**DPPL-01**

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

**Aplikasi Layanan Photographer *‘Jones’***

untuk:

Dipersiapkan oleh:

Aditya Mahendra Zakaria

Adiyasa Niastya Imam Fadhilah

Ahmad Afiq Fitrah

Rafi Anandita Wicaksono

Program Studi Informatika

Fakultas Informatika

Jl. Telekomunikasi 1, Dayeuhkolot Bandung

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Prodi S1- Informatika**  **Universitas Telkom** | Nomor Dokumen | | Halaman |
| *DPPL-01 <xx:no grp>* | | *<#>/<jml #* |
| Revisi | *-* | *Tgl: 19 April 2020* |

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

[Daftar Tabel 5](#_gjdgxs)

[Daftar Gambar 6](#_30j0zll)

[1. Pendahuluan 7](#_1fob9te)

[1.1](#_3znysh7) Tujuan Penulisan Dokumen 7

[1.2](#_2et92p0) Lingkup Masalah 7

[1.3](#_tyjcwt) Definisi dan Istilah 7

[1.4](#_3dy6vkm) Referensi 7

[1.5](#_1t3h5sf) Sistematika Pembahasan 7

[2](#_4d34og8) Deskripsi Perancangan Global 8

[2.1](#_2s8eyo1) Deskripsi Arsitektural 8

[2.2](#_17dp8vu) Deskripsi Komponen 8

[3](#_26in1rg) Perancangan Rinci 9

[3.1](#_lnxbz9) Realisasi Use Case 9

[3.1.1](#_35nkun2) Use Case #1 <nama use case 1> 9

[3.1.1.1](#_1ksv4uv) Perancangan Antarmuka Usecase #1 9

[3.1.1.1.1](#_44sinio) Tabel Identifikasi Antarmuka / Layar / Page 9

[3.1.1.2](#_2jxsxqh) Identifikasi Object Baru 9

[3.1.1.3](#_z337ya) Robustness Diagram 10

[3.1.1.4](#_3j2qqm3) Diagram Kelas 10

[3.1.1.5](#_2xcytpi) Sequence Diagram 10

[3.1.2](#_1ci93xb) Use Case #1 <nama use case 1> 10

[3.1.2.1](#_3whwml4) Perancangan Antarmuka Usecase #2 10

[3.1.2.1.1](#_2bn6wsx) Tabel Identifikasi Antarmuka / Layar / Page 10

[3.1.2.2](#_qsh70q) Identifikasi Object Baru 11

[3.1.2.3](#_3as4poj) Robustness Diagram 11

[3.1.2.4](#_1pxezwc) Diagram Kelas 11

[3.1.2.5](#_49x2ik5) Sequence Diagram 11

[4](#_2p2csry) Perancangan Detil 11

[4.1](#_147n2zr) Perancangan Detil Kelas 11

[4.2](#_3o7alnk) Perancangan Kelas Persistensi (\*\*\*Basis Data Skema Tabel) 11

[4.3](#_ihv636) Perancangan Algoritma 11

[4.3.1](#_32hioqz) Algoritma #1 11

[4.4](#_1hmsyys) Perancangan Query 12

[5](#_41mghml) Matriks Kerunutan (Requirement Traceability Matrix) 12

Setelah Daftar Isi Boleh ada Daftar Tabel dan Daftar Gambar

# Daftar Tabel

# Daftar Gambar

# 1. Pendahuluan

## Tujuan Penulisan Dokumen

Penulisan dokumen ini bertujuan agar user dapat memahami cara kerja perangkat lunak ini, juga memenuhi indikator penilaian mata kuliah Analisis Perancangan Perangkat Lunak (APPL). Dengan adanya dokumen ini, kami berharap bahwa user dapat menggunakan fitur-fitur yang ada pada perangkat lunak ini dengan memanfaatkan secara optimal. Aplikasi Registrasi Photo Studio *‘Jones’* mempunyai tujuan untuk mempermudah pelanggan yang hendak foto studio di foto studio *‘Jones’.*

## Lingkup Masalah

Aplikasi Registrasi Photo Studio *‘Jones’* mempunyai ruang lingkup tertentu yaitu pelanggan yang ingin foto studio. Aplikasi ini cara mengaksesnya dengan internet di web. Tidak perlu mengunduhnya terlebih dahulu.

## Definisi dan Istilah

* SKPL = Spesifikasi Kebutuhan Perangkat Lunak
* DPPL = Deskripsi Perancangan Perangkat Lunak
* DFD = Data Flow Diagram
* ERD = Entity Relationship Diagram
* Main Menu = daftar perintah-perintah suatu perangkat lunak (program) yang apabila dieksekusi akan menjalankan suatu perintah tertentu dari aplikasi
* Signup = tanda seseorang untuk membuat sebuah account atau ID di sebuah email atau yang lainnya
* Login = suatu Proses untuk masuk ke dalam sebuah layanan online yang berisi nama dan password
* Database = kumpulan data yang disimpan secara sistematis di dalam komputer yang dapat diolah atau dimanipulasi menggunakan perangkat lunak (software) program atau aplikasi untuk menghasilkan informasi
* User = Pemakai atau pengguna dalam komputer
* Use case = Sebuah kegiatan atau juga interaksi yang saling berkesinambungan antara aktor dan juga sistem. Atau dengan kata lain teknik secara umum digunakan, guna mengembangkan software / sistem informasi, guna memperoleh kebutuhan fungsional dari sistem yang ada

## Referensi

Dokumen SKPL ini mempunyai acuan cara kerja sistem di dunia nyata yaitu “Foto Studio Jonas’ dimana secara sistematis kami mengambil acuan dari mereka kemudian dimodifikasi sedikit. Selain dari itu, kami tidak ada mengambil referensi lain.

## Sistematika Pembahasan

Dokumen DPPL ini berisi deskripsi rancangan perangkat lunak yang akan dikembangkan berdasarkan dokumen SKPL

1. Bab 1, Membahas tentang overview dari rencana pengembangan perangkat lunak secara umum
2. Bab 2, Merupakan deskripsi perancangan global yang berisi peracangan perangkat lunak secara umum
3. Bab 3, Merupakan deskripsi perancangan rinci yang berisi tentang rincian keseluruhan dari perangkat lunak yang akan dikembangkan, seperti pendeskripsian secara rinci aplikasi tersebut
4. Bab 4, Merupakan deskripsi perancangan detail menyeluruh dari perangkat lunak yang akan digunakan

# Deskripsi Perancangan Global

## Deskripsi Arsitektural

## Deskripsi Komponen

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

***Isinya komponen diagram***

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Komponen** | **Keterangan** |
| 1 | mainmenu.php | Tampilan interface menu utama |
| 2 | welcome.php | Tampilan interface awal |
| 3 | signupmenu.php | Tampilan untuk membuat akun baru |
| 4 | loginmenu.php | Tampilan untuk masuk jika sudah memiliki akun |
| 5 | pendaftaranmenu.php | Tampilan untuk memesan jasa foto studio |
| 6 | pembayaranmenu.php | Tampilan untuk membayar jasa foto studio |
| 7 | pilihfoto.php | Tampilan untuk memilih foto yang akan di cetak |
| 8 | availablestudio.php | Tampilan untuk menampilkan studio yang kosong |
| 9 | availableschedule.php | Tampilan untuk menampilkan jadwal yang kosong |
|  |  |  |

# Perancangan Rinci

## Realisasi Use Case

Berisi TABEL USE CASE sebagai berikut :

|  |  |  |
| --- | --- | --- |
| No | Nama Use Case | Deskripsi Use Case |
| #1 |  |  |
| #2 |  |  |

### Use Case #1 <nama use case 1>

*Skenario Use Case #1 :*

*Primary Flow :*

*Alternate Flow :*

#### Perancangan Antarmuka Usecase #1

*Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case*

*Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detilnya.*

#### Tabel Identifikasi Antarmuka / Layar / Page

|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
|  | *Home Page* |  |
|  | *Page Registrasi* |  |
|  | *Page Transaksi* |  |

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detil

*Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}*

*Page HOMEPAGE*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *Button1* | *Button* | *OK* | *Jika diklik, akan mengaktifkan Function…..* |
| *Buuton 2* | *Button* | *SAVE* | *Jika diklik akan mengaktifkan ……* |
| *RTF1* | *RTF Box* |  | *Isi Teks yang disimpan pada File xxx* |
|  |  |  |  |

*Page REGISTRASI*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *TextBox1* | *Textbox* | *Username* | *Jika diklik, akan mengaktifkan Function…..* |
| *Buuton 2* | *Button* | *SAVE* | *Jika diklik akan mengaktifkan ……* |

#### Identifikasi Object Baru

*Identifikasi object yang terkait dengan use case tersebut.*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.*

*Gunakan tabel di bawah:*

*TABEL OBJECT PERANCANGAN*

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
|  |  |  |
|  |  |  |
|  |  |  |

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

#### Robustness Diagram

*Buatlah diagram robustness untuk masing – masing use case*

#### Diagram Kelas

*Buatlah diagram kelas untuk masing – masing use case*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis*

#### Sequence Diagram

*Buatlah diagram sequence untuk masing - masing use case sesuai skenario.*

*Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.*

### Use Case #2 <nama use case 2>

*Skenario Use Case #1*

*Primary Flow*

*Alternate Flow*

#### Perancangan Antarmuka Usecase #2

*Bagian ini diisi dengan versi awal prototipe antarmuka untuk per Use Case*

*Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detilnya.*

#### Tabel Identifikasi Antarmuka / Layar / Page

|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
|  | *Home Page* |  |
|  | *Page Registrasi* |  |
|  | *Page Transaksi* |  |

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detil

*Antarmuka XXX: {diisi dengan no. layar atau no gambar rancangan antarmuka}*

*Page HOMEPAGE*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *Button1* | *Button* | *OK* | *Jika diklik, akan mengaktifkan Function…..* |
| *Buuton 2* | *Button* | *SAVE* | *Jika diklik akan mengaktifkan ……* |
| *RTF1* | *RTF Box* |  | *Isi Teks yang disimpan pada File xxx* |
|  |  |  |  |

*Page REGISTRASI*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *TextBox1* | *Textbox* | *Username* | *Jika diklik, akan mengaktifkan Function…..* |
| *Buuton 2* | *Button* | *SAVE* | *Jika diklik akan mengaktifkan ……* |

#### Identifikasi Object Baru

*Identifikasi object yang terkait dengan use case tersebut.*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis.*

*Gunakan tabel di bawah:*

TABEL OBJECT PERANCANGAN

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
|  |  |  |
|  |  |  |
|  |  |  |

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

#### Robustness Diagram

*Buatlah diagram robustness untuk masing – masing use case*

#### Diagram Kelas

*Buatlah diagram kelas untuk masing – masing use case*

*Kelas pada tahap perancangan berbeda dengan kelas pada tahap analisis*

#### Sequence Diagram

*Buatlah diagram sequence untuk masing - masing use case sesuai skenario.*

*Skenario harus melibatkan kelas-kelas perancangan yang baru diidentifikasi.*

# Perancangan Detil

#### Perancangan Detil Kelas

*Bagian ini diisi dengan daftar keseluruhan kelas yang akan digunakan dalam PL. menggunakan model MVC*

**TABEL KELAS :**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID Kelas** | **Nama Kelas Perancangan** | **Atribute (visibility)** | **Method / Operation** |
| *1* | *Pelanggan* | *Nama,NomorHP,Alamat (Public)* | *setNomorHP(),setAlamat(),getNama(),getNomorHP(), getAlamat(), report(),pilihFoto(), editFoto()* |
| *2* | *PhotoGrapher* | *idPegawai, Nama, dataRegis(Public)* | *ShowFoto()* |
| *3* | *Registrasi* | *idRegistrasi,dataPembayaran, Jadwal, Studio, pemesan, dataFoto (Public)* | *setJadwal(), setStudio(), getJadwal(), getStudio(), getidRegis(), setDataFoto(), getDataFoto(), setidRegis()* |
| *4* | *Kasir* | *idPegawai, Nama (Public)* | *verifikasiPembayaran(), verifikasiPemesanan()* |
| *5* | *Pembayaran* | *idPembayaran, Jumlah (Public)* | *setidPembayaran(), setJumlah(), getJumlah(), getidPembayaran()* |

#### Perancangan Kelas Persistensi (\*\*\*Basis Data Skema Tabel)

Skema Relasi database yang digunakan

#### Perancangan Algoritma

#### Algoritma #1

*Nama Kelas : Pelanggan*

*Nama Operasi : setNomorHP()*

*Algoritma :*

if (getNomorHP() != NULL) then

Add nomorHP into DB

else

Output (‘Nomor Hp tidak valid’)

#### Algoritma #2

*Nama Kelas : Pelanggan*

*Nama Operasi : setAlamat()*

*Algoritma :*

*If (getAlamat() != NULL) then*

*Add alamat into DB*

*else*

*Output(‘Alamat Tidak Valid’)*

#### Algoritma #3

*Nama Kelas : Pelanggan*

*Nama Operasi : getAlamat()*

*Algoritma :*

*Input (Alamat)*

#### Algoritma #4

*Nama Kelas : Pelanggan*

*Nama Operasi : getNama()*

*Algoritma :*

*Input (Nama)*

#### Algoritma #5

*Nama Kelas : Pelanggan*

*Nama Operasi : getNomorHP()*

*Algoritma :*

*Input(NomorHP)*

#### Algoritma #6

*Nama Kelas : Pelanggan*

*Nama Operasi : report()*

*Algoritma :*

*Input( data pelanggan)*

*Input (Tanggal)*

*Input( Keluhan) // berupa text box*

*Add keluhan into DB*

*Output(‘Keluhan anda akan segera kami proses’)*

#### Algoritma #7

*Nama Kelas : Pelanggan*

*Nama Operasi : editFoto()*

*Algoritma :*

*status=false*

*while (!status) do*

*edit.foto*

*if (edit.Foto done) then*

*status=true*

*//endif*

*//endwhile*

#### Algoritma #8

*Nama Kelas : Pelanggan*

*Nama Operasi : pilihFoto()*

*Algoritma :*

*i=1*

*while (i<= maxFoto) do*

*selectedFoto[i]<- select.Foto*

*if (select.Foto != NULL) then*

*i++*

*Add select.Foto into DB*

*//endif*

*//endwhile*

#### Algoritma #9

*Nama Kelas : PhotoGrapher*

*Nama Operasi : ShowFoto()*

*Algoritma :*

*i=1*

*while (selectedFoto[i]!= NULL) do*

*show.selectedFoto[i]*

*//endWhile*

#### Algoritma #10

*Nama Kelas : Registrasi*

*Nama Operasi : setJadwal()*

*Algoritma :*

if (setJadwal() != NULL) then

Add jadwal into DB

else

Output (‘Nomor Hp tidak valid’)

#### Algoritma #11

*Nama Kelas : Registrasi*

*Nama Operasi : setStudio()*

*Algoritma :*

if (getStudio() != NULL) then

Add studio into DB

else

Output (‘Nomor Hp tidak valid’)

#### Algoritma #12

*Nama Kelas : Registrasi*

*Nama Operasi : getJadwal()*

*Algoritma :*

*while (selectedJadwal=NULL) do*

*select.Jadwal*

*If (select.Jadwal=0) then*

*selectedJadwal=’PENUH’*

*//endif*

*//endwhile*

#### Algoritma #13

*Nama Kelas : Registrasi*

*Nama Operasi : getStudio()*

*Algoritma :*

*Input(Studio)*

#### Algoritma #14

*Nama Kelas : Registrasi*

*Nama Operasi : getidRegis()*

*Algoritma :*

*input(idRegis)*

#### Algoritma #15

*Nama Kelas : Registrasi*

*Nama Operasi : setDataFoto()*

*Algoritma :*

if (getDataFoto() != NULL) then

Add DataFoto into DB

else

Output (‘Data Foto kosong’)

#### Algoritma #16

*Nama Kelas : Registrasi*

*Nama Operasi : getDataFoto()*

*Algoritma :*

*if (sudah foto) then*

*input(dataFoto)*

*else*

*lakukan foto dulu*

#### Algoritma #17

*Nama Kelas : Registrasi*

*Nama Operasi : setidRegis()*

*Algoritma :*

if (getidRegis() != NULL) then

Add id regis into DB

else

Output (‘ID Regis tidak valid’)

#### Algoritma #18

*Nama Kelas : Kasir*

*Nama Operasi : verifikasiPembayaran()*

*Algoritma :*

*if (verifikasiPemesanan=True) and (sudah foto) then*

*verifikasiPembayaran<- TRUE*

*else*

#### Algoritma #19

*Nama Kelas : Kasir*

*Nama Operasi : verifikasiPemesanan()*

*Algoritma :*

*if ( data pemesanan true) then*

*return true*

*else*

*return false*

#### Algoritma #20

*Nama Kelas : Pembayaran*

*Nama Operasi : setidPembayaran()*

*Algoritma :*

if (getNomorHP() != NULL) then

Add nomorHP into DB

else

Output (‘Nomor Hp tidak valid’)

#### Algoritma #21

*Nama Kelas : Pembayaran*

*Nama Operasi : setJumlah()*

*Algoritma :*

if (getJumlah() != NULL) then

Add jumlah into DB

else

Output (‘Jumlah tidak valid’)

#### Algoritma #22

*Nama Kelas : Pembayaran*

*Nama Operasi : getidPembayaran()*

*Algoritma :*

*if (verifikasiPemesanan() = TRUE) then*

*Output (idPembayaran)*

*else*

*Output(‘Silahkan Pesan terlebih dahulu’)*

#### Algoritma #23

*Nama Kelas : Pembayaran*

*Nama Operasi : getJumlah()*

*Algoritma :*

*Jumlah<-paketFoto.tarif*

#### Perancangan Query

*Query :*

|  |  |  |
| --- | --- | --- |
| *No Query* | *Query* | *Keterangan* |
| *Q-001* | *INSERT INTO ‘PELANGGAN’ (NomorHP) values (nomorHP)* | *Memasukkan Nomor HP ke tabel pelanggan* |
| *Q-002* | *INSERT INTO ‘FOTO’ (DATAFOTO) VALUES (‘DATAFOTO’)* | *Memasukkan data foto ke dalam tabel Foto* |
| *Q-003* | *INSERT INTO ‘TRANSAKSI’ (TARIF) VALUES (‘JUMLAH’)* | *Memasukkan nominal yang harus dibayar kedalam tabel Transaksi* |
| *Q-004* | *INSERT INTO ‘PEMOTRETAN’ (FOTO) VALUES (‘ ’)* | *Memasukkan foto kedalam tabel pemotretan* |

# Matriks Kerunutan (Requirement Traceability Matrix)

*Mapping requirement dengan Use Case yang direalisasikan*

|  |  |  |
| --- | --- | --- |
| **Kode FR** | **Nama Functional Requirement** | **Nama Use Case** |
| FR-06 | Pembayaran | Pembayaran |
| FR-07 | Pemilihan Foto | Pilih Foto |
| FR-03 | Registrasi Pemotretan | Registrasi Pemesanan |
| FR-10 | Verifikasi Pendaftaran | Verifikasi Pemesanan |
| FR-12 | Verifikasi Pembayaran | Verifikasi Pembayaran |

# 