|  |
| --- |
| Untitled  **Software Design Document**  **e-Reservations**  **Peminjaman Fasilitas UII**  **Disiapkan oleh:**  **Rahmat Nurhabibie (16523029)**  **Ivan Firmansyah (16523036)**  **Fikri Azizi (16523054)**  **Muh Nizomuddin Fauza Sidiq (16523071)**  **Saputro Andi Pamungkas (16523189)**  **FAKULTAS TEKNOLOGI INDUSTRI**  **UNIVERSITAS ISLAM INDONESIA**  **YOGYAKARTA**  **April, 2017** |

DAFTAR ISI

[**BAB I.**](#_1fob9te) **Rancangan Arsitektur 3**

[**1.1**](#_3znysh7) **PUC-01 Booking Ruangan 3**

[**1.2**](#_3dy6vkm) **PUC-02 Mengecek Data Ruangan dan Fasilitas 4**

[**1.3**](#_2s8eyo1) **PUC-03 Memberikan Kritik dan Saran 4**

[**1.4**](#_26in1rg) **PUC-04 Mengelola Data Ruangan 5**

[**1.5**](#_1ksv4uv) **PUC-05 Mengelola Data Pengguna** 6

[**1.7**](#_1y810tw) **Rancangan Kelas Keseluruhan 7**

[**1.8**](#_1ci93xb) **Diagram Arsitektur 8**

[**BAB II.**](#_3as4poj) **Rancangan Perilaku 9**

[**2.1**](#_1pxezwc) **PUC-01 Booking Ruangan 9**

[**2.2**](#_49x2ik5) **PUC-02 Mengecek Data Ruangan dan Fasilitas 9**

[**2.3**](#_2p2csry) **PUC-03 Memberikan Kritik dan Saran 9**

[**2.4**](#_147n2zr) **PUC-04 Mengelola Data Ruangan 10**

[**2.5**](#_19c6y18) **PUC-05 Mengelola Data Pengguna 17**

[**BAB III.**](#_3l18frh) **Rancangan Struktur Data 19**

[**3.1**](#_206ipza) **Diagram Kelas Keseluruhan 19**

[**3.2**](#_4k668n3) **Data Pengguna 19**

[**3.3**](#_3ygebqi) **Data Pemesanan 20**

[**3.4**](#_3cqmetx) **Data Komentar 20**

[**3.5**](#_2r0uhxc) **Data Ruangan 21**

[**BAB IV.**](#_1opuj5n) **Rancangan Antarmuka 25**

[**4.1**](#_48pi1tg) **Halaman LoginForm 25**

[**4.2**](#_3mzq4wv) **Halaman MenuPage 25**

[**4.3**](#_319y80a) **Halaman PenggunaList 26**

[**4.4**](#_2fk6b3p) **Halaman PenggunaForm 27**

[**4.5**](#_1tuee74) **Halaman SkemaList 28**

[**4.6**](#_184mhaj) **Halaman SkemaForm 28**

[**4.7**](#_meukdy) **Halaman ModelForm 29**

[**4.8**](#_45jfvxd) **Halaman ProyekList 30**

[**4.9**](#_3jtnz0s) **Halaman ProyekForm 30**

[**4.10**](#_2y3w247) **Halaman DokumenList 31**

[**4.11**](#_2ce457m) **Halaman DokumenForm 32**

[**4.12**](#_1qoc8b1) **Halaman DokumenPage 32**

[**4.13**](#_14ykbeg) **Halaman DokumenPrinter 33**

[**4.14**](#_j8sehv) **Halaman ReviewList 34**

[**4.15**](#_42ddq1a) **Halaman ReviewForm 34**

[**BAB V.**](#_3gnlt4p) **Evaluasi 36**

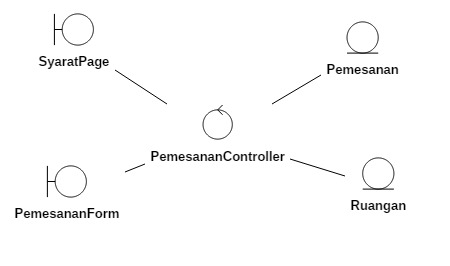
# Rancangan Arsitektur

Rancangan arsitektur pada sistem **e-Reservations** menggunakan arsitektur MVC (*Model View Controller*). Oleh karena itu, dari analisis yang sudah dibuat, perlu diidentifikasi kelas-kelas *Model*, *View* dan juga *Controller*. Dari *class diagram* pada tahap analisis di mana telah diperoleh 4 kelas, kelas-kelas menjadi bagian dari kelas *Model* pada arsitektur MVC. Selain iu, kelas *View* dan *Controller* diidentifikasi dari setiap *use case* yang ada.

Dalam perancangan pada sistem **e-Reservations** ini, dilakukan beberapa konvensi agar identifikasi kelas dapat dilakukan dengan konsisten. Kelas *Model* diperoleh dari *class diagram* pada tahap analisis. Kelas *Controller* diidentifikasi satu buah untuk setiap *use case*. Kelas *View* dibagi menjadi kelas yang menampilkan daftar suatu entitas (*List*) dan kelas yang menampilkan formulir pengisian data (*Form*) serta kelas tambahan yang berhubungan dengan sistem/perangkat lain. Berikut dijelaskan kelas-kelas MVC yang diperoleh dari setiap *use case*.

## PUC-01 *Booking* Ruangan

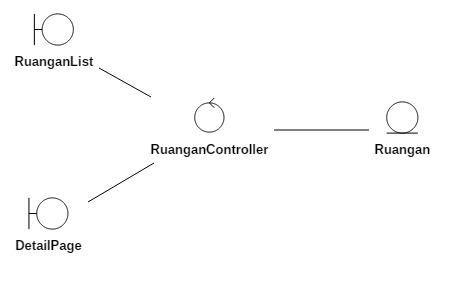
*Use case* pertama berhubungan dengan dua kelas *model* yaitu Pengguna dan Pemesanan. Kelas *view* yang diperlukan yaitu halaman yang menampilkan form (PemesananForm) dan juga halaman yang berisi syarat dan ketentuan peminjaman fasilitas (SyaratPage). Oleh karena itu, use case PUC-01 akan memiliki 5 kelas yang saling berinteraksi seperti diperlihatkan pada Gambar 11. Kelas *controller* pada *use case* ini diberi nama PemesananContoller sebagai *contoller* pemesanan peminjaman fasilitas UII di sistem **e-Reservations**.



**Gambar 11. PCD-01, *class diagram* perancangan *use case* PUC-01**

## PUC-02 Mengecek Data Ruangan dan Fasilitas

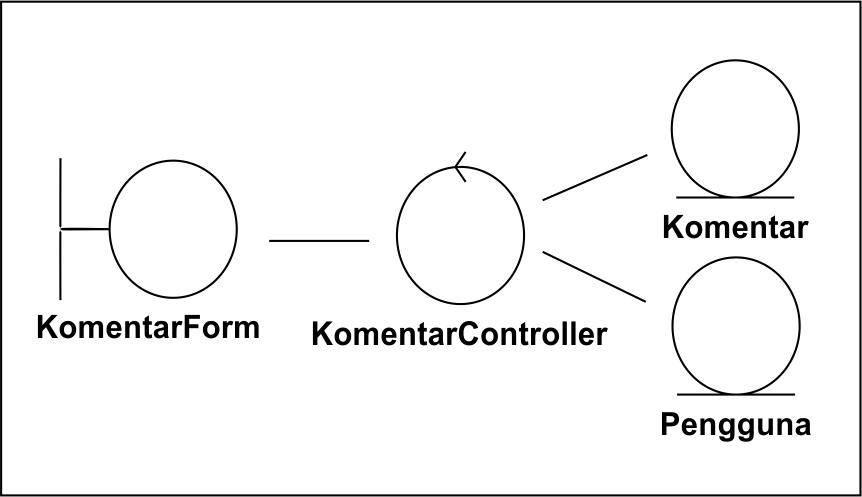
*Use case* PUC-02 akan berhubungan dengan entitas/kelas Ruangan. Kelas. Kemudian kelas *view* yang diperlukan yaitu kelas RuanganList yang akan memperlihatkan daftar ruangan atau fasilitas beserta *attribute*-nya dan kelas DetailPage yang akan menampilkan informasi detail suatu ruangan atau fasilitas. Dengan demikian, *use case* ini akan terdiri dari 4 kelas seperti diperlihatkan pada Gambar 12. Dalam *use case* ini, kelas *controller* diberi nama RuanganContoller sebagai *controller* pertama yang ada dalam sistem **e-Reservations**.



**Gambar 12. PCD-02, *class diagram* perancangan *use case* PUC-02**

## PUC-03 Memberikan Kritik dan Saran

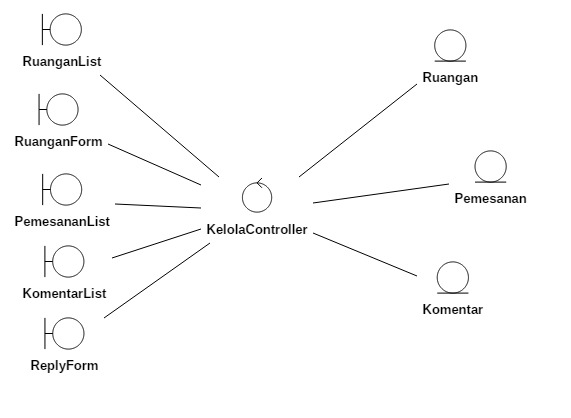
*Use case* ini akan berhubungan 1 entitas, yaitu kelas Komentar. Kemudian, kelas *view* yang diperlukan yaitu kelas yang menampilkan *form* komentar atau kritik dan saran (KomentarForm). Adapun sebuah *controller* yang dibutuhkan diberi nama KomentarController, maka *use case* ini akan memiliki 4 kelas seperti pada Gambar 13.



**Gambar 13. PCD-03, *class diagram* perancangan *use case* PUC-03**

## PUC-04 Mengelola Data Ruangan

Use case keempat ini akan berhubungan langsung dengan kelas *model* Komentar dan kelas *model* Penilaian. Namun demikian, interaksi dengan kelas *model* yang lain juga diperlukan seperti: Pengguna, Dokumentasi dan Artifak. Kelas *view* yang diperlukan yaitu ReviewList dan ReviewForm untuk mendaftarkan dan membuat komentar serta satu kelas lagi untuk menampilkan daftar dokumentasi yang akan direview (DokumenList). Seperti pada Gambar 14, *use case* Mereview akan memiliki 9 kelas yang saling berinteraksi.



**Gambar 14. PCD-04, *class diagram* perancangan *use case* PUC-04**

## PUC-05 Mengelola Data Pengguna

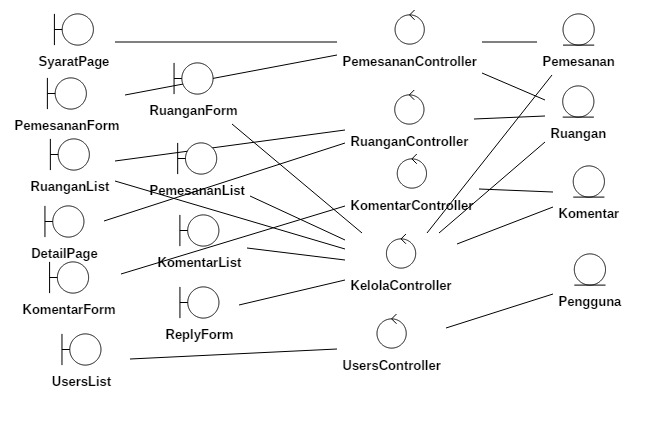
*Use case* Mencetak dokumen berarti mencetak informasi yang ada disimpan dalam kelas *model* Dokumentasi dan juga kelas *model* Artifak sebagai bagian dari dokumentasi. Kelas *view* yang diperlukan yaitu: DokumenList yang mendaftarkan dokumen yang akan dicetak, DokumenPage yang memperlihatkan tampilan awal dokumentasi dan DokumenPrinter yang menghubungkan sistem **e-Reservations** dengan mesin cetak yang ada (*printer*). Ditambah kelas CetakController, maka *use case* PUC-05 akan terdiri dari 6 kelas seperti pada Gambar 14.



**Gambar 15. PCD-05, *class diagram* perancangan *use case* PUC-05**

## Rancangan Kelas Keseluruhan

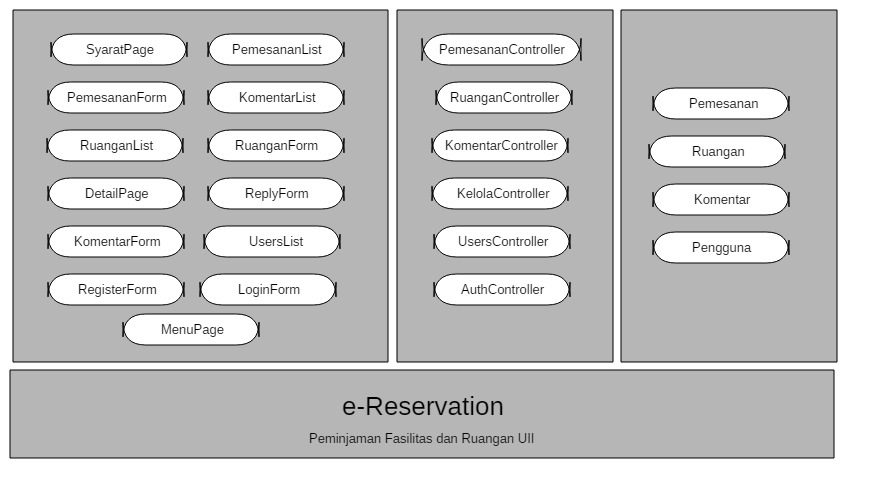
Dari kelima rancangan *class diagram* tiap *use case* yang telah disampaikan sebelumnya, digabungkan sehingga membentuk rancangan class diagram keseluruhan. Dari Gambar 17, diperoleh 8 kelas *view*, 5 kelas *controller* dan 4 kelas *model*.



**Gambar 17. PCD, *class diagram* perancangan secara keseluruhan**

## Diagram Arsitektur

Dari diagram kelas keseluruhan tahap perancangan yang telah ditunjukkan sebelumnya, dapat dibuat sebuah diagram arsitektur yang menampilkan akan seperti apa susunan perangkat lunak **e-Reservations** ini. Gambar 18 menunjukkan diagram arsitekur dari prototipe peminjaman fasilitas UII ini. Dari rancangan kelas diagram sebelumnya, masih diperlukan sebuah kelas *controller* untuk mengatur login (AuthController) dan kelas *view* pra (LoginForm) dan pasca login (MenuPage).



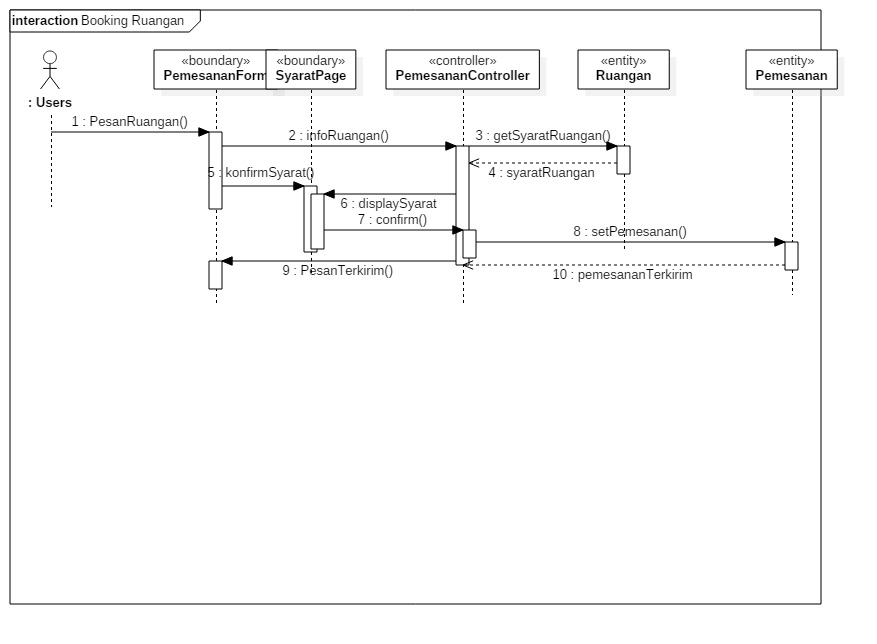
**Gambar 18. PAR, *arcitectural diagram* *software* e-Reservations**

# Rancangan Perilaku

Tahap selanjutnya dalam perancangan sistem e-Reservations ini yaitu merancang diagram sekuens untuk setiap *use case*. Berikut dijelaskan diagram sekuens untuk setiap *use case*:

## PUC-01 *Booking* Ruangan

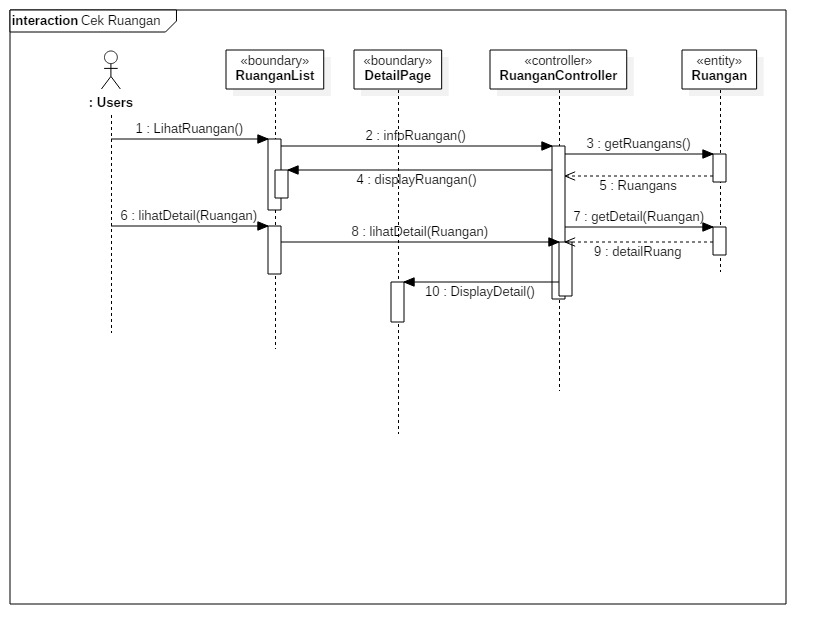
Berdasarkan diagram kelas pada Gambar 11 maka diagram sekuens untuk *use case* Membuat skema dokumentasi RPL akan tersusun atas 5 kelas. Aktor yang berperan pada *use case* ini adalah *Users* akan berinteraksi melalui kelas *view/boundary* PemesananForm. Data persisten akan disimpan ke kelas *model/entity* Pemesanan.. Perilaku *use case* ini dalam sebuah diagram sekuens diperlihatkan pada gambar 19.



**Gambar 19. *Sequence diagram* dari UC-01 *Booking* Ruangan**

## PUC-02 Mengecek Data Ruangan dan Fasilitas

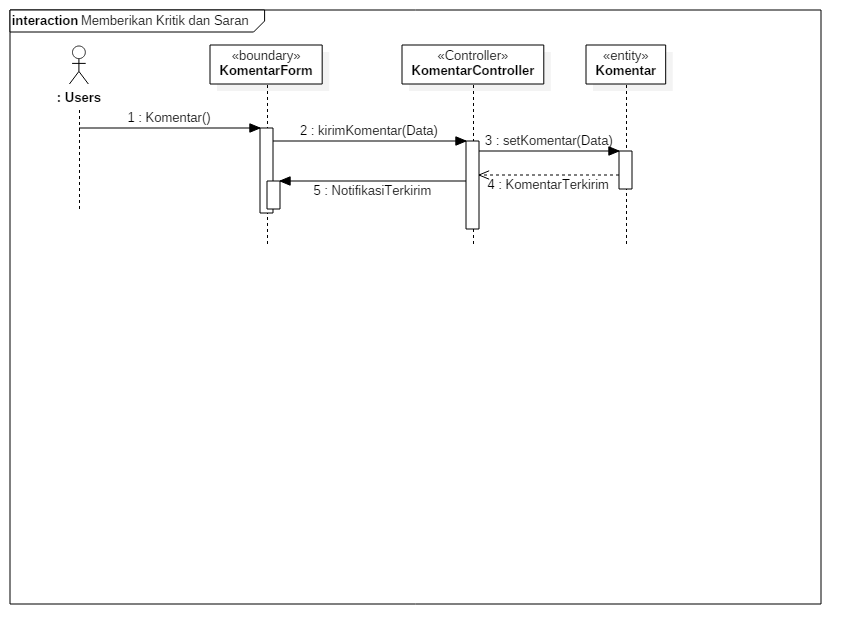
Berdasarkan diagram kelas PCD-02 maka diagram sekuens untuk *use case* PUC-02 akan tersusun atas 4 kelas. Users sebagai aktor yang berperan dalam *use case* ini akan berinteraksi melalui kelas *view* RuanganList. Data persisten akan diambil melalui kelas *model* Ruangan. Kelas Ruangan akan diambil datanya sebagai informasi ruangan dan fasilitas yang ada di UII. Gambar 20 memperlihatkan diagram sekuens untuk *use case* Mengecek data ruangan dan fasilitas ini.



**Gambar 20. *Sequence diagram* dari UC-02 *Mengecek Data Ruangan dan Fasilitas***

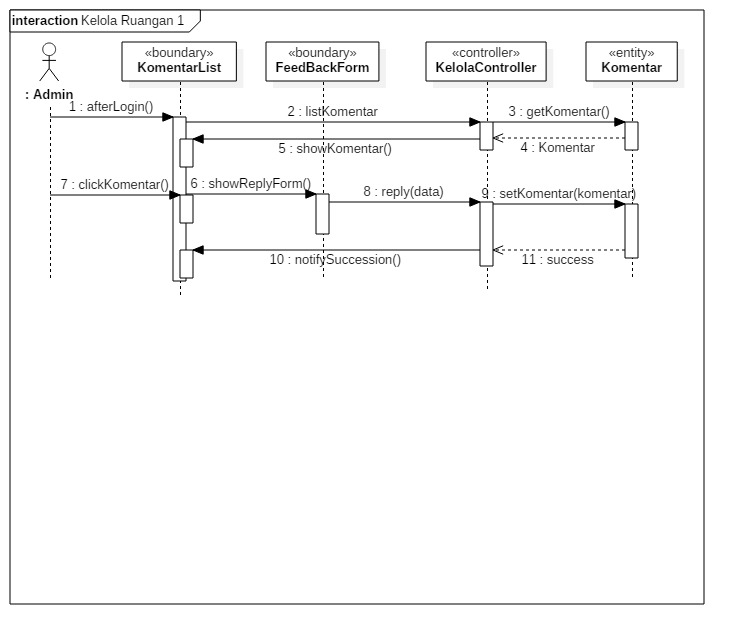
## PUC-03 Memberikan Kritik dan Saran

*Use case* memberikan kritik dan saran berisikan 5 kelas berdasarkan diagram kelas perancangan untuk *use case* PUC-03. Aktor yang berperan yaitu *Users*, akan berinteraksi dengan *boundary* KomentarForm dan data persisten akan disimpan ke dalam kelas Komentar. Diagram sekuens untuk skenario ini dijelaskan pada Gambar 21.

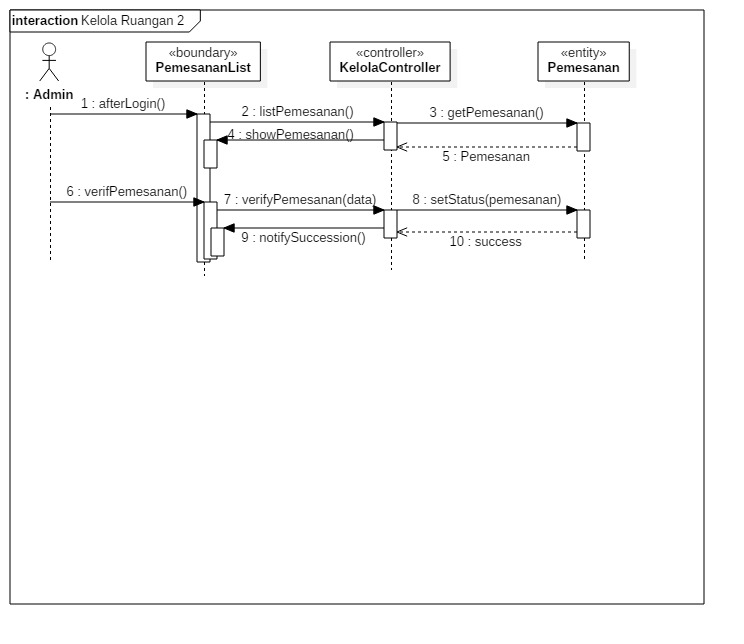


**Gambar 21. *Sequence diagram* dari UC-03 *Memberikan kritik dan saran***

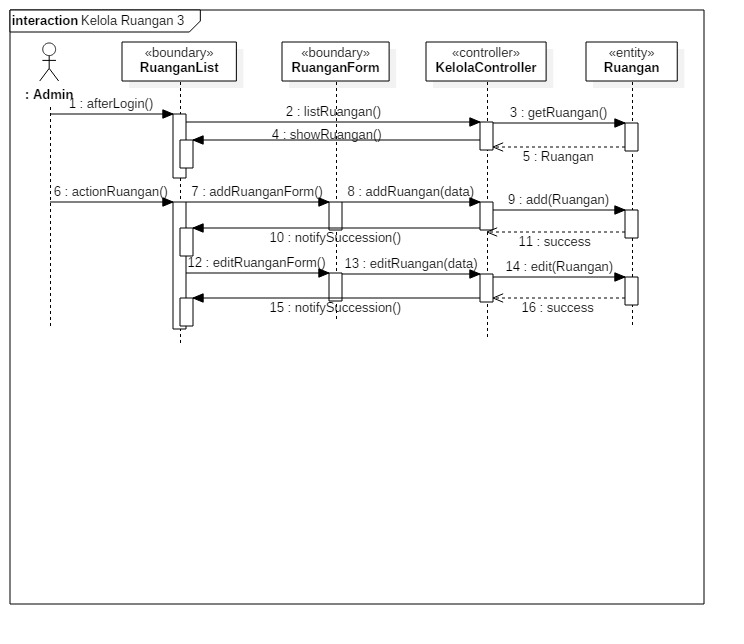
## PUC-04 Mengelola Data Ruangan

Diagram sekuens *use case* PUC-04 (Gambar 22) akan terdiri dari 9 kelas yang dibagi menjadi 3 skenario ketika mengelola pemesanan masuk, mengelola komentar, dan mengelola data ruangan tersebut. sesuai rancangan diagram kelas pada Gambar 14. Aktor yang berperan dalam use case ini adalah Admin. 

**Gambar 22. *Sequence diagram* dari UC-04 Mengelola ruangan skenario mengelola komentar**



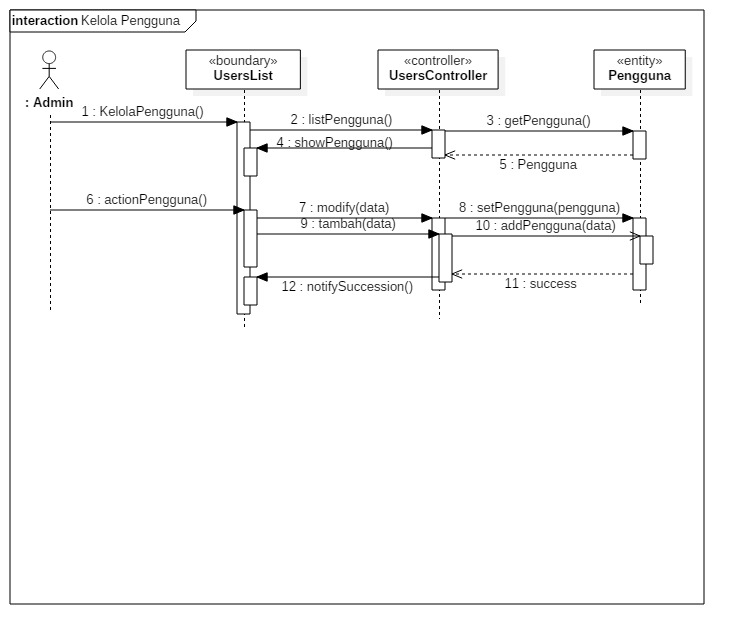
**Gambar 23. *Sequence diagram* dari UC-04 Mengelola ruangan skenario mengelola pemesanan masuk**



**Gambar 24. *Sequence diagram* dari UC-04 Mengelola ruangan skenario mengelola ruangan**

## **PUC-05 Mengelola Data Pengguna**

Diagram sekuens dari *use case* mengelola data pengguna akan ditemui 3 kelas yang saling berinteraksi

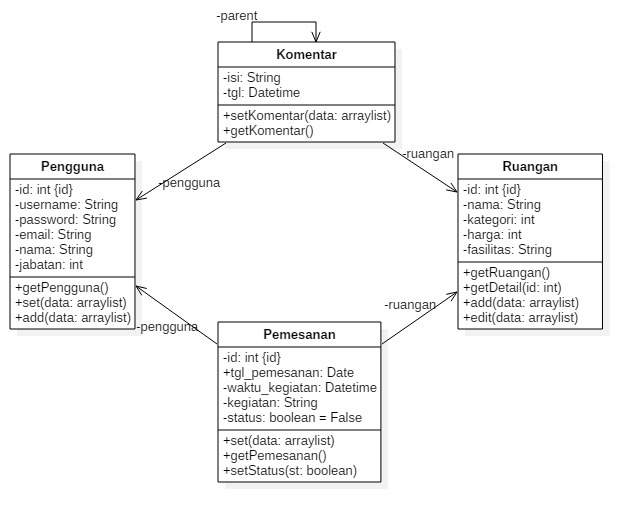


**Gambar 25. *Sequence diagram* dari UC-05 Mengelola pengguna**

# Rancangan Struktur Data

Dari diagram arsitektur pada Gambar 18, diperoleh 4 kelas *model*. Oleh karena itu, akan dirancangan 8 tabel dalam basis data yang akan menyimpan data *model* tersebut secara persisten. Berikut ditampilkan 4 data yang dirancang dalam sistem **e-Reservations**.

## Diagram Kelas Keseluruhan



**gambar 26. Diagram kelas entitas**

## 

## Data Pengguna

Data pengguna merepresentasikan data pengguna yang diperlukan dalam autentifikasi dalam sistem e-Reservations. Oleh karena itu, diperlukan *field* username,nama, dan password. Pengguna sendiri dibagi dalam 3 jabatan yaitu User, Guest dan Admin. Setiap pengguna yang ditambahkan akan langsung aktif. Pengguna yang tidak lagi aktif, penanda is\_aktif dapat diisi FALSE sehingga tidak perlu banyak menghapus data pengguna. Perancangan detilnya diperlihatkan dalam Tabel 5.

**Tabel 5. PTD-060, rancangan basis data pengguna**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Field** | **Tipe** | **Keterangan** |
| 1 | id | int | Primary Key |
| 2 | username | varchar(16) | Unique |
| 3 | password | varchar(20) |  |
| 4 | nama | varchar(64) |  |
| 5 | jabatan | tinyint |  |
| 6 | is\_aktif | boolean | Default TRUE |

## Data Pemesanan

Data skema merepresentasikan data pemesanan yang digunakan untuk meminjam ruangan. Setiap pemesanan memerlukan tgl\_pemesanan, waktu\_pemesanan, dan kegiatan untuk menandai. Setiap skema yang dibuat akan langsung aktif. Namun skema yang sudah tidak diperlukan, is\_aktif skema tersebut dapat diisi FALSE. Terakhir terdapat field timestamp untuk menyimpan waktu di mana data tersebut diubah. Perancangan detilnya diperlihatkan dalam Tabel 6.

**Tabel 6. PTD-011, rancangan basis data pemesanan**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Field** | **Tipe** | **Keterangan** |
| 1 | id | int | Primary Key |
| 2 | id\_pengguna | int | Foreign Key (Field id data pengguna) |
| 3 | id\_ruangan | int | Foreign Key (Field id data ruangan) |
| 4 | tgl\_pemesanan | date |  |
| 5 | waktu\_kegiatan | date |  |
| 6 | kegiatan | varchar(255) |  |
| 7 | status | boolean | Default FALSE |

## Data Komentar

Data komentar merepresentasikan data kritik dan saran yang diterima oleh sistem. dimana diperlukan pula id\_parent sebagai penanda bahwa suatu komentar adalah anak atau balasan dari suatu komentar lain. Tabel 7 memperlihatkan rancangan basis data untuk model ini.

**Tabel 7. PTD-012, rancangan basis data komentar**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Field** | **Tipe** | **Keterangan** |
| 1 | id | int | Primary Key |
| 2 | id\_pengguna | int | Foreign Key (Field id data pengguna) |
| 3 | id\_ruangan | int | Foreign Key (Field id data ruangan) |
| 4 | isi | varchar(255) |  |
| 5 | tgl | date |  |
| 6 | id\_parent | int |  |

## Data Ruangan

Data ruangan akan merepresentasikan obyek-obyek ruangan dan fasilitas yang ada di UII untuk dipesan pemakaian tempatnya. Adapun deskripsi dari setiap ruangan dapat dijelaskan dalam sebuah field fasilitas dengan tipe data TEXT, sehingga dapat menampung isi deskripsi dari ruangan. Tabel 8 memperlihatkan rancangan basis data ruangan.

**Tabel 8. PTD-013, rancangan basis data ruangan**

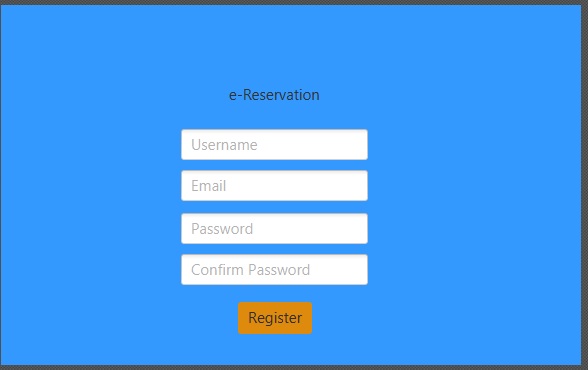
|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Nama Field** | **Tipe** | **Keterangan** |
| 1 | id | int | Primary Key |
| 2 | nama | varchar(65) |  |
| 3 | kategori | int |  |
| 4 | harga | int |  |
| 5 | fasilitas | TEXT |  |

# Rancangan Antarmuka

Dari diagram arsitektur pada Gambar 18, diperoleh 15 kelas *view*. Oleh karena itu, akan dirancangan 15 file untuk menyajikan tampilan/halaman website sebagai antarmuka dalam sistem **e-Reservations** ini. Berikut ditampilkan sketsa rancangan-rancangan antarmuka tersebut.

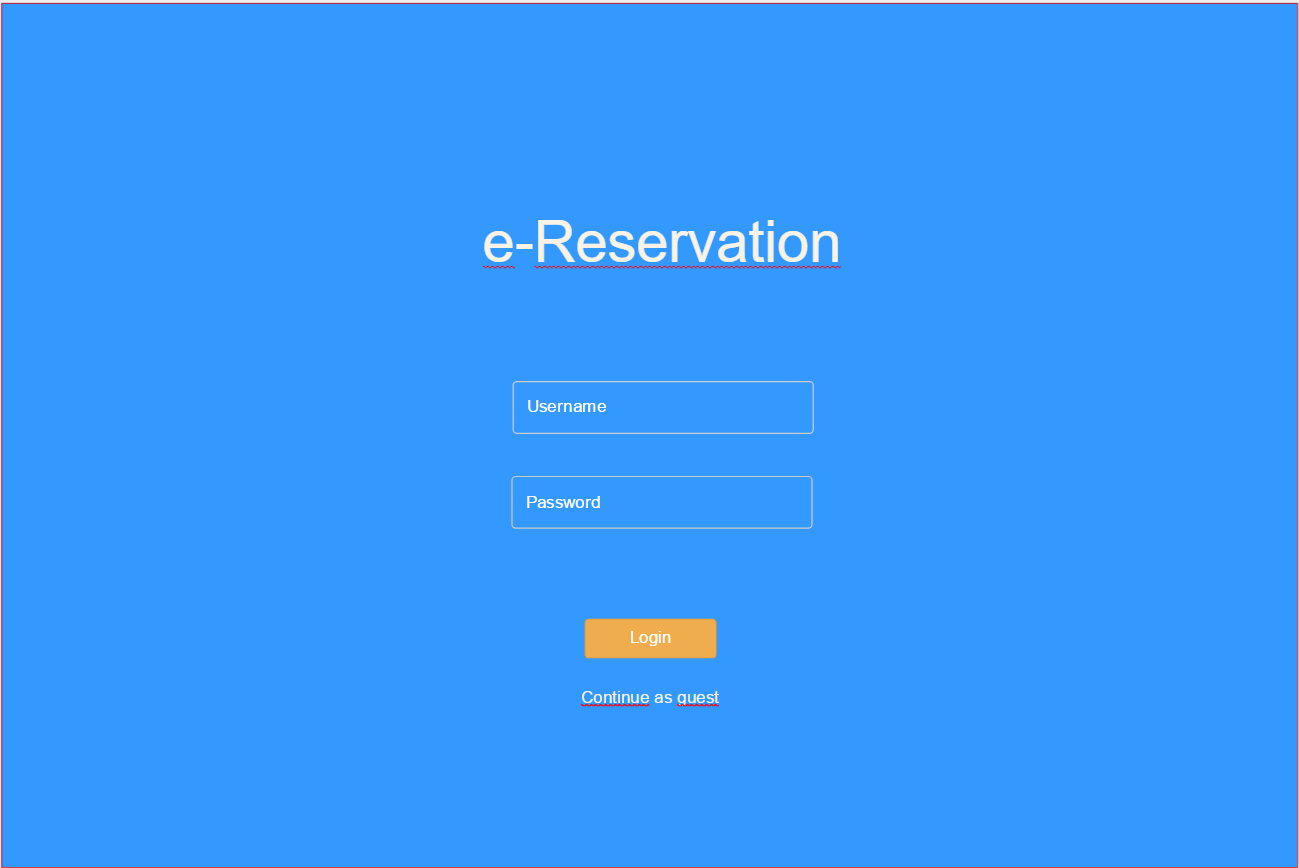
## Halaman Register

Halaman Register Form menampilkan halaman awal sistem **e-Reservation** sebelum seseorang masuk ke halaman LoginForm.Di halaman ini terdapat gambaran umum mengenai syarat /aturan untuk melakukan login atau autentikasi.Rancangan halaman ini di perlihatkan pada Gambar 1.8



## Halaman LoginForm

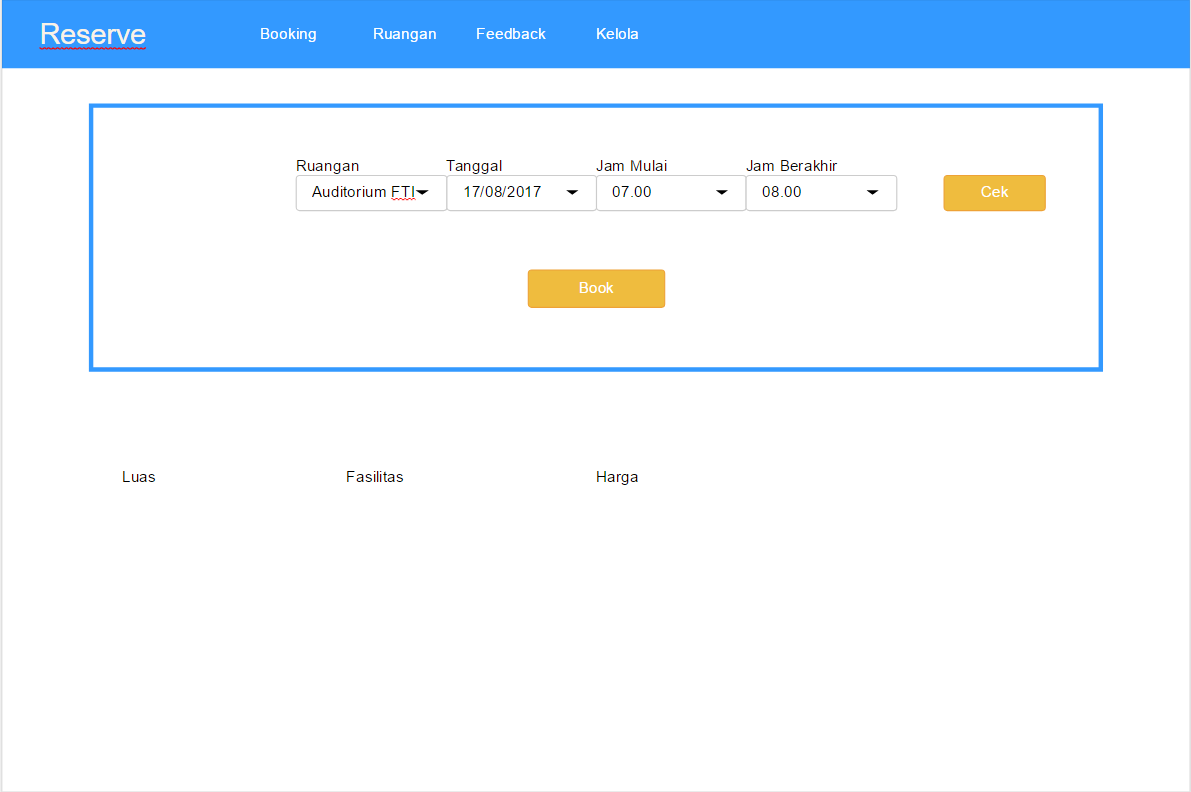
Halaman LoginForm memperlihatkan halaman awal sistem **e-Reservations** sebelum seseorang terautentifikasi masuk ke dalam sistem. Di halaman ini terdapat gambaran umum mengenai proyek-proyek yang ada dan sebuah form untuk melakukan autentifikasi atau login. Rancangan halaman ini diperlihatkan pada Gambar 28.



**Gambar 28. PID-001, rancangan antarmuka form Login**

## Halaman RuanganList dan DetailPage

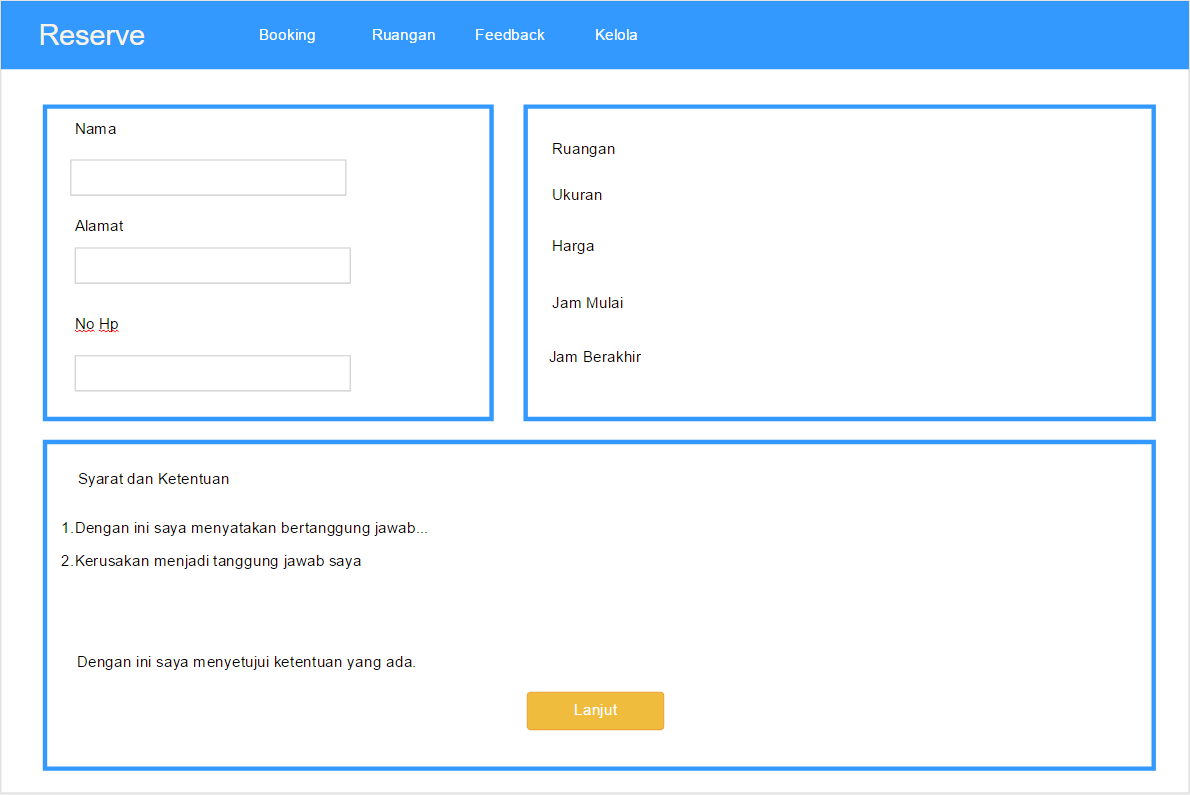
Halaman Ruangan (Gambar 29) memperlihatkan halaman awal sistem **e-Reservations** seteleh seorang pengguna terautentifikasi dan masuk ke dalam sistem. Di halaman ini, akan memperlihatkan daftar ruangan dan fasilitas yang ada di UII beserta status peminjamannya. Kemudian jika yang terautentifikasi adalah Admin maka halaman awalnya ditambah dengan menu *edit*.



**Gambar 29. PID-002, rancangan antarmuka halaman menu Utama: RuanganList.**

## Halaman PemesananForm dan SyaratPage

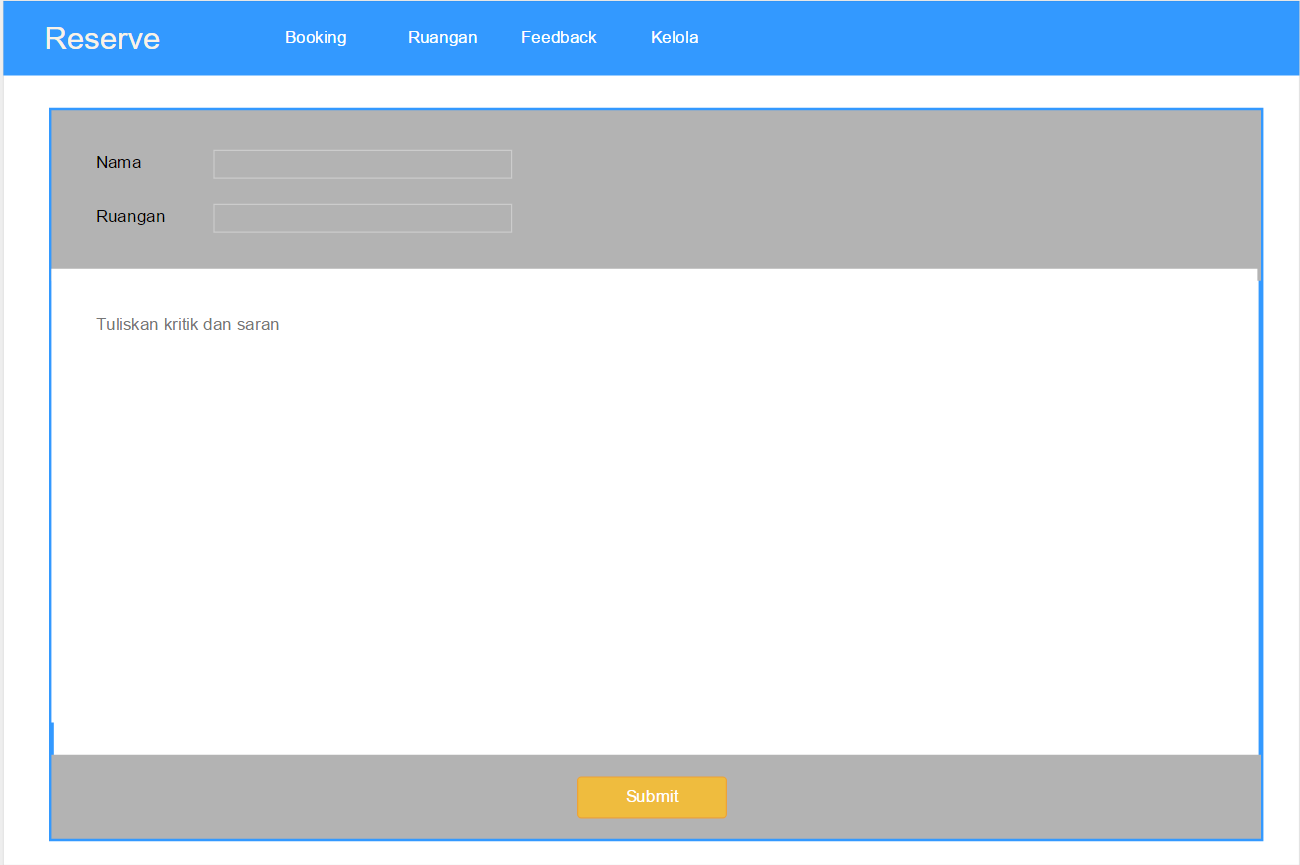
Halaman PemesananForm dan SyaratPage akan menampilkan form pemesanan untuk peminjaman ruangan dan fasilitas yang ada di UII. Rancangan halaman PemesananForm dan SyaratPage diperlihatkan Gambar 30.



**Gambar 30. PID-061, rancangan antarmuka halaman PemesananForm dan SyaratPage**

## Halaman KomentarForm

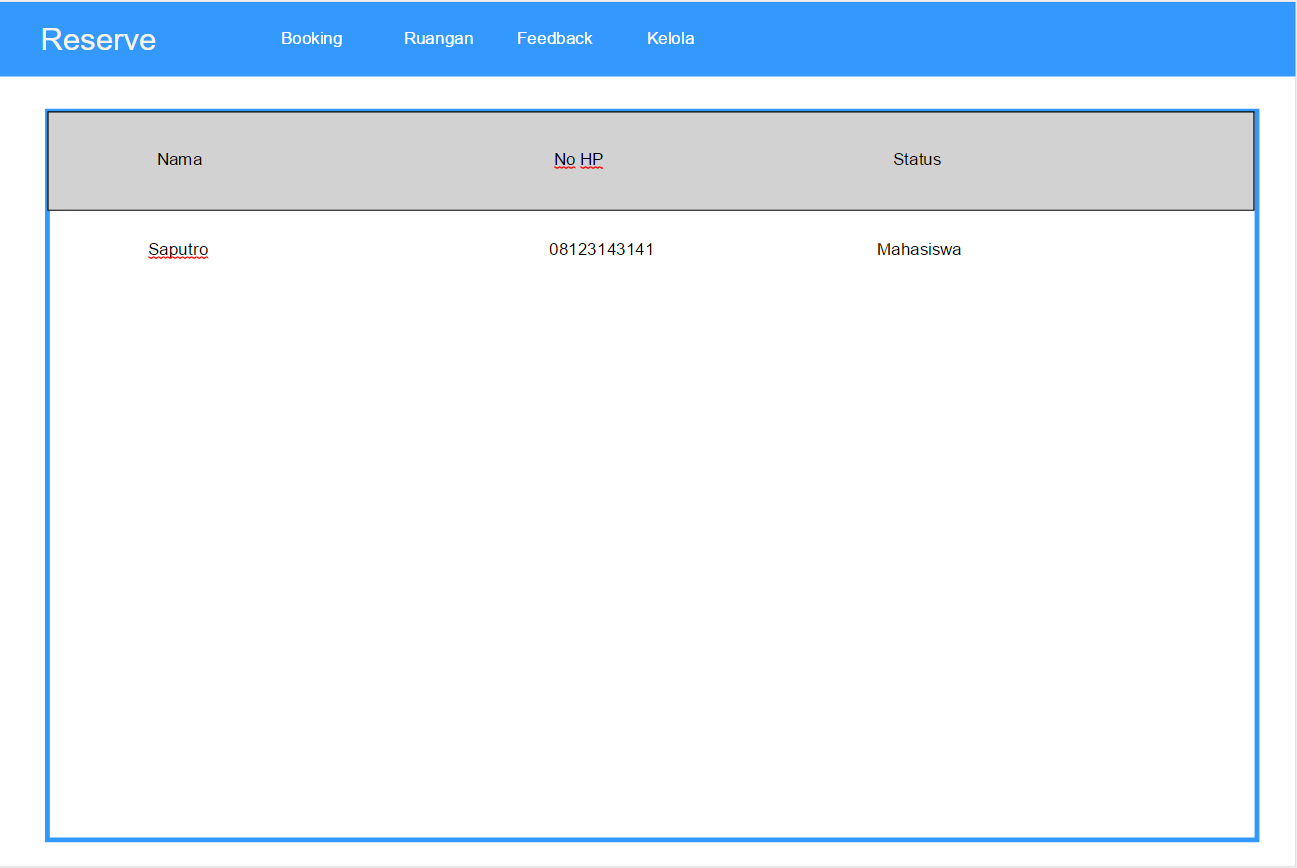
Halaman KomentarForm merupakan halaman yang akan menampilkan formulir kritik dan saran terhadap ruangan dan fasilitas yang ada di UII. Gambar 31 memperlihatkan rancangan halaman KomentarForm.



**Gambar 31. PID-062, rancangan antarmuka halaman Form Pengguna**

## Halaman UsersList

Halaman UserList akan menampilkan daftar pengguna. Tampilan ini diperlihatkan pada Gambar 32.



**Gambar 32. PID-011, rancangan antarmuka halaman UserList**

# Evaluasi

Evaluasi prototipe perangkat lunak peminjaman fasilitas UII diawali dengan terlebih dahulu melakukan penelusuran (*tracing*) terhadap artifak-artifak dokumentasi yang telah dibuat untuk sistem **e-Reservations**. Penelusuran dimulai dari tahap analisis kebutuhan dalam kebutuhan perangkat lunak, *use case* dan diagram aktivitas hingga tahap perancangan model dalam diagram kelas, diagram sekuens, rancangan tabel dan rancangan antarmuka.

Dari Tabel 14, diperlihatkan daftar penelusuran dari tahap analisis hingga tahap perancangan, ditandai dengan kode-kode yang sesuai dengan artifak analisis dan perancangan yang telah disampaikan di bab sebelumnya. Dari tabel tersebut, diperoleh bahwa sistem e-Reservations ini telah dirancang sesuai kebutuhan perangkat lunak yang telah ditentukan. Keenam kebutuhan fungsionalitas memiliki representasi model analisis dan model rancangannya sehingga setiap kebutuhan tersebut siap untuk diimplementasikan.

Evaluasi prototipe selanjutnya dilakukan dengan menyerahkan hasil analisis dan perancangan kepada *programmer* untuk kemudian diimplementasikan olehnya. Proses evaluasinya adalah proses pengujian hasil analisis dan perancangan **e-Reservations**, apakah bisa diimplementasikan dengan baik dan benar sesuai harapan atau tidak. Dari hasil pengujian tersebut, dinyatakan secara langsung oleh *programmer* bahwa analisis dan perancangan yang telah dibuat ini cukup mudah untuk dipahami. Namun demikian, masih ada beberapa poin di mana *programmer* masih bertanya dan memerlukan tambahan informasi karena kurang jelasnya analisis dan perancangan tersebut. Secara umum, terdapat 12 pertanyaan yang disampaikan *programmer* kepada *analyst*.

**Tabel 14. Daftar *tracability* kebutuhan fungsional hingga perancangan e-Reservations**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Kebutuhan Fungsional** | **Use Case** | **Activity Diagram** | **Class Diagram** | **Sequence Diagram** | **Tabel Basis Data** | **Interface** |
| PKF-01 | PUC-01 | PAD-01 | PCD-01 | PSD-01 | PTD-011  PTD-012  PTD-013 | PID-011  PID-012  PID-013 |
| PKF-02 | PUC-02 | PAD-02 | PCD-02 | PSD-02 | PTD-011  PTD-020 | PID-021  PID-022 |
| PKF-03 | PUC-03 | PAD-03 | PCD-03 | PSD-03 | PTD-011 | PID-031  PID-032  PID-033 |
| PKF-04 | PUC-04 | PAD-04 | PCD-04 | PSD-04a  PSD-04b  PSD-04c | PTD-041  PTD-042 | PID-031  PID-041  PID-042 |
| PKF-05 | PUC-05 | PAD-05 | PCD-05 | PSD-05 | PTD-031  PTD-032 | PID-031  PID-033  PID-051 |