



Application of Access Rights in Academic Database Design to Improve Data Security

Penerapan Hak Akses pada Perancangan Database Akademik untuk Meningkatkan Keamanan Data

Joko Triyono

Program Studi Rekayasa Sistem Komputer, Fakultas Teknologi dan Bisnis,
Institut Sains dan Teknologi AKPRIND Yogyakarta, Indonesia

E-Mail: jack@akprind.ac.id

*Received Apr 12th 2023; Revised May 20th 2023; Accepted Jun 18th 2023
Corresponding Author: Joko Triyono*

Abstract

Complexity in developing applications in the digital era is increasingly being demanded up to the data security stage, the term "data is oil" is something that really deserves attention, so that data security is something that cannot be avoided. In addition to security in terms of application interfaces, application placement, security settings on networking systems, also setting access rights to databases, tables, fields and even routines in RDBMS. In this research, the design, implementation and testing of an academic RDBMS has been carried out, the result is that the design with proper and correct application of access rights will increase RDBMS security from unwanted activities, so that the information value of the RDBMS is trustworthy. With this model, when developing applications it will be easier because a lot of data access security has been carried out by the RDBMS itself, developing applications only using the account that has been provided which is adjusted to the user group that uses it.

Keyword: Access Rights, Data, Implementation, RDBMS, Security

Abstrak

Kompleksitas dalam pengembangan aplikasi pada era digital semakin dituntut sampai tahap keamanan data, istilah "data is oil" menjadi hal yang sangat patut diperhatikan, sehingga keamanan akan data menjadi sesuatu yang tidak bisa dihindarkan. Selain pengamanan dari segi interface aplikasi, penempatan aplikasi, pengaturan keamanan pada sistem networking, juga pengaturan hak akses pada database, tabel, field-field bahkan sampai routine-routine dalam RDBMS. Pada penelitian ini telah dilakukan perancangan, implementasi dan pengujian pada RDBMS akademik diperoleh hasil bahwa rancangan dengan penerapan hak akses secara tepat dan benar akan meningkatkan keamanan RDBMS dari kegiatan-kegiatan yang tidak diinginkan, sehingga nilai informasi dari RDBMS tersebut layak untuk dipercaya. Dengan model ini, maka saat melakukan develop aplikasi akan menjadi lebih mudah karena pengamanan akses data sudah banyak dilakukan oleh RDBMS sendiri, develop aplikasi hanya menggunakan account yang telah disediakan yang disesuaikan dengan kelompok user yang menggunakan.

Kata Kunci: Data, Hak Akses, Implementasi, Keamanan, RDBMS

1. PENDAHULUAN

Data menjadi hal yang sangat berharga dalam era digital saat ini, kemunculan istilah "data is oil" tidaklah berlebihan, perkembangan ke AI merupakan implementasi dari ketersediaan data yang dikelola secara optimal sebagai misal ChatGPT. Tanpa adanya dukungan data yang lengkap, tidaklah akan bisa terwujud. Seiring dengan perkembangan tersebut, maka keamanan data dan pengaturan akses terhadap data juga sangat diperlukan sehingga privasi informasi menjadi bidang yang ikut mengiringi perkembangan tersebut.

Banyak metode dalam pengembangan sistem informasi untuk melakukan pengamanan data baik dari sisi sistem informasi maupun dari sistem database. Kemampuan perancang RDBMS dalam mengembangkan sebuah RDBMS menjadi salah satu kunci aman tidaknya sebuah rancangan RDBMS, tidak hanya dari sisi referensial integrity saja, akan tetapi juga tentang siap yang berhak untuk mengakses sebuah data atau tabel, apa saja hak yang diberikan, apakah hanya membaca, menambah, mengedit, menghapus serta field-field mana

saja yang diperbolehkan. Dalam sebuah penelitian oleh Joko Triyono dan Prafira Dimas [1] tentang penerapan rancangan database akademik telah menghasilkan kesimpulan bahwa dengan penerapan teknik tersebut diperoleh kemudahan dari sisi develop aplikasi, karena hampir semua rule bisnis yang ada telah di masukkan kedalam rancangan RDBMS, sehingga sangat kecil kemungkinannya terjadi kesalahan persepsi dari sisi develop aplikasi.

Sebuah buku dengan judul Keamanan Basis Data Relasional [2] tentang hak akses atas data dalam sebuah basis data juga harus sudah diperhatikan pada saat analisis kebutuhan sistem. Sebuah basis data bisa dikatakan pemiliknya adalah institusi yang memiliki sistem tersebut. Hak akses tertinggi adalah pada pemilik atau *owner* dari basis data ini. Dari pemilik ini nanti bisa mengatur hak akses terhadap objek yang ada dalam basis data tersebut. Pengaturan hak akses mulai dari *database*, *tabel*, *field* dan *routine* yang ada bisa diatur. Pada buku SQL The Complete Reference Third Edition [3] didefinisikan bahwa perubahan dalam tabel, pengguna dapat menentukan tindakan terkait yang harus dilakukan oleh DBMS yang disebut sebagai konsep trigger.

Beberapa penelitian yang telah dipublikasi tentang keamanan aplikasi oleh Ade Bastian dkk [4] telah diuji menggunakan penetrasi testing dan SQL Injection. Juga oleh Firstyanid kk [5] tentang *constraint* CHECK pada *basisdata*. Juga Pangestu dkk [6] keamanan *cloud database*, Juga Ardian dkk [7] tentang pengamanan database perpustakaan menggunakan algoritma AES-128. Juga Pratama dkk [8] tentang kriptografi untuk pengamanan *database*, juga Wang dkk [9] tentang solusi database security. Khamis dkk [10] Security Framework pada *database* terdistribusi Juga Joko Triyono dkk [11] dikatakan perancangan database pada data terdistribusi dan aplikasi menjadi sebuah permasalahan dimana informasi harus lebih condong kepada pemakai sehingga pemakai menjadi lebih fleksibel dalam mengelola informasi untuk menunjang kegiatan dan pekerjaan.

Penerapan replikasi *database* dikombinasikan dengan *web service server* menjadi salah satu cara agar pemakai kelompok transaksional dan manajerial bisa lebih leluasa dalam mengelola informasi. Erfanti etc [12] menjelaskan bahwa kebanyakan aplikasi, masih lebih banyak menfokuskan bagaimana proses mendaftar dan tanpa menginformasikan fasilitas atau kapasitas secara transparan. Hasil penelitian Joko Triyono dikatakan bahwa [13] sebuah RDBMS harus bisa mengatasi dan mengelola semua proses pada sebuah aplikasi tanpa harus mengganggu bahkan menyentuh sisi aplikasi dari sistem informasi. Termasuk juga dari sisi keamanan dan histori data sebuah sistem informasi harus selalu terjaga secara berkelanjutan dan mudah diakses oleh pihak tertentu, sehingga bisa dianalisis untuk mendapatkan pola dalam pengembangan sistem selanjutnya.

2. BAHAN DAN METODE

2.1. Kebutuhan Bahan

Peralatan dan bahan yang dibutuhkan untuk penelitian ini diantaranya yaitu:

1. Hardware, laptop dengan RAM 8,00 GB.
2. Sistem Operasi Window 11
3. Software XAMPP for Windows 8.1.6

2.2. Metode Pengumpulan Data

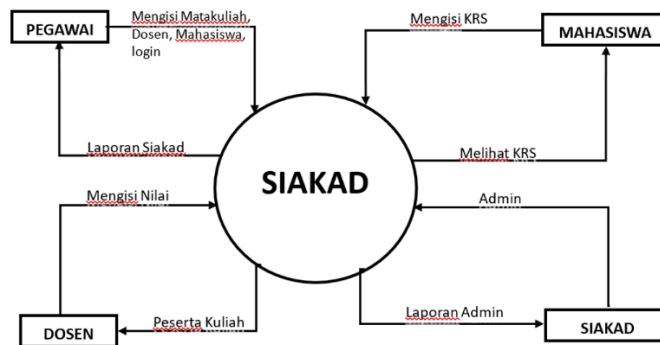
Metode pengumpulan data yang digunakan dalam penelitian, yaitu:

1. Metode Observasi, digunakan untuk pengumpulan data dengan pengamatan secara langsung maupun tidak langsung terhadap obyek yang diteliti.
2. Metode Studi Kepustakaan, merupakan sebuah cara dalam pengumpulan data dengan mempelajari bahan pustaka baik berupa dokumen tertulis ataupun berupa gambar dengan membandingkan beberapa referensi.
3. Metode Eksperimen, digunakan untuk mengadakan uji coba dan simulasi yang telah dibuat dengan menguji secara langsung melalui terminal (SQL Manipulation) dari berbagai hak akses *user*.

2.3. Perancangan Sistem

Peraturan Bisnis, adalah bahwa semua proses akademik dilakukan di RDBMS dan *account* sesuai tugasnya, menggunakan aturan bahwa tiap aktor akan memiliki *user* dan *password* untuk masuk ke terminal SQL dan akan melakukan transaksi pengujian. Aktor siacad akan melakukan kegiatan administrator RDBMS, aktor pegawai akan melakukan transaksi pengisian Mahasiswa dan Mata Kuliah, aktor Mahasiswa akan mengisikan transaksi KRS, aktor Dosen akan melakukan transaksi pengisian nilai. Desain Sistem, berdasarkan aturan yang telah disebutkan, maka ditampilkan pada gambar 1. Dalam kegiatan penelitian ini dibangun dan disimulasikan:

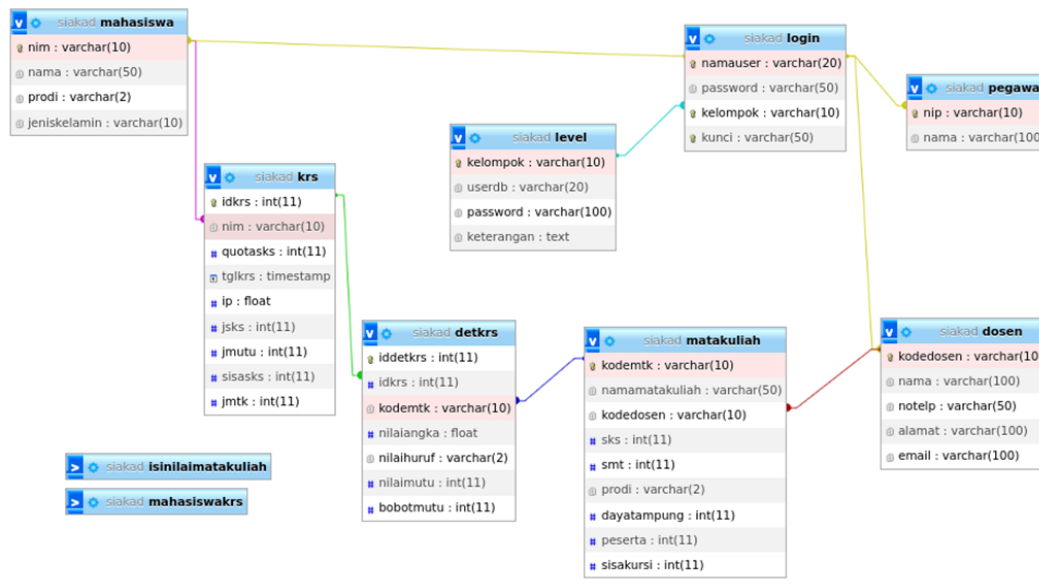
1. RDBMS, meliputi tabel-tabel *relasional*, *trigger*, *view*, *store procedure* dan *function*.
2. Penerapan hak akses untuk masing-masing entitas.



Gambar 1. Desain Sistem

2.4. Perancangan Database

Design tabel dan relationship yang digunakan dalam penelitian ini seperti terlihat pada gambar 2.



Gambar 2. Desain Tabel Relationship

2.5. Perancangan Hak Akses

Dalam RDBMS untuk proses transaksional di dukung oleh *function*, *procedure*, *trigger* dan *view*. Dikarenakan dalam sebuah transaksi terhadap tabel akan mempengaruhi tabel yang lain, atau tabel itu sendiri pada *field* tertentu dengan isian hasil penghitungan tertentu, sehingga pemberian hak akses terhadap sebuah user akan terlihat dari proses ini

2.5.1. Pegawai

Pegawai menurut gambar 1 akan melakukan pekerjaan untuk mengisi data Mata Kuliah, Mahasiswa, Dosen dan Pegawai. Sehingga secara umum pegawai akan diberi hak akses untuk *insert*, *delete* dan *update* terhadap tabel-tabel tersebut. Khusus saat melakukan penambahan pada data Mahasiswa, Dosen dan Pegawai ada trigger yang secara otomatis akan menambahkan data tersebut ketabel *login* sehingga pemberian hak akses insert pada tabel *login* seperti terlihat pada tabel 1 tentang hak akses pegawai dan Tabel 2 hak akses Pegawai terhadap procedure.

```
CREATE TRIGGER `generateuser` AFTER INSERT ON `dosen`
FOR EACH ROW insert into login values(new.kodedosen,'1234','dosen','user dosen')
```

```
CREATE TRIGGER `generateusermhs` AFTER INSERT ON `mahasiswa`
FOR EACH ROW insert into login values(new.nim,'1234','mahasiswa','user mahasiswa')
```

```
CREATE TRIGGER `generateuserpegawai` AFTER INSERT ON `pegawai`
FOR EACH ROW insert into login values(new.nip,'1234','pegawai','user pegawai')
```

Tabel 1. Hak akses Pegawai terhadap tabel

Table_Name	Table_Priv
pegawai	Select, Insert, Update, Trigger
krs	Select, Delete, Trigger
detkrs	Select, Delete, Trigger
matakuliah	Select, Insert, Update, Delete, Trigger
dosen	Select, Insert, Update, Delete, Trigger
level	Select, Update
mahasiswa	Select, Insert, Update, Delete, Trigger
login	Select

Tabel 2. Hak akses Pegawai terhadap *procedure*

Routine_name	Routine_type	Grantor	Proc_priv
DelDetKRS	PROCEDURE	root@localhost	Execute
getDetKRS	PROCEDURE	root@localhost	Execute
getKRS	PROCEDURE	root@localhost	Execute
getIsiNilai	PROCEDURE	root@localhost	Execute

2.5.2. Dosen

Dosen menurut gambar 1 akan melakukan transaksi pengisian nilai, pada proses ini Dosen hanya akan mengisikan nilai pada tabel detkrs pada kolom/ *field* nilai angka dari data KRS yang telah ada, sehingga pemberian hak akses Dosen hanya pada *field* tersebut untuk melakukan *update*. Pada saat melakukan pengisian nilai tersebut ada function yang secara otomatis akan mengupdate *nilaimutu* dan *nilaihuruf* pada tabel detkrs serta akan mengupdate tabel KRS pada *field jmutu* menggunakan *trigger updatenilai* juga saat melakukan konversi nilaiangka ke huruf digunakan fungsi *cNilaiHuruf* dan saat melakukan konversi nilaimutu digunakan fungsi *cNilaiMutu*. Pada saat melakukan transaksi pengisian nilai akan dibantu oleh sebuah *store procedure updatenilai*, dengan perintah *call updatenilai(iddetkrs, nilaiangka)*;

Dengan begitu, maka pemberian akses terhadap Dosen akan terlihat pada tabel 3 terhadap tabel, pada tabel 4 terhadap *field* dan pada tabel 5 terhadap *procedure*.

Tabel 3. Hak Akses Dosen terhadap tabel

Table_Name	Table_Priv	Column_Priv
dosen	Select	Update,References
detkrs	Select,Trigger	Update,References
krs	Select,Trigger	Update
matakuliah	Select	
login		Update

Tabel 4. Hak Akses Dosen terhadap field

Table_name	Column_name	Column_priv
dosen	nama	Update
dosen	notelp	Update
dosen	alamat	Update
dosen	email	Update
dosen	kodedosen	References
detkrs	nilaiangka	Update
detkrs	nilaihuruf	Update
detkrs	nilaimutu	Update
detkrs	bobotmutu	Update
krs	ip	Update
krs	jmutu	Update
detkrs	idkrs	References
detkrs	kodemtk	References
login	password	Update

Tabel 5. Hak akses Dosen terhadap procedure

Routine_name	Routine_type	Grantor	Proc_priv
updatenilai	PROCEDURE	root@localhost	Execute
cNilaiHuruf	FUNCTION	root@localhost	Execute
cNilaiMutu	FUNCTION	root@localhost	Execute

2.5.3. Mahasiswa

Mahasiswa menurut gambar 1 akan melakukan transaksi pengisian KRS, pada proses ini Mahasiswa akan menambahkan *record* pada tabel KRS dan tabel detkrs, sehingga pemberian hak akses Mahasiswa hanya pada tabel-tabel tersebut untuk melakukan insert. Pada saat melakukan pengisian KRS tersebut ada *function*, *trigger* dan *procedure* yang secara otomatis akan mempengaruhi tabel-tabel lain.

Proses pengisian KRS: Pertama, memanggil fungsi *NewKRS (nim)*, fungsi ini akan melakukan proses insert data *nim* pada tabel krs serta menghasilkan nilai balik berupa *idkrs*. Kedua, mengisikan Mata Kuliah yang diambil dengan menggunakan fungsi *DetKRS(idkrs,kodemtk)* proses ini akan melakukan penambahan data pada table detkrs. Setelah dilakukan proses insert tersebut, terdapat sebuah trigger *adddetkrs* yang akan melakukan perubahan data pada tabel KRS pada field *jsks* dan *jmtk*, serta perubahan data yang terjadi pada *field* peserta dan sisakursi dari tabel Mata Kuliah. Sehingga dari proses-proses tersebut di susun hak akses untuk Mahasiswa pada tabel 6 terhadap tabel, pada tabel 7 terhadap *field* dan pada tabel 8 terhadap *procedure*.

Tabel 6. Hak Akses Mahasiswa terhadap tabel

table_name	table_priv	column_priv
krs	Select,Insert,Show view,Trigger	Update,References
detkrs	Select,Insert,Show view,Trigger	References
login	Select,Show view,Trigger	Update,References
Mahasiswa	Select,Insert,Delete,Show view,Trigger	Update,References
Matakuliah	Select,Show view,Trigger	References
dosen	Select	
level	Select	

Tabel 7. Hak akses Mahasiswa terhadap field

Table_name	Column_name	Column_priv
krs	jsks	Update
krs	sisasks	Update
krs	jmtk	Update
krs	nim	References
detkrs	idkrs	References
detkrs	kodemtk	References
login	password	Update
login	kelompok	References
mahasiswa	prodi	References
mahasiswa	nama	Update,References
mahasiswa	jeniskelamin	Update,References
matakuliah	kodedosen	References
matakuliah	prodi	References
mahasiswa	nim	References

Tabel 8. Hak akses Mahasiswa terhadap procedure

Host	Db	User	Routine_name	Routine_type	Grantor	Proc_priv
%	siakad	mahasiswa	NewKRS	FUNCTION	root@localhost	Execute
%	siakad	mahasiswa	DetKRS	PROCEDURE	root@localhost	Execute

3. HASIL DAN PEMBAHASAN

Implementasi perancangan dalam penelitian ini dibuktikan dengan melakukan proses kegiatan mulai dari login sampai transaksi SQL pada terminal.

3.1. Pegawai

Pegawai akan melakukan login pada terminal mysql, mengisikan matakuliah, dosen dan mahasiswa serta melihat data.

Login ke terminal:

```

abab@abab-HP-Laptop-14s-dq0xxx:~$ mysql -u pegawai -p
...
...
MariaDB [(none)]> use siakad
...
...
Database changed
MariaDB [siakad]> show tables;
+-----+
| Tables_in_siakad |
+-----+

```

```

+-----+
| detkrs |
| dosen |
| krs    |
| level |
| login  |
| mahasiswa |
| matakuliah |
| pegawai |
+-----+
    
```

8 rows in set (0,000 sec)

Melakukan transaksi isi data Mata Kuliah, sebelumnya dilihat dulu isi tabel matakuliah lalu melakukan *insert* dan dilihat tabel setelah *insert*, data bisa sukses dimasukkan.

```

MariaDB [siakad]> select * from matakuliah;
+-----+-----+-----+-----+-----+-----+-----+-----+
| kodemtk | namamatakuliah | kodedosen | sks | smt | prodi | dayatampung | peserta | sisakursi |
+-----+-----+-----+-----+-----+-----+-----+-----+
| TKP001 | Pancasila      | d001      | 2   | 1   | 01    | 40           | 1       | 39        |
| TKP002 | Bahasa Inggris | d002      | 2   | 1   | 01    | 40           | 1       | 39        |
| TKP003 | Bahasa Indonesia | d003     | 3   | 1   | 01    | 40           | 0       | 40        |
+-----+-----+-----+-----+-----+-----+-----+-----+
    
```

3 rows in set (0,000 sec)

```

MariaDB [siakad]> insert into matakuliah(kodemtk,namamatakuliah,kodedosen,sks,smt,prodi)
values('TKP004','Fisika Dasar','d003',4,1,'01');
Query OK, 1 row affected (0,002 sec)
    
```

```

MariaDB [siakad]> select * from matakuliah;
+-----+-----+-----+-----+-----+-----+-----+-----+
| kodemtk | namamatakuliah | kodedosen | sks | smt | prodi | dayatampung | peserta | sisakursi |
+-----+-----+-----+-----+-----+-----+-----+-----+
| TKP001 | Pancasila      | d001      | 2   | 1   | 01    | 40           | 1       | 39        |
| TKP002 | Bahasa Inggris | d002      | 2   | 1   | 01    | 40           | 1       | 39        |
| TKP003 | Bahasa Indonesia | d003     | 3   | 1   | 01    | 40           | 0       | 40        |
| TKP004 | Fisika Dasar   | d003     | 4   | 1   | 01    | 40           | 0       | 40        |
+-----+-----+-----+-----+-----+-----+-----+-----+
    
```

4 rows in set (0,001 sec)

Melakukan transaksi menambahkan Dosen, dilihat data Dosen dan login sebelum dan sesudah ditambahkan.

```

MariaDB [siakad]> select * from login;
+-----+-----+-----+-----+
| namauser | password | kelompok | kunci |
+-----+-----+-----+-----+
| 05001    | 1234     | pegawai  | user pegawai |
| 05006    | 1234     | mahasiswa | user mahasiswa |
| 9999     | 1234     | pegawai  | user pegawai |
| d001     | 1234     | dosen    | user dosen |
| d002     | 1234     | dosen    | user dosen |
| d003     | 1234     | dosen    | user dosen |
+-----+-----+-----+-----+
    
```

6 rows in set (0,001 sec)

```

MariaDB [siakad]> select * from dosen;
+-----+-----+-----+-----+-----+
| kodedosen | nama          | notelp    | alamat    | email |
+-----+-----+-----+-----+-----+
| d001      | Slamet Rahardjo | 08132778888 | Klodangan | slamet@gmail.com |
| d002      | Endang         | 08132733    | Klaten    | endang@gmail.com |
| d003      | susi           | 08132733    | Klaten    | susi@gmail.com |
+-----+-----+-----+-----+-----+
    
```

3 rows in set (0,001 sec)

```

MariaDB [siakad]> insert into dosen values('d004','Joko','081327780101','Jogjakarta','joko@gmail.com');
Query OK, 1 row affected (0,002 sec)
    
```

```

MariaDB [siakad]> select * from dosen;
+-----+-----+-----+-----+-----+
| kodedosen | nama          | notelp    | alamat    | email |
+-----+-----+-----+-----+-----+
| d001      | Slamet Rahardjo | 08132778888 | Klodangan | slamet@gmail.com |
| d002      | Endang         | 08132733    | Klaten    | endang@gmail.com |
| d003      | susi           | 08132733    | Klaten    | susi@gmail.com |
| d004      | Joko           | 081327780101 | Jogjakarta | joko@gmail.com | ←data baru
+-----+-----+-----+-----+-----+
    
```

4 rows in set (0,000 sec)

```

MariaDB [siakad]> select * from login;
+-----+-----+-----+-----+
| namauser | password | kelompok | kunci |
+-----+-----+-----+-----+
| 05001    | 1234     | pegawai  | user pegawai |
| 05006    | 1234     | mahasiswa | user mahasiswa |
| 9999     | 1234     | pegawai  | user pegawai |
| d001     | 1234     | dosen    | user dosen |
| d002     | 1234     | dosen    | user dosen |
| d003     | 1234     | dosen    | user dosen |
| d004     | 1234     | dosen    | user dosen | ←data baru
+-----+-----+-----+-----+
    
```

7 rows in set (0,001 sec)

Terbukti bahwa setelah memasukkan data Dosen baru, secara otomatis akan ditambahkan pada tabel login. Melakukan transaksi menambahkan Mahasiswa, dilihat data Mahasiswa dan login sebelum dan sesudah ditambahkan.

```
MariaDB [siakad]> select * from login;
```

namauser	password	kelompok	kunci
05001	1234	pegawai	user pegawai
05006	1234	mahasiswa	user mahasiswa
9999	1234	pegawai	user pegawai
d001	1234	dosen	user dosen
d002	1234	dosen	user dosen
d003	1234	dosen	user dosen
d004	1234	dosen	user dosen

```
7 rows in set (0,001 sec)
```

```
MariaDB [siakad]> select * from mahasiswa;
```

nim	nama	prodi	jeniskelamin
05001	Joko	01	L
05002	Edi	01	L
05006	ahmad	01	L

```
3 rows in set (0,001 sec)
```

```
MariaDB [siakad]> insert into mahasiswa values('05007','kenzie','01','L');
```

```
Query OK, 1 row affected (0,002 sec)
```

```
MariaDB [siakad]> select * from mahasiswa;
```

nim	nama	prodi	jeniskelamin
05001	Joko	01	L
05002	Edi	01	L
05006	ahmad	01	L
05007	kenzie	01	L

```
4 rows in set (0,001 sec)
```

```
MariaDB [siakad]> select * from login;
```

namauser	password	kelompok	kunci
05001	1234	pegawai	user pegawai
05006	1234	mahasiswa	user mahasiswa
05007	1234	mahasiswa	user mahasiswa
9999	1234	pegawai	user pegawai
d001	1234	dosen	user dosen
d002	1234	dosen	user dosen
d003	1234	dosen	user dosen
d004	1234	dosen	user dosen

```
8 rows in set (0,001 sec)
```

Terbukti bahwa setelah memasukkan data Mahasiswa baru, secara otomatis akan ditambahkan pada tabel *login*.

3.2. Mahasiswa

Mahasiswa akan melakukan login pada terminal mysql, mengisi krs dan melihat data

Login ke terminal

```
abah@abab-HP-Laptop-14s-dq0xxx:~$ mysql -u mahasiswa -p
...
MariaDB [(none)]> use siakad
...
MariaDB [siakad]> show tables;
```

Mengisikan KRS, dilihat dulu isi tabel krs, detkrs dan matakuliah serta setelah proses pengisian.

```
MariaDB [siakad]> select * from detkrs;
```

iddetkrs	idkrs	kodemtk	nilaiangka	nilaihuruf	nilaimutu	bobotmutu
2	1	TKP001	30	B	3	6
3	1	TKP002	NULL	NULL	0	0

```
2 rows in set (0,001 sec)
```

```
MariaDB [siakad]> select * from krs;
```

idkrs	nim	quotask	tgkrs	ip	jsks	jmutu	sisask	jmtk
1	05001	24	2023-06-15 23:35:46	1.5	4	6	20	2

```
1 row in set (0,001 sec)
```

```
MariaDB [siakad]> select * from matakuliah;
```

kodemtk	namamatakuliah	kodedosen	sks	smt	prodi	dayatampung	peserta	sisakursi
TKP001	Pancasila	d001	2	1	01	40	1	39
TKP002	Bahasa Inggris	d002	2	1	01	40	1	39
TKP003	Bahasa Indonesia	d003	3	1	01	40	0	40
TKP004	Fisika Dasar	d003	4	1	01	40	0	40

4 rows in set (0,001 sec)

```
MariaDB [siakad]> select NewKRS('05007'); ←Perintah mengisi KRS step-1
```

```
+-----+
| NewKRS('05007') |
+-----+
|                2 | ←Nilai balik idkrs
+-----+
1 row in set (0,002 sec)
```

```
MariaDB [siakad]> select * from krs;
```

idkrs	nim	quotasks	tgkrs	ip	jsks	jmutu	sisasks	jmtk
1	05001	24	2023-06-15 23:35:46	1.5	4	6	20	2
2	05007	24	2023-06-17 08:55:08	0	0	0	24	0

←Data Baru
2 rows in set (0,001 sec)

```
MariaDB [siakad]> select * from detkrs;
```

iddetkrs	idkrs	kodemtk	nilaiangka	nilaihuruf	nilaimutu	bobotmutu
2	1	TKP001	30	B	3	6
3	1	TKP002	NULL	NULL	0	0

2 rows in set (0,001 sec)

```
MariaDB [siakad]> select * from matakuliah;
```

kodemtk	namamatakuliah	kodedosen	sks	smt	prodi	dayatampung	peserta	sisakursi
TKP001	Pancasila	d001	2	1	01	40	1	39
TKP002	Bahasa Inggris	d002	2	1	01	40	1	39
TKP003	Bahasa Indonesia	d003	3	1	01	40	0	40
TKP004	Fisika Dasar	d003	4	1	01	40	0	40

4 rows in set (0,001 sec)

```
MariaDB [siakad]> call DetKRS(2,'TKP004'); ←Perintah mengisi KRS step-2
Query OK, 3 rows affected (0,002 sec)
```

```
MariaDB [siakad]> select * from matakuliah;
```

kodemtk	namamatakuliah	kodedosen	sks	smt	prodi	dayatampung	peserta	sisakursi
TKP001	Pancasila	d001	2	1	01	40	1	39
TKP002	Bahasa Inggris	d002	2	1	01	40	1	39
TKP003	Bahasa Indonesia	d003	3	1	01	40	0	40
TKP004	Fisika Dasar	d003	4	1	01	40	1	39

←
4 rows in set (0,001 sec)

```
MariaDB [siakad]> select * from detkrs;
```

iddetkrs	idkrs	kodemtk	nilaiangka	nilaihuruf	nilaimutu	bobotmutu
2	1	TKP001	30	B	3	6
3	1	TKP002	NULL	NULL	0	0
4	2	TKP004	NULL	NULL	0	0

←Data baru
3 rows in set (0,001 sec)

```
MariaDB [siakad]> select * from krs;
```

idkrs	nim	quotasks	tgkrs	ip	jsks	jmutu	sisasks	jmtk
1	05001	24	2023-06-15 23:35:46	1.5	4	6	20	2
2	05007	24	2023-06-17 08:55:08	0	4	0	20	1

←Perubahan Data
2 rows in set (0,001 sec)

Dari pengujian tersebut proses berjalan sukses, akan tetapi jika user mahasiswa ini akan melakukan beberapa transaksi yang tidak sesuai SOP akan ditolak oleh sistem, beberapa contoh dibawah ini menunjukkan kegiatan tersebut. Misalkan melakukan perubahan Nilaiangka pada tabel detkrs, IP pada tabel krs, penambahan dan atau perubahan data pada tabel Mata Kuliah .

```
MariaDB [siakad]> update detkrs set nilaiangka=50 where iddetkrs=4;
ERROR 1142 (42000): UPDATE command denied to user 'mahasiswa'@'localhost' for table `siakad`.`detkrs`
```



```

MariaDB [siakad]> update krs set ip=2 where idkrs=2;
ERROR 1143 (42000): UPDATE command denied to user 'mahasiswa'@'localhost' for column 'ip' in table 'krs'

MariaDB [siakad]> update matakuliah set kodedosen='d004' where kodemtk='TKP003';
ERROR 1142 (42000): UPDATE command denied to user 'mahasiswa'@'localhost' for table
'siakad`.`matakuliah

MariaDB [siakad]> insert into matakuliah(kodemtk,namamatakuliah,kodedosen,sks,smt,prodi)
values('TKPxxx','Kimia umum','d004',3,2,'01');
ERROR 1142 (42000): INSERT command denied to user 'mahasiswa'@'localhost' for table
'siakad`.`matakuliah

```

3.3. Dosen

Dosen akan melakukan login pada terminal mysql, mengisi nilai dan melihat data.

```

abab@abab-HP-Laptop-14s-dq0xxx:~$ mysql -u dosen -p
...
MariaDB [(none)]> use siakad;
...
MariaDB [siakad]> show tables;

```

Melakukan transaksi pengisian nilaiangka

```

MariaDB [siakad]> select * from matakuliah;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| kodemtk | namamatakuliah | kodedosen | sks | smt | prodi | dayatampung | peserta | sisakursi |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| TKP001 | Pancasila      | d001      | 2   | 1   | 01    | 40           | 1       | 39        |
| TKP002 | Bahasa Inggris | d002      | 2   | 1   | 01    | 40           | 1       | 39        |
| TKP003 | Bahasa Indonesia | d003     | 3   | 1   | 01    | 40           | 0       | 40        |
| TKP004 | Fisika Dasar   | d003      | 4   | 1   | 01    | 40           | 1       | 39        |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (0,001 sec)

```

```

MariaDB [siakad]> select * from detkrs;
+-----+-----+-----+-----+-----+-----+-----+
| iddetkrs | idkrs | kodemtk | nilaiangka | nilaihuruf | nilai mutu | bobot mutu |
+-----+-----+-----+-----+-----+-----+-----+
| 2        | 1     | TKP001 | 30          | B          | 3          | 6          |
| 3        | 1     | TKP002 | NULL        | NULL       | 0          | 0          |
| 4        | 2     | TKP004 | NULL        | NULL       | 0          | 0          |
+-----+-----+-----+-----+-----+-----+-----+
3 rows in set (0,001 sec)

```

```

MariaDB [siakad]> select * from krs;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| idkrs | nim | quotasks | tglkrs | ip | jsks | jmutu | sisasks | jmtk |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1     | 05001 | 24 | 2023-06-15 23:35:46 | 1.5 | 4 | 6 | 20 | 2 |
| 2     | 05007 | 24 | 2023-06-17 08:55:08 | 0   | 4 | 0 | 20 | 1 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0,001 sec)

```

```

MariaDB [siakad]> call updatenilai(4,90); ← Proses Pengisian Nilaiangka
Query OK, 2 rows affected (0,004 sec)

```

```

MariaDB [siakad]> select * from krs;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| idkrs | nim | quotasks | tglkrs | ip | jsks | jmutu | sisasks | jmtk |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1     | 05001 | 24 | 2023-06-15 23:35:46 | 1.5 | 4 | 6 | 20 | 2 |
| 2     | 05007 | 24 | 2023-06-17 08:55:08 | 4   | 4 | 16 | 20 | 1 | ← Data
+-----+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0,001 sec)

```

```

MariaDB [siakad]> select * from detkrs;
+-----+-----+-----+-----+-----+-----+-----+
| iddetkrs | idkrs | kodemtk | nilaiangka | nilaihuruf | nilai mutu | bobot mutu |
+-----+-----+-----+-----+-----+-----+-----+
| 2        | 1     | TKP001 | 30          | B          | 3          | 6          |
| 3        | 1     | TKP002 | NULL        | NULL       | 0          | 0          |
| 4        | 2     | TKP004 | 90         | A          | 4          | 16         | ← Data Baru
+-----+-----+-----+-----+-----+-----+-----+
3 rows in set (0,001 sec)

```

Jika user Dosen ini melakukan kegiatan diluar haknya akan terlihat seperti melakukan perubahan kodemtk pada tabel detkrs, menambah data pada tabel detkrs, merubah nim pada tabel krs, menambah data pada tabel krs, merubah dan atau menambah data pada tabel matakuliah.

```

MariaDB [siakad]> update detkrs set kodemtk='TKP004' where iddetkrs=3;
ERROR 1143 (42000): UPDATE command denied to user 'dosen'@'localhost' for column 'kode mtk'
in table 'detkrs'
MariaDB [siakad]> insert detkrs (idkrs,kodemtk) values(2,'TKP002');
ERROR 1142 (42000): INSERT command denied to user 'dosen'@'localhost' for table
'siakad`.`detkrs'
MariaDB [siakad]> update krs set nim='05006' where idkrs=2;

```

```

ERROR 1143 (42000): UPDATE command denied to user 'dosen'@'localhost' for column 'nim' in
table 'krs'
MariaDB [siakad]> insert krs (nim) values('05006');
ERROR 1142 (42000): INSERT command denied to user 'dosen'@'localhost' for table
'siakad`.`krs`
MariaDB [siakad]> update matakuliah set kodedosen='d004' where kodemtk='TKP003';
ERROR 1142 (42000): UPDATE command denied to user 'dosen'@'localhost' for table
'siakad`.`matakuliah`
MariaDB [siakad]> insert into matakuliah(kodemtk,namamatakuliah,kodedosen,sk,smt,prodi)
values('TKPxxx','Kimia umum','d004',3,2,'01');
ERROR 1142 (42000): INSERT command denied to user 'dosen'@'localhost' for table
'siakad`.`matakuliah`
MariaDB [siakad]>

```

4. CONCLUSION

Dari hasil rancangan, implementasi dan pengujian pada RDBMS akademik diperoleh kesimpulan bahwa rancangan dengan penerapan hak akses secara tepat dan benar akan meningkatkan keamanan RDBMS dari kegiatan-kegiatan yang tidak diinginkan, sehingga nilai informasi dari RDBMS tersebut layak untuk dipercaya. Dengan model ini, maka saat melakukan *develop* aplikasi akan menjadi lebih mudah karena pengamanan akses data sudah banyak dilakukan oleh RDBMS sendiri, *develop* aplikasi hanya menggunakan account yang telah disediakan yang disesuaikan dengan kelompok user yang menggunakan.

REFERENCES

- [1]. J. Triyono and P. D. Akbar, "Implementasi Rancangan Database Akademik Menggunakan Function, Store Procedure, Trigger dan View," FAHMA - Jurnal Teknologi dan Ilmu Komputer, vol. Vol 21 No 1, pp. 45-59, 2023.
- [2]. S. Raharjo and E. Utami, Keamanan Basis Data Relasional, Yogyakarta: Andi, 2022.
- [3]. MySQL Tutorial, "MySQL Views," MySQL Tutorial, [Online]. Available: <https://www.mysqltutorial.org/mysql-views-tutorial.aspx>. [Accessed 15 Desember 2022].
- [4]. A. Bastian, H. Sujadi and L. Abror, "ANALISIS KEAMANAN APLIKASI DATA POKOK PENDIDIKAN (DAPODIK) MENGGUNAKAN PENETRATION TESTING DAN SQL INJECTION," infotech, vol. 6, no. 2, pp. 65-70, 2020.
- [5]. F. I. Rahma, T. Agustin, R. M. James and E. Utami, "Implementasi Constraint CHECK Pada Basis Data Aplikasi LaundryPOS Dalam Aspek Kebenaran Data," Creative Information Technology Journal (CITEC Journal), vol. 7, no. 2, pp. 133-141, 2020.
- [6]. D. Pangestu and A. Syahputra, "PERANCANGAN APLIKASI KEAMANAN CLOUD DATABASE MENGGUNAKAN OPERASI XOR DENGAN ALGORITMA AFFINE BERBASIS ANDROID," IT (INFORMATIC TECHNIQUE) JOURNAL, vol. 8, no. 1, pp. 54-64, April 2020.
- [7]. M. T. Ardian and W. Pramusinto, "Pengamanan Database Perpustakaan Dengan Algoritma AES-128 Pada SMA Waskito," in Prosiding Seminar Nasional Mahasiswa Fakultas Teknologi Informasi (SENAFTI), Jakarta, 2022.
- [8]. Y. P. Putra, F. Nuraeni and R. A. Jatnika, "Implementasi Kriptografi Dalam Pengamanan Database E-Voting Menggunakan Algoritma Rsa Dan Base64 Berbasis Progressive Web Apps," JUSITI (Jurnal Sistem Informasi dan Teknologi Informasi), vol. 10, no. 1, pp. 30-40, 2021.
- [9]. Y. Wang, J. Xi and T. Cheng, "The Overview of Database Security Threats' Solutions: Traditional and Machine Learning," Journal of Information Security, vol. 12, no. 1, pp. 34-35, 2021.
- [10]. A. D. Khamis and S. Subair, "Security Framework for Distributed Database System," Journal of Data Analysis and Information Processing, vol. 7, pp. 1-13, 2019.
- [11]. J. Triyono, P. Nadira and C. A. Subhkan, "IMPLEMENTASI SISTEM TERDISTRIBUSI MENGGUNAKAN REPLIKASI DATABASE DAN WEB SERVICE," in Prosiding Seminar Nasional Multidisiplin Ilmu, Yogyakarta, 2021.
- [12]. E. Fatkhiyah, J. Triyono and E. N. Cahyo, "PERANCANGAN DATABASE PENDAFTARAN SWAB ANTIGEN BERBASIS ONLINE," in Prosiding Seminar Nasional Multidisiplin Ilmu, Yogyakarta, 2021.
- [13]. J. Triyono, P. Haryani and A. Padmanaba, "Model Kontrol Transaksi RDBMS Menggunakan Trigger dan Waktu Server," Jurnal Teknologi, vol. 12, no. 1, pp. 80-86, 2019.