

Dokumentasi Fitur dan kekurangan

Saran pada saat testing : https://docs.google.com/document/d/11ete6sgNJiESLzCLalxAxs2g5P9feF_TjRtxVoJKuvs/edit?usp=sharing

Note: `public const HOME = '/dashboard';`

Perbaiki itu ada di routeserviceprovider-> jika sudah login ,dan Kembali ke halaman utama, maka redirect kedashboard, padahal dashboard hanya ada di fitur admin

Fitur (1)	Controller (2)	Views (3)	Catatan (4)
LandingPage	Via web.php	Landingpage.index	-Masih Statis->perlu pengelolaan di admin
Login	Via web.php & loginController	Login	-Masih menggunakan Oauth Google
Dashboard	Dashboard	dashboard	✓
CRUD User	User	Users.index	Perlu adanya validasi file import excel
Crud TPI	TPI	Tim.index	Ketika AT di import dan sudah ada di database maka bisa error Db anggota perlu informasi tahun, agar dinamis hasOne nya
	Pengawasan	Tim.pengawasan	✓
CRUD Persyaratan	Persyaratan	persyaratan.index	Perlu adanya validasi file import excel
CRUD Pertanyaan LKE			-refresh and stay postion belum ada -accordion parent belum ada -jika memilih Kembali opsi yang ada di DB masih menimpa <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;"> Ya, Sesuai </div> <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;"> Tidak </div> <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;"> Ya, </div> <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;"> Tidak </div> -opsi input jika tidak diisi error

Fitur	Controller	Views	Catatan
(1)	(2)	(3)	(4)
CRUD Rincian Hasil	RincianHasil	lke.hasil.index	Opsi jika diluar pilihan maka akan error pada saat import
Monitoring	monitoring	Monitoring.index	✓
		Monitoring.catatan	✓
		Monitoring.LHE	✓
SelfAssessment	LKE	selfAssessment.index	✓
		selfAssessment.LKE	Masih ada yang menggunakan App\Models (3 line)
		selfAssessment.pertanyaan	Masih ada yang menggunakan App\Models (2 line)
	Rekapitulasi	self-assessment.surat	Masih ada yang menggunakan App\Models (2 line)
Penilaian Pendahuluan	EvaluatorProvinsi	EvalProv.index	✓
	SuratPersetujuanProvinsi	EvalProv.surat	Jika status rekap sudah >4 , maka akan ke refresh jika di upload ulang suratnya, solusi :-> waktu evaluasi
Desk Evaluation	DeskEvaluation	Tpi.index	Hubungan antara pengawasan dan rekapitulasi yaitu melalui satker_id(masalah jika berbeda tahun)
		Tpi.lke	Masih ada yang menggunakan App\Models (5 line) ✓
		Tpi.pertanyaan	Ditambahkan baris @if (\$selfAssessment != null) Agar TPI hanya menilai satker yang ngisi saja ✓
	LHE	Tpi.lhe	Upload LHE, dan pastikan download sudah sesuai dengan format, ada kendala/kebingungan di Field catatan template

Penjelasan Fitur

- **Create**

```
<form method="post" action="/users">
    @csrf
```

- **Update**

```
<form method="post" action="users/{ { $user->id } }">
    @method('put')
    @csrf
```

- **Delete**

```
<form action="/users/{ { $user->id } }" method="POST" class="d-inline">
    @method('delete')
    @csrf
```

- **app/imports/Usersimport**

->method ini terdiri dari 2 function yaitu model(membaca input yang masuk berdasarkan judul tabel dan rules(validasi input yang masuk))

- **HasOne**

-> bisa dipilih sekali setahun sekali, tambahkan informasi tersebut

```
return $this->hasOne(TPI::class, 'ketua_tim')->where('tahun', date('Y'));
```

- **ImportExcel**

Logika:

-gunakan library dari Laravel : <https://docs.laravel-excel.com/3.1/imports/multiple-sheets.html>

-buat template dengan table yang memiliki header

-terdapat dua function didalam class app/imports/nama_class, yaitu model(ambil data based on header) dan rules(aturan)

- **Generata PHPWord :** <https://github.com/PHPOffice/PHPWord>

Logika:

-buat template docx

-masukkan variabel \${data}

-untuk bentuk table, lakukan perulangan dan masukkan nilai tersebut pada array

- **Kirim Notifikasi Gmail**

Logika:

-atur alamat email pengirim di .env

-Buat 1 function yang mengarahkan pada app/Mail/{namaEmail} yang memiliki template blade

-kirim gmail menuju orang tertentu dan \$data yang telah di definisikan.

Penjelasan Controller dan kegunaan

Controller 1	Function 2	Penjelasan 3	Database/Function 4
GoogleController	redirectToGoogle	Untuk menuju halaman https://accounts.google.com/	User(R)
	handleGoogleCallback	Validasi login by gmail, Ketika menggunakan Oauth Google	
	login	Validasi login manual	
	logout	Keluar dari sistem	
DashboardController	index	Untuk menuju halaman dashboard admin yang terdiri dari jumlah data yang bisa dikelola oleh admin	Tpi,user, persyaratan, pertanyaan, rekaphasil, rekapitulasi(R)
UserController	index	Menuju halaman kelola pengguna	User(CRUD),level&satker(R)
	store	-Menyimpan data pengguna dengan validasi (email&id:unique)	
	import	-terdapat fitur download template -Create data baru -Update data lama jika id sama -halaman menuju app/imports/UsersImport	
	update	Mengubah data pengguna	
	Destroy	Hapus data pengguna by id	
TPIController	index	-Menuju halaman kelola data TPI -Ketua tim dan anggota hanya bisa dipilih sekali (HasOne) <pre>User::doesntHave('ketua') -> controller public function ketua() ->Models { return \$this->hasOne(TPI::class, 'ketua_tim'); }</pre>	tpi, dalnis, ketua_tim, anggota (R)
	store	-Menyimpan data tpi sesuai validasi	TPI & anggota_tpi(C)

Controller	Function	Penjelasan	Database/Function
1	2	3	4
	import	-terdapat fitur download template -Create data baru -Update data lama jika id sama-> hapus data anggota , baru create lagi -halaman menuju app/imports/TPIImport -import maksimal 3 anggota tim	
	show	Menuju halaman wilayah tugas berdasarkan id TPI	Tpi,pengawasan dan satker(R)
	update	-Bisa mengubah atribut id->logika:hapus id lama, buat yang baru -Jika tidak mengubah id, maka bisa update data selain id.	TPI & anggota_tpi(U)
	destroy	menghapus data TPI	TPI(D)
PengawasanController	store	-Bisa menyimpan multiple satker pengawasan dari anggota TPI -satker hanya bisa dipilih sekali (hasOne)	Pengawasan(C)
	update	Fungsi ini digunakan oleh TPI untuk melakukan update table pengawasan	Pengawasan(U) DLEmailDL(email)
	destroy	Menghapus data pengawasan by id	Pengawasan(D)
PersyaratanController	index	-Menuju halaman persyaratan wbk/wbbm -satker memiliki hasOne(hanya bisa dipilih sekali)	Persyaratan & satker(R)
	Store	-menyimpan multiple satker data persyaratan	Persyaratan (CU)
	Import	-terdapat fitur download template -Create data baru -Update data lama jika id sama-> hapus data anggota , baru create lagi -halaman menuju app/imports/PersyaratanImport	
	Update	-melakukan update pada id tertentu	Persyaratan (U)
	Destroy	Menghapus data pada id tertentu	Persyaratan (D)
RincianController	Store	Menyimpan data rincian LKE (Lv1), jika id sama, maka ditambah satu huruf keatas	Rincian(C)
	Update	Mengubah data rincian LKE by ID (LV1) , pengubahan belum mengubah ID, jika katanya diubah	Rincian(U)
	Destroy	Menghapus data rincian LKE by ID(LV1)	Rincian(D)
SubRincianController	Store	Menyimpan data subrincian LKE (Lv2) , jika id sama, maka ditambah satu huruf keatas	SubRincian(C)
	Update	Mengubah data subrincian LKE by ID (LV2), pengubahan belum mengubah ID, jika katanya diubah	SubRincian (U)
	Destroy	Menghapus data subrincian LKE by ID(LV2)	SubRincian (D)

Controller	Function	Penjelasan	Database/Function
1	2	3	4
PilarController	Store	Menyimpan data Pilar LKE (Lv3)	Pilar(C)
	Update	Mengubah data Pilar LKE by ID (LV3)	Pilar (U)
	Destroy	Menghapus data Pilar LKE by ID(LV3)	Pilar (D)
SubPilarController	Store	-Menyimpan data SubPilar LKE (Lv4) -penambahan bobot ada pada controller ini -logika:bobot+bobotlama(bobot yang ada di database)	SubPilar (C)
	Update	Mengubah data SubPilar LKE by ID (LV4) -logika: cek apakah ada perubahan pada nilai bobot, jika ada maka nilai bobot baru dikurangkan dengan nilai bobot lama, dan lakukan update pada lv diatasnya	SubPilar (U)
	Destroy	Menghapus data SubPilar LKE by ID(LV4) -logika: ambil nilai bobot pada subpilar yang ingin dihapus, Lakukan pengurangan dengan nilai yang ada di database untuk lv diatasnya	SubPilar (D)
PertanyaanController	index	-Menuju halaman pertanyaan LKE, model yang diambil adalah rincian -rincian akan di foreach sampai level pertanyaan.	Rincian(R)
	Store	-menyimpan pertanyaan, opsi, dan dokumen LKE -opsi dibedakan menjadi input dan checkbox, keduanya dilakukan foreach berdasarkan banyak baris yang masuk (rincian[] dan input[]) -opsi input memiliki bobot 1 -opsi checkbox memiliki bobot Ya/Tidak ->(1, 0) , A/B/C ->(1,0.5,0) , A/B/C/D -> (1,0.67,0.33,0) , A/B/C/D/E -> (1,0.75, 0.50, 0.25, 0) -dokumen dilakukan foreach berdasarkan banyaknya dokumen yang masuk (dokumen[]) -untuk opsi dan dokumen memainkan peran JS	Pertanyaan(C), Opsi(C), dokumenLKE(C)
	show	Menuju halaman create pertanyaan by id subpilar	subPilar(R)
	Edit	Menujut halaman update pertanyaan by id subpilar	Subpilar(R), Pertanyaan, dokumen, opsi, opsiInput & checkbox(R)
	Update	Logika: hapus semua opsi dan dokumenLKE Khusus untuk checkbox, lihat apakah hanya mengubah isi atau mengubah opsi (misal dari opsi ya/tidak -> a/b/c) hal tersebut memiliki fungsi yang berbeda, jika tidak dibedakan maka akan kebaca double	Pertanyaan(U), Opsi(U), dokumenLKE(U)
	Delete	Menghapus pertanyaan by id	

Controller	Function	Penjelasan	Database/Function
1	2	3	4
RincianHasilController	index	-Menuju halaman kelola rincian hasil LKE -Rincianhasil memasukkan nilai by pilar (terdapat 3 pilar yang diisi oleh admin) -Satker memiliki HasOne(hanya bisa dipilih sekali) -RekapHasil memiliki 2 read (hasilSatker->groupby satker, dan hasil->all data rekapHasil)	Pilar(R), RekapHasil(R), Satker(R)
	Store	-Bisa create mutiple satker (perulangan satker dilanjutkan pilar) -melakukan perulangan berdasarkan 3 pilar - logika nilai =bobot_pilar * bobot_opsi	RekapHasil(C)
	Import	-terdapat fitur download template -Create data baru -Update data lama jika id sama -perulangan pada pilar	RekapHasilImport
	Update	-perulangan pada pilar - logika nilai =bobot_pilar * bobot_opsi -Untuk menampilkan data pada view, maka dipanggil id_rekaphasil berdasarkan pilar	RekapHasil(U)
	Destroy	-menghapus data rekapHasil by ID	RekapHasil(D)
Monitoring	index	-Menuju halaman monitoring ZI	Rekapitulasi, RekapPengungkit, RekapHasil(R)
	lhe	-Menuju halaman rekap nilai by rekapitulasi_id -perulangan mulai dari rincian -RekapHasil memiliki pilar_id - Nilai: ->rincian :tambah semua nilai pada RincianPengungkit ->subRincian: tambahkan nilai yang sesuai dengan subrincian_id (pakai where 'like') ->pilar:tambahkan nilai sesuai dengan pilar_id ->subPilar: penjumlahan dari nilai pertanyaan, yang didapat dari nilaiSA * penimbang Penimbang =bobot_subpilar/jumlah_pertanyaan ->pertanyaan = nilai dari SA	Rekapitulasi, RekapPengungkit, RekapHasil, Rincian (R)
	catatan	-Menuju halaman catatan TPI by rekapitulasi_id - perulangan mulai dari rincian hingga pertanyaan -ketika dapat pertanyaan maka cari catatan berdasarkan deskevaluation	Rekapitulasi,Rincian, DeskEvaluation (R)
	destroy	Menghapus data rekapitulasi	Rekapitulasi (D)

Controller	Function	Penjelasan	Database/Function
1	2	3	4
LKEController	index	<ul style="list-style-type: none"> -Menuju halaman dashboard pengajuan ZI -Melihat apakah bisa mengajukan ZI(persyaratan) berdasarkan tahun ->logika:tombol pengajuan hanya bisa diakses jika jumlah rekap =0 atau pengajuan terakhir (tahun) != pengajuan sekarang (tahun) -Melihat status Rekap (Rekapitulasi) -Melihat status pengawasan satker (Pengawasan) 	Persyaratan, Rekapitulasi, Pengawasan (R)
	store	<ul style="list-style-type: none"> -Menyimpan data rekapitulasi dengan id acak (UUID) -Menyimpan data LHE dengan id_rekap 	Rekapitulasi (C) LHE(C)
	show	<ul style="list-style-type: none"> -Menuju halaman LKE dengan id_rekap -LKE dapat diakses oleh dua aktor yaitu EvalProv dan PIC satker -menghitung jumlah soal (Pertanyaan) -Menghitung soal yang terjawab oleh satker (SelfAssessment) -Menghitung soal yang terjawab oleh TPI (DeskEvaluation)-> terbuka ketika tahapan status di 5,6,dan 7 -Foreach Subrincian hingga pertanyaan (subrincian) -Membaca data pilar pda rincianHasil (pilar) -mengambil nilai pengungkit berdasarkan id_rekap (RekapPengungkit) -mengambi nilai hasil berdasarkan tahun dan satker_id (RekapHasil) Progress jawab = $(\text{tot_soal_terjawab} * 100) / \text{tot_jumlah_soal}$ -jumlah_soal, soal_terjawab, dan progress dikalkulasikan berdasarkan total keseluruhan dan total pilar -Pewarnaan pada card disesuaikan berdasarkan 3 warna (belum mengisi, sudah mengisi minimal 1 pertanyaan, dan nilai telah melewati Batasan minimal) -Fitur lihat perubahan dapat diakses Ketika status lv 5 ->logika : baris akan ceklis ketika nilai_SA berbeda dengan nilai_DL atau nilai_SA=0 -Jika nilai rekappengungkit belum > nilai minimal pilar, maka tidak bisa kirim -> logika: mengambil data rekappengungkit, serta nilai minimal pilar. Jadi jika ingin bisa kirim LKE maka nilai rekappengungkit harus lebih besar dari nilai minimal pilar (untuk semua pilar , terdapat 12 pilar). 	Rekapitulasi, pertanyaan, selfAssessment, DeskEvaluation, SubRincian, Pilar, RekapPengungkit, RekapHasil (R)
	pertanyaan	<ul style="list-style-type: none"> -menuju halaman pertanyaan berdasarkan id_pilar (Pilar) -terdapat accordion berdasarkan subPilar (SubPilar) -terdapat hasil DeskEvaluation dari TPI , hanya bisa diakses Ketika status di 5,6,7 (DeskEvaluation) 	Pilar, SubPilar, DeskEvaluation , Rekapitulasi(R)

Controller	Function	Penjelasan	Database/Function
1	2	3	4
		-perhitungan nilai dilakukan pada level subpilar Penimbang = bobot/jumlah pertanyaan nilaiPertanyaan = nilai*penimbang (nilai = nilai dari opsi_id) nilaiSubpilar = jumlah dari NilaiPertanyaan	
	update	-Fungsi ini digunakan oleh BPS provinsi ketika kirim LKE atau kab/kota yang dapat tindak lanjut dari Tpi -terdapat service kirim gmail -PIC provinsi akan mengirimkan gmail kepada evalprov dan status menjadi 8(Cetak surat pengantar prov) -sedangkan PIC kab/kota mengirimkan gmail kepada TPI dan status menjadi 4(penilaian TPI)	SAEmail Rekapitulasi (U)
SelfAssessment Controller	Store	SelfAssessment -terdapat fungsi untuk scroll pada id tertentu (pertanyaan_id) -membedakan fungsi untuk input value dan checkbox, perbedaan terdapat pada nilai opsi_id(khusus input nilainya adalah “-“) dan nilai(input value nilainya 1) Input value -melakukan create secara manual dengan mengecek jumlah input value Logika untuk mendapatkan data nilai ->jika 2 : nilai2 /nilai1 ->jika 5 :nilai 4/(nilai1+2+3), dengan (nilai 1+2+3) tidak lebih dari nilai 4 ->jika 3 →cek jika id_pertanyaan ='PRC3A',maka nilai2/nilai1 Jika tidak, nilai 2/(nilai1+nilai 2) -Semua input value dan hasil nilai akan dimasukkan pada tabel InputField -terdapat inputvalue yang readonly , cara mengisinya adalah melakukan update pada opsi0 dengan nilai yang telah dihitung dokumenWajib -melakukan perulangan berdasarkan dokumenlke_id yang diupload (name ='dokumen[]') -FKnya adalah dokumenlke_id yang didapat dengan membaca perulangan based on id, menangkap dokumenlke_id <pre>\$request->input('id' . \$key)</pre> dokumenTambahan -melakukan multiple upload berdasarkan file yang diunggah (fileCreate[]) -Semua dokumen disimpan pada folder dokumen/ tahun / satker_id/pertanyaan_id	SelfAssessment,Upload Dokumen,InputField,R ekapPengungkit(C)

Controller	Function	Penjelasan	Database/Function
1	2	3	4
		RekapPengungkit -menyimpan nilai berdasarkan pilar $\$penimbang = \text{bobot/jumlah pertanyaan (pada subpilar)}$ Cek jika ada nilai lama di DB,maka $\$total = \text{nilai (nilai dari opsi_id) * penimbang} + \text{nilaiLama (total nilai based on pilar)}$ Jika tidak ada nilai lama , maka $\$total = \text{nilai (nilai dari opsi_id) * penimbang}$	
	Update	SelfAssessment Logikanya sama, hanya mengganti nilai, opsi_id, dan catatan(checkbox), nilai,catatan(input) InputField Logikanya sama, hanya mengganti input_sa, dokumenWajib logika sama, jika ada yang update maka akan hapus file lama dokumenTambahan -terdapat create dan update file, Jika create maka logika sama seperti di store, jika update maka akan update pada file tertentu dengan menghilangkan file lamanya RekapPengungkit $\$total = \text{nilaiSA_baru(nilai dari opsi_id)*penimbang} + \text{nilaiLama} - (\text{nilaiSALama*penimbang})$ ->Kenapa langsung ada nilai_lama, dikarenakan jika create perlu melihat nilai lamanya dulu terlebih dahulu , sedangkan update sudah ada nilai lama based on create	SelfAssessment,Upload Dokumen,InputField,RekapPengungkit(U)
	Destroy	Menghapus UploadDokumen berdasarkan id_uploaddokumen	
RekapitulasiController	Store	-Menyimpan surat pengantar BPS kab/kota -Fitur kirim gmail kepada evalprov -penamaan file ='satker_id-nama_surat' dan disimpan pada 'surat_pengantar/kabkota/\$tahun/\$namafile -Update rekapitulasi dengan status 1	SAEmail Rekapitulasi(U) LHE(U)
	Show	-Menuju halaman upload surat pengantar bps kab/kota -Mengambil total nilai pengungkit (RekapPengungkit) -Melakukan perulangan rincian hingga pertanyaan (Rincian)	Rekapitulasi, RekapPengungkit, , Rincian(R)
	Catatan	Mirip fitur monitoring	
	rekap	Mirip fitur monitoring	

Controller	Function	Penjelasan	Database/Function																												
1	2	3	4																												
	Cetak	<div><div><div>-Generate template otomatis</div><div>-Masukkan informasi no_surat</div><div>-terdapat perulangan pada nilaiPengungkit dan Hasil (harus bentuk table)</div></div><table><tr><th>No.</th><th>Jenis Komponen</th><th>Bobot</th><th>Hasil Self Assessment</th></tr><tr><th>(1)</th><th>(2)</th><th>(3)</th><th>(4)</th></tr><tr><td colspan="2">PENGUNGKIT</td><td>60.00</td><td>\$(total_sa)</td></tr><tr><td colspan="2">PEMENUHAN</td><td>30.00</td><td>\$(nilaiP_sa)</td></tr><tr><td>\$(no)</td><td>\$(pilar)</td><td>\$(bb)</td><td>\$(hasil_sa)</td></tr><tr><td colspan="2">REFORM</td><td>30.00</td><td>\$(nilaiR_sa)</td></tr><tr><td>\$(no)</td><td>\$(pilar)</td><td>\$(bb)</td><td>\$(hasil_sa)</td></tr></table><div></div></div> <div>-penamaan docs ‘pengantar \$satker’</div>	No.	Jenis Komponen	Bobot	Hasil Self Assessment	(1)	(2)	(3)	(4)	PENGUNGKIT		60.00	\$(total_sa)	PEMENUHAN		30.00	\$(nilaiP_sa)	\$(no)	\$(pilar)	\$(bb)	\$(hasil_sa)	REFORM		30.00	\$(nilaiR_sa)	\$(no)	\$(pilar)	\$(bb)	\$(hasil_sa)	
	No.	Jenis Komponen	Bobot	Hasil Self Assessment																											
	(1)	(2)	(3)	(4)																											
	PENGUNGKIT		60.00	\$(total_sa)																											
	PEMENUHAN		30.00	\$(nilaiP_sa)																											
\$(no)	\$(pilar)	\$(bb)	\$(hasil_sa)																												
REFORM		30.00	\$(nilaiR_sa)																												
\$(no)	\$(pilar)	\$(bb)	\$(hasil_sa)																												
	Rekap2	Rekapitulasi hingga level subpilar																													
	Rekap3	Rekapitulasi hingga level pilar																													
	Rekap4	Rekapitulasi hingga level pertanyaan																													
EvaluasiProvinsi	index	<div><div>-menuju halaman dashboard pengajuan ZI</div><div>-melihat rekapitulasi berdasarkan satker_id dan tahun</div><div>-melihat status pengawasan</div><div>-terdapat tombol cetak surat pengantar provinsi</div></div>	Rekapitulasi,Pengawasan(R)																												
	show	<div><div>-Mirip method show di LKEController</div><div>-Terdapat tombol persetujuan (Setuju, revisi, dan Tolak) ->diakses ketika status =1</div></div>	Rekapitulasi, SelfAssessment, DeskEvaluation, SubRincian, Pilar, RekapPengungkit, dan RekapHasil (R)																												
	pertanyaan	Mirip method pertanyaan di LKEController																													
	rekap	Mirip fitur monitoring																													
	catatan	Mirip fitur monitoring																													
	update	<div><div>-Function yang digunakan evaluator provinsi dan TPI dalam melakukan persetujuan</div><div>-evaluator provinsi akan melakukan tolak atau revisi pada function ini, dengan mengirim gmail notifikasi</div><div>-evaluator provinsi akan melakukan setuju, dan mengubah status menjadi 8</div><div>-TPI digunakan Ketika AT dan KT dalam mengirim LKE kepada atasan</div></div>	Rekapitulasi(U), EPEmail																												
	index	<div><div>-menuju halaman surat pengantar</div></div>	Rekapitulasi(R)																												

Controller	Function	Penjelasan	Database/Function
1	2	3	4
SuratPersetujuanProv		-mengambil semua data rekapitulasi berdasarkan status dan satker_id -hanya bisa diakses jika satuan kerja sudah di status >4 -melihat nilai ZI pengungkit	
	store	-melakukan validasi dan menyimpan surat pengantar yang telah diupload -nama surat akan menjadi \$satker_id-\$nama_surat, -surat pengantar akan disimpan dalam 'surat_pengantar/prov/\$tahun/name -notifikasi kirim gmail kepada TPI	Rekapitulasi, LHE(U) EPEmailDL
	destroy	Menghapus surat pengantar provinsi berdasarkan satker_id dan status=4	LHE Rekapitulasi (U)
	cetak	-digitalisasi pembuatan surat pengantar provinsi -memasukkan informasi no_surat -pembuatan table nama_satker yang mengajukan (melalui perulangan array) \$filename = 'usulan WBK-WBBM \$nama_satker'	
DeskEvaluation	index	-menuju kehalaman desk-evaluation satker -cek yang login apakah AT,KT,DL ->AT: ambil data (first) \$anggota_tpi dan reference ke table pengawasan \$data = \$anggota->pengawasan; ->KT:ambil data TPI(first)(dengan where=ketua_tim), dan cari table pengawasan \$tpi = \$ketua->id; //ambil id_ketua \$data = \$pengawasan->where('tpi_id', \$tpi); ->dalnis: ambil data dalnis (get)(dengan where=dalnis) dan cari di table pengawasan (foreach) \$tpi[] = \$d->id; //ambil id_dalnis \$data = \$pengawasan->whereIn('tpi_id', \$tpi); (endforeach) -setelah mendapatkan \$data maka direferensi ke rekapitulasi(id=satker_id), maka akan dapat table rekapitulasi. (ini permasalahan,dikarenakan yang diambil satker_id, jika berbeda tahun akan terjadi permasalahan) -melihat status pengawasan (Pengawasan) -mengambil data surat pengantar prov (satker)	TPI,anggota_tpi, pengawasan, RekapHasil,Satker
	store	--terdapat fungsi untuk scroll pada id tertentu (pertanyaan_id) Fungsi untuk Anggota Tim buat create table deskevaluation Anggota tim	DeskEvaluation(C) Rekappengungkit(U)

Controller	Function	Penjelasan	Database/Function
1	2	3	4
		<p>-akan menyimpan data catatan ,opsi_id ,dan input field at, kt, dan dl (generate kt dan dl)</p> <p>-untuk kolom nilai_kt dan nilai_dl tidak digenerate otomatis (masih null)</p> <p>-akan mengirim LKE kepada ketua tim (notif email kepada KT) ->pengawasanController(function Update)</p> <p>Ketua Tim</p> <p>-update kolom nilai_kt</p> <p>-melakukan replace(jika ada perubahan) pada kolom catatan_dl,opsi_dl, input_dl (generate dl)</p> <p>-untuk kolom nilai_dl tidak direplace</p> <p>-akan mengirim LKE kepada dalnis (notif email kepada dl) ->pengawasanController(function Update)</p> <p>-jika nilai_kt =1 ,maka updated_kt=2 ->berarti KT sudah melakukan checklist hingga tahap 2</p> <p>Dalnis</p> <p>-update kolom nilai_dl</p> <p>-jika nilai_dl =1 ,maka updated_dl=2 ->berarti DL sudah melakukan checklist hingga tahap 2</p> <p>Nilai pengungkit</p> <p>-menyimpan nilai berdasarkan pilar</p> <p>\$penimbang = bobot/jumlah pertanyaan (pada subpilar)</p> <p>Cek jika ada nilai lama di DB,maka</p> <p>\$total = nilai (nilai dari opsi_id) * penimbang +nilaiLama (total nilai based on pilar)</p> <p>Jika tidak ada nilai lama , maka</p> <p>\$total = nilai (nilai dari opsi_id) * penimbang</p> <p>LKE tidak bisa dikirim jika deskevaluation belum selesai dilakukan ->dilihat dari totProgress dan soal terjawab</p>	
	show	-sama seperti show yang terdapat pada LKEcontroller	

Controller	Function	Penjelasan	Database/Function
1	2	3	4
		<p>- jumlah evaluasi (desk-evaluation) harus sama dengan jumlah desk-evaluation, jika tidak maka tidak bisa meneruskan ke KT, khusus KT dan DL jumlah evaluasi dilihat dengan bantuan kolom updated_dl dan updated_kt.</p> <p>-perbedaan terdapat pada penggunaan table pengawasan, berfungsi untuk mengambil informasi status pengawasan dan tahap pengawasan (untuk membantu menghilangkan tombol pada kirim LKE atau lihat perubahan)</p> <p>-fitur Lihat pengisian/perubahan</p> <p>→tahap evaluasi =1</p> <p>->pada awalnya semuanya tidak ceklis, jika sudah mengisi maka akan ceklis</p> <p>→ tahap evaluasi =2</p> <p>->di cek apakah satker melakukan perubahan, jika berubah maka unchecked, jika tidak melakukan perubahan maka check.</p>	
	pertanyaan	<p>-sama seperti show yang terdapat pada LKEcontroller</p> <p>-perbedaan terdapat pada penggunaan table pengawasan, berfungsi untuk mengambil informasi status pengawasan dan tahap pengawasan (buat fitur lihat perubahan)</p> <p>-fitur lihat pengisian/perubahan logikanya sama seperti method show diatas</p>	
	rekap	Sama dengan fitur monitoring	
	update	<p>Logika sama seperti create</p> <p>RekapPengungkit</p> <p>$\\$total = \text{nilaiat_baru}(\text{nilai dari opsi_id}) * \text{penimbang} + \text{nilaiLama} - (\text{nilaiatLama} * \text{penimbang})$</p>	
LHE	store	<p>-menyimpan dokumen LHE pada folder 'LHE/\$tahun/\$tahap/\$name'</p> <p>-melakukan update status rekap sesuai kolom persetujuan</p> <p>-kirim gmail ke PIC satker</p>	Rekapitulasi, LHE(U) DLEmailPT
	cetak	<p>-digitalisasi pembuatan LHE</p> <p>-memasukkan informasi no_surat</p> <p>-pembuatan table nilai pengungkit yang mengajukan (melalui perulangan array)</p> <p>\$filename = 'LHE \$satker Tahap \$tahap '</p>	
	show	<p>-Menuju halaman buat LHE berdasarkan id_rekap (Rekapitulasi)</p> <p>-mengambil nilai zi (RekapPengungkit)</p> <p>-mengambil informasi nama TPI(TPI dan User)</p>	TPI, rekapitulasi, User, RekapPengungkit

Penjelasan Database

Note:

-Penggunaan ID menggunakan konsep UUID (untuk id yang diperlihatkan pada URL) dan *auto increments* (untuk id yang tidak diperlihatkan pada URL).

database	id	Length	Null
User	nip		-
TPI	\$nama . \$tahun . 'wil' . \$wilayah		-
Pengawasan_satker	\$anggota_id . \$satker_id		-
Persyaratan	\$tahun . \$satker_id		-
Rincian	\$data['rincian']-> ambil satu huruf awal	1	-
SubRincian	\$id_rincian . \$data['subrincian']->ambil satu huruf	2	-
Pilar	\$id_subrincian . \$id ->(id berbentuk huruf yang dimulai dari A)	3	
SubPilar	\$id_pilar . \$id (id berbentuk angka yang dimulai dari 1)	4	-
Pertanyaan	\$id_subpilar . \$id ->(id berbentuk huruf yang dimulai dari A)	5	-
Opsi	\$id_pertanyaan . \$id ((id berbentuk angka yang dimulai dari 1)	6	-
dokumenLKE	\$id_pertanyaan . \$id ((id berbentuk angka yang dimulai dari 1)	6	-
RekapHasil	date('y') . \$satker . \$pilar_id	11	-
Rekapitulasi	\$id(uuid) . \$satker_id	36	-
LHE	\$id = Auto Increment	11	Surat_pengantar_kab_kota,provinsi,lhe_1,lhe_2
SelfAssessment	\$tahun.\$satker_id.\$pertanyaan_id		
UploadDokumen	Untuk dokumenWajib -id=tahun. Dokumenlke_id.Satker_id -dokumenlke_id =reference ke table dokumenlke_id Untuk dokumen tambahan		

	- id =selfAssessment_id.id (id berbentuk huruf yang dimulai dari 1) -dokumenlke_id =\$id_pertanyaan		
RekapPengungkit	\$Uuid. \$satker_id	36	Nilai_at, nilai_at,nilai_kt,nilai_dl
inputfield	\$opsi_id.\$selfAssessment_id	20	Input_at,input_kt,input_dl

Penjelasan Relasi pada Database

Nama Entitas	<i>Multiplicity</i>	Relasi	<i>Multiplicity</i>	Nama Entitas	Penjelasan
(1)	(2)	(3)	(4)	(5)	
Subrincian	0..*	Memiliki	1..1	Rincian	1 subrincian pasti memiliki 1 rincian, sedangkan 1 rincian bisa tidak dimiliki atau >1 pada table subrincian
Pilar	0..*	Memiliki	1..1	Subrincian	1 pilar pasti memiliki 1 subrincian , sedangkan 1 subrincian bisa tidak dimiliki atau >1 pada table pilar
Subpilar	0..*	Memiliki	1..1	Pilar	1 subpilar pasti memiliki 1 pilar , sedangkan 1 pilar bisa tidak dimiliki atau >1 pada table subpilar
Pertanyaan	0..*	Memiliki	1..1	Subpilar	1 pertanyaan pasti memiliki 1 subpilar , sedangkan 1 subpilar bisa tidak dimiliki atau >1 pada table pertanyaan
Opsi	1..*	Memiliki	1..1	Pertanyaan	1 opsi pasti memiliki 1 pertanyaan , sedangkan 1 pertanyaan bisa memiliki >1 pada table opsi
Dokumen LKE	1..*	Memiliki	1..1	Pertanyaan	1 dokumen pasti memiliki 1 pertanyaan , sedangkan 1 pertanyaan bisa memiliki >1 pada table dokumen
<i>Self-assessment</i>	0..*	Memiliki	1..1	Opsi	1 self-assessment pasti memiliki 1 opsi, sedangkan 1 opsi bisa tidak dimiliki atau >1 pada table self-assessment (namun opsi, fk ny dimatiin dikarenakan untuk mengatasi permasalahan input field
<i>Self-assessment</i>	0..*	Memiliki	1..1	Satker	
<i>Self-assessment</i>	0..*	Memiliki	1..1	Pertanyaan	1 self-assessment pasti memiliki 1 satker, sedangkan 1 satker bisa tidak dimiliki atau >1 pada table self-assessment
<i>Self-assessment</i>	0..*	Memiliki	1..1	Rekapitulasi	1 self-assessment pasti memiliki 1 rekapitulasi, sedangkan 1 rekapitulasi bisa tidak dimiliki atau >1 pada table self-assessment
Input Field	0..*	Memiliki	1..1	Opsi	1 inputfield pasti memiliki 1 opsi, sedangkan 1 opsi bisa tidak dimiliki atau >1 pada table input field
Input Field	0..*	Memiliki	1..1	<i>Self-assessment</i>	1 inputfield pasti memiliki 1 self-asessement, sedangkan 1 self-asessement bisa tidak dimiliki atau >1 pada table input field
Upload Dokumen	0..*	Memiliki	1..1	Dokumen LKE	1 uploaddokumen pasti memiliki 1 dokumenLKE, sedangkan 1 dokumen bisa tidak dimiliki atau >1 pada table upload_dokumen
Upload Dokumen	0..*	Memiliki	1..1	<i>Self-assessment</i>	1 uploaddokumen pasti memiliki 1 selfassessment, sedangkan 1 selfassessment bisa tidak dimiliki atau >1 pada table upload_dokumen
<i>Desk-evaluation</i>	1..1	Memiliki	1..1	<i>Self-assessment</i>	ini sebenarnya bisa digabung saja
<i>Desk-evaluation</i>	0..*	Memiliki	1..1	Pengawasan	1 deskevaluation pasti memiliki 1 pengawasan, sedangkan 1 pengawasan bisa tidak dimiliki atau >1 pada table deskevaluation

Nama Entitas	<i>Multiplicity</i>	Relasi	<i>Multiplicity</i>	Nama Entitas	Penjelasan
(1)	(2)	(3)	(4)	(5)	
Rekapitulasi	0..*	Memiliki	1..1	Status Rekap	1 rekapitulasi pasti memiliki 1 statusrekap, sedangkan 1 status rekap bisa tidak dimiliki atau >1 pada table rekapitulasi
Rekapitulasi	0..*	Memiliki	1..1	Satker	1 rekapitulasi pasti memiliki 1 satker, sedangkan 1 satker bisa tidak dimiliki atau >1 pada table rekapitulasi
Rekap Pengungkit	1..12	Memiliki	1..1	Rekapitulasi	1 rekappengungkit pasti memiliki 1 rekapitulasi, sedangkan 1 rekapitulasi bisa dimiliki maksimal 12 pada table rekappengungkit
Rekap Pengungkit	0..12	Memiliki	1..1	Pilar	1 rekappengungkit pasti memiliki 1 pilar, sedangkan 1 pilar bisa tidak dimiliki atau maks 12pada table rekappengungkit
Rekap Hasil	0..*	Memiliki	1..1	Opsi	Isian RekapHasil pasti punya satu opsi, sedangkan opsi belum tentu ada di table rekapHasil, dan bisa saja lebih dari 1
Rekap Hasil	1..3	Memiliki	1..1	Pilar	Isian pilar pasti terdapat di 3 kolom pada table rekapHasil, sedangkan 1 rekapHasil pasti memiliki 1 pilar
Rekap Hasil	0..*	Memiliki	1..1	Satker	1 rekaphasil pasti memiliki 1 satker, 1 satker bisa tidak ada di table rekapHasil ataupun lebih dari 1
LHE	1..1	Memiliki	1..1	Rekapitulasi	(ini bisa digabung, tapi terlanjut dipisah)
User	0..*	Memiliki	1..1	Level	1 user pasti memiliki maks1 level, 1 level bisa dimiliki 0 user atau banyak user
User	0..*	Memiliki	1..1	Satker	1 user pasti memiliki maks 1 satker, 1 satker bisa dimiliki 0 user atau banyak user
Persyaratan	0..*	Memiliki	1..1	Satker	Satu persyaratan pasti memiliki maks 1 satker, sedangkan satker belum tentu ada di table persyaratan dan bisa lebih dari 1
Anggota TPI	1..2	Memiliki	1..1	TPI	1 anggota tim hanya masuk dalam satu tim, 1Tim maksimal punya 2 anggota
Pengawasan Satker	0..14	Memiliki	1..1	Anggota TPI	1 pengawasan satker pasti hanya diawasi oleh satu anggota tim, 1 anggota tim dapat mengawasi 0 hingga maksimal 14 satker
Pengawasan Satker	0..*	Memiliki	1..1	Satker	Satu id_Pengawasan satker pasti mengawasi satu satker, sedangkan satuan kerja ada yang tidak masuk ke tabel itu dan bisa lebih dari 1