

BAB I

PENDAHULUAN

1.1. Latar Belakang

Di era digital yang terus berkembang, kualitas perangkat lunak menjadi aspek utama yang perlu dijaga, terutama dalam sektor pendidikan. Website EduALL merupakan platform pendidikan berbasis web yang menyediakan berbagai fitur dan program, seperti admission mentoring, academic & test preparation, serta EduJournal. Sebagai aplikasi web, platform ini juga dilengkapi dengan admin dashboard yang memiliki sistem CRUD untuk mengelola dan memantau konten yang tersedia. Hingga saat ini, website EduALL masih terus melakukan pembaruan fitur untuk menyesuaikan kebutuhan konten yang terdapat pada website. Oleh karena itu, untuk memastikan kualitas perangkat lunak yang optimal, proses pengujian perangkat lunak menjadi langkah penting dalam mengidentifikasi bug dan memastikan fungsionalitas sistem berjalan dengan baik [1].

Metode pengujian manual (*Manual Testing*) masih sering dipakai dalam proses pengujian perangkat lunak, termasuk pada website EduALL. Walaupun metode ini efektif dalam menemukan bug secara eksploratif, prosesnya memakan waktu dan tenaga yang cukup besar, khususnya saat harus menguji berbagai macam kombinasi input [2]. Sebagai solusi yang lebih efisien, pengujian otomatis (*Automation Testing*) menjadi alternatif, terutama untuk menguji fitur yang berulang dan regresi. Dengan menggunakan skrip, proses pengujian dapat berjalan secara otomatis, sehingga menghemat waktu, terutama saat menghadapi perubahan atau pembaruan yang dilakukan secara berkala [3].

Untuk menjaga kualitas website yang optimal, diperlukan proses pengujian yang menyeluruh dan efisien. Salah satu metode pengujian otomatis yang dapat diimplementasikan dalam perangkat lunak yang terus mengalami pembaruan adalah *Model-Based Testing* (MBT) [4]. MBT merupakan metode pengujian perangkat lunak yang menggunakan model sebagai representasi dari perilaku sistem yang diharapkan, sehingga memungkinkan pengujian otomatis yang lebih sistematis [5]. Selain itu, penerapan MBT dapat membantu penguji dalam merancang rangkaian pengujian yang berkualitas dengan cakupan yang optimal, sekaligus mengurangi biaya, waktu, dan usaha dalam seluruh proses pengujian [6]. Penerapan MBT juga membuat pengujian lebih mudah dipelihara, ditinjau, dan diperbarui dibandingkan dengan pengujian manual [7].

Beberapa tools yang mendukung MBT, seperti *MOTES*, *NModel*, *Spec Explorer*, dan *TestOptimal*, dibandingkan untuk mencari tools yang dapat menangani sistem CRUD serta menangani pembaruan atau perubahan pada website dengan cepat dan efisien. Setelah dilakukan perbandingan, *TestOptimal* dipilih karena merupakan rangkaian alat yang dirancang untuk desain dan otomatisasi pengujian yang terintegrasi serta didukung oleh pendekatan MBT [8]. *TestOptimal* dapat membantu mengurangi siklus pengembangan, mencapai cakupan pengujian yang lebih luas, serta menangani pembaruan atau perubahan dengan cepat dan efisien [8]. Model dalam *TestOptimal* berbentuk graf yang merupakan *Extended Finite State Machine* (EFSM) di mana simpul-simpulnya merepresentasikan state (keadaan) dan sisi-sisinya merepresentasikan transition (transisi) [9]. Selain dilengkapi dengan state dan transition, EFSM juga memiliki *guard condition* atau parameter yang berperan sebagai penentu apakah suatu transisi dapat dijalankan atau tidak. Dengan fitur ini, EFSM mampu mengelola berbagai skenario pengujian secara lebih efektif dan efisien [10].

Selain itu, *TestOptimal* menyediakan algoritma atau *model sequencer* yang dapat digunakan secara gratis yaitu *Optimal*, *Random*, dan *Weighted Random Sequencer* yang digunakan untuk mengenerate atau menghasilkan test case. Jika dibandingkan, *Random sequencer* membutuhkan waktu lebih lama untuk mencapai cakupan model yang ditargetkan. Di sisi lain, *Optimal* dan *Weighted Random sequencer* mampu menghasilkan waktu eksekusi model yang lebih singkat [11]. Oleh karena itu, penelitian ini akan membandingkan *Optimal* dan *Weighted Random sequencer* pada *TestOptimal* dengan menerapkan *Model-Based Testing* (MBT) menggunakan *Extended Finite State Machine* (EFSM) untuk menentukan *model sequencer* yang mampu menghasilkan pengujian yang lebih efektif dan efisien.

1.2. Perumusan Masalah

Berdasarkan latar belakang di atas, rumusan masalah yang akan dijawab dalam penelitian ini adalah sebagai berikut.

1. Bagaimana penerapan *Model-Based Testing* (MBT) menggunakan *Extended Finite State Machine* (EFSM) pada pengujian otomatis website EduALL?
2. Bagaimana perbandingan hasil pengujian Sequencer TestOptimal dengan menerapkan *Model-Based Testing* (MBT) menggunakan *Extended Finite State Machine* (EFSM) pada pengujian otomatis website EduALL?

1.3. Batasan Masalah

Batasan masalah dalam penelitian ini adalah:

1. Cakupan website EduALL hanya ada pada lingkup Admin Dashboard
2. Tools *TestOptimal* yang digunakan merupakan free version atau versi gratis

1.4. Tujuan

Berdasarkan rumusan masalah diatas, tujuan dari penelitian ini adalah:

1. Mengimplementasikan *Model-Based Testing* (MBT) menggunakan *Extended Finite State Machine* (EFSM) pada pengujian otomatis website EduALL.
2. Menganalisa perbandingan hasil pengujian Sequencer TestOptimal dengan menerapkan *Model-Based Testing* (MBT) menggunakan *Extended Finite State Machine* (EFSM) pada pengujian otomatis website EduALL.