

Pengembangan Aplikasi *White Label* PPOB berbasis Web menggunakan Pemodelan *Reuse Oriented Development* (Studi pada PT XYZ)

Andi Hafiidh¹, Herman Tolle², Welly Purnomo³

Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹andi.course@gmail.com, ²emang@ub.ac.id, ³wepe@ub.ac.id

Abstrak

Whitelabel adalah kegiatan yang merujuk pada membeli suatu produk / jasa oleh *reseller* lalu diberi label baru seolah pihak *reseller* tersebut yang memproduksinya. Sedangkan *payment point online bank* (*PPOB*) merupakan sistem pembayaran tagihan dan pembelian produk seperti pulsa, paket data, token listrik, *voucher* elektronik, dan sebagainya secara *online*. Latar belakang dilakukannya penelitian ini adalah saat ini terdapat salah satu penyedia layanan *PPOB* di Indonesia yaitu PT XYZ (nama perusahaan disamarkan). *PPOB* PT XYZ kurang lebih memiliki 26.000 pengguna, 17% diantaranya adalah member milik mitra PT XYZ. PT XYZ membuka kerjasama kemitraan dengan perusahaan/organisasi lain untuk mengembangkan *PPOB* atas nama perusahaan/organisasi tersebut. Metode tersebut merupakan metode pemasaran yang disebut dengan *whitelabel*. Meski saat ini aplikasi *PPOB whitelabel* dengan mitra sudah berjalan dengan baik, namun terdapat beberapa kebutuhan baru diantaranya penyesuaian keuntungan produk, penggunaan rekening bank milik mitra, notifikasi SMS pada pembelian token PLN, dan aplikasi *PPOB* berbasis web. Untuk dapat menyelesaikan kebutuhan baru tersebut dengan mudah, cepat, dan memiliki kualitas yang cukup baik maka penggunaan metode *reuse oriented development* sangat dianjurkan. Pada penelitian ini dihasilkan sistem *PPOB* sederhana yang dapat memenuhi kebutuhan baru dari *whitelabel PPOB* milik PT XYZ yang sudah ada, dengan menggunakan struktur *microservice* dan menggunakan *library (reusable component)* dan layanan dari pihak ketiga (*software as a service*). Sistem tersebut juga telah diuji menggunakan *validation testing* dan mendapatkan hasil 100% valid, tidak ada *critical issues* pada *compatibility testing*, dan mendapat nilai sebesar 81,5% pada *user acceptance test (UAT)*.

Kata kunci: *reusable component, microservice, validation testing, compatibility testing, user acceptance test*

Abstract

Whitelabel is an activity that refers to buying a product/service by a reseller and then given a new label as if the reseller produced it. Meanwhile, *payment point online bank (PPOB)* is a bill payment system and products selling such as phone credit, phone quota, electricity tokens, electronic vouchers, etc. The background of this research is that currently there is one *PPOB* service provider in Indonesia, PT XYZ (the company name is obscured). *PPOB* PT XYZ has approximately 26,000 users, 17% of which are members owned by PT XYZ partners. PT XYZ opened a partnership with other companies / organizations to develop *PPOB* on behalf of the company / organization. This method is a marketing method called *whitelabel*. Although currently the *PPOB whitelabel* application with partners has been running well, but there are some new needs including adjustment of product profits, use of partner bank accounts, SMS notifications on PLN token purchases, and web-based *PPOB* applications. To be able to solve these new needs easily, quickly, and have a fairly good quality, the use of *reuse oriented development* methods is highly recommended. In this study a simple *PPOB* system was produced that could meet the new needs of PT XYZ's existing *PPOB whitelabel*, using a *microservice* structure and using a *library (reusable component)* and services from third parties (*software as a service*). The system gets 100% valid results on *validation testing*, also there are no *critical issues* in *compatibility testing*, and gets a score of 81.5% on *user acceptance tests (UAT)*.

Keywords: *reusable component, microservice, validation testing, compatibility testing, user acceptance test*

1. PENDAHULUAN

Payment Point Online Bank atau yang biasa disebut *PPOB* merupakan sebuah sistem pembayaran berbagai macam tagihan secara online dengan pihak bank sebagai pihak penyelenggara dan juga penampung dana *customer* kepada mitra kerjanya. Saat ini terdapat berbagai macam layanan yang disediakan oleh *PPOB* diantaranya adalah pembelian pulsa, pembelian paket data, pembelian token listrik, pembayaran tagihan listrik PLN, pembayaran tagihan PDAM, pembayaran tagihan telepon, pembayaran tagihan kartu pascabayar, pembayaran tagihan internet, dan pembayaran tagihan *multi finance*. Salah satu badan usaha yang bekerjasama dengan pihak bank dalam menjadi mitra penyedia jasa *PPOB* adalah PT XYZ (nama perusahaan disamarkan).

Saat ini PT XYZ sudah memiliki 26.000 lebih pelanggan yang menggunakan layanan *PPOB* nya. Dari total seluruh pelanggan tersebut 17% diantaranya merupakan pelanggan dari mitra yang bekerjasama dengan PT XYZ. Mitra tersebut bekerjasama secara *whitelabel* dengan PT XYZ untuk membuat *PPOB* atas nama mitra tersebut, namun dengan memanfaatkan *core system PPOB* milik PT XYZ. *Whitelabel* merujuk pada kegiatan membangun *brand* baru oleh pihak *reseller* kepada sebuah produk atau layanan seolah olah merupakan produk baru dengan *reseller* sebagai pemilik nya (Techopedia, no date). Keuntungan yang diperoleh dari perusahaan penyedia barang atau jasa adalah mengurangi biaya untuk promosi sehingga dana investasi dapat di fokuskan ke arah lini produksi, dan bagi *reseller* tidak perlu biaya dalam produksi barang dan jasa, cukup fokus ke pembiayaan untuk *marketing* (Investopedia, 2018). Kerjasama *whitelabel* yang dijalankan PT XYZ dengan para mitranya yaitu dalam pembagian keuntungan dari transaksi yang berhasil sesuai dengan perjanjian awal kerjasama.

Saat ini sudah ada beberapa aplikasi *whitelabel* yang digunakan oleh mitra PT XYZ, namun aplikasi yang sudah ada masih memiliki beberapa kebutuhan yang belum direalisasikan diantaranya adalah pengaturan keuntungan yang di inginkan oleh mitra, penggunaan akun rekening bank milik mitra sebagai akun bank pengisian saldo, notifikasi SMS pada pembelian produk token PLN, aplikasi *PPOB* berbasis web, serta pengelolaan dan kontrol data oleh mitra PT

XYZ itu sendiri. Beberapa kebutuhan baru tersebut muncul dikarenakan kurangnya *reusable component* yang digunakan dalam membangun aplikasi *whitelabel PPOB* yang ada saat ini. Saat ini PT XYZ hanya memiliki kode program aplikasi *mobile PPOB* berbasis android sebagai *reuseable component* yang digunakan. Ketika ada mitra yang ingin bekerja sama dalam untuk membuat aplikasi *PPOB*, maka yang dilakukan oleh *developer* PT XYZ adalah menyalin kode program tersebut lalu melakukan perubahan dalam segi tampilan. Hal tersebut membuat seluruh fitur yang ada pada aplikasi *whitelabel PPOB* milik PT XYZ sama dengan aplikasi *PPOB* utama milik PT XYZ. Dengan menambahkan beberapa *reusable component* yang memiliki fungsi *core system* milik PT XYZ maka fitur fitur yang dibutuhkan untuk mitra tersebut dapat terealisasi.

Untuk dapat menerapkan *whitelabel* secara maksimal, maka dibutuhkan *reusable component* sebanyak mungkin agar saat proses pembuatan aplikasi *whitelabel* dapat lebih cepat, lebih mudah, dan memiliki kualitas yang bagus. Sesuai dengan beberapa kelebihan *reuse oriented development* yaitu dapat mengurangi resiko kegagalan dalam proses pembuatan, dan lebih cepat dalam pengerjaan nya (Sommerville, 2011). Dikarenakan aplikasi *whitelabel* yang dibuat hanya memiliki sedikit perbedaan antar aplikasinya. Untuk dapat memodelkan *reuse component* maka dapat menggunakan *reuse-based software engineering* yang juga dikenal dengan *reuse oriented development*. Dalam *reuse oriented development* terdapat beberapa tahapan yang penggalan kebutuhan, analisa komponen, modifikasi kebutuhan, desain sistem berbasis penggunaan ulang, pengembangan dan integrasi, serta validasi sistem. Dari tahapan tahapan tersebut maka kebutuhan komponen yang dapat digunakan kembali (*reusable component*) dapat dipetakan dengan baik.

Berdasarkan latar belakang yang sudah dijelaskan sebelumnya, peneliti ingin mengajukan sebuah penelitian yang diharapkan dapat digunakan oleh PT XYZ dalam mengembangkan aplikasi *whitelabel PPOB* nya, guna melengkapi kebutuhan baru aplikasi *whitelabel PPOB* sebelumnya, serta mempermudah mitra dalam mengelola dan mengontrol aplikasi *whitelabel PPOB* nya. Dengan adanya hasil penelitian ini diharapkan juga akan ada lebih banyak calon mitra yang tertarik untuk bekerja sama dengan PT XYZ dalam membuat aplikasi *PPOB* nya, sehingga

jumlah member dan transaksi akan meningkat, dan pendapatan dari PT XYZ akan bertambah

2. LANDASAN KEPUSTAKAAN

Terdapat penelitian yang terkait dengan pengembangan sistem yang menggunakan metode serupa yaitu “Pengembangan Sistem Informasi Aspirasi Online Berbasis Web Menggunakan Pemodelan Reuse-Oriented Development (Studi Kasus : DPM Universitas Brawijaya)” oleh Ali Fikri pada tahun 2018. Menurut hasil penelitian tersebut didapatkan hasil uji rata rata efisiensi waktu dalam penyampaian dan penyerapan aspirasi mahasiswa bagi DPM UB sebesar 84,4%, hasil tersebut menunjukkan bahwa proses bisnis usulan memiliki efisiensi waktu yang lebih baik dibanding proses bisnis sebelumnya (Fikri, Aknuranda and Pradana, 2019). Dari hasil penelitian tersebut dibuktikan bahwa penggunaan metode *reuse oriented development* dapat dapat digunakan untuk mengembangkan sebuah sistem, dan juga menghasilkan sistem yang cukup baik.

Selain itu penelitian tersebut juga membuktikan bahwa metode *reused oriented development* yang digunakan menghasilkan kualitas dan produktivitas yang lebih tinggi daripada menggunakan metode pengembangan konvensional (Baldassarre *et al.*, 2005). Dengan membuat beberapa *reusable component* yang dapat digunakan sebagai standar dalam pengembangan sistem dan aplikasi dalam penelitian ini maka diharapkan dapat meningkatkan kualitas dan produktivitas pada saat proses implementasinya.

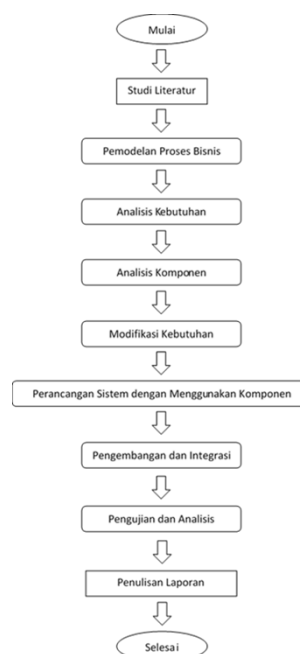
Pada penelitian lain yang berjudul "Lihonga — a Microservice-based Virtual Learning Environment" oleh Samuel Kapembe pada tahun 2018, dimana beliau mendesain dan membangun sistem yang dapat mendukung aplikasi *e-learning* modern menggunakan *microservice architecture* yang memiliki fokus dalam bagaimana mendesain dan membangun *microservice* pada *backend* yang dibutuhkan untuk menunjang aplikasinya. Selain itu, pada penelitian tersebut memanfaatkan *application container technology* yaitu Docker, serta *continuous integration/deployment* dalam melakukan *deployment* secara otomatis. Dengan memanfaatkan *microservice* pada struktur sistem, maka manfaat yang didapatkan diantaranya mudah dalam melakukan pemeliharaan dan uji coba, hubungan antar

microservice tidak terlalu rapat sehingga pada saat terjadi gangguan / kerusakan pada satu *microservice* tidak akan mengganggu *microservice* lainnya, dapat digunakan secara terpisah sehingga bisa digunakan pada sistem lain (Richardson, 2018). Dari penelitian tersebut akan menjadi contoh dalam membangun *microservice* pada penelitian ini.

Untuk dapat membangun *microservice* dengan mudah dan cepat, maka dapat memanfaatkan *reusable component* yang sudah dibuat untuk menjadi standar dalam pengembangan sistem dan aplikasi. Dalam pengembangan *microservice* dibutuhkan *API gateway* sebagai salah satu komponen standar seperti yang dijelaskan pada artikel "*Management of API Gateway Based on Microservice Architecture*" yang dibuat oleh J T Zhao pada tahun 2019. Penelitian tersebut akan menjadi acuan dalam pembuatan *API gateway microservice* pada penelitian ini.

3. METODOLOGI

Pengembangan aplikasi pada penelitian ini menggunakan metode *reuse oriented development*. Tahapan tahapan dalam metode yang digunakan dalam penelitian terdapat dalam Gambar 1.



Gambar 1 Metodologi Penelitian

Metode penelitian ini dimulai dengan studi literatur untuk memahami teori terkait penelitian yang dilakukan. Selanjutnya pemodelan proses

bisnis yang dilakukan untuk mendefinisikan proses serta memodelkan aliran aktivitas bisnis dan prosedur yang berjalan. Kemudian analisis persyaratan merupakan tahapan untuk mengetahui spesifikasi persyaratan yang dibutuhkan sistem. Lalu analisis komponen merupakan tahapan pengumpulan komponen yang dapat digunakan. Setelah itu komponen yang memerlukan modifikasi akan disesuaikan pada tahapan modifikasi kebutuhan. Dilanjutkan dengan tahapan perancangan sistem menggunakan komponen. Kemudian melakukan pengembangan dan integrasi. Lalu melakukan pengujian dan analisis. Dan diakhiri dengan penulisan laporan. Setelah tahapan studi literatur dan sebelum penulisan laporan merupakan tahapan tahapan pada pemodelan *reuse-oriented development*.

4. PEMODELAN PROSES BISNIS DAN ANALISIS KEBUTUHAN

Sistem yang dibuat pada penelitian ini merupakan sistem *PPOB* sederhana yang menjadikan sistem PT XYZ sebagai *vendor* penyedia layanan *PPOB* dari sistem *whitelabel PPOB* pada penelitian ini. Pada tahap pemodelan proses bisnis tersebut menghasilkan gambaran proses bisnis yang berjalan pada saat ini (*as-is*), yang kemudian akan dikembangkan menjadi proses bisnis usulan (*to-be*). Beberapa proses bisnis yang menjadi bahasan utama dalam penelitian ini diantaranya adalah:

Proses bisnis *member login* dimana member pada sistem *PPOB* dapat masuk ke dalam aplikasi dan memiliki hak akses dalam menggunakan aplikasi *PPOB* tersebut. Terdapat perbedaan proses *login* yang ada pada sistem *PPOB* saat ini dengan sistem *PPOB* usulan. Pada sistem *PPOB* saat ini *login member* menggunakan nomor telepon genggam dan *password* sedangkan pada proses *login* usulan menggunakan nomor telepon genggam dan kode sekali pakai atau yang lebih dikenal dengan *one-time password (OTP)* yang dikirimkan melalui SMS.

Kemudian proses bisnis isi saldo member dimana member dapat melakukan pengisian saldo sebelum dapat menggunakan aplikasi dalam bertransaksi baik pembelian produk maupun pembayaran tagihan. Terdapat perbedaan pada rekening bank yang digunakan untuk mengisi saldo member. Pada sistem *PPOB* saat ini rekening bank yang digunakan adalah rekening bank milik PT XYZ, sedangkan pada

sistem *PPOB* usulan menggunakan rekening bank milik mitra. Perbedaan tersebut mengakibatkan kebutuhan sistem untuk dapat mengetahui mutasi rekening bank milik mitra itu sendiri.

Lalu terdapat juga proses bisnis pembelian produk dimana member dapat melakukan pembelian produk melalui layanan *PPOB* ini. Dalam hal ini sistem *PPOB* milik PT XYZ menjadi *vendor* dari sistem *PPOB* usulan. Dengan menjadikan sistem *PPOB* PT XYZ saat ini sebagai *vendor*, maka sistem *PPOB* usulan akan dianggap sebagai member pada sistem *PPOB* PT XYZ yang sedang melakukan transaksi pembelian produk.

Serta proses bisnis pembayaran tagihan yang tidak jauh berbeda dengan proses pembelian produk. Pada proses pembayaran tagihan, sistem *PPOB* milik PT XYZ menjadi *vendor* pada proses bisnis usulan yang dibuat.

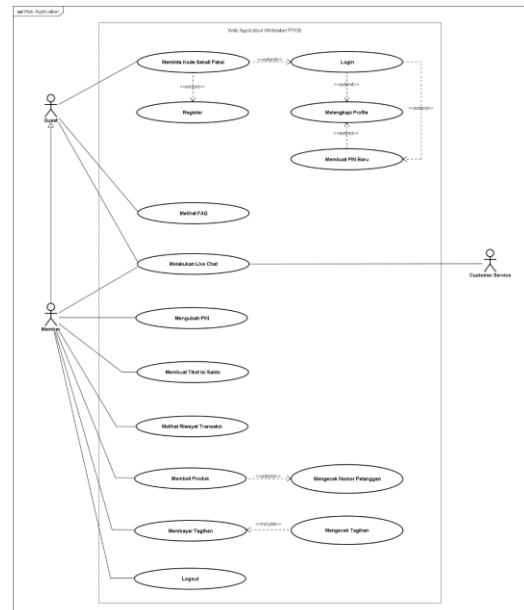
Setelah melakukan pemodelan proses bisnis maka tahapan selanjutnya adalah penggalan spesifikasi kebutuhan sistem dimana pada tahapan ini ditemukan 58 kebutuhan fungsional dan 4 kebutuhan non fungsional yang dapat dilihat pada Tabel 1 dan Tabel 2 dibawah ini.

Tabel 1 Kebutuhan Fungsional

Kode	Kebutuhan Fungsional
PF-01	Request kode sekali pakai
PF-02	Login
PF-03	Logout
PF-04	Member register
PF-05	Membuat PIN baru
PF-06	Melengkapi identitas
PF-07	Mengubah PIN
PF-08	Membuat tiket isi saldo
PF-09	Membeli pulsa
PF-10	Memproses pembelian pulsa
PF-11	Mengubah status transaksi pembelian pulsa
PF-12	Membeli paket data
PF-13	Memproses pembelian paket data
PF-14	Mengubah status transaksi pembelian paket data
PF-15	Membeli token PLN
PF-16	Memproses pembelian token PLN
PF-17	Mengubah status transaksi pembelian token PLN
PF-18	Membeli voucher elektronik
PF-19	Memproses pembelian voucher elektronik
PF-20	Mengubah status transaksi pembelian voucher elektronik
PF-21	Membayar tagihan PLN
PF-22	Memproses pembayaran tagihan PLN
PF-23	Mengubah status transaksi pembayaran tagihan PLN
PF-24	Membayar tagihan PDAM
PF-25	Memproses pembayaran tagihan PDAM

PF-26	Mengubah status transaksi pembayaran tagihan PDAM
PF-27	Membayar tagihan Telkom
PF-28	Memproses pembayaran tagihan Telkom
PF-29	Mengubah status transaksi pembayaran tagihan Telkom
PF-30	Membayar tagihan Indihome
PF-31	Memproses pembayaran tagihan Indihome
PF-32	Mengubah status transaksi pembayaran tagihan Indihome
PF-33	Membayar tagihan kartu pascabayar
PF-34	Memproses pembayaran tagihan kartu pascabayar
PF-35	Mengubah status transaksi pembayaran tagihan kartu pascabayar
PF-36	Membayar tagihan <i>multi finance</i>
PF-37	Memproses pembayaran tagihan <i>multi finance</i>
PF-38	Mengubah status transaksi pembayaran tagihan <i>multi finance</i>
PF-39	<i>Live chat</i>
PF-40	Halaman <i>FAQ</i>
PF-41	Melihat riwayat transaksi
PF-42	<i>Login administrator</i>
PF-43	<i>Logout administrator</i>
PF-44	Menampilkan data grafik, dan informasi saldo vendor
PF-45	Membuat tiket isi saldo vendor PPOB
PF-46	Mengatur rekening bank
PF-47	Mengelola mutasi rekening bank
PF-48	Mengatur konfigurasi sistem
PF-49	Mengelola data member
PF-50	Isi saldo manual
PF-51	Mengelola data transaksi
PF-52	Mengelola data kategori admin
PF-53	Mengelola data admin
PF-54	Mengelola data hak akses admin
PF-55	Meminta data produk
PF-56	Mengirimkan sms
PF-57	Menerima mutasi rekening
PF-58	Pengecekan transaksi isi saldo member

aktor dengan aplikasi *whitelabel PPOB* dan *dashboard backoffice*.



Gambar 2 Use case Diagram Aplikasi Web *Whitelabel PPOB*

Dapat dilihat pada Gambar 2 dan Gambar 3 pada masing masing use case diagram terdapat 3

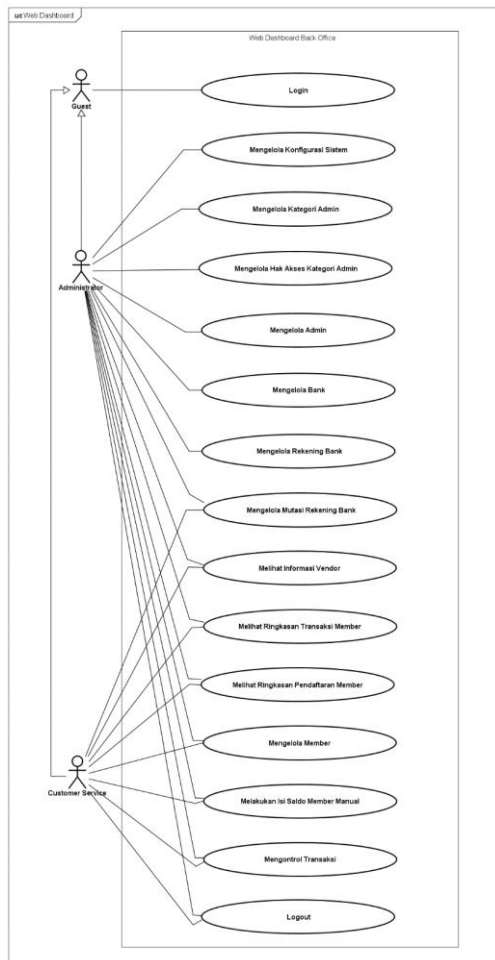
Kebutuhan kebutuhan tersebut didapatkan dari hasil observasi dan wawancara yang dilakukan pada perusahaan XYZ.

Tabel 2 Kebutuhan Non Fungsional

Kode	Kebutuhan Non Fungsional
PNF-01	<i>Security</i>
PNF-02	<i>Usability</i>
PNF-03	<i>Compatibility</i>
PNF-04	<i>Accessibility</i>

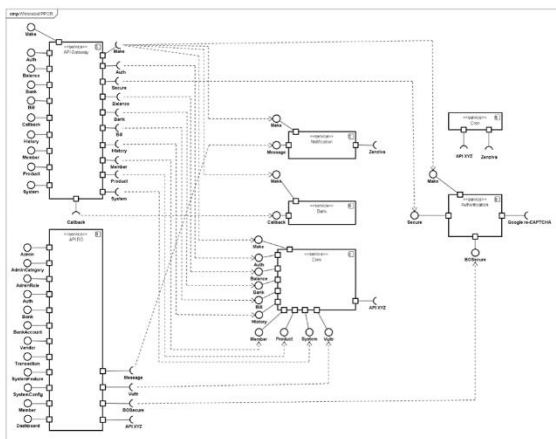
Setelah kebutuhan fungsional dan non fungsional didapatkan, maka tahapan berikutnya adalah pembuatan *use case diagram*. Dimana diagram tersebut akan menunjukkan interaksi apa saja yang terjadi antara aktor dengan sistem. Berikut ini terdapat 2 buah *use case diagram* yang dibuat untuk menggambarkan interaksi

orang aktor dan 15 use case di dalamnya.



Gambar 3 Use Case Diagram Dashboard Backoffice

Selain pemodelan menggunakan use case diagram, diperlukan pemodelan menggunakan komponen diagram guna mengetahui interaksi antar komponen yang digunakan pada sistem *whitelabel PPOB* yang dibuat ini. Hasil pemodelan komponen diagram dapat dilihat pada Gambar 4 di bawah ini.

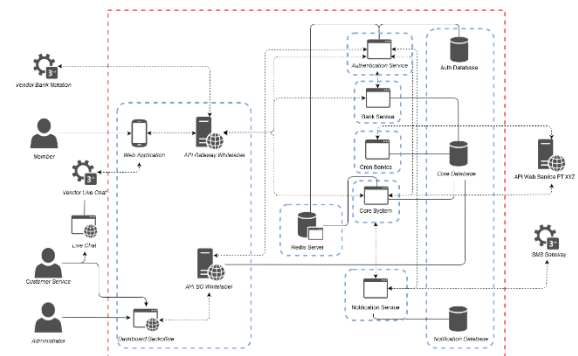


Gambar 4 Komponen Diagram Backend API

Whitelabel PPOB

5. ANALISIS KOMPONEN DAN PERANCANGAN

Tahap analisis komponen dilakukan untuk mencari komponen yang sesuai dengan spesifikasi kebutuhan (Sommerville, 2011). Ditemukan 31 *reusable component* dan layanan *software as a service (SAAS)* yang dapat digunakan dalam proses implementasi sistem, 3 komponen yang memerlukan modifikasi guna memenuhi kebutuhan. Selain itu menghasilkan 6 jenis rancangan sistem, yaitu berupa rancangan arsitektur sistem yang terdapat pada Gambar 5, beberapa rancangan sequence diagram, 7 buah rancangan diagram kelas pada masing masing *microservice*, 3 buah rancangan *database*, beberapa algoritma sistem, 18 rancangan antarmuka aplikasi web *whitelabel PPOB*, dan 10 rancangan antarmuka *dashboard backoffice*.



Gambar 5 Arsitektur Sistem

Dari komponen ke 31 reusable component yang dapat digunakan dalam mengembangkan sistem terdapat kebutuhan untuk memodifikasi 3 komponen diantaranya, yang dapat dilihat pada Tabel 3 di bawah ini.

Tabel 3 Daftar Modifikasi Komponen

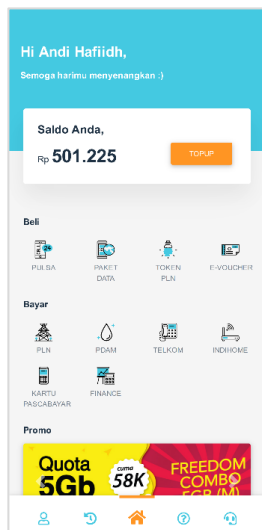
Nama	Sebelum	Sesudah
Kustomisasi tampilan <i>button</i> pada <i>plugin freshchat</i>	<i>Button</i> asli milik <i>plugin freshchat</i> menggunakan <i>floating button</i> .	Diubah menjadi <i>icon</i> yang terdapat pada <i>navigation menu</i> di bawah halaman.
Penyesuaian <i>callback function</i> milik <i>cekmutasi.co.id</i>	<i>Cekmutasi.co.id</i> menyediakan contoh fungsi penerima data mutasi dalam bentuk file berekstensi <i>php</i>	Fungsi disesuaikan dengan <i>framework Code Igniter</i> .

Penyesuaian kelas <i>API_Controller</i>	Fungsi fungsi yang terdapat pada kelas <i>API_Controller</i> merupakan fungsi standar dari <i>reusable component</i> yang dibuat.	Terdapat penambahan fungsi pada kelas <i>API_Controller</i> dalam beberapa <i>microservice</i> .
---	---	--

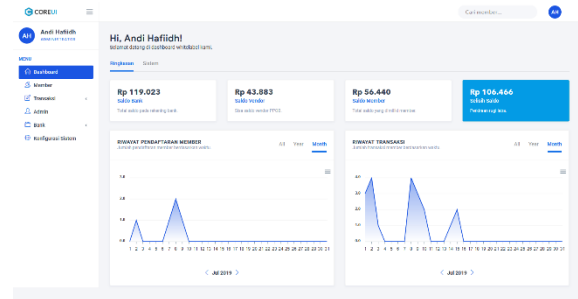
Setelah komponen berhasil dikumpulkan dan disesuaikan dengan kebutuhan maka tahapan berikutnya adalah perancangan dimana pada tahap ini memanfaatkan hasil reusable component yang sudah dianalisis sebelumnya.

6. IMPLEMENTASI

Berikut ini adalah beberapa contoh gambar yang menunjukkan hasil implementasi dari antarmuka pengguna pada aplikasi web *whitelabel PPOB* dan *dashboard backoffice*. Gambar 6 menunjukkan hasil implementasi halaman utama aplikasi web *whitelabel PPOB*. Sedangkan Gambar 7 merupakan hasil implementasi antarmuka pada halaman utama dari *dashboard backoffice*.



Gambar 6 Hasil Implementasi Halaman Utama Aplikasi *Whitelabel PPOB*



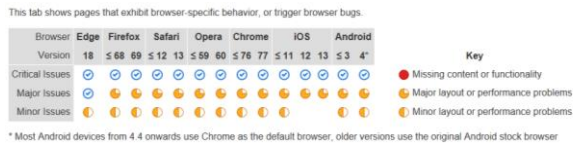
Gambar 7 Hasil Implementasi Halaman Utama *Dashboard Backoffice*

7. PENGUJIAN DAN ANALISIS

Untuk dapat menjamin sistem dapat berjalan dengan baik maka dibutuhkan validasi hasil implementasi apakah sesuai dengan kebutuhan yang telah didapat pada tahap analisa kebutuhan. Berdasarkan rumusan masalah, pengujian dilakukan menggunakan pengujian validasi pada kebutuhan fungsional sistem. Sedangkan pada kebutuhan non fungsional menggunakan pengujian kompatibilitas (*compatibility*), dan pengujian penerimaan pengguna (*user acceptance testing*).

Hasil analisis pengujian validasi menggunakan pengujian *black-box* guna mengetahui error pada kinerja aplikasi (Simanjuntak, Tolle and Dewi, 2019) menghasilkan status 100% valid, karena 25 kasus uji yang dijalankan seluruhnya menghasilkan nilai valid. Pengujian yang menggunakan metode *black-box* bertujuan untuk memvalidasi kebutuhan sistem yang terdapat pada hasil analisa kebutuhan telah terselesaikan dengan melihat kesesuaian hasil dengan hasil yang diharapkan pada kasus uji.

Pada pengujian kompatibilitas menggunakan aplikasi SortSite yang terdapat pada Gambar 8, dapat dilihat bahwa tidak ada *critical issues* yang terdapat pada aplikasi *web whitelabel PPOB*. Namun banyak terdapat *major issues* dan *minor issues* yang ditemukan. Pada seluruh jenis *browser* kecuali *microsoft edge* ditemukan *major issues*. Terdapat 4 permasalahan pada 2 file CSS, permasalahan tersebut berhubungan dengan property CSS yang tidak dapat di *support* oleh beberapa browser. Selain itu, *minor issues* ditemukan pada seluruh jenis *browser* dan ditemukan 14 permasalahan pada 2 file CSS. Keseluruhan permasalahan yang ditemukan merupakan property yang terdapat pada *framework bootstrap v4.3.1* yang sudah disesuaikan untuk *VueJS*.



Gambar 8 Hasil Pengujian Kompatibilitas Menggunakan SortSite

Sedangkan pada *user acceptance test* rata rata yang didapat dari hasil prosentase penilaian adalah 81,5% dimana pada hasil tersebut dapat disimpulkan bahwa hasil analisis pengujian UAT pada aplikasi whitelabel PPOB memiliki hasil yang Sangat Setuju.

8. PENUTUP

Berdasarkan tahapan tahapan dalam penelitian ini, pengembangan aplikasi *whitelabel PPOB* menggunakan metode *reuse oriented development* dapat disimpulkan bahwa pada tahap analisis kebutuhan didapatkan 4 tipe pemangku kepentingan, 5 permasalahan pemangku kepentingan, 7 kebutuhan pemangku kepentingan, 5 tipe pengguna sistem, dan 28 fitur dalam kebutuhan fungsionalnya. Pada spesifikasi kebutuhan sistem ditemukan 58 kebutuhan fungsional sistem dan 4 kebutuhan non-fungsional sistem yang telah diverifikasi dan divalidasi oleh pihak PT XYZ. Hasil pemodelan *use case* diagram terdapat 3 aktor yang terlibat, dan 15 *use case* beserta spesifikasinya pada aplikasi whitelabel PPOB. Sedangkan pada aplikasi *dashboard backoffice* terdapat 3 orang aktor, dan 15 *use case* beserta spesifikasinya yang terdapat pada tahapan elisitasi kebutuhan.

Selain itu, pada tahap analisis komponen dan perancangan menggunakan pemodelan *reuse oriented development* didapatkan rancangan sistem berdasarkan hasil analisis komponen, modifikasi kebutuhan dan perancangan dengan komponen ditemukan 31 komponen *reuse* dan layanan *software as a service (SAAS)* yang dapat digunakan dalam proses implementasi sistem, 3 komponen yang memerlukan modifikasi guna memenuhi kebutuhan, sebuah rancangan arsitektur sistem, beberapa rancangan interaksi objek (*sequence diagram*), rancangan diagram kelas pada 7 *microservice* beserta relasinya, rancangan *table* pada 3 buah *database* beserta relasinya, beberapa rancangan algoritma fungsi sistem, serta 28 rancangan antarmuka aplikasi 18 antarmuka *web whitelabel PPOB* dan 10 antarmuka *dashboard backoffice*.

Dan pada hasil implementasi sistem berisikan spesifikasi lingkungan implementasi, implementasi dan integrasi dari algoritma pada beberapa kelas dan fungsi yang meliputi: kelas *Response*, *API_Controller*, *API_Model*, dan *API_XYZ* yang merupakan contoh dari *reusable component* yang dibuat pada penelitian ini. Selain itu terdapat fungsi *login member*, isi saldo member, pembelian produk, dan pembayaran tagihan yang merupakan contoh implementasi kode program yang dibuat. Terdapat juga gambar hasil implementasi antarmuka aplikasi *web whitelabel PPOB* dan *dashboard backoffice* yang telah dibuat.

Terdapat juga hasil dari pengujian yang telah dilakukan pada *whitelabel PPOB* bagi mitra PT XYZ. Dimana pengujian validasi dengan metode *black-box* pada masing-masing *use case* berhasil mendapatkan tingkat validitas sebesar 100%, dan dapat diambil kesimpulan bahwa sistem sudah memenuhi kebutuhan fungsional dan siap dirilis kepada pengguna akhir. Pada pengujian kompatibilitas yang dilakukan pada kebutuhan non-fungsional terhadap *compatibility* dan *accessibility* aplikasi *web whitelabel PPOB* tidak memiliki *critical issues* yang membuat aplikasi tidak dapat digunakan. Dan pada pengujian penerimaan pengguna mendapatkan hasil rata rata keseluruhan sebesar 81,5%, sehingga dapat disimpulkan bahwa sistem yang telah dibuat sangat disetujui oleh pengguna untuk digunakan sebagai aplikasi *whitelabel PPOB* bagi mitra PT XYZ.

9. DAFTAR PUSTAKA

- Baldassarre, M. T. *et al.* (2005) ‘An industrial case study on reuse oriented development An Industrial Case Study on Reuse Oriented Development’, (January). doi: 10.1109/ICSM.2005.20.
- Fikri, A., Aknuranda, I. and Pradana, F. (2019) ‘Pengembangan Sistem Informasi Aspirasi Online Berbasis Web Menggunakan Pemodelan Reuse-Oriented Development (Studi Kasus : DPM Universitas Brawijaya)’, *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer (J-PTIHK) Universitas Brawijaya*, 3(2).
- Investopedia (2018) *White Label Product*. Available at: <https://www.investopedia.com/terms/w/w-hite-label-product.asp> (Accessed: 15

January 2018).

- Kapembe, S. and Quenum, J. (2018) 'Lihonga-a microservice-based virtual learning environment', *Proceedings - IEEE 18th International Conference on Advanced Learning Technologies, ICALT 2018*, pp. 98–100. doi: 10.1109/ICALT.2018.00030.
- Richardson, C. (2018) *What are microservices?* Available at: <https://microservices.io/> (Accessed: 24 April 2019).
- Simanjuntak, R. H., Tolle, H. and Dewi, R. K. (2019) 'Pengembangan Aplikasi Mobile Geotagging Fasilitas Tanggap Darurat Bencana Alam Menggunakan Algoritma Polylines sebagai Pencarian Rute Terdekat', 3(9), pp. 8964–8971.
- Sommerville, I. (2011) *Software Engineering*. 9th edn. Boston: Addison-Wesley.
- Techopedia (no date) *White Label*. Available at: <https://www.techopedia.com/definition/24401/white-label> (Accessed: 15 January 2018).
- ZHAO, J. T., Jiang, S. Y. and Jiang, L. Z. (2018) 'Management of API Gateway Based on Micro-service Architecture Management of API Gateway Based on Micro-service Architecture'.