

BAB I PENDAHULUAN

1.1 Latar Belakang Masalah

Perkembangan teknologi Informasi saat ini berkembang sangat pesat, sehingga transformasi teknologi menjadi lebih canggih dan cepat. Kebutuhan teknologi saat ini yang sangat membantu masyarakat berupa aplikasi dan website yang berguna dalam menunjang perkembangan usaha dan aktivitas keseharian pada manusia di bidang digital, tentunya kecepatan suatu aplikasi secara *real-time* sangat dibutuhkan. Dari hal ini diperlukan Bahasa pemrograman yang tepat untuk membantu programmer dalam mempercepat pekerjaan, terdapat suatu proses pekerjaan dalam menjalankan aplikasi dan website yaitu metode *synchronous* merupakan proses mengeksekusi kode sesuai urutan kode seperti yang tertulis, metode *asynchronous* merupakan proses mengeksekusi kode yang tidak dalam urutan yang ada atau dapat dipanggil untuk menjalankan perintah berikutnya tanpa menunggu perintah sebelumnya selesai .

Penggunaan bahasa pemrograman PHP berbasis *server-side* yang sering digunakan developer membangun aplikasi menggunakan PHP *Native* dalam melakukan pembuatan program kerangka kerja sendiri dengan membangun aplikasi website yang dinamis. program yang ditulis pada PHP tujuan utamanya adalah untuk menjalankan proses sederhana terdiri dari banyak tugas yang dimana harus dijalankan secara berurutan seperti pemrosesan data yang disebut metode pengiriman data secara *synchronous*. Hal ini kurang efektif dalam melakukan pekerjaan karena memakan waktu cukup lama. Para developer melakukan berbagai cara dalam mempercepat pekerjaannya dengan menggunakan bahasa pemrograman berbasis *client-side* seperti Javascript yang juga sering digunakan developer dalam mempercepat melakukan pembuatan aplikasi. Bahasa pemrograman Javascript berjalan disisi *client*/browser saja, Node.JS hadir untuk melengkapi peran Javascript, sehingga dapat juga digunakan sebagai bahasa pemrograman yang berjalan di sisi *server* (Lutfi, 2017).

Node.JS menjadi platform yang mendukung Bahasa pemrograman Javascript dengan teknik *asynchronous*, Penanganan *web service* oleh *server side* dengan PHP masih kurang mendukung proses data yang cepat. Node.js hadir sebagai *server-side* yang dapat digunakan oleh bahasa pemrograman Javascript. Cara kerja dari Node.JS dengan pendekatan *asynchronous* yang merupakan hasil dari eksekusi atau *output*, tidak berdasarkan urutan kode, tetapi dengan waktu dalam pemrosesan. Eksekusi *asynchronous* tidak memblokir atau menunggu perintah selesai (Nababan, 2019). Hal ini tentu penggunaan Node.JS sangat dibutuhkan oleh seorang pengguna dan pengembang dalam membuat aplikasi yang mampu menangani permintaan data secara *real-time* dalam menangani permintaan data secara bersamaan.

Penggunaan Node.JS saat ini dalam pembuatan proses database sangat menyulitkan pekerjaan pengguna dikarenakan proses koneksi database, pembuatan fitur *backup* dan *restore table* pada database harus dilakukan secara *native* dan penggunaan *package* yang terpisah yang diharuskan untuk download secara manual melalui *command line* dalam menjalankan proses pembuatan database pada Node.JS, tentu hal ini tidak efektif dalam mempercepat pekerjaan pengguna dalam pembuatan proses database. Dari hal ini dalam memanfaatkan penggunaan Node.JS kami melakukan perancangan dan penambahan fitur pada *framework* Aurora.JS dalam mempercepat pembuatan proses database.

Framework Aurora.JS merupakan rancangan inovasi mahasiswa jurusan Sistem Informasi pada Universitas Pembangunan Jaya. *Framework* Aurora.JS diciptakan untuk mempermudah pengguna dan pengembang dalam merancang aplikasi di Node.JS dan adanya *framework* Aurora.JS pembuatan proses database dapat dilakukan secara cepat, karena tidak perlu melakukan install *package* dan pembuatan proses tidak dilakukan secara *native* dalam pembuatan proses database karena *framework* Aurora.JS sudah tersedia dalam melakukan koneksi database dan pembuatan schema database, hal ini sangat membantu pengembang dan pengguna dalam mempercepat pekerjaannya.

Namun, *framework* Aurora.JS masih memiliki kekurangan pada proses pembuatan database yaitu belum tersedianya fitur *backup* dan *restore table* pada database hal ini masih menjadi kendala dalam melakukan perancangan database yang dilakukan oleh seorang pengguna dan pengembang sehingga bila terjadi kesalahan oleh seorang pengguna dalam melakukan perancangan database maka pengguna harus melakukan secara manual. Dengan adanya fitur ini mampu menyempurnakan kekurangan-kekurangan yang terjadi di *framework* Aurora.JS.

Adanya rencana pengembangan fitur ini pada teknologi yang diterapkan di Node.JS tentu sangat membantu pengguna dan pengembang untuk melakukan perancangan dan memudahkan dalam membangun aplikasi. Namun penggunaan Node.JS saat ini masih belum banyak dibanding penggunaan *framework* PHP, karena *framework* yang mendukung aplikasi berbasis Node.JS masih sedikit dan jika dilakukan secara *native* akan menyulitkan pengguna. Dari hal tersebut agar dapat teratasi permasalahan yang dihadapi pengguna dan pengembang dengan cara pengembangan *framework* bernama Aurora.JS berbasis pada Node.JS yang akan dibuat oleh penulis. Pengembangan *framework* dibagi menjadi beberapa modul yaitu untuk manipulasi data yang mencakup fungsi *Create, Read, Update, Delete* (CRUD) dan modul database. Salah satu modul yang dikembangkan oleh penulis pada *framework* Aurora.JS yaitu modul *backup* dan *restore table* dalam mempermudah proses database. Pengembangan modul pada *framework* Aurora.JS yang dirancang penulis dijadikan sebagai bahan untuk penelitian penulis yang berjudul **“PENGEMBANGAN FITUR BACKUP DAN RESTORE DALAM MEMPERMUDAH PROSES DATABASE PADA FRAMEWORK AURORA.JS BERBASIS NODE.JS”**.

1.2 Rumusan/Identifikasi Masalah

Berikut uraian pada latar belakang, dapat diidentifikasi yang terjadi pada permasalahan tersebut, maka penulis mengidentifikasi masalah sebagai berikut :

1. *Framework* Aurora.JS tidak memiliki fitur *backup* dan *restore table* pada database.
2. Terjadinya kesalahan pembuatan database yang terjadi karena kesalahan pemrograman dan ketidaksengajaan dalam melakukan pekerjaan secara tim oleh pengguna dan pengembang, sehingga diperlukan penambahan fitur *backup* dan *restore table* di database pada *framework* Aurora.JS

Berdasarkan beberapa uraian pada diidentifikasi masalah, penulis dapat merumuskan permasalahan sebagai berikut:

1. Bagaimana cara pengguna dan pengembang dapat melakukan *backup* dan *restore* data pada *table* tanpa melakukan penerapan akses database?
2. Bagaimana cara pengguna dan pengembang dapat melakukan *backup* dan *restore table* secara *single table* dan *multiple table* database pada *framework* Aurora.JS?
3. Bagaimana cara pengguna bisa lebih fleksibel dalam berbagi data *dummy* pada sekelompok tim pengguna dan pengembang tanpa meminta data *master* dari pihak pertama?

1.3 Ruang Lingkup dan Batasan Masalah

Berdasarkan rumusan masalah dapat disimpulkan masalah yang dibatasi pada penulis mengenai perancangan pengembangan fitur *backup* dan *restore* data tabel pada database. Fungsi ini bertujuan agar penerapan pembuatan data di SQL yaitu MySQL. Pembuatan fungsi ini untuk bertujuan untuk melengkapi fitur pada *framework* aurora.js. Penulis memilih MySQL sebagai dasar pengembangan tahap awal dikarenakan MySQL sudah banyak digunakan oleh pengguna dan pengembang dalam merancang aplikasi.

1.4 Maksud dan Tujuan Penelitian

1.4.1 Maksud Penelitian

Maksud dari penelitian ini adalah untuk melengkapi perancangan *framework* aurora.JS yang berbasis pada Node.JS dengan

menyediakan modul pada fitur *backup* dan *restore table* dalam mempermudah proses database. Pengembangan *framework* Aurora.JS dapat dimanfaatkan sebagai hasil karya mahasiswa dengan tujuan untuk memotivasi dan meningkatkan kreatifitas yang lebih bermanfaat pada mahasiswa Sistem Informasi Universitas Pembangunan Jaya.

1.4.2 Tujuan Penelitian

tujuan dari penelitian ini adalah:

1. Menambah pilihan *framework* Aurora.JS yang mendukung Node.JS dan memiliki fitur *backup* dan *restore* dalam mempermudah proses pembuatan database untuk para pengguna dan pengembang.
2. Mempercepat waktu pengembangan aplikasi Aurora.JS berbasis Node.js, terutama dalam berbagi data dengan memanfaatkan fitur yang sedang dirancang yaitu *backup* dan *restore* database.
3. Menyediakan *framework* yang dapat dijadikan sebagai produk yang terus berkembang untuk program studi Sistem Informasi UPJ.

1.5 Manfaat Penelitian

Berikut Manfaat dari penelitian yang dilakukan penulis yaitu sebagai berikut:

a. Bagi Penulis

- Dapat mengembangkan dan memahami pengetahuan dalam merancang inovasi yang dikerjakan, sebelumnya oleh mahasiswa prodi sistem informasi.
- Menambah wawasan mengenai algoritma pemrograman dan pada *client-side programming*.
- Mempercepat proses pekerjaan dengan menggunakan *framework* Aurora.JS dalam merancang program yang diciptakan oleh seorang pengguna dan pengembang.

b. Bagi Pelajar

- Memperoleh kemudahan dalam mengerjakan tugas-tugas pemrograman yang diberikan dengan menggunakan *Framework* Aurora.JS
- Meningkatkan kualitas ilmu dan pengetahuan mahasiswa dengan mempelajari pemrograman pada *Client-side programming*.

c. Bagi Universitas Pembangunan Jaya

- Munculnya inovasi-inovasi yang berkembang pada mahasiswa dengan menerapkan ide dalam perancangan program di dalam *framework* Aurora.JS
- Meningkatkan kualitas mahasiswa yang terampil dalam berinovasi dari penelitian yang dirancang penulis selama menempuh ilmu dan pembelajaran di Universitas Pembangunan Jaya.

1.6 Sistematika Penulisan

Untuk memahami lebih sistematis laporan penelitian ini, penulis membagi materi menjadi 5 bab, dan penulis menyusunnya secara sistematis sebagai berikut:

BAB I PENDAHULUAN

Pada bab ini mengenai gambaran umum mengenai penelitian yang dirancang pada penulis. Bab ini menjelaskan mengenai latar belakang, perumusan masalah, ruang lingkup & batasan masalah, maksud dan tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini membahas tentang teori yang relevan berupa pengertian dan definisi dengan apa yang akan dijelaskan pada penelitian. Pada bab ini berisi teori dasar dari beberapa kutipan buku atau sumber yang digunakan untuk

dijadikan sebagai referensi serta tinjauan studi yang membahas teori-teori hasil penelitian.

BAB III METODE PENELITIAN

Bab ini membahas tentang analisa sistem yang berjalan pada pembuatan modul *backup* dan *restore table* pada database dengan *framework* Aurora.JS berbasis Node.JS. Analisa yang dilakukan pada bab ini akan menghasilkan informasi dalam membantu kebutuhan pengguna dan melengkapi modul yang kurang pada *framework* Aurora.JS berbasis Node.JS.

BAB IV HASIL DAN ANALISIS PENELITIAN

Bab ini membahas tentang hasil dari penelitian yang dijelaskan dalam bentuk narasi yang menggambarkan penelitian secara lengkap, dan juga perancangan sistem yang diusulkan dalam penelitian ini akan dibahas. Perancangan tersebut berupa analisa perancangan sistem, rancangan sistem usulan, UML, dan cara penggunaan fungsi yang telah disediakan pada modul backup dan restore database.

BAB V PENUTUP

Pada bab ini menjelaskan tentang kesimpulan, saran dan pengembangan lebih lanjut yang mencakup keseluruhan penulisan tugas akhir.

DAFTAR PUSTAKA

Bagian ini memuat seluruh referensi yang digunakan penulis untuk tugas akhir ini.

LAMPIRAN

Bagian ini berisi dokumen-dokumen tambahan yang dilampirkan untuk melengkapi penelitian utama dan penulisan tugas akhir.

UNIVERSITAS
PEMBANGUNAN
JAYA

