BAB 2
LANDASAN TEORI

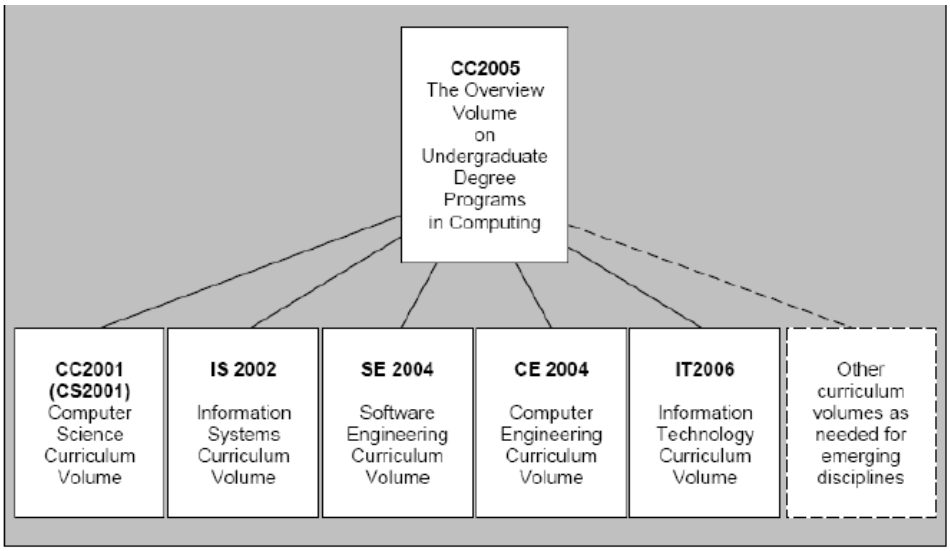
2.1 *Computing Curricula 2005*

Kurikulum yang berkembang sekarang ini di negara republik indonesia sangat berkembang dengan pesat terutama di kampus yang berasaskan komputer, dimana kurikulum Software Engineering merupakan bagian dari kampus yang berasaskan komputer, yang merupakan disiplin ilmu dan ada juga yang merupakan prodi/jurusan software engineering. Program ACM antara lain adalah mengadakan konferensi, memberikan penghargaan dalam bidang tertentu, penerbitan jurnal-jurnal ilmiah, dan pengembangan rekomendasi kurikulum untuk bidang studi yang terkait dengan komputer.

Computing Curricula 2005 (*CC2005*) merupakan projek dari tiga asosiasi kenamaan dunia yaitu *The Association for Computing Machinery* (ACM), *The Association for Information Systems* (AIS) dan *The Computer Society* (IEEE-CS) yang bergabung membentuk suatu gugus tugas untuk menyusun kurikulum terkait dengan dunia per-komputer-an, yang kemudian diacu oleh banyak institusi pendidikan seluruh dunia. *CC2005* yang dirilis pada 30 September 2005 itu sendiri secara tegas membedakan antara disiplin ilmu CE, CS, IS, IT dan SE. Secara singkat, deskripsinya adalah sebagai berikut :

1. *Computer Engineering* (CE) mempelajari antara lain tentang bagaimana mendisain dan merancang konstruksi komputer dan sistem berbasis komputer yang baik. Ilmu yang terlibat adalah tentang *hardware, software,* komunikasi dan tentu saja memahami proses interaksi antar bagian tersebut.
2. *Computer Science* (CS) mempelajari antara lain tentang bagaimana mengembangkan robotik, *computer vision*, *intelligence system,* bio-informatika, dan hal-hal lainnya yang terkait dengan pengembangan komputer kedepannya. Untuk itu, ilmuwan komputer ini harus memelajari berbagai hal secara teoritis hingga dasar-dasar algoritma dan mengetahui cara mengimplementasikannya.

3. *Information Systems* (IS) mempelajari antara lain tentang bagaimana membangun dan mengintegrasikan solusi teknologi informasi dengan proses bisnis yang ada, sehingga suatu institusi bisnis akan dapat mencapai tujuannya dengan efektif dan efisien.
4. *Information Technology* (IT) mempelajari antara lain tentang bagaimana sebuah sistem informasi di sebuah institusi sudah tepat keberadaannya, berfungsi sesuai harapan, mudah dalam pengaturan operasional, serta aman. Selain itu aspek non-teknis semisal memahami kebutuhan dari pengguna akhir, menterjemahkan kebutuhan manajemen, dan hal-hal praktis lainnya juga dipelajari.
5. *Software Engineering* (SE) mempelajari antara lain tentang bagaimana merancang-bangun dan memelihara sebuah sistem software yang reliabel, efisien, efektif serta mudah dikembangkan dan dapat memenuhi kebutuhan dari kustomer atau pengguna.



Gambar 2.1 *Structure of the Computing Curricula Series* (Sumber*: CC2005*)

Kelompok keilmuan Teknik Informatika UNIKOM mengadopsi Computing Curricula 2005[7], yaitu :

1. **Kelompok A (Sistem Informasi),** yang mencakup di antaranya:
   a) Sistem Informasi Geografis (SIG)

      SIG menyajikan informasi dalam bentuk grafis dengan menggunakan peta sebagai antar muka, saat ini banyak digunakan untuk perencanaan, pelaksanaan, dan pengendalian yang berkaitan dengan wilayah geografis. Contoh yang termasuk dalam topik ini diantaranya :

      a. Sistem Informasi Geografis Untuk Menunjang Rencana Strategis Penanggulangan Tumpahan Minyak Di Selat Malaka
      b. Sistem Penentuan Lokasi untuk Membuka Cabang Baru Toko Kue X Menggunakan Sistem Informasi Geografis
      c. Sistem Informasi Geografis Asrama Universitas ABC
      d. Sistem Informasi Geografis Pemantauan Penyebaran Penyakit Demam Berdarah Di Kota XYZ
      e. Sistem Informasi Geografis Penempatan BTS Provider XYZ Di Kabupaten X
      f. Dan lain-lain

   b) *Supply Chain Management (SCM)*

      SCM adalah koordinasi dari bahan, informasi dan arus keuangan antara perusahaan yang berpartisipasi. Contoh yang termasuk dalam topik ini diantaranya :

      a. Sistem Pengendalian Distribusi Sepatu Menggunakan Pendekatan Supply Chain Management di PT. ABC
      b. Supply Chain Management Produksi Obat ABC di PT. XYZ
      c. Dan lain-lain

   c) Sistem Informasi Manajemen (SIM)

      SIM adalah sistem perencanaan bagian dari pengendalian internal suatu bisnis yang meliputi pemanfaatan manusia, dokumen, teknologi, dan prosedur oleh pihak manajerial untuk memecahkan masalah bisnis seperti biaya produk, layanan, atau suatu strategi bisnis. Sistem

informasi manajemen dibedakan dengan sistem informasi biasa karena SIM digunakan untuk menganalisis sistem informasi lain yang diterapkan pada aktivitas operasional organisasi. SIM dibangun agar organisasi memiliki informasi yang bermanfaat dalam pembuatan keputusan manajemen, baik yang meyangkut keputusankeputusan rutin maupun keputusan-keputusan yang strategis. Contoh yang termasuk dalam topik ini diantaranya :

a. Sistem Informasi Manajemen Bagian Produksi PT. XYZ
b. Sistem Informasi Manajemen Simpan Pinjam di Bank Y
c. Dan lain-lain

d) Sistem Informasi Eksekutif (EIS)

EIS merupakan sebuah sistem berbasis komputer yang menyediakan data & informasi (baik internal maupun eksternal) yang dibutuhkan oleh top eksekutif, dimana informasi disediakan dengan cepat & tepat serta disajikan dalam bentuk laporan-laporan yang didukung grafik-grafik ataupun kemampuan drill-down. Pada umumnya, informasi tersebut akan digunakan oleh top eksekutif untuk memantau kondisi bisnis perusahaan secara umum atau hasil sebuah proses bisnis secara khusus. Oleh karena itu, penelitian skripsi dengan tema EIS merupakan penelitian yang berada dalam lingkup pembangunan EIS ataupun pengembangan EIS. Contoh yang termasuk dalam topik ini diantaranya:

a. Sistem Informasi Eksekutif PT. ABC
b. Sistem Informasi Eksekutif Departemen Logistik PT. ABC
c. Sistem Informasi Eksekutif Departemen Logistik PT. ABC untuk Menganalisis Total Biaya Penyimpanan Optimum
d. Sistem Informasi Eksekutif PT. ABC untuk Memonitoring Kinerja Departemen Logistik.
e. Sistem Informasi Eksekutif PT. ABC untuk Memonitoring Kinerja Perusahaan
f. Dan lain lain

e) *Enterprise Architecture (EA)*

EA merupakan sebuah gambaran As-Is & To-Be suatu enterprise yang berkaitan dengan mengorganisasikan seluruh proses bisnis enterprise, informasi yang dibutuhkan serta teknologi pendukungnya. Oleh karena itu, penelitian skripsi dengan tema ini merupakan penelitian yang menganalisis serta menyelaraskan antara penerapan sistem informasi dengan kebutuhan enterprise (organisasi), sehingga terbentuk suatu rancangan strategis dan komponen arsitektural (seperti arsitektur bisnis, arsitektur data, arsitektur informasi & arsitektur teknologi). Contoh yang termasuk dalam topik ini diantaranya:

a. Perancangan Enterprise Arsitektur Planning di PT.ABC
b. Perancangan Enterprise Arsitektur Planning (EAP) pada Proses Manajemen Aset dengan Zachman Framework (Studi Kasus Divisi Manajemen Fasilitas PT. ABC)
c. Perancangan Enterprise Arsitektur Menggunakan Togaf Versi 9 (Studi Kasus : Dewan Kehormatan Penyelenggaraan Pemilu (DKPP))
d. Pemodelan Enterprise Arsitektur Zachman Framework pada Sistem Informasi Fakultas Teknologi Informasi Universitas ABC
e. Dan lain lain

f) *Knowledge Management System (KMS)*

Knowledge Management (KM) berisi serangkaian strategi, sistem dan teknik yang digunakan oleh individu, team dan organisasi untuk mengelola 'knowledge'. Sesuai dengan sifat 'knowledge' yang demikian subjective, validasi sebuah 'individual knowledge' menjadi 'collective knowledge' hanya bisa tercapai jika terbangun proses knowledge sharing tukar menukar individual knowledge. Collective knowledge yang tervalidasi dan terdokumentasi adalah sumber kekuatan organisasi. Contoh yang termasuk dalam topik ini diantaranya:

a. Sistem Pengelolaan Pengetahuan Untuk Meningkatkan Kreasi Dan Inovasi Perusahaan
b. Pengelolaan Materi Ajar dengan Pendekatan Knowledge Management System Di Instansi ABC
c. Dan lain-lain

g) *Costumer Relationship Management (CRM)*

CRM adalah sebuah sistem informasi yang terintegrasi yang digunakan untuk merencanakan, menjadwalkan, dan mengendalikan aktivitas-aktivitas prapenjualan dan pascapenjualan dalam sebuah organisasi. CRM melingkupi semua aspek yang berhubungan dengan calon pelanggan dan pelanggan saat ini, termasuk di dalamnya adalah pusat panggilan (call center), tenaga penjualan (sales force), pemasaran, dukungan teknis (technical support), layanan lapangan dan strategi organisasi atau perusahaan yang berfokus pada pembangunan dan pemeliharaan hubungan dengan pelanggan. Contoh yang termasuk dalam topik ini diantaranya:

a. Pengaruh Penerapan Customer Relationship Management (CRM) Terhadap Loyalitas Pelanggan PT. Bank Mandiri Tbk, Cabang Braga Bandung.
b. Dan lain-lain

h) Sistem Informasi Manajemen Proyek

Manajemen proyek merupakan aplikasi atau implementasi dari pengetahuan, ketrampilan, perangkat dan teknik pada suatu aktifitas proyek untuk memenuhi kebutuhan atau tujuan suatu proyek. Perencanaan, pemantauan dan pengontrolan terhadap aspek yang terdapat dalam sebuah proyek, serta motivasi yang ada didalamnya untuk mencapai tujuan proyek dengan waktu, biaya, kualitas dan performasi yang telah ditentukan. Sekumpulan penugasan/pekerjaan, teknik, serta perangkat yang diaplikasikan selama eksekusi atau pelaksanaan proyek. Contoh yang termasuk dalam topik ini diantaranya:

a. Sistem Informasi Manajemen Proyek PT. ABC
b. Manajemen Resiko Proyek PT.XYZ
c. Sistem Informasi Monitoring Proyek PT.ABC

2. **Kelompok B (RPL, Data dan BI),** yang mencakup di antaranya:

a) Rekayasa Perangkat Lunak

Bidang keilmuan Teknik Informatika yang membahas tentang konsep keilmuan yang digunakan dalam membangun/menilai sebuah perangkat lunak. Kajian untuk bidang ini diantaranya :

a. Coding Assesment
b. Pembangunan Framework
c. Design Pattern Finding
d. Code Refinement
e. Pembangunan Class Library
f. Format Method
g. Human Computer Interaction
h. Software Quality
i. Software Matrics
j. Software Testing
k. Software Migration
l. Software Refinement

b) Rekayasa Data

Bidang keilmuan Teknik Informatika yang membahas model data, implementasi teknologi basis data dan terapannya dalam kehidupan sehari-hari. Kajian untuk bidang ini diantaranya :

a. Data Mining
b. Datawarehouse
c. Clustering Database
d. Big Data
e. Web Mining/Text Mining
f. Basis Data Terdistribusi
g. NoSQL Performance

h. Data SpatioTemporal
i. Data Spatial
j. Query Processing and Optimization
k. Data Master Management

c) Business Intelligence

Aplikasi, teknologi, dan metodologi untuk mengumpulkan, menyimpan dan menyediakan akses terhadap data untuk menolong enterprise mengembangkan bisnisnya.

3. **Kelompok C (Jaringan Komputer dan Multimedia),** yang mencakup di antaranya:

a) Multimedia

Terkait dengan pengolahan data untuk menghasilkan informasi dengan media-media tertentu – atau gabungan media – menghasilkan nilai informasi yang lebih tinggi. Kajian untuk bidang ini diantaranya :

a. Voice Tecnology
   Speech Recognition, Speaker Recognition
b. Video Recognition
   Bisa berupa pengenalan wajah/objek pada video
c. Multimedia sebagai sarana pemberi informasi
d. Virtual Museum, Virtual Reality dan Augmented Reality
e. Chatbot Multimedia
f. Game (Algoritma/Metode)
g. Kajian
   Analisis Metode, Prototipe, Implementasi
h. Platform
   Dekstop, WEB, Mobile

b) Jaringan Komputer dan Internet

Membangun atau mengembangkan sistem dan konten yang terdapat dilingkungan jaringan komputer (LAN/WAN). Bukan untuk membangun jaringan (LAN/WAN). Bentuk pembahasan dalam skripsi untuk bidang ini dapat berupa :

a. Pembangunan Sistem / Aplikasi terkait dengan Sistem Jaringan
b. Komputer (Dekstop, Web atau Mobile)
c. Pembangunan Sistem dengan Konfigurasi Tools – Untuk Menghasilkan Sistem Baru.
d. Menganalisis Performansi – Jaringan Komputer
e. Menganalisis Quality Of Services (QoS) Pada Sistem Jaringan
f. Komputer

Termasuk didalamnya Teknologi Distributed Cmputing (Grid, Paralel Computing, Pervasive Computing) hingga teknologi Cloud Computiing.

Kajian untuk bidang ini diantaranya :

a. Aplikasi dengan protokol multimedia jaringan untuk menyediakan layanan multimedia didalam jaringan (Video on Demand (VOD), Live Streaming (Audio, Video). Fungsi komunikasi, entertainment, edukasi.
b. Pembangunan VOIP (Voice Over Internet protocol), VPN (Virtual Private Network). Metode, dan sistem.
c. Kajian-Kajian Keamanan Pada jaringan Komputer dan Internet
d. Metode/Algoritma pada Sistem Komputer Terdistribusi, Grid/Paralel Sistem
e. Cloud Computing – Aplikasi, Sekurity dan Arsitektur.
f. Kajian IPv6, Metode, Migrasi Plan dsb

c) Keamanan Sisten

Membahas kajian keamanan dari sistem disertai solusinya. Kajian untuk bidang ini diantaranya :

a. Kajian Keamanan Sistem
   Dilingkungan Standalone, Dilingkungan Jaringan + Internet
b. Platform
   Dektop, Web, Mobile
c. Tema Keamanan Sistem

VPN, Firewall, Proxy, Kriptografi (Metode/Algoritma), Steganografi (Metode/Algoritma), Digital Signature, Secure Coding - Kajian Ancaman-ancaman Keamanan disertai Solusinya

d) Teknologi IT

Pemanfaatan teknologi-teknologi IT pada berbagai bidang/aplikasi. Kajian untuk bidang ini diantaranya :

a. QRCode

Untuk berbagai keperluan basis WEB dan mobile(HTML 5) dengan akses data via mobile mengguakan data asli di Web

b. RFID

Untuk berbagai keperluan, Contoh Dapat RFID sistem Untuk Unikom atau RFID untuk Stadion, Pembayaran Bus Transjakarta/Trans bandung dsb)

c. NFC

Transfer pulsa, data dsb (teknologi baru, terbatas pada HP yang mendukung NFC)

d. Internet of Things

Smart System (Smart City, Smart Transportation, Smart Traffic, Smart Farming, Smart Fishing dsb)

Dapat diintegrasikan dengan Microcontorller, Micro PC (Adruino), sensor dsb. Namun tetap inti tetap aplikasi software sistemnya

Cakupan luas, Desa, rumah sakit, Kantor, UKM, Puskesmas, pendidikan, bengkel, Pesantren, Masjid dsb.

4. **Kelompok D (Teknologi Internet dan Mobile),** yang mencakup di antaranya:

a) E-Learning

Pembelajaran yang diselenggarakan melalui media elektronik. Audio, Video, Komputer, Tablet, Perangkat Mobile, Blog, Webcam, Smart

Whiteboards, Virtual Classroom, atau Penggabungan Teknologi. Contoh Kajian untuk bidang ini diantaranya :

a. Pembangunan Perangkat Lunak Pembelajaran (Fisika, Matematika, Membaca Al-Quran, Tulisan Sunda, dll) (CD-Interaktif, Situs)
b. Pembangunan Perangkat Lunak Untuk Pembuatan Materi Pembelajaran Online.
c. Situs Aggregator Konten Pembelajaran di Departemen Pendidikan

b) Learning Management System (LMS)

Perangkat lunak yang digunakan untuk mengirimkan/mendistribusikan, menelusuri dan mengatur pembelajaran. Dengan sub sistem Pendataan peserta, materi, instruktur, pemberian tugas, kehadiran, pengumuman, penilaian, latihan, diskusi dan pelacakan progress pembelajaran. Contoh Kajian untuk bidang ini diantaranya :

a. Pengembangan Learning Management System untuk SMA XYZ untuk pemeriksaan ujian essay secara otomatis.
b. Pengembangan Learning Management System untuk Lembaga Kursus XYZ untuk Penjadwalan Kursus yang optimal dengan algoritma ABC.
c. Pengembangan Learning Management Sistem berbasis Cloud

c) WEB/Internet Technology

Pemanfaatan teknologi web/internet sesuai kebutuhan. Teknologi yang bisa digunakan diantaranya Cloud Computing, Web 2.0, Web Service, Networking, HTML5, XML, CSS, JSON. Contoh Kajian untuk bidang ini diantaranya :

a. Kolaborasi Sistem Informasi Dampak Bencana dan Penanggulangannya (Kombinasi dari Online Map, GPS, Mobile)
b. Pemanfaatan Google Maps dalam Sistem Monitoring Persediaan Pangan di Badan Logistik Jawa Barat.

c. Monitoring kondisi lalu lintas di kota Bandung berdasarkan informasi dari media sosial. (Web Analytic, Media Sosial, Online Map).
d. Pembangunan Infrastruktur sistem E-Desa berbasis cloud computing.
e. Pembangunan Sistem Layanan File Sharing Khusus Untuk Materi Pembelajaran

d) Pengembangan E-Commerce

Tidak membuat situs e-commerce dari nol, tetapi pengembangan sistem e-commerce yang sudah ada. Sistem yang sudah ada boleh dari system apa pun. Teknologi yang bisa ditambahkan adalah Data Mining (Association Rule, Classification, dll), Web Analytic, Supply Chain Management (SCM), Customer Relationship Management (CRM), Online Map, Online Payment Gateway, Tracking, Social Media. Contoh Kajian untuk bidang ini diantaranya :

a. Pemanfaatan Association Rule Untuk Promosi Produk di Toko Online XYZ (Data Mining di data toko online)
b. Integrasi Customer Relationship Management (CRM) di Toko Online XYZ
c. Optimalisasi Penjadwalan Produksi Berdasarkan Pesanan Online di PT. XYZ (termasuk Supply Chain Management, Resource Management)
d. Pembangunan Infrastruktur Sistem Integrator Data Produk di Factory Outlet Bandung. (Data mining, Online Maps, Web Service)
e. Sistem Monitoring Trend di Social Media untuk Memprediksi Produk yang Dibutuhkan Masyarakat (Targeted Marketing, Web Analysis, Data Mining)

e) Mobile Technology

Pemanfaatan teknologi mobile di berbagai bidang. Teknologi yang bisa anda manfaatkan diantaranya Location Based Service (LBS) / GPS,

Web Service, Image Processing / Augmented Reality, Data Mining, Sensor. Contoh Kajian untuk bidang ini diantaranya :

a. Pemanfaatan GPS di perusahaan Taxi XYZ (GPS, Web, Database, Mobile Application, Web Service)
b. Pemanfaatan Teknologi Augmented Reality sebagai Virtual Guide Lokasi Bersejarah Kota Bandung.
c. Aplikasi Wisata Kuliner di Kota Bandung (Data Aggregator, Data Mining, Augmented Reality)
d. Pembangunan Aplikasi Pelaporan Distribusi Bantuan untuk Daerah Bencana Alam Berbasis Mobile
e. Sistem Peringatan Dini Banjir Berbasis Mobile (Sistem Notifikasi, Web Service, GPS)

5. **Kelompok E (Computer Science / CS)** kajian teoritis dan algoritmik disertai aplikasinya, yang mencakup di antaranya:

a) Artificial Intelligence

Kajian yang terkait adaptasi kemampuan manusia atau hewan ke dalam sistem komputer. Contoh Kajian untuk bidang ini diantaranya :

a. Perceptual Computing
b. Machine Learning (Statistical Machine Learning, Machine Learning in Robot)
c. Optimization case

b) Intelligent Tutoring

Kajian tentang sistem yang membantu proses pembelajaran dan evaluasinya. Contoh Kajian untuk bidang ini diantaranya :

a. Essay Grading
b. Question Generation
c. Programming learning (error-checking, suggestions, peer grading, style analysis)

c) Information Retrieval

Kajian yang mempelajari pencarian informasi tertentu yang relevan dengan kebutuhan (query) berdasarkan sekumpulan sumber informasi. Contoh Kajian untuk bidang ini diantaranya :

a. Indexing
b. Search Engine

d) Natural Language Processing

Kajian yang terkait pengenalan, pembelajaran suatu bahasa manusia supaya dapat dipahami komputer. Contoh Kajian untuk bidang ini diantaranya :

a. Information Extraction,
b. Question Answering,
c. Sentiment & Social Meaning,
d. Parsing & Tagging,
e. Machine Translation,
f. Dialog & Speech Processing,
g. Multilingual NLP,
h. Linguistic Structure,
i. Summarization

e) Programing Language & Environments

Kajian yang terkait pembangunan sistem kompilasi bahasa pemrograman dan elemen-elemen di dalamnya. Contoh Kajian untuk bidang ini diantaranya :

a. Automatic Analyzer of Correctness Program,
b. Languages Comparison,
c. Software Support for Languages (Compiler),
d. Program Optimization,
e. Formal Reasoning,
f. Automated Debugging (Finding Bug Fixes, Bugs Detection)

f) Image Recognition

Kajian yang terkait pengenalan pola citra (gambar 2 dimensi). Contoh Kajian untuk bidang ini diantaranya :

a. Pengenalan Pola Citra,
b. Pre-processing (Thinning, Edge Detection, Ekstraksi Ciri)

g) Voice Recognition

Kajian yang terkait pengenalan pola suara atau bunyi. Contoh Kajian untuk bidang ini diantaranya :

a. Speaker Recognition,
b. Speech Recognition

h) Teori Komputasi

Kajian yang terkait pencarian solusi atau pembuatan model tertentu untuk menyelesaikan masalahmasalah komputasi teoritik. Contoh Kajian untuk bidang ini diantaranya :

a. Teori Kompleksitas Komputasi (Masalah P = NP),
b. Kriptografi,
c. Teori Quantum Computing,
d. Teori Automata

2.2. Metode Preprocessing

Preprocessing adalah proses pengolahan data asli yang akan dipersiapkan untuk diolah pada tahap selanjutnya. Masukan data awal yaitu berupa dokumen. Tahapan preprocessing yang dilakukan pada penelitian ini meliputi *Tokenizing* , proses *Filtering*, proses *Case Folding*, proses *Stopword Removal*, proses *Stemming* dan perhitungan bobot TF-IDF serta Normalisasi.

2.2.1 *Tokenizing*

Proses *Tokenizing* adalah proses pemotongan *string input* berdasarkan tiap kata yang menyusunnya, memecahkan dokumen dan memisahkannya menjadi setiap kata. Pemecahan dokumen menjadi kata-kata tunggal dilakukan dengan men-*scan* dokumen dan setiap kata terindentifikasi atau terpisahkan dengan kata yang lain oleh pemisah spasi [8]. Fitur ini terdiri dari tipe kapitalisasi (tipe case), keberadaan digit, tanda baca, karakter spesial dan lain sebagainya. Hasil keluaran dari proses tokenizing akan dipergunakan sebagai masukan dalam tahap transformasi teks. Tahapan-tahapan yang dilakukan dalam *tokenizing* yaitu :

1. Membaca data masukan berupa teks
2. Memeriksa apakah sudah EOF. Jika sudah maka algorima berakhir, jika belum maka ke tahap 3.
3. Memeriksa teks dengan memecehnya menjadi kata (*term*). Mengecek apakah menemukan spasi. Jike bertemu spasi maka ke tahap 4. Jika tidak maka scan kembali.
4. Mengambil kata (*term*) dari teks dan membuat indeks baru untuk menandai asal kata (*term*).
5. Algoritma selesai.

2.2.2 *Filtering*

Filtering mengacu pada proses memutuskan istilah mana yang harus digunakan untuk merepresentasikan dokumen sehingga dapat digunakan untuk menggambarkan isi dokumen dan membedakan dokumen dengan dokumen lain yang ada pada koleksi [9]. Dengan kata lain filtering merupakan proses dimana teks selain karakter "a" sampai "z" dan spasi akan dihilangkan dan hanya menerima spasi. Tahapan-tahapan yang dilakukan yaitu :

1. Membaca data masukan berupa teks
2. Mengecek apakah ada huruf selain "a" sampai "z" dan " " (spasi). Jika tidak ditemukan, maka algoritma berakhir. Jika ada.
3. Hapus karakter yang bukan termasuk "a" sampai "z" dan " " (Spasi).
4. Algoritma selesai

2.2.3 *Case Folding*

Case folding adalah tahapan pemrosesan teks dimana semua teks diubah ke dalam *case* yang sama [10]. Pada penelitian ini semua huruf dalam teks dokumen diseragamkan menjadi huruf kecil . Tahapan-tahapan yang dilakukan yaitu :

1. Membaca data masukan berupa teks
2. Mengecek apakah ada huruf kapital (Jika menggunakan *lower case*). Jika ada maka ke tahap 3, jika tidak maka algoritma selesai.
3. Lakukan perubahan string lower pada setiap string.
4. Algoritma selesai

2.2.4 *Stopword Removal*

Proses pembuangan *stopword* dimaksudkan untuk mengetahui suatu kata masuk ke dalam *stopword* atau tidak. Stoplist berisi kumpulan kata yang 'tidak relevan', tetapi seringkali muncul dalam sebuah dokumen. Dengan kata lain, *stoplist* berisi sekumpulan *stopwords*. *Stopwords removal* adalah proses menghilangkan kata yang 'tidak relevan' dari sebuah dokumen teks dengan cara membandingkannya dengan *stoplist* yang ada [11].

Stopword biasanya berisi kata-kata yang sering kali muncul berupa kata sambung, kata depan, kata ganti, kata penghubung, dll. Namun artinya tidak deskriptif dan tidak memiliki keterkaitan dengan topik tertentu. Tahapan-tahapan yang dilakukan yaitu :

1. Membaca data masukan berupa teks
2. Mengecek apakah ada kata yang termasuk *Stopword Removal* yang ada dalam kamus, jika tidak ada maka algoritma selesai, jika ada
3. Menghapus semua kata yang termasuk *stopword removal*
4. Algoritma selesai.

2.2.5 *Stemming*

Stemming merupakan suatu proses yang mentransformasi kata-kata yang terdapat dalam suatu dokumen ke kata-kata akarnya (*root word*) dengan menggunakan aturan-aturan tertentu. Sebagai contoh, kata bersama, kebersamaan, menyamai, akan distem ke root wordnya yaitu “sama”. Proses stemming pada teks berbahasa Indonesia berbeda dengan stemming pada teks berbahasa Inggris. Pada teks berbahasa Inggris, proses yang diperlukan hanya proses menghilangkan sufiks. Sedangkan pada teks berbahasa Indonesia, selain sufiks, prefiks, dan konfiks juga dihilangkan [12]. Pada umumnya kata dasar pada bahasa Indonesia terdiri dari kombinasi:

Prefiks 1 + ***Prefiks 2*** + ***Kata dasar*** + ***Sufiks 3*** + ***Sufiks 2*** + ***Sufiks 1***

Pada penelitian ini Algoritma yang digunakan dalam proses ini yaitu nazief dan andriani. Algoritma Stemming Bahasa Indonesia M. Adriani dan B Nazief ini mempunyai aturan imbuhan sendiri dengan model, seperti :

[[[AW+]AW+]AW+] Kata-Dasar [[+AK][+KK][+P]]

AW : Awalan

AK : Akhiran

KK : Kata Ganti kepunyaan

P : Partikel

Tanda kurung besar menandakan bahwa imbuhan adalah opsional [13]. Algoritma yang dibuat oleh Bobby Nazief dan Mirna Adriani ini memiliki tahap-tahap sebagai berikut:

1. Cari kata yang akan distem dalam kamus. Jika ditemukan maka diasumsikan bahwa kata tesebut adalah root word. Maka algoritma berhenti.
2. Inflection Suffixes (“-lah”, “-kah”, “-ku”, “-mu”, atau “-nya”) dibuang. Jika berupa particles (“-lah”, “-kah”, “-tah” atau “-pun”) maka langkah ini diulangi lagi untuk menghapus Possesive Pronouns (“-ku”, “-mu”, atau “-nya”), jika ada.
3. Hapus Derivation Suffixes (“-i”, “-an” atau “-kan”). Jika kata ditemukan di kamus, maka algoritma berhenti. Jika tidak maka ke langkah 3a
    a. Jika “-an” telah dihapus dan huruf terakhir dari kata tersebut adalah “-k”, maka “-k” juga ikut dihapus. Jika kata tersebut ditemukan dalam kamus maka algoritma berhenti. Jika tidak ditemukan maka lakukan langkah 3b.
    b. Akhiran yang dihapus (“-i”, “-an” atau “-kan”) dikembalikan, lanjut ke langkah 4.
4. Hapus Derivation Prefix. Jika pada langkah 3 ada sufiks yang dihapus maka pergi ke langkah 4a, jika tidak pergi ke langkah 4b.
    b. Periksa tabel kombinasi awalan-akhiran yang tidak diijinkan. Jika ditemukan maka algoritma berhenti, jika tidak pergi ke langkah 4b.
    c. For i = 1 to 3, tentukan tipe awalan kemudian hapus awalan. Jika root word belum juga ditemukan lakukan langkah 5, jika sudah maka

algoritma berhenti. Catatan: jika awalan kedua sama dengan awalan pertama algoritma berhenti.

5. Melakukan Recoding.
6. Jika semua langkah telah selesai tetapi tidak juga berhasil maka kata awal diasumsikan sebagai root word. Proses selesai.

Tipe awalan ditentukan melalui langkah-langkah berikut:

1. Jika awalannya adalah: “di-”, “ke-”, atau “se-” maka tipe awalannya secara berturut-turut adalah “di-”, “ke-”, atau “se-”.
2. Jika awalannya adalah “te-”, “me-”, “be-”, atau “pe-” maka dibutuhkan sebuah proses tambahan untuk menentukan tipe awalannya.
3. Jika dua karakter pertama bukan “di-”, “ke-”, “se-”, “te-”, “be-”, “me-”, atau “pe-” maka berhenti.
4. Jika tipe awalan adalah “none” maka berhenti. Jika tipe awalan adalah bukan “none” maka awalan dapat dilihat pada Tabel 2. Hapus awalan jika ditemukan.

Tabel 2. 1 Kombinasi Awalan Yang Tidak Diajukan

Awalan	Akhiran yang tidak diijinkan
be-	-i
di-	-an
ke-	-i, -kan
me-	-an
se-	-i, -kan

Tabel 2. 2 Cara Menentukan Awalan Untuk Kata Yang Diawali Dengan “te-“

Following Characters				Tipe Awalan
Set 1	Set 2	Set 3	Set 4	
“-r-“	“-r-“	-	-	none
“-r-“	Vowel	-	-	ter-luluh
“-r-“	not (vowel or “-r-”)	“-er-“	vowel	ter
“-r-“	not (vowel or “-r-”)	“-er-“	not vowel	ter-
“-r-“	not (vowel or “-r-”)	not “-er-“	-	ter
not (vowel or “-r-”)	“-er-“	vowel	-	none
not (vowel or “-r-”)	“-er-“	not vowel	-	te

Tabel 2. 3 Jenis Awalan Berdasarkan Tipe Awalannya

Tipe Awalan	Awalan yang harus dihapus
di-	di-
ke-	ke-
se-	se-
te-	te-
ter-	ter-
ter-luluh	ter

2.2.6 Metode Pembobotan Kata TF-Idf

Term Frequency – Inverse Document Frequency digunakan untuk menentukan nilai frekuensi sebuah kata di dalam banyaknya dokumen. Perhitungan statistik numerik yang dimaksudkan untuk mencerminkan betapa pentingnya dan seberapa relevannya sebuah kata di dalam sebuah dokumen[14].

Pembobotan diperoleh dari frekuensi jumlah kemunculan sebuah kata yang terdapat di dalam sebuah dokumen, *term frequency* (tf). Sebuah kata atau jumlah kemunculan *term* di dalam koleksi dokumen, *inverse document frequency* (idf). TF-IDF dapat berhasil digunakan dalam penyaringan di berbagai bidang, termasuk *text summarization* dan klasifikasi[14].

Bobot suatu istilah semakin besar jika istilah tersebut sering muncul dalam suatu dokumen dan semakin kecil jika istilah tersebut muncul dalam banyak dokumen [14].

Nilai idf sebuah *term* (kata) dapat dihitung menggunakan persamaan (2.1) berikut :

$$IDF_t = \log(N/df) \quad (2.1)$$

Untuk menghitung bobot (W) masing-masing dokumen terhadap setiap *term* (kata) dapat menggunakan persamaan (2.2) berikut:

$$W_{dt} = tf_{dt} * IDF_t \quad (2.2)$$

Dimana:

W = bobot dokumen ke-d terhadap kata ke-*t*

d = dokumen ke-*d*

t = kata ke-*t*

tf = banyaknya kata yang dicari pada sebuah dokumen

N = total dokumen

df = banyak dokumen yang mengandung tiap kata

Metode ini mampu menghitung bobot setiap *term* yang ada pada dokumen. Namun dalam metode ini tidak ada solusi untuk mengatasi resiko adanya redudansi (kemiripan) *term* yang dihasilkan dalam dokumen.

2.2.7 Normalisasi Bobot

Normalisasi dilakukan terhadap vektor fitur dokumen untuk menghilangkan pengaruh anggapan bahwa dokumen panjang lebih relevan dibandingkan dokumen pendek. Dengan normalisasi ini dapat membantu menormalkan batas nilai dengan melakukan standarisasi nilai ke dalam interval 0 sampai dengan 1. Sehingga dapat dituliskan sebagai berikut :

$$w_{ij} = \frac{tf_{ij}\left(\log\left(\frac{N}{n}\right)+1\right)}{\sqrt{\sum_{k=1}^{r}(tf_{ij})^2 \text{ x}\left(\log\left(\frac{N}{n}\right)+1\right)^2}} \quad (2.3)$$

2.3 Metode Klasifikasi *Support Vector Machine* (SVM)

SVM dikembangkan oleh Boser, Guyon, Vapnik pada tahun 1992. SVM merupakan suatu teknik untuk menemukan hyperplane yang bisa memisahkan dua set data dari dua kelas yang berbeda (Vapnik, 1999). *Hyperplane* adalah garis batas pemisah data antar kelas, sedangkan *margin* adalah jarak antara hyperplane dengan data terdekat pada masing-masing kelas. Adapun data terdekat dengan *hyperplane* pada masing-masing kelas inilah yang disebut *support vector*. Karena konsep awal SVM adalah untuk mengatasi masalah klasifikasi dua kelas maka, ada dua metode diusulkan agar SVM bisa digunakan untuk klasifikasi *multi-class* dengan pendekatan mengombinasikan beberapa *binary classifier* atau menggabungkan semua data yang terdiri dari beberapa kelas ke dalam permasalahan optimasi[15].

Dalam linear Support Vector Machine permisah merupakan fungsi linear. Data latih dinyatakan oleh (x_i, y_i) dan x_i = { x_1, x_2, ... x_{iq}} merupakan atribut (fitur) set untuk data latih kelas ke-i. Untuk yi € {-1,1} menyatakan label kelas. pendefinisian persamaan suatu *hyperplane* pemisah yang dituliskan dengan:

$$w * x_i + b = 0 \quad (2.4)$$

Data x_i yang terbagi ke dalam dua kelas, yang termasuk kelas -1 (sampel negatif) didefinisikan sebagai vektor yang memenuhi pertidaksamaan (2.5) berikut.

$$w * x_i + b < 0 \text{ untuk } y_i = -1 \quad (2.5)$$

Sedangkan yang termasuk kelas +1 (sampel positif) memenuhi pertidaksamaan (2.6) berikut.

$$w * x_i + b > 0 \text{ untuk } y_i = +1 \quad (2.6)$$

Dimana:

x_i = data *input*

y_i = label yang diberikan

w = nilai dari bidang normal

b = posisi bidang relatif terhadap pusat koordinat

Parameter w dan b adalah parameter yang akan dicari nilainya. Bila label data y_i = -1, maka pembatas menjadi persamaan (2.7) berikut:

$$w * x_i + b \leq -1 \quad (2.7)$$

Bila label data y_i = +1, maka pembatas menjadi persamaan (2.8) berikut:

$$w * x_i + b \geq +1 \quad (2.8)$$

Margin terbesar dapat dicari dengan cara memaksimalkan jarak antar bidang pembatas kedua kelas dan titik terdekatnya, yaitu 2/|w|. Hal ini dirumuskan sebagai permasalahan *quadratic programming* (QP) *problem* yaitu mencari titik minimal persamaan (2.9) dengan memperhatikan persamaan (2.10) berikut:

$$min\ \tau(w) = \frac{1}{2}||w||^2 \quad (2.9)$$

$$y_i(w * x_i + b) - 1 \geq 0, (i = 1, .., n) \quad (2.10)$$

Permasalahan ini ini dapat dipecahkan dengan berbagi teknik komputasi. Lebih mudah diselesaikan dengan mengubah persamaan (2.9) ke dalam fungsi *Lagrangian* pada persamaan (2.11), dan menyederhanakannya menjadi persamaan (2.12) berikut:

$$L(w, b, a) = \frac{1}{2}||w||^2 - \sum_{i=1}^{n} a_i(y_i\big((w^T x_i + b) - 1\big)\big) \quad (2.11)$$

$$L(w, b, a) = \frac{1}{2}||w||^2 - \sum_{i=1}^{n} a_i y_i(w^T x_i + b) + \sum_{i=1}^{n} a_i \quad (2.12)$$

Dimana a_i adalah *lagrange multiplier* yang bernilai nol atau positif ($a_i \geq 0$). Nilai optimal dari persamaan (2.12) dapat dihitung dengan meminimalkan L terhadap w, b dan a. Dapat dilihat pada persamaan (2.13) sampai (2.15) berikut:

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^{n} a_i\, y_i x_i = 0 \quad (2.13)$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{n} a_i\, y_i = 0 \quad (2.14)$$

$$\frac{\partial L}{\partial a} = \sum_{i=1}^{n} a_i y_i (w^T x_i + b) - \sum_{i=1}^{n} a_i = 0 \quad (2.15)$$

Maka masalah *Lagrange* untuk klasifikasi dapat dinyatakan pada persamaan (2.16) berikut:

$$Min\ L(w, b, a) = \frac{1}{2}||w||^2 - \sum_{i=1}^{n} a_i y_i (w^T x_i + b) - \sum_{i=1}^{n} a_i \quad (2.16)$$

Dengan memperhatikan persamaan (2.17) dan (2.18) berikut:

$$w - \sum_{i=1}^{n} a_i\, y_i x_i = 0 \quad (2.17)$$

$$\sum_{i=1}^{n} a_i y_i = 0 \quad (2.18)$$

Model persamaan (2.16) diatas merupakan model primal *Lagrange.* Sedangkan dengan memaksimalkan L terhadap a_i, persamannya menjadi persamaan (2.19) berikut:

$$Max \sum_{i=1}^{n} a_i - \frac{1}{2} \sum_{i=1,j=1}^{n} a_i a_j y_i {y_j}^T x_i x_j^T \quad (2.19)$$

Dengan memperhatikan persamaan (2.20) berikut:

$$\sum_{i=1}^{n} a_i y_i = 0,\ a_i \geq 0\ (i, j = 1, .., n) \quad (2.20)$$

Untuk mencari nilai x_i dan y_i dapat dilakukan ketika sudah didapatkan nilai tiap kata (*term*) dari pembobotan tf-idf dan inisialisasi kelas . Hasil dari pembobotan tf-idf diubah ke dalam bentuk format data svm, sedangkan data kelas menjadi label data svm.

Untuk mendapatkan nilai a_i, langkah pertama adalah mengubah setiap abstrak menjadi nilai vektor (*support vector*) = $\binom{x}{y}$. Kemudian nilai vektor dari setiap abstrak dimasukkan ke persamaan (2.21) kernel *trick phi* berikut.

$$\phi \begin{bmatrix} x \\ y \end{bmatrix} = \begin{cases} \sqrt{x_n^2 + y_n^2} > 2\ maka \begin{bmatrix} \sqrt{x_n^2 + y_n^2} - x + |x - y| \\ \sqrt{x_n^2 + y_n^2} - y + |x - y| \end{bmatrix} \\ \sqrt{x_n^2 + y_n^2} \leq 2\ maka \begin{bmatrix} x \\ y \end{bmatrix} \end{cases} \quad (2.21)$$

Nilai x didapatkan dari persamaan (2.22) kernel linear untuk x berikut:

$$\sum_{i=1,j=1}^{n} x_i x_j^T\ , (i, j = 1, .., n) \quad (2.22)$$

Nilai y didapatkan dari persamaan (2.23) kernel linear untuk y berikut:

$$\sum_{i=1,j=1}^{n} y_i y_j^T , (i,j = 1,..,n) \quad (2.23)$$

Untuk mendapatkan jarak tegak lurus yang optimal dengan mempertimbangkan vektor positif, maka hasil perhitungan dari substitusi nilai x dan nilai y ke persamaan (2.21) diberi nilai bias = 1 . Kemudian cari parameter a_i, dengan terlebih dahulu mencari nilai fungsi setiap abstrak menggunakan persamaan (2.24), lalu mencari nilai a_i pada persamaan linear menggunakan persamaan (2.25) dengan memperhatikan $i,j = 1,..,n$ berikut:

$$\sum_{i=1,j=1}^{n} a_i S_i^T S_j \quad (2.24)$$

$$\sum_{i=1,j=1}^{n} a_i S_i^T S_j = y_i \quad (2.25)$$

Setelah parameter a_i didapatkan, kemudian masukkan ke persamaan (2.26) berikut:

$$\widetilde{W} = \sum_{i=1}^{n} a_i S_i \quad (2.26)$$

Hasil yang didapatkan menggunakan persamaan (2.26), selanjutnya digunakan persamaan (2.27) untuk mendapatkan nilai w dan b:

$$y = wx + b \quad (2.27)$$

Sedemikian sehingga didapatkanlah nilai w dan nilai b atau nilai *hyperplane* untuk mengklasifikasikan kedua kelas.

Sebuah fungsi bisa menjadi fungsi kernel jika memenuhi Teorema Mercer, yang menyatakan bahwa matriks kernel yang dihasilkan harus bersifat semi *positive* semi *definite*[3]. berikut ini adalah beberapa fungsi kernel yang umum digunakan yaitu:

a. Kernel linier

$K(xi,x) = xi^T x$

b. Polynomial

$K(xi,x) = (\gamma . xi^T x + r)^p , \gamma > 0$

c. Radial basis function

$K(xi,x) = exp(-\gamma \left| xi - x \right|^2), \gamma > 0$

d. Sigmoid kernel

$K(xi,x) = tanh(\gamma xi^T + r)$.

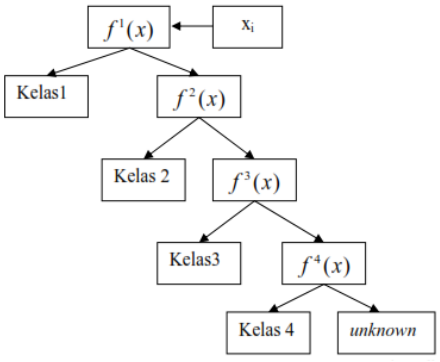
2.3.1 Multiclass SVM dengan Metode ”one-against-all”

SVM saat pertama kali diperkenalkan oleh Vapnik, hanya dapat mengklasifikasikan data ke dalam dua kelas (klasifikasi biner). Namun, penelitian lebih lanjut untuk mengembangkan SVM sehingga bisa mengklasifikasi data yang memiliki lebih dari dua kelas, terus dilakukan. Ada dua pilihan untuk mengimplementasikan *multi class* SVM yaitu dengan menggabungkan beberapa SVM biner atau menggabungkan semua data yang terdiri dari beberapa kelas ke dalam sebuah bentuk permasalah optimasi. Namun, pada pendekatan yang kedua permasalahan optimasi yang harus diselesaikan jauh lebih rumit. Berikut ini adalah salah satu metode yang umum digunakan untuk mengimplementasikan *multiclass* SVM yaitu dengan pendekatan *”one-against-all”.*

Dengan menggunakan metode ini, dibangun k buah model SVM biner (k adalah jumlah kelas). Contohnya, terdapat permasalahan klasifikasi dengan 4 buah kelas. Untuk pelatihan digunakan 4 buah SVM biner seperti pada tabel 1 dan penggunannya dalam mengklasifikasi data baru dapat dilihat pada tabel 2.4 [16].

Tabel 2. 4 Klasifikasi 4 Kelas

$y\ i=1$	$y\ i=-1$	Hipotesis
Kelas 1	Bukan Kelas 1	$f^1(x) = (w^1)\ x+\ b^1$
Kelas 2	Bukan Kelas 2	$f^2(x) = (w^2)\ x+\ b^2$
Kelas 3	Bukan Kelas 3	$f^3(x) = (w^3)\ x+\ b^3$
Kelas 4	Bukan Kelas 4	$F^4(x) = (w^4)\ x+\ b^4$



Gambar 2. 1 Diagram Keputusan *One-Againts-All*

2.4 Software Aplikasi

2.4.1 JAVA

a. Sebuah Bahasa Pemrograman

sebuah bahasa pemrograman, Java dapat membuat seluruh bentuk aplikasi, *desktop, web* dan lainnya, sebagaimana dibuat dengan menggunakan bahasa pemrograman konvensional yang lain.

Java adalah bahasa pemrograman yang berorientasi objek (OOP) dan dapat dijalankan pada berbagai *platform* sistem operasi. Perkembangan Java tidak hanya terfokus pada satu sistem operasi, tetapi dikembangkan untuk berbagai sistem operasi dan bersifat *open source*.

b. Sebuah *Development Environment*

Sebagai sebuah peralatan pembangun, teknologi Java menyediakan banyak *tools* : *compiler, interpreter,* penyusun dokumentasi, paket kelas dan sebagainya.

c. Sebuah Aplikasi

Aplikasi dengan teknologi Java secara umum adalah aplikasi serba guna yang dapat dijalankan pada seluruh mesin yang memiliki *Java Runtime Environment* (JRE).

d. Sebuah *Deployment Environment*

Terdapat dua komponen utama dari *Deployment Environment*. Yang pertama adalah JRE, yang terdapat pada paket J2SDK, mengandung kelas–kelas untuk semua paket teknologi Java yang meliputi kelas dasar dari Java, komponen GUI dan sebagainya. Komponen yang lain terdapat pada Web Browser. Hampir seluruh Web Browser komersial menyediakan *interpreter* dan *runtime environment* dari teknologi Java [17].

2.4.2 MySQL

MySQL termasuk dalam kategori *database management system*, yaitu suatu database yang terstruktur dalam pengolahan dan penampilan datanya MySQL merupakan database yang bersifat *client server*, dimana data diletakkan di *server* yang bisa diakses melalui komputer *client.* Pengaksesan dapat dilakukan apabila komputer telah terhubung dengan *server*, berbeda dengan database desktop, dimana segala pemrosesan data harus dilakukan pada komputer yang bersangkutan.

MySQL dapat juga dikatakan sebagai *Relational Database Management System* (RDBMS), yaiut hubungan antar tabel yang berisi data-data pada suatu database, dengan demikian dapat mempercepat pencarian suatu data. Tabel-tabel tersebut di-*link* oleh suatu relasi yang memungkinkan kombinasi data dari beberapa tabel ketika user menginginkan tampilannya informasi dari suatu database.

MySQL merupakan database yang dikembangkan dari bahasa SQL (*Structured Query Language*), SQL merupakan bahasa terstruktur yang digunakan untuk interaksi antara script program dengan database server dalam hal pengolahan data. Dengan SQL, maka dapat dibuat tabel yang akan diisi data, memanipulasi data seperti menambah, menghapus dan mengupdate data.

Sebuah model SQL terdiri dari tabel, baris, field dan elemen. Setiap user pada suatu sistem dapat memiliki database yang terpisah dan independen antara yang satu dengan yang lainnya. Suatu tabel pada database memiliki kolom dan baris yang memiliki data. Pada tabel tersebut tidak boleh ada dua data atau lebih yang sama persis.

2.5 Metodologi penelitian

2.5.1 Metode Pengumpulan Data

Studi Literatur yaitu pengumpulan data dengan mempelajari masalah yang akan dibahas serta mengumpulkan literatur. Sumber-sumber yang diperlukan berupa e-book, jurnal, tutorial dan beberapa informasi yang relevan dengan penelitian.

2.5.2 Model Perangkat Lunak *Prototype*

Pembuatan prototipe evolusioner didasarkan pada ide pengembangan implementasi awal, memperlihatkannya kepada user untuk dikomentari, dan menyempurnakannya melalui banyak tahap sampai sistem yang memadai telah dikembangkan. Pendekatan terhadap pengembangan ini pada awalnya digunakan untuk sistem-sistem (seperti AI) yang sulit atau tidak mungkin di spesifikasi.

Tidak ada kegiatan spesifikasi, pengembangan dan validasi yang terpisah, alih-alih kegiatan-kegiatan ini dilakukan pada saat yang bersamaan dengan umpan balik yang cepat untuk masing-masing kegiatan. Ada dua jenis pengembangan evalusioner :

1. *Pengembangan eksploratori.* Tujuan proses ini adalah bekerja dengan pelanggan untuk menyelidiki persyaratan mereka dan mengirimkan sistem akhir. Pengembangan dimulai dengan bagian-bagian sistem yang dipahami. Sistem berubah dengan adanya tambahan fitur-fitur baru sesuai usulan pelanggan.
2. *Prototipe yang dapat dibuang (throw-away).* Tujuan pengembangan evolusioner adalah untuk memahami persyaratan pelanggan dan dengan demikian mengembangkan definisi persyaratan yang lebih baik untuk sistem. Prototipe berkonsentrasi pada eksperimen, dengan persyaratan pelanggan yang tidak dipahami dengan baik [6].

Pengembang perangkat lunak mungkin merasa tidak pasti tentang efisiensi suatu algoritma yang akan digunakan dalam pengembangan perangkat lunak, atau juga merasa tidak pasti akan kemampuan perangkat lunak untuk beradaptasi dengan sistem operasi yang akan digunakan, atau merasa tidak pasti akan bentuk interaksi manusia-komputer yang akan digunakan. Dalam kasus-kasus seperti ini dan dalam banyak situasi yang lain, paradigma pembuatan prototipe (*prototyping*) mungkin menawarkan pendekatan yang paling baik.

Meskipun pembuatan prototipe dapat digunakan sebagai model proses yang berdiri sendiri, pembuatan prototipe lebih umum digunakan sebagai teknik yang dapat diimplementasikan di dalam konteks setiap model proses perangkat lunak. Dalam hal ini, tidak terlalu peduli dengan dimana ia diterapkan, paradigma pembuatan prototipe seringkali membantu tim pengembang perangkat lunak dan para *stakeholder* untuk memahami lebih baik apa yang akan dikembangkan saat spesifikasi kebutuhan belum jelas[18].

2.6 UML (*Unified Modeling Language*)

UML *(Unified Modeling Language)* adalah notasi yang lengkap untuk membuat visualisasi model suatu sistem. Sistem berisi informasi dan fungsi, tetapi secara normal digunakan untuk memodelkan sistem komputer. Di dalam pemodelan objek guna menyajikan sistem yang berorientasi objek kepada orang lain, akan sangat sulit di lakukan dalam bentuk kode bahasa pemrograman [19].

UML disebut sebagai bahasa pemodelan bukan metode. Bahasa pemodelan (sebagian besar grafik) merupakan notasi model yang digunakan untuk mendesain secara cepat. Bahasa pemodelan merupakan bagian terpenting dari metode. UML merupakan bahasa standar untuk penulisan *blueprint software* yang digunakan untuk visualisasi, spesifikasi, pembentukan dan pendokumentasia

+n alat-alat dari sistem perangkat lunak. UML biasanya disajikan dalam diagram atau gambar yang meliputi *class* beserta atribut dan operasinya, serta hubungan antar kelas. UML terdiri dari banyak diagram diantaranya *use case diagram, activity diagram, class diagram* dan *sequence diagram.*

2.6.1 *Use Case Diagram*

Dalam konteks UML , tahap konseptualisasi dilakukan dengan pembuatan *use case diagram* yang sesungguhnya merupakan deskripsi peringkat tinggi bagaimana perangkat lunak (aplikasi) akan digunakan oleh penggunanya.

Use case diagram merupakan deskripsi lengkap tentang interaksi yang terjadi antara para actor denga sistem[20]. Dalam hal ini, setiap objek yang berinteraksi dengan sistem merupakan actor untuk sistem, sementara *use case* merupakan deskripsi lengkap tentang bagaimana sistem berperilaku kepada actornya. Actor dalam *use case diagram* digambarkan sebagai ikon yang berbentuk manusia, sementara *use case* digambarkan sebagai elips yang berisi nama *use case*yang bersangkutan. Untuk mempermudah pemahaman, actor biasanya dituliskan sebagai kata benda, sementara *use case* biasanya dituliskan sebagai kata kerja.

2.6.2 *Activity Diagram*

Activity diagram pada dasarnya menggambarkan skenario secara grafis. Serta *activity diagram* cukup serupa dengan diagram alir (*flow chart),* yang membedakan mungkin hanya *swimlane* yang menunjukkan suatu *state* berada pada objek/kelas tertentu. Keunggulan dari *activity diagram* adalah bahwa diagram tersebut lebih mudah dipahami dibanding skenario. Selain itu, dengan menggunakan *activity diagram,* kita dapat melihat dibagian manakah sistem dari suatu skenario akan berjalan[20].

2.6.3 *Class Diagram*

Diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Class diagram menggambarkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain seperti containment , pewarisan, asosiasi, dan lain-lain.

2.6.4 *Sequence Diagram*

Diagram sekuen menggambarkan kelakuan/perilaku objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Objek-objek yang berkaitan dengan proses berjalannya operasi diurutkan dari kiri ke kanan berdasarkan waktu terjadinya dalam pesan yang terurut. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah use case beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.

2.7 Akurasi

Untuk menghitung akurasi dari SVM *Multiclass* dilakukan perhitungan dengan rumus [21] :

$$Akurasi = \frac{jumlah\ yang\ diklasifikasi\ secara\ benar}{total\ sample\ testing\ yang\ diuji}$$