**DPPL-04**

**DESKRIPSI PERANCANGAN PERANGKAT LUNAK**

**Portal Course – Sistem Belajar Mengajar Coding dan Design Online**

**[PC]**

untuk:

Pengajar Kursus

Dipersiapkan oleh:

Bayu Muhammad Iqbal (1301184241)

R. Ardityo Cahyo Putro Hutomo (1301183507)

Nabila Alissa Alifia Putri (1301184417)

Nurul Khaeriyah Putri M. (1301184048)

Program Studi S1 Teknik Informatika – Fakultas Informatika

Universitas Telkom

Jalan Telekomunikasi Terusan Buah Batu, Bandung

Indonesia

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Prodi S1- Informatika**  **Universitas Telkom** | **Nomor Dokumen** | | **Halaman** |
| ***DPPL-04 <xx:no grp>*** | | *<#>/<jml #* |
| **Revisi** | *<nomor revisi>* | *Tgl: <isi tanggal>* |

**DAFTAR PERUBAHAN**

|  |  |
| --- | --- |
| **Revisi** | **Deskripsi** |
| **A** | Lengkapi dokumen, perbaiki deskripsi dokumen, perbaiki matriks keturunan |
| **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** |
| 12  67 | Lengkapi dokumen    Perbaiki deskripsi dokumen      Perbaiki matriks keturunan |  |  |

**Daftar Isi**

[**1. Pendahuluan** 9](#_Toc39781259)

[***1.1*** ***Tujuan Penulisan Dokumen*** 9](#_Toc39781260)

[***1.2*** ***Lingkup Masalah*** 9](#_Toc39781261)

[***1.3*** ***Definisi dan Istilah*** 9](#_Toc39781262)

[***1.4*** ***Referensi*** 9](#_Toc39781263)

[***1.5*** ***Sistematika Pembahasan*** 10](#_Toc39781264)

[**2** **Deskripsi Perancangan Global** 11](#_Toc39781265)

[***2.1*** ***Deskripsi Arsitektural*** 11](#_Toc39781266)

[***2.2*** ***Deskripsi Komponen*** 12](#_Toc39781267)

[**3** **Perancangan Rinci** 13](#_Toc39781268)

[***3.1*** ***Realisasi Use Case*** 13](#_Toc39781269)

[**3.1.1** **Use Case #1 <Membeli Kelas>** 14](#_Toc39781270)

[**3.1.1.1** **Perancangan Antarmuka Usecase #1** 15](#_Toc39781271)

[**3.1.1.1.1** **Tabel Identifikasi Antarmuka / Layar / Page** 17](#_Toc39781272)

[**3.1.1.2** **Identifikasi Object Baru** 17](#_Toc39781273)

[**3.1.1.3** **Robustness Diagram** 18](#_Toc39781274)

[**3.1.1.4** **Diagram Kelas** 18](#_Toc39781275)

[**3.1.1.5** **Sequence Diagram** 18](#_Toc39781276)

[**3.1.2** **Use Case #2 <Melihat Kelas>** 19](#_Toc39781277)

[**3.1.2.1** **Perancangan Antarmuka Usecase #2** 19](#_Toc39781278)

[**3.1.2.1.1** **Tabel Identifikasi Antarmuka / Layar / Page** 20](#_Toc39781279)

[**3.1.2.2** **Identifikasi Object Baru** 20](#_Toc39781280)

[**3.1.2.3** **Robustness Diagram** 20](#_Toc39781281)

[**3.1.2.4** **Diagram Kelas** 21](#_Toc39781282)

[**3.1.2.5** **Sequence Diagram** 21](#_Toc39781284)

[**3.1.3** **Use Case #3 <Diskusi Forum>** 22](#_Toc39781285)

[**3.1.3.1** **Perancangan Antarmuka Usecase #3** 22](#_Toc39781286)

[**3.1.3.1.1** **Tabel Identifikasi Antarmuka / Layar / Page** 24](#_Toc39781287)

[**3.1.3.2** **Identifikasi Object Baru** 25](#_Toc39781288)

[**3.1.3.3** **Robustness Diagram** 25](#_Toc39781289)

[**3.1.3.4** **Diagram Kelas** 27](#_Toc39781290)

[**3.1.3.5** **Sequence Diagram** 27](#_Toc39781291)

[**3.1.4** **Use Case #4 <Menambahkan Materi>** 28](#_Toc39781292)

[**3.1.4.1** **Perancangan Antarmuka Usecase #4** 28](#_Toc39781293)

[**3.1.4.1.1** **Tabel Identifikasi Antarmuka / Layar / Page** 29](#_Toc39781294)

[**3.1.4.2** **Identifikasi Object Baru** 29](#_Toc39781295)

[**3.1.4.3** **Robustness Diagram** 30](#_Toc39781296)

[**3.1.4.4** **Diagram Kelas** 30](#_Toc39781297)

[**3.1.4.5** **Sequence Diagram** 30](#_Toc39781298)

[**3.1.5** **Use Case #5 <Mengubah Materi>** 31](#_Toc39781299)

[**3.1.5.1** **Perancangan Antarmuka Usecase #5** 31](#_Toc39781300)

[**3.1.5.1.1** **Tabel Identifikasi Antarmuka / Layar / Page** 32](#_Toc39781301)

[**3.1.5.2** **Identifikasi Object Baru** 32](#_Toc39781302)

[**3.1.5.3** **Robustness Diagram** 33](#_Toc39781303)

[**3.1.5.4** **Diagram Kelas** 34](#_Toc39781304)

[**3.1.5.5** **Sequence Diagram** 34](#_Toc39781305)

[**3.1.6** **Use Case #6 <Melihat Materi>** 36](#_Toc39781306)

[**3.1.6.1** **Perancangan Antarmuka Usecase 6** 36](#_Toc39781307)

[**3.1.6.1.1** **Tabel Identifikasi Antarmuka / Layar / Page** 39](#_Toc39781308)

[**3.1.6.2** **Identifikasi Object Baru** 40](#_Toc39781309)

[**3.1.6.3** **Robustness Diagram** 40](#_Toc39781310)

[**3.1.6.4** **Diagram Kelas** 41](#_Toc39781311)

[**3.1.6.5** **Sequence Diagram** 42](#_Toc39781312)

[**3.1.7** **Use Case #7 <Menghapus Materi>** 44](#_Toc39781313)

[**3.1.7.1** **Perancangan Antarmuka Usecase 7** 44](#_Toc39781314)

[**3.1.7.1.1** **Tabel Identifikasi Antarmuka / Layar / Page** 45](#_Toc39781315)

[**3.1.7.2** **Identifikasi Object Baru** 45](#_Toc39781316)

[**3.1.7.3** **Robustness Diagram** 46](#_Toc39781317)

[**3.1.7.4** **Diagram Kelas** 47](#_Toc39781318)

[**3.1.7.5** **Sequence Diagram** 47](#_Toc39781319)

[**3.1.8** **Use Case #8 <Merekapitulasi Materi>** 49](#_Toc39781320)

[**3.1.8.1** **Perancangan Antarmuka Usecase #8** 49](#_Toc39781321)

[**3.1.8.1.1** **Tabel Identifikasi Antarmuka / Layar / Page** 50](#_Toc39781322)

[**3.1.8.2** **Identifikasi Object Baru** 50](#_Toc39781323)

[**3.1.8.3** **Robustness Diagram** 51](#_Toc39781324)

[**3.1.8.4** **Diagram Kelas** 51](#_Toc39781325)

[**3.1.8.5** **Sequence Diagram** 52](#_Toc39781326)

[**3.1.9** **Use Case #9 <Merekapitulasi Transaksi Pembayaran>** 52](#_Toc39781327)

[**3.1.9.1** **Perancangan Antarmuka Usecase #9** 52](#_Toc39781328)

[**3.1.9.1.1** **Tabel Identifikasi Antarmuka / Layar / Page** 53](#_Toc39781329)

[**3.1.9.2** **Identifikasi Object Baru** 53](#_Toc39781330)

[**3.1.9.3** **Robustness Diagram** 54](#_Toc39781331)

[**3.1.9.4** **Diagram Kelas** 54](#_Toc39781332)

[**3.1.9.5** **Sequence Diagram** 55](#_Toc39781333)

[**3.1.10** **Use Case #10 <Merekapitulasi Anggota Terdaftar>** 55](#_Toc39781334)

[**3.1.10.1** **Perancangan Antarmuka Usecase #10** 55](#_Toc39781335)

[**3.1.10.1.1** **Tabel Identifikasi Antarmuka / Layar / Page** 56](#_Toc39781336)

[**3.1.10.2** **Identifikasi Object Baru** 56](#_Toc39781337)

[**3.1.10.3** **Robustness Diagram** 57](#_Toc39781338)

[**3.1.10.4** **Diagram Kelas** 57](#_Toc39781339)

[**3.1.10.5** **Sequence Diagram** 57](#_Toc39781340)

[**3.1.11** **Use Case #11 <Login>** 58](#_Toc39781341)

[**3.1.11.1** **Perancangan Antarmuka Usecase #11** 58](#_Toc39781342)

[**3.1.11.1.1** **Tabel Identifikasi Antarmuka / Layar / Page** 59](#_Toc39781343)

[**3.1.11.2** **Identifikasi Object Baru** 59](#_Toc39781344)

[**3.1.11.3** **Robustness Diagram** 60](#_Toc39781345)

[**3.1.11.4** **Diagram Kelas** 62](#_Toc39781346)

[**3.1.11.5** **Sequence Diagram** 62](#_Toc39781347)

[**3.1.12** **Use Case #12 <Registrasi>** 63](#_Toc39781348)

[**3.1.12.1** **Perancangan Antarmuka Usecase #12** 63](#_Toc39781349)

[**3.1.12.1.1** **Tabel Identifikasi Antarmuka / Layar / Page** 66](#_Toc39781350)

[**3.1.12.2** **Identifikasi Object Baru** 67](#_Toc39781351)

[**3.1.12.3** **Robustness Diagram** 67](#_Toc39781352)

[**3.1.12.4** **Diagram Kelas** 69](#_Toc39781353)

[**3.1.12.5** **Sequence Diagram** 69](#_Toc39781354)

[**4** **Perancangan Detil** 70](#_Toc39781355)

[**4.1** **Perancangan Detil Kelas** 70](#_Toc39781356)

[**4.2** **Perancangan Kelas Persistensi (\*\*\*Basis Data Skema Tabel)** 72](#_Toc39781357)

[**4.3** **Perancangan Algoritma** 72](#_Toc39781358)

[**4.3.1** **Algoritma #1** 72](#_Toc39781359)

[**4.3.2** **Algoritma #2** 74](#_Toc39781360)

[**4.3.3** **Algoritma #3** 74](#_Toc39781361)

[**4.3.4** **Algoritma #4** 75](#_Toc39781362)

[**4.3.5** **Algoritma #5** 75](#_Toc39781363)

[**4.3.6** **Algoritma #6** 76](#_Toc39781364)

[**4.3.7** **Algoritma #7** 76](#_Toc39781365)

[**4.3.8** **Algoritma #8** 77](#_Toc39781366)

[**4.3.9** **Algoritma #9** 77](#_Toc39781367)

[**4.3.10** **Algoritma #10** 78](#_Toc39781368)

[**4.3.11** **Algoritma #11** 78](#_Toc39781369)

[**4.3.12** **Perancangan Query** 80](#_Toc39781370)

[**5** **Matriks Kerunutan (Requirement Traceability Matrix)** 81](#_Toc39781371)

# **1. Pendahuluan**

## ***Tujuan Penulisan Dokumen***

Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) ini dibuat dengan tujuan untuk mempermudah jalannya Belajar Mengajar Kursus Komputerisasi dalam hal Pengembangan Web atau Mobile secara Online. Dokumen ini juga bertujuan sebagai bahan acuan dan pertimbangan antara 2 (dua) pihak terkait. Pihak tersebut yakni Pengembang dan Pengguna. Untuk Pengembangan, Dokumen ini digunakan sebagai bahan acuan dan pertimbangan dalam hal pengembangan Sistem/ Program. Sedangkan Pengguna disini terbagi menjadi 2 (dua), yakni Pengguna Kursus dan Pengajar Kursus. Untuk Pengguna, Dokumen ini digunakan untuk pertimbangan dari *Spesification Requirement* yang akan dibutuhkan nanti oleh Pengguna untuk menjalankan Sistem/ Program agar dapat dijalankan dengan lancar. Piranti lunak pembelajaran online yang dibuat ini berfungsi untuk membantu kegiatan belajar dan mengajar Pengajar Kursus maupun Pengguna Kursus.

## ***Lingkup Masalah***

Portal Course merupakan Program Kursus Belajar Online terkait dengan Code, Design, Programming, dan Karir dibidang IT. Program ini dirancang dalam basis Web. Dalam program membutuhkan konektivitas Internet dan juga meliputi sistem informasi terdistribusi yang berakar atau terstruktur dan diolah oleh server Pusat. Kursus Online ini dapat diikuti oleh ruang lingkup pelajar, Mahasiswa, ataupun Pekerja dimana Program ini bertujuan untuk mengatasi permasalahan mengenai pemrograman maupun desain.

## ***Definisi dan Istilah***

*Semua definisi dan singkatan yang digunakan dalam dokumen ini dan penjelasannya*

* DPPL (Dokumen Deskripsi Perancangan Perangkat Lunak) bertujuan untuk mendefinisikan perancangan perangkat lunak yang akan dikembangkan. Dokumen DPPL tersebut digunakan oleh pengembang perangkat lunak sebagai acuan untuk implementasi pada tahap selanjutnya.
* Flowchart adalah representasi grafik dari sistem informasi, proses-proses, aliran-aliran data logis, masukan-masukan, keluaran-keluaran, aliran-aliran data fisik dan kegiatan-kegiatan operasi yang berhubungan dengan sistem informasi.
* ERD (Entity Relationship Diagram) adalah diagram dan notasi yang digunakan untuk merepresentasikan struktur data statis pada perangkat.
* Sequence Diagram menggambarkan interaksi antara aktor dan sistem dalam suatu skenario use case, menggambarkan interaksi seluruh objek yang terkait dengan skenario.
* Activity diagram adalah diagram yang memodelkan workflow proses bisnis dan urutan aktivitas dalam sebuah proses.
* Component Diagram adalah diagram yang menggambarkan physical struktur dari sebuah sistem dan digunakan untuk menggambarkan bagaimana code program dibagi menjadi beberapa komponen, serta menggambarkan hubungan antar komponen tersebut.

## ***Referensi***

* Ardityo C., Bayu I., Nabila A., dan Nurul K. 2019. *Spesifikasi Kebutuhan Perangkat Lunak Portal Course.* Jurusan Informatika, Telkom University
* Staff ITB. 2000. *Panduan Penggunaan dan Pengisian Spesifikasi Kebutuhan Perangkat Lunak (SKPL).* Jurusan Teknik Informatika, Institut Teknologi Bandung.
* Herdyan S. dan Deni R. 2014. *Spesifikasi Kebutuhan Perangkat Lunak Sistem Informasi Nilai Akademik (SINA).* Program Keahlian Manajemen Informatika, Institut Pertanian Bogor.
* Andi P., Aang N., dan Arifa D. 2011. *Spesifikasi Kebutuhan Perangkat Lunak Vending Machine System.* Departemen Ilmu Komputer, Institut Pertanian Bogor.
* Ardhi A., Dwi S., Rif’at, dan Angga G. 2014. *Spesifikasi Kebutuhan Perangkat Lunak Taking Order Application for Sale.* Jurusan Teknik Elektro, Universitas Negeri Malang.
* Smartdraw.com. 2020. *Component Diagram.* Diakses: 05 Mei 2020. <https://www.smartdraw.com/component-diagram/> (Online).
* Bell, Donald. 2004. *The component diagram. Diakses 06 Mei 2020.* [*https://developer.ibm.com/technologies/web-development/articles/the-component-diagram/*](https://developer.ibm.com/technologies/web-development/articles/the-component-diagram/) *(Online).*

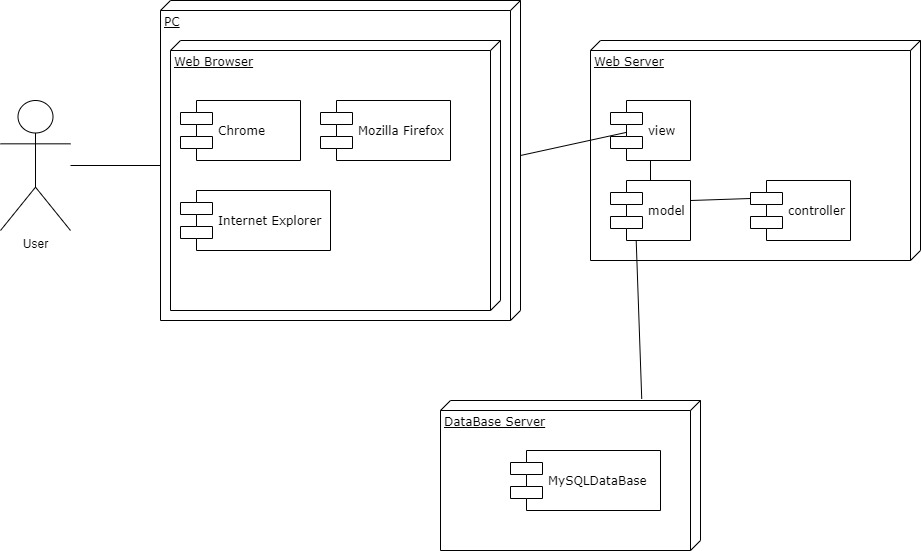
## ***Sistematika Pembahasan***

Dokumen Deskripsi Perancangan Perangkat Lunak (DPPL) merupakan Dokumen penjelasan kebutuhan tingkat lanjut atau terperinci mengenai Sistem/ Program dari Portal Course, Penjelasan terbagi menjadi beberapa bagian :

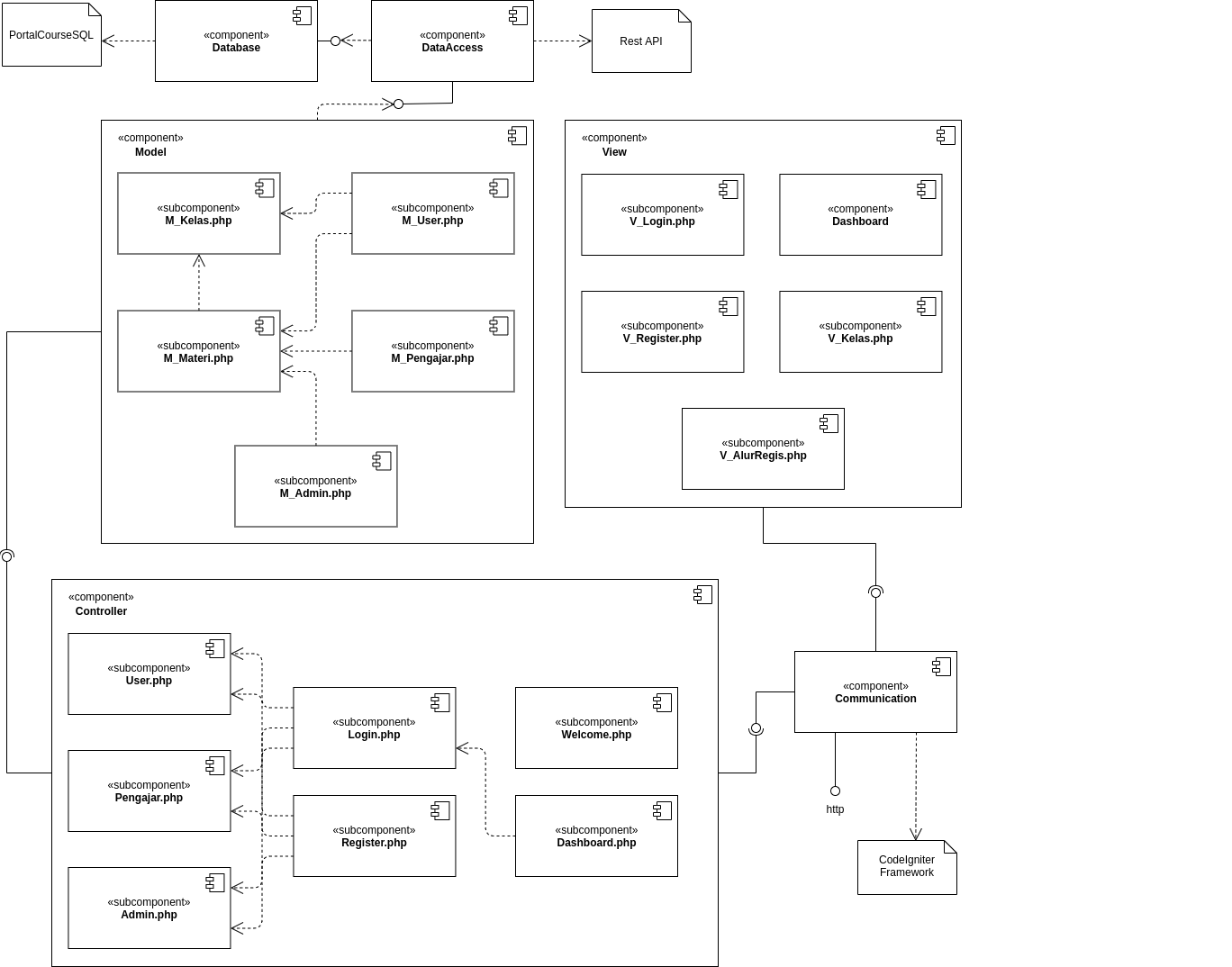
* Bab 1, Menjelaskan mengenai untuk Siapa, Apa dan Mengapa Perancangan atau Sistem/ Program ini dibuat, beserta sedikit penjelasan mengenai penjelasan atau pengertian umum suatu istilah.
* Bab 2, Menjelaskan Rancangan Global Sistem/ Program secara terstruktur mengenai keselarasan *Software* dengan *Hardware* beserta Komponen apa saja yang diperlukan untuk menjalankan Sistem/ Program ini dengan lancar.
* Bab 3, Menjelaskan Perancangan Program lebih terperinci. Penjelasan lebih mendalam dengan digambarkannya Diagram agar mudah dimengerti dan beberapa *Permission* yang dimana menjelaskan tiap sisi dari pengguna dengan masing-masing fungsionalitasnya.

# **Deskripsi Perancangan Global**

## ***Deskripsi Arsitektural***

****

## ***Deskripsi Komponen***



|  |  |  |
| --- | --- | --- |
| **No** | **Nama Komponen** | **Keterangan** |
| 1 | Database | Database Portal Course menggunakan mysqlf. |
| 2 | DataAccess | Untuk menjadikan sistem dengan performa yang baik, terutama dalam hal pertukaran data. |
| 3 | Model | Kumpulan Model yang bertugas dalam mengatur, menyiapkan, dan mengelola data sesuai instruksi Controller. |
| 4 | Controller | Kumpulan Controller yang bertugas dalam mengatur dan merelasikan Model dengan View. |
| 5 | View | Kumpulan View yang bertugas dalam menyajikan Interface dan Informasi sesuai instruksi Controller. |
| 6 | Communication | Komponen Framework yang akan mengelola Controller dan View. |

# **Perancangan Rinci**

## ***Realisasi Use Case***

Berisi TABEL USE CASE sebagai berikut :

|  |  |  |
| --- | --- | --- |
| No | Nama Use Case | Deskripsi Use Case |
| #1 | Membeli Kelas | Pengguna membeli dan memilih kkelas dengan kelas yang sudah disediakan |
| #2 | Melihat Kelas | Pengguna dapat melihat kelas yang sudah dibeli |
| #3 | Diskusi Forum | Pengguna dan Pengajar Kursus dapat berdiskusi tentang materi kursus |
| #4 | Menambahkan Materi | Pengajar Kursus dapat menambahkan materi kursus |
| #5 | Mengubah Materi | Pengajar Kursus, Back Office (Kursus) dapat mengubah materi kursus |
| #6 | Melihat materi | Pengguna, Pengajar Kursus, Back Office (Kursus) dapat melihat materi kursus |
| #7 | Menghapus Materi | Pengajar Kursus, dan Back Office (Kursus) dapat menghapus materi kursus |
| #8 | Merekapitulasi Materi | Back Office (Kursus) dapat merekapitulasi materi kursus |
| #9 | Merekapitulasi Transaksi Pembayaran | Back Office (Pembayaran) dapat merekapitulasi transaksi pembayaran |
| #10 | Merekapitulasi Anggota Terdaftar | Back Office (Anggota) dapat merekapitulasi anggota yang terdaftar |
| #11 | Login | Pengguna, Pengajar Kursus, Back Office (Kursus), Back Office (Pembayaran), Back Office (Anggota) dapat login |
| #12 | Registrasi | Pengguna registrasi untuk memulai kursus |

**Logo Portal Course**



Tampilan Welcome Page



|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *btnkelas* | *Button* | *kelas* | *Jika diklik, akan menuju ke halaman v\_pilihKelas* |
| *btnsc* | *Button* | *showcase* | *Jika diklik akan menuju ke halaman v\_showcase* |
| *btnalur* | *Buttin* | *alur* | *Jika diklik akan menuju ke halaman v\_alurRegis* |
| *btndaftar* | *Button* | *regemail* | *Jika diklik akan menuju ke halaman v\_regisEmail* |
| *btnlogin* | *Buttin* | *login* | *Jika diklik akan menuju ke halaman v\_login* |

### **Use Case #1 <Membeli Kelas>**

*Skenario Use Case #1 :*

*Primary Flow :*

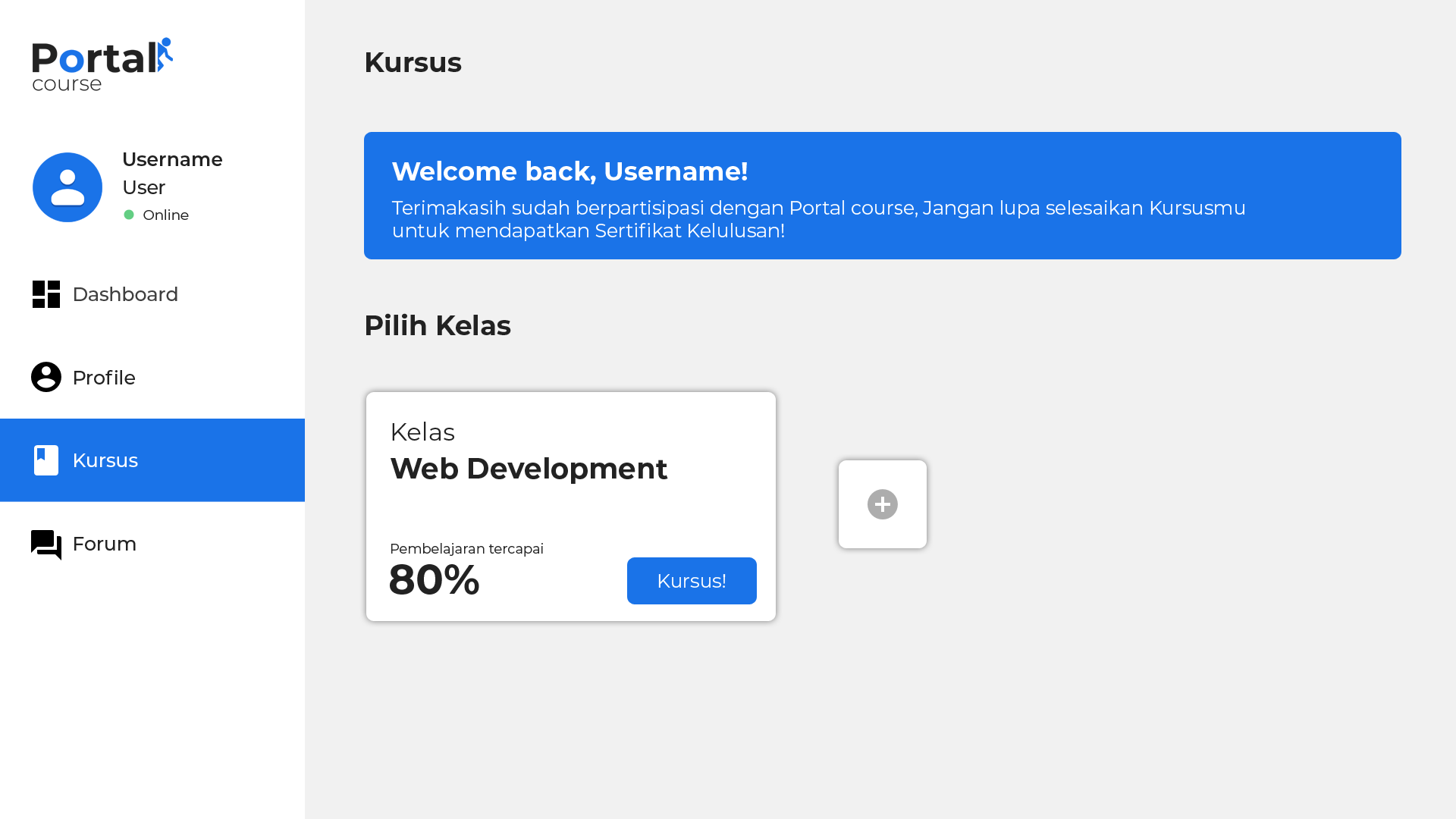
1. Pengguna klik menu kelas
2. Pengguna memilih kelas
3. Pengguna memasukkan email terkait tata cara pembayaran
4. Pengguna memasukkan token

*Alternate Flow :*

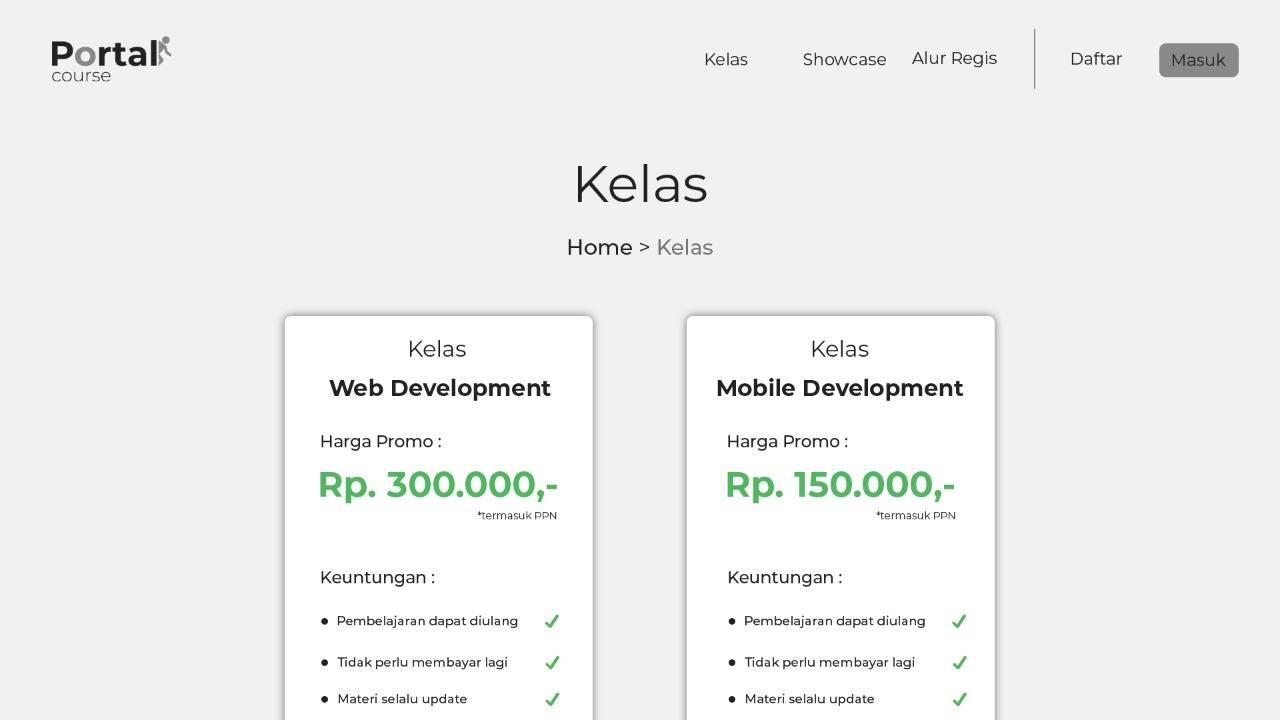
1. Jika pengguna belum mempunyai email maka akan dialihkan pada Halaman Registrasi
2. Jika token gagal, maka lapor kepada pihak terkait
3. Jika email yang digunakan untuk mendaftar sudah ada, maka gunakan email lainnya

#### **Perancangan Antarmuka Usecase #1**

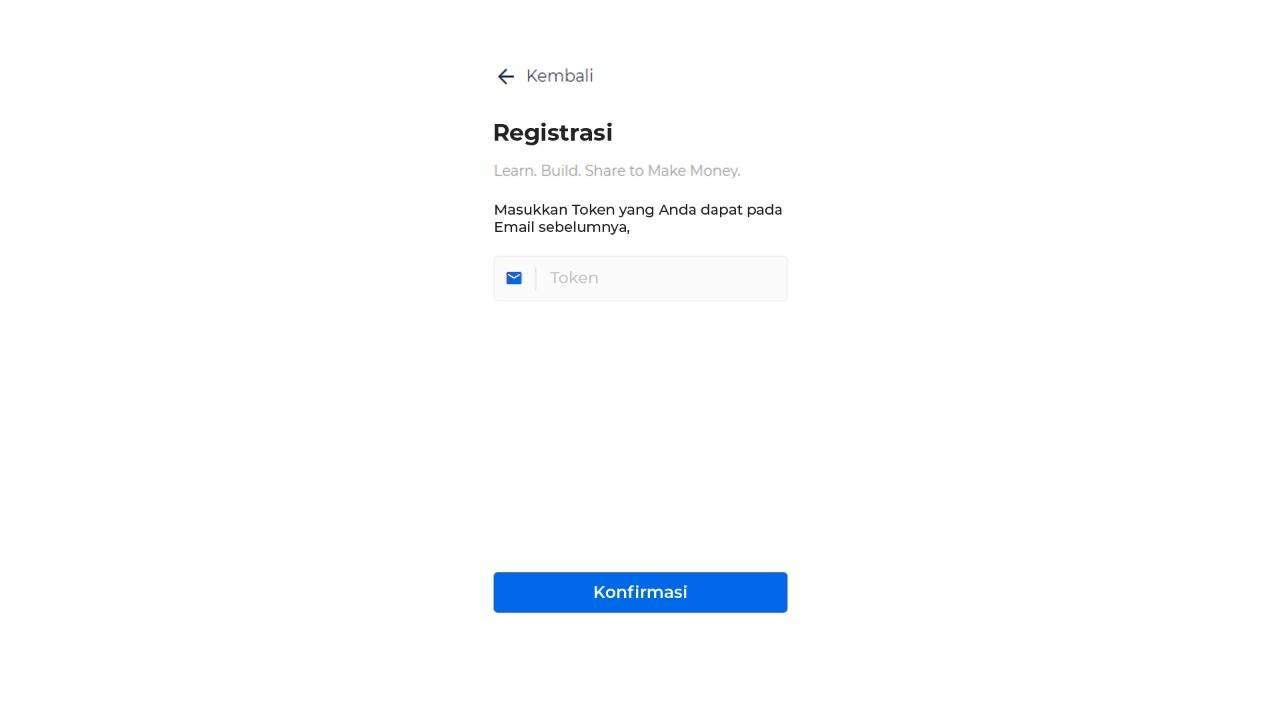
Halaman v\_kursus



Halaman v\_kelas



Halaman v\_konfirmToken



#### **Tabel Identifikasi Antarmuka / Layar / Page**

|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
| *1* | *v\_kursus* | Halaman utama |
| *2* | *v\_pilihKelas* | Halaman kelas menampilkan pilihan kelas |
| *3* | *v\_konfirmToken* | Halaman untuk mengisi token yang diberikan oleh sistem |

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detil

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

*Page v\_pilihKelas*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *btnKelas1* | *Button* | *kelas1* | *Jika diklik, akan mengaktifkan Function beli\_kursus()* |
| *btnKelas2* | *Button* | *kelas2* | *Jika diklik akan mengaktifkan Function beli\_kursus()* |

*Page v\_konfirmToken*

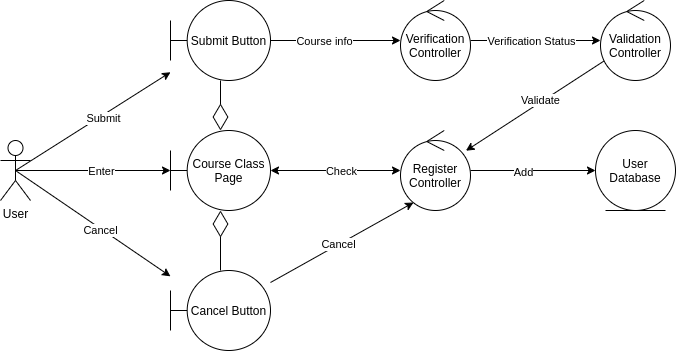
|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *btn1* | *Button* | *buatKonfirmasi* | *Jika diklik, akan mengaktifkan Function Konfirmasi Token* |
| *tx1* | *Text Input* | *token* | *Mengisi kotak teks dengan token yang sudah terkirim oleh sistem di email* |

#### **Identifikasi Object Baru**

*TABEL OBJECT PERANCANGAN*

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
| 1 | *Course Class Page* | *Boundary* |
| 2 | *User* | *Entity* |
| 3 | *Register* | *Controller* |
| 4 | *Verification* | *Controller* |
| 5 | *Validation* | *Controller* |

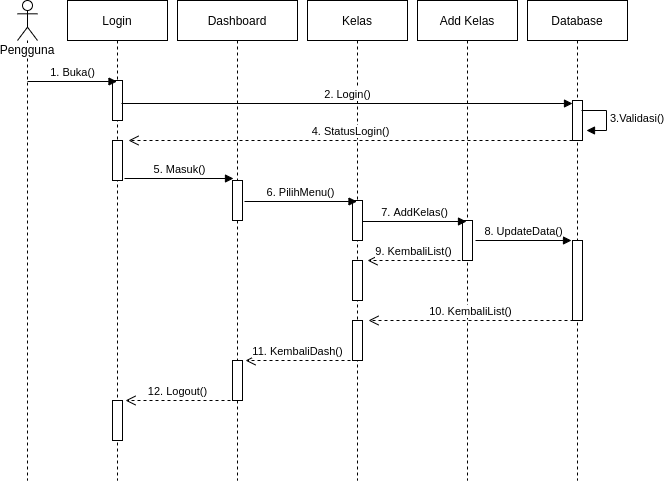
#### **Robustness Diagram**



#### **Diagram Kelas**



#### **Sequence Diagram**



### **Use Case #2 <Melihat Kelas>**

*Skenario Use Case #1*

*Primary Flow:*

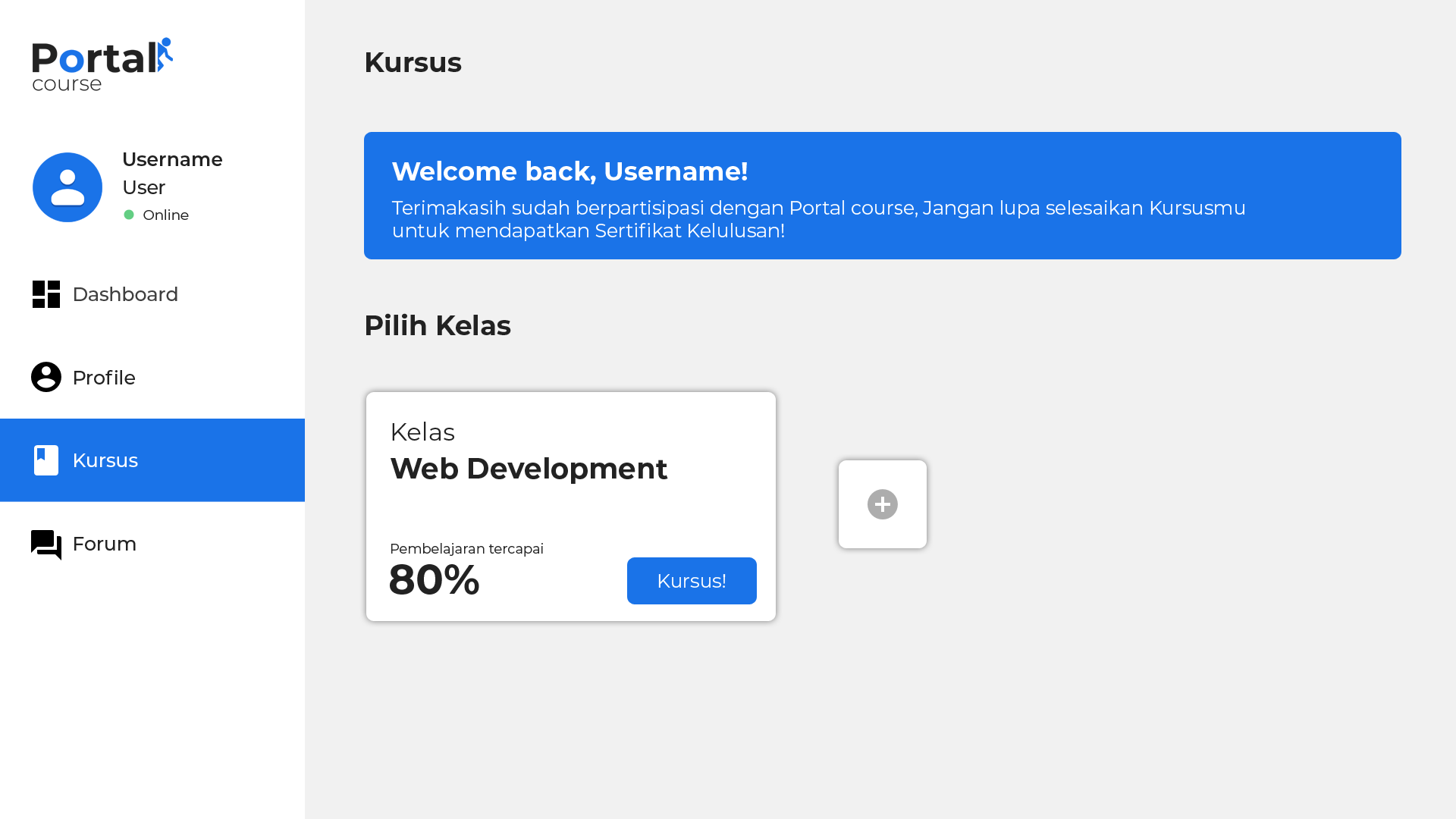
1. Pengguna memilih kelas
2. Pengguna melakukan kursus

*Alternate Flow:*

1. Jika ingin melihat kursus lainnya maka disarankan untuk membeli terlebih dahulu

#### **Perancangan Antarmuka Usecase #2**

Halaman v\_kursus



#### **Tabel Identifikasi Antarmuka / Layar / Page**

|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
| 1 | Halaman v\_kursus | Antarmuka pengguna untuk melihat Kelas yang telah dibeli |

*Page v\_kursus*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| btnKursus | Button | tinjau | Jika diklik, akan menuju ke halaman *mengaktifkan Function lihat\_kursus()* yakni kumpulan materi |

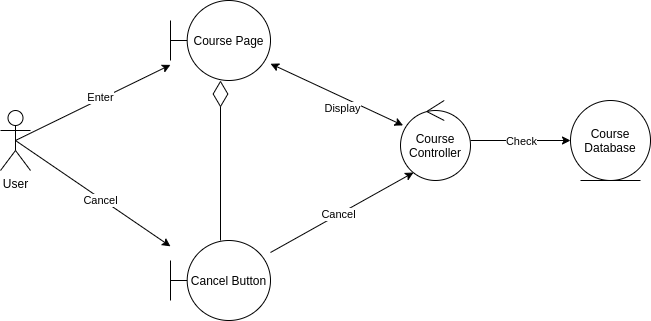
#### **Identifikasi Object Baru**

TABEL OBJECT PERANCANGAN

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
| *1* | *Course Page* | *Boundary* |
| *2* | *Course* | *Entity* |
| *3* | *Course* | *Controller* |

#### **Robustness Diagram**

* User

****

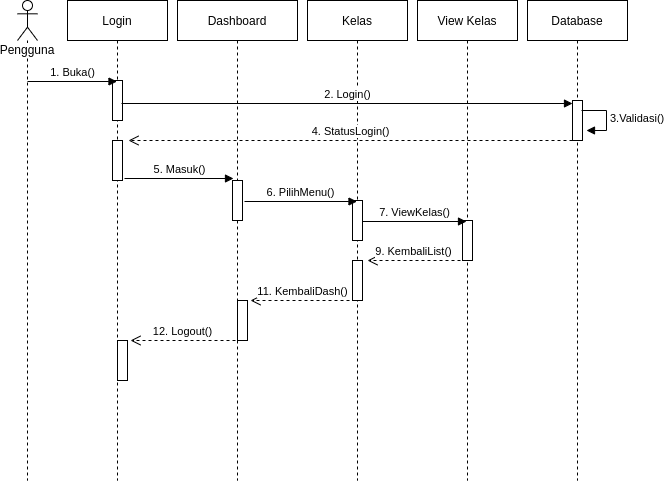
#### **Diagram Kelas**

#### 

#### **Sequence Diagram**

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

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



### **Use Case #3 <Diskusi Forum>**

*Skenario Use Case #3*

*Primary Flow :*

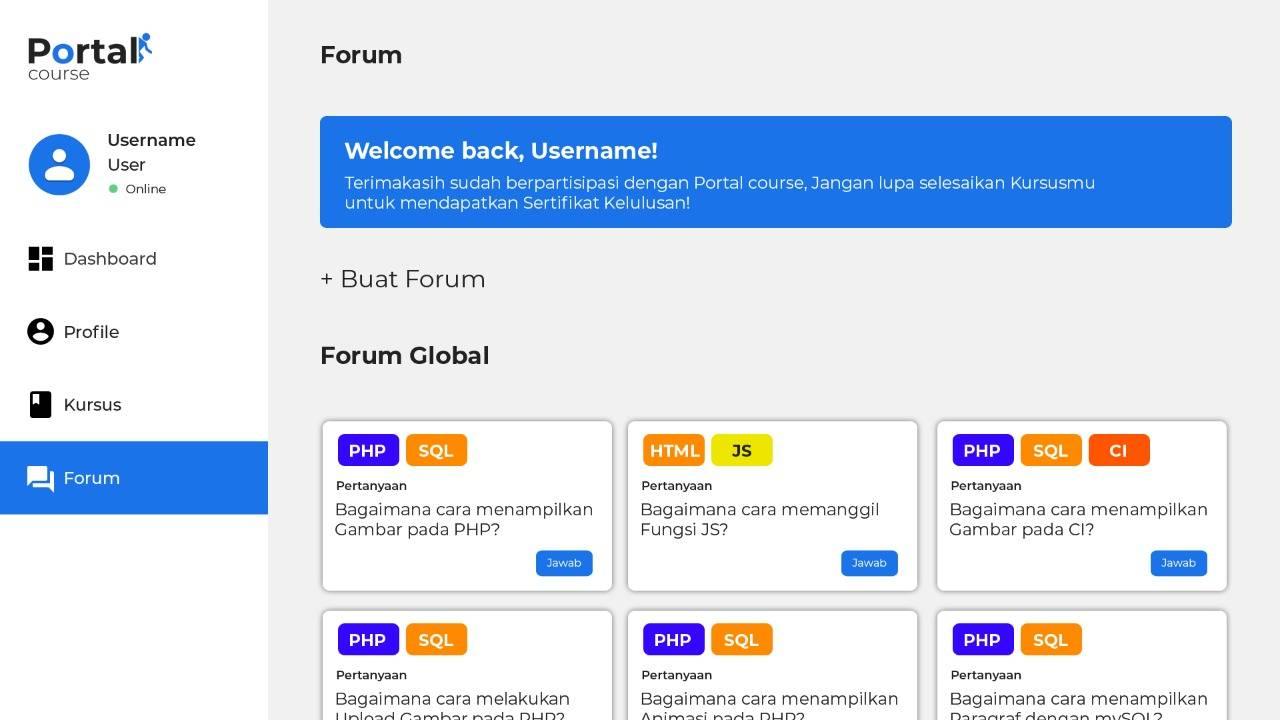
1. Pengguna dapat memilih membuat atau menjawab forum
2. Jika ingin membuat Forum, maka klik button Buat Forum
3. Pengguna atau Pengajar Kursus menjawab forum
4. Klik button Jawab

*Alternate Flow:*

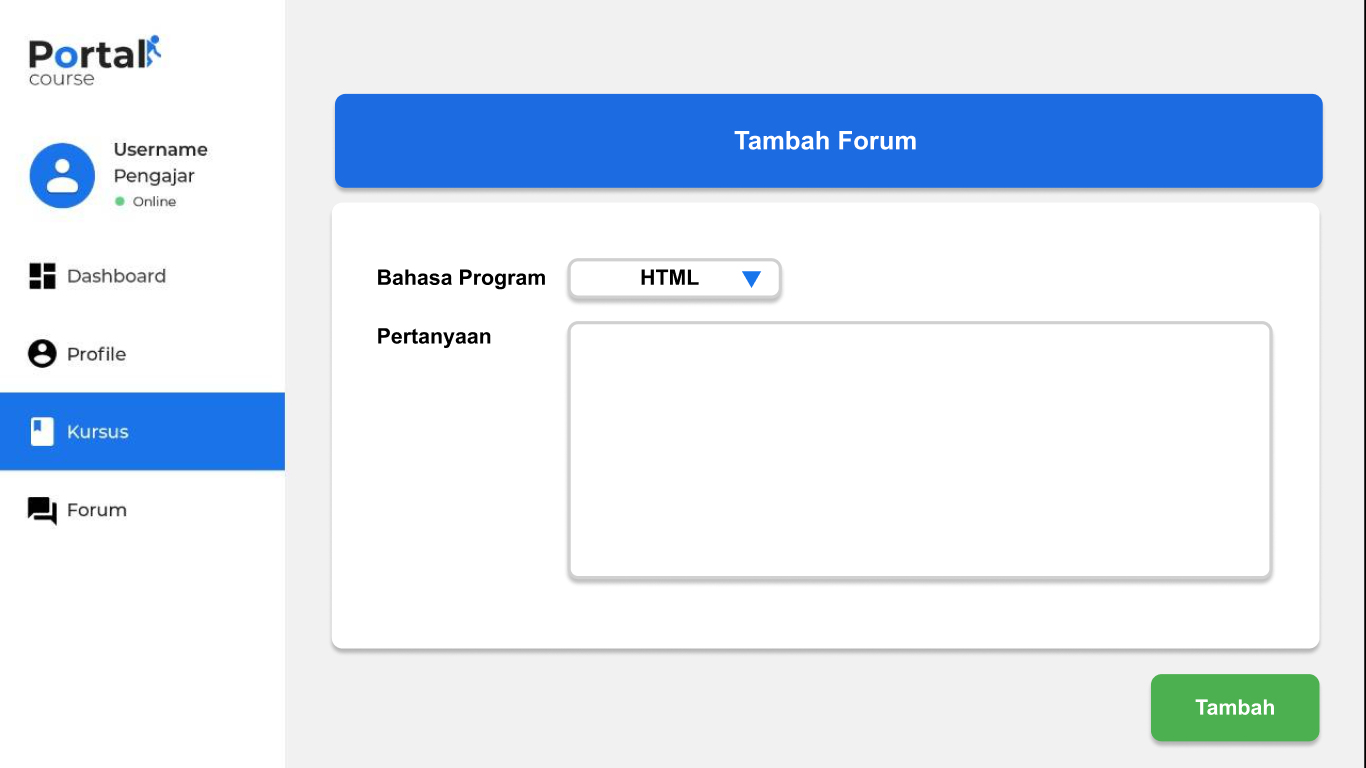
1. Jika belum ada forum maka disarankan tambahkan forum terlebih dahulu

#### **Perancangan Antarmuka Usecase #3**

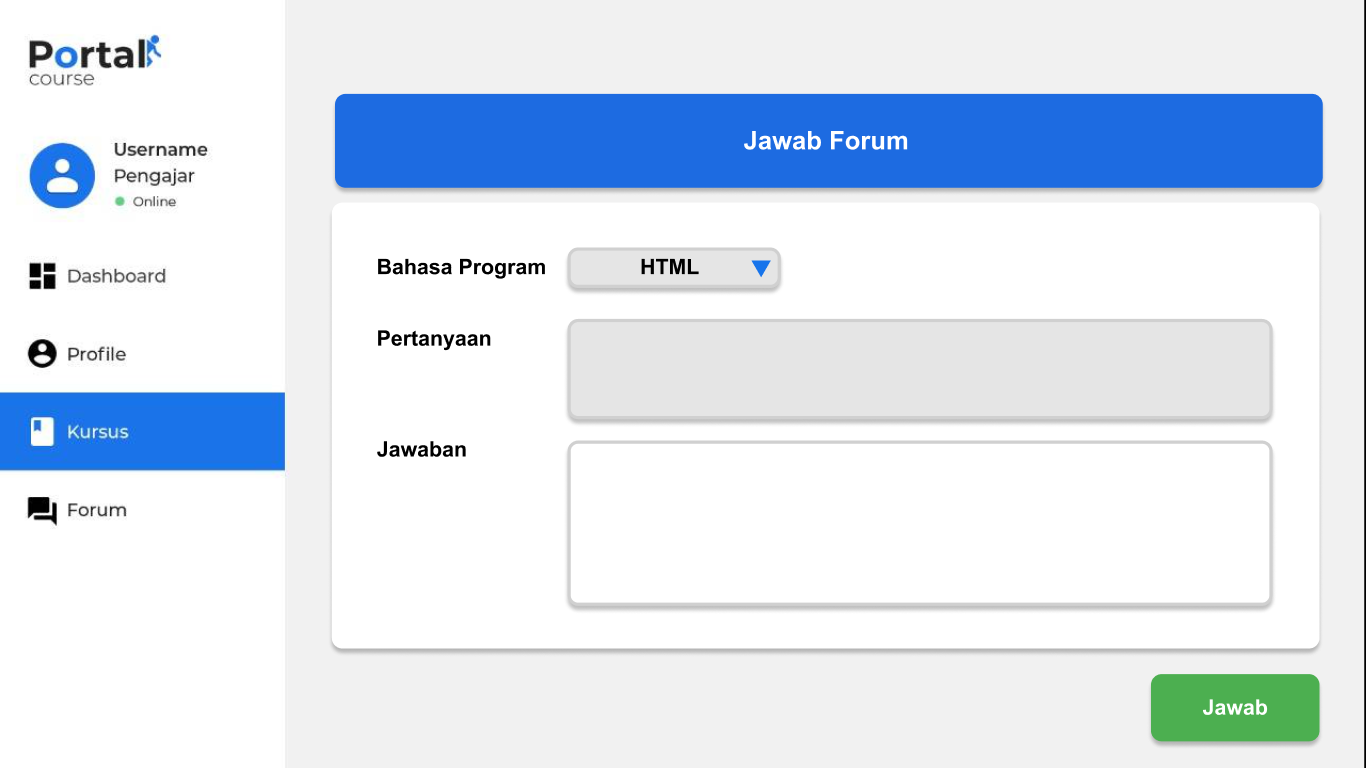
v\_forum



v\_buatForum



v\_jawabForum



#### **Tabel Identifikasi Antarmuka / Layar / Page**

|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
| *4* | Halaman v\_forum | Halaman untuk berdiskusi dengan menambah forum dan menjawab forum antar pengguna |
| *5* | Halaman v\_buatForum | Halaman bagi pengguna untuk membuat forum baru |
| *6* | Halaman v\_jawabForum | Halaman bagi pengguna dan pengajar untuk menjawab forum |

*Halaman v\_forum*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *btnBuatForum* | *Button* | *buatForum* | *Jika diklik, akan menuju ke halaman buat forum* |
| *btnJawab* | *Button* | *jawab* | *Jika diklik akan menuju ke halaman jawab* |

*Halaman v\_buatForum*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *btn1* | *Button* | *bahasa* | *Memilih bahasa : HTML, PHP, Java, SQL, JS, CI* |
| *tb1* | *Textbox* | *pertanyaan* | *Mengisi textbox dengan pertanyaan* |
| *btnTambah* | *Button* | *tambah* | *Jika diklik akan mengaktifkan fungsi buat\_forum()* |

*Halaman v\_jawabForum*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *btn1* | *Button* | *bahasa* | *Menampilkan bahasa* |
| *tb1* | *Textbox* | *pertanyaan* | *Menampilkan pertanyaan* |
| *tb2* | *Textbox* | *jawaban* | *Mengisi text boc dengan jawaban* |
| *btnJawab* | *Button* | *jawab* | *Jika diklik akan mengaktifkan fungsi jawab\_forum()* |

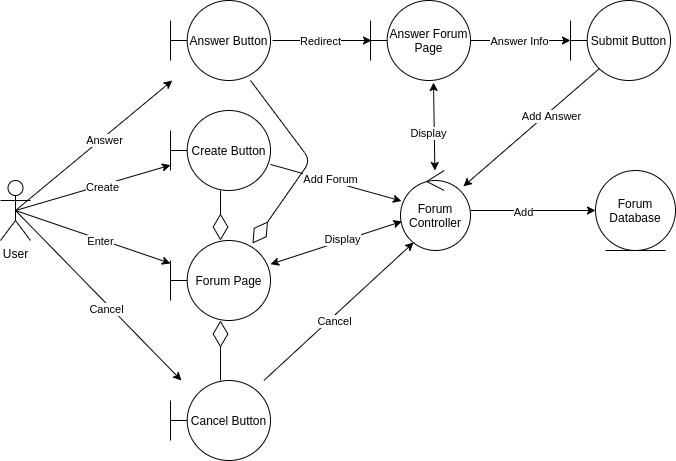
#### **Identifikasi Object Baru**

TABEL OBJECT PERANCANGAN

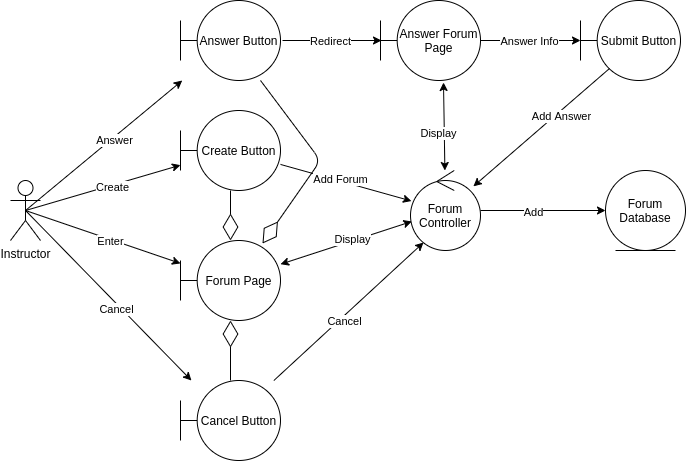
|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
| *1* | *Forum Page* | *Boundary* |
| *2* | *Answer Forum Page* | *Boundary* |
| *3* | *Forum* | *Entity* |
| *4* | *Forum* | *Controller* |

#### **Robustness Diagram**

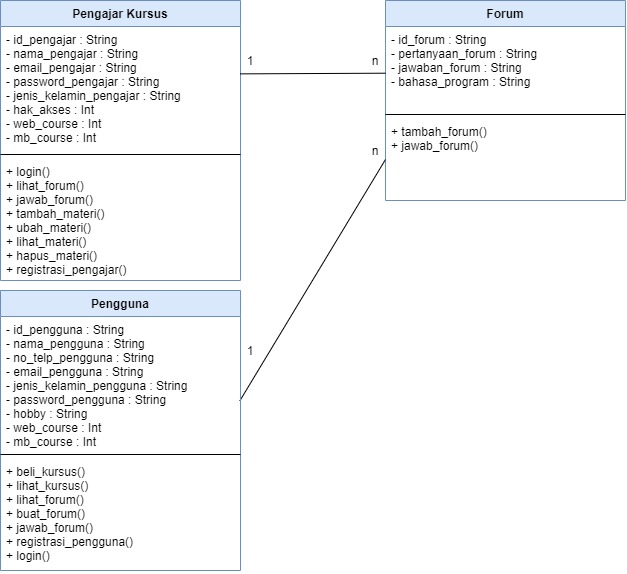
* Pengguna



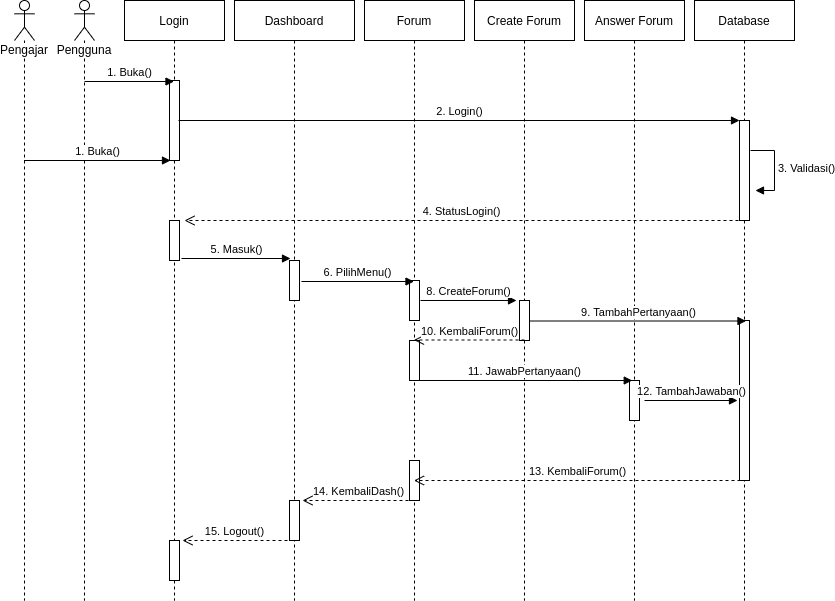
* Pengajar



#### **Diagram Kelas**



#### **Sequence Diagram**

**

### **Use Case #4 <Menambahkan Materi>**

*Skenario Use Case #3*

*Primary Flow:*

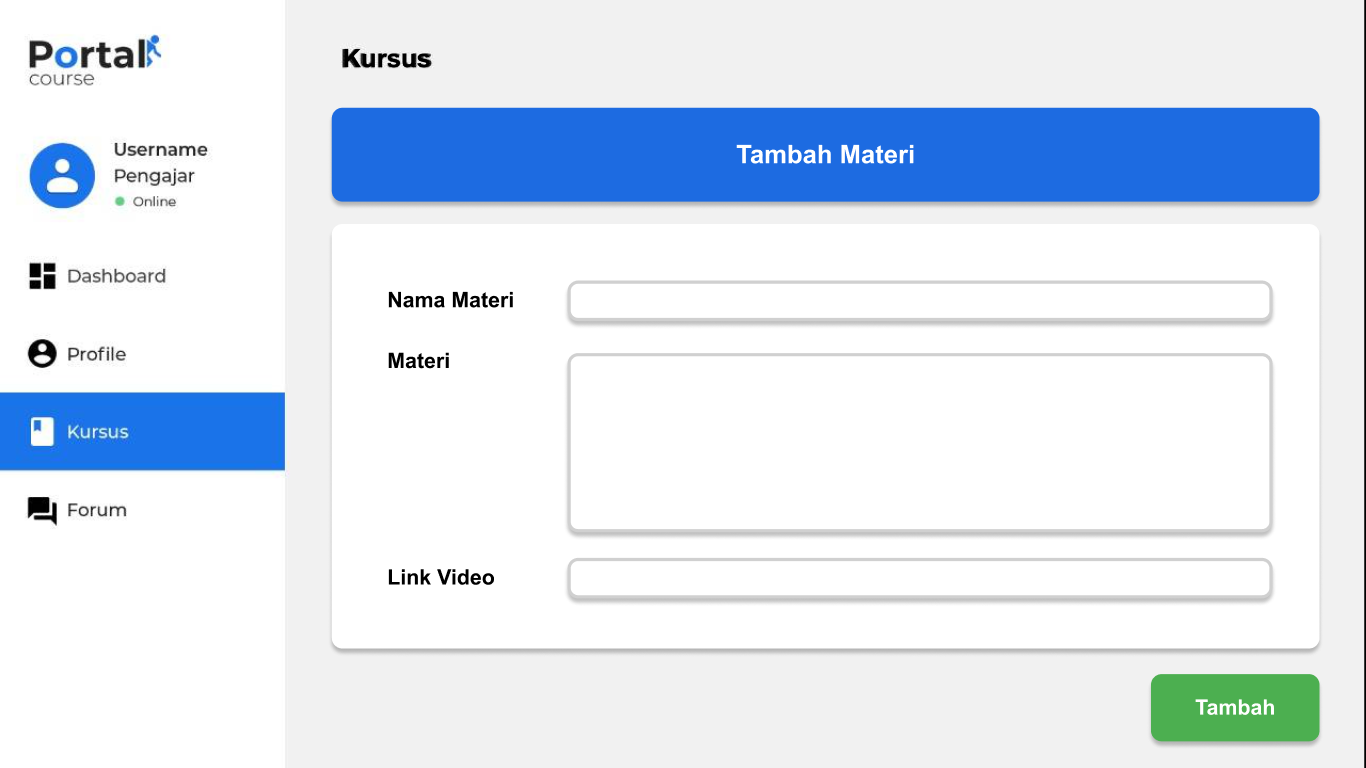
1. Klik button tambah materi
2. Isi materi yang ingin ditambahkan
3. Klik button simpan

*Alternate Flow:*

1. Jika penambahan materi gagal akan dialihkan ke *Error page*

#### **Perancangan Antarmuka Usecase #4**

*v\_tambahMateri*



#### **Tabel Identifikasi Antarmuka / Layar / Page**

|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
| *7* | *Halaman v\_tambahMateri* | *Halaman untuk menambah materi* |

*Page v\_tambahMateri*

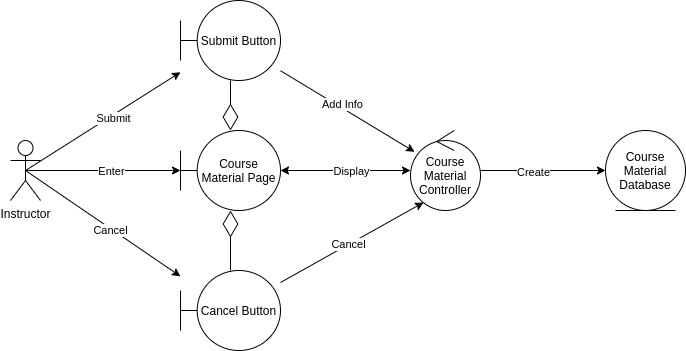
|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *tb1* | *Textbox* | *namaMateri* | *Mengisi text box dengan nama materi* |
| *tb2* | *Textbox* | *materi* | *Mengisi text box dengan materi* |
| *tb3* | *Textbox* | *video* | *Mengisi text box dengan link video* |
| *btnSimpan* | *Button* | *simpanMateri* | *Jika diklik akan mengaktifkan Function tambah\_materi* |

#### **Identifikasi Object Baru**

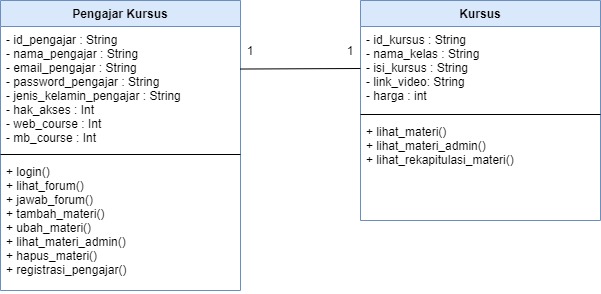
TABEL OBJECT PERANCANGAN

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
| *1* | *Course Material Page* | *Boundary* |
| *2* | *Course Material* | *Entity* |
| *3* | *Course Material* | *Controller* |

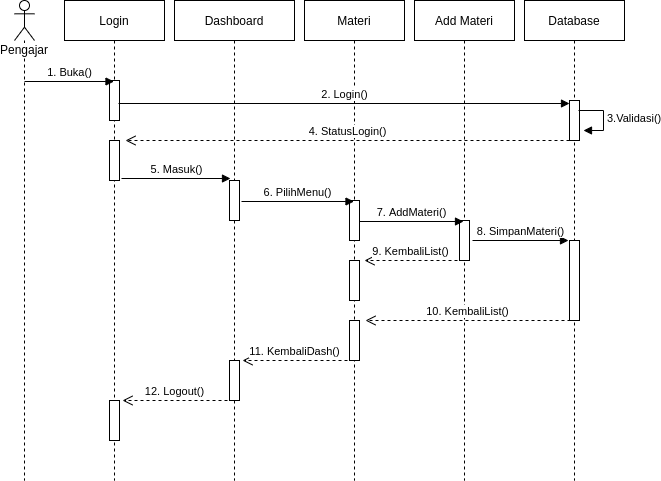
#### **Robustness Diagram**

****

#### **Diagram Kelas**



#### **Sequence Diagram**

**

### **Use Case #5 <Mengubah Materi>**

*Skenario Use Case #3*

*Primary Flow :*

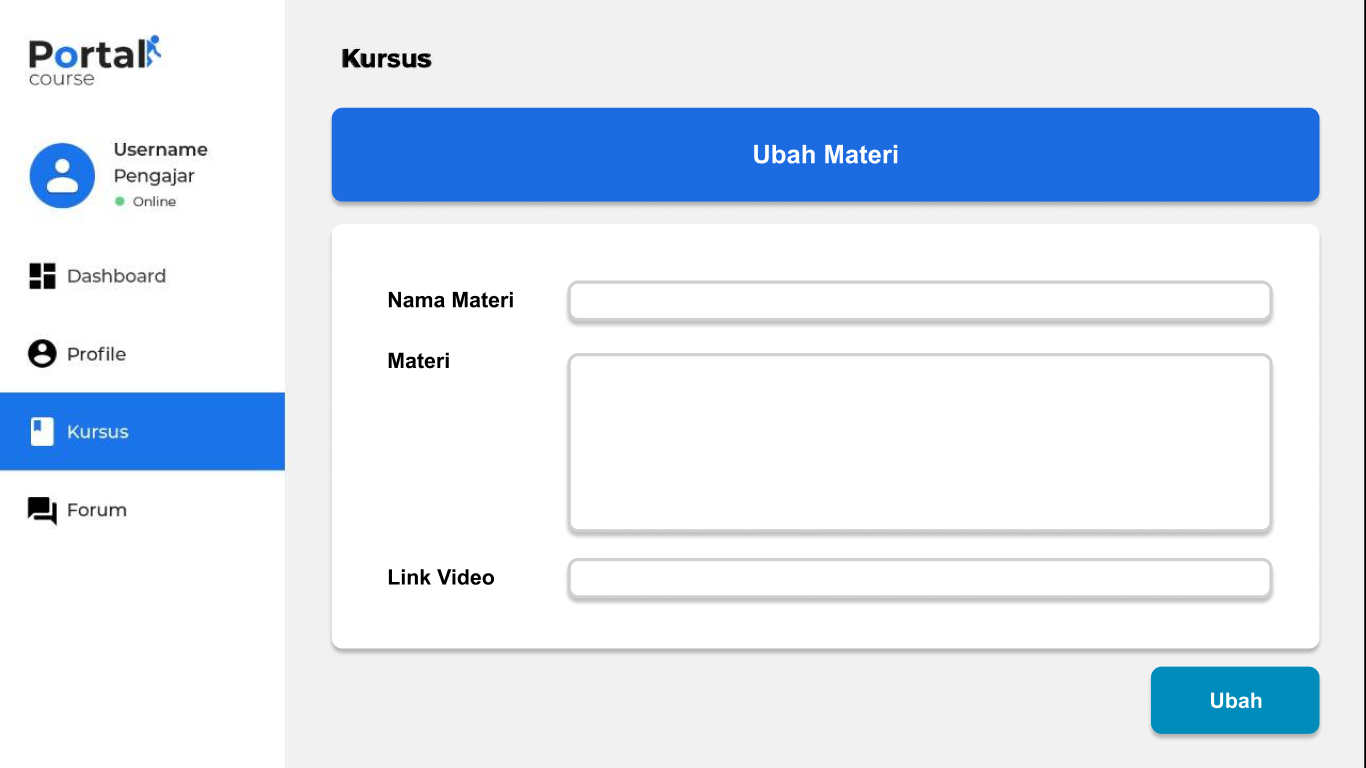
1. Pilih materi yang diubah
2. Klik button ubah
3. Isi text box dengan materi
4. Klik button ubah

*Alternate Flow:*

1. Jika pengubahan materi gagal akan dialihkan ke *Error page*

#### **Perancangan Antarmuka Usecase #5**

v\_ubahMateri



#### **Tabel Identifikasi Antarmuka / Layar / Page**

|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
| *8* | *v\_ubahMateri* | *Halaman untuk mengubah materi* |

*Page v\_ubahMateri*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *tb1* | *Textbox* | *nama* | *Mengisi text box dengan nama materi* |
| *tb2* | *Textbox* | *materi* | *Mengisi text box dengan materi* |
| *tx1* | *Text Input* | *video* | *Mengisi text input dengan link video* |
| *btnUbah* | *Button* | *ubahMateri* | *Jika diklik akan mengaktifkan Function ubah\_materi* |

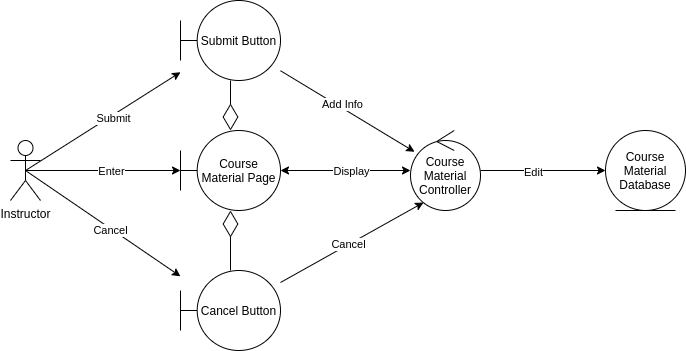
#### **Identifikasi Object Baru**

TABEL OBJECT PERANCANGAN

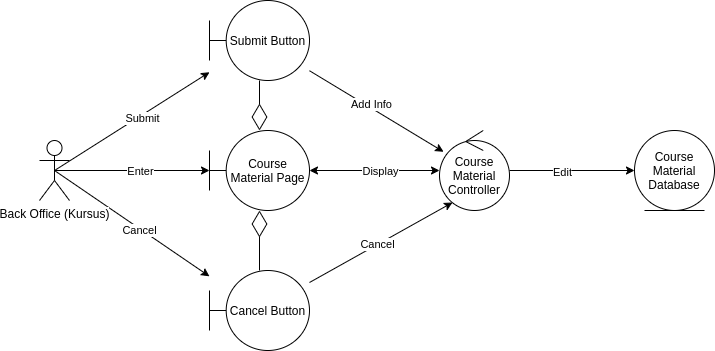
|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
| *1* | *Course Material Page* | *Boundary* |
| *2* | *Course Material* | *Entity* |
| *3* | *Course Material* | *Controller* |

#### **Robustness Diagram**

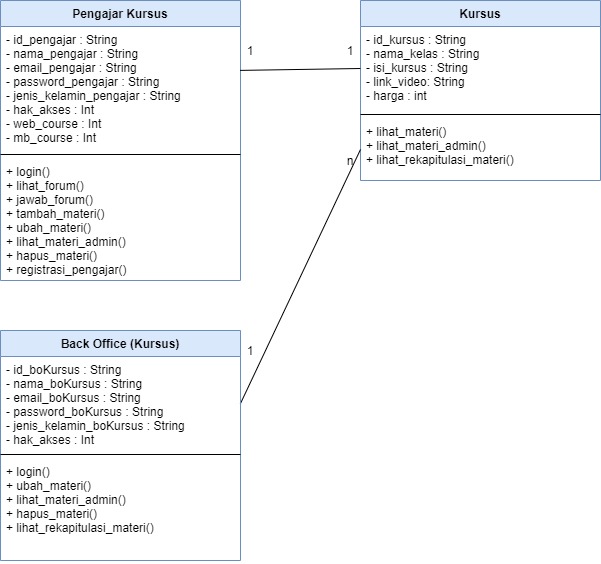
* Pengajar Kursus

****

* Back Office (Kursus)

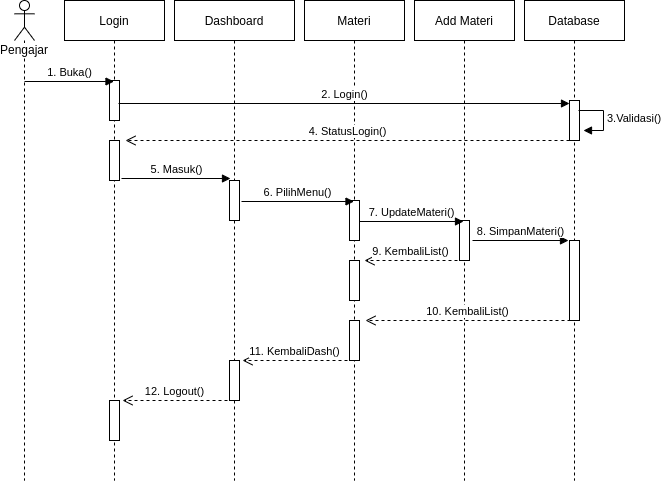
****

#### **Diagram Kelas**

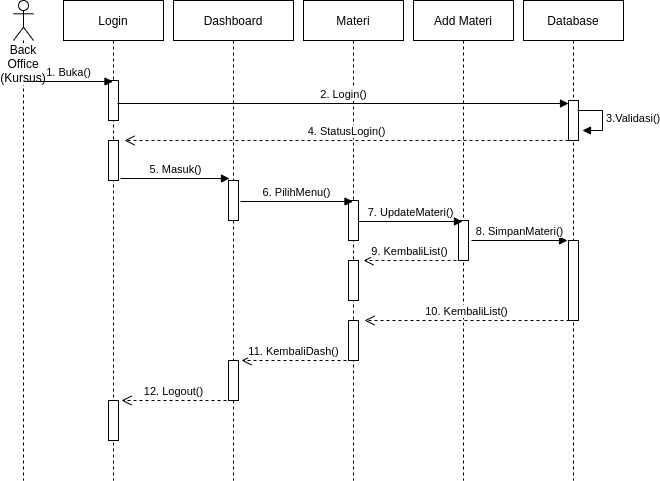


#### **Sequence Diagram**

* *Pengajar*

**

* *Back Office (Kursus)*

**

### **Use Case #6 <Melihat Materi>**

*Skenario Use Case #3*

*Primary Flow:*

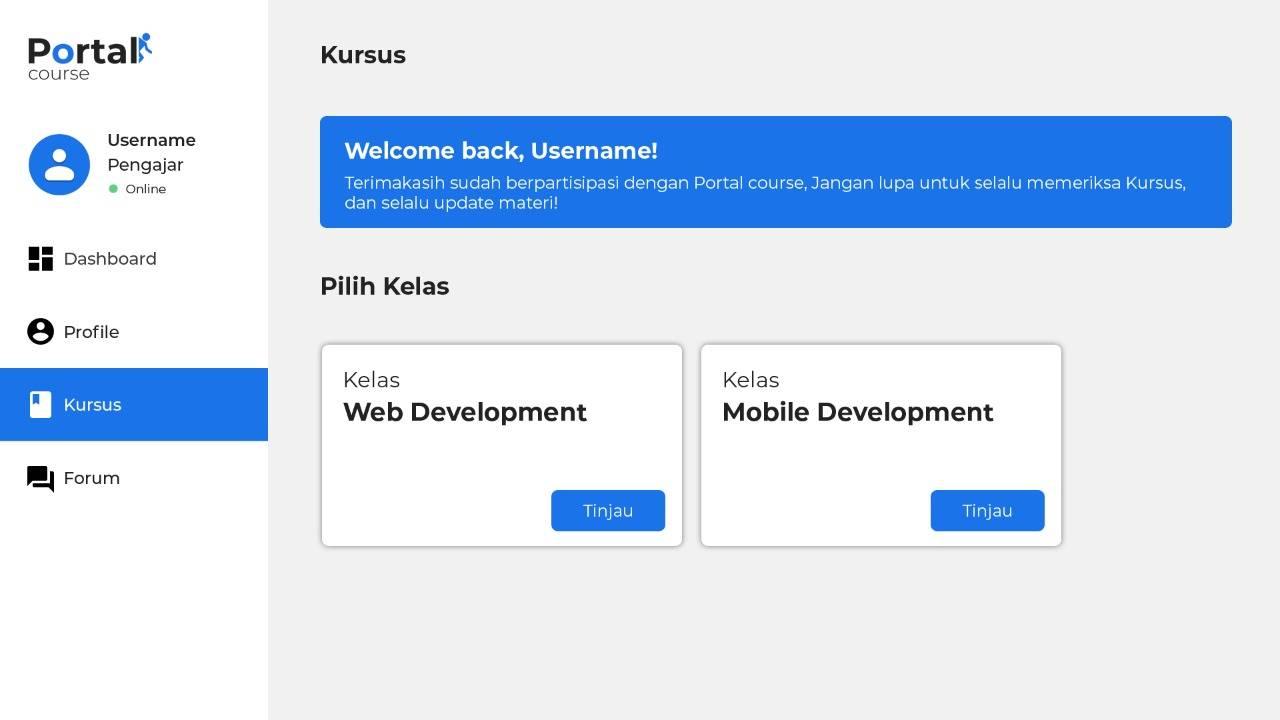
1. Klik button tinjau

*Alternate Flow*

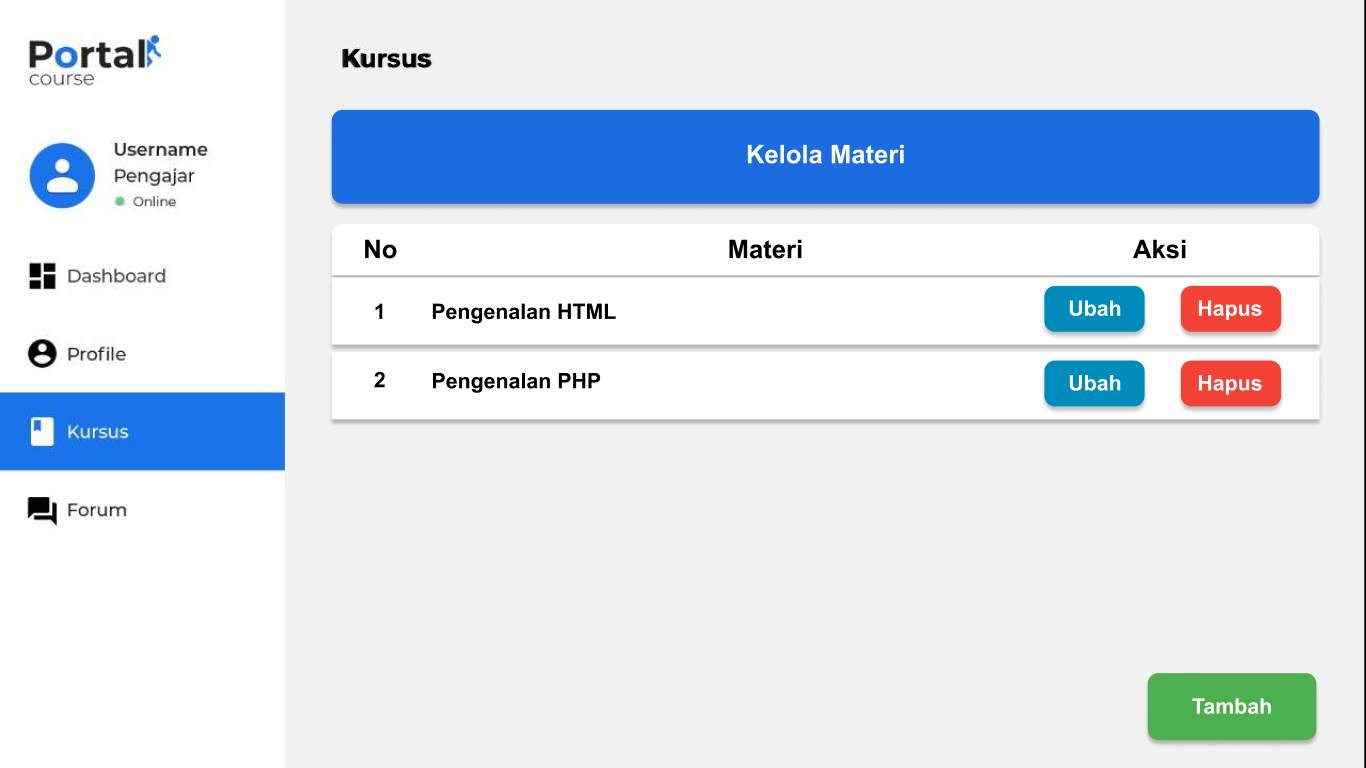
1. *Jika materi belum ada maka harus menambah materi terlebih dahulu*

#### **Perancangan Antarmuka Usecase 6**

v\_materiAdmin



v\_materiPengajar



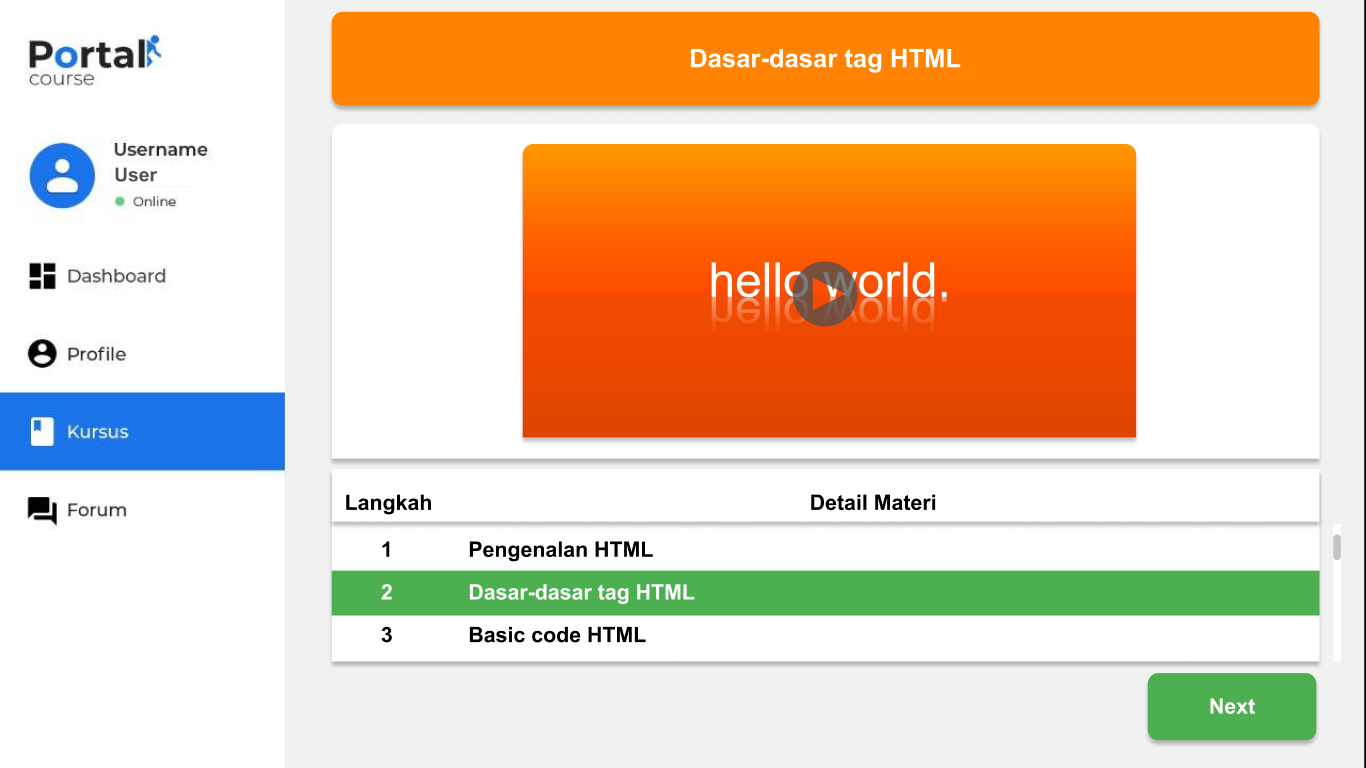
v\_materi



v\_materiDetail



v\_materiDetail



#### **Tabel Identifikasi Antarmuka / Layar / Page**

|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
| *9* | *v\_kursusPengajar* | *Halaman bagi pengajar kursus untuk meninjau materi* |
| *10* | *v\_materiPengajar* | *Halaman bagi pengajar kursus dan back office (kursus) untuk mengelola materi yaitu menambahkan, mengubah, menghapus, dan mengubah materi* |
| *16* | *v\_materi* | *Halaman bagi pengguna untuk melihat materi* |

*Page v\_kursusPengajar*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *btnTinjau* | *Button* | *tinjau* | *Jika diklik, akan menuju ke halaman v\_materiPengajar* |

*Page v\_materiPengajar*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *btnTambah* | *Button* | *tambah* | *Jika diklik, akan mengaktifkan Function tambah\_materi* |
| *btnUbah* | *Button* | *ubah* | *Jika diklik, akan mengaktifkan Function ubah\_materi* |
| *tbnHapus* | *Button* | *hapus* | *Jika diklik, akan mengaktifkan Function hapus\_materi* |

*Page v\_materi*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *btnSubmit* | *Button* | *submit* | *Jika diklik, akan menuju ke halaman materi yang dipilih* |

*Page v\_materiDetail*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *btnNext* | *Button* | *next* | *Jika diklik, akan menuju ke halaman langkah materi selanjutnya* |

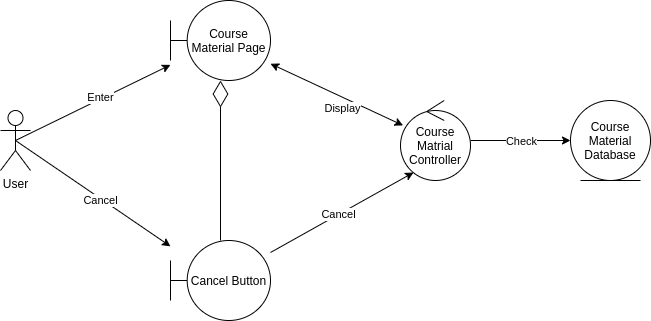
#### **Identifikasi Object Baru**

TABEL OBJECT PERANCANGAN

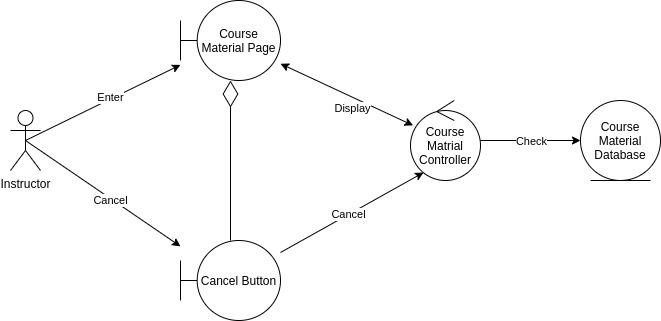
|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
| *1* | *Course Material Page* | *Boundary* |
| *2* | *Course Material* | *Entity* |
| *3* | *Course Material* | *Controller* |

#### **Robustness Diagram**

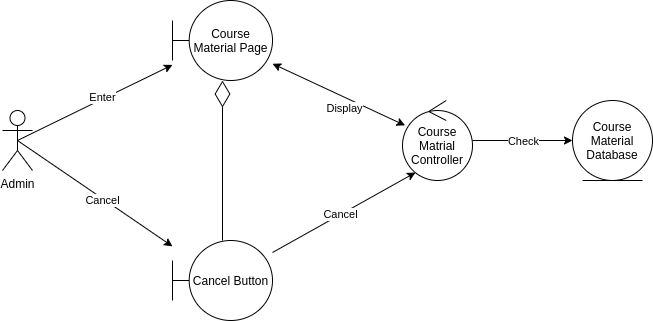
* User

****

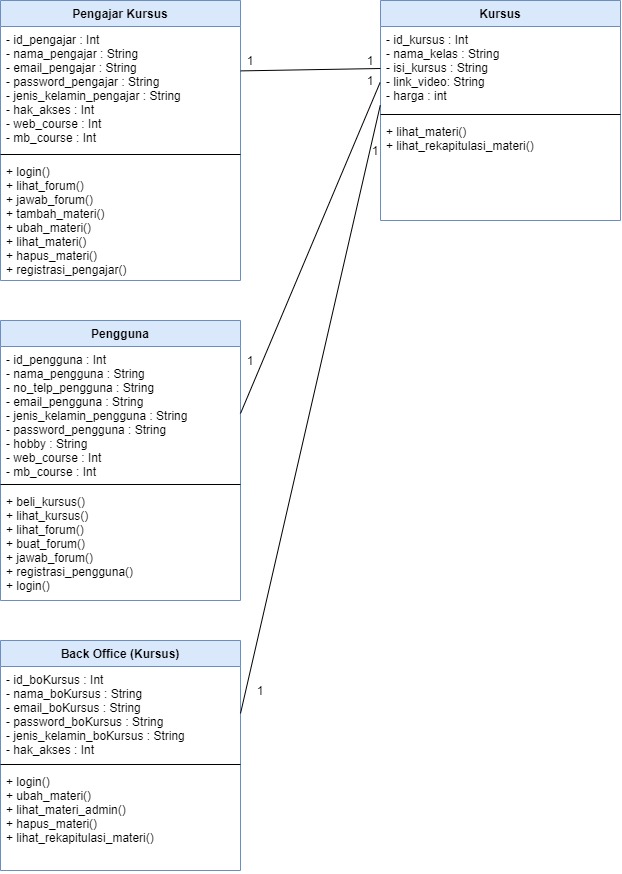
* Pengajar

****

* Back Office (Kursus)

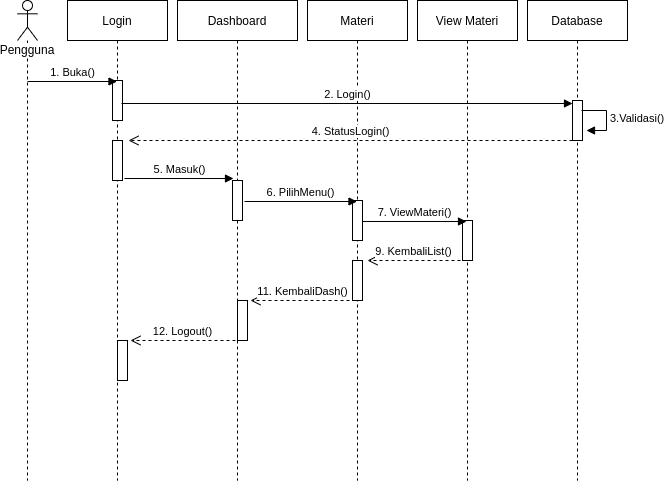
****

#### **Diagram Kelas**

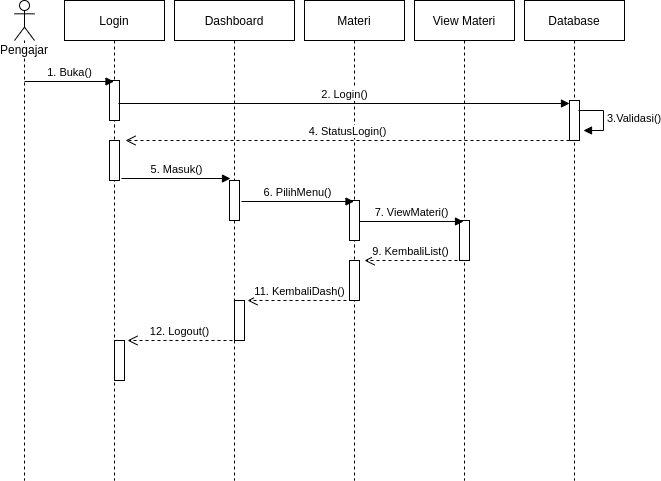


#### **Sequence Diagram**

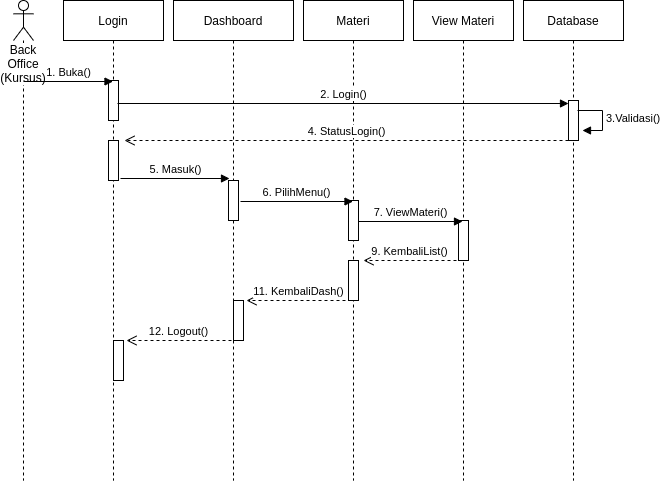
* *Pengguna*

**

* *Pengajar*

**

* *Back Office (Kursus)*

**

### **Use Case #7 <Menghapus Materi>**

*Skenario Use Case #3*

*Primary Flow:*

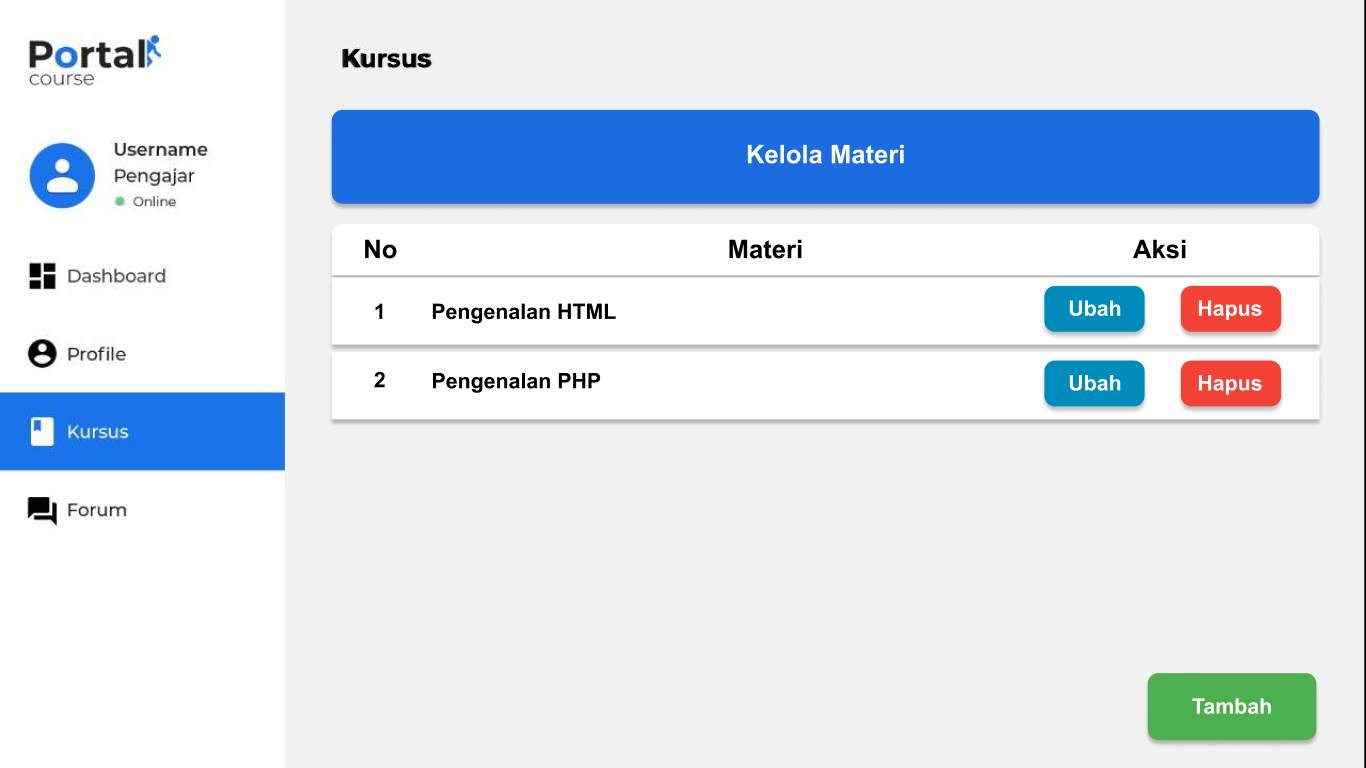
1. Klik button hapus pada data materi yang dipilih

*Alternate Flow:*

1. Jika penghapusan materi gagal akan dialihkan ke *Error page*

#### **Perancangan Antarmuka Usecase 7**

v\_materiPengajar



#### **Tabel Identifikasi Antarmuka / Layar / Page**

|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
| *10* | *v\_materiPengajar* | *Halaman bagi pengajar kursus dan back office (kursus) untuk mengelola materi yaitu menambahkan, mengubah, menghapus, dan mengubah materi* |

*Page v\_materiPengajar*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *btnTambah* | *Button* | *tambah* | *Jika diklik, akan mengaktifkan Function tambah\_materi* |
| *btnUbah* | *Button* | *ubah* | *Jika diklik, akan mengaktifkan Function ubah\_materi* |
| *tbnHapus* | *Button* | *hapus* | *Jika diklik, akan mengaktifkan Function hapus\_materi* |

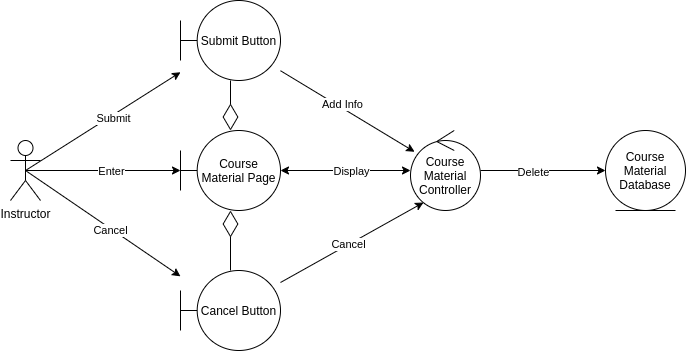
#### **Identifikasi Object Baru**

TABEL OBJECT PERANCANGAN

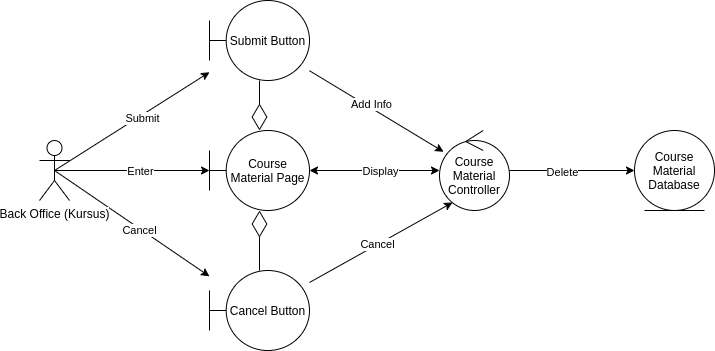
|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
| *1* | *Course Material Page* | *Boundary* |
| *2* | *Course Material* | *Entity* |
| *3* | *Course Material* | *Controller* |

#### **Robustness Diagram**

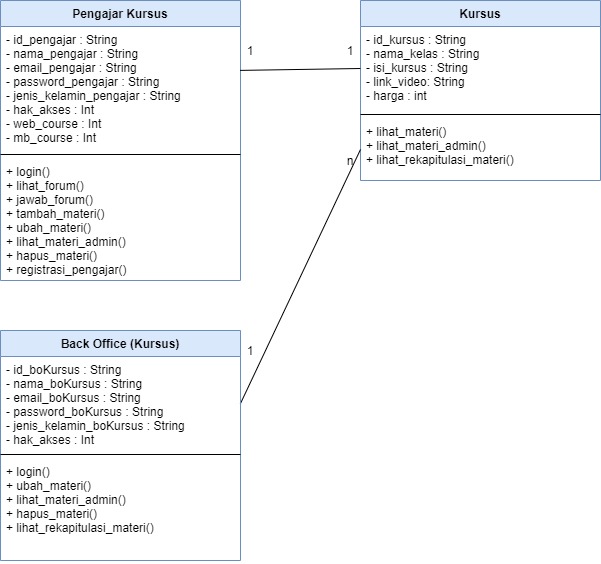
* Pengajar Kursus

****

* Back Office (Kursus)

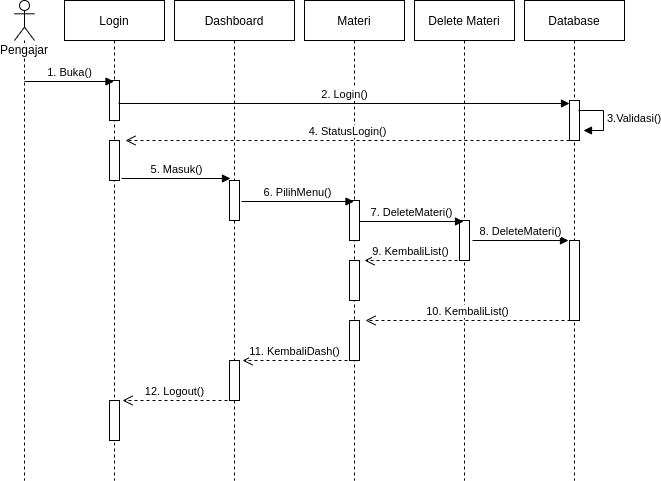
****

#### **Diagram Kelas**

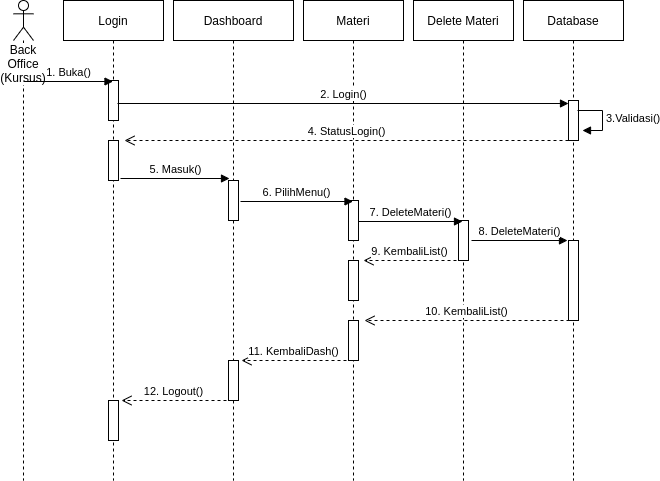


#### **Sequence Diagram**

* *Pengajar*

**

* *Back Office (Kursus)*

**

### **Use Case #8 <Merekapitulasi Materi>**

*Skenario Use Case #3*

*Primary Flow:*

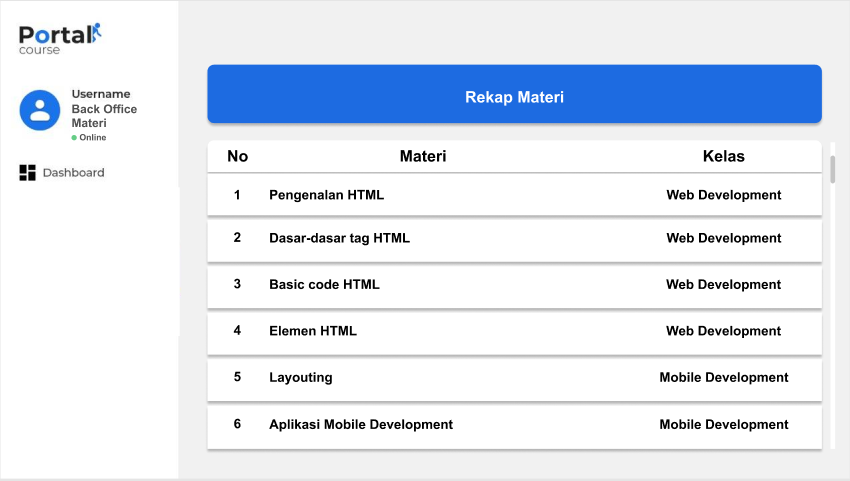
1. Lihat rekapitulasi materi

*Alternate Flow*

1. Jika rekapitulasi materi gagal akan dialihkan ke *Error page*

#### **Perancangan Antarmuka Usecase #8**

v\_rekapMateri



#### **Tabel Identifikasi Antarmuka / Layar / Page**

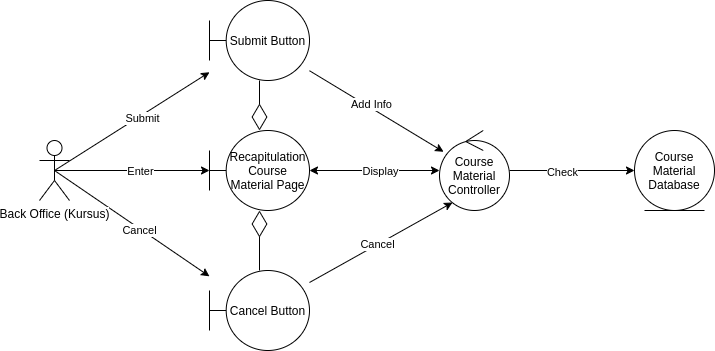
|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
| *11* | *v\_rekapMateri* | *Halaman bagi Back Office (Materi) untuk melihat rekapitulasi materi* |

#### **Identifikasi Object Baru**

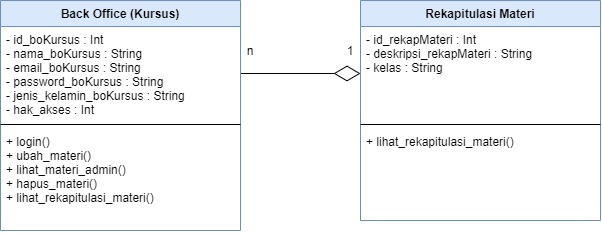
TABEL OBJECT PERANCANGAN

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
| *1* | *Recapitulation Course Material Page* | *Boundary* |
| *2* | *Course Material* | *Entity* |
| *3* | *Course Material* | *Controller* |

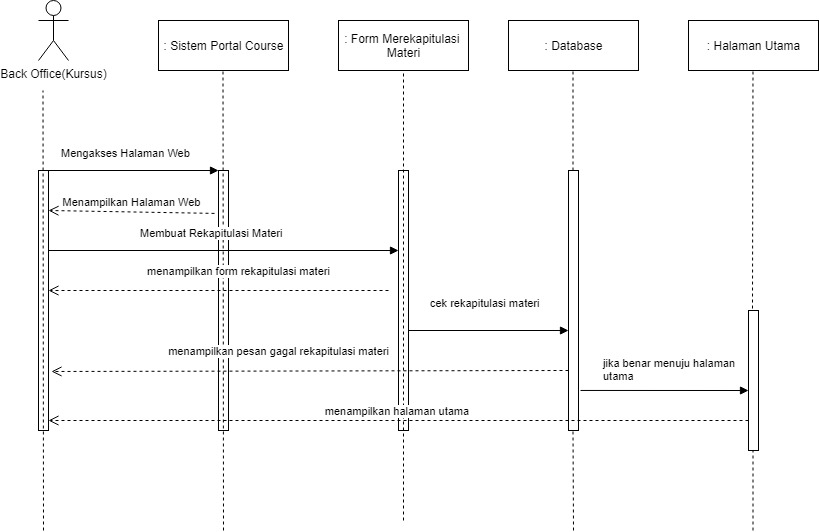
#### **Robustness Diagram**

****

#### **Diagram Kelas**



#### **Sequence Diagram**



### **Use Case #9 <Merekapitulasi Transaksi Pembayaran>**

*Skenario Use Case #3*

*Primary Flow:*

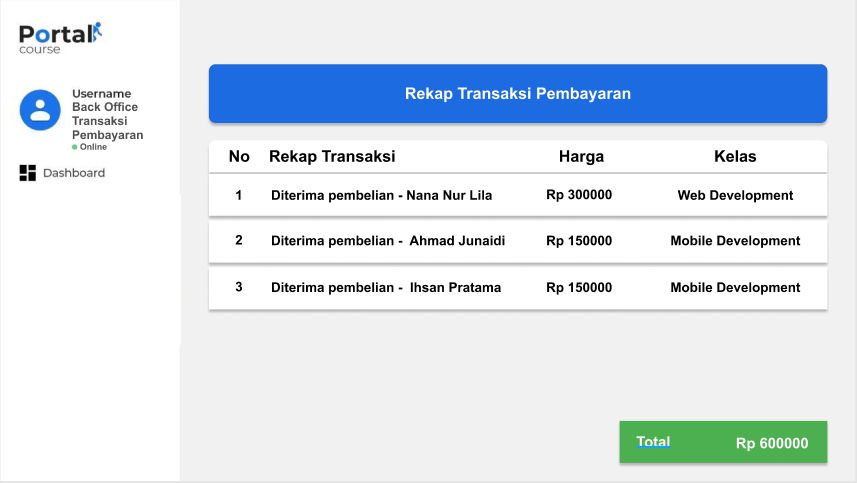
1. Lihat rekapitulasi transaksi pembayaran

*Alternate Flow:*

1. Jika rekapitulasi transaksi gagal akan dialihkan ke *Error page*

#### **Perancangan Antarmuka Usecase #9**

v\_rekapTransaksi



#### **Tabel Identifikasi Antarmuka / Layar / Page**

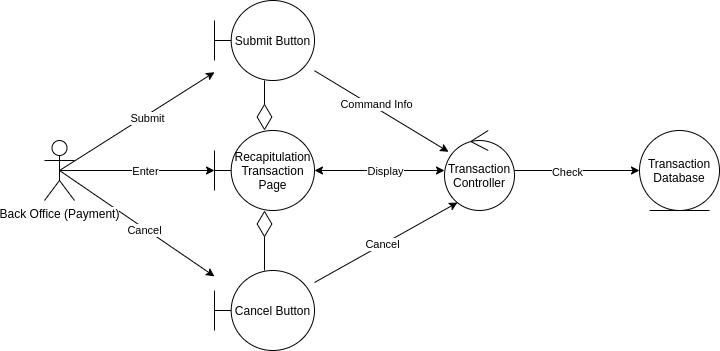
|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
| *12* | *v\_rekapTransaksi* | *Halaman bagi Back Office (Pembayaran) untuk kelol rekapitulasi transaksi* |

#### **Identifikasi Object Baru**

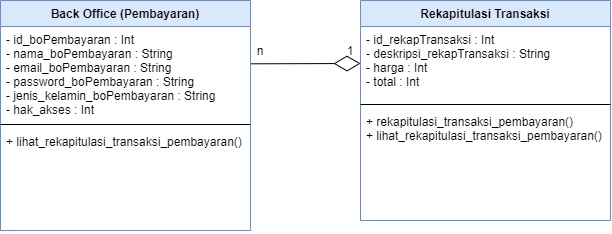
TABEL OBJECT PERANCANGAN

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
| *1.* | *Recapitulastion Transaction Page* | *Boundary* |
| *2.* | *Transaction* | *Entity* |
| *3.* | *Transaction* | *Controller* |

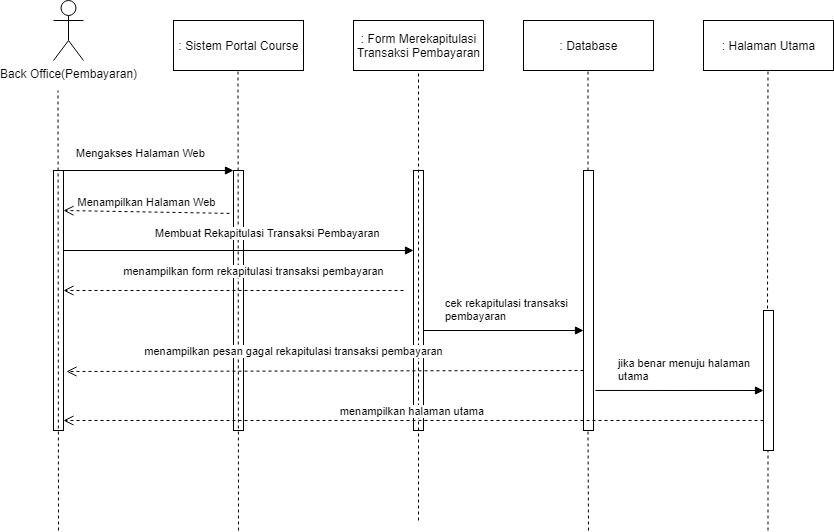
#### **Robustness Diagram**

****

#### **Diagram Kelas**



#### **Sequence Diagram**

**

### **Use Case #10 <Merekapitulasi Anggota Terdaftar>**

*Skenario Use Case #3*

*Primary Flow:*

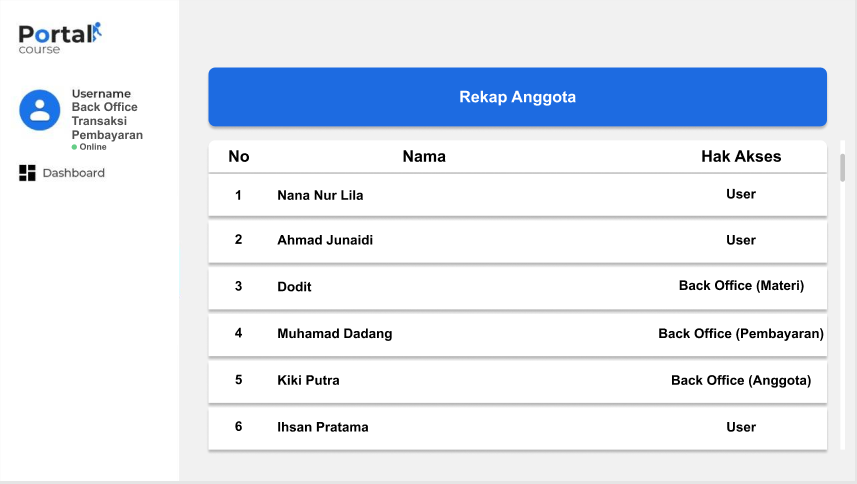
1. Lihat rekapitulasi anggota terdaftar

*Alternate Flow:*

1. Jika rekapitulasi transaksi gagal akan dialihkan ke *Error page*

#### **Perancangan Antarmuka Usecase #10**

v\_rekapAnggota



#### **Tabel Identifikasi Antarmuka / Layar / Page**

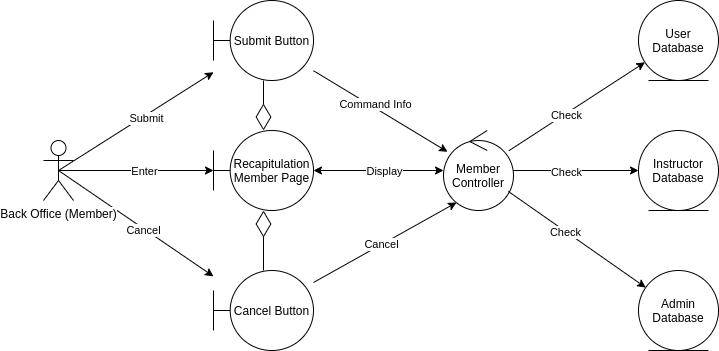
|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
| *13* | *v\_rekapAnggota* | *Halaman bagi Back Office (Anggota) untuk kelola rekapitulasi anggota* |

#### **Identifikasi Object Baru**

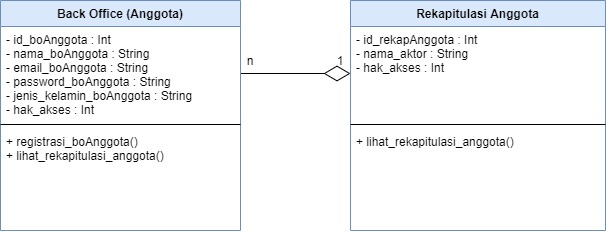
TABEL OBJECT PERANCANGAN

|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
| *1* | *Recapitulation Member Page* | *Boundary* |
| *2* | *User* | *Entity* |
| *3* | *Instructor* | *Entity* |
| *4* | *Admin* | *Entity* |
| *5* | *Member* | *Controller* |

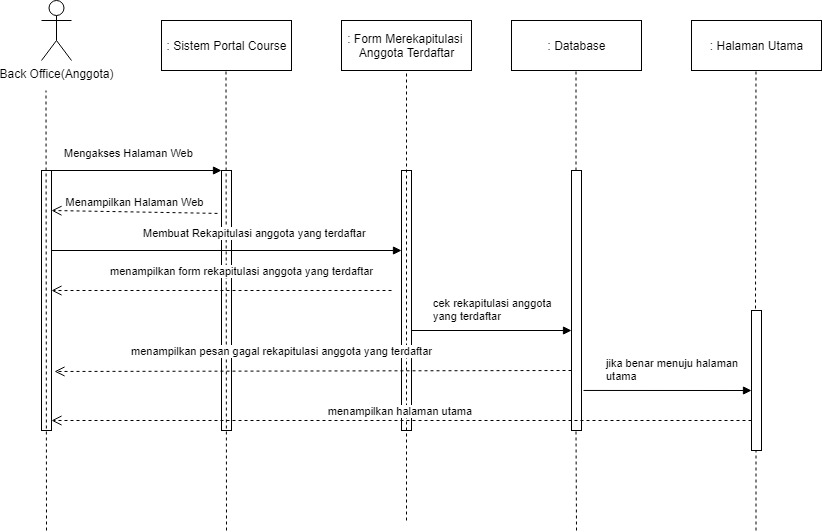
#### **Robustness Diagram**

****

#### **Diagram Kelas**



#### **Sequence Diagram**



### **Use Case #11 <Login>**

*Skenario Use Case #3*

*Primary Flow:*

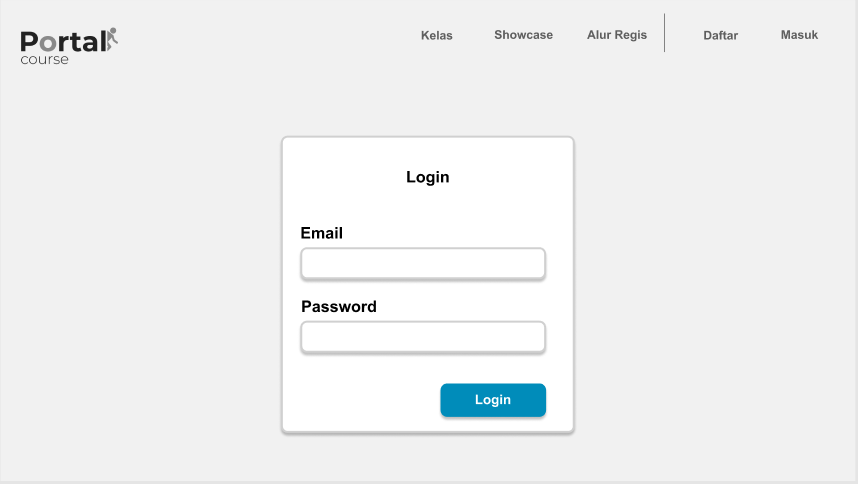
1. Masukkan username
2. Masukkan password
3. Klik button Login

*Alternate Flow:*

1. Registrasi terlebih dahulu jika belum mempunyai akun
2. Akan dialihkan ke Login Page kembali jika Login gagal

#### **Perancangan Antarmuka Usecase #11**

v\_login



#### **Tabel Identifikasi Antarmuka / Layar / Page**

|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
| *14* | *v\_login* | *Halaman bagi semua aktor untuk login* |

*Page v\_login*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *btnKelas* | *Button* | *kelas* | *Jika diklik akan menuju halaman v\_kelas* |
| *btnAlurRegis* | *Button* | *alur* | *Jika diklik akan menuju halaman v\_regisEmail* |
| *btnDaftar* | *Button* | *daftar* | *Jika diklik akan menuju halaman v\_regisEmail* |
| *btnMasuk* | *Button* | *masuk* | *Jika diklik akan menuju halaman v\_login* |
| *tbUser* | *Textbox* | *username* | *Mengisi kotak teks dengan username* |
| *tbPass* | *Textbox* | *password* | *Mengisi kotak teks dengan password* |
| *btnLogin* | *Button* | *login* | *Jika diklik akan mengaktifkan Function login* |

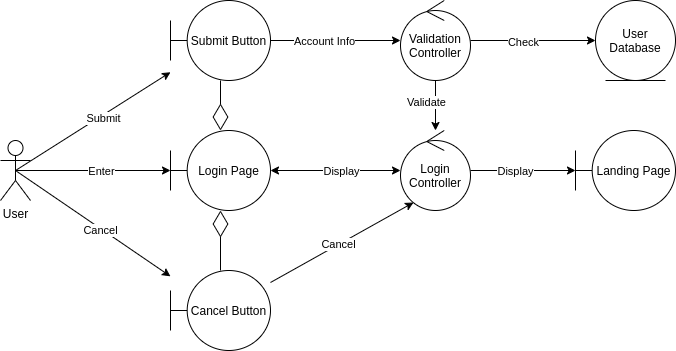
#### **Identifikasi Object Baru**

TABEL OBJECT PERANCANGAN

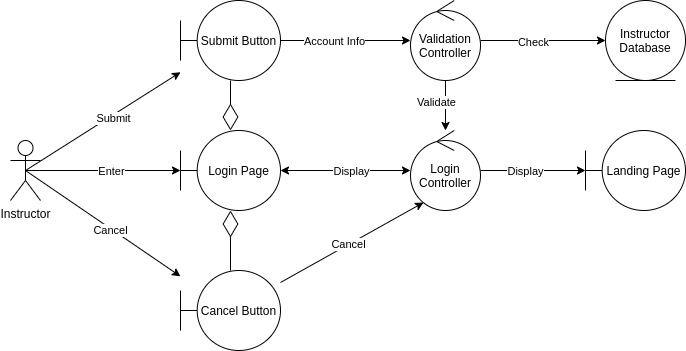
|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
| *1* | *Login Page* | *Boundary* |
| *2* | *Landing Page/Dashboard* | *Boundary* |
| *3* | *User* | *Entity* |
| *4* | *Instructor* | *Entity* |
| *5* | *Admin* | *Entity* |
| *6* | *Login* | *Controller* |
| *7* | *Validation* | *Controller* |

#### **Robustness Diagram**

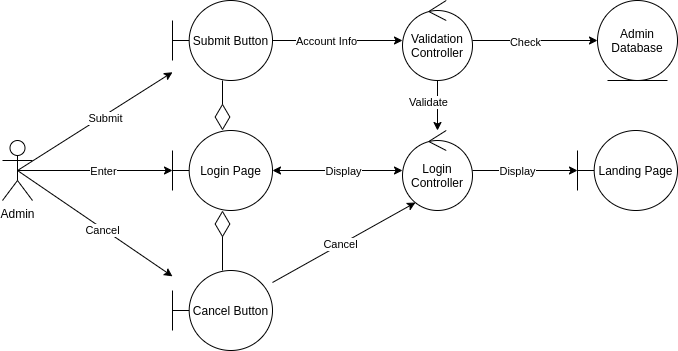
* User

****

* Pengajar Kursus

****

* Admin

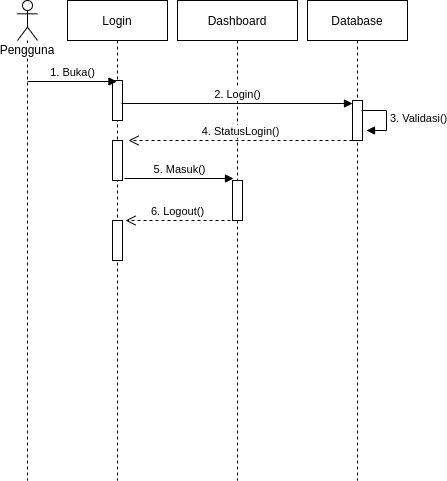


#### **Diagram Kelas**

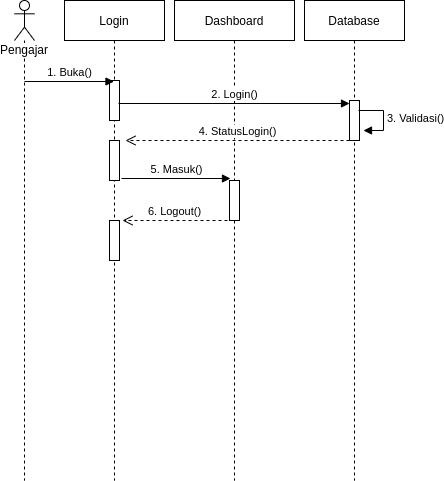


#### **Sequence Diagram**

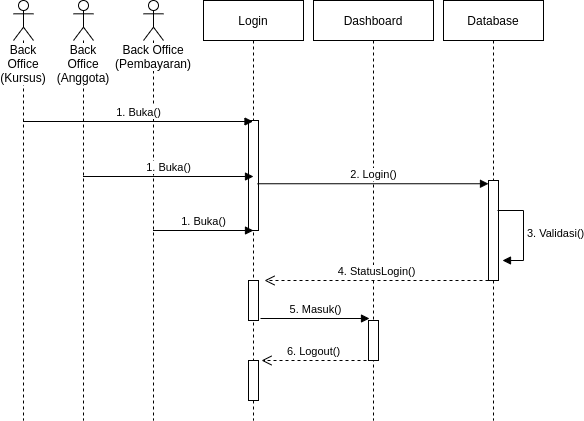
* *Pengguna*

**

* *Pengajar*

**

* *Admin*

**

### **Use Case #12 <Registrasi>**

*Skenario Use Case #3*

*Primary Flow:*

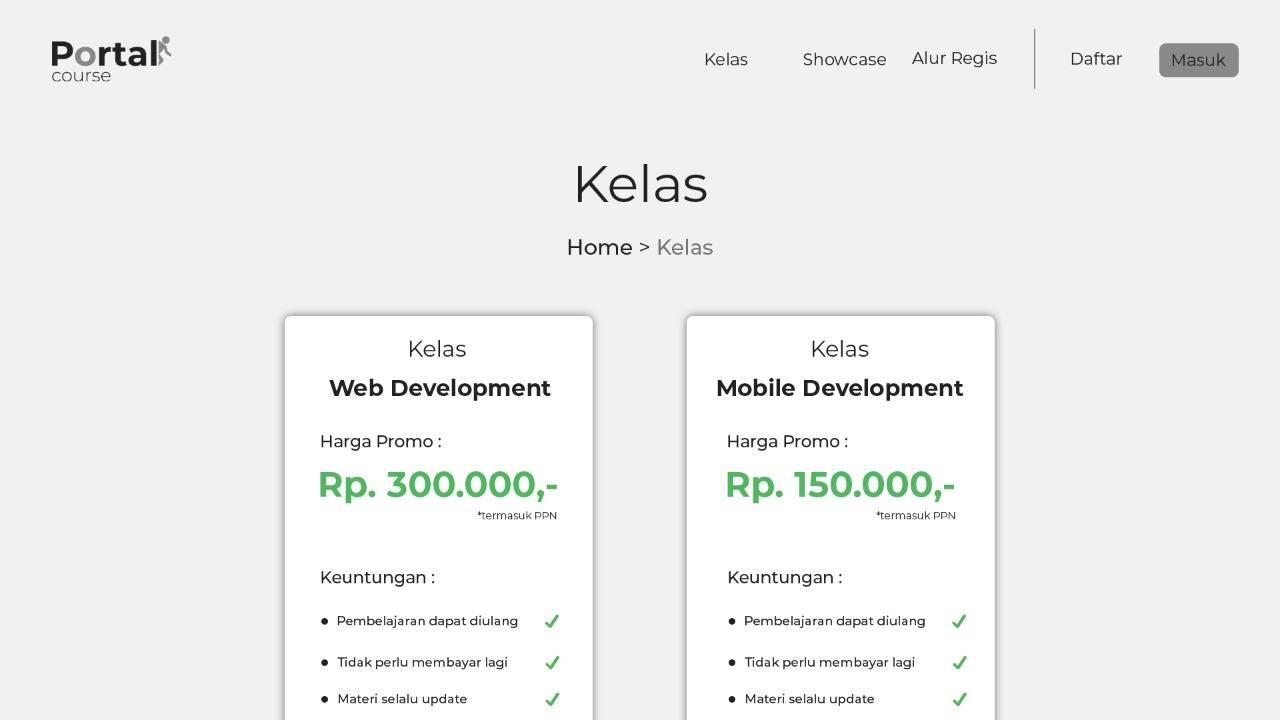
1. Masukkan email
2. Klik button buat akun
3. Masukkan token
4. Klik button konfirmasi
5. Masukkan data diri lengkap
6. Klik button buat akun

*Alternate Flow:*

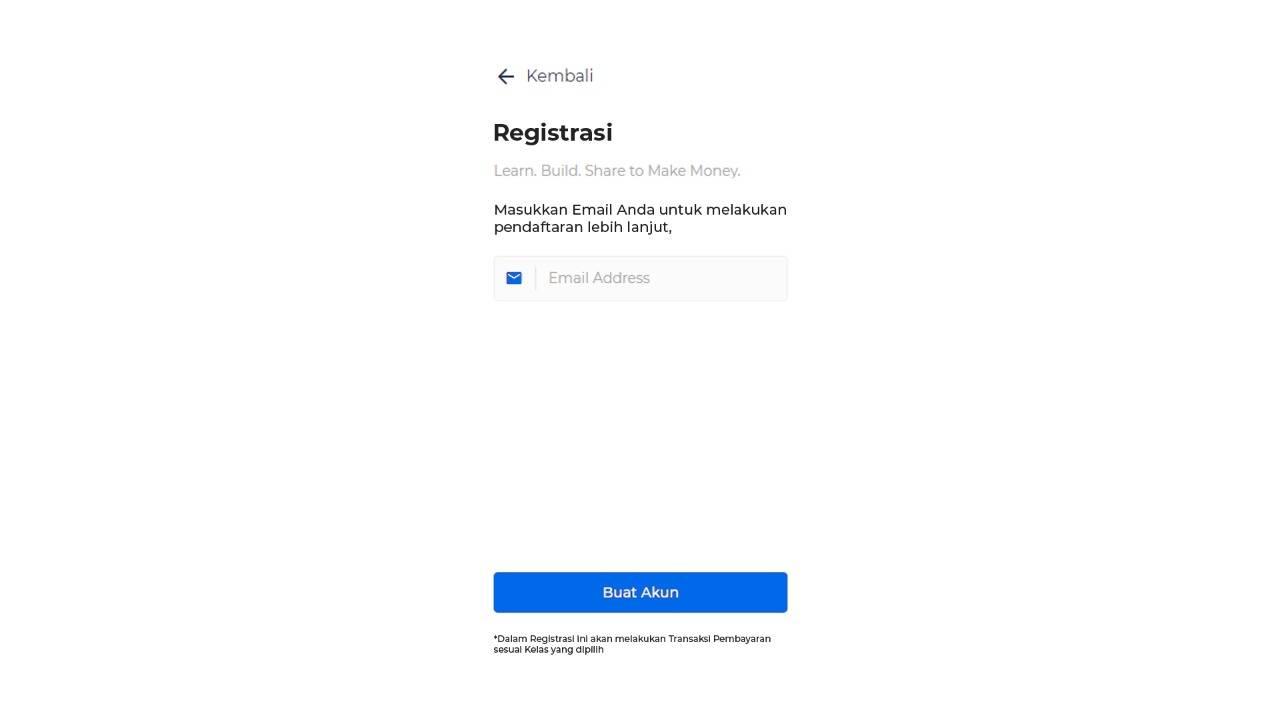
1. Jika sudah mempunyai akun langsung klik button masuk

#### **Perancangan Antarmuka Usecase #12**

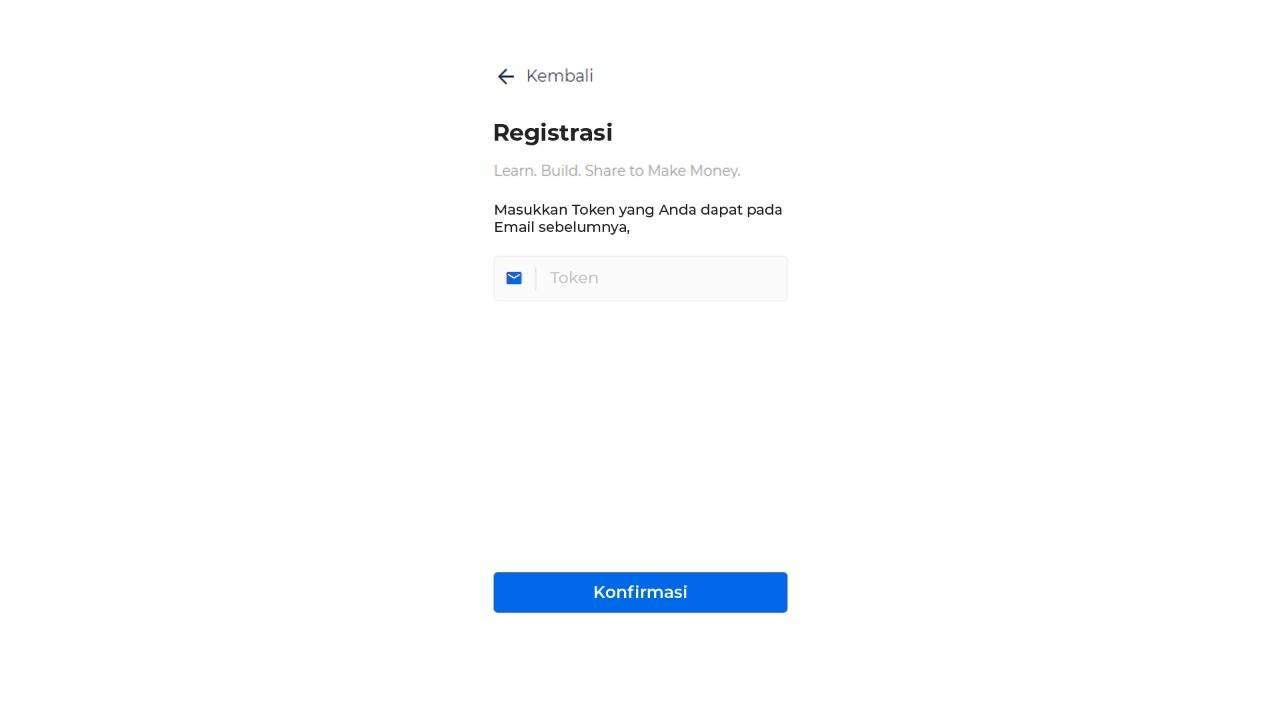
Halaman v\_pilihKelas



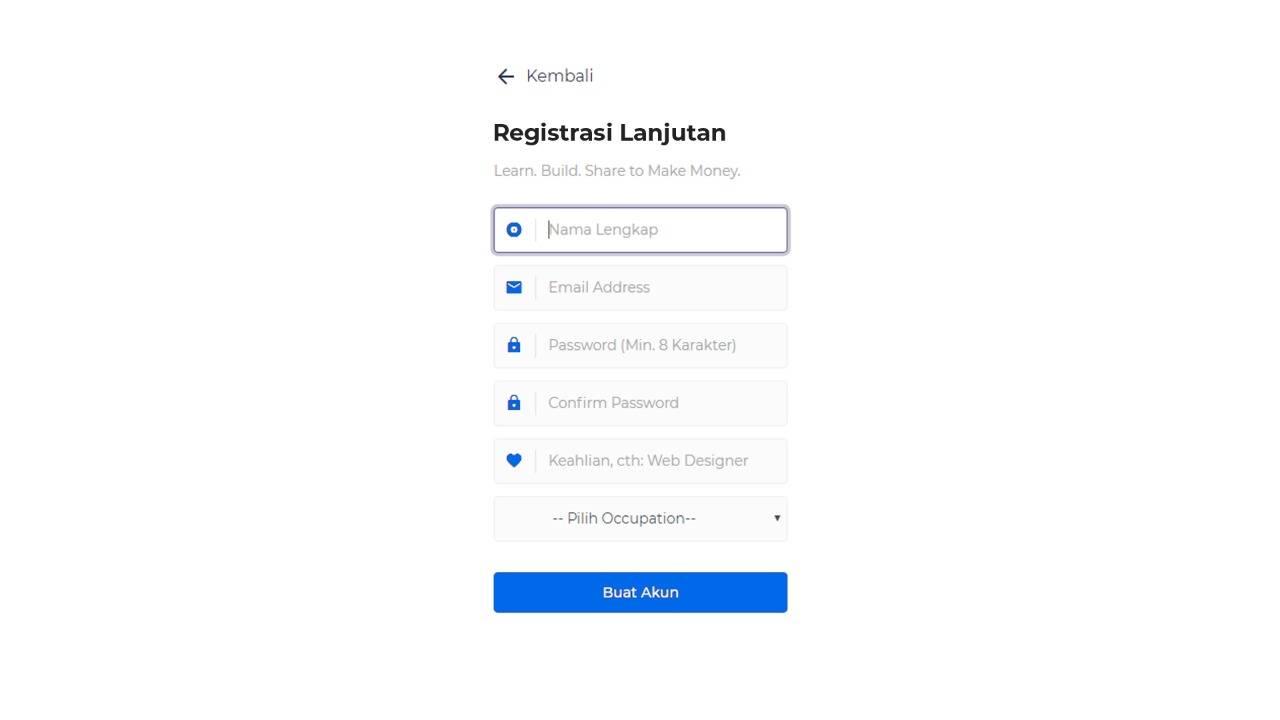
Halaman v\_regisEmail



Halaman v\_regisToken



Halaman v\_regisFull



#### **Tabel Identifikasi Antarmuka / Layar / Page**

|  |  |  |
| --- | --- | --- |
| **ID. LAYAR** | **NAMA LAYAR** | **DESKRIPSI** |
| *15* | *v\_regisEmail* | Halaman untuk mengisi email yang akan dikirim cara pembayaran |
| *3* | *v\_regisToken* | Halaman untuk mengisi token yang diberikan oleh sistem |
| *16* | *v\_regisFull* |  |

UNTUK MASING – MASING ANTAR MUKA / PAGE dibuatkan spesifikasi detil

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

*Page v\_pilihKelas*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *btnKelas1* | *Button* | *kelas1* | *Jika diklik, akan mengaktifkan Function beli\_kursus()* |
| *btnKelas2* | *Button* | *kelas2* | *Jika diklik akan mengaktifkan Function beli\_kursus()* |

*Page v\_regisEmail*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *btn1* | *Button* | *buatAkun* | *Jika diklik, akan mengaktifkan Function Buat Akun* |
| *tx1* | *Text Input* | *email* | *Mengisi kotak teks dengan email pengguna* |

*Page v\_regisToken*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *btn1* | *Button* | *buatKonfirmasi* | *Jika diklik, akan mengaktifkan Function Konfirmasi Token* |
| *tx1* | *Text Input* | *email* | *Mengisi kotak teks dengan token yang sudah terkirim oleh sistem di email* |

*Page v\_regisFull*

|  |  |  |  |
| --- | --- | --- | --- |
| **Id\_Objek** | **JENIS** | **LABEL\*** | **Keterangan\*\*** |
| *txNama* | *Text Input* | *nama* | *Mengisi kotak teks dengan nama lengkap* |
| *txEmail* | *Text Input* | *email* | *Mengisi kotak teks dengan email* |
| *txPass* | *Text Input* | *password* | *Mengisi kotak teks dengan password* |
| *txCPass* | *Text Input* | *cpassword* | *Mengisi kotak teks dengan konfirmasi password* |
| *txAhli* | *Text Input* | *ahli* | *Mengisi kotak teks dengan keahlian* |
| *dd* | *Drop Down* | *occupation* | *Memilih occupation* |

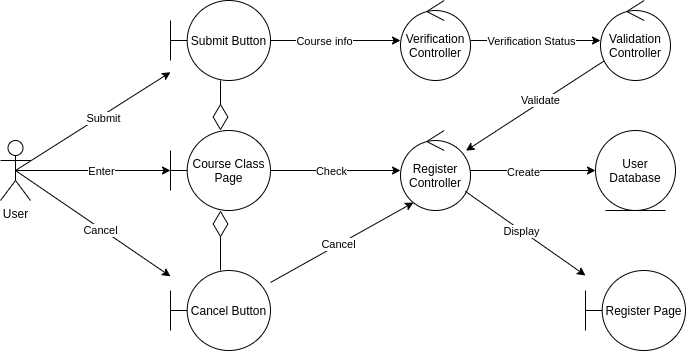
#### **Identifikasi Object Baru**

TABEL OBJECT PERANCANGAN

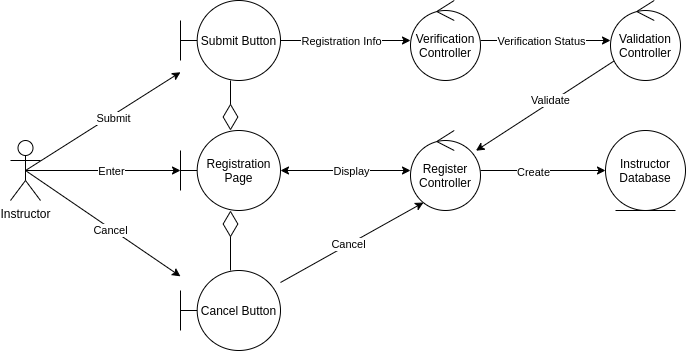
|  |  |  |
| --- | --- | --- |
| **No** | **Nama Object Baru** | **Jenis / Tipe Kelas** |
| 1 | *Course Class Page* | *Boundary* |
| 2 | *Registration Page* | *Boundary* |
| 3 | *User* | *Entity* |
| 4 | *Instructor* | *Entity* |
| 5 | *Admin* | *Entity* |
| 6 | *Register* | *Controller* |
| 7 | *Verification* | *Controller* |
| 8 | *Validation* | *Controller* |

#### **Robustness Diagram**

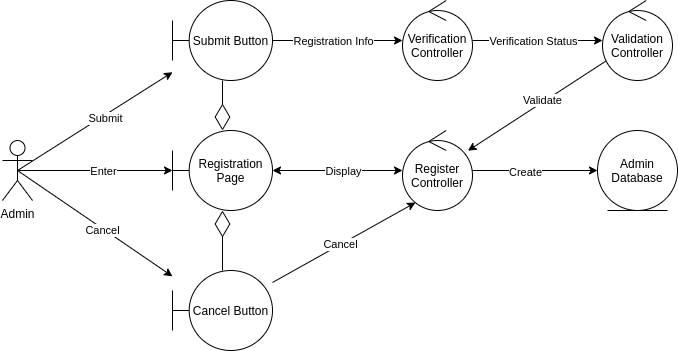
* User



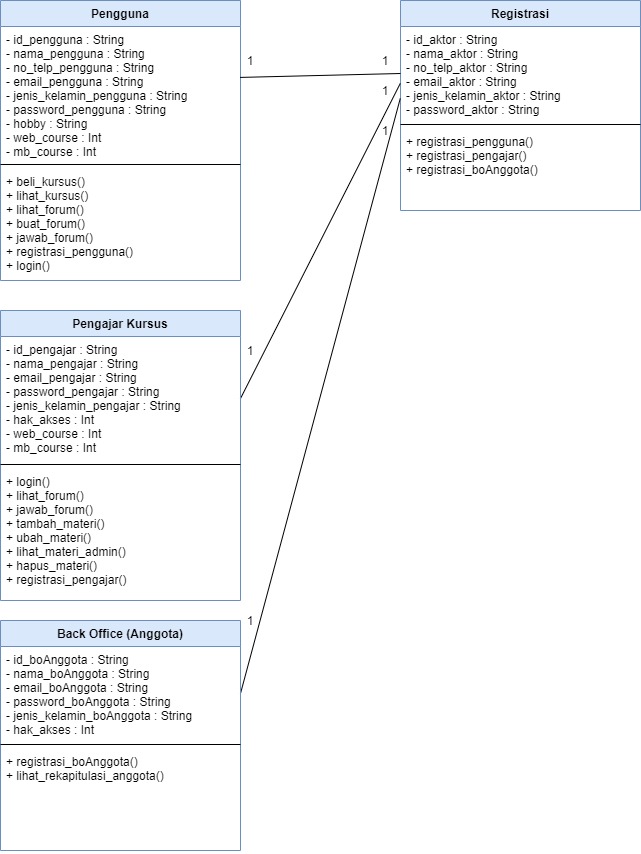
* Pengajar Kursus



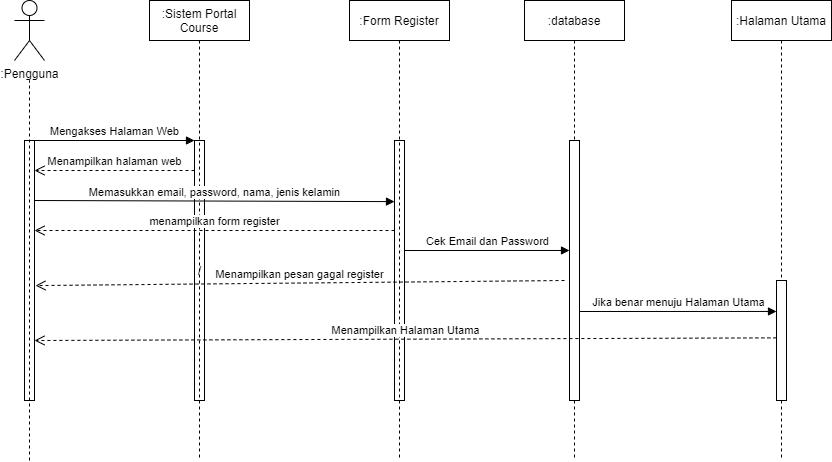
* Admin



#### **Diagram Kelas**



#### **Sequence Diagram**

**

# **Perancangan Detil**

#### **Perancangan Detil Kelas**

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

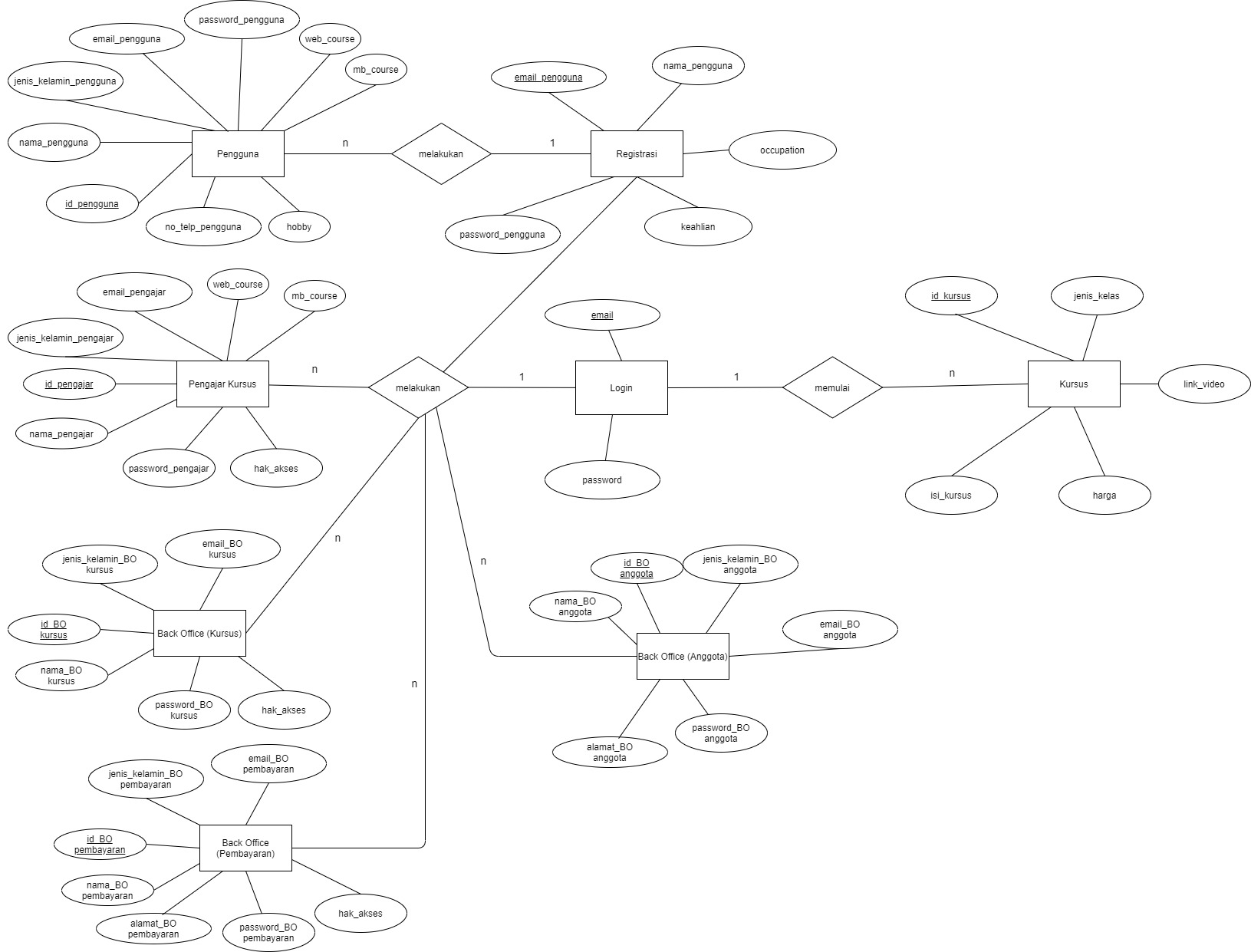
**TABEL KELAS :**

|  |  |  |  |
| --- | --- | --- | --- |
| **ID Kelas** | **Nama Kelas Perancangan** | **Atribute (visibility)** | **Method / Operation** |
| *1* | *Pengguna* | *- id\_pengguna : Int*  *- nama\_pengguna : String*  *- no\_telp\_pengguna : String*  *- email\_pengguna : String*  *- jenis\_kelamin\_pengguna : String*  *- password\_pengguna : String*  *- hobby : String*  *- web\_course : Int*  *- mb\_course : Int* | *+ beli\_kursus()*  *+ lihat\_kursus()*  *+ lihat\_forum()*  *+ buat\_forum()*  *+ jawab\_forum()*  *+ registrasi\_pengajar()*  *+ login()* |
| *2* | *Pengajar Kursus* | *- id\_pengajar : Int*  *- nama\_pengajar : String*  *- email\_pengajar : String*  *- password\_pengajar : String*  *- jenis\_kelamin\_pengajar : String*  *- hak\_akses : Int*  *- web\_course : Int*  *- mb\_course : Int* | *+ login()*  *+ lihat\_forum()*  *+ jawab\_forum()*  *+ tambah\_materi()*  *+ ubah\_materi()*  *+ lihat\_materi\_admin()*  *+ hapus\_materi()*  *+ registrasi\_pengajar()* |
| *3* | *Back Office (Kursus)* | *- id\_boKursus : Int*  *- nama\_boKursus : String*  *- email\_boKursus : String*  *- password\_boKursus : String*  *- jenis\_kelamin\_boKursus : String*  *- hak\_akses : Int* | *+ login()*  *+ ubah\_materi()*  *+ lihat\_materi\_admin()*  *+ hapus\_materi()*  *+ lihat\_rekapitulasi\_materi()* |
| *4* | *Rekapitulasi Materi* | *- id\_rekapMateri : Int*  *- deskripsi\_rekapMateri : String*  *- kelas : String* | *+ lihat\_rekapitulasi\_materi()* |
| *5* | *Back Office (Pembayaran)* | *- id\_boPembayaran : Int*  *- nama\_boPembayaran : String*  *- email\_boPembayaran : String*  *- password\_boPembayaran : String*  *- jenis\_kelamin\_boPembayaran : String*  *- hak\_akses : Int* | *+ lihat\_rekapitulasi\_pembayaran()* |
| *6* | *Rekapitulasi Transaksi* | *- id\_rekapTransaksi : Int*  *- deskripsi\_rekapTransaksi : String*  *- harga : Int*  *- total : Int* | *+ lihat\_rekapitulas\_transaksi\_pembayaran()* |
| *7* | *Back Office (Anggota)* | *- id\_boAnggota : Int*  *- nama\_boAnggota : String*  *- email\_boPAnggota : String*  *- password\_boAnggota : String*  *- jenis\_kelamin\_boAnggota : String*  *- hak\_akses : Int* | *+ lihat\_rekapitulasi\_anggota()*  *+registrasi\_boAnggota()* |
| *8* | *Rekapitulasi Anggota* | *- id\_rekapAnggota : Int*  *- nama\_aktor : String*  *- hak\_akses : Int* | *+ lihat\_rekapitulasi\_anggota()* |
| *9* | *Kursus* | *- id\_kursus : Int*  *- nama\_kelas : String*  *- isi\_kursus : String*  *- link\_vide : String*  *- harga : Int* | *+ lihat\_materi()*  *+ lihat\_materi\_admin()*  *+ lihat\_rekapitulasi\_materi()* |
| *10* | *Forum* | *- id\_forum : Int*  *- pertanyaan\_forum : String*  *- jawaban\_forum : String*  *- bahasa\_program : String* | *+ tambah\_forum()*  *+ jawab\_forum()* |
| *11* | *Login* | *- email : String*  *- password : String* | *+ login()* |
| *12* | *Registrasi* | *- id\_aktor : Int*  *- nama\_aktor : String*  *- no\_telp\_aktor : String*  *- jenis\_kelamin\_aktor : String* | *+ registrasi\_pengguna()*  *+ registrasi\_pengajar()*  *+ registrasi\_boAnggota()* |

*Untuk setiap kelas:*

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

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



#### **Perancangan Algoritma**

*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.*

#### **Algoritma #1**

*Nama Kelas : Login*

*Nama Operasi : login()*

*Algoritma :*

|  |
| --- |
| <?php  $username = $\_POST['username'];  $password = md5($\_POST['password']);    $loginUser = mysql\_query("SELECT \* FROM User WHERE username\_user='$username' and password\_user='$password'");  $cekUser = mysql\_num\_rows($loginUser);  $loginPengajar = mysql\_query("SELECT \* FROM Pengajar WHERE username\_pengajar='$username' and password\_pengajar='$password'");  $cekPengajar = mysql\_num\_rows($loginPengajar);  $loginAdmin = mysql\_query("SELECT \*FROM Admin WHERE username\_admin='$username' and password\_admin='$password'");  $cekAdmin = mysql\_num\_rows($loginAdmin);  if($cekUser > 0){  session\_start();  $\_SESSION['username'] = $username;  $\_SESSION['status'] = "login";  header("location:user/dashboard\_user-home.php");  }else if($cekPengajar > 0){  session\_start();  $\_SESSION['username'] = $username;  $\_SESSION['status'] = "login";  header("location:pengajar/dashboard\_pengajar-home.php");  }else if($cekAdmin > 0){  session\_start();  $\_SESSION['username'] = $username;  $\_SESSION['status'] = "login";  $AdminBOK = mysql\_query("*SELECT \* FROM Admin WHERE username\_admin LIKE ‘BOK\_%’*");  $cekBOK = mysql\_num\_rows($AdminBOK);  $AdminBOP = mysql\_query("*SELECT \* FROM Admin WHERE username\_admin LIKE ‘BOP\_%’*");  $cekBOP = mysql\_num\_rows($AdminBOP);  $AdminBOA = mysql\_query("*SELECT \* FROM Admin WHERE username\_admin LIKE ‘BOA\_%’*");  $cekBOA = mysql\_num\_rows($AdminBOA);  if($cekBOK > 0){  header("location:admin/dashboard\_bok-home.php");  }else if($cekBOP > 0){  header("location:admin/dashboard\_bop-home.php");  }else if($cekBOA > 0){  header("location:admin/dashboard\_boa-home.php");  }  }else{  header("location:index.php");  }  ?> |

#### **Algoritma #2**

*Nama Kelas : Registrasi*

*Nama Operasi : registrasi\_pengguna()*

*Algoritma :*

|  |
| --- |
| <?php  if(isset($\_POST['register'])){  $name = filter\_input(INPUT\_POST, 'name', FILTER\_SANITIZE\_STRING);  $username = filter\_input(INPUT\_POST, 'username', FILTER\_SANITIZE\_STRING);  $password = md5($\_POST["password"], PASSWORD\_DEFAULT);  $email = filter\_input(INPUT\_POST, 'email', FILTER\_VALIDATE\_EMAIL);  $hobby = filter\_input(INPUT\_POST, 'hobby', FILTER\_SANITIZE\_STRING);  $web\_course = filter\_input(INPUT\_POST, 'web\_course', FILTER\_SANITIZE\_NUMBER\_INT);  $mb\_course = filter\_input(INPUT\_POST, 'mb\_course', FILTER\_SANITIZE\_NUMBER\_INT);  $sql = "INSERT INTO User (name, username, email, password, hobby)  VALUES (:name, :username, :email, :password, :hobby, :web\_course, :mb\_course)";  $stmt = $db->prepare($sql);  $params = array(  ":name" => $name,  ":username" => $username,  ":password" => $password,  ":email" => $email,  ":hobby" => $hobby,  ":web\_course" => $web\_course,  ":mb\_course" => $mb\_course  );  $saved = $stmt->execute($params);  if($saved) header("Location: login.php");  }  ?> |

#### **Algoritma #3**

*Nama Kelas : Registrasi*

*Nama Operasi : registrasi\_pengajar()*

*Algoritma :*

|  |
| --- |
| <?php  if(isset($\_POST['register'])){  $name = filter\_input(INPUT\_POST, 'name', FILTER\_SANITIZE\_STRING);  $username = filter\_input(INPUT\_POST, 'username', FILTER\_SANITIZE\_STRING);  $password = md5($\_POST["password"], PASSWORD\_DEFAULT);  $email = filter\_input(INPUT\_POST, 'email', FILTER\_VALIDATE\_EMAIL);  $web\_course = filter\_input(INPUT\_POST, 'web\_course', FILTER\_SANITIZE\_NUMBER\_INT);  $mb\_course = filter\_input(INPUT\_POST, 'mb\_course', FILTER\_SANITIZE\_NUMBER\_INT);  $sql = "INSERT INTO User (name, username, email, password, hobby)  VALUES (:name, :username, :email, :password, :web\_course, :mb\_course)";  $stmt = $db->prepare($sql);  $params = array(  ":name" => $name,  ":username" => $username,  ":password" => $password,  ":email" => $email,  ":web\_course" => $web\_course,  ":mb\_course" => $mb\_course  );  $saved = $stmt->execute($params);  if($saved) header("Location: login.php");  }  ?> |

#### **Algoritma #4**

*Nama Kelas : Registrasi*

*Nama Operasi : registrasi\_boAnggota()*

*Algoritma :*

|  |
| --- |
| <?php  if(isset($\_POST['register'])){  $name = filter\_input(INPUT\_POST, 'name', FILTER\_SANITIZE\_STRING);  $username = filter\_input(INPUT\_POST, 'username', FILTER\_SANITIZE\_STRING);  $password = md5($\_POST["password"], PASSWORD\_DEFAULT);  $email = filter\_input(INPUT\_POST, 'email', FILTER\_VALIDATE\_EMAIL);  $sql = "INSERT INTO User (name, username, email, password, hobby)  VALUES (:name, :username, :email, :password)";  $stmt = $db->prepare($sql);  $params = array(  ":name" => $name,  ":username" => $username,  ":password" => $password,  ":email" => $email  );  $saved = $stmt->execute($params);  if($saved) header("Location: login.php");  }  ?> |

#### **Algoritma #5**

Contoh:

*Nama Kelas : Materi*

*Nama Operasi : tambah\_materi()*

*Algoritma :*

|  |
| --- |
| <?php  include 'koneksi.php';  $judul = $\_GET['judul'];  $deskripsi = $\_GET['desc'];  $link\_video = $\_GET['link'];  $kelas = $\_GET['kelas'];    mysqli\_query($koneksi,"INSERT INTO Materi values('','$judul','$deskripsi','$link\_video','$kelas')");    header("location:dashboard\_admin-materi.php");    ?> |

#### **Algoritma #6**

Contoh:

*Nama Kelas : Materi*

*Nama Operasi : lihat\_materi\_admin()*

*Algoritma :*

|  |
| --- |
| <table border="1">  <tr>  <th>No</th>  <th>Materi</th>  <th>Aksi</th>  </tr>  <?php  include 'koneksi.php';  $no = 1;  $data = mysqli\_query($koneksi,"SELECT \* from Materi");  while($d = mysqli\_fetch\_array($data)){  ?>  <tr>  <td><?php echo $no++; ?></td>  <td><?php echo $d['judul']; ?></td>  <td>  <a href="edit.php?id=<?php echo $d['id']; ?>">EDIT</a>  <a href="hapus.php?id=<?php echo $d['id']; ?>">HAPUS</a>  </td>  </tr>  <?php  }  ?>  </table> |

#### **Algoritma #7**

Contoh:

*Nama Kelas : Materi*

*Nama Operasi : ubah\_materi()*

*Algoritma :*

|  |
| --- |
| <?php  include 'koneksi.php';  $id = $\_GET['id'];  $judul = $\_GET['judul'];  $deskripsi = $\_GET['desc'];  $link\_video = $\_GET['link'];  $kelas = $\_GET['kelas'];    mysqli\_query($koneksi,"UPDATE Materi set judul='$judul', desc='$deskripsi', link = '$link\_video',kelas = '$kelas' WHERE id = $id");    header("location:dashboard\_admin-materi.php");    ?> |

#### **Algoritma #8**

Contoh:

*Nama Kelas : Materi*

*Nama Operasi : hapus\_materi()*

*Algoritma :*

|  |
| --- |
| <?php  include 'koneksi.php';  $id = $\_GET['id'];    mysqli\_query($koneksi,"DELETE FROM Materi WHERE id='$id'");    header("location:dashboard\_admin-materi.php");    ?> |

#### **Algoritma #9**

Contoh:

*Nama Kelas : Rekapitulasi Materi*

*Nama Operasi : lihat\_rekapitulasi\_materi()*

*Algoritma :*

|  |
| --- |
| <table border="1">  <tr>  <th>No</th>  <th>Materi</th>  <th>Kelas</th>  </tr>  <?php  include 'koneksi.php';  $no = 1;  $data = mysqli\_query($koneksi,"SELECT \* from Materi");  while($d = mysqli\_fetch\_array($data)){  ?>  <tr>  <td><?php echo $no++; ?></td>  <td><?php echo $d['judul']; ?></td>  <td><?php echo $d['kelas']; ?></td>  </tr>  <?php  }  ?>  </table> |

#### **Algoritma #10**

Contoh:

*Nama Kelas : Rekapitulasi Transaksi*

*Nama Operasi : lihat\_rekapitulasi\_pembayaran()*

*Algoritma :*

|  |
| --- |
| <table border="1">  <tr>  <th>No</th>  <th>Rekap Transaksi</th>  <th>Harga</th>  <th>Kelas</th>  </tr>  <?php  include 'koneksi.php';  $no = 1;  $data = mysqli\_query($koneksi,"SELECT \* from Transaksi");  while($d = mysqli\_fetch\_array($data)){  ?>  <tr>  <td><?php echo $no++; ?></td>  <td><?php echo $d['nama']; ?></td>  <td><?php echo $d['harga']; ?></td>  <td><?php echo $d['kelas']; ?></td>  </tr>  <?php  }  ?>  </table> |

#### **Algoritma #11**

Contoh:

*Nama Kelas : Rekapitulasi Anggota*

*Nama Operasi : lihat\_rekapitulasi\_anggota()*

*Algoritma :*

|  |
| --- |
| <table border="1">  <tr>  <th>No</th>  <th>Nama</th>  <th>Username</th>  <th>Email</th>  <th>No. Telp</th>  <th>Hobby</th>  <th>Web Class Status</th>  <th>Mobile Class Status</th>  </tr>  <?php  include 'koneksi.php';  $no = 1;  $dataUser = mysqli\_query($koneksi,"SELECT \* from User");  while($d = mysqli\_fetch\_array($dataUser)){  ?>  <tr>  <td><?php echo $no++; ?></td>  <td><?php echo $d['nama']; ?></td>  <td><?php echo $d['username']; ?></td>  <td><?php echo $d['email']; ?></td>  <td><?php echo $d['nomor\_telepon']; ?></td>  <td><?php echo $d['hobby']; ?></td>  <td><?php echo $d['web\_course']; ?></td>  <td><?php echo $d['mb\_course']; ?></td>  </tr>  <?php  }  ?>  </table>  <table border="1">  <tr>  <th>No</th>  <th>Nama</th>  <th>Username</th>  <th>Email</th>  <th>No. Telp</th>  <th>Web Class Status</th>  <th>Mobile Class Status</th>  </tr>  <?php  include 'koneksi.php';  $no = 1;  $dataPengajar = mysqli\_query($koneksi,"SELECT \* from Pengajar");  while($d = mysqli\_fetch\_array($dataPengajar)){  ?>  <tr>  <td><?php echo $no++; ?></td>  <td><?php echo $d['nama']; ?></td>  <td><?php echo $d['username']; ?></td>  <td><?php echo $d['email']; ?></td>  <td><?php echo $d['nomor\_telepon']; ?></td>  <td><?php echo $d['web\_course']; ?></td>  <td><?php echo $d['mb\_course']; ?></td>  </tr>  <?php  }  ?>  </table>  <table border="1">  <tr>  <th>No</th>  <th>Nama</th>  <th>Username</th>  <th>Email</th>  <th>No. Telp</th>  </tr>  <?php  include 'koneksi.php';  $no = 1;  $dataAdmin = mysqli\_query($koneksi,"SELECT \* from Admin");  while($d = mysqli\_fetch\_array($dataAdmin)){  ?>  <tr>  <td><?php echo $no++; ?></td>  <td><?php echo $d['nama']; ?></td>  <td><?php echo $d['username']; ?></td>  <td><?php echo $d['email']; ?></td>  <td><?php echo $d['nomor\_telepon']; ?></td>  </tr>  <?php  }  ?>  </table> |

#### **Perancangan Query**

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

*Query :*

|  |  |  |
| --- | --- | --- |
| *No Query* | *Query* | *Keterangan* |
| *Q-001* | *SELECT \* FROM User* | *Menampilkan keseluruhan User* |
| *Q-002* | *SELECT \* FROM Pengajar* | *Menampilkan keseluruhan Pengajar* |
| *Q-003* | *SELECT \* FROM Admin* | *Menampilkan keseluruhan Admin* |
| *Q-004* | *SELECT \* FROM User WHERE web\_course = 1 or mb\_course = 1* | *Menampilkan Kelas yang telah dibeli* |
| *Q-005* | *SELECT \* FROM Admin WHERE username\_admin LIKE ‘BOK\_%’* | *Menampilkan Admin bagian Back Office (Kursus)* |
| *Q-006* | *SELECT \* FROM Admin WHERE username\_admin LIKE ‘BOP\_%’* | *Menampilkan Admin bagian Back Office (Pembayaran)* |
| *Q-007* | *SELECT \* FROM Admin WHERE username\_admin LIKE ‘BOA\_%’* | *Menampilkan Admin bagian Back Office (Anggota)* |
| *Q-008* | *SELECT judul FROM Materi* | *Menampilkan keseluruhan Judul Materi* |
| *Q-009* | *SELECT video FROM Materi* | *Menampilkan keseluruhan Video Materi* |
| *Q-010* | *SELECT \* FROM Transaksi* | *Menampilkan keseluruhan Transaksi* |
| *Q-011* | *SELECT \* FROM Forum* | *Menampilkan keseluruhan Forum* |

# **Matriks Kerunutan (Requirement Traceability Matrix)**

*Mapping requirement dengan Use Case yang direalisasikan*

|  |  |  |
| --- | --- | --- |
| **Kode FR** | **Nama Functional Requirement** | **Nama Use Case** |
| FR-001 | Overall Project   * Service * Payment | Login  Registrasi  Diskusi Forum  Membeli Kelas  Melihat Kelas  Melihat Materi  Menambah Materi  Mengubah Materi  Menghapus Materi  Merekapitulasi Materi  Merekapitulasi Transaksi Pembayaran  Merekapitulasi Anggota Terdaftar |
| FR-002 | Project (Pengguna)   * Service | Melihat Kelas  Melihat Materi  Diskusi Forum |
| FR-003 | Service (Pengguna)   * Purchase | Registrasi  Membeli Kelas |
| FR-004 | Project (Pengajar)   * Service | Melihat Materi  Menambah Materi  Mengubah Materi  Menghapus Materi  Diskusi Forum |
| FR-005 | Project (BackOffice(Kursus))   * Service | Melihat Materi  Mengubah Materi  Menghapus Materi  Merekapitulasi Materi |
| FR-006 | Project (BackOffice(Pembayaran))   * Service | Merekapitulasi Transaksi Pembayaran |
| FR-007 | Project (BackOffice(Anggota))   * Service | Registrasi  Merekapitulasi Anggota Terdaftar |