Spesifikasi Kebutuhan Perangkat Lunak

untuk

Aplikasi *e-Filing* Pada PT Al-Amanah

Versi 1.0

Disiapkan oleh

Kelompok 4

Parera Jumano - 181022000012

Rachmat Rizal – 181022000003

Arif Rahman Andika - 181022000006

Januar Ilham - 182022000002

Danil Satria - 181022000011

20 April 2019

STMIK Eresha

Table of Contents

1. Pendahuluan 1

1.1 Tujuan 1

1.2 Audien yang Dituju 1

1.3 Ruang Lingkup 1

1.4 Definisi dan Istilah 1

1.5 Refrensi 1

2. Deskripsi Keseluruhan 2

2.1 Deskripsi Produk 2

2.2 Fungsi Produk 2

2.3 Penggolongan Karakterik Pengguna 2

2.4 Lingkungan Operasi 2

2.5 Batasan Desain dan Implementasi 2

2.6 Dokumentasi Pengguna 3

3. Kebutuhan Antarmuka Eksternal 4

3.1 Antarmuka Pengguna 4

3.2 Antarmuka Perangkat Keras 4

3.3 Antarmuka Perangkat Lunak 4

3.4 Antarmuka Komunikasi 4

4. Kebutuhan Fungsional 5

4.1 Pendefinisian Aktor 5

4.2 Use Case Admin Diagram 5

4.3 Use Case User Diagram 5

4.4 Pendefinisian Use Case 5

4.5 Use Case Scenario 5

4.5.1 Use Case Scenario Login 5

4.5.2 Use Case Scenario Kelola Data Divisi 5

4.5.3 Use Case Scenario Kelola Data Pengguna 5

4.5.4 Use Case Scenario Kelola Data Log 5

4.5.5 Use Case Scenario Kelola Data Kategori 5

4.5.6 Use Case Scenario Kelola Data Agenda Masuk 5

4.5.7 Use Case Scenario Kelola Data Agenda Keluar 5

4.5.8 Use Case Scenario Kelola Password 5

4.5.9 Use Case Scenario Logout 5

4.6 Activity Diagram 5

4.6.1 Activity Diagram Login 5

4.6.2 Activity Diagram Kelola Data Divisi 5

4.6.3 Activity Diagram Kelola Data Pengguna 5

4.6.4 Activity Diagram Kelola Data Log 5

4.6.5 Activity Diagram Kelola Data Kategori 5

4.6.6 Activity Diagram Kelola Data Agenda Masuk 5

4.6.7 Activity Diagram Kelola Data Agenda Keluar 5

4.6.8 Activity Diagram Kelola Password 5

4.6.9 Activity Diagram Logout 5

5. Kebutuhan Non-Fungsional 14

5.1 Kebutuhan Kinerja 14

5.2 Kebutuhan Keamanan 14

5.3 Kebutuhan Keamana Perlindungan 14

6. Kebutuhan Lain 15

7. Appendix A: Daftar Istilah 16

Riwayat Revisi

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama** | **Tanggal** | **Alasan Perubahan** | **Versi** |
| eFiling | 20-04-2019 |  | 1.0 |
|  |  |  |  |
|  |  |  |  |

# Pendahuluan

## Tujuan

Tujuan pembuatan software ini adalah untuk membantu pengguna dalam mengelola dokumen secara elektronik. Selama ini pengelolaan dokumen dilakukan secara manual di tempat penyimpanan lemari/rak yang disediakan, tetapi dengan semakin banyaknya dokumen, maka tempat penyimpanan yang dibutuhkan akan semakin banyak. Disisi lain proses pencarian kembali dokumen memerlukan waktu yang cukup lama. Dengan software ini, dokumen akan disimpan secara elektronik. Setiap dokumen dibuatkan narasinya dengan mengisi form yang disediakan, guna mempermudah disaat pencarian. Dokumen asli di scan, kemudian file hasil scan nya di upload sebagai lampiran.

Dalam dokumen ini akan disajikan deskripsi dari perangkat lunak e-filing secara lengkap yaitu mulai dari tujuan, fitur, antarmuka, kebutuhan perangkat dan lain sebagainya.

## Audien yang Dituju

Dokumen ini ditujukan untuk:

* **Pengembang aplikasi**, yaitu programmer dan tim pengembang yang tertarik terhadap aplikasi ini, dan ingin mengembangkan lebih lanjut, dan atau ingin memperbaiki kesalahan aplikasi yang sudah ada.
* **Pengguna**, yaitu sekretaris, admin, dan pengguna lainnya yang ingin menggunakan aplikasi *e-filing* untuk membantu pekerjaannya.
* **Pihak lain** yang ingin berkontribusi terhadap aplikasi ini.

## Ruang Lingkup

*e-Filing* adalah sistem/aplikasi berbasis web yang digunakan untuk membantu mengelola dokumen secara elektronik. Aplikasi ini memiliki beberapa fasilitas yaitu:

1. Aplikasi dapat menampilkan daftar dokumen sesuai kategori tertentu.
2. Aplikasi dapat menyimpan narasi dokumen.
3. Aplikasi dapat menyimpan dokumen elektronik (file hasil scan dokumen).
4. Aplikasi dapat melakukan pencarian secara cepat terhadap dokumen tertentu.
5. Aplikasi dapat memberi peringatan melalui email.
6. Aplikasi dapat mencetak daftar dokumen.

## Definisi dan Istilah

Berikut adalah definisi dan istilah yang digunakan dalam dokumen ini:

* *Server* adalah sebuah sistem komputer yang menyediakan jenis layanan (service) tertentu dalam sebuah jaringan computer
* *Windows* adalah sistem operasi yang dikembangkan oleh Microsoft, dengan menggunakan antarmuka pengguna grafis.
* *IIS* *Internet Information Services* atau *Internet Information Server* adalah sebuah HTTP web server yang digunakan dalam sistem operasi server Windows
* *ASP.Net* Active Server Pages .NET adalah kumpulan teknologi dalam Framework .NET untuk membangun aplikasi web dinamis dan XML Web Service (Layanan Web XML)
* *.NET Framework* merupakan sebuah perangkat lunak kerangka kerja yang menyediakan sejumlah besar pustaka pemrograman komputer dan mendukung beberapa bahasa pemrograman serta interoperabilitas yang baik sehingga memungkinkan bahasa-bahasa tersebut berfungsi satu dengan lain dalam pengembangan system.
* *Browser* adalah software/aplikasi/perangkat lunak yang digunakan untuk mengakses/ menampilkan halaman web. Contohnya adalah Microsoft Internet Explorer, Google Chrome, Apple Safari dan Opera, Netscape Navigator, Mozilla Firefox.
* *SRS* *Software Requirements Specification*, atau Spesifikasi Kebutuhan Perangkat Lunak (SKPL)
* *IEEE* *Institute of Electrical and Electronics Engineering atau* Standar internasional untuk pengembangan dan perancangan produk.

## Referensi

Berikut adalah daftar acuan yang digunakan dalam pendokumentasian spesifikasi kebutuhan perangkat lunak ini.

* IEEE Template for System Requirement Specification Documents:

<https://goo.gl/nsUFwy>

* e-filing

<https://github.com/ocopflame/Kelompok_IV_Software_Engineering_ERESHA>

* Source Code e-Filing

https://github.com/ocopflame/Kelompok\_IV\_Software\_Engineering\_ERESHA/tree/master/source\_code

* Aplikasi e-Filing ter compile

https://github.com/ocopflame/Kelompok\_IV\_Software\_Engineering\_ERESHA/tree/master/compiled

* Dokumentasi e-Filing

<https://github.com/ocopflame/Kelompok_IV_Software_Engineering_ERESHA/tree/master/docs>

* Demo Aplikasi e-Filing

<http://intra.pertamedika.co.id/tugas>

admin : admin

user : user

password : password

# Deskripsi Keseluruhan

## Deskripsi Produk

e-Filing adalah sistem/aplikasi berbasis web yang digunakan untuk membantu mengelola dokumen secara elektronik. Efiling bisa di akses melalui browser yang sudah mendukung teknologi terkini. E-Filing dibagi menjadi 2 (dua) antarmuka, yaitu antarmuka admin dan antarmuka pengguna.

e-Filing dikembangkan oleh Kelompok 4 S2 Magister Komputer STMIK Eresha Angkatan 2018/2019 sebagai bagian dari tugas mata kuliah Software Engineering.

## Fungsi Produk

Fungsi yang terdapat pada aplikasi e-filing antara lain:

1. Masuk aplikasi (*login*)
2. Menampilkan daftar divisi
3. Mengelola data divisi
4. Menampilkan daftar pengguna
5. Mengelola data pengguna aplikasi
6. Melihat log data penggunaan aplikasi
7. Menampilkan daftar kategori
8. Mengelola data kategori
9. Menampilkan/mencari daftar agenda masuk
10. Mengelola data agenda masuk
11. Upload file lampiran agenda masuk
12. Mengunduh file lampiran agenda masuk
13. Menampilkan/mencari daftar agenda keluar
14. Mengelola data agenda keluar
15. Upload file lampiran agenda keluar
16. Mengunduh file lampiran agenda keluar
17. Ubah password pengguna aplikasi.
18. Keluar aplikasi (*logout*)

## Penggolongan Karakterik Pengguna

Dalam aplikasi e-Filing ini, pengguna aplikasi dapat digolongkan sebagai berikut:

**2.3.1 Admin**

Admin bertugas/berfungsi mengelola aplikasi. Admin memiliki hak akses sebagai berikut:

* Mengelola (lihat/tambah/ubah) data divisi.
* Mengelola (lihat/tambah/ubah) data pengguna,
* Melihat log data penggunaan aplikasi,
* Mengelola (lihat/tambah/ubah) data kategori,
* Mengubah password miliknya.

1. ***User***

*User* bertugas menginput data dokumen. *User* memiliki hak akses sebagai berikut:

* Mengelola(lihat/cari/tambah/ubah/hapus/cetak) data agenda masuk.
* Mengelola(lihat/cari/tambah/ubah/hapus/cetak) data agenda keluar.
* Mengubah password miliknya.

## Lingkungan Operasi

Aplikas e-Filing adalah aplikasi berbasis web yang berjalan di lingkungan operasi berikut:

1. Server
   1. Kebutuhan Hardware
      1. CPU Intel Dual Xeon Bit, RAM 4 GB, Hardisk 500 GB
      2. Monitor
      3. Keyboard
      4. Mouse
      5. Storage HDD minimal 250 GB
      6. Network
   2. Kebutuhan Software
      1. Sistem operasi Windows Server 2008, 2012, 2016, 2019
      2. Web server IIS
      3. Database MySQL
      4. .Net Framework 4.0
2. Client
   1. Kebutuhan Hardware
3. CPU Intel Dual Core, RAM 4 GB
4. Monitor LED Wide Screen
5. Keyboard + Mouse
6. Printer
7. Mouse
8. Network
   1. Kebutuhan Software
      1. Sistem operasi Windows XP, Windows 7, Windows 8, Windows 10, Linux, MacOS
      2. Browser Mozilla Firefox, Google Chrome, atau browser modern lainnya.

## Batasan Desain dan Implementasi

e-Filing di kembangkan dengan batasan sebagai berikut:

* Bahasa pemrograman yang digunakan adalah Visual Basic .Net (ASP .Net Webform) di atas platform Microsoft .net framework 4.0
* Dikembangkan menggunakan IDE software Visual Studio 2010.
* Menggunakan Library Bootstrap 3, Jquery, Material Design dan HTML 5.
* Database menggunakan MySQL dengan konektivitas ODBC
* Web Server menggunakan IIS
* Aplikasi di desain per modul sesuai dengan kebutuhan pengguna, sehingga mudah untuk digunakan.
* Aplikasi membutuhkan media penyimpanan yang cukup besar pada server, karena dokumen elektronik akan di upload ke server, dan akan memakan banyak space pada media penyimpanan.

## Dokumentasi Pengguna

Dokumentasi pengguna system/aplikasi e-Filing dibagi menjadi 2 (dua):

1. **Panduan pengoperasian aplikasi**, meliputi alur proses dan cara pengoperasian seluruh modul dan fitur yang ada dalam aplikasi.

[https://github.com/ocopflame/Kelompok\_IV\_Software\_Engineering\_ERESHA/tree/master/docs/panduan penggunaan.pdf](https://github.com/ocopflame/Kelompok_IV_Software_Engineering_ERESHA/tree/master/docs/panduan%20penggunaan.pdf)

1. **Panduan teknis**, meliputi aspek pengembangan aplikasi dan implementasi yang berdiperuntukan bagi tim pengembang aplikasi.

[https://github.com/ocopflame/Kelompok\_IV\_Software\_Engineering\_ERESHA/tree/master/docs/panduan teknis.pdf](https://github.com/ocopflame/Kelompok_IV_Software_Engineering_ERESHA/tree/master/docs/panduan%20teknis.pdf)

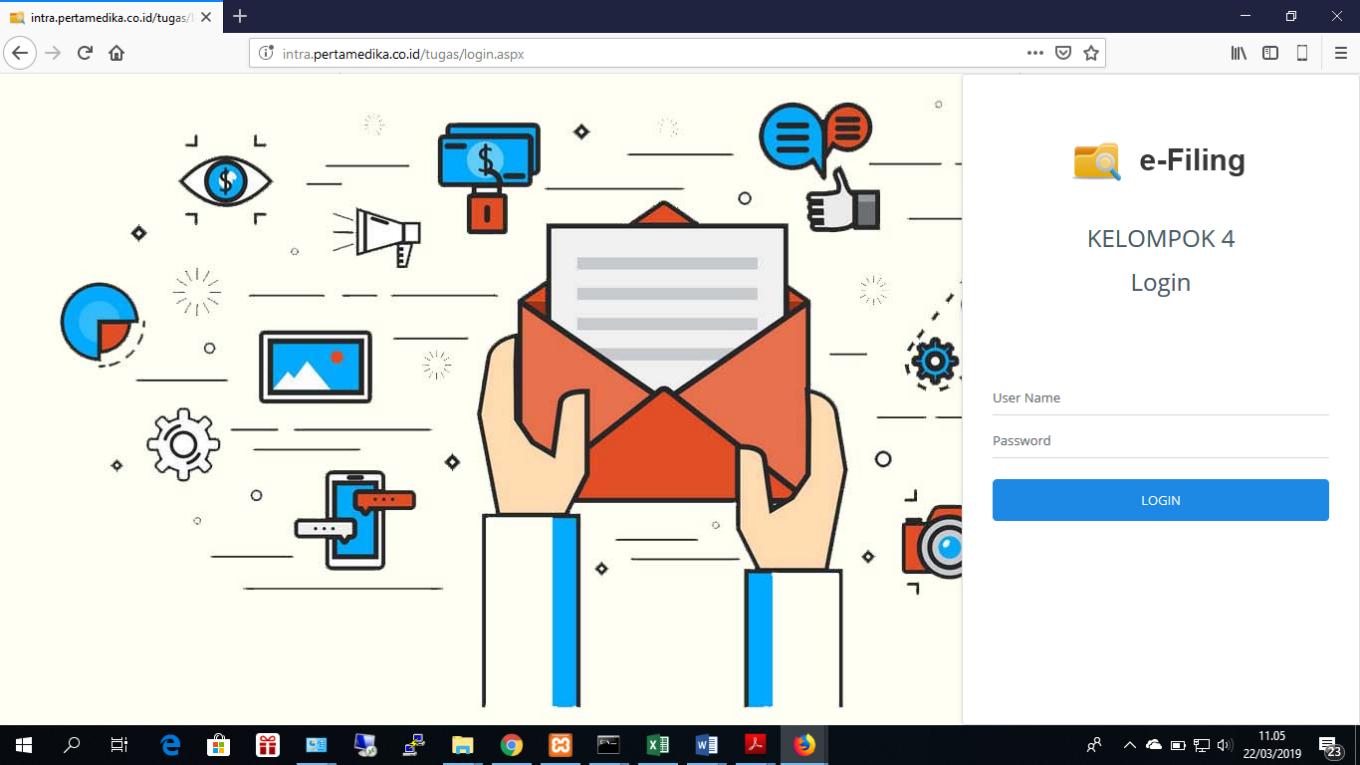
# Kebutuhan Antarmuka Eksternal

## Kebutuhan antar muka eksternal pada perangkat lunak Sistem Informasi e-Filing meliputi kebutuhan antarmuka pengguna, antarmuka perangkat keras, antarmuka perangkat lunak, antarmuka komunikasi.

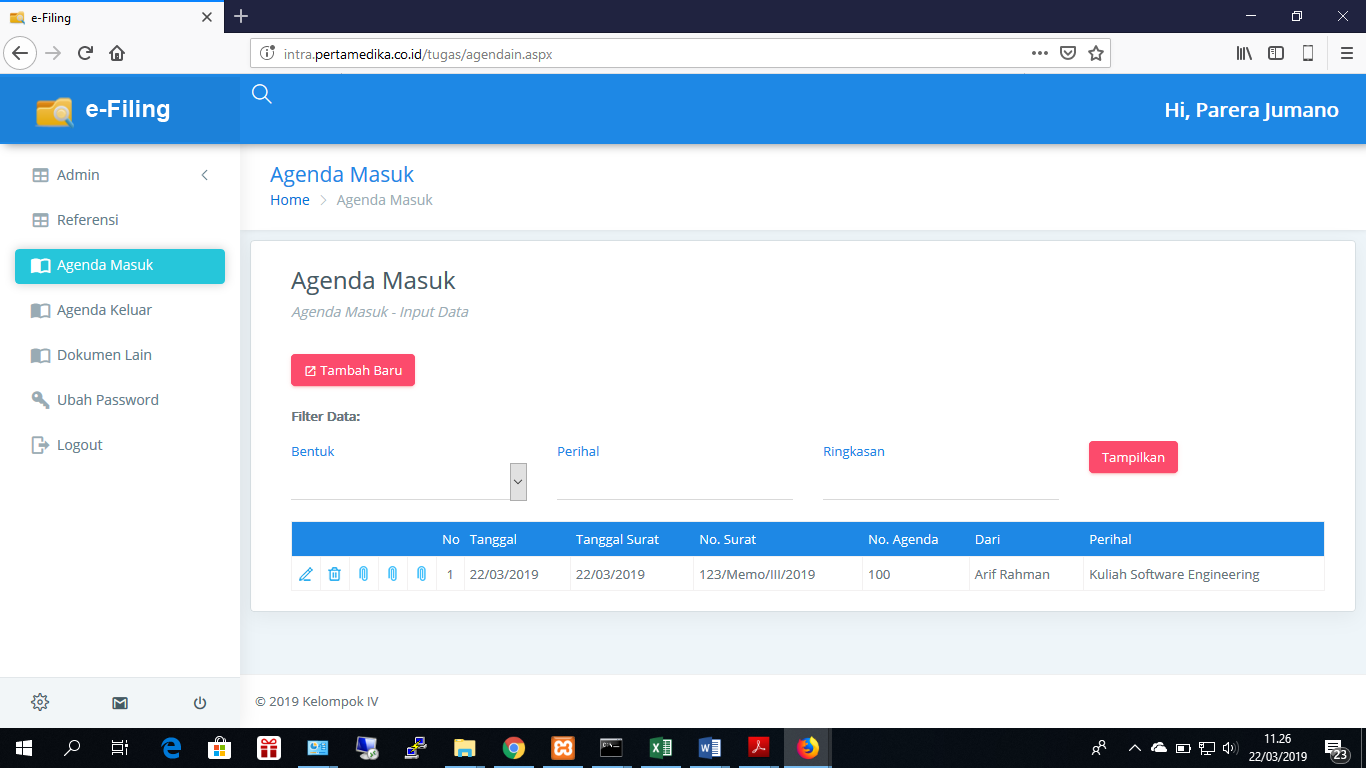
## Antarmuka Pengguna

Perangkat lunak untuk e-Filling ini dibuat dengan menggunakan Aplikasi web, untuk pengolahan User Interface digunakan aplikasi Visual Basic .Net (ASP .Net Webform) di atas platform Microsoft .net framework 4.0. Dikembangkan menggunakan IDE software Visual Studio 2010. Dimana tampilan web didesain menggunakan template yang ada. Perangkat lunak untuk layanan dalam E-filling ini dilengkapi dengan menu untuk pengaksesan berbagai fungsi yang disediakan. Interaksi antara pengguna dan perangkat lunak dilakukan dengan menggunakan keyboard dan mouse. Ada beberapa fungsi yang hanya bisa dilakukan dengan mouse dan ada yang bisa dilakukan baik dengan keyboard dan mouse (misalnya pengaksesan menu).

Aplikasi e-Filing adalah aplikasi web yang dikembangkan dengan responsive css, sehingga tampilannya dapat menyesuaikan dengan perangkat PC maupun perangkat mobil. Berikut adalah tampilan dari aplikasi e-filing di perangkat PC dengan resolusi monitor 1366x768.

1. Halaman Login

Isian login pengguna berada di bagan kanan layar, pengguna di haruskan menginput User ID dan password yang telah diberikan, lalu menekan tombol login untuk masuk ke aplikasi.

1. Halaman Data

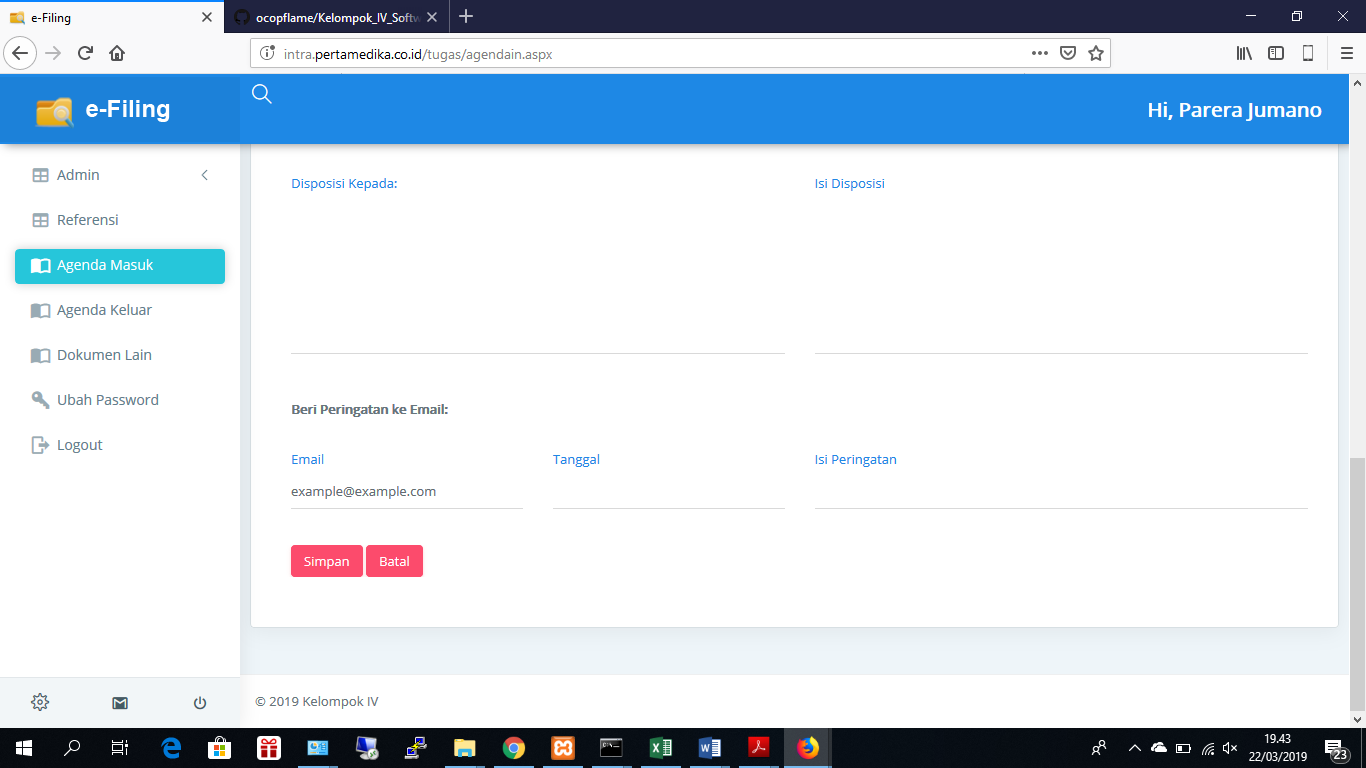
Pada aplikasi e-filing standar layout halaman adalah sebagai berikut: menu utama aplikasi berada di sebelah kiri layar, sebelah kanan adalah halaman yang sedang dibuka.

Didalam Aplikasi E-filling ini terdapat beberapa menu yang perlu diketahui diantarnya :

1. Admin : Menu Admin hanya bisa di akses oleh pengguna aplikasi yang level otorisasi nya adalah administrator. Menu admin terdiri dari 3 (tiga) sub menu yaitu:
   * + 1. Divisi: menu ini digunakan untuk melihat/mengelola data divisi.
       2. Pengguna: menu ini digunakan untuk melihat/mengelola data pengguna aplikasi.
       3. Log: menu ini digunakan untuk melihat log data penggunaan aplikasi.
2. Kategori : Menu Kategori hanya bisa di akses oleh pengguna aplikasi yang level otorisasi nya adalah administrator. menu kategori terdiri dari 3 (tiga) sub menu:
   * + - 1. Jenis dokumen: menu ini digunakan untuk melihat/mengeola data jenis dokumen.
         2. Tingkat dokumen: menu ini digunakan untuk melihat/mengelola tingkat dokumen.
         3. Status dokumen: menu ini digunakan untuk melihat/mengelola status dokumen.
3. Agenda masuk : Menu Agenda Masuk hanya bisa di akses oleh pengguna aplikasi yang level otorisasi nya adalah user. menu ini digunakan untuk mencari, melihat dan mengelola data Agenda Masuk. Data Agenda Masuk terdiri dari field-field yaitu: tanggal terima, tanggal surat, nomor surat, nomor agenda, dari (pengirim), perihal, ringkasan isi, kategori, serta disposisi.
4. Agenda keluar : Menu Agenda Keluar hanya bisa di akses oleh pengguna aplikasi yang level otorisasi nya adalah user. menu ini digunakan untuk mencari, melihat dan mengelola data Agenda Keluar. Data Agenda Keluar terdiri dari field-field yaitu: tanggal kirim, tanggal surat, nomor surat, nomor agenda, kepada (tujuan) perihal, serta ringkasan isi.
5. Ubah password : Menu bisa di akses oleh semua pengguna aplikasi. Menu ini digunakan untuk melakukan pergantian password secara berkala guna mencegah sesuatu hal yang tidak di inginkan
6. Logout : Menu bisa di akses oleh semua pengguna aplikasi. Menu ini digunakan untuk keluar dari aplikasi

Ketika satu halaman dibuka, utamanya merupakan sarana pencarian data dengan filter yang sudah disediakan, dilanjutkan dengan menekan tombol tampilkan, maka akan tampil daftar data yang dicari. Pada tabel daftar data yang ditampilkan terdapat tombol ubah dan hapus untuk mengelola data yang sudah ada. Untuk menambahkan data baru, di bagian halaman atas terdapat tombol tambah data, yang jika di tekan akan muncul form input data baru.

1. Halaman Input Data



Form Input Data berisi template narasi dari suatu data, field isian terdiri dari 2 (dua) bagian yaitu mandatory (wajib di isi) dan non mandatory (tidak wajib di isi). Jika form sudah di isi lengkap, dibagian bawah halaman terdapat tombol simpan untuk menyimpan data, atau tombol batal untuk membatalkan proses penginputan data.

## Antar muka Perangkat Keras

Aplikasi ini berjalan di atas perangkat keras berupa beberapa komputer yang saling terhubung oleh jaringan internet dan berkomunikasi dengan protocol http. Dimana file aplikasi ditempatkan pada webhosting yang dioperasikan oleh Administrator.

Namun untuk menjalankan aplikasi ini diperlukan 2 jenis perangkat keras, yaitu :

1. Komputer server, adalah komputer yang berfungsi sebagai komputer penyimpan, pemroses, penyedia, dan pendistribusian keseluruhan data dan informasi.
2. Komputer client, adalah komputer yang digunakan untuk mengakses layanan yang disediakan oleh komputer server.

Antarmuka perangkat keras yang digunakan dalam perangkat lunak ini adalah :

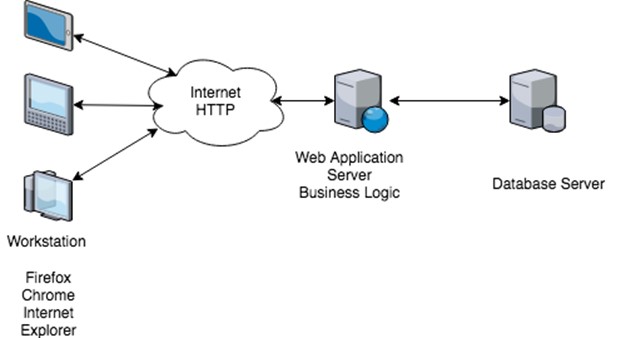
1. Sistem Informasi ini berjalan di komputer server. Semua komputer yang digunakan untuk server harus saling terhubung dalam suatu jaringan misalnya LAN, WAN, Intranet dan Internet.
2. Selain digunakan untuk mengakses data dan informasi dari komputer server, komputer client juga dapat digunakan untuk komputer kerja diluar layanan yang ditawarkan oleh server

Adapun spesifikasi perangkat keras yang dibutuhkan adalah:

* 1. PC
  2. Monitor VGA mempunyai resolusi minimal 1366 x 768 pixel.
  3. Keyboard dan mouse untuk melakukan kegiatan user.
  4. Semua perangkat keras yang digunakan merupakan perangkat standar dalam sistem komputer

## Antarmuka Komunikasi

Jaringan komunikasi dalam aplikasi ini menggunakan jaringan internet untuk dapat saling berkomunikasi. Protokol yang digunakan untuk aplikasi ini adalah HTTP (Hyper Text Transfer Protocol). Berikut merupakan gambaran antarmuka komunikasi yang digunakan :



Gambar-3.3.1: Antarmuka komunikasi

# Kebutuhan Fungsional

## Pendefinisian Aktor

|  |  |  |
| --- | --- | --- |
| No | Aktor | Deskripsi |
| 1 | Admin | Admin adalah pengguna yang memiliki otorisasi Administrator. Petugas admin adalah staf sekretaris yang diberikan kewenangan untuk mengelola tugas admin aplikasi yaitu:   * mengelola divisi, * mengelola pengguna, * melihat log, * mengelola kategori, dan * mengelola password. |
| 3 | User | User adalah pengguna yang memiliki otorisasi User. User merupakan pengguna aplikasi dari masing-masing Divisi yang mempunyai kewenangan berikut:  - mengelola agenda masuk,  - mengelola agenda keluar, dan  - mengelola password. |

## Use Case Diagram

### Use Case Admin Diagram



### Use Case User Diagram

## Pendefinisan Use Case

|  |  |
| --- | --- |
| Use Case | Use Case Description |
| **Login** | |
| 1. Login | Proses masuk ke aplikasi |
| **Kelola Data Divisi** | |
| 2. Lihat data Divisi | Proses menampilkan data divisi |
| 3. Tambah data Divisi | Proses mengisi data baru divisi pada aplikasi |
| 4. Ubah data Divisi | Proses memperbarui data divisi pada aplikasi |
| **Kelola Data Pengguna** | |
| 5. Lihat data Pengguna | Proses menampilkan data pengguna |
| 6. Tambah data Pengguna | Proses mengisi data pengguna baru pada aplikasi |
| 7. Ubah data Pengguna | Proses memperbarui data pengguna pada aplikasi |
| **Kelola Log Data** |  |
| 8. Melihat Log Data | Proses menampilkan histori penggunaan |
| **Kelola Data Kategori** | |
| 9. Lihat data Kategori | Proses menampilkan data kategori |
| 10. Tambah data Kategori | Proses mengisi data baru kategori pada aplikasi |
| 11. Ubah data Kategori | Proses meperbarui data kategori pada aplikasi |
| **Kelola Agenda Masuk** | |
| 12. Lihat data Agenda Masuk | Proses menampilkan data Agenda Masuk |
| 13. Cari data Agenda Masuk | Proses menampilkan data Agenda Masuk berdasarkan filter yang ditentukan |
| 14. Tambah data Agenda Masuk | Proses mengisi data baru Agenda Masuk pada aplikasi |
| 15. Ubah data Agenda Masuk | Proses memperbarui data Agenda Masuk pada aplikasi |
| 16. Hapus data Agenda Masuk | Proses menghapus data Agenda Masuk pada aplikasi |
| **Kelola Agenda Keluar** | |
| 17. Lihat data Agenda Keluar | Proses menampilkan data Agenda Keluar |
| 18. Cari data Agenda Masuk | Proses menampilkan data Agenda Keluar berdasarkan filter yang ditentukan |
| 19. Tambah data Agenda Keluar | Proses mengisi data baru Agenda Keluar pada aplikasi |
| 20. Ubah data Agenda Keluar | Proses memperbarui data Agenda Keluar pada aplikasi |
| 21. Hapus data Agenda Keluar | Proses menghapus data Agenda Keluar pada aplikasi |
| **Kelola Password** | |
| 22. Ubah Password | Proses memperbarui password dan menyimpannya ke database |
| **Logout** | |
| 23. Logout | Proses keluar dari aplikasi |

**4.4 Use Case Scenario**

**4.4.1 Use Case Login**

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 1 | |
| *Use Case Name* | Login | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan login pengguna untuk dapat masuk dan menggunakan aplikasi e-Filing. | |
| *Actor* | Admin, User masing-masing divisi | |
| *Pre Condition* | Admin / User sudah berhasil membuka alamat web aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna mengakses alamat aplikasi e-filing |  |
|  | 1. Aplikasi menampilkan form login |
| 1. Pengguna mengisi User ID dan Password. |  |
| 1. Pengguna menekan tombol Login |  |
|  | 1. Aplikasi melakukan validasi User ID dan Password. |
|  | 1. Jika data login valid, aplikasi menampilkan halaman utama aplikasi. |
|  | 1. Aplikasi menyimpan history login |
|  | 1. Aplikasi menampilkan Menu Utama |
| *Error flow of Event* |  | 5a. login tidak valid |
|  | 6a. Aplikasi menampilkan error message bahwa data login tidak valid. |
| *Post Condition* | Aplikasi menyimpan data login dan menampilkan Menu Utama | |

**4.4.2 Use Case Kelola Data Divisi**

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 2 | |
| *Use Case Name* | Lihat Data Divisi | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan melihat data Divisi | |
| *Actor* | Admin | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Admin - Divisi pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login |
|  | 1. Jika login valid, aplikasi menampilkan form utama Divisi |
|  | 1. Aplikasi menampilkan daftar Divisi yang tersimpan di database |
| *Error flow of Event* |  |  |
| *Post Condition* | Aplikasi menampilkan daftar Divisi yang sudah ada di database | |

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 3 | |
| *Use Case Name* | Tambah Data Divisi | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan menambah data Divisi | |
| *Actor* | Admin | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Admin - Divisi pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login. |
|  | 1. Jika login valid, aplikasi menampilkan form utama Divisi |
|  | 1. Aplikasi menampilkan daftar Divisi yang tersimpan di database |
| 1. Pengguna menekan tombol tambah data |  |
|  | 1. Aplikasi menampilkan form tambah data Divisi |
| 1. Pengguna menginput data Divisi |  |
| 1. Pengguna menekan tombol simpan |  |
|  | 1. Aplikasi melakukan validasi data inputan |
|  | 1. Jika valid, aplikasi akan menyimpan data divisi |
|  | 1. Aplikasi menyimpan history penggunaan |
|  | 1. Aplikasi menampilan pesan data berhasil disimpan |
|  | 1. Aplikasi menampilkan form utama divisi |
|  | 1. Aplikasi menampilkan daftar divisi ter-*update* |
| *Error flow of Event* |  | 2a. Login tidak valid |
|  | 3a. Aplikasi menampilkan halaman login. Kembali ke nomor 1 |
|  | 9a. Data tidak valid |
|  | 10a. Aplikasi menampilkan pesan error data tidak valid pada form tambah data. Kembali ke nomor 7. |
| *Post Condition* | Aplikasi menampilkan form utama dan daftar Divisi ter-*update* yang tersimpan di database | |

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 4 | |
| *Use Case Name* | Ubah Data Divisi | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan mengubah data Divisi | |
| *Actor* | Admin | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Admin - Divisi pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login. |
|  | 1. Jika login valid, aplikasi menampilkan form utama Divisi |
|  | 1. Aplikasi menampilkan daftar Divisi yang tersimpan di database |
| 1. Pengguna memilih data divisi yang akan diubah |  |
| 1. Pengguna menekan tombol Ubah Data |  |
|  | 1. Aplikasi menampilkan form Ubah Data Divisi |
| 1. Pengguna mengubah data Divisi |  |
| 1. Pengguna menekan tombol Simpan |  |
|  | 1. Aplikasi melakukan validasi data inputan |
|  | 1. Jika valid, aplikasi akan menyimpan data divisi |
|  | 1. Aplikasi menyimpan history penggunaan |
|  | 1. Aplikasi menampilan pesan data telah disimpan |
|  | 1. Aplikasi menampilkan form utama divisi |
|  | 1. Aplikasi menampilkan daftar divisi ter-*update* |
| *Error flow of Event* |  | 2a. Login tidak valid |
|  | 3a. Aplikasi menampilkan halaman login. Kembali ke nomor 1 |
|  | 10a. Data tidak valid |
|  | 11a. Aplikasi menampilkan pesan error data tidak valid pada form ubah data. Kembali ke nomor 8. |
| *Post Condition* | Aplikasi menampilkan form utama dan daftar Divisi ter-*update* yang tersimpan di database | |

**4.4.3 Use Case Kelola Data Pengguna**

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 5 | |
| *Use Case Name* | Lihat Data Pengguna | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan melihat data Pengguna | |
| *Actor* | Admin | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Admin - Pengguna pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login |
|  | 1. Jika login valid, aplikasi menampilkan form utama Pengguna |
|  | 1. Aplikasi menampilkan daftar Pengguna yang tersimpan di database |
| *Error flow of Event* |  |  |
| *Post Condition* | Aplikasi menampilkan daftar Pengguna yang tersimpan di database | |

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 6 | |
| *Use Case Name* | Tambah Data Pengguna | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan menambah data Pengguna | |
| *Actor* | Admin | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Admin - Pengguna pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login. |
|  | 1. Jika login valid, aplikasi menampilkan form utama Pengguna |
|  | 1. Aplikasi menampilkan daftar Pengguna yang tersimpan di database |
| 1. Pengguna menekan tombol Tambah Data |  |
|  | 1. Aplikasi menampilkan form Tambah Data Pengguna |
| 1. Pengguna menginput data pengguna |  |
| 1. Pengguna menekan tombol Simpan |  |
|  | 1. Aplikasi melakukan validasi data inputan |
|  | 1. Jika valid, aplikasi akan menyimpan data pengguna |
|  | 1. Aplikasi menyimpan history penggunaan |
|  | 1. Aplikasi menampilkan pesan data telah disimpan |
|  | 1. Aplikasi menampilkan form utama Pengguna |
|  | 1. Aplikasi menampilkan daftar pengguna ter-*update* |
| *Error flow of Event* |  | 2a. Login tidak valid |
|  | 3a. Aplikasi menampilkan halaman login. Kembali ke nomor 1 |
|  | 9a. Data tidak valid |
|  | 10a. Aplikasi menampilkan pesan error data tidak valid pada form tambah data. Kembali ke nomor 7. |
| *Post Condition* | Aplikasi menampilkan form utama dan daftar pengguna ter-*update* yang tersimpan di database | |

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 7 | |
| *Use Case Name* | Ubah Data Pengguna | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan mengubah data pengguna | |
| *Actor* | Admin | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Admin - Pengguna pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login. |
|  | 1. Jika login valid, aplikasi menampilkan form utama pengguna |
|  | 1. Aplikasi menampilkan daftar pengguna yang tersimpan di database |
| 1. Pengguna memilih data pengguna yang akan diubah |  |
| 1. Pengguna menekan tombol Ubah Data |  |
|  | 1. Aplikasi menampilkan form Ubah Data Pengguna |
| 1. Pengguna mengubah data pengguna |  |
| 1. Pengguna menekan tombol Simpan |  |
|  | 1. Aplikasi melakukan validasi data inputan |
|  | 1. Jika valid, aplikasi akan menyimpan data Pengguna |
|  | 1. Aplikasi menyimpan history status penggunaan |
|  | 1. Aplikasi menampilkan pesan data telah disimpan |
|  | 1. Aplikasi menampilkan form utama Pengguna |
|  | 1. Aplikasi menampilkan daftar pengguna ter-*update* |
| *Error flow of Event* |  | 2a. Login tidak valid |
|  | 3a. Aplikasi menampilkan halaman login. Kembali ke nomor 1 |
|  | 10a. Data tidak valid |
|  | 11a. Aplikasi menampilkan pesan error data tidak valid pada form ubah data. Kembali ke nomor 8. |
| *Post Condition* | Aplikasi menampilkan form utama dan daftar pengguna ter-*update* yang tersimpan di database | |

**4.4.4 Use Case Kelola Data Log**

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 8 | |
| *Use Case Name* | Lihat Data Log | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan melihat data Log | |
| *Actor* | Admin | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Admin - Log pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login |
|  | 1. Jika login valid, aplikasi menampilkan form utama Log |
|  | 1. Aplikasi menampilkan daftar Log yang tersimpan di database |
| *Error flow of Event* |  |  |
| *Post Condition* | Aplikasi menampilkan daftar Log yang tersimpan di database | |

**4.4.5 Use Case Kelola Data Kategori**

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 9 | |
| *Use Case Name* | Lihat Data Kategori | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan melihat data Kategori | |
| *Actor* | Admin | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Kategori pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login |
|  | 1. Jika login valid, aplikasi menampilkan form utama Kategori |
|  | 1. Aplikasi menampilkan daftar Kategori yang tersimpan di database |
| *Error flow of Event* |  |  |
| *Post Condition* | Aplikasi menampilkan daftar Kategori yang tersimpan di database | |

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 10 | |
| *Use Case Name* | Tambah Data Kategori | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan menambah data Kategori | |
| *Actor* | Admin | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Kategori pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login. |
|  | 1. Jika login valid, aplikasi menampilkan form utama Kategori |
|  | 1. Aplikasi menampilkan daftar Kategori yang tersimpan di database |
| 1. Kategori menekan tombol Tambah Data |  |
|  | 1. Aplikasi menampilkan form Tambah Data Kategori |
| 1. Pengguna menginput data kategori |  |
| 1. Pengguna menekan tombol Simpan |  |
|  | 1. Aplikasi melakukan validasi data inputan |
|  | 1. Jika valid, aplikasi akan menyimpan data kategori |
|  | 1. Aplikasi menyimpan history penggunaan |
|  | 1. Aplikasi menampilan pesan data telah disimpan |
|  | 1. Aplikasi menampilkan form utama Kategori |
|  | 1. Aplikasi menampilkan daftar kategori ter-*update* |
| *Error flow of Event* |  | 2a. Login tidak valid |
|  | 3a. Aplikasi menampilkan halaman login. Kembali ke nomor 1 |
|  | 9a. Data tidak valid |
|  | 10a. Aplikasi menampilkan pesan error data tidak valid pada form tambah data. Kembali ke nomor 7. |
| *Post Condition* | Aplikasi menampilkan form utama dan daftar Kategori ter-*update* yang tersimpan di database | |

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 11 | |
| *Use Case Name* | Ubah Data Kategori | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan mengubah data Kategori | |
| *Actor* | Admin | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Kategori memilih menu Admin - Kategori pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login. |
|  | 1. Jika login valid, aplikasi menampilkan form utama Kategori |
|  | 1. Aplikasi menampilkan daftar Kategori yang tersimpan di database |
| 1. Pengguna memilih data kategori yang akan diubah |  |
| 1. Pengguna menekan tombol Ubah Data |  |
|  | 1. Aplikasi menampilkan form Ubah Data Kategori |
| 1. Pengguna mengubah data Kategori |  |
| 1. Pengguna menekan tombol Simpan |  |
|  | 1. Aplikasi melakukan validasi data inputan |
|  | 1. Jika valid, aplikasi akan menyimpan data kategori |
|  | 1. Aplikasi menyimpan history penggunaan |
|  | 1. Aplikasi menampilkan pesan data berhasil disimpan |
|  | 1. Aplikasi menampilkan form utama Kategori |
|  | 1. Aplikasi menampilkan daftar kategori ter-*update* |
| *Error flow of Event* |  | 2a. Login tidak valid |
|  | 3a. Aplikasi menampilkan halaman login. Kembali ke nomor 1 |
|  | 10a. Data tidak valid |
|  | 11a. Aplikasi menampilkan pesan error data tidak valid pada form ubah data. Kembali ke nomor 8. |
| *Post Condition* | Aplikasi menampilkan form utama dan daftar Kategori ter-*update* yang tersimpan di database | |

**4.4.6 Use Case Kelola Data Agenda Masuk**

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 12 | |
| *Use Case Name* | Lihat Data Agenda Masuk | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan melihat data Agenda Masuk | |
| *Actor* | User dari masing-masing Divisi | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Agenda Masuk pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login |
|  | 1. Jika login valid, aplikasi menampilkan form utama Agenda Masuk |
|  | 1. Aplikasi menampilkan daftar Agenda Masuk yang tersimpan di database |
| *Error flow of Event* |  |  |
| *Post Condition* | Aplikasi menampilkan daftar Agenda Masuk yang tersimpan di database | |

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 13 | |
| *Use Case Name* | Cari Data Agenda Masuk | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan mencari data Agenda Masuk | |
| *Actor* | User dari masing-masing Divisi | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Agenda Masuk pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login |
|  | 1. Jika login valid, aplikasi menampilkan form utama Agenda Masuk |
|  | 1. Aplikasi menampilkan daftar Agenda Masuk yang tersimpan di database |
|  | 1. Pengguna mengisi isian dasar pencarian |  |
|  | 1. Pengguna menekan tombol Tampilkan |  |
|  |  | 1. Aplikasi mencari daftar Agenda Masuk sesuai dasar pencarian. |
|  |  | 1. Aplikasi menampilkan daftar Agenda Masuk yang tersimpan di database sesuai dasar pencarian |
| *Error flow of Event* |  |  |
| *Post Condition* | Aplikasi menampilkan daftar Agenda Masuk yang tersimpan di database sesuai dasar pencarian | |

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 14 | |
| *Use Case Name* | Tambah Data Agenda Masuk | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan menambah data Agenda Masuk | |
| *Actor* | User dari masing-masing Divisi | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Agenda Masuk pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login. |
|  | 1. Jika login valid, aplikasi menampilkan form utama Agenda Masuk |
|  | 1. Aplikasi menampilkan daftar Agenda Masuk yang tersimpan di database |
| 1. Pengguna menekan tombol Tambah Data |  |
|  | 1. Aplikasi menampilkan form Tambah Data Agenda Masuk |
| 1. Pengguna menginput data Agenda Masuk |  |
| 1. Pengguna menekan tombol Simpan |  |
|  | 1. Aplikasi melakukan validasi data inputan |
|  | 1. Jika valid, aplikasi akan menyimpan data Agenda Masuk |
|  | 1. Aplikasi menyimpan history penggunaan |
|  | 1. Aplikasi menampilkan pesan data berhasil disimpan |
|  | 1. Aplikasi menampilkan form utama Agenda Masuk |
|  | 1. Aplikasi menampilkan daftar Agenda Masuk ter-*update* |
| *Error flow of Event* |  | 2a. Login tidak valid |
|  | 3a. Aplikasi menampilkan halaman login. Kembali ke nomor 1 |
|  | 9a. Data tidak valid |
|  | 10a. Aplikasi menampilkan pesan error data tidak valid pada form Tambah Data. Kembali ke nomor 7. |
| *Post Condition* | Data baru disimpan dalam database. Aplikasi menampilkan form utama dan daftar Agenda Masuk ter-*update* yang tersimpan di database | |

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 15 | |
| *Use Case Name* | Ubah Data Agenda Masuk | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan mengubah data Agenda Masuk | |
| *Actor* | User dari masing-masing Divisi | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Agenda Masuk pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login. |
|  | 1. Jika login valid, aplikasi menampilkan form utama Agenda Masuk |
|  | 1. Aplikasi menampilkan daftar Agenda Masuk yang tersimpan di database |
| 1. Pengguna memilih data Agenda Masuk yang akan diubah |  |
| 1. Pengguna menekan tombol Ubah Data |  |
|  | 1. Aplikasi menampilkan form Ubah Data Agenda Masuk |
| 1. Pengguna mengubah data Agenda Masuk |  |
| 1. Pengguna menekan tombol Simpan |  |
|  | 1. Aplikasi melakukan validasi data inputan |
|  | 1. Jika valid, aplikasi akan menyimpan data Agenda Masuk |
|  | 1. Aplikasi menyimpan history penggunaan |
|  | 1. Aplikasi menampilkan pesan data berhasil disimpan |
|  | 1. Aplikasi menampilkan form utama Agenda Masuk |
|  | 1. Aplikasi menampilkan daftar Agenda Masuk ter-*update* |
| *Error flow of Event* |  | 2a. Login tidak valid |
|  | 3a. Aplikasi menampilkan halaman login. Kembali ke nomor 1 |
|  | 10a. Data tidak valid |
|  | 11a. Aplikasi menampilkan pesan error data tidak valid pada form Ubah Data. Kembali ke nomor 8. |
| *Post Condition* | Data ubahan disimpan dalam database. Aplikasi menampilkan form utama dan daftar Agenda Masuk ter-*update* yang tersimpan di database | |

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 16 | |
| *Use Case Name* | Hapus Data Agenda Masuk | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan menghapus data Agenda Masuk | |
| *Actor* | User dari masing-masing Divisi | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Agenda Masuk pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login. |
|  | 1. Jika login valid, aplikasi menampilkan form utama Agenda Masuk |
|  | 1. Aplikasi menampilkan daftar Agenda Masuk yang tersimpan di database |
| 1. Pengguna memilih data Agenda Masuk yang akan di hapus |  |
| 1. Pengguna menekan tombol Hapus Data |  |
|  | 1. Aplikasi menampilkan konfirmasi hapus Data Agenda Masuk |
| 1. Pengguna menekan tombol Ok |  |
|  | 1. Aplikasi menghapus data Agenda Masuk |
|  | 1. Aplikasi menyimpan history penggunaan |
|  | 1. Aplikasi menampilkan pesan data berhasil dihapus |
|  | 1. Aplikasi menampilkan daftar Agenda Masuk ter-*update* |
| *Error flow of Event* |  | 2a. Login tidak valid |
|  | 3a. Aplikasi menampilkan halaman login. Kembali ke nomor 1 |
| *Post Condition* | Data Agenda Masuk dihapus dari database. Aplikasi menampilkan daftar Agenda Masuk ter-*update* yang tersimpan di database | |

**4.4.7 Use Case Kelola Data Agenda Keluar**

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 17 | |
| *Use Case Name* | Lihat Data Agenda Keluar | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan melihat data Agenda Keluar | |
| *Actor* | User dari masing-masing Divisi | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Agenda Keluar pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login |
|  | 1. Jika login valid, aplikasi menampilkan form utama Agenda Keluar |
|  | 1. Aplikasi menampilkan daftar Agenda Keluar yang tersimpan di database |
| *Error flow of Event* |  |  |
| *Post Condition* | Aplikasi menampilkan daftar Agenda Keluar yang tersimpan di database | |

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 18 | |
| *Use Case Name* | Cari Data Agenda Keluar | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan mencari data Agenda Keluar | |
| *Actor* | User dari masing-masing Divisi | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Agenda Keluar pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login |
|  | 1. Jika login valid, aplikasi menampilkan form utama Agenda Keluar |
|  | 1. Aplikasi menampilkan daftar Agenda Keluar yang tersimpan di database |
|  | 1. Pengguna mengisi isian dasar pencarian |  |
|  | 1. Pengguna menekan tombol Tampilkan |  |
|  |  | 1. Aplikasi mencari daftar Agenda Keluar sesuai dasar pencarian. |
|  |  | 1. Aplikasi menampilkan daftar Agenda Keluar yang tersimpan di database sesuai dasar pencarian |
| *Error flow of Event* |  |  |
| *Post Condition* | Aplikasi menampilkan daftar Agenda Keluar yang tersimpan di database sesuai dasar pencarian | |

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 19 | |
| *Use Case Name* | Tambah Data Agenda Keluar | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan menambah data Agenda Keluar | |
| *Actor* | User dari masing-masing Divisi | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Agenda Keluar pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login. |
|  | 1. Jika login valid, aplikasi menampilkan form utama Agenda Keluar |
|  | 1. Aplikasi menampilkan daftar Agenda Keluar yang tersimpan di database |
| 1. Pengguna menekan tombol Tambah Data |  |
|  | 1. Aplikasi menampilkan form Tambah Data Agenda Keluar |
| 1. Pengguna menginput data Agenda Keluar |  |
| 1. Pengguna menekan tombol Simpan |  |
|  | 1. Aplikasi melakukan validasi data inputan |
|  | 1. Jika valid, aplikasi akan menyimpan data Agenda Keluar |
|  | 1. Aplikasi menyimpan history penggunaan |
|  | 1. Aplikasi menampilkan pesan data berhasil disimpan |
|  | 1. Aplikasi menampilkan form utama Agenda Keluar |
|  | 1. Aplikasi menampilkan daftar Agenda Keluar ter-*update* |
| *Error flow of Event* |  | 2a. Login tidak valid |
|  | 3a. Aplikasi menampilkan halaman login. Kembali ke nomor 1 |
|  | 9a. Data tidak valid |
|  | 10a. Aplikasi menampilkan pesan error data tidak valid pada form Tambah Data. Kembali ke nomor 7. |
| *Post Condition* | Data baru disimpan dalam database. Aplikasi menampilkan form utama dan daftar Agenda Keluar ter-*update* yang tersimpan di database | |

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 20 | |
| *Use Case Name* | Ubah Data Agenda Keluar | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan mengubah data Agenda Keluar | |
| *Actor* | User dari masing-masing Divisi | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Agenda Keluar pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login. |
|  | 1. Jika login valid, aplikasi menampilkan form utama Agenda Keluar |
|  | 1. Aplikasi menampilkan daftar Agenda Keluar yang tersimpan di database |
| 1. Pengguna memilih data Agenda Keluar yang akan diubah |  |
| 1. Pengguna menekan tombol Ubah Data |  |
|  | 1. Aplikasi menampilkan form Ubah Data Agenda Keluar |
| 1. Pengguna mengubah data Agenda Keluar |  |
| 1. Pengguna menekan tombol Simpan |  |
|  | 1. Aplikasi melakukan validasi data inputan |
|  | 1. Jika valid, aplikasi akan menyimpan data Agenda Keluar |
|  | 1. Aplikasi menyimpan history penggunaan |
|  | 1. Aplikasi menampilkan pesan data berhasil disimpan |
|  | 1. Aplikasi menampilkan form utama Agenda Keluar |
|  | 1. Aplikasi menampilkan daftar Agenda Keluar ter-*update* |
| *Error flow of Event* |  | 2a. Login tidak valid |
|  | 3a. Aplikasi menampilkan halaman login. Kembali ke nomor 1 |
|  | 10a. Data tidak valid |
|  | 11a. Aplikasi menampilkan pesan error data tidak valid pada form Ubah Data. Kembali ke nomor 8. |
| *Post Condition* | Data ubahan disimpan dalam database. Aplikasi menampilkan form utama dan daftar Agenda Keluar ter-*update* yang tersimpan di database | |

|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 21 | |
| *Use Case Name* | Hapus Data Agenda Keluar | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan menghapus data Agenda Keluar | |
| *Actor* | User dari masing-masing Divisi | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Agenda Keluar pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login. |
|  | 1. Jika login valid, aplikasi menampilkan form utama Agenda Keluar |
|  | 1. Aplikasi menampilkan daftar Agenda Keluar yang tersimpan di database |
| 1. Pengguna memilih data Agenda Keluar yang akan di hapus |  |
| 1. Pengguna menekan tombol Hapus Data |  |
|  | 1. Aplikasi menampilkan konfirmasi hapus Data Agenda Keluar |
| 1. Pengguna menekan tombol Ok |  |
|  | 1. Aplikasi menghapus data Agenda Keluar |
|  | 1. Aplikasi menyimpan history penggunaan |
|  | 1. Aplikasi menampilkan pesan data berhasil dihapus |
|  | 1. Aplikasi menampilkan daftar Agenda Keluar ter-*update* |
| *Error flow of Event* |  | 2a. Login tidak valid |
|  | 3a. Aplikasi menampilkan halaman login. Kembali ke nomor 1 |
| *Post Condition* | Data Agenda Keluar dihapus dari database. Aplikasi menampilkan daftar Agenda Keluar ter-*update* yang tersimpan di database | |

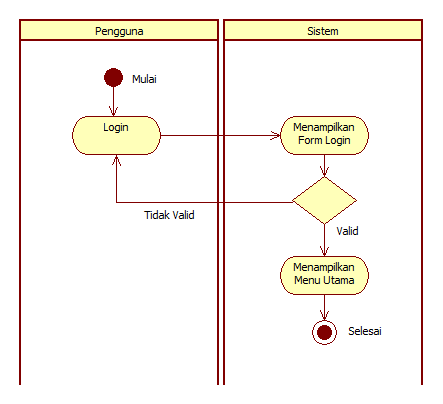
**4.4.8 Use Case Kelola Password**

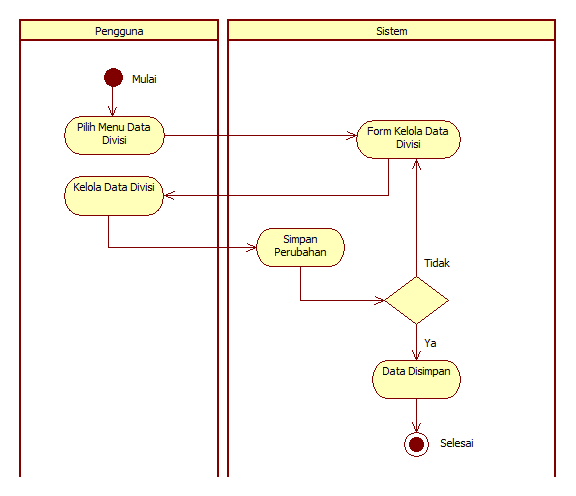
|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 22 | |
| *Use Case Name* | Ubah Password | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan mengubah password pengguna | |
| *Actor* | Admin, User dari masing-masing Divisi | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Ubah Password pada aplikasi e-Filing |  |
|  | 1. Aplikasi mengecek status login. |
|  | 1. Jika login valid, aplikasi menampilkan form Ubah Password |
| 1. Pengguna mengisi form ubah password |  |
| 1. Pengguna menekan tombol Simpan |  |
|  | 1. Aplikasi melakukan validasi data inputan |
|  | 1. Jika valid, aplikasi akan menyimpan data perubahan password |
|  | 1. Aplikasi menyimpan history penggunaan |
|  | 1. Aplikasi menampilkan pesan data berhasil disimpan |
|  | 1. Aplikasi membersihkan isian pada form Ubah Password |
| *Error flow of Event* |  | 2a. Login tidak valid |
|  | 3a. Aplikasi menampilkan halaman login. Kembali ke nomor 1 |
| *Post Condition* | Data pembaharuan password disimpan dalam database | |

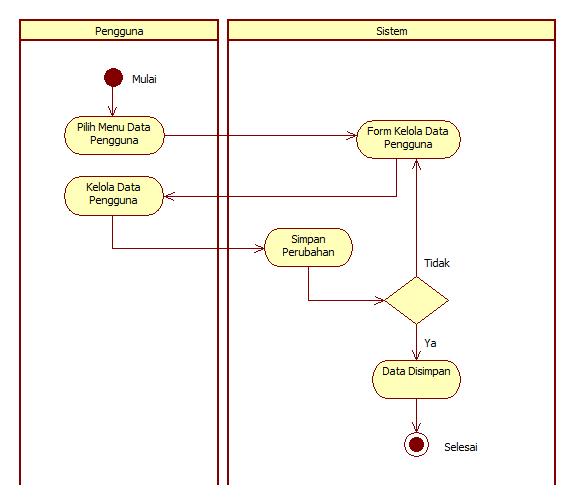
**4.4.9 Use Case Logout**

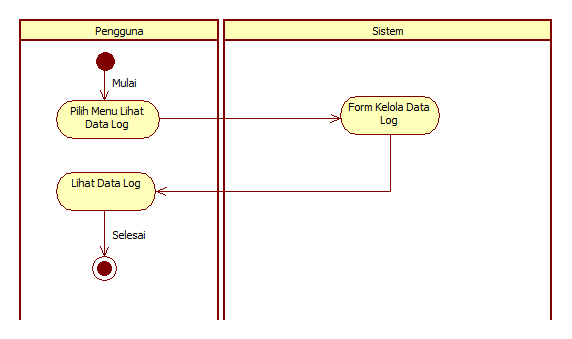
|  |  |  |
| --- | --- | --- |
| *Use Case ID* | 23 | |
| *Use Case Name* | Logout | |
| *Use Case Description* | Use Case ini menggambarkan kegiatan keluar dari aplikasi | |
| *Actor* | Admin, User dari masing-masing Divisi | |
| *Pre Condition* | Pengguna sudah berhasil login ke aplikasi e-Filing | |
| *Primary flow of Event* | *User Action* | *System Response* |
| 1. Pengguna memilih menu Logout pada aplikasi e-Filing |  |
|  | 1. Aplikasi menampilkan form Login |
| *Post Condition* | Aplikasi membersihkan cookies | |

## 4.5 Activity Diagram

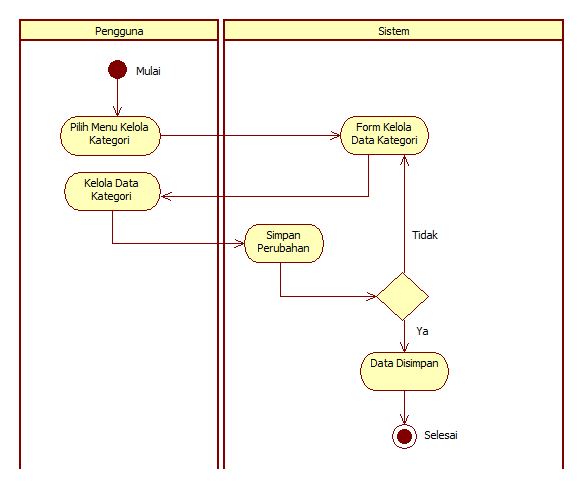
**4.5.1 Activity Diagram Login**

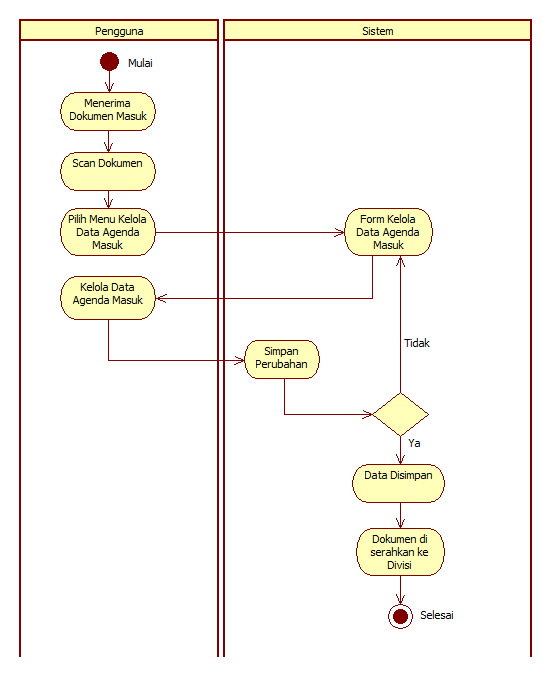
**4.5.2 Activity Diagram Kelola Data Divisi**

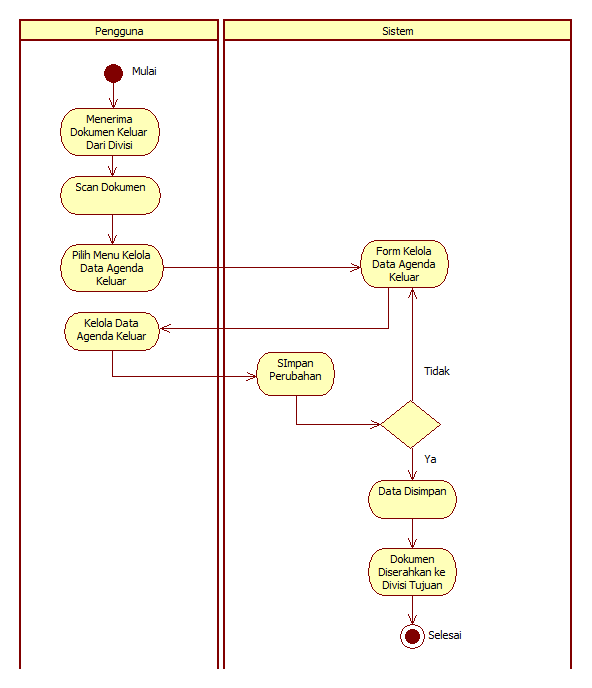
**4.5.3 Activity Diagram Kelola Data Pengguna**

**4.5.4 Activity Diagram Lihat Data Log**

**4.5.5 Activity Diagram Kelola Data Kategori**



**4.5.6 Activity Diagram Kelola Data Agenda Masuk**

**4.5.7 Activity Diagram Kelola Data Agenda Keluar**

# Non Functional Requirements

| **Parameter** | **Kebutuhan** |
| --- | --- |
| Availability | 24 jam nonstop, kecuali ada *maintenance* / perbaikan system/ dan force major |
| Reliability | Kegagalan yang ditolerir sekitar 3% |
| Ergonomy | Sistem informasi ini harus *user friendly dan mudah di akses penggunaannya* |
| Portability | Aplikasi ini berjalan pada platform Sistem operasi Windows XP, Windows 7, Windows 8, Windows 10, Linux, MacOS dan Browser Mozilla Firefox, Google Chrome, atau browser modern lainnya. |
| Memory | Minimum memory 128 Mb |
| Response time | Tidak lebih dari 7 detik |
| Safety | Menggunakan *secure socket layer* dengan sertifikasi, https:... |
| Security | *Login* dan validasi data sangat penting  karena menyangkut dokumentasi dokument |
|  |  |
| Others 1: Bahasa komunikasi | Fasilitas untuk menggunakan bahasa lain selain bahasa Indonesia |
| Halaman Utama | Setiap layar mengandung identitas program aplikasi |
|  |  |

Analisis kebutuhan non fungsional memiliki beberapa kebutuhan, yaitu sebagai berikut:

•

Operasional

-

Menggunakan Server yang mempunyai

web server,

dan

mySQL server

-

Minimal RAM 4 GB

-

Minimal Hardisk 10 GB

-

Koneksi minimal LAN

-

Printer

•

Keamanan

-

Sistem dilengkapi dengan autentikasi per level.

•

Informasi

-

Digunakan untuk menampilkan informasi aturan Perwalian.

-

Digunakan untuk menampilkan informasi bila pengguna lupa sandi atau salah sandi.

•

Kinerja

-

Waktu perwalian akan dibalas maksimal satu hari jam kerja setelah mahasiswa mengisi formulir perwalian

Analisis kebutuhan non fungsional memiliki beberapa kebutuhan, yaitu sebagai berikut:

•

Operasional

-

Menggunakan Server yang mempunyai

web server,

dan

mySQL server

-

Minimal RAM 4 GB

-

Minimal Hardisk 10 GB

-

Koneksi minimal LAN

-

Printer

•

Keamanan

-

Sistem dilengkapi dengan autentikasi per level.

•

Informasi

-

Digunakan untuk menampilkan informasi aturan Perwalian.

-

Digunakan untuk menampilkan informasi bila pengguna lupa sandi atau salah sandi.

•

Kinerja

-

Waktu perwalian akan dibalas maksimal satu hari jam kerja setelah mahasiswa mengisi formulir perwalian

Analisis kebutuhan non fungsional memiliki beberapa kebutuhan, yaitu sebagai berikut:

Operasional

-

Menggunakan Server yang mempunyai

web server,

dan

mySQL server

-

Minimal RAM 4 GB

-

Minimal Hardisk 10 GB

-

Koneksi minimal LAN

-

Printer

Operasional

-

Menggunakan Server yang mempunyai

web server,

dan

mySQL server

-

Minimal RAM 4 GB

-

Minimal Hardisk 10 GB

-

Koneksi minimal LAN

-

Printe

Analisis kebutuhan non fungsional memiliki beberapa kebutuhan, yaitu sebagai berikut:

•

Operasional

-

Menggunakan Server yang mempunyai

web server,

dan

mySQL server

-

Minimal RAM 4 GB

-

Minimal Hardisk 10 GB

-

Koneksi minimal LAN

-

Printer

•

Keamanan

-

Sistem dilengkapi dengan autentikasi per level.

•

Informasi

-

Digunakan untuk menampilkan informasi aturan Perwalian.

-

Digunakan untuk menampilkan informasi bila pengguna lupa sandi atau salah sandi.

•

Kinerja

-

Waktu perwalian akan dibalas maksimal satu hari jam kerja setelah mahasiswa mengisi formulir perwalian

**5.1** **Kebutuhan Kinerja**

Kebutuhan kinerja dalam program ini antara lain adalah kebutuhan perangkat keras berupa *Personal Computer* (PC) berupa *Central* *Processing Unit* (CPU), *mouse*, *keyboard*, *monitor*, dimana perangkatPC harus terhubung dengan jaringan *intranet* dan internet, sehingga membutuhkan *Local Area Network* (LAN) *Card*. Sedangkan untuk kebutuhan perangkat lunak yang harus disediakan adalah berupa sebuah web *browser* seperti Internet Explorer, Mozilla Firefox, dan sebagainya untuk menjalankan aplikasi berbasis *web*..2

**5.2 Kebutuhan keamanan**

Sistem dilengkapi dengan autentikasi per level.

**5.3 Kebutuhan Keamanan Perlindungan**

Setiap data yang di unggah dalam betuk dokument akan terlindungi dengan Menggunakan *secure socket layer* dengan sertifikasi