

1. Pendahuluan

1.1 Latar Belakang

Sistem Informasi Akuntansi Badan Usaha Milik Desa (SIABDes) adalah *startup* sekaligus aplikasi pencatatan laporan keuangan. Aplikasi ini dikembangkan oleh Fakultas Informatika dan Fakultas Ekonomi Bisnis Universitas Telkom. Aplikasi ini dicetuskan oleh *founder* SIABDes Dr. Koenta Adji Koerniawan. SIABDes menggunakan Standar Akuntansi Keuangan Entitas Mikro Kecil Menengah (SAK EMKM) yang kompatibel dengan kasus badan usaha milik desa (BUMDes). *Startup* SIABDes juga mengikuti program inkubasi WRAP *Entrepreneurship* yang diselenggarakan Bandung Techno Park. Pada program ini, terutama anggota *startup*, digilir setiap tahun.

Tim pengembang SIABDes mulai mengembangkan *Minimum Viable Product* (MVP) pada 4 April 2023 [Lampiran poin L-2]. Pada bulan Mei 2023, tim pengembang SIABDes berhasil mengembangkan MVP sampai implementasi pencatatan laporan keuangan. MVP ini telah diujikan dan divalidasi pada BUMDes sekitar kota Bandung. Hasil pengujian MVP SIABDes ini mendapat beberapa umpan balik. Umpan balik yang pertama adalah *User Experience* (UX) yang kurang baik untuk pengguna awam akuntansi [1]. Kedua, terdapat BUMDes yang memiliki beberapa unit yang data per unit harus diisolasi, sehingga pencatatan keuangan setiap unit perlu dipisah [2]. Ketiga, tidak ada fitur perpajakan untuk memudahkan perhitungan pajak barang dan karyawan [1].

Selain dari sisi bisnis SIABDes, terdapat berbagai permasalahan pada tim pengembang SIABDes. Masalah pertama dalam tim pengembang SIABDes adalah tidak terdokumentasinya Spesifikasi Perangkat Lunak (SKPL). Hal ini menyebabkan tim pengembang sering melakukan revisi dan validasi kepada tim bisnis, sehingga memperlambat pengembangan. Selain tidak terdokumentasi SKPL, pengembangan MVP ini juga dalam kondisi tergesa-gesa dalam durasi satu minggu, sehingga kualitas kode tidak diimplementasikan dengan baik [Lampiran poin L-5]. Masalah kedua adalah *Quality Assurance* (QA) *Engineer* hanya bekerja sebagai pelengkap SKPL, sehingga proses pengujian tidak berjalan dengan baik [Lampiran poin L-5]. Ketiga proses *deployment* SIABDes ke *production* dilakukan secara manual [Lampiran poin L-5]. Hal ini juga mengurangi efisiensi dan dapat menyebabkan *human error*. Proses *deployment* secara manual berpotensi terjadi *human error* ketika terdapat pembaruan sistem atau fitur.

Berdasarkan lampiran laporan analisis kode menggunakan Sonar Cloud, ditemukan bahwa 642 *code smells* pada sumber kode MVP SIABDes yang ditinggalkan oleh tim pengembang MVP SIABDes [Lampiran poin L-3]. Isu *code smells* ini wajar dalam sumber kode yang bersifat MVP. Sumber kode MVP SIABDes menggunakan Laravel versi 9.19. Pada versi ini terdapat isu keamanan yang dilaporkan terkait otentikasi dan otorisasi [3]. Selain isu keamanan pada versi Laravel, terdapat isu halaman masih dapat mengakses halaman dashboard tanpa melakukan login. Hal ini menandakan bahwa tidak terdapat komponen pemeriksa bahwa pengguna sudah login atau belum. Pada sisi basis data, MVP SIABDes terdapat isu mengenai kualitas skalabilitas dalam pemodelan tabel jurnal umum. Tabel ini menyimpan hasil kalkulasi pada setiap akun yang digunakan dalam jurnal [Lampiran L-14]. Hal ini menjadi isu penting dalam konsistensi data. Jika terdapat kesalahan input dari sisi user, tidak dapat dilakukan *rollback* pada data, dikarenakan tidak terdapat riwayat transaksi pada jurnal. Pada sisi kebutuhan bisnis, *founder* SIABDes juga memperkirakan pada tahun 2024 akan ada 270 BUMDes kabupaten Bandung yang akan menggunakan SIABDes [Lampiran poin L-6].

Selain faktor kurangnya dokumentasi dan *technical debt* pada aplikasi MVP, terdapat beberapa isu dalam arsitektur aplikasi MVP. Aplikasi MVP SIABDes menggunakan *framework* Laravel yang berbasis pola arsitektur *Model View Controller* (MVC). Dalam MVC terdapat komponen *View* sebagai representasi antarmuka pengguna, komponen *Model* sebagai representasi data dan *Controller* sebagai orkestrator logika bisnis. Dalam implementasi MVC pada aplikasi MVP SIABDes, komponen antarmuka dan logika bisnis tidak terisolasi dengan baik sesuai dengan tanggung jawab komponen yang semestinya. Hal ini menjadi penting karena artefak proyek tidak tersedia secara lengkap. Terdapat contoh sumber kode pada MVP SIABDes, yang melakukan kalkulasi kompleks pada komponen *View* untuk halaman Neraca Lajur dan Buku Besar [Lampiran L-13]. Tanpa adanya artefak tentang bagaimana Neraca Lajur dan Buku Besar diproses, dapat menyulitkan pemeliharaan atau bahkan pengembangan fitur baru. Hal ini juga menjadi kurang baik, karena pemrosesan Neraca Lajur dan Buku Besar pada komponen *View* tidak dapat digunakan kembali, sehingga mengakibatkan duplikasi kode dan tidak memiliki sumber perhitungan yang sama. Dalam sistem informasi, konsistensi data dari sisi pemrosesan dan penyimpanan harus dijaga, sehingga penggunaan prinsip penggunaan ulang sebuah logika bisnis harus diterapkan, agar menjaga konsistensi pemrosesan data untuk keseluruhan sistem.

Berdasarkan hal tersebut, peneliti menyarankan dilakukan rekonstruksi secara menyeluruh pada aplikasi SIABDes, dengan dipisahkan komponen antarmuka dan logika bisnis menjadi dua aplikasi berbeda. Aplikasi

komponen antarmuka merupakan aplikasi front-end, sedangkan komponen yang berfokus pada orkestrasi logika bisnis adalah aplikasi back-end. Rekonstruksi aplikasi SIABDes disarankan, karena penambahan fitur dilakukan, sehingga kualitas kemudahan dalam pemeliharaan kode diperlukan. Selain itu, *refactoring* tanpa pengujian unit berpotensi untuk terjadinya *fault* yang tidak terdeteksi. Oleh karena itu, rekonstruksi secara menyeluruh masih dapat dilakukan, karena MVP SIABDes masih dalam tahap uji coba dan belum digunakan untuk pencatatan keuangan sehari-hari oleh BUMDes [Lampiran poin L-4]. Dengan belum adanya penggunaan secara aktif, rekonstruksi secara menyeluruh dapat dilakukan dengan risiko yang minimal.

Untuk membedakan antara MVP SIABDes dengan hasil rekonstruksi MVP tersebut, penerus dari MVP SIABDes diberi nama SIABDes TAXion. SIABDes TAXion berfokus pada pengembangan fitur perpajakan dan perubahan sistem pencatatan laporan keuangan. Untuk mempermudah dan mempercepat proses pengembangan, SIABDes TAXion menggunakan arsitektur modular monolitik. Arsitektur ini diimplementasikan pada konteks aplikasi back-end dari SIABDes TAXion.

Arsitektur modular monolitik dipilih agar mudah diimplementasikan pada tim kecil. Selain itu, arsitektur ini dipilih agar tim pengembang di masa depan memiliki berbagai opsi dalam keputusan teknis. Hal ini dikarenakan modular monolitik mengelompokkan setiap domain ke dalam modul yang sesuai. Salah satu masalah yang sangat mungkin terjadi pada SIABDes adalah masalah skalabilitas. Salah satu solusi skalabilitas dari arsitektur monolitik adalah dengan migrasi ke *microservice*. Sehingga, modular monolitik dapat mempermudah migrasi ke arsitektur *microservice* yang sangat bergantung pada pembagian domain yang jelas.

Berdasarkan uraian sebelumnya, rekonstruksi SIABDes diajukan dengan mempertimbangkan faktor keamanan, perubahan fitur pencatatan laporan keuangan, penambahan fitur perpajakan. Selain peningkatan secara fungsionalitas dan keamanan, rekonstruksi SIABDes diharapkan dapat meningkatkan kualitas *maintainability* dengan konsep modular yang didukung oleh modular monolitik. Sehingga tim pengembang SIABDes saat ini, dapat membuka opsi keputusan teknis yang lebih luas kepada tim pengembang SIABDes di masa depan.

1.2 Topik dan Batasannya

Batasan masalah pada penelitian ini, hanya berfokus pada konteks implementasi back-end pada aplikasi SIABDes TAXion. Dengan demikian, masalah yang diidentifikasi di antaranya:

1. Bagaimana implementasi back-end SIABDes agar dapat memenuhi kebutuhan pencatatan laporan keuangan dan perpajakan PPN dan PPh 21?
2. Bagaimana implementasi back-end SIABDes dengan arsitektur modular monolitik?
3. Bagaimana agar implementasi back-end SIABDes dapat digunakan oleh 270 BUMDes secara bersamaan, tanpa penurunan performa signifikan?
4. Bagaimana perbaikan terhadap isu keamanan pada MVP SIABDes, sehingga dapat meningkatkan aspek keamanan pada versi kedua aplikasi, yaitu SIABDes TAXion?

1.3 Tujuan

Tujuan dari penelitian ini adalah:

1. Melakukan implementasi SIABDes agar memenuhi kebutuhan pencatatan laporan keuangan dan perpajakan PPN dan PPh 21.
2. Mengimplementasikan back-end pada aplikasi SIABDes dengan arsitektur modular monolitik.
3. Mengimplementasikan back-end pada aplikasi SIABDes, agar dapat digunakan 270 BUMDes secara bersamaan tanpa penurunan performa signifikan.
4. Memperbaiki isu keamanan mengenai otentikasi dan otorisasi MVP SIABDes melalui aplikasi back-end SIABDes TAXion.

1.4 Organisasi Tulisan

Guna memperjelas laporan penelitian ini, struktur laporan ini adalah:

- A. BAB I Pendahuluan, menjelaskan alasan mengapa penelitian ini dilakukan, serta tujuan dan batasan dari laporan ini.
- B. BAB II Kajian Pustaka, menjabarkan penelitian, teori dan teknologi terkait yang digunakan pada penelitian ini.
- C. BAB III Sistem yang Dibangun, menjelaskan mengenai metodologi, alur pengerjaan, arsitektur aplikasi dan penggunaan *Domain Driven Design*.
- D. BAB IV Evaluasi dan Hasil, menjabarkan implementasi dari penelitian ini dan hasil dari evaluasi implementasi penelitian ini.
- E. BAB V Kesimpulan dan Saran, berisikan kesimpulan dari hasil penelitian ini, beserta saran untuk peneliti yang melanjutkan mengenai penelitian SIABDes TAXion.