

DESKRIPSI PERANCANGAN PERANGKAT LUNAK

KelasinAja

untuk:

Pelajar dan Pengajar

Dipersiapkan oleh:

Azhar Baihaqi Nugraha (1301194353)

Bijak Algifan Putra (1301190461)

Fendi Irfan Amorokhman (1301191447)

Zendy Bramantia Alfareza (1301194145)

Program Studi Informatika

Fakultas Informatika

Jl. Telekomunikasi 1, Dayeuhkolot Bandung

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

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**](#_heading=h.gjdgxs) **5**

[Tujuan Penulisan Dokumen](#_heading=h.30j0zll) [6](#_heading=h.30j0zll)

[Lingkup Masalah](#_heading=h.1fob9te) [6](#_heading=h.1fob9te)

[Definisi dan Istilah](#_heading=h.3znysh7) [6](#_heading=h.3znysh7)

[Referensi](#_heading=h.2et92p0) [6](#_heading=h.2et92p0)

[Sistematika Pembahasan](#_heading=h.tyjcwt) [6](#_heading=h.tyjcwt)

[**Deskripsi Perancangan Global**](#_heading=h.3dy6vkm) **6**

[Rancangan Lingkungan Implementasi](#_heading=h.1t3h5sf) [7](#_heading=h.1t3h5sf)

[Deskripsi Arsitektural](#_heading=h.4d34og8) [7](#_heading=h.4d34og8)

[Deskripsi Komponen](#_heading=h.2s8eyo1) [7](#_heading=h.2s8eyo1)

[**Perancangan Rinci**](#_heading=h.3rdcrjn) **8**

[Realisasi Use Case](#_heading=h.26in1rg) [8](#_heading=h.26in1rg)

[Use Case <nama use case 1>](#_heading=h.lnxbz9) [8](#_heading=h.lnxbz9)

[Identifikasi Kelas](#_heading=h.35nkun2) [8](#_heading=h.35nkun2)

[Sequence Diagram](#_heading=h.1ksv4uv) [8](#_heading=h.1ksv4uv)

[Diagram Kelas](#_heading=h.z337ya) [8](#_heading=h.z337ya)

[Perancangan Detil Kelas](#_heading=h.3j2qqm3) [8](#_heading=h.3j2qqm3)

[Kelas <nama kelas>](#_heading=h.1y810tw) [8](#_heading=h.1y810tw)

[Kelas <nama kelas>](#_heading=h.4i7ojhp) [9](#_heading=h.4i7ojhp)

[Diagram Kelas Keseluruhan](#_heading=h.2xcytpi) [9](#_heading=h.2xcytpi)

[Algoritma/Query](#_heading=h.1ci93xb) [9](#_heading=h.1ci93xb)

[Diagram Statechart](#_heading=h.1pxezwc) [9](#_heading=h.1pxezwc)

[Perancangan Antarmuka](#_heading=h.3whwml4) [9](#_heading=h.3whwml4)

[Perancangan Representasi Persistensi Kelas](#_heading=h.2bn6wsx) [10](#_heading=h.2bn6wsx)

[**Matriks Kerunutan**](#_heading=h.qsh70q) **10**

Setelah Daftar Isi Boleh ada Daftar Tabel dan Daftar Gambar

# 1. Pendahuluan

## Tujuan Penulisan Dokumen

## Deskripsi Perancangan Perangkat Lunak (DPPL) atau disebut juga Software Requirement Spesification (SRS) untuk KelasinAja. Tujuan penulisan dokumen ini adalah untuk memberikan penjelasan mengenai rancangan perangkat lunak yang berfokus pada pembuatan sistem manajemen pembelajaran online. Penjelasan yang akan dimasukan adalah penjelasan mengenai penjelasan umum terkait spesifikasi dan sebagainya

## Dokumen ini digunakan sebagai acuan teknis pengembangan perangkat lunak KelasinAja sekaligus sebagai bahan evaluasi baik proses maupun akhir pengembangannya. Diharapkan dengan adanya dokumen ini, proses pengembagan akan lebih terarah sehingga mendapatkan hasil yang memuaskan dan terhindar dari kekurangan yang lainnya.

## Lingkup Masalah

*Tuliskan dengan ringkas nama aplikasi dan deskripsinya. Maksimal 1 paragraf. Sama dengan yang ditulis di SKPL.*

Perangkat lunak yang akan dibangun adalah perangkat lunak berupa sistem manajemen pembelajaran online, perangkat lunak ini adalah perangkat lunak yang digunakan oleh pengajar dan juga pelajar yang nantinya di dalam website ini pengajar dapat mengatur, membuat dan melaksanakan pembelajaran sedangkan pelajar dapat mengikuti atau melaksanakan pembelajaran yang dibuat oleh pengajar mereka masing masing. Dalam website KelasinAja dapat melakukan hal-hal berikut :

1. User Pengajar dapat membuat kelas
2. User Pengajar dapat membuat form tugas atau ujian
3. User Pengajar dapat memberi sebuah materi berupa file maupun tulisan
4. User Pengajar dapat melaksanakan penyampaian materi secara langsung bertatap muka secara online
5. User Pelajar dapat mengikuti kelas
6. User Pelajar dapat mengupload tugas
7. User Pelajar dapat mengunduh materi pembelajaran
8. User Pelajar dapat mengikuti sesi tatap muka online dengan pembelajarannya
9. Sistem Menyediakan fitur notifikasi

## Definisi dan Istilah

| SKPL | Spesifikasi Kebutuhan Perangkat Lunak dan merupakan spesifikasi dari perangkat lunak yang akan dikembangkan |
| --- | --- |
| DPPL | Dokumen Deskripsi Perancangan Perangkat Lunak(DPPL) merupakan dokumen deskripsi dari perancangan perangkat lunak yang akan dikembangkan dan bertujuan untuk memberikan landasan yang diperlukan dalam proses pengkodean aplikasi. |
| RDMS | adalah suatu sistem atau perangkat lunak yang dirancang untuk mengelola suatu basis data dan menjalankan operasi terhadap data yang diminta banyak pengguna. |
| HTTPS | Singkatan dari Hypertext Transfer Protocol Secure, sebuah ekstensi dari [Hypertext Transfer Protocol](https://id.wikipedia.org/wiki/Hypertext_Transfer_Protocol) (HTTP). HTTPS Ini digunakan untuk [komunikasi aman](https://id.wikipedia.org/w/index.php?title=Keamanan_komunikasi&action=edit&redlink=1) melalui [jaringan komputer](https://id.wikipedia.org/wiki/Jaringan_komputer), dan banyak digunakan di Internet. |
| Laravel | Laravel adalah kerangka kerja aplikasi web berbasis PHP yang sumber terbuka, menggunakan konsep Model-View-Controller. Laravel berada dibawah lisensi MIT, dengan menggunakan GitHub sebagai tempat berbagi kode. |
| HTML | Hypertext Markup Language adalah bahasa markah standar untuk dokumen yang dirancang untuk ditampilkan di peramban internet. |
| PHP | PHP: Hypertext Preprocessor adalah bahasa skrip yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk memprogram situs web dinamis. |
| Bootstrap | Bootstrap adalah kerangka kerja CSS yang sumber terbuka dan bebas untuk merancang situs web dan aplikasi web. |
| Video *Conference* | Konferensi video adalah seperangkat teknologi telekomunikasi interaktif yang memungkinkankan dua pihak atau lebih di lokasi berbeda dapat berinteraksi melalui pengiriman dua arah audio dan video secara bersamaan. |

## Referensi

## Dokumen acuan yang dipergunakan dalam penulisan dokumen ini adalah :

## Template dokumen Deskripsi Perancangan Perangkat Lunak ( DPPL )

## Spesifikasi Kebutuhan Perangkat Lunak ( SKPL )

## Sistematika Pembahasan

*Bagian ini merupakan deskripsi umum dokumen. Tuliskan sistematika pembahasan dokumen DPPL ini.*

*contoh: misalkan dalam bab1 berbicara tentang apa, bab 2 berbicara tentang apa dll*

Dokumen DPPL untuk Perangkat Lunak Pembelajaran Online ini berisi deskripsi kebutuhan pengembangan perangkat lunak secara rinci. Pengorganisasian dokumen dikelompokkan dalam tiga bagian utama, yaitu :

1. Bab 1 Pendahuluan, berisi tujuan penulisan dokumen, lingkup masalah yang ditangani pada perangkat lunak yang akan dibangun, dan deskripsi umum dokumen.
2. Bab 2 Deskripsi Global Perangkat Lunak, berisi penjelasan perangkat lunak yang akan diimplementasikan di lingkungan pengguna secara global. Bagian ini terdiri dari perspektof produk, fungsi produk, karakteristik pengguna, batasan – batasan dan asumsi dan kebergantungan

# Deskripsi Perancangan Global

## Rancangan Lingkungan Implementasi

## Untuk meningkatkan kinerja sistem ini dibutuhkan jaringan internet yang cepat. Sistem inizakan menggunakan firebase sehingga diharapkan dapat membutuhkan waktu yang sedikit dalam menyelesaikan setiap tahapan proses pada sistem. Meningkatnya kecepatan ini diharapkan dapat terjadi di setiap tahapan proses pada sistem. Sistem ini pula akan dijalankan dengan minimum spesifikasi device sebagai berikut:

## Software pada sisi server yang dibutuhkan oleh KelasinAja adalah:

* + Sistem Operasi : *Linux,Windows dan Mac*
  + *Scripting Language* : PHP
  + RDBMS: MySQL

*Hardware* yang dibutuhkan oleh KelasinAja adalah :

* Diatas Windows , Linux, dan Mac.
* An Intel Pentium 4 processor atau yang lebih baru yang mana mendukung SSE3.

## Deskripsi Arsitektural

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

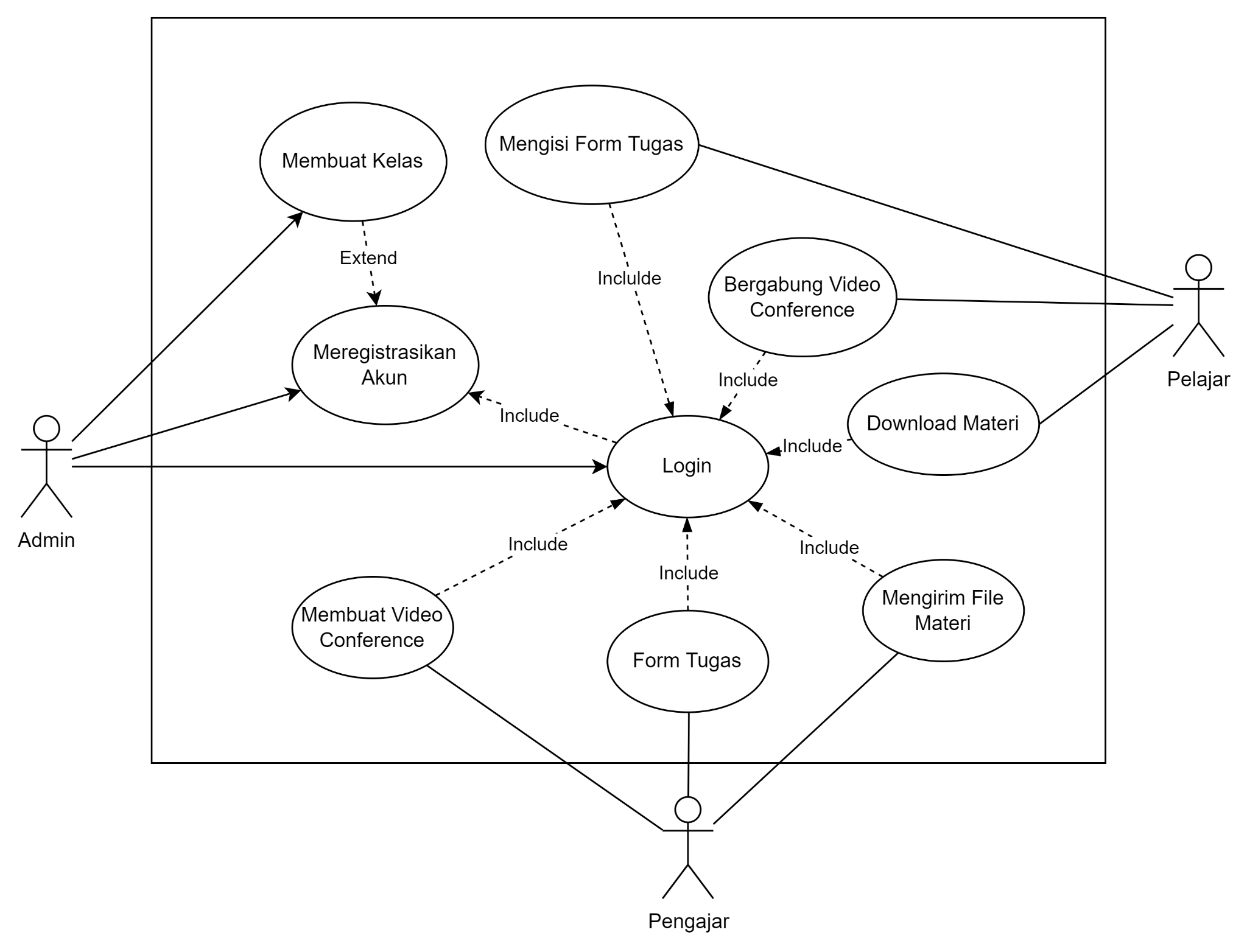
## Deskripsi Komponen

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

| **No** | **Nama Komponen** | **Keterangan** |
| --- | --- | --- |
| 1 | Pengajar (*Teacher*) | Menyampaikan Materi secara langsung dan tidak langsung dan memberikan tugas atau ujian |
| 2 | Pelajar (*Student*) | Mengikuti pembelajaran yang disediakan oleh pengajar, mengerjakan tugas atau ujian dan mengikuti pertemuan tatap muka |

# Perancangan Rinci

## Realisasi Use Case



### Use Case Admin Meregistrasikan Akun

**Use Case Scenario :**

Precondition: Admin belum meregistrasikan akun.

Postcondition : Admin pengguna berhasil membuat akun pelajar/pengajar.

Primary Flow:

1. Pengguna masuk ke halaman website dalam keadaan belum login dan berada di halaman login.
2. Pengguna mengklik tombol registrasi.
3. Sistem akan memberikan form untuk memasukan nama, memasukan email, dan memilih kelas.
4. Pengguna memasukan nama, email dan pilih kelas.
5. Pengguna mengklik submit.
6. Sistem akan menyimpan seluruh atribut yang dimasukan oleh user kedalam database
7. Pengguna tinggal menunggu konfirmasi dari admin
8. Bila sudah dikonfirmasi maka akun sudah dibuat

Alternate Flow:

1. Pada Primary Flow langkah ke 3, ketika pengguna gagal/tidak memasukan nama, email dan kelas maka akan diminta kembali untuk melakukan langkah ke 3.
2. Pada Primary Flow langkah ke 6, sistem gagal menyimpan data ke database maka pengguna akan diarahkan ke login page dan sistem akan memberikan pemberitahuan kepada pengguna bahwa proses gagal.

#### Identifikasi Kelas

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

| *No* | *Nama Kelas Perancangan* | *Tipe Kelas* |
| --- | --- | --- |
| 1 | Admin | Admin |
| *2* | Pelajar | Pelajar |
| *3* | Pengajar | Pengajar |

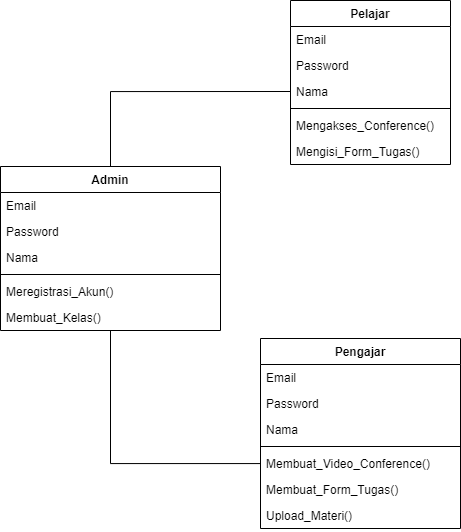
*\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller*

#### Sequence Diagram

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

#### Diagram Kelas

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

******

### Use Case Pelajar bergabung ke video conference

**Use Case Scenario :**

Precondition: Pengguna sudah login dan belum masuk ke video conference, video conference sudah dibuat oleh pengajar.

Postcondition : Pengguna sudah masuk ke video conference

Primary Flow :

1. Pengguna berada di halaman kelas.
2. Pengguna membuka menekan tombol untuk masuk ke video conference.
3. Pengguna akan diarahkan ke halaman video conference

#### Identifikasi Kelas

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

| *No* | *Nama Kelas Perancangan* | *Tipe Kelas* |
| --- | --- | --- |
| 1 | Pelajar | Pelajar |
| *2* | Kelas | Kelas |
|  |  |  |

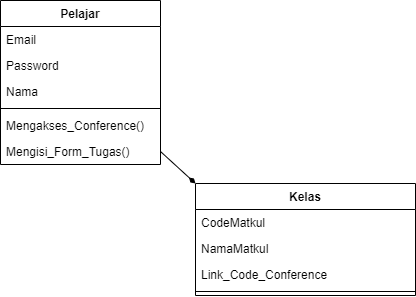
*\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller*

#### Sequence Diagram

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

#### Diagram Kelas

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

******

### Use Case Pengajar membuat video conference

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

**Use Case Scenario :**

Precondition : Pengajar sudah login dan belum ada video conference yang sudah dibuat.

Postcondition : Pengajar berhasil membuat video conference

Primary Flow :

1. Pengajar sudah berada di menu kelas yang ingin dibuatkan video conference
2. Pengajar mengklik tombol “Meet”
3. Pengajar memilih opsi “instant meeting” atau “Schedule meeting”
4. Pengajar masuk ke menu setup audio/video device
5. Pengajar berhasil membuat video conference

#### Identifikasi Kelas

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

| *No* | *Nama Kelas Perancangan* | *Tipe Kelas* |
| --- | --- | --- |
| 1 | Pengajar | Pengajar |
| *2* | Kelas | Kelas |
|  |  |  |

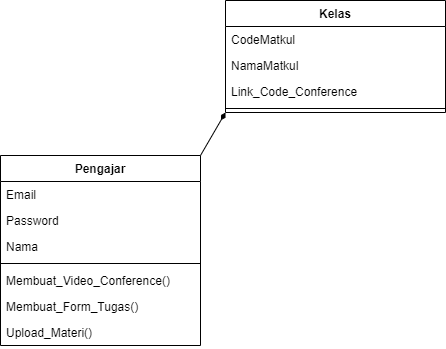
*\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller*

#### Sequence Diagram

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

#### Diagram Kelas

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

******

### Use Case Actor melakukan login

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

**Use Case Scenario :**

Precondition: mengakses platform atau website

Postcondition : Actor telah masuk kedalam akun mereka

Primary Flow:

1. Actor masuk ke dalam website dan dibawa ke halaman login page
2. Actor menekan tombol login, dan sistem melanjutkan ke halaman yang berisi form untuk memasukan email dan password
3. Actor memasukan email dan password dan menekan tombol
4. Sistem akan melakukan pengecekan email dan password ke database sistem
5. Actor akan dibawa ke halaman homepage dan tersedia kelas yang sudah dipilih

Alternate Flow:

1. Pada langkah 4, apabila disaat pengecekan ternyata tidak ditemukan. maka sistem akan memberikan informasi bila email dan password salah. dan akan diminta kembali pada langkah 3

#### Identifikasi Kelas

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

| *No* | *Nama Kelas Perancangan* | *Tipe Kelas* |
| --- | --- | --- |
| 1 | Pengajar | Pengajar |
| *2* | Pelajar | Pelajar |
|  |  |  |

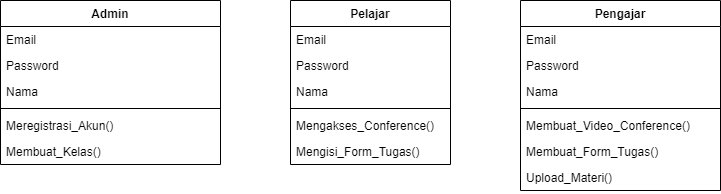
*\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller*

#### Sequence Diagram

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

#### Diagram Kelas

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

******

### Pengajar Mengirim File Materi

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

**Use Case Scenario :**

Precondition : Pengguna berada di halaman Kelas, pengguna telah menyiapkan file yang akan ditambahkan.

Postcondition : File materi telah diunggah dan bisa diakses oleh seluruh anggota kelas.

Primary Flow :

1. Pengguna menekan tombol “Upload Materi”.
2. Pengguna memasukkan judul materi, deskripsi, dan file.
3. Pengguna menekan tombol submit
4. Sistem menyimpan file yang telah diunggah ke database.

Alternate Flow :

1. pada primary flow langkah ke 3, sistem akan memberikan peringatan “harap masukkan data materi” apabila pengguna belum input data yang diperlukan.

#### Identifikasi Kelas

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

| *No* | *Nama Kelas Perancangan* | *Tipe Kelas* |
| --- | --- | --- |
| 1 | Pengajar | Pengajar |
| *2* | Kelas | Kelas |
|  |  |  |

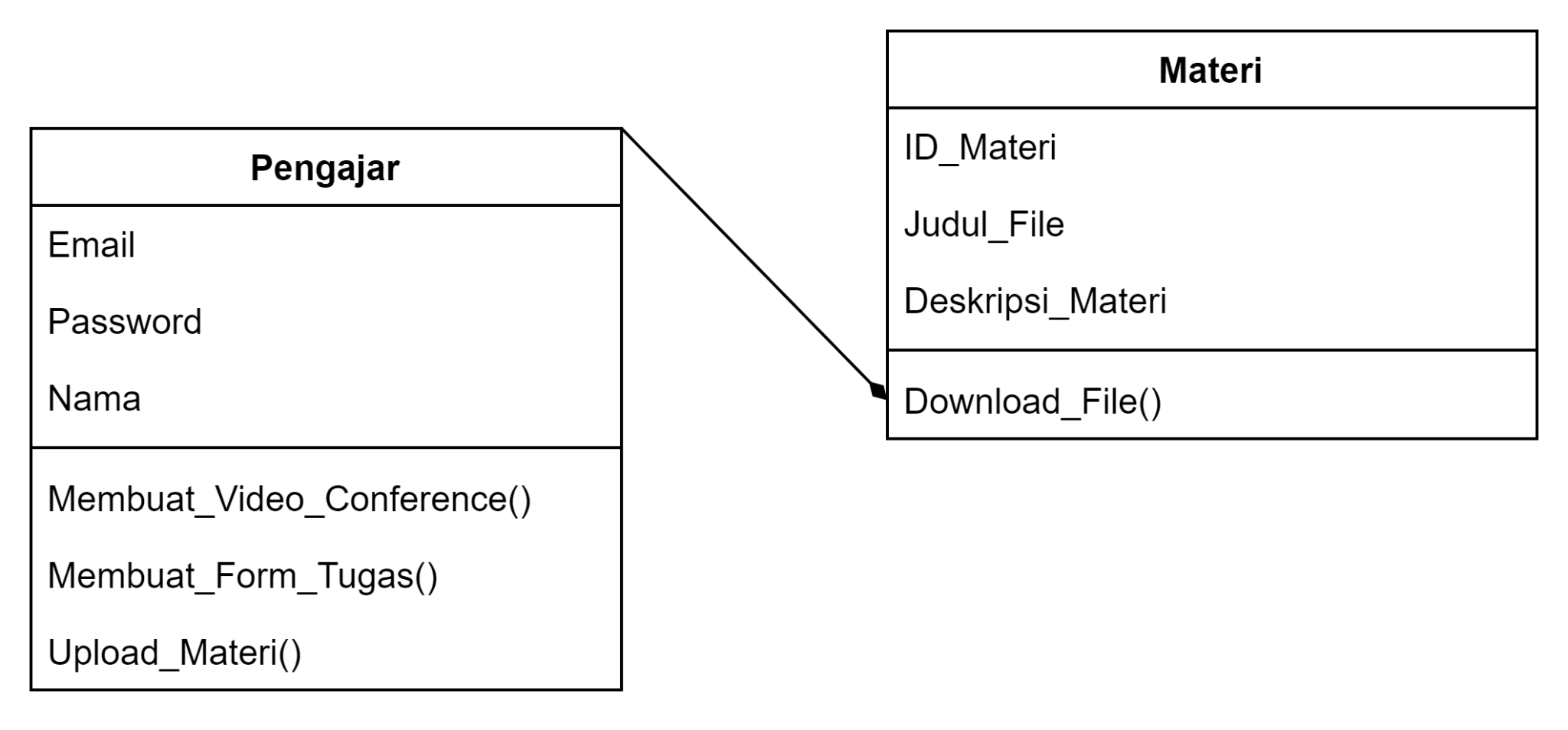
*\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller*

#### Sequence Diagram

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

#### Diagram Kelas

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

******

### Membuat Form Tugas

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

**Use Case Scenario :**

Precondition : Pengajar berada di halaman kelas

Postcondition : Tugas sudah terbuat

Primary Flow :

1. Pengajar masuk ke halaman “Buat Form Tugas”.
2. Pengajar input judul form , deskripsi, dan link embed dari google form.
3. Pengajar mengklik submit.
4. Tugas sudah terbuat

Alternate Flow :

1. Pengajar belum mengisi deskripsi dan embed dari google form saat mengklik submit, sistem akan menampilkan peringatan gagal.

#### Identifikasi Kelas

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

| *No* | *Nama Kelas Perancangan* | *Tipe Kelas* |
| --- | --- | --- |
| 1 | Form\_Tugas | Form\_Exam |
| *2* | Pengajar | User |
|  |  |  |

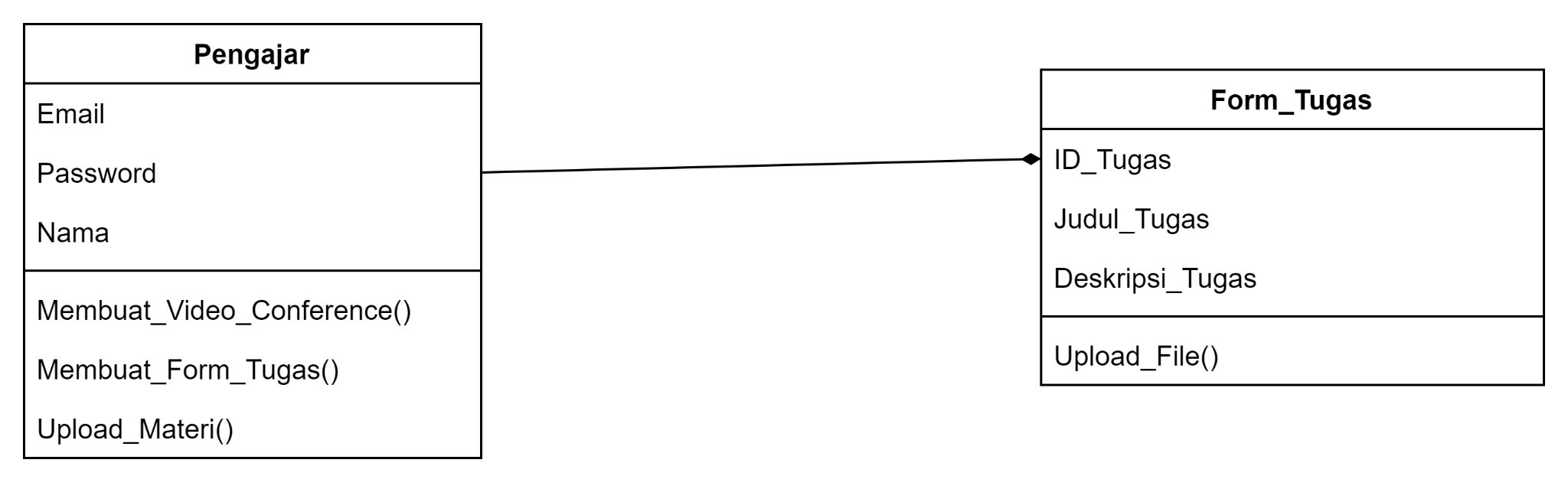
*\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller*

#### Sequence Diagram

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

#### Diagram Kelas

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

******

### Mengisi Form Tugas

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

**Use Case Scenario :**

Precondition : Pengguna berada di halaman tugas dan telah login.

Postcondition : Pengguna berhasil mengisi tugas

Primary Flow :

1. Pelajar telah berada pada halaman kelas
2. Pelajar melakukan input data pada form embed google form.
3. Pelajar mengklik submit

Alternate Flow :

1. pada primary flow langkah ke 3, sistem akan memberikan peringatan “harap masukkan data materi” apabila pengguna belum input data yang diperlukan.

#### Identifikasi Kelas

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

| *No* | *Nama Kelas Perancangan* | *Tipe Kelas* |
| --- | --- | --- |
| 1 | Pelajar | User |
| *2* | Form\_Tugas | From\_exam |
|  |  |  |

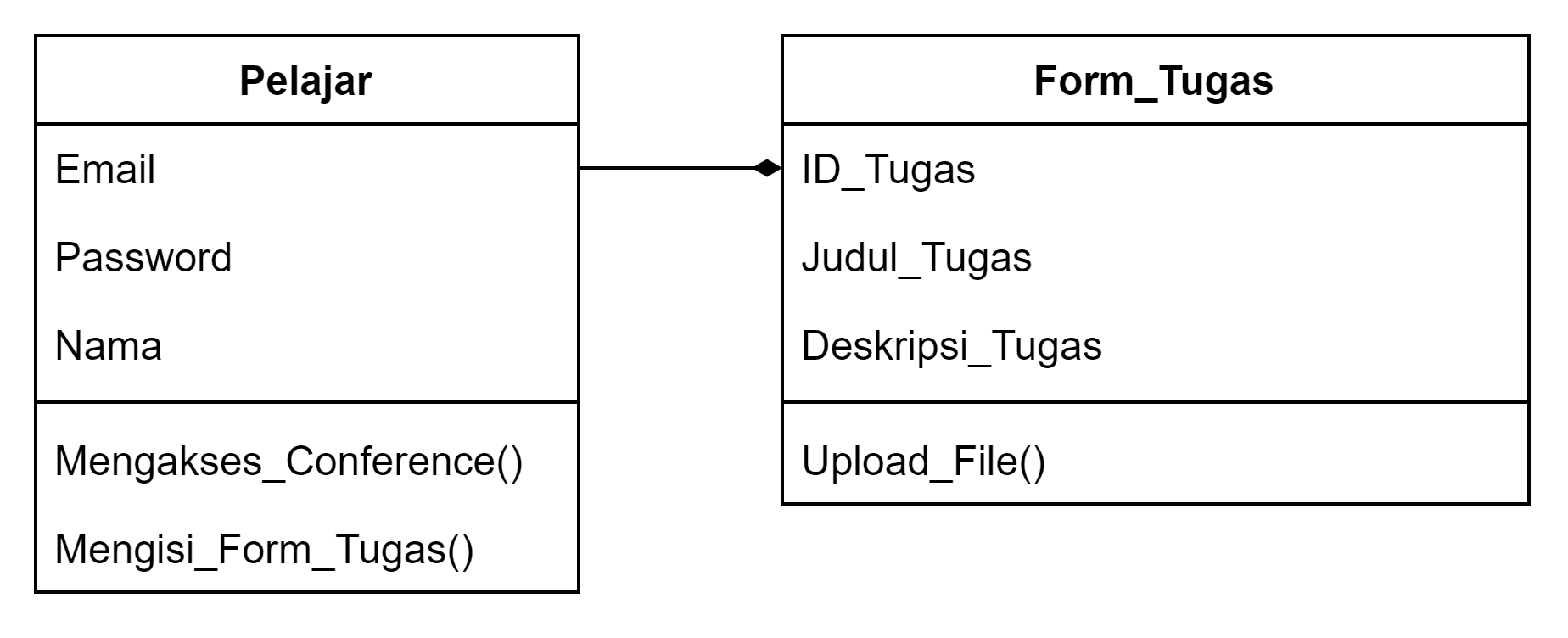
*\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller*

#### Sequence Diagram

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

#### Diagram Kelas

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

******

### Pengajar Mengirim File Materi

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

**Use Case Scenario :**

Precondition : Pengguna berada di halaman Kelas, pengguna telah menyiapkan file yang akan ditambahkan.

Postcondition : File materi telah diunggah dan bisa diakses oleh seluruh anggota kelas.

Primary Flow :

1. Pengguna menekan tombol “Upload Materi”.
2. Pengguna memasukkan judul materi, deskripsi, dan file.
3. Pengguna menekan tombol submit
4. Sistem menyimpan file yang telah diunggah ke database.

Alternate Flow :

1. pada primary flow langkah ke 3, sistem akan memberikan peringatan “harap masukkan data materi” apabila pengguna belum input data yang diperlukan.

#### Identifikasi Kelas

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

| *No* | *Nama Kelas Perancangan* | *Tipe Kelas* |
| --- | --- | --- |
| 1 | Pengajar | Pengajar |
| *2* | Kelas | Kelas |
|  |  |  |

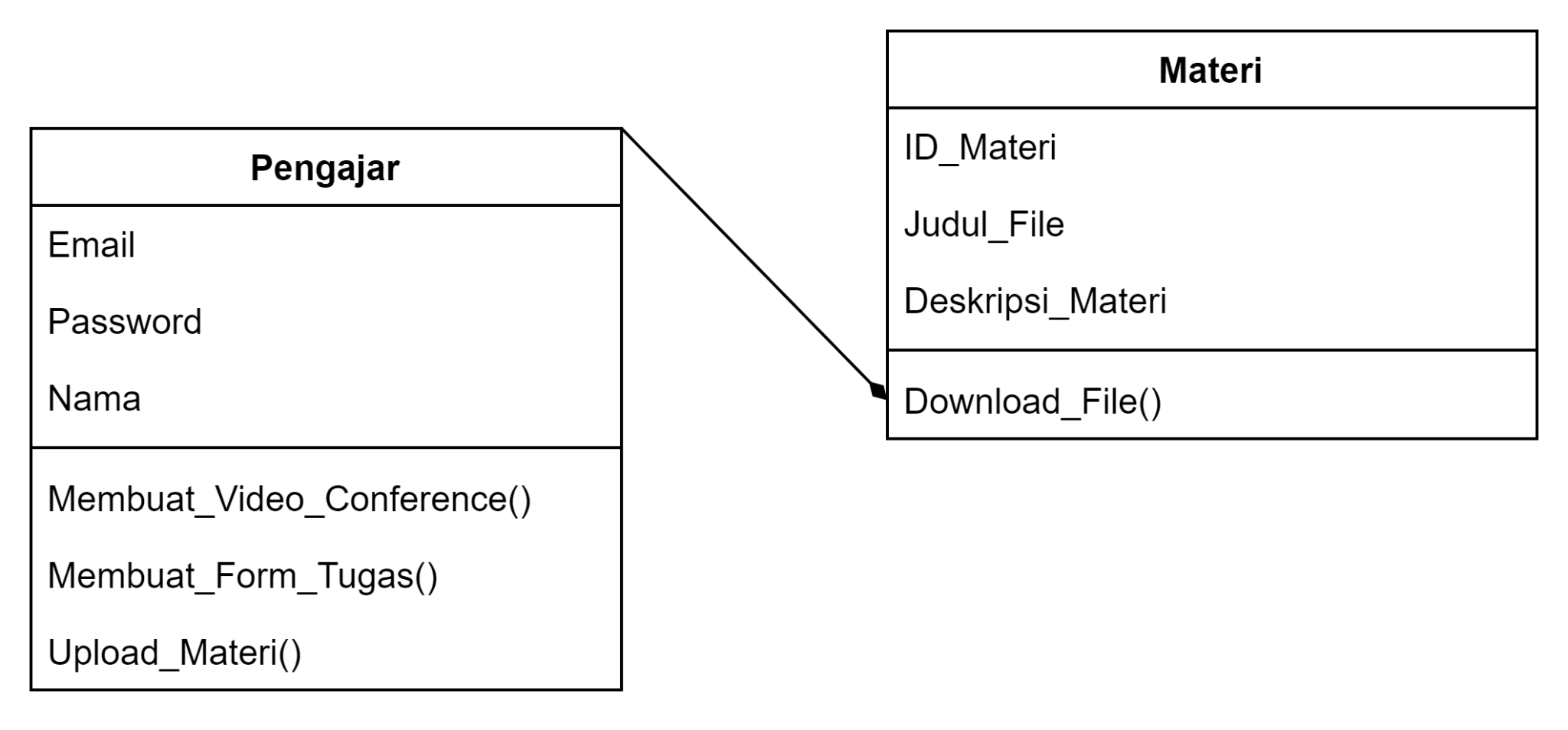
*\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller*

#### Sequence Diagram

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

#### Diagram Kelas

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

******

### Pengajar Mengirim File Materi

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

**Use Case Scenario :**

Precondition : Pengguna berada di halaman Kelas, pengguna telah menyiapkan file yang akan ditambahkan.

Postcondition : File materi telah diunggah dan bisa diakses oleh seluruh anggota kelas.

Primary Flow :

1. Pengguna menekan tombol “Upload Materi”.
2. Pengguna memasukkan judul materi, deskripsi, dan file.
3. Pengguna menekan tombol submit
4. Sistem menyimpan file yang telah diunggah ke database.

Alternate Flow :

1. pada primary flow langkah ke 3, sistem akan memberikan peringatan “harap masukkan data materi” apabila pengguna belum input data yang diperlukan.

#### Identifikasi Kelas

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

| *No* | *Nama Kelas Perancangan* | *Tipe Kelas* |
| --- | --- | --- |
| 1 | Pengajar | Pengajar |
| *2* | Kelas | Kelas |
|  |  |  |

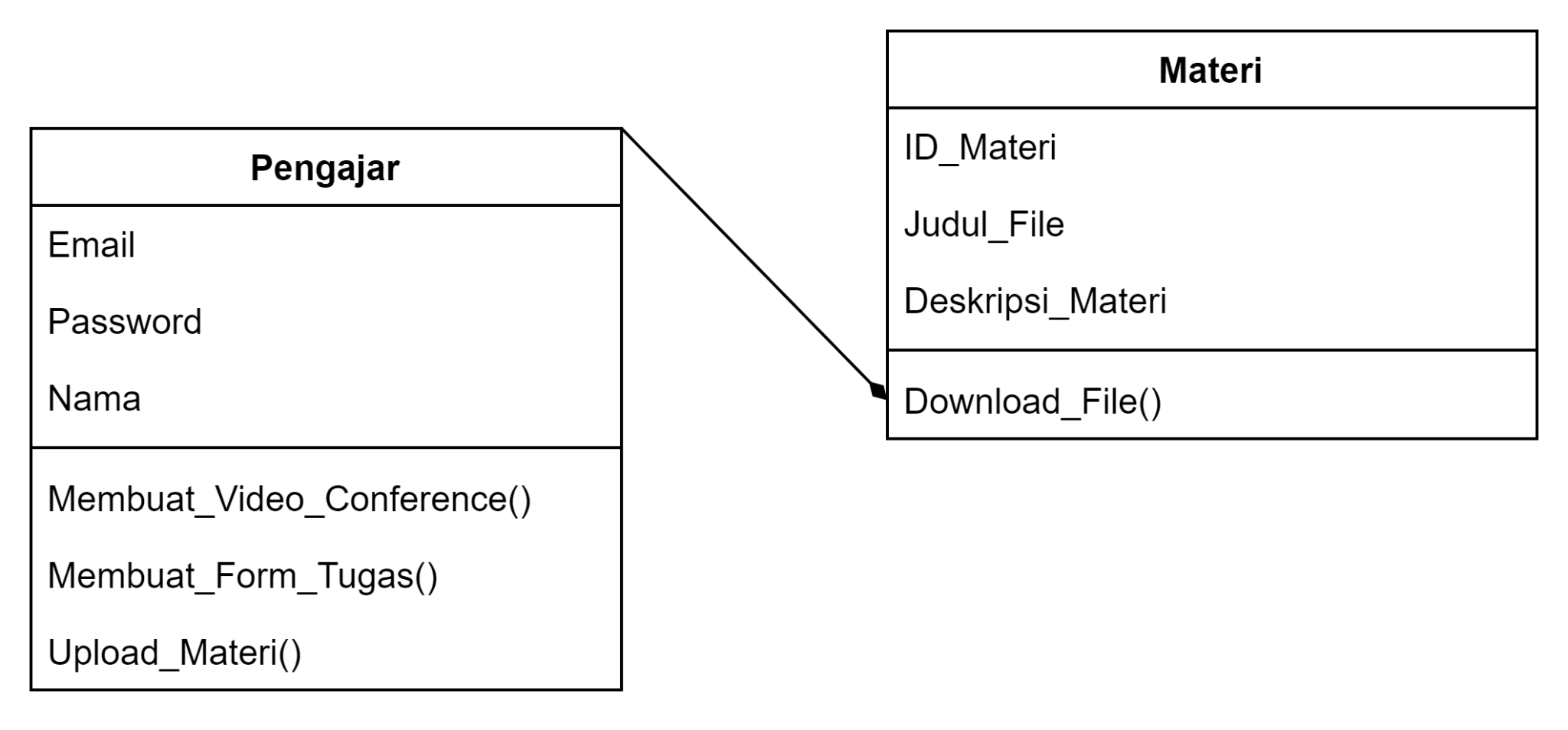
*\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller*

#### Sequence Diagram

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

#### Diagram Kelas

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

******

### Pengajar Mengirim File Materi

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

**Use Case Scenario :**

Precondition : Pengguna berada di halaman Kelas, pengguna telah menyiapkan file yang akan ditambahkan.

Postcondition : File materi telah diunggah dan bisa diakses oleh seluruh anggota kelas.

Primary Flow :

1. Pengguna menekan tombol “Upload Materi”.
2. Pengguna memasukkan judul materi, deskripsi, dan file.
3. Pengguna menekan tombol submit
4. Sistem menyimpan file yang telah diunggah ke database.

Alternate Flow :

1. pada primary flow langkah ke 3, sistem akan memberikan peringatan “harap masukkan data materi” apabila pengguna belum input data yang diperlukan.

#### Identifikasi Kelas

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

| *No* | *Nama Kelas Perancangan* | *Tipe Kelas* |
| --- | --- | --- |
| 1 | Pengajar | Pengajar |
| *2* | Kelas | Kelas |
|  |  |  |

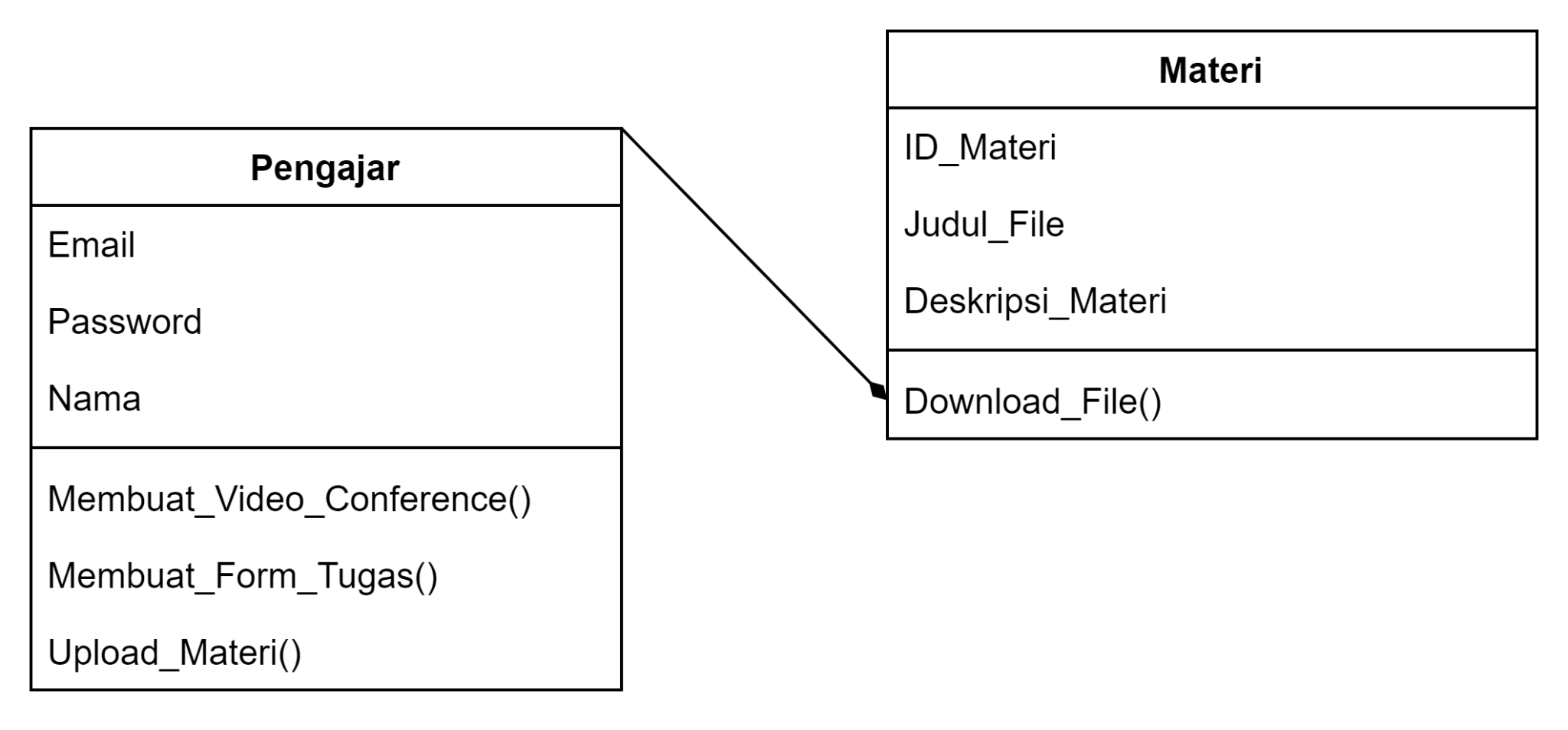
*\*Tipe kelas seperti Boundary(Interface), Entity(Database), Controller*

#### Sequence Diagram

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

#### Diagram Kelas

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

******

## Perancangan Detail Kelas

*Bagian ini diisi dengan daftar seluruh kelas dalam tabel berikut:*

| *No* | *Nama Kelas Perancangan* | *Nama Kelas Analisis Terkait* |
| --- | --- | --- |
| *1* | *Pelajar* | *Pelajar* |
| *2* | *Admin* | *Admin* |
| *3* | *Pengajar* | *Pengajar* |
| *4* | *Kelas* | *Kelas* |
| *5* | *Form\_Tugas* | *Form\_Tugas* |
| *6* | *Materi* | *Materi* |

*Untuk setiap kelas:*

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

### Kelas Pelajar

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

*Nama Kelas : Pelajar*

| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| --- | --- | --- |
| Mengakses\_Conference | public | Method untuk join meeting |
| Mengisi\_Form\_Tugas | public | Method untuk mengisi form tugas |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| Email | private | string |
| Password | private | string |
| Nama | private | string |

### Kelas Pengajar

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

*Nama Kelas : Pengajar*

| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| --- | --- | --- |
| Membuat\_Video\_Conference | public | Method untuk membuat meet conference |
| Membuat\_Form\_Tugas | public | Method untuk membuat form tugas |
| Upload\_Materi | public | Method untuk upload file materi |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| Email | private | string |
| Password | private | string |
| Nama | private | string |

### Kelas Admin

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

*Nama Kelas : Admin*

| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| --- | --- | --- |
| Meregistrasi\_Akun() | public | Method untuk meregistrasikan akun pengajar dan pelajar |
| Membuat\_Kelas() | public | Method untuk membuat kelas baru ke database |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| Email | private | string |
| Password | private | string |
| Nama | private | string |

### Kelas Kelas

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

*Nama Kelas : Kelas*

| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| --- | --- | --- |
| CodeMatkul | private | string |
| NamaMatkul | private | string |
| Link\_Code\_Conference | private | string |

### Kelas Form\_Tugas

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

*Nama Kelas : Form\_Tugas*

| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| --- | --- | --- |
| Upload\_File() | public | Method untuk mengupload file ke database |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| ID\_Tugas | private | string |
| Judul\_Tugas | private | string |
| Deskripsi Tugas | private | string |
| Link\_Embed\_Form | private | string |

### Kelas Materi

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

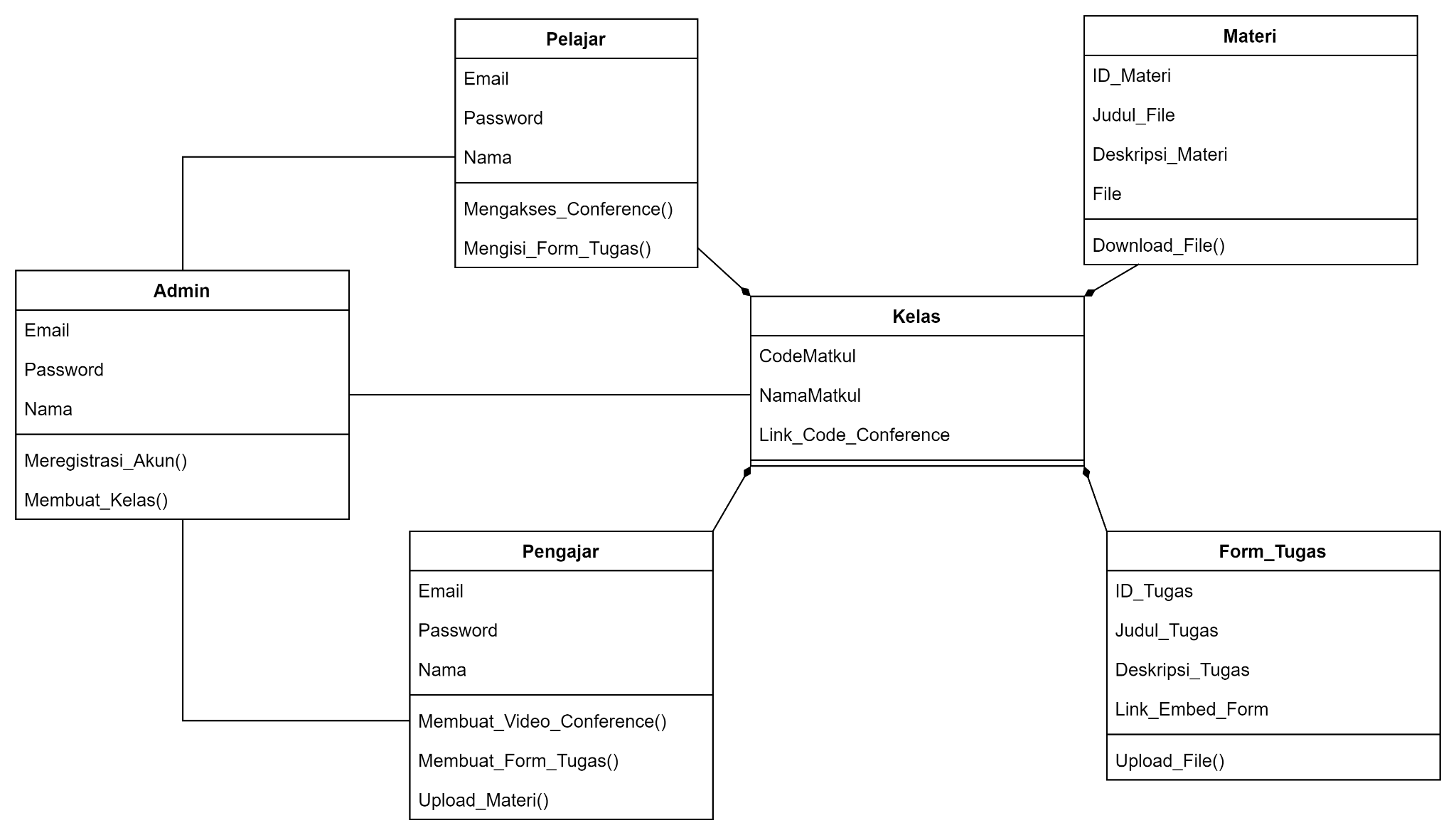
*Nama Kelas : Materi*

| ***Nama Operasi*** | ***Visibility***  ***(private, public)*** | ***Keterangan*** |
| --- | --- | --- |
| Download\_File() | public | Method untuk download file dari database |
| ***Nama Atribut*** | ***Visibility***  ***(private, public)*** | ***Tipe*** |
| ID\_Materi | private | string |
| File\_Name | private | string |
| Deskripsi Materi | private | string |
| File | private | ? |

### 

## Diagram Kelas Keseluruhan

*Bagian ini diisi dengan diagram kelas keseluruhan.*

**

## Algoritma/Query

*Bagian ini hanya diisi untuk kerangka algoritma untuk* **method-method****dari Class** *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* |
|  |  |  |
|  |  |  |

## Perancangan Antarmuka

*Bagian ini diisi dengan versi awal prototipe antarmuka.*

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

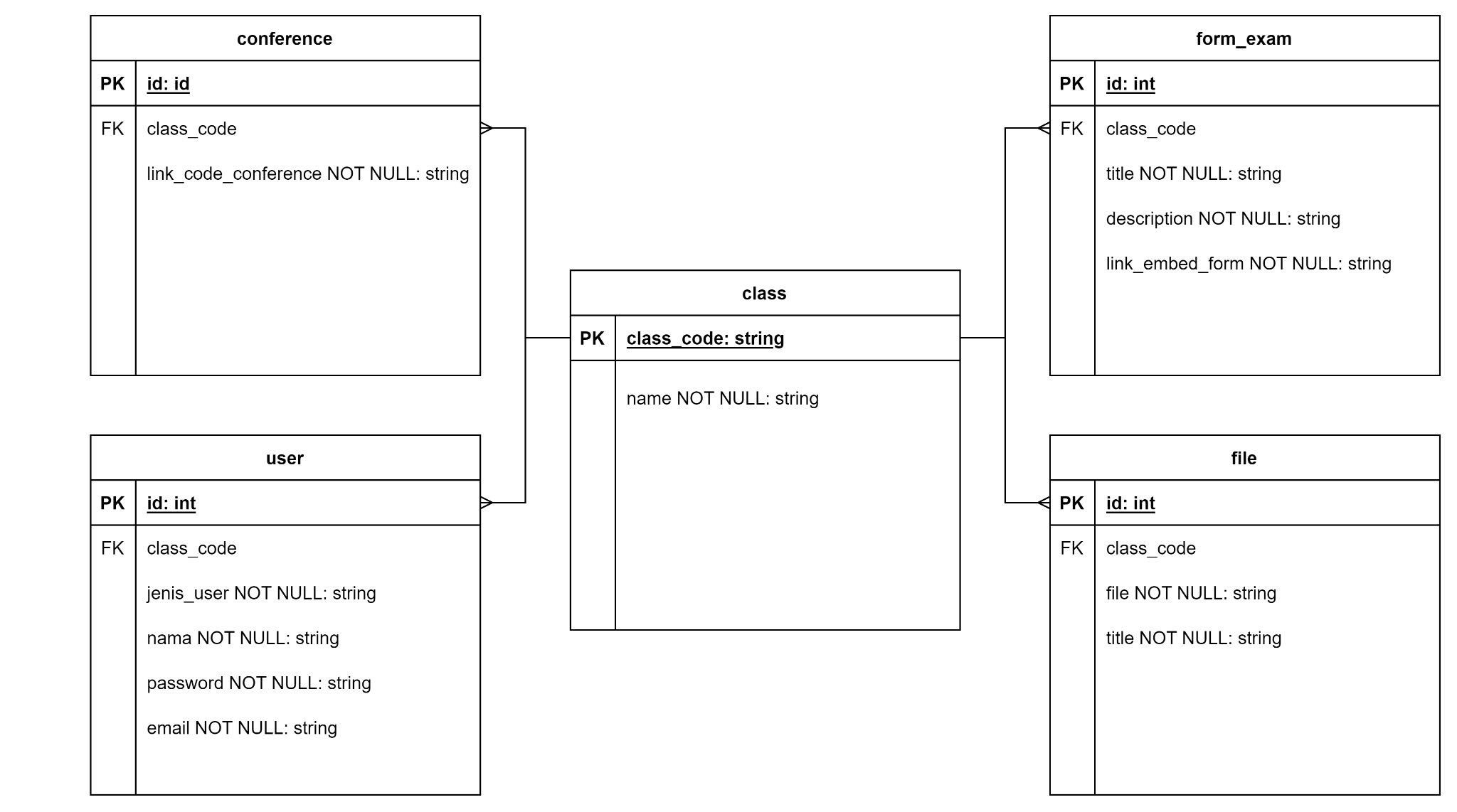
*Antarmuka : {diisi dengan no. layar atau no gambar rancangan antarmuka}*

| **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

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



# Matriks Kerunutan

*Mapping use case dengan kelas-kelas terkait*

| **Requirement** | **Usecase Terkait** | **Kelas** |
| --- | --- | --- |
| FR-01 |  |  |
| FR-02 |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# 