

PERANCANGAN SISTEM PENDUKUNG KEPUTUSAN TOPIK TUGAS
AKHIR PADA PROGRAM STUDI TEKNIK KOMPUTER DAN JARINGAN
POLITEKNIK NEGERI UJUNG PANDANG



SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar
Sarjana Terapan Program Studi Teknik Komputer dan Jaringan
Jurusan Teknik Informatika dan Komputer
Politeknik Negeri Ujung Pandang

MARYAM AZZAHRA
425 20 011

PROGRAM STUDI S1 TERAPAN TEKNIK KOMPUTER DAN JARINGAN
JURUSAN TEKNIK INFORMATIKA DAN KOMPUTER
POLITEKNIK NEGERI UJUNG PANDANG
MAKASSAR
2024

HALAMAN PENGESAHAN

Skripsi dengan judul “PERANCANGAN SISTEM PENDUKUNG KEPUTUSAN TOPIK TUGAS AKHIR PADA PROGRAM STUDI TEKNIK KOMPUTER DAN JARINGAN POLITEKNIK NEGERI UJUNG PANDANG” oleh MARYAM AZZAHRA dengan Nomor Induk Mahasiswa 425 20 011 telah diterima dan disahkan sebagai salah satu syarat untuk memperoleh gelar Sarjana Terapan pada Program Studi Teknik Komputer dan Jaringan Jurusan Teknik Informatika dan Komputer Politeknik Negeri Ujung Pandang.

Makassar, 14 Agustus 2024

Mengesahkan,

Dosen Pembimbing I



Meylanie Olivya, S.T., M.T.
NIP. 198205032014042002

Dosen Pembimbing II



Fadli Tamrin, S.Kom., M.Cs.
NIP. 198912232022031006

Mengetahui,

Ketua Program Studi
Teknik Komputer dan Jaringan
Politeknik Negeri Ujung Pandang



Meylanie Olivya, S.T., M.T.
NIP. 198205032014042002

HALAMAN PENERIMAAN

Pada hari ini, Kamis tanggal 28 Agustus 2024, Tim Penguji Ujian Sidang Skripsi telah menerima dengan baik skripsi mahasiswa: **MARYAM AZZAHRA** Nomor Induk Mahasiswa **425 20 011** dengan judul “**PERANCANGAN SISTEM PENDUKUNG KEPUTUSAN TOPIK TUGAS AKHIR PADA PROGRAM STUDI TEKNIK KOMPUTER DAN JARINGAN POLITEKNIK NEGERI UJUNG PANDANG**”

Makassar, 28 Agustus 2024

Tim Penguji Ujian Sidang Skripsi:

- | | | |
|--|---------------|---------|
| 1. Irmawati, S.T., M.T. | Ketua | (.....) |
| 2. Rini Nur, S.T., M.T. | Sekretaris | (.....) |
| 3. Muhammad Nur Yasir Utomo, S.ST., M.Eng. | Anggota | (.....) |
| 4. Muh. Fajri Raharjo, S.T., M.T. | Anggota | (.....) |
| 5. Meylanie Olivya, S.T., M.T. | Pembimbing I | (.....) |
| 6. Fadli Tamrin, S.Kom., M.Cs. | Pembimbing II | (.....) |

KATA PENGANTAR

Alhamdulillah, segala puji dan syukur atas nikmat dan karunia yang tak terhingga dari Allah Subhanahu Wa Ta'ala, sehingga penulis dapat menyelesaikan skripsi ini dengan baik. Shalawat dan salam terhatur kepada Rasulullah Sallallahu Alaihi Wasallam, panutan terbaik bagi seluruh umat manusia.

Sebagai salah satu syarat untuk menyelesaikan studi dan memperoleh gelar (S1 Terapan) di Program Studi Teknik Komputer dan Jaringan, Jurusan Teknik Informatika dan Komputer, Politeknik Negeri Ujung Pandang, skripsi ini disusun dengan sebaik-baiknya. Penulis menyadari bahwa keberhasilan skripsi ini tidak terlepas dari bantuan berbagai pihak, baik secara langsung maupun tidak langsung. Oleh karena itu, penulis ingin menyampaikan apresiasi dan terima kasih yang sebesar-besarnya kepada:


1. Kedua orang yang paling berjasa dalam hidup penulis, yaitu kedua orang tua penulis, Bapak Pakabbir dan Ibu Nurhayati. Terima kasih atas kepercayaan yang telah diberikan dengan mengizinkan penulis merantau, serta atas pengorbanan, cinta, doa, motivasi, semangat, dan nasihat yang tiada hentinya. Semoga Allah SWT selalu menjaga kalian dalam kebaikan dan kemudahan, serta semoga kalian selalu sehat agar bisa melihat anak-anak mu sukses di kemudian hari, Aamiin.
2. Saudara saudari penulis yaitu Qois Al Faruqi, Aisyah, Khadijah, Ayyub, dan Syufiah yang selalu menjadi motivator dan memotivasi penulis untuk terus belajar supaya menjadi adik dan kakak yang baik dalam menyelesaikan studi.
3. Bapak Prof. Ir. Ilyas Mansur, M.T selaku Direktur Politeknik Negeri Ujung Pandang.
4. Ibu Iin Karmila Yusri, S.ST., M.Eng., Ph.D. selaku Ketua Jurusan Teknik Informatika dan komputer Politeknik Negeri Ujung Pandang.
5. Ibu Meylanie Olivya, S.T., M.T. selaku Koordinator Program Studi Teknik Komputer dan Jaringan.
6. Ibu Meylanie Olivya, S.T., M.T. selaku Pembimbing I dan Fadli Tamrin, S.Kom., M.Cs. selaku Pembimbing II atas segala ilmu, motivasi, nasehat,

arahan, pandangan, bantuan dan kesedian waktu dan kesabarannya dalam membimbing penulis hingga terselesaikannya penelitian ini.

7. Seluruh dosen dan Staf Jurusan Teknik Infomatika dan Komputer, khususnya Program Studi S1 Terapan Teknik Komputer dan Jaringan.
8. Teman-teman seperjuangan di Program Studi Teknik Komputer dan Jaringan Angkatan 2020 yang telah berjuang bersama selama 4 tahun, mengajarkan berbagai banyak hal baik dari segi akademik maupun non akademik.
9. Semua pihak yang telah memberikan bantuan moril maupun materil yang tidak dapat disebutkan satu per satu.
10. Dan yang terakhir untuk diri saya sendiri yang telah berjuang sejauh ini meskipun tak mudah untuk sampai ketitik ini pasti banyak rintangan tapi Alhamdulillah dengan izin Allah saya bisa melewatinya.

Penulis menyadari bahwa skripsi ini masih jauh dari kesempurnaan, sehingga penulis mengharap kritik dan saran yang membangun demi perbaikan dimasa mendatang.

Makassar, 26 Agustus 2024



Penulis

DAFTAR ISI

HALAMAN PENGESAHAN	ii
HALAMAN PENERIMAAN.....	iii
KATA PENGANTAR	iv
DAFTAR ISI.....	vi
DAFTAR TABEL.....	viii
DAFTAR GAMBAR	ix
DAFTAR LAMPIRAN.....	xi
SURAT PERNYATAAN	xii
RINGKASAN.....	1
BAB I PENDAHULUAN	2
1.1 Latar Belakang	2
1.2 Rumusan Masalah	5
1.3 Ruang Lingkup Penelitian	6
1.4 Tujuan Penelitian.....	6
1.5 Manfaat Penelitian.....	6
BAB II TINJAUAN PUSTAKA	8
2.1 Tugas Akhir (TA).....	8
2.1.1 Profil Lulusan Program Studi TKJ PNUP Kurikulum 2021	8
2.1.2 Data kelulusan alumni TKJ PNUP	9
2.2 Sistem Rekomendasi	10
2.3 <i>Machine Learning</i>	10
2.3.1 <i>Random Forest</i>	10
2.4 <i>Flask</i>	13
2.5 Penelitian Terkait.....	14
2.6 Pengujian	18
2.6.1 <i>Black Box Testing</i>	18
2.6.2 Pengujian akurasi <i>Random Forest</i>	19
BAB III METODOLOGI PENELITIAN.....	20
3.1 Tempat dan Waktu Penelitian	20

3.2	Prosedur Penelitian	20
3.2.1	Identifikasi Masalah	20
3.2.2	Analisis Kebutuhan	21
3.2.3	Pengumpulan Data	22
3.3	Perancangan Sistem.....	23
3.3.1	Penerapan <i>Random Forest</i>	24
3.3.2	Desain Implementasi <i>Website</i>	29
3.4	Pengujian	33
BAB IV HASIL DAN PEMBAHASAN.....		35
4.1	Pengumpulan Data	35
4.2	Perancangan Sistem.....	36
4.2.1	Praprocessing Data	36
4.2.2	Pemrograman untuk menentukan label	36
4.2.3	Pemrograman <i>Random Forest</i>	38
4.2.4	Implementasi dan <i>Deployment Model Random Forest</i>	41
4.2.5	Implementasi <i>User Interface (UI)</i>	42
4.3	Pengujian Sistem	48
4.3.1	Hasil Pengujian Model <i>Random Forest</i>	48
4.3.4	Validasi Efektivitas Sistem Pendukung Keputusan Topik Tugas Akhir	58
BAB V PENUTUP.....		61
5.1	Kesimpulan.....	61
5.2	Saran.....	61
DAFTAR PUSTAKA.....		63
LAMPIRAN.....		66

DAFTAR TABEL

Tabel 2. 1 Perbandingan dengan penelitian terdahulu	16
Tabel 3. 1 Kebutuhan Perangkat Keras	22
Tabel 3. 2 Kebutuhan Perangkat Lunak	22
Tabel 4. 1 Kelompok Mata Kuliah	36
Tabel 4. 2 Konversi Grade ke Nilai Numerik	39
Tabel 4. 4 Pengujian halaman <i>login</i>	50
Tabel 4. 5 Pengujian halaman utama.....	50
Tabel 4. 6 Pengujian halaman klasifikasi	50
Tabel 4. 7 Pengujian halaman <i>history</i>	51
Tabel 4. 8 Pengujian fungsi <i>logout</i>	52
Tabel 4. 9 Tabulasi data kuesioner	52
Tabel 4. 10 Jumlah aspek data manfaat.....	54
Tabel 4. 11 Aspek data kemudahan	54
Tabel 4. 12 Aspek data kepuasan user.....	55
Tabel 4. 13 Skor aspek manfaat	55
Tabel 4. 14 Skor aspek kemudahan.....	56
Tabel 4. 15 Kepuasan user	56
Tabel 4. 16 Pengujian kuesioner manfaat	57
Tabel 4. 17 Pengujian kuesioner kemudahan	57
Tabel 4. 18 Pengujian kuesioner kepuasan user.....	58
Tabel 4. 19 Validasi Efektivitas.....	59

DAFTAR GAMBAR

Gambar 2. 1 Data Kelulusan	9
Gambar 2. 2 Ilustrasi <i>Random Forest</i>	11
Gambar 2. 3 <i>Framework Flask</i>	14
Gambar 2. 4 Black Box <i>testing</i>	19
Gambar 3. 1 Prosedur Penelitian.....	20
Gambar 3. 2 Arsitektur sistem.....	23
Gambar 3. 3 Flowchart penerapan <i>Random Forest</i>	24
Gambar 3. 4 Kategori Mata Kuliah.....	27
Gambar 3. 5 Ilustrasi Pengelompokkan Mahasiswa Model.....	28
Gambar 3. 6 Ilustrasi Proses Klasifikasi <i>Random Forest</i> dalam setiap <i>Desicion Tree</i>	28
Gambar 3. 7 <i>Flowchart</i> sistem <i>website</i>	29
Gambar 3. 8 <i>Use Case Diagram</i>	30
Gambar 3. 9 Halaman Utama.....	30
Gambar 3. 10 <i>Form</i> tampilan nilai dan input minat.....	31
Gambar 3. 11 Halaman Hasil Rekomendasi	32
Gambar 4. 1 Data nilai akademik.....	35
Gambar 4. 2 File data awal dan hasil pengelompokkan.....	37
Gambar 4. 3 Hasil Pengelompokkan.....	38
Gambar 4. 4 Dataset yang akan digunakan.....	39
Gambar 4. 5 Dataset dan model hasil <i>Random Forest</i>	41
Gambar 4. 6 <i>Deployment</i> model.....	41
Gambar 4. 7 Halaman login	42
Gambar 4. 8 Halaman <i>dashboard</i>	43
Gambar 4. 9 Halaman tampilan nilai	44
Gambar 4. 10 Form input minat.....	44
Gambar 4. 11 Login akun Arico.....	45
Gambar 4. 12 Form nilai dan minat serta hasil rekomendasi.....	45
Gambar 4. 13 Login akun Fitrah	46

Gambar 4. 14 Form nilai dan minat serta hasil rekomendasi.....	46
Gambar 4. 15 Login Akun Jihan	47
Gambar 4. 16 Form nilai dan minat serta hasil rekomendasi.....	47
Gambar 4. 17 Halaman <i>history</i>	48

DAFTAR LAMPIRAN

Lampiran 1 Code Data Dummy	67
Lampiran 2 <i>Code</i> Pengelompokkan untuk mendapatkan label.....	67
Lampiran 3 Visualisasi Hasil Pengelompokkan.....	69
Lampiran 4 <i>Code</i> Klasifikasi Metode <i>Random Forest</i>	70
Lampiran 5 <i>Learning Curve</i>	72
Lampiran 6 Hasil Responden	72

SURAT PERNYATAAN

Saya yang bertanda tangan dibawah ini :

Nama : Maryam Azzahra

NIM : 425 20 011

Menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam skripsi ini yang berjudul **“PERANCANGAN SISTEM PENDUKUNG KEPUTUSAN TOPIK TUGAS AKHIR PADA PROGRAM STUDI TEKNIK KOMPUTER DAN JARINGAN POLITEKNIK NEGERI UJUNG PANDANG”** merupakan gagasan dan hasil karya saya sendiri dengan arahan komisi pembimbing, dan belum pernah diajukan dalam bentuk apapun pada perguruan tinggi dan instansi manapun.

Semua data dan informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan dari penulis lain telah disebutkan dalam naskah dan dicantumkan dalam skripsi ini.

Jika pernyataan saya tersebut diatas tidak benar, saya siap menanggung risiko yang ditetapkan oleh Politeknik Negeri Ujung Pandang.

Makassar, 28 Agustus 2024

Penulis

Maryam Azzahra
NIM. 425 20 011

PERANCANGAN SISTEM PENDUKUNG KEPUTUSAN TOPIK TUGAS AKHIR PADA PROGRAM STUDI TEKNIK KOMPUTER DAN JARINGAN POLITEKNIK NEGERI UJUNG PANDANG

RINGKASAN

Penelitian ini dilatarbelakangi oleh kesulitan yang sering dialami mahasiswa dalam menentukan topik tugas akhir yang sesuai dengan minat dan kemampuannya, yang mengakibatkan keterlambatan kelulusan. Tujuan dari penelitian ini adalah untuk membuat model sistem pendukung keputusan topik tugas akhir yang akurat menggunakan Algoritma *Random Forest*, merancang sistem pendukung keputusan berbasis *website* untuk membantu mahasiswa dalam menentukan topik tugas akhir, dan mengevaluasi efisiensi serta kualitas sistem tersebut. Data yang digunakan adalah data nilai akademik dan minat mahasiswa program studi Teknik Komputer dan Jaringan di Politeknik Negeri Ujung Pandang.

Hasil penelitian menunjukkan bahwa Model *Random Forest* mampu menghasilkan rekomendasi topik tugas akhir dengan akurasi *Cross-validation* 61,36%. Sistem pendukung keputusan yang dirancang berbasis *website* dapat membantu mahasiswa dalam menentukan topik tugas akhir sesuai dengan minat dan kemampuan akademiknya, serta meningkatkan efisiensi dan kualitas proses pemilihan topik tugas akhir.

Penelitian ini memberikan kontribusi dalam mempermudah mahasiswa menemukan topik tugas akhir yang sesuai, serta memberikan panduan bagi pengembangan sistem rekomendasi serupa di masa depan.

Kata Kunci: Topik Tugas Akhir, *Random Forest*, Sistem Pendukung Keputusan.

BAB I PENDAHULUAN

1.1 Latar Belakang

Tugas akhir (TA) atau skripsi merupakan sebuah tulisan akademis yang harus disusun oleh mahasiswa sebagai bagian dari syarat kelulusan dalam pendidikan tingkat akademis. Tugas akhir tersebut mencerminkan kemampuan akademik mahasiswa dalam melakukan penelitian yang terkait dengan masalah pendidikan sesuai dengan bidang studinya. Proses penyusunan tugas akhir ini diperlukan untuk meraih gelar Sarjana Strata Satu. Selain itu, penelitian juga merupakan bentuk pembelajaran yang efektif di luar lingkungan akademis, di mana mahasiswa memiliki potensi untuk memberikan kontribusi yang bermanfaat bagi masyarakat umum atau organisasi tertentu (Maryadi, 2019).

Bagi mahasiswa, melakukan penelitian merupakan aktivitas yang umum terjadi dalam lingkup perkuliahan di kampus. Namun, banyak mahasiswa yang mengalami kesulitan dalam menemukan topik Tugas Akhir yang cocok dengan minat dan keahliannya (Maryadi, 2019). Banyak faktor baik internal maupun eksternal yang sering menjadi hambatan bagi mahasiswa dalam proses penyusunan skripsi, terutama ketika topik yang dipilih tidak sesuai dengan kemampuannya (Fahmi et al., 2021). Situasi serupa juga terjadi pada mahasiswa di Program Studi Teknik Komputer dan Jaringan Politeknik Negeri Ujung Pandang, mahasiswa sering kali tidak mempertimbangkan secara menyeluruh faktor-faktor pendukung dan penghambat ketika memilih topik skripsi yang akan diteliti. Sebagian besar mahasiswa memilih topik TA berdasarkan referensi dari teman seangkatan atau internet, yang tidak selalu sesuai dengan kemampuannya, hal ini dapat menghambat proses penyelesaian TA (Haviluddin et al., 2021).

Program studi Teknik Komputer dan Jaringan di Politeknik Negeri Ujung Pandang memiliki 4 empat profile lulusan. Berdasarkan hasil rekapan lulusan program studi Teknik Komputer dan Jaringan mulai tahun 2009 hingga 2019 terdapat sekiranya kurang lebih 15% mahasiswa pada program studi ini mengalami keterlambatan waktu kelulusan yang seharusnya. Juga berdasarkan hasil kuesioner

dan wawancara terhadap 14 alumni dan 30 mahasiswa akhir prodi TKJ, sebesar 47.7% mengungkapkan bahwa faktor penghambat yang termasuk cukup banyak dialami mahasiswa dikarenakan kesulitan menentukan topik tugas akhir yang mengakibatkan kesulitan menguasai teori maupun praktek dalam pengerjaan skripsi karena tidak sesuai dengan kemampuannya. Sebanyak 34.1% mengungkapkan kesulitan dalam pengumpulan data atau pengerjaan sistem. 9.1% kesulitan karena penelitian kurang dikuasai atau dipahami. Dan 9.1% kesulitan dalam komunikasi dengan Pembimbing

Pentingnya memilih topik tugas akhir yang tidak hanya sesuai dengan minat, tetapi juga kemampuan mahasiswa perlu ditekankan. Data akademik mahasiswa dari semester satu hingga tujuh dapat menjadi indikator kemampuan. Hampir semua mata kuliah memiliki relevansi dengan profile lulusan, yang dapat sekaligus menentukan topik tugas akhir. Analisis data akademik dapat membantu menentukan topik skripsi yang sesuai. Dengan demikian, mahasiswa dapat efektif dalam menyelesaikan skripsi tepat waktu, sekaligus memberikan manfaat bagi program studi khususnya dalam konteks akreditasi (Muttaqin & Defriani, 2020). Maka diperlukan adanya sistem pendukung keputusan klasifikasi yang dapat merekomendasi mahasiswa dalam memilih topik tugas akhir yang sesuai dengan kemampuan dan pemahamannya terhadap mata kuliah, yang berdasarkan pada rekam jejak akademiknya (Salam et al., 2022).

Ada banyak metode yang dapat digunakan dalam melakukan klasifikasi prediksi, seperti pada penelitian yang dilakukan oleh (Darmawan et al., 2023) memprediksi kelulusan siswa Madrasah Aliyah Swasta (MAS) menggunakan Algoritma *Support Vector Machine* (SVM) dan *Random Forest* (RF), serta membandingkan performa kedua algoritma tersebut berdasarkan data nilai dan sikap. Hasil penelitian menunjukkan bahwa SVM dan RF sangat baik dalam memprediksi kelulusan siswa MAS dengan akurasi yang sangat tinggi, yaitu SVM (98,98%) dan RF (99,49%). Random Forest terbukti sedikit lebih unggul dibandingkan SVM dengan hasil parameter pengukuran Akurasi (99,49% vs 98,98%), *Precision* (99,74% vs 99,23%), *Recall* (99,74% untuk keduanya), dan *F-Measure* (99,74% vs 99,48%). Juga penelitian oleh (Zeniarta et al., 2022)

mengusulkan model klasifikasi terbaik dengan membandingkan tingkat akurasi dari beberapa algoritma klasifikasi, termasuk *Naïve Bayes*, *Random Forest*, *Decision Tree*, *K-Nearest Neighbor (K-NN)*, dan *Support Vector Machine (SVM)*, untuk memprediksi kelulusan mahasiswa. Hasilnya menunjukkan bahwa Algoritma *Random Forest* memiliki akurasi tertinggi, mencapai 77,35%, yang lebih baik dibandingkan dengan algoritma lainnya. Dan (Mu'tashim & Zaidiah, 2023) melakukan penelitian untuk menganalisis ketepatan waktu kelulusan mahasiswa menggunakan Metode *Random Forest* dan *Gradient Boosting* untuk menentukan model klasifikasi terbaik. Pada sampel 590 data, *Random Forest* dengan *10-fold Cross-validation* mencapai akurasi 82,64%, sedangkan *Gradient Boosting* dengan *3-fold Cross-validation* mencapai akurasi 79,66%.

Berdasarkan paparan beberapa hasil penelitian di atas, Algoritma *Random Forest* dapat menjadi pilihan yang unggul untuk kasus serupa. Oleh karena itu, metode ini digunakan dalam penelitian ini untuk melakukan klasifikasi terhadap profil lulusan sehingga dapat tergambar topik tugas akhir mahasiswa. *Random Forest* memiliki sejumlah keunggulan, seperti kemampuan untuk mengelola data yang kompleks dan besar, toleransi terhadap *overfitting*, kemampuan untuk menangani variabel yang tidak penting, serta memberikan perkiraan keakuratan model yang baik (Agnes et al., 2023). Data yang digunakan adalah data nilai akademik dan minat mahasiswa program studi Teknik Komputer dan Jaringan Politeknik Negeri Ujung Pandang.

Adapun beberapa penelitian mengenai topik tugas akhir dilakukan (Triawan & Lintang Melinda, 2020) yang membuat aplikasi sistem pendukung keputusan topik tugas akhir untuk mahasiswa di perguruan tinggi dengan Metode *Naïve Bayes*. Aplikasi tersebut dinilai layak digunakan dengan persentase kelayakan 79%, dan memiliki akurasi komparasi sebesar 60% dibandingkan dengan alat data mining yang ada. Penelitian lainnya dilakukan oleh (Haviluddin et al., 2021), yang mengimplementasikan metode *K-Means* untuk mengelompokkan rekomendasi Tugas Akhir (TA) mahasiswa. Eksperimen menunjukkan bahwa *cluster* TA dilakukan ke dalam 3 *cluster* (C), dengan hasil pengujian akurasi *cluster* menggunakan metode *Sum of Squared Errors (SSE)* 0.6566 dan *Silhouette*

Coefficient (SC) 5.8329, menunjukkan bahwa nilai Mata Kuliah Wajib (MKW) memengaruhi penentuan TA.

Penelitian lain dengan kasus serupa dan menggunakan Algoritma *Random Forest* dilakukan oleh (Aini et al., 2024) Dalam penelitiannya yang mengimplementasikan Algoritma *Random Forest* untuk memetakan bidang MSIB di Prodi Pendidikan Informatika. Atribut yang digunakan adalah nilai mata kuliah mahasiswa. Hasil menunjukkan tingkat akurasi 80%, *precision* 80%, dan *recall* 82%, sehingga model klasifikasi menggunakan *Random Forest* termasuk kategori baik. Penelitian ini berfokus pada implementasi Algoritma *Random Forest* dalam konteks sistem rekomendasi topik tugas akhir mahasiswa, menguji potensinya untuk memberikan rekomendasi yang tepat dan relevan berdasarkan minat dan kemampuan akademik mahasiswa. Penelitian ini bertujuan untuk mengisi celah pengetahuan dalam penggunaan Algoritma *Random Forest* di domain pendidikan tinggi, khususnya dalam penentuan topik tugas akhir, serta memberikan pemahaman yang lebih baik tentang kinerjanya dalam konteks tersebut.

Berdasarkan pemaparan masalah diatas maka dibuat penelitian **“Perancangan Sistem Pendukung Keputusan Topik Tugas Akhir pada Program Studi Teknik Komputer dan Jaringan Politeknik Negeri Ujung Pandang”** yang diharapkan sistem ini akan memberikan bantuan kepada mahasiswa yang akan menyelesaikan Tugas Akhir dalam menemukan topik yang sesuai dengan minat dan keahliannya, sehingga dapat mengurangi jumlah mahasiswa yang mengalami keterlambatan dalam lulus karena kesulitan menemukan topik yang sesuai dengan minat dan keahliannya.

1.2 Rumusan Masalah

Berdasarkan pada latar belakang masalah diatas maka diajukan rumusan masalah sebagai berikut :

1. Bagaimana mengoptimalkan Algoritma *Random Forest* untuk menghasilkan model rekomendasi topik tugas akhir yang akurat?

2. Bagaimana merancang sistem pendukung keputusan berbasis *website* untuk membantu mahasiswa menentukan topik tugas akhir sesuai minat dan kemampuannya?
3. Bagaimana mengevaluasi efisiensi dan kualitas sistem pendukung keputusan dalam pemilihan topik tugas akhir mahasiswa?

1.3 Ruang Lingkup Penelitian

Ruang lingkup penelitian ini memiliki batasan-batasan sebagai berikut :

1. Penelitian ini membahas mengenai proses sistem pendukung keputusan topik tugas akhir mahasiswa pada data akademik
2. Penerapan Algoritma *Random Forest* pada sistem pendukung keputusan topik tugas akhir sebagai metode penyelesaian masalah
3. Pengujian tingkat akurasi dari Algoritma *Random Forest*
4. Data yang digunakan adalah data nilai akademik dan minat mahasiswa program studi Teknik Komputer dan Jaringan Politeknik Negeri Ujung Pandang

1.4 Tujuan Penelitian

Berdasarkan pada latar belakang dan rumusan masalah yang telah diuraikan diatas, maka tujuan penelitian ini adalah :

1. Menganalisis Algoritma *Random Forest* untuk menghasilkan model yang akurat.
2. Membuat sistem pendukung keputusan berbasis *website* yang dapat membantu mahasiswa dalam menentukan topik tugas akhir yang sesuai dengan minat dan kemampuan akademiknya
3. Mengevaluasi kinerja sistem pendukung keputusan dalam meningkatkan efisiensi dan kualitas proses pemilihan topik tugas akhir mahasiswa.

1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini yaitu :

1. Pengembang/peneliti dapat memperoleh pemahaman Algoritma *Random Forest* dan meningkatkan keahlian dan kontribusi di bidang ilmu komputer.

2. Memudahkan mahasiswa/pengguna dalam menemukan topik tugas akhir sesuai minat dan kemampuan, serta meningkatkan efisiensi dan kualitas tugas akhir.
3. Perguruan tinggi atau lembaga pendidikan dapat meningkatkan kualitas pendidikan dan kepuasan mahasiswa.

BAB II TINJAUAN PUSTAKA

2.1 Tugas Akhir (TA)

Menurut (Ramadani, 2021) Tugas Akhir yang juga dikenal sebagai skripsi, adalah karya tulis ilmiah yang bertujuan untuk memecahkan masalah dalam bidang tertentu dengan mematuhi aturan-aturan yang berlaku dalam ilmu tersebut. Ini merupakan bagian dari metode ilmiah yang dilakukan untuk mengembangkan dasar-dasar pengetahuan ilmiah guna praktik yang efektif dan efisien, dengan peneliti bertanggung jawab dalam meningkatkan kualitas layanan dan menciptakan cara untuk meningkatkan kualitas hidup masyarakat. Penelitian skripsi atau tugas akhir pada umumnya bertujuan untuk mengembangkan ilmu dan pengetahuan yang sudah ada, serta menemukan fakta baru atau temuan yang dapat digunakan untuk menyusun teori, konsep, hukum, atau metodologi baru yang dapat diterapkan dalam pemecahan masalah yang ada.

Penelitian memiliki tujuan khusus yang meliputi pembuktian teori-teori yang sudah ada, penemuan teori atau produk baru untuk memenuhi kebutuhan zaman dan perkembangan teknologi, serta pengembangan hasil penelitian sebelumnya untuk memperkaya ilmu pengetahuan dan teknologi. Dengan demikian, penelitian skripsi atau tugas akhir tidak hanya bertujuan untuk mengonfirmasi kebenaran teori-teori yang sudah ada, tetapi juga untuk menciptakan inovasi baru yang dapat meningkatkan kualitas hidup manusia melalui pengembangan ilmu pengetahuan dan teknologi (Ramadani, 2021).

2.1.1 Profil Lulusan Program Studi TKJ PNUP Kurikulum 2021

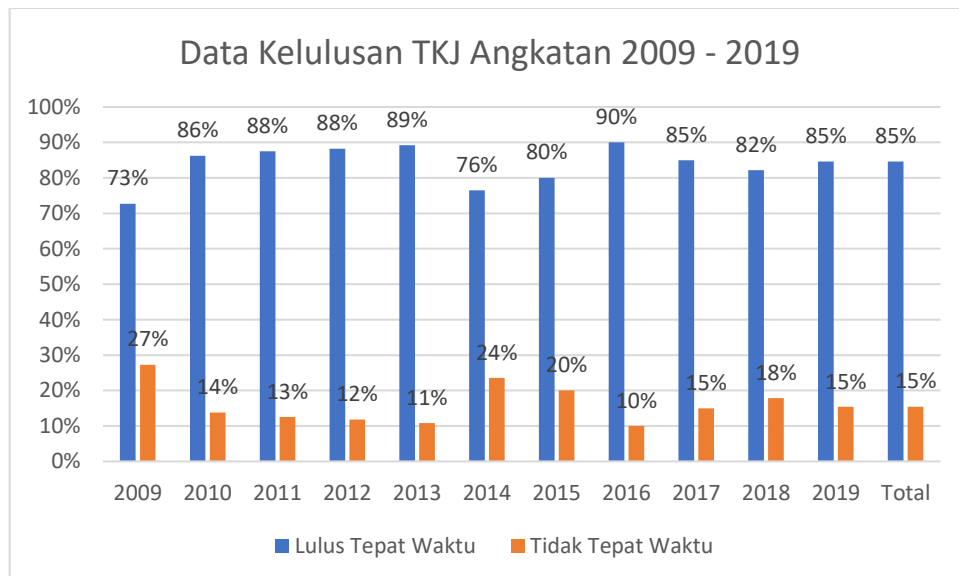
Profil lulusan program studi Teknik Komputer dan Jaringan (TKJ) mencakup individu yang memiliki pengetahuan dan keterampilan dalam berbagai aspek teknologi komputer dan jaringan. Mahasiswa terlatih dalam merancang, mengembangkan, mengelola, dan memelihara infrastruktur jaringan komputer, sistem komunikasi data, dan aplikasi perangkat lunak. Lulusan TKJ juga memiliki kemampuan dalam mengelola keamanan informasi, mengintegrasikan perangkat keras dan perangkat lunak, serta menganalisis dan menyelesaikan masalah teknis

dalam lingkungan teknologi informasi. Dengan demikian, mahasiswa siap untuk mengejar berbagai karier di industri teknologi, mulai dari administrator jaringan hingga pengembang perangkat lunak, serta peran lainnya dalam pengelolaan teknologi informasi. Berikut adalah empat profile lulusan TKJ PNUP Kurikulum 2021 :

- 1) *Network Administrator*
- 2) *Software Engineer*
- 3) *Associate Data Engineer*
- 4) *IoT Engineer*

2.1.2 Data kelulusan alumni TKJ PNUP

Pada Gambar 2.1 memaparkan data lulusan alumni menunjukkan data mahasiswa yang lulus tepat waktu dan yang mengalami keterlambatan kelulusan di Program Studi Teknik Komputer dan Jaringan Politeknik Negeri Ujung Pandang angkatan 2009 hingga 2019. Warna biru menunjukkan banyaknya mahasiswa yang lulus tepat waktu, dan warna *orange* menunjukkan data mahasiswa yang telat lulus atau tidak tepat waktu.



Gambar 2. 1 Data Kelulusan

2.2 Sistem Rekomendasi

Sistem rekomendasi adalah sebuah sistem dan metodologi yang memberikan rekomendasi terhadap produk atau item yang mungkin diminati atau disukai oleh pengguna. Tujuan utama sistem rekomendasi adalah membantu pengguna dalam membuat keputusan dengan memberikan saran yang relevan, seperti daftar lagu yang diprediksi akan disukai oleh pengguna (Theodorus et al., 2021). Penerapan sistem rekomendasi tidak hanya terbatas pada industri hiburan, tetapi juga ditemukan dalam konteks pendidikan. Sistem rekomendasi juga sering disebut sebagai Sistem Pendukung Keputusan (SPK) dalam literatur (Pratama et al., 2021).

2.3 *Machine Learning*

Menurut (Theodorus et al., 2021) *Machine Learning* merupakan suatu teknik yang menggunakan pendekatan matematis untuk melakukan inferensi terhadap data dengan tujuan membangun model yang merepresentasikan pola-pola dalam data tersebut. Teknik ini sering mengandalkan konsep statistika dan aljabar *linear*. Dataset dalam *Machine Learning* terbagi menjadi dua bagian utama, yaitu dataset *training* yang digunakan untuk melatih model dan dataset *testing* yang digunakan untuk menguji kinerja model yang telah dibangun. *Machine Learning* memiliki berbagai kegunaan, antara lain:

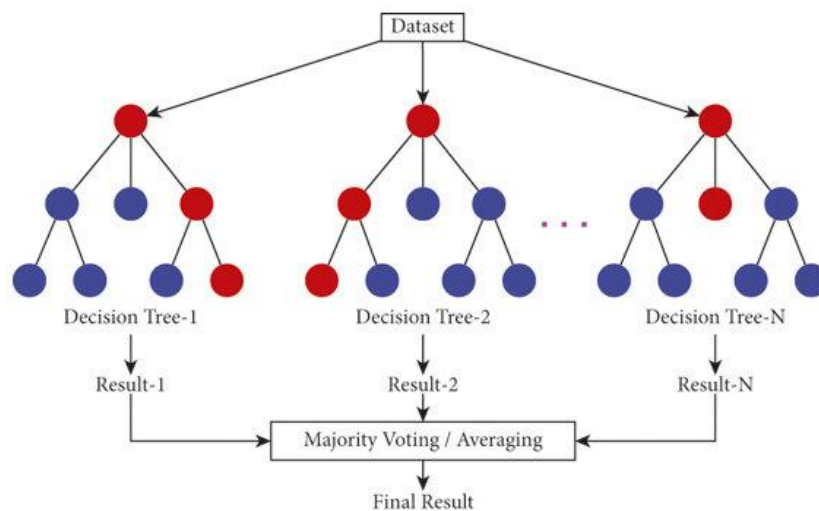
- 1) *Classification* yang digunakan untuk memprediksi nilai atau kelas dari individu dalam suatu populasi
- 2) *Similarity Matching* yang mengidentifikasi kemiripan antara individu berdasarkan data yang ada.
- 3) *Clustering* yang membagi data ke dalam kelompok berdasarkan kriteria tertentu.

2.3.1 *Random Forest*

Random Forest merupakan sebuah teknik klasifikasi yang merupakan pengembangan dari *Decision Tree*. *Decision Tree* sendiri adalah sebuah metode non-parametrik yang dapat diterapkan baik untuk regresi maupun klasifikasi. Pada

dasarnya, metode ini memanfaatkan seluruh data dan atribut yang tersedia untuk membuat satu pohon keputusan, sedangkan *Random Forest* akan membentuk lebih dari satu pohon keputusan, seperti yang diilustrasikan dalam Gambar 2.1. Setiap pohon dalam Metode *Random Forest* tidak akan menggunakan seluruh atribut, melainkan secara acak memilih beberapa atribut untuk digunakan dalam pembentukan pohon keputusan. Untuk melakukan prediksi, Metode *Random Forest* menggunakan pengambilan suara (*voting*) yang melibatkan seluruh pohon keputusan yang ada (Pratama et al., 2021).

Ada lima tahap utama yang diperlukan untuk menerapkan Model *Random Forest*. Tahap pertama adalah menentukan jumlah *Decision Tree* yang akan dibuat, yang biasanya disebut sebagai n . Langkah berikutnya adalah membuat sampel acak dari data menggunakan teknik *bootstrap* dan kemudian membuat satu *Decision Tree* untuk setiap sampel tersebut. Langkah selanjutnya adalah memvalidasi jumlah *Decision Tree* yang dibuat dengan jumlah n yang telah ditentukan sebelumnya. Setelah itu, hasil prediksi dari setiap *Decision Tree* digabungkan untuk menghasilkan prediksi akhir (Pahlevi et al., 2023).



Gambar 2. 2 Ilustrasi *Random Forest*
Sumber: researchgate.net

Berikut adalah rumus-rumus yang berjalan pada Algoritma Random Forest:

Persamaan 2.1 hingga 2.8 sudah terintegrasi dalam *library sklearn*.

1) *Bootstrap Sampling*

Bootstrap Sampling adalah teknik pengambilan sampel dengan penggantian. Dari dataset asli berukuran N , dibuat beberapa subset dataset (*bootstrap samples*) dengan penggantian.

$$\text{Bootstrap Sample} = \{x_i \mid i \sim \text{Uniform}(1, N)\} \quad (2.1)$$

2) Pembentukan Pohon Keputusan

Setiap pohon keputusan dibentuk dengan subset acak dari fitur-fitur yang tersedia. Pada setiap *node*, fitur terbaik dipilih untuk melakukan pemisahan data berdasarkan kriteria tertentu (misalnya, *Gini Impurity* atau *Entropy*).

Gini Impurity mengukur ketidaksempurnaan campuran kelas dalam *node*. Rumus *Gini Impurity* untuk *node t* adalah:

$$G(t) = 1 - \sum_{i=1}^c p_i^2 \quad (2.2)$$

di mana p_i adalah proporsi sampel yang termasuk dalam kelas i di node t , dan C adalah jumlah total kelas.

Entropy mengukur ketidaksempurnaan atau ketidakpastian dalam node. Rumus *Entropy* untuk *node t* adalah:

$$H(t) = - \sum_{i=1}^c p_i \log_2(p_i) \quad (2.3)$$

3) Pemilihan Fitur Terbaik

Fitur terbaik dipilih untuk memisahkan data pada setiap node berdasarkan pengurangan *Impurity* terbesar. Pengurangan impurity ΔI dihitung sebagai:

$$\Delta I = I(\text{parent}) - \left(\frac{N_{\text{left}}}{N} I(\text{left}) + \frac{N_{\text{right}}}{N} I(\text{right}) \right) \quad (2.4)$$

di mana I adalah *Impurity* (*Gini* atau *entropy*), N adalah jumlah total sampel, N_{left} dan N_{right} adalah jumlah sampel di node kiri dan kanan setelah pemisahan.

4) *Voting* atau *Averaging*

Setelah pohon-pohon keputusan dibentuk, prediksi dilakukan dengan mengumpulkan suara (*voting*) atau menghitung rata-rata prediksi dari semua pohon.

Untuk klasifikasi, setiap pohon memberikan satu suara untuk kelas *output*. Kelas dengan suara terbanyak dipilih sebagai prediksi akhir:

$$\hat{y} = \text{mode}(\{\hat{y}_1, \hat{y}_2, \dots, \hat{y}_T\}) \quad (2.5)$$

di mana \hat{y}_i adalah prediksi dari pohon ke- i dan T adalah jumlah total pohon.

Untuk regresi, rata-rata nilai prediksi dari semua pohon diambil sebagai prediksi akhir:

$$\hat{y} = \frac{1}{T} \sum_{i=1}^T \hat{y}_i \quad (2.6)$$

5) Penghitungan Akurasi

Akurasi dihitung dengan membandingkan prediksi model dengan nilai sebenarnya pada dataset pengujian:

$$\text{Akurasi} = \frac{\text{Jumlah prediksi benar}}{\text{Jumlah total data}} \times 100\% \quad (2.7)$$

6) Cross-Validation

Cross-validation adalah teknik evaluasi model dengan membagi data menjadi beberapa subset (*folds*) dan menjalankan beberapa iterasi pelatihan dan pengujian. Misalnya, dalam 10-fold cross-validation:

$$\text{Akurasi Rata - rata} = \frac{1}{k} \sum_{i=1}^k \text{Akurasi}_i \quad (2.8)$$

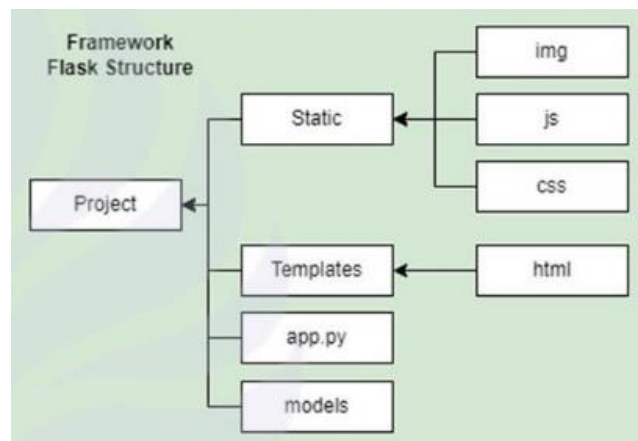
di mana k adalah jumlah *folds* (misalnya 10), dan Akurasi_i adalah akurasi pada *fold* ke- i .

2.4 Flask

Flask adalah kerangka kerja web yang dikembangkan menggunakan bahasa pemrograman *Python*. Kerangka kerja ini termasuk dalam kategori microframework. *Flask* menyediakan berbagai *library* yang memudahkan pengembang dalam membangun situs web tanpa perlu memulai dari awal. Dengan menggunakan *library* yang sudah ada dalam *Flask*, pengembang dapat membuat aplikasi sesuai dengan kebutuhannya dan dengan kreativitas yang tinggi. *Flask* memungkinkan penggunaan ekstensi tambahan seperti untuk *database*, validasi formulir, dan penanganan unggahan (*upload handling*). Keunggulan *Flask* lainnya

adalah fleksibilitasnya yang tinggi dibandingkan dengan kerangka kerja lainnya (Wibowo Putri & Susetyo, 2022). Kerangka kerja ini juga dikenal karena kesederhanaan dan kemudahannya dalam penggunaan (Ningrum & Utami, 2023), memungkinkan pengembang untuk membuat aplikasi dengan sesuai dengan kebutuhan dan permintaan pengguna.

Flask memiliki dua komponen utama, yakni *Werkzeug* dan *Jinja2*. *Werkzeug* bertanggung jawab atas penyediaan *routing*, *debugging*, dan *Web Server Interface Gateway* (WSGI), sementara *Jinja2* digunakan sebagai mesin *template*. Berbeda dengan *Django*, *Flask* tidak menyediakan fungsi-fungsi seperti akses basis data, otentikasi pengguna, dan perangkat "*high-level*" lainnya secara bawaan. Namun, *Flask* dapat diperluas fungsinya dengan menggunakan berbagai ekstensi tambahan (Ghanbari & Najafzadeh, 2020).



Gambar 2. 3 *Framework Flask*

Sumber : Screenshot dari jurnal (Ningrum & Utami, 2023)

2.5 Penelitian Terkait

Penelitian mengenai sistem rekomendasi pernah dilakukan oleh (Triawan & Lintang Melinda, 2020) membahas mengenai masalah ketidakakuratan dalam pemilihan topik tugas akhir oleh mahasiswa di perguruan tinggi dengan menerapkan Metode *Naïve Bayes*. Dalam pendekatan ini, sebuah aplikasi rekomendasi topik tugas akhir dibangun berdasarkan daftar hasil studi mahasiswa. Aplikasi tersebut membantu memberikan rekomendasi kepada mahasiswa dalam menentukan topik untuk tugas akhirnya. Uji kelayakan dilakukan terhadap aplikasi

yang dibuat, dengan nilai persentase kelayakan secara keseluruhan dari tiga indikator, yaitu *Efficiency*, *Effectiveness*, dan *Satisfaction*, sebesar 79%. Hal ini menandakan bahwa aplikasi masuk dalam kategori layak digunakan berdasarkan tabel kategori kelayakan skala *likert*. Selain itu, dilakukan komparasi antara output dari aplikasi yang telah dibuat dengan *output* dari alat data mining yang sudah ada, yang menghasilkan tingkat akurasi komparasi sebesar 60%.

Oleh (Faidhani et al., 2021) dengan penelitiannya dalam Sistem Pendukung Keputusan (SPK) untuk mahasiswa Program Studi Informatika di Universitas Tanjungpura dengan Metode *ELECTRE*. SPK memberikan rekomendasi bidang keahlian berdasarkan peringkat, dengan akurasi 40% pada pengujian 10 mahasiswa. Hasil menunjukkan kriteria minat memengaruhi penentuan bidang keahlian. Penelitian dengan kasus yang hampir serupa juga dilakukan oleh (Haviluddin et al., 2021) dengan mengimplementasikan metode *K-Means* untuk mengelompokkan rekomendasi Tugas Akhir (TA) mahasiswa. Berdasarkan eksperimen, dilakukan *cluster* area penelitian TA ke dalam 3 *cluster* (C), dengan C1 memiliki sedikit anggota (1 MKW), C2 sedang (6 MKW), dan C3 banyak (3 MKW). Pengujian akurasi *cluster* menggunakan metode *Sum of Squared Errors* (SSE) dan *Silhouette Coefficient* (SC) memberikan hasil SSE 0.6566 dan SC 5.8329. Hasil ini menunjukkan bahwa nilai MKW (Mata Kuliah Wajib) memiliki dampak dalam menentukan TA.

Penelitian berikutnya mengenai pemilihan judul Tugas Akhir menggunakan Metode *Naïve Bayes* dilakukan oleh (Yulindawati et al., 2024). Penelitian ini menyoroti pentingnya pencarian referensi sesuai dengan kemampuan dan minat mahasiswa, khususnya dalam Program Studi Teknik Informatika. Salah satu masalah yang dihadapi mahasiswa adalah kesulitan dalam menentukan topik Tugas Akhir karena kurangnya pemahaman terhadap matakuliah pendukung. Dari hasil perhitungan probabilitas, ditemukan bahwa mahasiswa yang memiliki konsentrasi pada jaringan memiliki peluang tertinggi untuk direkomendasikan ke kategori jaringan sebagai topik Tugas Akhirnya, dengan nilai probabilitas multimedia sebesar 0,0366306, jaringan sebesar 0,0026318, dan sistem cerdas sebesar 0.00090467, masing-masing. Data yang digunakan dalam analisis ini adalah nilai

mahasiswa. Penelitian lain mengenai sistem rekomendasi dengan *Random Forest* dilakukan oleh (Pratama et al., 2021) yang membandingkan berbagai model klasifikasi yang berbasis pembelajaran mesin untuk sistem rekomendasi pemilihan program studi sarjana. Dalam upaya meningkatkan aksesibilitas informasi, perguruan tinggi perlu mengadopsi inovasi, seperti penyediaan sistem rekomendasi pemilihan program studi. Hasil analisis menunjukkan bahwa model klasifikasi *single-stage* menggunakan Algoritma *Random Forest* memberikan performa terbaik dengan nilai akurasi mencapai 92%, melebihi model klasifikasi *multi-stages* dan *preliminary study* sebesar 6%. Penelitian yang dilakukan oleh Aini et al. (2024) mengimplementasikan Algoritma *Random Forest* untuk memetakan bidang MSIB di Program Studi Pendidikan Informatika. Dalam penelitian tersebut, atribut yang digunakan adalah nilai mata kuliah mahasiswa. Hasilnya menunjukkan tingkat akurasi 80%, *precision* 80%, dan *recall* 82%, sehingga model klasifikasi menggunakan *Random Forest* dinilai baik.

Tabel 2. 1 Perbandingan dengan penelitian terdahulu

No	Pustaka	Judul	Metode	Objek penelitian
1	(Triawan & Lintang Melinda, 2020)	Penerapan Metode <i>Naïve Bayes</i> Untuk Rekomendasi Topik Tugas Akhir Berdasarkan Daftar Hasil Studi Mahasiswa di Perguruan Tinggi	<i>Naïve Bayes</i>	Mahasiswa
2	(Faidhani et al., 2021)	Sistem Pendukung Keputusan Penentu Bidang Keahlian Mahasiswa Program Studi Informatika Universitas	<i>Elimination Choice and Translating Reality</i> (ELECTRE)	Mahasiswa Program Studi Informatika Universitas Tanjungpura

		Tanjungpura dengan Metode ELECTRE		
3	(Haviluddin et al., 2021)	Implementasi Metode <i>K-Means</i> untuk <i>Cluster</i> Rekomendasi Tugas Akhir	<i>K-Means</i>	Mahasiswa
4	(Yulindawati et al., 2024)	Rekomendasi Pemilihan Judul Tugas Akhir Menggunakan Metode Naïve Bayes	<i>Naïve Bayes</i>	Mahasiswa
5	(Pratama et al., 2021)	Studi Komparasi Model Klasifikasi Berbasis Pembelajaran Mesin untuk Sistem Rekomendasi Pemilihan Program Studi Sarjana	<i>Multinomial Logistic Regression, Random Forest, Support Vector Machine</i>	Mahasiswa
6	(Aini et al., 2024)	Implementasi Algoritma <i>Random Forest</i> untuk Klasifikasi Bidang MSIB di Prodi Pendidikan Informatika	<i>Random Forest</i>	Mahasiswa

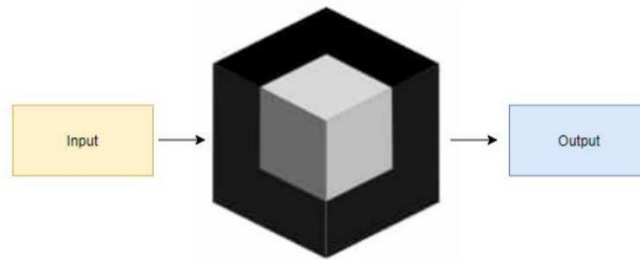
Berdasarkan pemaparan penelitian terkait diatas Metode *Random Forest* menunjukkan hasil yang cukup baik dalam melakukan pengklasifikasian termasuk dalam penelitian sistem rekomendasi dan sistem pendukung keputusan. Alasan lain mengapa memilih *Random Forest* untuk melakukan klasifikasi karena keunggulannya dalam mengatasi *overfitting*, kemampuannya untuk menangani dataset besar, serta kemudahannya dalam implementasi tanpa memerlukan penyetelan parameter yang rumit.

2.6 Pengujian

Pengujian merupakan tahapan krusial dalam menghasilkan sistem yang handal. Tanpa pengujian yang tepat, sulit untuk menilai kualitas sistem secara akurat, sehingga risiko kerusakan atau kesalahan dalam sistem dapat terjadi. Dengan melakukan pengujian dapat dilakukan identifikasi kelemahan sistem sejak awal dan memperbaikinya dengan cepat (Ijudin & Saifudin, 2020).

2.6.1 *Black Box Testing*

Pengujian *Black Box Testing*, yang sering disebut sebagai pengujian perilaku, merupakan metode di mana struktur internal dan logika perangkat lunak yang diuji tidak diketahui oleh penguji. Pada pengujian ini, penguji berfokus pada spesifikasi kebutuhan dan tidak perlu menganalisis kode secara detail. Pendekatan ini dilakukan dari perspektif pengguna akhir. Ada berbagai jenis pengujian *Black Box Testing*, termasuk partisi, analisis nilai batas, grafik penyebab efek, pengujian orthogonal array, pengujian transisi negara, dan *Fuzzing*. Meskipun memiliki kelebihan dalam menemukan aspek yang tidak terpenuhi dari spesifikasi kebutuhan perangkat lunak, *Black Box Testing* juga memiliki kekurangan karena pengujian tidak dapat dilakukan secara menyeluruh akibat keterbatasan pengetahuan penguji tentang perangkat lunak yang diuji (Praniffa et al., 2023).



Gambar 2. 4 Black Box *Testing*

Sumber : Screenshot dari jurnal (Ningrum & Utami, 2023)

2.6.2 Pengujian akurasi *Random Forest*

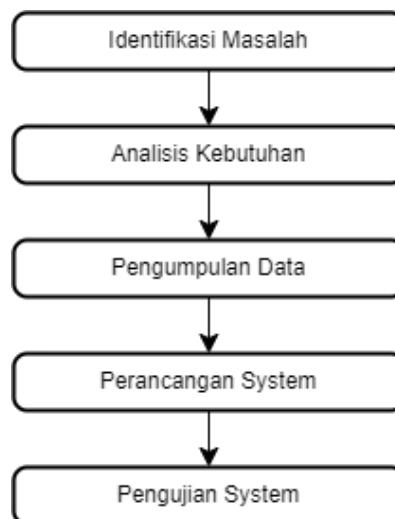
Dalam penelitian ini, proses pengujian akurasi Model *Random Forest* menjadi tahap krusial dalam evaluasi kinerja algoritma yang telah diterapkan. Pengujian dilakukan dengan menggunakan dataset yang telah terlebih dahulu dibagi menjadi dua bagian: dataset pelatihan untuk melatih model dan dataset uji untuk menguji model yang telah dilatih. Pertama, Model *Random Forest* akan dilatih menggunakan dataset pelatihan untuk mempelajari pola-pola yang ada dalam data. Setelah pelatihan selesai, model akan diuji menggunakan dataset uji yang terpisah untuk mengevaluasi seberapa baik model dapat menggeneralisasi pola yang telah dipelajari dari data pelatihan ke data baru. Akurasi merupakan parameter yang menunjukkan sejauh mana kemampuan model pengujian dalam mengklasifikasikan data secara tepat. Penentuan akurasi dilakukan dengan membagi jumlah klasifikasi yang benar oleh model dengan total data yang diuji (Saepudin et al., 2024). Evaluasi ini penting untuk memastikan bahwa model yang dikembangkan dapat memberikan rekomendasi topik tugas akhir yang tepat dan relevan bagi mahasiswa berdasarkan data nilai akademik dan data minat yang dimiliki.

BAB III METODOLOGI PENELITIAN

3.1 Tempat dan Waktu Penelitian

Penelitian ini dilakukan di Laboratorium Rekayasa Perangkat Lunak Program Studi S1 Terapan Teknik Komputer dan Jaringan Politeknik Negeri Ujung Pandang dengan waktu penelitian dilaksanakan mulai bulan Maret 2024 hingga Agustus 2024.

3.2 Prosedur Penelitian



Gambar 3. 1 Prosedur Penelitian

Gambar 3.1 adalah rangkaian prosedur penelitian yang akan di lakukan dalam membangun sistem rekomendasi topik tugas akhir. Prosedur penelitian ini terdiri atas lima proses untuk mencapai tujuan penelitian yang di inginkan yaitu mulai dari identifikasi masalah, analisis kebutuhan, pengumpulan data, perancangan sistem hingga pengujian system.

3.2.1 Identifikasi Masalah

Tahapan pertama yang dilakukan adalah mengidentifikasi serta membatasi masalah yang akan diselesaikan dalam kasus sistem rekomendasi topik tugas akhir mahasiswa menggunakan Metode *Random Forest*. Dengan mengetahui batasan masalah, maka dapat dilakukan proses analisis kebutuhan berdasarkan masalah

yang ada. Proses identifikasi masalah dilakukan melalui observasi dan studi literatur terkait. Adapun masalah yang diperoleh adalah sebagai berikut:

1. Keterlambatan kelulusan mahasiswa : Banyak mahasiswa yang mengalami keterlambatan dalam menyelesaikan studi karena kesulitan dalam menemukan dan menguasai topik skripsi yang sesuai dengan minat dan kemampuannya. Hal ini mengakibatkan penundaan dalam proses akademiknya.
2. Ketidaksesuaian topik dengan minat dan kemampuan : Mahasiswa sering kali memilih topik skripsi yang kurang sesuai dengan minat dan kemampuannya, karena kurangnya pemahaman mengenai bidang studi yang diminati. Ini dapat mengakibatkan penurunan motivasi dan kualitas penelitian.
3. Kesulitan pihak program studi dalam merekap data judul yang tidak konsisten : Pihak program studi sering menghadapi kesulitan dalam merekap dan memelihara data mengenai judul skripsi mahasiswa karena seringnya perubahan topik skripsi dan kurangnya konsistensi dalam pelaporan. Hal ini dapat menyulitkan proses pengelolaan dan pemantauan progres akademik mahasiswa.
4. Perlunya sistem rekomendasi berbasis teknologi : Dalam menghadapi kompleksitas dalam menentukan topik skripsi yang sesuai, diperlukan adanya sistem rekomendasi berbasis teknologi yang dapat membantu mahasiswa dan dosen pembimbing dalam menemukan topik skripsi yang sesuai dengan minat, kemampuan, dan kebutuhan akademiknya.

3.2.2 Analisis Kebutuhan

Kebutuhan penelitian bertujuan untuk mengidentifikasi kebutuhan apa saja yang akan digunakan dalam penelitian ini baik dari segi perangkat lunak maupun perangkat keras. Berikut adalah analisis kebutuhan dalam melaksanakan penelitian ini :

Tabel 3. 1 Kebutuhan Perangkat Keras

No	Nama Perangkat Keras	Spesifikasi	Justifikasi
1.	PC/Laptop	- <i>Processor Core i3 Gen 7/AMD Ryzen 3</i> - <i>RAM 4 GB</i> - <i>Storage 250 GB</i>	Sebagai perangkat penunjang penelitian

Tabel 3. 2 Kebutuhan Perangkat Lunak

No	Nama Perangkat Lunak	Justifikasi
1.	Sistem Operasi <i>Windows 10 Pro/Home</i>	Sebagai sistem penunjang penelitian
2.	<i>Visual Studio Code</i>	Sebagai aplikasi untuk membuat dan menjalankan kode
3.	<i>Microsoft Excel</i>	Penyimpanan data sebelum <i>Pra-prosesing</i> data awal dalam format CSV
4.	<i>Python</i>	Untuk mengimplementasikan Algoritma <i>Random Forest</i> dan melakukan pemrosesan data
5.	<i>Flask</i>	Framework Python yang digunakan untuk mengembangkan aplikasi web
6.	<i>Web Browser</i>	Untuk menguji dan melihat antarmuka pengguna yang dikembangkan dengan <i>Flask</i>

3.2.3 Pengumpulan Data

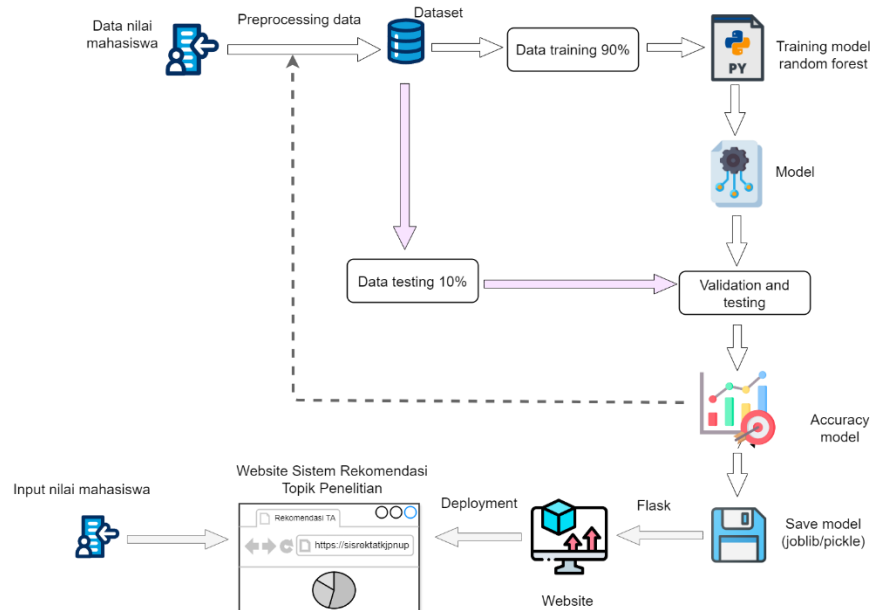
Pada tahap ini dilakukan pengumpulan data dengan beberapa cara berikut :

1. Studi Pustaka, mengamati referensi yang berhubungan dengan Metode *Random Forest* seperti buku, jurnal, skripsi, internet, dan sebagainya
2. Observasi, melakukan pengawasan langsung ke tempat objek untuk mendapatkan data yang berhubungan dengan tugas akhir atau skripsi

3. Wawancara, mengumpulkan data-data dengan cara tanya jawab langsung ke pimpinan atau beberapa pihak yang terlibat untuk mengetahui dan melihat kebutuhan pengguna terhadap sistem yang akan dibangun
4. Permintaan data terdahulu ke pihak jurusan TIK prodi TKJ PNUP berdasarkan prosedur yang berlaku. Data yang digunakan untuk *training* dan pembuatan model adalah data akademik mahasiswa angkatan 2017 hingga 2020 yaitu data nilai mata kuliah terkait dengan topik tugas akhir serta data lainnya yang dibutuhkan untuk mendukung keakurasian *training* model seperti data bidang keilmuan dan profile lulusan yang di minati.

3.3 Perancangan Sistem

Perancangan sistem merujuk pada penyusunan konseptual yang menggambarkan secara jelas desain sistem yang diinginkan dalam konteks penelitian ini. Alur sistem penelitian ini dapat dilihat pada gambar dibawah.

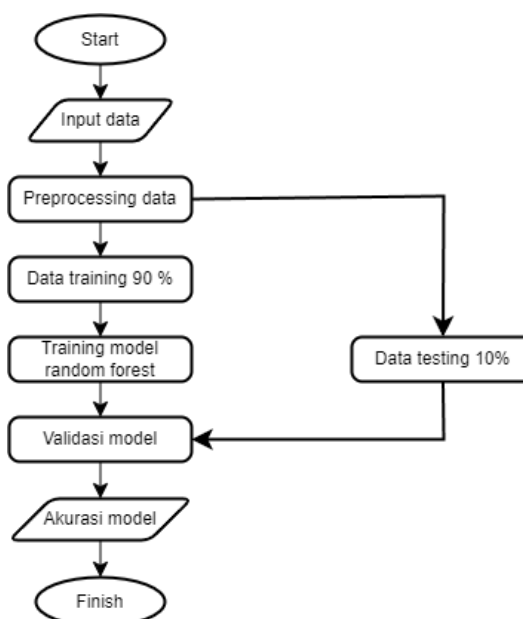


Gambar 3. 2 Arsitektur sistem

Gambar 3.2 adalah arsitektur sistem yang akan di gunakan dalam penelitian ini. Data nilai akademik dan minat mahasiswa diolah dan dipreproses untuk membentuk dataset, yang kemudian dipisahkan menjadi data *training* (90%) dan data *testing* (10%). Data *training* digunakan untuk melatih Model *Random Forest*. Model yang diperoleh setelah *training* menggunakan *Random Forest* kemudian

dievaluasi menggunakan data *testing*. Setelah evaluasi, Model *Random Forest* disimpan untuk integrasi dengan aplikasi sistem rekomendasi. Sistem ini menerima data mahasiswa dan menggunakan model untuk memberikan rekomendasi topik tugas akhir. Hasil rekomendasi disajikan melalui antarmuka pengguna yang dibangun menggunakan *Flask*. Ini memungkinkan pengguna dengan mudah menggunakan sistem untuk memilih topik tugas akhir.

3.3.1 Penerapan *Random Forest*



Gambar 3. 3 Flowchart penerapan *Random Forest*

Gambar 3.3 diatas memaparkan alur penerapan *Random Forest* untuk penelitian ini. Berikut adalah penjelasan lebih lengkapnya :

1. Data *Source*

Dalam tahap ini, data awal yang diperlukan untuk pelatihan model diperoleh dari sumber-sumber yang relevan, seperti basis data kampus atau repositori data lainnya. Data akan mencakup catatan akademik mahasiswa yang meliputi nilai mata kuliah, bidang keilmuan yang disukai dan profile lulusan yang di minati

2. Data *Preparation*

Data yang diperoleh dari sumber kemudian disiapkan untuk analisis. Tahap ini melibatkan pembersihan data untuk mengatasi nilai yang hilang, duplikat, atau

tidak valid. Fitur-fitur yang tidak relevan atau kurang informatif juga dapat dihapus agar fokus pada fitur-fitur yang lebih penting dalam prediksi.

3. *Feature Engineering*

Fitur-fitur yang dianggap penting dalam prediksi topik skripsi dipilih untuk digunakan dalam model. Pengkodean variabel kategorikal dilakukan jika diperlukan, dan fitur-fitur baru mungkin dibuat berdasarkan informasi yang ada untuk meningkatkan performa model.

4. Pembagian Data

Data kemudian dibagi menjadi dua bagian: data pelatihan (*training data*) dan data uji (*test data*). Sebagian besar data (biasanya sekitar 70-90%) digunakan untuk pelatihan model, sedangkan sisanya digunakan untuk menguji performa model.

5. Pelatihan Model

Model *Random Forest* dipilih dan diterapkan untuk melatih data pelatihan. Model dilatih untuk mengklasifikasikan atau merekomendasikan topik skripsi berdasarkan fitur-fitur yang ada dalam data pelatihan.

6. Validasi Model

Model yang telah dilatih dievaluasi menggunakan data uji yang tidak dilihat sebelumnya. Performa model dievaluasi menggunakan metrik evaluasi seperti akurasi, presisi, *recall*, atau metrik lainnya yang sesuai dengan kasus Anda.

7. Pengujian dan Evaluasi

Model yang telah disesuaikan dievaluasi kembali untuk memastikan kinerjanya sebelum diterapkan secara luas. Pengujian sensitivitas model terhadap data baru juga dapat dilakukan untuk mengukur keandalannya.

8. Penerapan Model

Model yang telah dilatih dan divalidasi siap untuk diterapkan pada data baru atau skenario produksi. Model dapat digunakan untuk memberikan rekomendasi topik skripsi kepada mahasiswa berdasarkan data yang diberikan.

9. Pemeliharaan dan Pemantauan

Setelah model diterapkan, perlu dilakukan pemantauan dan pemeliharaan secara berkala untuk memastikan konsistensi dan kualitasnya. Pengukuran kinerja

model juga perlu dilakukan secara teratur untuk memastikan bahwa model tetap relevan dan akurat seiring waktu.

3.3.1.1 Proses Pengelompokkan dan *Classification*

1. Parameter Input : Data nilai matakuliah mahasiswa digunakan sebagai parameter utama dalam proses pengelompokkan dan klasifikasi.

Tabel 3.1 Kategori Mata Kuliah

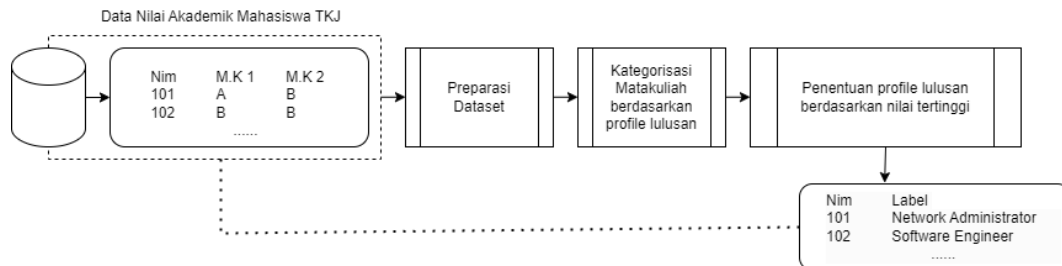
No	Mata kuliah	Profile Lulusan	Bidang Keilmuan
1	Sistem Embedded dan IoT	<i>IoT Engineer</i>	<i>Internet of Things (IoT)</i>
2	Rekayasa Perangkat Lunak	<i>Software Engineer</i>	<i>AI, Software Engineering</i>
3	Desain dan Pemrograman Web		
4	Pemrograman Berorientasi Objek		
5	Sistem Digital		
6	Sistem Terdistribusi		
7	Aplikasi Mobile		
8	Dasar Pemrograman		
9	Rekayasa Web		
10	Statistik dan Probabilitas	<i>Associate Data Engineer</i>	<i>Data Analytics, DBMS, Big Data</i>
11	Big Data		
12	Administrasi Basis Data		
13	Basis Data		
14	Sistem Cerdas		
15	Metode Komputasi		
16	Desain Algoritma		
17	Struktur Data & Analisis Algoritma		
18	Cyber Security	<i>Network Administrator</i>	<i>Information and Network Security, Cloud Computing, Network Performance</i>
19	Virtualisasi dan Komputasi Awan		
20	Teknologi Jaringan Komputer		
21	Rekayasa Jaringan Komputer Dasar		

22	Jaringan Wireles & Mobile		
23	Rekayasa Jaringan Komputer Lanjut		
24	Bengkel Komputer & Jaringan		
25	Komunikasi Data		
26	Jaringan Komputer		
27	Manajemen dan Desain Jaringan		
28	Administrasi Jaringan		
29	Arsitektur Komputer		
30	Sistem Operasi		

2. Pelabelan berbasis aturan :

- 1) Pengambilan Data: Data mahasiswa diambil dari basis data yang berisi informasi NIM, nama dan nilai mata kuliah.
- 2) Persiapan Dataset: Data mentah yang berisi NIM dan nilai mata kuliah dipersiapkan. Setiap mahasiswa memiliki nilai untuk setiap mata kuliah yang diambil.
- 3) Kategorisasi mata kuliah berdasarkan profil lulusan: Mata kuliah dikelompokkan berdasarkan kategori profil lulusan. Misalnya, mata kuliah yang berkaitan dengan '*Network Administrator*', '*Software Engineer*', '*Associate Data Engineer*', atau '*IoT Engineer*'.
- 4) Penghitungan nilai rata-rata untuk setiap profil: Nilai rata-rata dihitung untuk setiap kategori profil lulusan. Ini melibatkan mengambil nilai rata-rata dari semua mata kuliah yang termasuk dalam satu profil lulusan.
- 5) Penentuan profil lulusan berdasarkan nilai tertinggi: Mahasiswa diberi label berdasarkan profil lulusan yang memiliki nilai rata-rata tertinggi dari mata kuliah yang berkaitan. Misal, jika nilai tertinggi rata-rata adalah pada mata kuliah yang berkaitan dengan '*Software Engineering*', maka mahasiswa tersebut diberi label '*Software Engineering*'.

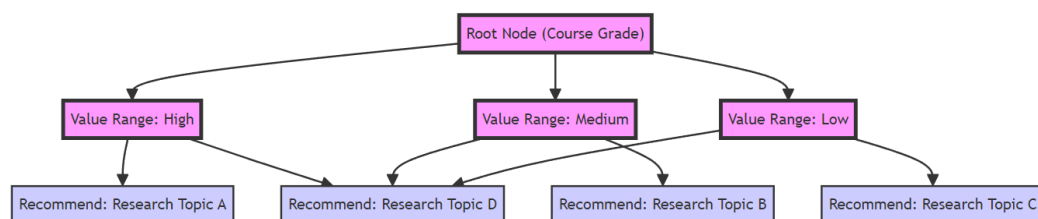
- 6) Output: Dataset akhir berisi tambahan atribut yang akan digunakan sebagai label yang menunjukkan profil lulusan yang paling sesuai berdasarkan nilai mata kuliah tertinggi dari rata-rata mata kuliah yang berkaitan.



Gambar 3. 4 Ilustrasi Pengelompokkan Mahasiswa

3. Klasifikasi menggunakan *Random Forest* :

- 1) Pemilihan sampel data : Data yang telah diberi label pengelompokkan digunakan sebagai sampel pelatihan untuk Model *Random Forest*.
- 2) Pembuatan pohon keputusan : Model *Random Forest* terdiri dari banyak pohon keputusan, di mana setiap pohon dilatih pada subset acak dari data pelatihan.
- 3) Pemilihan fitur terbaik : Setiap pohon memilih fitur terbaik untuk membagi data berdasarkan kriteria yang mengoptimalkan pemisahan kelas.
- 4) Agregasi prediksi : Prediksi dari setiap pohon digabungkan menggunakan teknik mayoritas suara untuk menentukan prediksi akhir.



Gambar 3. 5 Ilustrasi Proses Klasifikasi *Random Forest* dalam setiap Decision Tree

Gambar diatas menjelaskan di mana sistem mengambil keputusan dengan melihat nilai mata kuliah sebagai titik awal (*Root Node*), mirip dengan pertanyaan awal "Bagaimana nilai mata kuliah ini?" Selanjutnya, data dibagi menjadi tiga kelompok berdasarkan rentang nilai, ditampilkan sebagai *node* dan cabang.

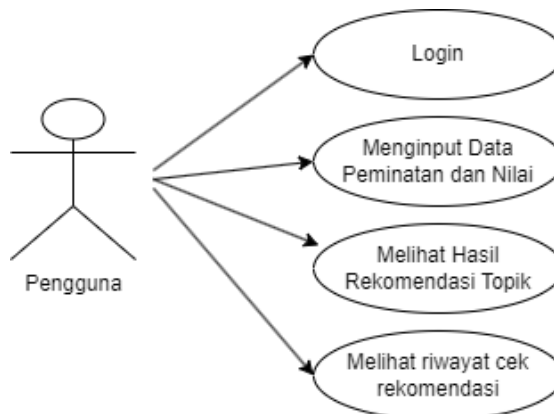
Analoginya, seperti memilah kertas ujian berdasarkan skornya menjadi kelompok tinggi, sedang, dan rendah. Setiap kelompok kemudian menerima rekomendasi yang berbeda sebagai hasil akhir (Daun - Rekomendasi), misalnya, kelompok dengan skor tinggi akan direkomendasikan topik tugas akhir A, sementara kelompok lain mendapat rekomendasi yang berbeda. Dalam implementasinya, banyak proses serupa atau "pohon" digabungkan untuk menghasilkan satu rekomendasi yang kuat, dikenal sebagai *Random Forest*.

4. Hasil Klasifikasi: Prediksi akhir dari Model *Random Forest* memberikan rekomendasi topik tugas akhir untuk setiap data berdasarkan fitur-fitur yang telah diberikan.

3.3.2 Desain Implementasi *Website*

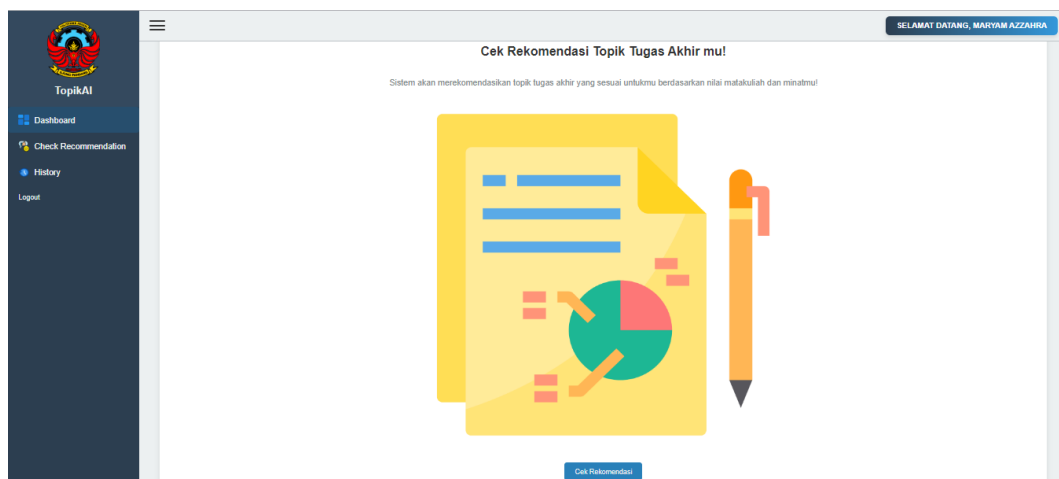


Gambar 3. 6 *Flowchart* Sistem *Website*




Gambar 3. 7 Use Case Diagram

Use Case diagram mendeskripsikan mengenai apa yang dapat pengguna lakukan pada sistem. *Use Case* author dapat menginputkan data peminatan dan nilai serta *Use Case* author dapat melihat hasil rekomendasi topik tugas akhir dari proses perhitungan. *Use Case* sistem rekomendasi dapat dilihat pada Gambar 3.7




Gambar 3. 8 Halaman Utama




TopikAI

- Dashboard
- Check Recommendation
- History
- Logout




TopikAI

- Dashboard
- Check Recommendation
- History
- Logout



TopikAI

- Dashboard
- Check Recommendation
- History
- Logout



TopikAI

- Dashboard
- Check Recommendation
- History
- Logout

Cek Rekomendasi Topik Tugas Akhir!

Berikut adalah nilai, skalan cek dan pilih minatu untuk mendapatkan rekomendasi topik tugas akhir.

arsitektur komputer

A

komunikasi data

B+

bangket komputer jaringan

A-

dasar pemrograman

B+

struktur data analisis algoritma

A-

sistem digital

A

jaringan komputer

A-

statistik dan probabilitas

A-

metode komputasi

A

desain algoritma

A-

teknologi jaringan komputer

A-

basis data

A

sistem operasi

A

rekayasa perangkat lunak

B

rekayasa jaringan komputer dasar

B

administrasi basis data

A

pemrograman berorientasi objek

B

desain dan pemrograman web

A-

sistem cerdas

A

administrasi jaringan

B

rekayasa jaringan komputer lanjut

A-

sistem terdistribusi

A

sistem embedded dan iot

B+

aplikasi mobile

A-

manajemen dan desain jaringan

A-

rekayasa web

A

manajemen dan desain jaringan

A-

rekayasa web

A

virtualisasi dan komputasi awan

A-

cyber security

A-

jaringan wireless mobile

A

big data

A-

Minat Bidang Ilmu:

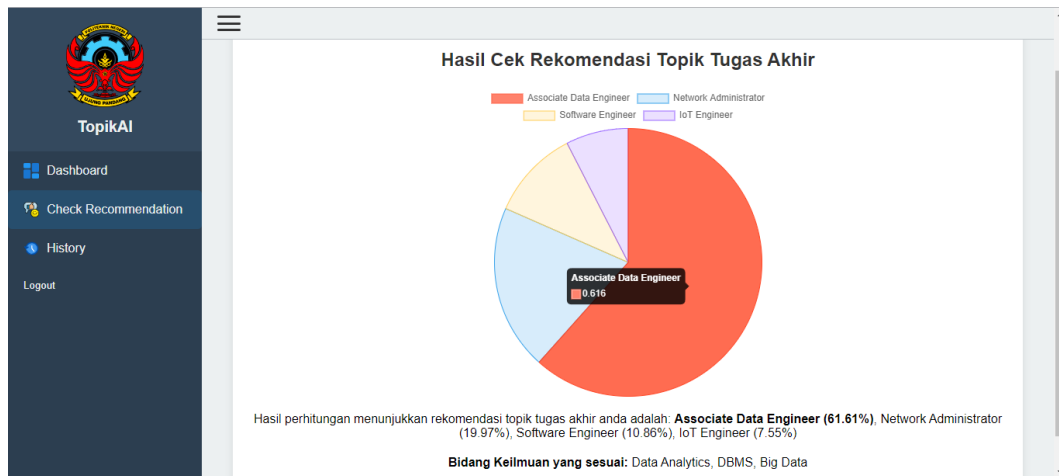
IoT (Internet of Things)

Minat Profile Lulusan:

Network Administrator

Check Recommendation

Gambar 3. 9 *Form* Tampilan Nilai dan Input Minat



Gambar 3. 10 Halaman Hasil Rekomendasi

Mockup website yang direncanakan akan menampilkan tampilan seperti yang digambarkan di atas. Halaman pertama adalah halaman *dashboard*, pengguna dapat memanfaatkan fitur cek rekomendasi topik skripsi, lihat *profile*, atau *logout*. Di halaman beranda atau *dashboard* menampilkan tombol untuk melakukan cek rekomendasi, pada halaman cek rekomendasi pengguna akan menemukan formulir di mana mahasiswa dapat memasukkan data akademik. Dalam *form* ini, pengguna dapat memasukkan nilai mata kuliah berdasarkan mata kuliah akademik yang berkaitan dengan topik tugas akhir yang diperoleh selama perkuliahan, dengan pilihan nilai yang terdiri dari 'A', 'A-', 'B+', 'B', 'B-', 'C+', 'C', 'C-', 'D', dan 'E'. Setelah mengisi semua nilai mata kuliah, maka akan diarahkan ke halaman selanjutnya untuk memasukkan data bidang keilmuan yang disukai dan profile lulusan yang diminati.

Setelah pengguna menginputkan data, informasi tersebut akan diproses menggunakan Algoritma *Random Forest*. Setelah proses selesai halaman akan beralih ke tampilan hasil rekomendasi untuk pengguna. Hasil topik rekomendasi akan disajikan dalam bentuk diagram lingkaran (*Pie Chart*) agar pengguna dapat dengan mudah memahami persentase dari berbagai kemungkinan rekomendasi yang diberikan. Topik rekomendasi yang ditampilkan pada *chart* di kategorikan dalam 4 *profile* lulusan yaitu *Network Administrasi*, *Software Engineer*, *Associate Data Engineer*, dan *IoT Engineer*. Informasi utama tentang rekomendasi akan

ditampilkan di bawah diagram lingkaran untuk memberikan pemahaman yang lebih mendalam kepada pengguna.

3.4 Pengujian

Pengujian merupakan tahap penting dalam mengevaluasi kinerja sistem yang telah dikembangkan. Dalam konteks pengujian untuk sistem rekomendasi topik tugas akhir mahasiswa menggunakan *Random Forest*, terdapat beberapa metode yang akan dilakukan.

1. Pengujian Akurasi *Random Forest*

Pengujian ini bertujuan untuk mengevaluasi seberapa baik Model *Random Forest* yang telah dilatih dalam memprediksi topik skripsi mahasiswa. Dilakukan dengan membagi data menjadi dua bagian, yaitu data pelatihan dan data uji. Model kemudian diuji menggunakan data uji yang tidak pernah dilihat sebelumnya, dan akurasi prediksinya dihitung untuk menentukan seberapa baik model dapat melakukan prediksi dengan benar.

2. Pengujian *Black Box*

Pengujian ini dilakukan untuk menguji kinerja model secara keseluruhan tanpa perlu mengetahui detail internal dari algoritma yang digunakan. Metode ini memperhatikan keluaran atau respons model terhadap berbagai masukan tanpa memperhatikan bagaimana model tersebut membuat keputusan. Dengan demikian, pengujian *Black Box* dapat memberikan gambaran tentang seberapa baik Model *Random Forest* dapat menangani berbagai situasi atau masukan yang berbeda.

3. Penggunaan Kuesioner

Selain pengujian teknis, penggunaan kuesioner juga penting untuk mendapatkan umpan balik langsung dari pengguna, dalam hal ini mahasiswa dan kemungkinan pihak terkait lainnya seperti dosen pembimbing. Kuesioner dirancang untuk mengevaluasi kepuasan pengguna terhadap rekomendasi topik skripsi yang diberikan oleh sistem, serta untuk mengidentifikasi area-area yang perlu diperbaiki atau ditingkatkan. Dengan demikian, kuesioner dapat memberikan

wawasan yang berharga untuk pengembangan dan peningkatan sistem rekomendasi.

BAB IV HASIL DAN PEMBAHASAN

Pada bagian ini dijelaskan mengenai hasil dan pembahasan dari pengumpulan data, perancangan sistem, dan pengujian yang digunakan seperti pengujian Algoritma *Random Forest*, pengujian *Black Box*, dan pengujian kuesioner

4.1 Pengumpulan Data

Pada tahap ini dihasilkan data nilai akademik real diambil dari data program studi Teknik Komputer Jaringan di SIMPONI (Sistem Informasi Akademik Politeknik Negeri Ujung Pandang). Dan data minat mahasiswa diperoleh dari hasil kuesioner terhadap alumni 2017-2019 dan mahasiswa akhir TKJ angkatan 2020, mahasiswa atau alumni yang tidak mengisi kuesioner datanya di ambil dari judul skripsi yang di rekap dari simponi dan situs repositori skripsi kampus. Data yang berhasil di kumpulkan untuk pembuatan model sebanyak 213 data mahasiswa. Data matakuliah di kumpulkan dengan memilah matakuliah yang dapat digunakan dalam proses *training*, yaitu 30 mata kuliah yang berfokus pada kejuruan/prodi di mana setiap mata kuliah dikelompokkan ke dalam bidang keilmuan dan profile lulusan yang bersangkutan. Selain daripada mata kuliah bersangkutan tidak di masukkan kedalam dataset karena tidak diperlukan. Gambaran data nilai akademik dan minat mahasiswa adalah seperti pada gambar 4.1

Nim	Nama Mahasiswa	Arsitektur Komputer	Komunikasi Data	Bengkel Komputer & Jaringan	Dasar Pemrograman	Struktur Data & Analisis Algoritma	Sistem Digital	Jaringan Komputer	Statistik dan Probabilitas	Metode Komputasi	Desain Algoritma	Teknologi Jaringan Komputer	Basis Data	Sistem Operasi	Rekayasa Perangkat Lunak	Rekayasa Jaringan Komputer Dasar	Administrasi Basis Data	Pemrograman Berorientasi Objek
42517002	MUHAMMAD RAHMAT	B+	C	A	C	A	B+	B	B+	B	B	B	B+	B	A	C+	A	C
42517003	RIKA HARIANTI	B+	C+	A	B+	B	B+	B+	A	A	B	A	A	A	A	B+	A	B+
42517004	MUHAMMAD YUSRIL ZAENAL	B	C+	A	B+	A	A	B+	A	B+	B+	A	A	A	A	B+	A	B
42517005	ANDI FACHRUL REZA	B+	C+	B+	C	C+	B+	C+	A	A	B+	B+	B+	A	A	B+	B	B
42517006	SEPTY AINUN ARIATI	B+	B	A	C+	A	A	A	A	A	B	A	A	A	A	B+	A	B
42517007	A. M. YUSRIL IKA RAMADHAN	A	B	A	B+	A	C+	B	A	B	B	A	B+	A	A	B	A	B+
42517008	HASRI AINUN IKA PERTIWI	B	A	A	C	B	B+	B+	B+	C	B+	A	A	A	A	B+	A	B

Gambar 4. 1 Data Nilai Akademik

4.2 Perancangan Sistem

Tujuan utama dari perancangan ini adalah untuk memastikan bahwa sistem yang dibangun dapat berfungsi sesuai dengan spesifikasi yang diinginkan, efisien, dan mudah digunakan oleh pengguna.

4.2.1 Praprocessing Data

Setelah melalui tahap pengumpulan data, praprocessing data dilakukan sebelum melakukan proses pelabelan yang mencakup beberapa langkah utama: mengonversi nilai-nilai grade menjadi angka numerik menggunakan *dictionary grade_dict*. Nilai yang hilang (*missing value*) pada kolom numerik kemudian diisi dengan rata-rata kolom tersebut. Kolom-kolom kategorikal, seperti "Minat Bidang Ilmu" dan "Minat Profile Lulusan", diencode menggunakan *LabelEncoder*.


4.2.2 Pemrograman untuk menentukan label

Data terlebih dahulu perlu diketahui labelnya atau kategori profile lulusannya untuk dapat melakukan klasifikasi menggunakan *Random Forest*, maka dilakukan proses penentuan label (profil lulusan) menggunakan logika berbasis aturan. Hasil pengelompokkannya akan digunakan sebagai label. Proses pengelompokkan ini memperhatikan beberapa kondisi yaitu menghitung dan menganalisis data nilai akademik rata rata mahasiswa paling tinggi terdapat pada kategori profile lulusan apa. Berikut adalah tabel kelompok setiap matakuliah.

Tabel 4. 1 Kelompok Mata Kuliah

No	Mata Kuliah	Network Admin	Software Engineer	Data Engineer	IoT Engineer
1	arsitektur_komputer	✓			
2	komunikasi_data	✓			
3	bengkel_komputer_jaringan	✓			
4	jaringan_komputer	✓			
5	teknologi_jaringan_komputer	✓			
6	sistem_operasi	✓			
7	rekayasa_jaringan_komputer_dasar	✓			✓
8	administrasi_jaringan	✓			

9	rekayasa_jaringan_komputer_lanjut	✓			✓
10	manajemen_dan_desain_jaringan	✓			
11	virtualisasi_dan_komputasi_awan	✓			
12	cyber_security	✓			✓
13	jaringan_wireless_mobile	✓			✓
14	dasar_pemrograman		✓		
15	sistem_digital		✓		
16	rekayasa_perangkat_lunak		✓		
17	pemrograman_berorientasi_objek		✓		
18	desain_dan_pemrograman_web		✓		
19	sistem_terdistribusi		✓		✓
20	aplikasi_mobile		✓		
21	rekayasa_web		✓		
22	sistem_cerdas		✓	✓	✓
23	struktur_data_analisis_algoritma			✓	
24	statistik_dan_probabilitas			✓	
25	metode_komputasi			✓	✓
26	desain_algoritma			✓	
27	basis_data			✓	
28	administrasi_basis_data			✓	
29	big_data			✓	
30	sistem_embedded_dan_iot				✓

 DATASET GABUNG CLUSTER.c...

 clustered_data_berd_kondisi.csv

Gambar 4. 2 File Data Awal dan Hasil Pengelompokkan

	Minat Bidang Ilmu	Minat Profile Lulusan	Highest_Avg_Field \
0	3	0	Avg_Data_Engineer
1	7	3	Avg_Data_Engineer
2	4	1	Avg_Network_Admin
3	7	3	Avg_Data_Engineer
4	3	1	Avg_IoT_Engineer
..
208	7	3	Avg_Network_Admin
209	4	0	Avg_Network_Admin
210	7	3	Avg_Network_Admin
211	2	0	Avg_Network_Admin
212	4	0	Avg_Network_Admin

	Predicted Profile
0	Associate Data Engineer
1	Associate Data Engineer
2	Network Administrator
3	Associate Data Engineer
4	IoT Engineer
..	...
208	Network Administrator
209	Network Administrator
210	Network Administrator
211	Network Administrator
212	Network Administrator

Gambar 4. 3 Hasil Pengelompokkan

4.2.3 Pemrograman *Random Forest*

Selanjutnya tahap yang perlu kembali dilakukan sebelum melakukan pemrograman *Random Forest* adalah pre-processing tambahan. *Pre Processing* tambahan dilakukan agar dataset dapat di proses sesuai dengan tujuan yang di inginkan sebelum di latih. Data yang telah di bersihkan dan siap digunakan kemudian di simpan kedalam folder yang sama dengan program Model *Random Forest* yang akan digunakan agar file dataset dapat terbaca.

Pada dataset dilakukan penghapusan fitur identitas yaitu “NIM” dan “nama_mahasiswa” dengan alasan fitur NIM dapat mempengaruhi dan mengganggu proses *training* data, di mana seharusnya NIM ini hanya sebagai identitas namun program akan melakukan *training* juga pada data ini, maka untuk menghindari kesalahan prediksi fitur tersebut di hapus. Berbeda dengan fitur “nama mahasiswa”, fitur ini tidak mempengaruhi proses dan hasil *training* jika ada atau tidak ada. Diputuskan pula untuk menghapusnya untuk menghindari hal-hal yang tidak di inginkan saat membuat program, misalnya lupa menuliskan fitur yang terdapat pada dataset. “Predicted Profile” adalah label hasil pemrograman logika berbasis aturan yang dihitung berdasarkan profile dengan rata-rata nilai tertinggi. Gambaran dataset csv yang digunakan dapat dilihat pada gambar berikut.

sistem_ter	sistem_em	aplikasi_m	manajeme	rekayasa_	virtualisasi	cyber_secu	jaringan_w	big_data	Minat Bid	Minat Prof	Predicted Profile
B+	B+	B+	B	A	A-	B+	A	B+	DBMS / Bi	Associate I	Associate Data Eng
B	B+	B	B	C+	A-	B	A	C+	Software E	Software E	Associate Data Eng
B+	B+	B+	B	A	A-	B+	A	C	Data Analy	Software E	Network Administr
B+	A	B+	A	A	A-	A	A	A	Software E	Software E	Associate Data Eng
B+	A	C+	A	A	A-	A	A	A	Software E	Network A	IoT Engineer
B+	A	B+	B+	B+	A-	B+	A	B	Data Analy	Network A	Associate Data Eng
A	A	B+	B+	B	A-	A	B	C	Software E	Software E	Network Administr
B+	A	A	A	B+	A-	B	A	B	DBMS / Bi	Network A	Network Administr
B+	B+	B	A	C+	A-	B	A	B	Software E	Software E	Network Administr
B+	B+	B+	A	A	A-	A	A	B+	AI (Artifici	Associate I	IoT Engineer
C+	B+	B	B+	C+	A-	B+	A	B+	AI (Artifici	Network A	IoT Engineer
A	B	A	A	A	A-	A	A	B+	AI (Artifici	Associate I	Associate Data Eng
A	B	B+	A	A	A-	A	A	B	Networkin	Network A	Network Administr

Gambar 4. 4 Dataset yang akan digunakan

Pemrograman *Random Forest* dilakukan dengan menggunakan bahasa pemrograman *Python* dan pustaka-pustaka seperti *pandas*, *scikit-learn*, dan *joblib*. Langkah pertama dalam pemrograman adalah memuat dataset yang telah dikumpulkan dan dibersihkan. Dataset ini kemudian diolah lebih lanjut dengan mengonversi nilai-nilai grade menjadi bentuk numerik yang dapat diproses oleh model, dapat dilihat pada tabel 4.2. Pengisian nilai hilang pada kolom numerik dilakukan dengan menggunakan rata-rata dari kolom tersebut, untuk memastikan tidak ada data yang hilang yang dapat mengganggu proses pelatihan model.

Tabel 4. 2 Konversi Grade ke Nilai Numerik

Grade	Nilai Numerik
A	4
A-	3.7
B+	3.3
B	3
B-	2.7
C+	2.3
C	2
C-	1.7
D	1.3
D-	1
E	0
T	0

Setelah data diolah, langkah berikutnya adalah *encoding* kolom-kolom kategorikal. Proses ini menggunakan teknik *Label Encoding* yang mengubah data kategorikal menjadi nilai numerik yang unik. Kolom-kolom seperti 'Minat Bidang Ilmu' dan 'Minat Profile Lulusan' yang berisi data kategorikal diencoding menjadi

angka. Sebagai contoh, kategori pada kolom 'Minat Bidang Ilmu' seperti *Software Engineering*, *IoT (Internet of Things)*, *AI (Artificial Intelligence)*, *Cyber Security*, *DBMS / Big Data*, *Networking*, dan *Data Analytics* diencoding menjadi angka yang unik. Data yang telah diencoding kemudian dipisahkan menjadi fitur (X) dan target (y). Fiturnya adalah data nilai akademik, minat bidang ilmu, dan minat profil lulusan sedangkan targetnya adalah *Predicted Profile*.

Normalisasi data numerik juga dilakukan untuk memastikan bahwa semua fitur berada dalam skala yang sama. Normalisasi ini penting untuk meningkatkan kinerja model, karena fitur yang memiliki skala yang berbeda dapat mempengaruhi hasil prediksi. Setelah data siap, langkah berikutnya adalah menangani ketidakseimbangan kelas dalam data. Teknik *RandomOverSampler* digunakan untuk menyeimbangkan distribusi kelas dengan cara menambahkan sampel dari kelas minoritas secara acak hingga distribusi kelas menjadi lebih seimbang. Dengan cara ini, distribusi kelas menjadi lebih seimbang, sehingga model dapat belajar dari data dengan lebih efektif.

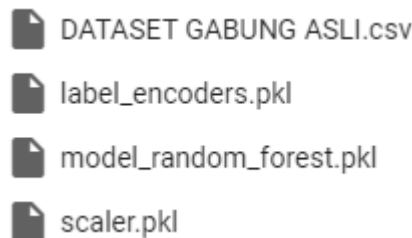
Setelah data seimbang, langkah berikutnya adalah melakukan pencarian hyperparameter untuk menemukan kombinasi hyperparameter terbaik yang dapat meningkatkan kinerja model. Pencarian *hyperparameter* dilakukan menggunakan *Randomized Search CV*. Dalam pencarian ini, beberapa hyperparameter seperti jumlah pohon dalam hutan (*n_estimators*), kedalaman maksimum pohon (*max_depth*), jumlah minimum sampel per *split* (*min_samples_split*), jumlah minimum sampel per daun (*min_samples_leaf*), dan penggunaan *bootstrap* dipertimbangkan. Proses pencarian ini melibatkan iterasi melalui sejumlah besar kombinasi hyperparameter untuk menemukan set yang memberikan hasil terbaik berdasarkan *cross-validation*.

Setelah hyperparameter terbaik ditemukan, model dievaluasi menggunakan *Cross-validation* untuk memastikan kinerja yang stabil. *Cross-validation* membagi data menjadi beberapa subset (*folds*) dan melatih serta menguji model pada subset-subset ini secara bergantian. Dengan cara ini, kita dapat memastikan bahwa model tidak *overfitting* dan dapat bekerja dengan baik pada data yang belum pernah dilihat sebelumnya. Model yang telah dilatih kemudian disimpan menggunakan *joblib*,

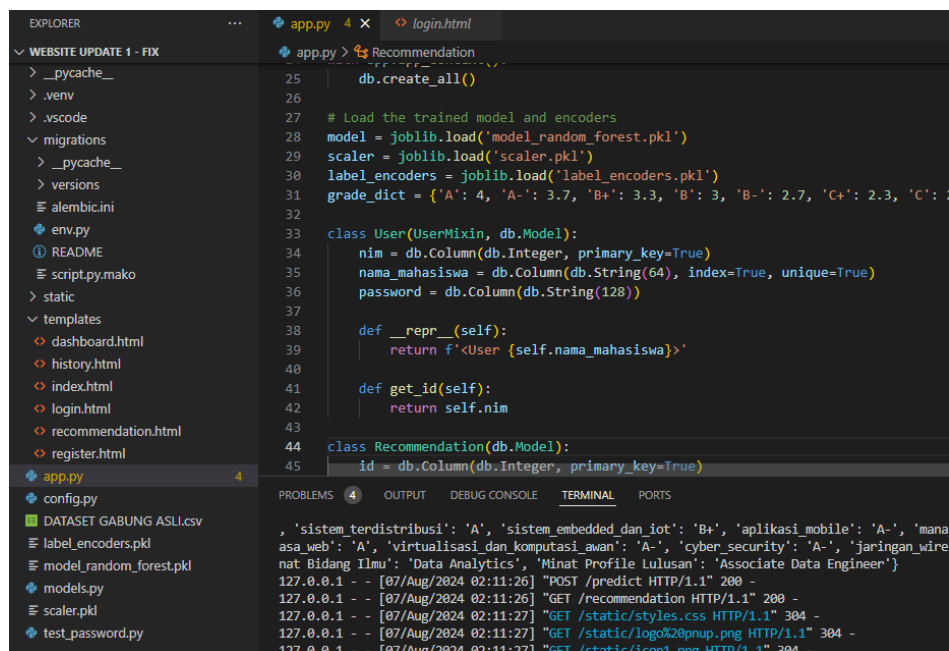
sehingga dapat digunakan kembali di kemudian hari tanpa perlu melatih ulang dari awal. *Encoder* yang digunakan untuk mengubah data kategorikal menjadi numerik juga disimpan untuk memastikan konsistensi dalam proses prediksi di masa depan.

4.2.4 Implementasi dan *Deployment* Model *Random Forest*

Model *Random Forest* yang menyimpan setiap fitur dari dataset telah disimpan secara lokal di *Google Colaboratory*. Agar dapat digunakan lebih lanjut, model ini disimpan dalam format *pickle* untuk diterapkan pada *website* dengan framework *Flask*. Hasil penyimpanan model dapat dilihat pada gambar.



Gambar 4. 5 Dataset dan Model Hasil *Random Forest*



Gambar 4. 6 *Deployment* Model

Aplikasi berbasis website ini dibuat menggunakan bahasa pemrograman *Python*. Model *Random Forest* yang telah dibuat akan dibaca sebagai salah satu aset yang akan dipanggil dalam aplikasi. Untuk melakukan prediksi, dibuat kode *Python* yang memungkinkan *user* untuk melihat prediksi topik tugas akhir berdasarkan

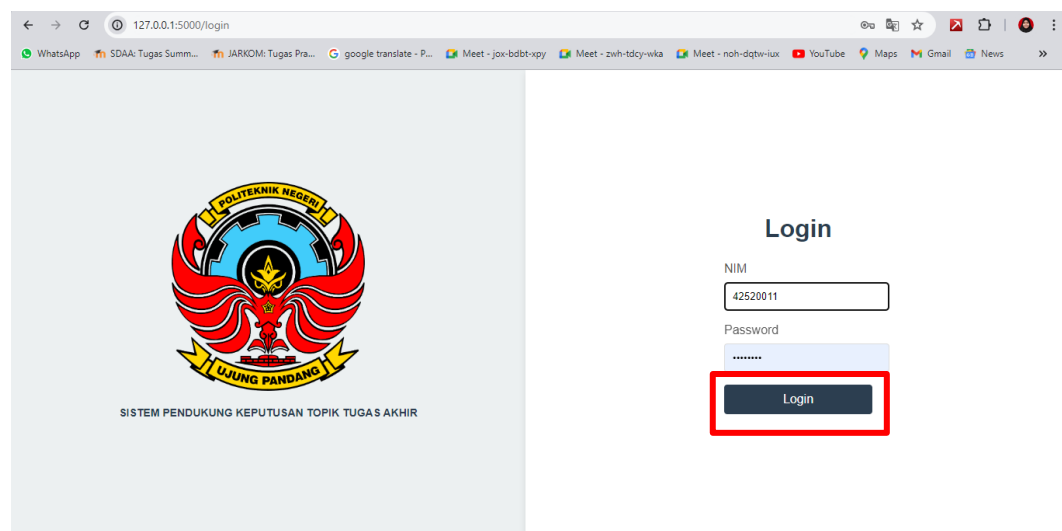
nilai akademik dan *input* minat bidang ilmu serta minat profil lulusan. Sistem kemudian akan memproses data tersebut sesuai dengan model yang telah dibuat hingga tahap prediksi. *Output* dari fungsi prediksi ini akan ditampilkan pada aplikasi *website* sebagai hasil klasifikasi pada sistem pendukung keputusan topik tugas akhir mahasiswa.

4.2.5 Implementasi *User Interface* (UI)

Desain interface ini berfokus pada bagaimana cara user menggunakannya. Kecerdasan Model yang telah dibuat dengan menggunakan Algoritma *Random Forest* akan di *convert* kedalam bentuk *pickle*. Implementasi ini berisi mulai dari proses input gambar pada aplikasi sampai dengan proses prediksi yang akan dimunculkan pada aplikasi yang telah dibuat.

1. Halaman *Login*

Halaman *login* ini merupakan halaman agar user dapat masuk ke aplikasi Sistem Pendukung Keputusan Topik Tugas Akhir. Di mana user akan memasukkan NIM dan password yang telah terdaftar di database. Halaman *login* dapat dilihat pada Gambar 4.7

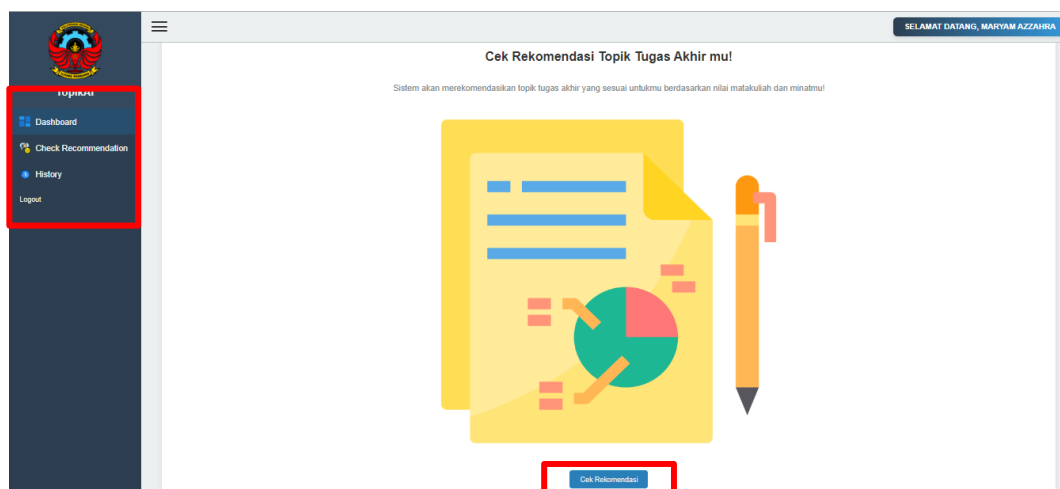


Gambar 4. 7 Halaman *Login*

2. Halaman utama atau *Dashboard*

Pada *website* ini dan halaman-halaman lainnya terdapat *sidebar* yang terdapat tombol untuk kehalaman *dashboard*, *check recommendation*, *history*, dan *logout*.

Halaman utama atau *dashboard* yang ditampilkan setelah *user* berhasil *login* ke *website*. Di mana ada *button* “Cek Rekomendasi” untuk menuju ke halaman di mana *user* dapat melakukan cek rekomendasi topik tugas akhirnya. Halaman ini dapat dilihat pada Gambar 4.8



Gambar 4. 8 Halaman *Dashboard*

3. Halaman melihat nilai serta input minat bidang ilmu dan profile lulusan

Halaman ini menampilkan nilai mahasiswa untuk 30 matakuliah dapat dilihat pada Gambar 4.9. Dan untuk Gambar 4.10 di mana mahasiswa dapat menginput minat bidang ilmu dan minat *profile* lulusan untuk di prediksi. Dan *button* “*Check Recommendation*” untuk melihat hasil rekomendasi topik tugas akhir.

Gambar 4. 9 Halaman Tampilan Nilai

Gambar 4. 10 Form Input Minat

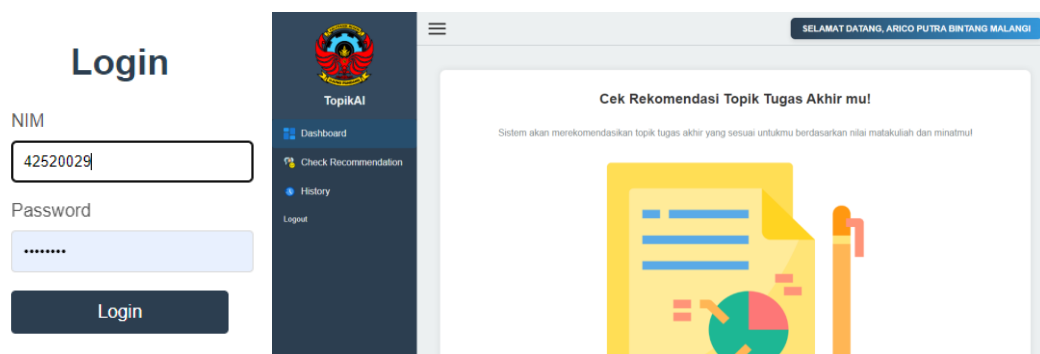
4. Halaman hasil klasifikasi

Halaman ini merupakan halaman di mana user akan melihat hasil prediksi model yang menjadi rekomendasi profile lulusan dan topik tugas akhir yang sesuai dengan bidang ilmu, di mana dengan mengetahui rekomendasi keduanya diharapkan mahasiswa dapat lebih mudah memutuskan topik tugas akhir yang cocok. Hasil rekomendasi topik tugas akhir ini berdasarkan nilai matakuliah, untuk minat bidang ilmu dan profile lulusan menjadi point penambah persentase ketika rekomendasi berdasarkan nilai sesuai dengan minatnya. Skenario implementasi aplikasi pada *user* dapat dilihat pada pemaparan berikut :

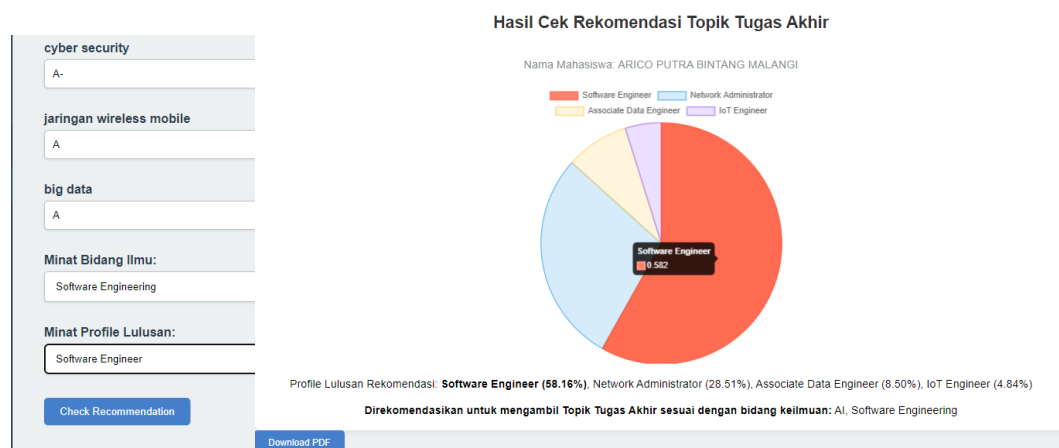
Pengguna 1: Arico Putra Bintang Malangi (42520029)

Arico membuka aplikasi dan memasukkan kredensial login untuk mengakses sistem dapat dilihat pada Gambar 4.11. Kemudian pada *button* “*Check Recommendation*” data nilai akademik Arico ditampilkan dan pada form paling

bawah, Arico menginputkan data minat bidang ilmu nya yaitu *Software Engineering* dan minat profile lulusannya *Software Engineer*. Klik button “*Check Recommendation*” pada halaman ini dan aplikasi menampilkan profile lulusan yang direkomendasikan untuk Arico yaitu *Software Engineer* dan topik tugas akhir sesuai dengan bidang ilmu terkait *AI* dan *Software Engineering*. Arico mendapatkan rekomendasi topik tugas akhir yang sesuai dengan minat dan kemampuan akademiknya berdasarkan nilai matakuliah tertinggi dan minatnya. Dapat dilihat pada Gambar 4.12.



Gambar 4. 11 Login Akun Arico

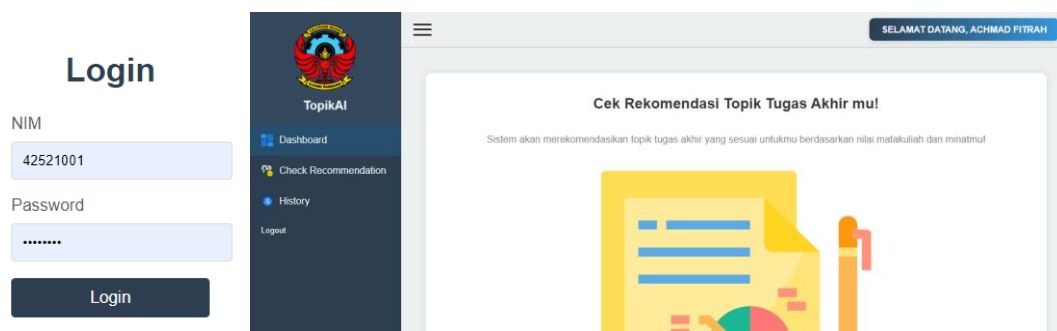


Gambar 4. 12 Form Nilai dan Minat serta Hasil Rekomendasi

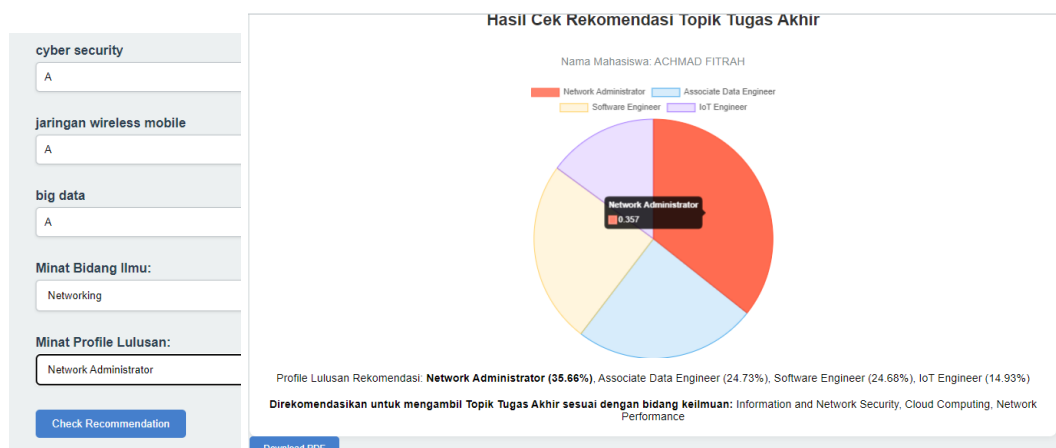
Pengguna 2: Achmad Fitrah (42521001)

Fitrah membuka aplikasi dan memasukkan kredensial login untuk mengakses sistem dapat dilihat pada Gambar 4.13. Kemudian pada button “*Check Recommendation*” data nilai akademik Fitrah ditampilkan dan pada form paling bawah, Fitrah menginputkan data minat bidang ilmu nya yaitu *Networking* dan minat profile lulusannya *Network Administrator*. Klik button “*Check*

Recommendation” pada halaman ini dan aplikasi menampilkan profile lulusan yang direkomendasikan untuk Fitrah yaitu *Network Administrator* dan topik tugas akhir sesuai dengan bidang ilmu terkait *Information and Network Security, Cloud Computing, Network Performance*. Fitrah mendapatkan rekomendasi topik tugas akhir yang sesuai dengan minat dan kemampuan akademiknya berdasarkan nilai matakuliah tertinggi dan minatnya. Dapat dilihat pada Gambar 4.14.



Gambar 4. 13 Login Akun Fitrah



Gambar 4. 14 Form Nilai dan Minat serta Hasil Rekomendasi

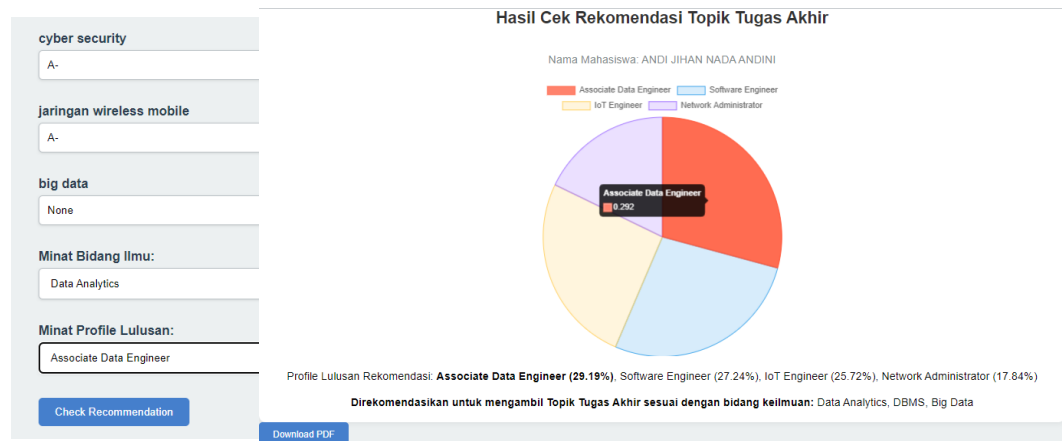
Pengguna 3: Andi Jihan Nada Andini (42521004)

Jihan membuka aplikasi dan memasukkan kredensial login untuk mengakses sistem dapat dilihat pada Gambar 4.15. Kemudian pada *button* “*Check Recommendation*” data nilai akademik Jihan ditampilkan dan pada form paling bawah, Jihan menginputkan data minat bidang ilmu nya yaitu *Data Analytics* dan minat profile lulusannya *Associate Data Engineer*. Klik *button* “*Check Recommendation*” pada halaman ini dan aplikasi menampilkan profile lulusan yang direkomendasikan untuk Jihan yaitu *Associate Data Engineer* dan topik tugas akhir sesuai dengan bidang ilmu terkait *Data Analytics, DBMS, Big Data*. Jihan mendapatkan

rekomendasi topik tugas akhir yang sesuai dengan minat dan kemampuan akademiknya berdasarkan nilai mata kuliah tertinggi dan minatnya. Dapat dilihat pada Gambar 4.16.



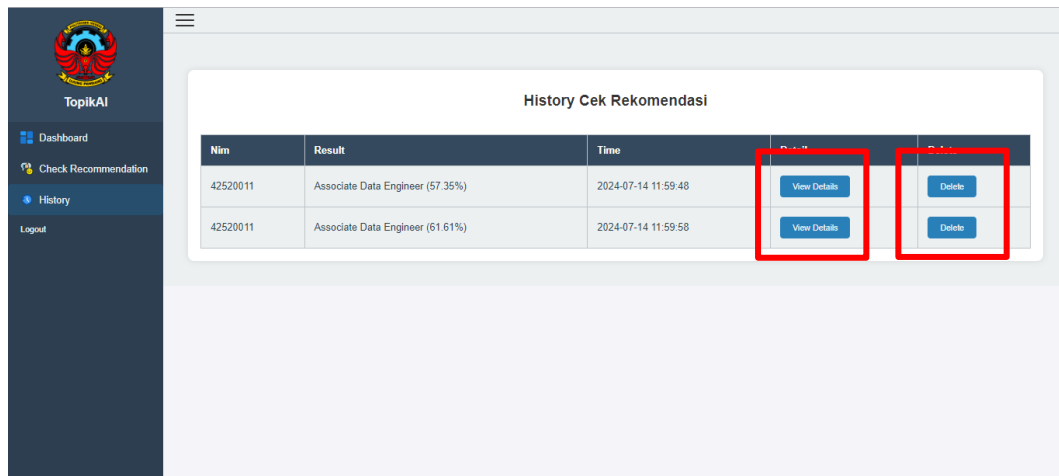
Gambar 4. 15 Login Akun Jihan



Gambar 4. 16 Form Nilai dan Minat serta Hasil Rekomendasi

5. Halaman *history*

Halaman ini menampilkan riwayat mahasiswa dalam melakukan cek rekomendasi. Ini diperlukan agar mahasiswa dapat melihat ulang hasil rekomendasinya yang sebelumnya, tanpa harus melakukan cek rekomendasi ulang. Seperti di tampilkan pada Gambar 4.17



Gambar 4. 17 Halaman *History*

4.3 Pengujian Sistem

Tujuan utama dari pengujian sistem ini adalah untuk memastikan bahwa sistem yang dibangun dapat memberikan rekomendasi topik tugas akhir yang akurat dan sesuai dengan preferensi mahasiswa, efisien dalam prosesnya, dan mudah digunakan oleh pengguna.

4.3.1 Hasil Pengujian Model *Random Forest*

Fitting 5 folds for each of 50 candidates, totalling 250 fits

Best parameters found: {'bootstrap': False, 'max_depth': 8, 'min_samples_leaf': 5, 'min_samples_split': 7, 'n_estimators': 348}

Cross-validated scores: [0.53846154 0.47058824 0.58823529 0.68627451 0.78431373]

Average score: 0.6135746606334841

1. *Fitting 5 Folds for Each of 50 Candidates, Totaling 250 Fits*

Proses fitting melibatkan pencarian hyperparameter terbaik untuk Model *Random Forest* menggunakan *RandomizedSearchCV*. Dalam proses ini:

- 1) *5 folds*: Data dibagi menjadi 5 subset yang digunakan untuk validasi silang (*cross-validation*).
- 2) *50 candidates*: Sebanyak 50 kombinasi *hyperparameter* yang berbeda dicoba.
- 3) *250 fits*: Total 250 model dilatih dan divalidasi (5 folds x 50 kombinasi).

RandomizedSearchCV mencari kombinasi terbaik dari *hyperparameter* yang ditentukan dalam *param_dist* dengan mencoba 50 kombinasi acak dari kemungkinan nilai *hyperparameter*.

2. *Best Parameters Found*

Hasil dari pencarian *hyperparameter* menunjukkan bahwa kombinasi terbaik yang ditemukan adalah:

- 1) '*bootstrap*': *False*: Mengindikasikan bahwa pengambilan sampel dengan penggantian (bootstrap sampling) tidak digunakan.
- 2) '*max_depth*': 8: Kedalaman maksimum setiap pohon keputusan adalah 8 tingkat.
- 3) '*min_samples_leaf*': 5: Setiap daun pohon keputusan harus memiliki setidaknya 5 sampel.
- 4) '*min_samples_split*': 7: Minimal 7 sampel diperlukan untuk membagi *node*.
- 5) '*n_estimators*': 348: Model terdiri dari 348 pohon keputusan.

3. *Cross-Validated Scores*

Hasil validasi silang (*cross-validation*) pada 5 *fold* menunjukkan akurasi berikut untuk setiap *fold*:

- 1) *Fold 1*: 0.53846154
- 2) *Fold 2*: 0.47058824
- 3) *Fold 3*: 0.58823529
- 4) *Fold 4*: 0.68627451
- 5) *Fold 5*: 0.78431373

Rata-rata skor dari validasi silang adalah **0.6136**, yang memberikan gambaran umum tentang kinerja model pada data yang tidak terlihat selama pelatihan.

4.3.2 Pengujian *Black Box*

Pengujian *Black Box* atau fungsional adalah pengujian yang dilakukan pada aplikasi untuk memeriksa respons *website* apakah berjalan sesuai dengan fungsinya atau tidak, sehingga kesimpulan dapat ditarik berdasarkan berbagai kondisi masukan.

Tabel 4. 3 Pengujian halaman *login*

Skenario	Yang Diharapkan	Pengamatan	Hasil	Rujukan
Terdapat <i>Button</i> " <i>Login</i> "	<i>Button</i> " <i>Login</i> " akan merujuk ke halaman utama setelah login berhasil	Setelah login, pengguna diarahkan ke halaman <i>dashboard</i>	Berhasil	Gambar 4.7

Tabel 4. 4 Pengujian halaman utama

Skenario	Yang Diharapkan	Pengamatan	Hasil	Rujukan
Terdapat <i>form</i> yang menampilkan nilai mahasiswa	<i>Form</i> tampil dengan <i>field</i> yang sesuai dengan nilai mahasiswa	<i>Field</i> yang menampilkan nilai mahasiswa tampil dengan benar	Berhasil	Gambar 4.9
Terdapat <i>form</i> untuk memasukkan minat bidang keilmuan	<i>Form</i> tampil dengan <i>field</i> yang sesuai untuk input minat bidang keilmuan	<i>Field</i> untuk input minat bidang keilmuan tampil dengan benar	Berhasil	Gambar 4.10
Terdapat <i>form</i> untuk memasukkan minat profil lulusan	<i>Form</i> tampil dengan <i>field</i> yang sesuai untuk input minat profil lulusan	<i>Field</i> untuk input minat profil lulusan tampil dengan benar	Berhasil	Gambar 4.10
<i>Button</i> " <i>Check Recommendation</i> "	Setelah klik, data disimpan dan pengguna diarahkan ke halaman hasil rekomendasi	Data berhasil disimpan dan pengguna diarahkan ke halaman hasil rekomendasi	Berhasil	Gambar 4.10

Tabel 4. 5 Pengujian halaman klasifikasi

Skenario	Yang Diharapkan	Pengamatan	Hasil	Rujukan
Tampilkan hasil rekomendasi	Halaman menampilkan informasi mengenai hasil rekomendasi dalam bentuk pie chart	Informasi mengenai hasil rekomendasi ditampilkan dengan jelas dan mudah dipahami oleh pengguna. Pie chart menunjukkan	Berhasil	Gambar 4.11- Gambar 4.16

		rekomendasi profil lulusan.		
Terdapat informasi detail hasil rekomendasi	Halaman menampilkan informasi detail mengenai persentase profile lulusan dan topik tugas akhir	Informasi detail mengenai hasil rekomendasi profile lulusan dan topik tugas akhir berdasarkan bidang keilmuan dengan jelas sesuai kemampuan akademik dan minat mahasiswa	Berhasil	Gambar 4.11- Gambar 4.16

Tabel 4. 6 Pengujian halaman *history*

Skenario	Yang Diharapkan	Pengamatan	Hasil	Rujukan
<i>Button "History"</i>	Setelah mengklik tombol " <i>History</i> ", pengguna diarahkan ke halaman riwayat.	Pengguna diarahkan ke halaman riwayat yang menampilkan catatan aktivitas sebelumnya yaitu NIM, result, time, serta action " <i>views details</i> dan <i>delete</i> ".	Berhasil	Gambar 4.12
<i>Button "View Details"</i>	Setelah mengklik tombol ini, akan di tampilkan pop up riwayat pengecekan yang lebih detail	Pengguna dapat melihat riwayat pengecekan yaitu nilai, minat bidang ilmu yang dipilih, minat profile lulusan yang dipilih, dan hasil prediksinya.	Berhasil	Gambar 4.12
<i>Button "Delete"</i>	Setelah mengklik tombol ini, akan di tampilkan alert bahwa riwayat berhasil di hapus	Data yang dipilih akan dihapus dari list riwayat pengecekan	Berhasil	Gambar 4.12

Tabel 4. 7 Pengujian fungsi *logout*

Skenario	Yang Diharapkan	Pengamatan	Hasil	Rujukan
<i>Button “Logout”</i>	Setelah mengklik tombol “Logout”, pengguna keluar dari sistem dan diarahkan ke halaman login.	Pengguna keluar dari sistem dan diarahkan ke halaman login.	Berhasil	Gambar 4.8

Berdasarkan pengujian di atas, dapat disimpulkan bahwa pengujian *Black Box* pada *website* berjalan dengan baik dan sesuai dengan fungsi yang diharapkan. Seluruh skenario yang diuji berhasil dengan hasil yang sesuai harapan. Berdasarkan hasil tersebut, pengujian lanjutan terhadap pengguna akan dilakukan, dan hasilnya akan dijelaskan pada bagian pengujian hasil kuesioner.

4.3.3 Pengujian Hasil Kuesioner

Pengujian ini bertujuan untuk mengukur kepuasan pengguna terhadap sistem yang telah dibuat. Pengujian kuesioner menggunakan model TAM (*Technology Acceptance Model*), yang terdiri dari 9 pertanyaan untuk mengumpulkan pendapat para responden mengenai penggunaan aplikasi tersebut. Pengujian ini melibatkan 47 responden yang merupakan mahasiswa Teknik Komputer dan Jaringan Politeknik Negeri Ujung Pandang angkatan 2020 dan 2021, Langkah-langkah analisis skala *Likert* adalah sebagai berikut:

4.3.3.1 Pengumpulan Data Kuesioner

Dibawah ini merupakan hasil kuesioner yang telah diubah ke dalam bentuk tabulasi data. Dapat dilihat pada tabel 4.8

Tabel 4. 8 Tabulasi data kuesioner

No. Responden	Manfaat				Kemudahan						Kepuasan	
	X1	X1	X1	Total	X2	X2	X2	X2	X2	Total	X3	Total
	1	1	1	X1	2	2	2	2	2	X2	3	X3
1	4	4	3	11	3	3	3	3	3	15	3	3
2	3	3	2	8	4	3	3	2	3	15	3	3
3	4	4	4	12	3	4	3	3	4	17	4	4
4	3	2	3	8	3	2	3	3	2	13	3	3
5	4	4	4	12	4	4	4	4	4	20	4	4

6	3	3	3	9	3	3	3	3	3	15	3	3
7	3	3	3	9	2	3	3	3	2	13	3	3
8	3	3	3	9	4	4	4	4	4	20	4	4
9	3	3	3	9	3	3	3	3	3	15	3	3
10	3	3	3	9	3	3	3	3	3	15	3	3
11	4	4	4	12	4	3	4	4	4	19	4	4
12	4	4	4	12	4	4	4	3	4	19	4	4
13	3	3	3	9	3	3	3	3	3	15	3	3
14	4	4	4	12	4	4	4	4	4	20	4	4
15	3	3	3	9	4	3	3	3	3	16	3	3
16	4	3	4	11	4	4	3	4	4	19	4	4
17	3	3	3	9	3	3	3	3	3	15	3	3
18	4	4	4	12	4	4	4	4	4	20	4	4
19	3	3	3	9	2	2	3	3	2	12	3	3
20	4	4	4	12	4	4	4	4	4	20	4	4
21	3	4	3	10	3	3	3	3	3	15	3	3
22	3	3	4	10	3	3	3	4	3	16	3	3
23	3	3	3	9	4	4	4	3	4	19	3	3
24	3	3	3	9	3	3	4	4	4	18	4	4
25	3	3	4	10	3	3	3	3	3	15	3	3
26	4	4	3	11	3	3	4	4	3	17	3	3
27	3	3	3	9	3	3	3	3	3	15	3	3
28	3	4	3	10	4	4	4	4	4	20	4	4
29	4	4	4	12	4	4	4	4	4	20	4	4
30	3	3	3	9	3	3	3	3	3	15	3	3
31	4	4	4	12	4	4	4	4	4	20	4	4
32	4	4	4	12	4	4	4	4	4	20	4	4
33	3	3	3	9	3	4	4	4	4	19	4	4
34	3	2	2	7	3	2	3	3	4	15	3	3
35	3	3	3	9	3	3	3	3	3	15	3	3
36	3	3	3	9	3	3	3	3	3	15	3	3
37	4	3	3	10	3	4	3	4	3	17	4	4
38	3	3	3	9	4	3	4	3	3	17	3	3
39	3	3	3	9	3	3	3	3	3	15	3	3
40	3	3	3	9	4	3	3	3	3	16	3	3
41	3	3	3	9	3	3	4	4	3	17	4	4
42	4	4	3	11	4	3	4	3	4	18	3	3
43	3	3	3	9	3	3	3	3	3	15	3	3
44	3	3	3	9	3	3	3	3	3	15	3	3

45	3	3	3	9	3	3	4	3	3	16	3	3
46	3	3	3	9	3	3	3	3	3	15	3	3
47	3	4	4	11	4	4	3	3	3	17	3	3

4.3.3.2 Menjumlahkan Seluruh Data

Setelah data terkumpul, langkah berikutnya adalah mengelompokkan data berdasarkan jenis jawaban yang diperoleh. Misalnya, jawaban "sangat setuju" dikelompokkan bersama dengan responden yang juga menjawab "sangat setuju", dan begitu seterusnya. Setelah itu, jumlahkan masing-masing jawaban dari responden seperti yang ditunjukkan pada Tabel 4.9 sampai Tabel 4.11

Tabel 4. 9 Jumlah aspek data manfaat

No	Pernyataan Manfaat	SS	S	TS	STS	SKOR
1	Aplikasi ini memudahkan untuk menentukan topik tugas akhir.	15	32	0	0	47
2	Hasil rekomendasi yang ditampilkan oleh aplikasi ini sangat membantu dalam memilih topik tugas akhir yang sesuai.	16	29	2	0	47
3	Aplikasi ini membantu membuat keputusan yang lebih baik tentang topik tugas akhir mahasiswa.	14	31	2	0	47

Tabel 4. 10 Aspek data kemudahan

No	Pernyataan Kemudahan	SS	S	TS	STS	SKOR
1	Aplikasi ini mudah digunakan bahkan untuk pengguna baru	19	26	2	0	47
2	Antarmuka aplikasi ini intuitif dan mudah dipahami.	16	28	3	0	47
3	Proses login ke dalam aplikasi ini cepat dan tidak memakan waktu.	19	28	0	0	47
4	Tidak menemui kesulitan di halaman tampilan nilai dan menginput minat mahasiswa di halaman utama.	17	29	1	0	47
5	Menavigasi dari halaman utama ke halaman hasil rekomendasi sangat mudah dan cepat.	18	26	3	0	47

Tabel 4. 11 Aspek data kepuasan user

No	Pernyataan Kepuasan	SS	S	TS	STS	SKOR
1	Secara keseluruhan, pengguna puas dengan kinerja aplikasi ini.	17	30	0	0	47

Keterangan:

STS : Sangat Tidak Setuju

TS : Tidak Setuju

S : Setuju

SS : Sangat Setuju

SKOR : Total Responden

4.3.3.3 Pemberian Bobot

Setelah data berhasil dijumlahkan, langkah selanjutnya adalah memberikan bobot pada setiap jawaban dari responden. Tanpa pemberian bobot ini, data tidak dapat diolah lebih lanjut. Pembobotan dilakukan sebagai berikut:

Sangat Tidak Setuju	1
Tidak Setuju	2
Setuju	3
Sangat Setuju	4

Tabel 4. 12 Skor aspek manfaat

No	Pernyataan Manfaat	SS	S	TS	STS	SKOR
1	Aplikasi ini memudahkan untuk menentukan topik tugas akhir.	60	96	0	0	156
2	Hasil rekomendasi yang ditampilkan oleh aplikasi ini sangat membantu dalam memilih topik tugas akhir yang sesuai.	64	87	4	0	155
3	Aplikasi ini membantu membuat keputusan yang lebih baik tentang topik tugas akhir mahasiswa.	56	93	4	0	153

Tabel 4. 13 Skor aspek kemudahan

No	Pernyataan Kemudahan	SS	S	TS	STS	SKOR
1	Aplikasi ini mudah digunakan bahkan untuk pengguna baru	76	78	4	0	158
2	Antarmuka aplikasi ini intuitif dan mudah dipahami.	64	84	6	0	154
3	Proses login ke dalam aplikasi ini cepat dan tidak memakan waktu.	76	84	0	0	160
4	Tidak menemui kesulitan di halaman tampilan nilai dan menginput minat mahasiswa di halaman utama.	68	87	2	0	157
5	Menavigasi dari halaman utama ke halaman hasil rekomendasi sangat mudah dan cepat.	72	78	6	0	156

Tabel 4. 14 Kepuasan user

No	Pernyataan Kepuasan	SS	S	TS	STS	SKOR
1	Secara keseluruhan, pengguna puas dengan kinerja aplikasi ini.	68	90	0	0	158

Untuk mengetahui skor maksimum rumus yang digunakan adalah jumlah responden x skor tertinggi. Sedangkan untuk mengetahui skor minimum, rumus yang digunakan adalah jumlah responden x skor terendah.

$$\text{Skor Maksimum} \quad 47 \times 4 = 188$$

$$\text{Skor Minimum} \quad 47 \times 1 = 47$$

Rumus interval

$$I = \frac{188}{\text{Jumlah Skor}}$$

$$I = \frac{188}{4} = 47$$

Maka $I = 47$

Berikut merupakan interval dari jarak terendah 0% hingga 100%. Dibawah ini adalah kriteria interpretasi skor berdasarkan intervalnya

- 1) Angka 0% - 24,99% = Sangat tidak baik

- 2) Angka 25% - 49,99% = Tidak baik
- 3) Angka 50% - 74,99% = Baik
- 4) Angka 75% - 100% = Sangat baik

4.3.3.4 Perhitungan Persentase

Tahap terakhir adalah menghitung persentase. Rumus untuk mengetahui indeks dalam bentuk persentase adalah total skor dibagi total skor maksimum kemudian dikali 100

$$\text{Rumus Index \%} = \frac{\text{Total Skor}}{\text{Skor Maksimum}} \times 100 \quad (4.1)$$

Adapun hasil persentase pengujian kuesioner bisa dilihat pada tabel. Tabel 4.15 sampai 4.17 di mana hasil tersebut memperlihatkan perhitungan kuesioner dengan interval dari jarak 0% - 100%.

Tabel 4. 15 Pengujian kuesioner manfaat

No	Pernyataan Manfaat	SS %	S %	TS %	STS %	PERSENTASE %
1	Aplikasi ini memudahkan untuk menentukan topik tugas akhir.	32	51	0	0	82.9787234
2	Hasil rekomendasi yang ditampilkan oleh aplikasi ini sangat membantu dalam memilih topik tugas akhir yang sesuai.	34	46	2.13	0	82.44680851
3	Aplikasi ini membantu membuat keputusan yang lebih baik tentang topik tugas akhir mahasiswa.	30	49	2.13	0	81.38297872

Dari nilai persentase skor pada tabel 4.15 diatas, diketahui bahwa responden dalam kategori sangat baik. Dengan demikian dapat disimpulkan bahwa para mahasiswa Teknik Komputer dan Jaringan menilai “sangat baik” dari aspek manfaat Sistem Pendukung Keputusan Topik Tugas Akhir Mahasiswa yang telah dibuat.

Tabel 4. 16 Pengujian kuesioner kemudahan

No	Pernyataan Kemudahan	SS %	S %	TS %	STS %	PERSENTASE %
1	Aplikasi ini mudah digunakan bahkan untuk pengguna baru	40	41	2.13	0	84.04255319

2	Antarmuka aplikasi ini intuitif dan mudah dipahami.	34	45	3.19	0	81.91489362
3	Proses login ke dalam aplikasi ini cepat dan tidak memakan waktu.	40	45	0	0	85.10638298
4	Tidak menemui kesulitan di halaman tampilan nilai dan menginput minat mahasiswa di halaman utama.	36	46	1.06	0	83.5106383
5	Menavigasi dari halaman utama ke halaman hasil rekomendasi sangat mudah dan cepat.	38	41	3.19	0	82.9787234

Dari nilai persentase skor pada tabel 4.16 diatas, diketahui bahwa responden dalam kategori sangat baik. Dengan demikian dapat disimpulkan bahwa para mahasiswa Teknik Komputer dan Jaringan menilai “sangat baik” dari aspek kemudahan Sistem Pendukung Keputusan Topik Tugas Akhir Mahasiswa yang telah dibuat.

Tabel 4. 17 Pengujian kuesioner kepuasan user

No	Pernyataan Kepuasan	SS %	S %	TS %	STS %	PERSENTASE %
1	Secara keseluruhan, pengguna puas dengan kinerja aplikasi ini.	36	48	0	0	84.04255319

Dari nilai persentase skor pada tabel 4.17 diatas, diketahui bahwa responden dalam kategori sangat baik. Dengan demikian dapat disimpulkan bahwa para mahasiswa Teknik Komputer dan Jaringan menilai “sangat baik” dari aspek kepuasan user Sistem Pendukung Keputusan Topik Tugas Akhir Mahasiswa yang telah dibuat.

4.3.4 Validasi Efektivitas Sistem Pendukung Keputusan Topik Tugas Akhir

Untuk menguji efektivitas sistem rekomendasi topik tugas akhir yang telah di buat, dilakukan pengujian terhadap alumni dari program studi tempat penelitian ini dilakukan. Pengujian ini bertujuan untuk mengetahui persentase mahasiswa yang lulus tepat waktu dan memilih topik penelitian yang sesuai dengan hasil rekomendasi sistem.

4.3.4.1 Pengumpulan Data

Data dikumpulkan dari 55 alumni yang datanya tersedia dan status kelulusan tepat waktu diketahui. Data tersebut mencakup topik penelitian yang dipilih saat masih menyusun skripsi atau tugas akhir, juga dari hasil kuesioner/wawancara

4.3.4.2 Analisis Keselarasan

Topik penelitian yang dipilih alumni dibandingkan dengan topik yang direkomendasikan oleh sistem. Analisis dilakukan untuk menentukan keselarasan antara pilihan topik alumni dengan rekomendasi sistem.

4.3.4.3 Perhitungan Persentase

Persentase kesesuaian dihitung berdasarkan jumlah alumni yang topik penelitiannya sesuai dengan rekomendasi sistem dibagi dengan total alumni yang diketahui lulus tepat waktu, kemudian dikalikan 100%.

$$\text{Persentase kesesuaian} : \frac{\text{Jumlah alumni sesuai}}{\text{Jumlah total alumni}} \times 100\% \quad (4.2)$$

Dari 55 data alumni:

- 19 alumni (34.54%) memilih topik skripsi yang sesuai dengan rekomendasi sistem.
- 24 alumni (43.63%) memilih topik skripsi yang berada di urutan kedua dalam rekomendasi sistem.
- 12 alumni (21.81%) memilih topik skripsi yang berada di luar dua urutan teratas rekomendasi sistem.

Tabel 4. 18 Validasi Efektivitas

Status Kelulusan Tepat Waktu	Jumlah Alumni	Prediksi Topik Sesuai Rekomendasi (%)	Prediksi Topik di Urutan Kedua Rekomendasi (%)	Prediksi Topik di Luar Dua Urutan Teratas (%)
Ya	55	34.54%	43.63%	21.81%
Total		78.17%		21.81%

Hasil analisis menunjukkan bahwa dari 55 alumni yang lulus tepat waktu, 34.54% memilih topik skripsi yang sesuai dengan rekomendasi utama sistem, sementara 43.63% memilih topik skripsi yang berada di urutan kedua dari rekomendasi sistem. Ini berarti total 78.17% alumni yang lulus tepat waktu memilih topik skripsi yang berada di dua urutan teratas rekomendasi sistem. Sebaliknya, hanya 21.81% alumni yang lulus tepat waktu memilih topik skripsi yang berada di luar dua urutan teratas rekomendasi sistem. Hasil ini mengindikasikan bahwa sistem memiliki tingkat akurasi yang tinggi dalam memberikan rekomendasi topik penelitian yang relevan dan bermanfaat bagi mahasiswa.

BAB V PENUTUP

5.1 Kesimpulan

Model *Random Forest* yang telah dikembangkan dan dilatih menunjukkan hasil yang memuaskan dalam memprediksi topik tugas akhir mahasiswa. Akurasi model yang diperoleh dari rata rata *Cross-validation* mencapai 61.36%. Implementasi dan evaluasi model pada sistem pendukung keputusan berbasis website mengonfirmasi bahwa model dapat membantu mahasiswa dalam menentukan topik tugas akhir yang sesuai berdasarkan kemampuan akademik dan minatnya. Fitur-fitur utama seperti nilai akademik, minat bidang ilmu dan minat profil lulusan telah berhasil diintegrasikan, memberikan kontribusi yang signifikan dalam peningkatan akurasi model.

Selain itu, hasil pengamatan setelah melakukan uji coba langsung data mahasiswa pada sistem menunjukkan bahwa data "minat bidang ilmu" dan "minat profil lulusan" berpengaruh dalam rekomendasi. Jika kedua data ini sesuai dengan rekomendasi algoritma, persentase pada diagram *pie* juga meningkat sekitar 2-3% pada rekomendasi profil lulusan pertama.

Aplikasi ini tidak hanya meningkatkan akurasi dalam pemilihan topik tugas akhir, tetapi juga mengefisienkan waktu mahasiswa dalam mencari dan mengevaluasi topik yang sesuai. Dengan sistem yang terintegrasi dan otomatis, mahasiswa dapat dengan cepat mendapatkan rekomendasi yang relevan tanpa harus melalui proses pencarian yang panjang dan memakan waktu.

Meskipun hasil yang diperoleh cukup baik, masih terdapat ruang untuk perbaikan. Validasi lebih lanjut dengan menggunakan dataset yang lebih besar dan bervariasi perlu dilakukan untuk memastikan bahwa model tidak mengalami overfitting dan memiliki kemampuan generalisasi yang baik. Hal ini penting agar sistem dapat digunakan secara efektif dalam berbagai situasi dan kondisi.

5.2 Saran

Saran untuk pengembangan lebih lanjut dari penelitian ini adalah sebagai berikut:

1. Meningkatkan Akurasi Model: Diperlukan upaya untuk meningkatkan akurasi model yang telah dikembangkan. Hal ini dapat dilakukan dengan menggunakan teknik pemrosesan data yang lebih canggih, eksplorasi model *machine learning* lainnya, atau melakukan tuning *hyperparameter* yang lebih mendalam.
2. Adaptasi terhadap Perubahan Kurikulum: Mengingat bahwa kurikulum berubah setiap 4 tahun, di mana mata kuliah dan profil lulusan biasanya mengalami perubahan seperti penambahan dan pengurangan, perlu dikembangkan sistem yang memudahkan pengembang di masa mendatang. Sistem ini harus dapat menyesuaikan perubahan kurikulum tanpa harus melakukan pengelompokkan ulang pada data baru untuk mendapatkan label maupun melatih model kembali menggunakan algoritma.
3. Memperbanyak Data: Untuk meningkatkan performa dan generalisasi model, perlu diperbanyak data yang digunakan dalam pelatihan. Selain itu, pengembangan lebih lanjut dapat mempertimbangkan minat mahasiswa secara lebih mendetail dan tidak hanya berfokus pada nilai tertinggi yang diperoleh.
4. Meningkatkan Antarmuka Pengguna: Tampilan aplikasi yang telah dibuat perlu ditingkatkan agar lebih *user-friendly* dan menarik. Hal ini penting agar pengguna merasa nyaman dan mudah dalam menggunakan aplikasi, yang pada akhirnya akan meningkatkan kepuasan pengguna terhadap sistem yang dikembangkan.
5. Penambahan Interface untuk Penginputan Nilai: Perlu ditambahkan antarmuka yang memungkinkan penginputan nilai secara langsung oleh admin. Ini akan mempermudah proses *update* data dan memastikan bahwa data yang digunakan dalam model selalu *up-to-date* dan akurat, yang pada akhirnya akan meningkatkan akurasi dan relevansi rekomendasi topik tugas akhir.

DAFTAR PUSTAKA

- Agnes, T., Sinaga, V., Debra, L. B., Ariva, J. T., Aditya, R., Gunawan, V. F., & Maryamah, M. (2023). *Sistem Rekomendasi Pencarian Indekos di Surabaya Menggunakan Random Forest*. 2023(Senada), 172–177.
- Aini, N., Arif, M., Agustin, I. T., & Toyibah, Z. B. (2024). Implementasi Algoritma Random Forest untuk Klasifikasi Bidang MSIB di Prodi Pendidikan Informatika. *Jurnal Informatika*, 11(1), 11–16. <https://doi.org/10.31294/inf.v11i1.20637>
- Darmawan, A., Yudhisari, I., Anwari, A., & Makruf, M. (2023). Pola Prediksi Kelulusan Siswa Madrasah Aliyah Swasta dengan Support Vector Machine dan Random Forest. *Jurnal Minfo Polgan*, 12(1), 387–400. <https://doi.org/10.33395/jmp.v12i1.12388>
- Fahmi, N. R. I., Prihandoko, A. C., & Retnani, W. E. Y. (2021). Implementasi Metode Fuzzy AHP pada Sistem Penunjang Keputusan Penentuan Topik Skripsi (Studi Kasus : Program Studi Sistem Informasi Universitas Jember) (Implementation of Fuzzy AHP Method in Decission Support System Determination of Topic Thesis) (Case S. *Berkala Sainstek*, 2, 76–81.
- Faidhani, F., Tursina, T., & Sukamto, A. S. (2021). Sistem Pendukung Keputusan Penentu Bidang Keahlian Mahasiswa Program Studi Informatika Universitas Tanjungpura dengan Metode ELECTRE. *Jurnal Sistem Dan Teknologi Informasi (Justin)*, 9(1), 41. <https://doi.org/10.26418/justin.v9i1.31357>
- Ghanbari, E., & Najafzadeh, S. (2020). Implementasi Machine Learning Prediksi Harga Sewa Apartemen Menggunakan Algoritma Random Forest Melalui Framework Website Flask Python (Studi Kasus: Apartemen Di Dki Jakarta Pada Website Mamikos.Com). *Machine Learning and Big Data: Concepts, Algorithms, Tools and Applications*, 155–207. <https://doi.org/10.1002/9781119654834.ch7>
- Haviluddin, H., Patandianan, S. J., Putra, G. M., Puspitasari, N., & Pakpahan, H. S. (2021). Implementasi Metode K-Means Untuk Pengelompokkan Rekomendasi Tugas Akhir. *Informatika Mulawarman : Jurnal Ilmiah Ilmu*

- Komputer*, 16(1), 13. <https://doi.org/10.30872/jim.v16i1.5182>
- Ijudin, A., & Saifudin, A. (2020). Pengujian Black Box pada Aplikasi Berita Online dengan Menggunakan Metode Boundary Value Analysis. *Jurnal Informatika Universitas Pamulang*, 5(1), 8. <https://doi.org/10.32493/informatika.v5i1.3717>
- Maryadi, R. (2019). *Analisis Dan Perancangan Sistem Rekomendasi Topik Tugas Akhir Menggunakan Metode Hybrid And Ensemble-Based Recommender Systems*.
- Mu'tashim, M. L., & Zaidiah, A. (2023). Klasifikasi Ketepatan Lama Studi Mahasiswa Dengan Algoritma Random Forest Dan Gradient Boosting (Studi Kasus Fakultas Ilmu Komputer Universitas Pembangunan Nasional Veteran Jakarta). *Seminar Nasional Mahasiswa Bidang Ilmu Komputer Dan Aplikasinya (SENAMIKA)*, 4(1), 155–166.
- Muttaqin, M. R., & Defriani, M. (2020). Algoritma K-Means untuk Pengelompokan Topik Skripsi Mahasiswa. *ILKOM Jurnal Ilmiah*, 12(2), 121–129. <https://doi.org/10.33096/ilkom.v12i2.542.121-129>
- Ningrum, A. S., & Utami, A. W. (2023). Sistem Deteksi Tulisan Tangan Aksara Hangeul Satu Silabel Menggunakan Metode Convolutional Neural Network (CNN) Berbasis Framework Flask. *Jeisbi*, 04(04), 9–16.
- Pahlevi, O.-, Amrin, A.-, & Handrianto, Y.-. (2023). Implementasi Algoritma Klasifikasi Random Forest Untuk Penilaian Kelayakan Kredit. *Jurnal Infortech*, 5(1), 71–76. <https://doi.org/10.31294/infortech.v5i1.15829>
- Praniffa, A. C., Syahri, A., Sandes, F., Fariha, U., Giansyah, Q. A., & Hamzah, M. L. (2023). Pengujian Black Box Dan White Box Sistem Informasi Parkir Berbasis Web Black Box and White Box Testing of Web-Based Parking Information System. *Jurnal Testing Dan Implementasi Sistem Informasi*, 1(1), 1–16.
- Pratama, A. R., Rio Rizki Aryanto, & Lizda Iswari. (2021). Studi Komparasi Model Klasifikasi Berbasis Pembelajaran Mesin untuk Sistem Rekomendasi Program Studi. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 5(5), 853–862. <https://doi.org/10.29207/resti.v5i5.3392>

- Ramadani, R. P. (2021). *Klasifikasi Judul Tugas Akhir Prodi Teknik Informatika Menggunakan Metode Naïve Bayes Classifier*.
<https://www.academia.edu/download/94326268/392605122.pdf>
- Saepudin, A., Faqih, A., & Dwilestari, G. (2024). *Perbandingan Algoritma Klasifikasi Support Vector Machine , Random Forest dan Logistic Regression Pada Ulasan Shopee*. 18(1), 178–192.
- Salam, A., Albahri, F. P., & Fathurrahmad. (2022). Sistem Rekomendasi Tugas Akhir Mahasiswa pada AMIK Indonesia untuk Mendukung Merdeka Belajar-Kampus Merdeka Menggunakan Metode Collaborative Filtering (CF). *Jurnal JTik (Jurnal Teknologi Informasi Dan Komunikasi)*, 6(2), 281–288.
<https://doi.org/10.35870/jtik.v6i2.420>
- Theodorus, D., Defit, S., & Nurcahyo, G. W. (2021). Machine Learning Rekomendasi Produk dalam Penjualan Menggunakan Metode Item-Based Collaborative Filtering. *Jurnal Informasi Dan Teknologi*, 3, 202–208.
<https://doi.org/10.37034/jidt.v3i4.151>
- Triawan, A., & Lintang Melinda, D. (2020). Penerapan Metode Naïve Bayes Untuk Rekomendasi Topik Tugas Akhir Berdasarkan Daftar Hasil Studi Mahasiswa di Perguruan Tinggi. *Teknois : Jurnal Ilmiah Teknologi Informasi Dan Sains*, 10(2), 58–70. <https://doi.org/10.36350/jbs.v10i2.91>
- Wibowo Putri, A. A., & Susetyo, Y. A. (2022). Implementation of Flask for Stock Checking in Distribution Center & Store on Monitoring Stock Application in Pt. Xyz. *Jurnal Teknik Informatika (Jutif)*, 3(5), 1265–1274.
<https://doi.org/10.20884/1.jutif.2022.3.5.334>
- Yulindawati, Lailiyah, S., Yusnita, A., & Hafifah, .A. (2024). *Teknik Informatika STMIK Widya Cipta Dharma Samarinda Abstraksi Keywords : Pendahuluan Tinjauan Pustaka Metode Penelitian Hasil dan Pembahasan*. 5(2).
- Zeniarja, J., Salam, A., & Ma'ruf, F. A. (2022). Seleksi Fitur dan Perbandingan Algoritma Klasifikasi untuk Prediksi Kelulusan Mahasiswa. *Jurnal Rekayasa ElektriKa*, 18(2), 102–108. <https://doi.org/10.17529/jre.v18i2.24047>

L

A

M

P

I

R

A

N

Lampiran 1 code Data Dummy untuk data minat bidang ilmu dan minat profile lulusan

```
# Mengisi nilai kosong pada kolom minat dengan nilai acak dari
daftar yang diberikan
minat_bidang_ilmu_choices = ['Networking', 'Software
Engineering', 'Data Analytics', 'IoT (Internet of Things)',
'AI (Artificial Intelligence)', 'Cyber Security', 'DBMS / Big
Data']
minat_profile_lulusan_choices = ['Network Administrator',
'Software Engineer', 'Associate Data Engineer', 'IoT
Engineer']

# Mengisi nilai kosong pada kolom "Minat Bidang Ilmu" dengan
nilai acak dari daftar pilihan
empty_indices = data[data['Minat Bidang Ilmu'] == ''].index
data.loc[empty_indices, 'Minat Bidang Ilmu'] =
np.random.choice(minat_bidang_ilmu_choices,
size=len(empty_indices))

# Mengisi nilai kosong pada kolom "Minat Profile Lulusan"
dengan nilai acak dari daftar pilihan
empty_indices = data[data['Minat Profile Lulusan'] ==
''].index
data.loc[empty_indices, 'Minat Profile Lulusan'] =
np.random.choice(minat_profile_lulusan_choices,
size=len(empty_indices))

# Menyimpan dataset yang sudah diproses ke file CSV baru
data.to_csv('data_dummy_filled.csv', index=False)
```

Lampiran 2 Code Pengelompokkan untuk mendapatkan label

```
# Mengelompokkan mata kuliah ke dalam kategori
network_admin_courses = [
    'arsitektur_komputer', 'komunikasi_data',
    'bengkel_komputer_jaringan', 'jaringan_komputer',
    'teknologi_jaringan_komputer', 'sistem_operasi',
    'rekayasa_jaringan_komputer_dasar',
    'administrasi_jaringan',
    'rekayasa_jaringan_komputer_lanjut',
    'manajemen_dan_desain_jaringan',
    'virtualisasi_dan_komputasiawan', 'cyber_security',
    'jaringan_wireless_mobile'
]
```

```

software_engineer_courses = [
    'dasar_pemrograman', 'sistem_digital',
    'rekayasa_perangkat_lunak', 'pemrograman_berorientasi_objek',
    'desain_dan_pemrograman_web', 'sistem_terdistribusi',
    'aplikasi_mobile', 'rekayasa_web', 'sistem_cerdas'
]

data_engineer_courses = [
    'struktur_data_analisis_algoritma',
    'statistik_dan_probabilitas', 'metode_komputasi',
    'desain_algoritma', 'basis_data',
    'administrasi_basis_data', 'sistem_cerdas', 'big_data'
]

iot_engineer_courses = [
    'sistem_embedded_dan_iot', 'jaringan_wireless_mobile',
    'sistem_cerdas', 'cyber_security',
    'sistem_terdistribusi', 'metode_komputasi',
    'rekayasa_jaringan_komputer_lanjut',
    'rekayasa_jaringan_komputer_dasar'
]

# Menghitung rata-rata nilai untuk setiap kategori
data['Avg_Network_Admin'] =
data[network_admin_courses].mean(axis=1)
data['Avg_Software_Engineer'] =
data[software_engineer_courses].mean(axis=1)
data['Avg_Data_Engineer'] =
data[data_engineer_courses].mean(axis=1)
data['Avg_IoT_Engineer'] =
data[iot_engineer_courses].mean(axis=1)

# Menentukan bidang dengan rata-rata nilai tertinggi untuk
setiap mahasiswa
data['Highest_Avg_Field'] = data[['Avg_Network_Admin',
'Avg_Software_Engineer', 'Avg_Data_Engineer',
'Avg_IoT_Engineer']].idxmax(axis=1)

# Mapping untuk profil yang sesuai
field_profile_mapping = {
    'Avg_Network_Admin': 'Network Administrator',
    'Avg_Software_Engineer': 'Software Engineer',
    'Avg_Data_Engineer': 'Associate Data Engineer',
    'Avg_IoT_Engineer': 'IoT Engineer'
}

```



```

}

# Menentukan profil berdasarkan minat bidang ilmu
data['Predicted Profile'] =
data['Highest_Avg_Field'].map(field_profile_mapping)

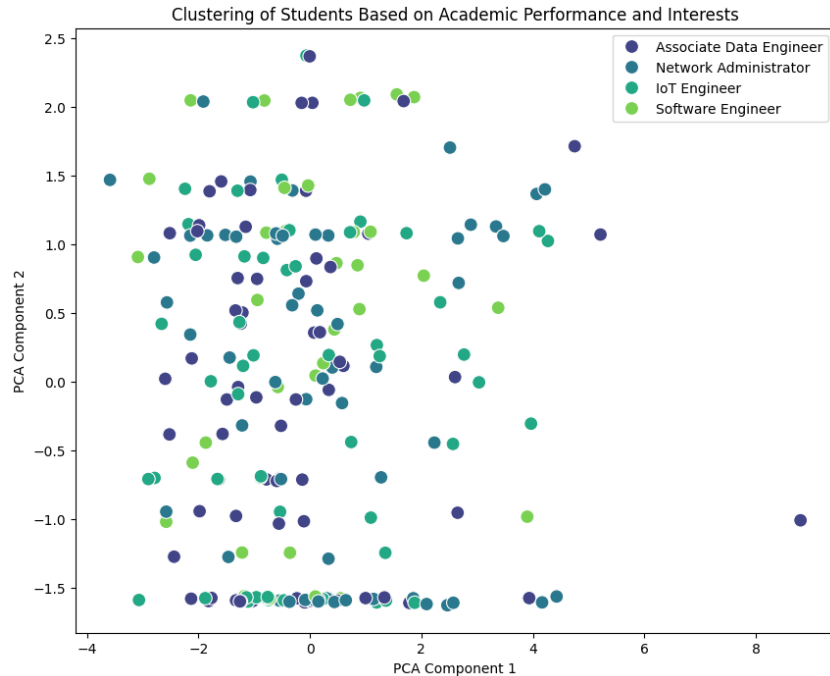
# Encoding kolom kategorikal
label_encoders = {}
for col in ['Minat Bidang Ilmu', 'Minat Profile Lulusan']:
    le = LabelEncoder()
    data[col] = le.fit_transform(data[col])
    label_encoders[col] = le

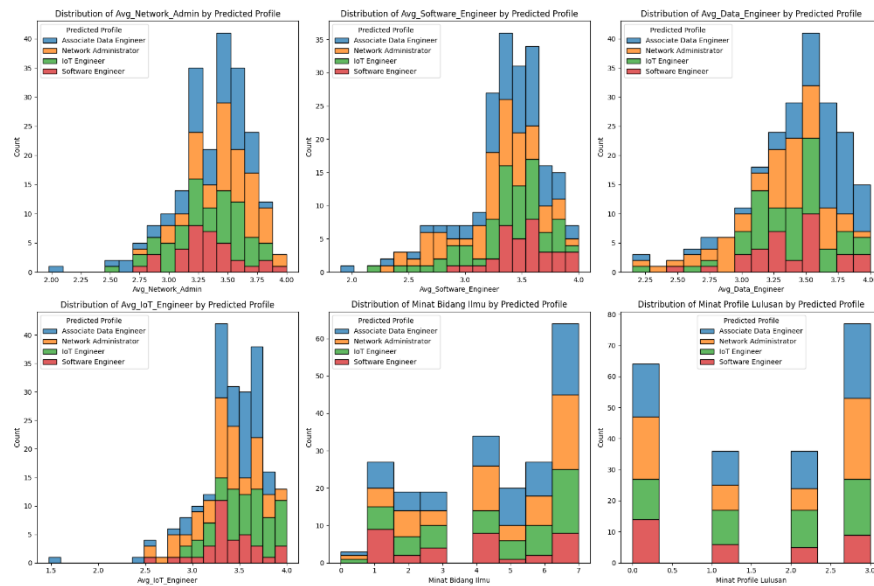
# Menyimpan hasil ke dalam file CSV
data.to_csv('clustered_data_berd_kondisi.csv', index=False)

# Menampilkan hasil pengelompokkan
print(data[['Minat Bidang Ilmu', 'Minat Profile Lulusan',
'Highest_Avg_Field', 'Predicted Profile']])

```

Lampiran 3 Visualisasi Hasil Pengelompokkan





Lampiran 4 *Code* Klasifikasi Metode *Random Forest*

```
# 2. Mengkonversi nilai-nilai grade menjadi numerik
grade_dict = {'A': 4, 'A-': 3.7, 'B+': 3.3, 'B': 3, 'B-': 2.7,
              'C+': 2.3, 'C': 2, 'C-': 1.7, 'D': 1.3, 'D-': 1.0, 'E': 0,
              'T': 0}
for col in data.columns[:-3]: # Diasumsikan tiga kolom
    terakhir adalah kolom kategorikal
    data[col] = data[col].replace(grade_dict).fillna(0)

# 3. Mengisi nilai yang hilang pada kolom numerik dengan rata-
rata kolom tersebut
numeric_columns =
data.select_dtypes(include=['number']).columns
data[numeric_columns] =
data[numeric_columns].fillna(data[numeric_columns].mean())

# 4. Encoding kolom kategorikal
label_encoders = {}
for col in data.columns[-3:]: # Diasumsikan tiga kolom
    terakhir adalah kolom kategorikal
    le = LabelEncoder()
    data[col] = le.fit_transform(data[col])
    label_encoders[col] = le

# 5. Mempersiapkan data untuk pelatihan model
X = data.drop('Predicted Profile', axis=1)
y = data['Predicted Profile']
```

```

# 6. Normalisasi data numerik
scaler = StandardScaler()
X = pd.DataFrame(scaler.fit_transform(X), columns=X.columns)

# 7. Mengatasi ketidakseimbangan kelas dengan
RandomOverSampler
ros = RandomOverSampler(random_state=1)
X_res, y_res = ros.fit_resample(X, y)

# 8. Mendefinisikan model RandomForest dengan pencarian acak
untuk hyperparameter tuning
param_dist = {
    'n_estimators': randint(100, 500),
    'max_depth': randint(3, 20),
    'min_samples_split': randint(2, 20),
    'min_samples_leaf': randint(1, 20),
    'bootstrap': [True, False]
}

rf = RandomForestClassifier(random_state=1,
class_weight='balanced') # Menggunakan keseimbangan kelas
random_search = RandomizedSearchCV(estimator=rf,
param_distributions=param_dist, n_iter=50, cv=5, verbose=2,
random_state=1, n_jobs=-1)
random_search.fit(X_res, y_res)

# 9. Menampilkan hyperparameter terbaik
print("Best parameters found: ", random_search.best_params_)

# 10. Melakukan Cross-validation dengan model terbaik
best_model = random_search.best_estimator_
scores = cross_val_score(best_model, X_res, y_res, cv=5) # 5-
fold cross-validation
print("Cross-validated scores:", scores)
print("Average score:", scores.mean())

# 11. Memisahkan data menjadi set pelatihan dan pengujian
X_train, y_train

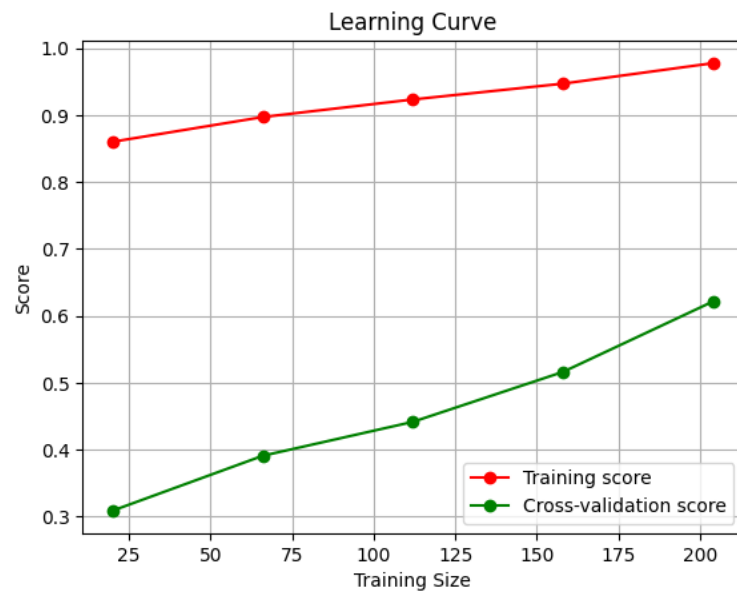
# 12. Melatih model dengan set pelatihan
best_model.fit(X_train, y_train)

# 13. Menyimpan model dan scaler serta encoder
joblib.dump(best_model, 'model_random_forest.pkl')
joblib.dump(scaler, 'scaler.pkl')

```

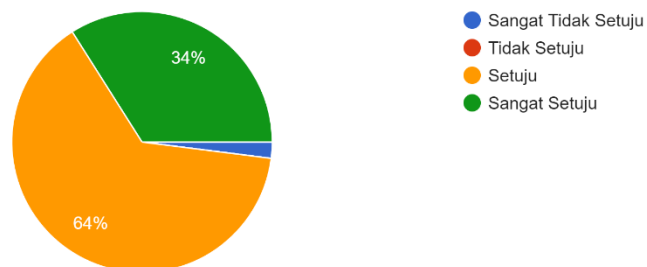
```
joblib.dump(label_encoders, 'label_encoders.pkl')
```

Lampiran 5 *Learning Curve*



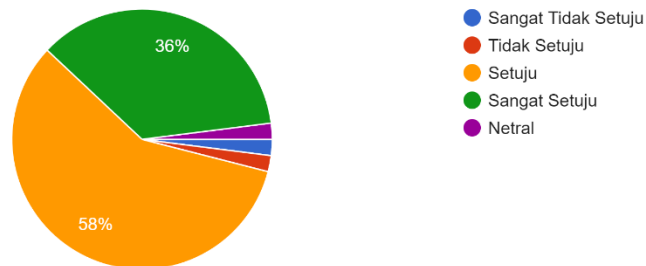
Lampiran 6 Hasil Responden

Aplikasi ini memudahkan saya untuk menentukan topik tugas akhir.
50 jawaban



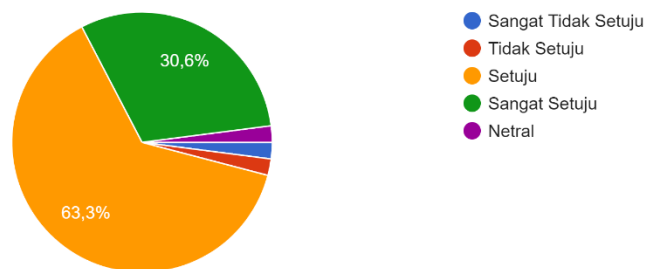
Hasil rekomendasi yang ditampilkan oleh aplikasi ini sangat membantu dalam memilih topik tugas akhir yang sesuai.

50 jawaban



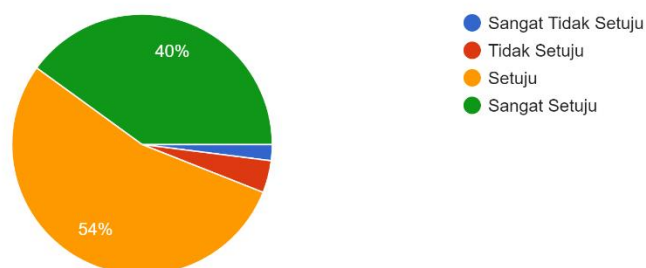
Aplikasi ini membantu saya membuat keputusan yang lebih baik tentang topik tugas akhir mahasiswa.

49 jawaban



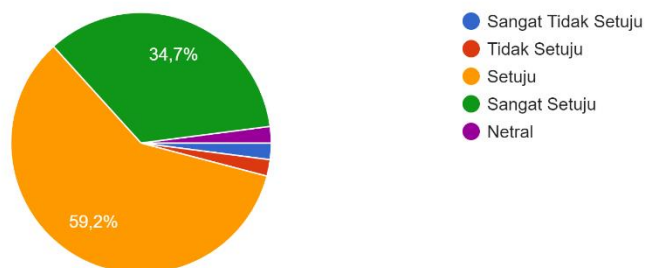
Aplikasi ini mudah digunakan bahkan untuk pengguna baru.

50 jawaban



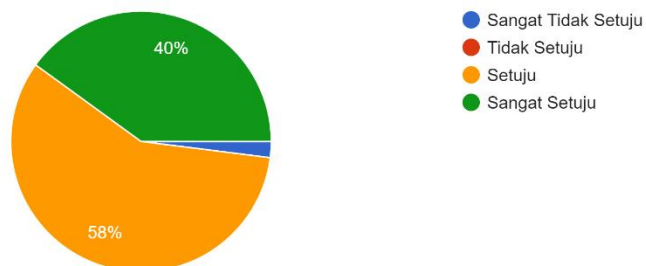
Saya merasa antarmuka aplikasi ini intuitif dan mudah dipahami.

49 jawaban



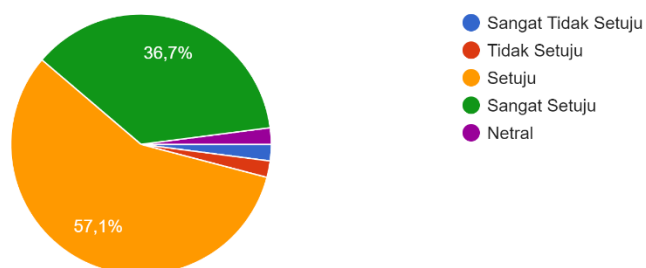
Proses login ke dalam aplikasi ini cepat dan tidak memakan waktu.

50 jawaban



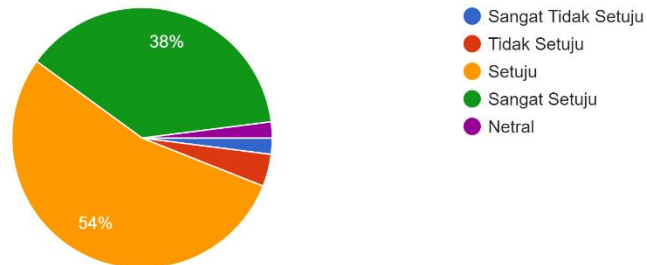
Saya tidak menemui kesulitan di halaman tampilan nilai dan menginput minat mahasiswa di halaman utama.

49 jawaban



Menavigasi dari halaman utama ke halaman hasil rekomendasi sangat mudah dan cepat.

50 jawaban



Secara keseluruhan, saya puas dengan kinerja aplikasi ini.

50 jawaban

