

BAB I

PENDAHULUAN

1.1. Latar Belakang

Platform web yang pada mulanya hanya ditujukan untuk memudahkan pengiriman dan menampilkan dokumen kini telah berevolusi hingga menjadi *platform* yang cukup mumpuni untuk digunakan sebagai *platform* utama berjalannya perangkat lunak. Dewasa ini banyak aplikasi yang dapat dengan mudah diakses oleh pengguna hanya dengan membuka peramban mereka. Mulai dari aplikasi desain seperti Figma, aplikasi pembuatan dokumen seperti Google Docs dan Microsoft Office Online, aplikasi manipulasi gambar seperti Canva dan PhotoPea, aplikasi konferensi video seperti WebEx dan Google Meet, hingga gim sekalipun.

Dengan makin mumpuninya *platform web* ini, sebagian peneliti telah memprediksi bahwa akan makin banyak aplikasi yang beralih ke *platform web* kedepannya dikarenakan beberapa keunggulan yang tidak dimiliki *platform desktop* (Taivalaari et al., 2011), yaitu tidak memerlukan instalasi dan/atau peningkatan versi secara manual dari sisi pengguna, *pen-deploy-an* versi baru yang lebih mudah dengan cakupan pengguna yang lebih luas, serta format aplikasi terbuka yang memudahkan konten-konten yang tersedia di web untuk dapat dibaca oleh manusia dan cukup mudah untuk diproses oleh mesin, sehingga memicu berkembangnya jenis aplikasi baru yang mengagregasi data dari berbagai situs web.

Kendati demikian, masih terdapat sebagian aplikasi yang masih didistribusikan melalui *platform desktop* saja. Salah satunya adalah RAPTOR (<https://raptor.martincarlisle.com/>). Aplikasi tersebut digunakan untuk memvisualisasikan jalannya suatu diagram alir sehingga memudahkan untuk mengetahui apakah suatu diagram alir yang dibuat telah sesuai atau tidak. RAPTOR juga digunakan dalam pengajaran mata kuliah Algoritma di Fakultas Ilmu Komputer Universitas Pembangunan Nasional “Veteran” Jawa Timur.

Tetapi dikarenakan RAPTOR hanya ditujukan untuk *platform desktop* dan tidak memiliki kelebihan-kelebihan aplikasi berbasis web yang telah kami sebutkan sebelumnya. Salah satu limitasi yang dimiliki RAPTOR adalah

meskipun telah tersedia versi RAPTOR untuk sistem operasi Linux dan Mac yang disebut RAPTOR Avalonia Edition, terdapat keterbatasan pada RAPTOR versi ini, seperti: fitur yang tidak selengkap RAPTOR versi Windows serta tidak kompatibelnya berkas diagram alir RAPTOR Windows dengan RAPTOR Avalonia Edition. Selain itu, RAPTOR juga tidak dilengkapi dengan beberapa fitur modern yang memudahkan penggunaan program tersebut, seperti penyimpanan *cloud*, kemudahan pengguna untuk membuat lebih dari satu *workspace*, dan kemudahan untuk membagikan *workspace* pada orang lain.

Dengan limitasi-limitasi di atas, penulis merasa diperlukan sebuah aplikasi visualisasi diagram alir alternatif yang lebih mudah diakses di mana pengguna hanya perlu menginstal peramban internet untuk dapat mengaksesnya tanpa diharuskan menginstal komponen atau perangkat lunak tambahan apapun serta tanpa terhalang sistem operasi yang digunakan pengguna. Selain itu, sistem tersebut harus cukup fleksibel untuk memvisualisasikan diagram alir pengguna serta memfasilitasi ekspresi pemrograman yang diinputkan pengguna. Untuk memenuhi kedua ekspektasi tersebut, digunakan HTML 5 Canvas yang merupakan teknologi yang tersedia pada platform web yang memang ditujukan untuk keperluan penggambaran elemen-elemen grafis (Mozilla Developer Network, 2023). Selain itu, diterapkan Algoritma *Recursive Descent Parsing* untuk pemrosesan teks ekspresi pemrograman yang diinputkan pengguna. Sifat algoritma ini yang tidak terlalu sulit untuk diimplementasikan tetapi tetap ampuh cocok untuk diterapkan pada penelitian ini karena sistem baru ini tidak memiliki kebutuhan untuk memproses ekspresi pemrograman yang kompleks.

1.2. Rumusan Masalah

Dari latar belakang di atas, didapati rumusan masalah yang merupakan dasar dilakukannya penelitian ini:

1. Bagaimanakah cara membuat sistem visualisasi diagram alir yang lebih mudah diakses daripada solusi yang sudah ada?
2. Bagaimana pengguna RAPTOR melihat sistem baru ini, apakah Dializer berhasil memenuhi tujuannya dengan akses yang lebih mudah dengan fungsionalitas yang memadai?

1.3. Tujuan Penelitian

Adapun tujuan yang hendak dicapai oleh penelitian ini adalah sebagai berikut: Membangun sistem visualisasi diagram alir yang lebih mudah diakses daripada RAPTOR dengan fungsionalitas yang cukup memadai untuk keperluan visualisasi diagram alir pengguna;

1.4. Manfaat Penelitian

Penelitian ini memiliki beberapa manfaat untuk berbagai pihak, yaitu:

1. Tersedianya referensi penelitian tambahan terkait pengembangan aplikasi berbasis web yang memanfaatkan HTML Canvas;
2. Tersedianya referensi implementasi sistem visualisasi diagram alir yang dapat dipelajari dan dikembangkan oleh siapapun dikarenakan sifat kode sumbernya yang terbuka;
3. Tersedianya alternatif aplikasi visualisasi diagram alir yang lebih mudah diakses dan digunakan dibandingkan sistem yang sudah ada dan juga disertai dengan beberapa fitur modern yang memudahkan penggunaannya.

1.5. Batasan Masalah

Untuk menghindari pembahasan yang terlalu luas, berikut adalah batasan-batasan yang ditetapkan oleh penulis dalam penelitian ini:

1. Aplikasi terdiri atas 2 bagian, yaitu bagian *backend* dan *frontend*;
2. *Stack* teknologi yang dipilih adalah sebagai berikut:
 - *Frontend*: React.js;
 - *Backend*: Nest.js;
 - Basis data: PostgreSQL.
3. Kode sumber aplikasi bersifat terbuka dan dapat diakses melalui *platform* GitHub;
4. Hanya terdapat 7 jenis *node* yang disediakan untuk digunakan dalam pembuatan *flowchart*: *start*, *end*, *process*, *branching*, *loop*, *input*, serta *output*.