

FR.MUK.04. TUGAS PRAKTIK DEMONSTRASI

Nomor Skema	:	S-2019-016
Skema Sertifikasi	:	Pengembang Web Pratama (Junior Web Developer)
Nama asesor:	:	

A. Petunjuk

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

B. Skenario

Anda sebagai seorang pengembang web mendapatkan tugas untuk meneruskan pekerjaan dari programmer lain. Pekerjaan untuk membuat halaman web tersebut sudah dikerjakan sebagian oleh programmer sebelumnya, namun masih ada beberapa bagian yang tidak sesuai dengan kebutuhan pengguna yang tergambar dalam rancangan tampilan (*mockup*). Tugas Anda adalah memperbaiki dan menyempurnakan halaman web tersebut agar sesuai dengan kebutuhan pengguna/rancangan.

Halaman web tersebut dimaksudkan untuk memasukkan dan menampilkan data siswa yang terdiri dari nama, nomor induk, jenis kelamin, domisili, dan nilai-nilai. Silakan mengikuti petunjuk-petunjuk di bawah ini untuk menyelesaikan pekerjaan Anda.

- Rancangan tampilan antarmuka (*mockup*) terlampir di bawah.
- Anda diberikan sebuah *file* (terlampir dalam *file* zip, nama file: "*index.php*") berisi kode pemrograman untuk diselesaikan/disempurnakan.

Kondisi	Petunjuk
Belum tersedia <i>input dropdown</i> untuk memasukkan data domisili.	<ul style="list-style-type: none">▪ Buatlah sebuah <i>array</i> yang berisi data domisili (sesuai urutan): Jakarta, Depok, Bogor, Tangerang, Bekasi. (TMS 3.1, 5.1, 5.4, TRS 4.1)▪ Urutkan isi <i>array</i> tersebut (dengan menggunakan fungsi bawaan dari PHP). (TS 5.4, TRS 4.1)▪ Buatlah <i>input dropdown</i> dengan isi pilihan dari dalam <i>array</i>. Gunakan perulangan. (CMS 1.2 TMS 2.3, 5.1, 5.2, 5.4, TRS 4.1)

Tampilan data jenis kelamin masih belum sesuai dengan rancangan.	Sesuaikan tampilan data jenis kelamin sesuai dengan rancangan. Gunakan percabangan. (CMS 1.2, 5.2, TMS 2.3, TRS 4.1)
Tampilan data total nilai belum terlihat.	<ul style="list-style-type: none"> ▪ Buatlah fungsi untuk menghitung total nilai. Tuliskan baris komentar yang menerangkan fungsi tersebut. (TRS 3.1, 4.1, TMS 5.1, 5.3) ▪ Tambahkan baris kode untuk menampilkan total nilai. (CMS 1.2, 5.3, TMS 2.3, TRS 4.1)
Sudah tersedia <i>library file</i> CSS namun belum terhubung dengan <i>file index.php</i> .	Tambahkan baris kode untuk terhubung dengan <i>file</i> CSS. (TRS 4.1, CMS 6.2)
Belum tersedia file <i>readme</i> .	Buatlah file <i>readme</i> . (TRS 3.2, JRES 3.2)
Susunan file dan folder pekerjaan belum rapi.	Rapikan file dan folder pekerjaan sesuai dengan <i>best practice</i> . (CMS 3.2)

Catatan:

Data pada web tersebut disimpan dalam bentuk *array* JSON pada *folder/file data/data.json*. Baris kode untuk membaca dan menulis serta mengkonversi (*encode* dan *decode*) data dalam bentuk *array* JSON ke array PHP ataupun sebaliknya sudah tersedia. Anda **tidak perlu mengubah** *folder/file data/data.json* ataupun mengubah baris kode tersebut.

Lakukan pekerjaan diatas sesuai langkah kerja dibawah ini dan terapkan aturan K3.

LAMPIRAN: Mockup/Rancangan Tampilan Antarmuka

Formulir Data Siswa

Nama:

Nomor Induk:

Jenis Kelamin:

O Laki-laki

O Perempuan

Domisili

Jakarta

Nilai Matematika

Nilai Bahasa Indonesia

Nilai Bahasa Inggris

Kirim

Nama	Nomor Induk	Jenis Kelamin	Domisili	Nilai Matematika	Nilai Bahasa Indonesia	Nilai Bahasa Inggris	Total Nilai
Ade	123	Perempuan	Jakarta	9	7	8	24
Bagus	456	Laki-laki	Depok	7	8	8	23
Candra	789	Laki-laki	Bekasi	6	9	8	23

Langkah kerja : (Elemen)

1. Mengimplementasikan User Interface
 - 1.1 Mengidentifikasi rancangan user interface
 - 1.2 Melakukan implementasi rancangan user interface
2. Menerapkan perintah eksekusi bahasa pemrograman berbasis text, grafik, dan multimedia
 - 2.1 Mengidentifikasi mekanisme running atau eksekusi source code
 - 2.2 Mengeksekusi source code
 - 2.3 Mengidentifikasi hasil eksekusi
3. Menyusun Fungsi, File atau Sumber Daya Pemrograman yang Lain dalam Organisasi yang Rapi
 - 3.1 Mengelola sumber daya pemrograman sesuai karakter
 - 3.2 Mengorganisasikan sumber daya pemrograman sesuai konteks
4. Menulis Kode dengan Prinsip Sesuai Guidelines dan Best Practices
 - 4.1 Menerapkan coding guidelines dan best practices dalam penulisan program (kode sumber)
 - 4.2 Menggunakan ukuran performansi dalam menuliskan kode sumber
5. Mengimplementasikan Pemrograman Terstruktur
 - 5.1 Menggunakan tipe data dan control program
 - 5.2 Membuat program sederhana
 - 5.3 Membuat program menggunakan prosedur dan fungsi
 - 5.4 Membuat program menggunakan array
 - 5.5 Membuat program untuk akses file
 - 5.6 Mengkompilasi program
6. Menggunakan Library atau Komponen Pre- Existing
 - 6.1 Melakukan pemilihan unit-unit reuse yang potensial
 - 6.2 Melakukan integrasi library atau komponen pre-existing dengan source code yang ada
 - 6.3 Melakukan pembaharuan library atau komponen pre-existing yang digunakan