



PERANCANGAN DESAIN BASIS DATA SEMINAR HASIL BERBASIS DBLC DENGAN INTEGRASI KALENDER DOSEN DAN NOTIFIKASI OTOMATIS UNTUK EFISIENSI DAN TRANSPARANSI PENJADWALAN

Khansa Firyal Adiila^{a,1,*}, Aurelia Callysta Mamahit^{b,2}, Diva Kurnianingtyas, Dr., S.Kom.^{c,3}

^{a, b, c} Fakultas Ilmu Komputer, Universitas Brawijaya, Malang, 65154, Indonesia

¹ khansafiryaladiila@gmail.com* ² aureliacallysta44@gmail.com; ³ divaku@ub.ac.id;

* corresponding author

ARTICLE INFO

ABSTRACT

Kata kunci: Basis Data, Database Life Cycle (DBLC), Normalisasi, Penjadwalan Seminar Hasil, Kalender Dosen, Notifikasi Otomatis.

Scheduling seminars at universities is still sometimes hampered by manual processes and systems that are not integrated with faculty calendars, leading to scheduling conflicts, delayed confirmations, and a lack of transparency. The purpose of this study is to design an efficient and transparent database for scheduling seminars that is integrated with faculty calendars and automatic notifications, thereby minimizing delays and cancellations. The methods used include a literature review, interviews, and the Database Life Cycle (DBLC), encompassing the conceptual, logical, and physical design stages. Normalization was carried out to the third normal form (3NF) to ensure structured data without redundancy. Quantitative results show that the system produces 13 main tables interconnected through primary and foreign keys, which are capable of eliminating data duplication and remain consistent in validation tests. In addition, the system supports the sending of notifications at least seven days before the seminar. This database design can improve the efficiency, accuracy, and transparency of seminar scheduling. This study differs from previous research as it not only applies the DBLC for database design but also emphasizes the integration of faculty calendars and automatic notification mechanisms, aspects that have not been explored in depth in earlier works.

1. Pendahuluan

Seiring berjalannya waktu, teknologi terus berkembang dengan pesat mengikuti perkembangan zaman. Saat ini, sudah banyak manfaat teknologi yang diterapkan dalam kehidupan sehari-hari, termasuk di lingkungan perguruan tinggi[1]. Salah satu contoh penerapan teknologi yang terlihat di perguruan tinggi adalah penerapan dalam penjadwalan seminar hasil. Namun, masih terdapat beberapa kekurangan pada sistem tersebut. Di antaranya adalah jadwal seminar yang sering diumumkan secara mendadak, kurangnya transparansi status dokumen dan proses penjadwalan, serta tidak adanya mekanisme yang efektif untuk memastikan kehadiran dosen pembimbing dan penguji. Kondisi ini menyebabkan mahasiswa kesulitan untuk mempersiapkan diri secara optimal, sementara dosen pembimbing dan dosen penguji terkadang mengalami benturan jadwal, sehingga berakibat pada tertundanya pelaksanaan seminar hasil.

Untuk mewujudkan sistem tersebut, dibutuhkan fondasi pengelolaan data yang kuat, yaitu melalui penerapan basis data. Basis data adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis yang bisa diperiksa menggunakan suatu program komputer untuk mendapatkan informasi dari basis data tersebut[2]. Basis data berfungsi untuk menyimpan, mengelola, dan memanipulasi data dengan efisien[3]. Ada pun cara untuk merepresentasikan model data dalam perancangan basis data, yaitu dengan model basis data relasional[4]. Salah satu contohnya adalah *Entity Relational Database* (ERD) yang telah dipakai sejak lama dan dalam



penerapannya juga membutuhkan metode yang tepat[5].

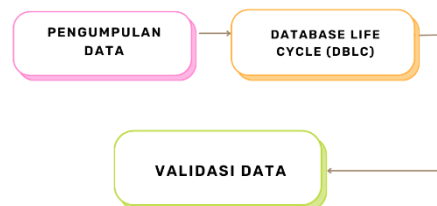
Tujuan dari penelitian ini adalah untuk merancang dan mengembangkam sistem informasi berbasis basis data yang dapat membantu proses penjadwalan seminar hasil secara lebih efisien dan transparan dibandingkan sistem yang sudah ada, sehingga proses administrasi dapat berjalan lebih cepat dan terstruktur. Selain itu, penelitian ini juga bertujuan untuk membangun sistem yang terintegrasi dengan kalender dosen agar penjadwalan seminar hasil menjadi lebih mudah, mengurangi risiko benturan waktu, serta meningkatkan keterlibatan dosen pembimbing maupun dosen penguji. Selain itu, penelitian ini diarahkan untuk menyediakan fitur notifikasi dan konfirmasi otomatis bagi mahasiswa, sehingga mahasiswa dapat memperoleh informasi secara tepat waktu dan dapat mendukung kelancaran pelaksanaan seminar hasil.

2. Metodologi Penelitian

Metodologi penelitian yang digunakan dalam studi ini meliputi pengumpulan data, *Database Life Cycle* (DBLC), dan validasi data. Metodologi penelitian yang pertama kali dilakukan adalah pengumpulan data dengan melakukan studi literatur untuk mengumpulkan teori-teori dari penelitian sebelumnya yang relevan dengan penjadwalan seminar hasil. Selanjutnya, dilakukan pengumpulan data dengan melakukan wawancara untuk mendapatkan informasi secara langsung mengenai sistem penjadwal seminar hasil. Dalam penyusunan basis data, digunakan *Database Life Cycle* (DBLC).

DBLC adalah serangkaian tahapan sistematis dalam mengembangkan dan mengelola basis data dari awal hingga akhir. Aspek penelitian dalam penyusunan basis data menggunakan DBLC, meliputi perancangan basis data konseptual, perancangan basis data logikal, dan perancangan basis data fisik[9]. Setelah perancangan konseptual selesai, model basis data diterjemahkan ke dalam bentuk logikal. Pada tahap ini, dilakukan normalisasi hingga 3NF untuk memastikan struktur tabel efisien dan bebas redundansi, sebelum diterapkan pada tahap perancangan fisik. Selain itu, validasi data dilakukan melalui uji struktural untuk memastikan rancangan basis data bebas redundansi, konsisten, serta sesuai dengan prinsip desain basis data yang baik.

2. Tahapan Penelitian



Gambar 1. Tahapan Penelitian

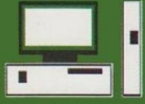
3. Hasil dan Pembahasan

Pada hasil dan pembahasan akan dibahas mengenai tiga tahapan dalam metodologi *Database Life Cycle* (DBLC). Dalam membuat sistem basis data, diperlukan perancangan yang baik. Oleh karena itu, perancangan basis data yang dilakukan meliputi, perancangan basis data konseptual, logikal, dan fisik.

Tabel yang dihasilkan:

nama_dosen_pembimbing	jumlah
Budianto Kusuma, S.T., M.T., Alvian Putra, S.T....	1
Karyadi Budianto, S.Kom., M.Kom., Rohmah Ya...	1
Agung Kusyandi, S.T., M.T., Erwindi Nur Rahma...	1
Alvian Putra, S.T., M.T, Karyadi Budianto, S.Ko...	1
Budianto Kusuma, S.T., M.T., Agung Kusyandi, ...	1

Hasil *query* sebelum dilakukan normalisasi menghasilkan lima baris data, tetapi terdapat duplikasi



atau redundansi data di setiap barisnya. Redundansi adalah kondisi ketika suatu data tersimpan berulang kali di dalam basis data tanpa adanya kebutuhan. Oleh karena itu, basis data tersebut harus dilakukan proses normalisasi.

b. Setelah Normalisasi

```
USE semhas_setelah_normalisasi;  
SELECT dosen_pembimbing.nama_dosen_pembimbing, COUNT(*) AS jumlah  
FROM relasi_jadwal_dosen_pembimbing  
JOIN dosen_pembimbing ON relasi_jadwal_dosen_pembimbing.id_dosen_pembimbing =  
dosen_pembimbing.id_dosen_pembimbing  
GROUP BY dosen_pembimbing.nama_dosen_pembimbing;
```

Tabel yang dihasilkan:

nama_dosen_pembimbing	jumlah
Budianto Kusuma, S.T., M.T.	1
Alvian Putra, S.T., M.T.	2
Karyadi Budianto, S.Kom., M.Kom.	2
Rohmah Yanti, S.T., M.T	1
Agung Kusyandi, S.T., M.T.	2
Erwindi Nur Rahmat, S.T., M.T.	2

Setelah dilakukan normalisasi, *query* tersebut menghasilkan enam baris pada tabel dengan satu dosen pembimbing di setiap barisnya dan tidak ada duplikasi atau redundansi data pada setiap barisnya. Proses normalisasi hingga 3NF dapat menghilangkan redundansi atau duplikasi data.

1. Uji Konsistensi Data (*Data Consistency Test*)

- a. Mencari jumlah mahasiswa
- 1. Sebelum Normalisasi

```
USE semhas_sebelum_normalisasi;  
SELECT COUNT(*) AS total_mahasiswa FROM  
semhas_sebelum_normalisasi
```

Tabel yang dihasilkan:

total_mahasiswa
5

Uji konsistensi data yang pertama, dilakukan dengan mencari total atau jumlah mahasiswa dari database sistem seminar hasil sebelum dilakukannya proses normalisasi. Dari hasil uji *query* tersebut, diperoleh total atau jumlah mahasiswa sebanyak lima orang.

2. Sesudah Normalisasi

```
USE semhas_setelah_normalisasi;  
SELECT COUNT(*) AS total_mahasiswa FROM  
mahasiswa;
```

Tabel yang dihasilkan:

total_mahasiswa
5



Setelah dilakukan proses normalisasi dari 1NF–3NF. Kemudian dilakukan uji *query* untuk mendapatkan jumlah mahasiswa, diperoleh jumlah atau total mahasiswa terdapat lima orang sama seperti sebelum dilakukanya normalisasi. Hal tersebut membuktikan bahwa proses normalisasi 3NF memberikan hasil yang konsisten saat dilakukan uji *query*.

b. Cek data setiap mahasiswa

Selain melakukan uji konsistensi data dengan mencari jumlah suatu baris dalam kolom, kita juga dapat melakukan uji konsistensi data dengan melakukan pengecekan data pada setiap baris pada satu kolom, contohnya yaitu pada kolom mahasiswa.

1. Sebelum Normalisasi

```
USE semhas_sebelum_normalisasi;  
SELECT nama_mahasiswa,nama_dosen_pembimbing, nama_dosen_penguji  
FROM semhas_sebelum_normalisasi  
WHERE id_mahasiswa = 'M10001';
```

Tabel yang dihasilkan:

	nama_mahasiswa	nama_dosen_pembimbing	nama_dosen_penguji
▶	Rakha Wijaya	Budianto Kusuma, S.T., M.T., Alvian Putra, S.T....	Pratama Dwi Saputra, S.T., M.T., Agus Kuncoro...

Pada pengujian yang pertama kita melakukannya dengan mengecek baris dengan mahasiswa yang memiliki id “M10001”. Dari hasil uji *query* tersebut, diperoleh mahasiswa bernama “Rakha Wijaya” dengan nama dosen pembimbing “Budianto Kusuma, S.T., M.T. dan Alvian Putra, S.T., M.T.” dan nama dosen penguji “Pratama Dwi Saputra, S.T., M.T. dan Agus Kuncoro, S.T., M.T.”, tetapi pada kolom nama dosen pembimbing dan nama dosen penguji masih terdapat redundansi atau duplikat data.

2. Setelah Normalisasi

```
USE semhas_setelah_normalisasi;  
SELECT  
mahasiswa.nama_mahasiswa,  
dosen_pembimbing.nama_dosen_pembimbing,  
dosen_penguji.nama_dosen_penguji  
FROM jadwal_seminar_hasil  
JOIN relasi_jadwal_dosen_pembimbing  
ON jadwal_seminar_hasil.id_jadwal = relasi_jadwal_dosen_pembimbing.id_jadwal  
JOIN dosen_pembimbing  
ON relasi_jadwal_dosen_pembimbing.id_dosen_pembimbing =  
dosen_pembimbing.id_dosen_pembimbing  
JOIN relasi_jadwal_dosen_penguji  
ON jadwal_seminar_hasil.id_jadwal = relasi_jadwal_dosen_penguji.id_jadwal  
JOIN dosen_penguji  
ON relasi_jadwal_dosen_penguji.id_dosen_penguji = dosen_penguji.id_dosen_penguji  
JOIN mahasiswa  
ON jadwal_seminar_hasil.id_mahasiswa = mahasiswa.id_mahasiswa  
WHERE mahasiswa.id_mahasiswa = 'M10001';
```

Tabel yang dihasilkan:



	nama_mahasiswa	nama_dosen_pembimbing	nama_dosen_penguji
▶	Rakha Wijaya	Budianto Kusuma, S.T., M.T.	Pratama Dwi Saputra, S.T., M.T.
	Rakha Wijaya	Budianto Kusuma, S.T., M.T.	Agus Kuncoro, S.T., M.T.
	Rakha Wijaya	Alvian Putra, S.T., M.T.	Pratama Dwi Saputra, S.T., M.T.
	Rakha Wijaya	Alvian Putra, S.T., M.T.	Agus Kuncoro, S.T., M.T.

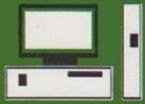
Setelah dilakukan normalisasi, data mahasiswa dengan ID M10001 memiliki isi yang sama seperti sebelum normalisasi. Perbedaannya, jumlah baris menjadi lebih banyak karena data dipisahkan ke dalam beberapa tabel untuk menghindari redundansi. Hal ini menunjukkan bahwa meskipun normalisasi dapat menambah jumlah baris, informasi yang dihasilkan tetap sama.

5. STUDI KASUS

Seorang mahasiswa tingkat akhir bernama Ersya Maulida dengan nim 2299828124 akan mendaftar seminar hasil, mahasiswa tersebut dapat mengakses aplikasi khusus untuk mendaftar seminar hasil. Proses pendaftaran seminar hasil tersebut diantaranya:

1. Pendaftaran Seminar Hasil
 - Mengisi formulir pendaftaran.
 - Mengumpulkan berkas skripsi yang telah dikerjakannya.
2. Proses Verifikasi Berkas
 - Jika berkas skripsi telah lengkap, administrasi akademik akan menyetujui berkas.
 - Status berkas berubah menjadi "Disetujui".
3. Penjadwalan Seminar Hasil
 - Administrasi akademik akan mengatur jadwal seminar hasil dengan melihat kalender dosen pembimbing dan kalender dosen penguji untuk mengetahui jadwal ketersediaan dosen untuk menghadiri seminar hasil.
 - Notifikasi jadwal seminar hasil akan dikirimkan kepada mahasiswa paling lambat H-7 sebelum dilaksanakannya seminar hasil.

Notifikasi dikirimkan maksimal H-7 sebelum dilaksanakannya seminar hasil, sehingga mahasiswa dapat melakukan persiapan dengan matang. Selain itu, tidak akan terjadi benturan jadwal antara dosen pembimbing dan dosen penguji karena sistem aplikasi tersebut memiliki kalender dosen pembimbing dan kalender dosen penguji, sehingga pihak administrasi akademik dapat mengatur jadwal ketersediaan antara dosen pembimbing dengan dosen penguji.

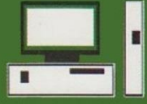


Kesimpulan

Penelitian ini merancang basis data penjadwalan seminar hasil dengan metode *Database Life Cycle* (DBLC) dan normalisasi hingga 3NF. Kebaruan penelitian ini terletak pada integrasi kalender dosen dan fitur notifikasi otomatis yang belum dikaji pada penelitian sebelumnya. Rancangan yang dihasilkan efisien, bebas redundansi, serta mampu mengurangi benturan jadwal dan keterlambatan informasi. Hasil ini menawarkan solusi praktis untuk penjadwalan seminar hasil yang lebih transparan di perguruan tinggi. Untuk penelitian selanjutnya, disarankan integrasi dengan sistem akademik yang sudah ada, pengujian dengan dataset berskala besar, serta penambahan fitur penjadwalan ulang otomatis dan analisis beban dosen.

Daftar Pustaka

- [1] Salsabila, U. H., & Agustian, N. (2021). *Peran Teknologi Pembelajaran*
- [2] Jamaludin, et al. (2022). *Sistem Basis Data*. Padang: PT Global Eksekutif Teknologi.
- [3] M. Riyan Dirgantara, S. S. (2023). *Pengenalan Database Management System (DBMS)*.
<https://doi.org/10.5281/ZENODO.8123019>
- [4] Tri Amri Wijaya, Constantin Menteng, Afis Julianto, Adi Surya, & Ema Utami. (2021). *Perancangan Desain Basis Data Sistem Informasi Geografi Tanah Penduduk Dengan Menerapkan Model Data Relasional (Studi Kasus: Desa Tumbang Mantuhe Kabupaten Gunung Mas Provinsi Kalimantan Tengah)*
Jurnal Teknologi Informasi: Jurnal Keilmuan dan Aplikasi Bidang Teknik Informatika, 15(1), 72–81. <https://doi.org/10.47111/jti.v15i1.1867>
- [5] Pradipta, R. A., Wintoro, P. B., & Budiyo, D. (2022). Perancangan Pemodelan Basis Data Sistem Informasi Secara Konseptual Dan Logikal. *Jurnal Informatika dan Teknik Elektro Terapan*, 10(2). <https://doi.org/10.23960/jitet.v10i2.2541>
- [6] Titik Rahmawati, Eka Yulia Sari, Anjasmara Tanjung Shakti, & Atthaya Nanda Yomura. (2023). *Analisis Perancangan Database Management System Untuk Sistem Penunjang Proses Bisnis Wedang Uwuh Digital*. TEKNIMEDIA: Teknologi Informasi dan Multimedia, 4(1), 61–69. <https://doi.org/10.46764/teknimedia.v4i1.104>
- [7] Adawiyah, H., Krisnanik, E., Kom, S., & Purabaya, R. H. (2022). *Analisis Dan Perancangan Basis Data Sistem Informasi Akademik Pondok Pesantren Al-Qur'an An Assa'adiyah*.
- [8] Trihapningsari, D., & Putri, M. A. (2024). *Perancangan Basis Data Sistem Reservasi Wisma Universitas Terbuka dengan Metode Database Life Cycle*. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 7(4), 1638–1646. <https://doi.org/10.32493/jtsi.v7i4.45421>
- [9] Pieris, D., Wijegunasekera, M. C., & J Dias, N. G. (2020). *ER Model Partitioning: Towards Trustworthy Automated Systems Development*, 11(6).
- [10] Azzahra, Z. F., & Anggoro, A. D. (2022). *Analisis Teknik Entity-Relationship Diagram dalam Perancangan Database Sebuah Literature Review*. *INTECH (Informatika dan Teknologi)*, 3(2), 70-74.
- [11] Rashkovits, R., & Lavy, I. (2021). *Mapping Common Errors in Entity Relationship Diagram Design of Novice Designers*. *International Journal of Database Management Systems*, 13(1), 1-19.
- [12] Blacher, M., Giesen, J., Laue, S., Klaus, J., & Leis, V. (2022). Machine learning, linear algebra, and more: Is SQL all you need?. In *CIDR*.
- [13] Mulyana, D., & Sidiq, M. (2023). *Perancangan Desain Sistem Basis Data Pengelolaan Aset (Studi Kasus: Universitas Galuh)*. *Journal of Galuh Information System*, 1(1), 17. <https://doi.org/10.25157/JSIG.V1I1.2917>
- [14] Taseska, T. (2024). *Conceptual And Logical Design Of Relational Database: A Case Study*. *Journal of Student Research*, 1(1). <https://doi.org/10.20544/>



- [15] Permana, K. E., Sophan, M. K., Muntasa, A., & Rahmat, A. B. (2023). Perbandingan Kinerja Query SQL Join Tables Dengan Menggunakan Index. Jurnal Simantec, 11(2), 241–249. <https://doi.org/10.21107/simantec.v11i2.20288>