Software Design Document

Sistem Informasi Survei Kepuasan Masyarakat Studi Kasus Kecamatan Sigumpar

Disusun Oleh:

|  |  |
| --- | --- |
| 12S20003 | Marcel Jhosua Simanjuntak |
| 12S20007 | Rufina Aprilina Sitorus |
| 12S20026 | Mastuari Octafina Rumapea |
| 12S20030 | Lamsihar Siahaan |
| 12S20041 | Theresia Agatha Silas Siahaan |
| 12S20049 | Meida Enggelica Butarbutar |

Untuk:  
Kantor Camat Sigumpar  
Sigumpar, Laguboti

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Proyek Sistem Informasi 2023**  **Institut Teknologi Del** | | | |  |
| No. Dokumen: SDD-PSI--06.doc | | | Versi: xx.xx | Tanggal : 13-05-23 | Jumlah Halaman : 74 | |

# DAFTAR ISI

DAFTAR TABLE iv

DAFTAR GAMBAR vi

1. Introduction 8

1.1 Purpose of Document 8

1.2 Scope 8

1.3 Definition, Acronim, and Abbreviation 8

1.4 Identification and Numbering 9

1.5 Reference Documents 10

1.6 Document Summary 10

2 Design Description 11

2.1 Environment 11

2.2 Data Description 11

2.2.1 Domain/Type Definition 11

2.2.2 E-R Diagram 12

2.2.3 Conceptual Data Model 19

2.2.4 Physical Data Model 20

2.2.5 Tables 21

2.3 Functional Decomposition 22

3 Detail Design Description 27

3.1 Table Structure 27

3.1.1 Table password\_reset\_tokens 27

3.1.2 Table failed\_jobs 28

3.1.3 Table provinces 29

3.1.4 Table regencies 30

3.1.5 Table districts 31

3.1.6 Table villages 32

3.1.7 Table tenants 33

3.1.8 Table divisions 35

3.1.9 Table services 36

3.1.10 Table users 37

3.1.11 Table occupation\_options 39

3.1.12 Table answer\_types 40

3.1.13 Table answer\_options 41

3.1.14 Table question\_sets 42

3.1.15 Table questions 43

3.1.16 Table age\_options 44

3.1.17 Table respondents 45

3.1.18 Table answer 47

3.1.19 Table migrations 48

3.1.20 Table personal\_acces\_tokens 49

3.2 Detail Functional Description 51

3.2.1 Function Specification Melakukan Registrasi untuk Instansi dan Operator 51

3.2.2 Function Spesification Melakukan Login 54

3.2.3 Function Spesification List Operator 57

3.2.4 Function Spesification Menambah Operator Baru 58

3.2.5 Function Spesification Penonaktifan (Diaktivasi) Akun Operator 60

3.2.6 Function Spesification Update Profile 62

3.2.7 Function Spesification Update Tenant 64

3.2.8 Function Spesification Mengubah Kata Sandi 66

3.2.9 Function Spesification Menambah Question Set 68

3.2.10 Function Spesification List Question Set 70

3.2.11 Function Spesification Menambah Question 72

3.2.12 Function Spesification List Question 74

3.2.13 Function Spesification Report 76

3.3 Physical File 89

3.4 Traceability 89

3.4.1 Data 89

3.4.2 Requirements 91

Lampiran 92

Sejarah Versi 93

Sejarah Perubahan 94

# DAFTAR TABLE

[Table 1. Definition 8](#_Toc136778694)

[Table 2. Acronim and Abbreviation 8](#_Toc136778695)

[Table 3. Identification and Numbering 9](#_Toc136778696)

[Table 4. Environment: Hardware 11](#_Toc136778697)

[Table 5. Environment: Software 11](#_Toc136778698)

[Table 6. Atribut Entity User 13](#_Toc136778699)

[Table 7. Atribut Entity AgeOption 13](#_Toc136778700)

[Table 8. Atribut Entity OccupationOption 13](#_Toc136778701)

[Table 9. Atribut Entity Province 13](#_Toc136778702)

[Table 10. Atribut Entity Regency 14](#_Toc136778703)

[Table 11. Atribut Entity District 14](#_Toc136778704)

[Table 12. Atribut Entity Village 14](#_Toc136778705)

[Table 13. Atribut Entity Tenant 15](#_Toc136778706)

[Table 14. Atribut Entity Division 15](#_Toc136778707)

[Table 15. Atribut Entity Service 15](#_Toc136778708)

[Table 16. Atribut Entity Respondent 16](#_Toc136778709)

[Table 17. Atribut Entity Question Set 16](#_Toc136778710)

[Table 18. Atribut Entity Question 17](#_Toc136778711)

[Table 19. Atribut Entity Answer 17](#_Toc136778712)

[Table 20. Atribut Entity AnswerType 17](#_Toc136778713)

[Table 21. Atribut Entity AnswerOption 17](#_Toc136778714)

[Table 22. Tabel pada Basis Data 21](#_Toc136778715)

[Table 23. Functional Decomposition 22](#_Toc136778716)

[Table 24. Tabel password\_reset\_tokens 27](#_Toc136778717)

[Table 25. Tabel failed\_jobs 28](#_Toc136778718)

[Table 26. Tabel provinces 29](#_Toc136778719)

[Table 27. Tabel regencies 30](#_Toc136778720)

[Table 28. Tabel districts 31](#_Toc136778721)

[Table 29. Tabel villages 32](#_Toc136778722)

[Table 30. Tabel tenants 33](#_Toc136778723)

[Table 31. Tabel divisions 35](#_Toc136778724)

[Table 32. Tabel services 36](#_Toc136778725)

[Table 33. Tabel users 37](#_Toc136778726)

[Table 34. Tabel occupation\_options 39](#_Toc136778727)

[Table 35. Tabel answer\_types 40](#_Toc136778728)

[Table 36. Tabel answer\_options 41](#_Toc136778729)

[Table 37. Table question\_sets 42](#_Toc136778730)

[Table 38. Tabel questions 43](#_Toc136778731)

[Table 39. Tabel age\_options 44](#_Toc136778732)

[Table 40. Tabel respondents 45](#_Toc136778733)

[Table 41. Tabel answer 47](#_Toc136778734)

[Table 42. Tabel migrations 49](#_Toc136778735)

[Table 43. Tabel personal\_access\_token 50](#_Toc136778736)

[Table 44. Field Registrasi untuk Instansi dan Operator 53](#_Toc136778737)

[Table 45. Objects Specification 53](#_Toc136778738)

[Table 46. Error Message pada Registrasi untuk Instansi dan Operator 54](#_Toc136778739)

[Table 47. Field pada Login 55](#_Toc136778740)

[Table 48. Object pada Login 55](#_Toc136778741)

[Table 49. Error Message pada Login 56](#_Toc136778742)

[Table 50. Field pada List Operator 57](#_Toc136778743)

[Table 51. Object pada List Operator 58](#_Toc136778744)

[Table 52. Field pada Menambah Operator Baru 59](#_Toc136778745)

[Table 53. Object pada Menambah Operator Baru 59](#_Toc136778746)

[Table 54. Error Message pada Menambah Operator Baru 60](#_Toc136778747)

[Table 55. Field pada Penonaktifan (Diaktivasi) Akun Operator 61](#_Toc136778748)

[Table 56. Object pada Penonaktifan (Diaktivasi) Akun Operator 61](#_Toc136778749)

[Table 57. Error Message pada Penonaktifan (Diaktivasi) Akun Operator 62](#_Toc136778750)

[Table 58. Field pada Update Profile 63](#_Toc136778751)

[Table 59. Object pada Update Profile 63](#_Toc136778752)

[Table 60. Error Message pada Update Profile 64](#_Toc136778753)

[Table 61. Field pada Update Tenant 65](#_Toc136778754)

[Table 62. Object pada Update Tenant 65](#_Toc136778755)

[Table 63. Error Message pada Update Tenant 66](#_Toc136778756)

[Table 64. Field pada Mengubah Kata Sandi 67](#_Toc136778757)

[Table 65. Object pada Mengubah Kata Sandi 67](#_Toc136778758)

[Table 66. Error Message pada Mengubah Kata Sandi 68](#_Toc136778759)

[Table 67. Field pada Menambah Services 69](#_Toc136778760)

[Table 68. Objek pada Menambah Services 69](#_Toc136778761)

[Table 69. Error Message pada Menambah Services 70](#_Toc136778762)

[Table 70. Field pada List Question Set 71](#_Toc136778763)

[Table 71. Object pada List Question Set 71](#_Toc136778764)

[Table 72. Field pada Fungsi Menambah Question 73](#_Toc136778765)

[Table 73. Objek dari Fungsi Menambah Question 73](#_Toc136778766)

[Table 74. Error Message pada Fungsi Menambah Question 73](#_Toc136778767)

[Table 75. Objek dari Fungsi List question 75](#_Toc136778768)

[Table 76. Fields Description Menambah Divisi Baru 79](#_Toc136778769)

[Table 77. Field Description Mengedit Divisi 80](#_Toc136778770)

[Table 78. Traceability Tabel Aplikasi Terhadap DFD dan ER 88](#_Toc136778771)

[Table 79. Requirements 90](#_Toc136778772)

# DAFTAR GAMBAR

[Gambar 1. ER-Diagram 12](#_Toc136778773)

[Gambar 2. Conceptual Data Model 19](#_Toc136778774)

[Gambar 3. Physical Data Model 20](#_Toc136778775)

[Gambar 4. Database password\_reset\_tokens 27](#_Toc136778776)

[Gambar 5. Database failed\_jobs 29](#_Toc136778777)

[Gambar 6. Database provinces 30](#_Toc136778778)

[Gambar 7. Database regencies 31](#_Toc136778779)

[Gambar 8. Database districts 32](#_Toc136778780)

[Gambar 9. Database villages 33](#_Toc136778781)

[Gambar 10. Database tenants 35](#_Toc136778782)

[Gambar 11. Database divisions 36](#_Toc136778783)

[Gambar 12. Database services 37](#_Toc136778784)

[Gambar 13. Database users 39](#_Toc136778785)

[Gambar 14. Database occupation\_options 40](#_Toc136778786)

[Gambar 15. Database answer\_types 41](#_Toc136778787)

[Gambar 16. Database answer\_options 42](#_Toc136778788)

[Gambar 17. Database question\_sets 43](#_Toc136778789)

[Gambar 18. Database questions 44](#_Toc136778790)

[Gambar 19. Database age\_options 45](#_Toc136778791)

[Gambar 20. Database respondents 47](#_Toc136778792)

[Gambar 21. Database answer 48](#_Toc136778793)

[Gambar 22. Database migrations 49](#_Toc136778794)

[Gambar 23. Database personal\_access\_token 51](#_Toc136778795)

[Gambar 24. Detail Layout Registrasi untuk Instansi dan Operator 53](#_Toc136778796)

[Gambar 25. Detail Layout Login 55](#_Toc136778797)

[Gambar 26. Detail Layout pada List Operator 57](#_Toc136778798)

[Gambar 27. Detail Layout pada Menambah Operator 59](#_Toc136778799)

[Gambar 28. Detail Layout pada Penonaktifan (Diaktivasi) Akun Operator 61](#_Toc136778800)

[Gambar 29. Detail Layout pada Update Profile 63](#_Toc136778801)

[Gambar 30. Detail Layout pada Update Tenant 65](#_Toc136778802)

[Gambar 31. Detail Layout pada Mengubah Kata Sandi 67](#_Toc136778803)

[Gambar 32. Detail Layout Menambah Question Set 69](#_Toc136778804)

[Gambar 33. Detail Layout List Question Set 71](#_Toc136778805)

[Gambar 34. Detail Layout Menambah Question 73](#_Toc136778806)

[Gambar 35. Detail Layout List Question 75](#_Toc136778807)

[Gambar 36. Detail layout Report 77](#_Toc136778808)

[Gambar 37. Detail Layout Menambah Divisi 79](#_Toc136778809)

[Gambar 38. Detail Layout Mengedit Divisi 81](#_Toc136778810)

[Gambar 39. Detail layout Menghapus Divisi 82](#_Toc136778811)

[Gambar 40. Detail Layout pada Menambah Layanan Baru 84](#_Toc136778812)

[Gambar 41. Detail Layout Mengedit Layanan 86](#_Toc136778813)

[Gambar 42. Detail Layout Menghapus Layanan 88](#_Toc136778814)

# Introduction

Pada bab ini dibahas mengenai tujuan penulisan dokumen, ruang lingkup dokumen, definisi, akronim, dan singkatan yang terdapat dalam dokumen, identifikasi dan aturan penomoran dokumen, dokumen rujukan, serta ringkasan dokumen. Berikut akan dijelaskan pada masing-masing sub-bab tersebut.

## Purpose of Document

Dokumen ini digunakan oleh tim *developer* sebagai acuan dalam mengimplementasikan sistem yang dikembangkan. Pada dokumen ini diuraikan struktur serta fungsi-fungsi yang terdapat pada Sistem Informasi SUKEMAS.

## Scope

Ruang lingkup dokumen ini meliputi penjelasan mengenai desain sistem yang akan dikembangkan dan penjelasan mengenai batasan pembangunan aplikasi serta fungsi-fungsi yang digunakan.

## Definition, Acronim, and Abbreviation

Berikut adalah daftar definisi, akronim, dan singkatan yang digunakan dalam dokumen. Definisi yang digunakan pada dokumen ini dapat dilihat padaTable 1**.**

Table . Definition

|  |  |  |
| --- | --- | --- |
| **No.** | **Definisi** | **Keterangan** |
| 1. | Client | Orang yang memberikan proyek pada *developer.* |
| 2. | Developer | Orang yang mengembangkan produk sesuai kebutuhan *client.* |
| 3. | Hardware | Perangkat keras dan peranti yang mendukung sistem komputer. |
| 4. | Software | Perangkat lunak yang digunakan untuk membangun sistem. |
| 5. | Software Engineering | Teknik pembuatan perangkat lunak. |
| 6. | Processor | Perangkat yang digunakan untuk mengatur segala proses dalam suatu komputer. |
| 7. | Tools | Aplikasi yang digunakan selama pengerjaan proyek. |
| 8. | Entity | Objek pada dunia nyata yang dapat dibedakan dari objek-objek lain. |
| 9. | Primary Key | Kunci utama yang merupakan identifikasi khusus yang unik dari suatu *field* pada tabel. |

Daftar akronim dan singkatan yang digunakan dalam dokumen ini diuraikan padaTable 2.

Table . Acronim and Abbreviation

|  |  |  |
| --- | --- | --- |
| **No** | **Akronim dan Singkatan** | **Keterangan** |
| 1. | PiP | *Project Implementation Plan* adalah dokumen perencanaan pembangunan *website* Sistem Informasi yang menjadi langkah awal dalam pengerjaan proyek |
| 2. | SyRS | *System Requirement Specification* adalah dokumen yang menguraikan kebutuhan sistem, Gambaran sistem yang akan dibangun, beserta fungsi-fungsi yang digunakan pada sistem. |
| 3. | SDD | *Software Design Document* adalah dokumen yang mendeskripsikan Gambaran dan desain dari *software* yang akan dibuat. |
| 4. | PSI | Proyek Sistem Informasi. |
| 5. | IT Del | Institut Teknologi Del. |
| 6. | DBMS | Database Management System. |
| 7. | ERD | Entity Relationship Diagram. |
| 8. | CDM | Conceptual Data Model. |
| 9. | PDM | Physical Data Model. |

## Identification and Numbering

Beberapa aturan penamaan dan penomoran yang disepakati oleh anggota kelompok untuk memudahkan pemahaman tentang dokumen ini dapat dilihat padaTable 3.

Table . Identification and Numbering

|  |  |
| --- | --- |
| **No.** | **Deskripsi Ketentuan** |
| 1. | Aturan Penamaan dokumen dengan ketentuan: SDD-ISP-YY-SIXX  Dimana YY adalah tahun penyusunan dokumen ini dan XX adalah nomor kelompok  Maka dokumen ini dinamai dengan SDD-ISP-23-SI06. |
| 2. | Aturan penulisan :  Font: Times New Roman dan Arial, ukuran 12pt  Heading 1: bold, 16 pt  Heading 2: bold, 12 pt  Heading 3: bold, italic, 12 pt  Paragraf:  Line-spacing: 1.5 lines  Before and after pada spacing: 0 pt |
|  | Paragraf tidak dimulai dengan menjorok ke dalam  Paragraf rata kiri kanan(*justify*)  Istilah asing ditulis dengan format *italic*(tulisan miring)  *Caption* table/Gambar  Line spacing yang digunakan pada caption tabel/Gambar: single, before and after : 0pt  Ukuran huruf yang digunakan 10pt  Penamaan *caption* tabel dibuat rata tengah *(center)* atas tabel  Penamaan *caption* Gambar dibuat di sebelah kiri bawah Gambar  Tabel  *Header* tabel diulang jika isi tabel berada di beberapa halaman  Penulisan *header*: *Arial*, ukuran 10pt, dan *bold* |
| 3. | Aturan penomoran dan penamaan bab dan subbab :  Untuk bab: 1, 2, 3, dan seterusnya. Contoh : 1. Introduction  Untuk subbab: 1.1, 1.2, 1.3, dan seterusnya. Contoh:  1.1 Purpose of Document  Untuk sub subbab : 1.1.1, 1.1.2, 1.1.3 Contoh:  2.2.1 Domain/Type Definition  Penomoran dan penulisan pada tabel dan Gambar : Untuk tabel : Tabel 1. Daftar Definisi  Untuk Gambar: Gambar 1. Struktur Organisasi |
| 4. | Aturan penamaan *entity* dengan nama t\_XXX t\_masyarakat  t\_akun  XXX : Nama entity |

## Reference Documents

Dokumen yang menjadi rujukan dalam penulisan dokumen ini adalah:

* + 1. STD-PSI-2020, Standar Penomoran dan Tata Nama Artefak Proyek Sistem Informasi terbitan tahun 2020.
    2. ToR-PSI-23-06, Term of Reference SUKEMAS sebagai Sarana Penyampaian Kepuasan Masyarakat Berbasis Web Online.
    3. PiP-PSI-23-06, Project Implementation Plan SUKEMAS Sarana Penyampaian Kepuasan Masyarakat Berbasis Web Online.
    4. URS-PSI-23-06, User Requirement Specification SUKEMAS sebagai Sarana Penyampaian Kepuasan Masyarakat Berbasis Web Online.
    5. SyRS-PSI-23-06, System Requirement Specification SUKEMAS sebagai Sarana Penyampaian Kepuasan Masyarakat Berbasis Web Online.

## Document Summary

Sistematika dokumen ini disusun ke dalam tiga bab, diantaranya Bab 1 *Introduction*. Pada bab pertama dijelaskan tujuan penulisan dokumen, ruang lingkup dokumen, definisi, akronim, dan singkatan yang terdapat dalam dokumen, identifikasi dan aturan penomoran dokumen, dokumen rujukan, serta ringkasan dokumen. Bab 2 *Design Description*, pada bab kedua dijelaskan rancangan lingkungan implementasi, deskripsi data dan dekomposisi fungsional dari sistem yang dibangun. Yang terakhir Bab 3 *Detail Design Description,* pada bab ketiga dijelaskan struktur tabel, spesifikasi fungsi yang digunakan pada pembangunan sistem dan *traceability* dari sistem yang dikembangkan.

# Design Description

Pada bab ini dibahas mengenai rancangan lingkungan sistem akan diimplementasi, deskripsi data dan dekomposisi fungsional dari sistem yang dikembangkan.

## Environment

Pada subbab ini dibahas mengenai spesifikasi teknis perangkat lunak yang dipakai dalam pengembangan.

1. Operating System. Sistem operasi yang digunakan adalah *Windows 10.*
2. *Hardware*. Spesifikasi *hardware* yang digunakan untuk pengembangan Sistem Informasi SUKEMAS, yaitu:

Table . Environment: Hardware

|  |  |  |
| --- | --- | --- |
| No. | Hardware | Spesifikasi |
| 1. | Processor | Quad Core, 2.3 GHz (2M Cache, up to 4 GHz). |
| 2. | RAM | 8.00GB |

1. Software. *Software* yang digunakan untuk pembangunan SUKEMAS sebagai Sarana Penyampaian Kepuasan Masyarakat Online Berbasis Web, yaitu:

Table . Environment: Software

|  |  |  |
| --- | --- | --- |
| No. | Software | Spesifikasi |
| 1. | Bahasa Pemrograman | PHP 8.2.5 (cli) |
| 2. | Development Tools | Visual Studi Code |
| 3. | Word Processor | Microsoft Word 2013 |
| 4. | DBMS | SQLite, MariaDB |
| 5. | Drawing ERD | Draw.io |
| 6. | Drawing CDM, PDM | SQLYog, Draw.io |
| 7. | Web Browser | MozillaFirefox, Google Chrome |
| 8. | Web Server | PHP Native Web Server |

## Data Description

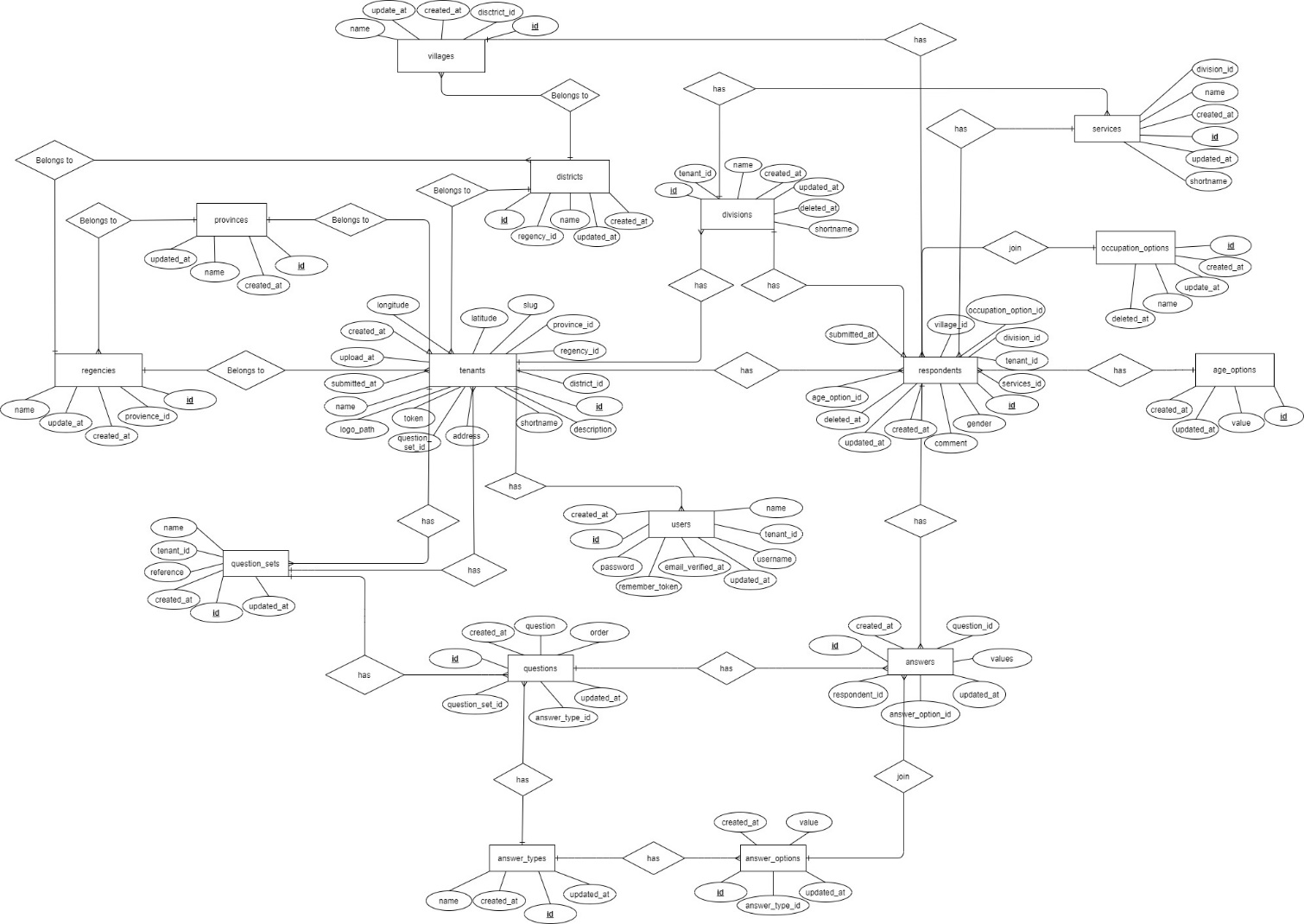
Bagian ini menjelaskan mengenai data-data dari Sistem Informasi SUKEMAS. Diantaranya adalah nama domain atau tipe domain, *ER-Diagram*, pemodelan data secara konseptual dan fisikal, serta deskripsi tabel-tabel pada basis data.

### Domain/Type Definition

Tidak ada domain/type definition.

### E-R Diagram

Gambar 1ER-Diagrammenampilkan ERD yang menggambarkan hubungan antar entity didalam Sistem Informasi SUKEMAS.



Gambar . ER-Diagram

Aplikasi SUKEMAS memiliki 16 entitas, yaitu:

1. User. Atribut-atribut dari *entity* User adalah:

Table . Atribut Entity User

|  |  |
| --- | --- |
| **No.** | **Atribut** |
| 1. | id digunakan sebagai *primary key* dalam entitas User |
| 2. | name merupakan nama dari User |
| 3. | username merupakan nama unik User dalam sistem |
| 4. | email merupakan alamat surel unik bagi User dalam sistem untuk otentikasi |
| 5. | password merupakan kata sandi rahasia bagi User untuk otentikasi |
| 6. | email\_verified\_at merupakan cap waktu yang menandakan waktu verifikasi surel |
| 7. | tenant\_id merupakan id dari entitas Tenant tempat User terkait ditugaskan |
| 8. | remember\_token digunakan untuk mengidentifikasi User secara otomatis saat kembali ke website setelah memilih opsi “remember me”, sehingga tidak perlu login kembali setiap kali membuka website tersebut. |
| 9. | created\_at merupakan cap waktu yang menandakan waktu pembuatan entitas User |
| 10. | updated\_at merupakan cap waktu yang menandakan waktu pengubahan entitas User |
| 11. | deleted\_at merupakan cap waktu yang menandakan waktu penonaktifan entitas User |

1. AgeOption. Atribut-atribut dari *entity* AgeOption adalah:

Table . Atribut Entity AgeOption

|  |  |
| --- | --- |
| **No.** | **Atribut** |
| 1. | id digunakan sebagai *primary key* dalam entitas AgeOption |
| 2. | value merupakan isi dari AgeOption |
| 3. | created\_at merupakan cap waktu yang menandakan waktu pembuatan entitas AgeOption |
| 4. | updated\_at merupakan cap waktu yang menandakan waktu pengubahan entitas AgeOption |
| 5. | deleted\_at merupakan cap waktu yang menandakan waktu penonaktifan entitas AgeOption |

1. OccupationOption. Atribut-atribut dari *entity* OccupationOption adalah:

Table . Atribut Entity OccupationOption

|  |  |
| --- | --- |
| **No.** | **Atribut** |
| 1. | id digunakan sebagai *primary key* dalam entitas OccupationOption |
| 2. | name merupakan nama dari OccupationOption |
| 3. | created\_at merupakan cap waktu yang menandakan waktu pembuatan entitas OccupationOption |
| 4. | updated\_at merupakan cap waktu yang menandakan waktu pengubahan entitas OccupationOption |
| 5. | deleted\_at merupakan cap waktu yang menandakan waktu penonaktifan entitas OccupationOption |

1. Province. Atribut-atribut dari *entity* Province adalah:

Table . Atribut Entity Province

|  |  |
| --- | --- |
| **No.** | **Atribut** |
| 1. | id digunakan sebagai *primary key* dalam entitas Province |
| 2. | name merupakan nama dari Province |
| 3. | created\_at merupakan cap waktu yang menandakan waktu pembuatan entitas Province |
| 4. | updated\_at merupakan cap waktu yang menandakan waktu pengubahan entitas Province |
| 5. | deleted\_at merupakan cap waktu yang menandakan waktu penonaktifan entitas Province |

1. Regency. Atribut-atribut dari *entity* Regency adalah:

Table . Atribut Entity Regency

|  |  |
| --- | --- |
| **No.** | **Atribut** |
| 1. | id digunakan sebagai *primary key* dalam entitas Regency |
| 2. | name merupakan nama dari Regency |
| 3. | province\_id merupakan id dari Province dimana Regency terkait terletak |
| 4. | created\_at merupakan cap waktu yang menandakan waktu pembuatan entitas Regency |
| 5. | updated\_at merupakan cap waktu yang menandakan waktu pengubahan entitas Regency |
| 6. | deleted\_at merupakan cap waktu yang menandakan waktu penonaktifan entitas Regency |

1. District. Atribut-atribut dari entity District adalah:

Table . Atribut Entity District

|  |  |
| --- | --- |
| **No.** | **Atribut** |
| 1. | id digunakan sebagai primary key dalam entitas District |
| 2. | name merupakan nama dari District |
| 3. | regency\_id merupakan id dari Regency dimana District terkait terletak |
| 4. | created\_at merupakan cap waktu yang menandakan waktu pembuatan entitas District |
| 5. | updated\_at merupakan cap waktu yang menandakan waktu pengubahan entitas District |
| 6. | deleted\_at merupakan cap waktu yang menandakan waktu penonaktifan entitas District |

1. Village. Atribut-atribut dari *entity* Village adalah:

Table . Atribut Entity Village

|  |  |
| --- | --- |
| **No.** | **Atribut** |
| 1. | id digunakan sebagai *primary key* dalam entitas Village |
| 2. | name merupakan nama dari Village |
| 3. | district\_id merupakan id dari District dimana Village terkait terletak |
| 4. | created\_at merupakan cap waktu yang menandakan waktu pembuatan entitas Village |
| 5. | updated\_at merupakan cap waktu yang menandakan waktu pengubahan entitas Village |
| 6. | deleted\_at merupakan cap waktu yang menandakan waktu penonaktifan entitas Village |

1. Tenant. Atribut-atribut dari *entity* Tenant adalah:

Table . Atribut Entity Tenant

|  |  |
| --- | --- |
| **No.** | **Atribut** |
| 1. | id digunakan sebagai *primary key* dalam entitas Tenant |
| 2. | name merupakan nama dari Tenant |
| 3. | address merupakan alamat dari Tenant |
| 4. | province\_id merupakan id dari Province dimana Tenant terkait terletak |
| 5. | regency\_id merupakan id dari Regency dimana Tenant terkait terletak |
| 6. | district\_id merupakan id dari District dimana Tenant terkait terletak |
| 7. | slug merupakan sepotong kata unik yang merepresentasikan entitas Tenant |
| 8. | token merupakan sebuah deretan karakter unik yang menjadi “alamat” laman Tenant |
| 9. | logo\_path merupakan alamat penyimpanan Gambar logo Tenant |
| 10. | latitude merupakan posisi Tenant pada garis lintang |
| 11. | longitude merupakan posisi Tenant pada garis bujur |
| 12. | description merupakan teks deskripsi entitas Tenant |
| 13. | submitted\_at merupakan cap waktu yang menandakan waktu penyerahan entitas User |
| 14. | created\_at merupakan cap waktu yang menandakan waktu pembuatan entitas User |
| 15. | updated\_at merupakan cap waktu yang menandakan waktu pengubahan entitas User |
| 16. | deleted\_at merupakan cap waktu yang menandakan waktu penonaktifan entitas User |

1. Division. Atribut-atribut dari *entity* Division adalah:

Table . Atribut Entity Division

|  |  |
| --- | --- |
| **No.** | **Atribut** |
| 1. | id digunakan sebagai *primary key* dalam entitas Division |
| 2. | name merupakan nama dari Division |
| 3. | tenant\_id merupakan id dari Tenant yang menjadi induk dari Division terkait |
| 4. | created\_at merupakan cap waktu yang menandakan waktu pembuatan entitas Division |
| 5. | updated\_at merupakan cap waktu yang menandakan waktu pengubahan entitas Division |
| 6. | deleted\_at merupakan cap waktu yang menandakan waktu penonaktifan entitas Division |

1. Service. Atribut-atribut dari *entity* Service adalah:

Table . Atribut Entity Service

|  |  |
| --- | --- |
| **No.** | **Atribut** |
| 1. | id digunakan sebagai *primary key* dalam entitas Service |
| 2. | name merupakan nama dari Service |
| 3. | tenant\_id merupakan id dari Division yang menjadi induk dari Service terkait |
| 4. | created\_at merupakan cap waktu yang menandakan waktu pembuatan entitas Service |
| 5. | updated\_at merupakan cap waktu yang menandakan waktu pengubahan entitas Service |
| 6. | deleted\_at merupakan cap waktu yang menandakan waktu penonaktifan entitas Service |

1. Respondent. Atribut-atribut dari *entity* Respondent adalah:

Table . Atribut Entity Respondent

|  |  |
| --- | --- |
| **No.** | **Atribut** |
| 1. | id digunakan sebagai *primary key* dalam entitas Respondent |
| 2. | village\_id merupakan id dari Village tempat tinggal Respondent |
| 3. | occupation\_id merupakan id dari OccupationOption profesi Respondent |
| 4. | tenant\_id merupakan id dari Tenant dimana Service terkait beroperasi |
| 5. | division\_id merupakan id dari Division dimana Service terkait beroperasi |
| 6. | service\_id merupakan id dari Service yang dinilai oleh Respondent |
| 7. | age\_option\_id merupakan id dari AgeOption usia dari Respondent |
| 8. | gender merupakan jenis kelamin dari Respondent |
| 9. | comment merupakan komentar maupun buah pikiran dari Respondent |
| 10. | submitted\_at merupakan cap waktu yang menandakan waktu penyerahan entitas User |
| 11. | created\_at merupakan cap waktu yang menandakan waktu pembuatan entitas User |
| 12. | updated\_at merupakan cap waktu yang menandakan waktu pengubahan entitas User |
| 13. | deleted\_at merupakan cap waktu yang menandakan waktu penonaktifan entitas User |

1. Question Set. Atribut-atribut dari *entity* QuestionSet adalah:

Table . Atribut Entity Question Set

|  |  |
| --- | --- |
| **No.** | **Atribut** |
| 1. | id digunakan sebagai *primary key* dalam entitas QuestionSet |
| 2. | name merupakan nama dari QuestionSet |
| 3. | tenant\_id merupakan id dari Tenant dimana QuestionSet terkait digunakan |
| 4. | reference merupakan referensi hukum dimana QuestionSet dirumuskan |
| 5. | created\_at merupakan cap waktu yang menandakan waktu pembuatan entitas QuestionSet |
| 6. | updated\_at merupakan cap waktu yang menandakan waktu pengubahan entitas QuestionSet |
| 7. | deleted\_at merupakan cap waktu yang menandakan waktu penonaktifan entitas QuestionSet |

1. Question. Atribut-atribut dari *entity* Question adalah:

Table . Atribut Entity Question

|  |  |
| --- | --- |
| **No.** | **Atribut** |
| 1. | id digunakan sebagai *primary key* dalam entitas Question |
| 2. | question merupakan pertanyaan |
| 3. | order merupakan nomor urut dari Question |
| 4. | answer\_type\_id merupakan id dari AnswerType yang digunakan oleh Question |
| 5. | question\_set\_id merupakan id dari QuestionSet tempat Question digunakan |
| 6. | created\_at merupakan cap waktu yang menandakan waktu pembuatan entitas Question |
| 7. | updated\_at merupakan cap waktu yang menandakan waktu pengubahan entitas Question |
| 8. | deleted\_at merupakan cap waktu yang menandakan waktu penonaktifan entitas Question |

1. Answer. Atribut-atribut dari *entity* Answer adalah:

Table . Atribut Entity Answer

|  |  |
| --- | --- |
| **No.** | **Atribut** |
| 1. | id digunakan sebagai *primary key* dalam entitas Answer |
| 2. | value merupakan jawaban |
| 3. | order merupakan nomor urut dari Question |
| 4. | respondent\_id merupakan id dari Respondent yang mengisi Answer terkait |
| 5. | question\_id merupakan id dari Question yang dijawab oleh Answer digunakan |
| 6. | answer\_option\_id merupakan id dari AnswerOption yang digunakan oleh Answer terkait |
| 7. | created\_at merupakan cap waktu yang menandakan waktu pembuatan entitas Answer |
| 8. | updated\_at merupakan cap waktu yang menandakan waktu pengubahan entitas Answer |
| 9. | deleted\_at merupakan cap waktu yang menandakan waktu penonaktifan entitas Answer |

1. AnswerType. Atribut-atribut dari *entity* AnswerType adalah:

Table . Atribut Entity AnswerType

|  |  |
| --- | --- |
| **No.** | **Atribut** |
| 1. | id digunakan sebagai *primary key* dalam entitas AnswerType |
| 2. | name merupakan nama dari AnswerType |
| 3. | created\_at merupakan cap waktu yang menandakan waktu pembuatan entitas AnswerType |
| 4. | updated\_at merupakan cap waktu yang menandakan waktu pengubahan entitas AnswerType |
| 5. | deleted\_at merupakan cap waktu yang menandakan waktu penonaktifan entitas AnswerType |

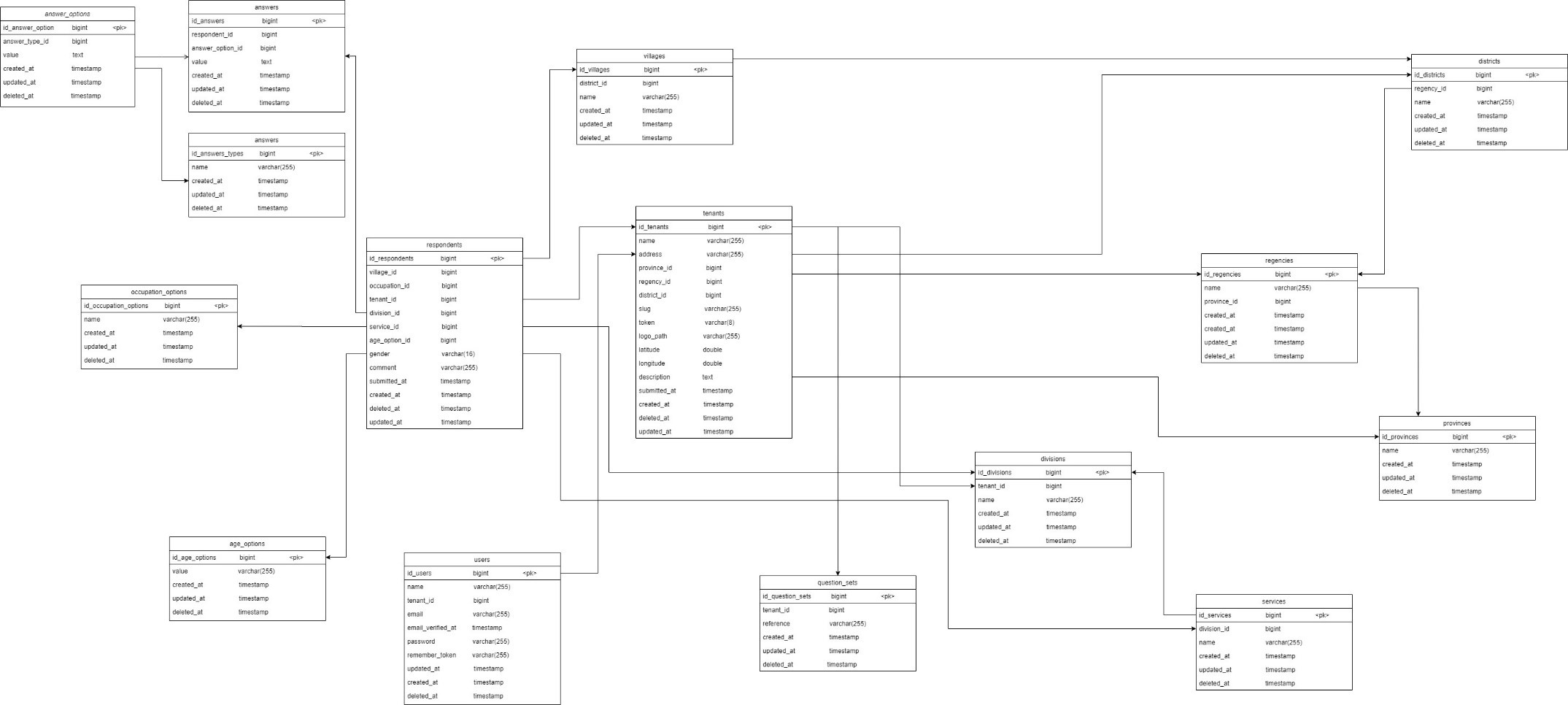
1. AnswerOption. Atribut-atribut dari *entity* AnswerOption adalah:

Table . Atribut Entity AnswerOption

|  |  |
| --- | --- |
| **No.** | **Atribut** |
| 1. | id digunakan sebagai *primary key* dalam entitas AnswerOption |
| 2. | value merupakan nilai isi dari AnswerOption |
| 3. | answer\_type\_id merupakan id dari AnswerType induk dari AnswerOption |
| 4. | created\_at merupakan cap waktu yang menandakan waktu pembuatan entitas AnswerOption |
| 5. | updated\_at merupakan cap waktu yang menandakan waktu pengubahan entitas AnswerOption |
| 6. | deleted\_at merupakan cap waktu yang menandakan waktu penonaktifan entitas AnswerOption |

### Conceptual Data Model

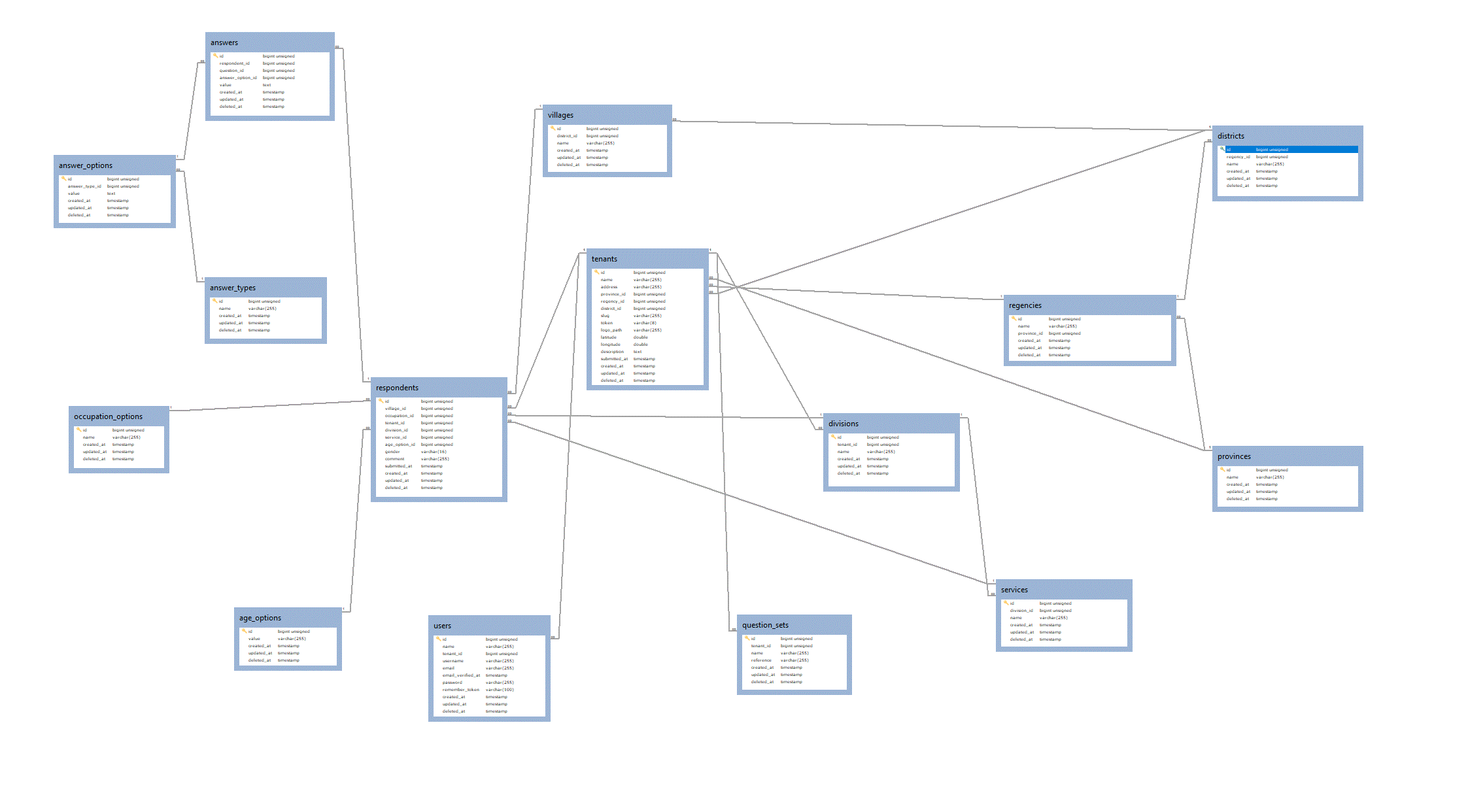
Pada conceptual data model dijelaskan hubungan antar data dalam Sistem Informasi SUKEMAS yang akan dikembangkan dan hubungan yang terjadi antara tabel pada basis data. Hubungan tersebut dapat dilihat pada Gambar 2.



Gambar . Conceptual Data Model

### Physical Data Model

Pada physical data model diGambarkan tabel master dan tabel lain yang saling berhubungan berbeda dengan conceptual data model. Pada physical data model, setiap tabel yang berelasi many to many menghasilkan tabel baru. Hubungan tersebut dapat dilihat pada Gambar 3. Physical Data Model di bawah ini.



Gambar . Physical Data Model

### Tables

Deskripsi dari tabel-tabel yang ada pada basis data dapat dilihat pada Table 22.

Table . Tabel pada Basis Data

|  |  |  |
| --- | --- | --- |
| **Nama Tabel** | **PK** | **Deskripsi Isi** |
| provinces | id | Berfungsi untuk menyimpan data yang mencakup province\_id dan nama dari suatu Provinsi dimana Instansi yang ingin diberi penilaian. |
| regincies | id | Berfungsi untuk menyimpan data yang mencakup provinces\_id dan name |
| districts | id | Berfungsi untuk menyimpan data yang mencakup regency\_id dari suatu Instansi yang digunkan untuk membedakan Instansi yang satu dengan yang lainnya. |
| villages | id | Berfungsi untuk menyimpan data yang mencakup district\_id dan nama dari suatu daerah dimana Instansi yang ingin diberi penilaian. |
| tenants | id | Berfungsi untuk menyimpan data yang mencakup nama tenant, address tenant, province\_id, regency\_id, district\_id, slug, token, logo\_path, latitude, longitude, description, dan submitted\_at dari sebuah Instansi |
| divisions | id | Berfungsi untuk menyimpan data yang mencakup tenant\_id yang digunakan untuk menyimpan divisi apa saja yang ada pada Instansi tertentu. |
| services | id | Berfungsi untuk menyimpan data yang mencakup division\_id yang dimana division\_id ini digunkan untuk membedakan layanan apa saja yang ada pada divisi satu dengan divisi lainnya. |
| users | id | Berfungsi untuk menyimpan data yang memncakup tenant\_id, nama, username, email, email\_verified\_at, dan password dari si user. |
| occupation\_options | id | Berfungsi untuk menyimpan data yang mencakup apa pekerjaan dari seorang respondent yang telah menerima pelayanan dan hendak mengisi kuesioner pada suatu Instansi. |
| answer\_types | id | Berfungsi untuk menyimpan opsi jawaban yang akan digunakan oleh si respondent sebelum menjawab kuesioner. |
| answer\_options | id | Berfungsi untuk menyimpan data yang mencakup answerType\_id yang telah dipilih oleh si respondent pada saat mengisi kuesioner. |
| question\_sets | id | Berfungsi untuk menyimpan data yang mencakup tenant-id, name, dan reference. reference yang dimaksud adalah suatu aturan yang digunakan dalam suatu Negara untuk melakukan survey kepuasan terhadap pelayanan pada suatu Instansi. |
| questions | id | Berfungsi untuk menyimpan data yang mencakup answerType\_id, questionSet\_id, question dan order. |
| age\_options | id | Berfungsi untuk menyimpan opsi umur yang akan digunakan oleh si respondent sebelum mengisi kuesioner. |
| respondents | id | Berfungsi untuk menyimpan data yang mencakup village\_id, occupation\_id, tenant\_id, service\_id, ageOption\_id, gender pada saat melalukan pengisian kuesioner. |
| answers | id | Berfungsi untuk menyimpan data yang mencakup respondent\_id, question\_id, dan answerOption\_id. |
| migrations | id | Berfungsi untuk menyimpan data yang mencakup migration\_id dan batch |
| personal\_access\_tokens | id | Berfungsi untuk menyimpan data yang mencakup id, tokenable\_type, tokenable\_id, name, token, abilities, last\_used\_at, created\_at, update\_at. |

## Functional Decomposition

Dekomposisi dari setiap fungsi pada Siatem Informasi SUKEMAS yang akan dibuat dapat dilihat pada Table 23.

Table . Functional Decomposition

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **No.Fungsi** | **Fungsi/Proses** | **Tabel/Data Input** | | **Tabel/Data Output** | **Deskripsi** |
| SyRS-FR-01 | Registration for Instance and Operator | | Data Instansi:  (Nama, Alamat, Logo)  Tabel provinces  Tabel regincies  Tabel villages  Data Operator (Nama, Email, Password). | Data Instansi di simpan di tabel tenants.  Data Operator di simpan di tabel users. | Digunakan untuk proses registrasi Instansi dan Operator yang dilakukan oleh operator pada Instansi tertentu. |
| SyRS-FR-02 | Login to SUKEMAS | | Name  Email  Password | Nama, Email, Password di simpan di tabel users. | Digunakan untuk memvalidasi Instansi dan Operator nyang telah melakukan registrasi agar dapat mengakses dan masuk ke Sistem Informasi SUKEMAS. |
| SyRS-FR-03 | Adding Question Sets and Questions | | Tabel question set  Tabel Questions  Tabel answers | Berhasil disimpan pada tabel question\_sets, tabel question, dan tabel answers. | Digunakan untuk proses menambahkan question set pada Sistem Informasi SUKEMAS. |
| SyRS-FR-04 | Select Question Set | | Tabel question\_sets | Disimpan pada tabel question\_sets. | Digunakan untuk memilih question set yang akan digunakan untuk menentukan pertanyaan apa yang akan digunakan dalam suatu Instansi tertentu. |
| SyRS-FR-05 | Adding New Operator | | Nama  Email Operator Baru | Nama dan email operator baru berhasil disimpan pada tabel users. | Digunakan untuk menambahkan operator baru yang dilakukan oleh Operator Utama pada Instansi tertentu. |
| SyRS-FR-06 | Operator Account Deactivation | | Password | Berhasil disimpan pada tabel users. | Digunakan untuk mendeaktivasikan akun dari operator yang dilakukan oleh Operator yang sedang Login pada Sistem Informasi SUKEMAS. |
| SyRS-FR-07 | Show Report | | Tidak ada inputan ketika menampilkan laporan. | Berhasil dilihat pada statistic. | Digunakan untuk menampilkan report yang dimana report berisi hasil dari semua penilaian survey yang telah dilakukan oleh Respondent yang dilakukan oleh Operator. |
| SyRS-FR-08 | Download the Report in pdf or xlxs format | | Tidak ada inputan ketika melakukan proses pengunduhan laporan. | Berhasil dilihat pada statistic | Digunakan untuk mengunduh report yang dimana report berisi hasil dari semua penilaian survey yang telah dilakukan oleh respondent yang dilakukan oleh Operator. |
| SyRS-FR-09 | Download the Poster | | Tidak ada inputan ketika melakukan proses pengunduhan poster. | Mengunduh poster berhasil dilakukan. | Digunakan untuk mengunduh poster yang berisi QR Code yang dimana poster ini akan diunduh oleh operator dan dicetak untuk dipajang di kantor Instansi. |
| SyRS-FR-10 | Answer the Questionnaire | | Token Instansi  Jenis Kelamin  Usia  Pendidikan  Pekerjaan  Domisili  Saran atau Kritik  Tabel answers | Jenis, Kelamin, Usia, Pendidikan, Domisili, Saran atau Kritik dapat dilihat pada tabel users.  Pekerjaan dapat dilihat pada tabel occupation\_options.  Tabel answers. | Digunakan untuk menjawab pertanyaan-pertanyaan yang terdapat pada kuesioner pada saat melakukan survey di sebuah Instansi. |
| SyRS-FR-11 | Verification Account | | Tidak ada Inputan dari proses verifikasi akun operator | Verifikasi Akun Operator berhasil dilakukan. | Digunakan untuk memverifikasi akun operator yang sudah dibuat dan hanya dapat dilakukan oleh operator yang sudah selesai mendaftarkan akunnya. Ketika selesai mendaftarkan akun, operator bisa langsung mengakses Sistem Informasi SUKEMAS. Akan tetapi, pada sistem akan ditampilkan *pop-up* berisi peringatan agar operator segera memverifikasi akunnya. Peringatan ini hanya akan hilang jika operator sudah memverifikasi akun yang dimilikinya. |
| SyRS-FR-12 | Change Password | | Kata Sandi Lama  Kata Sandi Baru | Kata Sandi Lama dan Kata Sandi Baru dapat dilihat pada tabel users. | Digunakan untuk mengganti password yang hanya dapat dilakukan oleh operator yang sudah mendaftarkan akun sebelumnya. |
| SyRS-FR-13 | Answer the Questionnaire using QR Code | | Token  Jenis Kelamin  Usia  Pendidikan  Pekerjaan  Domisili  Tabel answer | Jenis, Kelamin, Usia, Pendidikan, Domisili  dapat dilihat pada tabel users.  Pekerjaan dapat dilihat pada tabel occupation\_options.  Tabel answers | Digunakan untuk mengisi kuesioner dengan melakukan *scan* QR Code yang tersedia di poster. |
| SyRS-FR-14 | Password Recovery | | Email | Dapat dilihat pada tabel users. | Digunakan untuk dapat kembali mengakses akun ketika operator lupa password dan dilakukan oleh operator yang sudah mendaftarkan akun sebelumnya. |
| SyRS-FR-15 | View Statistics | | Token | Dapat dilihat pada report dan statistics | Digunakan untuk melihat statistik dari sebuah Instansi dan dilakukan oleh masyarakat umum. |

# Detail Design Description

Bab ini akan membahas struktur tabel yang akan dibangun untuk Aplikasi Survey Kepuasan Masyarakat. Pada bab ini akan dibahas juga spesifikasi setiap fungsifungsi yang terdapat pada sistem.

## Table Structure

Pada bagian ini akan dijelaskan spesifikasi dari seluruh tabel yang saling berelasi yang digunakan untuk membangun untuk Aplikasi Survey Kepuasan Masyarakat.

### Table password\_reset\_tokens

Identifikasi/Nama : password\_reset\_tokens

Deskripsi Isi : email, token, created\_at

Jenis : tabel data transaksi

Primary Key : email

Detail Description : -

Table . Tabel password\_reset\_tokens

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| email | berisikan email yang digunakan sebagai kode unik atau PK | varchar (255) | no | PK |
| token | atribut yang digunakan sebagai kunci atau kode unik yang digunakan dalam proses autentikasi atau otorisasi untuk mengidentifikasi dan mengakses tenant. | varchar (255) | no |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |

SQL script:

CREATE TABLE `password\_reset\_tokens` (

`email` varchar(255) NOT NULL,

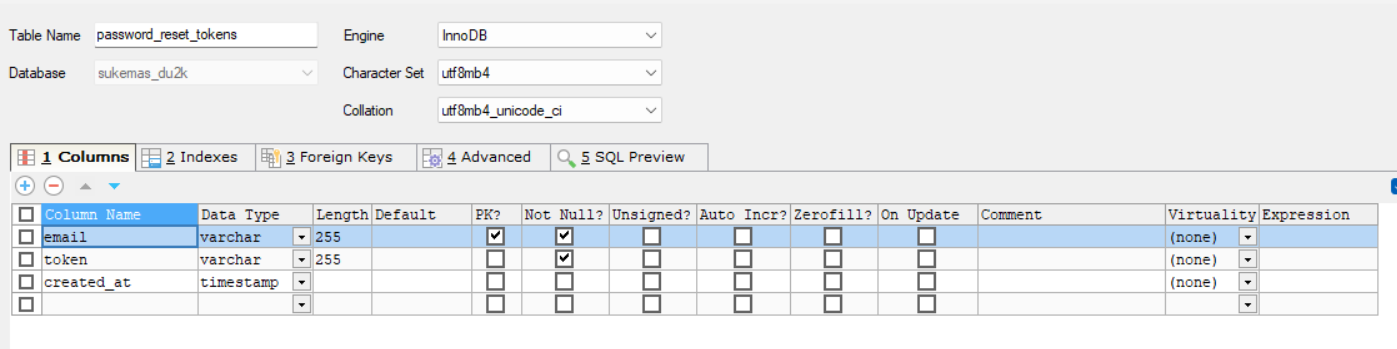
`token` varchar(255) NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

PRIMARY KEY (`email`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan:



Gambar . Database password\_reset\_tokens

### Table failed\_jobs

Identifikasi/Nama : failed\_jobs

Deskripsi Isi : id, uuid, connection, queue, payload, exception, failed\_at.

Jenis : tabel data transaksi

Primary Key : id

Detail Description : -

Table . Tabel failed\_jobs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | kode unik dari setiap failed\_jobs. | bigint (20) | no | PK |
| uuid | atribut yang digunakan untuk menyimpan sebuah nilai unik dan acak yang digunakan sebagai identifikasi untuk sebuah entitas atau objek dalam sistem. | varchar (255) | no |  |
| connection | atribut yang digunakan dalam konteks sistem antrian atau sistem pengiriman pesan untuk menunjukkan koneksi atau sumber daya yang digunakan untuk mengirimkan atau menerima pesan atau tugas yang terkait. | text | no |  |
| queue | atribut yang digunakan dalam konteks sistem antrian atau sistem pengiriman pesan untuk menunjukkan antrian atau daftar tugas yang harus dikerjakan atau diproses. | text | no |  |
| payload | yang digunakan dalam konteks sistem antrian atau sistem pengiriman pesan untuk menyimpan data atau informasi yang terkait dengan tugas atau pesan yang akan diproses. | longtext | no |  |
| exception | atribut yang digunakan untuk menyimpan informasi tentang kesalahan atau pengecualian yang terjadi saat sistem memproses sebuah tugas atau pesan. | longtext | no |  |
| failed\_at | yang menunjukkan waktu atau tanggal ketika sebuah tugas atau pesan gagal diproses oleh sistem. | longtext | no |  |

SQL script:

CREATE TABLE `failed\_jobs` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`uuid` varchar(255) NOT NULL,

`connection` text NOT NULL,

`queue` text NOT NULL,

`payload` longtext NOT NULL,

`exception` longtext NOT NULL,

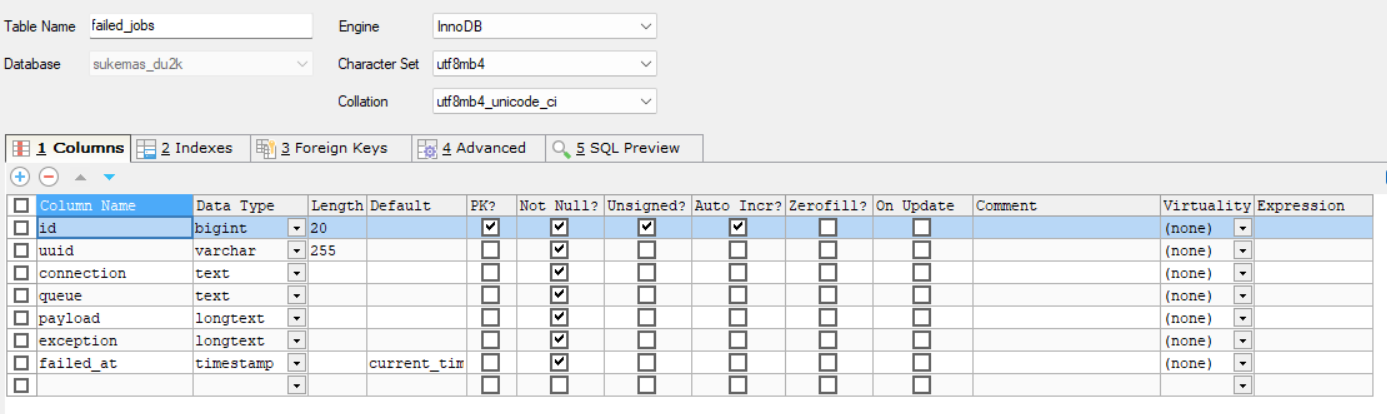
`failed\_at` timestamp NOT NULL DEFAULT current\_timestamp(),

PRIMARY KEY (`id`),

UNIQUE KEY `failed\_jobs\_uuid\_unique` (`uuid`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan:

****

Gambar . Database failed\_jobs

### Table provinces

Identifikasi/Nama : provinces

Deskripsi Isi : id, name, created\_at, updated\_at, deleted\_at.

Jenis : tabel data reference

Primary Key : id

Detail Description : -

Table . Tabel provinces

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | id merupakan kode unik dari setiap province. | bigint (20) | no | PK |
| name | name berisikan nama province dari sebuah tenant. | varchar (255) | no |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | yes |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | yes |  |

SQL script:

CREATE TABLE `provinces` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`name` varchar(255) NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

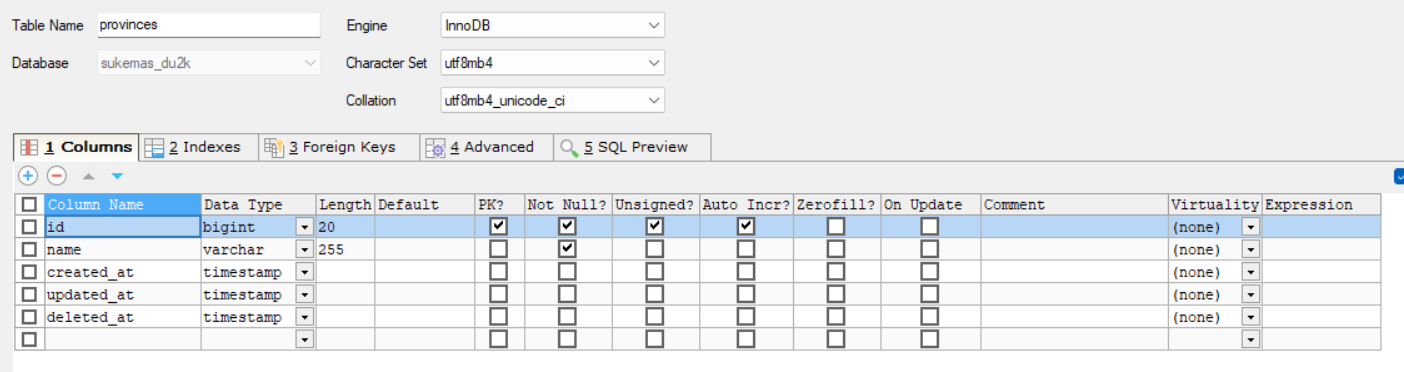
`updated\_at` timestamp NULL DEFAULT NULL,

`deleted\_at` timestamp NULL DEFAULT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=10 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan:



Gambar . Database provinces

### Table regencies

Identifikasi/Nama : regencies

Deskripsi Isi : id, province\_id, name, created\_at, updated\_at, deleted\_at.

Jenis : tabel data reference

Primary Key : id

Detail Description : berelasi dengan province\_id.

Table . Tabel regencies

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | id merupakan kode unik dari setiap regency. | bigint (20) | no | PK |
| province\_id | province\_id merupakan foreign key. | bigint (20) | no | FK |
| name | name berisikan nama dari regency. | varchar (255) | no |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | yes |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | yes |  |

SQL script:

CREATE TABLE `regencies` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`name` varchar(255) NOT NULL,

`province\_id` bigint(20) unsigned NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL,

`deleted\_at` timestamp NULL DEFAULT NULL,

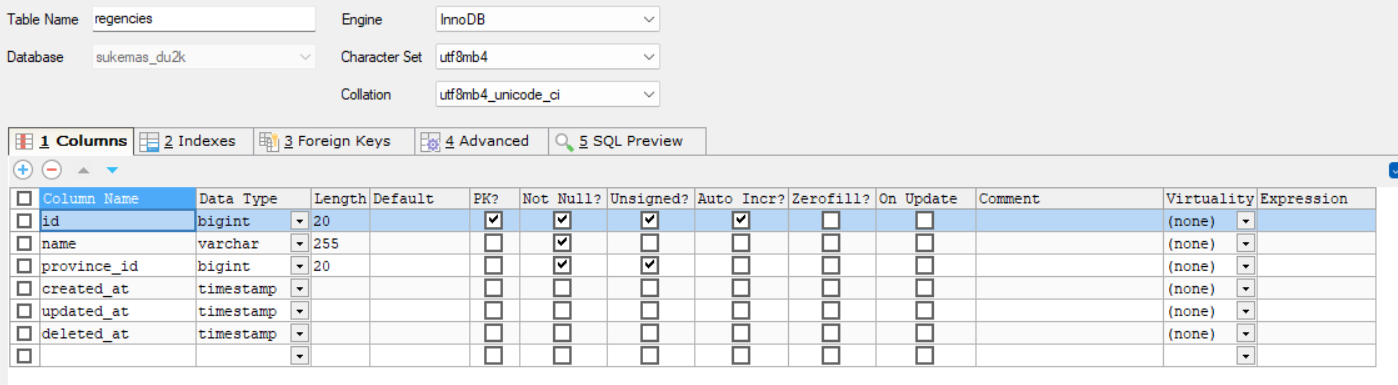
PRIMARY KEY (`id`),

KEY `regencies\_province\_id\_foreign` (`province\_id`),

CONSTRAINT `regencies\_province\_id\_foreign` FOREIGN KEY (`province\_id`) REFERENCES `provinces` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=10 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan:



Gambar . Database regencies

### Table districts

Identifikasi/Nama : districs

Deskripsi Isi : id, regency\_id, name, created\_at, updated\_at, deleted\_at.

Jenis : tabel data reference

Primary Key : id

Detail Description : berelasi dengan regency\_id.

Table . Tabel districts

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | berisikan kode unik dari setiap district. | bigint (20) | no | PK |
| regency\_id | regency\_id merupakan foreign key. | text | no | FK |
| name | nama dari setiap district. | varchar (255) | no |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | yes |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | yes |  |

SQL script:

CREATE TABLE `districts` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`regency\_id` bigint(20) unsigned NOT NULL,

`name` varchar(255) NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL,

`deleted\_at` timestamp NULL DEFAULT NULL,

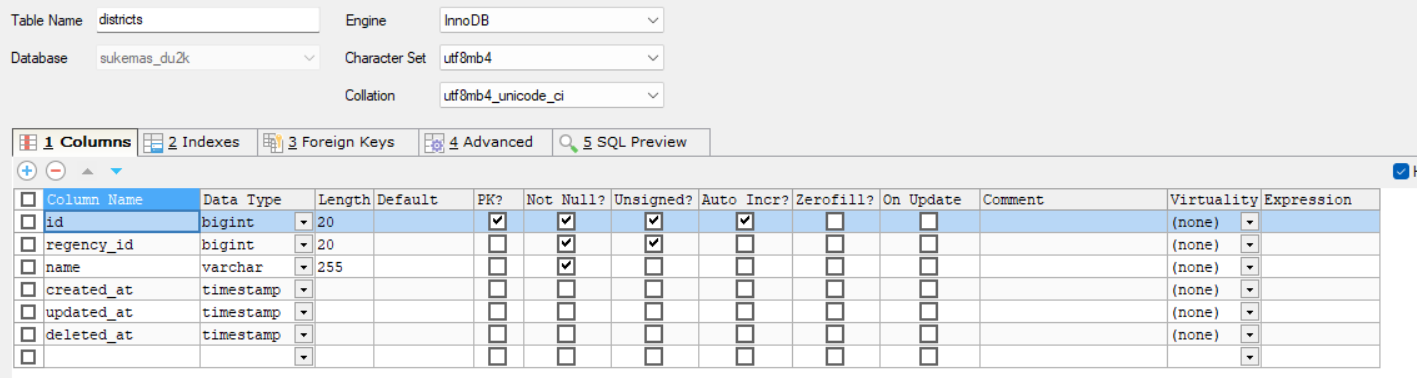
PRIMARY KEY (`id`),

KEY `districts\_regency\_id\_foreign` (`regency\_id`),

CONSTRAINT `districts\_regency\_id\_foreign` FOREIGN KEY (`regency\_id`) REFERENCES `regencies` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=11 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan :



Gambar . Database districts

### Table villages

Identifikasi/Nama : villages

Deskripsi Isi : id, district\_id, name, created\_at, updated\_at, deleted\_at.

Jenis : tabel data reference

Primary Key : id

Detail Description : berelasi dengan district\_id.

Table . Tabel villages

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | kode unik dari setiap village. | bigint (20) | no | PK |
| dictrict\_id | foreign key. | bigint (20) | no | FK |
| name | berisikan nama villages. | varchar (255) | no |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | yes |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | yes |  |

SQL script:

CREATE TABLE `villages` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`district\_id` bigint(20) unsigned NOT NULL,

`name` varchar(255) NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL,

`deleted\_at` timestamp NULL DEFAULT NULL,

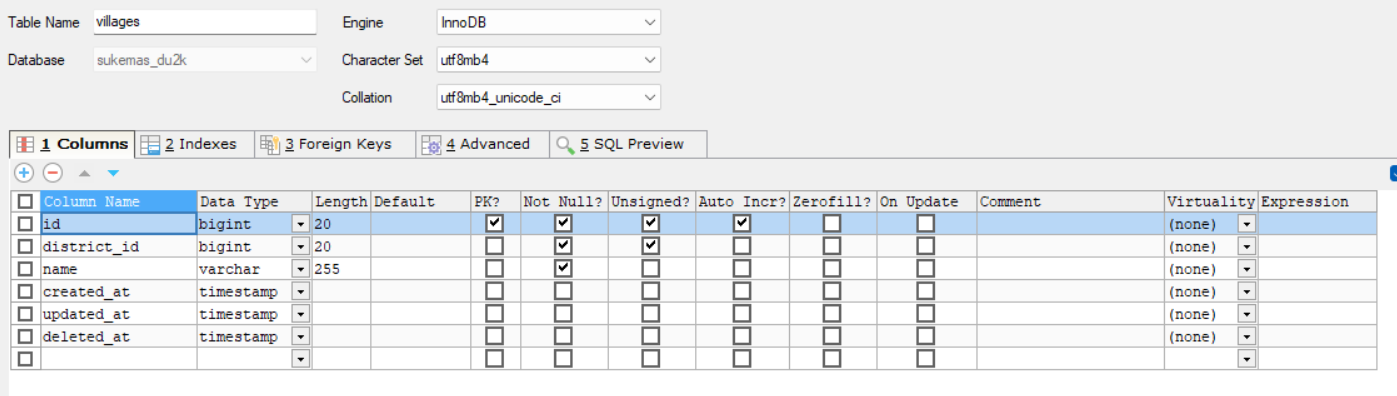
PRIMARY KEY (`id`),

KEY `villages\_district\_id\_foreign` (`district\_id`),

CONSTRAINT `villages\_district\_id\_foreign` FOREIGN KEY (`district\_id`) REFERENCES `districts` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=11 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan :



Gambar . Database villages

### Table tenants

Identifikasi/Nama : tenants

Deskripsi Isi : id, name, address, province\_id, regency\_id, district\_id,question\_set\_id,slug, token, logo\_path, latitude, longitude, description, submitted\_at, created\_at, deleted\_at.

Jenis : tabel data induk

Primary Key : id

Detail Description : berelasi dengan province\_id, district\_id, regency\_id, question\_set\_id.

Table . Tabel tenants

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | berisikan id dari tabel tenants yang menjadi primary key | bigint (20) | no | PK |
| name | berisikan nama instansi/ kantor | varchar (255) | no |  |
| address | berisi alamat dari instansi atau kantor. | varchar (255) | no |  |
| province\_id | merupakan foreign key | bigint (20) | no | FK |
| regency\_id | merupakan foreign key | bigint (20) | no | FK |
| district\_id | merupakan foreign key | bigint (20) | no | FK |
| question\_set\_id | merupakan foreign key | bigint (20) |  | FK |
| slug | atribut yang digunakan dalam konteks web development dan content management system (cms) untuk menunjukkan bagian unik dari url suatu halaman. | varchar (255) | no |  |
| token | berisikan token yang digunakan untuk setiap instansi atau tenants | varchar (8) | no |  |
| logo\_path | berisikan link dari logo instansi atau tenant | varchar (255) | no |  |
| latitude | menunjukkan lokasi geografis dari instansi atau tenant | double | no |  |
| longitude | menunjukkan lokasi geografis dari instansi atau tenant | double | no |  |
| description | merupakan penjelasan singkat tentang instansi (tenant) | text | no |  |
| submitted\_at | atribut yang menunjukkan waktu atau tanggal saat data atau informasi telah dikirim atau diajukan. | timestamp | yes |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | yes |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | yes |  |

SQL script:

CREATE TABLE `tenants` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`name` varchar(255) NOT NULL,

`address` varchar(255) NOT NULL,

`province\_id` bigint(20) unsigned NOT NULL,

`regency\_id` bigint(20) unsigned NOT NULL,

`district\_id` bigint(20) unsigned NOT NULL,

`question\_set\_id` bigint(20) unsigned DEFAULT NULL,

`slug` varchar(255) NOT NULL,

`token` varchar(8) NOT NULL,

`logo\_path` varchar(255) NOT NULL,

`latitude` double NOT NULL,

`longitude` double NOT NULL,

`description` text NOT NULL,

`submitted\_at` timestamp NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL,

`deleted\_at` timestamp NULL DEFAULT NULL,

PRIMARY KEY (`id`),

KEY `tenants\_province\_id\_foreign` (`province\_id`),

KEY `tenants\_regency\_id\_foreign` (`regency\_id`),

KEY `tenants\_district\_id\_foreign` (`district\_id`),

KEY `tenants\_question\_set\_id\_foreign` (`question\_set\_id`),

CONSTRAINT `tenants\_district\_id\_foreign` FOREIGN KEY (`district\_id`) REFERENCES `districts` (`id`),

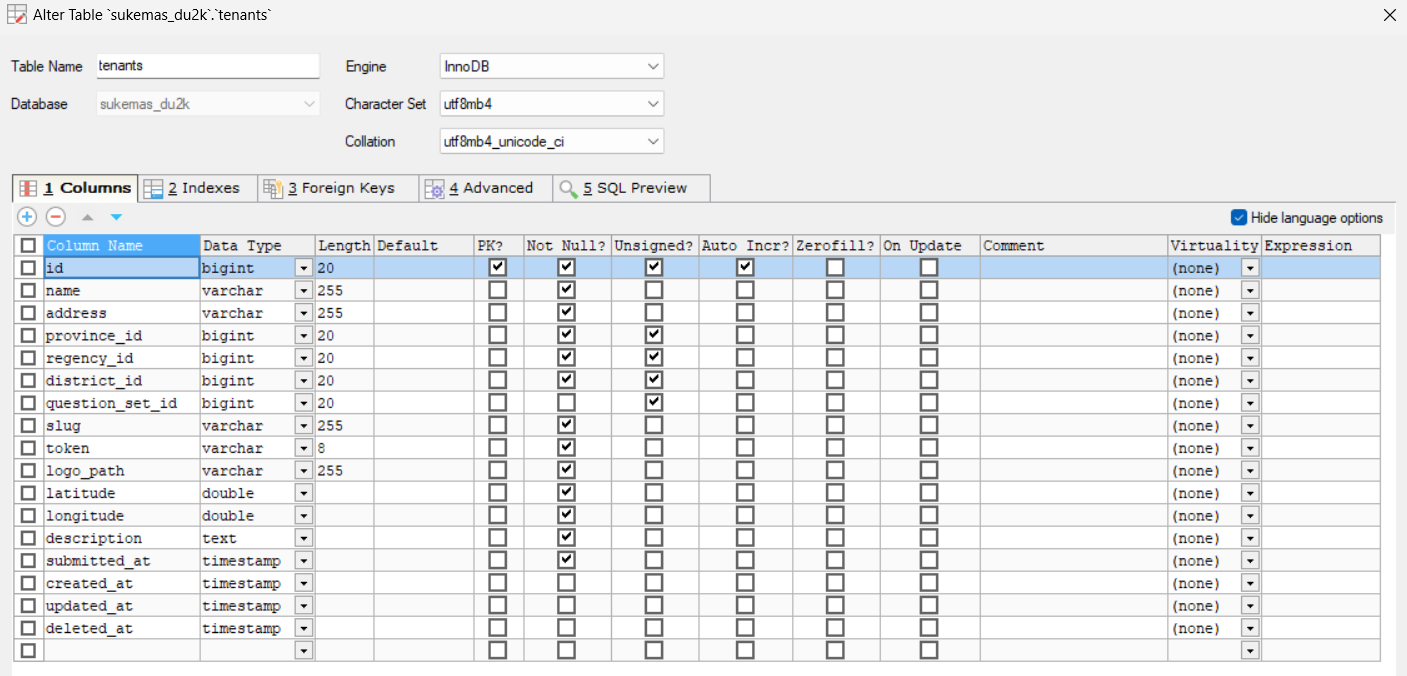
CONSTRAINT `tenants\_province\_id\_foreign` FOREIGN KEY (`province\_id`) REFERENCES `provinces` (`id`),

CONSTRAINT `tenants\_question\_set\_id\_foreign` FOREIGN KEY (`question\_set\_id`) REFERENCES `question\_sets` (`id`),

CONSTRAINT `tenants\_regency\_id\_foreign` FOREIGN KEY (`regency\_id`) REFERENCES `regencies` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan:



Gambar . Database tenants

### Table divisions

Identifikasi/Nama : divisions

Deskripsi Isi : id, tenant\_id, name, created\_at, updated\_at, deleted\_at.

Jenis : tabel data reference

Primary Key : id

Detail Description : berelasi dengan tenant\_id.

Table . Tabel divisions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | kode unik dari setiap division. | bigint (20) | no | PK |
| tenant\_id | merupakan foreign key. | bigint (20) | no | FK |
| name | berisikan nama division. | varchar (255) | no |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | yes |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | yes |  |

SQL script:

CREATE TABLE `divisions` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`tenant\_id` bigint(20) unsigned NOT NULL,

`name` varchar(255) NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL,

`deleted\_at` timestamp NULL DEFAULT NULL,

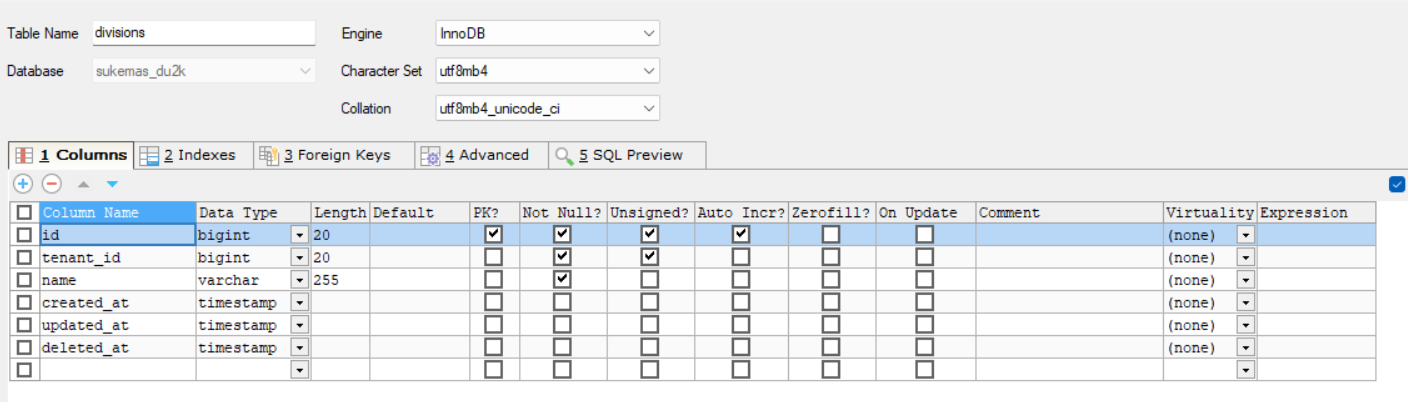
PRIMARY KEY (`id`),

KEY `divisions\_tenant\_id\_foreign` (`tenant\_id`),

CONSTRAINT `divisions\_tenant\_id\_foreign` FOREIGN KEY (`tenant\_id`) REFERENCES `tenants` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan:



Gambar . Database divisions

### Table services

Identifikasi/Nama : services

Deskripsi Isi : id, division\_id, name, created\_at, updated\_at, deleted\_at.

Jenis : tabel data reference

Primary Key : id

Detail Description : berelasi dengan division\_id.

Table . Tabel services

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | kode unik dari setiap services. | bigint (20) | no | PK |
| division\_id | merupakan foreign key. | bigint (20) | no | FK |
| name | berisikan nama-nama dari setiap layanan. | varchar (255) | no |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | yes |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | yes |  |

SQL script:

CREATE TABLE `services` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`division\_id` bigint(20) unsigned NOT NULL,

`name` varchar(255) NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL,

`deleted\_at` timestamp NULL DEFAULT NULL,

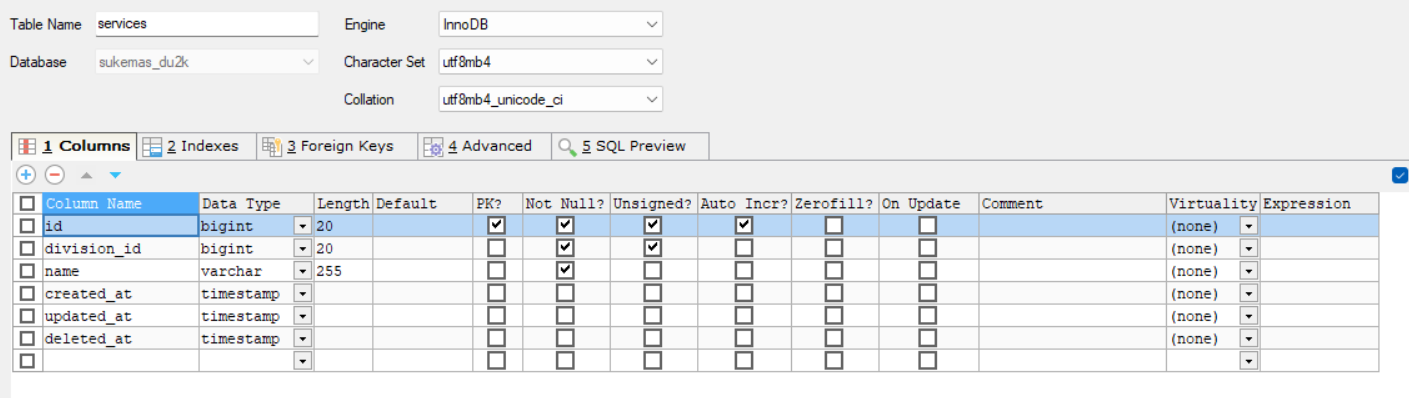
PRIMARY KEY (`id`),

KEY `services\_division\_id\_foreign` (`division\_id`),

CONSTRAINT `services\_division\_id\_foreign` FOREIGN KEY (`division\_id`) REFERENCES `divisions` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=12 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan:



Gambar . Database services

### Table users

Identifikasi/Nama : users

Deskripsi Isi : id, tenant\_id, name, username, email, email\_verified\_at, password, remember\_token , submitted\_at, created\_at, updated\_at, deleted\_at.

Jenis : tabel data reference

Primary Key : id

Detail Description : berelasi dengan tenant\_id

Table . Tabel users

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | berisi kode unik dari setiap user (operator). | bigint (20) | no | PK |
| tenant\_id | atribut yang mengacu pada identitas instansi | bigint (20) | no | FK |
| name | atribut yang mengacu pada nama lengkap atau nama asli dari seseorang. | varchar (255) | no |  |
| username | atribut yang digunakan untuk menyimpan nama pengguna atau username yang digunakan untuk login. | varchar (255) | no |  |
| email | atribut yang digunakan untuk menyimpan alamat email seseorang | varchar (255) | no |  |
| email\_verified\_at | atribut yang menunjukkan waktu atau tanggal saat alamat email seseorang telah diverifikasi. | timestamp | yes |  |
| password | atribut yang digunakan untuk menyimpan kata sandi atau password yang digunakan untuk login atau mengakses suatu platform atau aplikasi. | varchar (255) | no |  |
| remember\_token | atribut yang digunakan untuk menyimpan token atau kode unik yang digunakan untuk menjaga sesi login seseorang agar tetap terbuka selama jangka waktu tertentu, bahkan setelah mereka menutup atau mematikan browser mereka. token ini biasanya digunakan untuk meningkatkan kenyamanan pengguna saat mengakses platform atau aplikasi, karena mereka tidak perlu login setiap kali mereka membuka atau mengaksesnya. | varchar (100) | no |  |
| submitted\_at | atribut yang menunjukkan waktu atau tanggal saat data atau informasi telah dikirim atau diajukan. | timestamp | yes |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | yes |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | yes |  |

SQL script:

CREATE TABLE `users` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`name` varchar(255) NOT NULL,

`tenant\_id` bigint(20) unsigned NOT NULL,

`username` varchar(255) DEFAULT NULL,

`email` varchar(255) NOT NULL,

`email\_verified\_at` timestamp NULL DEFAULT NULL,

`password` varchar(255) DEFAULT NULL,

`remember\_token` varchar(100) DEFAULT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL,

`deleted\_at` timestamp NULL DEFAULT NULL,

PRIMARY KEY (`id`),

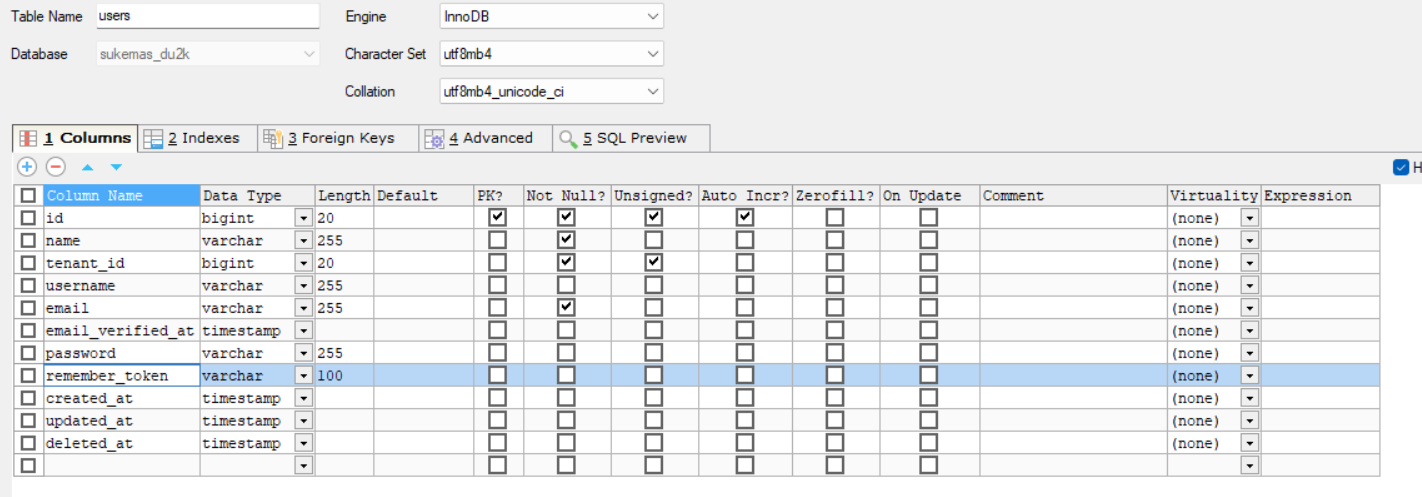
UNIQUE KEY `users\_email\_unique` (`email`),

KEY `users\_tenant\_id\_foreign` (`tenant\_id`),

CONSTRAINT `users\_tenant\_id\_foreign` FOREIGN KEY (`tenant\_id`) REFERENCES `tenants` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=7 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan:



Gambar . Database users

### Table occupation\_options

Identifikasi/Nama : occupation\_options

Deskripsi Isi : id, name, created\_at, updated\_at, deleted\_at.

Jenis : tabel data induk

Primary Key : id

Detail Description : -

Table . Tabel occupation\_options

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | berisikan kode unik dari setiap occupation\_options | bigint (20) | no | PK |
| name | berisikan nama occupation\_options. | varchar (255) | no |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | yes |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | yes |  |

SQL script:

CREATE TABLE `occupation\_options` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`name` varchar(255) NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

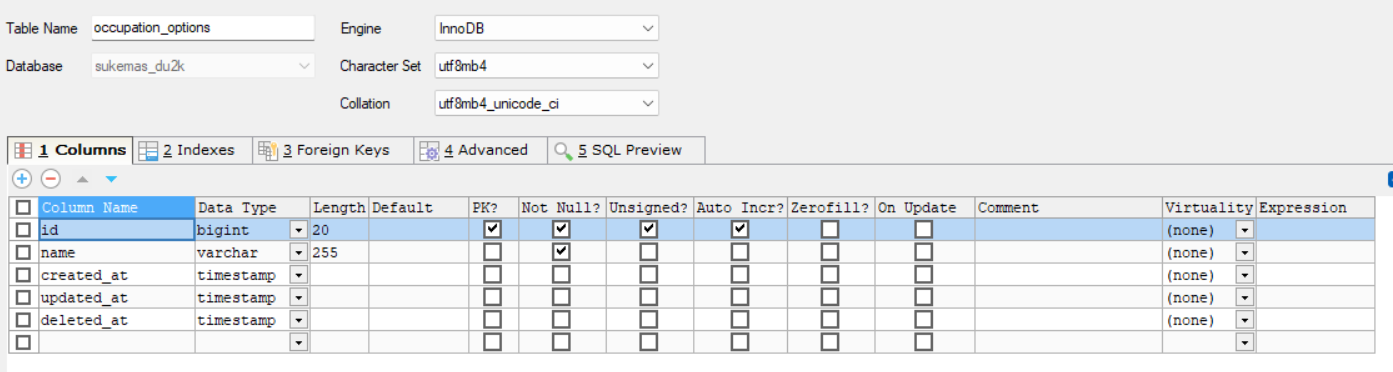
`updated\_at` timestamp NULL DEFAULT NULL,

`deleted\_at` timestamp NULL DEFAULT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=6 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan:



Gambar . Database occupation\_options

### Table answer\_types

Identifikasi/Nama : answer\_types

Deskripsi Isi : id, name, created\_at, updated\_at, deleted\_at.

Jenis : tabel data reference

Primary Key : id

Detail Description : -

Table . Tabel answer\_types

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | berisikan kode unik dari answer\_type. | Bigint (20) | NO | PK |
| name | berisikan type\_name dari answer | Varchar (20) | NO |  |
| created\_at | tanggal dan waktu record di insert | timestamp | YES |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | YES |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | YES |  |

SQL script:

CREATE TABLE `answer\_types` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`name` varchar(255) NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

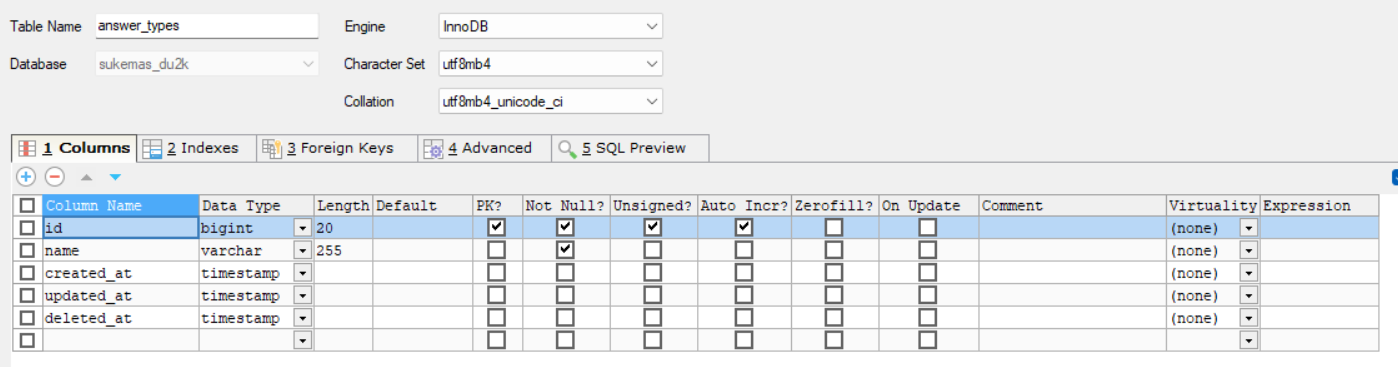
`updated\_at` timestamp NULL DEFAULT NULL,

`deleted\_at` timestamp NULL DEFAULT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=9 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan :



Gambar . Database answer\_types

### Table answer\_options

Identifikasi/Nama : answer\_options

Deskripsi Isi : id, answer\_type\_id, value, created\_at, updated\_at, deleted\_at.

Jenis : tabel data reference

Primary Key : id

Detail Description : berelasi dengan answer\_type\_id.

Table . Tabel answer\_options

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | berisi kode unik dari answer\_options | bigint (20) | no | PK |
| answer\_type\_id | merupakan foreign key. | bigint (20) | no | FK |
| value | berisikan nilai dari answer\_options. | text | no |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | yes |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | yes |  |

SQL script:

CREATE TABLE `answer\_options` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`answer\_type\_id` bigint(20) unsigned NOT NULL,

`value` text NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL,

`deleted\_at` timestamp NULL DEFAULT NULL,

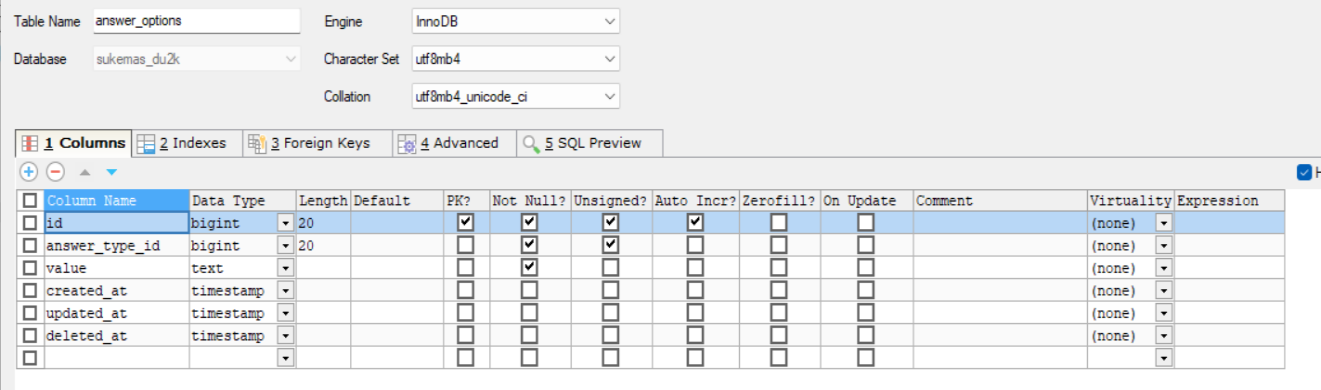
PRIMARY KEY (`id`),

KEY `answer\_options\_answer\_type\_id\_foreign` (`answer\_type\_id`),

CONSTRAINT `answer\_options\_answer\_type\_id\_foreign` FOREIGN KEY (`answer\_type\_id`) REFERENCES `answer\_types` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=5 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan:



Gambar . Database answer\_options

### Table question\_sets

Identifikasi/Nama : question\_sets

Deskripsi Isi : id, tenant\_id, name, reference, created\_at, updated\_at, deleted\_at.

Jenis : tabel data reference

Primary Key : id

Detail Description : berelasi dengan tenant\_id.

Table . Table question\_sets

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | berisikan kode unik dari setiap question\_sets. | bigint (20) | no | PK |
| tenant\_id | merupakan foreign key. | bigint (20) | no | FK |
| name | berisi nama dari question set. | varchar (255) | no |  |
| reference | berisikan referensi yang digunakan untuk membuat pertanyaan survey. | varchar (255) | no |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | yes |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | yes |  |

SQL script:

CREATE TABLE `question\_sets` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`tenant\_id` bigint(20) unsigned NOT NULL,

`name` varchar(255) NOT NULL,

`reference` varchar(255) NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL,

`deleted\_at` timestamp NULL DEFAULT NULL,

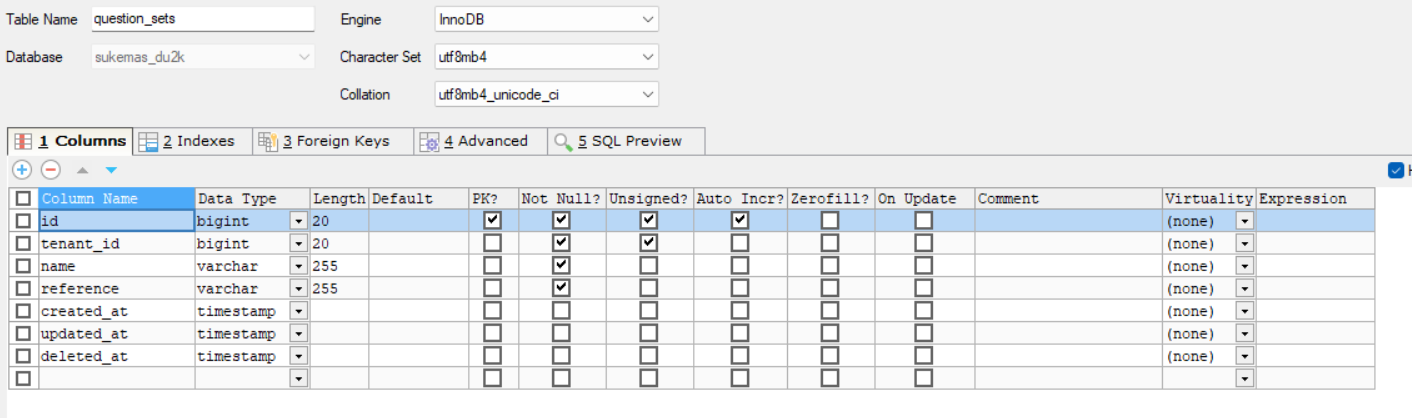
PRIMARY KEY (`id`),

KEY `question\_sets\_tenant\_id\_foreign` (`tenant\_id`),

CONSTRAINT `question\_sets\_tenant\_id\_foreign` FOREIGN KEY (`tenant\_id`) REFERENCES `tenants` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan:



Gambar . Database question\_sets

### Table questions

Identifikasi/Nama : questions

Deskripsi Isi : id, answer\_type\_id, question\_set\_id, question, order, created\_at, updated\_at, deleted\_at.

Jenis : tabel data reference

Primary Key : id

Detail Description : berelasi dengan answer\_type\_id, question\_set\_id.

Table . Tabel questions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | berisikan kode unik dari questions. | bigint (20) | no | PK |
| answer\_type\_id | merupakan foreign key. | bigint (20) | no | FK |
| question\_set\_id | merupakan foreign key. | bigint (20) | no | FK |
| question | berisikan pertanyaan-pertanyaan yang digunakan untuk survey. | varchar (255) | no |  |
| order | berisikan jumlah order pertanyaan yang diberikan oleh survey. | varchar (255) | no |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | yes |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | yes |  |

SQL script:

CREATE TABLE `questions` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`answer\_type\_id` bigint(20) unsigned NOT NULL,

`question\_set\_id` bigint(20) unsigned NOT NULL,

`question` varchar(255) NOT NULL,

`order` varchar(255) NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL,

`deleted\_at` timestamp NULL DEFAULT NULL,

PRIMARY KEY (`id`),

KEY `questions\_answer\_type\_id\_foreign` (`answer\_type\_id`),

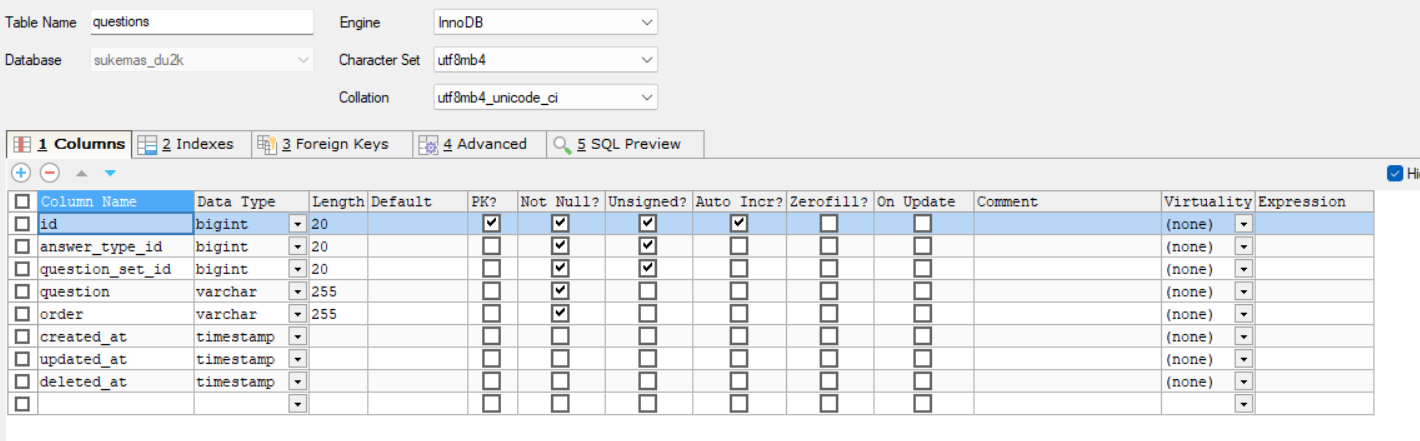
KEY `questions\_question\_set\_id\_foreign` (`question\_set\_id`),

CONSTRAINT `questions\_answer\_type\_id\_foreign` FOREIGN KEY (`answer\_type\_id`) REFERENCES `answer\_types` (`id`),

CONSTRAINT `questions\_question\_set\_id\_foreign` FOREIGN KEY (`question\_set\_id`) REFERENCES `question\_sets` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=10 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan :



Gambar . Database questions

### Table age\_options

Identifikasi/Nama : age\_options

Deskripsi Isi : id, value, created\_at, updated\_at, deleted\_at.

Jenis : tabel data reference

Primary Key : id

Detail Description : -

Table . Tabel age\_options

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | berisikan kode unik dari age options. | bigint (20) | no | PK |
| value | berisikan nilai dari age yang dituliskan dalam bentuk text. | text | no |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | yes |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | yes |  |

SQL script:

CREATE TABLE `age\_options` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`value` varchar(255) NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

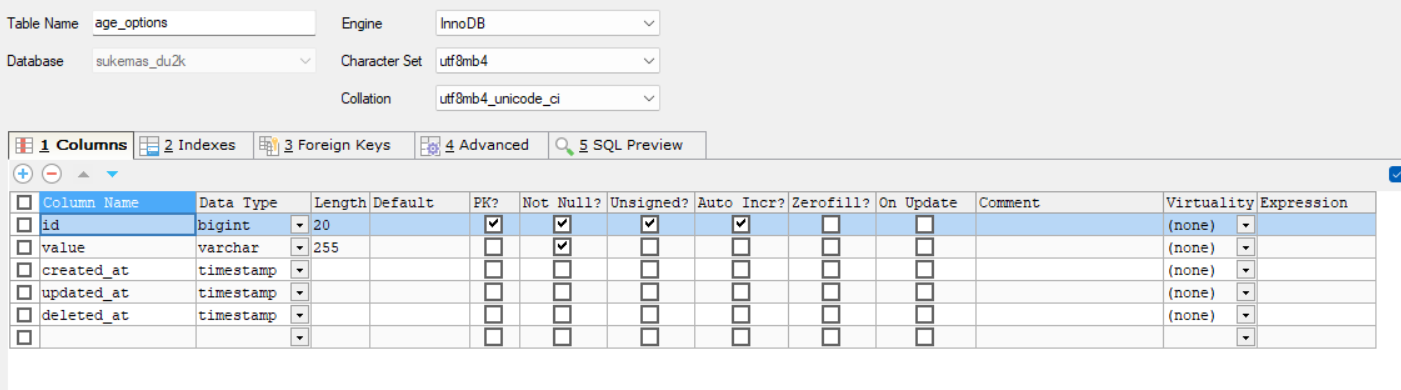
`updated\_at` timestamp NULL DEFAULT NULL,

`deleted\_at` timestamp NULL DEFAULT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=4 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan :



Gambar . Database age\_options

### Table respondents

Identifikasi/Nama : respondents

Deskripsi Isi : id, village\_id, occupation\_id, tenant\_id, division\_id, service\_id, age\_option\_id, gender, comment, submitted\_at, created\_at, updated\_at, deleted\_at.

Jenis : tabel data reference

Primary Key : id

Detail Description : berelasi dengan village\_id, occupation\_id, tenant\_id, division\_id, service\_id, age\_option\_id.

Table . Tabel respondents

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id Field | Deskripsi | Tipe & length | Nullabel | Key |
| id | berisi id dari setiap respondent yang mengisi survey. | bigint (20) | no | PK |
| village\_id | atribut yang mengacu pada identitas unik sebuah desa atau kampung di suatu wilayah atau daerah. | bigint (20) | no |  |
| occupation\_id | atribut yang mengacu pada identitas unik dari jenis pekerjaan atau profesi yang dijalankan oleh seseorang. | bigint (20) | no | FK |
| tenant\_id | atribut yang mengacu pada identitas unik dari instansi | bigint (20) | no | FK |
| division\_id | atribut yang mengacu pada identitas unik dari sebuah divisi yang terdapat pada instansi. | bigint (20) | no | FK |
| service\_id | atribut yang mengacu pada identitas unik dari jenis layanan atau servis yang diberikan oleh sebuah tenant. | bigint (20) |  | FK |
| age\_option\_id | atribut yang mengacu pada identitas unik dari kategori atau opsi umur seseorang, seperti anak-anak, remaja, dewasa, atau lansia. | bigint (20) | no | FK |
| gender | atribut yang mengacu pada jenis kelamin seseorang, seperti laki-laki atau perempuan. | varchar (16) | no |  |
| comment | atribut yang digunakan untuk menyimpan komentar atau catatan tambahan pada data atau informasi tertentu. | varchar (255) | no |  |
| submitted\_at | tanggal dan waktu record telah dikirim atau diajukan. | timestamp | yes |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | yes |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | yes |  |

SQL script:

CREATE TABLE `respondents` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`village\_id` bigint(20) unsigned NOT NULL,

`occupation\_id` bigint(20) unsigned NOT NULL,

`tenant\_id` bigint(20) unsigned NOT NULL,

`division\_id` bigint(20) unsigned NOT NULL,

`service\_id` bigint(20) unsigned NOT NULL,

`age\_option\_id` bigint(20) unsigned NOT NULL,

`gender` varchar(16) NOT NULL,

`comment` varchar(255) NOT NULL,

`submitted\_at` timestamp NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL,

`deleted\_at` timestamp NULL DEFAULT NULL,

PRIMARY KEY (`id`),

KEY `respondents\_village\_id\_foreign` (`village\_id`),

KEY `respondents\_occupation\_id\_foreign` (`occupation\_id`),

KEY `respondents\_tenant\_id\_foreign` (`tenant\_id`),

KEY `respondents\_division\_id\_foreign` (`division\_id`),

KEY `respondents\_service\_id\_foreign` (`service\_id`),

KEY `respondents\_age\_option\_id\_foreign` (`age\_option\_id`),

CONSTRAINT `respondents\_age\_option\_id\_foreign` FOREIGN KEY (`age\_option\_id`) REFERENCES `age\_options` (`id`),

CONSTRAINT `respondents\_division\_id\_foreign` FOREIGN KEY (`division\_id`) REFERENCES `divisions` (`id`),

CONSTRAINT `respondents\_occupation\_id\_foreign` FOREIGN KEY (`occupation\_id`) REFERENCES `occupation\_options` (`id`),

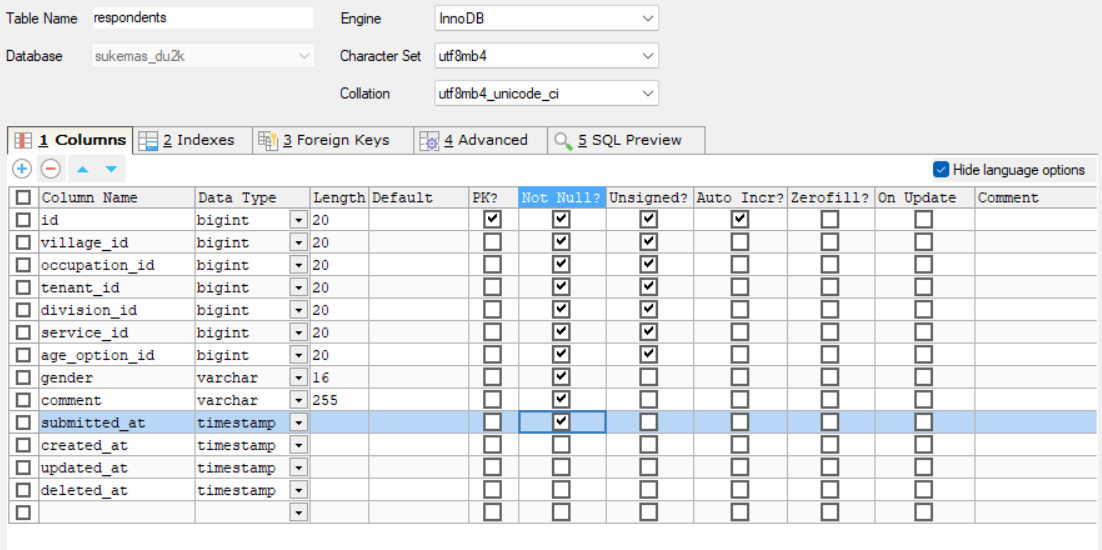
CONSTRAINT `respondents\_service\_id\_foreign` FOREIGN KEY (`service\_id`) REFERENCES `services` (`id`),

CONSTRAINT `respondents\_tenant\_id\_foreign` FOREIGN KEY (`tenant\_id`) REFERENCES `tenants` (`id`),

CONSTRAINT `respondents\_village\_id\_foreign` FOREIGN KEY (`village\_id`) REFERENCES `villages` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=39 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan :



Gambar . Database respondents

### Table answer

Identifikasi/Nama : answers

Deskripsi Isi : id, Respondent\_id, Question\_id , answer\_option\_id, value, created\_at, updated\_at, deleted\_at.

Jenis : tabel data reference

Primary Key : id

Detail Description : berelasi dengan Respondent\_id, Question\_id , answer\_option\_id.

Table . Tabel answer

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | berisi kode unik dari answers. | bigint (20) | no | PK |
| respondent\_id | merupakan foreign key. | bigint (20) | no |  |
| question\_id | merupakan foreign key. | bigint (20) | no |  |
| answer\_option\_id | merupakan foreign key. | bigint (20) | no |  |
| value | berisikan nilai dari answer yang dituliskan dalam bentuk text. | text | no |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | yes |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | yes |  |

SQL script:

CREATE TABLE `answers` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`respondent\_id` bigint(20) unsigned NOT NULL,

`question\_id` bigint(20) unsigned NOT NULL,

`answer\_option\_id` bigint(20) unsigned NOT NULL,

`value` text NOT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL,

`deleted\_at` timestamp NULL DEFAULT NULL,

PRIMARY KEY (`id`),

KEY `answers\_respondent\_id\_foreign` (`respondent\_id`),

KEY `answers\_question\_id\_foreign` (`question\_id`),

KEY `answers\_answer\_option\_id\_foreign` (`answer\_option\_id`),

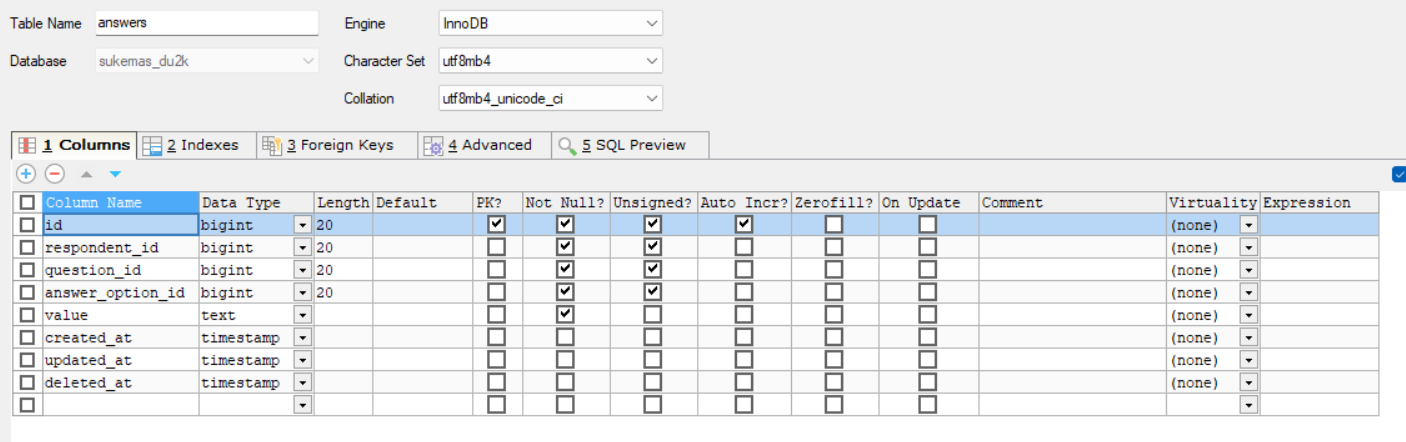
CONSTRAINT `answers\_answer\_option\_id\_foreign` FOREIGN KEY (`answer\_option\_id`) REFERENCES `answer\_options` (`id`),

CONSTRAINT `answers\_question\_id\_foreign` FOREIGN KEY (`question\_id`) REFERENCES `questions` (`id`),

CONSTRAINT `answers\_respondent\_id\_foreign` FOREIGN KEY (`respondent\_id`) REFERENCES `respondents` (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=343 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan :



Gambar . Database answer

### Table migrations

Identifikasi/Nama : migrations

Deskripsi Isi : id, migration, batch

Jenis : tabel data transaksi

Primary Key : id

Detail Description : -

Table . Tabel migrations

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | berisikan kode unik dari setiap migrations. | bigint (20) | no | PK |
| migration | atribut migration mengacu pada sebuah file script yang berisi instruksi atau perintah sql untuk membuat atau memodifikasi tabel atau struktur database lainnya. | varchar (255) | no |  |
| batch | atribut yang digunakan dalam konteks migration untuk menunjukkan urutan eksekusi dari setiap migrasi atau perubahan skema database. setiap migrasi dalam batch dieksekusi secara berurutan, dimulai dari batch pertama hingga batch terakhir, sehingga memastikan bahwa perubahan skema database dilakukan secara terurut dan terorganisir. | int (11) | no |  |

SQL script:

CREATE TABLE `migrations` (

`id` int(10) unsigned NOT NULL AUTO\_INCREMENT,

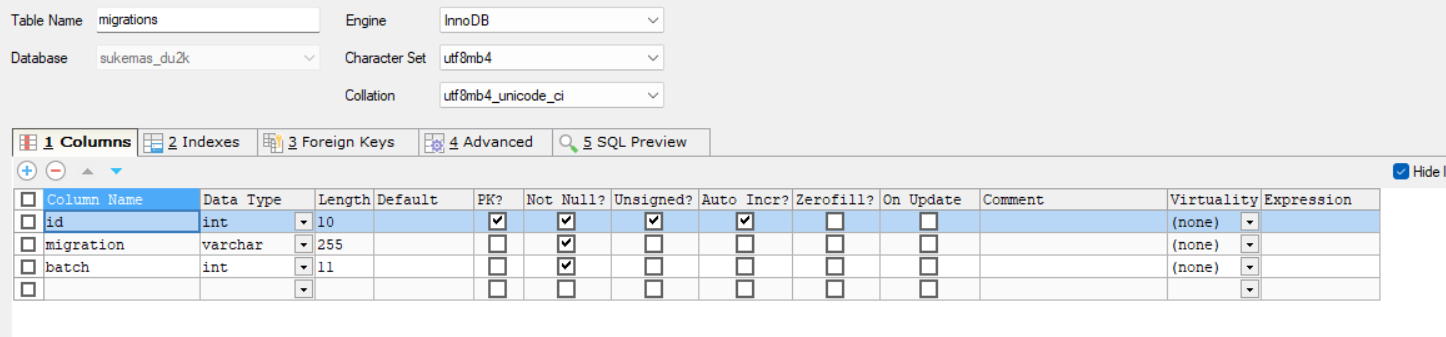
`migration` varchar(255) NOT NULL,

`batch` int(11) NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=21 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan :



Gambar . Database migrations

### Table personal\_acces\_tokens

Identifikasi/Nama : personal\_acces\_tokens.

Deskripsi Isi : id, tokenable\_type, tokenable\_id, name, token, abilities, last\_used\_at, created\_at, updated\_at, deleted\_at.

Jenis : tabel data transaksi

Primary Key : id

Detail Description : berelasi dengan tokenable\_id.

Table . Tabel personal\_access\_token

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Id Field** | **Deskripsi** | **Tipe & length** | **Nullabel** | **Key** |
| id | berisikan kode unik untuk setiap personal acces tokens. | bigint (20) | no | PK |
| tokenable\_id | atribut yang digunakan dalam konteks autentikasi atau otorisasi untuk menunjukkan identitas unik dari entitas atau objek yang dapat menggunakan token untuk mengakses tenant. | bigint (20) | no | FK |
| tokenable\_type | atribut yang digunakan dalam konteks autentikasi atau otorisasi untuk menunjukkan jenis entitas atau objek yang dapat menggunakan token untuk mengakses tenant. | varchar (255) | no |  |
| name | atribut yang digunakan untuk memberikan nama atau label yang bermakna pada token tertentu. nama token ini biasanya diberikan agar mudah dikenali atau dimengerti oleh pengguna atau pihak yang memanfaatkannya. | varchar (255) | no |  |
| token | atribut yang digunakan sebagai kunci atau kode unik yang digunakan dalam proses autentikasi atau otorisasi untuk mengidentifikasi dan mengakses tenant. | varchar (255) | no |  |
| abilities | atribut yang digunakan dalam konteks autentikasi atau otorisasi untuk menunjukkan kemampuan atau hak akses yang diberikan pada token tertentu. | text | no |  |
| last\_used\_at | atribut yang menunjukkan waktu atau tanggal terakhir kali token digunakan atau digunakan untuk mengakses sumber daya tertentu. atribut ini digunakan untuk melacak aktivitas penggunaan token dan juga untuk menghapus token yang tidak aktif atau tidak digunakan dalam jangka waktu tertentu, guna menjaga keamanan dan privasi sumber daya yang diakses. | timestamp | yes |  |
| created\_at | tanggal dan waktu record di insert | timestamp | yes |  |
| updated\_at | tanggal dan waktu record diperbarui atau diubah. | timestamp | yes |  |
| deleted\_at | tanggal dan waktu record dihapus. | timestamp | yes |  |

SQL script:

CREATE TABLE `personal\_access\_tokens` (

`id` bigint(20) unsigned NOT NULL AUTO\_INCREMENT,

`tokenable\_type` varchar(255) NOT NULL,

`tokenable\_id` bigint(20) unsigned NOT NULL,

`name` varchar(255) NOT NULL,

`token` varchar(64) NOT NULL,

`abilities` text DEFAULT NULL,

`last\_used\_at` timestamp NULL DEFAULT NULL,

`expires\_at` timestamp NULL DEFAULT NULL,

`created\_at` timestamp NULL DEFAULT NULL,

`updated\_at` timestamp NULL DEFAULT NULL,s

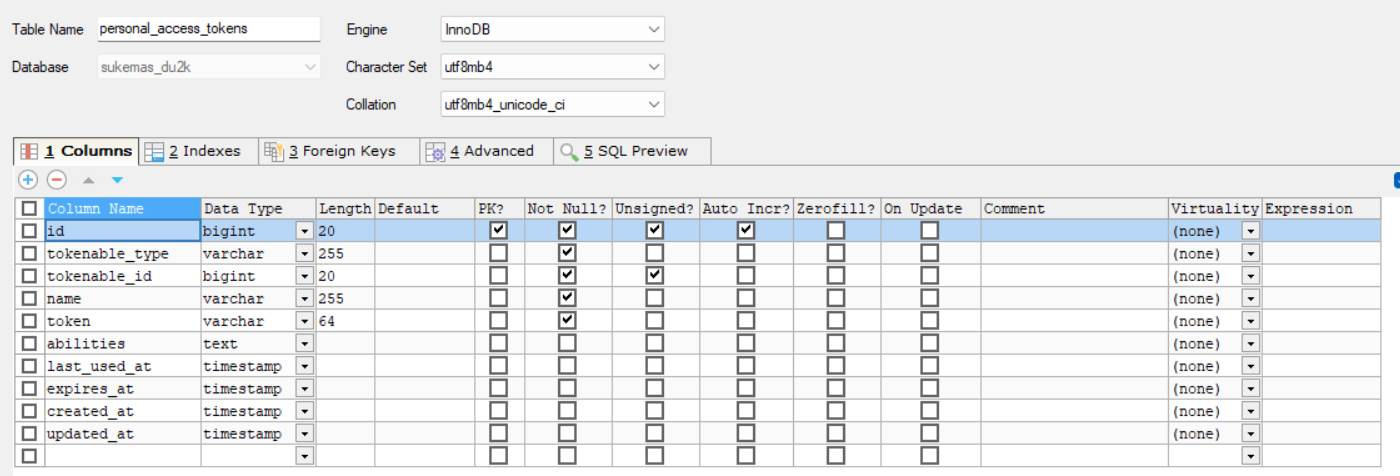
PRIMARY KEY (`id`),

UNIQUE KEY `personal\_access\_tokens\_token\_unique` (`token`),

KEY `personal\_access\_tokens\_tokenable\_type\_tokenable\_id\_index` (`tokenable\_type`,`tokenable\_id`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_unicode\_ci;

Struktur tabel yang dihasilkan:



Gambar . Database personal\_access\_token

## Detail Functional Description

Bab ini akan menjelaskan setiap fungsi yang terdapat pada Sistem Informasi Survey Kepuasan Masyarakat.

### Function Specification Melakukan Registrasi untuk Instansi dan Operator

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi Melakukan Registrasi untuk Instansi dan Operator. |
| Deskripsi Isi | : | Fungsi ini digunakan untuk melakukan pendaftaran terhadap instansi dan akun operator yang akan mengelola instansi tersebut. Setiap orang yang ingin mendaftarkan instansinya ke dalam Sistem Informasi Kepuasan Masyarakat harus melakukan Registrasi terlebih dahulu. |
| Jenis | : | Form Register. |

#### Detail Input Specification

Tidak ada tabel yang digunakan sebagai inputan.

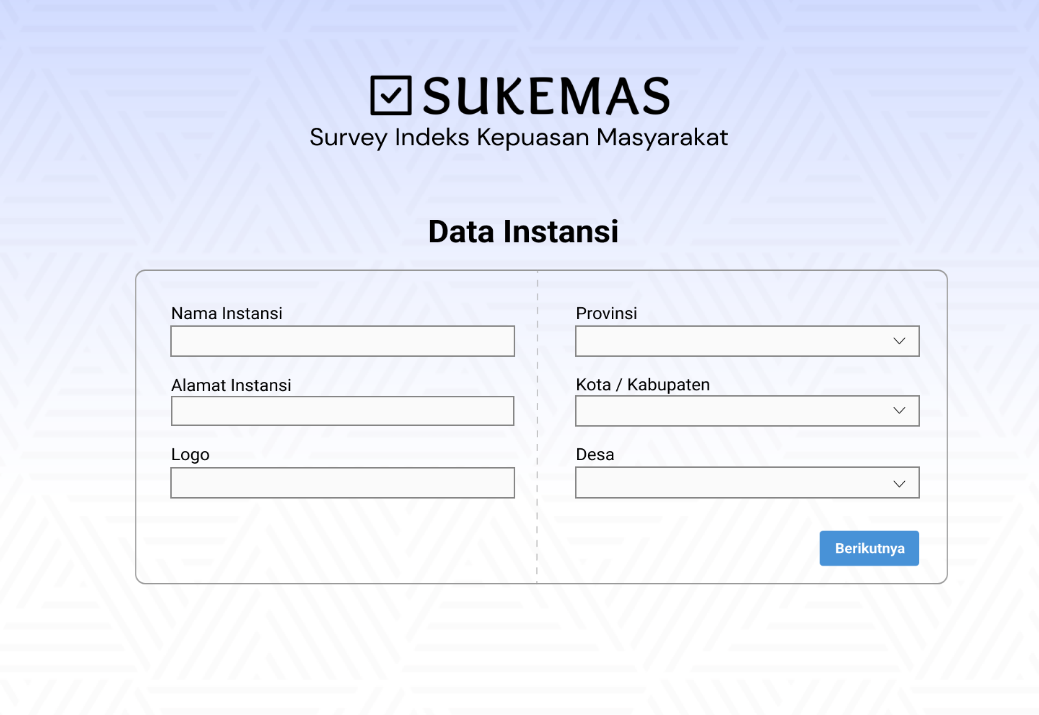
#### Detail Output Specification

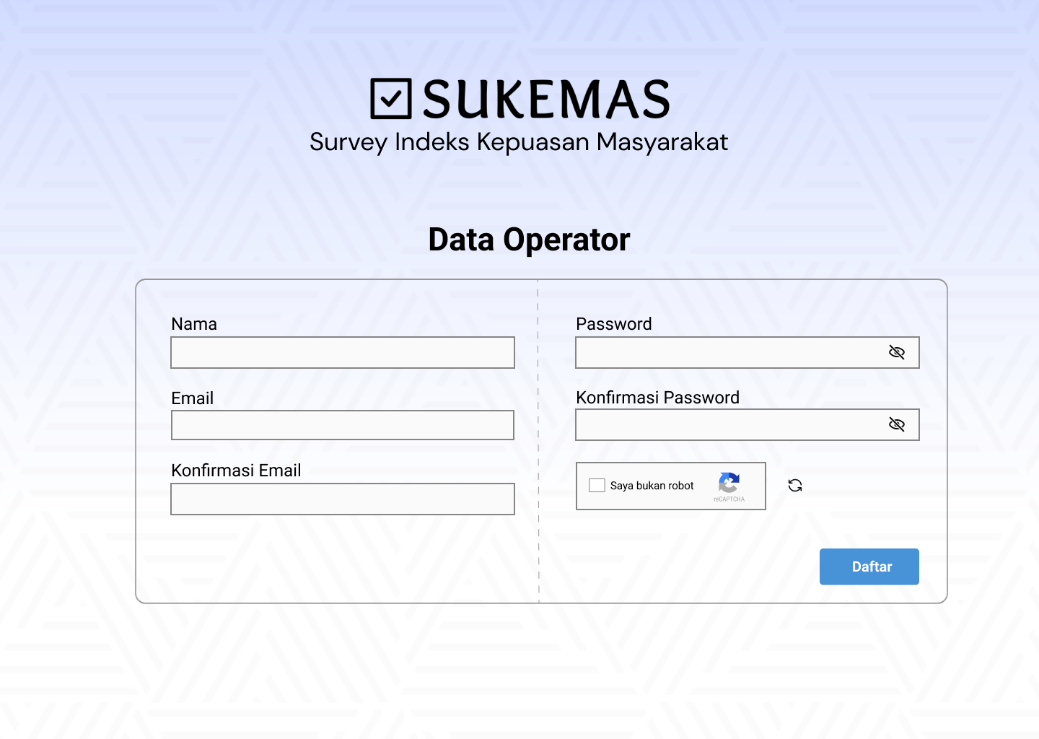
Nama Tabel : 1. tenants

2. users

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* Fungsi Melakukan Registrasi untuk Instansi dan Operator yang akan diisi oleh masyarakat yang hendak mendaftarkan isntansi dan akunnya. Tampilan dapat dilihat pada Gambar 24.





Gambar . Detail Layout Registrasi untuk Instansi dan Operator

#### URL

/daftar.

#### Fields description

Deskripsi *field* pada Fungsi Melakukan Registrasi untuk Instansi dan Operator dapat dilihat pada Table 44.

Table . Field Registrasi untuk Instansi dan Operator

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Label** | **Field** | **Tabel** | **I/O** | **Format** | **Validasi** |
| Nama Instansi | tenants.name | tenants | I | varchar(255) | Harus diisi |
| Alamat Instansi | tenants.address | tenants | I | varchar(255) | Harus diisi |
| Logo | tenants.logo\_path | tenants | I | varchar(255) | Opsional |
| Provinsi | tenant.province\_id | tenants | I | bigint(20) | Harus diisi |
| Kota/Kabupaten | tenants.regency\_id | tenants | I | bigint(20) | Harus diisi |
| Desa | tenants.district\_id | tenants | I | bigint(20) | Harus diisi |
| Nama | users.name | users | I | varchar(255) | Harus diisi |
| Email | users.email | users | I | varchar(255) | Harus diisi |
| Konfirmasi Email | users.email | users | I | varchar(255) | Harus diisi |
| Password | users.password | users | I | varchar(255) | Harus diisi |
| Konfirmasi Password | users. password | users | I | varchar(255) | Harus diisi |

#### Objects Specification

Spesifikasi objek dari Fungsi Melakukan Registrasi untuk Instansi dan Operator dapat dilihat pada Table 45.

Table . Objects Specification

|  |  |  |
| --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Keterangan** |
| Next | Button | Jika diklik akan menampilkan data operator yang harus diisi. |
| Back | Button | Jika diklik akan menampilkan data instansi yang harus diisi dan ingin diedit. |
| Submit | Button | Jika diklik maka data instansi dan operator yang sudah dimasukkan akan disimpan dan akun selesai dibuat. |

#### Error Message

Semua *error message* yang muncul pada Fungsi Melakukan Registrasi untuk Instansi dan Operator dapat dilihat pada Table 46

*Table 46. Error Message pada Registrasi untuk Instansi dan Operator*

|  |  |  |
| --- | --- | --- |
| **No.** | **Keterangan** | **Pesan Error yang disampaikan** |
| 1. | Field Tenant name tidak diisi. | The tenant name field is required. |
| 2. | Field Tenant address tidak diisi. | The tenant name field is required. |
| 3. | Field User name tidak diisi. | The tenant name field is required. |
| 4. | Field User email tidak diisi. | The tenant name field is required. |

#### Algorithm

Algoritma yang terdapat pada Fungsi Melakukan Registrasi untuk Instansi dan Operator terdapat pada tabel berikut ini.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id\_Proses | | : | SDD\_Process\_001\_ Melakukan Registrasi untuk Instansi dan Operator Register | |
| Objek terkait | | : | Register | |
| Event | | : | Click | |
| **Initial State (IS):**  Tabel tenants sudah terdapat pada database dan dapat diinput.  Tabel users sudah terdapat pada database dan dapat diinput. | | |
| **Final State (FS):**  Instansi berhasil terdaftar pada Sistem Informasi Survey Kepuasan Masyarakat dan akun operator yang akan mengelola instansi tersebut berhasil dibuat. | | |
| **Spesifikasi Proses/Algoritma:**   1. Mengisi semua data yang diminta oleh sistem. 2. IF email yang dimasukkan sudah terdapat pada database, masukkan email lain   ELSE  email belum terdapat pada database, akun operator dapat dibuat.   1. Disimpan di database. | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification Melakukan Login

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi Melakukan Login. |
| Deskripsi Isi | : | Fungsi ini digunakan ketika operator hendak masuk dan mengelola Sistem Informasi Survey Kepuasan Masyarakat. |
| Jenis | : | Form *Login*. |

#### Detail Input Specification

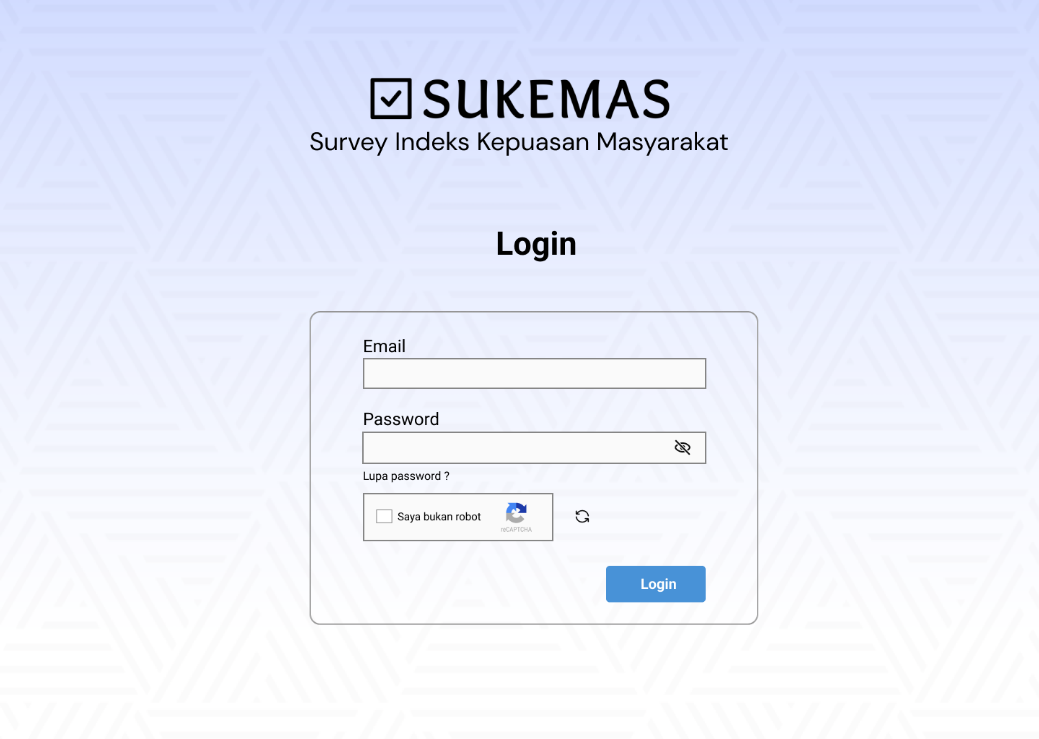
Tidak ada tabel yang digunakan sebagai inputan.

#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | users |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* Login yang akan diisi oleh operator yang hendak masuk ke dalam Sistem Informasi Survey Kepuasan Masyarakat. Tampilan dapat dilihat pada Gambar 25.



Gambar . Detail Layout Login

#### URL

/login.

#### Fields description

Deskripsi *field* pada Fungsi Login dapat dilihat pada Table 47.

*Table 47. Field pada Login*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Label** | **Field** | **Tabel** | **I/O** | **Format** | **Validasi** |
| Email | email | users | I | varchar(255) | Harus diisi |
| Password | password | users | I | varchar(255) | Harus diisi |

#### Objects Specification

Spesifikasi objek dari Fungsi Login dapat dilihat pada Table 48.

*Table 48. Object pada Login*

|  |  |  |
| --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Keterangan** |
| Login | Button | Jika diklik sistem akan memastikan apakah Email address dan Password yang dimasukkan sudah benar atau belum. Jika sudah benar, maka operator berhasil masuk ke dalam sistem. Jika belum, maka operator harus kembali memasukkan Email address dan Password sampai benar. |

#### Error Message

Semua *error message* yang muncul pada Fungsi Login dapat dilihat pada Table 49.

*Table 49. Error Message pada Login*

|  |  |  |
| --- | --- | --- |
| **No.** | **Keterangan** | **Pesan Error yang disampaikan** |
| 1. | Field Email address tidak diisi. | Please fill out this field. |
| 2. | Field Password tidak diisi. | Please fill out this field. |
| 3. | Field Email address tidak diisi sesuai aturan. | Please include an '@' in the email address. 'xxx' is missing an '@'. |
| 4. | Field Email address diisi namun tidak terdaftar pada database. | These credentials do not match our records. |
| 5. | Field Password diisi tetapi salah. | These credentials do not match our records. |

#### Algorithm

Algoritma yang terdapat pada Fungsi Login terdapat pada tabel berikut ini.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id\_Proses | | : | SDD\_Process\_002\_Login | |
| Objek terkait | | : | Login | |
| Event | | : | Click | |
| **Initial State (IS):**  Instansi dan akun Operator sudah berhasil dibuat. | | |
| **Final State (FS):**  Operator berhasil masuk ke dalam Sistem Informasi Survey Kepuasan Masyarakat. | | |
| **Spesifikasi Proses/Algoritma:**   1. Mengisi semua data yang diminta oleh sistem. 2. IF Email address yang dimasukkan salah, masukkan Email address yang benar   ELSE  Email address dapat digunakan.   1. IF Password yang dimasukkan salah, masukkan Password yang benar   ELSE  Password dapat digunakan.   1. IF Email address dan Password yang dimasukkan salah, masukkan Email address dan Password yang benar   ELSE  Operator berhasil masuk ke dalam Sistem Informasi Survey Kepuasan Masyarakat.   1. Disimpan di database. | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification List Operator

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi List Operator |
| Deskripsi Isi | : | Fungsi ini digunakan untuk melihat siapa saja operator yang berada dalam sebuah tenant (instansi). |
| Jenis | : | Form User. |

#### Detail Input Specification List Operator

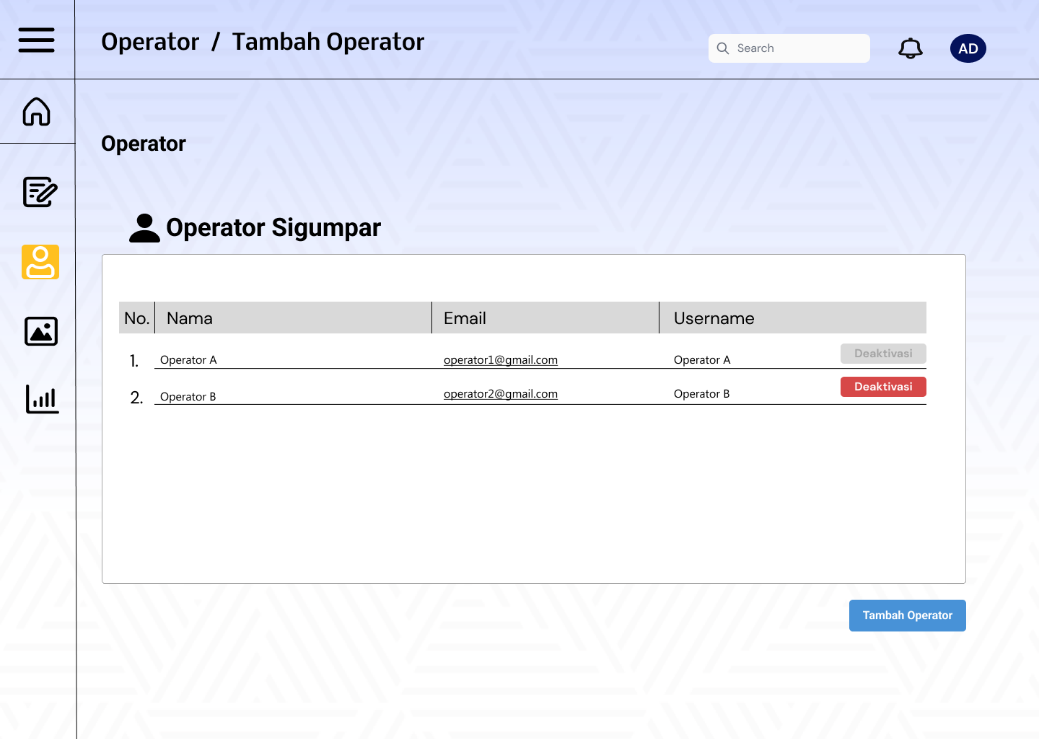
Tidak ada tabel yang digunakan sebagai inputan.

#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | users |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* List Operator yang akan menampilkan list dari operator-operator yang terdapat dalam sebuah instansi yang sama. Tampilan dapat dilihat pada Gambar 26.



Gambar . Detail Layout pada List Operator

#### URL

/listOperator.

#### Fields description

Deskripsi *field* pada Fungsi List Operator dapat dilihat pada Table 50.

Table . Field pada List Operator

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Label** | **Field** | **Tabel** | **I/O** | **Format** | **Validasi** |
| Name | name | users | I | varchar(255) | Harus diisi |
| Email | email | users | I | varchar(255) | Harus diisi |
| Username | username | users | I | varchar(255) | Harus diisi |

#### Object Specifitation

Spesifikasi objek dari Fungsi List Operator dapat dilihat pada Table 51.

*Table 51. Object pada List Operator*

|  |  |  |
| --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Keterangan** |
| Deaktivasi | Button | Jika diklik, operator akan dialihkan ke fungsi mendeaktivasi operator. |
| Tambah Operator | Button | Jika diklik, operator akan dialihkan ke fungsi menambah operator baru. |

#### Error Message

Tidak terdapat *error message* pada fungsi ini.

#### Algorithm

Algoritma yang terdapat pada Fungsi List Operator terdapat pada tabel berikut ini.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id\_Proses | | : | SDD\_Process\_003\_List Operator | |
| Objek terkait | | : | User | |
| Event | | : | Click | |
| **Initial State (IS):**  Operator sudah login ke dalam Sistem Informasi Survey Kepuasan Masyarakat. | | |
| **Final State (FS):**  Operator berhasil melihat operator-operator yang terdaftar di dalam Sistem Informasi Survey Kepuasan Masyarakat. | | |
| **Spesifikasi Proses/Algoritma:**  Operator masuk ke dalam Sistem Informasi Survey Kepuasan Masyarakat.  Disimpan di database. | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification Menambah Operator Baru

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi Menambah Operator Baru. |
| Deskripsi Isi | : | Fungsi ini digunakan oleh Operator yang sudah login untuk menambahkan operator lain ke dalam Sistem Informasi Kepuasan Masyarakat yang berada pada instansi yang sama. |
| Jenis | : | Form User. |

#### Detail Input Specification

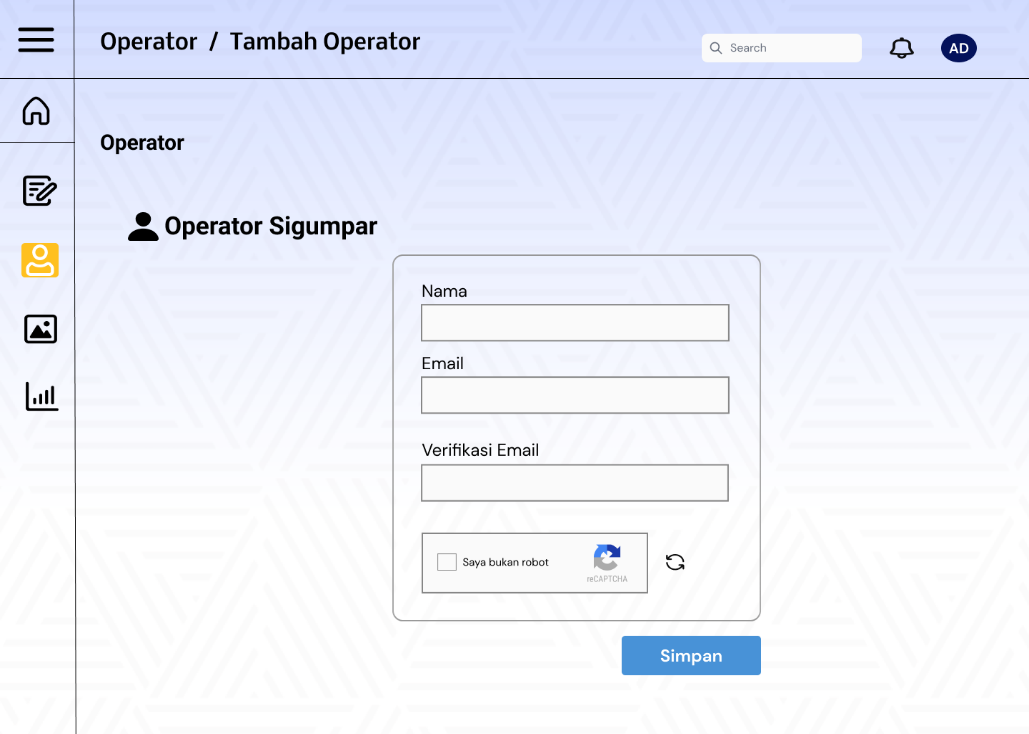
Tidak ada tabel yang digunakan sebagai inputan.

#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | users |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* Menambah Operator Baru yang akan dilakukan oleh Operator yang sudah login untuk menambah operator lain untuk mengelola instansi mereka. Tampilan dapat dilihat pada Gambar 27.



Gambar . Detail Layout pada Menambah Operator

#### URL

/tambahOperator.

#### Fields description

Deskripsi *field* pada Fungsi Menambah Operator Baru dapat dilihat pada Table 52.

*Table 52. Field pada Menambah Operator Baru*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Label** | **Field** | **Tabel** | **I/O** | **Format** | **Validasi** |
| Nama | name | users | I | varchar(255) | Harus diisi |
| Email | email | users | I | varchar(255) | Harus diisi |
| Verifikasi Email | email | users | I | varchar(255) | Harus diisi |

#### Objects Specification

Spesifikasi objek dari Fungsi Menambah Operator Baru dapat dilihat pada Table 53.

*Table 53. Object pada Menambah Operator Baru*

|  |  |  |
| --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Keterangan** |
| Tambah Operator | Button | Jika diklik akan membuat operator baru. |
| Simpan | Button | Jika diklik akan menyimpan data operator baru yang ingin dibuat dan operator baru berhasil dibuat. |

#### Error Message

Semua *error message* yang muncul pada Fungsi Menambah Operator Baru dapat dilihat pada Table 54.

*Table 54. Error Message pada Menambah Operator Baru*

|  |  |  |
| --- | --- | --- |
| **No.** | **Keterangan** | **Pesan Error yang disampaikan** |
| 1. | Field Email tidak diisi. | The email field is required. |
| 2. | Field Password tidak diisi. | Please fill out this field. |
| 3. | Field Email diisi tidak sesuai dengan aturan. | The email field must be a valid email address. |
| 4. | Field Email diisi dengan Email yang sudah pernah dibuat sebelumnya. | The email has already been taken. |
| 5. | Field Verifikasi Email tidak sama dengan Email yang sudah dimasukkan sebelumnya. | Email verification field is not the same as email field. |

#### Algorithm

Algoritma yang terdapat pada Fungsi Menambah Operator Baru terdapat pada tabel berikut ini.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id\_Proses | | : | SDD\_Process\_004\_ Menambah Operator Baru | |
| Objek terkait | | : | User | |
| Event | | : | Click | |
| **Initial State (IS):**  Operator yang sudah login sudah login terlebih dahulu ke dalam Sistem Informasi Survey Kepuasan Masyarakat. | | |
| **Final State (FS):**  Operator yang sudah login berhasil menambahkan operator lain untuk membantu mengelola instansinya. | | |
| **Spesifikasi Proses/Algoritma:**   1. Mengisi semua data yang diminta oleh sistem. 2. IF email yang dimasukkan sudah terdapat pada database, masukkan email lain   ELSE  email belum terdapat pada database, akun operator dapat ditambahkan.   1. Disimpan di database. | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification Penonaktifan (Diaktivasi) Akun Operator

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi Penonaktifan (Diaktivasi) Akun Operator. |
| Deskripsi Isi | : | Fungsi ini digunakan oleh Operator yang sudah login ke dalam Sistem Informasi Survey Kepuasan Masyarakat untuk mendeaktivasi akun operator lain yang berada pada instansi mereka. |
| Jenis | : | Form User. |

#### Detail Input Specification

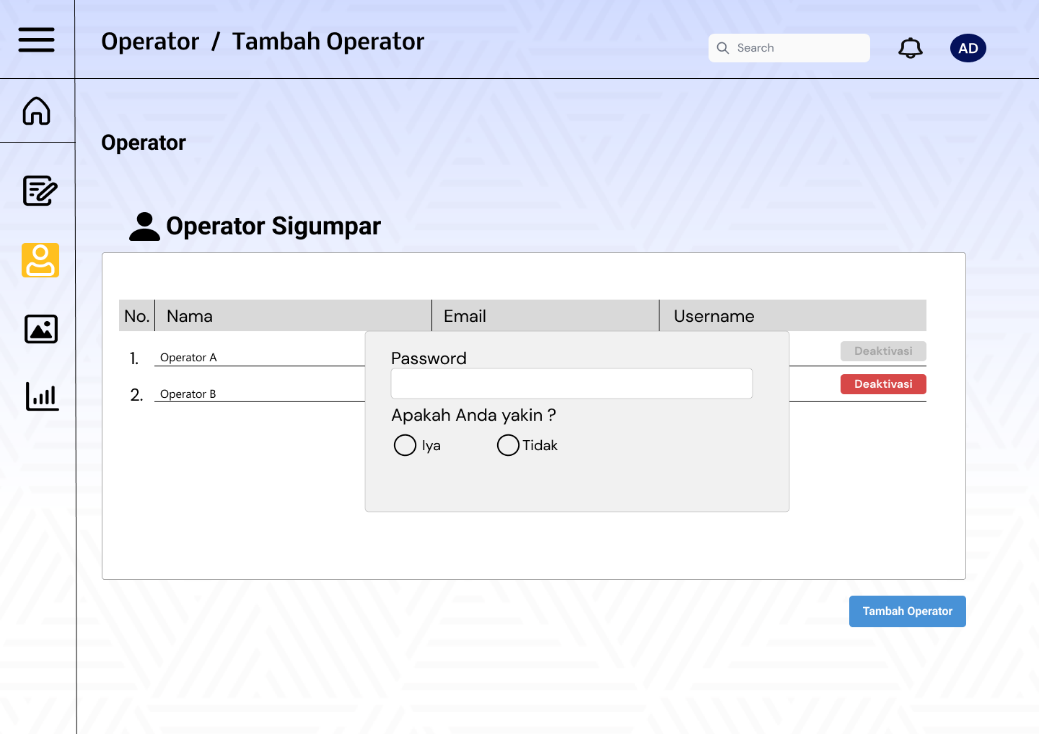
|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | users |

#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | users |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* Penonaktifan (Diaktivasi) Akun Operator yang akan digunakan oleh operator yang sudah login terhadap operator lain yang berada pada instansinya. Tampilan dapat dilihat pada Gambar 28.



Gambar . Detail Layout pada Penonaktifan (Diaktivasi) Akun Operator

#### URL

/ diaktivasiAkun.

#### Fields description

Deskripsi *field* pada Fungsi Penonaktifan (Diaktivasi) Akun Operator dapat dilihat pada Table 55.

*Table 55. Field pada Penonaktifan (Diaktivasi) Akun Operator*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Label** | **Field** | **Tabel** | **I/O** | **Format** | **Validasi** |
| Password | password | users | I | varchar(255) | Harus diisi |

#### Objects Specification

Spesifikasi objek dari Fungsi Penonaktifan (Diaktivasi) Akun Operator dapat dilihat pada Table 56.

*Table 56. Object pada Penonaktifan (Diaktivasi) Akun Operator*

|  |  |  |
| --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Keterangan** |
| Password | Text box | Tempat untuk memasukkan password. |
| Ya | Radio | Jika diklik maka proses deaktivasi berhasil dilakukan. |
| Tidak | Radio | Jika diklik maka proses deaktivasi dibatalkan. |

#### Error Message

Semua *error message* yang muncul pada Fungsi Penonaktifan (Diaktivasi) Akun Operator dapat dilihat pada Table 57.

*Table 57. Error Message pada Penonaktifan (Diaktivasi) Akun Operator*

|  |  |  |
| --- | --- | --- |
| **No.** | **Keterangan** | **Pesan Error yang disampaikan** |
| 1. | Field Password diisi dengan password yang salah. | Password does not match. |
| 2. | Field Password tidak diisi. | Please fill out this field. |

#### Algorithm

Algoritma yang terdapat pada Fungsi Penonaktifan (Diaktivasi) Akun Operator terdapat pada tabel berikut ini.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id\_Proses | | : | SDD\_Process\_005\_ Penonaktifan (Diaktivasi) Akun Operator | |
| Objek terkait | | : | User | |
| Event | | : | Click | |
| **Initial State (IS):**  Sudah terdapat operator yang ingin dideaktivasi. | | |
| **Final State (FS):**  Proses deaktivasi operator lain berhasil dilakukan oleh operator yang sedang login. | | |
| **Spesifikasi Proses/Algoritma:**   1. Mengisi password yang diminta oleh sistem. 2. IF password yang dimasukkan salah, masukkan password lain sampai benar   ELSE  Password benar, proses deaktivasi berhasil.   1. Disimpan di database. | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification Update Profile

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi Update Profile. |
| Deskripsi Isi | : | Fungsi ini digunakan oleh Operator yang sudah login ke dalam Sistem Informasi Survey Kepuasan Masyarakat untuk mengedit dan mengupdate data diri mereka. |
| Jenis | : | Form User. |

#### Detail Input Specification

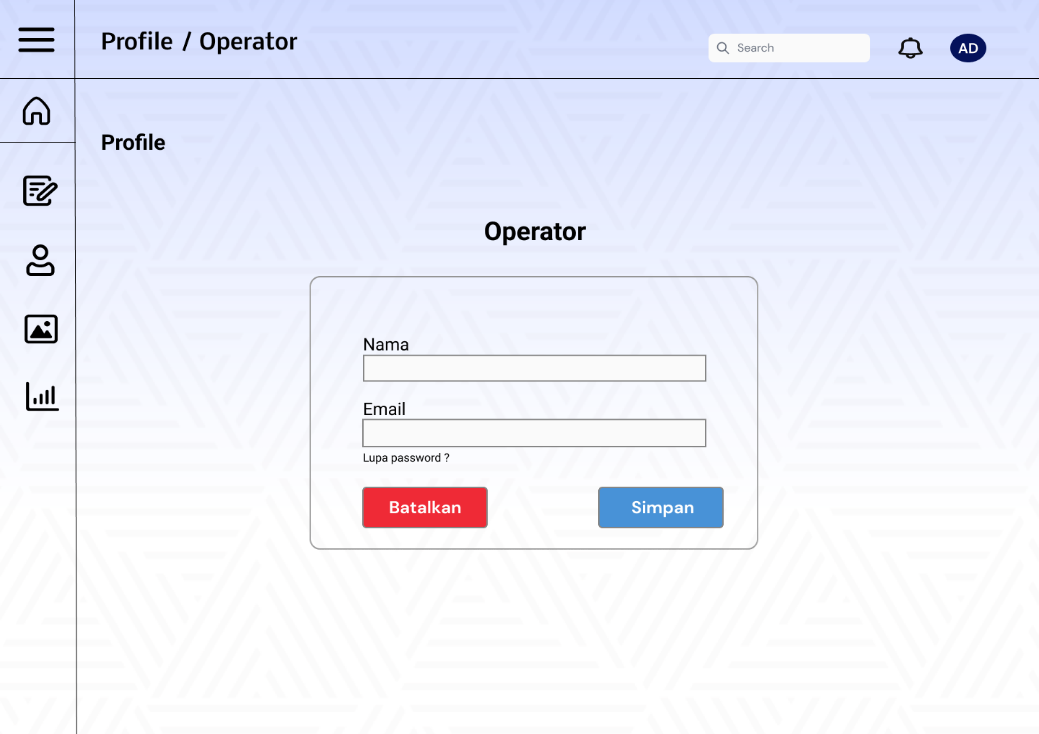
|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | users |

#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | users |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* Update Profile User yang akan digunakan oleh operator yang sudah login untuk mengupdate data diri mereka. Tampilan dapat dilihat pada Gambar 29.



Gambar . Detail Layout pada Update Profile

#### URL

/updateProfile.

#### Fields description

Deskripsi *field* pada Fungsi Update Profile dapat dilihat pada Table 58.

*Table 58. Field pada Update Profile*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Label** | **Field** | **Tabel** | **I/O** | **Format** | **Validasi** |
| Nama | name | users | I | varchar(255) | Harus diisi |
| Email | email | users | I | varchar(255) | Harus diisi |

#### Objects Specification

Spesifikasi objek dari Fungsi Update Profile dapat dilihat pada Table 59.

*Table 59. Object pada Update Profile*

|  |  |  |
| --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Keterangan** |
| Simpan | Button | Jika diklik maka proses update profile dari operator yang sudah login ke dalam Sistem Informasi Survey Kepuasan Masyarakat berhasil dilakukan. |
| Batalkan | Button | Jika diklik maka proses update profile dibatalkan. |

#### Error Message

Semua *error message* yang muncul pada Fungsi Update Profile dapat dilihat pada Table 60.

Table . Error Message pada Update Profile

|  |  |  |
| --- | --- | --- |
| No. | Keterangan | Pesan Error yang disampaikan |
| 1. | Field Email tidak diisi. | The email field is required. |
| 2. | Field Name tidak diisi. | Please fill out this field. |
| 3. | Field Email diisi tidak sesuai dengan aturan. | The email field must be a valid email address. |
| 4. | Field Email diisi dengan Email yang sudah pernah dibuat sebelumnya. | The email has already been taken. |

#### Algorithm

Algoritma yang terdapat pada Fungsi Update Profile terdapat pada tabel berikut ini.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id\_Proses | | : | SDD\_Process\_006\_ Update Profile | |
| Objek terkait | | : | User | |
| Event | | : | Click | |
| **Initial State (IS):**  Sudah terdapat operator yang ingin diupdate data dirinya. | | |
| **Final State (FS):**  Proses update data diri operator berhasil dilakukan. | | |
| **Spesifikasi Proses/Algoritma:**   1. Mengubah data diri sesuai dengan yang diinginkan. 2. Disimpan di database. | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification Update Tenant

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi Update Tenant. |
| Deskripsi Isi | : | Fungsi ini digunakan oleh Operator yang sudah login ke dalam Sistem Informasi Survey Kepuasan Masyarakat untuk mengedit dan mengupdate data instansi mereka. |
| Jenis | : | Form Tenant. |

#### Detail Input Specification

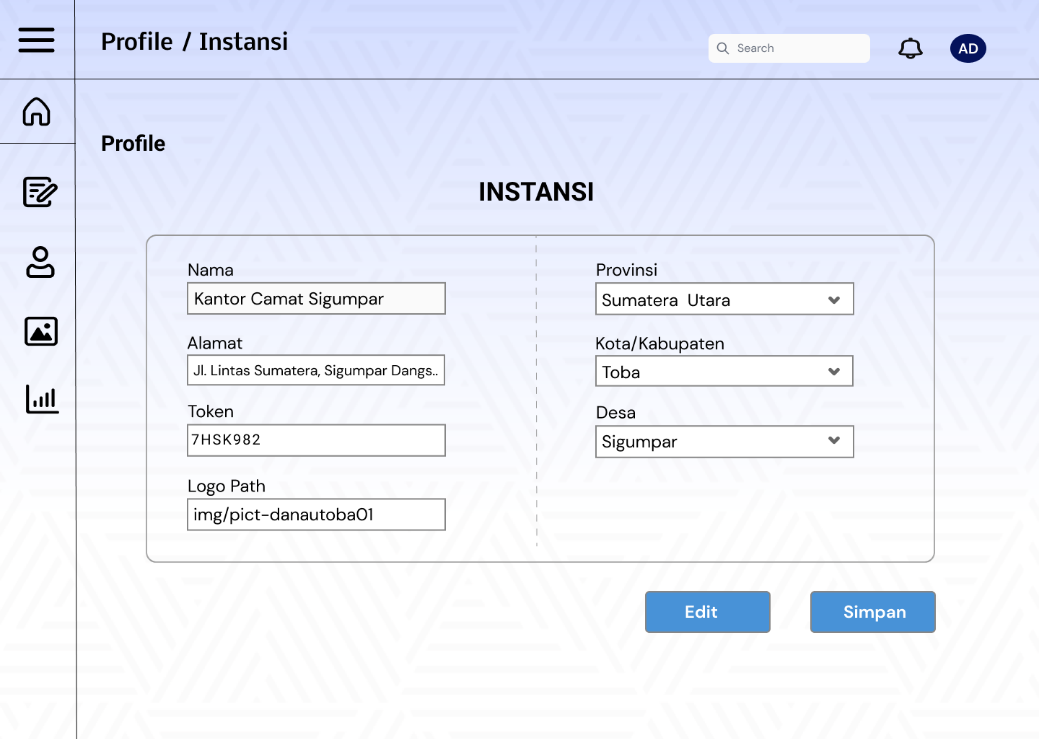
|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | tenants |

#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | tenants |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* Update Tenant yang akan digunakan oleh operator yang sudah login untuk mengupdate data dari instansi mereka. Tampilan dapat dilihat pada Gambar 30.



Gambar . Detail Layout pada Update Tenant

#### URL

/updateTenant.

#### Fields description

Deskripsi *field* pada Fungsi Update Tenant dapat dilihat pada Table 61.

Table . Field pada Update Tenant

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Label** | **Field** | **Tabel** | **I/O** | **Format** | **Validasi** |
| Nama | name | tenants | I | varchar(255) | Harus diisi |
| Alamat | address | tenants | I | varchar(255) | Harus diisi |
| Token | token | tenants | I | varchar(255) | Opsional |
| Logo Path | logo\_path | tenants | I | varchar(255) | Opsional |
| Provinsi | province\_id | tenants | I | bigint(20) | Harus diisi |
| Kota/Kabupaten | regency\_id | tenants | I | bigint(20) | Harus diisi |
| Desa | district\_id | tenants | I | bigint(20) | Harus diisi |

#### Objects Specification

Spesifikasi objek dari Fungsi Update Tenant dapat dilihat pada Table 62.

*Table 62. Object pada Update Tenant*

|  |  |  |
| --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Keterangan** |
| Simpan | Button | Jika diklik maka proses update tenant berhasil dilakukan. |
| Edit | Button | Jika diklik maka proses untuk mengedit Tenant dimulai. |

#### Error Message

Semua *error message* yang muncul pada Fungsi Update Profile dapat dilihat pada Table 63.

Table . Error Message pada Update Tenant

|  |  |  |
| --- | --- | --- |
| **No.** | **Keterangan** | **Pesan Error yang disampaikan** |
| 1. | Field Nama tidak diisi. | Please fill out this field. |
| 2. | Field Alamat tidak diisi. | Please fill out this field. |
| 3. | Field Provinsi tidak diisi. | Please fill out this field. |
| 4. | Field Kabupaten tidak diisi. | Please fill out this field. |
| 5. | Field Kecamatan tidak diisi. | Please fill out this field. |
| 6. | Field Logo Path diisi tidak sesuai type yang diminta. | File of invalid type. |
| 7. | Field Logo Path diisi tidak sesuai size yang diminta. | File is too small atau File is too large. |

#### Algorithm

Algoritma yang terdapat pada Fungsi Update Tenant terdapat pada tabel berikut ini.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id\_Proses | | : | SDD\_Process\_007\_ Update Tenant | |
| Objek terkait | | : | User | |
| Event | | : | Click | |
| **Initial State (IS):**  Sudah terdapat instansi yang ingin diupdate datanya. | | |
| **Final State (FS):**  Proses update data instansi berhasil dilakukan oleh operator yang berada pada instansi tersebut. | | |
| **Spesifikasi Proses/Algoritma:**   1. Mengedit data instansi yang diinginkan oleh operator yang berada pada instansi tersebut. 2. IF logo yang dimasukkan tidak sesuai size, masukkan size yang sesuai   ELSE  logo sudah sesuai size yang diminta, logo dapat digunakan.   1. Disimpan di database. | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification Mengubah Kata Sandi

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi Mengubah Kata Sandi. |
| Deskripsi Isi | : | Fungsi ini digunakan oleh Operator yang sudah login ke dalam Sistem Informasi Survey Kepuasan Masyarakat dan ingin mengganti kata sandi mereka. |
| Jenis | : | Form User. |

#### Detail Input Specification

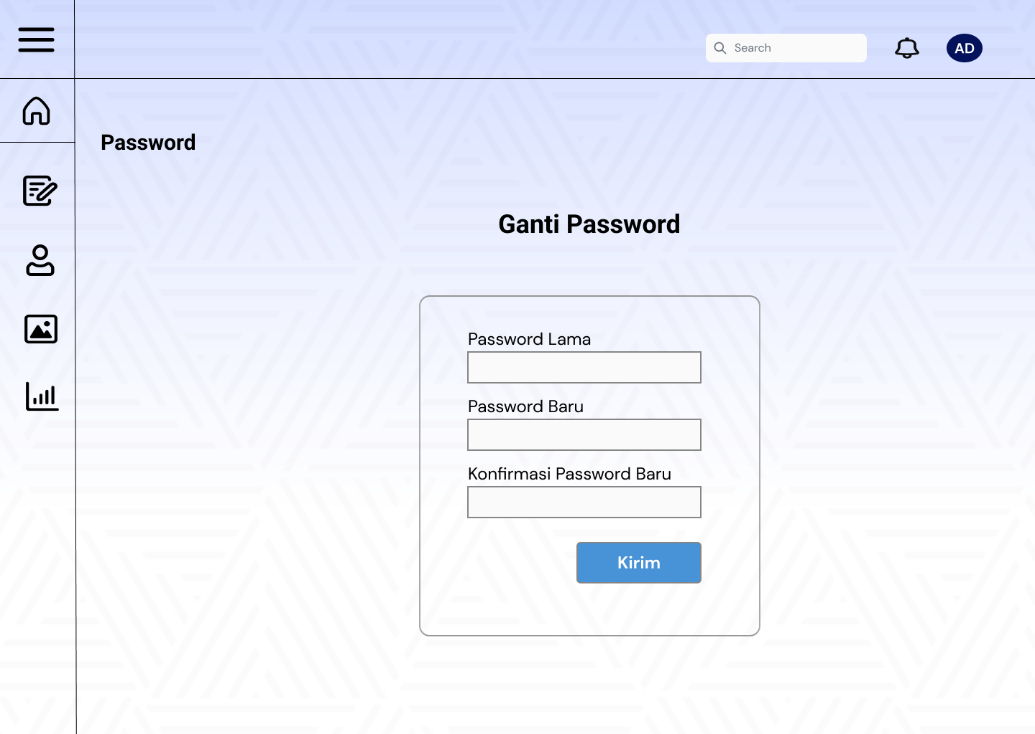
|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | users |

#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | users |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* Mengubah Kata Sandi yang akan digunakan oleh operator. Tampilan dapat dilihat pada Gambar 31.



Gambar . Detail Layout pada Mengubah Kata Sandi

#### URL

/ubahKataSandi.

#### Fields description

Deskripsi *field* pada Fungsi Mengubah Kata Sandi dapat dilihat pada Table 64.

*Table 64. Field pada Mengubah Kata Sandi*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Label** | **Field** | **Tabel** | **I/O** | **Format** | **Validasi** |
| Password Lama | password | users | I | varchar(255) | Harus diisi |
| Password Baru | password | users | I | varchar(255) | Harus diisi |
| Konfirmasi Password Lama | password | users | I | varchar(255) | Harus diisi |

#### Objects Specification

Spesifikasi objek dari Fungsi Mengubah Kata Sandi dapat dilihat pada Table 65.

Table . Object pada Mengubah Kata Sandi

|  |  |  |
| --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Keterangan** |
| Kirim | Button | Jika diklik maka proses mengganti password berhasil dilakukan. |

#### Error Message

Semua *error message* yang muncul pada Fungsi Mengubah Kata Sandi dapat dilihat pada Table 66.

*Table 66. Error Message pada Mengubah Kata Sandi*

|  |  |  |
| --- | --- | --- |
| **No.** | **Keterangan** | **Pesan Error yang disampaikan** |
| 1. | Field Password Lama tidak diisi. | Please fill out this field. |
| 2. | Field Password Baru tidak diisi. | Please fill out this field. |
| 3. | Field Konfirmasi Password Baru tidak diisi. | Please fill out this field. |
| 4. | Field Konfirmasi Password Baru diisi tidak sama dengan Password Baru. | New password confirmation field is not the same as New Password field. |

#### Algorithm

Algoritma yang terdapat pada Fungsi Mengubah Kata Sandi terdapat pada tabel berikut ini.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id\_Proses | | : | SDD\_Process\_008\_ Melakukan Perbaikan karena Lupa Kata Sandi | |
| Objek terkait | | : | User | |
| Event | | : | Click | |
| **Initial State (IS):**  Sudah terdapat instansi yang ingin diupdate datanya. | | |
| **Final State (FS):**  Proses update data instansi berhasil dilakukan oleh operator yang berada pada instansi tersebut. | | |
| **Spesifikasi Proses/Algoritma:**   1. Mengisi password lama, password baru, dan konfirmasi password baru. 2. IF konfirmasi password baru yang dimasukkan tidak sama dengan password baru, masukkan konfirmasi password baru yang sama dengan password baru   ELSE  konfirmasi password baru sama dengan password baru, klik ‘Kirim’.   1. Disimpan di database. | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification Menambah Question Set

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi Menambah Question Set. |
| Deskripsi Isi | : | Fungsi ini digunakan oleh Operator yang sudah login ke dalam Sistem Informasi Survey Kepuasan Masyarakat untuk menambah Question Set pada instansi mereka. |
| Jenis | : | Form Question Set*.* |

#### Detail Input Specification

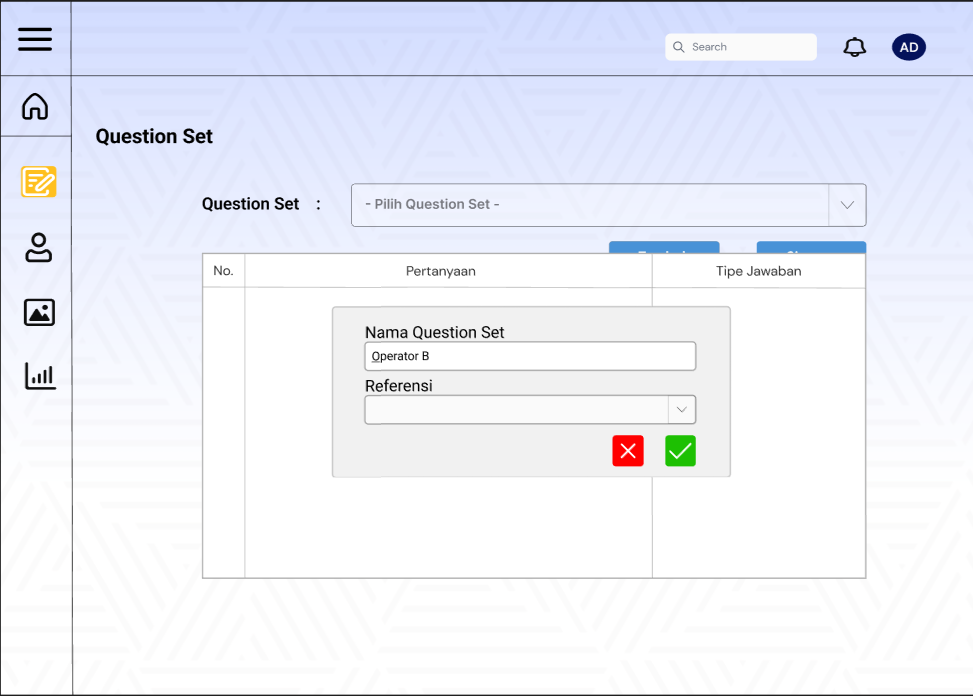
|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | Tidak terdapat table untuk inputan |

#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | Question Set |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* Menambah Question Set yang akan digunakan oleh operator yang sudah login. Tampilan dapat dilihat pada Gambar 32.



Gambar . Detail Layout Menambah Question Set

#### URL

/question-sets/create

#### Fields description

Deskripsi *field* pada Fungsi Menambah Services dapat dilihat pada Table 67.

Table . Field pada Menambah Services

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Label** | **Field** | **Tabel/Query** | **I/O** | **Format** | **Validasi** |
| Nama Question Set | nama | questionSet | I | varchar | Harus diisi |
| Reference | reference | questionSet | I | varchar | Harus diisi |

#### Objects Specification

Spesifikasi objek dari Fungsi Menambah Services dapat dilihat pada Table 68.

Table . Objek pada Menambah Services

|  |  |  |
| --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Keterangan** |
|  | Button | Jika diklik akan membuat Question Set baru. |
| X | Button | Jika diklik maka proses membuat Question Set baru dibatalkan. |

#### Error Message

Semua *error message* yang muncul pada Fungsi Menambah Services dapat dilihat pada Table 69.

Table . Error Message pada Menambah Services

|  |  |  |
| --- | --- | --- |
| No. | Keterangan | Pesan Error yang disampaikan |
| 1. | Semua field tidak diisi oleh user, sementara user memilih tombol Create | Akan muncul error message yang mengatakan “Please fill out this field” |
| 2. | Field Nama Question Set tidak diisi, sementara user memilih tombol Create | Akan muncul error message yang mengatakan “Please fill out this field” |
| 3. | Field Reference tidak diisi, sementara user memilih tombol Create | Akan muncul error message yang mengatakan “Please fill out this field” |

#### Algorithm

Algoritma yang terdapat pada Fungsi Menambah Services terdapat pada tabel berikut ini.

|  |  |  |
| --- | --- | --- |
| Id\_Proses | : | SDD\_Process\_009\_ Menambah Question Set |
| Objek terkait | : | Question Set |
| Event | : | Click |
| **Initial State (IS):**  Operator sudah masuk ke dalam Sistem Informasi Survey Kepuasan Masyarakat. | | |
| **Final State (FS):**  Proses menambahkan Question Set baru berhasil dilakukan oleh operator yang berada pada instansi tersebut. | | |
| **Spesifikasi Proses/Algoritma:**  Operator yang berada pada instansi tersebut mengisi nama Question Set apa yang ingin ditambahkan. | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification List Question Set

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi List Question Set. |
| Deskripsi Isi | : | Fungsi ini digunakan oleh Operator yang sudah login ke dalam Sistem Informasi Survey Kepuasan Masyarakat untuk melihat Question Set apa saja yang terdapata pada instansi mereka. |
| Jenis | : | Form Question Set*.* |

#### Detail Input Specification

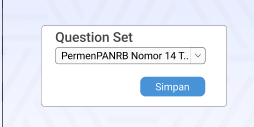
|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | Question Set |

#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | Question Set |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* List Question Set yang akan digunakan oleh operator yang sudah login untuk melihat divisi apa saja yang terdaftar pada instansi mereka. Tampilan dapat dilihat pada Gambar 33.



Gambar . Detail Layout List Question Set

#### URL

/question-sets/list

#### Fields description

Deskripsi *field* pada Fungsi List Question Set dapat dilihat pada Table 70

Table . Field pada List Question Set

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Label** | **Field** | **Tabel/Query** | **I/O** | **Format** | **Validasi** |
| Question Set | questionSet | Question Set | I | Varchar | Harus diisi |

#### Objects Specification

Spesifikasi objek dari Fungsi List Question Set dapat dilihat pada Table 71.

Table . Object pada List Question Set

|  |  |  |
| --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Keterangan** |
| List | Drop-down | Jika diklik akan menampilkan pilihan untuk berapa Question Set yang ingin ditampilkan dalam list. |
| Simpan | Button | Jika diklik, Question Set yang baru akan disimpan ke dallam sistem. |
| New Division | Button | Jika diklik, operator akan dialihkan ke fungsi menambah Question Set. |

#### Error Message

Tidak ada *error message* yang ditampilkan.

#### Algorithm

Algoritma yang terdapat pada Fungsi List Divisi terdapat pada tabel berikut ini.

|  |  |  |
| --- | --- | --- |
| Id\_Proses | : | SDD\_Process\_010\_ List Question Set |
| Objek terkait | : | Question Set |
| Event | : | Click |
| **Initial State (IS):**  Sudah terdapat Question Set pada database. | | |
| **Final State (FS):**  Operator berhasil melihat Question Set apa saja yang terdapat pada instansi mereka. | | |
| **Spesifikasi Proses/Algoritma:**  Operator login ke dalam Sistem Informasi Survey Kepuasan Masyarakat.  Operator melihat Question Set apa saja yang terdapat pada instansi mereka dengan masuk ke dalam menu Question Set. | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification Menambah Question

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi Menambah Question. |
| Deskripsi Isi | : | Fungsi ini digunakan oleh Operator yang sudah login ke dalam Sistem Informasi Survey Kepuasan Masyarakat untuk menambah Question pada instansi mereka. |
| Jenis | : | Form Question*.* |

#### Detail Input Specification

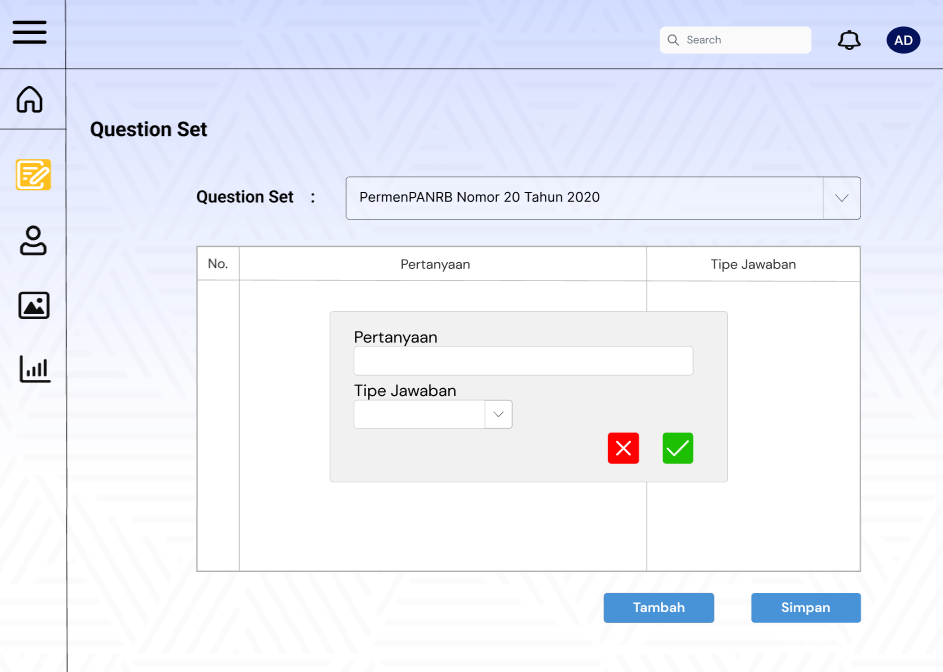
|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | - |

#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | Question |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* Menambah Question yang akan digunakan oleh operator yang sudah login. Tampilan dapat dilihat pada Gambar 34.



Gambar . Detail Layout Menambah Question

#### URL

/questions/create

#### Fields description

Deskripsi *field* pada Fungsi Menambah Question dapat dilihat pada Table 72.

Table . Field pada Fungsi Menambah Question

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Label** | **Field** | **Tabel/Query** | **I/O** | **Format** | **Validasi** |
| Pertanyaan | question | question | I | varchar | Harus diisi |
| Tipe Jawaban | answer\_type | Answer\_type | I | varchar | Harus diisi |

#### Objects Specification

Spesifikasi objek dari Fungsi Menambah Question dapat dilihat pada Table 73.

Table . Objek dari Fungsi Menambah Question

|  |  |  |
| --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Keterangan** |
|  | Button | Jika diklik akan membuat Question baru. |
| X | Button | Jika diklik maka proses membuat Question baru dibatalkan. |
| Tambah | Button | Jika diklik akan membuat Question baru dan menampilkan data services lain yang ingin dibuat. |
| Simpan | Button | Jika diklik maka question baru akan dimasukkan ke dalam sistem. |

#### Error Message

Semua *error message* yang muncul pada Fungsi Menambah Question dapat dilihat pada Table 74.

Table . Error Message pada Fungsi Menambah Question

|  |  |  |
| --- | --- | --- |
| **No.** | **Keterangan** | **Pesan Error yang disampaikan** |
| 1. | Semua field tidak diisi oleh user, sementara user memilih tombol Create | Akan muncul error message yang mengatakan “Please fill out this field” |
| 2. | Field Name tidak diisi, sementara user memilih tombol Create | Akan muncul error message yang mengatakan “Please fill out this field” |
| 3. | Field Reference tidak diisi, sementara user memilih tombol Create | Akan muncul error message yang mengatakan “Please fill out this field” |
| 4. | Semua field tidak diisi oleh user, sementara user memilih tombol Create & create another | Akan muncul error message yang mengatakan “The question field is required.” dan “The answer Type field is required.” |
| 5. | Field Question tidak diisi, sementara user memilih tombol Create & create another | Akan muncul error message yang mengatakan “The question field is required.” |
| 6. | Field Answer Type tidak diisi, sementara user memilih tombol Create & create another | Akan muncul error message yang mengatakan “The answer typr field is required.” |

#### Algorithm

Algoritma yang terdapat pada Fungsi Menambah Services terdapat pada tabel berikut ini.

|  |  |  |
| --- | --- | --- |
| Id\_Proses | : | SDD\_Process\_011\_ Menambah Question |
| Objek terkait | : | Question |
| Event | : | Click |
| **Initial State (IS):**  Operator sudah masuk ke dalam Sistem Informasi Survey Kepuasan Masyarakat. | | |
| **Final State (FS):**  Proses menambahkan Question baru berhasil dilakukan oleh operator yang berada pada instansi tersebut. | | |
| **Spesifikasi Proses/Algoritma:**  Operator yang berada pada instansi tersebut mengisi nama Question apa yang ingin ditambahkan. | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification List Question

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi List Question. |
| Deskripsi Isi | : | Fungsi ini digunakan oleh Operator yang sudah login ke dalam Sistem Informasi Survey Kepuasan Masyarakat untuk melihat Question apa saja yang terdapata pada instansi mereka. |
| Jenis | : | Form Question*.* |

#### Detail Input Specification

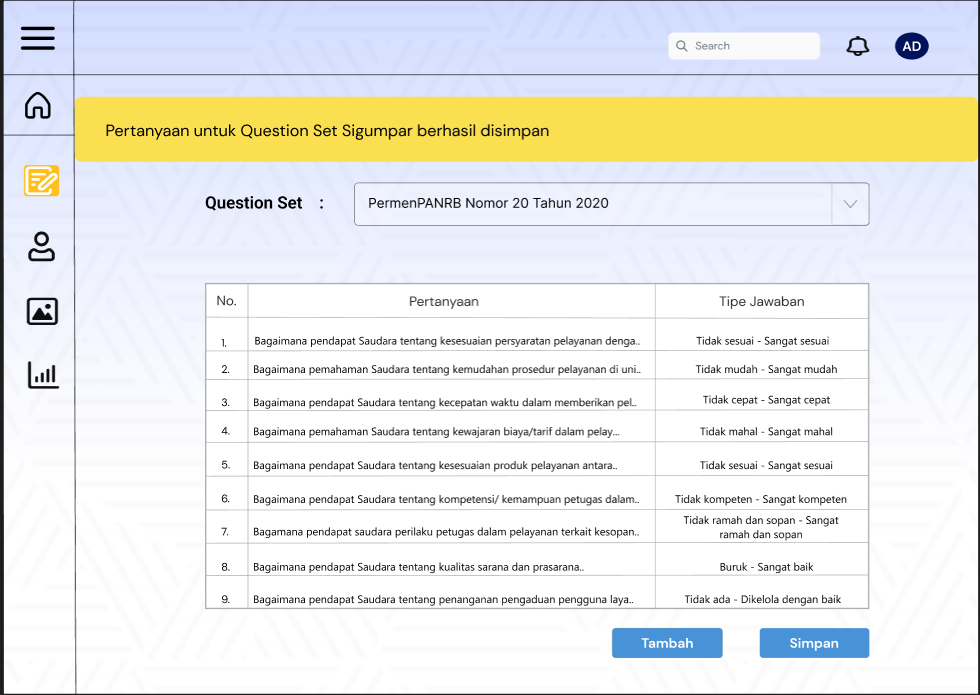
|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | Question |

#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | Question |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* List Question yang akan digunakan oleh operator yang sudah login untuk melihat divisi apa saja yang terdaftar pada instansi mereka. Tampilan dapat dilihat pada Gambar 35.



Gambar . Detail Layout List Question

#### URL

/question/list

#### Fields description

Tidak ada *field* *description* pada Fungsi list question dapat dilihat pada tabel berikut ini.

#### Objects Specification

Spesifikasi objek dari Fungsi List question dapat dilihat pada Table 75.

Table . Objek dari Fungsi List question

|  |  |  |
| --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Keterangan** |
| List | Drop-down | Jika diklik akan menampilkan pilihan untuk berapa Question yang ingin ditampilkan dalam list. |
| Search | Search box | Digunakan untuk mencari apa saja yang berkaitan dengan Question dengan cara mengetikkan kata kuncinya. |
| Tambah | Button | Jika diklik, operator akan dialihkan ke fungsi menambah Question. |
| Simpan | Button | Jika diklik, maka question akan disimpan. |

#### Error Message

Tidak ada *error message* yang ditampilkan.

#### Algorithm

Algoritma yang terdapat pada Fungsi List question terdapat pada tabel berikut ini.

|  |  |  |
| --- | --- | --- |
| Id\_Proses | : | SDD\_Process\_012\_ List Question |
| Objek terkait | : | Question |
| Event | : | Click |
| **Initial State (IS):**  Sudah terdapat Question pada database. | | |
| **Final State (FS):**  Operator berhasil melihat Question apa saja yang terdapat pada instansi mereka. | | |
| **Spesifikasi Proses/Algoritma:**  Operator login ke dalam Sistem Informasi Survey Kepuasan Masyarakat.  Operator melihat Question apa saja yang terdapat pada instansi mereka dengan masuk ke dalam menu Question. | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification Report

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi Melihat Report. |
| Deskripsi Isi | : | Fungsi ini digunakan oleh Operator yang sudah login ke dalam Sistem Informasi Survey Kepuasan Masyarakat untuk melihat Report yang terdapata pada instansi mereka. |
| Jenis | : | Form Report*.* |

#### Detail Input Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | respondent |
|  |  | division |
|  |  | service |

#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | respondent |
|  |  | division |
|  |  | service |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* Report yang akan digunakan oleh operator yang sudah login untuk melihat divisi apa saja yang terdaftar pada instansi mereka. Tampilan dapat dilihat pada Gambar 36.



Gambar . Detail layout Report

#### URL

/report

#### Fields description

Tidak terdapat *field* pada Fungsi Report.

#### Objects Specification

Tidak terdapat spesifikasi objek dari Fungsi report.

#### Error Message

Tidak ada *error message* yang ditampilkan.

#### Algorithm

Algoritma yang terdapat pada Fungsi List question terdapat pada tabel berikut ini.

|  |  |  |  |
| --- | --- | --- | --- |
| Id\_Proses | | : | SDD\_Process\_013\_ Reports |
| Objek terkait | | : | Report |
| Event | | : | Click |
| **Initial State (IS):**  Sudah terdapat Report pada database. | | | |
| **Final State (FS):**  Operator berhasil melihat Report apa saja yang terdapat pada instansi mereka. | | | |
| **Spesifikasi Proses/Algoritma:**  Operator login ke dalam Sistem Informasi Survey Kepuasan Masyarakat.  Operator melihat Report apa saja yang terdapat pada instansi mereka dengan masuk ke dalam menu Reports. | | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification Menambah Divisi Baru

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi Menambah Divisi Baru. |
| Deskripsi Isi | : | Fungsi ini digunakan oleh Operator akan menambahkan divisi baru pada instansinya |
| Jenis | : | Form Divisions*.* |

#### Detail Input Specification

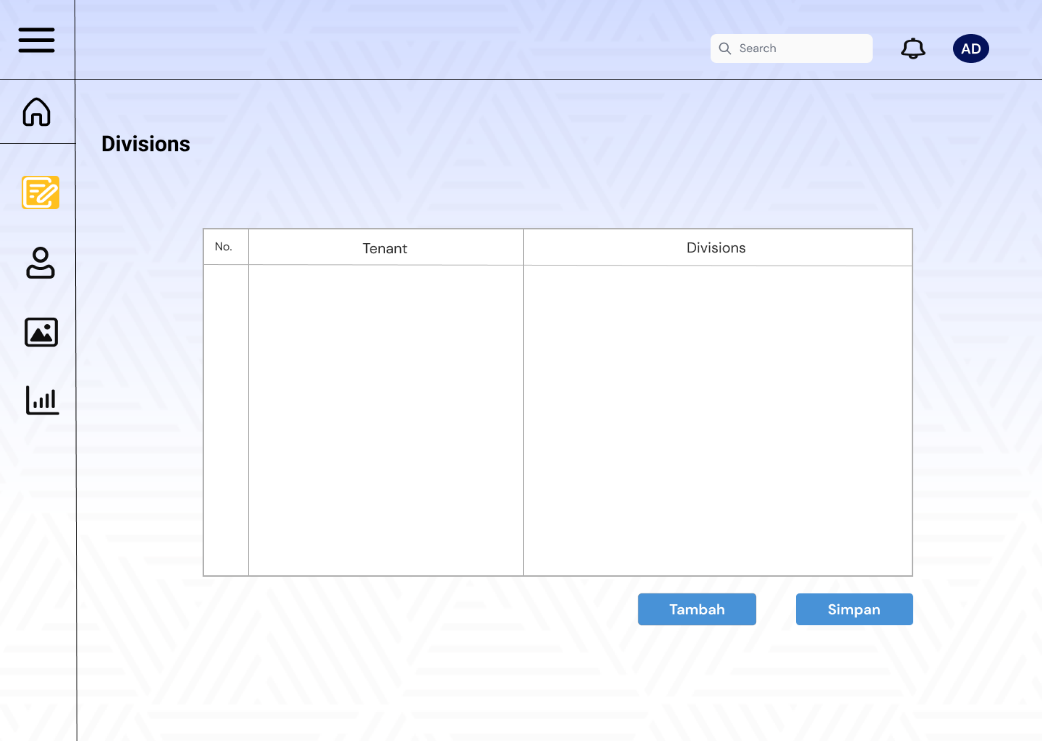
Tidak ada table input yang digunakan.

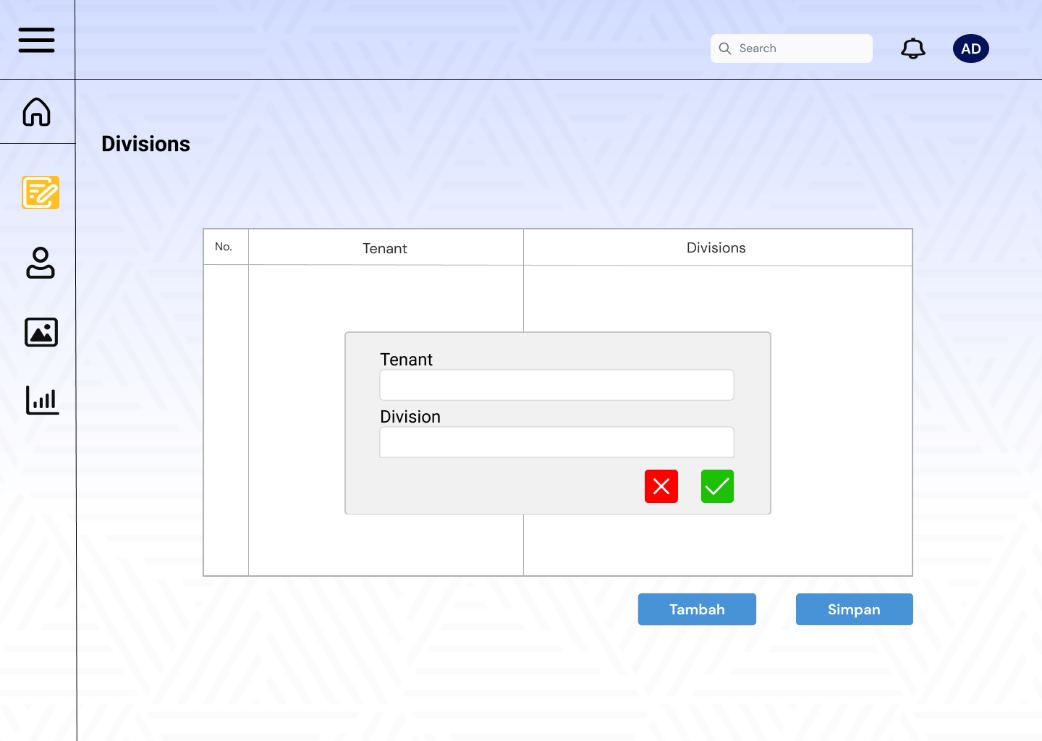
#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | divisions |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* Menambah Divisi Baru yang akan digunakan oleh operator untuk menambah divisi baru pada instansi mereka. Tampilan dapat dilihat pada .





Gambar . Detail Layout Menambah Divisi

#### URL

/divisions/create.

#### Fields description

Field Description untuk Menambah Divisi Baru ditampilkan pada Table 76.

Table . Fields Description Menambah Divisi Baru

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Label | Field | Tabel | I/O | Format | Validasi |
| Nama | name | users | I | varchar(255) | Harus diisi |

#### Objects Specification

Tidak terdapat spesifikasi objek dari Fungsi Menambah Divisi Baru.

#### Error Message

Tidak ada *error message* yang ditampilkan.

#### Algorithm

Algoritma yang terdapat pada Fungsi Menambah Divisi Baru terdapat pada tabel berikut ini.

|  |  |  |  |
| --- | --- | --- | --- |
| Id\_Proses | | : | SDD\_Process\_014\_ Reports |
| Objek terkait | | : | Report |
| Event | | : | Click |
| **Initial State (IS):**  Operator sudah berada pada Dashboard. | | | |
| **Final State (FS):**  Operator berhasil membuat divisi baru. | | | |
| **Spesifikasi Proses/Algoritma:**   1. Operator membuka menu Divisions. 2. Klik “New Division”. 3. Masukkan nama divisi. 4. Klik “Save changes”.   Disimpan ke database. | | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification Mengedit Divisi

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi Mengedit Divisi. |
| Deskripsi Isi | : | Fungsi ini digunakan oleh Operator yang sudah login ke dalam Sistem Informasi Survey Kepuasan Masyarakat untuk mengedit divisi yang terdapat pada instansi mereka. |
| Jenis | : | Form Divisions*.* |

#### Detail Input Specification

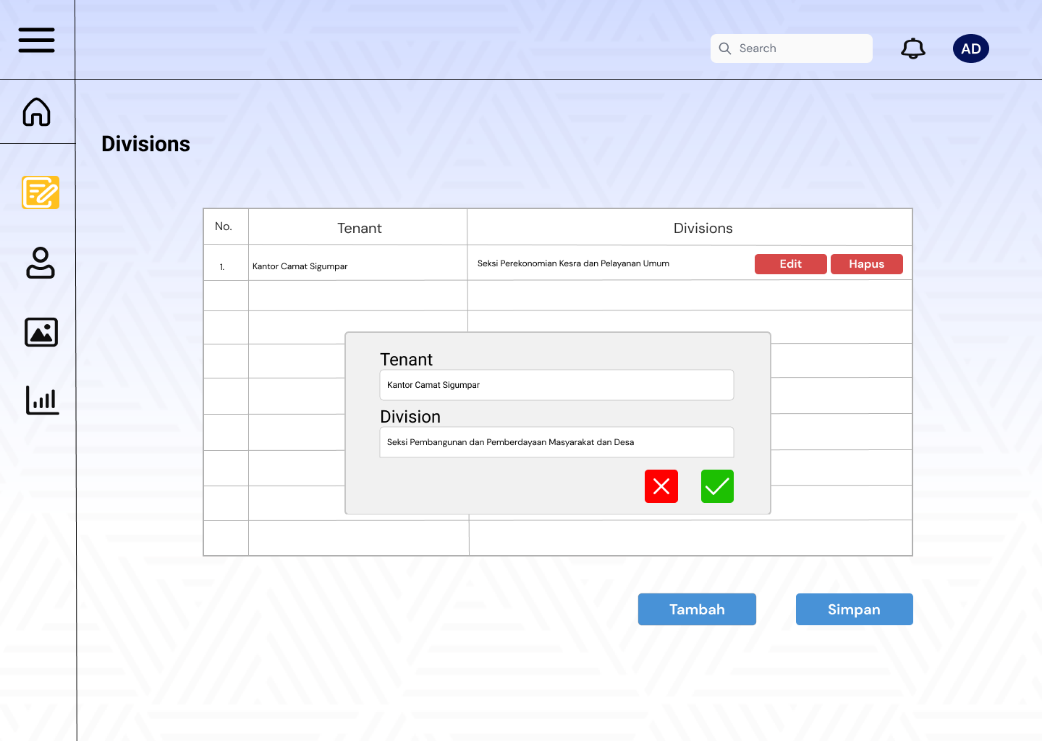
|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | divisions |

#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | divisions |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* Mengedit Divisi yang akan digunakan oleh operator yang sudah login untuk mengedit divisi yang terdaftar pada instansi mereka. Tampilan dapat dilihat pada Gambar 38.



Gambar . Detail Layout Mengedit Divisi

#### URL

/divisions/edit.

#### Fields description

Field Description untuk Menambah Divisi Baru ditampilkan pada Table 77.

Table . Field Description Mengedit Divisi

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Label** | **Field** | **Tabel** | **I/O** | **Format** | **Validasi** |
| Nama | name | users | I | varchar(255) | Harus diisi |

#### Objects Specification

Tidak terdapat spesifikasi objek dari Fungsi Mengedit Divisi.

#### Error Message

Tidak ada *error message* yang ditampilkan.

#### Algorithm

Algoritma yang terdapat pada Fungsi Mengedit Divisi terdapat pada tabel berikut ini.

|  |  |  |  |
| --- | --- | --- | --- |
| Id\_Proses | | : | SDD\_Process\_015\_ Reports |
| Objek terkait | | : | Report |
| Event | | : | Click |
| **Initial State (IS):**  Sudah terdapat Divisi. | | | |
| **Final State (FS):**  Operator berhasil mengedit divisi. | | | |
| **Spesifikasi Proses/Algoritma:**   1. Operator membuka menu Divisions. 2. Klik “edit”. 3. Mengganti nama divisi sesuai dengan yang diinginkan. 4. Klik “Save changes”.   Disimpan di database. | | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification Menghapus Divisi

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi Menghapus Divisi. |
| Deskripsi Isi | : | Fungsi ini digunakan oleh Operator yang sudah login ke dalam Sistem Informasi Survey Kepuasan Masyarakat untuk menghapus divisi yang terdapat pada instansi mereka. |
| Jenis | : | Form Report*.* |

#### Detail Input Specification

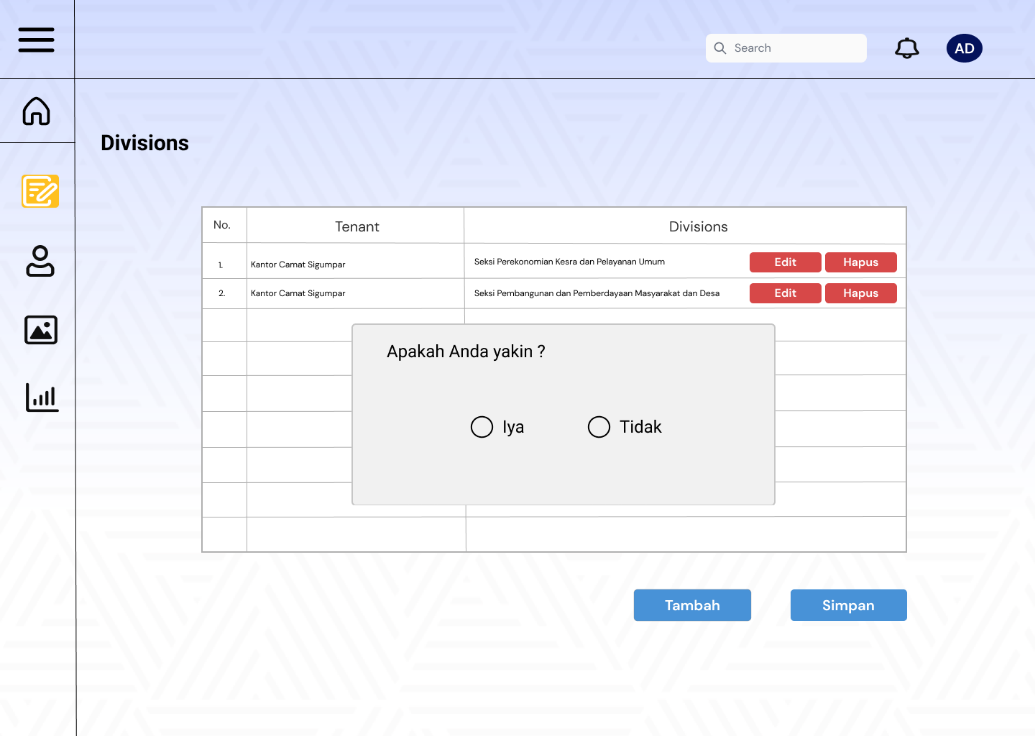
|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | divisions |

#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | divisions |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* Menghapus Divisi yang akan digunakan oleh operator yang sudah login untuk menghapus divisi yang terdaftar pada instansi mereka. Tampilan dapat dilihat pada Gambar 39.



Gambar . Detail layout Menghapus Divisi

#### URL

/divisions/delete

#### Fields description

Tidak terdapat *field* pada Fungsi Menghapus Divisi.

#### Objects Specification

Tidak terdapat spesifikasi objek dari Fungsi Menghapus Divisi.

#### Error Message

Tidak ada *error message* yang ditampilkan.

#### Algorithm

Algoritma yang terdapat pada Fungsi Menghapus Divisi terdapat pada tabel berikut ini.

|  |  |  |  |
| --- | --- | --- | --- |
| Id\_Proses | | : | SDD\_Process\_016\_ Reports |
| Objek terkait | | : | Report |
| Event | | : | Click |
| **Initial State (IS):**  Sudah terdapat divisi. | | | |
| **Final State (FS):**  Operator berhasil menghapus divisi. | | | |
| **Spesifikasi Proses/Algoritma:**   1. Operator membuka menu Divisions. 2. Memilih salah satu divisi yang hendak dihapus. 3. Klik “Delete”. 4. Pilih “Confirm”.   Disimpan di database. | | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification Menambah Layanan Baru

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi Menambah Layanan Baru. |
| Deskripsi Isi | : | Fungsi ini digunakan oleh Operator akan menambahkan layanan baru pada instansinya |
| Jenis | : | Form Divisions*.* |

#### Detail Input Specification

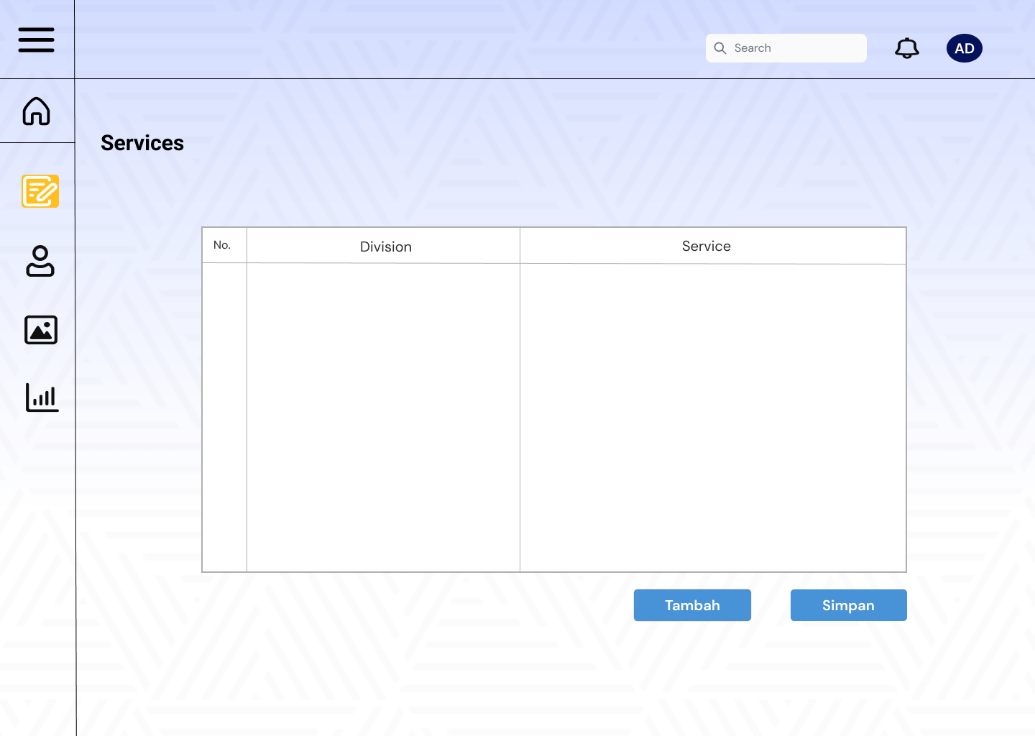
Tidak ada table input yang digunakan.

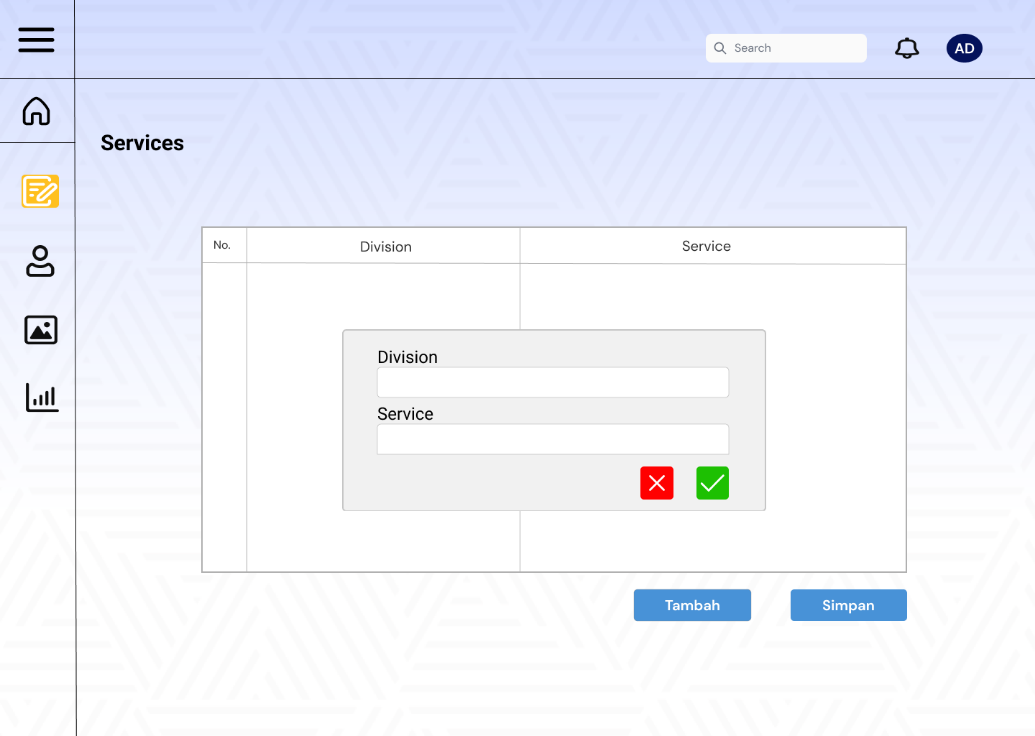
#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | services |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* Menambah Layanan Baru yang akan digunakan oleh operator untuk menambah divisi baru pada instansi mereka. Tampilan dapat dilihat pada Gambar 40.





Gambar . Detail Layout pada Menambah Layanan Baru

#### URL

/services/create.

#### Fields description

Field Description untuk Menambah Layanan Baru ditampilkan pada Table 78.

Table . Field Description Menambah Layanan Baru

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Label** | **Field** | **Tabel** | **I/O** | **Format** | **Validasi** |
| Nama Divisi | divisions.name | users | I | varchar(255) | Harus diisi |
| Nama Layanan | services.name | users | I | varchar(255) | Harus diisi |

#### Objects Specification

Tidak terdapat spesifikasi objek dari Fungsi Menambah Layanan Baru.

#### Error Message

Tidak ada *error message* yang ditampilkan.

#### Algorithm

Algoritma yang terdapat pada Fungsi Menambah Layanan Baru terdapat pada tabel berikut ini.

|  |  |  |  |
| --- | --- | --- | --- |
| Id\_Proses | | : | SDD\_Process\_017\_ Reports |
| Objek terkait | | : | Report |
| Event | | : | Click |
| **Initial State (IS):**  Operator sudah berada pada Dashboard. | | | |
| **Final State (FS):**  Operator berhasil membuat service baru. | | | |
| **Spesifikasi Proses/Algoritma:**   1. Operator membuka menu Services. 2. Klik “New Service”. 3. Memilih divisi. 4. Masukkan nama service. 5. Klik “Save changes”.   Disimpan ke database. | | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification Mengedit Layanan

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi Mengedit Layanan. |
| Deskripsi Isi | : | Fungsi ini digunakan oleh Operator yang sudah login ke dalam Sistem Informasi Survey Kepuasan Masyarakat untuk mengedit layanan yang terdapat pada instansi mereka. |
| Jenis | : | Form Services*.* |

#### Detail Input Specification

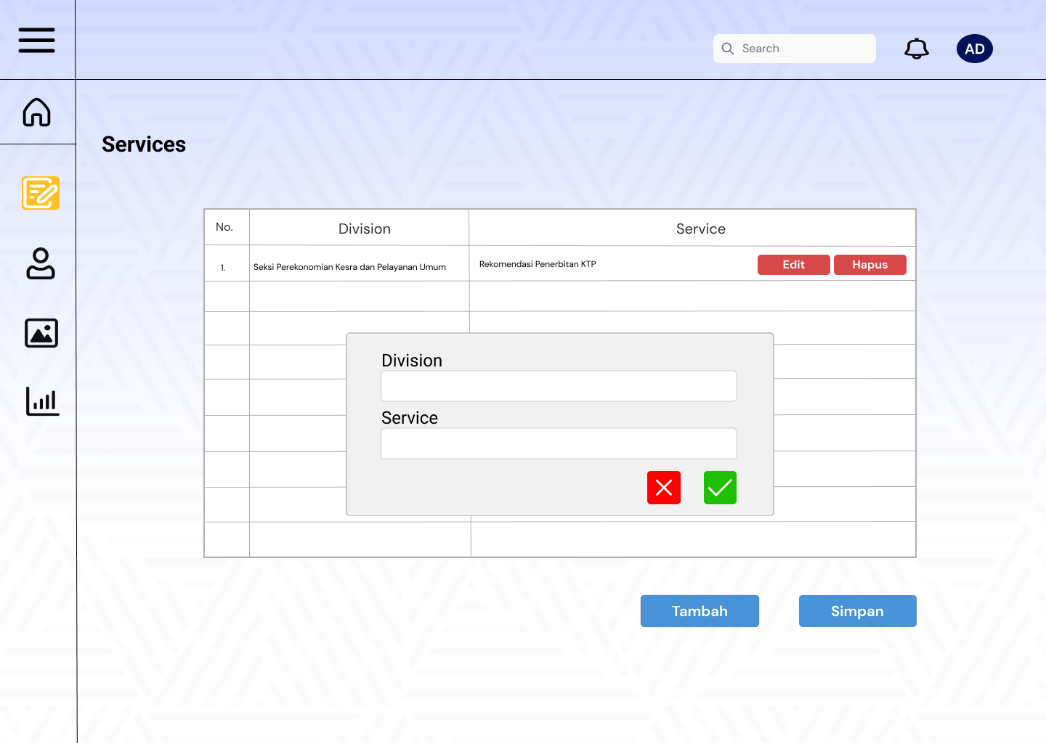
|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | services |

#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | services |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* Mengedit Layanan yang akan digunakan oleh operator yang sudah login untuk mengedit layanan yang terdaftar pada instansi mereka. Tampilan dapat dilihat pada Gambar 41.



Gambar . Detail Layout Mengedit Layanan

#### URL

/divisions/edit.

#### Fields description

Berikut Field Description Menambah Layanan Baru, dilihat pada Table 79.

Table . Field Description Mengedit Layanan

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Label** | **Field** | **Tabel** | **I/O** | **Format** | **Validasi** |
| Nama Divisi | divisions.name | users | I | varchar(255) | Harus diisi |
| Nama Layanan | services.name | users | I | varchar(255) | Harus diisi |

#### Objects Specification

Tidak terdapat spesifikasi objek dari Fungsi Mengedit Layanan.

#### Error Message

Tidak ada *error message* yang ditampilkan.

#### Algorithm

Algoritma yang terdapat pada Fungsi Mengedit Layanan terdapat pada tabel berikut ini.

|  |  |  |  |
| --- | --- | --- | --- |
| Id\_Proses | | : | SDD\_Process\_018\_ Reports |
| Objek terkait | | : | Report |
| Event | | : | Click |
| **Initial State (IS):**  Sudah terdapat Services. | | | |
| **Final State (FS):**  Operator berhasil mengedit service. | | | |
| **Spesifikasi Proses/Algoritma:**   1. Operator membuka menu Services. 2. Klik “edit”. 3. Mengganti nama service sesuai dengan yang diinginkan. 4. Klik “Save changes”.   Disimpan di database. | | | |

#### Spesifikasi Report

Tidak ada.

### Function Spesification Menghapus Layanan

|  |  |  |
| --- | --- | --- |
| Identifikasi/Nama | : | Fungsi Menghapus Layanan. |
| Deskripsi Isi | : | Fungsi ini digunakan oleh Operator yang sudah login ke dalam Sistem Informasi Survey Kepuasan Masyarakat untuk menghapus layanan yang terdapat pada instansi mereka. |
| Jenis | : | Form Report*.* |

#### Detail Input Specification

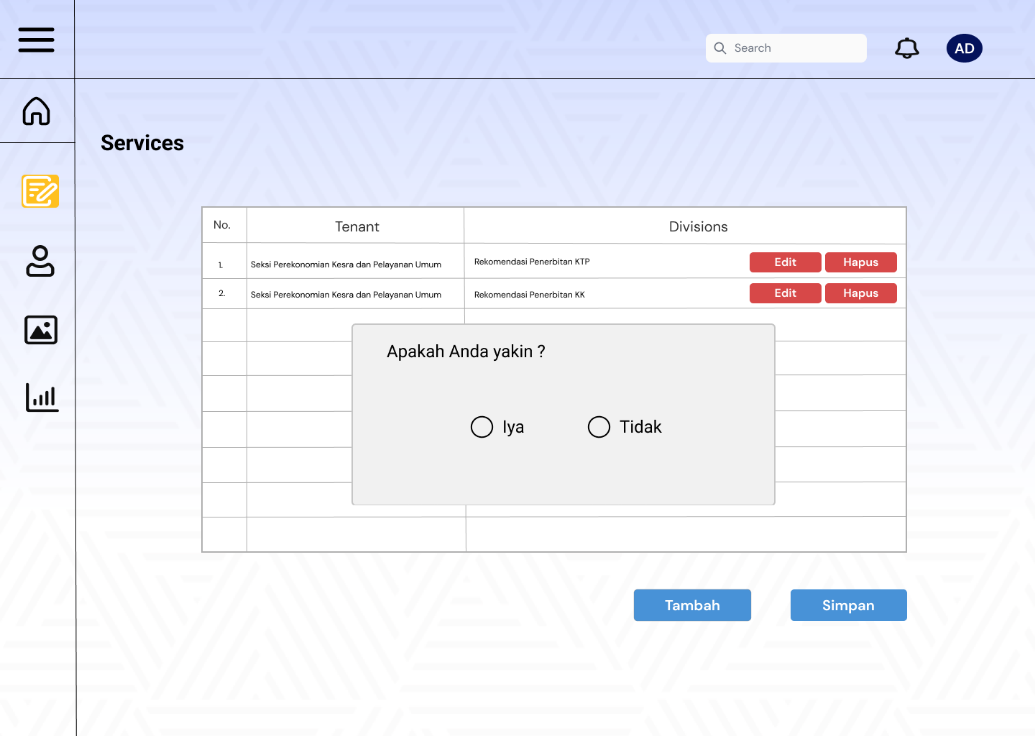
|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | services |

#### Detail Output Specification

|  |  |  |
| --- | --- | --- |
| Nama Tabel | : | services |

#### Detail Screen Layout

Sub-bab ini menampilkan *detail layout* Menghapus Layanan yang akan digunakan oleh operator yang sudah login untuk menghapus layanan yang terdaftar pada instansi mereka. Tampilan dapat dilihat pada Gambar 42



Gambar . Detail Layout Menghapus Layanan

#### URL

/serviecs/delete

#### Fields description

Tidak terdapat *field* pada Fungsi Menghapus Layanan.

#### Objects Specification

Tidak terdapat spesifikasi objek dari Fungsi Menghapus Layanan.

#### Error Message

Tidak ada *error message* yang ditampilkan.

#### Algorithm

Algoritma yang terdapat pada Fungsi Menghapus Layanan terdapat pada tabel berikut ini.

|  |  |  |  |
| --- | --- | --- | --- |
| Id\_Proses | | : | SDD\_Process\_019\_ Reports |
| Objek terkait | | : | Report |
| Event | | : | Click |
| **Initial State (IS):**  Sudah terdapat service. | | | |
| **Final State (FS):**  Operator berhasil menghapus servide. | | | |
| **Spesifikasi Proses/Algoritma:**   1. Operator membuka menu Services. 2. Memilih salah satu service yang hendak dihapus. 3. Klik “Delete”. 4. Pilih “Confirm”.   Disimpan di database. | | | |

#### Spesifikasi Report

Tidak ada.

## Physical File

Tidak terdapat *physical file*.

## Traceability

Pada bab ini dijelaskan penyimpanan data yang digunakan pada sistem dan keterkaitannya dengan ER.

### Data

Pada bagian ini dijelaskan traceability dari tabel aplikasi yang dirancang terhadap Entity Class dan ER. Traceability ditampilkan pada Table 80.

Table . Traceability Tabel Aplikasi Terhadap DFD dan ER

| **Nama Tabel** | **Primary key** | **Data Store** | **E/R** | **Deskripsi isi** |
| --- | --- | --- | --- | --- |
| age\_options | id | AgeOption | age\_option | Berelasi one-to-many dengan tabel tenants |
| answers | id | Answer | answer | Berelasi many-to-one dengan tabel respondents, tabel questions, dan tabel answer\_options |
| answer\_options | id | AnswerOption | answer\_option | Berelasi one-to-many dengan tabel answers dan many-to-one dengan tabel answer\_types |
| answer\_types | id | AnswerType | answer\_type | Berelasi one-to-many dengan tabel answer\_options dan tabel questions |
| districts | id | District | district | Berelasi many-to-one dengan tabel regencies dan one-to-many tabel villages |
| divisions | id | Division | division | Berelasi one-to-many dengan tabel respondents dan tabel services, dan many-to-one dengan tabel tenants |
| occupation\_options | id | Occupation | occupation | Berelasi one-to-many dengan tabel respondents |
| provinces | id | Province | province | Berelasi one-to-many dengan tabel tenants dan tabel regencies |
| questions | id | Question | question | Berelasi one-to-many dengan tabel answers dan many-to-one dengan tabel answer\_types dan question\_sets |
| question\_sets | id | QuestionSet | question\_set | Berelasi one-to-many dengan tabel questions dan many-to-one dengan tabel tenants |
| regencies | id | Regency | regency | Berelasi one-to-many dengan tabel districts dan many-to-one dengan tabel provinces |
| respondents | id | Respondent | respondent | Berelasi one-to-many dengan tabel answers dan many-to-one dengan tabel tabel age\_options, tabel occupation\_options, tabel villages, tabel tenants, tabel divisions, dan tabel services |
| services | id | Service | service | Berelasi one-to-many dengan tabel respondents dan many-to-one dengan tabel divisions |
| tenants | id | Tenant | tenant | Berelasi one-to-many dengan tabel respondents, tabel divisions, tabel users, dan tabel question\_sets, dan berelasi many-to-one dengan tabel provinces, tabel districts, dan tabel regencies |
| users | id | User | user | Berelasi many-to-one dengan tabel tenants |
| villages | id | Village | village | Berelasi one-to-many dengan tabel respondents dan many-to-one dengan tabel districts |

### Requirements

Daftar traceability functional ditampilkan pada Table 81. Pada sistem terdapat beberapa fungsi utama, yakni sebagai berikut.

Table . Requirements

| **SRS-Id** | **No. Fungsi** | **Keterangan** |
| --- | --- | --- |
| SyRS-FR-01 | Registrasi | *User* yaitu operator melakukan registrasi dengan mengisi *form* registrasi |
| SyRS-FR-02 | Login | *User* yaitu operator *login* ke sistem dengan memasukkan *email address* dan *password* |
| SyRS-FR-03 | Melihat Divisions | *User* yaitu operator melihat seluruh division dari sebuah tenant |
| SyRS-FR-04 | Menambah Division | *User* yaitu operator menambah division untuk sebuah tenant |
| SyRS-FR-05 | Melihat Question Sets | *User* yaitu operator melihat seluruh question set dari sebuah tenant |
| SyRS-FR-06 | Menambah Question Set | *User* yaitu operator menambah question set untuk sebuah tenant |
| SyRS-FR-07 | Melihat Services | *User* yaitu operator melihat seluruh services dari sebuah tenant |
| SyRS-FR-08 | Menambah Services | *User* yaitu operator menambah service untuk sebuah tenant |
| SyRS-FR-09 | Melihat Users | *User* yaitu operator melihat seluruh users (operator) dari sebuah tenant |
| SyRS-FR-10 | Menambah User | *User* yaitu operator menambah user sebagai operator untuk sebuah tenant |
| SyRS-FR-11 | Mengekspor Laporan | *User* yaitu operator mengekspor data laporan kinerja tenant terkait dalam bentuk PDF atau Excel |
| SyRS-FR-12 | Melihat Statistik | *Guest* melihat statistik kinerja sebuah tenant |
| SyRS-FR-13 | Mengisi Kuesioner | *Guest* mengisi kuesioner mengenai kinerja sebuah division pada sebuah tenant |

# Lampiran

Lampiran yang diperlukan oleh dokumen ini dapat dibuat subbab sesuai keperluan

# Sejarah Versi

Pada bagian ini, dijelaskan semua versi yang pernah di-deliver, dan ciri serta perubahannya.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Versi | Ditulis Oleh | Tanggal | Disetujui Oleh | Tanggal |
| 1 | Kelompok PSI06 | 01 Mei 2023 | Mario E. S. Simaremare, S.Kom., M.Sc. | 01 Mei 2023 |
| 2 | Kelompok PSI06 | 02 Mei 2023 | Mario E. S. Simaremare, S.Kom., M.Sc | 02 Mei 2023 |
| 3 | Kelompok PSI06 | 05 Mei 2023 | Mario E. S. Simaremare, S.Kom., M.Sc | 05 Mei 2023 |
| 4 | Kelompok PSI06 | 06 Mei 2023 | Mario E. S. Simaremare, S.Kom., M.Sc |  |
|  |  |  |  |  |

# Sejarah Perubahan

No. dokumen :

No. versi :

| Halaman | Semula | Menjadi | Alasan perubahan |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

No. dokumen :

No. versi :

| Halaman | Semula | Menjadi | Alasan perubahan |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |