**PEMBUATAN APLIKASI SURAT TUGAS DAN SURAT PERINTAH PERJALANAN DINAS MENGGUNAKAN EMBARCADERO DELPHI XE8**

**(Studi Kasus Pada Sekolah Dasar Negeri Mekarsari Kecamatan Baleendah)**

**SKRIPSI**

Karya Tulis sebagai syarat memperoleh

Gelar Sarjana Komputer dari Fakutas Teknologi Informasi

Universitas Bale Bandung

Disusun oleh :

AGUS EKA PURNAMA

NPM. C1A160045 / 311160045



PROGRAM STRATA 1

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNOLOGI INFORMASI

UNIVERSITAS BALE BANDUNG

BANDUNG

2021

# **LEMBAR PERSETUJUAN PEMBIMBING**

PEMBUATAN APLIKASI SURAT TUGAS DAN SURAT

PERINTAH PERJALANAN DINAS MENGGUNAKAN

EMBARCADERO DELPHI XE8

(Studi Kasus Pada Sekolah Dasar Negeri Mekarsari Kecamatan Baleendah)

Disusun oleh :

AGUS EKA PURNAMA

NPM. C1A160045 / 311160045

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar

**SARJANA KOMPUTER**

Pada

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS BALE BANDUNG**

Baleendah, Juli 2021

Disetujui oleh:

|  |  |
| --- | --- |
| Pembimbing Utama | Pembimbing Pendamping |

|  |  |
| --- | --- |
| Yudi Herdiana, S.T., M.T.  NIK. 04104808008 | Yaya Suharya, S.Kom., M.T.  NIK. 01043170007 |

# **LEMBAR PERSETUJUAN PENGUJI**

PEMBUATAN APLIKASI SURAT TUGAS DAN SURAT

PERINTAH PERJALANAN DINAS MENGGUNAKAN

EMBARCADERO DELPHI XE8

(Studi Kasus Pada Sekolah Dasar Negeri Mekarsari Kecamatan Baleendah)

Disusun oleh :

AGUS EKA PURNAMA

NPM. C1A160045 / 311160045

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar

**SARJANA KOMPUTER**

Pada

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS BALE BANDUNG**

Baleendah, Juli 2021

Disetujui oleh:

|  |  |
| --- | --- |
| Penguji 1 | Penguji 2 |

|  |  |
| --- | --- |
| Rosmalina, S.T., M.Kom.  NIK. 04104808122 | Yusuf Muharam, M.Kom.  NIK. 04104820003 |

# **LEMBAR PENGESAHAN PROGRAM STUDI**

PEMBUATAN APLIKASI SURAT TUGAS DAN SURAT

PERINTAH PERJALANAN DINAS MENGGUNAKAN

EMBARCADERO DELPHI XE8

(Studi Kasus Pada Sekolah Dasar Negeri Mekarsari Kecamatan Baleendah)

Disusun oleh :

AGUS EKA PURNAMA

NPM. C1A160045 / 311160045

Telah diterima dan disetujui untuk memenuhi persyaratan mencapai gelar

**SARJANA KOMPUTER**

Pada

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**UNIVERSITAS BALE BANDUNG**

Baleendah, Juli 2021

Disetujui oleh:

|  |  |
| --- | --- |
| Mengetahui,  Dekan | Mengesahkan,  Ketua Program Studi |

|  |  |
| --- | --- |
| Yudi Herdiana, ST., MT.  NIK. 04104808008 | Yusuf Muharam, M.Kom.  NIK. 04104820003 |

# **LEMBAR PERNYATAAN KEASLIAN**

Saya yang bertanda tangan dibawah ini :

Nama : AGUS EKA PURNAMA

NPM : C1A160045 / 311160045

Judul Skripsi : **PEMBUATAN APLIKASI SURAT TUGAS DAN SURAT PERINTAH PERJALANAN DINAS MENGGUNAKAN EMBARCADERO DELPHI XE8**

Menyatakan dengan sebenarnya bahwa penulisan skripsi ini berdasarkan hasil penelitian, pemikiran, dan pemaparan asli dari saya sendiri, baik untuk naskah laporan maupun kegiatan *programing* yang tercantum sebagai bagian dari skripsi ini. Jika terdapat karya orang lain, saya mencantumkan sumber yang jelas.

Pernyataan ini saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidak benaran dalam pernyataan ini, maka saya bersedia menerima sangsi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini dan sangsi lain sesuai dengan peraturan yang berlaku di FAKULTAS TEKNOLOGI INFORMASI UNIVERSITAS BALE BANDUNG.

Demikian surat pernyataan ini saya buat dalam keadaan sadar tanpa paksaan dari pihak manapun.

Baleendah, Juli 2021

Yang membuat pernyataan,

**AGUS EKA PURNAMA**

NIM. C1A160045 / 311160045

# ***ABSTRACT***

*Management of letters, especially Letters of Assignment and Official Travel Orders (SPPD) is very important in an educational institution such as a school. Assignment letter is an official document containing an assignment from an authorized official to someone to carry out an activity. While the Official Travel Order (SPPD) is a trip outside the domicile either individually or collectively.*

*SD Negeri Mekarsari is located at Jl. R.A.A Wiranatakusumah kec. Baleendah, Bandung Regency. The school still does not have a special application that helps School Operators in managing Assignment Letters and SPPD. This is because the making of the current letter is still using Ms. Word, so the process must be edited every time you go on a business trip. In addition, there are several problems in the process of storing letter archives and the process of recording school letter reports.*

*The stages of developing this application use the Waterfall model, including analysis, design, coding, testing and maintenance. This application was developed using Embarcadero Delphi XE8 and using Microsoft Access 2010 as the database. The purpose of the research is to create an application that can assist in the process of making to reporting assignments and SPPD. This application is expected to be able to help operators starting from the process of making, printing letters, storing letter data, to reporting the recap of the letter itself so that it is made faster, and avoids Human Error.*

*Keywords: letter of assignment, SPPD, Embarcadero Delphi XE8, Microsoft access 2010, Database, application, waterfall model*

# **ABSTRAK**

Pengelolaan surat terutama Surat Tugas dan Surat Perintah Perjalanan Dinas (SPPD) sangat penting dalam sebuah instansi pendidikan seperti sekolah. Surat tugas adalah naskah dinas yang berisi penugasan dari pejabat yang berwenang kepada seseorang untuk melaksanakan suatu kegiatan. Sedangkan Surat Perintah Perjalanan Dinas (SPPD) adalah perjalanan ke luar tempat kedudukan baik perseorangan maupun secara bersama.

SD Negeri Mekarsari beralamat di Jl. R.A.A Wiranatakusumah kec. Baleendah Kabupaten Bandung. Sekolah tersebut masih belum memiliki aplikasi khusus yang membantu Operator Sekolah dalam pengelolaan Surat Tugas dan SPPD. Hal ini dikarenakan pembuatan pembuatan surat saat ini masih menggunakan Ms. Word, sehingga proses tersebut harus dilakukan pengeditan setiap kali akan melakukan perjalanan dinas. Selain itu terdapat beberapa masalah dalam proses penyimpanan arsip surat dan proses perekapan laporan surat sekolah.

Tahap-tahap pengembangan aplikasi ini menggunakan model *Waterfall*, meliputi analis, desain, pengodean, pengujian dan pemerliharaan. Aplikasi ini dikembangkan dengan menggunakan Embarcadero Delphi XE8 dan menggunakan Microsoft access 2010 sebagai *database*. Tujuan penelitian adalah untuk membuat sebuah aplikasi yang dapat membantu dalam proses pembuatan sampai pelaporan surat tugas dan SPPD. Aplikasi ini diharapkan mampu membantu operator mulai dari proses pembuatan, pencetakan surat, penyimpanan data surat, sampai ke pelaporan rekap surat itu sendiri agar pembuatannya lebih cepat, dan terhindar dari *Human Error*.

Kata Kunci: surat tugas, SPPD, Embarcadero Delphi XE8, Microsoft access 2010, *Database*, aplikasi, model *waterfall*

# **KATA PENGANTAR**

Puji dan Syukur kami panjatkan kehadirat Allah S.W.T, karena berkat limpahan Rahmat dan Karunia-Nya sehingga Penulis dapat menyelesaikan skrisi ini dengan baik dan tepat pada waktunya. Dalam skipsi ini saya membahas mengenai Pembuatan Aplikasi Surat Tugas Dan Surat Perintah Perjalanan Dinas Menggunakan Embarcadero Delphi XE8

Skripsi ini dibuat dengan berbagai observasi dan beberapa bantuan dari berbagai pihak untuk membantu menyelesaikan tantangan dan hambatan selama mengerjakan skripsi ini. Oleh karena itu, Penulis mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Kedua orang tua (Agus Widodo dan Sumiyati). Keduanya adalah anugerah terindah yang Allah berikan dalam hidup penulis. Selalu memberikan support, kasih sayang, dan doa yang tiada hentinya untuk penulis serta cinta yang tak akan tergantikan di dunia ini
2. Bapak Yudi Herdiana, S.T., M.T. selaku Dekan Fakultas Teknologi Informasi Universitas Bale Bandung dan selaku pembimbing utama yang telah memberikan arahan serta bimbingannya untuk penulis, sehingga penulis dapat menyelesaikan tepat pada waktunya.
3. Bapak Yusuf Muharam, M.Kom. selaku Ketua Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Bale Bandung.
4. Bapak Yaya Suharya, S.Kom., M.T. selaku pembimbing pendamping karena berkat arahan dan bimbingannya penelitian ini bisa selesai dengan maksimal serta tepat pada waktunya.
5. Seluruh dosen Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Bale Bandung yang tidak dapat disebutkan satu per satu, terima kasih atas ilmu yang bermanfaat untuk penulis
6. Ibu Komalasari, S.Pd, M,Pd. Selaku Kepala Sekolah SDN Mekarsari yang telah memberikan ijin penelitian sehingga penulis dapat menyelesaikan penelitian nya dengan lancar dan tepat waktu.
7. Resta Agus Saputri, adik perempuan satu-satunya yang sudah mendoakan dan memberikan semangat kepada penulis.
8. Fahriza Zuraida Nurmeita, S.S, orang yang tak lelah memberikan penulis semangat, memberikan nasehat, memberikan motivasi kehidupan, memberikan cinta dan sayangnya kepada penulis serta hari-harinya yang pernah dilewatkaan bersama penulis serta dukunganya kepada penulis di saat terjatuh.
9. Rekan-rekan FTI 2016 dan 2017 yang selalu memberikan semangatnya serta teman seperjuangan yang sampai sekarang berjuang bersama.
10. Serta pihak-pihak yang namanya tidak bisa disebutkan satu per satu. Terima kasih atas doa dan dukungan nya sehingga penulis bisa menyelesaikan skripsi ini.

Penulis menyadari bahwa masih banyak kekurangan dalam skripsi ini. Oleh karena itu, segala kritik dan saran yang membangun akan Penulis terima dengan baik. Semoga skripsi ini bermanfaat bagi kita semua.

|  |  |
| --- | --- |
|  | Baleendah, Juli 2021  Penulis  AGUS EKA PURNAMA |

# **DAFTAR ISI**

[LEMBAR PERSETUJUAN PEMBIMBING i](#_Toc79753144)

[LEMBAR PERSETUJUAN PENGUJI ii](#_Toc79753145)

[LEMBAR PENGESAHAN PROGRAM STUDI iii](#_Toc79753146)

[LEMBAR PERNYATAAN KEASLIAN iv](#_Toc79753147)

[*ABSTRACT* v](#_Toc79753148)

[ABSTRAK vi](#_Toc79753149)

[KATA PENGANTAR vii](#_Toc79753150)

[DAFTAR ISI ix](#_Toc79753151)

[DAFTAR GAMBAR xii](#_Toc79753152)

[DAFTAR TABEL xiv](#_Toc79753153)

[BAB I PENDAHULUAN 1](#_Toc79753154)

[1.1. Latar Belakang 1](#_Toc79753155)

[1.2. Rumusan Masalah 3](#_Toc79753156)

[1.3. Batasan Masalah 3](#_Toc79753157)

[1.4. Tujuan Penelitian 4](#_Toc79753158)

[1.5. Metodologi 4](#_Toc79753159)

[1.5.1. Metode Pengumpulan Data 4](#_Toc79753160)

[1.5.2. Metode Pengembangan Sistem 4](#_Toc79753161)

[1.6. Sistematika Penulisan 5](#_Toc79753162)

[BAB II TINJAUAN PUSTAKA 6](#_Toc79753163)

[2.1. Landasan Teori 6](#_Toc79753164)

[2.2. Dasar Teori 7](#_Toc79753165)

[2.2.1. Pengertian Aplikasi 7](#_Toc79753166)

[2.2.2. Pengertian Database 8](#_Toc79753167)

[2.2.3. PERMENDIKBUD No 6 Tahun 2013 (Mendikbud, 2013) 9](#_Toc79753168)

[2.2.4. Lampiran PERMENDIKBUD No 6 Tahun 2013 11](#_Toc79753169)

[2.2.5. PERMENKEU No 45 Tahun 2007 (PERMENKEU, 2007) 12](#_Toc79753170)

[2.2.6. Lampiran PERMENKEU No 45 Tahun 2007 13](#_Toc79753171)

[2.2.7. Delphi 14](#_Toc79753172)

[2.2.8. Microsoft Access 2010 (Heryanto, 2017) 18](#_Toc79753173)

[2.2.9. Balsamiq Mockup 23](#_Toc79753174)

[2.2.10. Unified Modeling Language (UML) 24](#_Toc79753175)

[2.2.11. Model Waterfall 26](#_Toc79753176)

[BAB III METODOLOGI PENELITIAN 29](#_Toc79753177)

[3.1. Kerangka Pikir 29](#_Toc79753181)

[3.2. Deskripsi 30](#_Toc79753182)

[3.2.1. Investigasi Pendahuluan 30](#_Toc79753187)

[3.2.2. Analisis Masalah 31](#_Toc79753188)

[3.2.3. Perencanaan kebutuhan 32](#_Toc79753189)

[3.2.4. Desain 33](#_Toc79753190)

[3.2.5. Pembuatan Aplikasi 34](#_Toc79753191)

[3.2.6. Pengujian Aplikasi 34](#_Toc79753192)

[3.2.7. Implementasi 34](#_Toc79753193)

[3.2.8. Pemeliharaan (*maintenance)* 34](#_Toc79753194)

[BAB IV ANALISIS DAN PERANCANGAN 35](#_Toc79753195)

[4.1 Analisis 35](#_Toc79753196)

[4.1.1 Analisis Masalah 35](#_Toc79753197)

[4.1.2 Analisis *Software* 35](#_Toc79753198)

[4.1.3 Analisis Pengguna 35](#_Toc79753202)

[4.1.4 User Interface 36](#_Toc79753203)

[4.1.5 Fitur-fitur 37](#_Toc79753204)

[4.1.6 Analisis Data 37](#_Toc79753205)

[4.2 Perancangan 38](#_Toc79753206)

[4.2.1 *Use Case Diagram* 38](#_Toc79753207)

[4.2.2. *Class Diagram* 55](#_Toc79753208)

[4.2.3. *Activity Diagram* 55](#_Toc79753209)

[4.3. Kamus Data 64](#_Toc79753210)

[4.4. Desain 66](#_Toc79753214)

[BAB V IMPLEMENTASI DAN PENGUJIAN 74](#_Toc79753215)

[5.1. Implementasi Sistem 74](#_Toc79753216)

[5.1.1. Implementasi Perangkat Keras 74](#_Toc79753217)

[5.1.2. Implementasi Perangkat Lunak 74](#_Toc79753218)

[5.1.3. Implementasi Antar Muka 74](#_Toc79753219)

[5.2. Pengujian 87](#_Toc79753220)

[5.2.1. Rencana Pengujian 87](#_Toc79753221)

[5.2.2. Deskripsi dan Hasil Pengujian 92](#_Toc79753222)

[BAB VI KESIMPULAN DAN SARAN 106](#_Toc79753223)

[6.1. Kesimpulan 106](#_Toc79753224)

[6.2. Saran 107](#_Toc79753225)

[DAFTAR PUSTAKA xvi](#_Toc79753226)

[LAMPIRAN xviii](#_Toc79753227)

# **DAFTAR GAMBAR**

[Gambar 2.1 Contoh Format Surat Tugas 11](#_Toc76382251)

[Gambar 2.2 Contoh Format SPPD Halaman 1 13](#_Toc76382252)

[Gambar 2.3 Contoh Format SPPD Halaman 2 14](#_Toc76382253)

[Gambar 2.4 Ilustrasi Model Waterfall 27](#_Toc76382254)

[Gambar 3.1 Kerangka Pikir 29](#_Toc76382255)

[Gambar 4.1 Struktur Rancangan *User Interface* 36](#_Toc76382256)

[Gambar 4.2 *Use Case* Aplikasi 38](#_Toc76382257)

[Gambar 4.3 *Class Diagram* 55](#_Toc76382258)

[Gambar 4.4 *Activity Diagram* Login 56](#_Toc76382259)

[Gambar 4.5 *Activity Diagram* Ubah Data Sekolah 56](#_Toc76382260)

[Gambar 4.6 *Activity Diagram* unggah kop surat 57](#_Toc76382261)

[Gambar 4.7 *Activity Diagram* Tambah Data Pegawai 57](#_Toc76382262)

[Gambar 4.8 *Activity Diagram* Ubah Data Pegawai 58](#_Toc76382263)

[Gambar 4.9 *Activity Diagram* Hapus Data Pegawai 58](#_Toc76382264)

[Gambar 4.10 *Activity Diagram* Tambah Data Surat 59](#_Toc76382265)

[Gambar 4.11 *Activity Diagram* Ubah Data Surat 59](#_Toc76382266)

[Gambar 4.12 *Activity diagram* Hapus Data Surat 60](#_Toc76382267)

[Gambar 4.13 *Activity Diagram* Cetak Surat Tugas 60](#_Toc76382268)

[Gambar 4.14 *Activity Diagram* Cetak SPPD 1 61](#_Toc76382269)

[Gambar 4.15 *Activity Diagram* Cetak SPPD 2 61](#_Toc76382270)

[Gambar 4.16 *Activity Diagram* Cetak Laporan Data Pegawai 62](#_Toc76382271)

[Gambar 4.17 *Activity Diagram* Cetak Laporan Data Surat Berdasarkan Nama Pegawai 62](#_Toc76382272)

[Gambar 4.18 *Activity Diagram* Cetak Laporan Data Surat Berdasarkan Periode Tanggal 63](#_Toc76382273)

[Gambar 4.19 *Activity Diagram* Cetak Laporan Semua Data Surat 63](#_Toc76382274)

[Gambar 4.20 Desain Form Menu Utama 66](#_Toc76382275)

[Gambar 4.21 Desain Form Login 67](#_Toc76382276)

[Gambar 4.22 Desain Data Sekolah 67](#_Toc76382277)

[Gambar 4.23 Desain Form Data Pegawai 68](#_Toc76382278)

[Gambar 4.24 Desain Form Tambah Pegawai 69](#_Toc76382279)

[Gambar 4.25 Desain Form Ubah Data Pegawai 69](#_Toc76382280)

[Gambar 4.26 Desain Form Data Surat 70](#_Toc76382281)

[Gambar 4.27 Desain Form Cetak Surat 71](#_Toc76382282)

[Gambar 4.28 Desain Form Cetak Laporan 71](#_Toc76382283)

[Gambar 4.29 Desain Form Cetak Data Surat Berdasarkan Data Pegawai 72](#_Toc76382284)

[Gambar 4.30 Desain Form Cetak Data Surat Berdasarkan Tanggal 72](#_Toc76382285)

[Gambar 4.31 Desain Form Cetak Semua Data Surat 73](#_Toc76382286)

[Gambar 5.1 Tampilan Login 75](#_Toc76382287)

[Gambar 5.2 Tampilan Menu Utama 75](#_Toc76382288)

[Gambar 5.3 Tampilan Data Sekolah 76](#_Toc76382289)

[Gambar 5.4 Tampilan Data Pegawai 77](#_Toc76382290)

[Gambar 5.5 Tampilan Tambah Data Pegawai 77](#_Toc76382291)

[Gambar 5.6 Tampilan Ubah Data Pegawai 78](#_Toc76382292)

[Gambar 5.7 Tampilan data surat 79](#_Toc76382293)

[Gambar 5.8 Tampilan Cetak Surat 80](#_Toc76382294)

[Gambar 5.9 Hasil Cetak Surat Tugas 80](#_Toc76382295)

[Gambar 5.10 Hasil Cetak SPPD Halaman 1 81](#_Toc76382296)

[Gambar 5.11 Hasil Cetak SPPD Halaman 2 82](#_Toc76382297)

[Gambar 5.12 Tampilan Pilih Jenis Cetak Laporan 83](#_Toc76382298)

[Gambar 5.13 Hasil Cetak Laporan Rekap Data Pegawai 83](#_Toc76382299)

[Gambar 5.14 Tampilan Cetak Data Surat 84](#_Toc76382300)

[Gambar 5.15 Tampilan Cetak Berdasarkan Nama Pegawai 84](#_Toc76382301)

[Gambar 5.16 Hasil Cetak Laporan Rekap Surat Berdasarkan Nama Pegawai 85](#_Toc76382302)

[Gambar 5.17 Tampilan Cetak Berdasarkan Tanggal 85](#_Toc76382303)

[Gambar 5.18 Hasil Cetak Laporan Rekap Surat Berdasarkan Tanggal 86](#_Toc76382304)

[Gambar 5.19 Tampilan Cetak Semua 86](#_Toc76382305)

[Gambar 5.20 Hasil Cetak Laporan Rekap Semua Data 87](#_Toc76382306)

# **DAFTAR TABEL**

[Tabel 2.1 Simbol – Simbol Use Case 25](#_Toc77418006)

[Tabel 2.2 Simbol – Simbol Activity Diagram 26](#_Toc77418007)

[Tabel 4.1 Analisi Software 35](#_Toc77418008)

[Tabel 4.2 Analisis data aplikiasi yang akan di buat 38](#_Toc77418009)

[Tabel 4.3 Tabel Definisi Aktor 39](#_Toc77418010)

[Tabel 4.4 Tabel Definisi *Use Case* 39](#_Toc77418011)

[Tabel 4.5 Skenario *Login* 43](#_Toc77418012)

[Tabel 4.6 SkenarioUbah Data Sekolah 43](#_Toc77418013)

[Tabel 4.7 Skenario Unggah Kop Surat 44](#_Toc77418014)

[Tabel 4.8 Skenario Tambah Data Pegawai 45](#_Toc77418015)

[Tabel 4.9 Skenario Ubah Data Pegawai 46](#_Toc77418016)

[Tabel 4.10 Skenario Hapus Data Pegawai 47](#_Toc77418017)

[Tabel 4.11 Skenario Tambah Data Surat 48](#_Toc77418018)

[Tabel 4.12 Skenario Ubah Data Surat 49](#_Toc77418019)

[Tabel 4.13 Skenario Hapus Data Surat 50](#_Toc77418020)

[Tabel 4.14 Skenario Cetak Surat Tugas 51](#_Toc77418021)

[Tabel 4.15 Skenario Cetak SPPD 1 51](#_Toc77418022)

[Tabel 4.16 Skenario Cetak SPPD 2 51](#_Toc77418023)

[Tabel 4.17 Cetak Rekap Data Pegawai 52](#_Toc77418024)

[Tabel 4.18 Cetak Berdasarkan Nama Pegawai 52](#_Toc77418025)

[Tabel 4.19 Cetak berdasarkan periode tanggal 53](#_Toc77418026)

[Tabel 4.20 Cetak Semua Data Surat 54](#_Toc77418027)

[Tabel 4.21 Tabel Data Pengguna 64](#_Toc77418028)

[Tabel 4.22 Tabel data sekolah 64](#_Toc77418029)

[Tabel 4.23 Tabel data sekolah 65](#_Toc77418030)

[Tabel 4.24 Tabel 4.25 Tabel Data Surat 65](#_Toc77418031)

[Tabel 4.26 Tabel data sekolah 66](#_Toc77418032)

[Tabel 5.1 Tabel Rencana Pengujian 87](#_Toc77418033)

[Tabel 5.2 Tabel Rencana Pengujian 89](#_Toc77418034)

[Tabel 5.3 Hasil Pengujian PDHUPL-01 92](#_Toc77418035)

[Tabel 5.4 Hasil Pengujian PDHUPL-02 92](#_Toc77418036)

[Tabel 5.5 Hasil Pengujian PDHUPL-03 93](#_Toc77418037)

[Tabel 5.6 Hasil Pengujian PDHUPL-04 93](#_Toc77418038)

[Tabel 5.7 Hasil Pengujian PDHUPL-05 94](#_Toc77418039)

[Tabel 5.8 Hasil Pengujian PDHUPL-06 94](#_Toc77418040)

[Tabel 5.9 Hasil Pengujian PDHUPL-07 95](#_Toc77418041)

[Tabel 5.10 Hasil Pengujian PDHUPL-08 95](#_Toc77418042)

[Tabel 5.11 Hasil Pengujian PDHUPL-09 96](#_Toc77418043)

[Tabel 5.12 Hasil Pengujian PDHUPL-10 96](#_Toc77418044)

[Tabel 5.13 Hasil Pengujian PDHUPL-11 97](#_Toc77418045)

[Tabel 5.14 Hasil Pengujian PDHUPL-12 97](#_Toc77418046)

[Tabel 5.15 Hasil Pengujian PDHUPL-13 98](#_Toc77418047)

[Tabel 5.16 Hasil Pengujian PDHUPL-14 98](#_Toc77418048)

[Tabel 5.17 Hasil Pengujian PDHUPL-15 99](#_Toc77418049)

[Tabel 5.18 Hasil Pengujian PDHUPL-16 99](#_Toc77418050)

[Tabel 5.19 Hasil Pengujian PDHUPL-17 100](#_Toc77418051)

[Tabel 5.20 Hasil Pengujian PDHUPL-18 100](#_Toc77418052)

[Tabel 5.21 Hasil Pengujian PDHUPL-19 101](#_Toc77418053)

[Tabel 5.22 Hasil Pengujian PDHUPL-20 101](#_Toc77418054)

[Tabel 5.23 Hasil Pengujian PDHUPL-21 102](#_Toc77418055)

[Tabel 5.24 Hasil Pengujian PDHUPL-22 102](#_Toc77418056)

[Tabel 5.25 Hasil Pengujian PDHUPL-23 103](#_Toc77418057)

[Tabel 5.26 Hasil Pengujian PDHUPL-24 103](#_Toc77418058)

[Tabel 5.27 Hasil Pengujian PDHUPL-25 104](#_Toc77418059)

[Tabel 5.28 Hasil Pengujian PDHUPL-26 104](#_Toc77418060)

[Tabel 5.29 Hasil Pengujian PDHUPL-27 105](#_Toc77418061)

# **BAB I PENDAHULUAN**

## **Latar Belakang**

“Sistem teknologi dan informasi yang ada pada bidang pendidikan sebenarnya tidak hanya digunakan pada kegiatan maupun sistem belajar mengajar dan siswa-siswinya saja melainkan seluruh kegiatan dan sistem yang ada sekolah meliputi seluruh pegawainya. Salah satunya adalah pada sistem Persuratan, terutama pada bagian Surat Tugas Perjalanan Dinas” (Fadillah, 2018).

Berdasarkan Peraturan Menteri Pendidikan Dan Kebudayaan Republik Indonesia Nomor 6 Tahun 2013 tentang Tata Naskah Di Lingkungan Kementrian Pendidikan Dan Kebudayaan Bagian Kedua Belas, Surat tugas adalah naskah dinas yang berisi penugasan dari pejabat yang berwenang kepada seseorang untuk melaksanakan suatu kegiatan (Mendikbud, 2013)

Surat Perintah Perjalanan Dinas (SPPD) diterangkan dalam Pasal 1 Ayat 9 Peraturan Menteri Keuangan Nomor 45 Tahun 2007 adalah perjalanan ke luar tempat kedudukan baik perseorangan maupun secara bersama yang jaraknya sekurang-kurangnya 5 (lima) kilometer dari batas kota, yang dilakukan dalam wilayah Republik Indonesia untuk kepentingan Negara atas perintah Pejabat yang Berwenang, termasuk perjalanan dari tempat kedudukan ke tempat meninggalkan Indonesia untuk bertolak ke luar negeri dan dari tempat tiba di Indonesia dari luar negeri ke tempat yang dituju di dalam negeri (PERMENKEU, 2007)

Melalui wawancara yang dilakukan dengan pihak SD Negeri Mekarsari Baleendah dapat diketahui bahwa dalam sistem tata naskah terutama untuk Surat Tugas dan SPPD belum ada aplikasi yang mengatur sistem tata naskahnya. Proses pembuatan Surat Tugas dan SPPD masih menggunakan aplikasi Microsoft Word dimana harus dilakukan pengeditan

setiap kali akan melakukan perjalanan dinas. Selain itu pembuat Surat Tugas dan SPPD harus mencari data setiap pegawai yang akan melakukan perjalanan dinas pada Microsoft Excel. Dalam pembuatan Surat Tugas dan SPPD yang ada saat ini ditemui banyak kekurangan-kekurangan yang harus dihadapi seperti terjadinya kesalahan-kesalahan pengetikan dalam menginputkan data sehingga pembuatan Surat Tugas dan SPPD dilakukan berulang - ulang, dan juga pencarian data pegawai yang diusulkan dan pegawai pemberi tugas dalam pembuatan surat tugas belum dilakukan secara otomatis sehingga membutuhkan waktu yang lebih lama. Sistem pengarsipan di SD Negeri Mekarsari Baleendah sendiri masih menggunakan cara mencatat (tulis tangan) pada buku dan melihat ulang urutan surat sebelumnya untuk memberi nomor pada surat baru. Dengan cara tersebut tentu tidak efisien karena memerlukan buku yang tebal terlebih lagi jika untuk mencatat dalam jangka waktu yang lama dan banyaknya jenis surat yang akan di arsipkan. Apalagi permasalahan yang rentan terjadi adalah penyimpanan berkas yang terlalu banyak sehingga menjadi kendala saat melakukan pencarian data. Dengan masalah tersebut, penelitian ini akan membuat sistem komputer yang mampu menyimpan arsip tersebut ke dalam memori penyimpanan komputer dan mampu menampilkan nomor surat secara otomatis sehingga pengarsipan dan pemberian nomor surat menjadi lebih efisien. Dalam proses perekapan laporan pun, SD Negeri Mekarsari membutuhkan waktu yang lama karena harus membuka kembali satu-persatu arsip perjalanan dinas yang sangat banyak. Cara hitung satu persatu tersebut untuk melihat berapa kali Guru atau Tenaga Kependidikan melakukan Pejalanan Dinas dalam satu bulan atau satu tahun berdasarkan rekap Surat Tugas dan SPPD yang dicatat pada buku surat dan SPJ. Hal ini tentu saja menyulitkan dalam perekapan laporan bulanan dan tahunan karena membutuhkan waktu yang lama.

Pernah dilakukan penenlitian serupa untuk masalah diatas oleh Saprina Mamase pada Jurnal Teknik Informatika yang berjudul “Rancang Bangun Sistem Informasi Surat Perintah Perjalanan Dinas”. Hasil penelitian tersebut adalah membuat aplikasi surat tugas menggunakan Web (Mamase, 2016). Namun terdapat beberapa kekurangan diantaranya aplikasi tersebut tidak bisa diimplementasikan di instansi sekolah khusus nya di SD Negeri Mekarsari dikarenakan adanya perbedaan dalam format SPPD dan aplikasi tersebut hanya membahas terkait surat perintah perjalanan dinas (SPPD) dan tidak membahas terkait surat tugas. Karena pada dasar nya surat perintah perjalanan dinas(SPPD) dan surat tugas adalah dua jenis surat yang berbeda dan tidak dapat dipisahkan.

Solusi dari masalah diatas adalah dengan membuat aplikasi surat tugas dan surat perintah perjalanan dinas(SPPD) yang sesuai dengan format di SD Negeri Mekarsari Baleendah. Tujuan penelitian adalah untuk membuat sebuah aplikasi yang dapat membantu dalam proses pembuatan sampai pelaporan surat tugas dan SPPD. Aplikasi ini diharapkan mampu membantu operator mulai dari proses pembuatan, pencetakan surat, penyimpanan data surat, sampai ke pelaporan rekap surat itu sendiri agar pembuatannya lebih cepat, dan terhindar dari Human Error.

## **Rumusan Masalah**

1. Bagaimana memudahkan pembuatan dan pencetakan Surat Tugas dan SPPD?
2. Bagaimana mengefisienkan proses penyimpanan arsip Surat Tugas dan SPPD secara sistematis dan terstruktur ?
3. Bagaimana mempercepat proses perekapan dan pencetakan laporan Surat Tugas dan SPPD?

## **Batasan Masalah**

Agar penelitian lebih terfokus dan terarah serta agar tujuan dari penelitian ini dapat tercapai, maka perlu diterapkan batasan-batasan terhadap permasalahan yang diteliti, yaitu :

1. Hanya mengelola Surat Tugas dan SPPD di SD Negeri Mekarsari.
2. Tidak membahas terkait biaya yang dikeluarkan per satu Perjalan Dinas.
3. Hanya merekap laporan berdasarkan tanggal dan nama pegawai.

## **Tujuan Penelitian**

1. Memudahkan pembuatan dan pencetakan Surat Tugas dan SPPD.
2. Mengefisienkan proses penyimpanan arsip Surat Tugas dan SPPD menjadi sistematis dan terstruktur.
3. Mempercepat proses perekapan dan pencetakan laporan Surat Tugas dan SPPD.

## **Metodologi**

### **Metode Pengumpulan Data**

Metode pengumpulan data ini yaitu meliputi :

1. Observasi

Observasi dilakukan oleh penulis dengan cara pengamatan dan pencatatan langkah demi langkah untuk suatu penentuan masalah yang di alami di SD Negeri Mekarsari Baleendah terkait pembuatan, pengarsipan dan perekapan surat tugas dan surat perintah perjalanan dinas (SPPD)

1. Wawancara

Penelitian ini dilakukan dengan proses tanya jawab kepada Ibu Komalasari, S.Pd, M.Pd selaku Kepala Sekolah SD Negeri Mekarsari Baleendah

1. Studi Pustaka

Studi pustaka yang di lakukan penulis antara lain :

* 1. Mengumpulkan Jurnal yang memuat artikel-artikel tentang penelitian sebelumnya dan terkait dengan permasalahan yang ditemukan di SD Negeri Mekarsari
  2. Mencari buku sumber yang relevan dengan hal-hal yang di butuhkan untuk dijadikan dasar teori penelitian.

### **Metode Pengembangan Sistem**

Metode Pengembangan Sistem pada penelitian ini menggunakan Model Waterfall.

## **Sistematika Penulisan**

Sistematika penulisan yang akan di bahas dalam penulisan laporan tugas akhir ini terdiri dari 6 (enam) bab yaitu sebagai berikut :

BAB I PENDAHULUAN

Pada bab ini dijelaskan beberapa hal yang mendasari atau yang melatarbelakangi penulisan tugas akhir ini. Di dalam bab I ini dideskripsikan tentang Latar belakang, Rumusan masalah, Batasan masalah, Tujuan, Metodologi dan Sistematika penulisan yang digunakan.

BAB II TINJUAN PUSTAKA

Uraian dan penjelasan teori yang berhubungan dengan judul yaitu tentang *Pembuatan Aplikasi Surat Tugas Dan Surat Perintah Perjalanan Dinas Menggunakan Embarcadero Delphi Xe8* dan beberapa hal mengenai permasalahan konsep dasar dan penelitian terdahulu dalam jurnal yang berhubungan dengan penelitian ini Lengkapnya dalam bab ini dijelaskan beberapa teori pendukung yang diperlukan dalam penelitian ini.

BAB III METODOLOGI

Bab ini berisi penjelasan mengenai kerangka pikir dan deskripsinya yang berisi gambaran pola hubungan antar variabel atau kerangka konsep yang akan digunakan untuk menjelaskan masalah yang diteliti dan disusun berdasarkan kajian teoritik.

BAB IV ANALISIS DAN PERANCANGAN SISTEM

Pada bab ini berisi penjelasan mengenai analisis yang dilakukan untuk menentukan masalah, kebutuhan aplikasi, perancangan *database* dan perancangan sistem aplikasi yang akan dibangun.

BAB V IMPLEMENTASI DAN PENGUJIAN

Bab ini berisi pengujian serta hasil pengamatan secara keseluruhan pada aplikasi Surat Tugas Dan Surat Perintah Perjalanan Dinas Menggunakan Embarcadero Delphi Xe8.

BAB VI KESIMPULAN DAN SARAN

Bab ini akan diuraikan kesimpulan dan hasil analisis yang dapat memberikan saran-saran kepada SD Negeri Mekarsari.

# **BAB II TINJAUAN PUSTAKA**

## **Landasan Teori**

Dalam penelitian ini akan digunakan tiga tinjauan studi yang nantinya mendukung penelitian yang akan dilakukan, dimana tinjauan studi yang diambil adalah :

1. Oleh Saprina Mamase (2016) dari Program Studi Teknik Informatika Politeknik Gorontalo dengan judul Rancang Bangun Sistem Informasi Surat Perintah Perjalanan Dinas. Penulis membahas tentang Mekanisme pembuatan SPPD yang masih menggunakan aplikasi Microsoft Word sehingga dalam pembuatan SPPD banyak ditemui banyak kekurangan, seperti kesalahan – kesalahan pengetikan seperti di SD Negeri Mekarsari. Hasil Penelitian ini yaitu terbentuknya sistem yang dapat membantu dalam proses pembuatan Surat Perintah Perjalanan Tugas (SPPD). Di Kantor Sekretariat Daerah agar bisa mempermudah efektifitas kinerja pembuatan Surat Perintah Dinas (SPPD) dengan lebih mudah, hemat biaya, dan hemat waktu. Serta dapat membantu proses penyimpanan data SPPD tanpa harus takut untuk kehilangan surat ataupun rusak akibat hal yang tidak di inginkan. Pemberian perintah perjalanan dinas menjadi terstruktur dan tidak akan terjadi kesalahan jadwal perjalanan dinas (Mamase, 2016).
2. Oleh Rendi Handika, Ady Widjaja (2018) dari Program Studi Fakultas Teknik Informasi Universitas Budi Luhur dengan Rancangan Sistem Informasi Surat Perintah Perjalanan Dinas Pada Balai Teknologi Air Dan Limbah Dengan Berbasis Aplikasi Web. Salah satu masalah yang diangkat dalam penelitian tersebut adalah Bagaimana Memudahkan proses penyimpanan arsip Surat Tugas dan SPPD menjadi sistematis dan terstruktur (Handika & Widjaja, 2018).
3. Oleh Hendri Fonda (2019) dari Program Studi Sistem Informasi,STMIK Hang Tuah Pekanbaru yang berjudul Penerapan Aplikasi Surat Perintah Perjalanan Dinas (SPPD) Pada Bagian Komunikasi dan Informatika Sekertariat Daerah Kabupaten Kepulauan Meranti Berbasis Web. Tujuan penelitan penulis salah satunya adalah untuk memudahkan admin dalam pembuatan rekapan laporan surat perintah perjalanan dinas yang harus disiapkan untuk keperluan surat pertanggung jawaban (SPJ) (Fonda, 2019).

## **Dasar Teori**

Dalam Melakukan penelitian, penulis Juga memiliki dasar untuk menunjang penelitian yang tengah dilakukan dan dasar tersebut adalah :

### **Pengertian Aplikasi**

Istilah aplikasi berasal dari bahasa inggris application yang berarti penerapan, lamaran ataupun penggunaan. Sedangkan secara istilah, pengertian aplikasi adalah suatu program yang siap untuk digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju. Menurut kamus komputer eksekutif, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputasi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan.

Suatu sistem informasi adalah seperangkat aplikasi komputer yang dapat memberi dukungan operasi dari suatu organisasi seperti operasi, instalasi serta pada saat perawatan komputer, perangkat lunak dan data. Aplikasi biasanya berupa perangkat lunak yang berbentuk software yang berisi kesatuan perintah atau program yang dibuat untuk melaksanakan sebuah pekerjaan yang diinginkan.

Selain itu aplikasi juga mempunyai fungsi sebagai pelayan kebutuhan beberapa aktivitas yang dilakukan oleh manusia seperti sistem untuk software jual beli, permainan atau game online, pelayanan masyarakat dan hampir semua proses yang dilakukan oleh manusia dapat dibantu dengan menggunakan suatu aplikasi. Beberapa aplikasi jika digabungkan akan menjadi satu paket atau sering juga disebut dengan aplication suite, di mana aplikasi tersebut memiliki posisi antar muka yang mempunyai kesamaan sehingga dapat dengan mudah digunakan atau dipelajari penggunaan tiap aplikasi tersebut.(D & Sriwinar, 2021)

### **2.2.2. Pengertian Database**

*database* adalah kumpulan data terstruktur. Agar dapat menambahkan, mengakses, dan memproses data yang tersimpan dalam database komputer, dibutuhkan sistem manajemen basis data (*database management system*) (Suharyanto et al., 2017).

Pengertian *database* yang paling sederhana adalah kumpulan dari tabel. Satu tabel merepresentasikan suatu entitas tertentu. Suatu entitas terdiri atas beberapa atribut. Tabel-tabel yang ada dikumpulan dalam satu *container* atau wadah. Wadah inilah yang sering disebut *database*. Jadi dalam satu *database* terdiri atas beberapa tabel.

Database pada *Microsoft Access* adalah sekumpulan object yang terdiri dari *table, query form, report*. Objek-objek ini ditampung dalam satu wadah atau *database*. Jadi dalam *Microsoft Access* yang dimaksud *database* adalah kumpulan dari *table, query form, report*. Di dalam *Microsoft Access* sebuah *database* disimpan dalam satu file berekstensi mdb. Jadi dalam *Microsoft Access* satu *file* mewakili atau mempresentasikan satu *database*, di mana satu *database* ini bisa terdiri atas beberapa *table, query* atau *report*. Ada 4 jenis *databse* yang sering muncul yaitu :

1. *Hierarchy*
2. *Network*
3. *Relational*
4. *Object Oriented*

Dari ke-empat jenis tersebut, jenis *database* relasional-lah yang paling sering digunakan. Namun, bukan berarti jenis *database* yang lain tidak dipakai jenis *database* yang lain, seperti *database network* pun digunakan untuk keperluan tertentu sesuai dengan kebutuhan dan kompleksitas permasalahan atau kompleksitas sistem yang dibangun.

### **2.2.3. PERMENDIKBUD No 6 Tahun 2013** (Mendikbud, 2013)

Berdasarkan Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia Nomor 6 Tahun 2013 tentang Tata Naskah Dinas di Lingkungan Kementrian Pendidikan dan Kebudayaan jenis tata naskah dinas terdiri dari beberapa surat, diantaranya surat tugas. Dalam Permendikbud tersebut, surat tugas di atur pada bagian kedua belas yang selanjutnya di bagi dalam beberapa pasal, diantaranya :

1. **Pasal 79**

(1) Surat tugas adalah naskah dinas yang berisi penugasan dari pejabat yang berwenang kepada seseorang untuk melaksanakan suatu kegiatan.

(2) Surat tugas dapat berbentuk lembaran surat atau kolom.

(3) Bagian-bagian surat tugas terdiri atas:

a. kepala surat;

b. pembuka surat;

c. isi surat; dan

d. penutup surat.

1. **Pasal 80**

Kepala surat tugas, baik yang berbentuk lembaran surat maupun kolom, dibuat dengan menggunakan ketentuan sebagaimana dimaksud dalam Pasal 2 ayat (2) Peraturan Menteri ini.

1. **Pasal 81**

(1) Pembuka surat tugas terdiri atas:

a. frasa surat tugas; dan

b. nomor.

(2) Frasa surat tugas ditulis di bawah kepala surat dengan huruf kapital secara simetris.

(3) Kata nomor ditulis sejajar dengan frasa surat tugas, diawali dengan huruf kapital.

1. **Pasal 82**

(1) Isi surat tugas yang berbentuk lembaran surat terdiri atas:

a. nama jabatan pemberi tugas;

b. nama, NIP, pangkat dan golongan, serta jabatan penerima tugas; dan

c. maksud, tanggal, dan tempat penugasan.

(2) Nama jabatan pemberi tugas ditulis di sebelah kiri di bawah kata nomor.

(3) Kata nama ditulis di bawah dan sejajar dengan awal frasa nama jabatan pemberi tugas.

(4) Singkatan NIP ditulis di bawah dan sejajar dengan kata nama.

(5) Kata pangkat dan golongan ditulis di bawah dan sejajar dengan NIP.

(6) Kata jabatan ditulis di bawah dan sejajar dengan kata pangkat dan golongan.

(7) Maksud, tanggal, dan tempat penugasan, ditulis di bawah dan sejajar dengan kata jabatan, didahului dengan kata untuk.

1. **Pasal 83**

(1) Isi surat tugas yang berbentuk kolom terdiri atas:

a. nama jabatan pemberi tugas;

b. kolom isian surat tugas; dan

c. maksud, tanggal, dan tempat penugasan.

(2) Nama jabatan pemberi tugas ditulis di sebelah kiri di bawah kata nomor.

(3) Kolom isian surat tugas berisi nomor, nama, NIP, pangkat dan golongan, serta jabatan yang diberi tugas ditulis di bawah dan sejajar dengan kalimat awal nama jabatan pemberi tugas.

(4) Maksud, tanggal, dan tempat penugasan ditulis di bawah kolom sejajar dengan nomor isi kolom dan didahului dengan kata untuk.

1. **Pasal 84**

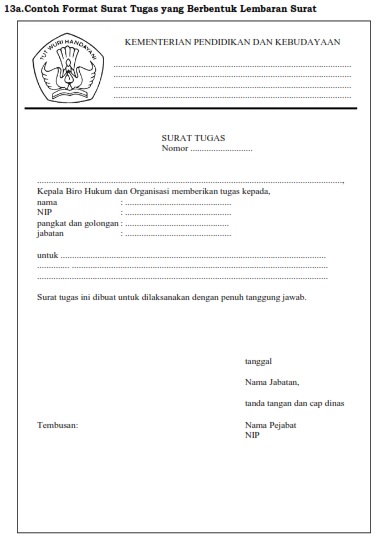
Penutup surat tugas dibuat dengan menggunakan ketentuan sebagaimana dimaksud dalam Pasal 39 dan Pasal 40 Peraturan Menteri ini.

1. **Pasal 85**

Bentuk surat tugas dibuat dengan menggunakan contoh sebagaimana tercantum pada Nomor 13a dan 13b dalam Lampiran yang merupakan bagian tidak terpisahkan dari Peraturan Menteri ini.

### **2.2.4. Lampiran PERMENDIKBUD No 6 Tahun 2013**

Lampiran Permendikbud ini berisi tentang format tata naskah di Lingkungan Kementrian Pendidikan dan Kebudayaan Republik Indonesia. Format Surat Tugas terdapat pada point 13a. Berikut Format Surat Tugas berdasarkan Permendikbud No 6 Tahun 2013 point 13a :



Gambar 2.1 Contoh Format Surat Tugas

### **2.2.5. PERMENKEU No 45 Tahun 2007** (PERMENKEU, 2007)

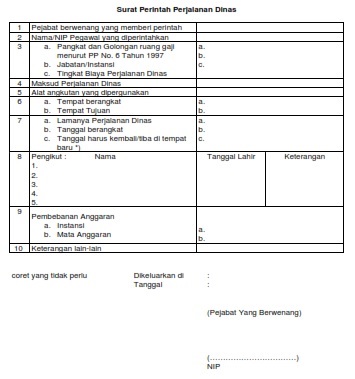
Surat Perintah Perjalanan Dinas (SPPD) diatur dalam Peraturan Menteri Keuangan Republik Indonesia No 45 Tahun 2007 tentang Perjalanan Dinas Jabatan Dalam Negeri Bagi Pejabat Negara, Pegawai Negeri dan Pegawai Tidak Tetap. Salah satu pasal yang mengatur tentang hal tersebut Pasal 1 yang di bagi lagi dalam beberapa ayat diantaranya :

* 1. Pejabat Negara, Pegawai Negeri Sipil, dan Pegawai Tidak Tetap adalah Pejabat Negara, Pegawai Negeri Sipil, dan Pegawai Tidak Tetap sebagaimana dimaksud dalam UndangUndang Nomor 43 Tahun 1999 tentang perubahan atas Undang-Undang Nomor 8 Tahun 1974 tentang Pokok-pokok Kepegawaian;
  2. Pegawai Negeri adalah Pegawai Negeri Sipil, Anggota Tentara Nasional Indonesia (TNI), dan Anggota Kepolisian Negara Republik Indonesia (Polri).
  3. Pegawai Tidak Tetap adalah Pegawai yang diangkat untuk jangka waktu tertentu guna melaksanakan tugas pemerintahan dan pembangunan yang bersifat teknis profesional dan administrasi sesuai dengan kebutuhan dan kemampuan organisasi dalam kerangka sistem kepegawaian, yang tidak berkedudukan sebagai pegawai negeri.
  4. Pejabat yang Berwenang adalah Pengguna Anggaran / Kuasa Pengguna Anggaran atau pejabat yang diberi wewenang oleh Pengguna Anggaran/Kuasa Pengguna Anggaran di lingkungan Kementerian Negara/Lembaga.
  5. Perjalanan dinas dalam negeri yang selanjutnya disebut perjalanan dinas adalah perjalanan ke luar tempat kedudukan baik perseorangan maupun secara bersama yang jaraknya sekurang-kurangnya 5 (lima) kilometer dari batas kota, yang dilakukan dalam wilayah Republik Indonesia untuk kepentingan Negara atas perintah Pejabat yang Berwenang, termasuk perjalanan dari tempat kedudukan ke tempat meninggalkan Indonesia untuk bertolak ke luar negeri dan dari tempat tiba di Indonesia dari luar negeri ke tempat yang dituju di dalam negeri.
  6. Lumpsum adalah uang yang dibayarkan sekaligus.
  7. Biaya Riil adalah biaya yang dikeluarkan sesuai dengan bukti pengeluaran yang sah.
  8. Perhitungan Rampung adalah perhitungan biaya perjalanan yang dihitung sesuai kebutuhan riil berdasarkan ketentuan yang berlaku.
  9. Surat Perintah Perjalanan Dinas yang selanjutnya disebut SPPD adalah surat perintah kepada Pejabat Negara, Pegawai Negeri, dan Pegawai Tidak Tetap untuk melaksanakan perjalanan dinas.

### **2.2.6. Lampiran PERMENKEU No 45 Tahun 2007**

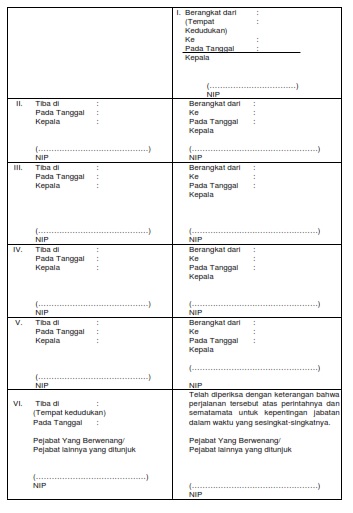
Berikut bentuk tampilan format SPPD berdasarkan lampiran Peraturan Menteri Keuangan No 45 Tahun 2007.

* 1. Halaman 1



Gambar 2.2 Contoh Format SPPD Halaman 1

* 1. Halaman 2



Gambar 2.3 Contoh Format SPPD Halaman 2

### **2.2.7. Delphi**

Delphi adalah salah satu bahasa pemrograman berbasis visual yang digunakan untuk membuat program aplikasi pada komputer (seperti Visual basic). Bahasa pemrograman yang digunakan oleh Delphi sebenarnya merupakan turunan dari bahasa pemrograman pascal, yang dahulu pada Delphi dikenal sebagai objek pascal. Bagi Anda yang telah mengenal bahasa pemrograman pascal, maka mungkin Anda tidak akan terlalu kesulitan dalam mempelajari Delphi(ir. Inge Martina, 2012).

Delphi relatif lebih mudah dipahami dibandingkan dengan bahasa pemrograman lainnya, disamping itu banyak referensi terutama dalam bentuk buku yang membahas tentang bahasa pemrograman ini. Delphi telah terbukti mampu menghasilkan software-software yang berkualitas, baik yang berskala besar maupun kecil, teknologi yang digunakan pada Delphi pun cukup uptodate (Hafid Mukhlasin, 2008).

Delphi merupakan program aplikasi database yang berbasis Object Pascal dari Borland. Selain itu, Delphi juga memberikan fasilitas pembuatan aplikasi visual. Delphi merupakan pilihan dalam pembuatan aplikasi visual karena memberikan produktivitas yang tinggi.Delphi Memberikan fasilitas untuk dua platform, yaitu untuk platform Windows dan Linux. Delphi untuk linux sebelumnya dikemas dalam sebuah aplikasi terpisah bernama kylix, tetapi Delphi selanjutnya menyatukan dalam sebuah aplikasi. Library untuk windows disebut VCL dan library untuk Linux disebut CLX.(ir. Inge Martina, 2012)

Delphi adalah sebuah bahasa pemrograman dan lingkungan pengembangan perangkat lunak Delphi awalnya sebuah proyek penelitian rahasia di Borland yang berevolusi menjadi sebuah produk yang disebut AppBuilder. Borland sebagai pengembang perangkat lunak yang sudah ada sejak era 1980an menggunakan ikon dan nama Delphi sebagai salah satu perangkat lunak pembantu pemrograman untuk membuat program yang sudah mengarah ke Pemrograman yang berorientasi pada objek atau dikenal dengan istilah OOP (Object Oriented Programming). Salah satu tujuan asli dari Delphi adalah untuk menyediakan konektivitas database untuk programmer sebagai kunci dan sebuah paket database yang populer pada saat itu adalah database Oracle. (Aulina, n.d.)

1. **Borland Delphi 1 (windows 3.1/16bit)**

Borland Delphi 1 dirilis pada tahun 1995, untuk aplikasi 16-bit Windows 3.1 dan merupakan contoh awal dari apa yang kemudian dikenal sebagai Rapid Application Development (RAD). Delphi 1 adalah penerus Turbo Pascal dan Borland Pascal, sangat sedikit cepat dalam men-compile aplikasi 16bit. dengan canggihnya perusahaan ini, memiliki Integrated Development Environment (IDE) dan toolkit antarmuka pengguna grafis untuk DOS (Borland Pascal ditambah Windows IDE dan juga kode generasi Windows ). Seperti Turbo Pascal.

1. **Borland Delphi 2 (windows 95/Delphi 32bit)**

Borland Delphi 2, dirilis pada tahun 1996, yang didukung 32-bit aplikasi windows. Yang didalamnya termasuk Delphi 1 dan dibundel bersama-sama untuk menciptakan 32bit aplikasi windows 3.1.

1. **Borland delphi 3 (windows 95 keatas)**

Borland Delphi 3 dirilis pada tahun 1997, yang merupakan versi klasik dari delphi. Tidak jauh berbeda dengan versi sebelumnya hanya saja ada sedikit tambahan fitur internet atau web.

1. **Borland delphi 4**

Borland delphi 4 dirilis pada tahun 1998, Borland kembali merevisi ulang Delphi 3 dan menelurkan produk Borland Delphi 4.

1. **Borland delphi 5**

Borland delphi 5 dirilis pada tahun 1999, Borland kembali merevisi ulang Delphi 4 dan menelurkan produk Borland Delphi 5.\

1. **Kylix For Linux**

Kylix dirilis pada tahun 2001, yang merupakan sebuah versi linux dari Borland delphi.

1. **Borland Delphi 6**

Borland Delphi 6 dirilis pada tahun 2001, Borland kembali merevisi ulang Delphi 5 dan menelurkan produk Borland Delphi 6.

1. **Borland Delphi 7**

Borland Delphi 7 dirilis pada tahun 2002, Borland kembali merevisi ulang Delphi 6 dan menelurkan produk Borland Delphi 7.

1. **Borland Delphi 8.NET**

Borland Delphi 8 dirilis pada tahun 2003, bertepatan dengan The Mother Day alias Hari Ibu Nasional, Borland melakukan migrasi untuk memindahkan platform Delphi ke arah pemrograman .NET (dot NET) dengan menelurkan produk Borland Delphi 8 for .NET.

1. **Borland Developer Studio 2005**

Borland Developer Studio 2005 dirilis pada tahun 2004, Borland menggabungkan pengembangan perangkat lunak C# dan Delphi ke dalam satu kendali Integrated Develompment Environment (IDE) dengan menelurkan produk Borland Delphi 2005.

1. **Borland Developer Studio 2006**

Borland Developer Studio 2006 dirilis pada akhir tahun 2005, Borland kembali merevisi ulang perangkat lunak Delphi Developer Studio 2005 ke dalam perangkat lunak baru Borland Developer Studio 2006

1. **CodeGear RAD Studio 2007**

CodeGear RAD Studio 2007 dirilis pada tahun 2007, versi ini merupakan versi pertama dari CodeGear.

1. **Embarcadero RAD Studio 2009**

Embarcadero RAD Studio 2009 dirilis pada tahun 2009, setelah Borland dan CodeGear sah dijual kepada Embarcadero tahun 2008. Embarcadero RAD Studio 2009 menambah banyak fitur baru seperti VCL dan RTL untuk pendukung Unicode, serta sebuah produk baru Delphi Prism untuk pembangunan .NET.

1. **Embarcadero RAD Studio 2010**

Embarcadero RAD studio 2010 dirilis pada tahun 2009, Embarcadero Technologieskembali merevisi ulang perangkat lunak Embarcadero RAD Studio 2009 ke dalam perangkat lunak baru Embarcadero RAD Studio 2010. Salah satu fitur baru yang mendukung teknologi layar sentuh.

1. **Embarcadero RAD Studio XE**

Embarcadero RAD studio XE dirilis pada tahun 2010, Embarcadero Technologieskembali merevisi ulang perangkat lunak Embarcadero RAD Studio 2010 ke dalam perangkat lunak baru Embarcadero RAD Studio XE (alias dephi 2011 dengan kode nama fulcrum). Dilanjutkan dengan versi XE2(2011), XE3(2012), XE4(2013), XE5(2013), XE6(2014), XE7(2014), XE8(2015) (Indriawan et al., 2011).

### **2.2.8. Microsoft Access 2010** (Heryanto, 2017)

Microsoft Office Access adalah sebuah program aplikasi basis data komputer relasional yang ditujukan untuk kalangan rumahan dan perusahaan kecil hingga menengah. Aplikasi ini merupakan anggota dari beberapa aplikasi Microsoft Office, selain tentunya Microsoft Word, Microsoft Excel, dan Microsoft PowerPoint. Aplikasi ini menggunakan mesin basis data Microsoft Jet Database Engine, dan juga menggunakan tampilan grafis yang intuitif sehingga memudahkan pengguna. Versi terakhir adalah Microsoft Office Access 2007 yang termasuk ke dalam Microsoft Office System 2007.

Microsoft Access dapat menggunakan data yang disimpan di dalam format Microsoft Access, Microsoft Jet Database Engine, Microsoft SQL Server, Oracle Database, atau semua kontainer basis data yang mendukung standar ODBC. Para pengguna/programmer yang mahir dapat menggunakannya untuk mengembangkan perangkat lunak aplikasi yang kompleks, sementara para programmer yang kurang mahir dapat menggunakannya untuk mengembangkan perangkat lunak aplikasi yang sederhana. Access juga mendukung teknik-teknik pemrograman berorientasi objek, tetapi tidak dapat digolongkan ke dalam perangkat bantu pemrograman berorientasi objek.

Microsoft merilis Microsoft Access 1.0 pada bulan November 1992 dan dilanjutkan dengan merilis versi 2.0 pada tahun 1993. Microsoft menentukan spesifikasi minimum untuk menjalankan Microsoft Access 2.0 adalah sebuah komputer dengan sistem operasi Microsoft Windows 3.0, RAM berkapasitas 4 megabyte (6 megabyte lebih disarankan) dan ruangan kosong hard disk yang dibutuhkan 8 megabyte (14 megabyte lebih disarankan). Versi 2.0 dari Microsoft Access ini datang dengan tujuh buah disket floppy 3½ inci berukuran 1.44 megabyte.

Perangkat lunak tersebut bekerja dengan sangat baik pada sebuah basis data dengan banyak record tapi terdapat beberapa kasus di mana data mengalami kerusakan. Sebagai contoh, pada ukuran basis data melebihi 700 megabyte sering mengalami masalah seperti ini (pada saat itu, memang hard disk yang beredar masih berada di bawah 700 megabyte). Buku manual yang dibawanya memperingatkan bahwa beberapa kasus tersebut disebabkan oleh driver perangkat yang kuno atau konfigurasi yang tidak benar.

Nama kode (codename) yang digunakan oleh Access pertama kali adalah Cirrus yang dikembangkan sebelum Microsoft mengembangkan Microsoft Visual Basic, sementara mesin pembuat form antarmuka yang digunakannya dinamakan dengan Ruby. Bill Gates melihat purwarupa (prototype) tersebut dan memutuskan bahwa komponen bahasa pemrograman BASIC harus dikembangkan secara bersama-sama sebagai sebuah aplikasi terpisah tapi dapat diperluas. Proyek ini dinamakan dengan Thunder. Kedua proyek tersebut dikembangkan secara terpisah, dan mesin pembuat form yang digunakan oleh keduanya tidak saling cocok satu sama lainnya. Hal tersebut berakhir saat Microsoft merilis Visual Basic for Applications (VBA).

Microsoft Access (atau Microsoft Office Access) adalah sebuah program aplikasi basis data komputer relasional yang ditujukan untuk kalangan rumahan dan perusahaan kecil sampai menengah.Aplikasi ini merupakan anggota dari beberapa aplikasi Microsoft Office, selain tentunya Microsoft Word, Microsoft Excel, dan Microsoft PowerPoint.Aplikasi ini menggunakan mesin basis data Microsoft Jet Database Engine, dan juga menggunakan tampilan grafis yang intuitif sehingga memudahkan pengguna.Versi terakhir adalah Microsoft Office Access 2007 yang termasuk ke dalam Microsoft Office System 2007.

Microsoft Access dapat menggunakan data yang disimpan di dalam format Microsoft Access, Microsoft Jet Database Engine, Microsoft SQL Server, Oracle Database, atau semua kontainer basis data yang mendukung standar ODBC. Para pengguna/programmer yang mahir dapat menggunakannya untuk mengembangkan perangkat lunak aplikasi yang kompleks, sementara para programmer yang kurang mahir dapat menggunakannya untuk mengembangkan perangkat lunak aplikasi yang sederhana.Access juga mendukung teknik-teknik pemrograman berorientasi objek, tetapi tidak dapat digolongkan ke dalam perangkat bantu pemrograman berorientasi objek.

Microsoft Access digunakan kebanyakan oleh bisnis-bisnis kecil dan menengah, di dalam sebuah organisasi yang kecil bahkan mungkin juga digunakan oleh perusahaan yang cukup besar, dan juga para programmer untuk membuat sebuah sistem buatan sendiri untuk menangani pembuatan dan manipulasi data. Access juga dapat digunakan sebagai sebuah basis data untuk aplikasi Web dasar yang disimpan di dalam server yang menjalankan Microsoft Internet Information Services (IIS) dan menggunakan Microsoft Active Server Pages (ASP).Meskipun demikian, penggunaan Access kurang disarankan, mengingat telah ada Microsoft SQL Server yang memiliki kemampuan yang lebih tinggi.

Beberapa pengembang aplikasi profesional menggunakan Microsoft Access untuk mengembangkan aplikasi secara cepat (digunakan sebagai Rapid Application Development/RAD tool), khususnya untuk pembuatan purwarupa untuk sebuah program yang lebih besar dan aplikasi yang berdiri sendiri untuk para salesman.

Microsoft Access kurang begitu bagus jika diakses melalui jaringan sehingga aplikasi-aplikasi yang digunakan oleh banyak pengguna cenderung menggunakan solusi sistem manajemen basis data yang bersifat klien/server. Meskipun demikian, tampilan muka Access (form, report, query, dan kode Visual Basic) yang dimilikinya dapat digunakan untuk menangani basis data yang sebenarnya diproses oleh sistem manajemen basis data lainnya, seperti halnya Microsoft Jet Database Engine (yang secara default digunakan oleh Microsoft Access), Microsoft SQL Server, Oracle Database, dan beberapa produk lainnya yang mendukung ODBC

Beberapa pengembang aplikasi professional menggunakan Microsoft Access untuk mengembangkan aplikasi secara cepat atau yang sering disebut dengan Rapid Application Development / RAD Tool khususnya untuk pembuatan program yang lebih besar dan aplikasi yang berdiri sendiri untuk para salesman. Dilihat dari segi perspektif programmer keunggulan Microsoft Acces adalah kompatibilitasnya dengan bahasa pemrograman Structured Query Language (SQL).

Microsoft Access mengizinkan pengembangan yang relatif cepat karena semua table basis data, kueri, form, dan report disimpan di dalam berkas basis data miliknya. Misalnya untuk membuat query, Microsoft menggunakan Query Design Grid, sebuah program berbasis grafis yang mengizinkan para penggunanya untuk membuat query tanpa harus mengetahui bahasa SQL nya. Bahasa Pemrograman yang digunakan di dalam Microsoft Access adalah Microsoft Visual Basic for Applications ( VBA ) seperti halnya dalam beberapa aplikasi Microsoft Office lainnya yang dikeluarkan oleh Microsoft.Inc .

Microsoft merilis Microsoft Access 1.0 pada bulan November 1992 dan dilanjutkan dengan merilis versi 2.0 pada tahun 1993. Microsoft menentukan spesifikasi minimum untuk menjalankan Microsoft Access 2.0 adalah sebuah komputer dengan sistem operasi Microsoft Windows 3.0, RAM berkapasitas 4 megabyte (6 megabyte lebih disarankan) dan ruangan kosong hard disk yang dibutuhkan 8 megabyte (14 megabyte lebih disarankan). Versi 2.0 dari Microsoft Access ini datang dengan tujuh buah disket floppy 3½ inci berukuran 1.44 megabyte.

Perangkat lunak tersebut bekerja dengan sangat baik pada sebuah basis data dengan banyak record tapi terdapat beberapa kasus di mana data mengalami kerusakan. Sebagai contoh, pada ukuran basis data melebihi 700 megabyte sering mengalami masalah seperti ini (pada saat itu, memang hard disk yang beredar masih berada di bawah 700 megabyte). Buku manual yang dibawanya memperingatkan bahwa beberapa kasus tersebut disebabkan oleh driver perangkat yang kuno atau konfigurasi yang tidak benar.

Ruby. Bill Gates melihat purwarupa (prototype) tersebut dan memutuskan bahwa komponen bahasa pemrograman BASIC harus dikembangkan secara bersama-sama sebagai sebuah aplikasi terpisah tapi dapat diperluas. Proyek ini dinamakan dengan Thunder. Kedua proyek tersebut dikembangkan secara terpisah, dan mesin pembuat form yang digunakan oleh keduanya tidak saling cocok satu sama lainnya. Hal tersebut berakhir saat Microsoft merilis Visual Basic for Applications (VBA).

Perkembangan Microsoft Acces terus mengalami peningkatan seiring dengan berkembangnya teknologi informatika, maka dari itu pula versi-versi

Microsoft Access digunakan kebanyakan oleh bisnis-bisnis kecil dan menengah, di dalam sebuah organisasi yang kecil bahkan mungkin juga digunakan oleh perusahaan yang cukup besar, dan juga para programmer untuk membuat sebuah sistem buatan sendiri untuk menangani pembuatan dan manipulasi data. Access juga dapat digunakan sebagai sebuah basis data untuk aplikasi Web dasar yang disimpan di dalam server yang menjalankan Microsoft Internet Information Services (IIS) dan menggunakan

Microsoft Active Server Pages (ASP). Meskipun demikian, penggunaan Access kurang disarankan, mengingat telah ada Microsoft SQL Server yang memiliki kemampuan yang lebih tinggi.

Beberapa pengembang aplikasi profesional menggunakan Microsoft Access untuk mengembangkan aplikasi secara cepat (digunakan sebagai Rapid Application Development/RAD tool), khususnya untuk pembuatan purwarupa untuk sebuah programyang lebih besar dan aplikasi yang berdiri sendiri untuk para salesman.

Biasanya Karyawan kalau Belanja itu lebih satu barang. untuk itu kita perlu membuat form (salah satu aplikasi Acces) yang dapat mengakomodasi keperluan tersebut. Jika Atasan Anda meminta laporan penjualan kepada anda, tentunya anda akan segera menyiapkannya anda bisa mencetak table penjualan sebagai laporan tetapi hasil cetakannya tidak seperti laporan yang diinginkan. Cara yang terbaik adalah dengan membuat Laporan Penjualan tersebut dengan menggunakan Report (salah satu aplikasi pada Microsoft Acces).

Tetapi tidak terbatas pada aplikasi diatas, kita juga dapat membuat aplikasi Microsoft Acces sesuai dengan kebutuhan yang perlukan misalnya database mahasiswa di kelas, atau database lainnya.

### **2.2.9. Balsamiq Mockup**

Balsamiq Mockups adalah salah satu software yang digunakan dalam pembuatan desain atau prototyping dalam pembuatan tampilan user interface sebuah aplikasi. Dengan menggunakan Balsamiq Mockup kita dimudahkan dalam pembuatan user interface karena Balsamiq Mockup sudah menyediakan tools yang dapat memudahkan dalam membuat desain prototyping aplikasi yang akan kita buat.

Software ini berfokus pada konten yang ingin digambar dan fungsionalitas yang dibutuhkan oleh pengguna.

Beberapa alasan kami menggunakan aplikasi ini adalah :

1. Low-Fi Sketch Wireframes:

Samar, rendah-fidelity wireframes membiarkan Anda berfokus percakapan desain pada fungsi

1. Komponen UI & Icon:

75 komponen built-in antarmuka pengguna dan 187 ikon, ditambah seluruh banyak komponen yang dihasilkan.

1. Click-Through Prototipe:

Menghubungkan memungkinkan Anda menghasilkan klik melalui prototipe untuk demo & pengujian kegunaan

1. Ekspor ke PNG atau PDF:

Saham atau hadir maket dengan menggunakan link tertanam ekspor PDF, atau menggunakan alat pihak ke-3 untuk ekspor ke kode.

Balsamiq merupakan aplikasi yang disediakan untuk para designer guna mendesign mockups, dimana mockups itu menurut wiki adalah sebagai sebuah model dari suatu struktur atau alat baik full size ataupun berupa miniatur yang digunakan untuk pembelajaran, demo, test desain, promosi, dsb.

Kelebihan Balsamiq Mockups dibanding software pembuat mockup lainnya adalah aplikasi ini berbasis cloud, disertai aplikasi desktop yang memungkinkan kita dengan cepat dan mudah membuat rancangan website. Dengan konten yang terbuat seperti dari gambaran tangan, akan membuat kita fokus pada pemecahan masalah user interface yang lebih besar, daripada pada perincian website. Di websitenya sendiri ada dua pilihan untuk para pengguna, ada versi trial for dekstop dan ada juga yang bisa kita download untuk versi dekstop. Namun ada juga yang disediakan dalam versi berbayar. Aplikasi ini bisa digunakan untuk sistem operasi Windows, Mac OS, dan Linux.

### **2.2.10. Unified Modeling Language (UML)**

“Unified Modeling Language (UML) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek”(Rosa & Shalahudin, 2018)

1. Use Case Diagram

“Usecase diagram merupakan pemodelan untuk melakukan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat secara kasar yang digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siaga siapa saja yang berhak menggunakan fungsi-fungsi itu” (Rosa & Shalahudin, 2018)

Simbol Simbol yang digunakan pada use case diagram bisa dilihat pada tabel 2.1 Simbol – Simbol Use Case.

1. Activity Diagram

“Activity Diagram menggambarkan workflow atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor jadi aktivitas yang dapat dilakukan oleh sistem” (Rosa & Shalahudin, 2018)

Simbol Simbol yang digunakan pada Activty diagram bisa dilihat pada tabel 2.2 Simbol – Simbol Activity Diagram.

Tabel 2.1 Simbol – Simbol Use Case

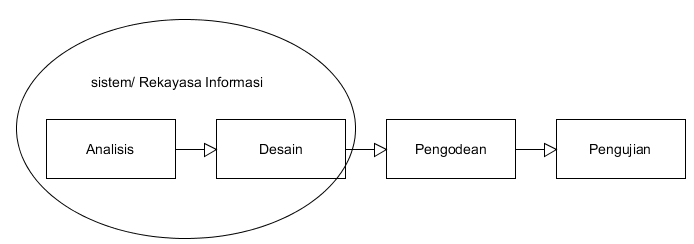
|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **Simbol** | **Nama** | **Keterangan** |
| 1 |  | Actor | Menspesifikasikan himpuan peran yang pengguna mainkan ketika berinteraksi dengan use case. |
| 2 |  | Use Case | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor |
| 3 |  | Association | Apa yang menghubungkan antara objek satu dengan objek lainnya. |
| 4 |  | Extend | Menspesifikasikan bahwa use case target memperluas perilaku dari use case sumber pada suatu titik yang diberikan |
| 5 |  | Generalization | Hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (ancestor). |
| 6 |  | Include | Menspesifikasikan bahwa use case sumber secara eksplisit. |

Tabel 2.2 Simbol – Simbol Activity Diagram

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **Simbol** | **Nama** | **Keterangan** |
| 1 |  | Status Awal | Aktivitas yang dilakukan sistem,aktivitas biasanya diawali dengan kata kerja |
| 2 |  | Aktivitas | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor |
| 3 |  | Percabangan | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu |
| 4 |  | Penggabungan | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu |
| 5 |  | Status Akhir | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir |

### **2.2.11. Model Waterfall**

Model SDLC air terjun (waterfall) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau berurut dimulai dari analisis, desain, pengodean, pengujian dan tahap pendukung (*support*)(Rosa & Shalahudin, 2018). Berikut ini adalah gambar model air terjun:



Gambar 2.4 Ilustrasi Model Waterfall

1. Analisis Kebutuhan Perangkat Lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user.* Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

1. Desain

Desain Perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuuk struktur data, arsitektur perangkat lunak, representasi antar muka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat di implementasikan menjadi program pada tahap selanjutnya, desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikaan.

1. Pembuatan Kode Program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

1. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi lojic dan fungsionalitas dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

1. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirim ke *user.* Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru.

Dari kenyataan yang terjadi sangat jarang model air terjun dapat dilakukan sesuai alurnya karena sebab berikut :

1. Perubahan spesifikasi perangkat lunak terjadi ditengah alur pengembangan.
2. Sangat sulit bagi pelanggan untuk mendefiniskan semua spesifikasi di awal alur pengembangan. Pelanggan sering kali butuh contoh (*prototype*) untuk menjabarkan spesifikasi kebutuhan sistem lebih lanjut.
3. Pelanggan tidak mungkin bersabar mengakomodasi perbuhan yang diperlukan di akhir alur pengembangan.

Dengan berbagai kelemahan yang dimiliki model air terjun tapi model ini telah menjadi dasar model-model yang lain dalam melakukan perbaikan model pengembangan perangkat lunak.

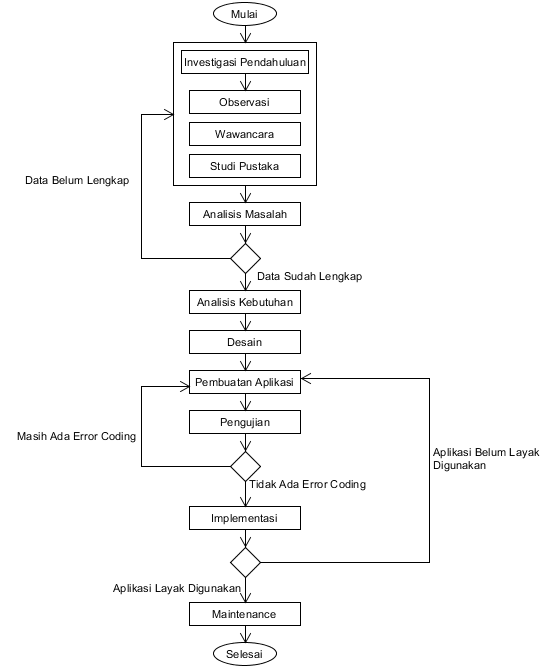
Model air terjun sangat cocok digunakan kebutuhan pelanggan sudah sangat dipahami dan kemukingkan terjadinya perubahan kebutuhan selama pengembangan perangkat lunak kecil. Hal positif dari model air terjun adalah struktur tahap pengembangan sistem jelas, dokumentasi dihasilkan di setiap tahap pengembangan, dan sebuah tahap dijalankan setelah tahap sebelumnya selesai dijalankan (tidak ada tumpang tindih pelaksanaan tahap).

# **BAB III METODOLOGI PENELITIAN**



## **Kerangka Pikir**

Berikut adalah langkah-langkah yang dilakukan untuk mencapai tujuan dari penelitian ini.



Gambar 3.1 Kerangka Pikir

## **Deskripsi**



### **Investigasi Pendahuluan**

Investigasi dilakukan di SDN Mekarsari yang beralamat di R.A.A Wiranatakusumah Kelurahan Baleendah Kecamatan Baleendah Kabupaten Bandung. Dalam hal investigasi penulis melakukan 3 metode berikut adalah uraiannya:

1. Observasi

Observasi dilakukan dengan mengamati sistem yang sedang berjalan saat ini bertujuan untuk mengetahui secara langsung bagaimana proses pengelolaan data surat tugas dan SPPD yang sedang berjalan.

Pada tahap observasi peneliti mengamati proses pada saat pembuatan surat, penyimpanan berkas, pencatatan surat pada buku surat, dan perekapan surat masih yang masih menggunakan pencatatan tertulis dimana sangat rentan mengalami kerusakan pada dokumen tersebut.

1. Wawancara

Metode ini digunakan sebagai pengumpulan data dengan cara mengajukan berbagai pertanyaan kepada pihak Sekolah yang berhubungan dengan kegiatan pembuatan aplikasi. Penulis melakukan interview kepada Ibu Komalasari, S.Pd., M.Pd selaku Kepala Sekolah SDN Mekarsari Perolehan dari wawancara adalah penjelasan detail mengenai proses pembuatan, pengearsipan dan pelaporan Surat Tugas dan Surat Perintah Perjalanan Dinas (SPPD).

Selain penjelasan detail, penulis juga mendapatkan beberapa permasalahan yang sering terjadi pada proses pengelolaan surat diantaranya adalah:

1. Sulitnya pencarian data pegawai yang akan melakukan perjalanan dinas, kesalahan-kesalahan pengetikan dalam menginputkan data sehingga pembuatan dilakukan berulang – ulang.
2. Untuk proses pengarsipannya membutuhkan buku yang tebal untuk mencatat. Buku yang rawan rusak dan hilang, dan terkadang lupa dalam mencatat data surat ke buku. Selain itu juga butuh waktu untuk mencari data surat yang agak lama
3. Proses pembuatan laporannya membutuhkan waktu lama karena harus mengetik ulang dan melihat dari buku surat dan SPJ BOS.
4. Studi pustaka

Penulis mengumpulkan acuan yang diperoleh dari jurnal, buku, buku digital (*ebook)*, dan beberapa Peraturan Pemerintah yang berhubungan dengan masalah yang ada di SD Negeri Mekarsari Baleendah. Disini penulis membaca 3 jurnal yang berelevasi dengan topik penelitian dan mengambil 1 jurnal dan merangkum jurnal tersebut.

### **Analisis Masalah**

Dari fakta-fakta yang sudah di temukan maka tahap selanjutnya adalah melakukan analisis masalah. Berikut adalah hasil dari analisis masalah yang di lakukan :

1. Sulitnya pencarian data pegawai yang akan melakukan perjalanan dinas, kesalahan-kesalahan pengetikan dalam menginputkan data sehingga pembuatan dilakukan berulang – ulang. dan juga tiap membuat surat baru harus melakukan pengeditan.
2. Untuk proses pengarsipannya membutuhkan buku yang tebal untuk mencatat. Buku yang rawan rusak dan hilang, dan terkadang lupa dalam mencatat data surat ke buku. Selain itu juga butuh waktu untuk mencari data surat yang agak lama.
3. Proses pembuatan laporannya membutuhkan waktu lama karena harus mengetik ulang dan melihat dari buku surat dan SPJ BOS.

Berdasarkan hasil dari analisis masalah yang di lakukan maka di lakukanlah studi banding dengan pihak yang lebih berpengalaman, yaitu kepada bapak Yaya Suharya, S.Kom., MT., dandi sarankan untuk membuat aplikasi yang mampu mengelola pembuatan surat tugas dan SPPD serta mengatasi masalah-masalah diatas.

Jika di dalam tahap ini masih ada kekurangan maka akan kembali ke tahap investigasi awal sampai mendapat hasil yang tepat, kemudian di lanjutkan ketahap berikutnya.

### **Perencanaan kebutuhan**

* **Analisis Kebutuhan Fungsional Sistem**

Kebutuhan fungsional merupakan kebutuhan system dari segi fungsionalitas yang akan muncul pada system yang dirancang. Adapun beberapa kebutuhan fungsional yaitu :

1. Mengolah data sekolah

2. Mengolah data pegawai

3. Mengolah data surat

4. Mencetak surat

5. Mencetak laporan

* **Analisis Kebutuhan Non Fungsional Sistem**

Analisis kebutuhan merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan aplikasi. Spesifikasi ini juga juga meliput semua elemen dan komponen yang dibutuhkan untuk aplikasi yang akan dibuat, sampai dengan aplikasi tersebut diimplementasi. Kebutuhan aplikasi terbagi menjadi beberapa analisis, yaitu sebagai berikut :

1. Analisis Perangkat Keras (Hardware)

Spesifikasi perangkat keras yang digunakan untuk pembuatan aplikasi surat tersebut adalah satu unit laptop dengan spesifikasi sebagai berikut:

* 1. *Processor AMD A12-9720P Radeon R7, 12 Compute Cores 4C+8G 2.70 GHz*
  2. *VGA AMD Radeon R7, Shared Memory 4277 MB*
  3. System Type 64-bit Operating System, x64-based processor
  4. Memory berkapasitas 8 GB
  5. Printer Epson L120

1. Analisis Perangkat Lunak (Software)

Perangkat lunak yang digunakan untuk membuat aplikasi surat tersebut adalah sebagai berikut :

1. Sistem Operasi Window 10

Perangkat lunak yang digunakan oleh laptop untuk mengatur sumber daya dari perangkat keras dan perangkat lunak

1. Embarcadero® Delphi XE8

Aplikasi yang digunakan untuk pembuatan suatu aplikasi

1. Microsoft Access 2010

Aplikasi yang digunakan untuk pembuatan dan penyimpanan *database*

1. Balsamic Mockup

Aplikasi untuk membuat desain antarmuka suatu aplikasi

### **Desain**

Pada tahap ini ada dua jenis desain yang dibuat yaitu:

* + - * 1. Desain Sistem

Pada tahap ini perancangan di lakukan dengan membuat *Unified Modeling Language* (UML). Perancangan dengan *Unified Modeling Language* (UML) ini bertujuan untuk merancang aplikasi dengan mudah dalam proses pengembangan dan juga untuk memvisualisasikan, membangun dan mendokumentasikan dari sebuah sistem pembangunan perangkat lunak berbasis objek, maka dibuatlah:

* 1. Use Case untuk penyimpanan dokumen dan melengkapi indeks pada dokumen.
  2. Class Diagram yaitu diagram untuk menggambarkan struktur dan penjelasan hubungan antar class dalam sebuah sistem yang dibuat.
  3. Activity Diagram yaitu diagram untuk menggambarkan alur aktivitas yang sedang di rancang.
     + - 1. Desain Pemrograman

Desain pemrograman dilakukan dengan membuat desain yang diperlukan untuk pemrograman (penulisan source code program) berdasarkan desain proses yang telah dibuat. Desain pemrograman yang akan diurai pada tahapan ini terdiri dari :

* 1. Desain Database

Perancangan database apliksi ini , terdiri dari :

* + - * 1. Normalisasi database.
        2. Spesifikasi tabel data yang digunakan untuk melihat struktur tabel yang dibuat.

1. Desain Screen Layout

Pada tahapan ini dilakukan perancangan interface yang terdiri dari: tampilan input data dan tampilan output dari keseluruhan system dengan membuat rancangan layar tampilan GUI (Graphical User Interface) yang user friendly.

### **Pembuatan Aplikasi**

Pada pembuatan aplikasi digunakan perangkat lunak dan bahasa pemrograman sebagai berikut:

1. Embarcadero® Delphi XE8 adalah *text editor* yang digunakan untuk pembuatan aplikasi.
2. Microsoft Access 2010, digunakan untuk penyimpanan *database.*
   * 1. **Pengujian Aplikasi**

Pada tahapan ini dilakukan pengujian aplikasi yang telah dibuat untuk memastikan apakah hasil aplikasi sudah sesuai dengan rancangan yang diharapkan. Jika masih ada kekurangan atau kesalahan maka kembali ke tahap pembuatan aplikasi untuk diperbaiki sampai benar-benar sesuai rancangan.

### **Implementasi**

Pada tahap ini, aplikasi akan mulai diimplementasikan untuk pembutan surat tugas dan SPPD di SD Negeri Mekarsari Baleendah. Tahap ini juga dapat disebut tahap evaluasi untuk pengembangan aplikasi yang sudah jadi supaya menjadi lebih baik. Hasil evaluasi ini dapat digunakan sebagai masukan untuk tahap konsep pada aplikasi selanjutnya.

### **Pemeliharaan (*maintenance)***

Untuk tahap pemeliharaan (*maintenance*) penulis hanya melakukan maintenance mulai dari pembuatan aplikasi sampai aplikasi digunakan di sekolah. Selanjutnya pemeliharaan (maintenance) akan menjadi tugas operator sekolah masing-masing.

# **BAB IV** **ANALISIS DAN PERANCANGAN**

## **Analisis**

### **Analisis Masalah**

Berdasarkan analisis yang dilakukan penulis di SDN Mekarsari Baleendah, menemui kendala yaitu :

* + 1. Sulitnya pencarian data pegawai yang akan melakukan perjalanan dinas,
    2. Kesalahan-kesalahan pengetikan dalam menginputkan data surat.
    3. pengarsipannya membutuhkan buku yang tebal untuk mencatat. Buku yang rawan rusak dan hilang.
    4. Proses pembuatan laporannya membutuhkan waktu’lama

Untuk itu penulis dalam penelitian ini akan mengusulkan membuat aplikasi surat tugas dan SPPD untuk mengatasi masalah-masalah diatas sehingga waktu yang diperlukan menjadi efisien dan laporan tersimpan di media elektronik dengan klasifikasi.

### **Analisis *Software***

Beberapa *software* yang penulis gunakan untuk membuat aplikasi ini adalah sebagai berikut :

Tabel 4.1 Analisis Software

|  |  |  |
| --- | --- | --- |
| No | *Software* yang digunakan | Fungsi |
| 1 | Windows 10 | Sistem operasi |
| 2 | Balsamic Mockup | Digunakan untuk membuat interface aplikasi |
| 3 | Embarcadero Delphi XE8 | Digunakan untuk membuatan aplikasi |
| 4 | Microsoft Access 2010 | Digunakan sebagai database |
| 5 | Microsoft Word 2010 | Digunakan untuk membuat laporan |



### **Analisis Pengguna**

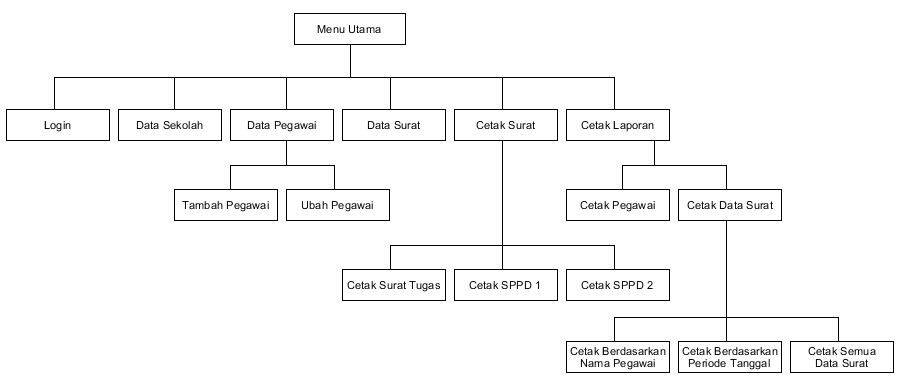
Pengguna yang akan memakai aplikasi ini adalah tenaga administrasi sekolah yang ditugaskan untuk mengelola semua administrasi sekolah termasuk pembuatan surat tugas dan SPPD sekolah. Pengguna ini nantinya

disebut admin yang akan melakukan aktifitas tambah, ubah hapus data dan melakukan cetak laporan rekap surat.

### **User Interface**

*User interface* dari aplikasi ini dibuat *user friendly* artinya dalam penggunaannya *user* akan dimudahkan dari sisi penempatan-penempatan konten yang efisien, sehingga lebih menonjolkan isi dari informasi yang akan ditampilkan oleh setiap konten. Karena aplikasi yang di buat di khusus kan untuk kemudahan pengguna saat melakukan pembuatan surat dan pencetakan surat untuk memberikan kemudahan lebih kepada sekolah untuk pembuatan surat dan laporan rekap surat. Untuk merancang desain user interface akan digunakan aplikasi balsamic mockup dengan susunan halaman sebagai berikut

* + 1. Menu Utama
    2. Login
    3. Data Sekolah
    4. Data Pegawai
    5. Data Surat
    6. Cetak Surat
    7. Cetak Laporan



Gambar 4.1 Struktur Rancangan *User Interface*

### **Fitur-fitur**

Fitur-fitur yang dibuat dalam aplikasi ini dimaksudkan untuk para pengguna agar lebih mudah mengelola sebuah data. Berikut adalah fitur-fitur yang disediakan oleh aplikasi ini :

1. Pengelolaan data Sekolah. Proses ini dikelola aplikasi untuk penginputan dan pengeditan data sekolah.
2. Pengelolaan data Pegawai. Proses ini dikelola aplikasi untuk penginputan, pengeditan, penghapusan data pegawai
3. pengelolaan data Surat. Proses ini dikelola aplikasi untuk penginputan dan penghapusan data surat.
4. Cetak surat tugas dan surat perintah perjalanan dinas (SPPD).
5. Cetak laporan adalah fitur untuk mencetak laporan.

### **Analisis Data**

Berikut adalah analisis data berupa data masukkan, proses, dan keluaran yang menunjang aplikasi ini.

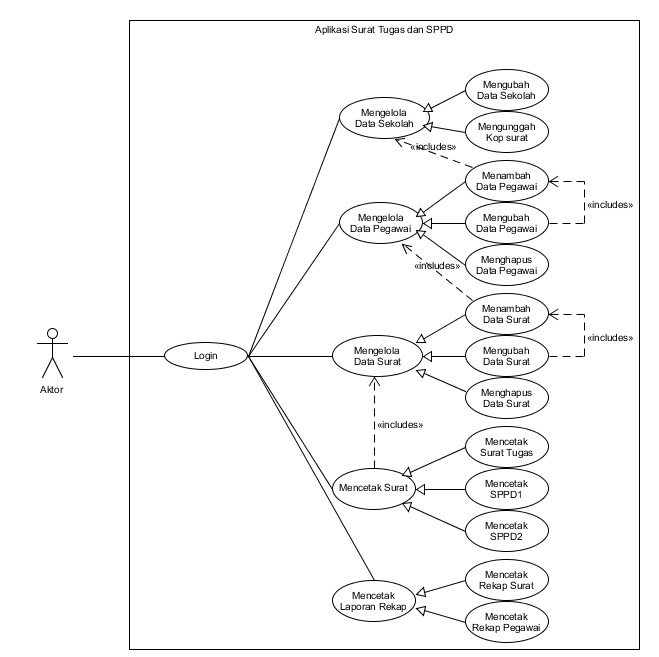
|  |  |  |
| --- | --- | --- |
| Masukan | Proses | Keluaran |
| Pengelolaan data sekolah | Proses pengelolaan data sekolah berupa penginputan dan pengeditan data sekolah oleh pengguna. Selain data sekolah, penguna juga bisa memasukan KOP surat sesuai dengan kebutuhan pengguna | Informasi data sekolah dan kop surat yang tersimpan di database |
| Pengelolaan data pegawai | Proses pengelolaan data Pegawai berupa pengimputan, pengeditan, dan penghapusan data pegawai oleh pengguna | Informasi data pegawai yang tersimpan di database |
| Data surat | Proses pengelolaan data surat berupa pengimputan, dan penghapusan data surat oleh pengguna | Informasi data surat yang tersimpan di database |
| Cetak surat tugas dan SPPD | Proses cetak surat tugas dan SPPD yang di pilih oleh pengguna. | Surat tugas dan SPPD untuk di cetak. |
| Cetak laporan | Proses cetak rekap laporan data sekolah, data pegawai dan data surat. | Laporan rekap data sekolah, pegawai, dan surat |

Tabel 4.2 Analisis data aplikiasi yang akan di buat

## **Perancangan**

### ***Use Case Diagram***

*Use case diagram* ini dibuat untuk menunujukkan fungsionalitas utama dari setiap level admin pada aplikasi yang digambarkan dengan aktor. Dalam hal ini admin melakukan input data, edit data, hapus data, cetak surat dan cetak laporan.



Gambar 4.2 *Use Case* Aplikasi

* + 1. **Definisi Aktor**

Berikut adalah deskripsi pendefinisian aktor pada aplikasi ini :

Tabel 4.3 Tabel Definisi Aktor

|  |  |  |
| --- | --- | --- |
| **No** | **Aktor** | **Deskripsi** |
| 1 | Aktor | Orang yang bertugas dan memiliki hak akses untuk melakukan operasi pengelolaan data sekolah, data pegawai, data surat serta pencetakan surat dan laporan rekap.  . |

* + 1. **Definisi *Use Case***

Berikut adalah deskripsi pendefinisisan use *case* pada aplikasi ini :

Tabel 4.4 Tabel Definisi *Use Case*

|  |  |  |
| --- | --- | --- |
| **No** | **Aktor** | **Deskripsi** |
| 1 | Login | Merupakan proses untuk melakukan login aktor |
| 2 | Mengelola data sekolah | Mengelola data sekolah merupakan proses generalisasi yang meliputi dua buah proses pengelolaan data sekolah yaitu Mengubah data dan mengunggah kop surat  Diperlukan proses login terlebih dahulu untuk bisa menjalankan proses ini. |
| 3 | Mengubah Data sekolah | Proses mengubah data sekolah yang ada di *database* |
| 4 | Mengunggah kop surat | Proses mengunggah kop surat dari komputer/laptop pengguna ke *database* |
| 5 | Mengelola data pegawai | Mengelola data pegawai merupakan proses generalisasi yang meliputi tiga buah proses pengelolaan data pegawai yaitu menambah data, mengubah data dan menghapus data pegawai  Diperlukan proses login terlebih dahulu untuk bisa menjalankan proses ini. |
| 6 | Menambah data pegawai | Proses menambah data pegawai ke *database*.  Memerlukan minimal satu buah *record* data sekolah yang tersimpan di *database.* |
| 7 | Mengubah data pegawai | Proses mengubah data pegawai yang ada di *database.*  Memerlukan minimal satu buah *record* data pegawai yang tersimpan di *database.*  Dalam proses ini, untuk NIP/NUPTK/ID tidak bisa diubah karena data tersebut hanya di buat satu kali pada saat proses tambah data |
| 8 | Menghapus data pegawai | Proses menghapus data pegawai yang ada di *database*. |
| 9 | Mengelola data pegawai | Mengelola data surat merupakan proses generalisasi yang meliputi tiga buah proses pengelolaan data surat yaitu menambah data, mengubah data dan menghapus data surat  Diperlukan proses login terlebih dahulu untuk bisa menjalankan proses ini. |
| 10 | Menambah data surat | Proses menambah data surat ke *database*.  Memerlukan minimal satu buah *record* data pegawai yang tersimpan di *database.* |
| 11 | Mengubah data surat | Proses mengubah data surat yang ada di *database.*  Memerlukan minimal satu buah *record* data surat yang tersimpan di *database.*  Dalam proses ini, untuk NoSurat tidak bisa diubah karena data tersebut hanya di buat satu kali pada saat proses tambah data |
| 12 | Menghapus data surat | Proses menghapus data surat yang ada di *database*. |
| 13 | Mencetak Surat | Mencetak surat merupakan proses generalisasi yang meliputi tiga buah proses cetak surat yaitu mencetak surat tugas, mencetak SPPD 1, dan mencetak SPPD 2.  Diperlukan proses login terlebih dahulu untuk bisa menjalankan proses ini. |
| 14 | Mencetak surat tugas | Proses mencetak surat dengan format surat tugas berdasarkan data surat di *database*  Memerlukan minimal satu buah *record* data surat yang tersimpan di *database.* |
| 15 | Mencetak SPPD 1 | Proses mencetak surat dengan format SPPD Halaman 1 berdasarkan data surat di *database*  Memerlukan minimal satu buah *record* data surat yang tersimpan di *database* |
| 15 | Mencetak SPPD 2 | Proses mencetak surat dengan format SPPD Halaman 2 berdasarkan data surat di *database*  Memerlukan minimal satu buah *record* data surat yang tersimpan di *database* |
| 16 | Mencetak laporan rekap | Mencetak laporan rekap merupakan proses generalisasi yang meliputi dua buah proses cetak laporan rekap yaitu mencetak data pegawai dan mencetak data surat  Diperlukan proses login terlebih dahulu untuk bisa menjalankan proses ini. |
| 17 | Mencetak rekap data pegawai | Proses mencetak laporan rekap data pegawai di *database*  Memerlukan minimal satu buah *record* data pegawai yang tersimpan di *database.* |
| 18 | Mencetak rekap data surat | Proses mencetak laporan rekap data surat di *database*  Memerlukan minimal satu buah *record* data surat yang tersimpan di *database.* |
| 19 | Cetak berdasarkan nama pegawai | Proses mencetak laporan rekap data surat di *database* berdasarkan nama pegawai |
| 20 | Cetak berdasarkan periode tanggal | Proses mencetak laporan rekap data surat di *database* berdasarkan periode tanggal |
| 21 | Cetak semua | Proses mencetak laporan rekap semua data surat di *database* |

* + 1. **Skenario *Use Case***

Berikut adalah deskripsi skenario use *case* pada aplikasi ini :

1. *Login*

Tabel 4.5 Skenario *Login*

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Skenario Normal** | |
| 1. Memilih menu *login* |  |
|  | 2. Menampilkan *form login* |
| 3. Memasukan nama pengguna dan kata sandi |  |
|  | 4. memeriksa terdaftar tidaknya data masukan dengan memeriksa tabel pengguna |
|  | 5. kembali ke *form* menu utama dan mengaktifkan fitur menu utama |
| **Skenario alternatif** | |
| 1. Memilih menu *login* |  |
|  | 2. Menampilkan *form login* |
| 3. Memasukan nama pengguna dan kata sandi |  |
|  | 4. memeriksa terdaftar tidaknya data masukan dengan memeriksa tabel pengguna |
|  | 5. menamplikan pesan pengguna dan kata sandi tidak terdaftar |
| 6. Memasukan nama pengguna dan kata sandi yang terdaftar |  |
|  | 6. memeriksa terdaftar tidaknya data masukan dengan memeriksa tabel pengguna |
|  | 7. kembali ke *form* menu utama dan mengaktifkan fitur menu utama |

1. Mengubah data sekolah

Tabel 4.6 SkenarioUbah Data Sekolah

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Skenario Normal** | |
| 1. Memilih menu data sekolah |  |
|  | 2. Menampilkan *form* data sekolah |
| 2. Menekan tombol ubah |  |
|  | 3. Mengaktifkan fitur ubah data sekolah |
| 4. Mengubah data sekolah dengan mengisi semua kolom yang wajib diisi |  |
|  | 5. Memerikasa lengkaptidaknya data masukan |
|  | 6. Menyimpan data sekolah yang telah diubah ke *database* |
|  | 7. Menampilkan pesan bahwa data berhasil disimpan |
| **Skenario Alternatif** | |
| 1. Memilih menu data sekolah |  |
|  | 2. Menampilkan *form* data sekolah |
| 2. Menekan tombol ubah |  |
|  | 3. Mengaktifkan fitur ubah data sekolah |
| 4. Mengubah data sekolah tanpa mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) |  |
|  | 5. Memeriksa lengkaptidaknya data masukan |
|  | 6. Menampilkan pesan bahwa data yang diubah tidak lengkap |
| 7. Mengubah data sekolah dengan mengisi semua kolom yang wajib diisi |  |
|  | 8. Memeriksa lengkaptidaknya data masukan |
|  | 9. Menyimpan data sekolah yang telah diubah ke *database* |
|  | 10. Menampilkan pesan bahwa data berhasil disimpan |

1. Mengunggah kop surat

Tabel 4.7 Skenario Unggah Kop Surat

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Skenario Normal** | |
| 1. Memilih Menu data sekolah |  |
|  | 2. Menampilkan *form* data sekolah |
| 2. Menekan tombol ubah |  |
|  | 3. Mengaktifkan fitur ubah data sekolah |
| 4. Menekan tombol unggah kop surat |  |
|  | 5. Menampilkan *open picture dialog* |
| 6. Memilih gambar kop surat di komputer/laptop |  |
|  | 7. Menyimpan data sekolah ke *database* |
|  | 8. Menampilkan pesan bahwa data berhasil disimpan |

1. Menambah data pegawai

Tabel 4.8 Skenario Tambah Data Pegawai

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Skenario Normal** | |
| 1. Memilih menu data pegawai |  |
|  | 2. Menampilkan *form* data pegawai |
| 3. Menekan tombol tambah |  |
|  | 4. Menampilkan *form* tambah data pegawai |
| 5. Memasukan data pegawai dengan mengisi semua kolom yang wajib diisi |  |
|  | 6. Memerikasa lengkaptidaknya data masukan |
|  | 7. Menyimpan data pegawai ke *database* |
|  | 8. Menampilkan pesan bahwa data berhasil disimpan |
| **Skenario Alternatif** | |
| 1. Memilih menu data pegawai |  |
|  | 2. Menampilkan *form* data pegawai |
| 3. Menekan tombol tambah |  |
|  | 4. Menampilkan *form* tambah data pegawai |
| 5. Memasukan data sekolah tanpa mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) |  |
|  | 6. Memerikasa lengkaptidaknya data masukan |
|  | 7. Menampilkan pesan bahwa data yang dimasukan tidak lengkap |
| 8. Memasukan data pegawai dengan mengisi semua kolom yang wajib diisi |  |
|  | 9. Memeriksa lengkaptidaknya data masukan |
|  | 10. Menyimpan data pegawai ke *database* |
|  | 11. Menampilkan pesan bahwa data berhasil disimpan |

1. Mengubah data pegawai

Tabel 4.9 Skenario Ubah Data Pegawai

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Skenario Normal** | |
| 1. Memilih menu data pegawai |  |
|  | 2. Menampilkan *form* data pegawai |
| 3. Memilih data pegawai yang akan diubah |  |
| 4. Menekan tombol ubah |  |
|  | 5. Menampilkan semua kolom data pegawai yang akan diubah pada *form* udah data pegawai |
| 6. Mengubah data pegawai dengan mengisi semua kolom yang wajib diisi |  |
|  | 8. Memerikasa lengkaptidaknya data masukan |
|  | 9. Menyimpan data pegawai yang telah diubah ke *database* |
|  | 10. Menampilkan pesan bahwa data berhasil disimpan |
| **Skenario Alternatif** | |
| 1. Memilih menu data pegawai |  |
|  | 2. Menampilkan *form* data pegawai |
| 3. Memilih data pegawai yang akan diubah |  |
| 4. Menekan tombol ubah |  |
|  | 5. Menampilkan semua kolom data pegawai yang akan diubah pada *form* udah data pegawai |
| 6. Mengubah data sekolah tanpa mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) |  |
|  | 7. Memerikasa lengkaptidaknya data masukan |
|  | 8. Menampilkan pesan bahwa data yang diubah tidak lengkap |
| 9. Mengubah data pegawai dengan mengisi semua kolom yang wajib diisi |  |
|  | 10. Memeriksa lengkaptidaknya data masukan |
|  | 11. Menyimpan data sekolah yang telah diubah ke *database* |
|  | 12. Menampilkan pesan bahwa data berhasil disimpan |

1. Menghapus data pegawai

Tabel 4.10 Skenario Hapus Data Pegawai

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Skenario Normal** | |
| 1. Memilih menu data pegawai |  |
|  | 2. Menampilkan *form* data pegawai |
| 3. Memilih data pegawai yang akan dihapus |  |
| 4. Menekan tombol hapus |  |
|  | 5. Menampilkan pesan konfirmasi bahwa apakah anda yakin menghapus data pegawai yang dipilih |
| 6. Memilih pilihan “YA” saat muncul konfirmasi hapus data |  |
|  | 7. Menghapus data pegawai dari *database* |
|  | 8. Menampilkan pesan bahwa data terpilih telah dihapus |
| **Skenario Alternatif** | |
| 1. Memilih menu data pegawai |  |
|  | 2. Menampilkan *form* data pegawai |
| 3. Memilih data pegawai yang akan dihapus |  |
| 4. Menekan tombol hapus |  |
|  | 5. Menampilkan pesan konfirmasi bahwa apakah anda yakin menghapus data pegawai yang dipilih |
| 6. Memilih pilihan “TIDAK” saat muncul konfirmasi hapus data |  |
|  | 7. Kembali ke *form* data pegawai |

1. Menambah data surat

Tabel 4.11 Skenario Tambah Data Surat

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Skenario Normal** | |
| 1. Memilih menu data surat |  |
|  | 2. Menampilkan *form* data surat |
| 3. Menekan tombol tambah |  |
|  | 4. Mengaktifkan fitur tambah data surat |
| 5. Memasukan data surat dengan mengisi semua kolom yang wajib diisi |  |
|  | 6. Memerikasa lengkaptidaknya data masukan |
|  | 7. Menyimpan data surat ke *database* |
|  | 8. Menampilkan pesan bahwa data berhasil disimpan |
| **Skenario Alternatif** | |
| 1. Memilih menu data surat |  |
|  | 2. Menampilkan *form* data surat |
| 3. Menekan tombol tambah |  |
|  | 4. Mengaktifkan fitur tambah data surat |
| 5. Memasukan data surat tanpa mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) |  |
|  | 6. Memerikasa lengkaptidaknya data masukan |
|  | 7. Menampilkan pesan bahwa data yang dimasukan tidak lengkap |
| 8. Memasukan data surat dengan mengisi semua kolom yang wajib diisi |  |
|  | 9. Memeriksa lengkaptidaknya data masukan |
|  | 10. Menyimpan data surat ke *database* |
|  | 11. Menampilkan pesan bahwa data berhasil disimpan |

1. Mengubah data surat

Tabel 4.12 Skenario Ubah Data Surat

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Skenario Normal** | |
| 1. Memilih menu data surat |  |
|  | 2. Menampilkan *form* data surat |
| 3. Memilih data surat yang akan diubah |  |
| 4. Menekan tombol ubah |  |
|  | 5. Menampilkan semua data surat yang akan diubah pada kolom yang tersedia |
| 6. Mengubah data surat dengan mengisi semua kolom yang wajib diisi |  |
|  | 8. Memerikasa lengkaptidaknya data masukan |
|  | 9. Menyimpan data surat yang telah diubah ke *database* |
|  | 10. Menampilkan pesan bahwa data berhasil disimpan |
| **Skenario Alternatif** | |
| 1. Memilih menu data surat |  |
|  | 2. Menampilkan *form* data surat |
| 3. Memilih data surat yang akan diubah |  |
| 4. Menekan tombol ubah |  |
|  | 5. Menampilkan semua data surat yang akan diubah pada kolom yang tersedia |
| 6. Mengubah data surat tanpa mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) |  |
|  | 8. Memerikasa lengkaptidaknya data masukan |
|  | 8. Menampilkan pesan bahwa data yang diubah tidak lengkap |
| 9. Mengubah data surat dengan mengisi semua kolom yang wajib diisi |  |
|  | 10. Memeriksa lengkaptidaknya data masukan |
|  | 11. Menyimpan data surat yang telah diubah ke *database* |
|  | 12. Menampilkan pesan bahwa data berhasil disimpan |

1. Menghapus data surat

Tabel 4.13 Skenario Hapus Data Surat

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Skenario Normal** | |
| 1. Memilih menu data surat |  |
|  | 2. Menampilkan *form* data surat |
| 3. Memilih data surat yang akan dihapus |  |
| 4. Menekan tombol hapus |  |
|  | 5. Menampilkan pesan konfirmasi bahwa apakah anda yakin menghapus data surat yang dipilih |
| 6. Memilih pilihan “YA” saat muncul konfirmasi hapus data |  |
|  | 7. Menghapus data surat dari *database* |
|  | 8. Menampilkan pesan bahwa data terpilih telah dihapus |
| **Skenario Alternatif** | |
| 1. Memilih menu data surat |  |
|  | 2. Menampilkan *form* data surat |
| 3. Memilih data surat yang akan dihapus |  |
| 4. Menekan tombol hapus |  |
|  | 5. Menampilkan pesan konfirmasi bahwa apakah anda yakin menghapus data surat yang dipilih |
| 6. Memilih pilihan “TIDAK” saat muncul konfirmasi hapus data |  |
|  | 7. Kembali ke *form* data surat |

1. Mencetak surat tugas

Tabel 4.14 Skenario Cetak Surat Tugas

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Skenario Normal** | |
| 1. Memilih menu cetak surat |  |
|  | 2. Menampilkan *form* cetak surat |
| 3. Memilih data surat yang akan cetak |  |
| 4. Menekan tombol cetak surat tugas |  |
|  | 5. Menampilkan format surat tugas dengan isi berdasarkan data surat yang dipilih |

1. Mencetak SPPD 1

Tabel 4.15 Skenario Cetak SPPD 1

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Skenario Normal** | |
| 1. Memilih menu cetak surat |  |
|  | 2. Menampilkan *form* cetak surat |
| 3. Memilih data surat yang akan cetak |  |
| 4. Menekan tombol cetak SPPD 1 |  |
|  | 5. Menampilkan format SPPD 1 dengan isi berdasarkan data surat yang dipilih |

1. Mencetak SPPD 2

Tabel 4.16 Skenario Cetak SPPD 2

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Skenario Normal** | |
| 1. Memilih menu cetak surat |  |
|  | 2. Menampilkan *form* cetak surat |
| 3. Memilih data surat yang akan cetak |  |
| 4. Menekan tombol cetak SPPD 2 |  |
|  | 5. Menampilkan format SPPD 2 dengan isi berdasarkan data surat yang dipilih |

1. Mencetak rekap data pegawai

Tabel 4.17 Cetak Rekap Data Pegawai

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Skenario Normal** | |
| 1. Memilih menu cetak laporan |  |
|  | 2. Menampilkan *form* cetak laporan |
| 3. Memilih cetak data pegawai |  |
|  | 4. Menampilkan laporan rekap data pegawai |

1. Cetak berdasarkan nama pegawai

Tabel 4.18 Cetak Berdasarkan Nama Pegawai

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Skenario Normal** | |
| 1. Memilih menu cetak laporan |  |
|  | 2. Menampilkan *form* cetak laporan |
| 3. Memilih cetak data surat |  |
|  | 4. Menampilkan *form* cetak data surat berdasarkan |
| 5. Memilih cetak berdasarkan nama pegawai |  |
| 6. Memilih nama pegawai |  |
|  | 7. Memeriksa tersedia tidaknya riwatyat perjalanan dinas pada pegawai terpilih |
|  | 8. Menampilkan laporan rekap data surat berdasarkan nama pegawai yang dipilih |
| **Skenario Alternatif** | |
| 1. Memilih menu cetak laporan |  |
|  | 2. Menampilkan *form* cetak laporan |
| 3. Memilih cetak data surat |  |
|  | 4. Menampilkan *form* cetak data surat berdasarkan |
| 5. Memilih cetak berdasarkan nama pegawai |  |
| 6. Memilih nama pegawai |  |
|  | 7. Memeriksa tersedia tidaknya riwatyat perjalanan dinas pada pegawai terpilih |
|  | 8. Menampilkan pesan bahwa tiada riwayat pejalanan dinas pada pegawai terpilih |
| 9. Memilih nama pegawai lain yang memiliki riwayat perjalanan dinas |  |
|  | 10. Memeriksa tersedia tidaknya riwatyat perjalanan dinas pada pegawai terpilih |
|  | 11. Menampilkan laporan rekap data surat berdasarkan nama pegawai yang dipilih |

1. Cetak berdasarkan periode tanggal

Tabel 4.19 Cetak berdasarkan periode tanggal

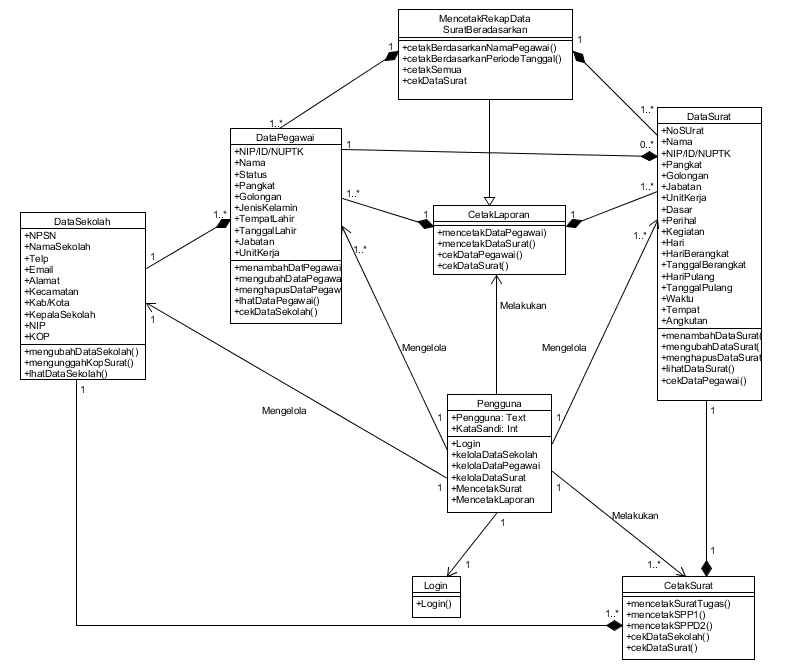
|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Skenario Normal** | |
| 1. Memilih menu cetak laporan |  |
|  | 2. Menampilkan *form* cetak laporan |
| 3. Memilih cetak data surat |  |
|  | 4. Menampilkan *form* cetak data surat berdasarkan |
| 5. Memilih cetak berdasarkan nama tanggal |  |
| 6. Memilih periode tanggal |  |
|  | 7. Memeriksa tersedia tidaknya riwatyat perjalanan dinas pada periode tanggal terpilih |
|  | 8. Menampilkan laporan rekap data surat berdasarkan periode tanggal yang dipilih |
| **Skenario Alternatif** | |
| 1. Memilih menu cetak laporan |  |
|  | 2. Menampilkan *form* cetak laporan |
| 3. Memilih cetak data surat |  |
|  | 4. Menampilkan *form* cetak data surat berdasarkan |
| 5. Memilih cetak berdasarkan nama tanggal |  |
| 6. Memilih periode tanggal |  |
|  | 7. Memeriksa tersedia tidaknya riwatyat perjalanan dinas pada periode tanggal terpilih |
|  | 8. Menampilkan pesan bahwa tidak ada riwayat pejalanan dinas pada periode tanggal terpilih |
| 9. Memilih periode tanggal lain yang memiliki riwayat perjalanan dinas |  |
|  | 10. Memeriksa tersedia tidaknya riwatyat perjalanan dinas pada periode tanggal terpilih |
|  | 11. Menampilkan laporan rekap data surat berdasarkan periode tanggal yang dipilih |

1. Cetak semua data surat

Tabel 4.20 Cetak Semua Data Surat

|  |  |
| --- | --- |
| **Aksi Aktor** | **Reaksi Sistem** |
| **Skenario Normal** | |
| 1. Memilih menu cetak laporan |  |
|  | 2. Menampilkan *form* cetak laporan |
| 3. Memilih cetak data surat |  |
|  | 4. Menampilkan *form* cetak data surat berdasarkan |
| 5. Memilih cetak semua |  |
|  | 6. Memeriksa tersedia tidaknya riwatyat perjalanan dinas di *database* |
|  | 7. Menampilkan laporan rekap semua data surat |
| **Skenario Alternatif** | |
| 1. Memilih menu cetak laporan |  |
|  | 2. Menampilkan *form* cetak laporan |
| 3. Memilih cetak data surat |  |
|  | 4. Menampilkan *form* cetak data surat berdasarkan |
| 5. Memilih cetak semua |  |
|  | 7. Memeriksa tersedia tidaknya riwatyat perjalanan dinas di *database* |
|  | 8. Menampilkan pesan bahwa tidak ada riwayat pejalanan dinas di *database* |
|  | 9. kembali ke *form* cetak berdasarkan |

### ***Class Diagram***

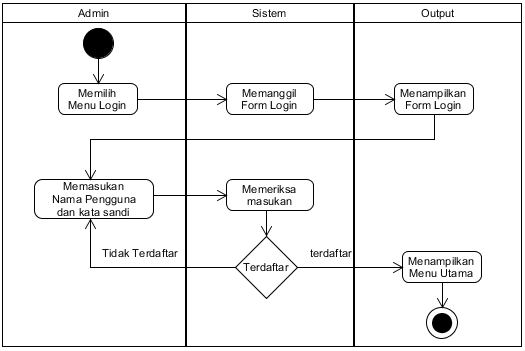


Gambar 4.3 *Class Diagram*

### ***Activity Diagram***

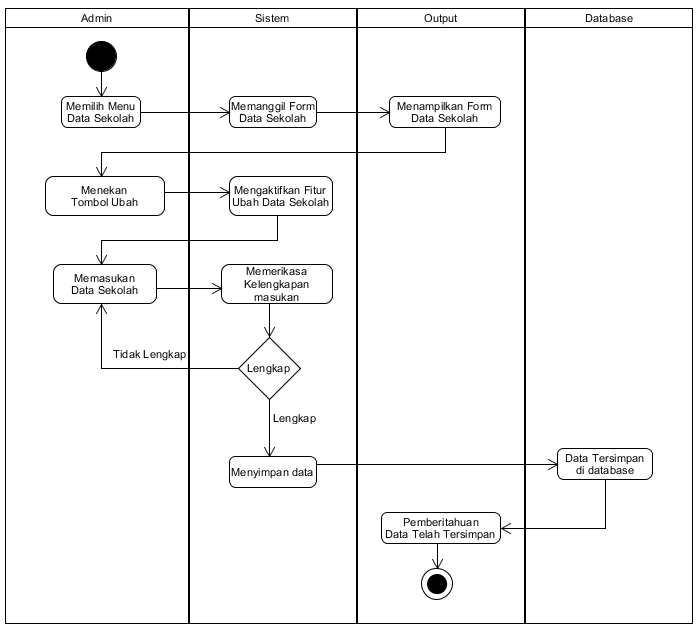
*Activity diagram* ini dibuat untuk menunjukkan aktivitas yang dilakukan admin dan timbal balik yang dilakukan aplikasi terhadap aktivitas admin secara sistematis.

1. *Activity diagram login*



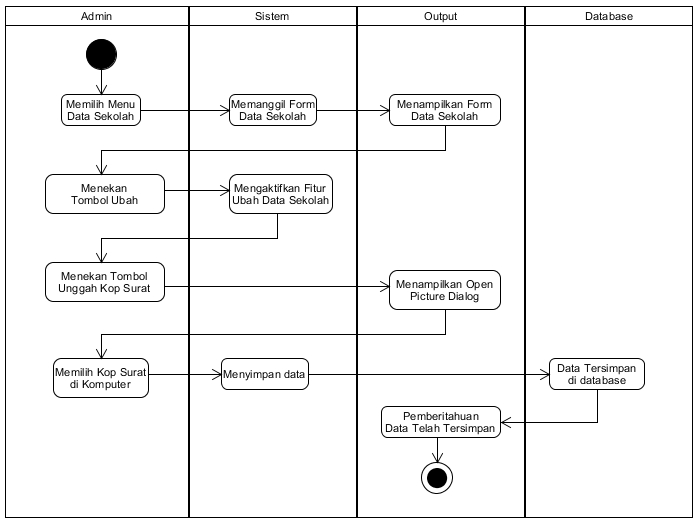
Gambar 4.4 *Activity Diagram* Login

1. *Activity Diagram* Data Sekolah



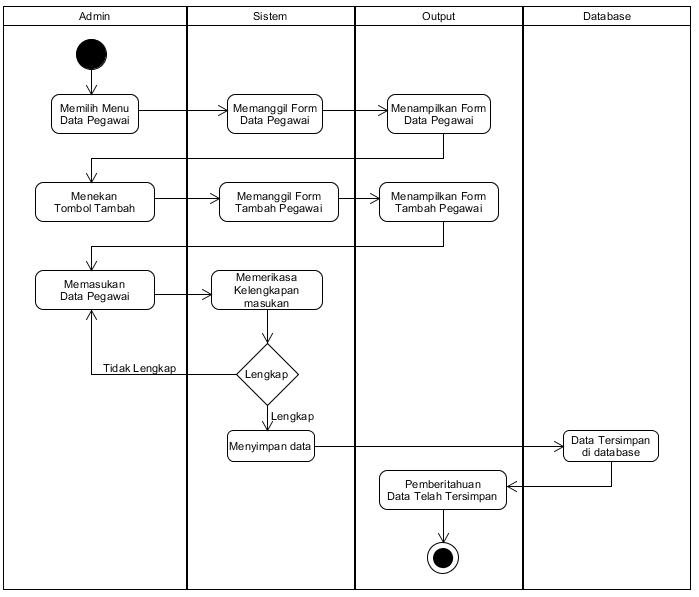
Gambar 4.5 *Activity Diagram* Ubah Data Sekolah

1. *Activity diagram* unggah kop surat



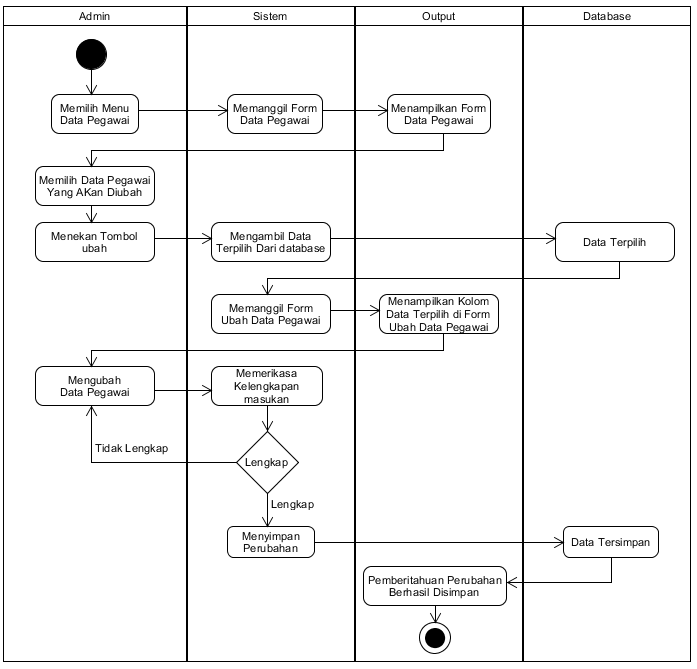
Gambar 4.6 *Activity Diagram* unggah kop surat

1. *Activity diagram* tambah data pegawai



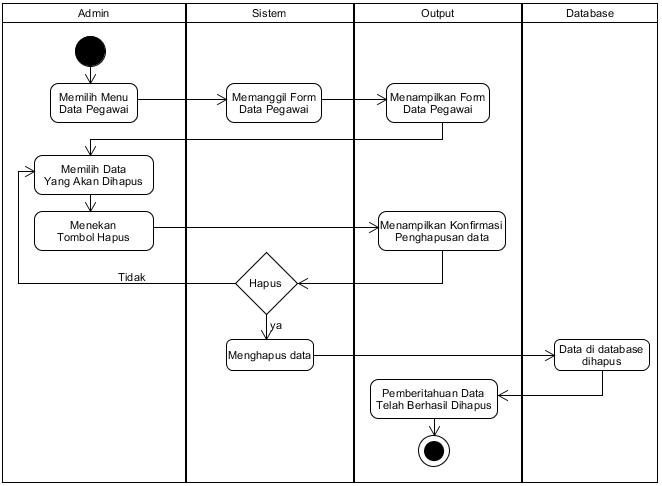
Gambar 4.7 *Activity Diagram* Tambah Data Pegawai

1. *Activity Diagram* Ubah Data Pegawai



Gambar 4.8 *Activity Diagram* Ubah Data Pegawai

1. *Activity diagram* hapus data pegawai



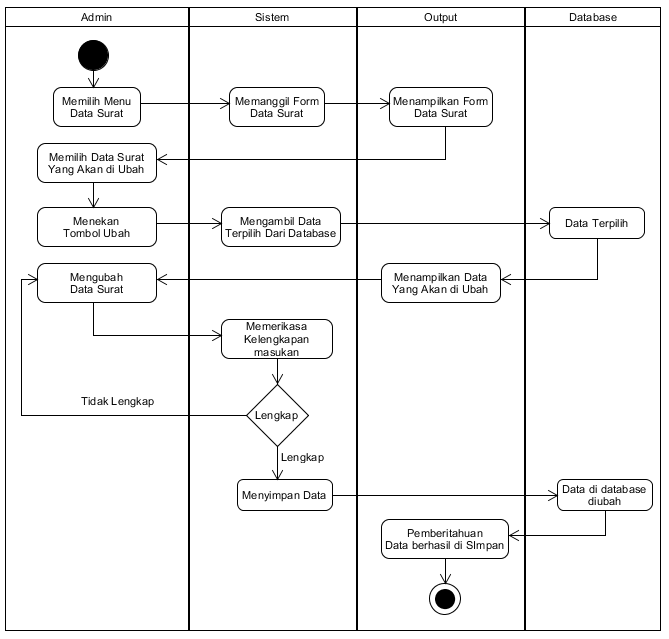
Gambar 4.9 *Activity Diagram* Hapus Data Pegawai

1. *Activity diagram* tambah data surat



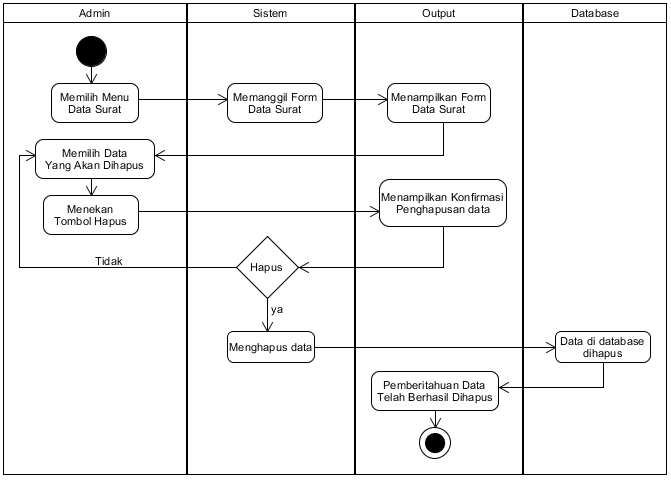
Gambar 4.10 *Activity Diagram* Tambah Data Surat

1. *Activity diagram* ubah data surat



Gambar 4.11 *Activity Diagram* Ubah Data Surat

1. *Activity Diagram* hapus data surat



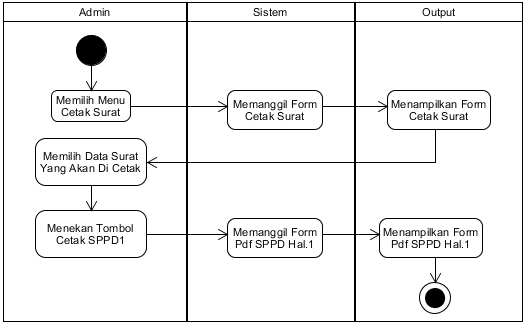
Gambar 4.12 *Activity diagram* Hapus Data Surat

1. *Activity diagram* cetak surat tugas



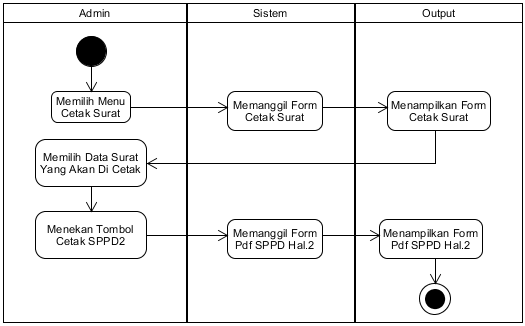
Gambar 4.13 *Activity Diagram* Cetak Surat Tugas

1. *Activity diagram* cetak SPPD1



Gambar 4.14 *Activity Diagram* Cetak SPPD 1

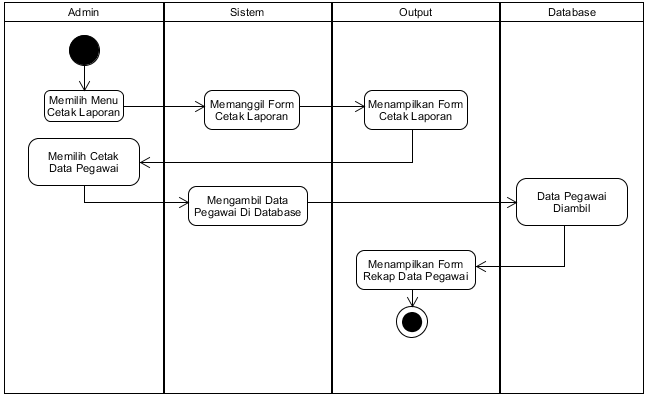
1. *Activity Diagram* Cetak SPPD2



Gambar 4.15 *Activity Diagram* Cetak SPPD 2

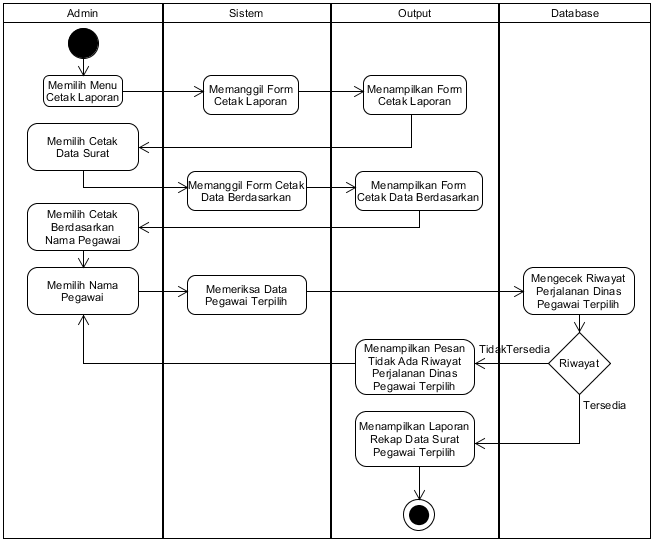


1. *Activity diagram* cetak laporan data pegawai



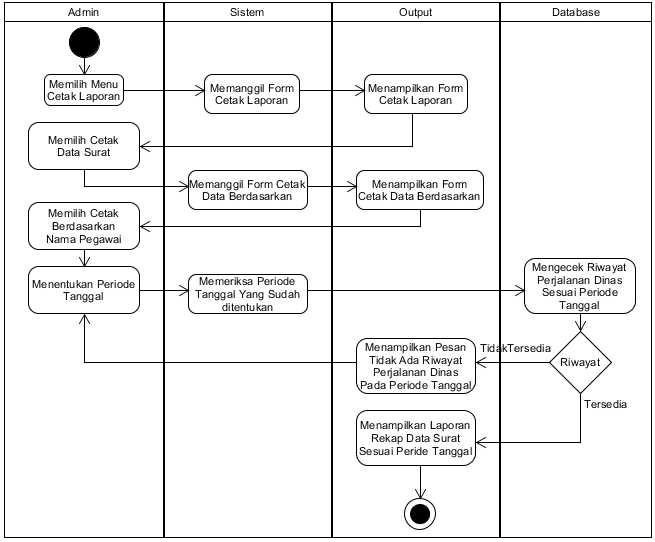
Gambar 4.16 *Activity Diagram* Cetak Laporan Data Pegawai

1. *Activity diagram* cetak laporan data surat berdasarkan nama pegawai



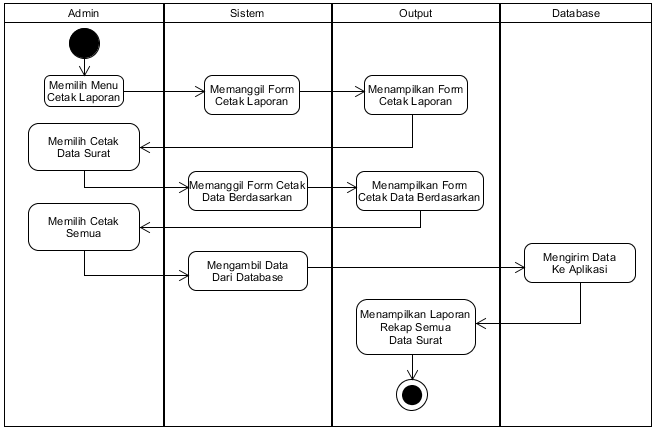
Gambar 4.17 *Activity Diagram* Cetak Laporan Data Surat Berdasarkan Nama Pegawai

1. *Activity diagram* cetak laporan data surat berdasarkan tanggal



Gambar 4.18 *Activity Diagram* Cetak Laporan Data Surat Berdasarkan Periode Tanggal

1. *Activity diagram* cetak laporan rekap semua data surat



Gambar 4.19 *Activity Diagram* Cetak Laporan Semua Data Surat

## **Kamus Data**

Kamus data untuk aplikasi ini adalah :

* + - 1. Tabel Data Pengguna

Tabel data pengguna terdiri dari dua *field* yaitu Pengguna dan KataSandi. Tabel data pengguna berfungsi untuk menyimpan data pengguna dan untuk keperluan proses *login.*

Tabel 4.21 Tabel Data Pengguna

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Type** | **Length** | **Keterangan** |
| Pengguna | Text | 20 | Primary Key |
| KataSandi | Number | 8 |  |

* + - 1. Tabel Data Sekolah

Tabel data sekolah terdiri dari sepuluh *field* yaitu NPSN, NamaSekolah, Telp, Email, Alamat, Kecamatan, Kab/Kota, KepalaSekolah, NIP, dan Kop. Tabel data Sekolah berfungsi untuk menyimpan data sekolah dan kop surat.

Tabel 4.22 Tabel data sekolah

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Type** | **Length** | **Keterangan** |
| NPSN | Number | 8 | Primary Key |
| NamaSekolah | Text | 25 |  |
| Telp | Text | 15 |  |
| Email | Text | 40 |  |
| Alamat | Text | 50 |  |
| Kecamatan | Text | 30 |  |
| Kab/Kota | Text | 30 |  |
| KepalaSekolah | Text | 30 |  |
| NIP | Text | 18 |  |
| Kop | OLE Object |  |  |

* + - 1. Tabel Data Pegawai

Tabel data sekolah terdiri dari sembilan *field* yaitu NIP/NUPTK/ID, Nama, Status, Pangkat, Golongan, Jenis Kelamin, Tempat Lahir, Tanggal Lahir, KepalaSekolah, dan Jabatan. Tabel data Sekolah berfungsi untuk menyimpan data pegawai serta untuk membantu proses pembuatan surat dan proses cetak data pegawai.

Tabel 4.23 Tabel data sekolah

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Type** | **Length** | **Keterangan** |
| NIP/NUPTK/ID | Text | 18 | Primary Key |
| Nama | Text | 40 |  |
| Status | Text | 8 |  |
| Pangkat | Text | 20 |  |
| Golongan | Text | 5 |  |
| Jenis Kelamin | Text | 15 |  |
| Tempat Lahir | Text | 30 |  |
| Tanggal Lahir | Date/Time | Short Date |  |
| Jabatan | Text | 20 |  |

* + - 1. Tabel Data Surat

Tabel data sekolah terdiri dari tujuh belas *field* yaitu NoSurat, Nama, Pangkat, Golongan, Jabatan, UnitKerja, Dasar, Perihal, Kegiatan, Hari, HariBerangkat, TanggalBerangkat, HariPulang, TanggalPulang, Waktu, Tempat, dan Angkutan. Tabel data surat berfungsi untuk menyimpan data surat serta untuk membantu proses cetak surat dan proses cetak data surat.

Tabel 4.24 Tabel 4.25 Tabel Data Surat

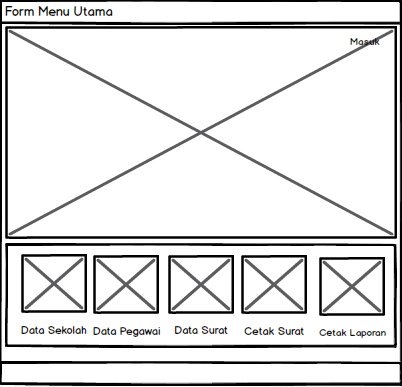
|  |  |  |  |
| --- | --- | --- | --- |
| **Nama Field** | **Type** | **Length** | **Keterangan** |
| NoSurat | Text | 20 | Primary Key |
| Nama | Text | 40 |  |
| Pangkat | Text | 20 |  |
| Golongan | Text | 5 |  |
| Jabatan | Text | 20 |  |
| UnitKerja | Text | 25 |  |
| Dasar | Text | 200 |  |
| Perihal | Text | 200 |  |
| Kegiatan | Text | 100 |  |
| Hari | Text | 6 |  |
| HariBerangkat | Text | 10 |  |
| TanggalBerangkat | Date/Time | Short Date |  |
| HariPulang | Text | 10 |  |
| TanggalPulang | Date/Time | Short Date |  |
| Waktu | Text | 20 |  |
| Tempat | Text | 50 |  |
| Angkutan | Text | 25 |  |

Tabel 4.26 Tabel data sekolah



## **Desain**

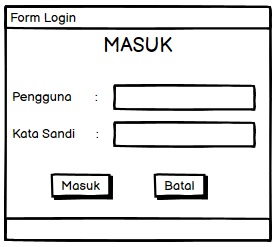
1. Tampilan Menu Utama



Gambar 4.20 Desain Form Menu Utama

Tampilan gambar diatas merupakan desain dasboard untuk menampilkan beranda dan menu bar yang ada pada aplikasi tersebut. Menu tersebut akan terbuka apabila admin sudah berhasil masuk ke aplikasi.

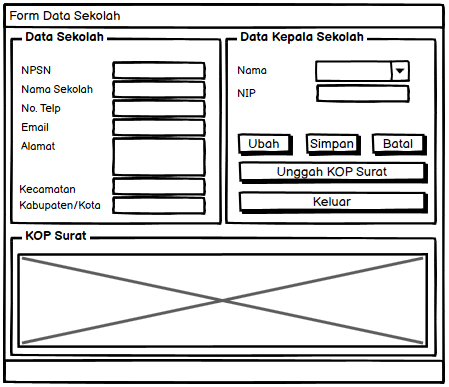
1. Tampilan Login



Gambar 4.21 Desain Form Login

Tampilan gambar di atas merupakan desan tampilan login aplikasi, jika pengguna dan kata sandi benar maka fitur menu di menu utama akan terbuka, dan admin dapat mengakses fitur tersebut tetapi jika pengguna dan kata sandi salah admin tidak dapat mengakses fitur menu tersebut .

1. Tampilan Data Sekolah



Gambar 4.22 Desain Data Sekolah

Tampilan gambar di atas merupakan tampilan input data sekolah berfungsi untuk memasukan data sekolah dan kop surat. Data tersebut akan digunakan pada surat tugas dan SPPD. Proses ini dilakukan oleh admin yang dapat mengubah dan mengunggah kop surat.

1. Tampilan Data Pegawai



Gambar 4.23 Desain Form Data Pegawai

Tampilan gambar di atas merupakan tampilan data pegawai. Form diatas berfungsi untuk menampilkan data pegawai, memanggil form tambah pegawai dan ubah pegawai. Pada form tesebut admin juga bisa menghapus data pegawai yang terpilih. Data tersebut akan digunakan pada surat tugas dan SPPD.

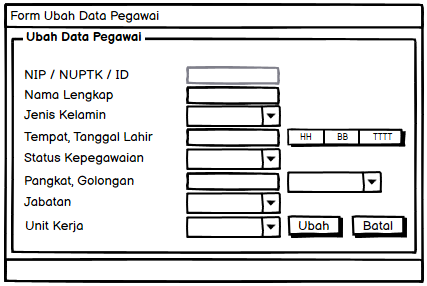
1. Tampilan Tambah Data Pegawai



Gambar 4.24 Desain Form Tambah Pegawai

Gambar diatas merupakan desain tampilan tambah data pegawai. Form diatas akan muncul apabila admin menekan tombol tambah pada form data pegawai sebelumnya. Disini admin bisa menambah data pegawai dengan memasukan data yang diminta. Untuk jenis kelamin, status kepegawaian, golongan, jabatan dan unit kerja, admin hanya tinggal memilih data yang sudah disediakan oleh aplikasi. Untuk pangkat akan otomatis terisi sesuai dengan golongan yang dipilih.

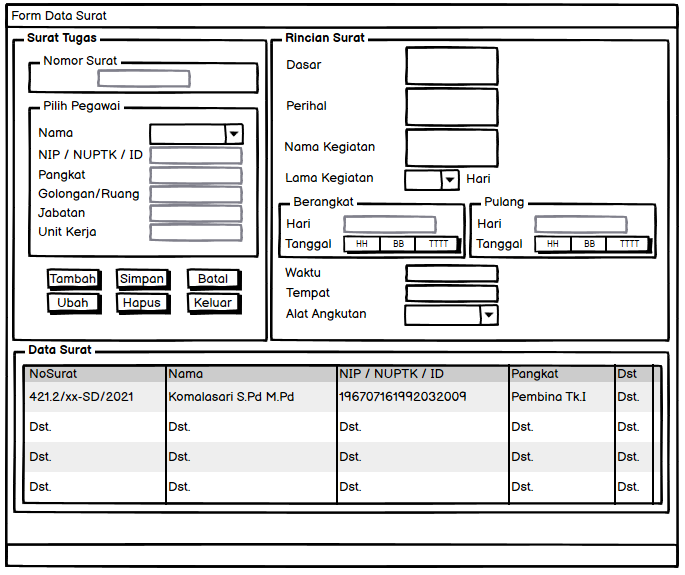
1. Tampilan Ubah Data Pegawai



Gambar 4.25 Desain Form Ubah Data Pegawai

Gambar diatas adalah desain form ubah data pegawai. Admin dapat memanggil form diatas dengan menekan tombol ubah di form data sekolah. Form diatas berfungsi untuk mengubah data pegawai yang telah dipilih. Sama seperti form tambah pegawai, admin akan diminita memasukan data yang akan diubah. Akan tetapi, untuk data NIP / NUPTK/ ID, admin tidak bisa merubah data tersebut dikarenakan data tersebut merupakan primary key. Setelah selesai, admin bisa menekan tombol ubah maka data tersebut akan tersimpan di database.

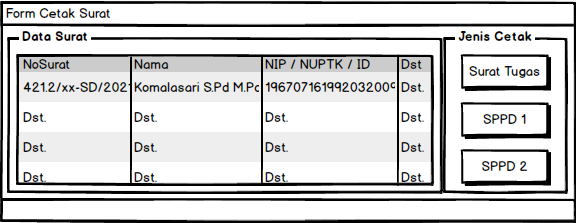
1. Tampilan Data Surat



Gambar 4.26 Desain Form Data Surat

Gambar diatas merupakan desain form data surat. Pada form ini admin bisa menambah, mengubah, dan menghapus data surat. Nomor surat akan otomatis terisi setelah admin menekan tombol tambah. Untuk data pegawai, admin hanya tinggal memilih nama pegawai, maka data pegawai lainnya akan otomatis terisi. Bagian rincian surat, admin bisa mengisi sesuai surat undangan yang diterima oleh sekolah.

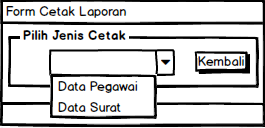
1. Tampilan Cetak Surat



Gambar 4.27 Desain Form Cetak Surat

Gambar diatas adalah desain tampilan form cetak surat. Disini admin bisa mencetak surat tugas, SPPD halaman 1, dan SPPD halaman 2 setelah memilih data surat yang akan di cetak.

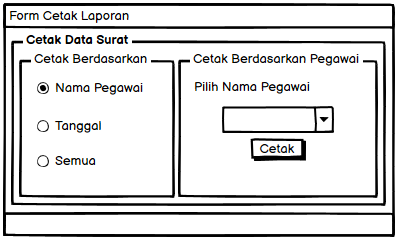
1. Tampilan Cetak Laporan



Gambar 4.28 Desain Form Cetak Laporan

Gambar diatas adalah desain form cetak laporan. Setelah admin menekan tombol cetak laporan di menu utama, maka akan muncul form tersebut. Admin bisa memilih antara mencetak data pegawai atau data surat pada combo box. Jika admin memilih data pegawai maka pengguna akan langsung di arahkan ke form rekap data pegawai. Namun jika memilih data surat makan pengguna akan di arahkan ke form cetak surat. Berikut tampilan form cetak surat.

1. Tampilan Cetak Data Surat Berdasarkan Data Pegawai



Gambar 4.29 Desain Form Cetak Data Surat Berdasarkan Data Pegawai

Diatas Merupakan desain form cetak data surat berdasarkan nama pegawai. Sebelum mencetak, admin harus memilih nama pegawai untuk di cetak.

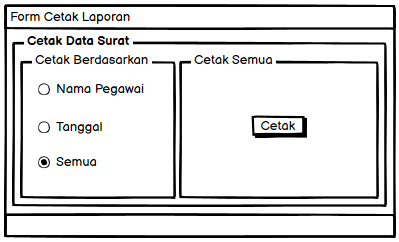
1. Tampilan Cetak Data Surat Berdasarkan Tanggal



Gambar 4.30 Desain Form Cetak Data Surat Berdasarkan Tanggal

Diatas Merupakan desain form cetak data surat berdasarkan Tanggal. Sebelum mencetak, admin harus memilih surat berdasarkan tanggal untuk di cetak.

1. Tampilan Cetak Semua Data Surat



Gambar 4.31 Desain Form Cetak Semua Data Surat

Diatas Merupakan desain form cetak semua data surat. Disini admin bisa mencetak seluruh data surat.

# **BAB V IMPLEMENTASI DAN PENGUJIAN**

## **Implementasi Sistem**

Implementasi aplikasi surat tugas dan surat perintah perjalanan dinas ini dilaksanakan pada :

Hari/Tanggal : Sabtu, 26 Juni 2021

Waktu : 08:00 WIB – Selesai

Tempat : SDN Mekarsari Baleendah

### **Implementasi Perangkat Keras**

Spesifikasi perangkat keras minimum yang dibutuhkan untuk menjalan kan aplikasi surat tugas dan surat perintah perjalanan dinas ini adalah sebagai berikut :

|  |  |  |
| --- | --- | --- |
| No | Perangkat Keras | Spesifikasi |
| 1. | Komputer / Laptop | *Processor* Intel Core2 Duo 1.8GHz / AMD Athlon Ⅱ X2 |
| RAM 2 GB |
| Graphics Intel HD Graphics / ATI Radeon HD4850 |
| *Hardisk* 320 GB |
| 2. | Printer | Epson L120 |

### **Implementasi Perangkat Lunak**

Spesifikasi perangkat lunak minimum yang dibutuhkan untuk menjalan kan aplikasi surat tugas dan surat perintah perjalanan dinas ini adalah sebagai berikut :

|  |  |  |
| --- | --- | --- |
| No | Perangkat lunak | Spesifikasi |
| 1. | *Windows* | *Windows* 10 – 32 Bit |
| 2. | PDF *Viewer* | Foxit Reader |

### **Implementasi Antar Muka**

Berikut ini adalah contoh implementasi antar muka aplikasi surat tugas dan surat perintah perjalanan dinas yang telah dibuat :

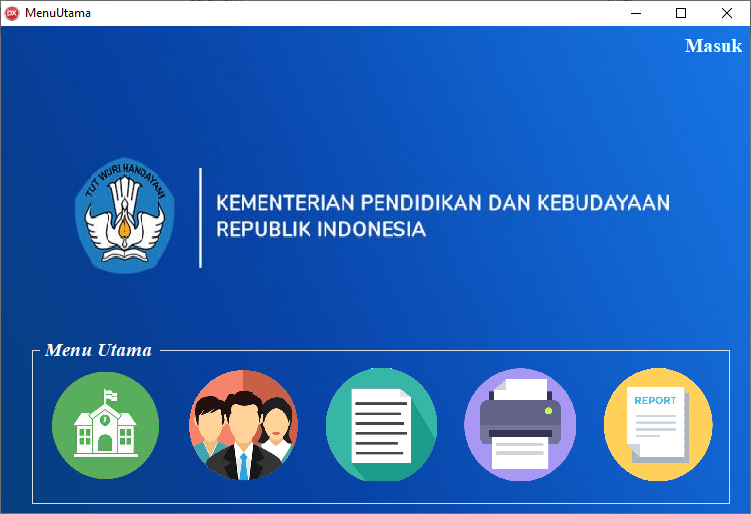
1. Tampilan Login



Gambar 5.1 Tampilan Login

gambar diatas merupakan tampilan *login* pengguna untuk masuk ke dalam sistem, jika pengguna dan kata sandi salah maka pengguna tidak diizinkan untuk menampilkan menu utaman dan akan terus ditampilkan *form login* tersebut, jika pengguna dan kata sandi benar maka dapat mengakses fitur menu utama.

1. Tampilan Menu



Gambar 5.2 Tampilan Menu Utama

Gambar diatas adalah tampilan menu utama. Di menu utama ini pengguna bisa mengakses semua fitur aplikasi seperti : data sekolah, data pegawai, data surat, cetak surat, cetak laporan dan lain-lain. Tombol tersebut akan muncul apabila pengguna memasukan nama pengguna dan kata sandi yang benar pada form login sebelumnya.

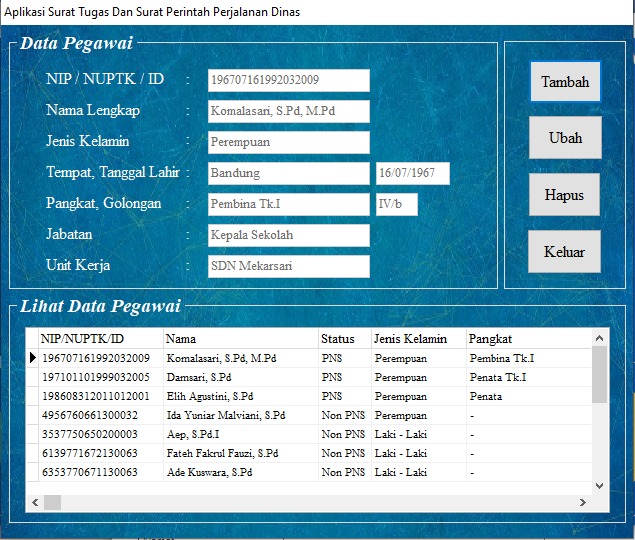
1. Tampilan data sekolah



Gambar 5.3 Tampilan Data Sekolah

Gambar diatas merupakan tampilan data sekolah. Data sekolah akan muncul apabila pengguna menekan tombol data sekolah di menu utama. Pengguna bisa mengganti data sekolah berdasarkan data sekolah yang ada. Pengguna bisa memasukan data sekolah apabila pengguna sudah menekan tombol ubah. Hal ini karena *text field* pada *form* ini bersifat nonaktif (tidak bisa dirubah) sebelum pengguna menekan tombol ubah dan hanya bisa diaktifkan dengan menekan tombol ubah. Disini selain data sekolah, pengguna juga dapat mengunggah kop surat sesuai kebutuhan dengan menekan tombol unggah surat yang akan aktif setelah pengguna menekan tombol ubah.

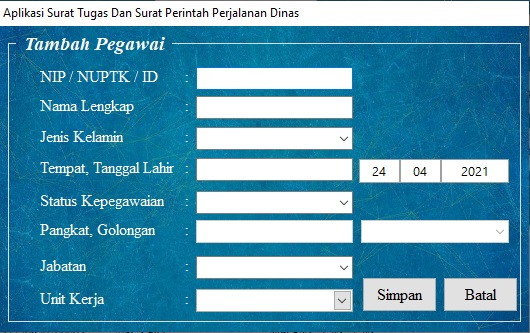
1. Tampilan data pegawai



Gambar 5.4 Tampilan Data Pegawai

Gambar diatas adalah tampilan data pegawai. *Form* ini memiliki beberapa tombol yang memiliki fungsi masing-masing, tombol tersebut diantaranya :

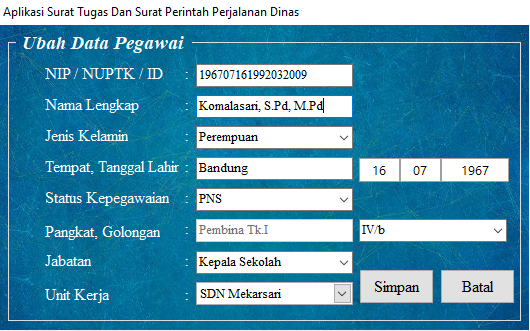
* 1. Tambah data pegawai



Gambar 5.5 Tampilan Tambah Data Pegawai

*Form* diatas akan muncul apabila pengguna menekan tombol tambah pada *form* data pegawai sebelumnya. *Form* tambah data pegawai berfungsi untuk menambah data data pegawai. Pengguna bisa mengisi data pegawai sesuai dengan *text field* yang tersedia pada form tersebut. Apabali semua data sudah terisi, pengguna bisa menyimpan data tersebut dengan menekan tombol simpan.

* 1. Ubah data pegawai



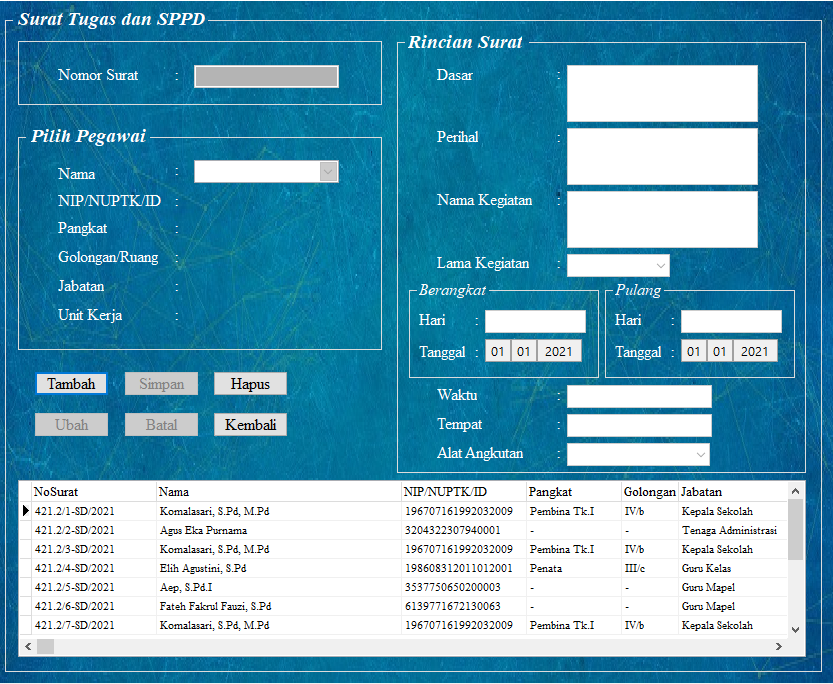
Gambar 5.6 Tampilan Ubah Data Pegawai

Gambar diatas merupakan *form* ubah data pegawai yang berfungsi untuk mengubah data pegawai yang sudah tersimpan di *database. Form* tersebut akan muncul apbila pengguna menekan tombol ubah. Sebelum menenkan tombol ubah, pengguna harus memilih terlebih dahulu data pegawai mana yang akan di ubah melalui *tab* lihat data pegawai yang berada pada *form* data pegawai.

* 1. Hapus Data

Tombol tersebut berfungsi untuk menghapus data pegawai yang sudah tersimpan di *database*. Sebelum menghapus, pengguna terlebih dahulu memilih data pegawai yang akan di hapus pada *tab* lihat data pegawai. Setelah memilih dan menekan tombol hapus, akan muncul konfirmasi penghapusan data yang apabila pengguna menekan tombol ‘ya’ maka data tersebut akan terhapus dari *database*.

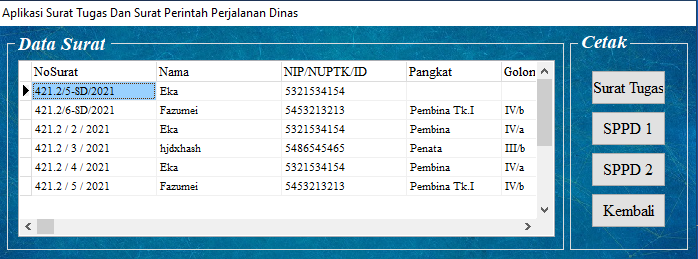
1. Tampilan data surat



Gambar 5.7 Tampilan data surat

Gambar diatas merupakan tampilandata surat yang berfungsi untuk mengelola data surat. *Form*  ini akan muncul apabila pengguna menekan tombol data surat di menu utama. Langkah pertama untuk membuat data surat adalah menekan tombol tambah. Setelah pengguna menekan tombol tambah maka nomor surat akan otomatis muncul. Selanjutnya untuk data pegawai yang akan melakukan perjalanan dinas, pengguna hanya perlu memilih nama pegawai pada *combobox* yang tersedia maka data lain seperti NIP/NUPTK/ID, Pangkat, Golongan, Jabatan dan Unit kerja akan otomatis terisi berdasarkan data pegawai terpilih yang tersimpan di tabel pegawai. Untuk rincian surat, pengguna tinggal menyesuaikan dengan kebutuhan surat. Setelah semua data sudah sesuai, maka langkah terakhir adalah menyimpan data tersebut ke *database* dengan menekan tombol simpan.

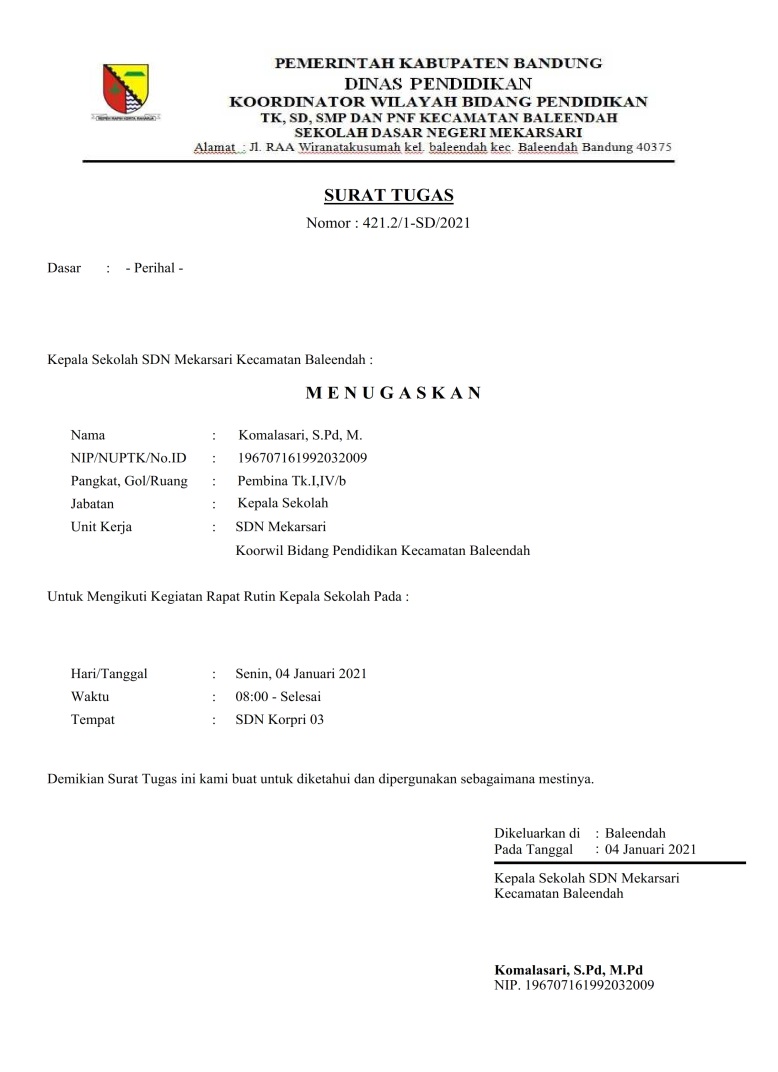
1. Tampilan cetak surat



Gambar 5.8 Tampilan Cetak Surat

Gambar diatas merupakan tampilan *form* cetak surat. *Form* ini berfungsi untuk mencetak surat tugas dan SPPD. Disini pengguna bisa memilih data surat mana yang akan di cetak. Berikut hasil cetak *form* tersebut :

* 1. Surat Tugas



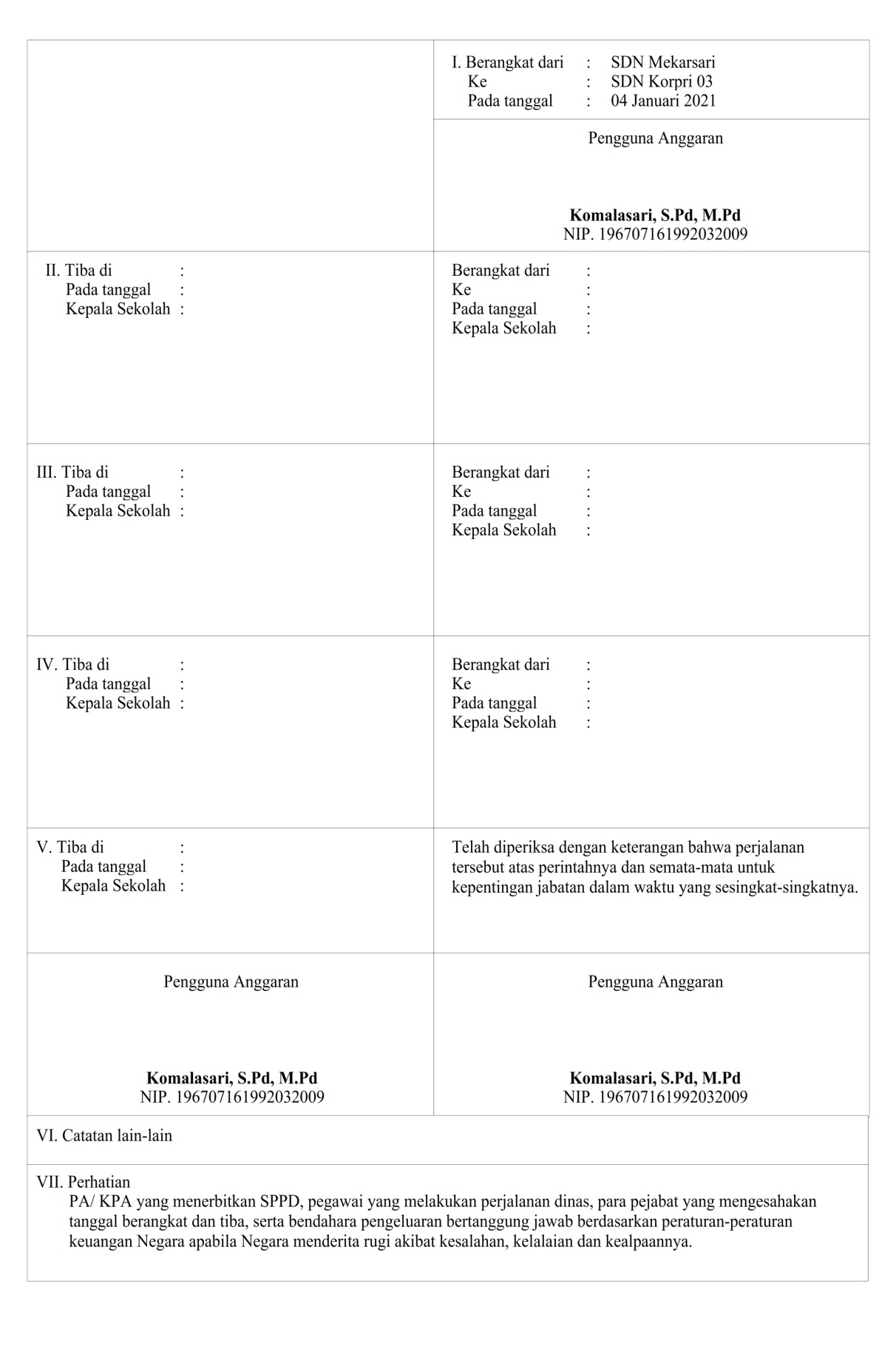
Gambar 5.9 Hasil Cetak Surat Tugas

* 1. SPPD 1 (SPPD Halaman 1)



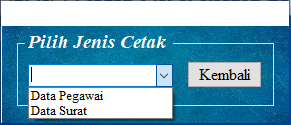
Gambar 5.10 Hasil Cetak SPPD Halaman 1

* 1. SPPD 2 (SPPD Halaman 2)



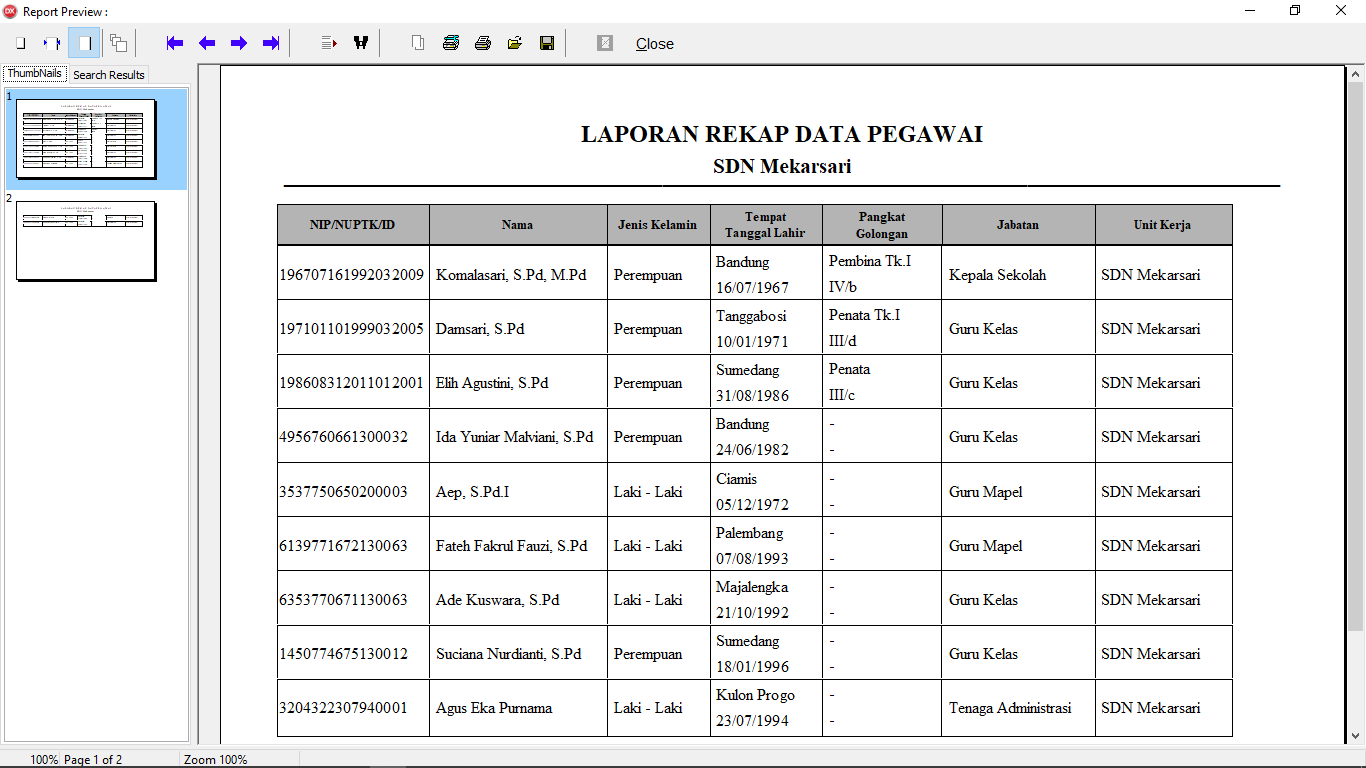
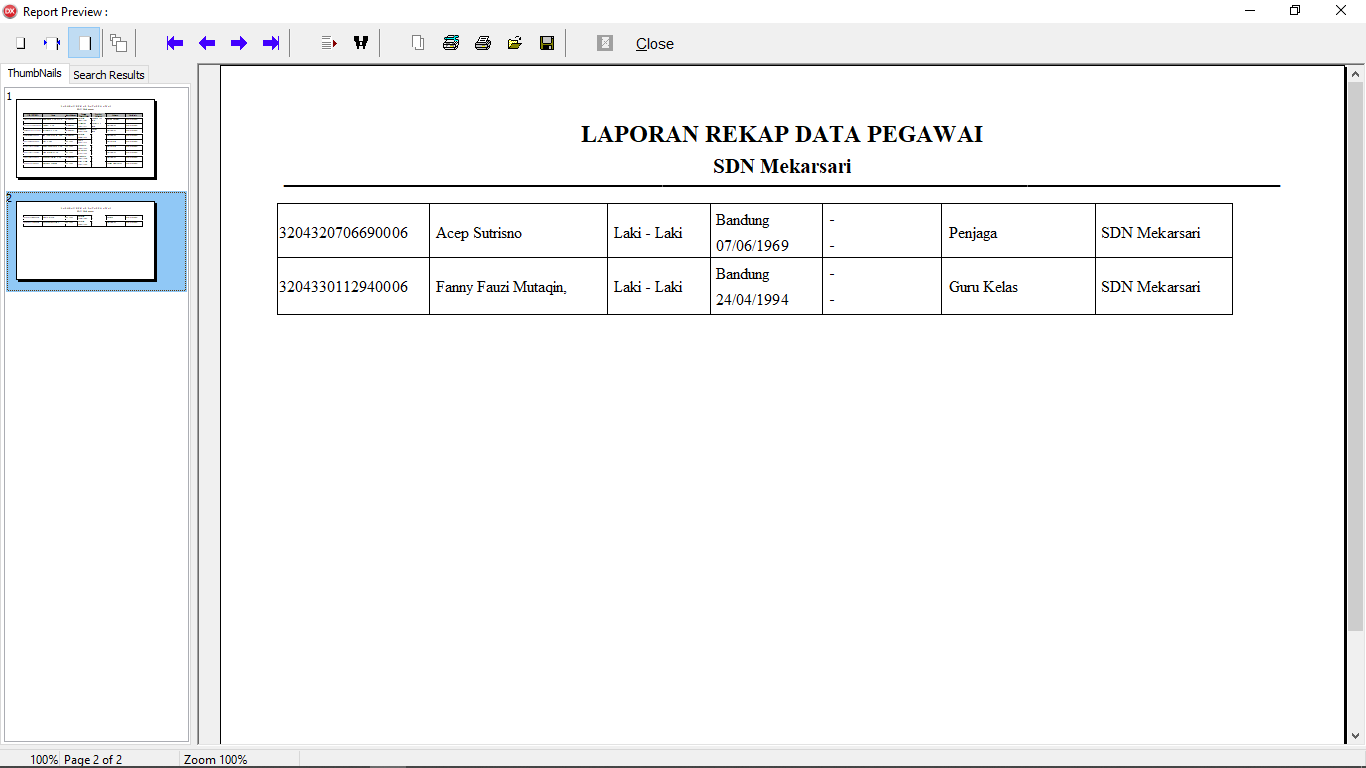
Gambar 5.11 Hasil Cetak SPPD Halaman 2

1. Tampilan Cetak Laporan



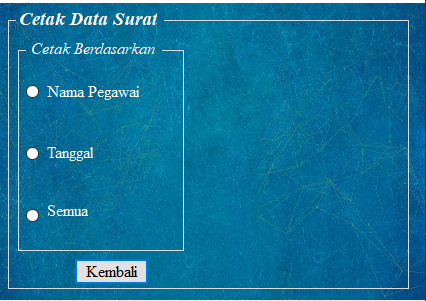
Gambar 5.12 Tampilan Pilih Jenis Cetak Laporan

Gambar diatas adalah tampilan *form* pilih jenis cetak pegawai. *Form* berfungsi untuk memilih jenis cetak laporan rekap data pegawai dan data surat. Apabila pengguna memilih cetak data pegawai maka akan muncul laporan rekap data pegawai seperti dibawah ini.

Gambar 5.13 Hasil Cetak Laporan Rekap Data Pegawai

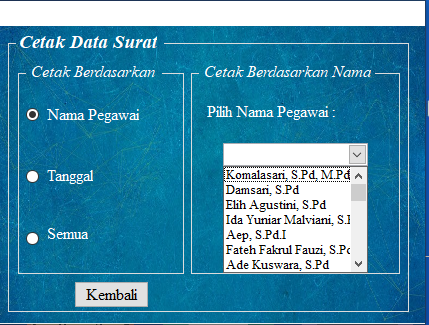
Apabila pengguna memilih data surat, maka akan muncul *form* baru yaitu *form* cetak data surat seperti di bawah ini :



Gambar 5.14 Tampilan Cetak Data Surat

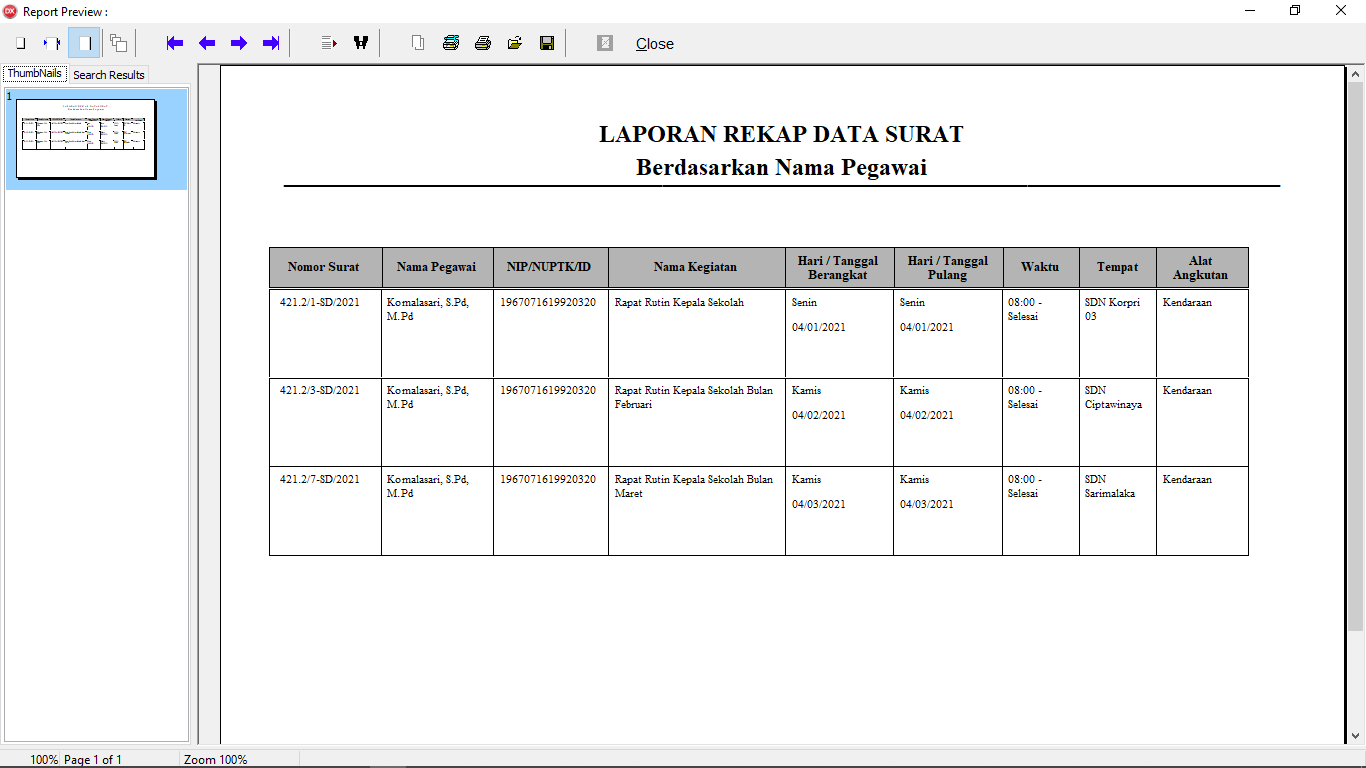
Pada *form* ini pengguna bisa memilih jenis cetak surat berdasarkan tiga kategori yaitu :

* + - 1. Cetak Berdasarkan Nama Pegawai



Gambar 5.15 Tampilan Cetak Berdasarkan Nama Pegawai

Jika pengguna memilih cetak berdasarkan nama pegawai, maka pengguna bisa mencetak lapran rekap data surat berdasarkan data pegawai yang dipilih pada *combobox* nama pegawai. Apabila nama pegawai tersebut tidak memilih data surat, maka akan muncul notifikasi yang menyatakan bahwa pegawai tersebut tidak memiliki riwayat perjalanan dinas. Berikut hasil cetak data surat berdarkan data pegawai.



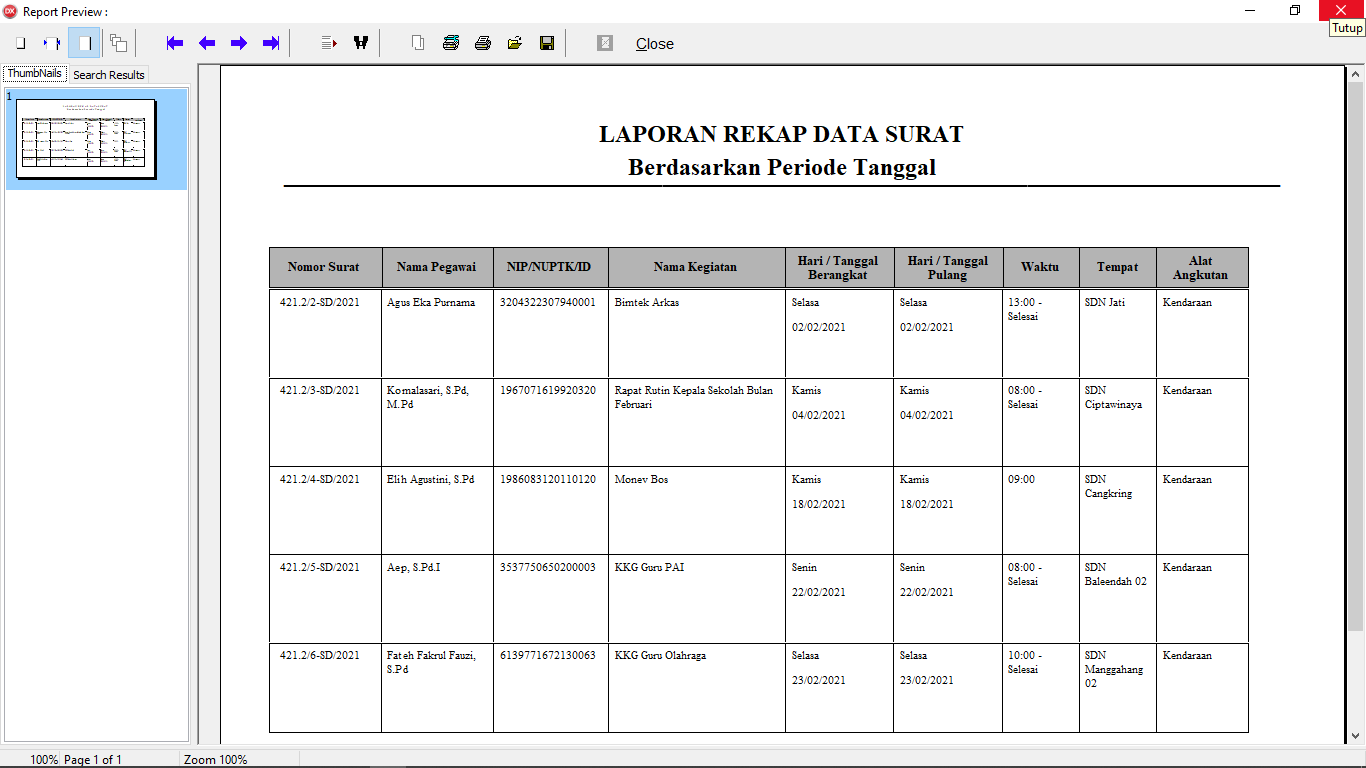
Gambar 5.16 Hasil Cetak Laporan Rekap Surat Berdasarkan Nama Pegawai

* + - 1. Cetak Berdasarkan Tanggal



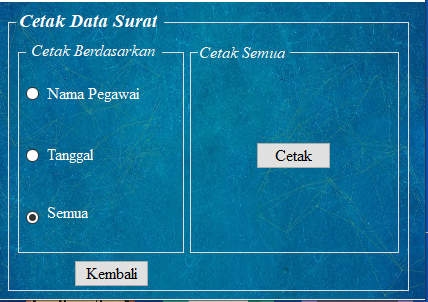
Gambar 5.17 Tampilan Cetak Berdasarkan Tanggal

Jika pengguna memilih cetak berdasarkan tanggal, maka pengguna bisa mencetak laporan rekap surat tugas berdasarkan rentang tanggal yang dipilih. Apabila pada rentang tanggal yang dipilih tidak memiliki riwayat perjalanan dinas maka akan muncul notifikasi yang menyatakan bahwa pada rentang tanggal tersebut tidak memiliki riwayat perjalanan dinas. Berikut hasil cetak laporan rekap surat berdasarkan tanggal.



Gambar 5.18 Hasil Cetak Laporan Rekap Surat Berdasarkan Tanggal

* + - 1. Cetak Semua Data



Gambar 5.19 Tampilan Cetak Semua

Pilihan terakhir adalah cetak semua data. Pada pilihan ini pengguna bisa mencetak semua data surat tugas yang tersimpan dalam *database.* Berikut adalah hasil cetak laporan rekap semua data.



Gambar 5.20 Hasil Cetak Laporan Rekap Semua Data

## **Pengujian**

### **Rencana Pengujian**

Sebelum aplikasi diterapkan di lapangan, perlu adanya proses pengujian untuk menentukan kesalahan pada aplikasi. Pada tahap pengujian ini penulis menggunakan metode blackbox yaitu metode pengujian perangkat lunak dengan hanya dilihat berdasarkan keluaran yang dihasilkan dari data atau kondisi masukan yang diberikan untuk fungsi yang ada tanpa melihat bagaimana proses untuk mendapatkan keluaran tersebut.

Berikut adalah hasil pengujian yang telah dilakukan oleh penulis :

1. Rencana Pengujian

Tabel 5.1 Tabel Rencana Pengujian

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No. | Kode | Nama Proses | Input | Output |
| 1 | SKPL-01 | Login | Masukan username dan password ke halaman login aplikasi | Mengaktifkan fitur menu utama |
| 2 | SKPL-02 | Ubah data sekolah | Memasukan data sekolah | Data Tersimpan ke *database* |
| 3 | SKPL-03 | Unggah kop surat | Memilih gambar kop surat | Data Tersimpan ke *database* |
| 4 | SKPL-04 | Tambah data pegawai | Memasukan data pegawai | Data Tersimpan ke *database* |
| 5 | SKPL-05 | Ubah data pegawai | Merubah data lama dengan data baru | Data Tersimpan ke database |
| 6 | SKPL-06 | Hapus data pegawai | Memilih data yang akan di hapus | Data akan dihapus dari *database* |
| 7 | SKPL-7 | Tambah data surat | Memasukan data surat | Data Tersimpan ke *database* |
| 8 | SKPL-8 | Hapus data surat | Memilih data yang akan di hapus | Data akan dihapus dari *database* |
| 9 | SKPL-9 | Ubah Data Surat | Merubah data lama dengan data baru | Data Tersimpan ke database |
| 10 | SKPL-10 | Cetak surat | Memilih data surat yang akan di cetak | Menampilkan data surat yang akan dicetak oleh aplikasi |
| 11 | SKPL-11 | Cetak laporan rekap | Memilih data yang akan di cetak | Menampilkan data laporan rekap yang akan dicetak oleh aplikasi |
| 12 | SKPL-12 | Cetak laporan rekap data surat berdasarkan nama pegawai | Memilih nama pegawai | Menampilkan laporan rekap data surat berdasarkan nama pegawai |
| 13 | SKPL-13 | Cetak laporan rekap data surat berdasarkan periode tanggal | Menentukan periode tanggal awal dan akhir yang akan di cetak | Menampilkan laporan rekap data surat berdasarkan nama periode tanggal |
| 14 | SKPL-14 | Cetak Semua laporan rekap data surat | Menekan tombol cetak | Menampilkan semua laporan rekap data surat |

Tabel 5.2 Tabel Rencana Pengujian

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Kelas Uji | Butir Uji | | Identifikasi | | | | Jenis Pengujian | | Teknik Pengujian | |
| SKPL | | PDHUPL | |
| Login | Masuk dengan pengguna dan kata sandi yang terdaftar | | SKPL-01 | | PDHUPL-01 | | Pengujian Sistem | | Blackbox | |
|
|
| Masuk dengan pengguna dan kata sandi yang tidak terdaftar | | SKPL-01 | | PDHUPL-02 | | Pengujian Sistem | | Blackbox | |
|
| Ubah data sekolah | Mengubah data sekolah dengan mengisi semua kolom yang wajib diisi | | SKPL-02 | | PDHUPL-03 | | Pengujian Sistem | | Blackbox | |
|
|
| Mengubah data sekolah tanpa mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) | | SKPL-02 | | PDHUPL-04 | | Pengujian Sistem | | Blackbox | |
|
| Unggah kop surat | Memilih gambar kop surat di komputer/laptop | | SKPL-03 | | PDHUPL-05 | | Pengujian Sistem | | Blackbox | |
| Tambah data pegawai | Memasukan data pegawai dengan mengisi semua kolom yang wajib diisi | | SKPL-04 | | PDHUPL-06 | | Pengujian Sistem | | Blackbox | |
| Memasukan data sekolah tanpa mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) | | SKPL-04 | | PDHUPL-07 | | Pengujian Sistem | | Blackbox | |
| Ubah data pegawai | Mengubah data pegawai dengan mengisi semua kolom yang wajib diisi | | SKPL-05 | | PDHUPL-08 | | Pengujian Sistem | | Blackbox | |
| Mengubah data sekolah tanpa mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) | | SKPL-05 | | PDHUPL-09 | | Pengujian Sistem | | Blackbox | |
| Hapus data pegawai | Memilih pilihan “YA” saat muncul konfirmasi hapus data | SKPL-06 | | PDHUPL-10 | | Pengujian Sistem | | Blackbox | |
| Memilih pilihan “TIDAK” saat muncul konfirmasi hapus data | SKPL-06 | | PDHUPL-11 | | Pengujian Sistem | | Blackbox | |
| Tambah data surat | Memasukan data surat dengan mengisi semua kolom yang wajib diisi | SKPL-07 | | PDHUPL-12 | | Pengujian Sistem | | Blackbox | |
| Memasukan data surat tanpa mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) | SKPL-07 | | PDHUPL-13 | | Pengujian Sistem | | Blackbox | |
| Ubah data surat | Mengubah data surat dengan mengisi semua kolom yang wajib diisi | SKPL-08 | | PDHUPL-14 | | Pengujian Sistem | | Blackbox | |
| Mengubah data surat tanpa mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) | SKPL-08 | | PDHUPL-15 | | Pengujian Sistem | | Blackbox | |
| Hapus data surat | Memilih pilihan “YA” saat muncul konfirmasi hapus data | SKPL-09 | | PDHUPL-16 | | Pengujian Sistem | | Blackbox | |
| Memilih pilihan “TIDAK” saat muncul konfirmasi hapus data | SKPL-09 | | PDHUPL-17 | | Pengujian Sistem | | Blackbox | |
| Cetak surat | Memilih data surat dan menekan tombol “Surat Tugas” | SKPL-10 | | PDHUPL-18 | | Pengujian Sistem | | Blackbox | |
| Memilih data surat dan menekan tombol “SPPD 1” | SKPL-10 | | PDHUPL-19 | | Pengujian Sistem | | Blackbox | |
| Memilih data surat dan menekan tombol “SPPD 2” | SKPL-10 | | PDHUPL-20 | | Pengujian Sistem | | Blackbox | |
| Cetak laporan rekap | Memilih pilihan “Data Pegawai” di *combo box* pilih jenis cetak | SKPL-11 | | PDHUPL-21 | | Pengujian Sistem | | Blackbox | |
| Memilih pilihan “Data surat” di *combo box* pilih jenis cetak | SKPL-11 | | PDHUPL-22 | | Pengujian Sistem | | Blackbox | |
| Cetak laporan rekap data surat berdasarkan nama pegawai | Nama yang dipilih memiliki riwayat perjalanan dinas | SKPL-12 | | | PDHUPL-23 | | Pengujian Sistem | | Blackbox | |
| Nama yang dipilih tidak memiliki riwayat perjalanan dinas | SKPL-12 | | | PDHUPL-24 | | Pengujian Sistem | | Blackbox | |
| Cetak laporan rekap data surat berdasarkan periode tanggal | Periode tanggal yang dipilih memiliki riwayat perjalanan dinas | SKPL-13 | | | PDHUPL-25 | | Pengujian Sistem | | Blackbox | |
| Periode tanggal yang dipilih tidak memiliki riwayat perjalanan dinas | SKPL-13 | | | PDHUPL-26 | | Pengujian Sistem | | Blackbox | |
| Cetak Semua laporan rekap data surat | Menekan tombol cetak pada pilhan cetak semua data surat | SKPL-14 | | | PDHUPL-27 | | Pengujian Sistem | | Blackbox | |

### **Deskripsi dan Hasil Pengujian**

Tabel 5.3 Hasil Pengujian PDHUPL-01

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-01 | | |
| Nama Butir Uji | Masuk dengan pengguna dan kata sandi yang terdaftar | | |
| Tujuan | Melihat reaksi sistem apabila memasukan pengguna dan kata sandi yang terdaftar | | |
| Kondisi Awal | Pengguna masuk ke *form*  login | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memasukan pengguna dan kata sandi | | | |
| 2. Menekan tombol masuk | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Username=admin | Sistem mengaktifkan fitur menu utama | kembali ke *form* menu utama dan mengaktifkan fitur menu utama | OK |
| Password=123456 |

Tabel 5.4 Hasil Pengujian PDHUPL-02

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-02 | | |
| Nama Butir Uji | Masuk dengan pengguna dan kata sandi yang tidak terdaftar | | |
| Tujuan | Melihat reaksi sistem apabila memasukan pengguna dan kata sandi yang tidak terdaftar | | |
| Kondisi Awal | Pengguna masuk ke *form*  login | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memasukan pengguna dan kata sandi | | | |
| 2. Menekan tombol masuk | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Username=admin | Menamplikan pesan pengguna dan kata sandi tidak terdaftar | Aplikasi menamplikan pesan pengguna dan kata sandi tidak terdaftar . kembali ke *form* login | OK |
| Password=admin |

Tabel 5.5 Hasil Pengujian PDHUPL-03

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-03 | | |
| Nama Butir Uji | Mengubah data sekolah dengan mengisi semua kolom yang wajib diisi | | |
| Tujuan | Melihat reaksi sistem apabila mengisi semua kolom yang wajib diisi | | |
| Kondisi Awal | Pengguna masuk ke *form* data sekolah | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memasukan data sekolah | | | |
| 2. menekan tombol simpan | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Data Sekolah | Data tersimpan di *database* dan menampilkan pesan bahwa data berhasil disimpan | Data tersimpan di *database* dan menampilkan pesan bahwa data berhasil disimpan | OK |

Tabel 5.6 Hasil Pengujian PDHUPL-04

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-04 | | |
| Nama Butir Uji | Mengubah data sekolah tanpa mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) | | |
| Tujuan | Melihat reaksi sistem apabila tidak mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) | | |
| Kondisi Awal | Pengguna masuk ke *form* data sekolah | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memasukan data sekolah | | | |
| 2. menekan tombol simpan | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Data Sekolah | Data tidak tersimpan di *database* dan menampilkan pesan bahwa data yang wajib diisi tidak boleh kosong | Data tidak tersimpan di *database* dan menampilkan pesan bahwa data wajib diisi tidak boleh kosong | OK |

Tabel 5.7 Hasil Pengujian PDHUPL-05

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-05 | | |
| Nama Butir Uji | Memilih gambar kop surat di komputer/laptop | | |
| Tujuan | Melihat apakah data gambar bisa tersimpan di *database* dan bisa di tampilkan di *form* data sekolah | | |
| Kondisi Awal | Pengguna masuk ke *form* data sekolah | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memilih kop surat | | | |
| 2. Menekan tombol simpan | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Gambar kop surat | Data tersimpan di *database* dan gambar di tampilkan di *form* data sekolah | Data tersimpan di *database* dan gambar di tampilkan di *form* data sekolah | OK |

Tabel 5.8 Hasil Pengujian PDHUPL-06

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-06 | | |
| Nama Butir Uji | Memasukan data pegawai dengan mengisi semua kolom yang wajib diisi | | |
| Tujuan | Melihat reaksi sistem apabila mengisi semua kolom yang wajib diisi | | |
| Kondisi Awal | Pengguna masuk ke *form* tambah pegawai | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memasukan data pegawai | | | |
| 2. Menekan tombol simpan | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Data Pegawai | Data tersimpan di *database* dan menampilkan pesan bahwa data berhasil disimpan | Data tersimpan di *database* dan menampilkan pesan bahwa data berhasil disimpan | OK |

Tabel 5.9 Hasil Pengujian PDHUPL-07

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-07 | | |
| Nama Butir Uji | Memasukan data pegawai tanpa mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) | | |
| Tujuan | Melihat reaksi sistem apabila tidak mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) | | |
| Kondisi Awal | Pengguna masuk ke *form* tambah data pegawai | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memasukan data pegawai | | | |
| 2. Menekan tombol simpan | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Data pegawai | Data tidak tersimpan di *database* dan menampilkan pesan bahwa data yang wajib diisi tidak boleh kosong | Data tidak tersimpan di *database* dan menampilkan pesan bahwa data yang wajib diisi tidak boleh kosong | OK |

Tabel 5.10 Hasil Pengujian PDHUPL-08

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-08 | | |
| Nama Butir Uji | Mengubah data pegawai dengan mengisi semua kolom yang wajib diisi | | |
| Tujuan | Melihat reaksi sistem apabila mengisi semua kolom yang wajib diisi | | |
| Kondisi Awal | Pengguna masuk ke *form* ubah data pegawai | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Mengubah data pegawai | | | |
| 2. Menekan tombol simpan | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Data pegawai | Data tersimpan di *database* dan menampilkan pesan bahwa data berhasil disimpan | Data tersimpan di *database* dan menampilkan pesan bahwa data berhasil disimpan | OK |

Tabel 5.11 Hasil Pengujian PDHUPL-09

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-09 | | |
| Nama Butir Uji | Mengubah data pegawai tanpa mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) | | |
| Tujuan | Melihat reaksi sistem apabila tidak mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) | | |
| Kondisi Awal | Pengguna masuk ke *form* ubah data pegawai | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Mengubah data pegawai | | | |
| 2. Menekan tombol simpan | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Data pegawai | Data tidak tersimpan di *database* dan menampilkan pesan bahwa data yang wajib diisi tidak boleh kosong | Data tidak tersimpan di *database* dan menampilkan pesan bahwa data yang wajib diisi tidak boleh kosong | OK |

Tabel 5.12 Hasil Pengujian PDHUPL-10

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-10 | | |
| Nama Butir Uji | Memilih pilihan “YA” saat muncul konfirmasi hapus data | | |
| Tujuan | Melihat reaksi sistem apabila memilih pilihan “YA” | | |
| Kondisi Awal | Pengguna masuk ke *form* data pegawai | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1.Memilih data pegawai | | | |
| 2. Menekan tombol hapus | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Data pegawai | Data di *database* dihapus dan menampilkan pesan bahwa data terpilih telah dihapus | Data di *database* dihapus dan menampilkan pesan bahwa data terpilih telah dihapus | OK |

Tabel 5.13 Hasil Pengujian PDHUPL-11

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-11 | | |
| Nama Butir Uji | Memilih pilihan “tidak” saat muncul konfirmasi hapus data | | |
| Tujuan | Melihat reaksi sistem apabila memilih pilihan “tidak” | | |
| Kondisi Awal | Pengguna masuk ke *form* data pegawai | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1.Memilih data pegawai | | | |
| 2. Menekan tombol hapus | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Data pegawai | Data di *database* tidak terhapus hapus dan kembali ke *form* data pegawai | Data di *database* tidak terhapus hapus dan kembali ke *form* data pegawai | OK |

Tabel 5.14 Hasil Pengujian PDHUPL-12

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-12 | | |
| Nama Butir Uji | Memasukan data surat dengan mengisi semua kolom yang wajib diisi | | |
| Tujuan | Melihat reaksi sistem apabila mengisi semua kolom yang wajib diisi | | |
| Kondisi Awal | Pengguna masuk ke *form* data surat | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memasukan data surat | | | |
| 2. Menekan tombol simpan | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Data Surat | Data tersimpan di *database* dan menampilkan pesan bahwa data berhasil disimpan | Data tersimpan di *database* dan menampilkan pesan bahwa data berhasil disimpan | OK |

Tabel 5.15 Hasil Pengujian PDHUPL-13

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-13 | | |
| Nama Butir Uji | Memasukan data surat tanpa mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) | | |
| Tujuan | Melihat reaksi sistem apabila tidak mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) | | |
| Kondisi Awal | Pengguna masuk ke *form* data surat | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memasukan data surat | | | |
| 2. Menekan tombol simpan | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Data surat | Data tidak tersimpan di *database* dan menampilkan pesan bahwa data yang wajib diisi tidak boleh kosong | Data tidak tersimpan di *database* dan menampilkan pesan bahwa data yang wajib diisi tidak boleh kosong | OK |

Tabel 5.16 Hasil Pengujian PDHUPL-14

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-14 | | |
| Nama Butir Uji | Mengubah data surat dengan mengisi semua kolom yang wajib diisi | | |
| Tujuan | Melihat reaksi sistem apabila mengisi semua kolom yang wajib diisi | | |
| Kondisi Awal | Pengguna masuk ke *form* data surat | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Mengubah data surat | | | |
| 2. Menekan tombol simpan | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Data surat | Data tersimpan di *database* dan menampilkan pesan bahwa data berhasil disimpan | Data tersimpan di *database* dan menampilkan pesan bahwa data berhasil disimpan | OK |

Tabel 5.17 Hasil Pengujian PDHUPL-15

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-15 | | |
| Nama Butir Uji | Mengubah data surat tanpa mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) | | |
| Tujuan | Melihat reaksi sistem apabila tidak mengisi semua kolom yang wajib diisi (Mengosongkan salah satu kolom) | | |
| Kondisi Awal | Pengguna masuk ke *form* ubah data pegawai | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Mengubah data surat | | | |
| 2. Menekan tombol simpan | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Data surat | Data tidak tersimpan di *database* dan menampilkan pesan bahwa data yang wajib diisi tidak boleh kosong | Data tidak tersimpan di *database* dan menampilkan pesan bahwa data yang wajib diisi tidak boleh kosong | OK |

Tabel 5.18 Hasil Pengujian PDHUPL-16

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-16 | | |
| Nama Butir Uji | Memilih pilihan “YA” saat muncul konfirmasi hapus data | | |
| Tujuan | Melihat reaksi sistem apabila memilih pilihan “YA” | | |
| Kondisi Awal | Pengguna masuk ke *form* data pegawai | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1.Memilih data surat | | | |
| 2. Menekan tombol hapus | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Data surat | Data di *database* dihapus dan menampilkan pesan bahwa data terpilih telah dihapus | Data di *database* dihapus dan menampilkan pesan bahwa data terpilih telah dihapus | OK |

Tabel 5.19 Hasil Pengujian PDHUPL-17

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-17 | | |
| Nama Butir Uji | Memilih pilihan “tidak” saat muncul konfirmasi hapus data | | |
| Tujuan | Melihat reaksi sistem apabila memilih pilihan “tidak” | | |
| Kondisi Awal | Pengguna masuk ke *form* data pegawai | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1.Memilih data surat | | | |
| 2. Menekan tombol hapus | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Data surat | Data di *database* tidak terhapus hapus dan kembali ke *form* data pegawai | Data di *database* tidak terhapus hapus dan kembali ke *form* data pegawai | OK |

Tabel 5.20 Hasil Pengujian PDHUPL-18

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-18 | | |
| Nama Butir Uji | Memilih data surat dan menekan tombol “Surat Tugas” | | |
| Tujuan | Melihat apakah hasil cetak sesuai dengan tombol yang ditekan | | |
| Kondisi Awal | Pengguna masuk ke *form* cetak surat | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1.Memilih data surat | | | |
| 2. Menekan tombol “Surat Tugas” | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Pilihan data surat | Hasil keluaran sesuai dengan data yang dipilih dan tombol yang ditekan | Aplikasi menampilkan format surat tugas sesuai dengan data yang dipilih | OK |

Tabel 5.21 Hasil Pengujian PDHUPL-19

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-19 | | |
| Nama Butir Uji | Memilih data surat dan menekan tombol “SPPD1” | | |
| Tujuan | Melihat apakah hasil cetak sesuai dengan tombol yang ditekan | | |
| Kondisi Awal | Pengguna masuk ke *form* cetak surat | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memilih data surat | | | |
| 2. Menekan tombol “SPPD1” | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Pilihan data surat | Hasil keluaran sesuai dengan data yang dipilih dan tombol yang ditekan | Aplikasi menampilkan format SPPD halaman 1 sesuai dengan data yang dipilih | OK |

Tabel 5.22 Hasil Pengujian PDHUPL-20

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-20 | | |
| Nama Butir Uji | Memilih data surat dan menekan tombol “SPPD2” | | |
| Tujuan | Melihat apakah hasil cetak sesuai dengan tombol yang ditekan | | |
| Kondisi Awal | Pengguna masuk ke *form* cetak surat | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memilih data surat | | | |
| 2. Menekan tombol “SPPD2” | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Pilihan data surat | Hasil keluaran sesuai dengan data yang dipilih dan tombol yang ditekan | Aplikasi menampilkan format SPPD halaman 2 sesuai dengan data yang dipilih | OK |

Tabel 5.23 Hasil Pengujian PDHUPL-21

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-21 | | |
| Nama Butir Uji | Memilih pilihan “Data Pegawai” di combo box pilih jenis cetak | | |
| Tujuan | Melihat apakah aplikasi akan meanmpilkan hasil sesuai dengan pilihan pengguna | | |
| Kondisi Awal | Pengguna masuk ke *form* cetak laporan rekap | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memilih jenis cetak | | | |
| 2. Menekan pilihan “Data Pegawai” | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Pilhan cetak data pegawai | Aplikasi menampilkan laporan rekap data pegawai | Aplikasi menampilkan cetak laporan rekap data pegawai | OK |

Tabel 5.24 Hasil Pengujian PDHUPL-22

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-22 | | |
| Nama Butir Uji | Memilih pilihan “Data surat” di combo box pilih jenis cetak | | |
| Tujuan | Melihat apakah aplikasi akan meanmpilkan hasil sesuai dengan pilihan pengguna | | |
| Kondisi Awal | Pengguna masuk ke *form* cetak laporan rekap | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memilih jenis cetak | | | |
| 2. Menekan pilihan “Data Surat” | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Pilhan cetak data surat | Pengguna diarahkan ke *form* cetak laporan rekap data surat | Aplikasi akan menutup *form* cetak laporan rekap dan menampilkan *form* cetak laporan rekap data surat | OK |

Tabel 5.25 Hasil Pengujian PDHUPL-23

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-23 | | |
| Nama Butir Uji | Nama yang dipilih memiliki riwayat perjalanan dinas | | |
| Tujuan | Melihat reaksi sistem apabila nama pegawai yang dipilih memiliki riwayat perjalanan dinas pada saat cetak laporan rekap data surat berdasarkan nama pegawai | | |
| Kondisi Awal | Pengguna masuk ke *form* cetak laporan rekap data surat | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memilih cetak berdasarkan nama pegawai | | | |
| 2. Memilih nama pegawai | | | |
| 3. Menekan tombol cetak | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Pilhan nama pegawai | Aplikasi menampilkan laporan rekap data surat berdasarkan nama pegawai | Aplikasi menampilkan laporan rekap data surat berdasarkan nama pegawai yang dipilih | OK |

Tabel 5.26 Hasil Pengujian PDHUPL-24

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-24 | | |
| Nama Butir Uji | Nama yang dipilih tidak memiliki riwayat perjalanan dinas | | |
| Tujuan | Melihat reaksi sistem apabila nama pegawai yang dipilih tidak memiliki riwayat perjalanan dinas pada saat cetak laporan rekap data surat berdasarkan nama pegawai | | |
| Kondisi Awal | Pengguna masuk ke *form* cetak laporan rekap data surat | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memilih cetak berdasarkan nama pegawai | | | |
| 2. Memilih nama pegawai | | | |
| 3. Menekan tombol cetak | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Pilhan nama pegawai | Aplikasi menampilkan pesan bahwa nama pegawai terpilih tidak memiliki riwayat pejalanan dinas | Aplikasi menampilkan pesan bahwa nama pegawai terpilih tidak memiliki riwayat pejalanan dinas | OK |

Tabel 5.27 Hasil Pengujian PDHUPL-25

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-25 | | |
| Nama Butir Uji | Periode tanggal yang dipilih memiliki riwayat perjalanan dinas | | |
| Tujuan | Melihat reaksi sistem apabila periode tanggal yang dipilih memiliki riwayat perjalanan dinas pada saat cetak laporan rekap data surat berdasarkan periode tanggal | | |
| Kondisi Awal | Pengguna masuk ke *form* cetak laporan rekap data surat | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memilih cetak berdasarkan periode tanggal | | | |
| 2. Menentukan periode tanggal | | | |
| 3. Menekan tombol cetak | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Periode tanggal yang akan di cetak | Aplikasi menampilkan laporan rekap data surat berdasarkan periode tanggal | Aplikasi menampilkan laporan rekap data surat berdasarkan periode tanggal yang dipilih | OK |

Tabel 5.28 Hasil Pengujian PDHUPL-26

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-26 | | |
| Nama Butir Uji | Periode tanggal yang dipilih tidak memiliki riwayat perjalanan dinas | | |
| Tujuan | Melihat reaksi sistem apabila periode tanggal yang dipilih tidak memiliki riwayat perjalanan dinas pada saat cetak laporan rekap data surat berdasarkan periode tanggal | | |
| Kondisi Awal | Pengguna masuk ke *form* cetak laporan rekap data surat | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memilih cetak berdasarkan periode tanggal | | | |
| 2. Menentukan periode tanggal | | | |
| 3. Menekan tombol cetak | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Periode tanggal yang akan di cetak | Aplikasi menampilkan pesan bahwa periode tanggal terpilih tidak memiliki riwayat pejalanan dinas | Aplikasi menampilkan pesan bahwa periode tanggal terpilih tidak memiliki riwayat pejalanan dinas | OK |

Tabel 5.29 Hasil Pengujian PDHUPL-27

|  |  |  |  |
| --- | --- | --- | --- |
| Identifikasi | PDHUPL-27 | | |
| Nama Butir Uji | Menekan tombol cetak pada pilhan cetak semua data surat | | |
| Tujuan | Melihat reaksi sistem apabila menekan tombol cetak pada saat cetak semua laporan rekap data surat | | |
| Kondisi Awal | Pengguna masuk ke *form* cetak laporan rekap data surat | | |
| Tanggal Pengujian | 26/06/2021 | | |
| Penguji | Agus Eka Purnama | | |
| Skenario | | | |
| 1. Memilih cetak semua | | | |
| 3. Menekan tombol cetak | | | |
| Hasil | | | |
| Data yang diberikan | Hasil yang diharapkan | Pengamatan | Kesimpulan |
| Data surat | Aplikasi menampilkan pesan bahwa periode tanggal terpilih tidak memiliki riwayat pejalanan dinas | Aplikasi menampilkan pesan bahwa periode tanggal terpilih tidak memiliki riwayat pejalanan dinas | OK |

# **BAB VI KESIMPULAN DAN SARAN**

## **Kesimpulan**

Aplikasi surat tugas dan surat tugas perjalanan dinas ini dibangun untuk membantu tenaga administrasi selaku admin aplikasi ini untuk :

1. Mempermudah proses pembuatan surat yang sebelumnya menggunakan aplikasi *Microsoft Word* dimana terdapat beberapa permasalahan diantaranya, admin harus melakukan pengeditan setiap kali akan membuat surat, pencarian data setiap pegawai yang akan melakukan perjalanan dinas pada *Microsoft Excel*, dan kesalahan-kesalahan pengetikan dalam menginputkan data sehingga pembuatan surat dilakukan berulang – ulang kini menjadi lebih mudah karena dengan aplikasi ini admin hanya perlu memilih data pegawai yang sudah disimpan sebelumnya sehing terhindar dari masalah diatas.
2. Proses penyimpanan arsip lebih efisien karena dengan aplikasi ini data surat akan disimpan pada perangkat penyimpanan laptop atau komputer sehingga admin tidak perlu mencatat riwayat surat pada buku sehingga terhindar dari lupa mencatat atau salah menuliskan nomor surat karena dengan aplikasi ini riwayat surat akan otomatis tersimpan pada *database*. Selain itu dengan aplikasi ini sekolah tidak memerlukan buku yang tebal untuk pencatatan riwayat surat sehingga terhindar dari resiko buku hilang dan rusak.
3. Mempercepat proses perekapan dan pencetakan dimana sebelumnya membutuhkan waktu yang lama karena harus membuka kembali satu-persatu arsip perjalanan dinas yang sangat banyak dan melakukan pengetikan ulang pada aplikasi M*icrosoft Word*. Dengan aplikasi ini proses tersebut bisa di selesaikan dalam kurun waktu kurang dari lima menit.

## **Saran**

Pada penulisan skripsi ini penulisan meyakini tentu masih banyak terdapat kesalahan dan kekurangan, baik dalam sisi penulisan dokumen serta pembuatan aplikas yang dibuat ini masih perlu banyak pengembangan. Maka saran yang diberikan penulis untuk membantu pengembangan aplikasi ini kedepannya antara lain:

1. Memberikan tampilan user interface yang lebih menarik.
2. Perlu adanya pengembangan aplikasi dalam segi fungsional agar aplikasi tidak hanya mengelola surat tugas dan SPPD saja.
3. Memberikan penambahan data pegawai seperti foto pegawai sehingga data pegawai menjadi lebih lengkap.
4. Penyesuaian hasil cetak sesuaI dengan kebutuhan sekolah masing-masing.
5. Pengembangan ruang lingkup aplikasi agar aplikasi tidak hanya bisa digunakan untuk satu sekolah saja tetapi juga bisa digunakan untuk banyak sekolah bahkan untuk instansi lain.
6. Pengembangan fitur sehingga aplikasi dapat digunakan secara online.

# **DAFTAR PUSTAKA**

Aulina, L. (n.d.). Bahasa Pemrograman Delphi. In *Delphi*.

D, B., & Sriwinar. (2021). *Implementasi J-Query Pada Aplikasi Monitoring Kehadiran Siswa Untuk Orang Tua Dengan Sms*. *06*(01), 44–52.

Fadillah, F. (2018). Aplikasi Pengajuan dan Monitoring Surat Perintah Perjalanan Dinas Berbasis WEB. *Teknologi Informasi*. http://library.oum.edu.my/repository/725/2/Chapter\_1.pdf

Fonda, H. (2019). Penerapan Aplikasi Surat Perintah Perjalanan Sekretariat Daerah Kabupaten Kepulauan Meranti. *Jurnal Ilmu Komputer*, *8*(1), 144–149.

Hafid Mukhlasin. (2008). *Tutorial Delphi For Newbie Be Master Without Teacher*. *52*, 52. http://himamasta.ub.ac.id/wp-content/uploads/2015/03/Tutorial-Delphi-7.pdf

Handika, R., & Widjaja, A. (2018). Rancangan Sistem Informasi Surat Perintah Perjalanan Dinas Pada Balai Teknologi Air Dan Limbah Dengan Berbasis Aplikasi Web. *Jurnal IDEALIS*, *1*(1), 416–422.

Heryanto, I. (2017). *Membuat Database Dengan Microsoft Access*. Informatika Bandung.

Indriawan, E., Liasie, F. A., & Sayugo, T. (2011). *Mastering Delphi XE* (M. Agustina (ed.)). CV. Andi.

ir. Inge Martina. (2012). *36 Jam Belajar Komputer*.

Mamase, S. (2016). Rancang Bangun Sistem Informasi Surat Perintah Perjalanan Dinas. *Jurnal Teknik Informatika Dan Sistem Informasi*, *6*(2), 7–11.

Mendikbud. (2013). *Permendikbud No 6 Tahun 2013 tentang Tata Naskah Dinas di Lingkungan Kemendikbud*.

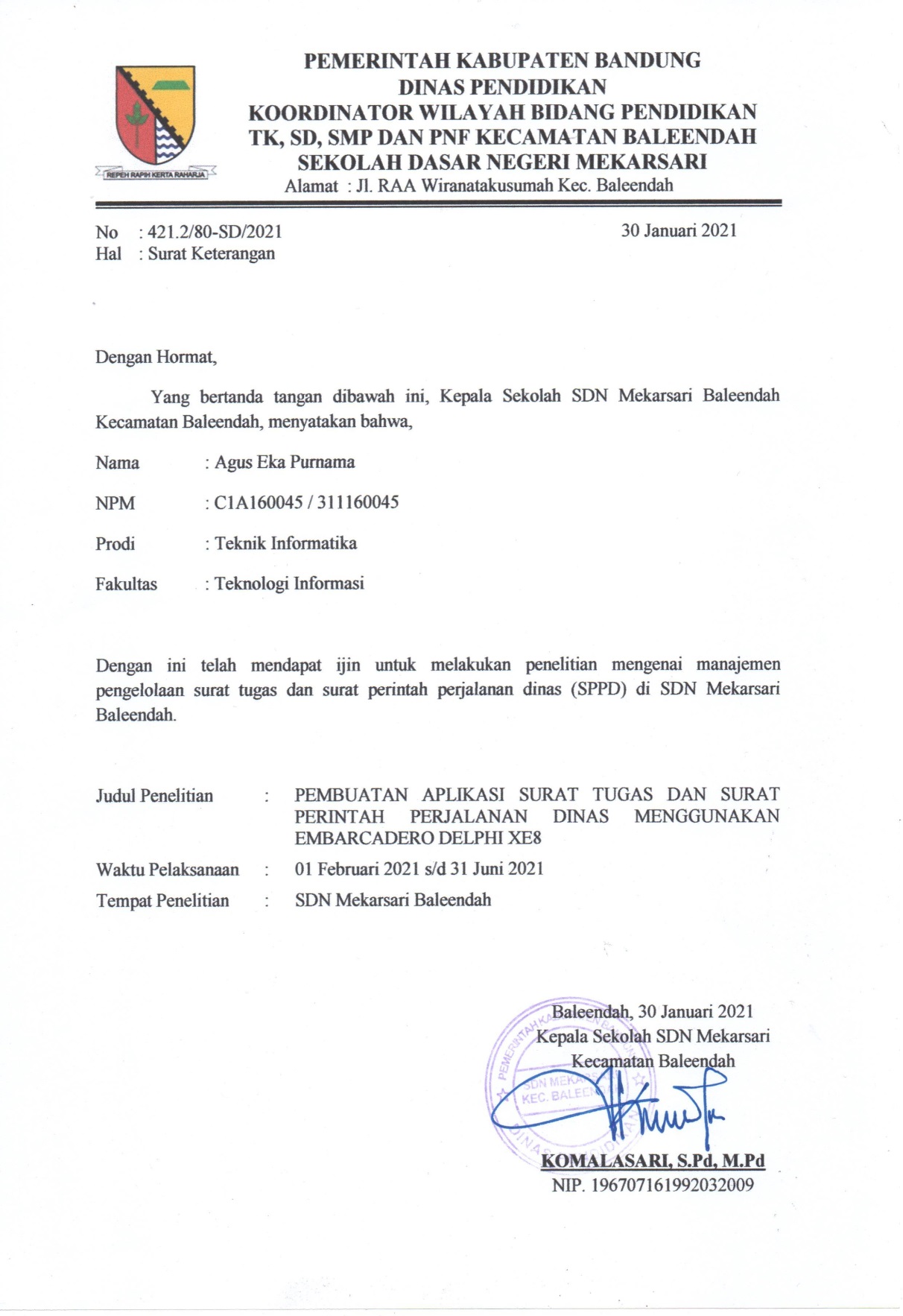
PERMENKEU. (2007). *PERATURAN MENTERI KEUANGAN REPUBLIK INDONESIA NOMOR*. 5–7. https://toolsfortransformation.net/wp-content/uploads/2017/05/UNDANG-UNDANG-NOMOR-26-TAHUN-2007.pdf

Rosa, A. ., & Shalahudin, M. (2018). *Rekayasa Perangkat Lunak*. Informatika Bandung.

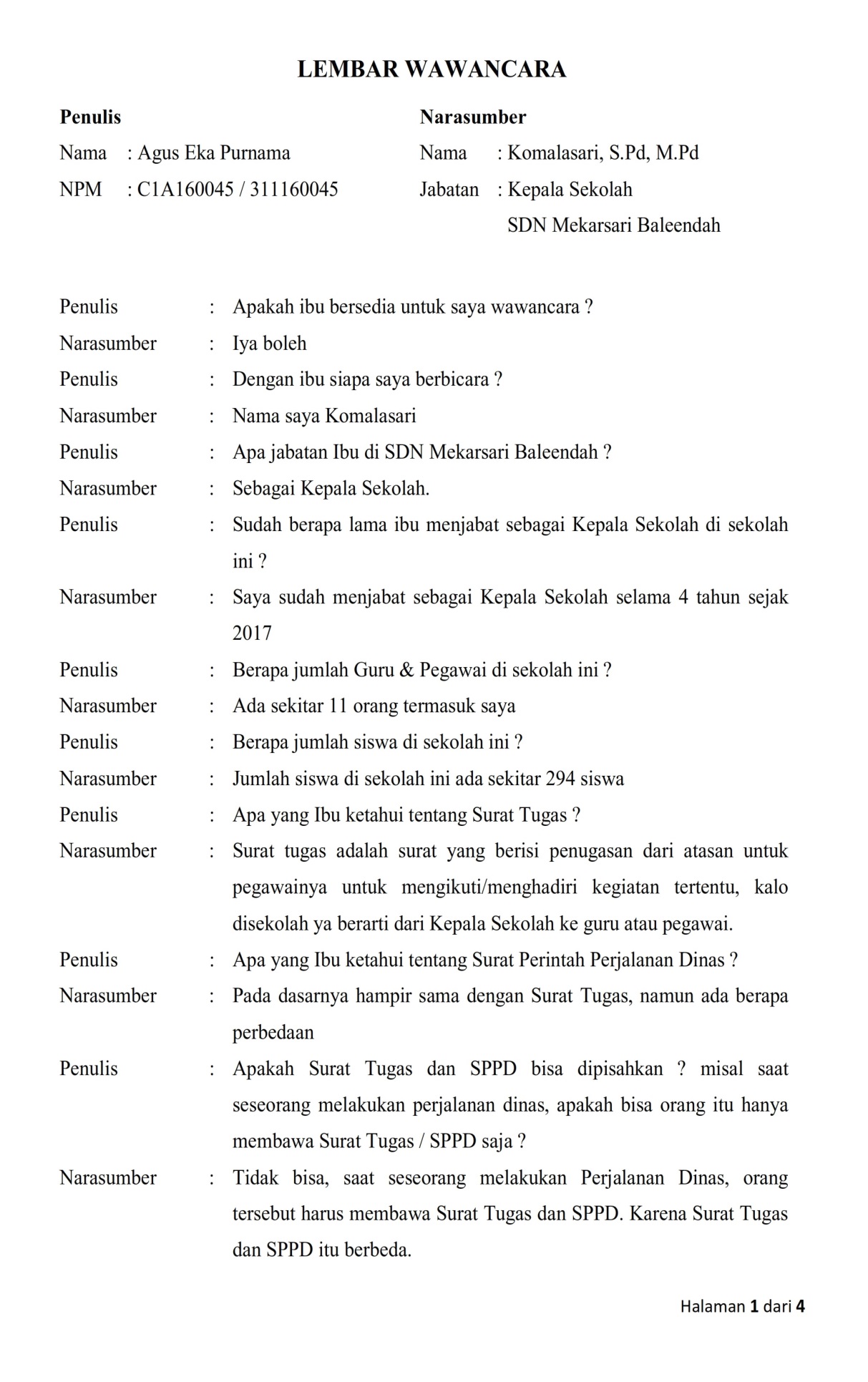
Suharyanto, C. E., Chandra, J. E., & Gunawan, F. E. (2017). Perancangan Sistem Informasi Penggajian Terintegrasi Berbasis Web (Studi Kasus di Rumah Sakit St. Elisabeth). *Jurnal Nasional Teknologi Dan Sistem Informasi*, *3*(2), 225–232. https://doi.org/10.25077/teknosi.v3i2.2017.225-232

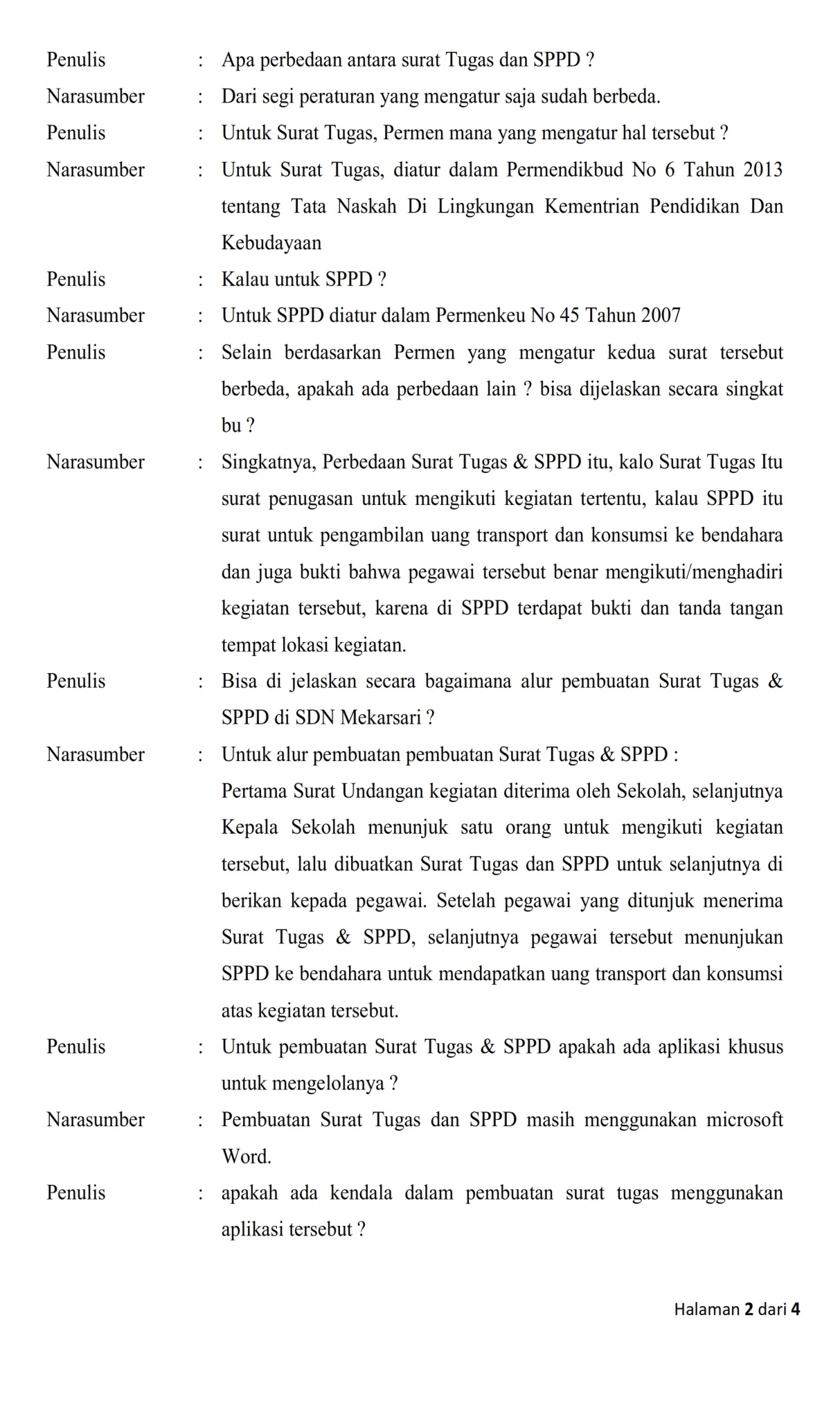
# **LAMPIRAN**

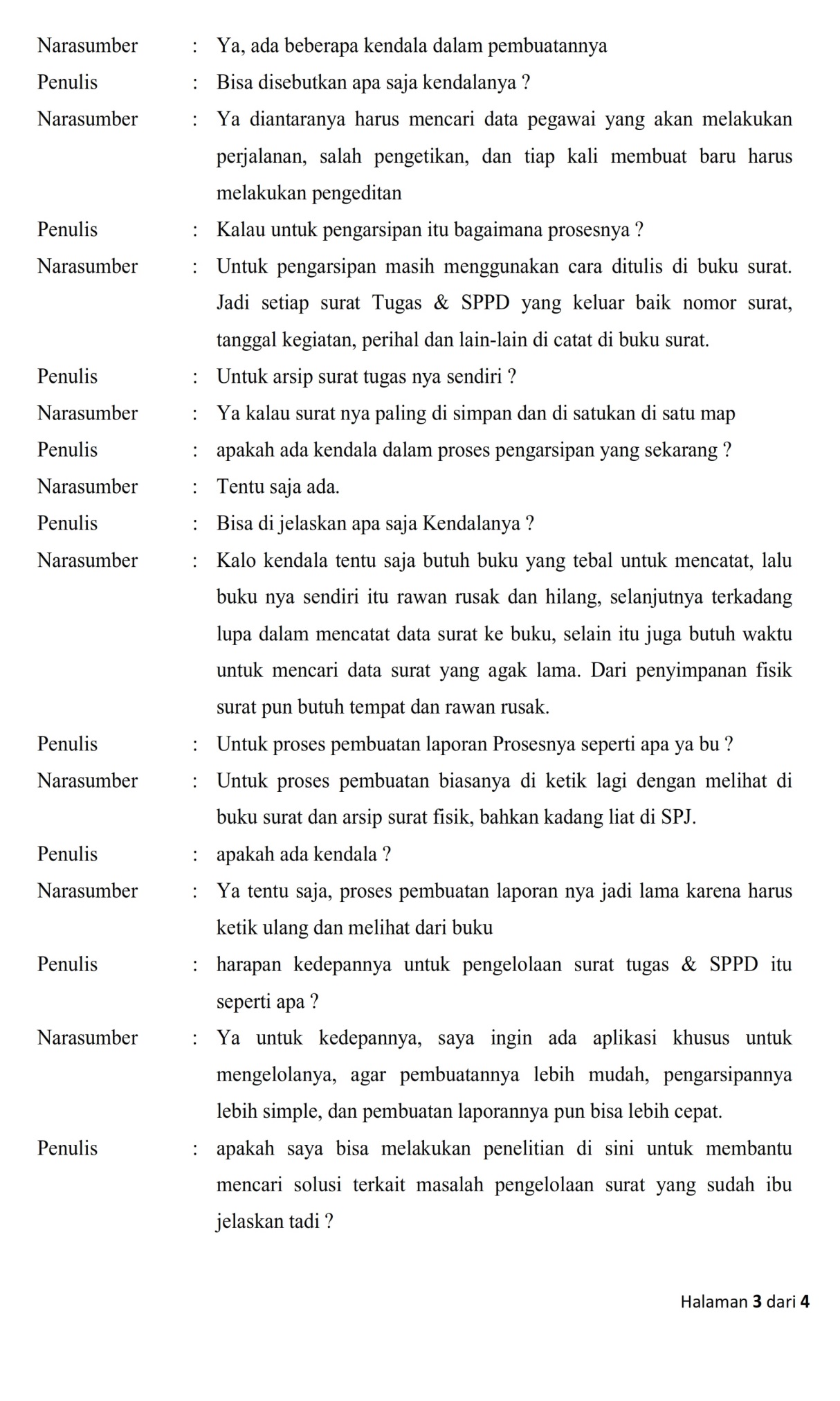
**Lampiran 1 Surat Ijin Penelitian**

****

**Lampiran 2 Lembar Wawancara**









**Lampiran 3 *Source Code***

*Form Login*

var

Masuk: TMasuk;

implementation

{$R \*.dfm}

uses FrmKoneksi, FrmMenutama;

procedure TMasuk.btnBatalClick(Sender: TObject);

begin

MenuUtama.Show;

Masuk.Close;

end;

procedure TMasuk.btnSimpanClick(Sender: TObject);

Begin

Begin

Koneksi.adoqMasuk.Open;

Koneksi.adoqMasuk.First;

While not Koneksi.adoqMasuk.Eof do

if(ePengguna.Text=koneksi.adoqMasuk['Pengguna']) and (eKataSandi.Text= koneksi.adoqMasuk ['KataSandi']) Then

Begin

MenuUtama.Show;

Masuk.Close;

MenuUtama.lHalo.Caption:='Selamat Datang '+ePengguna.Text;

MenuUtama.lMasuk.Visible:=False;

MenuUtama.lHalo.Visible:=True;

MenuUtama.lKeluar.Visible:=True;

MenuUtama.Menu.Visible:=True;

exit;

end

else

begin

Koneksi.adoqMasuk.Next;

if(ePengguna.Text=koneksi.adoqMasuk['Pengguna']) and (eKataSandi.Text=koneksi.adoqMasuk['KataSandi']) Then

Begin

MenuUtama.Show;

Masuk.Close;

MenuUtama.lHalo.Caption:='Selamat Datang '+ePengguna.Text;

Exit;

end

Else

Begin

ShowMessage('Anda Tidak Terdaftar');

ePengguna.Text:='';

eKataSandi.Text:='';

ePengguna.SetFocus;

end

end;

end;

end;

end.

*Form* Menu Utama

var

MenuUtama: TMenuUtama;

implementation

{$R \*.dfm}

uses FrmDataSekolah, FrmDataPegawai, FrmDataSurat, FrmCetakSurat, FrmMasuk,FrmPilihCetak;

procedure TMenuUtama.btnDataPegawaiClick(Sender: TObject);

begin

DataPegawai.show;

MenuUtama.Enabled:=False;

end;

procedure TMenuUtama.btnDataSekolahClick(Sender: TObject);

begin

DataSekolah.show;

MenuUtama.Enabled:=False;

end;

procedure TMenuUtama.btnSuratTugasClick(Sender: TObject);

begin

CetakSurat.show;

MenuUtama.Enabled:=False;

end;

procedure TMenuUtama.Button1Click(Sender: TObject);

begin

CetakLaporan.Show;

end;

procedure TMenuUtama.Button2Click(Sender: TObject);

begin

DataSurat.show;

MenuUtama.Enabled:=False;

end;

procedure TMenuUtama.Button3Click(Sender: TObject);

begin

Application.Terminate;

end;

procedure TMenuUtama.DataPegawaiClick(Sender: TObject);

begin

DataPegawai.show;

MenuUtama.Enabled:=False;

end;

procedure TMenuUtama.DataSekolahClick(Sender: TObject);

begin

DataSekolah.show;

MenuUtama.Enabled:=False;

end;

procedure TMenuUtama.Image2Click(Sender: TObject);

begin

DataSekolah.show;

MenuUtama.Enabled:=False;

end;

procedure TMenuUtama.Image2MouseEnter(Sender: TObject);

begin

Label1.Visible:=True;

end;

procedure TMenuUtama.Image2MouseLeave(Sender: TObject);

begin

Label1.Visible:=False;

end;

procedure TMenuUtama.Image3Click(Sender: TObject);

begin

DataPegawai.show;

MenuUtama.Enabled:=False;

end;

procedure TMenuUtama.Image3MouseEnter(Sender: TObject);

begin

Label2.Visible:=True;

end;

procedure TMenuUtama.Image3MouseLeave(Sender: TObject);

begin

Label2.Visible:=False;

end;

procedure TMenuUtama.Image4Click(Sender: TObject);

begin

DataSurat.show;

MenuUtama.Enabled:=False;

end;

procedure TMenuUtama.Image4MouseEnter(Sender: TObject);

begin

Label3.Visible:=True;

end;

procedure TMenuUtama.Image4MouseLeave(Sender: TObject);

begin

Label3.Visible:=False;

end;

procedure TMenuUtama.Image4MouseMove(Sender: TObject; Shift: TShiftState; X,

Y: Integer);

begin

Label3.Visible:=False;

end;

procedure TMenuUtama.Image5Click(Sender: TObject);

begin

CetakLaporan.Show;

end;

procedure TMenuUtama.Image5MouseEnter(Sender: TObject);

begin

Label5.Visible:=True;

end;

procedure TMenuUtama.Image5MouseLeave(Sender: TObject);

begin

Label5.Visible:=False;

end;

procedure TMenuUtama.Image6Click(Sender: TObject);

begin

CetakSurat.show;

MenuUtama.Enabled:=False;

end;

procedure TMenuUtama.Image6MouseEnter(Sender: TObject);

begin

Label4.Visible:=True;

end;

procedure TMenuUtama.Image6MouseLeave(Sender: TObject);

begin

Label4.Visible:=False;

end;

procedure TMenuUtama.lKeluarClick(Sender: TObject);

begin

if (application.messagebox('Apakah Anda Yakin Akan Keluar ?','Peringatan', mb\_YesNo) = idYes) Then

Begin

MenuUtama.lMasuk.Visible:=True;

MenuUtama.lHalo.Visible:=False;

MenuUtama.lKeluar.Visible:=False;

MenuUtama.Menu.Visible:=False;

End

Else

Abort;

end;

procedure TMenuUtama.lMasukClick(Sender: TObject);

begin

Masuk.Show;

end;

end.

*Form* Data Sekolah

var

DataSekolah: TDataSekolah;

Stream:TMemoryStream;

Jpeg:TJpegImage;

implementation

{$R \*.dfm}

Uses FrmKoneksi, FrmMenutama;

procedure TDataSekolah.BtnBatalClick(Sender: TObject);

begin

dbeNPSN.Enabled:=False;

dbeNamaSekolah.Enabled:=False;

dbeTelp.Enabled:=False;

dbeEmail.Enabled:=False;

dbmAlamat.Enabled:=False;

dbeKecamatan.Enabled:=False;

dbeKabKota.Enabled:=False;

dblNamaKepsek.Enabled:=False;

dbeNIP.Enabled:=False;

btnSimpan.Enabled:=False;

btnBatal.Enabled:=False;

btnUnggah.Enabled:=False;

btnUbah.Enabled:=True;

btnKeluar.Enabled:=True;

end;

procedure TDataSekolah.btnKeluarClick(Sender: TObject);

begin

DataSekolah.Close;

MenuUtama.Show;

MenuUtama.Enabled:=true;

end;

procedure TDataSekolah.btnSimpanClick(Sender: TObject);

begin

Koneksi.adoqSekolah.Open;

Stream:=TMemoryStream.Create;

iKop.Picture.Graphic.SaveToStream(stream);

Stream.Position:=0;

Koneksi.AdoqSekolah.Edit;

Koneksi.AdoqSekolah.FieldByName('NPSN').AsInteger:= StrToInt(dbeNPSN.Text);

Koneksi.AdoqSekolah.FieldByName('NamaSekolah').AsString:= dbeNamaSekolah.Text;

Koneksi.AdoqSekolah.FieldByName('Telp').AsString:=dbeTelp.Text;

Koneksi.AdoqSekolah.FieldByName('Email').AsString:=dbeEmail.Text;

Koneksi.AdoqSekolah.FieldByName('Alamat').AsString:=dbmAlamat.Text;

Koneksi.AdoqSekolah.FieldByName('Kecamatan').AsString:= dbeKecamatan.Text;

Koneksi.AdoqSekolah.FieldByName('Kab/Kota').AsString:=dbeKabKota.Text;

Koneksi.AdoqSekolah.FieldByName('KepalaSekolah').AsString:= dblNamaKepsek.Text;

Koneksi.AdoqSekolah.FieldByName('NIP').AsString:=dbeNIP.Text;

TblobField(Koneksi.AdoqSekolah.FieldByName('Kop')).LoadFromStream (Stream);

Stream.Free;

Koneksi.AdoqSekolah.Post;

showmessage('Data Telah Di Simpan');

dbeNPSN.Enabled:=False;

dbeNamaSekolah.Enabled:=False;

dbeTelp.Enabled:=False;

dbeEmail.Enabled:=False;

dbmAlamat.Enabled:=False;

dbeKecamatan.Enabled:=False;

dbeKabKota.Enabled:=False;

dblNamaKepsek.Enabled:=False;

dbeNIP.Enabled:=False;

btnSimpan.Enabled:=False;

btnBatal.Enabled:=False;

btnUnggah.Enabled:=False;

btnUbah.Enabled:=True;

btnKeluar.Enabled:=True;

end;

procedure TDataSekolah.btnUbahClick(Sender: TObject);

begin

dbeNPSN.Enabled:=True;

dbeNamaSekolah.Enabled:=True;

dbeTelp.Enabled:=True;

dbeEmail.Enabled:=True;

dbmAlamat.Enabled:=True;

dbeKecamatan.Enabled:=True;

dbeKabKota.Enabled:=True;

dblNamaKepsek.Enabled:=True;

btnSimpan.Enabled:=True;

btnBatal.Enabled:=True;

btnUnggah.Enabled:=True;

btnUbah.Enabled:=False;

btnKeluar.Enabled:=False;

end;

procedure TDataSekolah.btnUnggahClick(Sender: TObject);

begin

if BrowseKop.Execute then

Begin

ePath.Text:=BrowseKop.FileName;

iKop.Picture.LoadFromFile(ePath.text);

End;

end;

procedure TDataSekolah.FormShow(Sender: TObject);

begin

Jpeg:=TJpegImage.Create;

Stream:=TMemoryStream.Create;

TblobField(Koneksi.AdoqSekolah.FieldByName('Kop')).SaveToStream (Stream);

Stream.Position:=0;

Jpeg.LoadFromStream(Stream);

iKop.Picture.Graphic:=Jpeg;

end;

end.

*Form* Data Pegawai

var

DataPegawai: TDataPegawai;

implementation

{$R \*.dfm}

Uses FrmKoneksi, FrmMenutama, FrmTambahPegawai, FrmUbahPegawai;

procedure TDataPegawai.btnHapusClick(Sender: TObject);

begin

if (application.messagebox('Apakah Anda Yakin Akan Menghapus Data Pegawai Terpilih ?','Peringatan',mb\_YesNo)=idYes) Then

Begin

Koneksi.AdoqPegawai.Delete;

Koneksi.AdoqPegawai.Close;

Koneksi.AdoqPegawai.SQL.Clear;

Koneksi.AdoqPegawai.SQL.Add('Select \* From Tbl\_Pegawai');

Koneksi.AdoqPegawai.Open;

showmessage('Data Telah Dihapus');

End

Else

Abort;

end;

procedure TDataPegawai.btnKeluarClick(Sender: TObject);

begin

DataPegawai.Close;

MenuUtama.Show;

MenuUtama.Enabled:=true;

end;

procedure TDataPegawai.btnTambahClick(Sender: TObject);

begin

TambahPegawai.Show;

DataPegawai.Enabled:=False;

TambahPegawai.eNIP.Text:='';

TambahPegawai.eNama.Text:='';

TambahPegawai.CbKepegawaian.Text:='';

TambahPegawai.ePangkat.Text:='';

TambahPegawai.cbGolongan.Text:='';

TambahPegawai.cbJK.Text:='';

TambahPegawai.eTempat.Text:='';

TambahPegawai.cbJabatan.Text:='';

end;

procedure TDataPegawai.btnUbahClick(Sender: TObject);

begin

UbahPegawai.eNIP.Text:=Koneksi.AdoqPegawaiNIPNUPTKID.value;

UbahPegawai.eNama.Text:=Koneksi.adoqPegawaiNama.value;

UbahPegawai.cbKepegawaian.Text:=Koneksi.adoqPegawaiStatus.value;

UbahPegawai.ePangkat.Text:=Koneksi.adoqPegawaiPangkat.value;

UbahPegawai.cbGolongan.Text:=Koneksi.adoqPegawaiGolongan.value;

UbahPegawai.cbJK.Text:=Koneksi.adoqPegawaiJenisKelamin.value;

UbahPegawai.eTempat.Text:=Koneksi.adoqPegawaiTempatLahir.value;

UbahPegawai.dtTanggal.Date:=Koneksi.adoqPegawaiTanggalLahir.value;

UbahPegawai.cbJabatan.Text:=Koneksi.adoqPegawaiJabatan.value;

UbahPegawai.Show;

DataPegawai.Enabled:=False;

end;

end.

*Form* Data Surat

var

DataSurat: TDataSurat;

Stream:TMemoryStream;

Jpeg:TJpegImage;

implementation

{$R \*.dfm}

Uses FrmKoneksi, FrmMenutama, FrmLihatSuratTugas, FrmLihatSPPD1, FrmLihatSPPD2;

procedure TDataSurat.Btn\_BatalClick(Sender: TObject);

begin

eNoSurat.Text:='';

eDasar.Text:=''; edasar.Enabled:=False;

ePerihal.Text:=''; ePerihal.Enabled:=False;

dblNama.Enabled:=False;

eKegiatan.Text:=''; eKegiatan.Enabled:=False;

dtTanggal1.Enabled:=False;

dtTanggal2.Enabled:=False;

eWaktu.Text:=''; eWaktu.Enabled:=False;

eTempat.Text:=''; eTempat.Enabled:=False;

dbeNIP.Visible:=False;

dbePangkat.Visible:=False;

dbeGolongan.Visible:=False;

dbeJabatan.Visible:=False;

dbeUnitKerja.Visible:=False;

cbHari.Text:=''; cbHari.Enabled:=False;

dtTanggal1.Enabled:=False;

dtTanggal2.Enabled:=False;

cbAngkutan.Enabled:=False;

Btn\_Simpan.Enabled:=False;

Btn\_Simpan.Visible:=True;

Btn\_Batal.Enabled:=False;

Btn\_Tambah.Enabled:=True;

Btn\_Hapus.Enabled:=True;

Btn\_Keluar.Enabled:=True;

end;

procedure TDataSurat.Btn\_HapusClick(Sender: TObject);

begin

if (application.messagebox('Apakah Anda Yakin Akan Menghapus Surat Terpilih ?','Peringatan',mb\_YesNo)=idYes) Then

Begin

Koneksi.AdoqSurat.Delete;

Koneksi.AdoqSurat.Close;

Koneksi.AdoqSurat.SQL.Clear;

Koneksi.AdoqSurat.SQL.Add('Select \* From Tbl\_Surat');

Koneksi.AdoqSurat.Open;

End

Else

Abort;

end;

procedure TDataSurat.Btn\_KeluarClick(Sender: TObject);

begin

DataSurat.Close;

MenuUtama.Show;

MenuUtama.Enabled:=true;

end;

procedure TDataSurat.Btn\_SimpanClick(Sender: TObject);

begin

Koneksi.AdoqSurat.open;

Koneksi.AdoqSurat.Append;

Koneksi.AdoqSurat.FieldByName('NoSurat').AsString:=eNoSurat.Text;

if edasar.Text='' then edasar.Text:='-';

Koneksi.AdoqSurat.FieldByName('Dasar').AsString:=eDasar.Text;

if eperihal.Text='' then eperihal.Text:='-';

Koneksi.AdoqSurat.FieldByName('Perihal').AsString:=ePerihal.Text;

Koneksi.AdoqSurat.FieldByName('Nama').AsString:=dblNama.Text;

Koneksi.AdoqSurat.FieldByName('NIP/NUPTK/ID').AsString:=dbeNIP.Text;

Koneksi.AdoqSurat.FieldByName('Pangkat').AsString:=dbePangkat.Text;

Koneksi.AdoqSurat.FieldByName('Golongan').AsString:=dbeGolongan.Text;

Koneksi.AdoqSurat.FieldByName('Jabatan').AsString:=dbeJabatan.Text;

Koneksi.AdoqSurat.FieldByName('UnitKerja').AsString:=dbeUnitKerja.Text;

Koneksi.AdoqSurat.FieldByName('Kegiatan').AsString:=eKegiatan.Text;

Koneksi.AdoqSurat.FieldByName('Hari').AsString:=cbHari.Text;

Koneksi.AdoqSurat.FieldByName('HariBerangkat').AsString:=eHari1.Text;

Koneksi.AdoqSurat.FieldByName('TanggalBerangkat').AsDateTime:=

dtTanggal1.Date;

Koneksi.AdoqSurat.FieldByName('HariPulang').AsString:=eHari2.Text;

Koneksi.AdoqSurat.FieldByName('TanggalPulang').AsDateTime:=

dtTanggal2.Date;

Koneksi.AdoqSurat.FieldByName('Waktu').AsString:=eWaktu.Text;

Koneksi.AdoqSurat.FieldByName('Tempat').AsString:=eTempat.Text;

Koneksi.AdoqSurat.FieldByName('Angkutan').AsString:=cbAngkutan.Text;

Koneksi.AdoqSurat.Post;

Showmessage('Data Berhasil Di Simpan');

Begin

Koneksi.AdoqSurat.Close;

Koneksi.AdoqSurat.SQL.Clear;

Koneksi.AdoqSurat.SQL.Add('Select \* From Tbl\_Surat');

Koneksi.AdoqSurat.Open;

eNoSurat.Text:='';

eDasar.Text:=''; eDasar.Enabled:=False;

ePerihal.Text:=''; ePerihal.Enabled:=False;

dblNama.Enabled:=False;

eKegiatan.Text:=''; eKegiatan.Enabled:=False;

cbHari.Enabled:=False;

dtTanggal1.Enabled:=False;

dtTanggal2.Enabled:=False;

eWaktu.Text:=''; eWaktu.Enabled:=False;

eTempat.Text:=''; eTempat.Enabled:=False;

dbeNIP.Visible:=False;

dbePangkat.Visible:=False;

dbeGolongan.Visible:=False;

dbeJabatan.Visible:=False;

dbeJabatan.Visible:=False;

dbeUnitKerja.Visible:=False;

Btn\_Simpan.Enabled:=False;

Btn\_Batal.Enabled:=False;

Btn\_Tambah.Enabled:=True;

Btn\_Hapus.Enabled:=True;

Btn\_Keluar.Enabled:=True;

End

end;

procedure TDataSurat.Btn\_TambahClick(Sender: TObject);

begin

eNoSurat.Text:='421.2/'+IntToStr(Koneksi.AdoqSurat.recordcount+1)+'-SD/'+FormatDateTime('yyyy',Now);

dblNama.Enabled:=True;

eDasar.Enabled:=True;

ePerihal.Enabled:=True;

eKegiatan.Enabled:=True;

cbHari.Enabled:=True;

dtTanggal1.Enabled:=True;

dtTanggal2.Enabled:=True;

eWaktu.Enabled:=True;

eTempat.Enabled:=True;

cbAngkutan.Enabled:=True;

Btn\_Simpan.Enabled:=True;

Btn\_Batal.Enabled:=True;

Btn\_Tambah.Enabled:=False;

Btn\_Hapus.Enabled:=False;

dblNama.SetFocus;

Btn\_Keluar.Enabled:=False;

end;

procedure TDataSurat.dblNamaClick(Sender: TObject);

begin

dbeNIP.Visible:=True;

dbePangkat.Visible:=True;

dbeGolongan.Visible:=True;

dbeJabatan.Visible:=True;

dbeUnitKerja.Visible:=True;

end;

procedure TDataSurat.dtTanggal1Change(Sender: TObject);

begin

eHari1.Text:=FormatDateTime('dddd',dtTanggal1.Date);

end;

procedure TDataSurat.dtTanggal2Change(Sender: TObject);

begin

eHari2.Text:=FormatDateTime('dddd',dtTanggal2.Date);

end;

end.

*Form* Cetak Surat

var

CetakSurat: TCetakSurat;

Stream:TMemoryStream;

Jpeg:TJpegImage;

implementation

{$R \*.dfm}

uses FrmKoneksi, FrmLihatSuratTugas, FrmMenutama, FrmLihatSPPD1, FrmLihatSPPD2;

procedure TCetakSurat.btnKeluarClick(Sender: TObject);

begin

CetakSurat.Close;

MenuUtama.Show;

MenuUtama.Enabled:=true;

end;

procedure TCetakSurat.btnSPPD1Click(Sender: TObject);

begin

Jpeg:=TJpegImage.Create;

Stream:=TMemoryStream.Create;

TblobField(Koneksi.AdoqSekolah.FieldByName('Kop')).SaveToStream

(Stream);

Stream.Position:=0;

Jpeg.LoadFromStream(Stream);

LihatSPPD1.qriKOP.Picture.Graphic:=Jpeg;

LihatSPPD1.qrlNoSurat.Caption:=Koneksi.adoqSuratNoSurat.value;

LihatSPPD1.qrlPengguna.Caption:='Kepala

'+Koneksi.adoqSekolahNamaSekolah.value;

LihatSPPD1.qrlNama.Caption:=Koneksi.adoqSuratNama.value+' /

+Koneksi.adoqSuratNIPNUPTKID.value;

LihatSPPD1.qrlJabatan.Caption:=Koneksi.adoqSuratJabatan.value;

LihatSPPD1.qrlPangkat.Caption:=Koneksi.adoqSuratPangkat.value+','+

Koneksi.adoqSuratGolongan.value;

LihatSPPD1.qrlAngkutan.Caption:=Koneksi.adoqSuratAngkutan.value;

LihatSPPD1.qrlKegiatan.Caption:=Koneksi.adoqSuratKegiatan.value;

LihatSPPD1.qrlAwal.Caption:=Koneksi.adoqSekolahNamaSekolah.value;

LihatSPPD1.qrlAkhir.Caption:=Koneksi.adoqSuratTempat.value;

LihatSPPD1.qrlHari.Caption:=Koneksi.adoqSuratHari.value+' Hari';

LihatSPPD1.qrlKegiatan.Caption:=Koneksi.adoqSuratKegiatan.value;

LihatSPPD1.qrlTanggalBerangkat.Caption:=FormatDateTime('dd MMMM

yyyy',Koneksi.adoqSuratTanggalBerangkat.Value);

LihatSPPD1.qrlTanggalPulang.Caption:=FormatDateTime('dd MMMM

yyyy',Koneksi.adoqSuratTanggalPulang.Value);

LihatSPPD1.qrlKeluar.Caption:=Koneksi.adoqSekolahKecamatan.value;

LihatSPPD1.qrlTanggal.Caption:=FormatDateTime('dd MMMM

yyyy',Koneksi.adoqSuratTanggalBerangkat.Value);

LihatSPPD1.qrlSekolah.Caption:='Kepala Sekolah

'+Koneksi.adoqSekolahNamaSekolah.value;

LihatSPPD1.qrlKecamatan.Caption:='Kecamatan

'+Koneksi.adoqSekolahKecamatan.value;

LihatSPPD1.qrlKepsek.Caption:=Koneksi.adoqSekolahKepalaSekolah.value;

LihatSPPD1.qrlNIP.Caption:='NIP '+Koneksi.adoqSekolahNIP.value;

LihatSPPD1.qrLihatSPPD1.Preview;

end;

procedure TCetakSurat.btnSPPD2Click(Sender: TObject);

begin

LihatSPPD2.qrlBerangkat.Caption:=Koneksi.adoqSekolahNamaSekolah.value;

LihatSPPD2.qrlke.Caption:=Koneksi.adoqSuratTempat.value;

LihatSPPD2.qrlPadaTanggal.Caption:=FormatDateTime('dd MMMM

yyyy',Koneksi.adoqSuratTanggalBerangkat.value);

LihatSPPD2.qrlKepsek.Caption:=Koneksi.adoqSekolahKepalaSekolah.value;

LihatSPPD2.qrlNIP.Caption:='NIP. '+Koneksi.adoqSekolahNIP.value;

LihatSPPD2.qrlKepsek2.Caption:=Koneksi.adoqSekolahKepalaSekolah.value;

LihatSPPD2.qrlNIP2.Caption:='NIP. '+Koneksi.adoqSekolahNIP.value;

LihatSPPD2.qrlKepsek3.Caption:=Koneksi.adoqSekolahKepalaSekolah.value;

LihatSPPD2.qrlNIP3.Caption:='NIP. '+Koneksi.adoqSekolahNIP.value;

LihatSPPD2.qrLihatSPPD2.Preview;

end;

procedure TCetakSurat.Btn\_CetakClick(Sender: TObject);

begin

Jpeg:=TJpegImage.Create;

Stream:=TMemoryStream.Create;

TblobField(Koneksi.AdoqSekolah.FieldByName('Kop')).SaveToStream

(Stream);

Stream.Position:=0;

Jpeg.LoadFromStream(Stream);

LihatSuratTugas.qriKop.Picture.Graphic:=Jpeg;

LihatSuratTugas.qrlNoSurat.Caption:='Nomor

'+Koneksi.adoqSuratNoSurat.value;

LihatSuratTugas.qrlDasar.Caption:=Koneksi.adoqSuratDasar.value+' Perihal '

+Koneksi.adoqSuratPerihal.value;

LihatSuratTugas.qrlKalimat1.Caption:='Kepala Sekolah

'+Koneksi.AdoqSekolahNamaSekolah.value+' Kecamatan

'+Koneksi.AdoqSekolahKecamatan.value+' : ';

LihatSuratTugas.qrlNama.Caption:=Koneksi.adoqSuratNama.value;

LihatSuratTugas.qrlNIP.Caption:=Koneksi.adoqSuratNIPNUPTKID.value;

LihatSuratTugas.qrlPangkat.Caption:=Koneksi.adoqSuratPangkat.value+','+

Koneksi.adoqSuratGolongan.value;

LihatSuratTugas.qrlJabatan.Caption:=Koneksi.adoqSuratJabatan.value;

LihatSuratTugas.qrlUnit.Caption:=Koneksi.adoqSuratUnitKerja.value;

LihatSuratTugas.qrlKec.Caption:='Koorwil Bidang Pendidikan Kecamatan

'+Koneksi.adoqSekolahKecamatan.value;

LihatSuratTugas.qrlHari.Caption:=Koneksi.adoqSuratHariBerangkat.value+',

'+FormatDateTime('dd MMMM

yyyy',Koneksi.adoqSuratTanggalBerangkat.Value);

LihatSuratTugas.qrlWaktu.Caption:=Koneksi.adoqSuratWaktu.value;

LihatSuratTugas.qrlTempat.Caption:=Koneksi.adoqSuratTempat.value;

LihatSuratTugas.qrlKalimat2.Caption:='Untuk Mengikuti Kegiatan

'+Koneksi.AdoqSuratKegiatan.value+' Pada : ';

LihatSuratTugas.qrlKalimat3.Caption:='Demikian Surat Tugas ini kami buat

untuk diketahui dan dipergunakan sebagaimana mestinya. ';

LihatSuratTugas.qrlKepsek.Caption:='Kepala Sekolah

'+Koneksi.adoqSekolahNamaSekolah.value;

LihatSuratTugas.qrlKeluar.Caption:=Koneksi.adoqSekolahKecamatan.value;

LihatSuratTugas.qrlTglSurat.Caption:=FormatDateTime('dd MMMM

yyyy',Koneksi.adoqSuratTanggalBerangkat.Value);

LihatSuratTugas.qrlKepsek2.Caption:=

Koneksi.adoqSekolahKepalaSekolah.value;

LihatSuratTugas.qrlKecamatan.Caption:='Kecamatan

'+Koneksi.adoqSekolahKecamatan.value;

LihatSuratTugas.qrlNIPKS.Caption:='NIP. '+Koneksi.adoqSekolahNIP.value;

LihatSuratTugas.qrSuratTugas.Preview;

end;

end.

*Form* Cetak Laporan

var

CetakLaporan: TCetakLaporan;

implementation

{$R \*.dfm}

uses FrmCetakDataPegawai, FrmPilihCetakSurat, FrmKoneksi, FrmMenutama;

procedure TCetakLaporan.btnPilihClick(Sender: TObject);

begin

CetakLaporan.Close;

MenuUtama.Show;

MenuUtama.Enabled:=true;

end;

procedure TCetakLaporan.cbPilihChange(Sender: TObject);

begin

if cbPilih.ItemIndex<0 then Showmessage('Silahkan Pilih Jenis Cetak')

else

If cbPilih.ItemIndex=0 then

Begin

CetakPegawai.qrCetakPegawai.Preview;

End

else

PilihCetakSurat.Show;

end;

end.

*Form* Pilih Cetak Surat

var

PilihCetakSurat: TPilihCetakSurat;

implementation

{$R \*.dfm}

uses FrmKoneksi, FrmCetakBerdasarkanPegawai, FrmPilihCetak;

procedure TPilihCetakSurat.btnCetakClick(Sender: TObject);

begin

Koneksi.adoqSurat.SQL.Clear;

Koneksi.adoqSurat.SQL.Add('Select \* From Tbl\_Surat Where

Tbl\_Surat.Nama =:NamaTerpilih');

Koneksi.AdoqSurat.Parameters.ParamValues['NamaTerpilih']:=

dblPegawai.Text;

Koneksi.AdoqSurat.ExecSQL;

Koneksi.AdoqSurat.Prepared;

Koneksi.AdoqSurat.Open;

If Koneksi.AdoqSurat.RecordCount > 0 Then

Begin

cetBerPegawai.qrlberdasarkan.Caption:='Berdasarkan Nama Pegawai';

cetBerPegawai.qrBerPegawai.Preview;

Koneksi.AdoqSurat.Close;

Koneksi.AdoqSurat.SQL.Clear;

Koneksi.AdoqSurat.SQL.Add('Select \* From Tbl\_Surat');

Koneksi.AdoqSurat.Open;

End

Else

Application.MessageBox('Tidak Ada Riwayat Perjalanan Dinas Pada

Pegawai Terpilih Terpilih','Info', mb\_Ok + Mb\_ICONINFORMATION);

end;

procedure TPilihCetakSurat.btnPilihClick(Sender: TObject);

begin

CetakLaporan.show;

PilihCetakSurat.Close;

end;

procedure TPilihCetakSurat.Button2Click(Sender: TObject);

begin

Koneksi.AdoqSurat.SQL.Clear;

Koneksi.AdoqSurat.SQL.Add('Select \* From Tbl\_Surat where

Tbl\_Surat.TanggalBerangkat >=:Awal and

Tbl\_Surat.TanggalBerangkat<=:Akhir');

Koneksi.AdoqSurat.Parameters.ParamValues['Awal']:=datetostr(dtDari.Date);

Koneksi.AdoqSurat.Parameters.ParamValues['Akhir']:=

datetostr(dtSampai.Date);

Koneksi.AdoqSurat.ExecSQL;

Koneksi.AdoqSurat.Prepared;

Koneksi.AdoqSurat.Open;

If Koneksi.AdoqSurat.RecordCount > 0 Then

Begin

cetBerPegawai.qrlBerdasarkan.Caption:='Berdasarkan Periode Tanggal';

cetBerPegawai.qrBerPegawai.Preview;

Koneksi.AdoqSurat.Close;

Koneksi.AdoqSurat.SQL.Clear;

Koneksi.AdoqSurat.SQL.Add('Select \* From Tbl\_Surat');

Koneksi.AdoqSurat.Open;

End

Else

Application.MessageBox('Tidak Ada Riwayat Perjalanan Dinas Pada

Periode Tanggal Terpilih','Info', mb\_Ok + Mb\_ICONINFORMATION);

end;

procedure TPilihCetakSurat.Button3Click(Sender: TObject);

begin

cetBerPegawai.qrlberdasarkan.Caption:=' ';

cetBerPegawai.qrBerPegawai.Preview;

end;

procedure TPilihCetakSurat.rbPilihClick(Sender: TObject);

begin

If rbPilih.ItemIndex=0 Then

Begin

gbPegawai.Visible:=True;

gbPegawai.Enabled:=True;

gbTanggal.Visible:=False;

gbTanggal.Enabled:=False;

gbSemua.Visible:=False;

gbSemua.Enabled:=False;

end

Else

If rbPilih.ItemIndex=1 Then

Begin

gbTanggal.Visible:=True;

gbTanggal.Enabled:=True;

gbPegawai.Visible:=False;

gbPegawai.Enabled:=False;

gbSemua.Visible:=False;

gbSemua.Enabled:=False;

end

Else

Begin

gbsemua.Visible:=True;

gbsemua.Enabled:=True;

gbTanggal.Visible:=False;

gbTanggal.Enabled:=False;

gbPegawai.Visible:=False;

gbPegawai.Enabled:=False;

End

end;

end.