

SOAL UJIAN AKHIR SEMESTER (UAS)

Pemrograman Web & Database – Kelas XI RPL

Setiap siswa akan mendapatkan **1 judul studi kasus** yang berbeda.
Kerjakan **database + aplikasi web** berdasarkan studi kasus yang diberikan.

INSTRUKSI UMUM

1. **Gunakan MySQL** sebagai database utama.
 2. Buat database dengan nama:
 3. `uas11rpl_<NAMA_KAMU>`
 4. Untuk setiap table, field, file coding dan variabel di coding, **wajib menambahkan nama kamu**:
Contoh:
 - o `barang_gani`
 - o `pengguna_rizki`
 5. Field wajib dirancang sendiri (bebas & bervariasi antar siswa) namun harus memenuhi kebutuhan studi kasus.
 6. Buat **Primary Key, Foreign Key, dan relasi** sesuai entitas yang diberikan.
 7. Aplikasi boleh dibuat menggunakan:
 - o **PHP Native**, atau
 - o **Python Flask**
 8. Fitur aplikasi minimal:
 - o **Login (validasi dari database)**
 - o **1 Master Data CRUD lengkap (Create, Read, Update, Delete)**
 - o **1 Data Transaksi (dengan relasi ke master)**
 - o **Halaman dashboard sederhana**
 - o **Logout**
 9. Diperbolehkan menggunakan Bootstrap
 10. Wajib menggunakan minimal **1 JOIN** pada salah satu fitur (umumnya pada transaksi).
-

5 JUDUL STUDI KASUS

Judul 1: Sistem Inventaris Peralatan Laboratorium Sekolah

Deskripsi

Laboratorium sekolah ingin mencatat seluruh peralatan, jenis peralatan, kondisi, serta riwayat peminjaman oleh guru atau siswa.

Entitas Wajib Dibuat

(semua tambahkan nama siswa, ex: alat_gani)

1. **pengguna_nama** – untuk login
2. **kategori_alat_nama** – master
3. **alat_nama** – master detail
4. **peminjaman_nama** – transaksi
5. **detail_peminjaman_nama** – transaksi detail

Transaksi Wajib

Peminjaman alat → memilih alat + peminjam + tanggal.

Judul 2: Sistem Reservasi Fasilitas Olahraga

Deskripsi

Siswa dapat memesan fasilitas olahraga (lapangan, ruang senam, ruang panahan). Petugas mencatat fasilitas, jadwal, dan reservasi.

Entitas Wajib

1. pengguna_nama
2. fasilitas_nama
3. jadwal_fasilitas_nama
4. reservasi_nama

Transaksi Wajib

Pemesanan fasilitas → memilih fasilitas & jadwal.

Judul 3: Sistem Pengelolaan Kursus (Les Privat)

Deskripsi

Sebuah lembaga kursus membutuhkan sistem untuk mengelola guru, mata pelajaran, siswa kursus, dan pendaftaran siswa ke kelas tertentu.

Entitas Wajib

1. pengguna_nama
2. guru_nama
3. mapel_nama
4. siswa_nama
5. pendaftaran_kelas_nama

Transaksi Wajib

Pendaftaran siswa → memilih siswa + mapel + guru.

Judul 4: Sistem Penjualan Produk Handmade

Deskripsi

UMKM sekolah menjual produk handmade (kerajinan tangan). Mereka butuh sistem untuk mencatat produk, kategori, serta transaksi penjualan.

Entitas Wajib

1. pengguna_nama
2. kategori_produk_nama
3. produk_nama
4. penjualan_nama
5. detail_penjualan_nama

Transaksi Wajib

Penjualan produk → memilih produk + qty + total (perhitungan aritmatika wajib).

Judul 5: Sistem Manajemen Perpustakaan Mini Kelas

Deskripsi

Kelas memiliki perpustakaan mini. Sistem diperlukan untuk mencatat buku, jenis buku, anggota kelas, dan peminjaman.

Entitas Wajib

1. pengguna_nama
2. kategori_buku_nama
3. buku_nama
4. anggota_kelas_nama
5. peminjaman_buku_nama

Transaksi Wajib

Peminjaman buku → anggota + buku + tanggal.

INSTRUKSI DETAIL DATABASE

Siswa harus:

1. Membuat tabel sesuai entitas.
2. Menentukan:
 - Primary Key
 - Foreign Key
 - Tipe data yang sesuai
 - Relasi antar tabel
3. Minimal **5 field** untuk setiap tabel.
4. Harus menggunakan **JOIN** pada salah satu fitur (ideal pada transaksi).

INSTRUKSI DETAIL APLIKASI

1. LOGIN

- Validasi username & password dari tabel pengguna_nama
- Password minimal dienkripsi MD5 / hashing Python

2. DASHBOARD

Tampilkan ringkas jumlah data dari setiap tabel master (gunakan COUNT).

3. MASTER DATA (CRUD)

Pilih 1 tabel master (misal produk, buku, fasilitas).

Fitur wajib:

- Tambah data
- Tampil data
- Update
- Delete

Gunakan query SELECT dengan pengurutan (ORDER BY).

4. TRANSAKSI

Pilih transaksi sesuai studi kasus.

Fitur wajib:

- Form transaksi
- Relasi ke master
- Menggunakan query JOIN
- Perhitungan wajib (misal total harga, total qty, durasi waktu)

5. LOGOUT

- Menghapus session / token login
- Redirect ke halaman login

6. DESAIN

- Gunakan CSS Framework Bootstrap atau custom
- Form harus rapi
- Tabel data jelas dan konsisten
- Navigasi mudah dipahami

RUBRIK PENILAIAN

No	Komponen Penilaian	0	1	2	3	4
1	Struktur Database (tabel & nama DB sesuai studi kasus)	Tidak ada DB/tabel penting	Banyak tabel hilang/salah	Tabel ada tetapi struktur kurang tepat	Struktur hampir benar, beberapa minor issue	Struktur lengkap, logis, sesuai studi kasus
2	Field & Tipe Data (kecocokan & variasi field)	Field tidak relevan atau sangat minim	Ada beberapa field, banyak tipe salah	Field cukup lengkap, beberapa tipe kurang tepat	Field lengkap & umumnya tepat	Field sangat lengkap, tepat, kreatif sesuai kebutuhan
3	Primary Key & Foreign Key	Tidak ada PK/FK	PK ada tapi FK banyak salah/tidak ada	PK & FK ada tapi konsistensi kurang	PK & FK tepat dengan sedikit perbaikan	PK & FK lengkap, konsisten, mengikuti best practice
4	Relasi Antar Tabel (ERD logis)	Tidak ada relasi	Relasi salah / tidak logis	Relasi ada namun kurang lengkap	Relasi baik & mendukung transaksi	Relasi sangat logis, lengkap, mendukung seluruh alur
5	Login (fungsi & validasi dasar)	Tidak berfungsi	Ada tetapi mudah dilewati / hardcode	Berfungsi tapi validasi minim	Berfungsi baik, validasi memadai	Berfungsi sangat baik, hashing atau proteksi dasar ada
6	Dashboard (ringkasan/count data)	Tidak ada	Ada tapi tidak menampilkan data	Tampil namun data kurang akurat	Tampil akurat & informatif	Tampil sangat informatif, layout ringkas & relevan
7	CRUD — Interface / Form & UI	Form/tampilan CRUD tidak ada	UI ada tapi sangat buruk / membingungkan	UI cukup jelas namun kurang rapi	UI rapi & mudah digunakan	UI sangat baik, form informatif & user friendly
8	CRUD — Functionality	CRUD tidak berjalan	Hanya 1–2 aksi berjalan	CRUD lengkap tapi	CRUD lengkap & stabil	CRUD lengkap,

	(Create/Read/Update/Delete)			sering error		stabil, edge-case ditangani
9	CRUD — Validasi & Integritas	Tidak ada validasi	Validasi minimal / mudah dilewati	Validasi ada namun tidak lengkap	Validasi cukup lengkap	Validasi lengkap (server & client), integritas terjaga
10	Transaksi — Form & Input UX	Tidak ada form transaksi	Form ada tapi tidak usable	Form usable namun input kurang lengkap	Form baik, alur jelas	Form sangat baik, validasi input jelas, user-friendly
11	Transaksi — Business Logic & Perhitungan (cek stok, hitung total, dll.)	Tidak ada logika / salah total	Logika minim / hitung sering keliru	Logika ada tapi tidak lengkap	Logika cukup baik & akurat	Logika lengkap, edge-case & perhitungan akurat
12	Transaksi — Penyimpanan, JOIN & Konsistensi Data	Tidak menyimpan / error	Menyimpan tapi tanpa JOIN / data inconsistent	Menyimpan & JOIN ada tapi kurang optimal	Menyimpan & JOIN benar, sebagian edge-case belum	Menyimpan sempurna, JOIN optimal, data konsisten & rollback/transaction handling jika perlu
13	Keamanan Dasar Aplikasi (session, sanitasi, proteksi halaman)	Tidak ada sama sekali	Proteksi sangat lemah	Proteksi dasar ada tapi banyak celah	Proteksi memadai pada banyak area	Proteksi kuat: session, sanitasi, CSRF minimal/pengamanan input aman
14	Desain UI/UX (keterbacaan, konsistensi warna/typography)	Sangat berantakan, sulit dibaca	Kurang konsisten & kurang rapi	Cukup rapi tapi biasa saja	Rapi, konsisten, nyaman dipakai	Sangat rapi, harmonis, konsisten & mudah digunakan
15	Kerapihan Kode & Struktur Folder	Kode acak, tidak ada struktur	Struktur folder membingungkan	Cukup rapi tapi tidak konsisten	Rapi & mudah diikuti	Sangat rapi, dokumentasi minimal, standar penamaan baik