

Pengembangan Sistem Informasi Koperasi Penjualan dan Simpan Pinjam berbasis *Website* pada Primer Koperasi Angkatan Laut Akademi Angkatan Laut (Primkopal AAL) Surabaya

Rangga Baghas Nugroho¹, Nurudin Santoso², Denny Sagita Rusdianto³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹ranggabn@student.ub.ac.id, ²nurudin.santoso@ub.ac.id, ³denny.sagita@ub.ac.id

Abstrak

Primkopal AAL Surabaya merupakan koperasi penyedia kebutuhan dari anggota AAL Surabaya. Dalam menjalankan seluruh fungsi, pengurus Primkopal AAL melakukan pencatatan data yang dimulai dari pencatatan data anggota, simpanan, penjualan, pinjaman, dan kredit. Seluruh proses tersebut dilakukan secara manual dimana hal tersebut dinilai kurang efektif dan efisien dikarenakan dalam prakteknya, membutuhkan waktu yang cukup lama saat pengajuan pinjaman dan kredit, sulitnya penyampaian informasi kepada anggota, banyaknya kesalahan pencatatan data, dan banyak hilangnya data yang disimpan oleh pengurus koperasi. Berdasarkan permasalahan diatas, penulis mengusulkan sebuah sistem yang dapat membantu melakukan penyampaian informasi dan pembukuan dari proses penjualan dan simpan bagi pihak pengurus Primer Koperasi Angkatan Laut AAL Surabaya. Sistem yang akan diimplementasikan memiliki fungsi untuk melakukan pendaftaran anggota, penjualan barang, simpanan sukarela, kredit, pinjaman, dan penyampaian informasi alur seluruh proses. Implementasi sistem akan menggunakan metode SDLC *Waterfall* dengan bahasa pemrograman Javascript pada *framework* React JS pada *frontend* dan Express JS pada *backend* dengan database *MySQL*. Sistem ini akan diuji dengan menggunakan metode pengujian *unit testing*, *integration testing*, *validation testing*, dan *compatibility testing*.

Kata kunci: sistem informasi koperasi, pembukuan, penjualan, simpan, pinjam, sistem informasi berbasis web, react js, express js

Abstract

Primkopal AAL Surabaya is a cooperative that provides the necessities of AAL Surabaya. In carrying out all functions, the Primkopal AAL management records data starting from recording member data, deposits, sales, loans, and credits. The whole process is done manually which is considered less effective and efficient because in practice, it takes a long time when applying for loans and credits, it is difficult to convey information to members, there are many data recording errors, and a lot of data loss is stored by the cooperative management. Based on the problems above, the author proposes a system that can help deliver information and bookkeeping from the sales and storage process for the primary management of the Surabaya AAL Navy Cooperative. The system that will be implemented has the function of registering members, selling goods, voluntary savings, credits, loans, and delivering information on the flow of the entire process. The system implementation will use the SDLC Waterfall method with the Javascript programming language on the React JS framework on the frontend and Express JS on the backend with a MySQL database. This system will be tested using unit testing, integration testing, validation testing, and compatibility testing methods.

Keywords: cooperative information system, bookkeeping, selling, saving, borrowing, web-based information system, react js, express js

1. PENDAHULUAN

Berdirinya Primkopal AAL sendiri memiliki tujuan untuk meningkatkan

kesejahteraan anggota AAL dengan pelayanan prima. Dengan adanya tujuan tersebut, Primkopal AAL memiliki misi yang diciptakan untuk mencapai tujuan tersebut. Salah satu misi

dari Primkopal AAL adalah menjalankan manajemen organisasi yang transparan dan akuntabel dengan didukung sistem informasi yang handal. Saat ini, Primkopal AAL mengandalkan sistem pengarsipan yang bersifat manual dan sistem informasi yang berupa media cetak. Namun dengan kondisi sistem tersebut, masih menimbulkan beberapa masalah sebagaimana yang telah didapatkan dari hasil wawancara dengan pengelola koperasi. Berdasarkan wawancara tersebut, masalah yang terjadi diantaranya sering terjadinya kesalahan input data ke dalam pembukuan, proses pengajuan pinjaman dan kredit yang memakan waktu hingga 3 – 4 hari, penyampaian informasi menggunakan media cetak sehingga anggota koperasi harus datang terlebih dahulu ke koperasi untuk mendapatkan informasi tersebut, sering terjadinya kendala buku tabungan hilang atau rusak yang mengakibatkan anggota koperasi harus membeli kembali buku tabungan tersebut dan pencarian data yang lama ketika anggota ingin melakukan transaksi dengan tabungannya.

Masalah – masalah di atas serupa dengan masalah yang dialami beberapa penelitian sejenis. Penelitian – penelitian tersebut dilakukan oleh Diah Puspitasari (2015), Indra Griha Tofik Isa dan George Pri Hartawan (2017), dan Juniarni Soklastika Maruao (2019) dimana penelitian – penelitian tersebut memiliki solusi yang sama yaitu pembuatan sistem berbasis web. Dari sistem yang dihasilkan penelitian sejenis, perbedaan sistem pada skripsi kali ini yaitu terdapat pada pembuatan buku tabungan digital yang tidak ada pada penelitian sebelumnya, mempercepat proses transaksi dengan adanya halaman Administrator, dan adanya penyampaian informasi suatu barang yang tersedia atau tidak. Sehingga berdasarkan permasalahan yang ada, skripsi ini memberikan solusi yaitu pengembangan sistem informasi koperasi penjualan dan simpan pinjam berbasis website Primkopal AAL dengan menggunakan metode Waterfall. Sistem ini diharapkan dapat mempermudah pengguna dalam mengakses kegiatan koperasi baik dari sisi anggota koperasi, pengelola koperasi, dan pihak terkait lainnya.

2. LANDASAN KEPUSTAKAAN

2.1. Kajian Pustaka

Penelitian pertama adalah penelitian mengenai rancang bangun sistem informasi koperasi simpan pinjam karyawan berbasis web.

Penelitian ini dilakukan karena adanya sebuah masalah yaitu proses pengelolaan data simpan pinjam yang masih dilakukan secara manual sehingga anggota harus secara langsung datang ke koperasi untuk bertanya terkait informasi mengenai simpan pinjam dan kesulitan dalam melakukan pendaftaran anggota karena belum bisa dilakukan secara online. Hal tersebut mengakibatkan kurangnya efektifitas dan efisien dari proses Kelola data. Sehingga penelitian ini menawarkan solusi berupa rancang bangun sistem informasi koperasi simpan pinjam karyawan berbasis web dengan menggunakan metode waterfall (Puspitasari, 2015).

Penelitian kedua adalah penelitian mengenai perancangan aplikasi koperasi simpan pinjam berbasis web pada Koperasi Mitra Setia. Pada penelitian ini terdapat beberapa masalah yang dibahas yang diantaranya pengurus harus melakukan pencatatan pada beberapa buku untuk pendataan transaksi sehingga memerlukan waktu yang tidak sedikit ketika melakukan pencarian data. Hal tersebut mengakibatkan tidak efektifnya waktu dalam proses pencatatan dan perekapan data. Penelitian ini menawarkan solusi berupa perancangan aplikasi koperasi simpan pinjam berbasis web dengan metode perancangan berbasis objek (Indra Griha Tofik Isa, 2017).

Penelitian ketiga adalah penelitian mengenai penerapan model waterfall pada pengembangan sistem informasi koperasi simpan pinjam cipta usaha mandiri berbasis web pada Koperasi Cipta Usaha Mandiri. Penelitian ini menghasilkan beberapa laporan yaitu laporan data anggota, laporan simpanan, laporan pinjaman uang, dan laporan sisa angsuran uang anggota koperasi. Sistem ini diharapkan dapat memudahkan proses pekerjaan dan menyelesaikan masalah – masalah yang terjadi di Koperasi Simpan Pinjam Cipta Usaha Mandiri (Anik Sri Wahyuningsih, 2019).

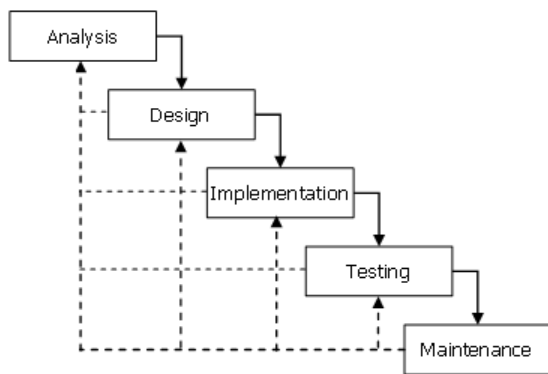
2.2. Pendekatan Berorientasi Objek

Pada hal mengembangkan sistem menggunakan pendekatan berorientasi objek akan memiliki perbedaan dengan pengembangan sistem yang menggunakan memandang bahwa tidak ada hubungan diantara fungsi dan data pada perangkat lunak atau yang biasa disebut pendekatan konvensional. Pendekatan konvensional akan lebih memiliki fokus terhadap fungsi walaupun terdapat beberapa pendekatan yang memiliki juga terhadap data yang menitik beratkan pada informasi dan

pemakaian basis data. Sedangkan, mengkombinasikan data dan fungsionalitas merupakan fokus dari pendekatan berorientasi objek. Konsep objek yang mengandung unsur data beserta fungsinya merupakan pengertian dari berorientasi objek. Pendekatan berorientasi objek adalah pendekatan yang bekerja dengan memandang persoalan menggunakan model-model yang melakukan penggabungan antara struktur data dan perilaku dari entitas berdasarkan konsep objek yang diorganisasikan. Pada dasarnya, kumpulan objek diskrit yang saling berkomunikasi, bekerja sama, dan berinteraksi menuju tujuan khusus adalah pengertian dari organisasi perangkat lunak (Eniyati, 2006).

2.3. SDLC Waterfall

Metode Waterfall merupakan metode pengembangan perangkat lunak berbentuk seperti air terjun yang dimana prosesnya bergerak mengalir dari atas ke bawah. Setiap tahapan metode Waterfall wajib terselesaikan sepenuhnya secara berurutan agar dapat berpindah ke tahap selanjutnya. Metode Waterfall ini terdiri dari lima tahap yaitu analisis, perancangan, implementasi, pengujian, dan pemeliharaan (Bassil, 2012).



Gambar 1. Tahapan SDLC Waterfall

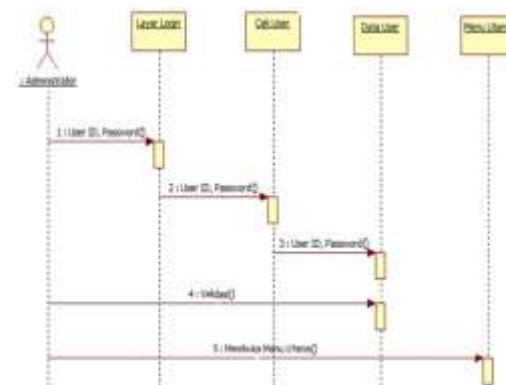
2.4. Use Case Diagram

Pada tahun 1986, Ivar Jacobson mengusulkan diagram bernama *use case*. *Use case* berarti suatu metode yang dilakukan untuk identifikasi, menjelaskan, dan menyusun syarat-syarat yang harus ada pada sebuah sistem dalam proses analisis sistem. Diagram use case termasuk dalam UML (*Unified Modeling Language*) yaitu suatu notasi standar yang berfungsi sebagai pemodelan suatu objek dan sistem pada dunia nyata. Diagram *use case* merupakan sub kelas diagram perilaku dalam *Unified Modeling Language* (UML). *Use Case Diagram* juga merupakan salah satu dari *Object*

Oriented Diagrams yang berarti bahwa use case menunjukkan bagaimana sistem berinteraksi dengan entitas eksternal. Jadi, *use case* tidak menjelaskan tentang detail bagaimana sistem berperilaku secara internal dan bagaimana lingkungan eksternal diatur (Aleryani, 2016).

2.5. Sequence Diagram

Sequence diagram adalah diagram yang menggambarkan interaksi antar objek di suatu sistem yang dihubungkan oleh suatu *message* dan ditampilkan berdasarkan waktu eksekusi. Sequence diagram terdiri atas 2 dimensi yaitu dimensi vertikal (waktu) dan dimensi horizontal (objek-objek pada *sequence*). Suatu pengembalian atau respons dari sebuah *event* digambarkan melalui skenario atau rangkaian langkah-langkah. Penggambaran *sequence diagram* berawal dari apa yang menyebabkan aktivitas yang sedang berjalan itu terjadi, proses dan perubahan-perubahan yang telah terjadi secara internal, dan hasil apa yang dikeluarkan (Mahdiana, 2011).



Gambar 2. Sequence Diagram

2.6. Pengujian

Sebuah perangkat lunak yang telah atau sedang dibuat dapat dipastikan berfungsi sesuai dengan yang diharapkan maka diperlukan melakukan suatu kegiatan yaitu pengujian *software*. Agar kesalahan atau kekurangan pada *software* dapat terdeteksi sejak saat pengembangan dan dapat dibenarkan secepat mungkin, maka dilakukanlah suatu pengujian yang seharusnya disediakan oleh pengembang dari sistem tersebut. Jaminan kualitas perangkat lunak adalah hal yang penting dalam membangun sebuah *software*. Sehingga, pengujian atau *testing* sendiri merupakan elemen yang sangat berpengaruh untuk mewujudkannya. Tanpa adanya pengujian maka siklus pengembangan *software* akan tidak lengkap, karena pengujian tidak dapat

dipisahkan dari siklus tersebut seperti halnya analisis, perancangan, dan implementasi (Mustaqbal, et al., 2015).

2.7. React Js

ReactJS adalah *library* JavaScript yang digunakan untuk mengembangkan komponen antarmuka pengguna modular. React pada dasarnya dapat digunakan dalam pengembangan aplikasi berbasis web yang kompleks dengan kelebihan dapat mengubah datanya tanpa memuat ulang halamannya. React berfungsi sebagai *View (V)* pada konsep *Model-View-Controller (MVC)*. Selain itu, React mengabstraksi Model Objek Dokumen (DOM), sehingga menawarkan pengalaman pengembangan aplikasi yang sederhana, berkinerja, dan kuat. Pada sisi *server*, React sebagian besar di *render* menggunakan NodeJS, sedangkan untuk aplikasi *mobile* dikembangkan menggunakan React Native. React terbukti jauh lebih mudah daripada pengikatan data tradisional karena pada pengimplementasiannya React menggunakan aliran data searah sehingga menyederhanakan *boilerplate*. Dalam React, model objek dokumen yang tersedia juga lebih efisien dan ringan. Lalu ReactJS juga memiliki sifat yang mudah dan tidak kompleks sehingga memungkinkan seseorang dengan cepat merasa nyaman dengan *framework*. Hal tersebut merupakan faktor paling kuat yang menjadikan *framework* ini lebih menonjol daripada banyak *framework* lainnya di dunia. Fitur virtual DOM merupakan salah satu dasar yang menjadikan React sangat efisien untuk digunakan (Aggarwal, 2018).

2.8. Node Js

NodeJs merupakan teknologi perangkat lunak yang memungkinkan untuk membuat *server web* sendiri dan membangun aplikasi web di atasnya. NodeJs sendiri bukan *server web* dan juga bukan Bahasa, tetapi merupakan *library server HTTP* bawaan, dimana pengembang tidak perlu menjalankan program secara terpisah seperti Apache atau IIS. Hal ini berdampak baik dengan memberikan kendali yang lebih besar terhadap kerja *server web*, tetapi meningkatkan kerumitan untuk mengaktifkan dan menjalankannya. Pada dasarnya saat membuat aplikasi dengan NodeJs pengembang melakukan konfigurasi *server Node.js*. hal ini berbeda dengan aplikasi yang dikembangkan menggunakan bahasa PHP. Dengan PHP, pengembang dapat dengan mudah menemukan *host web sharedserver* yang digunakan untuk

menjalankan Apache dan mengirim file melalui FTP. Hal itu dapat berjalan karena *host web* telah mengkonfigurasi Apache untuk seluruh penggunaanya (Bangare, et al., 2016). NodeJs juga biasa disebut sebagai perangkat lunak yang dirancang dalam sintaks bahasa pemrograman Javascript yang bertujuan untuk pengembangan aplikasi berbasis web. JavaScript dapat diimplementasikan sebagai bahasa pemrograman yang dapat dijalankan pada sisi server apabila NodeJs juga ditautkan kedalam sistem yang sedang dikembangkan. Pada sistem operasi Windows, Mac OS X dan Linux, Node.js dapat diimplementasikan tanpa perlu ada perubahan kode program. NodeJs bersifat asinkronis atau *non blocking*. Node.js juga berjalan dengan menggunakan basis *event (event-driven)* (Firdaus, et al., 2019).

3. METODOLOGI PENELITIAN

Pada bagian metodologi penelitian terdapat penjelasan mengenai tahapan dari penelitian mengenai Sistem Informasi Koperasi Penjualan dan Simpan Pinjam pada Primkopal AAL Surabaya. Dalam pengembangannya, penelitian ini mengadopsi metode dari SDLC Waterfall. Sesuai dengan metode tersebut, penelitian ini dibangun mengikuti alur sehingga penelitian ini dikerjakan secara teratur dan berurutan dimulai dari studi literatur, analisis kebutuhan, perancangan, implementasi, pengujian, dan pengambilan kesimpulan dan saran. Seperti pada tahapan tersebut, maka terdapat diagram alir yang menggambarkan tahapan dari penelitian ini. Gambar 3 menunjukkan diagram alir penelitian Pembangunan Sistem Informasi Koperasi Penjualan dan Simpan Pinjam.



Gambar 3. Metodologi Penelitian

4. ANALISIS KEBUTUHAN

4.1. Deskripsi Sistem

Sistem Informasi Koperasi Penjualan dan Simpan Pinjam terdiri dari lima aktor, yaitu *user*, anggota, *admin*, kaprim, dan kasatker. Aktor adalah seluruh pihak yang dapat menggunakan sistem. Sistem Informasi Koperasi ini juga terdapat beberapa fungsi utama yang akan ada pada sistem, bagian-bagian tersebut adalah penjualan, peminjaman, simpanan, kredit, dan pembukuan.

4.2. Elisitasi Kebutuhan

Proses elisitasi kebutuhan sistem ini dilakukan dengan cara melakukan wawancara kepada pengurus dari Primer Koperasi Angkatan Laut AAL Surabaya. Wawancara tersebut dilakukan untuk menggali informasi terkait kendala-kendala yang terdapat pada proses bisnis Primkopal saat menggunakan sistem yang sebelumnya sudah ada. Selain itu, wawancara juga dilakukan untuk mengetahui kebutuhan apa saja yang harus terdapat pada sistem yang akan dibangun. Kebutuhan tersebut didiskusikan dengan pengurus dan Ketua Primkopal agar nantinya sesuai dengan yang dibutuhkan oleh pengurus maupun anggota koperasi.

Tabel 1. Identifikasi Aktor

No	Aktor	Deskripsi
1	User	User adalah aktor yang belum dikenali oleh sistem. User harus melakukan pendaftaran dan login agar sistem dapat mengenali peran dari user tersebut.
2	Admin	Admin adalah pengurus Primkopal AAL Surabaya yang bertugas sebagai pengelola dari kegiatan yang ada di koperasi.
3	Anggota	Anggota adalah aktor yang dapat mengakses fungsi simpanan, pinjaman, kredit, buku tabungan, dan catatan pinjaman dan kredit.
4	Kaprim	Kaprim (Ketua Primkopal) adalah aktor yang dapat mengakses dashboard persetujuan melakukan pinjaman dan kredit.
5	Kasatker	Kasatker (Ketua Satuan Kerja) adalah aktor yang dapat mengakses dashboard persetujuan melakukan pinjaman.

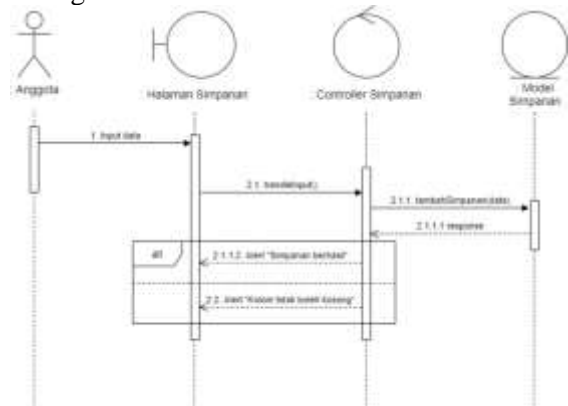
Pada proses analisis kebutuhan penelitian ini menghasilkan kebutuhan fungsional yang

berjumlah 46 kebutuhan dan kebutuhan non fungsional yang berjumlah 1 kebutuhan. Kebutuhan-kebutuhan tersebut nantinya akan menjadi landasan saat proses implementasi sistem dilakukan.

5. PERANCANGAN SISTEM

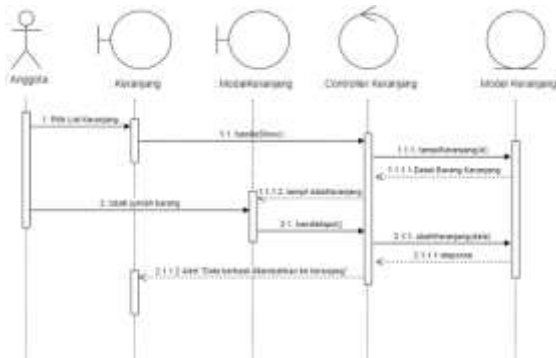
5.1. Sequence Diagram

Sequence diagram menjelaskan objek-objek beserta interaksinya yang terdapat pada sistem. Sequence diagram berasal dari hasil analisis kebutuhan yang telah dituliskan sebagai use case scenario. Setiap sequence diagram akan menggambarkan satu use case scenario yang telah dituliskan sebelumnya. Pada bagian ini, akan diberikan dua sequence diagram sistem sebagai contoh, yaitu sequence diagram fungsi tambah simpanan, ubah keranjang, dan tambah barang.



Gambar 4. Sequence Diagram Tambah Simpanan

Gambar 4 menggambarkan alur untuk menambah data simpanan. Aktor harus login terlebih dahulu agar dikenali sebagai anggota, selanjutnya mengisi data pada formulir tambah simpanan. Pada alternatif, ketika terdapat kolom data yang belum diisi, maka sistem akan menampilkan pesan peringatan bahwa kolom tidak boleh kosong. Lalu, anggota akan menekan tombol simpan untuk mengirimkan data. Selanjutnya data tersebut dikirimkan controller simpanan melalui tambahSimpanan() yang menggunakan method post untuk dikirimkan ke database simpanan. Sistem akan mengirimkan response bahwa data berhasil ditambahkan yang akan ditampilkan dalam bentuk alert.



Gambar 5. Sequence Diagram Ubah Keranjang

Gambar 5 menggambarkan alur untuk mengubah data keranjang. Aktor harus login terlebih dahulu agar dikenali sebagai anggota dan telah melakukan tahap menambahkan barang ke keranjang. Anggota memilih list keranjang mana yang ingin diubah jumlahnya. Selanjutnya, sistem akan mengambil data barang yang dipilih sesuai dengan id keranjangnya dan ditampilkan pada modal keranjang. Anggota akan mengubah data jumlah barang yang ingin dibeli dan menekan tombol simpan. Selanjutnya data tersebut dikirimkan ke controller keranjang yang dimana akan disimpan ke dalam database keranjang melalui `ubahKeranjang()` menggunakan `method put`. Setelah data keranjang berhasil disimpan pada database, sistem akan mengirimkan response bahwa data berhasil ditambahkan yang akan ditampilkan dalam bentuk alert.

5.2. Class Diagram

Perancangan class diagram dari Sistem Informasi Koperasi Penjualan dan Simpan Pinjam Primkopal AAL Surabaya memiliki kelas view terdiri dari 31 kelas yaitu, H_Masuk, H_Daftar, H_Daftar Anggota, H_Daftar Penjualan, H_Keranjang, H_Ubah Keranjang, H_Ubah Anggota, H_Tabungan, H_Daftar Simpanan, H_Ubah Simpanan, H_Simpan, H_Daftar Barang, H_Toko, H_Ubah Barang, H_Pinjam, H_Daftar Pinjaman, H_Data Pinjaman, H_Ubah Pinjaman, H_Ubah Kredit, H_Data Kredit, H_Daftar Kredit, H_Kredit, H_Daftar Pengambilan, H_Informasi Lupa Password, H_Informasi Pembelian, H_Informasi Simpanan, H_Informasi Kredit, H_Informasi Pinjaman, H_Home, H_Profil dan H_Komplain. Pada kelas controller, terdapat 9 kelas yaitu Kredit, Pinjaman, Barang, Keranjang, Anggota, Simpanan, Pengambilan, Komplain dan Penjualan. Kelas view pada diagram terhubung dengan kelas controller dan kelas controller terhubung dengan kelas controller lainnya menggunakan relasi asosiasi.

5.3. Perancangan Basis Data

Relational Model Sistem Primkopal memiliki 15 tabel yang terdiri dari tabel User, Akses_token, Role, Penjualan, Barang, Keranjang, Status, Peraturan, Kategori, Cicilan, Pengambilan, Komplain, Kredit, Pinjaman, dan Simpan. Dalam Relational Model Sistem Primkopal terdapat 2 relasi, yaitu relasi one to one dan one to many.

5.4. Perancangan Antarmuka

Perancangan antarmuka menggambarkan beberapa desain antarmuka yang akan diimplementasikan pada sistem Primkopal. Beberapa perancangan yang telah dibuat antara lain adalah perancangan antarmuka halaman simpan, halaman toko, modal keranjang, dan halaman tambah barang. Salah satu contoh perancangan antarmuka akan ditampilkan pada Gambar 6 di bawah ini.



Gambar 6. Perancangan Antarmuka Halaman Toko

5.5. Perancangan Komponen

Perancangan komponen adalah bagian perancangan algoritma yang akan diimplementasikan pada kode program sistem Primkopal. Terdapat beberapa algoritma dibawah, diantaranya yaitu algoritma fungsi tambah simpanan yang menjadi contoh di bawah ini.

Tabel 2. Algoritma Tambah Simpanan

No	Algoritma
1	Start
2	function Simpan()
3	function handleChange()
4	function handlesubmit()
5	if(!authenticated)
6	redirect to halaman masuk
	else
7	return
8	username =
9	handleChange(username)
	id = handleChange(nrp)
10	jumlah_simpanan =
	handleChange(jumlah_simpanan)

```

11   terbilang =
12     handleChange(terbilang)
13   bukti_transfer =
14     handleChange(buktitransfer)
15   handleSubmit(data)
16     alert success
17     set form empty
18   end if
19 end

```

6. IMPLEMENTASI SISTEM

2.1. Implementasi Basis Data

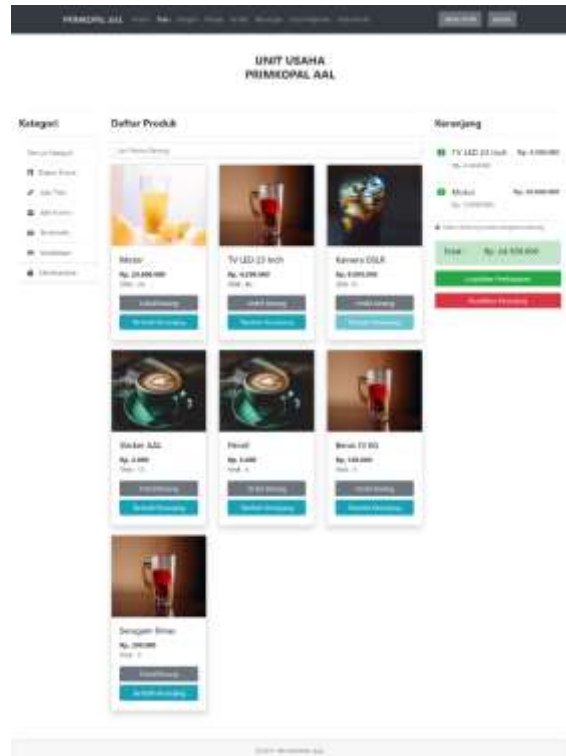
Gambar 7 merupakan hasil dari implementasi basis data sistem Primkopal yang merepresentasikan kebutuhan penyimpanan data yang telah didapatkan berdasarkan kebutuhan fungsional yang ada pada bagian analisis kebutuhan. Implementasi basis data dibuat berdasarkan pada perancangan basis data yang telah dilakukan pada bagian perancangan sistem.



Gambar 7. Implementasi Basis Data

2.2. Implementasi Antarmuka

Gambar 8 merupakan salah satu contoh implementasi antarmuka yang telah dibuat sebelumnya. Contoh pada Gambar 8 adalah antarmuka halaman toko dimana menggambarkan pilihan kategori, barang, dan keranjang serta tombol-tombol yang ada di dalamnya.



Gambar 8. Implementasi Antarmuka Toko

7. PENGUJIAN

Pengujian yang dilakukan pada Sistem Informasi Koperasi Penjualan dan Simpan Pinjam Primkopal AAL Surabaya meliputi pengujian unit, pengujian integrasi, dan pengujian validasi sebagai metode pengujian untuk kebutuhan fungsional dan *compatibility testing* sebagai metode pengujian untuk kebutuhan non fungsional. Hasil dari pengujian Sistem Informasi Koperasi Penjualan dan Simpan Pinjam Primkopal AAL Surabaya dapat dilihat pada Tabel 3 dibawah ini.

Tabel 3. Hasil Pengujian Sistem

No	Jenis Pengujian	Test Case	Hasil Pengujian
1	Pengujian Unit	7 test case	100% Valid
2	Pengujian Integrasi	2 test case	100% Valid
3	Pengujian Validasi	62 test case	100% Valid
4	Compatibility Testing	4 web browser	Dapat dijalankan di semua browser uji

Pada hasil *compatibility testing* yang telah dilakukan pada Sistem Informasi Koperasi Penjualan dan Simpan Pinjam di Primkopal AAL Surabaya yang dibangun menggunakan bahasa pemrograman Javascript dapat berjalan di seluruh *browser* yang diujikan. Namun, masih terdapat kesalahan atau kerusakan pada tampilan

yang diuji pada *browser* Firefox sehingga masih perlu adanya perbaikan dari sisi tampilan sistem.

8. KESIMPULAN DAN SARAN

8.1. Kesimpulan

Terdapat beberapa kesimpulan yang didapatkan dari setiap tahapan penelitian yang telah dilakukan dan dijelaskan sebagai berikut:

1. Analisis kebutuhan sistem yang dilakukan menghasilkan 5 aktor yang akan berperan dalam sistem, lalu terdapat 46 kebutuhan fungsional dan 1 kebutuhan non fungsional dari sistem.
2. Proses perancangan sistem dibedakan menjadi empat bagian, yaitu perancangan arsitektur, perancangan basis data, perancangan komponen, dan perancangan antarmuka.
3. Hasil perancangan tersebut kemudian diimplementasikan pada Sistem Informasi Koperasi Penjualan dan Simpan Pinjam di Primkopal AAL Surabaya dengan menggunakan bahasa pemrograman Javascript yang dibantu oleh teknologi pengolahan *package* yaitu Node Js menjadi sebuah sistem berbasis *website*. Untuk pembuatan antarmuka sistem, penelitian ini menggunakan HTML, CSS, dan Javascript yang dibantu oleh *framework* React Js. Lalu pada bagian backend sistem, dibuat dengan menggunakan bahasa pemrograman Javascript yang dibantu oleh *framework* Express Js. Sedangkan, untuk *database* sistem menggunakan teknologi MySQL dengan bantuan PHPMyAdmin.
4. Pada pengujian telah diselesaikan dengan menerapkan 4 macam pengujian yang terdiri dari pengujian unit, pengujian integrasi, dan pengujian validasi untuk kebutuhan fungsional dan *compatibility testing* untuk kebutuhan non fungsional. Pengujian unit telah dilakukan menggunakan 3 fungsi sebagai sampel dari sistem yang menghasilkan bahwa ketiga fungsi tersebut adalah *valid*. Selanjutnya, pada pengujian integrasi juga telah dilakukan pada fitur pembelian barang yang melibatkan fungsi tambahPengambilan dan hapusKeranjang dan juga menghasilkan hasil *valid*. Selanjutnya untuk pengujian validasi telah dilakukan dengan menggunakan 62 kasus uji yang didapatkan dari 46 kebutuhan fungsional dan telah dihasilkan bahwa 62 kasus tersebut adalah 100% "Sesuai". Pada

compatibility testing menghasilkan bahwa sistem dapat berjalan di seluruh *browser* yang menjadi kasus uji. Namun, masih terdapat kerusakan pada tampilan pada *browser* Firefox.

8.2. Saran

Terdapat beberapa saran yang diberikan sebagai bahan pertimbangan untuk penelitian selanjutnya terhadap penelitian Sistem Informasi Koperasi Penjualan dan Simpan Pinjam Primkopal AAL Surabaya yang disebutkan sebagai berikut:

1. Penambahan fitur yang dapat secara otomatis mengirimkan dan menerima data penjualan, pinjaman, kredit, dan pembayarannya berdasarkan tanggal kepada pihak yang membayarkan gaji karyawan.
2. Penambahan fitur pengambilan dana dari simpanan beserta proses autentikasinya.
3. Penambahan beberapa fungsi untuk terhubung dengan layanan jasa yang ada pada instansi.
4. Perbaikan tampilan melalui CSS sebagaimana yang telah dilihat dari pengujian *compatibility* masih terdapat kerusakan pada tampilan yang diujikan melalui *browser* Firefox.

9. DAFTAR PUSTAKA

- Aleryani, A. Y. 2016. Comparative Study between Data Flow Diagram and Use Case Diagram. *International Journal of Scientific and Research Publications*, 6(3), 124-127.
- Anik Sri Wahyuningsih, J. S. 2019. PENERAPAN MODEL WATERFALL PADA PENGEMBANGAN SISTEM INFORMASI KOPERASI SIMPAN PINJAM CIPTA USAHA MANDIRI BERBASIS WEB. *Jurnal Informatika SIMANTIK*, 4(2), 62.
- Bangare, S. L., Gupta, S., Dalal, M., & Inamdar, A. 2016. Using Node.js to Build High Speed and Scalable Backend Database Server. *International Journal of Research in Advent Technology*, 1(1), 61-64.
- Eniyati, S. 2006. Pengembangan Berorientasi Objek Metode Fusion. *Jurnal Teknologi Informasi DINAMIK*, 11(2), 104-110.
- Firdaus, A., Widodo, S., Sutrisman, A., Nasution, S. G., & Mardiana, R. 2019.

RANCANG BANGUN SISTEM INFORMASI PERPUSTAKAAN MENGGUNAKAN WEB SERVICE PADA JURUSAN TEKNIK KOMPUTER POLSRI. *Jurnal Informanika*, 5(2), 81-87.

Hermawan, R., & Hidayat, A. 2016. Sistem Informasi Penjadwalan Kegiatan Belajar Mengajar Berbasis Web (Studi Kasus: Yayasan Ganesha Operation Semarang). *Jurnal Sains dan Manajemen*, 4(1).

Indra Griha Tofik Isa, G. P. 2017. PERANCANGAN APLIKASI KOPERASI SIMPAN PINJAM BERBASIS WEB (STUDI KASUS KOPERASI MITRA SETIA). *Jurnal Ilmiah Ilmu Ekonomi*, 5(10), 139-141.

Isnardi. (2016). Monitoring Bus Trans Padang Berbasis Web. *Jurnal J-Click*, 3(2), 32-37.

Mustaqbal, M. S., Firdaus, R. F., & Rahmadi, H. 2015. PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS (STUDI KASUS : APLIKASI PREDIKSI KELULUSAN SNMPTN). *Jurnal Ilmiah Teknologi Informasi Terapan*, 1(3).

Puspitasari, D. 2015. RANCANG BANGUN SISTEM INFORMASI KOPERASI SIMPAN PINJAM KARYAWAN BERBASIS WEB. *Jurnal Pilar Nusa Mandiri*, 11(2), 186-187.

Sukamto, R., & Shalahuddin, M. 2013. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek* (Black-Box Testing ed.). Bandung: Informatika.