DAFTAR ISI

[BAB 2 1](#_Toc444702599)

[TINJAUAN PUSTAKA 1](#_Toc444702600)

[2.1 Mixed Type Audio 1](#_Toc444702603)

[2.2 Klasifikasi 1](#_Toc444702604)

[2.3 Ekstraksi Fitur 1](#_Toc444702605)

[2.4 Decision Tree 1](#_Toc444702606)

[2.5 Algoritma CART 1](#_Toc444702607)

[2.6 Pemrograman Berbasis Objek 1](#_Toc444702608)

[2.7 Unified Model Language 2](#_Toc444702609)

[2.8 Desktop Aplication 2](#_Toc444702610)

[2.9 C++ 2](#_Toc444702611)

# BAB 2

# TINJAUAN PUSTAKA



## Mixed Type Audio

## Klasifikasi

## Ekstraksi Fitur

## Decision Tree

## Algoritma CART

## Pemrograman Berbasis Objek

Pemrograman berorientasi objek adalah sebuah pendekatan untuk pembangunan perangkat lunak di mana struktur dari perangkat lunak didasari dari objek yang saling berinteraksi satu sama lain untuk menyelesaikan suatu tugas. Sudut pandang dari pemrograman berorientasi objek menyelesaikan masalah dengan mengunakan objek, persis dengan apa yang ada di dunia nyata. Pemrograman berorientasi objek (PBO) dirancang untuk menutup kelemahan dari pemrograman berorientasi prosedural yang sulit untuk dikelola dan *debug* ketika kode yang dibuat telah mencapai beribu-ribu baris. Maka diperkenalkanlah struktur pemrograman untuk memecah kode ke dalam beberapa segmen atau biasa disebut prosedur atau fungsi. Dengan struktur tersebut kode mengalami perbaikantetapi secara program dapat melakukan fungsionalitas yang lebih kompleks.

Berikut adalah konsep mendasar mengenai bahasa permrograman berorientasi objek.

1. Kelas

Kelas adalah struktur yang akan digunakan oleh objek yang berisikan atribut-atribut yang mendefiniskan data yang akan diolah dan juga berisikan *method* yang dapat berbentuk prosedur ataupun fungsi untuk dapat mengolah data. Dalam konteks PBO biasanya didasari dari pemodelan dari objek yang ada di dunia nyata dalam pemrograman untuk menyelesaikan masalah. Sebagai contoh, dilakukan pemodelan untuk sepeda untuk dijadikan sebuah kelas. Dalam sebuah sepeda misalkan terdiri dari roda, rangka, dan setir. Ketiga unsur tersebut bisa disebut sebagai atribut. Kemampuan dari sepeda itu bisa berjalan dan melompat. Kemampuan tersebut bisa disebut *method* yang ada di dalam kelas.

1. Objek

Objek adalah instansiasi dari sebuah *class*. Kemampuan yang bisa dilakukan oleh suatu objek bergantung apa dari isi yang didefinisikan dari *class* tersebut.

1. Abstraksi

Ketika membuat objek dalam aplikasi PBO, penting untuk menggunakan konsep dari abstraksi. Objek hanya berisi informasi yang relevan dengan konteks dari aplikasi. Contoh untuk pembangunan aplikasi pengiriman. Akan dibuat sebuah objek dengan nama produk dengan atribut ukuran dan berat. Warna akan menjadi informasi yang tidak berhubungan dan bisa diabaikan sebagi atribut dari objek produk. Di kasus lain seperti pembangunan aplikasi pencatatan pemesanan, warna menjadi informasi yang berhubungan dan bisa dijadikan atribut dari suatu objek.

1. Enkapsulasi

Enkapsulasi adalah sebuat perses yang tidak mengizinkan akses langsung kepada data. Jika ingin mengakases dari data tersebut, diharuskan menggunakan objek yang memiliki tanggung jawab untuk terhadap data tersebut. Menggunakan contoh sebelumnya, ketika akan menggunakan *method* berjalan tidak bisa dilakukan secara langsung. Kelas sepeda tersebut harus diinstansiasi terlebih dahulu menjadi objek sepeda sehingga bisa mengakses *method* berjalan.

1. *Polymorphism*  
   *Polymorphism* adalah kemempuan untuk merespon pesan yang sama dengan cara yang berbeda. Misalkan ada dua objek yaitu sepeda dan motor. Dalam objek tersebut memiliki kemampuan method yang sama yaitu berjalan. Ketika *method* berjalan dari kedua objek tersebut dipanggil. Kedua objek tersebut menampilkan rekasi yang berbeda, sepeda berjalan dengan cara mengayuh pedal, sedangkan motor berjalan dengan cara menarik gas.
2. *Inheritance*  
   *Inheritance* dalam PBO untuk mengklasifikasikan objek di dalam program berdasarkan karakteristik umum. Dengan *inheritance* pemrograman lebih mudah karena bisa mengkombinasikan karakteristik umum ke dalam kelas induk kemudian bisa diturunkan ke kelas anaknya. Contoh dibuat kelas kendaraan yang berisikan *method* berjalan. Kemudian dibuat kelas kendaraan darat. Dalam kelas kendaraan darat tidak perlu membuat *method* berjalan di kelas kendaraan darat cukup dengan melakukan *inheritance* atau pewarisan *method* yang ada di kelas kendaraan ke kelas kendaraan darat.

## Unified Model Language

*Unified modeling language* (UML) adalah bahasa pemodelan standar untuk pembangunan perangkat lunak ataupun sistem yang dibangun dengan pendekatan  
*object-oriented.* Dalam merancang sistem, model memiliki peranan penting yaitu untuk mengelola kompleksitas. Pemodelan membantu untuk fokus menghubungkan aspek penting dari perancangan sistem. Sebuah model adalah sebuah abstraksi dari benda nyata. Model yang dibangun adalah penyederhanaan dari sistem yang dibangun. Memberikan maksud dan kelangsungan hidup sistem untuk dipahami, dievaluasi, dikritisasi bahkan lebih cepat dibandingkan harus langsung ke sistem sebenarnya.

UML itu sendiri sudah mengalami beberapa generasi, dimulai dari UML 1.0, 1.5, dan yang terakhir 2.0. Versi pertama dari UML memberikan komuniasi rancangan secara jelas, menyampaikan esensi dari suat perancangan, bahkan menangkap peta kebutuhan fungsional dari solusi perangkat lunak. Bagaimanapun, dunia berubah secara fundamental dengan pengenalan pemodelan sistem, dibanding hanya pemodelan perangkat lunak. Dari perubahan itulah yang membuat versi UML itu sendiri berkembang hingga kini 2.0. Secara keseluruhan ada 13 diagram yang dapat dimanfaatkan dari UML 2.0 ini.



## Desktop Aplication

*Desktop Application* (aplikasi desktop) adalah sebuah program mandiri yang mengerjakan sejumlah tugas-tugas yang telah terdefinisi sebelumnya di bawah kendali pengguna [3]. Aplikasi desktop dijalankan dari *drive* lokal dan tidak membutuhkan sebuah jaringan atau konektifitas untuk beroperasi atau berfungsi dengan baik.

Dalam hal fungsionalitas, aplikasi desktop mempunyai lebih banyak kelebihan. Walaupun aplikasi web belakangan ini mulai menyusul, akan tetapi masih tertinggal. Kinerja dari aplikasi desktop, selain dari pemograman dari pengembang program tersebut, tergantung pada pengguna itu sendiri. Sementara kinerja aplikasi web tergantung pada berbagai hal. Seperti konektifitas pengguna (yakni koneksi internet), *web server* (*host*), dan peramban web.

## C++

## Database Management System (DBMS)

DBMS (*Database Management System*) merupakan sistem perangkat lunak yang memungkinkan pengguna *database* (*database user*) untuk memelihara, mengontrol dan mengakses data secara praktis dan efisien. Dengan kata lain semua akses ke basis data akan ditangani oleh DBMS. Ada beberapa fungsi yang harus ditangani DBMS yaitu mengolah pendefinisian data, menangani permintaan pemakai untuk mengakses data, memeriksa keamanan dan integritas data yang didefinisikan oleh DBA (*Database Administrator*), menangani kegagalan dalam pengaksesan data yang disebabkan oleh kerusakan sistem maupun disk dan menangani unjuk kerja semua fungsi secara efisien [10].

Tujuan utama dari DBMS adalah untuk memberikan tinjauan abstrak data kepada pengguna. Jadi sistem menyembunyikan informasi tentang bagaimana data disimpan, dipelihara dan tetap bisa diambil (diakses) secara efisien. Pertimbangan efisien di sini adalah bagaimana merancang struktur data yang kompleks tetapi masih tetap bisa digunakan oleh pengguna awam tanpa mengetahui kompleksitas strukturnya.

## MySQL

*MySQL* adalah sistem manajemen *database* yang digunakan untuk menyimpan data dalam tabel terpisah dan menempatkan semua data dalam satu gudang besar [10]. Struktur *database* disusun dalam file fisik dioptimalkan untuk kecepatan. Model logis, dengan benda-benda seperti *database*, tabel, baris, dan kolom, menawarkan lingkungan pemrograman yang fleksibel.

*SQL* bagian dari *MySQL* singkatan dari "*Structured Query Language*". *SQL* adalah bahasa standar yang paling umum digunakan untuk mengakses database. *SQL* didefinisikan oleh *ANSI / ISO* *Standard SQL*. Standar *SQL* telah berkembang sejak tahun 1986 dan beberapa versi dari *SQL* telah ada, salah satunya adalah *MySQL*.