**MODUL IV : JQuery**

**KODE MODUL** : TIK.RPL02.002.001.01

**DESKRIPSI MODUL** : Formulir pendaftaran dengan validasi jQuery

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Elemen Kompetensi | Indikator Kinerja | Jml Jam | halaman |
| 1 | Menggunakan jQuery untuk validasi form | Form dapat divalidasi dengan jQuery untuk memastikan tidak ada kolom yang kosong dan panjang kata sandi minimal 6 karakter sebelum dikirim. | 2 |  |
| 2 | Memberikan feedback visual dengan jQuery | Elemen input yang tidak valid akan ditandai dengan warna merah dan pesan kesalahan akan ditampilkan menggunakan alert. | 1 |  |
|  | Mengelola event form dengan jQuery | Event submit form ditangani dengan jQuery untuk mencegah pengiriman form standar dan mengeksekusi validasi. | 1 |  |

**PRELAB**

Persiapan Modul Praktikum Validasi Form dengan JavaScript:

1. jQuery (<https://www.w3schools.com/jquery/>) untuk memahami dasar-dasar jQuery dan manipulasi DOM.
2. Validasi Form dengan jQuery (<https://www.w3schools.com/jquery/jquery_validation.asp>) untuk mempelajari cara melakukan validasi form menggunakan jQuery.

**DAFTAR PERTANYAAN**

1. Bagaimana jQuery mencegah pengiriman form secara default dan apa fungsi dari ***event.preventDefault()*** dalam konteks validasi form?

jQuery mencegah pengiriman form secara default dengan menggunakan metode `event.preventDefault()` dalam konteks validasi form. Ketika suatu form disubmit, browser secara default akan mengirimkan data form tersebut ke server, menyebabkan halaman untuk direfresh. Namun, dengan menggunakan jQuery dan metode `event.preventDefault()`, kita dapat menghentikan perilaku default ini. Dengan demikian, dalam konteks validasi form, ketika pengguna mengklik tombol submit, jQuery dapat menjalankan fungsi validasi untuk memeriksa entri pengguna sebelum data dikirim ke server. Jika validasi gagal, `event.preventDefault()` mencegah pengiriman form secara default, memungkinkan pengguna untuk memperbaiki kesalahan mereka tanpa memperbarui halaman.

1. Apa keuntungan menggunakan jQuery untuk validasi form dibandingkan dengan JavaScript murni?

kemudahan penggunaan dan konsistensi lintas browser. jQuery menyediakan fungsi-fungsi siap pakai yang mempercepat proses pengembangan, seperti metode untuk memilih elemen DOM dengan sintaks yang sederhana dan memanipulasi properti mereka dengan mudah. Selain itu, jQuery mengatasi perbedaan dalam perilaku browser dengan memberikan abstraksi yang konsisten, sehingga pengembang tidak perlu khawatir tentang penyesuaian kode untuk setiap browser yang berbeda. Ini membuat proses pengembangan menjadi lebih efisien dan memungkinkan pengembang untuk fokus pada logika validasi form daripada menangani perbedaan implementasi browser.

1. Bagaimana cara jQuery memberikan feedback visual kepada pengguna ketika terdapat kesalahan input pada form?

dengan cara mengubah tampilan elemen-elemen form yang bermasalah secara dinamis. Biasanya, pengembang akan menambahkan kelas CSS atau properti gaya langsung ke elemen-elemen yang memerlukan perhatian, seperti menyorot kotak input dengan warna merah atau menampilkan pesan kesalahan di bawah input. Dengan menggunakan metode addClass(), removeClass(), atau css()

1. Dalam kode yang diberikan, bagaimana kita bisa memodifikasi skrip untuk menampilkan pesan kesalahan validasi tanpa menggunakan alert() dan sebaliknya menampilkan pesan di bawah input yang relevan di dalam form?

kita dapat menggunakan jQuery untuk menambahkan elemen baru, seperti <span>, di bawah setiap input yang memerlukan validasi. Setelah melakukan validasi, jika terjadi kesalahan, kita bisa menambahkan elemen <span> ini secara dinamis menggunakan metode append() atau after() jQuery. Pesan kesalahan yang relevan dapat dimasukkan ke dalam elemen <span> ini. Selain itu, kita dapat mengatur gaya CSS untuk elemen <span> ini agar terlihat jelas sebagai pesan kesalahan.

**TEORI SINGKAT**

Validasi form adalah proses yang memeriksa apakah input yang diberikan oleh pengguna memenuhi kriteria tertentu sebelum form dikirimkan ke server. Dalam hal ini, kita menggunakan jQuery, sebuah library JavaScript yang memudahkan manipulasi DOM, handling event, dan animasi.

jQuery menyediakan metode yang mudah untuk menangani validasi form. Dengan jQuery, kita dapat mengintersep event submit dari form dan memeriksa input menggunakan fungsi yang disediakan oleh jQuery.

* Menghentikan Pengiriman Form

Dengan menggunakan event.preventDefault(), jQuery mencegah form dari dikirim secara standar. Ini memungkinkan kita untuk menjalankan validasi dan hanya mengirim form jika semua validasi telah terpenuhi.

* Memberikan Feedback Visual

jQuery memungkinkan kita untuk memanipulasi CSS elemen-elemen HTML secara langsung. Contohnya, jika input tidak valid, kita dapat mengubah warna border menjadi merah dengan .css() untuk menunjukkan kesalahan. Hal ini memberikan respons yang cepat dan intuitif kepada pengguna.

* Penanganan Kesalahan Input

Dalam kode yang diberikan, validasi dijalankan saat form di-submit. Setiap input diperiksa dan jika terdapat input yang tidak memenuhi kriteria, seperti kolom kosong atau kata sandi yang terlalu pendek, jQuery akan menampilkan pesan kesalahan menggunakan alert(). Namun, dalam praktik terbaik, pesan kesalahan sebaiknya ditampilkan dalam form itu sendiri, bukan melalui alert() yang mengganggu.

* jQuery dan DOM

jQuery membuat seleksi dan manipulasi elemen DOM menjadi lebih sederhana. Dengan menggunakan selektor seperti $('#elementId'), kita dapat dengan cepat memilih elemen HTML dan melakukan tindakan terhadapnya, seperti mengambil atau mengubah nilai dengan .val() atau .css().

Penggunaan jQuery untuk validasi form memperkaya interaksi pengguna dan meningkatkan usability dari web aplikasi, sekaligus memastikan data yang dikumpulkan adalah valid sebelum diproses lebih lanjut oleh server.

* jQuery dan JavaScript sering dibahas bersamaan karena jQuery adalah library yang dibangun di atas JavaScript. Namun, ada beberapa perbedaan utama di antara keduanya:
* **JavaScript**
* JavaScript adalah bahasa pemrograman yang digunakan untuk membuat halaman web menjadi interaktif. Ini merupakan salah satu dari tiga bahasa inti yang digunakan dalam pengembangan web, dua lainnya adalah HTML dan CSS.
* JavaScript dapat digunakan tanpa library atau framework tambahan untuk membuat logika web yang kompleks.
* Menulis script dengan JavaScript murni untuk manipulasi DOM atau event handling bisa menjadi sangat verbos dan kompleks terutama saat berurusan dengan browser cross-compatibility.
* JavaScript murni memberikan kontrol penuh kepada pengembang, yang memungkinkan untuk optimisasi performa dan scripting yang lebih kustom.
* **jQuery**
* jQuery adalah library JavaScript yang cepat, kecil, dan kaya fitur. Ia dirancang untuk memudahkan penulisan script pada sisi klien.
* jQuery menangani banyak kompleksitas yang ada pada JavaScript murni, seperti manipulasi DOM dan AJAX calls, dengan sintaks yang lebih singkat dan mudah dipahami.
* Menggunakan jQuery dapat mengurangi jumlah kode yang harus ditulis, terutama untuk animasi, event handling, dan AJAX.
* jQuery memberikan kompatibilitas cross-browser yang baik, mengurangi kekhawatiran pengembang mengenai perbedaan perilaku di browser yang berbeda.
* Dikarenakan jQuery adalah library yang terpisah, hal ini berarti perlu memastikan library tersebut dimuat pada halaman web sebelum dapat menggunakannya.

Secara singkat, jQuery dirancang untuk mempermudah dan mempercepat pengembangan JavaScript. Namun, dengan peningkatan dukungan standar web modern dan ES6+, beberapa pengembang memilih untuk menggunakan JavaScript murni untuk menghindari ketergantungan pada library eksternal.

**ELEMEN KOMPETENSI**

**Deskripsi :** Formulir pendaftaran dengan validasi jQuery

**Kompetensi Dasar** : Mampu menggunakan jQuery untuk seleksi dan manipulasi DOM, handling event, dan melakukan AJAX request dengan sintaks yang lebih ringkas dan efisien dibandingkan JavaScript murni.

**Latihan 1**

* 1. Copy code kadalam sublime atau visual code

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Form Registrasi</title>

<style>

body {

font-family: Arial, sans-serif;

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

margin: 0;

}

form {

background-color: #f2f2f2;

padding: 20px;

border-radius: 5px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

width: fit-content;

max-width: 90%;

}

h2 {

color: #333;

text-align: center;

}

label {

display: block;

margin-top: 10px;

}

input[type=text], input[type=email], input[type=password] {

width: calc(100% - 20px);

padding: 10px;

margin-top: 5px;

margin-bottom: 20px;

border: 1px solid #ccc;

border-radius: 4px;

}

button {

background-color: #4CAF50;

color: white;

padding: 10px 20px;

border: none;

border-radius: 4px;

cursor: pointer;

width: 100%;

}

button:hover {

background-color: #45a049;

}

@media (max-width: 768px) {

body {

padding: 20px;

flex-direction: column;

}

form {

width: 100%;

margin: 0;

box-shadow: none;

}

}

</style>

</head>

<body>

<form id="registrationForm">

<h2>Daftar Akun Baru</h2>

<label for="name">Nama:</label>

<input type="text" id="name" name="name">

<label for="email">Email:</label>

<input type="email" id="email" name="email">

<label for="password">Kata Sandi (minimal 6 karakter):</label>

<input type="password" id="password" name="password">

<label for="confirmPassword">Konfirmasi Kata Sandi:</label>

<input type="password" id="confirmPassword" name="confirmPassword">

<button type="submit">Daftar</button>

</form>

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>

<script>

$(document).ready(function() {

$('#registrationForm').submit(function(event) {

event.preventDefault(); // Prevent the form from submitting via the browser.

let isValid = true; // Flag for validation state.

const password = $('#password').val().trim();

const confirmPassword = $('#confirmPassword').val().trim();

// Define array of inputs to validate.

const inputs = ['#name', '#email', '#password', '#confirmPassword'];

// Loop through inputs for validation.

inputs.forEach(function(input) {

const $input = $(input);

if ($input.val().trim() === "") {

$input.css('borderColor', 'red'); // Change border to red if empty.

alert(`Isian kolom ${$input.attr('name')} tidak boleh kosong`); // Alert for empty input.

isValid = false; // Update validation flag.

return false; // Break out of the loop.

} else {

$input.css('borderColor', ''); // Reset border if input is filled.

}

});

// Validate password length

if (password.length < 6) {

$('#password').css('borderColor', 'red');

alert('Kata Sandi harus memiliki minimal 6 karakter.');

isValid = false;

}

// Validate password and confirm password

if (password !== confirmPassword) {

$('#confirmPassword').css('borderColor', 'red');

alert('Kata Sandi dan Konfirmasi Kata Sandi harus sama.');

isValid = false;

}

if (isValid) {

alert("Pendaftaran berhasil!");

// Uncomment the next line to actually submit the form.

// this.submit(); // Submit form if all inputs are valid.

}

});

});

</script>

</body>

</html>

* 1. Hasilnya ketika data semua terisi penuh dan dan benar ketika klik Daftar akan ada alert pendaftaran berhasil!

|  |
| --- |
|  |

* 1. Simpanlah ke dalams \*.html (contoh : web\_NAMA\_ANDA.html) pada sebuah folder.

**Latihan 2**

* 1. pembagian aplikasi perkelompok dengan data form masing – masing aplikasi yang sudah di tentukan dengan menambahkan fungsi jQuery dan CSS. Seperti latihan 1 memunculkan alert pada semua form dan tombol.

**Kelompok 3 Aplikasi IC:**

Minimal merekam informasi:

* deskripsi masalah
* deskripsi fitur
* deskripsi label
* kategori masalah berupa: Klasifikasi, prediksi numerik, prediksi probabilistik, Deteksi, Rekognisi;
* tanggal mulai dan selesai pemodelan
* dan status penyelesaian perancangan berupa pilihan: selesai, berlangsung, gagal; serta deskripsi catatan akhir kegiatan. Sistem juga merekap waktu perekaman, dan waktu update informasi terakhir.

|  |
| --- |
|  |

Source code:

|  |
| --- |
| **CSS:**  body {      padding: 0;      margin: 0;      box-sizing: border-box;      overflow-x: hidden;      display: flex;      justify-content: center;      background-image: url("wave.svg");      background-repeat: no-repeat;      background-position: bottom;      background-color: #f4bd71; /\* Warna latar belakang form \*/  }  .ic-form {      background-image: url("wave.svg");      background-repeat: no-repeat;      background-position: bottom;      margin-top: 15px;      margin-bottom: 15px;      width: 900px;      height: 150vh;      display: grid;      place-items: center;      /\* border: 1px solid black; \*/      border-radius: 20px;      -webkit-box-shadow: 0px 9px 26px -10px rgba(50, 48, 48, 0.65);      -moz-box-shadow: 0px 9px 26px -10px rgba(0,0,0,0.65);      box-shadow: 0px 9px 26px -10px rgba(16, 15, 15, 0.65);      background-color: rgba(45, 43, 43, 0.6); /\* Warna latar belakang form \*/  }  .category-problems, .status-info{      padding-top: 10px;      padding-bottom: 10px;      text-align: center;  }  label{      margin-top: 8px;      margin-bottom: 20px;      font-size: 15px;      text-align: center;      font-weight: 700;  }  textarea{      border-radius: 5px;  }  .masalah, .fitur, .label, .catatan{      margin-bottom: 10px;  }  #dataTable {      width: 100%;      border-collapse: collapse;      background: rgb(253,55,29);      background: -moz-linear-gradient(90deg, rgba(253,55,29,1) 50%, rgba(252,176,69,1) 100%);      background: -webkit-linear-gradient(90deg, rgba(253,55,29,1) 50%, rgba(252,176,69,1) 100%);      background: linear-gradient(90deg, rgba(253,55,29,1) 50%, rgba(252,176,69,1) 100%);      filter: progid:DXImageTransform.Microsoft.gradient(startColorstr="#fd371d",endColorstr="#fcb045",GradientType=1);      color: white; /\* ubah warna teks \*/  }  #dataTable th, #dataTable td {      padding: 8px;      text-align: left;      border-bottom: 2px solid rgb(1, 1, 1); /\* garis bawah \*/      border-right: 1px solid black; /\* garis kanan, warna hitam \*/  }  #dataTable tr:last-child th {      border-bottom: 2px solid rgb(1, 1, 1); /\* garis bawah di baris terakhir untuk header \*/      border-right: 2px solid black; /\* garis kanan di baris terakhir untuk header, warna hitam \*/  }  #dataTable th {      font-weight: bold;      border-right: 2px solid black; /\* garis kanan untuk header, warna hitam \*/  }  button{      width: 120px;      margin-top: 25px;      height: 5vh;      margin-bottom: 20px;      background-color: orange;      color: white;      border: none;      border-radius: 5px;  }  button:hover{      background-color: orangered;  }    **HTML:**  <!DOCTYPE html>  <html lang="en">  <head>      <meta charset="UTF-8">      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <title>Intelligence Creation Form</title>      <!-- Style -->      <link rel="stylesheet" href="ic1.css">  </head>  <body>      <form class="ic-form" id="ic-form">          <!-- Deskripsi Masalah  -->          <label for="deskripsi-masalah">Deskripsi Masalah : </label>          <textarea placeholder="Masukkan deskripsi terkait masalah" name="masalah" id="masalah" cols="60" rows="5" class="masalah"></textarea>            <!-- Deskripsi Fitur -->          <label for="deskripsi-fitur">Deskripsi Fitur : </label>          <textarea placeholder="Masukkan deskripsi-fitur" name="fitur" id="fitur" cols="60" rows="5" class="fitur"></textarea>            <!-- Deskripsi Label -->          <label for="deskripsi-label">Deskripsi Label : </label>          <textarea placeholder="Masukkan deskripsi label" name="label" id="label" cols="40" rows="5" class="label"></textarea>          <div class="category-problems">              <!-- Kategori Masalah -->              <div class="label-category">                  <label for="kategori-masalah">Kategori Masalah : </label>              </div>              <select name="kategori" id="kategori">                  <option value="Pilih Kategori">--Pilih Kategori--</option>                  <option value="Klasifikasi">Klasifikasi</option>                  <option value="Prediksi">Prediksi</option>                  <option value="Prediksi Probabilistik">Prediksi Probabilistik</option>                  <option value="Deteksi">Deteksi</option>                  <option value="Rekognisi">Rekognisi</option>              </select>          </div>          <!-- Tanggal Mulai -->          <label for="start-date">Awal Tanggal : </label>          <input type="date" name="awal" id="tanggal-awal">          <!-- Akhir Waktu -->          <label for="last-date">Akhir Tanggal : </label>          <input type="date" name="akhir" id="tanggal-akhir">          <div class="status-info">               <!-- Status Penyelesaian -->               <div class="status-category">                  <label for="status">Status Penyelesaian : </label>               </div>              <select name="status" id="status">                  <option value="Pilih Status">--Pilih Status--</option>                  <option value="Selesai">Selesai</option>                  <option value="Berlangsung">Berlangsung</option>                  <option value="Gagal">Gagal</option>              </select>          </div>          <!-- Catatan -->          <label for="catatanAkhir">Deskripsi Catatan : </label>          <textarea placeholder="Masukkan Catatan" name="catatan" id="catatan" cols="60" rows="5" class="catatan"></textarea>          <button type="submit">SUBMIT</button>          <!-- POP UP -->          <div class="pop-up"></div>      </form>      <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"></script>  <script>      $(document).ready(function() {          $('#ic-form').submit(function(event) {              event.preventDefault(); // Prevent the form from submitting via the browser.              let isValid = true; // Flag for validation state.              // const password = $('#password').val().trim();              // const confirmPassword = $('#confirmPassword').val().trim();              // Define array of inputs to validate.              const inputs = ['#masalah', '#fitur', '#label', '#catatan'];              const currentTime = new Date();              // Loop through inputs for validation.              inputs.forEach(function(input) {                  const $input = $(input);                  if ($input.val().trim() === "") {                      $input.css('borderColor', 'red'); // Change border to red if empty.                      alert(`Isian kolom ${$input.attr('name')} tidak boleh kosong`); // Alert for empty input.                      isValid = false; // Update validation flag.                      return false; // Break out of the loop.                  } else {                      $input.css('borderColor', ''); // Reset border if input is filled.                  }              });              if (isValid) {                  const lastCurrentTime = currentTime;                  alert(`Entry Berhasil \n                  Waktu Penyimpanan = ${currentTime} \n                  Waktu Akhir Penyimpanan = ${lastCurrentTime}`);                  // Uncomment the next line to actually submit the form.                  // this.submit(); // Submit form if all inputs are valid.              }          });      });  </script>  </body>  </html> |

**KESIMPULAN**

Pada pertemuan praktikum pemograman web kali ini kita dapat menggunakan jQuery untuk seleksi dan manipulasi DOM, handling event, dan melakukan AJAX request dengan sintaks yang lebih ringkas dan efisien dibandingkan JavaScript murni. Kita juga dapat membuat form baru dengan isi: deskripsi masalah , deskripsi fitur, deskripsi label, kategori masalah berupa (Klasifikasi, prediksi numerik, prediksi probabilistik, Deteksi, Rekognisi), tanggal mulai dan selesai pemodelan, dan status penyelesaian perancangan berupa pilihan: selesai, berlangsung, gagal; serta deskripsi catatan akhir kegiatan. Sistem juga merekap waktu perekaman, dan waktu update informasi terakhir.

**CEK LIST (√)**

1. Menerapkan validasi pada form input data dengan jQuery ()
2. Menangani event submit form dengan jQuery untuk memvalidasi data sebelum pengiriman ()

**FORM UMPAN BALIK**

|  |  |  |  |
| --- | --- | --- | --- |
| **Elemen Kompetensi** | **Tingkat Kesulitan**  **(1: sangat mudah,2: mudah,3: biasa,4: sulit,5: sangat sulit)** | **Tingkat Ketertarikan**  **(1: tidak tertarik, 2: cukup tertarik, 3: tertarik; 4: sangat tertarik)** | **Waktu Penyelesaian**  **Dalam menit** |
| **Menangani event submit form untuk memvalidasi data sebelum pengiriman** | **Mudah** | **Tertarik** | **40 menit** |
| **Menerapkan validasi pada form input data dengan jQuery** | **Mudah** | **Tertarik** | **40 menit** |