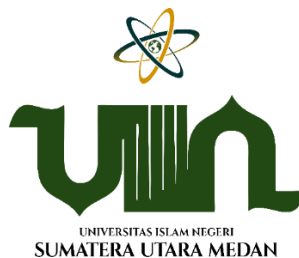


**PENELITIAN DASAR PROGRAM STUDI
NO. REGISTRASI: 22115000057988**

**LAPORAN PENELITIAN
PENERAPAN TEXT MINING PADA SISTEM PENYELEKSIAN JUDUL
SKRIPSI MAHASISWA MENGGUNAKAN ALGORITMA LATENT
DIRICHLET ALLOCATION DI PROGRAM STUDI ILMU KOMPUTER
UIN SUMATERA UTARA MEDAN**



Peneliti:

Rakhmat Kurniawan. R, M.Kom (Ketua)

Ilka Zufria, M.Kom (Anggota)

Rizky Pratama Putra (Anggota)

LEMABAGA PENELITIAN DAN PENGABDIAN KEPADA MASYARAKAT

(LP2M)

UNIVERSITAS ISLAM NEGERI SUMATERA UTARA MEDAN

2022

LEMBAR PENGESAHAN

1. a. Judul Penelitian : Penerapan Text Mining Pada Sistem Penyeleksian Judul Skripsi Mahasiswa Menggunakan Algoritma LDA (*Latent Dirichlet Allocation*) di Program Studi Ilmu Komputer UIN Sumatera Utara Medan
- b. Kluster Penelitian : Penelitian Dasar Program Studi
- c. Bidang Keilmuan : Ilmu Komputer
- d. Kategori : Kelompok
2. Peneliti : Rakhmat Kurniawan. R, M.Kom
3. ID Peneliti : 201603850113000
4. Peneliti : Ilka Zufria, M.Kom
5. ID Peneliti : 010407840510000
7. NIM : 0701172060
8. Peneliti : Rizky Pratama Putra
9. Unit Kerja : Program Studi Ilmu Komputer – Fakultas Sains dan Teknologi UIN SU Medan
10. Waktu Penelitian : 4 Bulan
11. Lokasi Penelitian : Program Studi Ilmu Komputer – Fakultas Sains dan Teknologi UIN SU Medan
12. Biaya Penelitian : Rp. 40.000.000,- (*Empat puluh juta rupiah*)

Medan, Oktober 2022

Disahkan oleh:
Lembaga Penelitian dan Pengabdian
kepada Masyarakat (LP2M) UIN SU
Medan

Ketua Peneliti,



Dr. Hasan Sazai, MA
NIP. 197602222007011018

A handwritten signature in black ink, corresponding to the name Rakhmat Kurniawan.

Rakhmat Kurniawan. R, M.Kom
NIP. 198503162015031003

SURAT PERNYATAAN BEBAS PLAGIASI

Yang bertanda tangan dibawah ini:

Nama : Rakhmat Kurniawan. R, M.Kom
Jabatan : Sekretaris Program Studi Ilmu Komputer
Unit Kerja : Fakultas Sains dan Teknologi UIN SU Medan
Alamat : Jl. Lapangan Golf, Desa Durian Jangak, Kec. Pancur
Batu, Kab. Deli Serdang, Prov. Sumatera Utara, Kode
Pos 20353

Dengan ini menyatakan bahwa:

1. Judul Penelitian **“Penerapan Text Mining Pada Sistem Penyeleksian Judul Skripsi Mahasiswa Menggunakan Algoritma LDA (*Latent Dirichlet Allocation*) di Program Studi Ilmu Komputer UIN Sumatera Utara Medan”** merupakan karya orisinal saya.
2. Jika dikemudian hari ditemukan fakta bahwa judul, hasil atau bagian dari laporan Penelitian saya merupakan karya orang lain dan/atau plagiasi, maka saya akan bertanggung jawab untuk mengembalikan 100% dana hibah penelitian yang telah saya terima, dan siap mendapatkan sanksi sesuai ketentuan yang berlaku.

Demikian pernyataan ini dibuat untuk digunakan sebagaimana mestinya.

Medan, Oktober 2022

Yang menyatakan,



Rakhmat Kurniawan. R, M.Kom
NIP. 198503162015031003

Kata Pengantar

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Alhamdulillahirabbil 'alamin, segala puji dan syukur peneliti ucapkan atas kehadiran Allah Subhanahu Wa Ta'ala yang telah memberikan rahmat dan karunia-Nya kepada peneliti sehingga peneliti diberikan kesempatan untuk mampu menyelesaikan laporan penelitian ini dengan judul **“Penerapan Text Mining Pada Sistem Penyeleksian Judul Skripsi Mahasiswa Menggunakan Algoritma LDA (*Latent Dirichlet Allocation*) di Program Studi Ilmu Komputer UIN Sumatera Utara Medan”**

Dalam menyelesaikan laporan penelitian ini banyak bantuan dan bimbingan dari berbagai pihak, baik berupa materil, spiritual, maupun informasi. Sehingga laporan ini dapat diselesaikan. Maka selayaknya peneliti dan tim peneliti mengucapkan terima kasih kepada:

1. Bapak Prof. Dr. Mhd. Syahnan, MA selaku Dekan Fakultas Sains dan Teknologi UIN Sumatera Utara Medan
2. Bapak Dr. Abdul Halim Daulay, M.Si selaku Wakil Dekan Bidang Akademik dan Kelembagaan
3. Bapak Dr. Hasan Sazali, MA, selaku Ketua Lembaga Penelitian dan Pengabdian Masyarakat (LP2M) UIN Sumatera Utara Medan
4. Bapak/Ibu rekan-rekan Dosen tetap Fakultas Sains dan Teknologi UIN Sumatera Utara Medan, khususnya Program Studi Ilmu Komputer.

Penerapan *Text Mining* Pada Sistem Penyeleksian Judul Skripsi Mahasiswa Menggunakan Algoritma LDA (*Latent Dirichlet Allocation*) di Program Studi Ilmu Komputer UIN Sumatera Utara Medan merupakan suatu sistem yang dibangun untuk mempermudah dan mempercepat tahapan

pengajuan judul proposal skripsi yang diajukan oleh mahasiswa. Dengan menggunakan sistem ini, mahasiswa dapat mengetahui persentase kesamaan topik dan potensi kemungkinan diterimanya proposal yang diajukan dalam waktu yang sangat singkat.

Peneliti dan tim peneliti menyadari bahwa masih banyak kekurangan dalam penelitian ini. Untuk itu, peneliti dan tim peneliti meminta maaf dan meminta saran agar penelitian ini dapat dilanjutkan dan menjadi lebih baik. Semoga laporan penelitian ini dapat memberikan manfaat bagi kita semua, khususnya bagi peneliti sendiri.

Medan, Oktober 2022
Penulis

A handwritten signature in black ink, consisting of stylized, overlapping loops and a long horizontal stroke extending to the right.

Rakhmat Kurniawan. R, M.Kom
NIP. 198503162015031003

Daftar Isi

LEMBAR PENGESAHAN	i
SURAT PERNYATAAN BEBAS PLAGIASI	ii
Kata Pengantar.....	iii
Daftar Isi	v
Daftar Gambar	vii
Daftar Tabel.....	ix
ABSTRAK.....	x
BAB I.....	1
A. Latar Belakang.....	1
B. Permasalahan	2
C. Tujuan dan Manfaat.....	3
D. Signifikansi	3
E. Sistematika Penulisan	4
BAB II	5
A. Kajian Teori.....	5
1. <i>Machine Learning</i>	5
2. <i>Text Mining</i>	6
3. <i>Text Pre-Processing</i>	6
4. Pembobotan TF-IDF	9
5. Topic Modeling.....	10
6. Latent Dirichlet Allocation	11
7. <i>Topic Coherence</i>	12
8. <i>Flowchart Diagram</i>	13
9. Python	14
B. Penelitian Terdahulu.....	15

BAB III.....	17
A. Jenis Penelitian	17
B. Pendekatan Penelitian.....	17
C. Pengumpulan Data.....	18
1. Preprocessing Data.....	19
2. Pembobotan TF-IDF	20
3. Topic Modeling.....	20
BAB IV.....	21
A. Representasi Data	21
B. Hasil Analisis Data	21
1. Cleaning	22
2. Case Folding	23
3. Tokenizing	23
4. Stopword Removal.....	24
5. Stemming.....	25
6. Pembobotan TF – IDF	26
C. Perancangan User Interface	38
D. Implementasi Sistem.....	51
E. Penerapan.....	69
BAB V	70
A. Kesimpulan	70
B. Saran	70
Daftar Pustaka.....	71
SOURCE CODE APLIKASI	74

Daftar Gambar

Gambar 2.1 Gambar 1. Latent Dirichlet Allocation	12
Gambar 3. 1 Tahapan Preprocessing	19
Gambar 4. 1 Perancangan Halaman Login.....	38
Gambar 4. 2 Perancangan Halaman Dashboard	39
Gambar 4. 3 Perancangan Halaman Kategori (Admin).....	39
Gambar 4. 4 Perancangan Halaman New Kategori (Admin).....	40
Gambar 4. 5 Perancangan Halaman Edit Kategori (Admin)	41
Gambar 4. 6 Perancangan Halaman Mahasiswa (Admin).....	41
Gambar 4. 7 Perancangan Halaman Add New Mahasiswa (Admin)	42
Gambar 4. 8 Perancangan Halaman Edit Mahasiswa (Admin)	42
Gambar 4. 9 Perancangan Halaman Kategori (Admin).....	43
Gambar 4. 10 Perancangan Halaman Detail Kategori (Admin).....	44
Gambar 4. 11 Perancangan Halaman Add New Dataset (Admin)	44
Gambar 4. 12 Perancangan Halaman Detail Dataset (Admin).....	44
Gambar 4. 13 Perancangan Halaman Training (Admin)	45
Gambar 4. 14 Perancangan Halaman Proposal Mahasiswa (Admin).....	45
Gambar 4. 15 Halaman Detail Proposal Mahasiswa (Admin)	46
Gambar 4. 16 Halaman Add New Proposal (Admin).....	46
Gambar 4. 17 Perancangan Halaman Admin Account (Admin)	47
Gambar 4. 18 Perancangan Halaman Add New Admin	47
Gambar 4. 19 Perancangan Halaman Edit Admin.....	48
Gambar 4. 20 Perancangan Dashboard (Mahasiswa).....	48
Gambar 4. 21 Perancangan Halaman Detail Proposal (Mahasiswa)	49
Gambar 4. 22 Perancangan Add New Proposal (Mahasiswa).....	49
Gambar 4. 23 Perancangan Halaman Detail Proposal (Mahasiswa)	50
Gambar 4. 24 Perancangan Halaman Profil (Mahasiswa).....	51

Gambar 4. 25 Halaman Login	52
Gambar 4. 26 Halaman Dashboard (Admin).....	52
Gambar 4. 27 Halaman Kategori (Admin).....	53
Gambar 4. 28 Halaman New Kategori (Admin).....	54
Gambar 4. 29 Halaman Edit Kategori (Admin)	55
Gambar 4. 30 Halaman Data Mahasiswa (Admin).....	56
Gambar 4. 31 Halaman New Data Mahasiswa (Admin)	57
Gambar 4. 32 Halaman Edit Data Mahasiswa (Admin).....	57
Gambar 4. 33 Halaman Dataset Kategori (Admin)	58
Gambar 4. 34 Halaman Detail Dataset Kategori (Admin)	59
Gambar 4. 35 Halaman New Dataset (Admin).....	60
Gambar 4. 36 Halaman Detail Dataset (Admin)	60
Gambar 4. 37 Halaman Training Dataset (Admin)	61
Gambar 4. 38 Halaman Set Topik	62
Gambar 4. 39 Halaman Proposal Mahasiswa (Admin)	62
Gambar 4. 40 Halaman Detail Proposal (Admin)	63
Gambar 4. 41 Halaman New Proposal (Admin).....	64
Gambar 4. 42 Halaman Admin Account (Admin)	64
Gambar 4. 43 Halaman New Admin(Admin)	65
Gambar 4. 44 Halaman Edit Admin Account (Admin).....	65
Gambar 4. 45 Halaman Dashboard (Mahasiswa).....	66
Gambar 4. 46 Halaman New Proposal (Mahasiswa).....	67
Gambar 4. 47 Halaman Detail Proposal (Mahasiswa)	68
Gambar 4. 48 Halaman Profil (Mahasiswa)	69

Daftar Tabel

Tabel 2.1 Contoh Proses Case Folding pada Kalimat	7
Tabel 2.2 Contoh Proses Tokenizing pada Kalimat	8
Tabel 2.3 Contoh Proses Stopword Removal pada Kalimat	8
Tabel 2.4 Contoh Proses Stemming pada Kalimat	9
Tabel 4. 1 Tabel Judul Skripsi	21
Tabel 4. 2 Tabel Hasil Cleaning	22
Tabel 4. 3 Tabel Hasil Case Folding	23
Tabel 4. 4 Tabel Hasil Tokenizing	24
Tabel 4. 5 Tabel Hasil Stopword Removal.....	24
Tabel 4. 6 Tabel Hasil Stemming	25
Tabel 4. 7 Tabel TF	26
Tabel 4. 8 Tabel IDF.....	29
Tabel 4. 9 Tabel TF-IDF.....	32
Tabel 4. 10 Tabel Inisialisasi	35
Tabel 4. 11 Tbel Pembagian Topik	36
Tabel 4. 12 Tabel Probabilitas Topik	37

ABSTRAK

Dalam proses pengajuan judul proposal skripsi, banyak mahasiswa yang judulnya ditolak dikarenakan adanya kesamaan judul atau tema penelitian yang telah dilakukan sebelumnya. Proses ini dilakukan secara manual dimana judul skripsi direkapitulasi dengan menggunakan aplikasi Microsoft Excel, sehingga membuka peluang kekeliruan dalam pemeriksaan yang disebabkan oleh tim penyeleksi memeriksa judul secara manual baris per baris. Selanjutnya dirasa perlu melakukan penelitian untuk mengatasi masalah tersebut. Dengan harapan memudahkan pengelola program studi dalam menentukan judul skripsi yang potensial dan berkualitas pada mahasiswa. Selanjutnya mempercepat proses penentuan judul skripsi mahasiswa dan tentunya juga akan mempercepat proses penyelesaian studi mahasiswa strata 1. Maka dirancanglah sebuah sistem yang mampu merekomendasi kelayakan judul skripsi yang diajukan melalui teknologi text mining menggunakan algoritma LDA. Algoritma LDA mampu untuk mendeteksi topik yang ada pada suatu koleksi dokumen beserta besarnya kemunculan topik tersebut.

Kata Kunci: Skripsi, Judul, Text mining, LDA

BAB I

PENDAHULUAN

A. Latar Belakang

Skripsi merupakan karya tulis ilmiah yang disusun oleh mahasiswa pada tingkat Strata I yang dilaksanakan melalui field research ataupun library research. Penelitian ini dilakukan untuk memenuhi salah syarat untuk memperoleh Gelar Sarjana. Setiap mahasiswa yang telah memenuhi syarat, diwajibkan untuk mengajukan judul skripsi.

Setiap judul skripsi yang telah diajukan ke program studi, akan melalui proses seleksi yang ketat oleh tim penyeleksi. Banyak mahasiswa yang judulnya tertolak dikarenakan kesamaan judul atau tema penelitian yang telah dilakukan sebelumnya. Proses ini dilakukan secara manual dimana judul skripsi direkapitulasi dengan menggunakan aplikasi Microsoft Excel, sehingga membuka peluang kekeliruan dalam pemeriksaan yang disebabkan oleh tim penyeleksi memeriksa judul secara manual baris per baris. Saat ini di Program Studi Ilmu Komputer menerima pengajuan proposal judul skripsi mahasiswa lebih dari 500 judul per semester. Selain kesamaan judul dan tema, kontribusi penelitian terhadap ilmu pengetahuan khususnya Ilmu Komputer juga ikut menentukan diterima atau ditolaknya judul skripsi.

Text Mining merupakan suatu proses eksplorasi dan analisis dataset yang besar dalam bentuk text untuk mendapatkan suatu informasi yang bermanfaat untuk tujuan tertentu. Salah satu algoritma yang dapat digunakan dalam text mining adalah LDA (*Latent Dirichlet Allocation*). Algoritma LDA mampu untuk mendeteksi topik yang ada pada suatu koleksi dokumen beserta besarnya kemunculan topik tersebut pada koleksi dokumen maupun di dokumen tertentu.

Pada penelitian terdahulu yang dilakukan oleh Ayu Lestari pada tahun 2018 dengan judul “Implementasi Data Mining Untuk Seleksi Judul Skripsi Mahasiswa Menggunakan *Metode Association Rule* (Studi Kasus: Fakultas Ilmu Komputer Universitas Pasir Pengaraian)” hanya memberikan output judul diterima atau ditolak, sedangkan dalam penelitian ini output yang dihasilkan tidak hanya diterima atau ditolaknya judul skripsi tersebut, namun akan memberikan rekomendasi topik yang potensial untuk diterima.

Berdasarkan uraian diatas, maka Program Studi Ilmu Komputer dirasa perlu untuk melakukan penelitian terkait hal tersebut. Dengan demikian memudahkan pengelola program studi dalam menentukan judul skripsi yang potensial dan berkualitas. Selanjutnya mempercepat proses penentuan judul skripsi mahasiswa dan tentunya juga akan mempercepat proses penyelesaian studi strata I di program studi ilmu komputer UIN Sumatera Utara Medan dan meningkatkan jumlah mahasiswa yang lulus tepat waktu.

Berdasarkan permasalahan dan uraian penyelesaian masalah diatas, maka akan diajukan proposal penelitian dasar berbasis program studi melalui BOPTN Kemenag tahun 2022 dengan judul: “**Penerapan Text Mining Pada Sistem Penyeleksian Judul Skripsi Mahasiswa Menggunakan Algoritma LDA (*Latent Dirichlet Allocation*) di Program Studi Ilmu Komputer UIN Sumatera Utara Medan**”.

B. Permasalahan

Berdasarkan uraian pada latar belakang diatas, maka peneliti mengidentifikasi, membatasi, dan merumuskan permasalahan sebagai berikut:

1. Identifikasi Masalah

- a. Tingginya potensi *human error* dalam proses penyeleksian judul skripsi mahasiswa
- b. Lambatnya proses seleksi yang diakibatkan oleh cara pemeriksaan secara manual

2. Batasan Masalah

- a. Data yang digunakan berasal dari *database* judul proposal skripsi Program Studi Ilmu Komputer
- b. Bahasa pemrograman yang digunakan untuk *topic modeling* menggunakan Bahasa pemrograman *python*
- c. Algoritma yang digunakan dalam *topic modeling* adalah *Latent Dirichlet Allocation*

3. Rumusan Masalah

- a. Bagaimana menerapkan algoritma Latent Dirichlet Allocation dalam menyeleksi judul skripsi?
- b. Bagaimana menyelesaikan masalah lambatnya proses penentuan judul proposal skripsi mahasiswa?
- c. Bagaimana sistem dapat memberikan rekomendasi judul yang potensial, dari judul-judul dengan kemiripan topik yang tinggi?

C. Tujuan dan Manfaat

1. Merancang Sistem Penyeleksian Judul Skripsi Mahasiswai di Program Studi Ilmu Komputer UIN Sumatera Utara Medan dengan konsep Text Mining menggunakan algoritma *Latent Dirichlet Allocation*.
2. Menciptakan sistem yang mampu secara cepat menentukan judul skripsi yang diterima ataupun ditolak.
3. Memberikan rekomendasi judul yang potensial berdasarkan judul yang sudah pernah diajukan sebelumnya.

D. Signifikansi

1. Tersedianya sistem penyeleksian judul skripsi mahasiswa di Program Studi Ilmu Komputer UIN Sumatera Utara Medan.
2. Mempermudah pengelola Program Studi dalam menentukan judul skripsi mahasiswa.

3. Memberikan pelayanan terbaik dalam penyeleksian judul skripsi mahasiswa dalam waktu yang singkat.
4. Mempercepat proses penyelesaian skripsi mahasiswa, sehingga dapat meningkatkan jumlah lulusan yang tepat waktu.
5. *Outcome* berupa jurnal yang di publikasi pada jurnal terakreditasi dan HKI

E. Sistematika Penulisan

Adapun sistematika penulisan dalam laporan penelitian ini adalah sebagai berikut:

BAB I PENDAHULUAN

Bab ini terdiri dari 5 (lima) bagian yaitu Latar Belakang, Permasalahan (dibagi menjadi 3 (tiga) sub bagian yaitu identifikasi, batasan dan rumusan), Tujuan dan Manfaat, Signifikansi dan Sistematika Penulisan.

BAB II LANDASAN TEORI

Bab ini menyajikan kajian teori yang digunakan dalam pembahasan di penelitian ini, serta menyajikan Penelitian Terdahulu.

BAB III METODE

Bab ini menyajikan 4 komponen besar yang terdiri dari Jenis Penelitian, Pendekatan Penelitian, Teknik Penetapan Responden, dan Teknik Analisa Data.

BAB IV HASIL

Bab ini menyajikan Hasil Penelitian dan Diskusi Data atau Temuan Penelitian.

BAB V PENUTUP

Bab ini menyajikan Kesimpulan dan Saran.

BAB II

TEORI

A. Kajian Teori

Kajian-kajian teori yang digunakan dalam penelitian adalah sebagai berikut:

1. *Machine Learning*

Machine Learning pertama kali didefinisikan oleh Arthur Samuel (1959). Menurut Arthur Samuel, *machine learning* adalah salah satu bidang ilmu komputer yang memberikan kemampuan pembelajaran kepada computer untuk mengetahui sesuatu tanpa pemrogram yang jelas. Menurut (Mohri et al., 2014) *machine learning* juga dapat didefinisikan sebagai metode komputasi berdasarkan pengalaman untuk meningkatkan performa atau membuat prediksi yang akurat (Mohri et al., 2014). Definisi pengalaman di sini ialah informasi sebelumnya yang telah tersedia dan bisa dijadikan data pembelajar.

Dalam pembelajaran machine learning, terdapat beberapa konsep umpan balik untuk belajar. Seperti: *Supervised Learning*, *Unsupervised Learning*, *Semi-Supervised Learning* dan *Reinforcement Learning* (Russell & Norvig, 2021). *Supervised Learning* adalah pembelajaran dari pasangan *input-output* (contekan), contekan di sini yaitu masukkan data yang telah diberi label. Setelah itu membuat prediksi dari data yang telah diberi label. *Unsupervised Learning* adalah pembelajaran tanpa pasangan *input-output* atau kebalikan dari *supervised learning* yaitu masukkan data yang tidak diberi label. *Semi-Supervised Learning* adalah pembelajaran dari gabungan 2 (dua) konsep umpan balik yaitu *supervised learning* dan *unsupervised learning*. *Semi-Supervised Learning* biasanya karena banyak *noise* atau kurangnya data.

Reinforcement Learning adalah pembelajaran dari efek suatu tindakan, yang berupa hadiah (*reward*) dan hukuman (*punishment*) (Hakim, 2019).

2. Text Mining

Menurut Jurafsky dkk, *text mining* merupakan suatu kegiatan penambangan data dimana data yang digunakan berbentuk teks dan biasanya yang menjadi sumber dari data ini adalah dokumen, dengan tujuan untuk mencari kata yang dapat mewakili isi dokumen tersebut sehingga dapat dilakukannya analisis terkait dokumen tersebut. Pada *text mining* sendiri terdapat beberapa kegiatan penelitian antara lain ekstraksi dan penyimpanan teks, *preprocessing text*, dan analisis konten yang terdapat pada dokumen (Nurzahputra & Muslim, 2016).

Menurut Saraswati *text mining* adalah proses pengambilan informasi yang memiliki kualitas tinggi dari sebuah teks. Biasanya informasi yang diambil berpatokan pada kombinasi yang *relevansi*, kebaruan dan *interestingness* (Sunardi et al., 2018).

Text mining pada dasarnya memiliki cara kerja yang sama dengan *data mining* yang bertujuan untuk mengetahui informasi dari data yang besar (Rahim et al., 2018) tetapi ada hal yang membedakannya, jika pada *data mining* menggunakan data-data yang sudah terstruktur didalam *database* maka pada *text mining* data yang digunakan belum terstruktur sehingga membutuhkan klasifikasi, pada saat terjadinya klasifikasi maka data akan dirubah menjadi terstruktur sehingga dapat dipahami oleh sistem yang berjalan.

3. Text Pre-Processing

Text pre-processing merupakan metode yang dapat digunakan untuk menghasilkan data terstruktur dari data yang tidak terstruktur, biasanya data yang digunakan akan berubah menjadi dalam bentuk numerik dan data siap

diolah lebih lanjut (Ulfah Siregar et al., 2019). Untuk memperolehnya dilakukan dengan proses, yaitu:

a. *Cleaning*

Kalimat yang akan digunakan sebagai data dalam penelitian ini terlebih dahulu harus melewati proses *cleaning* untuk membersihkan setiap *noise* yang terdapat pada kalimat. Kata yang termasuk kedalam *noise* harus dihilangkan, seperti alamat email, *emoticon*, (:*, (:, :D), tanda baca seperti titik (.), koma (,), tanda tanya (?), dan tanda baca lainnya.

b. *Case Folding*

Setiap responden memiliki cara tersendiri dalam menuliskan tanggapan, ada yang menggunakan huruf besar ataupun kecil. Untuk membuat data yang digunakan menjadi terstruktur diperlukannya penyeragaman bentuk huruf menjadi huruf kecil (*lower case*), proses penyeragaman ini dinamakan *case folding* (Muljono et al., 2018). Yang dapat dirubah menjadi *case folding* hanya karakter berupa huruf “a” sampai “z” selain karakter huruf akan diabaikan dan dianggap menjadi delimiter. Contoh proses *case folding* dapat dilihat pada tabel dibawah ini.

Tabel 2.1 Contoh Proses Case Folding pada Kalimat

Kalimat Input	Kalimat Output
Saya sudah lama menunggu penerapan pembelajaran tatap muka karna lebih efisien dan mudah di pahami	saya sudah lama menunggu penerapan pembelajaran tatap muka karna lebih efisiean dan mudah dipahami

c. *Tokenizing*

Proses memisahkan kalimat *input* menjadi kata tunggal sesuai dengan kebutuhan yang disebut dengan *tokenizing*, tujuan dari proses ini untuk membentuk *token* (kata dasar) dari kalimat *input* (S & R, 2016). Untuk lebih

mudah dalam memahami proses *tokenizing* dapat melihat contoh pada tabel dibawah ini

Tabel 2.2 Contoh Proses Tokenizing pada Kalimat

Kalimat <i>Input</i>	Kalimat <i>Output</i>
Saya sudah lama menunggu penerapan pembelajaran tatap muka karna lebih efisien dan mudah dipahami	'saya', 'sudah', 'lama', 'menunggu', 'penerapan', 'pembelajaran', 'tatap', 'muka', 'karna', 'lebih', 'efisien', 'dan', 'mudah', 'dipahami'

d. *Stopword Removal*

Proses *stopword removal* bertujuan untuk menghapus token yang tidak memiliki arti dalam kalimat yang telah melalui proses *tokenizing* (Gilang Kencana & Sibaroni, n.d.) menggunakan fitur *bag-of-words* (Balya, 2019). Berikut contoh dari *stopword*, yaitu “dan”, “di”, “yang” dan lain-lain. Penerapan *stopword removal* dapat dilihat pada tabel dibawah ini.

Tabel 2.3 Contoh Proses *Stopword Removal* pada Kalimat

Kalimat <i>Input</i>	Kalimat <i>Output</i>
Saya sudah lama menunggu penerapan pembelajaran tatap muka karna lebih efisien dan mudah dipahami	'sudah', 'lama', 'menunggu', 'penerapan', 'pembelajaran', 'tatap', 'muka', 'lebih', 'efisien', 'mudah', 'paham'

e. *Stemming*

Proses *stemming* merupakan proses penghapusan imbuhan baik itu *affix*, *suffix* ataupun *prefix* guna memperoleh kata dasar (*root word*) dari setiap token yang telah melalui proses *stopword* (Putu et al., n.d.). Untuk mempermudah proses *stemming* pada bahasa Indonesia, dapat menggunakan

kamus dan *library* sastrawi yang didalamnya terdapat bentuk dasar dari kata-kata dalam bahasa Indonesia. Untuk lebih mudah dalam memahami proses *stemming* dapat melihat contoh pada tabel dibawah ini :

Tabel 2.4 Contoh Proses Stemming pada Kalimat

Kalimat <i>Input</i>	Kalimat <i>Output</i>
sudah lama menunggu penerapan pembelajaran tatap muka lebih efisien mudah pahami	‘sudah’, ‘lama’, ‘tunggu’, ‘terap’, ‘belajar’, ‘tatap’, ‘muka’, ‘lebih’, ‘efisien’, ‘mudah’, ‘paham’

4. Pembobotan TF-IDF

Pembobotan merupakan proses pemberian bobot pada setiap kata yang muncul pada satu dokumen (Gunawan et al., 2018). Proses pembobotan dilakukan untuk memperoleh nilai dari setiap kata yang berhasil di ekstrak (Buntoro, 2017). Salah satu metode yang terkenal untuk melakukan pembobotan kata adalah metode *Term Frequency – Invers Document Frequency* (TF – IDF).

Pada proses *Term Frequency* (TF) lebih fokus kepada kata yang sering terdapat pada data atau dokumen (Gunawan et al., 2018) setiap kata yang terdapat pada data akan diasumsikan memiliki kepentingan yang sesuai dengan jumlah kemunculan kata tersebut (Sevsa & Wahyudi, 2019). Sementara itu pada proses *Invers Document Frequency* (IDF) lebih berfokus pada kemunculan kata pada banyak dokumen yang berbeda, pada proses IDF semakin banyak kata yang muncul pada banyak dokumen akan diberikan bobot rendah (Gunawan et al., 2018). Untuk memudahkan dalam memperoleh nilai IDF (idf_t) dapat menggunakan persamaan berikut:

$$idf_t = \log \frac{N}{df_t} \dots \dots \dots (2.2)$$

Dimana :

- idf_t = nilai idf dari istilah t
- N = banyaknya dokumen yang tersedia
- D_{ft} = intensitas kemunculan kata pada dokumen

Adapun persamaan matematis yang dapat digunakan untuk menghitung nilai TF – IDF dapat dilihat pada persamaan berikut:

$$tf - idf_{t,d} = tf_{td} \times idf_t \dots\dots\dots(2.3)$$

Dimana :

- $idf_{t,d}$ = bobot TF – IDF dari kata t yang terdapat pada dokumen d
- tf_{td} = frekuensi kemunculan kata t pada dokumen d
- idf_t = nilai IDF (idf)

5. Topic Modeling

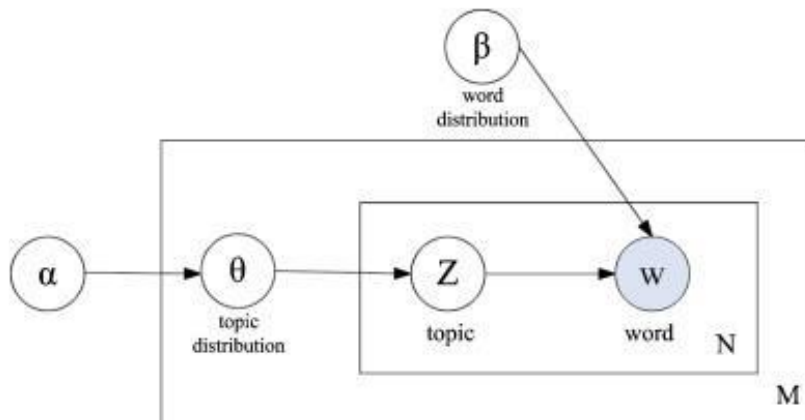
Konsep *topic modeling* menurut (Blei, 2012) terdiri dari entitas-entitas yaitu “kata”, “dokumen”, dan “*corpora*”. “Kata” dianggap sebagai unit dasar dari data diskrit dalam dokumen, didefinisikan sebagai item dari kosa kata yang diberi indeks untuk setiap kata unik pada dokumen. “Dokumen” adalah susunan N kata-kata. Sebuah *corpus* adalah kumpulan M dokumen dan *corpora* merupakan bentuk jamak dari *corpus*. Sementara “*topic*” adalah distribusi dari beberapa kosakata yang bersifat tetap. Secara sederhana, setiap dokumen dalam *corpus* mengandung proporsi tersendiri dari topik-topik yang dibahas sesuai kata-kata yang terkandung di dalamnya. *Topic modeling* merupakan sebuah topik yang terdiri dari kata-kata tertentu yang menyusun topik tersebut, dan dalam satu dokumen memiliki probabilitas masing-masing dari beberapa topik yang dihasilkan. Secara sederhana, *Topic Modeling* adalah algoritma yang memiliki tujuan untuk menemukan suatu topik yang tersembunyi dari rangkaian kata pada dokumen yang tidak terstruktur. Untuk menemukan topik yang berada antara teks tersebut dengan algoritma *Topic*

Modeling menganalisis dari teks asli, bagaimana topik-topik dapat saling terhubung satu dengan yang lain, bagaimana tema-tema bisa berubah dari waktu ke waktu, sehingga bisa dikembangkan untuk pencarian, atau dengan meringkas teks yang terdapat dalam dokumen (Putra & Kusumawardani, 2017).

Topic Modeling merupakan dokumen teks yang terdiri dari kata-kata, topik yang dapat dituliskan dalam banyak dokumen dapat dinyatakan dengan kombinasi kata-kata yang saling terkait dan teknik yang dapat digunakan untuk menyimpulkan suatu topik yang tersembunyi dalam sebuah dokumen teks. Karena *topic modeling* ini mewakili dari setiap dokumen sebagai kombinasi kompleks dari beberapa topik dan setiap topiknya sebagai kombinasi kompleks dari beberapa kata, kemudian juga digunakan sebagai alat *text mining* untuk mengklasifikasikan sebuah dokumen berdasarkan hasil kesimpulan topik (Nugroho & Alamsyah, 2018). Salah satu pemodelan topik yang paling populer yaitu *Latent Dirichlet Allocation* (LDA) (Putra & Kusumawardani, 2017).

6. Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) merupakan salah satu teknik yang terdapat dalam metode Pemodelan Topik. Kemampuan dasar dari LDA adalah dimana sebuah dokumen mengandung bermacam topik (Blei, 2012). LDA merupakan bentuk pemodelan statistik dari koleksi dokumen untuk menemukan intuisi ini. (Blei, 2012) memaknai topik sebagai distribusi pada kosakata yang tidak berubah. Berdasarkan probabilitasnya, LDA mewakilkan topik yang dikandung dalam suatu dokumen. Probabilitas topik tersebut mewakilkan arti dan pembahasan dari suatu dokumen. Pada gambar 1 Blei memaparkan bahwa suatu dokumen memiliki komponen penyusun.



Gambar 2.1 Gambar 1. Latent Dirichlet Allocation

Alpha (α) merupakan parameter untuk menghitung banyaknya topik yang muncul dalam suatu dokumen. Banyak dan sedikitnya topik yang terkandung dalam suatu dokumen diperoleh dari nilai alpha pada dokumen tersebut. Semakin besar nilai alpha-nya maka semakin banyak topik yang terkandung didalam dokumen tersebut, sebaliknya semakin kecil nilai alpha maka semakin sedikit pula topik yang terkandung pada dokumen tersebut. θ merepresentasikan distribusi topik dalam suatu dokumen. Z disimbolkan sebagai topik dari kata tertentu yang terkandung dalam suatu dokumen. w merepresentasikan setiap kata dalam suatu dokumen. Sedangkan Beta (β) merupakan parameter untuk menghitung kemunculan setiap kata dalam topik. Hampir sama seperti alpha nilai beta juga mempengaruhi sedikit dan banyaknya kata yang terdistribusi pada topik. Semakin besar nilai beta-nya maka semakin banyak kata yang terdistribusi di dalam topik, sebaliknya semakin kecil nilai beta-nya maka semakin spesifik pula topik yang dibahas karena mengandung sedikit kata di dalam topik.

7. *Topic Coherence*

Topic modeling membahas mengenai kumpulan dari sebuah kata-kata dari sebuah dokumen ataupun corpus. Berdasarkan dari kata-kata yang

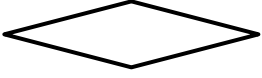

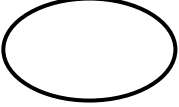



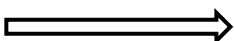
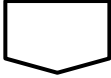

terdapat dalam dokumen yang digunakan, penggalian dari relasi topik dilakukan dengan asumsi bahwa pada satu dokumen meliputi suatu set kecil dari topik yang ringkas, dimana topik-topik ini perlu dikorelasikan dengan interpretasi manusia. Pada penelitian ini akan menggunakan validasi topik dengan menggunakan *coherence topic* (Putra & Kusumawardani, 2017).

Topic coherence yaitu dimana satu set dari kata-kata yang dihasilkan pada topik model dengan dinilai berdasarkan tingkat koherensi atau dalam diinterpretasi oleh manusia dengan tingkat kemudahannya. *Topic Coherence* mengukur nilai dari suatu topik dengan mengukur tingkat kesamaan semantik antara kata-kata yang ada dalam topik. Pengukuran ini dapat membantu dalam membedakan antara topik 22 yang dapat diinterpretasi secara semantik dengan topik yang memiliki keterkaitan secara statistik (Putra & Kusumawardani, 2017). *Topic Coherence* merupakan suatu ukuran yang akan digunakan untuk mengevaluasi *Topic Modeling*, dimana jika *coherence* skor topik yang tinggi maka model yang dihasilkan tersebut yang baik. *Topic Coherence* dapat dianggap memberikan kemampuan interpretasi lebih baik terhadap hasil dari *Topic Modeling* dibandingkan dengan Perplexity. Namun hasil dari matriks *perplexity* terkadang tidak memiliki korelasi yang baik pada interpretasi model oleh manusia (Listari, 2019).

8. *Flowchart Diagram*

Flowchart adalah representasi secara simbolik dari suatu algoritma atau prosedur untuk menyelesaikan suatu masalah, dengan menggunakan *flowchart* akan memudahkan pengguna melakukan pengecekan bagian-bagian yang terlupakan dalam analisis masalah, disamping itu *flowchart* juga berguna sebagai fasilitas untuk berkomunikasi antara pemrogram yang bekerja dalam tim suatu proyek (Pratama, 2019).

Tabel 2. 5 Simbol *Flowchart Diagram*

No	Simbol	Fungsi
1		Perbandingan, pernyataan, penyelesaian data yang memberikan pilihan untuk langkah selanjutnya
2		Permulaan sub program
3		Penghubung bagian-bagian <i>Flowchart</i> yang berada pada satu halaman.
4		Permulaan /akhir program
5		Proses penghitung/proses pengolahan data
6		Proses input/output data
7		Arah aliran program
8		Penghubung bagian-bagian <i>Flowchart</i> yang berada pada halaman berbeda
9		Proses inisialisasi/pemberian harga awal

9. Python

Sejak awal kemunculan nya hingga saat ini pengguna *python* terus bertambah banyak, hal ini dikarenakan *python* merupakan salah satu bahasa pemrograman yang serba guna, mulai dari pembuatan program sederhana hingga penggunaan pada *machine learning*. Hal ini yang menyebabkan *python*

dinobatkan sebagai salah satu bahasa yang populer dalam untuk digunakan pada *machine learning* dan *artificial intelegence* (kecerdasan buatan).

Popularitas *python* di *machine learning* didukung karna tersedianya *library* atau modul siap pakai seperti *Matpolib*, *Numpy*, *Pandas*, *NLTK*, *Scikit-learn* dan *library* lainnya yang dapat menunjang performa dari bahasa ini. *Python* merupakan salah satu bahasa yang dapat dieksekusi tanpa menggunakan *compiler*. Hal ini dikarenakan *python* termasuk kedalam salah satu bahasa *interpreted language* dimana *source code* yang telah dibuat dapat dijalankan hanya dengan menggunakan *interpreted* yang telah disediakan oleh bahasa tersebut tanpa perlu menggunakan *compiler* lagi, dimana *source code* akan dijalankan secara baris per baris dengan menggunakan *interpreted* yang telah di sediakan. Salah satu *interpreted* yang dimiliki *python* adalah *Ipython Project*, yang memiliki keunggulan lebih mudah digunakan dibandingkan *interpreted* lainnya. *Ipyhton* dapat diunggulkan dalam kecepatan pada saat proses *coding* dan *debugging* (Kurniawan, 2021).

Dibalik semua kelebihan yang dimilikinya, *python* juga masih memiliki keterbatasan. Dimana *pyhton* akan menggunakan banyak CPU (*central processor unit*) untuk melakukan operasi yang panjang, *python* juga kurang cocok untuk digunakan dalam pembuatan aplikasi dengan tampilan grafis yang kompleks.

B. Penelitian Terdahulu

1. Implementasi Data Mining Untuk Seleksi Judul Skripsi Mahasiswa Menggunakan Metode *Association Rule* (Studi Kasus: Fakultas Ilmu Komputer Universitas Pasir Pengaraian) yang dilakukan oleh Ayu Lestari pada Tahun 2018. Pada penelitian ini membahas pemanfaatan *data mining* dalam menentukan diterima atau ditolaknya judul skripsi yang diajukan tanpa memberikan rekomendasi judul yang potensial.

2. Implementasi Algoritma *Latent Dirichlet Allocation* pada data teks terjemah hadits bahasa Inggris yang dilakukan oleh Miftahul Huda pada Tahun 2020. Pada penelitian ini membahas pemanfaatan algoritma *Latent Dirichlet Allocation* pada teks terjemah hadits Bahasa Inggris.
3. Identifikasi Topik Artikel Berita Menggunakan *Topic Modeling* Dengan Metode *Latent Dirichlet Allocation* yang dilakukan oleh Kania Afrianti pada Tahun 2019. Pada penelitian ini membahas tentang identifikasi topik pada portal berita dan memberikan *output* berupa topik yang sedang menjadi trend pada portal berita.

BAB III

METODE

A. Jenis Penelitian

Jenis penelitian yang diadopsi dalam penelitian ini adalah penelitian R&D (*Research and Development*) dan menggunakan metode *waterfall* untuk pengembangan sistem. Pemilihan metode R&D berdasarkan pada tujuan penelitian untuk menghasilkan produk yang efisien. Metode penelitian *Research and Development*, memiliki beberapa tahapan sebagaimana dikemukakan oleh (Sugiyono, 2016) yaitu: “1. Potensi dan masalah, 2. Pengumpulan data, 3. Desain produk, 4. Validasi Desain, 5. Perbaikan Desain, 6. Uji Coba Produk, 7. Revisi Produk, 8. Uji Pelaksanaan Lapangan, 9. Penyempurnaan Produk, 10. Implementasi ”.

B. Pendekatan Penelitian

Penelitian ini menggunakan tipe penelitian eksperimental. Menurut (Sugiyono, 2016) penelitian eksperimental adalah penelitian eksperimen yaitu penelitian yang digunakan untuk mencari pengaruh perlakuan tertentu terhadap yang lain dalam kondisi yang terkendali. (Arikunto, 2017) juga berpendapat yang sama, mendefinisikan penelitian eksperimen merupakan penelitian yang dimaksudkan untuk mengetahui ada tidaknya akibat dari *treatment* pada subjek yang diselidiki. Cara untuk mengetahuinya yaitu membandingkan satu atau lebih kelompok eksperimen yang diberi *treatment* dengan satu kelompok pembanding yang tidak diberi *treatment*.

Menurut (Sugiyono, 2016) terdapat beberapa bentuk desain eksperimen yaitu:

1. *Pre-experimental design*
2. *True experimental design*

3. *Factorial design*

4. *Quasi experimental design.*

(Sugiyono, 2016) juga menyatakan bahwa ciri utama dari *quasi experimental design* adalah pengembangan dari *true experimental design*, yang mempunyai kelompok control namun tidak dapat berfungsi sepenuhnya untuk mengontrol variabel-variabel dari luar yang mempengaruhi eksperimen.

Berdasarkan uraian diatas dapat diambil kesimpulan bahwa *quasi experimental design* adalah jenis penelitian yang memiliki kelompok control dan kelompok eksperimen tidak dipilih secara acak. Pada penelitian ini peneliti menggunakan *quasi experimental design* karena dalam penelitian ini terdapat variabel-variabel dari luar yang tidak dapat dikontrol oleh peneliti.

C. Pengumpulan Data

Data yang digunakan dalam penelitian ini berupa dokumen proposal skripsi mahasiswa Program Studi Ilmu Komputer UIN Sumatera Utara Medan yang diperoleh langsung dari Program Studi Ilmu Komputer. Data yang digunakan adalah proposal skripsi Program Studi Ilmu Komputer dari tahun 2020 – 2022 dengan jumlah 395 *rows*. Data ini akan terus bertambah seiring berjalannya waktu.

Data yang diperoleh dari Program Studi Ilmu Komputer berupa *file spreadsheet* dalam format *gsheet*. *File* tersebut berisi *field-field* judul, dimana tiap *row* terdiri dari 3 (tiga) judul proposal. Ketiga judul proposal pada setiap *row* ini nantinya akan dipecah menjadi 3 (tiga) *row*. Dengan demikian data asli yang berjumlah 395 *row* nantinya akan pecah hingga menjadi 1185 *row*.

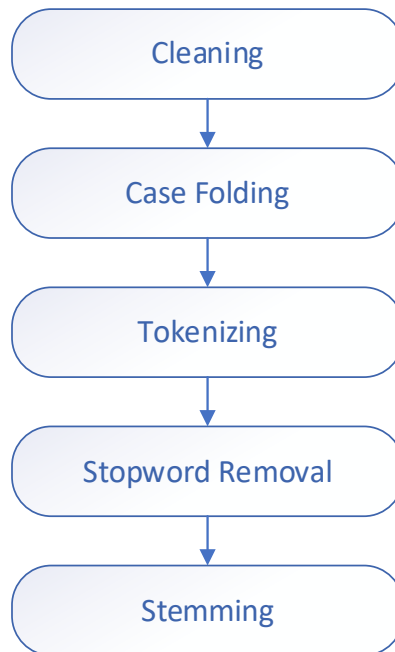
A. Analisa Data

Dalam melakukan analisa data yang akan melalui 3 (tiga) tahap pemrosesan, yaitu *preproccessing* data, pembobotan kata dengan TF-IDF, dan *topic modeling* dengan menggunakan *Latent Dirichlet Allocation*. Pada tahap

preprocessing data, dilakukan pembersihan data berupa tanda baca dan *stopword* yang sesuai dengan Bahasa Indonesia yang baku. Pada tahapan berikutnya yaitu pembobotan kata yang dilakukan dengan menggunakan TF-IDF, dimana setiap kata akan dijadikan data numerik dari dokumen yang akan diproses. Tahap akhir dari analisa data adalah *topic modelling* dimana algoritma *Latent Dirichlet Allocation* yang akan mengolah data hasil pembobotan menjadi hasil dari pemodelan topik.

1. Preprocessing Data

Langkah awal dalam penelitian ini sebelum melakukan tahap pengelompokan teks yaitu proses *preprocessing data*. Tahap *preprocessing data* ini dilakukan untuk membersihkan atau menghilangkan teks dalam dokumen yang tidak diperlukan. Tahapan *preprocessing* ini akan melalui 5 (lima) langkah proses sebelum dilakukan pembobotan pada tahap berikutnya. 5 (langkah) proses *preprocessing* dapat dilihat pada Gambar 3.1 dibawah ini.



Gambar 3. 1 Tahapan Preprocessing

2. Pembobotan TF-IDF

Proses tf-idf dilakukan untuk memperoleh nilai *numeric* dari setiap *teks* yang akan dikelompokkan. Tahap tf – idf dimulai dari menghitung nilai tf yang merupakan nilai *teks* berdasarkan seberapa sering *teks* tersebut muncul pada data, dilanjutkan dengan proses perhitungan idf yang berfokus pada seberapa sering *teks* muncul pada data yang berbeda dan diakhiri dengan proses perkalian antara nilai tf dan idf untuk memperoleh nilai *numeric* dari setiap teks, sehingga dapat dilakukan pengelompokkan dengan algoritma *Latent Dirichlet Allocation*.

3. Topic Modeling

Proses *topic modeling* menggunakan algoritma *Latent Dirichlet Allocation* dimulai dengan merubah kata yang terdapat pada *teks* judul skripsi kedalam bentuk *numeric* dan melakukan perhitungan frekuensi penyebaran kata menggunakan metode TF – IDF, dilanjutkan dengan menentukan jumlah *topic* yang digunakan, dalam penelitian ini menggunakan lebih dari satu *topic*. Untuk mengetahui *teks* judul skripsi yang digunakan masuk kedalam *topic 1*, *topic 2* atau *topic n* dapat dilakukan perhitungan probabilitas, proses penentuan *topic* dari *teks* berdasarkan nilai *probabilitas* terbesar dari beberapa *topic* yang telah ditentukan.

BAB IV

HASIL DAN PEMBAHASAN

A. Representasi Data

Data yang digunakan dalam penelitian ini merupakan judul skripsi yang diajukan mahasiswa kepada program studi Ilmu Komputer Universitas Islam Negeri Sumatera Utara dalam priode pengajuan tahun 2019 – 2022 dengan jumlah 395 *row*.

Pada masing-masing *row* memiliki tiga kolom judul, sehingga total keseluruhan dari judul skripsi yang digunakan pada penelitian inii sebanyak 1185 *row*.

B. Hasil Analisis Data

Hasil analisis data digunakan untuk mengetahui apakah data yang digunakan sesuai dengan penelitian. Tahap awal adalah *preprocessing teks* yang bertujuan untuk merubah *teks* menjadi data terstruktur. Berikut ini merupakan *sample teks* berupa judul yang digunakan.

Tabel 4. 1 Tabel Judul Skripsi

Judul Skripsi			
IMPLEMENTASI	METODE	PREFERENCE	RANKING
ORGANIZATION	METHOD	FOR ENRICHMENT	EVALUATION
(PROMETHEE II)	UNTUK PENENTUAN	KENAIKAN	JABATAN
KARYAWAN	PT AGUNG	SARANA	UTAMA
Implementasi	Algoritma	Principal Component	Analysis (PCA) Untuk
Aplikasi	Absensi	Pengenalan	Wajah

Judul Skripsi
Implementasi Data Mining Menggunakan Algoritma C4.5 untuk Penentuan Penerima Bantuan Siswa Miskin Pada Masa Pandemi Covid-19 di SMKN 1 Lubuk Pakam
Implementasi Pengamanan File Citra Bitmap dengan Algoritma Chacha20
analisis perbandingan penentuan jalur terpendek dengan menggunakan algoritma general and test dengan best first search

1. Cleaning

Cleaning dilakukan untuk membersihkan data (*teks*) dari *noise* berupa tanda baca. Berikut ini *teks* hasil *cleaning*.

Tabel 4. 2 Tabel Hasil Cleaning

Judul Skripsi
IMPLEMENTASI METODE PREFERENCE RANKING ORGANIZATION METHOD FOR ENRICHMENT EVALUATION PROMETHEE II UNTUK PENENTUAN KENAIKAN JABATAN KARYAWAN PT AGUNG SARANA UTAMA
Implementasi Algoritma Principal Component Analysis PCA Untuk Aplikasi Absensi Pengenalan Wajah
Implementasi Data Mining Menggunakan Algoritma C4 5 untuk Penentuan Penerima Bantuan Siswa Miskin Pada Masa Pandemi Covid 19 di SMKN 1 Lubuk Pakam
Implementasi Pengamanan File Citra Bitmap dengan Algoritma Chacha20
analisis perbandingan penentuan jalur terpendek dengan menggunakan algoritma general and test dengan best first search

2. Case Folding

Tahapan ini bertujuan untuk melakukan penyeragaman *case* menjadi huruf kecil pada penulisan judul skripsi, dikarenakan bahasa yang digunakan dalam penerapan *topic modeling* merupakan *case sensitive* yang menganggap judul skripsi yang ditulis dengan huruf kapital dan huruf kecil berbeda walaupun sebenarnya kedua judul tersebut sama dan hal ini dapat menyebabkan terjadinya data kembar.

Tabel 4. 3 Tabel Hasil Case Folding

Judul Skripsi
implementasi metode preference ranking organization method for enrichment evaluation promethee ii untuk penentuan kenaikan jabatan karyawan pt agung sarana utama
implementasi algoritma principal component analysis pca untuk aplikasi absensi pengenalan wajah
implementasi data mining menggunakan algoritma c4 5 untuk penentuan penerima bantuan siswa miskin pada masa pandemi covid 19 di smkn 1 lubuk pakam
implementasi pengamanan file citra bitmap dengan algoritma chacha20
analisis perbandingan penentuan jalur terpendek dengan menggunakan algoritma general and test dengan best first search

3. Tokenizing

Tahapan *tokenizing* untuk memecah *teks* judul skripsi menjadi elemen kecil berupa kata tunggal yang dipisahkan dengan tanda petik (‘ ’) sehingga dapat mempermudah tahapan selanjutnya. Berikut ini *teks* hasil *tokenizing*

Tabel 4. 4 Tabel Hasil Tokenizing

Judul Skripsi
'implementasi', 'metode', 'preference', 'ranking', 'organization', 'method', 'for', 'enrichment', 'evaluation', 'promethee', 'ii', 'untuk', 'penentuan', 'kenaikan', 'jabatan', 'karyawan', 'pt', 'agung', 'sarana', 'utama'
'implementasi', 'algoritma', 'principal', 'component', 'analysis', 'pca', 'untuk', 'aplikasi', 'absensi', 'pengenalan', 'wajah'
'implementasi', 'data', 'mining', 'menggunakan', 'algoritma', 'c4', '5', 'untuk', 'penentuan', 'penerima', 'bantuan', 'siswa', 'miskin', 'pada', 'masa', 'pandemi', 'covid', '19', 'di', 'smkn', '1', 'lubuk', 'pakam'
'implementasi', 'pengamanan', 'file', 'citra', 'bitmap', 'dengan', 'algoritma', 'chacha20'
'analisis', 'perbandingan', 'penentuan', 'jalur', 'terpendek', 'dengan', 'menggunakan', 'algoritma', 'general', 'and', 'test', 'dengan', 'best', 'first', 'search'

4. Stopword Removal

Stopword removal digunakan untuk menghapus kata – kata yang tidak memiliki makna penting dalam *teks* judul skripsi. Berikut ini hasil *stopword removal*

Tabel 4. 5 Tabel Hasil Stopword Removal

Judul Skripsi
'implementasi', 'metode', 'preference', 'ranking', 'organization', 'method', 'for', 'enrichment', 'evaluation', 'promethee', 'ii', 'penentuan', 'kenaikan', 'jabatan', 'karyawan', 'pt', 'agung', 'sarana', 'utama'

Judul Skripsi
'implementasi', 'algoritma', 'principal', 'component', 'analysis', 'pca', 'aplikasi', 'absensi', 'pengenalan', 'wajah'
'implementasi', 'data', 'mining', 'menggunakan', 'algoritma', 'c4', '5', , 'penentuan', 'penerima', 'bantuan', 'siswa', 'miskin', 'pandemi', 'covid', '19', 'smkn', '1', 'lubuk', 'pakam'
'implementasi', 'pengamanan', 'file', 'citra', 'bitmap', 'algoritma', 'chacha20'
'analisis', 'perbandingan', 'penentuan', 'jalur', 'terpendek', 'menggunakan', 'algoritma', 'general', 'and', 'test', 'best', 'first', 'search'

5. Stemming

Tahapan untuk menghapus imbuhan yang terdapat pada kata baik itu berupa *suffix*, *prefix* maupun *affix*. Sehingga setiap kata akan kembali kebentuk kata dasar. Berikut ini hasil tahapan *stemming*

Tabel 4. 6 Tabel Hasil Stemming

Judul Skripsi
'implementasi', 'metode', 'preference', 'ranking', 'organization', 'method', 'for', 'enrichment', 'evaluation', 'promethee', 'ii', 'tentu', 'naik', 'jabat', 'karyawan', 'pt', 'agung', 'sarana', 'utama'
'implementasi', 'algoritma', 'principal', 'component', 'analysis', 'pca', 'aplikasi', 'absensi', 'kenal', 'wajah'
'implementasi', 'data', 'mining', 'guna', 'algoritma', 'c4', '5', , 'tentu', 'terima', 'bantu', 'siswa', 'miskin', 'pandemi', 'covid', '19', 'smkn', '1', 'lubuk', 'pakam'
'implementasi', 'aman', 'file', 'citra', 'bitmap', 'algoritma', 'chacha20'

‘analisis’, ‘banding’, ‘tentu’, ‘jalur’, ‘pendek’, ‘guna’, ‘algoritma’,
‘general’, ‘and’, ‘test’, ‘best’, ‘first’, ‘search’

6. Pembobotan TF – IDF

TF – IDF dilakukan untuk mencari nilai *numeric* dari setiap kata yang terdapat pada *teks* hasil *preprocessing* berdasarkan intensitas kemunculan kata tersebut.

Tahap awal yang di lakukan adalah menghitung nilai TF dari semua kata yang terdapat dalam *teks*, berikut ini hasil perhitungan nilai TF :

Tabel 4. 7 Tabel TF

Kata	TF					DF
	T1	T2	T3	T4	T5	
implementasi	1	1	1	1	0	4
metode	1	0	0	0	0	1
preference	1	0	0	0	0	1
ranking	1	0	0	0	0	1
organization	1	0	0	0	0	1
method	1	0	0	0	0	1
for	1	0	0	0	0	1
enrichment	1	0	0	0	0	1
evaluation	1	0	0	0	0	1
promethee	1	0	0	0	0	1
ii	1	0	0	0	0	1
tentu	1	0	1	0	1	3
naik	1	0	0	0	0	1

Kata	TF					DF
	T1	T2	T3	T4	T5	
jabat	1	0	0	0	0	1
karyawan	1	0	0	0	0	1
pt	1	0	0	0	0	1
agung	1	0	0	0	0	1
sarana	1	0	0	0	0	1
utama	1	0	0	0	0	1
algoritma	0	1	1	1	1	4
principal	0	1	0	0	0	1
component	0	1	0	0	0	1
analysis	0	1	0	0	0	1
pca	0	1	0	0	0	1
aplikasi	0	1	0	0	0	1
absen	0	1	0	0	0	1
kenal	0	1	0	0	0	1
wajah	0	1	0	0	0	1
data	0	0	1	0	0	1
mining	0	0	1	0	0	1
c4	0	0	1	0	0	1
terima	0	0	1	0	0	1
bantu	0	0	1	0	0	1
siswa	0	0	1	0	0	1
miskin	0	0	1	0	0	1
pandemi	0	0	1	0	0	1
covid	0	0	1	0	0	1
smkn	0	0	1	0	0	1

Kata	TF					DF
	T1	T2	T3	T4	T5	
lubuk	0	0	1	0	0	1
pakam	0	0	1	0	0	1
aman	0	0	0	1	0	1
file	0	0	0	1	0	1
citra	0	0	0	1	0	1
bitmap	0	0	0	1	0	1
analisis	0	0	0	0	1	1
banding	0	0	0	0	1	1
jalur	0	0	0	0	1	1
pendek	0	0	0	0	1	1
general	0	0	0	0	1	1
and	0	0	0	0	1	1
test	0	0	0	0	1	1
best	0	0	0	0	1	1
first	0	0	0	0	1	1
search	0	0	0	0	1	1

Tahap selanjutnya menghitung nilai IDF menggunakan persamaan

$idf_t = \log \frac{N}{df_t}$, berikut ini adalah proses perhitungannya,

a. $idf_{implementasi} = \log \frac{4}{5} = -0,97$

b. $idf_{metode} = \log \frac{1}{5} = -0,699$

c. $idf_{preference} = \log \frac{1}{5} = -0,699$

d. $idf_{ranking} = \log \frac{1}{5} = -0,699$

e. $idf_{organization} = \log \frac{1}{5} = -0,699$

Untuk proses perhitungan nilai IDF tidak dijabarkan keseluruhan proses perhitungan yang terdapat pada *sample teks* yang digunakan. Untuk hasil nilai IDF keseluruhan kata telah disajikan dalam bentuk tabel dibawah ini

Tabel 4. 8 Tabel IDF

Kata	DF	IDF
implementasi	4	-0,097
metode	1	-0,699
preference	1	-0,699
ranking	1	-0,699
organization	1	-0,699
method	1	-0,699
for	1	-0,699
enrichment	1	-0,699
evaluation	1	-0,699
promethee	1	-0,699
ii	1	-0,699
tentu	3	-0,125
naik	1	-0,699
jabat	1	-0,699
karyawan	1	-0,699
pt	1	-0,699
agung	1	-0,699
sarana	1	-0,699
utama	1	-0,699
algoritma	4	-0,097

Kata	DF	IDF
principal	1	-0,699
component	1	-0,699
analysis	1	-0,699
pca	1	-0,699
aplikasi	1	-0,699
absen	1	-0,699
kenal	1	-0,699
wajah	1	-0,699
data	1	-0,699
mining	1	-0,699
c4	1	-0,699
terima	1	-0,699
bantu	1	-0,699
siswa	1	-0,699
miskin	1	-0,699
pandemi	1	-0,699
covid	1	-0,699
smkn	1	-0,699
lubuk	1	-0,699
pakam	1	-0,699
aman	1	-0,699
file	1	-0,699
citra	1	-0,699
bitmap	1	-0,699
analisis	1	-0,699

Kata	DF	IDF
banding	1	-0,699
jalur	1	-0,699
pendek	1	-0,699
general	1	-0,699
and	1	-0,699
test	1	-0,699
best	1	-0,699
first	1	-0,699
search	1	-0,699

Setelah diperoleh nilai TF dari semua kata yang terdapat dalam *teks* judul skripsi, dilanjutkan dengan menghitung nilai TF – IDF, dengan cara melakukan perkalian antara nilai TF dan IDF dari masing – masing kata yang telah diperoleh. Berikut ini cara perhitungan TF – IDF untuk masing – masing kata.

$$tf - idf_{t,d} = tf_{td} \times idf_t$$

$$\begin{aligned} \text{a. } tf - idf (\text{implementasi, T1}) &= tf_{\text{implementasi.T1}} \times idf_{\text{implementasi}} \\ &= 1 \times -0,097 = -0,097 \end{aligned}$$

$$\begin{aligned} \text{b. } tf - idf (\text{implementasi, T2}) &= tf_{\text{implementasi.T2}} \times idf_{\text{implementasi}} \\ &= 1 \times -0,097 = -0,097 \end{aligned}$$

$$\begin{aligned} \text{c. } tf - idf (\text{implementasi, T3}) &= tf_{\text{implementasi.T3}} \times idf_{\text{implementasi}} \\ &= 1 \times -0,097 = -0,097 \end{aligned}$$

$$\begin{aligned} \text{d. } tf - idf (\text{implementasi, T4}) &= tf_{\text{implementasi.T4}} \times idf_{\text{implementasi}} \\ &= 1 \times -0,097 = -0,097 \end{aligned}$$

$$\text{e. } tf - idf (\text{implementasi, T5}) = tf_{\text{implementasi.T5}} \times idf_{\text{implementasi}}$$

$$= 0 \times -0,097 = 0$$

Untuk proses perhitungan nilai TF- IDF tidak dijabarkan keseluruhan proses perhitungan yang terdapat pada *sample teks* yang digunakan. Untuk hasil nilai IDF keseluruhan kata telah disajikan dalam bentuk tabel dibawah ini

Tabel 4. 9 Tabel TF-IDF

Kata	TF					DF	IDF	TF-IDF				
	T1	T2	T3	T4	T5			T1	T2	T3	T4	T5
implementasi	1	1	1	1	0	4	-0,097	-0,097	-0,097	-0,097	-0,097	0
metode	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0
preference	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0
ranking	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0
organisasi	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0
method	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0
for	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0
enrichment	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0
evaluation	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0
promethee	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0
ii	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0
tentu	1	0	1	0	1	3	-0,125	-0,125	0	-0,125	0	-0,125
naik	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0
jabat	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0
karyawan	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0

Kata	TF					DF	IDF	TF-IDF				
	T1	T2	T3	T4	T5			T1	T2	T3	T4	T5
pt	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0
agung	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0
sarana	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0
utama	1	0	0	0	0	1	-0,699	-0,699	0	0	0	0
algoritma	0	1	1	1	1	4	-0,097	0	-0,097	-0,097	-0,097	-0,097
principal	0	1	0	0	0	1	-0,699	0	-0,699	0	0	0
component	0	1	0	0	0	1	-0,699	0	-0,699	0	0	0
analysis	0	1	0	0	0	1	-0,699	0	-0,699	0	0	0
pca	0	1	0	0	0	1	-0,699	0	-0,699	0	0	0
aplikasi	0	1	0	0	0	1	-0,699	0	-0,699	0	0	0
absen	0	1	0	0	0	1	-0,699	0	-0,699	0	0	0
kenal	0	1	0	0	0	1	-0,699	0	-0,699	0	0	0
wajah	0	1	0	0	0	1	-0,699	0	-0,699	0	0	0
data	0	0	1	0	0	1	-0,699	0	0	-0,699	0	0
mining	0	0	1	0	0	1	-0,699	0	0	-0,699	0	0
c4	0	0	1	0	0	1	-0,699	0	0	-0,699	0	0
terima	0	0	1	0	0	1	-0,699	0	0	-0,699	0	0
bantu	0	0	1	0	0	1	-0,699	0	0	-0,699	0	0
siswa	0	0	1	0	0	1	-0,699	0	0	-0,699	0	0
miskin	0	0	1	0	0	1	-0,699	0	0	-0,699	0	0
pandemi	0	0	1	0	0	1	-0,699	0	0	-0,699	0	0
covid	0	0	1	0	0	1	-0,699	0	0	-0,699	0	0
smkn	0	0	1	0	0	1	-0,699	0	0	-0,699	0	0

Kata	TF					DF	IDF	TF-IDF				
	T1	T2	T3	T4	T5			T1	T2	T3	T4	T5
lubuk	0	0	1	0	0	1	-0,699	0	0	-0,699	0	0
pakam	0	0	1	0	0	1	-0,699	0	0	-0,699	0	0
aman	0	0	0	1	0	1	-0,699	0	0	0	-0,699	0
file	0	0	0	1	0	1	-0,699	0	0	0	-0,699	0
citra	0	0	0	1	0	1	-0,699	0	0	0	-0,699	0
bitmap	0	0	0	1	0	1	-0,699	0	0	0	-0,699	0
analisis	0	0	0	0	1	1	-0,699	0	0	0	0	-0,699
banding	0	0	0	0	1	1	-0,699	0	0	0	0	-0,699
jalur	0	0	0	0	1	1	-0,699	0	0	0	0	-0,699
pendek	0	0	0	0	1	1	-0,699	0	0	0	0	-0,699
general	0	0	0	0	1	1	-0,699	0	0	0	0	-0,699
and	0	0	0	0	1	1	-0,699	0	0	0	0	-0,699
test	0	0	0	0	1	1	-0,699	0	0	0	0	-0,699

Kata	TF					DF	IDF	TF-IDF				
	T1	T2	T3	T4	T5			T1	T2	T3	T4	T5
best	0	0	0	0	1	1	-0,699	0	0	0	0	-0,699
first	0	0	0	0	1	1	-0,699	0	0	0	0	-0,699
search	0	0	0	0	1	1	-0,699	0	0	0	0	-0,699

7. Topic Modeling Menggunakan Latent Dirichlet Allocation

Pembuatan *topic modeling* menggunakan *Latent Dirichlet Allocation*, menggunakan nilai TF-IDF yang telah diperoleh. *Teks* yang diprediksi topiknya dibatasi hanya menggunakan satu *teks* judul skripsi yang berasal dari T1. Berikut ini adalah nilai TF-IDF dari T1.

Tabel 4. 10 Tabel Inisialisasi

Kata	Inisialisasi Kata	Penyebaran Kata (DF)
implementasi	1	4
metode	2	1
preference	3	1
ranking	4	1
organization	5	1

Kata	Inisialisasi Kata	Penyebaran Kata (DF)
method	6	1
for	7	1
enrichment	8	1
evaluation	9	1
promethee	10	1
ii	11	1
tentu	12	3
naik	13	1
jabat	14	1
karyawan	15	1
pt	16	1
agung	17	1
sarana	18	1

Kemudian tentukan jumlah topik yang akan digunakan, dalam proses ini peneliti menggunakan 2 topik, yang di inisialisasi kan sebagai TA dan TB. Dilanjutkan dengan menentukan topik dari setiap kata yang terdapat pada *teks* judul skripsi secara acak. Berikut ini adalah pembagain kata berdasarkan topik nya.

Tabel 4. 11 Tbel Pembagian Topik

TA		TB	
Kata	DF	Kata	DF
1	4	3	1
2	1	5	1
4	1	6	1

TA		TB	
Kata	DF	Kata	DF
7	1	8	1
9	1	10	1
11	1	12	3
13	1	14	1
16	1	15	1
18	1	17	1

Berdasarkan penyebaran kata pada dua topik, maka dapat diketahui nilai probabilitas dari masing – masing topik, yaitu

Tabel 4. 12 Tabel Probabilitas Topik

Teks	Topik			
	TA		TB	
	Frekuensi	Probabilitas	Frekuensi	Probabilitas
T1	12	53%	11	47%

Berdasarkan nilai probabilitas yang telah diperoleh dapat disimpulkan bahwa T1 memiliki topik TA dengan nilai probabilitas sebesar 53%.

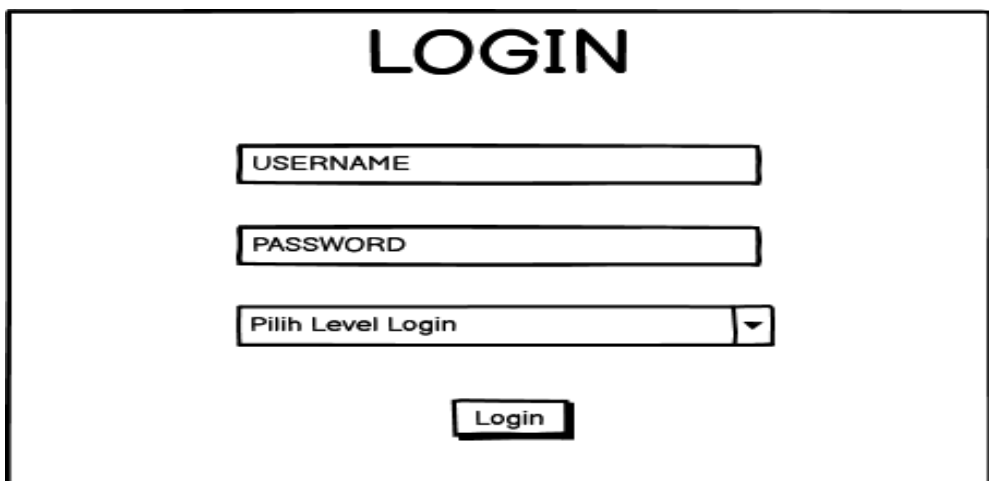
Berdasarkan proses analisis data serta representasi data, maka *topic modeling* menggunakan algoritma *Latent Dirichlet Allocation*, dapat membantu dalam menentukan topik terkait judul skripsi yang diajukan mahasiswa sehingga membantu proses penyeleksian judul skripsi pada program studi ilmu komputer.

C. Perancangan User Interface

Perancangan *user interface* sangat membantu proses pembuatan aplikasi dari penelitian ini, berikut ini adalah rancangan *user interface* pada aplikasi yang digunakan

1. Perancangan Halaman Login

Halaman ini berisi *field* untuk pengisian *username* dan *password* serta pilihan level *login* sebagai administrator atau sebagai mahasiswa. Berikut ini adalah desain tampilannya :



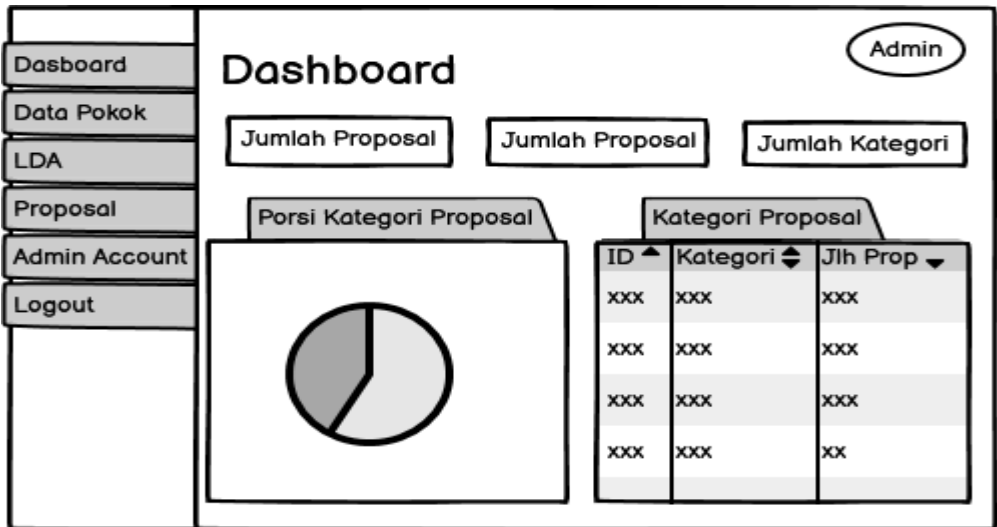
The image shows a login form with the following elements:

- A title "LOGIN" centered at the top.
- A text input field labeled "USERNAME".
- A text input field labeled "PASSWORD".
- A dropdown menu labeled "Pilih Level Login" with a downward arrow.
- A button labeled "Login" centered below the input fields.

Gambar 4. 1 Perancangan Halaman Login

2. Perancangan Halaman Dashboard (Admin)

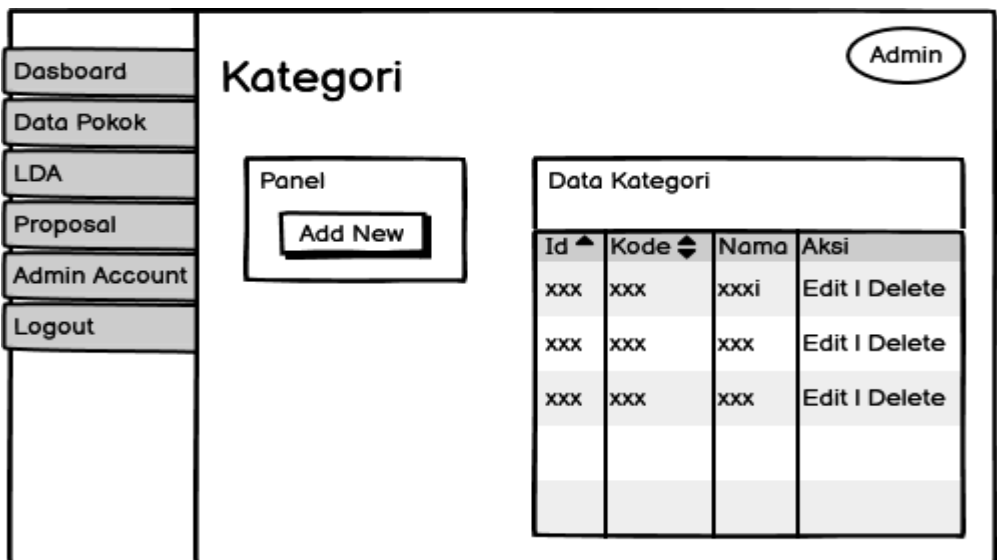
Halaman ini merupakan halaman beranda ketika admin berhasil melewati proses *login*. Halaman ini berisi informasi terkait jumlah mahasiswa, jumlah proposal, serta jumlah kategori yang telah masuk ke dalam *database* sistem. Pada halaman ini interaksi *admin* hanya sebatas melihat informasi ringkas saja.



Gambar 4. 2 Perancangan Halaman Dashboard

3. Perancangan Halaman Kategori (Admin)

Halaman ini digunakan *admin* untuk melihat daftar kategori topik pada sistem. Pada halaman ini juga terdapat fitur untuk menambahkan kategori untuk judul skripsi serta dapat melihat dan memberikan aksi kepada judul skripsi yang telah masuk ke dalam *database*.



Gambar 4. 3 Perancangan Halaman Kategori (Admin)

4. Halaman New Kategori (Admin)

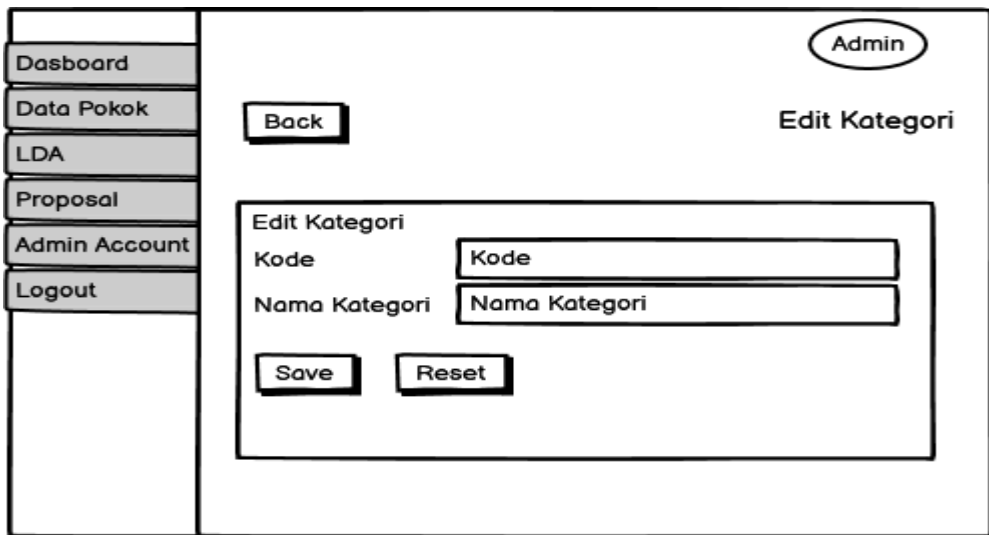
Halaman ini berfungsi untuk menambahkan kategori terkait topik dari judul skripsi. Interaksi yang dapat dilakukan *admin* adalah dengan mengisi *field* Kode Kategori dan Nama Kategori. Tombol *Save* digunakan untuk menyimpan Kode dan Nama Kategori yang telah diisi pada masing-masing *field*, sedangkan tombol *Reset* digunakan untuk memabatalkan.

The image shows a web interface for an administrator to create a new category. On the left is a vertical sidebar with buttons for 'Dashboard', 'Data Pokok', 'LDA', 'Proposal', 'Admin Account', and 'Logout'. The main area is titled 'Kategori' and contains a 'Back' button. Below this is a form titled 'New Kategori' with two input fields: 'Kode : Kode Kategori' and 'Nama Kategori : Nama Kategori'. At the bottom of the form are 'Save' and 'Reset' buttons. In the top right corner of the main area, there is a circular button labeled 'Admin'.

Gambar 4. 4 Perancangan Halaman New Kategori (Admin)

5. Halaman Edit Kategori (Admin)

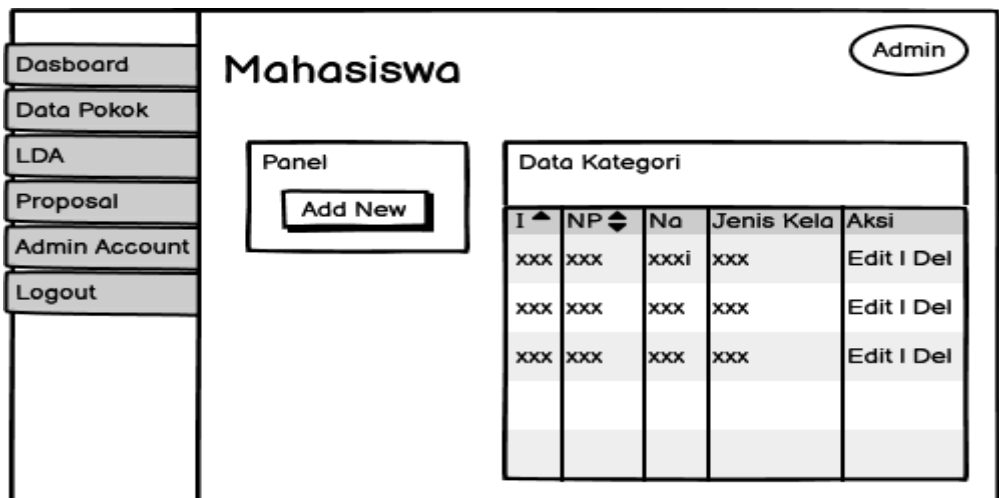
Halaman ini berfungsi untuk melakukan perubahan pada kategori yang telah dimiliki sebelumnya. Perubahan data pada Kategori dapat dilakukan dengan merubah isi dari *field* Kode dan *Field* Nama Kategori. Tombol *Save* digunakan untuk menyimpan perubahan, sedangkan tombol *Reset* digunakan untuk membatalkan.



Gambar 4. 5 Perancangan Halaman Edit Kategori (Admin)

6. Halaman Data Mahasiswa (Admin)

Pada halaman ini, admin dapat melihat data mahasiswa aktif yang memiliki akun dan dapat melakukan *submit* proposal skripsi. Aksi yang disediakan pada halaman ini adalah menambah, mengubah, dan menghapus data mahasiswa. Dimana seluruh aksi yang tersedia ini akan diarahkan ke halaman aksi masing-masing.



Gambar 4. 6 Perancangan Halaman Mahasiswa (Admin)

7. Halaman *Add New Data Mahasiswa* (Admin)

Halaman *add new* dapat membantu admin untuk menambahkan data mahasiswa.

The screenshot shows a web interface for adding a new student. On the left is a vertical sidebar menu with the following items: Dashboard, Data Pokok, LDA, Proposal, Admin Account, and Logout. The main content area is titled "Mahasiswa" and "Add New Mahasiswa". In the top right corner of the main area, there is a circled "Admin" label. The form for "New Mahasiswa" includes four input fields: "NPM" (with placeholder "NPM Mahasiswa"), "Nama Mahasiswa" (with placeholder "Nama Mahasiswa"), "Jenis Kelamin" (with placeholder "Jenis Kelamin"), and "Password Login" (with placeholder "*****"). Below the form are two buttons: "Save" and "Reset".

Gambar 4. 7 Perancangan Halaman Add New Mahasiswa (Admin)

8. Halaman Edit Data Mahasiswa (Admin)

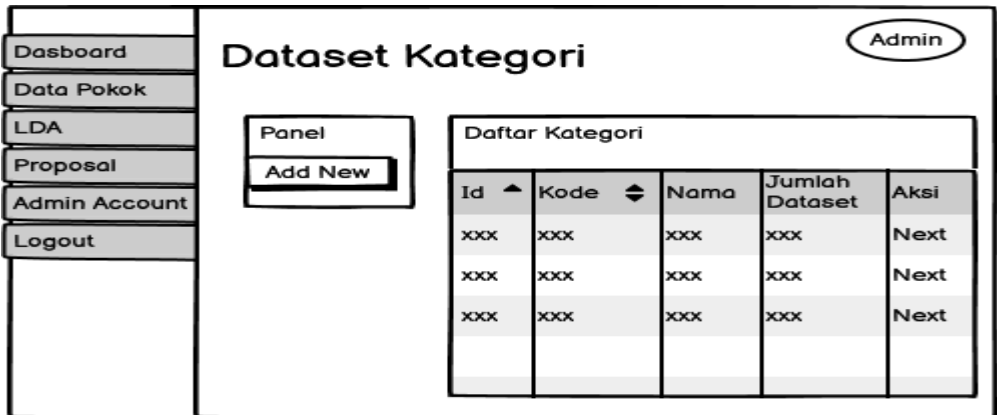
Halaman ini dapat membantu admin untuk merubah data mahasiswa yang telah mengajukan judul, apabila terdapat kesalahan.

The screenshot shows a web interface for editing a student's data. On the left is a vertical sidebar menu with the following items: Dashboard, Data Pokok, LDA, Proposal, Admin Account, and Logout. The main content area is titled "Edit Mahasiswa". In the top right corner of the main area, there is a circled "Admin" label. At the top left of the main area is a "Back" button. The form for "Edit Mahasiswa" includes four input fields: "NPM" (with placeholder "NPM"), "Nama" (with placeholder "Nama"), "Jenis Kelamin" (with placeholder "Jenis Kelamin"), and "Password Login" (with placeholder "*****"). Below the form are two buttons: "Save" and "Reset".

Gambar 4. 8 Perancangan Halaman Edit Mahasiswa (Admin)

9. Halaman Dataset Kategori (Admin)

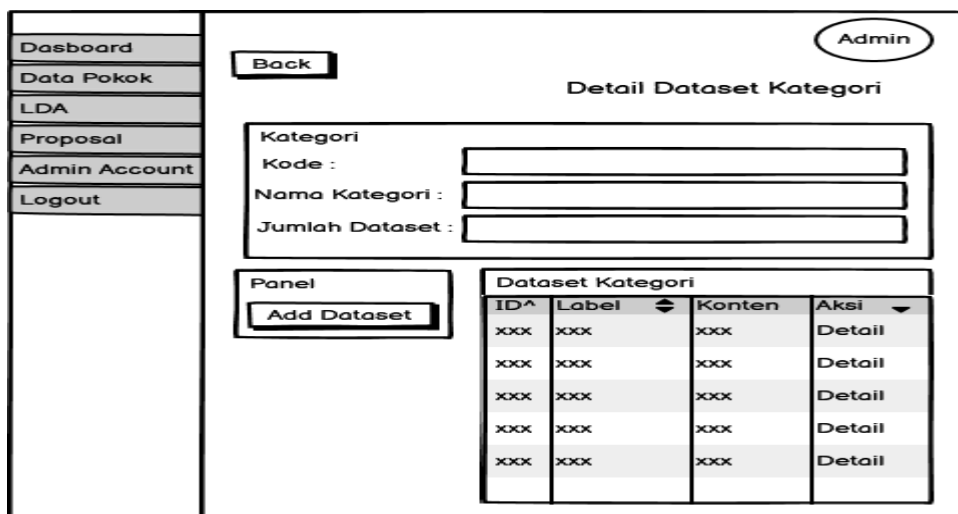
Pada halaman, dapat melihat daftar kategori yang telah dimiliki oleh *database sistem*.



Gambar 4. 9 Perancangan Halaman Kategori (Admin)

10. Halaman Detail Dataset Kategori (Admin)

Pada halaman ini admin dapat melihat detail dari kategori yang telah tersedia di *database sistem*, yang terdiri kode, nama kategori, jumlah dataset.



Gambar 4. 10 Perancangan Halaman Detail Kategori (Admin)

11. Halaman *add new* dataset (Admin)

Halaman ini berfungsi untuk menambah *dataset* yang baru, sehingga jumlah kategori yang dapat dipilih lebih banyak variasinya.

The screenshot shows a web interface for an administrator. On the left is a vertical navigation menu with the following items: Dashboard, Data Pokok, LDA, Proposal, Admin Account, and Logout. The main content area is titled 'Add New Dataset' and includes a 'Back' button. The form contains three input fields: 'Kategori', 'Label', and 'Konten Dataset', each with a corresponding label. Below the form are 'Save' and 'Reset' buttons. In the top right corner, there is a circular 'Admin' user indicator.

Gambar 4. 11 Perancangan Halaman Add New Dataset (Admin)

12. Halaman *Detail* Dataset (Admin)

Pada halaman ini, admin dapat mengetahui lebih detail terkait kategori dataset beserta jumlah datasetnya, apabila terdapat ketidakcocokan dapat dihapus.

The screenshot shows a web interface for an administrator. On the left is a vertical navigation menu with the following items: Dashboard, Data Pokok, LDA, Proposal, Admin Account, and Logout. The main content area is titled 'Detail Dataset' and includes a 'Back' button. The form is divided into two sections. The top section, titled 'Kategori', contains three input fields: 'Kode', 'Nama Kategori', and 'Jumlah Dataset', each with a corresponding label. The bottom section, titled 'Detail Dataset Kategori', contains two input fields: 'Label' and 'Konten', each with a corresponding label. Below the second section is a 'Delete' button. In the top right corner, there is a circular 'Admin' user indicator.

Gambar 4. 12 Perancangan Halaman Detail Dataset (Admin)

13. Halaman *Training* (Admin)

Pada halaman ini admin dapat melihat data kategori yang terdiri dari id, kode, nama, jumlah dataset, LDA indeks, dan LDA Term. Untuk melakukan *training* dapat dengan menekan tombol paling bawah yang tertera pada halaman ini.

ID ▲	Kode ⇅	Nama	Jumlah Dataset ▼	LDA Indeks ▼	LDA Terms
xxx	xxx	xxx	xxx	xxx	xxx
xxx	xxx	xxx	xxx	xxx	xxx
xxx	xxx	xxx	xxx	xxx	xxx
xxx	xxx	xxx	xxx	xxx	xxx
xxx	xxx	xxx	xxx	xxx	xxx

START TRAINING

Gambar 4. 13 Perancangan Halaman *Training* (Admin)

14. Halaman *Proposal Mahasiswa* (Admin)

Pada halaman ini admin dapat melihat daftar proposal yang telah di daftar kan oleh mahasiswa.

ID ▲	NPM ⇅	Nama	Jenis Kelami ▼	Jumlah Proposa ▼	Proposal Menungg	Aksi
xxx	xxx	xxx	xxx	xxx	xxx	Next
xxx	xxx	xxx	xxx	xxx	xxx	Next
xxx	xxx	xxx	xxx	xxx	xxx	Next
xxx	xxx	xxx	xxx	xxx	xxx	Next
xxx	xxx	xxx	xxx	xxx	xxx	Next

Gambar 4. 14 Perancangan Halaman *Proposal Mahasiswa* (Admin)

15. Halaman *Detail Proposal Mahasiswa* (Admin)

Pada halaman ini admin dapat melihat proposal yang telah diajukan mahasiswa secara lebih detail secara satu per satu.

ID	Judul	Status	Kategori	Topik Kecocokan	Peluang Diterima	Aksi
xxx	xxx	xxx	xxx	xxx	xxx	Aksi
xxx	xxx	xxx	xxx	xxx	xxx	Aksi
xxx	xxx	xxx	xxx	xxx	xxx	Aksi

Gambar 4. 15 Halaman Detail Proposal Mahasiswa (Admin)

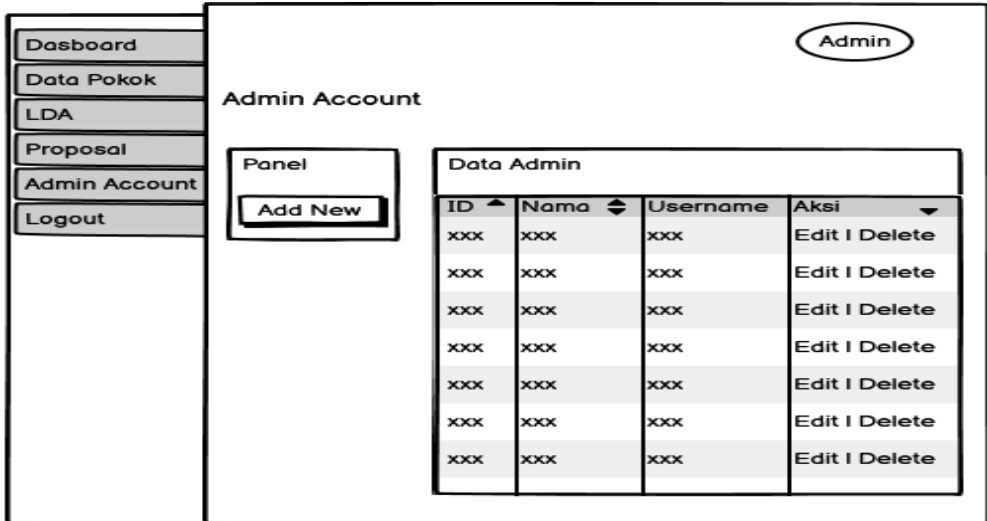
16. Halaman *Add New Proposal* (Admin)

Halaman ini dapat membantu admin untuk menambahkan proposal baru milik mahasiswa tanpa perlu menggunakan akun mahasiswa

Gambar 4. 16 Halaman Add New Proposal (Admin)

17. Halaman *Admin Account* (Admin)

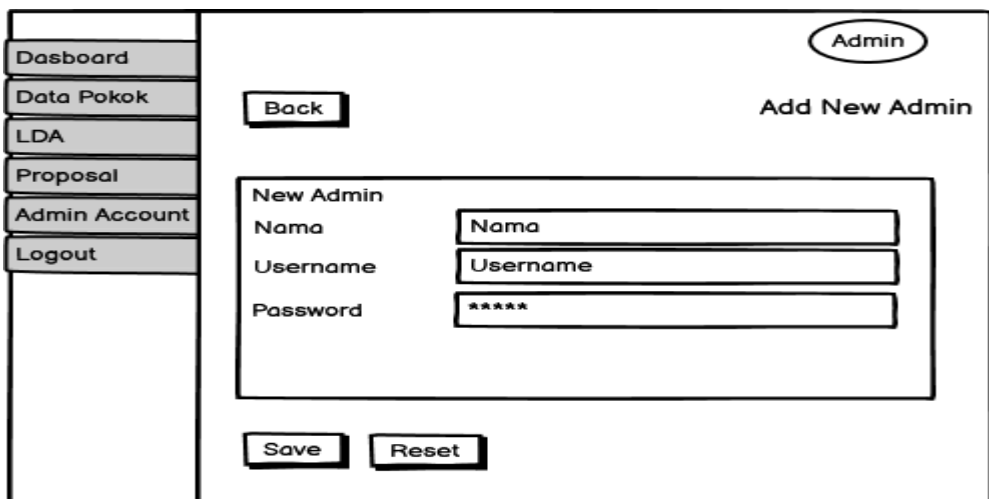
Halaman ini berisi daftar akun yang memiliki level sebagai admin.



Gambar 4. 17 Perancangan Halaman Admin Account (Admin)

18. Halaman *Add New Admin* (Admin)

Halaman ini dapat digunakan untuk membuat akun baru yang memiliki level sebagai admin.



Gambar 4. 18 Perancangan Halaman Add New Admin

19. Halaman Edit Admin (Admin)

Halaman ini untuk melakukan pembaharuan pada akun dengan level admin yang telah ada.

Admin

Back

Edit Admin

Edit Admin

Nama

Username

Password

Save

Gambar 4. 19 Perancangan Halaman Edit Admin

20. Halaman Dashboard (Mahasiswa)

Pada halaman ini mahasiswa dapat melihat data diri dan riwayat proposal yang telah diajukan.

Dashboard

Mhs

Dashboard

Profil Mahasiswa

NPM : NPM

Nama : Nama

Jenis Kelamin JK

Riwayat Proposal

ID	Judul	Kategori	Status
xxx	xxx	xxx	xxx
xxx	xxx	xxx	xxx
xxx	xxx	xxx	xxx
xxx	xxx	xxx	xxx
xxx	xxx	xxx	xxx
xxx	xxx	xxx	xxx
xxx	xxx	xxx	xxx

Gambar 4. 20 Perancangan Dashboard (Mahasiswa)

21. Halaman Detail Proposal Mahasiswa (Mahasiswa)

Pada halaman ini mahasiswa dapat melihat lebih detail terkait judul proposal yang diajukan sebelumnya.

Detail Proposal Mahasiswa							
Mahasiswa							
NPM :	<input type="text" value="NPM"/>						
Nama :	<input type="text" value="Nama"/>						
Jenis Kelamin	<input type="text" value="JK"/>						
Jumlah Prop	<input type="text" value="Jumlah Proposal"/>						
Riwayat Proposal							
ID	Jud	Pendahulu	Kateg	Statu	Kecocok Topik	Pelua	Aksi
xx	xxx	xxx	xxx	xxx	xxx	xxx	Detail
xx	xxx	xxx	xxx	xxx	xxx	xxx	Deta
xx	xxx	xxx	xxx	xxx	xxx	xxx	Detail

Gambar 4. 21 Perancangan Halaman Detail Proposal (Mahasiswa)

22. Halaman Add New Proposal (Mahasiswa)

Pada halaman ini mahasiswa dapat mengajukan judul proposal secara mandiri tanpa perlu menghubungi admin terlebih dahulu.

Add New Proposal	
New Proposal	
NPM	<input type="text" value="NPM"/>
Nama	<input type="text" value="Nama"/>
Judul Proposal	<input type="text" value="Judul Proposal"/>
Pendahuluan	<input type="text" value="Pendahuluan"/>
Kategori	<input type="text" value="Kategori"/>
Kecocokan Topik	<input type="text" value="%"/>

Gambar 4. 22 Perancangan Add New Proposal (Mahasiswa)

23. Halaman Detail Proposal (Mahasiswa)

Pada halaman ini mahasiswa dapat melihat status dari judul skripsi yang telah diajukan secara detail, serta dapat mengetahui status dari judul yang diberikan dan melihat saran yang diberikan admin terkait judul yang diajukan apabila ada.

Dashboard	Mhs	Detail Proposal
Proposal		
Profil		
Logout		
Back		
Detail Proposal		
NPM	NPM	
Nama	Nama	
Judul Proposal	Judul Proposal	
Pendahuluan	Pendahuluan	
Kategori	Kategori	
Kecocokan Topik	%	
Status	Status	
Saran	Saran	

Gambar 4. 23 Perancangan Halaman Detail Proposal (Mahasiswa)

24. Halaman Profil (Mahasiswa)

Pada halaman ini mahasiswa dapat melihat profil pribadi dari akun mahasiswa terkait.

The image shows a web interface for a student profile. On the left is a vertical sidebar with four buttons: 'Dashboard', 'Proposal', 'Profil', and 'Logout'. The main area is titled 'Profil' and contains a form titled 'Profil Mahasiswa'. The form has four input fields: 'NPM', 'Nama', 'Jenis Kelamin', and 'Password'. The 'Password' field contains six asterisks. In the top right corner of the main area, there is a small oval containing the text 'Mhs'.

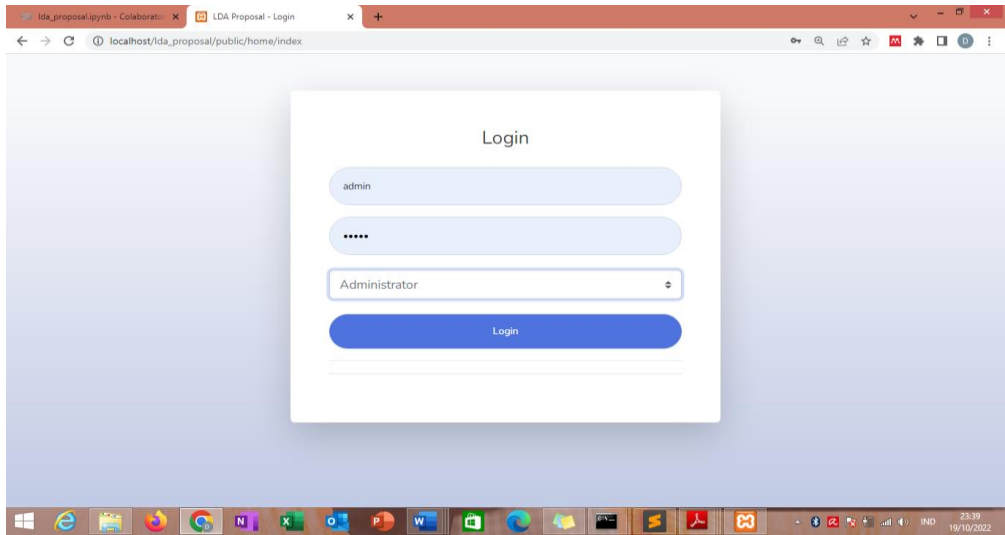
Gambar 4. 24 Perancangan Halaman Profil (Mahasiswa)

D. Implementasi Sistem

Implementasi dari algoritma dan perancangan *user interface* yang telah disajikan pada sub bab sebelumnya, sebagai berikut :

1. Perancangan Halaman Logim

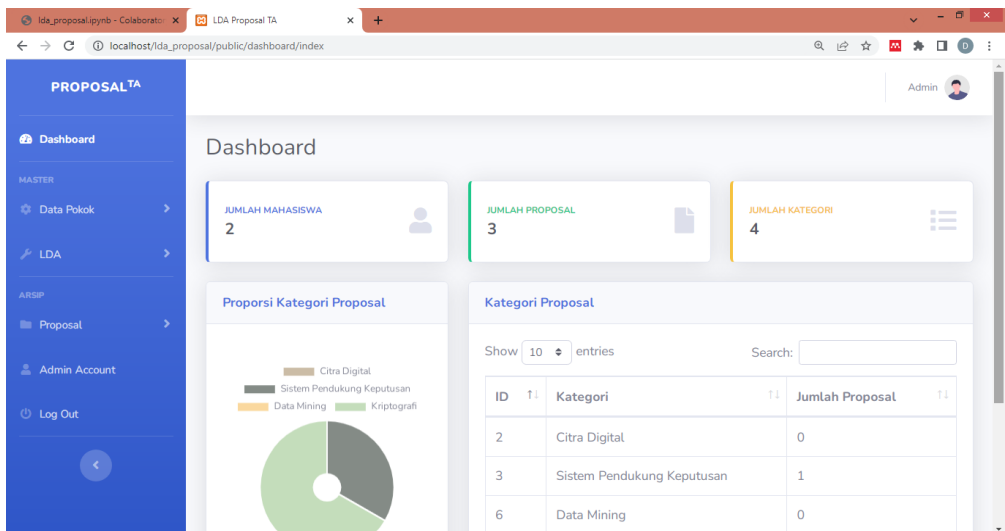
Halaman ini digunakan oleh *user* dan *admin* untuk *login* kedalam sistem. *User* dan *admin* mengisi *field username* dan *password* kemudian memilih *level login*. Jika *login* berhasil maka *user* dan *admin* akan diarahkan ke halaman *dashboard* masing-masing. Berikut ini adalah tampilan halaman *login* pada sistem yang telah dibangun:



Gambar 4. 25 Halaman Login

2. Perancangan Halaman *Dashboard* (Admin)

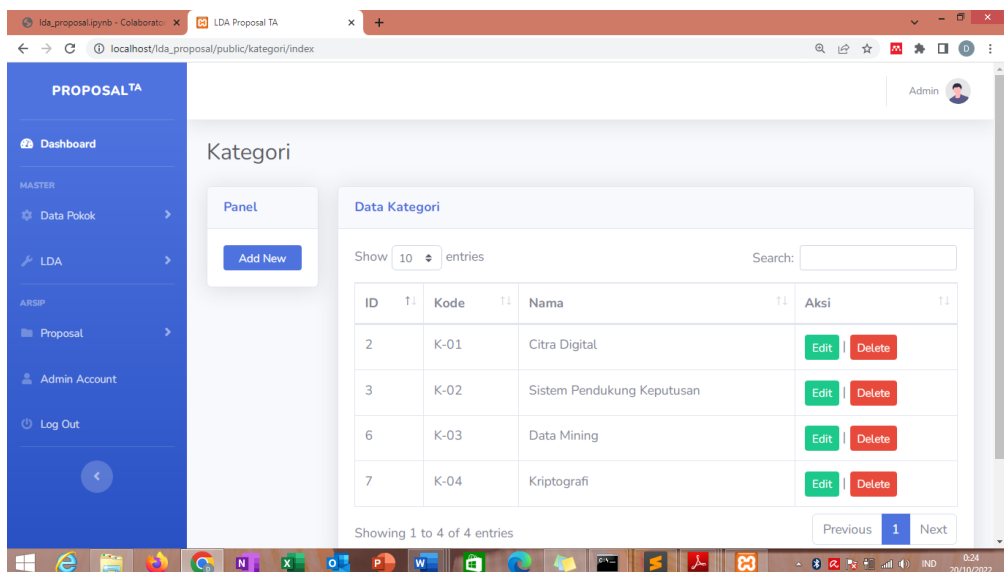
Halaman *dashboard admin* berisi informasi terkait jumlah akun mahasiswa, jumlah proposal serta jumlah kategori yang telah masuk ke dalam *database* sistem yang disajikan dalam bentuk *card*. Tersedia juga informasi berupa *pie chart* terkait proporsi kategori proposal.



Gambar 4. 26 Halaman Dashboard (Admin)

3. Perancangan Halaman Kategori (Admin)

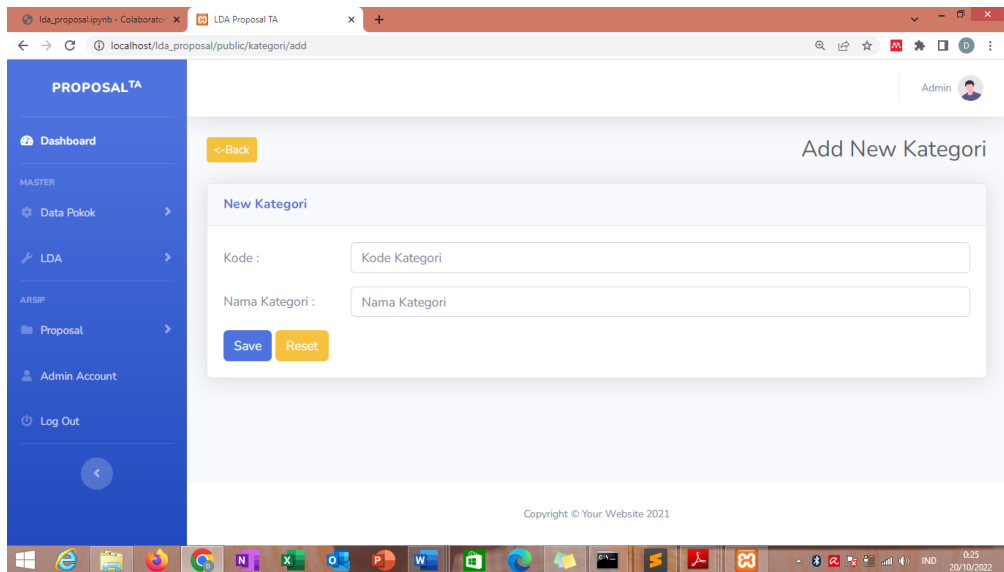
Pada halaman Kategori *admin* akan disajikan oleh daftar kategori yang telah masuk kedalam sistem. Pada halaman kategori ini juga terdapat fitur untuk menambahkan kategori, mengubah, dan menghapus. Untuk menambahkan dan mengedit kategori, nantinya *admin* akan diarahkan pada halaman masing-masing.



Gambar 4. 27 Halaman Kategori (Admin)

4. Halaman New Kategori (Admin)

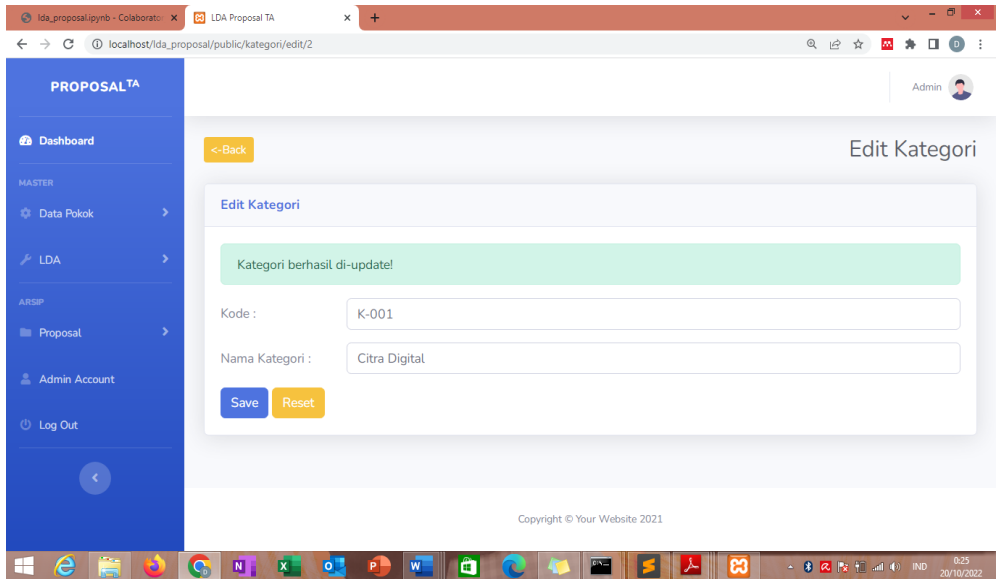
Halaman *New Kategori* berfungsi untuk menambahkan kategori terkait judul proposal skripsi. Pada halaman ini *admin* akan diminta untuk memasukkan Kode Kategori dan Nama Kategori. Tombol *Save* dapat digunakan oleh *admin* untuk menyimpan kategori baru yang akan ditambahkan kedalam sistem. Tombol *Reset* dapat digunakan untuk membatalkan aksi jika *admin* tidak ingin atau urung menambahkan kategori kedalam sistem. Tampilan halaman *New Kategori* dapat dilihat pada Gambar 4. 28 dibawah ini:



Gambar 4. 28 Halaman New Kategori (Admin)

5. Halaman Edit Kategori (Admin)

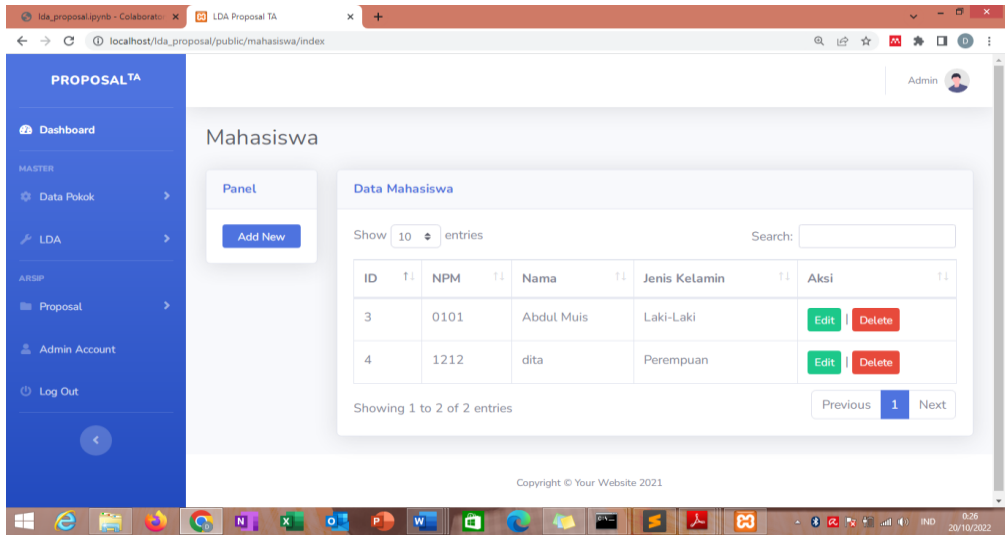
Secara tampilan, Halaman Edit Kategori ini mirip dengan Halaman New Kategori, namun secara fungsi memiliki perbedaan. Halaman Edit Kategori berfungsi untuk melakukan perubahan pada kategori yang telah masuk kedalam sistem. Pada halaman ini *admin* dapat mengganti Kode dan Nama Kategori sesuai dengan perbaikan yang dibutuhkan. Terdapat Tombol *Save* untuk menyimpan perubahan dan tombol *Reset* untuk membatalkan aksi jika *admin* tidak ingin melakukan perubahan ataupun *update* pada kategori yang sudah ada. Tampilan halaman Edit Kategori dapat dilihat pada Gambar 4. 29.



Gambar 4. 29 Halaman Edit Kategori (Admin)

6. Halaman Data Mahasiswa (Admin)

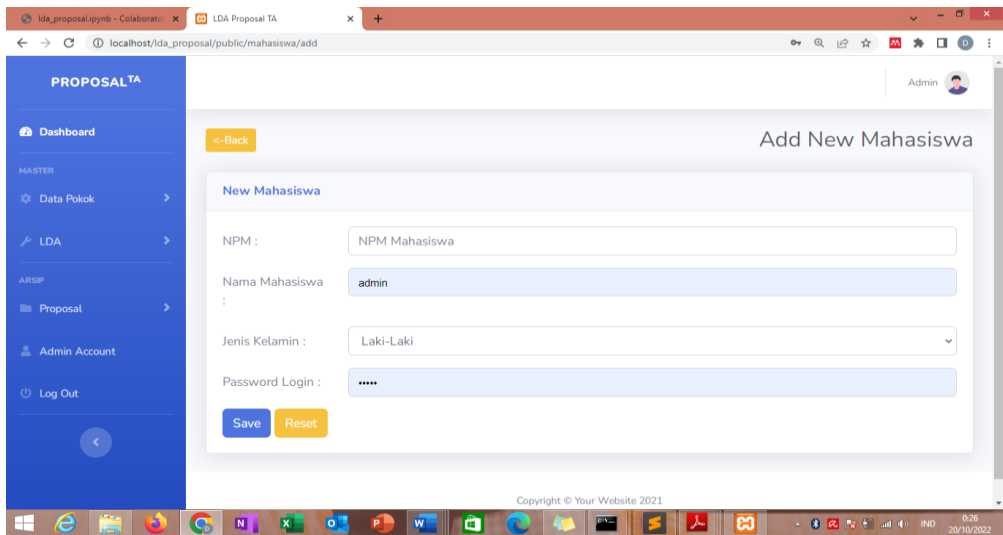
Halaman Data Mahasiswa merupakan halaman untuk melihat daftar mahasiswa yang telah memiliki akun pada sistem. Daftar mahasiswa disajikan dalam bentuk tabel, dan terdapat tombol aksi *edit* dan *delete* pada *field* aksi. Aksi *edit* akan mengarahkan *admin* ke halaman *edit Data Mahasiswa* dan aksi *delete* akan mengarahkan *admin* pada *dialog box* konfirmasi penghapusan akun mahasiswa. Pada halaman ini juga *admin* dapat menambahkan akun mahasiswa dengan menggunakan tombol *Add New* yang terdapat dibagian sebelah kiri dari tabel. Nantinya *admin* akan diarahkan ke halaman *Add New Mahasiswa* untuk melakukan penambahan data akun mahasiswa. Tampilan halaman Data Mahasiswa dapat dilihat pada Gambar 4. 30 dibawah ini.



Gambar 4. 30 Halaman Data Mahasiswa (Admin)

7. Haman *Add New* Data Mahasiswa (Admin)

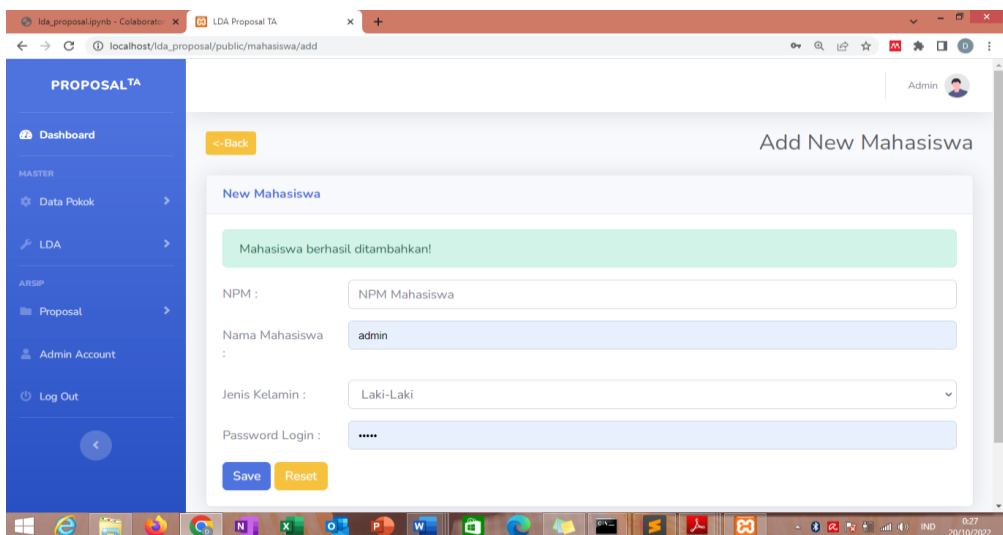
Halaman *add new* data mahasiswa merupakan halaman yang digunakan untuk menambahkan data akun mahasiswa. dapat membantu admin untuk menambahkan data mahasiswa. Data akun mahasiswa yang dapat isikan oleh diantaranya adalah NIM, Nama Mahasiswa, Jenis Kelamin, dan *password default* yang digunakan oleh mahasiswa untuk *login* kedalam sistem. Mahasiswa yang dapat *login* dan menggunakan sistem ini adalah mahasiswa yang telah terdaftar didalam sistem. Tampilan halaman *add new* data mahasiswa dapat dilihat pada Gambar 4. 31 dibawah ini.



Gambar 4. 31 Halaman New Data Mahasiswa (Admin)

8. Halaman Edit Data Mahasiswa (Admin)

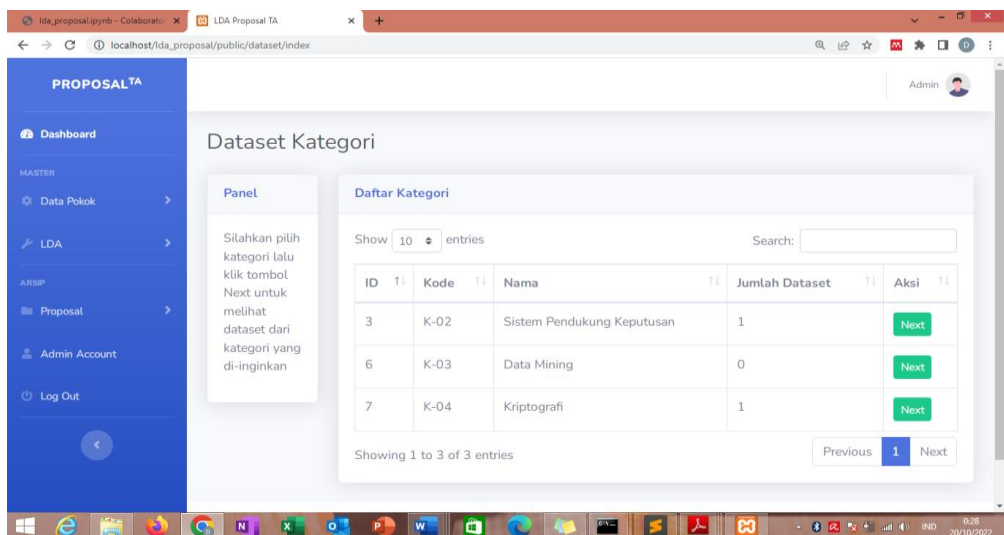
Halaman ini digunakan oleh admin untuk melakukan perubahan data akun mahasiswa yang telah terdaftar kedalam sistem. Perubahan data yang dilakukan berupa NIM, Nama, Jenis Kelamin, dan *password* yang digunakan mahasiswa untuk *login* kedalam sistem.



Gambar 4. 32 Halaman Edit Data Mahasiswa (Admin)

9. Halaman Dataset Kategori (Admin)

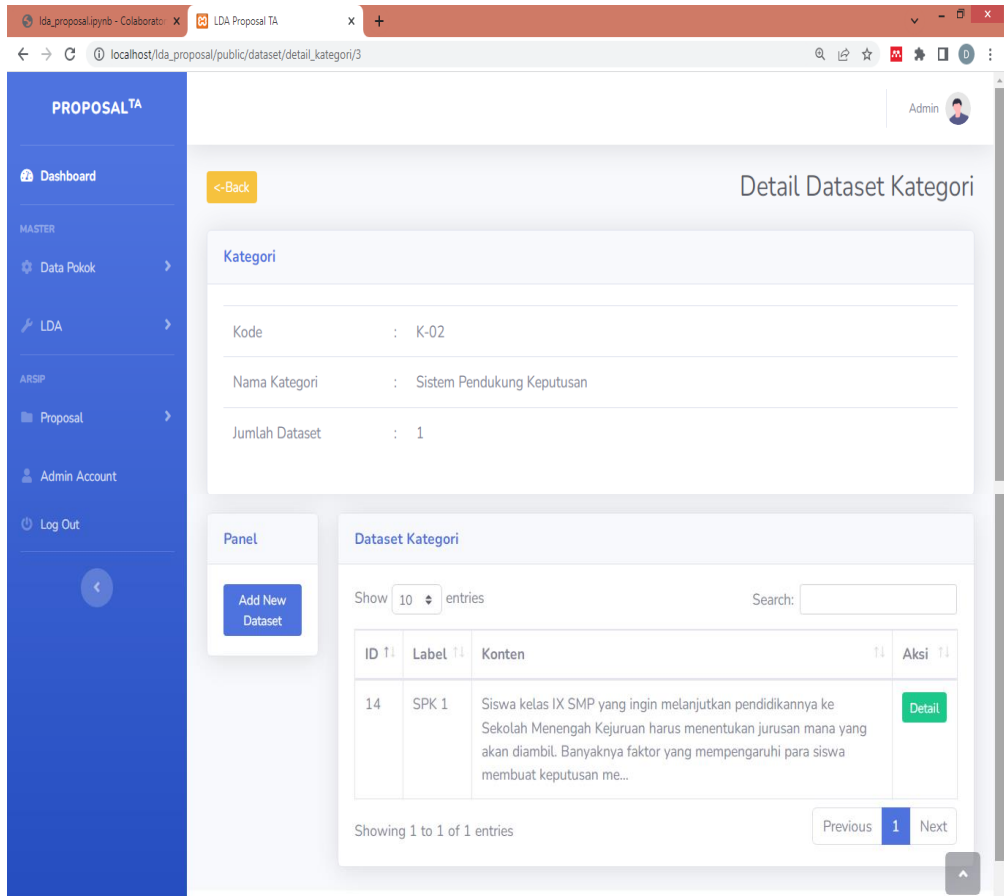
Pada halaman ini *admin* disajikan oleh sebuah tabel yang berisi Dataset dari Kategori Proposal. Pada halaman ini dapat dilihat jumlah *dataset* untuk setiap kategori. Pada Tabel Daftar Dataset Kategori, terdapat sebuah tombol aksi *Next* yang akan mengarahkan *admin* ke halaman Detail Dataset Kategori. Tampilan halaman Dataset Kategori dapat dilihat pada Gambar 4. 33 dibawah ini.



Gambar 4. 33 Halaman Dataset Kategori (Admin)

10. Halaman Detail Dataset Kategori (Admin)

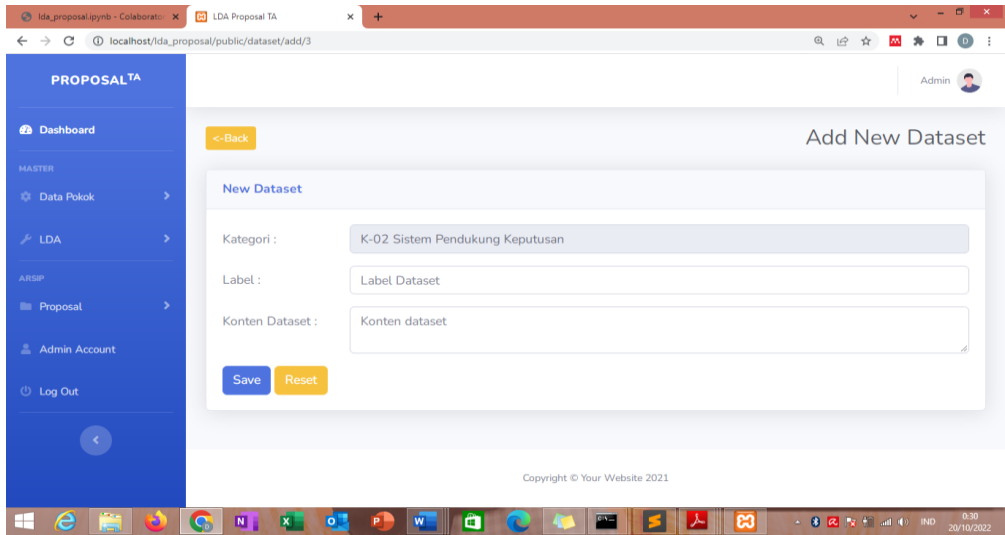
Pada halaman ini admin dapat melihat detail dari kategori yang telah tersedia di *database* sistem, yang terdiri kode, nama kategori, jumlah dataset. Pada halaman ini akan menyajikan seluruh *dataset* pada kategori yang telah dipilih. Pada tabel daftar Konten dari kategori *dataset* terdapat sebuah tombol aksi *Detail*, tombol ini akan mengarahkan *admin* pada halaman detail dari *entry dataset*. Pada halaman ini juga terdapat tombol *Add New Dataset* yang digunakan untuk menuju ke halaman tambah *dataset*. Tampilan halaman Detail *Dataset* Kategori dapat dilihat pada Gambar 4. 34 dibawah ini.



Gambar 4. 34 Halaman Detail Dataset Kategori (Admin)

11. Halaman *add new dataset* (Admin)

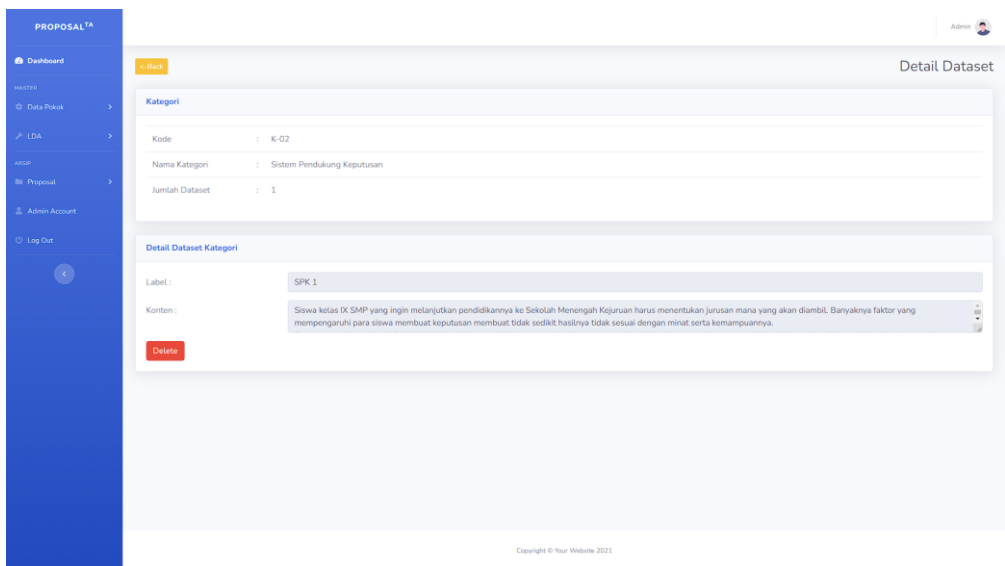
Halaman ini berfungsi untuk menambah *dataset* yang baru, pada kategori yang telah dipilih, sehingga jumlah *dataset* pada kategori yang dapat dipilih lebih banyak varian. Sama seperti dengan halaman *Add New* lainnya, halaman ini juga dilengkapi dengan tombol *Save* untuk menyimpan penambahan data dan tombol *Reset* untuk membatalkan. Tampilan halaman *New Dataset* dapat dilihat pada gambar 4. 34 dibawah ini.



Gambar 4. 35 Halaman New Dataset (Admin)

12. Halaman *Detail* Dataset (Admin)

Pada halaman ini, *admin* dapat mengetahui lebih detail *dataset* dari kategori tertentu. Pada halaman ini *admin* disajikan oleh sebuah tampilan rinci dari *dataset*, seperti Kode, Nama Kategori, Jumlah Dataset pada kategori tersebut, Label *Dataset*, dan Konten dari *Dataset*.



Gambar 4. 36 Halaman Detail Dataset (Admin)

13. Halaman *Training* (Admin)

Pada halaman ini admin dapat melihat data kategori yang terdiri dari id, kode, nama, jumlah dataset, LDA indeks, dan LDA Term. Untuk melakukan *training* dapat dengan menekan tombol paling bawah yang tertera pada halaman ini.

The screenshot shows the 'Training' page in the PROPOSAL TA system. The page features a sidebar menu on the left with options like Dashboard, Data Pokok, LDA, and Admin Account. The main content area displays a table titled 'Data Kategori' with the following data:

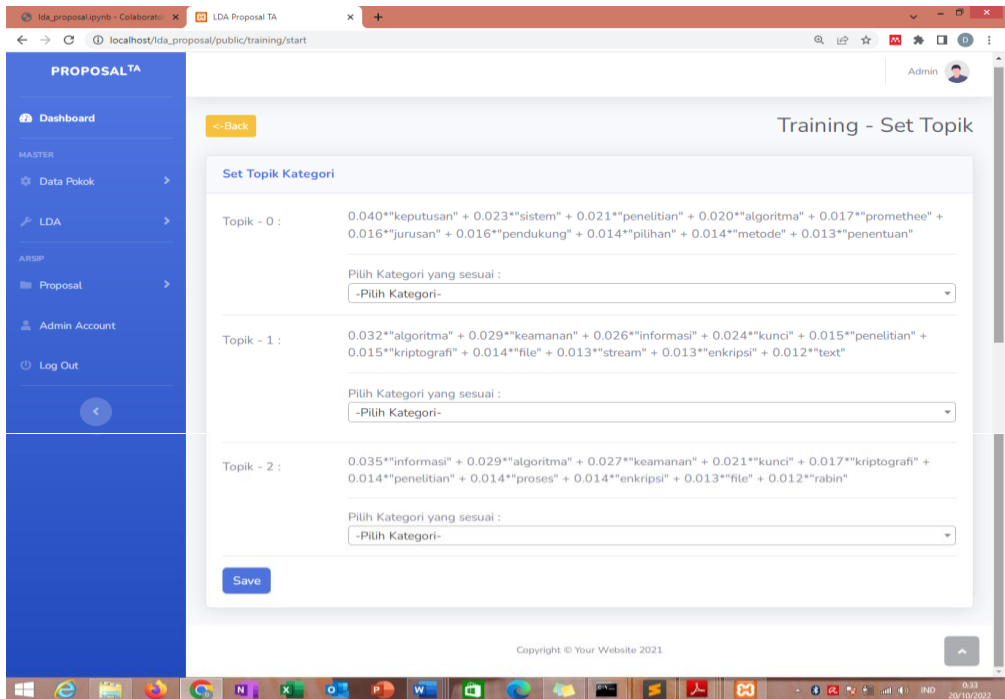
ID	Kode	Nama	Jumlah Dataset	LDA Indeks	LDA Terms
3	K-02	Sistem Pendukung Keputusan	2	0	0.028**keputusan" + 0.026**algoritma" + 0.019**sistem" + 0.019**penelitian" + 0.016**informasi" + 0.016**keamanan" + 0.013**jurusan" + 0.013**promethee" + 0.012**pilihan" + 0.011**kunci"
6	K-03	Data Mining	0	-1	
7	K-04	Kriptografi	1	1	0.028**algoritma" + 0.025**informasi" + 0.022**keamanan" + 0.022**kunci" + 0.015**penelitian" + 0.013**file" + 0.013**kriptografi" + 0.012**proses" + 0.012**enkripsi" + 0.011**hybrid"

Below the table, there is a 'Start Training' button. The page also includes a search bar and pagination controls.

Gambar 4. 37 Halaman Training Dataset (Admin)

14. Halaman *Set Topik*

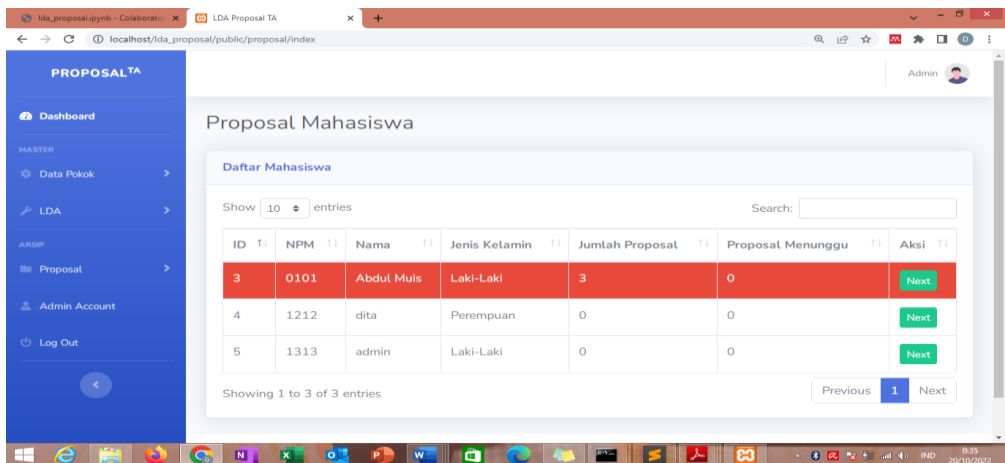
Pada halaman ini admin dapat menentukan topik dari setiap judul yang telah di daftarkan mahasiswa berdasarkan probabilitas dan LDA yang dimiliki oleh setiap judul



Gambar 4. 38 Halaman Set Topik

15. Halaman Proposal Mahasiswa (Admin)

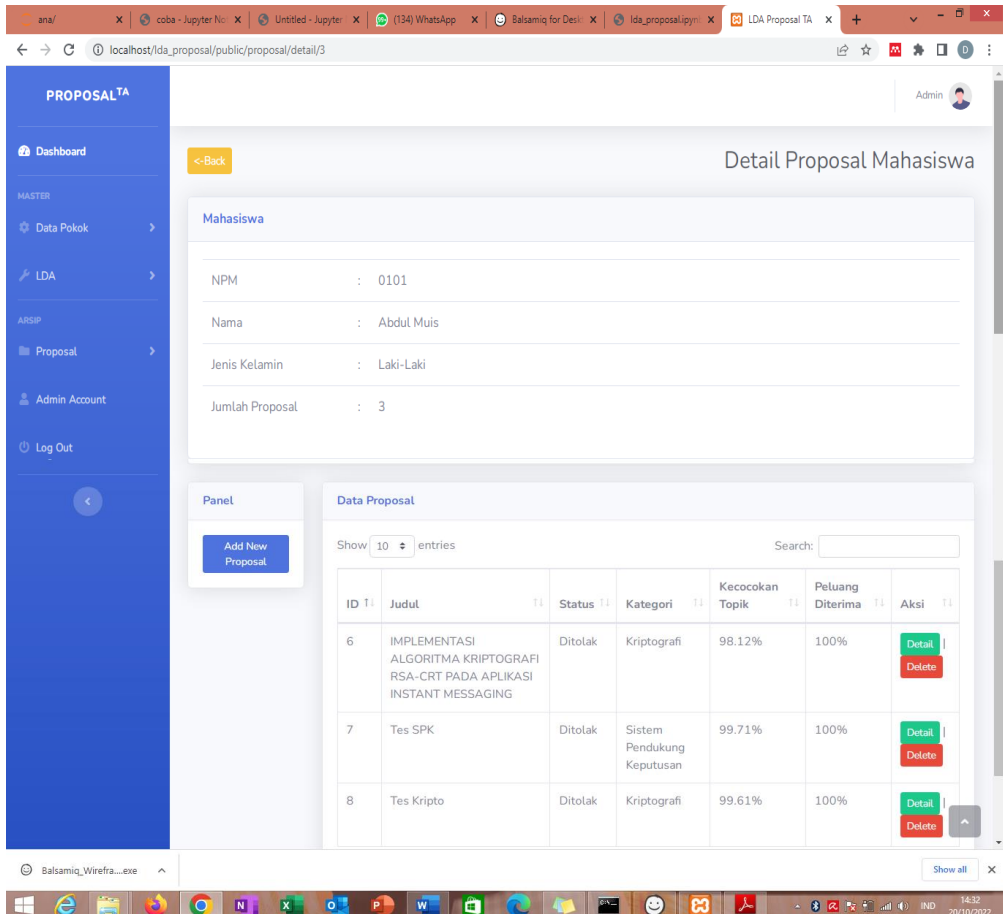
Pada halaman ini admin dapat melihat daftar proposal yang telah di daftar kan oleh mahasiswa.



Gambar 4. 39 Halaman Proposal Mahasiswa (Admin)

16. Halaman Detail Proposal Mahasiswa (Admin)

Pada halaman ini admin dapat melihat proposal yang telah diajukan mahasiswa secara lebih detail secara satu per satu.



The screenshot shows a web application interface for an admin. The left sidebar is blue and contains a menu with items: Dashboard, Data Pokok, LDA, Proposal, Admin Account, and Log Out. The main content area is titled 'Detail Proposal Mahasiswa' and features a 'Mahasiswa' section with the following details:

- NPM : 0101
- Nama : Abdul Muis
- Jenis Kelamin : Laki-Laki
- Jumlah Proposal : 3

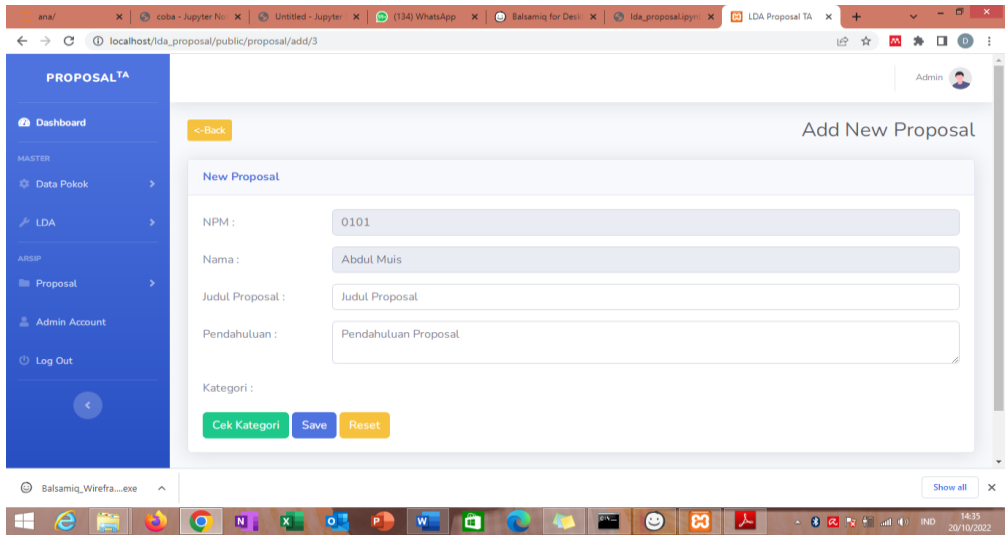
Below this is a 'Data Proposal' section with a table. The table has columns: ID, Judul, Status, Kategori, Kecocokan Topik, Peluang Diterima, and Aksi. There are three rows of data:

ID	Judul	Status	Kategori	Kecocokan Topik	Peluang Diterima	Aksi
6	IMPLEMENTASI ALGORITMA KRIPTOGRAFI RSA-CRT PADA APLIKASI INSTANT MESSAGING	Ditolak	Kriptografi	98.12%	100%	Detail, Delete
7	Tes SPK	Ditolak	Sistem Pendukung Keputusan	99.71%	100%	Detail, Delete
8	Tes Kripto	Ditolak	Kriptografi	99.61%	100%	Detail, Delete

Gambar 4. 40 Halaman Detail Proposal (Admin)

17. Halaman Add New Proposal (Admin)

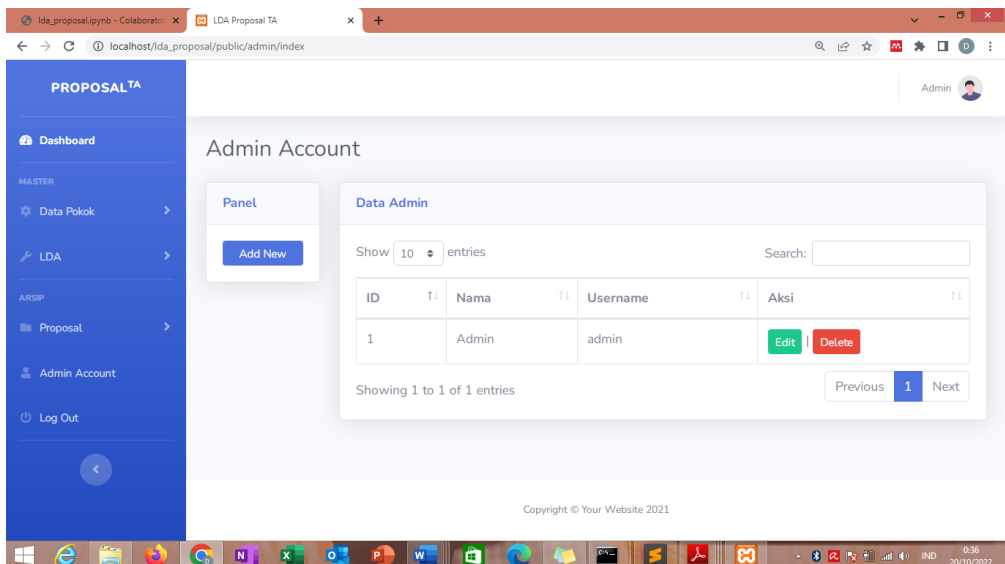
Halaman ini dapat membantu admin untuk menambahkan proposal baru milik mahasiswam tanpa perlu menggunakan akun mahasiswa



Gambar 4. 41 Halaman New Proposal (Admin)

18. Halaman Admin Account (Admin)

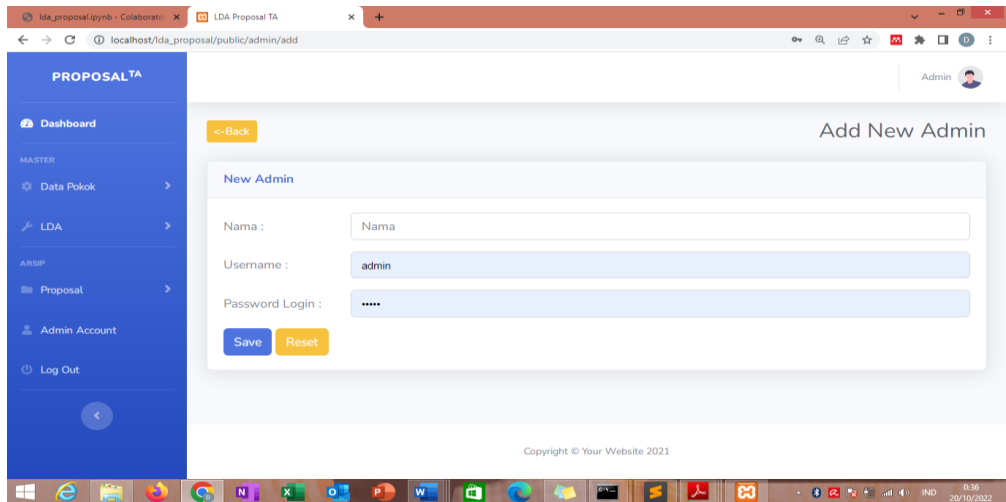
Halaman ini berisi daftar akun yang memiliki level sebagai admin.



Gambar 4. 42 Halaman Admin Account (Admin)

19. Halaman *Add New Admin* (Admin)

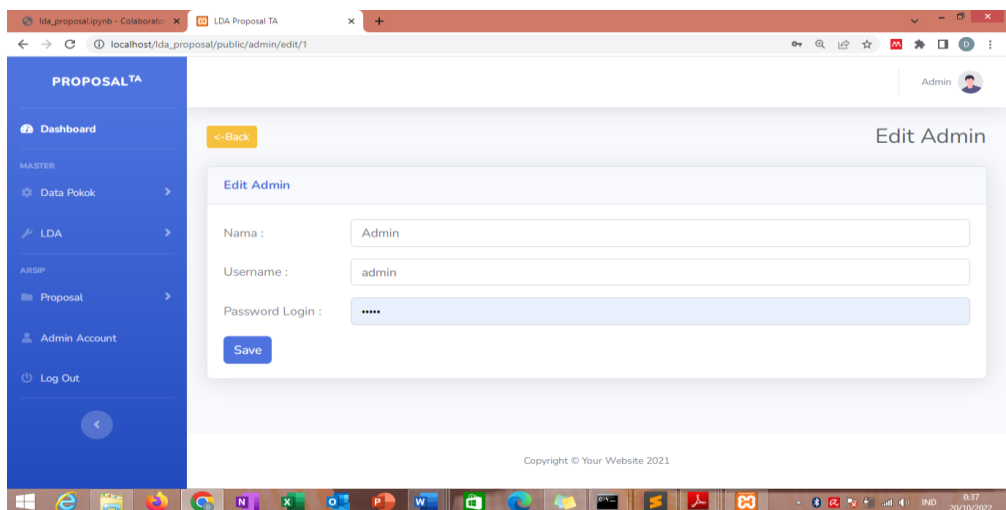
Halaman ini dapat digunakan untuk membuat akun baru yang memiliki level sebagai admin.



Gambar 4. 43 Halaman New Admin(Admin)

20. Halaman *Edit Admin* (Admin)

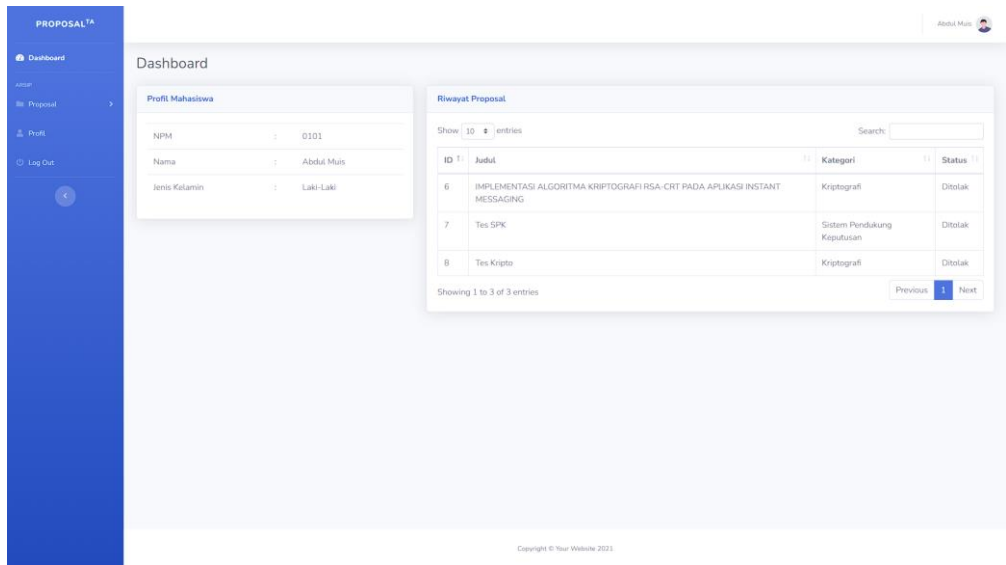
Halaman ini untuk melakukan pembaharuan pada akun dengan level admin yang telah ada.



Gambar 4. 44 Halaman Edit Admin Account (Admin)

21. Halaman Dashboard (Mahasiswa)

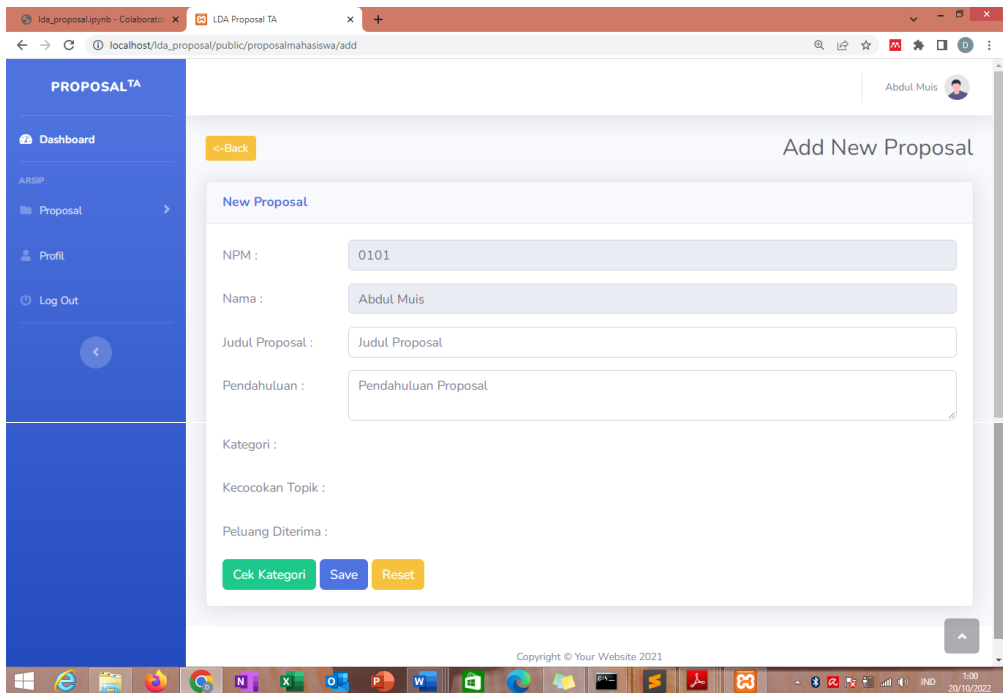
Pada halaman ini mahasiswa dapat melihat informasi ringkas dari data diri dan riwayat proposal yang telah diajukan. Data yang disajikan dalam bentuk tabel yang disusun secara horizontal.



Gambar 4. 45 Halaman Dashboard (Mahasiswa)

22. Halaman Add New Proposal (Mahasiswa)

Pada halaman ini mahasiswa disajikan dengan data riwayat dan hasil pengajuan dari proposalnya. Pada tabel Data Proposal terdapat Judul, Pendahuluan, Kategori, Status, Kecocokan Topik, Peluang, dan Aksi. Pada kolom Aksi terdapat tombol *Detail* yang akan mengarahkan *user* mahasiswa ke halaman *detail* dari proposalnya.



Gambar 4. 46 Halaman New Proposal (Mahasiswa)

23. Halaman Detail Proposal (Mahasiswa)

Pada halaman ini mahasiswa dapat melihat status dari judul skripsi yang telah diajukan secara detail, serta dapat mengetahui status dari judul yang diberikan dan melihat saran yang diberikan admin terkait judul yang diajukan apabila ada.

PROPOSAL TA

Abdul Muis

Detail Proposal Mahasiswa

Mahasiswa

NPM : 0101

Nama : Abdul Muis

Jenis Kelamin : Laki-Laki

Jumlah Proposal : 3

Panel

Add New Proposal

Data Proposal

Show 10 entries Search:

ID	Judul	Pendahuluan	Kategori	Status	Kecocokan Topik	Peluang
6	IMPLEMENTASI ALGORITMA KRIPTOGRAFI RSA-CRT PADA APLIKASI INSTANT MESSAGING	Komunikasi merupakan elemen terpenting dari manusia sebagai makhluk sosial. Salah satu cara berkomunikasi yaitu dengan surat menyurat. Di era globalisasi, surat menyurat telah	Kriptografi	Ditolak	98.12 %	100 %
8	Tes Kripto	Informasi sangat penting bagi kehidupan, ada yang bersifat rahasia yaitu informasi yang ingin dilindungi dan disembunyikan oleh pemilik informasi tersebut. Aktivitas penyimpanan data dan pertukaran in...	Kriptografi	Ditolak	99.61 %	100 %

Showing 1 to 3 of 3 entries

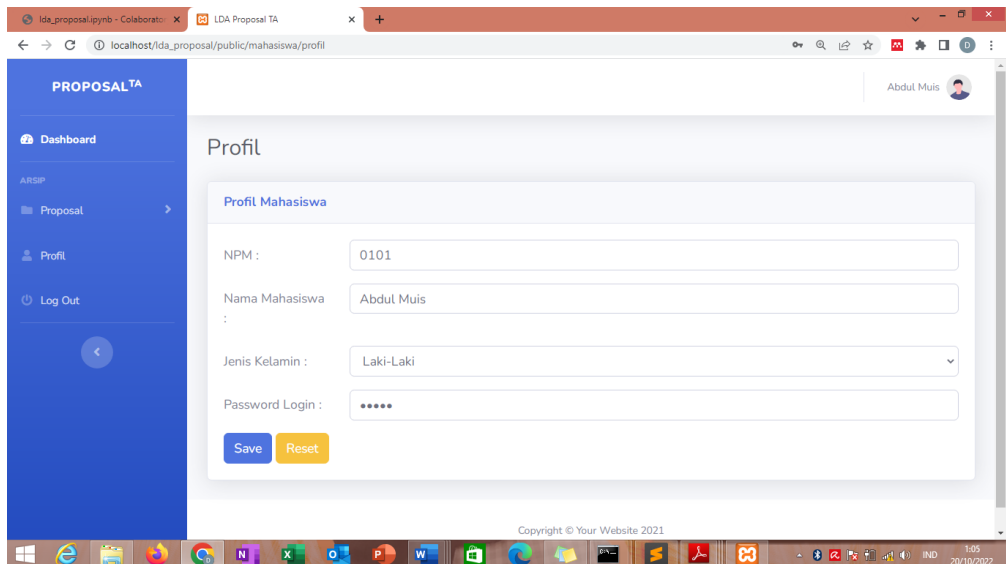
Previous 1 Next

Copyright © Your Website 2021

Gambar 4. 47 Halaman Detail Proposal (Mahasiswa)

24. Halaman Profil (Mahasiswa)

Pada halaman ini mahasiswa dapat melihat profil pribadi dari akun mahasiswa terkait.



Gambar 4. 48 Halaman Profil (Mahasiswa)

E. Penerapan

Setelah dilakukannya implementasi kedalam bentuk sistem, maka sistem yang dibuat dapat diterapkan pada lingkungan program studi ilmu komputer UIN-SU Medan dalam melakukan penyeleksian judul skripsi mahasiswa. Penerapan dilakukan setelah seluruh *database* judul proposal yang ada di Prodi Ilmu Komputer UIN-SU telah diinput kedalam sistem dan menjadi *dataset*.

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil penelitian dengan judul “Penerapan Text Mining Pada Sistem Penyeleksian Judul Skripsi Mahasiswa Menggunakan Algoritma *Latent Dirichlet Allocation* Di Program Studi Ilmu Komputer UIN Sumatera Utara Medan” didapat kesimpulan bahwa algoritma *Latent Dirichlet Allocation* dapat digunakan untuk menentukan topik dari judul skripsi yang diajukan mahasiswa. Mahasiswa dapat mengetahui secara langsung kesesuaian topik dan peluang diterimanya setiap judul proposal yang diajukan. Dengan penggunaan sistem ini, proses penyeleksian judul dapat di lakukan dalam waktu yang singkat.

B. Saran

Berdasarkan hasil penelitian dengan judul “Penerapan Text Mining Pada Sistem Penyeleksian Judul Skripsi Mahasiswa Menggunakan Algoritma *Latent Dirichlet Allocation* Di Program Studi Ilmu Komputer UIN Sumatera Utara Medan”, adapun saran yang dapat disampaikan agar dilakukan penelitian dengan menggunakan data yang berbeda untuk membuktikan kemampuan algoritma *Latent Dirichlet Allocation* dalam menentukan topik serta dapat dilakukannya pengembangan pada sistem agar sekiranya mahasiswa dapat mengetahui kemungkinan dosen pembimbingnya dan sistem dapat memberikan referensi yang ditulis oleh dosen prodi untuk setiap topik yang diajukan.

Daftar Pustaka

- Arikunto, S. (2017). *Prosedur Penelitian Suatu Pendekatan Praktek*.
- Balya. (2019). *Analisis Sentimen Pengguna Youtube di Indonesia pada Review Smartphone Menggunakan Naïve Bayes*. <http://repositori.usu.ac.id/handle/123456789/23217>
- Blei, D. M. (2012). Probabilistic topic models. *Communications of the ACM*, 55(4), 77–84. <https://doi.org/10.1145/2133806.2133826>
- Buntoro, G. A. (2017). Analisis Sentimen Calon Gubernur DKI Jakarta 2017 Di Twitter. *Integer Journal*, 2(1), 32–41. <https://t.co/jrvaMsgBdH>
- Gilang Kencana, C., & Sibaroni, Y. (n.d.). *Klasifikasi Sentiment Analysis pada Review Buku Novel Berbahasa Inggris dengan Menggunakan Metode Support Vector Machine (SVM)*.
- Gunawan, B., Pratiwi, H. S., & Pratama, E. E. (2018). Sistem Analisis Sentimen pada Ulasan Produk Menggunakan Metode Naive Bayes. *JEPIN (Jurnal Edukasi Dan Penelitian Informatika)*, 4(2), 113–118.
- Hakim, D. M. (2019). *Optical Music Recognition Pada Citra Notasi Musik Menggunakan Convolutional Neural Network*. UNIKOM.
- Kurniawan, D. (2021). *Pengenalan Machine Learning dengan Python* (Ed. 2). PT Elex Media Komputindo.
- Listari. (2019). *Inisiasi Natural Language Processing (NLP) dan Klasifikasi Jenis Wisata Kuliner Untuk Program Chatbot (Studi Kasus: Informasi Wisata Kuliner Daerah Istimewa Yogyakarta)*. Universitas Islam Indonesia.
- Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2014). Foundations in Machine learning. In *SpringerBriefs in Computer Science* (Vol. 0, Issue 9783319056050).
- Muljono, Artanti, D. P., Syukur, A., Prihandono, A., & Setiadi, D. R. I. M. (2018). Analisa Sentimen Untuk Penilaian Pelayanan Situs Belanja Online Menggunakan Algoritma Naive Bayes. *Konferensi Nasional Sistem Informasi*, 165–170.

- Nugroho, D. D. A., & Alamsyah, A. (2018). Analisis Konten Pelanggan Airbnb Pada Network Sosial Media Twitter. *EProceedings of Management*, 1622–1628.
- Nurzahputra, A., & Muslim, A. (2016). Analisis Sentimen pada Opini Mahasiswa Menggunakan Natural Language Processing. *Seminar Nasional Ilmu Komputer*.
- Putra, K. B., & Kusumawardani, R. P. (2017). Analisis Topik Informasi Publik Media Sosial di Surabaya Menggunakan Pemodelan Latent Dirichlet Allocation (LDA). *Jurnal Teknik ITS*, 6(2). <https://doi.org/10.12962/j23373539.v6i2.23205>
- Putu, I., Wirayasa, M., Made, I., Wirawan, A., Pradnyana, A., Kunci, K., Stemming, :, Bali, B., & Bastal, A. (n.d.). *ALGORITMA BASTAL: ADAPTASI ALGORITMA NAZIEF & ADRIANI UNTUK STEMMING TEKS BAHASA BALI* (Vol. 8).
- Rahim, R., Zufria, I., Kurniasih, N., Simargolang, M. Y., Hasibuan, A., Sutiksno, D. U., Nanuru, R. F., Anamofa, J. N., Ahmar, A. S., & Achmad Daeng, G. S. (2018). C4.5 classification data mining for inventory control. *International Journal of Engineering and Technology(UAE)*, 7, 68–72. <https://doi.org/10.14419/ijet.v7i2.3.12618>
- Russell, S., & Norvig, P. (2021). Artificial Intelligence A Modern Approach (4th Edition). In *Pearson Series*.
- S, V., & R, J. (2016). Text Mining: open Source Tokenization Tools – An Analysis. *Advanced Computational Intelligence: An International Journal (ACII)*, 3(1), 37–47. <https://doi.org/10.5121/acii.2016.3104>
- Sevsa, B. A., & Wahyudi, M. D. R. (2019). Analisis Sentimen pada Indeks Kinerja Dosen Fakultas SAINTEK UIN Sunan Kalijaga Menggunakan Naive Bayes Classifier. *Jurnal Buana Informatika*, 10(2), 112–123.
- Sugiyono. (2016). Metode Penelitian Bisnis. Alfabeta: Bandung. *Jurnal Analisis*, 6(2).
- Sunardi, Fadlil, A., & Suprianto. (2018). Analisis Sentimen Menggunakan Metode Naive Bayes Classifier Pada Angket Mahasiswa. *SAINTEKBU: Jurnal Sains Dan Teknologi*, 10(2541–1942), 1–9.

Ulfah Siregar, Z., Ruli, R., Siregar, A., & Arianto, R. (2019).
*KLASIFIKASI SENTIMENT ANALYSIS PADA KOMENTAR
PESERTA DIKLAT MENGGUNAKAN METODE K-NEAREST
NEIGHBOR*. 8(1).

LAMPIRAN

SOURCE CODE APLIKASI

1. Anvil_Code

```
import anvil.server

# This is a server module. It runs on the Anvil server,
# rather than in the user's browser.
#
# To allow anvil.server.call() to call functions here,
we mark
# them with @anvil.server.callable.
# Here is an example - you can replace it with your own:
#
# @anvil.server.callable
# def say_hello(name):
#     print("Hello, " + name + "!")
#     return 42
#

@anvil.server.http_endpoint("/lda-tes",methods=["POST"])
def get_lda(jumlah_topik, **q):
    hasil = anvil.server.call('proses_lda',jumlah_topik)
    return hasil

@anvil.server.http_endpoint("/lda-cek",methods=["POST"])
def cek_lda(konten, **q):
    hasil = anvil.server.call('cek_kategori_lda',konten)
    return hasil

@anvil.server.http_endpoint("/clear-training")
def clear_training(**q):
    hasil = anvil.server.call('clear_training_data')
    return hasil

@anvil.server.http_endpoint("/delete-
training",methods=["POST"])
def delete_training(id,kategori,**q):
    hasil =
anvil.server.call('delete_training_data',kategori,id)
    return hasil

@anvil.server.http_endpoint("/create-
training",methods=["POST"])
def create_training(id,kategori,konten, **q):
```

```

    hasil =
anvil.server.call('create_training_data',id,kategori,kon
ten)
    return hasil

```

2. lda_proposal.ipynb

```

{
  "nbformat": 4,
  "nbformat_minor": 0,
  "metadata": {
    "colab": {
      "provenance": [],
      "collapsed_sections": []
    },
    "kernelspec": {
      "name": "python3",
      "display_name": "Python 3"
    },
    "language_info": {
      "name": "python"
    }
  },
  "cells": [
    {
      "cell_type": "code",
      "execution_count": 1,
      "metadata": {
        "colab": {
          "base_uri": "https://localhost:8080/",
          "height": 809
        },
        "id": "2WcbeLjcgolb",
        "outputId": "66e53b0e-0c8b-46a9-8a75-9a6bc46253a3"
      },
      "outputs": [
        {
          "output_type": "stream",
          "name": "stdout",
          "text": [
            "Looking in indexes:
https://pypi.org/simple, https://us-
python.pkg.dev/colab-wheels/public/simple/\n",
            "Collecting anvil-uplink\n",
            "  Downloading anvil_uplink-0.4.0-py2.py3-
none-any.whl (88 kB)\n",

```



```

        "metadata": {}
    },
    {
        "output_type": "stream",
        "name": "stderr",
        "text": [
            "[nlTK_data] Downloading package stopwords
to /root/nltk_data...\n",
            "[nlTK_data] Unzipping
corpora/stopwords.zip.\n",
            "[nlTK_data] Downloading package punkt to
/root/nltk_data...\n",
            "[nlTK_data] Unzipping
tokenizers/punkt.zip.\n"
        ]
    },
    {
        "output_type": "stream",
        "name": "stdout",
        "text": [
            "Connecting to wss://anvil.works/uplink\n",
            "Anvil websocket open\n",
            "Connected to \"Default environment\" as
SERVER\n"
        ]
    },
    {
        "output_type": "error",
        "ename": "KeyboardInterrupt",
        "evalue": "ignored",
        "traceback": [
            "\u001b[0;31m-----
-----\u001b[0m",
            "\u001b[0;31mKeyboardInterrupt\u001b[0m
Traceback (most recent call last)",
            "\u001b[0;32m<ipython-input-1-
ed2c14f61ff5>\u001b[0m in
\u001b[0;36m<module>\u001b[0;34m\u001b[0m\n\u001b[1;32m
125\u001b[0m     \u001b[0;32mreturn\u001b[0m
\u001b[0;34m\"Success\"\u001b[0m\u001b[0;34m\u001b[0m\u001b[0m\u001b[0m\u001b[0m
01b[0;34m\u001b[0m\u001b[0m\u001b[0m\n\u001b[1;32m
126\u001b[0m
\u001b[0;34m\u001b[0m\u001b[0m\u001b[0m\n\u001b[0;32m-->
127\u001b[0;31m
\u001b[0manvil\u001b[0m\u001b[0;34m.\u001b[0m\u001b[0m\u001b[0mse
rver\u001b[0m\u001b[0;34m.\u001b[0m\u001b[0m\u001b[0mwait_forever

```



```

        "        # deacc=True removes punctuations\n",
        "
yield(gensim.utils.simple_preprocess(str(sentence),
deacc=True))\n",
        "\n",
        "def remove_stopwords(texts):\n",
        "    return [[word for word in
simple_preprocess(str(doc)) \n",
        "                if word not in stop_words] for doc
in texts]\n",
        "\n",
        "class
NumpyFloatValuesEncoder(json.JSONEncoder):\n",
        "    def default(self, obj):\n",
        "        if isinstance(obj, np.float32):\n",
        "            return float(obj)\n",
obj)\n",
        "\n",
        "def proses(jumlah_topik):\n",
        "    fpath = 'drive/MyDrive/lda_training/'\n",
        "    data_training = []\n",
        "    list_kategori =
str(jumlah_topik).split(',')\n",
        "    dummy = ''\n",
        "    len_kategori = 0\n",
        "    for i in list_kategori:\n",
        "        data_kategori = []\n",
        "        files =
glob.glob(fpath+str(i)+'_*.txt')\n",
        "        for f in files:\n",
        "            try:\n",
        "                t = open(f,'r',encoding=\"utf8\")\n",
        "                contents = ''\n",
        "                contents = t.read()\n",
        "                t.close()\n",
        "                data_kategori.append(contents)\n",
        "                len_kategori = len_kategori + 1\n",
        "            except IOError as e:\n",
        "                dummy = ''\n",
        "            data_training.append(data_kategori)\n",
        "\n",
        "        data_words =
list(sent_to_words(data_training))\n",
        "        data_words = remove_stopwords(data_words)\n",
        "        \n",

```

```

        " id2word = corpora.Dictionary(data_words)\n",
        " texts = data_words\n",
        " corpus = [id2word.doc2bow(text) for text in
texts]\n",
        " num_topics = len_kategori\n",
        " lda_model =
gensim.models.LdaMulticore(corpus=corpus,\n",
        "
id2word=id2word,\n",
        "
num_topics=num_topics)\n",
        " lda_model.save(fpath+'lda_model.lda')\n",
        " lda_list = lda_model.print_topics();\n",
        " jsonString = json.dumps(lda_list)\n",
        " return jsonString;\n",
        "\n",
        "def cek_kategori(konten):\n",
        " fpath = 'drive/MyDrive/lda_training/'\n",
        " lda_model =
gensim.models.LdaMulticore.load(fpath+'lda_model.lda')\n",
        "
        " \n",
        " word_tokens = word_tokenize(konten)\n",
        " word_tokens_lower = [token.lower() for token
in word_tokens]\n",
        " word_tokens_no_stopwords = [w for w in
word_tokens_lower if not w in stop_words]\n",
        "\n",
        " bow =
lda_model.id2word.doc2bow(word_tokens_no_stopwords)\n",
        " topics_list =
lda_model.get_document_topics(bow)\n",
        " jsonString =
json.dumps(dict(topics_list),cls=NumpyFloatValuesEncoder
)\n",
        " return jsonString\n",
        "\n",

"anvil.server.connect(\"ESYSO4HQOSTFLT3A7FESKM52-
BFN7RKPZ55BZ3KP6\")\n",
        "@anvil.server.callable\n",
        "def proses_lda(jumlah_topik):\n",
        " return proses(jumlah_topik)\n",
        "\n",
        "@anvil.server.callable\n",
        "def cek_kategori_lda(konten):\n",

```

```

        " konten = urllib.parse.unquote(konten)\n",
        " konten =
urllib.parse.unquote_plus(konten)\n",
        " return cek_kategori(konten)\n",
        "\n",
        "@anvil.server.callable\n",
        "def clear_training_data():\n",
        " files =
glob.glob('drive/MyDrive/lda_training/*.txt')\n",
        " for f in files:\n",
        " try:\n",
        " os.remove(f)\n",
        " except OSError as e:\n",
        " return \"Error: %s : %s\" % (f,
e.strerror)\n",
        " return \"Success\"\n",
        "\n",
        "@anvil.server.callable\n",
        "def delete_training_data(kategori,id):\n",
        " f =
'drive/MyDrive/lda_training/'+kategori+'_'+id+'.txt'\n",
        " try:\n",
        " os.remove(f)\n",
        " except OSError as e:\n",
        " return \"Error: %s : %s\" % (f,
e.strerror)\n",
        " \n",
        " return \"Success\"\n",
        "\n",
        "@anvil.server.callable\n",
        "def create_training_data(id, kategori,
konten):\n",
        " try:\n",
        " f =
open('drive/MyDrive/lda_training/'+kategori+'_'+id+'.txt
', 'w')\n",
        " konten = urllib.parse.unquote(konten)\n",
        " konten =
urllib.parse.unquote_plus(konten)\n",
        " f.write(konten)\n",
        " except IOError as e:\n",
        " return \"Error: %s\" % (e.strerror)\n",
        " return \"Success\"\n",
        "\n",
        "anvil.server.wait_forever()"
]

```

```
}  
  ]  
}
```

LAMPIRAN
BIODATA PENELITI

A. Biodata Ketua

Nama Lengkap dan gelar : Rakhmat Kurniawan. R, ST., M.Kom
NIP : 19850316 201503 1 003
Pangkat/Jabatan : Penata /III.c/ Lektor
Jenis Kelamin : Laki-laki
Tempat dan tanggal lahir : Medan, 16 Maret 1985
Alamat : Jl. Karya Tani Gg. Melinjo No. 8
Medan Johor
No. Telephone/HP : 08566296036
Email : rakhmat.kr@uinsu.ac.id
Riwayat Pendidikan :
1. SD Nasional Khalsa Medan Tamat Tahun 1996
2. SMP Kemala Bhayangkari I Medan Tamat Tahun 1999
3. SMA Negeri 13 Medan Tamat Tahun 2002
4. S-1 STT Harapan Medan Tamat Tahun 2006
5. S-2 UPI “YPTK” Padang Tamat Tahun 2012

B. Anggota 1

1. Nama lengkap : Ilka Zufria, M.Kom
2. NIP : 19850604 201503 1 006
3. Pangkat/ Gol / Jabfung : Penata Tk.I / III.d / Lektor
4. Tempat/Tanggal Lahir : Medan, 04 Juni 1985
5. Jenis Kelamin : Laki-laki
6. Bidang Keahlian : Sistem Informasi
7. Kantor/Unit Kerja : UINSU Medan/ Fak Saintek
8. Alamat Kantor : Jln. Lapangan Golf No 120, Durin
Jangak, Kec. P.Batu, Deli Serdang

Kabupaten : Deli Serdang
Kode Pos : 20353
Telepon : (+6261) 6615683; 6622925
Faksimile : (+6261) 6615683
Email : ilkazufria@uinsu.ac.id
9. Alamat Rumah : Jln. Sederhana Gg. Raya 34, No. 04
Sambirejo Timur, Pasar 7 Tembung
Kec. Percut Sei Tuan
10. Kabupaten : Deli Serdang Kode Pos:20371
Telepon : -

Faksimile : -
No. HP. : +6281397238909

11. Pendidikan (S1 ke atas)

No	Perguruan Tinggi	Kota & Negara	Jenjang Pendidikan	Tahun Lulus	Bidang Studi
1.	UPI "YPTK" Padang	Padang, Indonesia	S2	2009	Ilmu Komputer (Sistem Informasi)
2.	UPI "YPTK" Padang	Padang, Indonesia	S1	2006	Sistem Informasi

C. Biodata Anggota 2

Nama Lengkap : Rizky Pratama Putra
NIM : 0701172060
Status : Mahasiswa Semester IX
Program Studi : Ilmu Komputer
Jenis Kelamin : Laki-laki
Tempat dan tanggal lahir : Medan, 19 Januari 1999
Alamat : Jalan Gatot Subroto Gg Rasmi No. 34
Medan
No. Telephone/HP : 0853 7271 1573