

FR.IA.02.	TPD - TUGAS PRAKTIK DEMONSTRASI
-----------	---------------------------------

Skema Sertifikasi (KKM/Okupasi/Klaster)	Judul	:	Programmer
	Nomor	:	SS/TI/01/LSP_POLITALA/I/2024
TUK	:	Sewaktu/Tempat Kerja/Mandiri*	
Nama Asesor	:		
Nama Asesi	:		
Tanggal	:		

*Coret yang tidak perlu

A. Petunjuk

1. Baca dan pelajari setiap instruksi kerja di bawah ini dengan cermat sebelum melaksanakan praktek
2. Klarifikasi kepada asesor kompetensi apabila ada hal-hal yang belum jelas
3. Laksanakan pekerjaan sesuai dengan urutan proses yang sudah ditetapkan
4. Seluruh proses kerja mengacu kepada SOP/WI yang dipersyaratkan (Jika Ada)

B. Skenario Tugas Praktik Demonstrasi

Kelompok Pekerjaan 1	No.	Kode Unit	Judul Unit
Melakukan Perancangan	1.	J.620100.009.02	Menggunakan Spesifikasi Program

Situation (S):

Andi, seorang siswa baru yang baru saja lulus sekolah menengah atas, sangat antusias untuk memulai perkuliahan. Ia pun memutuskan untuk mendaftarkan diri secara online. Setelah mengakses situs web pendaftaran, Andi mengisi formulir pendaftaran dengan lengkap, mulai dari data pribadi, hingga mengunggah berkas-berkas persyaratan seperti fotokopi ijazah dan transkrip nilai.

Setelah proses pendaftaran selesai, Andi kini memiliki akun untuk mengakses sistem informasi akademik. Dengan akun ini, ia dapat melihat jadwal perkuliahan, memilih mata kuliah yang ingin diambil, dan memasukkannya ke dalam Kartu Rencana Studi (KRS). Andi juga dapat memantau nilai-nilai yang telah diperoleh pada setiap mata kuliah, menghitung Indeks Prestasi Kumulatif (IPK), dan mencetak transkrip nilai.

Sebagai seorang mahasiswa, Andi merasa sangat terbantu dengan adanya sistem informasi akademik ini. Ia dapat mengurus segala keperluan akademiknya secara mandiri dan efisien, tanpa harus mengantre di bagian administrasi.

Diagram UML dan ERD untuk sebuah sistem informasi akademik yang memenuhi persyaratan berikut:

Skenario:

1. Seorang mahasiswa baru mendaftar secara online, termasuk mengunggah berkas.
2. Mahasiswa yang telah terdaftar dapat mengakses sistem untuk melihat jadwal kuliah, memilih mata kuliah, dan melihat nilai.
3. Mahasiswa dapat mencetak transkrip nilai secara online.

Task (T)

- Anda ditugaskan untuk melaksanakan pekerjaan yang dimulai dari:

No	Unit Kompetensi
----	-----------------

1	Menggunakan Spesifikasi Program
---	---------------------------------

- Laksanakan seluruh pekerjaan tersebut sesuai prosedur (SOP)

Action (A):

Langkah kerja untuk setiap *Task* diuraikan sebagai berikut.

1. Menggunakan Spesifikasi Program
 - a. Menggunakan metode pengembangan program
 - b. Menggunakan diagram program dan deskripsi program
 - c. Menerapkan hasil pemodelan ke dalam pengembangan program

Result (R):

Diagram yang harus dibuat:

1. Diagram Use Case: Gambarkan aktor (mahasiswa) dan use case yang relevan (misalnya, mendaftar, login, melihat jadwal).
2. Diagram Kelas: Identifikasi kelas-kelas utama (Mahasiswa, MataKuliah, Nilai) dan hubungan antar kelas.
3. Diagram Aktivitas: Gambarkan alur proses pendaftaran mahasiswa baru secara detail.
4. Diagram Sequence: Tunjukkan interaksi antara aktor dan sistem saat mahasiswa login dan melihat jadwal kuliah.
5. Entity Relationship Diagram (ERD): Representasikan entitas-entitas data dan relasinya dalam bentuk diagram.

Persyaratan Tambahan:

1. Detail: Setiap diagram harus dibuat dengan detail yang cukup, termasuk atribut, operasi, dan hubungan antar elemen.
2. Konsistensi: Pastikan semua diagram saling konsisten dan menggambarkan sistem yang sama.
3. Penjelasan: Jelaskan setiap diagram secara singkat dan berikan alasan untuk pilihan desain yang Anda buat.
4. Tools: Gunakan tools pemodelan UML yang sesuai (misalnya, Visual Paradigm, StarUML, Lucidchart) untuk membuat diagram.

Perlengkapan dan Peralatan :

1. Personal Komputer atau Laptop
2. Aplikasi pengolah kata
3. Aplikasi pengolah diagram

Waktu : 90 Menit

Kelompok Pekerjaan 2 Menulis Kode Sumber	No.	Kode Unit	Judul Unit
	1.	J.620100.016.01	Menulis Kode Dengan Prinsip Sesuai Guidelines dan Best Practices
	2.	J.620100.017.02	Mengimplementasikan Pemrograman Terstruktur
	3.	J.620100.018.02	Mengimplementasikan Pemrograman Berorientasi Objek
	4.	J.620100.019.02	Menggunakan Library atau Komponen Pre-Existing
	5.	J.620100.021.02	Menerapkan Akses Basis Data
	6.	J.620100.023.02	Membuat Dokumen Kode Program
	7.	J.620100.025.02	Melakukan Debugging

Situation (S):

Anda diminta untuk mengembangkan **bagian dari sistem informasi akademik yang memungkinkan mahasiswa:**

1. **Mendaftar:** Mengisi formulir pendaftaran dengan data pribadi dan mengunggah berkas persyaratan.
2. **Login:** Masuk ke sistem menggunakan akun yang telah dibuat.
3. **Melihat jadwal kuliah:** Melihat daftar mata kuliah yang diambil beserta jadwalnya.
4. **Mencetak transkrip nilai:** Mencetak transkrip matakuliah yang sudah diikuti mahasiswa.

Persyaratan:

1. **Pilihan Bahasa Pemrograman dan Framework:** Bebas memilih bahasa pemrograman (misalnya, Python, Java, PHP) dan framework yang sesuai (misalnya, Django, Spring Boot, Laravel).
2. **Database:** Gunakan database relasional seperti MySQL, PostgreSQL, atau SQL Server.
3. **Fitur Aplikasi:** Anda dapat menambahkan fitur-fitur berikut:
 - a. **Notifikasi:** Mengirimkan notifikasi kepada mahasiswa terkait jadwal kuliah.
 - b. **Laporan:** Menghasilkan laporan data mahasiswa dan nilai.
 - c. **Fitur lupa password:** Buat fungsi untuk mereset password mahasiswa.
 - d. **Fitur pencarian:** Perbolehkan mahasiswa mencari mata kuliah berdasarkan nama atau dosen pengampu.

Tahapan yang harus dilakukan:

1. Buat model data sesuai dengan skema database.
2. Implementasikan fungsi-fungsi untuk setiap *use case* (misalnya, fungsi untuk mendaftar, login, melihat jadwal).
3. Gunakan *library* atau framework yang telah dipilih untuk mempermudah pengembangan.
4. Pastikan kode mengikuti prinsip-prinsip pemrograman yang baik.
5. Buat dokumentasi kode yang jelas dan mudah dipahami, termasuk komentar dalam kode dan dokumentasi keseluruhan sistem.
6. Mulailah dengan bagian yang sederhana: Fokus pada fitur inti terlebih dahulu, kemudian tambahkan fitur-fitur tambahan secara bertahap.
7. Gunakan *version control*: Gunakan Git untuk mengelola perubahan kode dan kolaborasi tim.
8. Lakukan *refactoring*: Perbaiki struktur kode secara berkala untuk meningkatkan kualitas kode.
9. Program dibuat dengan menggunakan pemrograman berorientasi objek.

Task (T):

- Anda ditugaskan untuk melaksanakan pekerjaan yang dimulai dari:

No	Unit Kompetensi
1.	Menulis Kode Dengan Prinsip Sesuai Guidelines dan Best Practices
2.	Mengimplementasikan Pemrograman Terstruktur
3.	Mengimplementasikan Pemrograman Berorientasi Objek
4.	Menggunakan Library atau Komponen Pre-Existing
5.	Menerapkan Akses Basis Data
6.	Membuat Dokumen Kode Program
7.	Melakukan Debugging

- **Laksanakan seluruh pekerjaan tersebut sesuai prosedur (SOP)**

Action (A):

Langkah kerja untuk setiap *Task* diuraikan sebagai berikut.

1. Menulis Kode Dengan Prinsip Sesuai Guidelines dan Best Practices
 - a. Menerapkan coding guidelines dan best practices dalam penulisan program (kode sumber)
 - b. Menggunakan ukuran performansi dalam menuliskan kode sumber
2. Mengimplementasikan Pemrograman Terstruktur
 - a. Menggunakan tipe data dan control program
 - b. Membuat program sederhana
 - c. Membuat program Menggunakan prosedur dan fungsi
 - d. Membuat program menggunakan array
 - e. Membuat program untuk akses file
 - f. Mengkompilasi Program
3. Mengimplementasikan Pemrograman Berorientasi Objek
 - a. Membuat program berorientasi objek dengan memanfaatkan class
 - b. Menggunakan tipe data dan control program pada metode atau operasi dari suatu kelas
 - c. Membuat program dengan konsep berbasis objek
 - d. Membuat program object oriented dengan interface dan paket
 - e. Mengkompilasi Program
4. Menggunakan Library atau Komponen Pre-Existing
 - a. Melakukan pemilihan unit-unit reuse yang potensial
 - b. Melakukan integrasi library atau komponen pre-existing dengan source code yang ada
 - c. Melakukan pembaharuan library atau komponen pre existing yang digunakan
5. Menerapkan Akses Basis Data
 - a. Membuat berbagai operasi terhadap basis data
 - b. Membuat prosedur akses terhadap basis data
 - c. Membuat koneksi basis data
 - d. Menguji program basis data
6. Membuat Dokumen Kode Program
 - a. Melakukan identifikasi kode program
 - b. Membuat dokumentasi modul program
 - c. Membuat dokumentasi fungsi, prosedur atau method program
 - d. Mengenerate dokumentasi
7. Melakukan Debugging
 - a. Mempersiapkan kode program
 - b. Melakukan debugging
 - c. Memperbaiki program
8. Menulis Kode Dengan Prinsip Sesuai Guidelines dan Best Practices
 - a. Menerapkan coding guidelines dan best practices dalam penulisan program (kode sumber)
 - b. Menggunakan ukuran performansi dalam menuliskan kode sumber

Result (R):

1. **Pilihan Bahasa Pemrograman dan Framework:** Bebas memilih bahasa pemrograman (misalnya, Python, Java, PHP) dan framework yang sesuai (misalnya, Django, Spring Boot, Laravel).
2. **Database:** Gunakan database relasional seperti MySQL, PostgreSQL, atau SQL Server.
3. **Fitur Aplikasi:** Anda dapat menambahkan fitur-fitur berikut:

Notifikasi: Mengirimkan notifikasi kepada mahasiswa terkait jadwal kuliah.

Laporan: Menghasilkan laporan data mahasiswa dan nilai.

Fitur lupa password: Buat fungsi untuk mereset password mahasiswa.

Fitur pencarian: Perbolehkan mahasiswa mencari mata kuliah berdasarkan nama atau dosen pengampu.

Perlengkapan dan Peralatan :

1. Personal Komputer atau Laptop
2. Aplikasi pengolah kata
3. Aplikasi pengolah diagram
4. Server web lokal
5. *Integrated Development Tools* (IDE)
6. Server basis data lokal

Waktu : 240 Menit

Kelompok Pekerjaan 3	No.	Kode Unit	Judul Unit
Melakukan pengujian perangkat lunak	1.	J.620100.033.02	Melaksanakan Pengujian Unit Program

Situation (S):

Anda telah berhasil mengembangkan sistem informasi akademik sederhana yang memungkinkan mahasiswa mendaftar, login, melihat jadwal kuliah, dan membayar UKT. Sekarang, Anda akan melakukan pengujian untuk memastikan aplikasi berjalan dengan baik.

Task (T)

- Anda ditugaskan untuk melaksanakan pekerjaan yang dimulai dari:

No	Unit Kompetensi
1	Melaksanakan Pengujian Unit Program

- Laksanakan seluruh pekerjaan tersebut sesuai prosedur (SOP)

Action (A):

Langkah kerja untuk setiap *Task* diuraikan sebagai berikut.

1. Melaksanakan Pengujian Unit Program
 - a. Menentukan kebutuhan uji coba dalam pengembangan
 - b. Mempersiapkan dokumentasi uji coba
 - c. Mempersiapkan data uji
 - d. Melaksanakan prosedur uji coba
 - e. Mengevaluasi hasil uji coba

Result (R):

1. Identifikasi Jenis Pengujian:
 - a. Sebutkan jenis-jenis pengujian yang relevan untuk aplikasi ini (misalnya, unit testing, integrasi testing, UAT).
 - b. Jelaskan tujuan dari masing-masing jenis pengujian tersebut.
2. Buat Kasus Uji:
 - a. Buatlah setidaknya 2 kasus uji untuk setiap fitur yang ada (misalnya, pendaftaran, login, melihat jadwal).
 - b. Pertimbangkan berbagai skenario, termasuk kasus normal, kasus batas, dan kasus error.

3. Tulis Kode Pengujian:
 - a. Gunakan framework pengujian yang sesuai (misalnya, unittest untuk Python, JUnit untuk Java, PHPUnit untuk PHP) untuk menulis kode pengujian.
 - b. Pastikan kode pengujian mencakup semua kasus uji yang telah dibuat.
4. Lakukan Pengujian:
 - a. Jalankan semua kasus uji dan amati hasilnya.
 - b. Catat setiap bug atau kesalahan yang ditemukan.
5. Buat Laporan Pengujian, susun laporan pengujian yang berisi:
 - a. Daftar kasus uji yang telah dijalankan.
 - b. Hasil dari setiap kasus uji.
 - c. Bug yang ditemukan dan langkah-langkah untuk memperbaikinya.
 - d. Kesimpulan keseluruhan tentang kualitas aplikasi.

Perlengkapan dan Peralatan :

1. Personal Komputer atau Laptop
2. Aplikasi pengolah kata
3. Aplikasi pengolah diagram
4. Server web lokal
5. Integrated Development Tools (IDE)
6. Server basis data lokal

Waktu : 120 Menit

ASESI :		
Nama	:	
Tanda tangan dan Tanggal	:	
ASESOR :		
Nama	:	
No. Reg	:	
Tanda tangan dan Tanggal	:	