RANCANG BANGUN SISTEM INFORMASI Manajemen SURAT PERINTAH TUGAS

(BALAI PEMASYARAKATAN KLAS ii PEKANBARU)

LAPORAN KERJA PRAKTiK

Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Teknik

Pada Jurusan Teknik Informatika

Oleh:

KURNIADO

11651103676



FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM NEGERI SULTAN SYARIF KASIM RIAU

PEKANBARU

2019

# LEMBAR PENGESAHAN BALAI PEMASYARAKATAN PEKANBARU

RANCANG BANGUN SISTEM INFORMASI manajemen SURAT PERINTAH TUGAS

( BALAI PEMASYARAKATAN KLAS ii PEKANBARU )

LAPORAN KERJA PRAKTEK

Jurusan Teknik Informatika

Fakultas Sains dan Teknologi

Universitas Islam Negeri Sultan Syarif Kasim Riau

Oleh:

KURNIADO

11651103676

Telah diperiksa dan disetujui sebagai laporan Kerja Praktik

di Pekanbaru, pada tanggal 8 Oktober 2019

Pembimbing,

NURSAL

NIP.

# LEMBAR PENGESAHAN JURUSAN

RANCANG BANGUN SISTEM INFORMASI manajemen SURAT PERINTAH TUGAS

(balai pemasyarakatan klas II PEKANBARU)

LAPORAN KERJA PRAKTEK

Oleh:

kurniado

11651103676

Telah disetujui dan disahkan sebagai laporan Kerja Praktik

di Pekanbaru, pada tanggal 8 Oktober 2019

Pembimbing Kerja Praktik,

dila

NIk. 130 517 107

Mengetahui,

Ketua Jurusan Teknik Informatika

Fakultas Sains dan Teknologi

Universitas Islam Negeri Sultan Syarif Kasim Riau

Pekanbaru, 8 Oktober 2019

elin haerani, ST, MT

NIP. 19810523 200710 2 003

# ABSTRAK

Sistem Informasi manajemen surat perintah tugas merupakan sistem yang dibangun untuk mengelola proses surat perintah tugas dimulai dari input data surat masuk, pembuatan disposisi, persetujua disposisi, pembuatan surat perintah tugas, penginputan data pegawai, persetujuan surat perintah tugas dan pengontrolan surat perintah tugas. Penelitian ini dilakukan pada Balai Pemasyarakatan Klas II Pekabaru Provinsi Riau dengan menggunakan metode penelitian yaitu metode identifikasi masalah, pengumpulan data, imlementasi dan pengujian. Program yang telah dibangun berjudul Sistem Informasi Manajemen Surat Peintah Tugas Berbasis Website dengan desain sistem mengunakan Usecase Diagram, Usecase Sequence, Class Diagram dan Entity Relationship Diagram (ERD).

Sistem yang dibangun berbasis website. Dari hasil implementasi sistem, dapat disimpulkan bahwa dengan penggunaan sistem dapat membantu pegawai dalam mengelola surat perintah tugas pada balai pemasyarakatan klas ii pekanbaru provinsi riau

Kata Kunci : Sistem Informasi, Surat Perintah Tugas, Website

# KATA PENGANTAR

***Bismilahirahmanirahim,***

*Assalamu’alaikum Warahmatullahi Wabarakatuh*

Puji dan syukur ke hadirat Allah SWT, karena hanya atas berkah dan karunia-Nya penulis dapat menyelesaikan kerja praktek beserta laporan yang berjudul *“***RANCANG BANGUN SISTEM INFORMASI PENGELOLAAN SURAT PERINTAH TUGAS BERBASIS WEB***”*. Laporan ini disusun guna memenuhi Kerja Praktek dan syarat untuk lulus dari program *study* Teknik Informatika.

Dalam penyusunan laporan, penyusun memperoleh banyak bantuan dari berbagai pihak, untuk itu penyusun mengucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang terlibat yang telah bersedia meluangkan waktunya untuk membantu penulis sehingga penulis dapat menyelesaikan laporan ini tepat pada waktunya. Untuk itu penulis ingin menyampaikan ucapan terima kasih kepada:

1. Terima kasih kepada Ayah, Ibu dan seluruh anggota keluarga yang tiada hentinya memanjatkan doa, memberikan dukungan dan semangat untuk kesuksesan penulis.
2. Bapak Prof. Dr. H. Akhmad Mujahidin, M.Ag, selaku Rektor Universitas Islam Negeri Sultan Syarif Kasim Riau
3. Bapak Dr. Drs. H. Mas’ud Zein, M.Pd, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Sultan Syarif Kasim Riau.
4. Ibu Elin Haerani, ST, MT. Selaku Ketua Jurusan Teknik Informatika Fakultas Sains dan Teknologi Uin Suska Riau.
5. Ibu Sonya Meitarice, S.T, M.S.Eng selaku Koordinator Kerja praktek Jurusan Teknik Informatika Fakultas Sains dan Teknologi UIN SUSKA RIAU.
6. Ibu Dila selaku pembimbing kerja praktek jurusan Teknik Informatika.
7. Terima kasih kepada Nurul Astri Melati selaku teman seperjuangan kerja praktek.
8. Terimakasih kepada teman-teman seperjuangan TIF D angkatan 2016 dan Terkhusus Kepada Imam Khoiri Jelfi, Muhammad ardiyansyah, Muhammad Fauzan Wijanarko Selaku Teman yang sangat berperan dalam membantu kerja praktek ini dan terimakasih.
9. Terima kasih kepada pihak-pihak yang membantu secara langsung maupun tidak langsung dalam pelaksanaan kerja praktek dan pembuatan sistem serta laporan.

Semoga laporan ini dapat bermanfaat bagi penulis khususnya maupun pembaca umumnya. Penulis berharap mendapatkan masukan dari pembaca atas isi laporan ini. Kritik dan saran tersebut dapat pembaca sampaikan ke alamat email penulis: [kurniado729@gmail.com](mailto:kurniado729@gmail.com) Akhir kata penulis mengucapkan terima kasih.

Pekanbaru, 24 November 2019

Penulis

# DAFTAR ISI

# DAFTAR GAMBAR

**Gambar Halaman**

# DAFTAR TABEL

# DAFTAR SIMBOL

Tabel Notasi *Flow Chart*

|  |  |  |
| --- | --- | --- |
| **SIMBOL** | **NAMA** | **KETERANGAN** |
|  | *Terminator* | Menyatakan permulaan (awal) atau akhir dari suatu program. |
|  | *Process* | Menyatakan suatu tindakan (proses) yang dilakukan oleh komputer. |
|  | *Decision* | Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban: Ya (*Yes*) / Tidak (*No*). |
|  | *Flow Line* | Menunjukkan arus/aliran dari proses. |

Tabel Notasi *Use Case Diagram*

|  |  |  |
| --- | --- | --- |
| **SIMBOL** | **NAMA** | **KETERANGAN** |
|  | *Actor* | Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan *use case.* |
|  | *Include* | Menspesifikasikan bahwa *use case* sumber secara *eksplisit.* |
|  | *Association* | Menghubungkan antara objek satu dengan objek lainnya. |
|  | *System Boundary* | Menjelaskan batasan antara sistem dengan *actor.* |
|  | *Use Case* | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu *actor.* |

Tabel Notasi *Sequence Diagram*

|  |  |  |
| --- | --- | --- |
| **SIMBOL** | **NAMA** | **KETERANGAN** |
|  | *Actor* | Menggambarkan orang yang sedang berinterkasi dengan sistem. |
|  | *Entity Class* | Menggambarkan hubungan kegiatan yang akan dilakukan. |
|  | *Boundary Class* | Menggambarkan sebuah *interface* atau *form.* |
|  | *Control Class* | Menggambarkan penghubung antara *boundary* dengan tabel. |
|  | *Life Line* | Menggambarkan tempat mulai dan berakhirnya sebuah *Message.* |
| A Message() | *A Message* | Mengambarkan pengiriman pesan. |

Tabel Notasi *Class Diagram*

|  |  |  |
| --- | --- | --- |
| **SIMBOL** | **NAMA** | **KETERANGAN** |
| operations  attributes  name  Shape  origin  move()  resize()  display() | Operasi | *Class* adalah blok-blok pembangun pada pemrograman berorientasi objek. Sebuah *class* digambarkan sebagai sebuah kotak yang terbagi atas 3 bagian. Bagian atas adalah bagian nama dari *class*. Bagian tengah mendefinisikan property / atribut *class*. Bagian akhir mendefinisikan *method-method* dari sebuah *class.* |
|  | Generalisasi | Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus). |

Tabel Notasi *Deployment Diagram*

|  |  |  |
| --- | --- | --- |
| **SIMBOL** | **NAMA** | **KETERANGAN** |
|  | *Association* | Sebuah *association* digambarkan sebagai sebuah garis yang menghubungkan dua node yang mengindikasikan jalur komunikasi antara elemen-elemen *hardware.* |
| Node Name | *Node* | *Node* menggambarkan bagian-bagian *hardware* dalam sebuah sistem. Notasi untuk *node* digambarkan sebagai sebuah kubus 3 dimensi. |

# PENDAHULUAN

## Latar Belakang

Penggunaan komputer sebagai salah satu alat teknologi informasi sangat dibutuhkan keberadaannya hampir disetiap aspek kehidupan seperti lingkungan pendidikan, lingkungan sosial, lingkungan ekonomi dan termasuk dilingkungan instansi pemerintahan.

Penggunaan perangkat komputer sebagai perangkat pendukung manajemen dan pengolahan data sangat tepat dengan mempertimbangkan kuantitas dan kualitas data, dengan demikian penggunaan perangkat komputer dalam setiap informasi sangat mendukung dalam pengambilan keputusan.

Badan Pemasyarakatan (BAPAS) Pekanbaru Provinsi Riau merupakan instansi pemerintah yang bertugas melaksananakan bimbingan klien pemasyarakatam di daerah Riau, bimbingan yang dilakukan oleh balai pemasyarakatan dilakukan terhadap terpidana bersyarat, Narapidana, Anak Pidana, dan Anak Negara yang mendapat pembebasan bersyarat atau cuti menjelang bebas. dalam aktifitasnya perintah tugas menjadi hal utama dalam melaksanakan tugas pokok dan fungsi instansi ini, sehingga membuat surat perintah tugas menjadi hal utama sekaligus kendala karena pengelolaan data perintah tugas, masih dilakukan dengan manual, menggunakan buku dan kertas-kertas, sehingga pengelolaan surat perintah tugas masih belum akurat dan relatif lama dalam penyelesaiannya.

Data-data yang berhubungan dengan surat perintah tugas di Balai Pemasyarakatan (BAPAS) yang terhimpun itu, tentu saja sangat diperlukan untuk memenuhi kebutuhan dalam pencarian informasi dan laporan bagi pegawai. Sehingga dibutuhkan suatu sistem informasi yang dapat diandalkan untuk mengolah data menjadi informasi yang bermanfaat dalam mengambil keputusan manajemen baik rutin maupun strategis. Keadaan ini mendorong suatu usaha untuk merancang suatu sistem informasi manajemen surat perintah tugas, agar masalah pengolahan data surat dapat berjalan dengan baik.

## Rumusan Masalah

Berdasarkan latar belakang masalah yang telah dikemukakan, maka dapat diambil suatu rumusan masalah yaitu bagaimana membangun sistem informasi yang dapat membantu pegawai balai pemasyarakatan dalam memanajemen surat perintah tugas .

## Batasan Masalah

Pelaksanaan dalam perancangan Sistem Informasi surat perintah tugas dibatasi pada hal-hal berikut:

1. Sistem ini berbasis *Web System*
2. Menjadikan balai pemasyarakatan Pekanbaru sebagai tempat Implementasi
3. Sistem hanya berfokus ke mengelola surat perintah tugas

## Tujuan

Tujuan dari kerja praktek ini adalah untuk membangun Sistem Informasi pengelola surat perintah tugas untuk balai pemasyarakatan Pekanbaru

## Waktu dan Tempat Kerja Praktek

Kerja praktek ini dilaksanakan pada:

Waktu : 8 Oktober 2019 s/d 8 November 2019

Tempat : Balai Pemasyarakatan Klas II Pekanbaru.

## Sistematika Penulisan

Untuk memudahkan memahami hasil dan kerja praktek ini, maka akan dikemukakan sistematika penulisannya.

1. **PENDAHULUAN**

Bab ini menjelaskan dasar-dasar dari penulisan laporan kerja praktek, yang terdiri dari Menjelaskan mengenai latar belakang masalah, rumusan masalah, batasan masalah, tujuan penulisan, waktu dan tempat kerja praktek, serta sistematika penulisan.

1. **PROFIL** **INSTANSI**

Bab ini menjelaskan uraian secara teoritis tentang hal-hal khusus mengenai sejarah umum sekolah, visi dan misi, dan struktur organisasi.

1. **TUGAS KERJA PRAKTIK**

Bab ini berisi uraian tugas kerja praktik, metode pengerjaan kerja praktik, dan jadwal pengerjaan.

1. **LANDASAN TEORI**

Bab ini membahas teori-teori yang berhubungan dengan topik penelitian*.* Dimulai dari *use* *case* *Diagram*, *sequence* *Diagram*, *class* *Diagram*, *development* *Diagram*.

1. **ANALISA DAN PERANCANGAN**

Menjelaskan tentang bagaimana pembuatan aplikasi *web* layanan yang akan dibuat. Analisa berupa Analisa dari sistem yang sedang diguanakan, Analisa sistem baru, Analisa kebutuhan sistem, Analisa fungsional sistem, *use case Diagram, sequence Diagram, class Diagram.* Dan perancangan berupa perancangan *database,* danperancangan antar muka.

1. **IMPLEMENTASI DAN PENGUJIAN**

Menjelaskan gambaran sistem dan alur pengisian data pada sistem dimulai dari implementasi sistem, impelentasi Sistem Informasi Penilaian dan Rapor Siswa Kurikulum 2013, dan pengujian sistem.

1. **PENUTUP**

Bab ini berisi kesimpulan dari pembahasan tentang Sistem Informasi Penilaian dan Rapor Siswa Kurikulum 2013 serta beberapa saran sebagai hasil akhir dari penelitian yang telah dilakukan.

# PROFIL INSTANSI



## Sejarah Umum Instansi

Berdasarkan Surat Keputusan Direktorat Jendral Bina Tuna warga (BTW) dan mendahului Surat Keputusan Menteri Kehakiman R.I Nomor : DDP.4.I/10/43 tanggal 14 Juni 1974 dan berdasarkan Surat Keputusan Menteri Hukum dan HAM R.I Nomor : j.s.416/3 Tahun 1997 tanggal 30 Juli 1977 berdiri Balai Bispa Pekanbaru yang berstatus Klas II A berkantor di bekas Lembaga Pemasyarakatan di Jl. Indrayani Pekanbaru dengan jumlah pegawai sebanyak 3 (tiga) orang.

Pada tahun 1979 dengan DIP T.A. 1979/1980 dibangunlah gedung Balai Bispa di Jl. Kesuma No. 2 Pekanbaru seluas 180 m2. Berdasarkan Keputusan Menteri Kehakiman R.I Nomor : M.01-PR,07,03 Tahun 1997 tanggal 12 Pebruari 1997 Balai Bispa berubah namanya menjadi BAPAS ( Balai Pemasyarakatan ). Karena perkembangan kota dan dianggap gedung Balai Bispa tidak cocok berada ditengah-tengah pasar (saat ini gedung Citra Plaza Pekanbaru) maka didirikanlah gedung Bapas tepatnya tanggal 30 Juli 1999. Bapas menempati gedung baru di Jl. H.R. Soebrantas KM. 10.5 Panam Pekanbaru yang dibangun di atas lahan seluas 4,000 m2 dengan luas bangunan seluas 353 m2 dan diresmikan oleh Dirjen PAS Thaer Abdullah, Bc.IP

## Visi dan Misi

Visi dan misi Balai Pemasyarakatan Klas II Pekanbaru adalah Sebagai Berikut :

### Visi

Kemandirian Klien Pemasyarakatan Kebanggaan Kami

### Misi

Melaksanakan Pembimbingan Klien Pemasyarakatan Dalam Rangka Penegakan Hukum, Kemandirian, Dan Pelayanan Pemasyarakatan

## Struktur Organisasi

Struktur Organisasi Balai Pemasyarakatan Klas II Pekanbaru dapat dilihat pada gambar berikut :

*Gambar 2. 1 Struktur Organisasi Yayasan Pendidikan Tigama*

# TUGAS KERJA PRAKTEK



## Uraian Tugas Kerja Praktek

Kerja praktek dilaksanakan di Balai Pemasyarakatan Klas II Pekanbaru. Kerja praktek dimulai pada tanggal 8 Oktober 2019 dan berakhir pada tanggal 10 September 2019. Adapun berbagai kegiatan dan pekerjaan dilakukan ditujukan untuk membantu pihak bapas, dan juga untuk mendapat pengalaman yang nantinya menjadi gambaran akan suasana dunia kerja sebenarnya. Berikut adalah tugas dan kegiatan yang dilakukan selama melaksanakan kerja praktek :

Tabel 3. 1 Tugas dan Kegiatan Selama Melaksanakan Kerja Praktek

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Kegiatan | Tugas kerja praktek dalam hitungan minggu pada  1 Oktober 2019 – 18 September 2019 | | | |
| 1 | 2 | 3 | 4 |
| 1 | Melakukan pengelolaan tentang surat masuk dan surat keluar, baik menginputkan data surat masuk, data surat keluar, membuat disposisi surat dan lain lain. |  |  |  |  |
| 2 | Melakukan analisa sistem yang akan dibuat |  |  |  |  |
| 3 | Mengelola Email masuk dan Email keluar |  |  |  |  |
| 4 | Melakukan analisa data dan pengumpulan data yang dibutuhkan |  |  |  |  |
| 5 | Melakukan perancangan sistem yang dibuat |  |  |  |  |
| 6 | Melakukan pembangunan sistem terhadap analisa dan perancangan |  |  |  |  |

## 

## Metode Pengerjaan Kerja Praktek

Metode pelaksanaan kerja praktek ini adalah menggunakan metode *waterfall* sebagai berikut:

1. Identifikasi Masalah

Dari latar belakang yang telah dijelaskan, identifikasi masalah yang dijadikan bahan penelitian adalah pengelolaan surat perintah tugas yang dilakukan secara manual oleh pegawai, yang memiliki prosedur yang lama dan data yang tidak terintegrasi.

1. Pengumpulan Data

Untuk mendapatkan data yang benar-benar akurat, relevan, *valid* (sahih) dan  terpercaya maka  penulis  mengumpulkan  sumber  data  dengan cara:

1. Metode Observasi

Pengumpulan data melalui pengamatan dan pencatatan terhadap gejala peristiwa yang diselidiki pada obyek penelitian secara langsung.

1. Wawancara

Wawancara merupakan proses pengumpulan data melalui tatap muka dan tanya jawab langsung dengan sumber data atau pihak-pihak yang berkepentingan yang berhubungan dengan penelitian. Pada metode ini, penulis mewawancarai Bagian Kurikulum dan aktor-aktor yang akan terlibat nantinya di dalam sistem.

1. Analisa dan Perancangan

Pada tahap ini dilakukan analisa terhadap sistem yang akan dibangun berdasarkan hasil wawancara dan observasi terhadap kebutuhan pihak balai pemasyarakatan dalam mengelola surat perintah tugas. Selanjutnya, dilakukan kegiatan perancangan sistem berdasarkan data-data yang telah dikumpulkan pada tahap sebelumnya dengan cara merancang *use case*, *database* dan juga *interface* sistem dan aplikasi. Hasil dari tahap ini merupakan dokumentasi *use case*, *database* dan *interface* sistem dan aplikasi*.*

1. Implementasi dan Pengujian

Implementasi pada tahapan ini adalah bertujuan mengubah hasil dari rancangan dan pengujian kedalam bentuk nyata yang dapat digunakan pada perangkat atau komputer di Balai Pemasyarakatan Klas II Pekanbaru.

1. Dokumentasi

Dokumentasi dilakukan selama proses penelitian, dari proses pengumpulan data, perencanaan, hingga perancangan. Dan hasil akhir (*output*) dari dokumentasi ini berisi laporan mengenai implementasi dan pengujian dari sistem yang telah dibuat.

## Jadwal Pengerjaan

Kerja praktek dilaksanakan selama lebih kurang tiga bulan, dengan rincian pengerjaan seperti tabel dibawah ini:

Tabel 3. 2 Jadwal Pengerjaan Kerja Praktek

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| No | Kegiatan | Pengerjaan dalam hitungan bulan dan minggu | | | | | |
|  |  | Oktober | | | | September | |
|  |  | I | II | III | IV | I | II |
| 1 | Identifikasi Masalah |  |  |  |  |  |  |
| 2 | Pengumpulan data |  |  |  |  |  |  |
| 3 | Implementasi dan Pengujian |  |  |  |  |  |  |
| 4 | Dokumentasi |  |  |  |  |  |  |

# LANDASAN TEORI



## Pengertian Sistem

* + - 1. Pengertian Sistem menurut Mulyadi (2008) yaitu “Sekelompok dua atau lebih komponen-komponen yang saling berkaitan (subsistem-subsistem yang bersatu untuk mencapai tujuan yang sama)”.
      2. Pengertian Sistem menurut Winarno (2006) yaitu “Sekumpulan komponen yang saling bekerja sama untuk mencapai tujuan tertentu.
      3. Pengertian Sistem menurut McLeod yang dikutip oleh Machmud (2013) yaitu “A sistem is a group of elements that are integrated with the common porpose of achieving an objective”. Sistem adalah sekelompok elemen yang terintegritasi dengan maksud yang sama untuk mencapai suatu tujuan.(MACHMUD, 2013)

Berdasarkan beberapa pendapat yang dikemukakan diatas dapat ditarik kesimpulan bahwa “Sistem adalah kumpulan elemen-elemen komponen atau subsistem yang terintegrasi atau saling terkait serta bekerja sama untuk mencapai suatu tujuan tertentu”.

## Pengertian Informasi

Secara *Etimologi*, Kata informasi ini berasal dari kata bahasa *Perancis kuno informacion* (tahun 1387) mengambil istilah dari bahasa Latin yaitu *informationem* yang berarti “konsep, ide atau garis besar,”. Informasi ini merupakan kata benda dari informasi yang berarti aktivitas. Aktivitas dalam “pengetahuan yang dikomunikasikan” informasi bisa menjadi fungsi penting  dalam membantu mengurangi rasa cemas pada seseorang. Menurut pendapat Notoatmodjo bahwa semakin banyak memiliki informasi dapat memengaruhi atau menambah pengetahuan terhadap seseorang dan dengan pengetahuan tersebut bisa menimbulkan kesadaran yang akhirnya seseorang itu akan berperilaku sesuai dengan pengetahuan yang dimilikinya (Kosasih, 2006).

## Konsep Dasar Informasi

Informasi adalah kumpulan data yang diolah sedemikian rupa sehingga menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya. Sumber informasi adalah data. Data kenyataannya yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian (event) adalah kejadian yang terjadi pada saat tertentu.

Menurut Gordon B Davis (2015:8) Informasi adalah data yang telah diolah menjadi suatu bentuk yang penting bagi si penerima dan mempunyai nilai nyata yang dapat dirasakan dalam keputusan – keputusan yang sekarang atau keputusan-keputusan yang akan datang.

Pengertian Informasi menurut Kusrini (2007) yaitu “Informasi adalah data yang sudah diolah menjadi sebuah bentuk yang berguna bagi pengguna yang bermanfaat dalam pengambilan keputusan saat ini atau mendukung sumber informasi”.

Pengertian Informasi menurut Jogiyanto yang dikutip oleh Machmud (2013) yaitu “Informasi diartikan sebagai data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya”. Jadi Informasi adalah data yang diproses kedalam bentuk yang lebih berarti bagi penerima dan berguna dalam pengambilan keputusan, sekarang atau untuk masa yang akan datang.(MACHMUD, 2013)

## Pengertian Sistem Informasi

Sistem informasi yaitu suatu sistem yang menyediakan informasi untuk manajemen dalam mengambil keputusan dan juga untuk menjalankan operasional perusahaan, dimana sistem tersebut merupakan kombinasi teratur dari orang-orang, *hardware*, *software*, jaringan komunikasi dan sumber daya data yang mengumpulkan, mengubah, dan menyebarkan informasi dalam sebuah organisasi (Anggraeni & Irviani, 2017). Fungsi sistem informasi menurut Anggraeni & Irviani adalah sebagai berikut:

1. Untuk meningkatkan aksebilitas data yang ada secara efektif dan efisien kepada pengguna, tanpa dengan perantara sistem informasi.
2. Memperbaiki produktivitas aplikasi pengembangan dan pemeliharaan sistem.
3. Menjamin tersedianya kualitas dan keterampilan dalam memanfaatkan sistem informasi secara kritis.
4. Mengidentifikasi kebutuhan mengenai keterampilan pendukung sistem informasi.
5. Mengantisipasi dan memahami akan konsekuensi ekonomi.
6. Menetapkan investasi yang akan diarahkan pada sistem informasi.
7. Mengembangkan proses perencanaan yang efektif.

## Konsep Dasar Sistem Informasi

Sistem Informasi adalah kumpulan atau susunan yang terdiri dari perangkat keras dan perangkat lunak serta tenaga pelaksanaannya yang bekerja dalam sebuah proses berurutan dan secara bersama-sama saling mendukung untuk menghasilkan suatu produk (Dengen, 2009:48).

Definisi Sistem Informasi menurut Azhar Susanto (2008) adalah sebagai berikut: “Sistem informasi adalah kumpulan dari subsistem apapun baik phisik ataupun non phisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan yaitu mengolah data menjadi informasi yang berarti dan berguna”.

Menurut Laudon yang dikutip oleh Mukti,dkk (2013), an information system can be defined technically as a set of interrelated components that collect (or retrieve), process, store, and distribute information to support decision making and control in an organization. Sistem informasi adalah kumpulan komponen yang saling berhubungan dalam mengumpulkan, memproses, menyimpan, menyediakan dan mendistribusikan informasi untuk mendukung pengambilan keputusan dan pengendalian di dalam organisasi.

Menurut Robert yang dikutip oleh Minarni dan Saputra (2011), Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan - laporan yang diperlukan.

Menurut Leitch Davis yang dikutip oleh Minarni dan Saputra (2011), Sistem informasi adalah “suatu sistem didalam sebuah organisasi yang mempertemukan kebutuhan pengolahan data transaksi harian yang mendukung operasi dan bersifat manajerial dan kegiatan strategis yang diperlukan bagi pihak luar tertentu”. (Minarni dkk, 2011)

## Komponen Sistem Informasi

Komponen-komponen dari sistem informasi (Anggraeni & Irviani, 2017) adalah sebagai berikut:

1. Komponen *input*, adalah data yang masuk ke dalam sistem informasi.
2. Komponen model, adalah kombinasi prosedur, logika, dan model matematika yang memproses data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.
3. Komponen *output*, adalah hasil informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.
4. Komponen teknologi, adalah alat dalam sistem informasi, teknologi digunakan dalam menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan *output*, dan memantau pengendalian sistem.
5. Komponen basis data, adalah kumpulan data yang saling berhubungan yang tersimpan di dalam komputer dengan menggunakan *software database*.
6. Komponen kontrol, adalah komponen yang mengendalikan gangguan terhadap sistem informasi.

## Pengertian Manajemen

Menurut Stephen dan Marry Coulter (2012:37) manajemen merupakan sebuah proses mengkoordinasikan dan melihat aktivitas pekerjaan yang berguna untuk melihat apakah pekerjaan yang dikerjakan tersebut menjadi efektif dan efisien. Efesiensi ialah melakukan suatu pekerjaan dengan hasil baik serta mendapatkan hasil (output) yang setidaknya lebih banyak dari jumlah input yang ada. Sedangkan efektifitas ialah suatu hal dengan benar serta menyelesaikan kegiatan yang membuat organisasi mencapai tujuan yang telah ditetapkan. Fungsi-fungsi

manajemen tersebut dapat di uraikan sebagai berikut:

Perencanaan (Planning). Perencanaan adalah proses menentukan arah yang akan ditempuh dan kegiatan-kegiatan yang diperlukan untuk mencapai tujuan yang telah

diterapkan.

Pengorganisaisan (Organizing). Pengorganisasian adalah proses mengelompokkan kegiatan-kegiatan atau pekerjaan-pekerjaan dalam unit-unit. Tujuannya adalah supaya tertata dengan jelas antara tugas, wewenang dan tanggung jawab serta hubungan kerja dengan sebaik mungkin.

Pelaksanaan (Actuating). Menggerakkan atau melaksanakan adalah proses untuk menjalankan kegiatan atau pekerjaan dalam organisasi. Dalam menjalankan rganisasi

para manajer harus menggerakkan bawahanya (para karyawan).

* + - 1. Pengawasan (Leading). Pengawasan adalah proses untuk mengukur dan menilai pelaksanaan tugas apakah sesuai dengan rencana. Jika dalam proses tersebut terjadi pemyimpangan, maka akan segera dikendalikan.

## Pengertian Manajemen Sistem Informasi

Menurut McLeod dan Schell (2008:40) Sistem Informasi manajemen merupakan suatu sistem berbasis komputer yang menyediakan informasi bagi beberapa pemakai ebutuhan yang serupa. Informasi tersebut tersedia dalam bentuk laporan periodik, laporan khusus, dan output dari simulasi informasi yang ada. Para pemakai biasanya membentuk suatu entitas organisasi formal instansi atau sub-unit di bawahnya. Informasi digunakan oleh pengelola maupun staf lainnya pada saat mereka membuat keputusan untuk memecahkan masalah.

Menurut O’Brien yang diterjemahkan oleh Fitriasari dan Kwary (2010:443) mengemukakan bahwa sistem informasi manajemen adalah jenis awal dari sistem informasi yang dikembangkan untuk mendukung pengambilan keputusan menajerial. Sistem informasi menghasilkan produk informasi yang mendukung banyak kebutuhan pengambilan keputusan harian dari para manajer dan praktisi bisnis. Laporan, tampilan, dan respon yang dihasilkan oleh sistem informasi manajemen menyediakan informasi yang telah ditetapkan oleh para pengambil keputusan untuk mencukupi kebutuhan informasi.

Menurut McLeod dan Schell (2008:41-43) SIM menghasilkan informasi yang dapat dikelompokan menjadi terdapat beberapa karakteristik sistem dalam SIM, seperti : memiliki elemen – elemen (elements), mempunyai suatu batass (boundary), memiliki lingkungan luar (environment), memiliki penghubung (interface), empunyai masukan (input), mempunyai keluaran (output), mempunyai pengolahan suatu proses (process), dan memiliki objek serta tujuan yang jelas ( objectives and goals).

Bedasarkan pemahaman dari pendapat beberapa ahli tersebut, maka Sistem Informasi Manajemen Desa Mandiri Energi adalah suatu sistem berbasis internet yang menyediakan informasi yang mendukung fungsi managemen pejabat berwewenang Kemenko Perekonomian saat melaksanakan tugas mengkoordinasi Kegiatan Desa Mandiri Energi agar dapat terealisasi sesuai dengan Rencana Kerja Tahunan (RKT) Pemerintah Republik Indonesia. Fungsi Managemen kegiatan Desa Mandiri Energi yang dimaksud berupa kegiatan perencanaan saat menetapkan sasaran kegiatan tahun berjalan, mengorganisasi kegiatan melalui rapat koordinasi, menggerakkan anggota pokja untuk merespon suatu masalah melalui disposisi, dan melakukan pengawasan kegiatan melalui mekanisme penerbitan surat tugas observasi lapangan SIM DME mengatur bagaimana SI kegiatan DME di Kemenko Perekonomian dapat dikelola dengan baik dan tepat waktu. Informasi yang dibutuhkan dapat terseleksi sesuai dengan keinginan pengguna sehingga ketika informasi dibutuhkan kembali oleh pengguna dapat diakses secara real-time.

## Pengertian Surat

Surat sebagai alat barometer kemajuan organisasi karena semakin banyak aktivitas atau kegiatan suatu organisasi, maka kegiatan surat menyurat jelas akan banyak. Ada beberapa pendapat yang mengemukakan definisi tentang surat, antara lain:

Suprapto (2006) menjelaskan bahwa surat merupakan alat komunikasi antara dua pihak yang berupa tulisan dalam kertas atau lainnya, seseorang menulis surat tidak lain untuk mengkomunikasikan atau menginformasikan suatu gagasan dan pikirannya kepada pihak lain, baik atas nama pribadi atau yang lainnya.

Dewi (2011) menjelaskan bahwa surat merupakan salah satu alat komunikasi tertulis sehari hari didalam sebuah organisasi. Jenis surat yang diterima maupun yang dikirim (keluar) berbagai macam, sehingga arus surat dengan frekuensi tinggi sering menimbulkan kemacetan dan kekacauan pada arus informasi. Untuk itu dibutuhkan pengelolahan surat dengan teknik, metode dan sistem tertentu sesuai instansi masing-masing.

* + - 1. Poerwadarminta dalam Simorangkir (2005) menjelaskan bahwa surat adalah kertas yang bertuliskan atau surat adalah alat untuk menyampaikan suatu maksud secara tertulis. Panji dalam Simorangkir (2005) menyatakan surat adalah sehelai

kertas atau lebih yang membuat suatu bahan komunikasi yang disampaikan oleh seseorang kepada orang lain, baik atas nama pribadi maupun kedudukannya dalam organisasi atau kantor.

* + - 1. Martono dalam Simorangkir (2005) menyatakan surat hanyalah secarik kertas yang bertuliskan berita namun berperan sebagai sarana komunikasi individu atau kelompok. Jadi, dapat disimpulkan bahwa surat adalah sebagai alat atau media komunikasi tertulis untuk menyampaikan warta atau berita dari satu pihak kepada pihak lain.

## Penggolongan Surat

Surat-surat yang digunakan baik oleh individu ataupun instansi memiliki berbagai macam penggolongan surat, baik menurut isi, tujuan, wujud, sifat, keamanan isi, dan juga urgensi pengiriman/penyelesaiannyan.

Menurut Suryani,dkk (2013:4), Penggolongan surat terdiri dari:

1. Penggolongan surat menurut wujudnya:
   * + - 1. Kartu Pos, yaitu surat yang ditulis pada selembar kertas sejenis karton yang berukuran 15 x 10 cm.
         2. Warkat Pos, yaitu surat yang ditulis pada sehelai kertas yang didesain sedemikian rupa sehingga isi surat tidak dapat dibaca orang lain seperti layaknya surat yang bersampul. Warkatpos dikeluarkan oleh PN POSTEL.
         3. Surat bersampul, yaitu surat yang ditulis pada secarik kertas (biasanya kertas dengan ukuran A4) yang dimasukkan dalam sampul surat/amplop. Adapun ukuran sampul surat ada berukuran kecil (berukuran 10 x 15 cm) dan sampul panjang (berukuran 10 x 24 cm).
         4. Memorandum (Memo) dan Nota, yaitu surat-menyurat yang digunakan dikalangan internal suatu kantor. Memo dan nota merupakan surat yang isinya pokok-pokok masalah yang ditulis secara singkat.
2. Penggolongan surat berdasarkan tujuan atau isinya:

Surat pemberitahuan, yaitu surat yang isinya memberitahukan sesuatu informasi agar diketahui oleh penerima surat.

Surat perintah, yaitu surat yang isinya memerintahkan suatu hal untuk dilaksanakan oleh penerima surat.

Surat permintaan/permohonan, yaitu surat yang isinya suatu permintaan atau permohonan sesuatu dari penulis surat kepada penerima surat. Permintaan atau permohonan itu dapat berupa suatu barang, melaksanakan sesuatu, atau tidak melaksanakan sesuatu.

Surat teguran/peringatan, yaitu surat yang isinya mengingatkan atau menegur atas suatu tindakan yang dilakukan oleh penerima surat.

Surat panggilan, yaitu surat yang isinya tentang suatu harapan atau keinginan atau perintah agar penerima surat melaksanakan apa yang diharapkan, diinginkan atau diperintahkan dari penulis surat kepada penerima surat.

Surat pengantar, yaitu surat yang isinya menjelaskan atau tentang sesuatu yang harus dilakukan atau diteruskan yang oleh penerima surat atas surat/lembar petunjuk yang dikirimkan bersamanya.

Surat keputusan, yaitu surat yang isinya memutuskan atau memerintahkan seperti yang tertulis dalam surat yang dikirim kepada penerima surat (lihat surat bentuk khusus).

1. Penggolongan surat menurut isi atau kepentingannnya, yaitu:
2. Surat niaga atau bisnis, yaitu surat yang digunakan sebagai alat komunikasi persoalan bisnis dikalangan pengusaha atau badanbadan usaha dalam menjalankan kegiatan usahanya.
3. Surat dinas, yaitu surat yang digunakan di lingkungan pemerintahan atau dinas-dinas pemerintahan sebagai alat komunikasi dalam menjalankan kegiatan kedinasan.
4. Surat sosial, yaitu surat yang digunakan oleh lembaga/badan sosial sebagai alat komunikasi dalam menjalankan aktivitasnya yang bersifat non profit.
5. Surat bentuk lain, yaitu surat yang dapat digunakan secara cepat untuk suatu kepentingan yang mendadak atau harus dilaksanakan secara gerak cepat.
6. Surat pribadi atau personal, yaitu surat yang digunakan sebagai alat komunikasi diantara keluarga atau kerabat. Isi surat pribadi adalah tentang kekeluargaan dan persahabatan.
7. Penggolongan surat berdasarkan keamanan isinya:
8. Surat sangat rahasia, yaitu surat atau dokumen yang diberi tanda “RHS” dan pada umumnya kode ini digunakan dalam lingkungan bisnis/organisasi/lembaga/perorangan.
9. Surat rahasia atau sering disebut surat konfidensial, yaitu surat yang isinya tidak boleh diketahui oleh orang lain selain yang namanya atau jabatannya disebutkan pada alamat surat tersebut.
10. Surat biasa, yaitu surat pada umumnya yang dapat dilakukan dengan menggunakan Kartu Pos, Warkat Pos atau surat tertutup yang dimasukkan ke dalam sampul atau cara lain. Apabila isinya dibaca orang lain tidak akan menimbulkan suatu akibat buruk atau merugikan yang bersangkutan atau organisasi/pejabat yang bersangkutan .
11. Penggolongan surat berdasarkan urgensi pengiriman/penyelesaiannya:
12. Surat “Sangat Segera” atau kilat, yaitu surat yang harus dikirim dan secepatnya harus diketahui oleh penerima surat dan mendapat tanggapan atas isi surat dengan secepat mungkin.
13. Surat “Segera”, yaitu surat yang perlu mendapat tanggapan dari penerima surat atas isinya sesegara mungkin ditindaklanjuti.
14. Surat Biasa, yaitu surat yang isinya maupun cara pengirimannya tidak memerlukan cara secepatnya, tetapi sesuai uratan penyelesaian berdasarkan jadwal aktivitasnya.

## Pengertian Surat Perintah Tugas

yaitu surat yang isinya memerintahkan suatu hal untuk dilaksanakan oleh penerima surat. Suryani,dkk (2013:4), Surat Perintah berisi macam macam, sesuai dengan tugas atau perintah apa yang telah dimandadkan oleh pemberi tugas. Sebagai salah satu surat resmi, surat perintah wajib mencantumkan Kop Surat dari instansi atau perusahaan pembuat surat dengan alamat yang lengkap. Untuk menandakan bahwa surat yang dikeluarkan merupakan surat perintah maka wajib and kudu menuliskan kalimat ‘Surat Perintah’ di bagian bawah kop surat sebagai judul surat.  
Setelah judul, merupakan badan surat, yaitu isi dari surat perintah tersebut. Termasuk didalamnya nama-nama beserta jabatan yang lengkap, NIP, golongan dari yang bersangkutan yang akan diperintahkan dan detail tugas yang akan diberikan kepada nama-nama yang tertera dalam surat perintah tersebut. Terakhir adalah penutup surat. untuk penutup surat, tuliskan nama atasan yang memberi tugas, tanggal, bulan dan tahun pembuatan surat, sehingga jelas surat perintah itu kapan dikeluarkan.  
Untuk diperhatikan, dalam Surat Perintah Tugas (SPT) juga di cantumkan poin poin pekerjaan yang harus dituntaskan.

## Perangkat Analisa Sistem

Berikut ini adalah beberapa perangkat analisa sistem yang digunakan.

### *Use Case Diagram*

*Use case* *Diagram* adalah gambar dari beberapa atau seluruh actor dan usecase dengan tujuan mengenali interaksi mereka dalam suatu sistem. Use Case Diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem, yang ditekankan adalah adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah use case mempresentasikan sebuah interaksi antara actor dengan sistem. Use case menggambarkan kata kerja seperti login ke sistem, maintenance user dan sebagainya. Oleh karena itu, use case diagram dapat membantu menganalisa kebutuhan suatu sistem. Dalam use case diagram terdapat istilah seperti actor, use case dan use case relationship (Yasin, 2012).

### *Class Diagram*

*Class Diagram* UML digunakan untuk menunjukkan kelas objek untuk suatu sistem. Pada *Class Diagram*, persegi panjang mewakili kelas, dan garis yang menghubungkan persegi panjang menunjukkan asosiasi di antara kelas. Pada *Class Diagram*, ada tiga jenis hubungan di antara kelas objek: hubungan asosiasi, hubungan generalisasi / spesialisasi, dan hubungan keseluruhan / bagian (Satzinger et al., 2012).

### *Sequence Diagram*

*Sequence diagram* digunakan untuk menggambarkan interaksi antara aktor dan sistemdi dalam dan di sekitar aplikasi yang berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri antara dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait)*. Message* adalah tindakan yang dipanggil pada objek tujuan, seperti sebuah perintah (Satzinger et al., 2012).

### *Deployment Diagram*

*Deployment diagram* digunakan untuk merepresentasikan hubungan antara komponen perangkat keras yang digunakan dalam infrastruktur fisik dari suatu sistem informasi. Misalnya, kapan merancang sistem informasi yang akan menggunakan jaringan area luas, penyebaran diagram dapat digunakan untuk menunjukkan hubungan komunikasi di antara yang berbeda *node* dalam jaringan. *Deployment diagram* juga dapat digunakan untuk mewakili komponen perangkat lunak dan bagaimana komponen tersebut dikerahkan atas arsitektur fisik atau infrastruktur suatu sistem informasi (Dennis, Wixom, & Tegarden, 2009).

## *MySQL*

*MySQL* merupakan salah satu sistem *database* yang sangat handal karena menggunakan sistem *SQL*. Pada awalnya *SQL* berfungsi sebagai bahasa penghubung antara program *database* dengan bahasa pemrograman yang digunakan. *SQL* dikembangkan lagi menjadi sistem *database* dengan munculnya *MySQL*. *MySQL* merupakan sistem manajemen *database*, relasional sistem *database*, dan *software open source* (Sakur, 2005).

## *Hypertext Markup Language* (HTML)

Secara harfiah, HTML adalah *Hypertext Markup Language.* HTML adalah *script* pemrograman yang mengatur bagaimana menyajikan informasi di dunia internet dan bagaimana membawa pengguna dari satu dokumen informasi ke dokumen informasi lainnya(Enterprise, 2014).

## *Codeigniter* *Framework*

Codeigniter adalah sebuah framework php yang bersifat open source dan menggunakan metode MVC (Model, View, Controller). Codeigniter adalah framework gratis jika ingin menggunakannya. Framework codeigniter dibuat dengan tujuan sama seperti framework lainnya yaitu untuk memudahkan developer atau programmer dalam membangun sebuah aplikasi berbasis web tanpa harus membuatnya dari awal.

# ANALISA DAN PERANCANGAN



## Analisa Sistem Lama

Dalam merancang sebuah sistem yang baik dan sesuai dengan hasil yang diharapkan, maka terlebih dahulu harus memperoleh data atau informasi tentang sistem yang sedang berjalan. Dari informasi tersebut, dapat diketahui sejauh mana sistem yang sedang berjalan saat ini dapat memenuhi kebutuhan dan kebutuhan apa saja yang ingin dicapai tetapi belum bisa ditangani oleh sistem yang berjalan.

Sistem manajemen surat perintah tugas di Balai Pemasyarakatan Pekanbaru sebelumnya menggunakan pencatatatn manual dengan menggunakan buku, untuk melakukan manajemen pada surat perintah tugas. Oleh karena itu, analisa sistem lama yang digunakan yaitu analisa sistem yang dilihat dari proses bisnis antara bagian TU, bagian SDM, bagian SDA, dan Kepala.

Proses bisnis dimulai dari pegawai bagian tata usaha mencatat data-data surat masuk yang didapat, seperti nomor surat, tanggal surat, perihaj, pengirim dan data penting lainnya mengenai surat masuk. Setelah selesai melakukan pendataan terhadap surat masuk dilanjutkan dengan pembuatan disposisi surat masuk yang digunakan untuk meminta persetujuan kepada kepala. Setelah selesai membuat disposisi, kemudian lembar disposisi diajukan ke kepala untuk meminta persetujuan, pada saat kepala menerima surat disposisi, kepala harus menentukan surat masuk yang telah didisposisi tersebut diarahkan ke bagian mana, di balai pemasyarakatan klas ii pekanbaru meiliki dua bagian yaitu BKD dan BKA, BKD adalah bagian yang mengurus klien dewasa sedangkan BKA akan mengurus klien anak. Setelah kepala menyetujui dan memilih tujuan dari surat masuk yang telah didisposisi tadi barulah disposisi akan dibagikan kepada bagian yang dituju, kemudian setiap bagian baik BKD maupun BKA bisa membuat surat perintah tugas, dengan menginputkan data surat masuk dan pegawai dari masing-masing bagian yang dipilih untuk menangani perihal dari surat tersebut, setelah selesai membuat surat perintah tugas dan kemudian dilanjutkan dengan meminta persetujuan surat perintah tugas kepada kepala, dan apabila kepala menyetujui surat perintah tersebut barulah pegawai yang bersangkutan bisa melakukan tugasnya, dan kemudian data surat perintah tugas yang telah disetujui tersebut diberikan ke bagian tata usaha untuk dilakukan monitoring mengenai surat perintah tugas tersebut. monitoring yang dilakukan mengenai waktu pengerjaan surat perintah tugas, jadi pada saat melakukan tugas mengenai surat perintah tugas tadi pegawai dituntut untuk membuat laporan mengenai perihal yang sedang dilaksanakan, setelah selesai membuat laporan kemudian pegawai yang bersangkutan akan memberikan laporan kepada bagian tat usaha, setiap bagian memiliki tenggang waktu yang berbeda, bagian BKD memiliki tenggang waktu tujuh hari sedangkan BKA memiliki tenggang waktu 3 hari, apabila pegawai tidak memberikan laporan lengkap mengenai perihal yang dia kerjakan ke bagian tata usaha, maka pegawai tersebut tidak diberikan uang selama melaksanakan tugasnya tersebut.

TAHAP

PROSES BISNIS

KEPALA

Mulai

Menuliskan data surat masuk

Membuat disposisi

TATA USAHA

BKA/BKD

Persetujuan disposisi

Membuat surat perintah tugas

Mengajukan surat perintah tugas

Persetujuan surat perintah tugas

Monitoring surat perintah tugas

Selesai

Gambar 5. 1 Flowchart Analisa Sistem Lama

## Analisa Sistem Baru

Analisa sistem baru dirancang sesuai dengan kebutuhan instansi terhadap admin dan pengguna. Pada sistem ini, *login* yang dilakukan oleh admin dan pengguna menggunakan *username dan password*. Tugas Admin TU pada sistem pertama untuk mengelola data surat masuk seperti menginputkan nomor surat, pengirim tanggal surat, perihal dan hal penting lainnya, tugas selanjutnya yaitu memonitoring surat perintah tugas yang telah diajukan oleh kepala, baik surat perintah tugas bagian BKD dan surat perintah bagian BKA sehingga pihak admin tata usaha bisa mengetahui mana surat perintah tugas yang telat dan yang tidak yelat. Tugas Kepala pada sistem untuk mengelola persetujuan disposisi surat masuk, tentunya pihak lain seperti BKD dan BKA harus mendapatan persetujuan dari kepala terlebih dahulu agar bisa lanjut ke tahap selanjutnya dan begitu juga dengan persetujuan surat perintah tugas baik bagian BKD dan bagian BKA, haruslah mendapatkan persetujuan dari kepala agar bisa dilanjutkan ke tahap monitoring surat perintah kerja. Tugas Admin BKD pada sistem untuk mengelola surat perintah tugas bagian BKD, pada saat membuat surat perintah tugas akan ada satu pegawai dari BKD yang ditunjuk untuk menjadi penanggung jawab dalam menyelesaikan tugas yang dibuat disurat perintah tersebut, dan setelah selesai membuat surat perintah tugas, kemudian surat perintah tugas akan diajukan kembali keada kepala untuk meminta persetujuan surat peintah tugas. Tugas Admin BKA pada sistem untuk mengelola surat perintah tugas bagian BKA. Sama dengan prosedur yang ada di Admin BKD melakukan pembuatan surat perintah tugas dan mengajukannya ke kepala untuk meminta persetujuan surat perintah tugas.

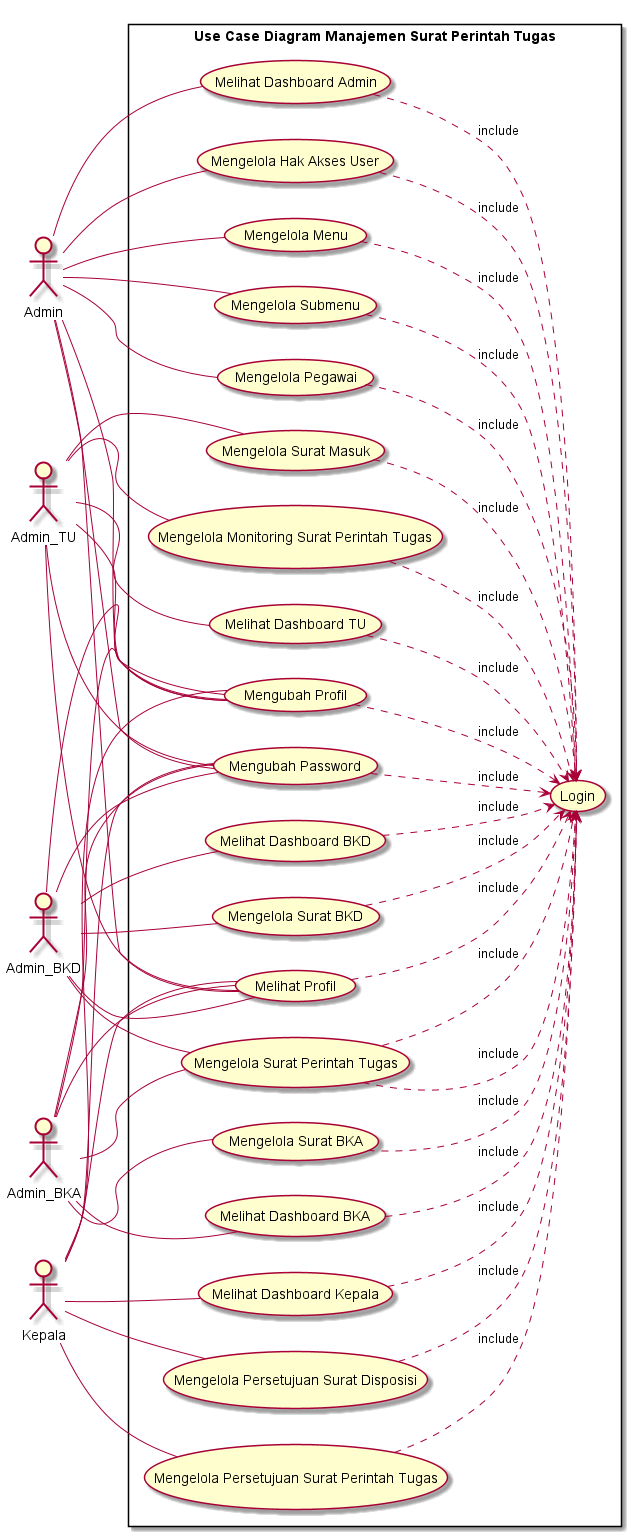
## Kebutuhan Fungsional

Kebutuhan fungsional adalah rancang bangun Sistem Informasi Manajemen Surat Perintah Tugas Berbasis web mempunyai fungsi-fungsi yang bisa membuat sistem informasi ini lebih mendukung proses manajemen surat perintah tugas diantaranya yaitu:

1. Sistem bisa mengelola dan menampilkan informasi surat masuk
2. Sistem bisa mengelola dan menampilkan informasi surat disposisi dan persetujuan disposisi
3. Sistem bisa mengelola proses pembuatan dan pengajuan surat perintah tugas baik di bagian BKD dan bagian BKA
4. Sistem bisa mengelola dan menampilkan informasi surat perintah tugas, melakukan persetujuan surat perintah tugas dan melakukan monitoring terhadap surat perintah tugas.

## *Use Case Diagram*

*Use Case Diagram* berfungsi untuk mendeskripsikan interaksi antara suatu atau banyak aktor ke dalam sistem yang akan dibuat. *Use Case Diagram* juga berguna untuk mengetahui fitur apa saja yang ada dalam sebuah sistem dan siapa saja yang terkait kedalam sistem serta siapa saja yang berhak atau boleh mengakses fitur tersebut. Berikut adalah *Use Case Diagram* Sistem Informasi Sistem Informasi Manajemen Surat Perintah Tugas Berbasis Web.



Gambar 5. 2 Use Case Diagram

Tabel 5.1 berikut ini adalah *Use Case Description* pada *Use Case* *Login* dengan Admin, Admin TU, Admin BKD, Admin BKA, Kepala sebagai *Actor*.

Tabel 5.1 Use Case Description Login

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Login* | |
| *Actor* | Admin, Admin TU, Admin BKD, Admin BKA, Kepala | |
| *Precondition* | Actor belum login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* input *email* dan *password* 2. *Actor* klik *login* | * 1. Sistem memastikan *form* terisi   2. Sistem melakukan validasi |
| *Alternative Flows* | * 1. Apabila data di form email yang diinputkan tipenya tidak email maka akan muncul pesan **the email field must contain a valid email address**   2. Apabila Email belum terdaftar di database maka muncul pesan **email is not registered**   3. Apabila Email sudah terdaftar di database dan password tidak sesuai dengan email maka muncul pesan **wrong password** | |
| *Input* | Email, password | *Output*: - |

Tabel 5.2 berikut ini adalah *Use Case Description* pada *Use Case* *Forgot Password* dengan Admin, Admin TU, Admin BKD, Admin BKA, Kepala sebagai *Actor*.

Tabel 5.2 Use Case Description Forgot Password

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Forgot Password* | |
| *Actor* | Admin, Admin TU, Admin BKD, Admin BKA, Kepala | |
| *Precondition* | Actor belum login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* input *email* 2. Actor klik reset password | * 1. Sistem memastikan *form* terisi   2. Sistem melakukan verifikasi ke email |
| *Alternative Flows* | * 1. Apabila data di form email yang diinputkan tipenya tidak email maka akan muncul pesan **the email field must contain a valid email address**   2. Apabila Actor klik reset password akan muncul pesan p**lease check your email to reset your password** | |
| *Input* | Email | *Output*: - |

Tabel 5.3 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Dashboard dengan Admin, Admin TU, Admin BKD, Admin BKA, Kepala sebagai *Actor*.

Tabel 5.3 Use Case Description Melihat Dashboard

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Melihat Dashboard* | |
| *Actor* | Admin, Admin TU, Admin BKD, Admin BKA, Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu dashboard* | * 1. sistem menampilkan halaman dashboard |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.4 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Profil dengan Admin, Admin TU, Admin BKD, Admin BKA, Kepala sebagai *Actor*.

Tabel 5.4 Use Case Description Melihat Profil

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Melihat Profil* | |
| *Actor* | Admin, Admin TU, Admin BKD, Admin BKA, Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu my profil* | * 1. Sistem menampilkan halaman profil user |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.5 berikut ini adalah *Use Case Description* pada *Use Case* Mengubah Profil dengan Admin, Admin TU, Admin BKD, Admin BKA, Kepala sebagai *Actor*.

Tabel 5.5 Use Case Description Mengubah Profil

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Mengubah Profil* | |
| *Actor* | Admin, Admin TU, Admin BKD, Admin BKA, Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu edit profil 2. Actor mengubah data pada form profil user 3. Actor klik tombol edit | * 1. Sistem menampilkan halaman edit profil   2. Sistem memastikan *form* terisi   3. Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | * 1. Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.   2. Apabila Actor klik edit akan muncul pesan **user berhasil di edit** | |
| *Input* | Email, full name, picture | *Output*: - |

Tabel 5.6 berikut ini adalah *Use Case Description* pada *Use Case* Mengubah Password dengan Admin, Admin TU, Admin BKD, Admin BKA, Kepala sebagai *Actor*.

Tabel 5.6 Use Case Description Mengubah Password

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Mengubah Password* | |
| *Actor* | Admin, Admin TU, Admin BKD, Admin BKA, Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu change password 2. Actor mengubah password pada form change password 3. Actor klik tombol change password | * 1. Sistem menampilkan halaman change password   2. Sistem memastikan *form* terisi   3. Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | * 1. Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.   2. Apabila data di form new password dan repeat password yang diinputkan datanya kurang dari 3digit maka akan muncul pesan **the field must be at least 3 characthers in length**   3. Apabila data di form new password dan repeat password yang diinputkan tidak sama maka akan muncul pesan **the field does not match** | |
| *Input* | Current password, new password, repeat password | *Output*: - |

Tabel 5.7 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Role dengan Admin sebagai *Actor*.

Tabel 5.7 Use Case Description Melihat Role

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Melihat Role* | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu role* | * 1. sistem menampilkan halaman role |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.8 berikut ini adalah *Use Case Description* pada *Use Case* Menambah Role dengan Admin sebagai *Actor*.

Tabel 5.8 Use Case Description Menambah Role

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Menambah Role* | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu role 2. *Acto*r klik tombol add role 3. Actor menambah data pada form tambah role 4. Actor klik tombol add | * 1. Sistem menampilkan halaman role   2.1 Sistem menampilkan halaman add role  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik add akan muncul pesan **role berhasil ditambah** | |
| *Input* | Role | *Output*: - |

Tabel 5.9 berikut ini adalah *Use Case Description* pada *Use Case* Mengubah Role dengan Admin sebagai *Actor*.

Tabel 5.9 Use Case Description Mengubah Role

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Mengubah Role* | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu role 2. *Actor klik tombil edit role* 3. Actor mengubah data pada form edit role 4. Actor klik tombol edit | * 1. Sistem menampilkan halaman role   2.1 Sistem menampilkan halaman edit role  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik edit akan muncul pesan **role berhasil diedit** | |
| *Input* | Role | *Output*: - |

Tabel 5.10 berikut ini adalah *Use Case Description* pada *Use Case* Menghapus Role dengan Admin sebagai *Actor*.

Tabel 5.10 Use Case Description Menghapus Role

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Menghapus Role* | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu role 2. Actor klik tombol delete pada data yang ingin dihapus 3. Actor klik tombol ya pada modal konfirmasi | * 1. Sistem menampilkan halaman role   2. Sistem menampilkan modal konfirmasi apakah anda yakin, data akan dihapus**.**   3. Sistem menghapus data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil. |
| *Alternative Flows* | 3.1 Apabila Actor klik tombol tidak akan muncul pesan **canceled**  3.2 Apabila Actor klik tombol ya akan muncul pesan **role berhasil dihapus** | |
| *Input* | Role | *Output*: - |

Tabel 5.11 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Menu dengan Admin sebagai *Actor*.

Tabel 5.11 Use Case Description Melihat Role

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Melihat Menu* | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu menu management* | * 1. sistem menampilkan halaman menu management |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.12 berikut ini adalah *Use Case Description* pada *Use Case* Menambah Menu dengan Admin sebagai *Actor*.

Tabel 5.12 Use Case Description Menambah Menu

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Menambah Menu* | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu menu management 2. Actor klik tombol add menu 3. Actor menambah data pada form tambah menu 4. Actor klik tombol add | * 1. Sistem menampilkan halaman menu management   2.1 Sistem menampilkan halaman add menu management  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik add akan muncul pesan **menu berhasil ditambah** | |
| *Input* | Menu | *Output*: - |

Tabel 5.13 berikut ini adalah *Use Case Description* pada *Use Case* Mengubah Menu dengan Admin sebagai *Actor*.

Tabel 5.13 Use Case Description Mengubah Menu

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Mengubah* Menu | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu menu management 2. *Actor klik tombol edit menu* 3. Actor mengubah data pada form edit menu management 4. Actor klik tombol edit | * 1. Sistem menampilkan halaman menu management   2.1 Sistem menampilkan halaman edit menu management  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik edit akan muncul pesan **menu berhasil diedit** | |
| *Input* | Role | *Output*: - |

Tabel 5.14 berikut ini adalah *Use Case Description* pada *Use Case* Menghapus Menu dengan Admin sebagai *Actor*.

Tabel 5.14 Use Case Description Menghapus Menu

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Menghapus Menu* | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu menu management 2. Actor klik tombol delete pada data yang ingin dihapus 3. Actor klik tombol ya pada modal konfirmasi | * 1. Sistem menampilkan halaman role   2. Sistem menampilkan modal konfirmasi apakah anda yakin, data akan dihapus**.**   3. Sistem menghapus data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil. |
| *Alternative Flows* | 3.1 Apabila Actor klik tidak akan muncul pesan **canceled**  3.1 Apabila Actor klik ya akan muncul pesan **role berhasil dihapus** | |
| *Input* |  | *Output*: - |

Tabel 5.15 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Submenu dengan Admin sebagai *Actor*.

Tabel 5.15 Use Case Description Melihat Submenu

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Melihat* Submenu | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu submenu management* | * 1. sistem menampilkan halaman *submenu* management |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.16 berikut ini adalah *Use Case Description* pada *Use Case* Menambah Submenu dengan Admin sebagai *Actor*.

Tabel 5.16 Use Case Description Menambah Submenu

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Menambah Submenu* | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu submenumanagement 2. *Actor klik tombol add submenu* 3. Actor menambah data pada form tambah submenu 4. Actor klik tombol add | * 1. Sistem menampilkan halaman submenumanagement   2. Sistem menampilkan halaman add submenumanagement   3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik add akan muncul pesan **submenu****berhasil ditambah** | |
| *Input* | Title, menu, url, icon, active | *Output*: - |

Tabel 5.17 berikut ini adalah *Use Case Description* pada *Use Case* Mengubah Submenu dengan Admin sebagai *Actor*.

Tabel 5.17 Use Case Description Mengubah Submenu

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Mengubah* Submenu | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu submenu management 2. *Actor klik tombol edit submenu* 3. Actor mengubah data pada form edit submenu management 4. Actor klik tombol edit | * 1. Sistem menampilkan halaman submenu management   2.1 Sistem menampilkan halaman edit submenu management  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik edit akan muncul pesan **subemenu berhasil diedit** | |
| *Input* | Title, menu, url, icon, active | *Output*: - |

Tabel 5.18 berikut ini adalah *Use Case Description* pada *Use Case* Menghapus Submenu dengan Admin sebagai *Actor*.

Tabel 5.18 Use Case Description Menghapus Submenu

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Menghapus Submenu* | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu submenu management 2. Actor klik hapus pada data yang ingin dihapus 3. Actor klik tombol ya pada modal konfirmasi | * 1. Sistem menampilkan halaman submenu   2. Sistem menampilkan modal konfirmasi apakah anda yakin, data akan dihapus**.**   3. Sistem menghapus data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil. |
| *Alternative Flows* | 3.1 Apabila Actor klik tidak akan muncul pesan **canceled**  3.2 Apabila Actor klik edit akan muncul pesan **submenu berhasil dihapus** | |
| *Input* |  | *Output*: - |

Tabel 5.19 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Pegawai TU dengan Admin sebagai *Actor*.

Tabel 5.19 Use Case Description Melihat Pegawai TU

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Pegawai TU | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu pegawai TU* | * 1. sistem menampilkan halaman *pegawau TU* |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.20 berikut ini adalah *Use Case Description* pada *Use Case* Menambah Pegawai TU dengan Admin sebagai *Actor*.

Tabel 5.20 Use Case Description Menambah Pegawai TU

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Menambah* Pegawai TU | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Pegawai TU 2. *Actor klik tombol add pegawai* 3. Actor menambah data pada form tambah Pegawai TU 4. Actor klik tombol add | * 1. Sistem menampilkan halaman Pegawai TU   2.1 Sistem menampilkan halaman add Pegawai TU  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik add akan muncul pesan **Pegawai****berhasil ditambah** | |
| *Input* | Nama, NIP, jenis-kelamin, tempat-lahir, tanggal-lahir, alamat, jabatan, bagian | *Output*: - |

Tabel 5.21 berikut ini adalah *Use Case Description* pada *Use Case* Mengubah Pegawai TU dengan Admin sebagai *Actor*.

Tabel 5.21 Use Case Description Mengubah Pegawai TU

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Mengubah pegawai TU* | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu pegawai TU 2. *Actor klik tombol edit pegawai* 3. Actor mengubah data pada form edit pegawai TU 4. Actor klik tombol edit | * 1. Sistem menampilkan halaman pegawai TU   2.1 Sistem menampilkan halaman edit pegawai TU  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik edit akan muncul pesan **pegawai berhasil diedit** | |
| *Input* | Nama, NIP, jenis-kelamin, tempat-lahir, tanggal-lahir, alamat, jabatan, bagian | *Output*: - |

Tabel 5.22 berikut ini adalah *Use Case Description* pada *Use Case* Menghapus Pegawai TU dengan Admin sebagai *Actor*.

Tabel 5.21 Use Case Description Menghapus Pegawai TU

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Menghapus* Pegawai TU | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Pegawai TU 2. Actor klik delete pada data yang ingin dihapus 3. Actor klik tombol ya pada modal konfirmasi | * 1. Sistem menampilkan halaman Pegawai TU   2. Sistem menampilkan modal konfirmasi apakah anda yakin, data akan dihapus**.**   3. Sistem menghapus data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil. |
| *Alternative Flows* | 3.1 Apabila Actor klik tidak akan muncul pesan **canceled**  3.2 Apabila Actor klik edit akan muncul pesan **pegawai berhasil dihapus** | |
| *Input* |  | *Output*: - |

Tabel 5.23 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Pegawai BKD dengan Admin sebagai *Actor*.

Tabel 5.23 Use Case Description Melihat Pegawai BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Pegawai BKD | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu pegawai* BKD | * 1. sistem menampilkan halaman *pegawai* BKD |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.24 berikut ini adalah *Use Case Description* pada *Use Case* Menambah Pegawai BKD dengan Admin sebagai *Actor*.

Tabel 5.24 Use Case Description Menambah Pegawai BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Menambah* Pegawai BKD | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Pegawai BKD 2. *Actor klik tombol add pegawai* 3. Actor menambah data pada form tambah Pegawai BKD 4. Actor klik tombol add | * 1. Sistem menampilkan halaman Pegawai BKD   2.1 Sistem menampilkan halaman add Pegawai BKD  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik add akan muncul pesan **Pegawai****berhasil ditambah** | |
| *Input* | Nama, NIP, jenis-kelamin, tempat-lahir, tanggal-lahir, alamat, jabatan, bagian | *Output*: - |

Tabel 5.25 berikut ini adalah *Use Case Description* pada *Use Case* Mengubah Pegawai BKD dengan Admin sebagai *Actor*.

Tabel 5.25 Use Case Description Mengubah Pegawai BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Mengubah pegawai* BKD | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu pegawai BKD 2. *Actor klik tombol edit pegawai* 3. Actor mengubah data pada form edit pegawai BKD 4. Actor klik tombol edit | * 1. Sistem menampilkan halaman pegawai BKD   2.1 Sistem menampilkan halaman edit pegawai BKD  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik edit akan muncul pesan **pegawai berhasil diedit** | |
| *Input* | Nama, NIP, jenis-kelamin, tempat-lahir, tanggal-lahir, alamat, jabatan, bagian | *Output*: - |

Tabel 5.26 berikut ini adalah *Use Case Description* pada *Use Case* Menghapus Pegawai BKD dengan Admin sebagai *Actor*.

Tabel 5.26 Use Case Description Menghapus Pegawai BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Menghapus* Pegawai BKD | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Pegawai BKD 2. Actor klik delete pada data yang ingin dihapus 3. Actor klik tombol ya pada modal konfirmasi | * 1. Sistem menampilkan halaman Pegawai BKD   2. Sistem menampilkan modal konfirmasi apakah anda yakin, data akan dihapus**.**   3. Sistem menghapus data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil. |
| *Alternative Flows* | 3.1 Apabila Actor klik tidak akan muncul pesan **canceled**  3.2 Apabila Actor klik edit akan muncul pesan **pegawai berhasil dihapus** | |
| *Input* |  | *Output*: - |

Tabel 5.27 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Pegawai BKA dengan Admin sebagai *Actor*.

Tabel 5.27 Use Case Description Melihat Pegawai BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Pegawai BKA | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu pegawai* BKA | * 1. sistem menampilkan halaman *pegawai* BKA |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.5 berikut ini adalah *Use Case Description* pada *Use Case* Menambah Pegawai BKA dengan Admin sebagai *Actor*.

Tabel 5.2 Use Case Description Menambah Pegawai BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Menambah* Pegawai BKA | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Pegawai BKA 2. *Actor klik tombol add pegawai* 3. Actor menambah data pada form tambah Pegawai BKA 4. Actor klik tombol add | * 1. Sistem menampilkan halaman Pegawai BKA   2.1 Sistem menampilkan halaman add Pegawai BKA  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik add akan muncul pesan **Pegawai****berhasil ditambah** | |
| *Input* | Nama, NIP, jenis-kelamin, tempat-lahir, tanggal-lahir, alamat, jabatan, bagian | *Output*: - |

Tabel 5.28 berikut ini adalah *Use Case Description* pada *Use Case* Mengubah Pegawai BKA dengan Admin sebagai *Actor*.

Tabel 5.28 Use Case Description Mengubah Pegawai BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Mengubah pegawai* BKA | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu pegawai BKA 2. *Actor klik tombol edit pegawai* 3. Actor mengubah data pada form edit egawai BKA 4. Actor klik tombol edit | * 1. Sistem menampilkan halaman pegawai BKA   2.1 Sistem menampilkan halaman edit pegawai BKA  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik edit akan muncul pesan **pegawai berhasil diedit** | |
| *Input* | Nama, NIP, jenis-kelamin, tempat-lahir, tanggal-lahir, alamat, jabatan, bagian | *Output*: - |

Tabel 5.29 berikut ini adalah *Use Case Description* pada *Use Case* Menghapus Pegawai BKA dengan Admin sebagai *Actor*.

Tabel 5.29 Use Case Description Menghapus Pegawai BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Menghapus* Pegawai BKA | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Pegawai BKA 2. Actor klik delete pada data yang ingin dihapus 3. Actor klik tombol ya pada modal konfirmasi | * 1. Sistem menampilkan halaman Pegawai BKA   2. Sistem menampilkan modal konfirmasi apakah anda yakin, data akan dihapus**.**   3. Sistem menghapus data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil. |
| *Alternative Flows* | 3.1 Apabila Actor klik tidak akan muncul pesan **canceled**  3.2 Apabila Actor klik edit akan muncul pesan **pegawai berhasil dihapus** | |
| *Input* |  | *Output*: - |

Tabel 5.30 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Trash Pegawai dengan Admin sebagai *Actor*.

Tabel 5.30 Use Case Description Melihat Trash Pegawai

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Trash Pegawai | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu trash* | * 1. sistem menampilkan halaman trash *pegawai* |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.31 berikut ini adalah *Use Case Description* pada *Use Case* Restore Pegawai dengan Admin sebagai *Actor*.

Tabel 5.31 Use Case Description Restore Pegawai

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Restore Pegawai | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu trash 2. Actor klik tombol restore pada data yang ingin dikembalikan 3. Actor klik tombol ya pada modal konfirmasi | * 1. Sistem menampilkan halaman trash pegawai   2. Sistem menampilkan modal konfirmasi apakah anda yakin, data akan dikembalikan**.**   3. Sistem mengembalikan data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil. |
| *Alternative Flows* | 3.1 Apabila Actor klik tidak akan muncul pesan **canceled**  3.2 Apabila Actor klik edit akan muncul pesan **pegawai berhasil direstore** | |
| *Input* |  | *Output*: - |

Tabel 5.32 berikut ini adalah *Use Case Description* pada *Use Case* Hapus Permanen Pegawai dengan Admin sebagai *Actor*.

Tabel 5.32 Use Case Description Hapus Permanen Pegawai

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Hapus Permanen Pegawai | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu trash 2. Actor klik tombol delete permanent pada data yang ingin dihapus permanen 3. Actor klik tombol ya pada modal konfirmasi | * 1. Sistem menampilkan halaman trash pegawai   2. Sistem menampilkan modal konfirmasi apakah anda yakin, data akan dihapus permanen**.**   3. Sistem menghapus permanen data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil. |
| *Alternative Flows* | 3.1 Apabila Actor klik tidak akan muncul pesan **canceled**  3.2 Apabila Actor klik edit akan muncul pesan **pegawai berhasil dihapus permanen** | |
| *Input* |  | *Output*: - |

Tabel 5.33 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Surat Masuk dengan Admin TU sebagai *Actor*.

Tabel 5.33 Use Case Description Melihat Surat Masuk

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Surat Masuk | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu surat masuk* | * 1. sistem menampilkan halaman *surat masuk* |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.34 berikut ini adalah *Use Case Description* pada *Use Case* Menambah Surat Masuk dengan Admin TU sebagai *Actor*.

Tabel 5.34 Use Case Description Menambah Surat Masuk

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Menambah Surat Masuk | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu surat masuk 2. *Actor klik tombol add surat masuk* 3. Actor menambah data pada form tambah surat masuk 4. Actor klik tombol add | * 1. Sistem menampilkan halaman surat masuk   2.1 Sistem menampilkan halaman add surat masuk  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik add akan muncul pesan **surat masuk****berhasil ditambah** | |
| *Input* | Pengirim, no surat, tanggal surat masuk, ringkasan, file surat | *Output*: - |

Tabel 5.35 berikut ini adalah *Use Case Description* pada *Use Case* Mengubah Surat masuk dengan Admin TU sebagai *Actor*.

Tabel 5.35 Use Case Description Mengubah Surat masuk

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Mengubah* Surat masuk | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu surat masuk 2. *Actor klik tombol edit* surat masuk 3. Actor mengubah data pada form edit surat masuk 4. Actor klik tombol edit | * 1. Sistem menampilkan halaman surat masuk   2.1 Sistem menampilkan halaman edit surat masuk  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik edit akan muncul pesan **surat masuk berhasil diedit** | |
| *Input* | Nama, NIP, jenis-kelamin, tempat-lahir, tanggal-lahir, alamat, jabatan, bagian | *Output*: - |

Tabel 5.36 berikut ini adalah *Use Case Description* pada *Use Case* Menghapus Surat Masuk dengan Admin TU sebagai *Actor*.

Tabel 5.36 Use Case Description Menghapus Surat Masuk

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Menghapus* Surat Masuk | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu surat masuk 2. Actor klik delete pada data yang ingin dihapus 3. Actor klik tombol ya pada modal konfirmasi | * 1. Sistem menampilkan halaman surat masuk   2. Sistem menampilkan modal konfirmasi apakah anda yakin, data akan dihapus**.**   3. Sistem menghapus data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil. |
| *Alternative Flows* | 3.1 Apabila Actor klik tidak akan muncul pesan **canceled**  3.2 Apabila Actor klik edit akan muncul pesan **surat masuk berhasil dihapus** | |
| *Input* |  | *Output*: - |

Tabel 5.53 berikut ini adalah *Use Case Description* pada *Use Case* Melihat File Disposisi dengan Admin TU sebagai *Actor*.

Tabel 5.53 Use Case Description Melihat File Disposisi BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat File Disposisi | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Surat Masuk 2. Actor klik tombol lihat file disposisi pada data yang ingin dilihat | * 1. Sistem menampilkan halaman surat masuk   2. Sistem menampilkan file disposisi yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.37 berikut ini adalah *Use Case Description* pada *Use Case* Mendisposisi Surat Masuk dengan Admin TU sebagai *Actor*.

Tabel 5.37 Use Case Description Mendisposisi Surat Masuk

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mendisposisi Surat Masuk | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu surat masuk 2. Actor klik tombol disposisi pada data yang ingin didisposisi 3. Actor klik tombol ya pada modal konfirmasi | * 1. Sistem menampilkan halaman surat masuk   2. Sistem menampilkan modal konfirmasi apakah anda yakin, data akan didisposisi**.**   3. Sistem memvalidasi data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil. |
| *Alternative Flows* | 3.1 Apabila Actor klik tidak akan muncul pesan **canceled**  3.2 Apabila Actor klik edit akan muncul pesan **surat masuk berhasil didisposisi** | |
| *Input* |  | *Output*: - |

Tabel 5.38 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Trash Surat Masuk dengan Admin TU sebagai *Actor*.

Tabel 5.38 Use Case Description Melihat Trash Surat Masuk

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Trash Surat Masuk | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu trash* | * 1. sistem menampilkan halaman trash surat masuk |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.39 berikut ini adalah *Use Case Description* pada *Use Case* Restore Surat Masuk dengan Admin TU sebagai *Actor*.

Tabel 5.39 Use Case Description Restore Surat Masuk

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Restore Surat Masuk | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu trash 2. Actor klik tombol restore pada data yang ingin dikembalikan 3. Actor klik tombol ya pada modal konfirmasi | * 1. Sistem menampilkan halaman trash surat masuk   2. Sistem menampilkan modal konfirmasi apakah anda yakin, data akan dikembalikan**.**   3. Sistem mengembalikan data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil. |
| *Alternative Flows* | 3.1 Apabila Actor klik tidak akan muncul pesan **canceled**  3.2 Apabila Actor klik edit akan muncul pesan **surat masuk berhasil direstore** | |
| *Input* |  | *Output*: - |

Tabel 5.40 berikut ini adalah *Use Case Description* pada *Use Case* Hapus Permanen Surat Masuk dengan Admin TU sebagai *Actor*.

Tabel 5.40 Use Case Description Hapus Permanen Surat Masuk

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Hapus Permanen Surat Masuk | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu trash 2. Actor klik tombol delete permanent pada data yang ingin dihapus permanen 3. Actor klik tombol ya pada modal konfirmasi | * 1. Sistem menampilkan halaman trash surat masuk   2. Sistem menampilkan modal konfirmasi apakah anda yakin, data akan dihapus permanen**.**   3. Sistem menghapus permanen data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil. |
| *Alternative Flows* | 3.1 Apabila Actor klik tidak akan muncul pesan **canceled**  3.2 Apabila Actor klik edit akan muncul pesan **surat masuk berhasil dihapus permanen** | |
| *Input* |  | *Output*: - |

Tabel 5.41 berikut ini adalah *Use Case Description* pada *Use Case* Melihat SPT BKD dengan Admin TU sebagai *Actor*.

Tabel 5.41 Use Case Description Melihat SPT BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat SPT BKD | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu SPT BKD* | * 1. sistem menampilkan halaman SPT BKD |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.42 berikut ini adalah *Use Case Description* pada *Use Case* Melihat File SPT BKD dengan Admin TU sebagai *Actor*.

Tabel 5.42 Use Case Description Melihat File SPT BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat File SPT BKD | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT BKD 2. Actor klik tombol lihat file SPT BKD pada file yang ingin dilihat | * 1. Sistem menampilkan halaman SPT BKD   2. Sistem menampilkan file SPT yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.43 berikut ini adalah *Use Case Description* pada *Use Case* Melihat File SPT Lengkap BKD dengan Admin TU sebagai *Actor*.

Tabel 5.43 Use Case Description Melihat File SPT Lengkap BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat File SPT Lengkap BKD | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT lengkap BKD 2. Actor klik tombol lihat file SPT lengkap BKD pada file yang ingin dilihat | * 1. Sistem menampilkan halaman SPT BKD   2. Sistem menampilkan file SPT lengkap yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.44 berikut ini adalah *Use Case Description* pada *Use Case* Mengupload SPT Lengkap BKD dengan Admin TU sebagai *Actor*.

Tabel 5.44 Use Case Description Mengupload SPT Lengkap BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mengupload SPT Lengkap BKD | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT BKD 2. *Actor klik tombol upload SPT lengkap* pada spt yang ingin di upload 3. Actor menambah file spt lengkap yang ingin di upload pada form upload spt lengkap 4. Actor klik tombol upload | * 1. Sistem menampilkan halaman SPT BKD   2.1 Sistem menampilkan halaman upload SPT lengkap BKD  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik upload akan muncul pesan **SPT lengkap berhasil diedit** | |
| *Input* | SPT lengkap | *Output*: - |

Tabel 5.45 berikut ini adalah *Use Case Description* pada *Use Case* Verifikasi SPT Lengkap BKD dengan Admin TU sebagai *Actor*.

Tabel 5.45 Use Case Description Verifikasi SPT Lengkap BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Verifikasi SPT Lengkap BKD | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT BKD 2. *Actor klik tombol accept SPT lengkap* pada spt yang ingin di verifikasi | * 1. Sistem menampilkan halaman SPT BKD   2.1 Sistem memverifikasi data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil |
| *Alternative Flows* | 2.1 Apabila Actor klik accept akan muncul pesan **SPT lengkap berhasil diverifikasi** | |
| *Input* |  | *Output*: - |

Tabel 5.46 berikut ini adalah *Use Case Description* pada *Use Case* Melihat SPT BKA dengan Admin TU sebagai *Actor*.

Tabel 5.46 Use Case Description Melihat SPT BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat SPT BKA | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu SPT* BKA | * 1. sistem menampilkan halaman SPT BKA |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.47 berikut ini adalah *Use Case Description* pada *Use Case* Melihat File SPT BKA dengan Admin TU sebagai *Actor*.

Tabel 5.47 Use Case Description Melihat File SPT BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat File SPT BKA | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT BKA 2. Actor klik tombol lihat file SPT BKA pada file yang ingin dilihat | * 1. Sistem menampilkan halaman SPT BKA   2. Sistem menampilkan file SPT yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.48 berikut ini adalah *Use Case Description* pada *Use Case* Melihat File SPT Lengkap BKA dengan Admin TU sebagai *Actor*.

Tabel 5.48 Use Case Description Melihat File SPT Lengkap BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat File SPT Lengkap BKA | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT lengkap BKA 2. Actor klik tombol lihat file SPT lengkap BKA pada file yang ingin dilihat | * 1. Sistem menampilkan halaman SPT BKA   2. Sistem menampilkan file SPT lengkap yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.49 berikut ini adalah *Use Case Description* pada *Use Case* Mengupload SPT Lengkap BKA dengan Admin TU sebagai *Actor*.

Tabel 5.49 Use Case Description Mengupload SPT Lengkap BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mengupload SPT Lengkap BKA | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT BKA 2. *Actor klik tombol upload SPT lengkap* pada spt yang ingin di upload 3. Actor menambah file spt lengkap yang ingin di upload pada form upload spt lengkap 4. Actor klik tombol upload | * 1. Sistem menampilkan halaman SPT BKA   2.1 Sistem menampilkan halaman upload SPT lengkap BKA  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik upload akan muncul pesan **SPT lengkap berhasil diedit** | |
| *Input* | SPT lengkap | *Output*: - |

Tabel 5.50 berikut ini adalah *Use Case Description* pada *Use Case* Verifikasi SPT Lengkap BKD dengan Admin TU sebagai *Actor*.

Tabel 5.50 Use Case Description Verifikasi SPT Lengkap BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Verifikasi SPT Lengkap BKA | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT BKA 2. *Actor klik tombol accept SPT lengkap* pada spt yang ingin di verifikasi | * 1. Sistem menampilkan halaman SPT BKA   2.1 Sistem memverifikasi data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil |
| *Alternative Flows* | 2.1 Apabila Actor klik accept akan muncul pesan **SPT lengkap berhasil diverifikasi** | |
| *Input* |  | *Output*: - |

Tabel 5.41 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Persetujuan Disposisi dengan Kepala sebagai *Actor*.

Tabel 5.41 Use Case Description Melihat Persetujuan Disposisi

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Persetujuan Disposisi | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu Persetujuan Disposisi* | * 1. sistem menampilkan halaman *Persetujuan Disposisi* |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.47 berikut ini adalah *Use Case Description* pada *Use Case* Melihat File Disposisi dengan Kepala sebagai *Actor*.

Tabel 5.47 Use Case Description Melihat File Disposisi

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat File Disposisi | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Persetujuan Disposisi 2. Actor klik tombol lihat file disposisi pada file yang ingin dilihat | * 1. Sistem menampilkan halaman Persetujuan Disposisi   2. Sistem menampilkan file disposisi yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.51 berikut ini adalah *Use Case Description* pada *Use Case* Menyetujui Disposisi Ke BKD dengan Kepala sebagai *Actor*.

Tabel 5.51 Use Case Description Menyetujui Disposisi Ke BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Menyetujui Disposisi Ke BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu persetujuan disposisi 2. Actor klik tombol disposisi ke BKD sesuai dengan data yang ingin disetujui 3. Actor klik tombol ya pada modal konfirmasi | * 1. Sistem menampilkan halaman persetujuan disposisi   2. Sistem menampilkan modal konfirmasi apakah anda yakin, surat akan didisposisi ke BKD.   3. Sistem memvalidasi data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil. |
| *Alternative Flows* | 3.1 Apabila Actor klik tidak akan muncul pesan **canceled**  3.2 Apabila Actor klik edit akan muncul pesan **surat masuk berhasil didisposisi ke BKD** | |
| *Input* |  | *Output*: - |

Tabel 5.51 berikut ini adalah *Use Case Description* pada *Use Case* Menyetujui Disposisi Ke BKA dengan Kepala sebagai *Actor*.

Tabel 5.51 Use Case Description Menyetujui Disposisi Ke BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Menyetujui Disposisi Ke BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu persetujuan disposisi 2. Actor klik tombol disposisi ke BKA sesuai dengan data yang ingin disetujui 3. Actor klik tombol ya pada modal konfirmasi | * 1. Sistem menampilkan halaman persetujuan disposisi   2. Sistem menampilkan modal konfirmasi apakah anda yakin, surat akan didisposisi ke BKA.   3. Sistem memvalidasi data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil. |
| *Alternative Flows* | 3.1 Apabila Actor klik tidak akan muncul pesan **canceled**  3.2 Apabila Actor klik edit akan muncul pesan **surat masuk berhasil didisposisi ke BKA** | |
| *Input* |  | *Output*: - |

Tabel 5.52 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Disposisi BKD dengan Kepala sebagai *Actor*.

Tabel 5.52 Use Case Description Melihat Disposisi BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Disposisi BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu Disposisi* BKD | * 1. sistem menampilkan halaman Disposisi BKD |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.53 berikut ini adalah *Use Case Description* pada *Use Case* Melihat File Disposisi BKD dengan Kepala sebagai *Actor*.

Tabel 5.53 Use Case Description Melihat File Disposisi BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat File Disposisi BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Disposisi BKD 2. Actor klik tombol lihat file disposisi BKD pada file yang ingin dilihat | * 1. Sistem menampilkan halaman Disposisi BKD   2. Sistem menampilkan file disposisi yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.54 berikut ini adalah *Use Case Description* pada *Use Case* Melihat File Persetujuan Disposisi BKD dengan Kepala sebagai *Actor*.

Tabel 5.54 Use Case Description Melihat File Persetujuan Disposisi BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat File Persetujuan Disposisi BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu disposisi BKD 2. Actor klik tombol lihat file persetujuan disposisi BKD pada file yang ingin dilihat | * 1. Sistem menampilkan halaman SPT BKD   2. Sistem menampilkan file persetujuan disposisi yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.55 berikut ini adalah *Use Case Description* pada *Use Case* Mengupload Persetujuan Disposisi BKD dengan Kepala sebagai *Actor*.

Tabel 5.55 Use Case Description Mengupload Persetujuan Disposisi BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mengupload Persetujuan Disposisi BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu disposisi BKD 2. *Actor klik tombol upload persetujuan disposisi* pada disposisi yang ingin di upload 3. Actor menambah file persetujuan disposisi yang ingin di upload pada form upload persetujuan disposisi 4. Actor klik tombol upload | * 1. Sistem menampilkan halaman disposisi BKD   2.1 Sistem menampilkan halaman upload Persetujuan Disposisi BKD  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik upload akan muncul pesan **persetujuan disposisi** **berhasil diupload** | |
| *Input* | Persetujuan Disposisi | *Output*: - |

Tabel 5.56 berikut ini adalah *Use Case Description* pada *Use Case* Setujui Disposisi BKD dengan Kepala sebagai *Actor*.

Tabel 5.56 Use Case Description Setujui Disposisi BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Setujui Disposisi BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu disposisi BKD 2. *Actor klik tombol setuju disposisi* pada disposisi yang ingin di setujui | * 1. Sistem menampilkan halaman disposisi BKD   2.1 Sistem memverifikasi data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil |
| *Alternative Flows* | 2.1 Apabila Actor klik accept akan muncul pesan **Disposisi berhasil disetujui** | |
| *Input* |  | *Output*: - |

Tabel 5.57 berikut ini adalah *Use Case Description* pada *Use Case* Setujui Disposisi BKD dengan Kepala sebagai *Actor*.

Tabel 5.57 Use Case Description Tolak Disposisi BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Tolak Disposisi BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu disposisi BKD 2. *Actor klik tombol tolak disposisi* pada disposisi yang ingin di setujui | * 1. Sistem menampilkan halaman disposisi BKD   2.1 Sistem memverifikasi data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil |
| *Alternative Flows* | 2.1 Apabila Actor klik accept akan muncul pesan **Disposisi berhasil ditolak** | |
| *Input* |  | *Output*: - |

Tabel 5.58 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Disposisi BKA dengan Kepala sebagai *Actor*.

Tabel 5.58 Use Case Description Melihat SPT BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Disposisi BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu Disposisi* BKA | * 1. sistem menampilkan halaman Disposisi BKA |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.59 berikut ini adalah *Use Case Description* pada *Use Case* Melihat File Disposisi BKA dengan Kepala sebagai *Actor*.

Tabel 5.59 Use Case Description Melihat File Disposisi BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat File Disposisi BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Disposisi BKA 2. Actor klik tombol lihat file disposisi BKA pada file yang ingin dilihat | * 1. Sistem menampilkan halaman Disposisi BKA   2. Sistem menampilkan file disposisi yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.60 berikut ini adalah *Use Case Description* pada *Use Case* Melihat File Persetujuan Disposisi BKA dengan Kepala sebagai *Actor*.

Tabel 5.60 Use Case Description Melihat File Persetujuan Disposisi BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat File Persetujuan Disposisi BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu disposisi BKA 2. Actor klik tombol lihat file persetujuan disposisi BKA pada file yang ingin dilihat | * 1. Sistem menampilkan halaman SPT BKA   2. Sistem menampilkan file persetujuan disposisi yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.61 berikut ini adalah *Use Case Description* pada *Use Case* Mengupload Persetujuan Disposisi BKA dengan Kepala sebagai *Actor*.

Tabel 5.61 Use Case Description Mengupload Persetujuan Disposisi BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mengupload Persetujuan Disposisi BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu disposisi BKA 2. *Actor klik tombol upload persetujuan disposisi* pada disposisi yang ingin di upload 3. Actor menambah file persetujuan disposisi yang ingin di upload pada form upload persetujuan disposisi 4. Actor klik tombol upload | * 1. Sistem menampilkan halaman disposisi BKA   2.1 Sistem menampilkan halaman upload Persetujuan Disposisi BKA  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik upload akan muncul pesan **persetujuan disposisi** **berhasil diupload** | |
| *Input* | Persetujuan Disposisi | *Output*: - |

Tabel 5.62 berikut ini adalah *Use Case Description* pada *Use Case* Setujui Disposisi BKA dengan Kepala sebagai *Actor*.

Tabel 5.62 Use Case Description Setujui Disposisi BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Setujui Disposisi BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu disposisi BKA 2. *Actor klik tombol setuju disposisi* pada disposisi yang ingin di setujui | * 1. Sistem menampilkan halaman disposisi BKA   2.1 Sistem memverifikasi data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil |
| *Alternative Flows* | 2.1 Apabila Actor klik accept akan muncul pesan **Disposisi berhasil disetujui** | |
| *Input* |  | *Output*: - |

Tabel 5.63 berikut ini adalah *Use Case Description* pada *Use Case* Setujui Disposisi BKA dengan Kepala sebagai *Actor*.

Tabel 5.63 Use Case Description Tolak Disposisi BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Tolak Disposisi BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu disposisi BKA 2. *Actor klik tombol tolak disposisi* pada disposisi yang ingin di setujui | * 1. Sistem menampilkan halaman disposisi BKA   2.1 Sistem memverifikasi data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil |
| *Alternative Flows* | 2.1 Apabila Actor klik accept akan muncul pesan **Disposisi berhasil ditolak** | |
| *Input* |  | *Output*: - |

Tabel 5.64 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Persetujuan SPT BKD dengan Kepala sebagai *Actor*.

Tabel 5.64 Use Case Description Melihat Persetujuan SPT BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Case* Melihat Persetujuan SPT BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu* persetujuan SPTBKD | * 1. sistem menampilkan halaman persetujuan SPT BKD |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.65 berikut ini adalah *Use Case Description* pada *Use Case* Melihat File SPT BKD dengan Kepala sebagai *Actor*.

Tabel 5.65 Use Case Description Melihat File SPT BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat File SPT BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Persetujuan SPT BKD 2. Actor klik tombol lihat file SPT BKD pada file yang ingin dilihat | * 1. Sistem menampilkan halaman persetujuan SPT BKD   2. Sistem menampilkan file SPT yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.66 berikut ini adalah *Use Case Description* pada *Use Case* Melihat File Persetujuan SPT BKD dengan Kepala sebagai *Actor*.

Tabel 5.66 Use Case Description Melihat File Persetujuan SPT BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat File Persetujuan SPT BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu persetujuan SPT BKD 2. Actor klik tombol lihat file persetujuan SPT BKD pada file yang ingin dilihat | * 1. Sistem menampilkan halaman Persetujuan SPT BKD   2. Sistem menampilkan file persetujuan SPT yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.67 berikut ini adalah *Use Case Description* pada *Use Case* Mengupload Persetujuan SPT BKD dengan Kepala sebagai *Actor*.

Tabel 5.67 Use Case Description Mengupload Persetujuan SPT BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mengupload Persetujuan SPT BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu persetujuan SPT BKD 2. *Actor klik tombol upload persetujuan SPT* pada disposisi yang ingin di upload 3. Actor menambah file persetujuan SPT yang ingin di upload pada form upload persetujuan SPT 4. Actor klik tombol upload | * 1. Sistem menampilkan halaman persetujuan SPT BKD   2.1 Sistem menampilkan halaman upload Persetujuan SPT BKD  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik upload akan muncul pesan **persetujuan SPT** **berhasil diupload** | |
| *Input* | Persetujuan SPT | *Output*: - |

Tabel 5.68 berikut ini adalah *Use Case Description* pada *Use Case* Setujui SPT BKD dengan Kepala sebagai *Actor*.

Tabel 5.68 Use Case Description Setujui SPT BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Setujui SPT BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu persetujuan SPT BKD 2. *Actor klik tombol setuju SPT* pada SPT yang ingin di setujui | * 1. Sistem menampilkan halaman persetjuan SPT BKD   2.1 Sistem memverifikasi data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil |
| *Alternative Flows* | 2.1 Apabila Actor klik accept akan muncul pesan **SPT berhasil disetujui** | |
| *Input* |  | *Output*: - |

Tabel 5.69 berikut ini adalah *Use Case Description* pada *Use Case* Setujui SPT BKD dengan Kepala sebagai *Actor*.

Tabel 5.69 Use Case Description Tolak SPT BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Tolak SPT BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu persetujuan SPT BKD 2. *Actor klik tombol tolak SPT* pada SPT yang ingin di setujui | * 1. Sistem menampilkan halaman persetujuan SPT BKD   2.1 Sistem memverifikasi data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil |
| *Alternative Flows* | 2.1 Apabila Actor klik accept akan muncul pesan **SPT berhasil ditolak** | |
| *Input* |  | *Output*: - |

Tabel 5.70 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Persetujuan SPT BKA dengan Kepala sebagai *Actor*.

Tabel 5.70 Use Case Description Melihat Persetujuan SPT BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Case* Melihat Persetujuan SPT BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu* persetujuan SPTBKA | * 1. sistem menampilkan halaman persetujuan SPT BKA |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.71 berikut ini adalah *Use Case Description* pada *Use Case* Melihat File SPT BKA dengan Kepala sebagai *Actor*.

Tabel 5.71 Use Case Description Melihat File SPT BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat File SPT BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Persetujuan SPT BKA 2. Actor klik tombol lihat file SPT BKA pada file yang ingin dilihat | * 1. Sistem menampilkan halaman persetujuan SPT BKA   2. Sistem menampilkan file SPT yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.72 berikut ini adalah *Use Case Description* pada *Use Case* Melihat File Persetujuan SPT BKA dengan Kepala sebagai *Actor*.

Tabel 5.72 Use Case Description Melihat File Persetujuan SPT BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat File Persetujuan SPT BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu persetujuan SPT BKA 2. Actor klik tombol lihat file persetujuan SPT BKA pada file yang ingin dilihat | * 1. Sistem menampilkan halaman Persetujuan SPT BKA   2. Sistem menampilkan file persetujuan SPT yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.73 berikut ini adalah *Use Case Description* pada *Use Case* Mengupload Persetujuan SPT BKA dengan Kepala sebagai *Actor*.

Tabel 5.73 Use Case Description Mengupload Persetujuan SPT BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mengupload Persetujuan SPT BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu persetujuan SPT BKA 2. *Actor klik tombol upload persetujuan SPT* pada disposisi yang ingin di upload 3. Actor menambah file persetujuan SPT yang ingin di upload pada form upload persetujuan SPT 4. Actor klik tombol upload | * 1. Sistem menampilkan halaman persetujuan SPT BKA   2.1 Sistem menampilkan halaman upload Persetujuan SPT BKA  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik upload akan muncul pesan **persetujuan SPT** **berhasil diupload** | |
| *Input* | Persetujuan SPT | *Output*: - |

Tabel 5.74 berikut ini adalah *Use Case Description* pada *Use Case* Setujui SPT BKA dengan Kepala sebagai *Actor*.

Tabel 5.74 Use Case Description Setujui SPT BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Setujui SPT BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu persetujuan SPT BKA 2. *Actor klik tombol setuju SPT* pada SPT yang ingin di setujui | * 1. Sistem menampilkan halaman persetjuan SPT BKA   2.1 Sistem memverifikasi data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil |
| *Alternative Flows* | 2.1 Apabila Actor klik accept akan muncul pesan **SPT berhasil disetujui** | |
| *Input* |  | *Output*: - |

Tabel 5.75 berikut ini adalah *Use Case Description* pada *Use Case* Setujui SPT BKA dengan Kepala sebagai *Actor*.

Tabel 5.75 Use Case Description Tolak SPT BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Tolak SPT BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu persetujuan SPT BKA 2. *Actor klik tombol tolak SPT* pada SPT yang ingin di setujui | * 1. Sistem menampilkan halaman persetujuan SPT BKA   2.1 Sistem memverifikasi data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil |
| *Alternative Flows* | 2.1 Apabila Actor klik accept akan muncul pesan **SPT berhasil ditolak** | |
| *Input* |  | *Output*: - |

Tabel 5.76 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Surat BKD dengan Admin BKD sebagai *Actor*.

Tabel 5.76 Use Case Description Melihat Surat BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Case* Melihat Surat BKD | |
| *Actor* | Admin BKD | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu* surat BKD | * 1. sistem menampilkan halaman surat BKD |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.77 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Persetujuan Disposisi BKD dengan Admin BKD sebagai *Actor*.

Tabel 5.77 Use Case Description Melihat Persetujuan Disposisi BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Persetujuan Disposisi BKD | |
| *Actor* | Admin BKD | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu surat BKD 2. Actor klik tombol lihat persetujuan disposisi BKD pada file yang ingin dilihat | * 1. Sistem menampilkan halaman surat BKD   2. Sistem menampilkan file persetujuan disposisi yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.78 berikut ini adalah *Use Case Description* pada *Use Case* Menambah SPT BKD dengan Admin BKD sebagai *Actor*.

Tabel 5.78 Use Case Description Menambah SPT BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Menambah SPT BKD | |
| *Actor* | Admin BKD | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu surat BKD 2. *Actor klik tombol add SPT* 3. Actor menambah data pada form tambah SPT BKD 4. Actor klik tombol add | * 1. Sistem menampilkan halaman surat BKD   2.1 Sistem menampilkan halaman add SPT BKD  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik add akan muncul pesan **SPT****berhasil ditambah** | |
| *Input* | Pengirim, no surat, tanggal surat, ringkasan, tanggal akhir SPT, nama pegawai, nip pegawai, jabatan pegawai | *Output*: - |

Tabel 5.79 berikut ini adalah *Use Case Description* pada *Use Case* Melihat SPT BKD dengan Admin BKD sebagai *Actor*.

Tabel 5.79 Use Case Description Melihat SPT BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat SPT BKD | |
| *Actor* | Admin BKD | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT 2. Actor klik tombol lihat spt BKD pada file yang ingin dilihat | * 1. Sistem menampilkan halaman SPT   2. Sistem menampilkan file spt yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.80 berikut ini adalah *Use Case Description* pada *Use Case* Ajukan SPT BKD dengan Admin BKD sebagai *Actor*.

Tabel 5.80 Use Case Description Ajukan SPT BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Ajukan SPT BKD | |
| *Actor* | Admin BKD | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT 2. *Actor klik tombol ajukan SPT* pada SPT yang ingin di ajukan | * 1. Sistem menampilkan halaman SPT   2.1 Sistem memverifikasi data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil |
| *Alternative Flows* | 2.1 Apabila Actor klik accept akan muncul pesan **SPT berhasil diajukan** | |
| *Input* |  | *Output*: - |

Tabel 5.81 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Surat BKA dengan Admin BKA sebagai *Actor*.

Tabel 5.81 Use Case Description Melihat Surat BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | *Case* Melihat Surat BKA | |
| *Actor* | Admin BKA | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor klik menu* surat BKA | * 1. sistem menampilkan halaman surat BKA |
| *Alternative Flows* | **-** | |
| *Input* | - | *Output*: - |

Tabel 5.82 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Persetujuan Disposisi BKA dengan Admin BKA sebagai *Actor*.

Tabel 5.82 Use Case Description Melihat Persetujuan Disposisi BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Persetujuan Disposisi BKA | |
| *Actor* | Admin BKA | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu surat BKA 2. Actor klik tombol lihat persetujuan disposisi BKA pada file yang ingin dilihat | * 1. Sistem menampilkan halaman surat BKA   2. Sistem menampilkan file persetujuan disposisi yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.83 berikut ini adalah *Use Case Description* pada *Use Case* Menambah SPT BKA dengan Admin BKA sebagai *Actor*.

Tabel 5.83 Use Case Description Menambah SPT BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Menambah SPT BKA | |
| *Actor* | Admin BKA | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu surat BKA 2. *Actor klik tombol add SPT* 3. Actor menambah data pada form tambah SPT BKA 4. Actor klik tombol add | * 1. Sistem menampilkan halaman surat BKA   2.1 Sistem menampilkan halaman add SPT BKA  3.1 Sistem memastikan *form* terisi  4.1 Sistem melakukan validasi form sesuai tipe inputan masing-masing |
| *Alternative Flows* | 3.1 Apabila *form* tidak diisi (dikosongkan) maka akan muncul pemberitahuan **The field is required**.  4.1 Apabila Actor klik add akan muncul pesan **SPT****berhasil ditambah** | |
| *Input* | Pengirim, no surat, tanggal surat, ringkasan, tanggal akhir SPT, nama pegawai, nip pegawai, jabatan pegawai | *Output*: - |

Tabel 5.84 berikut ini adalah *Use Case Description* pada *Use Case* Melihat SPT BKA dengan Admin BKA sebagai *Actor*.

Tabel 5.84 Use Case Description Melihat SPT BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat SPT BKA | |
| *Actor* | Admin BKA | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT 2. Actor klik tombol lihat spt BKA pada file yang ingin dilihat | * 1. Sistem menampilkan halaman SPT   2. Sistem menampilkan file spt yang dipilih dalam bentuk pdf |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.85 berikut ini adalah *Use Case Description* pada *Use Case* Ajukan SPT BKA dengan Admin BKA sebagai *Actor*.

Tabel 5.85 Use Case Description Ajukan SPT BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Ajukan SPT BKA | |
| *Actor* | Admin BKA | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT 2. *Actor klik tombol ajukan SPT* pada SPT yang ingin di ajukan | * 1. Sistem menampilkan halaman SPT   2.1 Sistem memverifikasi data didatabase yang sesuai dengan data yang dipilih dan menampilkan pemberitahuan berhasil |
| *Alternative Flows* | 2.1 Apabila Actor klik accept akan muncul pesan **SPT berhasil diajukan** | |
| *Input* |  | *Output*: - |

Tabel 5.86 berikut ini adalah *Use Case Description* pada *Use Case* Mencari Menu dengan Admin sebagai *Actor*.

Tabel 5.86 Use Case Description Mencari Menu

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mencari Menu | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Menu Management 2. Actor menuliskan keyword yang ingin dicari pada form pencarian 3. Actor klik tombol cari | * 1. Sistem menampilkan halaman Menu Managemet   2.1 Sistem memastikan form terisi  3.1 Sistem mencari data didatabase yang sesuai dengan keyword menampilkan hasilnya di halaman menu management |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.87 berikut ini adalah *Use Case Description* pada *Use Case* Mencari Submenu dengan Admin sebagai *Actor*.

Tabel 5.87 Use Case Description Mencari Submenu

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mencari Submenu | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Submenu Management 2. Actor menuliskan keyword yang ingin dicari pada form pencarian 3. Actor klik tombol cari | * 1. Sistem menampilkan halaman Submenu Managemet   2.1 Sistem memastikan form terisi  3.1 Sistem mencari data didatabase yang sesuai dengan keyword menampilkan hasilnya di halaman Submenu management |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.88 berikut ini adalah *Use Case Description* pada *Use Case* Mencari Pegawai TU dengan Admin sebagai *Actor*.

Tabel 5.88 Use Case Description Mencari Pegawai TU

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mencari Pegawai TU | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu pegawai TU 2. Actor menuliskan keyword yang ingin dicari pada form pencarian 3. Actor klik tombol cari | * 1. Sistem menampilkan halaman pegawai TU   2.1 Sistem memastikan form terisi  3.1 Sistem mencari data didatabase yang sesuai dengan keyword menampilkan hasilnya di halaman pegawai TU |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.89 berikut ini adalah *Use Case Description* pada *Use Case* Mencari Pegawai BKD dengan Admin sebagai *Actor*.

Tabel 5.89 Use Case Description Mencari Pegawai BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mencari Pegawai BKD | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu pegawai BKD 2. Actor menuliskan keyword yang ingin dicari pada form pencarian 3. Actor klik tombol cari | * 1. Sistem menampilkan halaman pegawai BKD   2.1 Sistem memastikan form terisi  3.1 Sistem mencari data didatabase yang sesuai dengan keyword menampilkan hasilnya di halaman pegawai BKD |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.90 berikut ini adalah *Use Case Description* pada *Use Case* Mencari Pegawai BKA dengan Admin sebagai *Actor*.

Tabel 5.90 Use Case Description Mencari Pegawai BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mencari Pegawai BKA | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu pegawai BKA 2. Actor menuliskan keyword yang ingin dicari pada form pencarian 3. Actor klik tombol cari | * 1. Sistem menampilkan halaman pegawai BKA   2.1 Sistem memastikan form terisi  3.1 Sistem mencari data didatabase yang sesuai dengan keyword menampilkan hasilnya di halaman pegawai BKA |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.91 berikut ini adalah *Use Case Description* pada *Use Case* Mencari Pegawai Trash dengan Admin sebagai *Actor*.

Tabel 5.91 Use Case Description Mencari Pegawai Trash

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mencari Pegawai Trash | |
| *Actor* | Admin | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu pegawai trash 2. Actor menuliskan keyword yang ingin dicari pada form pencarian 3. Actor klik tombol cari | * 1. Sistem menampilkan halaman pegawai trash   2.1 Sistem memastikan form terisi  3.1 Sistem mencari data didatabase yang sesuai dengan keyword menampilkan hasilnya di halaman pegawai trash |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.92 berikut ini adalah *Use Case Description* pada *Use Case* Mencari Persetujuan Disposisi dengan Kepala sebagai *Actor*.

Tabel 5.92 Use Case Description Mencari Persetujuan Disposisi

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mencari Persetujuan Disposisi | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu persetujuan disposisi 2. Actor menuliskan keyword yang ingin dicari pada form pencarian 3. Actor klik tombol cari | * 1. Sistem menampilkan halaman persetujuan disposisi   2.1 Sistem memastikan form terisi  3.1 Sistem mencari data didatabase yang sesuai dengan keyword menampilkan hasilnya di halaman persetujuan disposisi |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.93 berikut ini adalah *Use Case Description* pada *Use Case* Mencari Disposisi BKD dengan Kepala sebagai *Actor*.

Tabel 5.93 Use Case Description Mencari Disposisi BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mencari Disposisi BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu disposisi BKD 2. Actor menuliskan keyword yang ingin dicari pada form pencarian 3. Actor klik tombol cari | * 1. Sistem menampilkan halaman disposisi BKD   2.1 Sistem memastikan form terisi  3.1 Sistem mencari data didatabase yang sesuai dengan keyword menampilkan hasilnya di halaman disposisi BKD |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.94 berikut ini adalah *Use Case Description* pada *Use Case* Mencari Disposisi BKA dengan Kepala sebagai *Actor*.

Tabel 5.94 Use Case Description Mencari Disposisi BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mencari Disposisi BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu disposisi BKA 2. Actor menuliskan keyword yang ingin dicari pada form pencarian 3. Actor klik tombol cari | * 1. Sistem menampilkan halaman disposisi BKA   2.1 Sistem memastikan form terisi  3.1 Sistem mencari data didatabase yang sesuai dengan keyword menampilkan hasilnya di halaman disposisi BKA |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.95 berikut ini adalah *Use Case Description* pada *Use Case* Mencari Persetujuan SPT BKD dengan Kepala sebagai *Actor*.

Tabel 5.95 Use Case Description Mencari Persetujuan SPT BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mencari Persetujuan SPT BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu persetujuan SPT BKD 2. Actor menuliskan keyword yang ingin dicari pada form pencarian 3. Actor klik tombol cari | * 1. Sistem menampilkan halaman persetujuan SPT BKD   2.1 Sistem memastikan form terisi  3.1 Sistem mencari data didatabase yang sesuai dengan keyword menampilkan hasilnya di halaman persetujuan SPT BKD |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.96 berikut ini adalah *Use Case Description* pada *Use Case* Mencari Persetujuan SPT BKA dengan Kepala sebagai *Actor*.

Tabel 5.96 Use Case Description Mencari Persetujuan SPT BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mencari Persetujuan SPT BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu persetujuan SPT BKA 2. Actor menuliskan keyword yang ingin dicari pada form pencarian 3. Actor klik tombol cari | * 1. Sistem menampilkan halaman persetujuan SPT BKA   2.1 Sistem memastikan form terisi  3.1 Sistem mencari data didatabase yang sesuai dengan keyword menampilkan hasilnya di halaman persetujuan SPT BKA |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.97 berikut ini adalah *Use Case Description* pada *Use Case* Mencari Surat BKD dengan Admin BKD sebagai *Actor*.

Tabel 5.97 Use Case Description Mencari Surat BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mencari Surat BKD | |
| *Actor* | Admin BKD | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Surat BKD 2. Actor menuliskan keyword yang ingin dicari pada form pencarian 3. Actor klik tombol cari | * 1. Sistem menampilkan halaman Surat BKD   2.1 Sistem memastikan form terisi  3.1 Sistem mencari data didatabase yang sesuai dengan keyword menampilkan hasilnya di halaman Surat BKD |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.98 berikut ini adalah *Use Case Description* pada *Use Case* Mencari SPT BKD dengan Admin BKD sebagai *Actor*.

Tabel 5.98 Use Case Description Mencari Surat BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mencari SPT BKD | |
| *Actor* | Admin BKD | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT BKD 2. Actor menuliskan keyword yang ingin dicari pada form pencarian 3. Actor klik tombol cari | * 1. Sistem menampilkan halaman SPT BKD   2.1 Sistem memastikan form terisi  3.1 Sistem mencari data didatabase yang sesuai dengan keyword menampilkan hasilnya di halaman SPT BKD |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.99 berikut ini adalah *Use Case Description* pada *Use Case* Mencari Surat BKA dengan Admin BKA sebagai *Actor*.

Tabel 5.97 Use Case Description Mencari Surat BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mencari Surat BKA | |
| *Actor* | Admin BKA | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Surat BKA 2. Actor menuliskan keyword yang ingin dicari pada form pencarian 3. Actor klik tombol cari | * 1. Sistem menampilkan halaman Surat BKA   2.1 Sistem memastikan form terisi  3.1 Sistem mencari data didatabase yang sesuai dengan keyword menampilkan hasilnya di halaman Surat BKA |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.100 berikut ini adalah *Use Case Description* pada *Use Case* Mencari SPT BKA dengan Admin BKA sebagai *Actor*.

Tabel 5.100 Use Case Description Mencari Surat BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Mencari SPT BKA | |
| *Actor* | Admin BKA | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT BKA 2. Actor menuliskan keyword yang ingin dicari pada form pencarian 3. Actor klik tombol cari | * 1. Sistem menampilkan halaman SPT BKA   2.1 Sistem memastikan form terisi  3.1 Sistem mencari data didatabase yang sesuai dengan keyword menampilkan hasilnya di halaman SPT BKA |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.100 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Pemberitahuan Surat Masuk dengan Admin TU sebagai *Actor*.

Tabel 5.100 Use Case Description Melihat Pemberitahuan Surat Masuk

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Pemberitahuan Surat Masuk | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Surat Masuk 2. *Actor* klik tombol pemberitahuan Surat Masuk | * 1. Sistem menampilkan halaman Surat Masuk   2.1 Sistem menampilkan modal yang berisi data surat masuk yang belum di disposisi. |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.100 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Pemberitahuan SPT BKD dengan Admin TU sebagai *Actor*.

Tabel 5.100 Use Case Description Melihat Pemberitahuan SPT BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Pemberitahuan SPT BKD | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT BKD 2. *Actor* klik tombol pemberitahuan SPT BKD | * 1. Sistem menampilkan halaman SPT BKD   2.1 Sistem menampilkan modal yang berisi data SPT yang belum di upload SPT lengkapnya |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.100 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Pemberitahuan SPT BKA dengan Admin TU sebagai *Actor*.

Tabel 5.100 Use Case Description Melihat Pemberitahuan SPT BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Pemberitahuan SPT BKA | |
| *Actor* | Admin TU | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT BKA 2. *Actor* klik tombol pemberitahuan SPT BKA | * 1. Sistem menampilkan halaman SPT BKA   2.1 Sistem menampilkan modal yang berisi data SPT yang belum di upload SPT lengkapnya |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.100 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Pemberitahuan Surat Disposisi dengan Kepala sebagai *Actor*.

Tabel 5.100 Use Case Description Melihat Pemberitahuan Surat Disposisi

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Pemberitahuan Surat Disposisi | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Persetujuan Disposisi 2. *Actor* klik tombol pemberitahuan Persetujuan Disposisi | * 1. Sistem menampilkan halaman Persetujuan Disposisi   2.1 Sistem menampilkan modal yang berisi data surat masuk yang belum di disposisi |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.100 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Pemberitahuan Surat Disposisi BKD dengan Kepala sebagai *Actor*.

Tabel 5.100 Use Case Description Melihat Pemberitahuan Surat Disposisi BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Pemberitahuan Surat Disposisi BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Disposisi BKD 2. *Actor* klik tombol pemberitahuan Disposisi BKD | * 1. Sistem menampilkan halaman Disposisi BKD   2.1 Sistem menampilkan modal yang berisi data surat disposisi yang belum di upload persetujuan disposisinya |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.100 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Pemberitahuan Surat Disposisi BKA dengan Kepala sebagai *Actor*.

Tabel 5.100 Use Case Description Melihat Pemberitahuan Surat Disposisi BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Pemberitahuan Surat Disposisi BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Disposisi BKA 2. *Actor* klik tombol pemberitahuan Disposisi BKA | * 1. Sistem menampilkan halaman Disposisi BKA   2.1 Sistem menampilkan modal yang berisi data surat disposisi yang belum di upload persetujuan disposisinya |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.100 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Persetujuan SPT BKD dengan Kepala sebagai *Actor*.

Tabel 5.100 Use Case Description Melihat Persetujuan SPT BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Persetujuan SPT BKD | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Persetujuan SPT BKD 2. *Actor* klik tombol pemberitahuan Persetujuan SPT BKD | * 1. Sistem menampilkan halaman Persetujuan SPT BKD   2.1 Sistem menampilkan modal yang berisi data SPT yang belum di upload persetujuan SPTnya |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.100 berikut ini adalah *Use Case Description* pada *Use Case* Melihat Persetujuan SPT BKA dengan Kepala sebagai *Actor*.

Tabel 5.100 Use Case Description Melihat Persetujuan SPT BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat Persetujuan SPT BKA | |
| *Actor* | Kepala | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu Persetujuan SPT BKA 2. *Actor* klik tombol pemberitahuan Persetujuan SPT BKA | * 1. Sistem menampilkan halaman Persetujuan SPT BKA   2.1 Sistem menampilkan modal yang berisi data SPT yang belum di upload persetujuan SPTnya |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.100 berikut ini adalah *Use Case Description* pada *Use Case* Melihat SPT BKD dengan Admin TU sebagai *Actor*.

Tabel 5.100 Use Case Description Melihat SPT BKD

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat SPT BKD | |
| *Actor* | Admin BKD | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT BKD 2. *Actor* klik tombol pemberitahuan SPT BKD | * 1. Sistem menampilkan halaman SPT BKD   2.1 Sistem menampilkan modal yang berisi data Surat BKD yang belum dibuat SPTnya |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.100 berikut ini adalah *Use Case Description* pada *Use Case* Melihat SPT BKD dengan Admin TU sebagai *Actor*.

Tabel 5.100 Use Case Description Melihat SPT

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat SPT | |
| *Actor* | Admin BKD | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT 2. *Actor* klik tombol pemberitahuan SPT | * 1. Sistem menampilkan halaman SPT   2.1 Sistem menampilkan modal yang berisi data SPT yang belum diajukan |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.100 berikut ini adalah *Use Case Description* pada *Use Case* Melihat SPT BKA dengan Admin BKA sebagai *Actor*.

Tabel 5.100 Use Case Description Melihat SPT BKA

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat SPT BKA | |
| *Actor* | Admin BKA | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT BKA 2. *Actor* klik tombol pemberitahuan SPT BKA | * 1. Sistem menampilkan halaman SPT BKA   2.1 Sistem menampilkan modal yang berisi data Surat BKA yang belum dibuat SPTnya |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |

Tabel 5.100 berikut ini adalah *Use Case Description* pada *Use Case* Melihat SPT BKA dengan Admin BKA sebagai *Actor*.

Tabel 5.100 Use Case Description Melihat SPT

|  |  |  |
| --- | --- | --- |
| *Use Case Name* | Melihat SPT | |
| *Actor* | Admin BKA | |
| *Precondition* | Actor sudah login | |
| *Flows of Events* | *Actor Action* | *System Response* |
| 1. *Actor* klik menu SPT 2. *Actor* klik tombol pemberitahuan SPT | * 1. Sistem menampilkan halaman SPT   2.1 Sistem menampilkan modal yang berisi data SPT yang belum diajukan |
| *Alternative Flows* | **-** | |
| *Input* |  | *Output*: - |