

BAB IV

KESIMPULAN

4.1 Kesimpulan

Berdasarkan Berdasarkan proyek migrasi sistem akses database pada sistem enterprise di PT Dika, dari JDBC menuju R2DBC dengan penerapan keamanan berbasis OAuth 2.0, diperoleh beberapa temuan penting. Hasil implementasi menunjukkan bahwa sistem yang telah dimigrasikan mampu berjalan dengan baik pada arsitektur reaktif yang ditawarkan oleh R2DBC. Selain itu, R2DBC memberikan peningkatan efisiensi dalam pengelolaan koneksi database secara non-blocking.

Proses migrasi akses database dari JDBC ke R2DBC dilakukan dengan menyesuaikan seluruh lapisan pengelolaan data agar kompatibel dengan pendekatan reaktif. Ini mencakup pemetaan ulang query, penggunaan entitas reaktif, serta pemanfaatan fitur non-blocking untuk meningkatkan performa sistem. Hingga saat ini, proses migrasi masih berada dalam tahap pengembangan dan pengujian internal, serta dirancang secara bertahap untuk memastikan stabilitas dan kompatibilitas sistem sebelum diterapkan ke lingkungan produksi.

Dalam mempertahankan integrasi keamanan berbasis OAuth2, sistem otorisasi dipecah menjadi layanan terpisah agar lebih fleksibel dan tidak bergantung langsung pada proses otentikasi di masing-masing layanan. Setiap microservice dapat berkomunikasi melalui token otorisasi yang dihasilkan secara terpusat, sehingga mendukung skalabilitas dan keamanan yang lebih baik.

Beberapa tantangan teknis yang dihadapi selama proses migrasi antara lain adalah kompatibilitas library, pemetaan data kompleks, serta kebutuhan untuk menyesuaikan desain arsitektur agar tetap modular dan maintainable. Tantangan tersebut diatasi dengan pendekatan clean architecture dan pembuatan service terpisah.

4.2 Saran

Dalam migrasi dari JDBC ke R2DBC serta penerapan OAuth 2.0 di PT Dika, disarankan untuk menangani data kompleks seperti BLOB dan JSON melalui service terpisah agar performa reaktif tetap optimal, mengingat R2DBC memerlukan utilitas khusus untuk pemrosesan non-blocking. Selain itu, pemetaan langsung ke DTO di repository layer lebih efektif dibandingkan pemetaan manual, sehingga konsistensi dalam pemetaan perlu dijaga. Mengingat perbedaan paradigma antara JDBC dan R2DBC, dokumentasi teknis dan standarisasi kode juga penting untuk mempercepat pemahaman tim dan menjaga kualitas pengembangan sistem.

