

BAB 1 PENDAHULUAN

1.1 Latar Belakang

Politeknik Negeri Bengkalis adalah institusi pendidikan tinggi yang terletak di Jl. Bathin Alam, Desa Sungai Alam, Kecamatan Bengkalis, Kabupaten Bengkalis, Provinsi Riau. Politeknik Negeri Bengkalis menyelenggarakan 18 Program Studi aktif dengan jumlah mahasiswa tahun 2021 semester genap sebanyak 2.819 mahasiswa. Setiap tahunnya Politeknik Negeri Bengkalis menyelenggarakan Penerimaan Mahasiswa Baru (PMB). Pada tahun 2020 menerima 261 mahasiswa dan tahun 2021 menerima 289 mahasiswa (PDDikti, 2022).

Politeknik Negeri Bengkalis menyelenggarakan seleksi penerimaan mahasiswa menggunakan tiga jalur penerimaan yaitu melalui: (1) Penelusuran Minat dan Bakat-Politeknik Negeri (PMDK-PN), (2) Ujian Masuk Politeknik Negeri (UMPN), dan (3) Ujian Mandiri Politeknik Negeri Bengkalis. Seleksi jalur ujian mandiri dilakukan menggunakan *Computer Based Test* (CBT) yakni aplikasi komputer yang dibuat untuk menyelenggarakan ujian berbasis komputer. Karakteristik dari tes ini sama dengan test tertulis berbasis kertas yaitu menggunakan suatu perangkat test (soal) untuk dijawab oleh peserta, hanya saja lembar soal dan lembar jawaban pada ujian CBT digantikan oleh perangkat lunak komputer. Ujian diselenggarakan di suatu ruangan yang berisi beberapa peserta test sekaligus, dengan panjang waktu tes yang sama yang telah ditentukan oleh panitia (Tedyyana & Danuri, 2017).

Sejak tahun 2017 Politeknik Negeri Bengkalis telah menggunakan aplikasi CBT untuk menyelenggarakan seleksi mahasiswa baru jalur mandiri. Mempertimbangkan siklus hidup perangkat lunak, setelah beberapa tahun operasional dan perawatan, aplikasi CBT tersebut perlu dievaluasi untuk kemudian diubah atau diganti dengan aplikasi CBT yang baru, yang sesuai dengan kebutuhan dan teknologi saat ini.

Mengembangkan perangkat lunak atau membuat perangkat lunak baru membutuhkan metodologi. Metodologi Pengembangan Perangkat Lunak dapat diartikan sebagai proses membuat suatu perangkat lunak baru untuk menggantikan perangkat lunak lama secara keseluruhan atau memperbaiki perangkat lunak yang telah ada seperti ketika melakukan pengujian aplikasi terdapat banyak error dan fitur-fitur kurang lengkap (Wanas dkk., 2015). Dari sekian banyak metodologi pengembangan perangkat lunak seperti *Waterfall*, *Parallel*, *V-Model*, *Iterative*, *System Prototyping*, *Throw-away Prototyping*, dan *Agile Development*, menyarankan penggunaan kriteria kejelasan kebutuhan pengguna, penguasaan teknologi, tingkat kerumitan sistem, tingkat kehandalan sistem, waktu pelaksanaan, dan visibilitas jadwal pelaksanaan sebagai pertimbangan pemilihan metodologi. Menurut mereka, metodologi *Throw-away Prototyping* memiliki penilaian “sangat baik” khususnya pada kriteria kejelasan kebutuhan pengguna, penguasaan teknologi, tingkat kerumitan sistem, dan tingkat kehandalan sistem, dan memiliki penilaian “baik” pada kriteria waktu pelaksanaan dan visibilitas jadwal pelaksanaan.

Model prototipe atau *prototyping* adalah model pengembangan perangkat lunak yang digunakan untuk membuat versi prototipe dari perangkat lunak. Model ini digunakan saat pemilik proyek atau pengguna tidak memiliki informasi spesifik atau terperinci tentang produk. Pengembang mulai mengembangkan perangkat lunak dengan informasi dan persyaratan minimal kemudian mengambil umpan balik dari pengguna dan membuat perubahan menurut pengguna dan mendefinisikan kembali produk. *Prototyping* adalah proses berulang, proses *trial and error* antara pengembang dan pengguna. Tujuan utama *prototyping* adalah untuk menyediakan sistem fungsional secara keseluruhan sehingga klien dapat memeriksa dan memberikan saran perubahan yang diperlukan. Model *Prototyping* memiliki enam tahapan yaitu : (1) menentukan persyaratan, (2) perancangan cepat, (3) membangun prototipe, (4) evaluasi pengguna, (5) memperbaiki prototipe, (6) mengimplementasikan dan memelihara (Gurung dkk., 2020).

Terdapat empat pendekatan berbeda dalam model prototyping yaitu: (1) *Rapid Throwaway Prototype*, digunakan untuk mendapatkan umpan balik cepat, mendapatkan ide dan perubahan dari pengguna untuk membuat prototipe dengan cepat dan dibuang dan mungkin bukan bagian dari produk akhir, (2) *Evolutionary Prototype*, yaitu membuat beberapa versi prototipe berdasarkan pada umpan balik pengguna yang dapat menghemat waktu dan upaya pengembangan perangkat lunak, (3) *Incremental Prototype*, produk final dibagi menjadi beberapa bagian kecil dan kemudian pada akhirnya terintegrasi menjadi satu, (4) *Extreme Prototype*, digunakan khusus untuk pengembangan web, semua prototipe dibangun menggunakan format HTML dengan *services layer* dan kemudian terintegrasi ke dalam produk terakhir (Gurung et al., 2020).

Bentuk prototype dapat dikategorikan menjadi *low fidelity* dan *hight fidelity prototype*. Prototipe *low fidelity* tidak menggambarkan sistem terlalu rinci, memiliki batasan fungsi atau interaksi, tetapi lebih menggambarkan konsep desain dan tata letak. Prototipe *hight fidelity* lebih rinci menjelaskan sistem, memiliki interaksi penuh dengan pengguna di mana pengguna dapat memasukkan data dan berinteraksi dengan sistem, dan memiliki tampilan yang sangat mirip dengan produk yang sebenarnya dalam (Susanto & Meiryani, 2019). Beberapa *low fidelity prototyping tools* diantaranya *paper sketching*, *storyboarding*, *scripted simulations*, dan *Wizard of Oz (WOZ)*. Sedangkan dengan menggunakan komputer disebut sebagai *medium prototyping* yaitu menggunakan aplikasi *axure*, *balsamiq*, *pencil*, *sketch*, dan *figma*.

Selain memilih metodologi, memilih teknologi implementasi seperti memilih arsitektur sistem dan bahasa pemrograman merupakan hal penting pada pengembangan perangkat lunak. Purnawati dan Sarwono mengimplementasikan framework codeigniter dalam sistem ujian berbasis online pada SMK Bina Teknologi Purwokerto. Framework CodeIgniter dipilih karena dapat mempercepat pengembangan aplikasi, memiliki ukuran yang kecil dibanding dengan framework lain, bersifat *opensorce* sehingga dapat digunakan dengan bebas, dan dokumentasi yang lengkap (Purnawati & Sarwono, 2020). (Riski dkk., 2022) menggunakan framework codeigniter untuk mengembangkan aplikasi Computer Based Test

(CBT) Berbasis Web Pada Universitas Peradaban. Framework codeigniter digunakan karena dapat menghasilkan website yang interaktif dan dinamis dalam waktu yang singkat, mudah dipelajari dokumentasinya, ringan dan cepat.

Perangkat lunak atau aplikasi berbasis web hasil pemrograman harus diuji untuk memperoleh aplikasi yang berkualitas. Jenis pengujian aplikasi berbasis web yaitu *content testing*, *user interface testing*, *usability testing*, *compatibility testing*, *navigation testing*, *configuration testing*, *security testing*, dan *performance testing* (Di Lucca dkk., 2002). ISO/EIC 25010 memberikan karakteristik kualitas perangkat lunak yang dapat dianalisis yaitu *functional suitability*, *performance efficiency*, *compatibility*, *usability*, *reliability*, *security*, *maintainability*, dan *portability* (Ristanto dkk., 2020).

Performance testing atau pengujian kinerja yang dapat dilakukan yaitu (1) *load testing* atau pengujian beban yaitu bentuk pengujian yang dilakukan untuk memahami perilaku sistem di bawah beban tertentu, (2) *stress testing* yaitu pengujian untuk menemukan kapasitas bagian atas yang membatasi sistem dan juga menentukan bagaimana sistem kinerja jika beban saat ini jauh di atas maksimum yang diharapkan (Di Lucca dkk., 2002). Menurut ISO/EIC 25010 sub kriteria yang dapat dianalisis dari *performance efficiency* yaitu *time behaviour*, *resource utilization*, dan *capacity* (Ristanto dkk., 2020).

Load testing dapat dilakukan menggunakan *tools* LoadRunner, Neoload, JMeter (Suryadevara & Ali, 2020). Aplikasi Apache JMeter adalah perangkat lunak open source, aplikasi Java murni, 100% dirancang untuk memuat perilaku fungsional tes dan mengukur kinerja. Apache JMeter dapat digunakan untuk menguji kinerja baik pada sumber daya statis dan dinamis (Web services (SOAP / REST), Web bahasa dinamis - PHP, Java, ASP.NET, File, dll. JMeter dapat digunakan untuk mensimulasikan beban berat pada server, sekelompok server, jaringan atau objek untuk menguji kekuatan atau untuk menganalisa kinerja di bawah jenis beban yang berbeda (Anita Chaudhari, Brinzel Rodrigues, 2016).

Artifak dokumentasi ketika melakukan pengujian web yaitu (1) *test plan*: strategi pengujian, sumberdaya, dan jadwal, (2) *test case*: menjelaskan cara pengujian sistem, apa yang diuji, dengan input apa, dalam kondisi bagaimana, (3) *Bug report*: daftar bug yang ditemukan selama pengujian pada produk yang sedang diuji (Ali & Xiaoling, 2019).

Pengujian aplikasi CBT ini dilakukan pengujian di localhost dan shared hosting. Karena, pengujian ini harus stabil sesuai apa yang diharapkan oleh pencipta dan pengguna aplikasi ini dan dapat membandingkan hasil pengujian, maka pengujian ini dilakukan di localhost dan shared hosting yang mana akan mendapatkan hasil perbedaan dari respon time, throughput, dan error rate nya dari hasil pengujian tersebut.

Berdasarkan latar belakang diatas penelitian bermaksud mengembangkan perangkat lunak CBT untuk Politeknik Negeri Bengkalis dengan menggunakan *Throw-away Prototyping*, dengan bentuk prortotype yang dihasilkan yaitu prototype *medium fidelity* menggunakan aplikasi *Balsamiq Mockup*. Rancangan aplikasi CBT akan diimplementasikan menggunakan teknologi web dengan pemrograman mengikuti framework codeigniter. Sedangkan pengujian yang akan dilakukan adalah pengujian *blackbox* untuk menguji fungsionalitas aplikasi dan pengujian beban (*load testing*) untuk menguji kinerja (*performance*) aplikasi menggunakan aplikasi JMeter.

1.2 Rumusan Masalah

Berdasarkan latar belakang, rumusan masalah penelitian adalah sebagai berikut:

1. Bagaimana merancang perangkat lunak CBT menggunakan model *throw-away prototyping*?
2. Bagaimana menghasilkan perangkat lunak CBT menggunakan *framework codeigniter*?
3. Bagaimana menguji beban (*load testing*) dan mengevaluasi (menilai) penggunaan perangkat lunak pada server menggunakan alat JMeter?

1.3 Tujuan

Adapun dalam penelitian ini yaitu sebagai berikut:

1. Menghasilkan rancangan perangkat lunak CBT menggunakan model *throw-away prototyping* menggunakan aplikasi *Balsamiq Mockup*.
2. Menghasilkan kode program perangkat lunak CBT menggunakan *framework codeigniter*.
3. Menguji fungsionalitas menggunakan *blackbox testing* dan menguji beban (*load testing*) penggunaan perangkat lunak pada server menggunakan jMeter kemudian memberikan saran penggunaan atau perbaikan perangkat lunak terkait hasil pengujian.

1.4 Manfaat

Adapun manfaat dalam penelitian yang penulis lakukan ada beberapa yaitu sebagai berikut:

1. Bagi mahasiswa, penelitian menjadi kegiatan yang mengasah kemampuan melakukan pengembangan perangkat lunak, dan untuk menyelesaikan salah satu syarat menyelesaikan Program Studi Diploma IV di Politeknik Negeri Bengkalis.
2. Bagi instansi tempat penelitian, hasil penelitian dapat digunakan sebagai *prototype* perangkat lunak yang dapat diperbaiki untuk pengembangan CBT Politeknik Negeri Bengkalis yang lebih baik.
3. Bagi perguruan tinggi atau almamater, hasil penelitian dapat menjadi referensi sumber belajar mahasiswa, dan menjadi referensi penelitian-penelitian selanjutnya.