**DPPL-W-08**

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

Citizen Report Project

untuk:

Pemerintah Kota Bandung

Dipersiapkan oleh:

Kelompok 8 :

**Steve Immanuel Harnadi 13512035**

**Fahmi Dumadi 13512047**

**Mario Filino 13512055**

**Lutfi Fadlan 13512087**

**Edmund Ophie 13512095**

Program Studi Teknik Informatika/Sistem dan Teknologi Informasi

STEI - ITB

Jl. Ganesha 10, Bandung 40132

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Program Studi Teknik Informatika / Sistem dan Teknologi Informasi**  **STEI – ITB** | Nomor Dokumen | | Halaman |
| *DPPL-W-08* | |  |
| 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 5

1.1 Tujuan Penulisan Dokumen 5

1.2 Lingkup Masalah 5

1.3 Definisi dan Istilah 5

1.4 Aturan Penamaan dan Penomoran 5

1.5 Referensi 5

1.6 Ikhtisar Dokumen 5

2 Deskripsi Perancangan Global 6

2.1 Rancangan Lingkungan Implementasi 6

2.2 Deskripsi Arsitektural 6

2.3 Deskripsi Komponen 6

2.4 Pedoman Perancangan 6

3 Perancangan Rinci 7

3.1 Realisasi Use Case 7

3.1.1 Use Case <nama use case 1> 7

3.1.1.1 Identifikasi Elemen WAE - Logical View 7

3.1.1.2 Identifikasi Kelas 7

3.1.1.3 Sequence Diagram 7

3.1.1.4 Diagram Kelas 7

3.1.2 Use Case <nama use case 2> 7

3.1.2.1 Identifikasi Elemen WAE - Logical View 8

3.1.2.2 Identifikasi Kelas 8

3.1.2.3 Sequence Diagram 8

3.1.2.4 Diagram Kelas 8

3.2 Diagram Kelas Keseluruhan 8

3.3 Perancangan Detil Elemen Logical View 8

3.3.1 Stereotyped Class <nama elemen logical view> 9

3.3.2 Stereotyped Class <nama elemen logical view> 9

3.4 Perancangan Detil Kelas 9

3.4.1 Kelas <nama kelas> 9

3.4.2 Kelas <nama kelas> 10

3.5 Algoritma/Query 10

3.6 Diagram Statechart 11

3.7 Perancangan Antarmuka 11

3.8 Perancangan Representasi Persistensi Kelas 11

3.9 Deployment Diagram 11

4 Matriks Kerunutan 11

Setelah Daftar Isi Boleh ada Daftar Tabel dan Daftar Gambar

# 1. Pendahuluan (Steve Immanuel H.)

## Tujuan Penulisan Dokumen

Dokumen DPPL ini ditulis sebagai rancangan mendetail aplikasi yang kami kembangkan. Dokumen ini sekaligus juga memberikan gambaran bagi pengembang aplikasi mengenai perangkat lunak pengaduan masyarakat Kota Bandung, sehingga mempermudah pengembang dalam mengimplementasikan setiap modul/kelas terkait. Di samping itu, dokumen ini dapat juga menjadi jembatan antara kebutuhan user sampai dengan tahap implementasi perangkat lunak setelahnya.

## Lingkup Masalah

Citizen Report merupakan sebuah perangkat lunak berbasis web yang digunakan untuk mengakomodasi pengaduan dari masyarakat Bandung. Perangkat lunak ini berfungsi sebagai saluran komunikasi yang efektif antara masyarakat dan SKPD yang ada di Bandung. Pengaduan dari masyarakat akan ditampung dan diteruskan ke SKPD yang terkait pengaduan tersebut. Dengan adanya perangkat lunak ini diharapkan pengaduan masyarakat dapat ditangani dengan cepat.

## Definisi dan Istilah

|  |  |
| --- | --- |
| **Istilah** | **Arti/Singkatan** |
| Dukcapil | Kependudukan dan catatan sipil |
| SKPD | Satuan Kerja Perangkat Daerah |
| DISKOMINFO | Dinas Komunikasi dan Informatika |
| Use Case | Diagram yang menggambarkan aksi apa saja yang dilakukan sistem terhadap aktor luar. |

## Aturan Penamaan dan Penomoran

|  |  |
| --- | --- |
| **Istilah** | **Arti** |
| FR-xx | Penomoran kebutuhan fungsional |
| NFR-xx | Penomoran kebutuhan nonfungsional |
| T-xx | Penomoran tabel |
| G-xx | Penomoran gambar |
| UC-xx | Penomoran use case |

## Referensi

Dokumen SKPL-K1-G08

## Ikhtisar Dokumen

Dokumen ini secara umum terdiri dari tiga bagian besar : pendahuluan, deskripsi perancangan global, perancangan rinci. Bagian pendahuluan berisi gambaran dokumen secara umum seperti tujuan penulisan, lingkup masalah, aturan penomoran, dan lain-lain. Bagian perancangan global berisi rancangan garis besar modul/kelas yang membentuk perangkat lunak. Bagian perancangan rinci adalah uraian mendetail dari bagian perancangan global. Sedangkan bagian akhir dari dokumen ini adalah bab matriks keterunutan, yang berisi keterkaitan antara modul/kelas dengan kebutuhan fungsional/nonfungsional perangkat lunak.

# Deskripsi Perancangan Global (Mario Filino)

## Rancangan Lingkungan Implementasi

Sebutkan Operating system, DBMS, development tools, filing system, bahasa pemrograman yang dipakai

## Deskripsi Arsitektural

*Berikan penjelasan singkat tentang arsitekturP /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** |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

## Pedoman Perancangan

Bagian ini diisi dengan uraian mengenai framework atau arsitektur khusus yang digunakan pada saat perancangan dan implementasi. Jika tidak ada, hilangkan subbab ini

# Perancangan Rinci

## Realisasi Use Case (Sesuai pembagian implementasi use case)

### Use Case Membuat Pengaduan

#### Identifikasi Elemen WAE - Logical View

Bagian ini diisi dengan hasil identifikasi elemen WAE-Logical View yang merealisasikan use case (antara lain berasal dari kelas analisis yang TIDAK AKAN direalisasikan menjadi kelas biasa, tetapi akan menjadi PAGE atau FORM):

* *server page*
* *client page*
* *html form*

|  |  |  |
| --- | --- | --- |
| No | Nama Elemen Lojik WAE | Stereotype |
| 1 | Buat Pengaduan | Client Page |
| 2 | BuatPengaduanForm | HTML Form |
| 3 | BuatPengaduan | ServerPage |
| 4 | BuatPengaduanStatus | Client Page |

#### Identifikasi Kelas

Buat daftar kelas perancangan (berasal dari kelas analisis yang TIDAK direalisasikan menjadi elemen WAE ogical View) yang merealisasikan use case dalam tabel berikut:

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Kelas* | *Library yang digunakan* |
| *1* | *PagesController* | *-* |
| *2* | *Pengaduan* | *-* |

#### Sequence Diagram

Buatlah diagram sequence untuk setiap skenario use case. Skenario MELIBATKAN elemen WAE Logical View (jika berbasis web) dan Kelas

#### Diagram Kelas

Buatlah diagram kelas lengkap dengan Elemen WAE-Logical View (jika berbasis web).

### Use Case Melihat Daftar Pengaduan

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

#### Identifikasi Elemen WAE - Logical View

Bagian ini diisi dengan hasil identifikasi elemen WAE-Logical View yang merealisasikan use case (antara lain berasal dari kelas analisis yang TIDAK AKAN direalisasikan menjadi kelas biasa, tetapi akan menjadi PAGE atau FORM):

* *server page*
* *client page*
* *html form*

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Elemen Lojik WAE* | *Stereotype* |
|  |  | *Server Page* |
|  |  | *Server Page* |
|  |  | *Client Page* |

#### Identifikasi Kelas

Buat daftar kelas perancangan (berasal dari kelas analisis yang TIDAK direalisasikan menjadi elemen WAE ogical View) yang merealisasikan use case dalam tabel berikut:

|  |  |
| --- | --- |
| *No* | *Nama Kelas* |
|  |  |
|  |  |
|  |  |

#### Sequence Diagram

Buatlah diagram sequence untuk setiap skenario use case. Skenario MELIBATKAN elemen WAE Logical View (jika berbasis web) dan Kelas

#### Diagram Kelas

Buatlah diagram kelas lengkap dengan Elemen WAE-Logical View (jika berbasis web).

### Use Case Mengurutkan Pengaduan

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

#### Identifikasi Elemen WAE - Logical View

Bagian ini diisi dengan hasil identifikasi elemen WAE-Logical View yang merealisasikan use case (antara lain berasal dari kelas analisis yang TIDAK AKAN direalisasikan menjadi kelas biasa, tetapi akan menjadi PAGE atau FORM):

* *server page*
* *client page*
* *html form*

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Elemen Lojik WAE* | *Stereotype* |
|  |  | *Server Page* |
|  |  | *Server Page* |
|  |  | *Client Page* |

#### Identifikasi Kelas

Buat daftar kelas perancangan (berasal dari kelas analisis yang TIDAK direalisasikan menjadi elemen WAE ogical View) yang merealisasikan use case dalam tabel berikut:

|  |  |
| --- | --- |
| *No* | *Nama Kelas* |
|  |  |
|  |  |
|  |  |

#### Sequence Diagram

Buatlah diagram sequence untuk setiap skenario use case. Skenario MELIBATKAN elemen WAE Logical View (jika berbasis web) dan Kelas

#### Diagram Kelas

Buatlah diagram kelas lengkap dengan Elemen WAE-Logical View (jika berbasis web).

### Use Case Menghapus Pengaduan

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

#### Identifikasi Elemen WAE - Logical View

Bagian ini diisi dengan hasil identifikasi elemen WAE-Logical View yang merealisasikan use case (antara lain berasal dari kelas analisis yang TIDAK AKAN direalisasikan menjadi kelas biasa, tetapi akan menjadi PAGE atau FORM):

* *server page*
* *client page*
* *html form*

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Elemen Lojik WAE* | *Stereotype* |
|  |  | *Server Page* |
|  |  | *Server Page* |
|  |  | *Client Page* |

#### Identifikasi Kelas

Buat daftar kelas perancangan (berasal dari kelas analisis yang TIDAK direalisasikan menjadi elemen WAE ogical View) yang merealisasikan use case dalam tabel berikut:

|  |  |
| --- | --- |
| *No* | *Nama Kelas* |
|  |  |
|  |  |
|  |  |

#### Sequence Diagram

Buatlah diagram sequence untuk setiap skenario use case. Skenario MELIBATKAN elemen WAE Logical View (jika berbasis web) dan Kelas

#### Diagram Kelas

Buatlah diagram kelas lengkap dengan Elemen WAE-Logical View (jika berbasis web).

### Use Case Evaluasi dan Verifikasi Bukti Pengaduan

#### Identifikasi Elemen WAE - Logical View

Bagian ini diisi dengan hasil identifikasi elemen WAE-Logical View yang merealisasikan use case (antara lain berasal dari kelas analisis yang TIDAK AKAN direalisasikan menjadi kelas biasa, tetapi akan menjadi PAGE atau FORM):

* *server page*
* *client page*
* *html form*

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Elemen Lojik WAE* | *Stereotype* |
|  |  | *Server Page* |
|  |  | *Server Page* |
|  |  | *Client Page* |

#### Identifikasi Kelas

Buat daftar kelas perancangan (berasal dari kelas analisis yang TIDAK direalisasikan menjadi elemen WAE ogical View) yang merealisasikan use case dalam tabel berikut:

|  |  |
| --- | --- |
| *No* | *Nama Kelas* |
|  |  |
|  |  |
|  |  |

#### Sequence Diagram

Buatlah diagram sequence untuk setiap skenario use case. Skenario MELIBATKAN elemen WAE Logical View (jika berbasis web) dan Kelas

#### Diagram Kelas

Buatlah diagram kelas lengkap dengan Elemen WAE-Logical View (jika berbasis web).

### Use Case Mengubah Status Pengaduan

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

#### Identifikasi Elemen WAE - Logical View

Bagian ini diisi dengan hasil identifikasi elemen WAE-Logical View yang merealisasikan use case (antara lain berasal dari kelas analisis yang TIDAK AKAN direalisasikan menjadi kelas biasa, tetapi akan menjadi PAGE atau FORM):

* *server page*
* *client page*
* *html form*

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Elemen Lojik WAE* | *Stereotype* |
|  |  | *Server Page* |
|  |  | *Server Page* |
|  |  | *Client Page* |

#### Identifikasi Kelas

Buat daftar kelas perancangan (berasal dari kelas analisis yang TIDAK direalisasikan menjadi elemen WAE ogical View) yang merealisasikan use case dalam tabel berikut:

|  |  |
| --- | --- |
| *No* | *Nama Kelas* |
|  |  |
|  |  |
|  |  |

#### Sequence Diagram

Buatlah diagram sequence untuk setiap skenario use case. Skenario MELIBATKAN elemen WAE Logical View (jika berbasis web) dan Kelas

#### Diagram Kelas

Buatlah diagram kelas lengkap dengan Elemen WAE-Logical View (jika berbasis web).

### Use Case Mengupload Laporan

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

#### Identifikasi Elemen WAE - Logical View

Bagian ini diisi dengan hasil identifikasi elemen WAE-Logical View yang merealisasikan use case (antara lain berasal dari kelas analisis yang TIDAK AKAN direalisasikan menjadi kelas biasa, tetapi akan menjadi PAGE atau FORM):

* *server page*
* *client page*
* *html form*

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Elemen Lojik WAE* | *Stereotype* |
|  |  | *Server Page* |
|  |  | *Server Page* |
|  |  | *Client Page* |

#### Identifikasi Kelas

Buat daftar kelas perancangan (berasal dari kelas analisis yang TIDAK direalisasikan menjadi elemen WAE ogical View) yang merealisasikan use case dalam tabel berikut:

|  |  |
| --- | --- |
| *No* | *Nama Kelas* |
|  |  |
|  |  |
|  |  |

#### Sequence Diagram

Buatlah diagram sequence untuk setiap skenario use case. Skenario MELIBATKAN elemen WAE Logical View (jika berbasis web) dan Kelas

#### Diagram Kelas

Buatlah diagram kelas lengkap dengan Elemen WAE-Logical View (jika berbasis web).

### Use Case Verifikasi Tindak Lanjut Pengaduan

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

#### Identifikasi Elemen WAE - Logical View

Bagian ini diisi dengan hasil identifikasi elemen WAE-Logical View yang merealisasikan use case (antara lain berasal dari kelas analisis yang TIDAK AKAN direalisasikan menjadi kelas biasa, tetapi akan menjadi PAGE atau FORM):

* *server page*
* *client page*
* *html form*

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Elemen Lojik WAE* | *Stereotype* |
|  |  | *Server Page* |
|  |  | *Server Page* |
|  |  | *Client Page* |

#### Identifikasi Kelas

Buat daftar kelas perancangan (berasal dari kelas analisis yang TIDAK direalisasikan menjadi elemen WAE ogical View) yang merealisasikan use case dalam tabel berikut:

|  |  |
| --- | --- |
| *No* | *Nama Kelas* |
|  |  |
|  |  |
|  |  |

#### Sequence Diagram

Buatlah diagram sequence untuk setiap skenario use case. Skenario MELIBATKAN elemen WAE Logical View (jika berbasis web) dan Kelas

#### Diagram Kelas

Buatlah diagram kelas lengkap dengan Elemen WAE-Logical View (jika berbasis web).

### Use Case <nama use case 9>

### Use Case <nama use case 10>

### Use Case <nama use case 11>

### Use Case <nama use case 12>

## Diagram Kelas Keseluruhan (Fahmi Dumadi)

*Bagian ini diisi dengan diagram kelas keseluruhan.*

## Perancangan Detil Elemen Logical View (Fahmi Dumadi)

|  |  |  |
| --- | --- | --- |
| *No* | *Nama Elemen Lojik WAE* | *Stereotype* |
|  |  | *Server Page* |
|  |  | *Server Page* |
|  |  | *Client Page* |

Untuk setiap elemen:

* identifikasi atribut
* identifikasi operasi

### Stereotyped Class <nama elemen logical view>

*Nama Kelas : ……..*

*Jenis : Server Page / Client Page / HTML Form*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| *Diisi dengan signature operasi* |  |  |
|  |  |  |
|  |  |  |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Diisi dengan nama atribut* |  | *Tuliskan tipenya sesuai dengan yang dikenal pada bahasa pemrograman yang digunakan* |
|  |  |  |
|  |  |  |

### Stereotyped Class <nama elemen logical view>

*Nama Kelas : ……..*

*Jenis : Server Page / Client Page / HTML Form*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| *Diisi dengan signature operasi* |  |  |
|  |  |  |
|  |  |  |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Diisi dengan nama atribut* |  | *Tuliskan tipenya sesuai dengan yang dikenal pada bahasa pemrograman yang digunakan* |
|  |  |  |
|  |  |  |

## Perancangan Detil Kelas (Edmund Ophie)

Bagian ini diisi dengan daftar kelas dalam tabel berikut:

|  |  |
| --- | --- |
| *No* | *Nama Kelas* |
|  |  |
|  |  |
|  |  |

Untuk setiap kelas:

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

### Kelas <nama kelas>

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

*Nama Kelas : ……..*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| *Diisi dengan signature operasi* |  |  |
|  |  |  |
|  |  |  |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Diisi dengan nama atribut* |  | *Tuliskan tipenya sesuai dengan yang dikenal pada bahasa pemrograman yang digunakan* |
|  |  |  |
|  |  |  |

### Kelas <nama kelas>

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

*Nama Kelas: ……..*

|  |  |  |
| --- | --- | --- |
| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| *Diisi dengan signature operasi* |  |  |
|  |  |  |
|  |  |  |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| *Diisi dengan nama atribut* |  | *Tuliskan tipenya sesuai dengan yang dikenal pada bahasa pemrograman yang digunakan* |
|  |  |  |
|  |  |  |

## Algoritma/Query (Edmund Ophie)

*Bagian ini hanya diisi untuk kerangka algoritma untuk proses-proses yang dianggap cukup penting. Implementasi skeleton code juga sudah dapat dilakukan untuk kelas-kelas yang terdefinisi pada bahasa pemrograman tertentu. Boleh dibuat subbab per kelas.*

Contoh:

*Nama Kelas :*

*Nama Operasi :*

*Algoritma : (Algo-xxx)*

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

*Query :*

|  |  |  |
| --- | --- | --- |
| *No Query* | *Query* | *Keterangan* |
| *Q-xxx* |  | *Tuliskan fungsi dari querynya* |
|  |  |  |
|  |  |  |

## Diagram Statechart (Lutfi Fadlan)

*Bagian ini hanya diisi jika ada kelas yang kompleks. Perubahan status kelas tersebut harus digambarkan dalam bentuk diagram statechart. Boleh dibuat subba per kelas.*

## Perancangan Antarmuka (Lutfi Fadlan)

*Bagian ini diisi dengan versi awal prototipe antarmuka.*

*Selanjutnya, untuk setiap antarmuka/layar, tuliskan spesifikasi detilnya, misalnya seperti di bawah ini:*

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

| **Id\_Objek** | **Jenis** | **Nama** | **Keterangan** |
| --- | --- | --- | --- |
|  |  | *Diisi dengan string yg tampil pd layar* | Diisi dengan penjelasan reaksi sistem, misalnya membuka layar apa, link kemana. Jika menyangkut suatu kode yang cukup rumit, acu algoritma yang telah diuraikan di atas. |
| *Button1* | Button | OK | Jika diklik, akan mengaktifkan Proses AlgoXXX. |
| *RTF1* | *RTF Box* |  | Isi Teks yang disimpan pada File xxx |

Jika objek dikaitkan ke File lain (misalnya file gambar, file teks), berikan nama file terkait dan deskripsi ringkas dalam kolom keterangan

## Perancangan Representasi Persistensi Kelas (Mario Filino)

*Bagian ini diisi dengan rancangan skema basisdata dan traceability-nya terhadap kelas entity.*

## Deployment Diagram (Steve Immanuel H.)

*Bagian ini diisi dengan deployment diagram yang menggambarkan alokasi proses pada node.*

# Matriks Kerunutan (Fahmi Dumadi)

Mapping kelas dan elemen WAE logical view dengan use case terkait

|  |  |  |
| --- | --- | --- |
| **No** | **Kelas** | **Use Case Terkait** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Nama Elemen WAE (Logical View)** | **Stereotype** | **Use Case Terkait** |
|  |  | *Client Page* |  |
|  |  | *Server Page* |  |
|  |  | *HTML Form* |  |

# 