

PERANCANGAN PERANGKAT LUNAK KOPERASI SIMPAN PINJAM BISMILLAH MASDAR SALIHA DESA PUGU BERBASIS JAVA STANDARD EDITION DAN MYSQL DENGAN ARSITEKTUR MVC (MODEL VIEW CONTROLLER)

***Software Design Of Bismillah Masdar Saliha Savings And Loan Cooperative
Software Pugu Village Based On Java And Mysql With Mvc Architecture
(Model View Controller)***

¹ Mhd Theo Ari Bangsa*, ² Albet Triadi

^{1,2}Sistem Informasi, Sains dan Teknologi, Universitas Islam Negeri Sultan Thaha Saifuddin Jambi
Jl. Arif Rahman Hakim No.111, Simpang IV Sipin, Kec. Telanaipura, Kota Jambi, Indonesia

*e-mail: amhd.theo@gmail.com

Abstrak

Saat ini banyak sekali metode pengembangan perangkat lunak dan bahasa pemrograman yang ada, salah satunya merupakan perancangan perangkat lunak berorientasi objek (OOP) dengan bahasa pemrograman JAVA tipe pemrograman ini adalah berbasis object. Dalam hal ini berbagai plugin visual editor, membuat perancangan antarmuka menjadi lebih mudah. Namun, pada umumnya, hasil perancangannya masih terasa sulit untuk dipahami atau dipelihara, untuk dilacak kesalahannya, dan seringkali tidak dapat digunakan ulang (unreusable). Untuk itu pada penelitian ini digunakan konsep MVC (Model-View-Controller) dalam merancang perangkat lunak Koperasi Simpan Pinjam Bismillah Masdar Saliha Desa Pugu dengan bahasa pemrograman JAVA dan MySQL sebagai database server. Konsep utama MVC adalah membagi kerja sistem antara Model, View, dan Controller. Dapat disimpulkan dengan adanya perangkat lunak ini akan membantu pengurus koperasi dalam mengolah data dan juga dengan mengimplementasikan arsitektur MVC menjadikan kode program lebih rapih mudah dikembangkan.

Kata kunci: Model-view-controller, koperasi, java, mysql

Abstract

Currently, there are many software development methods and programming languages that exist, one of which is object-oriented software design (OOP) with the JAVA programming language. This type of programming is object-based. Here, various visual editor plugins, make interface design easier. However, in general, the design results are still difficult to understand or maintain, to trace errors, and often cannot be reused (unreusable). For this reason, this study used the MVC (Model-View-Controller) concept in designing the Bismillah Masdar Saliha Savings and Loan Cooperative software in Pugu Village with the JAVA programming language and MySQL as the database server. The main concept of MVC is to divide the work of the system between the Model, View, and Controller. It can be concluded that the existence of this software will help cooperative management in processing data and also by implementing the MVC architecture, making the program code neater and easier to develop.

Keywords:Model-view-controller, cooperative, java, mysql.

1 Pendahuluan

Saat ini banyak sekali metode pengembangan perangkat lunak yang ada, begitu juga dengan jenis-jenis bahasa pemrograman, yaitu diantaranya bahasa pemrograman yang menggunakan pendekatan prosedural dan yang menggunakan pendekatan berorientasi objek dengan mengimplementasikan arsitektur MVC (model-view-controller).

Bahasa pemrograman yang menggunakan pendekatan prosedural atau konvensional, masih mencapur kode program antara logika proses dan antarmuka. Antarmuka dalam pemrograman prosedural hanya dapat digunakan dalam satu proses logika, hal ini akan mengurangi modularitas aplikasi, membuat pemilihan aplikasi lebih sulit, dan juga membuat antarmuka sulit untuk dimodifikasi ketika digunakan untuk aplikasi lain.

Permasalahan tersebut telah menimbulkan gagasan untuk memisahkan logika aplikasi dengan antarmuka, sehingga aplikasi yang dibangun dengan mudah dapat diganti antarmukanya setiap saat. Pada tahun 1970-an penemu smalltalk, Trygve Reenskaug, mendefinisikan sebuah arsitektur untuk menyelesaikan masalah tersebut, yang disebut arsitektur MVC (Model-View-Controller) yang memisahkan logika bisnis dan presentasi (tampilan) aplikasi.

MVC (Model-View-Controller) adalah arsitektur aplikasi yang memisahkan kode-kode aplikasi dalam tiga lapisan, Model, View dan Control. MVC termasuk dalam arsitektural design pattern yang menghendaki organisasi kode yang terstruktur dan tidak bercampur aduk. Inti dari arsitektur MVC adalah pemisahan tampilan pengguna dari kendali asupan (control) pengguna dan model informasi yang mendasarinya [1].

Dalam penelitian ini Pola MVC (Model-View-Controller) akan diterapkan dalam merancang perangkat lunak Koperasi simpan pinjam untuk mempermudah dalam pengelolaan dan pemilihan aplikasi, karena adanya pemisahan antara model, view, dan controller.

2 Tinjauan Literatur

Setiap perangkat lunak memiliki pola arsitektur pengembangan. Pola arsitektur diklasifikasikan ke 4 kategori, yaitu mud to structure, sistem terdistribusi, sistem interaktif, dan adaptable system [2]. Sistem interaktif fokus pada interaksi pengguna, *Interface* pengguna merupakan yang paling sering berubah dan beradaptasi, sehingga diperlukan arsitektur sistem yang menjaga core sistem tetap independent terhadap *user interface* [2]. *Model-view-controller* (MVC) atau pola desain arsitektur yang baik untuk perangkat lunak interaktif, ide utamanya yaitu memisahkan user interface dan model data [3]. Arsitektur MVC pernah digunakan pada penelitian sebelumnya untuk membuat sistem kuis online secara adaptive [4], juga pada sistem Business Intelligent [2], dimana sistem yang dibangun memiliki tingkat modularitas dan reuseabilitas yang sangat tinggi.

3 Metode Penelitian (or Research Method)

3.1 Analisa Sistem

Analisa sistem dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam bagian-bagian komponennya, dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

Analisa masalah dapat dilakukan agar penemuan masalah yang ada dapat diketahui penyebabnya, sehingga nantinya dari analisa masalah tersebut didapat suatu bentuk pemecahan masalah. Dari hasil analisa, penulis merumuskan masalah-masalah atau kendala-kendala dalam kegiatan simpan pinjam di Koperasi Bismillah Masdar Saliha. Berdasarkan rumusan masalah yang ada, penulis menemukan solusi agar dibuat suatu aplikasi Koperasi Simpan Pinjam dengan pola MVC (Model-View-Controller) agar dapat membantu koperasi tersebut untuk meningkatkan kegiatan simpan pinjam di koperasi tersebut juga dengan adanya pola MVC(Model-View-Controller) dapat memudahkan aplikasi untuk dipelihara dan kode program menjadi lebih rapi, sehingga masalah-masalah yang ada dapat diatasi dengan baik.

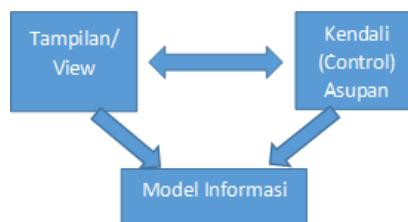
Sistem yang berjalan pada Koperasi Simpan Pinjam Bismillah Masdar Saliha saat ini untuk proses transaksi simpan pinjam dilakukan secara manual, hal tersebut dapat dilihat dalam pengelolaan informasi yang masih tersusun dalam suatu arsip ataupun data yang tersimpan pada media komputer.

3.1 Konsep MVC (Model-View-Controller)

Pendekatan MVC merupakan cara yang efektif untuk mendukung presentasi multipel dari data. User dapat berinteraksi dengan setiap presentasi dengan menggunakan gaya yang sesuai dengannya. Data yang akan ditampilkan dienkapsulasi pada objek model. Setiap objek model dapat memiliki sejumlah objek view yang terpisah yang berhubungan dengannya, dimana setiap view merupakan representasi display yang berbeda dari model [5].



Gambar 1 Sistem yang sedang berjalan



Gambar 2 Model MVC

3.2 Perancangan Menggunakan UML (Unified Modeling Language)

Perancangan aplikasi Koperasi Simpan Pinjam dengan arsitektur MVC ini dirancang menggunakan alat bantu berupa UML (Unified Modeling Language) agar mempermudah memindahkan konsep sistem yang dirancang ke dalam bentuk program, dimana perancangan programnya digambarkan dalam bentuk diagram-diagram berikut:

1. Use Case Diagram

Use Case Diagram menggambarkan bagaimana proses-proses yang dilakukan oleh aktor terhadap sebuah sistem. Ada 4 aktor yang terlibat dalam aplikasi Koperasi Simpan Pinjam ini adalah sekretaris, bendahara, anggota, dan calon anggota.

a. Definisi Aktor

Tabel 1 Definisi Aktor

No	Aktor	Deskripsi
1.	Sekretaris	Orang yang mengurus calon anggota yang ingin mendaftar menjadi anggota dan anggota yang ingin mengajukan peminjaman.
2.	Bendahara	Orang yang melayani transaksi simpanan anggota dan transaksi pinjaman.
3.	Anggota	Orang yang telah terdaftar menjadi anggota dan mempunyai simpanan anggota di Koperasi Simpan Pinjam Bismillah Masdar Saliha Desa Pugu.

4. Calon Anggota	Orang yang akan mendaftar sebagai anggota Koperasi Simpan Pinjam Bismillah Masdar Saliha Desa Pugu.
------------------	---

b. Definisi Use Case

Tabel 2 Definisi Use Case

No	Use Case	Aktor	Deskripsi
1.	Login	Sekretaris Bendahara Anggota	<i>Use case</i> ini memungkinkan seorang sekretaris, bendahara, dan anggota untuk masuk kedalam sistem dengan menggunakan <i>username</i> dan <i>password</i> .
2.	Manajemen Anggota	Sekretaris	<i>Use case</i> ini menggambarkan kegiatan manajemen data anggota.
3.	Manajemen Pengurus	Sekretaris	<i>Use case</i> ini menggambarkan kegiatan manajemen data pengurus.
4.	Manajemen Pengajuan Pinjaman	Sekretaris	<i>Use case</i> ini menggambarkan kegiatan manajemen data pengajuan pinjaman
5.	Ganti Password	Sekretaris Bendahara Anggota	<i>Use case</i> ini memungkinkan seorang sekretaris, bendahara, dan anggota untuk mengganti <i>password</i> sesuai keinginan.
6.	Manajemen Biaya	Bendahara	<i>Use case</i> ini menggambarkan kegiatan Manajemen Biaya
7.	Manajemen Simpanan	Bendahara	<i>Use case</i> ini menggambarkan kegiatan manajemen simpanan
8.	Manajemen Pinjaman	Bendahara	<i>Use case</i> ini menggambarkan kegiatan manajemen pinjaman.
9.	Cetak Buku Simpanan	Bendahara	<i>Use case</i> ini menggambarkan kegiatan bendahara mencetak buku simpanan anggota.
10.	Cetak Buku Pinjaman	Bendahara	<i>Use case</i> ini menggambarkan kegiatan bendahara mencetak buku pinjaman anggota.
11.	Lihat saldo simpanan	Anggota	<i>Use case</i> ini menggambarkan kegiatan anggota melihat saldo simpanan anggota.
12.	Lihat Pinjaman	Anggota	<i>Use case</i> ini menggambarkan kegiatan anggota melihat informasi pinjaman.

2. Class Diagram

Class diagram menggambarkan bagaimana struktur dari perancangan sistem. Semua proses yang dilakukan oleh aktor terhadap aplikasi akan didefinisikan dengan menggunakan class diagram. Class diagram menunjukkan bentuk visualisasi dalam pembuatan sistem. Masing-masing class memiliki attribute dan metoda / fungsi sesuai dengan proses yang terjadi.

a. Model Class Diagram

Tabel 1 Keterangan Class Diagram

No	Aktor	Deskripsi
1.	SekretarisView	Merupakan kelas yang menangani tampilan Sekretaris.

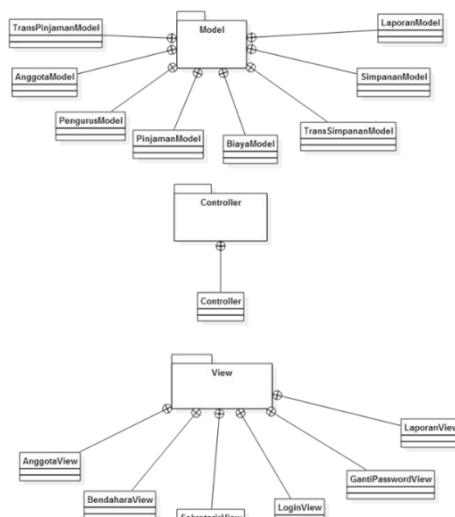
2. AnggotaView	Merupakan kelas yang menangani tampilan Anggota.
3. BendaharaView	Merupakan kelas yang menangani Bendahara.
4. LoginView	Merupakan kelas yang menangani tampilan Login
5. GantiPasswordView	Merupakan kelas yang menangani tampilan ganti password.
6. FormLaporanView	Merupakan kelas yang menangani tampilan laporan keuangan.
7. Controller	Merupakan kelas yang menangani logika bisnis aplikasi.
8. AnggotaModel	Merupakan kelas yang merepresentasikan <i>object</i> anggota
9. BiayaModel	Merupakan kelas yang merepresentasikan <i>object</i> biaya.
10. LaporanModel	Merupakan kelas yang merepresentasikan <i>Object</i> laporan keuangan.
11. PengurusModel	Merupakan kelas yang merepresentasikan <i>Object</i> pengurus.
12. PinjamanModel	Merupakan kelas yang merepresentasikan <i>Object</i> pinjaman anggota
13. SimpananModel	Merupakan kelas yang merepresentasikan <i>Object</i> simpanan anggota.
14. TransSimpananModel	Merupakan kelas yang merepresentasikan <i>Object</i> transaksi simpanan.
15. TransPinjamanModel	Merupakan kelas yang merepresentasikan <i>Object</i> transaksiPinjaman.
16. DaoInterface	Merupakan kelas <i>interface</i> yang berisi metode operasi yang menangani perintah query ke database.
17. DaoImplement	Merupakan kelas yang mewarisi (<i>implements</i>) kelas DaoInterface yang menenkapsulasi operasi query pada database tanpa perlu mengekspos detail database.
18. Koneksi	Merupakan kelas utilitas untuk koneksi ke basis data dan melakukan query

3. Package Diagram

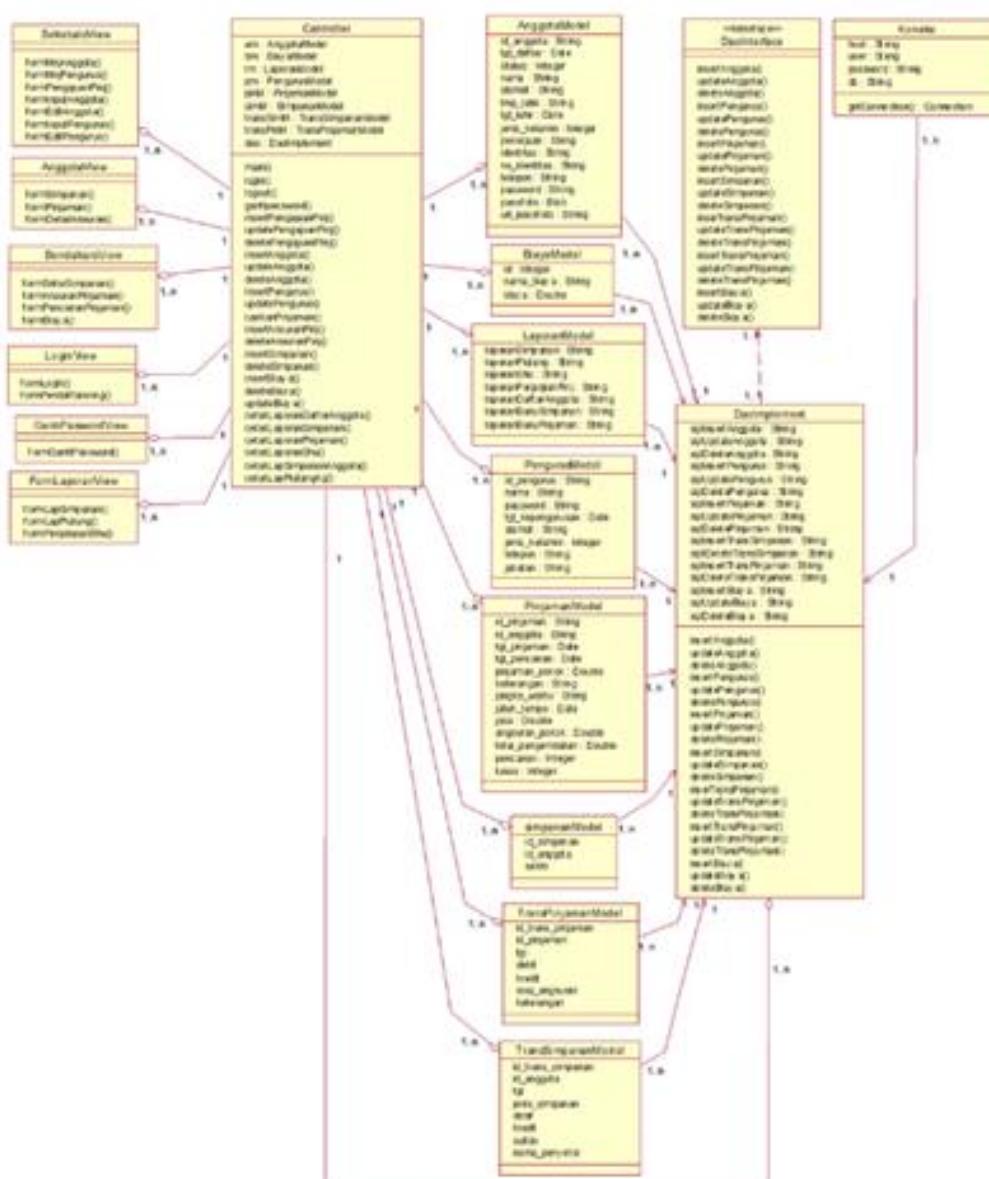
Package Diagram digunakan untuk mengelompokkan elemen-elemen yang saling terkait dalam diagram UML. Terdapat tiga package utama dari beberapa lainnya yang akan dibuat dalam perangkat lunak KSP Bismillah Masdar Saliha yaitu package model, view, dan controller.



Gambar 2 Use Case Diagram



Gambar 2 Package Diagram



Gambar 3 Class Diagram

4 Hasil dan Pembahasan

4.1 Database

Database pada perangkat lunak Koperasi Simpan Pinjam ini menggunakan DBMS MySQL dan akan di letakkan pada PC server yang dapat diakses oleh PC client melalui jaringan local dengan settingan IP address server 192.168.0.1 dan untuk PC client 192.168.0.2. Berikut tabel-tabel yang terdapat didalam database tersebut:

1. Tabel anggota

Tabel anggota merupakan tempat penyimpanan informasi mengenai data-data anggota koperasi untuk dapat login kedalam sistem yang dibuat.

Field	Type	Collation	Attributes	Null	Default	Extra	Action
id_anggota	varchar(6)	latin1_swedish_ci		No	None		
tgl_daftar	date			No	0000-00-00		
status	int(1)			No	None		
nama	varchar(50)	latin1_swedish_ci		No	None		
alamat	text	latin1_swedish_ci		No	None		
tmp_lahir	varchar(30)	latin1_swedish_ci		No	None		
tgl_lahir	date			No	0000-00-00		
jenis_kelamin	int(1)			No	None		
pekerjaan	varchar(30)	latin1_swedish_ci		No	None		
identitas	varchar(3)	latin1_swedish_ci		No	None		
no_identitas	varchar(30)	latin1_swedish_ci		No	None		
telepon	varchar(16)	latin1_swedish_ci		No	None		
password	varchar(15)	latin1_swedish_ci		No	None		
pasfoto	blob		BINARY	No	None		

Gambar 4.1 Tabel Anggota

2. Tabel biaya

Tabel biaya merupakan tempat penyimpanan informasi mengenai data-data biaya pengeluaran koperasi.

Field	Type	Collation	Attributes	Null	Default	Extra	Action
id	int(5)			No	None	AUTO_INCREMENT	
nama	varchar(50)	latin1_swedish_ci		No	None		
jumlah	double			No	None		
tgl	date			No	0000-00-00		

Gambar 4.2 Tabel Biaya

3. Tabel pengurus

Tabel pengurus merupakan tempat penyimpanan informasi mengenai data-data pengurus koperasi untuk dapat login kedalam sistem yang dibuat.

Field	Type	Collation	Attributes	Null	Default	Extra	Action
id_pengurus	varchar(6)	latin1_swedish_ci		No	None		
nama	varchar(50)	latin1_swedish_ci		No	None		
password	varchar(30)	latin1_swedish_ci		No	None		
tgl_kepengurusan	date			No	0000-00-00		
alamat	text	latin1_swedish_ci		No	None		
jenis_kelamin	int(1)			No	None		
telepon	varchar(16)	latin1_swedish_ci		No	None		
jabatan	varchar(10)	latin1_swedish_ci		No	None		

Gambar 4.3 Tabel Pengurus

4. Tabel pinjaman

Tabel pinjaman merupakan tempat penyimpanan informasi mengenai data-data pinjaman koperasi

Field	Type	Collation	Attributes	Null	Default	Extra	Action
id_pinjaman	varchar(8)	latin1_swedish_ci		No	None		
id_anggota	varchar(6)	latin1_swedish_ci		No	None		
tgl_pinjaman	date			No	0000-00-00		
pinjaman_pokok	double			No	None		
keterangan	text	latin1_swedish_ci		No	None		
jangka_waktu	int(3)			No	None		
jatuh_tempo	date			No	0000-00-00		
jasa	double			No	None		
angsuran_pokok	double			No	None		
total_angsuran	double			No	None		
total_pengembalian	double			No	None		
pencairan	int(1)			No	None		
tgl_pencairan	date			Yes	2000-02-02		
lunas	int(1)			No	0		

Gambar 4.4 Tabel Pinjaman

1. Tabel simpanan

Tabel simpanan merupakan tempat penyimpanan informasi mengenai data-saldo simpanan anggota koperasi.

Field	Type	Collation	Attributes	Null	Default	Extra	Action
id_simpanan	varchar(10)	latin1_swedish_ci		No	None		
id_anggota	varchar(10)	latin1_swedish_ci		No	None		
saldo	double			No	None		

Gambar 4.5 Tabel Simpanan

2. Tabel trans_pinjaman

Tabel trans_pinjaman merupakan tempat penyimpanan informasi mengenai data-data transaksi pinjaman koperasi.

Field	Type	Collation	Attributes	Null	Default	Extra	Action
id_trans_pinjaman	varchar(15)	latin1_swedish_ci		No	None		
id_pinjaman	varchar(8)	latin1_swedish_ci		No	None		
tgl	date			No	0000-00-00		
debit	double			No	None		
kredit	double			No	None		
sisa_angsuran	double			No	None		

Gambar 4.6 Tabel Trans_pinjaman

3. Tabel trans_simpanan

Tabel trans_simpanan merupakan tempat penyimpanan informasi mengenai data-data transaksi simpanan anggota koperasi.

Gambar 4.7 Tabel Trans_simpanan

4.2 User Interface

1. Form Login

Form ini adalah form *User Interface* pertama yang terdapat fasilitas login, melihat laporan keuangan dan pendaftaran calon anggota koperasi.



Gambar 4.8 Form Login

2. Form Laporan Keuangan

Pada form Laporan Keuangan user dapat melihat dan mencetak laporan daftar simpanan anggota, daftar piutang anggota, dan penjelasan SHU.

Gambar 4.9 Form Laporan Keuangan

3. Form pendaftaran anggota bagian calon anggota

Pada form ini calon anggota dapat mengentrikan data pendaftaran calon anggota, setelahnya menuju sekretaris untuk melengkapi syarat dan validasi anggota.

Gambar 4.10 Input Data Anggota

5.6 Laporan

1. Laporan data simpanan anggota

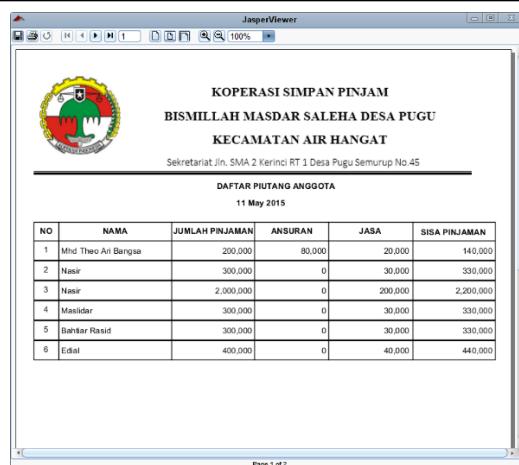
Gambar 4.31 merupakan implementasi laporan data simpanan anggota koperasi yang ada pada aplikasi koperasi simpan pinjam tersebut:

NO	NAMA	SIMPANAN			JUMLAH SIMPANAN
		POKOK	WAJIB	SUKARELA	
1	Mhd Theo Ari Bangsa	100.000	5.000	700.000	805.000
2	Nasir	100.000	5.000	50.000	155.000
3	Nurliama	100.000	5.000	45.000	150.000
4	Desi Nurhayatilis	100.000	5.000	35.000	140.000
5	Edial	100.000	5.000	30.000	135.000
6	Bantiar Rasd	100.000	5.000	25.000	130.000
7	Maslidaer	200.000	5.000	15.000	220.000
8	theo	100.000	0	0	100.000
9	Razor	100.000	0	0	100.000
10	Hanya	100.000	0	0	100.000
11	coba2	100.000	0	0	100.000
		JUMLAH	1.200.000	35.000	900.000
Pugu, 11 May 2015					
PENGURUS KOPERASI BISMILLAH MASDAR SALEHA		SEKRETARIS,		BENDAHARA,	
KETUA,					
Zulyadi S.Pd		Eni Afrida, S.Pd		Aida Sasmardi, S.Pd	

Gambar 4.31 Laporan Data Simpanan Anggota

2. Laporan Piutang Anggota

Gambar 4.31 merupakan implementasi laporan piutang anggota koperasi yang ada pada aplikasi koperasi simpan pinjam tersebut:



NO	NAMA	JUMLAH PINJAMAN	ANSURAN	JASA	SISA PINJAMAN
1	Mhd Theo Ari Bangsa	200.000	80.000	20.000	140.000
2	Nasir	300.000	0	30.000	330.000
3	Nasir	2.000.000	0	200.000	2.200.000
4	Masildar	300.000	0	30.000	330.000
5	Bahar Rasid	300.000	0	30.000	330.000
6	Edal	400.000	0	40.000	440.000

Gambar 4.32 Laporan Piutang Anggota

5 Kesimpulan

Setelah melakukan perancangan sistem dan penerapan aplikasi ini dapat diambil beberapa kesimpulan, diantaranya:

- 1) Dengan menggunakan arsitektur MVC (Model-View-Controller) kode program pada aplikasi tersebut menjadi lebih rapih, karena ada pembagian yang jelas antara masing-masing layer/lapisan, akan tetapi waktu yang dibutuhkan dalam pembuatan aplikasi tersebut jauh lebih lama, karena pembuatan aplikasi tersebut dilakukan secara bertahap, mulai dari lapisan Model, View, Controller.
- 2) Atas konsep arsitektur MVC ini, pemrogram akan dihadapkan dengan kenyataan bahwa pemeliharaan perangkat lunak ini memudahkan programer bila membutuhkan pengembangan dan penambahan lainnya. Ketika fungsi lain dalam Model dikembangkan, maka nantinya kedua komponen lainnya dapat menyesuaikan Model, begitu juga dengan View yang biasanya mengalami perubahan untuk interaksi kedua pengguna. Sedangkan Controller dalam hal ini hanya penghubung kedua komponen lainnya.
- 3) Komponen-komponen yang terdapat pada aplikasi tersebut dapat dipergunakan oleh aplikasi lain yang membutuhkan komponen yang sama.
- 4) Perangkat lunak Koperasi Simpan Pinjam ini dapat memudahkan pengurus dalam mengolah data transaksi simpan pinjam sehingga pembuatan laporan-laporan lebih efesien.

Referensi (Reference)

- [1] A. Nugroho, *Rekayasa perangkat lunak berorientasi objek dengan metode USDP*. Penerbit Andi, 2010.
- [2] M. Kalelkar, P. Churi, and D. Kalelkar, “Implementation of model-view-controller architecture pattern for business intelligence architecture,” *International Journal of Computer* Citeseer, 2014.
- [3] A. Leff and J. T. Rayfield, “Web-application development using the Model/View/Controller design pattern,” *Proc. - 5th IEEE Int. Enterp. Distrib. Object Comput. Conf.*, vol. 2001-January, no. January, pp. 118–127, 2001.
- [4] A. Hidayat and B. Surarso, “Penerapan Arsitektur Model View Controller (MVC) Dalam Rancang Bangun Sistem Kuis Online Adaptif,” *Seminar Nasional Teknologi Informasi dan* academia.edu, 2012.
- [5] I. Sommerville, *Engineering software products*. Pearson London, 2020.