**TUGAS**

**PRAKTIKUM PEMOGRAMAN JARIGAN**



**OLEH:**

**Reno Nilam Sari**

**22343068**

**Dosen Pengampu:**

**Randi Proska Sandra, S.Pd, M.Sc**

**PRODI INFORMATIKA**

**JURUSAN ELEKTRONIKA  
FAKULTAS TEKNIK**

**UNIVERSITAS NEGERI PADANG**   
**2024**

**JavaScript Engine, V8 dan Perbedaan Antara Node.js dan JavaScript Engine pada Browser Google Chrome**

**JavaScript Engine** adalah perangkat lunak yang bertugas mengeksekusi kode JavaScript. Engine ini menerjemahkan JavaScript yang berbentuk kode sumber menjadi kode mesin yang dapat dipahami oleh komputer. JavaScript Engine biasanya digunakan dalam browser untuk menjalankan script di halaman web, tetapi juga dapat digunakan di lingkungan non-browser seperti Node.js.

Salah satu contoh JavaScript Engine yang paling terkenal adalah **V8**, yang dibuat oleh Google.

**V8** adalah JavaScript Engine open-source yang dikembangkan oleh Google, ditulis dalam C++.Digunakan di browser **Google Chrome** untuk menjalankan JavaScript dengan performa tinggi.Memiliki dua komponen utama:

* 1. **Parser dan Compiler**: Menerjemahkan kode JavaScript menjadi bytecode atau kode mesin.
  2. **Garbage Collector**: Mengelola memori dengan membersihkan objek-objek yang tidak digunakan.

V8 juga digunakan di lingkungan **Node.js**, sehingga memungkinkan pengembang menggunakan JavaScript di luar browser.

**Node.js vs JavaScript Engine pada Browser Google Chrome**

**Persamaan**

1. **Menggunakan V8**: Baik Node.js maupun browser Google Chrome menggunakan V8 untuk mengeksekusi kode JavaScript.
2. **Eksekusi JavaScript**: Keduanya mampu menjalankan JavaScript dengan cepat dan efisien.

**Perbedaan**

| **Aspek** | **Node.js** | **JavaScript Engine pada Chrome** |
| --- | --- | --- |
| **Tujuan Utama** | Untuk menjalankan JavaScript di server dan aplikasi backend. | Untuk menjalankan JavaScript di lingkungan browser. |
| **API yang Disediakan** | Menyediakan API berbasis server, seperti fs (file system), HTTP, dll. | Menyediakan API berbasis DOM dan browser, seperti document, window. |
| **Lingkungan Eksekusi** | Tidak memiliki DOM atau elemen visual. | Mendukung manipulasi elemen DOM dan interaksi dengan pengguna. |
| **Modul Tambahan** | Menggunakan modul CommonJS atau ES Module (require, import). | Tidak memiliki akses ke modul backend seperti file system. |
| **Event Loop** | Event loop pada Node.js dioptimalkan untuk server-side (I/O intensive). | Event loop pada browser dirancang untuk mengelola rendering dan user interaction. |