

## BAB 2

### DASAR TEORI DAN TINJAUAN PUSTAKA

Pada bab 2 akan dibahas mengenai dasar teori dan tinjauan pustaka yang digunakan dalam penyusunan Proyek Akhir ini.’

#### 2.1 Dasar Teori

Dasar teori berisi tentang konsep atau pengetahuan, metode, dan tools yang digunakan untuk mendukung penyelesaian Proyek Akhir salah satu metode yang digunakan adalah Uji fungsionalitas dan *Stress Testing*. Selain itu juga terdapat beberapa tools yang digunakan dalam penyusunan Proyek Akhir.

##### 2.1.1 *Test Case*

*Test Case* adalah serangkaian tindakan yang dilakukan pada suatu sistem untuk menentukan apakah sistem tersebut memenuhi persyaratan dan fungsi perangkat lunak dengan benar. Tujuan dari uji kasus adalah untuk menentukan apakah fitur-fitur yang berbeda dalam suatu sistem bekerja sesuai yang diharapkan dan untuk memastikan bahwa sistem tersebut memenuhi semua standar, pedoman, dan persyaratan pelanggan terkait. (Brush, K., 2020)

Struktur *Test Case*: Sebuah *Test Case* biasanya terdiri dari beberapa komponen, seperti:

1. *Fitur* : Merupakan nama fitur atau fungsionalitas yang akan diuji.
2. *Test Case ID*: Identifikasi unik untuk setiap *Test Case* yang membantu dalam pelacakan dan pengelolaan *Test Case*
3. *Deskripsi Test Case*: Penjelasan ini memberikan gambaran umum tentang tujuan dan cakupan pengujian.
4. *Input* : data atau tindakan yang harus dimasukkan atau dilakukan untuk menjalankan *Test Case*.
5. *Expected Result*: Hasil yang diharapkan dari sistem setelah langkah-langkah dijalankan.
6. *Actual Result*: Hasil yang sebenarnya terjadi saat pengujian.
7. *Status*: *Status* akhir dari *Test Case* yang menunjukkan apakah *Test Case* tersebut lulus (pass) atau gagal (*fail*). *Status* ini ditentukan dengan

membandingkan hasil aktual dengan hasil yang diharapkan

Struktur ini sudah mengadopsi dari dengan pedoman ISO/IEC/IEEE 29119-3 yang menetapkan standar internasional untuk dokumentasi pengujian perangkat lunak, memastikan konsistensi dan kualitas dalam proses pengujian. Terutama pada bagian Performance test diantaranya : *test cases, defects, activity progress, and resource consumption*

### **2.1.2 Analisis**

Menurut Komarudin (2001) Analisis adalah kegiatan berfikir untuk menguraikan suatu keseluruhan menjadi komponen sehingga dapat mengenal tanda-tanda komponen, hubungannya satu sama lain dan fungsi masing-masing dalam satu keseluruhan yang terpadu.

### **2.1.3 Stress Testing**

*Stress Testing* adalah proses menjalankan sebuah aplikasi dengan tujuan menemukan sebuah permasalahan. *Performance Testing* dilakukan dengan cara melakukan permintaan dalam jumlah besar, seperti mengakses sistem dengan banyak user dalam waktu bersamaan. *Stress Testing* adalah bagian dari *Performance Testing*. *Performance Testing* sendiri memiliki tujuan untuk mengevaluasi kemampuan dari suatu sistem dalam menangani sebuah permintaan atau request. Adapun tujuan dari *Stress Testing* dan *performance Testing* pada umumnya adalah mengetahui *Response Time* atau latency, throughput, utilisasi sumberdaya, dan *workLoad* (Ridho Subhi et al., n.d.).

### **2.1.4 Throughput**

*Throughput* adalah jumlah bit atau data yang diterima dengan sukses perdetik melalui sebuah sistem atau media komunikasi dalam selang waktu pengamatan tertentu. Aspek utama *throughput* yaitu berkisar pada ketersediaan bandwidth yang cukup untuk menjalankan aplikasi. Hal ini menentukan besarnya trafik yang dapat diperoleh suatu aplikasi saat melewati jaringan, oleh sebab itu semakin besar *throughput* yg dihasilkan akan semakin baik. (Ahdan, 2015)

### **2.1.5 *Response Time***

Waktu Tanggap (*Response Time*) adalah waktu tanggap yang diberikan oleh antar muka interface ketika user merequest permintaan ke server. Secara umum, waktu tanggap yang baik adalah waktu tanggap yang sependekpendeknya. (Linda, 2015)

### **2.1.6 *HTTPS***

*HTTPS* adalah sebuah protokol meminta atau menjawab antara client dan server. Sebuah client HTTP seperti web browser, biasanya memulai permintaan dengan membuat hubungan TCP/IP ke port tertentu di tuan rumah yang jauh (biasanya port 80). Hypertext Transfer Protocol Secure memiliki kelebihan fungsi di bidang keamanan (secure). Dengan menggunakan Secure Socket Layer (*SSL*) atau Transport Layer Security (*TLS*) sebagai sublayer di bawah http aplikasi layer yang biasa. Teknologi *HTTPS* protokol mencegah kemungkinan “dicurinya” informasi penting yang dikirimkan selama proses komunikasi berlangsung antara user dengan web server atau sebaliknya (Zabar & Novianto, 2015)

### **2.1.7 *Black Box Testing***

Pengujian perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program untuk mengetahui apakah fungsi, masukan dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan. Metode *BlackboxTesting* merupakan salah satu metode yang mudah digunakan karena hanya memerlukan batas bawah dan batas atas dari data yang di harapkan, Estimasi banyaknya data uji dapat dihitung melalui banyaknya field data entri yang akan diuji, aturan entri yang harus dipenuhi serta kasus batas atas dan batas bawah yang memenuhi. Dan dengan metode ini dapat diketahui jika fungsionalitas masih dapat menerima masukan data yang tidak diharapkan maka menyebabkan data yang disimpan kurang valid (Wahyu Nur Cholifah . 2018)

1. *Teknik Equivalence partitions*

*Equivalence Partitions* merupakan pengujian *Black Box* berdasarkan masukan data setiap form (Maturidi, 2014) . Metode ini membagi domain masukan dari program ke masing-masing kelas data. *Equivalence partitioning* merupakan pengujian berdasarkan masukan data setiap form yang memecah domain masukan ke dalam kelompok valid atau tidak valid (Priyaungga, Aji, Syahroni, Aji, & Saifudin, 2020)

### 2.1.8 *Apache Jmeter*

*Apache Jmeter* adalah perangkat lunak open-source, murni 100% aplikasi *Java* yang dirancang untuk melakukan uji fungsional dan mengukur kinerja suatu server perangkat lunak (seperti aplikasi web) dan perlu juga diketahui bahwa *Jmeter* bukanlah browser. Dari sisi webservice mungkin *Jmeter* terlihat seperti browser (atau lebih tepatnya beberapa browser), namun *Jmeter* tidak melakukan semua tindakan layaknya browser pada umumnya. Secara khusus, *Jmeter* tidak mengeksekusi *Javascript* yang ditemukan di HTML. Juga tidak merender halaman HTML sebagaimana layaknya browser



Gambar 2.1 *Apache Jmeter*

## 2.2 Tinjauan Pustaka

Beberapa penelitian sebelumnya yang dijadikan referensi untuk pembuatan Proyek Akhir ini meliputi:

Tabel 2.1 Tinjauan Pustaka

Peneliti	Judul Penelitian	Hasil
Fadhila Cahya Ningrum, dkk. (2019)	Pengujian <i>Black Box</i> pada Aplikasi Sistem Seleksi Sales Terbaik Menggunakan Teknik <i>Equivalence partitions</i>	Pengujian pada suatu aplikasi bertujuan untuk memeriksa apakah sebuah program telah berjalan dengan semestinya atau masih ada kesalahan yang harus diperbaiki agar program yang dibuat menjadi program yang memiliki kualitas baik metode <i>Black Box</i> berbasis <i>Equivalence Partitions</i> dapat membantu proses pembuatan case pengujian, uji kualitas, dan menemukan kesalahan yang tidak terdeteksi yang disebabkan oleh kesalahan pengetikan
Madhani, Dzaki, Eko Darwiyanto, dan Arfive Gandhi.(2023)	Performance <i>Testing</i> Menggunakan Metode <i>Load Testing</i> dan <i>Stress Testing</i> pada Sistem Core Banking PT. XYZ	<i>Load Testing</i> adalah pengujian aplikasi di bawah beban yang diharapkan untuk memastikan kinerja aplikasi tetap stabil. Sebelum <i>Load Testing</i> dilakukan, <i>Stress Testing</i> dilakukan terlebih dahulu untuk menentukan beban maksimum yang dapat ditangani aplikasi. Hasil dari performance <i>Testing</i> menunjukkan bahwa sistem core banking PT. XYZ cukup stabil dalam menangani beban yang diharapkan.
Wijaya, Yahya Dwi, and	Pengujian blackbox sistem informasi penilaian kinerja karyawan pt inka (persero)	Sistem informasi penilaian kinerja karyawan PT INKA (Persero) dirancang untuk

Muna Wardah Astuti.(2021)	berbasis <i>partitions</i> <i>Equivalence</i>	membuat proses penilaian kinerja lebih efisien dan paperless. Sebelum digunakan, sistem ini diuji untuk menghindari kesalahan fungsionalitas menggunakan metode <i>Black Box Testing</i> berbasis <i>Equivalence partitions</i> . Pengujian ini berfokus pada fungsionalitas perangkat lunak dengan tujuan menemukan berbagai jenis kesalahan. Setelah melakukan 11 butir <i>Test Case</i> , disimpulkan bahwa tidak ada kesalahan fungsionalitas pada setiap fitur, sehingga sistem informasi siap digunakan.
Krisna Atmajaya	Uji Fungsional dan Analisis Performa Sistem BK_Delivery menggunakan Stress Testing	Hasil pengujian menunjukkan bahwa pengujian <i>Fungsionalitas</i> berhasil memastikan sistem berfungsi sesuai harapan tanpa kesalahan. Stress Testing mengungkap adanya <i>bottleneck</i> pada komponen basis data, jaringan, dan pemrosesan aplikasi saat menghadapi lonjakan beban dengan skenario 200 pengguna. Untuk menjaga kinerja optimal dalam situasi beban tinggi, diperlukan optimalisasi infrastruktur agar sistem dapat terus beroperasi efisien dan memenuhi kebutuhan pengguna dengan baik