NULL;

} elseif ($nama\_opd) {

$ret = $this->db->get\_where('opd', ['nama\_opd' => strtoupper($nama\_opd)])->result\_array();

if ($ret) return $ret[0]['id\_ebud'];

else return NULL;

} else {

$this->db->select('id\_ebud')->from('opd');

return $this->db->get()->result\_array();

}

}

public function get\_idemov($id\_opd = NULL, $nama\_opd = NULL)

{

if ($id\_opd) {

$ret = $this->db->get\_where('opd', ['id\_opd' => $id\_opd])->result\_array();

if ($ret) return $ret[0]['kode\_emov'];

else return NULL;

Kode Sumber 9 Model Opd

} elseif ($nama\_opd) {

$ret = $this->db->get\_where('opd', ['nama\_opd' => strtoupper($nama\_opd)])->result\_array();

if ($ret) return $ret[0]['id\_emov'];

else return NULL;

} else {

$this->db->select('id\_ebud')->from('opd');

return $this->db->get()->result\_array();

}

}

public function get\_idopd\_by\_name($name)

{

$ret = $this->db->get\_where('opd', array('nama\_opd' => $name))->result();

if ($ret != NULL)

return $ret[0]->nama\_opd;

else return NULL;

}

public function get\_id\_from\_ebud($kode\_ebud)

{

if ($kode\_ebud == NULL) {

return NULL;

}

$ret = $this->db->get\_where('opd', ['kode\_ebud' => $kode\_ebud])->result\_array();

if ($ret != NULL) {

return $ret[0]['id\_opd'];

}

return NULL;

}

public function get\_ekin\_by\_id($id\_opd)

{

$this->db->select('opd.kode\_ekin');

$this->db->from('opd');

$this->db->where('opd.id\_opd', $id\_opd);

$row = $this->db->get()->row\_array();

if ($row['kode\_ekin'] != '' && $row['kode\_ekin'] != NULL) {

return $row['kode\_ekin'];

}

return NULL;

}

}

* + 1. **Model Tipelaporan**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Tipelaporan\_model extends CI\_Model

{

public function gets()

{

return $this->db->order\_by('nama\_laporan','ASC')->get('tipe\_laporan')->result\_array();

}

**Kode Sumber 10 Model Tipelaporan**

public function get\_namalaporan($id\_tipe)

{

if ($id\_tipe == NULL)

return NULL;

$ret = $this->db->get\_where('tipe\_laporan', array('id\_tipe' => $id\_tipe))->result\_array();

return $ret;

}

public function get\_idtipe\_by\_kodetipe($namalaporan)

{

$ret = $this->db->get\_where('tipe\_laporan', array('kode\_tipe' => strtolower($namalaporan)))->result();

if($ret != NULL)

return $ret[0]->id\_tipe;

return NULL;

}

public function get\_idtipe\_by\_idlaporan($id\_laporan)

{

$ret = $this->db->select('id\_tipe')

->from('tipe\_laporan')

->join('laporan', 'laporan.id\_tipe = tipe\_laporan.id\_tipe')

->where('id\_laporan', $id\_laporan)->get()->result\_array();

if($ret != NULL)

return $ret['id\_laporan'][0];

else return NULL;

} }

}

* + 1. **Model Tipelaporanopd**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Tipelaporanopd\_model extends CI\_Model

{

public function gets()

{

return $this->db->select('tipelaporan\_per\_opd')->result\_array();

}

public function get\_opd\_namatipelaporan($id\_opd = NULL)

{

if ($id\_opd != NULL) {

$this->db->select('id\_opd, tipe\_laporan.\*')

->from('tipelaporan\_per\_opd')

->where('id\_opd', $id\_opd)

->join('tipe\_laporan', 'tipelaporan\_per\_opd.id\_tipe = tipe\_laporan.id\_tipe');

} else {

$this->db->select('id\_opd, tipe\_laporan.\*')

->from('tipelaporan\_per\_opd')

->join('tipe\_laporan', 'tipelaporan\_per\_opd.id\_tipe = tipe\_laporan.id\_tipe');

}

return $this->db->get()->result\_array();

}

public function get\_tipelaporan\_by\_idopd($id\_opd)

{

$this->db->select('tipe\_laporan.id\_tipe, tipe\_laporan.nama\_laporan, tipe\_laporan.kode\_tipe')

->from('tipelaporan\_per\_opd')

->where('tipelaporan\_per\_opd.id\_opd = ' . $id\_opd)

->join('tipe\_laporan', 'tipelaporan\_per\_opd.id\_tipe = tipe\_laporan.id\_tipe')

->order\_by('tipe\_laporan.nama\_laporan', 'asc');

return $this->db->get()->result\_array();

}

public function update\_tipelaporan\_per\_opd($id\_opd, $data)

{

$this->db->trans\_start();

// delete saved data first

$this->db->delete('tipelaporan\_per\_opd', array('id\_opd' => $id\_opd));

activity\_log();

$to\_insert = array();

if ($data == NULL || sizeof($data) <= 0) {

$this->db->trans\_complete();

return;

}

foreach ($data as $value) {

if ($value == NULL) continue;

array\_push($to\_insert, ['id\_opd' => $id\_opd, 'id\_tipe' => $value]); }

$this->db->insert\_batch('tipelaporan\_per\_opd', $to\_insert);

activity\_log();

$this->db->trans\_complete();

}

}

**Kode Sumber 11 Model Tipelaporanopd**

* + 1. **Model User**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class User\_model extends CI\_Model

{

public function gets()

{

$this->db->from('user')->order\_by('username', 'ASC');

return $this->db->get()->result\_array();

}

public function get\_login($username)

{

return $this->db->get\_where('user', ['username' => $username])->result\_array()[0];

}

public function insert($data)

{

**Kode Sumber 12 Model User**

$tmp = new DateTime();

$data['created\_at'] = $tmp->format('Y-m-d H:i:s');

$data['last\_login'] = $tmp->format('Y-m-d H:i:s');

$this->db->insert('user', $data);

activity\_log();

}

public function last\_login($id)

{

$this->db->update('user', ['last\_login' => date('Y-m-d H:i:s', now())], ['id' => $id]);

activity\_log();

}

public function reset\_password($id, $newpassword)

{

$this->db->update('user', ['password' => password\_hash($newpassword, PASSWORD\_BCRYPT)], ['id' => $id]);

activity\_log();

}

public function delete($id)

{

$this->db->delete('user', ['id' => $id]);

activity\_log();

} }

}

* + 1. **Model Ikm**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Ikm\_model extends CI\_Model

{

public function get\_data($id\_laporan = NULL, $id\_opd = NULL)

{

$ret = NULL;

$temp = NULL;

$this->db->select('ikm.\*, opd.id\_opd, opd.nama\_opd')->from('ikm');

if ($id\_opd != NULL) { // per opd

$this->db->where('ikm.id\_opd', $id\_opd);

}

if ($id\_laporan != NULL) { // laporan spesifik

$this->db->where('ikm.id\_laporan', $id\_laporan);

}

$this->db->join('ikm\_opd', 'ikm.id\_laporan = ikm\_opd.id\_laporan')

->join('opd', 'opd.id\_opd = ikm\_opd.id\_opd');

return $this->db->get()->result\_array();

}

public function get\_data\_by\_id($id)

{

$ikmdata = $this->db->get\_where('ikm', ['id\_laporan' => $id])->result\_array()[0];

$ikmopddata = $this->db->select('ikm\_opd.\*, opd.nama\_opd')

->from('ikm\_opd')

->join('opd', 'opd.id\_opd = ikm\_opd.id\_opd')

->where('ikm\_opd.id\_laporan', $id)

->order\_by('nama\_opd')

->get()->result\_array();

return array('ikm' => $ikmdata, 'ikmopd' => $ikmopddata);

}

public function init\_insert($id\_opd, $datalaporan, $data)

{

$this->db->trans\_start();

$this->load->model('laporan\_model', 'lp');

$this->db->insert(

'laporan',

[

'id\_opd' => $datalaporan['id\_opd'],

'id\_tipe' => $datalaporan['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

$this->db->order\_by('updated\_at', 'DESC');

$datalaporan = $this->db->get\_where('laporan', ['id\_opd' => $datalaporan['id\_opd'], 'id\_tipe' => $datalaporan['id\_tipe'],])->result\_array()[0];

$datalaporan['tgl'] = $data['tgl'];

$this->db->insert('ikm', $datalaporan);

activity\_log();

// insert second etc. table data here

$this->db->trans\_complete();

if ($this->db->trans\_status() === FALSE) {

$this->db->trans\_rollback();

return NULL;

}

return $datalaporan['id\_laporan'];

}

public function update\_data($id\_laporan, $data)

{

$table = $data['nama\_tabel'];

unset($data['nama\_tabel']);

$insdata = array();

if ($data != NULL) {

for ($i = 0; $i < sizeof(reset($data)); $i += 1) {

array\_push($insdata, array(

'id\_laporan' => $id\_laporan,

'id\_opd' => $data['id\_opd'][$i],

'predikat' => $data['predikat'][$i],

'nilai' => $data['nilai'][$i]

));

}

}

$this->db->trans\_begin();

**Kode Sumber 13 Model Ikm**

if ($table == 'ikm\_opd') {

$this->db->delete('ikm\_opd', "id\_laporan = $id\_laporan");

activity\_log();

if ($data != NULL) {

$this->db->insert\_batch('ikm\_opd', $insdata);

activity\_log();

}

}

$this->db->trans\_complete();

}

public function delete\_data($id\_laporan)

{

$this->db->trans\_begin();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('ikm');

activity\_log();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan');

activity\_log();

$this->db->trans\_complete();

}

}

* + 1. **Model Jadwalpelaksanaan**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Jadwalpelaksanaan\_model extends CI\_Model

{

public function get\_data($id\_laporan = NULL, $id\_opd = NULL)

{

$this->db->select("\*")->from('jadwal\_pelaksanaan');

if ($id\_opd != NULL) {

$this->db->where('jadwal\_pelaksanaan.id\_opd', $id\_opd);

}

if ($id\_laporan != NULL) {

$this->db->where('jadwal\_pelaksanaan.id\_laporan', $id\_laporan);

}

$this->db->join('jadwal\_pelaksanaan\_opd', 'jadwal\_pelaksanaan.id\_laporan = jadwal\_pelaksanaan\_opd.id\_laporan')

->join('opd', 'opd.id\_opd = jadwal\_pelaksanaan\_opd.id\_opd')

->join('auditor', 'jadwal\_pelaksanaan\_opd.id\_jadwal\_pelaksanaan\_opd = auditor.id\_jadwal\_pelaksanaan\_opd');

return $this->db->get()->result();

}

public function get\_data\_by\_id($id)

{

$jpdata = $this->db->get\_where('jadwal\_pelaksanaan', ['id\_laporan' => $id])->result\_array()[0];

$jpopddata = $this->db->select('jadwal\_pelaksanaan\_opd.\*, opd.nama\_opd')

->from('jadwal\_pelaksanaan\_opd')

->join('opd', 'jadwal\_pelaksanaan\_opd.id\_opd = opd.id\_opd')

->order\_by('nama\_opd', 'ASC')

->where('id\_laporan', $id)->get()->result\_array();

$adata = array();

if ($jpopddata != NULL) {

foreach ($jpopddata as $d) {

$adata[$d['id\_jadwal\_pelaksanaan\_opd']] =

$this->db->select('auditor.\*, opd.nama\_opd')->from('auditor')

->join('jadwal\_pelaksanaan\_opd', 'auditor.id\_jadwal\_pelaksanaan\_opd = jadwal\_pelaksanaan\_opd.id\_jadwal\_pelaksanaan\_opd')

->join('opd', 'jadwal\_pelaksanaan\_opd.id\_opd = opd.id\_opd')

->where('jadwal\_pelaksanaan\_opd.id\_laporan', $id)

->where("auditor.id\_jadwal\_pelaksanaan\_opd", $d['id\_jadwal\_pelaksanaan\_opd'])

->order\_by('nama\_opd', 'ASC')->get()->result\_array();

}

}

// GET AUDITOR DARI EKIN

$this->load->model('api\_ekin\_model');

$this->load->model('opd\_model');

$id\_opd = $this->session->tempdata('id\_opd');

$kode\_ekin = $this->opd\_model->get\_ekin\_by\_id($id\_opd);

$auditor = $this->api\_ekin\_model->get\_peg($kode\_ekin);

$auditor['kode\_ekin'] = $kode\_ekin;

return array('jp' => $jpdata, 'jpopd' => $jpopddata, 'adata' => $adata, 'auditor' => $auditor);

}

public function init\_insert($id\_opd, $datalaporan, $data)

{

$this->db->trans\_start();

$this->load->model('laporan\_model', 'lp');

$this->db->insert(

'laporan',

[

'id\_opd' => $datalaporan['id\_opd'],

'id\_tipe' => $datalaporan['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

$this->db->order\_by('updated\_at', 'DESC');

$datalaporan = $this->db->get\_where('laporan', ['id\_opd' => $datalaporan['id\_opd'], 'id\_tipe' => $datalaporan['id\_tipe'],])->result\_array()[0];

$datalaporan['tgl'] = $data['tgl'];

$this->db->insert('jadwal\_pelaksanaan', $datalaporan);

activity\_log();

// insert second etc. table data here

$this->db->trans\_complete();

if ($this->db->trans\_status() === FALSE) {

$this->db->trans\_rollback();

return NULL;

}

return $datalaporan['id\_laporan'];

}

public function update\_data($id\_laporan, $data)

{

$table = $data['nama\_tabel'];

unset($data['nama\_tabel']);

$insdata = array();

$updata = array();

$this->db->trans\_begin();

if ($table == 'jadwal\_pelaksanaan\_opd') {

if ($data != NULL) {

// new data

if (isset($data['new'])) {

for ($i = 0; $i < sizeof(reset($data['new'])); $i += 1) {

array\_push($insdata, array(

'id\_laporan' => $id\_laporan,

'id\_opd' => $data['new']['id\_opd'][$i],

'jenis\_pengawasan' => $data['new']['jenis\_pengawasan'][$i],

'rmp' => $data['new']['rmp'][$i],

'rpl' => $data['new']['rpl'][$i],

'output\_lhp' => $data['new']['output\_lhp'][$i],

'hari\_pengawasan' => $data['new']['hari\_pengawasan'][$i],

'keterangan' => $data['new']['keterangan'][$i]

));

}

}

if ($insdata != NULL) {

$this->db->insert\_batch('jadwal\_pelaksanaan\_opd', $insdata);

activity\_log();

}

unset($data['new']);

// updated data

if (isset($data['id\_jadwal\_pelaksanaan\_opd'])) {

for ($i = 0; $i < sizeof($data['id\_jadwal\_pelaksanaan\_opd']); $i += 1) {

array\_push($updata, array(

'id\_laporan' => $id\_laporan,

'id\_jadwal\_pelaksanaan\_opd' => $data['id\_jadwal\_pelaksanaan\_opd'][$i],

'id\_opd' => $data['id\_opd'][$i],

'jenis\_pengawasan' => $data['jenis\_pengawasan'][$i],

'rmp' => $data['rmp'][$i],

'rpl' => $data['rpl'][$i],

'output\_lhp' => $data['output\_lhp'][$i],

'hari\_pengawasan' => $data['hari\_pengawasan'][$i],

'keterangan' => $data['keterangan'][$i],

));

}

$this->db->update\_batch('jadwal\_pelaksanaan\_opd', $updata, 'id\_jadwal\_pelaksanaan\_opd');

activity\_log();

}

// unused data

if (isset($data['to\_del']))

$this->db->where\_in('id\_jadwal\_pelaksanaan\_opd', $data['to\_del'])

->delete('jadwal\_pelaksanaan\_opd');

activity\_log();

} else {

$this->db->delete('jadwal\_pelaksanaan\_opd', "id\_laporan = $id\_laporan");

activity\_log();

}

} else if ($table == 'auditor') {

$tmp = $data['id\_jadwal\_pelaksanaan\_opd'];

unset($data['id\_jadwal\_pelaksanaan\_opd']);

if ($data != NULL) {

foreach ($tmp as $idx) {

if (isset($data['nama\_auditor'][$idx])) {

for ($i = 0; $i < sizeof($data['nama\_auditor'][$idx]); $i += 1) {

if ($data['nama\_auditor'][$idx][$i] == "" && $data['jabatan'][$idx][$i] == "")

continue;

array\_push($insdata, array(

'id\_jadwal\_pelaksanaan\_opd' => $idx,

'nama\_auditor' => $data['nama\_auditor'][$idx][$i],

'jabatan' => $data['jabatan'][$idx][$i],

'editable' => $data['editable'][$idx][$i]

));

}

}

}

$this->db->where\_in('id\_jadwal\_pelaksanaan\_opd', $tmp)

->delete('auditor');

activity\_log();

if ($insdata != NULL)

$this->db->insert\_batch('auditor', $insdata);

activity\_log();

}

} elseif ($table == 'tambah\_auditor') {

$idx = $data['id\_jadwal\_pelaksanaan\_opd'][0];

$data\_insert = [

'id\_jadwal\_pelaksanaan\_opd' => $idx,

'nama\_auditor' => $data['nama\_auditor'][$idx],

'jabatan' => $data['jabatan'][$idx],

'editable' => 0

];

if ($data['select\_auditor'] == '' || $data['select\_auditor'] == null) {

$data\_insert['editable'] = 1;

}

$this->db->insert('auditor', $data\_insert);

activity\_log();

}

$this->db->trans\_complete();

}

public function delete\_data($id\_laporan)

{

$this->db->trans\_begin();

$this->db->where('id\_laporan', $id\_laporan);

**Kode Sumber 14 Model Jadwalpelaksanaan**

$this->db->delete('jadwal\_pelaksanaan');

activity\_log();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan');

activity\_log();

$this->db->trans\_complete();

}

}

* + 1. **Model Laporankinerjatriwulan**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Laporankinerjatriwulan\_model extends CI\_Model

{

public function get\_data($id\_laporan = NULL, $id\_opd = NULL)

{

$this->db->select()->from('laporan\_kinerja\_triwulan');

if ($id\_opd != NULL) {

$this->db->where('laporan\_kinerja\_triwulan.id\_opd', $id\_opd);

}

if ($id\_laporan != NULL) {

$this->db->where('laporan\_kinerja\_triwulan.id\_laporan', $id\_laporan);

}

return $this->db->get()->result();

}

public function get\_data\_by\_id($id)

{

$lktdata = $this->db->get\_where('laporan\_kinerja\_triwulan', ['id\_laporan' => $id])->result\_array()[0];

$lktdetaildata = $this->db->from('detail\_laporan\_kinerja\_triwulan')

->where('id\_laporan', $id)->get()->result\_array();

return array('lkt' => $lktdata, 'lktdetail' => $lktdetaildata);

}

public function init\_insert($id\_opd, $datalaporan, $data)

{

$this->db->trans\_start();

$this->load->model('laporan\_model', 'lp');

$this->db->insert(

'laporan',

[

'id\_opd' => $datalaporan['id\_opd'],

'id\_tipe' => $datalaporan['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

$this->db->order\_by('updated\_at', 'DESC');

$datalaporan = $this->db->get\_where('laporan', ['id\_opd' => $datalaporan['id\_opd'], 'id\_tipe' => $datalaporan['id\_tipe'],])->result\_array()[0];

$datalaporan['tgl'] = $data['tgl'];

$this->db->insert('laporan\_kinerja\_triwulan', $datalaporan);

activity\_log();

// insert second etc. table data here

$this->db->trans\_complete();

if ($this->db->trans\_status() === FALSE) {

$this->db->trans\_rollback();

return NULL;

}

return $datalaporan['id\_laporan'];

}

public function update\_data($id\_laporan, $data)

{

// printf("<pre>%s</pre>", json\_encode($data, JSON\_PRETTY\_PRINT)); die();

$table = $data['nama\_tabel'];

unset($data['nama\_tabel']);

$insdata = array();

if ($data != NULL) {

for ($i = 0; $i < sizeof(reset($data)); $i += 1) {

array\_push($insdata, array(

'id\_laporan' => $id\_laporan,

'uraian' => $data['uraian'][$i],

'indikator\_kinerja' => $data['indikator\_kinerja'][$i],

'target' => $data['target'][$i],

'realisasi\_target' => $data['realisasi\_target'][$i],

'program' => $data['program'][$i],

'anggaran' => $data['anggaran'][$i],

'capaian\_realisasi\_keuangan' => $data['capaian\_realisasi\_keuangan'][$i],

));

}

}

$this->db->trans\_begin();

if ($table == 'detail\_laporan\_kinerja\_triwulan') {

$this->db->delete('detail\_laporan\_kinerja\_triwulan', "id\_laporan = $id\_laporan");

activity\_log();

if ($insdata != NULL) {

$this->db->insert\_batch('detail\_laporan\_kinerja\_triwulan', $insdata);

activity\_log();

}

}

$this->db->trans\_complete();

}

public function delete\_data($id\_laporan)

{

$this->db->trans\_begin();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan\_kinerja\_triwulan');

activity\_log();

$this->db->where('id\_laporan', $id\_laporan);

**Kode Sumber 15 Model Laporankinerjatriwulan**

$this->db->delete('laporan');

activity\_log();

$this->db->trans\_complete();

}

}

* + 1. **Model Laporanrbareaperubahan**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Laporanrbareaperubahan\_model extends CI\_Model

{

public function get\_data\_by\_id($id)

{

$rbdata = $this->db->get\_where('laporan\_rb\_area\_perubahan', ['id\_laporan' => $id])->result\_array()[0];

$rbap = $this->db->order\_by('rincian', 'ASC')->get\_where('rb\_area\_perubahan', "id\_laporan = $id")->result\_array();

$rbaps = array();

if ($rbap != NULL) {

foreach ($rbap as $d) {

$rbaps[$d['id\_rb\_area\_perubahan']] = $this->db->get\_where('rb\_area\_perubahan\_program', "id\_rb\_area\_perubahan = $d[id\_rb\_area\_perubahan]")->result\_array();

}

}

$rbapk = array();

if ($rbaps != NULL && sizeof($rbaps) > 0) {

foreach ($rbaps as $d) {

foreach ($d as $k) {

$rbapk[$k['id\_rb\_area\_perubahan\_program']] = $this->db->get\_where('rb\_area\_perubahan\_kegiatan', "id\_rb\_area\_perubahan\_program = $k[id\_rb\_area\_perubahan\_program]")->result\_array();

}

}

}

return array('rb' => $rbdata, 'rbap' => $rbap, 'rbaps' => $rbaps, 'rbapk' => $rbapk);

}

public function init\_insert($id\_opd, $datalaporan, $data)

{

$this->db->trans\_start();

$this->load->model('laporan\_model', 'lp');

$this->db->insert(

'laporan',

[

'id\_opd' => $datalaporan['id\_opd'],

'id\_tipe' => $datalaporan['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

$this->db->order\_by('updated\_at', 'DESC');

$datalaporan = $this->db->get\_where('laporan', ['id\_opd' => $datalaporan['id\_opd'], 'id\_tipe' => $datalaporan['id\_tipe'],])->result\_array()[0];

$datalaporan['tgl'] = $data['tgl'];

$this->db->insert('laporan\_rb\_area\_perubahan', $datalaporan);

activity\_log();

// insert second etc. table data here

$this->db->trans\_complete();

if ($this->db->trans\_status() === FALSE) {

$this->db->trans\_rollback();

return NULL;

}

return $datalaporan['id\_laporan'];

}

public function update\_data($id\_laporan, $data) {

$table = $data['nama\_tabel'];

unset($data['nama\_tabel']);

$insdata = array();

$updata = array();

$this->db->trans\_begin();

if ($table == 'laporan\_rb\_area\_perubahan') {

if ($data != NULL) {

$this->db->update('laporan\_rb\_area\_perubahan', $data, "id\_laporan = $id\_laporan");

activity\_log();

}

} elseif ($table == 'rb\_area\_perubahan') {

if ($data != NULL) {

// new data

for ($i = 0; $i < sizeof(reset($data['new'])); $i += 1) {

array\_push($insdata, array(

'id\_laporan' => $id\_laporan,

'rincian' => $data['new']['rincian'][$i]

));

}

if ($insdata != NULL) {

$this->db->insert\_batch('rb\_area\_perubahan', $insdata);

activity\_log();

}

unset($data['new']);

// updated data

if (isset($data['id\_rb\_area\_perubahan'])) {

for ($i = 0; $i < sizeof($data['id\_rb\_area\_perubahan']); $i += 1) {

array\_push($updata, array(

'id\_laporan' => $id\_laporan,

'id\_rb\_area\_perubahan' => $data['id\_rb\_area\_perubahan'][$i],

'rincian' => $data['rincian'][$i],

));

}

$this->db->update\_batch('rb\_area\_perubahan', $updata, 'id\_rb\_area\_perubahan');

activity\_log();

}

// unused data

if (isset($data['to\_del']))

$this->db->where\_in('id\_rb\_area\_perubahan', $data['to\_del'])

->delete('rb\_area\_perubahan');

activity\_log();

} else {

$this->db->delete('rb\_area\_perubahan', "id\_laporan = $id\_laporan");

activity\_log();

}

} else if ($table == 'rb\_area\_perubahan\_program') {

if ($data != NULL) {

// new data

if (isset($data['new'])) {

for ($i = 0; $i < sizeof(reset($data['new'])); $i += 1) {

array\_push($insdata, array(

'id\_rb\_area\_perubahan' => $data['new']['id\_rb\_area\_perubahan'][$i],

'nama\_program' => $data['new']['nama\_program'][$i]

));

}

if ($insdata != NULL) {

$this->db->insert\_batch('rb\_area\_perubahan\_program', $insdata);

activity\_log();

}

unset($data['new']);

}

// updated data

if (isset($data['id\_rb\_area\_perubahan\_program'])) {

for ($i = 0; $i < sizeof($data['id\_rb\_area\_perubahan\_program']); $i += 1) {

array\_push($updata, array(

'id\_rb\_area\_perubahan\_program' => $data['id\_rb\_area\_perubahan\_program'][$i],

'id\_rb\_area\_perubahan' => $data['id\_rb\_area\_perubahan'][$i],

'nama\_program' => $data['nama\_program'][$i]

));

}

$this->db->update\_batch('rb\_area\_perubahan\_program', $updata, 'id\_rb\_area\_perubahan\_program');

activity\_log();

}

// unused data

if (isset($data['to\_del']))

$this->db->where\_in('id\_rb\_area\_perubahan\_program', $data['to\_del'])

->delete('rb\_area\_perubahan\_program');

activity\_log();

} else {

$del = $this->db->select('id\_rb\_area\_perubahan\_program')

->from('rb\_area\_perubahan\_program')

->join('rb\_area\_perubahan', 'rb\_area\_perubahan.id\_rb\_area\_perubahan=rb\_area\_perubahan\_program.id\_rb\_area\_perubahan')

->join('laporan\_rb\_area\_perubahan', "laporan\_rb\_area\_perubahan.id\_laporan = rb\_area\_perubahan.id\_laporan")

->where('laporan\_rb\_area\_perubahan.id\_laporan', $id\_laporan)->get()->result\_array();

$dels = array();

foreach ($del as $key => $values) {

array\_push($dels, $values['id\_rb\_area\_perubahan\_program']);

}

$this->db->where\_in('id\_rb\_area\_perubahan\_program', $dels)

->delete('rb\_area\_perubahan\_program');

activity\_log();

}

} else if ($table == 'rb\_area\_perubahan\_kegiatan') {

if ($data != NULL) {

$tmp = $data['id\_rb\_area\_perubahan\_program'];

for ($i = 0; $i < sizeof($tmp); $i += 1) {

array\_push($insdata, array(

'id\_rb\_area\_perubahan\_program' => $data['id\_rb\_area\_perubahan\_program'][$i],

'nama\_kegiatan' => $data['nama\_kegiatan'][$i],

'target\_waktu' => $data['target\_waktu'][$i],

'realisasi\_waktu' => $data['realisasi\_waktu'][$i],

'target\_anggaran' => $data['target\_anggaran'][$i],

'realisasi\_anggaran' => $data['realisasi\_anggaran'][$i],

'capaian' => $data['capaian'][$i],

'ket' => $data['ket'][$i]

));

}

$del = $this->db->select('id\_rb\_area\_perubahan\_program')

->from('rb\_area\_perubahan\_program')

->join('rb\_area\_perubahan', 'rb\_area\_perubahan.id\_rb\_area\_perubahan=rb\_area\_perubahan\_program.id\_rb\_area\_perubahan')

->join('laporan\_rb\_area\_perubahan', "laporan\_rb\_area\_perubahan.id\_laporan = rb\_area\_perubahan.id\_laporan")

->where('laporan\_rb\_area\_perubahan.id\_laporan', $id\_laporan)->get()->result\_array();

$dels = array();

foreach ($del as $key => $values) {

array\_push($dels, $values['id\_rb\_area\_perubahan\_program']);

}

$this->db->where\_in('id\_rb\_area\_perubahan\_program', $dels)

->delete('rb\_area\_perubahan\_kegiatan');

activity\_log();

if ($insdata != NULL) {

$this->db->insert\_batch('rb\_area\_perubahan\_kegiatan', $insdata);

activity\_log();

}

} else {

$del = $this->db->select('id\_rb\_area\_perubahan\_program')

->from('rb\_area\_perubahan\_program')

->join('rb\_area\_perubahan', 'rb\_area\_perubahan.id\_rb\_area\_perubahan=rb\_area\_perubahan\_program.id\_rb\_area\_perubahan')

->join('laporan\_rb\_area\_perubahan', "laporan\_rb\_area\_perubahan.id\_laporan = rb\_area\_perubahan.id\_laporan")

->where('laporan\_rb\_area\_perubahan.id\_laporan', $id\_laporan)->get()->result\_array();

$dels = array();

foreach ($del as $key => $values) {

array\_push($dels, $values['id\_rb\_area\_perubahan\_program']);

}

**Kode Sumber 16 Model Laporanrbareaperubahan**

$this->db->where\_in('id\_rb\_area\_perubahan\_program', $dels)

->delete('rb\_area\_perubahan\_kegiatan');

activity\_log();

}

}

$this->db->trans\_complete();

}

public function delete\_data($id\_laporan)

{

$this->db->trans\_begin();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan\_rb\_area\_perubahan');

activity\_log();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan');

activity\_log();

$this->db->trans\_complete();

}

}

* + 1. **Model Laporanrbfokus**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Laporanrbfokus\_model extends CI\_Model

{

public function get\_data\_by\_id($id)

{

$rbdata = $this->db->get\_where('laporan\_rb\_fokus', ['id\_laporan' => $id])->result\_array()[0];

$rbf = $this->db->order\_by('rincian', 'ASC')->get\_where('rb\_fokus', "id\_laporan = $id")->result\_array();

$rbfs = array();

if ($rbf != NULL) {

foreach ($rbf as $d) {

$rbfs[$d['id\_rb\_fokus']] = $this->db->get\_where('rb\_fokus\_sasaran', "id\_rb\_fokus = $d[id\_rb\_fokus]")->result\_array();

}

}

$rbfk = array();

if ($rbfs != NULL && sizeof($rbfs) > 0) {

foreach ($rbfs as $d) {

foreach ($d as $k) {

$rbfk[$k['id\_rb\_fokus\_sasaran']] = $this->db->get\_where('rb\_fokus\_kegiatan', "id\_rb\_fokus\_sasaran = $k[id\_rb\_fokus\_sasaran]")->result\_array();

}

} }

return array('rb' => $rbdata, 'rbf' => $rbf, 'rbfs' => $rbfs, 'rbfk' => $rbfk);

}

public function init\_insert($id\_opd, $datalaporan, $data) {

$this->db->trans\_start();

$this->load->model('laporan\_model', 'lp');

$this->db->insert(

'laporan',

[

'id\_opd' => $datalaporan['id\_opd'],

'id\_tipe' => $datalaporan['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

$this->db->order\_by('updated\_at', 'DESC');

$datalaporan = $this->db->get\_where('laporan', ['id\_opd' => $datalaporan['id\_opd'], 'id\_tipe' => $datalaporan['id\_tipe'],])->result\_array()[0];

$datalaporan['tgl'] = $data['tgl'];

$this->db->insert('laporan\_rb\_fokus', $datalaporan);

activity\_log();

// insert second etc. table data here

$this->db->trans\_complete();

if ($this->db->trans\_status() === FALSE) {

$this->db->trans\_rollback();

return NULL;

}

return $datalaporan['id\_laporan'];

}

public function update\_data($id\_laporan, $data)

{

$table = $data['nama\_tabel'];

unset($data['nama\_tabel']);

$insdata = array();

$updata = array();

$this->db->trans\_begin();

if ($table == 'laporan\_rb\_fokus') {

if ($data != NULL) {

$this->db->update('laporan\_rb\_fokus', $data, "id\_laporan = $id\_laporan");

activity\_log();

}

} elseif ($table == 'rb\_fokus') {

if ($data != NULL) {

// new data

for ($i = 0; $i < sizeof(reset($data['new'])); $i += 1) {

array\_push($insdata, array(

'id\_laporan' => $id\_laporan,

'rincian' => $data['new']['rincian'][$i] )); }

if ($insdata != NULL) {

$this->db->insert\_batch('rb\_fokus', $insdata);

activity\_log();

}

unset($data['new']);

// updated data

if (isset($data['id\_rb\_fokus'])) {

for ($i = 0; $i < sizeof($data['id\_rb\_fokus']); $i += 1) {

array\_push($updata, array(

'id\_laporan' => $id\_laporan,

'id\_rb\_fokus' => $data['id\_rb\_fokus'][$i],

'rincian' => $data['rincian'][$i],

));

}

$this->db->update\_batch('rb\_fokus', $updata, 'id\_rb\_fokus');

activity\_log();

}

// unused data

if (isset($data['to\_del']))

$this->db->where\_in('id\_rb\_fokus', $data['to\_del'])

->delete('rb\_fokus');

activity\_log();

} else {

$this->db->delete('rb\_fokus', "id\_laporan = $id\_laporan");

activity\_log();

}

} else if ($table == 'rb\_fokus\_sasaran') {

if ($data != NULL) {

// new data

if (isset($data['new'])) {

for ($i = 0; $i < sizeof(reset($data['new'])); $i += 1) {

array\_push($insdata, array(

'id\_rb\_fokus' => $data['new']['id\_rb\_fokus'][$i],

'sasaran' => $data['new']['sasaran'][$i],

'nama\_program' => $data['new']['nama\_program'][$i]

));

}

if ($insdata != NULL) {

$this->db->insert\_batch('rb\_fokus\_sasaran', $insdata);

activity\_log();

}

unset($data['new']);

}

// updated data

if (isset($data['id\_rb\_fokus\_sasaran'])) {

for ($i = 0; $i < sizeof($data['id\_rb\_fokus\_sasaran']); $i += 1) {

array\_push($updata, array(

'id\_rb\_fokus\_sasaran' => $data['id\_rb\_fokus\_sasaran'][$i],

'id\_rb\_fokus' => $data['id\_rb\_fokus'][$i],

'sasaran' => $data['sasaran'][$i],

'nama\_program' => $data['nama\_program'][$i]

));

}

$this->db->update\_batch('rb\_fokus\_sasaran', $updata, 'id\_rb\_fokus\_sasaran');

activity\_log();

}

// unused data

if (isset($data['to\_del']))

$this->db->where\_in('id\_rb\_fokus\_sasaran', $data['to\_del'])

->delete('rb\_fokus\_sasaran');

activity\_log();

} else {

$del = $this->db->select('id\_rb\_fokus\_sasaran')

->from('rb\_fokus\_sasaran')

->join('rb\_fokus', 'rb\_fokus.id\_rb\_fokus=rb\_fokus\_sasaran.id\_rb\_fokus')

->join('laporan\_rb\_fokus', "laporan\_rb\_fokus.id\_laporan = rb\_fokus.id\_laporan")

->where('laporan\_rb\_fokus.id\_laporan', $id\_laporan)->get()->result\_array();

$dels = array();

foreach ($del as $key => $values) {

array\_push($dels, $values['id\_rb\_fokus\_sasaran']);

}

$this->db->where\_in('id\_rb\_fokus\_sasaran', $dels)

->delete('rb\_fokus\_sasaran');

activity\_log();

}

} else if ($table == 'rb\_fokus\_kegiatan') {

if ($data != NULL) {

$tmp = $data['id\_rb\_fokus\_sasaran'];

for ($i = 0; $i < sizeof($tmp); $i += 1) {

array\_push($insdata, array(

'id\_rb\_fokus\_sasaran' => $data['id\_rb\_fokus\_sasaran'][$i],

'nama\_kegiatan' => $data['nama\_kegiatan'][$i],

'indikator' => $data['indikator'][$i],

'target\_output' => $data['target\_output'][$i],

'realisasi\_output' => $data['realisasi\_output'][$i],

'target\_waktu' => $data['target\_waktu'][$i],

'realisasi\_waktu' => $data['realisasi\_waktu'][$i],

'target\_anggaran' => $data['target\_anggaran'][$i],

'realisasi\_anggaran' => $data['realisasi\_anggaran'][$i],

'capaian' => $data['capaian'][$i],

'ket' => $data['ket'][$i]

));

}

$del = $this->db->select('id\_rb\_fokus\_sasaran')

->from('rb\_fokus\_sasaran')

->join('rb\_fokus', 'rb\_fokus.id\_rb\_fokus=rb\_fokus\_sasaran.id\_rb\_fokus')

->join('laporan\_rb\_fokus', "laporan\_rb\_fokus.id\_laporan = rb\_fokus.id\_laporan")

->where('laporan\_rb\_fokus.id\_laporan', $id\_laporan)->get()->result\_array();

$dels = array();

foreach ($del as $key => $values) {

array\_push($dels, $values['id\_rb\_fokus\_sasaran']);

}

$this->db->where\_in('id\_rb\_fokus\_sasaran', $dels)

->delete('rb\_fokus\_kegiatan');

activity\_log();

if ($insdata != NULL) {

$this->db->insert\_batch('rb\_fokus\_kegiatan', $insdata);

activity\_log();

}

} else {

**Kode Sumber 17 Model Laporanrbfokus**

$del = $this->db->select('id\_rb\_fokus\_sasaran')

->from('rb\_fokus\_sasaran')

->join('rb\_fokus', 'rb\_fokus.id\_rb\_fokus=rb\_fokus\_sasaran.id\_rb\_fokus')

->join('laporan\_rb\_fokus', "laporan\_rb\_fokus.id\_laporan = rb\_fokus.id\_laporan")

->where('laporan\_rb\_fokus.id\_laporan', $id\_laporan)->get()->result\_array();

$dels = array();

foreach ($del as $key => $values) {

array\_push($dels, $values['id\_rb\_fokus\_sasaran']);

}

$this->db->where\_in('id\_rb\_fokus\_sasaran', $dels)

->delete('rb\_fokus\_kegiatan');

activity\_log();

}

}

$this->db->trans\_complete();

}

public function delete\_data($id\_laporan)

{

$this->db->trans\_begin();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan\_rb\_fokus');

activity\_log();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan');

activity\_log();

$this->db->trans\_complete();

}

}

* + 1. **Model Laporanrbprioritas**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Laporanrbprioritas\_model extends CI\_Model

{

public function get\_data\_by\_id($id)

{

$rbdata = $this->db->get\_where('laporan\_rb\_prioritas', ['id\_laporan' => $id])->result\_array()[0];

$rbp = $this->db->order\_by('rincian', 'ASC')->get\_where('rb\_prioritas', "id\_laporan = $id")->result\_array();

$rbps = array();

if ($rbp != NULL) {

foreach ($rbp as $d) {

$rbps[$d['id\_rb\_prioritas']] = $this->db->get\_where('rb\_prioritas\_sasaran', "id\_rb\_prioritas = $d[id\_rb\_prioritas]")->result\_array();

}

}

$rbpk = array();

if ($rbps != NULL && sizeof($rbps) > 0) {

foreach ($rbps as $d) {

foreach ($d as $k) {

$rbpk[$k['id\_rb\_prioritas\_sasaran']] = $this->db->get\_where('rb\_prioritas\_kegiatan', "id\_rb\_prioritas\_sasaran = $k[id\_rb\_prioritas\_sasaran]")->result\_array();

}

}

}

return array('rb' => $rbdata, 'rbp' => $rbp, 'rbps' => $rbps, 'rbpk' => $rbpk);

}

public function init\_insert($id\_opd, $datalaporan, $data)

{

$this->db->trans\_start();

$this->load->model('laporan\_model', 'lp');

$this->db->insert(

'laporan',

[

'id\_opd' => $datalaporan['id\_opd'],

'id\_tipe' => $datalaporan['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

$this->db->order\_by('updated\_at', 'DESC');

$datalaporan = $this->db->get\_where('laporan', ['id\_opd' => $datalaporan['id\_opd'], 'id\_tipe' => $datalaporan['id\_tipe'],])->result\_array()[0];

$datalaporan['tgl'] = $data['tgl'];

$this->db->insert('laporan\_rb\_prioritas', $datalaporan);

activity\_log();

// insert second etc. table data here

$this->db->trans\_complete();

if ($this->db->trans\_status() === FALSE) {

$this->db->trans\_rollback();

return NULL;

}

return $datalaporan['id\_laporan'];

}

public function update\_data($id\_laporan, $data)

{

$table = $data['nama\_tabel'];

unset($data['nama\_tabel']);

$insdata = array();

$updata = array();

$this->db->trans\_begin();

if ($table == 'laporan\_rb\_prioritas') {

if ($data != NULL) {

$this->db->update('laporan\_rb\_prioritas', $data, "id\_laporan = $id\_laporan");

activity\_log();

}

} elseif ($table == 'rb\_prioritas') {

if ($data != NULL) {

// new data

for ($i = 0; $i < sizeof(reset($data['new'])); $i += 1) {

array\_push($insdata, array(

'id\_laporan' => $id\_laporan,

'rincian' => $data['new']['rincian'][$i]

));

}

if ($insdata != NULL) {

$this->db->insert\_batch('rb\_prioritas', $insdata);

activity\_log();

}

unset($data['new']);

// updated data

if (isset($data['id\_rb\_prioritas'])) {

for ($i = 0; $i < sizeof($data['id\_rb\_prioritas']); $i += 1) {

array\_push($updata, array(

'id\_laporan' => $id\_laporan,

'id\_rb\_prioritas' => $data['id\_rb\_prioritas'][$i],

'rincian' => $data['rincian'][$i],

));

}

$this->db->update\_batch('rb\_prioritas', $updata, 'id\_rb\_prioritas');

activity\_log();

}

// unused data

if (isset($data['to\_del']))

$this->db->where\_in('id\_rb\_prioritas', $data['to\_del'])

->delete('rb\_prioritas');

activity\_log();

} else {

$this->db->delete('rb\_prioritas', "id\_laporan = $id\_laporan");

activity\_log();

}

} else if ($table == 'rb\_prioritas\_sasaran') {

if ($data != NULL) {

// new data

if (isset($data['new'])) {

for ($i = 0; $i < sizeof(reset($data['new'])); $i += 1) {

array\_push($insdata, array(

'id\_rb\_prioritas' => $data['new']['id\_rb\_prioritas'][$i],

'sasaran' => $data['new']['sasaran'][$i],

'nama\_program' => $data['new']['nama\_program'][$i]

));

}

if ($insdata != NULL) {

$this->db->insert\_batch('rb\_prioritas\_sasaran', $insdata);

activity\_log();

}

unset($data['new']);

}

// updated data

if (isset($data['id\_rb\_prioritas\_sasaran'])) {

for ($i = 0; $i < sizeof($data['id\_rb\_prioritas\_sasaran']); $i += 1) {

array\_push($updata, array(

'id\_rb\_prioritas\_sasaran' => $data['id\_rb\_prioritas\_sasaran'][$i],

'id\_rb\_prioritas' => $data['id\_rb\_prioritas'][$i],

'sasaran' => $data['sasaran'][$i],

'nama\_program' => $data['nama\_program'][$i]

));

}

$this->db->update\_batch('rb\_prioritas\_sasaran', $updata, 'id\_rb\_prioritas\_sasaran');

activity\_log();

}

// unused data

if (isset($data['to\_del']))

$this->db->where\_in('id\_rb\_prioritas\_sasaran', $data['to\_del'])

->delete('rb\_prioritas\_sasaran');

activity\_log();

} else {

$del = $this->db->select('id\_rb\_prioritas\_sasaran')

->from('rb\_prioritas\_sasaran')

->join('rb\_prioritas', 'rb\_prioritas.id\_rb\_prioritas=rb\_prioritas\_sasaran.id\_rb\_prioritas')

->join('laporan\_rb\_prioritas', "laporan\_rb\_prioritas.id\_laporan = rb\_prioritas.id\_laporan")

->where('laporan\_rb\_prioritas.id\_laporan', $id\_laporan)->get()->result\_array();

$dels = array();

foreach ($del as $key => $values) {

array\_push($dels, $values['id\_rb\_prioritas\_sasaran']);

}

$this->db->where\_in('id\_rb\_prioritas\_sasaran', $dels)

->delete('rb\_prioritas\_sasaran');

activity\_log();

}

} else if ($table == 'rb\_prioritas\_kegiatan') {

if ($data != NULL) {

$tmp = $data['id\_rb\_prioritas\_sasaran'];

for ($i = 0; $i < sizeof($tmp); $i += 1) {

array\_push($insdata, array(

'id\_rb\_prioritas\_sasaran' => $data['id\_rb\_prioritas\_sasaran'][$i],

'nama\_kegiatan' => $data['nama\_kegiatan'][$i],

'indikator' => $data['indikator'][$i],

'target\_output' => $data['target\_output'][$i],

'realisasi\_output' => $data['realisasi\_output'][$i],

'target\_waktu' => $data['target\_waktu'][$i],

'realisasi\_waktu' => $data['realisasi\_waktu'][$i],

'target\_anggaran' => $data['target\_anggaran'][$i],

'realisasi\_anggaran' => $data['realisasi\_anggaran'][$i],

'capaian' => $data['capaian'][$i],

'ket' => $data['ket'][$i]

));

}

$del = $this->db->select('id\_rb\_prioritas\_sasaran')

->from('rb\_prioritas\_sasaran')

->join('rb\_prioritas', 'rb\_prioritas.id\_rb\_prioritas=rb\_prioritas\_sasaran.id\_rb\_prioritas')

->join('laporan\_rb\_prioritas', "laporan\_rb\_prioritas.id\_laporan = rb\_prioritas.id\_laporan")

->where('laporan\_rb\_prioritas.id\_laporan', $id\_laporan)->get()->result\_array();

$dels = array();

**Kode Sumber 18 Model Laporanrbprioritas**

foreach ($del as $key => $values) {

array\_push($dels, $values['id\_rb\_prioritas\_sasaran']);

}

$this->db->where\_in('id\_rb\_prioritas\_sasaran', $dels)

->delete('rb\_prioritas\_kegiatan');

activity\_log();

if ($insdata != NULL) {

$this->db->insert\_batch('rb\_prioritas\_kegiatan', $insdata);

activity\_log();

}

} else {

$del = $this->db->select('id\_rb\_prioritas\_sasaran')

->from('rb\_prioritas\_sasaran')

->join('rb\_prioritas', 'rb\_prioritas.id\_rb\_prioritas=rb\_prioritas\_sasaran.id\_rb\_prioritas')

->join('laporan\_rb\_prioritas', "laporan\_rb\_prioritas.id\_laporan = rb\_prioritas.id\_laporan")

->where('laporan\_rb\_prioritas.id\_laporan', $id\_laporan)->get()->result\_array();

$dels = array();

foreach ($del as $key => $values) {

array\_push($dels, $values['id\_rb\_prioritas\_sasaran']);

}

$this->db->where\_in('id\_rb\_prioritas\_sasaran', $dels)

->delete('rb\_prioritas\_kegiatan');

activity\_log();

}

}

$this->db->trans\_complete();

}

* + 1. **Model Laporanrbquickwins**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Laporanrbquickwins\_model extends CI\_Model

{

public function get\_data\_by\_id($id)

{

$rbdata = $this->db->get\_where('laporan\_rb\_quick\_wins', ['id\_laporan' => $id])->result\_array()[0];

$rbqw = $this->db->order\_by('rincian', 'ASC')->get\_where('rb\_quick\_wins', "id\_laporan = $id")->result\_array();

$rbqws = array();

if ($rbqw != NULL) {

foreach ($rbqw as $d) {

$rbqws[$d['id\_rb\_quick\_wins']] = $this->db->get\_where('rb\_quick\_wins\_sasaran', "id\_rb\_quick\_wins = $d[id\_rb\_quick\_wins]")->result\_array();

}

}

$rbqwk = array();

if ($rbqws != NULL && sizeof($rbqws) > 0) {

foreach ($rbqws as $d) {

foreach ($d as $k) {

$rbqwk[$k['id\_rb\_quick\_wins\_sasaran']] = $this->db->get\_where('rb\_quick\_wins\_kegiatan', "id\_rb\_quick\_wins\_sasaran = $k[id\_rb\_quick\_wins\_sasaran]")->result\_array();

}

}

}

return array('rb' => $rbdata, 'rbqw' => $rbqw, 'rbqws' => $rbqws, 'rbqwk' => $rbqwk);

}

public function init\_insert($id\_opd, $datalaporan, $data)

{

$this->db->trans\_start();

$this->load->model('laporan\_model', 'lp');

$this->db->insert(

'laporan',

[

'id\_opd' => $datalaporan['id\_opd'],

'id\_tipe' => $datalaporan['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

$this->db->order\_by('updated\_at', 'DESC');

$datalaporan = $this->db->get\_where('laporan', ['id\_opd' => $datalaporan['id\_opd'], 'id\_tipe' => $datalaporan['id\_tipe'],])->result\_array()[0];

$datalaporan['tgl'] = $data['tgl'];

$this->db->insert('laporan\_rb\_quick\_wins', $datalaporan);

activity\_log();

// insert second etc. table data here

$this->db->trans\_complete();

if ($this->db->trans\_status() === FALSE) {

$this->db->trans\_rollback();

return NULL;

}

return $datalaporan['id\_laporan'];

}

public function update\_data($id\_laporan, $data)

{

$table = $data['nama\_tabel'];

unset($data['nama\_tabel']);

$insdata = array();

$updata = array();

$this->db->trans\_begin();

if ($table == 'laporan\_rb\_quick\_wins') {

if ($data != NULL) {

$this->db->update('laporan\_rb\_quick\_wins', $data, "id\_laporan = $id\_laporan");

activity\_log();

}

} elseif ($table == 'rb\_quick\_wins') {

if ($data != NULL) {

// new data

for ($i = 0; $i < sizeof(reset($data['new'])); $i += 1) {

array\_push($insdata, array(

'id\_laporan' => $id\_laporan,

'rincian' => $data['new']['rincian'][$i]

));

}

if ($insdata != NULL) {

$this->db->insert\_batch('rb\_quick\_wins', $insdata);

activity\_log();

}

unset($data['new']);

// updated data

if (isset($data['id\_rb\_quick\_wins'])) {

for ($i = 0; $i < sizeof($data['id\_rb\_quick\_wins']); $i += 1) {

array\_push($updata, array(

'id\_laporan' => $id\_laporan,

'id\_rb\_quick\_wins' => $data['id\_rb\_quick\_wins'][$i],

'rincian' => $data['rincian'][$i],

));

}

$this->db->update\_batch('rb\_quick\_wins', $updata, 'id\_rb\_quick\_wins');

activity\_log();

}

// unused data

if (isset($data['to\_del'])) {

$this->db->where\_in('id\_rb\_quick\_wins', $data['to\_del'])

->delete('rb\_quick\_wins');

activity\_log();

}

} else {

$this->db->delete('rb\_quick\_wins', "id\_laporan = $id\_laporan");

activity\_log();

}

} else if ($table == 'rb\_quick\_wins\_sasaran') {

if ($data != NULL) {

// new data

if (isset($data['new'])) {

for ($i = 0; $i < sizeof(reset($data['new'])); $i += 1) {

array\_push($insdata, array(

'id\_rb\_quick\_wins' => $data['new']['id\_rb\_quick\_wins'][$i],

'sasaran' => $data['new']['sasaran'][$i],

'nama\_program' => $data['new']['nama\_program'][$i]

));

}

if ($insdata != NULL) {

$this->db->insert\_batch('rb\_quick\_wins\_sasaran', $insdata);

activity\_log();

}

unset($data['new']);

}

// updated data

if (isset($data['id\_rb\_quick\_wins\_sasaran'])) {

for ($i = 0; $i < sizeof($data['id\_rb\_quick\_wins\_sasaran']); $i += 1) {

array\_push($updata, array(

'id\_rb\_quick\_wins\_sasaran' => $data['id\_rb\_quick\_wins\_sasaran'][$i],

'id\_rb\_quick\_wins' => $data['id\_rb\_quick\_wins'][$i],

'sasaran' => $data['sasaran'][$i],

'nama\_program' => $data['nama\_program'][$i]

));

}

$this->db->update\_batch('rb\_quick\_wins\_sasaran', $updata, 'id\_rb\_quick\_wins\_sasaran');

activity\_log();

}

// unused data

if (isset($data['to\_del'])) {

$this->db->where\_in('id\_rb\_quick\_wins\_sasaran', $data['to\_del'])

->delete('rb\_quick\_wins\_sasaran');

activity\_log();

}

} else {

$del = $this->db->select('id\_rb\_quick\_wins\_sasaran')

->from('rb\_quick\_wins\_sasaran')

->join('rb\_quick\_wins', 'rb\_quick\_wins.id\_rb\_quick\_wins=rb\_quick\_wins\_sasaran.id\_rb\_quick\_wins')

->join('laporan\_rb\_quick\_wins', "laporan\_rb\_quick\_wins.id\_laporan = rb\_quick\_wins.id\_laporan")

->where('laporan\_rb\_quick\_wins.id\_laporan', $id\_laporan)->get()->result\_array();

$dels = array();

foreach ($del as $key => $values) {

array\_push($dels, $values['id\_rb\_quick\_wins\_sasaran']);

}

$this->db->where\_in('id\_rb\_quick\_wins\_sasaran', $dels)

->delete('rb\_quick\_wins\_sasaran');

activity\_log();

}

} else if ($table == 'rb\_quick\_wins\_kegiatan') {

if ($data != NULL) {

$tmp = $data['id\_rb\_quick\_wins\_sasaran'];

for ($i = 0; $i < sizeof($tmp); $i += 1) {

array\_push($insdata, array(

'id\_rb\_quick\_wins\_sasaran' => $data['id\_rb\_quick\_wins\_sasaran'][$i],

'nama\_kegiatan' => $data['nama\_kegiatan'][$i],

'indikator' => $data['indikator'][$i],

'target\_output' => $data['target\_output'][$i],

'realisasi\_output' => $data['realisasi\_output'][$i],

'target\_waktu' => $data['target\_waktu'][$i],

'realisasi\_waktu' => $data['realisasi\_waktu'][$i],

'target\_anggaran' => $data['target\_anggaran'][$i],

'realisasi\_anggaran' => $data['realisasi\_anggaran'][$i],

'capaian' => $data['capaian'][$i],

'ket' => $data['ket'][$i]

));

}

$del = $this->db->select('id\_rb\_quick\_wins\_sasaran')

**Kode Sumber 19 Model Laporanrbquickwins**

->from('rb\_quick\_wins\_sasaran')

->join('rb\_quick\_wins', 'rb\_quick\_wins.id\_rb\_quick\_wins=rb\_quick\_wins\_sasaran.id\_rb\_quick\_wins')

->join('laporan\_rb\_quick\_wins', "laporan\_rb\_quick\_wins.id\_laporan = rb\_quick\_wins.id\_laporan")

->where('laporan\_rb\_quick\_wins.id\_laporan', $id\_laporan)->get()->result\_array();

$dels = array();

foreach ($del as $key => $values) {

array\_push($dels, $values['id\_rb\_quick\_wins\_sasaran']);

}

$this->db->where\_in('id\_rb\_quick\_wins\_sasaran', $dels)

->delete('rb\_quick\_wins\_kegiatan');

activity\_log();

if ($insdata != NULL) {

$this->db->insert\_batch('rb\_quick\_wins\_kegiatan', $insdata);

activity\_log();

}

} else {

$del = $this->db->select('id\_rb\_quick\_wins\_sasaran')

->from('rb\_quick\_wins\_sasaran')

->join('rb\_quick\_wins', 'rb\_quick\_wins.id\_rb\_quick\_wins=rb\_quick\_wins\_sasaran.id\_rb\_quick\_wins')

->join('laporan\_rb\_quick\_wins', "laporan\_rb\_quick\_wins.id\_laporan = rb\_quick\_wins.id\_laporan")

->where('laporan\_rb\_quick\_wins.id\_laporan', $id\_laporan)->get()->result\_array();

$dels = array();

foreach ($del as $key => $values) {

array\_push($dels, $values['id\_rb\_quick\_wins\_sasaran']);

}

$this->db->where\_in('id\_rb\_quick\_wins\_sasaran', $dels)

->delete('rb\_quick\_wins\_kegiatan');

activity\_log();

}

}

$this->db->trans\_complete();

}

public function delete\_data($id\_laporan)

{

$this->db->trans\_begin();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan\_rb\_quick\_wins');

activity\_log();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan');

activity\_log();

$this->db->trans\_complete();

}

}

* + 1. **Model Laporanrbziwbk**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Laporanrbziwbk\_model extends CI\_Model

{

public function get\_data\_by\_id($id)

{

$rbdata = $this->db->get\_where('laporan\_rb\_zi\_wbk', ['id\_laporan' => $id])->result\_array()[0];

$rbziwbks = $this->db->order\_by('sasaran', 'ASC')->get\_where('rb\_zi\_wbk\_sasaran', "id\_laporan = $id")->result\_array();

$rbziwbkk = array();

if ($rbziwbks != NULL) {

foreach ($rbziwbks as $d) {

$rbziwbkk[$d['id\_rb\_zi\_wbk\_sasaran']] = $this->db->get\_where('rb\_zi\_wbk\_kegiatan', "id\_rb\_zi\_wbk\_sasaran = $d[id\_rb\_zi\_wbk\_sasaran]")->result\_array();

}

}

return array('rb' => $rbdata, 'rbziwbks' => $rbziwbks, 'rbziwbkk' => $rbziwbkk);

}

public function init\_insert($id\_opd, $datalaporan, $data)

{

$this->db->trans\_start();

$this->load->model('laporan\_model', 'lp');

$this->db->insert(

'laporan',

[

'id\_opd' => $datalaporan['id\_opd'],

'id\_tipe' => $datalaporan['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

$this->db->order\_by('updated\_at', 'DESC');

$datalaporan = $this->db->get\_where('laporan', ['id\_opd' => $datalaporan['id\_opd'], 'id\_tipe' => $datalaporan['id\_tipe'],])->result\_array()[0];

$datalaporan['tgl'] = $data['tgl'];

$this->db->insert('laporan\_rb\_zi\_wbk', $datalaporan);

activity\_log();

// insert second etc. table data here

$this->db->trans\_complete();

if ($this->db->trans\_status() === FALSE) {

$this->db->trans\_rollback();

return NULL;

}

return $datalaporan['id\_laporan'];

}

public function update\_data($id\_laporan, $data)

{

$table = $data['nama\_tabel'];

unset($data['nama\_tabel']);

$insdata = array();

$updata = array();

$this->db->trans\_begin();

if ($table == 'laporan\_rb\_zi\_wbk') {

if ($data != NULL) {

$this->db->update('laporan\_rb\_zi\_wbk', $data, "id\_laporan = $id\_laporan");

activity\_log();

}

} elseif ($table == 'rb\_zi\_wbk') {

if ($data != NULL) {

// new data

for ($i = 0; $i < sizeof(reset($data['new'])); $i += 1) {

array\_push($insdata, array(

'id\_laporan' => $id\_laporan,

'rincian' => $data['new']['rincian'][$i]

));

}

if ($insdata != NULL) {

$this->db->insert\_batch('rb\_zi\_wbk', $insdata);

activity\_log();

}

unset($data['new']);

// updated data

if (isset($data['id\_rb\_zi\_wbk'])) {

for ($i = 0; $i < sizeof($data['id\_rb\_zi\_wbk']); $i += 1) {

array\_push($updata, array(

'id\_laporan' => $id\_laporan,

'id\_rb\_zi\_wbk' => $data['id\_rb\_zi\_wbk'][$i],

'rincian' => $data['rincian'][$i],

));

}

$this->db->update\_batch('rb\_zi\_wbk', $updata, 'id\_rb\_zi\_wbk');

activity\_log();

}

// unused data

if (isset($data['to\_del'])) {

$this->db->where\_in('id\_rb\_zi\_wbk', $data['to\_del'])

->delete('rb\_zi\_wbk');

activity\_log();

}

} else {

$this->db->delete('rb\_zi\_wbk', "id\_laporan = $id\_laporan");

activity\_log();

}

} else if ($table == 'rb\_zi\_wbk\_sasaran') {

if ($data != NULL) {

// new data

if (isset($data['new'])) {

for ($i = 0; $i < sizeof(reset($data['new'])); $i += 1) {

array\_push($insdata, array(

'id\_laporan' => $id\_laporan,

'sasaran' => $data['new']['sasaran'][$i],

'nama\_program' => $data['new']['nama\_program'][$i]

));

}

if ($insdata != NULL) {

$this->db->insert\_batch('rb\_zi\_wbk\_sasaran', $insdata);

activity\_log();

}

unset($data['new']);

}

// updated data

if (isset($data['id\_rb\_zi\_wbk\_sasaran'])) {

for ($i = 0; $i < sizeof($data['id\_rb\_zi\_wbk\_sasaran']); $i += 1) {

array\_push($updata, array(

'id\_rb\_zi\_wbk\_sasaran' => $data['id\_rb\_zi\_wbk\_sasaran'][$i],

'sasaran' => $data['sasaran'][$i],

'nama\_program' => $data['nama\_program'][$i]

)); }

$this->db->update\_batch('rb\_zi\_wbk\_sasaran', $updata, 'id\_rb\_zi\_wbk\_sasaran');

activity\_log();

}

// unused data

if (isset($data['to\_del'])) {

$this->db->where\_in('id\_rb\_zi\_wbk\_sasaran', $data['to\_del'])

->delete('rb\_zi\_wbk\_sasaran');

activity\_log();

}

} else {

$del = $this->db->select('id\_rb\_zi\_wbk\_sasaran')

->from('rb\_zi\_wbk\_sasaran')

->join('laporan\_rb\_zi\_wbk', "laporan\_rb\_zi\_wbk.id\_laporan = rb\_zi\_wbk\_sasaran.id\_laporan")

->where('laporan\_rb\_zi\_wbk\_sasaran.id\_laporan', $id\_laporan)->get()->result\_array();

$dels = array();

foreach ($del as $key => $values) {

array\_push($dels, $values['id\_rb\_zi\_wbk\_sasaran']);

}

$this->db->where\_in('id\_rb\_zi\_wbk\_sasaran', $dels)

->delete('rb\_zi\_wbk\_sasaran');

activity\_log();

}

} else if ($table == 'rb\_zi\_wbk\_kegiatan') {

if ($data != NULL) {

$tmp = $data['id\_rb\_zi\_wbk\_sasaran'];

for ($i = 0; $i < sizeof($tmp); $i += 1) {

array\_push($insdata, array(

'id\_rb\_zi\_wbk\_sasaran' => $data['id\_rb\_zi\_wbk\_sasaran'][$i],

'nama\_kegiatan' => $data['nama\_kegiatan'][$i],

'indikator' => $data['indikator'][$i],

'target\_output' => $data['target\_output'][$i],

'realisasi\_output' => $data['realisasi\_output'][$i],

'target\_waktu' => $data['target\_waktu'][$i],

'realisasi\_waktu' => $data['realisasi\_waktu'][$i],

'target\_anggaran' => $data['target\_anggaran'][$i],

**Kode Sumber 20 Model Laporanrbziwbk**

'realisasi\_anggaran' => $data['realisasi\_anggaran'][$i],

'capaian' => $data['capaian'][$i],

'ket' => $data['ket'][$i]

));

}

$del = $this->db->select('id\_rb\_zi\_wbk\_sasaran')

->from('rb\_zi\_wbk\_sasaran')

->join('laporan\_rb\_zi\_wbk', "laporan\_rb\_zi\_wbk.id\_laporan = rb\_zi\_wbk\_sasaran.id\_laporan")

->where('laporan\_rb\_zi\_wbk.id\_laporan', $id\_laporan)->get()->result\_array();

$dels = array();

foreach ($del as $key => $values) {

array\_push($dels, $values['id\_rb\_zi\_wbk\_sasaran']);

}

$this->db->where\_in('id\_rb\_zi\_wbk\_sasaran', $dels)

->delete('rb\_zi\_wbk\_kegiatan');

activity\_log();

if ($insdata != NULL) {

$this->db->insert\_batch('rb\_zi\_wbk\_kegiatan', $insdata);

activity\_log();

}

} else {

$del = $this->db->select('id\_rb\_zi\_wbk\_sasaran')

->from('rb\_zi\_wbk\_sasaran')

->join('laporan\_rb\_zi\_wbk', "laporan\_rb\_zi\_wbk.id\_laporan = rb\_zi\_wbk\_sasaran.id\_laporan")

->where('laporan\_rb\_zi\_wbk.id\_laporan', $id\_laporan)->get()->result\_array();

$dels = array();

foreach ($del as $key => $values) {

array\_push($dels, $values['id\_rb\_zi\_wbk\_sasaran']);

}

$this->db->where\_in('id\_rb\_zi\_wbk\_sasaran', $dels)

->delete('rb\_zi\_wbk\_kegiatan');

activity\_log();

}

}

$this->db->trans\_complete();

}

public function delete\_data($id\_laporan)

{

$this->db->trans\_begin();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan\_rb\_zi\_wbk');

activity\_log();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan');

activity\_log();

$this->db->trans\_complete();

}

}

* + 1. **Model Monitoringkelembagaan**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Monitoringkelembagaan\_model extends CI\_Model

{

public function get\_data($id\_laporan = NULL, $id\_opd = NULL)

{

$this->db->from('monitoring\_kelembagaan');

if ($id\_opd != NULL) {

$this->db->where('monitoring\_kelembagaan.id\_opd', $id\_opd);

}

if ($id\_laporan != NULL) {

$this->db->where('monitoring\_kelembagaan.id\_laporan', $id\_laporan); }

$this->db->join('permasalahan\_kelembagaan', 'monitoring\_kelembagaan.id\_laporan = permasalahan\_kelembagaan.id\_laporan')

->join('opd', 'opd.id\_opd = permasalahan\_kelembagaan.id\_opd');

return $this->db->get()->result();

}

public function get\_data\_by\_id($id)

{

$mkdata = $this->db->get\_where('monitoring\_kelembagaan', ['id\_laporan' => $id])->result\_array()[0];

$pkdata = $this->db->select('permasalahan\_kelembagaan.\*, opd.nama\_opd')

->from('permasalahan\_kelembagaan')

->join('opd', 'permasalahan\_kelembagaan.id\_opd = opd.id\_opd')

->order\_by('nama\_opd')

->where('id\_laporan', $id)->get()->result\_array();

return array('mk' => $mkdata, 'pk' => $pkdata);

}

public function init\_insert($id\_opd, $datalaporan, $data)

{

$this->db->trans\_start();

$this->load->model('laporan\_model', 'lp');

$this->db->insert(

'laporan',

[

'id\_opd' => $datalaporan['id\_opd'],

'id\_tipe' => $datalaporan['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

$this->db->order\_by('updated\_at', 'DESC');

$datalaporan = $this->db->get\_where('laporan', ['id\_opd' => $datalaporan['id\_opd'], 'id\_tipe' => $datalaporan['id\_tipe'],])->result\_array()[0];

$datalaporan['tgl'] = $data['tgl'];

$this->db->insert('monitoring\_kelembagaan', $datalaporan);

activity\_log();

// insert second etc. table data here

$this->db->trans\_complete();

if ($this->db->trans\_status() === FALSE) {

$this->db->trans\_rollback();

return NULL;

}

return $datalaporan['id\_laporan'];

}

public function update\_data($id\_laporan, $data)

{

$table = $data['nama\_tabel'];

unset($data['nama\_tabel']);

$this->db->trans\_begin();

if ($table == 'permasalahan\_kelembagaan') {

$insdata = array();

if ($data != NULL) {

for ($i = 0; $i < sizeof(reset($data)); $i += 1) {

array\_push($insdata, array(

'id\_laporan' => $id\_laporan,

'id\_opd' => $data['id\_opd'][$i],

'permasalahan\_kelembagaan' => $data['permasalahan\_kelembagaan'][$i],

'usulan' => $data['usulan'][$i],

'dasar\_hukum' => $data['dasar\_hukum'][$i],

'ket' => $data['ket'][$i],

));

}

}

$this->db->delete('permasalahan\_kelembagaan', "id\_laporan = $id\_laporan");

activity\_log();

if ($data != NULL) {

$this->db->insert\_batch('permasalahan\_kelembagaan', $insdata);

activity\_log();

}

} elseif ($table == 'tambah\_permasalahan\_kelembagaan') {

$data\_insert = [

'id\_laporan' => $id\_laporan,

'id\_opd' => $data['id\_opd'],

'permasalahan\_kelembagaan' => $data['permasalahan\_kelembagaan'],

'usulan' => $data['usulan'],

'dasar\_hukum' => $data['dasar\_hukum'],

'ket' => $data['ket']

];

$this->db->insert('permasalahan\_kelembagaan', $data\_insert);

activity\_log();

}

$this->db->trans\_complete();

}

public function delete\_data($id\_laporan)

{

$this->db->trans\_begin();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('monitoring\_kelembagaan');

activity\_log();

$this->db->where('id\_laporan', $id\_laporan);

**Kode Sumber 21Model Monitoringkelembagaan**

$this->db->delete('laporan');

activity\_log();

$this->db->trans\_complete();

}

}

* + 1. **Model Pelayananpublik**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Pelayananpublik\_model extends CI\_Model

{

public function get\_data($id\_laporan = NULL, $id\_opd = NULL)

{

$this->db->from('pelayanan\_publik');

if ($id\_opd != NULL) {

$this->db->where('pelayanan\_publik.id\_opd', $id\_opd);

}

if ($id\_laporan != NULL) {

$this->db->where('pelayanan\_publik.id\_laporan', $id\_laporan);

}

$this->db->join('pelayanan\_publik\_opd', 'pelayanan\_publik.id\_laporan = pelayanan\_publik\_opd.id\_laporan')

->join('opd', 'opd.id\_opd = pelayanan\_publik\_opd.id\_opd');

return $this->db->get()->result();

}

public function get\_data\_by\_id($id)

{

$ppdata = $this->db->get\_where('pelayanan\_publik', ['id\_laporan' => $id])->result\_array()[0];

$ppopddata = $this->db->select('pelayanan\_publik\_opd.\*, opd.nama\_opd')

->from('pelayanan\_publik\_opd')

->join('opd', 'pelayanan\_publik\_opd.id\_opd = opd.id\_opd')

->order\_by('nama\_opd')

->where('id\_laporan', $id)->get()->result\_array();

return array('pp' => $ppdata, 'ppopd' => $ppopddata);

}

public function init\_insert($id\_opd, $datalaporan, $data)

{

$this->db->trans\_start();

$this->load->model('laporan\_model', 'lp');

$this->db->insert(

'laporan',

[

'id\_opd' => $datalaporan['id\_opd'],

'id\_tipe' => $datalaporan['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

**Kode Sumber 22 Model Pelayananpublik**

$this->db->order\_by('updated\_at', 'DESC');

$datalaporan = $this->db->get\_where('laporan', ['id\_opd' => $datalaporan['id\_opd'], 'id\_tipe' => $datalaporan['id\_tipe'],])->result\_array()[0];

$datalaporan['tgl'] = $data['tgl'];

$this->db->insert('pelayanan\_publik', $datalaporan);

activity\_log();

// insert second etc. table data here

$this->db->trans\_complete();

if ($this->db->trans\_status() === FALSE) {

$this->db->trans\_rollback();

return NULL;

}

return $datalaporan['id\_laporan'];

}

public function update\_data($id\_laporan, $data)

{

$table = $data['nama\_tabel'];

unset($data['nama\_tabel']);

$insdata = array();

if ($data != NULL) {

for ($i = 0; $i < sizeof(reset($data)); $i += 1) {

array\_push($insdata, array(

'id\_laporan' => $id\_laporan,

'id\_opd' => $data['id\_opd'][$i],

'indeks\_pelayanan\_publik' => $data['indeks\_pelayanan\_publik'][$i]

));

}

}

$this->db->trans\_begin();

if ($table == 'pelayanan\_publik\_opd') {

$this->db->delete('pelayanan\_publik\_opd', "id\_laporan = $id\_laporan");

activity\_log();

if ($data != NULL) {

$this->db->insert\_batch('pelayanan\_publik\_opd', $insdata);

activity\_log();

}

}

$this->db->trans\_complete();

}

public function delete\_data($id\_laporan)

{

$this->db->trans\_begin();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('pelayanan\_publik');

activity\_log();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan');

activity\_log();

$this->db->trans\_complete();

}

}

* + 1. **Model Pemantauantindaklanjut**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Pemantauantindaklanjut\_model extends CI\_Model

{

public function get\_data($id\_laporan = NULL, $id\_opd = NULL)

{

$this->db->from('pemantauan\_tindak\_lanjut');

if ($id\_opd != NULL) {

$this->db->where('pemantauan\_tindak\_lanjut.id\_opd', $id\_opd);

}

if ($id\_laporan != NULL) {

$this->db->where('pemantauan\_tindak\_lanjut.id\_laporan', $id\_laporan);

}

$this->db->join('temuan', 'temuan.id\_laporan = pemantauan\_tindak\_lanjut.id\_laporan')

->join('hasil\_temuan', 'hasil\_temuan.id\_temuan = temuan.id\_temuan');

return $this->db->get()->result();

}

public function get\_data\_by\_id($id)

{

$ptldata = $this->db->get\_where('pemantauan\_tindak\_lanjut', ['id\_laporan' => $id])->result\_array()[0];

$temuandata = $this->db->order\_by('nama\_temuan')->get\_where('temuan', "id\_laporan = $id")->result\_array();

$htemuan = array();

if ($temuandata != NULL) {

foreach ($temuandata as $d) {

$htemuan[$d['id\_temuan']] = $this->db->get\_where('hasil\_temuan', "id\_temuan = $d[id\_temuan]")->result\_array();

}

}

return array('ptl' => $ptldata, 'temuan' => $temuandata, 'htemuan' => $htemuan);

}

public function init\_insert($id\_opd, $datalaporan, $data)

{

$this->db->trans\_start();

$this->load->model('laporan\_model', 'lp');

$this->db->insert(

'laporan',

[

'id\_opd' => $datalaporan['id\_opd'],

'id\_tipe' => $datalaporan['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

$this->db->order\_by('updated\_at', 'DESC');

$datalaporan = $this->db->get\_where('laporan', ['id\_opd' => $datalaporan['id\_opd'], 'id\_tipe' => $datalaporan['id\_tipe'],])->result\_array()[0];

$datalaporan['tgl'] = $data['tgl'];

$this->db->insert('pemantauan\_tindak\_lanjut', $datalaporan);

activity\_log();

// insert second etc. table data here

$this->db->trans\_complete();

if ($this->db->trans\_status() === FALSE) {

$this->db->trans\_rollback();

return NULL;

}

return $datalaporan['id\_laporan'];

}

public function update\_data($id\_laporan, $data)

{

$table = $data['nama\_tabel'];

unset($data['nama\_tabel']);

$insdata = array();

$updata = array();

$this->db->trans\_begin();

if ($table == 'pemantauan\_tindak\_lanjut') {

if ($data != NULL) {

$this->db->update('pemantauan\_tindak\_lanjut', $data, "id\_laporan = $id\_laporan");

activity\_log();

}

} elseif ($table == 'temuan') {

if ($data != NULL) {

// new data

if (isset($data['new'])) {

for ($i = 0; $i < sizeof(reset($data['new'])); $i += 1) {

array\_push($insdata, array(

'id\_laporan' => $id\_laporan,

'nama\_temuan' => $data['new']['nama\_temuan'][$i]

));

}

}

if ($insdata != NULL) {

$this->db->insert\_batch('temuan', $insdata);

activity\_log();

}

unset($data['new']);

// updated data

if ($data['id\_temuan'] != NULL) {

for ($i = 0; $i < sizeof($data['id\_temuan']); $i += 1) {

array\_push($updata, array(

'id\_temuan' => $data['id\_temuan'][$i],

'nama\_temuan' => $data['nama\_temuan'][$i]

));

}

$this->db->update\_batch('temuan', $updata, 'id\_temuan');

activity\_log();

}

// unused data

if (isset($data['to\_del'])) {

$this->db->where\_in('id\_temuan', $data['to\_del'])->delete('temuan');

activity\_log(); }

} else {

$this->db->delete('temuan', "id\_laporan = $id\_laporan");

activity\_log();

}

} else if ($table == 'hasil\_temuan') {

if ($data != NULL) {

$tmp = $data['id\_temuan'];

for ($i = 0; $i < sizeof($tmp); $i += 1) {

array\_push($insdata, array(

'id\_temuan' => $data['id\_temuan'][$i],

'rekomendasi' => $data['rekomendasi'][$i],

'status\_rekomendasi' => $data['status\_rekomendasi'][$i],

'tindak\_lanjut' => $data['tindak\_lanjut'][$i],

'status\_tindak\_lanjut' => $data['status\_tindak\_lanjut'][$i],

'catatan\_bpk' => $data['catatan\_bpk'][$i]

));

}

$this->db->where\_in('id\_temuan', $tmp)

->delete('hasil\_temuan');

activity\_log();

if ($insdata != NULL) {

$this->db->insert\_batch('hasil\_temuan', $insdata);

activity\_log();

}

} else {

$todel = $this->db->select('id\_temuan')->from('temuan')

->where('temuan.id\_laporan', $id\_laporan)->get()->result\_array();

activity\_log();

$del = array();

foreach ($todel as $key => $d) {

array\_push($del, $d['id\_temuan']);

}

$this->db->where\_in('id\_temuan', $del)->delete('hasil\_temuan');

activity\_log();

}

} elseif ($table == 'tambah\_hasil\_temuan') {

$data\_insert = [

'id\_temuan' => $data['id\_temuan'],

'rekomendasi' => $data['rekomendasi'],

'status\_rekomendasi' => $data['status\_rekomendasi'],

'tindak\_lanjut' => $data['tindak\_lanjut'],

'status\_tindak\_lanjut' => $data['status\_tindak\_lanjut'],

'catatan\_bpk' => $data['catatan\_bpk']

];

if ($data\_insert != NULL) {

$this->db->insert('hasil\_temuan', $data\_insert);

activity\_log();

}

}

$this->db->trans\_complete();

}

public function delete\_data($id\_laporan)

{

**Kode Sumber 23 Model Pemantauantindaklanjut**

$this->db->trans\_begin();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('pemantauan\_tindak\_lanjut');

activity\_log();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan');

activity\_log();

$this->db->trans\_complete();

}

}

* + 1. **Model Realisasifisik**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Realisasifisik\_model extends CI\_Model

{

public function get\_data($id\_laporan = NULL, $id\_opd = NULL)

{

$this->db->from('realisasi\_fisik');

if ($id\_opd != NULL) {

$this->db->where('realisasi\_fisik.id\_opd', $id\_opd);

}

if ($id\_laporan != NULL) {

$this->db->where('realisasi\_fisik.id\_laporan', $id\_laporan);

}

$this->db->join('program', 'realisasi\_fisik.id\_laporan = program.id\_laporan')

->join('capaian', 'program.kode\_program = capaian.kode\_program')

->join('kegiatan', 'program.kode\_program = kegiatan.kode\_program')

->join('keluaran', 'kegiatan.kode\_kegiatan = keluaran.kode\_kegiatan')

->join('hasil', 'kegiatan.kode\_kegiatan = hasil.kode\_kegiatan');

return $this->db->get()->result\_array();

}

public function get\_data\_by\_id($id)

{

$rfdata = $this->db->get\_where('realisasi\_fisik', ['id\_laporan' => $id])->result\_array();

if ($rfdata != NULL) $rfdata = $rfdata[0];

$progdata = $this->db->from('program')->like('kode\_program', $id, 'after')->get()->result\_array();

$kgdata = array();

foreach ($progdata as $d) {

$kgdata[$d['kode\_program']] = $this->db->from('kegiatan')->select('program.nama\_program,kegiatan.\*')

->join('program', 'program.kode\_program = kegiatan.kode\_program')

->like('kode\_kegiatan', $d['kode\_program'], 'after')->get()->result\_array();

}

return array('rf' => $rfdata, 'prog' => $progdata, 'kg' => $kgdata);

}

public function insert\_fetch($data\_rf, $data\_prog, $data\_kg)

{

$this->db->trans\_begin();

$this->db->insert('realisasi\_fisik', $data\_rf);

activity\_log();

$this->db->insert\_batch('program', $data\_prog);

activity\_log();

$this->db->insert\_batch('kegiatan', $data\_kg);

activity\_log();

$this->db->trans\_complete();

}

public function init\_insert($id\_opd, $datalaporan, $data)

{

$this->load->model('laporan\_model', 'lp');

$this->load->model('api\_sipp\_model', 'sipp');

$this->load->model('api\_sikd\_model', 'sikd');

$this->db->trans\_start();

$this->db->insert(

'laporan',

[

'id\_opd' => $datalaporan['id\_opd'],

'id\_tipe' => $datalaporan['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

$this->db->order\_by('updated\_at', 'DESC');

$datalaporan = $this->db->get\_where('laporan', ['id\_opd' => $datalaporan['id\_opd'], 'id\_tipe' => $datalaporan['id\_tipe'],])->result\_array()[0];

$datalaporan['tgl'] = $data['tgl'];

$this->db->insert('realisasi\_fisik', $datalaporan);

activity\_log();

$fet = $this->sipp->api\_fetch\_data($id\_opd, $datalaporan, date('Y', strtotime($data['tgl'])));

if ($fet != NULL && $fet != false && sizeof($fet) > 0) {

$this->db->insert\_batch('program', $fet['prog']);

activity\_log();

$this->db->insert\_batch('kegiatan', $fet['kg']);

activity\_log();

}

$this->db->trans\_complete();

if ($this->db->trans\_status() === FALSE) {

$this->db->trans\_rollback();

return NULL;

}

return $datalaporan['id\_laporan'];

}

public function update\_data($id\_laporan, $data)

{

$this->load->model('api\_sipp\_model', 'sipp');

$tmp = -1;

$table = $data['nama\_tabel'];

unset($data['nama\_tabel']);

$this->db->trans\_begin();

if ($table == 'program') {

$kode = $data['kode\_program'];

$rkinerja = $data['capaian\_realisasi\_kinerja'];

$rkeuangan = $data['capaian\_realisasi\_keuangan'];

for ($i = 0; $i < sizeof($kode); $i += 1) {

$this->db->update(

$table,

[

'capaian\_realisasi\_kinerja' => "$rkinerja[$i]",

'capaian\_realisasi\_keuangan' => "$rkeuangan[$i]"

],

"kode\_program = '$kode[$i]'"

);

activity\_log();

}

} else if ($table == 'kegiatan') {

$kode = $data['kode\_kegiatan'];

$rkkinerja = $data['keluaran\_realisasi\_kinerja'];

$rhkinerja = $data['hasil\_realisasi\_kinerja'];

$rkeuangan = $data['realisasi\_keuangan'];

for ($i = 0; $i < sizeof($kode); $i += 1) {

$this->db->update(

$table,

[

'keluaran\_realisasi\_kinerja' => "$rkkinerja[$i]",

'hasil\_realisasi\_kinerja' => "$rhkinerja[$i]",

'realisasi\_keuangan' => "$rkeuangan[$i]"

],

"kode\_kegiatan = '$kode[$i]'"

);

activity\_log();

}

} elseif ($table == 'updateapi') {

$id\_opd = $this->session->tempdata('id\_opd');

$datalaporan = $this->db->get\_where('laporan', ['id\_laporan' => $id\_laporan])->result\_array();

if ($datalaporan != NULL) {

$datalaporan = $datalaporan[0];

}

$fet = $this->sipp->api\_fetch\_data($id\_opd, $datalaporan, date('Y', strtotime($data['tgl'])));

if ($fet != NULL && sizeof($fet) > 0) {

$this->db->update\_batch('program', $fet['prog'], 'kode\_program');

activity\_log();

$this->db->update\_batch('kegiatan', $fet['kg'], 'kode\_kegiatan');

activity\_log();

}

}

if ($this->db->trans\_status() === FALSE) {

echo "GAGAL";

$this->db->trans\_rollback();

return NULL;

} else {

echo "SUKSES";

$this->db->trans\_commit();

}

}

**Kode Sumber 24 Model Realisasifisik**

public function delete\_data($id\_laporan)

{

$this->db->trans\_begin();

$this->db->like('kode\_kegiatan', $id\_laporan . '-', 'after');

$this->db->delete('kegiatan');

activity\_log();

$this->db->like('kode\_program', $id\_laporan . '-', 'after');

$this->db->delete('program');

activity\_log();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('realisasi\_fisik');

activity\_log();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan');

activity\_log();

$this->db->trans\_complete();

}

}

* + 1. **Model Rekappokja**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Rekappokja\_model extends CI\_Model

{

public function get\_data($id\_laporan = NULL, $id\_opd = NULL)

{

$this->db->from('rekap\_pokja');

if ($id\_opd != NULL) {

$this->db->where('rekap\_pokja.id\_opd', $id\_opd);

}

if ($id\_laporan != NULL) {

$this->db->where('rekap\_pokja.id\_laporan', $id\_laporan);

}

$this->db->join('paket\_kerja', 'rekap\_pokja.id\_laporan = paket\_kerja.id\_laporan');

return $this->db->get()->result();

}

public function get\_paket\_kerja()

{

return $this->db->from('paket\_kerja')

->order\_by('nama\_paket\_kerja')

->join('detail\_rekap\_pokja', 'detail\_rekap\_pokja.id\_detail\_rekap\_pokja = paket\_kerja.id\_detail\_rekap\_pokja')

->get()->result\_array();

}

public function get\_data\_by\_id($id)

{

$rpdata = $this->db->get\_where('rekap\_pokja', ['id\_laporan' => $id])->result\_array();

if ($rpdata != NULL) $rpdata = $rpdata[0];

else return NULL;

$pdata = $this->db->get\_where('pegawai', "id\_laporan = $id")->result\_array();

$drpdata = array();

if ($pdata != NULL) {

foreach ($pdata as $d) {

$drpdata[$d['id\_pegawai']] =

$this->db->select("id\_pegawai, detail\_rekap\_tender.id\_detail\_rekap\_tender,jabatan, detail\_rekap\_pokja.ket, nama\_paket\_kerja, pagu")

->from("detail\_rekap\_pokja")->join("detail\_rekap\_tender", "detail\_rekap\_pokja.id\_detail\_rekap\_tender = detail\_rekap\_tender.id\_detail\_rekap\_tender")

->where("detail\_rekap\_pokja.id\_pegawai", $d['id\_pegawai'])->get()->result\_array();

}

}

return array('rp' => $rpdata, 'p' => $pdata, 'drp' => $drpdata);

}

public function init\_insert($id\_opd, $datalaporan, $data)

{

$this->db->trans\_start();

$this->load->model('laporan\_model', 'lp');

$this->db->insert(

'laporan',

[

'id\_opd' => $datalaporan['id\_opd'],

'id\_tipe' => $datalaporan['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

$this->db->order\_by('updated\_at', 'DESC');

$datalaporan = $this->db->get\_where('laporan', ['id\_opd' => $datalaporan['id\_opd'], 'id\_tipe' => $datalaporan['id\_tipe'],])->result\_array()[0];

$datalaporan['tgl'] = $data['tgl'];

$this->db->insert('rekap\_pokja', $datalaporan);

activity\_log();

// insert second etc. table data here

$this->db->trans\_complete();

if ($this->db->trans\_status() === FALSE) {

$this->db->trans\_rollback();

return NULL;

}

return $datalaporan['id\_laporan'];

}

public function update\_data($id\_laporan, $data)

{

$table = $data['nama\_tabel'];

unset($data['nama\_tabel']);

$insdata = array();

$updata = array();

$this->db->trans\_begin();

if ($table == 'pegawai') {

if ($data != NULL) {

// new data

for ($i = 0; $i < sizeof(reset($data['new'])); $i += 1) {

array\_push($insdata, array(

'id\_laporan' => $id\_laporan,

'nama' => $data['new']['nama'][$i]

));

}

if ($insdata != NULL) {

$this->db->insert\_batch('pegawai', $insdata);

activity\_log();

}

unset($data['new']);

// updated data

if (isset($data['id\_pegawai'])) {

for ($i = 0; $i < sizeof($data['id\_pegawai']); $i += 1) {

array\_push($updata, array(

'id\_laporan' => $id\_laporan,

'id\_pegawai' => $data['id\_pegawai'][$i],

'nama' => $data['nama'][$i]

));

}

$this->db->update\_batch('pegawai', $updata, 'id\_pegawai');

activity\_log();

}

// unused data

if (isset($data['to\_del'])) {

$this->db->where\_in('id\_pegawai', $data['to\_del'])

->delete('pegawai');

activity\_log();

}

} else {

$this->db->delete('pegawai', "id\_laporan = $id\_laporan");

activity\_log();

}

} else if ($table == 'detail\_rekap\_pokja') {

if ($data != NULL) {

// new data

for ($i = 0; $i < sizeof($data['jabatan']); $i += 1) {

array\_push($insdata, array(

'id\_detail\_rekap\_tender' => $data['id\_detail\_rekap\_tender'][$i],

'id\_pegawai' => $data['id\_pegawai'][$i],

'jabatan' => $data['jabatan'][$i],

'ket' => $data['ket'][$i]

));

}

$iddel = array\_values($data['id\_pegawai']);

$this->db->where\_in('id\_pegawai', $iddel)->delete('detail\_rekap\_pokja');

activity\_log();

if ($insdata != NULL) {

$this->db->insert\_batch('detail\_rekap\_pokja', $insdata);

activity\_log();

}

} else {

$todel = $this->db->select('id\_pegawai')->from('pegawai')->join('rekap\_pokja', 'rekap\_pokja.id\_laporan = pegawai.id\_laporan')

**Kode Sumber 25 Modelrekappokja**

->where('pegawai.id\_laporan', $id\_laporan)->get()->result\_array();

$dels = array();

foreach ($todel as $key => $values) {

array\_push($dels, $values['id\_pegawai']);

}

if ($dels != NULL) {

$this->db->where\_in('id\_pegawai', $dels)->delete('detail\_rekap\_pokja');

activity\_log();

}

}

}

$this->db->trans\_complete();

}

public function delete\_data($id\_laporan)

{

$this->db->trans\_begin();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('rekap\_pokja');

activity\_log();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan');

activity\_log();

$this->db->trans\_complete();

}

}

* + 1. **Model Rekaptender**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Rekaptender\_model extends CI\_Model

{

public function get\_data($id\_laporan = NULL, $id\_opd = NULL)

{

$this->db->from('rekap\_tender');

if ($id\_opd != NULL) {

$this->db->where('rekap\_tender.id\_opd', $id\_opd);

}

if ($id\_laporan != NULL) {

$this->db->where('rekap\_tender.id\_laporan', $id\_laporan);

}

$this->db->join('detail\_rekap\_tender', 'rekap\_tender.id\_laporan = detail\_rekap\_tender.id\_laporan')

->join('paket\_kerja', 'detail\_rekap\_tender.id\_paket\_kerja = paket\_kerja.id\_paket\_kerja');

return $this->db->get()->result();

}

public function get\_data\_by\_id($id)

{

$rtdata = $this->db->get\_where('rekap\_tender', ['id\_laporan' => $id])->result\_array()[0];

$drtdata = $this->db->select('detail\_rekap\_tender.\*, opd.id\_opd, opd.nama\_opd')

->from('detail\_rekap\_tender')

->join('opd', 'detail\_rekap\_tender.id\_opd = opd.id\_opd')

->where('id\_laporan', $id)->get()->result\_array();

// var\_dump($drtdata); die();

return array('rt' => $rtdata, 'drt' => $drtdata);

}

public function get\_paket\_kerja()

{

return $this->db->select("id\_detail\_rekap\_tender, nama\_paket\_kerja, pagu")

->from('detail\_rekap\_tender')->get()->result\_array();

}

public function init\_insert($id\_opd, $datalaporan, $data)

{

$this->db->trans\_start();

$this->load->model('laporan\_model', 'lp');

$this->db->insert(

'laporan',

[

'id\_opd' => $datalaporan['id\_opd'],

'id\_tipe' => $datalaporan['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

$this->db->order\_by('updated\_at', 'DESC');

$datalaporan = $this->db->get\_where('laporan', ['id\_opd' => $datalaporan['id\_opd'], 'id\_tipe' => $datalaporan['id\_tipe'],])->result\_array()[0];

$datalaporan['tgl'] = $data['tgl'];

$this->db->insert('rekap\_tender', $datalaporan);

activity\_log();

// insert second etc. table data here

$this->db->trans\_complete();

if ($this->db->trans\_status() === FALSE) {

$this->db->trans\_rollback();

return NULL;

}

return $datalaporan['id\_laporan'];

}

public function update\_data($id\_laporan, $data)

{

$table = $data['nama\_tabel'];

unset($data['nama\_tabel']);

$insdata = array();

$updata = array();

$this->db->trans\_begin();

if($table == 'rekap\_tender'){

$this->db->update('rekap\_tender', $data, "id\_laporan = $id\_laporan");

}

elseif ($table == 'detail\_rekap\_tender') {

if ($data != NULL) {

// new data

if (isset($data['new'])) {

for ($i = 0; $i < sizeof(reset($data['new'])); $i += 1) {

array\_push($insdata, array(

'id\_laporan' => $id\_laporan,

'id\_opd' => $data['new']['id\_opd'][$i],

'nama\_paket\_kerja' => $data['new']['nama\_paket\_kerja'][$i],

'pagu' => $data['new']['pagu'][$i],

'nilai\_hps' => $data['new']['nilai\_hps'][$i],

'pemenang' => $data['new']['pemenang'][$i],

'harga\_kontrak' => $data['new']['harga\_kontrak'][$i],

'ket' => $data['new']['ket'][$i]

));

}

}

if ($insdata != NULL) {

$this->db->insert\_batch('detail\_rekap\_tender', $insdata);

activity\_log();

}

unset($data['new']);

// updated data

if ($data['id\_detail\_rekap\_tender'] != NULL) {

for ($idx = 0; $idx < sizeof($data['id\_detail\_rekap\_tender']); $idx += 1) {

array\_push($updata, array(

'id\_laporan' => $id\_laporan,

'id\_detail\_rekap\_tender' => $data['id\_detail\_rekap\_tender'][$idx],

'id\_opd' => $data['id\_opd'][$idx],

'nama\_paket\_kerja' => $data['nama\_paket\_kerja'][$idx],

'pagu' => $data['pagu'][$idx],

'nilai\_hps' => $data['nilai\_hps'][$idx],

'pemenang' => $data['pemenang'][$idx],

'harga\_kontrak' => $data['harga\_kontrak'][$idx],

'ket' => $data['ket'][$idx]

));

}

$this->db->update\_batch('detail\_rekap\_tender', $updata, 'id\_detail\_rekap\_tender');

activity\_log();

}

// unused data

if (isset($data['to\_del'])) {

$this->db->where\_in('id\_detail\_rekap\_tender', $data['to\_del'])->delete('detail\_rekap\_tender');

activity\_log();

}

} else {

$this->db->delete('detail\_rekap\_tender', "id\_laporan = $id\_laporan");

activity\_log();

}

}

$this->db->trans\_complete();

}

public function delete\_data($id\_laporan)

{

**Kode Sumber 26 Model Rekaptender**

$this->db->trans\_begin();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('rekap\_tender');

activity\_log();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan');

activity\_log();

$this->db->trans\_complete();

}

}

* + 1. **Model Sotk**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Sotk\_model extends CI\_Model

{

public function get\_data($id\_laporan = NULL, $id\_opd = NULL)

{

$this->db->from('sotk');

if ($id\_opd != NULL) {

$this->db->where('sotk.id\_opd', $id\_opd);

}

if ($id\_laporan != NULL) {

$this->db->where('sotk.id\_laporan', $id\_laporan);

}

$this->db->join('sotk\_opd', 'sotk.id\_laporan = sotk\_opd.id\_laporan')

->join('opd', 'opd.id\_opd = sotk\_opd.id\_opd')

->order\_by('sotk.id\_laporan', 'ASC');

$res = $this->db->get()->result\_array();

$current\_id = $res[0]['id\_laporan'];

$ret = [];

foreach ($res as $row) {

if ($current\_id != $row['id\_laporan']) {

$current\_id = $row['id\_laporan'];

$ret['id\_laporan'] = $row['id\_laporan'];

$ret['id\_opd'] = $row['id\_opd'];

$ret['id\_tipe'] = $row['id\_tipe'];

$row['created\_at'] = $row['created\_at'];

$row['updated\_at'] = $row['updated\_at'];

$row['tahun'] = $row['tahun'];

$row['sotk\_opd'] = [];

}

}

}

public function get\_data\_by\_id($id)

{

$sotkdata = $this->db->get\_where('sotk', ['id\_laporan' => $id])->result\_array()[0];

$sotkopddata = $this->db->select('sotk\_opd.\*, opd.nama\_opd')

->from('sotk\_opd')

->join('opd', 'sotk\_opd.id\_opd = opd.id\_opd')

->order\_by('nama\_opd')

->where('id\_laporan', $id)->get()->result\_array();

return array('sotk' => $sotkdata, 'sotkopd' => $sotkopddata);

}

public function init\_insert($id\_opd, $datalaporan, $data)

{

$this->db->trans\_start();

$this->load->model('laporan\_model', 'lp');

$this->db->insert(

'laporan',

[

'id\_opd' => $datalaporan['id\_opd'],

'id\_tipe' => $datalaporan['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

$this->db->order\_by('updated\_at', 'DESC');

$datalaporan = $this->db->get\_where('laporan', ['id\_opd' => $datalaporan['id\_opd'], 'id\_tipe' => $datalaporan['id\_tipe'],])->result\_array()[0];

$datalaporan['tgl'] = $data['tgl'];

$this->db->insert('sotk', $datalaporan);

activity\_log();

// insert second etc. table data here

$this->db->trans\_complete();

if ($this->db->trans\_status() === FALSE) {

$this->db->trans\_rollback();

return NULL;

}

return $datalaporan['id\_laporan'];

}

public function update\_data($id\_laporan, $data)

{

$table = $data['nama\_tabel'];

unset($data['nama\_tabel']);

$insdata = array();

if ($data != NULL) {

for ($i = 0; $i < sizeof(reset($data)); $i += 1) {

array\_push($insdata, array(

'id\_laporan' => $id\_laporan,

'id\_opd' => $data['id\_opd'][$i],

'besaran' => $data['besaran'][$i]

));

}

}

$this->db->trans\_begin();

if ($table == 'sotk\_opd') {

$this->db->delete('sotk\_opd', "id\_laporan = $id\_laporan");

if ($data != NULL) {

$this->db->insert\_batch('sotk\_opd', $insdata);

**Kode Sumber 27 Model Sotk**

activity\_log();

}

}

$this->db->trans\_complete();

}

public function delete\_data($id\_laporan)

{

$this->db->trans\_begin();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('sotk');

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan');

$this->db->trans\_complete();

}

}

* + 1. **Model Tatalaksana**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Tatalaksana\_model extends CI\_Model

{

public function get\_data($id\_laporan = NULL, $id\_opd = NULL)

{

$this->db->from('tatalaksana');

if ($id\_opd != NULL) {

$this->db->where('tatalaksana.id\_opd', $id\_opd);

}

if ($id\_laporan != NULL) {

$this->db->where('tatalaksana.id\_laporan', $id\_laporan);

}

$this->db->join('tatalaksana\_opd', 'tatalaksana.id\_laporan = tatalaksana\_opd.id\_laporan')

->join('opd', 'opd.id\_opd = tatalaksana\_opd.id\_opd');

return $this->db->get()->result();

}

public function get\_data\_by\_id($id)

{

$tdata = $this->db->get\_where('tatalaksana', ['id\_laporan' => $id])->result\_array()[0];

$topddata = $this->db->select('tatalaksana\_opd.\*, opd.nama\_opd')

->from('tatalaksana\_opd')

->join('opd', 'tatalaksana\_opd.id\_opd = opd.id\_opd')

->where('id\_laporan', $id)->get()->result\_array();

return array('t' => $tdata, 'topd' => $topddata);

}

public function init\_insert($id\_opd, $datalaporan, $data)

{

$this->db->trans\_start();

$this->load->model('laporan\_model', 'lp');

$this->db->insert(

'laporan', [

'id\_opd' => $datalaporan['id\_opd'],

'id\_tipe' => $datalaporan['id\_tipe'],

'created\_at' => date('Y-m-d H:i:s', time()),

'updated\_at' => date('Y-m-d H:i:s', time()),

]

);

activity\_log();

$this->db->order\_by('updated\_at', 'DESC');

$datalaporan = $this->db->get\_where('laporan', ['id\_opd' => $datalaporan['id\_opd'], 'id\_tipe' => $datalaporan['id\_tipe'],])->result\_array()[0];

$datalaporan['tgl'] = $data['tgl'];

$this->db->insert('tatalaksana', $datalaporan);

activity\_log();

// insert second etc. table data here

$this->db->trans\_complete();

if ($this->db->trans\_status() === FALSE) {

$this->db->trans\_rollback();

return NULL;

}

return $datalaporan['id\_laporan'];

}

public function update\_data($id\_laporan, $data)

{

$table = $data['nama\_tabel'];

unset($data['nama\_tabel']);

$insdata = array();

if ($data != NULL) {

for ($i = 0; $i < sizeof(reset($data)); $i += 1) {

array\_push($insdata, array(

'id\_laporan' => $id\_laporan,

'id\_opd' => $data['id\_opd'][$i],

'uji\_kompetensi' => $data['uji\_kompetensi'][$i],

'sop' => $data['sop'][$i],

'tata\_naskah\_dinas' => $data['tata\_naskah\_dinas'][$i],

'pakaian\_dinas' => $data['pakaian\_dinas'][$i],

'jam\_kerja' => $data['jam\_kerja'][$i],

'jml\_nilai' => $data['jml\_nilai'][$i],

'ket' => $data['ket'][$i]

));

}

}

$this->db->trans\_begin

();

if ($table == 'tatalaksana\_opd') {

$this->db->delete('tatalaksana\_opd', "id\_laporan = $id\_laporan");

activity\_log();

if ($data != NULL) {

$this->db->insert\_batch('tatalaksana\_opd', $insdata);

activity\_log();

}

}

**Kode Sumber 28 Model Tatalaksana**

$this->db->trans\_complete();

}

public function delete\_data($id\_laporan)

{

$this->db->trans\_begin();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('tatalaksana');

activity\_log();

$this->db->where('id\_laporan', $id\_laporan);

$this->db->delete('laporan');

activity\_log();

$this->db->trans\_complete();

}

}

## Implementasi Controller

* + 1. **Controller Admin**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Admin extends CI\_Controller

{

public $data;

public function \_\_construct()

{

parent::\_\_construct();

$this->sess\_ver();

$this->data['user'] = array(

'id\_opd' => $this->session->tempdata('id\_opd'),

'nama\_opd' => $this->session->tempdata('nama\_opd')

);

$this->data['title'] = "E-Laporan " . strtoupper($this->session->tempdata('nama\_opd'));

}

public function index()

{

redirect('database/opdtipe', 'refresh');

}

public function f($formfilename = NULL, $id\_opd = NULL)

{

if ($formfilename == NULL || !file\_exists(APPPATH . "views/formtemplate/$formfilename.php")) {

redirect('admin', 'refresh');

return;

}

$this->data['contents'] = APPPATH . "views/formtemplate/$formfilename.php";

// data to send to view for option

if ($formfilename == 'registrationform') {

$this->load->model('opd\_model', 'opd');

$data = $this->opd->gets();

$this->data['opsi\_opd'] = array();

foreach ($data as $row) {

if ($row['id\_opd'] == '1') // id\_opd admin

continue;

array\_push($this->data['opsi\_opd'], $row);

}

$this->data['tipe\_opsi'] = 'register';

} elseif ($formfilename == 'resetpasswordform') {

$this->load->model('user\_model', 'user');

$data = $this->user->gets();

$this->data['opsi\_user'] = array();

foreach ($data as $row) {

array\_push($this->data['opsi\_user'], $row);

}

$this->data['tipe\_opsi'] = 'reset';

} elseif ($formfilename == 'tipelaporanopdform') {

// load list OPD

$this->load->model('opd\_model', 'opd');

$data = $this->opd->gets();

$this->data['opsi\_opd'] = array();

foreach ($data as $row) {

if ($row['id\_opd'] == '1') continue;

array\_push($this->data['opsi\_opd'], $row);

}

// load tipe laporan yang nyala

if ($id\_opd == NULL) {

$id\_opd = $data[0]['id\_opd'];

}

$this->data['selected'] = $id\_opd;

$this->load->model('tipelaporanopd\_model', 'tlo');

$data = $this->tlo->get\_tipelaporan\_by\_idopd($id\_opd);

$this->data['tipelaporan\_on'] = array();

foreach ($data as $d) {

$this->data['tipelaporan\_on'][$d['id\_tipe']] = "on";

}

// load seluruh tipelaporan

$this->load->model('tipelaporan\_model', 'tipelaporan');

$data = $this->tipelaporan->gets();

// var\_dump($data); die();

$this->data['opsi\_tipelaporan'] = array();

foreach ($data as $row) {

array\_push($this->data['opsi\_tipelaporan'], $row);

}

$this->data['tipe\_opsi'] = 'tipelaporan';

} else {

redirect('admin', 'refresh');

return;

}

$this->load->view('template/index\_admin', array('data' => $this->data));

}

public function add\_user()

{

$data = [

'username' => htmlspecialchars($this->input->post('username', true)),

'password' => password\_hash($this->input->post('password', true), PASSWORD\_BCRYPT),

'id\_opd' => $this->input->post('id\_opd', true)

];

$this->load->model('user\_model');

$this->user\_model->insert($data);

}

public function update\_tipelaporan\_per\_opd()

{

if ($this->session->tempdata('id\_opd') != '1') {

redirect('auth', 'refresh');

}

$this->load->model('tipelaporanopd\_model', 'tipelaporanopd');

$this->tipelaporanopd->update\_tipelaporan\_per\_opd($this->input->post('id\_opd'), $this->input->post('id\_tipe'));

}

public function reset\_password($id = NULL)

{

$this->load->model('user\_model');

if ($id) {

$this->set\_flash('RESET PASSWORD', 'Reset password berhasil', 'notice');

$this->user\_model->reset\_password($id, '123456');

redirect('database/user', 'refresh');

}

$this->user\_model->reset\_password($this->input->post('id'), $this->input->post('password'));

}

public function submit()

{

$flag\_option = $this->input->post('tipe\_opsi'); // tipe\_opsi from form hidden attribute

if ($flag\_option == 'register') {

$this->add\_user();

$this->set\_flash('Tambah User', 'BERHASIL', 'success');

redirect('database/user', 'refresh');

} elseif ($flag\_option == 'reset') {

$this->reset\_password();

redirect('admin/f/resetpasswordform', 'refresh');

} elseif ($flag\_option == 'tipelaporan') {

$this->update\_tipelaporan\_per\_opd();

$this->set\_flash('Update Akses OPD', 'Update Akses OPD BERHASIL', 'success');

redirect('database/opdtipe', 'refresh');

}

}

public function querylog($filename = NULL)

{

$this->data['title'] = 'Query Logger';

if ($filename) {

$this->data['contents'] = APPPATH . "logs/" . $filename;

} else {

$this->data['contents'] = APPPATH . "views/admin/querylogger.php";

}

**Kode Sumber 29 Controller Admin**

$this->load->view('template/index\_admin', ['data' => $this->data]);

}

public function set\_flash($title, $string, $type)

{

$this->session->set\_flashdata('message', '<body onload="new PNotify({

title: \'' . $title . '\',

text: \'' . $string . '\',

type: \'' . $type . '\',

styling: \'bootstrap3\'

});">

</body>');

}

private function sess\_ver()

{

if ($this->session->tempdata() == NULL or $this->session->tempdata('id\_opd') != '1') {

redirect('auth');

}

}

}

* + 1. **Controller Apicheck**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Apicheck extends CI\_Controller

{

public function index()

{

echo '<a href="' . base\_url('apicheck/sipp/tahun/2020') . '">SIPP</a><br>';

echo '<a href="' . base\_url('apicheck/sikd\_serapan/2019/2') . '">SIKD SERAPAN</a><br>';

echo '<a href="' . base\_url('apicheck/sikd\_lra/2019/1.18.01') . '">SIKD LRA</a><br>';

echo '<a href="' . base\_url('apicheck/ekin\_pegawai') . '">MASTER AUDITOR</a><br>';

echo '<a href="' . base\_url('apicheck/ekin\_pegawai/16') . '">AUDITOR BY ID</a><br>';

echo '<a href="' . base\_url('apicheck/cek\_kode\_opd\_sipp') . '">CEK KODE EBUD OPD DARI SIPP</a><br>';

}

public function cek\_kode\_opd\_sipp()

{

$key = 'tahun';

$val = '2019';

$this->load->model('api\_sipp\_model');

$data = $this->api\_sipp\_model->get\_api($key, $val);

$data = $data['data'];

print\_r($data);

}

public function sipp($key = 'tahun', $val = '2020')

{

$this->load->model('api\_sipp\_model');

$data = $this->api\_sipp\_model->get\_api($key, $val);

$json = json\_encode($data, JSON\_PRETTY\_PRINT);

**Kode Sumber 30 Controller Apicheck**

printf('Data<pre>%s</pre>', $json);

}

public function sikd\_serapan($tahun = '2019', $tw = '2')

{

$this->load->model('api\_sikd\_model');

$data = $this->api\_sikd\_model->get\_serapan($tahun, $tw);

$json = json\_encode($data['arr'], JSON\_PRETTY\_PRINT);

printf('Data<pre>%s</pre>', $json);

}

public function sikd\_lra($tahun, $kode\_skpd)

{

$this->load->model('api\_sikd\_model');

$data = $this->api\_sikd\_model->get\_lra($tahun, $kode\_skpd);

$json = json\_encode($data, JSON\_PRETTY\_PRINT);

printf('Data<pre>%s</pre>', $json);

}

public function test($id\_laporan, $tahun, $id\_opd) {

$this->load->model('opd\_model', 'opd');

$this->load->model('api\_sikd\_model', 'p');

$kode\_skpd = $this->opd->get\_idemov($id\_opd);

if ($kode\_skpd == NULL) return NULL;

$fet = $this->p->get\_lra($tahun, $kode\_skpd);

printf("<pre>%s</pre>", json\_encode($fet, JSON\_PRETTY\_PRINT));

die();

$ret = array();

if ($fet != NULL) {

foreach ($fet as $f) {

array\_push($ret, array(

'kode\_kegiatan' => $id\_laporan . '-' . $f['KODE\_PROGRAM'] . '-' . $f['KODE\_KEGIATAN'],

'realisasi' => $f['REALISASI']

));

}

return $ret;

} else return NULL;

}

public function ekin\_pegawai($kode\_ekin) {

if ($kode\_ekin) {

$this->load->model('api\_ekin\_model');

$data = $this->api\_ekin\_model->get\_peg($kode\_ekin);

$json = json\_encode($data, JSON\_PRETTY\_PRINT);

printf('Data<pre>%s</pre>', $json);

} else {

$this->load->model('api\_ekin\_model');

$data = $this->api\_ekin\_model->get\_api();

$json = json\_encode($data, JSON\_PRETTY\_PRINT);

printf("Data<pre>%s</pre>", $json);

}

}

}

* + 1. **Controller Auth**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Auth extends CI\_Controller

{

private function isAuth()

{

if ($this->session->tempdata() != NULL) {

$this->\_roledirect($this->session->tempdata('id\_opd'));

}

}

public function index()

{

$this->isAuth();

$this->form\_validation->set\_rules('username', 'Username', 'trim|required');

$this->form\_validation->set\_rules('password', 'Password', 'trim|required');

if ($this->form\_validation->run() == TRUE) {

// Jika Validasi sesuai jalankan login

$this->\_login();

} else {

// Jika Validasi salah

$data['title'] = 'Login';

$this->load->view('template/header\_admin', array('data' => $data));

$this->load->view('auth/login\_view');

$this->load->view('template/footer\_admin');

}

}

public function logout()

{

$this->session->sess\_destroy();

redirect('auth');

}

private function \_login()

{

$this->load->model('user\_model');

$username = $this->input->post('username');

$password = $this->input->post('password');

$user = $this->user\_model->get\_login($username);

if ($user) {

if (password\_verify($password, $user['password'])) {

$this->\_sessionbuilder($user);

$this->user\_model->last\_login($this->session->tempdata('id'));

$this->\_roledirect($user['id\_opd']);

} else {

redirect('auth', 'refresh');

}

} else {

redirect('auth', 'refresh');

}

}

private function \_sessionbuilder($user) {

**Kode Sumber 31 Controller Auth**

$this->load->model('opd\_model', 'opd');

$data = [

'id' => $user['id'],

'username' => $user['username'],

'id\_opd' => $user['id\_opd'],

'nama\_opd' => $this->opd->get\_namaopd($user['id\_opd'])

];

$this->session->set\_tempdata($data, NULL, 14400);

}

private function \_roledirect($id\_opd)

{

if ($id\_opd == '1') {

redirect('admin', 'refresh');

} elseif ($id\_opd != '1') {

redirect('opd', 'refresh');

}

}

}

* + 1. **Controller Database**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Database extends CI\_Controller

{

public function \_\_construct()

{

parent::\_\_construct();

$this->load->model('dbf\_model');

$this->data['user'] = array(

'id\_opd' => $this->session->tempdata('id\_opd'),

'nama\_opd' => $this->session->tempdata('nama\_opd')

);

}

public function opd()

{

$this->data['title'] = 'OPD';

$this->data['rawdata'] = $this->dbf\_model->get\_opd();

$this->data['contents'] = APPPATH . "views/admin/database/db\_opd.php";

$this->load->view('template/index\_admin', ['data' => $this->data]);

}

public function user()

{

$this->data['title'] = 'User';

$this->data['rawdata'] = $this->dbf\_model->get\_user();

$this->data['contents'] = APPPATH . "views/admin/database/db\_user.php";

$this->load->view('template/index\_admin', ['data' => $this->data]);

}

public function deluser($id)

{

**Kode Sumber 32 Controller Database**

$this->load->model('user\_model');

$this->user\_model->delete($id);

$this->session->set\_flashdata('message', '<div class="alert alert-danger" id="success-alert" role="alert"><strong>Data telah dihapus!!!</strong></div>');

redirect('database/user', 'refresh');

}

public function opdtipe()

{

$this->data['title'] = 'Tipe Laporan OPD';

$this->data['rawdata'] = $this->dbf\_model->get\_opd\_tipe();

$this->load->model('tipelaporan\_model');

$this->data['pil'] = $this->tipelaporan\_model->gets();

$this->data['contents'] = APPPATH . "views/admin/database/db\_opdtipe.php";

$this->load->view('template/index\_admin', ['data' => $this->data]);

}

public function updateopdtipe($id\_opd)

{

$this->data['contents'] = APPPATH . "views/admin/database/form/form\_akses\_laporan.php";

$this->load->model('dbf\_model');

$this->data['rawdata'] = $this->dbf\_model->get\_tipe\_laporan();

$this->data['selected'] = $this->dbf\_model->get\_tipelaporan\_per\_opd($id\_opd);

$this->data['id\_opd'] = $id\_opd;

$nama = $this->dbf\_model->get\_opd($id\_opd);

$nama = $nama['nama\_opd'];

$this->data['title'] = 'Edit Akses Laporan ' . $nama;

$this->load->view('template/index\_admin', ['data' => $this->data]);

}

public function cudlog()

{

$this->data['title'] = 'CUD Log';

if ($this->input->post()) {

$day = $this->input->post('day');

$this->data['rawdata'] = $this->dbf\_model->get\_log($day);

} else {

$this->data['rawdata'] = $this->dbf\_model->get\_log();

}

$this->data['contents'] = APPPATH . "views/admin/query/q\_cudlog.php";

$this->load->view('template/index\_admin', ['data' => $this->data]);

}

}

* + 1. **Controller Opd**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

class Opd extends CI\_Controller

{

private $tipelaporan;

public $data;

public function \_\_construct() {

parent::\_\_construct();

$this->sess\_ver();

$this->get\_tipelaporan($this->session->tempdata('id\_opd'));

$this->data['sidebar'] = $this->tipelaporan;

$this->data['user'] = array(

'id\_opd' => $this->session->tempdata('id\_opd'),

'nama\_opd' => $this->session->tempdata('nama\_opd')

);

$this->data['title'] = "E-Laporan " . strtoupper($this->session->tempdata('nama\_opd'));

}

public function get\_tipelaporan($id\_opd)

{

$this->load->model('tipelaporanopd\_model', 'tso');

$this->tipelaporan = $this->tso->get\_tipelaporan\_by\_idopd($id\_opd);

}

public function get\_list\_tipelaporan()

{

return $this->tipelaporan;

}

public function index($err = NULL) {

if ($err != NULL) {

$this->data['error'] = $err;

}

if ($this->tipelaporan == NULL) {

$this->data['contents'] = APPPATH . "views/opd/riwayatlaporan.php";

$this->load->view('template/index\_opd', array('data' => $this->data));

} else {

redirect('opd/f/' . $this->tipelaporan[0]['kode\_tipe'], 'refresh');

}

}

public function f($formname, $page\_number = 10) // display laporan by kode\_tipe

{

$this->form\_ver($formname);

$this->load->model('tipelaporan\_model', 'ts');

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['kode\_tipe'] = $formname;

$from = $this->input->post('from');

$to = $this->input->post('to');

$this->data['list\_laporan'] = $this->get\_datalaporan\_by\_kodetipe($formname, $page\_number, $from, $to);

$this->data['contents'] = APPPATH . "views/opd/riwayatlaporan.php";

$this->load->view('template/index\_opd', array('data' => $this->data));

}

public function get\_datalaporan\_by\_kodetipe($kode\_tipe, $page\_number, $from, $to)

{

$this->load->model('laporan\_model', 'laporan');

return $this->laporan->get\_laporan\_by\_kodetipe($kode\_tipe, $page\_number, $this->session->tempdata('id\_opd'), 20, $from, $to);

}

public function c($formname) // create new laporan by kode\_tipe

{

$this->form\_ver($formname);

$tgl = $this->input->post('tgl');

// buat di tabel laporan

$this->load->model('tipelaporan\_model', 'tl');

$id\_tipe = $this->tl->get\_idtipe\_by\_kodetipe($formname);

// buat di table nama\_surat

$initdata = $this->input->post();

$this->load->model("tipelaporan/" . str\_replace('\_', '', $formname) . "\_model", 'p');

$newid = $this->p->init\_insert(

$this->session->tempdata('id\_opd'),

array('id\_opd' => $this->session->tempdata('id\_opd'), 'id\_tipe' => $id\_tipe),

$initdata

);

if ($newid == NULL) {

redirect("opd/e/$formname", 'refresh');

}

$this->set\_flash('Tambah', 'Berhasil menambah ' . ucwords(str\_replace('\_', ' ', $formname)), 'success');

redirect("opd/e/$formname/$newid", "refresh"); }

public function e($formname, $id\_laporan) // edit existing data

{

if ($this->input->post() != NULL) { // update data

$this->load->model("tipelaporan/" . str\_replace('\_', '', $formname) . "\_model", 'lp');

$this->lp->update\_data($id\_laporan, $this->input->post());

redirect("opd/thanks/$formname/$id\_laporan", 'refresh');

}

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->load->model('laporan\_model', 'laporan');

if (

$formname == "ikm" || $formname == "sotk" || $formname == 'tatalaksana' ||

$formname == "pelayanan\_publik" || $formname == "monitoring\_kelembagaan" ||

$formname == "rekap\_tender" || $formname == "jadwal\_pelaksanaan"

) {

$this->load->model('opd\_model', 'opd');

$this->data['opsi\_opd'] = $this->opd->gets();

unset($this->data['opsi\_opd'][0]);

}

if ($formname == "rekap\_tender") {

$this->load->model('api\_inaproc\_rup\_model', 'ina');

$this->data['opsi\_paket\_kerja'] = $this->ina->get\_rup();

}

if ($formname == "rekap\_pokja") {

$this->load->model('tipelaporan/rekaptender\_model', 'rt');

$this->data['opsi\_paket\_kerja'] = $this->rt->get\_paket\_kerja();

}

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['formname'] = $formname;

$this->data['contents'] = APPPATH . "views/formtemplate/$formname.php";

$this->data['id\_laporan'] = $id\_laporan;

$this->load->view('template/index\_opd', array('data' => $this->data));

}

public function d($formname, $id\_laporan)

{

$this->load->model("tipelaporan/" . str\_replace('\_', '', $formname) . "\_model", 'd');

$this->d->delete\_data($id\_laporan);

$this->set\_flash('Hapus', 'Berhasil menghapus ' . ucwords(str\_replace('\_', ' ', $formname)) . ' dengan id ' . $id\_laporan, 'warning');

redirect("opd/f/$formname", 'refresh');

}

public function thanks($formname, $id\_laporan)

{

$this->set\_flash('Update', 'Berhasil mengubah ' . ucwords(str\_replace('\_', ' ', $formname)) . ' dengan id ' . $id\_laporan, 'info');

redirect("opd/e/$formname/$id\_laporan", 'refresh');

}

public function p($formname, $id\_laporan) // print existing data

{

$this->load->model('laporan\_model', 'laporan');

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->load->view('print/' . $formname, array('data' => $this->data));

}

public function pdf($formname, $id\_laporan)

{

$this->load->model('laporan\_model', 'laporan');

$this->load->library('pdf');

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->pdf->setPaper('A3', 'landscape');

$this->pdf->filename = $this->data['nama\_laporan'];

$this->pdf->load\_view('print/' . $formname, array('data' => $this->data));

}

public function carilaporan($page\_number)

{

$this->load->model('laporan\_model', 'laporan');

$this->data['list\_laporan'] = $this->laporan->search($this->input->get());

$this->data['contents'] = APPPATH . "views/opd/riwayatlaporan.php";

$this->load->view('template/index\_opd', array('data' => $this->data));

}

public function submit()

{

$id\_laporan = NULL;

if ($this->input->post('id\_laporan') != NULL) {

$id\_laporan = $this->input->post('id\_laporan');

}

$data = $this->input->post();

if ($id\_laporan == NULL) { // new data

$this->laporan->add\_data($data);

} else { // editted data

$this->laporan->update\_data($id, $data);

}

}

private function sess\_ver()

{

**Kode Sumber 33 Controller Opd**

if ($this->session->tempdata() == NULL or $this->session->tempdata('id\_opd') == '1') {

redirect('auth');

}

}

private function form\_ver($formname)

{

$this->load->model('tipelaporanopd\_model', 'a');

$temp = $this->a->get\_tipelaporan\_by\_idopd($this->session->tempdata('id\_opd'));

$flag = false;

if ($temp != NULL) {

foreach ($temp as $d) {

if ($d['kode\_tipe'] == $formname) {

$flag = true;

break;

}

}

}

if (!$flag) {

return $this->index();

}

}

public function set\_flash($title, $string, $type)

{

$this->session->set\_flashdata('message', '<body onload="new PNotify({

title: \'' . $title . '\',

text: \'' . $string . '\',

type: \'' . $type . '\',

styling: \'bootstrap3\'

});">

</body>');

}

}

* + 1. **Controller Ikm**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class Ikm extends CI\_Controller

{

public $data;

public $spreadsheet;

public function \_\_construct()

{

parent::\_\_construct();

$this->load->model('laporan\_model', 'laporan');

$this->spreadsheet = new Spreadsheet();

}

public function index($formname, $id\_laporan)

{

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->export($formname, $id\_laporan);

$this->download();

var\_dump("test"); die();

}

public function style($stylereq)

{

switch ($stylereq) {

case 'foo\_css':

$style = [];

break;

case 'allborder':

$style = [

'borders' => [

'allBorders' => [

'borderStyle' => \PhpOffice\PhpSpreadsheet\Style\Border::BORDER\_THIN

]

]

];

break;

default:

# code...

break;

}

return $style;

}

public function export($formname, $id\_laporan)

{

// ini bikin sheetnya

$this->spreadsheet->setActiveSheetIndex(0)->setTitle($this->data['nama\_laporan']);

// ini atur sheet

$sheet = $this->spreadsheet->getActiveSheet();

// ini atur width

$sheet->getDefaultRowDimension()->setRowHeight(-1);

$sheet->getColumnDimension('A')->setAutoSize(true);

$sheet->getColumnDimension('B')->setWidth(60);

$sheet->getColumnDimension('C')->setWidth(20);

$sheet->getColumnDimension('D')->setWidth(20);

// ini stylenya

$sheet->getStyle('1:3')->getAlignment()->setHorizontal('center');

$sheet->getStyle('5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C:D')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A:D')->getAlignment()->setVertical('center');

$sheet->getStyle('A:D')->getAlignment()->setWrapText(true);

$sheet->getStyle('A1:D5')->getFont()->setBold(true);

// ini atur header

$sheet->setCellValue('A1', 'NILAI INDEKS KEPUASAAN MASYARAKAT (IKM)')

->mergeCells('A1:D1');

**Kode Sumber 34 Controller Ikm**

$sheet->setCellValue('A2', 'TAHUN ' . date('Y', strtotime($this->data['fetch']['ikm']['tgl'])) . ' DI LINGKUNGAN PEMERINTAH KOTA MADIUN')

->mergeCells('A2:D2');

$sheet->setCellValue('A3', $this->data['nama\_opd'])

->mergeCells('A3:D3');

// ini tablenya

// th numrow 5

$sheet->setCellValue('A5', 'No.');

$sheet->setCellValue('B5', 'PERANGKAT DAERAH');

$sheet->setCellValue('C5', 'NILAI TAHUN ' . date('Y', strtotime($this->data['fetch']['ikm']['tgl'])));

$sheet->setCellValue('D5', 'PREDIKAT');

// td numrow 6

$numrow = 6;

$nomor = 0;

foreach ($this->data['fetch']['ikmopd'] as $ikm) {

$nomor++;

$sheet->setCellValue('A' . $numrow, $nomor);

$sheet->setCellValue('B' . $numrow, ucwords($ikm['nama\_opd']));

$sheet->setCellValue('C' . $numrow, $ikm['nilai']);

$sheet->setCellValue('D' . $numrow, $ikm['predikat']);

$numrow++;

}

// ini style tablenya

$sheet->getPageSetup()->setFitToWidth(1);

$sheet->getPageSetup()->setFitToHeight(0);

$sheet->setShowGridlines(true);

$sheet->getStyle('A5:D' . ($numrow - 1))->applyFromArray($this->style('allborder'));

}

public function download()

{

$file = new Xlsx($this->spreadsheet);

header('Content-Type: application/vnd.ms-excel');

header('Content-Disposition: attachment;filename="' . $this->data['nama\_laporan'] . '.xlsx"');

header('Cache-Control: max-age=0');

$file->save('php://output'); // download file

}

}

* + 1. **Controller Jadwal\_pelaksanaan**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class jadwal\_pelaksanaan extends CI\_Controller

{

public $data;

public $spreadsheet;

public function \_\_construct()

{

parent::\_\_construct();

$this->load->model('laporan\_model', 'laporan');

$this->spreadsheet = new Spreadsheet();

}

public function index($formname, $id\_laporan)

{

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->export($formname, $id\_laporan);

$this->download();

}

public function style($stylereq)

{

switch ($stylereq) {

case 'foo\_css':

$style = [];

break;

case 'allborder':

$style = [

'borders' => [

'allBorders' => [

'borderStyle' => \PhpOffice\PhpSpreadsheet\Style\Border::BORDER\_THIN

]

]

];

break;

default:

# code...

break;

}

return $style;

}

public function export($formname, $id\_laporan)

{

// ini bikin sheetnya

$this->spreadsheet->setActiveSheetIndex(0)->setTitle($this->data['nama\_laporan']);

// ini atur sheet

$sheet = $this->spreadsheet->getActiveSheet();

// ini atur width

$sheet->getDefaultRowDimension()->setRowHeight(-1);

$sheet->getColumnDimension('A')->setAutoSize(true);

$sheet->getColumnDimension('B')->setWidth(60);

$sheet->getColumnDimension('C')->setWidth(20);

$sheet->getColumnDimension('D')->setWidth(10);

$sheet->getColumnDimension('E')->setWidth(10);

$sheet->getColumnDimension('F')->setWidth(20);

$sheet->getColumnDimension('G')->setWidth(15);

$sheet->getColumnDimension('H')->setWidth(20);

$sheet->getColumnDimension('I')->setWidth(20);

// ini stylenya

$sheet->getStyle('1:6')->getAlignment()->setHorizontal('center'); // STYLE HEADER

$sheet->getStyle('A')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C:I')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A:I')->getAlignment()->setVertical('center');

$sheet->getStyle('A:I')->getAlignment()->setWrapText(true);

$sheet->getStyle('A1:I6')->getFont()->setBold(true);

// ini atur header

$sheet->setCellValue('A1', 'JADWAL RENCANA PELAKSANAAN PROGRAM KERJA PENGAWASAN TAHUNAN KOTA MADIUN')

->mergeCells('A1:I1');

$sheet->setCellValue('A2', 'TAHUN ' . date('Y', strtotime($this->data['fetch']['jp']['tgl'])))

->mergeCells('A2:I2');

$sheet->setCellValue('A3', $this->data['nama\_opd'])

->mergeCells('A3:I3');

// ini tablenya

// th numrow 5

$sheet->setCellValue('A5', 'No.');

$sheet->setCellValue('B5', 'Objek Pengawasan');

$sheet->setCellValue('C5', 'Jenis Pengawasan');

$sheet->setCellValue('D5', 'RMP');

$sheet->setCellValue('E5', 'RPL');

$sheet->setCellValue('F5', 'Nama Auditor');

$sheet->setCellValue('G5', 'Jabatan');

$sheet->setCellValue('H5', 'Output LHP');

$sheet->setCellValue('I5', 'Hari Pengawasan');

$sheet->setCellValue('A6', '1');

$sheet->setCellValue('B6', '2');

$sheet->setCellValue('C6', '3');

$sheet->setCellValue('D6', '4');

$sheet->setCellValue('E6', '5');

$sheet->setCellValue('F6', '6');

$sheet->setCellValue('G6', '7');

$sheet->setCellValue('H6', '8');

$sheet->setCellValue('I6', '9');

// td numrow 7

$numrow = 7;

$counter = 0;

foreach ($this->data['fetch']['jpopd'] as $jpopd) {

$jpopd\_rowspan = (sizeof($this->data['fetch']['adata'][$jpopd['id\_jadwal\_pelaksanaan\_opd']]) - 1);

$counter += 1;

$sheet->setCellValue('A' . $numrow, $counter)

->mergeCells('A' . $numrow . ':A' . ($numrow + $jpopd\_rowspan));

$sheet->setCellValue('B' . $numrow, ucwords($jpopd['nama\_opd']))

->mergeCells('B' . $numrow . ':B' . ($numrow + $jpopd\_rowspan));

$sheet->setCellValue('C' . $numrow, $jpopd['jenis\_pengawasan'])

->mergeCells('C' . $numrow . ':C' . ($numrow + $jpopd\_rowspan));

$sheet->setCellValue('D' . $numrow, $jpopd['rmp'])

->mergeCells('D' . $numrow . ':D' . ($numrow + $jpopd\_rowspan));

$sheet->setCellValue('E' . $numrow, $jpopd['rpl'])

->mergeCells('E' . $numrow . ':E' . ($numrow + $jpopd\_rowspan));

$sheet->setCellValue('F' . $numrow, $this->data['fetch']['adata'][$jpopd['id\_jadwal\_pelaksanaan\_opd']][0]['nama\_auditor']);

$sheet->setCellValue('G' . $numrow, $this->data['fetch']['adata'][$jpopd['id\_jadwal\_pelaksanaan\_opd']][0]['jabatan']);

**Kode Sumber 35 Controller Jadwal\_pelaksanaan**

$sheet->setCellValue('H' . $numrow, $jpopd['output\_lhp'])

->mergeCells('H' . $numrow . ':H' . ($numrow + $jpopd\_rowspan));

$sheet->setCellValue('I' . $numrow, $jpopd['hari\_pengawasan'])

->mergeCells('I' . $numrow . ':I' . ($numrow + $jpopd\_rowspan));

$numrow++;

$flag = TRUE;

foreach ($this->data['fetch']['adata'][$jpopd['id\_jadwal\_pelaksanaan\_opd']] as $jpopd) {

if ($flag) {

$flag = FALSE;

continue;

}

$sheet->setCellValue('F' . $numrow, $jpopd['nama\_auditor']);

$sheet->setCellValue('G' . $numrow, $jpopd['jabatan']);

$numrow++;

}

}

// ini style tablenya

$sheet->getStyle('A5:I' . ($numrow - 1))->applyFromArray($this->style('allborder'));

// ini style page setup

$sheet->getPageSetup()->setOrientation('landscape');

$sheet->getPageSetup()->setFitToWidth(1);

$sheet->getPageSetup()->setFitToHeight(0);

$sheet->setShowGridlines(true);

}

public function download()

{

$file = new Xlsx($this->spreadsheet);

header('Content-Type: application/vnd.ms-excel');

header('Content-Disposition: attachment;filename="' . $this->data['nama\_laporan'] . '.xlsx"');

header('Cache-Control: max-age=0');

$file->save('php://output'); // download file

}

}

* + 1. **Controller Laporan\_kinerja\_triwulan**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class laporan\_kinerja\_triwulan extends CI\_Controller

{

public $data;

public $spreadsheet;

public function \_\_construct()

{

parent::\_\_construct();

$this->load->model('laporan\_model', 'laporan');

$this->spreadsheet = new Spreadsheet();

}

public function index($formname, $id\_laporan)

{

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->export($formname, $id\_laporan);

$this->download();

}

public function style($stylereq)

{

switch ($stylereq) {

case 'foo\_css':

$style = [];

break;

case 'allborder':

$style = [

'borders' => [

'allBorders' => [

'borderStyle' => \PhpOffice\PhpSpreadsheet\Style\Border::BORDER\_THIN

]

]

];

break;

default:

# code...

break;

}

return $style;

}

public function export($formname, $id\_laporan)

{

// ini bikin sheetnya

$this->spreadsheet->setActiveSheetIndex(0)->setTitle($this->data['nama\_laporan']);

// ini atur sheet

$sheet = $this->spreadsheet->getActiveSheet();

// ini atur width

$sheet->getDefaultRowDimension()->setRowHeight(-1);

$sheet->getColumnDimension('A')->setAutoSize(true);

$sheet->getColumnDimension('B')->setWidth(20);

$sheet->getColumnDimension('C')->setWidth(20);

$sheet->getColumnDimension('D')->setWidth(20);

$sheet->getColumnDimension('E')->setWidth(20);

$sheet->getColumnDimension('F')->setWidth(20);

$sheet->getColumnDimension('G')->setWidth(20);

$sheet->getColumnDimension('H')->setWidth(20);

// ini stylenya

$sheet->getStyle('A:H')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A:H')->getAlignment()->setVertical('center');

$sheet->getStyle('A:H')->getAlignment()->setWrapText(true);

$sheet->getStyle('A1:H8')->getFont()->setBold(true);

// ini atur header

$sheet->setCellValue('A1', 'LAPORAN KINERJA')

->mergeCells('A1:H1');

$sheet->setCellValue('A2', 'TRIWULAN')

->mergeCells('A2:H2');

$sheet->setCellValue('A3', $this->data['nama\_opd'])

->mergeCells('A3:H3');

$sheet->setCellValue('A4', 'TAHUN ' . date('Y', strtotime($this->data['fetch']['lkt']['tgl'])))

->mergeCells('A4:H4');

// ini tablenya

// th numrow 4

$sheet->setCellValue('A6', 'No.')->mergeCells('A6:A7');

$sheet->setCellValue('B6', 'Sasaran Kegiatan')->mergeCells('B6:E6');

$sheet->setCellValue('F6', 'Program')->mergeCells('F6:F7');

$sheet->setCellValue('G6', 'Anggaran')->mergeCells('G6:G7');

$sheet->setCellValue('H6', 'Realisasi')->mergeCells('H6:H7');

$sheet->setCellValue('B7', 'Uraian');

$sheet->setCellValue('C7', 'Indikator Kinerja');

$sheet->setCellValue('D7', 'Target');

$sheet->setCellValue('E7', 'Realisasi');

$sheet->setCellValue('A8', '1');

$sheet->setCellValue('B8', '2');

$sheet->setCellValue('C8', '3');

$sheet->setCellValue('D8', '4');

$sheet->setCellValue('E8', '5');

$sheet->setCellValue('F8', '6');

$sheet->setCellValue('G8', '7');

$sheet->setCellValue('H8', '8');

// td numrow 6

$numrow = 9;

// data

$counter = 0;

foreach ($this->data['fetch']['lktdetail'] as $lktdetail) {

$counter += 1;

$sheet->setCellValue('A' . $numrow, $counter);

$sheet->setCellValue('B' . $numrow, $lktdetail['uraian']);

$sheet->setCellValue('C' . $numrow, $lktdetail['indikator\_kinerja']);

$sheet->setCellValue('D' . $numrow, $lktdetail['target']);

$sheet->setCellValue('E' . $numrow, $lktdetail['realisasi\_target']); lktdetail['capaian\_realisasi\_keuangan']);

$numrow++;

}

$sheet->setCellValue('F' . $numrow, $lktdetail['program']);

$sheet->setCellValue('G' . $numrow, $lktdetail['anggaran']);

$sheet->setCellValue('H' . $numrow, $lktdetail['capaian\_realisasi\_keuangan']);

$numrow++;

}

// ini style tablenya

$sheet->getPageSetup()->setOrientation('landscape');

$sheet->getPageSetup()->setFitToWidth(1);

$sheet->getPageSetup()->setFitToHeight(0);

$sheet->setShowGridlines(true);

$sheet->getStyle('A6:H' . ($numrow - 1))->applyFromArray($this->style('allborder'));

}

public function download()

{

**Kode Sumber 36 Controller Laporan\_kinerja\_triwulan**

$file = new Xlsx($this->spreadsheet);

header('Content-Type: application/vnd.ms-excel');

header('Content-Disposition: attachment;filename="' . $this->data['nama\_laporan'] . '.xlsx"');

header('Cache-Control: max-age=0');

$file->save('php://output'); // download file

}

}

* + 1. **Controller Laporan\_rb\_area\_perubahan**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class laporan\_rb\_area\_perubahan extends CI\_Controller

{

public $data;

public $spreadsheet;

public function \_\_construct()

{

parent::\_\_construct();

$this->load->model('laporan\_model', 'laporan');

$this->spreadsheet = new Spreadsheet();

}

public function index($formname, $id\_laporan)

{

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->export($formname, $id\_laporan);

$this->download();

}

public function style($stylereq)

{

switch ($stylereq) {

case 'foo\_css':

$style = [];

break;

case 'allborder':

$style = [

'borders' => [

'allBorders' => [

'borderStyle' => \PhpOffice\PhpSpreadsheet\Style\Border::BORDER\_THIN

]

]

];

break;

case 'font-red':

$style = [

'font' => [

'color' => [

'argb' => \PhpOffice\PhpSpreadsheet\Style\Color::COLOR\_RED

]

]

];

break;

default:

# code...

break;

}

return $style;

}

public function export($formname, $id\_laporan)

{

// ini bikin sheetnya

$this->spreadsheet->setActiveSheetIndex(0)->setTitle($this->data['nama\_laporan']);

// ---- INI SHEET PERTAMA ----

// ini atur sheet

$sheet = $this->spreadsheet->getActiveSheet();

// ini atur width

$sheet->getDefaultRowDimension()->setRowHeight(-1);

$sheet->getColumnDimension('A')->setAutoSize(true);

$sheet->getColumnDimension('B')->setWidth(20);

$sheet->getColumnDimension('C')->setWidth(20);

$sheet->getColumnDimension('D')->setWidth(20);

$sheet->getColumnDimension('E')->setWidth(20);

$sheet->getColumnDimension('F')->setWidth(20);

$sheet->getColumnDimension('G')->setWidth(20);

$sheet->getColumnDimension('H')->setWidth(20);

$sheet->getColumnDimension('I')->setWidth(20);

$sheet->getColumnDimension('J')->setWidth(20);

$sheet->getColumnDimension('K')->setWidth(20);

// ini stylenya

$sheet->getStyle('1')->getAlignment()->setHorizontal('center');

$sheet->getStyle('1')->getAlignment()->setVertical('center');

$sheet->getStyle('A3:A5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A3:A5')->getAlignment()->setVertical('center');

$sheet->getStyle('A6:D6')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C3:C5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C3:C5')->getAlignment()->setVertical('center');

$sheet->getStyle('D3:D5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('D3:D5')->getAlignment()->setVertical('center');

$sheet->getStyle('E:K')->getAlignment()->setHorizontal('center');

$sheet->getStyle('B3:B5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('B3:B5')->getAlignment()->setVertical('center');

$sheet->getStyle('E:K')->getAlignment()->setVertical('center');

$sheet->getStyle('A:L')->getAlignment()->setWrapText(true);

$sheet->getStyle('A1')->getFont()->setBold(true);

$sheet->getStyle('A3:K6')->getFont()->setBold(true);

// ini atur header

$sheet->setCellValue('A1', 'Laporan Rencana Aksi Reformasi Birokrasi Pemerintah Daerah Tahun ' . date('Y', strtotime($this->data['fetch']['rb']['tgl'])) . ' pada Delapan Area Perubahan (per 30 Desember ' . date('Y', strtotime($this->data['fetch']['rb']['tgl'])) . ' )')

->mergeCells('A1:L1');

// ini tablenya

// th numrow 5

$sheet->setCellValue('A3', 'NO')

->mergeCells('A3:A5');

$sheet->setCellValue('B3', 'AREA PERUBAHAN')

->mergeCells('B3:B5');

$sheet->setCellValue('C3', 'PROGRAM')

->mergeCells('C3:C5');

$sheet->setCellValue('D3', 'KEGIATAN')

->mergeCells('D3:D5');

$sheet->setCellValue('E3', 'PERIODE PELAKSANA')

->mergeCells('E3:F3');

$sheet->setCellValue('G3', 'ANGGARAN (Rp.000.000)')

->mergeCells('G3:H3');

$sheet->setCellValue('I3', 'STATUS CAPAIAN (V)')

->mergeCells('I3:J3');

$sheet->setCellValue('K3', 'ALASAN TIDAK TERCAPAI')

->mergeCells('K3:K5');

// th numrow 6

$sheet->setCellValue('E4', 'TARGET')

->mergeCells('E4:E5');

$sheet->setCellValue('F4', 'REALISASI')

->mergeCells('F4:F5');

$sheet->setCellValue('G4', 'TARGET')

->mergeCells('G4:G5');

$sheet->setCellValue('H4', 'REALISASI')

->mergeCells('H4:H5');

$sheet->setCellValue('I4', 'TERCAPAI')

->mergeCells('I4:I5');

$sheet->setCellValue('J4', 'TIDAK TERCAPAI')

->mergeCells('J4:J5');

$sheet->setCellValue('A6', '1');

$sheet->setCellValue('B6', '2');

$sheet->setCellValue('C6', '3');

$sheet->setCellValue('D6', '4');

$sheet->setCellValue('E6', '5');

$sheet->setCellValue('F6', '6');

$sheet->setCellValue('G6', '7');

$sheet->setCellValue('H6', '8');

$sheet->setCellValue('I6', '9');

$sheet->setCellValue('J6', '10');

$sheet->setCellValue('K6', '11');

// // td numrow 7

$numrow = 7;

//for buat data

$counter = 0;

foreach ($this->data['fetch']['rbap'] as $rbap) {

$rowspan\_rincian = 0;

foreach ($this->data['fetch']['rbaps'][$rbap['id\_rb\_area\_perubahan']] as $rbaps) {

$rowspan\_rincian += sizeof($this->data['fetch']['rbapk'][$rbaps['id\_rb\_area\_perubahan\_program']]);

}

$rowspan\_rincian -= 1;

if ($rowspan\_rincian < 0) {

$rowspan\_rincian = 0;

}

$counter += 1;

$flag = FALSE;

$sheet->setCellValue('A' . $numrow, $counter)->mergeCells('A' . $numrow . ':A' . ($numrow + $rowspan\_rincian));

$sheet->setCellValue('B' . $numrow, $rbap['rincian'])->mergeCells('B' . $numrow . ':B' . ($numrow + $rowspan\_rincian));

foreach ($this->data['fetch']['rbaps'][$rbap['id\_rb\_area\_perubahan']] as $rbaps) {

$rowspan\_program = sizeof($this->data['fetch']['rbapk'][$rbaps['id\_rb\_area\_perubahan\_program']]) - 1;

if ($rowspan\_program < 0) {

$rowspan\_program = 0;

}

$sheet->setCellValue('C' . $numrow, $rbaps['nama\_program'])->mergeCells('C' . $numrow . ':C' . ($numrow + $rowspan\_program));

$flag2 = FALSE;

$rbapk = reset($this->data['fetch']['rbapk'][$rbaps['id\_rb\_area\_perubahan\_program']]);

$sheet->setCellValue('D' . $numrow, $rbapk['nama\_kegiatan']);

$sheet->setCellValue('E' . $numrow, $rbapk['target\_waktu']);

$sheet->setCellValue('F' . $numrow, $rbapk['realisasi\_waktu']);

$sheet->setCellValue('G' . $numrow, $rbapk['target\_anggaran']);

$sheet->setCellValue('H' . $numrow, $rbapk['realisasi\_anggaran']);

if ($rbapk['capaian'] == '0') {

$sheet->setCellValue('J' . $numrow, 'V');

} else {

$sheet->setCellValue('I' . $numrow, 'V');

}

$sheet->setCellValue('K' . $numrow, $rbapk['ket']);

foreach ($this->data['fetch']['rbapk'][$rbaps['id\_rb\_area\_perubahan\_program']] as $rbapk) {

if ($flag2) {

$sheet->setCellValue('D' . $numrow, $rbapk['nama\_kegiatan']);

$sheet->setCellValue('E' . $numrow, $rbapk['target\_waktu']);

$sheet->setCellValue('F' . $numrow, $rbapk['realisasi\_waktu']);

$sheet->setCellValue('G' . $numrow, $rbapk['target\_anggaran']);

$sheet->setCellValue('H' . $numrow, $rbapk['realisasi\_anggaran']);

if ($rbapk['capaian'] == '0') {

$sheet->setCellValue('J' . $numrow, 'V');

} else {

$sheet->setCellValue('I' . $numrow, 'V');

}

$sheet->setCellValue('K' . $numrow, $rbapk['ket']);

} else {

$flag2 = TRUE;

}

$numrow++;

}

}

}

// end ambil data

**Kode Sumber 37 Controller Laporan\_rb\_area\_perubahan**

// ini style tablenya

$sheet->getPageSetup()->setFitToWidth(1);

$sheet->getPageSetup()->setFitToHeight(0);

$sheet->setShowGridlines(true);

$sheet->getStyle('A3:K' . ($numrow - 1))->applyFromArray($this->style('allborder'));

}

public function download()

{

$file = new Xlsx($this->spreadsheet);

header('Content-Type: application/vnd.ms-excel');

header('Content-Disposition: attachment;filename="' . $this->data['nama\_laporan'] . '.xlsx"');

header('Cache-Control: max-age=0');

$file->save('php://output'); // download file

}

}

* + 1. **Controller Laporan\_rb\_fokus**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class laporan\_rb\_fokus extends CI\_Controller

{

public $data;

public $spreadsheet;

public function \_\_construct()

{

parent::\_\_construct();

$this->load->model('laporan\_model', 'laporan');

$this->spreadsheet = new Spreadsheet();

}

public function index($formname, $id\_laporan)

{

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->export($formname, $id\_laporan);

$this->download();

}

public function style($stylereq)

{

switch ($stylereq) {

case 'foo\_css':

$style = [];

break;

case 'allborder':

$style = [

'borders' => [

'allBorders' => [

'borderStyle' => \PhpOffice\PhpSpreadsheet\Style\Border::BORDER\_THIN

]

]

];

break;

case 'font-red':

$style = [

'font' => [

'color' => [

'argb' => \PhpOffice\PhpSpreadsheet\Style\Color::COLOR\_RED

]

]

];

break;

default:

# code...

break;

}

return $style;

}

public function export($formname, $id\_laporan)

{

// ini bikin sheetnya

$this->spreadsheet->setActiveSheetIndex(0)->setTitle($this->data['nama\_laporan']);

// ---- INI SHEET PERTAMA ----

// ini atur sheet

$sheet = $this->spreadsheet->getActiveSheet();

// ini atur width

$sheet->getDefaultRowDimension()->setRowHeight(-1);

$sheet->getColumnDimension('A')->setAutoSize(true);

$sheet->getColumnDimension('B')->setWidth(20);

$sheet->getColumnDimension('C')->setWidth(20);

$sheet->getColumnDimension('D')->setWidth(20);

$sheet->getColumnDimension('E')->setWidth(20);

$sheet->getColumnDimension('F')->setWidth(20);

$sheet->getColumnDimension('G')->setWidth(20);

$sheet->getColumnDimension('H')->setWidth(20);

$sheet->getColumnDimension('I')->setWidth(20);

$sheet->getColumnDimension('J')->setWidth(20);

$sheet->getColumnDimension('K')->setWidth(20);

$sheet->getColumnDimension('L')->setWidth(20);

$sheet->getColumnDimension('M')->setWidth(20);

$sheet->getColumnDimension('N')->setWidth(20);

$sheet->getColumnDimension('O')->setWidth(20);

// ini stylenya

$sheet->getStyle('1')->getAlignment()->setHorizontal('center');

$sheet->getStyle('1')->getAlignment()->setVertical('center');

$sheet->getStyle('A3:A5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A3:A5')->getAlignment()->setVertical('center');

$sheet->getStyle('A6:D6')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C3:C5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C3:C5')->getAlignment()->setVertical('center');

$sheet->getStyle('D3:D5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('D3:D5')->getAlignment()->setVertical('center');

$sheet->getStyle('E:O')->getAlignment()->setHorizontal('center');

$sheet->getStyle('B3:B5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('B3:B5')->getAlignment()->setVertical('center');

$sheet->getStyle('E:O')->getAlignment()->setVertical('center');

$sheet->getStyle('A:O')->getAlignment()->setWrapText(true);

$sheet->getStyle('A1')->getFont()->setBold(true);

$sheet->getStyle('A3:O6')->getFont()->setBold(true);

// ini atur header

$sheet->setCellValue('A1', 'Laporan Capaian Rencana Aksi Reformasi Birokrasi Pemerintah Daerah Kota Madiun ' . date('Y', strtotime($this->data['fetch']['rb']['tgl'])) . ' pada Prioritas yang Terkait dengan Peningkatan Kualitas Pelayanan Fokus Pelayanan Sektor Tertentu (per 30 Desember ' . date('Y', strtotime($this->data['fetch']['rb']['tgl'])) . ' )')

->mergeCells('A1:O1');

// ini tablenya

// th numrow 5

$sheet->setCellValue('A3', 'NO')

->mergeCells('A3:A5');

$sheet->setCellValue('B3', 'SEKTOR PELAYANAN')

->mergeCells('B3:B5');

$sheet->setCellValue('C3', 'SASARAN')

->mergeCells('C3:C5');

$sheet->setCellValue('D3', 'PROGRAM')

->mergeCells('D3:D5');

$sheet->setCellValue('E3', 'KEGIATAN')

->mergeCells('E3:E5');

$sheet->setCellValue('F3', 'INDIKATOR')

->mergeCells('F3:F5');

$sheet->setCellValue('G3', 'OUTPUT')

->mergeCells('G3:H3');

$sheet->setCellValue('I3', 'BULAN KE-')

->mergeCells('I3:J3');

$sheet->setCellValue('K3', 'ANGGARAN')

->mergeCells('K3:L3');

$sheet->setCellValue('M3', 'STATUS CAPAIAN (V)')

->mergeCells('M3:N3');

$sheet->setCellValue('O3', 'KETERANGAN')

->mergeCells('O3:O5');

// th numrow 6

$sheet->setCellValue('G4', 'TARGET')

->mergeCells('G4:G5');

$sheet->setCellValue('H4', 'REALISASI')

->mergeCells('H4:H5');

$sheet->setCellValue('I4', 'TARGET')

->mergeCells('I4:I5');

$sheet->setCellValue('J4', 'REALISASI')

->mergeCells('J4:J5');

$sheet->setCellValue('K4', 'TARGET')

->mergeCells('K4:K5');

$sheet->setCellValue('L4', 'REALISASI')

->mergeCells('L4:L5');

$sheet->setCellValue('M4', 'TERCAPAI')

->mergeCells('M4:M5');

$sheet->setCellValue('N4', 'TIDAK TERCAPAI')

->mergeCells('N4:N5');

$sheet->setCellValue('A6', '1');

$sheet->setCellValue('B6', '2');

$sheet->setCellValue('C6', '3');

$sheet->setCellValue('D6', '4');

$sheet->setCellValue('E6', '5');

$sheet->setCellValue('F6', '6');

$sheet->setCellValue('G6', '7');

$sheet->setCellValue('H6', '8');

$sheet->setCellValue('I6', '9');

$sheet->setCellValue('J6', '10');

$sheet->setCellValue('K6', '11');

$sheet->setCellValue('L6', '12');

$sheet->setCellValue('M6', '13');

$sheet->setCellValue('N6', '14');

$sheet->setCellValue('O6', '15');

// // td numrow 7

$numrow = 7;

//for buat data

$counter = 0;

foreach ($this->data['fetch']['rbf'] as $rbf) {

$rowspan\_rincian = 0;

foreach ($this->data['fetch']['rbfs'][$rbf['id\_rb\_fokus']] as $rbfs) {

$rowspan\_rincian += sizeof($this->data['fetch']['rbfk'][$rbfs['id\_rb\_fokus\_sasaran']]);

}

$rowspan\_rincian -= 1;

if ($rowspan\_rincian < 0) {

$rowspan\_rincian = 0;

}

$counter += 1;

$flag = FALSE;

$sheet->setCellValue('A' . $numrow, $counter)->mergeCells('A' . $numrow . ':A' . ($numrow + $rowspan\_rincian));

$sheet->setCellValue('B' . $numrow, $rbf['rincian'])->mergeCells('B' . $numrow . ':B' . ($numrow + $rowspan\_rincian));

foreach ($this->data['fetch']['rbfs'][$rbf['id\_rb\_fokus']] as $rbfs) {

$rowspan\_program = sizeof($this->data['fetch']['rbfk'][$rbfs['id\_rb\_fokus\_sasaran']]) - 1;

if ($rowspan\_program < 0) {

$rowspan\_program = 0;

}

$sheet->setCellValue('C' . $numrow, $rbfs['sasaran'])->mergeCells('C' . $numrow . ':C' . ($numrow + $rowspan\_program));

$sheet->setCellValue('D' . $numrow, $rbfs['nama\_program'])->mergeCells('D' . $numrow . ':D' . ($numrow + $rowspan\_program));

$flag2 = FALSE;

$rbapk = reset($this->data['fetch']['rbfk'][$rbfs['id\_rb\_fokus\_sasaran']]);

$sheet->setCellValue('E' . $numrow, $rbapk['nama\_kegiatan']);

$sheet->setCellValue('F' . $numrow, $rbapk['indikator']);

$sheet->setCellValue('G' . $numrow, $rbapk['target\_output']);

$sheet->setCellValue('H' . $numrow, $rbapk['realisasi\_output']);

$sheet->setCellValue('I' . $numrow, $rbapk['target\_waktu']);

$sheet->setCellValue('J' . $numrow, $rbapk['realisasi\_waktu']);

**Kode Sumber 38 Controller Laporan\_rb\_fokus**

$sheet->setCellValue('K' . $numrow, $rbapk['target\_anggaran']);

$sheet->setCellValue('L' . $numrow, $rbapk['realisasi\_anggaran']);

if ($rbapk['capaian'] == '0') {

$sheet->setCellValue('N' . $numrow, 'V');

} else {

$sheet->setCellValue('M' . $numrow, 'V');

}

$sheet->setCellValue('O' . $numrow, $rbapk['ket']);

foreach ($this->data['fetch']['rbfk'][$rbfs['id\_rb\_fokus\_sasaran']] as $rbfk) {

if ($flag2) {

$sheet->setCellValue('E' . $numrow, $rbapk['nama\_kegiatan']);

$sheet->setCellValue('F' . $numrow, $rbapk['indikator']);

$sheet->setCellValue('G' . $numrow, $rbapk['target\_output']);

$sheet->setCellValue('H' . $numrow, $rbapk['realisasi\_output']);

$sheet->setCellValue('I' . $numrow, $rbapk['target\_waktu']);

$sheet->setCellValue('J' . $numrow, $rbapk['realisasi\_waktu']);

$sheet->setCellValue('K' . $numrow, $rbapk['target\_anggaran']);

$sheet->setCellValue('L' . $numrow, $rbapk['realisasi\_anggaran']);

if ($rbapk['capaian'] == '0') {

$sheet->setCellValue('N' . $numrow, 'V');

} else {

$sheet->setCellValue('M' . $numrow, 'V');

}

$sheet->setCellValue('O' . $numrow, $rbapk['ket']);

} else {

$flag2 = TRUE;

}

$numrow++;

}

}

}

// end ambil data

// ini style tablenya

$sheet->getPageSetup()->setFitToWidth(1);

$sheet->getPageSetup()->setFitToHeight(0);

$sheet->setShowGridlines(true);

$sheet->getStyle('A3:O' . ($numrow - 1))->applyFromArray($this->style('allborder'));

}

public function download()

{

$file = new Xlsx($this->spreadsheet);

header('Content-Type: application/vnd.ms-excel');

header('Content-Disposition: attachment;filename="' . $this->data['nama\_laporan'] . '.xlsx"');

header('Cache-Control: max-age=0');

$file->save('php://output'); // download file

}

}

* + 1. **Controller Laporan\_rb\_prioritas**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class laporan\_rb\_prioritas extends CI\_Controller

{

public $data;

public $spreadsheet;

public function \_\_construct()

{

parent::\_\_construct();

$this->load->model('laporan\_model', 'laporan');

$this->spreadsheet = new Spreadsheet();

}

public function index($formname, $id\_laporan)

{

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->export($formname, $id\_laporan);

$this->download();

}

public function style($stylereq)

{

switch ($stylereq) {

case 'foo\_css'

$style = [];

break;

case 'allborder':

$style = [

'borders' => [

'allBorders' => [

'borderStyle' => \PhpOffice\PhpSpreadsheet\Style\Border::BORDER\_THIN

]

]

];

break;

case 'font-red':

$style = [

'font' => [

'color' => [

'argb' => \PhpOffice\PhpSpreadsheet\Style\Color::COLOR\_RED

]

]

];

: break;

default:

# code...

break;

}

return $style;

}

public function export($formname, $id\_laporan)

{

// ini bikin sheetnya

$this->spreadsheet->setActiveSheetIndex(0)->setTitle($this->data['nama\_laporan']);

// ---- INI SHEET PERTAMA ----

// ini atur sheet

$sheet = $this->spreadsheet->getActiveSheet();

// ini atur width

$sheet->getDefaultRowDimension()->setRowHeight(-1);

$sheet->getColumnDimension('A')->setAutoSize(true);

$sheet->getColumnDimension('B')->setWidth(25);

$sheet->getColumnDimension('C')->setWidth(20);

$sheet->getColumnDimension('D')->setWidth(20);

$sheet->getColumnDimension('E')->setWidth(20);

$sheet->getColumnDimension('F')->setWidth(20);

$sheet->getColumnDimension('G')->setWidth(20);

$sheet->getColumnDimension('H')->setWidth(20);

$sheet->getColumnDimension('I')->setWidth(20);

$sheet->getColumnDimension('J')->setWidth(20);

$sheet->getColumnDimension('K')->setWidth(20);

$sheet->getColumnDimension('L')->setWidth(20);

$sheet->getColumnDimension('M')->setWidth(20);

$sheet->getColumnDimension('N')->setWidth(20);

$sheet->getColumnDimension('O')->setWidth(20);

// ini stylenya

$sheet->getStyle('1')->getAlignment()->setHorizontal('center');

$sheet->getStyle('1')->getAlignment()->setVertical('center');

$sheet->getStyle('A3:A5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A3:A5')->getAlignment()->setVertical('center');

$sheet->getStyle('A6:D6')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C3:C5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C3:C5')->getAlignment()->setVertical('center');

$sheet->getStyle('D3:D5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('D3:D5')->getAlignment()->setVertical('center');

$sheet->getStyle('E:O')->getAlignment()->setHorizontal('center');

$sheet->getStyle('B3:B5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('B3:B5')->getAlignment()->setVertical('center');

$sheet->getStyle('E:O')->getAlignment()->setVertical('center');

$sheet->getStyle('A:O')->getAlignment()->setWrapText(true);

$sheet->getStyle('A1')->getFont()->setBold(true);

$sheet->getStyle('A3:O6')->getFont()->setBold(true);

// ini atur header

$sheet->setCellValue('A1', 'Laporan Capaian Rencana Aksi Reformasi Birokrasi Pemerintah Daerah Kota Madiun ' . date('Y', strtotime($this->data['fetch']['rb']['tgl'])) . ' pada Prioritas yang Harus Terus Dipelihara (per 30 Desember ' . date('Y', strtotime($this->data['fetch']['rb']['tgl'])) . ' )')

->mergeCells('A1:O1');

// ini tablenya

// th numrow 5

$sheet->setCellValue('A3', 'NO')

->mergeCells('A3:A5');

$sheet->setCellValue('B3', 'PRIORITAS TERPELIHARA')

->mergeCells('B3:B5');

$sheet->setCellValue('C3', 'SASARAN')

->mergeCells('C3:C5');

$sheet->setCellValue('D3', 'PROGRAM')

->mergeCells('D3:D5');

$sheet->setCellValue('E3', 'KEGIATAN')

->mergeCells('E3:E5');

$sheet->setCellValue('F3', 'INDIKATOR')

->mergeCells('F3:F5');

$sheet->setCellValue('G3', 'OUTPUT')

->mergeCells('G3:H3');

$sheet->setCellValue('I3', 'BULAN KE-')

->mergeCells('I3:J3');

$sheet->setCellValue('K3', 'ANGGARAN')

->mergeCells('K3:L3');

$sheet->setCellValue('M3', 'STATUS CAPAIAN (V)')

->mergeCells('M3:N3');

$sheet->setCellValue('O3', 'KETERANGAN')

->mergeCells('O3:O5');

// th numrow 6

$sheet->setCellValue('G4', 'TARGET')

->mergeCells('G4:G5');

$sheet->setCellValue('H4', 'REALISASI')

->mergeCells('H4:H5');

$sheet->setCellValue('I4', 'TARGET')

->mergeCells('I4:I5');

$sheet->setCellValue('J4', 'REALISASI')

->mergeCells('J4:J5');

$sheet->setCellValue('K4', 'TARGET')

->mergeCells('K4:K5');

$sheet->setCellValue('L4', 'REALISASI')

->mergeCells('L4:L5');

$sheet->setCellValue('M4', 'TERCAPAI')

->mergeCells('M4:M5');

$sheet->setCellValue('N4', 'TIDAK TERCAPAI')

->mergeCells('N4:N5');

$sheet->setCellValue('A6', '1');

$sheet->setCellValue('B6', '2');

$sheet->setCellValue('C6', '3');

$sheet->setCellValue('D6', '4');

$sheet->setCellValue('E6', '5');

$sheet->setCellValue('F6', '6');

$sheet->setCellValue('G6', '7');

$sheet->setCellValue('H6', '8');

$sheet->setCellValue('I6', '9');

$sheet->setCellValue('J6', '10');

$sheet->setCellValue('K6', '11');

$sheet->setCellValue('L6', '12');

$sheet->setCellValue('M6', '13');

$sheet->setCellValue('N6', '14');

$sheet->setCellValue('O6', '15');

// // td numrow 7

$numrow = 7;

//for buat data

$counter = 0;

foreach ($this->data['fetch']['rbp'] as $rbp) {

$rowspan\_rincian = 0;

foreach ($this->data['fetch']['rbps'][$rbp['id\_rb\_prioritas']] as $rbps) {

$rowspan\_rincian += sizeof($this->data['fetch']['rbpk'][$rbps['id\_rb\_prioritas\_sasaran']]);

}

$rowspan\_rincian -= 1;

if ($rowspan\_rincian < 0) {

$rowspan\_rincian = 0;

}

$counter += 1;

$flag = FALSE;

$sheet->setCellValue('A' . $numrow, $counter)->mergeCells('A' . $numrow . ':A' . ($numrow + $rowspan\_rincian));

$sheet->setCellValue('B' . $numrow, $rbp['rincian'])->mergeCells('B' . $numrow . ':B' . ($numrow + $rowspan\_rincian));

foreach ($this->data['fetch']['rbps'][$rbp['id\_rb\_prioritas']] as $rbps) {

$rowspan\_program = sizeof($this->data['fetch']['rbpk'][$rbps['id\_rb\_prioritas\_sasaran']]) - 1;

if ($rowspan\_program < 0) {

$rowspan\_program = 0;

}

$sheet->setCellValue('C' . $numrow, $rbps['sasaran'])->mergeCells('C' . $numrow . ':C' . ($numrow + $rowspan\_program));

$sheet->setCellValue('D' . $numrow, $rbps['nama\_program'])->mergeCells('D' . $numrow . ':D' . ($numrow + $rowspan\_program));

$flag2 = FALSE;

$rbpk = reset($this->data['fetch']['rbpk'][$rbps['id\_rb\_prioritas\_sasaran']]);

$sheet->setCellValue('E' . $numrow, $rbpk['nama\_kegiatan']);

$sheet->setCellValue('F' . $numrow, $rbpk['indikator']);

$sheet->setCellValue('G' . $numrow, $rbpk['target\_output']);

$sheet->setCellValue('H' . $numrow, $rbpk['realisasi\_output']);

$sheet->setCellValue('I' . $numrow, $rbpk['target\_waktu']);

$sheet->setCellValue('J' . $numrow, $rbpk['realisasi\_waktu']);

$sheet->setCellValue('K' . $numrow, $rbpk['target\_anggaran']);

$sheet->setCellValue('L' . $numrow, $rbpk['realisasi\_anggaran']);

if ($rbpk['capaian'] == '0') {

$sheet->setCellValue('N' . $numrow, 'V');

} else {

$sheet->setCellValue('M' . $numrow, 'V');

}

$sheet->setCellValue('O' . $numrow, $rbpk['ket']);

foreach ($this->data['fetch']['rbpk'][$rbps['id\_rb\_prioritas\_sasaran']] as $rbpk) {

if ($flag2) {

$sheet->setCellValue('E' . $numrow, $rbpk['nama\_kegiatan']);

$sheet->setCellValue('F' . $numrow, $rbpk['indikator']);

$sheet->setCellValue('G' . $numrow, $rbpk['target\_output']);

$sheet->setCellValue('H' . $numrow, $rbpk['realisasi\_output']);

$sheet->setCellValue('I' . $numrow, $rbpk['target\_waktu']);

$sheet->setCellValue('J' . $numrow, $rbpk['realisasi\_waktu']);

$sheet->setCellValue('K' . $numrow, $rbpk['target\_anggaran']);

$sheet->setCellValue('L' . $numrow, $rbpk['realisasi\_anggaran']);

**Kode Sumber 39 Controller Laporan\_rb\_prioritas**

if ($rbpk['capaian'] == '0') {

$sheet->setCellValue('N' . $numrow, 'V');

} else {

$sheet->setCellValue('M' . $numrow, 'V');

}

$sheet->setCellValue('O' . $numrow, $rbpk['ket']);

} else {

$flag2 = TRUE;

}

$numrow++;

}

}

}

// end ambil data

// ini style tablenya

$sheet->getPageSetup()->setFitToWidth(1);

$sheet->getPageSetup()->setFitToHeight(0);

$sheet->setShowGridlines(true);

$sheet->getStyle('A3:O' . ($numrow - 1))->applyFromArray($this->style('allborder'));

}

public function download()

{

$file = new Xlsx($this->spreadsheet);

header('Content-Type: application/vnd.ms-excel');

header('Content-Disposition: attachment;filename="' . $this->data['nama\_laporan'] . '.xlsx"');

header('Cache-Control: max-age=0');

$file->save('php://output'); // download file

}

}

* + 1. **Controller Laporan\_rb\_quick\_wins**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class laporan\_rb\_quick\_wins extends CI\_Controller

{

public $data;

public $spreadsheet;

public function \_\_construct()

{

parent::\_\_construct();

$this->load->model('laporan\_model', 'laporan');

$this->spreadsheet = new Spreadsheet();

}

public function index($formname, $id\_laporan)

{

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->export($formname, $id\_laporan);

$this->download();

}

public function style($stylereq)

{

switch ($stylereq) {

case 'foo\_css':

$style = [];

break;

case 'allborder':

$style = [

'borders' => [

'allBorders' => [

'borderStyle' => \PhpOffice\PhpSpreadsheet\Style\Border::BORDER\_THIN

]

]

];

break;

case 'font-red':

$style = [

'font' => [

'color' => [

'argb' => \PhpOffice\PhpSpreadsheet\Style\Color::COLOR\_RED

]

]

];

break;

default:

# code...

break;

}

return $style;

}

public function export($formname, $id\_laporan)

{

// ini bikin sheetnya

$this->spreadsheet->setActiveSheetIndex(0)->setTitle($this->data['nama\_laporan']);

// ---- INI SHEET PERTAMA ----

// ini atur sheet

$sheet = $this->spreadsheet->getActiveSheet();

// ini atur width

$sheet->getDefaultRowDimension()->setRowHeight(-1);

$sheet->getColumnDimension('A')->setAutoSize(true);

$sheet->getColumnDimension('B')->setWidth(25);

$sheet->getColumnDimension('C')->setWidth(20);

$sheet->getColumnDimension('D')->setWidth(20);

$sheet->getColumnDimension('E')->setWidth(20);

$sheet->getColumnDimension('F')->setWidth(20);

$sheet->getColumnDimension('G')->setWidth(20);

$sheet->getColumnDimension('H')->setWidth(20);

$sheet->getColumnDimension('I')->setWidth(20);

$sheet->getColumnDimension('J')->setWidth(20);

$sheet->getColumnDimension('K')->setWidth(20);

$sheet->getColumnDimension('L')->setWidth(20);

$sheet->getColumnDimension('M')->setWidth(20);

$sheet->getColumnDimension('N')->setWidth(20);

$sheet->getColumnDimension('O')->setWidth(20);

// ini stylenya

$sheet->getStyle('1')->getAlignment()->setHorizontal('center');

$sheet->getStyle('1')->getAlignment()->setVertical('center');

$sheet->getStyle('A3:A5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A3:A5')->getAlignment()->setVertical('center');

$sheet->getStyle('A6:D6')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C3:C5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C3:C5')->getAlignment()->setVertical('center');

$sheet->getStyle('D3:D5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('D3:D5')->getAlignment()->setVertical('center');

$sheet->getStyle('E:O')->getAlignment()->setHorizontal('center');

$sheet->getStyle('B3:B5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('B3:B5')->getAlignment()->setVertical('center');

$sheet->getStyle('E:O')->getAlignment()->setVertical('center');

$sheet->getStyle('A:O')->getAlignment()->setWrapText(true);

$sheet->getStyle('A1')->getFont()->setBold(true);

$sheet->getStyle('A3:O6')->getFont()->setBold(true);

// ini atur header

$sheet->setCellValue('A1', 'Laporan Capaian Rencana Aksi Reformasi Birokrasi Pemerintah Daerah Kota Madiun per 30 Desember ' . date('Y', strtotime($this->data['fetch']['rb']['tgl'])) . ' pada Prioritas yang Terkait dengan Peningkatan Kualitas Pelayanan Fokus Pelayanan Quick Wins')

->mergeCells('A1:O1');

// ini tablenya

// th numrow 5

$sheet->setCellValue('A3', 'NO')

->mergeCells('A3:A5');

$sheet->setCellValue('B3', 'QUICK WINS')

->mergeCells('B3:B5');

$sheet->setCellValue('C3', 'SASARAN')

->mergeCells('C3:C5');

$sheet->setCellValue('D3', 'PROGRAM')

->mergeCells('D3:D5');

$sheet->setCellValue('E3', 'KEGIATAN')

->mergeCells('E3:E5');

$sheet->setCellValue('F3', 'INDIKATOR')

->mergeCells('F3:F5');

$sheet->setCellValue('G3', 'OUTPUT')

->mergeCells('G3:H3');

$sheet->setCellValue('I3', 'BULAN KE-')

->mergeCells('I3:J3');

$sheet->setCellValue('K3', 'ANGGARAN')

->mergeCells('K3:L3');

$sheet->setCellValue('M3', 'STATUS CAPAIAN (V)')

->mergeCells('M3:N3');

$sheet->setCellValue('O3', 'KETERANGAN')

->mergeCells('O3:O5');

// th numrow 6

$sheet->setCellValue('G4', 'TARGET')

->mergeCells('G4:G5');

$sheet->setCellValue('H4', 'REALISASI')

->mergeCells('H4:H5');

$sheet->setCellValue('I4', 'TARGET')

->mergeCells('I4:I5');

$sheet->setCellValue('J4', 'REALISASI')

->mergeCells('J4:J5');

$sheet->setCellValue('K4', 'TARGET')

->mergeCells('K4:K5');

$sheet->setCellValue('L4', 'REALISASI')

->mergeCells('L4:L5');

$sheet->setCellValue('M4', 'TERCAPAI')

->mergeCells('M4:M5');

$sheet->setCellValue('N4', 'TIDAK TERCAPAI')

->mergeCells('N4:N5');

$sheet->setCellValue('A6', '1');

$sheet->setCellValue('B6', '2');

$sheet->setCellValue('C6', '3');

$sheet->setCellValue('D6', '4');

$sheet->setCellValue('E6', '5');

$sheet->setCellValue('F6', '6');

$sheet->setCellValue('G6', '7');

$sheet->setCellValue('H6', '8');

$sheet->setCellValue('I6', '9');

$sheet->setCellValue('J6', '10');

$sheet->setCellValue('K6', '11');

$sheet->setCellValue('L6', '12');

$sheet->setCellValue('M6', '13');

$sheet->setCellValue('N6', '14');

$sheet->setCellValue('O6', '15');

// // td numrow 7

$numrow = 7;

//for buat data

$counter = 0;

foreach ($this->data['fetch']['rbqw'] as $rbqw) {

$rowspan\_rincian = 0;

foreach ($this->data['fetch']['rbqws'][$rbqw['id\_rb\_quick\_wins']] as $rbqws) {

$rowspan\_rincian += sizeof($this->data['fetch']['rbqwk'][$rbqws['id\_rb\_quick\_wins\_sasaran']]);

}

$rowspan\_rincian -= 1;

if ($rowspan\_rincian < 0) {

$rowspan\_rincian = 0;

}

$counter += 1;

$flag = FALSE;

$sheet->setCellValue('A' . $numrow, $counter)->mergeCells('A' . $numrow . ':A' . ($numrow + $rowspan\_rincian));

$sheet->setCellValue('B' . $numrow, $rbqw['rincian'])->mergeCells('B' . $numrow . ':B' . ($numrow + $rowspan\_rincian));

foreach ($this->data['fetch']['rbqws'][$rbqw['id\_rb\_quick\_wins']] as $rbqws) {

$rowspan\_program = sizeof($this->data['fetch']['rbqwk'][$rbqws['id\_rb\_quick\_wins\_sasaran']]) - 1;

if ($rowspan\_program < 0) {

$rowspan\_program = 0;

}

$sheet->setCellValue('C' . $numrow, $rbqws['sasaran'])->mergeCells('C' . $numrow . ':C' . ($numrow + $rowspan\_program));

$sheet->setCellValue('D' . $numrow, $rbqws['nama\_program'])->mergeCells('D' . $numrow . ':D' . ($numrow + $rowspan\_program));

$flag2 = FALSE;

$rbqwk = reset($this->data['fetch']['rbqwk'][$rbqws['id\_rb\_quick\_wins\_sasaran']]);

$sheet->setCellValue('E' . $numrow, $rbqwk['nama\_kegiatan']);

$sheet->setCellValue('F' . $numrow, $rbqwk['indikator']);

$sheet->setCellValue('G' . $numrow, $rbqwk['target\_output']);

$sheet->setCellValue('H' . $numrow, $rbqwk['realisasi\_output']);

$sheet->setCellValue('I' . $numrow, $rbqwk['target\_waktu']);

$sheet->setCellValue('J' . $numrow, $rbqwk['realisasi\_waktu']);

$sheet->setCellValue('K' . $numrow, $rbqwk['target\_anggaran']);

$sheet->setCellValue('L' . $numrow, $rbqwk['realisasi\_anggaran']);

if ($rbqwk['capaian'] == '0') {

$sheet->setCellValue('N' . $numrow, 'V');

} else {

$sheet->setCellValue('M' . $numrow, 'V');

}

$sheet->setCellValue('O' . $numrow, $rbqwk['ket']);

foreach ($this->data['fetch']['rbqwk'][$rbqws['id\_rb\_quick\_wins\_sasaran']] as $rbqwk) {

if ($flag2) {

$sheet->setCellValue('E' . $numrow, $rbqwk['nama\_kegiatan']);

$sheet->setCellValue('F' . $numrow, $rbqwk['indikator']);

$sheet->setCellValue('G' . $numrow, $rbqwk['target\_output']);

$sheet->setCellValue('H' . $numrow, $rbqwk['realisasi\_output']);

$sheet->setCellValue('I' . $numrow, $rbqwk['target\_waktu']);

$sheet->setCellValue('J' . $numrow, $rbqwk['realisasi\_waktu']);

$sheet->setCellValue('K' . $numrow, $rbqwk['target\_anggaran']);

$sheet->setCellValue('L' . $numrow, $rbqwk['realisasi\_anggaran']);

if ($rbqwk['capaian'] == '0') {

$sheet->setCellValue('N' . $numrow, 'V');

} else {

$sheet->setCellValue('M' . $numrow, 'V');

}

$sheet->setCellValue('O' . $numrow, $rbqwk['ket']);

} else {

$flag2 = TRUE;

}

$numrow++;

}

}

}

// end ambil data

// ini style tablenya

**Kode Sumber 40 Controller Laporan\_rb\_quick\_wins**

$sheet->getPageSetup()->setFitToWidth(1);

$sheet->getPageSetup()->setFitToHeight(0);

$sheet->setShowGridlines(true);

$sheet->getStyle('A3:O' . ($numrow - 1))->applyFromArray($this->style('allborder'));

}

public function download()

{

$file = new Xlsx($this->spreadsheet);

header('Content-Type: application/vnd.ms-excel');

header('Content-Disposition: attachment;filename="' . $this->data['nama\_laporan'] . '.xlsx"');

header('Cache-Control: max-age=0');

$file->save('php://output'); // download file

}

}

* + 1. **Controller Laporan\_rb\_zi\_wbk**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class laporan\_rb\_zi\_wbk extends CI\_Controller

{

public $data;

public $spreadsheet;

public function \_\_construct() {

parent::\_\_construct();

$this->load->model('laporan\_model', 'laporan');

$this->spreadsheet = new Spreadsheet();

}

public function index($formname, $id\_laporan) {

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->export($formname, $id\_laporan);

$this->download();

}

public function style($stylereq) {

switch ($stylereq) {

case 'foo\_css':

$style = [];

break;

case 'allborder':

$style = [

'borders' => [

'allBorders' => [

'borderStyle' => \PhpOffice\PhpSpreadsheet\Style\Border::BORDER\_THIN

]

]

];

break;

case 'font-red':

$style = [

'font' => [

'color' => [

'argb' => \PhpOffice\PhpSpreadsheet\Style\Color::COLOR\_RED

]

]

];

break;

default:

# code...

break;

}

return $style;

}

public function export($formname, $id\_laporan)

{

// ini bikin sheetnya

$this->spreadsheet->setActiveSheetIndex(0)->setTitle($this->data['nama\_laporan']);

// ---- INI SHEET PERTAMA ----

// ini atur sheet

$sheet = $this->spreadsheet->getActiveSheet();

// ini atur width

$sheet->getDefaultRowDimension()->setRowHeight(-1);

$sheet->getColumnDimension('A')->setAutoSize(true);

$sheet->getColumnDimension('B')->setWidth(25);

$sheet->getColumnDimension('C')->setWidth(20);

$sheet->getColumnDimension('D')->setWidth(20);

$sheet->getColumnDimension('E')->setWidth(20);

$sheet->getColumnDimension('F')->setWidth(20);

$sheet->getColumnDimension('G')->setWidth(20);

$sheet->getColumnDimension('H')->setWidth(20);

$sheet->getColumnDimension('I')->setWidth(20);

$sheet->getColumnDimension('J')->setWidth(20);

$sheet->getColumnDimension('K')->setWidth(20);

$sheet->getColumnDimension('L')->setWidth(20);

$sheet->getColumnDimension('M')->setWidth(20);

$sheet->getColumnDimension('N')->setWidth(20);

// ini stylenya

$sheet->getStyle('1')->getAlignment()->setHorizontal('center');

$sheet->getStyle('1')->getAlignment()->setVertical('center');

$sheet->getStyle('A3:A5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A3:A5')->getAlignment()->setVertical('center');

$sheet->getStyle('A6:D6')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C3:C5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C3:C5')->getAlignment()->setVertical('center');

$sheet->getStyle('D3:D5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('D3:D5')->getAlignment()->setVertical('center');

$sheet->getStyle('E:O')->getAlignment()->setHorizontal('center');

$sheet->getStyle('B3:B5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('B3:B5')->getAlignment()->setVertical('center');

$sheet->getStyle('E:N')->getAlignment()->setVertical('center');

$sheet->getStyle('A:N')->getAlignment()->setWrapText(true);

$sheet->getStyle('A1')->getFont()->setBold(true);

$sheet->getStyle('A3:N6')->getFont()->setBold(true);

// ini atur header

$sheet->setCellValue('A1', 'Laporan Capaian Rencana Aksi Reformasi Birokrasi Pemerintah Daerah Kota Madiun per 30 Desember ' . date('Y', strtotime($this->data['fetch']['rb']['tgl'])) . ' pada Prioritas Zona Integrasi Menuju Wilayah Bebas Korupsi (WBK)')

->mergeCells('A1:O1');

// ini tablenya

// th numrow 5

$sheet->setCellValue('A3', 'NO')

->mergeCells('A3:A5');

$sheet->setCellValue('B3', 'SASARAN')

->mergeCells('B3:B5');

$sheet->setCellValue('C3', 'PROGRAM')

->mergeCells('C3:C5');

$sheet->setCellValue('D3', 'KEGIATAN')

->mergeCells('D3:D5');

$sheet->setCellValue('E3', 'INDIKATOR')

->mergeCells('E3:E5');

$sheet->setCellValue('F3', 'OUTPUT')

->mergeCells('F3:G3');

$sheet->setCellValue('H3', 'BULAN KE-')

->mergeCells('H3:I3');

$sheet->setCellValue('J3', 'ANGGARAN')

->mergeCells('J3:K3');

$sheet->setCellValue('L3', 'STATUS CAPAIAN (V)')

->mergeCells('L3:M3');

$sheet->setCellValue('N3', 'KETERANGAN')

->mergeCells('N3:N5');

// th numrow 6

$sheet->setCellValue('F4', 'TARGET')

->mergeCells('F4:F5');

$sheet->setCellValue('G4', 'REALISASI')

->mergeCells('G4:G5');

$sheet->setCellValue('H4', 'TARGET')

->mergeCells('H4:H5');

$sheet->setCellValue('I4', 'REALISASI')

->mergeCells('I4:I5');

$sheet->setCellValue('J4', 'TARGET')

->mergeCells('J4:J5');

$sheet->setCellValue('K4', 'REALISASI')

->mergeCells('K4:K5');

$sheet->setCellValue('L4', 'TERCAPAI')

->mergeCells('L4:L5');

$sheet->setCellValue('M4', 'TIDAK TERCAPAI')

->mergeCells('M4:M5');

$sheet->setCellValue('A6', '1');

$sheet->setCellValue('B6', '2');

$sheet->setCellValue('C6', '3');

$sheet->setCellValue('D6', '4');

$sheet->setCellValue('E6', '5');

$sheet->setCellValue('F6', '6');

$sheet->setCellValue('G6', '7');

$sheet->setCellValue('H6', '8');

$sheet->setCellValue('I6', '9');

$sheet->setCellValue('J6', '10');

$sheet->setCellValue('K6', '11');

$sheet->setCellValue('L6', '12');

$sheet->setCellValue('M6', '13');

$sheet->setCellValue('N6', '14');

// // td numrow 7

$numrow = 7;

//for buat data

$counter = 0;

foreach ($this->data['fetch']['rbziwbks'] as $rbziwbks) {

$counter += 1;

$rowspan\_rincian = sizeof($this->data['fetch']['rbziwbkk'][$rbziwbks['id\_rb\_zi\_wbk\_sasaran']]);

$rowspan\_rincian -= 1;

if ($rowspan\_rincian < 0) {

$rowspan\_rincian = 0;

}

$sheet->setCellValue('A' . $numrow, $counter)->mergeCells('A' . $numrow . ':A' . ($numrow + $rowspan\_rincian));

$sheet->setCellValue('B' . $numrow, $rbziwbks['sasaran'])->mergeCells('B' . $numrow . ':B' . ($numrow + $rowspan\_rincian));

$sheet->setCellValue('C' . $numrow, $rbziwbks['nama\_program'])->mergeCells('C' . $numrow . ':C' . ($numrow + $rowspan\_rincian));

$flag2 = FALSE;

foreach ($this->data['fetch']['rbziwbkk'][$rbziwbks['id\_rb\_zi\_wbk\_sasaran']] as $rbziwbkk) {

if ($flag2) {

$sheet->setCellValue('D' . $numrow, $rbziwbkk['nama\_kegiatan']);

$sheet->setCellValue('E' . $numrow, $rbziwbkk['indikator']);

$sheet->setCellValue('F' . $numrow, $rbziwbkk['target\_output']);

$sheet->setCellValue('G' . $numrow, $rbziwbkk['realisasi\_output']);

$sheet->setCellValue('H' . $numrow, $rbziwbkk['target\_waktu']);

$sheet->setCellValue('I' . $numrow, $rbziwbkk['realisasi\_waktu']);

$sheet->setCellValue('J' . $numrow, $rbziwbkk['target\_anggaran']);

$sheet->setCellValue('K' . $numrow, $rbziwbkk['realisasi\_anggaran']);

if ($rbziwbkk['capaian'] == '0') {

$sheet->setCellValue('M' . $numrow, 'V');

} else {

$sheet->setCellValue('L' . $numrow, 'V');

}

$sheet->setCellValue('N' . $numrow, $rbziwbkk['ket']);

} else {

$flag2 = TRUE;

$sheet->setCellValue('D' . $numrow, $rbziwbkk['nama\_kegiatan']);

$sheet->setCellValue('E' . $numrow, $rbziwbkk['indikator']);

$sheet->setCellValue('F' . $numrow, $rbziwbkk['target\_output']);

$sheet->setCellValue('G' . $numrow, $rbziwbkk['realisasi\_output']);

$sheet->setCellValue('H' . $numrow, $rbziwbkk['target\_waktu']);

$sheet->setCellValue('I' . $numrow, $rbziwbkk['realisasi\_waktu']);

$sheet->setCellValue('J' . $numrow, $rbziwbkk['target\_anggaran']);

$sheet->setCellValue('K' . $numrow, $rbziwbkk['realisasi\_anggaran']);

if ($rbziwbkk['capaian'] == '0') {

**Kode Sumber 41Controller Laporan\_rb\_zi\_wbk**

$sheet->setCellValue('M' . $numrow, 'V');

} else {

$sheet->setCellValue('L' . $numrow, 'V');

}

$sheet->setCellValue('N' . $numrow, $rbziwbkk['ket']);

}

$numrow++;

}

}

// end ambil data

// ini style tablenya

$sheet->getPageSetup()->setFitToWidth(1);

$sheet->getPageSetup()->setFitToHeight(0);

$sheet->setShowGridlines(true);

$sheet->getStyle('A3:N' . ($numrow - 1))->applyFromArray($this->style('allborder'));

}

public function download()

{

$file = new Xlsx($this->spreadsheet);

header('Content-Type: application/vnd.ms-excel');

header('Content-Disposition: attachment;filename="' . $this->data['nama\_laporan'] . '.xlsx"');

header('Cache-Control: max-age=0');

$file->save('php://output'); // download file

}

}

* + 1. **Controller Monitoring\_kelembagaan**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class monitoring\_kelembagaan extends CI\_Controller

{

public $data;

public $spreadsheet;

public $wizard;

public function \_\_construct()

{

parent::\_\_construct();

$this->load->model('laporan\_model', 'laporan');

$this->spreadsheet = new Spreadsheet();

$this->wizard = new PhpOffice\PhpSpreadsheet\Helper\Html;

}

public function index($formname, $id\_laporan)

{

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->export($formname, $id\_laporan);

$this->download();

}

public function style($stylereq)

{

switch ($stylereq) {

case 'foo\_css':

$style = [];

break;

case 'allborder':

$style = [

'borders' => [

'allBorders' => [

'borderStyle' => \PhpOffice\PhpSpreadsheet\Style\Border::BORDER\_THIN

]

]

];

break;

default:

# code...

break;

}

return $style;

}

public function export($formname, $id\_laporan)

{

// ini bikin sheetnya

$this->spreadsheet->setActiveSheetIndex(0)->setTitle($this->data['nama\_laporan']);

// ---- INI SHEET PERTAMA ----

// ini atur sheet

$sheet = $this->spreadsheet->getActiveSheet();

// ini atur width

$sheet->getDefaultRowDimension()->setRowHeight(-1);

$sheet->getColumnDimension('A')->setAutoSize(true);

$sheet->getColumnDimension('B')->setWidth(50);

$sheet->getColumnDimension('C')->setWidth(20);

$sheet->getColumnDimension('D')->setWidth(20);

$sheet->getColumnDimension('E')->setWidth(20);

$sheet->getColumnDimension('F')->setWidth(20);

// ini stylenya

$sheet->getStyle('1:4')->getAlignment()->setHorizontal('center');

$sheet->getStyle('1:4')->getAlignment()->setVertical('center');

$sheet->getStyle('A')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C:F')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A:F')->getAlignment()->setVertical('center');

$sheet->getStyle('A:F')->getAlignment()->setWrapText(true);

$sheet->getStyle('A4:F4')->getAlignment()->setWrapText(true);

$sheet->getStyle('A1:F4')->getFont()->setBold(true);

// ini atur header

$sheet->setCellValue('A1', 'INVENTARISASI DATA PERMASALAHAN KELEMBAGAAN PADA SKPD DI LINGKUNGAN PEMERINTAH MADIUN')

->mergeCells('A1:F1');

**Kode Sumber 42 Controller Monitoring\_kelembagaan**

$sheet->setCellValue('A2', $this->data['nama\_opd'])

->mergeCells('A2:F2');

// ini tablenya

// th numrow 5

$sheet->setCellValue('A4', 'No.');

$sheet->setCellValue('B4', 'Nama Perangkat Daerah');

$sheet->setCellValue('C4', 'Permasalahan Kelembagaan');

$sheet->setCellValue('D4', 'Dasar Hukum/Dasar Pertimbangan');

$sheet->setCellValue('E4', 'Usulan');

$sheet->setCellValue('F4', 'Keterangan');

// td numrow 6

$numrow = 5;

$counter = 0;

foreach ($this->data['fetch']['pk'] as $pk) {

$counter += 1;

$sheet->setCellValue('A' . $numrow, $counter);

$sheet->setCellValue('B' . $numrow, ucwords($pk['nama\_opd']));

$sheet->setCellValue('C' . $numrow, $pk['permasalahan\_kelembagaan']);

$dasar\_hukum = $this->wizard->toRichTextObject($pk['dasar\_hukum']);

$sheet->setCellValue('D' . $numrow, $dasar\_hukum);

$usulan = $this->wizard->toRichTextObject($pk['usulan']);

$sheet->setCellValue('E' . $numrow, $usulan);

$sheet->setCellValue('F' . $numrow, $pk['ket']);

$numrow++;

}

// ini style tablenya

$sheet->getPageSetup()->setFitToWidth(1);

$sheet->getPageSetup()->setFitToHeight(0);

$sheet->setShowGridlines(true);

$sheet->getStyle('A4:F' . ($numrow - 1))->applyFromArray($this->style('allborder'));

}

public function download()

{

$file = new Xlsx($this->spreadsheet);

header('Content-Type: application/vnd.ms-excel');

header('Content-Disposition: attachment;filename="' . $this->data['nama\_laporan'] . '.xlsx"');

header('Cache-Control: max-age=0');

$file->save('php://output'); // download file

}

}

* + 1. **Controller Pelayanan\_publik**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class pelayanan\_publik extends CI\_Controller

{

public $data;

public $spreadsheet;

public function \_\_construct()

{

parent::\_\_construct();

$this->load->model('laporan\_model', 'laporan');

$this->spreadsheet = new Spreadsheet();

}

public function index($formname, $id\_laporan)

{

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->export($formname, $id\_laporan);

$this->download();

}

public function style($stylereq)

{

switch ($stylereq) {

case 'foo\_css':

$style = [];

break;

case 'allborder':

$style = [

'borders' => [

'allBorders' => [

'borderStyle' => \PhpOffice\PhpSpreadsheet\Style\Border::BORDER\_THIN

]

]

];

break;

case 'font-red':

$style = [

'font' => [

'color' => [

'argb' => \PhpOffice\PhpSpreadsheet\Style\Color::COLOR\_RED

]

]

];

break;

default:

# code...

break;

}

return $style;

}

public function export($formname, $id\_laporan)

{

// ini bikin sheetnya

$this->spreadsheet->setActiveSheetIndex(0)->setTitle($this->data['nama\_laporan']);

// ---- INI SHEET PERTAMA ----

// ini atur sheet

$sheet = $this->spreadsheet->getActiveSheet();

// ini atur width

$sheet->getDefaultRowDimension()->setRowHeight(-1);

$sheet->getColumnDimension('A')->setAutoSize(true);

$sheet->getColumnDimension('B')->setWidth(50);

$sheet->getColumnDimension('C')->setWidth(20);

$sheet->getColumnDimension('D')->setWidth(20);

$sheet->getColumnDimension('E')->setWidth(20);

// ini stylenya

$sheet->getStyle('1:4')->getAlignment()->setHorizontal('center');

$sheet->getStyle('1:4')->getAlignment()->setVertical('center');

$sheet->getStyle('A')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C:E')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A:E')->getAlignment()->setVertical('center');

$sheet->getStyle('A:E')->getAlignment()->setWrapText(true);

$sheet->getStyle('A4:E4')->getAlignment()->setWrapText(true);

$sheet->getStyle('A1:E4')->getFont()->setBold(true);

// ini atur header

$sheet->setCellValue('A1', 'LAPORAN HASIL PENGAMATAN PELAYANAN PUBLIK TAHUN ' . date('Y', strtotime($this->data['fetch']['pp']['tgl'])))

->mergeCells('A1:E1');

$sheet->setCellValue('A2', $this->data['nama\_opd'])

->mergeCells('A2:E2');

// ini tablenya

// th numrow 5

$sheet->setCellValue('A4', 'No.');

$sheet->setCellValue('B4', 'Nama Perangkat Daerah');

$sheet->setCellValue('C4', 'Indeks Pelayanan Publik');

$sheet->setCellValue('D4', 'Konversi 100');

$sheet->setCellValue('E4', 'Ket');

// td numrow 6

$numrow = 5;

$counter = 0;

foreach ($this->data['fetch']['ppopd'] as $ppopd) {

$counter += 1;

$ket\_flag = true;

$ket = floatval($ppopd['indeks\_pelayanan\_publik']);

if ($ket > 4.50) $ket = "Pelayanan Prima";

elseif ($ket > 4.00) $ket = "Sangat Baik";

elseif ($ket > 3.50) $ket = "Baik";

elseif ($ket > 3.00) $ket = "Baik (DC)";

elseif ($ket > 2.50) $ket = "Cukup";

elseif ($ket > 2.00) $ket = "Cukup (DC)";

elseif ($ket > 1.50) $ket = "Buruk";

elseif ($ket > 1.00) $ket = "Sangat Buruk";

else {

$ket\_flag = false;

$ket = "Gagal";

}

$sheet->setCellValue('A' . $numrow, $counter);

$sheet->setCellValue('B' . $numrow, ucwords($ppopd['nama\_opd']));

$sheet->setCellValue('C' . $numrow, $ppopd['indeks\_pelayanan\_publik']);

$sheet->setCellValue('D' . $numrow, strval(floatval($ppopd['indeks\_pelayanan\_publik']) \* 20.0));

if ($ket\_flag) {

**Kode Sumber 43 Controller Pelayanan\_publik**

$sheet->setCellValue('E' . $numrow, $ket);

} else {

$sheet->setCellValue('E' . $numrow, $ket);

$sheet->getStyle('E' . $numrow)->applyFromArray($this->style('font-red'));

}

$numrow++;

}

// ini style tablenya

$sheet->getPageSetup()->setFitToWidth(1);

$sheet->getPageSetup()->setFitToHeight(0);

$sheet->setShowGridlines(true);

$sheet->getStyle('A4:E' . ($numrow - 1))->applyFromArray($this->style('allborder'));

}

public function download()

{

$file = new Xlsx($this->spreadsheet);

header('Content-Type: application/vnd.ms-excel');

header('Content-Disposition: attachment;filename="' . $this->data['nama\_laporan'] . '.xlsx"');

header('Cache-Control: max-age=0');

$file->save('php://output'); // download file

}

}

* + 1. **Controller Pemantauan\_tindak\_lanjut**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class pemantauan\_tindak\_lanjut extends CI\_Controller

{

public $data;

public $spreadsheet;

public function \_\_construct()

{

parent::\_\_construct();

$this->load->model('laporan\_model', 'laporan');

$this->spreadsheet = new Spreadsheet();

}

public function index($formname, $id\_laporan)

{

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->export($formname, $id\_laporan);

$this->download();

}

public function style($stylereq)

{

switch ($stylereq) {

case 'foo\_css':

$style = [];

break;

case 'allborder':

$style = [

'borders' => [

'allBorders' => [

'borderStyle' => \PhpOffice\PhpSpreadsheet\Style\Border::BORDER\_THIN

]

]

];

break;

case 'font-red':

$style = [

'font' => [

'color' => [

'argb' => \PhpOffice\PhpSpreadsheet\Style\Color::COLOR\_RED

]

]

];

break;

default:

# code...

break;

}

return $style;

}

public function export($formname, $id\_laporan)

{

// ini bikin sheetnya

$this->spreadsheet->setActiveSheetIndex(0)->setTitle($this->data['nama\_laporan']);

// ---- INI SHEET PERTAMA ----

// ini atur sheet

$sheet = $this->spreadsheet->getActiveSheet();

// ini atur width

$sheet->getDefaultRowDimension()->setRowHeight(-1);

$sheet->getColumnDimension('A')->setAutoSize(true);

$sheet->getColumnDimension('B')->setWidth(20);

$sheet->getColumnDimension('C')->setWidth(20);

$sheet->getColumnDimension('D')->setWidth(20);

$sheet->getColumnDimension('E')->setWidth(20);

$sheet->getColumnDimension('F')->setWidth(20);

$sheet->getColumnDimension('G')->setWidth(20);

$sheet->getColumnDimension('H')->setWidth(20);

$sheet->getColumnDimension('I')->setWidth(20);

$sheet->getColumnDimension('J')->setWidth(20);

$sheet->getColumnDimension('K')->setWidth(20);

// ini stylenya

$sheet->getStyle('A:K')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A:K')->getAlignment()->setVertical('center');

$sheet->getStyle('A:K')->getAlignment()->setWrapText(true);

$sheet->getStyle('A1:K7')->getFont()->setBold(true);

// ini atur header

$sheet->setCellValue('A1', $this->data['fetch']['ptl']['judul\_laporan'])

->mergeCells('A1:K1');

$sheet->setCellValue('A2', $this->data['nama\_opd'])

->mergeCells('A2:K2');

// ini tablenya

// th numrow 5

$sheet->setCellValue('A4', 'No.')

->mergeCells('A4:A6');

$sheet->setCellValue('B4', 'Temuan')

->mergeCells('B4:B6');

$sheet->setCellValue('C4', 'Rekomendasi')

->mergeCells('C4:C6');

$sheet->setCellValue('D4', 'Status Tahun')

->mergeCells('D4:F4');

$sheet->setCellValue('G4', 'Tindak Lanjut')

->mergeCells('G4:G6');

$sheet->setCellValue('H4', 'Status Tahun')

->mergeCells('H4:J4');

$sheet->setCellValue('K4', 'Catatan BPK')

->mergeCells('K4:K6');

// th buat status tahun

$tgl\_status\_rekomendasi = 'dd/mm/YYYY';

$tgl\_status\_tindak\_lanjut = 'dd/mm/YYYY';

if (isset($this->data['fetch']['ptl']['tgl\_status\_rekomendasi']))

$tgl\_status\_rekomendasi = date('d-m-Y', strtotime($this->data['fetch']['ptl']['tgl\_status\_rekomendasi']));

if (isset($this->data['fetch']['ptl']['tgl\_status\_tindak\_lanjut']))

$tgl\_status\_tindak\_lanjut = date('d-m-Y', strtotime($this->data['fetch']['ptl']['tgl\_status\_tindak\_lanjut']));

$sheet->setCellValue('D5', $tgl\_status\_rekomendasi)

->mergeCells('D5:F5');

$sheet->setCellValue('H5', $tgl\_status\_tindak\_lanjut)

->mergeCells('H5:J5');

// th buat ts tb tt

$sheet->setCellValue('D6', 'TS');

$sheet->setCellValue('E6', 'TB');

$sheet->setCellValue('F6', 'TT');

$sheet->setCellValue('H6', 'TS');

$sheet->setCellValue('I6', 'TB');

$sheet->setCellValue('J6', 'TT');

$sheet->setCellValue('A7', '1');

$sheet->setCellValue('B7', '2');

$sheet->setCellValue('C7', '3');

$sheet->setCellValue('D7', '4');

$sheet->setCellValue('E7', '5');

$sheet->setCellValue('F7', '6');

$sheet->setCellValue('G7', '7');

$sheet->setCellValue('H7', '8');

$sheet->setCellValue('I7', '9');

$sheet->setCellValue('J7', '10');

$sheet->setCellValue('K7', '11');

**Kode Sumber 44 Controller Pemantauan\_tindak\_lanjut**

// td numrow 6

$numrow = 8;

$counter = 0;

foreach ($this->data['fetch']['temuan'] as $temuan) {

$rowspan = (sizeof($this->data['fetch']['htemuan'][$temuan['id\_temuan']]) - 1);

$counter += 1;

$sheet->setCellValue('A' . $numrow, $counter)->mergeCells('A' . $numrow . ':A' . ($numrow + $rowspan));

$sheet->setCellValue('B' . $numrow, $temuan['nama\_temuan'])->mergeCells('B' . $numrow . ':B' . ($numrow + $rowspan));

foreach ($this->data['fetch']['htemuan'][$temuan['id\_temuan']] as $kg) {

$sheet->setCellValue('C' . $numrow, $kg['rekomendasi']);

if ($kg['status\_rekomendasi'] == 'TS') {

$sheet->setCellValue('D' . $numrow, 'TS');

} elseif ($kg['status\_rekomendasi'] == 'TB') {

$sheet->setCellValue('E' . $numrow, 'TB');

} elseif ($kg['status\_rekomendasi'] == 'TT') {

$sheet->setCellValue('F' . $numrow, 'TT');

}

$sheet->setCellValue('G' . $numrow, $kg['rekomendasi']);

if ($kg['status\_tindak\_lanjut'] == 'TS') {

$sheet->setCellValue('H' . $numrow, 'TS');

} elseif ($kg['status\_tindak\_lanjut'] == 'TB') {

$sheet->setCellValue('I' . $numrow, 'TB');

} elseif ($kg['status\_tindak\_lanjut'] == 'TT') {

$sheet->setCellValue('J' . $numrow, 'TT');

}

$sheet->setCellValue('K' . $numrow, $kg['catatan\_bpk']);

$numrow++;

}

}

// ini style tablenya

$sheet->getPageSetup()->setFitToWidth(1);

$sheet->getPageSetup()->setFitToHeight(0);

$sheet->setShowGridlines(true);

$sheet->getStyle('A4:K' . ($numrow - 1))->applyFromArray($this->style('allborder'));

}

public function download()

{

$file = new Xlsx($this->spreadsheet);

header('Content-Type: application/vnd.ms-excel');

header('Content-Disposition: attachment;filename="' . $this->data['nama\_laporan'] . '.xlsx"');

header('Cache-Control: max-age=0');

$file->save('php://output'); // download file

}

}

* + 1. **Controller Realisasi\_fisik**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class realisasi\_fisik extends CI\_Controller

{

public $data;

public $spreadsheet;

public function \_\_construct()

{

parent::\_\_construct();

$this->load->model('laporan\_model', 'laporan');

$this->spreadsheet = new Spreadsheet();

}

public function index($formname, $id\_laporan)

{

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->export($formname, $id\_laporan);

$this->download();

}

public function style($stylereq)

{

switch ($stylereq) {

case 'foo\_css':

$style = [];

break;

case 'allborder':

$style = [

'borders' => [

'allBorders' => [

'borderStyle' => \PhpOffice\PhpSpreadsheet\Style\Border::BORDER\_THIN

]

]

];

break;

case 'font-red':

$style = [

'font' => [

'color' => [

'argb' => \PhpOffice\PhpSpreadsheet\Style\Color::COLOR\_RED

]

]

];

break;

default:

# code...

break;

}

return $style;

}

public function export($formname, $id\_laporan)

{

// ini bikin sheetnya

$this->spreadsheet->setActiveSheetIndex(0)->setTitle($this->data['nama\_laporan']);

// ---- INI SHEET PERTAMA ----

// ini atur sheet

$sheet = $this->spreadsheet->getActiveSheet();

// ini atur width

$sheet->getDefaultRowDimension()->setRowHeight(-1);

$sheet->getColumnDimension('A')->setAutoSize(true);

$sheet->getColumnDimension('B')->setWidth(20);

$sheet->getColumnDimension('C')->setWidth(50);

$sheet->getColumnDimension('D')->setWidth(50);

$sheet->getColumnDimension('E')->setWidth(20);

$sheet->getColumnDimension('F')->setWidth(20);

$sheet->getColumnDimension('G')->setWidth(20);

$sheet->getColumnDimension('H')->setWidth(20);

$sheet->getColumnDimension('I')->setWidth(20);

$sheet->getColumnDimension('J')->setWidth(20);

$sheet->getColumnDimension('K')->setWidth(20);

$sheet->getColumnDimension('L')->setWidth(20);

// ini stylenya

$sheet->getStyle('1')->getAlignment()->setHorizontal('center');

$sheet->getStyle('1')->getAlignment()->setVertical('center');

$sheet->getStyle('E:L')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A:L')->getAlignment()->setVertical('center');

$sheet->getStyle('A7:L9')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A7:L9')->getAlignment()->setVertical('center');

$sheet->getStyle('A7:L9')->getFont()->setBold(true);

$sheet->getStyle('A:L')->getAlignment()->setWrapText(true);

$sheet->getStyle('A1')->getFont()->setBold(true);

// ini atur header

$sheet->setCellValue('A1', 'LAPORAN REALISASI FISIK DAN KEUANGAN SERTA CAPAIAN KINERJA TAHUN ' . date('Y', strtotime($this->data['fetch']['rf']['tgl'])))

->mergeCells('A1:L1');

$sheet->setCellValue('A3', 'UNIT ORGANISASI')->mergeCells('A3:B3');

$sheet->setCellValue('C3', ': ' . $this->data['nama\_opd']);

$sheet->setCellValue('A4', 'BULAN')->mergeCells('A4:B4');

$sheet->setCellValue('C4', ': ' . date('M', strtotime($this->data['fetch']['rf']['tgl'])));

$sheet->setCellValue('A5', 'TAHUN ANGGARAN')->mergeCells('A5:B5');

$sheet->setCellValue('C5', ': ' . date('Y', strtotime($this->data['fetch']['rf']['tgl'])));

// ini tablenya

// th numrow 5

$sheet->setCellValue('A7', 'NO.')

->mergeCells('A7:A8');

$sheet->setCellValue('B7', 'PROGRAM')

->mergeCells('B7:B8');

$sheet->setCellValue('C7', 'URAIAN KEGIATAN')

->mergeCells('C7:C8');

$sheet->setCellValue('D7', 'INDIKATOR')

->mergeCells('D7:D8');

$sheet->setCellValue('E7', 'SATUAN')

->mergeCells('E7:E8');

$sheet->setCellValue('F7', 'KINERJA')

->mergeCells('F7:H7');

$sheet->setCellValue('I7', 'KEUANGAN')

->mergeCells('I7:K7');

$sheet->setCellValue('L7', 'KET')

->mergeCells('L7:L8');

// th numrow 6

$sheet->setCellValue('F8', 'TARGET');

$sheet->setCellValue('G8', 'REALISASI');

$sheet->setCellValue('H8', '%');

$sheet->setCellValue('I8', 'ANGGARAN');

$sheet->setCellValue('J8', 'REALISASI');

$sheet->setCellValue('K8', '%');

$sheet->setCellValue('A9', '1');

$sheet->setCellValue('B9', '2');

$sheet->setCellValue('C9', '3');

$sheet->setCellValue('D9', '4');

$sheet->setCellValue('E9', '6');

$sheet->setCellValue('F9', '7');

$sheet->setCellValue('G9', '8');

$sheet->setCellValue('H9', '9');

$sheet->setCellValue('I9', '10');

$sheet->setCellValue('J9', '11');

$sheet->setCellValue('K9', '12');

$sheet->setCellValue('L9', '13');

// // td numrow 7

$numrow = 10;

$counter = 0;

foreach ($this->data['fetch']['prog'] as $prog) {

$prog\_rowspan = (2 \* sizeof($this->data['fetch']['kg'][$prog['kode\_program']]));

$counter += 1;

$capaian\_realisasi\_kinerja = "";

$capaian\_realisasi\_keuangan = 0;

$persen\_capaian = 0;

$capaian\_realisasi\_kinerja = 0;

$capaian\_anggaran\_keuangan = 0;

$t\_kinerja = 0;

$r\_kinerja = 0;

$r\_keuangan = 0;

foreach ($this->data['fetch']['kg'][$prog['kode\_program']] as $sum) {

$t\_kinerja += (floatval($sum['hasil\_target\_ppas\_final']) + floatval($sum['keluaran\_target\_ppas\_final']));

$r\_kinerja += (floatval($sum['hasil\_realisasi\_kinerja']) + floatval($sum['keluaran\_realisasi\_kinerja']));

$r\_keuangan += intval($sum['realisasi\_keuangan']);

$capaian\_anggaran\_keuangan += intval($sum['pagu\_ppas\_final']);

}

$capaian\_realisasi\_kinerja = floatval($r\_kinerja / $t\_kinerja) \* 100.0;

$capaian\_realisasi\_keuangan = $r\_keuangan;

if ($prog['capaian\_target\_ppas\_final'] != '0') {

$persen\_capaian = round(floatval($capaian\_realisasi\_kinerja) / floatval($prog['capaian\_target\_ppas\_final']) \* 100.0, 2);

}

$persen\_capaian\_keuangan = 0;

if ($capaian\_anggaran\_keuangan != 0) {

$persen\_capaian\_keuangan = floatval($capaian\_realisasi\_keuangan) / floatval($capaian\_anggaran\_keuangan) \* 100.0;

}

$sheet->setCellValue('A' . $numrow, $counter)->mergeCells('A' . $numrow . ':A' . ($numrow + $prog\_rowspan));

$sheet->setCellValue('B' . $numrow, ucwords($prog['nama\_program']))->mergeCells('B' . $numrow . ':B' . ($numrow + $prog\_rowspan));

$sheet->setCellValue('C' . $numrow, '');

$sheet->setCellValue('D' . $numrow, $prog['capaian\_indikator']);

$sheet->setCellValue('E' . $numrow, $prog['capaian\_satuan']);

$sheet->setCellValue('F' . $numrow, $prog['capaian\_target\_ppas\_final']);

$sheet->setCellValue('G' . $numrow, round($capaian\_realisasi\_kinerja, 2));

$sheet->setCellValue('H' . $numrow, round($persen\_capaian, 2));

$sheet->setCellValue('I' . $numrow, $capaian\_anggaran\_keuangan);

$sheet->setCellValue('J' . $numrow, $capaian\_realisasi\_keuangan);

$sheet->setCellValue('K' . $numrow, round($persen\_capaian\_keuangan, 2));

$sheet->setCellValue('L' . $numrow, '')->mergeCells('L' . $numrow . ':L' . ($numrow + $prog\_rowspan));

$numrow++;

$persen\_keluaran = "0";

$persen\_hasil = "0";

foreach ($this->data['fetch']['kg'][$prog['kode\_program']] as $kg) {

$keluaran\_realisasi\_kinerja = "0";

$realisasi\_keuangan = "0";

$hasil\_realisasi\_kinerja = "0";

$persen\_keluaran = "0";

$persen\_hasil = "0";

// $hasil\_realisasi\_keuangan = "";

if (isset($kg['keluaran\_realisasi\_kinerja'])) $keluaran\_realisasi\_kinerja = $kg['keluaran\_realisasi\_kinerja'];

if (isset($kg['realisasi\_keuangan'])) $realisasi\_keuangan = $kg['realisasi\_keuangan'];

if (isset($kg['hasil\_realisasi\_kinerja'])) $hasil\_realisasi\_kinerja = $kg['hasil\_realisasi\_kinerja'];

// if(isset($kg['hasil\_realisasi\_keuangan'])) $hasil\_realisasi\_keuangan = $kg['hasil\_realisasi\_keuangan'];

if ($kg['keluaran\_target\_ppas\_final'] != '0') {

$persen\_keluaran = round(floatval($keluaran\_realisasi\_kinerja) / floatval($kg['keluaran\_target\_ppas\_final']) \* 100.0, 2);

}

if ($kg['hasil\_target\_ppas\_final'] != '0') {

$persen\_hasil = round(floatval($hasil\_realisasi\_kinerja) / floatval($kg['hasil\_target\_ppas\_final']) \* 100.0, 2);

}

$persen\_keuangan = 0;

if (floatval($kg['pagu\_ppas\_final']) != 0) {

$persen\_keuangan = floatval($realisasi\_keuangan) / floatval($kg['pagu\_ppas\_final']) \* 100.0;

}

**Kode Sumber 45 Controller Realisasi\_fisik**

$sheet->setCellValue('C' . $numrow, ucwords($kg['nama\_kegiatan']))->mergeCells('C' . $numrow . ':C' . ($numrow + 1));

$wizard = new PhpOffice\PhpSpreadsheet\Helper\Html;

$richText = $wizard->toRichTextObject('<strong><u>Output:</u></strong>' . ucwords($kg['keluaran\_indikator']));

$sheet->setCellValue('D' . $numrow, $richText);

$richText = $wizard->toRichTextObject('<strong><u>Output:</u></strong>' . ucwords($kg['hasil\_indikator']));

$sheet->setCellValue('D' . ($numrow + 1), $richText);

$sheet->setCellValue('E' . $numrow, $kg['keluaran\_satuan']);

$sheet->setCellValue('F' . $numrow, $kg['keluaran\_target\_ppas\_final']);

$sheet->setCellValue('G' . $numrow, $keluaran\_realisasi\_kinerja);

$sheet->setCellValue('H' . $numrow, round($persen\_keluaran));

$sheet->setCellValue('I' . $numrow, $kg['pagu\_ppas\_final'])->mergeCells('I' . $numrow . ':I' . ($numrow + 1));

$sheet->setCellValue('J' . $numrow, $realisasi\_keuangan)->mergeCells('J' . $numrow . ':J' . ($numrow + 1));

$sheet->setCellValue('K' . $numrow, round($persen\_keuangan))->mergeCells('K' . $numrow . ':K' . ($numrow + 1));

$numrow++;

$sheet->setCellValue('E' . $numrow, $kg['hasil\_satuan']);

$sheet->setCellValue('F' . $numrow, $kg['hasil\_target\_ppas\_final']);

$sheet->setCellValue('G' . $numrow, $kg['hasil\_realisasi\_kinerja']);

$sheet->setCellValue('H' . $numrow, round($persen\_hasil, 2));

$numrow++;

}

}

// ini style tablenya

$sheet->getPageSetup()->setFitToWidth(1);

$sheet->getPageSetup()->setFitToHeight(0);

$sheet->setShowGridlines(true);

$sheet->getStyle('A7:L' . ($numrow - 1))->applyFromArray($this->style('allborder'));

}

public function download()

{

$file = new Xlsx($this->spreadsheet);

header('Content-Type: application/vnd.ms-excel');

header('Content-Disposition: attachment;filename="' . $this->data['nama\_laporan'] . '.xlsx"');

header('Cache-Control: max-age=0');

$file->save('php://output'); // download file

}

}

* + 1. **Controller Rekap\_pokja**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class rekap\_pokja extends CI\_Controller {

{

public $data;

public $spreadsheet;

public function \_\_construct()

{

parent::\_\_construct();

$this->load->model('laporan\_model', 'laporan');

$this->spreadsheet = new Spreadsheet();

}

public function index($formname, $id\_laporan)

{

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->export($formname, $id\_laporan);

$this->download();

}

public function style($stylereq)

{

switch ($stylereq) {

case 'foo\_css':

$style = [];

break;

case 'allborder':

$style = [

'borders' => [

'allBorders' => [

'borderStyle' => \PhpOffice\PhpSpreadsheet\Style\Border::BORDER\_THIN

]

]

];

break;

case 'font-red':

$style = [

'font' => [

'color' => [

'argb' => \PhpOffice\PhpSpreadsheet\Style\Color::COLOR\_RED

]

]

];

break;

default:

# code...

break;

}

return $style;

}

public function export($formname, $id\_laporan)

{

// ini bikin sheetnya

$this->spreadsheet->setActiveSheetIndex(0)->setTitle($this->data['nama\_laporan']);

// ---- INI SHEET PERTAMA ----

// ini atur sheet

$sheet = $this->spreadsheet->getActiveSheet();

// ini atur width

$sheet->getDefaultRowDimension()->setRowHeight(-1);

$sheet->getColumnDimension('A')->setAutoSize(true);

$sheet->getColumnDimension('B')->setWidth(50);

$sheet->getColumnDimension('C')->setWidth(50);

$sheet->getColumnDimension('D')->setWidth(20);

$sheet->getColumnDimension('E')->setWidth(20);

$sheet->getColumnDimension('F')->setWidth(20);

// ini stylenya

$sheet->getStyle('1:4')->getAlignment()->setHorizontal('center');

$sheet->getStyle('1:4')->getAlignment()->setVertical('center');

$sheet->getStyle('A')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C:F')->getAlignment()->setHorizontal('center');

$sheet->getStyle('B5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A:F')->getAlignment()->setVertical('center');

$sheet->getStyle('A:F')->getAlignment()->setWrapText(true);

$sheet->getStyle('A1:F5')->getFont()->setBold(true);

// ini atur header

$sheet->setCellValue('A1', 'DAFTAR PENUGASAN KELOMPOK KERJA ')

->mergeCells('A1:F1');

$sheet->setCellValue('A2', $this->data['nama\_opd'])

->mergeCells('A2:F2');

// ini tablenya

// th numrow 5

$sheet->setCellValue('A4', 'No.');

$sheet->setCellValue('B4', 'Nama');

$sheet->setCellValue('C4', 'Paket Pekerjaan');

$sheet->setCellValue('D4', 'Nilai Pagu');

$sheet->setCellValue('E4', 'Jabatan dalam Tim');

$sheet->setCellValue('F4', 'Keterangan');

// th numrow 6

$sheet->setCellValue('A5', '1');

$sheet->setCellValue('B5', '2');

$sheet->setCellValue('C5', '3');

$sheet->setCellValue('D5', '4');

$sheet->setCellValue('E5', '5');

$sheet->setCellValue('F5', '6');

// // td numrow 7

$numrow = 6;

$counter = 0;

foreach ($this->data['fetch']['p'] as $prog) {

$prog\_rowspan = (sizeof($this->data['fetch']['drp'][$prog['id\_pegawai']]) - 1);

$counter += 1;

$sheet->setCellValue('A' . $numrow, $counter)

->mergeCells('A' . $numrow . ':A' . ($numrow + $prog\_rowspan));

$sheet->setCellValue('B' . $numrow, ucwords($prog['nama']))

->mergeCells('B' . $numrow . ':B' . ($numrow + $prog\_rowspan));

foreach ($this->data['fetch']['drp'][$prog['id\_pegawai']] as $kg) {

$sheet->setCellValue('C' . $numrow, ucwords($kg['nama\_paket\_kerja']));

$sheet->setCellValue('D' . $numrow, $kg['pagu']);

$sheet->setCellValue('E' . $numrow, ucwords($kg['jabatan']));

$sheet->setCellValue('F' . $numrow, ucwords($kg['ket']));

**Kode Sumber 46 Controller Rekap\_pokja**

$numrow++;

}

}

// ini style tablenya

$sheet->getPageSetup()->setFitToWidth(1);

$sheet->getPageSetup()->setFitToHeight(0);

$sheet->setShowGridlines(true);

$sheet->getStyle('A4:F' . ($numrow - 1))->applyFromArray($this->style('allborder'));

}

public function download()

{

$file = new Xlsx($this->spreadsheet);

header('Content-Type: application/vnd.ms-excel');

header('Content-Disposition: attachment;filename="' . $this->data['nama\_laporan'] . '.xlsx"');

header('Cache-Control: max-age=0');

$file->save('php://output'); // download file

}

}

* + 1. **Controller Rekap\_tender**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class rekap\_tender extends CI\_Controller

{

public $data;

public $spreadsheet;

public function \_\_construct()

{

parent::\_\_construct();

$this->load->model('laporan\_model', 'laporan');

$this->spreadsheet = new Spreadsheet();

}

public function index($formname, $id\_laporan)

{

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->export($formname, $id\_laporan);

$this->download();

}

public function style($stylereq)

{

switch ($stylereq) {

case 'foo\_css':

$style = [];

break;

case 'allborder':

$style = [

'borders' => [

'allBorders' => [

'borderStyle' => \PhpOffice\PhpSpreadsheet\Style\Border::BORDER\_THIN

]

]

];

break;

case 'font-red':

$style = [

'font' => [

'color' => [

'argb' => \PhpOffice\PhpSpreadsheet\Style\Color::COLOR\_RED

]

]

];

break;

default:

# code...

break;

}

return $style;

}

public function export($formname, $id\_laporan)

{

// ini bikin sheetnya

$this->spreadsheet->setActiveSheetIndex(0)->setTitle($this->data['nama\_laporan']);

// ---- INI SHEET PERTAMA ----

// ini atur sheet

$sheet = $this->spreadsheet->getActiveSheet();

// ini atur width

$sheet->getDefaultRowDimension()->setRowHeight(-1);

$sheet->getColumnDimension('A')->setAutoSize(true);

$sheet->getColumnDimension('B')->setWidth(50);

$sheet->getColumnDimension('C')->setWidth(20);

$sheet->getColumnDimension('D')->setWidth(20);

$sheet->getColumnDimension('E')->setWidth(20);

$sheet->getColumnDimension('F')->setWidth(20);

$sheet->getColumnDimension('G')->setWidth(20);

$sheet->getColumnDimension('H')->setWidth(50);

$sheet->getColumnDimension('I')->setWidth(20);

// ini stylenya

$sheet->getStyle('1:4')->getAlignment()->setHorizontal('center');

$sheet->getStyle('1:4')->getAlignment()->setVertical('center');

$sheet->getStyle('A')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C:G')->getAlignment()->setHorizontal('center');

$sheet->getStyle('I4')->getAlignment()->setHorizontal('center');

$sheet->getStyle('B5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('G5:I5')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A:I')->getAlignment()->setVertical('center');

$sheet->getStyle('A:I')->getAlignment()->setWrapText(true);

$sheet->getStyle('G4')->getAlignment()->setWrapText(true);

$sheet->getStyle('A1:I5')->getFont()->setBold(true);

// ini atur header

$sheet->setCellValue('A1', $this->data['fetch']['rt']['judul\_rekap\_tender'])

->mergeCells('A1:I1');

$sheet->setCellValue('A2', $this->data['nama\_opd'])

->mergeCells('A2:I2');

// ini tablenya

// th numrow 5

$sheet->setCellValue('A4', 'No.');

$sheet->setCellValue('B4', 'Paket Pekerjaan');

$sheet->setCellValue('C4', 'Nilai Pagu');

$sheet->setCellValue('D4', 'Nilai HPS');

$sheet->setCellValue('E4', 'Pemenang');

$sheet->setCellValue('F4', 'Harga Kontrak');

$sheet->setCellValue('G4', 'Prosentase Harga Kontrak terhadao HPS');

$sheet->setCellValue('H4', 'OPD');

$sheet->setCellValue('I4', 'Ket');

// th numrow 6

$sheet->setCellValue('A5', '1');

$sheet->setCellValue('B5', '2');

$sheet->setCellValue('C5', '3');

$sheet->setCellValue('D5', '4');

$sheet->setCellValue('E5', '5');

$sheet->setCellValue('F5', '6');

$sheet->setCellValue('G5', '7');

$sheet->setCellValue('H5', '8');

$sheet->setCellValue('I5', '9');

// // td numrow 7

$numrow = 6;

$counter = 0;

foreach ($this->data['fetch']['drt'] as $drt) {

$vl = 0;

if(floatval($drt['nilai\_hps'] != 0)) $vl = strval(floatval($drt['harga\_kontrak'])/floatval($drt['nilai\_hps'])\*100.0);

else $vl = "nilai HPS 0";

$counter += 1;

$sheet->setCellValue('A' .$numrow, $counter);

$sheet->setCellValue('B' .$numrow, $drt['nama\_paket\_kerja']);

$sheet->setCellValue('C' .$numrow, $drt['pagu']);

$sheet->setCellValue('D' .$numrow, $drt['nilai\_hps']);

$sheet->setCellValue('E' .$numrow, $drt['pemenang']);

$sheet->setCellValue('F' .$numrow, $drt['harga\_kontrak']);

$sheet->setCellValue('G' .$numrow, $vl);

$sheet->setCellValue('H' .$numrow, ucwords($drt['nama\_opd']));

$sheet->setCellValue('I' .$numrow, $drt['ket']);

$numrow ++;

}

// ini style tablenya

$sheet->getPageSetup()->setFitToWidth(1);

$sheet->getPageSetup()->setFitToHeight(0);

$sheet->setShowGridlines(true);

$sheet->getStyle('A4:I' . ($numrow - 1))->applyFromArray($this->style('allborder'));

}

**Kode Sumber 47 Controller Rekap\_tender**

public function download()

{

$file = new Xlsx($this->spreadsheet);

header('Content-Type: application/vnd.ms-excel');

header('Content-Disposition: attachment;filename="' . $this->data['nama\_laporan'] . '.xlsx"');

header('Cache-Control: max-age=0');

$file->save('php://output'); // download file

}

}

* + 1. **Controller Sotk**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class sotk extends CI\_Controller

{

public $data;

public $spreadsheet;

public function \_\_construct()

{

parent::\_\_construct();

$this->load->model('laporan\_model', 'laporan');

$this->spreadsheet = new Spreadsheet();

}

public function index($formname, $id\_laporan)

{

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->export($formname, $id\_laporan);

$this->download();

}

public function style($stylereq)

{

switch ($stylereq) {

case 'foo\_css':

$style = [];

break;

case 'allborder':

$style = [

'borders' => [

'allBorders' => [

'borderStyle' => \PhpOffice\PhpSpreadsheet\Style\Border::BORDER\_THIN

]

]

];

break;

case 'font-red':

$style = [

'font' => [

'color' => [

'argb' => \PhpOffice\PhpSpreadsheet\Style\Color::COLOR\_RED

]

]

];

break;

default:

# code...

break;

}

return $style;

}

public function export($formname, $id\_laporan)

{

// ini bikin sheetnya

$this->spreadsheet->setActiveSheetIndex(0)->setTitle($this->data['nama\_laporan']);

// ---- INI SHEET PERTAMA ----

// ini atur sheet

$sheet = $this->spreadsheet->getActiveSheet();

// ini atur width

$sheet->getDefaultRowDimension()->setRowHeight(-1);

$sheet->getColumnDimension('A')->setAutoSize(true);

$sheet->getColumnDimension('B')->setWidth(50);

$sheet->getColumnDimension('C')->setWidth(20);

// ini stylenya

$sheet->getStyle('1:4')->getAlignment()->setHorizontal('center');

$sheet->getStyle('1:4')->getAlignment()->setVertical('center');

$sheet->getStyle('A')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A:C')->getAlignment()->setVertical('center');

$sheet->getStyle('A:C')->getAlignment()->setWrapText(true);

$sheet->getStyle('A1:C4')->getFont()->setBold(true);

// ini atur header

$sheet->setCellValue('A1', 'JUMLAH LEMBAGA/SOTK ')

->mergeCells('A1:C1');

$sheet->setCellValue('A2', $this->data['nama\_opd'])

->mergeCells('A2:C2');

// ini tablenya

// th numrow 5

$sheet->setCellValue('A4', 'No.');

$sheet->setCellValue('B4', 'OPD');

$sheet->setCellValue('C4', 'BESARAN');

// td numrow 6

$numrow = 5;

$counter = 0;

foreach ($this->data['fetch']['sotkopd'] as $sotkopd) {

$counter += 1;

$sheet->setCellValue('A' .$numrow, $counter);

$sheet->setCellValue('B' .$numrow, ucwords($sotkopd['nama\_opd']));

$sheet->setCellValue('C' .$numrow, $sotkopd['besaran']);

**Kode Sumber 48 Controller Sotk**

$numrow ++;

}

// ini style tablenya

$sheet->getPageSetup()->setFitToWidth(1);

$sheet->getPageSetup()->setFitToHeight(0);

$sheet->setShowGridlines(true);

$sheet->getStyle('A4:C' . ($numrow - 1))->applyFromArray($this->style('allborder'));

}

public function download()

{

$file = new Xlsx($this->spreadsheet);

header('Content-Type: application/vnd.ms-excel');

header('Content-Disposition: attachment;filename="' . $this->data['nama\_laporan'] . '.xlsx"');

header('Cache-Control: max-age=0');

$file->save('php://output'); // download file

}

}

* + 1. **Controller Tatalaksana**

<?php

defined('BASEPATH') or exit('No direct script access allowed');

use PhpOffice\PhpSpreadsheet\Spreadsheet;

use PhpOffice\PhpSpreadsheet\Writer\Xlsx;

class tatalaksana extends CI\_Controller

{

public $data;

public $spreadsheet;

public function \_\_construct() {

parent::\_\_construct();

$this->load->model('laporan\_model', 'laporan');

$this->spreadsheet = new Spreadsheet();

}

public function index($formname, $id\_laporan) {

$this->data['nama\_laporan'] = ucwords(str\_replace('\_', ' ', $formname));

$this->data['fetch'] = $this->laporan->get\_laporan\_data\_by\_name\_id($formname, $id\_laporan);

$this->data['nama\_opd'] = $this->session->tempdata('nama\_opd');

$this->export($formname, $id\_laporan);

$this->download();

}

public function style($stylereq) {

switch ($stylereq) {

case 'foo\_css':

$style = [];

break;

case 'allborder':

$style = [

'borders' => [

'allBorders' => [

'borderStyle' => \PhpOffice\PhpSpreadsheet\Style\Border::BORDER\_THIN

]

]

];

break;

case 'font-red':

$style = [

'font' => [

'color' => [

'argb' => \PhpOffice\PhpSpreadsheet\Style\Color::COLOR\_RED

]

]

];

break;

default:

# code...

break;

}

return $style;

}

public function export($formname, $id\_laporan) {

// ini bikin sheetnya

$this->spreadsheet->setActiveSheetIndex(0)->setTitle($this->data['nama\_laporan']);

// ---- INI SHEET PERTAMA ----

// ini atur sheet

$sheet = $this->spreadsheet->getActiveSheet();

// ini atur width

$sheet->getDefaultRowDimension()->setRowHeight(-1);

$sheet->getColumnDimension('A')->setAutoSize(true);

$sheet->getColumnDimension('B')->setWidth(50);

$sheet->getColumnDimension('C')->setWidth(20);

$sheet->getColumnDimension('D')->setWidth(20);

$sheet->getColumnDimension('E')->setWidth(20);

$sheet->getColumnDimension('F')->setWidth(20);

$sheet->getColumnDimension('G')->setWidth(20);

$sheet->getColumnDimension('H')->setWidth(20);

$sheet->getColumnDimension('I')->setWidth(20);

$sheet->getColumnDimension('J')->setWidth(20);

$sheet->getColumnDimension('K')->setWidth(20);

$sheet->getStyle('1:4')->getAlignment()->setHorizontal('center');

$sheet->getStyle('1:4')->getAlignment()->setVertical('center');

$sheet->getStyle('A')->getAlignment()->setHorizontal('center');

$sheet->getStyle('C:K')->getAlignment()->setHorizontal('center');

$sheet->getStyle('A:K')->getAlignment()->setVertical('center');

$sheet->getStyle('1:4')->getAlignment()->setWrapText(true);

$sheet->getStyle('A1:K4')->getFont()->setBold(true);

// ini atur header

$sheet->setCellValue('A1', 'LAPORAN HASIL PENGAMATAN TATALAKSANA TAHUN ' . date('Y', strtotime($this->data['fetch']['t']['tgl'])))

->mergeCells('A1:K1');

**Kode Sumber 49 Controller Tatalaksana**

$sheet->setCellValue('A2', $this->data['nama\_opd'])

->mergeCells('A2:K2');

// ini tablenya

// th numrow 5

$sheet->setCellValue('A4', 'No.');

$sheet->setCellValue('B4', 'Nama Perangkat Daerah');

$sheet->setCellValue('C4', 'Nilai Peserta Uji Kompetensi Ketatalaksanaan');

$sheet->setCellValue('D4', 'Nilai Hasil Uji Kompetensi x 40%');

$sheet->setCellValue('E4', 'SOP');

$sheet->setCellValue('F4', 'Tata Naskah Dinas');

$sheet->setCellValue('G4', 'Pakaian Dinas');

$sheet->setCellValue('H4', 'Hari dan Jam Kerja');

$sheet->setCellValue('I4', 'Nilai Penilaian Lapangan x 60%');

$sheet->setCellValue('J4', 'Jumlah Nilai');

$sheet->setCellValue('K4', 'Keterangan');

// td numrow 6

$numrow = 5;

$counter = 0;

foreach ($this->data['fetch']['topd'] as $topd) {

$counter += 1;

$sheet->setCellValue('A' .$numrow, $counter);

$sheet->setCellValue('B' .$numrow, ucwords($topd['nama\_opd']));

$sheet->setCellValue('C' .$numrow, $topd['uji\_kompetensi']);

$sheet->setCellValue('D' .$numrow, strval(floatval($topd['uji\_kompetensi']) \* 4.0 / 10.0));

$sheet->setCellValue('E' .$numrow, $topd['sop']);

$sheet->setCellValue('F' .$numrow, $topd['tata\_naskah\_dinas']);

$sheet->setCellValue('G' .$numrow, $topd['pakaian\_dinas']);

$sheet->setCellValue('H' .$numrow, $topd['jam\_kerja']);

$sheet->setCellValue('I' .$numrow, strval((floatval($topd['sop']) + floatval($topd['tata\_naskah\_dinas']) + floatval($topd['pakaian\_dinas']) + floatval($topd['jam\_kerja'])) \* 6.0 / 10.0));

$sheet->setCellValue('J' .$numrow, strval((floatval($topd['uji\_kompetensi']) \* 4.0 / 10.0) + ((floatval($topd['sop']) + floatval($topd['tata\_naskah\_dinas']) + floatval($topd['pakaian\_dinas']) + floatval($topd['jam\_kerja'])) \* 6.0 / 10.0)));

$sheet->setCellValue('K' .$numrow, $topd['ket']);

$numrow ++;

}

// ini style tablenya

$sheet->getPageSetup()->setFitToWidth(1);

$sheet->getPageSetup()->setFitToHeight(0);

$sheet->setShowGridlines(true);

$sheet->getStyle('A4:K' . ($numrow - 1))->applyFromArray($this->style('allborder'));

}

public function download() {

$file = new Xlsx($this->spreadsheet);

header('Content-Type: application/vnd.ms-excel');

header('Content-Disposition: attachment;filename="' . $this->data['nama\_laporan'] . '.xlsx"');

header('Cache-Control: max-age=0');

$file->save('php://output'); // download file

}

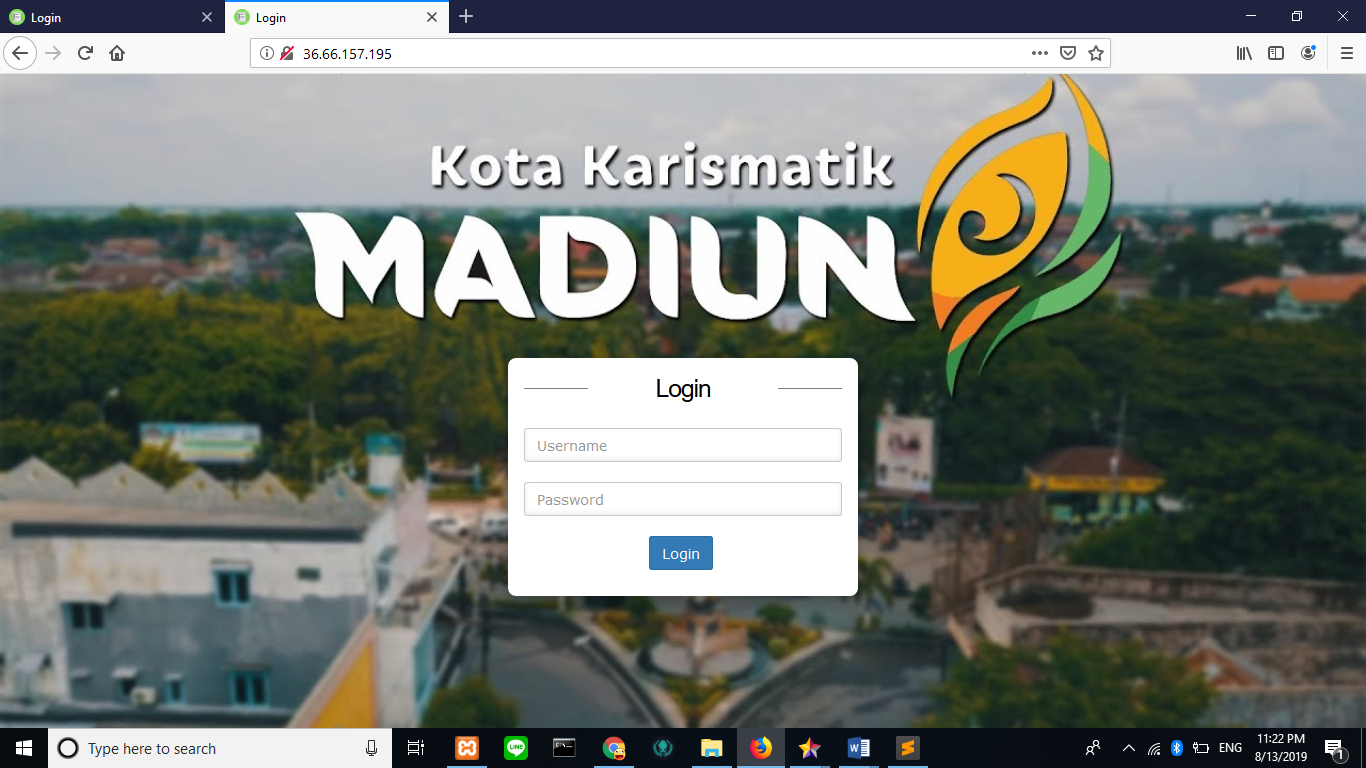
}

## Implementasi View

Bagian ini merupakan bagian tampilan antarmuka halaman-halaman yang ada di dalam sistem *report management.*

* + 1. **Halaman Login**

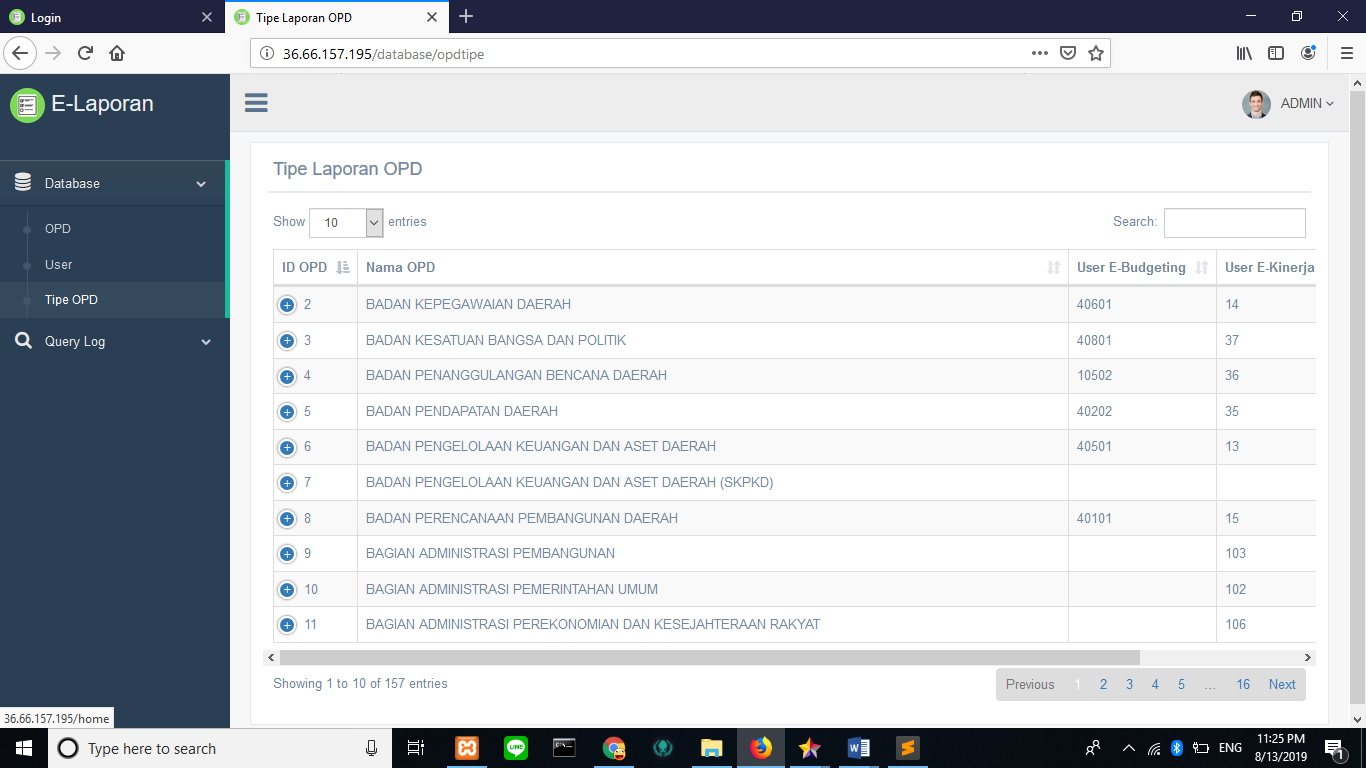
Semua pengguna harus melakukan login terlebih dahulu.



**Gambar 15 Halaman Login**

* + 1. **Halaman Home Admin**

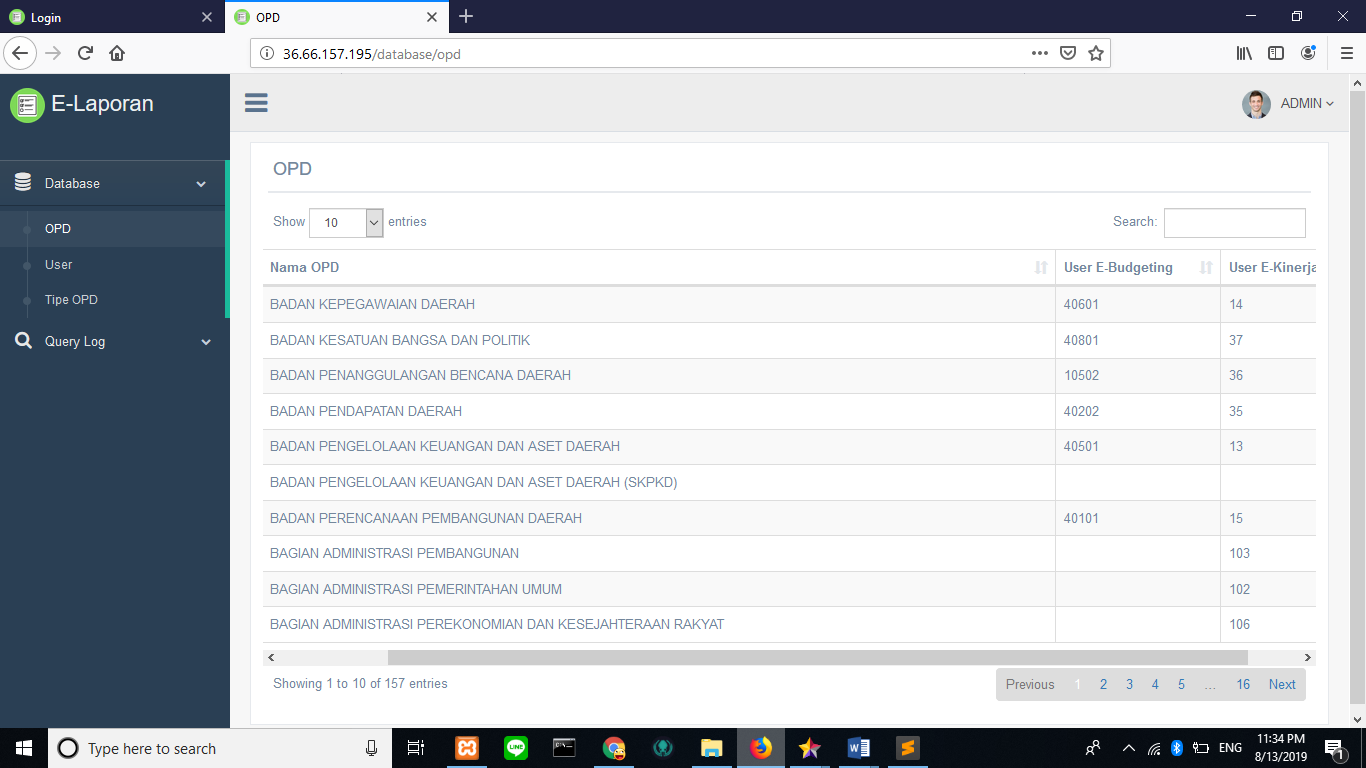
Halaman ini khusus untuk Admin. Halaman ini merupakan halaman utama setelah Admin melakukan login. Admin dapat melihat data OPD, melihat data User OPD, melihat data tipe laporan yang bisa dibuat oleh OPD, serta melihat log aktifitas User OPD.



**Gambar 16 Halaman Home Admin**

* + 1. **Halaman Melihat Data OPD**

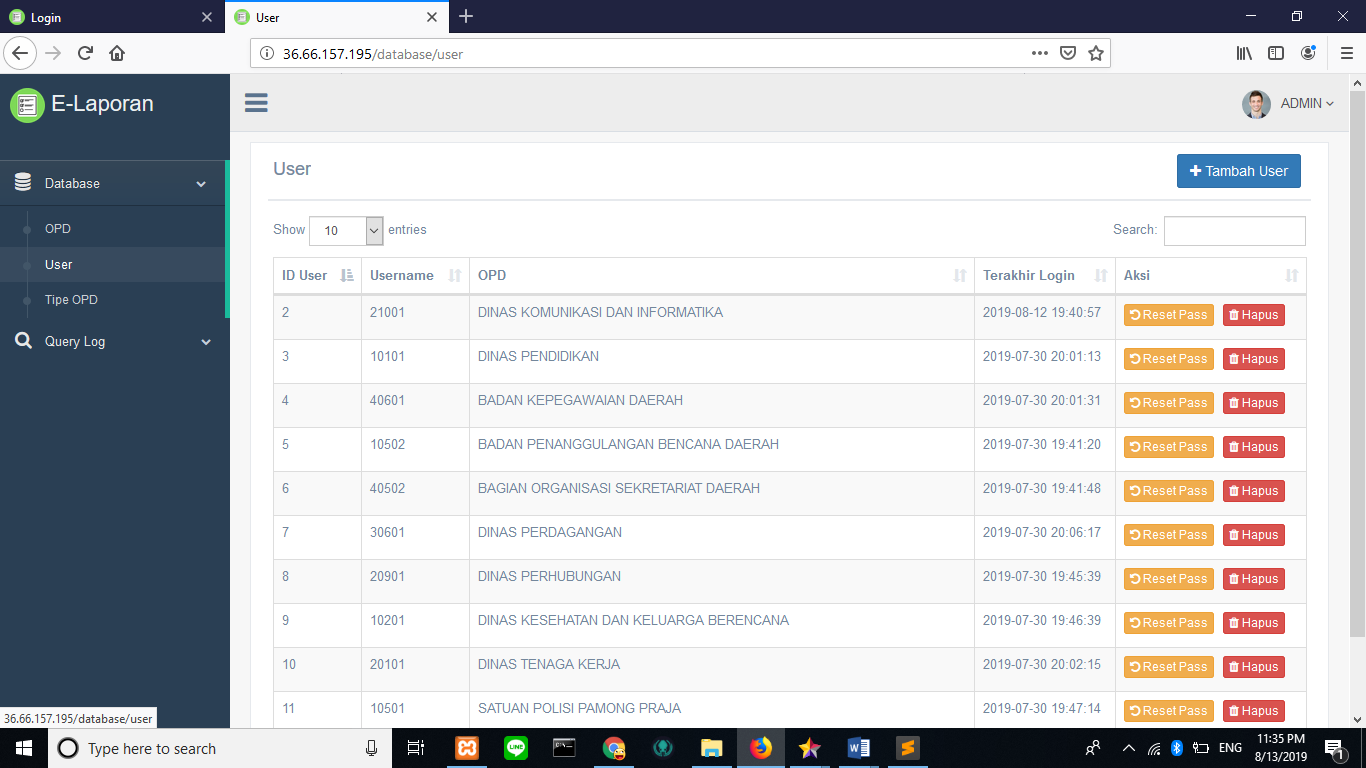
Halaman ini khusus untuk Admin. Halaman ini merupakan halaman yang menampilkan data Organisasi Perangkat Daerah (OPD) di Kota Madiun.



**Gambar 17 Halaman Melihat Data OPD**

* + 1. **Halaman Melihat Data User**

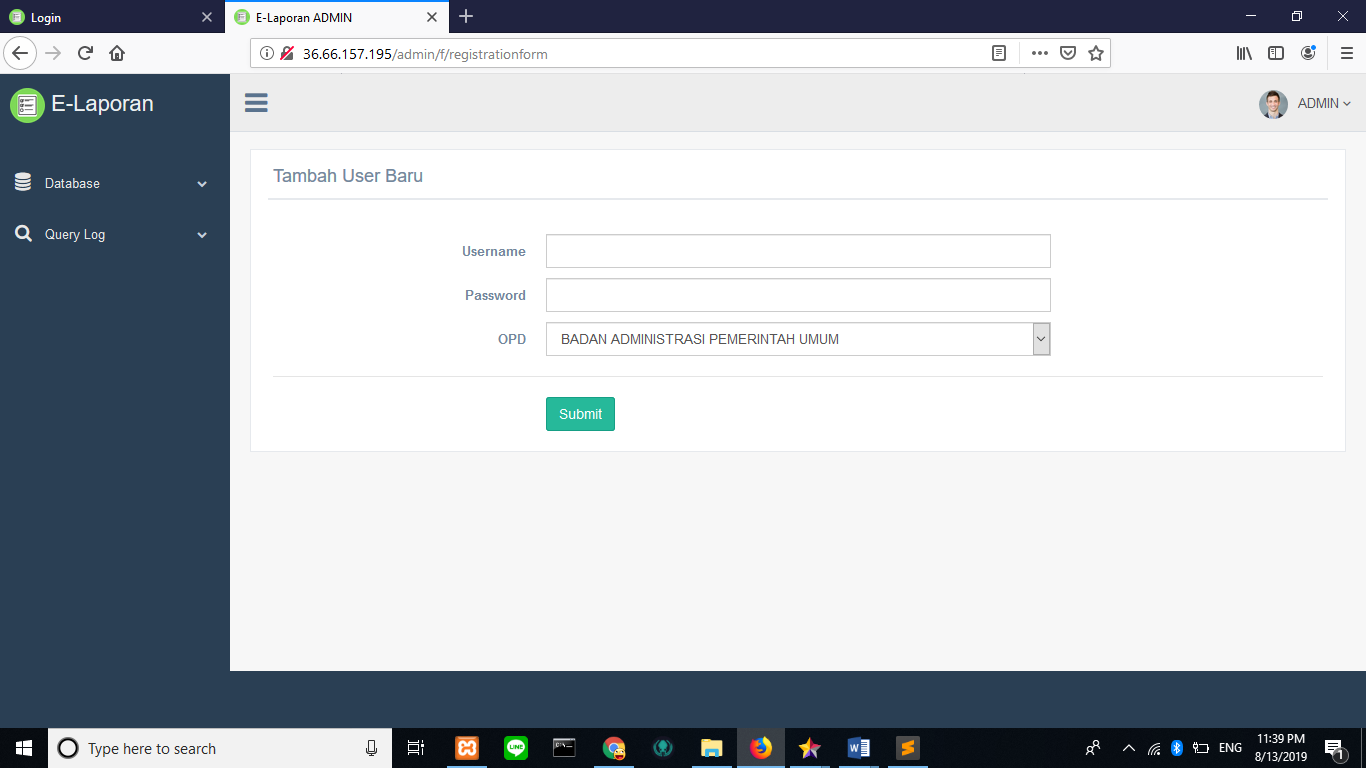
Halaman ini khusus untuk Admin. Halaman ini merupakan halaman yang menampilkan data User Organisasi Perangkat Daerah (OPD) di Kota Madiun. Admin dapat melakukan tambah user, reset password user, dan menghapus user.



**Gambar 18 Halaman Melihat Data User OPD**

* + 1. **Halaman Menambah Data User**

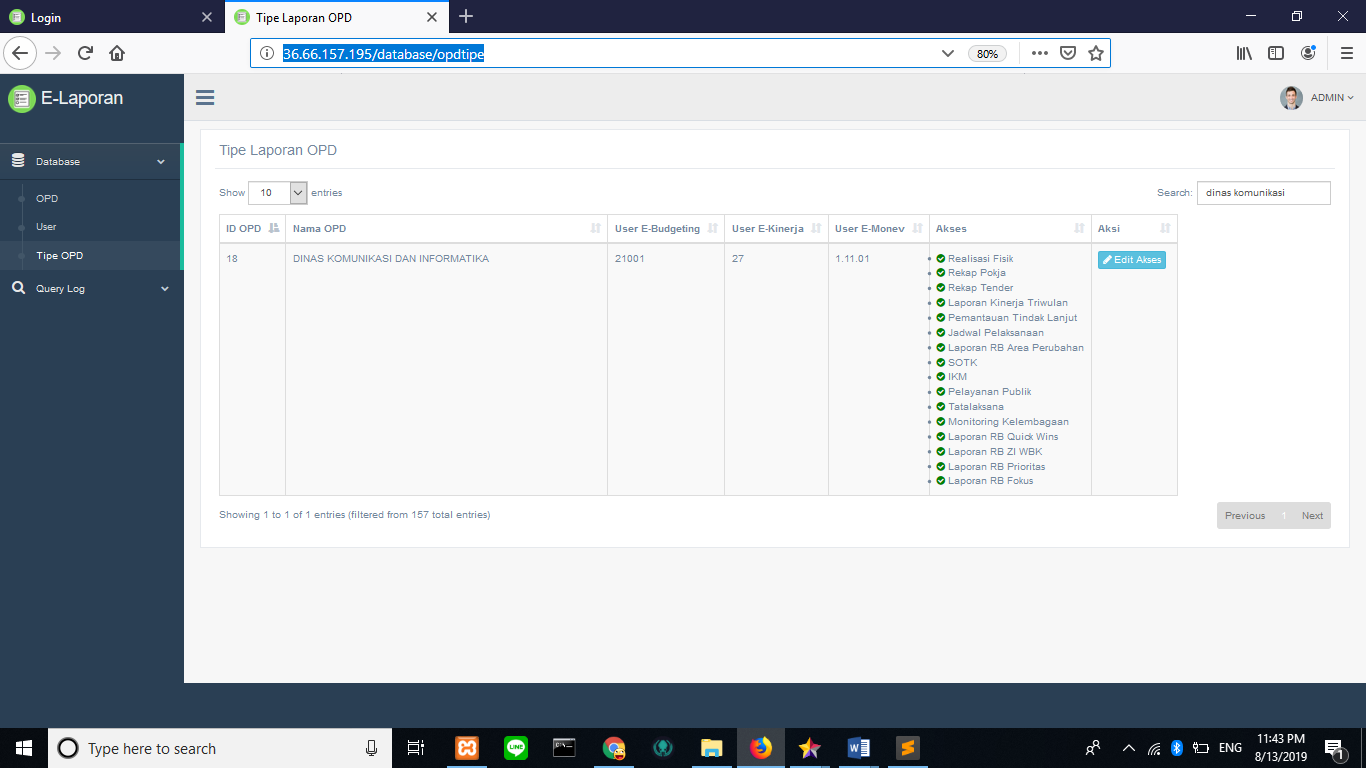
Halaman ini khusus untuk Admin. Halaman ini merupakan halaman untuk input data user yang akan ditambah.



**Gambar 19 Halaman Menambah Data User**

* + 1. **Halaman Melihat Tipe Laporan OPD**

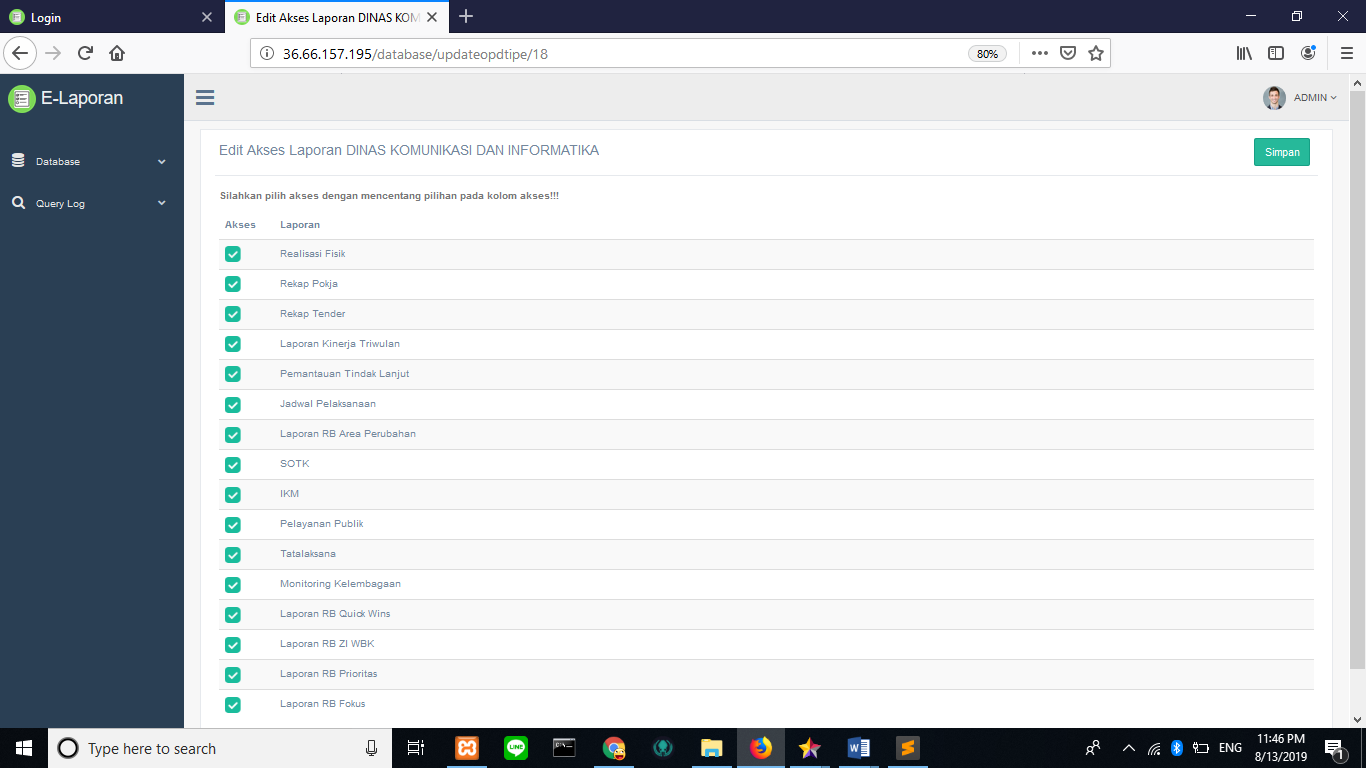
Halaman ini khusus untuk Admin. Halaman ini merupakan halaman untuk melihat tipe laporan yang bisa diakses oleh OPD.



**Gambar 20 Halaman Melihat Tipe Laporan OPD**

* + 1. **Halaman Edit Akses User OPD**

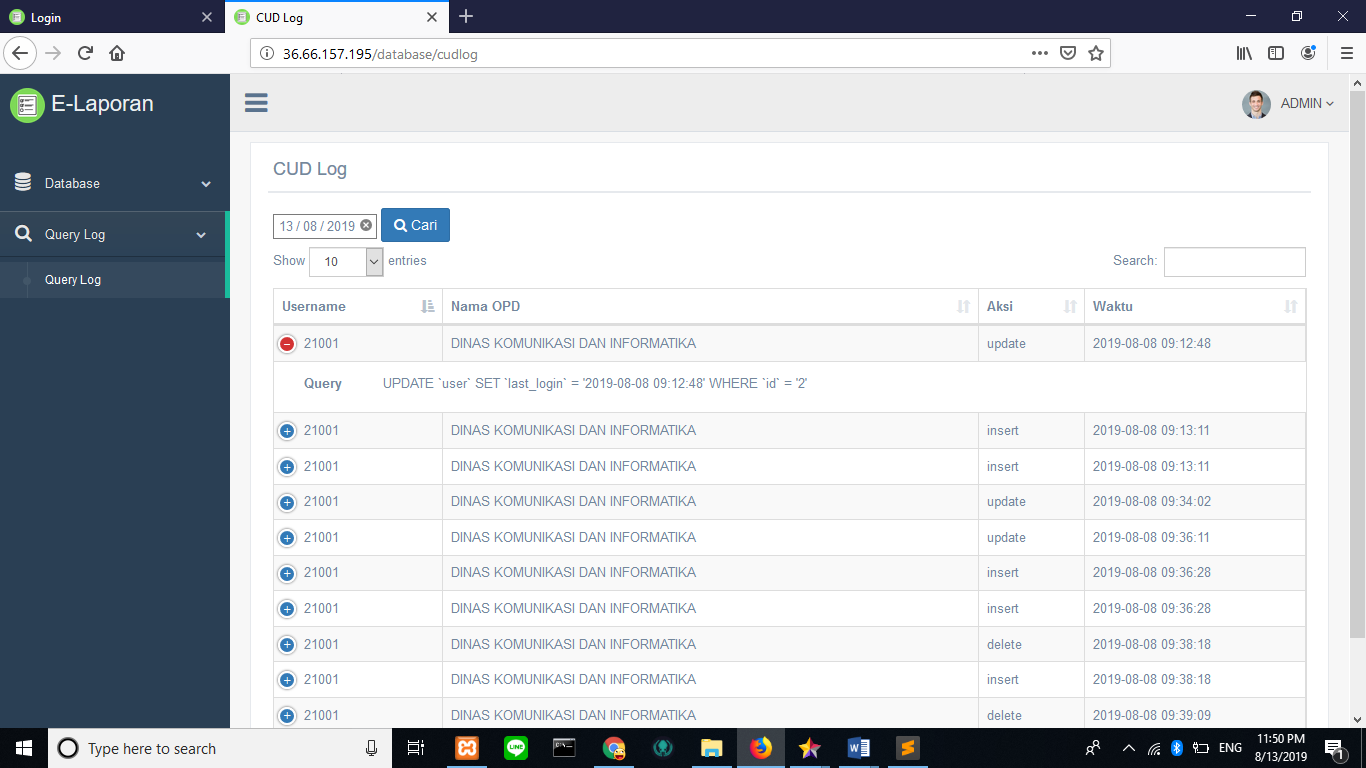
Halaman ini khusus untuk Admin. Halaman ini merupakan halaman untuk mengubah tipe laporan yang bisa diakses oleh OPD.



**Gambar 21 Halaman Edit Akses User OPD**

* + 1. **Halaman Melihat Log Aktifitas User OPD**

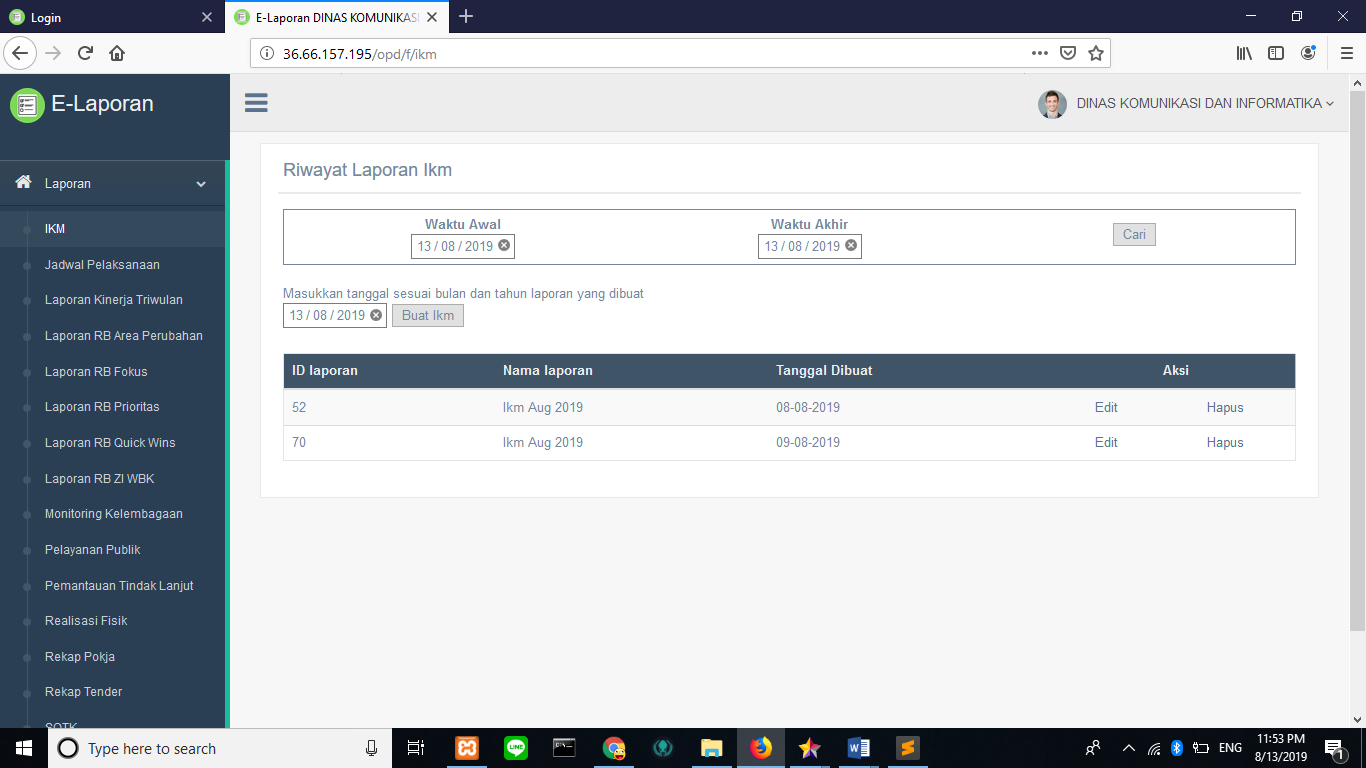
Halaman ini khusus untuk Admin. Halaman ini merupakan halaman yang menampilkan aktifitas yang dilakukan oleh User OPD.



**Gambar 22 Halaman Melihat Log Aktifitas User OPD**

* + 1. **Halaman Home User**

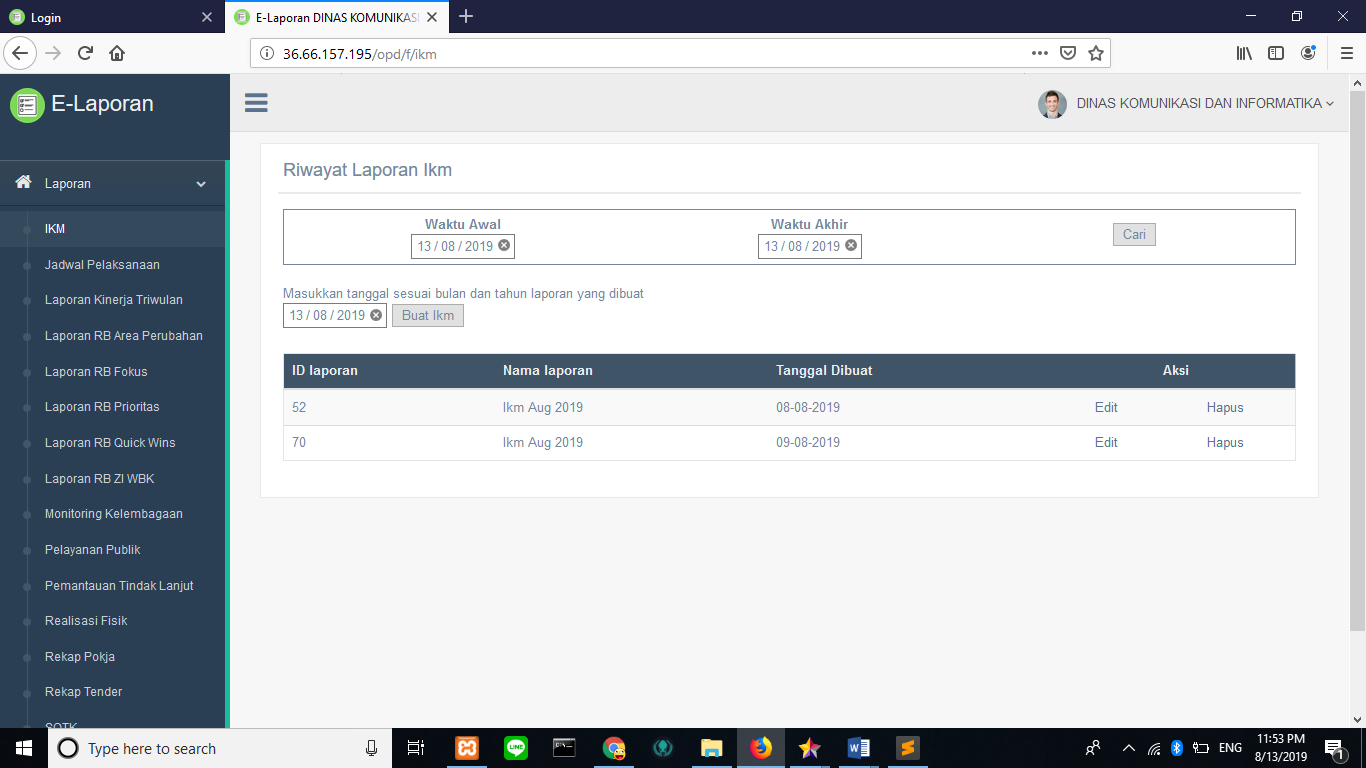
Halaman ini khusus untuk User OPD. Halaman ini merupakan halaman utama setelah User melakukan login. User OPD dapat membuat laporan yang diberi akses oleh Admin.



**Gambar 23 Halaman Home User**

* + 1. **Halaman Melihat Laporan**

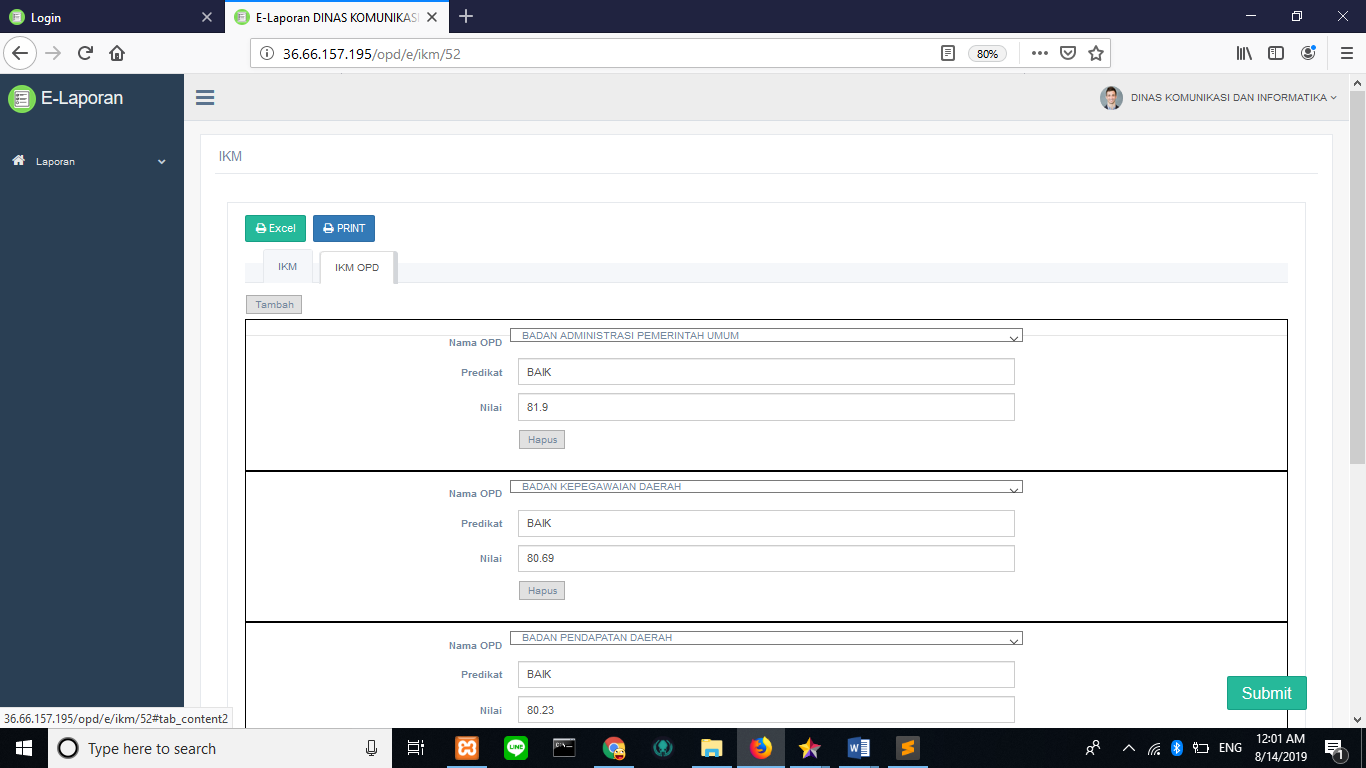
Halaman ini khusus untuk User OPD. Halaman ini merupakan halaman yang menampilkan data laporan yang dibuat oleh User OPD. User OPD dapat mencari laporan berdasarkan rentang waktu yang diinput, membuat laporan, mengedit laporan, dan menghapus laporan.



**Gambar 24 Halaman Melihat Laporan**

* + 1. **Halaman Mengedit Laporan**

Halaman ini khusus untuk User OPD. Halaman ini merupakan halaman untuk melakukan edit data laporan yang dipilih.



**Gambar 25 Halaman Mengedit Laporan**

*[Halaman ini sengaja dikosongkan]*

# BAB VI PENGUJIAN DAN EVALUASI

Pengujian dilakukan untuk memastikan kualitas perangkat lunak yang dikembangkan dan kesesuaian hasil eksekusi perangkat lunak dengan analisis dan perancangan perangkat lunak.

## Tujuan Pengujian

Pengujian dilakukan terhadap sistem *report manegement* untuk mengetahui beberapa hal berikut:

1. Menguji kesesuaian dan ketepatan fungsionalitas.
2. Memudahkan programmer selanjutnya untuk memahami kekurangan sistem.

## Lingkungan Pengujian

Spesifikasi perangkat keras dan perangkat lunak yang digunakan untuk pengujian yaitu :

|  |  |
| --- | --- |
| **Komponen** | **Spesifikasi** |
| CPU | Intel ® Core ™ i5-6200U CPU @2.30 GHz |
| RAM | 4.00 GB |
| Sistem Operasi | Windows 10 Pro 64-bit |
| Web Browser | Google Chrome, Mozilla Firefox |
| Keterangan | Sebagai Admin dan User OPD |

**Tabel 55 Lingkungan Pengujian**

## Identifikasi dan Rencana Pengujian

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| ***Kelas Uji*** | ***Butir Uji*** | ***Identifikasi*** | | ***Tingkat Pengujian*** | ***Jenis Pengujian*** | ***Jadwal*** |
| ***SKPL*** | ***PDHUPL*** |
| Melihat Data | Melihat Data OPD | F01 | PDHUPL-01 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Melihat Data User OPD | F05 | PDHUPL-02 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Melihat Data Tipe Laporan OPD | F12 | PDHUPL-03 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Melihat Log Aktifitas User OPD | F15 | PDHUPL-04 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Melihat Laporan | F17 | PDHUPL-05 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Pencari-an Data | Pencarian Data OPD | F02 | PDHUPL-06 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Pencarian Data User OPD | F07 | PDHUPL-07 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Pencarian Data Tipe Laporan OPD | F13 | PDHUPL-08 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Pencarian Data Log Aktifitas User OPD | F16 | PDHUPL-09 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Pencarian Riwayat Laporan | F18 | PDHUPL-10 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Penam-bahan Data | Penambahan User OPD | F06 | PDHUPL-11 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Pembuatan Laporan | F19 | PDHUPL-12 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Peruba-han Data | Melakukan Reset Password User OPD | F08 | PDHUPL-13 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Melakukan Edit Akses OPD Terhadap Laporan | F14 | PDHUPL-14 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Melakukan Edit Laporan | F20 | PDHUPL-15 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Pengha-pusan Data | Menghapus User OPD | F09 | PDHUPL-16 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Menghapus Laporan | F21 | PDHUPL-17 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |
| Mence-tak Laporan | Mencetak Laporan | F22 | PDHUPL-18 | Pengujian Sistem | *Black Box* | 22 Agustus 2019 |

**Tabel 56 Identifikasi dan Rencana Pengujian**

## Daftar Data Test

Pencarian Data:

|  |  |  |
| --- | --- | --- |
| Nama | **Field** | **Nilai** |
| Data Test-F01 | Search | dinas komunikasi |
| Data Test-G01 | Search | satuan polisi |
| Data Test-H01 | Search | badan kepegawaian |
| Data Test-I01 | Search | badan perencanaan |
| Data Test-I02 | Search Berdasarkan Tanggal | 08/15/2019 |
| Data Test-J01 | Tanggal Awal Pencarian  Tanggal Akhir Pencarian | 08/08/2019  09/09/2019 |

**Tabel 57 Data Test Pencarian Data**

Penambahan Data:

|  |  |  |
| --- | --- | --- |
| Nama | **Field** | **Nilai** |
| Data Test-K01 | Username  Password  OPD | Username  123456  Badan Administrasi Pemerintahan Umum |
| Data Test-K02 | Username  Password  OPD | -  -  Badan Administrasi Pemerintahan Umum |

**Tabel 58 Data Test Penambahan Data**

Perubahan Data:

|  |  |  |
| --- | --- | --- |
| Nama | **Field** | **Nilai** |
| Data Test-O01 | OPD  Predikat  Nilai | Badan Administrasi Pemerintah Umum  A  84 |

**Tabel 59 Data Test Perubahan Data**

## Deskripsi dan Hasil Uji

* + 1. **Melihat Data**
       1. **Melihat Data OPD**

Pre Kondisi:

* Pengguna adalah user dengan level Admin.
* Pengguna sudah melakukan prosedur login ke sistem.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-01 | Tidak ada masukan | * Pilih menu Database. * Pilih submenu OPD. | Kosong | Sistem menampilkan data OPD. | Pengguna dapat melihat data OPD. | Sistem menampilkan data OPD. | Diterima |

**Tabel 60 Melihat Data OPD**

* + - 1. **Melihat Data User OPD**

Pre Kondisi:

* Pengguna adalah user dengan level Admin.
* Pengguna sudah melakukan prosedur login ke sistem.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-02 | Tidak ada masukan | * Pilih menu Database. * Pilih submenu User. | Kosong | Sistem menampilkan data User OPD. | Pengguna dapat melihat data User OPD. | Sistem menampilkan data User OPD. | Diterima |

**Tabel 61 Melihat Data User OPD**

* + - 1. **Melihat Tipe Laporan OPD**

Pre Kondisi:

* Pengguna adalah user dengan level Admin.
* Pengguna sudah melakukan prosedur login ke sistem.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-03 | Tidak ada masukan | * Pilih menu Database. * Pilih submenu Tipe OPD. | Kosong | Sistem menampilkan data laporan yang dapat diakses oleh OPD. | Pengguna dapat melihat data laporan yang dapat diakses oleh OPD. | Sistem menampilkan data laporan yang dapat diakses oleh OPD. | Diterima |

**Tabel 62 Melihat Tipe Laporan OPD**

* + - 1. **Melihat Log Aktifitas User OPD**

Pre Kondisi:

* Pengguna adalah user dengan level Admin.
* Pengguna sudah melakukan prosedur login ke sistem.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-04 | Tidak ada masukan | * Pilih menu Query Log. * Pilih submenu Query Log. | Kosong | Sistem menampilkan data log aktifitas yang dilakukan User OPD. | Pengguna dapat melihat log aktifitas User OPD. | Sistem menampilkan data log aktifitas yang dilakukan User OPD. | Diterima |

**Tabel 63 Melihat Log Aktifitas User OPD**

* + - 1. **Melihat Laporan**

Pre Kondisi:

* Pengguna adalah user dengan level User OPD.
* Pengguna sudah melakukan prosedur login ke sistem.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-05 | Tidak ada masukan | * Pilih menu Laporan. * Pilih submenu laporan yang akan dilihat. | Kosong | Sistem menampilkan laporan yang dipilih oleh pengguna. | Pengguna dapat melihat laporan yang dipilih. | Sistem menampilkan laporan yang dipilih. | Diterima |

**Tabel 64 Melihat Laporan**

* + 1. **Pencarian Data**
       1. **Pencarian Data OPD**

Pre Kondisi:

* Pengguna adalah user dengan level Admin.
* Pengguna telah melakukan prosedur login ke sistem.
* Pengguna telah melihat data OPD.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-06 | Mencari data OPD dengan memasukkan parameter pencarian. | * Ketik parameter pencarian pada kolom Search. | Data Test-F01 | Sistem menampilkan data OPD yang sesuai dengan parameter pencarian pengguna. | Pengguna dapat mencari data OPD menggunakan parameter pencarian nama OPD atau kode user e-budgeting, atau kode user e-kinerja. | Sistem menampilkan data OPD yang sesuai dengan parameter pencarian pengguna. | Diterima |

**Tabel 65 Pencarian Data OPD**

* + - 1. **Pencarian Data User OPD**

Pre Kondisi:

* Pengguna adalah user dengan level Admin.
* Pengguna telah melakukan prosedur login ke sistem.
* Pengguna telah melihat data User OPD.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-07 | Mencari data User OPD dengan memasukkan parameter pencarian. | * Ketik parameter pencarian pada kolom Search. | Data Test-G01 | Sistem menampilkan data User OPD yang sesuai dengan parameter pencarian pengguna. | Pengguna dapat mencari data User OPD menggunakan parameter pencarian nama OPD atau username. | Sistem menampilkan data User OPD yang sesuai dengan parameter pencarian pengguna. | Diterima |

**Tabel 66 Pencarian Data User OPD**

* + - 1. **Pencarian Data Tipe Laporan OPD**

Pre Kondisi:

* Pengguna adalah user dengan level Admin.
* Pengguna telah melakukan prosedur login ke sistem.
* Pengguna telah melihat data tipe laporan OPD.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-08 | Mencari data tipe laporan OPD dengan memasukkan parameter pencarian. | * Ketik parameter pencarian pada kolom Search. | Data Test-H01 | Sistem menampilkan data tipe laporan OPD yang sesuai dengan parameter pencarian pengguna. | Pengguna dapat mencari data tipe laporan OPD menggunakan parameter pencarian nama OPD, atau kode user e-budgeting, atau kode user e-kinerja, atau kode user e-monev. | Sistem menampilkan data tipe laporan OPD yang sesuai dengan parameter pencarian pengguna. | Diterima |

**Tabel 67 Pencarian Data Tipe Laporan OPD**

* + - 1. **Pencarian Data Log Aktifitas User OPD**

Pre Kondisi:

* Pengguna adalah user dengan level Admin.
* Pengguna telah melakukan prosedur login ke sistem.
* Pengguna telah melihat log aktifitas User OPD.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-09 | Mencari data log aktifitas User OPD dengan memasukkan parameter pencarian. | * Ketik parameter pencarian pada kolom Search. | Data Test-I01 | Sistem menampilkan data log aktifitas User OPD yang sesuai dengan parameter pencarian pengguna. | Pengguna dapat mencari data log aktifitas User OPD menggunakan parameter pencarian nama OPD, atau username, atau aksi. | Sistem menampilkan data log aktifitas User OPD yang sesuai dengan parameter pencarian pengguna. | Diterima |
| PDHUPL-09 | Mencari data log aktifitas User OPD berdasarkan tanggal. | * Pilih tanggal pencarian. * Klik Cari. | Data Test-I02 | Sistem menampilkan data log aktifitas User OPD yang sesuai dengan tanggal yang dipilih pengguna. | Pengguna dapat mencari log aktifitas User OPD berdasarkan tanggal. | Sistem menampilkan data log aktifitas User OPD yang sesuai dengan tanggal yang dipilih pengguna. | Diterima |

**Tabel 68 Pencarian Data Log Aktifitas User OPD**

* + - 1. **Pencarian Riwayat Laporan**

Pre Kondisi:

* Pengguna adalah user dengan level User OPD.
* Pengguna telah melakukan prosedur login ke sistem.
* Pengguna telah melihat laporan.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-10 | Mencari riwayat laporan berdasarkan rentang tanggal yang dimasukkan. | * Pilih tanggal awal pencarian. * Pilih tanggal akhir pencarian. * Klik Cari. | Data Test-J01 | Sistem menampilkan riwayat laporan dari tanggal awal sampai tanggal akhir. | Pengguna dapat mencari riwayat laporan berdasarkan rentang tanggal yang dimasukkan. | Sistem menampilkan riwayat laporan dari tanggal awal sampai tanggal akhir. | Diterima |

**Tabel 69 Pencarian Riwayat Laporan**

* + 1. **Penambahan Data**
       1. **Penambahan User OPD**

Pre Kondisi:

* Pengguna adalah user dengan level Admin.
* Pengguna telah melakukan prosedur login ke sistem.
* Pengguna telah melihat data User OPD.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-11 | Data lengkap | * Klik tombol Tambah User. * Isi data username, password, dan pilih OPD. * Klik Submit. | Data Test-K01 | Penambahan User OPD berhasil. | Pengguna dapat menambah User di tiap OPD. | Muncul pesan “Tambah User Berhasil”. | Diterima |
| PDHUPL-03 | Data tidak lengkap | Data Test-K02 | Muncul pesan kesalahan data masukan tidak lengkap. | Pengguna tidak dapat melakukan penambahan User OPD jika data yang dimasukkan tidak lengkap. | Muncul pesan “Please fill out this field” pada kolom yang belum diisi. | Diterima |

**Tabel 70 Penambahan User OPD**

* + - 1. **Pembuatan Laporan**

Pre Kondisi:

* Pengguna adalah user dengan level User OPD.
* Pengguna telah melakukan prosedur login ke sistem.
* Pengguna telah melihat laporan.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-12 | Tidak ada masukan | * Pilih tanggal laporan dibuat. * Klik Buat. | Kosong | Laporan berhasil dibuat. | Pengguna dapat membuat laporan. | Sistem menampilkan pesan “Berhasil membuat laporan”. | Diterima |

**Tabel 71 Pembuatan Laporan**

* + 1. **Perubahan Data**
       1. **Melakukan Reset Password OPD**

Pre Kondisi:

* Pengguna adalah user dengan level Admin.
* Pengguna telah melakukan prosedur login ke sistem.
* Pengguna telah melihat data User OPD

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-13 | Tidak ada masukan | * Klik tombol Reset Pass. | Kosong | User yang dipilih telah di reset password menjadi password default. | Pengguna dapat melakukan reset password User OPD menjadi password default. | Sistem menampilkan pesan “Reset Password Berhasil”. | Diterima |

**Tabel 72 Melakukan Reset Password OPD**

* + - 1. **Melakukan Edit Akses OPD Terhadap Laporan**

Pre Kondisi:

* Pengguna adalah user dengan level Admin.
* Pengguna telah melakukan prosedur login ke sistem.
* Pengguna telah melihat data tipe laporan OPD.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-14 | Tidak ada masukan | * Klik tombol plus pada OPD yang dipilih. * Klik tombol Edit Akses. * Pilih tipe laporan yang akan di enable. * Klik Simpan. | Kosong | OPD mempunyai akses terhadap laporan yang di enable. | Pengguna dapat mengedit akses OPD terhadp laporan. | Sistem menampilkan pesan “Edit Akses OPD Berhasil”. | Diterima |

**Tabel 73 Melakukan Edit Akses OPD Terhadap Laporan**

* + - 1. **Melakukan Edit Laporan**

Pre Kondisi:

* Pengguna adalah user dengan level User OPD.
* Pengguna telah melakukan prosedur login ke sistem.
* Pengguna telah melihat laporan.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-15 | Data laporan IKM. | * Klik tombol Edit. * Isi data laporan yang di edit. * Klik Submit. | Data Test-O01 | Laporan berhasil di edit. | Pengguna dapat melakukan edit laporan yang sudah dibuat. | Sistem menampilkan pesan “Berhasil mengubah laporan”. | Diterima |

**Tabel 74 Melakukan Edit Laporan**

* + 1. **Penghapusan Data**
       1. **Menghapus User OPD**

Pre Kondisi:

* Pengguna adalah user dengan level Admin.
* Pengguna telah melakukan prosedur login ke sistem.
* Pengguna telah melihat data User OPD.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-16 | Tidak ada masukan | * Klik Hapus | Kosong | Data User OPD telah dihapus dari database. | Pengguna dapat menghapus data User OPD yang dipilih. | Sistem menampilkan notifikasi “Data Telah Dihapus”. | Diterima |

**Tabel 75 Menghapus User OPD**

* + - 1. **Menghapus Laporan**

Pre Kondisi:

* Pengguna adalah user dengan level User OPD.
* Pengguna telah melakukan prosedur login ke sistem.
* Pengguna sudah melihat laporan.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-17 | Tidak ada masukan | * Klik hapus. | Kosong | Data laporan dihapus dari database. | Pengguna dapat memilih laporan yang akan dihapus. | Sistem menampilkan pesan “Berhasil menghapus laporan”. | Diterima |

**Tabel 76 Menghapus Laporan**

* + 1. **Mencetak Laporan**
       1. **Mencetak Laporan**

Pre Kondisi:

* Pengguna adalah user dengan level User OPD.
* Pengguna telah melakukan prosedur login ke sistem.
* Pengguna sudah membuat laporan atau mengedit laporan.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Identifikasi** | **Deskripsi** | **Prosedur Pengujian** | **Masukan** | **Keluaran yang Diharapkan** | **Kriteria Evaluasi Hasil** | **Hasil yang Didapat** | **Kesimpulan** |
| PDHUPL-18 | Tidak ada masukan, mencetak format PDF | * Klik Print | Kosong | Sistem menampilkan laporan dalam format PDF. | Pengguna dapat mencetak laporan denga format PDF. | Sistem menampilkan laporan dalam format PDF. | Diterima |
| PDHUPL-18 | Tidak ada masukan, mencetak format Excel | * Klik Excel | Kosong | Sistem membuat laporan dalam format Excel. | Pengguna dapat mencetak laporan denga format Excel. | Sistem melakukan download laporan dalam format Excel. | Diterima |

**Tabel 77 Mencetak Laporan**

*[Halaman sengaja dikosongkan]*

# BAB VII KESIMPULAN dan SARAN

## Kesimpulan

Kesimpulan dari hasil perancangan sistem *report management* berbasis web untuk Dinas Komunikasi dan Informatika (Kominfo) Kota Madiun sebagai berikut :

* Sistem yang dibangun berguna untuk memudahkan Organisasi Perangkat Daerah (OPD) di Kota Madiun dalam pembuatan laporan.
* Format laporan tiap OPD di Kota Madiun dapat diseragamkan.
* Kemudahan akses laporan yang dibuat oleh OPD di Kota Madiun.
* Kemudahan pengelolaan data dari aplikasi lain yang digunakan oleh OPD di Kota Madiun.

## Saran

Dalam pembuatan sistem *report management* memiliki banyak kekurangan. Adapun saran untuk pengembangan sistem *report management* sebagai berikut:

* Fungsi *backup* dan *restore* data API yang digunakan untuk antisipasi ketika server dari API tidak bisa diakses masih belum bisa.
* Perbaikan *style* ketika melakukan *export* laporan dalam bentuk excel.
* Perbaikan form penambahan data laporan karena data yang dimasukkan akan hilang ketika User OPD klik tambah dan belum klik tombol submit.
* Perbaikan form data laporan yang terlalu panjang ketika data laporan banyak, sehingga menyusahkan User OPD ketika harus *scroll* data.
* Ada beberapa data yang seharusnya bisa daimbil dari API tapi belum diimplementasikan ke dalam sistem, yaitu: data realisasi laporan kinerja triwulan yang seharusnya diambil dari lra, data realisasi laporan realisasi fisik yang seharusnya diambil dari lra, dan data nilai hps diambil dari rup.

# DAFTAR PUSTAKA

[1] What is CodeIgniter. [Online]

Available

<https://github.com/bct-ci/CodeIgniter>

[Accessed 10 July 2019]

[2] PHP Manual. [Online]

Available

<https://www.php.net/manual/en>

[Accessed 10 July 2019]

[3] Crontab Guru. [Online]

Available

<https://crontab.guru>

[Accessed 10 July 2019]

[4] Dinas Komunikasi dan Informatika. [Online]

Available

<http://kominfo.madiunkota.go.id>

[Accessed 10 July 2019]

# BIODATA PENULIS I

Nama : Marde Fasma Ul’aza

Tempat, Tanggal Lahir : Surabaya, 06 Agustus 1998

Jenis Kelamin : Laki-laki

Agama : Islam

Status : Belum Menikah

Alamat Asal : Jl. Pejajaran No. 1, Winongo, Mangunharjo Kota Madiun

Alamat Surabaya : Jl. Keputih Gg. Nur Hasyim No. 30A

Telepon : 089615125347

Email : mardefasma123up@gmail.com

PENDIDIKAN *FORM*AL

2016 – Sekarang : Mahasiswa S1 Informatika ITS

2013 – 2016 : SMAN 3 Kota Madiun

2010 – 2013 : SMPN 1 Kota Madiun

2004 – 2010 : SDN 1 Winongo

KEMAMPUAN

AKADEMIS

Kuliah : Departemen Informatika – Fakultas Teknologi  
 Informasi dan Komunikasi, Institut Teknologi   
 Sepuluh Nopermber Surabaya.

Angkatan : 2016

Semester : 6

IPK : 3,63

# BIODATA PENULIS II

Nama : Ulfatuz Zahroh

Tempat, Tanggal Lahir : Ngawi, 04 Februari 1999

Jenis Kelamin : Perempuan

Agama : Islam

Status : Belum Menikah

Alamat Asal : Soco, Jogorogo, Ngawi, Jawa Timur

Alamat Surabaya : Jl. Keputih Makam Blok D No. 20, Sukolilo, Surabaya

Telepon : 085735531270

Email : ulfatuzzahroh2@gmail.com

PENDIDIKAN *FORM*AL

2016 – Sekarang : Mahasiswa S1 Informatika ITS

2013 – 2016 : MA Darul Huda Ponorogo

2010 – 2013 : MTs Darul Huda Ponorogo

2004 – 2010 : SDN Soco 1

KEMAMPUAN

AKADEMIS

Kuliah : Departemen Informatika – Fakultas Teknologi  
 Informasi dan Komunikasi, Institut Teknologi   
 Sepuluh Nopermber Surabaya.

Angkatan : 2016

Semester : 6

IPK : 3,31