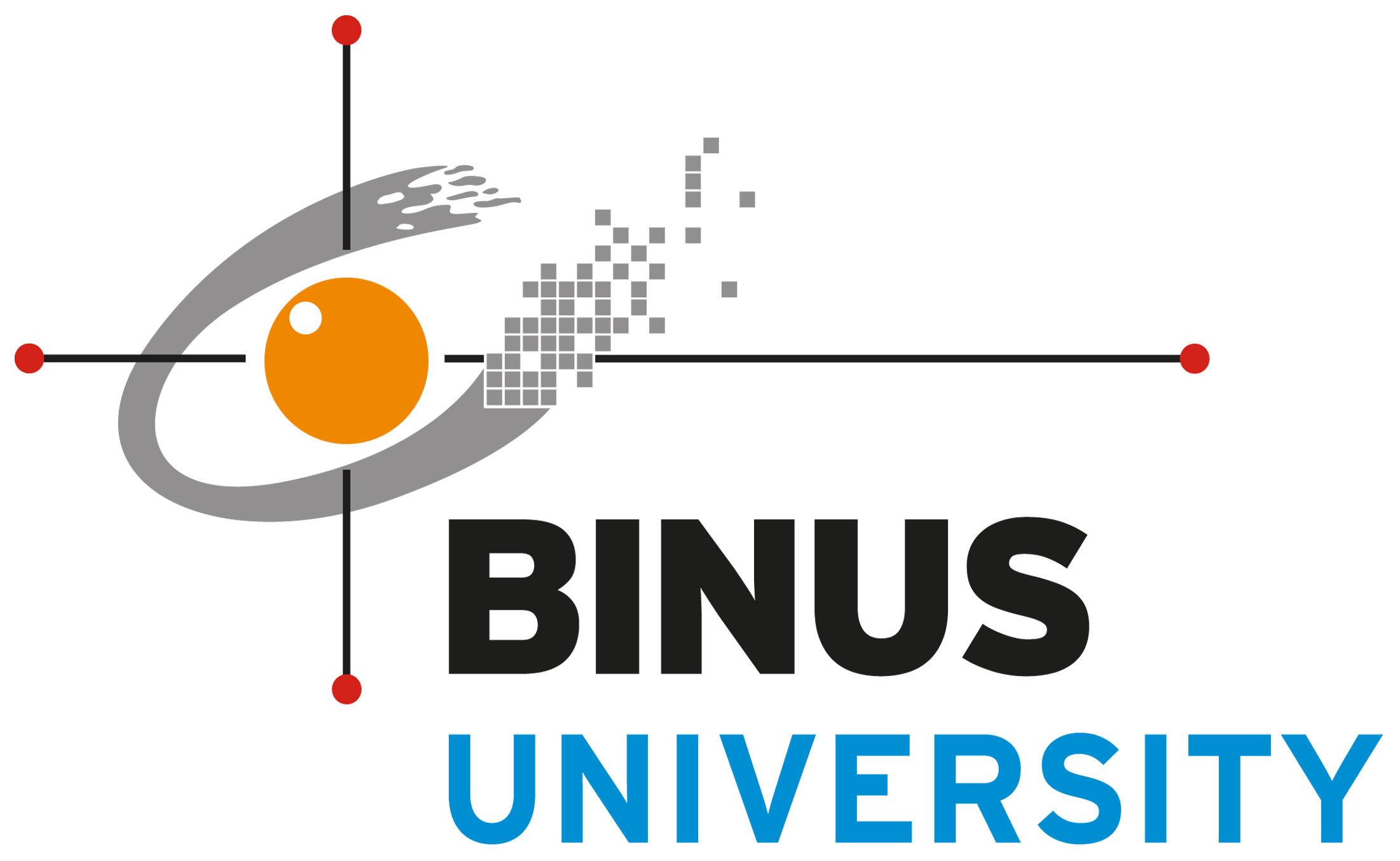
**PERANCANGAN DATABASE KEPENDUDUKAN DI TINGKAT RT/RW**

**UNTUK PEMBAYARAN IURAN LINGKUNGAN DAN PENGAJUAN SURAT SURAT**

LAPORAN PROYEK AKHIR

MATA KULIAH ISYS6028 - DATABASE SYSTEM

KELAS BA20 - LAB



Oleh:

2301943491 - Julio Rivaldo Gunawan

2301878952 - Putri Aurelia Shilo

2301879583 - Vincentius Sutanto

Semester Ganjil 2020/2021

MALANG

**LEMBAR PERSETUJUAN PROYEK AKHIR**

**PERANCANGAN DATABASE KEPENDUDUKAN DI TINGKAT RT/RW**

**UNTUK PEMBAYARAN IURAN LINGKUNGAN DAN PENGAJUAN SURAT SURAT**

ISYS6028 - DATABASE SYSTEM

KELAS BA20 - LAB

Semester Ganjil 2020/2021

Laporan akhir proyek ini adalah benar karya kami:

2301943491 – Julio Rivaldo Gunawan 2301878952 – Putri Aurelia Shilo

2301879583 – Vincentius Sutanto

**Malang, 15 Januari 2021**

**Wina Permana Sari**

**D5975**

**DAFTAR ISI**

LEMBAR PERSETUJUAN 2

DAFTAR ISI 3

DAFTAR GAMBAR 5

BAB 1 PENDAHULUAN 6

1.1 Latar Belakang 6

1.2 Rumusan Masalah 7

1.3 Tujuan Penelitian 8

1.4 Manfaat Penelitian 8

1.5 Sistematika Penulisan 8

BAB 2 LANDASAN TEORI 10

2.1 Database 10

2.1.1 Pengertian Database 10

2.1.2 Jenis-jenis Database 11

2.1.3 Fungsi Database 12

2.1.4 Proses Perancangan Database 12

2.2 Kependudukan 14

2.2.1 Pengertian Kependudukan 14

2.2.2 Administrasi Kependudukan 15

2.2.3 Struktur Pemerintahan dalam Kependudukan 15

2.3 Data Kependudukan 16

2.3.1 Pengertian Data Kependudukan 16

2.3.2 Manfaat Data Kependudukan 16

BAB 3 ANALISIS 18

3.1 System Definition 18

3.2 Identifikasi Permasalahan 19

3.3 Identifikasi Kebutuhan Pengguna 19

BAB 4 DESIGN 21

4.1 Conceptual Design 21

4.1.1 Entity Types 21

4.1.2 Relationship Types 22

4.1.3 Attributes 23

4.1.4 Attribute Domains 24

4.1.5 Candidate Keys, Primary Keys, dan Alternate Keys 26

4.1.6 Entity Relationship Diagram (ERD) 27

4.2 Logical Model (Class Diagram) 28

4.3 Physical Model 29

4.3.1 List of Index 29

4.3.2 Analyze Transaction 29

4.3.3 User View, Procedure, and Function 30

DAFTAR PUSTAKA 35

LAMPIRAN 36

LEMBAR PENILAIAN 61

**DAFTAR GAMBAR**

Gambar 3.1 Use Case Diagram 18

Gambar 4.1 Entity Types 21

Gambar 4.2 Relationship Types 22

Gambar 4.3 Attributes 23

Gambar 4.4 Candidate Keys, Primary Keys, dan Alternate Keys 26

Gambar 4.5 Entity Relationship Diagram (ERD) 27

Gambar 4.6 Logical Model (Class Diagram) 28

Gambar 4.7 List of Index 29

Gambar 4.8 Analyze Transaction 29

Gambar 4.9 View 1 30

Gambar 4.10 View 2 30

Gambar 4.11 View 3 31

Gambar 4.12 Procedure 1 31

Gambar 4.13 Procedure 2 32

Gambar 4.14 Procedure 3 32

Gambar 4.15 Procedure 4 33

Gambar 4.16 Function 1 33

Gambar 4.17 Function 2 34

Gambar 4.18 Function 3 34

**BAB 1**

**PENDAHULUAN**

**1.1 Latar Belakang**

Seiring berkembangnya jaman, teknologi semakin banyak digunakan oleh masyarakat. Bahkan, dapat dikatakan bahwa teknologi sudah menjadi kebutuhan bagi masyarakat. Hal ini dikarenakan dengan adanya teknologi, pekerjaan manusia menjadi lebih mudah. Salah satu contoh teknologi yang banyak digunakan oleh masyarakat sekarang adalah *gadget*. *Gadget* merupakan suatu perangkat elektronik yang dapat membuat hidup manusia menjadi lebih praktis. Adapun contoh dari *gadget*, yakni *smartphone*, laptop, komputer, tablet, dan sebagainya.

*Gadget* dapat membantu mempermudah pekerjaan manusia, seperti berkomunikasi, memperoleh informasi melalui internet, memperoleh wawasan, hiburan, dan sebagainya. Saat ini, perkembangan teknologi yang pesat juga dapat memungkinkan manusia untuk melakukan transaksi jual beli serta mengurus dokumen-dokumen penting hanya dengan menggunakan *gadget*. Bahkan, untuk membayar tagihan listrik, air dan tagihan bulanan lainnya pun manusia tidak perlu repot lagi untuk pergi keluar rumah. Hal ini menunjukkan bahwa di kemudian hari, segala bentuk transaksi pembayaran dan kepengurusan dokumen akan dapat dilakukan secara praktis melalui aplikasi pada *gadget*, termasuk pembayaran pajak, kepengurusan surat-surat, serta iuran bulanan di lingkungan RT/RW. Faktanya, masih banyak sistem pelayanan masyarakat yang kurang efektif jika dilihat dari perkembangan teknologi yang sangat pesat ini, terutama di lingkungan RT/RW. Hal ini dapat dilihat dari rumitnya proses kepengurusan surat-surat kependudukan serta masih banyaknya sistem pembayaran iuran yang belum terstruktur (banyak masyarakat yang lupa atau tidak tahu kapan harus membayar iuran hingga ketua RT yang harus mendatangi rumah-rumah warga atau mengirimkan surat tagihan secara satu persatu).

Seiring berjalannya waktu, sistematika kepengurusan surat-surat kependudukan dan pembayaran iuran di lingkungan RT/RW yang masih menggunakan cara lama (non-aplikasi) akan semakin “tertinggal” oleh perkembangan teknologi yang sangat pesat. Masyarakat (terutama orang tua) juga tidak akan dapat terbiasa untuk menggunakan teknologi dalam kehidupan sehari-hari. Selain itu, proses pelayanan yang kurang efektif tersebut juga dapat menyebabkan masyarakat mengabaikan tanggung jawabnya dalam membayar iuran serta mengurus segala perizinan atau surat-surat penting. Kurangnya sistem pembayaran yang terstruktur juga dapat memperbesar kemungkinan adanya ketidaksesuaian dalam mengolah keluar masuknya dana dari masyarakat. Permasalahan lain yang akan muncul adalah akan timbul kemungkinan terdapat suatu dokumen atau surat-surat penting yang hilang karena tidak adanya *back up* data yang terkait. Hal-hal tersebut sangatlah penting untuk diperhatikan karena sangat berpengaruh bagi perkembangan lingkungan RT/RW tersebut.

Jika permasalahan mengenai sistem pelayanan masyarakat di lingkungan RT/RW tersebut tidak segera diatasi, maka lingkungan RT/RW tersebut tidak akan dapat berkembang dengan baik. Oleh karena itu, untuk mengatasi permasalahan tersebut diperlukan adanya pengembangan dalam prosedur pembayaran iuran dan kepengurusan surat-surat kependudukan. Hal ini dapat dilakukan dengan pembuatan database untuk menyimpan data-data masyarakat di lingkungan RT/RW serta data mengenai iuran bulanan yang wajib dibayarkan. Dengan dibuatnya database tersebut, maka data-data masyarakat akan lebih aman serta terstruktur. Masyarakat juga dapat terbiasa dalam menggunakan teknologi untuk mempermudah pekerjaannya serta dapat dengan mudah mengikuti perkembangan teknologi. Selain itu, adanya aplikasi mengenai kependudukan tersebut juga dapat mempermudah masyarakat dalam membayar iuran bulanan serta pengajuan kepengurusan dokumen-dokumen tertentu yang dibutuhkan.

Hal-hal di atas lah yang melatarbelakangi pemilihan topik Kependudukan dalam perancangan database ini. Selanjutnya, hasil perancangan database tersebut akan diuraikan dalam laporan berjudul *Perancangan Database Kependudukan di Tingkat RT/RW untuk Pembayaran Iuran Lingkungan dan Pengajuan Surat-Surat* ini.

**1.2 Rumusan Masalah**

Rumusan masalah yang digunakan sebagai pedoman dalam penulisan laporan ini adalah sebagai berikut.

1. Bagaimana pengaruh perancangan database kependudukan terhadap sistem pembayaran iuran lingkungan pada tingkat RT/RW?
2. Bagaimana pengaruh perancangan database kependudukan terhadap sistem pengajuan surat-surat pada lingkungan RT/RW?

**1.3 Tujuan Penelitian**

Tujuan dilakukannya perancangan database kependudukan pada tingkat RT/RW untuk pembayaran iuran lingkungan dan pengajuan surat-surat adalah sebagai berikut.

1. Untuk mengetahui bagaimana pengaruh perancangan database kependudukan terhadap sistem pembayaran iuran lingkungan pada tingkat RT/RW.
2. Untuk mengetahui bagaimana pengaruh perancangan database kependudukan terhadap sistem pengajuan surat-surat pada lingkungan RT/RW.

**1.4 Manfaat Penelitian**

Adanya penelitian dan perancangan database kependudukan ini diharapkan dapat bermanfaat bagi para pembaca. Manfaat-manfaat yang dapat diambil dalam penelitian dan perancangan ini adalah sebagai berikut.

1. Bagi Penulis, hasil penelitian ini dapat dijadikan sebagai referensi untuk menambah wawasan mengenai proses perancangan database kependudukan serta pengaruhnya terhadap sistem pembayaran iuran lingkungan dan pengajuan surat-surat pada tingkat RT/RW.
2. Bagi Masyarakat, hasil penelitian ini dapat dijadikan sebagai referensi untuk lebih mengerti mengenai dampak perkembangan teknologi bagi sistem pembayaran dan pengajuan surat-surat di lingkungan RT/RW serta lebih mudah dalam memanfaatkan aplikasi yang berbasis database untuk melakukan pembayaran iuran dan pengajuan surat-surat.
3. Bagi Pengurus RT/RW, hasil penelitian ini dapat dijadikan sebagai referensi untuk lebih mengerti akan pentingnya perancangan database kependudukan serta penggunaannya untuk mempermudah berjalannya segala proses pembayaran iuran dan pengajuan surat-surat dari masyarakat.

**1.5 Sistematika Penulisan**

Sistematika penulisan dari laporan yang berjudul *Perancangan Database Kependudukan di Tingkat RT/RW untuk Pembayaran Iuran Lingkungan dan Pengajuan Surat-Surat* ini terdiri atas halaman judul, daftar isi, daftar tabel, daftar gambar, bab 1-4, daftar pustaka dan lampiran. Penjelasan mengenai subbab dari setiap bab akan dijabarkan sebagai berikut.

1. Pada Bab 1, yakni Pendahuluan, berisikan tentang latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.
2. Pada Bab 2 , yakni Landasan Teori, berisikan tentang database (pengertian, jenis-jenis, fungsi, dan proses perancangan database), kependudukan (pengertian dan administrasi kependudukan), dan data kependudukan (pengertian dan manfaat data kependudukan).
3. Pada Bab 3, yakni Analisis, berisikan tentang *system definition*, identifikasi permasalahan, dan identifikasi kebutuhan pengguna.
4. Pada Bab 4, yakni Design, berisikan tentang *conceptual design* (*entity types*; *relationship types*; *attributes*; *attribute domains*; *candidate keys*, *primary keys*, *and alternate keys*; *redundant relationship*; dan *entity relationship diagram* (ERD)), *logical model* (*Class Diagram*), dan *physical model* (*list of index*; *analyze transaction*; *user view, procedure, and function*)

**BAB 2**

**LANDASAN TEORI**

Landasan teori pada penelitian ini meliputi subbab (1) database, (2) kependudukan, dan (3) data kependudukan. Berikut ini merupakan penjelasan dari setiap subbab tersebut.

**2.1 Database**

**2.1.1 Pengertian Database**

Manusia menggunakan berbagai macam perangkat elektronik seperti *smartphone*, laptop, komputer, dan sebagainya untuk mengakses berbagai macam aplikasi yang ada. Faktanya, pada setiap aplikasi pasti memiliki suatu hal yang sangat berkaitan dengan data-data pada aplikasi tersebut, yakni *database*. Database merupakan salah satu istilah yang paling sering ditemui pada dunia komputer. Database berasal dari kata “data” dan “*base* (basis)”. Data berarti sekumpulan fakta atau informasi yang mewakili suatu objek (manusia, hewan, barang, dan sebagainya), sedangkan *base* berarti dasar, wadah atau tempat untuk berkumpul.

Secara singkat, *database* merupakan sekumpulan data atau informasi yang saling berhubungan dan disimpan secara sistematik pada suatu komputer dengan tujuan tertentu, yakni memperoleh informasi yang baru dari data-data tersebut. Data atau informasi yang terdapat pada suatu *database* berupa potongan-potongan fakta atau pengetahuan yang dijabarkan dalam bentuk tabel serta memiliki penjelasan secara terstruktur. Dengan kata lain, data-data yang terdapat pada suatu perangkat komputer akan disimpan dalam suatu *database* sehingga dapat dengan mudah diatur dan dikelola oleh pengguna komputer tersebut kapan saja. Adapun contoh dari penggunaan *database* adalah data pegawai, data mahasiswa, data penjualan produk, data perpustakaan, dan sebagainya.

*Database* dapat dikelola dengan menggunakan *Database Management System* (DBMS). DBMS merupakan suatu perangkat lunak yang dapat memungkinkan pengguna untuk melakukan berbagai macam tindakan pada *database* yang ada secara efektif dan efisien. Tindakan-tindakan yang dimaksud adalah membuat, memelihara, mengatur, mengontrol, serta mengakses *database* tersebut. Dengan adanya DBMS, pengguna dapat lebih mudah dalam mengelola *database* yang ada pada perangkat komputernya.

**2.1.2 Jenis-jenis Database**

Dalam suatu *database*, terdapat berbagai macam data yang digunakan oleh lebih dari satu pengguna yang memiliki tujuan yang berbeda-beda. Oleh karena itu, *database* dapat dibagi menjadi beberapa jenis. Berikut penjelasan mengenai beberapa jenis *database* menurut fungsinya yang sering ditemui.

1. *Operational Database*

*Operational database* merupakan salah satu jenis *database* yang dapat menyimpan data secara rinci. Hal ini bertujuan agar data tersebut dapat dioperasikan dari seluruh organisasi. Data-data yang dioperasikan dapat disebut sebagai *Subject Area Database* (SADB), transaksi *database*, atau produksi *database*. Adapun contoh dari *operational database*, yakni *database* pelanggan, *database* akuntansi, *database* inventaris, *database* pribadi, dan sebagainya.

1. *Relational Database*

*Relational Database* merupakan jenis *database* yang paling banyak digunakan sebagai standar komputasi bisnis. Jenis *database* ini menggunakan berbagai macam *table* yang berbeda untuk memudahkan pengguna dalam mengakses atau mencari informasi. Pada jenis ini, *query* yang dilakukan dapat melibatkan beberapa tabel karena adanya fungsi relasi.

1. *Distributed Database*

*Distributed database* merupakan *database* yang dapat mendistribusikan data-data yang ada secara tersebar namun dapat tetap saling berhubungan dan diakses secara bersamaan. Jenis *database* ini mencakup dua segmen, yakni operasional dan *user database*. Nantinya, data yang dihasilkan dari jenis ini akan dapat digunakan oleh pengguna situs sendiri. *Distributed database* banyak digunakan pada perusahaan yang memiliki beberapa cabang untuk membantu mempermudah akses dari cabang satu ke cabang yang lain.

1. *External Database*

*External database* merupakan *database* yang dapat menyediakan akses pada bagian eksternal. Data yang disimpan pada *database* jenis ini nantinya akan digunakan untuk keperluan komersial. Kelebihan dari jenis *database* ini adalah adanya kemudahan bagi pengguna untuk mengaksesnya. Hal ini dikarenakan *database* ini bersifat publik serta kaya akan informasi sehingga pengguna tidak perlu mencari informasi dengan menggunakan internet.

Penjelasan di atas merupakan jenis-jenis *database* yang paling sering ditemui dan digunakan. Adapun jenis-jenis *database* lainnya, yakni *analytical database*, *end-user database*, *data warehouse*, *in-memory database*, *real time database*, *navigational database*, *hypermedia database on the web*, *document-oriented database*, dan sebagainya.

**2.1.3 Fungsi Database**

*Database* merupakan kumpulan data yang memiliki banyak fungsi pada bidang komputer. Bahkan, saat ini *database* juga sudah banyak digunakan pada bidang-bidang lain di luar bidang komputer, seperti bidang keuangan, pendidikan, bisnis, perbankan, dan sebagainya. Seperti yang telah diketahui, *database* merupakan suatu sistem yang dapat digunakan untuk menyimpan dan mengelola data atau informasi secara sistematis. Adapun beberapa fungsi dari *database*, yakni:

1. Mempermudah dan mempercepat proses identifikasi data secara lengkap dan akurat dengan cara mengelompokkan serta menyiapkan data yang sesuai dengan permintaan pengguna.
2. Membantu meminimalisir terjadinya kemungkinan data yang sama (data ganda atau rangkap) serta *redudancy* data pada suatu *database* dengan memberikan informasi atau peringatan kepada pengguna.
3. Membantu memudahkan beberapa pengguna dalam mengakses, meng*update* (mengedit atau menghapus), serta menyimpan data secara bersamaan.
4. Membantu mencegah terjadinya kerusakan data yang disimpan pada *database* dengan menjaga kualitas data atau informasi agar data tersebut dapat tetap sama pada saat *entry* berlangsung dan setelahnya.
5. Menyediakan ruang penyimpanan untuk data-data dari suatu sistem atau aplikasi beserta dengan pengembangan-pengembangan yang akan dilakukan.
6. Menjaga keamanan suatu data dengan mencegah terjadinya pengaksesan data dari pihak-pihak asing yang tidak bertanggung jawab.

**2.1.4 Proses Perancangan Database**

Perancangan suatu *database* yang baik untuk memenuhi kebutuhan pengguna merupakan suatu hal yang sangat penting untuk diperhatikan. Hal ini bertujuan agar pengguna dapat merasa puas serta dimudahkan dalam mengelola data-datanya. Oleh karena itu, penting bagi seorang pengembang untuk dapat merancang model fisik dari *database* tersebut serta model konseptual yang dapat menunjukkan entitas dan relasi yang sesuai dengan kebutuhan pengguna. Adapun beberapa tahapan pada proses perancangan *database*, yakni:

1. Pengumpulan dan analisis data.

Pada tahap ini, perancang akan melakukan proses identifikasi dan analisa terhadap berbagai macam kebutuhan data. Hal ini dilakukan dengan menentukan kelompok pengguna yang akan memakai *database* tersebut, meninjau dokumen yang berhubungan dengan *database* yang dianalisa, menganalisa lingkungan operasi serta rencana penggunaan data, dan membuat kuesioner atau wawancara dengan pihak-pihak terkait untuk memperoleh informasi yang diperlukan.

1. Perancangan atau desain *database* secara konseptual.

Pada tahap ini, data diproses dengan menggunakan kombinasi beberapa cara. Dalam melakukan pendekatan data, terdapat dua macam teknik yang digunakan, yakni

1. Teknik Normalisasi, yakni pengelompokan elemen pada data ke dalam beberapa tabel yang menunjukkan entitas dan relasinya.
2. Teknik *Entity Relationship*, yakni teknik yang menunjukkan hubungan dari berbagai entitas data yang terdapat pada suatu database.
3. Pemilihan *Database Management System* (DBMS) yang sesuai.

Pada tahap ini, terdapat berbagai macam faktor yang mempengaruhi pemilihan DBMS, yakni faktor teknis (berkaitan dengan jenis DBMS yang akan digunakan), faktor ekonomi (berhubungan dengan seluruh biaya yang akan dibutuhkan dalam pembuatan database), dan faktor organisasi (berhubungan dengan struktur organisasi serta pengguna yang akan memakai sistem tersebut).

1. Perancangan *database* secara logika dengan menggunakan *data model mapping* (pemetaan model data).

Pada tahap ini, terdapat dua proses pemetaan model data, yakni pemetaan yang tidak bergantung pada sistem (*system-independent*) dan penyesuaian skema pada DBMS yang spesifik. Pemetaan *system-independent* merupakan pemetaan yang tidak mempertimbangkan karakteristik khusus pada implementasi DBMS, sedangkan penyesuaian skema pada DBMS yang spesifik dilakukan dengan mengatur skema yang telah dihasilkan serta menyesuaikannya dengan implementasi khusus dari model data yang digunakan pada DBMS.

1. Perancangan atau desain *database* secara fisik.

Pada tahap ini terjadi proses pemilihan struktur penyimpanan serta jalur-jalur akses pada file *database* guna mencapai tampilan yang terbaik dari berbagai aplikasi. Terdapat beberapa petunjuk dalam pemilihan desain *database* secara fisik, yakni *response time* (waktu yang dibutuhkan oleh transaksi suatu *database* untuk menjalankan respon saat eksekusi), *space utility* (jumlah memori yang dibutuhkan untuk *file database* dan struktur-struktur jalur akses), dan *transaction throughput* (rata-rata jumlah transaksi yang dapat diproses oleh sistem *database* pada setiap menitnya).

1. Implementasi sistem *database*.

Pada tahap ini, segala perintah dari DBMS dalam *Data Definition Language* (DDL) dan *Storage Definition Language* (SDL) akan dihimpun serta digunakan untuk membuat skema dan *file-file* *database* yang kosong. Kemudian, *database* akan disatukan atau dipopulasikan dengan datanya (apabila data diubah dari sistem komputer sebelumnya, maka diperlukan adanya konversi rutin untuk memformat kembali data dan menyimpannya pada *database* baru). Selanjutnya, transaksi *database* harus diterapkan pada aplikasi yang telah dirancang serta diuji dengan menggunakan perintah DML. Setelah semua proses tersebut selesai, maka data akan disimpan pada *database* dan sistem dapat digunakan.

**2.2 Kependudukan**

**2.2.1 Pengertian Kependudukan**

Kependudukan merupakan suatu hal yang berhubungan dengan seluruh aspek kehidupan atau data individu masyarakat. Data-data tersebut dapat berupa data kelahiran, kematian, seluruh data pada KTP (umur, jenis kelamin, agama, status perkawinan), struktur, persebaran, mobilitas, kehamilan, dan sebagainya. Kependudukan sendiri berasal dari kata dasar penduduk, yang berarti seorang individu yang hidup dan tinggal di negara yang bersangkutan. Sebagai contoh, seorang warga negara Indonesia maupun orang asing yang hidup dan bertempat tinggal di Indonesia merupakan penduduk Indonesia. Kependudukan sangat mempengaruhi kualitas suatu negara. Hal ini dikarenakan tingkat produktivitas masyarakat dalam suatu negara akan turut mempengaruhi kehidupan dalam negara tersebut.

**2.2.2 Administrasi Kependudukan**

Kependudukan memiliki kaitan yang erat dengan administrasi kependudukan. Administrasi kependudukan merupakan serangkaian kegiatan yang menyangkut penataan serta penertiban dokumen dan data kependudukan. Penertiban tersebut dilakukan melalui pendaftaran penduduk, pencatatan sipil, pengelolaan informasi penduduk, serta pendayagunaan hasil tersebut untuk pelayanan publik dan pembangunan pada sektor lain. Pendaftaran penduduk merupakan kegiatan pencatatan biodata penduduk, pelaporan peristiwa kependudukan (perubahan alamat penduduk, status tinggal penduduk, pindah datang, dan sebagainya), serta penerbitan dokumen kependudukan (kartu identitas atau surat keterangan penduduk). Pencatatan sipil merupakan kegiatan pencatatan suatu peristiwa penting yang dialami oleh penduduk. Peristiwa penting yang dimaksud adalah perkawinan, perceraian, kelahiran, kematian, lahir mati, pengakuan anak, pengangkatan anak, pengesahan anak, perubahan nama, perubahan status kewarganegaraan, dan sebagainya.

Setiap penduduk yang telah tercatat dalam administrasi kependudukan pasti akan memiliki Nomor Induk Kependudukan (NIK) masing-masing. NIK merupakan serangkaian angka sebagai identitas seorang penduduk yang bersifat unik, tunggal, serta melekat pada seorang penduduk Indonesia yang telah terdaftar secara resmi. Adapun salah satu contoh dokumen kependudukan yang diterbitkan oleh Instansi Pelaksana dan berlaku di seluruh wilayah NKRI, yakni Kartu Tanda Penduduk (KTP). KTP merupakan kartu identitas resmi penduduk Indonesia yang berfungsi sebagai bukti diri dan dapat diperoleh apabila seorang penduduk Indonesia sudah berumur 17 tahun ke atas.

Melalui penjelasan di atas, dapat disimpulkan bahwa penting bagi suatu negara untuk melakukan pendataan kepada penduduknya secara berkala untuk dapat mengamati perkembangan dan perubahan pada kondisi penduduk yang ada serta memberikan pelayanan yang memadai guna meningkatkan perlindungan, kesejahteraan dan kenyamanan rakyat, serta pembangunan negara.

**2.2.3 Struktur Pemerintahan dalam Kependudukan**

* Pemerintah pusat: Presiden
* Provinsi: Gubernur
* Kabupaten/Kota: Bupati/Wali Kota
* Kecamatan: Camat
* Kelurahan: Kepala Desa
* Dusun: Kepala Dusun
* RW: Kepala RW
* RT: Kepala RT

**2.3 Data Kependudukan**

**2.3.1 Pengertian Data Kependudukan**

Data kependudukan berasal dari kata “data” dan “kependudukan”. Data berarti sekumpulan fakta atau informasi yang dapat diperoleh melalui pengamatan pada suatu objek. Kependudukan merupakan suatu ilmu yang mempelajari tentang dinamika atau perubahan yang terjadi dalam kependudukan manusia. Perubahan-perubahan tersebut meliputi distribusi penduduk, perubahan jumlah penduduk (kelahiran, kematian, migrasi, dan sebagainya), struktur, dan lain-lain.

Menurut Undang-Undang Republik Indonesia Nomor 24 Tahun 2013 tentang Perubahan atas Undang-Undang Nomor 23 Tahun 2006 tentang Administrasi Kependudukan, data kependudukan merupakan data perseorangan dan/ atau data agregat yang terstruktur sebagai hasil dari kegiatan Pendaftaran Penduduk dan Pencatatan Sipil.

Data kependudukan merupakan segala bentuk data penduduk yang diterbitkan baik secara resmi maupun tidak resmi oleh badan-badan pencatatan kependudukan (badan milik pemerintah maupun non pemerintah). Data-data kependudukan tersebut dapat ditampilkan dalam bentuk grafik, angka, gambar, dan sebagainya.

Melalui penjelasan-penjelasan di atas, dapat disimpulkan bahwa data kependudukan merupakan sekumpulan informasi terstruktur (berupa grafik, angka, gambar, dan sebagainya) yang diterbitkan oleh badan-badan pencatatan kependudukan yang berasal dari pemerintahan maupun non pemerintahan, serta berkaitan erat dan diperoleh melalui pengamatan maupun pencatatan terhadap dinamika kependudukan.

**2.3.2 Manfaat Data Kependudukan**

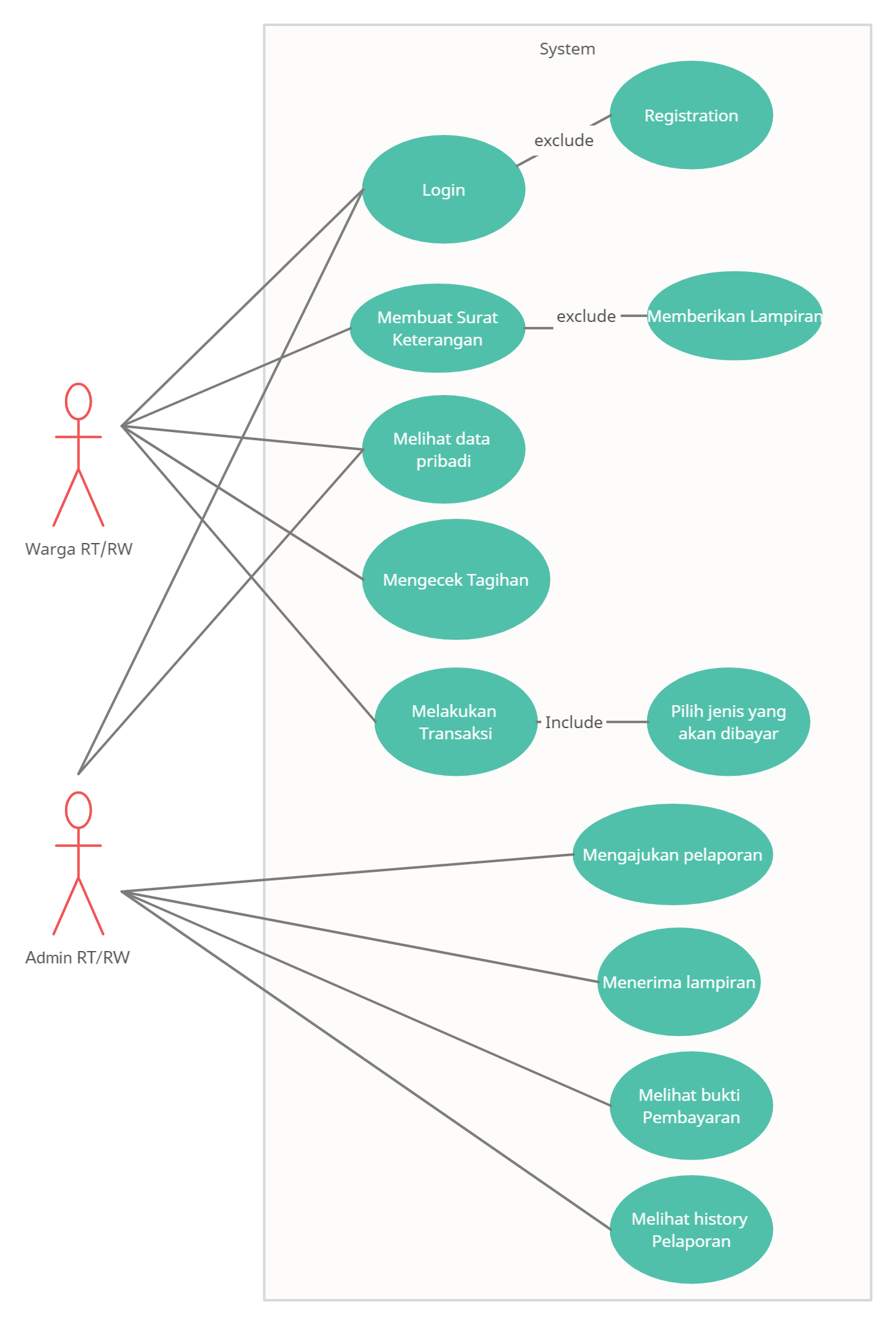
Data kependudukan yang telah diperoleh melalui kegiatan pendaftaran penduduk dan pencatatan sipil dapat berguna untuk memenuhi segala keperluan dari Kementrian Dalam Negeri. Adapun beberapa manfaat dari data kependudukan menurut UU No. 24 Tahun 2013 pasal 58 yang dijabarkan sebagai berikut.

1. Pelayanan publik, yakni penerbitan surat izin mengemudi, izin usaha, pelayanan perbankan, pelayanan penerbitan sertifikat tanah, pelayanan wajib pajak, asuransi, jaminan sosial tenaga kerja, dan jaminan kesehatan masyarakat.
2. Perencanaan pembangunan, yakni perencanaan pendidikan, perencanaan kesehatan, perencanaan pembangunan nasional, perencanaan tenaga kerja, dan pengentasan masyarakat dari kemiskinan.
3. Alokasi anggaran, yakni untuk penentuan dana alokasi umum (DAU) dan perhitungan potensi perpajakan.
4. Pembangunan demokrasi, yakni penyiapan data agregat kependudukan per kecamatan (DAK2) dan penyiapan data penduduk potensial pemilih Pemilu (DP4).
5. Penegakan hukum dan pencegahan kriminal, yakni memudahkan pelacakan pelaku kriminal, mencegah terjadinya perdagangan manusia, dan mencegah pengiriman tenaga kerja secara illegal.

**BAB 3**

**ANALISIS**

**3.1 System Definition**

****

*Gambar 3.1 Use Case Diagram*

**3.2 Identifikasi Permasalahan**

Adapun beberapa identifikasi permasalahan berdasarkan latar belakang yang telah dijabarkan pada Bab Pendahuluan, yakni:

1. Banyaknya sistematika kepengurusan surat-surat, pengajuan laporan, serta pembayaran iuran pada lingkungan RT/RW yang masih menggunakan cara-cara konvensional (non-aplikasi).
2. Kurang efektifnya pelaksanaan proses kepengurusan surat-surat, pengajuan laporan, serta pembayaran iuran pada tingkatan RT/RW dengan menggunakan cara lama.
3. Resiko yang akan timbul pada proses pelayanan masyarakat pada lingkungan RT/RW dengan cara konvensional akan lebih tinggi jika dibandingkan dengan menggunakan teknologi.
4. Pentingnya sistem pembayaran iuran dan pengajuan surat-surat serta *database* terhadap proses pelayanan masyarakat di lingkungan RT/RW.

**3.3 Identifikasi Kebutuhan Pengguna**

Pada saat ini, perkembangan teknologi yang sangat pesat dapat membantu masyarakat dalam menyelesaikan pekerjaannya sehari-hari. Bahkan, kegiatan-kegiatan seperti pembayaran tagihan, transaksi jual beli, serta kepengurusan dokumen-dokumen penting (yang dulu harus dilakukan dengan cara lama) kini dapat dilakukan dengan lebih praktis, yakni melalui *gadget* sehingga masyarakat tidak perlu lagi untuk pergi keluar rumah. Oleh karena itu, dibutuhkan pula adanya sistem pembayaran iuran, pelaporan, serta pengajuan surat-surat pada tingkat RT/RW yang lebih praktis. Padatnya kegiatan masyarakat saat ini membuat mereka membutuhkan suatu aplikasi yang dapat memudahkan mereka untuk menjalankan kewajiban dan hak mereka tanpa harus membuang waktu untuk mengantre di kantor RT/RW. Selain itu, para staf yang bekerja sebagai pengurus RT/RW juga membutuhkan suatu aplikasi yang dapat menyimpan data-data masyarakat untuk meminimalisir adanya kesalahan atau hilangnya dokumen-dokumen penting.

Untuk memenuhi kebutuhan-kebutuhan tersebut, maka dibuatlah sistem database kependudukan untuk pembayaran iuran lingkungan dan pengajuan surat-suratpada tingkat RT/RW. Sistem ini akan menyimpan informasi mengenai data-data penduduk. Selain itu, sistem ini dilengkapi dengan fitur-fitur penting seperti pembayaran iuran, pengajuan surat-surat, serta pelaporan terkait permasalahan di lingkungan RT/RW. Selain memudahkan warga, pengurus RT/RW juga lebih terbantu dalam mengolah data-data masyarakat, terutama ketika hendak melakukan verifikasi data, pembuatan surat keterangan, pengecekan lampiran, pengecekan status pembayaran tagihan, serta demografi warga yang berada pada wilayah tersebut. Nantinya, sistem ini juga akan menyimpan *record* riwayat-riwayat pengajuan surat serta bukti pembayaran iuran yang telah dilunasi oleh masyarakat. Dengan adanya sistem ini, tidak hanya masyarakat yang terbantu, tetapi para pengurus RT/RW juga turut dimudahkan dalam melaksanakan tugasnya sehingga segala proses yang dilakukan dapat berjalan dengan lebih efektif.

**BAB 4**

**DESIGN**

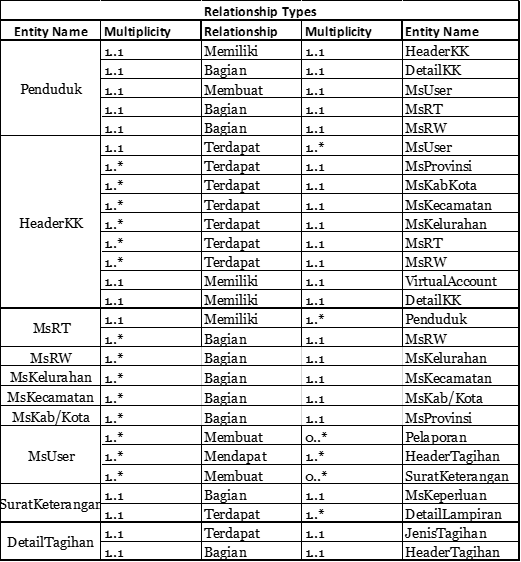
**4.1 Conceptual Design**

**4.1.1 Entity Types**

****

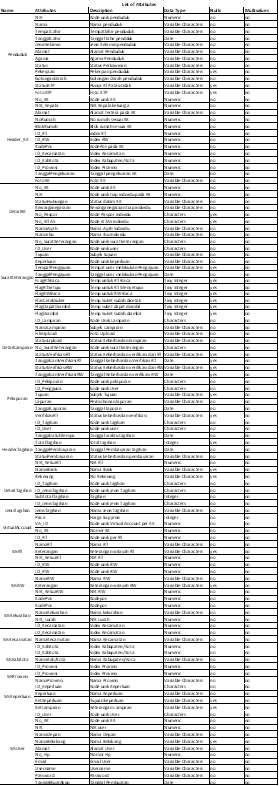
*Gambar 4.1 Entity Types*

**4.1.2 Relationship Types**

****

*Gambar 4.2 Relationship Types*

**4.1.3 Attributes**

****

*Gambar 4.3 Attrributes*

**4.1.4 Attribute Domains**

1. Attribut domain pada tabel PENDUDUK yang memenuhi AGAMA adalah constraint check dimana constraint ini hanya boleh berisi ISLAM /KATOLIK /KRISTEN /HINDU /BUDDHA /KONGHUCU.

2. Attribut domain pada tabel PENDUDUK yang memenuhi JENISKELAMIN adalah constraint check dimana constraint ini hanya boleh berisi L/P.

3. Attribut domain pada tabel PENDUDUK yang memenuhi STATUSKTP adalah constraint check dimana constraint ini hanya boleh berisi PUNYA / BELUM.

4. Attribut domain pada tabel SURATKETERANGAN yang memenuhi NO\_SURATKETERANGAN adalah constraint check dimana constraint ini memiliki panjang 6 karakter dimana karakter pertama dan kedua harus berupa huruf “SK” dan karakter selanjutnya berupa angka dari 0 hingga 9999.

5. Attribut domain pada tabel DETAILLAMPIRAN yang memenuhi ID\_LAMPIRAN adalah constraint check dimana constraint ini memiliki panjang 6 karakter dimana karakter pertama dan kedua harus berupa huruf “LP” dan karakter selanjutnya berupa angka dari 0 hingga 9999.

6. Attribut domain pada tabel PELAPORAN yang memenuhi ID\_PELAPORAN adalah constraint check dimana constraint ini memiliki panjang 6 karakter dimana karakter pertama dan kedua harus berupa huruf “PP” dan karakter selanjutnya berupa angka dari 0 hingga 9999.

7. Attribut domain pada tabel HEADERTAGIHAN yang memenuhi ID\_TAGIHAN adalah constraint check dimana constraint ini memiliki panjang 6 karakter dimana karakter pertama dan kedua harus berupa huruf “TA” dan karakter selanjutnya berupa angka dari 0 hingga 9999.

8. Attribut domain pada tabel JENISTAGIHAN yang memenuhi ID\_JENISTAGIHAN adalah constraint check dimana constraint ini memiliki panjang 4 karakter dimana karakter pertama dan kedua harus berupa huruf “JT” dan karakter selanjutnya berupa angka dari 0 hingga 99.

9. Attribut domain pada tabel MSUSER yang memenuhi ID\_USER adalah constraint check dimana constraint ini memiliki panjang 6 karakter dimana karakter pertama dan kedua harus berupa huruf “US” dan karakter selanjutnya berupa angka dari 0 hingga 9999.

10. Attribut domain pada tabel VIRTUALACCOUNT yang memenuhi VA\_ID adalah constraint check dimana constraint ini memiliki panjang 14 numerik dimana 4 numerik pertama berupa “3001” dan numerik selanjutnya terdiri dari 10 angka antara 0 sampai 9.

11. Attribut domain ID\_USER pada table MSUSER menghubungkan ID\_USER pada table PELAPORANA, HEADERTAGIHAN, dan SURATKETERANGAN.

12. Attribut domain ID\_JENISTAGIHAN pada table JENISTAGIHAN menghubungkan ID\_JENISTAGIHAN pada table DETAILTAGIHAN.

13. Attribut domain NO\_SURATKETERANGAN pada table SURATKETERANGAN menghubungkan SURATKETERANGAN pada table DETAILLAMPIRAN.

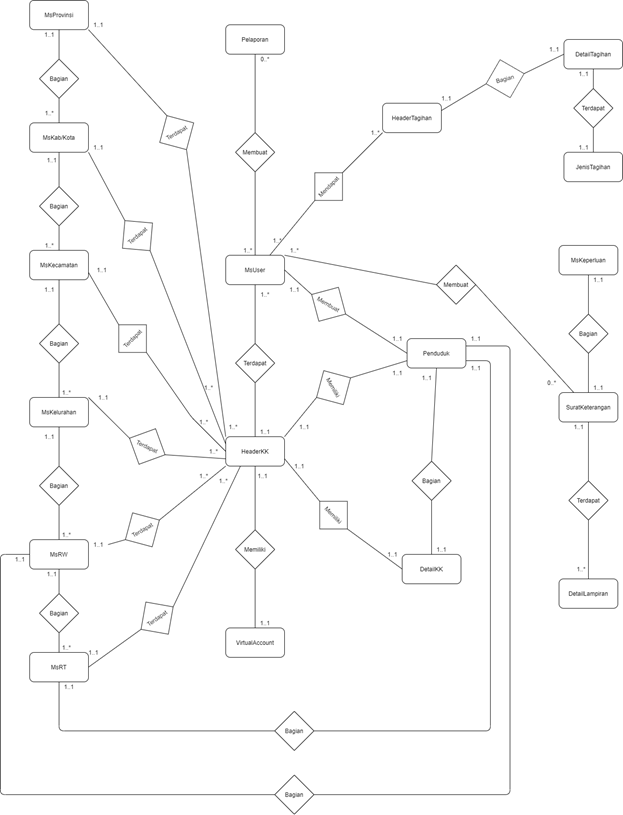
14. Attribut domain pada tabel PENDUDUK yang memenuhi STATUS adalah constraint check dimana constraint ini memiliki panjang 20 karakter hanya boleh berisi “ KAWIN / BELUM KAWIN / CERAI MATI / CERAI HIDUP “..

**4.1.5 Candidate Keys, Primary Keys, dan Alternate Keys**

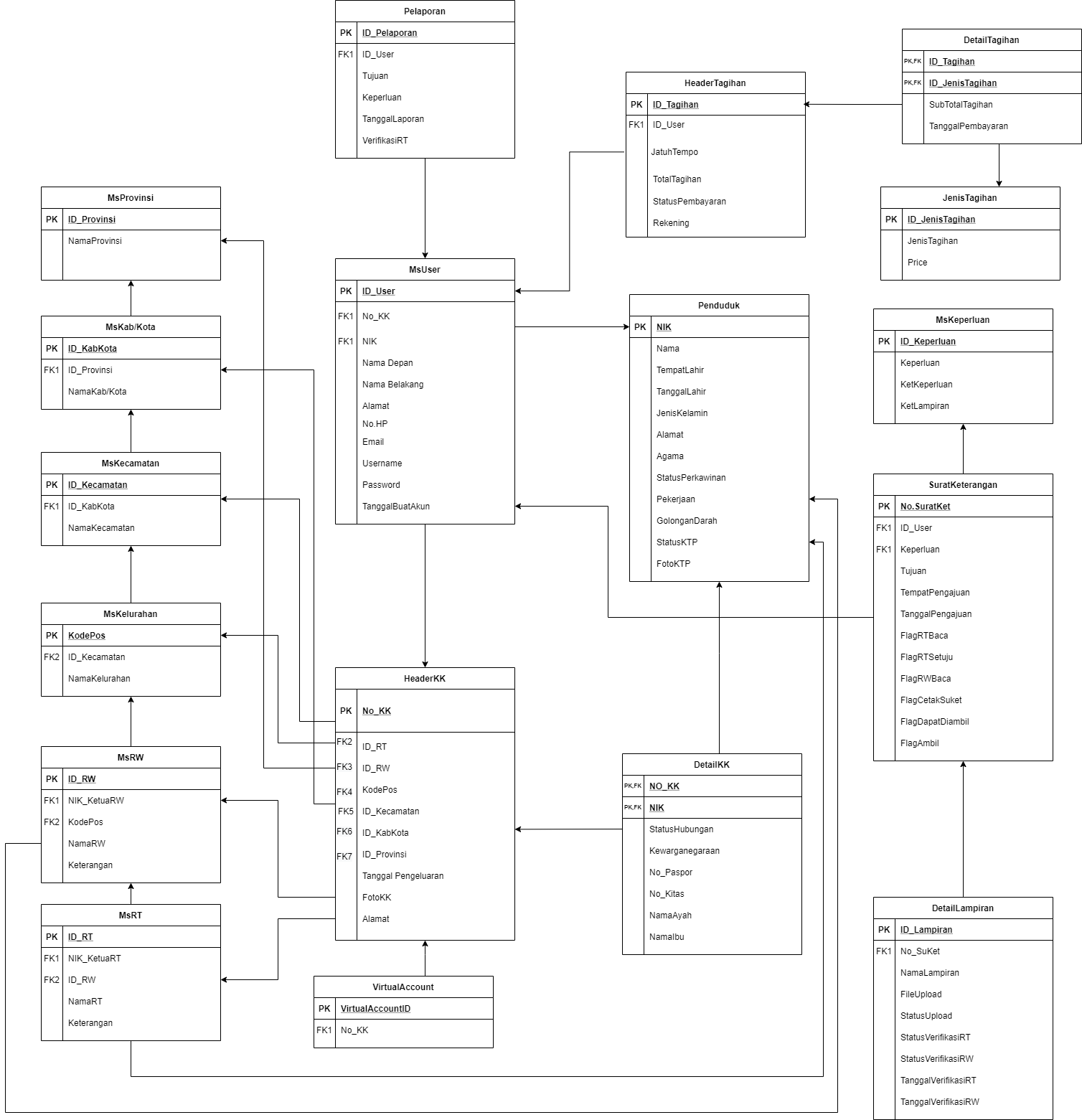
****

*Gambar 4.4 Candidate Keys, Primary Keys, dan Alternate Keys*

**4.1.6 Entity Relationship Diagram (ERD)**

*****Gambar 4.5 Entity Relationship Diagram (ERD)*

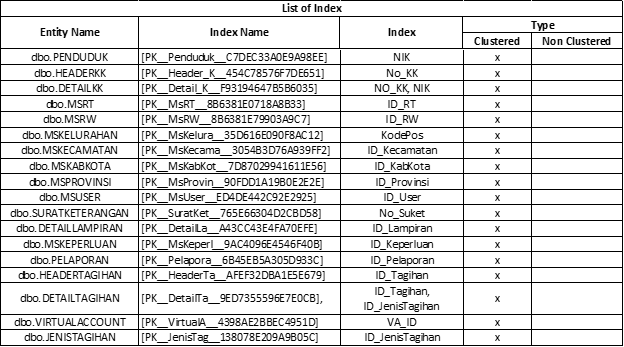
**4.2 Logical Model (Class Diagram)**

****

*Gambar 4.6 Logical Model (Class Diagram)*

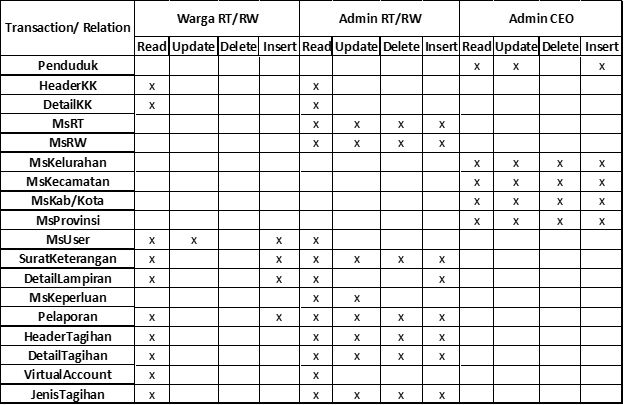
**4.3 Physical Model**

**4.3.1 List of Index**

****

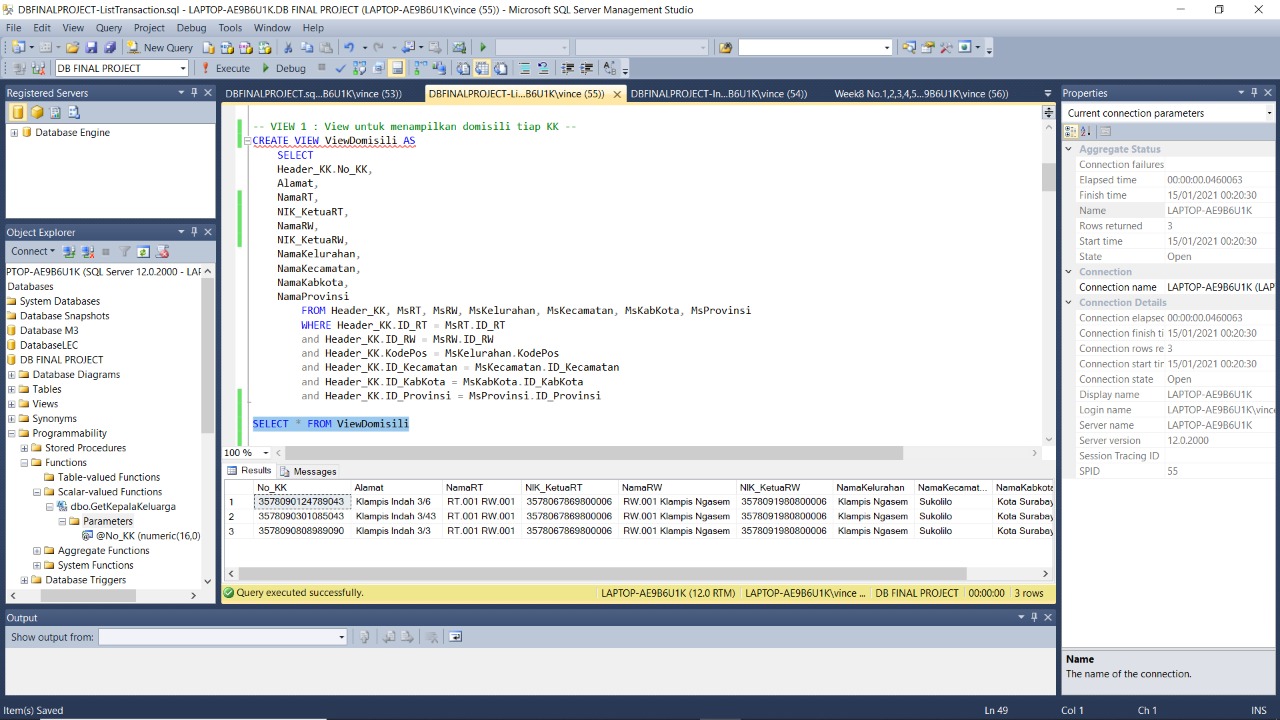
*Gambar 4.7 List of Index*

**4.3.2 Analyze Transaction**

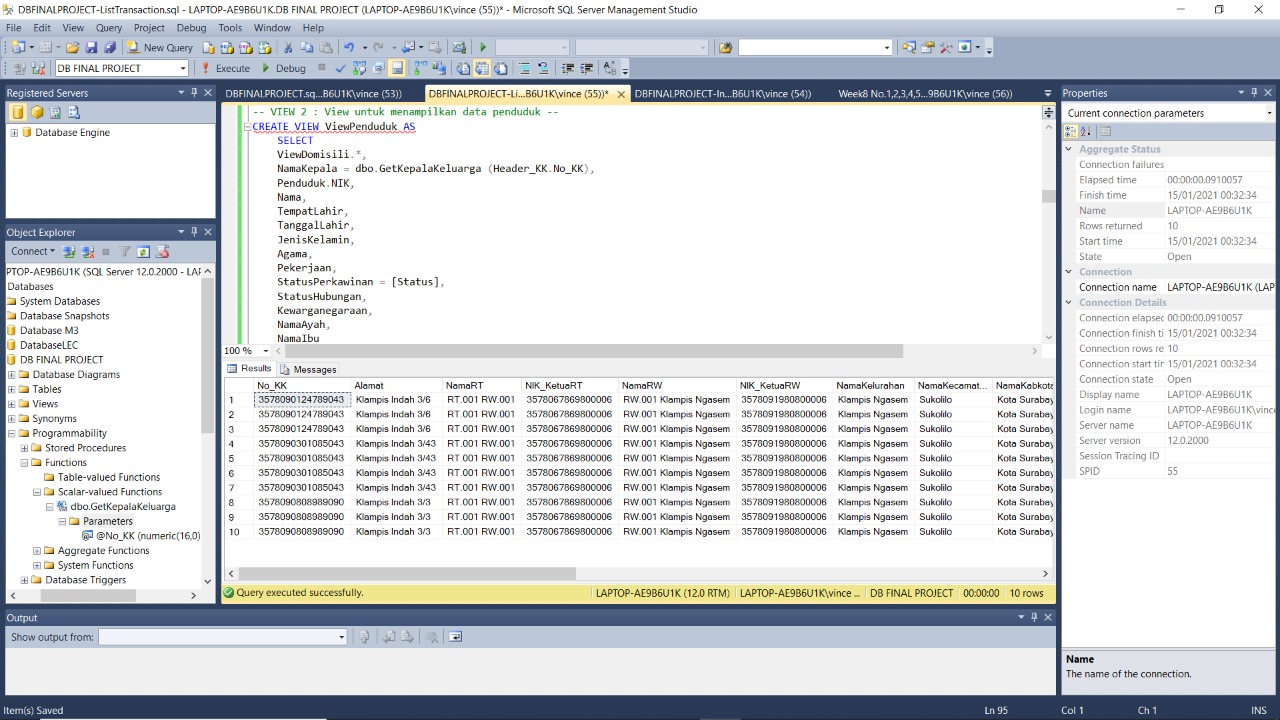
****

*Gambar 4.8 Analyze Transaction*

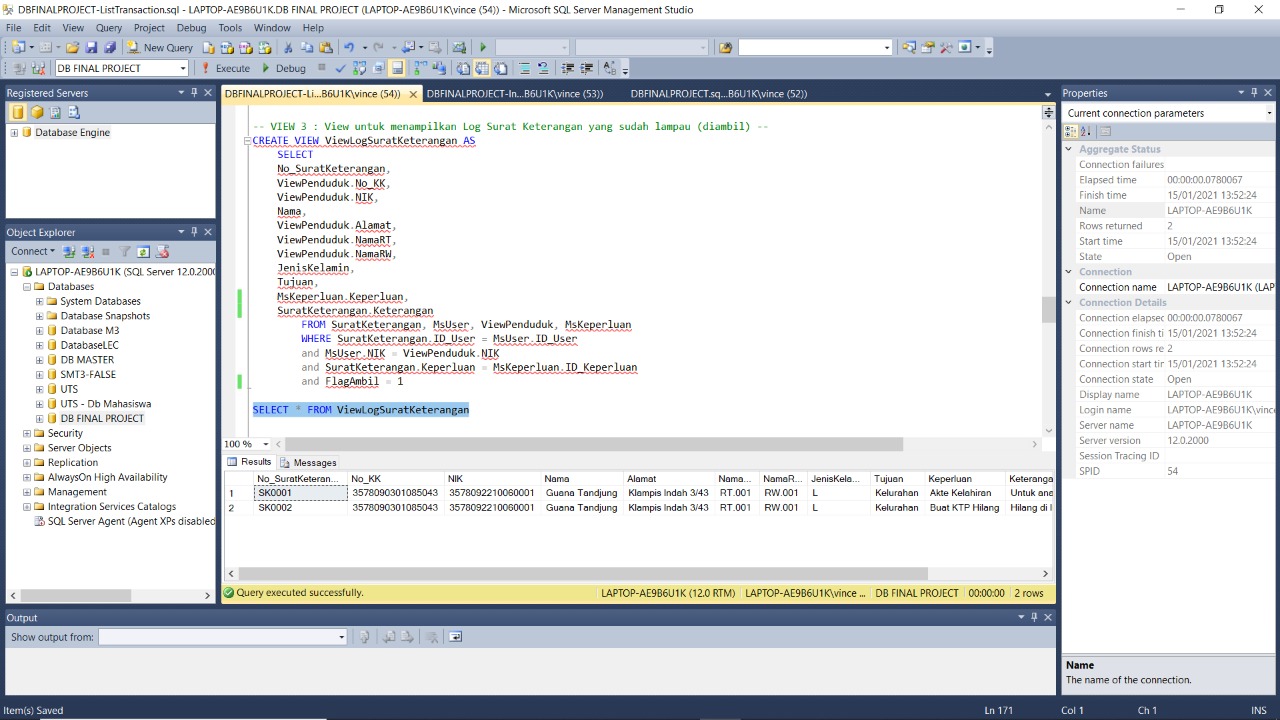
**4.3.3 User View, Procedure, and Function**



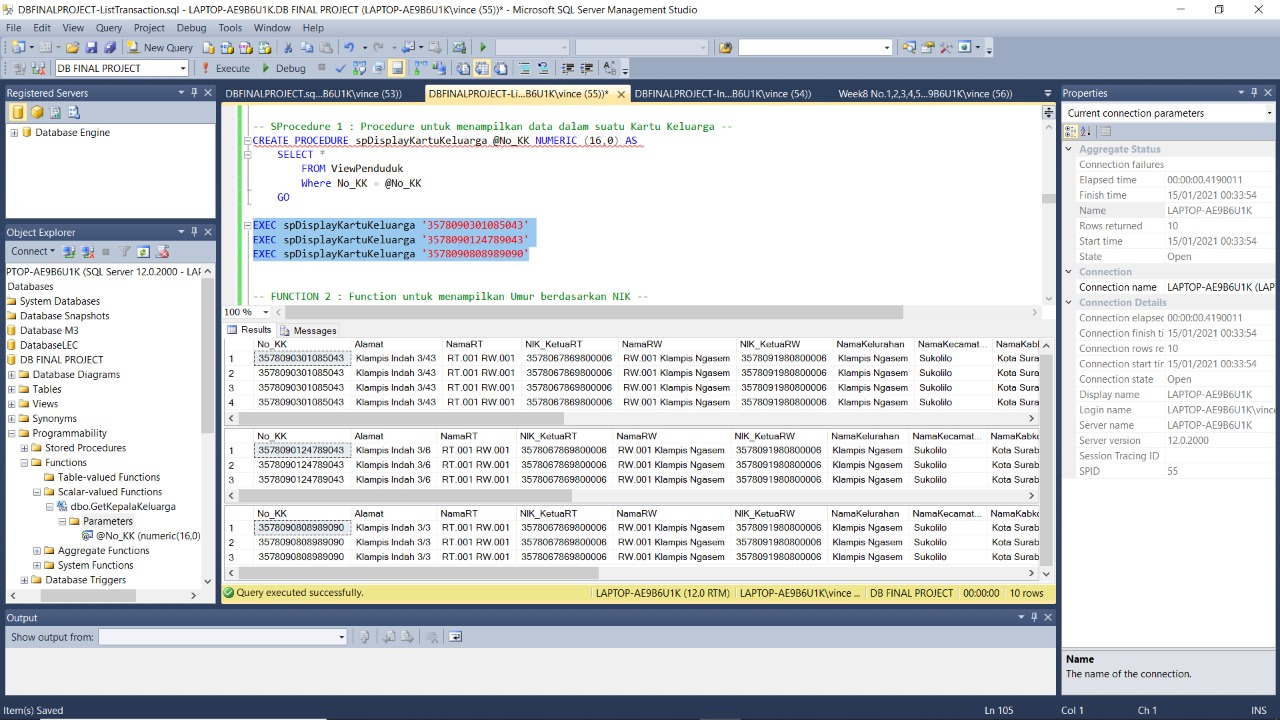
Gambar 4.9 View 1



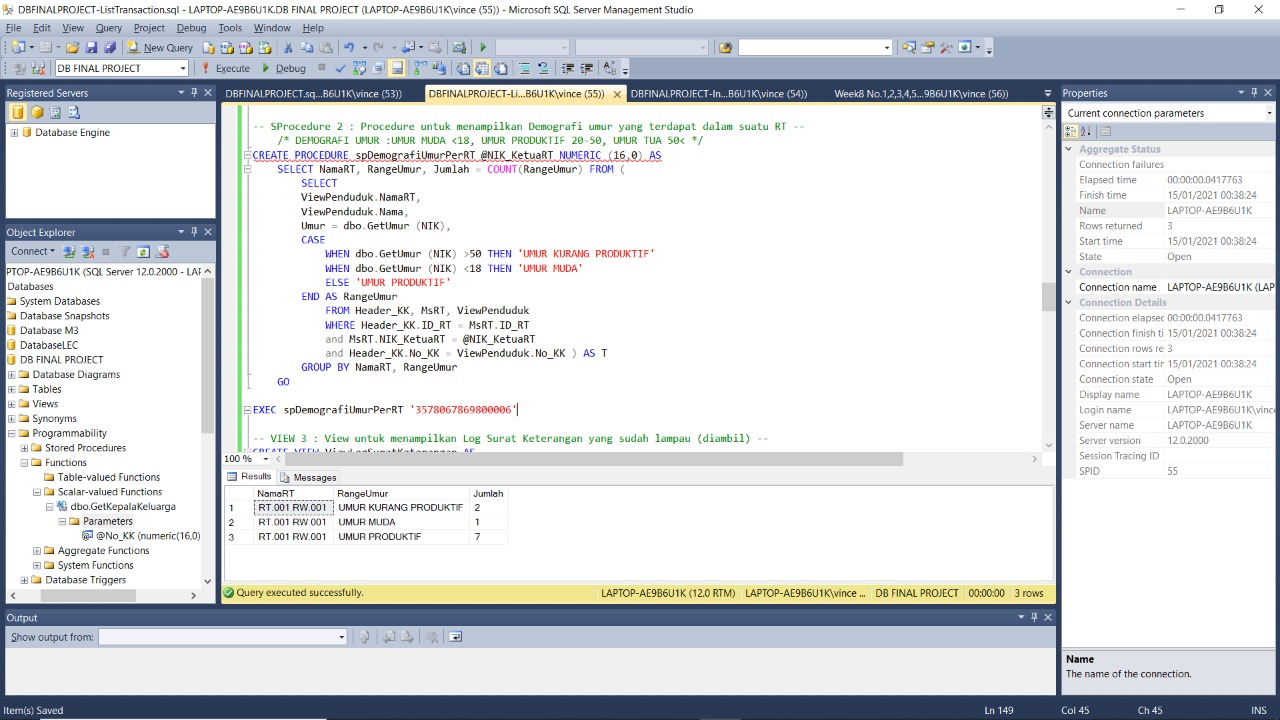
Gambar 4.10 View 2



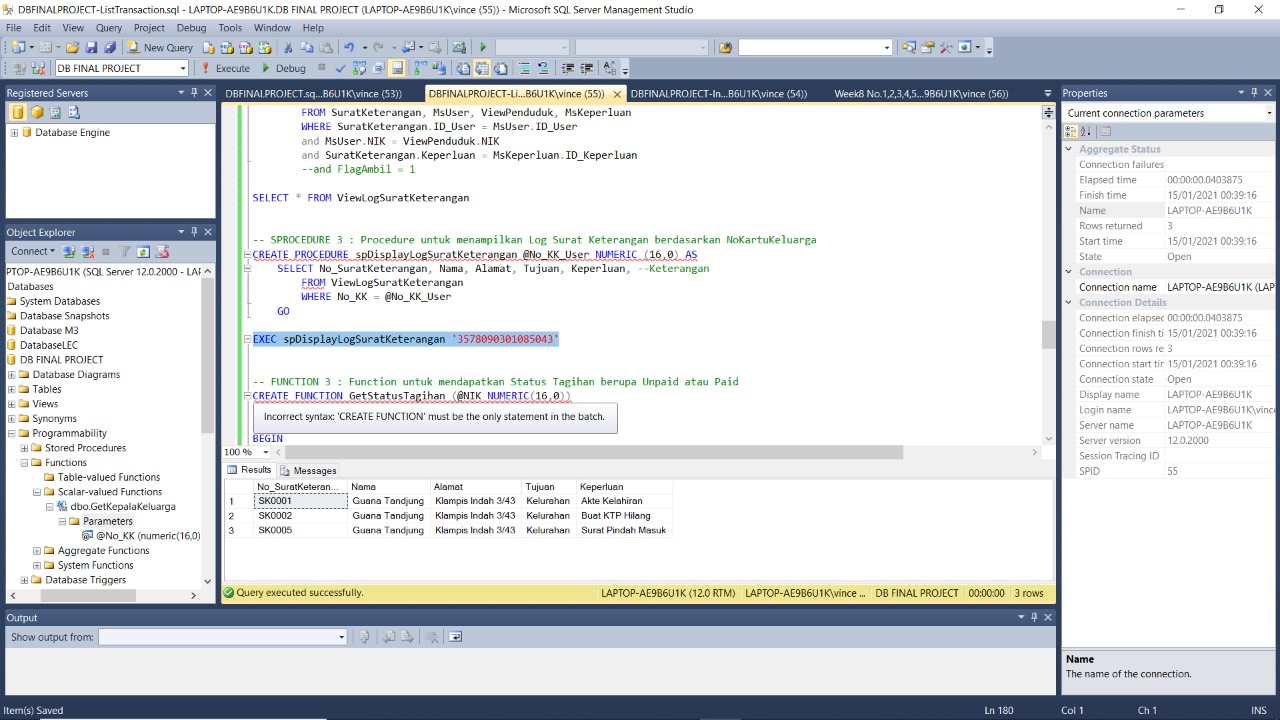
Gambar 4.11 View 3



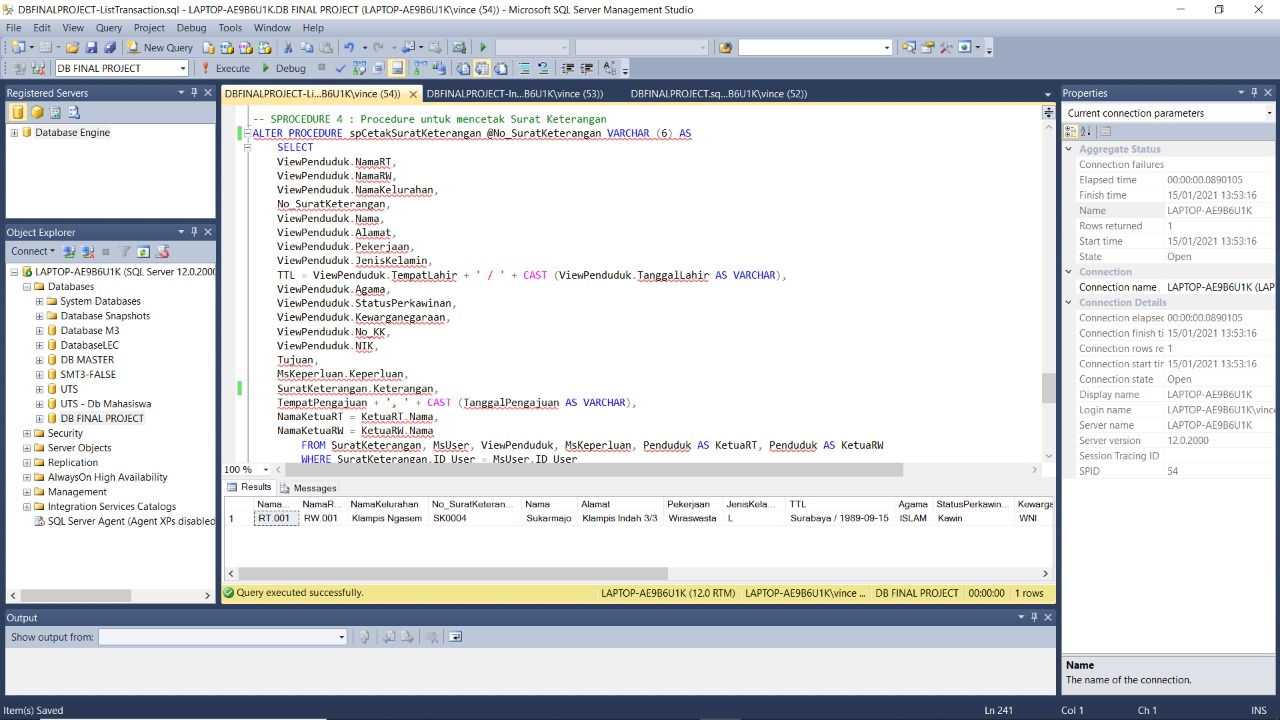
Gambar 4.12 Procedure 1



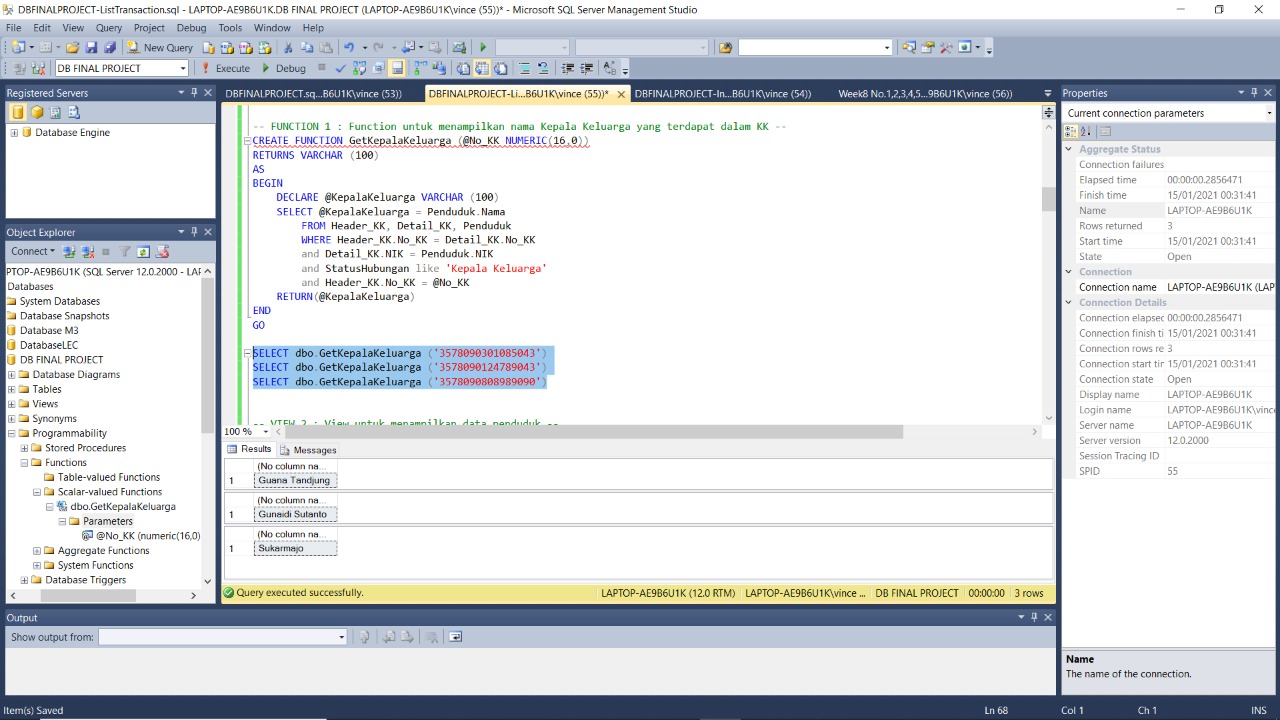
Gambar 4.13 Procedure 2



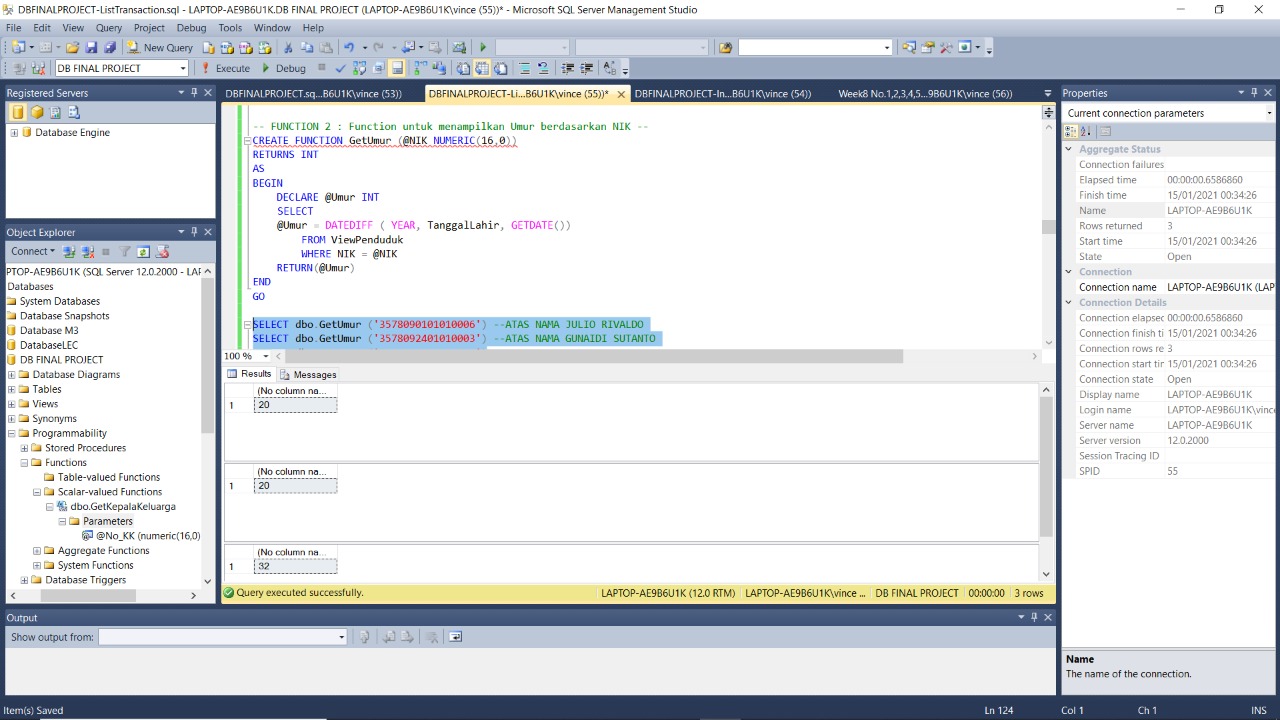
Gambar 4.14 Procedure 3



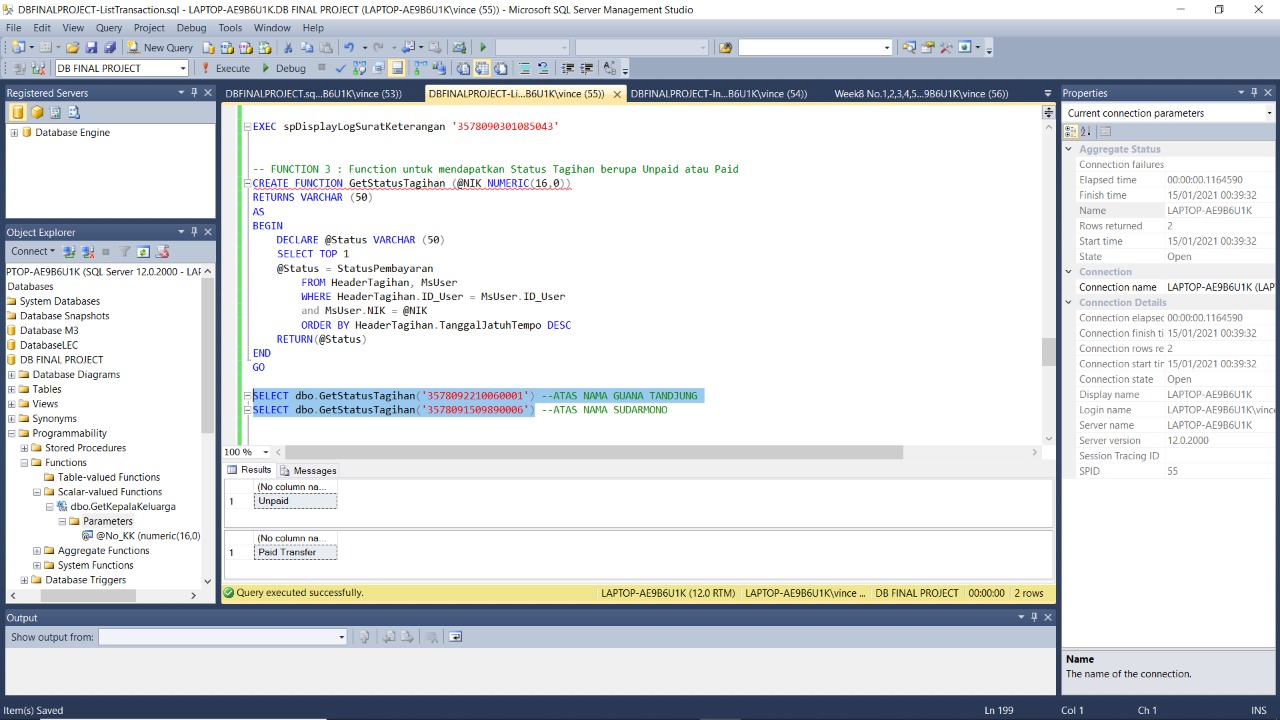
Gambar 4.15 Procedure 4



Gambar 4.16 Function 1



Gambar 4.17 Function 2



Gambar 4.18 Function 3

**DAFTAR PUSTAKA**

[1] bulelengkab.go.id. (2019). *Pengertian Gadget dan Apa itu Gadget*. [online] Available at: <https://www.bulelengkab.go.id/detail/artikel/pengertian-gadget-dan-apa-itu-gadget-77> [Accessed 3 Jan 2021].

[2] Examiner, P., & Gaffin, J. (2007). *(12) United States Patent*. *2*(12). [Accessed 3 Jan 2021].

[3] Heckler, P. E. M. (1989). *United States Patent (19)*. *19*. [Accessed 3 Jan 2020].

[4] kependudukan.denpasarkota.go.id. (2014). *Administrasi Kependudukan*. [online] Available at: <https://kependudukan.denpasarkota.go.id/artikel/read/6387> [Accessed 4 Jan. 2021].

[5] Andaru, A. (2018). *Pengertian database secara umum*. [Accessed 10 Jan 2021].

[6] Indonesia, P. R., & Rakyat, D. P. (2013). UU Nomor 24 Tahun 2013 Administrasi Kependudukan. *UU Nomor 24 Tahun 2013 Administrasi Kependudukan*, 43. [Accessed 10 Jan 2021].

[7] Pratama, A. (2006). *Pengenalan Database*. *1*, 1–9. [Accessed 10 Jan 2021].

[8] Utara, U. S. (2001). *BAB 2 LANDASAN TEORI 2.1 Pengertian Komputer Istilah komputer*. [Accessed 10 Jan 2021].

[9] Charzon. (2018). *Pengertian Dan Fungsi Database Untuk Program Delphi.* [Accessed 11 Jan 2021].

[10] kependudukan.denpasarkota.go.id. (2014). *Administrasi Kependudukan*. [online] Available at: <https://kependudukan.denpasarkota.go.id/artikel/read/6387> [Accessed 11 Jan. 2021].

**LAMPIRAN**

***Syntax SQL***

CREATE TABLE Penduduk

(

NIK NUMERIC (16,0) NOT NULL,

Nama VARCHAR (50) NOT NULL,

TempatLahir VARCHAR (15) NOT NULL,

TanggalLahir DATE NOT NULL,

JenisKelamin VARCHAR (3) NOT NULL CHECK ( JenisKelamin = 'L' or JenisKelamin = 'P'),

Alamat VARCHAR (100) NOT NULL,

Agama VARCHAR (15) NOT NULL CHECK ( Agama = 'ISLAM' or Agama = 'KATOLIK' or Agama = 'KRISTEN' or Agama = 'BUDDHA' or Agama = 'HINDU' or Agama = 'KONGHUCU' ),

[Status] VARCHAR (20) NOT NULL CHECK ( [Status] = 'Kawin' or [Status] = 'Belum Kawin' or [Status] = 'Cerai Mati' or [Status] = 'Cerai Hidup'),

Pekerjaan VARCHAR (25),

GolonganDarah VARCHAR (4) NOT NULL,

StatusKTP VARCHAR (10) CHECK ( StatusKTP like 'punya' or StatusKTP like 'belum' ),

FotoKTP VARCHAR (50),

PRIMARY KEY (NIK)

)

CREATE TABLE MsProvinsi

(

ID\_Provinsi NUMERIC (5,0) NOT NULL,

NamaProvinsi VARCHAR (25) NOT NULL,

PRIMARY KEY (ID\_Provinsi)

)

CREATE TABLE MsKabKota

(

ID\_KabKota NUMERIC (5,0) NOT NULL,

NamaKabKota VARCHAR (25) NOT NULL,

ID\_Provinsi NUMERIC (5,0) NOT NULL,

PRIMARY KEY (ID\_KabKota),

FOREIGN KEY (ID\_Provinsi) REFERENCES MsProvinsi (ID\_Provinsi) ON UPDATE CASCADE ON DELETE CASCADE

)

CREATE TABLE MsKecamatan

(

ID\_Kecamatan NUMERIC (5,0) NOT NULL,

NamaKecamatan VARCHAR (25) NOT NULL,

ID\_KabKota NUMERIC (5,0) NOT NULL,

PRIMARY KEY (ID\_Kecamatan),

FOREIGN KEY (ID\_KabKota) REFERENCES MsKabKota (ID\_KabKota) ON UPDATE CASCADE ON DELETE CASCADE

)

CREATE TABLE MsKelurahan

(

KodePos NUMERIC (10,0) NOT NULL,

NamaKelurahan VARCHAR (20) NOT NULL,

ID\_Kecamatan NUMERIC (5,0) NOT NULL,

PRIMARY KEY (KodePos),

FOREIGN KEY (ID\_Kecamatan) REFERENCES MsKecamatan (ID\_Kecamatan) ON UPDATE CASCADE ON DELETE CASCADE

)

CREATE TABLE MsRW

(

ID\_RW NUMERIC (10,0) NOT NULL,

NamaRW VARCHAR (50) NOT NULL,

Keterangan VARCHAR (100),

NIK\_KetuaRW NUMERIC (16,0) NOT NULL,

KodePos NUMERIC (10,0) NOT NULL,

PRIMARY KEY (ID\_RW),

FOREIGN KEY (NIK\_KetuaRW) REFERENCES Penduduk (NIK) ON UPDATE NO ACTION,

FOREIGN KEY (KodePos) REFERENCES MsKelurahan (KodePos) ON UPDATE CASCADE ON DELETE CASCADE

)

CREATE TABLE MsRT

(

ID\_RT NUMERIC (10,0) NOT NULL,

NamaRT VARCHAR (50) NOT NULL,

Keterangan VARCHAR (100),

NIK\_KetuaRT NUMERIC (16,0) NOT NULL,

ID\_RW NUMERIC (10,0) NOT NULL,

PRIMARY KEY (ID\_RT),

FOREIGN KEY (NIK\_KetuaRT) REFERENCES Penduduk (NIK) ON UPDATE NO ACTION,

FOREIGN KEY (ID\_RW) REFERENCES MsRW (ID\_RW) ON UPDATE CASCADE ON DELETE CASCADE

)

CREATE TABLE Header\_KK

(

No\_KK NUMERIC (16,0) NOT NULL,

Alamat VARCHAR (100) NOT NULL,

NoRumah NUMERIC (3,0) NOT NULL,

BlokRumah VARCHAR (5) NOT NULL,

ID\_RT NUMERIC (10,0) NOT NULL,

ID\_RW NUMERIC (10,0) NOT NULL,

KodePos NUMERIC (10,0) NOT NULL,

ID\_Kecamatan NUMERIC (5,0) NOT NULL,

ID\_KabKota NUMERIC (5,0) NOT NULL,

ID\_Provinsi NUMERIC (5,0) NOT NULL,

TanggalPengeluaran DATE NOT NULL,

FotoKK VARCHAR (50),

PRIMARY KEY (No\_KK),

FOREIGN KEY (ID\_RT) REFERENCES MsRT (ID\_RT) ON UPDATE NO ACTION,

FOREIGN KEY (ID\_RW) REFERENCES MsRW (ID\_RW) ON UPDATE NO ACTION,

FOREIGN KEY (KodePos) REFERENCES MsKelurahan (KodePos) ON UPDATE NO ACTION,

FOREIGN KEY (ID\_Kecamatan) REFERENCES MsKecamatan (ID\_Kecamatan) ON UPDATE NO ACTION,

FOREIGN KEY (ID\_KabKota) REFERENCES MsKabKota (ID\_KabKota) ON UPDATE NO ACTION,

FOREIGN KEY (ID\_Provinsi) REFERENCES MsProvinsi (ID\_Provinsi) ON UPDATE NO ACTION

)

CREATE TABLE Detail\_KK

(

No\_KK NUMERIC (16,0) NOT NULL,

NIK NUMERIC (16,0) NOT NULL,

StatusHubungan VARCHAR (50) NOT NULL,

Kewarganegaraan VARCHAR (10) NOT NULL,

No\_Paspor CHAR (8),

No\_KITAS CHAR (11),

NamaAyah VARCHAR (25) NOT NULL,

NamaIbu VARCHAR (25) NOT NULL,

PRIMARY KEY (No\_KK, NIK),

FOREIGN KEY (No\_KK) REFERENCES Header\_KK (No\_KK) ON UPDATE CASCADE ON DELETE CASCADE,

FOREIGN KEY (NIK) REFERENCES Penduduk (NIK) ON UPDATE NO ACTION

)

CREATE TABLE MsUser

(

ID\_User CHAR (6) NOT NULL CHECK (ID\_User like 'US[0-9][0-9][0-9][0-9]' ),

No\_KK NUMERIC (16,0) NOT NULL,

NIK NUMERIC (16,0) NOT NULL,

NamaDepan VARCHAR (25) NOT NULL,

NamaBelakang VARCHAR (25),

Alamat VARCHAR (100) NOT NULL,

No\_Hp VARCHAR (13) NOT NULL,

Email VARCHAR (50) NOT NULL,

Username VARCHAR (12) NOT NULL,

[Password] VARCHAR (10) NOT NULL,

TanggalBuatAkun DATE NOT NULL,

PRIMARY KEY (ID\_User),

FOREIGN KEY (No\_KK) REFERENCES Header\_KK (No\_KK) ON UPDATE NO ACTION,

FOREIGN KEY (NIK) REFERENCES Penduduk (NIK) ON UPDATE CASCADE ON DELETE CASCADE

)

CREATE TABLE MsKeperluan

(

ID\_Keperluan CHAR (6) NOT NULL CHECK ( ID\_Keperluan like 'KP[0-9][0-9][0-9][0-9]'),

Keperluan VARCHAR (35) NOT NULL,

KetKeperluan VARCHAR (100),

KetLampiran VARCHAR (200),

PRIMARY KEY (ID\_Keperluan)

)

CREATE TABLE SuratKeterangan

(

No\_SuratKeterangan CHAR (6) NOT NULL CHECK (No\_SuratKeterangan like 'SK[0-9][0-9][0-9][0-9]'),

ID\_User CHAR (6) NOT NULL,

Tujuan VARCHAR (25) NOT NULL,

Keperluan CHAR (6) NOT NULL,

Keterangan VARCHAR (150),

TempatPengajuan VARCHAR (15),

TanggalPengajuan DATE NOT NULL,

FlagRTBaca TINYINT,

FlagRTSetuju TINYINT,

FlagRWBaca TINYINT,

FlagRWSetuju TINYINT,

FlagCetakSuket TINYINT,

FlagDapatDiambil TINYINT,

FlagAmbil TINYINT,

PRIMARY KEY (No\_SuratKeterangan),

FOREIGN KEY (ID\_User) REFERENCES MsUser (ID\_User) ON UPDATE CASCADE ON DELETE CASCADE,

FOREIGN KEY (Keperluan) REFERENCES MsKeperluan (ID\_Keperluan) ON UPDATE CASCADE ON DELETE CASCADE

)

CREATE TABLE DetailLampiran

(

ID\_Lampiran CHAR (6) NOT NULL CHECK (ID\_Lampiran like 'LP[0-9][0-9][0-9][0-9]'),

NamaLampiran VARCHAR (30) NOT NULL,

FileUpload VARCHAR (100) NOT NULL,

StatusUpload VARCHAR (20) NOT NULL,

No\_SuratKeterangan CHAR (6) NOT NULL,

StatusVerifikasiRT VARCHAR (20),

TanggalJamVerifikasiRT DATE,

StatusVerifikasiRW VARCHAR (20),

TanggalJamVerifikasiRW DATE,

PRIMARY KEY (ID\_Lampiran),

FOREIGN KEY (No\_SuratKeterangan) REFERENCES SuratKeterangan (No\_SuratKeterangan) ON UPDATE CASCADE ON DELETE CASCADE

)

CREATE TABLE Pelaporan

(

ID\_Pelaporan CHAR (6) NOT NULL CHECK ( ID\_Pelaporan like 'PP[0-9][0-9][0-9][0-9]' ),

ID\_Pengguna CHAR (6) NOT NULL,

Tujuan VARCHAR (25),

Laporan VARCHAR (100) NOT NULL,

TanggalLaporan DATE NOT NULL,

VerifikasiRT VARCHAR (20),

PRIMARY KEY (ID\_Pelaporan),

FOREIGN KEY (ID\_Pengguna) REFERENCES MsUser (ID\_User) ON UPDATE CASCADE ON DELETE CASCADE

)

CREATE TABLE JenisTagihan

(

ID\_JenisTagihan CHAR (6) NOT NULL CHECK (ID\_JenisTagihan like 'JT[0-9][0-9][0-9][0-9]'),

JenisTagihan VARCHAR (50) NOT NULL,

Price INT NOT NULL,

PRIMARY KEY (ID\_JenisTagihan)

)

CREATE TABLE VirtualAccount

(

VA\_ID NUMERIC (14,0) NOT NULL CHECK ( VA\_ID like '3001[0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]' ),

No\_KK NUMERIC (16,0) NOT NULL,

PRIMARY KEY (VA\_ID),

FOREIGN KEY (No\_KK) REFERENCES Header\_KK (No\_KK) ON UPDATE CASCADE ON DELETE CASCADE,

)

CREATE TABLE HeaderTagihan

(

ID\_Tagihan CHAR (6) NOT NULL CHECK ( ID\_Tagihan like 'TA[0-9][0-9][0-9][0-9]' ),

ID\_User CHAR (6) NOT NULL,

TanggalJatuhTempo DATE NOT NULL,

TotalTagihan INT,

StatusPembayaran VARCHAR (50) NOT NULL CHECK ( StatusPembayaran = 'Unpaid' or StatusPembayaran = 'Paid Cash' or StatusPembayaran = 'Paid Transfer'),

TanggalPembayaran DATE,

NamaBank VARCHAR (20),

Rekening VARCHAR (20)

PRIMARY KEY (ID\_Tagihan),

FOREIGN KEY (ID\_User) REFERENCES MsUser (ID\_User) ON UPDATE CASCADE ON DELETE CASCADE,

)

CREATE TABLE DetailTagihan

(

ID\_Tagihan CHAR (6) NOT NULL,

ID\_JenisTagihan CHAR (6) NOT NULL,

SubTotalTagihan INT NOT NULL,

PRIMARY KEY (ID\_Tagihan, ID\_JenisTagihan),

FOREIGN KEY (ID\_Tagihan) REFERENCES HeaderTagihan (ID\_Tagihan) ON UPDATE CASCADE ON DELETE CASCADE,

FOREIGN KEY (ID\_JenisTagihan) REFERENCES JenisTagihan (ID\_JenisTagihan) ON UPDATE CASCADE ON DELETE CASCADE

)

-- INSERT u/ TABLE Penduduk

INSERT INTO Penduduk (NIK, Nama, TempatLahir, TanggalLahir, JenisKelamin, Alamat, Agama, [Status], Pekerjaan, GolonganDarah, StatusKTP, FotoKTP)

VALUES

('3578092401010003', 'Gunaidi Sutanto', 'Surabaya', '2001/01/24', 'L', 'Klampis Indah 3/6 C-2', 'KATOLIK', 'Kawin', 'Pelajar/Mahasiswa', 'A', 'punya', 'ktp1.jpg'),

('3578092301010001', 'Christine', 'Surabaya', '2001/01/23', 'P', 'Klampis Indah 3/16', 'KRISTEN', 'Belum Kawin', 'Pelajar/Mahasiswa', 'A', 'punya', 'ktp2.jpg'),

('3578092205990002', 'Novita Gunawan', 'Surabaya', '1999/05/22', 'P', 'Klampis Indah 4/8', 'KATOLIK', 'Belum Kawin', 'Pegawai Swasta', 'O', 'punya', 'ktp3.jpg'),

('3578090101010006', 'Julio Rivaldo', 'Surabaya', '2001/01/01', 'L', 'Klampis Indah 3/6 C-2', 'KATOLIK', 'Belum Kawin', 'Pelajar/Mahasiswa', 'B', 'punya', 'ktp4.jpg'),

('3671092305010002', 'Catherine Wijaya', 'Surabaya', '2001/05/23', 'P', 'Klampis Indah 3/6 C-2', 'KRISTEN', 'Kawin', 'Pelajar/Mahasiswa', 'A', 'punya', 'ktp5.jpg'),

('3578091409800006', 'Suparti', 'Surabaya', '1980/09/14', 'P', 'Klampis Indah 3/3 C-8', 'ISLAM', 'Kawin', 'Pegawai Negeri', 'O', 'punya', 'ktp6.jpg'),

('3578091509890006', 'Sukarmajo', 'Surabaya', '1989/09/15', 'L', 'Klampis Indah 3/3 C-8', 'ISLAM', 'Kawin', 'Wiraswasta', 'A', 'punya', 'ktp7.jpg'),

('3578091003040006', 'Bedjo', 'Surabaya', '2004/03/10', 'L', 'Klampis Indah 3/3 C-8', 'ISLAM', 'Belum Kawin', 'Pelajar/Mahasiswa', 'B', 'belum', NULL),

('3578093543450006', 'Agus Kuyy', 'Surabaya', '1998/07/12', 'L', 'Klampis Indah 4/1', 'ISLAM', 'Belum Kawin', 'Pegawai Negeri', 'AB', 'punya', 'ktp8.jpg'),

('3578094567650006', 'Suparti hehe', 'Surabaya', '1999/11/28', 'P', 'Klampis Indah 5/6', 'BUDDHA', 'Kawin', 'Pelajar/Mahasiswa', 'A', 'punya', 'ktp9.jpg'),

('3578091345200006', 'Su Su', 'Surabaya', '1996/06/16', 'P', 'Klampis Indah 12/3', 'HINDU', 'Kawin', 'Pegawai Negeri', 'B', 'punya', 'ktp10.jpg'),

('3578067869800006', 'Enrico SA', 'Surabaya', '1984/04/18', 'L', 'Klampis Indah 5/1', 'KONGHUCU', 'Cerai Mati', 'Pelajar/Mahasiswa', 'O', 'punya', 'ktp11.jpg'),

('3578091980800006', 'Agung Her', 'Surabaya', '1988/05/17', 'L', 'Klampis Indah 6/3', 'KRISTEN', 'Cerai Hidup', 'Pegawai Negeri', 'B', 'punya', 'ktp12.jpg'),

('3578092344440006', 'Dummy1', 'Surabaya', '1999/06/16', 'P', 'Klampis Indah 12/3', 'HINDU', 'Kawin', 'Pegawai Negeri', 'B', 'punya', 'ktp13.jpg'),

('3578067345340006', 'Dummy2', 'Surabaya', '1954/04/18', 'L', 'Klampis Indah 5/1', 'KONGHUCU', 'Kawin', 'Pelajar/Mahasiswa', 'O', 'punya', 'ktp14.jpg'),

('3578091564050006', 'Dummy3', 'Surabaya', '1967/05/17', 'L', 'Klampis Indah 6/3', 'KRISTEN', 'Kawin', 'Pegawai Negeri', 'B', 'punya', 'ktp15.jpg'),

('3578092345675004', 'Dummy4', 'Surabaya', '1999/06/16', 'P', 'Klampis Indah 9/3', 'HINDU', 'Kawin', 'Pegawai Negeri', 'B', 'punya', 'ktp16.jpg'),

('3578067345670002', 'Dummy5', 'Surabaya', '1954/04/18', 'L', 'Klampis ASRI 5/1', 'KONGHUCU', 'Kawin', 'Pelajar/Mahasiswa', 'O', 'punya', 'ktp17.jpg'),

('3578091564050001', 'Dummy6', 'Surabaya', '1967/05/17', 'L', 'Klampis ASRI 6/3', 'KRISTEN', 'Kawin', 'Pegawai Negeri', 'B', 'punya', 'ktp18.jpg'),

('3578092210060001', 'Guana Tandjung', 'Surabaya', '1960/10/22', 'L', 'Klampis Indah 3/43', 'KRISTEN', 'Kawin', 'Pegawai Negeri', 'B', 'punya', 'ktp19.jpg'),

('3578097110600001', 'Indira Suradja', 'Surabaya', '1960/10/31', 'P', 'Klampis Indah 3/43', 'KRISTEN', 'Kawin', 'Pegawai Negeri', 'B', 'punya', 'ktp20.jpg'),

('3578092210870003', 'Issac Tandjung', 'Surabaya', '1987/10/22', 'L', 'Klampis Indah 3/43', 'KRISTEN', 'Belum Kawin', 'Pelajar/Mahasiswa', 'O', 'punya', 'ktp21.jpg'),

('3578090804980004', 'Jeremy Tandjung', 'Surabaya', '1998/04/08', 'L', 'Klampis Indah 3/43', 'KRISTEN', 'Belum Kawin', 'Pelajar/Mahasiswa', 'B', 'punya', 'ktp22.jpg')

-- INSERT u/ TABLE MsProvinsi

INSERT INTO MsProvinsi (ID\_Provinsi, NamaProvinsi)

VALUES

('1', 'Nanggroe Aceh Darussalam'),

('2', 'Sumatera Utara'),

('3', 'Sumatera Selatan'),

('4', 'Sumatera Barat'),

('5', 'Bengkulu'),

('6', 'Riau'),

('7', 'Kepulauan Riau'),

('8', 'Jambi'),

('9', 'Lampung'),

('10', 'Bangka Belitung'),

('11', 'Kalimantan Barat'),

('12', 'Kalimantan Timur'),

('13', 'Kalimantan Selatan'),

('14', 'Kalimantan Tengah'),

('15', 'Kalimantan Utara'),

('16', 'Banten'),

('17', 'DKI Jakarta'),

('18', 'Jawa Barat'),

('19', 'Jawa Tengah'),

('20', 'DI Yogyakarta'),

('21', 'Jawa Timur'),

('22', 'Bali'),

('23', 'Nusa Tenggara Timur'),

('24', 'Nusa Tenggara Barat'),

('25', 'Gorontalo'),

('26', 'Sulawesi Barat'),

('27', 'Sulawesi Tengah'),

('28', 'Sulawesi Utara'),

('29', 'Sulawesi Tenggara'),

('30', 'Sulawesi Selatan'),

('31', 'Maluku Utara'),

('32', 'Maluku'),

('33', 'Papua Barat'),

('34', 'Papua')

-- INSERT u/ TABLE MsKabKota

INSERT INTO MsKabKota (ID\_KabKota, NamaKabKota, ID\_Provinsi)

VALUES

('1', 'Kabupaten Bangkalan', '21'),

('2', 'Kabupaten Banyuwangi', '21'),

('3', 'Kabupaten Blitar', '21'),

('4', 'Kabupaten Bojonegoro', '21'),

('5', 'Kabupaten Bondowoso', '21'),

('6', 'Kabupaten Gresik', '21'),

('7', 'Kabupaten Jember', '21'),

('8', 'Kabupaten Kediri', '21'),

('9', 'Kabupaten Malang', '21'),

('10', 'Kabupaten Pasuruan', '21'),

('11', 'Kabupaten Probolinggo', '21'),

('12', 'Kabupaten Sidoarjo', '21'),

('13', 'Kabupaten Situbondo', '21'),

('14', 'Kota Batu', '21'),

('15', 'Kota Blitar', '21'),

('16', 'Kota Kediri', '21'),

('17', 'Kota Madiun', '21'),

('18', 'Kota Malang', '21'),

('19', 'Kota Surabaya', '21'),

('20', 'Kota Probolinggo', '21'),

('21', 'Kota Pasuruan', '21'),

('22', 'Kota Mojokerto', '21'),

('23', 'Kabupaten Semarang', '19'),

('24', 'Kota Surakarta', '19'),

('25', 'Kota Salatiga', '19')

-- INSERT u/ TABLE MsKecamatan

INSERT INTO MsKecamatan (ID\_Kecamatan, NamaKecamatan, ID\_KabKota)

VALUES

('1', 'Asemrowo', '19'),

('2', 'Benowo', '19'),

('3', 'Bubutan', '19'),

('4', 'Bulak', '19'),

('5', 'Dukuh Pakis', '19'),

('6', 'Gayungan', '19'),

('7', 'Genteng', '19'),

('8', 'Gubeng', '19'),

('9', 'Gunung Anyar', '19'),

('10', 'Jambangan', '19'),

('11', 'Sukolilo', '19'),

('12', 'Tegalsari', '19'),

('13', 'Tenggilis Mejoyo', '19'),

('14', 'Wiyung', '19'),

('15', 'Wonokromo', '19')

-- INSERT u/ TABLE MsKelurahan

INSERT INTO MsKelurahan (KodePos, NamaKelurahan, ID\_Kecamatan)

VALUES

('60111', 'Keputih', '11'),

('60117', 'Klampis Ngasem', '11'),

('60118', 'Menur Pumpungan', '11'),

('60119', 'Semolowaru', '11'),

('60294', 'Gunung Anyar', '9'),

('60293', 'Rungkut Menanggal', '9')

-- INSERT u/ TABLE MsRW

INSERT INTO MsRW (ID\_RW, NamaRW, Keterangan, NIK\_KetuaRW, KodePos)

VALUES

('1', 'RW.001', 'Klampis Ngasem Sukolilo', '3578091980800006', '60117'),

('2', 'RW.002', 'Klampis Ngasem Sukolilo', '3578092344440006', '60117'),

('3', 'RW.003', 'Klampis Ngasem Sukolilo', '3578067345340006', '60117'),

('4', 'RW.004', 'Klampis Ngasem Sukolilo', '3578091564050006', '60117')

-- INSERT u/ TABLE MsRT

INSERT INTO MsRT (ID\_RT, NamaRT, Keterangan, NIK\_KetuaRT, ID\_RW)

VALUES

('1', 'RT.001', 'Klampis Indah III RW.001', '3578067869800006', '1'),

('2', 'RT.002', 'Klampis Indah V RW.001', '3578091345200006', '1'),

('3', 'RT.003', 'Klampis Indah VII RW.001', '3578091564050001', '1'),

('4', 'RT.004', 'Klampis Indah IX RW.001', '3578092345675004', '1'),

('5', 'RT.001', 'Klampis Asri I RW.002', '3578091564050001', '2'),

('6', 'RT.002', 'Klampis Asri II RW.002', '3578067345670002', '2')

-- INSERT u/ TABLE Header\_KK

INSERT INTO Header\_KK (No\_KK, Alamat, NoRumah, BlokRumah, ID\_RT, ID\_RW, KodePos, ID\_Kecamatan, ID\_KabKota, ID\_Provinsi, TanggalPengeluaran, FotoKK)

VALUES

('3578090301085043', 'Klampis Indah 3/43', '43', '3', '1', '1', '60117', '11', '19', '21', '2013/11/08', 'FotoKK1.jpg'),

('3578090124789043', 'Klampis Indah 3/6', '6', '3', '1', '1', '60117', '11', '19', '21', '2018/7/18', 'FotoKK2.jpg'),

('3578090808989090', 'Klampis Indah 3/3', '3', '3', '1', '1', '60117', '11', '19', '21', '2010/9/22', 'FotoKK3.jpg')

-- INSERT u/ TABLE Detail\_KK

INSERT INTO Detail\_KK (No\_KK, NIK, StatusHubungan, Kewarganegaraan, No\_Paspor, No\_KITAS, NamaAyah, NamaIbu)

VALUES

('3578090301085043', '3578092210060001', 'Kepala Keluarga', 'WNI', 'B980129', NULL, 'Leo Tandjung', 'Swandajani Tandjung'),

('3578090301085043', '3578097110600001', 'Istri', 'WNI', 'B888142', NULL, 'Didi Suradja', 'Sarah Setia'),

('3578090301085043', '3578092210870003', 'Anak', 'WNI', 'B891681', NULL, 'Guana Tandjung', 'Indira Suradja'),

('3578090301085043', '3578090804980004', 'Anak', 'WNI', NULL, NULL, 'Guana Tandjung', 'Indira Suradja'),

('3578090124789043', '3578092401010003', 'Kepala Keluarga', 'WNI', 'B456661', NULL, 'Winardi Sutanto', 'Nansi Cahyadi'),

('3578090124789043', '3671092305010002', 'Istri', 'WNI', NULL, NULL, 'Oei Wijaya', 'Niniek '),

('3578090124789043', '3578090101010006', 'Keluarga Lain', 'WNI', NULL, NULL, 'Herman Sutanto', 'Tere Lie'),

('3578090808989090', '3578091509890006', 'Kepala Keluarga', 'WNI', 'B478542', NULL, 'Maridjan', 'Marisa'),

('3578090808989090', '3578091409800006', 'Istri', 'WNI', 'B312568', NULL, 'Sudarmono', 'Sudarmini '),

('3578090808989090', '3578091003040006', 'Anak', 'WNI', 'B457568', NULL, 'Sukarajo', 'Suparti')

-- INSERT u/ TABLE MsUser

INSERT INTO MsUser (ID\_User, No\_KK, NIK, NamaDepan, NamaBelakang, Alamat, No\_HP, Email, Username, [Password], TanggalBuatAkun)

VALUES

('US0001', '3578090301085043', '3578092210060001', 'Guana', 'Tandjung', 'Klampis Indah 3/43', '081553265707', 'guana@abc.com', 'gundtan', 'gundtan123', '2020/11/09'),

('US0002', '3578090808989090', '3578091509890006', 'Sudarmono', 'Ahmad', 'Klampis Indah 3/3', '081677800097', 'sudarmono@abc.com', 'darmojoo', 'd4rm0j0', '2020/11/10')

-- INSERT u/ TABLE MsKeperluan

INSERT INTO MsKeperluan (ID\_Keperluan, Keperluan, KetKeperluan, KetLampiran)

VALUES

('KP0001', 'Akte Kelahiran', 'Pengajuan untuk membuat Akte Kelahiran Baru', 'Surat Lahir RS, KK, Buku Nikah'),

('KP0002', 'Akte Kematian', 'Surat Pengantar untuk orang mati', 'KTP, KK'),

('KP0003', 'Buat KTP Usia 17', 'Pengajuan pembuatan KTP karena berumur 17tahun', 'Akte Kelahiran, KK'),

('KP0004', 'Buat KTP Hilang', 'Pengajuan pembuatan KTP karena hilang', 'KTP lama, KK'),

('KP0005', 'Surat Pindah Keluar', 'Surat pengantar untuk keluar alamat', 'KTP, KK'),

('KP0006', 'Surat Pindah Masuk', 'Surat pengantar untuk masuk alamat', 'KTP, KK')

-- INSERT u/ TABLE SuratKeterangan

INSERT INTO SuratKeterangan (No\_SuratKeterangan, ID\_User, Tujuan, Keperluan, Keterangan, TempatPengajuan, TanggalPengajuan, FlagRTBaca, FlagRTSetuju, FlagRWBaca, FlagRWSetuju, FlagCetakSuket, FlagDapatDiambil, FlagAmbil )

VALUES

('SK0001', 'US0001', 'Kelurahan', 'KP0001', 'Untuk anak bernama Hendry Tandjung', 'Surabaya', '2019/01/01', '1', '1', '1', '1', '1', '1', '1'),

('SK0002', 'US0001', 'Kelurahan', 'KP0004', 'Hilang di luar kota', 'Surabaya', '2019/12/23', '1', '1', '1', '1', '1', '1', '1' ),

('SK0003', 'US0002', 'Kelurahan', 'KP0002', 'Untuk istri bernama blblbl', 'Surabaya', '2020/06/26', '1', '1', '1', '1', '1', '0', '0'),

('SK0004', 'US0002', 'Kelurahan', 'KP0005', NULL, 'Surabaya', '2020/08/28', '1', '1', '1', '1', '0', '0', '0' ),

('SK0005', 'US0001', 'Kelurahan', 'KP0006', NULL, 'Surabaya', '2020/08/29', '0', '0', '0', '0', '0', '0', '0' )

-- INSERT u/ TABLE DetailLampiran

INSERT INTO DetailLampiran (ID\_Lampiran, NamaLampiran, FileUpload, StatusUpload, No\_SuratKeterangan, StatusVerifikasiRT, TanggalJamVerifikasiRT, StatusVerifikasiRW, TanggalJamVerifikasiRW)

VALUES

('LP0001', 'Lampiran KK', 'KK1.jpg', 'Uploaded', 'SK0001', 'Verified', '2019/02/02', 'Verified', '2019/02/02'),

('LP0002', 'Lampiran Surat Lahir RS', 'Suratlahir1.jpg', 'Uploaded', 'SK0001', 'Verified', '2019/02/02', 'Verified', '2019/02/02'),

('LP0003', 'Lampiran Buku Nikah', 'Bukunikah1.jpg', 'Uploaded', 'SK0001', 'Verified', '2019/02/02', 'Verified', '2019/02/02'),

('LP0004', 'Lampiran KTP Lama', 'KTPlama1.jpg', 'Uploaded', 'SK0002', 'Verified', '2019/12/23', 'Verified', '2019/12/24'),

('LP0005', 'Lampiran KK', 'KK2.jpg', 'Uploaded', 'SK0001', 'Verified', '2019/02/23', 'Verified', '2019/02/24'),

('LP0006', 'Lampiran KTP', 'Bukunikah1.jpg', 'Uploaded', 'SK0001', 'Verified', '2019/02/02', 'Verified', '2019/02/02'),

('LP0007', 'Lampiran KK', 'KK3.jpg', 'Uploaded', 'SK0005', 'Unverified', '2019/02/23', 'Unverified', '2019/02/24'),

('LP0008', 'Lampiran KTP', 'KTP2.jpg', 'Uploaded', 'SK0005', NULL, NULL, NULL, NULL),

('LP0009', 'Lampiran KTP', 'KTP3.jpg', 'Uploaded', 'SK0003', 'Verified', '2020/06/28', NULL, NULL)

-- INSERT u/ TABLE Pelaporan

INSERT INTO Pelaporan (ID\_Pelaporan, ID\_Pengguna, Tujuan, Laporan, TanggalLaporan, VerifikasiRT)

VALUES

('PP0001', 'US0002', 'RT/RW', 'Meninggalkan Rumah', '2020/12/12', 'Verified'),

('PP0002', 'US0001', 'RT/RW', 'Sampah tidak diangkut', '2021/01/14', 'Verified')

-- INSERT u/ TABLE JenisTagihan

INSERT INTO JenisTagihan (ID\_JenisTagihan, JenisTagihan, Price)

VALUES

('JT0001', 'Iuran Kebersihan', '100000'),

('JT0002', 'Iuran Keamanan', '150000'),

('JT0003', 'Iuran PKK', '20000'),

('JT0004', 'Sokongan Pembetulan Jalan', '70000'),

('JT0005', 'Sokongan Gotong Royong', '100000'),

('JT0006', 'Sokongan Hari Raya', '50000')

-- INSERT u/ TABLE VirtualAccount

INSERT INTO VirtualAccount (VA\_ID, No\_KK)

VALUES

('30016780453537', '3578090301085043'),

('30016786784567', '3578090124789043'),

('30016780235834', '3578090808989090')

-- INSERT u/ TABLE HeaderTagihan

INSERT INTO HeaderTagihan (ID\_Tagihan, ID\_User, TanggalJatuhTempo, TotalTagihan, StatusPembayaran, TanggalPembayaran, NamaBank, Rekening)

VALUES

('TA0001', 'US0001', '2019/01/29', '250000', 'Paid Cash', '2019/01/03', NULL, NULL),

('TA0002', 'US0002', '2019/01/29', '250000', 'Paid Transfer', '2019/01/16', 'BCA', '520064352321'),

('TA0003', 'US0001', '2019/02/27', '270000', 'Paid Cash', '2019/02/26', NULL, NULL),

('TA0004', 'US0002', '2019/02/27', '270000', 'Paid Cash', '2019/02/23', NULL, NULL),

('TA0005', 'US0001', '2019/03/29', '250000', 'Paid Transfer', '2019/03/04', 'BCA', '720066546544'),

('TA0006', 'US0002', '2019/03/29', '250000', 'Paid Transfer', '2019/03/08', 'BCA', '520064352321'),

('TA0007', 'US0001', '2019/04/29', '350000', 'Paid Transfer', '2019/04/06', 'BCA', '720066546544'),

('TA0008', 'US0002', '2019/04/29', '350000', 'Paid Transfer', '2019/04/05', 'BCA', '520064352321'),

('TA0009', 'US0001', '2021/01/29', '350000', 'Unpaid', NULL, NULL, NULL),

('TA0010', 'US0002', '2021/01/29', '350000', 'Paid Transfer', '2021/01/10', 'BCA', '520064352321')

-- INSERT u/ TABLE DetailTagihan

INSERT INTO DetailTagihan (ID\_Tagihan, ID\_JenisTagihan, SubTotalTagihan)

VALUES

('TA0001', 'JT0001', '100000'),

('TA0001', 'JT0002', '150000'),

('TA0002', 'JT0001', '100000'),

('TA0002', 'JT0002', '150000'),

('TA0003', 'JT0001', '100000'),

('TA0003', 'JT0002', '150000'),

('TA0003', 'JT0003', '20000'),

('TA0004', 'JT0001', '100000'),

('TA0004', 'JT0002', '150000'),

('TA0004', 'JT0003', '20000'),

('TA0005', 'JT0001', '100000'),

('TA0005', 'JT0002', '150000'),

('TA0006', 'JT0001', '100000'),

('TA0006', 'JT0002', '150000'),

('TA0007', 'JT0001', '100000'),

('TA0007', 'JT0002', '150000'),

('TA0007', 'JT0005', '100000')

-- VIEW 1 : View untuk menampilkan domisili tiap KK --

CREATE VIEW ViewDomisili AS

SELECT

Header\_KK.No\_KK,

Alamat,

NamaRT,

NIK\_KetuaRT,

NamaRW,

NIK\_KetuaRW,

NamaKelurahan,

NamaKecamatan,

NamaKabkota,

NamaProvinsi

FROM Header\_KK, MsRT, MsRW, MsKelurahan, MsKecamatan, MsKabKota, MsProvinsi

WHERE Header\_KK.ID\_RT = MsRT.ID\_RT

and Header\_KK.ID\_RW = MsRW.ID\_RW

and Header\_KK.KodePos = MsKelurahan.KodePos

and Header\_KK.ID\_Kecamatan = MsKecamatan.ID\_Kecamatan

and Header\_KK.ID\_KabKota = MsKabKota.ID\_KabKota

and Header\_KK.ID\_Provinsi = MsProvinsi.ID\_Provinsi

SELECT \* FROM ViewDomisili

-- FUNCTION 1 : Function untuk menampilkan nama Kepala Keluarga yang terdapat dalam KK --

CREATE FUNCTION GetKepalaKeluarga (@No\_KK NUMERIC(16,0))

RETURNS VARCHAR (100)

AS

BEGIN

DECLARE @KepalaKeluarga VARCHAR (100)

SELECT @KepalaKeluarga = Penduduk.Nama

FROM Header\_KK, Detail\_KK, Penduduk

WHERE Header\_KK.No\_KK = Detail\_KK.No\_KK

and Detail\_KK.NIK = Penduduk.NIK

and StatusHubungan like 'Kepala Keluarga'

and Header\_KK.No\_KK = @No\_KK

RETURN(@KepalaKeluarga)

END

GO

SELECT dbo.GetKepalaKeluarga ('3578090301085043')

SELECT dbo.GetKepalaKeluarga ('3578090124789043')

SELECT dbo.GetKepalaKeluarga ('3578090808989090')

-- VIEW 2 : View untuk menampilkan data penduduk --

CREATE VIEW ViewPenduduk AS

SELECT

ViewDomisili.\*,

NamaKepala = dbo.GetKepalaKeluarga (Header\_KK.No\_KK),

Penduduk.NIK,

Nama,

TempatLahir,

TanggalLahir,

JenisKelamin,

Agama,

Pekerjaan,

StatusPerkawinan = [Status],

StatusHubungan,

Kewarganegaraan,

NamaAyah,

NamaIbu

FROM Header\_KK, Detail\_KK, Penduduk, ViewDomisili

WHERE Header\_KK.No\_KK = Detail\_KK.No\_KK

and Detail\_KK.NIK = Penduduk.NIK

and Header\_KK.No\_KK = ViewDomisili.No\_KK

SELECT \* FROM ViewPenduduk

-- SProcedure 1 : Procedure untuk menampilkan data dalam suatu Kartu Keluarga --

CREATE PROCEDURE spDisplayKartuKeluarga @No\_KK NUMERIC (16,0) AS

SELECT \*

FROM ViewPenduduk

Where No\_KK = @No\_KK

GO

EXEC spDisplayKartuKeluarga '3578090301085043'

EXEC spDisplayKartuKeluarga '3578090124789043'

EXEC spDisplayKartuKeluarga '3578090808989090'

-- FUNCTION 2 : Function untuk menampilkan Umur berdasarkan NIK --

CREATE FUNCTION GetUmur (@NIK NUMERIC(16,0))

RETURNS INT

AS

BEGIN

DECLARE @Umur INT

SELECT

@Umur = DATEDIFF ( YEAR, TanggalLahir, GETDATE())

FROM ViewPenduduk

WHERE NIK = @NIK

RETURN(@Umur)

END

GO

SELECT dbo.GetUmur ('3578090101010006') --ATAS NAMA JULIO RIVALDO

SELECT dbo.GetUmur ('3578092401010003') --ATAS NAMA GUNAIDI SUTANTO

SELECT dbo.GetUmur ('3578091509890006') --ATAS NAMA SUKARMAJO

-- SProcedure 2 : Procedure untuk menampilkan Demografi umur yang terdapat dalam suatu RT --

/\* DEMOGRAFI UMUR :UMUR MUDA <18, UMUR PRODUKTIF 20-50, UMUR TUA 50< \*/

CREATE PROCEDURE spDemografiUmurPerRT @NIK\_KetuaRT NUMERIC (16,0) AS

SELECT NamaRT, RangeUmur, Jumlah = COUNT(RangeUmur) FROM (

SELECT

ViewPenduduk.NamaRT,

ViewPenduduk.Nama,

Umur = dbo.GetUmur (NIK),

CASE

WHEN dbo.GetUmur (NIK) >50 THEN 'UMUR KURANG PRODUKTIF'

WHEN dbo.GetUmur (NIK) <18 THEN 'UMUR MUDA'

ELSE 'UMUR PRODUKTIF'

END AS RangeUmur

FROM Header\_KK, MsRT, ViewPenduduk

WHERE Header\_KK.ID\_RT = MsRT.ID\_RT

and MsRT.NIK\_KetuaRT = @NIK\_KetuaRT

and Header\_KK.No\_KK = ViewPenduduk.No\_KK ) AS T

GROUP BY NamaRT, RangeUmur

GO

EXEC spDemografiUmurPerRT '3578067869800006'

-- VIEW 3 : View untuk menampilkan Log Surat Keterangan yang sudah lampau (diambil) --

CREATE VIEW ViewLogSuratKeterangan AS

SELECT

No\_SuratKeterangan,

ViewPenduduk.No\_KK,

ViewPenduduk.NIK,

Nama,

ViewPenduduk.Alamat,

ViewPenduduk.NamaRT,

ViewPenduduk.NamaRW,

JenisKelamin,

Tujuan,

MsKeperluan.Keperluan,

SuratKeterangan.Keterangan

FROM SuratKeterangan, MsUser, ViewPenduduk, MsKeperluan

WHERE SuratKeterangan.ID\_User = MsUser.ID\_User

and MsUser.NIK = ViewPenduduk.NIK

and SuratKeterangan.Keperluan = MsKeperluan.ID\_Keperluan

and FlagAmbil = 1

SELECT \* FROM ViewLogSuratKeterangan

-- SPROCEDURE 3 : Procedure untuk menampilkan Log Surat Keterangan berdasarkan NoKartuKeluarga

CREATE PROCEDURE spDisplayLogSuratKeterangan @No\_KK\_User NUMERIC (16,0) AS

SELECT No\_SuratKeterangan, Nama, Alamat, Tujuan, Keperluan, Keterangan

FROM ViewLogSuratKeterangan

WHERE No\_KK = @No\_KK\_User

GO

EXEC spDisplayLogSuratKeterangan '3578090301085043'

-- FUNCTION 3 : Function untuk mendapatkan Status Tagihan berupa Unpaid atau Paid

CREATE FUNCTION GetStatusTagihan (@NIK NUMERIC(16,0))

RETURNS VARCHAR (50)

AS

BEGIN

DECLARE @Status VARCHAR (50)

SELECT TOP 1

@Status = StatusPembayaran

FROM HeaderTagihan, MsUser

WHERE HeaderTagihan.ID\_User = MsUser.ID\_User

and MsUser.NIK = @NIK

ORDER BY HeaderTagihan.TanggalJatuhTempo DESC

RETURN(@Status)

END

GO

SELECT dbo.GetStatusTagihan('3578092210060001') --ATAS NAMA GUANA TANDJUNG

SELECT dbo.GetStatusTagihan('3578091509890006') --ATAS NAMA SUDARMONO

-- SPROCEDURE 4 : Procedure untuk mencetak Surat Keterangan

ALTER PROCEDURE spCetakSuratKeterangan @No\_SuratKeterangan VARCHAR (6) AS

SELECT

ViewPenduduk.NamaRT,

ViewPenduduk.NamaRW,

ViewPenduduk.NamaKelurahan,

No\_SuratKeterangan,

ViewPenduduk.Nama,

ViewPenduduk.Alamat,

ViewPenduduk.Pekerjaan,

ViewPenduduk.JenisKelamin,

TTL = ViewPenduduk.TempatLahir + ' / ' + CAST (ViewPenduduk.TanggalLahir AS VARCHAR),

ViewPenduduk.Agama,

ViewPenduduk.StatusPerkawinan,

ViewPenduduk.Kewarganegaraan,

ViewPenduduk.No\_KK,

ViewPenduduk.NIK,

Tujuan,

MsKeperluan.Keperluan,

SuratKeterangan.Keterangan,

TempatPengajuan + ', ' + CAST (TanggalPengajuan AS VARCHAR),

NamaKetuaRT = KetuaRT.Nama,

NamaKetuaRW = KetuaRW.Nama

FROM SuratKeterangan, MsUser, ViewPenduduk, MsKeperluan, Penduduk AS KetuaRT, Penduduk AS KetuaRW

WHERE SuratKeterangan.ID\_User = MsUser.ID\_User

and MsUser.NIK = ViewPenduduk.NIK

and SuratKeterangan.Keperluan = MsKeperluan.ID\_Keperluan

and ViewPenduduk.NIK\_KetuaRT = KetuaRT.NIK

and ViewPenduduk.NIK\_KetuaRW = KetuaRW.NIK

and FlagRWSetuju = 1

and FlagCetakSuket = 0

and No\_SuratKeterangan = @No\_SuratKeterangan

GO

EXEC spCetakSuratKeterangan 'SK0001' --SUDAH AMBIL

EXEC spCetakSuratKeterangan 'SK0002' --SUDAH AMBIL

EXEC spCetakSuratKeterangan 'SK0003' --SUDAH DICETAK => TIDAK DAPAT DI CETAK LAGI

EXEC spCetakSuratKeterangan 'SK0004' --BELUM DICETAK => BISA DI CETAK

EXEC spCetakSuratKeterangan 'SK0005' --BELUM DIBACA RT => BELUM DAPAT DI CETAK

**LEMBAR PENILAIAN**

**PERANCANGAN DATABASE KEPENDUDUKAN DI TINGKAT RT/RW**

**UNTUK PEMBAYARAN IURAN LINGKUNGAN DAN PENGAJUAN SURAT SURAT**

MATA KULIAH ISYS6028 – DATABASE SYSTEM

KELAS BA20 -LAB

Semester Ganjil, 2020 / 2021

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **DAFTAR MAHASISWA** | **NILAI** | | | | **BOBOT** | | | | **KREDIT** | | | | **TOTAL KREDIT** |
| 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 |
| 2301943491 - Julio Rivaldo Gunawan |  |  |  |  | 20% | 20% | 24% | 36% |  |  |  |  |  |
| 2301878952 - Putri Aurelia Shilo |  |  |  |  | 20% | 20% | 24% | 36% |  |  |  |  |  |
| 2301879583 - Vincentius Sutanto |  |  |  |  | 20% | 20% | 24% | 36% |  |  |  |  |  |
| **TOTAL** | | | | | | | | | | | | |  |

**KETERANGAN :**

* **Skala Penilaian : 0 sd 100**
* **Komponen**

1 : Laporan

2 : Produk

3 : Pengetahuan

4 : Solusi

**Malang, 15 Januari 2020**

**Wina Permana Sari**

**D5975**