

**PEMBANGUNAN SISTEM INFORMASI PERPUSTAKAAN
BERBASIS WEB MENGGUNAKAN METODE *RAPID
APPLICATION DEVELOPMENT (RAD)*
(STUDI PADA : SMK NEGERI 11 MALANG)**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Pendidikan

Disusun oleh:

Nur Aini

NIM: 156150600111004



PROGRAM STUDI PENDIDIKAN TEKNOLOGI INFORMASI
JURUSAN SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2019

PENGESAHAN

PEMBANGUNAN SISTEM INFORMASI PERPUSTAKAAN BERBASIS WEB
MENGGUNAKAN METODE RAPID APPLICATION DEVELOPMENT (RAD)
(STUDI PADA : SMK NEGERI 11 MALANG)

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Pendidikan

Disusun Oleh :

Nur Aini

NIM : 156150600111004

Skripsi ini telah diuji dan dinyatakan lulus pada

26 Juli 2019

Telah diperiksa dan disetujui oleh:

Pembimbing I

Satrio Agung Wicaksono, S.Kom., M.Kom.

NIP : 19860521 201212 1 001

Pembimbing II

Issa Arwani, S.Kom., M.Sc.

NIP : 19830922 201212 1 003

Mengetahui



Dr. Herman Tolle, S.T., M.T.
NIP. 19740823 200012 1 001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar referensi.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 19 Maret 2019



PRAKATA

Segala puji bagi Tuhan Yang Maha Esa, atas rahmat dan hidayah yang diberikan sehingga penulis bisa menyelesaikan skripsi yang berjudul “PEMBANGUNAN SISTEM INFORMASI PERPUSTAKAAN BERBASIS WEB MENGGUNAKAN METODE RAPID APPLICATION DEVELOPMENT (RAD) (STUDI PADA : SMK NEGERI 11 MALANG)” dengan tepat waktu.

Penelitian ini mengajarkan banyak hal baru kepada penulis tentang bagaimana membangun sistem informasi dengan baik. Selain itu, penelitian ini juga memberikan pengalaman baru kepada penulis yang sangat berguna untuk menjadi pribadi yang lebih baik lagi ke depannya untuk senantiasa terus belajar. Penelitian ini tidak dapat diselesaikan tanpa dukungan dari berbagai pihak yang sangat membantu penulis selama proses penelitian, untuk itu penulis ingin menyampaikan ucapan terima kasih kepada :

1. Bapak Satrio Agung Wicaksono, S.Kom., M.Kom., selaku pembimbing satu yang telah memberikan pengarahan dan bimbingan kepada penulis, serta mengajarkan ketegasan dan kedisiplinan kepada penulis dalam menyelesaikan skripsi ini,
2. Bapak Issa Arwani, S.Kom., M.Sc., selaku pembimbing dua yang telah dengan sabar dalam membimbing, memberikan pengarahan, serta mengajarkan kedisiplinan kepada penulis,
3. Bapak Satrio Agung Wicaksono, S.Kom., M.Kom., selaku Ketua Program Studi Pendidikan Teknologi Informasi Fakultas Ilmu Komputer Universitas Brawijaya,
4. Bapak Dr. Eng. Herman Tolle, S.T., M.T., selaku ketua Jurusan Sistem Informasi Fakultas Ilmu Komputer Universitas Brawijaya,
5. Ibu Rahmah Widarti., S.Pd., selaku ketua perpustakaan SMK Negeri 11 Malang yang telah memberikan ijin penelitian kepada penulis serta telah membantu penulis dalam mendapatkan data-data wawancara penelitian,
6. Ibu Titik Ambarsari selaku petugas perpustakaan SMK Negeri 11 Malang yang telah membantu penulis dalam mendapatkan data-data wawancara penelitian mengenai *review* aplikasi yang terdapat di perpustakaan SMK Negeri 11 Malang,
7. Ibu Nanik Nurhayati, S.Kep., selaku guru bagian penerimaan mahasiswa penelitian di SMK Negeri 11 Malang yang telah memberikan ijin kepada penulis untuk melakukan penelitian di sekolah tersebut,
8. Lizma Nur Saida selaku teman dekat yang memberikan dukungan dan semangat, serta telah membantu kelancaran diawal proses penelitian di SMK Negeri 11 Malang,
9. Kedua orang tua yang selalu memberikan doa dan dukungan kepada penulis, yang selalu memberikan semangat kepada penulis dalam menyelesaikan skripsi ini,

10. Dan semua pihak terkait yang tidak dapat penulis sebutkan satu persatu yang telah membantu proses kelancaran penelitian ini dengan baik.

Semoga atas segala kebaikan pihak-pihak yang telah membantu dalam proses kelancaran skripsi penulis, diberikan balasan kebaikan yang berlipat oleh Tuhan. Penulis menyadari bahwa penelitian ini masih banyak kekurangan. Untuk itu kritik dan saran yang membangun diharapkan untuk lebih baik lagi kedepannya.

Malang, 19 Maret 2019



Penulis
nuraini119@gmail.com

ABSTRAK

Nur Aini, Pembangunan Sistem Informasi Perpustakaan Berbasis Web Menggunakan Metode *Rapid Application Development* (RAD) (Studi Pada : SMK Negeri 11 Malang)

Pembimbing: Satrio Agung Wicaksono, S.Kom., M.Kom dan Issa Arwani, S.Kom., M.Sc.

Penelitian ini dilakukan saat Praktik Pengalaman Lapangan (PPL) di SMK Negeri 11 Malang. Peneliti melihat bahwa perpustakaan yang terdapat di SMK Negeri 11 Malang masih manual dalam memberikan pelayanan kepada pengunjung perpustakaan yang meminjam buku. Setiap pengunjung yang akan meminjam buku terlebih dahulu dicatat dibuku catatan. Berdasarkan hasil wawancara dengan ketua perpustakaan Ibu Rahmah Widarti pada tanggal 15 Februari 2019, menjelaskan bahwa terdapat aplikasi SLA (*Smart Library Automation*) berbasis desktop dalam membantu petugas mengelola data – data di perpustakaan. Selain itu juga dilakukan wawancara dengan petugas perpustakaan Ibu Titik Widarti selaku admin yang menjalankan aplikasi SLA (*Smart Library Automation*) tersebut. Ibu Titik menjelaskan bahwa aplikasi SLA (*Smart Library Automation*) tidak bisa dibawa kemana-mana karena berbasis dekstop, dan data – data yang diinputkan masih harus satu per satu, sedangkan untuk saat ini peminjaman dilakukan manual dengan menulis dibuku catatan yang telah disediakan petugas. Maka dari itu dilakukan penelitian yang berjudul “Pembangunan Sistem Informasi Perpustakaan Berbasis Web Menggunakan Metode *Rapid Application Development* (RAD) (Studi pada : SMK Negeri 11 Malang)”. Metode *Rapid Application Development* (RAD) merupakan metode yang dilakukan peneliti dalam membangun sistem informasi perpustakaan berbasis web tersebut. *Rapid Application Development* (RAD) menurut (Pressman, 2001) adalah model pengembangan perangkat lunak tambahan yang menekankan siklus pengembangan yang sangat singkat, meskipun proses pembangunan singkat, tidak mengurangi kualitas produk tersebut. Hasil dari pembangunan sistem informasi perpustakaan dapat memudahkan admin dalam pengelolaan data-data perpustakaan setiap waktu dan memudahkan pengguna dalam melakukan peminjaman buku diperpustakaan dengan hasil pengujian *black box* sebanyak 100% valid. Dan hasil dari pengujian *User Acceptance Testing*(UAT) yang dilakukan kepada 5 penguji yang terdiri dari ketua perpustakaan, petugas perpustakaan, 1 guru, 1 staff dan 1 siswa memiliki hasil 84% pengguna setuju bahwa Sistem Informasi Perpustakaan Sekolah (SIPS) Malang telah memenuhi kebutuhan pengguna dalam peminjaman buku di perpustakaan SMKN 11 Malang.

Kata kunci: sistem informasi perpustakaan , *rapid application development* (RAD), *black box testing*, *User Acceptance Testing*(UAT).

ABSTRACT

Nur Aini, Development of Web-Based Library Information System Using Rapid Application Development (RAD) Method (Study in: SMK Negeri 11 Malang)

Supervisor: Satrio Agung Wicaksono, S.Kom., M.Kom and Issa Arwani, S.Kom., M.Sc.

This research was conducted during the Field Experience Practice (PPL) at SMK Negeri 11 Malang. Researchers see that the library at SMK Negeri 11 Malang is still manual in providing services to library visitors who borrow books. Every visitor who will borrow a book first is recorded in a note. Based on the results of interviews with library chair Ibu Rahmah Widarti on 15 February 2019, explaining that there is a desktop-based SLA (Smart Library Automation) application in helping officers manage data in the library. In addition, interviews were also conducted with library staff Ibu Titik Widarti as the admin who runs the SLA (Smart Library Automation) application. Ibu Titik explained that the SLA (Smart Library Automation) application cannot be carried anywhere because it is desktop-based, and the data inputted must still be one at a time, whereas for the time being borrowing is done manually by writing a notebook provided by the officer. Therefore a study entitled "Development of Web-Based Library Information Systems Using the Rapid Application Development (RAD) Method (Study at: SMK Negeri 11 Malang)". The Rapid Application Development (RAD) method is a method used by researchers in building a web-based library information system. Rapid Application Development (RAD) according to (Pressman, 2001) is an additional software development model that emphasizes a very short development cycle, even though the development process is short, does not reduce the quality of the product. The results of the development of library information systems can facilitate the admin in managing library data at any time and make it easier for users to borrow books in the library with the results of black box testing as much as 100% valid. And the results of the User Acceptance Testing (UAT) conducted on 5 examiners consisting of library chair, library officer, 1 teacher, 1 staff and 1 student have the results of 84% of users agree that the Malang School Library Information System (SIPS) has met the needs users in borrowing books in the library of SMK 11 Malang.

Keywords: *library information systems, rapid application development (RAD), black box testing, User Acceptance Testing (UAT).*

DAFTAR ISI	
PENGESAHAN	Error! Bookmark not defined.
PERNYATAAN ORISINALITAS	i
PRAKATA.....	iii
ABSTRAK.....	v
ABSTRACT.....	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR.....	xiii
LAMPIRAN	xvi
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan	3
1.4 Manfaat.....	3
1.5 Batasan Masalah.....	3
1.6 Sistematika Pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Perpustakaan	5
2.1.1 Jenis Perpustakaan	5
2.2 Sistem Informasi	6
2.2.1 Sistem Informasi Perpustakaan.....	6
2.3 Model Pengembangan.....	6
2.3.3 <i>CakePhp</i>	8
2.4 <i>Unified Modeling Language (UML)</i>	9
2.4.1 Use Case Diagram	9
2.4.2 Activity Diagram.....	11
2.4.3 Sequence Diagram	13
2.4.4 Class Diagram	15
2.5 PHP (<i>Hypertext Preprocessor</i>).....	17
2.6 MySQL.....	17
2.7 <i>Entity Relationship Diagram</i>	18

2.8 <i>Data Definition Language (DDL)</i>	18
2.9 PhpMyAdmin	18
2.10 <i>XAMPP</i>	18
2.11 <i>Visual Studio Code</i>	19
2.12 <i>Visual Paradigm</i>	19
2.13 Web.....	19
2.14 Pengujian	19
BAB 3 METODOLOGI	22
3.1 Metode Pengumpulan Data.....	22
3.2 Tahapan Penelitian	22
3.3 Waktu dan Lokasi Penelitian.....	24
3.4 Metode Pengembangan	24
3.4.1 Perencanaan Syarat-Syarat (<i>Requirement Planning</i>). 24	
3.4.2 Workshop Desain RAD	24
3.5 Implementasi	25
BAB 4 HASIL DAN PEMBAHASAN	27
4.1 Perencanaan Syarat - Syarat (<i>Requirement Planning</i>).....	27
4.2 Proses Desain (<i>Workshop Desain RAD</i>)	30
4.2.1 Proses iterasi ke-1.....	30
4.2.1.1. <i>Use Case Diagram</i> Iterasi-1	30
4.2.1.2. <i>Activity Diagram</i> Iterasi-1.....	31
4.2.1.3. <i>Sequence Diagram</i> iterasi-1	42
4.2.1.4. <i>Class Diagram</i> iterasi-1.....	48
4.2.1.5. Perancangan database iterasi-1	50
4.2.1.6. <i>Entity Relationship Diagram</i> iterasi-1	55
4.2.1.7. Rancangan Antarmuka (<i>Interface</i>) iterasi-1	56
4.2.2 Proses iterasi ke-2	60
4.2.2.1. <i>Use Case Diagram</i> Iterasi-2	60
4.2.2.2. <i>Activity Diagram</i> Iterasi-2.....	61
4.2.2.3. <i>Sequence Diagram</i> Iterasi-2	70
4.2.2.4. <i>Class Diagram</i> Iterasi-2	75
4.2.2.5. Perancangan database iterasi-2	77

4.2.2.6. <i>Entity Relationship Diagram</i> iterasi-2	82
4.2.2.7. Rancangan antarmuka (<i>interface</i>) iterasi-2.....	83
4.3 Implementasi	88
4.3.1 Implementasi sistem	88
4.3.2 Pengujian Sistem	104
BAB 5 PENUTUP	141
5.1 Kesimpulan.....	141
5.2 Saran	141
DAFTAR REFERENSI	142



DAFTAR TABEL

Tabel 2.1 Simbol Use Case	10
Tabel 2.2 Simbol Activity Diagram	11
Tabel 2.2 Simbol Activity Diagram (lanjutan).....	12
Tabel 2.3 Simbol Sequence Diagram.....	13
Tabel 2.3 Simbol Sequence Diagram(lanjutan).....	14
Tabel 2.4 Multiplicity Class Diagram	17
Tabel 2.5 Persentase Penilaian	21
Tabel 4.1 Kebutuhan Pengguna	27
Tabel 4.2 Kebutuhan Fungsional.....	28
Tabel 4.2 Kebutuhan Fungsional (lanjutan)	29
Tabel 4.2 Kebutuhan Fungsional (lanjutan)	30
Tabel 4.3 Tabel sekolah iterasi-1.....	50
Tabel 4.4 Tabel jurusan iterasi-1.....	50
Tabel 4.5 Tabel kelas iterasi-1.....	51
Tabel 4.6 Tabel siswa iterasi-1	51
Tabel 4.7 Tabel guru/staff iterasi-1.....	52
Tabel 4.8 Tabel users iterasi-1	52
Tabel 4.9 Tabel anggota iterasi-1.....	53
Tabel 4.10 Tabel transaksi iterasi-1	53
Tabel 4.11 Tabel buku iterasi-1	54
Tabel 4.12 Tabel denda iterasi-1.....	54
Tabel 4.13 Tabel sekolah iterasi-2.....	77
Tabel 4.14 Tabel jurusan iterasi-2	77
Tabel 4.15 Tabel kelas iterasi-2.....	78
Tabel 4.16 Tabel siswa iterasi-2	78
Tabel 4.17 Tabel guru/staff iterasi-2.....	79
Tabel 4.18 Tabel users iterasi-2	79
Tabel 4.19 Tabel anggota iterasi-2.....	80
Tabel 4.20 Tabel transaksi iterasi-2	80
Tabel 4.21 Tabel buku iterasi-2	81
Tabel 4.22 Tabel denda iterasi-2.....	81

Tabel 4.23 DDL tabel buku	88
Tabel 4.24 DDL tabel jurusan	89
Tabel 4.25 DDL tabel denda	89
Tabel 4.26 DDL tabel kelas	89
Tabel 4.27 DDL tabel anggota	89
Tabel 4.28 DDL tabel sekolah	90
Tabel 4.29 DDL tabel staff	90
Tabel 4.30 DDL tabel siswa	90
Tabel 4.31 DDL tabel transaksi	91
Tabel 4.32 DDL tabel user	91
Tabel 4.33 Potongan source code anggota	92
Tabel 4.33 Potongan source code anggota (lanjutan)	93
Tabel 4.33 Potongan source code anggota (lanjutan)	94
Tabel 4.34 Potongan source code transaksi	95
Tabel 4.35 Potongan source code buku	96
Tabel 4.35 Potongan source code buku (lanjutan)	97
Tabel 4.36 Potongan source code denda	98
Tabel 4.37 Potongan source code login	99
Tabel 4.38 Potongan source code registrasi	99
Tabel 4.39 Hasil Uji Black Box	105
Tabel 4.39 Hasil Uji Black Box (lanjutan)	106
Tabel 4.39 Hasil Uji Black Box (lanjutan)	107
Tabel 4.39 Hasil Uji Black Box (lanjutan)	108
Tabel 4.39 Hasil Uji Black Box (lanjutan)	109
Tabel 4.39 Hasil Uji Black Box (lanjutan)	110
Tabel 4.39 Hasil Uji Black Box (lanjutan)	111
Tabel 4.39 Hasil Uji Black Box (lanjutan)	112
Tabel 4.39 Hasil Uji Black Box (lanjutan)	113
Tabel 4.39 Hasil Uji Black Box (lanjutan)	114
Tabel 4.39 Hasil Uji Black Box (lanjutan)	115
Tabel 4.39 Hasil Uji Black Box (lanjutan)	116
Tabel 4.39 Hasil Uji Black Box (lanjutan)	117

Tabel 4.39 Hasil Uji Black Box (lanjutan)	118
Tabel 4.39 Hasil Uji Black Box (lanjutan)	119
Tabel 4.39 Hasil Uji Black Box (lanjutan)	120
Tabel 4.39 Hasil Uji Black Box (lanjutan)	121
Tabel 4.39 Hasil Uji Black Box (lanjutan)	122
Tabel 4.39 Hasil Uji Black Box (lanjutan)	123
Tabel 4.39 Hasil Uji Black Box (lanjutan)	124
Tabel 4.39 Hasil Uji Black Box (lanjutan)	125
Tabel 4.39 Hasil Uji Black Box (lanjutan)	126
Tabel 4.39 Hasil Uji Black Box (lanjutan)	127
Tabel 4.39 Hasil Uji Black Box (lanjutan)	128
Tabel 4.39 Hasil Uji Black Box (lanjutan)	129
Tabel 4.39 Hasil Uji Black Box (lanjutan)	130
Tabel 4.39 Hasil Uji Black Box (lanjutan)	131
Tabel 4.39 Hasil Uji Black Box (lanjutan)	132
Tabel 4.39 Hasil Uji Black Box (lanjutan)	133
Tabel 4.39 Hasil Uji Black Box (lanjutan)	134
Tabel 4.39 Hasil Uji Black Box (lanjutan)	135
Tabel 4.39 Hasil Uji Black Box (lanjutan)	136
Tabel 4.39 Hasil Uji Black Box (lanjutan)	137
Tabel 4.40 Hasil Pengujian UAT	138

DAFTAR GAMBAR

Gambar 2.1 Tahapan RAD	7
Gambar 2.2 Cara Kerja MVC	9
Gambar 2.3 Contoh Use Case	11
Gambar 2.4 Contoh Activity Diagram	13
Gambar 2.5 Contoh Sequence Diagram.....	15
Gambar 2.6 Asosiasi	15
Gambar 2.7 Agregasi.....	16
Gambar 2.8 Composition	16
Gambar 2.9 Dependency	16
Gambar 2.10 Generalisasi	16
Gambar 3.1 Tahapan Penelitian.....	23
Gambar 4.1 Aturan Penomoran Kebutuhan	28
Gambar 4.2 Use Case Diagram sistem iterasi-1	31
Gambar 4.3 Activity Diagram peminjaman buku (User) iterasi-1.....	32
Gambar 4.4 Activity Diagram tambah anggota (Admin) iterasi-1	32
Gambar 4.5 Activity Diagram edit anggota (Admin) iterasi-1.....	33
Gambar 4.6 Activity Diagram hapus anggota (Admin) iterasi-1	34
Gambar 4.7 Activity Diagram tambah transaksi (Admin) iterasi-1	35
Gambar 4.8 Activity Diagram edit transaksi (Admin) iterasi-1	36
Gambar 4.9 Activity Diagram mengembalikan buku (Admin) iterasi-1	36
Gambar 4.10 Activity Diagram tambah buku (Admin) iterasi-1	37
Gambar 4.11 Activity Diagram edit buku (Admin) iterasi-1.....	38
Gambar 4.12 Activity Diagram hapus buku (Admin) iterasi-1	39
Gambar 4.13 Activity Diagram tambah denda (Admin) iterasi-1	40
Gambar 4.14 Activity Diagram edit denda (Admin) iterasi-1.....	41
Gambar 4.15 Activity Diagram hapus denda (Admin) iterasi-1	42
Gambar 4.16 Sequence Diagram login (User) iterasi-1	43
Gambar 4.17 Sequence Diagram login (Admin) iterasi-1	43
Gambar 4.18 Sequence Diagram peminjaman (User) iterasi-1.....	44
Gambar 4.19 Sequence Diagram kelola anggota (Admin) iterasi-1.....	45
Gambar 4.20 Sequence Diagram kelola transaksi (Admin) iterasi-1	46

Gambar 4.21 Sequence Diagram kelola buku (Admin) iterasi-1.....	47
Gambar 4.22 Sequence Diagram kelola denda (Admin) iterasi-1.....	48
Gambar 4.23 Class Diagram Sistem Informasi Perpustakaan iterasi-1.....	49
Gambar 4.24 Entity Relationship Diagram Sistem Informasi Perpustakaan iterasi-1	55
Gambar 4.25 Halaman login iterasi-1	56
Gambar 4.26 Halaman register iterasi-1.....	57
Gambar 4.27 Halaman dashboard iterasi-1	58
Gambar 4.28 Halaman anggota (Admin) iterasi-1	58
Gambar 4.29 Halaman transaksi (Admin) iterasi-1.....	59
Gambar 4.30 Halaman buku (Admin) iterasi-1	59
Gambar 4.31 Halaman denda (Admin) iterasi-1	60
Gambar 4.32 Use case diagram sistem iterasi-2.....	61
Gambar 4.33 Activity Diagram user iterasi-2	62
Gambar 4.34 Activity Diagram tambah anggota (Admin) iterasi-2	62
Gambar 4.35 Activity Diagram edit anggota (Admin) iterasi-2.....	63
Gambar 4.36 Activity Diagram hapus anggota (Admin) iterasi-2	64
Gambar 4.37 Activity Diagram tambah transaksi (Admin) iterasi-2	65
Gambar 4.38 Activity Diagram edit transaksi (Admin) iterasi-2	65
Gambar 4.39 Activity Diagram mengembalikan buku (Admin) iterasi-2	66
Gambar 4.40 Activity Diagram tambah buku (Admin) iterasi-2	67
Gambar 4.41 Activity Diagram edit buku (Admin) iterasi-2.....	67
Gambar 4.42 Activity Diagram hapus buku (Admin) iterasi-2	68
Gambar 4.43 Activity Diagram tambah denda (Admin) iterasi-2	69
Gambar 4.44 Activity Diagram edit denda (Admin) iterasi-2.....	69
Gambar 4.45 Activity Diagram hapus denda (Admin) iterasi-2	70
Gambar 4.46 Sequence Diagram login (User) iterasi-2	71
Gambar 4.47 Sequence Diagram login (Admin) iterasi-2	71
Gambar 4.48 Sequence Diagram kelola anggota (Admin) iterasi-2.....	72
Gambar 4.49 Sequence Diagram kelola transaksi (Admin) iterasi-2	73
Gambar 4.50 Sequence Diagram kelola buku (Admin) iterasi-2.....	74
Gambar 4.51 Sequence Diagram kelola denda (Admin) iterasi-2.....	75
Gambar 4.52 Class diagram sistem informasi perpustakaan iterasi-2	76

Gambar 4.53 Entity Relationship Diagram Sistem Informasi Perpustakaan iterasi-2	82
Gambar 4.54 Halaman login iterasi-2	83
Gambar 4.55 Halaman register iterasi-2	84
Gambar 4.56 Halaman depan	84
Gambar 4.57 Halaman dashboard user	85
Gambar 4.58 Halaman dashboard iterasi-2	86
Gambar 4.59 Halaman anggota (Admin) iterasi-2	86
Gambar 4.60 Halaman transaksi (Admin) iterasi-2	87
Gambar 4.61 Halaman buku (Admin) iterasi-2	87
Gambar 4.62 Halaman denda (Admin) iterasi-2	88
Gambar 4.63 Implementasi database	91
Gambar 4.64 Implementasi halaman dashboard	100
Gambar 4.65 Implementasi halaman anggota	100
Gambar 4.66 Implementasi halaman transaksi	101
Gambar 4.67 Implementasi halaman buku	101
Gambar 4.68 Implementasi halaman denda	102
Gambar 4.69 Implementasi halaman depan	102
Gambar 4.70 Implementasi halaman login	103
Gambar 4.71 Implementasi halaman register	103
Gambar 4.72 Implementasi halaman dashboard user	104
Gambar 4.73 Hasil Pengujian UAT	139

LAMPIRAN

Lampiran A.1 Hasil Wawancara Petugas.....	147
Lampiran A.2 Hasil Wawancara Ketua Perpustakaan.....	149
Lampiran B.1 Iterasi Workshop Design RAD	151



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Pada SMK Negeri 11 Malang, penelitian dilakukan di tempat peneliti melaksanakan kegiatan Praktik Pengalaman Lapangan (PPL). Berdasarkan hasil dari observasi yang dilakukan, peneliti melihat pelayanan pada perpustakaan masih manual menggunakan buku catatan setiap kali ada yang meminjam buku, terutama siswa-siswi SMK Negeri 11 Malang. Kebutuhan siswa-siswi dalam mendapatkan sumber pengetahuan tidak hanya bergantung kepada guru, tetapi juga diperlukan dari sumber-sumber yang lain seperti perpustakaan. Perpustakaan termasuk sarana yang disediakan sekolah yang berguna sebagai penunjang pembelajaran siswa-siswi yang dapat memudahkan dalam menambah pengetahuan siswa-siswi. Menurut (Lasa, 2007) dalam (Yusri, 2015) perpustakaan yaitu gabungan bangunan fisik untuk tempat buku yang dikumpulkan secara tersusun menurut sistem tertentu maupun keperluan tertentu.

Berdasarkan hasil wawancara peneliti dengan ibu Rahmah Widarti selaku ketua perpustakaan yang dilaksanakan pada tanggal 15 Februari 2019, mengatakan bahwa perpustakaan di SMK Negeri 11 Malang, terdapat aplikasi SLA (*Smart Library Automation*) berbasis desktop yang diciptakan oleh Bapak Moh. Sholeh, S.T dalam memberikan pelayanan kepada pengunjung perpustakaan. Menurut (Prakoso, 2017) mengatakan bahwa “SLA merupakan sebuah perangkat lunak untuk manajemen perpustakaan yang dirancang dan dikembangkan agar dapat membantu pengelolaan perpustakaan modern dengan berbagai macam fasilitas untuk memberikan kecepatan, kemudahan dan keakuratan proses pengelolaan perpustakaan yang berbasis teknologi informasi”. Aplikasi perpustakaan berbasis desktop tersebut hanya dapat diakses oleh admin perpustakaan pada jam kerja saja dan juga tidak bisa diakses dikomputer yang berbeda. Selain itu, dilakukan wawancara kepada petugas perpustakaan yang mengatakan bahwa pada saat ini dalam pelayanan peminjaman buku masih manual dengan mencatat dibuku catatan meskipun sudah ada aplikasi SLA (*Smart Library Automation*), karena aplikasi SLA (*Smart Library Automation*) dalam menginputkan data masih manual harus satu per satu. Sehingga dapat disimpulkan bahwa permasalahan yang terdapat pada sistem perpustakaan SMK Negeri 11 Malang yaitu aplikasi SLA (*Smart Library Automation*) kurang optimal dalam penggunaannya karena tidak bisa diakses kapanpun dan dimanapun serta menyulitkan petugas perpustakaan dalam mengelola data-data perpustakaan.

Dari wawancara yang terlampir pada lampiran A.1 didapat analisis kebutuhan sistem informasi perpustakaan yang diinginkan pengguna. Diantaranya yaitu, yang pertama sistem mampu diakses admin dimanapun dan kapanpun tidak hanya jam kerja. Kedua, admin dapat melakukan pengelolaan data-data dengan mudah seperti menginputkan data-data buku. Ketiga, sistem mudah dipahami semua pengguna. Keempat, sistem dapat menambahkan anggota baru perpustakaan,

mengedit, maupun menghapus anggota perpustakaan. Kelima, sistem dapat memberikan pelayanan transaksi peminjaman.

Untuk memenuhi kebutuhan pengguna yang telah diuraikan tersebut, dilakukan pembangunan sebuah sistem informasi perpustakaan berbasis website. Definisi sistem informasi menurut (Hariadi, 2013) menjelaskan bahwa sistem informasi merupakan sistem yang terdapat didalam organisasi yang memungkinkan bertemunya kebutuhan pengolahan transaksi harian, mendukung operasi, yang bersifat manajerial dan juga kegiatan strategi dari suatu organisasi serta menyediakan pihak luar tertentu dengan laporan – laporan yang dibutuhkan. Dengan adanya sistem informasi perpustakaan berbasis website, dapat membantu petugas perpustakaan dalam pengelolaan data – data perpustakaan yang bisa diakses kapan pun dan dimana pun. Selain itu, sistem informasi perpustakaan berbasis website juga dapat mempermudah anggota perpustakaan ketika melakukan transaksi peminjaman maupun pengembalian buku serta juga bisa diakses anggota perpustakaan dalam melihat informasi perpustakaan kapan pun dan dimana pun.

Metode yang digunakan dalam pembangunan sistem informasi perpustakaan berbasis website yaitu menggunakan metode *Rapid Application Development* (RAD). Alasan menggunakan metode RAD karena metode RAD dapat mempersingkat waktu dalam pembangunan sistem informasi daripada metode tradisional. Menurut (Pressman, 2001) menjelaskan bahwa *Rapid Application Development* (RAD) adalah suatu model pengembangan *software* tambahan yang menekankan siklus pengembangan yang sangat singkat. Tahapan – tahapan pada RAD menurut (Kendal, 2011) terdapat tiga fase. Tiga fase tersebut yaitu *requirements planning*, *RAD design workshop*, dan *implementation*. Hasil yang akan didapat dengan metode RAD yaitu sistem dapat berfungsi dengan penuh sesuai dengan kebutuhan pengguna.

Berdasarkan latar belakang tersebut, maka dari itu dilakukan penelitian yang berjudul “PEMBANGUNAN SISTEM INFORMASI PERPUSTAKAAN BERBASIS WEB MENGGUNAKAN METODE RAPID APPLICATION DEVELOPMENT (RAD) (STUDI PADA : SMK NEGERI 11 MALANG)”.

1.2 Rumusan Masalah

Berdasarkan dari latar belakang tersebut, maka didapat rumusan masalah sebagai berikut:

1. Bagaimana hasil analisis kebutuhan sistem informasi perpustakaan berbasis web pada SMK Negeri 11 Malang?
2. Bagaimana hasil *workshop design* perancangan sistem informasi perpustakaan berbasis web pada SMKN 11 Malang?
3. Bagaimana hasil implementasi sistem informasi perpustakaan berbasis web pada SMKN 11 Malang?
4. Bagaimana hasil pengujian rancangan sistem informasi perpustakaan berbasis web pada SMK Negeri 11 Malang?

1.3 Tujuan

Tujuan berdasarkan rumusan masalah tersebut yaitu :

1. Menganalisis kebutuhan sistem informasi perpustakaan berbasis web pada SMK Negeri 11 Malang.
2. Merancang sistem informasi perpustakaan berbasis web dengan metode *Rapid Application Development* (RAD) yang bertujuan memudahkan pengelolaan data-data perpustakaan dan memudahkan pengguna dalam melakukan transaksi di perpustakaan SMK Negeri 11 Malang.
3. Mengetahui hasil kesimpulan perancangan sistem informasi perpustakaan berbasis web dengan metode *Rapid Application Development* (RAD).
4. Mengetahui hasil pengujian rancangan sistem informasi perpustakaan berbasis web tersebut.

1.4 Manfaat

Manfaat dari penelitian ini yaitu :

1. Dapat membantu petugas/staff perpustakaan dalam manajemen data-data perpustakaan.
2. Dapat memudahkan pengunjung perpustakaan dalam melakukan transaksi peminjaman dan pengembalian buku maupun melihat informasi perpustakaan.
3. Sistem dapat diakses pengguna kapan pun dan dimana pun.

1.5 Batasan Masalah

Batasan masalah dalam penelitian yaitu :

1. Pengujian sistem informasi perpustakaan berbasis web menggunakan *black box testing* dan juga menggunakan *User Acceptance Testing* (UAT).
2. Proses iterasi pada workshop desain hanya sampai dua iterasi.
3. Pembangunan sistem hanya sampai pada tahap pengujian.

1.6 Sistematika Pembahasan

Berikut ini sistematika pembahasan penelitian:

BAB I PENDAHULUAN

Pada bab ini berisi tentang latar belakang, rumusan masalah, tujuan, manfaat, batasan masalah, serta sistematika pembahasan.

BAB II LANDASAN KEPUSTAKAAN

Pada bab ini berisi uraian teori tentang pengertian perpustakaan, sistem informasi, model pengembangan, *Unified Modeling Language*, dan juga pengujian sistem.

BAB III METODOLOGI

Pada bab ini berisi tentang metode pengumpulan data, tahapan penelitian, waktu dan lokasi penelitian, metode pengembangan, dan juga implementasi.

BAB IV HASIL DAN PEMBAHASAN

Pada bab ini berisi hasil dari perencanaan syarat-syarat, proses desain RAD yang meliputi proses iterasi ke-1, proses iterasi ke-2, dan implementasi yang meliputi implementasi sistem serta hasil pengujian sistem.

BAB V PENUTUP

Pada bab ini berisi kesimpulan dan saran yang didapatkan dari keseluruhan proses yang telah dilakukan.



BAB 2 LANDASAN KEPUSTAKAAN

2.1 Perpustakaan

Pengertian perpustakaan menurut (Renatha, 2015) mengatakan bahwa perpustakaan adalah suatu sarana untuk memperoleh pengetahuan selain dari kegiatan belajar mengajar di sekolah dan universitas. Pendapat lainnya menurut (Sutarno, 2006) dalam (Abigail, 2018) menyatakan bahwa perpustakaan merupakan ruangan bagian dari bangunan yang berisi macam-macam buku-buku yang tersusun sedemikian rupa sehingga pembaca dapat dengan mudah mencari dan menggunakan buku. Sedangkan dalam pengertian lainnya, perpustakaan yaitu koleksi buku-buku dan juga majalah.

Dari pengertian perpustakaan tersebut, maka dapat ditarik kesimpulan bahwa perpustakaan adalah suatu fasilitas yang disediakan sekolah maupun institusi yang bertujuan untuk menunjang proses pembelajaran menjadi lebih mudah dan dapat dijangkau semua kalangan.

2.1.1 Jenis Perpustakaan

Berikut berbagai macam perpustakaan menurut (Saleh, 2014) yaitu:

1. Perpustakaan Nasional
Perpustakaan yang bertujuan untuk menyimpan semua bahan pustaka tercetak, terekam, serta multimedia yang diterbitkan oleh negara tersebut.
2. Perpustakaan Umum
Perpustakaan yang didirikan oleh masyarakat umum yang biayanya diberikan oleh masyarakat itu sendiri baik secara langsung (swadaya) maupun tidak langsung seperti melalui pajak.
3. Perpustakaan Khusus
Perpustakaan khusus adalah perpustakaan lembaga pemerintah maupun instansi swasta. Jenis perpustakaan tersebut berada pada lembaga tertentu, yang bertujuan untuk mendukung institusi dengan memberikan informasi bagi karyawan di lingkungan agensi.
4. Perpustakaan Sekolah
Merupakan salah satu fasilitas yang bertujuan membantu siswa-siswi untuk mencapai tujuan pendidikan.
5. Perpustakaan Perguruan Tinggi
Perpustakaan yang berlokasi di universitas, perguruan tinggi, dan pendidikan tinggi lainnya, yang bertujuan untuk mendukung pelaksanaan program pendidikan, penelitian dan juga pengabdian kepada masyarakat melalui pelayanan informasi.

2.2 Sistem Informasi

Sistem informasi didefinisikan oleh Robert A. Laitch dan K. Roscoe Bavis dalam (Hariadi, 2013) bahwa "Sistem informasi yaitu sistem di dalam organisasi yang menyatukan kebutuhan pemrosesan transaksi harian, mendukung operasi, adalah kegiatan manajerial dan strategis dari suatu organisasi dan menyediakan laporan yang diperlukan pihak luar tertentu". Sedangkan pendapat lainnya (Sutanta, 2011) dalam (Abigail, 2018) menjelaskan sistem informasi merupakan gabungan subsistem yang saling terhubung, berkumpul, dan membentuk satu kesatuan, berinteraksi satu sama lain, dan bekerja bersama antara bagian satu dengan bagian lainnya dengan cara tertentu untuk melakukan fungsi pengelolaan data, menerima input kemudian memproses dan menghasilkan *output* dalam bentuk informasi sebagai dasar untuk pengambilan keputusan yang bermanfaat dan memiliki nilai-nilai nyata yang dapat dirasakan sebagai hasilnya baik sekarang dan di masa depan, mendukung strategi operasional, manajerial dan organisasi dengan menggunakan berbagai sumber daya yang tersedia untuk fungsi untuk mencapai tujuan yang diinginkan.

Dari pengertian tersebut, maka dapat disimpulkan bahwa sistem informasi merupakan suatu sistem yang dapat memudahkan pekerjaan manusia melalui suatu perangkat lunak yang dirancang khusus menggunakan teknologi internet untuk melakukan suatu perintah sesuai dengan kebutuhan pengguna.

2.2.1 Sistem Informasi Perpustakaan

Sistem informasi perpustakaan menurut (Siregar, 2007) dalam (Saragih, 2017) menjelaskan sistem informasi perpustakaan yaitu sistem dalam organisasi layanan publik yang menyatukan kebutuhan pemrosesan transaksi pinjaman, pengembalian dan perpanjangan buku dan laporan harian, laporan bulanan atau tahunan untuk mendukung operasi, manajerial dan kegiatan organisasi dan menyediakan pihak luar tertentu dengan laporan yang diperlukan.

2.3 Model Pengembangan

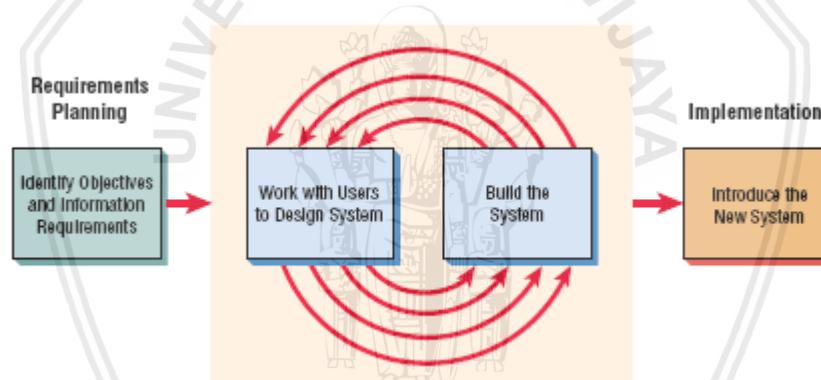
2.3.1 Software Development Life Cycle (SDLC)

Pengertian SDLC menurut (Putri, 2018) menjelaskan bahwa SDLC (*Software Development Life Cycle*) merupakan kerangka kerja terstruktur yang berisi proses sekuensial di mana sistem informasi dikembangkan. Sementara itu menurut (Rajkumar, 2019) menjelaskan *Software Development Life Cycle* (SDLC) bertujuan untuk menghasilkan suatu sistem berkualitas cukup tinggi yang dapat memenuhi atau melampaui harapan pelanggan, bekerja dengan efektif, efisien dalam infrastruktur teknologi informasi saat ini, yang direncanakan, tidak mahal untuk dipelihara, dan menghemat biaya untuk meningkat. Ada banyak model SDLC, tetapi dalam penelitian ini menggunakan model Rapid Application Development (RAD). Menurut (Martin, 1991) dalam (Setiawan, 2011) menjelaskan "*Rapid Application Development* (RAD) merupakan siklus pengembangan yang didesain dengan hasil kualitas cukup tinggi daripada yang dicapai dengan siklus hidup

tradisional dalam memberikan pengembangan yang lebih cepat. Ini didesain untuk mengambil keuntungan maksimal dari pengembangan perangkat lunak yang telah berkembang baru-baru ini”.

2.3.2 Rapid Application Development (RAD)

Rapid Application Development (RAD) menurut (Pressman, 2001) merupakan suatu model dari proses pengembangan perangkat lunak tambahan yang menekankan siklus pengembangan yang tidak terlalu lama. Sedangkan *Rapid Application Development (RAD)* menurut (Aswati, 2016), "RAD yaitu proses pengembangan perangkat lunak sekuensial linier yang menekankan siklus pengembangan dalam waktu yang lebih cepat. RAD menggunakan metode iteratif dalam mengembangkan sistem di mana model kerja dibangun pada awal tahap pengembangan dengan tujuan menentukan persyaratan (persyaratan) pengguna dan kemudian dihapus". Dari beberapa definisi ini, metode *Rapid Application Development (RAD)* adalah metode pengembangan aplikasi yang dilakukan dengan periode waktu yang lebih singkat. Berikut ini adalah tahapan RAD menurut (Kendall, 2011) dapat dilihat pada Gambar 2.1:



Gambar 2.1 Tahapan RAD

Sumber : Kendal (2011)

1. Perencanaan syarat-syarat (*Requirement planning*)

Dalam fase perencanaan persyaratan, pengguna dan analis bertemu untuk mengidentifikasi tujuan aplikasi maupun sistem dan juga untuk mengidentifikasi persyaratan informasi dari tujuan itu.

2. Workshop Desain RAD

Fase workshop/lokakarya desain RAD adalah deskripsi fase desain/perancangan dan peningkatan. Antara pengguna dan analis bekerja sama. Pengguna menanggapi prototipe yang ada dan analis mengoreksi modul yang didesain berdasarkan respons pengguna.

3. Implementasi

Setelah analis bekerja dengan pengguna secara intens selama workshop/lokakarya untuk merancang aspek bisnis serta sistem non-teknis,

aspek ini disepakati, dibangun, dan disempurnakan, sistem diuji dan sistem baru diperkenalkan ke organisasi.

Kekurangan RAD menurut (Kendall , 2011) yaitu:

1. Dalam mengerjakan proyek dengan metode RAD bagian analisis berusaha mengerjakan dengan terburu - buru.
2. Kelemahan terkait waktu dan perhatian terhadap detail.
3. Membuat sulit bagi programmer yang tidak berpengalaman untuk menggunakan perangkat ini di mana programmer dan analis diminta untuk menguasai kemampuan baru sementara pada saat yang sama mereka diharuskan bekerja untuk mengembangkan sistem.

2.3.3 *CakePhp*

Definisi *cakephp* menurut (Bari, 2008) *Cake* adalah kerangka pengembangan cepat untuk PHP sumber terbuka yang menggunakan pola desain yang dikenal umum seperti rekaman aktif, pemetaan data asosiasi, pengontrol depan dan MVC yang bertujuan untuk menyediakan kerangka kerja terstruktur dimana pengguna PHP disemua tingkatan untuk dengan cepat mengembangkan aplikasi web yang kuat, tanpa kehilangan fleksibilitas. Pola MVC (*Model View Controller*) pada *cakephp* yaitu pola desain yang global digunakan dalam pengembangan *software*, di mana kode dipisahkan menjadi tiga bagian utama yaitu *model*, *view*, dan *controller*.

Berikut pengertian MVC (*Model View Controller*) menurut (Bari, 2008) :

Model

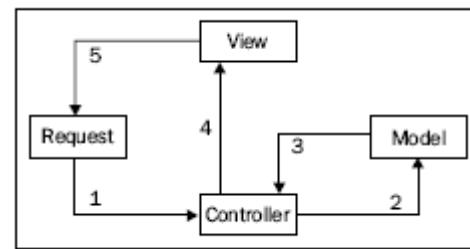
Dalam CakePHP, model mewakili tabel basis data tertentu. Setiap tabel *database* harus memiliki model yang mewakilinya. Jadi, di CakePHP setiap tabel *database* memiliki modelnya sendiri. Semua kode PHP yang terkait dengan mengakses, menambahkan, memodifikasi, atau menghapus catatan dari tabel terdapat dalam model. Model ini juga berisi kode yang mendefinisikan hubungannya dengan model lain.

Controller

Controller di CakePHP, mengontrol aliran aplikasi atau logika aplikasi. Setiap permintaan web diarahkan ke *controller* tertentu tempat input pengguna (POST atau GET data) diterima. Logika *controller* kemudian memutuskan respons apa yang dihasilkan. Logika pengontrol biasanya berisi panggilan ke model untuk mengakses data, dan juga fungsi-fungsi lain seperti pemeriksaan kontrol akses dll. Terakhir, *controller* meneruskan respons (keluaran) ke *view*.

View

View adalah output atau respons yang dikirim kembali ke pengguna setelah permintaan diproses. Mereka pada dasarnya terdiri dari kode *markup* (seperti HTML) dengan kode PHP yang disematkan, tetapi mereka juga bisa berupa bentuk output lain seperti XML, dokumen PDF dll. Tergantung pada situasinya. Tampilan dapat dianggap sebagai lapisan presentasi aplikasi.



Gambar 2.2 Cara Kerja MVC

Sumber : Bari (2008)

Penjelasan :

1. Permintaan dikirim ke *controller*, dengan data pengguna (POST dan GET data).
2. *Controller* memproses permintaan, dan memanggil model untuk mengakses data.
3. Model merespons panggilan *controller*, dengan mengirim atau menyimpan data.
4. *Controller* kemudian mengirimkan data *output* ke *view*.
5. *View* menampilkan data dalam format yang tepat.

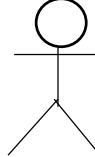
2.4 Unified Modeling Language (UML)

Menurut (Windu, 2013) dalam (Hendini, 2016) *Unified Modeling Language* (UML) adalah bahasa spesifikasi standar yang digunakan untuk mendokumentasikan, menentukan, dan membangun perangkat lunak. Alat yang digunakan dalam desain berorientasi objek berdasarkan UML meliputi:

2.4.1 Use Case Diagram

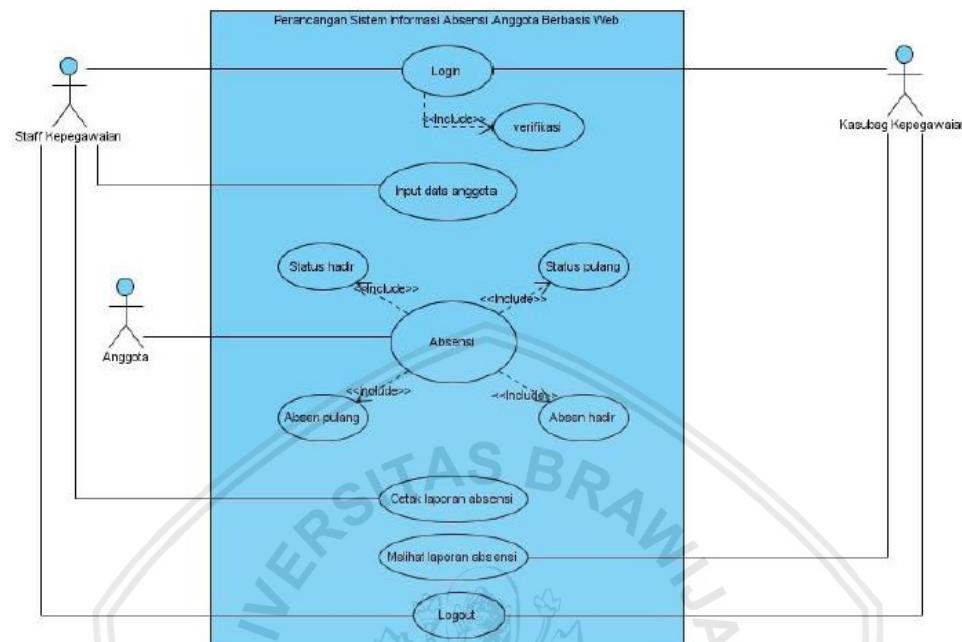
Use case diagram menurut (Hendini, 2016) adalah pemodelan untuk perilaku sistem informasi yang akan dibuat. *Use case* digunakan untuk mencari tahu fungsi apa saja yang ada di sistem informasi dan siapa yang berhak menggunakan fungsi ini. Simbol yang digunakan dalam *Use Case Diagram* antara lain dapat dilihat pada Tabel 2.1:

Tabel 2.1 Simbol Use Case

Gambar	Keterangan
	<i>Use Case</i> ini menggambarkan fungsionalitas yang disediakan oleh sistem sebagai unit yang bertukar pesan antara unit dengan aktor, diekspresikan dengan menggunakan kata kerja.
	Aktor adalah abstraksi dari orang atau sistem lain yang mengaktifkan fungsi sistem target. Untuk mengidentifikasi aktor, pembagian kerja dan tugas yang terkait dengan peran harus ditentukan dalam konteks sistem target. Orang maupun sistem dapat muncul dalam beberapa peran.
	Asosiasi antara aktor dan <i>use case</i> , dijelaskan dengan garis tanpa panah yang menunjukkan siapa atau apa yang meminta interaksi secara langsung dan bukan indikasi data.
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk menunjukkan jika aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i> , yaitu di dalam <i>use case</i> lain (wajib) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, misalnya pemanggilan sebuah fungsi program
	<i>Extend</i> , yaitu perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi

Sumber : Hendini (2016)

Berikut merupakan contoh *use case* :



Gambar 2.3 Contoh Use Case

Sumber : Mulyati (2018)

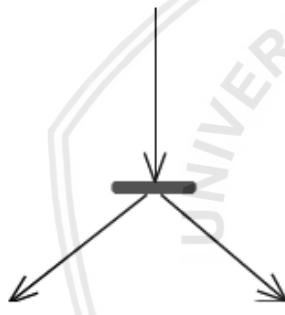
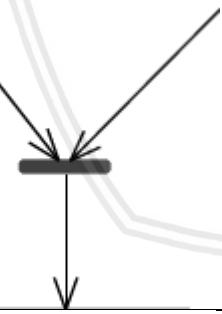
2.4.2 Activity Diagram

Activity diagram menurut (Hendini, 2016) adalah gambaran aliran kerja atau kegiatan dari sebuah sistem. *Activity diagram* dibuat menurut sebuah atau beberapa *use case* pada *use case diagram*. Simbol yang digunakan dalam *Activity Diagram* yaitu:

Tabel 2.2 Simbol Activity Diagram

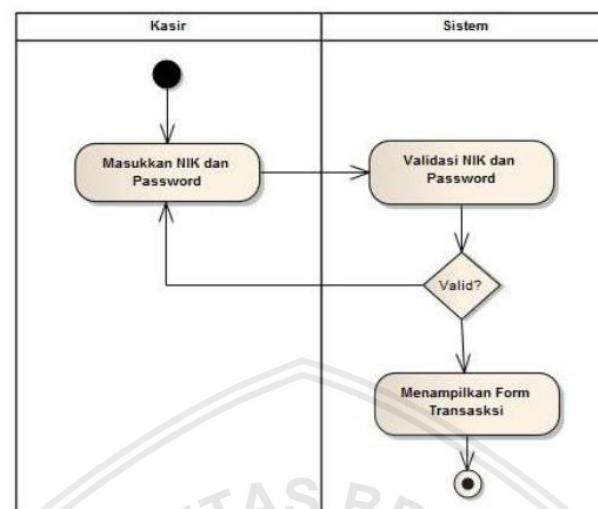
Gambar	Keterangan
	<i>Start Point</i> merupakan awal aktivitas, yang diletakkan pada bagian pojok kiri sebelah atas
	<i>End Point</i> , merupakan akhir aktivitas

Tabel 2.2 Simbol Activity Diagram (lanjutan)

Gambar	Keterangan
	<i>Swimlane</i> , merupakan sebuah cara untuk membagi kelompok aktivitas berdasarkan aktor
	Activities, menjelaskan suatu proses/kegiatan bisnis
	<i>Fork</i> /percabangan bertujuan menunjukkan kegiatan yang dilakukan secara paralel maupun untuk menyatukan dua kegiatan paralel
	<i>Join</i> (penggabungan) atau <i>rake</i> , untuk menunjukkan adanya dekomposisi
	<i>Decision Points</i> , menunjukkan pilihan untuk pengambilan keputusan, <i>true</i> atau <i>false</i>

Sumber : Hendini (2016)

Berikut merupakan contoh *Activity diagram*:



Gambar 2.4 Contoh Activity Diagram

Sumber : Hendini (2016)

2.4.3 Sequence Diagram

Sequence Diagram menurut (Hendini, 2016), menggambarkan perilaku objek pada *use case* dengan menjelaskan waktu hidup objek dan juga pesan yang dikirim dan diterima antar objek satu dengan yang lainnya. Simbol-simbol yang digunakan dalam *Sequence Diagram* yaitu:

Tabel 2.3 Simbol Sequence Diagram

Gambar	Keterangan
	<i>Entity Class</i> , bagian sistem yang isinya kumpulan kelas berupa banyak entitas yang menggambarkan gambaran awal sistem
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi antarmuka atau interaksi antara satu aktor atau lebih dengan sistem

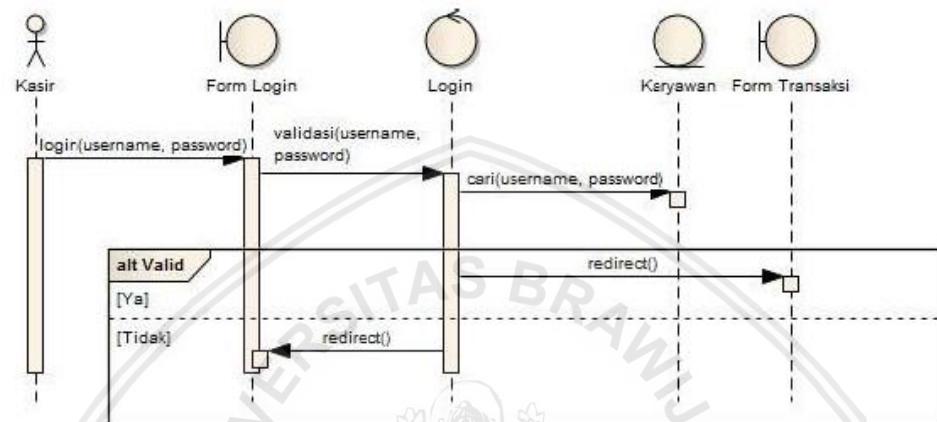
Sumber : Hendini (2016)

Tabel 2.3 Simbol Sequence Diagram(lanjutan)

Gambar	Keterangan
	<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak mempunyai tanggung jawab kepada entitas
	<i>Message</i> , simbol untuk mengirim pesan antar class satu dengan yang lainnya
	<i>Recursive</i> , menjelaskan pengiriman pesan yang dikirim untuk diri sendiri
	<i>Activation</i> , sebuah eksekusi operasi dari objek, panjang kotak yang berbanding lurus dengan banyaknya durasi aktivasi sebuah operasi
	<i>Lifeline</i> , garis dari kumpulan banyak titik yang terhubung dengan objek, dan pada <i>lifeline</i> terdapat <i>activation</i>

Sumber : Hendini (2016)

Berikut merupakan contoh *Sequence Diagram*:



Gambar 2.5 Contoh Sequence Diagram

Sumber : Hendini (2016)

2.4.4 Class Diagram

Class diagram menurut (Hendini, 2016) merupakan hubungan antar kelas dan penjelasan detail setiap kelas di dalam model desain dari sistem, dan juga memperlihatkan aturan serta tanggung jawab entitas yang menentukan perilaku sistem. Sedangkan *class diagram* menurut (Jacobson, 1996) menjelaskan bahwa *class diagram* mengambarkan struktur dan deskripsi suatu *class*, *package* dan objek serta hubungan dengan yang lainnya seperti *containment*, pewarisan, asosiasi, dan lain-lain. Berikut hubungan antar class menurut (Jacobson, 1996) :

1. Asosiasi, dimana menggambarkan hubungan statis antar class.

Contoh :

Gambar 2.6 Asosiasi

Sumber : (Jacobson, 1996)

2. Agregasi, hubungan yang menyatakan relasi mempunyai sebuah.

Contoh:



Gambar 2.7 Agregasi

Sumber : (Jacobson, 1996)

3. *Composition*, sebuah kelas harus merupakan bagian dari *class* yang lain, oleh karena itu *class* tersebut memiliki relasi *composition*.

Contoh:



Gambar 2.8 Composition

Sumber : (Jacobson, 1996)

4. *Dependency*, menunjukkan hubungan operasi pada suatu kelas yang menggunakan class yang lain.

Contoh:



Gambar 2.9 Dependency

Sumber : (Jacobson, 1996)

5. Generalisasi/Pewarisan, menggambarkan hubungan hirarkis antar *class*.

Contoh:



Gambar 2.10 Generalisasi

Sumber : (Jacobson, 1996)

Selain terdapat hubungan antar class pada class diagram terdapat *multiplicity*. Menurut (Jacobson, 1996) menjelaskan *multiplicity* pada asosiasi adalah jumlah banyaknya obyek sebuah *class* yang berelasi dengan obyek *class* lain yang berasosiasi dengan *class* tersebut. Berikut tabel *multiplicity* menurut (Hendini, 2016) :

Tabel 2.4 Multiplicity Class Diagram

Multiplicity	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimal 4

Sumber : Hendini (2016)

2.5 PHP (*Hypertext Preprocessor*)

Menurut (Sutopo, 2017) menjelaskan, “PHP merupakan salah satu *server side* yang didesain khusus untuk aplikasi *web*. Dimana PHP disisipkan diantara bahasa HTML dan oleh karena bahasa *server side*, maka dari itu bahasa PHP akan dieksekusi di server, sehingga yang dikirimkan ke alamat *browser* adalah hasil jadi dalam bentuk HTML, dan kode program PHP tidak akan terlihat”. Sedangkan menurut (Putri, 2013) “PHP (*PHP Hypertext Preprocessor*) adalah bahasa berbentuk *script* yang ditempatkan di bagian server dan juga diproses di server”. Berdasarkan pengertian tersebut dapat ditarik kesimpulan bahwa PHP (*Hypertext Preprocessor*) adalah suatu bahasa *script* yang dirancang yang dapat dikonfigurasikan pada *script* HTML.

2.6 MySQL

Definisi MySQL menurut (Sutopo, 2017) “MySQL adalah salah satu jenis database server yang sangat terkenal. Terkenalnya MySQL disebabkan MySQL menggunakan SQL sebagai bahasa dasar dalam mengakses databasenya. MySQL termasuk jenis RDBMS (*Relational Database Management System*)”. Pada perancangan database terdapat dua teknik yaitu teknik normalisasi dan juga teknik *entity relationship*. Menurut (Kristanto, 1994) menjelaskan bahwa proses normalisasi merupakan proses pengelompokan data elemen menjadi tabel-tabel yang menunjukkan entity dan relasinya. Terdapat *field/atribut* kunci di dalam setiap file suatu tabel database. Diantaranya yaitu *primary key* (kunci primer), *foreign key* (kunci tamu). Pengertian *primary key* menurut (Kristanto, 1994) mengatakan, “*Primary Key* adalah satu atribut satu set minimal atribut yang tidak hanya bisa identifikasi secara unik suatu kejadian spesifik, tetapi juga dapat mewakilkan setiap kejadian dari suatu *entity*”. Sedangkan *Foreign Key* menurut (Kristanto, 1994) mengatakan, “*Foreign Key* adalah satu atribut (atau satu set atribut) yang dapat melengkapi satu *relationship* yang menunjukkan ke induknya”.

2.7 Entity Relationship Diagram

Definisi *Entity Relationship Diagram* menurut (Iswandy, 2015) mengatakan bahwa “*Entity Relationship Diagram* adalah gambaran bentuk hubungan antara banyak file yang ada, dimana *entity* merupakan suatu kesatuan atau kesimpulan data yang mempunyai karakteristik yang sama”. Sedangkan (Zumaroh, 2004) dalam (Suryati, 2010) menjelaskan “Diagram *entity* (ERD) adalah suatu dokumentasi data yang memperlihatkan hubungan yang ada diantara *entity* tersebut”.

2.8 Data Definition Language (DDL)

Definisi *Data Definition Language* menurut (Kadir, 1999) menjelaskan bahwa DDL atau *Data Definition Language* adalah suatu kelompok perintah yang berfungsi untuk mendefinisikan tabel, atribut (kolom), atribut basis data, batasan terhadap atribut, serta hubungan antar tabel. Berikut termasuk dalam kelompok DDL adalah CREATE, ALTER, dan DROP. CREATE digunakan untuk membuat tabel, indeks, atau pandangan. ALTER digunakan untuk mengubah struktur tabel. DROP untuk menghapus tabel, indeks atau pandangan.

2.9 PhpMyAdmin

Definisi *phpMyAdmin* menurut (Putri, 2013), “*phpMyAdmin* dapat membantu programmer membuat sebuah database, mengisi data, membuat tabel, dan lain sebagainya dengan mudah, dan tanpa harus menghafal setiap baris perintahnya. *PhpMyAdmin* adalah bagian untuk mengelola basis data MySQL yang ada di komputer”. Sedangkan menurut (Rahman, 2013) dalam (Atmoko, 2016), “*phpMyAdmin* merupakan *software* berbasis pemrograman PHP (*Hypertext Preprocessor*) yang dipergunakan sebagai administrator MySQL melalui browser yang digunakan untuk managemen *database*”. Dari penjelasan tersebut, maka dapat ditarik kesimpulan bahwa *phpMyAdmin* merupakan suatu *software* yang *open source* dimana admin dapat mengelola database melalui browser.

2.10 XAMPP

Pengertian XAMPP menurut (Aryanto, 2013), menjelaskan bahwa “XAMPP merupakan suatu aplikasi web server instan yang memiliki tujuan untuk membangun sebuah aplikasi web. Fungsi XAMPP adalah sebagai server yang berdiri sendiri (*localhost*), terdiri dari program Apache HTTP Server, MySQL, Database, dan penerjemah bahasa yang ditulis dengan pemrograman php”. Sedangkan menurut (Priyanti, 2013), mengatakan XAMPP merupakan server yang berdiri sendiri sebagai perangkat lunak bebas yang mendukung banyak sistem operasi dan merupakan kompilasi dari beberapa macam program. Terdiri dari program Apache HTTP Server, MySQL *database*, dan penerjemah yang dituliskan dengan bahasa pemrograman PHP.

2.11 Visual Studio Code

Pengertian *Visual Studio Code* menurut (Wijaya, 2016) menjelaskan bahwa *Visual Studio Code* merupakan aplikasi kode editor yang tidak hanya tersedia untuk Windows, namun tersedia juga untuk sistem operasi Linux dan Mac OS. *Visual Studio Code* mendukung berbagai jenis bahasa pemrograman seperti *Javascript*, Java, PHP, C++, C#, Go, JSON, dan lainnya. Aplikasi editor ini dapat secara otomatis mengetahui jenis bahasa pemrograman yang digunakan dan memberikan macam-macam warna sesuai dengan masing-masing fungsi dalam rangkaian kode tersebut.

2.12 Visual Paradigm

Menurut (Khotimah, 2018) menjelaskan bahwa *visual paradigm* adalah sebuah model *software* dengan sistem visualisasi yang memungkinkan model dapat digunakan untuk merepresentasi banyak proyek lain dan juga dilengkapi banyak fitur yang ada didalamnya hingga pada menganalisis sebuah proyek yang akan dikerjakan.

2.13 Web

Menurut (Sutopo, 2017) definisi *web* adalah “*Web* merupakan suatu cara baru sebagai teknologi sistem informasi yang dapat menghubungkan data dari banyak sumber dan layanan yang beragam macamnya di internet.” Dengan adanya sebuah *web*, kita bisa mencari banyak data yang dibutuhkan dengan waktu yang cepat dan juga dapat mengaksesnya kapanpun dan dimanapun kita berada. Sehingga dapat menghemat waktu dan juga biaya.

2.14 Pengujian

Pengujian sistem ini dilakukan menggunakan pengujian *black box* dan pengujian *User Acceptance Test* (UAT). Berikut pengertian dari pengujian *black box* dan pengujian *User Acceptance Test* (UAT) :

2.14.1 Pengujian *Black Box*

Pengertian pengujian *black box* menurut (Pressman, 2001), menjelaskan bahwa pengujian *black box* disebut pengujian perilaku yang berfokus pada persyaratan fungsional suatu perangkat lunak, dimana pengujian yang memungkinkan untuk mendapatkan serangkaian kondisi masukan yang sepenuhnya akan menjalankan semua syarat-syarat fungsional untuk suatu program. *Black box testing* berupaya menemukan kesalahan dalam kategori berikut:

1. Suatu fungsi yang salah atau hilang.
2. Kesalahan desain antarmuka.
3. Dan kesalahan dalam struktur data maupun akses basis data eksternal.
4. Kesalahan perilaku atau kinerja.
5. Kesalahan inisialisasi dan terminasi.

Sedangkan menurut (Nidhra, 2012), menjelaskan bahwa pengujian *black box* merupakan pengujian berdasarkan spesifikasi persyaratan dan tidak perlu memeriksa kode dalam pengujian *black box* dan murni dilakukan berdasarkan sudut pandang pelanggan. Pengujian *black box* memainkan peran penting dalam pengujian perangkat lunak, hal ini dikarenakan membantu validasi fungsionalitas keseluruhan sistem. Pengujian *black box* dilakukan berdasarkan persyaratan pelanggan, sehingga persyaratan yang tidak lengkap atau tidak dapat diprediksi dapat dengan mudah diidentifikasi dan dapat diatasi nanti. Pengujian *black box* dilakukan berdasarkan perspektif pengguna akhir. Pentingnya pengujian kotak hitam ini menangani input yang valid dan tidak valid dari perspektif pelanggan.

Keuntungan utama dari pengujian *black box* adalah penguji tidak perlu memiliki pengetahuan tentang bahasa pemrograman tertentu, tidak hanya bahasa pemrograman tetapi juga pengetahuan tentang implementasi. Dalam pengujian *black box*, programmer dan penguji tidak tergantung satu sama lain. Keuntungan lain adalah pengujian dilakukan dari sudut pandang pengguna.

2.14.2 Pengujian *User Acceptance Testing* (UAT)

Pengujian *User Acceptance Testing* (UAT) menurut (Mutiara, 2014) menjelaskan bahwa *User Acceptance Testing* (UAT) merupakan suatu proses uji yang dilakukan *user* dan menghasilkan dokumen untuk dijadikan bukti bahwa pengembangan aplikasi/sistem dapat diterima *user* dan hasilnya dianggap memenuhi kebutuhan pengguna. Sedangkan pengujian *User Acceptance Test* (UAT) menurut (Betha, 2006) dalam (Wahyuningsih, 2017) yaitu suatu proses pengujian sistem yang dilakukan oleh pengguna untuk menghasilkan dokumen untuk dijadikan bukti bahwa pengembangan perangkat lunak dapat diterima oleh pengguna, apabila hasil pengujian sudah bisa dianggap memenuhi kebutuhan dari pengguna. Terdapat 3 proses psikologis untuk menggunakan teknologi tertentu, dan memeriksa peran pengalaman penggunaan perilaku sistem langsung dalam pembentukan menurut (Davis, 2004) yaitu *usefulness*, *ease of use*, dan *behavioral intentions*. Pengertian *usefulness* (kegunaan) didefinisikan sebagai sejauh mana individu percaya bahwa menggunakan sistem akan meningkatkan kinerja pekerjaannya. Sedangkan *ease of use* (memudahkan penggunaan) didefinisikan sebagai sejauh mana seseorang percaya menggunakan sistem akan bebas dari upaya dan usaha. Dan *behavioral intentions* adalah fungsi dari manfaat dan kemudahan penggunaan yang dirasakan.

2.14.3 Skala Likert

Menurut (Bertram, 2007) skala Likert merupakan teknik penskalaan *non-comparative* dan bersifat *unidimensional* (hanya mengukur satu sifat). Responden diminta untuk menunjukkan tingkat persetujuan mereka dengan pernyataan yang diberikan dengan skala ordinal. Berikut penilaian skala Likert menurut (Boone, 2012) :

1. Sangat Tidak Setuju (STS)
2. Tidak Setuju (TS)
3. Netral (N)
4. Setuju (S)
5. Sangat Setuju (SS)

Berikut perhitungan nilai jumlah pernyataan (n) yang dikalikan dengan bobot nilai skala likert :

- | | |
|--------------------------------|---------|
| 1. Jawaban Sangat setuju | = n x 5 |
| 2. Jawaban Setuju | = n x 4 |
| 3. Jawaban Netral | = n x 3 |
| 4. Jawaban Tidak setuju | = n x 2 |
| 5. Jawaban Sangat Tidak Setuju | = n x 1 |

$$\text{Total nilai} = (nx5)+(nx4)+(nx3)+(nx2)+(nx1) \quad (2.1)$$

Tabel 2.5 Persentase Penilaian

Nilai	Pernyataan
80 sampai 100%	Sangat Setuju
60 sampai 89,99%	Setuju
40 sampai 59,99%	Netral
20 sampai 39,99%	Kurang
0 sampai 19,99%	Sangat Kurang

Sumber : Fabiyanto (2019)

Berikut ini merupakan perhitungan nilai Y untuk mengetahui jumlah dari total perkalian jumlah skala, jumlah responden, dan jumlah pernyataan.

$$Y = N1 \times n \times U \quad (2.2)$$

Dan setelah itu melakukan perhitungan hasil akhir untuk mengetahui besar persentase dengan rumus sebagai berikut :

$$\text{Hasil akhir} = (\text{Total Nilai}/Y) \times 100\% \quad (2.3)$$

BAB 3 METODOLOGI

Pada bab 3 ini merupakan langkah – langkah metode penelitian yang terdiri dari metode pengumpulan data, alur tahapan penelitian, waktu dan lokasi penelitian, metode pengembangan menggunakan *Rapid Application Development* yang terdiri dari perencanaan syarat-syarat, workshop desain RAD dan juga implementasi yang terdiri dari implementasi sistem dan pengujian sistem. Berikut dijelaskan metode penelitian dan alur tahapan penelitian dapat dilihat pada gambar 3.1 :

3.1 Metode Pengumpulan Data

Pada metode pengumpulan data yang dilakukan peneliti untuk mendapatkan data-data yang dibutuhkan sebelum melakukan pengembangan sistem informasi perpustakaan yang ada di SMKN 11 Malang yaitu terdiri dari studi literatur, observasi, dan juga wawancara. Berikut ini penjelasan metode yang digunakan peneliti dalam pengumpulan data yaitu :

a. Studi Literatur

Studi Literatur dilakukan dengan mempelajari buku-buku yang berhubungan dengan penelitian ini dan mengunjungi situs-situs di internet yang mendukung penelitian. Pada tahap studi literatur ini dilakukan pengumpulan semua bahan referensi yang digunakan sebagai landasan dalam menyusun penelitian yang meliputi perpustakaan, sistem informasi berbasis web, metode *Rapid Application Development* (RAD), *Unified Modeling Language* (UML) *black box testing* dan *User Acceptance Testing* (UAT).

b. Observasi

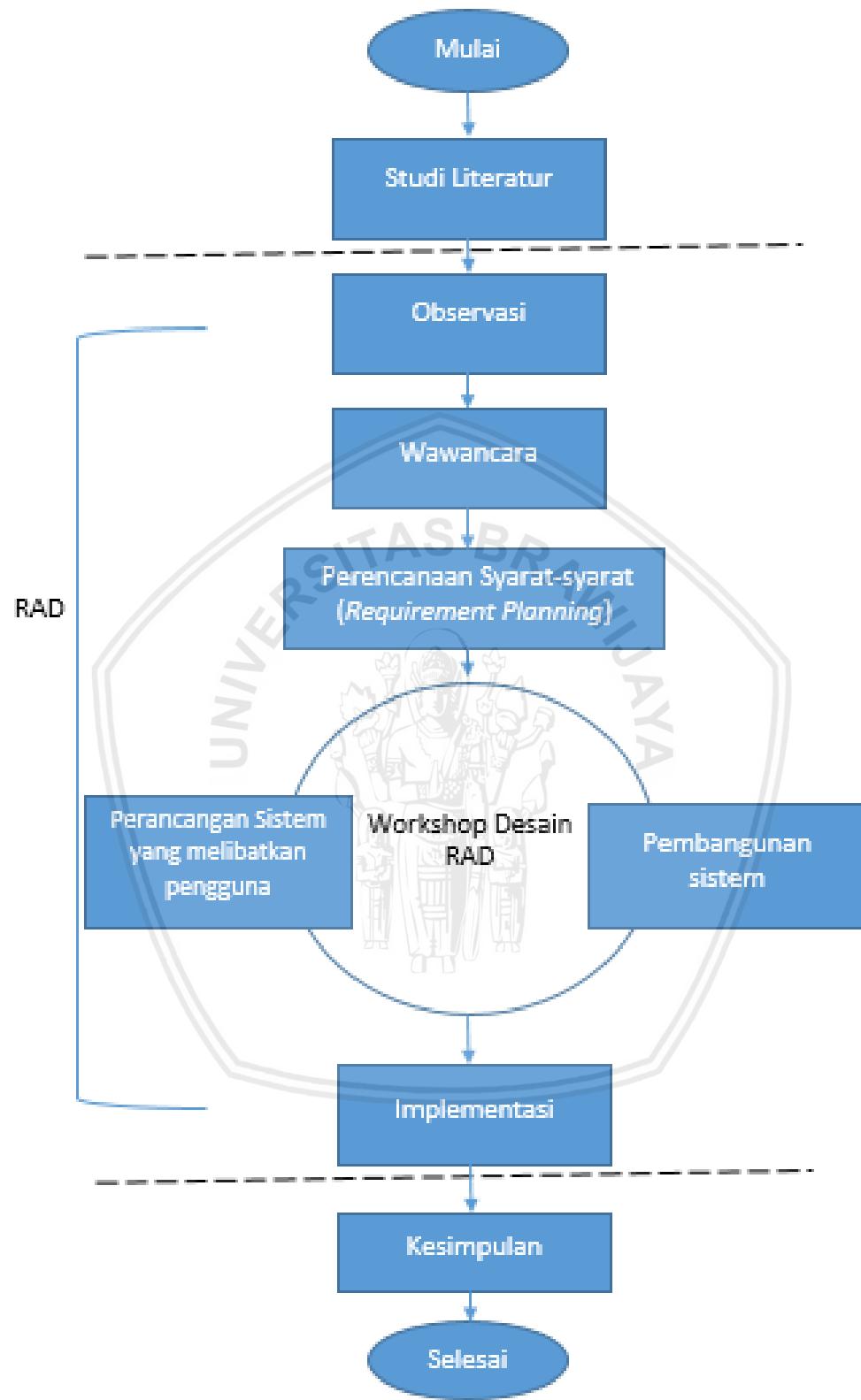
Observasi menurut (Nazir, 1988) yaitu “Pengamatan yang terdiri atas kegiatan pemasukan perhatian terhadap suatu objek dengan menggunakan semua panca indera”. Pada penelitian ini, peneliti melakukan pengamatan langsung di tempat peneliti melakukan Praktik Pengalaman Lapangan (PPL) yaitu pada perpustakaan yang terdapat di SMK Negeri 11 Malang.

c. Wawancara

Pengertian wawancara menurut (Nazir, 1988) menjelaskan wawancara yaitu suatu teknik dalam mengumpulkan data dengan cara lisan terhadap responden yang dapat memberikan suatu informasi yang relevan dengan masalah yang sedang diteliti. Pada penelitian ini dilakukan wawancara kepada 2 orang yang terdiri dari ketua perpustakaan, petugas perpustakaan di SMK Negeri 11 Malang.

3.2 Tahapan Penelitian

Dalam melakukan penelitian ini terdapat serangkaian tahapan-tahapan yang dilakukan peneliti meliputi metode pengumpulan data, metode perancangan sistem, hingga pengambilan kesimpulan. Berikut tahapan penelitian dapat dilihat pada gambar 3.1 :



Gambar 3.1 Tahapan Penelitian

3.3 Waktu dan Lokasi Penelitian

Penelitian dilakukan di SMK Negeri 11 Malang yang berlokasi di Jl. Pelabuhan Bakahuni No.1, Bakalankrajan, Sukun, Kota Malang, Jawa Timur 65148 yang merupakan tempat dimana peneliti melakukan Praktik Pengalaman Lapangan (PPL). Penelitian dilakukan dari bulan Januari 2019 sampai bulan Mei 2019.

3.4 Metode Pengembangan

Metode pengembangan sistem dalam penelitian ini yaitu menggunakan *Rapid Application Development* (RAD). Terdapat fase-fase RAD menurut (Kendall, 2011) yang terdiri dari perencanaan syarat-syarat (*requirement planning*), *RAD Design Workshop*, dan juga Implementasi. Berikut tahapan-tahapan metode RAD pada penelitian ini :

3.4.1 Perencanaan Syarat-Syarat (*Requirement Planning*)

Pada perencanaan syarat-syarat terdiri dari observasi, wawancara, dan analisis kebutuhan. Berikut tahapan perencanaan syarat-syarat untuk kebutuhan sistem informasi perpustakaan SMKN 11 Malang:

a. Observasi

Peneliti melakukan observasi langsung dengan mendatangi SMK Negeri 11 Malang yang berlokasi di Jl. Pelabuhan Bakahuni No.1, Bakalankrajan, Sukun, Kota Malang, Jawa Timur 65148 untuk melakukan pengamatan langsung terhadap bidang yang diteliti.

b. Wawancara

Pada tahap ini dilakukan tanya jawab yang dilakukan pada tanggal 15 Februari 2019 kepada ketua perpustakaan yang bernama Ibu Rahmah Widarti di SMK Negeri 11 Malang dengan mengajukan pertanyaan - pertanyaan yang bertujuan untuk memperoleh data – data yang dibutuhkan.

c. Analisis Kebutuhan

Pada tahap ini dilakukan penggalian kebutuhan apa saja yang dibutuhkan pengguna terhadap sistem informasi perpustakaan berbasis web yang akan dibangun. Analisis meliputi tujuan sistem, pengguna sistem, dan juga lingkungan digunakannya sistem.

3.4.2 Workshop Desain RAD

Pada tahap ini dilakukan pembangunan sistem dan analis bekerja dengan pengguna dalam melakukan proses desain menggunakan bantuan aplikasi *visual paradigm* dan melakukan perbaikan – perbaikan apabila masih terdapat ketidaksesuaian desain antara user dan analis. *Output* dari tahapan ini adalah spesifikasi *software* yang terdiri atas organisasi sistem secara umum, struktur data

dan lain sebagainya. Langkah-langkah yang dilakukan yaitu meliputi iterasi desain UML :

1. Membuat *Usecase Diagram*

Peneliti membuat pemodelan untuk perilaku sistem informasi yang akan dibuat sesuai dengan kebutuhan sistem.

2. Membuat *Sequence Diagram*

Menjelaskan perilaku objek pada *use case* dengan cara mendefinisikan waktu hidup suatu objek dan pesan yang dikirim dan diterima antar objek.

3. Membuat *Activity Diagram*

Membuat sebuah alur kerja aktivitas-aktivitas sistem informasi perpustakaan yang akan dibuat.

4. Membuat *Class Diagram*

Menghubungan antar kelas satu dengan yang lain dan menjelaskan detail setiap kelas di dalam model desain dari suatu sistem, juga memperlihatkan aturan-aturan serta tanggung jawab entitas yang menentukan perilaku sistem.

5. Merancang database

Membuat database sesuai kebutuhan yang telah ditentukan menggunakan phpMyAdmin.

6. Merancang antarmuka (*interface*)

Membuat desain antarmuka pengguna sesuai dengan kebutuhan pengguna menggunakan.

3.5 Implementasi

Pada tahap implementasi dilakukan setelah desain sistem dan rancangan *database* sebelumnya disetujui. Implementasi terdiri dari sebagai berikut:

3.5.1 Implementasi sistem

Pada tahap ini dilakukan implementasi database menggunakan phpMyAdmin, implementasi halaman admin, implementasi halaman anggota perpustakaan. Implementasi kode program dilakukan dengan menggunakan *Visual Studio Code* menggunakan bahasa pemrograman php yang kemudian dihubungkan dengan XAMPP untuk menampilkan hasil implementasi kode program.

3.5.2 Pengujian Sistem

Pengujian menggunakan teknik pengujian *black box*. Pengujian *black box* menurut (Pressman, 2001) menjelaskan bahwa pengujian *black-box* juga disebut pengujian perilaku yang berfokus pada persyaratan fungsional perangkat lunak. Selain itu pengujian juga menggunakan *User Acceptance Testing* (UAT), dimana pengujian dilakukan kepada lima penguji dengan memberikan kuisioner yang terdiri dari aspek *usefulness*, *ease of use*, dan *behavioral intention* yang bertujuan

untuk mengetahui hasil dari sistem yang sudah dibangun apakah sesuai dengan yang diharapkan oleh pengguna.



BAB 4 HASIL DAN PEMBAHASAN

Pada bab 4 ini merupakan langkah – langkah pembangunan sistem informasi perpustakaan yang terdiri dari perencanaan syarat – syarat, workshop desain RAD, dan juga implementasi. Yang pertama, dilakukan perencanaan syarat – syarat. Perencanaan syarat – syarat merupakan rangkaian proses yang dilakukan untuk mengidentifikasi tujuan sistem dan juga untuk mengidentifikasi persyaratan informasi dari tujuan tersebut yang meliputi observasi, wawancara, kemudian dilakukan analisis kebutuhan sistem dari lingkungan sistem, dan juga pengguna sistem. Yang kedua, dilakukan proses workshop desain RAD. Workshop desain RAD adalah rangkaian proses yang terdiri dari proses desain/perancangan dan juga perbaikan yang dilakukan dalam iterasi proses desain RAD hingga disetujui oleh pengguna. Yang ketiga, dilakukan proses implementasi. Implementasi merupakan proses yang dilakukan setelah aspek-aspek pada workshop desain RAD disetujui, dibangun, dan disempurnakan, sistem diuji kemudian sistem baru diperkenalkan ke organisasi.

4.1 Perencanaan Syarat - Syarat (*Requirement Planning*)

Berdasarkan hasil wawancara yang terlampir dalam lampiran A.1 yang dilakukan peneliti dengan petugas perpustakaan setelah dianalisis didapat kebutuhan pengguna. Berikut ini daftar kebutuhan pengguna sistem informasi perpustakaan SMKN 11 Malang :

Tabel 4.1 Kebutuhan Pengguna

No.	Nama Pengguna	Deskripsi
1	Siswa	Pengguna yang akan meminjam buku diperpustakaan
2	Guru/Staff	Pengguna yang akan meminjam buku diperpustakaan
3	Admin	Petugas yang mengontrol sistem dan mengelola data-data di perpustakaan.

Setelah diketahui kebutuhan pengguna, selanjutnya di analisis kebutuhan sistemnya. Hal ini dilakukan agar dapat mengetahui kebutuhan apa saja yang sesuai dengan kebutuhan pengguna sistem informasi perpustakaan SMKN 11 Malang. Berikut merupakan kebutuhan minimal sistem yang harus ada sesuai hasil wawancara yang terlampir pada lampiran A.1:

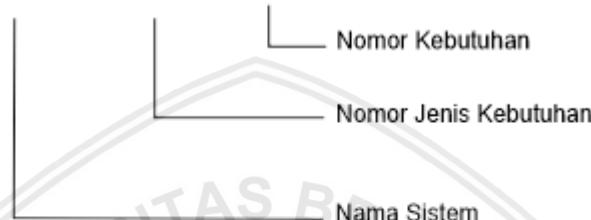
- a. Sistem dapat mengelola data transaksi peminjaman/pengembalian buku
- b. Sistem dapat mengelola data buku
- c. Sistem dapat mengelola data anggota

4.1.1 Kebutuhan Fungsional

Kebutuhan fungsional yaitu kebutuhan yang berisi suatu proses yang dilakukan oleh sistem berdasarkan hasil analisis kebutuhan sistem sebelumnya. Berikut merupakan gambar aturan penomoran kebutuhan fungsional dapat dilihat pada gambar 4.1 :

Kode Kebutuhan : SIPS-1-01

SIPS-1-01



Gambar 4.1 Aturan Penomoran Kebutuhan

Tabel 4.2 Kebutuhan Fungsional

No.	Kebutuhan Fungsional	Pernyataan
SIPS-1-01	Login	Fungsi login berfungsi dengan benar
SIPS-1-02	Logout	Fungsi logout berfungsi dengan benar
SIPS-1-03	Register	Fungsi register berfungsi dengan benar
SIPS-1-04	Pengelolaan data sekolah	Fungsi untuk ubah, hapus serta menampilkan data sekolah berfungsi dengan baik
SIPS-1-05	Pengelolaan data jurusan	Fungsi untuk menambah, ubah, hapus serta data jurusan berfungsi dengan baik
SIPS-1-06	Pengelolaan data kelas	Fungsi untuk menambah, hapus, ubah dan menampilkan data kelas berfungsi dengan baik

Tabel 4.2 Kebutuhan Fungsional (lanjutan)

No.	Kebutuhan Fungsional	Pernyataan
SIPS-1-07	Pencarian kelas	Fungsi pencarian kelas berdasarkan nama kelas yang dimasukkan berfungsi dengan benar
SIPS-1-08	Pengelolaan data siswa	Fungsi untuk menambah, hapus, ubah dan menampilkan data siswa berfungsi dengan baik
SIPS-1-09	Pencarian siswa	Fungsi pencarian siswa berdasarkan nama siswa berfungsi dengan benar
SIPS-1-10	Pengelolaan data staff	Fungsi untuk menambah, hapus, ubah dan menampilkan data staff berfungsi dengan baik
SIPS-1-11	Pencarian staff	Fungsi pencarian staff berdasarkan nama staff berfungsi dengan benar
SIPS-1-12	Pengelolaan anggota	Fungsi untuk menambah, hapus, ubah dan menampilkan data anggota berfungsi dengan baik
SIPS-1-13	<i>Export</i> data anggota	Fungsi <i>export</i> data anggota berfungsi dengan baik
SIPS-1-14	Pengelolaan transaksi	Fungsi untuk menambah, mengubah, mengembalikan buku dan menampilkan data transaksi berfungsi dengan baik
SIPS-1-15	Pencarian transaksi	Fungsi pencarian transaksi berdasarkan data masukan id berfungsi dengan benar
SIPS-1-16	Pengelolaan data buku	Fungsi untuk menambah, hapus, ubah dan menampilkan data buku berfungsi dengan baik

Tabel 4.2 Kebutuhan Fungsional (lanjutan)

No.	Kebutuhan Fungsional	Pernyataan
SIPS-1-17	Pencarian buku	Fungsi pencarian buku berdasarkan data masukan jurusan berfungsi dengan benar
SIPS-1-18	Export data buku	Fungsi export data buku berfungsi dengan baik
SIPS-1-19	Pengelolaan data denda	Fungsi untuk menambah, ubah, hapus dan menampilkan data denda berfungsi dengan baik

4.2 Proses Desain (*Workshop Desain RAD*)

Pada tahap ini analis bekerja dengan pengguna dalam melakukan serangkaian pemodelan sesuai dengan kebutuhan pengguna yang telah diidentifikasi sebelumnya yang meliputi desain *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram*. Berikut ini tahapan-tahapan iterasi rancangan desain sistem informasi perpustakaan :

4.2.1 Proses iterasi ke-1

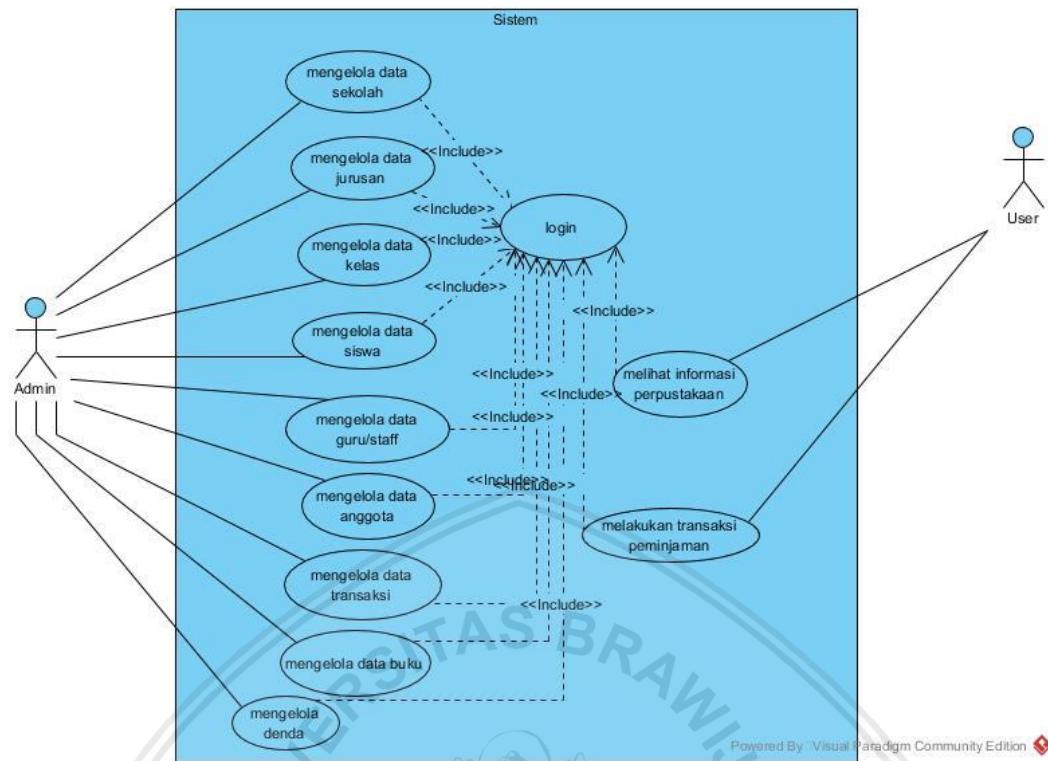
Pada proses iterasi ke-1 use case admin dapat melakukan pengelolaan data sekolah, data jurusan, data kelas data siswa, data guru/staff, data anggota, data transaksi, data buku, dan juga data denda. Kemudian pada use case user yang terdiri dari user siswa, guru, dan staff dapat mendaftar anggota, melihat informasi perpustakaan, dan melakukan transaksi peminjaman buku. Berikut dapat dilihat use case sistem pada gambar 4.2 :

4.2.1.1. *Use Case Diagram* Iterasi-1

Use case diagram menggambarkan perilaku sistem informasi perpustakaan SMKN 11 Malang.

1. Use Case Diagram Sistem iterasi-1

Berikut desain *use case diagram* sistem informasi perpustakaan dapat dilihat pada gambar 4.2 :



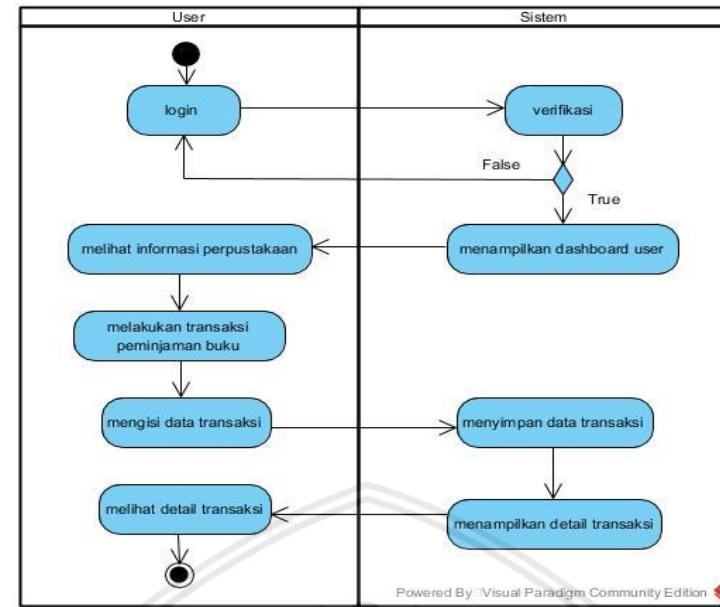
Gambar 4.2 Use Case Diagram Sistem iterasi-1

4.2.1.2. Activity Diagram Iterasi-1

Activity diagram menggambarkan aktivitas sistem informasi perpustakaan SMKN 11 Malang.

1. Activity Diagram Peminjaman Buku (Siswa) iterasi-1

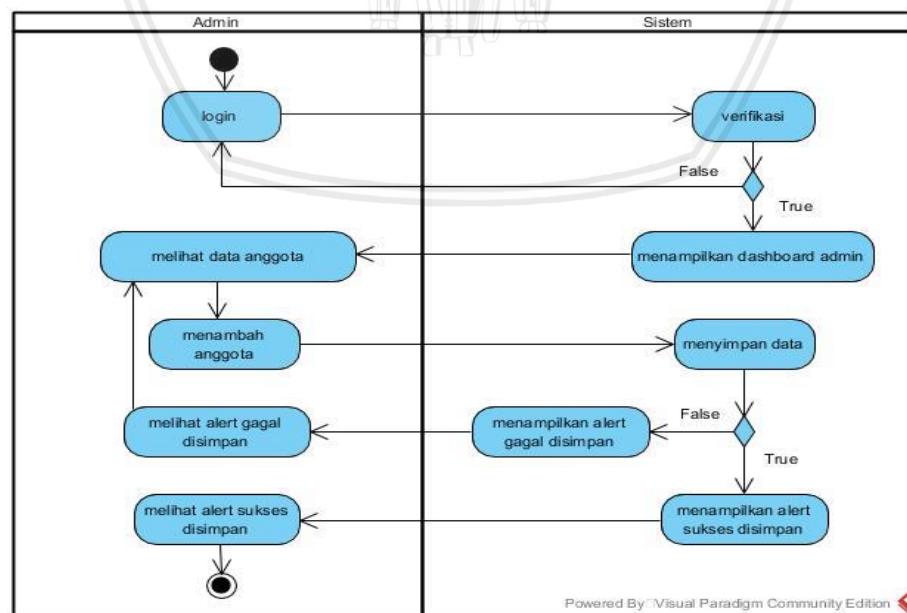
Berikut desain *activity diagram* peminjaman buku dengan user siswa dapat dilihat pada gambar 4.3. Aktifitas peminjaman buku dengan user siswa terdiri dari login yang menginputkan username dan juga password, setelah itu sistem akan memverifikasi data pada database, jika data ditemukan maka akan menampilkan halaman dashboard sistem. Tapi, jika tidak ditemukan maka user siswa harus terlebih dahulu mendaftar menjadi anggota pada halaman registrasi. Setelah melakukan registrasi, user siswa dapat login dan siswa dapat melihat informasi perpustakaan. Pada halaman tersebut, siswa dapat melakukan transaksi peminjaman buku dengan mengisi data transaksi peminjaman. Jika data sudah terisi semua, sistem menyimpan inputan di database. Dan siswa dapat melihat hasil inputan transaksi yang ditampilkan sistem.



Gambar 4.3 Activity Diagram peminjaman buku (User) iterasi-1

2. Activity Diagram tambah anggota (admin) iterasi-1

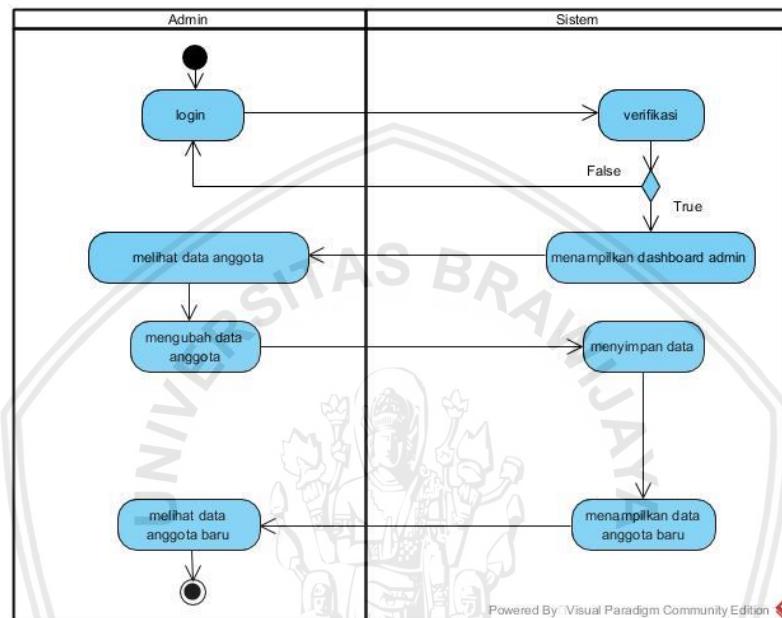
Pada *activity diagram* untuk menambah anggota yang hanya dapat diakses oleh admin, terlebih dahulu admin login setelah verifikasi sistem akan menampilkan halaman dashboard admin. Di halaman data anggota, admin dapat menambahkan anggota dengan menekan tombol tambah yang menampilkan halaman untuk menginputkan data-data anggota. Jika sudah, sistem akan menyimpannya di database. Jika berhasil, admin dapat melihat alert sukses disimpan, sebaliknya jika gagal, admin mendapat pesan alert gagal disimpan dan begitu seterusnya. Berikut dapat dilihat activity diagram tambah anggota dapat dilihat pada gambar 4.4 :



Gambar 4.4 Activity Diagram tambah anggota (Admin) iterasi-1

3. Activity Diagram edit anggota (admin) iterasi-1

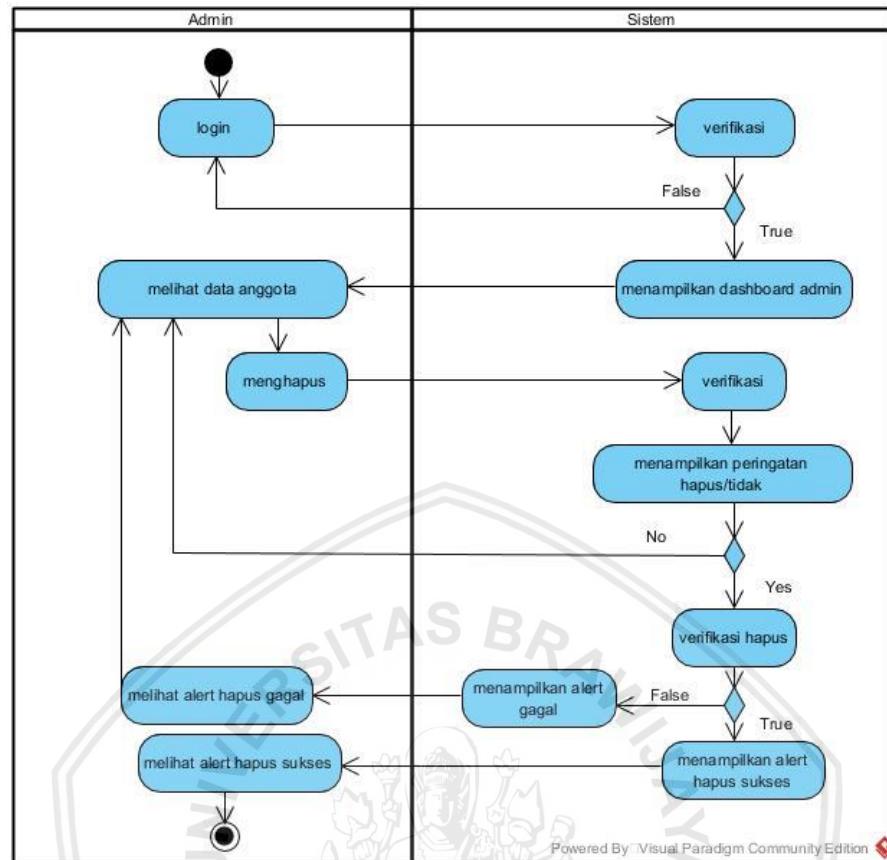
Activity diagram untuk edit anggota hanya dapat diakses oleh admin, terlebih dahulu admin login setelah berhasil login sistem menampilkan dashboard admin. Pada halaman data anggota, admin dapat menekan tombol ubah pada anggota yang dikehendaki, setelah mengubah data anggota, sistem menyimpan data tersebut dan menampilkan data anggota yang sudah diubah. Berikut dapat dilihat pada gambar 4.5 :



Gambar 4.5 Activity Diagram edit anggota (Admin) iterasi-1

4. Activity Diagram hapus anggota (admin) iterasi-1

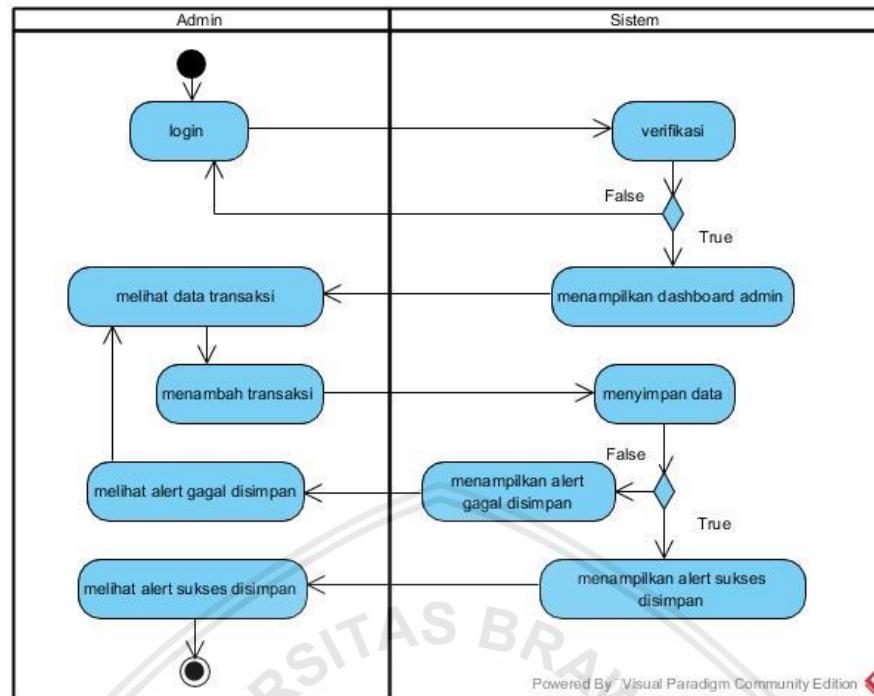
Activity diagram untuk menghapus anggota pada halaman data anggota, admin dapat menekan tombol hapus, setelah itu sistem memberikan tanggapan verifikasi berupa peringatan hapus atau tidak. Jika ya, maka sistem akan verifikasi perintah hapus, jika tidak sistem akan kembali ke halaman data anggota. Dan ketika sistem berhasil dihapus akan menampilkan alert hapus sukses, jika tidak berhasil dihapus akan menampilkan alert gagal hapus dan kembali ke halaman data anggota. Berikut dapat dilihat pada gambar 4.6 :



Gambar 4.6 Activity Diagram hapus anggota (Admin) iterasi-1

5. Activity Diagram tambah transaksi (admin) iterasi-1

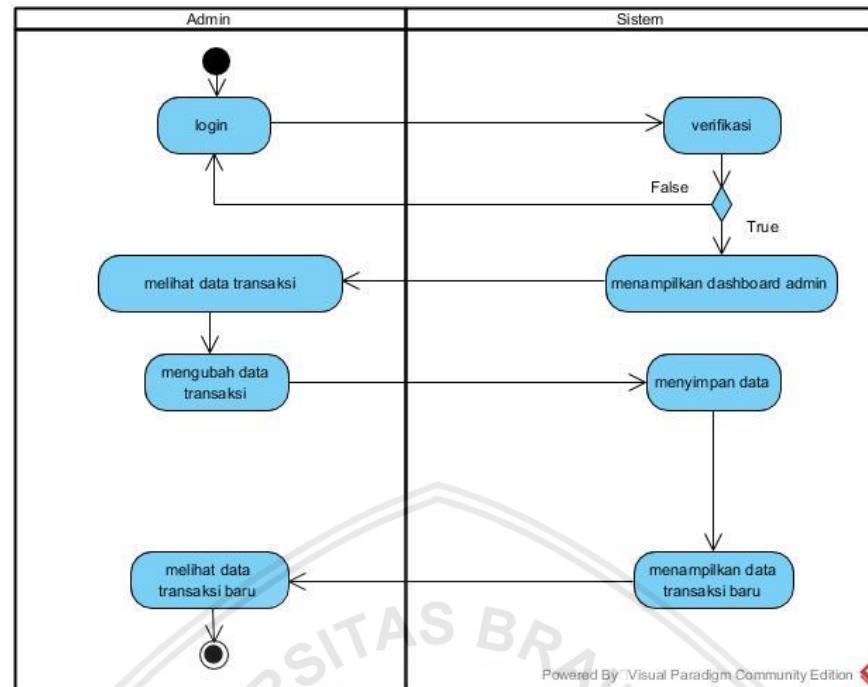
Activity diagram untuk menambah transaksi terlebih dahulu admin login setelah verifikasi sistem akan menampilkan halaman dashboard admin. Di halaman data transaksi, admin dapat menambahkan transaksi dengan menekan tombol tambah yang menampilkan halaman untuk menginputkan transaksi peminjaman. Jika sudah, sistem akan menyimpannya di database. Jika berhasil, admin dapat melihat alert sukses disimpan, sebaliknya jika gagal, admin mendapat pesan alert gagal disimpan dan begitu seterusnya. Berikut dapat dilihat *activity diagram* tambah transaksi dapat dilihat pada gambar 4.7 :



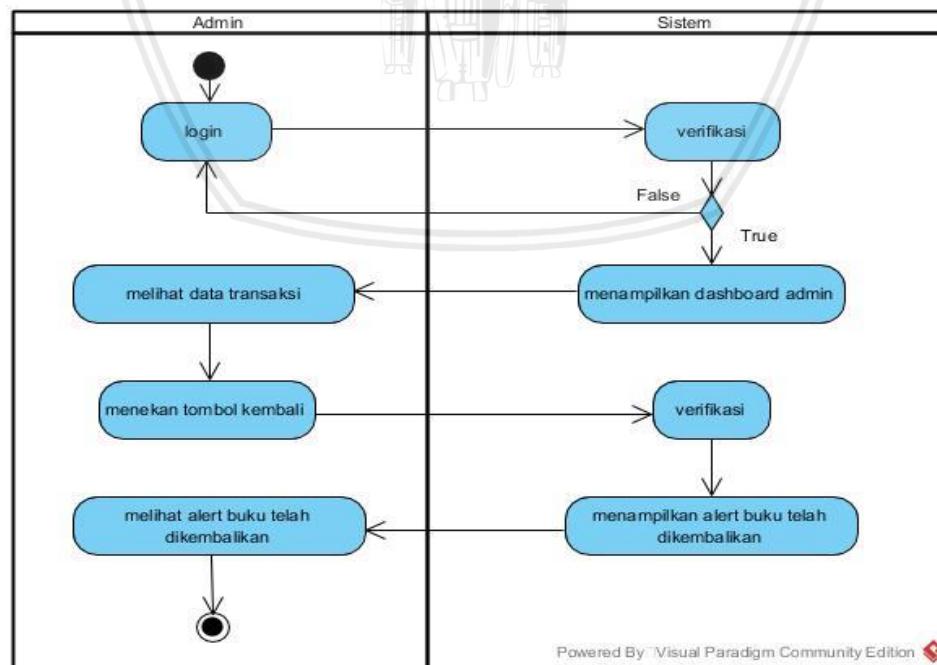
Gambar 4.7 Activity Diagram tambah transaksi (Admin) iterasi-1

6. Activity Diagram edit transaksi (admin) iterasi-1

Activity diagram untuk edit transaksi terlebih dahulu admin login setelah berhasil login sistem menampilkan dashboard admin. Pada halaman data transaksi, admin dapat menekan tombol ubah pada transaksi anggota yang dikehendaki, setelah mengubah data transaksi, sistem menyimpan data tersebut dan menampilkan data transaksi yang sudah diubah. Berikut dapat dilihat pada gambar 4.8 :

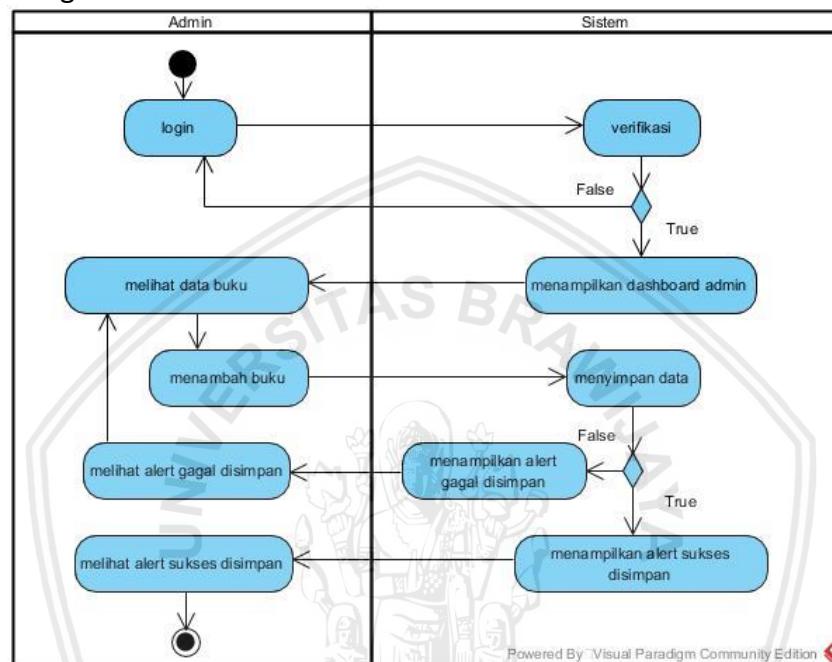
**Gambar 4.8 Activity Diagram edit transaksi (Admin) iterasi-1****7. Activity Diagram mengembalikan buku pada menu transaksi iterasi-1**

Activity diagram untuk mengembalikan buku pada menu transaksi, admin dapat menekan tombol kembali, setelah itu sistem menampilkan alert buku telah dikembalikan dan kembali ke halaman data transaksi. Berikut *activity diagram* mengembalikan buku pada menu transaksi iterasi ke-1 dapat dilihat pada gambar 4.9 :

**Gambar 4.9 Activity Diagram mengembalikan buku (Admin) iterasi-1**

8. Activity Diagram tambah buku (admin) iterasi-1

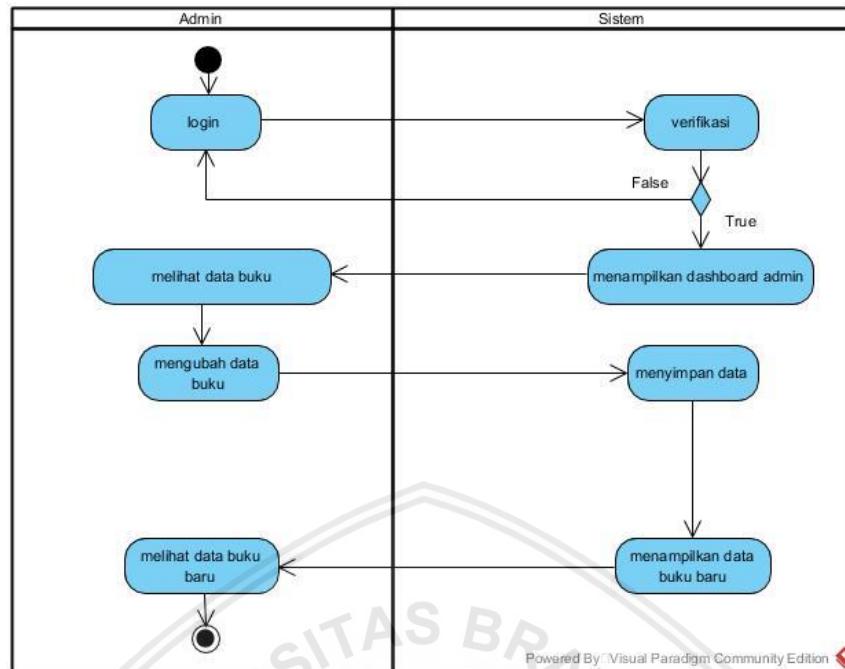
Activity diagram untuk menambah buku, pada halaman data buku, admin dapat menambahkan buku dengan menekan tombol tambah yang menampilkan halaman untuk menginputkan tambah buku. Jika sudah, sistem akan menyimpannya di database. Jika berhasil, admin dapat melihat alert sukses disimpan, sebaliknya jika gagal, admin mendapat pesan alert gagal disimpan dan begitu seterusnya. Berikut dapat dilihat *activity diagram* tambah buku dapat dilihat pada gambar 4.10 :



Gambar 4.10 Activity Diagram tambah buku (Admin) iterasi-1

9. Activity Diagram edit buku (admin) iterasi-1

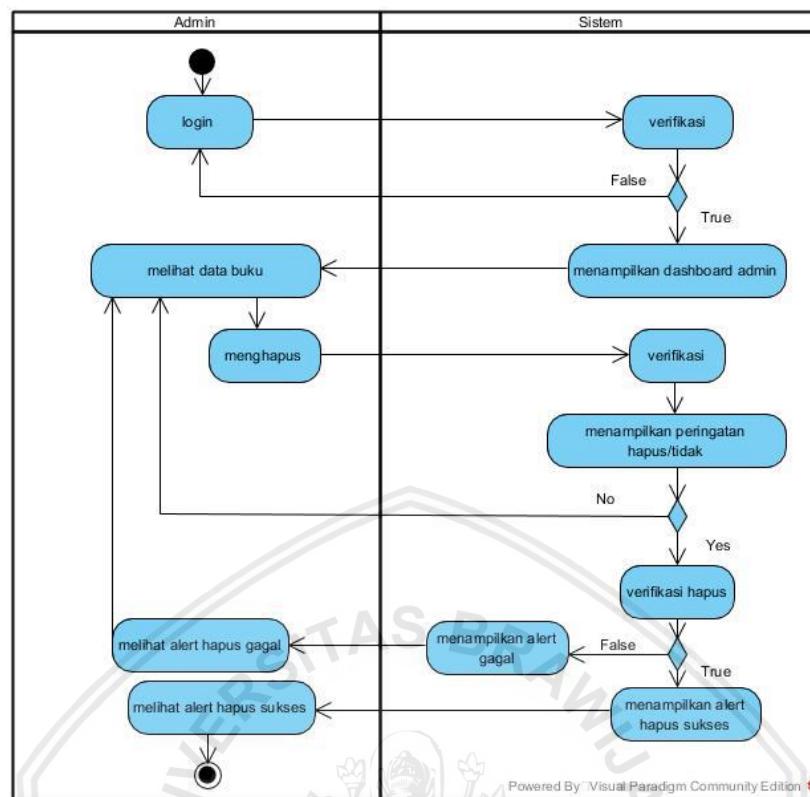
Activity diagram untuk edit buku, pada halaman data buku, admin dapat menekan tombol ubah pada halaman data buku yang dikehendaki, setelah mengubah data buku, sistem menyimpan data tersebut dan menampilkan data buku yang sudah diubah. Berikut dapat dilihat pada gambar 4.11 :



Gambar 4.11 Activity Diagram edit buku (Admin) iterasi-1

10. Activity Diagram hapus buku (admin) iterasi-1

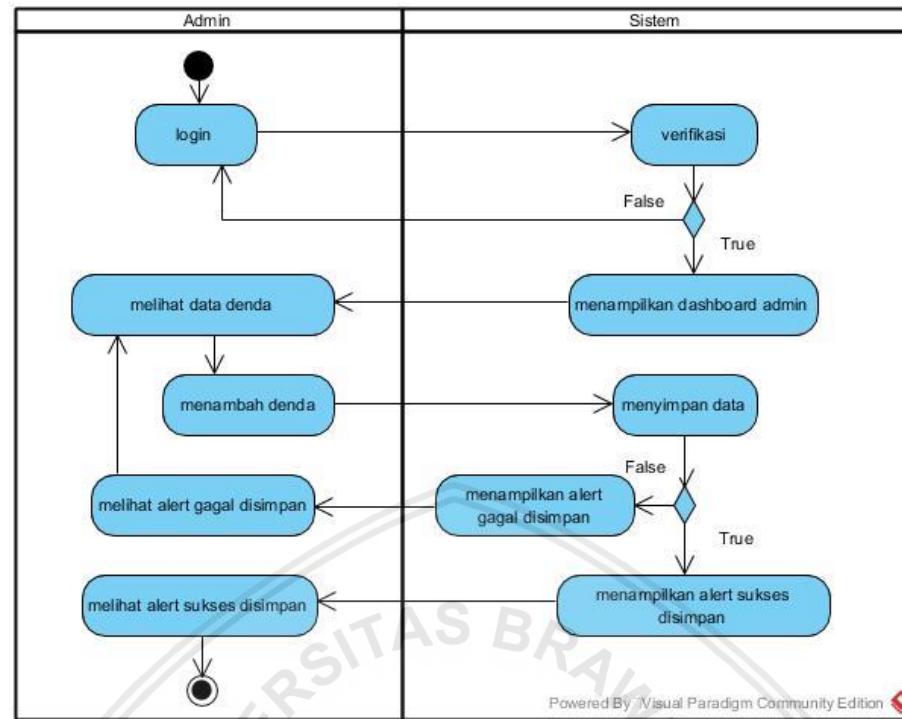
Activity diagram untuk menghapus buku pada halaman data buku, admin dapat menekan tombol hapus, setelah itu sistem memberikan tanggapan verifikasi berupa peringatan hapus atau tidak. Jika ya, maka sistem akan verifikasi perintah hapus, jika tidak sistem akan kembali ke halaman data buku. Dan ketika sistem berhasil dihapus akan menampilkan alert hapus sukses, jika tidak berhasil dihapus akan menampilkan alert gagal hapus dan kembali ke halaman data buku. Berikut dapat dilihat pada gambar 4.12 :



Gambar 4.12 Activity Diagram hapus buku (Admin) iterasi-1

11. Activity Diagram tambah denda (admin) iterasi-1

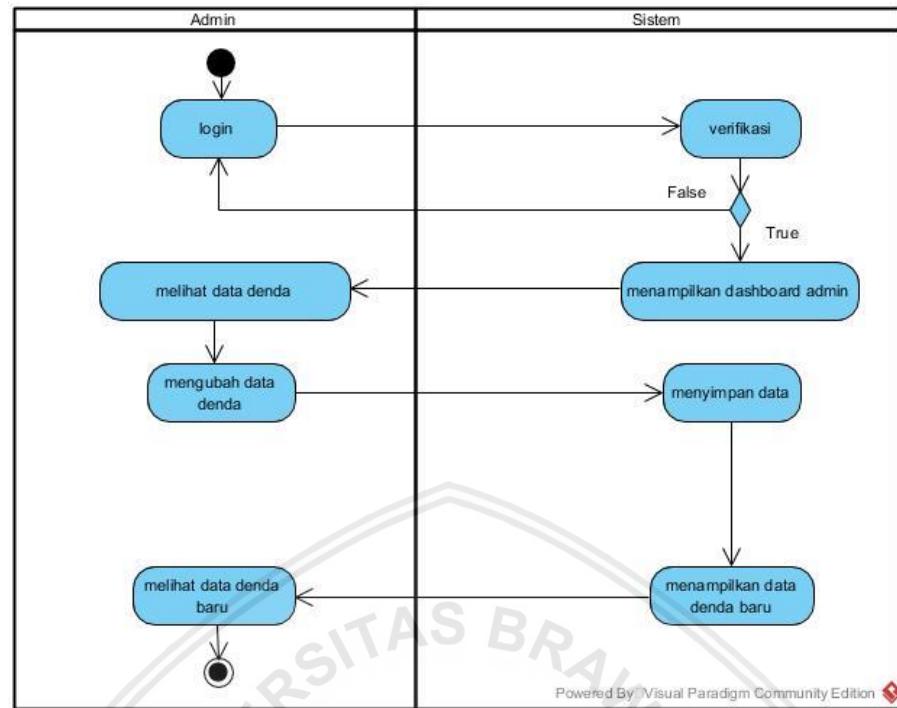
Activity diagram untuk menambah denda, pada halaman data denda, admin dapat menambahkan denda dengan menekan tombol tambah yang menampilkan halaman untuk menginputkan tambah denda. Jika sudah, sistem akan menyimpannya di database. Jika berhasil, admin dapat melihat alert sukses disimpan, sebaliknya jika gagal, admin mendapat pesan alert gagal disimpan dan begitu seterusnya. Berikut dapat dilihat *activity diagram* tambah denda dapat dilihat pada gambar 4.13 :



Gambar 4.13 Activity Diagram tambah denda (Admin) iterasi-1

12. Activity Diagram edit denda (admin) iterasi-1

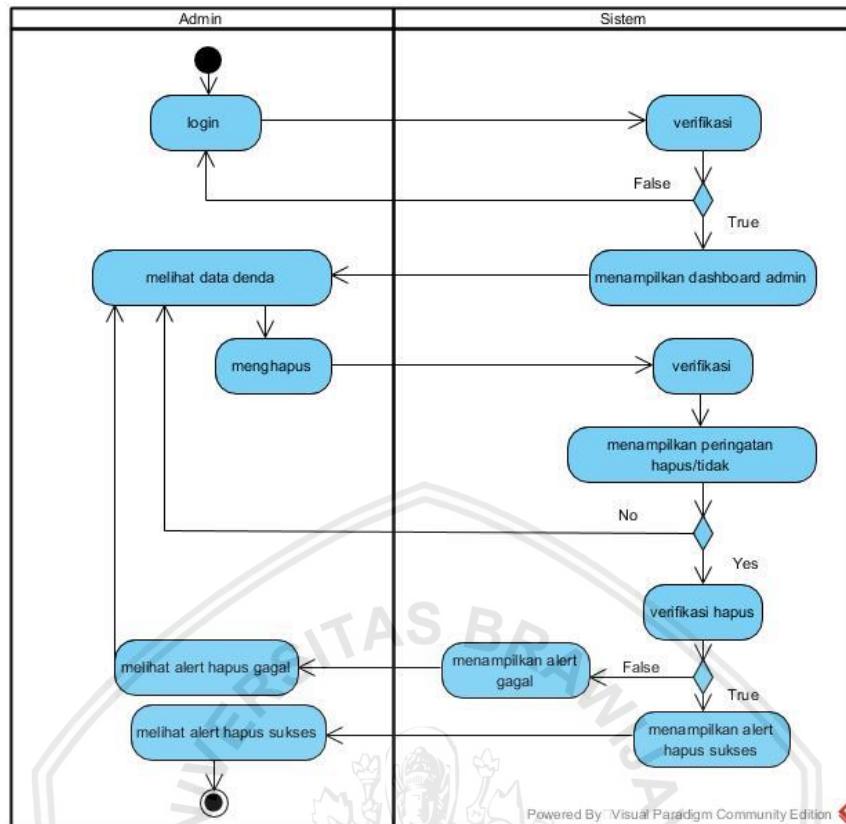
Activity diagram untuk edit denda, pada halaman data denda, admin dapat menekan tombol ubah pada halaman data denda yang dikehendaki, setelah mengubah data denda, sistem menyimpan data tersebut dan menampilkan data denda yang sudah diubah. Berikut dapat dilihat pada gambar 4.14 :



Gambar 4.14 Activity Diagram edit denda (Admin) iterasi-1

13. Activity Diagram hapus denda (admin) iterasi-1

Activity diagram untuk menghapus denda pada halaman data denda, admin dapat menekan tombol hapus, setelah itu sistem memberikan tanggapan verifikasi berupa peringatan hapus atau tidak. Jika ya, maka sistem akan verifikasi perintah hapus, jika tidak sistem akan kembali ke halaman data denda. Dan ketika sistem berhasil dihapus akan menampilkan alert hapus sukses, jika tidak berhasil dihapus akan menampilkan alert gagal hapus dan kembali ke halaman data denda. Berikut dapat dilihat pada gambar 4.15 :



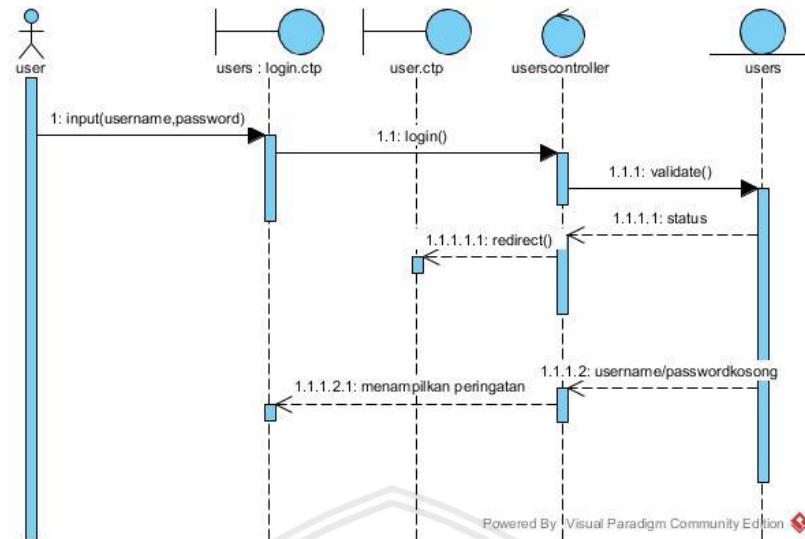
Gambar 4.15 Activity Diagram hapus denda (Admin) iterasi-1

4.2.1.3. Sequence Diagram iterasi-1

Sequence diagram menggambarkan perilaku pada *use case* sistem informasi perpustakaan SMKN 11 Malang berupa pesan yang terkirim dan diterima .

1. Sequence Diagram login (user) iterasi-1

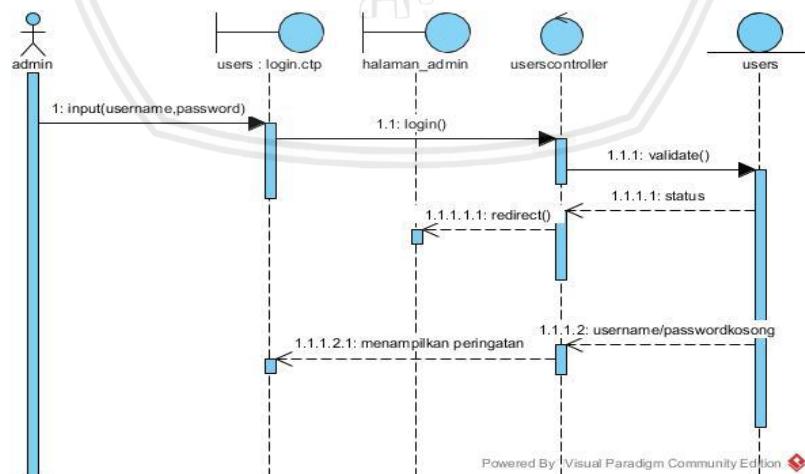
Pada desain *sequence diagram* login user yang terdiri dari siswa, guru, dan staff, terlebih dahulu login dengan menginputkan username dan juga password. Ketika selesai menginputkan username dan password dan menekan tombol login, segera sistem memproses data yang masuk, mencari nama dan password yang diperintahkan di database. Jika berhasil proses pencarian ditemukan, sistem akan menampilkan halaman user. Tapi, jika data yang diinputkan tersebut tidak ditemukan di database, maka akan kembali ke halaman login. Berikut *sequence diagram* login user dapat dilihat pada gambar 4.16 :



Gambar 4.16 Sequence Diagram login (User) iterasi-1

2. Sequence Diagram login (Admin) iterasi-1

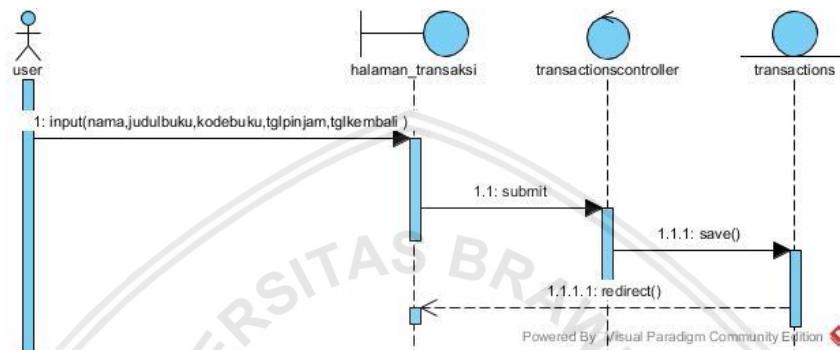
Pada desain *sequence diagram* login admin, terlebih dahulu login dengan menginputkan username dan juga password. Ketika selesai menginputkan username dan password dan menekan tombol login, segera sistem memproses data yang masuk, mencari nama dan password yang diperintahkan di database. Jika berhasil proses pencarian ditemukan, sistem akan menampilkan halaman admin. Tapi, jika data yang diinputkan tersebut tidak ditemukan di database, maka sistem kembali ke halaman login. Berikut *sequence diagram* login admin dapat dilihat pada gambar 4.17 :



Gambar 4.17 Sequence Diagram login (Admin) iterasi-1

3. Sequence Diagram peminjaman (user) iterasi-1

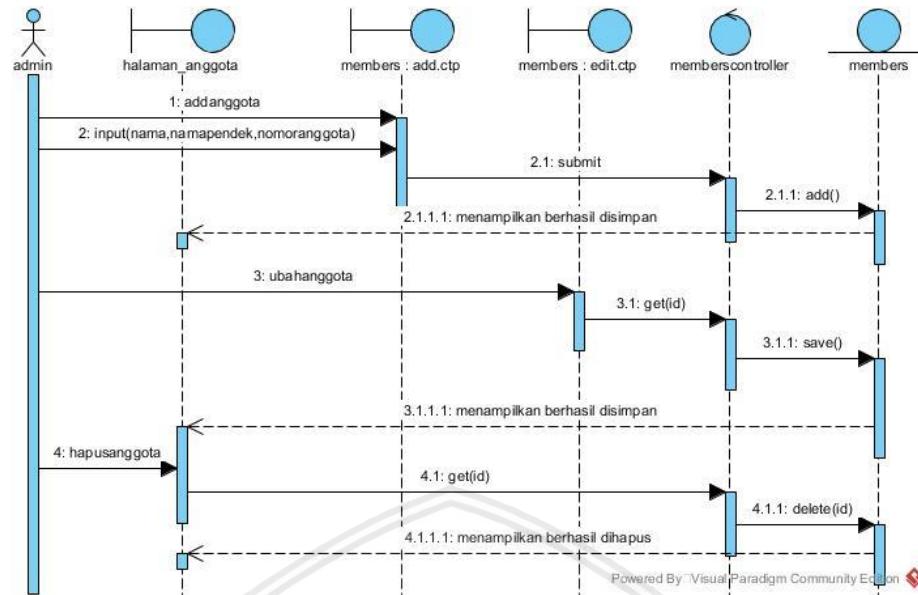
Sequence diagram peminjaman pada user, terlebih dahulu menginputkan nama, judul buku, kode buku, tanggal pinjam, dan juga tanggal kembali. Setelah itu menekan tombol submit yang kemudian diproses untuk disimpan di database dan sistem akan menampilkan data yang telah disimpan. Berikut gambar *sequence diagram* peminjaman dapat dilihat pada gambar 4.18 :



Gambar 4.18 Sequence Diagram peminjaman (User) iterasi-1

4. Sequence Diagram kelola anggota (Admin) iterasi-1

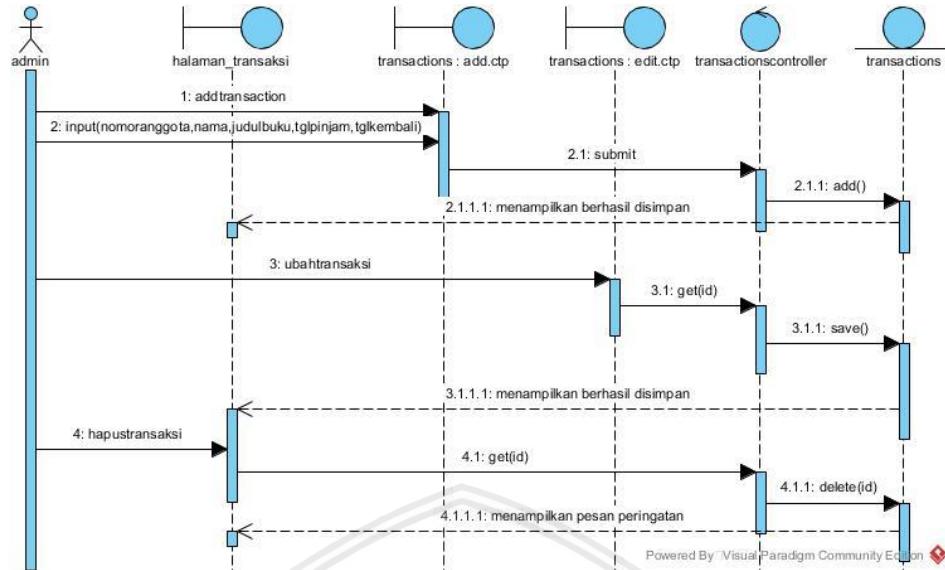
Pada *sequence diagram* kelola anggota, admin dapat melihat data anggota dengan masuk ke halaman anggota dan menekan tombol lihat maka sistem akan memprosesnya dan memanggil data di database kemudian menampilkan ke admin. Admin juga dapat mengubah data anggota dengan menekan tombol ubah yang berisi data yang akan diubah dan menekan tombol submit, maka sistem akan menyimpan data yang diinputkan tersebut. Kemudian jika admin ingin menambah data, admin menekan tombol tambah yang menampilkan halaman tambah yang berisi inputan. Setelah selesai diinputkan, data akan disimpan di database. Dan jika admin ingin menghapus data anggota, maka sistem akan memprosesnya mengambil data di database yang kemudian sistem menjalankan proses penghapusan data tersebut. Berikut dapat dilihat *sequence diagram* kelola anggota pada gambar 4.19 :



Gambar 4.19 Sequence Diagram kelola anggota (Admin) iterasi-1

5. Sequence Diagram kelola transaksi (Admin) iterasi-1

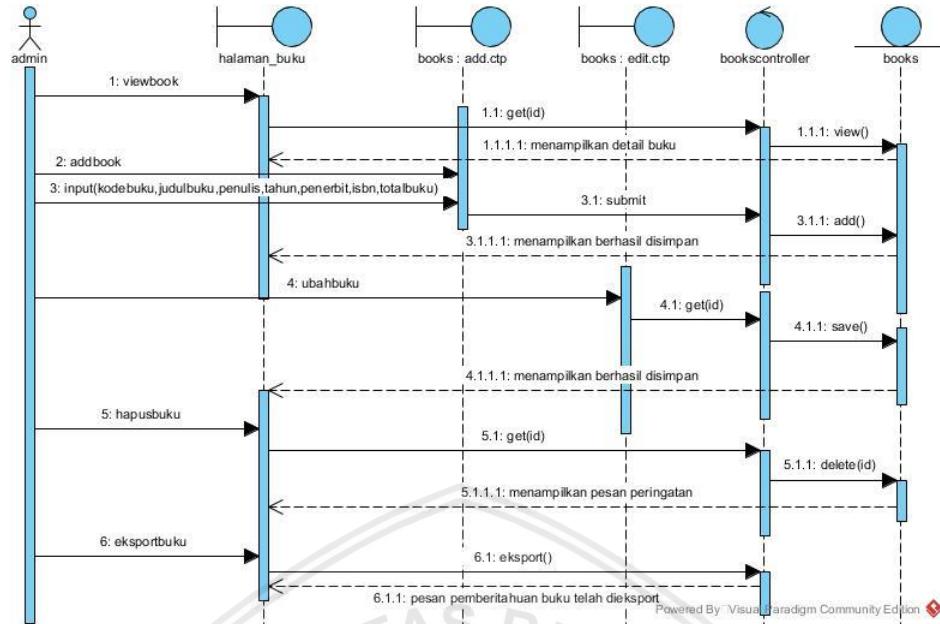
Sequence diagram kelola transaksi, admin dapat melihat data transaksi dengan masuk ke halaman transaksi dan menekan tombol lihat maka sistem akan memprosesnya dan memanggil data di database kemudian menampilkan ke admin. Admin juga dapat mengubah data transaksi dengan menekan tombol ubah yang berisi data yang akan diubah dan menekan tombol submit, maka sistem akan menyimpan data yang diinputkan tersebut. Kemudian jika admin ingin menambah data, admin menekan tombol tambah yang menampilkan halaman tambah yang berisi inputan. Setelah selesai diinputkan, data akan disimpan di database. Dan jika admin ingin mengembalikan data transaksi, maka sistem akan memprosesnya mengambil data di database yang kemudian sistem menjalankan proses pengembalian data tersebut. Berikut desain *sequence diagram* kelola transaksi iterasi-1 sistem informasi perpustakaan dapat dilihat pada gambar 4.20:



Gambar 4.20 Sequence diagram kelola transaksi (Admin) iterasi-1

6. Sequence Diagram kelola buku (Admin) iterasi-1

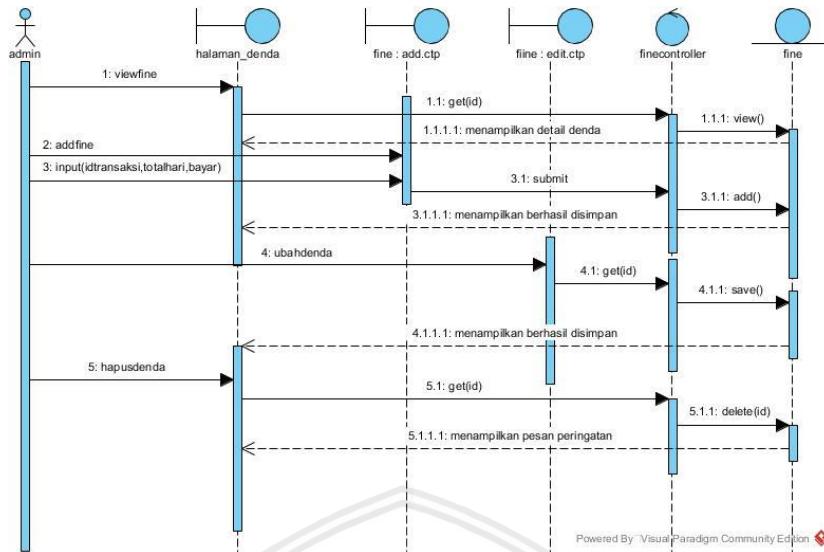
Sequence diagram kelola buku, admin dapat melihat data buku dengan masuk ke halaman buku dan menekan tombol lihat maka sistem akan memprosesnya dan memanggil data di database kemudian menampilkan ke admin. Admin juga dapat mengubah data buku dengan menekan tombol ubah yang berisi data yang akan diubah dan menekan tombol submit, maka sistem akan menyimpan data yang diinputkan tersebut. Kemudian jika admin ingin menambah data, admin menekan tombol tambah yang menampilkan halaman tambah yang berisi inputan. Setelah selesai diinputkan, data akan disimpan di database. Dan jika admin ingin menghapus data buku, maka sistem akan memprosesnya mengambil data di database yang kemudian sistem menjalankan proses penghapusan data tersebut. Berikut dapat dilihat *sequence diagram* kelola buku pada gambar 4.21 :



Gambar 4.21 Sequence Diagram kelola buku (Admin) iterasi-1

7. Sequence Diagram kelola denda (Admin) iterasi-1

Sequence diagram kelola denda, admin dapat melihat data denda dengan masuk ke halaman denda dan menekan tombol lihat maka sistem akan memprosesnya dan memanggil data di database kemudian menampilkan ke admin. Admin juga dapat mengubah data denda dengan menekan tombol ubah yang berisi data yang akan diubah dan menekan tombol submit, maka sistem akan menyimpan data yang diinputkan tersebut. Kemudian jika admin ingin menambah data, admin menekan tombol tambah yang menampilkan halaman tambah yang berisi inputan. Setelah selesai diinputkan, data akan disimpan di database. Dan jika admin ingin menghapus data denda, maka sistem akan memprosesnya mengambil data di database yang kemudian sistem menjalankan proses penghapusan data tersebut. Berikut dapat dilihat *sequence diagram* kelola denda pada gambar 4.22 :



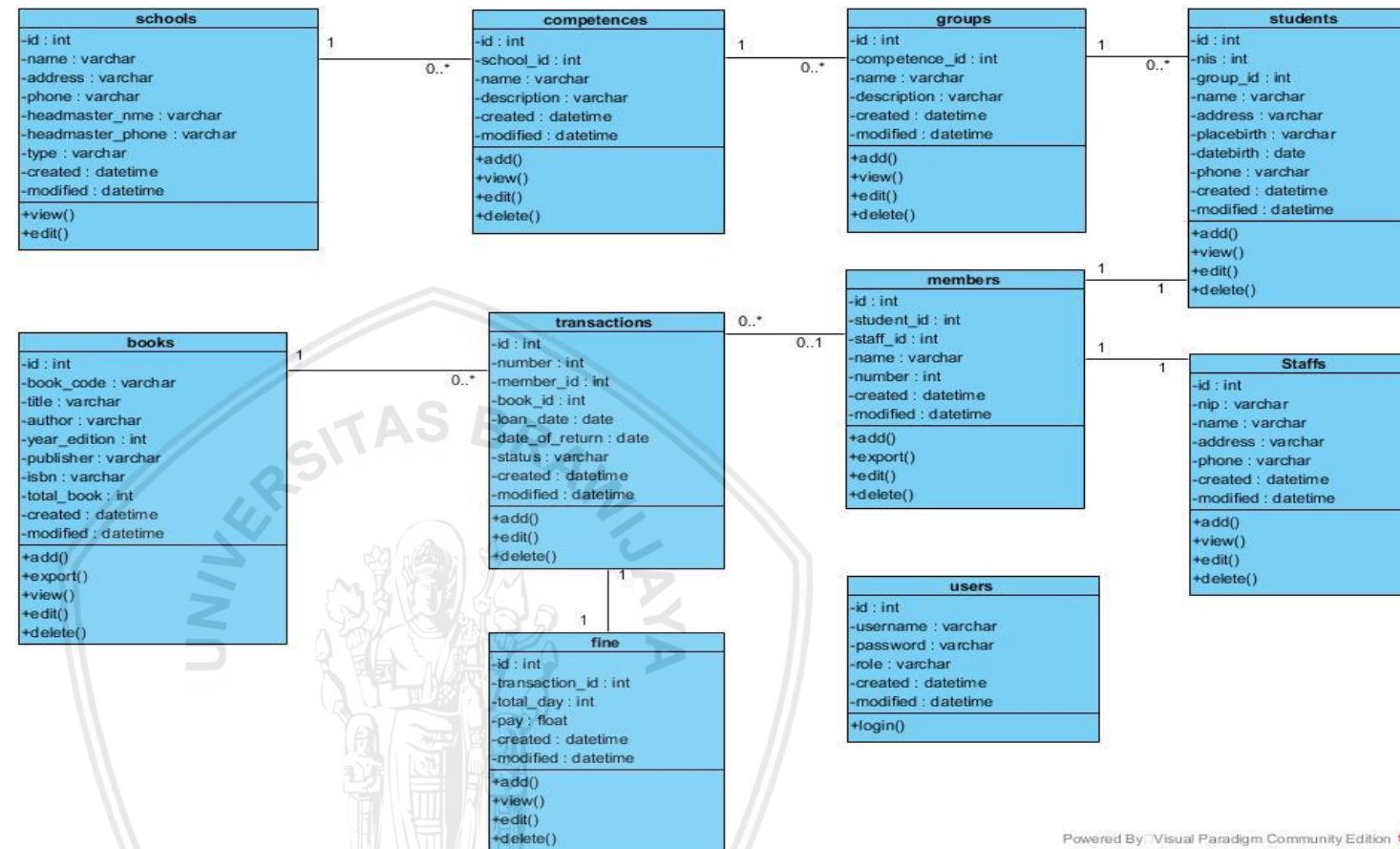
Gambar 4.22 Sequence Diagram kelola denda (Admin) iterasi-1

4.2.1.4. Class Diagram iterasi-1

Class diagram menggambarkan struktur database sistem informasi perpustakaan SMKN 11 Malang. *Class diagram* bertujuan untuk mendefinisikan kelas-kelas yang terdapat pada sistem. Berikut dapat dilihat class diagram iterasi-1 pada gambar 4.23 :

1. Class Diagram Sistem Informasi Perpustakaan iterasi-1

Berikut desain *class diagram* sistem informasi perpustakaan dapat dilihat pada gambar 4.23 :



Gambar 4.23 Class Diagram Sistem Informasi Perpustakaan iterasi-1

4.2.1.5. Perancangan database iterasi-1

Pada tahap ini dilakukan perancangan database sesuai dengan class diagram yang telah dibuat sebelumnya. Berikut merupakan rancangan database:

1. Tabel sekolah iterasi-1

Nama tabel : Schools

Primary key : Id

Foreign key : Id, name

Tabel 4.3 Tabel sekolah iterasi-1

Nama	Tipe
id	int(10)
name	varchar(100)
address	varchar(255)
phone	varchar(20)
headmaster_nme	varchar(50)
headmaster_phone	varchar(20)
type	varchar(255)
created	datetime
modified	datetime

2. Tabel jurusan iterasi-1

Nama tabel : competences

Primary key : Id

Foreign key : Id

Tabel 4.4 Tabel jurusan iterasi-1

Nama	Tipe
id	int(10)
school_id	int(10)
name	varchar(255)
description	varchar(255)
created	datetime
modified	datetime

3. Tabel kelas iterasi-1

Nama tabel : groups

Primary key : Id

Foreign key : Id, name

Tabel 4.5 Tabel kelas iterasi-1

Nama	Tipe
id	int(10)
competence_id	int(10)
name	varchar(255)
description	varchar(255)
created	datetime
modified	datetime

4. Tabel siswa iterasi-1

Nama tabel : students

Primary key : Id

Foreign key : Id, nis

Tabel 4.6 Tabel siswa iterasi-1

Nama	Tipe
id	int(10)
nis	int(10)
group_id	int(10)
name	varchar(50)
address	varchar(255)
placebirth	varchar(25)
datebirth	date
phone	varchar(20)
created	datetime
modified	datetime

5. Tabel guru/staff iterasi-1

Nama tabel : staffs

Primary key : Id

Foreign key : Id, nip

Tabel 4.7 Tabel guru/staff iterasi-1

Nama	Tipe
id	int(10)
nip	varchar(60)
name	varchar(50)
address	varchar(255)
phone	varchar(20)
created	datetime
modified	datetime

6. Tabel user iterasi-1

Nama tabel : users

Primary key : Id

Foreign key : Id, username

Tabel 4.8 Tabel users iterasi-1

Nama	Tipe
id	int(10)
username	varchar(50)
password	varchar(255)
role	varchar(20)
created	datetime
modified	datetime

7. Tabel anggota iterasi-1

Nama tabel : members

Primary key : Id

Foreign key : Id, number

Tabel 4.9 Tabel anggota iterasi-1

Nama	Tipe
id	int(10)
student_id	int(10)
staff_id	int(10)
name	varchar(60)
number	int(10)
created	datetime
modified	datetime

8. Tabel transaksi iterasi-1

Nama tabel : transactions

Primary key : Id

Foreign key : Id

Tabel 4.10 Tabel transaksi iterasi-1

Nama	Tipe
id	int(10)
number	int(10)
member_id	int(10)
book_id	int(10)
loan_date	date
date_of_return	date
status	varchar(60)
created	datetime
modified	datetime

9. Tabel buku iterasi-1

Nama tabel : books

Primary key : Id

Foreign Key : Id, book_code, isbn

Tabel 4.11 Tabel buku iterasi-1

Nama	Tipe
id	int(10)
book_code	varchar(10)
title	varchar(255)
author	varchar(255)
year_edition	int(10)
publisher	varchar(255)
isbn	varchar(20)
total_book	int(10)
created	datetime
modified	datetime

10. Tabel denda iterasi-1

Nama tabel : Fine

Primary key : Id

Foreign key: Id

Tabel 4.12 Tabel denda iterasi-1

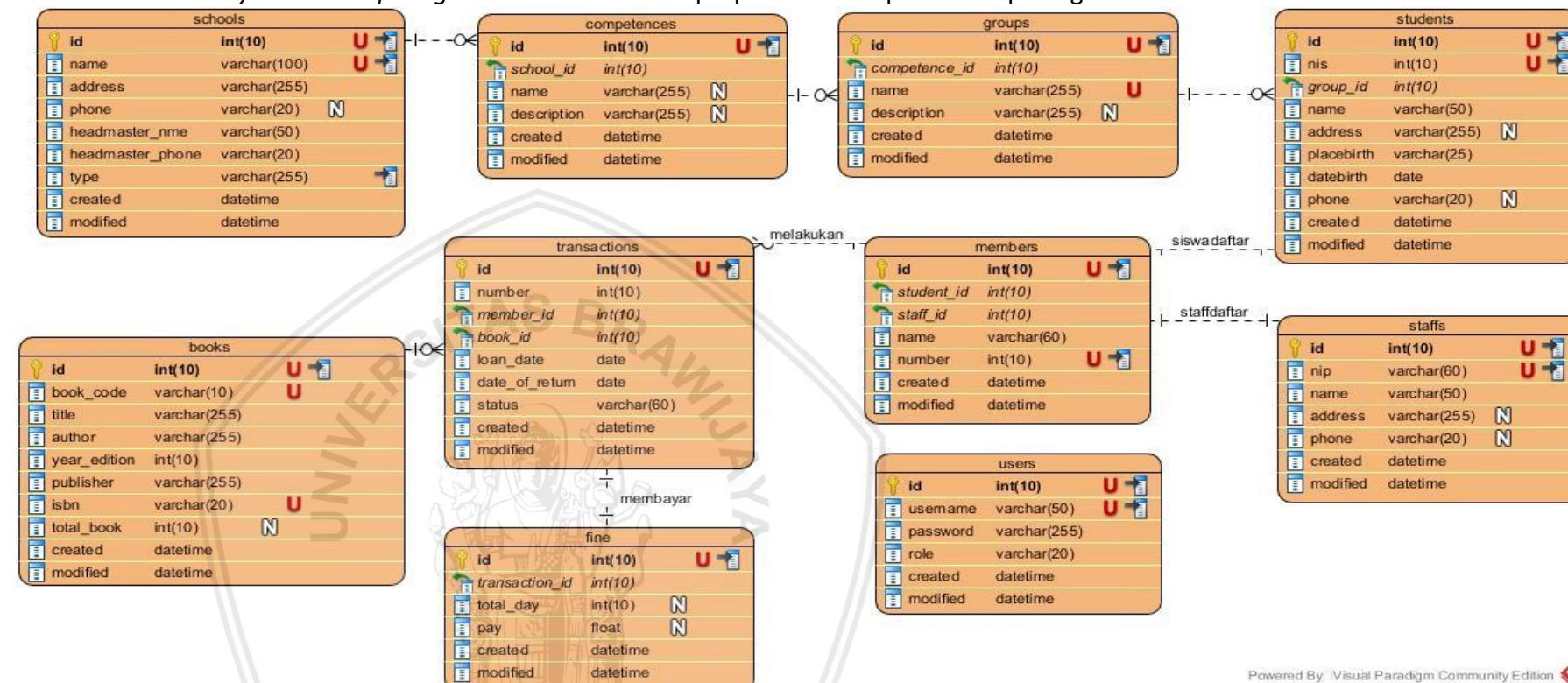
Nama	Tipe
id	int(10)
transaction_id	int(10)
total_day	int(10)
pay	float
created	datetime
modified	datetime

4.2.1.6. Entity Relationship Diagram iterasi-1

Entity Relationship Diagram merupakan gambaran dari hubungan diantara *entity* tersebut.

1. Entity Relationship Diagram Sistem Informasi Perpustakaan iterasi-1

Berikut desain *entity relationship diagram* sistem informasi perpustakaan dapat dilihat pada gambar 4.24 :



Gambar 4.24 Entity Relationship Diagram Sistem Informasi Perpustakaan iterasi-1

4.2.1.7. Rancangan Antarmuka (*Interface*) iterasi-1

Rancangan antarmuka (*interface*) merupakan rancangan mekanisme komunikasi antara pengguna (*user*) dengan sistem. Berikut rancangan antarmuka(*interface*) sistem informasi perpustakaan SMK Negeri 11 Malang.

1. Rancangan *inteface* halaman login iterasi-1

Halaman login terdiri dari username, password, tombol login dan juga tombol untuk registrasi anggota yang mengarahkan ke halaman registrasi. Berikut merupakan rancangan *interface* halaman login iterasi-1 dapat dilihat pada gambar 4.25 :

The image shows a screenshot of a web-based login interface. At the top, there is a header with the text "SIPS SMKN11" and "Malang". Below the header is a form with a title "Masukkan Username dan Password". Inside the form, there are two input fields: one for "Username" and one for "Password". Below these fields is a "Login" button. The background of the page features a watermark of the Universitas Brawijaya logo and text.

Gambar 4.25 Halaman login iterasi-1

2. Rancangan *inteface* halaman register iterasi-1

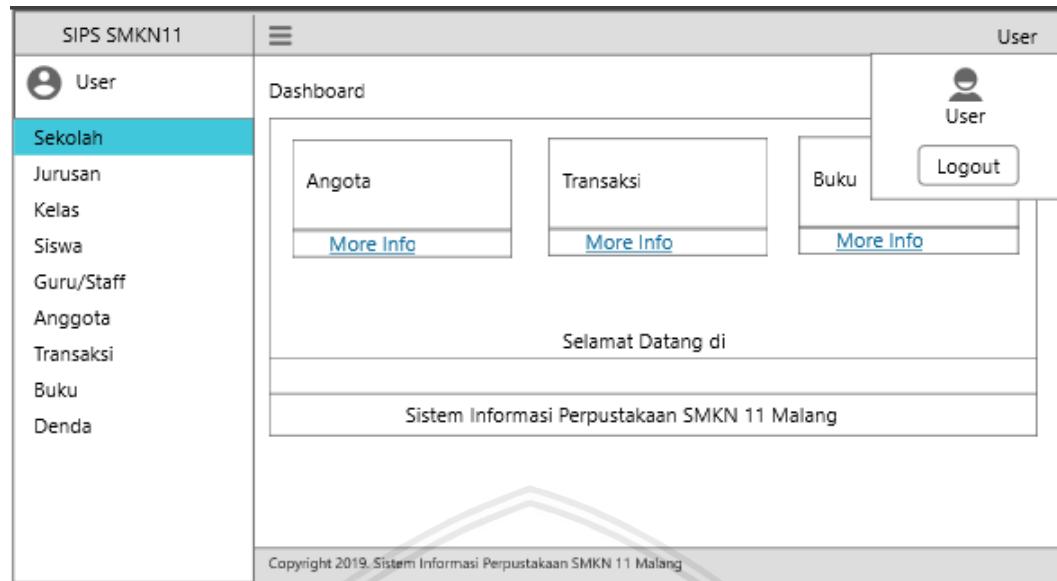
Pada halaman registrasi untuk anggota terdiri dari username, password, *role* untuk memilih jenis user siswa, atau user staff, tombol submit untuk menyimpan data pada database dan tombol *back* untuk kembali ke halaman login. Berikut merupakan rancangan *interface* halaman *register* iterasi-1 dapat dilihat pada gambar 4.26 :



Gambar 4.26 Halaman register iterasi-1

3. Rancangan *interface* halaman dashboard iterasi-1

Halaman dashboard terdiri dari menu yang terdiri dari menu sekolah, jurusan, keas, siswa, guru/staff, anggota, transaksi, buku, dan juga denda yang terdapat pada sidebar. Semua menu yang terdapat pada sidebar tersebut, memiliki fungsi untuk melihat, mengubah, menghapus, dan juga untuk menambah kecuali pada menu sekolah hanya terdapat fungsi untuk melihat dan fungsi untuk mengubah. Kemudian pada halaman utama menampilkan informasi anggota, transaksi, dan juga buku. Halaman utama tersebut hanya menampilkan informasi saja tanpa ada fungsi yang lain seperti yang terdapat pada menu *sidebar*. Selain itu juga terdapat tulisan selamat datang. Dan pada ujung sebelah kanan dengan nama user merupakan *dropdown* yang terdapat tombol untuk logout dan juga nama user yang sedang login. Berikut merupakan rancangan *interface* halaman dashboard iterasi-1 dapat dilihat pada gambar 4.27 :



Gambar 4.27 Halaman dashboard iterasi-1

4. Rancangan interface halaman anggota iterasi-1

Halaman anggota menampilkan tab untuk melihat anggota siswa dan juga anggota guru maupun staff. Terdiri dari tombol untuk menambah dan tombol *export* yang terletak dibawah *heading* tabel anggota. Dan untuk fungsi lihat yang merupakan untuk melihat detail, dan juga mengubah, serta menghapus terdapat di dalam tabel. Berikut merupakan rancangan *interface* (admin) halaman anggota iterasi-1 dapat dilihat pada gambar 4.28 :

SIPS SMKN11		User					
User	Sekolah	Daftar Anggota Siswa		Daftar Anggota Staff			
	Jurusan Kelas Siswa Guru/Staff Anggota Transaksi Buku Denda	+ Export	No.	Nama	Nama Pendek	Nomor Anggota	Aksi
			1.	Mila Sari	Mila	217	Ubah Hapus

Gambar 4.28 Halaman anggota (Admin) iterasi-1

5. Rancangan interface halaman transaksi iterasi-1

Halaman transaksi terdiri dari mesin pencari yang terletak dibawah heading tabel, kemudian tombol tambah yang terletak di bawah mesin pencari. Dan untuk tabel

menampilkan id, nomor anggota, nama anggota, judul buku, tanggal pinjam, tanggal kembali, status, dan juga fungsi mengubah, serta mengembalikan buku. Berikut merupakan rancangan *interface* (admin) halaman transaksi iterasi-1 dapat dilihat pada gambar 4.29 :

Gambar 4.29 Halaman transaksi (Admin) iterasi-1

6. Rancangan *inteface* halaman buku iterasi-1

Halaman buku terdapat mesin pencari yang terletak di bawah *heading* tabel, tombol tambah di bawah mesin pencari, dan di dalam tabel terdapat kode buku, judul buku, penulis, tahun, penerbit, isbn, dan total buku. Sedangkan untuk fungsi melihat detail mengubah, serta menghapus terletak disamping total buku. Berikut merupakan rancangan *interface* (admin) halaman buku iterasi-1 dapat dilihat pada gambar 4.30 :

Gambar 4.30 Halaman buku (Admin) iterasi-1

7. Rancangan *interface* halaman denda iterasi-1

Halaman denda menampilkan Id transaksi, total hari, bayar, dan tanggal dibuat. Serta terdapat fungsi untuk melihat, mengubah, dan menghapus. Sedangkan untuk tombol menambah terdapat di bawah heading tabel. Berikut merupakan rancangan *interface* (admin) halaman denda iterasi-1 dapat dilihat pada gambar 4.31 :

Gambar 4.31 Halaman denda (Admin) iterasi-1

4.2.2 Proses iterasi ke-2

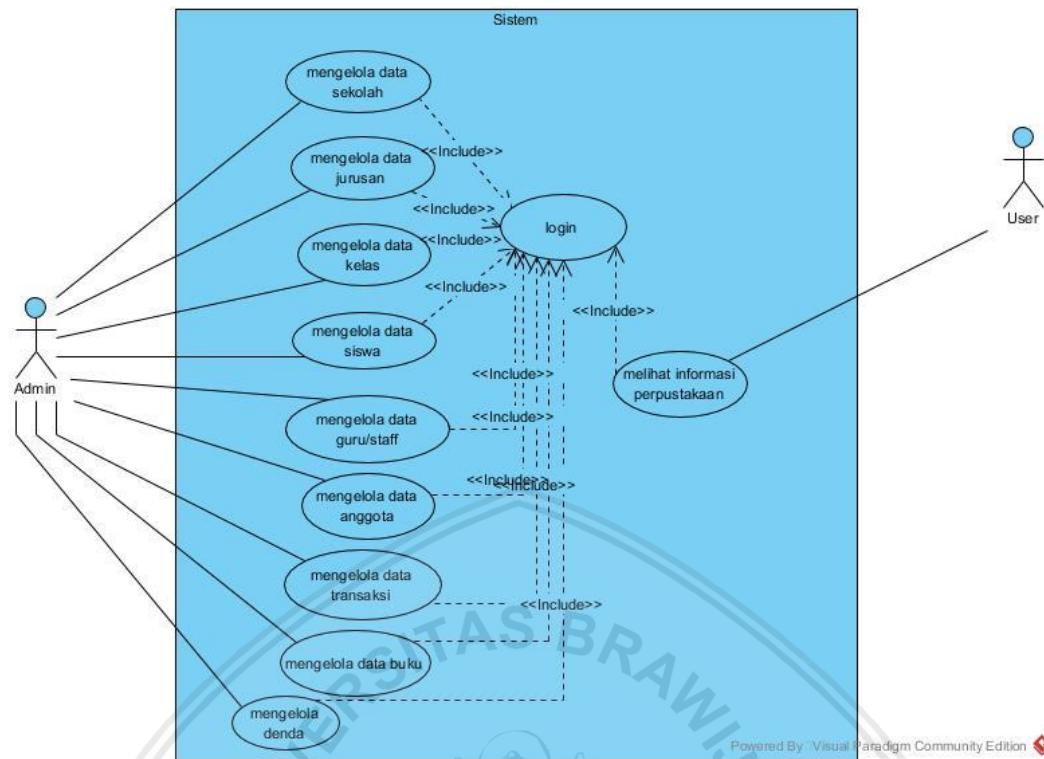
Pada proses iterasi ke-2 dilakukan penambahan nama jurusan pada menu buku. Hal ini berdasarkan hasil dari evaluasi iterasi pertama pada lampiran B.1. Dan use case user hanya dapat mendaftar (*register anggota*) dan juga melihat informasi perpustakaan. Hal ini dikarenakan agar tidak terjadi kesalahan memasukkan data. Sehingga, jika meminjam melalui admin terlebih dahulu. Berikut merupakan workshop desain pada iterasi ke-2 :

4.2.2.1. Use Case Diagram Iterasi-2

Use case diagram menggambarkan perilaku sistem informasi perpustakaan SMKN 11 Malang.

1. Use case sistem iterasi-2

Pada *use case* sistem iterasi-2 terdiri dari admin yang dapat mengelola data sekolah, data jurusan, data kelas, data siswa, data guru/staff, data angota, data transaksi, data buku dan juga data denda. Sedangkan *use case* untuk user yang terdiri dari siswa, guru, dan staff dapat melihat informasi perpustakaan, tetapi siswa maupun guru dan staff harus terlebih dahulu login. Berikut *use case* sistem pada iterasi ke-2 dapat dilihat pada gambar 4.32 :



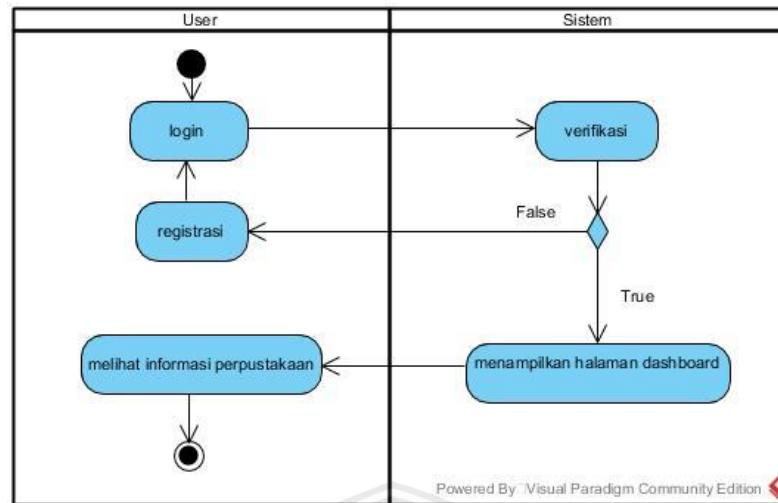
Gambar 4.32 Use case diagram sistem iterasi-2

4.2.2.2. Activity Diagram Iterasi-2

Activity diagram menggambarkan aktivitas sistem informasi perpustakaan SMKN 11 Malang.

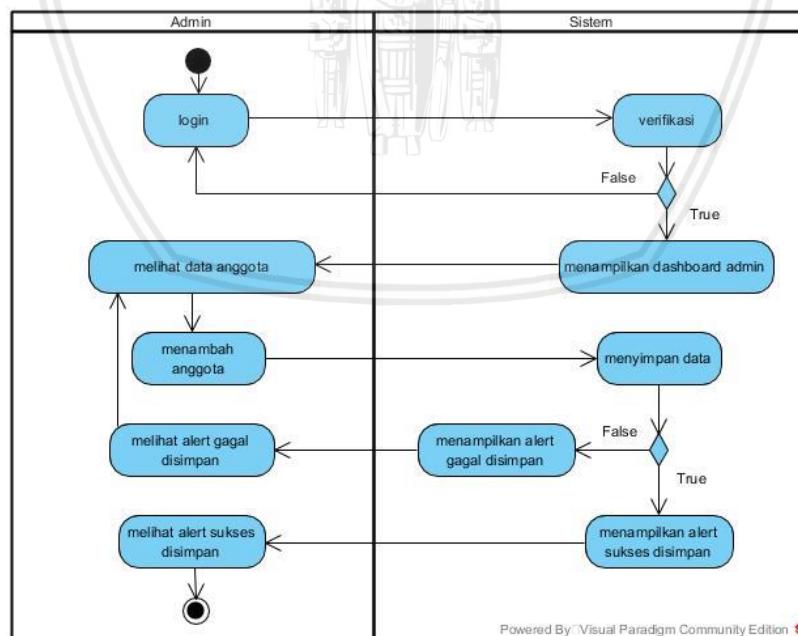
1. Activity diagram user iterasi -2

Activity diagram siswa terlebih dahulu user harus login yang kemudian diverifikasi oleh sistem jika data yang dimasukkan valid maka akan sistem akan menampilkan halaman dashboard yang berisi informasi tentang perpustakaan. Tetapi, jika data user tidak ditemukan maka user harus mendaftar menjadi anggota terlebih dahulu. Berikut desain activity diagram user iterasi ke-2 sistem informasi perpustakaan dapat dilihat pada gambar 4.33:

**Gambar 4.33 Activity diagram user iterasi-2**

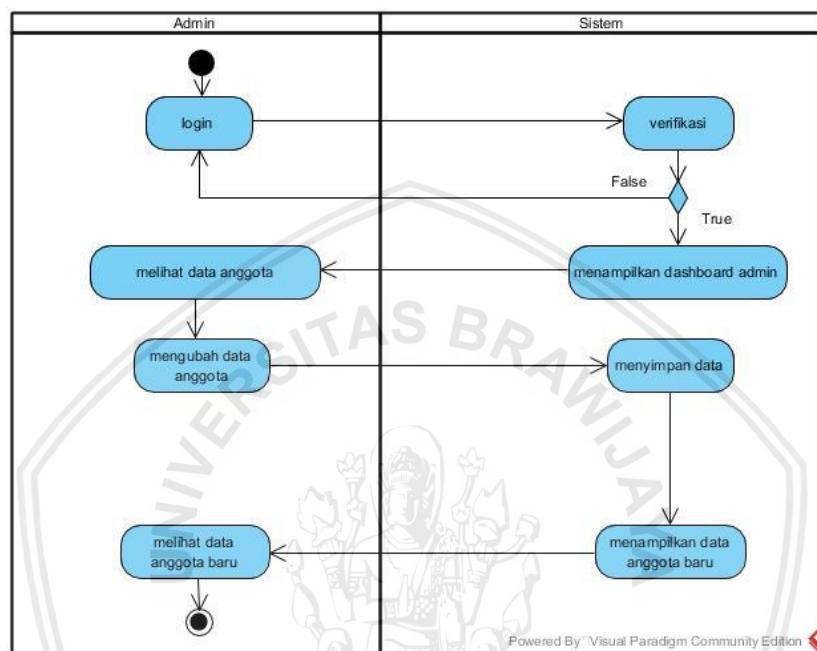
2. Activity diagram tambah anggota iterasi-2

Pada *activity diagram* untuk menambah anggota yang hanya dapat diakses oleh admin, terlebih dahulu admin login setelah verifikasi sistem akan menampilkan halaman dashboard admin. Di halaman data anggota, admin dapat menambahkan anggota dengan menekan tombol tambah yang menampilkan halaman untuk menginputkan data-data anggota. Jika berhasil, sistem akan menyimpannya di database. Jika berhasil, admin dapat melihat alert sukses disimpan, sebaliknya jika gagal, admin mendapat pesan alert gagal disimpan dan begitu seterusnya. Berikut activity diagram tambah anggota pada iterasi ke 2 dapat dilihat pada gambar 4.34:

**Gambar 4.34 Activity diagram tambah anggota (Admin) iterasi-2**

3. Activity diagram edit anggota iterasi-2

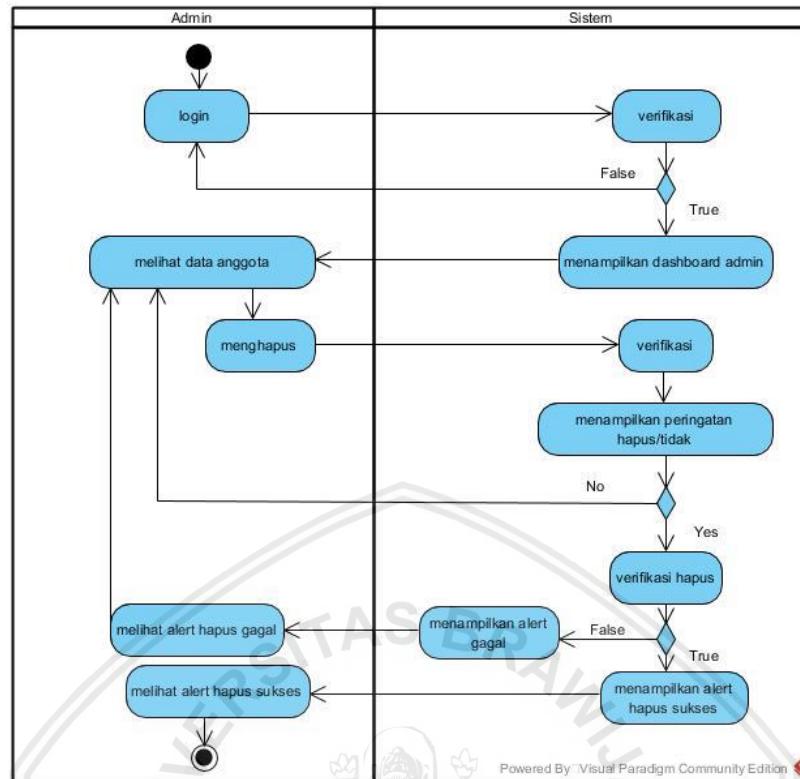
Activity diagram untuk edit anggota hanya dapat diakses oleh admin, terlebih dahulu admin login setelah berhasil login sistem menampilkan dashboard admin. Pada halaman data anggota, admin dapat menekan tombol ubah pada anggota yang dikehendaki, setelah mengubah data anggota, sistem menyimpan data tersebut dan menampilkan data anggota yang sudah diubah. Berikut activity diagram edit anggota pada iterasi ke 2 dapat dilihat pada gambar 4.35 :



Gambar 4.35 Activity Diagram edit anggota (Admin) iterasi-2

4. Activity diagram hapus anggota iterasi-2

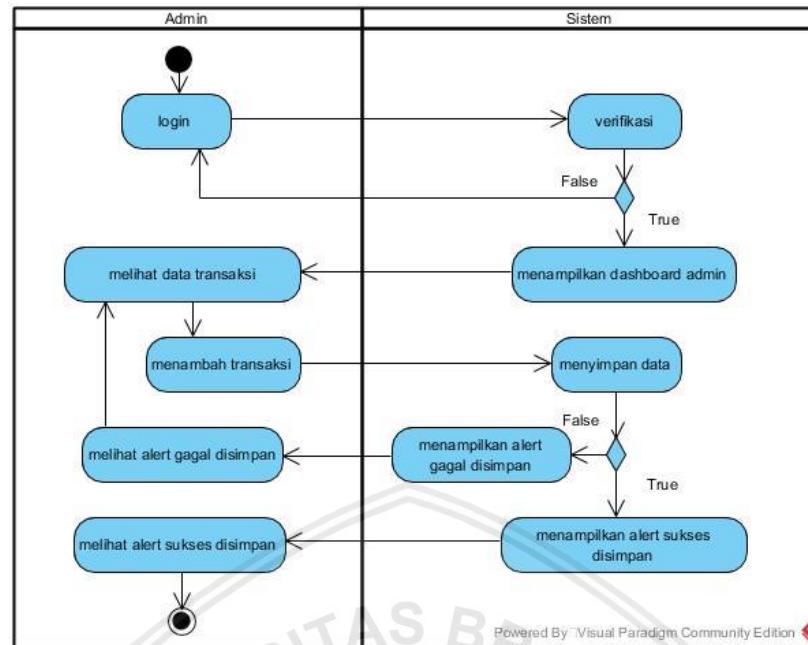
Activity diagram untuk menghapus anggota pada halaman data anggota, admin dapat menekan tombol hapus, setelah itu sistem memberikan tanggapan verifikasi berupa peringatan hapus atau tidak. Jika ya, maka sistem akan verifikasi perintah hapus, jika tidak sistem akan kembali ke halaman data anggota. Dan ketika sistem berhasil dihapus akan menampilkan alert hapus sukses, jika tidak berhasil dihapus akan menampilkan *alert gagal hapus* dan kembali ke halaman data anggota. Berikut *activity diagram* hapus anggota pada iterasi ke 2 dapat dilihat pada gambar 4.36:



Gambar 4.36 Activity Diagram hapus anggota (Admin) iterasi-2

5. Activity diagram tambah transaksi iterasi-2

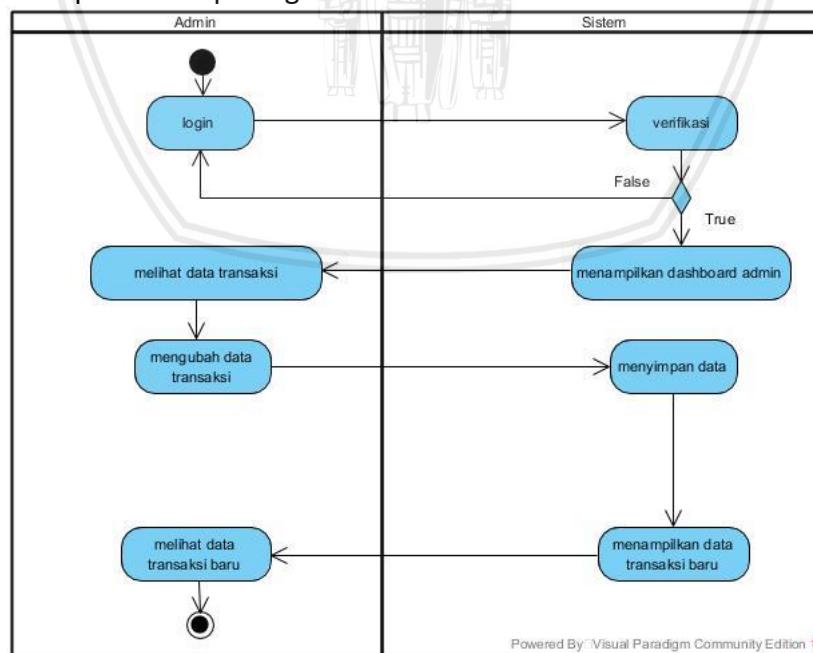
Activity diagram untuk menambah transaksi terlebih dahulu admin login setelah verifikasi sistem akan menampilkan halaman dashboard admin. Di halaman data transaksi, admin dapat menambahkan transaksi dengan menekan tombol tambah yang menampilkan halaman untuk menginputkan transaksi peminjaman. Jika sudah, sistem akan menyimpannya di database. Jika berhasil, admin dapat melihat alert sukses disimpan, sebaliknya jika gagal, admin mendapat pesan alert gagal disimpan dan begitu seterusnya. Berikut *activity diagram* tambah transaksi pada iterasi ke 2 dapat dilihat pada gambar 4.37 :



Gambar 4.37 Activity diagram tambah transaksi (Admin) iterasi-2

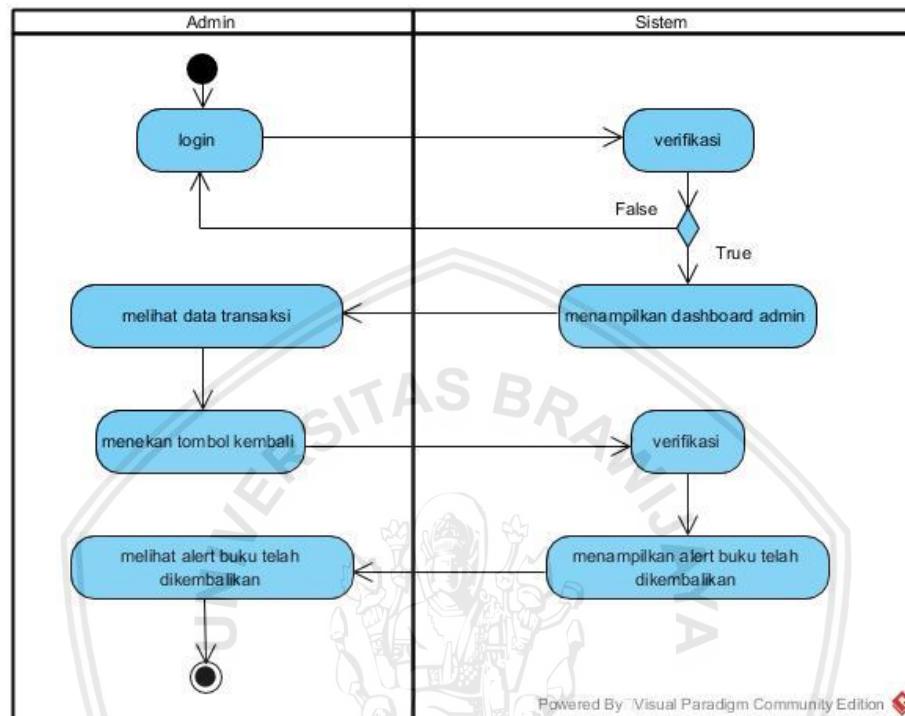
6. Activity diagram edit transaksi iterasi-2

Activity diagram untuk edit transaksi terlebih dahulu admin login setelah berhasil login sistem menampilkan dashboard admin. Pada halaman data transaksi, admin dapat menekan tombol ubah pada transaksi anggota yang dikehendaki, setelah mengubah data transaksi, sistem menyimpan data tersebut dan menampilkan data transaksi yang sudah diubah. Berikut activity diagram edit transaksi pada iterasi ke 2 dapat dilihat pada gambar 4.38 :



Gambar 4.38 Activity Diagram edit transaksi (Admin) iterasi-2

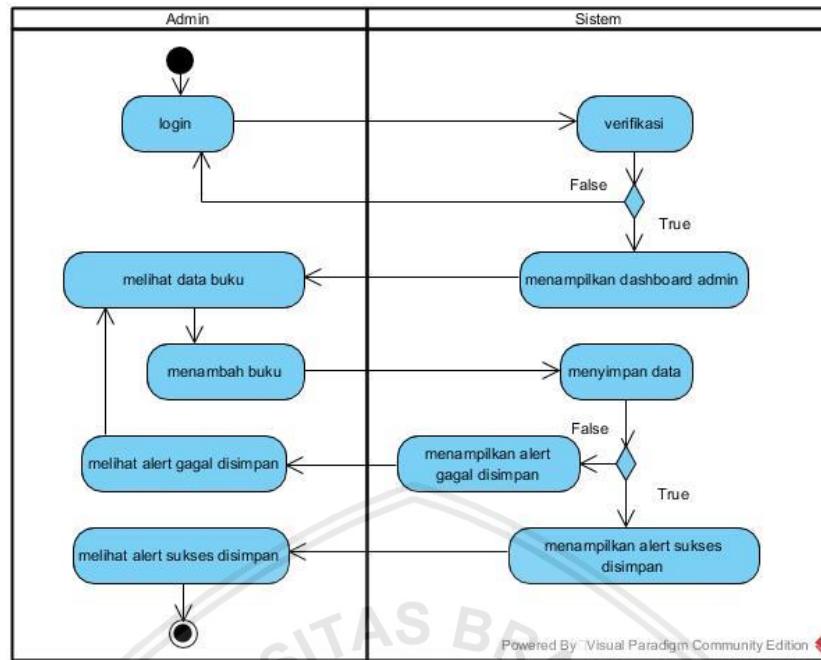
7. Activity diagram mengembalikan buku pada menu transaksi iterasi-2
Activity diagram untuk mengembalikan buku pada menu transaksi, admin dapat menekan tombol kembali, setelah itu sistem menampilkan alert buku telah dikembalikan dan kembali ke halaman data transaksi. Berikut *activity diagram* mengembalikan buku pada menu transaksi iterasi ke 2 dapat dilihat pada gambar 4.39 :



Gambar 4.39 Activity Diagram mengembalikan buku (Admin) iterasi-2

8. Activity diagram tambah buku iterasi-2

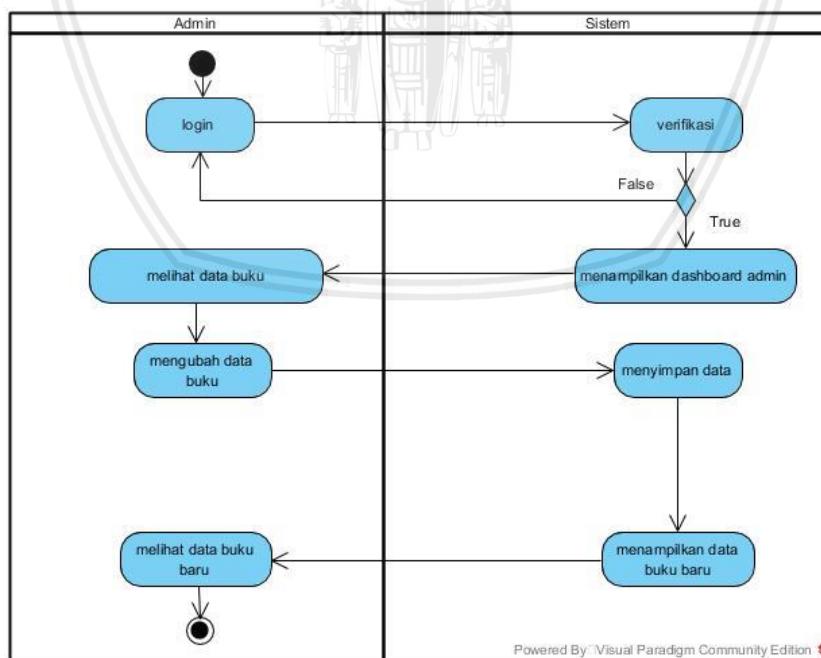
Activity diagram untuk menambah buku, pada halaman data buku, admin dapat menambahkan buku dengan menekan tombol tambah yang menampilkan halaman untuk menginputkan tambah buku. Jika sudah, sistem akan menyimpannya di database. Jika berhasil, admin dapat melihat alert sukses disimpan, sebaliknya jika gagal, admin mendapat pesan alert gagal disimpan dan begitu seterusnya. Berikut *activity diagram* tambah buku pada iterasi ke 2 dapat dilihat pada gambar 4.40 :



Gambar 4.40 Activity diagram tambah buku (Admin) iterasi-2

9. Activity diagram edit buku iterasi-2

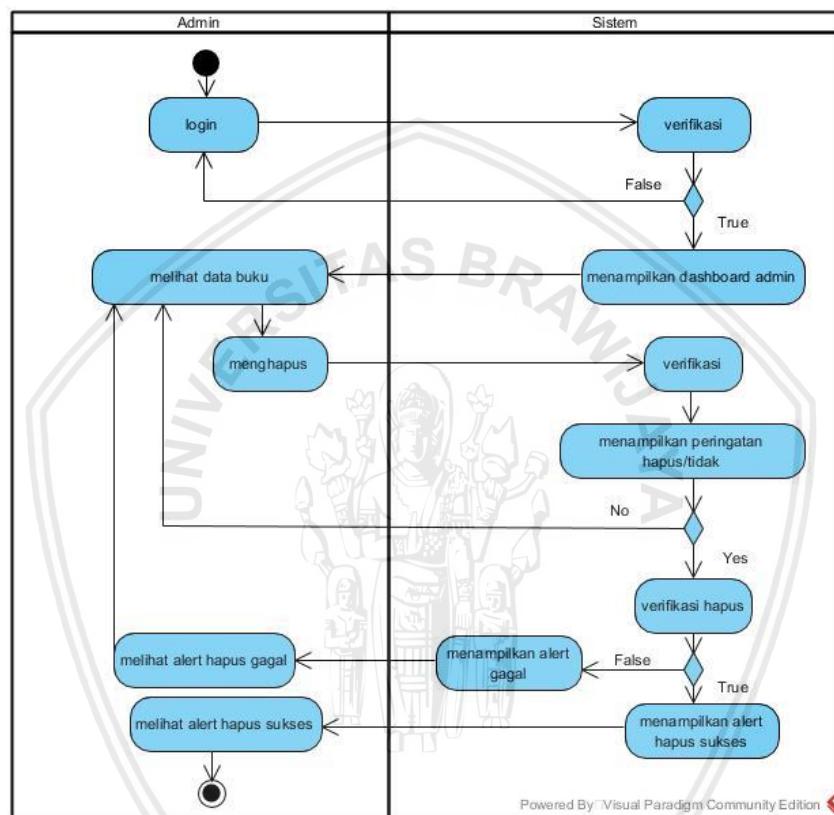
Activity diagram untuk edit buku, pada halaman data buku, admin dapat menekan tombol ubah pada halaman data buku yang dikehendaki, setelah mengubah data buku, sistem menyimpan data tersebut dan menampilkan data buku yang sudah diubah. Berikut *activity diagram* edit buku pada iterasi ke 2 dapat dilihat pada gambar 4.41:



Gambar 4.41 Activity Diagram edit buku (Admin) iterasi-2

10. Activity diagram hapus buku iterasi-2

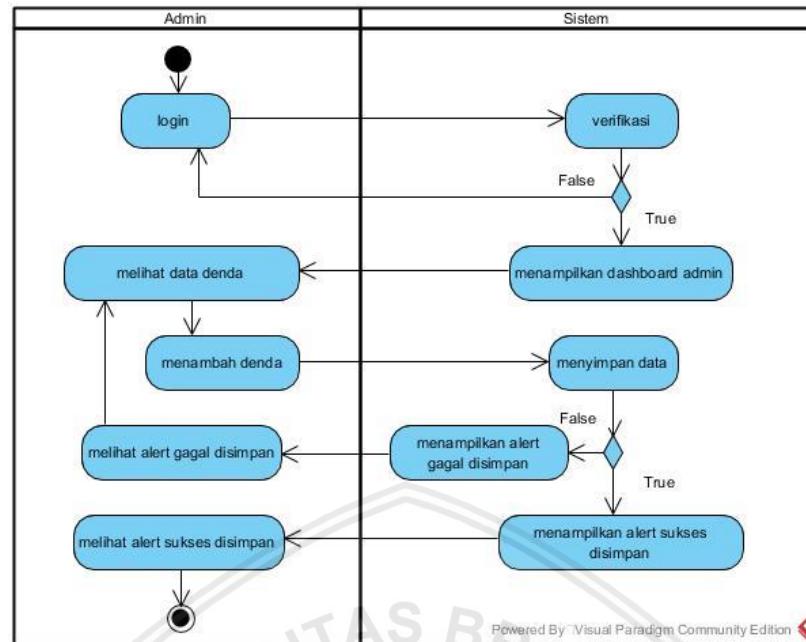
Activity diagram untuk menghapus buku pada halaman data buku, admin dapat menekan tombol hapus, setelah itu sistem memberikan tanggapan verifikasi berupa peringatan hapus atau tidak. Jika ya, maka sistem akan verifikasi perintah hapus, jika tidak sistem akan kembali ke halaman data buku. Dan ketika sistem berhasil dihapus akan menampilkan alert hapus sukses, jika tidak berhasil dihapus akan menampilkan *alert gagal hapus* dan kembali ke halaman data buku. Berikut *activity diagram* hapus buku pada iterasi ke 2 dapat dilihat pada gambar 4.42:



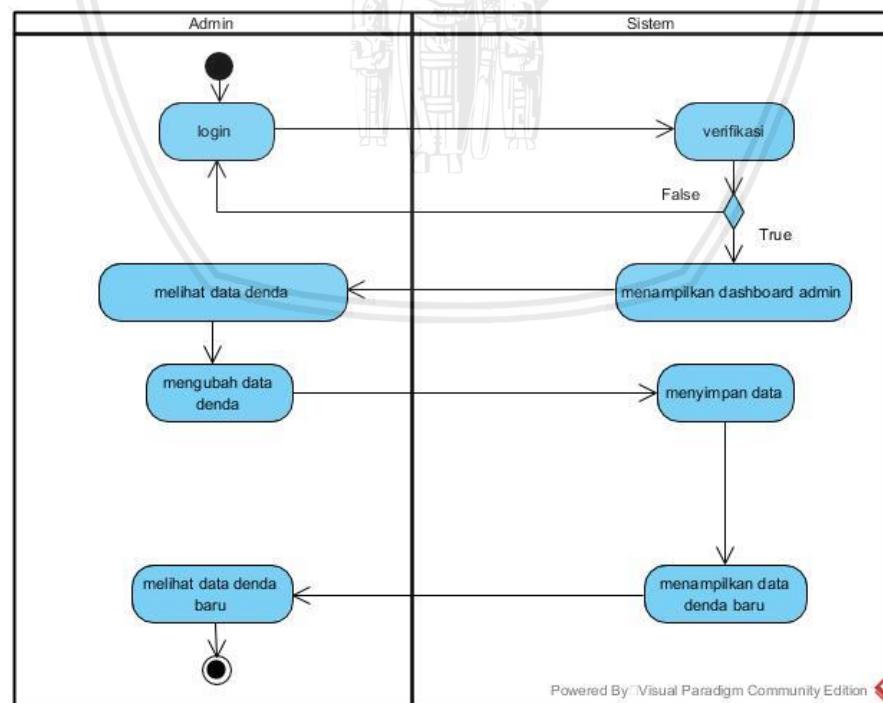
Gambar 4.42 Activity Diagram hapus buku (Admin) iterasi-2

11. Activity Diagram tambah denda (admin) iterasi-2

Activity diagram untuk menambah denda, pada halaman data denda, admin dapat menambahkan denda dengan menekan tombol tambah yang menampilkan halaman untuk menginputkan tambah denda. Jika sudah, sistem akan menyimpannya di database. Jika berhasil, admin dapat melihat alert sukses disimpan, sebaliknya jika gagal, admin mendapat pesan alert gagal disimpan dan begitu seterusnya. Berikut dapat dilihat *activity diagram* tambah denda dapat dilihat pada gambar 4.43 :

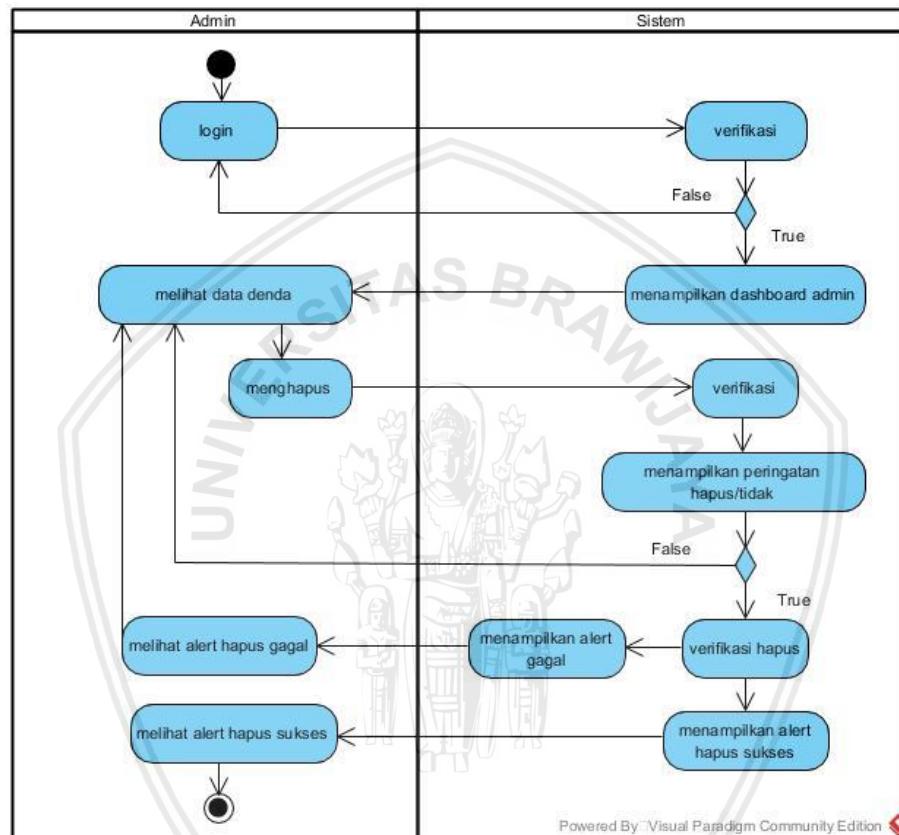
**Gambar 4.43 Activity Diagram tambah denda (Admin) iterasi-2****12. Activity Diagram edit denda (admin) iterasi-2**

Activity diagram untuk edit denda, pada halaman data denda, admin dapat menekan tombol ubah pada halaman data denda yang dikehendaki, setelah mengubah data denda, sistem menyimpan data tersebut dan menampilkan data denda yang sudah diubah. Berikut dapat dilihat pada gambar 4.44 :

**Gambar 4.44 Activity Diagram edit denda (Admin) iterasi-2**

13. Activity Diagram hapus denda (admin) iterasi-2

Activity diagram untuk menghapus denda pada halaman data denda, admin dapat menekan tombol hapus, setelah itu sistem memberikan tanggapan verifikasi berupa peringatan hapus atau tidak. Jika ya, maka sistem akan verifikasi perintah hapus, jika tidak sistem akan kembali ke halaman data denda. Dan ketika sistem berhasil dihapus akan menampilkan alert hapus sukses, jika tidak berhasil dihapus akan menampilkan alert gagal hapus dan kembali ke halaman data denda. Berikut dapat dilihat pada gambar 4.45 :



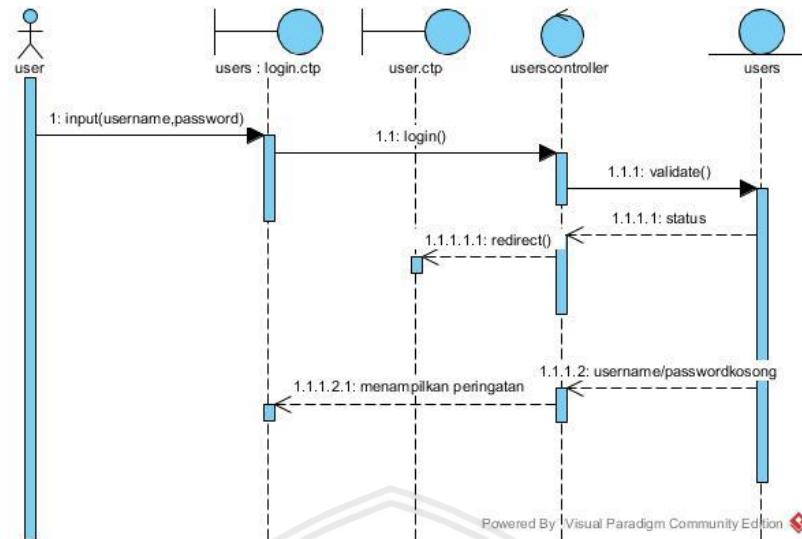
Gambar 4.45 Activity Diagram hapus denda (Admin) iterasi-2

4.2.2.3. Sequence Diagram Iterasi-2

Sequence diagram menggambarkan perilaku pada *use case* sistem informasi perpustakaan SMKN 11 Malang berupa pesan yang terkirim dan diterima .

1. Sequence diagram login user iterasi-2

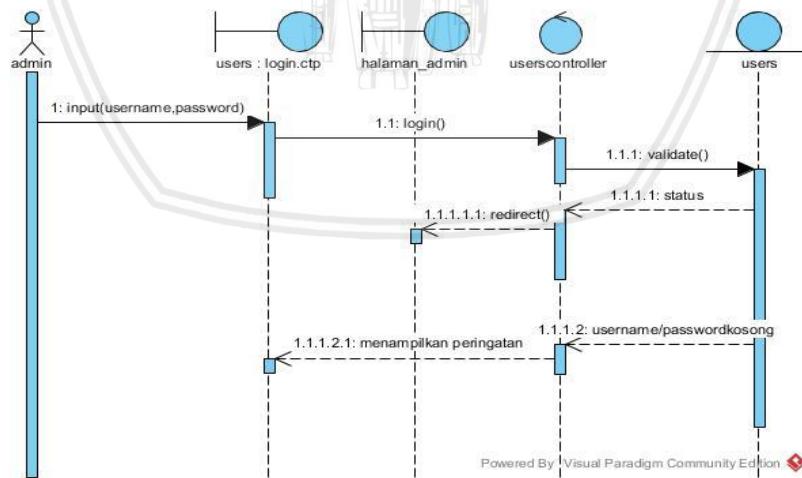
Pada desain *sequence diagram* login user, terlebih dahulu login dengan menginputkan *username* dan juga *password*. Ketika selesai menginputkan *username* dan *password* dan menekan tombol login, segera sistem memproses data yang masuk, mencari nama dan *password* yang diperintahkan di *database*. Jika berhasil proses pencarian ditemukan, sistem akan menampilkan halaman user. Tapi, jika data yang diinputkan tersebut tidak ditemukan di *database*, maka akan kembali ke halaman login. Berikut desain *sequence diagram* login user iterasi-2 sistem informasi perpustakaan dapat dilihat pada gambar 4.46:



Gambar 4.46 Sequence diagram login (User) iterasi-2

2. Sequence diagram login admin iterasi-2

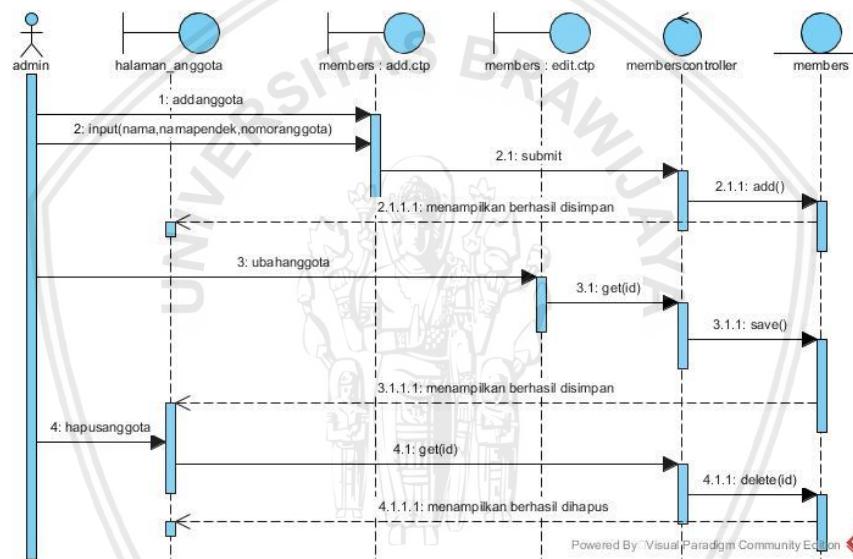
Pada desain *sequence diagram* login admin, terlebih dahulu login dengan menginputkan *username* dan juga *password*. Ketika selesai menginputkan *username* dan *password* dan menekan tombol login, segera sistem memproses data yang masuk, mencari nama dan password yang diperintahkan di *database*. Jika berhasil proses pencarian ditemukan, sistem akan menampilkan halaman dashboard. Tapi, jika data yang diinputkan tersebut tidak ditemukan di *database*, maka sistem kembali ke halaman login. Berikut desain *sequence diagram* login admin iterasi-2 sistem informasi perpustakaan dapat dilihat pada gambar 4.47:



Gambar 4.47 Sequence diagram login (Admin) iterasi-2

3. Sequence diagram kelola anggota iterasi-2

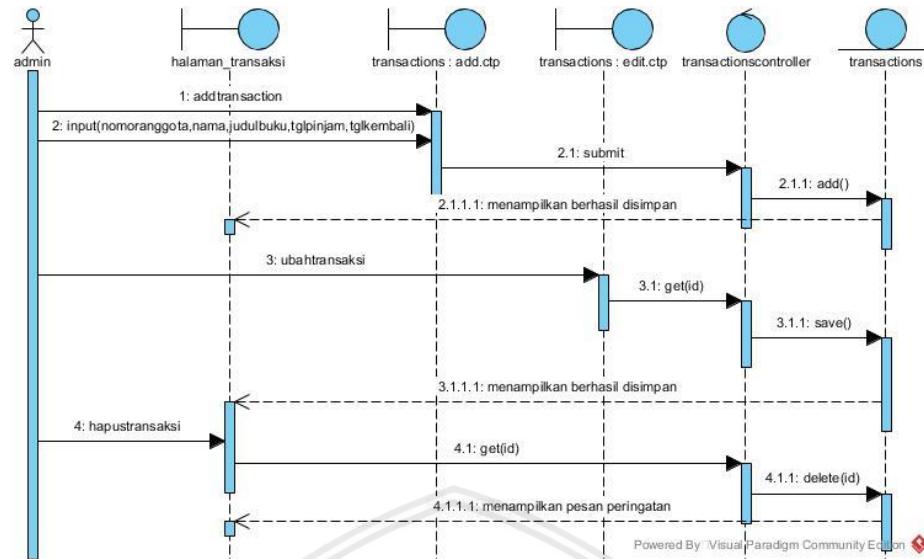
Pada *sequence diagram* kelola anggota, admin dapat melihat data anggota dengan masuk ke halaman anggota dan menekan tombol lihat maka sistem memprosesnya dan memanggil data di database kemudian menampilkan ke admin. Admin juga dapat mengubah data anggota dengan menekan tombol ubah yang berisi data yang akan diubah dan menekan tombol submit, maka sistem akan menyimpan data yang diinputkan tersebut. Kemudian jika admin ingin menambah data, admin menekan tombol tambah yang menampilkan halaman tambah yang berisi inputan. Setelah selesai diinputkan, dan data baru disimpan di database. Dan jika admin ingin menghapus data anggota, maka sistem akan memprosesnya mengambil data di database yang kemudian sistem menjalankan proses penghapusan data tersebut. Berikut desain *sequence diagram* kelola anggota iterasi-2 sistem informasi perpustakaan dapat dilihat pada gambar 4.48:



Gambar 4.48 Sequence diagram kelola anggota (Admin) iterasi-2

4. Sequence diagram kelola transaksi iterasi-2

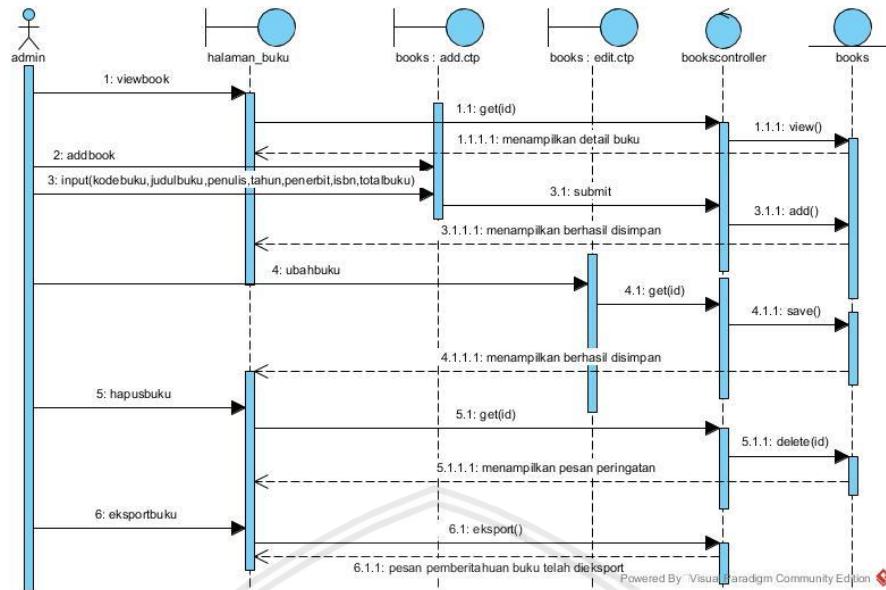
Sequence diagram kelola transaksi, admin dapat melihat data transaksi dengan masuk ke halaman transaksi dan menekan tombol lihat maka sistem akan memprosesnya dan memanggil data di database kemudian menampilkan ke admin. Admin juga dapat mengubah data transaksi dengan menekan tombol ubah yang berisi data yang akan diubah dan menekan tombol submit, maka sistem akan menyimpan data yang diinputkan tersebut. Kemudian jika admin ingin menambah data, admin menekan tombol tambah yang menampilkan halaman tambah yang berisi inputan. Setelah selesai diinputkan, data akan disimpan di database. Dan jika admin ingin mengembalikan data transaksi, maka sistem akan memprosesnya mengambil data di database yang kemudian sistem menjalankan proses pengembalian data tersebut. Berikut desain *sequence diagram* kelola transaksi iterasi-2 sistem informasi perpustakaan dapat dilihat pada gambar 4.49:



Gambar 4.49 Sequence diagram kelola transaksi (Admin) iterasi-2

5. Sequence diagram kelola buku iterasi-2

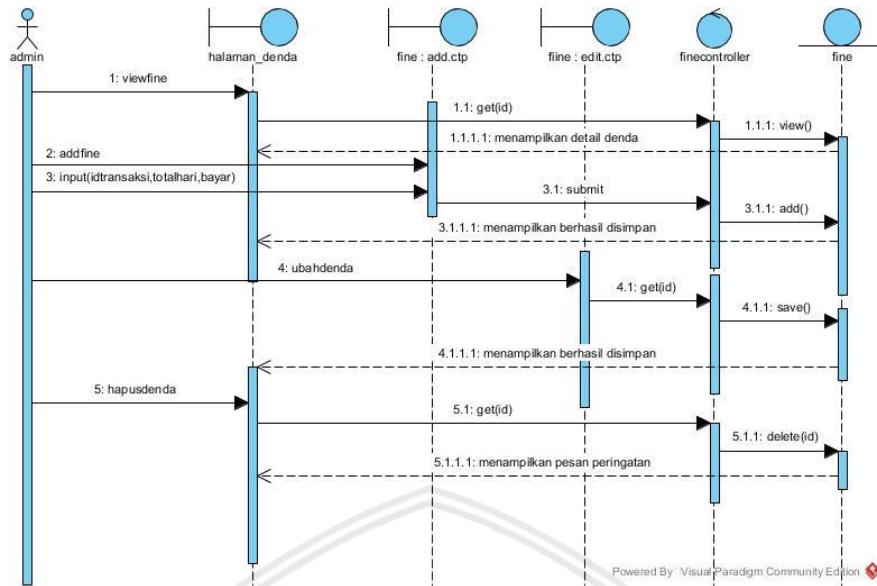
Sequence diagram kelola buku, admin dapat melihat data buku dengan masuk ke halaman buku dan menekan tombol lihat maka sistem akan memprosesnya dan memanggil data di database kemudian menampilkan ke admin. Admin juga dapat mengubah data buku dengan menekan tombol ubah yang berisi data yang akan diubah dan menekan tombol submit, maka sistem akan menyimpan data yang diinputkan tersebut. Kemudian jika admin ingin menambah data, admin menekan tombol tambah yang menampilkan halaman tambah yang berisi inputan. Setelah selesai diinputkan, data akan disimpan di database. Dan jika admin ingin menghapus data buku, maka sistem akan memprosesnya mengambil data di database yang kemudian sistem menjalankan proses penghapusan data tersebut. Berikut desain *sequence diagram* kelola buku iterasi-2 sistem informasi perpustakaan dapat dilihat pada gambar 4.50:



Gambar 4.50 Sequence Diagram kelola buku (Admin) iterasi-2

6. Sequence Diagram kelola denda (Admin) iterasi-2

Sequence diagram kelola denda, admin dapat melihat data denda dengan masuk ke halaman denda dan menekan tombol lihat maka sistem akan memprosesnya dan memanggil data di database kemudian menampilkan ke admin. Admin juga dapat mengubah data denda dengan menekan tombol ubah yang berisi data yang akan diubah dan menekan tombol submit, maka sistem akan menyimpan data yang diinputkan tersebut. Kemudian jika admin ingin menambah data, admin menekan tombol tambah yang menampilkan halaman tambah yang berisi inputan. Setelah selesai diinputkan, data akan disimpan di database. Dan jika admin ingin menghapus data denda, maka sistem akan memprosesnya mengambil data di database yang kemudian sistem menjalankan proses penghapusan data tersebut. Berikut dapat dilihat *sequence diagram* kelola denda pada gambar 4.51 :



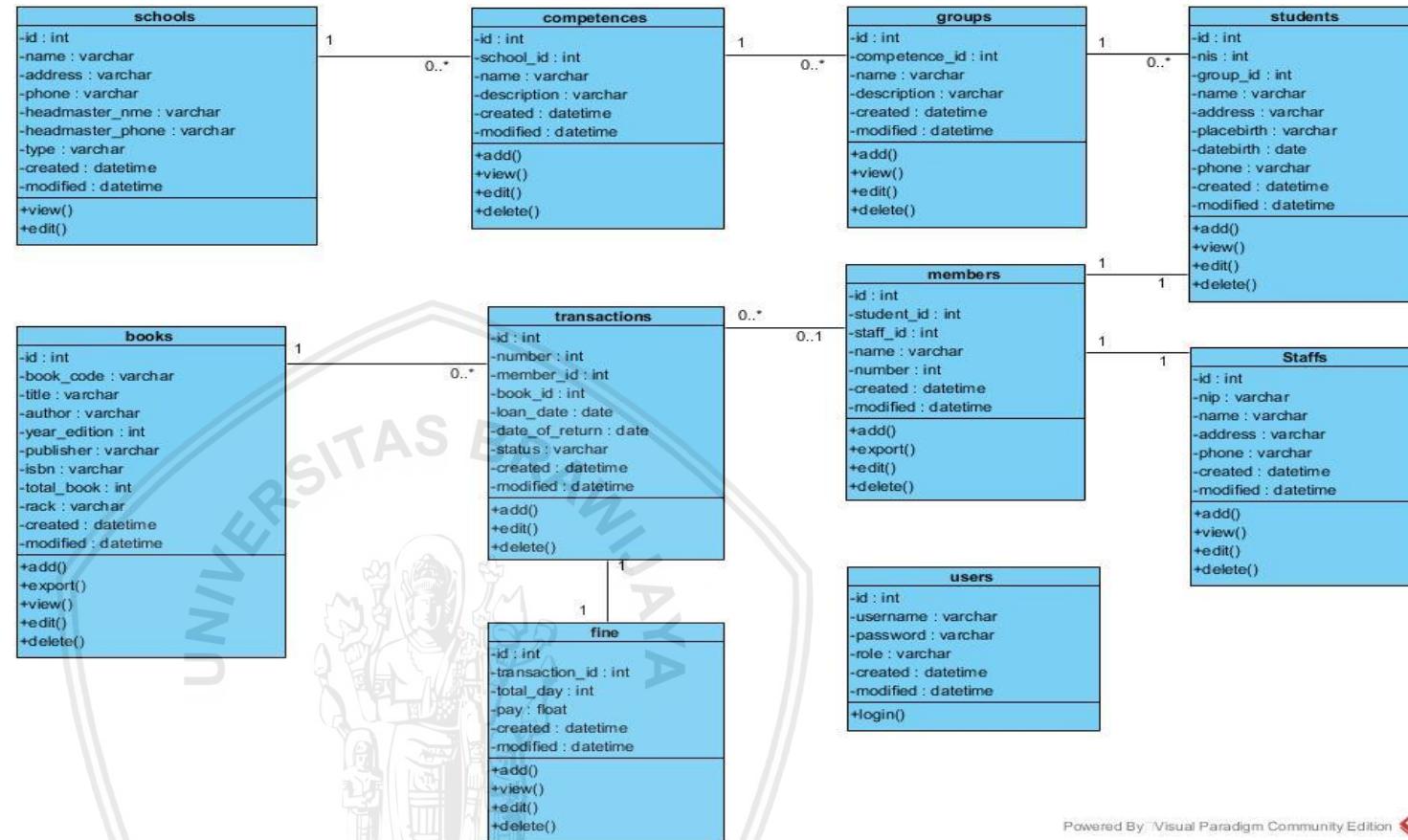
Gambar 4.51 Sequence Diagram kelola denda (Admin) iterasi-2

4.2.2.4. Class Diagram Iterasi-2

Class diagram menggambarkan struktur database sistem informasi perpustakaan SMKN 11 Malang. Class diagram bertujuan untuk mendefinisikan kelas-kelas yang terdapat pada sistem. Berikut dapat dilihat class diagram iterasi -2 pada gambar 4.52:

1. Class diagram sistem informasi perpustakaan iterasi-2

Berikut desain *class diagram* iterasi ke 2 sistem informasi perpustakaan dapat dilihat pada gambar 4.52 :



Gambar 4.52 Class diagram sistem informasi perpustakaan iterasi-2

4.2.2.5. Perancangan database iterasi-2

Pada tahap ini dilakukan perancangan database sesuai dengan *class diagram* yang telah dibuat sebelumnya. Berikut rancangan database dapat dilihat pada tabel berikut:

1. Tabel sekolah iterasi-2

Nama tabel : Schools

Primary key : Id

Foreign key : Id, name, type

Tabel 4.13 Tabel sekolah iterasi-2

Nama	Tipe
id	int(10)
name	varchar(100)
address	varchar(255)
phone	varchar(20)
headmaster_nme	varchar(50)
headmaster_phone	varchar(20)
type	varchar(255)
created	datetime
modified	datetime

2. Tabel jurusan iterasi-2

Nama tabel : competences

Primary key : Id

Foreign key : Id

Tabel 4.14 Tabel jurusan iterasi-2

Nama	Tipe
id	int(10)
school_id	int(10)
name	varchar(255)
description	varchar(255)
created	datetime
modified	datetime

3. Tabel kelas iterasi-2

Nama tabel : groups

Primary key : Id

Foreign key : Id, name

Tabel 4.15 Tabel kelas iterasi-2

Nama	Tipe
id	int(10)
competence_id	int(10)
name	varchar(255)
description	varchar(255)
created	datetime
modified	datetime

4. Tabel siswa iterasi-2

Nama tabel : students

Primary key : Id

Foreign key : Id, nis

Tabel 4.16 Tabel siswa iterasi-2

Nama	Tipe
id	int(10)
nis	int(10)
group_id	int(10)
name	varchar(50)
address	varchar(255)
placebirth	varchar(25)
datebirth	date
phone	varchar(20)
created	datetime
modified	datetime

5. Tabel guru/staff iterasi-2

Nama tabel : staffs

Primary key : Id

Foreign key : Id, nip

Tabel 4.17 Tabel guru/staff iterasi-2

Nama	Tipe
id	int(10)
nip	varchar(60)
name	varchar(50)
address	varchar(255)
phone	varchar(20)
created	datetime
modified	datetime

6. Tabel user iterasi-2

Nama tabel : users

Primary key : Id

Foreign key : Id, username

Tabel 4.18 Tabel users iterasi-2

Nama	Tipe
id	int(10)
username	varchar(50)
password	varchar(255)
role	varchar(20)
created	datetime
modified	datetime

7. Tabel anggota iterasi-2

Nama tabel : members

Primary key : Id

Foreign key : Id, number

Tabel 4.19 Tabel anggota iterasi-2

Nama	Tipe
id	int(10)
student_id	int(10)
staff_id	int(10)
name	varchar(60)
number	int(10)
created	datetime
modified	datetime

8. Tabel transaksi iterasi-2

Nama tabel : transactions

Primary key : Id

Foreign key : Id

Tabel 4.20 Tabel transaksi iterasi-2

Nama	Tipe
id	int(10)
number	int(10)
member_id	int(10)
book_id	int(10)
book_code	varchar(10)
loan_date	date
date_of_return	date
status	varchar(60)
created	datetime
modified	datetime

9. Tabel buku iterasi-2

Nama tabel : books

Primary key : Id

Foreign key:Id, book_code, isbn

Tabel 4.21 Tabel buku iterasi-2

Nama	Tipe
id	int(10)
book_code	varchar(10)
title	varchar(255)
author	varchar(255)
year_edition	int(10)
publisher	varchar(255)
isbn	varchar(20)
total_book	int(10)
rack	varchar(12)
created	datetime
modified	datetime

10. Tabel denda iterasi-2

Nama tabel : Fine

Primary key : Id

Foreign key: Id

Tabel 4.22 Tabel denda iterasi-2

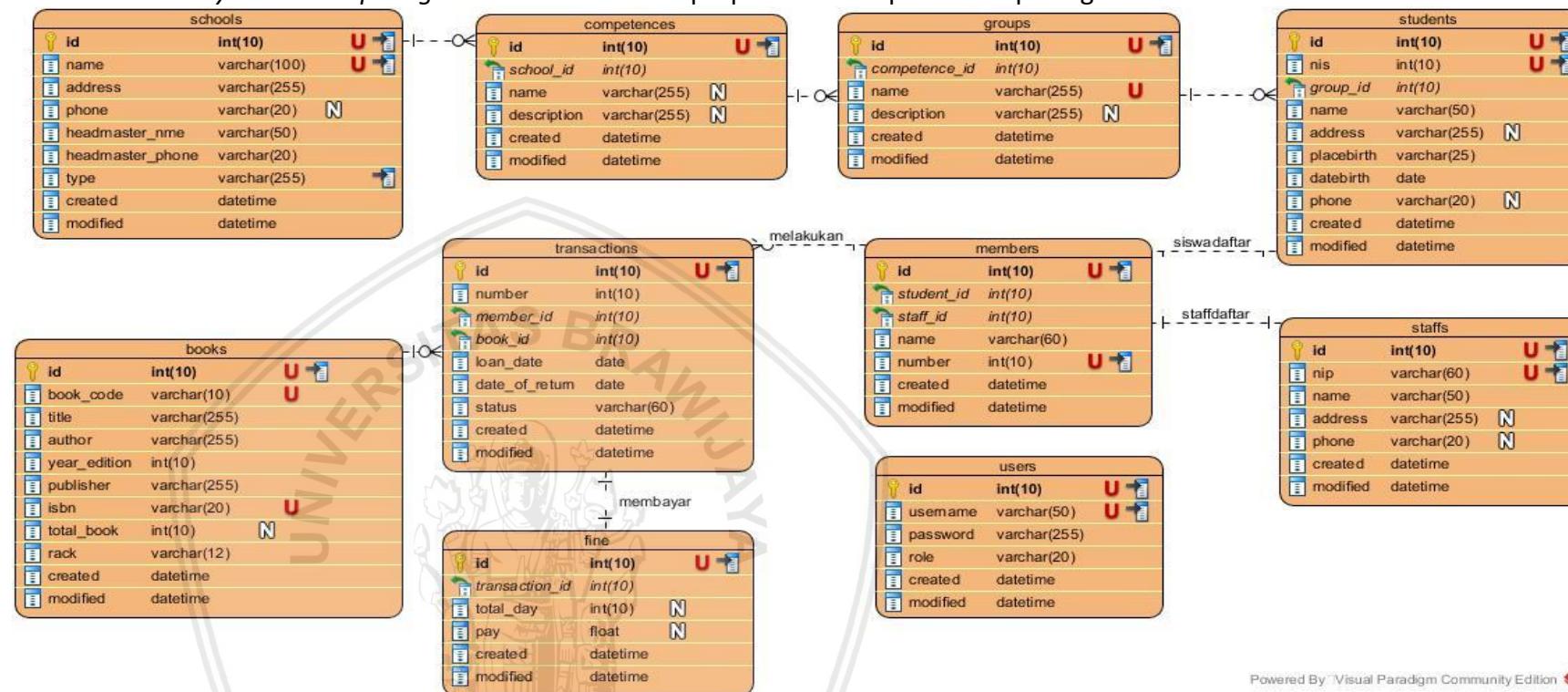
Nama	Tipe
id	int(10)
transaction_id	int(10)
total_day	int(10)
pay	float
created	datetime
modified	datetime

4.2.2.6. Entity Relationship Diagram iterasi-2

Entity Relationship Diagram merupakan gambaran dari hubungan diantara *entity* tersebut.

1. Entity Relationship Diagram Sistem Informasi Perpustakaan iterasi-2

Berikut desain *entity relationship diagram* sistem informasi perpustakaan dapat dilihat pada gambar 4.53 :



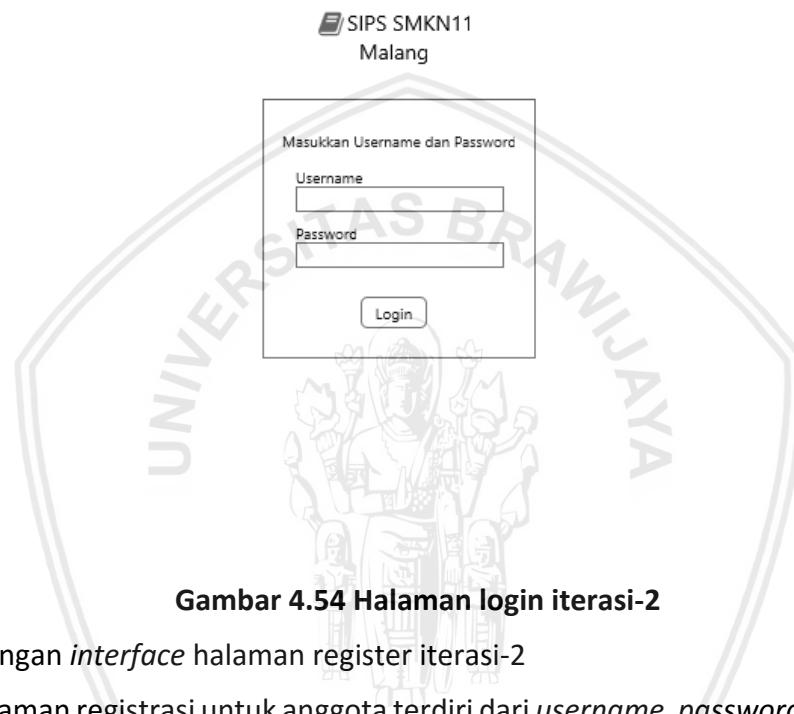
Gambar 4.53 Entity Relationship Diagram Sistem Informasi Perpustakaan iterasi-2

4.2.2.7. Rancangan antarmuka (*interface*) iterasi-2

Rancangan antarmuka (*inteface*) merupakan mekanisme komunikasi antara pengguna (*user*) dengan sistem.

1. Rancangan *interface* halaman login iterasi-2

Halaman login terdiri dari *username*, *password*, tombol login dan juga tombol untuk registrasi anggota yang mengarahkan ke halaman registrasi. Berikut rancangan *inteface* halaman login iterasi ke 2 dapat dilihat pada gambar 4.54:



Gambar 4.54 Halaman login iterasi-2

2. Rancangan *interface* halaman register iterasi-2

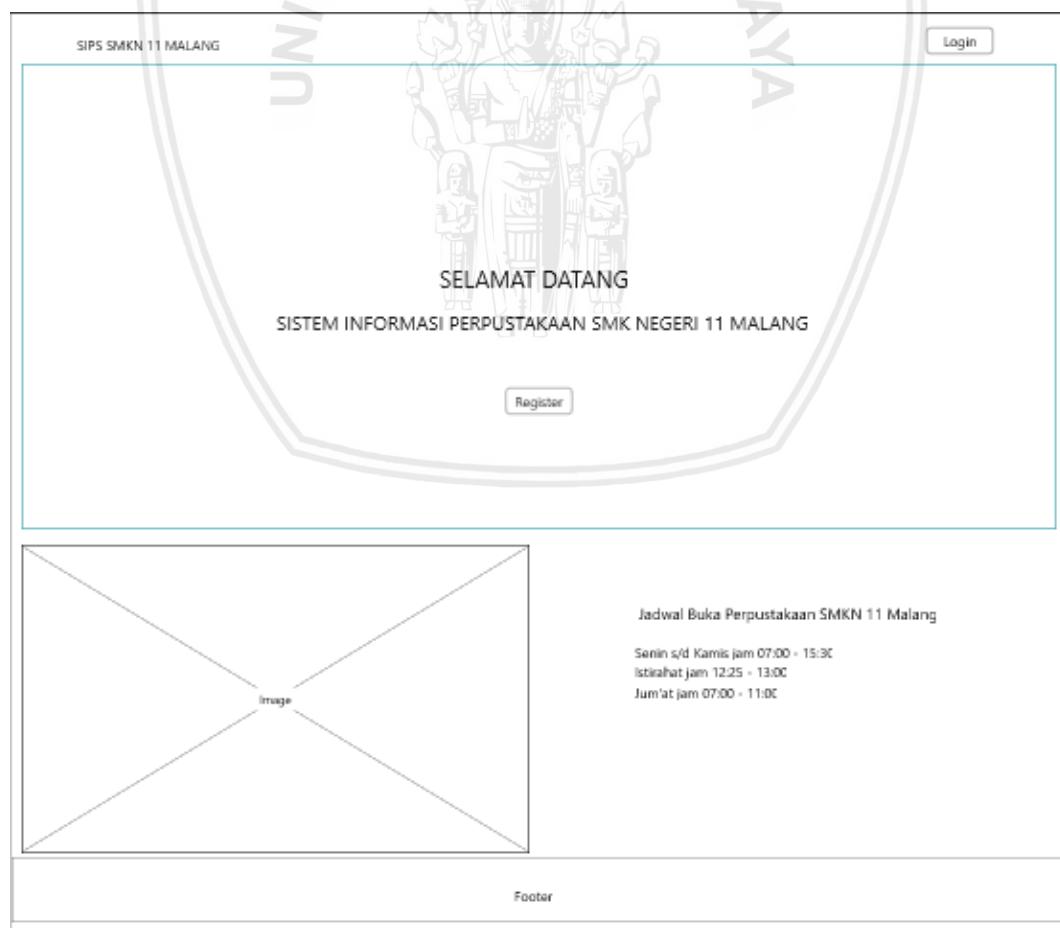
Pada halaman registrasi untuk anggota terdiri dari *username*, *password*, *role* untuk memilih jenis user siswa, atau user staff, tombol submit untuk menyimpan data pada database dan tombol *back* untuk kembali ke halaman login. Berikut rancangan *inteface* halaman register iterasi ke 2 dapat dilihat pada gambar 4.55:



Gambar 4.55 Halaman register iterasi-2

3. Rancangan *interface* halaman depan iterasi-2

Berikut rancangan *inteface* halaman depan iterasi ke 2 dapat dilihat pada gambar 4.56:



Gambar 4.56 Halaman depan

4. Rancangan *interface* halaman dashboard user iterasi-2

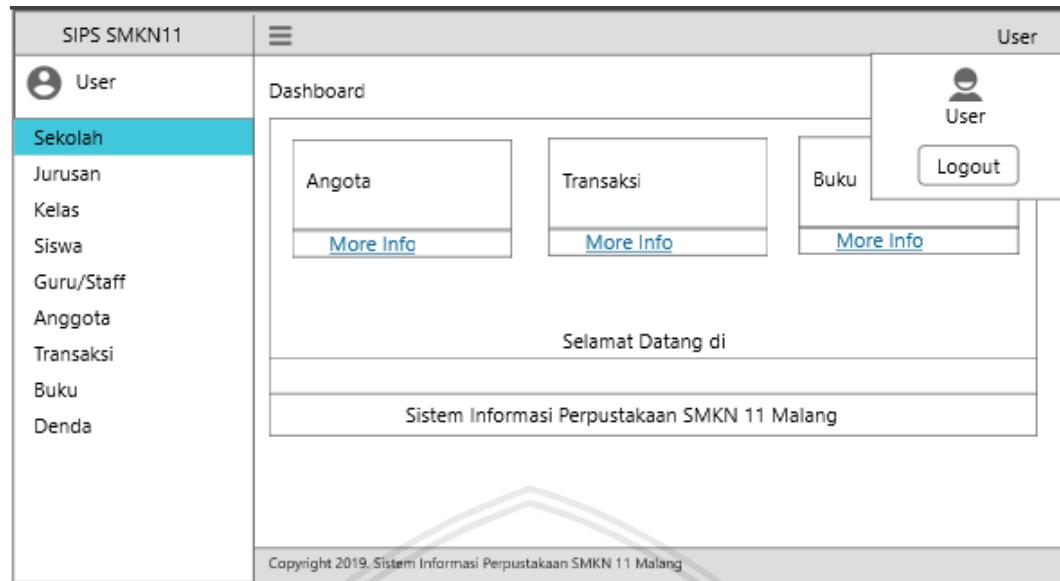
Berikut rancangan *inteface* halaman dashboard user iterasi ke 2 dapat dilihat pada gambar 4.57:



Gambar 4.57 Halaman dashboard user

5. Rancangan *inteface* halaman dashboard iterasi-2

Halaman dashboard terdiri dari menu yang terdiri dari menu sekolah, jurusan, keas, siswa, guru/staff, anggota, transaksi, buku, dan juga denda yang terdapat pada sidebar. Semua menu yang terdapat pada *sidebar* tersebut, memiliki fungsi untuk melihat, mengubah, menghapus, dan juga untuk menambah kecuali pada menu sekolah hanya terdapat fungsi untuk melihat dan fungsi untuk mengubah. Kemudian pada halaman utama menampilkan informasi anggota, transaksi, dan buku. Halaman utama tersebut menampilkan rekap data anggota, transaksi dan buku. Selain itu juga terdapat tulisan selamat datang. Dan pada ujung sebelah kanan dengan nama user merupakan *dropdown* yang terdapat tombol untuk *logout* dan juga nama user yang sedang login. Berikut merupakan rancangan *interface* halaman dashboard iterasi-2 dapat dilihat pada gambar 4.58:



Gambar 4.58 Halaman dashboard iterasi-2

6. Rancangan interface halaman anggota iterasi-2

Halaman anggota menampilkan tab untuk melihat anggota siswa dan juga anggota guru maupun staff. Terdiri dari tombol untuk menambah dan tombol *export* yang terletak dibawah *heading* tabel anggota. Dan untuk fungsi lihat yang merupakan untuk melihat detail, dan juga mengubah, serta menghapus terdapat di dalam tabel. Berikut merupakan rancangan *interface* (admin) halaman anggota iterasi-2 dapat dilihat pada gambar 4.59 :

SIPS SMKN11		User				
User	Sekolah	Daftar Anggota Siswa		Daftar Anggota Staff		
		+	Export			
		No.	Nama	Nama Pendek	Nomor Anggota	Aksi
		1.	Mila Sari	Mila	217	Ubah Hapus

Gambar 4.59 Halaman anggota (Admin) iterasi-2

7. Rancangan interface halaman transaksi iterasi-2

Halaman transaksi terdiri dari mesin pencari yang terletak dibawah heading tabel, kemudian tombol tambah yang terletak di bawah mesin pencari. Dan untuk tabel menampilkan id, nomor anggota, nama anggota, judul buku, tanggal pinjam, tanggal kembali, status, dan juga fungsi mengubah, serta mengembalikan buku.

Berikut merupakan rancangan *interface* (admin) halaman transaksi iterasi-2 dapat dilihat pada gambar 4.60 :

Daftar Transaksi							
Cari Id							
Id	No. Anggota	Nama Anggota	Judul Buku	Tgl Pinjam	Tgl Kembali	Status	Aksi
1	332	Anisa	Sejarah	02/07/2019	10/07/2019	Pinjam	Ubah Kembali

Copyright 2019. Sistem Informasi Perpustakaan SMKN 11 Malang

Gambar 4.60 Halaman transaksi (Admin) iterasi-2

8. Rancangan *inteface* halaman buku iterasi-2

Halaman buku terdapat mesin pencari yang terletak di bawah *heading* tabel, tombol tambah di bawah mesin pencari, dan di dalam tabel terdapat kode buku, judul buku, penulis, tahun, penerbit, isbn, total buku dan juga rak. Sedangkan untuk fungsi melihat detail mengubah, serta menghapus terletak disamping total buku. Berikut merupakan rancangan *interface* (admin) halaman buku iterasi-2 dapat dilihat pada gambar 4.61:

Daftar Buku									
Search..									
Eksport									
No	Kode Buku	Judul Buku	Penulis	Tahun	Penerbit	ISBN	Total Buku	Rak	Aksi
1	f546h7	Sejarah	Andi	2017	KEMDIKBUD	12453627	2C	RPL	Lihat Ubah Hapus

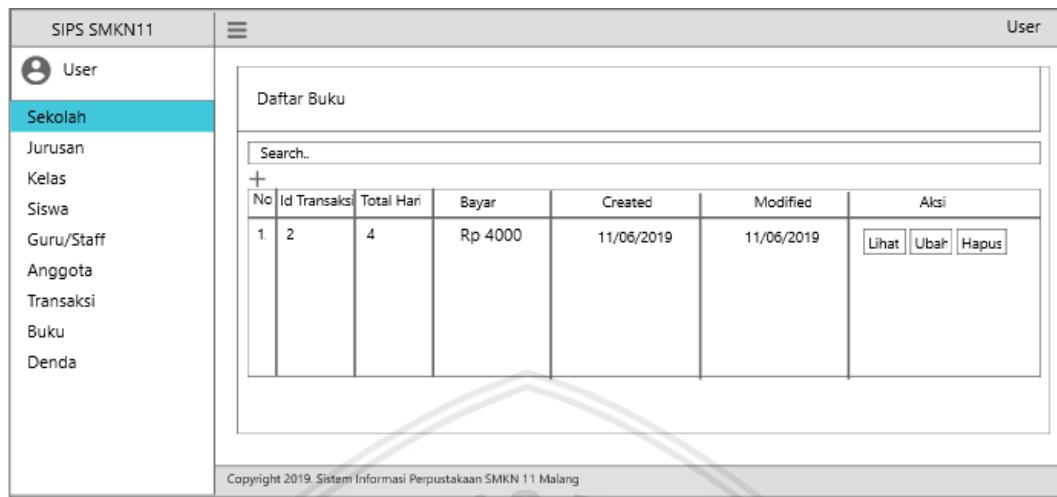
Copyright 2019. Sistem Informasi Perpustakaan SMKN 11 Malang

Gambar 4.61 Halaman buku (Admin) iterasi-2

9. Rancangan *inteface* halaman denda iterasi-2

Halaman denda menampilkan Id transaksi, total hari, bayar, dan tanggal dibuat. Serta terdapat fungsi untuk melihat, mengubah, dan menghapus. Sedangkan untuk tombol menambah terdapat di bawah heading tabel. Berikut merupakan

rancangan *interface* (admin) halaman denda iterasi-2 dapat diihat pada gambar 4.62 :



Gambar 4.62 Halaman denda (Admin) iterasi-2

4.3 Implementasi

Pada tahap implementasi merupakan tahap dimana setelah aspek-aspek pada workshop desain disepakati, sistem dibangun dan disempurnakan, sistem baru atau bagian dari sistem diuji dan kemudian diperkenalkan ke organisasi.

4.3.1 Implementasi sistem

Implementasi sistem terdiri dari implementasi database, implementasi *source code*, implementasi halaman admin, implementasi halaman anggota perpustakaan. Implementasi sistem berdasarkan iterasi-2 yang merupakan hasil akhir dari rancangan yang telah disepakati:

4.3.1.1. Implementasi database

Tabel 4.23 DDL tabel buku

```
CREATE TABLE books (
    id          int(10) NOT NULL AUTO_INCREMENT,
    book_code   varchar(10) NOT NULL UNIQUE,
    title       varchar(255) NOT NULL,
    author      varchar(255) NOT NULL,
    year_edition int(10) NOT NULL,
    publisher   varchar(255) NOT NULL,
    isbn        varchar(20) NOT NULL UNIQUE,
    total_book  int(10),
    rack        varchar(12) NOT NULL,
    created     datetime NOT NULL,
    modified    datetime NOT NULL,
    PRIMARY KEY (id),
    UNIQUE INDEX (id));
```

Tabel 4.24 DDL tabel jurusan

```
CREATE TABLE competences (
    id          int(10) NOT NULL AUTO_INCREMENT,
    school_id   int(10) NOT NULL,
    name        varchar(255),
    description varchar(255),
    created     datetime NOT NULL,
    modified    datetime NOT NULL,
    PRIMARY KEY (id),
    UNIQUE INDEX (id));
```

Tabel 4.25 DDL tabel denda

```
CREATE TABLE fine (
    id          int(10) NOT NULL AUTO_INCREMENT,
    transaction_id int(10) NOT NULL,
    total_day    int(10),
    pay          float,
    created      datetime NOT NULL,
    modified     datetime NOT NULL,
    PRIMARY KEY (id),
    UNIQUE INDEX (id));
```

Tabel 4.26 DDL tabel kelas

```
CREATE TABLE groups (
    id          int(10) NOT NULL AUTO_INCREMENT,
    competence_id int(10) NOT NULL,
    name        varchar(255) NOT NULL UNIQUE,
    description varchar(255),
    created     datetime NOT NULL,
    modified    datetime NOT NULL,
    PRIMARY KEY (id),
    UNIQUE INDEX (id));
```

Tabel 4.27 DDL tabel anggota

```
CREATE TABLE members (
    id          int(10) NOT NULL AUTO_INCREMENT,
    student_id int(10) NOT NULL,
    staff_id   int(10) NOT NULL,
    name        varchar(60) NOT NULL,
    number      int(10) NOT NULL,
    created     datetime NOT NULL,
    modified    datetime NOT NULL,
    PRIMARY KEY (id),
    UNIQUE INDEX (id),
    UNIQUE INDEX (number));
```

Tabel 4.28 DDL tabel sekolah

```
CREATE TABLE schools (
    id          int(10) NOT NULL AUTO_INCREMENT,
    name        varchar(100) NOT NULL,
    address     varchar(255) NOT NULL,
    phone       varchar(20),
    headmaster_nme  varchar(50) NOT NULL,
    headmaster_phone varchar(20) NOT NULL,
    type        enum('negeri','swasta') NOT NULL,
    created      datetime NOT NULL,
    modified     datetime NOT NULL,
    PRIMARY KEY (id),
    UNIQUE INDEX (id),
    UNIQUE INDEX (name),
    INDEX (type));
```

Tabel 4.29 DDL tabel staff

```
CREATE TABLE staffs (
    id          int(10) NOT NULL AUTO_INCREMENT,
    nip         varchar(60) NOT NULL,
    name        varchar(50) NOT NULL,
    address     varchar(255),
    phone       varchar(20),
    created      datetime NOT NULL,
    modified     datetime NOT NULL,
    PRIMARY KEY (id),
    UNIQUE INDEX (id),
    UNIQUE INDEX (nip));
```

Tabel 4.30 DDL tabel siswa

```
CREATE TABLE students (
    id          int(10) NOT NULL AUTO_INCREMENT,
    nis         int(10) NOT NULL,
    group_id    int(10) NOT NULL,
    name        varchar(50) NOT NULL,
    address     varchar(255),
    placebirth  varchar(25) NOT NULL,
    datebirth   date NOT NULL,
    phone       varchar(20),
    created      datetime NOT NULL,
    modified     datetime NOT NULL,
    PRIMARY KEY (id),
    UNIQUE INDEX (id),
    UNIQUE INDEX (nis));
```

Tabel 4.31 DDL tabel transaksi

```
CREATE TABLE transactions (
    id          int(10) NOT NULL AUTO_INCREMENT,
    number      int(10) NOT NULL,
    member_id   int(10) NOT NULL,
    book_id     int(10) NOT NULL,
    loan_date   date NOT NULL,
    date_of_return date NOT NULL,
    status      varchar(60) NOT NULL,
    created     datetime NOT NULL,
    modified    datetime NOT NULL,
    PRIMARY KEY (id),
    UNIQUE INDEX (id));

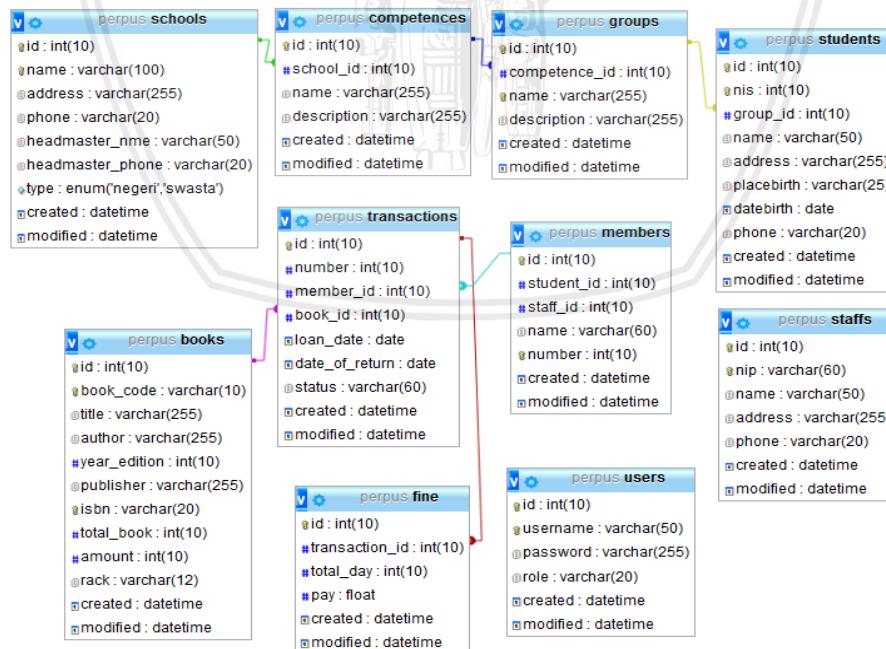
```

Tabel 4.32 DDL tabel user

```
CREATE TABLE users (
    id          int(10) NOT NULL AUTO_INCREMENT,
    username    varchar(50) NOT NULL,
    password    varchar(255) NOT NULL,
    role        varchar(20) NOT NULL,
    created     datetime NOT NULL,
    modified    datetime NOT NULL,
    PRIMARY KEY (id),
    UNIQUE INDEX (id),
    UNIQUE INDEX (username));

```

Berikut implementasi database dapat dilihat pada gambar 4.63 :

**Gambar 4.63 Implementasi database**

4.3.1.2. Implementasi *source code*

a. *Source code* anggota

Source code pada menu anggota terdiri dua tab yang terdiri dari anggota siswa dan juga anggota staff. Pada baris ke-1 merupakan tab yang didalamnya berisi anggota siswa. Baris ke-2, tombol untuk menambahkan anggota siswa berupa tombol button yang memanggil fungsi *add* pada *controller*. Baris ke-3, tombol untuk mengeksport data anggota ke dalam bentuk excel. Baris ke-4, fungsi perulangan yang berupa *foreach* dimana akan menampilkan semua isi *array* (kumpulan beberapa data yang disimpan) dengan perintah yang lebih singkat. Baris ke-5, perulangan *if* yang merupakan suatu kondisi jika tidak kosong variabel *member(anggota)* yang berisi *id* dari siswa akan menampilkan data siswa. Baris ke-6 *source code* untuk mendefinisikan baris tabel. Baris ke-7, mencetak nomor secara berurutan. Baris ke-8, menampilkan nama siswa yang jika ditekan akan memanggil nama siswa pada menu siswa berdasarkan id yang dipanggil. Baris ke-9, menampilkan nama pendek siswa. Baris ke-10, menampilkan nomor anggota siswa. Baris ke-11, merupakan kolom yang berupa aksi. Baris ke-12, tombol untuk mengubah data siswa yang memanggil fungsi *edit* pada *controller* untuk dieksekusi berdasarkan id siswa yang dipanggil. Baris ke-13, tombol untuk menghapus data siswa yang dikehendaki berdasarkan id yang dipanggil, jika yakin akan dihapus maka sistem akan menampilkan konfirmasi berupa pesan “Apakah Anda yakin ingin menghapus?” jika “ya” sistem akan menghapus jika “tidak” sistem akan kembali ke halaman anggota. Baris ke-14, penutup data tabel. Baris ke-15, penutup baris tabel. Baris ke-16, penutup perulangan *if*. Baris ke-17, penutup perulangan *foreach*.

Berikut implementasi potongan *source code* anggota dapat dilihat pada tabel 4.33:

Tabel 4.33 Potongan *source code* anggota

No.	Members
1	<div id="Siswa" class="tabcontent">
2	<?= \$this->Html->link('<i class="fa fa-plus"></i>', ['action' => 'add/siswa'], ['escape'=>false, 'class'=>'btn btn-info btn-sm']) ?>
3	<?= \$this->Html->link('<i class="fa fa-download"></i> Export', ['action' => 'export'], ['escape'=>false, 'class'=>'btn btn-warning btn-sm btn-social']) ?>
4	<?php \$no=1; foreach (\$students as \$member): ?>
5	<?php if(!empty(\$member->student_id)): ?>
6	<tr>
7	<td><?= \$no++; ?></td>
8	<td><?= \$this->Html->link(\$member->student->name, ['controller' => 'Students', 'action' => 'view', \$member->student->id]); ?></td>
9	<td><?= h(\$member->name) ?></td>
10	<td><?= \$this->Number->format(\$member->number) ?></td>
11	<td class="actions">
12	<?= \$this->Html->link('<i class="fa fa-edit"></i> Ubah', ['action' => 'edit', \$member->id], ['escape'=>false, 'class'=>'btn btn-warning btn-sm btn-social']) ?>

Selanjutnya, merupakan lanjutan potongan *source code* anggota. Pada baris ke-18, merupakan tab yang didalamnya berisi anggota staff. Baris ke-19, merupakan

tombol untuk menambahkan anggota staff berupa tombol button yang memanggil fungsi *add* pada *controller*. Baris ke-20, fungsi perulangan yang berupa *foreach* dimana akan menampilkan semua isi *array* (kumpulan beberapa data yang disimpan) dengan perintah yang lebih singkat. Baris ke-21, perulangan *if* yang merupakan suatu kondisi jika tidak kosong variabel *member*(anggota) yang berisi *id* dari staff akan menampilkan data staff. Baris ke-22 *source code* untuk mendefinisikan baris tabel. Baris ke-23, mencetak nomor secara berurutan. Baris ke-24, menampilkan nama staff yang jika ditekan akan memanggil nama staff pada menu staff berdasarkan id yang dipanggil. Baris ke-25, menampilkan nama pendek staff. Baris ke-26, menampilkan nomor anggota staff. Baris ke-27, merupakan kolom yang berupa aksi. Baris ke-28, tombol untuk mengubah data staff yang memanggil fungsi *edit* pada *controller* untuk dieksekusi berdasarkan id staff yang dipanggil. Baris ke-29, tombol untuk menghapus data staff yang dikehendaki berdasarkan id yang dipanggil, jika yakin akan dihapus maka sistem akan menampilkan konfirmasi berupa pesan “Apakah Anda yakin ingin menghapus?” jika “ya” sistem akan menghapus jika “tidak” sistem akan kembali ke halaman anggota. Baris ke-30, penutup data tabel. Baris ke-31, penutup baris tabel. Baris ke-32, penutup perulangan *if*. Baris ke-33, penutup perulangan *foreach*.

Berikut implementasi potongan *source code* anggota (lanjutan) dapat dilihat pada tabel 4.33:

Tabel 4.33 Potongan source code anggota (lanjutan)

Tabel 4.33 Potongan source code anggota (lanjutan)

No .	Members
27	<td class="actions">
28	<?= \$this->Html->link('<i
29	class="fa fa-edit"></i> Ubah', ['action' => 'editstaff', \$borrower->id], ['escape'=>false,'class'=>'btn btn-warning btn-sm btn-social']) ?>
30	<?= \$this->Form->postLink('<i class="fa fa-trash"></i> Hapus', ['action' => 'delete', \$borrower- >id], ['escape'=>false,'class'=>'btn btn-danger btn-sm btn-social' , 'confirm' => __('Apa Anda yakin ingin menghapus?', \$borrower- >id)) ?>
31	</td>
32	</tr>
33	<?php endif;?> <?php endforeach; ?>

b. *Source code* transaksi

Baris ke-1 perintah untuk membuat form pencarian dengan tipe *get*. Baris ke-2 merupakan class inputan untuk mengatur tata letak form pencarian. Baris ke-3 berisi label dengan *placeholder* “Cari Id ..” untuk memudahkan proses pencarian transaksi. Baris ke-4 merupakan class input button untuk pengaturan desain tombol di dalam form pencarian. Baris ke-5 tombol submit dalam bentuk *search engine*. Baris ke-6 penutup *class* button. Baris ke-7 penutup *class* span. Baris ke-8 penutup *class* input button. Baris ke-9 penutup form pencarian. Baris ke-10 merupakan tombol untuk menambahkan transaksi berupa tombol button yang memanggil fungsi *add* pada *controller*. Baris ke-11 merupakan fungsi perulangan yang berupa *foreach* dimana akan menampilkan semua isi *array* (kumpulan beberapa data yang disimpan) dengan perintah yang lebih singkat. Baris ke-12 kode baris tabel. Baris ke-13 menampilkan id transaksi. Baris ke-14 menampilkan nomor anggota. Baris ke-15 menampilkan nama anggota yang berupa link. Baris ke-16 menampilkan judul buku yang berupa link. Baris ke-17 menampilkan tanggal pinjam. Baris ke-18 menampilkan tanggal kembali. Baris ke-19 menampilkan status meminjam atau terlambat atau hilang. Baris ke-20 merupakan kolom aksi. Baris ke-21 tombol untuk melihat detail transaksi yang berupa modal dimana ketika ditekan akan menampilkan data tanpa harus membuka halaman baru. Baris ke-22 merupakan tombol untuk mengubah data. Baris ke-23 merupakan tombol untuk menghapus data transaksi yang dikehendaki berdasarkan id yang dipanggil, jika yakin akan dihapus maka sistem akan menampilkan konfirmasi berupa pesan “Apakah Anda yakin ingin menghapus?” jika “ya” sistem akan menghapus jika “tidak” sistem akan kembali ke halaman transaksi. Baris ke-24 dan baris ke-25 penutup tabel. Baris ke-26 penutup perulangan *foreach*.

Berikut implementasi potongan *source code* transaksi dapat dilihat pada tabel 4.34:

Tabel 4.34 Potongan source code transaksi

No.	Transactions	
1	<?= \$this->Form->create("", ['type'=>'get']) ?>	
2	<div ><="" class="input-group" td=""></div>	
3	<input type="text" name="keyword" class="form-control" placeholder="Cari Id..."/>	
4		
5	<button type="submit" class="btn btn-flat"><i class="fa fa-search">/</i></button>	
6		
7	</div>	
8	<?= \$this->Form->end() ?>	
9	<?= \$this->Html->link('<i class="fa fa-plus">/</i>', ['action' => 'add'], ['escape'=>false,'class'=>'btn btn-info btn-sm']) ?>	
10	<?php foreach (\$transactions as \$transaction): ?>	
11	<tr>< td=""></tr><>	
12	<td><?= \$this->Number->format(\$transaction->id) ?></td>	<?= \$this->Number->format(\$transaction->id) ?>
13	<td><?= \$this->Number->format(\$transaction->number) ?></td>	<?= \$this->Number->format(\$transaction->number) ?>
14	<td><?= \$transaction->has('member') ? \$this->Html->link(\$transaction->member->name, ['controller' => 'Members', 'action' => 'view_student', \$transaction->member->id]) : '' ?></td>	<?= \$transaction->has('member') ? \$this->Html->link(\$transaction->member->name, ['controller' => 'Members', 'action' => 'view_student', \$transaction->member->id]) : '' ?>
15	<td><?= \$transaction->has('book') ? \$this->Html->link(\$transaction->book->title, ['controller' => 'Books', 'action' => 'view', \$transaction->book->id]) : '' ?></td>	<?= \$transaction->has('book') ? \$this->Html->link(\$transaction->book->title, ['controller' => 'Books', 'action' => 'view', \$transaction->book->id]) : '' ?>
16	<td><?= h(\$transaction->loan_date) ?></td>	<?= h(\$transaction->loan_date) ?>
17		
18	<td><?= h(\$transaction->date_of_return) ?></td>	<?= h(\$transaction->date_of_return) ?>
19	<td><?= h(\$transaction->status) ?></td>	<?= h(\$transaction->status) ?>
20	<td><td class="actions"></td>	<td class="actions">
21	<?= \$this->Html->link('<i class="fa fa-eye">/</i> Lihat', ['action' => 'view', \$transaction->id], ['escape'=>false,'class'=>'btn btn-info btn-sm btn-social', 'data-toggle'=>'modal','data-target'=>'#perpus-modal']) ?>	
22	<?= \$this->Html->link('<i class="fa fa-edit">/</i> Ubah', ['action' => 'edit', \$transaction->id], ['escape'=>false,'class'=>'btn btn-warning btn-sm btn-social']) ?>	
23	<?= \$this->Form->postLink('<i class="fa fa-trash">/</i> Hapus', ['action' => 'delete', \$transaction->id], ['escape'=>false,'class'=>'btn btn-danger btn-sm btn-social', 'confirm' => __('Apa Anda yakin ingin menghapus?', \$transaction->id)]) ?>	
24	</td>	
25	</td>	
26	<?php endforeach; ?>	

c. Source code buku

Baris ke-1 perintah untuk membuat form pencarian dengan tipe *get*. Baris ke-2 merupakan class inputan untuk mengatur tata letak form pencarian. Baris ke-3 berisi label dengan *placeholder* “Ketik jurusan..” untuk memudahkan proses pencarian transaksi. Baris ke-4 merupakan class input button untuk pengaturan desain tombol di dalam form pencarian. Baris ke-5 tombol submit dalam bentuk

search engine. Baris ke-6 penutup *class* button. Baris ke-7 penutup *class* span. Baris ke-8 penutup *class* input button. Baris ke-9 penutup form pencarian. Baris ke-11 tombol untuk menambahkan data buku berupa tombol button yang memanggil fungsi *add* pada *controller*. Baris ke-12, tombol untuk mengeksport data buku ke dalam bentuk excel. Baris ke-14 merupakan fungsi perulangan yang berupa *foreach* dimana akan menampilkan semua isi *array* (kumpulan beberapa data yang disimpan) dengan perintah yang lebih singkat. Baris ke-15 menampilkan baris tabel. Baris ke-16 menampilkan nomor berdasarkan urutan. Baris ke-17 menampilkan kode buku. Baris ke-18 menampilkan judul buku. Baris ke-19 menampilkan penulis buku. Baris ke-20 menampilkan tahun edisi. Baris ke-21 menampilkan penerbit buku. Baris ke-22 menampilkan nomor ISBN. Baris ke-23 menampilkan total buku yang tersedia. Baris ke-24 menampilkan rak buku berdasarkan jurusan. Baris ke-25 menampilkan kolom aksi. Baris ke-26 tombol untuk melihat detail buku yang berupa modal dimana ketika ditekan akan menampilkan data tanpa harus membuka halaman baru. Baris ke-27 merupakan tombol untuk mengubah data. Baris ke-28 merupakan tombol untuk menghapus data buku yang dikehendaki berdasarkan id yang dipanggil, jika yakin akan dihapus maka sistem akan menampilkan konfirmasi berupa pesan "Apakah Anda yakin ingin menghapus?" jika "ya" sistem akan menghapus jika "tidak" sistem akan kembali ke halaman buku. Baris ke-29 dan baris ke-30 penutup tabel. Baris ke-31 penutup perulangan *foreach*.

Berikut implementasi potongan *source code* buku dapat dilihat pada tabel 4.35:

Tabel 4.35 Potongan source code buku

No.	Books
1	<?= \$this->Form->create("", ['type'=>'get']) ?>
2	<div class="input-group">
3	<input type="text" name="keyword" class="form-control" placeholder="Ketik jurusan...">
4	
5	<button type="submit" class="btn btn-flat"><i class="fa fa-search"></i></button>
6	
7	</div>
8	<?= \$this->Form->end() ?>
9	
10	
11	<?= \$this->Html->link('<i class="fa fa-plus"></i> ', ['action' => 'add'], ['escape'=>false, 'class'=>'btn btn-info btn-sm']) ?>
12	<?= \$this->Html->link('<i class="fa fa-download"></i> Export', ['action' => 'export'], ['escape'=>false, 'class'=>'btn btn-warning btn-sm btn-social']) ?>
13	
14	<?php \$no=1; foreach (\$books as \$book): ?>
15	<tr>
16	<td><?= \$no++; ?></td>
17	<td><?= h(\$book->book_code) ?></td>
18	<td><?= h(\$book->title) ?></td>
19	<td><?= h(\$book->author) ?></td>
20	<td><?= h(\$book->year_edition) ?></td>
21	<td><?= h(\$book->publisher) ?></td>
22	<td><?= h(\$book->isbn) ?></td>
23	<td><?= \$this->Number->format(\$book->total_book) ?></td>

Tabel 4.35 Potongan source code buku (lanjutan)

No.	Books
24	<td><?= h(\$book->rack) ?></td>
25	<td class="actions">
26	<?= \$this->Html->link('<i class="fa fa-eye"></i> Lihat', ['action' => 'view', \$book->id], ['escape'=>false,'class'=>'btn btn-info btn-sm btn-social', 'data-toggle'=>'modal', 'data-target'=> '#perpus-modal']) ?>
27	<?= \$this->Html->link('<i class="fa fa-edit"></i> Ubah', ['action' => 'edit', \$book->id], ['escape'=>false,'class'=>'btn btn-warning btn-sm btn-social']) ?>
28	<?= \$this->Form->postLink('<i class="fa fa-trash"></i> Hapus', ['action' => 'delete', \$book->id], ['escape'=>false,'class'=>'btn btn-danger btn-sm btn-social', 'confirm' => __('Apa Anda yakin ingin menghapus?', \$book->id)]) ?>
29	</td>
30	</tr>
31	<?php endforeach; ?>

d. Source code denda

Baris ke-1 tombol untuk menambahkan data denda berupa tombol button yang memanggil fungsi *add* pada *controller*. Baris ke-3 merupakan fungsi perulangan yang berupa *foreach* dimana akan menampilkan semua isi *array* (kumpulan beberapa data yang disimpan) dengan perintah yang lebih singkat. Baris ke-4 menampilkan baris tabel. Baris ke-5 menampilkan nomor berdasarkan urutan. Baris ke-6 menampilkan id transaksi yang terlambat mengembalikan. Baris ke-7 menampilkan total hari terlambat. Baris ke-8 menampilkan denda yang harus dibayar. Baris ke-9 menampilkan tanggal dibuat. Baris ke-10 menampilkan tanggal modifikasi. Baris ke-11 menampilkan kolom aksi. Baris ke-12 tombol untuk melihat detail denda yang berupa modal dimana ketika ditekan akan menampilkan data tanpa harus membuka halaman baru. Baris ke-13 merupakan tombol untuk mengubah data. Baris ke-14 merupakan tombol untuk menghapus data denda yang dikehendaki berdasarkan id yang dipanggil, jika yakin akan dihapus maka sistem akan menampilkan konfirmasi berupa pesan “Apakah Anda yakin ingin menghapus?” jika “ya” sistem akan menghapus jika “tidak” sistem akan kembali ke halaman denda. Baris ke-15 dan baris ke-16 penutup tabel. Baris ke-17 penutup perulangan *foreach*.

Berikut implementasi potongan *source code* denda dapat dilihat pada tabel 4.36:

Tabel 4.36 Potongan source code denda

No.	Fine
1	<?= \$this->Html->link('<i class="fa fa-plus"></i>', ['action' => 'add'], ['escape'=>false,'class'=>'btn btn-info btn-sm']) ?>
2	
3	<?php \$no=1; foreach (\$fine as \$fine): ?>
4	<tr>
5	<td><?= \$no++; ?></td>
6	<td><?= \$fine-
7	>has('transaction') ? \$this->Html->link(\$fine->transaction->id,
8	['controller' => 'Transactions', 'action' => 'view', \$fine-
9	>transaction->id]) : '' ?></td>
10	<td><?= \$this->Number-
11	>format(\$fine->total_day) ?></td>
12	<td><?= Rp. ,
13	h(\$fine->pay) ?></td>
14	<td><?= h(\$fine->created) ?></td>
15	<td><?= h(\$fine->modified) ?></td>
16	<td><?= class="actions">
17	<?= \$this->Html->link('<i class="fa fa-eye"></i> Lihat', ['action' => 'view', \$fine->id], ['escape'=>false,'class'=>'btn btn-info btn-sm btn-social', 'data-toggle'=>'modal', 'data-target'=>'#perpus-modal']) ?>
	<?= \$this->Html->link('<i class="fa fa-edit"></i> Ubah', ['action' => 'edit', \$fine->id], ['escape'=>false,'class'=>'btn btn-warning btn-sm btn-social']) ?>
	<?= \$this->Form->postLink('<i class="fa fa-trash"></i> Hapus', ['action' => 'delete', \$fine->id], ['escape'=>false,'class'=>'btn btn-danger btn-sm btn-social', 'confirm' => __("Apa Anda yakin ingin menghapus?", \$fine->id)]) ?>
	</td>
	</tr>
	<?php endforeach; ?>

e. Source code login

Baris ke-1 merupakan perintah form login dibuat. Baris ke-2 tag fieldset untuk mengelompokkan beberapa objek form menjadi sebuah kelompok. Baris ke-3 form untuk inputan username. Baris ke-4 form untuk inputan pasword. Baris ke-5 penutup tag fieldset. Baris ke-6 form untuk tombol buton login. Baris ke-7 merupakan bagian akhir form login.

Berikut implementasi potongan source code login dapat dilihat pada tabel 4.37:

Tabel 4.37 Potongan source code login

No.	login.ctp
1	<?= \$this->Form->create() ?>
2	<fieldset>
3	<?= \$this->Form->control(['username'], ['placeholder'=>'Username..', 'required'=>'required']) ?>
4	<?= \$this->Form->control(['password'], ['placeholder'=>'Password..', 'required'=>'required']) ?>
5	</fieldset>
6	<center><?= \$this->Form->button(__('Login')) ; ?></center>
7	<?= \$this->Form->end() ?>

f. Source code registrasi

Baris ke-1 merupakan perintah form dibuat. Baris ke-2 tag untuk style tabel. Baris ke-3 form inputan untuk username. Baris ke-4 form inputan untuk password. Baris ke-5 form untuk role user untuk memilih jenis user siswa, guru atau staff. Baris ke-6 penutup tag tabel. Baris ke-7 merupakan tombol submit. Baris ke-8 merupakan bagian akhir form.

Berikut implementasi potongan *source code* registrasi dapat dilihat pada tabel 4.38:

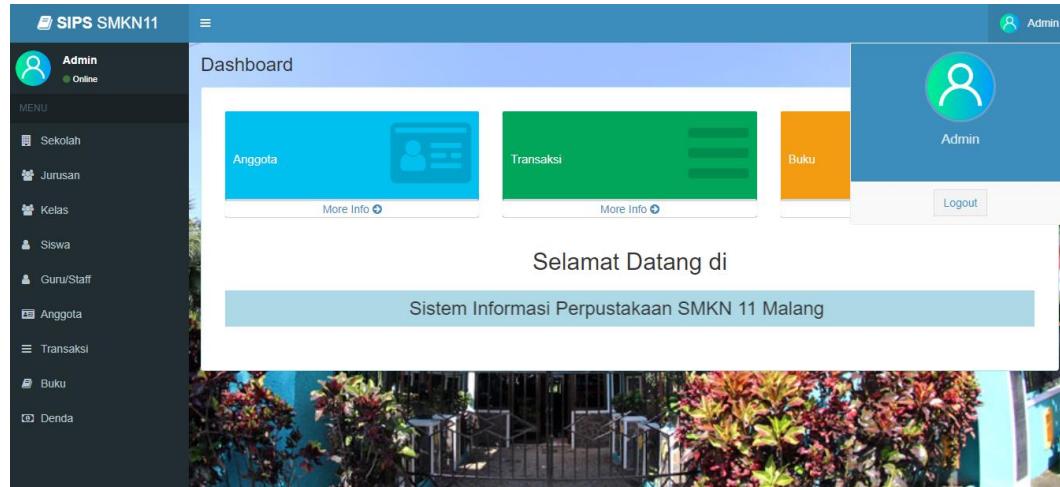
Tabel 4.38 Potongan source code registrasi

No.	add.ctp
1	<?= \$this->Form->create(\$user) ?>
2	<table class="table table-hover">
3	<?= \$this->Form->control('username', ['placeholder'=>'Username..', 'required'=>'required']) ?>
4	<?= \$this->Form->control('password', ['placeholder'=>'Password..', 'required'=>'required']) ?>
5	<?= \$this->Form->control('role', [
6	'options' => ['empty' => 'Pilih salah satu.', 'students' => 'Siswa', 'staffs' => 'Guru/Staff']
7]) ?>
8	</table>
	<?= \$this->Form->button(__('Submit')) ; ?>
	<?= \$this->Form->end() ?>

4.3.1.3. Implementasi halaman admin

a. Halaman dashboard

Berikut merupakan implementasi halaman dashboard dapat dilihat pada gambar 4.64:



Gambar 4.64 Implementasi halaman dashboard

b. Halaman anggota

Berikut merupakan implementasi halaman anggota dapat dilihat pada gambar 4.65:

No.	Nama	Nama Pendek	Nomor Anggota	Aksi
1	Mila Sari	Milasari	678	<input checked="" type="checkbox"/> Ubah <input type="checkbox"/> Hapus
2	Muhammad Iqbal	Iqbal	324	<input checked="" type="checkbox"/> Ubah <input type="checkbox"/> Hapus
3	Anisa Rizky	anisa	546	<input checked="" type="checkbox"/> Ubah <input type="checkbox"/> Hapus

Gambar 4.65 Implementasi halaman anggota

c. Halaman transaksi

Berikut merupakan implementasi halaman transaksi dapat dilihat pada gambar 4.66:

ID	Nomor Anggota	Nama Anggota	Judul Buku	Tanggal Pinjam	Tanggal Kembali	Status
7	221	Dodik	Sejarah Indonesia X	04/07/2019	07/07/2019	Pinjam (Belum terlambat)
6	557	Lilya	MATEMATIKA XI	03/07/2019	04/07/2019	Pinjam (Belum terlambat)
8	221	Dodik	MATEMATIKA XI	04/07/2019	12/07/2019	Pinjam (Belum terlambat)

Gambar 4.66 Implementasi halaman transaksi

d. Halaman buku

Berikut merupakan implementasi halaman buku dapat dilihat pada gambar 4.67:

No.	Kode Buku	Judul Buku	Penulis	Tahun	Penerbit	ISBN	Total Buku	Rak	Aksi
1	df87kr	Sejarah Indonesia X	Amurwani Dwi	2017	KEMDIKBUD	978-602-8432-97-9	20	ANM	
2	s36fkq	Pendidikan Agama	Eindi Suhendi	2017	KEMDIKBUD	978-552-8520-97-1	15	TKJ	
3	dg47ky	MATEMATIKA XI	Andri Kristianto	2017	KEMDIKBUD	928-552-887-97-2	10	ANM	
4	dc56pt	Bahasa Inggris X	Utami Widiatni	2017	KEMDIKBUD	935-367-887-97-4	20	TKR	
5	df80kr	Bahasa Arab	gggggg	2018	yyyy	2344566	6	RPL	

Gambar 4.67 Implementasi halaman buku

e. Halaman denda

Berikut merupakan implementasi halaman denda dapat dilihat pada gambar 4.68:

No.	Id Transaksi	Total Hari	Bayar	Created	Modified	Aksi
1	1	4	Rp. 4000	27/05/2019 03:56:35	27/05/2019 03:56:35	

< previous next >

Page 1 of 1, showing 1 record(s) out of 1 total

Gambar 4.68 Implementasi halaman denda

4.3.1.4. Implementasi halaman user

a. Halaman depan

Berikut merupakan implementasi halaman depan dapat dilihat pada gambar 4.69:

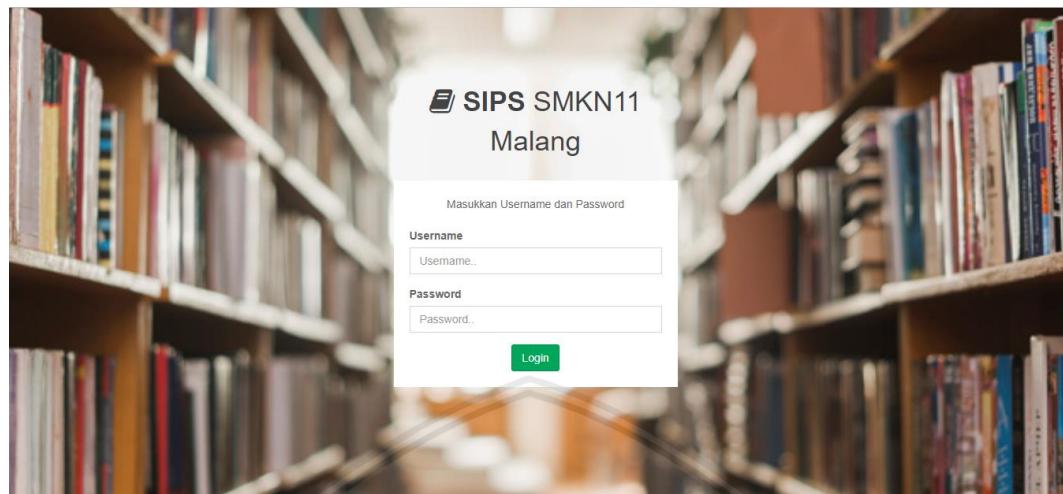
SELAMAT DATANG
SISTEM INFORMASI PERPUSTAKAAN SMK NEGERI 11
MALANG
[Register](#) [Login](#)

Jadwal Buka Perpustakaan
SMKN 11 MALANG
Senin s/d Kamis jam 07:00 WIB - 15:30 WIB
Istirahat jam 12:25 WIB - 13:00 WIB
Jum'at jam 07:00 WIB - 11:00 WIB

Gambar 4.69 Implementasi halaman depan

b. Halaman login

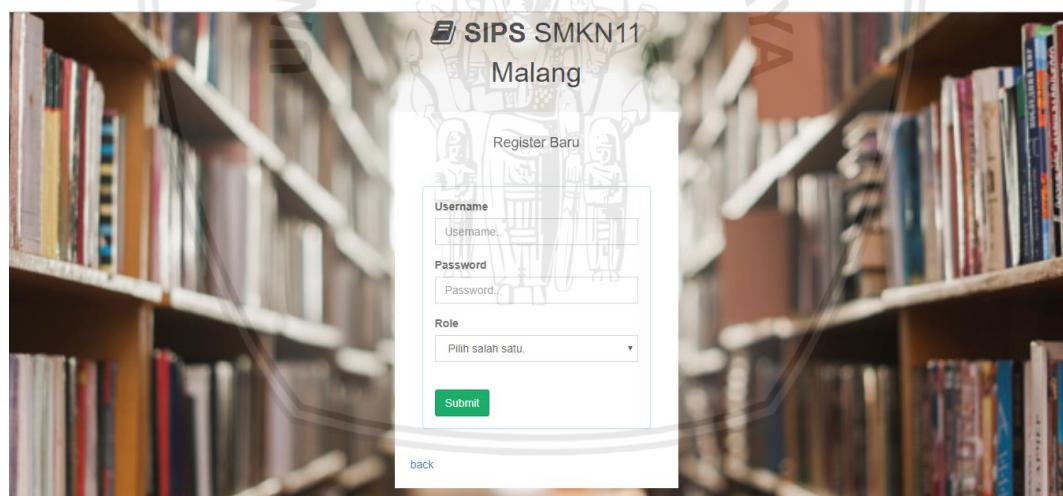
Berikut merupakan implementasi halaman login dapat dilihat pada gambar 4.70:



Gambar 4.70 Implementasi halaman login

c. Halaman register

Berikut merupakan implementasi halaman register dapat dilihat pada gambar 4.71:



Gambar 4.71 Implementasi halaman register

b. Halaman dashboard user

Berikut merupakan implementasi halaman user dapat dilihat pada gambar 4.72:



Gambar 4.72 Implementasi halaman dashboard user

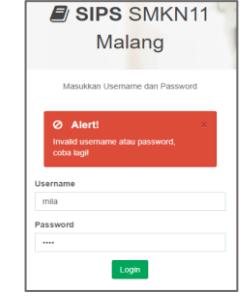
4.3.2 Pengujian Sistem

Pengujian menggunakan teknik pengujian *Black Box*, yaitu pengujian pada aspek fungsionalitas sistem yang memungkinkan untuk mendapatkan serangkaian kondisi masukan dari user yang sepenuhnya akan menjalankan semua persyaratan fungsional untuk suatu program. Selain itu pengujian juga dilakukan menggunakan *User Acceptance Testing* (UAT) yang merupakan proses pengujian oleh *user* dan menghasilkan keluaran dokumen yang bertujuan untuk bukti bahwa sistem yang dibangun dapat diterima *user* dan hasil dari pengujianya dianggap memenuhi kebutuhan pengguna.

4.3.2.1 Pengujian *Black Box*

Berikut hasil pengujian *Black Box* dapat dilihat pada tabel 4.45:

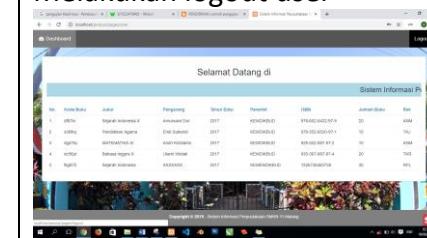
Tabel 4.39 Hasil Uji Black Box

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
SIPS-1-01	Login	Username, Password	Login hanya mengisi username tanpa mengisi password 	Sistem akan menampilkan “please fill out this field”	Sistem menampilkan “please fill out this field”  Valid
			Login dengan mengisi password yang salah 	Sistem akan menampilkan alert “Invalid username atau password, coba lagi!”	Sistem menampilkan alert “Invalid username atau password, coba lagi!”  Valid

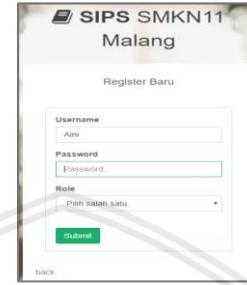
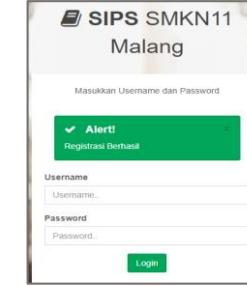
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
			Login user dengan mengisi username dan mengisi password yang benar 	Sistem akan masuk ke halaman dashboard user	<p>Sistem masuk ke halaman dashboard user</p>  <p>Valid</p>
			Login admin dengan mengisi username dan mengisi password yang benar 	Sistem akan masuk ke halaman dashboard admin	<p>Sistem masuk ke halaman dashboard admin</p>  <p>Valid</p>

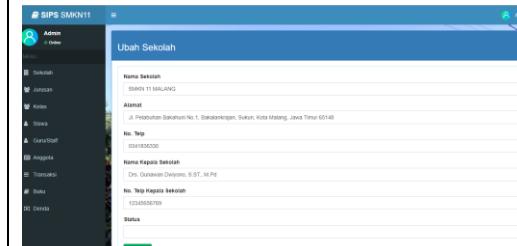
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
SIPS-1-02	Logout		Melakukan logout user	Sistem akan menuju ke halaman login	  Valid
			Melakukan logout admin	Sistem akan menuju ke halaman login	  Valid

Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
SIPS-1-03	Register	Username, Password, Role	Registrasi hanya dengan mengisi username saja 	Sistem akan menampilkan "please fill out this field"	Sistem akan menampilkan "please fill out this field"  Valid
			Registrasi dengan mengisi username, password, dan role. 	Sistem akan menampilkan alert "registrasi berhasil"	Sistem menampilkan alert "registrasi berhasil"  Valid

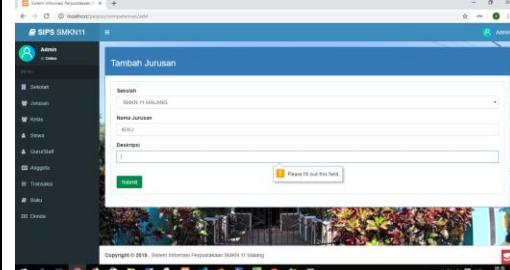
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
SIPS-1-04	Pengelolaan data sekolah		Melihat detail sekolah dengan menekan tombol lihat	Sistem akan menampilkan detail data sekolah	  <p>Valid</p>
			Mengubah data sekolah dengan menekan tombol ubah	Sistem akan menampilkan halaman ubah data sekolah	  <p>Valid</p>

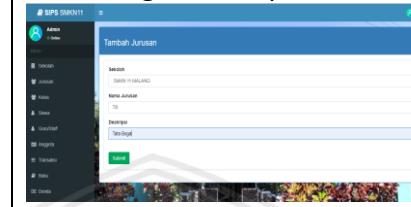
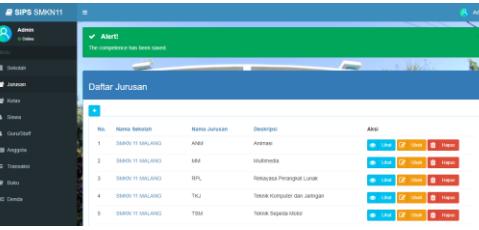
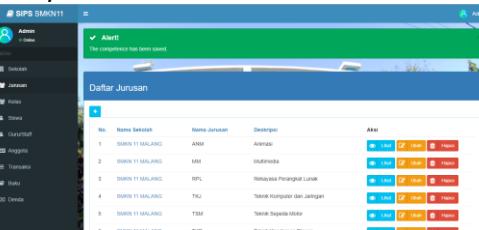
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
		-	Mengubah data sekolah dan menekan tombol simpan	Sistem akan menyimpan data yang sudah diubah dan menampilkan alert <i>"The school has been saved"</i>	 <p style="text-align: center;">Valid</p>
SIPS-1-05	Pengelolaan data jurusan		Melihat detail jurusan dengan menekan tombol lihat	Sistem akan menampilkan detail data jurusan	 <p style="text-align: center;">Valid</p>

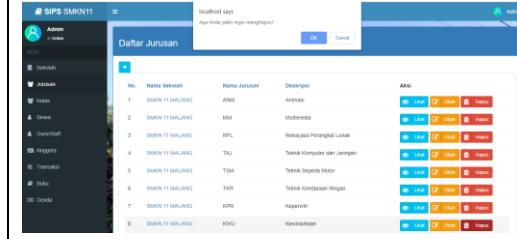
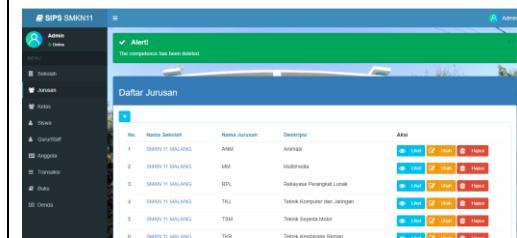
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
Nama jurusan, deskripsi			Menambah data jurusan dengan menekan tombol tambah pada pojok kiri atas	Sistem akan menampilkan halaman tambah jurusan	<p>Sistem menampilkan halaman tambah jurusan</p>  <p>Valid</p>
			Menambah data jurusan dengan mengisi nama jurusan dan mengosongkan deskripsi	Sistem akan menampilkan “please fill out this field”	<p>Sistem menampilkan “please fill out this field”</p>  <p>Valid</p>

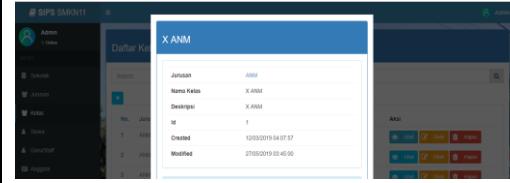
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
		Nama jurusan, deskripsi	Menambah data jurusan dengan mengisi nama jurusan dan mengisi deskripsi 	Sistem akan menyimpan data jurusan dan menampilkan alert " <i>The competences has been saved</i> "	 Valid
			Mengubah data jurusan dengan menekan tombol ubah 	Sistem akan menyimpan data yang sudah diubah dan menampilkan alert " <i>The competences has been saved</i> "	 Valid

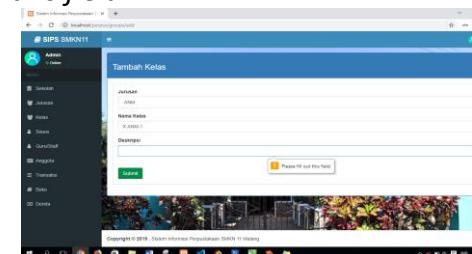
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
		Menghapus data jurusan yang dikehendaki dengan menekan tombol hapus		Sistem akan menampilkan alert “Apa Anda yakin ingin menghapus?”, jika memilih “Ok” sistem akan menampilkan alert <i>“The competence has been deleted”</i>	 Valid jika memilih “Ok” sistem akan menampilkan alert <i>“The competence has been deleted”</i>  Valid

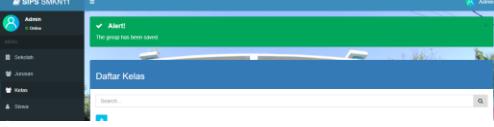
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
SIPS-1-06	Pengelolaan data kelas	-	Melihat detail kelas dengan menekan tombol lihat	Sistem akan menampilkan detail data kelas	<p>Sistem menampilkan detail data kelas</p>  <p style="text-align: center;">Valid</p>
	Nama kelas, deskripsi		Menambah data kelas dengan menekan tombol tambah pada pojok kiri atas	Sistem akan menampilkan halaman tambah kelas	<p>Sistem menampilkan halaman tambah kelas</p>  <p style="text-align: center;">Valid</p>

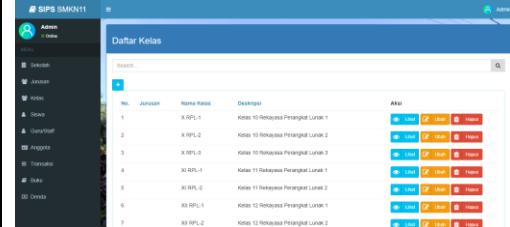
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
	Nama kelas, deskripsi		Menambah data kelas dengan mengisi nama kelas dan mengosongkan deskripsi	Sistem akan menampilkan “please fill out this field”	  Valid
			Menambah data kelas dengan mengisi nama kelas dan mengisi deskripsi	Sistem akan menyimpan data kelas dan menampilkan alert “The group has been saved”	  Valid

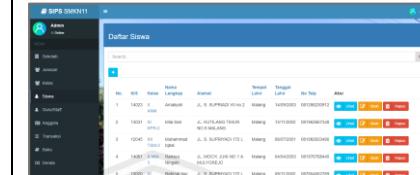
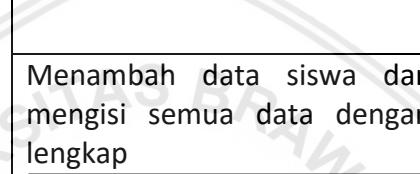
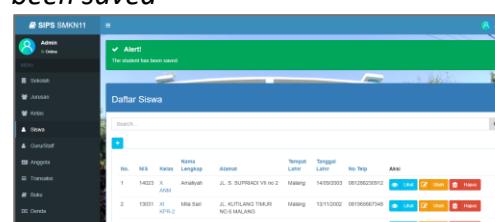
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
		-	Mengubah data kelas dengan menekan tombol ubah 	Sistem akan menyimpan data yang sudah diubah dan menampilkan alert " <i>The group has been saved</i> "	Sistem menyimpan data yang sudah diubah dan menampilkan alert " <i>The group has been saved</i> "  Valid
		-	Menghapus data kelas yang dikehendaki dengan menekan tombol hapus 	Sistem akan menampilkan alert "Apa Anda yakin ingin menghapus?", jika memilih "Ok" sistem akan menampilkan alert " <i>The group has been deleted</i> "	Sistem akan menampilkan alert "Apa Anda yakin ingin menghapus?"  Valid jika memilih "Ok" sistem akan menampilkan alert " <i>The group has been deleted</i> "  Valid

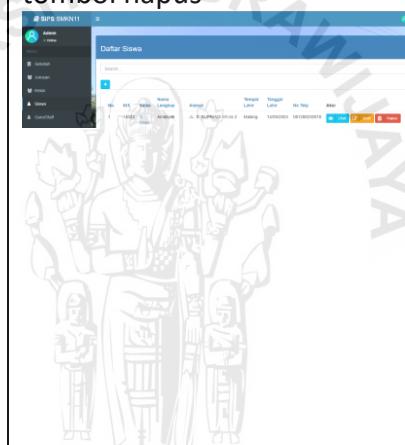
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
SIPS-1-07	Pencarian kelas	Nama Kelas	Menginputkan nama kelas rpl pada form pencarian	Sistem akan menampilkan daftar kelas rpl dari kelas X hingga kelas XII	 <p>Valid</p>
SIPS-1-08	Pengelolaan data siswa		Melihat detail siswa dengan menekan tombol lihat	Sistem menampilkan detail data siswa	 <p>Valid</p>

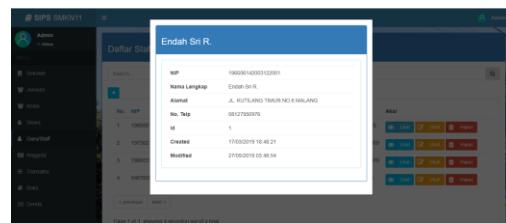
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
	NIS, kelas, nama lengkap, alamat, tempat lahir, tanggal lahir, nomor telp.	Menambah data siswa dengan menekan tombol tambah pada pojok kiri atas	 	Sistem akan menampilkan halaman tambah siswa	 <p>Valid</p>
				Sistem akan menyimpan data siswa dan menampilkan alert "The student has been saved"	 <p>Valid</p>

Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
		-	Mengubah data siswa dengan menekan tombol ubah 	Sistem akan menyimpan data yang sudah diubah dan menampilkan alert " <i>The student has been saved</i> "	Sistem menyimpan data yang sudah diubah dan menampilkan alert " <i>The student has been saved</i> " 
		-	Menghapus data siswa yang dikehendaki dengan menekan tombol hapus 	Sistem akan menampilkan alert "Apa Anda yakin ingin menghapus?", jika memilih "Ok" sistem akan menampilkan alert " <i>The student has been deleted</i> "	Sistem akan menampilkan alert "Apa Anda yakin ingin menghapus?" Valid jika memilih "Ok" sistem akan menampilkan alert " <i>The student has been deleted</i> "  Valid

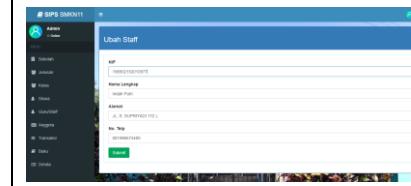
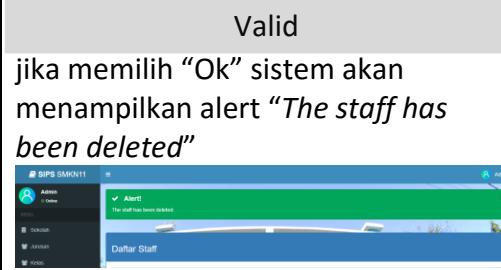
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
SIPS-1-09	Pencarian siswa	Nama Siswa	Menginputkan nama siswa=Ra pada form pencarian	Sistem akan menampilkan daftar nama siswa dengan huruf awal Ra	 <p>Valid</p>
SIPS-1-10	Pengelolaan data staff		Melihat detail staff dengan menekan tombol lihat	Sistem menampilkan detail data staff	 <p>Valid</p>

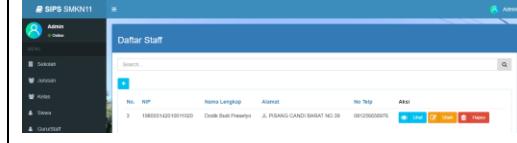
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
		NIP, nama lengkap, alamat, nomor telp.	<p>Menambah data staff dengan menekan tombol tambah pada pojok kiri atas</p>  <p>Menambah data staff dan mengisi semua data dengan lengkap</p> 	<p>Sistem akan menampilkan halaman tambah staff</p>	<p>Sistem menampilkan halaman tambah staff</p>  <p style="text-align: center;">Valid</p>

Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
		-	Mengubah data staff dengan menekan tombol ubah 	Sistem akan menyimpan data yang sudah diubah dan menampilkan alert " <i>The staff has been saved</i> "	Sistem menyimpan data yang sudah diubah dan menampilkan alert " <i>The staff has been saved</i> "  Valid
			Menghapus data staff yang dikehendaki dengan menekan tombol hapus 	Sistem akan menampilkan alert "Apa Anda yakin ingin menghapus?", jika memilih "Ok" sistem akan menampilkan alert " <i>The staff has been deleted</i> "	Sistem akan menampilkan alert "Apa Anda yakin ingin menghapus?"  Valid jika memilih "Ok" sistem akan menampilkan alert " <i>The staff has been deleted</i> " Valid

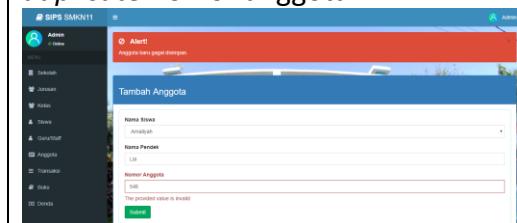
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
SIPS-1-11	Pencarian staff	Nama Staff	Menginputkan nama awal staff=do pada form pencarian 	Sistem akan menampilkan daftar nama staff dengan huruf awal do	 Valid
SIPS-1-12	Pengelolaan anggota	Nama pendek, nomor anggota	Menambah data anggota dengan menekan tombol tambah pada pojok kiri atas 	Sistem akan menampilkan halaman tambah anggota	 Valid

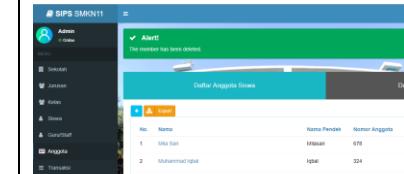
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
	Nama pendek, nomor anggota	Menambah data anggota siswa dan mengisi semua data dengan lengkap		Sistem akan menyimpan data anggota siswa dan menampilkan alert "Anggota baru berhasil disimpan"	 Valid

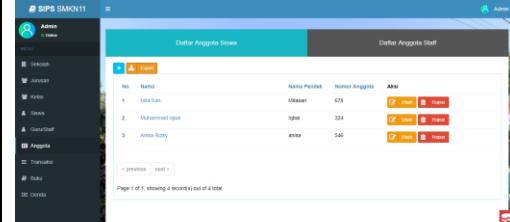
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
		Nama pendek, nomor anggota	Menambah data anggota dan mengisi data dengan menginputkan nomor anggota yang sebelumnya sudah ada	Sistem akan menampilkan alert “Anggota baru gagal disimpan” karena terjadi <i>duplicate</i> nomor anggota	<p>Sistem menampilkan alert “Anggota baru gagal disimpan” karena terjadi <i>duplicate</i> nomor anggota</p>   <p>Valid</p>
			Mengubah data anggota dengan menekan tombol ubah	Sistem akan menyimpan data yang sudah diubah dan menampilkan alert “ <i>The member has been saved</i> ”	<p>Sistem menyimpan data yang sudah diubah dan menampilkan alert “<i>The member has been saved</i>”</p>   <p>Valid</p>

Tabel 4.39 Hasil Uji Black Box (lanjutan)

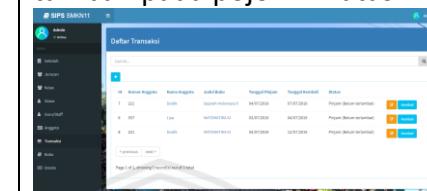
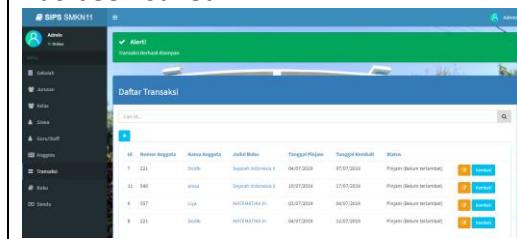
No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
			Menghapus data anggota yang dikehendaki dengan menekan tombol hapus	Sistem akan menampilkan alert “Apa Anda yakin ingin menghapus?”, jika memilih “Ok” sistem akan menampilkan alert <i>“The member has been deleted”</i>	  <p style="text-align: center;">Valid</p> <p>jika memilih “Ok” sistem akan menampilkan alert <i>“The member has been deleted”</i></p>  <p style="text-align: center;">Valid</p>

Tabel 4.39 Hasil Uji Black Box (lanjutan)

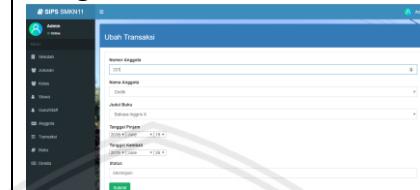
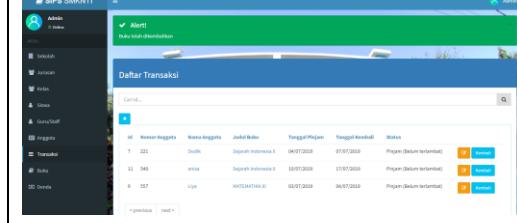
No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
SIPS-1-13	Export data anggota	-	Menekan tombol export	Sistem akan mendownload otomatis data yang di eksport ke dalam bentuk excel	 Sistem mendownload otomatis data yang di eksport ke dalam bentuk excel

Valid

Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
SIPS-1-14	Pengelolaan transaksi	Nomor anggota, nama anggota, judul buku, tanggal pinjam, tanggal kembali, status	Menambah data transaksi dengan menekan tombol tambah pada pojok kiri atas	Sistem akan menampilkan halaman tambah transaksi	<p>Sistem menampilkan halaman tambah transaksi</p> 
			Menambah data transaksi dan mengisi semua data dengan lengkap	Sistem menyimpan data transaksi dan menampilkan alert " <i>The transaction has been saved</i> "	<p>Valid</p>  

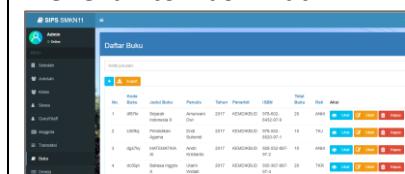
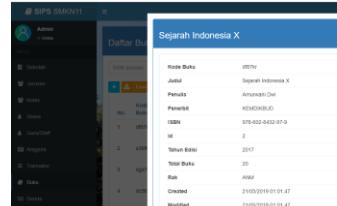
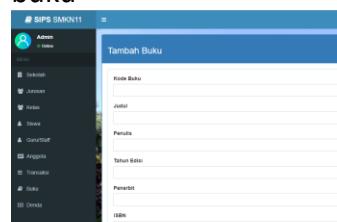
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
		-	Mengubah data transaksi dengan menekan tombol ubah 	Sistem akan menyimpan data yang sudah diubah dan menampilkan alert " <i>The transaction has been saved</i> "	 Valid
		-	Menekan tombol kembali data transaksi yang dikehendaki dengan menekan tombol hapus 	Sistem akan menampilkan alert "Buku telah dikembalikan"	 Valid

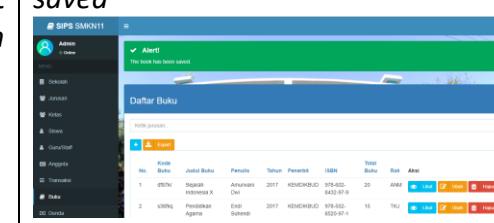
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
SIPS-1-15	Pencarian transaksi	Id	Menginputkan id transaksi=4 pada form pencarian	Sistem akan menampilkan data transaksi dengan id=4	 Valid

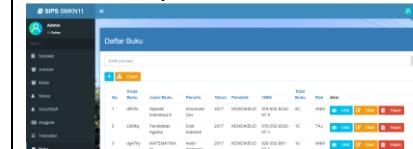
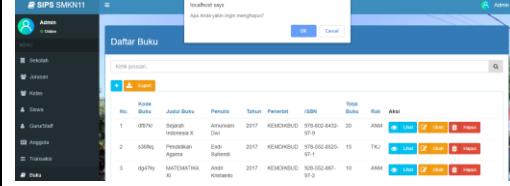
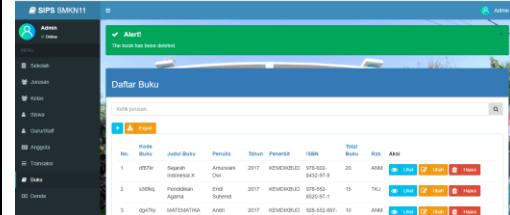
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
SIPS-1-16	Pengelolaan data buku	-	Melihat detail buku dengan menekan tombol lihat	Sistem akan menampilkan detail data buku	  <p>Valid</p>
	Kode buku, judul, penulis, tahun edisi, penerbit, isbn, total buku, rak		Menambah data buku dengan menekan tombol tambah pada pojok kiri atas	Sistem akan menampilkan halaman tambah buku	  <p>Valid</p>

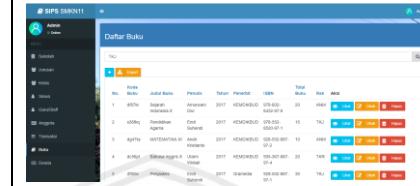
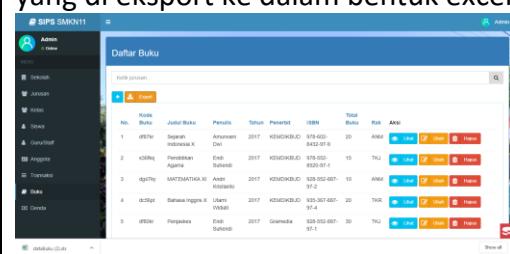
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
		Kode buku, judul, penulis, tahun edisi, penerbit, isbn, total buku, rak	Menambah data buku dan mengisi semua data dengan lengkap	Sistem akan menyimpan data buku dan menampilkan alert " <i>The book has been saved</i> "	 
			Mengubah data buku dengan menekan tombol ubah	Sistem akan menyimpan data yang sudah diubah dan menampilkan alert " <i>The book has been saved</i> "	 

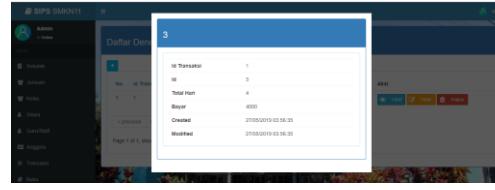
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
		Menghapus data buku yang dikehendaki dengan menekan tombol hapus	<p>Menghapus data buku yang dikehendaki dengan menekan tombol hapus</p> 	<p>Sistem akan menampilkan alert “Apa Anda yakin ingin menghapus?”, jika memilih “Ok” sistem akan menampilkan alert <i>“The book has been deleted”</i></p>	<p>Sistem akan menampilkan alert “Apa Anda yakin ingin menghapus?”</p>  <p style="text-align: center;">Valid</p> <p>jika memilih “Ok” sistem akan menampilkan alert <i>“The book has been deleted”</i></p>  <p style="text-align: center;">Valid</p>

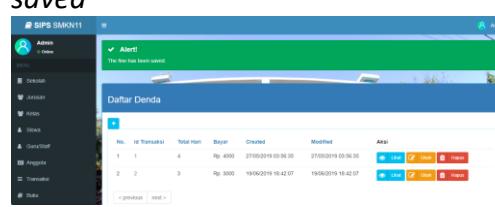
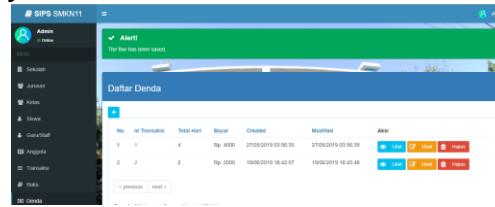
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
SIPS-1-17	Pencarian buku	Nama Jurusan	Menginputkan nama jurusan TKJ pada form pencarian	Sistem akan menampilkan data buku dengan jurusan TKJ	 <p>Sistem menampilkan data buku dengan jurusan TKJ</p>  <p>Valid</p>
SIPS-1-18	Export data buku		Mengekport data buku	Sistem akan mendownload otomatis data yang di eksport ke dalam bentuk excel	 <p>Sistem mendownload otomatis data yang di eksport ke dalam bentuk excel</p>  <p>Valid</p>

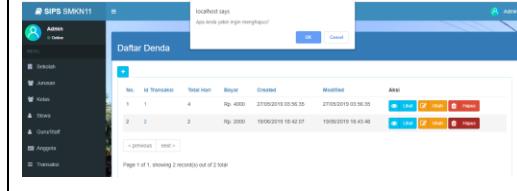
Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
SIPS-1-19	Pengelolaan data denda	-	Melihat detail denda dengan menekan tombol lihat	Sistem akan menampilkan detail data denda	 <p>Sistem menampilkan detail data denda</p> <p>Valid</p>
		Id transaksi, total hari, bayar	Menambah data denda dengan menekan tombol tambah pada pojok kiri atas	Sistem akan menampilkan halaman tambah denda	 <p>Sistem menampilkan halaman tambah denda</p> <p>Valid</p>

Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
		Id transaksi, total hari, bayar	Menambah data denda dan mengisi semua data dengan lengkap 	Sistem akan menyimpan data denda dan menampilkan alert " <i>The fine has been saved</i> "	Sistem menyimpan data denda dan menampilkan alert " <i>The fine has been saved</i> "  Valid
			Mengubah data denda dengan menekan tombol ubah 	Sistem akan menyimpan data yang sudah diubah dan menampilkan alert " <i>The fine has been saved</i> "	Sistem menyimpan data yang sudah diubah dan menampilkan alert " <i>The fine has been saved</i> "  Valid

Tabel 4.39 Hasil Uji Black Box (lanjutan)

No.	Nama Kasus Uji	Data Masukan	Test Case	Hasil yang diharapkan	Hasil Uji
		Menghapus data denda yang dikehendaki dengan menekan tombol hapus	<p>Menghapus data denda yang dikehendaki dengan menekan tombol hapus</p> 	<p>Sistem akan menampilkan alert “Apa Anda yakin ingin menghapus?”, jika memilih “Ok” sistem akan menampilkan alert <i>“The fine has been deleted”</i></p>	<p>Sistem akan menampilkan alert “Apa Anda yakin ingin menghapus?”</p>  <p>Valid</p> <p>jika memilih “Ok” sistem akan menampilkan alert <i>“The fine has been deleted”</i></p>  <p>Valid</p>

4.3.2.2 Pengujian *User Acceptance Testing*(UAT)

Pengujian UAT dilakukan kepada 5 penguji yang terdiri dari ketua perpustakaan SMKN 11 Malang, Petugas Perpustakaan, 1 Staff IT, 1 Guru, dan juga 1 Siswa. Pengujian UAT ini terdapat 3 aspek yang terdiri dari *Usefulness*, *Ease of use*, dan juga *Behavioral intentions* dengan 10 pernyataan. Pada aspek *usefulness* terdapat 4 pernyataan, aspek *ease of use* terdapat 4 pernyataan, dan *behavioral intention* terdapat 2 pernyataan. Berikut merupakan hasil pengujian UAT dapat dilihat pada Tabel 4.40:

Keterangan :

SS = Sangat Setuju

S = Setuju

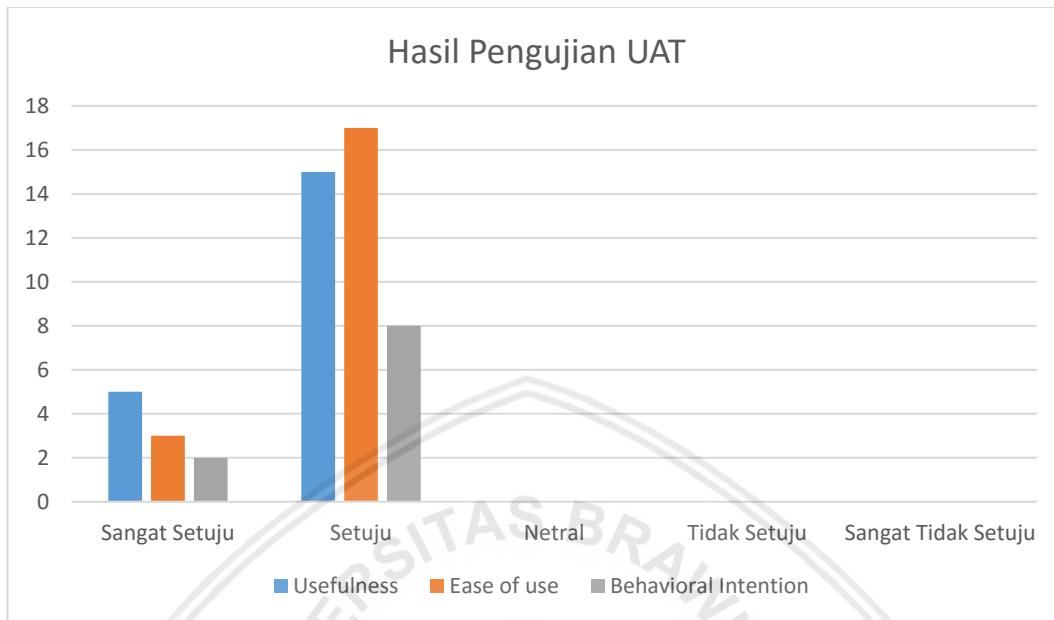
N = Netral

TS = Tidak Setuju

STS = Sangat Tidak Setuju

Tabel 4.40 Hasil Pengujian UAT

No.	Aspek	Pernyataan	SS	S	N	TS	STS
1.	<i>Usefulness</i>	Sistem sesuai dengan yang diharapkan pengguna	2	3			
		Sistem berguna dalam peminjaman buku di perpustakaan	1	4			
		Sistem meningkatkan kinerja di perpustakaan	1	4			
		Sistem meningkatkan efisiensi waktu dalam peminjaman buku di perpustakaan	1	4			
		Total	5	15			
2.	<i>Ease of use</i>	Sistem mudah dipahami penggunaanya	1	4			
		Interaksi sistem tidak membutuhkan banyak upaya/mudah digunakan	1	4			
		Sistem melakukan apa yang diinginkan pengguna		5			
		Sistem tidak membutuhkan waktu yang lama untuk mengakses	1	4			
		Total	3	17			
4.	<i>Behavioral intention</i>	Sistem dapat menampilkan data/mencari data dengan baik	1	4			
		Sistem dapat diakses oleh siswa, guru, staff, dan admin	1	4			
		Total	2	8			



Gambar 4.73 Hasil Pengujian UAT

Dari grafik tersebut dapat disimpulkan total jawaban dibawah ini :

1. Jawaban Sangat setuju = 10×5
2. Jawaban Setuju = 40×4
3. Jawaban Netral = 0×3
4. Jawaban Tidak setuju = 0×2
5. Jawaban Sangat Tidak Setuju = 0×1

Sehingga didapat total nilai dengan hasil perhitungan sebagai berikut :

$$\begin{aligned}
 \text{Total nilai} &= (10 \times 5) + (40 \times 4) + (0 \times 3) + (0 \times 2) + (0 \times 1) \\
 &= (50) + (160) + 0 + 0 + 0 \\
 &= 210
 \end{aligned}$$

Setelah diketahui total nilai, maka selanjutnya menghitung nilai Y untuk mengetahui hasil dari perkalian jumlah skala, jumlah responden, dan jumlah pernyataan sebagai berikut :

$$\begin{aligned}
 Y &= N \times n \times U \\
 &= \text{jumlah skala} \times \text{jumlah responden} \times \text{jumlah pernyataan} \\
 &= 5 \times 5 \times 10 \\
 &= 250
 \end{aligned}$$

$$\begin{aligned}
 \text{Hasil akhir} &= (\text{Total Nilai}/Y) \times 100\% \\
 &= (210/250 \times 100\%) \\
 &= 84\%
 \end{aligned}$$

4.3.2.3 Analisis hasil pengujian

Berdasarkan hasil dari pengujian *black box* yang terdiri dari 19 kebutuhan fungsional yang terdiri dari login, logout, register, pengelolaan data sekolah, pengelolaan data jurusan, pengelolaan data kelas, pencarian kelas, pengelolaan data siswa, pencarian siswa, pengelolaan data staff, pencarian staff, pengelolaan anggota, *export* data anggota, pengelolaan transaksi, pencarian transaksi, pengelolaan data buku, pencarian buku, *export* data buku, dan pengelolaan data denda memiliki hasil 100% valid.

Sedangkan hasil dari pengujian *User Acceptance Testing*(UAT) dengan hasil akhir 84%. Berdasarkan dari Tabel 2.5 yang terdapat pada bab 2, nilai hasil akhir tersebut sesuai dengan penilaian presentase yang memiliki rentang nilai sebesar 60 sampai 89,99% yang dinyatakan “Setuju”. Dengan demikian, Sistem Informasi Perpustakaan Sekolah(SIPS) SMKN 11 Malang telah memenuhi kebutuhan pengguna dalam peminjaman buku di perpustakaan SMKN 11 Malang.



BAB 5 PENUTUP

Pada bab 5 ini merupakan kesimpulan dan juga saran. Kesimpulan berupa hasil yang didapat selama proses keseluruhan pembangunan sistem hingga tahap pengujian.

5.1 Kesimpulan

Berikut kesimpulan yang didapat selama proses pembangunan sistem dari awal yang menggunakan metode *Rapid Application Development* (RAD) hingga tahap pengujian yang menggunakan *blackbox testing* dan *User Acceptance Testing* (UAT):

1. Sistem informasi perpustakaan berbasis web pada SMKN 11 Malang terdiri dari 19 kebutuhan fungsional. Dan sistem dapat diakses oleh admin, siswa, guru, dan juga staff SMKN 11 Malang.
2. Perancangan pada *workshop* desain melibatkan pengguna, yang menghasilkan desain sistem pada iterasi pertama terdiri dari menu sekolah, jurusan, kelas, siswa, guru/staff, anggota, transaksi, buku, dan juga denda. Pada admin dapat mengakses semua fungsi sistem yang meliputi melihat data, menambah, mengedit, menghapus, mengeksport, dan merekap data. Sedangkan pada user siswa, guru, maupun staff dapat melihat informasi perpustakaan dan juga meminjam buku. Tetapi terlebih dahulu harus registrasi jika belum terdaftar untuk dapat melihat informasi sistem. Pada iterasi kedua yaitu ditambahkan nama jurusan pada menu buku agar pada saat pencarian dapat menemukan buku pada jurusan masing-masing. Sehingga proses pencarian menjadi lebih mudah. Selain itu pada user siswa, guru, dan juga staff pada iterasi kedua hanya dapat melihat informasi perpustakaan saja. Dan terlebih dahulu harus registrasi jika belum terdaftar anggota agar bisa melihat informasi pada sistem.
3. Hasil dari implementasi sistem informasi perpustakaan SMKN 11 Malang merupakan hasil dari iterasi kedua yang didapatkan setelah tidak ada lagi perbaikan sistem yang telah disempurnakan, dibangun, kemudian dilakukan pengujian sistem kepada 5 user yang terdiri dari ketua perpustakaan, petugas perpustakaan, 1 guru, 1 staff dan 1 siswa.
4. Hasil pengujian *blackbox* dengan 19 kebutuhan fungsional secara keseluruhan memiliki hasil 100% valid. Sedangkan hasil pengujian UAT yaitu memiliki hasil akhir 84% dimana pengguna menyatakan "Setuju" terhadap Sistem Infomasi Perpustakaan Sekolah (SIPS) SMKN 11 Malang. Dan telah memenuhi kebutuhan pengguna dalam peminjaman buku di perpustakaan SMKN 11 Malang.

5.2 Saran

Saran untuk kedepannya yaitu, sistem dapat dikembangkan menjadi lebih baik lagi dengan menambahkan fitur-fitur yang lebih kompleks dan juga mengembangkan tampilan *interface* menjadi lebih menarik. Sehingga pelayanan pada perpustakaan menjadi lebih baik lagi.

DAFTAR REFERENSI

- Abigail, A.D., 2018. *Pengembangan Dan Analisis Sistem Informasi Perpustakaan Sekolah Berbasis Website Di SMK Negeri 1 Seyegan, Sleman, Yogyakarta.* S1. Universitas Negeri Yogyakarta. Tersedia di: <https://eprints.uny.ac.id/56146/1/Skripsi_Addis%20Sousan%20Abigail_13520241071.pdf> [Diakses 19 Januari 2019]
- Aryanto, A., 2017. Pembuatan Sistem Informasi Perpustakaan Smp Muhammadiyah 7 Surakarta. *Speed-Sentra Penelitian Engineering dan Edukasi*, [online] Tersedia di: <<http://speed.web.id/ejournal/index.php/speed/article/view/219/214>> [Diakses 19 Januari 2019]
- Aswati, S., & Siagian, Y. 2016. Model Rapid Application Development Dalam Rancang Bangun Sistem Informasi Pemasaran Rumah (Studi Kasus: Perum Perumnas Cabang Medan. [online] Tersedia di: <http://is.its.ac.id/pubs/oajis/index.php/file/download_file/1676> [Diakses 25 Juli 2019]
- Atmoko, Y. D., & Arizona, N. D. 2016. Aplikasi Penjualan dan Pembelian Secara Tunai dan Non Tunai Pada PD Istana Gypsum Pontianak. *Simnasiptek 2016*, 1(1), 38-44. [online] Tersedia di: <<http://seminar.bsi.ac.id/simnasiptek/index.php/simnasiptek-2016/article/viewFile/86/86>> [Diakses 25 Juli 2019]
- Bari, A., & Syam, A. 2008. *Cakephp application development*. Packt Publishing Ltd.
- Bertram, D. 2007. Likert scales. Retrieved November, 2, 2013. [online] Tersedia di:<https://www.researchgate.net/profile/Farhad_Waseel/post/How_can_one_help_people_create_simple_scales_from_Likert-scored_items/attachment/5c9e2e9acfe4a7299499df34/AS%3A741800755068929%401553870490367/download/topic-dane-likert.pdf> [Diakses 24 Juli 2019]
- Boone, H. N., & Boone, D. A. 2012. Analyzing likert data. *Journal of extension*, 50(2), 1-5. [online] Tersedia di: <https://www.researchgate.net/profile/Mahesh_Tengli2/post/What_statistical_analysis_should_I_use_for_Likert-Scale_data/attachment/5d09cd41cfe4a7968dac2e55/AS%3A771383042789382%401560923457797/download/JOE_v50_2tt2+likert+analysis+imp.pdf> [Diakses 25 Juli 2019]
- Davis, F. D., & Venkatesh, V. 2004. *Toward preprototype user acceptance testing of new information systems: implications for software project management*. *IEEE Transactions on Engineering management*, 51(1), 31-46. [online] Tersedia di: <https://www.researchgate.net/profile/Fred_Davis2/publication/3076634_Toward_Preprototype_User_Acceptance_Testing_of_New_Information_Systems_Implications_for_Software_Project_Management/links/54934b350cf25de74db4f195.pdf> [Diakses 25 Juli 2019]

- Fabyianto, A. R. 2019. Pengembangan Sistem Informasi Penilaian Kinerja Guru Menggunakan Metode Rational Unified Process (RUP) Berbasis Web (Studi Pada SD Negeri Prigen 1). S1. Universitas Brawijaya.
- Hariadi, F. 2012. Pembuatan sistem informasi perpustakaan pada SDN Sukoharjo pacitan berbasis web. *IJNS-Indonesian Journal on Networking and Security*, 2(4). [online] Tersedia di: <<https://ijns.org/journal/index.php/ijns/article/view/246/240>> [Diakses 15 Januari 2019]
- Hendini, A. 2016. Pemodelan Uml Sistem Informasi Monitoring Penjualan Dan Stok Barang (Studi Kasus: Distro Zhezha Pontianak). *Jurnal Khatulistiwa Informatika*, 4(2), 107-116. [online] Tersedia di: <<http://ejournal.bsi.ac.id/index.php/jki/article/download/78/73>> [Diakses 20 Januari 2019]
- Hendrianto, D. E. 2013. Pembuatan Sistem Informasi Perpustakaan Berbasis Website Pada Sekolah Menegah Pertama Negeri 1 Donorojo Kabupaten Pacitan. *IJNS-Indonesian Journal on Networking and Security*, 4(3). [online] Tersedia di: <<https://ijns.org/journal/index.php/ijns/article/viewFile/288/282>> [Diakses 19 Januari 2019]
- Iswandy, E. 2015. Sistem Penunjang Keputusan Untuk Menentukan Penerimaan Dana Santunan Sosial Anak Nagari Dan Penyalurannya Bagi Mahasiswa Dan Pelajar Kurang Mampu Di Kenagarian Barung-Barung Balantai Timur. *Jurnal Teknif*, 3(2). [online] Tersedia di: <<http://ejournal.itp.ac.id/index.php/tinformatika/article/viewFile/324/313>> [Diakses 3 Juni 2019]
- Kadir, A. 1999. *Konsep & Tuntunan Praktis Basis Data*. Penerbit Andi.
- Kendall, E. Kenneth and Julie E. Kendall. 2011. *System Analysis and Design Eighth Edition*. New Jersey. Pearson.
- Khotimah, H., & Patrie, H. 2018. *RANCANG BANGUN SISTEM INFORMASI POIN PELANGGARAN TATA TERTIB PADA SMP CENDERAWASIH II DENGAN METODOLOGI BERORIENTASI OBYEK*. *InDonEsiA journal Information System*, 1(3), 237-242. [online] Tersedia di: <<http://jom.fti.budiluhur.ac.id/index.php/IDEALIS/article/download/709/150>> [Diakses 25 Juli 2019]
- Kristanto, I. H. 1994. *Konsep & Perancangan Database*. Penerbit Andi.
- Mulyati, M., Tarmizi, R., & Panugali, A. (2018). Sistem Informasi Absensi Berbasis Web Pada Badan Penanggulangan Bencana Daerah Kota Tangerang. *ICIT Journal*, 4(2), 117-127. [online] Tersedia di: <<http://ejournal.raharja.ac.id/index.php/icit/article/download/558/412>> [Diakses 25 Juli 2019]
- Mutiara, A. B., Awaludin, R., Muslim, A., & Oswari, T. (2014). Testing Implementasi Website Rekam Medis Elektronik Opelgunasys Dengan Metode Acceptance Testing. *Prosiding KOMMIT*. [online] Tersedia di: <<http://ejournal.gunadarma.ac.id/index.php/kommit/article/viewFile/1010/874>> [Diakses 25 Juli 2019]

- Nazir, M. 1988. MetodePenelitian. *Jakarta: Ghalia Indonesia*. [online] Tersedia di: <http://a-research.upi.edu/operator/upload/s_pkn_032713_chapter3.pdf> [Diakses 25 Juli 2019]
- Nidhra, S., & Dondeti, J. 2012. Black box and white box testing techniques-a literature review. *International Journal of Embedded Systems and Applications (IJESA)*, 2(2), 29-50. [online] Tersedia di: <http://www.academia.edu/download/38077013/Black_Box_and_White_Box_Testing_Techniques_-_A_Literature_Review.pdf> [Diakses 25 Juli 2019]
- Putri, N. K., 2018. *Pembangunan Sistem Rekam Medis Gigi Berbasis Web*. S1. Universitas Brawijaya.
- Putri, N. S., Rochim, A. F., & Windasari, I. P. 2013. Perancangan Sistem Informasi Rekam Medis Rawat Inap Rumah Sakit Berbasis Web. *Jurnal Teknologi dan Sistem Komputer*, 1(1), 1-13. [online] Tersedia di: <<https://jtsiskom.undip.ac.id/index.php/jtsiskom/article/download/1614/12407>> [Diakses 25 Juli 2019]
- Prakoso, G., Kertahadi, K., & Susilo, H. 2017. Pengaruh Penerapan Otomasi Perpustakaan terhadap Kualitas Layanan dan Kinerja di Perpustakaan Umum (Studi pada Kantor Perpustakaan dan Dokumentasi Pemerintah Kota Batu). *Jurnal Administrasi Bisnis*, 50(6), 144-150. [online] Tersedia di: <<http://administrasibisnis.studentjournal.ub.ac.id/index.php/jab/article/download/2091/2483>> [Diakses 7 Maret 2019]
- Pressman, R. S. 2001. *Software engineering: a practitioner's approach* / Roger S. Pressman.—5th ed. p. cm.— (McGraw-Hill series in computer science). ISBN 0-07-365578-3.
- Priyanti, D., & Iriani, S. 2013. Sistem Informasi Data Penduduk Pada Desa Bogoharjo Keccamatian Ngadirojo Kabupaten Pacitan. *IJNS-Indonesian Journal on Networking and Security*, 2(4). [online] Tersedia di: <<http://ijns.org/journal/index.php/ijns/article/viewFile/181/176>> [Diakses 26 Maret 2019]
- Jacobson, I., Booch, G., & Rumbaugh, J. 1996. The Unified Modeling Language. *University Video Communications*. [online] Tersedia di: <http://www.academia.edu/download/34089098/UML_2IA.pdf> [Diakses 26 Maret 2019]
- Rajkumar. 2019. *Software Development Life Cycle – SDLC | Software Testing Material*. [online] Tersedia di: <<https://www.softwaretestingmaterial.com/sdlc-software-development-life-cycle/>> [Diakses 4 Februari 2019].
- Renatha, F. A., Satoto, K. I., & Nurhayati, O. D. 2015. Perancangan dan Pengembangan Sistem Informasi Perpustakaan Berbasis Web (Studi Kasus Jurusan Sistem Komputer). *Jurnal Teknologi dan Sistem Komputer*, 3(3), 343-353. [online] Tersedia di: <<https://jtsiskom.undip.ac.id/index.php/jtsiskom/article/download/12103/11755>> [Diakses 19 Januari 2019]
- Saleh, A. R., & Komalasari, R. 2014. Pengertian Perpustakaan dan Dasar-dasar Manajemen Perpustakaan. [online] Tersedia di:

- <<http://www.pustaka.ut.ac.id/lib/wp-content/uploads/pdfmk/PUST2229-M1.pdf>> [Diakses 14 Mei 2019]
- Saragih, A. H. 2017. *Sistem Informasi Perpustakaan Sekolah Berbasis Visual Basic Di SMK Muhammadiyah 2 Moyudan*. S1. Universitas Yogyakarta. Tersedia di: <<https://eprints.uny.ac.id/50802/1/SISTEM%20INFORMASI%20PERPUSTAKAAN%20SEKOLAH.pdf>> [Diakses 19 Januari 2019]
- Setiawan, A. 2011. *Rapid Application Development*. [online] Tersedia di: <<https://mazadie.files.wordpress.com/2012/01/rapid-application-development.pdf>> [Diakses 6 Februari 2019]
- Suryati, B. E. P. 2010. Pembangunan Sistem Informasi Pendataan Rakyat Miskin Untuk Program Beras Miskin (Raskin) Pada Desa Mantren Kecamatan Kebonagung Kabupaten Pacitan. *Indonesian Jurnal on Computer Science-Speed (IJCSS)*, 13, 1979-933. [online] Tersedia di: <http://www.academia.edu/download/49544139/Pembangunan_Sistem_Informasi_Pendataan_R.pdf> [Diakses 26 Maret 2019]
- Sutopo, P., Cahyadi, D., & Arifin, Z. 2017. Sistem Informasi Eksekutif Sebaran Penjualan Kendaraan Bermotor Roda 2 di Kalimantan Timur Berbasis Web. [online] Tersedia di: <<https://osf.io/preprints/inarxiv/5zyb8/>> [Diakses 29 Maret 2019]
- Wahyuningsih, Y. 2017. Pengembangan Aplikasi Test Online dengan Menggunakan Framework CodeIgniter di SMK Darul Ma'wa Plumpang pada Mata Pelajaran Jaringan Dasar Kelas X. *IT-EDU*, 2(02). [online] Tersedia di: <<https://jurnalmahasiswa.unesa.ac.id/index.php/it-edu/article/viewFile/22139/20290>> [Diakses 25 Juli 2019]
- Wijaya, K. K. 2016. *Visual Studio Code : Aplikasi Editor Kode dari Microsoft untuk Windows, Linux, dan OS X.* [online] Tersedia di: <<https://id.techinasia.com/visual-studio-code-editor-kode-microsoft>> [Diakses 26 Maret 2019]
- Yusri, M. 2015. Sistem Informasi Perpustakaan Berbasis Web pada SMP Frater Makassar. *JUPITER*, 14(2). [online] Tersedia di: <<http://journal.unhas.ac.id/index.php/jupiter/article/viewFile/39/37>> [Diakses 19 Januari 2019]



LAMPIRAN A. 1**A.1 HASIL WAWANCARA DENGAN PETUGAS PERPUSTAKAAN SMKN 11 MALANG**

Nama : Titik Ambarsari
Jabatan : Petugas Perpustakaan
Tempat : Ruang Perpustakaan
Tanggal : 5 Maret 2019
Q : *Questions*
A : *Answer*

Q : Apakah posisi Anda dibagian perpustakaan ini?

A : Sebenarnya untuk petugas perpustakaan di SMK Negeri 11 Malang ini belum ada, saya hanya ditugaskan untuk membantu pelayanan perpustakaan. Awal saya ditugaskan pada tahun 2013 membantu bu Rahmah selaku ketua perpustakaan disini.

Q : Sudah berapa lama Anda bekerja dibagian ini?

A : Kurang lebih 6 tahun saya bekerja sebagai petugas di perpustakaan ini.

Q : Bisa diceritakan mengenai sistem informasi perpustakaan yang ada saat ini?

A : Terdapat aplikasi SLA (*Smart Library Automation*) di perpustakaan ini, aplikasi ini sangat membantu petugas perpustakaan.

Q : Aktifitas apa saja yang ada pada perpustakaan dan khususnya pada sistem informasi perpustakaan ini?

A : Ada banyak, seperti pengolahan buku, daftar anggota, pelayanan peminjaman, dan lain sebagainya.

Q : Bagaimana prosedur peminjaman dan pengembalian buku? Data apa saja yang dicatat pada saat tersebut?

A : Aplikasi SLA untuk saat ini masih manual dalam memberikan pelayanan kepada pengunjung yang meminjam buku di perpustakaan. Terlebih dahulu dicatat nama peminjamnya, kelasnya, dan juga tanda tangan peminjam. Namun, untuk guru bisa langsung menginputkan data di Aplikasi SLA ini.

Q : Apa kendala pada sistem tersebut?

A : Kendalanya tidak bisa *copy paste* data-data yang masuk dan harus satu per satu dalam memindahkan data, aplikasi SLA juga tidak bisa dibawa kemana-kemana dalam mengaksesnya, jadi jika ingin mengaksesnya hanya pada komputer tersebut. Dan jika error, harus mendatangkan developernya kesini.

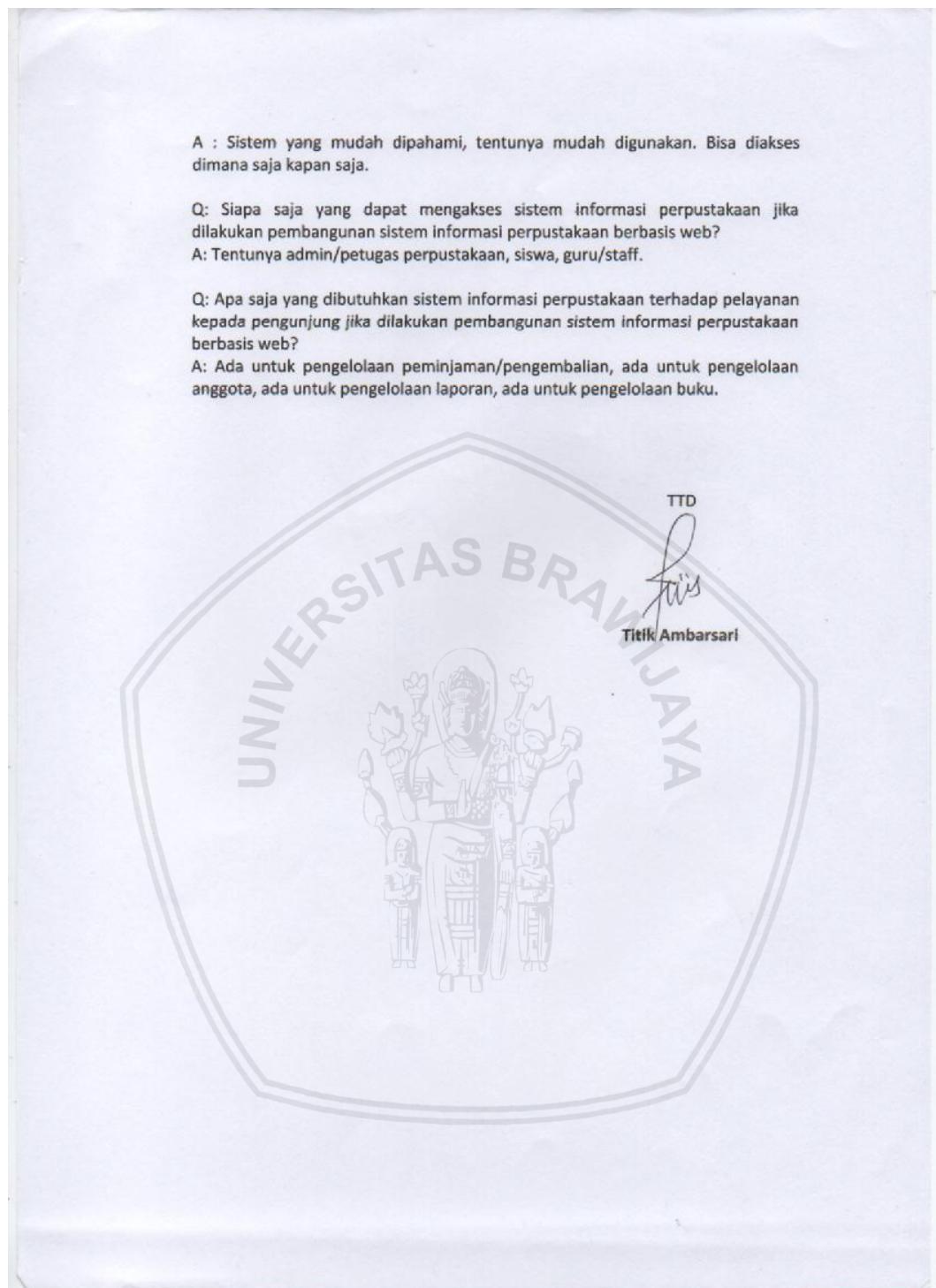
Q : Jika dilakukan pembangunan sistem informasi perpustakaan berbasis web, sistem seperti apa yang Anda inginkan?

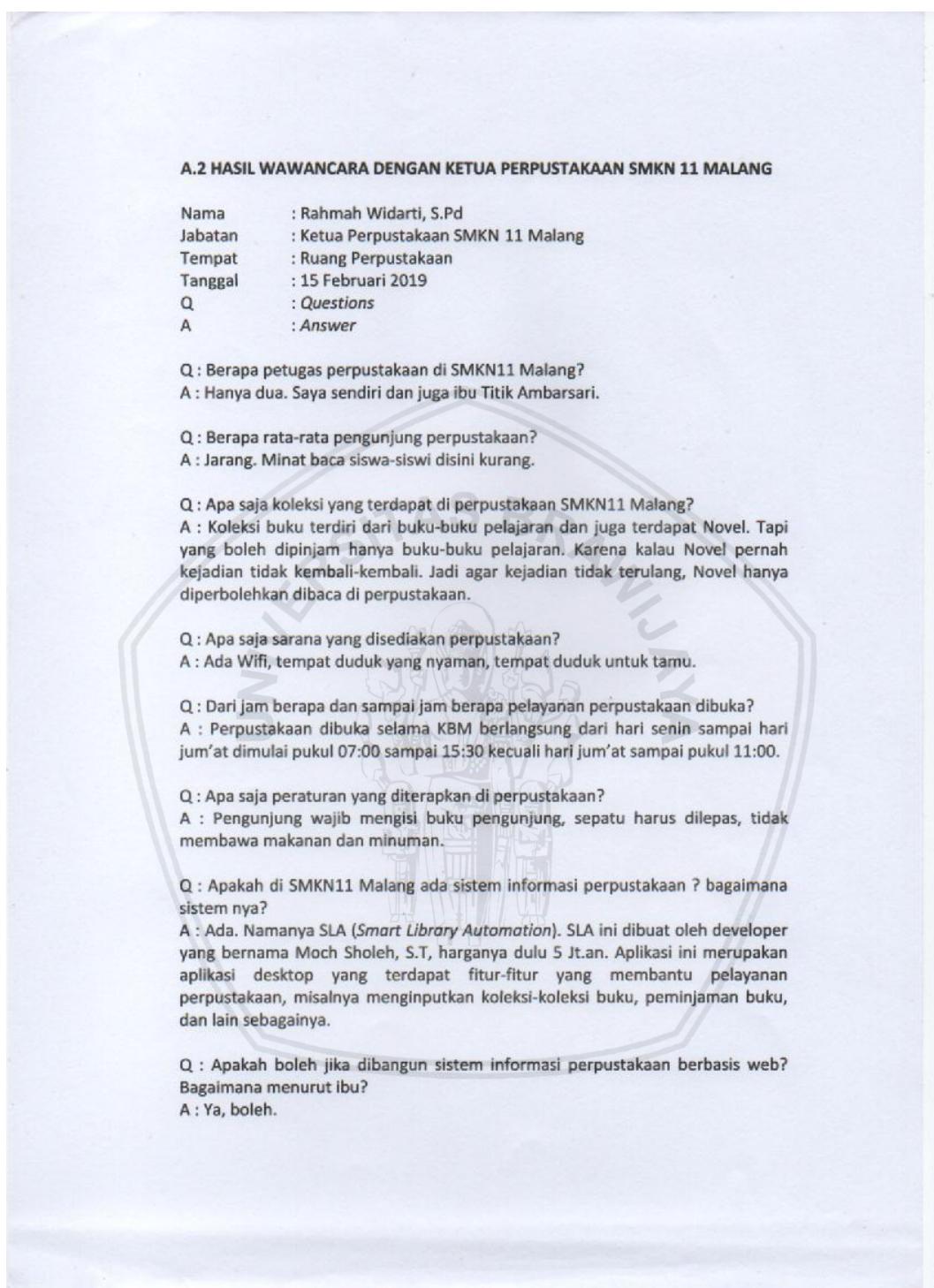
A : Sistem yang mudah dipahami, tentunya mudah digunakan. Bisa diakses dimana saja kapan saja.

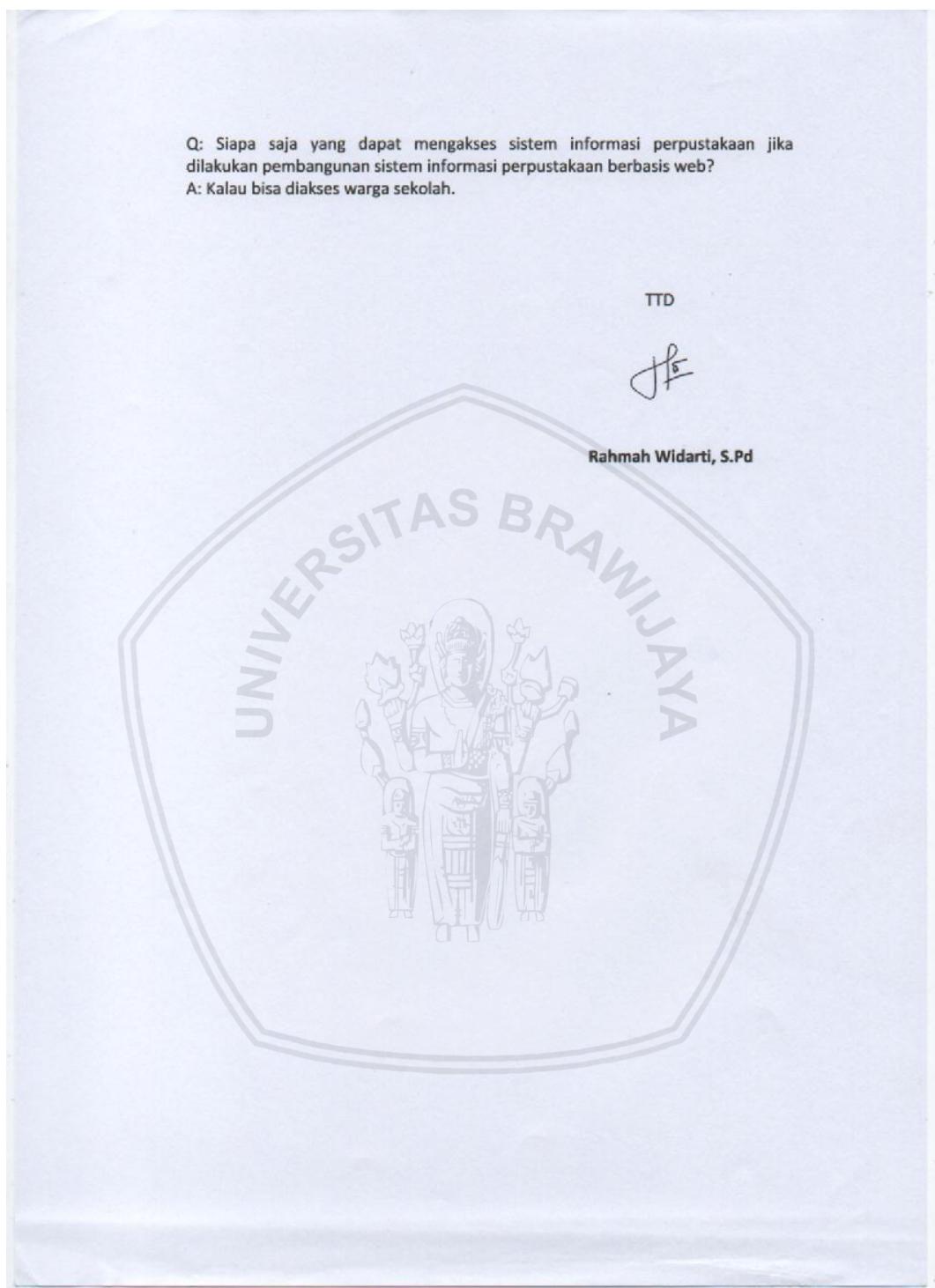
Q: Siapa saja yang dapat mengakses sistem informasi perpustakaan jika dilakukan pembangunan sistem informasi perpustakaan berbasis web?
A: Tentunya admin/petugas perpustakaan, siswa, guru/staff.

Q: Apa saja yang dibutuhkan sistem informasi perpustakaan terhadap pelayanan kepada pengunjung jika dilakukan pembangunan sistem informasi perpustakaan berbasis web?

A: Ada untuk pengelolaan peminjaman/pengembalian, ada untuk pengelolaan anggota, ada untuk pengelolaan laporan, ada untuk pengelolaan buku.



LAMPIRAN A. 2



LAMPIRAN B. 1**B.1 HASIL EVALUASI WORKSHOP DESAIN DENGAN KETUA PERPUSTAKAAN SMKN 11 MALANG**

Nama : Rahmah Widarti, S.Pd
 Jabatan : Ketua Perpustakaan
 Tempat : Ruang Perpustakaan SMKN 11 Malang
 Tanggal : 21 Mei 2019

Workshop Desain Iterasi-1	Workshop Desain Iterasi-2
<p>Pada workshop desain iterasi pertama terdapat menu sekolah, jurusan, kelas, siswa, staff, anggota, transaksi, buku, dan juga denda. Pada menu denda ditambahkan agar tidak melewati batas peminjaman.</p> <p>Admin dapat mengakses semua fungsi sistem yang meliputi menambah data, mengubah data, melihat data, dan juga menghapus data. Sedangkan user siswa dan user staff dapat mengakses sistem yang terdiri dari melihat informasi perpustakaan dan juga meminjam buku perpustakaan.</p>	<p>Pada workshop desain iterasi kedua, hasil dari evaluasi yaitu pada user sebaiknya hanya dapat melihat informasi perpustakaan saja, sedangkan untuk meminjam dilakukan melalui admin. Hal ini dilakukan agar tidak terjadi kesalahan memasukkan data.</p> <p>Selain itu, pada menu buku ditambahkan dengan nama jurusan agar pada saat pencarian bisa mengetahui buku apa saja yang sesuai dengan jurusan masing-masing.</p>

Berdasarkan hasil dari evaluasi iterasi ke-2, sistem informasi perpustakaan berbasis web cukup untuk memenuhi kebutuhan pengguna. Sehingga iterasi desain pada sistem informasi perpustakaan berbasis web dihentikan sampai iterasi ke-2 dan telah disepakati oleh Ibu Rahmah Widarti selaku ketua perpustakaan SMK Negeri 11 Malang.

Malang, 21 Mei 2019

Mengetahui,
 Ketua Perpustakaan

Rahmah Widarti, S.Pd
 NIP: 19790721 201407 2 002

Kuisisioner pengujian User Acceptance Testing(UAT)

Nama	Rahmah. Widarti, S.Pd.					Sangat Setuju (SS)	Setuju (S)	Netral (N)	Tidak Setuju (TS)	Sangat Tidak Setuju (STS)
Pekerjaan	Eva Kewirausahaan									
Petunjuk pengisian : Berilah centang (✓) pada kolom yang sesuai dengan pernyataan. Sangat Setuju (SS) = 5 Setuju (S) = 4 Netral (N) = 3 Tidak Setuju (TS) = 2 Sangat Tidak Setuju (STS) = 1										
No.	Pernyataan	Sangat Setuju (SS)	Setuju (S)	Netral (N)	Tidak Setuju (TS)	Sangat Tidak Setuju (STS)				
		5	4	3	2	1				
<i>USEFULNESS</i>										
1.	Sistem sesuai dengan yang diharapkan pengguna		✓							
2.	Sistem berguna dalam peminjaman buku di perpustakaan		✓							
3.	Sistem meningkatkan kinerja di perpustakaan		✓							
4.	Sistem meningkatkan efisiensi waktu dalam peminjaman buku di perpustakaan		✓							
<i>EASE OF USE</i>										
5.	Sistem mudah dipahami penggunaanya		✓							
6.	Interaksi sistem tidak membutuhkan banyak upaya/mudah digunakan		✓							
7.	Sistem melakukan apa yang diinginkan pengguna		✓							
8.	Sistem tidak membutuhkan waktu yang lama dalam mengakses		✓							

No.	Pernyataan	Sangat Setuju (SS)	Setuju (S)	Netral (N)	Tidak Setuju (TS)	Sangat Tidak Setuju (STS)
		5	4	3	2	1
<i>Behavioral intention</i>						
9.	Sistem dapat menampilkan data/mencari data dengan baik		✓			
10.	Sistem dapat diakses oleh siswa, guru, staff, dan admin		✓			

Malang, 24 Juni 2019

(Pahmah. M)

Kuisisioner pengujian User Acceptance Testing(UAT)

Nama : TITIK AMBARWATI	Pekerjaan : STAF PERPU	Sangat Setuju (SS)	Tidak Setuju (TS)	Setuju (S)	Netral (N)	Tidak Setuju (TS)	Sangat Tidak Setuju (STS)
------------------------	------------------------	--------------------	-------------------	------------	------------	-------------------	---------------------------

Petunjuk pengisian :
Berilah centang (✓) pada kolom yang sesuai dengan pernyataan.
Sangat Setuju (SS) = 5
Setuju (S) = 4
Netral (N) = 3
Tidak Setuju (TS) = 2
Sangat Tidak Setuju (STS) = 1

No.	Pernyataan	Sangat Setuju (SS)	Setuju (S)	Netral (N)	Tidak Setuju (TS)	Sangat Tidak Setuju (STS)
		5	4	3	2	1
<i>Usefulness</i>						
1.	Sistem sesuai dengan yang diharapkan pengguna		✓			
2.	Sistem berguna dalam peminjaman buku di perpustakaan		✓			
3.	Sistem meningkatkan kinerja di perpustakaan		✓			
4.	Sistem meningkatkan efisiensi waktu dalam peminjaman buku di perpustakaan		✓			
<i>Ease of use</i>						
5.	Sistem mudah dipahami penggunaanya		✓			
6.	Interaksi sistem tidak membutuhkan banyak upaya/mudah digunakan		✓			
7.	Sistem melakukan apa yang diinginkan pengguna		✓			
8.	Sistem tidak membutuhkan waktu yang lama dalam mengakses		✓			

No.	Pernyataan	Sangat Setuju (SS)	Setuju (S)	Netral (N)	Tidak Setuju (TS)	Sangat Tidak Setuju (STS)
		5	4	3	2	1
<i>Behavioral intention</i>						
9.	Sistem dapat menampilkan data/mencari data dengan baik		✓			
10.	Sistem dapat diakses oleh siswa, guru, staff, dan admin		✓			

Malang, 24 Juni 2019

TTD
[Signature]



The logo of Universitas Brawijaya (UB) is a circular emblem. It features a central figure, likely a deity or a historical figure, standing on a chariot pulled by four horses. The figure holds a sword in one hand and a shield in the other. The chariot has four wheels. Below the central figure, there is a smaller figure standing. The entire emblem is set against a background of horizontal lines, possibly representing a grid or a stylized landscape. The word "UNIVERSITAS" is written vertically along the left side of the circle, and "BRAWIJAYA" is written vertically along the right side.

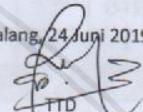
Kuisisioner pengujian User Acceptance Testing(UAT)

Nama : DE DI RESTU SUFAMTO
 Pekerjaan : Guru.

Petunjuk pengisian :
 Berilah centang (✓) pada kolom yang sesuai dengan pernyataan.
 Sangat Setuju (SS) = 5
 Setuju (S) = 4
 Netral (N) = 3
 Tidak Setuju (TS) = 2
 Sangat Tidak Setuju (STS) = 1

No.	Pernyataan	Sangat Setuju (SS)	Setuju (S)	Netral (N)	Tidak Setuju (TS)	Sangat Tidak Setuju (STS)
		5	4	3	2	1
<i>Usefulness</i>						
1.	Sistem sesuai dengan yang diharapkan pengguna	✓				
2.	Sistem berguna dalam peminjaman buku di perpustakaan	✓				
3.	Sistem meningkatkan kinerja di perpustakaan	✓				
4.	Sistem meningkatkan efisiensi waktu dalam peminjaman buku di perpustakaan	✓				
<i>Ease of use</i>						
5.	Sistem mudah dipahami penggunaannya	✓				
6.	Interaksi sistem tidak membutuhkan banyak upaya/mudah digunakan	✓				
7.	Sistem melakukan apa yang diinginkan pengguna		✓			
8.	Sistem tidak membutuhkan waktu yang lama dalam mengakses	✓				

No.	Pernyataan	Sangat Setuju (SS)	Setuju (S)	Netral (N)	Tidak Setuju (TS)	Sangat Tidak Setuju (STS)
		5	4	3	2	1
<i>Behavioral intention</i>						
9.	Sistem dapat menampilkan data/mencari data dengan baik	✓				
10.	Sistem dapat diakses oleh siswa, guru, staff, dan admin	✓				

Malang, 24 Juni 2019

TTD



Kuisisioner pengujian User Acceptance Testing(UAT)

Nama	Lestari					Sangat Setuju (SS)	Setuju (S)	Netral (N)	Tidak Setuju (TS)	Sangat Tidak Setuju (STS)
Pekerjaan	Petugas UKS					5	4	3	2	1

Petunjuk pengisian :
Berilah centang (✓) pada kolom yang sesuai dengan pernyataan.

Sangat Setuju (SS) = 5
 Setuju (S) = 4
 Netral (N) = 3
 Tidak Setuju (TS) = 2
 Sangat Tidak Setuju (STS) = 1

No.	Pernyataan	Sangat Setuju (SS)	Setuju (S)	Netral (N)	Tidak Setuju (TS)	Sangat Tidak Setuju (STS)
		5	4	3	2	1
<i>Usefulness</i>						
1.	Sistem sesuai dengan yang diharapkan pengguna		✓			
2.	Sistem berguna dalam peminjaman buku di perpustakaan		✓			
3.	Sistem meningkatkan kinerja di perpustakaan		✓			
4.	Sistem meningkatkan efisiensi waktu dalam peminjaman buku di perpustakaan		✓			
<i>Ease of use</i>						
5.	Sistem mudah dipahami penggunaanya		✓			
6.	Interaksi sistem tidak membutuhkan banyak upaya/mudah digunakan		✓			
7.	Sistem melakukan apa yang diinginkan pengguna		✓			
8.	Sistem tidak membutuhkan waktu yang lama dalam mengakses		✓			

No.	Pernyataan	Sangat Setuju (SS)	Setuju (S)	Netral (N)	Tidak Setuju (TS)	Sangat Tidak Setuju (STS)
		5	4	3	2	1
<i>Behavioral intention</i>						
9.	Sistem dapat menampilkan data/mencari data dengan baik		✓			
10.	Sistem dapat diakses oleh siswa, guru, staff, dan admin		✓			

Malang, 24 Juni 2019

TTD
Juli Lestari

Kuislone pengujian User Acceptance Testing(UAT)

Nama : SULIFA
 Pekerjaan : Siswa / kls XI kpr

Petunjuk pengisian :
 Berilah centang (✓) pada kolom yang sesuai dengan pernyataan.
 Sangat Setuju (SS) = 5
 Setuju (S) = 4
 Netral (N) = 3
 Tidak Setuju (TS) = 2
 Sangat Tidak Setuju (STS) = 1

No.	Pernyataan	Sangat Setuju (SS)	Setuju (S)	Netral (N)	Tidak Setuju (TS)	Sangat Tidak Setuju (STS)
		5	4	3	2	1
<i>Usefulness</i>						
1.	Sistem sesuai dengan yang diharapkan pengguna	✓				
2.	Sistem berguna dalam peminjaman buku di perpustakaan		✓			
3.	Sistem meningkatkan kinerja di perpustakaan		✓			
4.	Sistem meningkatkan efisiensi waktu dalam peminjaman buku di perpustakaan		✓			
<i>Ease of use</i>						
5.	Sistem mudah dipahami penggunaanya		✓			
6.	Interaksi sistem tidak membutuhkan banyak upaya/mudah digunakan		✓			
7.	Sistem melakukan apa yang diinginkan pengguna		✓			
8.	Sistem tidak membutuhkan waktu yang lama dalam mengakses	✓				

No.	Pernyataan	Sangat Setuju (SS)	Setuju (S)	Netral (N)	Tidak Setuju (TS)	Sangat Tidak Setuju (STS)
		5	4	3	2	1
<i>Behavioral intention</i>						
9.	Sistem dapat menampilkan data/mencari data dengan baik		✓			
10.	Sistem dapat diakses oleh siswa, guru, staff, dan admin		✓			

Malang, 24 Juni 2019

TTD
[Signature]

