

## BAB IV PENUTUP

### 4.1 Kesimpulan

Berdasarkan hasil perancangan, pengembangan, dan pengujian *back end* aplikasi *mobile* untuk pengelolaan dan monitoring inventaris di Fakultas Ilmu Komputer Universitas Pembangunan Nasional Veteran Jakarta, diperoleh kesimpulan sebagai berikut:

1. *Back end* aplikasi *mobile* Inventaris FIK UPNVJ berhasil dirancang dan dikembangkan menggunakan Laravel dengan fokus pada keandalan, keamanan, dan efisiensi dalam pengelolaan serta monitoring data inventaris. Aplikasi ini dirancang khusus untuk mendukung kebutuhan Tata Usaha Fakultas Ilmu Komputer melalui penyediaan layanan API yang terstruktur, manajemen data yang konsisten, serta integrasi fitur seperti pencatatan peminjaman dan pengembalian barang, pelaporan kondisi barang, pemindahan barang, hingga pencatatan perbaikan dan pelaporan inventaris secara menyeluruh.
2. Pengujian terhadap *back end* aplikasi *mobile* dilakukan secara bertahap selama proses pengembangan dengan menggunakan *unit testing* pada setiap komponen logika bisnis serta pengujian *black box* melalui integrasi *endpoint* API. Pengujian ini memastikan seluruh fungsionalitas berjalan sesuai spesifikasi dan bebas dari kesalahan kritical sehingga sistem *back end* mampu memenuhi standar keandalan dan kebutuhan operasional Tata Usaha Fakultas Ilmu Komputer. Selain itu, dilakukan pula *User Acceptance Testing* (UAT) oleh pihak Tata Usaha sebagai pengguna akhir untuk memastikan bahwa layanan *back end* layak digunakan dalam lingkungan operasional.

### 4.2 Saran

Dengan mempertimbangkan batasan yang ada dalam proyek ini, berikut beberapa usulan pengembangan lanjutan yang dapat dijadikan referensi untuk memperluas cakupan sistem di masa mendatang:

1. Pengembangan *back end* aplikasi ini dapat diperluas agar tidak hanya melayani kebutuhan Tata Usaha Fakultas Ilmu Komputer Universitas Pembangunan Nasional Veteran Jakarta, tetapi juga dapat digunakan di fakultas berbeda maupun di tingkat universitas. Dengan demikian, sistem ini memiliki potensi untuk mendukung pengelolaan dan monitoring inventaris secara terpusat dan lebih luas dalam lingkup institusi.
2. Sistem *back end* dapat dikembangkan lebih lanjut untuk memastikan kompatibilitas dan optimalisasi layanan API dalam mendukung pengembangan lintas platform, seperti integrasi dengan aplikasi iOS sehingga layanan dapat diakses secara konsisten oleh berbagai jenis perangkat *mobile* tanpa penyesuaian signifikan di sisi *server*.
3. Meskipun Laravel telah terbukti efektif dalam pengembangan *back end* aplikasi ini, pengembangan lebih lanjut dapat mempertimbangkan eksplorasi teknologi lain seperti Node.js, Django (Python), atau Spring Boot (Java) untuk membandingkan performa, skalabilitas, serta kemudahan integrasi sistem. Pendekatan ini dapat membuka peluang optimalisasi arsitektur *back end* sesuai kebutuhan sistem dengan skala yang lebih besar atau dengan kompleksitas yang lebih tinggi di masa mendatang.
4. Pengembangan di masa mendatang dapat mengeksplorasi penerapan metodologi lain selain *Extreme Programming (XP)*, seperti *Scrum* atau *Kanban*, guna menyesuaikan pendekatan kerja dengan kebutuhan tim, skala proyek yang lebih besar, serta untuk meninjau kembali efektivitas metode yang digunakan secara lebih menyeluruh.
5. Untuk meningkatkan kualitas sistem secara keseluruhan, pengujian pada sisi *back end* sebaiknya tidak hanya terbatas pada *unit testing*. Pengujian lanjutan seperti *integration testing*, *end-to-end testing*, dan otomatisasi pengujian dapat diterapkan guna memastikan kestabilan layanan API, terutama pada bagian-bagian sistem yang saling terhubung dan saling bergantung.
6. Dalam proyek serupa di masa mendatang, perencanaan pengujian sebaiknya mencakup tidak hanya *black box testing*, tetapi juga secara jelas

menyertakan *User Acceptance Testing* (UAT) sebagai bagian dari proses pengujian formal sejak tahap awal. UAT memiliki peran krusial dalam menilai apakah aplikasi layak digunakan berdasarkan perspektif pengguna akhir serta memastikan bahwa fungsionalitasnya telah sesuai dengan kebutuhan operasional institusi.

7. Selain berpedoman pada Prosedur Operasional Baku (POB) yang berlaku di Fakultas Ilmu Komputer Universitas Pembangunan Nasional Veteran Jakarta, pengembangan di masa mendatang dapat mempertimbangkan penerapan kebijakan atau standar yang ditetapkan di tingkat universitas. Langkah ini bertujuan untuk memastikan keselarasan administratif dan prosedural dalam cakupan yang lebih luas.