**Pertemuan 4 : Java Script Part 2**

**Tugas 4 : PopUp**

|  |  |
| --- | --- |
| **Nama / Nim** | **:SANDY EKA AKHMAD FIRDAUS** |
| **Kelas** | **:H** |
| **Pengajar** | **:Ahmad Risky Ramadhani** |

* 1. **Tujuan**

1. Mampu memahami dasar – dasar Javascript.
2. Mampu memahami penggunaan Javascript.
3. Mampu memahami DOM pada Javascript.
   1. **Alat dan Bahan**
4. Laptop/pc
5. Modul Praktikum Web 2024
6. TextEditor (Visual Studio Code / Sublime Text)
   1. **Langkah-langkah**
7. PopUp

Source code html :

|  |
| --- |
| <section class="pilihan">  <h1>Top Up</h1>  <div class="gallery">  <div class="gallery-item" id="openPopupBtn">  <img src="emel.jpg" alt="Gambar 1">  <div class="gallery-caption">MLBB</div>  </div>  <div class="gallery-item" id="openPopupBtn" style="background-color: #000;">  <img src="ff-logo.jpg" alt="Gambar 2" >  <div class="gallery-caption">Free Fire</div>  </div>  <div class="gallery-item" id="openPopupBtn">  <img src="pubg.jpg" alt="Gambar 3">  <div class="gallery-caption">PUBG</div>  </div>  </div>  <!-- Pop-Up Form -->  <div id="popupForm" class="popup">  <div class="popup-content">  <span id="closePopupBtn" class="close-btn">&times;</span>  <h2>Form Top Up</h2>  <form id="myForm">  <label for="name">ID:</label>  <input type="number" id="id" name="id" required>  <label for="email">Server:</label>  <input type="number" id="server" name="server" required>  <label for="category">Jumlah Top Up:</label>  <select id="topup" name="topup" style="text-align: center;">  <option value="">--- Pilih Jumlah Top Up ---</option>  <option value="100 Diamond">100 Diamond</option>  <option value="200 Diamond">200 Diamond</option>  <option value="300 Diamond">300 Diamond</option>  </select>  <label for="category">Metode Pembayaran:</label>  <select id="metode" name="metode" style="text-align: center;">  <option value="">--- Pilih Metode Pembayaran ---</option>  <option value="Indomart">Indomart</option>  <option value="Alfamart">Alfamart</option>  <option value="Transfer Bank">Transfer Bank</option>  <option value="Pulsa">Pulsa</option>  </select>  <button type="submit">Kirim</button>  </form>  </div>  <!-- Pop-Up Transaksi -->  <div id="processingPopup" class="popup">  <div class="popup-content">  <h2>Transaksi Sedang Diproses...</h2>  </div>  </div>  </div>  </section> |

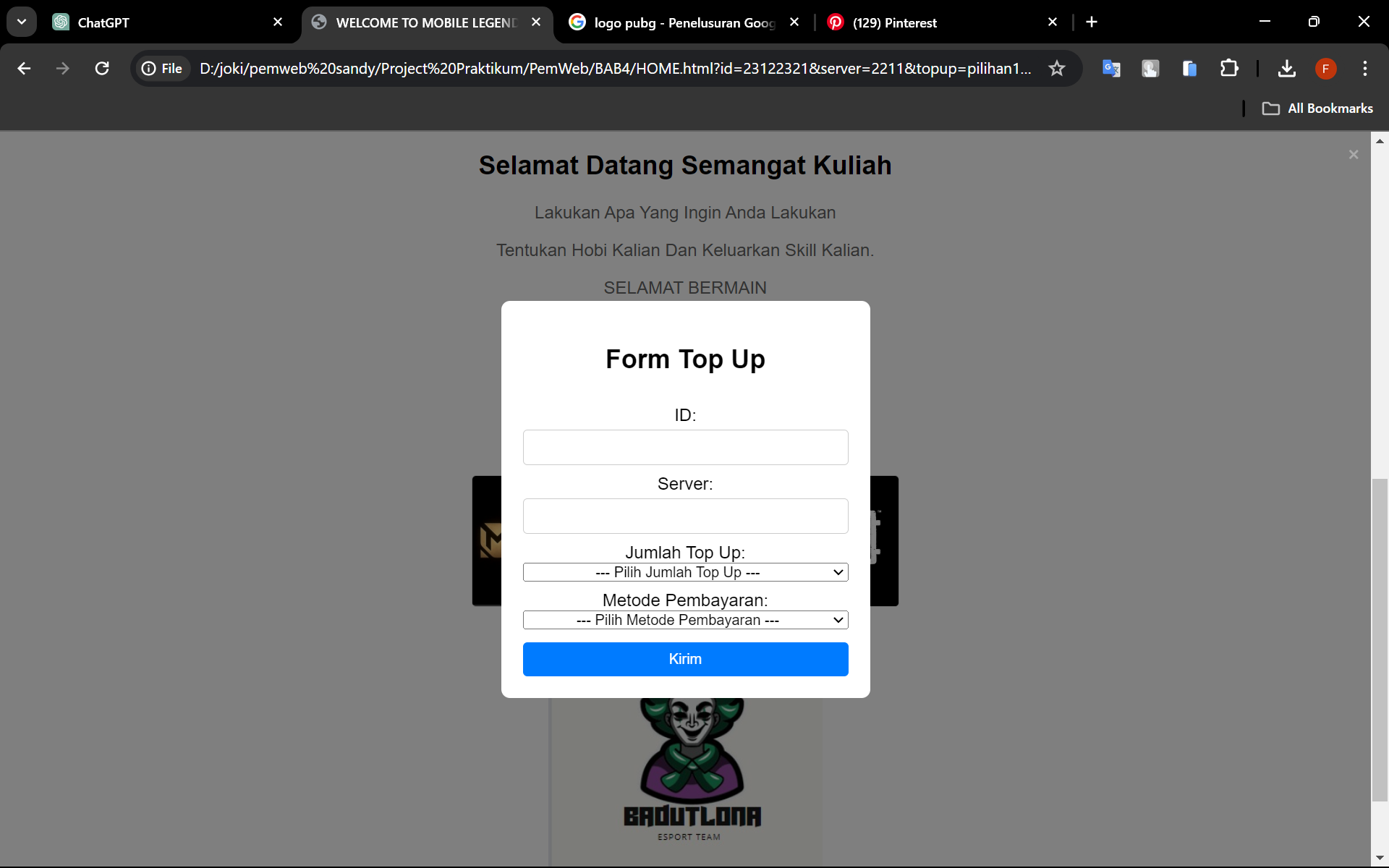
Source code css :

|  |
| --- |
| /\* galeri \*/  .gallery {  display: flex;  flex-wrap: wrap;  gap: 16px;  padding: 16px;  justify-content: center;  }  .gallery-item {  width: calc(10% - 3px); /\* Tiga item per baris, kurangi jarak antar item \*/  border: 1px solid #ddd;  border-radius: 4px;  overflow: hidden;  position: relative;  }  .gallery-item :hover{  cursor: pointer;  }  .gallery-item img {  width: 100%;  display: block;  }  .gallery-caption {  position: absolute;  bottom: 0;  background: rgba(0, 0, 0, 0.7);  color: white;  padding: 8px 0px;  width: 100%;  text-align: center;  }  /\* Mengatur gaya umum \*/  body {  font-family: Arial, sans-serif;  margin: 0;  padding: 0;  }  /\* Gaya untuk pop-up \*/  .popup {  display: none; /\* Awalnya tersembunyi \*/  position: fixed;  top: 0;  left: 0;  width: 100%;  height: 100%;  background-color: rgba(0, 0, 0, 0.5);  justify-content: center;  align-items: center;  }  .popup-content {  background-color: white;  padding: 20px;  border-radius: 8px;  width: 300px; /\* Lebih kecil \*/  }  .close-btn {  position: absolute;  top: 10px;  right: 10px;  cursor: pointer;  font-size: 20px;  color: #aaa;  }  .close-btn:hover {  color: black;  }  /\* Gaya untuk form \*/  form {  display: flex;  flex-direction: column;  }  form label {  margin-top: 8px;  }  form input,  form textarea {  padding: 8px;  margin-top: 4px;  border: 1px solid #ccc;  border-radius: 4px;  }  form button {  margin-top: 12px;  padding: 8px;  background-color: #007BFF;  color: white;  border: none;  border-radius: 4px;  cursor: pointer;  }  form button:hover {  background-color: #0056b3;  } |

Source code javascript :

|  |
| --- |
| const openPopupBtn = document.querySelectorAll('.gallery-item');  const popupForm = document.getElementById('popupForm');  const closePopupBtn = document.getElementById('closePopupBtn');  const myForm = document.getElementById('myForm');  const processingPopup = document.getElementById('processingPopup');  function openPopup() {  popupForm.style.display = 'flex';  }  function closePopup() {  popupForm.style.display = 'none';  }  function openProcessingPopup() {  processingPopup.style.display = 'flex';  }  function closeProcessingPopup() {  processingPopup.style.display = 'none';  }  // Event listener untuk membuka pop-up form  openPopupBtn.forEach((openPopupBtn) =>{  openPopupBtn.addEventListener('click', openPopup);  });  // Event listener untuk menutup pop-up form  closePopupBtn.addEventListener('click', closePopup);  // Menutup pop-up form ketika pengguna mengklik di luar konten form  window.addEventListener('click', function(event) {  if (event.target === popupForm) {  closePopup();  }  }); |

Hasil Tampilan :



Gambar 4.1 Tampilan Desktop Hasil Index.html

Analisa :

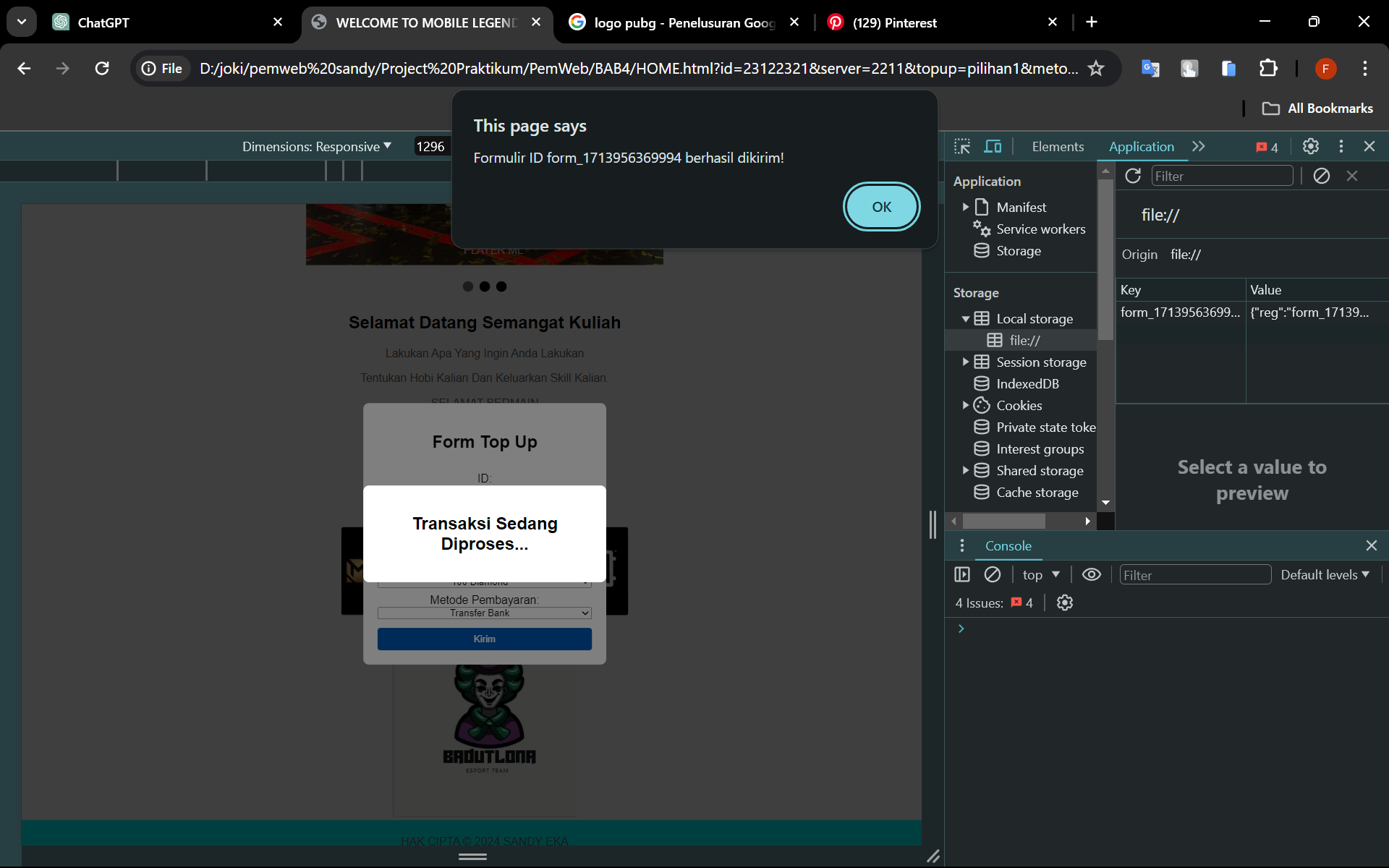
Kode JavaScript di atas mengatur interaksi pengguna dengan elemen pop-up form. Kode ini memilih semua elemen dengan kelas .gallery-item (openPopupBtn) dan menambahkan event listener untuk masing-masing elemen, sehingga saat salah satu dari mereka diklik, fungsi openPopup() akan dieksekusi untuk membuka pop-up form (popupForm). Kode juga menambahkan event listener ke tombol tutup pop-up (closePopupBtn) untuk menutup pop-up form saat tombol diklik. Selain itu, ada event listener yang memantau klik di luar konten form, sehingga jika pengguna mengklik di luar area pop-up form (popupForm), pop-up akan ditutup. Fungsi tambahan (openProcessingPopup dan closeProcessingPopup) juga disediakan untuk membuka dan menutup pop-up transaksi, meskipun mereka tidak digunakan dalam skrip ini. Secara keseluruhan, kode ini mengelola tampilan dan interaksi pop-up form berdasarkan aksi pengguna.

1. Local Storage, Asynchronous

Source code :

|  |
| --- |
| // Event listener untuk menangani pengiriman formulir  function generateFormID() {  // Menggunakan timestamp untuk menghasilkan ID unik  return `form\_${Date.now()}`;  }  // Event listener untuk menangani pengiriman formulir  myForm.addEventListener('submit', function(event) {  event.preventDefault(); // Menghentikan pengiriman formulir secara default  // Membuka pop-up transaksi  openProcessingPopup();  // Mendapatkan data formulir  const id = document.getElementById('id').value;  const server = document.getElementById('server').value;  const topup = document.getElementById('topup').value;  const metode = document.getElementById('metode').value;  // Menghasilkan ID form unik  const formID = generateFormID();  // Data yang akan disimpan ke local storage  const formData = {  reg: formID, // Menambahkan ID form  id,  server,  topup,  metode,  status: 'pending' // Menambahkan status 'pending'  };  // Menyimpan data ke local storage  localStorage.setItem(formID, JSON.stringify(formData));  // Tutup pop-up transaksi setelah beberapa waktu (simulasi pemrosesan)  setTimeout(() => {  closeProcessingPopup();  alert(`Formulir ID ${formID} berhasil dikirim!`);  // Bersihkan form setelah dikirim  myForm.reset();  // Tutup pop-up form  closePopup();  }, 3000);  }); |

Hasil Tampilan :



Gambar 2.4 Tampilan Desktop Hasil login.html

Analisa :

Kode di atas adalah bagian dari skrip JavaScript yang menangani pengiriman formulir dengan menambahkan event listener untuk elemen formulir (myForm). Saat formulir dikirimkan, event listener mencegah pengiriman default formulir dengan event.preventDefault() dan membuka pop-up transaksi menggunakan fungsi openProcessingPopup(). Kode kemudian mengambil nilai input dari elemen formulir berdasarkan ID (id, server, topup, dan metode) dan menghasilkan ID form unik (formID) menggunakan fungsi generateFormID(). Kemudian, data formulir yang dikumpulkan disimpan sebagai objek (formData) ke localStorage dengan ID unik sebagai kuncinya. Setelah simulasi pemrosesan (3 detik), pop-up transaksi ditutup, alert ditampilkan untuk memberi tahu pengguna bahwa formulir berhasil dikirim, dan formulir direset. Setelah itu, pop-up form ditutup dengan fungsi closePopup(). Kode ini mengotomatisasi proses pengiriman formulir, penyimpanan data, dan pemberitahuan pengguna tentang hasil pengiriman formulir.