



UJIAN AKHIR SEMESTER GASAL 2025/2026

Mata Kuliah	: Pemrograman Berorientasi Objek
SKS	: 4 SKS
Prodi/Angkatan	: D4 Manajemen Informatika / 2024
Dosen	: - M Adamu Islam Mashuri, S.Tr.T., M.Tr.Kom. - Binti Kholidah, S.Kom., M.Tr.Kom - Rosita, S.Pd., M.Pd. - Faris Abdi El Hakim, S.Kom., M.Tr.Kom
Waktu	: 2 Minggu
Sifat Ujian	: Demo Proyek

1. Objektif:

- Tujuan Umum:
 - Mengembangkan aplikasi berbasis OOP (Object-Oriented Programming) menggunakan bahasa pemrograman pilihan Anda.
 - Memahami dan menerapkan prinsip-prinsip OOP seperti encapsulation, inheritance, dan polymorphism.
 - Membuat dokumentasi lengkap, termasuk class diagram, penjelasan fitur, dan alur aplikasi.

2. Perencanaan Aplikasi:

- Pilihan Proyek:
 - Anda bebas memilih jenis aplikasi, misalnya game sederhana (contoh: menggunakan pygame), aplikasi pengelolaan data, atau simulasi berbasis GUI (Menggunakan Tkinter).
- Fitur Utama:
 - Implementasi konsep OOP:
 - **Encapsulation:** Penggunaan atribut dan metode dengan pengaturan akses (private, protected, public).
 - **Inheritance:** Demonstrasi pewarisan antar kelas untuk memperluas fungsionalitas.
 - **Polymorphism:** Penggunaan metode yang memiliki perilaku berbeda di kelas turunan.
 - Penggunaan struktur data atau algoritma pendukung seperti list, tree, atau graph, sesuai kebutuhan aplikasi.
 - Implementasi interaksi pengguna yang intuitif.

3. Pelaksanaan Project:

- Perancangan:
 - Buat **diagram class** untuk merancang hubungan antar kelas di aplikasi Anda.
 - Rancang alur aplikasi yang menggambarkan bagaimana pengguna berinteraksi dengan sistem.

- **Implementasi:**
 - Implementasikan kode sesuai dengan diagram class yang telah dirancang.
 - Gunakan struktur OOP yang baik untuk memastikan aplikasi modular dan mudah dipahami.
 - Jika memilih membuat game, gunakan framework seperti pygame atau yang setara.
- **Dokumentasi**
 - Lengkapi aplikasi Anda dengan komentar yang menjelaskan fungsi utama dan alur kerja setiap kelas atau metode.
 - Tulis laporan yang mencakup:
 - Penjelasan konsep OOP yang digunakan dalam aplikasi Anda.
 - Diagram class yang mendukung implementasi aplikasi.
 - Penjelasan fitur dan cara penggunaan aplikasi.
 - Screenshot aplikasi Anda yang menunjukkan fungsionalitas utama.

4. Evaluasi:

Konsep OOP (30%)

Object-Oriented Programming (OOP) adalah paradigma pemrograman yang berbasis pada konsep "objek," yang merepresentasikan data dan fungsionalitas. Berikut adalah penjelasan tentang prinsip-prinsip utama OOP yang akan dinilai:

1. Encapsulation (10%)

Encapsulation adalah proses menyembunyikan detail implementasi suatu objek dan hanya menyediakan antarmuka tertentu untuk berinteraksi dengan objek tersebut.

- **Contoh:**

- Menggunakan **modifier akses** seperti private, protected, dan public untuk mengatur visibilitas atribut atau metode.
- Membuat metode getter dan setter untuk mengakses atau mengubah nilai atribut secara aman.

2. Inheritance (10%)

Inheritance adalah mekanisme yang memungkinkan sebuah kelas (kelas turunan/child class) mewarisi atribut dan metode dari kelas lain (kelas induk/parent class).

- **Contoh:**

- Kelas Animal dengan atribut nama dan metode berjalan(), yang diwarisi oleh kucing dan anjing
- Kelas turunan dapat menambahkan fungsionalitas baru atau mengubah perilaku dengan **method overriding**.

3. Polymorphism (10%)

Polymorphism memungkinkan metode atau objek untuk memiliki bentuk atau perilaku yang berbeda berdasarkan konteks penggunaannya.

- **Contoh:**

- **Method Overloading:** Beberapa metode dengan nama yang sama tetapi parameter berbeda dalam satu kelas.
- **Method Overriding:** Metode pada kelas turunan yang mengubah implementasi dari metode kelas induk.

Fungsionalitas Aplikasi dan Keterbacaan Kode (20%):

- Aplikasi harus dapat berjalan sesuai dengan fitur yang direncanakan. Bug akan mengurangi nilai.
- Struktur kode harus rapi, mengikuti standar penulisan, dan mudah dipahami. Tambahkan komentar untuk menjelaskan kodingan.

Kriteria Penilaian Dokumentasi (40%):

- **Kejelasan Class Diagram (15%):**
 - Diagram class harus mencerminkan desain aplikasi dan mencakup semua hubungan antar kelas.
- **Laporan Proyek (25%):**
 - Laporan harus mencakup penjelasan lengkap tentang konsep OOP yang diterapkan, diagram class, alur aplikasi, serta tantangan yang dihadapi selama pengembangan.

Tambahan Nilai untuk Kreativitas Aplikasi (30%)

Untuk mendorong kreativitas, proyek akan diberi nilai tambahan berdasarkan inovasi dan keunikan dalam ide dan implementasi aplikasi.

Kriteria Penilaian Kreativitas:

1. Keunikan Ide (10%)

- Ide aplikasi harus berbeda dari aplikasi umum.
- Contoh: Jika membuat game, buat game dengan konsep unik seperti edukasi berbasis cerita atau simulasi yang mendidik.

2. Integrasi Fitur Tambahan (10%)

- Penambahan fitur menarik di luar persyaratan dasar.
- Contoh: Statistik penggunaan aplikasi, leaderboard untuk game, atau fitur ekspor data dalam format tertentu.

3. Desain dan Interaksi Pengguna (10%)

- Desain antarmuka pengguna yang menarik dan intuitif.
- Contoh: Tata letak yang bersih, penggunaan warna yang sesuai, atau animasi interaktif yang meningkatkan pengalaman pengguna.

Instruksi Pengerjaan:

1. Pengumpulan:

- Unggah proyek Anda ke repositori Git yang telah ditentukan, dengan commit log yang jelas.
- Sertakan file dokumentasi dalam format PDF di dalam repositori.

2. Presentasi:

- Persiapkan presentasi singkat untuk menjelaskan aplikasi Anda, desain OOP, dan fitur yang diimplementasikan.
- Gunakan diagram class dan jalankan aplikasi sebagai bahan pendukung.

Selamat Mengerjakan dan Semoga Sukses

Surabaya, 26 Oktober 2025
Divalidasi oleh,

Unit Penjamin Mutu
Prodi D4 Manajemen Informatika



(Moeh Deny Pratama, S.Tr.Kom., M.Kom.)

Tim Serumpun

(Dodik Arwin Dermawan, S.ST., S.T., M.T.)