

**IMPLEMENTASI ALGORITMA NAÏVE BAYES UNTUK SISTEM  
REKOMENDASI TOPIK PENELITIAN PADA JURUSAN ILMU  
KOMPUTER UNIVERSITAS LAMPUNG BERBASIS KLASIFIKASI**

**(SKRIPSI)**

**Oleh**

**DIANA ROHMA  
1717051060**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2021**

## ABSTRAK

### IMPLEMENTASI ALGORITMA NAÏVE BAYES UNTUK SISTEM REKOMENDASI TOPIK PENELITIAN PADA JURUSAN ILMU KOMPUTER UNIVERSITAS LAMPUNG BERBASIS KLASIFIKASI

Oleh

DIANA ROHMA

Algoritma naïve bayes merupakan algoritma Klasifikasi bersifat statistik yang dilakukan berdasarkan data training sebagai proses klasifikasi untuk data baru. Salah satu penerapan algoritma naïve bayes adalah pembuatan sistem rekomendasi topik penelitian pada jurusan ilmu komputer universitas lampung. Jurusan ilmu komputer memiliki beberapa konsentrasi tema topik penelitian sebagai acuan untuk menentukan judul penelitian. Meskipun banyak tema yang dapat diambil oleh mahasiswa, Saat ini masih banyak mahasiswa yang kesulitan dalam menentukan topik penelitian. Oleh karena itu diperlukan sistem rekomendasi untuk memudahkan mahasiswa dalam menentukan topik penelitian. Penelitian ini membahas tentang sistem rekomendasi yang menerapkan algoritma naïve bayes dengan menggunakan 240 data mahasiswa jurusan ilmu komputer universitas lampung yang telah selesai pendidikan tinggi negeri dengan pembagian data sebesar 90% sebagai data training dan 10% sebagai data testing. Berdasarkan proses pembangunan sistem rekomendasi, digunakan dua tahapan penting pada penelitian yaitu yang pertama pengembangan machine learning dengan algoritma klasifikasi naïve bayes pada bahasa pemrograman python yang menggunakan evaluasi model confusion matrix dan mendapatkan akurasi sebesar 68%. Yang kedua pengembangan sistem dilakukan dengan menggunakan *framework* Flask dengan tahapan pengerjaan menggunakan metode *Extreme Programming* dan telah menggunakan pengujian *black box testing* dengan teknik *Equivalence Partitioning* yang menunjukkan hasil bahwa sistem telah berjalan sesuai dengan kebutuhan yang diharapkan.

Kata Kunci : sistem rekomendasi, klasifikasi, naïve bayes, flask, *black bos testing*.

## **ABSTRAK**

### **IMPLEMENTATION OF NAÏVE BAYES ALGORITHM FOR RESEARCH TOPIC RECOMMENDATION SYSTEM AT COMPUTER SCIENCE DEPARTMENT OF UNIVERSITY BASED ON CLASSIFICATION**

**By**

**DIANA ROHMA**

Nave Bayes algorithm is a statistical classification algorithm that is implemented based on training data as a classification process for new data. One implementation of the Nave Bayes algorithm is the creation of a research topic recommendation system at the computer science department at the University of Lampung. The computer science department has several concentrations of research topic themes as a reference for determining research titles. Although there are many themes that can be taken by students, currently there are many students who have difficulty in determining research topics. Therefore, a recommendation system is needed to make it easier for students to determine research topics. In this study, a recommendation system was built that applies the nave Bayes algorithm by using 240 data of computer science students at the University of Lampung who have completed higher state education that is used 90% as training data and 10% as testing data. Based on the recommendation system development process. Two important stages were used in the research, the first was the development of machine learning with the nave Bayes classification algorithm in the python programming language which used the evaluation of the confusion matrix model and obtained an accuracy of 68%. The second, system development is carried out using the Flask framework with the stages of work using the Extreme Programming method and has used black box testing with the Equivalence Partitioning technique which shows the results that the system has been running as expected.

Keywords: recommendation system, classification, nave bayes, flask, black bos testing.

**IMPLEMENTASI ALGORITMA NAÏVE BAYES UNTUK SISTEM  
REKOMENDASI TOPIK PENELITIAN PADA JURUSAN ILMU  
KOMPUTER UNIVERSITAS LAMPUNG BERBASIS KLASIFIKASI**

**Oleh**

**DIANA ROHMA**

**Skripsi**

**Sebagai Salah Satu Syarat untuk Mencapai Gelar  
SARJANA ILMU KOMPUTER**

**Pada**

**Jurusan Ilmu Komputer  
Fakultas Matematika dan Ilmu Pengetahuan Alam**



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2021**

Judul Skripsi

: **IMPLEMENTASI ALGORITMA NAÏVE  
BAYES UNTUK SISTEM REKOMENDASI  
TOPIK PENELITIAN PADA JURUSAN  
ILMU KOMPUTER UNIVERSITAS  
LAMPUNG BERBASIS KLASIFIKASI**

Nama Mahasiswa

: **Diana Rohma**

Nomor Pokok Mahasiswa

: 1717051060

Jurusan

: Ilmu Komputer

Fakultas

: Matematika dan Ilmu Pengetahuan Alam



1. Komisi Pembimbing

**Dr. Ir. Kurnia Muludi, M.S.Sc.**  
NIP 19640616 198902 1 001

**Ir. Gigih Forda Nama, S.T., M.T.I.**  
NIP 19830712 200812 1 003

2. Ketua Jurusan Ilmu Komputer

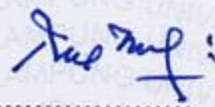
**Didik Kurniawan, S.Si., M.T.**  
NIP 19800419 200501 1 004

## MENGESAHKAN

### 1. Tim Penguji

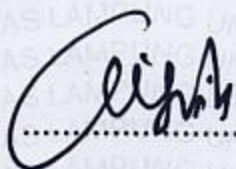
Ketua

: **Dr. Ir. Kurnia Muludi, M.S.Sc.**



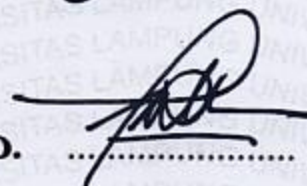
Penguji I  
Sekretaris

: **Ir. Gigih Forda Nama, S.T., M.T.I.**



Penguji II  
Penguji Pembahas

: **Favorisen R. Lumbanraja, Ph.D.**



### 2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



**Dr. Eng. Supto Dwi Yuwono, S.Si., M.T.**

NIP. 19740705 200003 1 001



Tanggal Lulus Ujian Skripsi : **9 Desember 2021**

## PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama: Diana Rohma

NPM: 1717051060

Dengan ini menyatakan bahwa skripsi yang berjudul "IMPLEMENTASI ALGORITMA NAÏVE BAYES UNTUK SISTEM REKOMENDASI TOPIK PENELITIAN PADA JURUSAN ILMU KOMPUTER UNIVERSITAS LAMPUNG BERBASIS KLASIFIKASI" adalah benar hasil karya sendiri dan bukan orang lain. Seluruh tulisan yang tertuang dalam skripsi ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Jika di kemudian hari terbukti skripsi penulis adalah hasil penjiplakan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan peraturan yang berlaku.

Bandar Lampung, 17 Desember 2021

Penulis



Diana Rohma

NPM. 1717051060

## RIWAYAT HIDUP



Penulis dilahirkan di Bumi Nabung Baru pada tanggal 20 Desember 1998 sebagai anak ketiga dari tiga bersaudara pasangan Bapak Ali Amin dan Ibu Karsinah. Penulis menyelesaikan Pendidikan Sekolah Dasar (SD) di SD Negeri 1 Bumi Nabung Baru pada tahun 2011. Kemudian pendidikan Sekolah Menengah Pertama (SMP) di SMP Negeri 1 Rumbia yang diselesaikan pada tahun 2014. Kemudian melanjutkan pendidikan Sekolah Menengah Kejuruan (SMK) di SMK Muhammadiyah 1 Rumbia yang diselesaikan pada tahun 2017.

Penulis terdaftar sebagai mahasiswa Jurusan Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung pada tahun 2017 melalui jalur Seleksi Bersama Masuk Perguruan Tinggi Negeri (SBMPTN). Selama menjadi mahasiswa, penulis melakukan beberapa kegiatan antara lain.

1. Menjadi anggota Adapter Himpunan Mahasiswa Jurusan Ilmu Komputer pada periode 2017/2018.
2. Menjadi anggota pengurus di Bidang Kaderisasi Himpunan Mahasiswa Jurusan Ilmu Komputer (Himakom) pada periode 2017/2018.
3. Menjadi Bendahara di Bidang Kaderisasi Himpunan Mahasiswa Jurusan Ilmu Komputer (Himakom) pada periode 2019.



4. Melaksanakan Kuliah Kerja Nyata (KKN) pada periode I tahun ajaran 2020 di desa Labuhan Baru, Kecamatan Way Serdang, Kabupaten Mesuji.
5. Melaksanakan Kerja Praktik di Kantor Desa Totokaton pada tahun 2020.
6. Mengikuti ujian sertifikasi dan mendapatkan sertifikat *Junior Office Operator* (JOO) yang diselenggarakan oleh Badan Nasional Sertifikasi Profesi (BNSP) pada tahun 2020.

## **MOTTO**

**“Lā yukallifullāhu nafsan illā wus'ahā”**

“Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya.”

**(Q.S Al-Baqarah: 286)**

“Kesabaran itu ada dua macam, pertama sabar atas sesuatu yang tidak kau inginkan, dan kedua sabar menahan diri dari sesuatu yang kau inginkan

**(Ali Bin Abi Thalib)**

“Tidak peduli seberapa kacau keadaannya, kamu selalu bisa menemukan solusi yang terbaik”

**(Eren Yeager)**

“Sabar adalah ilmu tingkat tinggi, bentuk syukur paling dalam, dan bentuk **Ikhlas Paling Serious**”

**(Fajar Sulaiman)**

## PERSEMBAHAN

### *Alhamdulillahillobbilamin*

Puji dan syukur saya panjatkan kepada Allah SWT atas rahmat dan hidayah-Nya sehingga saya dapat kemudahan dalam menyelesaikan penulisan skripsi ini.

Shalawat serta salam selalu tercurahkan kepada Nabi Muhammad SAW.

Kupersembahkan karya ini kepada:

### **Kedua Orang Tuaku Tercinta**

Yang senantiasa memberikan yang terbaik, dan melantunkan do'a yang selalu menyertaiku. Kuucapkan pula terima kasih sebesar-besarnya karena telah mendidik dan membesarkanku dengan cara yang dipenuhi kasih sayang, dukungan, dan pengorbanan yang belum bisa terbalaskan.

### **Kedua kakakku Yuli dan Dedi serta Kedua kakak iparku Edi dan Devi**

Terima kasih telah memberikan do'a, dukungan dan semangat, serta keceriaan yang kalian berikan untukku.

### **Seluruh Keluarga Besar, Sahabat, dan Teman-temanku**

yang selalu memberikan keceriaan, semangat dan dukungan.

### **Almamater Tercinta, Universitas Lampung**

## SANWACANA

Alhamdulillahirobbil alamiin, Puji syukur ke hadirat Allah SWT atas rahmat, karunia serta hidayah-Nya. Berkat-Nya serta petunjuk dan pedoman dari Rasulullah Nabi Muhammad Shalallahu Alaihi Wasallam penulis dapat menyelesaikan skripsi dengan judul “Implementasi Algoritma Naïve Bayes Untuk Sistem Rekomendasi Topik Penelitian Pada Jurusan Ilmu Komputer Universitas Lampung Berbasis Klasifikasi” dengan lancar.

Dalam kesempatan ini, rasa hormat dan terima kasih yang sebesar-besarnya penulis ucapkan kepada semua pihak yang telah berperan besar membimbing dan membantu dalam penyusunan skripsi ini. Secara tertulis penulis mengungkapkan terima kasih kepada:

1. Bapak, mamak, yayuk, mamas, kedua kakak ipar dan keluarga besar yang telah mencintai, menyayangi dan mendukung, degan do'a tanpa henti. Terima Kasih dan kasih sayang akan selalu diberikan untuk kalian.
2. Bapak Dr. Ir. Kurnia Muludi, M.S.Sc. sebagai pembimbing utama yang telah membimbing saya, memberikan kritik dan saran, serta membina dalam menyusun dan menyelesaikan skripsi ini dengan baik.
3. Bapak Gigih Forda Nama, S.T., M.T.I sebagai pembimbing kedua saya yang telah membimbing, mengarahkan, dan memberikan ide, kritik, saran sehingga skripsi ini dapat diselesaikan dengan baik.
4. Bapak Favorisen R. Lumbanraja, Ph. D. sebagai pembahas yang telah memberikan masukan dan saran yang bermanfaat dalam penyusunan dan perbaikan skripsi ini.
5. Bapak Rico Andrian, S.Si., M.Kom. selaku pembimbing akademik penulis yang telah memberikan bimbingan, arahan, dan nasihat kepada penulis dalam menyelesaikan proses belajar.

6. Bapak Dr. Suripto Dwi Yuwono, M.T. selaku Dekan FMIPA Universitas Lampung.
7. Bapak Didik Kurniawan, S.Si., M.T., selaku Ketua Jurusan Ilmu Komputer FMIPA Universitas Lampung.
8. Ibu Astria Hijriani, S.Kom., M.Kom. selaku sekretaris Jurusan Ilmu Komputer FMIPA Universitas Lampung.
9. Bapak dan Ibu Dosen Jurusan Ilmu Komputer FMIPA Universitas Lampung yang telah memberikan banyak ilmu bermanfaat dan pengalaman selama penulis menjadi mahasiswa.
10. Ibu Ade Nora Maela, Kak Zainuddin dan Kak Ardi Novalia yang telah membantu segala urusan administrasi penulis di Jurusan Ilmu Komputer.
11. Keluargaku adek Cinta, adek Khabir yang telah menghibur dan memberikan keceriaan kepada saya.
12. Nonik Aisyah selaku sahabat, teman sekamar, serta partner skripsi. Terima kasih telah bekerja sama dalam kehidupan selama perkuliahan dan penyusunan skripsi dengan sabar dan baik hati.
13. Intan Indah Larasati sebagai teman satu atap yang memarahi, mendukung, dan memberi semangat mengerjakan tugas dan menyusun skripsi.
14. Sahabat-sahabatku tersayang Maria Ulfa, Rita Anggraini, Noverina Ikatama, Reka Amelia, Reda Meiningiyas, M. Bella BN dan Adam Malik yang selalu membantu, mendukung, menyemangati dan menghibur, serta menjadi tempat keluh kesah. Terima kasih banyak untuk semua yang telah kalian berikan.
15. Eki Tri Suenda, Ester Caroline, Wildan Mutaqin, Naurah Nazhifah yang telah membantu menyusun skripsi ini dengan lancar.
16. Teman-teman Goyang Squad yang telah menyemangati dan menemani penulis dalam menjalani keseharian selama masa perkuliahan.
17. Teman-teman Jurusan Ilmu Komputer Angkatan 2017 yang telah menjadi keluarga satu angkatan selama menjalani kehidupan perkuliahan di Jurusan Ilmu Komputer Universitas Lampung.
18. Seluruh pihak yang tidak bisa saya sebutkan satu persatu yang telah banyak membantu selama perkuliahan.

19. Seseorang yang pernah hadir dalam waktu singkat dan membuat saya ingin segera menyelesaikan perkuliahan.

20. Diana Rohma. Terima kasih telah menjalani percaya dan berjuang dengan hebat.

Selama penyusunan skripsi, penulis menyadari terdapat kekurangan yang disebabkan oleh kurangnya pengalaman, kemampuan dan pengetahuan penulis. Semoga isi dari skripsi ini memberikan manfaat bagi pihak yang membaca.

Bandar Lampung, 17 Desember 2021

Penulis

Diana Rohma

NPM. 1717051060

## DAFTAR ISI

Halaman

<b>DAFTAR ISI</b> .....	<b>v</b>
<b>DAFTAR TABEL</b> .....	<b>vii</b>
<b>DAFTAR GAMBAR</b> .....	<b>viii</b>
<b>1. PENDAHULUAN</b> .....	<b>1</b>
<b>1.1 Latar Belakang</b> .....	1
<b>1.2 Rumusan Masalah</b> .....	3
<b>1.3 Batasan Masalah</b> .....	3
<b>1.4 Tujuan Penelitian</b> .....	3
<b>1.5 Manfaat Penelitian</b> .....	3
<b>2. TINJAUAN PUSTAKA</b> .....	<b>4</b>
<b>2.1 Sistem Rekomendasi</b> .....	4
<b>2.2 Data Mining</b> .....	4
<b>2.3 Klasifikasi</b> .....	5
<b>2.4 Naïve Bayes</b> .....	6
<b>2.5 Data Cleaning</b> .....	6
<b>2.6 Data Transformation</b> .....	7
<b>2.7 Correlation Based Feature Selection</b> .....	7
<b>2.8 SMOTE</b> .....	8
<b>2.10 Grid Search</b> .....	9
<b>2.12 Python</b> .....	10
<b>2.13 Flask</b> .....	11
<b>2.14 UML</b> .....	12
<b>2.15 Balsamiq Mockup</b> .....	13
<b>2.16 Extreme Programming</b> .....	13
<b>2.17 Black box Testing</b> .....	15
<b>2.18 Penelitian Terdahulu</b> .....	15

<b>III. METODOLOGI PENELITIAN.....</b>	<b>18</b>
<b>3.1 Tempat dan Waktu Penelitian .....</b>	<b>18</b>
<b>3.2 Data dan Alat Pendukung .....</b>	<b>19</b>
<b>3.3 Tahapan Penelitian .....</b>	<b>20</b>
<b>IV. HASIL DAN PEMBAHASAN.....</b>	<b>35</b>
<b>4.1 Hasil.....</b>	<b>35</b>
<b>4.2 Pembahasan.....</b>	<b>42</b>
<b>V. SIMPULAN DAN SARAN.....</b>	<b>63</b>
<b>5.1 Simpulan .....</b>	<b>63</b>
<b>5.2 Saran .....</b>	<b>64</b>
<b>DAFTAR PUSTAKA .....</b>	<b>65</b>



## DAFTAR TABEL

Tabel	Halaman
1. Pedoman Nilai Interpretasi Koefisien Korelasi.....	8
2. Tabel <i>Confusion Matrix</i> .....	9
3. Penelitian Terdahulu .....	15
4. Penelitian Saat Ini .....	17
5. User Stories Sistem .....	24
6. Black Box Testing Scenario Sistem Rekomendasi Topik Penelitian.....	32
7. Data Missing Value.....	36
8. Contoh Data Hasil Proses Data Cleaning .....	36
9. Proses Data Transformation Pada Data Atribut Mata Kuliah .....	37
10. Contoh Data Hasil Proses <i>Data Transformation</i> .....	37
11. Tabel Hasil Menentukan Perbandingan Nilai K .....	39
12. Hasil Perbandingan Menentukan Nilai <i>Training</i> Dan <i>Testing</i> .....	40
13. Hasil Nilai Parameters model Klasifikasi .....	41
14. Hasil Nilai TP, FP, FN.....	42
15. Hasil Evaluasi Model Klasifikasi.....	42
16. Dokumentasi Pengembangan Sistem dengan <i>Extreme Programming</i> .....	49
17. Hasil Pengujian <i>Black Box</i> .....	59

## DAFTAR GAMBAR

Gambar	Halaman
1 Arsitektur Flask (Raharjo, 2019) .....	12
2 Kerangka Extreme Programming (Pressman & Maxim, 2014) .....	14
3. Tahapan Penelitian.....	20
4 Flowchart Pengembangan Model Machine Learning .....	22
5. Class Responsibility Collaborator.....	25
6 Use Case Diagram Sistem.....	26
7 Activity Diagram Login.....	26
8 Activity Manage Data Mahasiswa .....	27
9 Activity Diagram Prediksi .....	27
10 Activity Diagram Lihat Kategori .....	28
11 Antarmuka Login.....	29
12 Antarmuka Manage Data Mahasiswa .....	29
13 Antarmuka Tambah Data .....	30
14 Antarmuka Edit Data .....	30
15 Antarmuka Halaman Prediksi .....	31
16 Antarmuka Lihat Kategori .....	31
17. Nilai Korelasi Antar Atribut .....	38
18. Hasil Nilai Korelasi Antar Atribut Setelah Menghapus Atribut Statistika .....	38
19. Gambar Data Sebelum Proses SMOTE .....	39
20. Hasil Data Setelah Proses SMOTE.....	40
21. Gambar Hasil Confusion Matrix.....	41
22. Hasil Tahap 1 Bagian Login .....	51
23. Hasil tahap 2 bagian beranda dan login .....	51
24. Tahap 2 Tampilan Tambah Data.....	52
25. Tahap 3 bagian tambah data.....	52
26. Tahap 3 Bagian Hasil Prediksi.....	53
27. Tahap 3 Bagian Sweet Alert .....	53
28. Halaman Beranda.....	54

29. Halaman Kategori .....	55
30. Halaman Prediksi .....	55
31. Halaman Pengelolaan Data Mahasiswa .....	56
32. Modal Tambah Data .....	57
33. Modal Edit Data .....	57

## DAFTAR KODE PROGRAM

Kode Program	Halaman
1. Kode Proses <i>Data Transformation</i> .....	43
2. Kode menampilkan korelasi.....	44
3. Kode Melakukan Proses SMOTE.....	45
4. Kode Pembagian Data Penelitian.....	45
5. Kode Membangun Model Klasifikasi.....	47
6. Kode Proses <i>Confusion Matrix</i> .....	47

# 1. PENDAHULUAN

## 1.1 Latar Belakang

Bidang Ilmu komputer melingkupi beragam topik yang berkaitan erat dengan komputer, seperti rekayasa perangkat lunak, komputasi, pemrograman komputer, dan hal-hal yang mencangkup komputer lainnya. Dalam bidang ilmu komputer, banyak topik penelitian yang dapat digunakan sebagai tema dalam pengambilan penelitian diantaranya *Data science*, sistem pakar, *machine learning*, *website*, pemrograman *mobile*, dan sebagainya. Masing-masing topik penelitian memiliki mata kuliah yang berkaitan erat dengan topiknya. Adapun mata kuliah wajib yang diperlukan dalam bidang ilmu komputer adalah matematika, statistika, logika, pemrograman, dan sebagainya.

Jurusan Ilmu Komputer Universitas Lampung merupakan salah satu jurusan yang terakreditasi A. setiap tahunnya banyak mahasiswa yang berhasil mendapatkan gelar S.Kom yang didapat dari Universitas Lampung dengan bidang ilmu komputer. Topik penelitian yang banyak diambil mahasiswa yaitu *machine learning*, sistem informasi, dan *mobile*. Dalam pengambilan tema penelitian, mahasiswa dibebaskan untuk memilih topik yang sesuai dengan kemampuannya. Beberapa mahasiswa mengambil *project* dari dosen, sehingga tema topik penelitiannya di tentukan oleh dosen. *Project* yang ditawarkan dosen memiliki batas maksimal kuota mahasiswa sehingga banyak juga mahasiswa yang masih kesulitan dalam menentukan topik penelitian untuk dijadikan sebagai skripsi. Karena itu, untuk memudahkan mahasiswa dalam menentukan topik skripsi dapat menggunakan sistem rekomendasi.

Sistem rekomendasi merupakan suatu sistem informasi yang dapat memberikan rekomendasi suatu item kepada pengguna berdasarkan data yang telah ada. Fungsi dari Sistem rekomendasi yaitu memberikan informasi dan fasilitas kepada pengguna dalam menentukan suatu keputusan (Aprilia &

Fachrurrozi, 2016). Implementasi sistem rekomendasi menggunakan analisis data bersifat klasifikasi. Yang dimaksud klasifikasi yaitu menentukan data baru yang dimasukkan kedalam kelas tertentu berdasarkan data yang ada sebelumnya. Dalam penggunaan analisis data klasifikasi terdapat beberapa algoritma yang dapat digunakan, salah satunya yaitu algoritma naïve bayes. Algoritma naïve bayes mengimplementasikan sistem rekomendasi yang berbasis klasifikasi.

Algoritma naïve bayes merupakan algoritma klasifikasi yang sering digunakan dalam pemecahan masalah terkait klasifikasi data. Pertama kali diperkenalkan oleh Thomas bayes. Untuk melakukan klasifikasi didalam data mining dengan menggunakan algoritma yang bersifat statistik, dapat menggunakan algoritma naïve bayes (Lavindi et al., 2019). Perhitungan pada algoritma ini menggunakan probabilitas berdasarkan data independen atau atribut. Algoritma ini sudah banyak digunakan dalam pengklasifikasian data. Naïve Bayes termasuk kedalam klasifikasi atau metode *supervised* yang berarti harus menggunakan data *training* sebelum melakukan proses klasifikasi data (Devita et al., 2018). Keuntungan penggunaan algoritma Naive Bayes yaitu bahwa algoritma ini hanya membutuhkan *Training Data* dengan jumlah yang kecil untuk menentukan prediksi parameter dalam proses pengklasifikasian (Saleh, 2015). Selain itu algoritma naïve bayes merupakan algoritma yang sederhana dan efisien (Pujianto et al., 2019).

Pada penelitian ini berfokus pada pembuatan sistem rekomendasi dengan menggunakan algoritma naïve bayes dalam bahasa pemrograman Python. Penelitian ini diharapkan dapat mempermudah mahasiswa Jurusan Ilmu Komputer dalam menentukan topik penelitiannya sesuai dengan nilai akhir mata kuliah dalam bidang ilmu komputer yang didapatkan.

## **1.2 Rumusan Masalah**

Rumusan masalah pada penelitian ini adalah Bagaimana merekomendasikan topik penelitian berbasis klasifikasi dengan menggunakan algoritma naïve bayes berdasarkan *input* nilai mata kuliah.

## **1.3 Batasan Masalah**

1. Sistem ini hanya dapat merekomendasikan topik penelitian berdasarkan data dari tahun 2016-2019 dalam bidang ilmu komputer di jurusan ilmu komputer Universitas Lampung
2. Dalam penelitian ini membahas penerapan algoritma naïve bayes dalam sistem rekomendasi yang dibuat.

## **1.4 Tujuan Penelitian**

Tujuan dari penelitian ini adalah mengembangkan sistem rekomendasi topik penelitian berdasarkan nilai akhir mata kuliah terkait dalam bidang ilmu komputer.

## **1.5 Manfaat Penelitian**

Adapun manfaat dari penelitian ini yaitu mahasiswa dapat menggunakan sistem rekomendasi ini untuk menentukan topik penelitian yang dapat diambil sesuai dengan kemampuannya.

## **2. TINJAUAN PUSTAKA**

### **2.1 Sistem Rekomendasi**

Konsep dari sistem rekomendasi sudah banyak digunakan oleh aplikasi atau sistem dalam dunia bisnis seperti youtube, netflix, instagram, shopee dan berbagai macam aplikasi lainnya. Ini berguna untuk mempengaruhi pengguna dalam memilih objek agar pengguna tertarik pada objek yang telah direkomendasikan oleh sistem. Sistem rekomendasi merupakan suatu program untuk memprediksi objek atau item yang dibutuhkan pengguna secara alternatif. Dalam sistem rekomendasi terdapat data sebagai data latih. Fungsi dari Sistem rekomendasi yaitu memberikan informasi dan fasilitas kepada pengguna dalam menentukan suatu keputusan (Aprilia & Fachrurrozi, 2016). Sistem rekomendasi dapat disebut dengan sistem pengambilan keputusan karena memberikan bantuan dalam menentukan suatu objek yang diperlukan. Bertujuan untuk memberikan saran objek kepada pengguna. Sistem rekomendasi dapat dikaitkan dengan model aplikasi yang merupakan hasil dari observasi terhadap keadaan dan keinginan pengguna. Oleh sebab itu, untuk mendapatkan rekomendasi yang sesuai dengan keinginan pengguna dan mempermudah pengguna dalam pengambilan keputusan untuk menentukan objek yang tepat yang akan di pilih, diperlukan sistem rekomendasi yang memiliki model rekomendasi yang tepat (Utomo & Anggriawan, 2015). Dalam implementasi sistem rekomendasi diperlukan algoritma yang tepat, karena setiap algoritma yang berbeda memiliki kinerja sistem, hasil rekomendasi, dan efisiensi operasi yang berbeda (Shuxian & Sen, 2019a).

### **2.2 Data Mining**

Data mining merupakan proses analisis kumpulan data dalam jumlah yang besar dengan tujuan untuk menemukan keterkaitan antar variable dan mengolah data agar mudah dipahamin (Mardi, 2017). Data Mining



menerapkan berbagai algoritma dan model dalam proses analisis kumpulan datanya (Larose, 2006). Berdasarkan jenis tugas dari data mining, dapat dibagi menjadi 6 (Mardi, 2017)

1. Description (Deskripsi)
2. Estimation (Estimasi)
3. Prediction (Prediksi)
4. Classification (Klasifikasi)
5. Clustering (Pengkusteran)
6. Association (Asosiasi)

### 2.3 Klasifikasi

Metode klasifikasi atau dapat juga disebut metode *supervised* merupakan proses memisahkan kelas data berdasarkan data yang ada untuk menentukan kelas data target. Klasifikasi digunakan dalam memprediksi kategori label kelas berdasarkan model yang telah dibangun dengan kumpulan data data latih dan label kelas yang dapat mengklasifikasikan data pengujian yang baru (Jadhav & Channe, 2016). Klasifikasi merupakan dasar dalam analisis data yang membutuhkan pembangunan fungsi yang menentukan label kelas yang telah ditetapkan oleh atribut. Dalam menentukan kelas dari data sampel yang akan kita bagi, dapat menggunakan tipe analisis data klasifikasi yang bertujuan untuk meningkatkan kehandalan hasil data yang telah diklasifikasi (Hendrian, 2018). Klasifikasi dilakukan berdasarkan data contoh yang disebut *training data* sebagai acuan pengklasifikasi untuk menentukan keputusan kelas data yang tidak ada di data *training*. Terdapat dua tahapan dalam klasifikasi data yaitu tahap pembelajaran dan tahap pengklasifikasian. Untuk tahap pembelajaran yang dilakukan yaitu pembentukan model klasifikasi, sedangkan untuk memprediksi label kelas dari suatu data termasuk kedalam tahap pengklasifikasian (Sartika & Indra, 2017). Banyak algoritma yang termasuk kedalam metode klasifikasi, termasuk algoritma naïve bayes.

## 2.4 Naïve Bayes

Salah satu algoritma data mining yang terkenal adalah naïve bayes sebagai kalsifikasi data. Naïve Bayes merupakan algoritma yang termasuk kedalam data mining yang biasa digunakan untuk melakukan klasifikasi yang bersifat statistik (Lavindi et al., 2019). Metode naïve bayes merupakan algoritma dengan pendekatan statistik untuk melakukan inferensi induksi pada persoalan klasifikasi (Annur, 2018). Klasifikasi yang dilakukan berdasarkan data yang sudah ada atau biasa disebut data *training* sebagai proses klasifikasi untuk data baru. Terdapat tahap pada proses klasifikasi yaitu tahap pengklasifikasian yang menghitung nilai probabilitas pada masing-masing label kelas terhadap data yang diberikan. Yang akan dijadikan label kelas data masukan tersebut adalah label kelas dengan nilai probabilitas paling besar (Sartika & Indra, 2017). Rumus umum dalam klasifikasi menggunakan algoritma naïve bayes yaitu sebagai berikut (Jadhav & Channe, 2016) :

$$P(C | X) = \frac{P(X|C)P(C)}{P(X)}$$

Keterangan :

X = Data dengan *class* yang belum diketahui

C = Hipotesis data X merupakan suatu *class* spesifik

P(C|X) = Probabilitas Posterior dari kelas target.

P(C) = Probabilitas Kelas Sebelumnya.

P(X|C) = Probabilitas X berdasarkan kelas tertentu.

P(X) = Probabilitas dari X

Keuntungan algoritma naïve bayes dalam klasifikasi (Jadhav & Channe, 2016):

1. Waktu Komputasi yang lebih cepat
2. Meningkatkan kinerja klasifikasi
3. Kinerja yang baik

## 2.5 Data Cleaning

*Data Cleaning* merupakan proses pembersihan data untuk mengisi nilai yang hilang, menghilangkan data yang duplikat, dan memperbaiki data yang

inkonsisten. Proses ini dibagi menjadi dua. Yaitu *Data Cleaning* untuk data noise dan *Data Cleaning* untuk data missing value (Agarwal, 2014). Dalam menangani missing value, data dapat diproses dengan mengabaikan kumpulan data yang hilang (tupel), mengisi data yang hilang dengan memasukkan nilai secara manual atau dengan komputasi (Nurvinda, 2021), dan mengisi data yang hilang menggunakan konstanta global seperti contohnya “Unknow” (Agarwal, 2014).

## **2.6 Data Transformation**

*Data transformation* merupakan proses mengubah dataset menjadi bentuk yang sesuai untuk dilakukan proses pada data mining dengan tidak mengurangi data asli (Astuti & Ferinanto, 2016). Proses transformasi data mengubah format data atau struktur pada dataset asli. Hal ini merupakan proses penting pada klasifikasi yang berfokus pada properti statistik, semantik, dan struktur data sebelum dilakukannya proses *Machine Learning* (Sajid et al., 2019).

## **2.7 Correlation Based Feature Selection**

Feature selection merupakan proses mendapatkan subset dari set fitur asli dengan memilih fitur yang memiliki hubungan erat dengan dataset dan menghapus fitur yang berlebihan dan tidak relevan (Cai et al., 2018). Terdapat dua tahap penting Seleksi fitur. Pertama, memilih fitur yang memiliki hubungan erat dengan kelas dalam dataset. Kedua, menghapus fitur yang sama dan fitur yang berlebihan. Proses Ini dapat mengurangi dimensi dataset asli dan meningkatkan kinerja algoritma (Gopika & Meena Kowshalaya, 2018). Berdasarkan hubungan dengan metode pembelajaran, kategori feature selection dapat diklasifikasikan ke dalam model filter, wrapper, dan embedded (Cai et al., 2018). Berbagai penelitian terdahulu menggunakan feature selection (Saifudin, 2013) dengan menggunakan metode filtering, seperti *Correlation Based Feature Selection* (Hall, 1999). *Correlation Based Feature Selection* menghasilkan fitur atau atribut yang tidak saling berkorelasi dan berkorelasi erat dengan kelas. Fitur yang tidak relevan dengan kelas harus diabaikan karena akan memiliki korelasi yang rendah. Fitur yang redundansi

harus dihapus karena akan berkorelasi satu sama lain (Hall, 1999). Pedoman nilai interpretasi koefisien korelasi dalam menentukan hubungan antar fitur dapat dilihat pada Tabel 1.

Tabel 1. Pedoman Nilai Interpretasi Koefisien Korelasi (Sugiyono, 2016).

Interval Koefisien	Tingkat Hubungan
0,00 – 0,199	Sangat Rendah
0,20 – 0,399	Rendah
0,40 – 0,599	Sedang
0,60 – 0,799	Kuat
0,80 – 1,000	Sangat Kuat

## 2.8 SMOTE

SMOTE atau singkatan dari *Syntetic Minority Over-sampling Technique* merupakan metode over-sampling dengan memperbanyak data pada kelas minoritas menggunakan data sintetik yang merupakan tiruan dari data pada kelas minoritas lalu mencari k-nearest neighbor dari setiap instance (Sutoyo & Fadlurrahman, 2020). Over-sampling pada metode SMOTE mengambil contoh data dari kelas minoritas dan menghasilkan contoh data sintetik dari kelas minoritas (Chawla et al., 2002). Penggunaan proses SMOTE pada bidang klasifikasi memberikann perbaikan performa dari metode klasifikasi yang digunakan (Siringoringo, 2018).

## 2.9 Cross Validation

*Cross validation* merupakan Teknik validasi *model macine learning* yang berguna dalam mengukur keefektifan dari suatu model *machine learning*. Teknik ini dilakukan dengan membagi dataset kedalam 2 bagian yaitu data pelatihan dan data pengujian (Darapureddy et al., 2019). Salah satu jenis *cross validation* adalah *k-fold cross validation*. *K-fold cross validation* digunakan untuk menghilangkan bias didalam data dengan pelatihan dan pengujian yang dilakukan secara berkali-kali sesuai jumlah *fold* yang ditetapkan (Tempola & Muhammad, 2017). Jumlah k-fold yang dianjurkan untuk digunakan dalam model klasifikasi adalah 10-fold (Kohavi, 1995).

## 2.10 Grid Search

Grid search merupakan metode dalam pengaturan parameter yang secara metodis digunakan untuk membangun dan mengevaluasi model untuk setiap kombinasi parameter algoritma yang ditentukan dalam grid (Ranjan et al., 2019). Metode grid search mencari nilai parameter yang terbaik dengan memberi range nilai pada parameter (Naufal et al., 2015) kedalam grid dan melintasi semua titik untuk mendapatkan parameter yang optimal. Dalam implementasinya, metode grid search harus diikuti oleh beberapa metrik kinerja, biasanya diukur dengan cross-validation (Mutmainnah & Widodo, 2018).

## 2.11 Confusion Matrix

*Confusion Matrix* merupakan literatur dari metode penilaian akurasi yang sering digunakan dalam penelitian yang memberikan dasar dalam menggambarkan akurasi dan mengkarakterisasi kesalahan dari klasifikasi, sehingga dapat membantu memperbaiki klasifikasi (Foody, 2002). didalam *Confusion Matrix* berisi nilai tentang klasifikasi aktual dan prediksi yang dilakukan dalam proses klasifikasi Untuk mengevaluasi Kinerja klasifikasi (Santra & Christy, 2012). Di antara kriteria evaluasi yang sering digunakan adalah kriteria yang dapat diturunkan dari *Confusion Matrix* seperti *accuracy*, *precision*, dan *recall* (Arisholm et al., 2010) .

Tabel 2. Tabel Confusion Matrix (Arisholm et al., 2010)

		<i>Actual Class</i>	
		<i>Positive</i>	<i>Negative</i>
<i>Predicted Class</i>	<i>Positive</i>	<i>True Positive (TP)</i>	<i>False Positive (FP)</i>
	<i>Negative</i>	<i>False Negative (FN)</i>	<i>True Negative (TN)</i>

Kriteria evaluasi model berdasarkan *confusion matix* (Arisholm et al., 2010) : *Accuracy* (AC) adalah presentasi dari jumlah total prediksi yang benar dan merupakan nilai akurat dari model klasifikasi yang digunakan. *Accuracy* dengan kelas lebih dari 2 memiliki persamaan (Purwiantono & Aditya, 2020):

$$Accuracy = \frac{TP}{N}$$

Dimana N adalah jumlah kelas yang digunakan.

*Recall* merupakan presentase berapa banyak kelas yang salah yang kemungkinan besar akan kita lewatkan jika kita menggunakan model prediksi (Santra & Christy, 2012). ditentukan dengan menggunakan persamaan:

$$Recall = \frac{(TP)}{(TP + FN)}$$

*Precision* merupakan proporsi kasus positif yang diprediksi dengan benar (Santra & Christy, 2012). ditentukan dengan menggunakan persamaan:

$$Presision = \frac{(TP)}{(TP + FP)}$$

## 2.12 Python

Bahasa pemrograman Python merupakan bahasa pemrograman yang bersifat open source. Python mendukung pemrograman berorientasi objek dan tersedia pada banyak sistem operasi seperti Windows, Linux, UNIX, Amigo, Mac OS, dll (Srinath, 2017). Bahasa pemrograman Python dioptimalkan pada *software quality*, *developer productivity*, *program portability*, dan *component integration*. Bahasa pemrograman ini telah banyak banyak digunakan dalam bidang *internet scripting*, *sistems programming*, *user interfaces*, *product customization*, *numeric programming* (Harismawan et al., 2018). Bahasa pemrograman Python termasuk kedalam bahasa pemrograman tingkat tinggi yang menjadi standar bahasa pemrograman dalam dunia komputasi ilmiah. Selain itu, Python merupakan bahasa pemrograman yang sangat fleksibel dan lebih mudah untuk dipelajari karena pada umumnya lebih mudah dibaca dan

jauh lebih sederhana (Herho, 2018). Bahasa pemrograman ini berkaitan erat dengan konsep perancangan sistem yang lebih fokus kepada tingkat keterbacaan kode program dengan kode yang lebih sedikit untuk pengembangan *web* daripada bahasa pemrograman lain (Aslam et al., 2015). Diklaim sebagai bahasa pemrograman yang menggabungkan kapabilitas, kemampuan, dan sintaksis kode yang sangat jelas, Python telah dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif (Jurnal Mitra Teknik Sipil, 2014). Menjadi bahasa pemrograman yang populer dalam bidang kerja dan akademik, Python menjadi bahasa pemrograman yang sering digunakan dalam bidang komputasi, *data science*, robotika dan dalam bidang lainnya. Terdapat beberapa *library* python yang digunakan dalam penelitian ini, sebagai berikut (Müller & Guido, 2017).

### 1. *Scikit-learn*

*Scikit-learn* merupakan *library* python yang berisi sejumlah algoritma yang digunakan dalam *Machine Learning*. Bersifat open source

### 2. *NumPy*

*Numpy* adalah *library* pada Python yang berfungsi untuk *array* multideminsi yang besar dan perhitungan matematika seperti aljabar linear. Data yang di gunakan dalam *scikit-learn* harus dikonversi ke *array numpy* agar dapat di proses.

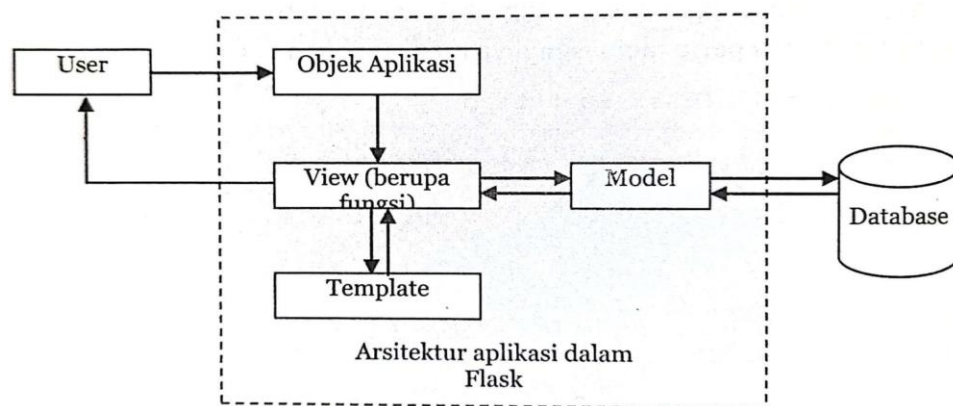
### 3. *Pandas*

*Pandas* merupakan *library* python yang berupa *dataframe*. *Dataframe* adalah sebuah *table* yang berbentuk seperti *spreadsheet excel* dimana setiap kolom dapat memiliki tipe data yang berbeda. Format file yang diterima *pandas* dapat berupa SQL, *Excel*, dan CSV.

## 2.13 *Flask*

*Flask* merupakan *framework* bahasa pemrograman Python dengan fungsi inti sederhana yang menyediakan fungsionalitas kerangka web yang memungkinkan plugin lebih ditambahkan sehingga fungsionalitas dan set fitur dapat diperluas (Aslam et al., 2015). Sama halnya dengan Python, *flask*

bersifat *open source* yang tersedia untuk umum. *Framework flask* memberikan koleksi dan perpustakaan kode yang digunakan dalam membangun web dengan fitur yang sederhana sehingga tidak perlu memulai membangun web dari awal dan akan lebih ringan (Mufid et al., 2019). Terdapat perbedaan antara *flask* dengan *framework* lain, pada *flask* tidak sepenuhnya menggunakan metode *Model View Controller (MVC)*. Dalam *flask* istilah *controller* tidak dikenal karena peran *controller* akan diwakili oleh objek dari kelas *flask* sendiri. Selain itu, istilah *view* pada *flask* menunjukkan ke fungsi yang menghasilkan tampilan (Raharjo, 2019). Arsitektur umum pada *flask* dapat di lihat pada Gambar 1.



Gambar 1 Arsitektur *Flask* (Raharjo, 2019)

## 2.14 UML

*Unified Modeling Language (UML)* merupakan bahasa dalam penulisan perangkat lunak yang digunakan dalam visualisasi, merancang, membangun, dan mendokumentasikan arsitek dari sistem yang berfokus pada struktur sistem perangkat lunak (Pressman & Maxim, 2020). Dengan adanya *UML*, analis dan calon pengguna sistem dapat memvisualisasikan dan memahami diagram yang digunakan dalam pengembangan sistem dengan mudah (Satzinger et al., 2016). Dalam membantu memvisualisasikan, menentukan, membuat dan mendokumentasikan rancangan sistem secara efektif yang memiliki manfaat bagi semua pemengembang aplikasi dapat menggunakan pendekatan *UML* sebagai solusi (Uke, 2013). Tujuan utama dari *UML* adalah mengeksplorasi desain-desain sistem dan menjaga komunikasi dalam tim pengembang sistem (Haviluddin, 2011).



### 1. *Use case diagram*

*Use case* menggambarkan interaksi antara sistem dan pengguna dari sudut pandang pengguna sistem. *Use case diagram* pada uml berfungsi sebagai gambaran umum fungsionalitas sistem. *Use case* akan dihubungkan langsung dengan aktor sesuai penggunaan yang dilakukan pengguna.

### 2. *Activity Diagram*

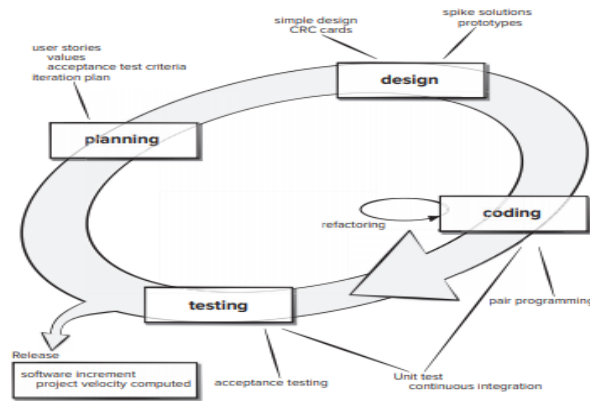
*Activity Diagram* menggambarkan alur aktivitas pengguna pada sistem secara berurutan. *Activity Diagram* memodelkan langkah-langkah alur kerja proses bisnis dalam aktivitas tertentu.

## **2.15 Balsamiq Mockup**

Balsamiq adalah *tools* perancangan *mockup* yang bersifat cloud yang dapat digunakan dalam sistem operasi windows, mac Os, dan linux (Putu, 2016). Balsamiq *Mockup* atau *wireframing* merupakan sketsa dalam pembuatan antarmuka web dalam bentuk kerangka dengan penggambaran menggunakan perangkat lunak. Dengan *tools* yang sederhana, balsamiq dapat dipahami dengan mudah sehingga pengguna dapat dengan mudah menggunakan. Kerangka kerja balsamiq dapat dibagikan dengan pengguna lain melalui email atau online (Faranello, 2012).

## **2.16 Extreme Programming**

Metode *Extreme Programming* (XP) merupakan metode pengembangan sistem skala kecil hingga medium dalam menyelesaikan *requirement* yang sulit dimengerti ataupun berubah-ubah (Beck & Andres, 2004). XP dalam pengembangan perangkat lunak menggunakan sistem berorientasi objek yang memiliki 4 kerangka kegiatan pada XP (Pressman & Maxim, 2014) dapat dilihat pada Gambar 2.



Gambar 2 Kerangka *Extreme Programming* (Pressman & Maxim, 2014)

Berikut Penjelasan dari setiap tahapan dalam kerangka metode XP :

### 1. *Planning*

Pada tahap ini dilakukan pengumpulan kebutuhan dan perencanaan untuk proyek. pengembang mendeskripsikan fitur, output, dan fungsionalitas sistem yang diperlukan pada perangkat lunak.

### 2. *Design*

Desain sistem dirancangan secara sederhana dan fungsionalitas dengan pembuatan pemodelan sistem UML dengan *Use Case Diagram*, *Activity Diagram* (Supriyatna, 2018), serta pembuatan prototipe desain sebagai implementasi dan evaluasi dari sistem yang akan di buat. Desain dilakukan sebelum dan sesudah pengkodean sistem yang berarti desain terjadi terus menerus selama sistem dibangun

### 3. *Coding*

Tahap ini merupakan implementasi dari desain yang telah di buat kedalam kode progam (Supriyatna, 2018). Dalam penelitian ini kode program di tulis dalam bahasa pemrograman Python.

### 4. *Testing*

Pada tahap ini pengujian dilakukan terhadap sistem yang telah dibangun (Supriyatna, 2018). Pengujian berfokus pada keseluruhan fitur dan fungsi dari sistem yang dapat dilakukan setiap hari dengan tujuan memberikan peringatan terhadap kesalahan yang terjadi.

### 2.17 Black box Testing

*Black box testing* merupakan pengujian perangkat lunak yang digunakan untuk menentukan fungsionalitas sistem. Pengujian dengan *black box testing* berfokus pada *input*, *output* yang diinginkan dan tidak berkaitan dengan internal kode pada sistem ( Verma et al., 2017). perangkat lunak yang diuji dengan metode *black box testing* akan dijalankan dan dites untuk memenuhi kebutuhan pengguna yang telah didefinisikan pada rancangan awal tanpa harus membongkar kode program (Salamah & Khasanah, 2017). Salah satu Teknik pada metode *black box testing* adalah *Equivalence Partitioning*.

*Equivalence Partitioning* Teknik pengujian yang membagi data input dari perangkat lunak menjadi partisi-partisi dari kasus uji yang diturunkan. Kasus uji yang digunakan teknik ini berisi data actual yang dimasukkan kedalam perangkat dan memberikan hasil yang diharapkan (Wu, 2012). Teknik *Equivalence Partitioning* mendapatkan kesalahan fungsi yang hilang atau salah pada perangkat lunak, seperti kesalahan validasi data, kesalahan dalam struktur data atau akses menuju database dan kesalahan performa pada perangkat lunak (Agustian et al., 2020).

### 2.18 Penelitian Terdahulu

Terdapat penelitian terdahulu yang dijadikan sebagai acuan oleh penulis dalam penelitian ini. Daftar penelitian terdahulu disajikan pada Tabel 3.

Tabel 3. Penelitian Terdahulu

Judul	Fitur				Akurasi
	1	2	3	4	
Implementasi Sistem Informasi Seleksi Penerima Beasiswa Dengan Metode Naive Bayes <i>Classifier</i> (Rahman & Suryanto, 2017).	CRUD data siswa	Prediksi			90,33%

Tabel 3. Penelitian Terdahulu (Lanjutan)

Judul	Fitur				Akurasi
	1	2	3	4	
Aplikasi Klasifikasi Penentuan Pengajuan Kartu Kredit Menggunakan Metode Naive Bayes Di Bank Bni Syariah Surabaya (Antaristi & Kurniawan, 2017).	CRUD data <i>Testing</i>	CRUD data <i>Training</i>	Pengujian	Prediksi	98,67%
Implementasi Algoritma Naive Bayes Dalam Menentukan Konsentrasi Skripsi Dan Rekomendasi Bahasa Pemrograman (Pratama & Yulmaini, 2018)	Lihat data <i>training</i>	Akurasi	Pengujian		96,67%
Penerapan Metode Naive Bayes Classifier Untuk Penjurusan Siswa Pada Madrasah Aliyah Al-Falah Jakarta (Mafakhir & Solichin, 2020).	CRUD data <i>training</i>	Grafik	Prediksi		33,34%
Pengembangan Sistem Pendukung Keputusan Penjurusan Bagi Siswa Baru Menggunakan Metode Naive Bayes (Mufid Musthofa & Atequlis Syaifidin, 2015).	CRUD data siswa	CRUD data jurusan	Lihat nilai siswa	Prediksi	77,48%
Design dan Implementation of Movie Recommendation System Based on Naive Bayes (Shuxian & Sen, 2019)					72,34%
Comparative Study of K-NN, Naive Bayes and Decision Tree Classification Techniques Sayali (Jadhav & Channe, 2016)					

Tabel 3. Penelitian Terdahulu (Lanjutan)

Judul	Fitur				Akurasi
	1	2	3	4	
Research Of Machine Learning Algorithms Using K-fold Cross Validation (Darapureddy et al., 2019)					

Sedangkan untuk penelitian yang saat ini dilakukan oleh penulis disajikan pada Tabel 4.

Tabel 4. Penelitian Saat Ini

Judul	Fitur			Akurasi
	1	2	3	
Implementasi Algoritma Naïve Bayes Untuk Sistem Rekomendasi Topik Penelitian Pada Jurusan Ilmu Komputer Universitas Lampung Berbasis Klasifikasi	CRUD Data set mahasiswa wisuda	Lihat data kategori skripsi	Prediksi	-

### III. METODOLOGI PENELITIAN

#### 3.1 Tempat dan Waktu Penelitian

##### 3.1.1 Tempat Penelitian

Penelitian ini dilakukan pada semester genap tahun ajaran 2020/2021 bertempat di Jurusan Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung yang berada di Jalan Soemantri Brjonegoro No.1 Gedung Meneng, Bandar Lampung.

##### 3.1.2 Waktu penelitian

Penelitian dilakukan pada semester tujuh di bulan Desember 2020 hingga penyelesaian pada bulan Oktober 2021. Perencanaan waktu pada penelitian ini dimulai dari bulan Desember dengan studi literatur untuk memahami materi-materi terkait penelitian sekaligus pengajuan tema skripsi hingga bulan Januari. Kemudian pada bulan Februari dilakukan pengumpulan data yang diperoleh dari pengelola server siacad UPT TIK Universitas Lampung tahun 2016 – 2019 yang telah di *export* kedalam format *xls*. Setelah data penelitian diperoleh, dilakukan perancangan sistem rekomendasi topik penelitian pada bulan Maret s.d bulan April. Proses selanjutnya pada bulan Juni s.d Juli dilakukan pembangunan model *machine learning* dengan implementasi algoritma *naïve bayes*. Setelah proses pembangunan model *machine learning*, dilakukan pengembangan sistem rekomendasi topik penelitian pada bulan Agustus s.d September. Proses terakhir pada bulan Oktober yaitu melakukan proses pengujian sistem dengan metode *black box testing*. Penelitian ini juga memuat alokasi waktu penulisan laporan bab 1 s.d bab 3 pada bulan Januari sampai bulan April, dengan seminar usul pada bulan Mei. Waktu penulisan laporan bab 4-5 dilakukan pada bulan Juli sampai dengan bulan Oktober.

## 3.2 Data dan Alat Pendukung

Alat yang digunakan untuk mendukung dan menunjang dalam pelaksanaan penelitian, antara lain sebagai berikut :

### 3.2.1 Data

Data yang digunakan dalam penelitian ini berupa 240 data mahasiswa yang telah selesai pendidikan tinggi negeri jurusan ilmu komputer Universitas Lampung yang diperoleh dari pengelola server siacad UPT TIK tahun 2016 – 2019 yang telah di *export* kedalam format xls. Data terdiri dari NPM mahasiswa, 35 mata kuliah dari 68 mata kuliah wajib maupun pilihan, judul skripsi, dan kategori topik penelitian. Mata kuliah yang digunakan sebagai atribut merupakan mata kuliah yang terdiri dari mata kuliah wajib dan mata kuliah pilihan yang berkaitan dengan bidang Komputer. Terdapat 14 kategori penelitian yang digunakan sebagai label pada data.

### 3.2.2 Alat Pendukung

#### a. Perangkat keras

Penelitian ini dilakukan dengan menggunakan laptop acer dengan spesifikasi sebagai berikut :

- Sistem Model : A514-51G-52PZ
- Processor : Intel® Core i5-8265U 1.6Hz to 3.9Ghz
- Installed RAM : 4,00 GB

*Smartphone* android Samsung A50s dengan spesifikasi berikut:

- Sistem Model : SM-A507FN/DS
- GPU : Mali-G72 MP3
- Installed RAM : 4,00 GB

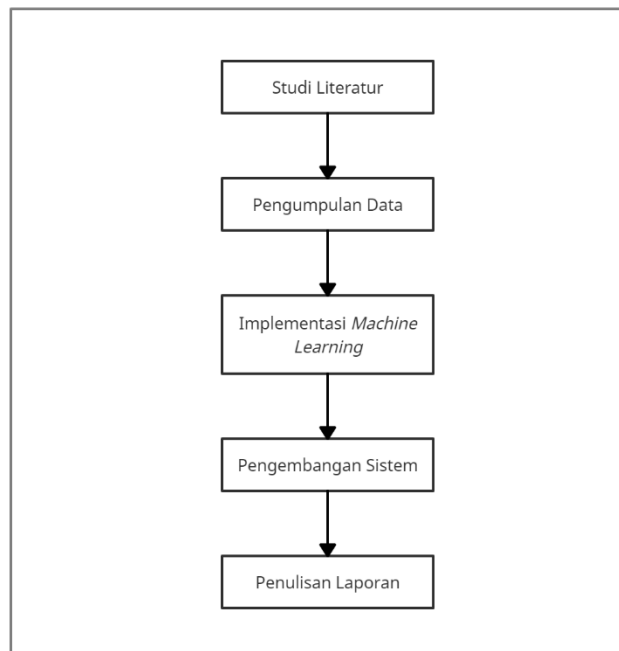
#### b. Perangkat lunak

Perangkat lunak yang digunakan dalam pembuatan system rekomendasi dalam penelitian ini adalah sebagai berikut :

- Sistem operasi Microsoft Windows 2010
- Notepad++
- Chrome
- XAMPP
- Start UML
- Jupiter Notebook
- Rapidminer
- Balsamiq

### 3.3 Tahapan Penelitian

Tahapan penelitian merupakan tahapan-tahapan yang akan dilakukan oleh peneliti dalam menyelesaikan penelitian. Terdapat beberapa tahapan yang akan dilakukan dalam membangun sistem rekomendasi topik penelitian yang disajikan pada Gambar 3.



Gambar 3. Tahapan Penelitian



### 3.3.1 Studi Literatur

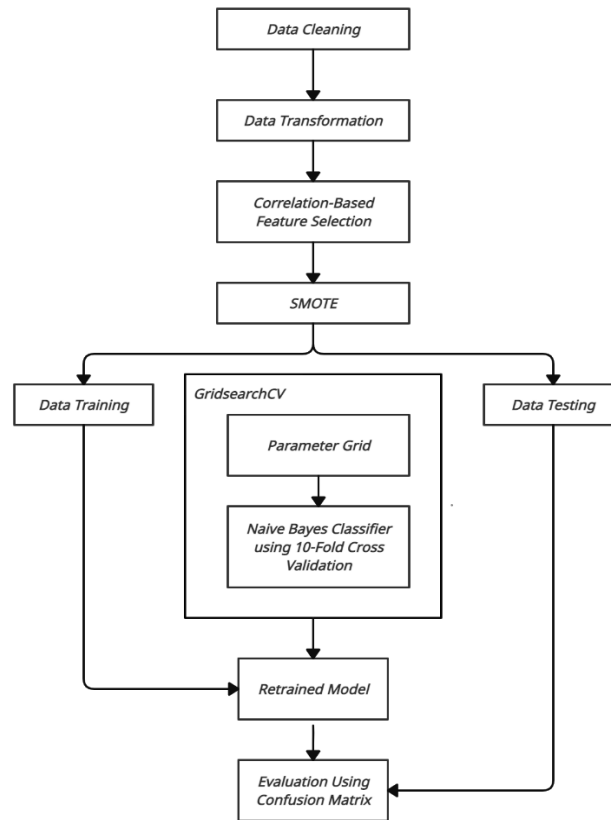
Tahapan pertama yang dilakukan adalah studi literatur. Pada tahap ini dilakukan dengan mencari dan mempelajari teori-teori yang terkait dengan algoritma naïve bayes, pemrograman Python, dan sistem rekomendasi. Studi literatur di dapat berdasarkan jurnal, artikel, dan buku.

### 3.3.2 Pengumpulan Data

Tahap kedua dari penelitian ini adalah pengumpulan data. Pengumpulan data dilakukan dengan berkonsultasi pada dosen dan sekretaris jurusan Ilmu Komputer Universitas Lampung dalam menentukan kategori topik penelitian di Jurusan Ilmu Komputer.

### 3.3.3 Implementasi *Machine Learning*

pada tahap ini merupakan proses pengkodean untuk model dari *machine learnin* dan disimpan dengan menggunakan *library pickle*. Model *machine learning* yang telah disimpan dengan format *pickle* dapat digunakan oleh sistem dengan perintah *read binary*. Tahapan proses dari model *Machine Learning* untuk system rekomendasi topik penelitian dapat di lihat pada Gambar 4.



Gambar 4 *Flowchart* Pengembangan Model *Machine Learning*

Berdasarkan *flowchart model machine learning* pada Gambar 15, berikut penjelasan setiap tahap yang diproses.

- *Data Cleaning*  
Proses menangani missing value dengan mengganti semua nilai atribut yang hilang dengan konstanta yang sama dan nilai yang paling mungkin untuk mengisi nilai yang hilang.
- *Data Transformation*  
Merubah data *string* menjadi data numerik agar dapat diproses dalam algoritma. *Transformation* dilakukan didalam pengkodean model *machine learning*.
- *Correlation-Based Feature Selection*  
Mendapatkan atribut yang tidak saling berkorelasi dan menghapus salah satu dari atribut yang memiliki koefisien korelasi yang kuat.

- **SMOTE**  
data kelas yang tidak seimbang diproses dengan metode smote sehingga data kelas/data target masing-masing memiliki jumlah data yang seimbang. Untuk menentukan jumlah  $k$  pada proses ini menggunakan pengujian perbandingan antara  $k = 1, k = 2, k = 3, k = 4$  dengan model klasifikasi
- **Pembagian Data**  
Pembagian data dilakukan dengan membagi data menjadi data *training* dan data *testing*. Data *training* digunakan untuk pelatihan model dan data *testing* digunakan untuk evaluasi model. Untuk menentukan jumlah perbandingan data *training* dan data *testing* dilakukan pengujian perbandingan antara *training:testing* sebesar 90:10, 80:20, 70:30, 60:40, 50:50.
- **Gridsearch**  
grid search digunakan untuk melatih beberapa model prediksi yang dibentuk melalui setiap kombinasi range parameter. Didalam grid search menggunakan *10-fold cross validation*. Pada setiap model dari kombinasi parameter yang terbentuk, dilakukan *cross-validation* untuk menghitung *score* rata-rata untuk semua lipatan (*fold*) yang diperoleh selama pelatihan model tersebut. Parameter yang memiliki *score* tertinggi dipilih sebagai parameter terbaik yang digunakan dalam pelatihan model prediksi.
- **Trained model**  
Tahapan ini dilaksanakan setelah tahap *grid search* selesai dilakukan. Pelatihan pada model prediksi menggunakan data *training*. Model prediksi naïve bayes dibangun menggunakan parameter optimal yang diperoleh dari tahap gridsearchCV dengan menggunakan *library* algoritma naïve bayes yang kemudian dijadikan sebagai model *machine learning* untuk sistem rekomendasi topik penelitian.

- *Evaluation*

*Evaluation* pada model menggunakan *Confusion Matrix* untuk mendapatkan akurasi, presisi, dan *recall* dari model classifier naïve bayes. Pengujian menggunakan data *testing* yang telah dibagi sebelum tahap gridsearchCV. Dengan confusion matrix, didapatkan nilai TP, FP, dan FN yang akan digunakan untuk menentukan nilai akurasi, presisi, dan *recall*.

### 3.3.4 Pengembangan Sistem

#### a. *Planning* (Perencanaan)

Di tahap ini menentukan kebutuhan-kebutuhan dalam membangun sistem rekomendasi, seperti fitur-fitur yang terdapat pada sistem, menentukan pengguna pada sistem, dan kebutuhan yang diperlukan pada sistem yang dirangkum pada *user stories*. Tabel 5. Menampilkan *user stories* pada sistem rekomendasi topik penelitian.

Tabel 5. *User Stories* Sistem

No.	Deskripsi fitur	Kode
1	Pengguna Admin mendapat akses untuk mengelola dan melatih ulang data mahasiswa	A
2	Pengguna mahasiswa dapat melihat seluruh judul skripsi yang sudah ada berdasarkan kategori topik penelitian sebagai referensi	B
3	Pengguna mahasiswa dapat melakukan prediksi topik penelitian berdasarkan nilai mata kuliah yang di inputkan	C

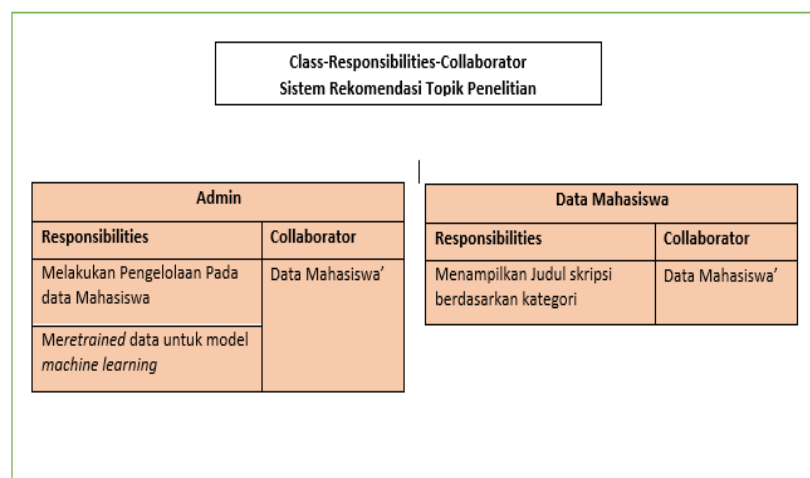
#### b. *Design* (Perancangan)

Tahap ini menampilkan rancangan dari sistem yang akan di bangun dalam bentuk desain yang bertujuan untuk memudahkan pengguna dalam melihat sistem yang akan bangun. Penelitian ini menggunakan desain sistem dengan diagram uml diantaranya *use case diagram*, *Activity Diagram* dan menggunakan perancangan

antarmuka sistem dengan mockup. Metode extreme programming mendorong pengembangan sistem menggunakan Class-Responsibility Collaborator untuk digunakan dalam mendeskripsikan kelas-kelas yang digunakan beserta fungsionalitas yang dibutuhkan dan hubungannya dengan kelas lain

- Class-ResponsibilityCollaborator

CRC pada sistem rekomendasi topik penelitian dapat dilihat pada Gambar 5.

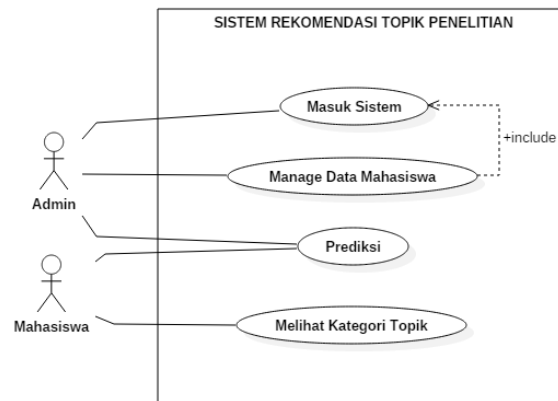


Gambar 5. Class Responsibility Collaborator

- Perancangan UML  
Perancangan ditampilkan pada diagram-diagram uml sebagai berikut.

- a) *Use case diagram*

*Use case diagram* menggambarkan fungsionalitas sistem dari pengguna. Pada sistem ini, terdapat 2 pengguna yaitu admin dan mahasiswa. hanya admin yang dapat login ke dalam sistem, dan mengelola data mahasiswa. Mahasiswa hanya dapat melakukan 2 interaksi antara lain prediksi dan melihat kategori topik. Desain use case diagram dapat dilihat pada Gambar 6.

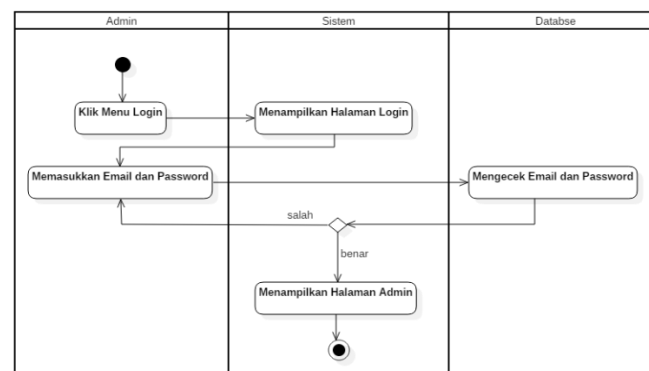


Gambar 6 Use Case Diagram Sistem

b) *Activity Diagram*

## 1) Login

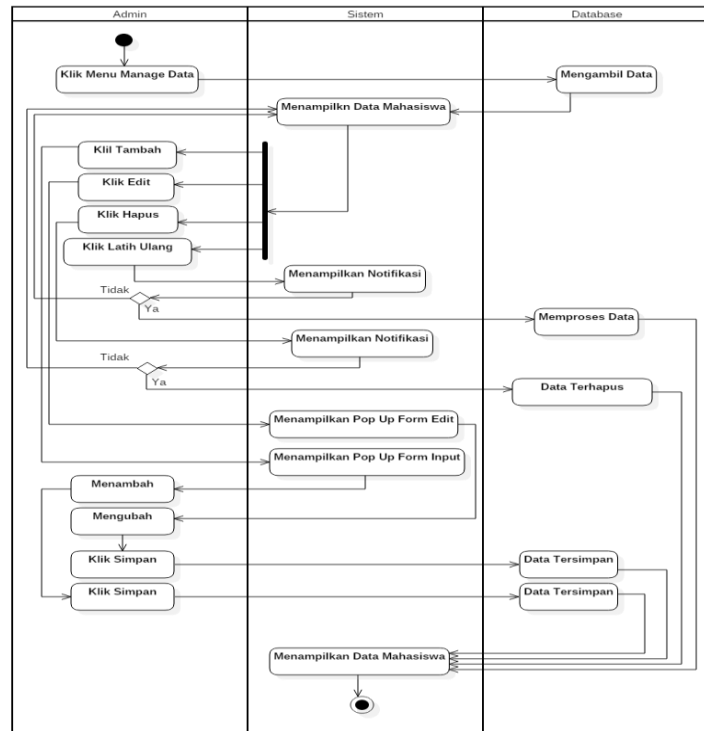
Pada *Activity Diagram Login* menggambarkan alur untuk masuk kedalam sistem sebagai admin dan ditunjukkan pada Gambar 7.



Gambar 7 Activity Diagram Login

2) *Manage data mahasiswa*

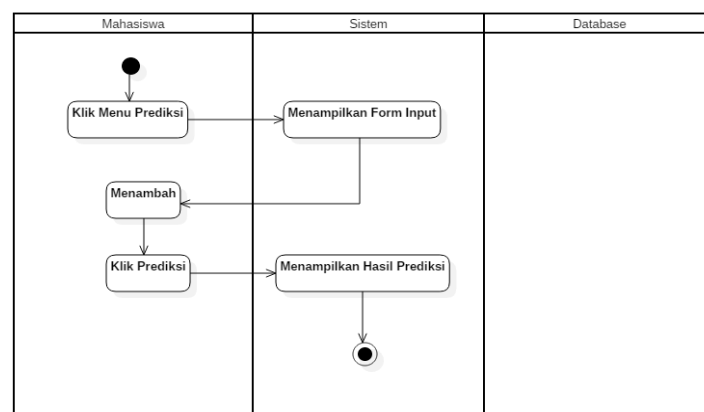
Pada *Activity Diagram manage data mahasiswa* menggambarkan alur dari proses pengelolaan data mahasiswa yang digunakan sebagai data latih dan ditunjukkan pada Gambar 8.



Gambar 8 Activity Manage Data Mahasiswa

### 3) Prediksi

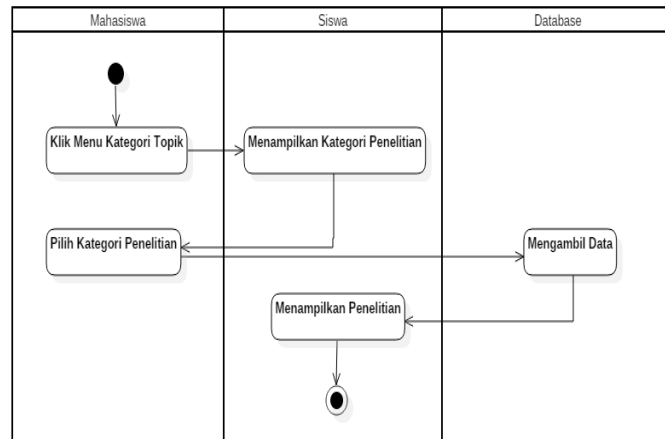
Pada *Activity Diagram* Prediksi menggambarkan alur dari proses fungsi prediksi dan ditunjukkan pada Gambar 9.



Gambar 9 Activity Diagram Prediksi

#### 4) Lihat Kategori

Pada *Activity Diagram* Lihat kategori menggambarkan alur dari fungsi lihat kategori pada sistem dan ditunjukkan pada Gambar 10.



Gambar 10 *Activity Diagram* Lihat Kategori

- Perancangan Antarmuka

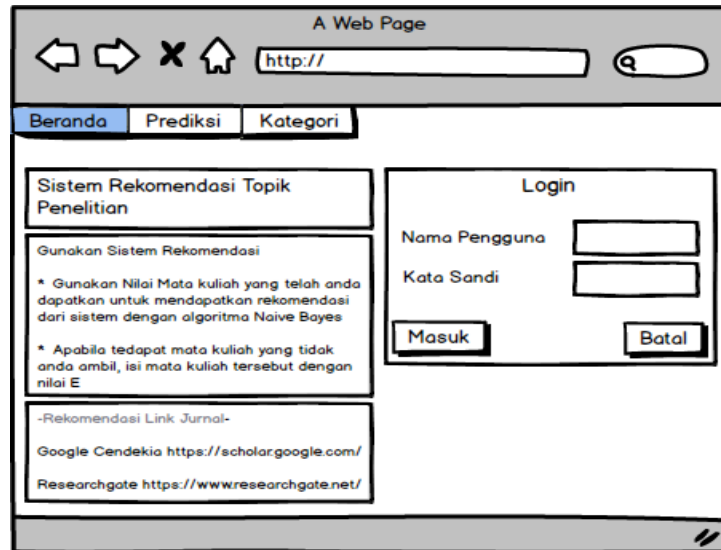
Rancangan Antarmuka merupakan tampilan desain sistem yang akan dibuat sebagai interaksi antara pengguna dan sistem. Rancangan antarmuka berupa tampilan output dan input data. Berikut merupakan rancangan antarmuka pada sistem rekomendasi topik penelitian yang dibuat.

##### a) Admin

##### 1) Antarmuka Login

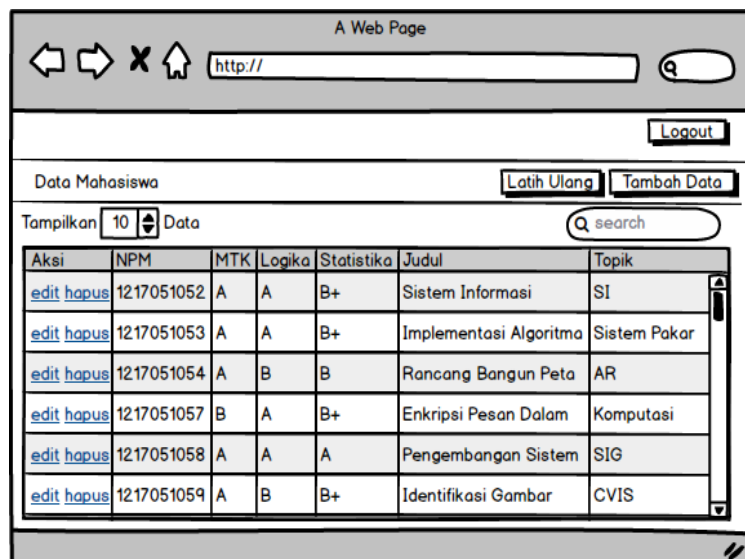
Pada antarmuka ini menampilkan halaman login untuk admin dengan memasukkan email dan password. Antarmuka login ditampilkan pada Gambar 11.



Gambar 11 Antarmuka *Login*

## 2) Manage Data Mahasiswa

Pada antarmuka ini menampilkan data mahasiswa yang digunakan sebagai data latih dan ditampilkan pada Gambar 12.

Gambar 12 Antarmuka *Manage Data Mahasiswa*

### 3) Tambah Data

Antarmuka ini menampilkan form input data mahasiswa yang akan digunakan dalam data latih sistem. Ditampilkan pada Gambar 13.

Tambah Data			
Logika	Matematika	Statistika	Sistem Pakar
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Multimedia	Basis Data	Mobile	Sistem Operasi
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Struktur Data	Aljabar Linear	Data Warehouse	Analisis Numerik
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Komdat	ADSI	DAA	AI
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Batal Simpan

Gambar 13 Antarmuka Tambah Data

### 4) Edit Data

Antarmuka ini menampilkan *form* edit data mahasiswa apabila terdapat nilai yang akan diubah. Ditampilkan pada Gambar 14.

Edit Data			
Logika	Matematika	Statistika	Sistem Pakar
<input type="text" value="A"/>	<input type="text" value="A"/>	<input type="text" value="B+"/>	<input type="text" value="C"/>
Multimedia	Basis Data	Mobile	Sistem Operasi
<input type="text" value="B"/>	<input type="text" value="B"/>	<input type="text" value="A"/>	<input type="text" value="B"/>
Struktur Data	Aljabar Linear	Data Warehouse	Analisis Numerik
<input type="text" value="B+"/>	<input type="text" value="C+"/>	<input type="text" value="A"/>	<input type="text" value="C"/>
Komdat	ADSI	DAA	AI
<input type="text" value="B+"/>	<input type="text" value="B"/>	<input type="text" value="A"/>	<input type="text" value="B+"/>

Batal Simpan

Gambar 14 Antarmuka Edit Data

## b) Mahasiswa

## 1) Prediksi

Antarmuka ini menampilkan *form* input data yang akan digunakan dalam prediksi. Mahasiswa dapat menginputkan nilai-nilai mata kuliah yang ada pada *form*. Ditampilkan pada Gambar 15

Gambar 15 Antarmuka Halaman Prediksi

## 2) Lihat Kategori Topik

Antarmuka ini menampilkan judul-judul skripsi yang telah ada berdasarkan topik penelitian. Ditampilkan pada Gambar 16.

Gambar 16 Antarmuka Lihat Kategori


c. *Coding* (Pengkodean)

Pada tahap ini, desain yang telah di buat pada tahap kedua akan diimplementasikan ke dalam kode program dengan menggunakan *framework flask* dan bahasa pemrograman Python.


d. *Testing* (Pengujian Sistem)

Pengujian sistem dilakukan setelah tahap coding selesai dilakukan dengan menggunakan *black box testing* atau pengujian fungsional pada setiap fungsi sistem rekomendasi untuk mengetahui kekurangan pada sistem. Pengujian sistem menggunakan teknik *Equivalence Partitioning* dengan membagi kedalam beberapa kasus uji. Skenario pengujian sistem dengan Teknik *Equivalence Partitioning* dapat dilihat pada tabel 6.

Tabel 6. *Black Box Testing Scenario* Sistem Rekomendasi Topik Penelitian

No	Kelas Uji	Kasus Uji	Hasil Yang Diharapkan
1	Menambah data mahasiswa	Klik tombol tambah data	Menampilkan modal <i>form</i> tambah data
		Menambahkan data mahasiswa dengan mengisi field required sesuai format kemudian klik tombol simpan	Sistem berhasil menambah data, menyimpan data, dan menampilkan notifikasi penambahan data telah berhasil dilakukan.
		Memeriksa respon sistem Ketika mengisi <i>form</i> dengan mengosongkan <i>field required</i>	Sistem Menampilkan pesan <i>field required</i> harus diisi pada <i>form</i>
		Mengisi <i>field</i> npm dengan format huruf	Menolak data dan menampilkan pesan error
		Mengisi <i>field</i> npm lebih dari 10 karakter	Sistem menolak data dan menampilkan contoh pengisian <i>field</i> npm yang benar
		Klik batal atau “X” pada modal	Sistem mengarahkan pengguna Kembali ke halaman pengelolaan data mahasiswa
2	Mengubah data mahasiswa	Klik <i>icon</i>  Pada kolom aksi	Menampilkan modal <i>form</i> edit data

Tabel 6. *Black Box Testing Scenario* Sistem Rekomendasi Topik Penelitian (Lanjutan)

No	Kelas Uji	Kasus Uji	Hasil Yang Diharapkan
		Mengubah data dengan mengubah satu atau lebih <i>field</i> pada <i>form</i> modal edit kemudian klik simpan	Sistem dapat mengubah data, menyimpan data dan menampilkan notifikasi bahwa data berhasil diperbaharui
		Mengubah data dengan mengosongkan satu atau lebih <i>field required</i> kemudian klik simpan	Sistem menampilkan pesan <i>field required</i> harus diisi
		Klik batal atau “X” pada modal	Sistem mengarahkan pengguna Kembali ke halaman pengelolaan data mahasiswa
3	Menghapus data mahasiswa	Klik <i>icon</i>  Pada kolom aksi	Sistem menampilkan <i>Pop up</i> konfirmasi hapus untuk data yang dipilih
		Klik tombol Ya pada <i>pop up</i> hapus	Sistem berhasil menghapus data yang telah dipilih dan sistem menampilkan notifikasi bahwa data telah berhasil dihapus, kemudian Kembali ke halaman pengelolaan data mahasiswa
		Klik tombol batal pada <i>pop up</i> hapus	Sistem menampilkan Kembali halaman pengelolaan data mahasiswa
4	Latih ulang data	Klik tombol latih ulang	Sistem menampilkan <i>Pop up</i> konfirmasi latih ulang data
		Klik tombol “Ya” pada <i>pop up</i> latih ulang	Sistem berhasil melatih ulang data dengan membuat file model baru dan menampilkan notifikasi bahwa data berhasil dilatih ulang
		Klik tombol batal pada <i>pop up</i> latih ulang	Sistem menampilkan Kembali halaman pengelolaan data mahasiswa
5	Pencarian data pada tabel data mahasiswa	Mengetik pada kolom pencarian	Menampilkan data berdasarkan <i>input</i> yang diketik pada kolom pencarian
6	Halaman prediksi	Klik Prediksi pada navbar	Sistem menampilkan <i>form</i> yang berisi inputan matakuliah
		Melakukan prediksi dengan memilih nilai mata kuliah pada <i>field required</i> dan klik tombol prediksi	Sistem berhasil memberikan rekomendasi topik penelitian dengan menampilkan <i>pop up</i> yang berisi kategori topik penelitian

Tabel 6. *Black Box Testing Scenario* Sistem Rekomendasi Topik Penelitian (Lanjutan)

No	Kelas Uji	Kasus Uji	Hasil Yang Diharapkan
7	<i>login</i>	Melakukan prediksi dengan memilih nilai mata kuliah pada <i>field required</i> dan klik tombol batal	Sistem mengosongkan seluruh <i>field</i> yang telah diisi
		Melakukan prediksi dengan Mengosongkan <i>field required</i>	Sistem menampilkan pesan <i>field required</i> harus diisi
		Masuk sebagai admin dengan mengisi <i>field required</i> sesuai dengan nama pengguna dan kata sandi kemudian klik masuk	Berhasil <i>login</i> , sistem menampilkan notifikasi bahwa <i>login</i> berhasil dan mengarahkan ke halaman pengelolaan data mahasiswa
		Masuk sebagai admin dengan mengisi <i>field required</i> kemudian klik batal	sistem mengosongkan <i>form login</i>
		Masuk sebagai admin dengan mengisi <i>field required</i> yang tidak sesuai antara nama pengguna dan kata sandi kemudian klik masuk	Sistem menampilkan pesan bahwa nama pengguna atau kata sandi salah
8	Halaman Kategori	Memeriksa respon sistem ketika mengisi <i>form login</i> dan Mengosongkan <i>field required</i>	Sistem Menampilkan pesan <i>field required</i> harus diisi pada <i>form</i>
		Klik kategori pada navbar	<i>Form</i> menampilkan 14 kategori topik skripsi dan seluruh data judul skripsi
		Melihat judul berdasarkan kategori dengan klik kategori topik penelitian	Sistem menampilkan judul berdasarkan kategori topik yang dipilih

### 3.3.5 Penulisan Laporan

Penulisan laporan bertujuan untuk mendokumentasikan penelitian dalam bentuk laporan yang dilakukan apabila semua tahap telah dilakukan dan semua fungsi pada sistem rekomendasi dapat berjalan dengan baik. Penulisan laporan membahas tahap-tahap dari penelitian dan memberikan kesimpulan dari penelitian yang dilakukan.

## V. SIMPULAN DAN SARAN

### 5.1 Simpulan

Dari hasil dan implementasi penelitian yang telah dilakukan, didapatkan simpulan sebagai berikut:

1. Penelitian memperoleh hasil klasifikasi dengan menggunakan metode naïve bayes dan parameter terbaik yaitu Alpha pada nilai **0,0001** dari proses *grid search* dengan menggunakan data mahasiswa yang telah melalui proses pembersihan data, penyeimbangan data, dan pembagian data. Proses pembersihan data terdiri dari tiga tahap yaitu *Data Cleaning*, *Correlation Based Feature Selection*, dan *Data Transformation*. Proses penyeimbangan data menggunakan SMOTE. Proses pembagian data dilakukan dengan aturan data *training* 90% dan data *testing* 10% .
2. hasil pengujian klasifikasi naïve bayes dengan menggunakan metode *Confusion Matrix* mendapatkan hasil rata-rata *accuracy* sebesar 0,68 hasil rata-rata *recall* 0,68 dan hasil rata-rata *precision* 0,69
3. Sistem rekomendasi topik penelitian dengan mengimplementasikan metode naïve bayes telah berhasil dikembangkan dengan *framework* Flask
4. Sistem yang dibuat telah berhasil memberikan rekomendasi topik penelitian kepada mahasiswa dengan menggunakan nilai mata kuliah yang telah didapatkan selama kuliah dan sistem dapat mengelola data mahasiswa yang mencakup penambahan data, pengeditan data, penghapusan data, serta pelatihan ulang data mahasiswa dengan model *Machine Learning* metode klasifikasi Naïve Bayes.
5. Pengujian *black box* dengan Teknik *Equivalence Partitioning* pada sistem rekomendasi topik penelitian telah mendapatkan hasil sesuai dengan yang telah diharapkan pada tiap kasus uji.

## 5.2 Saran

Adapun saran yang dapat diberikan pada penelitian ini agar pengembangan sistem rekomendasi dapat lebih baik lagi diajukan oleh penguji dan penulis. Beberapa saran tersebut adalah sebagai berikut :

1. Data yang digunakan untuk penelitian selanjutnya agar dapat dilakukan penambahan, guna mendapatkan data yang seimbang dan nilai akurasi yang jauh lebih baik.
2. Penelitian selanjutnya dapat menggunakan metode klasifikasi selain naïve bayes agar dapat dibandingkan dengan nilai akurasi penelitian ini.
3. Menambahkan fitur Riwayat prediksi, fitur filter mata kuliah yang diambil, dan menampilkan judul sesuai hasil prediksi pada halaman prediksi, serta menerapkan *lazy load* agar tidak lama dalam proses menampilkan data
4. Menggunakan nilai selain minimum untuk mengganti nilai *missing value*
5. Pada penelitian selanjutnya membangun sistem berbasis Mobile untuk sistem rekomendasi topik penelitian



## DAFTAR PUSTAKA

- Agarwal, S. (2014). Data mining: Data mining concepts and techniques. In *Proceedings - 2013 International Conference on Machine Intelligence Research and Advancement, ICMIRA 2013*. <https://doi.org/10.1109/ICMIRA.2013.45>
- Agustian, A., Andryani, I., Khoerunisa, S., Pangestu, A., & Saifudin, A. (2020). Implementasi Teknik Equivalence Partitioning pada Pengujian Aplikasi E-learning Berbasis Web. *Jurnal Teknologi Sistem Informasi Dan Aplikasi*, 3(3), 178. <https://doi.org/10.32493/jtsi.v3i3.5371>
- Annur, H. (2018). Klasifikasi Masyarakat Miskin Menggunakan Metode Naive Bayes. *ILKOM Jurnal Ilmiah*, 10(2), 160–165. <https://doi.org/10.33096/ilkom.v10i2.303.160-165>
- Antaristi, M., & Kurniawan, Y. I. (2017). Aplikasi Klasifikasi Penentuan Pengajuan Kartu Kredit Menggunakan Metode Naive Bayes di Bank BNI Syariah Surabaya. *Jurnal Teknik Elektro*, 9(2), 45–52. <https://doi.org/10.15294/jte.v9i2.12496>
- Aprilia, R. I., & Fachrurrozi, M. (2016). Sistem Rekomendasi Bacaan Tugas Akhir Jurusan Teknik Informatika Universitas Sriwijaya menggunakan Metode Collaborative Filtering dan Naive Bayes. *Prosiding ANNUAL RESEARCH SEMINAR*, 2(1), 343–347.
- Arisholm, E., Briand, L. C., & Johannessen, E. B. (2010). A systematic and comprehensive investigation of methods to build and evaluate fault prediction models. *Journal of Systems and Software*, 83(1), 2–17. <https://doi.org/10.1016/j.jss.2009.06.055>
- Aslam, F. A., Mohammed, H. N., Musab, J., & Munir, M. (2015). Efficient Way Of Web Development Using Python And Flask. *International Journal of Advanced Research in Computer Science*, 6(2), 54–57.

- Astuti, T., & Ferinanto, T. (2016). *Diagnosis Penyakit Parkinson Berdasarkan Kombinasi*. 28–29.
- Beck, K., & Andres, C. (2004). *Extreme Programming Explained: Embrace Change, Second Edition* By Kent Beck, Cynthia Andres Publisher: Addison Wesley Professional Pub Date: November 16, 2004 ISBN: 0-321-27865-8 Table of Contents Pages: 224 (second). Addison Wesley Professional.
- Cai, J., Luo, J., Wang, S., & Yang, S. (2018). Feature selection in machine learning: A new perspective. *Neurocomputing*, 300, 70–79. <https://doi.org/10.1016/j.neucom.2017.11.077>
- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). Unfolding the effects of different forestry treatments on microclimate in oak forests: results of a 4-yr experiment. *Ecological Applications*, 30(2), 321–357. <https://doi.org/10.1002/eap.2043>
- Darapureddy, N., Karatapu, N., & Battula, T. K. (2019). *RESEARCH OF MACHINE LEARNING ALGORITHMS USING K-FOLD CROSS VALIDATION*. March.
- Devita, R. N., Herwanto, H. W., & Wibawa, A. P. (2018). Perbandingan Kinerja Metode Naive Bayes dan K-Nearest Neighbor untuk Klasifikasi Artikel Berbahasa Indonesia. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 5(4), 427. <https://doi.org/10.25126/jtiik.201854773>
- Faranello, S. (2012). *Balsamiq wireframes quickstart guide: wireframe like a pro, the easy way*.
- Foody, G. M. (2002). Status of land cover classification accuracy assessment. *Remote Sensing of Environment*, 80(1), 185–201. [https://doi.org/10.1016/S0034-4257\(01\)00295-4](https://doi.org/10.1016/S0034-4257(01)00295-4)
- Gopika, N., & Meena Kowshalaya, A. E. A. (2018). Correlation Based Feature Selection Algorithm for Machine Learning. *Proceedings of the 3rd International Conference on Communication and Electronics Systems, ICCES 2018, Icces*, 692–695. <https://doi.org/10.1109/CESYS.2018.8723980>
- Hall, M. A. (1999). *Correlation-based Feature Selection for Machine Learning*. April.

- Harismawan, A. F., Kharisma, A. P., & Afirianto, T. (2018). Analisis Perbandingan Performa Web Service Menggunakan Bahasa Pemrograman Python , PHP , dan Perl pada Client Berbasis Android. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya*, 2(January), 237–245. <http://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/781>
- Haviluddin. (2011). Memahami Penggunaan UML ( Unified Modelling Language ). *Jurnal Informatika Mulawarman*, 6(1), 1–15. <https://informatikamulawarman.files.wordpress.com/2011/10/01-jurnal-informatika-mulawarman-feb-2011.pdf>
- Hendrian, S. (2018). Algoritma Klasifikasi Data Mining Untuk Memprediksi Siswa Dalam Memperoleh Bantuan Dana Pendidikan. *Faktor Exacta*, 11(3), 266–274. <https://doi.org/10.30998/faktorexacta.v11i3.2777>
- Herho, S. H. S. (2018). *Tutorial Pemrograman Python 2 Untuk Pemula*. <https://doi.org/10.31227/osf.io/bau26>
- HERNÁNDEZ, J. (2019). *Tactic 03. Hyperparameter optimization. GNB*. <https://www.kaggle.com/>. <https://www.kaggle.com/juanmah/tactic-03-hyperparameter-optimization-gnb>
- ICHI.PRO. (2020). *Mengapa & Bagaimana menggunakan algoritma Naive Bayes dalam industri yang diatur dengan sklearn. ICHI.PRO*. <https://ichi.pro/id/mengapa-bagaimana-menggunakan-algoritma-naive-bayes-dalam-industri-yang-diatur-dengan-sklearn-python-kode-241721569620687>
- Jadhav, S. D., & Channe, H. P. (2016). Comparative Study of K-NN, Naive Bayes and Decision Tree Classification Techniques. *International Journal of Science and Research (IJSR)*, 5(1), 1842–1845. <https://doi.org/10.21275/v5i1.nov153131>
- Jurnal Mitra Teknik Sipil, J. (2014). Python Fundamental For Programmer. In *JMTS: Jurnal Mitra Teknik Sipil* (Vol. 2, Issue 4). <https://doi.org/10.24912/jmts.v2i4.7885>
- Kohavi, R. (1995). A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. *International Joint Conference of Artificial Intelligence, March 2001*.

- Larose, D. T. (2006). *DATA MINING METHODS AND MODELS* (J. Wiley & Sons (eds.)). simultaneously in Canada.
- Lavindi, E. E., Wijanarto, W., & Rohmani, A. (2019). Aplikasi Hybrid Filtering Dan Naïve Bayes Untuk Sistem Rekomendasi Pembelian Laptop. *JOINS (Journal of Information System)*, 4(1), 54–64. <https://doi.org/10.33633/joins.v4i1.2518>
- Mafakhir, A. Z., & Solichin, A. (2020). Penerapan Metode Naïve Bayes Classifier Untuk Penjurusan Siswa Pada Madrasah Aliyah Al-Falah Jakarta. *Fountain of Informatics Journal*, 5(1), 21. <https://doi.org/10.21111/fij.v5i1.4007>
- Mardi, Y. (2017). Data Mining : Klasifikasi Menggunakan Algoritma C4 . 5. *Jurnal Edik Informatika*.
- Mufid, M. R., Basofi, A., & Udin, M. (2019). Design an MVC Model using Python for Flask Framework Development. *International Electronics Symposium (IES), Mvc*, 214–219.
- Mufid Musthofa, Y. W., & atequlis Syaifidin, M. A. (2015). Pengembangan sistem pendukung keputusan penjurusan bagi siswa baru menggunakan metode naive bayes. *Teknik Informatika, Teknologi Informasi, Polteknik Negeri Malang*.
- Müller, A. C., & Guido, S. (2017). Introduction to with Python Learning Machine. In *Proceedings of the Speciality Conference on Infrastructure Condition Assesmenr: Art, Science, Practice*.
- Mutmainnah, A., & Widodo, E. (2018). Application of Support Vector Machine (SVM) Methods on Stock Price Forecasting of PT Telekomunikasi Indonesia Tbk. *Seminar Nasional Edusainstek FMIPA UNIMUS 2018*, 50–60.
- Naufal, A. R., Satria, R., & Syukur, A. (2015). Penerapan Bootstrapping untuk Ketidakseimbangan Kelas dan Weighted Information Gain untuk Feature Selection pada Algoritma Support Vector Machine untuk Prediksi Loyalitas Pelanggan. *Journal of Intelligent Systems*, 1(2), 98–108.
- Nurvinda, G. (2021). *Pentingnya Preprocessing dalam Pengolahan Data Statistik*. [www.Dqlab.Id](http://www.Dqlab.Id). <https://www.dqlab.id/pentingnya-preprocessing-dalam-pengolahan-data-statistik>

- Pratama, T., & Yulmaini. (2018). Implementasi Algoritma Naive Bayes Dalam Menentukan Konsentrasi Skripsi Dan Rekomendasi Bahasa Pemrograman. *Jurnal Informatika*, 18(1), 1–13.
- Pressman, R. S., & Maxim, B. R. (2014). *Software engineering A Practitioner's Approach Eighth Edition*. McGraw-Hill Education.
- Pressman, R. S., & Maxim, B. R. (2020). *Software engineering A Practitioner's Approach Ninth Edition*. McGraw-Hill Education. <https://doi.org/10.1049/ic:20040411>
- Pujianto, U., Widiyaningtyas, T., Prasetya, D. D., & Romadhon, B. (2019). Penerapan algoritma naïve bayes classifier untuk klasifikasi judul skripsi dan tugas akhir berdasarkan Kelompok Bidang Keahlian. *Tekno*, 27(1), 79. <https://doi.org/10.17977/um034v27i1p79-92>
- Purwiantono, F. E., & Aditya, A. (2020). Klasifikasi Sentimen Sara, Hoaks Dan Radikal Pada Postingan Media Sosial Menggunakan Algoritma Naive Bayes Multinomial Text. *Jurnal Tekno Kompak*, 14(2), 68. <https://doi.org/10.33365/jtk.v14i2.709>
- Putu, I. K. R. (2016). Analisa Usability Pada Website Undiksha Dengan Menggunakan Metode Heuristic Evaluation. *KARMAPATI (Kumpulan Artikel Mahasiswa Pendidikan Teknik Informatika) ISSN: 2252-9063*, 5(2).
- Raharjo, B. (2019). *Kumpulan Solusi Pemrograman Python (Revisi)*. Informatika Bandung.
- Rahman, A., & Suryanto, A. (2017). Implementasi Sistem Informasi Seleksi Penerima Beasiswa Dengan Metode Naive Bayes Classifier. *Jurnal Penelitian Pendidikan Indonesia*, 2(3), 1–8.
- Ranjan, G. S. K., Kumar Verma, A., & Radhika, S. (2019). K-Nearest Neighbors and Grid Search CV Based Real Time Fault Monitoring System for Industries. *2019 IEEE 5th International Conference for Convergence in Technology, I2CT 2019*, 9–13. <https://doi.org/10.1109/I2CT45611.2019.9033691>

- Rofifah, D. (2020). IMPLEMENTASI ALGORITMA K-NEAREST NEIGHBOR DALAM PENGKLASIFIKASIAN FOLLOWER TWITTER YANG MENGGUNAKAN BAHASA INDONESIA. *Paper Knowledge . Toward a Media History of Documents*, 112, 12–26.
- Saifudin. (2013). Naïve Bayes Untuk Mendeteksi Gangguan Jaringan Komputer Dengan Seleksi Atribut Berbasis Korelasi. *Jurnal Bianglala Informatika*, 1(2), 1–11.
- Sajid, S., von Zernichow, B. M., Soylu, A., & Roman, D. (2019). Predictive Data Transformation Suggestions in Grafterizer Using Machine Learning. *Communications in Computer and Information Science*, 1057 CCIS(September), 137–149. [https://doi.org/10.1007/978-3-030-36599-8\\_12](https://doi.org/10.1007/978-3-030-36599-8_12)
- Salamah, U., & Khasanah, F. (2017). Pengujian Sistem Informasi Penjualan Undangan Pernikahan Online Berbasis Web Menggunakan Black Box Testing. *Information Management for Educators and Professionals*, 2(1), 35–46.
- Saleh, A. (2015). Implementasi Metode Klasifikasi Naïve Bayes Dalam Memprediksi Besarnya Penggunaan Listrik Rumah Tangga. *Creative Information Technology Journal*, 2(3), 207–217.
- Santra, A. K., & Christy, C. J. (2012). Genetic Algorithm and Confusion Matrix for Document Clustering. *International Journal of Computer Science Issues*, 9(1), 322–328.
- Sartika, D., & Indra, D. (2017). Perbandingan Algoritma Klasifikasi Naive Bayes, Nearest Neighbour, dan Decision Tree pada Studi Kasus Pengambilan Keputusan Pemilihan Pola Pakaian. *Jurnal Teknik Informatika Dan Sistem Informasi*, 1(2), 151–161.
- Satzinger, J. W., Jackson, R. B., & Burd, S. D. (2016). System Analysis and Design in a Changing World. In *In Course Technology Cengage Learning*.
- scikit-learn developers. (2021). *sklearn.naive\_bayes.CategoricalNB*. Scikit-Learn.Org.[https://scikitlearn.org/stable/modules/generated/sklearn.naive\\_bayes.CategoricalNB.html](https://scikitlearn.org/stable/modules/generated/sklearn.naive_bayes.CategoricalNB.html)

- Shuxian, L., & Sen, F. (2019a). Design and Implementation of Movie Recommendation System Based on Naive Bayes. *Journal of Physics: Conference Series*, 1345(4). <https://doi.org/10.1088/1742-6596/1345/4/042042>
- Shuxian, L., & Sen, F. (2019b). Design and Implementation of Movie Recommendation System Based on Naive Bayes. *Journal of Physics: Conference Series*, 1345(4), 1–8. <https://doi.org/10.1088/1742-6596/1345/4/042042>
- Siringoringo, R. (2018). Klasifikasi Data Tidak Seimbang Menggunakan Algoritma SMOTE dan k-Nearest Neighbor. *Jurnal ISD*, 3(1), 44–49.
- Srinath, K. R. (2017). Python – The Fastest Growing Programming Language. *International Research Journal of Engineering and Technology*, 354–357. [https://d1wqtxts1xzle7.cloudfront.net/55458585/IRJET-V4I1266.pdf?1515226715=&response-content-disposition=inline%3B+filename%3DPython\\_The\\_Fastest\\_Growing\\_Programming\\_L.pdf&Expires=1593202307&Signature=HBD7oa85wDxqRzTWX01uVRBIMacGX5mkGk1b~SVVTTkENJ6cf5diKz](https://d1wqtxts1xzle7.cloudfront.net/55458585/IRJET-V4I1266.pdf?1515226715=&response-content-disposition=inline%3B+filename%3DPython_The_Fastest_Growing_Programming_L.pdf&Expires=1593202307&Signature=HBD7oa85wDxqRzTWX01uVRBIMacGX5mkGk1b~SVVTTkENJ6cf5diKz)
- Sugiyono, P. D. (2016). *Metode Penelitian Kuantitatif, Kualitatif, Dan R&D*. ALFABETA cv.
- Supriyatna, A. (2018). Metode Extreme Programming Pada Pembangunan Web Aplikasi Seleksi Peserta Pelatihan Kerja. *Jurnal Teknik Informatika*, 11(1), 1–18. <https://doi.org/10.15408/jti.v11i1.6628>
- Sutoyo, E., & Fadlurrahman, M. A. (2020). Penerapan SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Television Advertisement Performance Rating Menggunakan Artificial Neural Network. *JEPIN (Jurnal Edukasi Dan Penelitian Informatika)*, 6(3), 379–385.
- Tempola, F., & Muhammad, M. (2017). Naive Bayes Classifier For Prediction Of Volcanic Status In Indonesia. *2018 5th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, 365–369.
- Uke, S. N. (2013). UML Modeling of Physical and Data Link Layer Security Attacks in WSN. *International Journal of Computer Applications*, 70(11), 25–28.

- Utomo, B. T. W., & Anggriawan, A. W. (2015). Sistem Rekomendasi Paket Wisata Se-Malang Raya Menggunakan Metode Hybrid Content Based Dan Collaborative. *Jurnal Ilmiah Teknologi Informasi Asia*, 9(1), 6–13.
- Verma, A., Khatana, A., & Chaudhary, S. (2017). A Comparative Study of Black Box Testing and White Box Testing. *International Journal of Computer Sciences and Engineering*, 5(12), 301–304. <https://doi.org/10.26438/ijcse/v5i12.301304>
- Wu, H. (2012). An effective equivalence partitioning method to design the test case of the WEB application. *2012 International Conference on Systems and Informatics, ICSAI 2012, Icsai*, 2524–2527. <https://doi.org/10.1109/ICSAI.2012.6223567>