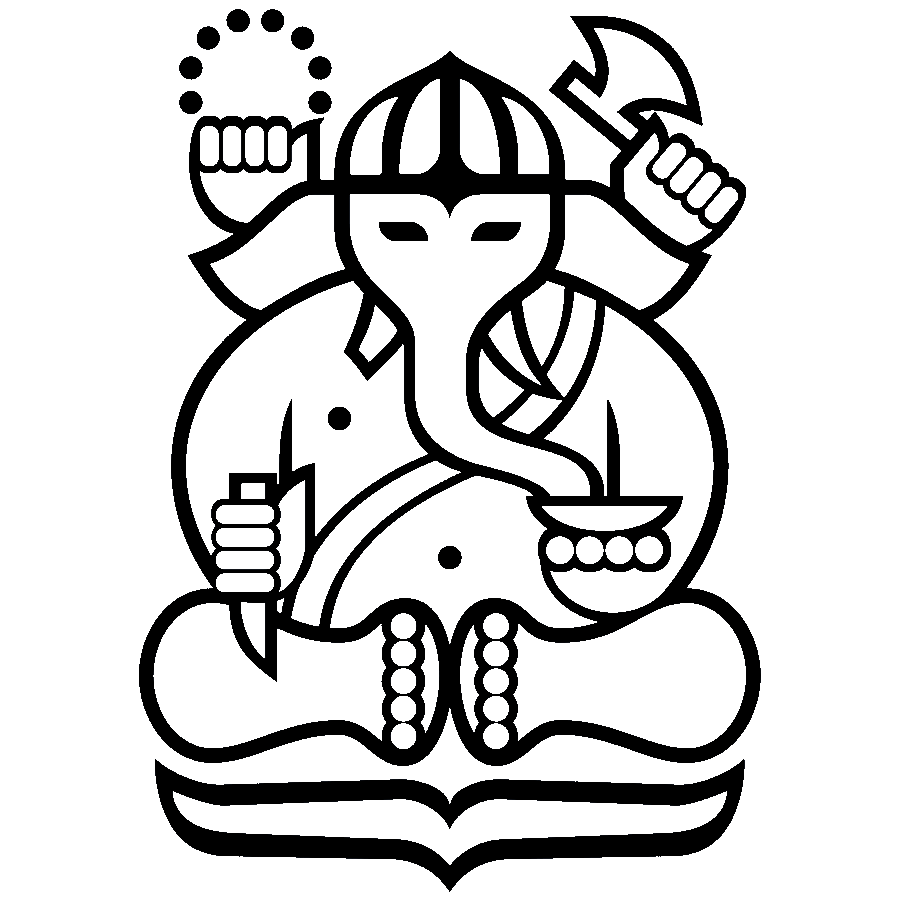
**MILESTONE 1 IF2240 BASIS DATA**

**SEMESTER II TAHUN 2022/2023**

**Pemodelan Entity-Relationship**

****

Disusun oleh :

13516055 - Nathaniel Evan Gunawan

13521057 - Hosea Nathanael Abetnego

13521105 - Haidar Hamda

13521127 - Marcel Ryan Antony

13521153 - Made Debby Almadea Putri

**SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA**

**INSTITUT TEKNOLOGI BANDUNG**

**2023**

# **DAFTAR ISI**

[**DAFTAR ISI 2**](#_d867cmubl5up)

[**DAFTAR GAMBAR 4**](#_3pop6s9u3y2g)

[**DAFTAR TABEL 5**](#_is453dgnu9f2)

[**Deskripsi Sistem 6**](#_ha68w79rzpou)

[**Relational Model 7**](#_drtmlnwyjev3)

[2.1. Model Relasional 7](#_mj4h7cdu8ulq)

[2.2. Deskripsi Functional Dependencies 7](#_upsszgw2if6n)

[2.2.1. username determines R & email determines R 7](#_fi91vgjnslmy)

[2.2.2. kodeMataKuliah, idModul determines noUrut 7](#_47t6mvi8dbrs)

[2.2.3. idModul determines nama 7](#_xycbbme8asvl)

[2.2.4. idMateri determines judul, link 8](#_ssyrzj5pjhad)

[2.2.5. idLatihanSoal determines judul, idModul 8](#_mko0o4nop91h)

[2.2.6. soal determines kunciJawaban 8](#_kbrk7jrdlwg4)

[2.2.7. soal, pilihan determines deskripsi 8](#_6kjt92f64a2i)

[2.2.8. singkatanFakultas determines namaFakultas 8](#_h44sm5ym0tj5)

[2.2.9. singkatanJurusan determines namaJurusan, singkatanFakultas 8](#_dvz0kgomjxui)

[2.2.10. kodeMataKuliah determines namaMataKuliah, singkatanJurusan 8](#_k7hr6qyy2twh)

[2.2.11. idMateri determines durasi 9](#_e5hy4d468w46)

[2.2.12. idMateri determines jumlahHalaman 9](#_hx92oxv7rngs)

[2.3. Bentuk Normal Setiap Relasi 9](#_62jrzjfrgkh)

[2.3.1. Relasi User 10](#_lkfqgccnwtd4)

[2.3.2. Relasi PesertaKuliah 10](#_rbce9rp5oim9)

[2.3.3. Relasi MataKuliah 10](#_8gh2ntpsip6l)

[2.3.4. Relasi Jurusan 10](#_3pv4h8p2cd)

[2.3.5. Relasi Fakultas 10](#_ohc8hlyscgkw)

[2.3.6. Relasi MatkulModul 10](#_fedvrk1my4w8)

[2.3.7. Relasi Modul 10](#_el37eg9rhb7i)

[2.3.8. Relasi MateriModul 11](#_onohpnd09n5g)

[2.3.9. Relasi Materi 11](#_syg0ztr2nwd4)

[2.3.10. Relasi VideoPembelajaran 11](#_vs24lfggwznt)

[2.3.11. Relasi DiktatPembelajaran 11](#_inibs0hvy6ab)

[2.3.12. Relasi RiwayatMateri 11](#_k00mmekg1y70)

[2.3.13. Relasi RiwayatLatihanSoal 11](#_nhmk67fglrim)

[2.3.14. Relasi LatihanSoal 11](#_8yzq9ftqaa02)

[2.3.15. Relasi Pertanyaan 11](#_41i7y6u8ze16)

[2.3.16. Relasi Jawaban 12](#_8sha6l24jho3)

[**Normalisasi Model Relasional 13**](#_spoxle1y9zep)

[3.1. Langkah Normalisasi 13](#_xdke32xyj6pm)

[3.1.1. Normalisasi Relasi Pertanyaan dan Jawaban 13](#_gud9k3lhoe91)

[3.2. Hasil Normalisasi 13](#_jbb8kurnptml)

[**Implementasi Business Rule 14**](#_ddyi3c3qko97)

[4.1. Query Penerapan Business Rule 14](#_tsghyw1fwiph)

[4.1.1. Penerapan Business Rule 1 14](#_bdqlwu8mdzvc)

[4.1.2. Penerapan Business Rule 2 15](#_9c2fw7csjdx0)

[4.1.3. Penerapan Business Rule 3 17](#_kn6izus0mc1k)

[4.2. Query Menunjukkan Penerapan Business Rule 18](#_r552ygrpw3h5)

[4.2.1. Hasil Penerapan Business Rule 2 18](#_1uklc5rwi6tu)

[4.2.2. Hasil Penerapan Business Rule 3 21](#_l9lzhtwkym08)

[**REFERENSI 22**](#_je0bi0nz56on)

[**LAMPIRAN 23**](#_b0hi3ph9l85)

[1. Deskripsi Wawancara 23](#_fjjogvaz6es9)

[Tabel Deskripsi Wawancara 23](#_epk6l6rl462g)

[2. Hasil Wawancara 23](#_f7qwj0ymth9)

[Tabel Hasil Wawancara 23](#_wlmwdq66fnqv)

# **DAFTAR GAMBAR**

# 

[Gambar 2.1.1](#_b80zhkvrpmho) Model Relasional 6

Gambar 3.2.1 Model Relasional Hasil Normalisasi9

[Gambar Dokumentasi](#_v5rq7tr6n5i3) 20

# 

# 

# 

# **DAFTAR TABEL**

# 

Tabel Normalitas Relasi 8

Tabel query penerapan business rule 1 10

Tabel query penerapan business rule 2 11

Tabel query penerapan business rule 3 13

[Tabel Deskripsi Wawancara](#_epk6l6rl462g) 19

[Tabel Hasil Wawancara](#_wlmwdq66fnqv) 19

# 

**BAB 1**

# **Deskripsi Sistem**

OpenCourseWare (OCW) adalah istilah platform pembelajaran daring yang dibuat oleh sebuah perguruan tinggi untuk menyediakan materi perkuliahan dari perguruan tinggi tersebut secara gratis. Materi perkuliahan dapat dibuat dalam bentuk slide perkuliahan, referensi tambahan, video pembelajaran, serta latihan soal. Materi ini dikelompokkan berdasar mata kuliah atau jurusan tertentu. Sekelompok mahasiswa ITB sedang mencoba untuk membuat website ITB OpenCourseWare dan meminta anda mendesain basis data untuk website ini. ITB OpenCourseWare adalah website yang menyediakan bahan ajar berupa slide perkuliahan, video, serta latihan soal dari kegiatan perkuliahan ITB untuk dapat diakses oleh berbagai kalangan.

Untuk mengakses materi, pengguna perlu melakukan pendaftaran terlebih dahulu untuk mengakses materi dan mengerjakan latihan soal. Untuk mendaftar, pengguna perlu mengisi nama lengkap, username, email, dan password. Kemudian pengguna diperlukan untuk memverifikasi email tersebut.

Setiap mata kuliah berada di bawah satu jurusan dan memiliki kode tertentu yang unik, dan tiap jurusan berada di bawah satu fakultas dengan singkatan yang unik untuk tiap fakultasnya. Setiap mata kuliah akan memiliki modul pembelajaran yang terurut mulai dari modul pertama. Modul pembelajaran tersebut akan berisi materi yang dapat dipelajari oleh pengguna. Untuk menyajikan materi, website ini akan memanfaatkan video pembelajaran dengan durasi tertentu dan berkas materi bacaan seperti slide dengan jumlah halaman tertentu yang akan diunggah pada website ini. Tiap materi dapat berisi beberapa media pembelajaran.

Selain penyajian materi, terdapat pula latihan soal yang dibuat dalam bentuk pilihan ganda dan dapat dikerjakan berkali-kali oleh pengguna. Hanya ada satu jawaban benar di antara pilihan lainnya. Latihan soal tidak harus selalu ada di tiap materi.

Website juga mencatat riwayat pengguna berupa tanggal dalam membaca materi ataupun mengerjakan latihan soal.

**BAB 2**

# **Relational Model**

## 2.1. Model Relasional

## 

Gambar 2.1.1 Model Relasional

## 2.2. Deskripsi *Functional Dependencies*

### 2.2.1. username *determines* R & email *determines* R

*Functional Dependency* berikut berlaku karena sistem dibuat sehingga sebuah *username* hanya dapat digunakan oleh satu *user* dan sebuah *email* hanya dapat digunakan pada sebuah akun. Oleh karena itu, *username* dan *email* dapat menentukan data lain dari sebuah user dan dinyatakan sebagai *functional dependency*.

### 2.2.2. kodeMataKuliah, idModul *determines* noUrut

Pada sistem yang akan diimplementasikan, sebuah modul pada mata kuliah tertentu hanya terdapat satu posisi urutan. Oleh karena itu, kodeMataKuliah dan idModul-nya dapat menentukan pada urutan keberapa modul tersebut dan menjadi sebuah *functional dependency*.

### 2.2.3. idModul *determines* nama

Sebuah modul pada sistem ini hanya memiliki sebuah nama, sehingga idModul tertentu dapat menentukan nama dari modul tersebut dan dinyatakan sebagai *functional dependency*.

### 2.2.4. idMateri *determines* judul, link

Sebuah materi pada sistem ini diasumsikan memiliki hanya satu judul dan *link*. Oleh karena itu, dinyatakan sebuah *functional dependency* yaitu idMateri menentukan judul dan *link*-nya.

### 2.2.5. idLatihanSoal *determines* judul, idModul

*Functional dependency* ini dinyatakan karena sebuah latihan soal hanya memiliki sebuah judul dan hanya berada pada satu modul saja. Ketika terdapat latihan soal yang memiliki konten yang sama, maka soal tersebut dinyatakan sebagai latihan soal baru dengan id baru. Oleh karena itu, *functional dependency* ini dapat dinyatakan pada sistem ini.

### 2.2.6. soal *determines* kunciJawaban

Soal pada sistem ini hanya memiliki sebuah kunci jawaban dan memiliki deskripsi yang selalu sama untuk sebuah latihan soal. Maka, dapat dinyatakan *functional dependency* yang menyatakan bahwa sebuah soal dapat menentukan kunci jawabannya.

### 2.2.7. soal, pilihan *determines* deskripsi

Sistem dapat memiliki soal yang sama dengan pilihan jawaban yang berbeda, sehingga untuk menentukan deskripsi pilihan pada suatu soal diperlukan kombinasi dari soal dan pilihannya. Oleh karena itu, dapat dinyatakan *functional dependency* bahwa soal dan pilihan digunakan untuk menentukan deskripsi dari pilihan tersebut.

### 2.2.8. singkatanFakultas *determines* namaFakultas

Pada sistem ini, diasumsikan bahwa sebuah fakultas memiliki sebuah nama. Maka dinyatakanlah *functional dependency* yang menyatakan bahwa singkatanFakultas menentukan nama dari fakultas tersebut.

### 2.2.9. singkatanJurusan *determines* namaJurusan, singkatanFakultas

*Functional dependency* ini diberlakukan karena digunakan asumsi bahwa sebuah jurusan hanya memiliki satu nama dan berada pada satu fakultas saja. Oleh karena itu, dinyatakan sebuah *functional dependency* yaitu singkatanJurusan menentukan nama jurusan dan fakultas dari jurusan tersebut.

### 2.2.10. kodeMataKuliah *determines* namaMataKuliah, singkatanJurusan

Pada *functional dependency* ini, diasumsikan bahwa sebuah mata kuliah hanya memiliki satu nama dan hanya berada pada sebuah jurusan saja. Maka, kodeMataKuliah dapat dipetakan terhadap hanya satu nama dan jurusan dan dianggap menjadi sebuah *functional dependency*.

### 2.2.11. idMateri *determines* durasi

Pada *functional dependency* ini, diasumsikan bahwa sebuah materi hanya memiliki satu videoPembelajaran. Maka, idMateri dapat dipetakan terhadap hanya satu durasi dan dianggap menjadi sebuah *functional dependency.*

### 2.2.12. idMateri *determines* jumlahHalaman

Pada *functional dependency* ini, diasumsikan bahwa sebuah materi hanya memiliki satu diktatPembelajaran. Maka, idMateri dapat dipetakan terhadap hanya satu jumlahHalaman dan dianggap menjadi sebuah *functional dependency.*

## 2.3. Bentuk Normal Setiap Relasi

**Tabel Normalitas Relasi**

| **Relasi** | **Bentuk Normal** |
| --- | --- |
| User | BCNF |
| pesertaKuliah | BCNF |
| MataKuliah | BCNF |
| Jurusan | BCNF |
| Fakultas | BCNF |
| MatkulModul | BCNF |
| Modul | BCNF |
| MateriModul | BCNF |
| Materi | BCNF |
| videoPembelajaran | BCNF |
| diktatPembelajaran | BCNF |
| RiwayatMateri | BCNF |
| RiwayatLatihanSoal | BCNF |
| LatihanSoal | BCNF |
| Pertanyaan | 1NF |
| Jawaban | 1NF |

### Relasi User

*Functional dependency* yang berlaku adalah {idMateri → R, email → R} dengan *candidate key* idMateri dan email. Relasi ini bersifat **BCNF** karena semua atribut kiri pada *functional dependency* merupakan *super key*.

### Relasi PesertaKuliah

Tidak ada *functional dependency* yang berlaku pada relasi ini, sedangkan *candidate key* yang berlaku adalah {username, kodeMataKuliah} yaitu semua atribut dalam relasi. Relasi ini bersifat **BCNF** karena tidak ada *functional dependency* yang menyalahi aturan BCNF.

### Relasi MataKuliah

*Functional dependency* yang berlaku adalah {kodeMataKuliah → namaMataKuliah singkatanJurusan} dengan *candidate key* kodeMataKuliah. Relasi ini bersifat **BCNF** karena semua atribut kiri pada *functional dependency* merupakan *super key*.

### Relasi Jurusan

*Functional dependency* yang berlaku adalah {singkatanJurusan → namaJurusan singkatanFakultas} dengan *candidate key* singkatanJurusan. Relasi ini bersifat **BCNF** karena semua atribut kiri pada *functional dependency* merupakan *super key*.

### Relasi Fakultas

*Functional dependency* yang berlaku adalah {singkatanFakultas → namaFakultas} dengan *candidate key* singkatanFakultas. Relasi ini bersifat **BCNF** karena semua atribut kiri pada *functional dependency* merupakan *super key*.

### Relasi MatkulModul

*Functional dependency* yang berlaku adalah {kodeMataKuliah idModul → noUrut} dengan *candidate key* {kodeMataKuliah, idModul}. Relasi ini bersifat **BCNF** karena semua atribut kiri pada *functional dependency* merupakan *super key*.

### Relasi Modul

*Functional dependency* yang berlaku adalah {idModul → nama} dengan *candidate key* idModul. Relasi ini bersifat **BCNF** karena semua atribut kiri pada *functional dependency* merupakan *super key*.

### Relasi MateriModul

Tidak ada *functional dependency* yang berlaku pada relasi ini, sedangkan *candidate key* yang berlaku adalah {idMateri, idModul} yaitu semua atribut dalam relasi. Relasi ini bersifat **BCNF** karena tidak ada *functional dependency* yang menyalahi aturan BCNF.

### Relasi Materi

*Functional dependency* yang berlaku adalah {idMateri → judul link} dengan *candidate key* idMateri. Relasi ini bersifat **BCNF** karena semua atribut kiri pada *functional dependency* merupakan *super key*.

### Relasi VideoPembelajaran

*Functional dependency* yang berlaku adalah {idMateri → durasi} dengan *candidate key* idMateri. Relasi ini bersifat **BCNF** karena semua atribut kiri pada *functional dependency* merupakan *super key*.

### Relasi DiktatPembelajaran

*Functional dependency* yang berlaku adalah {idMateri → jumlahHalaman} dengan *candidate key* idMateri. Relasi ini bersifat **BCNF** karena semua atribut kiri pada *functional dependency* merupakan *super key*.

### Relasi RiwayatMateri

Tidak ada *functional dependency* yang berlaku pada relasi ini, sedangkan *candidate key* yang berlaku adalah {username, idMateri, waktu} yaitu semua atribut dalam relasi. Relasi ini bersifat **BCNF** karena tidak ada *functional dependency* yang menyalahi aturan BCNF.

### Relasi RiwayatLatihanSoal

Tidak ada *functional dependency* yang berlaku pada relasi ini, sedangkan *candidate key* yang berlaku adalah {username, idLatihanSoal, waktu} yaitu semua atribut dalam relasi. Relasi ini bersifat **BCNF** karena tidak ada *functional dependency* yang menyalahi aturan BCNF.

### Relasi LatihanSoal

*Functional dependency* yang berlaku adalah {idLatihanSoal → judul idModul} dengan *candidate key* idLatihanSoal. Relasi ini bersifat **BCNF** karena semua atribut kiri pada *functional dependency* merupakan *super key*.

### Relasi Pertanyaan

*Functional dependency* yang berlaku adalah {soal → kunciJawaban} dengan *candidate key* {idLatihanSoal, soal}. Relasi ini tidak bersifat BCNF karena pada *functional dependency* soal → kunciJawaban, soal bukan merupakan *candidate key*. Relasi ini tidak bersifat 3NF karena pada *functional dependency* soal → kunciJawaban, {kunciJawaban} - {soal} = {kunciJawaban} bukan bagian dari *candidate key*. Relasi ini tidak bersifat 2NF karena pada *functional dependency* soal → kunciJawaban, soal merupakan bagian dari *candidate key* {idLatihanSoal, soal} sehingga terdapat *partial dependency*. Oleh karena itu, relasi ini bersifat **1NF**.

### Relasi Jawaban

*Functional dependency* yang berlaku adalah {soal pilihan → deskripsi} dengan *candidate key* {idLatihanSoal, soal, pilihan}. Relasi ini tidak bersifat BCNF karena pada *functional dependency* soal pilihan → deskripsi, soal pilihan bukan merupakan *candidate key*. Relasi ini tidak bersifat 3NF karena pada *functional dependency* soal pilihan → deskripsi, {deskripsi} - {soal pilihan} = {deskripsi} bukan bagian dari *candidate key*. Relasi ini tidak bersifat 2NF karena pada *functional dependency* soal pilihan → deskripsi, soal pilihan merupakan bagian dari *candidate key* {idLatihanSoal, soal, pilihan} sehingga terdapat *partial dependency*. Oleh karena itu, relasi ini bersifat **1NF**.

**BAB 3**

# **Normalisasi Model Relasional**

## Langkah Normalisasi

### Normalisasi Relasi Pertanyaan dan Jawaban

Relasi Pertanyaan dan Jawaban, yang sebelumnya memilki bentuk normal 1NF didekomposisi menjadi BCNF dengan algoritma dekomposisi BCNF. Ditemukan FD (soal → kunciJawaban) pada relasi Pertanyaan yang melanggar BCNF sehingga relasi tersebut perlu didekomposisi. Dekomposisi relasi Pertanyaan dilakukan dengan membuat tabel baru, yaitu relasi PertanyaanLatihanSoal yang memiliki atribut idLatihanSoal dan soal, dan relasi Pertanyaan berisi atribut soal dan kunciJawaban saja. Relasi yang terbentuk dari dekomposisi relasi pertanyaan kedua nya BCNF. Pada relasi Jawaban, ditemukan FD ({soal, pilihan} → deskripsi) yang melanggar BCNF sehingga relasi perlu didekomposisi. Dekomposisi pada relasi Jawaban dilakukan dengan memisahkan atribut yang menjadi bagian FD dengan atribut lainnya sehingga terbentuk relasi dengan atribut soal, pilihan, dan jawaban dan relasi dengan atribut idLatihanSoal dan soal. Akan tetapi, relasi yang berisikan atribut idLatihanSoal dan soal sudah ada, maka relasi tersebut dihilangkan agar tidak redundan. Proses normalisasi ini menghasilkan 3 relasi yang memenuhi syarat BCNF: relasi PertanyaanLatihanSoal, Pertanyaan dan Jawaban.

## 3.2. Hasil Normalisasi

## 

Gambar 3.2.1 Model Relasional Hasil Normalisasi

**BAB 4**

# **Implementasi Business Rule**

## Query Penerapan Business Rule

### Penerapan Business Rule 1

**Tabel query penerapan business rule 1**

| **QUERY**  -- Pertama akan dibuat terlebih dahulu view untuk menghitung jumlah akses materi  CREATE VIEW totalaksesmateri AS (SELECT idMateri, COUNT(username) as totalAkses  FROM riwayatmateri  GROUP BY (idMateri)  ORDER BY COUNT(username) DESC); |
| --- |
| **QUERY**  – Kedua akan dibuat view untuk menampilkan 50 materi teratas yang paling sering diakses oleh user  CREATE VIEW materi\_paling\_banyak\_diakses AS ( SELECT DISTINCT materi.idMateri, judul, tanggalMateri, link, totalAkses,  CASE WHEN materi.idMateri IN (SELECT idMateri FROM videoPembelajaran) THEN CONCAT("Video Pembelajaran Materi ", judul)  ELSE CONCAT("Bahan Bacaan Materi ", judul) END AS deskripsiMateri  FROM materi  INNER JOIN riwayatmateri ON materi.idMateri = riwayatmateri.idMateri  INNER JOIN totalaksesmateri ON materi.idMateri = totalaksesmateri.idMateri  ORDER BY totalAkses DESC  LIMIT 50;) |
| **SCREENSHOT** |

### Penerapan Business Rule 2

**Tabel query penerapan business rule 2**

| **--** Pertama-tama akan dibuat terlebih dahulu prosedur untuk mengupdate jumlahBahan dan jumlahModul pada tabel Modul dan tabel Matakuliah  **QUERY**  DELIMITER $$  CREATE PROCEDURE update\_jumlah\_bahan(  IN idModul INT  )  BEGIN  UPDATE modul M  SET jumlahBahan =  (SELECT COUNT(idMateri)  FROM materimodul  WHERE idModul = materimodul.idModul)  WHERE idModul = M.idModul;  END $$  DELIMITER $$  CREATE PROCEDURE update\_jumlah\_modul (  IN kodeMataKuliah VARCHAR(512)  )  BEGIN  UPDATE matakuliah MK  SET jumlahModul =  (SELECT COUNT(idModul)  FROM modulmatakuliah  WHERE kodeMataKuliah = modulmatakuliah.kodeMataKuliah)  WHERE kodeMataKuliah = MK.kodeMataKuliah;  END $$ |
| --- |
| **SCREENSHOT** |
| **--** Kedua, akan dibuat trigger untuk after\_insert dan after\_delete pada tabel materimodul  **QUERY**  DELIMITER $$  CREATE TRIGGER update\_jumlahBahan\_after\_insert  AFTER INSERT ON materimodul  FOR EACH ROW  BEGIN  CALL update\_jumlah\_bahan(NEW.idModul);  END $$  DELIMITER $$  CREATE TRIGGER update\_jumlahBahan\_after\_delete  AFTER DELETE ON materimodul  FOR EACH ROW  BEGIN  CALL update\_jumlah\_bahan(OLD.idModul);  END $$ |
| **SCREENSHOT** |
| **--** Ketiga, akan dibuat juga trigger untuk after\_insert dan after\_delete untuk tabel modulmatakuliah  **QUERY**  DELIMITER $$  CREATE TRIGGER update\_jumlahModul\_after\_insert  AFTER INSERT ON modulmatakuliah  FOR EACH ROW  BEGIN  CALL update\_jumlah\_modul(NEW.kodeMataKuliah);  END $$  DELIMITER $$  CREATE TRIGGER update\_jumlahModul\_after\_delete  AFTER DELETE ON modulmatakuliah  FOR EACH ROW  BEGIN  CALL update\_jumlah\_modul(OLD.kodeMataKuliah);  END $$ |
| **SCREENSHOT** |

### Penerapan Business Rule 3

**Tabel query penerapan business rule 3**

| **--** Pada business rule ini akan dibuat trigger yang akan mengecek apakah user telah terverifikasi dan mengecek apakah waktu akses saat itu adalah waktu maintenance bulanan  – NOTE : Untuk penerapan business rule ini diasumsikan maintenance dilakukan mulai pada bulan Januari sehingga maintenance akan selalu dilakukan pada bulan ganjil tanggal 6  **QUERY**  DELIMITER $$  CREATE TRIGGER riwayatmateri\_before\_insert  BEFORE INSERT ON riwayatmateri  FOR EACH ROW  BEGIN  DECLARE verified\_status VARCHAR(512);  SELECT isVerified INTO verified\_status FROM user WHERE username = NEW.username;  IF (verified\_status IS NULL OR verified\_status = 'FALSE') THEN  SIGNAL SQLSTATE '45000'  SET MESSAGE\_TEXT = 'User is not yet verified';  ELSEIF (DAYOFMONTH(NEW.waktu) = 6 AND MONTH(NEW.waktu) % 2 <> 0) THEN  SIGNAL SQLSTATE '45000'  SET MESSAGE\_TEXT = 'The system is in monthly maintenance cannot access the materi';  END IF;  END $$ |
| --- |
| **SCREENSHOT** |

## Query Menunjukkan Penerapan Business Rule

### Hasil Penerapan Business Rule 2

**Tabel screenshot hasil penerapan business rule 2**

| **-- CONTOH PENERAPAN TRIGGER DAN PROCEDURE PADA TABEL modulmatakuliah**  -- Berikut adalah totalModul pada matakuliah ‘WGS4634’ sebelum dimasukkan modul baru    -- Berikut adalah totalModul pada matakuliah ‘WGS4634’ setelah dimasukkan modul baru    -- Berikut adalah totalModul pada matakuliah ‘WGS4634’ apabila dilakukan delete terhadap suatu modul |
| --- |
| **-- CONTOH PENERAPAN TRIGGER DAN PROCEDURE PADA TABEL materimodul**  **--** Berikut adalah jumlahBahan pada modul VCD-1 sebelum dimasukkan materi baru    **--** Berikut adalah jumlahBahan pada modul VCD-1 setelah dimasukkan materi baru    **--** Berikut adalah jumlahBahan pada modul VCD-1 setelah dilakukan delete pada materi yang terdapat di modul VCD-1 |

### Hasil Penerapan Business Rule 3

**Tabel screenshot hasil penerapan business rule 3**

| **-- CONTOH PENERAPAN TRIGGER APABILA USER YANG INGIN MENGAKSES MATERI BELUM TERVERIFIKASI** |
| --- |
| **-- CONTOH PENERAPAN TRIGGER APABILA USER INGIN MENGAKSES MATERI SAAT SEDANG MAINTENANCE** |
| **-- CONTOH PENERAPAN TRIGGER APABILA USER YANG INGIN MENGAKSES MATERI TELAH TERVERIFIKASI DAN TIDAK DIAKSES PADA SAAT MAINTENANCE** |

### 

# **REFERENSI**

1. <https://edunex.itb.ac.id/>
2. [Tim PengajarIF2140 Semester I TA 2020/2021. Database Design using E-R Model](https://cdn-edunex.itb.ac.id/48686-Database-Parallel-Class/143662-Week-06-SQL/68095-Design-Database-Using-ER-Model/1677186974842_IF2240---SemII_2223---m06-2---Design-using-E-R-Model---part-1.pdf)
3. [Tim PengajarIF2140 Semester I TA 2020/2021. Database Design using E-R Model (p.2)](https://cdn-edunex.itb.ac.id/storages/files/1677646554405_IF2240---SemII_2223---m07-1---Design-using-E-R-Model---part-2.pdf)
4. [Tim PengajarIF2140 Semester I TA 2020/2021. Database Design using E-R Model (p.3)](https://cdn-edunex.itb.ac.id/storages/files/1678768255399_IF2240---SemII_2223---m09-1---Design-using-E-R-Model---part-3.pdf)
5. [Tim PengajarIF2140 Semester I TA 2020/2021. Reducing E-R Diagrams to Relational Schemas](https://cdn-edunex.itb.ac.id/38016-Database-Parallel-Class/71649-Week-10-Reducing-ER-Model-to-Relational-Schema-Relational-Database-Design/39059-Reducing-ER-Model-To-Relational-Schema/1647916372288_IF2240---Reducing-ER-Diagram-to-Relational-Model-slide-Edunex.pdf)
6. [Tim PengajarIF2140 Semester I TA 2020/2021. Relational Database Design part 2](https://cdn-edunex.itb.ac.id/38016-Database-Parallel-Class/71650-Week-11-Relational-Database-Design/39385-Relational-Database-Design-part-2/1648462375923_IF2240---Relational-Database-Design-part-2.pdf)
7. [Tim PengajarIF2140 Semester I TA 2020/2021. Relational Database Design part 3](https://cdn-edunex.itb.ac.id/38016-Database-Parallel-Class/71651-Week-12-Relational-Database-Design-Quiz-2/39835-Relational-Database-Design-part-3/1649387964094_IF2240---Relational-Database-Design-part-3.pdf)
8. [Tim PengajarIF2140 Semester I TA 2020/2021.Integrity Constraints](https://cdn-edunex.itb.ac.id/storages/files/1681181093855_IF2240---SemII_2223---m13-1-Integrity-Constraints.pdf)

# **LAMPIRAN**

Minutes of Meeting

## 1. Deskripsi Wawancara

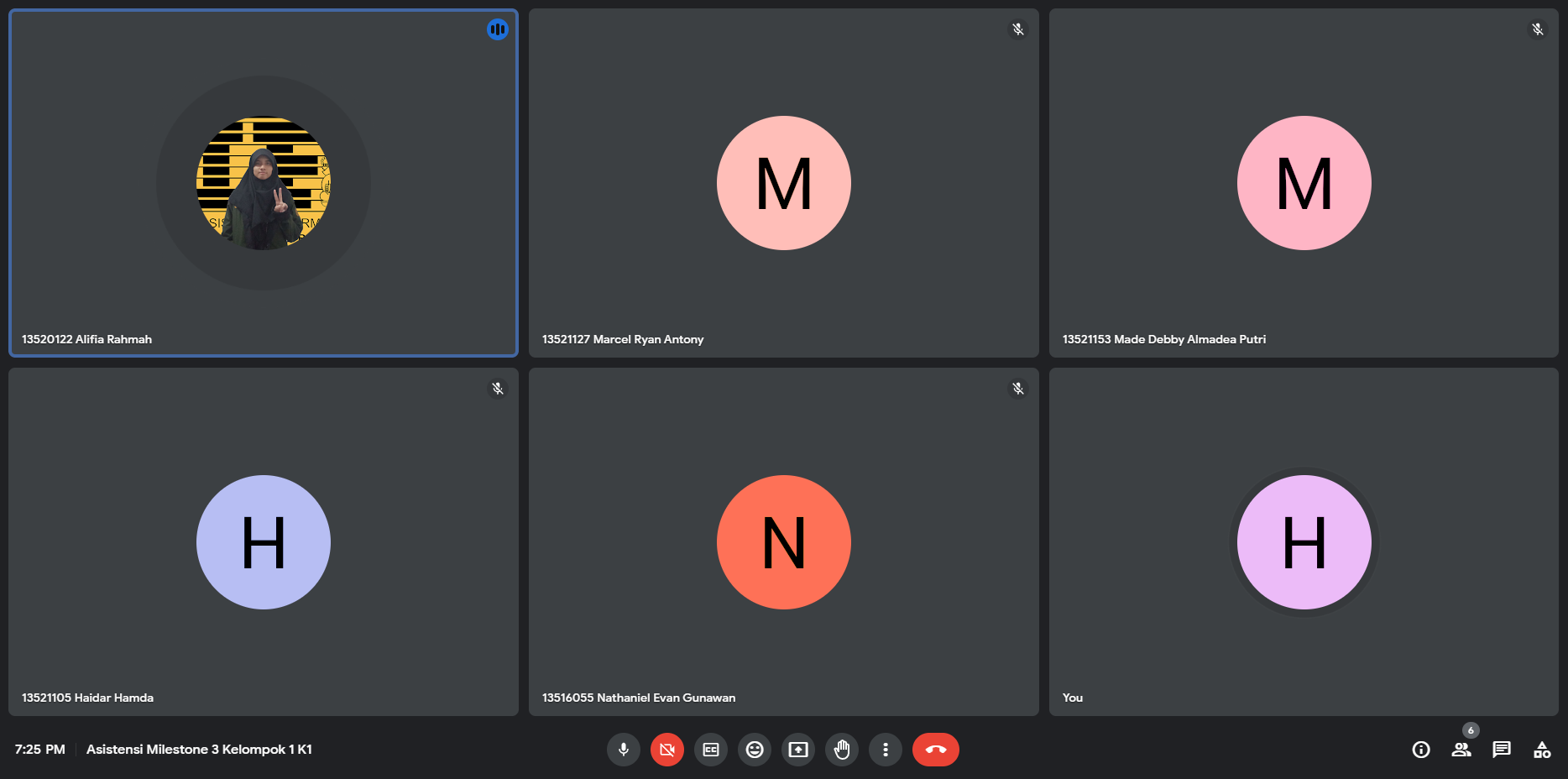
### Tabel Deskripsi Wawancara

| Nomor Kelompok | 1 |
| --- | --- |
| Anggota yang hadir saat wawancara | 1. Nathaniel Evan Gunawan (13516055) 2. Hosea Nathanael Abetnego (13521057) 3. Marcel Ryan Antony (13521127) 4. Haidar Hamda (13521105) 5. Made Debby Almadea Putri (13521153) |
| Tempat, tanggal | Google Meet, 28 April 2023 |
| Waktu | 19.00 |
| Nama Asisten | Alifia Rahmah |

## 2. Hasil Wawancara

### Tabel Hasil Wawancara

| Pertanyaan | Jawaban |
| --- | --- |
| Untuk business rule pertama, apabila kami tidak membuat atribut-atribut seperti tanggalMateri dan deskripsiMateri apakah perlu ditambahkan pada tabel? | Untuk atribut yang tidak ada pada tabel kalian sebenarnya ga perlu ditambahkan dan ga perlu ditampilkan aja. Cuma kalau mau ditambah atau ditampilkan juga boleh ga diwajibkan buat nambahin semua atribut yang ga ada juga |
| Untuk business rule ketiga itu berarti kita bakal bikin trigger ya kak untuk membatasi yang mengakses hanya bisa user yang sudah terverifikasi dan tidak diakses pada saat maintenance | Iya bakal bikin trigger yang ngecek status user dulu udah verified atau blm sama ngecek waktu aksesnya |
| Untuk business rule 3 untuk waktu mulainya boleh ditentukan sendiri ga ya kak? Kayak misalnya aku asumsiin mulai maintenance dari bulan Januari atau Februari | Untuk waktu mulai bulannya bebas mau dimulai dari Januari atau Februari yang jelas 2 bulan sekali tiap tanggal 6 |

Gambar Dokumentasi