

BAB II

LANDASAN TEORI

2.1 TINJAUAN STUDI

Adapun Penelitian terdahulu yang berhubungan dengan tema peneliti untuk dijadikan bahan referensi dalam menentukan metode yang akan digunakan nantinya .

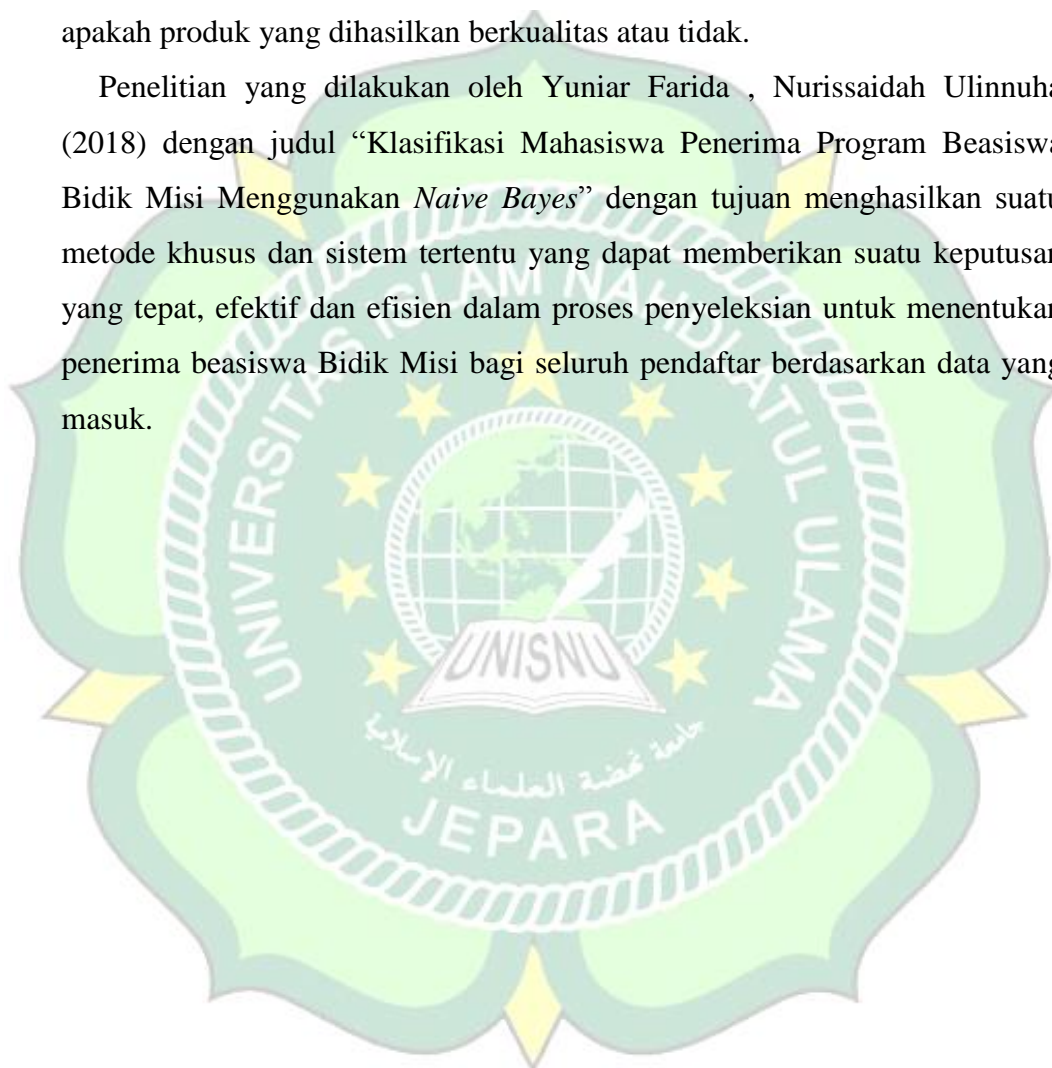
Penelitian yang dilakukan oleh Rizal Amegia Saputra Dan Shinta Ayuningtias (2016) dengan judul “Penerapan Algoritma Naive Bayes Untuk Penentuan Calon Penerima Beasiswa Pada SMK Pasim Plus Sukabumi” dengan tujuan menghasilkan sistem penentuan calon penerima beasiswa untuk mempermudah penentuan penerima beasiswa lebih efektif dan efisien.

Penelitian yang dilakukan oleh Diasrina Dahri, Fahrul Agus, Dyna Marisa Khairina (2016) dengan judul “Metode *Naive Bayes* Untuk Penentuan Penerima Beasiswa Bidikmisi Universitas Mulawarman”. Metode *Naive Bayes* merupakan sebuah pengklasifikasian probabilistik sederhana yang menghitung sekumpulan probabilitas dengan menjumlahkan frekuensi dan kombinasi nilai dari dataset yang diberikan.

Penelitian yang dilakukan oleh Febrian Anggoro Harimurti 2017 dengan judul “Klasifikasi Penerima Beasiswa Menggunakan Metode *Naive Bayes Classifier* (Studi Kasus Universitas Trunojoyo Madura). *Naive Bayes* didasarkan pada asumsi penyederhanaan bahwa nilai atribut secara kondisional saling bebas jika diberikan nilai output. maka dari itu, diberikan nilai output, probabilitas mengamati secara bersama merupakan produk dari probabilitas individu. Keuntungan menggunakan metode *Naive Bayes* adalah hanya membutuhkan jumlah data pelatihan (*Training Data*) yang kecil untuk menentukan estimasi parameter yang diperlukan dalam proses pengklasifikasian. *Naive Bayes* dapat bekerja jauh lebih baik dalam suatu situasi dunia nyata yang kompleks daripada yang diharapkan.

Penelitian yang dilakukan oleh Mochamad Ainun Rozaq, Nur Nafi'iyah, Masruroh (2019) dengan judul “Sistem Pendukung Keputusan Pemilihan Kualitas Songkok Berdasarkan Bahan Baku Menggunakan Metode *Naïve Bayes*” dengan tujuan menghasilkan suatu sistem yang dapat menentukan apakah produk yang dihasilkan berkualitas atau tidak.

Penelitian yang dilakukan oleh Yuniar Farida , Nurissaidah Ulinnuha (2018) dengan judul “Klasifikasi Mahasiswa Penerima Program Beasiswa Bidik Misi Menggunakan *Naive Bayes*” dengan tujuan menghasilkan suatu metode khusus dan sistem tertentu yang dapat memberikan suatu keputusan yang tepat, efektif dan efisien dalam proses penyeleksian untuk menentukan penerima beasiswa Bidik Misi bagi seluruh pendaftar berdasarkan data yang masuk.



2.2 TINJAUAN PUSTAKA

2.2.1 Sistem Pendukung Keputusan

Sistem pendukung keputusan sebagai sistem berbasis komputer yang terdiri dari tiga komponen yang saling berinteraksi, yaitu sistem bahasa, sistem pengetahuan dan sistem pemrosesan masalah.(Shella 2015)

Keuntungan dan Keterbatasan Sistem Pendukung Keputusan dapat memberikan berbagai manfaat atau keuntungan bagi penggunanya, antara lain:

1. Memperluas kemampuan pengambilan keputusan dalam memproses data atau informasi bagi penggunanya.
2. Membantu pengambilan keputusan dalam hal penghematan waktu yang dibutuhkan untuk memecahkan masalah terutama berbagai masalah yang sangat kompleks dan tidak terstruktur.
3. Dapat menghasilkan solusi dengan lebih cepat serta hasilnya dapat diandalkan.
4. Walaupun suatu sistem pendukung keputusan, mungkin saja tidak mampu memecahkan masalah yang dihadapi oleh pengambil keputusan, namun dapat menjadi stimulan bagi pengambil keputusan dalam memahami persoalannya, karena sistem pendukung keputusan mampu menyajikan berbagai alternatif.
5. Dapat menyediakan bukti tambahan untuk memberikan bukti tambahan untuk memberikan pembenaran sehingga posisi pengambil keputusan.

2.2.2 Beasiswa

Pada dasarnya, beasiswa adalah penghasilan bagi yang menerimanya. Beasiswa ini sesuai dengan ketentuan pasal 4 (ayat 1) UU PPh/2000. Disebutkan pengertian penghasilan adalah tambahan kemampuan ekonomis dengan nama dan dalam bentuk apapun yang diterima atau diperoleh dari sumber indonesia atau luar indonesia yang dapat digunakan untuk konsumsi

atau menambah kekayaan wajib pajak (WP), karena beasiswa bisa diartikan menambah kemampuan ekonomi bagi penerimanya, berarti beasiswa merupakan penghasilan. (Astuti and Muammar 2015)

2.2.3 Naive Bayes

Naive Bayes ialah sebuah pengklasifikasian probabilistik sederhana yang menghitung sekumpulan probabilitas dengan menjumlahkan frekuensi dan kombinasi nilai dari dataset yang diberikan. Menurut Bustami dalam Saleh *Naive Bayes* merupakan pengklasifikasian dengan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya. Metode *Naive Bayes* juga dinilai berpotensi baik dalam mengklasifikasikan dokumen dibandingkan dengan metode pengklasifikasian lain dalam hal akurasi dan efisiensi komputasi. (Dahri, Agus, and Khairina 2016)

Dimana persamaan Teori Bayes tersebut adalah:

$$P(H|X) = \frac{P(X|H).P(H)}{P(X)} \quad (1)$$

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 probabilitas)

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

$P(X)$: Probabilitas X

Untuk menjelaskan teorema *Naive Bayes*, perlu diketahui bahwa proses klasifikasi membutuhkan sejumlah petunjuk untuk menentukan kelas yang

cocok bagi sampel yang dianalisis. Maka dari itu, teorema *bayes* di atas di sesuaikan seperti berikut :

$$P(C|F_1 \dots F_n) = \frac{P(C)P(F_1 \dots F_n|C)}{P(F_1 \dots F_n)} \quad (2)$$

Diketahui variabel C mempresentasikan kelas, sementara variabel $F_1 \dots F_n$ mempresentasikan karakteristik petunjuk yang dibutuhkan untuk melakukan klasifikasi. Maka dari itu, rumus tersebut menjelaskan bahwa peluang masuknya sampel karakteristik tertentu dalam kelas C (*Posterior*) adalah peluang munculnya kelas C (sebelum masuknya sampel, biasanya disebut *prior*), dikali dengan peluang munculnya karakteristik sampel pada kelas C (biasanya disebut dengan *likelihood*), dibagi dengan peluang munculnya karakteristik sampel secara global (biasanya disebut dengan *evidence*). Maka rumus diatas bisa ditulis dengan sederhana yaitu sebagai berikut :

$$Posterior = \frac{Prior \times Likelihood}{Evidence} \quad (3)$$

Nilai *evidence* akan tetap untuk setiap kelas pada satu sampel. Nilai dari posterior tersebut akan dibandingkan dengan nilai-nilai posterior kelas lainnya untuk dapat menentukan ke kelas suatu sampel yang akan diklasifikasikan. Penjelasan lebih lanjut dalam rumus *Bayes* akan dilakukan dengan menjabarkan $(C|F_1 \dots F_n)$ menggunakan aturan perkalian sebagai berikut :

$$\begin{aligned} P(C|F_1 \dots F_n) &= P(C)P(F_1 \dots F_n|C) \\ &= P(C)P(F_1|C) P(C)P(F_2|C, F_1)P(F_3, \dots, F_n|C, F_1, F_2) \\ &(4) \\ &= P(C)P(F_1|C)P(F_2|C, F_1)P(F_3|C, F_1, F_2), P(F_4, \dots, F_n| C, F_1, F_2, F_3) \\ &= P(C)P(F_1|C)P(F_2|C, F_1)P(F_3|C, F_1, F_2) \dots P(F_n|C, F_1, F_2, F_3, \dots, F_{n-1}) \end{aligned}$$

Dapat diketahui bahwa hasil penjabaran tersebut menyebabkan semakin banyak dan semakin kompleks faktor-faktor syarat yang mempengaruhi nilai probabilitas, hampir tidak memungkinkan untuk dilakukan analisis satu per satu. Maka dari itu, perhitungan tersebut menjadi sangat sulit untuk dilakukan perhitungan. Digunakan asumsi independensi yang sangat tinggi (naif), bahwa

setiap petunjuk F_1, F_2, \dots, F_n saling bebas satu dengan lainnya. Dengan asumsi tersebut, maka berlaku suatu kesamaan seperti berikut :

$$P(P_i|F_j) = \frac{P(F_i \cap F_j)}{P(F_j)} = \frac{P(F_i)P(F_j)}{P(F_j)} = P(F_i) \quad (5)$$

Untuk $i \neq j$, sehingga didapatkan sebagai berikut :

$$P(P_i|C, F_j) = P(P_i|C)$$

$$P(X_i = x_i | Y = y_j) = \frac{1}{\sqrt{2\pi}\sigma_j} e^{-(x_i - \mu_j)^2 / 2\sigma_j^2} \quad (6)$$

Keterangan :

P : Peluang

X_i : Atribut ke i

x_i : Nilai atribut ke i

Y : Kelas yang dicari

y_j : Sub Kelas Y yang dicari

μ : Mean, ialah rata-rata dari seluruh atribut

σ : Standar deviasi, ialah varian dari seluruh atribut

Setelah itu ada alur dari metode Naive Bayes ialah sebagai berikut :

1. Baca data training
2. Hitung jumlah dan probabilitas, jika data numerik maka harus dilakukan :
 - a. Cari nilai mean dan standar deviasi dari masing-masing parameter yang data numerik
 - b. Cari nilai probabilitas dengan cara menghitung jumlah data yang sesuai dari kategori yang sama dibagi dengan jumlah data pada kategori tersebut
 - c. Menghasilkan nilai dalam tabel mean, standar deviasi, dan probabilitas
 - d. Solusi dapat dihasilkan

2.2.4 RAD

Rapid application development (RAD) adalah model pembangunan perangkat lunak yang tergolong teknik incremental. RAD menekankan pada siklus pembangunan singkat dan cepat. Rapid application development menggunakan metode iteratif (berulang) dalam mengembangkan sistem dimana working model sistem dikonstruksikan di awal pengembangan dengan tujuan menetapkan kebutuhan pengguna. Model kerja digunakan hanya sekali saja sebagai basis desain dan implementasi akhir. (Setiawan et al. 2011)

RAD mempunyai 3 tahapan:

1. *requirement planning*

Pada tahap ini yang dilakukan adalah melakukan pengumpulan dan identifikasi data yang nantinya dibutuhkan untuk pembuatan sistem, data bisa diperoleh dengan komunikasi antara user dan analyst.

2. *design system*

Pada tahap secara rinci dengan membuat rancangan. Perancangan bisa dibuat dengan model dfd ataupun *Usecase Diagram*.

3. *implementation*










Tahap implementasi merupakan tahap pengkonversian desain sistem menjadi sebuah program utuh dan dilakukan oleh seorang programmer menggunakan bahasa pemrograman tertentu.

D. Flowchart

Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. (Marlina, Oktaroza, and Nurhayati 2019)

Simbol-simbol dalam flowchart adalah sebagai berikut :

Tabel 2. 1 Simbol Flowchart



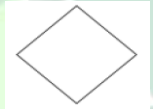
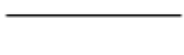
SIMBOL	NAMA SIMBOL	FUNGSI
	Terminator	Permulaan / akhir program
	Flow Line	Arah aliran program
	Preparation	Proses inisialisasi / pemberian harga awal
	Process	Proses perhitungan / proses pengolahan data
	Input / Output Data	Proses Input / output data, parameter, informasi
	Predefined Process Sub Program	Permulaan sub program / proses menjalankan sub program
	Decision Keputusan	Perbandingan pernyataan, penyeleksian data yang memberikan pihak untuk langkah selanjutnya
	On Page Connector	Penghubung bagian-bagian flowchart yang berada pada satu halaman
	Off Page Connector	Penghubung bagian-bagian flowchart yang berada pada halaman berbeda

2.2.5 Entity Relationship Diagram (ERD)

Entity Relationship Diagram adalah pemodelan awal basis data yang akan dikembangkan berdasarkan teori himpunan dalam bidang matematika untuk pemodelan basis data rasional.

ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow's foot, dan beberapa notasi lain. Namun yang banyak digunakan adalah notasi dari Chen.

Tabel 2. 2 Simbol ERD

No	Simbol	Keterangan
1		HIMPUNAN ENTITAS Digunakan untuk menggambarkan objek yang dapat diidentifikasi dalam lingkungan dalam lingkungan pemakai.
2		ATRIBUT Menggambarkan elemen-elemen dari satu entity yang menggambarkan <i>entity</i> .
3		RELASI Entity dapat berhubungan satu sama lain. Hubungan ini disebut <i>relationship</i> .
4		LINK Digunakan untuk menghubungkan <i>entity</i> dengan relasi dan <i>entity</i> dengan atribut

2.2.6 WEB

World Wide Web atau yang biasa kita kenal dengan sebutan WEB adalah salah satu layanan yang didapat oleh pemakai computer yang terhubung ke internet. Dalam web ini menyediakan beragam informasi yang dibutuhkan oleh pemakai computer yang terhubung

ke internet. Website atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, atau gabungan dari semuanya itu baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing konten atau isi web itu sendiri dihubungkan dengan jaringan jaringan halaman. (Afriliani, Fitriani 2015)

2.2.7 PHP

PHP adalah sebuah bahasa scripting yang dibundel dengan HTML, yang dijalankan di sisi server. Bahasa ini memungkinkan para pembuat aplikasi web menyajikan halaman HTML dinamis dan interaktif dengan cepat dan mudah, yang dihasilkan server. PHP juga dimaksudkan untuk mengganti teknologi lama seperti CGI (Common Gateway Interface). PHP biasa berinteraksi dengan hampir semua teknologi web yang sudah ada. Developer biasa menulis sebuah program PHP yang mengeksekusi suatu program CGI di server web lain.

Fleksibilitas ini amat bermanfaat bagi pemilik situs-situs web yang besar dan sibuk, karena pemilik masih biasaa mempergunakan aplikasi-aplikasi yang sudah terlanjur dibuat di masa lalu dengan CGI, ISAP, atau dengan script seperti perl, Awk atau python selama proses migrasi ke aplikasi baru yang dibuat dengan PHP. Ini mempermudah dan memperhalus peralihan antara teknologi lama den teknologi baru. Saat server melayani permintaan dari browser web akan suatu dokumen, server sebenarnya hanya mengambil suatu file di dalam disk dan melakukan beberapa pekerjaan untuk transmisi seperti menambahkan informasi tipe dokumen, merubah formatnya agar biasa dikirim menggunakan HTTP, yang mengirimkan semuanya ke browser. Browser web menerima file HTML dan menampilkannya ke layar monitor client. Sumbernya tetap berada di

server dan di sana ia tidak berubah sama sekali.(Yusman Dan Maryati. 2012)

2.2.8 XAMPP

XAMPP merupakan paket PHP dan MYSQL berbasis open source yang dapat digunakan sebagai tool pembantu pengembangan aplikasi berbasis PHP. Fungsi XAMPP adalah sebagai web server yang berdiri sendiri (localhost) yang terdiri dari program Apache, MySQL, PHP dan Pearl (Riyanto 2014).

2.2.9 MYSQL

SQL merupakan kependekan dari Structured Query Language (Bahasa Query Terstruktur). SQL lebih dekat dengan DML dari pada DDL. Namun tidak berarti SQL tidak menyediakan perintah DDL. SQL lebih menekankan pada aspek pencarian dari dalam tabel. Aspek pencarian ini sedemikian penting karena di sinilah sebenarnya inti dari segala upaya kita melakukan pengelolaan data. Data dalam basis data diorganisasi sedemikian rupa dengan tujuan untuk memudahkan pencarian di kemudian hari.

Sebagai sebuah bahasa, SQL telah distandarisasi dan mengalami beberapa perubahan atau penyempurnaan. SQL muncul pertama kali pada tahun 1970 dengan nama Sequel (nama yang masih sering digunakan hingga saat ini). Standarisasi yang pertama dibuat pada tahun 1986 oleh ANSI (American National Standards Institute) dan ISO (International Standard Organization), yang disebut SQL-86. Pada tahun 1989 SQL-86 diperbaharui menjadi SQL-89. Standar terakhir yang dibuat adalah SQL-92.

Pernyataan-pernyataan SQL digunakan untuk melakukan beberapa tugas seperti : update data pada basis data, atau

menampilkan data dari basis data. Beberapa software RDBMS yang dapat menggunakan SQL, seperti : Oracle, Sybase, Microsoft SQL Server, MySQL, Microsoft Access, Ingres, dsb. Setiap software basis data mempunyai mungkin bahasa perintah / sintaks yang berbeda, namun pada prinsipnya mempunyai arti dan fungsi yang sama.(Mulyanto 2008)

Perintah utama dalam SQL adalah select. Struktur utama perintah adalah sebagai berikut:

Select <kolom>

From <table>

Where <kondisi>

2.2.10 UML (Unified Modelling Language)

UML adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi obyek. Hal ini disebabkan karena UML menyediakan bahasa pemodeln visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku.

UML merupakan kesatuan dari bahasa yang dikembangkan oleh Booch, OMT (Object Modeliing Technique) dan OOSE (Object Oriented Software Engineering). Metode Booch dari Grady Booch sangat terkenal dengan nama metode *Design Object Oriented* . Metode ini menjadikan proses analisis dan design ke dalam empat tahapan iteratif, yaitu : identifikasi kelas-kelas dan obyek-obyek, identifikasi semantik dari hubungan obyek dan kelas tersebut , perincian interface dan implementasi. Keunggulan metode Booch adalah pada detil dan kayanya dengan notasi dan elemen. Pemodelan OMT yang dikembangkan oleh Rambaugh didasarkan analisis struktur dan pemodelan entity-relationship . (Anam and Muharram 2018)

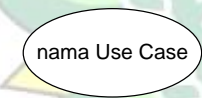
Ada tiga karakter penting yang melekat di UML yaitu sketsa, cetak program dan bahasa pemrograman. Sebagai sebuah alat bantu

modeling dalam suatu pengembangan sistem , UML memiliki beberapa diagram yang mampu membantu pengembang mengkomunikasikan sistem yang akan mereka buat , diagram-diagram tersebut antara lain adalah use case, activity diagram, class diagram dan sequence diagram.

2.2.11 USE CASE

Use case adalah rangkaian/uraian sekelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh sebuah aktor. Use case digunakan untuk membentuk tingkah-laku benda/ things dalam sebuah model serta di realisasikan oleh sebuah collaboration. Umumnya use case digambarkan dengan sebuah elips dengan garis yang solid, biasanya mengandung nama. Use case gambar 1 menggambarkan proses system (kebutuhan system dari sudut pandang user).(Rizal Sengkey 2015)

Tabel 2. 3 Simbol Use Case

Simbol	Nama Simbol	Deskripsi
	<i>Use Case</i>	<i>Use Case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktif, yang dinyatakan dengan menggunakan kata kerja

2.2.12 CLASS DIAGRAM

Class dalam notasi UML digunakan dengan kotak. Nama class menggunakan huruf besar diawal kalimatnya dan diletakkan diatas kotak. Bila class mempunyai nama yang terdiri dari 2 suku kata atau lebih, maka semua suku kata digabungkan tanpa spasi dengan huruf awal tiap suku kata menggunakan huruf besar .

Tabel 2. 4 Simbol Class Diagram





No	Simbol	Keterangan
1		Simbol ini adalah simbol untuk sebuah kelas pada struktur sistem. penulisan disana tidak diperbolehkan menggunakan spasi. simbol ini memiliki 3 susunan, yaitu kotak pertama adalah nama kelas, kedua atribut dan terakhir operasi.
2		Lingkaran ini adalah simbol untuk interface atau dalam bahasa indonesianya antar muka. konsep yang digunakan pun sama dengan pemrograman berorientasi object (OOP).
3		Simbol ini sering disebut dengan simbol Association atau dalam bahasa indonesianya yaitu asosiasi. Garis ini adalah garis yang digunakan untuk menghubungkan atau merelasikan kelas satu dengan kelas yang lainnya dengan makna umum.
4		Nama dari simbol ini adalah indirected association atau dalam bahasa indonesianya adalah asosiasi berarah. Simbol ini merupakan simbol relasi antar kelas seperti yang diatas, namun yang membedakan pada relasi ini adalah cara penggunaannya. Simbol ini digunakan jika kelas yang satu digunakan oleh kelas yang lainnya.
5		Simbol ini bernama Generalisasi. Generalisasi digunakan untuk menghubungkan antar kelas dengan arti umum-khusus. Jadi jika ada kelas bermakna umum dan kelas bermakna khusus dapat menggunakan simbol ini.
6		Nama dari simbol ini adalah Aggregation atau dalam bahasa indonesia nya Agregasi. Simbol ini adalah simbol yang menghubungkan antar kelas dengan makna untuk semua bagian. Jadi relasi ini digunakan jika kelas yang satu adalah semua bagian dari kelas yang lainnya.
7		Nama dari simbol ini adalah Dependency atau dalam bahasa indonesia nya ketergantungan. Kadangkala sebuah class menggunakan class yang lain. Umumnya penggunaan dependency digunakan untuk menunjukkan operasi pada suatu class yang menggunakan class yang lain. Sebuah dependency dilambangkan sebagai sebuah panah bertitik-titik.


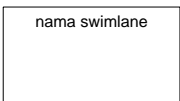
2.2.13 Activity Diagram

Activity diagram yaitu teknik untuk mendeskripsikan logika procedural, proses bisnis dan aliran kerja dalam banyak kasus. Activity diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. (Anam and Muharram 2018)

Activity diagram seperti sebuah flowchart. Activity diagram menunjukkan tahapan, pengambilan keputusan dan pencabangan. Diagram ini sangat berguna untuk menunjukkan operation sebuah obyek dan proses bisnis. Kelebihan activity diagram dibanding flowchart adalah kemampuannya dalam menampilkan aktivitas paralel serta dapat digunakan untuk menunjukkan siapa mengerjakan apa dengan teknik partition. (Anam and Muharram 2018)

Tabel 2. 5 Simbol Activity Diagram

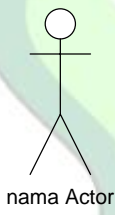
<i>Simbol</i>	<i>Nama Simbol</i>	<i>Deskripsi</i>
	<i>Initial State</i>	Status awal aktifitas sistem, sebuah diagram aktifitas memiliki sebuah status awal
	<i>Action State</i>	Aktifitas yang dilakukan sistem, aktifitas biasanya diawali dengan kata kerja
	<i>Decision</i>	Asosiasi percabangan dimana jika ada pilihan aktifitas lebih dari satu
	<i>Join</i>	Asosiasi penggabungan lebih dari satu aktifitas digabungkan menjadi satu

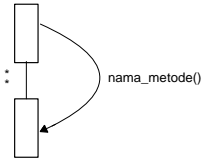
	<i>Final State</i>	Status akhir yang dilakukan sistem, sebuah diagram aktifitas memiliki sebuah status akhir
	<i>Swimlane</i>	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktifitas yang terjadi

2.2.14 Sequence Diagram

Sequence diagram digunakan untuk menggambarkan perilaku pada sebuah scenario. Diagram ini menunjukkan sejumlah contoh obyek dan message (pesan) yang diletakkan diantara obyek-obyek ini didalam usecase. Komponen utama sequence diagram terdiri atas obyek yang dituliskan dengan kotak segiempat bernama. Message diwakili oleh garis dengan tanda panah dan waktu yang ditunjukkan dengan progress vertical. (Nuripah 2017)

Tabel 2. 6 Simbol Sequence Diagram

<i>Simbol</i>	<i>Nama Simbol</i>	<i>Deskripsi</i>
	<i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor
	<i>Lifeline</i>	Menyatakan kehidupan suatu objek
	<i>Message (Call)</i>	Menyatakan suatu objek memanggil operasi-metode yang ada pada objek

	<p>lain atau dirinya sendiri, arah panah mengarah kepada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi</p>
---	--

2.2.15 Langkah-langkah penggunaan UML

Berikut ini adalah tipe pengembangan piranti lunak dengan menggunakan UML.(Ali and Septiani 2016):

1. Buatlah daftar businessprocess dari level tertinggi untuk mendefinisikan aktifitas dan proses yang mungkin muncul.
2. Petakan use case untuk tiap business process untuk mendefinisikan dengan tepat fungsionalitas yang harus disediakan oleh sistem. Kemudian perhalus use case diagram dan lengkapi dengan requirement, constraints, dan catatan-catatan lain.
3. Buatlah deploment diagram secara kasar untuk mendefinisikan arsitektur fisik sistem.
4. Definisikan requirement lain(non-fungsional, securitydan sebagainya) yang juga harus disediakan oleh sistem.
5. Berdasarkan use case diagram, mulailah membuat activity diagram.
6. Definisikan objek-objek level atas (package atau dominan) dan buatlah sequence dan collaboration diagram untuk tiap alir pekerjaan. Jika sebuah use case memiliki kemungkinan alir normal dan eror, buatlah satu diagram untukmasing-masing alir.
7. Buatlah rancangan user interface model yang menyediakan antarmuka bagi pengguna untuk menjalankan skenario use case.
8. Berdasarkan model-model yang sudah ada, buatlah class diagram. Setiap package atau domain dipecah menjadi hirarki class lengkap

dengan atribut dan motodenya. Akan lebih baik jika untuk setiap class dibuat unit test untuk menguji fungsionalitas class dan interaksi dengan class lain.

9. Setelah class diagram dibuat, kita dapat melihat kemungkinan pengelompokkan class menjadi komponen-komponen. Karena itu buatlah component diagram pada tahap ini. Juga didefinisikan tes integrasi untuk setiap komponen meyakinkan ia berinteraksi dengan baik.
10. Perhalus deployment diagram yang sudah dibuat. Detikan kemampuan dan requirement piranti lunak, sistem operasi, jaringan, dan sebagainya. Petakan komponen ke dalam mode.
11. Mulailah membangun sistem. Ada dua pendekatan yang dapat digunakan :
 - a) Pendekatan use case, dengan meng-assign setiap use case kepada tim pengembang tertentu untuk mengembangkan unit code yang lengkap dengan tes.
 - b) Pendekatan komponen, meng-assign setiap komponen kepada tim pengembang tertentu.
12. Lakukan uji modul dan uji integrasi serta perbaiki model beserta codenya. Model harus selalu sesuai dengan code yang aktual.
13. Piranti lunak siap dirilis.

2.2.16 Keunggulan-Keunggulan UML

Pada umumnya UML memiliki keunggulan-keunggulan sebagai berikut (Hasan and Pratama 2018):

1. **Uniformity** : Dengan metode UML para pengembang cukup menggunakan 1 metodologi dari tahap analisis hingga perancangan. Hal ini tidak bisa dilakukan dalam metodologi pengembangan terstruktur. Dengan pengembangan masa kini ke arah aplikasi GUI (Graphical User Interface), UML juga memungkinkan kita merancang komponen antar muka pengguna (User Interface) secara integrasi

bersama dengan perancangan perangkat lunak sekaligus dengan perancangan basisdata.

2. **Understandability** : Dengan metode ini kode yang dihasilkan dapat diorganisasi ke dalam kelas-kelas yang berhubungan dengan masalah sesungguhnya sehingga lebih mudah dipahami siapapun juga.
3. **Stability** : Kode program yang dihasilkan relatif stabil sepanjang waktu sebab sangat mendekati permasalahan sesungguhnya di lapangan.
4. **Reusability** : Dengan metode berorientasi obyek, dimungkinkan penggunaan ulang kode, sehingga pada gilirannya akan sangat mempercepat waktu pengembangan perangkat lunak.

2.2.17 MI Al Islam Bangsri

Madrasah Ibtidaiyah Al Islam merupakan salah satu yayasan pendidikan swasta yang ikut membantu Pemerintah Daerah, khususnya dibidang pendidikan Agama Islam. Madrasah Ibtidaiyah merupakan institusi pendidikan yang bercorak keislaman. Madrasah ibtidaiyah telah lama menjadi lembaga yang memiliki kontribusi penting dalam ikut serta mencerdaskan bangsa. Madrasah Ibtidaiyah memiliki beberapa fungsi,yaitu : Pendidikan yang melakukan transfer ilmu-ilmu agama islam dan nilai-nilai islam, sebagai lembaga keagamaan yang melakukan contoh sosial, dan berperan sebagai institusi pendidikan ditingkat dasar yang tidak hanya memperkenalkan ilmu pengetahuan secara umum tetapi juga memberikan nilai-nilai keagamaan.

Madrasah Ibtidaiyah Al Islam memilikin visi yaitu Terwujudnya generasi yang beriman, bertaqwa, berakhlak mulia, berprestasi dan berwawasan lingkungan hidup. Maka dari itu, Madrasah Ibtidaiyah Al Islam memiliki misi yang pertama Meletakkan dasar-dasar keimanan dan ketaqwaan kepada Allah SWT, Kedua Menerapkan norma-norma agama, hukum, sosial, dan susila, Ketiga Meningkatkan pelayanan pada kegiatan pembelajaran dan pengembangan diri, Keempat Menerapkan perilaku hidup bersih dan sehat di

lingkungan madrasah, Kelima Menciptakan lingkungan yang nyaman dan menyenangkan sebagai wahana bersosialisasi warga madrasah dengan masyarakat sekitar, Keenam Mengintegrasikan wawasan lingkungan hidup ke dalam mata pelajaran/ ekstrakurikuler, Ketujuh Mengoptimalkan pelaksanaan 7K di Madrasah.

Madrasah Ibtidaiyah Al Islam memiliki program beasiswa yang dapat diberikan kepada siswa atau siswi yang memiliki kriteria yang sesuai untuk program beasiswa. Ketentuan untuk mendapatkan program beasiswa yang ada di Madrasah Ibtidaiyah Al Islam yaitu sebagai berikut :

1. Perkerjaan Orang Tua
2. Penghasilan Orang Tua
3. Jumlah Tanggungan dalam satu keluarga
4. Daya Listrik (Watt)
5. Nilai Rata-rata Raport

