

**IF2240 - Basis Data**

# **Tugas Besar Milestone II**

## **Pemodelan Entity-Relationship**



**Kelompok 4**

**Anggota:**

<b>Kelvin Rayhan Alkarim</b>	<b>13521005</b>
<b>Azmi Hasna Zahrani</b>	<b>13521006</b>
<b>Jason Rivalino</b>	<b>13521008</b>
<b>Muhammad Syauqi Janattan</b>	<b>13521014</b>
<b>Varraz Hazzandra Abrar</b>	<b>13521020</b>
<b>Raditya Naufal Abiyu</b>	<b>13521022</b>

**Program Studi Teknik Informatika**  
**Sekolah Teknik Elektro dan Informatika**  
**Institut Teknologi Bandung**  
**2023**



# DAFTAR ISI

<b>BAB I: DESKRIPSI SISTEM</b>	<b>4</b>
<b>BAB II: REVISI ENTITY-RELATIONSHIP MODEL</b>	<b>6</b>
<b>I. Gambar Diagram Entity-Relationship (Revisi)</b>	<b>6</b>
<b>II. Deskripsi Penjelasan Diagram Entity-Relationship (Revisi)</b>	<b>7</b>
<b>BAB III: RELATIONAL MODEL</b>	<b>13</b>
<b>I. Langkah Pemetaan Pengubahan Model ER menjadi Model Relasional</b>	<b>13</b>
<b>II. Bentuk akhir Model Relasional</b>	<b>18</b>
<b>BAB IV: INSIGHT DAN QUERY BASIS DATA</b>	<b>20</b>
<b>I. Daftar Insight dan Query Basis Data</b>	<b>20</b>
<b>LAMPIRAN</b>	<b>23</b>
<b>Lampiran MoM Wawancara</b>	<b>24</b>
<b>Daftar Pustaka</b>	<b>26</b>

# **BAB I**

## **DESKRIPSI SISTEM**

Deskripsi sistem yang didapat oleh kelompok kami adalah mengenai beasiswa yang disediakan oleh LPDP (Lembaga Pengelola Dana Pendidikan). Lembaga ini adalah lembaga yang bertanggung jawab atas pengelolaan dana pendidikan untuk mahasiswa Indonesia yang ingin melanjutkan studi S2 atau S3 di dalam negeri maupun luar negeri. Untuk mempermudah pengelolaan data dari pendaftar seleksi beasiswa LPDP, diperlukan sistem basis data untuk sistem pendaftaran seleksi LPDP

LPDP membuka pendaftaran bagi mahasiswa yang ingin mendapatkan beasiswa untuk melanjutkan studi S2 atau S3 setiap tahunnya. Dalam proses pendaftaran, mahasiswa diwajibkan untuk melengkapi berbagai macam informasi pribadi seperti nama lengkap, jenis kelamin, tempat dan tanggal lahir, alamat rumah, nomor telepon, dan alamat email. Selain itu, mahasiswa juga harus melampirkan berbagai macam dokumen seperti ijazah S1, transkrip nilai, dan surat rekomendasi dari dosen atau supervisor penelitian.

Mahasiswa yang telah mendaftar akan mengikuti beberapa tahapan seleksi, seperti tahapan seleksi administrasi seleksi wawancara, seleksi psikotes, dan seleksi kesehatan. Setiap tahap seleksi memiliki tanggal pelaksanaan yang berbeda-beda dan berbeda juga untuk seluruh mahasiswa, kecuali seleksi administrasi yang hanya dilakukan sekali di awal periode dan sama untuk seluruh mahasiswa. Tidak semua mahasiswa akan mengikuti seluruh rangkaian seleksi, kecuali tahap seleksi administrasi. Oleh karena itu, perlu dicatat *passing grade* untuk setiap tahap seleksi. Untuk tahap seleksi administrasi, mahasiswa yang lulus adalah mahasiswa yang dokumennya lengkap.

Setelah pendaftaran administrasi dinyatakan lengkap, calon penerima beasiswa akan mengikuti tahap seleksi wawancara, psikotes, dan kesehatan. Untuk tahap seleksi wawancara, setiap calon penerima beasiswa akan diwawancara dengan jenis tertentu antara *online* atau *offline* dengan sesi wawancara tertentu. Untuk setiap jenis wawancara dan sesi wawancara, terdapat panitia yang berperan sebagai pewawancara. Pewawancara LPDP juga memiliki jabatan berbeda-beda.

Pada tahap seleksi psikotes, setiap calon penerima beasiswa harus mengikuti tes psikotes. Tes psikotes dilaksanakan di sesi dan ruangan tertentu, tetapi seluruh sesi dan ruangan dilaksanakan di gedung yang sama. Untuk setiap sesi dan ruangan, terdapat seorang pengawas yang memiliki divisi yang berbeda-beda.

Pada tahap seleksi kesehatan, setiap calon penerima beasiswa harus melakukan pemeriksaan kesehatan di rumah sakit tertentu yang berada di kota tertentu. Untuk setiap rumah sakit, terdapat panitia yang merupakan dokter dengan spesialis tertentu.

Nilai yang didapat dari ketiga jenis tes tersebut akan dicatat untuk dievaluasi apakah mahasiswa yang bersangkutan lulus sebuah tahap seleksi. Mahasiswa dapat melanjutkan tahapan seleksi apabila sudah lolos pada tahapan seleksi yang sebelumnya. Untuk setiap panitia, baik itu pewawancara, dokter, maupun pengawas perlu dicatat nama lengkapnya.

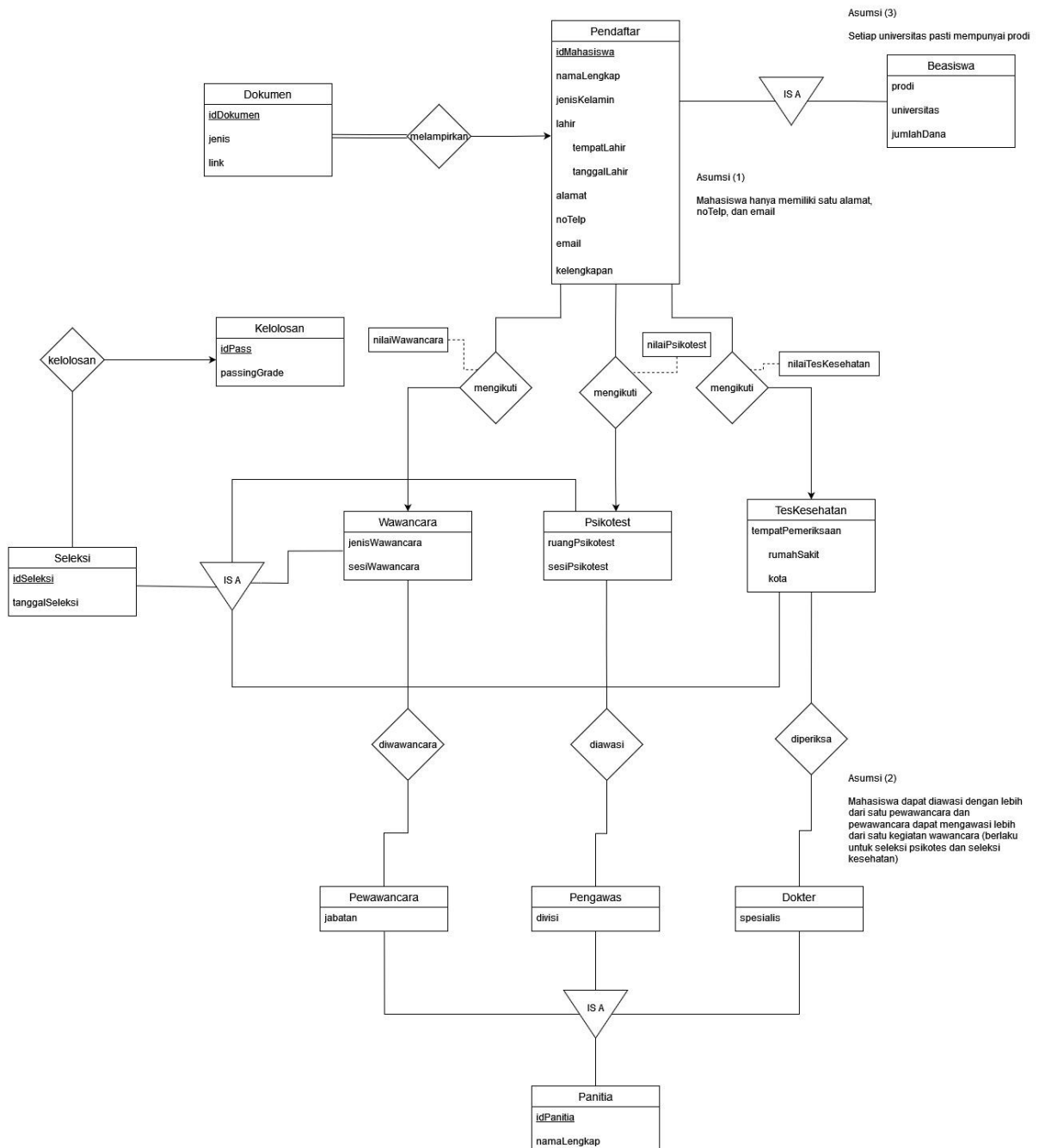
Setelah tahap seleksi selesai dilaksanakan, mahasiswa yang lulus seluruh tahap seleksi akan mendapatkan informasi mengenai pemberian beasiswa LPDP melalui email atau telepon. Selain itu, LPDP juga akan mencatat informasi mengenai profil mahasiswa yang mendapatkan beasiswa, seperti program studi yang diambil, universitas yang dituju, dan jumlah dana yang diberikan.

# BAB II

## REVISI ENTITY-RELATIONSHIP MODEL

### I. Gambar Diagram Entity-Relationship (Revisi)

Untuk revisi dari diagram *Entity-Relationship* yang telah dirancang oleh kelompok kami pada saat milestone pertama, gambar dari diagramnya adalah sebagai berikut:



## II. Deskripsi Penjelasan Diagram Entity-Relationship (Revisi)

Adapun penjelasan diagram *Entity-Relationship* yang telah direvisi adalah sebagai berikut:

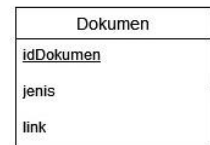
### a) Penjelasan Entitas (*Entity*)

Beberapa *entity* yang terdapat pada diagram *Entity-Relationship* yang telah kelompok kami revisi adalah sebagai berikut:

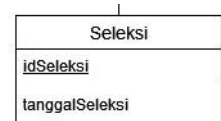
1. Pendaftar: entitas ini berfungsi untuk menyimpan data mahasiswa yang mendaftar beasiswa LPDP. Dalam entitas ini, beberapa atribut yang disimpan adalah idMahasiswa yang merupakan *primary key*, lalu atribut lainnya ada namaLengkap, jenisKelamin, lahir (terdiri atas tempatLahir dan tanggalLahir sebagai kompositnya), alamat, noTelp, dan email, dan kelengkapan dokumen saat seleksi pertama.



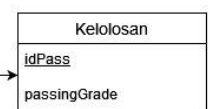
2. Dokumen: entitas ini berfungsi untuk menyimpan data dokumen apa saja yang dilampirkan oleh mahasiswa pendaftar beasiswa. Dalam entitas ini, beberapa atribut yang disimpan antara lain idDokumen sebagai *primary key* dan ada juga atribut jenis dan link dokumen.



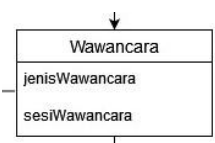
3. Seleksi: entitas ini berfungsi untuk menyimpan seleksi-seleksi yang dilakukan oleh mahasiswa. Dalam entitas ini, beberapa atribut yang disimpan antara lain idSeleksi sebagai *primary key* dan ada juga atribut tanggalSeleksi.



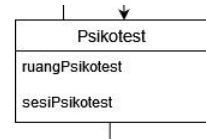
4. Kelolosan: entitas ini berfungsi untuk menyimpan daftar kelolosan mahasiswa dalam suatu seleksi. Dalam entitas ini, beberapa atribut yang disimpan antara lain idPass sebagai *primary key* dan ada atribut passingGrade



5. Wawancara: entitas ini berfungsi untuk menyimpan informasi terkait seleksi wawancara. Dalam entitas ini, beberapa atribut yang disimpan antara lain jenisWawancara dan sesiWawancara.



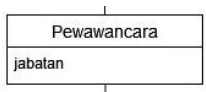
6. Psikotest: entitas ini berfungsi untuk menyimpan informasi terkait seleksi psikotest. Dalam entitas ini, beberapa atribut yang disimpan antara lain jenisPsikotest dan sesiPsikotest.



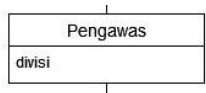
7. Kesehatan: entitas ini berfungsi untuk menyimpan informasi terkait seleksi kesehatan. Dalam entitas ini, atribut yang disimpan antara lain tempatPemeriksaan (terdiri atas rumahSakit dan kota sebagai kompositnya).



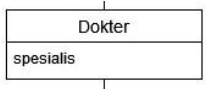
8. Pewawancara: entitas ini berfungsi untuk menyimpan informasi terkait pewawancara yang akan mewawancarai mahasiswa. Dalam entitas ini, atribut yang disimpan antara lain jabatan yang menyatakan jabatan dari pewawancara.



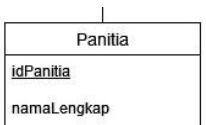
9. Pengawas: entitas ini berfungsi untuk menyimpan informasi terkait pengawas yang akan mengawasi mahasiswa. Dalam entitas ini, atribut yang disimpan antara lain divisi yang menyatakan divisi dari pengawas seleksi psikotes.



10. Dokter: entitas ini berfungsi untuk menyimpan informasi terkait dokter yang akan memeriksa mahasiswa. Dalam entitas ini, atribut yang disimpan antara lain spesialis yang menyatakan spesialis dari dokter yang memeriksa.



11. Panitia: entitas ini berfungsi untuk menyimpan informasi terkait panitia yang akan menguji mahasiswa dalam berbagai macam tahapan seleksi. Dalam entitas ini, beberapa atribut yang disimpan antara lain idPanitia sebagai *primary key*. Selain itu, terdapat atribut lainnya yaitu namaLengkap.



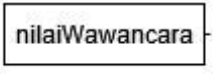
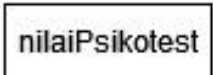



12. Beasiswa: entitas ini berfungsi untuk menyimpan mahasiswa yang mendapatkan beasiswa. Dalam entitas ini, beberapa atribut yang disimpan antara lain prodi, universitas, dan jumlahDana.



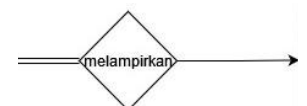
b) Penjelasan atribut-atribut lainnya

Beberapa atribut-atribut lainnya yang tidak termasuk ke dalam *entity* yang terdapat pada diagram *Entity-Relationship* yang telah kelompok kami revisi adalah sebagai berikut:

1. Atribut nilaiWawancara: atribut ini berfungsi untuk menyimpan nilai yang didapat mahasiswa pada saat seleksi wawancara. Nilai yang didapat ini kemudian akan dibandingkan dengan nilai passingGrade yang terdapat pada entitas kelolosan untuk menentukan apakah mahasiswa dapat lanjut ke tahapan seleksi selanjutnya atau tidak. 
2. Atribut nilaiPsikotest: atribut ini berfungsi untuk menyimpan nilai yang didapat mahasiswa pada saat seleksi psikotest. Nilai yang didapat ini kemudian akan dibandingkan dengan nilai passingGrade yang terdapat pada entitas kelolosan untuk menentukan apakah mahasiswa dapat lanjut ke tahapan seleksi selanjutnya atau tidak. 
3. Atribut nilaiTesKesehatan: atribut ini berfungsi untuk menyimpan nilai yang didapat mahasiswa pada saat seleksi kesehatan. Nilai yang didapat ini kemudian akan dibandingkan dengan nilai passingGrade yang terdapat pada entitas kelolosan untuk menentukan apakah mahasiswa akan mendapatkan beasiswa atau tidak. 

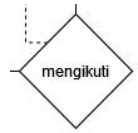
c) Penjelasan Relasi (*Relationship*)

1. Relasi melampirkan yang menghubungkan antara entitas Pendaftar dan Dokumen: Relasi ini bersifat *one-to-many* karena untuk satu pendaftar, harus melampirkan banyak dokumen. Untuk hubungan entitasnya, setiap pendaftar, harus melampirkan dokumen dan tidak semua dokumen akan dilampirkan oleh pendaftar.



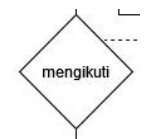
2. Relasi mengikuti yang menghubungkan antara entitas Pendaftar dan Wawancara:

Relasi ini bersifat *many-to-one* karena untuk banyak pendaftar, hanya akan mengikuti satu seleksi wawancara saja. Untuk hubungan entitasnya, tidak semua pendaftar akan mengikuti seleksi wawancara karena bisa saja ada pendaftar yang tidak lolos dalam seleksi administrasi.



3. Relasi mengikuti yang menghubungkan antara entitas Pendaftar dan Psikotest:

Relasi ini bersifat *many-to-one* karena untuk banyak pendaftar, hanya akan mengikuti satu seleksi psikotest saja. Untuk hubungan entitasnya, tidak semua pendaftar akan mengikuti seleksi psikotest karena bisa saja ada pendaftar yang tidak lolos dalam seleksi sebelumnya.



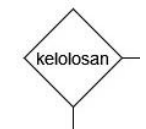
4. Relasi mengikuti yang menghubungkan antara entitas Pendaftar dan TesKesehatan:

Relasi ini bersifat *many-to-one* karena untuk banyak pendaftar, hanya akan mengikuti satu seleksi kesehatan saja. Untuk hubungan entitasnya, tidak semua pendaftar akan mengikuti seleksi kesehatan karena bisa saja ada pendaftar yang tidak lolos dalam seleksi sebelumnya.



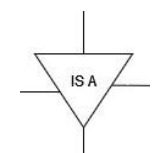
5. Relasi kelolosan yang menghubungkan antara entitas Seleksi dan Kelolosan:

Relasi ini bersifat *many-to-one* karena dalam dalam satu tahapan seleksi, terdapat banyak mahasiswa yang mengikuti seleksi, namun hanya ada satu passingGrade untuk menentukan kelolosan mahasiswa. Untuk hubungan entitasnya, tidak semua mahasiswa akan lolos mengikuti seleksi karena bisa saja ada nilai mahasiswa yang kurang dari nilai passingGrade.



6. Relasi spesialisasi antara Seleksi dan Wawancara, Psikotest, dan TesKesehatan:

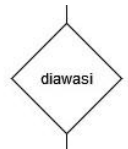
Relasi ini merupakan relasi spesialisasi karena dalam seleksi, terdapat berbagai macam jenis tahapan yang akan diselenggarakan, antara lain seleksi wawancara, seleksi psikotest, dan seleksi kesehatan. Oleh karena itu, hubungan relasi ini dinyatakan dengan spesialisasi.



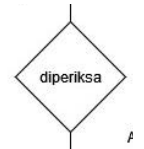
7. Relasi diwawancara yang menghubungkan antara entitas Wawancara dan Pewawancara: Relasi ini bersifat *many-to-many* karena untuk setiap mahasiswa, bisa diwawancarai dengan lebih dari satu pewawancara dan untuk pewawancara bisa mewawancarai lebih dari satu mahasiswa. Untuk hubungan entitasnya, tidak semua pewawancara akan mewawancarai seluruh mahasiswa yang mengikuti wawancara.



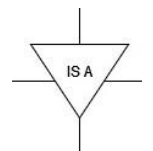
8. Relasi diawasi yang menghubungkan antara entitas Psikotest dan Pengawas: Relasi ini bersifat *many-to-many* karena untuk setiap mahasiswa, bisa diawasi dengan lebih dari satu pengawas dan untuk pengawas bisa mengawasi lebih dari satu mahasiswa. Untuk hubungan entitasnya, tidak semua pengawas akan mengawasi seluruh mahasiswa yang mengikuti psikotest.



9. Relasi diperiksa yang menghubungkan antara entitas TesKesehatan dengan Dokter: untuk relasi ini memiliki sifat *many-to-many* karena untuk setiap mahasiswa, bisa diperiksa dengan lebih dari satu dokter dengan spesialis yang berbeda-beda dan untuk dokter bisa memeriksa lebih dari satu mahasiswa. Untuk hubungan entitasnya, tidak semua dokter akan memeriksa seluruh mahasiswa yang mengikuti seleksi kesehatan.



10. Relasi spesialisasi antara Panitia dan Pewawancara, Pengawas, dan Dokter: Relasi ini merupakan relasi spesialisasi karena dalam panitia seleksi yang akan mengetes mahasiswa, terdapat berbagai macam jenis jenis panitia, antara lain pewawancara, pengawas, dan dokter. Oleh karena itu, hubungan relasi ini dinyatakan dengan spesialisasi.



Selain itu, terdapat beberapa asumsi yang digunakan dalam pembuatan diagram *Entity-Relationship* ini, antara lain:

1. Mahasiswa hanya memiliki satu alamat, noTelp, dan email.

2. Mahasiswa dapat diawasi dengan lebih dari satu pewawancara dan pewawancara dapat mengawasi lebih dari satu kegiatan wawancara (berlaku untuk seleksi psikotest dan seleksi kesehatan).
3. Setiap universitas pasti mempunyai prodi, sehingga untuk universitas dan prodi dijadikan satu entitas saja.

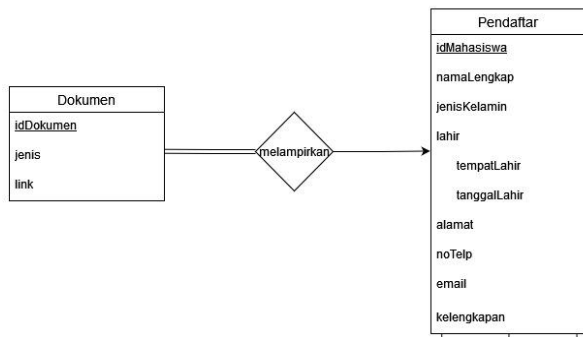
## BAB III

### RELATIONAL MODEL

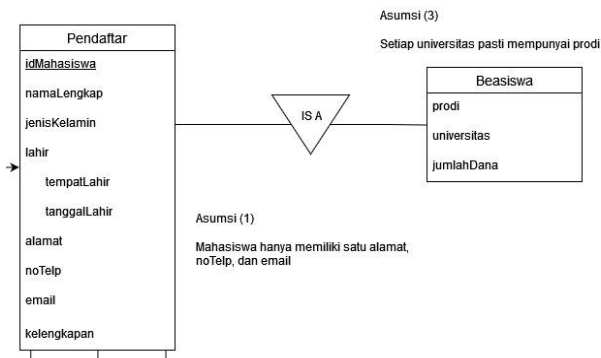
#### I. Langkah Pemetaan Pengubahan Model ER menjadi Model Relasional

Langkah-langkah pengubahan model ER yang sudah dibentuk menjadi model relasional adalah dengan menentukan *entity-relationship* mana saja yang dapat diubah ke dalam bentuk relasional. Pemetaan pengubahannya antara lain sebagai berikut:

1. Hubungan relasi antara entitas Pendaftar dan Dokumen. Pengubahan dilakukan dengan cara *Redudancy Schema*. Hasil dari pengubahan relasi ini akan membentuk relational data dengan tabel dan atributnya adalah sebagai berikut:
  - a) Pendaftar: { idMahasiswa, namaLengkap, jenisKelamin, tempatLahir, tanggalLahir, alamat, noTelp, email, kelengkapan }
  - b) Dokumen: { idDokumen, idMahasiswa, jenis, link }

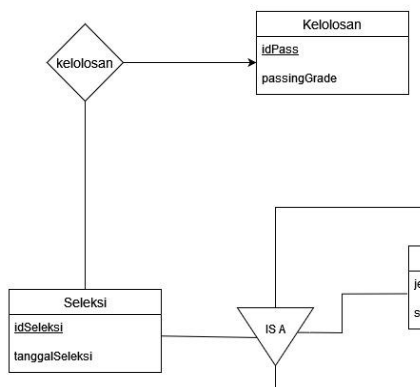


2. Hubungan relasi antara entitas Pendaftar dan Beasiswa. Pengubahan dilakukan dengan cara *Specialization Schema* cara pertama. Hasil dari pengubahan relasi ini akan membentuk relational data dengan tabel dan atributnya adalah sebagai berikut:
  - a) Pendaftar: { idMahasiswa, namaLengkap, jenisKelamin, tempatLahir, tanggalLahir, alamat, noTelp, email, kelengkapan }
  - b) Beasiswa: { idMahasiswa, prodi, universitas, jumlahDana }



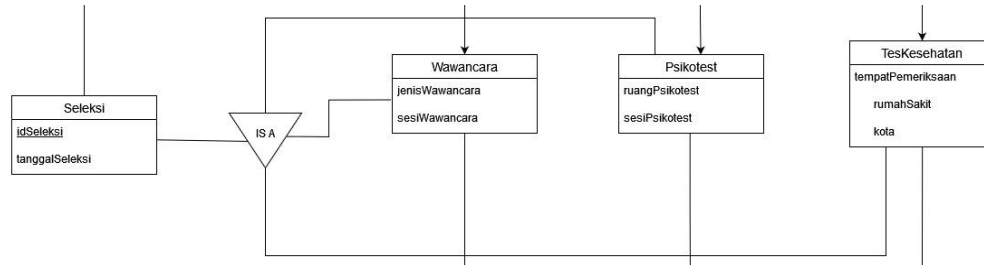
3. Hubungan relasi antara entitas Seleksi dan Kelolosan. Pengubahan dilakukan dengan cara *Redudancy Schema*. Hasil dari pengubahan relasi ini akan membentuk relational data dengan tabel dan atributnya adalah sebagai berikut:

- a) Seleksi: { idSeleksi, idPass, tanggalSeleksi }
- b) Kelolosan: { idPass, jenisSeleksi, passingGrade }



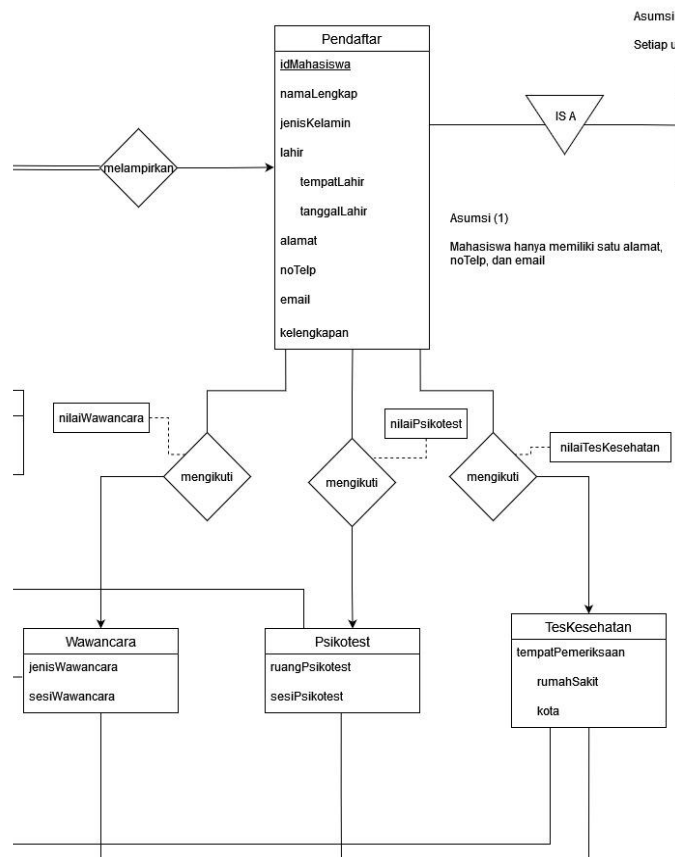
4. Hubungan relasi antara entitas Seleksi dan Wawancara, Psikotest, dan TesKesehatan. Pengubahan dilakukan dengan cara *Specialization Schema* cara pertama. Hasil dari pengubahan relasi ini akan membentuk relational data dengan tabel dan atributnya adalah sebagai berikut:

- a) Seleksi: { idSeleksi, idPass, tanggalSeleksi }
- b) Wawancara: { idWawancara, jenisWawancara, sesiWawancara }
- c) Psikotes: { idPsikotest, ruangPsikotest, sesiPsikotest }
- d) TesKesehatan: { idTesKesehatan, rumahSakit, kota }



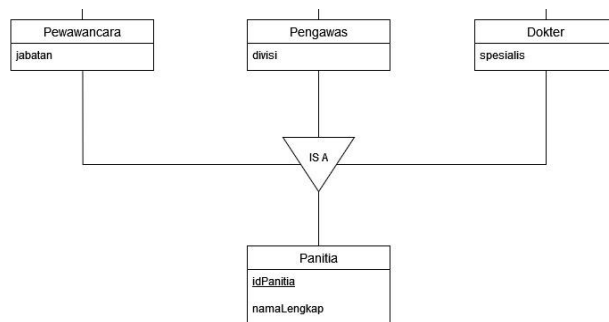
5. Hubungan relasi antara Pendaftar dan Wawancara, pendaftar dan psikotest, serta pendaftar dan TesKesehatan. Pengubahan dilakukan dengan cara *Redudancy Schema*. Hasil dari pengubahan relasi ini akan membentuk relational data dengan tabel dan atributnya adalah sebagai berikut:

- a) Pendaftar: { idMahasiswa, idWawancara, idPsikotest, idTesKesehatan, namaLengkap, jenisKelamin, tempatLahir, tanggalLahir, alamat, noTelp, email, kelengkapan, nilaiWawancara, nilaiPsikotest, nilaiTesKesehatan }
- b) Wawancara: { idWawancara, jenisWawancara, sesiWawancara }
- c) Psikotest: { idPsikotest, ruangPsikotest, sesiPsikotest }
- d) TesKesehatan: { idTesKesehatan, rumahSakit, kota }



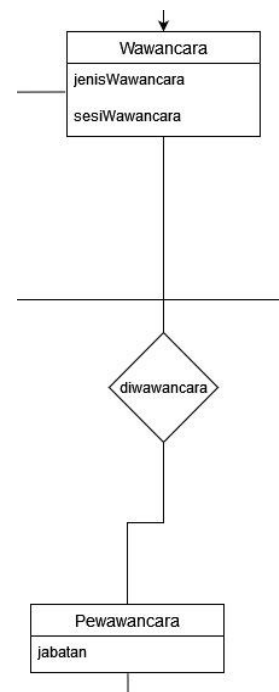
6. Hubungan relasi antara Panitia dan Pewawancara, Pengawas, dan Dokter. Pengubahan dilakukan dengan cara *Specialization Schema* cara pertama. Hasil dari pengubahan relasi ini akan membentuk relational data dengan tabel dan atributnya adalah sebagai berikut:

- a) Panitia: { idPanitia, namaLengkap }
- b) Pewawancara: { idPanitia, jabatan }
- c) Pengawas: { idPanitia, divisi }
- d) Dokter: { idPanitia, spesialis }



7. Hubungan relasi antara Wawancara dan Pewawancara. Pengubahan dilakukan dengan cara *Relationship Sets*. Hasil dari pengubahan relasi ini akan membentuk relational data dengan tabel dan atributnya adalah sebagai berikut:

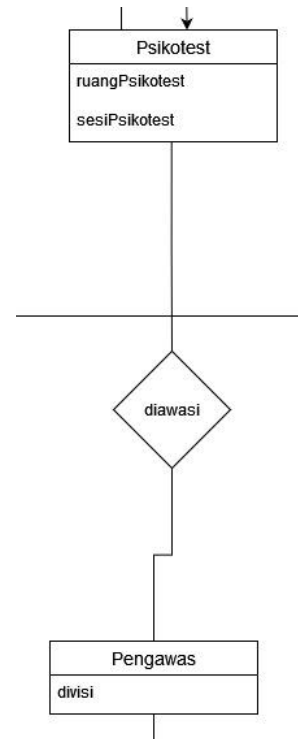
- a) Wawancara: { idWawancara, jenisWawancara, sesiWawancara }
- b) Diwawancara: { idWawancara, idPanitia }
- c) Pewawancara: { idPanitia, jabatan }





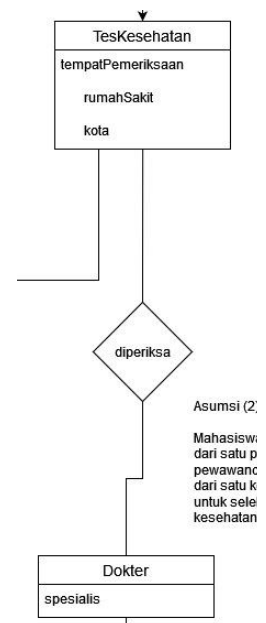
8. Hubungan relasi antara Psikotes dan Pengawas. Pengubahan dilakukan dengan cara *Relationship Sets*. Hasil dari pengubahan relasi ini akan membentuk relational data dengan tabel dan atributnya adalah sebagai berikut:

- a) Psikotes: { idPsikotest, ruangPsikotest, sesiPsikotest }
- b) Diawasi: { idPsikotes, idPanitia }
- c) Pengawas: { idPanitia, divisi }



9. Hubungan relasi antara Kesehatan dan Dokter. Pengubahan dilakukan dengan cara *Relationship Sets*. Hasil dari pengubahan relasi ini akan membentuk relational data dengan tabel dan atributnya adalah sebagai berikut:

- a) Kesehatan: { idTesKesehatan, rumahSakit, kota }
- b) Diperiksa: { idTesKesehatan, idPanitia }
- c) Dokter: { idPanitia, spesialis }



## II. Bentuk akhir Model Relasional

Bentuk akhir yang didapat dari model relasional yang telah dibuat adalah sebagai berikut:

Tabel dan Atribut:

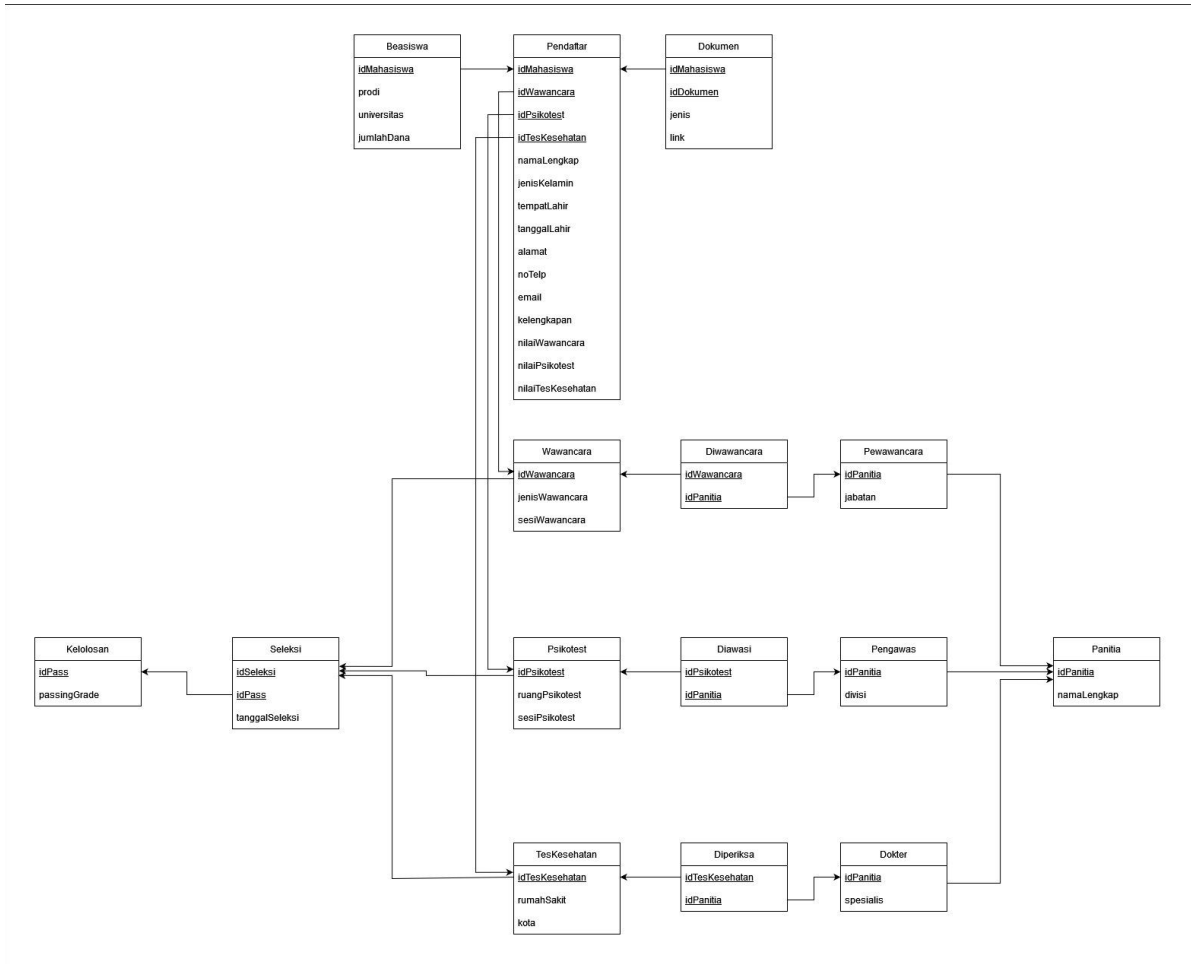
- Pendaftar: { idMahasiswa, idWawancara, idPsikotest, idTesKesehatan, namaLengkap, jenisKelamin, tempatLahir, tanggalLahir, alamat, noTelp, email, kelengkapan, nilaiWawancara, nilaiPsikotest, nilaiTesKesehatan }
- Dokumen: { idMahasiswa, idDokumen, jenis, link }
- Beasiswa: { idMahasiswa, prodi, universitas, jumlahDana }
- Seleksi: { idSeleksi, idPass, tanggalSeleksi }
- Kelolosan: { idPass, passingGrade }
- Panitia: { idPanitia, namaLengkap }
- Wawancara: { idWawancara, jenisWawancara, sesiWawancara }
- Diwawancara: { idWawancara, idPanitia }
- Pewawancara: { idPanitia, jabatan }
- Psikotest: { idPsikotest, ruangPsikotest, sesiPsikotest }
- Diawasi: { idPsikotest, idPanitia }
- Pengawas: { idPanitia, divisi }
- Kesehatan: { idTesKesehatan, rumahSakit, kota }
- Diperiksa: { idTesKesehatan, idPanitia }
- Dokter: { idPanitia, spesialis }

Foreign Key:

- Dokumen (idMahasiswa) -> Pendaftar (idMahasiswa)
- Beasiswa (idMahasiswa) -> Pendaftar (idMahasiswa)
- Seleksi (idPass) -> Kelolosan (idPass)
- Pendaftar (idWawancara) -> Wawancara (idWawancara)
- Pendaftar (idPsikotest) -> Psikotest (idPsikotest)
- Pendaftar (idTesKesehatan) -> TesKesehatan (idTesKesehatan)
- Wawancara (idWawancara) -> Seleksi (idSeleksi)
- Psikotest (idPsikotest) -> Seleksi (idSeleksi)
- TesKesehatan (idTesKesehatan) -> Seleksi (idSeleksi)
- Diwawancara (idWawancara) -> Wawancara (idWawancara)
- Diawasi (idPsikotest) -> Psikotest (idPsikotest)
- Diperiksa (idTesKesehatan) -> TesKesehatan (idTesKesehatan)
- Diwawancara (idTesKesehatan) -> Pewawancara (idPanitia)
- Diawasi (idPanitia) -> Pengawas (idPanitia)
- Diperiksa (idPanitia) -> Dokter (idPanitia)
- Pewawancara (idPanitia) -> Panitia (idPanitia)
- Pengawas (idPanitia) -> Panitia (idPanitia)

- Dokter (idPanitia) -> Panitia (idPanitia)

Gambar skema relasional data yang telah dibuat sebelumnya adalah sebagai berikut:



# BAB IV

## INSIGHT DAN QUERY BASIS DATA

### I. Daftar *Insight* dan *Query* Basis Data

#### Contoh *Insight*

1. Menampilkan nama mahasiswa yang memperoleh beasiswa

Query:

```
SELECT daftar.idMahasiswa, daftar.namaLengkap, bea.universitas, bea.prodi,
bea.jumlahDana as "jumlahDana ($USD)"
FROM pendaftar as daftar, beasiswa as bea
WHERE daftar.idMahasiswa = bea.idMahasiswa
ORDER BY idMahasiswa;
```

Screenshot hasil query:

```
MariaDB [tubesbasdat]> SELECT daftar.idMahasiswa, daftar.namaLengkap, bea.universitas, bea.prodi, bea.jumlahDana as "jumlahDana ($USD)"
-> FROM pendaftar as daftar, beasiswa as bea
-> WHERE daftar.idMahasiswa = bea.idMahasiswa
-> ORDER BY idMahasiswa;
```

idMahasiswa	namaLengkap	universitas	prodi	jumlahDana (\$USD)
3	Marjory Brisley	China University of Geosciences Wuhan	Geology	1312
5	Tommy Meffen	World University of Bangladesh	Communication	1025
19	Coop Celli	University of the West of England, Bristol	Computer Science	1316
20	Agnesse Ousley	Rocky Mountain College of Art and Design	Fashion Design	1238
25	Welby Fielder	Academy of Arts	Graphic Design	1090
26	Shaine Capp	Alahgaff University	Architecture	1408
36	Talyah Fante	Free University Institute "Carlo Cattaneo"	International Law	1125
44	Inigo Hambrook	Universidade Federal da Grande Dourados	Mechanical Engineering	1223
69	Hayes Bau	Université Ibn Zohr Agadir	Pharmacy	1488
88	Anjela Bosdet	Institute of Management Sciences, Lahore (IMS)	Management	1016

```
10 rows in set (0.007 sec)
```

2. Menampilkan semua pendaftar yang lulus seleksi psikotest

Query:

```
SELECT idMahasiswa, namaLengkap
FROM pendaftar
WHERE nilaiPsikotest > (SELECT passingGrade
                        FROM kelolosan
                        WHERE idpass = 2);
```

Screenshot hasil query:

```
MariaDB [tubesbasdat]> SELECT idMahasiswa, namaLengkap
-> FROM pendaftar
-> WHERE nilaiPsikotest > (SELECT passingGrade FROM kelolosan WHERE idpass = 2);
```

idMahasiswa	namaLengkap
3	Marjory Brisley
4	Trisha Gregon
5	Tommy Meffen
19	Coop Celli
20	Agnesse Ousley
21	Rebekah Hutten
25	Welby Fielder
26	Shaine Capp
27	Vilhelmina Giametti
28	Tanya Haquard
29	Cass Portal
36	Talyah Fante
39	Caryn Meader
43	Kettie Kleinsmuntz
44	Inigo Hambrook
53	Carrol Christofides
56	Chelsy Rizzo
60	Morton Davern
69	Hayes Bau
70	Ned Alderwick
88	Anjela Bosdet

21 rows in set (0.001 sec)

3. Menampilkan semua pendaftar yang diperiksa di rumah sakit Borrromeus

Query:

```
SELECT idMahasiswa, namaLengkap
FROM pendaftar NATURAL JOIN tesKesehatan
WHERE rumahSakit = "Borrromeus";
```

Screenshot hasil query:

```
MariaDB [tubesbasdat]> SELECT idMahasiswa, namaLengkap
-> FROM pendaftar NATURAL JOIN tesKesehatan
-> WHERE rumahSakit = "Borrromeus";
```

idMahasiswa	namaLengkap
43	Kettie Kleinsmuntz
5	Tommy Meffen
25	Welby Fielder
3	Marjory Brisley
19	Coop Celli
70	Ned Alderwick

6 rows in set (0.001 sec)

#### 4. Menampilkan 3 mahasiswa dengan nilai seleksi kesehatan tertinggi

Query:

```
SELECT idMahasiswa, namalengkap, nilaiTesKesehatan
FROM pendaftar
ORDER BY nilaiTesKesehatan desc
LIMIT 3;
```

Screenshot hasil query:

```
MariaDB [tubesbasdat]> SELECT idMahasiswa, namalengkap, nilaiTesKesehatan
-> FROM pendaftar
-> ORDER BY nilaiTesKesehatan desc
-> LIMIT 3;
+-----+-----+-----+
| idMahasiswa | namalengkap | nilaiTesKesehatan |
+-----+-----+-----+
| 25 | Welby Fielder | 368.10 |
| 20 | Agnesse Ousley | 299.82 |
| 19 | Coop Celli | 276.54 |
+-----+-----+-----+
3 rows in set (0.000 sec)
```

#### 5. Menampilkan mahasiswa yang memiliki 2 dokumen

Query:

```
SELECT idMahasiswa, namaLengkap, count(idDokumen)
FROM pendaftar NATURAL JOIN dokumen
GROUP BY idMahasiswa
HAVING count(idDokumen)=2;
```

Screenshot hasil query:

```
MariaDB [tubesbasdat]> SELECT idMahasiswa, namaLengkap, count(idDokumen)
-> FROM pendaftar NATURAL JOIN dokumen
-> GROUP BY idMahasiswa
-> HAVING count(idDokumen)=2;
+-----+-----+-----+
| idMahasiswa | namaLengkap | count(idDokumen) |
+-----+-----+-----+
| 50 | Pepi Turfitt | 2 |
| 51 | Gilly Chaffey | 2 |
| 54 | Coleman Brazener | 2 |
| 59 | Jelene Timpany | 2 |
| 61 | Marlow Mightly | 2 |
| 63 | Ernestine Vlahos | 2 |
| 64 | Ilsa Peiro | 2 |
| 67 | Deana Mafham | 2 |
| 76 | Nadya Abramowitch | 2 |
| 77 | Edgardo Elderton | 2 |
| 78 | Crawford Klimczak | 2 |
| 86 | Maure Spyby | 2 |
| 89 | Lazar Clampin | 2 |
| 90 | Paolo Chin | 2 |
| 93 | Saw Caslake | 2 |
| 94 | Heida Sabie | 2 |
| 95 | Pietrek Faire | 2 |
| 97 | Christabel Bigrigg | 2 |
| 100 | Sara MacCague | 2 |
| 102 | Brenna Kyd | 2 |
| 103 | Ferrel Vautrety | 2 |
| 107 | Salomon Esmead | 2 |
| 108 | Malynda Grissett | 2 |
+-----+-----+-----+
23 rows in set (0.002 sec)
```

# LAMPIRAN

## Lampiran MoM Wawancara

# Minutes of Meeting

### 1. Deskripsi Wawancara

Nomor Kelompok	04
Anggota yang hadir saat wawancara	1. Kelvin Rayhan Alkarim (13521005) 2. Azmi Hasna Zahrani (13521006) 3. Jason Rivalino (13521008) 4. Muhhamad Syauqi Janattan (13521014) 5. Varraz Hazzandra Abrar (13521020) 6. Raditya Naufal Abiyu (13521022)
Tempat, tanggal	Google Meet, 4 April 2023
Waktu	15.00
Nama Asisten	Addin Nabilal Huda

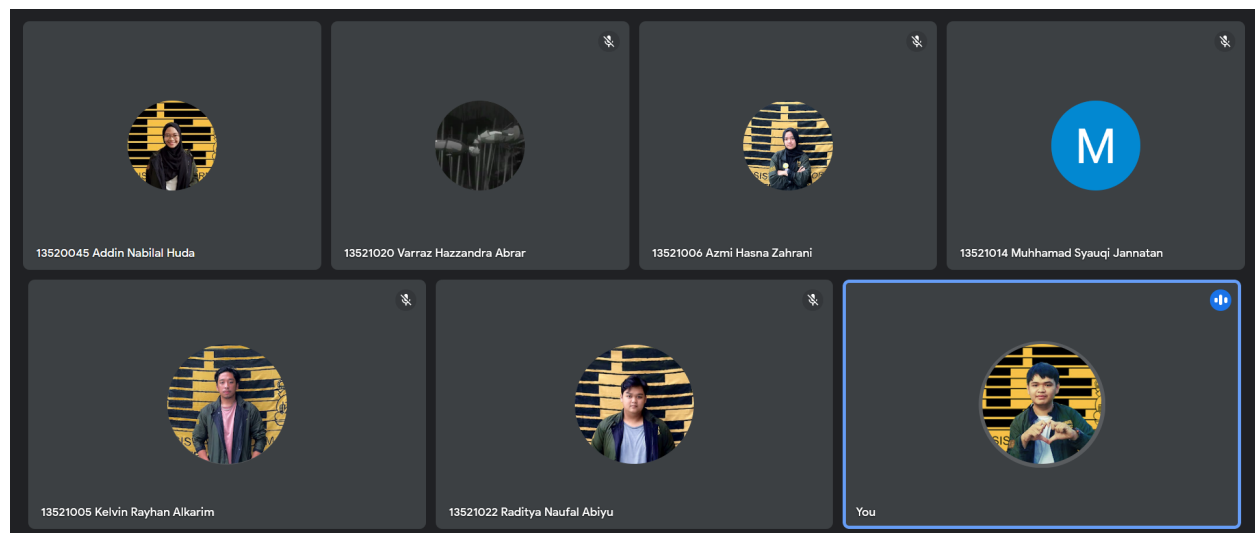
### 2. Hasil Wawancara

No.	Pertanyaan	Jawaban
1.	Kenapa simpen tanggal administrasi di ER Design??	Itu untuk deadline pengumpulan berkas dokumennya gitu Kak  Masukkan: seharusnya tidak perlu disimpan soalnya sama semua juga untuk tanggal batas seleksi pengumpulannya  Kelengkapan masuk relasi pendaftar
2	Relasi ada yang lolos bisa dijelaskan lebih lanjut gaa??	Ada sistem gugur, jadi kalo lolos admin baru bisa wawancara, dll  Masukkan: tidak semua yang di deskripsi sistem diubah ke ERD, contoh seperti tahapan seleksi, kalo diubah nanti ke bentuk SQL bisa agak aneh, jadi relasi lolos gak perlu disimpan



		Cara nentuin kelolosan, ada passing grade, jangan disimpan di situ (relasi lolos), kita bisa ambil nilai dan passing grade, simpen passing grade dalam fungsi gt (dalam database katanya bisa simpan isLolos cuma kelemahannya bisa ada inkonsistensi data), bisa juga bikin relasi baru yang nyimpen generalisasi
3	Beasiswa kenapa weak entity?	<p>Karena untuk setiap mahasiswa yang mendapat beasiswa, pasti ada mahasiswa yang mendaftar</p> <p>Masukkan: Tidak salah cuma bisa pake pake alternatif lain aja, seperti spesialisasi</p>

### 3. Dokumentasi



# Daftar Pustaka

- [1] Tim Pengajar IF2140/IF2240 (2020). *Database Design using E-R Model* [Powerpoint Slides]. Available: [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](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)
- [2] Tim Pengajar IF2140 Semester I (2020). *Database Design using E-R Model (p.2)* [PowerPoint Slides]. Available: [https://cdn-edunex.itb.ac.id/storages/files/1677646554405\\_IF2240---SemII\\_2223---m07-1---Design-using-E-R-Model---part-2.pdf](https://cdn-edunex.itb.ac.id/storages/files/1677646554405_IF2240---SemII_2223---m07-1---Design-using-E-R-Model---part-2.pdf)
- [3] Tim Pengajar IF2140 Semester I (2020). *Database Design using E-R Model (p.3)* [PowerPoint Slides]. Available: [https://cdn-edunex.itb.ac.id/storages/files/1678768255399\\_IF2240---SemII\\_2223---m09-1---Design-using-E-R-Model---part-3.pdf](https://cdn-edunex.itb.ac.id/storages/files/1678768255399_IF2240---SemII_2223---m09-1---Design-using-E-R-Model---part-3.pdf)
- [4] Silberschatz, A., Korth, H. F., & Sudarshan, S. (2019). *Database System Concepts*, 7th Edition. McGraw Hill Education.