

BAB V

PENUTUP

5.1 Kesimpulan

Proyek pengembangan aplikasi Minilemon Learning System bertujuan untuk menyediakan platform edukasi digital berbasis video tutorial, alur kerja, dan tugas yang ramah pengguna serta mudah diakses. Aplikasi ini dibangun menggunakan teknologi frontend React.js dan didukung oleh API backend yang terintegrasi, memfasilitasi sistem pembelajaran interaktif secara real-time.

Selama proses pengembangan yang berlangsung selama kurang lebih empat bulan, seluruh fitur utama berhasil diimplementasikan dan diuji secara fungsional. Fitur-fitur yang telah diselesaikan mencakup Ruang Kelas (untuk Mentor & Student), Alur Kerja (untuk Mentor, Student, & Admin), Student Tugas Saya, dan Student Tugas Saya Detail. Pengujian menunjukkan bahwa aplikasi cukup responsif di berbagai perangkat, memberikan pengalaman pengguna yang baik.

Integrasi frontend dengan backend berhasil dilakukan menggunakan RESTful API yang stabil, memungkinkan pengguna untuk melihat, menambahkan, dan mengedit konten secara efisien. Validasi data dan autentikasi juga telah diterapkan untuk menjamin keamanan dan integritas sistem.

Secara keseluruhan, aplikasi Minilemon Learning System telah berhasil memenuhi tujuan pengembangannya, yaitu menghadirkan solusi digital edukatif yang interaktif, ringan, dan dapat diakses dengan mudah oleh berbagai kalangan. Dengan sistem yang terstruktur dan fungsionalitas yang stabil, aplikasi ini siap untuk digunakan secara lebih luas dalam mendukung pendidikan di era digital.

5.2 Saran

Berdasarkan hasil pengembangan dan pengujian aplikasi Minilemon Learning System, terdapat beberapa saran yang dapat dijadikan acuan untuk pengembangan lebih lanjut guna meningkatkan kualitas dan skalabilitas aplikasi:

- **Restrukturisasi Komponen Frontend (Code Refactoring):** Terdapat duplikasi kode pada beberapa komponen antarmuka pengguna yang memiliki struktur dan fungsi serupa. Refactoring perlu dilakukan dengan mengabstraksikan komponen-komponen tersebut menjadi komponen *reusable* untuk meningkatkan efisiensi kode, struktur, dan kemudahan pemeliharaan di masa mendatang.

- **Peningkatan Responsivitas:** Meskipun sebagian besar tampilan aplikasi sudah responsif, masih terdapat beberapa bagian yang belum optimal ketika diakses melalui perangkat *mobile*. Diperlukan penyesuaian lebih lanjut terhadap *grid layout*, ukuran elemen, serta *breakpoint* pada Tailwind CSS untuk memastikan pengalaman pengguna yang konsisten di semua ukuran layar.
- **Peningkatan Performa pada Perangkat Mobile:** Performa aplikasi saat diakses di desktop berjalan sangat lancar, namun saat diakses menggunakan *mobile browser* terdapat pengurangan performa meskipun tidak signifikan, yang berpotensi mengurangi pengalaman pengguna. Perbaikan performa perlu dilakukan untuk menjaga pengalaman pengguna yang konsisten di seluruh perangkat.
- **Penambahan Fitur Edukatif Tambahan:** Untuk memperluas nilai edukatif aplikasi, dapat dipertimbangkan penambahan fitur-fitur seperti kuis interaktif, sistem *badge* atau pencapaian, serta pelacakan progres belajar yang lebih komprehensif.
- **Evaluasi dan Uji Coba Langsung oleh Target Pengguna:** Pengujian langsung terhadap siswa, mentor, dan admin sebagai target utama sangat penting untuk mengidentifikasi kemudahan penggunaan dan aksesibilitas aplikasi yang masih perlu ditingkatkan. Hal ini akan memastikan aplikasi benar-benar sesuai dengan kebutuhan pengguna nyata.
- **Peningkatan Skalabilitas dan Struktur Proyek:** Seiring bertambahnya fitur dan pengguna, struktur kode dan arsitektur aplikasi perlu dipersiapkan agar lebih *scalable*. Salah satu pendekatan yang disarankan adalah pemisahan struktur folder berdasarkan fitur (*feature-based architecture*) serta penguatan manajemen *state* menggunakan alat seperti Zustand atau Context API secara konsisten.

- **Audit Keamanan dan Validasi Data:** Penting untuk memastikan semua data yang dikirim dan diterima melalui API telah divalidasi dengan benar, serta memperkuat autentikasi dan otorisasi pengguna secara berkala.