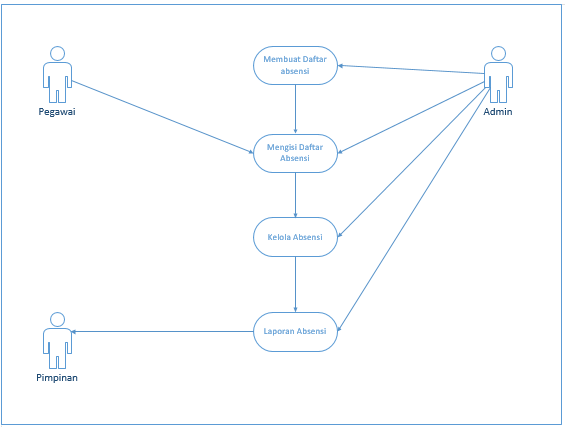
# BAB IV

# ANALISIS KERJA PRAKTEK

## Perancangan Sistem yang Berjalan

Perancangan sistem yang berjalan merupakan sebuah gambaran dari suatu sistem yang sedang berjalan, dengan hal ini kita bisa mengetahui bagaimana gambaran aliran-aliran informasi yang terjadi pada perusahaan tersebut yang di dokumentasikan dengan menggunakan diagram *use case, scenario,* dan *activity diagram* yang dimana pembahasan hanya berjalan pada bagian sistem informasi absensinya saja.

1. *Usecase*



1. Skenario
2. Deskripsi *user*
3. Admin : Orang yang bertugas untuk membuat daftar absensi, lalu juga melakukan pengelolahan absensi, dan membuat laporan untuk diserahkan ke pada pimpinan
4. Pegawai : Orang yang hanya di perbolehkan melakukan untuk mengisi daftar absensi saja.
5. Pimpinan : Orang yang menerima hasil laporan dari absensi yang dibuat oleh admin.
6. Skenario

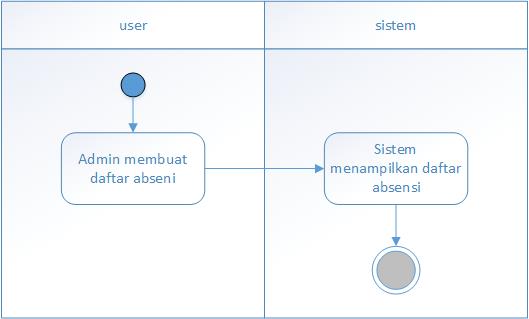
|  |  |  |  |
| --- | --- | --- | --- |
| **Nama *Use Case*** | | Membuat Daftar Absensi | |
| **Aktor** | | Admin | |
| **Deskripsi** | | Untuk melakukan Absensi, admin diharuskan membuat daftar absensi. | |
| Skenario Utama | | | |
| Kondisi Awal : Admin menerima daftar pegawai | | | |
| No. | Admin | No. | Reaksi Sistem |
| 1. | Admin membuat daftar kehadiran pegawai |  |  |
|  |  | 2. | Menerima daftar kehadiran pegawai |
| Kondisi Akhir : Admin telah selesai membuat daftar kehadiran pegawai. | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nama *Use Case*** | | Mengisi Daftar Absen | | |
| **Aktor** | | Admin | | |
| **Deskripsi** | | Admin melakukan pengisian daftar kehadiran absensi | | |
| Skenario Utama | | | | |
| Kondisi Awal : Admin mempersiapkan data kehadiran | | | | |
| No. | Admin | | No. | Reaksi Sistem |
| 1. | Admin mengabsen para pegawai yang telah hadir | |  |  |
| 2. | Pegawai memberitahu admin mengenai kehadiran | |  |  |
| 3. | Admin mencatat pegawai yang hadir | |  |  |
|  |  | | 2. | Sistem menyimpan data kehadiran yang telah diisi |
| Kondisi Akhir : Data kehadiran pegawai telah di simpan | | | | |

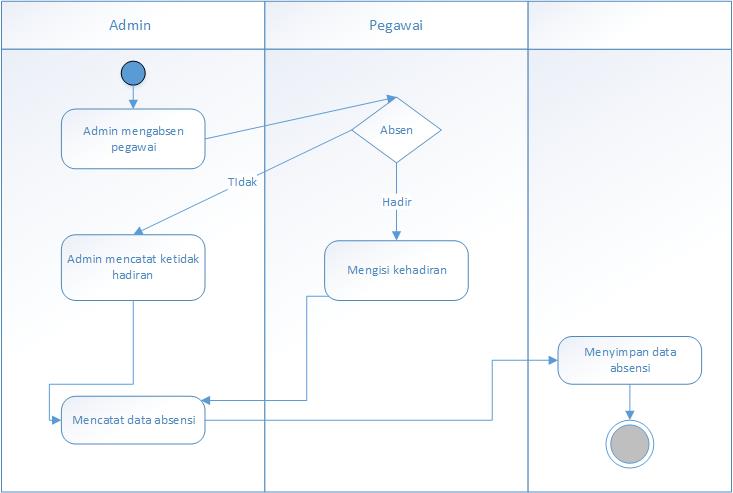
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nama *Use Case*** | | Kelola Absensi | | |
| **Aktor** | | Admin | | |
| **Deskripsi** | | Admin melakukan pengelolahan absensi untuk membuat laporan absensi. | | |
| Skenario Utama | | | | |
| Kondisi Awal : Admin akan melakukan pengelolan data kehadiran pegawai. | | | | |
| No. | Admin | | No. | Reaksi Sistem |
| 1. | Admin membuka kembali data kehadiran yang telah di simpan | |  |  |
|  |  | | 2. | Sistem menampilkan data |
| 3. | Admin memeriksa kembali data absensi tersebut apa telah sesuai apa tidak. | |  |  |
| Kondisi Akhir : Admin menyimpan data yang telah di kelola | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Nama *Use Case*** | | Laporan Absensi | |
| **Aktor** | | Admin/Pimpinan | |
| **Deskripsi** | | Laporan absensi yang telah di buat oleh admin diserahkan ke pimpinan. | |
| Skenario Utama | | | |
| Kondisi Awal : admin telah menyiapkan data yang telah dikelola | | | |
| No. | Admin | No. | Reaksi Sistem |
| 1. | Admin membuat laporan dari data absensi yang telah dikelola |  |  |
| 2. | Admin menyerahkan laporan absensi kepada pimpinan |  |  |
| 3. | Pimpinan menerima laporan absensi kepada pegawai |  |  |
| Kondisi Akhir : Pimpinan memeriksa laporan yang telah diterima dari admin | | | |

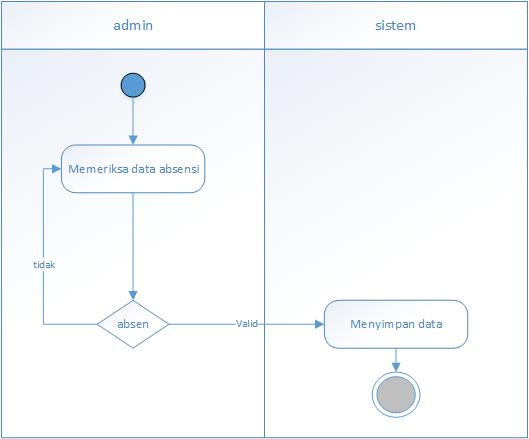
1. *Activity Diagram*
2. Membuat Daftar Absensi



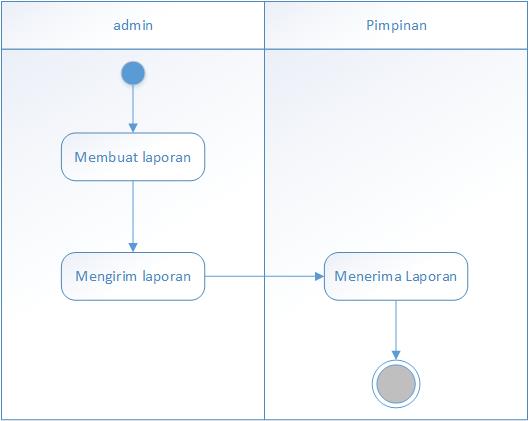
1. Mengisi Daftar Absensi



1. Kelola Absensi

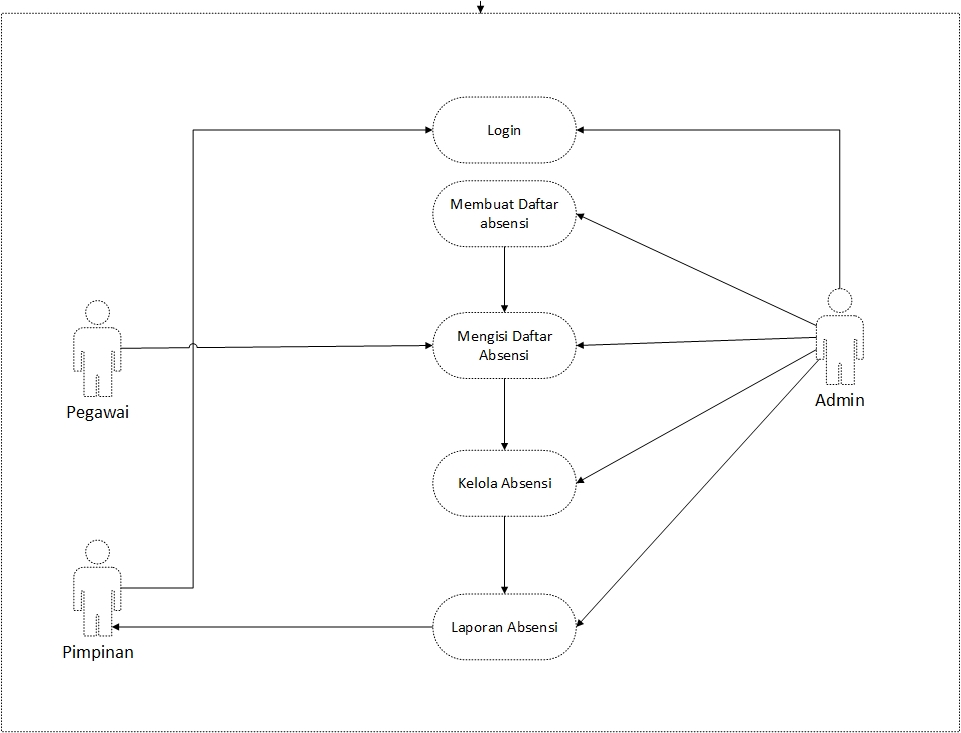


1. Laporan Absensi



## Perancangan Sistem yang Diusulkan

1. Use Case



1. Skenario
2. Deskripsi *user*
3. Admin : Orang yang bertugas untuk membuat daftar absensi, lalu juga melakukan pengelolahan absensi, dan membuat laporan untuk diserahkan ke pada pimpinan.
4. Pegawai : Orang yang hanya di perbolehkan melakukan untuk mengisi daftar absensi saja.
5. Pimpinan : Orang yang menerima hasil laporan dari absensi yang dibuat oleh admin.
6. Skenario

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nama *Use Case*** | | Login | | |
| **Aktor** | | Admin/Pimpinan | | |
| **Deskripsi** | | Actor melakukan login untuk dapat menggunakan program. | | |
| Skenario Utama | | | | |
| Kondisi Awal : Aktor ingin menggunakan program dan belum melakukan login | | | | |
| No. | Admin | | No. | Reaksi Sistem |
| 1. | Menginput *user name* dan *password* | |  |  |
| 2. | Mengklik login | | 3. | Verifikasi dengan database |
|  |  | | 4. | Gagal. Muncul notifikasi password atau username salah. |
|  |  | | 5. | Sesuai. Muncul notifikasi login berhasil. |
|  |  | | 6. | Menampilkan halaman utama program. |
| 7. | Melakukan aktivitas dalam program. | |  |  |
| Kondisi Akhir : Aktor telah melakukan login dan beraktivitas dalam program. | | | | |

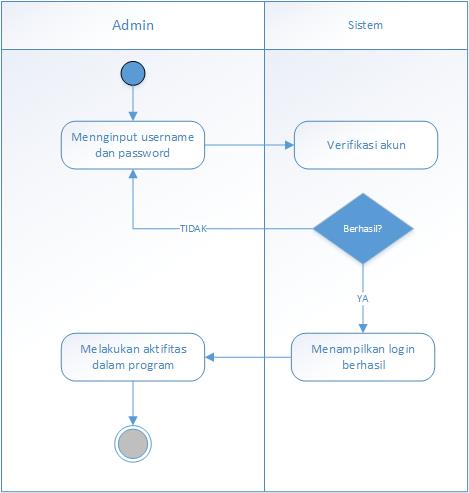
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nama *Use Case*** | | Membuat Daftar Absensi | | |
| **Aktor** | | Admin | | |
| **Deskripsi** | | Untuk melakukan Absensi, admin diharuskan membuat daftar absensi. | | |
| Skenario Utama | | | | |
| Kondisi Awal : Admin menerima daftar pegawai | | | | |
| No. | Admin | | No. | Reaksi Sistem |
| 1. | Admin membuat daftar kehadiran pegawai | |  |  |
|  |  | | 2. | Menerima daftar kehadiran pegawai |
| Kondisi Akhir : Admin telah selesai membuat daftar kehadiran pegawai. | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nama *Use Case*** | | Mengisi DaftarAbsensi | | |
| **Aktor** | | Pegawai/Admin | | |
| **Deskripsi** | | Pegawai/admin melakukan pengisian daftar kehadiran absensi | | |
| Skenario Utama | | | | |
| Kondisi Awal : Admin mempersiapkan data kehadiran | | | | |
| No. | Pegawai | | No. | Reaksi Sistem |
| 1. | Pegawai menginput NIP masing-masing | |  |  |
|  |  | | 2. | Sistem menyimpan data kehadiran yang telah diisi |
| Kondisi Akhir : Data kehadiran pegawai telah di simpan | | | | |

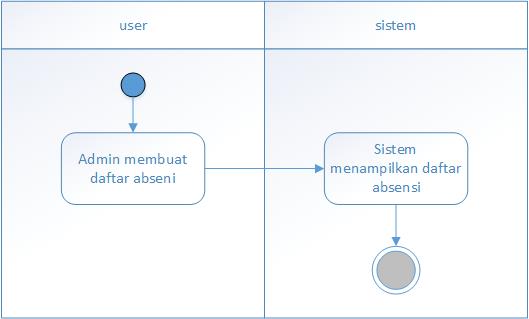
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nama *Use Case*** | | Kelola Absensi | | |
| **Aktor** | | Admin | | |
| **Deskripsi** | | Admin melakukan pengelolaaan absensi untuk membuat laporan absensi. | | |
| Skenario Utama | | | | |
| Kondisi Awal : Admin akan melakukan pengelolan data kehadiran pegawai. | | | | |
| No. | Admin | | No. | Reaksi Sistem |
| 1. | Admin membuka kembali data kehadiran yang telah di simpan | |  |  |
|  |  | | 2. | Sistem menampilkan data |
| 3. | Admin memeriksa kembali data absensi tersebut apa telah sesuai apa tidak. | |  |  |
| Kondisi Akhir : Admin menyimpan data yang telah di kelola | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nama *Use Case*** | | Laporan Absensi | | |
| **Aktor** | | Aktor/Pimpinan | | |
| **Deskripsi** | | Laporan absensi yang telah di buat oleh admin diserahkan ke pimpinan. | | |
| Skenario Utama | | | | |
| Kondisi Awal : admin telah menyiapkan data yang telah di kelola | | | | |
| No. | Admin | | No. | Reaksi Sistem |
| 1. | Admin membuat laporan dari data absensi yang telah di kelola | |  |  |
| 2. | Admin menyerahkan laporan absensi kepada pimpinan | |  |  |
| 3. | Pimpinan menerima laporan absensi kepada pegawai | |  |  |
| Kondisi Akhir : Pimpinan memeriksa laporan yang telah diterima dari admin | | | | |

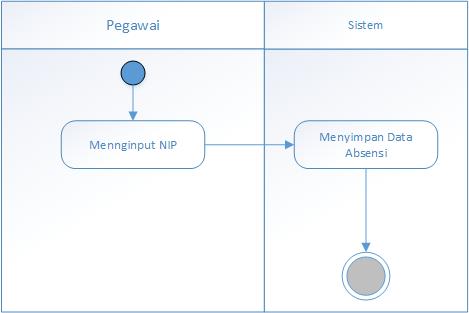
1. Activity Diagram
2. Login



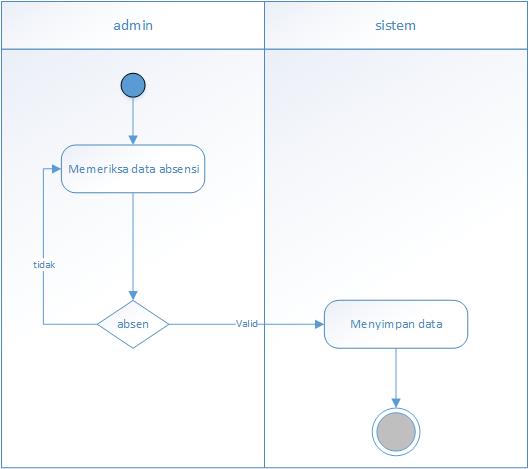
1. Membuat Daftar Absensi



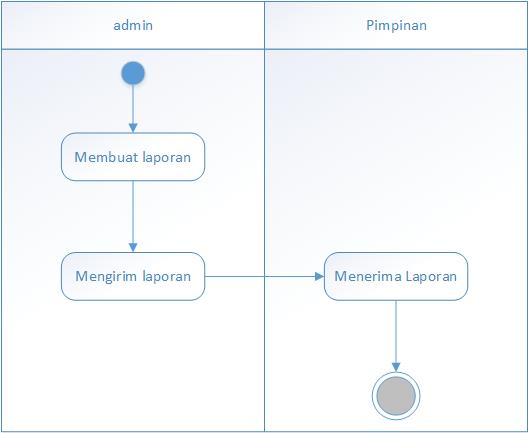
1. Mengisi Daftar Absensi



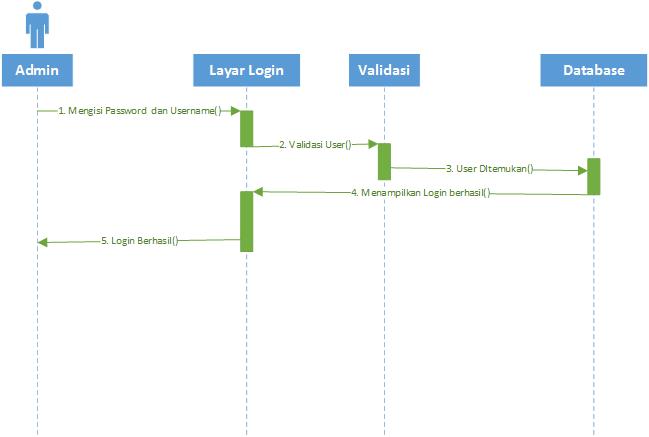
1. Kelola Absensi



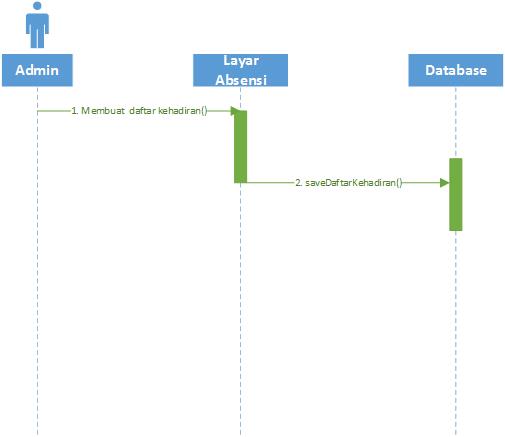
1. Laporan Absensi



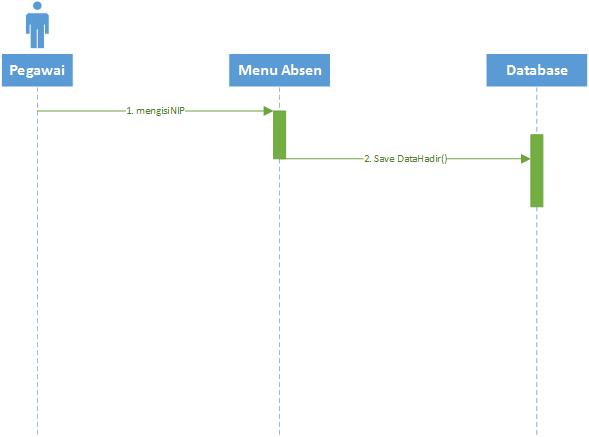
1. Sequence Diagram
2. Login

**

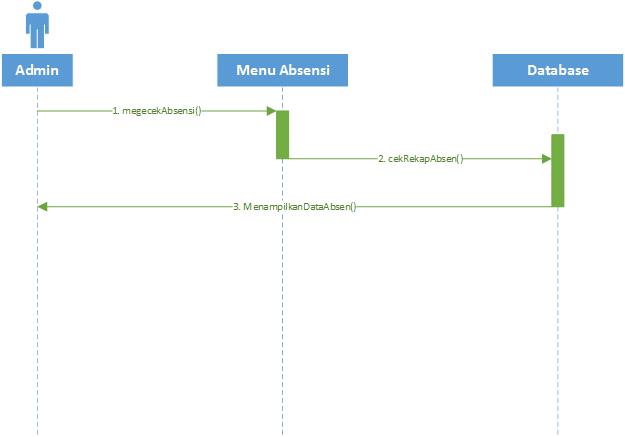
1. Membuat Daftar Absensi



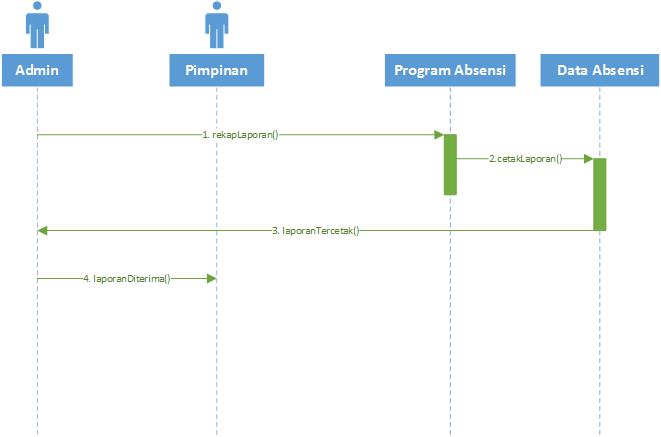
1. Mengisi Daftar Absensi



1. Kelola Absensi



1. Laporan Absensi



## Perancangan Antar Muka dan Jaringan

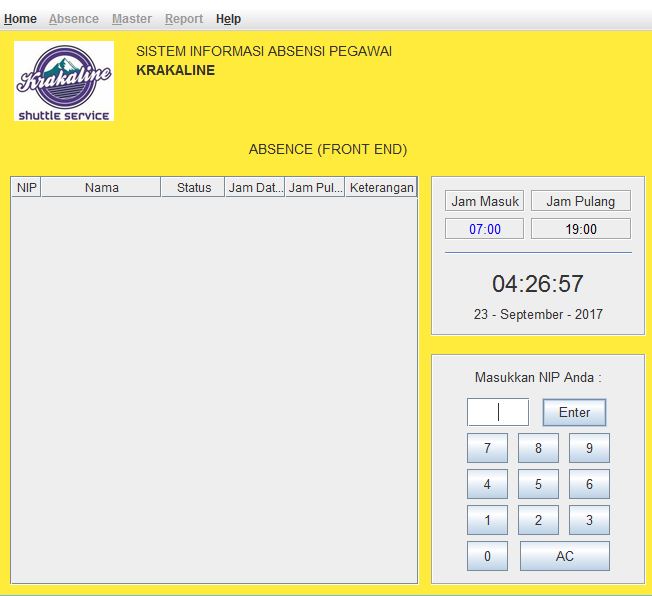
### Perancangan Antar Muka

Layar awal program

1. Perancangan *Input*
2. *Input Absence*

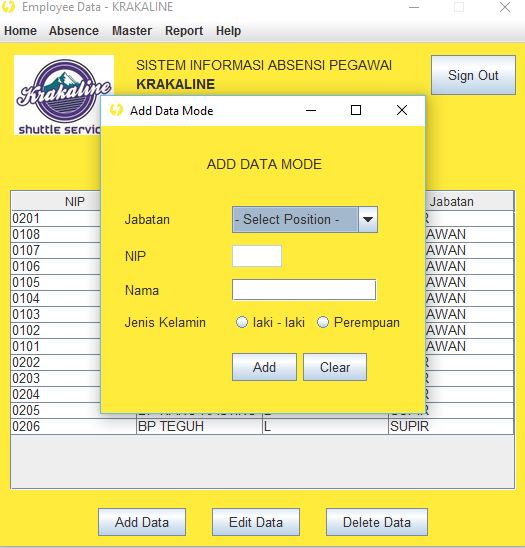
Menu ini terdapat 2 bagian yaitu *back-end* dan *front-end*. Dimana *front-end* untuk seluruh pegawai yang hendak mengisi absen nya. Sedangkan *back-end* hanya untuk pegawai yang memiliki hak akses. Untuk memasukkan absen, berikut caranya :

1. Pegawai memasukkan NIP mereka lalu klik enter.
2. Kemudian tabel di sebelah kiri menampilkan detail mulai dari NIP, Nama, Status, Jam Datang, Jam Pulang berikut Keterangan telat nya.



1. *Input Employee*

Menu ini hanya terdapat 1 bagian yaitu *back-end*. Menu ini digunakan untuk menampilkan data seluruh pegawai dan juga terdapat fungsi tambah data pegawai, edit data pegawai dan delete data pegawai. Berikut tampilan menu *Employee*.



Untuk *Add* Data :

1. Pilih *add* Data

2. Setelah itu akan muncul jendela baru.

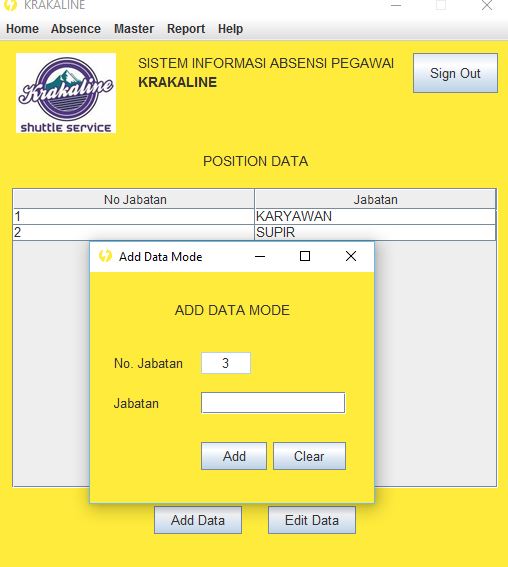
3. Pilih dan isi jabatan, nip, nama, dan jenis kelamin

4. Klik *Add*

5. Selesai

1. *Input Position*

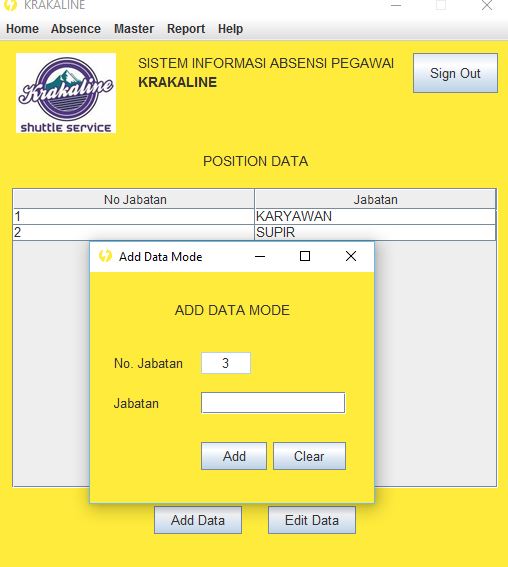
Menu ini berfungsi untuk mengedit, menambah jabatan-jabatan di perusahaan . Menu ini hanya tersedia *back-end* saja jadi harus *sign-in* terlebih dahulu. Berikut tampilan menu *position*.



Untuk *add* data, pertama-tama klik *add* lalu isi nama jabatan baru. Untuk no. jabatan sudah ter-*generate* otomatis lalu klik ‘*add*’

1. *Input Account*

Menu ini digunakan untuk menampilkan data seluruh akun yang memilikki akses ke aplikasi dan juga terdapat fungsi tambah data akun, edit data akun dan delete data akun. Berikut tampilan menu *Account*



Untuk *Add* Data :

1. Pilih *add* Data

2. Setelah itu akan muncul jendela baru Menu *add data mode*

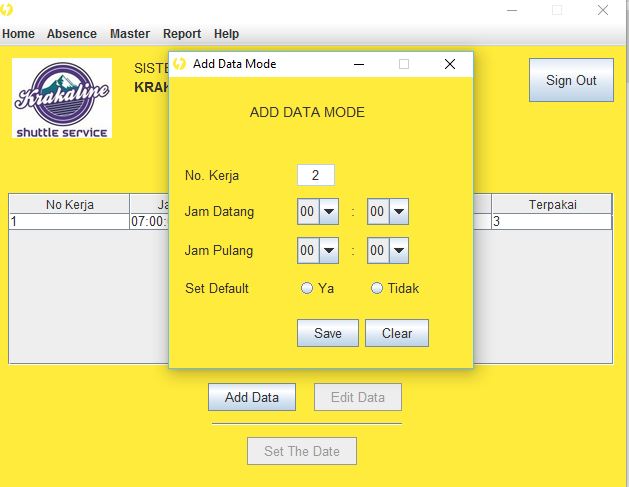
3. Pilih dan isi nip, *password*, dan jenis akses

4. Lalu klik *add*

5. Selesai

1. *Input Working Hour*

*Working Hour* digunakan untuk menambahkan Jam Kerja karyawan di Krakaline. Hanya bisa diakses oleh admin, jadi kita harus sign in terlebih dahulu seperti *back-end* untuk modul *Absence, Employee* dst. Berikut merupakan tampilan *Working Hour.*



Untuk *Add* Data :

1. Pilih *add* data.

2. Kemudian akan muncul submenu tambah data / *add data*.

3. Pilih Jam dan Menit masuk pegawai, misalkan 07 : 00. Berarti pegawai harus masuk pukul 07 : 00.

4. Pilih Jam dan Menit keluar pegawai, misalkan 17 : 00. Berarti pegawai akan pulang pukul 17 : 00.

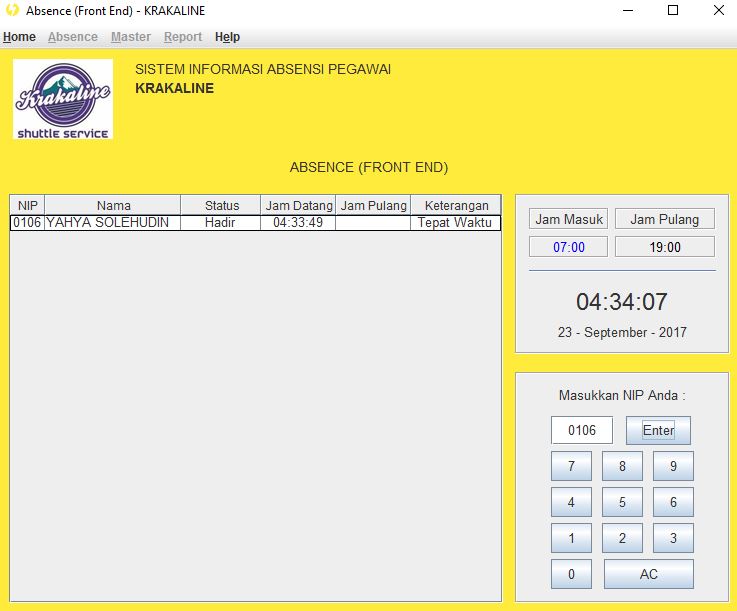
5. Set default digunakan jika ingin menetapkan Jam Masuk dan Jam Pulang pegawai saat hari biasa. Jika dalam keadaan khusus (masuk setengah hari dsb), maka kita harus memasukan *Working Hour* selanjutnya tanpa men-klik *default*.

6. Setelah itu kita klik *Save*.

7. Selesai

1. Perancangan *Output*
2. *Output Absence*

Inilah tampilan output dari menu absence dimana terdapat NIP, Nama, Status kehadiran, Jam Datang, Jam Pulang, dan Keterangan yang menunjukkan karyawan dating tepat waktu atau terlambat.



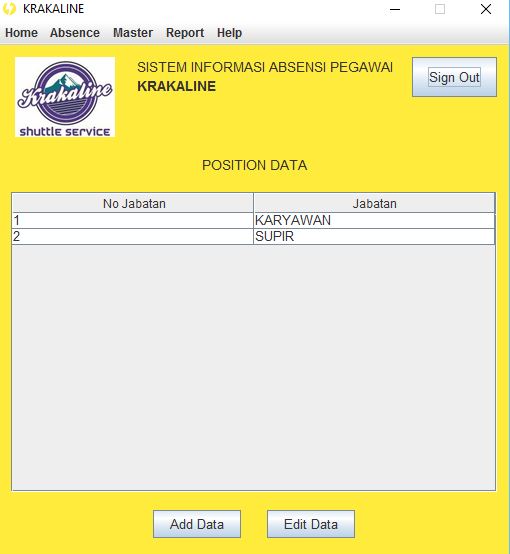
1. *Output Employee*

Tampilan output menu *Employee* ini menunjukkan semua pegawai dengan NIP, Nama, Jenis Kelamin dan Jabatan masing-masing.



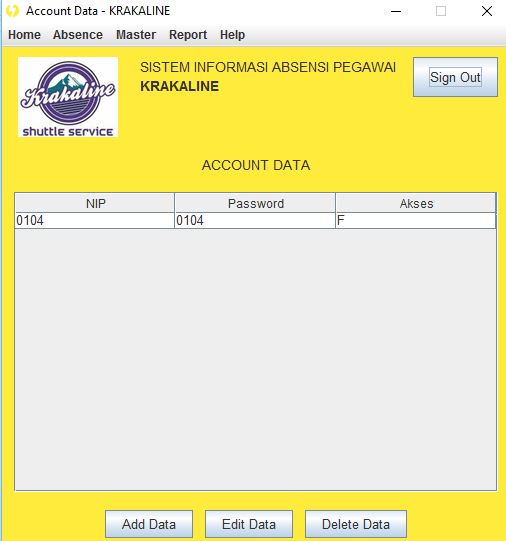
1. *Output Position*

*Output* ini menunjukkan jabatan apa saja yang tersedia di Krakaline.



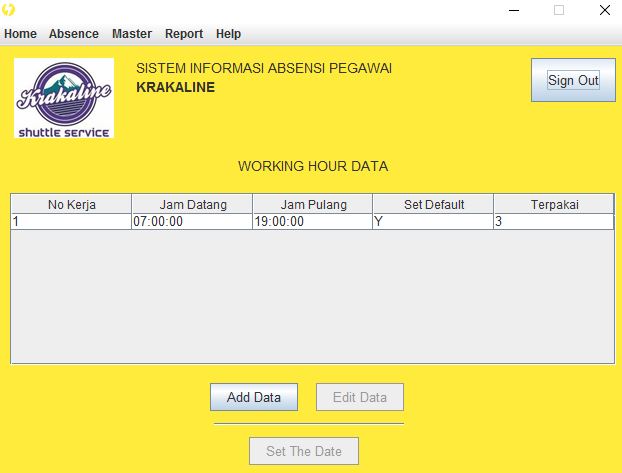
1. *Output Account*

*Output* ini menujukkan akun yang tersedia untuk menjadi admin dan mengolah data di program ini.



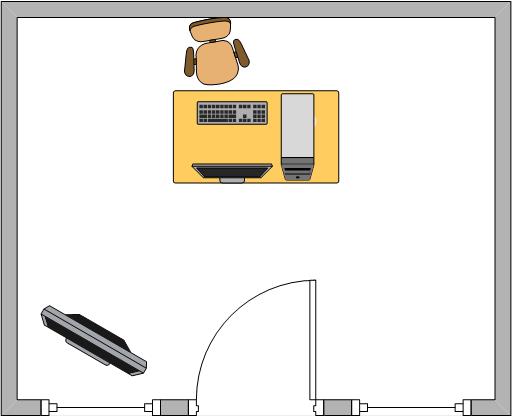
1. *Output Working Hour*

*Output* ini menunjukkan jam kerja di Krakaline dari Jam Datang dan Jam Pulang.



### Peracangan Arsitektur Jaringan

### Untuk Perancangan Arsitektur Jaringan sangat sederhana, yaitu seperti ini. Di mana Pintu masuk berada di depan meja admin.



### Implementasi Program

Sebelum program diimplementasikan, maka program harus bersih dari cacat dan kesalahan. Analisis perangkat keras sangat penting agar program berjalan dengan semestinya.

1. Spesifikasi Perangkat Keras

Prosesor : Intel Pentium g840

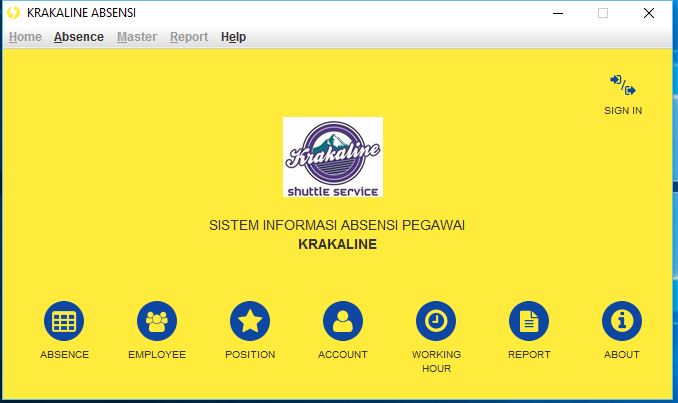
*RAM* : 4GB DDR3

*VGA* : Nvidia GT 730 2GB

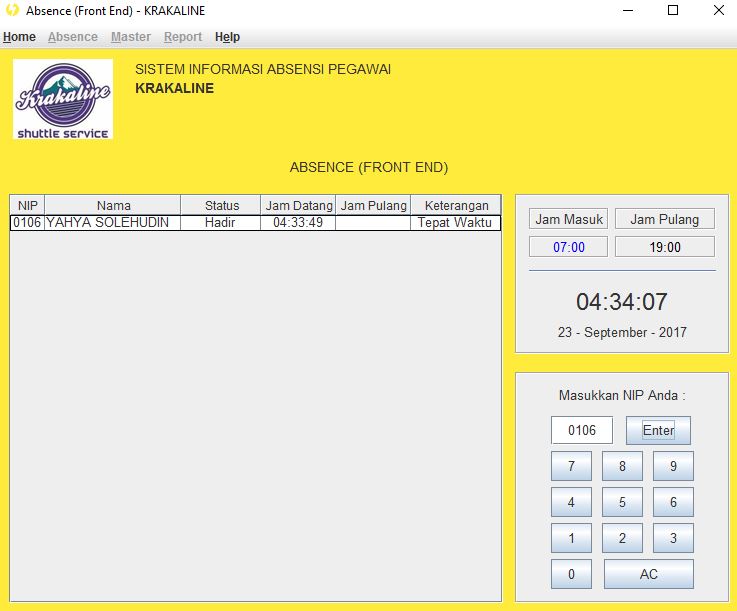
*Harddisk* : 500GB Sata

*Monitor* : BenQ 16”

1. Implementasi Antar Muka
2. *Beranda*



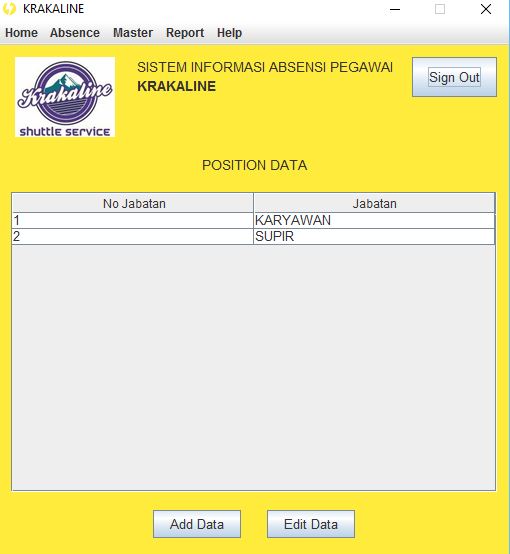
1. *Absence*



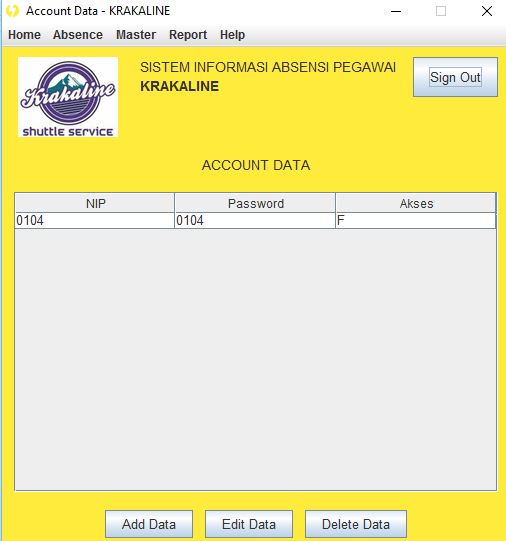
1. *Employee*



1. *Position*



1. *Account*



1. *Working Hour*

