

PENGEMBANGAN APLIKASI MANAGEMEN TUGAS SKRIPSI (STUDI KASUS : PROGRAM STUDI TEKNIK INFORMATIKA UNIVERSITAS TRUNOJOYO MADURA)

Ach. Khozaimi¹, Sigit Susanto Putro², Mujibur Rohman³

^{1,2,3}Teknik Informatika, Fakultas Teknik, Universitas Trunojoyo Madura Bangkalan, Indonesia

¹Email : khozaimi@gmail.com, ²Email : sigitida.06@gmail.com ,

³Email : mujibur1336@gmail.com

ABSTRAK

Skripsi merupakan hal yang penting dan selalu ada di setiap universitas. Skripsi juga sebagai wujud hasil karya tulis ilmiah mahasiswa yang tercatat dan tersusun rapi sesuai dengan kaidah yang ditentukan. Saat ini permasalahan yang timbul dari proses pengajuan skripsi di lingkungan Universitas Trunojoyo Madura secara umum adalah proses pencarian serta penentuan sebuah judul skripsi yang tepat untuk diajukan mahasiswa dan tidak tersedianya data hasil skripsi mahasiswa yang tersusun secara sistematis dan komputerisasi, sehingga dalam pembuatan skripsi terdapat banyak kesamaan baik berupa judul maupun metode yang sering digunakan oleh mahasiswa, hal ini berdampak ketidakberagaman hasil skripsi buatan mahasiswa. Dengan adanya pengembangan aplikasi manajemen skripsi menggunakan framework codeigniter dan cosine similarity diharapkan bisa mengevaluasi, memonitoring, serta memperbaiki kinerja sistem yang manual menjadi sistem terkomputasi sekaligus meminimalisir kesamaan dan plagiatisme serta dapat meningkatkan pelayanan kepada mahasiswa yang mengajukan skripsi dengan mudah sesuai dengan aturan yang berlaku di lingkungan Universitas Trunojoyo Madura.

Kata Kunci : Skripsi, Framework Codeigniter , Cosine Similarity

ABSTRACT

Thesis is an important thing and always exist in a university. The thesis also as a manifestation of the results of scientific papers of students registered and organized in accordance with the prescribed rules. This time the problems arising from the submission of the thesis at the University of Trunojoyo in general is a process of finding and determining a thesis titles appropriate to ask students and unavailability of data from student thesis systematically arranged and computerized, so that in the manufacture of paper there are many similarities, both in the form of titles and methods that are often used by students, this affects the uniformity of student-made thesis results. With the development of thesis management application using code igniter and cosine similarity framework is expected to evaluate, monitor, and improve the performance of the manual system into a computation system while minimizing similarity and plagiarism and can improve services to students who submit a thesis easily in accordance with the rules applicable in the University of Trunojoyo Madura.

Key word : Thesis, Code igniter, Framework, Cosine Similarity.

I. PENDAHULUAN

Skripsi merupakan karya tulis ilmiah yang disusun dan dibuat oleh mahasiswa menggunakan metode penelitian yang baik dan benar untuk memenuhi persyaratan memperoleh gelar Sarjana dari universitas tempat studi [1][2]. Untuk mendapatkan skripsi yang sesuai dengan aturan dan karakteristik yang diberlakukan, maka diperlukan proses yang ketat, bertahap, terevaluasi dan selalu termonitoring dengan baik, baik monitoring yang dilakukan oleh dosen pembimbing ataupun oleh kepala

prodi. Oleh karena itu, dalam menyusun skripsi mahasiswa harus dibimbing oleh satu sampai dua orang pembimbing yang berstatus sebagai dosen pada perguruan tinggi tempat mahasiswa menempuh studi. Permasalahan muncul ketika pelaksanaan administrasi skripsi tidak terintegrasi antar proses yang satu dengan proses yang lain. Beberapa contoh permasalahan terkait proses pengerjaan skripsi meliputi: informasi dosen pembimbing dan mahasiswa yang dibimbingnya, kesamaan judul skripsi, jadwal ujian proposal, jadwal ujian akhir atau sidang skripsi dan persoalan lain [3]. Permasalahan permasalahan tersebut

dapat diminimalisir bahkan diatasi jika ada sebuah sistem yang dapat memenej proses admistrasi pengerjaan skripsi setiap mahasiswa. Seiring dengan berjalannya waktu sudah saatnya Universitas Trunojoyo Madura memiliki sistem pendataan skripsi mahasiswa yang terintegrasi antar proses baik dosen pembimbing, dosen penguji, dan proses penjadwalan kegiatan skripsi tersebut. Dengan menggunakan metode komputasi berupa cosine similarity dapat mengurangi tingkat kesamaan judul maupun metode yang sering digunakan oleh mahasiswa.

Dengan melihat permasalahan yang ada di Universitas Trunojoyo Madura khususnya fakultas teknik program studi teknik informatika perlu adanya pengembangan aplikasi manajemen skripsi yang dalam hal ini menggunakan framework codeigniter dan cosine similarity sehingga termonitoring dan sebagai salah satu metode yang dapat diterapkan untuk membandingkan dokumen teks pada judul dan abstrak skripsi yang diajukan oleh mahasiswa.

A. Framework

Framework merupakan sekumpulan fungsi-fungsi atau prosedur dan class-class yang dibuat untuk tujuan tertentu dan siap digunakan untuk tujuan yang telah ditentukan tersebut[4][5], penggunaan framework akan memudahkan dan mempercepat developer aplikasi dalam membangun sebuah aplikasi tanpa harus membuat fungsi, prosedur atau class dari awal.

Berikut beberapa alasan kenapa framework penting digunakan dalam membangun aplikasi[6]:

- Framework dapat mempermudah programmer dalam membangun sebuah aplikasi.
- Framework biasanya hanya menyediakan fasilitas-fasilitas, fungsi, prosedur atau class yang umum dipakai.
- Proses pemeliharaan aplikasi yang dibangun menggunakan framework akan lebih mudah.

B. CodeIgniter

Code Igniter[5] adalah aplikasi open source, code igniter merupakan framework PHP yang dibangun menggunakan konsep MVC (Model, View, Controller)[7], code igniter digunakan untuk membangun website yang dinamis dengan menggunakan PHP.

CodeIgniter mempermudah dan mempercepat developer untuk membuat aplikasi web karena tidak harus membangun dari awal. CodeIgniter dirilis pertama kali pada 28 Februari 2006. Versi stabil terakhir adalah versi 3.0.4[4].

C. Metode Cosine Similarity

Cosine Similarity[5] adalah salah satu metode yang dapat digunakan untuk membandingkan kemiripan dokumen teks dengan dokumen teks yang lain. Metode *Cosine Similarity* menggunakan dua vector sebagai parameternya. Vektor tersebut didasarkan pada jumlah kata yang mirip pada kedua dokumen teks yang dibandingkan. Cosine Similarity hanya membandingkan dokumen secara tekstual, sehingga cosine similarity tidak dapat makna dari dokumen yang dibandingkan[8]. Metode cosine similarity memiliki akurasi yang sangat baik dalam melakukan pengecekan dokumen[9][10]. Untuk notasi himpunan dapat digunakan rumus berikut:

$$\text{Similarity}(X, Y) = \frac{|X \cap Y|}{|X|^{1/2} |Y|^{1/2}}$$

Dimana $|X \cap Y|$ adalah jumlah term yang ada pada dokumen X dan yang ada pada dokumen Y, $|X|$ adalah jumlah term yang ada pada dokumen X dan $|Y|$ adalah jumlah term yang ada pada dokumen Y. Dari notasi himpunan diatas dapat dibuat persamaan matematika sebagai berikut :

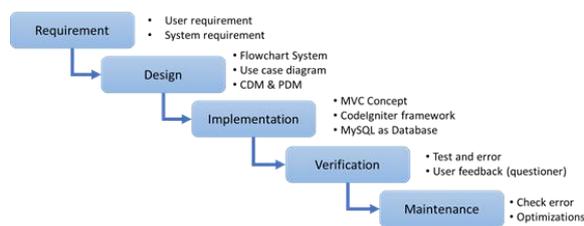
$$\text{Similarity}(X, Y) = \frac{\sum_{i=1}^i X_i Y_i}{\sqrt{\sum_{i=1}^i X_i^2 \sum_{i=1}^i Y_i^2}}$$

Dimana x dan y adalah dokumen yang berbeda, xi adalah term i yang ada di dokumen x, yi adalah term i yang ada pada dokumen y, sedangkan proses nilai cosines berdasarkan kata kunci pada tiap-tiap dokumen. Ketika dua dokumen sama, sudutnya adalah nol derajat (0°) dan kesamaannya adalah satu (1), dan ketika dua dokumen tidak sama sudutnya adalah 90 derajat (90°) dan kesamaannya adalah nol (0)[11].

II. METODOLOGI

Metode penelitian ini menggunakan siklus SDLC (Software Development Life Cycle) dengan model waterfall. Model SDLC melakukan pendekatan secara sistematis dan urutan mulai level analisis sistem lalu menuju ke tahap desain sistem, implementasi sistem, uji coba, dan pemeliharaan[12][13]. Tahapan demi

tahapan dilakukan secara berurutan sehingga model ini disebut dengan waterfall dan tahapan-tahapan tersebut harus tertib.



Gambar 1. Model Water fall

A. Requirement

Analisa kebutuhan sistem ini dibagi menjadi dua bagian diantaranya kebutuhan fungsional dan kebutuhan non fungsional. Untuk analisis kebutuhan fungsional lebih difokuskan pada fitur-fitur sistem yang berguna untuk mengatasi permasalahan. Sedangkan untuk analisis kebutuhan non fungsional lebih difokuskan pada perangkat keras maupun perangkat lunak sebagai penunjang penggunaan sistem.

B. Desain Sistem

Proses ini digunakan untuk mengubah kebutuhan-kebutuhan yang sudah dianalisa pada tahapan sebelumnya untuk dijadikan representasi ke dalam bentuk blueprint software sebelum implementasi di mulai.

C. Implementasi Sistem

Selanjutnya setelah proses desain telah dilakukan maka kemudian adalah proses pengubahan bentuk sehingga dapat dimenerti oleh mesin komputer, proses pengubahan tersebut adalah diubah ke dalam bahasa pemrograman melalui proses coding pada bahasa pemrograman tertentu dan dalam hal ini menggunakan bahasa pemrograman PHP.

D. Uji Coba

✓ Pengujian Fungsionalitas

Setelah proses implementasi selesai, selanjutnya adalah proses pengujian, karena sebuah aplikasi yang baik haruslah bisa dioperasikan dan diuji coba. Demikian juga dengan sistem ini, semua fungsi-fungsi yang sudah dirancang dan diimplementasikan harus dicoba dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya.

✓ Kuisisioner Penggunaan Aplikasi

Setelah melakukan uji fungsionalitas sistem, langkah selanjutnya adalah penyebaran kuisisioner untuk mengetahui seberapa efektif sistem ini diterapkan.

E. Pemeliharaan

Pemeliharaan suatu aplikasi mutlak diperlukan selama aplikasi tersebut masih aktif dan digunakan, karena proses pemeliharaan termasuk bagian dari pengembangan aplikasi, karena sistem yang dibuat tidak selamanya hanya seperti itu. Ketika aplikasi digunakan biasanya akan ditemukan bug atau error pada aplikasi yang dibangun, atau ada penambahan fitur-fitur atau fungsionalitas lain yang belum ada pada aplikasi tersebut sebelumnya.

III. HASIL DAN PEMBAHASAN

A. Implementasi

Sistem manajemen skripsi ini dibuat menggunakan Framework Codeigniter dengan konsep MVC (Model, View, Controller) dan metode cosine similarity. Beberapa potongan source code dapat dilihat dibawah ini.

✓ Model

Model adalah bagian yang memiliki tugas untuk memanipulasi data yang ada pada database seperti melakukan insert, update, delete, search dan query lain yang ada pada database, model juga digunakan untuk menangani validasi dari bagian controller, dan model tidak dapat dihubungkan langsung dengan bagian view. Beberapa fungsi yang ada pada model meliputi: ambil data, tambah data, ubah data, hapus data, dan fungsi lihat detail data.

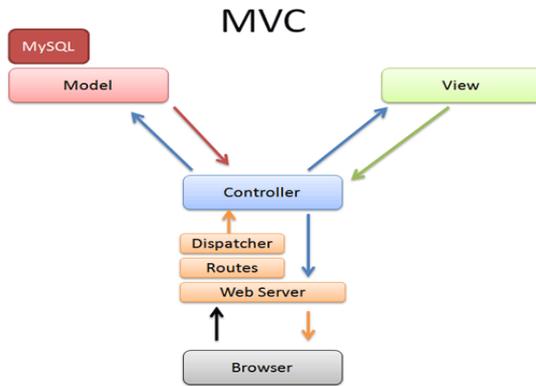
✓ View

View, bagian ini merupakan bagian yang digunakan menangani presentation logic. Pada suatu aplikasi web, view dapat berupa file HTML, JavaScript dan CSS dan diatur oleh controller. View berfungsi untuk menerima data dan menampilkan data tersebut kepada user.

✓ Controller

Controller, bagian ini bertugas untuk mengatur hubungan antara bagian model dan bagian view, controller berfungsi untuk menerima request dan data dari user

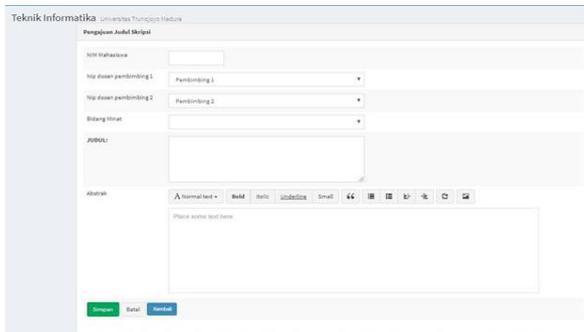
kemudian menentukan apa yang akan diproses oleh aplikasi. Berikut merupakan beberapa fungsi yang berada di controller pada aplikasi ini yaitu: fungsi index, detail, hapus, tambah dan ubah data, dan juga fungsi *cosine similarity* berada pada bagian controller ini.



Gambar 2. Diagram hubungan MVC

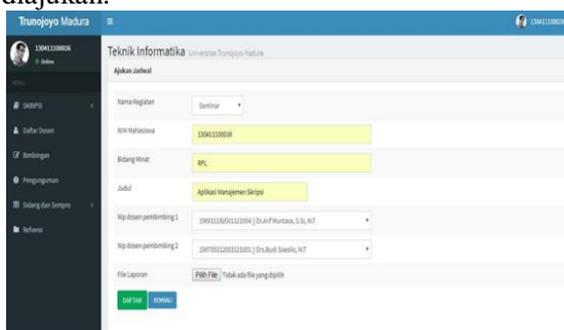
B. Hasil

Hasil dari implementasi aplikasi ini dapat dilihat dari beberapa gambar berikut :



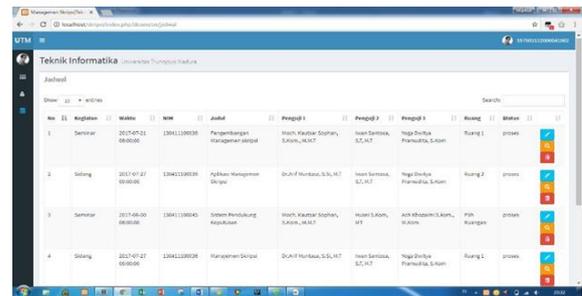
Gambar 3. Form pendaftaran judul skripsi

Pada gambar 3 terlihat tampilan untuk form pendaftaran bagi mahasiswa yang akan mengajukan judul skripsi, pada form ini nanti akan dilakukan pengecekan kemiripan judul dan abstrak dari judul skripsi yang akan diajukan.



Gambar 4. Form Pendaftaran Seminar Dan Sidang

Setelah proses pengajuan judul skripsi dinyatakan tidak memiliki kemiripan ataupun plagiasi terhadap data-data skripsi yang sudah ada sebelumnya, selanjutnya mahasiswa akan diminta untuk menseminarkan proposal yang telah disetujui tersebut, gambar 4 menunjukkan form pendaftaran seminar proposal dan form ini juga dapat digunakan untuk pendaftaran seminar hasil atau sidang skripsi.



Gambar 5. Pembuatan jadwal seminar dan sidang skripsi

Informasi jadwal seminar dan jadwal sidang mahasiswa beserta data para pembimbing dan penguji juga dapat dilihat secara online setelah mahasiswa melakukan pendaftaran seminar atau sidang, tampilan akan terlihat seperti pada gambar 5.

C. Pengujian

Tahap pengujian merupakan tahap terakhir dalam serangkaian pembuatan sistem, pengujian ini dilakukan terhadap sistem yang telah dibuat. Pengujian sistem meliputi uji fungsionalitas dan kuisisioner pengguna aplikasi

✓ Uji Fungsionalitas

Tahap uji fungsionalitas terhadap sistem dilakukan dengan tujuan untuk dapat mengetahui atau memastikan bahwa program dapat berjalan dengan baik. Proses pengujian merupakan proses mengevaluasi suatu sistem untuk menentukan apakah program dari tahap pengembangan telah sesuai. Ceklist pengujian dapat dilihat pada lampiran Fungsionalitas setiap user dari aplikasi ini.

✓ Uji User Experience

Setelah melakukan serangkaian uji coba dengan melakukan uji fungsionalitas sistem tiap user. Maka yang selanjutnya dilakukan adalah penyebaran kuisisioner untuk mengetahui seberapa efektif sistem ini diterapkan. Penyebaran kuisisioner ini ditujukan kepada responden disekitar lokasi penelitian ini berlangsung. Contoh kuisisioner bisa di lihat pada Lampiran. Penilaian yang diberikan dalam

kuisisioner diolah lebih lanjut dengan tahapan proses sebagai berikut:

1. Menghitung nilai rata-rata untuk setiap variabel
2. Menghitung nilai rata-rata dari semua variabel
3. Mengelompokkan hasil perhitungan ke dalam lima kategori, yaitu:
 - a. Sangat efektif, jika nilai rata-rata $\geq 8,5$
 - b. Efektif, jika nilai rata-rata $7,5 \leq$ nilai rata-rata $\leq 8,4$
 - c. Cukup efektif, jika nilai rata-rata $6 \leq$ nilai rata-rata $\leq 7,4$
 - d. Kurang efektif, jika nilai rata-rata $5 \leq$ nilai rata-rata $\leq 5,9$
 - e. Tidak efektif, jika nilai rata-rata < 5

Perhitungan nilai rata-rata tiap pertanyaan pada kuisisioner:

1. Setelah anda menggunakan sistem ini, apakah anda merasa nyaman dalam segi tampilan dan fasilitas yang di sediakan = 8,7
2. Apakah anda merasa terbantu dalam menggunakan sistem ini untuk mengecek judul skripsi yang ingin anda ajukan = 9
3. Apakah penggunaan sistem ini menghemat waktu dalam mengakses informasi mengenai judul, dosen, jadwal = 8,6
4. Apakah anda merasa puas dengan fitur-fitur yang disajikan dalam sistem ini = 8,6
5. Apakah dengan penggunaan sistem ini anda merasa terbantu dalam hal pendaftaran, melihat daftar dosen, dan jadwal = 9

Hasil akhir nilai rata-rata = $(8,7 + 9 + 8,6 + 8,6 + 9) / 5 = 8,78$

Dari hasil perhitungan kuisisioner diatas dapat disimpulkan bahwa Aplikasi Manajemen Skripsi masuk pada kategori cukup efektif untuk variabel penilaian seberapa efektif sistem ini diterapkan.

IV. SIMPULAN DAN SARAN

Dari pembuatan Aplikasi Manajemen Skripsi ini dapat diambil kesimpulan bahwa sistem ini terbilang cukup efektif untuk membantu mahasiswa dalam melakukan aktifitas skripsi. Hal itu dibuktikan dengan hasil uji coba yang menunjukkan penilaian rata-rata sebesar 8,82 dari nilai range antara 1 sampai 10 dan masuk dalam kategori cukup efektif.

Dalam pembuatan sistem ini masih memiliki kekurangan yaitu pada proses perhitungan metode *cosine similarity* memakan waktu yang lama, maka perlu dilakukan pengembangan selanjutnya untuk mendapatkan sistem yang lebih baik lagi. Adapun saran untuk pengembangan berikutnya yaitu dapat menggunakan metode kimiripan dokumen yang lebih cepat dalam proses komputasi.

V. UCAPAN TERIMA KASIH

Terima kasih kami sampaikan kepada segenap pihak telah memberikan dukungan kepada penulis dalam menyelesaikan penelitian ini. Dekan Fakultas Teknik Universitas Trunojoyo Madura, Prof. Dr. Rachmad Hidayat, M.T. Ketua Jurusan Teknik Informatika UTM, Koordinator Program Studi Teknik Informatika UTM.

REFERENSI

- [1] A. N. ; Berlianawati, "Rancang Bangun Sistem Informasi Tugas Akhir Jurusan Teknik Informatika Fakultas Teknik Unijoyo Berbasis Web," Bangkalan, 2008.
- [2] Y. Yamasari, Asmunin, and I. F. Kurniawan, "Pengembangan Sistem Informasi Manajemen Skripsi Di Jurusan Teknik Elektro, Ft Unesa," *J. Manaj. Inform.*, vol. 04, no. 01, pp. 1–10, 2015.
- [3] A. Khozaimi, Ach.; Solihin, Firdaus; Jauhari, "PERANCANGAN DAN PEMBUATAN SISTEM INFORMASI TUGAS AKHIR (SIMTAK)," *J. Simantec*, vol. 1, no. 3, pp. 203–211, 2010.
- [4] IDCloudHost, "Mengenal Apa Itu Framework Beserta Jenisnya," *IdCloudHost*, 2017. .
- [5] M. Destiningrum and Q. J. Adrian, "SISTEM INFORMASI PENJADWALAN DOKTER BERBASIS WEB DENGAN MENGGUNAKAN FRAMEWORK CODEIGNITER," *TEKNOINFO*, vol. 11, no. 2, pp. 30–37, 2017.
- [6] R. M. Achmad, "Tutorial Codeigniter Bahasa Indonesia Lengkap Plus Source Code," *My Notes Code*, 2018. .
- [7] A. R. Dayat and L. Angriani, "Pemanfaatan Model-View-Controller (MVC) Dalam Rancang Bangun Sistem

- Informasi Rakornas Aptikom 2017,” in *Seminar Nasional APTIKOM*, 2017, no. November, pp. 416–420.
- [8] Maskur and F. R. Andriansyah, “Implementasi Web Semantik Untuk Aplikasi Pencarian,” *J. Ilm. NERO*, vol. 2, no. 1, pp. 11–18, 2015.
- [9] S. Sugiyanto, B. Surarso, and A. Sugiharto, “Analisa Performa Metode Cosine Dan Jacard Pada Pengujian Kesamaan Dokumen,” *J. Masy. Inform.*, vol. 5, no. 10, pp. 1–8, 2016.
- [10] D. S. Nugroho, Y. F. A, and W. Maharani, “Analisis Dan Implementasi Perbandingan Metode Cosine Similarity Dan Correlation Based Similarity Pada Recomender System Berbasis Item-Based Collaborative Filtering,” 2010.
- [11] R. V. Imbar, Adelia, and A. Rehatta, “Implementasi Cosine Similarity dan Algoritma Smith-Waterman untuk Mendeteksi Kemiripan Teks,” *J. Inform.*, 2014.
- [12] Y. ; U. Firmansyah, “Penerapan Metode SDLC Waterfall Dalam Pembuatan Sistem Informasi Akademik Berbasis Web Studi Kasus Pondok Pesantren Al-Habi Sholeh Kabupaten Kubu Raya, Kalimantan Barat,” *J. Teknol. Manaj. Inform.*, vol. 4, no. 1, 2018.
- [13] A. Martias, “Analisa Kecukupan Penerapan Pengawasan Internal Dengan Metode System Development Life Cycle PT. XYZ,” *Moneter*, vol. 5, no. 2, pp. 23–32, 2018.

Lampiran: Tabel Fungsionalitas

Tabel. 1 fungsional User Admin

NO	DESKRIPSI	INPUT	PROSES	OUTPUT
1	LOGIN	Username dan password benar	Pengecekan database, tabel user	Menampilkan Halaman Admin
		Username dan password salah		Menampilkan Halaman Login kembali
2	Tambah data user	Nama, Password, dan status user	Menambahkan data di tabel user	Menampilkan halaman master data user
3	Ubah data user		Mengubah data di tabel user	
4	Hapus data user		Menghapus data di tabel user	
5	Tambah data dosen	Nama, Nip, Alamat, No.hp dosen	Menambahkan data di tabel dosen	Menampilkan halaman master data dosen
6	Ubah data dosen		Mengubah data di tabel dosen	
7	Hapus data dosen		Menghapus data di tabel dosen	
8	Tambah data mahasiswa	Nim, Nama, Alamat Mahasiswa	Menambahkan data di tabel mahasiswa	Menampilkan halaman master data mahasiswa
9	Ubah data mahasiswa		Mengubah data di tabel mahasiswa	
10	Hapus data mahasiswa		Menghapus data di tabel mahasiswa	
11	Tambah data ruangan	Nama ruangan sidang dan seminar	Menambahkan data di tabel ruangan	Menampilkan halaman data master ruangan
12	Ubah data ruangan		Mengubah data di tabel ruangan	
13	Hapus data ruangan		Menghapus data di tabel ruangan	
14	Tambah data bidang minat	Nama bidang minat	Menambahkan data di tabel bidang minat	Menampilkan halaman data master bidang minat
15	Ubah data bidang minat		Mengubah data di tabel bidang minat	
16	Hapus data bidang minat		Menghapus data di tabel bidang minat	
17	Tambah data level user	Nama level user	Menambahkan data di tabel level user	Menampilkan halaman data master level user
18	Ubah data level user		Mengubah data di tabel level user	
19	Hapus data level user		Menghapus data di tabel level user	
20	Mengubah data pengajuan judul skripsi	status, jadwal.	Mengubah data di table pengajuan	Menampilkan halaman data master judul dan abstrak
21	Mengubah data pengajuan	Dosen Penguji 1,	Mengubah data di	

	jadwal sidang dan seminar skripsi	Dosen Penguji 2, Dosen Penguji 3, status, jadwal.	table jadwal	Menampilkan halaman data master jadwal	
22	Tambah data Notifikasi	Judul, Nama, File	Menambahkan data di tabel notifikasi	Menampilkan halaman data master notifikasi	
23	Ubah data Notifikasi		Mengubah data di tabel notifikasi		
24	Hapus data Notifikasi		Menghapus data di tabel notifikasi		
25	Lihat data dosen pembimbing	-	Mengambil data di tabel dosen	Menampilkan halaman detail dosen pembimbing	
26	Lihat data mahasiswa	-	Mengambil data di tabel mahasiswa	Menampilkan halaman detail mahasiswa	
27	Lihat data pengajuan judul skripsi	-	Mengambil data di tabel pengajuan judul skripsi	Menampilkan halaman detail pengajuan judul skripsi	
28	Lihat data pengajuan jadwal seminar dan sidang skripsi	-	Mengambil data di tabel jadwal	Menampilkan halaman detail jadwal	
29	Tambah data abstrak	Judul, Abstrak, Nim, Nip Dosen	Menambah data pada tabel abstrak	Menampilkan halaman abstrak	
30	Hapus data abstrak		Menghapus data pada tabel abstrak		

Tabel. 2 fungsionalitas dosen/Pembimbing

NO	DESKRIPSI	INPUT	PROSES	OUTPUT	O / X
1	Menampilkan data mahasiswa bimbingan	-	Mengambil data pada tabel mahasiswa	Menampilkan halaman detail mahasiswa	
2	Menampilkan data bimbingan mahasiswa	-	Mengambil data pada tabel bimbingan	Menampilkan halaman detail bimbingan	
3	Menampilkan data jadwal seminar dan sidang skripsi	-	Menampilkan data pada tabel jadwal	Menampilkan halaman detail jadwal	
4	Mengubah data bimbingan mahasiswa	Status bimbingan	Mengubah data pada tabel bimbingan	Menampilkan data master bimbingan	

Tabel 3 Fungsionalitas Koordinator Skripsi

NO	DESKRIPSI	INPUT	PROSES	OUTPUT	O / X
1	Mengubah data pengajuan judul skripsi	status, jadwal.	Mengubah data di table pengajuan	Menampilkan halaman data master judul dan abstrak	
2	Mengubah data pengajuan jadwal sidang dan seminar skripsi	Dosen Penguji 1, Dosen Penguji 2, Dosen Penguji	Mengubah data di table jadwal	Menampilkan halaman data master jadwal	

		3, status, jadwal.			
3	Lihat data dosen pembimbing	-	Mengambil data di tabel dosen	Menampilkan halaman detail dosen pembimbing	

Tabel 4. fungsionalitas mahasiswa

NO	DESKRIPSI	INPUT	PROSES	OUTPUT	O / X
1	Lihat data dosen pembimbing	-	Mengambil data di tabel dosen	Menampilkan halaman detail dosen pembimbing	
2	Lihat data pengajuan judul skripsi	-	Mengambil data di tabel pengajuan judul skripsi	Menampilkan halaman detail pengajuan judul skripsi	
3	Lihat data pengajuan jadwal seminar dan sidang skripsi	-	Mengambil data di tabel jadwal	Menampilkan halaman detail jadwal	
4	Menampilkan data bimbingan mahasiswa	-	Mengambil data pada tabel bimbingan	Menampilkan halaman detail bimbingan	
5	Tambah data pengajuan judul skripsi	Nim, bidang minat, judul, abstrak, dosen pembimbing 1, dosen pembimbing 2	Menambahkan data pengajuan judul skripsi	Menampilkan halaman data pada tabel pengajuan judul skripsi	
6	Tambah data pengajuan jadwal sidang dan seminar skripsi	Nim, bidang minat, judul, dosen pembimbing 1, dosen pembimbing 2	Menambahkan data jadwal sidang dan seminar skripsi	Menampilkan halaman data pada tabel jadwal	
7	Lihat data pengumuman	-	Mengambil data dari tabel pengumuman	Menampilkan halaman data pada tabel pengumuman	