*Software Requirements Specification*

APLIKASI TIKET

IT HELPDESK

Version 1.0 approved

Prepared by

< 071022000034 - Ahmad Zulfikar >

< 071022000034 - M Muzofar Lambar >

<26 Maret 2018>

Table of Contents

1 Pendahuluan 2

1.1 Tujuan Penulisan Dokumen 2

1.2 Audien yang Dituju dan Pembaca yang Disarankan 2

1.3 Batasan Produk 2

1.4 Definisi dan Istilah 3

1.5 Refrensi 3

2 Deskripsi Keseluruhan 5

2.1 Deskripsi Produk 5

2.2 Fungsi Produk 5

2.3 Penggolongan Karakterik Pengguna 5

2.4 Batasan Desain dan Implementasi 6

2.5 Asumsi dan Ketergantungan 6

3 Kebutuhan Antarmuka Eksternal 7

3.1 User Interfaces 7

3.2 Hardware Interface 7

3.3 Software Interface 7

3.4 BatasanPerancangan 8

4 Functional Requirement 9

5 Non Functional Requirements 10

5.1 Usability requirements 10

5.2 Reability requirements 10

5.3 Safety requirements 10

5.4 Security requirements 10

5.5 Performance requirements 10

5.6 Maintainability and upgradability requirements 11

6 Dokumen Desain Aplikasi 12

6.1 Desain Fungsional 12

6.2 Perancangan Basis Data 12

6.3 Perancangan UML (*Unifiled Modelling Language*) 17

6.4 Implementasi Sistem 31

7 Implementasi Antarmuka 32

7.1 Tampilan Antarmuka 32

7.2 Pengujian Sistem 36

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen ini berisi Spesifikasi Kebutuhan Perangkat Lunak (SKPL) atau *Software Requirement Spesification (SRS)* untuk Aplikasi Tiket IT *Helpdesk* Berbasis Web. Tujuan dari penulisan dokumen ini adalah untuk memberikan penjelasan mengenai perangkat lunak yang akan dibangun baik berupa gambaran umum maupun penjelasan detil dan menyeluruh.

## Audien yang Dituju dan Pembaca yang Disarankan

Pengguna dari dokumen ini adalah pengembang perangkat lunak tiket IT Helpdesk dan pengguna (user) dari perangkat lunak atau personil-personil yang terlibat dalam sistem. Dokumen ini akan digunakan sebagai bahan acuan dalam proses pengembangan dan sebagai bahan evaluasi pada saat proses pengembangan perangkat lunak maupun di akhir pengembangannya. Dengan adanya dokumen SKPL atau SRS ini diharapkan pengembangan perangkat lunak akan lebih terarah dan lebih terfokus serta tidak menimbulkan ambiguitas terutama bagi pengembang perangkat lunak tiket IT Helpdesk.

## Batasan Produk

Perangkat lunak yang akan dikembangkan adalah perangkat lunak tiket IT Helpdesk berbasis Web, yaitu merupakan perangkat lunak yang digunakan untuk mempermudah pengguna perangkat IT pada suatu perusahaan untuk meminta bantuan dari IT *Support* pada suatu perusahaan dan dapat digunakan untuk mengevaluasi sistem IT yang terdapat pada suatu perusahaan berdasarkan laporan kendala peggunaan perangkat IT yang dihadapi oleh karyawan suatu perusahaan. Aplikasi tiket IT Helpdesk ini dapat melakukan hal-hal berikut ini :

### Fasilitas Login untuk admin, dan karyawan untuk menghindari penyalahgunaan hak akses.

### Menampilkan daftar jenis tiket ( jenis gangguan ) yang akan dilaporkan oleh karyawan.

### Menampilkan laporan jenis tiket ( jenis gangguan ) yang telah dilaporkan.

### Menampilkan laporan penanganan pada setiap laporan gangguan yang dilaporkan oleh karyawan.

### Laporan rekap presentasi jenis tiket ( jenis ganggunan ) yang dilaporkan oleh karyawan dalam jangka waktu yang bervariasi.

Dengan adanya Aplikasi ini diharapkan, karyawan dapan dengan mudah meminta bantuan IT Helpdesk pada suatu perusahaan jika mengalami kendala dalam menggunakan perangkat IT. Penanganan permasalahan IT akan dapat dilakukan secara cepat agar tidak banyak mengganggu aktivitas pekerjaan karyawan yang sangat tergantung dengan IT

.

## Definisi dan Istilah

Tabel 1.Definisi, Singkatan dan Akronim

|  |  |
| --- | --- |
| Istilah, AkronimdanSingkatan | Keterangan |
| * SRS | * Software Requirements Specification, atau   Spesifikasi Kebutuhan Perangkat Lunak (SKPL) |
| * Admin | * Merupakan seseorang yang bertanggungjawab untuk perawatan sistem dan serta bertanggungjawab terhadap operasional sistem. |
| * User | * IT Helpdesk yang akan menerima laporan kendala dari karyawan |
| * Karyawan | * Merupakan orang yang akan mengirimkan laporan kendala yang dihadapi saat menggunakan IT. |
| * Web | * adalahhalamaninformasi yang disediakanmelaluijalur internet sehingga bisa diakses di seluruhduniaselamaterkoneksidengan internet |
| * Transaction report | * Laporan rekapitulasi transaksi per satu pertandingan |
| * Monthly report | * Laporan rekapitulasi transaksi per satu bulan |

## Refrensi

Dokumen-dokumen yang digunakan sebagai referensi dalam pembuatan SKPL ini adalah sebagai berikut:

### DOKUMEN1: menjelaskan tentang database system dan database laporan.

### *DOKUMEN2: daftar Tiket, Laporan Tiket, Rekap Laporan Tiket*

### *DOKUMEN3: format keluhan standar.*

# Deskripsi Keseluruhan

## Deskripsi Produk

Perangkat lunak tiket IT *Helpdesk* berbasis web ini merupakan perangkat lunak yang digunakan untuk mempermudah pengguna perangkat IT pada suatu perusahaan untuk meminta bantuan dari IT Support pada suatu perusahaan dan dapat digunakan untuk mengevaluasi sistem IT yang terdapat pada suatu perusahaan berdasarkan laporan kendala peggunaan perangkat IT yang dihadapi oleh karyawan suatu perusahaan. Aplikasi tiket IT Helpdesk berbasis web ini berkaitan dengan beberapa entitas luar, yaitu karyawan, IT Helpdesk, dan Administrator. Aplikasi tiket IT *Helpdesk* berbasis web merupakan suatu web database application yang mengolah data tiket gangguan penggunaan IT oleh karyawan saat melakukan aktivitas pekerjaan yang menggunakan IT.

Karyawan mendapatkan fasilitas untuk melaporkan kendala penggunaan IT dengan mengisi tiket ( jenis gangguan IT ) yang akan diterima oleh IT Helpdesk yang kemudian akan langsung ditangani. Dengan mempertimbangkan kemudahan navigasi dan lalu lintas data dimana karyawan harus memberikan beberapa informasi gangguan pada tiket yang digunakan. Aplikasi tiket IT Helpdesk berbasis web juga dapat digunakan oleh manager IT atau pemilik perusahaan untuk mengevaluasi penggunaan IT pada suatu perusahaan dengan melihat tiket yang dikirim oleh karyawan ketika mengalami kendala dalam penggunaan IT.

## Fungsi Produk

Perangkat Lunak tiket IT Helpdesk berbasis web ini mempunyai beberapa fungsi utama, antara lain:

### Karywan dapat mengisi tiket IT ( jenis gangguan ) yang dihadapi saat mengalami gangguan penggunaan IT yang akan langsung diterima oleh IT Helpdesk dan kemudian akan ditangani.

### Memungkinkan karyawan mengetahui solusi yang digunakan untuk mengatasi kendala dari laporan tiket yang telah ditangani oleh IT Helpdesk.

### Manager IT ( administrator ) dapat menentukan IT Helpdesk yang akan menangani tiket yang dikirim oleh karyawan sesuai kapasitas dan kemampuan IT Helpdesk sehingga dapat lebih efisien dalam proses penanganan kendala yang dialami oleh karyawan yang mengirimkan tiket.

### Manager IT ( administrator ) dan pemilik perusahaan dapat melihat rekap tiket yang digunakan oleh karywan yang dapat digunakan sebagai bahan evaluasi penggunaan IT pada perusahaan tersebut.

## Penggolongan Karakterik Pengguna

Tabel 1 Karakteristik Pengguna

| **Kategori Pengguna** | **Tugas** | **Hak Akses ke aplikasi** | **Kemampuan yang harus dimiliki** |
| --- | --- | --- | --- |
| Administrator | Mengisi data karyawan,dan jenis tiket, | Read, Insert, Update, Delete Data | Entry data karyawan dan data Tiket |
| Supervisior (Example) | Memanipulasi Data jika ada kesalahan entry dari administrator dan menentukan penanggun jawab pada setiap jenis tiket | Read, Insert, Update, Delete Data | Manupulasi data karyawan dan data Tiket |
| IT Helpdesk | Menerima tiket yang dikirim oleh karyawan dan mengisi laporan tiket yang telah ditangani | Read, insert, update | Membaca dan mangisi data tiket |
| Kartawan | Mengisi tiket dan membaca laporan tiket yang telah ditangani | Read, Insert | Membaca dan mengisi data tiket |

## Batasan Desain dan Implementasi

Pengembangan Aplikasi penjualan webini memiliki keterbatasan yaitu sebagai berikut :

### Aplikasi bersifat web base

### Aplikasipenjualantiketakan dibangun menggunakan PHP dengan database menggunakan MySQL

## Asumsi dan Ketergantungan

Asumsi-asumsi pada sistem informasi poliklinik ini adalah:

### Setiap entitas mempunyai hak akses.

### Admin bisa melihat sistem secara keseluruhan, tetapi tidak dapat merubah data-data. Admin hanya mengatur data-data user.

### IT Helpdesk mempunyai wewenang untuk melakukan menerima tiket dan mengisi laporan pada tiket yang telah dikirim oleh karyawan.

### Karyawan dapat mengisi form tiket dan dapat membaca laporan tiket yang telah ditangani.

# Kebutuhan Antarmuka Eksternal

## User Interfaces

### *Antarmuka pemakai*

Aplikasi tiket IT Helpdesk ini menggunakan antarmuka berbasis web, dan pengguna mengoperasikannya menggunakan *keyboard dan mouse* dengan sistem operasi windows dan linux serta dapat diakses menggunakan perangkat mobile seperti HP, PDA, dan lain-lain.

### *Antarmuka Perangkat Keras*

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

### *Antarmuka perangkat lunak*

Sistem Aplikasi penjualan tiket bola adalah program yang akan dibangun menggunakan bahasa PHP, *DBMS* MySQL dan akan berjalan pada Sistem Operasi Windows danlinux dan akan memakai browser browser computer dan browser perangkat mobile

### *Antarmuka komunikasi*

Sistem SSL-128 merupakan sistem sertifikasi jaringan internet untuk menjagakeamananbertransaksi, system iniakanmengenkripsi data taransaksi yang dilakukan.

## Hardware Interface

Tidak ada batasan performansi karena kemampuan perangkat keras yang digunakan telah memadai.

### *Batasan Memori*

Besarnya memory yang dibutuhkan untuk menjalankan aplikasi penjualan tiket bolo sebesar 128 MB. Besarnya kapasitas harddisk pada PC yang digunakan untuk instalasi web browser adalah sebesar 100 MB.

## Software Interface

### *Keandalan*

Aplikasi penjualan tiket bole ini dapat digunakan selama 1 X 24 jam, dengan dukungan sistem operasi Windows dan Linux yang memiliki stabilitas yang tinggi.

### *Ketersediaan*

Aplikasi penjualan bola ini dapat berjalan dan tersedia selama tidak mengalami kendala, seperti pasokan suplai tenaga listrik atau terkendalanya jaringan lokal, jaringan internet serta web server hosting aplikasi.

### *Keamanan*

Admin dan user harus melakukan log in untuk dapat mengakses sistem dan didukung keandalan dari server penyedia hosting. Untuk melakukan transaksi, pelanggan dilengkapi dengan user id yang unik dan password. Site juga dilengkapi dengan enkripsi SSL-128 yang meng-enkripsi setiap data yang dikirim melalui jaringan internet.

### *Perawatan*

Aplikasi ini dibuat secara full parameter dan dinamis. Isi site, serta informasi yang terdapat di dalamnya dapat diupdate kapan saja sesuai dengan keputusan management stadion. Management hanya perlu mengakses aplikasi sebagai administrator dan mengakses menu yang khusus disediakan untuk administrator.

## BatasanPerancangan

Aplikasi Tiket IT *Helpdesk* berbasis Web ini akan diimplementasikan pada Sistem server berbasis apache, menggunakan bahasa pemrograman PHP.

# Functional Requirement

### Fungsi user

#### Login sebagai user

Sebelum mengunakan aplikasi, user harus login untuk mendapatkan hak akses sesuai dengan jenis login yg dipilih

#### Karyawan

Karyawan akan mendapatkan sebuah user id yang unik, dengan menggunakan user id ini, karyawan dapat melakukan pengisian tiket IT *Helpdesk*. Setelah tiket IT Helpdesk telah diproses karyawan akan mendapatkan laporan penanganan oleh IT *Helpdesk*.

#### Melihat rekapitulasi tiket IT Helpdesk

Pilih menu laporan pada halaman utama untuk login karyawan dan admin, pilih waktu mulai dan batas akhir melihat laporan. Hasil laporan akan ditampilkan menggunakan table berurut sesuai waktu. Jumlah data yang tampil per halaman adalah 10 data dan dilengkapi dengan tombol navigasi next untuk melihat record berikutnya. Rekapitulasi tiket IT Helpdesk juga dapat langsung di cetak.

### *Fungsi admin*

#### pengelolaan user

admin memiliki otoritas untuk menambahkan mengedit dan menghapus user yang terdiri dari pemilik, IT *Helpdesk* dan karyawan. Untuk pengelolaan user dengan mengklik menu admin dan sub menu pengelolaan user.

# Non Functional Requirements

## Usability requirements

Aplikasi  yang dibuat di sini dirancang khusus untuk memudahkan penggguna yang memakai aplikasi tersebut. Terdapat bermacam golongan yang mengetahui tentang komputer, dan yang kita bahas disini adalah golongan awam. Ini menjadikan kita perlu mendesign tampilan yang tidak dianggap susah untuk digunakan dan dikelola.

Details :

* Tampilan yang disesuaikan dengan pengguna.
* Kemudahan mengakses aplikasi
* Dukungan bantuan on-line

## Reability requirements

Dalam pengoperasiannya, aplikasi selalu harus dalam kondisi prima. Sewaktu – waktu pengguna ingin mengedit data di jam sibuk, dan keakuratan data yang tidak disebabkan oleh human error. Menjadikan pengguna nyaman dalam penggunanan aplikasi tersebut. Secara khusus ketersediaan aplikasi ini dapat dioperasikan 24 jam per hari dan 7 hari per minggu tanpa gagal

## Safety requirements

Pada saat digunakan aplikasi ini harus aman dari berbagai serangan dari virus. Bisa saja sewaktu mengunduh informasi yang tersedia pada konten di aplikasi, informasi pada komputer pengguna itu dirusak atau dibajak oleh virus atau seseorang. Sehingga mungkin saja keselamatan orang itu juga menjadi tercancam.

## Security requirements

Secara umum, dilakukan untuk meningkatkan keamanan komputer antara lain adalah dengan membatasi akses fisik terhadap komputer, menerapkan mekanisme pada perangkat keras dan sistem operasi untuk keamanan komputer, serta membuat strategi pemrograman untuk menghasilkan program komputer yang dapat diandalkan. Akses akan dikontrol dengan username dan password. Hanya pengguna yang memiliki akun dalam sistem informasi ini yang dapat mengaksesnya, sedangkan pengguna lain tidak bisa.

Details :

* Panjang karakter password harus terdiri dari minimal 5 dan maksimal 15 karakter
* Untuk pertama kali login menggunakan username dan password yang telah ditentukan untuk masing-masing actor.

## Performance requirements

Actor atau pengguna sistem  adalah dokter, pasien dan apoteker. Pelayanan kesehatan ini memerlukan tingkat kecepatan yang tinggi untuk membuka medical record dan resep pasien. Dokter dan apoteker memerlukan data-data tersebut dengan tepat dan segera. Pasien yang sakit keras tidak mungkin menunggu lama untuk mendapat perawatan medis.

Details :

* Memonitor bagian penting dari halaman web
* Server harus bersih dari virus

## Maintainability and upgradability requirements

Pada saat maintenance atau perawatan , tim situs  melakukan perbaikan ke dalam database yang tersimpan. Selanjutnya memeriksa script yang tidak lagi terpasang di situs dan kemudian membersikannya. Tim juga memeriksa sistem yang berhubungan dengan keamanan, cek ruang hardisk, memeriksa fungsi aplikasi, melakukan back up data sampai dengan menambahkan aplikasi baru. Biasanya perawatan server berlangsung beberapa jam, dan setelah itu kita bisa menyaksikan perubahan web seperti adanya aplikasi baru dan tampilan web yang bertambah segar.

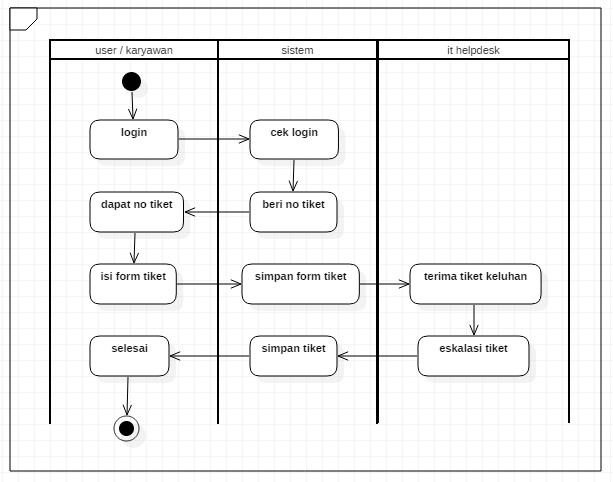
Details :

* Maintenance biasanya dilakukan 2 minggu sekali bahkan 1 bulan sekali
* Update database
* Pada saat maintenance, server akan dimatikan dan user tidak bisa masuk ke web
* Mengupgrade desain web jika perlu

# Dokumen Desain Aplikasi

## Desain Fungsional

Pada sistem yang akan diusulkan oleh penulis, penulis bertujuan untuk memudahkan karyawan dan *helpdesk*, dalam memberikan penanganan ataupun tindakan terhadap masalah yang dihadapi oleh karyawan.



***Activity Diagram* Sistem Proses**

## Perancangan Basis Data

Basis data telah mempunyai bentuk normal selanjutnya dijelaskan dalam spesifikasi basis data yang meliputi *file*, media penyimpanan yang digunakan, isi yang disimpan, *primary key,* dan struktur *file.*

### ERD *(Entity Relationship Diagram)*

Berikut adalah ERD dari transaksi pemesanan tiketing IT Helpdesk pada sistem usulan.



***Entity Relationship Diagram* Sistem Usulan**

* **Spesifikasi Basis Data**

Tabel–tabel yang dibutuhkan untuk menampung data kebutuhan sistem antara lain :

1. Tabel *User*

Tabel ini berfungsi untuk menampung data *user* yang mengelola data kebutuhan sistem. *Primary Key* (PK) dari tabel ini adalah *username.* Untuk lebih jelasnya ditunjukan pada tabel berikut ini :

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama** | ***Type*** | ***Size*** |
| 1. | *id\_user* (PK) | *BigInt* | 12 |
| 2. | *nm\_user* | *Varchar* | 100 |
| 3. | *user\_name* | *Varchar* | 25 |
| 4. | *almt\_user* | *Text* |  |
| 5. | *jns\_kelamin\_user* | *Char* | 1 |
| 6. | *password\_user* | *Varchar* | 60 |
| 7. | *level\_user* | *Int* | 1 |
| 8. | *stts\_user* | *Int* | 1 |
| 9. | *file\_name\_user* | *Text* |  |

**Tabel Spesifikasi *User***

1. Tabel Karyawan

Tabel ini berfungsi untuk menampung data karyawan. *Primary Key* (PK) dari tabel ini adalah nik*.* Untuk lebih jelasnya ditunjukan pada tabel berikut ini :

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama** | ***Type*** | ***Size*** |
| 1. | nik(PK) | *Varchar* | 25 |
| 2. | nm\_karyawan | *Varchar* | 100 |
| 3. | *user\_name\_* karyawan | *Varchar* | 25 |
| 4. | almt\_ karyawan | *Text* |  |
| 5. | jns\_kelamin\_ karyawan | *Char* | 1 |
| 6. | *password\_* karyawan | *Varchar* | 60 |
| 7. | stts\_ karyawan | *Int* | 1 |
| 8. | *file\_name\_* karyawan | *Text* |  |

**Tabel Spesifikasi Karyawan**

1. TabelJenis Tiket

Tabel ini berfungsi untuk menampung data jenis tiket. *Primary Key* (PK) dari tabel ini adalah id\_jns\_tiket*.* Untuk lebih jelasnya ditunjukan pada tabel berikut ini :

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama** | ***Type*** | ***Size*** |
| 1. | id\_jns\_tiket (PK) | *BigInt* | 12 |
| 2. | nm\_jns\_tiket | *Varchar* | 100 |

**Tabel Spesifikasi Jenis Tiket**

1. TabelTiket

Tabel ini berfungsi untuk menampung data tiket. *Primary Key* (PK) dari tabel ini adalah id\_tiket, dan *Foreign Key* (FK) adalah id\_user sebagai penghubung atau relasi terhadap tabel user, id\_jns\_tiket sebagai penghubung atau relasi ke tabel jenis tiket, nik sebagai penghubung atau relasi ke tabel karyawan*.* Untuk lebih jelasnya ditunjukan pada tabel berikut ini :

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama** | ***Type*** | ***Size*** |
| 1. | id\_tiket(PK) | *BigInt* | 12 |
| 2. | id\_jns\_tiket | *BigInt* | 12 |
| 3. | *id\_user* | *BigInt* | 12 |
| 4. | nik | *Varchar* | 25 |
| 5. | Judul\_tiket | *Varchar* | 100 |
| 6. | tgl\_buka | *Datetime* |  |
| 7. | tgl\_tutup | *Datetime* |  |
| 5. | stts\_tiket | *Int* | 1 |

**Tabel Spesifikasi Tiket**

1. TabelTiket Detail

Tabel ini berfungsi untuk menampung data tiket detail. *Primary Key* (PK) dari tabel ini adalah id\_tiket\_dtl, dan untuk *Foreign Key* (FK) id\_tiket sebagai penghubung ke table tiket. Untuk lebih jelasnya ditunjukan pada tabel berikut ini :

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama** | ***Type*** | ***Size*** |
| 1. | id\_tiket\_dtl(PK) | *BigInt* | 12 |
| 2. | id\_tiket (FK) | *BigInt* | 12 |
| 3. | tgl\_tiket | *Datetime* |  |
| 4. | Isi\_tiket | *Text* |  |

**Tabel Spesifikasi Tiket Detail**

1. TabelTindakan

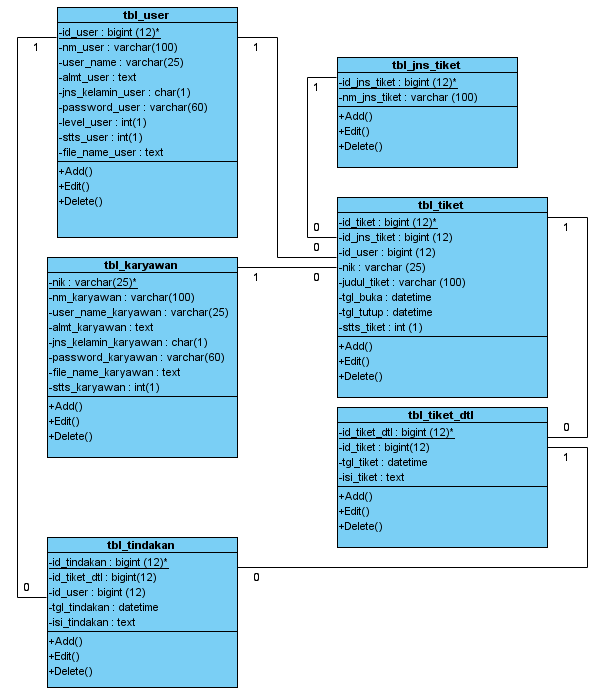
Tabel ini berfungsi untuk menampung data tindakan yang dilakukan oleh user *heldesk* terhadap permasalahan yang dihadapi oleh karyawan. *Primary Key* (PK) dari tabel ini adalah id\_tindakan, *Foreign Key* (FK) adalah id\_tiket\_dtl sebagai penghubung atau relasi terhadap tabel tiket\_dtl dan *id\_user* sebagai penghubung atau relasi terhadap tabel *user.* Untuk lebih jelasnya ditunjukan pada tabel berikut ini :

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama** | ***Type*** | ***Size*** |
| 1. | id\_tindakan(PK) | *BigInt* | 12 |
| 2. | id\_tiket\_dtl (FK) | *BigInt* | 12 |
| 3. | *id\_user* (FK) | *BigInt* | 12 |
| 4. | tgl\_tindakan | *Datetime* |  |
| 5. | *Isi\_tindakan* | *Text* |  |

**Tabel Spesifikasi Tindakan**

* **Class Diagram**

Dari daftar objek tabel di atas, struktur objek sistem yang diajukan dapat digambarkan *class diagram* sebagai berikut :



***Class Diagam* usulan**

*Class Diagram* Menggambarkan struktur objek sistem. Diagram ini menunjukan *class diagram* yang menyusun sistem dan hubungan antara *class object*tersebut. *Class diagram* adalah model statis yang menggambarkan struktur dan deskripsi *class* serta hubungannya antar *class*. *Class diagram* mirip ER-diagram pada perancangan *database*, bedanya pada ER-diagram tidak terdapat operasi atau *method* tapi hanya atribut*. Class* terdiri dari nama kelas, atribut, relasi dan operasi atau *method*.

1. Kelas

Seperti user, jenis tiket, tiket, karyawan, tindakan.

1. Atribut

Seperti id\_user, nm\_user, id\_tiket, judul\_tiket, id\_tindakan.

1. Operasi

Seperti add, edit, delete

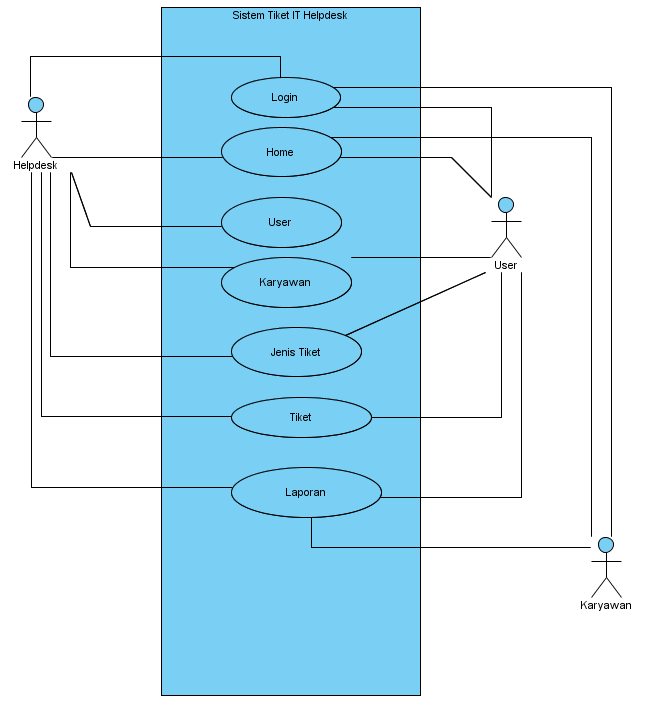
1. Relasi

Seperti garis penghubung antar kelas yang dapat menghubungkan kelas satu dengan yang lain sesuai dengan fungsinya.

## Perancangan UML (*Unifiled Modelling Language*)

* ***Use Case Diagram***

*Use case diagram* menggambarkan apa saja aktifitas yang dilakukan didalam sistem dari sudut pandang pengamatan luar. *Use case diagram* dekat kaitannya dengan kejadian (*scenario*) yang merupakan contoh yang terjadi ketika seseorang berinteraksi dengan sistem. Adapun *use case diagram* pada sistem ini dapat dilihat pada gambar *use case diagram* di bawah ini.



***Use Case diagram***

1. *Use case* : *Login*

Aktor : Administrator, User, Karyawan

Deskripsi :

1. Administrator, User, Karyawan membuka aplikasi
2. Kemudian Administrator, User, Karyawan dapat memasukan data *username* dan *password*
3. Administrator, User, Karyawan menekan tombol *login* untuk masuk ke *Menu* Home
4. *Use case* : *Menu Home*

Aktor : Administrator, User, Karyawan

Deskripsi : Menampilkan halaman utama website

1. *Use case* : *User*

Aktor : Administrator

Deskripsi :

1. Administrator memilih *Menu User*
2. Kemudian Administrator menambah, mengubah atau menghapus data *user*
3. Kemudian Administrator menyimpan perubahan dari data *user.*
4. *Use case* : Karyawan

Aktor : Administrator, User

Deskripsi :

1. Administrator, User memilih *Menu* Karyawan
2. Kemudian Administrator, User menambah data karyawan
3. Kemudian Administrator, User merubah data karyawan
4. Kemudian Administrator, User menyimpan data karyawan
5. Kemudian Administrator, User menghapus data karyawan
6. *Use case* : Jenis Tiket

Aktor : Administrator, User

Deskripsi :

1. Administrator, user memilih *Menu* Jenis Tiket
2. Administrator, user menambah atau merubah data jenis tiket
3. Kemudian Administrator, user menyimpan data jenis tiket
4. Kemudian Administrator, user menghapus data jenis tiket
5. *Use case* : Tiket

Aktor : Administrator, User, Karyawan

Deskripsi :

1. Administrator, User, Karyawan memilih *Menu* Jenis Proyek
2. Karyawan menambah atau merubah data tiket
3. Kemudian Administrator memilih *user* untuk melakukan tindakan terhadap tiket yang dibuat oleh karyawan
4. Kemudian User melakukan tindakan terhadap masalah yang dialami oleh karyawan berdasarkan tiket yang dibuat oleh karyawan.
5. *Use case* : Laporan

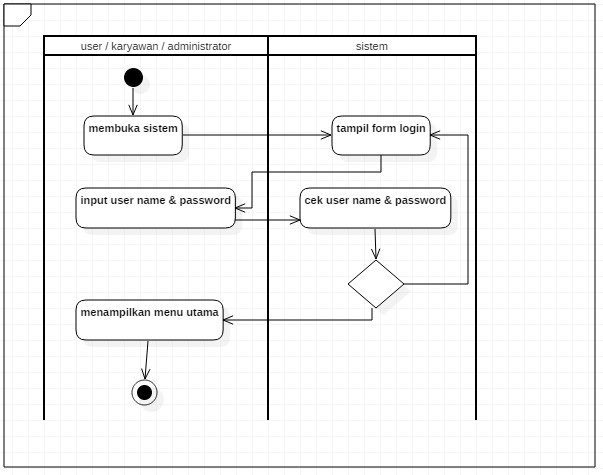
Aktor : Administrator, User, Karyawan

Deskripsi :

1. Administrator, User, Karyawan memilih *Menu* Laporan
2. Kemudian Administrator, User, Karyawan memilih tanggal untuk periode laporan

* ***Activity Diagram***

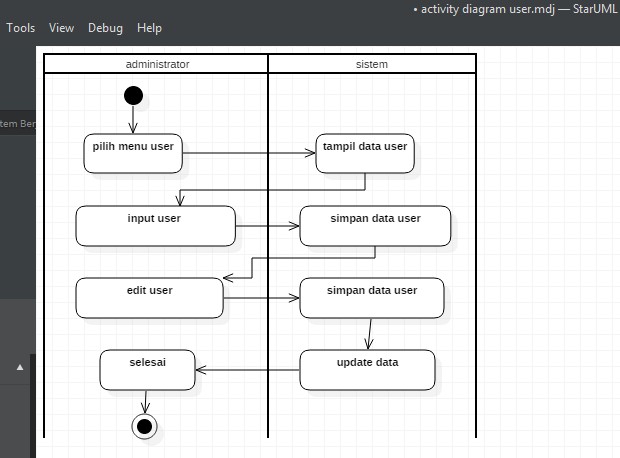
1. *Activity Diagram Login*



***Activity Diagram Login***

Dari diagram diatas dapat dijelaskan bahwa untuk dapat menggunakan sistem, administrator, user dan karyawan harus melakukan *login* terlebih dahulu. Administrator, user dan karyawan memasukan *username* dan *password,* jika data yang dimasukan benar maka sistem akan masuk ke *menu* utama. Kemudian jika *username* dan *password* yang dimasukan salah maka akan menampilkan pesan bahwa *username* dan *password* salah.

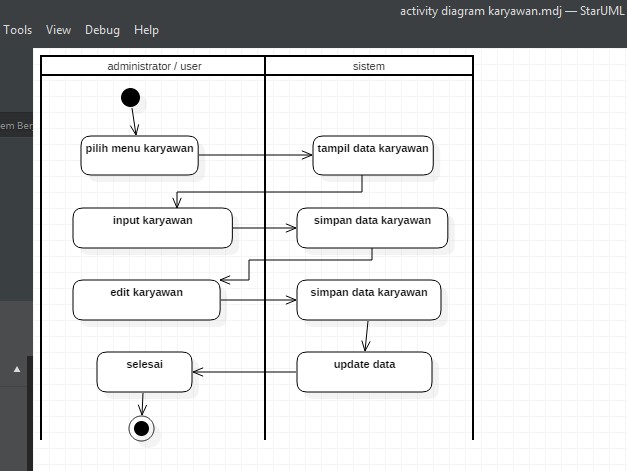
1. *Activity Diagram User*



***Activity Diagram User***

Dari diagram diatas dapat dijelaskan bahwa administrator dapat menambah, mengubah data *user* kemudian menyimpannya data kedalam *database*, dan administrator juga bias menghapus data *user* dari *database*.

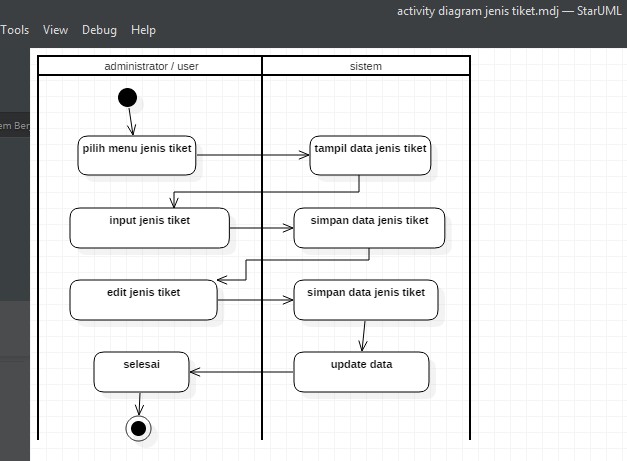
1. *Activity Diagram* Karyawan



***Acvtivity Diagram* Karyawan**

Dari diagram diatas dapat dijelaskan bahwa administrator dan *user* dapat menambah, mengubah data karyawan kemudian menyimpannya data kedalam *database*, dan administrator dan *user* juga bisa menghapus data karyawan dari *database*.

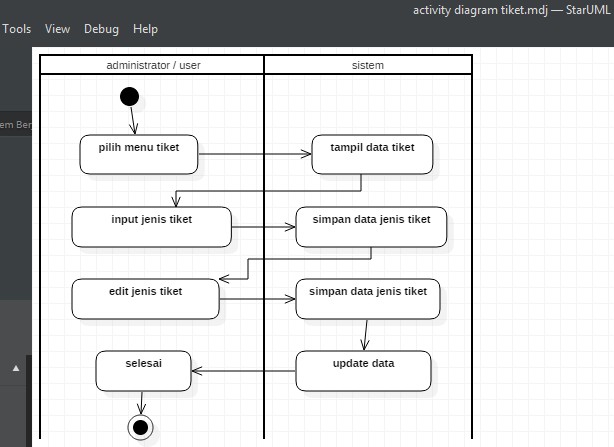
1. *Activity Diagram* Jenis Tiket



***Activity Diagram* Jenis Tiket**

Dari diagram diatas dapat dijelaskan bahwa administrator dan *user* dapat menambah, mengubah data jenis tiket kemudian menyimpannya data kedalam *database*, dan administrator, *user* juga bisa menghapus data jenis tiketdari *database*.

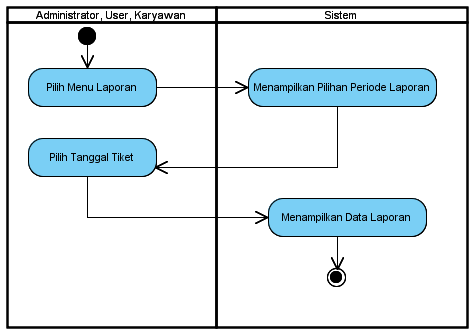
1. *Activity Diagram* Tiket



***Activity Diagram* Tiket**

Dari diagram diatas dapat dijelaskan bahwa administrator dapat menambah, mengubah data jenis proyek kemudian menyimpannya data kedalam *database*, dan administrator juga bisa menghapus data jenis proyekdari *database*.

1. *Activity Diagram* Laporan



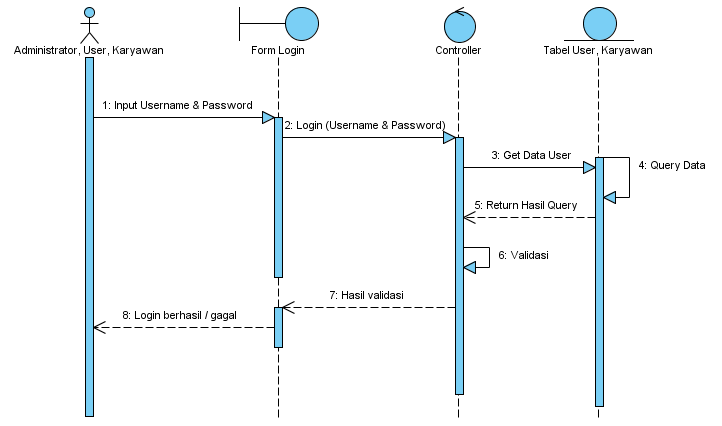
***Activity Diagram* Laporan**

Dari diagram diatas dapat dijelaskan bahwa administrator, *user* dan karyawandapat melihat data laporan berdasarkan *periode* tanggal tiket.

* + - ***Sequence Diagram***

*Sequence diagram* menggambarkan interaksi antar obyek di dalam dan di sekitar sistem (termasuk pengguna, *display* dan sebagainya).

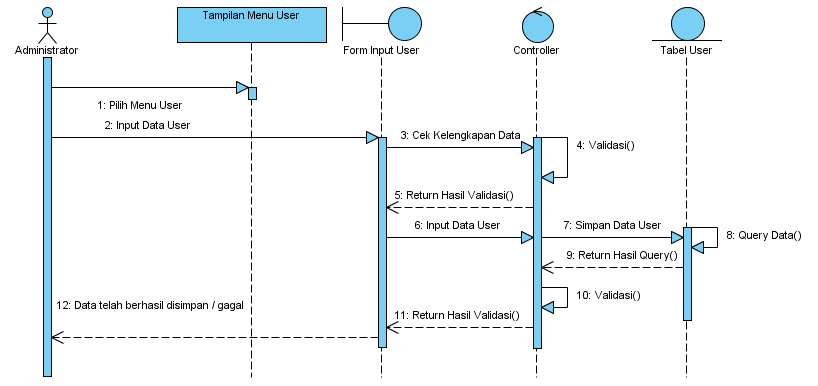
1. *Sequence Diagram Login*



***Sequence Diagram Login***

Dari gambar diatas dapat dijelaskan bahwa pada *Menu* ini kegiatan administrator, user dan karyawan adalah memasukan *username* dan *password,* jika data yang dimasukan benar maka sistem akan masuk ke *Menu* utama (Form utama). Kemudian jika *username* dan *password* yang dimasukan salah maka akan menmpilkan pesan bahwa *username* dan *password* salah.

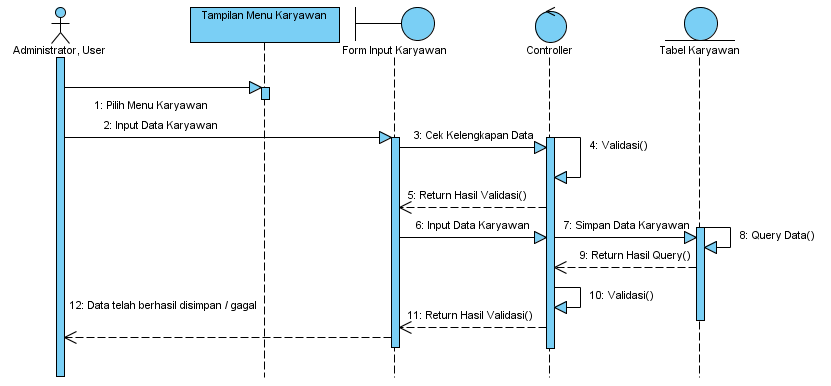
1. *Sequence Diagram User*



***Sequence Diagram User***

Dari diagram diatas dapat dijelaskan bahwa administrator dapat menambah, mengubah data dan menyimpan data pada *database*, dan juga dapat menghapus data *user* dari *database*

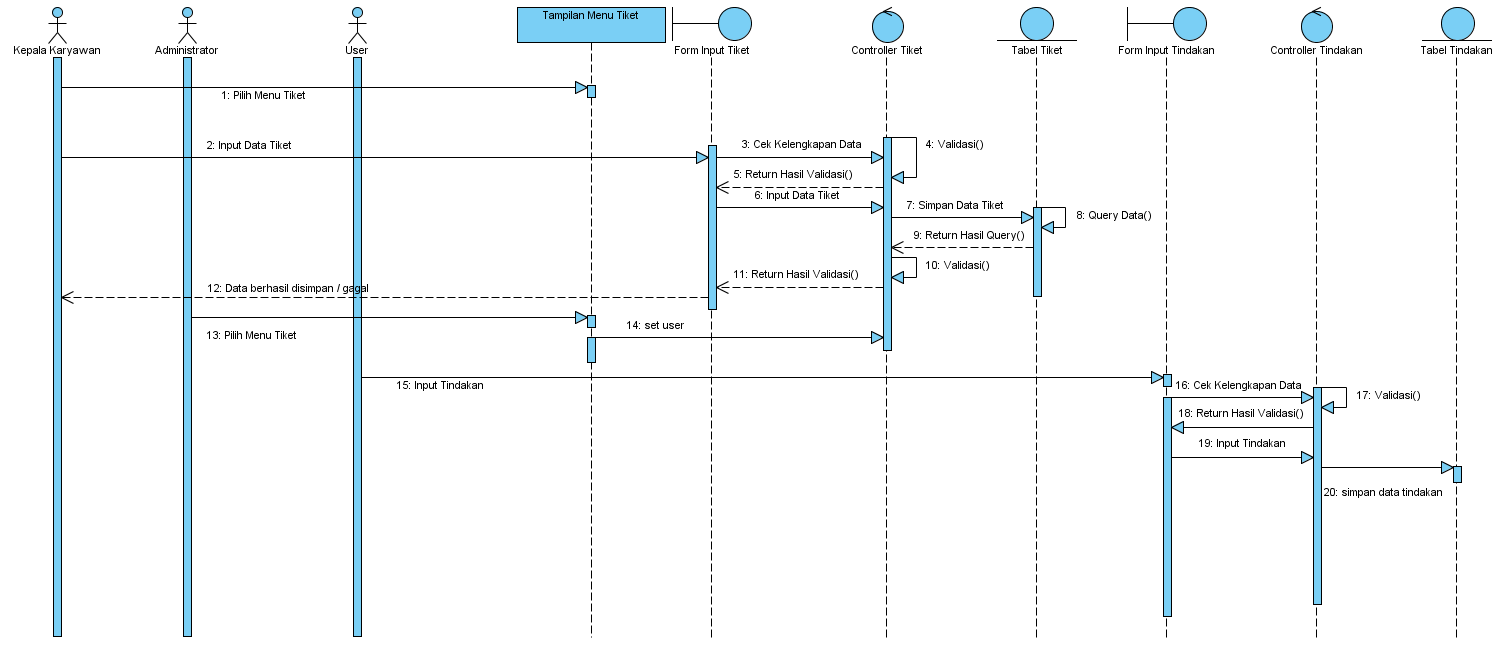
1. *Sequence Diagram* Karyawan



***Sequence Diagram* Karyawan**

Berdasarkan gambar diatas dapat dijelaskan bahwa administrator dan user dapat menambah, merubah data karyawandan menyimpannya kedalam *database,* dan juga dapat menghapus data karyawandari *database.* Semua tindakan atau aktifitas tersebut terintegrasi dengan data tabel karyawanyang ada pada *database* sistem.

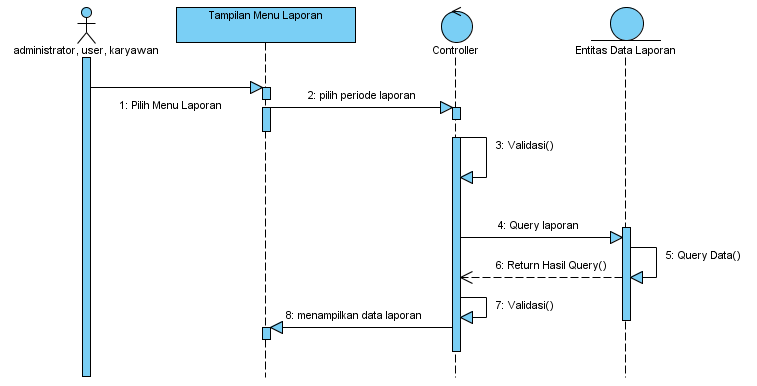
1. *Sequence Diagram* Tiket



***Sequence Diagram* Jenis Tiket**

Berdasarkan gambar diatas dapat dijelaskan bahwa karyawan membuat tiket terlebih dahulu yang kemudian oleh administrator dipilihkan seorang *user* untuk melakukan tindakan atau penanganan terhadap permasalahan karyawan*.* Semua tindakan atau aktifitas tersebut terintegrasi dengan data tabel tiket dan table tindakanyang ada pada *database* sistem.

1. *Sequence Diagram* Laporan



***Sequence Diagram* Laporan**

Berdasarkan gambar diatas dapat dijelaskan bahwa administrator, user dan karyawan dapat melihat data laporan*.* Semua tindakan atau aktifitas tersebut terintegrasi dengan data tabel tiket dan tabel tindakanyang ada pada *database* sistem.

* + - **Perancangan Antarmuka**

Antarmuka pengguna *(User Interface)* merupakan aspek sistem atau program yang dapat dilihat, komponen yang digunakan dan perintah-perintah yang digunakan pemakai untuk mengendalikan operasi dan masukan data.

* 1. Tampilan *Login*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  | Header | | | | |  |
|  |  | LOGIN |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  | Username | | |  |  |
|  |  | Password | | |  |  |
|  |  |  |  |  |  |  |
|  |  | Login |  | Reset |  |  |
|  |  |  |  |  |  |  |
|  | Footer | | | | |  |
|  |  |  |  |  |  |  |

**Tampilan *Login***

Pada gambar diatas dapat dilihat, komponen-komponen yang digunakan dalam perancangan pada halaman *login* sebagai berikut:

1. *Static Text* terdiri dari:
2. *Static Text header* untuk menampilkan teks *header*
3. *Static Text footer* untuk menampilkan teks *footer*
4. *Static Text login* untuk menampilkan teks *login*
5. *Static Text username* untuk menampilkan teks *username*
6. *Static Text password* untuk menampilkan teks *password*
7. *Edit Text* terdiri dari:
8. *Input Text username* untuk mengisikan *username*
9. *Input Text password* untuk mengisikan *password*
10. *Input Button* terdiri dari
11. *Input Button* *login* untuk masuk ke *Menu* utama
12. *Input Button* *reset* untuk membersihkan data pada *form login*
13. Rancangan Antarmuka *Menu Home*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  | Header | | | | |  |
|  |  |  |  |  |  |  |
|  | Menu | | | | |  |
|  |  |  |  |  |  |  |
|  |  |  | | |  |  |
|  |  | Text | | |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Footer | | | | |  |
|  |  |  |  |  |  |  |

**Rancangan Antarmuka *Menu Home***

Pada gambar diatas rancangan pada *Menu home* yang ditampilkan pada sistem. Komponen-komponen yang digunakan sebagai berikut:

1. *Static Text* terdiri dari:
2. *Static Text logout* untuk menampilkan logout
3. *Static Text* nama perusahaan untuk menampilkan nama perusahaan
4. *Static Text* menu untuk menampilkan menu
5. *Static Text footer* untuk menampilkan *footer*
6. Rancangan Antarmuka *Menu User*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Header | | | | |  |
|  |  |  |  |  |  |  |
|  | Menu | | | | |  |
|  |  |  |  |  |  |  |
|  | + Add New |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Daftar User | | | | |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Footer | | | | |  |
|  |  |  |  |  |  |  |

**Rancangan Antarmuka *Menu User***

Pada gambar diatas rancangan pada *Menu User* yang ditampilkan pada sistem. Komponen-komponen yang digunakan sebagai berikut:

1. *Static Text* terdiri dari:
2. *Static Text logout* untuk menampilkan logout
3. *Static Text* header untuk menampilkan header
4. *Static Text* menu untuk menampilkan menu
5. *Static Text footer* untuk menampilkan *footer*
6. *Ui table* terdiri dari:
7. *Ui table*1 untuk menampilkan data *user* yang ada pada *database*
8. *Input button* terdiri dari:
9. *Input button Add New* untuk menampilkan *form input user.*
10. *Input button Edit* untuk menampilkan *form edit user.*
11. *Input button Delete* menghapus data *user* dari *database.*
12. Rancangan Antarmuka *Menu* Karyawan

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Header | | | | |  |
|  |  |  |  |  |  |  |
|  | Menu | | | | |  |
|  |  |  |  |  |  |  |
|  | + Add New |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Daftar Karyawan | | | | |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Footer | | | | |  |
|  |  |  |  |  |  |  |

**Rancangan Antarmuka *Menu* Karyawan**

Pada gambar diatas rancangan pada *Menu* Karyawan yang ditampilkan pada sistem. Komponen-komponen yang digunakan sebagai berikut:

1. *Static Text* terdiri dari:
2. *Static Text logout* untuk menampilkan logout
3. *Static Text header* untuk menampilkan *header*
4. *Static Text* menu untuk menampilkan menu
5. *Static Text footer* untuk menampilkan *footer*
6. *Ui table* terdiri dari:
7. *Ui table* untuk menampilkan data karyawan yang ada pada *database*
8. *Input button* terdiri dari:
9. *Input button Add New* untuk menampilkan *form input* karyawan*.*
10. *Input button Edit* untuk menampilkan *form edit* karyawan*.*
11. *Input button Delete* menghapus data karyawan dari *database.*
12. Rancangan Antarmuka *Menu* Jenis Tiket

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Header | | | | |  |
|  |  |  |  |  |  |  |
|  | Menu | | | | |  |
|  |  |  |  |  |  |  |
|  | + Add New |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Daftar Jenis Tiket | | | | |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Footer | | | | |  |
|  |  |  |  |  |  |  |

**Rancangan Antarmuka *Menu* Jenis Tiket**

Pada gambar diatas rancangan pada *Menu* Jenis Tiket yang ditampilkan pada sistem. Komponen-komponen yang digunakan sebagai berikut:

1. *Static Text* terdiri dari:
2. *Static Text logout* untuk menampilkan logout
3. *Static Text header* untuk menampilkan *header*
4. *Static Text* menu untuk menampilkan menu
5. *Static Text footer* untuk menampilkan *footer*
6. *Ui table* terdiri dari:
7. *Ui table* untuk menampilkan data jenis tiket yang ada pada *database*
8. *Input button* terdiri dari:
9. *Input button Add New* untuk menampilkan *form input* jenis tiket*.*
10. *Input button Edit* untuk menampilkan *form edit* jenis tiket*.*
11. *Input button Delete* menghapus data jenis tiket dari *database.*
12. Rancangan Antarmuka *Menu* Tiket

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Header | | | | |  |
|  |  |  |  |  |  |  |
|  | Menu | | | | |  |
|  |  |  |  |  |  |  |
|  | + Add New |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Daftar Jenis Proyek | | | | |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  | Footer | | | | |  |
|  |  |  |  |  |  |  |

**Rancangan Antarmuka *Menu* Tiket**

Pada gambar diatas rancangan pada *Menu* Tiket yang ditampilkan pada sistem. Komponen-komponen yang digunakan sebagai berikut:

1. *Static Text* terdiri dari:
2. *Static Text logout* untuk menampilkan logout
3. *Static Text header* untuk menampilkan *header*
4. *Static Text* menu untuk menampilkan menu
5. *Static Text footer* untuk menampilkan *footer*
6. *Ui table* terdiri dari:
7. *Ui table* untuk menampilkan data tiket yang ada pada *database*
8. *Input button* terdiri dari:
9. *Input button Add New* untuk menampilkan *form input* tiket*.*
10. *Input button Edit* untuk menampilkan *form edit* tiket*.*
11. *Input button Delete* menghapus data tiket dari *database.*
12. Rancanagn Antarmuka *Menu* Laporan

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | Header | | | | | | | |  |
|  |  |  |  |  |  |  |  |  |  |
|  | Menu | | | | | | | |  |
|  |  |  |  |  |  |  |  |  |  |
|  | Tgl. Tiket : |  | ss/d |  |  | Vview |  | pPrint |  |
|  |  |  |  |  |  |  |  |  |  |
|  | Daftar Laporan | | | | | | | |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | Footer | | | | | | | |  |
|  |  |  |  |  |  |  |  |  |  |

**Rancangan Antarmuka *Menu* Laporan**

Pada gambar diatas rancangan pada *Menu* Laporan yang ditampilkan pada sistem. Komponen-komponen yang digunakan sebagai berikut:

1. *Static Text* terdiri dari:
2. *Static Text logout* untuk menampilkan logout
3. *Static Text header* untuk menampilkan *header*
4. *Static Text* menu untuk menampilkan menu
5. *Static Text footer* untuk menampilkan *footer*
6. *Static Text* Tanggal Pengiriman untuk menampilkan Tanggal Pengiriman
7. *Ui table* terdiri dari:
8. *Ui table* untuk menampilkan data tiket dan tindakan yang ada pada *database*
9. *Input Text* terdiri dari:
10. *Input text* tanggal awal untuk mengisi tanggal awal*.*
11. *Input text* tanggal akhir untuk mengisi tanggal akhir*.*
12. *Input button* terdiri dari:
13. *Input button View* untuk menampilkan data laporan tiket dan tindakan*.*
14. *Input button Print* untuk mencetak laporan tiket dan tindakan*.*

## Implementasi Sistem

Tahapan implementasi sistem merupakan tahap meletakan sistem agar siap untuk dioperasikan. Implementasi bertujuan untuk mengkonfirmasi modul-modul perancangan sehingga pengguna bisa memberi masukan kepada pengembang sistem.

* **Implementasi Perangkat Lunak**

Perangkat lunak (*software*) adalah piranti atau program yang terpasang dikomputer yang berguna untuk pengolahan data dan menunjang perangkat keras (*hardware*). Yang bertujuan untuk memudahkan pengguna (*user*) untuk mengeksekusi perintah – perintah tertentu sesuai dengan fungsi perangkat lunak itu sendiri. Adapun perangkat lunak yang dibutuhkan untuk mendukung pembuatan program aplikasi ini adalah sebagai berikut:

1. Sistem Operasi Windows 7
2. Browser seperti Chrome atau modzilla
3. Xampp
4. Notepad++

* **Implementasi Perangkat Keras**

Perangkat keras (*hardware*) adalah semua bagian fisik komputer, dan dibedakan dengan data yang berada didalamnya atau yang beroperasi didalamnya, dan dibedakan dengan perangkat lunak (*software*) yang menyediakan instruksi untuk perangkat keras dalam meyelesaikan tugasnya. Adapun perangkat keras yang digunakan untuk mendukung pembuatan program aplikasi ini adalah sebagai berikut:

1. Processor : Intel(R) Core™i3-2310M CPU @2.10GHz
2. Monitor : 14”
3. Memory : 2Gb DDR3
4. HardDisk : 500 Gb
5. VGA Card : NVIDIA GeForce GT450M CUDA™ 2GB

# Implementasi Antarmuka

## Tampilan Antarmuka

1. Implementasi Halaman *Login*



**Implementasi Halaman *Login***

Gambar diatas adalah tampilan *menu login*, dimana pada *menu* ini administrator, *user* dan karyawan memasukan *username* dan *password,* jika data yang dimasukan benar maka sistem akan masuk ke *menu* utama. Kemudian jika *username* dan *password* yang dimasukan salah maka akan menmpilkan pesan bahwa *username* dan *password* salah.

1. Implementasi *Menu Home*



**Rancangan Implementasi *Menu Home***

Pada gambar di atas adalah tampilan *menu* utama pada sistem Tiket IT *Helpdesk*. *Menu* ini merupakan tampilan awal program, pada *menu* ini pengguna aplikasi menentukan aktifitas sesuai dengan *submenu* yang ada didalamnya, antaralain :

1. *Menu Home*
2. *Menu User*
3. *Menu* Karyawan
4. *Menu* Jenis Tiket
5. *Menu* Tiket
6. *Menu* Laporan
7. Implementasi *Menu User*



**Implementasi *Menu User***

Pada gambar diatas adalah tampilan *menu user*, pada *menu* ini administrator dapat menambah data *user*, mengubah data *user*, dan juga menghapus data *user*. Administrator juga bias merubah level *user* menjadi administrator dan menyimpan data *user* kedalam *database.*

1. Implementasi *Menu* Karyawan



**Implementasi *Menu* Karyawan**

Pada gambar diatas adalah tampilan *menu* karyawan, pada *menu* ini administrator dan user dapat menambah data karyawan, mengubah data karyawan, dan juga menghapus data karyawan, dan menyimpan data karyawan kedalam *database.*

1. Implementasi *Menu* Jenis Tiket



**Implementasi *Menu* Jenis Tiket**

Pada gambar diatas adalah tampilan *menu* jenis tiket, pada *menu* ini administrator dan user dapat menambah data jenis tiket, mengubah data jenis tiket, dan juga menghapus data jenis tiket, dan menyimpan data jenis tiket kedalam *database.*

1. Implementasi *Menu* Tiket



**Implementasi *Menu* Tiket**

Pada gambar diatas adalah tampilan *menu* tiket, pada *menu* ini karyawan mambuat tiket, kemudian oleh administrator akan dipilihkan seorang *user* untuk melakukan tindakan atau penanganan terhadap tiket atau permasalahan yang sedang dihadapi oleh karyawan. *User* melakukan tindakan atau penanganan terhadap tiket yang dibuat oleh karyawan dan membuat tindakan ataupun penanganan yang dilakukan terhadap tiket yang dibuat oleh karyawan*.*

1. Implementasi *Menu* Laporan



**Implementasi *Menu* Laporan**

Pada gambar diatas adalah tampilan *menu* laporan, pada *menu* ini administrator, user dan karyawan dapat melihat data laporan tiket dan tindakan yang terlebih dalu memasukan tanggal tiket untuk menampilkan data laporan tiket dan tindakan sesuai dengan tanggal tiket*.*

## Pengujian Sistem

Sebelum program diterapkan, maka program harus bebas dari kesalahan dan program harus dites dengan menggunakan pengujian *black box* untuk menemukan yang mungkin terjadi seperti kesalahan dalam bahasa, kesalahan dalam memasukan data dan kesalahan logika program.

* **Pengujian Metode *Black Box***

Pengujian adalah proses dimana sistem di uji coba kelayakan dan kualitas dari perangkat lunak yang dibuat. Ada dua jenis pengujian dalam tahap ini yaitu secara *white box* dan *black box*. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian *black box* memungkinkan perekayasa perangkat lunak mendapatkan serangkai kondisi input yang sepenuhnya semua persyaratan fungsional untuk suatu program. Sedangakan pengujian dengan *white box* berisi mengenai struktur dari isi program berupa *coding* Dalam pengujian ini penulis menggunakan pengujian *black Box*.

1. Tabel Pengujian Metode *Black Box*

**Tabel 4.1 Pengujian Menu *Login***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Uji Benar** | | | | |
| **Pengujian** | **Proses** | **Data** | **Pengamatan** | **Kesimpulan** |
| Klik *login* | Masuk ke halaman menu *login* | Masukan data *Username* dan *password* | Berhasil *login* | Diterima |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Uji Salah** | | | | |
| **Pengujian** | **Proses** | **Data** | **Pengamatan** | **Kesimpulan** |
| Klik *login* | Masuk ke halaman menu *login* | *Username* dan *password* tidak di isi atau salah | Tidak berhasil *login* | Ditolak |

**Tabel Pengujian *User***

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Uji Benar** | | | | |
| **Pengujian** | **Proses** | **Data** | **Pengamatan** | **Kesimpulan** |
| Klik *Add New* | Masuk ke *form input user* | Masukan data Nama, alamat, *username, password* | Data berhasil disimpan | Diterima |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Uji Salah** | | | | |
| **Pengujian** | **Proses** | **Data** | **Pengamatan** | **Kesimpulan** |
| Klik *Add New* | Masuk ke *form input user* | Ada data yang tidak di isi | Data tidak berhasil disimpan | Ditolak |

**Tabel Pengujian Menu Karyawan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Uji Benar** | | | | |
| **Pengujian** | **Proses** | **Data** | **Pengamatan** | **Kesimpulan** |
| Klik *Add New* | Masuk ke *form input* karyawan | Masukan data NIK, Nama, alamat, *username, password* | Data berhasil disimpan | Diterima |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Uji Salah** | | | | |
| **Pengujian** | **Proses** | **Data** | **Pengamatan** | **Kesimpulan** |
| Klik *Add New* | Masuk ke *form input* karyawan | Ada data yang tidak di isi | Data tidak berhasil disimpan | Ditolak |

**Tabel Pengujian Menu Jenis Tiket**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Uji Benar** | | | | |
| **Pengujian** | **Proses** | **Data** | **Pengamatan** | **Kesimpulan** |
| Klik *Add New* | Masuk ke *form input* Jenis Tiket | Masukan data Nama Jenis Tiket | Data berhasil disimpan | Diterima |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Uji Salah** | | | | |
| **Pengujian** | **Proses** | **Data** | **Pengamatan** | **Kesimpulan** |
| Klik *Add New* | Masuk ke *form input* Jenis Tiket | Ada data yg tidak disi | Data tidak berhasil disimpan | Ditolak |

**Tabel Pengujian Laporan**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Uji Benar** | | | | |
| **Pengujian** | **Proses** | **Data** | **Pengamatan** | **Kesimpulan** |
| Klik Laporan | Masuk Halaman Laporan | Masukan data Tanggal Tiket | Menampilkan Data laporan tiket dan tindakan sesuai dengan tanggal tiket yang dimasukan | Diterima |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data Uji Salah** | | | | |
| **Pengujian** | **Proses** | **Data** | **Pengamatan** | **Kesimpulan** |
| Klik Laporan | Masuk Halaman Laporan | Masukan data Tanggal Tiket | Menampilkan Data laporan tiket dan tindakan tidak sesuai dengan tanggal tiket yang dimasukan | Ditolak |