

1. Pendahuluan

Latar Belakang

Quality control merupakan proses yang memegang peranan penting, karena mengacu pada proses pemantauan dan verifikasi terhadap kualitas suatu perangkat lunak [1]. Proses *quality control* ini memiliki tujuan untuk mengidentifikasi kesalahan, ataupun kecacatan yang ada dalam suatu produk. Jika proses *quality control* ini tidak dijalankan dengan baik, maka dapat memberikan dampak yang buruk terhadap kualitas perangkat lunak [2]. Laporan penelitian yang dilakukan oleh *Standish Group* menjelaskan, bahwa tingkat kesuksesan dalam proyek perangkat lunak itu hanya sebesar 34%, sementara tingkat proyek yang bermasalah itu mencapai 44%, dan tingkat kegagalan proyek mencapai 22% [3]. Penyebab utama dari kegagalan tersebut meliputi: perencanaan yang buruk, resiko yang tidak teridentifikasi, dan yang paling sering terjadi, yaitu persyaratan yang tidak lengkap atau tidak jelas [3]. Selama proses QC ini dilakukan, diperlukan suatu pengujian, sehingga kecacatan atau *bug* yang ada pada perangkat lunak dapat dicari dan langsung diidentifikasi [4]. Namun, dalam merancang suatu rangkaian pengujian yang efektif, dibutuhkan usaha yang signifikan, usaha tersebut nantinya dapat memengaruhi biaya, serta efektivitas pengujian pada pengembangan perangkat lunak, dengan dampak yang relevan terhadap jadwal pengembangan, pemeliharaan, tenggat waktu proyek, dan pada akhirnya memengaruhi kualitas produk akhir [5].

Salah satu unit IT *support* di lingkungan Telkom University, yaitu Direktorat Pusat Teknologi Informasi atau biasa disebut dengan PUTI tengah menghadapi tantangan serupa dalam pemeliharaan sebuah aplikasi. PUTI sendiri merupakan unit yang memiliki peran untuk mengelola sistem, infrastruktur, dan layanan IT di lingkungan Telkom University. PUTI memiliki beberapa aplikasi yang digunakan untuk membantu proses bisnis dalam kegiatan, baik pembelajaran maupun civitas akademik di Telkom University, salah satunya adalah aplikasi BaSO. Aplikasi BaSO ini merupakan aplikasi pembuatan surat dinas resmi secara online di lingkungan Telkom University. Aplikasi ini memungkinkan pengguna, baik itu staf administrasi, dosen, atau pihak terkait, untuk membuat surat dinas resmi secara elektronik. Aplikasi ini sudah berada dalam tahap operasional, yang berarti sudah dapat digunakan oleh pengguna yang berada dalam lingkup Telkom University. Namun, dalam proses pemeliharannya, aplikasi ini masih memerlukan pengujian lebih lanjut untuk memastikan setiap fungsi yang ada dapat dijalankan di berbagai lingkungan *browser* dan dapat digunakan di berbagai *device*.

Berdasarkan permasalahan tersebut, diperlukan suatu pendekatan yang dapat membantu proses pengujian, sehingga kualitas pada aplikasi tersebut dapat terjamin. Pendekatan tersebut adalah *reuse*. Proses *reuse* ini merupakan suatu praktik yang memanfaatkan kembali pengetahuan atau pengalaman yang telah ada [6]. Dalam penerapannya, proses *reuse* ini dapat membantu dalam meningkatkan efisiensi dan kualitas rekayasa sistem, serta mengurangi biaya dan waktu pengembangan [7]. Konsep *reuse* disini dilakukan dengan menggunakan kembali komponen yang ada pada beberapa proyek perangkat lunak yang telah dikembangkan. Fokus penerapan penggunaan kembali ini tertuju pada dokumen kebutuhan perangkat lunak. Dokumen kebutuhan ini mendefinisikan kebutuhan pada perangkat lunak sebagai hasil dari proses analisis yang dilakukan dalam konteks pengembangan perangkat lunak.

Ide utama dari penelitian ini adalah merancang katalog test case berbasis *pattern* untuk mempermudah proses pengujian pada aplikasi BaSO. Dengan memanfaatkan konsep *reuse* serta penggunaan *pattern* terhadap kalimat kebutuhan yang ada di dalam dokumen perangkat lunak, maka pembuatan kasus uji (*Test Case*) dapat dilakukan. Konsep pola atau *pattern* disini, merujuk pada pola yang biasa digunakan dalam kalimat kebutuhan atau kalimat persyaratan. Dengan memanfaatkan penggunaan *Natural Language Processing* (NLP) dan *Regex*, maka kalimat kebutuhan yang terkandung dalam dokumen kebutuhan dapat dikenali, sehingga kalimat tersebut dapat dilakukan proses ekstraksi. Berdasarkan hasil ekstraksi terhadap kalimat kebutuhan ini, maka fungsi yang ada didalamnya dapat diolah dan diproses menjadi sebuah kasus uji atau *test case*. Jika dibandingkan dengan membuat *test case* dari awal, maka menggunakan kembali *test case* yang sudah ada dapat meningkatkan produktivitas dan kualitas pada pengujian [8]. Untuk menilai pengaruh serta efektivitas yang diberikan oleh katalog test case ini, dilakukan penelitian secara eksperimental terhadap tim QA PUTI dalam menguji aplikasi BaSO. Hasil utama dari penelitian ini menunjukkan bahwa penggunaan katalog test case berbasis *pattern* ini dapat meningkatkan penerapan *reuse* serta meningkatkan variasi dalam proses pengujian.

Topik dan Batasannya

Berdasarkan latar belakang diatas, penelitian ini mengangkat beberapa permasalahan penting :

1. Bagaimana implementasi dari perancangan katalog test case berbasis *pattern* ini agar dapat menghasilkan kumpulan test case dari dokumen kebutuhan perangkat lunak yang sudah ada?
2. Apa dampak serta efektivitas yang diberikan oleh katalog test case berbasis *pattern* ini terhadap proses pengujian?
3. Bagaimana proses penerapan penggunaan kembali atau *reuse* ini dapat meningkatkan nilai variasi pada pengujian?

Batasan masalah dari perancangan katalog test case ini adalah sebagai berikut :

1. Pendekatan reuse ini memerlukan pengambilan fungsi berdasarkan dokumen kebutuhan dari proyek-proyek sebelumnya yang sudah melewati proses analisis dan pengambilan kebutuhan terhadap perangkat lunak seperti *Software Requirement Specification (SRS)*.
2. Dokumen yang diambil itu memiliki pola terhadap kalimat kebutuhan yang terkandung didalamnya, yaitu pola *boilerplate* seperti yang sudah ditetapkan.
3. Penelitian ini akan dilakukan terhadap pihak penguji perangkat lunak, seperti *quality assurance*. Hal ini dilakukan untuk melihat dampak dari penggunaan katalog test case selama proses pengujian pada perangkat lunak.

Tujuan

Tujuan yang ingin dicapai dari pengembangan katalog test case ini adalah sebagai berikut :

1. Untuk meningkatkan penerapan penggunaan kembali (*reuse*) terhadap fungsi yang terkandung dalam kalimat kebutuhan, sehingga kasus uji dapat dikembangkan berdasarkan fungsi tersebut.
2. Untuk mengetahui bagaimana efektivitas serta dampak yang diberikan dari penggunaan katalog test case berbasis pattern ini terhadap proses pengujian,
3. Untuk meningkatkan nilai variasi terhadap *test case* yang dihasilkan selama proses pengujian berlangsung.