

BAB 5. PENUTUP

5.1. Kesimpulan

Berdasarkan analisis data dan pembahasan yang telah dilakukan, penelitian ini menarik beberapa kesimpulan utama sebagai jawaban atas rumusan masalah:

1. Penelitian ini berhasil memvalidasi efektivitas implementasi algoritma *List Virtualization* dan *Graph Pruning* pada Dashboard Manajemen Inventaris, di mana sinergi kedua teknik tersebut dapat berjalan pada ekosistem Next.js tanpa mengubah logika bisnis utama aplikasi.
2. Hasil pengujian performa menunjukkan bahwa teknik optimasi berhasil mengubah kompleksitas rendering menjadi konstan relatif terhadap layar, di mana skenario *Virtualization* paling dominan menurunkan rata-rata TBT dari 1.175,5 ms menjadi 697,25 ms, sedangkan skenario Gabungan mencatatkan responsivitas interaksi terbaik dengan INP sebesar 62,7 ms.
3. Peningkatan performa teknis tersebut terbukti berkorelasi positif dan signifikan terhadap pengalaman pengguna (User Experience), ditandai dengan lonjakan skor SUS dari 38,75 (Poor) pada sistem tanpa optimasi menjadi 89,58 (Excellent) pada sistem dengan arsitektur gabungan.

5.2. Saran

Berdasarkan temuan dan keterbatasan penelitian ini, disarankan beberapa hal untuk pengembangan selanjutnya:

1. Penelitian selanjutnya disarankan memperluas cakupan pengujian ke perangkat seluler (smartphone) *entry-level* fisik untuk memvalidasi performa antarmuka pada layar sentuh serta dampaknya terhadap manajemen termal dan konsumsi baterai.
2. Pengujian skalabilitas data perlu ditingkatkan ke volume yang lebih besar (skala $N > 10.000$) untuk menguji batas toleransi memori peramban terhadap penyimpanan cache dari teknik *Memoization* serta konsistensi frame rate saat pengguliran cepat.
3. Penelitian masa depan dapat mengeksplorasi integrasi teknik virtualisasi dengan fitur modern *Next.js* seperti *React Server Components* (RSC). Pemindahan sebagian logika pemfilteran data dan *rendering* awal ke sisi server berpotensi mengurangi ukuran *bundle* JavaScript yang dikirim ke klien, sehingga dapat semakin mereduksi TBT pada jaringan yang tidak stabil.
4. Penelitian selanjutnya disarankan melakukan studi komparatif antara custom implementation *List Virtualization* yang telah dikembangkan dengan pustaka *production-ready* seperti *react-window*, *TanStack Virtual*, atau *react-virtuoso*. Perbandingan dapat difokuskan pada aspek metrik performa objektif seperti TBT, INP, dan memory footprint. Selain itu,

pengujian terhadap kelengkapan fitur seperti *dynamic heights* dan aksesibilitas akan memberikan kontribusi akademik berupa benchmark antara implementasi riset dengan solusi standar industri, sekaligus memberikan panduan bagi pengembang dalam memilih pendekatan yang tepat sesuai kebutuhan kasus penggunaan mereka.