

# LAPORAN PROGRES MINGGU 3: IMPLEMENTASI CRUD DASAR

## Proyek Database "Eventify" - Kelompok 9

Disusun oleh:

- Rafael Mahardika Arya Dewamurti (24/536279/PA/22755)
- Bobby Rahman Hartanto (24/539383/PA/22903)

## I. PENDAHULUAN

Sesuai dengan panduan proyek, tugas Minggu 3 adalah mengimplementasikan operasi **CRUD** (**C**reate, **R**ead, **U**pdate, **D**elete) untuk entitas utama. Laporan ini mendokumentasikan progres kami dalam membangun aplikasi web fungsional pertama yang terhubung langsung dengan database eventify\_db yang telah kami rancang di Week 2.

Kami telah berhasil membangun arsitektur 3-tier sederhana:

1. **Database (MySQL):** Pondasi yang menyimpan data, di-setup menggunakan schema\_eventify.sql dari Week 2.
2. **Backend (Node.js):** "Mesin" atau server yang bertindak sebagai perantara, ditulis dalam file backend.js.
3. **Frontend (HTML/CSS/JS):** "Tampilan" atau UI yang dilihat pengguna, ditulis dalam file frontend.html.

Laporan ini akan berfokus pada demo fungsionalitas dan penjelasan singkat arsitektur yang kami gunakan.

## II. ARSITEKTUR & PENJELASAN KONEKSI

Untuk proyek ini, kami tidak menghubungkan frontend langsung ke database karena sangat tidak aman. Kami menggunakan arsitektur *client-server* di mana frontend dan backend adalah dua program terpisah yang "berbicara" satu sama lain.

### 1. Frontend (frontend.html)

Ini adalah dasbor UI yang kami buat menggunakan HTML dan TailwindCSS. Bagian ini "buta" (agnostic), dia tidak tahu apa-apa soal database. Dia hanya tahu cara "meminta" data ke sebuah alamat URL (yaitu <http://localhost:3001/api/events>) menggunakan JavaScript fetch.

### 2. Backend (backend.js)

Ini adalah "otak" aplikasi. Dia berjalan di localhost:3001 menggunakan Node.js dan

Express. Backend adalah satu-satunya yang memegang "kunci" (password) ke database. Tugasnya adalah menerima permintaan dari frontend, menerjemahkannya menjadi perintah SQL (misal: SELECT \* FROM Events), mengeksekusinya ke database, dan mengirimkan hasilnya kembali ke frontend sebagai JSON.

### 3. Database (MySQL/XAMPP)

Ini adalah "kulkas" kami, yang berjalan di localhost (biasanya port 3306). Dia hanya menyimpan data dan menjalankan perintah SQL yang diberikan oleh backend.

## III. DEMO FUNGSIONALITAS APLIKASI

Untuk membuktikan bahwa koneksi ini berjalan, kami mendemokan 4 (empat) operasi CRUD pada entitas utama kami, yaitu **Events**.

### A. Tampilan Antarmuka (UI)

Kami mendesain UI profesional dengan tema *dark mode* dan efek *glassmorphism* (kaca buram) menggunakan TailwindCSS. Layout dibagi dua: formulir di kiri, dan daftar data di kanan.

The screenshot shows the Eventify Dashboard interface. On the right side, there is a card titled "Daftar Event (Read)" containing two event entries:

- Studi Banding with Universitas Brawijaya**  
Koordinat: -7.7676, 110.376  
Minggu, 23 November 2025 pukul 08.00  
Studi Banding antara BEM KM FMIPA UGM dan BEM FILKOM UB yang akan dilaksanakan di Auditorium Herman Yohanes Lt.7 FMIPA UGM.
- Malam Apresiasi 2025**  
Koordinat: -7.7747, 110.376  
Rabu, 19 November 2025 pukul 17.30  
Agenda tahunan KM FMIPA UGM yang ditujukan untuk mengapresiasi pencapaian di bidang akademik dan non-akademik. Acara ini menjadi panggung penghargaan sekaligus perayaan atas karya, inovasi, dan kontribusi seluruh civitas FMIPA UGM. Rangkaian Malam Apresiasi meliputi pentas seni oleh mahasiswa, penampilan dari guest star, serta penganugerahan apresiasi bagi sivitas FMIPA UGM atas pencapaian akademik maupun non-akademik.

On the left side, there is a form titled "Buat Event Baru" with fields for "Nama Acara", "Lokasi (Geser Pin)", and a text area for "Deskripsi". The "Lokasi (Geser Pin)" field contains a map of a city area with a pin indicating the location.

The screenshot shows the "Buat Event Baru" (Create New Event) form. It includes the following fields:

- Tanggal Mulai: dd/mm/yyyy --::--
- Kuota Peserta: 0
- Deskripsi: Jelaskan detail acaramu di sini...
- Submit Event button

- Penjelasan Gambar:** Screenshot ini menunjukkan tampilan utama dasbor "Eventify". Kolom kiri adalah formulir CREATE/UPDATE, dan kolom kanan adalah panel READ/DELETE.

## B. Operasi CREATE (Membuat Event)

Pengguna mengisi formulir di sebelah kiri (termasuk memilih lokasi di peta interaktif Leaflet.js) dan menekan "Submit Event".

The screenshot displays the Eventify application's interface. On the left, the 'Buat Event Baru' (Create New Event) panel is active, containing fields for event details and a map. On the right, the 'Daftar Event (Read)' (Event List) panel shows existing events.

**Buat Event Baru (Left Panel):**

- Nama Acara:** OmahTI Academy
- Lokasi (Geser Pin):** A map of the Universitas Gadjah Mada area in Yogyakarta, Indonesia, with several red markers indicating event locations. A blue pin marks the current location being selected.
- Tanggal Mulai:** 02/07/2026 09:30
- Kuota Peserta:** 250
- Deskripsi:** OmahTI Academy merupakan program pelatihan intensif di bidang Teknologi Informasi langsung bersama mentor-mentor yang ahli di bidangnya. Kegiatan ini menjadi salah satu bentuk

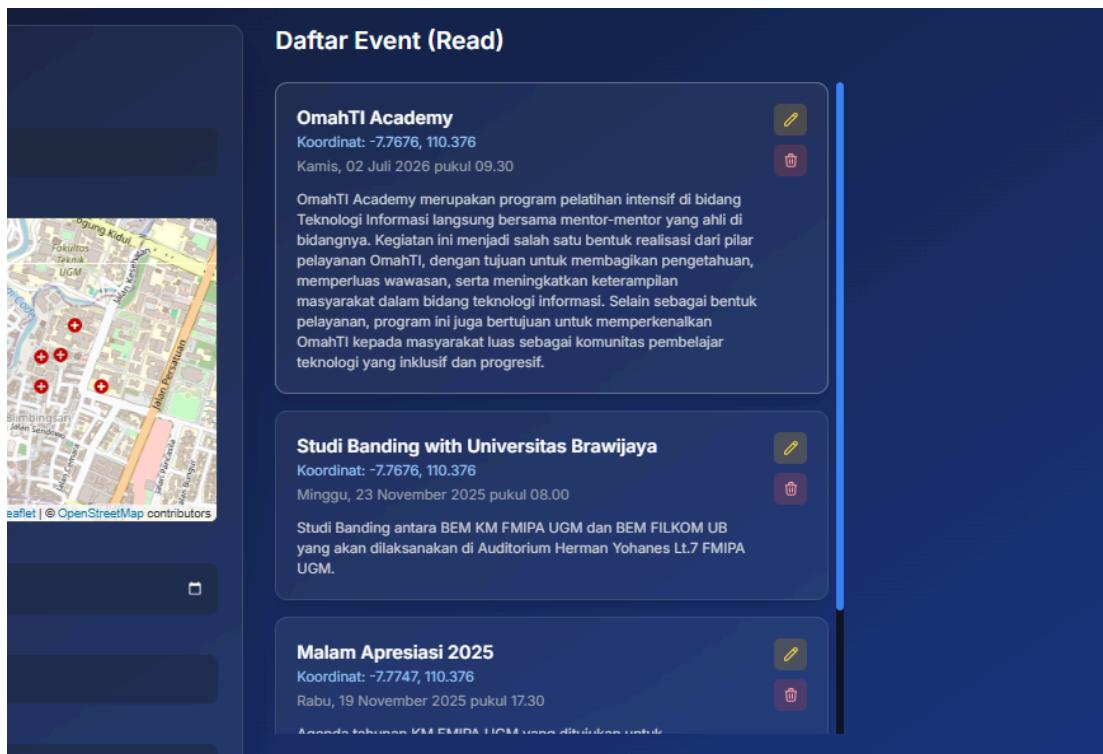
**Daftar Event (Read) (Right Panel):**

- Studi Banding with Universitas Brav**  
Koordinat: -7.7676, 110.376  
Minggu, 23 November 2025 pukul 08.00  
Studi Banding antara BEM KM FMIPA UGM dan akan dilaksanakan di Auditorium Herman Yohanes
- Malam Apresiasi 2025**  
Koordinat: -7.7747, 110.376  
Rabu, 19 November 2025 pukul 17.30  
Agenda tahunan KM FMIPA UGM yang ditujukan pencapaian di bidang akademik dan non-akademik panggung penghargaan sekaligus perayaan atas kontribusi seluruh civitas FMIPA UGM. Rangka meliputi pentas seni oleh mahasiswa, penampang serta penganugerahan apresiasi bagi sivitas FMIPA atas pencapaian akademik maupun non-akademik.

- Penjelasan Gambar:** Data event baru sedang diisi.
- Alur:** JavaScript di frontend akan mengirim data ini menggunakan fetch dengan method POST ke backend.js. Backend kemudian menjalankan perintah INSERT INTO Events ... ke database. Data yang baru dibuat akan langsung muncul di panel kanan.

## C. Operasi READ (Membaca Event)

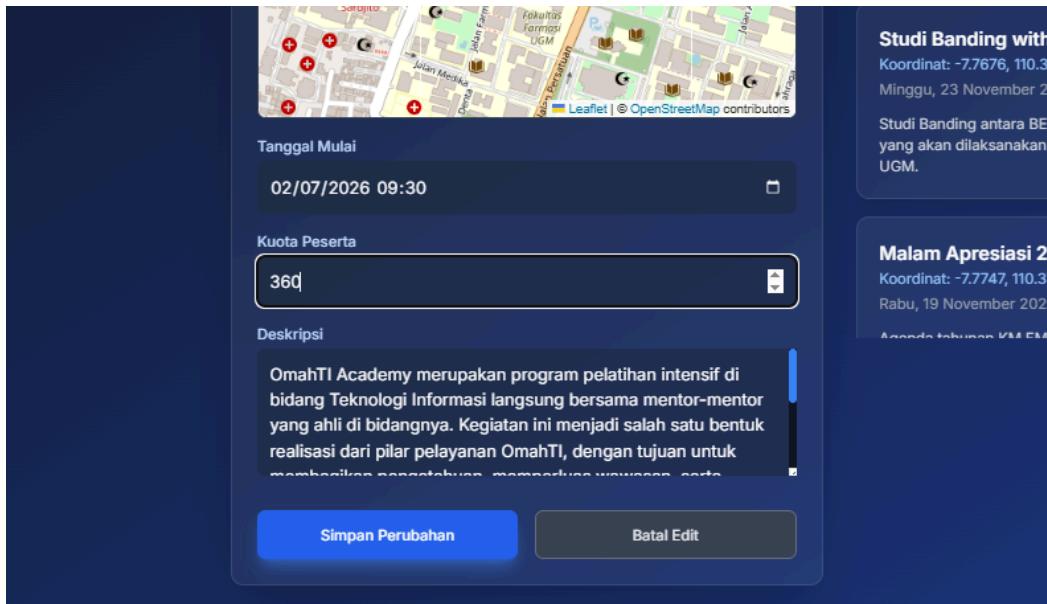
Saat halaman frontend.html pertama kali dibuka, JavaScript otomatis memanggil fungsi loadEvents().



- **Penjelasan Gambar:** Data event berhasil ditampilkan di panel kanan.
- **Alur:** Fungsi loadEvents() melakukan fetch dengan method GET ke backend.js. Backend menjalankan SELECT \* FROM Events, dan data yang diterima dari database ditampilkan sebagai daftar di UI.

## D. Operasi UPDATE (Mengedit Event)

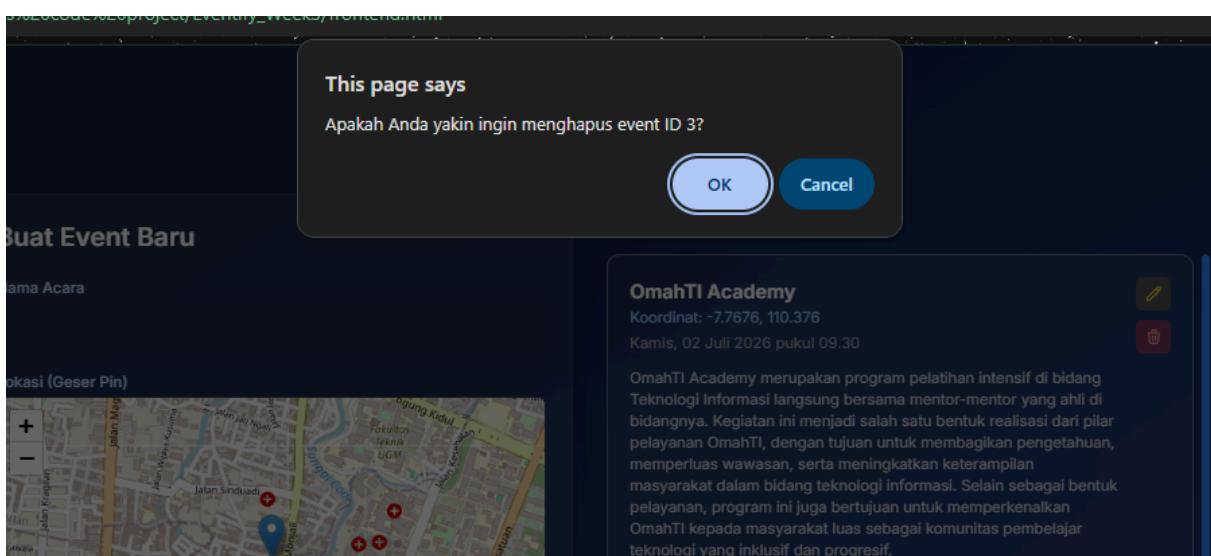
Saat pengguna mengklik tombol "Edit" (ikon pensil), data event tersebut akan dimuat kembali ke formulir di sebelah kiri.



- **Penjelasan Gambar:** Formulir otomatis terisi dengan data event yang dipilih, dan tombol berubah menjadi "Simpan Perubahan".
- **Alur:** Setelah diubah dan disubmit, JavaScript mengirim data dengan method PUT. Backend menjalankan UPDATE Events SET ... WHERE event\_id = ?. Daftar di sebelah kanan akan otomatis me-refresh data yang sudah ter-update.

## E. Operasi DELETE (Menghapus Event)

Saat pengguna mengklik tombol "Hapus" (ikon tong sampah), sebuah *pop-up* konfirmasi akan muncul.



- **Penjelasan Gambar:** Konfirmasi sebelum data dihapus permanen.
- **Alur:** Jika dikonfirmasi, JavaScript mengirim permintaan dengan method DELETE ke backend. Backend menjalankan DELETE FROM Events WHERE event\_id = ?. Data akan hilang dari daftar.

## IV. PENUTUP (Kesimpulan)

Pada Minggu 3 ini, kami telah berhasil membangun aplikasi web 3-tier fungsional dari nol. Kami telah membuktikan bahwa arsitektur backend Node.js dan frontend HTML/JS kami dapat terhubung dan melakukan empat operasi CRUD (Create, Read, Update, Delete) ke database MySQL (eventify\_db) yang kami rancang di Week 2.

Struktur ini sudah siap untuk dikembangkan lebih lanjut di Week 4, yaitu untuk mengintegrasikan tabel-tabel lain seperti Users (untuk login) serta Organizers dan Categories (untuk data *dropdown* di formulir).

# LAMPIRAN: Petunjuk Menjalankan Demo

Berikut adalah langkah-langkah ringkas untuk menjalankan demo ini di komputer lokal.

## 1. Prasyarat

- **XAMPP:** Pastikan Apache dan MySQL sedang berjalan.
- **Node.js:** Pastikan Node.js (termasuk npm) sudah terinstal di komputer.

## 2. Langkah 1: Setup Database

1. Buka phpMyAdmin (<http://localhost/phpmyadmin>).
2. Buat database baru dengan nama eventify\_db.
3. Klik database eventify\_db, lalu pilih tab "**Import**".
4. Pilih file schema\_eventify.sql (dari Week 2) dan klik "Go".  
(Pastikan 6 tabel sudah muncul: Users, Events, Organizers, dll).

## 3. Langkah 2: Setup Backend (Server)

1. Buka folder proyek (Eventify\_Week3) di Terminal/CMD.
2. Install semua library yang dibutuhkan (hanya sekali):  
`npm install`
3. Buka file backend.js dan **edit password database** (di baris 23) sesuaikan dengan XAMPP Anda (biasanya password: ").
4. Jalankan server:  
`node backend.js`
5. Biarkan terminal ini tetap berjalan (akan muncul pesan Server backend "Eventify" berjalan...).

## 3. Langkah 3: Menjalankan Frontend (Demo)

1. **Buka terminal BARU** (jangan tutup terminal backend).
2. Buka file frontend.html langsung di browser Anda (klik dua kali filenya).
3. Aplikasi CRUD siap didemokan.