

BAB II
TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Tabel 2. 1 Penelitian Terdahulu

Nama	Tahun	Judul	Metode Analisis	Hasil Analisis
Wahyuningsih, Budiman, Izzatul Umami	2022	Implementasi algoritma Naïve Bayes untuk menentukan calon penerima beasiswa di SMK YPM 14 Sumobito Jombang	Kuantitatif	Data calon penerima beasiswa SMK YPM 14 Sumobito Jombang dengan kriteria seperti, jumlah tanggungan anak, jarak rumah, pekerjaan orang tua, penghasilan orang tua, status anak, nilai dan sikap akan berpengaruh terhadap ketidaklayakan penerima beasiswa. Karena tidak semua calon penerima beasiswa bisa mendapatkannya. Hal yang sangat mempengaruhi tingkat ketidaklayakan sebuah bantuan beasiswa tersebut adalah kriteria berdasarkan jumlah tanggungan orang tua dan pekerjaan orang tua
Muhammad Husni Rifqo, Ardi Wijaya	2017	Implementasi Algoritma Naive Bayes Dalam Penentuan Pemberian	Kuantitatif	Model Naïve Bayes mampu menganalisa pelanggan yang baik dan buruk menggunakan data <i>Agiing leasing ACC</i>

		Kredit		maupun menggunakan data <i>credit approval</i> Negara Australia dan Japan dari UCI data set dengan tingkat akurasi yang baik. Kriteria yang digunakan yaitu jenis kelamin, keperluan pinjaman, jumlah pinjaman, angsuran, tenor, otr, nama pemilik kendaraan, penghasilan perbulan, status pekerjaan, lama bekerja / usaha, penghasilan per bulan penjamin / pasangan, status pekerjaan penjamin / pasangan, lama bekerja / usaha / penjamin / pasangan, status tempat tinggal, jumlah tanggungan, biaya pengeluaran bulanan, sumber order, dan status konsumen.
Amin Abdullah sidiq, Febrian Wahyu Christanto	2020	Algoritma Naive Bayes Untuk Penentuan PKH (Program Keluarga Harapan) Berbasis Sistem Pendukung Keputusan (Studi Kasus: Kelurahan Karanganyar Gunung Semarang)	Kuantitatif	Sistem dapat menentukan siapa yang berhak mendapatkan bantuan PKH dengan menggunakan algoritma Naive Bayes klasifikasi. Sehingga pihak kelurahan dapat menentukan dengan cepat warga yang berhak menerima bantuan PKH. kriteria-kriteria yang digunakan seperti status rumah, pekerjaan, jumlah

				tanggungan, bahan bakar untuk memasak, sumber air, dan daya listrik. Nilai dari kriteria akan menjadi patokan untuk penyeleksian penduduk yang menjadi prioritas utama untuk mendapatkan bantuan PKH (Program Keluarga Harapan).
--	--	--	--	--

2.2 Teori Terkait

2.2.1. Pengertian Beasiswa

Dalam Kamus Besar Bahasa Indonesia, beasiswa dimaknai sebagai tunjangan yang diberikan kepada pelajar atau mahasiswa sebagai bantuan biaya belajar. Beasiswa sering diberikan kepada pelajar atau mahasiswa yang memiliki prestasi dan juga diberikan untuk memberikan apresiasi terhadap pelajar yang berprestasi tetapi kurang mampu dalam bidang ekonomi untuk melanjutkan pendidikannya ke tingkat yang lebih tinggi. Beasiswa merupakan bantuan berbentuk dana yang diberikan oleh lembaga Pemerintah, perusahaan atau yayasan demi keberlangsungan pendidikan yang ditempuh oleh perorangan, atau pelajar.

2.2.2. Kartu Indonesia Pintar Kuliah

Menurut Kemendikbud Kartu Indonesia Pintar Kuliah atau KIP Kuliah adalah bantuan berupa uang tunai, perluasan akses, dan kesempatan belajar dari pemerintah dalam membantu siswa yang terkendala keterbatasan ekonomi terutama yang berprestasi untuk dapat melanjutkan studi ke jenjang perguruan tinggi. Dengan adanya KIP Kuliah, pelajar berekonomi rendah tidak perlu ragu untuk melanjutkan pendidikan ke perguruan tinggi. Ada beberapa manfaat yang didapatkan dari KIP Kuliah yaitu; pembebasan biaya pendaftaran perguruan tinggi, pembebasan biaya kuliah dan bantuan biaya hidup bulanan. Salah satu syarat penerima KIP Kuliah adalah siswa SMA atau sederajat yang akan lulus pada tahun berjalan atau lulus 2 (dua) tahun sebelumnya.

2.2.3. Algoritma Naïve Bayes

Algoritma Naive Bayes merupakan salah satu algoritma yang terdapat pada teknik klasifikasi. Naive Bayes merupakan pengklasifikasian dengan metode probabilitas dan statistik. Teorema Bayes dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya. Teorema tersebut dikombinasikan dengan Naive dimana diasumsikan kondisi antar atribut saling bebas. Naïve Bayes dinilai berpotensi baik dalam mengklasifikasikan dokumen dibandingkan dengan metode pengklasifikasian lain dalam hal akurasi dan efisiensi komputasi. Keuntungan dari Naive Bayes adalah menggunakan lebih sedikit data pelatihan, sehingga komputasi dapat dilakukan lebih cepat dan lebih efisien. Kelemahan Naive Bayes adalah pemilihan fitur yang buruk mengurangi akurasi dan meningkatkan waktu komputasi (Laurensz & Eko Sedyono, 2021). Adapun persamaan dari Teori Bayes sebagai berikut:

$$P(H|X) = \frac{P(X|H) \cdot P(H)}{P(X)}$$

Dimana :

X : Data dengan class yang belum diketahui

H : Hipotesis data merupakan suatu class yang spesifik

P (H|X) : Probabilistik hipotesis H berdasar kondisi X (posteriori probabilistik)

P (H) : Probabilistik hipotesis H (prior probabilistik)

P (X|H) : Probabilistik hipotesis X berdasar kondisi pada hipotesis H

P (X) : Probabilistik X

Dalam metode Naive Bayes diperlukan data latih dan data uji yang ingin diklasifikasikan, dalam Naive Bayes, semakin banyak data latih yang yang dilibatkan, semakin baik hasil yang prediksi yang diberikan. Menghitung $P(C_i)$ yang merupakan probabilitas prior untuk setiap sub kelas C yang akan dihasilkan menggunakan persamaan:

$$P(C_i) = \frac{s_i}{s}$$

Dimana s_i adalah jumlah data training dari kategori C_i , dan s adalah jumlah total data training. Menghitung $P(X_i|C_i)$ yang merupakan probabilitas posterior X_i dengan syarat C.

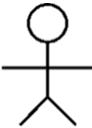
2.2.4. UML (Unified Modeling Language)

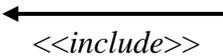
Menurut (Rosa & Shalahuddin, 2013) UML ialah Salah satu acuan bahasa yang dipakai di dunia industri untuk menjelaskan requirement, membuat analisa & perancangan arsitektur dalam pemrograman berorientasi objek. Di dalam UML seperti *use case* diagram, *activity* diagram, *class* diagram, *sequence* diagram. Seperti yang dikatakan oleh (Nugroho, 2010), UML ialah pemodelan untuk sebuah sistem atau perangkat lunak yang berorientasi objek. Pemodelan sesungguhnya dipakai untuk menyederhanakan masalah - masalah yang kompleksnya sedemikian rupa sehingga dapat dipahami dan mudah dipelajari. Berdasarkan beberapa pendapat yang dikemukakan diatas dapat ditarik kesimpulan bahwa “(UML) adalah sebuah bahasa yang berdasarkan analisa dan perancangan untuk membangun dari sebuah sistem pengembangan perangkat lunak berbasis OO (*Object Oriented*)”.

2.2.5. Use Case Diagram

Menurut Munawar (2018:89), *Use case* diagram adalah salah satu jenis diagram pada UML yang menggambarkan interaksi antara sistem dan actor. *Use case* diagram dapat mendeskripsikan atau menjelaskan interaksi yang terjadi antar *peserta* dalam sebuah sistem dengan sistemnya sendiri melalui sebuah alur atau cerita bagaimana sebuah sistem dipakai. Berikut Tabel 2.2 menjelaskan simbol simbol yang sering digunakan pada saat pembuatan *use case* diagram.

Tabel 2. 2 Simbolis *Use Case*

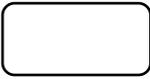
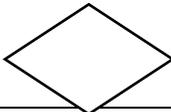
Lambang	Penjelasan
<i>Use case</i> 	Fungsional diberikan oleh sistem agar unit - unit saling bertukar pesan antar sistem dan actor.
Aktor atau <i>actor</i> 	Mewakili peran orang dari sistem yang memiliki interaksi dengan sistem informasi yang hendak dibuat di luar sistem informasi yang hendak dibuat sendiri.
Asosiasi atau <i>association</i> 	Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.
Ekstensi atau <i>extend</i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walaupun tanpa <i>use case</i> tambahan.
Generalisasi atau 	Hubungan generalisasi dan spesialisasi (umum - khusus) antara dua buah <i>use case</i> dimana fungsi yang umum dari

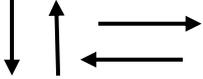
<i>generalization</i>	lainnya.
Menggunakan atau <i>include</i> 	Relasi <i>use case</i> tambahan ke <i>use case</i> dimana <i>use case</i> yang ditambahkan membutuhkan <i>use case</i> ini untuk menjalankan fungsinya.

2.2.6. Activity Diagram

Menurut Munawar (2018:127), *Activity* diagram adalah bagian penting dari UML yang menggambarkan aspek dinamis dari sistem. Berikut Tabel 2.3 menjelaskan simbol simbol yang sering digunakan dalam pembuatan *activity* diagram.

Tabel 2. 3 Simbolis *Activity* Diagram

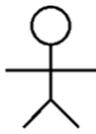
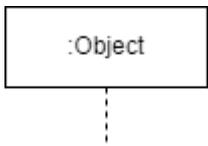
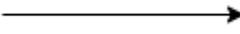
Lambang	Penjelasan
<i>Activity</i> 	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
<i>Action</i> 	State dari sistem yang mencerminkan eksekusi dari suatu aksi.
<i>Initial node</i> 	Bagaimana objek dibentuk atau diawali.
<i>Activity final node</i> 	Bagaimana objek dibentuk atau diakhiri.
<i>Decision</i> 	Digunakan untuk menggambarkan suatu keputusan / tindakan yang harus

	diambil pada kondisi tertentu.
<i>Line connector</i> 	Digunakan untuk menghubungkan satu simbol dengan simbol lainnya.

2.2.7. Sequence Diagram

Menurut Munawar (2018:138), *Sequence* diagram adalah salah satu jenis diagram pada UML yang menjelaskan interaksi objek yang berdasarkan urutan waktu. *Sequence* diagram dapat menunjukkan sejumlah objek atau pesan yang diletakkan diantara objek - objek di dalam use case diagram. Berikut Tabel 2.4 menjelaskan simbol - simbol yang sering digunakan pada saat pembuatan *sequence* diagram.

Tabel 2. 4 Simbolis *Sequence* Diagram

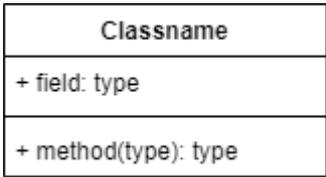
Lambang	Penjelasan
<i>Actor</i> 	Merupakan simbol yang mewakili peran orang dari suatu sistem.
<i>Objek</i> <i>Lifeline</i> 	Merupakan simbol entitas yang saling berinteraksi satu dengan lainnya.
<i>Synchronous</i> 	Merupakan pesan yang dikirimkan akan ditunggu sebelum diproses dengan urusannya.
<i>Sim</i> 	Perpindahan <i>control</i> dari satu <i>participant</i> ke <i>participant</i> yang lain.

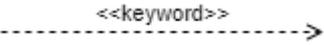
<i>ple</i>	
<i>Activation</i> 	Simbol yang menunjukkan objek melakukan sebuah aksi.

2.2.8. Class Diagram

Menurut Munawar (2018:101), *Class* diagram adalah salah satu jenis diagram pada UML yang digunakan untuk menampilkan kelas-kelas maupun paket-paket yang ada pada suatu sistem yang nantinya akan digunakan. *Class* diagram terdapat atribut, *operation* dan juga *constraint* yang terjadi pada sistem. Berikut Tabel 2.5 menjelaskan simbol – simbol yang sering digunakan dalam pembuatan *class* diagram.

Tabel 2. 5 Simbolis *Class* Diagram

Lambang	Penjelasan
<i>Class</i> 	Kumpulan objek yang memiliki <i>attribute</i> dan <i>operation</i> yang sama.
<i>Aggregation</i> 	Garis yang menangani berbagai objek, yang salah satunya bagian dari yang lain.

<p><i>Composition</i></p> 	<p>Sebuah tipe agregasi yang kuat dimana bagian objek tergantung pada keseluruhan objek.</p>
<p><i>Dependency</i></p> 	<p>Garis yang menunjukkan operasi pada suatu <i>class</i> yang menggunakan <i>class</i> lain.</p>
<p><i>Generalization</i></p> 	<p>Garis yang mewariskan atau menurunkan struktur data dan objek induk kepada objek anak yang dituju.</p>

2.2.9. *CodeIgniter*

Menurut Hakim (2010:8) *CodeIgniter* adalah sebuah framework PHP yang dapat membantu mempercepat developer dalam pengembangan aplikasi web berbasis PHP dibanding jika menulis semua kode program dari awal. Dibandingkan dengan framework web PHP lainnya, harus diakui bahwa *CodeIgniter* memiliki desain yang lebih sederhana dan bersifat fleksibel (tidak kaku). *CodeIgniter* mengizinkan para pengembang untuk menggunakan framework secara parsial atau secara keseluruhan. *CodeIgniter* merupakan sebuah toolkit yang ditujukan untuk orang yang ingin membangun aplikasi web dalam bahasa pemrograman PHP.

2.2.10. CDM (Conceptual Data Model)

Conceptual Data Model adalah model yang dibuat berdasarkan anggapan bahwa dunia nyata terdiri dari koleksi obyek-obyek dasar yang dinamakan entitas serta hubungan antara entitas-entitas itu. Biasanya CDM direpresentasikan dalam bentuk *Entity Relationship Diagram*. Adapun manfaat menggunakan CDM dalam perancangan database dapat memberikan gambaran yang lengkap mengenai struktur basis data yaitu arti, hubungan, dan batasan-batasan dalam memodelkan struktur logis dari keseluruhan aplikasi data, tidak tergantung pada software atau pertimbangan model struktur data serta menggambarkan secara detail struktur basis data dalam bentuk logik. (Shandy Ardianto,2012)

2.2.11. PDM (Physical Data Model)

Physical Data Model merupakan model yang menggunakan sejumlah tabel untuk menggambarkan data serta hubungan antara data-data tersebut. Setiap table mempunyai sejumlah kolom dimana setiap kolom memiliki nama yang unik dan merupakan representasi fisik dari database yang akan dibuat dengan mempertimbangkan DBMS yang akan digunakan. PDM dapat dihasilkan (generate) dari CDM yang valid. Dalam penerapannya dapat disamakan dengan skema relasi yang fungsinya adalah memodelkan struktur fisik dari suatu basis data yang merupakan gambaran secara detail suatu basis data dalam bentuk fisik serta memperlihatkan struktur penyimpanan data yang benar pada basis data yang digunakan sesungguhnya. (Shandy Ardianto, 2012).