**DPPL-01**

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

Aplikasi Tiket Travel

untuk:

Shuttle Travel

Dipersiapkan oleh:

Salsabila Putri J ( 1301170106 )

Harry Satria ( 1301172575 )

Ryan Pratama P ( 130114222 )

Zulfa Nurfauziah ( 1301174269 )

Program Studi Informatika

Fakultas Informatika

Jl. Telekomunikasi 1, Dayeuhkolot Bandung

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Prodi S1- Informatika**  **Universitas Telkom** | Nomor Dokumen | | Halaman |
| *DPPL-01 <xx:no grp>* | |  |
| Revisi |  |  |

DAFTAR PERUBAHAN

|  |  |
| --- | --- |
| Revisi | Deskripsi |
| A |  |
| B |  |
| C |  |
| D |  |
| E |  |
| F |  |
| G |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| INDEX  TGL | - | A | B | C | D | E | F | G |
| Ditulis oleh |  |  |  |  |  |  |  |  |
| Diperiksa oleh |  |  |  |  |  |  |  |  |
| Disetujui oleh |  |  |  |  |  |  |  |  |

Daftar Halaman Perubahan

|  |  |  |  |
| --- | --- | --- | --- |
| Halaman | Revisi | Halaman | Revisi |
|  |  |  |  |

Daftar Isi

[**1. Pendahuluan**](#_gjdgxs) **5**

[Tujuan Penulisan Dokumen](#_30j0zll) 6

[Lingkup Masalah](#_1fob9te) 6

[Definisi dan Istilah](#_3znysh7) 6

[Referensi](#_2et92p0) 6

[Sistematika Pembahasan](#_tyjcwt) 6

[**Deskripsi Perancangan Global**](#_3dy6vkm) **6**

[Rancangan Lingkungan Implementasi](#_1t3h5sf) 7

[Deskripsi Arsitektural](#_4d34og8) 7

[Deskripsi Komponen](#_2s8eyo1) 7

[**Perancangan Rinci**](#_3rdcrjn) **8**

[Realisasi Use Case](#_26in1rg) 8

[Use Case <nama use case 1>](#_lnxbz9) 8

[Identifikasi Kelas](#_35nkun2) 8

[Sequence Diagram](#_1ksv4uv) 8

[Diagram Kelas](#_z337ya) 8

[Perancangan Detil Kelas](#_3j2qqm3) 8

[Kelas <nama kelas>](#_1y810tw) 8

[Kelas <nama kelas>](#_4i7ojhp) 9

[Diagram Kelas Keseluruhan](#_2xcytpi) 9

[Algoritma/Query](#_1ci93xb) 9

[Diagram Statechart](#_1pxezwc) 9

[Perancangan Antarmuka](#_3whwml4) 9

[Perancangan Representasi Persistensi Kelas](#_2bn6wsx) 10

[**Matriks Kerunutan**](#_qsh70q) **10**

Setelah Daftar Isi Boleh ada Daftar Tabel dan Daftar Gambar

# 1. Pendahuluan

## Tujuan Penulisan Dokumen

Dokumen ini berisikan Deskripsi Perancangan Perangkat Lunak (DPPL) untuk Sistem pembelian tiket travel mau pun mengirim barang yang di tunjukan untuk para customer, Dengan adanya dokumen DPPL ini diharapkan pengembangan perangkat

lunak akan lebih terarah dan lebih terfokus serta tidak menimbulkan ambiguitas terutama bagi pengembang sistem serta untuk memenuhi tugas mata kuliah Analisis Perancangan Perangkat Lunak.

## Lingkup Masalah

Aplikasi tiket travel adalah sebuah perangkat lunak untuk mempermudah dalam pembelian tiket travel mobil antar Jawa Barat, fungsi lainnya juga untuk mengirim barang antar Jawa Baratk. Program ini dirancang dalam Bahasa pemrograman Java dan berbasis OOP. Dalam program ini meliputi sistem informasi terdistribusi yang berakar dan diolah.

Perangkat lunak tiket travel yang dibangun berupa sebuah aplikasi desktop yang di gunakan pada shuttle travel.

• Customer :

o Beli tiket

o Pilih jadwal

o Kirim barang

o Add barang

o Registrasi

o Pembayaran

• Admin :

o Login

o Add jadwal

o Update jadwal

o Hapus jadwal

o Cek barang

Dengan adanya aplikasi ini diharapkan dapat memudahkan semua para customer yang ingin membeli tiket untuk berpergian keluar kota agar lebih mempersingkat waktu untuk tidak memesan tiket di shuttle.

## Definisi dan Istilah

|  |  |
| --- | --- |
| ***Kata kunci*** | ***Definisi atau akronim*** |
| *DPPL* | ***Deskripsi Perancangan Perangkat Lunak,*** *merupakan deskripsi dari perancangan perangkat lunak yang akan dikembangkan. Dokumen ini merupakan lanjutan dari SKPL.* |
| *SKPL* | ***Spesifikasi Kebutuhan Perangkat Lunak****, atau dalam bahasa Inggrisnya sering juga disebut sebagai Software Requirements Spesification (SRS), dan merupakan spesifikasi dari perangkat lunak yang akan dikembangkan.* |
| *ERD* | ***Entity Relationship Diagram***, *diagram dan notasi yang digunakan untuk merepresentasikan struktur data statis pada perangkat lunak.* |
| *UML* | *Sekumpulan alat yang digunakan untuk melakukan abstraksi terhadap sebuah sistem atau perangkat lunak berbasis objek.* |
| *IEEE* | ***Institute of Electrical and Electronics Engineering Standar internasional*** *untuk pengembangan dan perancangan produk.* |
| *OOP* | ***Object Oriented Programming*** *adalah jenis pemrograman dengan metode enkapsulasi dengan membuat kelas-kelas dalam pendefinisian suatu kasus* |

## Referensi

* Jurusan Teknik Informatika – Institut Teknologi Bandung Panduan GL01, Panduan Penggunaan dan Pengisian Spesifikasi Kebutuhan Perangkat Lunak.
* Dokumen SKPL.
* Template DPPL 00 2019 Universitas Telkom Bandung.

## Sistematika Pembahasan

Dokumen DPPL ini berisikan deskripsi rancangan perangkat lunak sistem informasi perpustakaan yang akan dikembangkan berdasarkan dokumen SKPL. Pada dokumen DPPL ini akan dijelaskan rincian dari rancangan perangkat lunak sehingga dapat diimplementasikan. Dokumen ini secara garis besar teridiri dari empat bab dengan perincian sebagai berikut:

1. Pendahuluan: Pendahuluan berisi tujuan penulisan dokumen, lingkup masalah, definisi dan istilah, dan referensi yang digunakan dalam penulisan dokumen dan deskripsi umum dokumen.

2. Deskripsi Perancangan Global: Deskripsi perancangan global berisi tentang rancangan dari perangkat lunak yang akan dibangun meliputi, rancangan lingkungan implementasi, deskripsi arsitektural, dan deskripsi komponen.

3. Perancangan Rinci: Perancangan rinci berisi uraian kebutuhan perangkat lunak secara lebih rinci

4. Matriks Kerunutan: Matriks keterunutan berisi tentang hal fungsional yang terdapat pada dokumen SKPL.

# Deskripsi Perancangan Global

## Rancangan Lingkungan Implementasi

Lingkungan implementasi yang akan digunakan untuk pengembangan Aplikasi Tiket Travel ini adalah sebagai berikut:

Sistem Operasi : Windows , Linux , MacOs

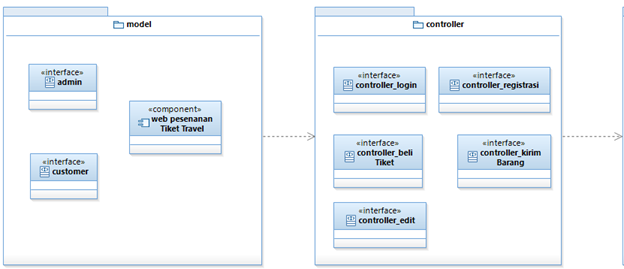
DBMS : MySQL

Development Tools : Visual Studio Code

Bahasa Pemrograman : HTML 5, CSS, JS, PHP (codeIgniter)

## Deskripsi Arsitektural

*Berikan penjelasan singkat tentang arsitektur /L yang akan dibangun. Gambarkan dalam bentuk diagram komponen.*

**

**

## Deskripsi Komponen

*Diisi dengan daftar modul. Daftar modul bisa dalam bentuk tabel berikut:*

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Komponen** | **Keterangan** |
| 1. | Admin | Admin dapat melihat data pemesanan customer dan dapat mengedit pemesanan customer |
| 2. | Customer | Customer dapat membeli tiket travel dan dapat mengirim barang |
| 3. | Web Pemesanan Tiket Travel | Menampilkan web pemesanan tiket travel |
| 4. | Controller\_login |  |
| 5. | Controller\_register |  |
| 6. | Controller\_beliTiket |  |
| 7. | Controller\_kirimBarang |  |
| 8. | Controller\_edit |  |
| 9. | login | Digunakan untuk admin untuk dapat melihat dan mengedit pemesanan customer |
| 10. | registrasi | Digunakan untuk customer jika sudah fix memesan tiket atau mengirim barang |
| 11. | beliTiket | Customer dapat membeli tiket travel antar Jawa Barat |
| 12. | kirimBarang | Customer dapat mengirim barang antar Jawa Barat |

# Perancangan Rinci

## Realisasi Use Case

|  |  |  |
| --- | --- | --- |
| **No.** | **Nama Use Case** | **Deskripsi Use Case** |
| 1. | Beli Tiket | Customer dapat melakukan proses pemesanan tiket travel sekaligus memilih jadwal yang tersedia dari system. |
| 2. | Kirim Barang | Customer dapat mengirim barang anatar jawa barat |
| 3 | Registrasi | Customer memasukan data diri ketika selesai memasukan data dari beli tiket atau kirim barang |
| 4 | tagihan | Customer mendapatkan tagihan setelah berhasil melewati tahapan registarsi |
| 5 | Pembayaran |  |
| 6 | Cek Status Bayar | Customer dapat melihat status pembayaran dari tagihan yang didapat. Admin dapat melihat status pembayaran dari customer yang di cari |
| 7 | Tambah jadwal | Admin dapat menambahkan jadwal ke daftar jadwal keberangkatan travel |
| 8 | Update Jadwal | Admin dapat mengubah jadwal yang ada pada suatu travel |
| 9 | Hapus jadwal | Admin dapat menghapus jadwal yang telah ada sebelumnya |
| 10 | Cek barang | Admin dapat melihat data suatu barang yang dicari beserta pengirimnya dan alamat yang dituju |
| 11 | Login | Admin dapat login untuk mejalankan fungsi fungs yang terdapat khusus untuk user admin |
| 12 | Tracking barang | Admin dan Customer dapat melihat status pengiriman barang yang dicari |

### Use Case

*Jika use case ini akan direalisasikan dalam bentuk aplikasi berbasis web, maka subbab yang terkait dengan perancangan elemen aplikasi berbasis web harus diisi.*

1. Use Case Skenario Beli Tiket

Use Case : Beli Tiket

Aktor : Customer

Pre Conditional : Jika ingin membeli tiket harus memilih jadwal keberangkatan

terlebih dahulu dan registrasi.

Post Conditional : Customer telah membeli tiket keberangkatan.

Description : Untuk mencatat, menambahkan, dan menyimpan data jadwal

keberangkatan customer.

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Membuka Menu dan pilih beli tiket |  |
|  | 2. Menampilkan Jadwal Keberangkatan |
| 3. Memilih jadwal keberangkatan |  |
|  | 4. Menampilkan kursi yang tersedia |
| 5. Memilih kursi yang masih tersedia |  |
|  | 6. Menampilkan tampilan form |
| 7. Masukan data actor |  |
| 8. Menekan tombol Membeli |  |
|  | 9. Memproses dan menyimpan data tiket ke dalam database |
|  | 10. System menampilkan konfirmasi yang telah di simpan |

2. Use Case Skenario Kirim Barang

Use Case : Kirim Barang

Aktor : Customer

Pre Conditional : Jika ingin mengirim barang harus memilih pilihan “Kirim

Barang”

Post Conditional : Customer telah menambahkan barang untuk di kirim

Description : Untuk mencatat, menambahkan, dan menyimpan data

Pengiriman barang customer.

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Membuka Menu dan pilih Kirim Barang |  |
| 2. Add barang yang ingin dikirim |  |
|  | 3. Menampilkan tampilan form barang yang akan di kirim |
|  | 4. Memproses dan menyimpan data barang yang akan dikirim ke dalam database |
|  | 5. System menampilkan konfirmasi yang telah di simpan |
| 6. Mengecek kembali data bahan yang telah di inputkan. |  |

3. Use Case Skenario Registrasi

Use Case : Registrasi

Aktor : Customer

Pre Conditional : Jika sudah membeli tiket atau mengirim barang customer di

haruskan untuk registrasi.

Post Conditional : Customer telah registrasi.

Description : Untuk mencatat, menambahkan, dan menyimpan data jadwal

keberangkatan customer ataupun data barang dan data

customer.

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Setelah menekan Membeli atau mengirim barang |  |
|  | 2. Sistem akan mengeluarkan form |
| 3. Masukan data diri |  |
| 4. Menekan tombol simpan |  |
|  | 5. Memproses dan menyimpan data diri customer ke dalam database data diri. |
|  | 6. Menampilkan data bahan yang telah disimpan. |
|  | 7. Kembali ke Home |

4. Use Case Skenario Pembayaran

Use Case : Pembayaran

Aktor : Customer

Pre Conditional : Jika sudah memenuhi registrasi akan diberi tagihan oleh

system lalu customer diharuskan untuk membayar.

Post Conditional : Customer telah membayar tiker.

Description : Bila customer ingin membayar tagihannya maka dapat memilih fungsionalitas ini, bila dalam waktu tertentu customer tidak membayar maka pesanannya akan tercancel atau terhapus.

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Customer bila ingin membayar dapat masuk ke halaman pembayaran |  |
| 2. Customer memasukan data tagihan yang diterima |  |
| 4. klik ok |  |
|  | 5. Sistem akan mengeluarkan data barang atau data tiket beserta tagihannya |
| 6. Customer memilih metode pembayaran |  |
| 7. klik ok |  |
|  | 8. sistem akan menyimpan data tagihan yang telah dibayar |
|  | 9. kembali ke home |

5. Use Case Skenario Tambah Jadwal

Use Case : Tambah Jadwal

Aktor : Admin

Pre Conditional : Admin sudah login dan memilih halaman tambah jadwal

Post Conditional : Admin telah menambahkan jadwal yang baru

Description : Admin dapat menambahkan jadwal yang baru ke daftar jadwal yang ada

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Masuk ke menu tambah jadwal |  |
| 2. Menambahkan jadwal baru |  |
| 3. klik ok |  |
|  | 4 Menyimpa jadwal yang telah ditambahkan |

6. Use Case Skenario Hapus Jadwal

Use Case : Hapus Jadwal

Aktor : Admin

Pre Conditional : Admin sudah login dan memilih menu hapus jadwal

Post Conditional : Jadwal yang dipilih telah terhapus

Description : Admin dapat mencari jadwal dan menghapus jadwal dari daftar jadwal yang ada

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Admin masuk ke dalam halaman hapus jadwal |  |
|  | 2. sistem menampilkan form untuk menghapus jadwal |
| 3. Admin memasukan jadwal yang ingin dihapus |  |
|  | 4. Sistem mencari jadwal yang dimasukan |
|  | 5. Sistem menampilkan detail jadwal yang dicari |
| 6. Admin memeriksa jadwal yang ditampilkan |  |
| 7. Bila sesuai dengan yang diinginkan admin menekan tombol hapus |  |
|  | 8. Sistem menampilkan popup bertulis “apakah ingin dihapus” |
| 9. klik ok |  |
|  | 10. sistem menghapus jadwal yang ada dan menampilkan popup bertulis data telah di hapus |
|  | 11. kembali ke home |

7. Use Case Skenario Update Jadwal

Use Case : Update Jadwal

Aktor : Admin

Pre Conditional : Admin sudah login dan memilih menu update jadwal

Post Conditional : Jadwal yang telah diupdate

Description : Admin dapat mencari jadwal yang diinginkan lalu mengupdatenya

|  |  |
| --- | --- |
| **Aktor** | **Sistemti** |
| 1. Admin masuk ke halaman hapus jadwal |  |
|  | 2. Sistem akan menampilan form update jadwal |
| 3. admin memasukan jadwal yang ingin dihapus |  |
|  | 4. sistem mencari jadwal yang dimaksud |
|  | 5. sistem menampilkan jadwal yang dicari |
| 6. admin mengupdate jadwal yang ada |  |
| 7. klik ok |  |
|  | 8. sistem memperbarui data yang telah dimasukan |

8. Use Case Skenario Cek Barang

Use Case : Cek barang

Aktor : Admin

Pre Conditional : Admin sudah login dan memilih menu cek barang

Post Conditional : Sistem mengeluarkan data barang yang dicari

Description : Admin dapat mencari barang yang ada lalu sistem mengeluarkan detail data barang yang dicari

|  |  |
| --- | --- |
| **Aktor** | **Sistem** |
| 1. Admin masuk ke halaman cek barang |  |
| 2. admin memasukan data barang yang ingin dicari |  |
|  | 3. sistem mencari barang yang ada |
|  | 4. sistem menampilkan data barang yang dicari |
| 5. admin memeriksa data yang ditampilkan |  |
| 6. klik home |  |
|  | 7. kembali ke home |

#### Identifikasi Kelas

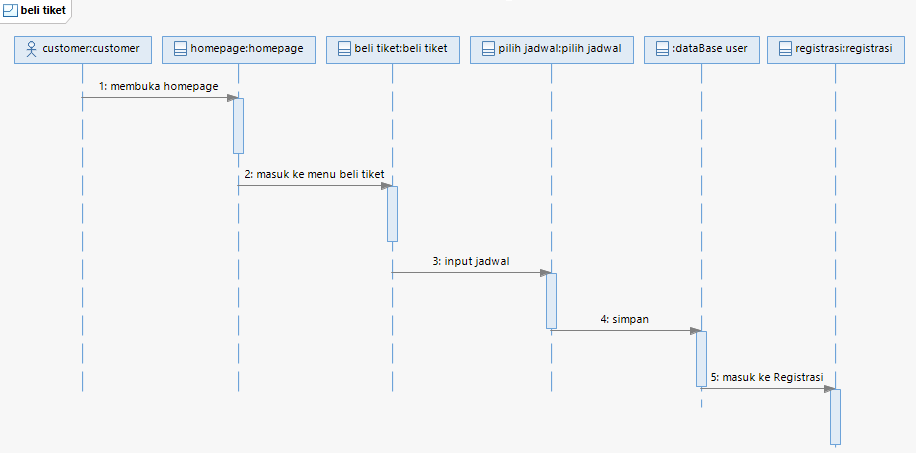
*Identifikasi kelas yang terkait dengan use case tersebut.Kelas di tahap perancangan dapat berbeda dengan dengan kelas di tahap analisis. Dapat menggunakan tabel di bawah:*

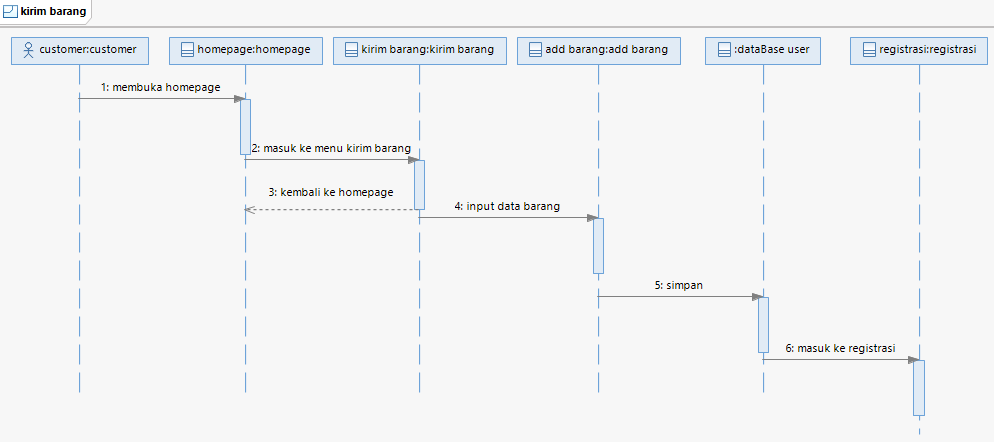
|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Tipe Kelas* |
| *1.* | *Admin* | *Model* |
| *2.* | *Customer* | *Model* |
| *3.* |  |  |

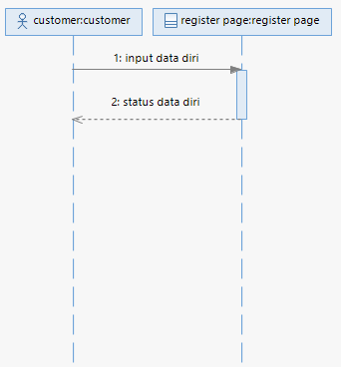
*\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller*

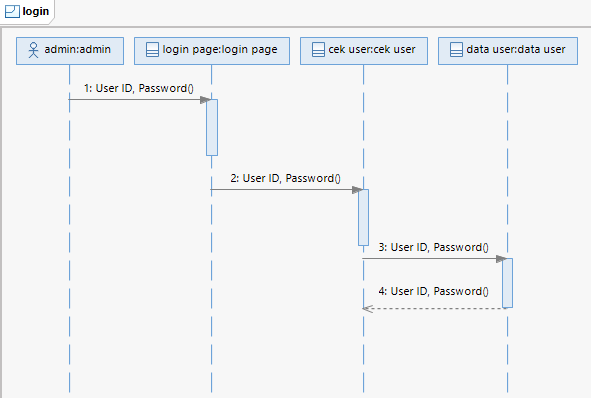
#### Sequence Diagram

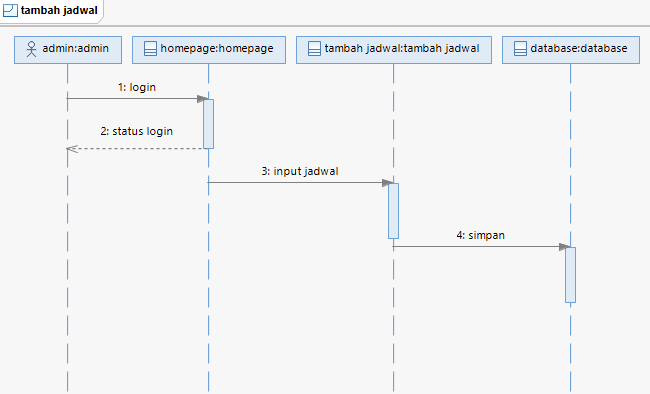
*Buatlah* **diagram sequence untuk setiap skenario use case***. Skenario melibatkan kelas-kelas yang sudah diidentifikasi.*

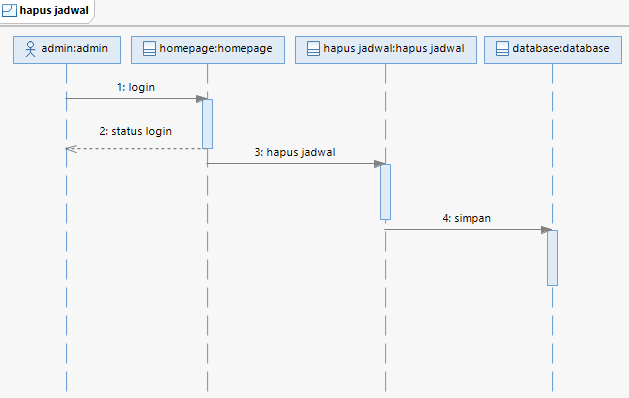


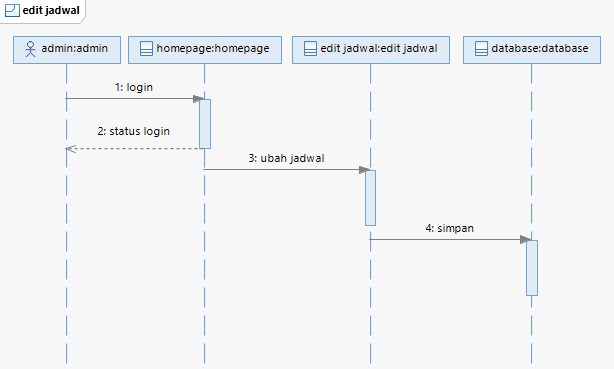


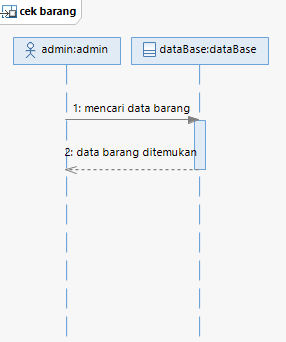












#### Diagram Kelas

*Buatlah diagram kelas untuk use case tersebut. buat class diagram* ***BUKAN KESELURUHAN, tapi PER USE CASE***

## Perancangan Detil Kelas

Tabel Kelas :

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| *KLS01* | Customer | Customer |
| *KLS02* | Admin | Admin |
| *KLS03* | Jadwal | Jadwal |
| *KLS04* | Tiket | Tiket |
| *KLS05* | Transaksi | Transaksi |
| *KLS06* | Barang | Barang |
| *KLS07* | Mobil | Mobil |

*Untuk setiap kelas:*

* *identifikasi operasi (mengacu pada tanggung-jawab kelas), termasuk visibility-nya*
* *identifikasi atribut, termasuk visibility-nya*

### Kelas Customer

Bagian ini diisi dengan daftar operasi dan atribut Buat untuk setiap kelas.

Nama Kelas : Customer

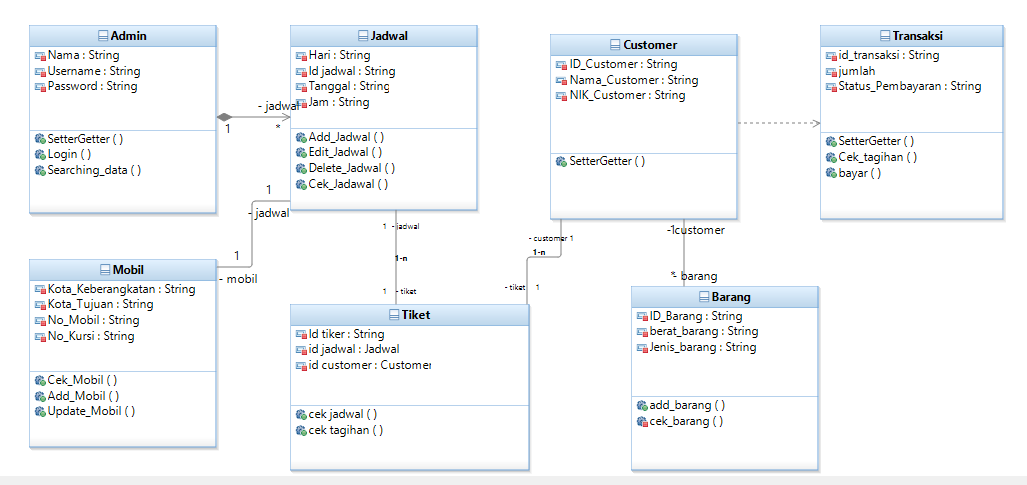
|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| IsiDataDiri | Public | Untuk memulai ingin pesan tiket atau kirim barang |
| PesanTiket | Public | Untuk memesan tiket travel antar Jawa Barat |
| KirimBarang | Public | Untuk mengirim barang antar Jawa Barat |
| Pembayaran | Private | Untuk pembayaran saat sudah fix memesan tiket travel atau mengirim barang |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| Nama | Private | String |
| Email | Private | String |
| No Telp | Private | String |

### Kelas Admin

Nama Kelas : Admin

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| Edit Pemesanan Customer | Private | Untuk mengedit pemesanan Customer jika ada permintaan edit dari customer |
| Login | Private | Masuk sebagai admin untuk dapat melihat dan mengedit data customer |
| Searching data Customer | Private | Untuk mencari data customer |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Username* | Private | String |
| *Password* | Private | String |

## Diagram Kelas Keseluruhan

*.*

## Algoritma/Query

Nama Kelas :Admin

Nama Operasi :login

Algoritma :  *(Algo-xxx)*

input(username);

input(password);

db\_username = Q1;

db\_pass = Q2;

if (username == db\_username && password==db\_password){

masuk = true

}

else{

masuk = false

}

*{Jika mengacu query tertentu, lengkapi tabel query di bawah}*

*Query :*

|  |  |  |
| --- | --- | --- |
| *No Query* | *Query* | *Keterangan* |
| *Q1* | select username from admin where username = “username”; | untuk mencari username yang berada pada tabel admin |
| Q2 | select password from admin where password = “password”; | untuk mencari password yang berada pada tabel admin |

Nama Kelas :Jadwal

Nama Operasi :add\_jadwal

Algoritma :  *(Algo-xxx)*

input(jadwal);

*{Jika mengacu query tertentu, lengkapi tabel query di bawah}*

*Query :*

|  |  |  |
| --- | --- | --- |
| *No Query* | *Query* | *Keterangan* |
| *Q1* | insert into jadwal value(jadwal) | untuk memasukan jadwal baru ke dalam table jadwal |

## Perancangan Antarmuka

**3.5.1 Menu Utama**

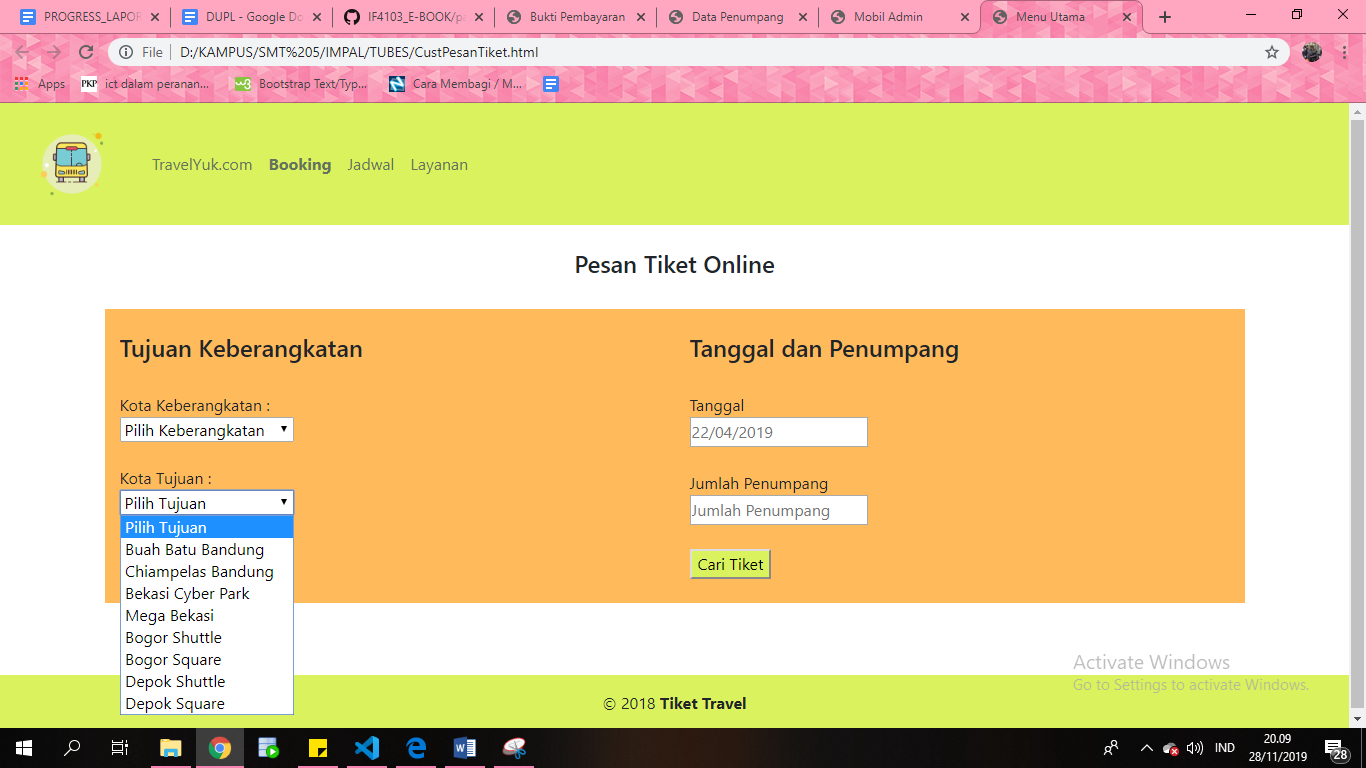
*Antarmuka : 01*



|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *Button1* | *Button* | *Cari Tiket* | *Jika diklik, akan muncul jadwal tiket keberangkatan.* |
| *Button2* | *Button* | *Icon Admin* | *Jika diklik, akan masuk ke login admin* |

**3.5.2 Ketika Memilih Tujuan**

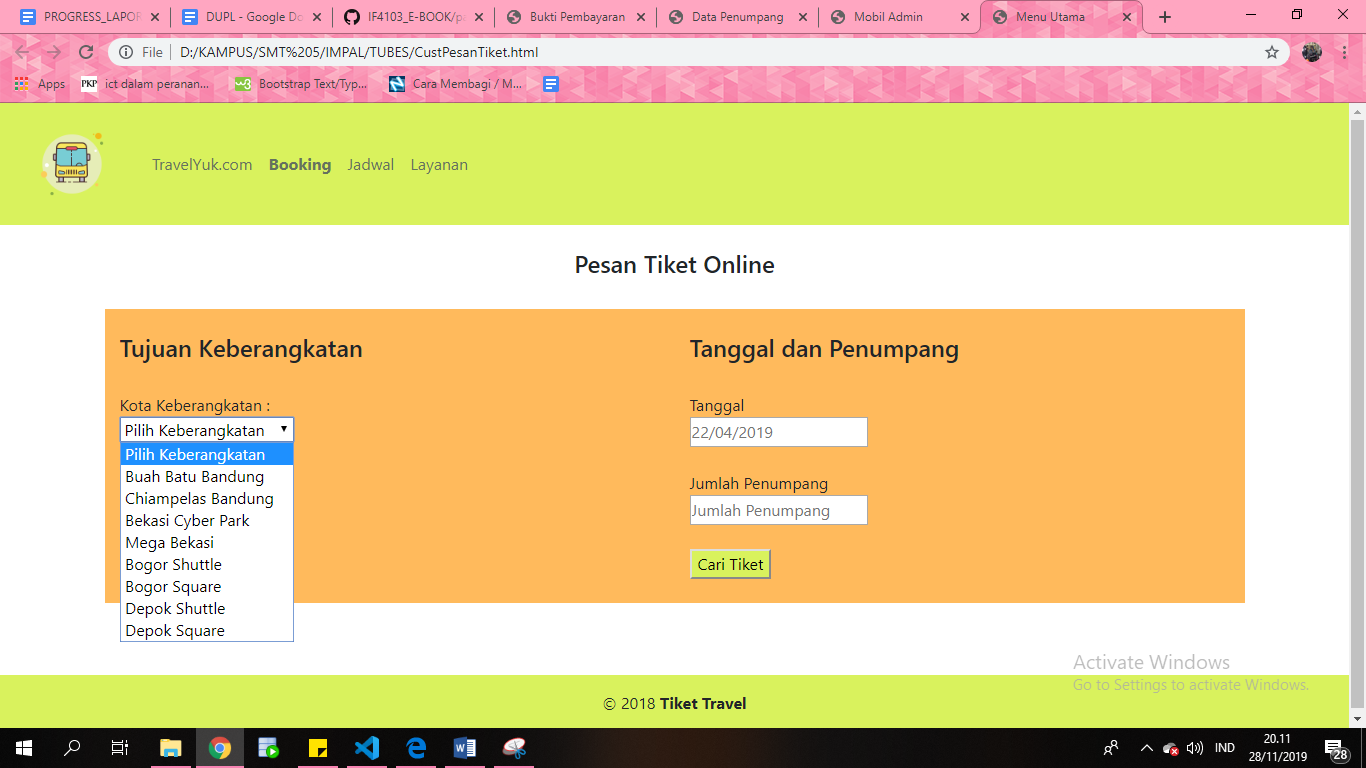
*Antarmuka : 02*



|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *Button1* | *Button* | *Back* | *Jika diklik, akan mengembalikan tampilan ke menu utama* |
| *RTF1* | *RTF Box* | *Beberapa box Kota keberangkatan* | *Isi teks akan di simpan di Pilihan Tujuan* |

**3.5.3 Keberangkatan**

*Antarmuka : 03*



|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *Button1* | *Button* | *Panah* | *Jika diklik, akan mengembalikan pada menu utama* |
| *RTF1* | *RTF Box* | *Beberapa box Kota keberangkatan* | *Isi teks akan di simpan di Pilihan Keberangkatan* |

**3.5.4 Atur Tanggal**

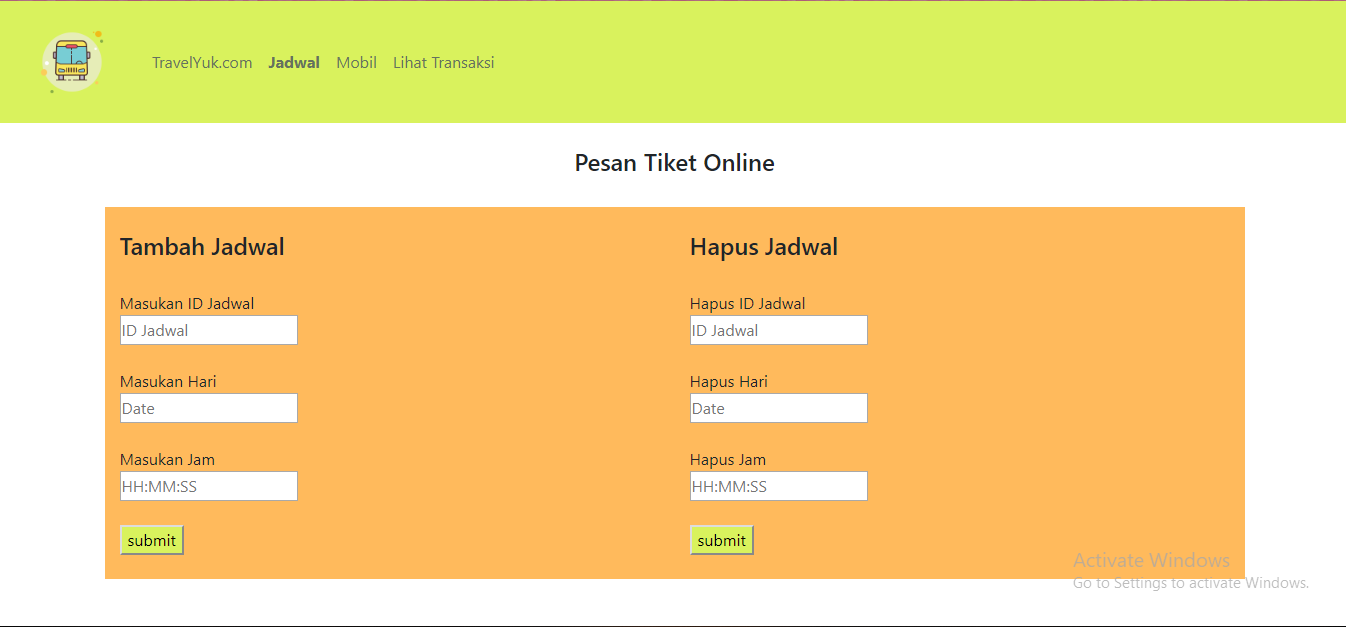
*Antarmuka : 04*



|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *Button1* | *Button* | *Back* | *Jika diklik, akan mengembalikan pada menu utama* |
| *RTF1* | *RTF Box* | *Kalendar tanggal keberangkatan* | *Isi Teks yang disimpan pada pilihan jadwal keberangkatan* |

**3.5.5 Edit Jadwal**

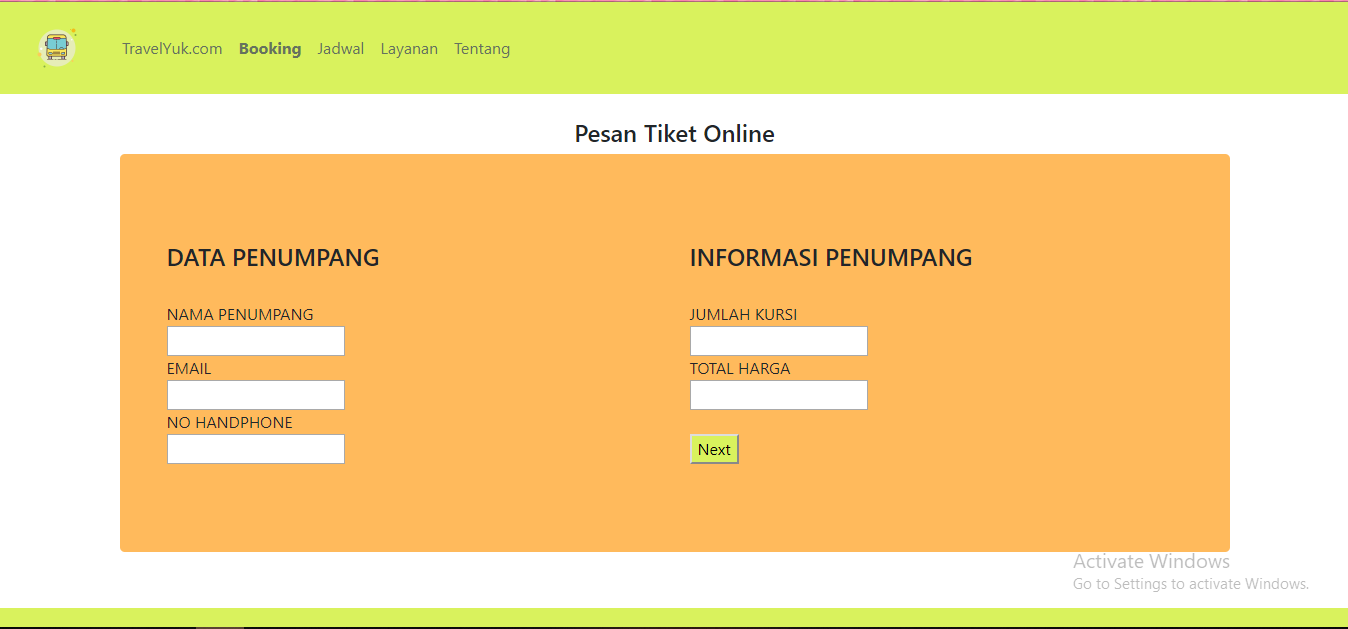
*Antarmuka : 05*

******

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *Button1* | *Button* | *Cari Tiket* | *Jika diklik, akan beralih ke tampilan registrasi* |
| *RTF1* | *RTF Box* | *Jam keberangkatan* | *Isi Teks yang disimpan pada pilihan waktu keberangkatan* |

**3.5.6 registrasi**

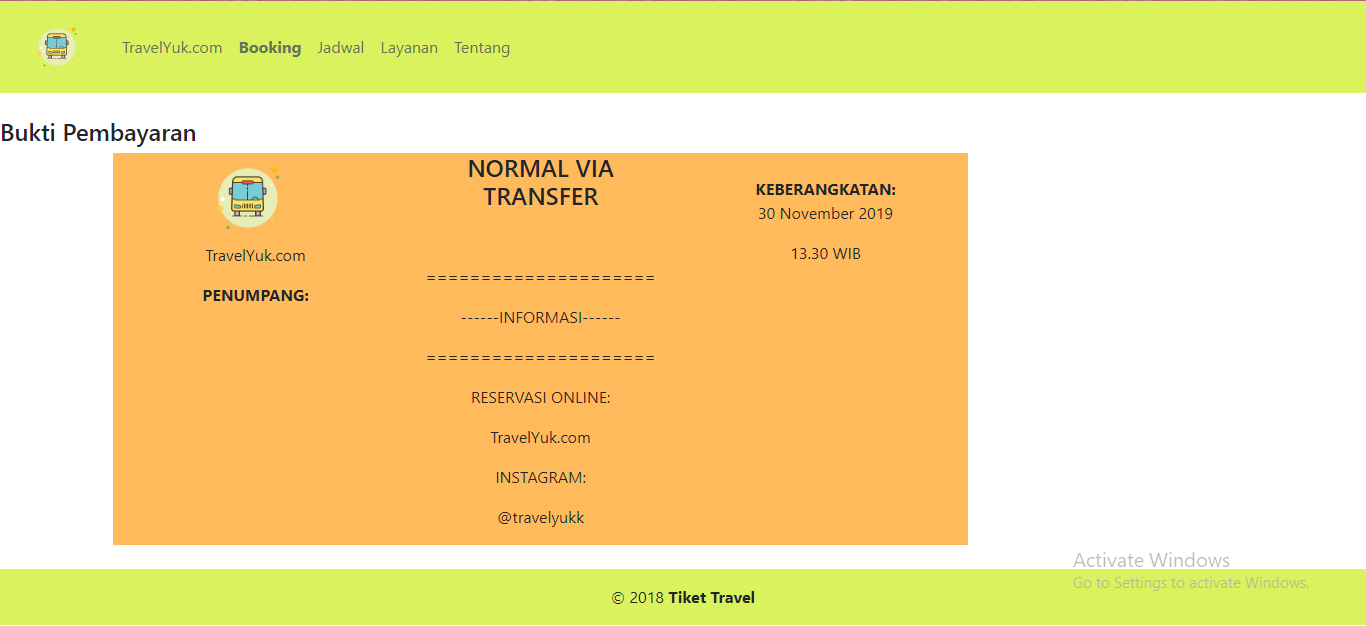
*Antarmuka : 06*



|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *Button1* | *Button* | *Next* | *Jika diklik, akan melanjutkan tampilan ke tampilan tiket* |
| *RTF1* | *RTF Box* | *Data Penumpang* | *Isi Teks yang disimpan pada database penumpang yang membeli tiket travel* |
| *RTF2* | *RTF Box* | *Informasi Penumpang* | *Isi Teks yang disimpan pada database penumpang yang membeli tiket travel* |

**3.5.7 Tampilan Tiket**

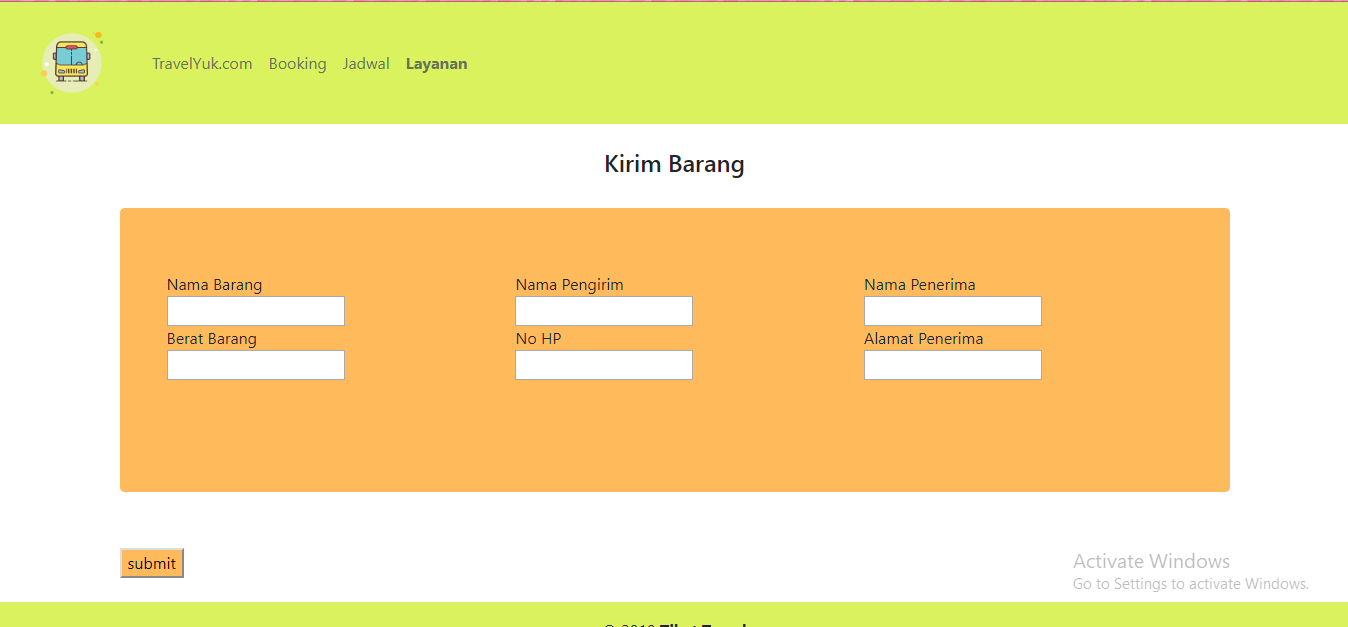
*Antarmuka : 07*



|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *RTF1* | *RTF Box* | *Bukti Pembayaran* | *Isi Teks tersebut bukti kita sudah membayar tiket tersebut.* |

**3.5.8 Kirim Barang**

*Antarmuka : 08*



|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *Button1* | *Button* | *Submit* | *Jika diklik, akan menyimpan data yang telah kita isi dan akan di proses* |

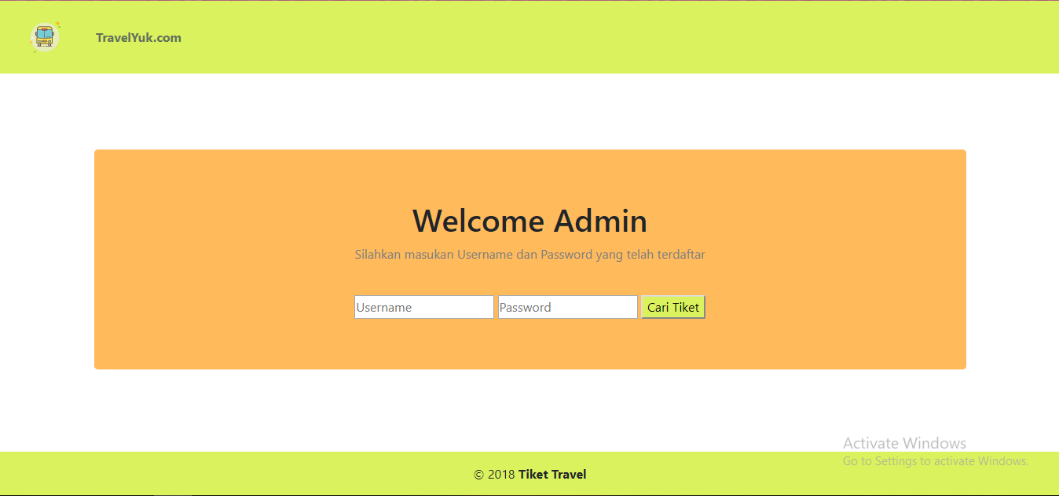
**3.5.9 Login Admin**

*Antarmuka : 09*

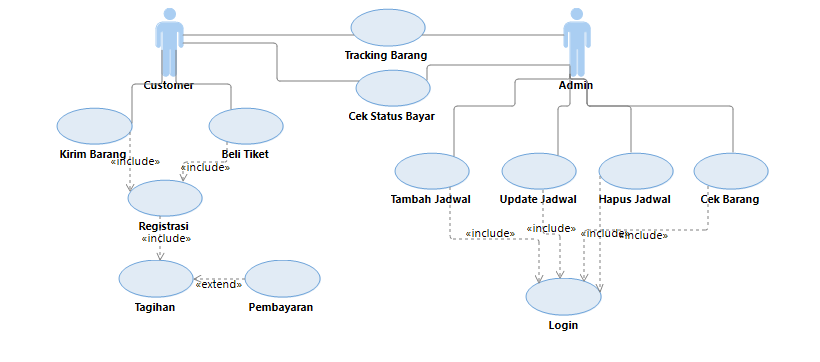
|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| *Button1* | *Button* | *Login* | *Jika di klick maka melakukan proses login.v* |

## Perancangan Representasi Persistensi Kelas

*Bagian ini diisi dengan rancangan skema basisdata dan traceability-nya terhadap kelas entity. (PEMBUATAN SKEMA RELASI)*



# Matriks Kerunutan

**

### 

|  |  |  |  |
| --- | --- | --- | --- |
| No. | Kode Kebutuhan | Fungsi | Kelas |
| 1. | FR-01 | Input Data Tiket | Customer |
| 2. | FR-02 | Input Data Barang | Customer |
| 3. | FR-03 | Input Data Mobil dan Supir | Admin |
| 4. | FR-04 | Registrasi | Customer |
| 5. | FR-05 | Login | Admin |
| 6. | FR-06 | Pesan Tiket | Customer |
| 7. | FR-07 | Kirim Barang | Customer |

# 