

**Hak Cipta Dilindungi Undang-Undang**

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Antarmuka merupakan salah satu komponen penting dalam sebuah perangkat lunak, antarmuka merupakan jembatan utama antara pengguna dengan perangkat lunak untuk berinteraksi. Dalam membangun sebuah perangkat lunak berbasis *web*, tentunya akan dilakukan tahap pembuatan antarmuka, yaitu seorang *User Interface (UI) Programmer* (selanjutnya disebut *UI Programmer*) akan menyajikan data yang disimpan di dalam basis data ke dalam bentuk antarmuka yang dilihat oleh pengguna.

Dalam konteks fungsi dalam manajemen basis data, konsep CRUD (*Create, Read, Update dan Delete*) digunakan untuk menggambarkan fungsi dari tempat penyimpanan (Martin, 1983, Dikutip Oleh Ismail dkk, 2015), maka pada antarmuka perangkat lunak yang dibangun hendaknya memiliki proses-proses dasar tersebut. *UI Programmer* harus menerjemahkan elemen-elemen dalam sebuah skema basis data ke dalam kumpulan *Tag HTML (Hypertext Markup Language)* yang menjadi komponen dasar antarmuka perangkat lunak berbasis *web*, tentunya untuk membangun antarmuka sebuah perangkat lunak seorang *UI Programmer* harus paham dengan struktur basis data yang digunakan. Dalam membangun antarmuka pada umumnya *UI Programmer* akan melakukannya dengan mengetikkan sintaks HTML secara manual.

Formulir *web* adalah komponen dari halaman *web* yang memiliki kontrol formulir, seperti *text field, button, check box, range*, atau *color picker*. Seorang pengguna dapat berinteraksi dengan formulir yang menyediakan data kemudian dapat dikirim ke *server* untuk diproses lebih lanjut (W3C, 2016). Formulir *web* merupakan salah satu komponen penting dalam antarmuka berbasis *web*, formulir menjadi antarmuka utama untuk proses *create* dan *update* dari proses CRUD.

Jika dilihat dari fase siklus pengembangan perangkat lunak tradisional atau *Software Development Life Cycle (SDLC)* menurut Peter Rob dan Carlos Coronel



Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

(2009, hal. 377) pada awal fase implementasi, desain basis data sudah diimplementasikan ke dalam *Database Management Systems* (DBMS) dan pada fase inilah siklus pembuatan kode program dan pengujian akan terjadi. Rob dan Coronel juga menyatakan bahwa tahap implementasi dan testing ini dapat memakan waktu 50% sampai 60% dari total waktu keseluruhan pengembangan perangkat lunak dan dengan adanya generator aplikasi dapat mengurangi waktu dalam proses pembuatan kode program dan pengujian. Kemudian berdasarkan penelitian yang dilakukan oleh Sutton dan Sprague di IBM (Alan Dix, 2016, hal. 235) diperoleh hasil yang memperkirakan bahwa 50% waktu desainer dihabiskan pada proses mendesain kode untuk antarmuka.

Seiring berjalannya waktu banyak dilakukan penelitian terkait otomatisasi dalam membangkitkan formulir *web*, seperti penelitian yang dilakukan oleh Rein Raudj r v yang membahas tentang membangkitkan *web form* berdasarkan skema dinamis yang ditulis dalam bahasa *Java*. Pada penelitian ini aplikasi pembangkit *web form DynaForm* menerima masukan berupa XML (*Extensible Markup Language*) skema yang menyediakan struktur *form*. Berdasarkan masukan yang diterima oleh *DynaForm*, maka akan dihasilkan keluran berupa *web form* dengan tambahan CSS. Dalam proses pembangkitan *form* dapat diberikan aturan tambahan menggunakan *DynaData* dan XML *instance* untuk menyediakan data nilai awal dari *form* (Raudj r v, 2010).

Penelitian berikutnya adalah yang dilakukan oleh Atia Mahmud Albh bah (2013) yang membahas tentang bagaimana membangun kerangka aturan dalam pembangkitan *form*, dalam penelitian ini dibahas tentang aturan-aturan yang digunakan dalam pembangkitan *form* berdasarkan metada dari skema basis data. Skema basis data diterjemahkan ke dalam format XML sebagai pusat aturan pembangkitan antarmuka *form* atau yang disebut dengan *RuleML*, berdasarkan aturan yang di tulis ke dalam format *RuleML* maka dapat dibangun antarmuka *form* dengan tambahan *JavaScript* dan CSS serta dapat dikembalikan ke dalam bentuk skema basis data.



Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Selanjutnya adalah penelitian yang dilakukan oleh Mohammed A. Mgheder dan M. J. Ridley (2008) yang membahas tentang generasi antarmuka berbasis *web* menggunakan metadata basis data yang ditulis dalam bahasa pemrograman PHP, penelitian ini menggunakan metadata basis data sebagai dasar dalam proses generasi antarmuka yang difokuskan dalam generasi formulir *web*.

Selain penelitian banyak juga bermunculan aplikasi-aplikasi yang dibangun untuk membangkitkan formulir berbasis *web*, salah satunya adalah GII Generator yang dikeluarkan oleh Yii *Framework* khusus untuk bahasa pemrograman PHP yang disertai pembangkit antarmuka CRUD yang disertai *JavaScript* dan *CSS* di dalamnya. Untuk melakukan pembangkitan antarmuka CRUD aplikasi ini harus melakukan pembangkitan *Model* dalam konsep *Model View Controller* (MVC) sebagai representasi basis data khususnya tabel dalam bentuk kode program, dimana aturan untuk pembangkitan *form* diletakan di dalam *model* tersebut (Winesett, 2012).

Aplikasi pembangkit lainnya adalah *ScriptCase* yang merupakan alat RAD (*Rapid Application Development*) yang menghasilkan aplikasi PHP berbasis tabel basis data lengkap dengan *query*, filter, grafik, laporan PDF, *forms*, menu navigasi, *export* data ke XLS, DOC, XML dan CSV (ScriptCase, 2016). Pada hasil keluaran dari aplikasi ini termasuk juga antarmuka yang dibangkitkan berdasarkan tabel basis data dan proses pembangkitan antarmuka dapat dikontrol dengan memberikan aturan tambahan.

Kekurangan dari dua penelitian yang dilakukan oleh Rein Raudj r v dan Atia Mahmud Albhbah adalah, untuk melakukan pembangkitan *form*, skema basis data haruslah dituliskan ke dalam format XML terlebih dahulu dan aturan tambahan untuk proses pembangkitan diletakan di dalam dokumen tersebut. Hal ini dianggap memperpanjang proses yang dilalui untuk melakukan pembangkitan. Sedangkan pada aplikasi GII Generator, proses pembangkitan *model* juga harus dilakukan sebelum melakukan pembangkitan antarmuka CRUD sedangkan kema basis data seharusnya dapat secara langsung menjadi masukan dalam proses pembangkitan antarmuka. Jika dilihat dari aplikasi *ScriptCase* kekurangan yang ditemukan adalah pada hasil antarmuka yang dihasilkan salah satu diantaranya adalah tidak mampu secara



otomatis menentukan *field* yang akan ditampilkan pada antarmuka terkait *field* yang merupakan *foreign key*. Selain itu, penelitian-penelitian sebelumnya dan aplikasi pembangkit formulir diatas tidak mampu mengenali relasi hubungan antar data dari basis data secara otomatis.

Berdasarkan uraian di atas, maka penulis ingin merancang sebuah aplikasi yang mampu membangkitkan elemen formulir berbasis *web* berdasarkan skema basis data secara langsung yang mampu mengenali hubungan antar data, tanpa harus melakukan pembangkitan dokumen lain untuk menjadi inputan dalam proses pembangkitan, serta aturan tambahan yang dibutuhkan untuk mengontrol proses pembangkitan diletakan didalam skema basis data tersebut.

1.2 Rumusan Masalah

Berdasarkan permasalahan yang timbul dari latar belakang, maka dapat dibuat suatu rumusan masalah yaitu bagaimana menerjemahkan skema basis data menjadi formulir *web* secara otomatis berdasarkan standar rekomendasi formulir *web* yang dikeluarkan oleh W3C dan spesifikasi SQL.

1.3 Batasan Masalah

Untuk membatasi ruang lingkup penelitian agar tidak meluas, penulis menetapkan batasan sebagai berikut :

1. Aplikasi yang dibangun hanya menerima input skema dari RDBMS MariaDB dan MySQL.
2. *Storage Engine* yang digunakan pada tabel dalam skema basis data adalah InnoDB.
3. Antarmuka dibangun dengan menggunakan HTML 5, *JavaScript*.

1.4 Tujuan Penelitian

Adapun tujuan yang ingin dicapai dalam penelitian ini adalah :

1. Membuat pemetaan tipe data menjadi elemen *input* formulir HTML.



2. Membuat aplikasi pembangkit antarmuka berdasarkan aturan-aturan yang telah dibuat sebelumnya.

1.5 Sistematika Penulisan

Sistematika dalam penulisan Tugas Akhir terdiri dari pokok-pokok permasalahan yang dibahas pada masing-masing yang diuraikan menjadi beberapa bagian :

BAB I PENDAHULUAN

Bagian ini berisi tentang deskripsi umum Tugas Akhir yang meliputi latar belakang masalah, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan penelitian, serta sistematika penulisan.

BAB II LANDASAN TEORI

Bagian ini menjelaskan tentang teori-teori umum dan teori-teori khusus yang berhubungan dan digunakan dalam penelitian ini. terutama tentang, Basis Data, Antarmuka dan *Hypertext Markup Language* (HTML).

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan tentang metode pengembangan perangkat lunak, kerangka teori penelitian, dan metodologi.

BAB IV ANALISA DAN PERANCANGAN PERANGKAT LUNAK

Bab ini berisi pembahasan mengenai kebutuhan perangkat lunak, yang terdiri dari : Analisa perangkat lunak, UML, Antarmuka, perancangan menggunakan pendekatan berorientasi objek sesuai dengan metodologi pada bab sebelumnya.

BAB V IMPLEMENTASI DAN PENGUJIAN

Bab ini akan membahas implementasi terhadap analisa dan perancangan sebelumnya dan melakukan pengujian penelitian.

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

BAB VI PENUTUP

Bagian ini berisi kesimpulan hasil dari semua tahap yang telah dilalui selama penelitian beserta saran untuk penelitian selanjutnya.



UIN SUSKA RIAU

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.