

Penerapan Arsitektur *Model View Controller* dan *Framework Zend* dalam Pengembangan Aplikasi *Category Online* di PT XYZ

Bayu Ponco Raharjo¹, Felix David^{2*}

^{1,2*} Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Kota Salatiga, Provinsi Jawa Tengah, Indonesia.

Email: 672020022@student.uksw.edu¹, felix@uksw.edu^{2*}

Histori Artikel:

Dikirim 20 Februari 2024; *Diterima dalam bentuk revisi* 20 Maret 2024; *Diterima* 30 Maret 2024; *Diterbitkan* 10 Mei 2024. Semua hak dilindungi oleh Lembaga Penelitian dan Pengabdian Masyarakat (LPPM) STMIK Indonesia Banda Aceh.

Abstrak

Penelitian ini membahas penerapan arsitektur Model View Controller (MVC) dan framework Zend dalam pengembangan aplikasi Category Online di PT XYZ, perusahaan retail terkemuka di Indonesia. Tim IT PT XYZ menghadapi tantangan dalam pengembangan aplikasi dengan PHP Native, dan pemilihan Zend Framework meningkatkan efisiensi pengembangan. Aplikasi ini menangani perubahan status item secara manual di banyak cabang, yang tidak efisien. Penelitian ini berfokus pada pengembangan modul Transformation Class Product (TCP). Metode penelitian melibatkan lima tahap, dari analisis kebutuhan hingga implementasi, dengan pendekatan model prototype. Uji coba sistem menggunakan black box testing pada tahap User Acceptance Test (UAT) berhasil, menunjukkan kesuksesan fungsionalitas utama aplikasi. Dengan demikian, penerapan MVC dan Zend Framework pada aplikasi Category Online di PT XYZ terbukti efektif meningkatkan efisiensi operasional perusahaan.

Kata Kunci: Model View Controller; Zend Framework; PHP; Manajemen Status Barang.

Abstract

This research discusses the implementation of the Model View Controller (MVC) architecture and Zend Framework in developing the Category Online application at PT XYZ, a leading retail company in Indonesia. The IT team at PT XYZ faced challenges in developing applications using PHP Native, and the adoption of the Zend Framework improved development efficiency. The application handles manual changes in item status across numerous branches, which is inefficient. The study focuses on developing the Transformation Class Product (TCP) module. The research method involves five stages, from requirements analysis to implementation, using a prototype model approach. System testing, utilizing black box testing during the User Acceptance Test (UAT) phase, was successful, indicating the effectiveness of the application's core functionalities. Thus, the implementation of MVC and Zend Framework in the Category Online application at PT XYZ has proven effective in enhancing the company's operational efficiency.

Keyword: Model View Controller; Zend Framework; PHP; Item Status Management.

1. Pendahuluan

Dalam era digital, PT XYZ, sebuah perusahaan ritel terbesar di Indonesia, membutuhkan sistem terstruktur untuk efisiensi manajemen informasi status barang. Pengembangan aplikasi untuk mengelola informasi status barang menjadi langkah penting dalam meningkatkan efisiensi operasional. Hal ini karena informasi sangatlah vital dalam bisnis, tanpa informasi internal atau eksternal, manajer kesulitan membuat keputusan internal (Kustina *et al.* 2022). Seiring dengan perkembangan tren teknologi bisnis, ritel dapat memberikan pengalaman pelanggan yang berbeda. Namun, kecepatan penyampaian pembaruan ritel menentukan seberapa kompetitif pasar, menekankan perlunya sistem informasi ritel yang adaptif, Anjani (2019).

Berdasarkan hasil wawancara ditemukan sebuah permasalahan dari tim IT PT XYZ, yaitu ketika tim IT di PT XYZ masih menggunakan bahasa pemrograman PHP *Native*, mereka merasa perlu mencari cara untuk membuat pengembangan aplikasi *web* yang lebih sistematis dan mudah. Ketika aplikasi menjadi lebih kompleks dan kebutuhan akan sebuah aplikasi yang berbasis *web* itu meningkat, sangat diperlukanlah sebuah *framework* untuk membantu dalam memecahkan permasalahannya. Pengembangan aplikasi berbasis *web* menjadi lebih mudah dengan penggunaan *framework* yang dikembangkan oleh *developer*. Beberapa *framework* yang ada saat ini sudah kompatibel dengan konsep arsitektur *Model View Controller* (MVC). Arsitektur MVC ideal untuk membangun aplikasi *web* yang membutuhkan interaksi pengguna. Ini membuat aplikasi lebih mudah untuk dirancang, dipelihara, dan diperluas (Principe dan Yoon, 2015).

Terdapat banyak pilihan *web framework* seperti Zend, CodeIgniter, Laravel, dan lain sebagainya. Namun, untuk keperluan komersial di PT XYZ, pemilihan *framework* dipengaruhi oleh pertimbangan lisensi. Zend Framework dipilih karena lisensinya memenuhi kebutuhan dan dapat diimplementasikan di PT XYZ. Dibandingkan dengan PHP *Native*, penggunaan Zend Framework diharapkan dapat membawa perubahan signifikan, memberikan efisiensi dalam pengembangan aplikasi kompleks. Zend Framework, sebuah *framework open source* untuk pengembangan aplikasi *web* berbasis PHP dengan arsitektur MVC, menawarkan keandalan, keberagaman *library*, desain modern, dan kemudahan pengembangan. Dirancang sederhana, berbasis komponen, dengan ketergantungan rendah, Zend Framework mendukung opsional arsitektur MVC, serta memiliki dokumentasi dan *source code* berkualitas tinggi (Adrian, Georgiana, dan Ana-Maria, 2013). Zend Framework menerapkan konsep arsitektur MVC dalam kerangka kerja *open source* untuk pengembangan aplikasi *web* dan layanan, dengan prinsip pemrograman berorientasi objek. MVC dalam penelitian ini merujuk pada arsitektur yang memisahkan *database (model)*, tampilan (*view*), dan aliran logika (*controller*). Pendekatan ini menawarkan metode sederhana dalam pengembangan aplikasi *web*, di mana *model* mengatur logika dan kebijakan bisnis, *view* menggambarkan antarmuka pengguna, dan *controller* bertindak sebagai perantara antara keduanya.

Aplikasi Category Online di PT XYZ ini adalah sebuah aplikasi yang penting dalam mengelola informasi status *item* atau barang, termasuk aktivasi, pembaharuan, dan penghapusan. Pentingnya status ini terlihat dalam informasi pembelian dan penjualan barang di toko utama dan cabang-cabang PT XYZ. Sebelum aplikasi ini, perubahan status *item* dilakukan manual di setiap cabang, menyebabkan ketidakefisienan dalam manajemen status. Fokus penelitian adalah pada pengembangan modul *Transformation Class Product* (TCP) yang esensial dalam mengubah status *item* dari *Class Product* (CP) ke TCP. Pengembangan aplikasi diperlukan untuk mengatasi tantangan perubahan status manual di setiap cabang, memaksimalkan efisiensi operasional PT XYZ dalam mengelola informasi pembelian dan penjualan.

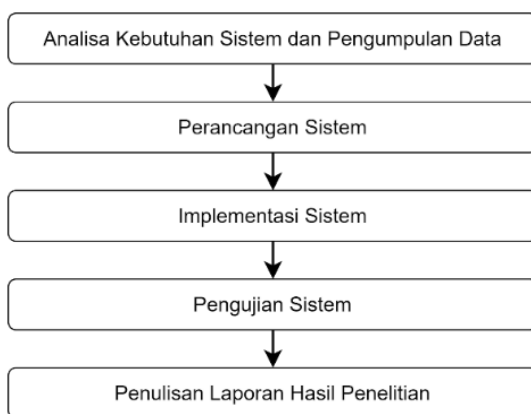
Penelitian berjudul “Penerapan *Framework* Zend dan Arsitektur *Model View Controller* dalam Perancangan Aplikasi User Manual Online” menyoroti penggunaan Zend Framework dan arsitektur MVC dengan fokus pada PHP dan MySQL, berbeda dengan penelitian ini yang menggunakan *database* Oracle. Dengan menggunakan *framework* Zend, dikatakan sistem mampu berjalan dengan cepat karena memberikan kemudahan dalam proses pengolahan data (Rahmadiansyah *et al.*, 2012). Sedangkan untuk penelitian ini akan menitikberatkan pada pengembangan aplikasi yang memiliki kegunaan yang

berbeda di industri ritel besar. Penelitian lain, berjudul “Perancangan Sistem Informasi Manajemen Distribusi Barang Pada CV Welas Asih Tasikmalaya Menggunakan Metode *Prototype*” mengeksplorasi penerapan teknologi dalam manajemen distribusi barang (Gunawan *et al.*, 2022). Penelitian tersebut mengindikasikan perlunya solusi yang lebih canggih dan terstruktur dalam manajemen informasi status barang, dan penelitian ini menerapkan teknologi yang berbeda yaitu Zend Framework untuk mengembangkan aplikasi manajemen status barang. Acuan penelitian terakhir yang berjudul “Implementasi Metode *Model View Controller* Menggunakan *Framework CodeIgniter* dalam Pengembangan Aplikasi Manajemen Depo Petikemas pada Unit Usaha Belawan *Logistics Center*” telah membuktikan efektivitas penerapan MVC dengan sebuah *framework* PHP dalam meningkatkan efisiensi operasional. Penelitian ini akan memperluas konsep tersebut dengan mengimplementasikan MVC, namun menggunakan *framework* yang berbeda, yaitu Zend Framework, untuk menjawab tantangan dalam pengembangan aplikasi Category Online di PT XYZ.

Berdasarkan penelitian dan permasalahan yang ada, penerapan *Zend Framework* pada pengembangan aplikasi Category Online di PT XYZ menjadi langkah penting dalam menjawab kebutuhan PT XYZ sebagai perusahaan ritel terbesar di Indonesia. Tujuan dari penelitian ini adalah untuk menerapkan arsitektur *Model View Controller* (MVC) dan juga *framework* Zend pada pengembangan aplikasi tersebut, membawa solusi sistematis dan efisien untuk mengelola informasi status barang di sebuah perusahaan.

2. Metode Penelitian

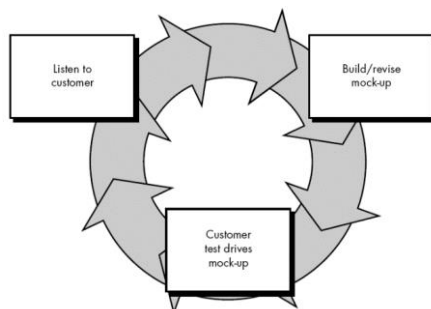
Pengembangan aplikasi Category Online ini diselesaikan dalam lima tahapan, di antaranya analisa kebutuhan sistem dan pengumpulan data, perancangan sistem, implementasi sistem, pengujian sistem, dan penulisan laporan. Tahapan penelitian ini ditunjukkan pada gambar 1.



Gambar 1. Tahapan Penelitian

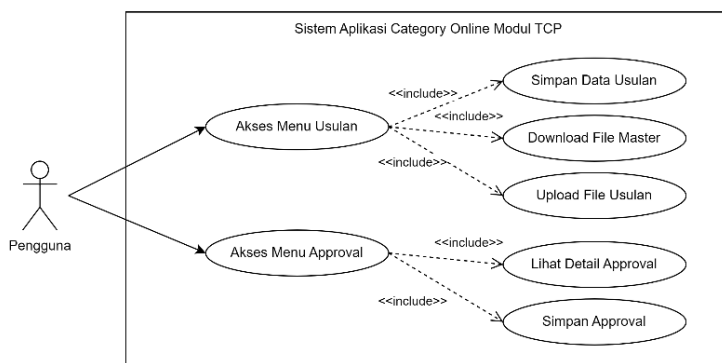
Gambar 1 di atas menjelaskan tahapan penelitian ini. Tahap pertama melibatkan analisis kebutuhan sistem dan pengumpulan data, mengidentifikasi masalah dan kebutuhan untuk aplikasi Category Online di PT XYZ melalui wawancara dan analisis. Tahap kedua, perancangan sistem, mencakup pembuatan rancangan menggunakan *Unified Modeling Language* (UML), termasuk *use case diagram*, *activity diagram*, dan model relasi *database*. Tahap ketiga, implementasi sistem, menerapkan rancangan dengan *Zend Framework*. Tahap keempat, pengujian sistem, mencari *bug* atau kekurangan aplikasi, dan melakukan perbaikan hingga memenuhi persyaratan, pada penelitian ini dilakukan pengujian *User Acceptance Test* (UAT) menggunakan metode *black box testing*. Tahap kelima, penyusunan laporan penelitian, mencakup seluruh proses pengembangan sistem dari awal hingga akhir.

Penelitian ini menerapkan pendekatan model *prototype* dalam perancangan sistem, memungkinkan pengembangan cepat untuk memverifikasi persyaratan dan keputusan desain. Pendekatan ini memfasilitasi validasi terhadap kebutuhan dan desain dengan melibatkan *prototype* sebagai perantara antara pengguna dan *developer* selama proses pengembangan sistem. Alur metode *prototype* dapat dilihat pada gambar 2.



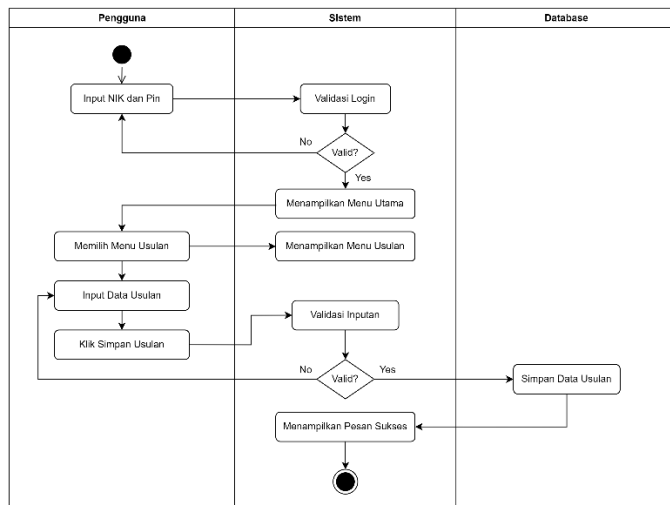
Gambar 2. Metode *Prototype Model*

Proses pengembangan dimulai dengan fase *listen to customer*, di mana data kebutuhan sistem diperoleh melalui wawancara dengan mentor di PT XYZ. Data tersebut disusun menjadi *System Design* sebelum diimplementasikan ke dalam sistem. Fase kedua melibatkan pembuatan atau penyempurnaan *mock-up* berdasarkan analisis kebutuhan, termasuk perancangan proses, basis data, dan antarmuka sistem menggunakan diagram UML. Tahap ketiga *customer test drives mock up*, melibatkan pengujian fungsionalitas sistem untuk memastikan kesesuaian dengan kebutuhan pengguna, dan proses ini berulang hingga sistem memenuhi standar dan siap diimplementasikan.



Gambar 3. *Use Case Diagram*

Gambar 3 menampilkan *Use Case Diagram* aplikasi Category Online pada modul TCP. Pengguna dapat mengakses dua menu utama, yaitu menu usulan dan *approval*. Di menu usulan, pengguna dapat menyimpan, mengunduh, dan mengunggah data usulan. Sedangkan di menu *approval*, pengguna dapat melihat detail data usulan yang membutuhkan *approval* dan melakukan *approval*.

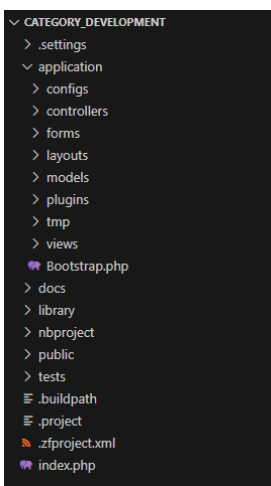


Gambar 4. Activity Diagram

Gambar 4 menampilkan *Activity Diagram* untuk proses menyimpan data usulan aktivasi atau *delete* di aplikasi *Category Online* modul TCP. Diagram ini membagi aktivitas menjadi tiga bagian utama, yaitu pengguna, sistem, dan *database*. Pengguna memulai proses dengan *login* ke dalam aplikasi, kemudian dalam menu usulan, pengguna menginputkan data-data usulan, yang kemudian diterima oleh sistem. Sistem melakukan kegiatan seperti validasi data, penyimpanan data usulan ke dalam *database*, dan memberikan konfirmasi kepada pengguna. Aktivitas tersebut mencerminkan alur proses secara visual, memberikan gambaran singkat tentang interaksi antara pengguna, sistem, dan *database* selama proses penyimpanan data usulan di aplikasi *Category Online* Modul TCP.

3. Hasil dan Pembahasan

Penelitian ini berhasil mengembangkan aplikasi *web* *Category Online* untuk manajemen status *item* dengan menggunakan *Zend Framework* dan arsitektur *Model View Controller* (MVC). Struktur MVC diilustrasikan pada gambar 5, menunjukkan penempatan *file* dan logika aplikasi yang terorganisir sesuai dengan prinsip desain MVC.



Gambar 5. Arsitektur MVC Zend pada Aplikasi

Kode Program 1. Kode Program untuk Simpan Data

```
public function simpanData()
{
    $model = new Application_Model_Tcp();
    if ($this->_request->isPost())
    {
        $pesan = '';
        $data = $this->_request->getPost();
        $save = $this->model->saveData($data);
        if($save['pesan'] == 'success') {
            $this->view->sukses = 'Proses Simpan Berhasil';
        } else {
            $pesan = 'Proses Simpan Gagal '.$save['pesan'];
        }
        $this->view->pesan = $pesan;
    }
}
```

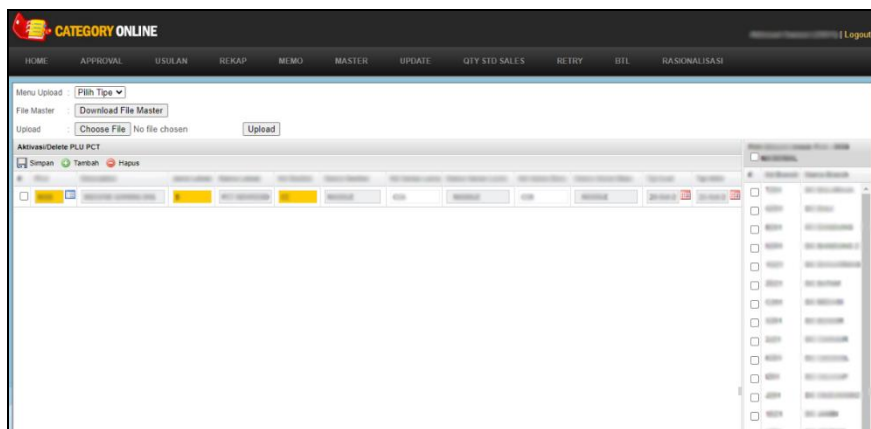
Kode Program 1 adalah bagian kode yang menangani penyimpanan data dari permintaan POST. Setelah membuat objek model `Application_Model_Tcp`, fungsi simpan data memeriksa apakah permintaan adalah metode POST. Jika benar, maka data formulir disimpan, dan metode save data dipanggil pada objek model. Hasilnya kemudian diperiksa, jika sukses, pesan sukses ditetapkan, namun jika gagal, pesan gagal beserta pesan dari model disiapkan. Pesan hasil (sukses atau gagal) disampaikan ke antarmuka pengguna untuk umpan balik terkait proses simpan data.

Kode Program 2. Kode Program untuk Insert Data ke Database

```
public function saveData($data)
{
    try {
        $this->_db->beginTransaction();
        $tipe = $data['tipe'];
        $arrNomor = array();
        foreach($data['kode'] as $idx => $kode)
        {
            $nomor = $this->getNomor(Zend_Registry::get('session')->cabang, $tipe);
            $arrNomor[] = $nomor;
            $lokasi = $data['lokasi'][$idx];
            $item = $data['item'][$idx];
            $sql = "INSERT INTO TABEL_DATA
                (NOMOR,LOKASI,KODE,ITEM,USER,TANGGAL)
                VALUES
                (:nomor,:lokasi,:kode,:item,:user,SYSDATE)";
            $stmt_ins = $this->_db->prepare($sql);
            $stmt_ins->bindParam(':nomor', $nomor);
            $stmt_ins->bindParam(':lokasi', $lokasi);
            $stmt_ins->bindParam(':kode', $kode);
            $stmt_ins->bindParam(':item', $item);
            $stmt_ins->bindParam(':user', Zend_Registry::get('session')->user);
            $stmt_ins->execute();
        }
        $this->_db->commit();
        return array('pesan'=>'success', 'nomor' => $arrNomor);
    } catch (Zend_Exception $e) {
        $this->_db->rollBack();
        return array('pesan'=>$e->getMessage());
    }
}
```

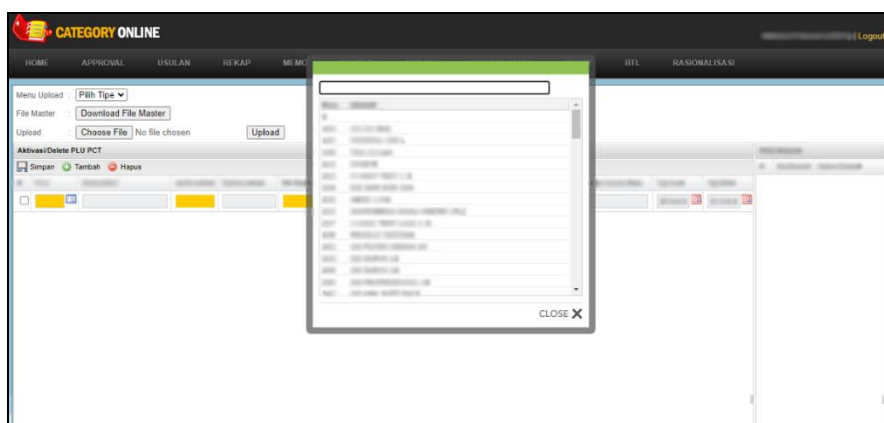
Kode program 2 terkait dengan fungsi simpan data dalam kode program 1. Fungsi save data menyimpan data formulir POST ke *database* dengan menggunakan transaksi untuk memastikan keintegritasan. Setiap elemen data diproses, nomor unik dihasilkan, dan data dimasukkan ke dalam tabel *database*. Jika sukses, Kode Program 1 memberikan pesan sukses beserta nomor yang dihasilkan dari kode program 2, jika gagal, pesan kesalahan ditampilkan. Ini memastikan penyimpanan data transaksional dalam aplikasi.

Pembahasan selanjutnya akan difokuskan pada beberapa tampilan antarmuka pengguna dari aplikasi. Setiap elemen tampilan akan diuraikan untuk memberikan pemahaman yang lebih mendalam tentang cara aplikasi berinteraksi dengan pengguna, mencerminkan prinsip-prinsip desain yang telah diterapkan dalam pengembangan aplikasi *Category Online*. Berikut adalah tampilan-tampilan dari aplikasi *Category Online*, yang dapat dilihat pada gambar 6, 7, dan 8.



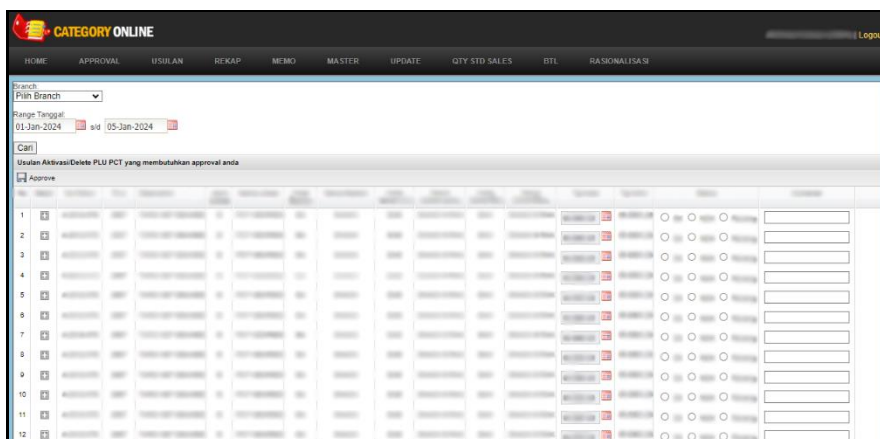
Gambar 6. Tampilan *Menu Form* Usulan Aktivasi/*Delete* PLU PCT

Gambar 6 menampilkan antarmuka menu *form* usulan aktivasi/*delete* PLU PCT di aplikasi *Category Online* dengan fitur-fitur utama. Menu ini digunakan untuk aktivasi, pembaharuan (*update*), dan penghapusan (*delete*) *item* atau barang dalam modul TCP. Fitur-fitur melibatkan unduh *file master* (.xls), unggah *file master* yang diisi oleh pengguna, tambah data baru, hapus data terpilih, dan simpan data yang diinput. Terdapat pula fitur *List of View* (LOV) untuk memilih nilai dari *database* dan elemen kalender untuk memilih tanggal dengan batasan waktu. Keseluruhan tampilan memberikan gambaran detail mengenai interaksi pengguna dengan aplikasi, mencerminkan prinsip-prinsip desain dalam pengembangan aplikasi *Category Online*.



Gambar 7. Tampilan *List of View* pada Form Usulan

Gambar 7 menunjukkan tampilan dari fitur *List of View* (LOV) pada form Usulan aplikasi *Category Online*. LOV ini berisi *field-field* kode dan deskripsi yang diambil dari *query database*, mempermudah pengguna dalam pemilihan nilai. Pada *window* LOV ini juga terdapat *field search data* yang memungkinkan pengguna melakukan pencarian data secara spesifik sesuai keinginan. Fitur ini dirancang untuk memberikan pengalaman pengguna yang lebih efisien dan akurat dalam menemukan dan memilih nilai dari *database*, sesuai dengan prinsip-prinsip desain yang telah diimplementasikan dalam pengembangan aplikasi *Category Online*.



Gambar 8. Tampilan *List Approval* Usulan Aktivasi/ Delete PLU PCT

Gambar 8 menunjukkan tampilan dari *list approval* usulan aktivasi/ delete PLU PCT aplikasi *Category Online*. *List* ini merupakan daftar usulan aktivasi/ delete PLU PCT yang telah di-input dan membutuhkan proses persetujuan. Pengguna dapat melakukan filter berdasarkan cabang dan rentang tanggal untuk mempermudah pencarian. Dilengkapi juga dengan tombol "Approve" untuk memudahkan proses persetujuan. Setiap *item* dalam *list* memiliki detail masing-masing, memberikan informasi rinci terkait usulan aktivasi atau penghapusan PLU PCT. Pengujian sistem yang telah dilakukan pada aplikasi *Category Online* adalah *User Acceptance Test* (UAT) dengan menggunakan metode *black box testing*. UAT merupakan tahap pengujian yang bertujuan untuk memastikan bahwa sistem yang telah dikembangkan sesuai dengan kebutuhan dan harapan pengguna. *Black box testing* sendiri dapat diartikan sebagai suatu metode pengujian yang memeriksa apakah seluruh fitur perangkat lunak beroperasi dengan benar sesuai dengan kebutuhan fungsional yang telah ditetapkan [13]. *Black box testing* pada pengujian ini dilakukan oleh tim *IT Quality Assurance* di PT XYZ untuk menguji fungsionalitas sistem tanpa memperhatikan struktur internal kode. Detail hasil dari pengujian *black box testing* dapat dilihat pada tabel 1 di bawah ini.

Tabel 1. Hasil Pengujian *Black Box* Aplikasi *Category Online*

Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
Tambah <i>form</i> pengisian usulan	Muncul <i>form</i> untuk pengisian usulan aktivasi/ delete PLU	Berhasil menambahkan <i>form</i> pengisian usulan aktivasi/ delete PLU	Valid
Hapus <i>form</i> pengisian usulan	<i>Form</i> pengisian yang dipilih akan terhapus dari tampilan	Berhasil menghapus <i>form</i> pengisian yang dipilih dari tampilan	Valid
Simpan data usulan	Data usulan akan tersimpan dengan benar di <i>database</i>	Berhasil menyimpan data usulan dengan benar di <i>database</i>	Valid
Mengisi <i>form</i> menggunakan LOV	<i>Form</i> terisi dengan benar sesuai dengan data yang dipilih dari LOV	Berhasil mengisi <i>form</i> dengan benar sesuai data yang dipilih di LOV	Valid
<i>Search</i> data pada LOV	Data terfilter dengan benar berdasarkan <i>keywords</i> dari <i>user</i>	Berhasil filter data dengan benar berdasarkan <i>keywords</i> dari <i>user</i>	Valid
<i>Download file master</i>	<i>File master</i> dengan format .xls yang berisi tabel usulan	Berhasil mengunduh <i>file master</i> dengan format .xls dan dapat	Valid

	dapat diunduh dan dibuka	dibuka dengan isi tabel yang sesuai	
Upload file yang berisi data usulan	File .xls dapat di-upload dan form langsung terisi sesuai dengan data yang ada di file tersebut	Berhasil upload file dengan format .xls dan form langsung terisi sesuai dengan data yang ada di file tersebut	Valid
Lihat data usulan pada menu approval	Data usulan yang telah disimpan dapat dilihat pada menu list approval	Berhasil melihat data usulan yang telah disimpan pada menu list approval	Valid
Approve data usulan	Data yang di-approve dapat tersimpan di database dan hilang dari list approval	Berhasil menyimpan data yang sudah di-approve ke dalam database dan hilang dari list approval	Valid
Filter data menggunakan cabang dan tanggal	Data yang muncul dapat terfilter sesuai dengan kriteria yang dipilih user	Berhasil filter dan menampilkan data sesuai dengan kriteria yang dipilih user	Valid

Berdasarkan hasil pengujian pada tabel 1, dapat diambil kesimpulan bahwa fungsionalitas utama dari modul TCP aplikasi Category Online telah diuji dengan berhasil. Seluruh skenario pengujian berhasil dilakukan sesuai dengan hasil yang diharapkan. Kesimpulan ini menunjukkan bahwa fitur-fitur kunci aplikasi telah melewati pengujian *black box* secara positif, memvalidasi keandalan dan keakuratan fungsionalitas aplikasi *Category Online* terutama modul TCP pada tahap *User Acceptance Test* (UAT). Pemohon juga telah mengirimkan dokumen *User Acceptance Test* yang menyatakan bahwa *user* telah menguji semua tugas dan aktivitas penggunaan fitur dan fungsi baru, dan menyatakan bahwa *project* aplikasi *Category Online* pada modul TCP ini telah sesuai dengan kebutuhan *user* berdasarkan tujuan dan ketentuan yang telah ditetapkan. Dengan demikian, aplikasi dapat dianggap siap untuk penggunaan yang lebih luas, dan terbukti berhasil meningkatkan efisiensi operasional perusahaan terutama dalam manajemen informasi status *item* atau barang.

4. Kesimpulan

Berdasarkan hasil dari penelitian dan pembahasan yang telah dilakukan, dapat disimpulkan bahwa dengan menerapkan arsitektur *Model View Controller* (MVC) dan memilih *Zend Framework* sebagai *framework* pengembangan, penelitian ini berhasil mengatasi kendala pengembangan aplikasi *web* di PT XYZ. Melalui pengujian *User Acceptance Test* (UAT) dengan metode *black box testing*, aplikasi *Category Online*, khususnya modul *Transformation Class Product* (TCP), terbukti berhasil dan siap untuk penggunaan lebih luas. Kesuksesan ini menandai peningkatan efisiensi operasional PT XYZ dalam manajemen informasi status *item* atau barang, menghilangkan proses manual yang tidak efektif, dan membuktikan bahwa beralih ke *Zend Framework* memberikan solusi yang signifikan dalam pengembangan aplikasi yang kompleks. Untuk pengembangan sistem selanjutnya dapat melibatkan pemeliharaan reguler dan pembaruan teknologi. Juga dapat mengeksplorasi pengembangan modul-modul lain di aplikasi *Category Online* ini, sehingga dapat memperluas fungsionalitas dan memberikan nilai tambah bagi PT XYZ dalam mengoptimalkan operasional bisnisnya.

5. Daftar Pustaka

- Abimanyu, A. J. (2020). *Penerapan Framework Zend dan Arsitektur Model View Controller dalam Perancangan Aplikasi User Manual Online: Studi Kasus PT. Sumber Alfaria Trijaya Tbk* (Doctoral dissertation, Program Studi Teknik Informatika FTI-UKSW).
- Adrian, L., Georgiana, D. E., & Ana-Maria, N. (2013). Zend Framework. *Risk in Contemporary Economy*, 115-122.
- Anjani, R. G. (2019). Peran Sistem Informasi Terhadap Operasional Retail. *Jurnal Ekonomi Manajemen Sistem Informasi*, 1(1), 79-86. DOI: <https://doi.org/10.31933/jemsi.v1i1.49>.
- Brown, S. K., Lo, N., & Allen, R. (2008). *Zend framework in Action*. Simon and Schuster.
- Fahrezi, A., Salam, F. N., Ibrahim, G. M., Syaiful, R. R., & Saifudin, A. (2022). Pengujian Black Box Testing pada Aplikasi Inventori Barang Berbasis Web di PT. AINO Indonesia. *LOGIC: Jurnal Ilmu Komputer dan Pendidikan*, 1(1), 1-5.
- Gunawan, D., Alfarizi, S., Ichsan, N., Basri, H., & Mulyawan, A. R. (2022). Implementasi Metode Prototype Dalam Perancangan Sistem Informasi Upah Pasang Material Konstruksi. *Bina Insani Ict Journal*, 9(1), 73-82. DOI: <https://doi.org/10.51211/biict.v9i1.1790>.
- Kurniyanti, V. A., & Murdiani, D. (2022). Perbandingan Model Waterfall Dengan Prototype Pada Pengembangan System Informasi Berbasis Website. *Jurnal Syntax Fusion*, 2(08), 669-675. DOI: <https://doi.org/10.54543/fusion.v2i08.210>.
- Kustina, K. T., Nurhayati, M., Pratiwi, E., Lesi Hertati, S. E., CAPM, A. C. C. A. C., Qodari, A., ... & Abdul Munim, S. E. (2022). *Sistem informasi manajemen*. Cendikia Mulia Mandiri.
- Nugraha, W. F., Adiwisastro, M. F., & Bahri, S. (2023). Perancangan Sistem Informasi Manajemen Distribusi Barang Pada CV Welas Asih Tasikmalaya Menggunakan Metode Prototype. *Indonesian Journal on Software Engineering (IJSE)*, 9(2), 136-145.
- Porebski, B., Przystalski, K., & Nowak, L. (2011). *Building PHP Applications with Symfony, CakePHP, and Zend Framework*. John Wiley and Sons.
- Principe, M., & Yoon, D. (2015). A web application using MVC framework. In *Proceedings of the International Conference on e-Learning, e-Business, Enterprise Information Systems, and e-Government (EEE)* (p. 10). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp).
- Rahmadiansyah, D., Irwan, D., Sekolah, D., & Teknik, T. (2012). Implementasi Metode Model View Controller Menggunakan Framework Code Igniter dalam Pengembangan Aplikasi Manajemen Depo Petikemas pada Unit Usaha Belawan Logistics Center. In *Seminar Nasional Teknologi Informasi dan Komunikasi (SNASTIKOM)* (pp. 1-6).
- Salam, I. A., Prihandani, K., & Purnamasari, I. (2023). RANCANG BANGUN APLIKASI PROFIT PENJUALAN MOTOR BERBASIS DESKTOP KONSEP ARSITEKTUR MODEL VIEW CONTROLLER (MVC). *Jurnal Informatika dan Teknik Elektro Terapan*, 11(3s1). DOI: <http://dx.doi.org/10.23960/jitet.v11i3s1.3495>.