

CSGE602070 – Basis Data
TA 2024/2025 – Semester Genap



Tugas Kelompok - TK04

Implementasi Aplikasi & Deployment Sistem Informasi Zoo Pintar (SIZOPI)

Case 2

Penulis : Tim Dosen & Tim Asisten Dosen
Versi : 1 (20250429-0800)



FAKULTAS
ILMU
KOMPUTER










© 2025, Fakultas Ilmu Komputer Universitas Indonesia

 Riwayat Versi

Setiap “**Versi**” yang dimaksudkan pada riwayat ini dan dijadikan rujukan utama bagi dokumen ini memuat perubahan yang bersifat substantif sehingga perlu diketahui oleh pemangku kepentingan dokumen ini. Riwayat versi ini diurutkan secara kronologis terurut dari versi paling akhir pada baris pertama hingga versi paling awal pada baris terakhir.

Versi	Timestamp	Halaman	Perubahan
1	20250429-0800	Semua	Rilis Pertama

Daftar Isi

 Riwayat Versi	2
 Daftar Isi	3
 Informasi Umum	4
 Ekspektasi Hasil Pembelajaran	5
 Petunjuk Pengerjaan	5
 Deskripsi	9
DAFTAR FITUR	9
DESKRIPSI TABEL	9
DATA DUMMY	9
TRIGGER & STORED PROCEDURE	9
SKENARIO & DESKRIPSI FITUR	13
 Spesifikasi	14
 Informasi Terkait Pengumpulan Berkas	14
 Peraturan	14
Keterlambatan	14
Plagiarisme	15

Tugas Kelompok – TK03 Sistem Informasi Zoo Pintar (SIZOPI)



Regenerate image

Prompt ChatGPT: buatlah gambar ilustrasi SIZOPI (Sistem Informasi Zoo Pintar) yang menarik dengan sedikit text

Oleh: Ilma Alpha Mannix, Syifa Nurhayati, Michelle Angelica Santoso,
Wahyu Ridho Anggoro, Abdul Zacky, Yosia Joseph Chandra, Lucinda Laurent,
Tara Mazaya Lababan, Judah Ariesaka Magaini



Informasi Umum

- Tipe Tugas : Kelompok
 - Batas Waktu Pengumpulan : Selasa, 29 Mei 2025 Jam 23.59 Waktu SCeLE
 - Format Penamaan Berkas : TK04_User Manual_[Kelas]_[Kode_Kelompok].pdf
TK04_Informasi Aplikasi_[Kelas]_[Kode_Kelompok].pdf
TK04_Trigger_[Kelas]_[Kode_Kelompok].zip
*isi zip trigger telah dijelaskan di bagian **petunjuk pengerjaan**.
 - List Pembagian Kelompok : [Klik di Sini](#)
 - Template User Manual : [Klik di Sini](#)
 - Template Informasi Aplikasi : [Klik di Sini](#)
- *skema relasional yang digunakan sebagai kunci jawaban TK02, pada TK03-TK04 ini terdapat penyesuaian dan penambahan atribut pada beberapa entitas.



Ekspektasi Hasil Pembelajaran

CPL	Mampu menerapkan penalaran yang kritis, sistematis, dan logis dalam menganalisa dan memformulasikan masalah serta mengikuti kaidah ilmiah untuk memperoleh solusinya secara komputasional
CPMK	Mahasiswa dapat merancang basis data secara benar dan sesuai dengan kebutuhan data pengguna.
Sub-CPMK	1. Mahasiswa dapat memodelkan kebutuhan data menggunakan (<i>enhanced</i>) <i>entity relationship diagram</i> . 2. Mahasiswa dapat mengevaluasi hasil rancangan basis data.



Petunjuk Pengerjaan

- Pada TK04 ini, kalian diminta untuk mengimplementasikan *backend* dari fitur-fitur yang terdapat pada section **Daftar Fitur** dan integrasikan dengan frontend yang sudah dibuat pada TK03.
 - Tampilan maupun flow fitur harus disesuaikan dengan penjelasan pada section **Skenario & Deskripsi Fitur**.
 - Estetika dan kerapian halaman dari tampilan tidak dinilai, hanya kesesuaian tampilan dengan deskripsi skenario saja yang dinilai.
 - Perhatikan kembali terkait revisi yang ada pada TK03 sebelumnya. Pada TK04 ini harus sudah diimplementasikan. Semua deskripsi selain “**Trigger & Stored Procedures**” dapat dilihat melalui Dokumen [Dokumen TK03 - Sizopi](#). Fitur yang dibuat harus sudah terimplementasikan dengan baik pada TK04 ini.
- Tugas Kelompok 4 merupakan tahap final, di mana aplikasi web perlu di-**deploy ke server** (bebas) dan terkoneksi dengan *database online* (eg. [Supabase](#), [Neon](#), [Railway](#), dll). Mahasiswa dapat menggunakan database gratis, seperti add-ons PostgreSQL pada Supabase, Railway atau Neon.
 - Buat tabel pada PostgreSQL di server dengan acuan tabel yang sudah tersedia dan isi tabel menggunakan data dummy dengan minimum jumlah data pada tabel.
 - Implementasikan **trigger dan stored procedures** yang terdapat di section “**Trigger & Stored Procedures**” pada PostgreSQL di server.
 - Tidak diperbolehkan menggunakan library ORM** seperti Django models, Spring JPA, ataupun TypeORM NodeJS. Mahasiswa harus menuliskan secara langsung query SQL pada kode aplikasi. Query dengan menuliskan SQL dapat dilakukan menggunakan library eksternal, seperti [psycopg](#) untuk Python atau [pg](#) untuk NodeJs.

3. Setiap fitur yang dikerjakan oleh masing-masing anggota kelompok disatukan dalam satu repository gitlab dengan **visibility private** dan **jadikan asdos pemeriksa kelompok sebagai maintainer dalam repository github/gitlab**.

Asisten Dosen akan memeriksa source code kalian. Pastikan kode dapat dijalankan secara lokal tanpa error, dan aplikasi dapat diakses melalui URL yang telah kalian deploy!

4. Manfaatkan waktu sebaik mungkin dan berikan waktu untuk men-*deploy* aplikasi agar tidak ada kendala saat dijalankan, karena **pada TK04 ini akan terdapat demo aplikasi dengan asisten dosen**.

5. Tugas yang dikumpulkan terdiri dari:

a. Dokumen dengan format **file PDF**:

i. **1 Dokumen yang berisi user manual** untuk menggunakan keseluruhan fitur-fitur yang ada pada sistem sesuai dengan skenarionya, yang **dilengkapi** dengan **URL Halaman**, **Tangkapan Layar**, dan **Penjelasan Fitur**. User manual harus dibuat lengkap dan jelas karena akan digunakan sebagai acuan UAT (*User Acceptance Testing*) dalam penilaian TK04.

File pdf dikumpulkan dengan format penamaan:

TK04_User Manual_[Kelas]_[Kode_Kelompok].pdf

ii. **1 Dokumen yang berisi:**

- Informasi URL aplikasi di server (termasuk dengan username dan password database yang digunakan)
- Akun dummy dari setiap pengguna (jumlahnya bebas)
- Link repository (Github/Gitlab)

File pdf dikumpulkan dengan format penamaan:

TK04_Informasi Aplikasi_[Kelas]_[Kode_Kelompok].pdf

b. **1 Folder** yang berisi file-file terkait trigger dan stored procedure yang dikerjakan dalam **format file sql**, dengan format penamaan file:

TK04_Trigger_[Kelas]_[Kode_Kelompok]_[Nomor Trigger].sql

setelah dijadikan folder zip file dengan format penamaan:

TK04_Trigger_[Kelas]_[Kode_Kelompok].zip

Struktur Folder sebelum di zip, Nomor Trigger diambil dari table warna ini.

```

TK04_Trigger_[Kelas]_[Kode_Kelompok]/
├── TK04_Trigger_[Nomor Trigger]_[Kelas]_[Kode_Kelompok].sql
├── TK04_Trigger_[Nomor Trigger]_[Kelas]_[Kode_Kelompok].sql
├── ....
└── TK04_Trigger_[Nomor Trigger]_[Kelas]_[Kode_Kelompok].sql

```

6. *Branch* untuk penilaian TK4 ini adalah *branch main/master*. **Penalti** akan dikenakan jika melakukan *commit-push* lebih dari jam *deadline* dengan mengacu keterlambatan tugas sesuai pada aturan di BRP.

7. Tugas Kelompok 3 dikumpulkan hanya dalam bentuk softcopy berformat yang sesuai yang telah disebutkan diatas. **Tidak perlu di ZIP**.

1 Kelompok cukup yang mengumpulkan hanya anggota saja di SCell!

8. Tuliskan **Kode Kelompok** dan **anggota-anggota dari kelompoknya** yang terdiri dari **Nama** dan **NPM** di halaman pertama lembar berkas jawaban anda. Kelalaian menuliskan informasi ini (lupa atau salah tulis) akan diberi **penalti 5 poin**.

9. Tulis/ketik ulang pernyataan **Kode Etik Akademik** di bawah ini pada bagian terakhir Laporan Tugas Kelompok Anda. Berkas Tugas Kelompok tanpa pernyataan seperti di bawah ini **tidak akan dinilai**.

Kode Etik Akademik:

Code of Conduct :

Kami menyatakan bahwa Tugas Kelompok ini kami kerjakan dengan usaha kami sendiri. Kami tidak menyalin jawaban dari sumber manapun. Kami

bertanggung jawab menjaga agar jawaban Tugas Kelompok kami tidak disalin oleh peserta kelompok lainnya.

Proporsi pengerjaan Tugas Kelompok adalah sbb:

No	NPM	Nama Anggota	Deskripsi Tugas	Kontribusi (0-5)				
				1	2	3	4	5
1	ex: 1234567890	ex: Budi	ex: Membuat entitas A, B, C dan atributnya (5 entity)		4	4	4	5
2	ex: 1234567891	ex: Ani	ex: Membuat entitas A, B, C dan atributnya (4 entity)	4		5	5	4
3	ex: 1234567890	ex: Dede	ex: Membuat entitas A, B, C dan atributnya (4 entity)	5	5		5	4
4	ex: 1234567890	ex: Joko	ex: Membuat entitas A, B, C dan atributnya (6 entity)	4	4	5		5
5	ex: 1234567890	ex: Edi	ex: Membuat entitas A, B, C dan atributnya (6 entity)	4	4	5	5	

*** Cara penilaian kontribusi adalah sebagai berikut:**

- Penilaian berkisar antara nilai 0 hingga 5. Nilai 0 jika anggota kelompok tersebut tidak berpartisipasi sama sekali.
- Angka 1-4 yang berwarna **biru** menunjukkan nomor tiap anggota (mengacu pada kolom No). Jika jumlah anggota kelompok adalah 3 orang, maka pada kolom **Kontribusi 4** tidak perlu diisi.
- Penilaian diberikan untuk anggota kelompok lainnya. Misal mahasiswa nomor 1 memberikan nilai untuk mahasiswa nomor 2, 3, dan 4 dengan meletakkan penilaiannya di kolom **1**. Begitupun untuk mahasiswa lainnya, harus meletakkan penilaiannya di kolom yang sesuai dengan nomornya masing-masing.

10. Jika ada keterangan soal yang dianggap ambigu atau belum dijelaskan, dipersilahkan mahasiswa **menuliskan asumsi** dan **implikasinya** di dalam file tugas yang dikumpulkan namun masih tetap sesuai dengan konteks studi kasus.

11. Jika ada pertanyaan, silakan bertanya melalui **forum Discord** atau melalui **ticket Discord** jika pertanyaan bersifat pribadi. Harap **tidak mengirim pesan langsung kepada asisten melalui direct message**.

12. Penalti keterlambatan dijelaskan pada dokumen soal mengacu pada aturan di BRP.

Deskripsi

DAFTAR FITUR

* Semua fitur harus terimplementasi. Perhatikan kembali pada revisi TK3 lalu, silakan lihat kembali pada [Dokumen TK03 - Sizopi](#).

DESKRIPSI TABEL

* Semua fitur harus terimplementasi. Perhatikan kembali pada revisi TK3 lalu, silakan lihat kembali pada [Dokumen TK03 - Sizopi](#).

DATA DUMMY

* Semua fitur harus terimplementasi. Perhatikan kembali pada revisi TK3 lalu, silakan lihat kembali pada [Dokumen TK03 - Sizopi](#).

TRIGGER & STORED PROCEDURE

NOTE (WAJIB DIBACA)!

- Kerjakan trigger sesuai dengan **kelompok warna fitur yang Anda pilih** (misal: Anda memilih mengerjakan kelompok fitur berwarna hijau, maka kerjakan trigger berwarna hijau).
- Apabila tim beranggotakan **3 orang**, kelompok fitur warna **Merah tidak dikerjakan**
- Kelompok fitur warna putih merupakan fitur wajib kelompok

Jangan lupa tampilkan pesan ke halaman *web* dengan memunculkan pesan error yang didapat dari hasil *Trigger & Stored Procedure* dibawah ini!

Return pesan **WAJIB** dari *Trigger & Stored Procedure*, bukan *logic backend*.

No	Nama dan Deskripsi	Keterangan
1	<p>1. Pemeriksaan Duplikasi Akun saat Registrasi</p> <p>Ketika pengguna melakukan registrasi akun baru, sistem secara otomatis akan memeriksa apakah akun dengan username yang sama sudah terdaftar sebelumnya. Jika sudah ada, sistem akan membatalkan proses pendaftaran.</p> <p>Pesan yang ditampilkan: "ERROR: Username "<username>" sudah digunakan, silakan pilih username lain."</p> <p>Contoh output error: ERROR: Username "akunku" sudah digunakan, silakan pilih username lain.</p> <p>2. Verifikasi Kredensial saat Login</p> <p>Ketika pengguna melakukan login, sistem akan memverifikasi apakah kombinasi username dan password yang dimasukkan sesuai dengan data yang tersimpan di database. Jika kredensial tidak cocok, sistem akan menolak akses.</p> <p>Pesan yang ditampilkan: "Username atau password salah, silakan coba lagi."</p>	Wajib
2	<p>1. Pemeriksaan Satwa Duplikat saat Pendaftaran</p> <p>Ketika pengguna menambahkan data satwa baru, sistem secara otomatis akan memeriksa apakah satwa dengan kombinasi nama_individu, spesies, dan asal_hewan sudah pernah terdaftar sebelumnya. Jika sudah ada, sistem akan membatalkan proses penyimpanan.</p> <p>Pesan yang ditampilkan: "Data satwa atas nama "<nama>", spesies "<spesies>", dan berasal dari "<asal_hewan>" sudah terdaftar."</p> <p>Contoh output: Data satwa atas nama "Moka", spesies "Anggora", dan berasal dari "Jakarta" sudah terdaftar."</p> <p>2. Pencatatan Riwayat Perubahan Data Satwa</p> <p>Ketika pengguna melakukan update data satwa dan mengganti status_kesehatan atau nama_habitat, sistem akan mencatat perubahan tersebut ke tabel RIWAYAT_SATWA. Tabel ini berfungsi untuk menyimpan log perubahan penting seperti riwayat habitat dan status medis hewan.</p>	Dikerjakan oleh 1 orang

	<p>Pesan yang ditampilkan: “SUKSES: Riwayat perubahan status kesehatan dari “<before>” menjadi “<after>” atau habitat dari “<before>” menjadi “<after>” telah dicatat.”</p> <p>Contoh output: SUKSES: Riwayat perubahan status kesehatan dari “sehat” menjadi “sakit” atau habitat dari “air” menjadi “air” telah dicatat.</p> <p><i>* dapat dilihat walau tidak ada update pesan “air” menjadi “air” tetap di munculkan.</i></p>	
3	<p>1. Sinkronisasi Rekam Medis dan Penjadwalan Pemeriksaan Kesehatan Setelah memasukkan data rekam medis baru, jika status kesehatan hewan yang baru dimasukkan adalah "Sakit", sistem otomatis mengubah tanggal pemeriksaan selanjutnya (jika ada) yang sama atau paling dekat setelah tanggal pemeriksaan rekam medis yang baru dimasukkan untuk hewan tersebut menjadi 7 hari dari tanggal pemeriksaan rekam medis yang baru dimasukkan. Jika tidak ditemukan tanggal pemeriksaan selanjutnya yang memenuhi syarat, buat jadwal pemeriksaan baru dengan aturan tanggal yang sama. Pesan yang ditampilkan: “SUKSES: Jadwal pemeriksaan hewan "<nama_hewan>" telah diperbarui karena status kesehatan "Sakit".”</p> <p>Contoh output: SUKSES: Jadwal pemeriksaan hewan "Simba" telah diperbarui karena status kesehatan "Sakit".</p> <p>2. Penambahan Jadwal Pemeriksaan sesuai Frekuensi Ketika dibuat jadwal pemeriksaan baru, perlu ditambahkan juga data jadwal pemeriksaan lainnya secara otomatis dalam tahun yang sama, yang dimana penambahan jadwal otomatis tersebut disesuaikan dengan frekuensi pemeriksaan rutin hewan. Pesan yang ditampilkan: “SUKSES: Jadwal pemeriksaan rutin hewan "<nama_hewan>" telah ditambahkan sesuai frekuensi.”</p> <p>Contoh output: SUKSES: Jadwal pemeriksaan rutin hewan "Molly" telah ditambahkan sesuai frekuensi.</p>	Dikerjakan oleh 1 orang
4	<p>1. Pemeriksaan Kapasitas Atraksi saat Reservasi Ketika pengunjung melakukan reservasi tiket untuk suatu atraksi, sistem secara otomatis akan memeriksa apakah jumlah tiket yang diminta tidak melebihi kapasitas tersisa untuk atraksi tersebut pada tanggal yang dipilih. Pesan yang ditampilkan: “ERROR: Kapasitas tersisa “<sisa_tiket>” tiket, atraksi tidak mencukupi untuk sejumlah “<permintaan_tiket>” tiket yang diminta.”</p> <p>Contoh output: ERROR: Kapasitas tersisa “5” tiket, atraksi tidak mencukupi untuk sejumlah “10” tiket yang diminta.</p> <p>2. Rotasi Otomatis Pelatih Hewan untuk Atraksi Ketika admin melakukan update data atraksi yang melibatkan hewan dan pelatih melalui relasi antara tabel ATRAKSI, HEWAN, dan PELATIH_HEWAN, sistem akan memeriksa apakah</p>	Dikerjakan oleh 1 orang

	<p>pelatih tersebut sudah bertugas di atraksi yang sama selama 3 bulan atau lebih berdasarkan data di tabel JADWAL_PENUGASAN.</p> <p>Pesan yang ditampilkan: “SUKSES: Pelatih "<nama_pelatih>" telah bertugas lebih dari 3 bulan di atraksi "<nama_atraksi>" dan akan diganti.”</p> <p>Contoh output: SUKSES: Pelatih "Toto" telah bertugas lebih dari 3 bulan di atraksi "Atraksi Singa" dan akan diganti.</p>	
5	<p>1. Sinkronisasi Total Kontribusi Adopter dengan Data Adopsi Total kontribusi <i>adopter</i> merupakan jumlahan dari nominal kontribusi setiap adopsi berstatus pembayaran lunas yang dilakukan seorang <i>adopter</i>. Dengan begitu, sistem memerlukan mekanisme <i>update</i> total kontribusi secara otomatis agar nilainya selalu sesuai dengan data adopsi yang disimpan oleh sistem.</p> <p>Pesan yang ditampilkan: “SUKSES: Total kontribusi adopter "<nama_adopter>" telah diperbarui.”</p> <p>Contoh output: SUKSES: Total kontribusi adopter "Taro" telah diperbarui.</p> <p>2. Pemeringkatan Adopter dengan Total Kontribusi Tertinggi selama Setahun Terakhir Pada halaman daftar <i>adopter</i> untuk <i>role</i> staf administrasi, sistem perlu menampilkan top 5 <i>adopter</i> dengan total kontribusi tertinggi. Untuk itu, diperlukan mekanisme pemeringkatan <i>adopter</i> berdasarkan jumlah nominal kontribusi mereka pada adopsi yang dimulai dalam rentang waktu satu tahun terakhir.</p> <p>Pesan yang ditampilkan: “SUKSES: Daftar Top 5 Adopter satu tahun terakhir berhasil diperbarui, dengan peringkat pertama dengan nama adopter "<nama_adopter>" berkontribusi sebesar "<nominal_kontribusi>.”</p> <p>Contoh output: SUKSES: Daftar Top 5 Adopter satu tahun terakhir berhasil diperbarui, dengan peringkat pertama dengan nama adopter "Saphira" berkontribusi sebesar "RP10.000.000".</p>	Dikerjakan oleh 1 orang

SKENARIO & DESKRIPSI FITUR

* Semua fitur harus terimplementasi. Perhatikan kembali pada revisi TK3 lalu, silakan lihat kembali pada [Dokumen TK03 - Sizopi](#).

Spesifikasi

Kerjakan sesuai dengan deskripsi di atas dari studi kasus **Sistem Informasi Zoo Pintar (SIZOPI)**!

Informasi Terkait Pengumpulan Berkas

Pada tugas ini, Anda hanya perlu mengerjakan dan mengumpulkan beberapa bagian file ke submisi SCeLE secara langsung. Telah kami sediakan **Kerangka Laporan Tugas** sebagai standar minimum isi laporan. Pastikan untuk menggunakannya agar tidak ada bagian penting yang terlewat.

Anda harus mengumpulkan berkas-berkas berikut:

1. Dokumen User Manual dengan format **file PDF**:

TK04_User Manual_[Kelas]_[Kode_Kelompok].pdf

2. Dokumen Informasi Aplikasi dengan format **file PDF**:

TK04_Informasi Aplikasi_[Kelas]_[Kode_Kelompok].pdf

3. File **Trigger** dari setiap nomor dijadikan satu file ZIP, dengan format penamaan file:

TK04_Trigger_[Kelas]_[Kode_Kelompok].zip

*isi zip trigger telah dijelaskan di bagian **petunjuk pengerjaan**.

Anda hanya perlu mengumpulkan sesuai format dan penamaan berikut ke slot submisi SCell.

Peraturan

Keterlambatan

Anda diharapkan dapat mengumpulkan hasil pekerjaan yang dilakukan sebelum batas waktu pengumpulan. Jika terdapat kondisi di mana Anda terpaksa terlambat mengumpulkan hasil pekerjaan, terdapat jangka waktu tambahan di mana Anda masih diperbolehkan mengumpulkan hasil pekerjaan dengan konsekuensi tertentu. Jika X adalah durasi setelah batas waktu pengumpulan yang ditetapkan sampai waktu Anda mengumpulkan hasil pekerjaan, Anda akan menerima penalti nilai pekerjaan sebagaimana diatur pada peraturan berikut ini:

- **X < 30 menit** : Penalti 10 poin dari nilai
- **30 menit ≤ X < 24 jam** : 50% penalti
- **X ≥ 24 jam** : Cut-off (Pekerjaan anda tidak akan diterima)

Plagiarisme

Anda diperbolehkan berdiskusi tentang pekerjaan Anda dengan peserta kuliah lain atau pihak lainnya, namun Anda harus memastikan bahwa **semua pekerjaan yang dikumpulkan adalah murni hasil pekerjaan Anda sendiri**. Anda dilarang keras melakukan tindak plagiarisme atau kecurangan akademik lainnya. Menurut kamus daring Merriam-Webster, plagiarisme berarti:

- Mencuri dan mengklaim (ide atau kata orang lain) sebagai milik sendiri
- Menggunakan hasil (karya/pekerjaan orang lain) sebagai milik sendiri
- Melakukan pencurian literatur/sastra
- Merepresentasikan ulang sebuah ide/produk yang sudah ada sebagai sesuatu yang bersifat baru dan orisinal.

Tim pengajar memiliki hak untuk meminta klarifikasi terkait dugaan ketidakjujuran akademik, terutama plagiarisme, dan memberikan konsekuensi berupa **pengurangan nilai hasil pekerjaan atau pencabutan nilai (nilai diubah menjadi nol) untuk hasil pekerjaan yang terkonfirmasi dikerjakan secara tidak jujur**.

of