



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk mengubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

JavaScript merupakan teknologi pemrograman paling populer peringkat pertama, dan diikuti dengan HTML/ CSS pada peringkat kedua. Selama tujuh tahun berturut-turut, JavaScript menjadi bahasa pemrograman yang paling umum digunakan (“Developer Survey Result”, 2019). JavaScript biasanya dipasangkan dengan HTML dan CSS untuk membuat *website* maupun aplikasi *web*. *Web Developer* adalah jenis pekerjaan yang selalu dibutuhkan dari tahun ke tahun di era teknologi ini. Apapun jenis bisnisnya ketika ingin memiliki nama berskala nasional maupun internasional membutuhkan minimal sebuah *website* untuk periklanan, sarana komunikasi antara klien dan pemilik, *web app* untuk melakukan transaksi daring, atau sekedar *company profiles* untuk menunjukkan profesionalitas perusahaan. JavaScript, HTML, dan CSS adalah teknologi pemrograman paling dasar dalam pembuatan *web* maupun *web app*. Namun, tidak jarang ditemukan sebuah *web* yang dibentuk hanya menggunakan JavaScript. Setiap elemen di dokumen seperti *head*, *tables*, *table headers*, maupun isi *table* merupakan bagian dari *Document Object Model* (DOM) untuk sebuah dokumen. Oleh karena itu, semua elemen tersebut dapat diakses dan dimanipulasi menggunakan DOM dan sebuah *scripting language* seperti JavaScript (MDN Web Docs Mozilla, 2019).

JavaScript memiliki kendali penuh dalam *web*, contoh kendalinya: JavaScript dapat mengubah semua elemen HTML pada halaman, JavaScript dapat mengubah semua atribut HTML pada halaman, JavaScript dapat mengubah semua CSS *styles* pada halaman *web*, JavaScript dapat menghapus elemen dan atribut HTML yang ada, JavaScript bisa menambah elemen dan atribut HTML baru, JavaScript dapat bereaksi terhadap seluruh HTML *events* di dalam halaman *web*, dan JavaScript dapat membuat HTML *events* baru di dalam halaman *web* (w3schools, 2019). JavaScript adalah bahasa pemrograman yang penting karena JavaScript merupakan bahasa dari *web browser*. Hubungannya dengan *browser* membuat JavaScript menjadi salah satu bahasa pemrograman paling populer di dunia. Web telah menjadi *platform* penting untuk pengembangan aplikasi, dan JavaScript merupakan satu-satunya bahasa yang ditemukan dalam semua *browser*. Selain kekurangannya, JavaScript sangatlah ringan dalam performa, tetapi agar dapat menggunakan JavaScript dengan baik, keterbatasan-keterbatasannya perlu diketahui secara jelas (Crockford, 2008). *Client-side* JavaScript berisi beberapa fitur yang membuatnya terlepas dari bahasa pemrograman lainnya. Pertama, JavaScript dieksekusi dibawah model *asynchronous*. Ini memungkinkan *event handlers* tereksekusi sesuai permintaan, saat pengguna berinteraksi dengan komponen aplikasi *web*. Kedua, kebanyakan bagian dari JavaScript dirancang untuk berinteraksi dengan entitas eksternal yang dikenal sebagai Document Object Model (DOM). Menggunakan DOM, JavaScript dapat digunakan untuk mengakses atau memanipulasi komponen-komponen yang tersimpan di dalam DOM sehingga memungkinkan halaman *web* berubah tanpa memerlukan sebuah *page reload*.

Meskipun fitur tersebut memungkinkan sebuah aplikasi *web* menjadi lebih interaktif, hal tersebut juga memperkenalkan kesalahan pada kode JavaScript. Mayoritas (65%) dari kesalahan kode JavaScript berhubungan dengan DOM, artinya *error* dan *exception* yang didapat ketika mengembangkan suatu web berasal dari *programmer* yang salah dalam melakukan interaksi dengan DOM (Ocariza, et al., 2013).

Oleh karena masalah yang ditemukan, diperlukan sebuah aplikasi pembelajaran untuk membantu pemula yang telah memahami dasar HTML, CSS, dan JavaScript agar dapat membuat *web* yang interaktif dan mengurangi kebingungan atau kesalahan pengembang yang belum fasih dalam pengembangan aplikasi *web* ketika berinteraksi dengan Document Object Model. Sudah ada penelitian tentang gamifikasi pembelajaran bahasa pemrograman Java (Beltran, et al., 2016). Sudah ada penelitian gamifikasi pembelajaran tentang konsep pemrograman (Elshiekh & Butgerit, 2017). Namun, belum ditemukan aplikasi metode gamifikasi tentang pembelajaran JavaScript HTML Document Object Model sehingga hal ini sangat cocok untuk dijadikan topik penelitian.

Di tahun 2015, Yu Kai Chou mencetuskan sebuah framework gamifikasi yang dapat mengakomodasi pembuatan aplikasi gamifikasi yang bernama Octalysis. Economou, dkk. menyimpulkan,

“We analyzed the platform’s gamified features using a small number of expert evaluators against Octalysis gamified framework and we encouraging review of a fairly balanced simulations that can lead to the creation of motivating and engaging experiences.” (Economou, dkk. 2015)

Oleh karena itu, aplikasi dengan *framework* Octalysis dinilai dapat menciptakan pengalaman yang memotivasi dan menarik bagi para pengguna. Membuat Octalysis cocok untuk dijadikan *framework* pengembangan aplikasi pembelajaran JavaScript HTML Document Object Model.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah dijelaskan, rumusan masalahnya adalah sebagai berikut:

1. Bagaimana merancang dan membangun DominateMyJS sebagai pembelajaran JavaScript HTML DOM?
2. Bagaimana tingkat *Behavioral Intention to Use* dalam menggunakan aplikasi dan *immersion* pengguna pada aplikasi pembelajaran JavaScript HTML DOM dengan *framework* octalysis yang didasarkan dengan Hedonic-Motivation System Adoption Model?

1.3 Batasan Masalah

Adapun batasan masalah untuk penelitian ini:

1. JavaScript DOM yang dimaksud hanya untuk HTML,
2. DOM yang dimaksud ada delapan buah *method*, yaitu: *getElementById*, *createElement*, *appendChild*, *removeChild*, *replaceChild*, *cloneNode*, *insertBefore*, *setAttribute*, *addEventListener*, dan *removeEventListener*,
3. *Platform* aplikasi yang digunakan adalah *web*,
4. Target pengguna adalah sivitas akademika Universitas Multimedia Nusantara,
5. Definisi kesuksesannya menggunakan *Hedonic-Motivation System Adoption Model* (HMSAM) (contoh: mengukur tingkat kebahagiaan pengguna aplikasi atau tingkat kegunaan aplikasi dalam menambah wawasan pengguna),
6. Gamifikasi digunakan untuk mendorong motivasi pengguna dalam pembelajaran JavaScript HTML Document Object Model.

1.4 Tujuan Penelitian

Berdasarkan rumusan masalah, penelitian ini bertujuan untuk merancang dan membangun sistem DominateMyJS sebagai pembelajaran JavaScript HTML DOM. Selain itu, penelitian ini juga bertujuan untuk mengukur tingkat *Behavioral Intention to Use* dalam menggunakan aplikasi dan *immersion* pengguna pada aplikasi gamifikasi pembelajaran JavaScript HTML DOM dengan framework Octalysis yang didasarkan dengan Hedonic-Motivation System Adoption Model.

1.5 Manfaat Penelitian

Manfaat yang diharapkan dari perancangan dan pembangunan aplikasi DominateMyJS untuk pembelajaran JavaScript HTML DOM dengan metode Octalysis adalah aplikasi dapat dipakai oleh mahasiswa di Universitas Multimedia Nusantara yang baru mempelajari mata kuliah Pengantar Teknologi Internet. Selain itu, aplikasi ini juga diharapkan dapat membawa manfaat sebagai sarana pembelajaran yang lebih mudah dan menarik bagi pemula yang ingin memahami tentang pembuatan *dynamic web* menggunakan JavaScript HTML DOM. Selain itu, penelitian ini dapat menambah referensi dalam penerapan metode gamifikasi Octalysis pada aplikasi pembelajaran JavaScript HTML DOM.

1.6 Sistematika Penulisan

Sistematika penulisan laporan skripsi ini terdiri dari lima buah bab seperti berikut.

1. BAB I PENDAHULUAN

Bab ini terdiri dari latar belakang masalah, rumusan masalah, Batasan masalah, tujuan penelitian, manfaat penelitian, dan sistematika penulisan.

2. BAB II LANDASAN TEORI

Bab ini terdiri dari landasan teori yang diperlukan dalam melakukan penelitian seperti Gamifikasi, Octalysis, atau Document Object Model.

3. BAB III METODOLOGI DAN PERANCANGAN APLIKASI

Bab ini terdiri dari metodologi penelitian dan dokumentasi perancangan aplikasi yang diperlukan dalam melakukan penelitian.

4. BAB IV IMPLEMENTASI DAN UJI COBA

Bab ini terdiri dari dokumentasi hasil implementasi dan pengujian aplikasi yang telah dirancang.

5. BAB V SIMPULAN DAN SARAN

Bab ini terdiri dari simpulan hasil penelitian serta saran pengembangan aplikasi yang dapat dilakukan.