

SOAL PRAKTIK UKK RPL

A. Petunjuk

1. Baca dan pelajari setiap instruksi kerja di bawah ini dengan cermat sebelum melaksanakan praktek
2. Klarifikasi kepada penguji 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

Sebagai seorang junior coder Anda diminta untuk mengembangkan sebuah aplikasi “todolist” berbasis web sesuai dengan spesifikasi yang dibutuhkan.

Fitur-fitur minimal yang disediakan antara lain :

1. Pengguna baru bisa mendaftar (sign up).
2. Setelah memiliki akun, pengguna bisa melakukan login ke dalam sistem dengan username dan password yang telah dimiliki.
3. Pengguna bisa menampilkan, membuat, mengubah, dan menghapus (crud) “todolist” satu per satu.
4. Pengguna bisa menentukan kategori “todolist”-nya menjadi “work”, “personal”, dan “other”.
5. Pengguna bisa memfilter “todolist” yang telah dibuat berdasarkan kategorinya.
6. Pengguna bisa meng-update status “todolist” yang sudah dibuat menjadi “pending” atau “done”.
7. “todolist” yang sudah dikerjakan masih tampil di layar, hanya saja menggunakan warna yang berbeda dari “todolist” lain.
8. Pengguna bisa logout dari sistem setelah selesai membuat “todolist”.

Untuk mendukung pencapaian hasil sesuai dengan spesifikasi yang telah ditentukan tersebut, Anda akan diperlengkapi dengan komputer, software pemrograman web, dan SOP/IK terkait.

Perlengkapan dan Peralatan :

Laptop / PC

Waktu : 7 jam (420 menit)

No	Langkah Kerja	Instruksi Kerja
1.	Mengecek semua perangkat hardware dan software	Periksa perangkat hardware dan software sesuai kebutuhan pemrograman.
2.	Menginstal dan mengkonfigurasi software utama dan pendukung sesuai kebutuhan	Pilih tools bahasa pemrograman sesuai dengan kebutuhan dan lingkungan pengembangan. Jika diperlukan, lakukan instalasi tools pemrograman sesuai dengan prosedur yang ada. Jalankan tools pemrograman di lingkungan pengembangan yang telah ditetapkan. Lakukan pengecekan terhadap hasil instalasi software utama dan pendukung.
3.	Membuat database	Buatlah database dengan nama “db_todolist_app”
4.	Membuat tabel dan relasi	Buatlah tabel-tabel di dalam database tersebut berdasarkan struktur ERD berikut ini :



5.	Mengisi tabel	Isi data awal ke dalam tabel dengan menggunakan perintah DML
6.	Membuat proyek aplikasi	Buatlah proyek baru dengan nama “My todolist”

7.	Membuat koneksi ke database	Buatlah script untuk melakukan koneksi dari proyek “app_todolist” ke database “db_todolist” dengan nama “koneksi.php”
8.	Melakukan coding aplikasi sesuai guidelines	<p>Implementasikan aplikasi “todolist” berdasarkan spesifikasi desain database yang telah disediakan pada file-file berikut ini :</p> <ul style="list-style-type: none"> - daftar - proses daftar - login - proses login - logout - index - tambah - proses tambah - edit - proses edit - hapus - Profil <p>Pastikan hanya pengguna yang sudah login yang bisa masuk ke dalam halaman utama (index).</p> <p>Buat beberapa macam filir untuk filtering :</p> <ul style="list-style-type: none"> ➤ Tambahkan fitur “filter kategori” di halaman utama untuk menampilkan “todolist” berdasarkan kategorinya (work, personal, dsb). ➤ Tambahkan fitur “filter status” di halaman utama untuk menampilkan “todolist” berdasarkan statusnya (done, pending) ➤ Tambahkan fitur “filter bookmark” di halaman utama untuk menampilkan “todolist” berdasarkan favorite (tambahkan boolean isFavorite) <p>Lakukan perubahan UI/UX dengan menambahkan CSS dan JavaScript agar aplikasi semakin mudah digunakan pengguna.</p>
9.	Melakukan debugging	Lakukan proses debugging setiap kali menambahkan fungsi program baru.
10.	Melakukan dokumentasi kode program	Tulislah dokumentasi program pada file, variabel, dan fungsi-fungsi penting di dalam kode program
11.	Melakukan pengujian unit program	Lakukan pengujian aplikasi secara menyeluruh pada setiap unit/fitur yang telah dikembangkan.

LAMPIRAN GAMBAR

Todolist

Login

Username

Password

Pengguna baru? [Sign Up](#)

Todolist

Daftar Pengguna Baru

Nama Lengkap

Email

Username

Password

Tanggal Lahir
 mm/dd/yyyy

Sudah punya akun? [Login di sini](#)

TodoList

Profile Logout

Filter Kategori: Semua

[+] Tambah

laravel
learn-laravel-11-tonight
Kategori: work
Status: Done

belajar java
belajar java karena besok ulangan
Kategori: work
Status: Pending

belajar ruby
belajar ruby sebagai modal untuk maju ke rails
Kategori: other
Status: Done

rust
beli buku rust
Kategori: other
Status: Pending

Edit Hapus

Edit Hapus

Edit Hapus

TodoList

Profile Logout

Filter Status: Done

[+] Tambah

laravel
learn-laravel-11-tonight
Kategori: work
Status: Done

belajar ruby
belajar ruby sebagai modal untuk maju ke rails
Kategori: other
Status: Done

Edit Hapus

Edit Hapus

TodoList

Profil Logout

Filter Bookmark: Favorite

[+] Tambah

belajar java
belajar java karena besok ulangan
Kategori: work
Status: Pending

Edit Hapus

This screenshot shows a single task card on the TodoList application. The task is titled "belajar java" with the subtitle "belajar java karena besok ulangan". It is categorized as "work" and has a status of "Pending". There are three buttons at the bottom of the card: "Edit" (blue), "Hapus" (red), and a red heart icon.

TodoList

Profil Logout

Tambah Tugas

Judul

Deskripsi

Kategori

work

Status

Pending

Tambah

This screenshot shows the "Tambah Tugas" (Add Task) form. It contains fields for "Judul" (Title), "Deskripsi" (Description), "Kategori" (Category), and "Status". The "Kategori" dropdown is set to "work" and the "Status" dropdown is set to "Pending". At the bottom is a large green "Tambah" (Add) button.