

**PENGEMBANGAN SISTEM INFORMASI KEUANGAN
(STUDI KASUS : KB & TK PERMATA IMAN MALANG)**

SKRIPSI

Disusun oleh:
Muhammad Rizsky
NIM: 135150401111092



PROGRAM STUDI SISTEM INFORMASI
JURUSAN SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018

PENGESAHAN

PENGEMBANGAN SISTEM INFORMASI KEUANGAN
(STUDI KASUS : KB & TK PERMATA IMAN MALANG)
SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Muhammad Rizsky
NIM: 135150401111092

Skripsi ini telah diuji dan dinyatakan lulus pada
31 Juli 2018

Telah di periksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

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

NIP: 19860521 201212 1 001

Mochamad Chandra Saputra, S.Kom., M.T.,
M.Eng.

NIK. 201606 900621 2 001

Mengetahui
Ketua Jurusan Sistem Informasi



Dr. Eng. Herman Tolle, S.T, M.T.

NIP: 19740823 200012 1 001

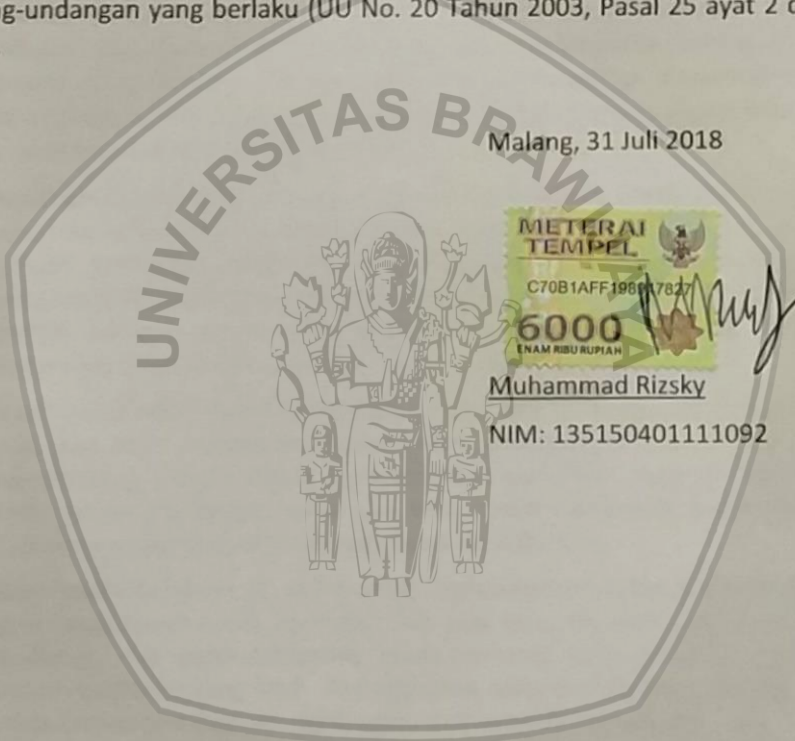


PERNYATAAN ORISINALITAS

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 pustaka.

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, 31 Juli 2018



Muhammad Rizsky

NIM: 135150401111092



KATA PENGANTAR

Segala puji bagi Allah SWT atas limpahan rahmat yang diberikan sehingga penulis dapat menyelesaikan skripsi berjudul “Pengembangan Sistem Informasi Keuangan Studi Kasus Kelompok Bermain dan Taman Kanak Kanak Permata Iman”

Penelitian ini tidak dapat penulis selesaikan tanpa didukung oleh ilmu yang didapatkan di kelas perkuliahan dan di saat yang sama juga mengajarkan banyak ilmu lain yang belum penulis dapatkan. Selain itu, penelitian ini juga memberikan penulis pengalaman untuk dapat berkomunikasi dengan banyak orang, menggali permasalahan yang ada, menyamakan persepsi yang penulis miliki unit keuangan kb & tk permata iman, pemangku kepentingan lain serta menyamakan persepsi dengan kedua dosen pembimbing. Proses penelitian yang dilakukan juga mengajarkan penulis untuk menjadi pribadi yang sabar dan senantiasa berbagi hal-hal bermanfaat yang berkaitan dengan kebutuhan pembelajaran. Penelitian ini tidak dapat diselesaikan tanpa dukungan dan bantuan dari orang-orang terdekat, sehingga saya ingin menyampaikan rasa terima kasih kepada:

1. Bapak Satrio Agung Wicaksono , S.Kom, M.Kom, selaku pembimbing satu yang telah memberikan kesempatan dan waktunya kepada penulis untuk menjadi mahasiswa bimbingan skripsinya dalam melakukan penelitian, serta menjadi pembimbing yang sabar, solutif dan senantiasa memberikan contoh sebagai pribadi yang profesional dalam membimbing topik penyelesaian penelitian penulis.
2. Bapak Mochamad Chandra Saputra, S.Kom., M.Eng. selaku pembimbing dua yang telah memberikan kesempatan, waktunya dan tenaga untuk membimbing penulis dalam menyelesaikan penelitian yang dilakukan, serta senantiasa dengan sabar dan teliti dalam memeriksa pengerjaan dokumen skripsi sampai terselesaikannya penelitian ini.
3. Ibu Fryanne S. Yohani, SE, M.Pd selaku kepala sekolah kelompok bermain dan taman kanak kanak Permata Iman yang telah bersedia memberikan waktunya dan ketersediaannya untuk menjadi narasumber, menguji sistem informasi yang telah dikembangkan serta memberikan masukan-masukannya terhadap pengembangan dari sistem informasi ini.
4. Ibu Asih Sri Winanti selaku bendahara kelompok bermain dan taman kanak kanak Permata Iman yang telah bersedia memberikan waktunya dan ketersediaannya untuk menjadi narasumber, menguji sistem informasi yang telah dikembangkan serta memberikan informasi data data yang di butuhkan terhadap pengembangan dari sistem informasi ini.
5. Ibu saya, Siti Hanifah yang selalu dengan sabar memberikan dukungan berupa doa, dorongan dan semangat agar penulis senantiasa termotivasi untuk menyelesaikan penelitian yang sedang dilakukan. Ayah saya, Zulham yang senantiasa memberikan dukungan materil, masukan, doa, perhatian

dan motivasi sehingga penulis mampu menyelesaikan penelitian ini. Adik saya, Dinda Aulia dan Angga Hafiz yang selalu mengingatkan dan memotivasi penulis dalam menyelesaikan penelitian. Keluarga besar saya, yang memberikan dukungan, motivasi dan nasihat kepada penulis agar tetap melangkah dan tidak kenal menyerah dalam menyelesaikan penelitian skripsi.

6. Raditya Rizky Putra, Fathin Naufal Hawi, Azha Akbar, Yusuf Reyhan Aditiawan, Bangkit Wahyu Saputra, M. Ilham Akbar Junior, Elmanda Krisna Jaya, Erzhal Risan Wikata, Dini Pratiwi, Shafira Viski Izabal, Laras Miranti, Rista Yasin Lamonahende, Ulfatisa Cahyani, dan Intan Rumaysa selaku sahabat penulis yang telah dengan setia, sabar, penuh perhatian mendampingi penulis serta saling menyemangati selama berjalannya perkuliahan, kepengurusan lembaga mahasiswa dan pengerjaan penelitian skripsi. I Gde Yuda, Yhouga Evan dan Masrur Anwar yang telah senantiasa menjadi teman diskusi dalam pengerjaan skripsi dan saling bertukar fikiran berbagi ilmu dalam membangun sistem berbasis *website*.
7. Teman-teman Departemen Informasi dan Komunikasi Lembaga Keluarga Besar Mahasiswa Sistem Informasi dan Eksekutif Mahasiswa Sistem Informasi yang telah menjadi sarana bagi penulis untuk berkembang menjadi pribadi yang solutif, peka terhadap permasalahan di sekitar dan aktif berorganisasi di kalangan masyarakat.
8. Teman-teman mahasiswa Jurusan Sistem Informasi dan jurusan lainnya dari Fakultas Ilmu Komputer yang tidak dapat penulis sebutkan namanya satu per satu, yang telah menjadi teman yang baik bagi penulis, senantiasa memberikan dukungan terhadap sistem informasi yang dikembangkan, dukungan moril, motivasi, sindiran kapan seminar hasil, kapan sidang dan kapan lulus, sehingga mampu terus menyemangati penulis untuk segera menyelesaikan penelitian yang dikerjakan.

Semoga Allah SWT melimpahkan rahmat berlipat ganda kepada seluruh pihak yang telah berkontribusi pada proses pengerjaan penelitian ini. Tidak lupa penulis sampaikan bahwa penelitian yang telah dilakukan memiliki banyak kekurangan, diskusi mengenai penelitian lanjutan, saran dan kritik yang membangun merupakan hal-hal yang penulis harapkan dapat disampaikan kepada penulis.

Malang, 11 Juli 2018

Penulis

muhammad.rizsky.job@gmail.com

ABSTRAK

Kelompok Bermain & Taman Kanak-kanak Permata Iman merupakan sebuah instansi pendidikan swasta yang berdiri sejak tahun 2005 dan berada di kota Malang. Salah satu unit kerja KB & TK Permata Iman yakni bagian keuangan yang dioperasikan oleh bendahara dan dipantau langsung oleh kepala sekolah memiliki alur kerja yang berhubungan dengan pengelolaan data keuangan terkait pencatatan dana, pencairan pengajuan dana, rekap segala jenis pencatatan dan membuat laporan keuangan setiap bulannya. Permasalahan yang terjadi pada proses pengajuan dana terdapat kendala berupa kepala sekolah terkadang tidak berada di sekolah karena adanya kepentingan di luar, sehingga berdampak kepada waktu tunggu yang cukup lama pada alur pengajuan dana dalam verifikasi pengajuan dana. Untuk proses pelaporan keuangan perbulan, bendahara melakukan seluruh rekapan pemasukan dan pengeluaran dana berdasarkan nota-nota yang telah di kumpulkan dan di catat sebelumnya oleh bendahara. Mengakibatkan pengeluaran waktu dan tenaga yang cukup banyak sebelum hasil rekapan dapat di laporkan ke kepala sekolah setiap bulannya. Berdasarkan permasalahan tersebut, dikembangkanlah Sistem Informasi Keuangan yang dapat digunakan stackholder KB & TK Permata Iman untuk mempermudah kegiatan kerja operasional unit keuangan. Sistem informasi keuangan menyediakan fasilitas verifikasi pengajuan dana, pencatatan dana, dan pengiriman laporan keuangan . Metode pengembangan perangkat lunak yang digunakan untuk mengembangkan sistem informasi ini adalah menggunakan waterfall model. Kemudian untuk menguji hasil implementasi akan dilakukan pengujian blackbox dengan metode validation, pengujian penerimaan pengguna dan pengujian compability. Pengujian validasi menghasilkan status valid pada 9 pengujian, penerimaan pengguna menghasilkan 94.6% pada seluruh kriteria dan pengujian compability dapat 2 critical error 2 major dan 7 minor issue.

Kata kunci: Sistem Informasi, Pelaporan, Verifikasi, Pencatatan, Keuangan, Pengiriman email, Website.

ABSTRACT

Play Group & Kindergarten Permata Iman is a private educational institution established in 2005 and located in Malang city. One of the work units of Play Group & Kindergarten Permata Iman is the financial section operated by the treasurer and monitored directly by the principal in relation to management record of fund, disbursement of funds, recap for each recording fund and make monthly financial report. Problems that occur in the funding process when the principal not present in the school because of some errand, causing the long wait for the fund request to disbursement the fund. For the financial recording process, the treasurer conducts all income recaps and disburses funds based on the notes that have been collected and recorded before by the treasurer. Resulting in considerable time and effort before recapture results before it can be reported to the principal each month. Based on these conditions, a Financial *Information System* developed by researchman was developed to facilitate the operational activities of financial units. Financial *information systems* provide funding facilities, recording of funds, and delivery of financial report monthly. Software development method used to develop this *information system* is using waterfall model. Then to test the results will be blackbox testing with method validation testing, user acceptance testing and compability testing. Validation testing result got valid on all 9 test case, user acceptance testing got 94.6% for all criteria, and compability got 2 critical error, 2 major and 7 minor issues.

Keywords: *Information System*, Reporting, Verification, Recording, Financial, Send by Email, Website.

DAFTAR ISI

PENGEMBANGAN SISTEM INFORMASI KEUANGAN (STUDI KASUS : KB & TK PERMATA IMAN MALANG)	i
PENGESAHAN	Error! Bookmark not defined.
PERNYATAAN ORISINALITAS	Error! Bookmark not defined.
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR TABEL.....	xiii
DAFTAR GAMBAR.....	xvii
DAFTAR LAMPIRAN	xix
BAB 1 PENDAHULUAN.....	20
1.1 Latar belakang.....	20
1.2 Rumusan masalah	22
1.3 Tujuan	22
1.4 Manfaat.....	22
1.5 Batasan Masalah	23
1.6 Sistematika Pembahasan	23
BAB 2 LANDASAN KEPUSTAKAAN	25
2.1 Kajian Pustaka	25
2.2 Gambaran KB & TK Permata Iman.....	25
2.2.1 Struktur Organisasi KB & TK Permata Iman	26
2.3 Sistem Informasi	27
2.4 Pengembangan (Development).....	27
2.5 <i>Software Development Life Cycle (SDLC)</i>	28
2.5.1 <i>Waterfall Model</i>	28
2.5.2 Kelebihan dan Kekurangan Model <i>Waterfall</i>	29
2.6 Analisis Kebutuhan	29
2.6.1 Proses Bisnis	29
2.6.2 <i>Business Process (As-Is & To-Be)</i>	30

2.6.3 <i>Bussines Process Model Notation</i>	30
2.6.4 Kebutuhan Pemangku Kepentingan	34
2.6.5 Fitur	36
2.6.6 Kebutuhan Fungsional.....	36
2.6.7 Kebutuhan Non-Fungsional	36
2.7 Perancangan Sistem.....	36
2.7.1 Unified Modeling Language(UML)	37
2.8 Implementasi Sistem	45
2.8.1 Hypertext Preprocessor(PHP)	45
2.8.2 Object Oriented Programming (<i>OOP</i>)	46
2.8.3 Model View Controller (<i>MVC</i>).....	46
2.8.4 CodeIgniter.....	47
2.9 Pengujian Sistem.....	47
2.9.1 Blackbox Testing	47
2.9.2 Pengujian Penerimaan Pengguna	48
2.9.3 <i>Compatibility Testing</i>	49
BAB 3 METODOLOGI	50
3.1 Studi Literatur	51
3.2 Analisis Kebutuhan Sistem.....	51
3.3 Perancangan	51
3.4 Implementasi Sistem	52
3.5 Pengujian Sistem.....	52
3.6 Kesimpulan dan Saran	52
BAB 4 ANALISIS KEBUTUHAN SISTEM	53
4.1 Analisis Proses Bisnis AS-IS dan TO-BE	53
4.1.1 Proses Bisnis As-Is	53
4.1.2 Analisis Permasalahan.....	57
4.1.3 Proses Bisnis To-Be.....	57
4.2 Pemetaan Proses Bisnis	64
4.3 Analisis Kebutuhan Pemangku Kepentingan	65
4.3.1 Tipe Pemangku Kepentingan	65
4.3.2 Identifikasi Aktor	66



4.3.3	Kebutuhan Pemangku Kepentingan dan Aktor	67
4.4	Analisis Spesifikasi Kebutuhan	70
4.4.1	Fitur Sistem	70
4.4.2	Kebutuhan Fungsional.....	72
4.4.3	Kebutuhan Non-Fungsional	75
BAB 5	PERANCANGAN.....	76
5.1	Pemodelan <i>Use case Diagram</i>	76
5.1.1	<i>Use case Scenario</i>	78
5.2	Pemodelan <i>Activity Diagram</i>	94
5.2.1	Membuat <i>Form</i> Pengajuan Dana	94
5.2.2	<i>Activity Diagram</i> Verifikasi Pengajuan Dana.....	95
5.2.3	<i>Activity Diagram</i> Membuat <i>Form</i> Pencatatan Dana.....	96
5.2.4	<i>Activity Diagram</i> Melihat Laporan Keuangan	97
5.2.5	<i>Activity Diagram</i> Mengelola Pengiriman Laporan.....	98
5.3	Perancangan <i>Sequence Diagram</i>	98
5.3.1	<i>Sequence diagram</i> menambah pengajuan dana	99
5.3.2	<i>Sequence diagram</i> verifikasi pengajuan dana	100
5.3.3	<i>Sequence diagram</i> menambah pencatatan dana	101
5.3.4	<i>Sequence diagram</i> melihat laporan	102
5.3.5	<i>Sequence diagram</i> mengelola pengiriman laporan.....	102
5.4	Perancangan <i>Class Diagram</i>	103
5.4.1	<i>Class Diagram</i> Relasi Antar Controller	105
5.4.2	<i>Class Diagram</i> Relasi Antar Model	106
5.4.3	<i>Class Diagram</i> Domain Model	107
5.5	Pemetaan <i>Class Diagram Domain Model</i> ke <i>Physical Data Model</i> ...	107
5.6	Normalisasi <i>Physical Data Model</i>	109
5.7	Perancangan Antar Muka	110
5.7.1	Halaman <i>Log in</i>	111
5.7.2	Halaman Kelola Pengguna	111
5.7.3	<i>Form</i> tambah pengguna	112
5.7.4	Melihat pengajuan dana	113
5.7.5	<i>Form</i> tambah pengajuan dana	113

5.7.6	Form pencatatan dana	114
5.7.7	Laporan keuangan	114
5.8	Perancangan Algoritme	115
5.8.1	Algoritme Menambah Pengajuan Dana.....	115
5.8.2	Algoritme Verifikasi Pengajuan Dana.....	115
5.8.3	Algoritme Pencatatan Dana	116
5.8.4	Algoritme Melihat Laporan Keuangan	116
5.8.5	Algoritme Mengelola Pengiriman Laporan Keuangan	116
5.9	Perancangan Pengujian <i>Black-Box</i> Validasi	117
5.9.1	Pengujian Validasi Membuat <i>Form</i> Pengajuan Dana.....	117
5.9.2	Pengujian Validasi Verifikasi Pengajuan Dana	119
5.9.3	Pengujian Validasi Membuat <i>Form</i> Pencatatan Dana	121
5.9.4	Pengujian Validasi Melihat Laporan Keuangan.....	122
5.9.5	Pengujian Validasi Mengelola Pengiriman Laporan.....	123
5.10	Perancangan Pengujian Penerimaan Pengguna	124
5.11	Perancangan Pengujian Compatibilty.....	131
BAB 6	IMPLEMENTASI SISTEM	133
6.1	Spesifikasi Sistem	133
6.1.1	Spesifikasi Perangkat Lunak	133
6.2	Batasan Implementasi	133
6.3	Implementasi Algoritme	134
6.3.1	Implementasi Menambah Pengajuan Dana.....	134
6.3.2	Implementasi Verifikasi Pengajuan Dana	135
6.3.3	Implementasi Menambah Pencatatan Dana	137
6.3.4	Implementasi Melihat Laporan Keuangan.....	138
6.3.5	Implementasi Mengelola Pengiriman Laporan Keuangan.....	140
6.4	Implementasi Antarmuka Pengguna	141
6.4.1	Antarmuka <i>Login</i>	141
6.4.2	Antarmuka Kelola Pengguna	142
6.4.3	Antarmuka Pengajuan Dana	142
6.4.4	Antarmuka Pencatatan Dana	143
6.4.5	Antarmuka Mengelola Laporan Keuangan	143



BAB 7 PENGUJIAN DAN ANALISIS HASIL	144
7.1 Pengujian <i>Black-box</i>	144
7.1.1 Hasil Pengujian Validasi Sistem Informasi Keuangan	144
7.2 Hasil Pengujian Penerimaan Pengguna	145
7.3 Hasil Pengujian <i>Compability</i>	146
7.4 Analisis Hasil Pengujian.....	147
7.4.1 Analisis Hasil Pengujian Validasi	147
7.4.2 Analisis Hasil Pengujian Penerimaan Pengguna	147
7.4.3 Analisis Hasil Pengujian <i>Compatibilty</i>	147
BAB 8 PENUTUP	148
8.1 Kesimpulan.....	148
8.2 Saran	149
DAFTAR PUSTAKA.....	150
LAMPIRAN A HASIL WAWANCARA KEPALA SEKOLAH	152
A.1 Wawancara Kepala Sekolah I.....	152
A.2 Wawancara Kepala Sekolah II.....	153
LAMPIRAN B HASIL PENGUJIAN PENERIMAAN PENGGUNA.....	154
B.1 Tugas Pengujian Penerimaan Pengguna	154
B.2 Hasil Pengujian Penerimaan Pengguna Kepala Sekolah.....	155
B.3 Hasil Pengujian Penerimaan Pengguna Bendahara.....	157
B.4 Hasil Pengujian Penerimaan Pengguna Peminta Dana	159



DAFTAR TABEL

Tabel 2.1 Tugas dan tanggungjawab <i>stakeholder</i>	26
Tabel 2.2 Template Pernyataan Permasalahan	30
Tabel 2.3 Tipe Event	31
Tabel 2.4 Tipe Activity	32
Tabel 2.5 Tipe Gateway.....	32
Tabel 2.6 Tipe Connections	33
Tabel 2.7 Tipe Swimlanes.....	33
Tabel 2.8 Tipe Artifacts	34
Tabel 2.9 Analisis Tipe Pemangku Kepentingan	35
Tabel 2.10 Peran dan Perwakilan Pemangku Kepentingan	35
Tabel 2.11 Fitur Sistem.....	36
Tabel 2.12 Elemen-Element <i>Use case Diagram</i>	37
Tabel 2.13 Notasi pada <i>activity diagram</i>	39
Tabel 2.14 Simbol-simbol diagram <i>class</i>	41
Tabel 2.15 Notasi pada <i>Sequence Diagram</i>	43
Tabel 2.16 Kriteria pengujian penerimaan pengguna	48
Tabel 4.1 Proses Bisnis Keadaan Saat Ini (<i>As-Is</i>)	54
Tabel 4.2 Analisis Permasalahan.....	57
Tabel 4.3 Penjelasan Aturan Penomoran Kebutuhan.....	58
Tabel 4.4 Daftar Proses Bisnis Usulan (<i>To-Be</i>)	58
Tabel 4.5 Perubahan Aktivitas Proses Bisnis Pengajuan Dana	59
Tabel 4.6 Perubahan Aktivitas Proses Bisnis Pencatatan Dana Beserta	61
Tabel 4.7 Perubahan Aktivitas Proses Bisnis Laporan Keuangan Bulanan Beserta Perubahan Waktu	63
Tabel 4.8 Penjelasan aturan penomoran kode aktivitas	64
Tabel 4.9 Pemetaan Proses Bisnis.....	65
Tabel 4.10 Identifikasi Pemangku Kepentingan.....	66
Tabel 4.11 Identifikasi Aktor	66
Tabel 4.12 Penjelasan aturan penomoran kode kebutuhan	67



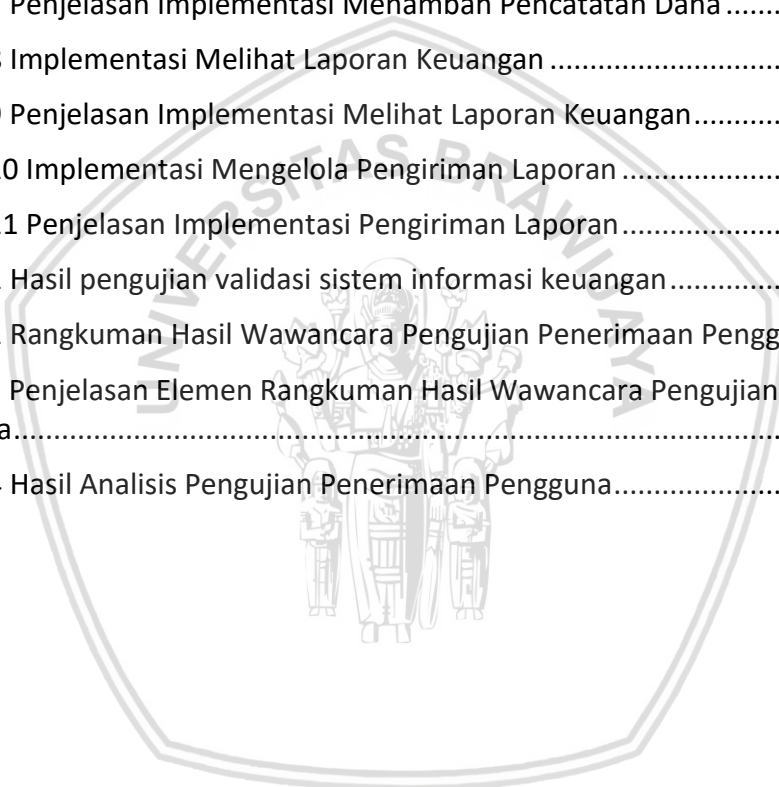
Tabel 4.13	Kebutuhan Pemangku Kepentingan dan Aktor	67
Tabel 4.14	Penjelasan Aturan Penomoran Fitur	70
Tabel 4.15	Fitur Sistem.....	70
Tabel 4.16	Penjelasan Aturan Penomoran Kebutuhan.....	73
Tabel 4.17	Kebutuhan Fungsional.....	73
Tabel 4.18	Kebutuhan Non-Fungsional.....	75
Tabel 5.1	Hubungan <i>Use case Diagram</i>	77
Tabel 5.2	Skenario <i>Use case Log in</i>	78
Tabel 5.3	Skenario <i>Use case Log out</i>	79
Tabel 5.4	Skenario <i>Use case</i> Melihat Pengguna	79
Tabel 5.5	Skenario <i>Use case</i> Menambah Pengguna	79
Tabel 5.6	Skenario <i>Use case</i> Mengubah Pengguna	80
Tabel 5.7	Skenario <i>Use case</i> Menghapus Pengguna.....	81
Tabel 5.8	Skenario <i>Use case</i> Melihat <i>Form</i> Pengajuan Dana.....	82
Tabel 5.9	Skenario <i>Use case</i> Membuat <i>Form</i> Pengajuan Dana	82
Tabel 5.10	Skenario <i>Use case</i> Mengubah <i>Form</i> Pengajuan Dana.....	83
Tabel 5.11	Skenario <i>Use case</i> Menghapus <i>form</i> pengajuan dana	84
Tabel 5.12	Skenario <i>Use case</i> Verifikasi <i>Form</i> Pengajuan Dana	84
Tabel 5.13	Skenario <i>Use case</i> Verifikasi Pencairan Dana.....	85
Tabel 5.14	Skenario <i>Use case</i> Melihat Data Keuangan.....	86
Tabel 5.15	Skenario <i>Use case</i> Membuat <i>Form</i> Pencatatan Dana	86
Tabel 5.16	Skenario <i>Use case</i> Mengubah <i>Form</i> Pencatatan Dana	87
Tabel 5.17	Skenario <i>Use case</i> Menghapus <i>Form</i> Pencatatan Dana.....	88
Tabel 5.18	Skenario <i>Use case</i> Melihat Kategori Keuangan.....	89
Tabel 5.19	Skenario <i>Use case</i> Membuat Kategori Keuangan	89
Tabel 5.20	Skenario <i>Use case</i> Mengubah Kategori Keuangan.....	90
Tabel 5.21	Skenario <i>Use case</i> Menghapus Kategori Keuangan	91
Tabel 5.22	Skenario <i>Use case</i> Melihat Laporan Keuangan	92
Tabel 5.23	Skenario <i>Use case</i> Mengelola Pengiriman Laporan	92
Tabel 5.24	Skenario <i>Use case</i> Mengirim Laporan Keuangan.....	93
Tabel 5.25	Pemetaan <i>Class Diagram Domain Model</i> ke <i>Physical Data Model</i> ...	108
Tabel 5.26	Normalisasi <i>Physical Data Model</i>	109



Tabel 5.27 <i>Pseudocode</i> Menambah Pengajuan Dana.....	115
Tabel 5.28 <i>Pseudocode</i> Verifikasi Pengajuan Dana.....	115
Tabel 5.29 <i>Pseudocode</i> Pencatatan Dana	116
Tabel 5.30 <i>Pseudocode</i> Melihat Laporan Keuangan	116
Tabel 5.31 <i>Pseudocode</i> Mengelola Pengiriman Laporan Keuangan	117
Tabel 5.32 Aturan Penomoran Pengujian Validasi.....	117
Tabel 5.33 Matriks Skenario Pengujian Validasi Menambah <i>Form</i> Pengajuan Dana	118
Tabel 5.34 Rencana Pengujian dan Kasus Uji <i>Black-Box</i> Fungsi Menambah <i>Form</i> Pengajuan Dana	118
Tabel 5.35 Rencana Pengujian dan Kasus Uji <i>Black-Box</i> Fungsi Menambah <i>Form</i> Pengajuan Dana <i>Data Tidak Lengkap</i>	118
Tabel 5.36 Matriks Skenario Pengujian Validasi Verifikasi Pengajuan Dana	119
Tabel 5.37 Rencana Pengujian dan Kasus Uji Validasi Fungsi Verifikasi Pengajuan Dana	119
Tabel 5.38 Rencana Pengujian dan Kasus Uji Validasi Fungsi Menambah <i>Form</i> Pengajuan Dana <i>Data Tidak Lengkap</i>	120
Tabel 5.39 Matriks Skenario Pengujian Validasi Membuat <i>Form</i> Pencatatan Dana	121
Tabel 5.40 Rencana Pengujian dan Kasus Uji Validasi Fungsi Membuat <i>Form</i> Pencatatan Dana	121
Tabel 5.41 Rencana Pengujian dan Kasus Uji Validasi Fungsi Menambah <i>Form</i> Pengajuan Dana <i>Data Tidak Lengkap</i>	122
Tabel 5.42 Rencana Pengujian dan Kasus Uji Validasi Melihat Laporan Keuangan	122
Tabel 5.43 Matriks Skenario Pengujian Validasi Mengelola Pengiriman Laporan Keuangan.....	123
Tabel 5.44 Rencana Pengujian dan Kasus Uji Validasi Mengelola Pengiriman Laporan Keuangan.....	123
Tabel 5.45 Rencana Pengujian dan Kasus Uji Validasi Mengelola Pengiriman Laporan Keuangan <i>Form</i> tidak lengkap.....	124
Tabel 5.46 Stakeholder yang terlibat pada pengujian penerimaan pengguna...	124
Tabel 5.47 Aturan Penomoran Pengujian Validasi.....	125
Tabel 5.48 Tugas Pengujian Penerimaan pengguna	125
Tabel 5.49 Tabel Pertanyaan Pengujian Penerimaan Pengguna	129



Tabel 5.50 Rencana Pengujian dan Kasus Uji <i>Compability</i> menggunakan <i>Sortsite</i>	132
Tabel 6.1 Spesifikasi Perangkat Lunak	133
Tabel 6.2 Implementasi Menambah Pengajuan Dana	134
Tabel 6.3 Penjelasan Implementasi Menambah Pengajuan Dana	135
Tabel 6.4 Implementasi Verifikasi Pengajuan Dana	136
Tabel 6.5 Penjelasan Implementasi Verifikasi Pengajuan Dana	136
Tabel 6.6 Implementasi Menambah Pencatatan Dana	137
Tabel 6.7 Penjelasan Implementasi Menambah Pencatatan Dana	138
Tabel 6.8 Implementasi Melihat Laporan Keuangan	139
Tabel 6.9 Penjelasan Implementasi Melihat Laporan Keuangan	139
Tabel 6.10 Implementasi Mengelola Pengiriman Laporan	140
Tabel 6.11 Penjelasan Implementasi Pengiriman Laporan	140
Tabel 7.1 Hasil pengujian validasi sistem informasi keuangan	144
Tabel 7.2 Rangkuman Hasil Wawancara Pengujian Penerimaan Pengguna	146
Tabel 7.3 Penjelasan Elemen Rangkuman Hasil Wawancara Pengujian Penerimaan Pengguna	146
Tabel 7.4 Hasil Analisis Pengujian Penerimaan Pengguna	147



DAFTAR GAMBAR

gambar 2.1 Logo KB dan TK Permata Iman.....	25
Gambar 2.2 Struktur Organisasi KB dan TK Permata Iman.....	26
Gambar 2.3 Waterfall Model Pressman (2012)	28
Gambar 2.4 Contoh <i>Use case</i> Diagram	38
Gambar 2.5 Contoh Class diagram.....	41
Gambar 2.6 Contoh Diagram Physical Data Model (sumber : images.visual-paradigm.com).....	42
Gambar 2.7 Contoh penulisan kode PHP diantara HTML.....	46
Gambar 2.8 Model-View-Controller (Pitt, 2012)	46
Gambar 3.1 Diagram Alur Metode Penelitian	50
Gambar 4.1 Diagram BPMN Pengajuan Dana.....	55
Gambar 4.2 Diagram BPMN pemasukan atau pengeluaran dana.....	56
Gambar 4.3 Diagram BPMN Pelaporan Keuangan Bulanan	56
Gambar 4.4 Aturan Penomoran Proses Bisnis.....	57
Gambar 4.5 Diagram BPMN Proses Bisnis <i>To-Be</i> Pengajuan Dana.....	58
Gambar 4.6 Diagram BPMN <i>To-Be</i> Proses Bisnis Pencatatan Pemasukan atau Pengeluaran Dana	61
Gambar 4.7 Diagram BPMN <i>To-Be</i> Proses Bisnis Laporan Keuangan Bulanan.....	62
Gambar 4.8 Aturan Penomoran Aktivitas Proses Bisnis	64
Gambar 4.9 Aturan Penomoran Kebutuhan	67
Gambar 4.10 Aturan Penomoran Fitur	70
Gambar 4.11 Aturan Penomoran Kebutuhan Fungsional dan Non-Fungsional ...	72
Gambar 5.1 <i>Use case</i> Diagram Sistem Informasi Keuangan	76
Gambar 5.2 Activity Diagram Membuat <i>Form</i> Pengajuan Dana.....	94
Gambar 5.3 Activity Diagram Verifikasi Pengajuan Dana	95
Gambar 5.4 Activity Diagram Membuat <i>Form</i> Pencatatan Dana	96
Gambar 5.5 Activity Diagram Melihat Laporan Keuangan	97
Gambar 5.6 Activity Diagram Pengiriman Laporan.....	98
Gambar 5.7 <i>Sequence diagram</i> membuat <i>form</i> pengajuan dana	99
Gambar 5.8 <i>Sequence Diagram</i> verifikasi pengajuan dana	100



Gambar 5.9 <i>Sequence diagram</i> menambah pencatatan dana	101
Gambar 5.10 <i>Sequence diagram</i> melihat laporan	102
Gambar 5.11 <i>sequence diagram</i> Mengelola Pengiriman Laporan	103
Gambar 5.12 Diagram Sistem Informasi Keuangan	104
Gambar 5.13 Class Diagram Relasi Antar Controller	105
Gambar 5.14 Rancangan <i>Class Diagram Domain Model</i>	107
Gambar 5.15 <i>Physical Data Model</i> sistem informasi keuangan	108
Gambar 5.16 Normalisasi <i>Physical Data Model</i> sistem informasi keuangan	110
Gambar 5.17 Rancangan Antar Muka Login	111
Gambar 5.18 Rancangan antar muka kelola pengguna	111
Gambar 5.19 Rancangan antar muka <i>form</i> tambah pengguna	112
Gambar 5.20 Rancangan antar muka menu pengajuan dana	113
Gambar 5.21 Rancangan Antarmuka <i>Form tambah pengajuan dana</i>	113
Gambar 5.22 Rancangan Antarmuka <i>Form</i> Pencatatan Dana	114
Gambar 5.23 Rancangan Antarmuka Laporan Keuangan	114
Gambar 5.24 Aturan Penomoran Pengujian Validasi	117
Gambar 5.25 Aturan Penomoran Pengujian Penerimaan Pengguna	125
Gambar 5.26 Simbol Tingkat Kompabilitas pada Compatibility Testing SortSite	131
Gambar 6.1 Hasil implementasi <i>login</i>	141
Gambar 6.2 Hasil implementasi antarmuka kelola pengguna	142
Gambar 6.3 Hasil implementasi antarmuka pengajuan dana	142
Gambar 6.4 Antarmuka Pencatatan Dana	143
Gambar 6.5 Antarmuka Laporan Keuangan	143
Gambar 7.1 Hasil Pengujian <i>Compatibilty</i> Menggunakan <i>SortSite Trial</i> versi 5.29.818.0	146



DAFTAR LAMPIRAN

LAMPIRAN A HASIL WAWANCARA KEPALA SEKOLAH	152
A.1 Wawancara Kepala Sekolah I	152
A.2 Wawancara Kepala Sekolah II	153
LAMPIRAN B HASIL PENGUJIAN PENERIMAAN PENGGUNA.....	154
B.1 Tugas Pengujian Penerimaan Pengguna.....	154
B.2 Hasil Pengujian Penerimaan Pengguna Kepala Sekolah	155
B.3 Hasil Pengujian Penerimaan Pengguna Bendahara.....	157
B.4 Hasil Pengujian Penerimaan Pengguna Peminta Dana.....	159



BAB 1 PENDAHULUAN

1.1 Latar belakang

Kelompok Bermain & Taman Kanak-kanak Permata Iman merupakan sebuah instansi pendidikan swasta yang berdiri sejak tahun 2005 dan berada di kota Malang. KB & TK Permata Iman bergerak di bidang pendidikan anak usia dini (PAUD). Saat ini KB & TK Permata Iman memiliki 3 buah cabang yang tersebar di kota Malang. Masing-masing berada di daerah Tlogomas, Soekarno Hatta, dan Sukun. Saat ini memiliki total 360 murid tersebar di masing-masing cabang yaitu Tlogomas sebanyak 120 murid, Soekarno Hatta sebanyak 120 murid, dan Sukun sebanyak 120 murid yang dibagi dari tingkat kelompok bermain hingga taman kanak-kanak. sehingga dapat terlihat bahwa proses bisnis yang terjadi didalamnya kompleks. Selain itu, KB & TK Permata Iman adalah instansi KB & TK pertama di Indonesia yang memperoleh sertifikasi standar internasional sistem manajemen mutu ISO 9001:2000.

KB & TK Permata Iman merupakan instansi pendidikan memiliki beberapa unit kerja dalam struktur organisasinya agar instansi dapat berjalan dengan semestinya. Salah satu unit kerja yakni bagian keuangan yang di operasikan oleh bendahara dan dipantau langsung oleh kepala sekolah. Unit kerja keuangan memiliki alur kerja yang berhubungan dengan pengelolaan data keuangan terkait pencatatan dana, pencairan dana, pengajuan dana, rekap segala jenis pencatatan dan membuat laporan keuangan setiap bulannya.

Ibu Fryanne SY, SE, M.Pd selaku Kepala KB dan TK Permata Iman, mengatakan pada hasil data wawancara yang dilakukan, proses bisnis keuangan yang terjadi di KB dan TK Permata Iman dibagi menjadi tiga permasalahan utama yaitu pengajuan dana, pencatatan alur pemasukan dan pengeluaran dan pelaporan keuangan perbulan. Pengajuan dana masih menggunakan *form* manual dan harus melaporkannya kepada kepala sekolah untuk diverifikasi oleh kepala sekolah yang pada nantinya dapat dilanjutkan ke bendahara instansi untuk proses pencairan dana. Permasalahan yang terjadi pada proses pengajuan dana terdapat kendala berupa kepala sekolah terkadang tidak berada di sekolah karena adanya kepentingan di luar, sehingga berdampak kepada waktu tunggu yang cukup lama pada alur pengajuan dana khususnya proses verifikasi pengajuan dana. Pencatatan dana masih menggunakan buku manual dan tidak terarsip dengan baik. Proses pelaporan keuangan perbulan memiliki permasalahan lamanya melakukan seluruh rekap pemasukan dan pengeluaran dana yang telah di catat sebelumnya oleh bendahara pada buku keuangan yang berdampak pada pengeluaran waktu dan tenaga yang cukup banyak sebelum laporan keuangan dapat di laporkan ke kepala sekolah setiap bulannya.

Sistem informasi mempunyai bagian vital dalam meningkatkan bisnis dan membantu proses bisnis dalam peningkatan efisiensi dan efektifitas, mengatur pilihan, dan bekerja sama dalam group(O'Brien & Marakas, 2010). Dengan adanya

sistem informasi dapat membantu dan mendukung kegiatan operasional suatu organisasi dan mempermudah penyediaan laporan, mempercepat kegiatan bisnis yang di perlukan oleh organisasi tersebut. Oleh karena itu KB &TK Permata Iman membutuhkan sebuah aplikasi sistem informasi yang dapat membantu dan mendukung serta menyelesaikan masalah yang ada pada kegiatan operasional unit keuangannya.

Software Development Life Cycle (SDLC) adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (berdasarkan *best practice* atau cara-cara yang sudah teruji baik (A.S. & Shalahuddin, 2014). Penulis akan mengembangkan sistem dengan bantuan *SDLC* agar proses pengembangan menjadi terstruktur dan dapat mempermudah penulis dalam pengembangan sistem informasi. *Unified Modelling Language* adalah keluarga notasi grafis yang di dukung oleh meta-model tunggal, yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek (OO) (Fowler, 2005). Tahap modelling atau perancangan memuat kebutuhan sistem ke dalam bentuk yang lebih spesifik dengan pendekatan UML berupa *use case diagram*, *activity diagram*, *sequence diagram*, *physical data model diagram* serta *class diagram* agar mempermudah pengembang dalam pengembangan sistemnya. Pengujian menggunakan *Blackbox* dengan metode *validation testing* agar dapat mengetahui kesesuaian sistem yang telah di implementasikan dengan kebutuhan yang sudah di tentukan di awal pengembangan. Pengujian *User Acceptance* agar dapat mengukur dan memberikan kesesuaian apakah sistem dapat bekerja dengan baik sesuai dengan keinginan pengguna. *Compability testing* untuk mengetahui apakah sistem sudah dapat di gunakan pada browser dan sistem operasi yang berbeda.

Menurut penjelasan di atas, Penelitian ini bermaksud melakukan pengembangan sistem informasi berjudul "**Pengembangan Sistem Informasi Keuangan (Studi Kasus : KB & TK Permata Iman)**". Dengan adanya sistem informasi manajemen keuangan di institut tersebut, diharapkan dapat membantu institut dalam mengatasi problematika operasional unit keuangan yang terjadi pada KB & TK Permata Iman.

1.2 Rumusan masalah

Pertanyaan penelitian:

1. Bagaimana hasil dari analisis proses bisnis dan kebutuhan sistem informasi keuangan KB & TK Permata Iman?
2. Bagaimana hasil dari perancangan sistem dan pengujian berdasarkan analisis kebutuhan sistem informasi keuangan KB & TK Permata Iman ?
3. Bagaimana hasil implementasi dari pengembangan sistem informasi keuangan KB & TK Permata Iman?
4. Bagaimana hasil dari pengujian sistem informasi keuangan KB & TK Permata Iman?

1.3 Tujuan

Tujuan Umum:

Mengembangkan sistem informasi yang dapat membantu institute pendidikan KB dan TK Permata Iman dalam proses manajemen keuangan, bisa membantu dalam pelaporan keuangan, pengajuan dana serta rekapitan keuangan perbulan.

Tujuan Khusus:

1. Menganalisis proses bisnis yang telah ada dan membuat proses bisnis baru yang siap di implementasikan ke dalam sistem informasi keuangan pada KB & TK Permata Iman.
2. Melakukan rancangan sistem informasi manajemen KB & TK Permata Iman.
3. Menghasilkan implementasi sistem informasi keuangan KB & TK Permata Iman.
4. Melakukan pengujian validasi, pengujian penerimaan pengguna dan pengujian *compability* pada hasil implementasi sistem informasi keuangan KB & TK Permata Iman.

1.4 Manfaat

Manfaat yang diharapkan dari penelitian ini adalah sebagai berikut:

1. Bagi penulis, meningkatkan pengetahuan dan wawasan mengenai perancangan dan Pengembangan sistem informasi.
2. Bagi institut pendidikan KB & TK Permata Iman, dapat meningkatkan efektifitas dan efisiensi terkhusus pada proses bisnis keuangan KB & TK Permata Iman.
3. Bagi kepala sekolah, pelaporan keuangan menjadi lebih detil dan membantu dalam pengambilan keputusan.

1.5 Batasan Masalah

Berdasarkan latar belakang yang telah penulis jabarkan sistem informasi yang akan dikembangkan memiliki batasan masalah yaitu, pengembangan sistem yang dilakukan berfokus pada kegiatan operasional unit keuangan di institut pendidikan KB & TK Permata Iman. Batasan selanjutnya dari penelitian yang di kembangkan adalah dilakukannya pengembangan berdasarkan tahap pada *waterfall model* dengan meniadakan tahap *planning*.

1.6 Sistematika Pembahasan

Sistematika penulisan dari skripsi ini terdiri dari beberapa bagian utama sebagai berikut:

BAB I Pendahuluan

Pada bab ini memuat latar belakang, rumusan masalah, tujuan, batasan masalah, manfaat, metodologi pembahasan dan sistematika penulisan penelitian skripsi.

BAB II Landasan Kepustakaan

Pada bab ini menguraikan tentang kajian pustaka dan dasar teori yang mendasari serta mendukung pengembangan sistem informasi keuangan

BAB III Metodologi Penelitian

Pada bab ini membahas dan menguraikan langkah-langkah yang akan digunakan dalam penelitian terkait penulisan dan metode yang digunakan.

BAB IV Analisis Kebutuhan

Pada bab ini membahas kebutuhan dari sistem yang akan di kembangkan.

BAB V Perancangan

Pada bab ini membahas perancangan dari sistem yan akan dibangun dan rancangan pengujian.

BAB VI Implementasi Sistem

Membahas mengenai hasil implementasi sistem informasi keuangan yang sesuai dengan perancangan yang telah dibuat.

BAB VII Pengujian Sistem

Memuat proses hasil pengujian dan analisis sistem yang telah di implementasikan dan membandingkan dengan rancangan dan tujuan awal sistem.

BAB VIII Kesimpulan dan Saran

Memuat kesimpulan dan saran yang diperoleh dari pembuatan dan pengujian sistem serta saran untuk pengembangan lebih lanjut dari sistem informasi keuangan.



BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Pada bagian ini penulis mengkaji bahwa perlunya mempelajari studi terkait penelitian yang memiliki kesamaan pada penelitian yang telah di lakukan sebelumnya. Kajian ini di harapkan dapat membantu penulis dalam melakukan penelitian kedepannya. Windana(2014) merujuk kepada penelitiannya yang berjudul “Pengembangan Sistem Informasi Akuntansi Penjualan Dan Penerimaan Kas Berbasis Komputer Pada Perusahaan Kecil (Studi Kasus Pada Pt. Trust Technology)”. Langkah yang di ambil penulis dengan mengembangkan sistem akuntansi dengan langkah siklus hidup pengembangan sistem.Terdapat enam langkah siklus hidup pengembangan sistem yaitu : Perencanaan Sistem, Analisis Sistem, Perancangan Sistem secara Umum/Konseptual, Evaluasi dan Seleksi Sistem, Fase Perancangan Sistem secara Detail/Fungsional, Implementasi Sistem dan Pemeliharaan Sistem. Penelitian bertujuan untuk mengembangkan sistem informasi akuntansi penjualan dan penerimaan kas pada PT. Trust Technology dari sistem manual menjadi sistem berbasis komputer.

Penelitian selanjutnya yang di lakukan oleh Tenardi(2013) yang berjudul “Sistem Informasi Keuangan pada Sekolah ST.Agatha”. Peneliti melakukan perancangan Sistem Informasi Keuangan. Hasil dari analisis dan perancangan sistem ini, dapat mempermudah pihak tata usaha dalam melakukan pembayaran dan laporan keuangan sekolah, sehingga dapat meningkatkan efektivitas dan efisiensi dari masing-masing karyawan sekolah.

2.2 Gambaran KB & TK Permata Iman



gambar 2.1 Logo KB dan TK Permata Iman

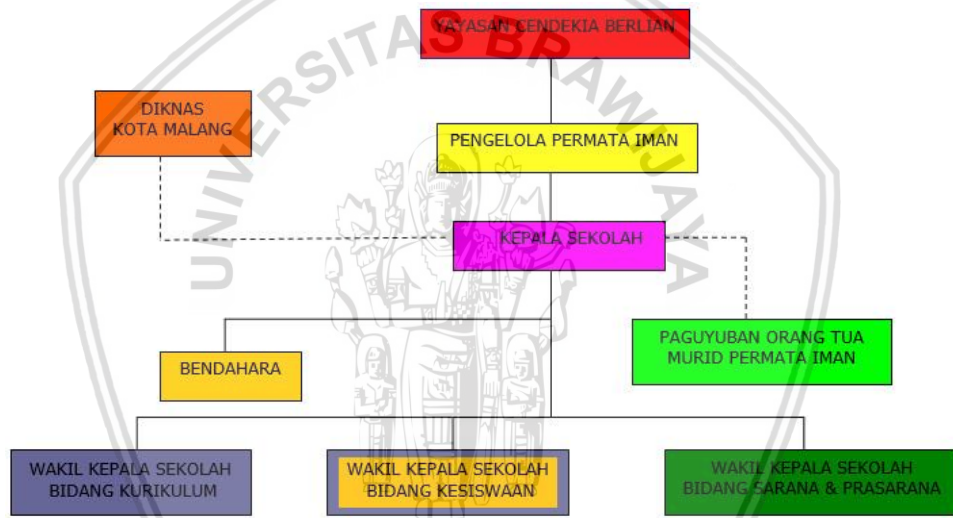
Gambar 2.1 merupakan logo KB dan TK Permata Iman. KB dan TK Permata Iman merupakan instansi yang bergerak dibidang jasa pendidikan dasar yaitu Kelompok Bermain (KB) dan Taman Kanak-kanak (TK) yang berada di kota Malang. Instansi pendidikan yang beralamat di Jalan Batu Permata No. 9 Tlogomas Malang tersebut didirikan pada 31 Mei 2005.

KB dan TK Permata Iman merupakan TK pertama dikota Malang yang mendapatkan ISO, yaitu ISO 9001:2000. ISO 9001:2000 adalah suatu standar internasional untuk sistem manajemen kualitas. ISO 9001:2000 menetapkan

persyaratan-persyaratan dan rekomendasi untuk desain dan penilaian dari suatu sistem manajemen mutu, yang bertujuan untuk menjamin bahwa organisasi akan memberikan produk yang dapat menjamin kepuasan pelanggan.

2.2.1 Struktur Organisasi KB & TK Permata Iman

KB dan TK Permata Iman mempunyai 120 murid pada tahun 2016 yang terbagi ke dalam 4 tingkatan yaitu KB A, KB B, TK A dan TK B yang ditangani oleh 18 orang guru. KB dan TK Permata Iman memiliki 6 unit Internal yaitu Akademik dan Kesiswaan, Kurikulum, HUMAS, Sarana Prasarana, Bendahara Sekolah, dan Kepala Sekolah yang mempunyai tugas masing-masing dalam menjalankan proses bisnis yang ada di KB dan TK Permata Iman. KB dan TK Permata Iman juga mempunyai hubungan dengan organisasi eksternal yang juga berperan dalam proses bisnis yang ada pada KB dan TK Permata Iman yaitu Paguyuban Orang Tua, Pengelola Permata Iman, Yayasan, dan DIKNAS Kota Malang. Struktur organisasi dijelaskan pada Gambar 2.2 Struktur Organisasi KB & TK Permata Iman.



Gambar 2.2 Struktur Organisasi KB dan TK Permata Iman

Pada Tabel 2.1 berikut dijelaskan tentang tugas dan tanggungjawab stakeholder yang ada pada KB dan TK Permata Iman.

Tabel 2.1 Tugas dan tanggungjawab stakeholder

No.	Stakeholder	Tugas dan Tanggungjawab
1.	Kepala Sekolah	<ol style="list-style-type: none"> 1. Melakukan cek kelas setiap hari 2. Melakukan monitoring dan evaluasi Guru 3. Mengadakan rapat koordinasi Kepala Sekolah 4. Membuat laporan bulanan tentang jumlah guru dan murid 5. Melakukan penggajian karyawan sekolah 6. Melakukan perekrutan karyawan KB dan TK Permata Iman.



Tabel 2.1 Tugas dan tanggungjawab *stakeholder*

No.	Stakeholder	Tugas dan Tanggungjawab
		7. Bertanggungjawab atas kegiatan yang ada di KB dan TK Permata Iman.
2.	Bendahara	1. Melakukan pencatatan pemasukan dan pengeluaran keuangan. 2. Mengelola uang kas sekolah. 3. Melakukan rekap keuangan secara berkala setiap bulan ke kepala sekolah.
3.	Akademik dan Kesiswaan	1. Mendata murid setiap awal tahun ajaran baru 2. Melakukan perbaruan data murid setiap bulan 3. Melakukan tracking kelulusan dari setiap murid pada akhir tahun ajaran 4. Melakukan surat menyurat tentang kegiatan murid Koordinasi kegiatan non akademik murid
4.	Kurikulum	1. Koordinasi penyusunan kurikulum, dari mulai program tahunan, semester, mingguan dan harian 2. Membuat <i>format</i> laporan evaluasi belajar Melakukan koordinasi try out untuk masuk sekolah dasar
5.	Sarana Prasarana	1. Melakukan cek saran belajar secara berkala 2. Mengajukan pengadaan kebutuhan sarana belajar Mengajukan program pemeliharaan sarana belajar

2.3 Sistem Informasi

Sistem informasi merupakan suatu kombinasi dari setiap unit yang dikelola oleh user atau manusia, hardware (perangkat keras), software (perangkat lunak), jaringan komputer dan jaringan komunikasi data (komunikasi), dan juga *database* (basis data) yang mengumpulkan, mengubah, dan menyebarkan informasi tentang suatu organisasi. Jadi, pada dasarnya, sistem informasi memang harus memiliki elemen – elemen tersebut agar dapat berguna dan juga bekerja dengan optimal (O'Brien & Marakas, 2008). Tiga alasan penting yang dilakukan sistem informasi bagi sebuah perusahaan adalah (O'Brien & Marakas, 2010 hal 8):

1. Mendukung proses dan kegiatan operasi bisnis.
2. Mendukung pengambilan keputusan oleh karyawan dan menejer.
3. Mendukung strategi bagi keuntungan kompetitif.

2.4 Pengembangan (Development)

Mengonversi desain ke sistem informasi yang lengkap termasuk bagaimana memperoleh dan melakukan instalasi lingkungan sistem yang dibutuhkan; membuat basis data dan mempersiapkan prosedur pengujian; mempersiapkan

berkas atau *file* pengujian, pengodean, pengompilasian, memperbaiki dan membersihkan program; peninjauan pengujian (A.S. & Shalahuddin, 2016).

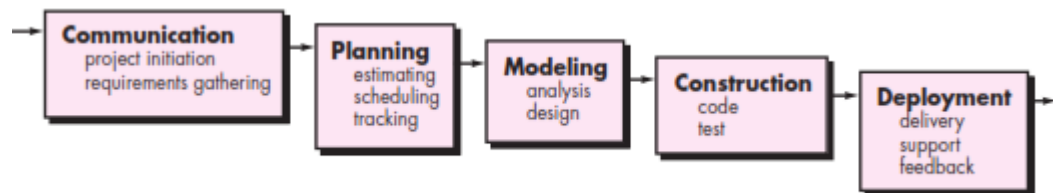
2.5 Software Development Life Cycle (SDLC)

Software Development Life Cycle (SDLC) adalah proses mengembangkan atau mengubah suatu sistem perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan sistem-sistem perangkat lunak sebelumnya (berdasarkan *best practice* atau cara-cara yang sudah teruji baik). Tahapan-tahapan yang ada pada SDLC secara global adalah sebagai berikut (A.S. & Shalahuddin, 2016:26):

1. Inisiasi (*initiation*)
2. Pengembangan konsep sistem (*system concept development*)
3. Perencanaan (*planning*)
4. Analisis kebutuhan (*requirements analysis*)
5. Desain (*design*)
6. Pengembangan (*development*)
7. Integrasi dan pengujian (*integration and test*)
8. Implementasi (*implementation*)
9. Operasi dan Pemeliharaan (*operation and maintenance*)
10. Disposisi (*disposition*)

2.5.1 Waterfall Model

Waterfall model menurut Pressman (2012). Metode air terjun atau yang sering disebut metode waterfall sering dinamakan siklus hidup klasik (*classic life cycle*), dimana hal ini menggambarkan pendekatan yang sistematis dan juga berurutan pada pengembangan perangkat lunak, dimulai dengan spesifikasi kebutuhan pengguna lalu berlanjut melalui tahapan-tahapan perencanaan (*planning*), permodelan (*modeling*), konstruksi (*construction*), serta penyerahan sistem ke para pelanggan/pengguna (*deployment*), yang diakhiri dengan dukungan pada perangkat lunak lengkap yang dihasilkan (Pressman, 2010). Gambar alur dijelaskan pada Gambar 2.3 di bawah ini :



Gambar 2.3 Waterfall Model Pressman (2012)

Gambar 2.2 merupakan *model pressman* yang akan di adaptasi pada penelitian terutama pada bagian *communication, modelling, construction dan deployment*.

2.5.2 Kelebihan dan Kekurangan Model *Waterfall*

Kelebihan dari menggunakan model *waterfall* diantaranya adalah sebagai berikut:

1. Kualitas dari sistem yang dihasilkan sangat baik. Hal ini dikarenakan oleh pelaksanaannya yang dilakukan secara bertahap. Sehingga tidak terfokus pada tahapan tertentu.
1. Metode ini masih lebih baik digunakan walaupun sudah tergolong kuno, daripada menggunakan pendekatan asal-asalan. Selain itu, metode ini juga masih masuk akal jika kebutuhan sudah diketahui dengan baik.
2. Dokumen pengembangan sistem terorganisir dengan baik, karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya. Jadi setiap fase atau tahapan akan mempunyai dokumen tertentu.

Kelemahan dari menggunakan *waterfall* diantaranya adalah sebagai berikut:

1. Diperlukan majemen yang baik, karena proses pengembangan tidak dapat dilakukan secara berulang sebelum terjadinya suatu produk.
2. Kesalahan kecil segera menjadi masalah besar jika tidak diketahui sejak awal pengembangan yang berakibat pada tahapan selanjutnya. Pemangku kepentingan dan pengguna baru dapat mengetahui wujud dari sistem informasi yang dibangun setelah memasuki tahap akhir.

Pendekatan yang digunakan pada penelitian pengembangan sistem informasi keuangan ini menggunakan model *waterfall*. Alasan penulis menggunakan model *waterfall* karena kebutuhan atau persyaratan dari perangkat lunak yang ingin dibangun sudah dapat didefinisikan secara utuh, eksplisit dan benar atau mudah dipahami dengan baik di awal pengembangan pada tahap analisis kebutuhan sistem. Definisi produk stabil dan tidak ada perubahan saat pengembangan untuk alasan apapun seperti perubahan eksternal, perubahan tujuan atau perubahan teknologi.

2.6 Analisis Kebutuhan

Analisis kebutuhan merupakan bagian dari proses kebutuhan perangkat lunak yang berperan menjembatani jurang yang sering terjadi antara level rekayasa kebutuhan dan perancangan perangkat lunak.(Pressman, 2008)

2.6.1 Proses Bisnis

Proses bisnis adalah serangkaian aktivitas yang membutuhkan sebuah (atau lebih) *input* dan menghasilkan *output* yang memiliki nilai bagi *customer* (Monk & Wagner, 2013). Monk dan Wagner menambahkan, *customer* sebuah proses bisnis terdiri dari *customer* eksternal dan *customer* internal. *Customer* eksternal merupakan seseorang atau sekelompok orang yang membeli produk, sedangkan *customer* internal merupakan seseorang atau sekelompok orang dalam berbagai macam departemen dalam perusahaan.

2.6.2 Business Process (As-Is & To-Be)

Analisis dan pemodelan proses bisnis *as-is* adalah kegiatan untuk mengumpulkan dan memodelkan proses bisnis yang aktual pada sebuah organisasi atau perusahaan yang bertujuan untuk menyajikan informasi mengenai situasi aktual perusahaan, untuk mengetahui kekurangan dari proses bisnis, dan dapat memunculkan kemungkinan peningkatan dalam proses bisnis yang ada pada sebuah organisasi atau perusahaan (Becker, et al., 2013).

Proses analisis *as-is* dilakukan untuk mengetahui permasalahan dan kelemahan yang ada pada proses bisnis tersebut. Permasalahan tersebut dipetakan dalam tabel analisis permasalahan, sehingga bisa diketahui bagaimana solusi dan perbaikan untuk proses bisnis tersebut. Pada Tabel 2.2 merupakan cara dalam melakukan analisis permasalahan.

Tabel 2.2 Template Pernyataan Permasalahan

<i>Problem Statement Template</i>	
<i>The problem of</i>	<i>[describe the problem]</i>
<i>Affects</i>	<i>[the stakeholders affected by the problem]</i>
<i>The impact of which is</i>	<i>[what is the impact of the problem?]</i>
<i>A successful solution would</i>	<i>[list some key benefits of a successful solution]</i>

sumber: *Use case Modeling* by Kurt Bittner and Ian Spence (2002)

Analisis dan pemodelan proses bisnis *to-be* merupakan kegiatan untuk mengidentifikasi dan memodelkan proses bisnis yang telah mengalami peningkatan (*improvement*) berdasarkan analisis terhadap proses bisnis *as-is* (Becker, et al., 2013).

2.6.3 Business Process Model Notation

Business Process Model and Notation (BPMN) merupakan standar untuk pemodelan proses bisnis yang memberikan notasi grafis untuk menentukan proses bisnis dalam Business Process Diagram (BPD), 2 berdasarkan teknik flowcharting tradisional. Tujuan dari BPMN adalah untuk mendukung pemodelan proses bisnis untuk kedua teknis pengguna dan pengguna bisnis, dengan memberikan notasi yang intuitif bagi pengguna bisnis, belum mampu mewakili semantik proses kompleks (Rosing, Scheel & Scheer, 2012). *Business Process Modelling Notation* (BPMN) adalah bentuk pemodelan untuk menentukan bagaimana proses yang terjadi dalam alur bisnis dan siapa saja aktor yang terlibat dalam sebuah alur bisnis tersebut. BPMN memiliki lima kategori elemen sebagai penunjang dalam pembuatan BPMN (Object Management Group, 2011). Lima kategori elemen tersebut adalah :




2.6.3.1 Flow Object

Flow Object dibagi menjadi 3, yaitu *event*, *activity*, dan *gateway*. Berikut penjelasannya :

1. Event

Event merupakan sesuatu yang terjadi dan memiliki dampak dalam proses bisnis. Suatu *event* dapat berasal dari internal dan eksternal suatu proses. *Event* dibagi menjadi tiga yaitu *start event*, *intermediate event*, dan *end event*. Setiap proses selalu memiliki sebuah *start event* untuk menunjukkan awal dari proses bisnis. Tabel 2.3 merupakan penjelasan dari tipe *event*.

Tabel 2.3 Tipe Event


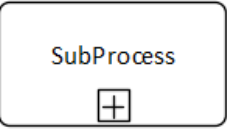
No	Tipe	Deskripsi	Simbol
1	<i>Start</i>	Mendeskripsikan dimana suatu proses dimulai	
2	<i>Intermediate</i>	Mendeskripsikan dimana suatu terjadi diantara awal dan akhir proses. Akan mempengaruhi alur dari proses, tapi tidak akan memulai atau memberhentikan proses.	
3	<i>End</i>	Mendeskripsikan suatu proses berakhir.	

Sumber : Object Management Group (2011)

2. Activity

Activity merupakan tugas yang dilakukan dalam sebuah proses bisnis. *Activity* ditunjukkan dengan kotak dengan ujung bulat dengan nama yang menjelaskan aktivitas yang dilakukan. Terdapat dua macam *activity* yaitu *task* dan *sub process*. Tabel 2.3 Merupakan penjelasan dari tipe *activity*.

Tabel 2.4 Tipe Activity





No	Tipe	Deskripsi	Simbol
1	<i>Task</i>	Aktivitas yang dilakukan pada alur proses	
2	<i>Sub Process</i>	Aktivitas majemuk yang dimasukkan dalam proses. Aktivitas majemuk tersebut dapat dijelaskan dengan lebih detail.	

Sumber : Object Management Group (2011)

3. Gateway

Gateway bertanggung jawab mengontrol bagaimana alur dari sebuah proses bisnis. Tabel 2.5 merupakan penjelasan dari tipe *gateway*.

Tabel 2.5 Tipe Gateway

No	Tipe	Deskripsi	Simbol
1	<i>Exclusive</i>	digunakan untuk membuat jalur alternatif dalam sebuah proses, tapi hanya satu yang dipilih. digunakan untuk menggabungkan jalur alternatif.	
2	<i>Parallel</i>	Mendeskripsikan proses yang dijalankan secara bersamaan	
3	<i>Inclusive</i>	Mendeskripsikan sebuah proses yang dipecah menjadi beberapa jalur.	
4	<i>Complex</i>	Mendeskripsikan alur yang kompleks pada sebuah proses bisnis	




Sumber : Object Management Group (2011)

2.6.3.2 Connections

Connections adalah elemen yang menghubungkan *flow objects*. Tabel 2.6 merupakan penjelasan dari tipe *connections*.



Tabel 2.6 Tipe Connections



No	Tipe	Deskripsi	Simbol
1	<i>Sequence Flow</i>	Menunjukkan urutan aktivitas yang dilakukan pada sebuah proses	
2	<i>Association</i>	Menunjukkan hubungan antara data, teks, artifak lain, dan <i>flow object</i> pada sebuah proses	
3	<i>Message Flow</i>	Menunjukkan alur pesan antara dua partisipan yang mampu mengirim dan menerima pesan	

Sumber : Object Management Group (2011)

2.6.3.3 *Swimlanes*

Swimlanes merupakan wadah grafis yang membagi suatu set aktivitas dengan aktivitas lain. Tabel 2.7 merupakan penjelasan tipe *swimlanes*.

Tabel 2.7 Tipe *Swimlanes*

No	Tipe	Deskripsi	Simbol
1	<i>Pool</i>	Merupakan wadah yang berisi satu proses dan <i>sequence flow</i> yang menghubungkan aktivitas	
2	<i>Lane</i>	Digunakan untuk mempresentasikan tanggungjawab aktivitas pada sebuah proses.	

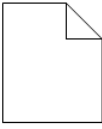



Sumber : Object Management Group (2011)

2.6.3.4 *Artifacts*

Artifacts mempresentasikan sebuah objek diluar sebuah proses. *Artifact* dapat mempresentasikan data atau catatan yang menjelaskan sebuah proses atau dapat digunakan untuk mengelola tugas atau proses. Tabel 2.8 merupakan penjelasan tipe *artifacts*.



Tabel 2.8 Tipe Artifacts

No	Tipe	Deskripsi	Simbol
1	<i>Data Object</i>	Mempresentasikan informasi yang mengalir pada sebuah proses seperti dokumen bisnis, surat, <i>email</i> dan lain-lain.	
2	<i>Data Store</i>	Tempat dimana proses dapat membaca atau menulis data	
3	<i>Annotation</i>	Menunjukkan informasi tambahan kepada pembaca sebuah diagram BPMN.	
4	<i>Group</i>	Memungkinkan untuk mengelompokkan elemen secara <i>informal</i>	

Sumber : Object Management Group (2011)

2.6.4 Kebutuhan Pemangku Kepentingan

Bittner & Spence (2002) menyatakan bahwa pemangku kepentingan adalah siapapun yang berdampak secara material pada *output* sebuah sistem (hasil produk) atau proyek dalam memproduksi sebuah sistem. Identifikasi kebutuhan dari pemangku kepentingan dilakukan untuk mengetahui bagaimana hubungan antara aktor dengan masalah-masalah yang ada sebelum diterapkannya aplikasi atau sistem. Dengan mengetahui siapa saja pemangku kepentingan yang akan menggunakan sistem tersebut, pengembang aplikasi bisa menentukan fungsi dan fitur apa saja yang ada dalam aplikasi dengan menyesuaikan karakteristik dari pemangku kepentingan tersebut, dimana pemangku kepentingan bertindak sebagai *user* atau pengguna dari sistem. Langkah-langkah yang dilakukan dalam analisis kebutuhan pemangku kepentingan adalah :

1. Tipe Pemangku Kepentingan.

Tipe Pemangku kepentingan merupakan kumpulan pemangku kepentingan yang memiliki karakteristik dan hubungan yang sama dengan sistem atau aplikasi yang dikembangkan. Pada Tabel 2.9 merupakan cara dalam melakukan analisis tipe pemangku kepentingan.



Tabel 2.9 Analisis Tipe Pemangku Kepentingan

Tipe Pemangku Kepentingan	Deskripsi	Pemangku Kepentingan
[Tipe pemangku kepentingan]	[Menjelaskan tipe pemangku kepentingan]	[Menyebutkan pemangku kepentingan yang termasuk dalam tipe pemangku kepentingan]

Sumber: Diadaptasi dari Bittner & Spence (2002)

2. Peran dan Perwakilan Pemangku Kepentingan

Peran pemangku kepentingan merupakan kumpulan pemangku kepentingan yang memiliki peran dan tanggung jawab yang sama pada pengembangan aplikasi. Tujuannya adalah untuk mengetahui peran dan tanggung jawab masing-masing pemangku kepentingan sehingga menghindari kesalahan komunikasi dalam pelaksanaan proyek.

Pemangku kepentingan sangat penting dalam proyek karena pemangku kepentingan nantinya pemangku kepentingan yang akan menggunakan secara langsung sistem yang telah dibuat. Dengan mengetahui peran pemangku kepentingan, dapat diambil perwakilan dari pemangku kepentingan tersebut. Tabel 2.9 merupakan cara dalam melakukan analisis peran dan perwakilan pemangku kepentingan.

Tabel 2.10 Peran dan Perwakilan Pemangku Kepentingan

Peran Pemangku Kepentingan	Perwakilan Pemangku Kepentingan	Deskripsi
[Peran pemangku kepentingan]	[Perwakilan pemangku kepentingan]	[Menjelaskan peran pemangku kepentingan]

Sumber: Diadaptasi dari Bittner & Spence (2002)

3. Peran Pengguna

Peran pengguna merupakan peran yang dijalankan oleh pengguna saat berinteraksi dengan fungsi fungsi yang ada di dalam sistem. Peran pengguna juga dapat merepresentasikan sistem lain yang dapat melakukan interaksi dengan sistem.

4. Kebutuhan Pemangku Kepentingan dan Pengguna

Kebutuhan pemangku kepentingan menggambarkan masalah dalam proses bisnis atau pribadi dari pemangku kepentingan yang perlu untuk dilakukan analisis bagaimana solusi yang dibutuhkan untuk mendukung tujuan bisnis dan menyelesaikan masalah yang ada. Identifikasi kebutuhan pemangku kepentingan akan memberikan pandangan mengenai masalah utama dan solusi yang dapat menyelesaikan masalah tersebut (Bittner & Spence, 2002).

2.6.5 Fitur

Fitur merupakan suatu kemampuan sistem yang menyediakan nilai bagi pengguna serta memenuhi kebutuhan pemangku kepentingan dan pengguna. Suatu fitur dapat memiliki beberapa kebutuhan fungsional (Bittner & Spence, 2002). Fitur Tabel 2.10 merupakan cara dalam menjelaskan fitur perangkat lunak.

Tabel 2.11 Fitur Sistem

Kode Fitur	Fitur	Deskripsi
[Kode suatu fitur, sebagai tiap fitur]	[Fitur sistem]	[Deskripsi dari fitur]

Sumber : Bittner & Spence (2002)

2.6.6 Kebutuhan Fungsional

Kebutuhan fungsional dari sistem menggambarkan apa dan bagaimana sistem yang harus dilakukan oleh sistem. Kebutuhan ini bergantung pada tipe aplikasi yang akan dikembangkan, apa yang diinginkan oleh pengguna dari aplikasi yang dikembangkan, dan bagaimana pendekatan dari pemangku kepentingan ketika menentukan kebutuhan aplikasi. Ketika menjelaskan kebutuhan dari pemangku kepentingan atau pengguna, Kebutuhan fungsional adalah salah satu cara untuk menjelaskan fungsi yang diberikan kepada pengguna yang bisa dipahami (Sommerville, 2011).

2.6.7 Kebutuhan Non-Fungsional

Kebutuhan non-fungsional adalah kebutuhan yang tidak secara langsung berkaitan dengan layanan yang diberikan oleh sistem kepada pengguna. Kebutuhan ini berkaitan dengan keandalan sistem, waktu proses dari sistem, dan keamanan sistem. Kebutuhan non fungsional juga menjelaskan bagaimana sebuah sistem dikembangkan sesuai dengan keadaan atau kondisi eksternal dari sistem seperti kebutuhan sistem untuk merespon perangkat *input output*, bagaimana sistem bisa menampilkan data yang representatif dan sesuai dengan keinginan pengguna, dan bagaimana sistem bisa menyesuaikan dengan sistem lain (Sommerville, 2011).

2.7 Perancangan Sistem

Menurut Pressman (2010), “perancangan adalah langkah pertama dalam fase pengembangan rekayasa produk atau sistem. Perancangan itu adalah proses penerapan berbagai teknik dan prinsip yang bertujuan untuk mendefinisikan sebuah peralatan, satu proses atau satu sistem secara detail yang membolehkan dilakukan realisasi fisik”.

2.7.1 Unified Modeling Language(UML)

Unified Modelling Language adalah keluarga notasi grafis yang di dukung oleh meta-model tunggal, yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang di bangun menggunakan pemrograman berorientasi objek (OO)(Fowler, 2005).

Unified Modelling Language(UML) adalah kerangka kerja yang menjadi standar untuk digunakan dalam pembuatan *blueprint* dari perangkat lunak. UML dapat digunakan untuk memvisualisasikan, menjabarkan spesifikasi, melakukan konstruksi, dan membuat dokumentasi artifak dari sistem perangkat lunak (Pressman, 2010). Diagram yang digunakan pada UML:


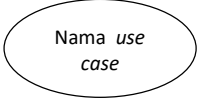
1. *Use case Diagram*
2. *Activity Diagram*
3. *Class Diagram*
4. *Physical Data Model Diagram*
5. *Sequence Diagram*

2.7.1.1 Use case Diagram

Use case adalah teknik untuk merekam persyaratan fungsional sebuah sistem, mendeskripsikan interaksi tipikal antara para pengguna sistem dengan sistem itu sendiri dengan memberi narasi bagaimana sistem tersebut digunakan(Fowler, 2005). *Use case* mempresentasikan kebutuhan fungsional ke dalam konteks daripada pengguna melakukan sesuatu yang berguna(Bittner dan Spence, 2002).

Pada Tabel 2.12 merupakan tabel penjelasan elemen *use case diagram* di bawah ini :

Tabel 2.12 Elemen-Elemen Use case Diagram

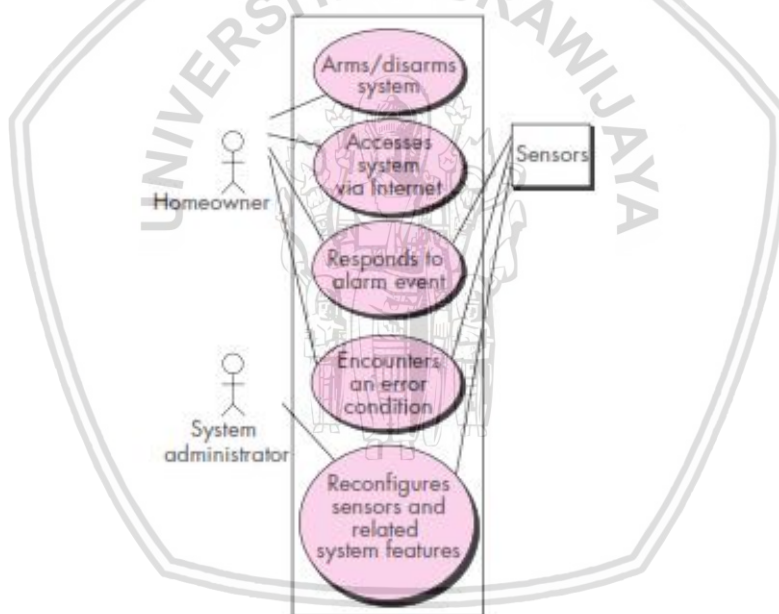
Elemen	Keterangan	Notasi
Aktor	Orang, proses, atau sistem yang dapat berinteraksi dengan sistem informasi yang akan dibuat. Walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.	 Nama aktor
Use case	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal di awal frase nama use case.	 Nama use case
Association Relationship	Komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor.	—————



Tabel 2.12 Elemen-Elemen *Use case Diagram*

Elemen	Keterangan	Notasi
<i>Include Relationship</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> , dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.	<<include>> ----->
<i>Extend Relationship</i>	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan tersebut.	<<extend>> ----->
<i>Generalization Relationship</i>	Untuk memperlihatkan bahwa beberapa aktor atau <i>use case</i> memiliki sesuatu hal yang bersifat umum.	—————>

Sumber: Shalahuddin (2016)



Gambar 2.4 Contoh *Use case Diagram*

Sumber : Roger S. Pressman 2010

Gambar 2.4 merupakan contoh dari *use case diagram*. Pada gambar tersebut memiliki elemen elemen seperti aktor ataupun yang di lambangkan dengan *stickman*(orang) lalu ada panah yang menggambarkan asosiasi yaitu hubungan elemen satu dengan yang lain. ada pula *balloon text* yang menggambarkan aktifitas yang di lakukan oleh aktor.



2.7.1.2 Use case Scenario


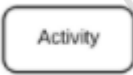


Model yang menggambarkan persyaratan fungsional suatu sistem atau klarifikasi lainnya dalam *use case*. Model *use case* menyajikan sebuah sistem dalam hal penggunaannya. Bila terjemahkan secara *formal*, model *use case* menjelaskan semua kemungkinan cara menggunakan sistem (Bittner dan Spence, 2002). Adapun properti yang harus di miliki oleh suatu *use case* skenario adalah :

1. *Use case* Name merupakan nama *use case*
2. *Use case* Description merupakan penjelasan singkat terkait tujuan *use case*
3. Actor merupakan pelaku yang berinteraksi dengan *use case*
4. Pre-Condition merupakan kondisi yang harus terpenuhi.
5. Basic Flow of Events urutan aktifitas *use case* pada umumnya.
6. Alternate Course urutan aktifitas alternatif (jika ada).
7. Post-Condition kondisi setelah *use case* berhasil dijalankan.

2.7.1.3 Activity Diagram




Activity diagram adalah teknik untuk menggambarkan logika prosedural, proses bisnis, dan jalur kerja. Dalam beberapa 21 hal, *activity diagram* memainkan peran mirip diagram alir, tetapi perbedaan prinsip antara notasi diagram alir adalah *activity diagram* mendukung *behavior paralel*. Node pada sebuah *activity diagram* disebut sebagai *action*, sehingga diagram tersebut menampilkan sebuah aktifitas yang tersusun dari action (Fowler, 2005:163) Notasi tersebut dijelaskan pada Tabel 2.13.

Tabel 2.13 Notasi pada *activity diagram*

No.	Notasi	Nama	Deskripsi
1		<i>Start</i>	Merupakan awal dari sebuah proses atau alur kerja pada <i>activity diagram</i> .
2		<i>Activity</i>	Menunjukkan kegiatan yang membentuk suatu proses. Simbol ini mencakup deskripsi singkat yang berada di dalam <i>shape</i> .
3		<i>Connector</i>	Menunjukkan aliran arah atau aliran kontrol dari aktivitas. Tanda panah masuk memulai langkah aktivitas, ketika langkah selesai, aliran berlanjut dengan panah keluar.
4		<i>Decision</i>	Memiliki setidaknya dua jalur yang bercabang dengan teks kondisi agar pengguna dapat melihat opsi. Simbol ini mewakili <i>branching</i> atau <i>merging</i>



Tabel 2.13 Notasi pada *activity diagram*

No.	Notasi	Nama	Deskripsi
			berbagai arus dengan simbol yang bertindak sebagai <i>container</i> .
5	[Condition]	<i>Condition</i>	Ditempatkan di sebelah <i>decision symbol</i> untuk memberi tahu pada kondisi apa aliran <i>activity</i> harus terpecah ke arah itu.
6		<i>Join</i>	Menggabungkan dua aktivitas bersamaan dan meletakkan kembali ke aliran dimana hanya satu aktivitas yang terjadi pada satu waktu. Diwakili dengan garis vertikal atau horizontal yang tebal.
7		<i>Fork</i>	Membagi aktivitas tunggal menjadi dua aktivitas bersamaan. Dilambangkan dengan beberapa baris panah dari join.
8		<i>End</i>	Menandai keadaan akhir sebuah aktivitas dan mewakili penyelesaian dari semua alur kerja.

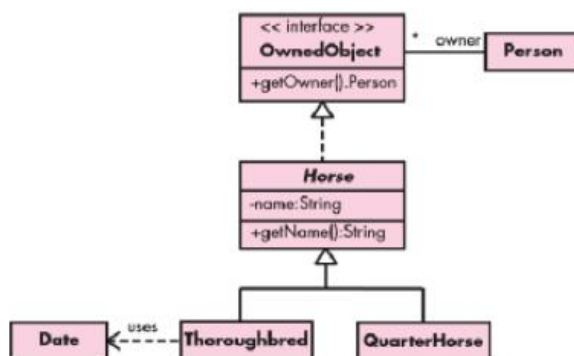
Sumber: Lucidchart

Tabel 2.13 merupakan contoh notasi artefak *activity diagram*. Start point digunakan untuk menunjukkan awalmulanya sebuah proses dan end point untuk berakhir. Activities berisikan aktifitas yang dilakukan oleh aktor, fork (percabangan) digunakan jika ada satu aktifitas yang mengarah kepada 2 aktifitas yang berbeda dan 2 aktifitas berjalan secara parallel. Join digunakan untuk jika 2 pekerjaan yang parallel menuju 1 aktifitas. *Decision* menunjukkan jika ada 2 aktifitas berdasarkan sebuah situasi (*if-else*). *Swimlane* mengemlompokkan *activity* berdasarkan aktor.

2.7.1.4 Class Diagram

Class diagram menurut Fowler (2005) merupakan pendeskripsian jenis-jenis objek di dalam sistem dan berbagai macam hubungan statis di antara mereka. *Class diagram* juga menjelaskan properti dan operasi sebuah *class* dan batasan-batasan yang terdapat dalam hubungan-hubungan objek tersebut.





Gambar 2.5 Contoh Class diagram

Sumber: Pressman (2010)

Dapat dilihat pada Gambar 2.5 memiliki banyak class. Dan untuk setiap classnya tingkat pertama digunakan dalam mendefinisikan nama dari kelas pada suatu sistem. Tingkat kedua merupakan atribut yang berisikan atribut-atribut yang dimiliki oleh suatu kelas. Dalam mendefinisikan atribut disertakan pula tipe data setiap atribut. Tingkat ketiga merupakan operasi atau fungsi, digunakan untuk mendefinisikan fungsi apa saja yang dapat dilakukan kelas tersebut. Atribut dan operasi memiliki tipe hak akses yang dinotasikan sebagai -, #, ~, + yang secara urut mewakili *private*, *protected*, *package*, dan *public*. Terdapat penghubung di antara class yang di sebut dengan asosiasi. Tabel 2.14 menjelaskan tentang simbol-simbol yang digunakan dalam pembuatan *class diagram* pada umumnya.

Tabel 2.14 Simbol-simbol diagram class

Nama	Simbol	Deskripsi
Class		Class pada suatu struktur sistem.
Asosiasi / association		Relasi antar class dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
Ketergantungan / dependency		Ketergantungan antar class.
Agregasi / aggregation		Relasi antar class dengan makna semua-bagian (<i>whole-part</i>).



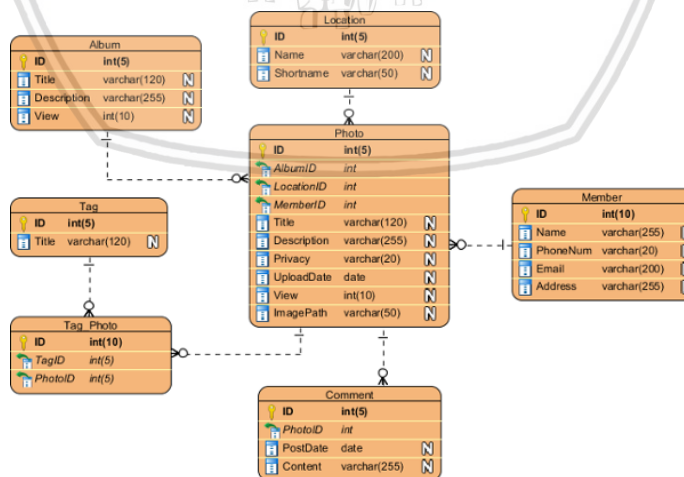
Tabel 2.14 Simbol-simbol diagram *class*

Nama	Simbol	Deskripsi
Asosiasi berarah / <i>directed association</i>		Relasi antar <i>class</i> dengan makna <i>class</i> yang satu digunakan oleh <i>class</i> yang lain.
Generalisasi		Relasi antar <i>class</i> dengan makna <i>class</i> yang satu digunakan oleh <i>class</i> yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .

Sumber : Booch (2005)

2.7.1.5 Physical Data Model Diagram

Merupakan model yang menggunakan sejumlah tabel untuk menggambarkan data serta hubungan antara data-data tersebut (Tutorial Pemrograman: 2009). Setiap tabel mempunyai sejumlah kolom di mana setiap kolom memiliki nama yang unik. Untuk menentukan kebutuhan-kebutuhan suatu sistem *database*. Menurut (A.S. & Shalahuddin, 2016), model relasional atau *Physical Data Model* (PDM) adalah model yang menggunakan sejumlah table untuk menggambarkan data serta hubungan antara data. Setiap table mempunyai sejumlah kolom di mana setiap kolom memiliki nama yang unim beserta tipe datanya. PDM merupakan konsep yang menerangkan detail dari bagaimana data di simpan di dalam basis data. PDM sudah merupakan bentuk fisik perancangan basis data yang sudah siap diimplementasikan ke dalam DBMS sehingga nama table juga sudah merupakan nama asli table yang diimplementasikan ke dalam DBMS. Gambar 2.6 Berikut merupakan contoh *physical data model diagram*



Gambar 2.6 Contoh Diagram Physical Data Model (sumber : images.visual-paradigm.com)

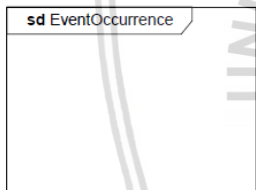
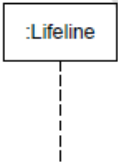
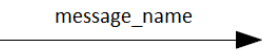


2.7.1.6 Sequence Diagram

Sebuah sequence diagram, secara khusus, menjabarkan *behavior* sebuah skenario tunggal. *Diagram* tersebut menunjukkan sejumlah objek contoh dan pesan-pesan yang melewati objek-objek ini di dalam *use case* (Fowler, 2005). Oleh karena itu untuk menggambarkan diagram sequence maka harus diketahui objek-objek yang terlibat didalam *use case* beserta metode-metode yang dimiliki. Para pengembang sistem banyak sekali menggunakan metode-metode untuk mempermudah pengangan sistem yang berjalan, seperti misalnya *Waterfall*.

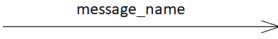
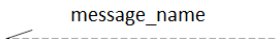
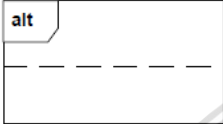
Membuat diagram sequence juga dibutuhkan untuk melihat skenario yang ada pada *use case*. Banyaknya diagram sequence yang harus digambar adalah minimal sebanyak pendefinisian *use case* yang memiliki proses sendiri atau yang penting semua *use case* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada *diagram sequence* sehingga semakin banyak *use case* yang didefinisikan maka *diagram sequence* yang harus dibuat harus semakin banyak. Untuk simbol-simbol yang digunakan dalam diagram sequence adalah dapat di lihat pada Tabel 2.15 berikut:

Tabel 2.15 Notasi pada Sequence Diagram

Notasi	Nama	Deskripsi
	<i>Frame</i>	Notasi yang berisi sekumpulan interaksi. Didalam notasi ini terdapat serangkaian (lanjutan) yang saling bertukar pesan sebagai visualisasi aspek dinamis bagian dari sistem. Nama interaksi dituliskan di dalam bangun segi lima di pojok kiri atas notasi <i>frame</i> .
	<i>Lifeline</i>	Notasi ini merupakan representasi sebuah <i>instances</i> yang terlibat dalam sebuah interaksi pertukaran pesan (<i>message</i>).
	<i>Message (call)</i>	Notasi yang merepresentasikan komunikasi yang bersifat <i>synchronous</i> di antara <i>lifeline</i> . Notasi <i>call</i> secara khusus merepresentasikan pemanggilan operasi atau metode. Jika notasi ini digunakan untuk memodelkan pemanggilan operasi dengan argumen, maka argumen tersebut

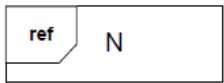


Tabel 2.15 Notasi pada *Sequence Diagram*

Notasi	Nama	Deskripsi
		harus disertakan dalam penamaan notasi.
	<i>Message (asynchronous)</i>	Notasi yang merepresentasikan komunikasi yang bersifat <i>asynchronous</i> di antara <i>lifeline</i> .
	<i>Message (reply)</i>	Notasi yang merepresentasikan pesan balasan dari penggunaan notasi <i>message call</i> atau <i>asynchronous</i> .
	<i>Combined Fragment (Fragment)</i>	<p>Notasi yang digunakan untuk mengelompokkan interaksi ke dalam fragmen. Setiap fragmen dapat memiliki beberapa <i>operand</i> yang masing-masing ditempatkan pada jalur yang dipisahkan oleh garis putus-putus di dalam fragmen. Masing-masing <i>operand</i> dapat memiliki sebuah <i>guard</i>, yang menandakan sebuah kondisi yang dipenuhi ketika <i>operand</i> tersebut dijalankan. Notasi ini memiliki beberapa varian yang ditandai dengan penamaan <i>interaction operator</i> di dalam bangun segi lima di pojok kiri atas notasi ini.</p> <p>Varian <i>alternative (alt)</i> merupakan varian yang merepresentasikan interaksi kondisional. Pada varian ini setidaknya 1 <i>operand</i> akan dijalankan.</p> <p>Varian <i>option (opt)</i> merupakan varian yang merepresentasikan interaksi kondisional. Varian ini hanya boleh memiliki 1 <i>operand</i>, yang bisa saja tidak dijalankan apabila kondisi pada <i>guard</i> tidak terpenuhi.</p> <p>Varian <i>loop</i> merupakan varian yang merepresentasikan interaksi yang</p>



Tabel 2.15 Notasi pada *Sequence Diagram*

Notasi	Nama	Deskripsi
		dilakukan berulang-ulang selama kondisi <i>operand</i> terpenuhi. Varian ini hanya boleh memiliki 1 <i>operand</i> .
	<i>Interaction Use (Fragment)</i>	Notasi yang digunakan untuk merujuk sebuah interaksi. Pemodelan notasi ini menggunakan notasi <i>combined fragment</i> dengan nama <i>interaction operator ref</i> . Nama interaksi yang dirujuk diletakkan di tengah notasi.

Sumber: Diadaptasi dari Object Management Group (2005)

2.8 Implementasi Sistem

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain (A.S. & Shalahuddin, 2014).

2.8.1 Hypertext Preprocessor(PHP)

PHP(*Hypertext Preprocessor*) merupakan bahasa pemrograman yang kuat yang dapat dijalankan dengan baris *code* pada semua komputer yang telah terinstall PHP. Tetapi, PHP tak dapat berdiri sendiri. Untuk menggunakan PHP dibutuhkan server yang memproses *script* PHP (Hansen & Lengstorf, 2014). Penggunaan bahasa pemrograman PHP pada tahap pengembangan aplikasi berbasis *web* diawali dengan tag pembuka (`<?php`) dan diakhiri dengan tag penutup (`?>`). Penggunaan PHP dapat dilakukan secara bersamaan dengan penggunaan bahasa HTML(*Hyper Text Markup Language*). Contoh penggunaan PHP yang berada diantara elemen-elemen HTML dapat dilihat pada gambar 2.7 di bawah ini :

```

<!DOCTYPE HTML>
<html>
  <head>
    <title>Example</title>
  </head>
  <body>
    <?php
      echo "Hi, I'm a PHP script!";
    ?>
  </body>
</html>

```

Gambar 2.7 Contoh penulisan kode PHP diantara HTML

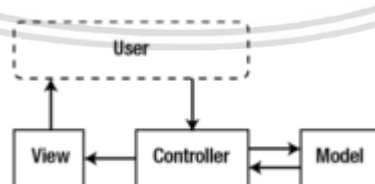
Dapat di lihat pada Gambar 2.7 terdapat tag <!DOCTYPE HTML> dan pada tag <body> terdapat tag yang di gunakan untuk inisiasi Bahasa pemrograman php dengan tag <?php dan di akhiri dengan ?>.

2.8.2 Object Oriented Programming (OOP)

Object Oriented Programming merupakan salah satu strategi pembanguna perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadapnya. Metodologi berorientasi objek merupakan suatu cara bagaimana sistem perangkat lunak dibangun melalui pendekatan objek secara sistematis. Metode berorientasi objek didasarkan pada penerapan prinsip-prinsip pengelolaan kompleksitas. Metode berorientasi objek meliputi rangkaian aktivitas analisis berorientasi objek, perancangan berorientasi objek, pemrograman berorientasi objek, dan pengujian berorientasi objek (A.S. & Shalahuddin, 2016).

2.8.3 Model View Controller (MVC)

MVC merupakan pola desain perangkat lunak yang dibangun di dalam 3 komponen utama, pada bahasa pemrograman seperti PHP yang sering berfokus pada paradigma *object oriented programming* (OOP). Ketiga komponen tersebut adalah *model*, *view*, dan *controller*. Pada Gambar 2.8 dapat dilihat hubungan antara ketiga komponen yaitu *Model*, *View*, dan *Controller* (Pitt, 2012).



Gambar 2.8 Model-View-Controller (Pitt, 2012)

Pada gambar 2.8 terdapat hubungan antara view, controller dan model. *Model* merupakan bagian dimana semua algoritme pada aplikasi disimpan. Algoritme dapat berupa semua hal yang secara spesifik menjelaskan tentang bagaimana aplikasi dapat menyimpan data, atau menggunakan layanan pihak ketiga untuk memenuhi kebutuhan jalannya aplikasi. Jika aplikasi diharuskan untuk mengakses

informasi dari database, maka *code* untuk menjalankan perintah itu terdapat di *model*.

View merupakan bagian dimana *user interface* sebuah aplikasi disimpan. Didalamnya terdapat file HTML, CSS *Style Sheets*, dan *JavaScript*. Semua hal yang dilihat atau berinteraksi dengan pengguna disimpan di dalam *view*, dan terkadang yang dilihat oleh pengguna merupakan kombinasi dari berbagai *view* dalam waktu pemanggilan yang bersamaan.

Controller merupakan bagian yang menghubungkan *model* dan *view*. *Controller* menyatukan logika fungsi yang ada pada *model* dan tampilannya diambil dari *view*, dan mengatasi bagaimana aplikasi akan merespon permintaan interaksi dalam *view*. *Controller* merupakan poin pertama yang menangani masukan diantara ketiga komponen ini, karena permintaan pertama kali akan dilewati pada *controller*, dimana nantinya akan di sambungkan kepada kebutuhan *model* dan *view* untuk memenuhi permintaan kepada aplikasi.

2.8.4 CodeIgniter

CodeIgniter merupakan *framework* bagi pengguna yang membangun suatu aplikasi berbasis *web* menggunakan PHP. Tujuannya adalah memungkinkan pengguna membangun suatu proyek menjadi lebih cepat jika dibandingkan apabila pengguna harus memasukkan *code* dari awal, dengan mempersembahkan berbagai *library* untuk kebutuhan umum yang biasa dipakai, dan juga antarmuka yang sederhana dan penggunaan dalam menggunakan *library*. *CodeIgniter* memungkinkan pengguna agar secara kreatif fokus terhadap proyeknya dengan meminimalisir baris *code* yang diperlukan dalam menjalankan suatu pekerjaan (Ellislab, 2006).

2.9 Pengujian Sistem

Pengujian perangkat lunak berisi kegiatan yang bertujuan untuk mencari dan mengevaluasi kesalahan atau *error* yang dapat ditemukan pada suatu perangkat lunak. Sasaran dari *testing* adalah untuk mencari *error*, sehingga pengujian yang baik adalah pengujian yang memiliki peluang yang besar untuk menemukan sebuah *error* (Pressman, 2010).

2.9.1 Blackbox Testing

Pengujian *blackbox*, juga disebut pengujian perilaku yang memfokuskan kebutuhan fungsional pada perangkat lunak, tanpa memperhatikan *source code* yang ada di dalam perangkat lunak. Pada pengujian *blackbox*, pengembang perangkat lunak memiliki seperangkat kondisi input yang siap dijalankan pada fungsional perangkat lunak yang akan diuji. Dengan menerapkan *blackbox*, pengujian perangkat lunak menyiapkan seperangkat kasus uji yang digunakan untuk mengetahui keberadaan *error* dari *class* yang ada. Pengujian *blackbox* bertujuan untuk menemukan kesalahan pada perangkat lunak (Pressman, 2010):

1. fungsi-fungsi yang salah atau hilang,
2. kesalahan antarmuka,
3. kesalahan struktur data,
4. kesalahan perilaku atau, dan
5. kesalahan inisialisasi atau terminasi.

2.9.2 Pengujian Penerimaan Pengguna

Pengujian penerimaan pengguna adalah pengujian untuk mengukur kualitas produk dengan menentukan apakah ekspektasi pengguna sudah memenuhi terhadap sistem yang di kembangkan (Naik & Kshirasagar, 2008).

Kriteria user acceptance testing yang digunakan menyesuaikan dengan kebutuhan spesifikasi pengguna yaitu, *functional correctness and completeness*, *confidentiality and availability*, dan *usability*. Pengujian akan dilaksanakan dengan menghampiri *tester* (penguji) untuk melakukan pengujian terhadap sistem yang telah dibuat (Naik & Kshirasagar, 2008). Kriteria tersebut dijelaskan pada Tabel 2.15.

Tabel 2.16 Kriteria pengujian penerimaan pengguna

No	Kriteria	Definisi
1	<i>Functional Correctness and Completeness</i>	Pengujian untuk mengetahui apakah sistem yang dibuat sudah sesuai dengan kebutuhan yang diinginkan pengguna dan semua fitur yang dijelaskan dalam spesifikasi persyaratan harus ada dalam sistem yang dikirimkan.
2	<i>Usability</i>	Evaluasi produk atau layanan dengan mengujinya untuk mengetahui sistem dapat dengan mudah digunakan dan kepuasan pengguna. (lanjutan)
3	<i>Confidentiality and Availability</i>	Confidentiality adalah sebuah aturan yang memiliki kemampuan untuk membatasi hak akses terhadap informasi tertentu. Availability adalah jaminan akses terhadap data yang dapat dipercaya berkaitan dengan informasi yang dipegang oleh orang-orang yang berwenang.

Sumber: Naik & Kshirasagar [2008]

2.9.2.1 Skala Guttman

Proses penilaian pada pengujian penerimaan pengguna menggunakan skala guttman. Skala Guttman, skala pengukuran dengan tipe ini akan didapat jawaban yang tegas yaitu : benar-salah, pernah-tidak pernah, ya-tidak. Skala ini dapat dibuat dengan bentuk centang maupun pilihan ganda (Sugiyono, 2012,140).

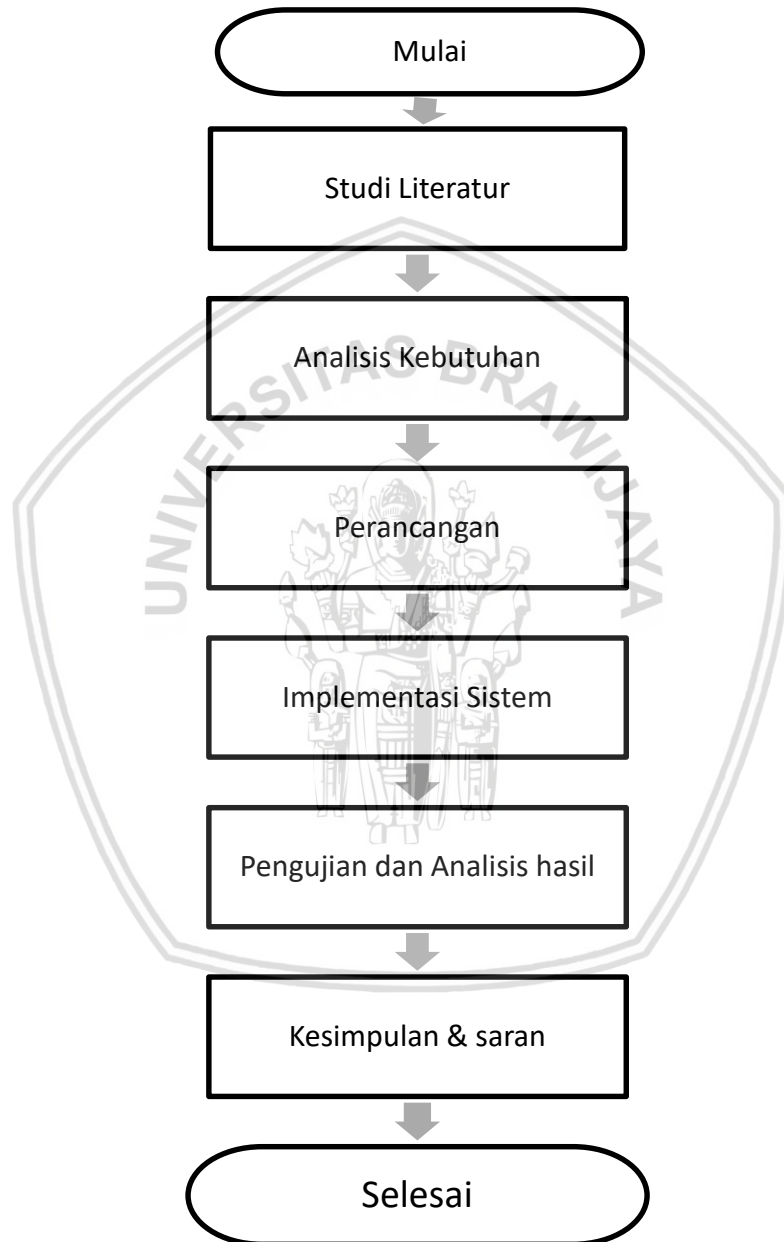
2.9.3 *Compatibility Testing*

Pengujian kompatibilitas merupakan salah satu dari beberapa pengujian yang perlu dilakukan pada antarmuka pengguna aplikasi berbasis web untuk memastikan apakah aplikasi tidak memiliki masalah ketika dijalankan pada komputer, perangkat tampilan, sistem operasi, aplikasi peramban, atau kecepatan jaringan yang berbeda (Pressman, 2010).



BAB 3 METODOLOGI

Sesuai dengan teori pengembangan terkait *waterfall model* yang diangkat dalam penelitian ini, dapat dihasilkan kerangka kerja dari penelitian yang akan dilakukan sesuai dengan metode yaitu:



Gambar 3.1 Diagram Alur Metode Penelitian



3.1 Studi Literatur

Studi literatur dilakukan dengan mencari referensi untuk mendapatkan landasan kerangka berpikir yang berkaitan dengan penelitian yang dilakukan oleh penulis sehingga didapatkan konsep dasar, dasar teori serta metode yang akan membantu dalam pengembangan sistem informasi keuangan. Beberapa teori yang dipelajari meliputi:

1. Penelitian sebelumnya
2. Sistem informasi
3. SDLC model *waterfall*
4. Proses Bisnis
5. UML (*Unified Modeling Language*)
6. OOP (*Object Oriented Programming*)
7. *Black box testing*
8. *User Acceptance Testing*
9. *Compatibility testing*

3.2 Analisis Kebutuhan Sistem

Analisis Kebutuhan sistem adalah tahap dimana nantinya peneliti melakukan akan mengidentifikasi kebutuhan serta di analisis. Data hasil dari wawancara akan di dokumentasikan dan informasi yang di butuhkan untuk mengembangkan sistem keuangan pada institute pendidikan TK Permata Iman itu sendiri. Dimana proses ini mengidentifikasi proses bisnis yang sedang berjalan di TK Permata Iman berfokus pada proses bisnis keuangannya.

Analisis kebutuhan sistem dilakukan dengan informasi dan data yang telah di dapatkan. Dimana proses ini melakukan analisis proses bisnis dengan membuat diagram dari proses bisnis yang dibutuhkan. Diagram proses bisnis keuangan pada TK Permata Iman dibuat menggunakan diagram BPMN serta melakukan identifikasi pada stakeholder yang terlibat dalam proses bisnis keuangan institut pendidikan TK Permata Iman.

Hasil analisis proses bisnis dan analisis kebutuhan pemangku kepentingan yang telah di definisikan menjadi dasar analisis kebutuhan sistem berupa fitur sistem, mendefinisikan kebutuhan fungsional dan non-fungsional sistem yang nanti di notasikan menjadi *use case diagram* dan *activity diagram*.

3.3 Perancangan

Tahap perancangan sistem mengacu kepada data dan hasil analisis proses bisnis menjadi rancangan yang bertujuan untuk menggambarkan alur implementasi agar implementasi yang dibuat sesuai dengan rancangan. Data – data kebutuhan fungsional dan non fungsional yang didapat akan dijadikan acuan untuk menentukan perancangan fungsi – fungsi dan alur sistem yang dibuat. Selain itu dilakukan pula perancangan basis data untuk media penyimpanan data – data sistem.

Perancangan yang akan di lakukan oleh peneliti di mulai dengan *sequence diagram*, *class diagram*, dan *physical data model diagram(PDM)* berdasarkan hasil dari analisa kebutuhan. Tahap selanjutnya adalah perancangan user interface (UI) atau tampilan yang akan di implementasikan ke dalam sistem. Tahap terakhir adalah perancangan *test case*(kasus uji) pengujian validasi dan pengujian penerimaan pengguna serta pengujian *compability*.

3.4 Implementasi Sistem

Tahap ini merupakan tahapan penerapan dari tahap Implementasi SDLC menggunakan *Waterfall Model*. Penulis mengimplementasikan sistem ke dalam bahasa pemrograman berdasarkan rancangan yang telah dibuat sebelumnya. Dalam penelitian ini proses implementasi menggunakan bahasa pemrogram *html*, *php*, *sql. Mysql* Sebagai *DBMS* untuk mengimplementasikan basis data media penyimpanan sistem informasi keuangan.

3.5 Pengujian Sistem

Pada tahap ini dilakukan pengujian dan analisis dari sistem yang berarti merupakan penerapan dari Tahap Pengujian dari SDLC menggunakan *Waterfall Model*. Pengujian perangkat lunak bertujuan untuk mengetahui kelayakan dan kesesuaian sistem informasi keuangan pada KB & TK Permata Iman dengan kebutuhan yang dirancang. Pengujian dilakukan menggunakan *Blackbox validation testing*, *User*, Pengujian penerimaan pengguna, dan pengujian *compability*. Penggunaan *Blackbox validation testing* di gunakan untuk mengetahui apakah fitur dan fungsi sistem telah sesuai dengan yang di definisikan pada tahap analisis kebutuhan. *User Acceptance Testing* di gunakan apakah sistem telah mencakup kebutuhan pengguna dan juga proses pengakuan *delivery* sistem kepada pengguna. *Compability* di gunakan agar dapat memastikan kebutuhan permasalahan terkait kepala sekolah tidak berada di tempat untuk melakukan verifikasi pengajuan dana.

3.6 Kesimpulan dan Saran

Pengambilan kesimpulan dilakukan setelah semua tahap implementasi perangkat lunak dan pengujian perangkat lunak telah dilakukan. Pengambilan kesimpulan diambil berdasarkan solusi dari masalah penelitian. Tahap akhir dari penulisan ini adalah saran yang dimaksud untuk memperbaiki kesalahan-kesalahan yang terjadi dan menyempurnakan penulisan serta memberi pertimbangan terhadap perangkat lunak lebih lanjut.

BAB 4 ANALISIS KEBUTUHAN SISTEM

Pada bab ini akan membahas mengenai analisis kebutuhan sistem informasi keuangan institut pendidikan KB & TK Permata Iman. Data di dapatkan dengan menggunakan metode wawancara dengan tujuan dapat di analisis menjadi kebutuhan sistem. Analisis kebutuhan di mulai dengan analisis proses bisnis *as-is* dan *to-be* yang akan dimodelkan dengan *diagram BPMN*. Proses bisnis *as-is* sendiri merupakan proses bisnis yang ada saat ini di dapatkan dari data wawancara. Proses bisnis *to-be* merupakan proses bisnis rekomendasi hasil analisis yang dibuat berdasarkan proses bisnis *as-is*.

Setelah menganalisis proses bisnis dan mendapatkan rekomendasi tahap selanjutnya dilakukan pemetaan proses bisnis agar mendapatkan proses bisnis mana yang dapat di jadikan kebutuhan oleh sistem itu sendiri. Lalu tahap berikutnya di lakukannya analisis kebutuhan dari perspektif pemangku kepentingan agar mendapatkan informasi yang jelas terkait permasalahan yang di hadapi oleh pemangku kepentingan. Berdasarkan permasalahan yang di hadapi pemangku kepentingan, di lakukannya identifikasi pemangku kepentingan dan di lakukannya identifikasi aktor.

Berdasarkan proses bisnis rekomendasi dan analisis pemangku kepentingan, tahap selanjutnya penulis melakukan analisis kebutuhan fungsional dan nonfungsional serta fitur sistem yang di dasari oleh analisis proses bisnis yang telah di rekomendasikan. Kebutuhan fungsional memuat kebutuhan yang di harapkan ada pada sistem dan non-fungsional bersifat kebutuhan pendukung.

4.1 Analisis Proses Bisnis AS-IS dan TO-BE

Pada sub bab ini dilakukan analisis terhadap sistem yang digunakan saat dilakukan penelitian. Hasil dari analisis ini didapatkan alur proses bisnis yang sedang berjalan (*as-is*), analisis permasalahan dan proses bisnis usulan (*to-be*) sebagai solusi dalam permasalahan.

4.1.1 Proses Bisnis As-Is

Proses bisnis *as-is* adalah proses bisnis yang berlaku untuk saat ini di institut pendidikan KB & TK Permata Iman. Ruang lingkup proses bisnis yang akan di analisis oleh penulis terkait proses bisnis keuangan KB & TK Permata Iman. Berdasarkan Hasil wawancara pada lampiran 1 di dapatkan bagaimana proses bisnis yang berlangsung pada keuangan KB dan TK Permata Iman. Keuangan pada KB dan TK Permata Iman berfokus kepada mengelola alur keuangan, pencatatan keuangan dan menyediakan laporan yang dapat di pertanggung jawabkan kepada kepala sekolah dan pihak yayasan.

Mengelola alur keuangan adalah bagaimana nanti bendahara memegang kontrol terhadap keuangan yang bersifat pengeluaran dan juga pemasukan. Secara umum dana pemasukan disimpan oleh bendahara dan pengeluaran dana

yang bersifat pengajuan membutuhkan persetujuan bendahara untuk pencairan dana. Alur pengajuan dana di mulai dari pihak yang meminta dana membutuhkan dana dengan kebutuhan tertentu dengan membuat *form* kertas yang berisikan kebutuhan apa dan jumlah dananya. *Form* kertas di berikan kepada kepala sekolah untuk di verifikasi. Setelah mendapatkan verifikasi kepala sekolah, *form* pengajuan dana di berikan kepada bendahara untuk penyerahan uang. Pihak yang melakukan pengajuan dana selanjutnya memberikan nota pengeluaran/ kwitansi hasil dari pembelanjaan dana tersebut lalu di serahkan kepada bendahara beserta kembaliannya (jika ada). Selanjutnya bendahara melakukan pencatatan dana berdasarkan nota yang di berikan dan dicatat ke dalam buku keuangan. setiap bulannya bendahara juga memiliki tanggung jawab untuk melakukan rekapan pencatatan dalam bentuk laporan yang nantinya di berikan kepada kepala sekolah. Kepala sekolah menerima laporan keuangan dan mempresentasikan laporan keuangan perbulan kepada pihak yayasan.

Berdasarkan uraian diatas penulis menyimpulkan proses bisnis saat ini pada keuangan KB & TK Permata Iman dijelaskan pada Tabel 4.1.

Tabel 4.1 Proses Bisnis Keadaan Saat Ini (As-Is)

Nama Proses bisnis	Deskripsi
Pengajuan Dana	Proses bisnis kegiatan pengajuan dana yang yang dilakukan oleh ketua bagian KB & TK Permata Iman selaku peminta dana dalam meminta dana terkait keperluan kegiatan, sarana dll. Proses ini di butuhkannya verifikasi oleh kepala sekolah dan pencairan oleh bendahara.
Pencatatan Dana	Proses bisnis terkait pencatatan pemasukan/pengeluaran dana, yang dilakukan oleh bendahara sekolah untuk mencatat setiap alur dana yang masuk dan keluar di KB dan TK Permata Iman
Laporan Keuangan Bulanan	Proses bisnis pelaporan keuangan bulanan yang dilakukan oleh bendahara sekolah pada setiap bulan.

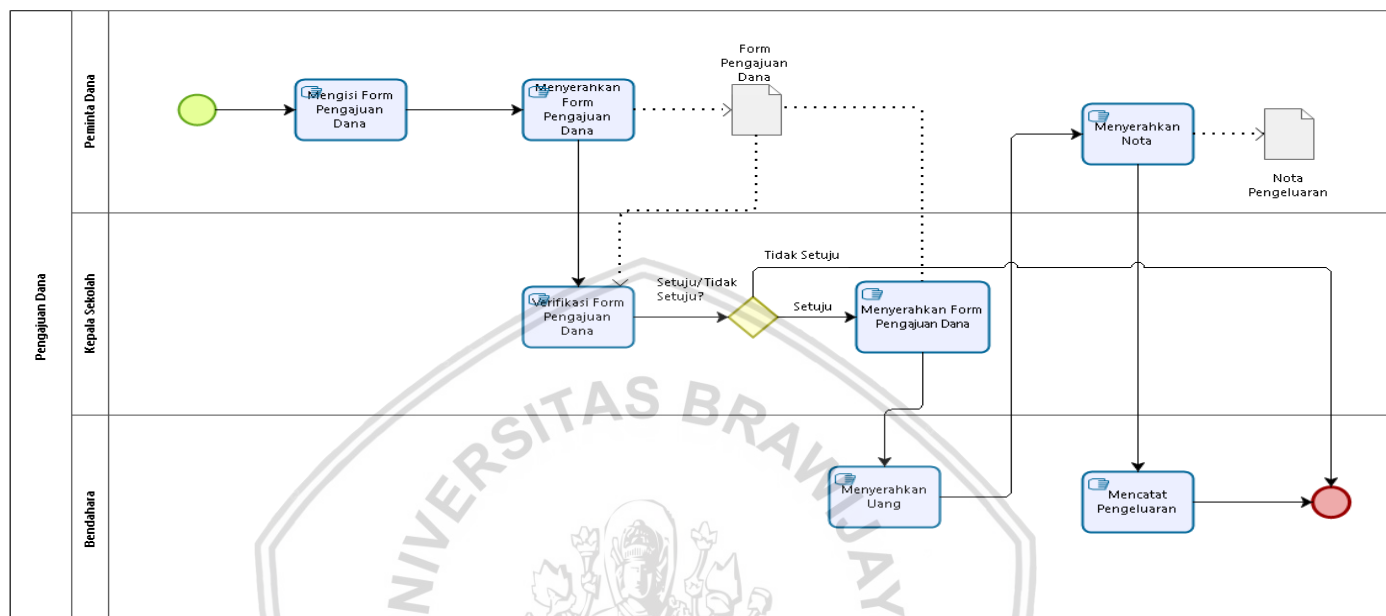
4.1.1.1 Pemodelan Proses Bisnis Pengajuan Dana

Berikut alur proses bisnis pengajuan dana:

1. Pihak yang mengajukan mengisi *form* pengajuan dana sesuai dana yang dibutuhkan
2. Menyerahkan *formulir* pengajuan dana kepada kepala sekolah
3. Kepala sekolah akan memeriksa dan melakukan verifikasi *formulir* tersebut
4. Jika disetujui *formulir* tersebut akan diserahkan kepada bendahara sekolah
5. Bendahara sekolah menyerahkan sejumlah uang kepada yang mengajukan dana

6. Setelah pihak yang mengajukan menggunakan uang tersebut, akan menyerahkan nota pengeluaran kepada bendahara
7. Bendahara mencatat pengeluaran tersebut

Pada Gambar 4.1 merupakan pemodelan diagram yang telah dimodelkan berdasarkan proses bisnis yang dijelaskan di atas.



Gambar 4.1 Diagram BPMN Pengajuan Dana

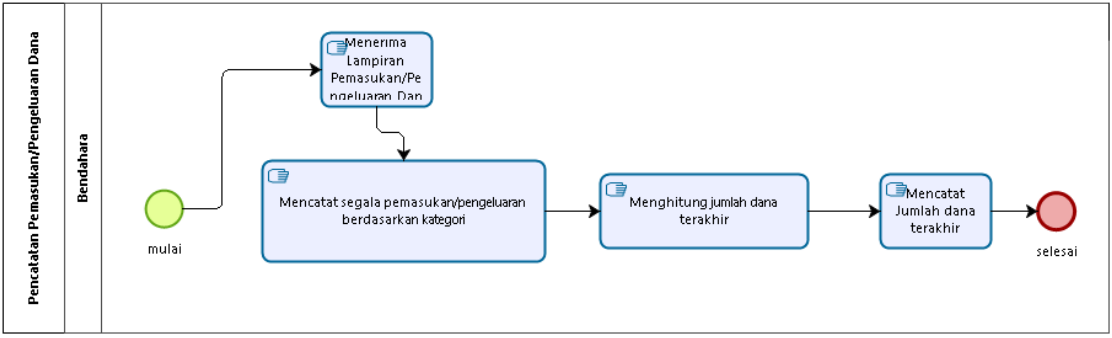
4.1.1.2 Pemodelan Proses Bisnis Pencatatan Pemasukan atau Pengeluaran Dana

Berikut alur proses bisnis pencatatan pemasukan atau pengeluaran dana:

1. Bendahara menerima lampiran keterangan dana. Berupa bukti pembayaran jika termasuk pemasukan dana dan berupa nota pengeluaran jika termasuk pengeluaran dana.
2. Bendahara mencatat pemasukan/pengeluaran dana pada sesuai dengan kategori keuangan.
3. Menghitung dan mencatat pemasukan dan pengeluaran dana.

Pada Gambar 4.2 merupakan gambar diagram yang telah dimodelkan berdasarkan proses bisnis yang dijelaskan di atas.





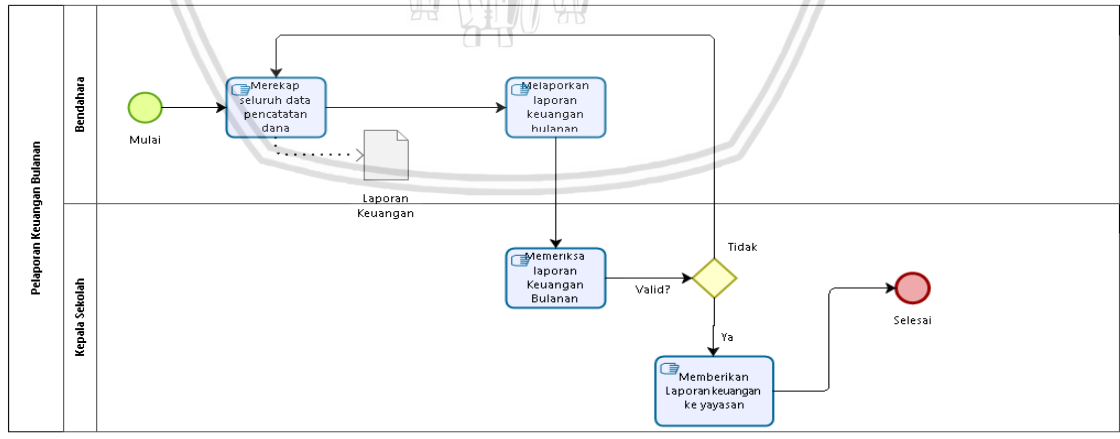
Gambar 4.2 Diagram BPMN pemasukan atau pengeluaran dana

4.1.1.3 Proses Bisnis Laporan Keuangan Bulanan

Berikut alur proses bisnis laporan keuangan per bulan:

1. Bendahara merekap seluruh catatan pengeluaran dan pemasukan selama satu bulan
2. Bendahara melaporkan keuangan kepada kepala sekolah
3. Kepala sekolah memeriksa laporan keuangan, jika tidak ada revisi maka laporan tersebut diserahkan kepada yayasan
4. Jika ada revisi, maka laporan akan diserahkan kembali ke bendahara sekolah untuk diperbaiki

Pada Gambar 4.3 merupakan gambar diagram yang telah dimodelkan berdasarkan proses bisnis yang telah di uraikan di atas.



Gambar 4.3 Diagram BPMN Pelaporan Keuangan Bulanan



4.1.2 Analisis Permasalahan

Analisis permasalahan menjelaskan masalah yang terjadi saat ini dalam proses bisnis keuangan. Data yang didapatkan pada analisis permasalahan berdasarkan hasil wawancara. Tabel 4.2 menjelaskan tentang analisis permasalahan yang ada pada proses bisnis keuangan KB dan TK Permata Iman.

Tabel 4.2 Analisis Permasalahan

<i>The problem of</i>	<ol style="list-style-type: none">1. Kepala sekolah sering berada di luar sekolah di saat di butuhkan untuk persetujuan pengajuan dana2. Menumpuknya pencatatan dana pada hari tertentu di karenakan banyak proses pemasukan dan pengeluaran yang berjalan berdampingan3. Proses pelaporan keuangan membutuhkan rekap data keuangan
<i>Affects</i>	Kepala Sekolah, Bendahara, Peminta Dana
<i>The impact of which is</i>	<ol style="list-style-type: none">1. Terlambatnya proses pencairan dana yang di butuhkan oleh peminta dana2. Proses pencatatan membutuhkan waktu yang lama3. Laporan keuangan perbulan sering terlambat
<i>A successful solution would</i>	<ol style="list-style-type: none">1. Menyediakan sistem untuk mempermudah kepala sekolah agar dapat memverifikasi jika berada di luar sekolah2. Menyediakan sistem yang dapat memfasilitasi pencatatan pemasukan dan pengeluaran3. Menyediakan sistem yang bisa mengolah data keuangan menjadi laporan dan di laporkan setiap bulannya

4.1.3 Proses Bisnis To-Be

Proses bisnis *to-be* merupakan rekomendasi solusi berdasarkan analisis permasalahan proses bisnis *as-is* yang telah diidentifikasi sebelumnya. Proses bisnis *to-be* berfokus bagaimana proses bisnis yang nantinya akan di fasilitasi oleh sistem informasi keuangan pada institut KB & TK Permata Iman.

PB-XX

Gambar 4.4 Aturan Penomoran Proses Bisnis

Gambar 4.4 merupakan aturan penomoran proses bisnis yang di butuhkan agar mempermudah *tracebility*. Dan setiap elemennya akan dijelaskan pada Tabel 4.3.

Tabel 4.3 Penjelasan Aturan Penomoran Kebutuhan

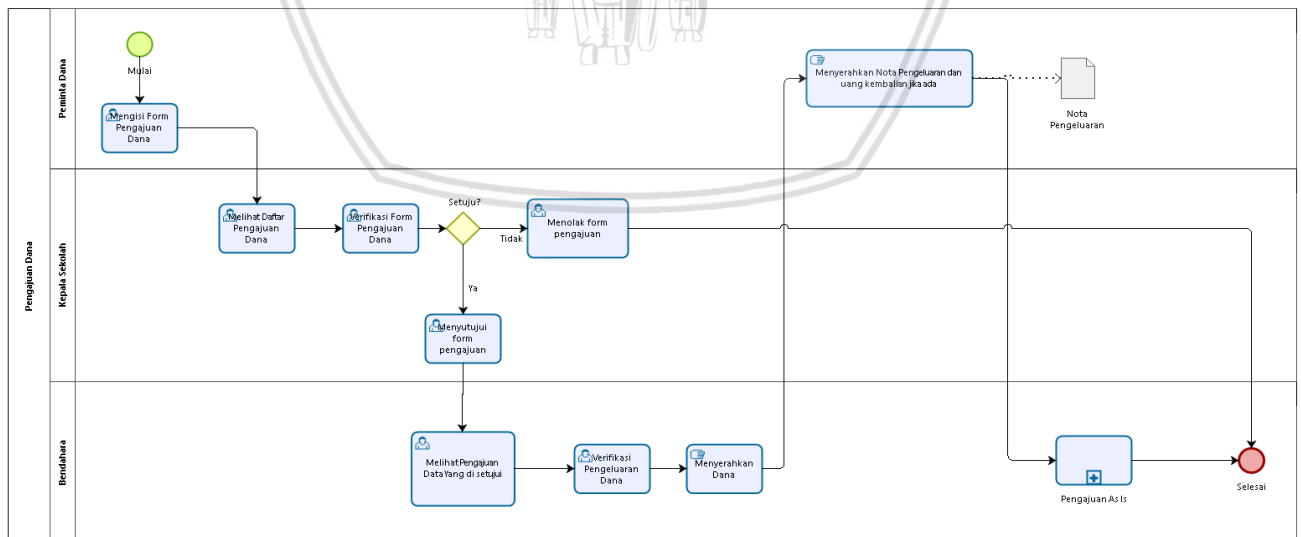
Aturan Penomoran	Deskripsi
PB	Proses Bisnis
XX	Merupakan nomor proses bisnis

Proses bisnis *to-be* merupakan rekomendasi solusi berdasarkan analisis permasalahan proses bisnis *as-is* yang telah diidentifikasi sebelumnya. Proses bisnis *to-be* berfokus bagaimana proses bisnis yang nantinya akan difasilitasi oleh sistem informasi keuangan pada institut KB & TK Permata Iman. Tabel 4.4 berikut menjelaskan daftar proses bisnis usulan.

Tabel 4.4 Daftar Proses Bisnis Usulan (To-Be)

Kode Proses Bisnis	Nama	Deskripsi
PB-01	Pengajuan Dana	Proses bisnis terkait pengajuan dana menggunakan sistem informasi keuangan
PB-02	Pencatatan Dana	Proses bisnis terkait pencatatan dana menggunakan sistem informasi keuangan
PB-03	Laporan Keuangan Bulanan	Proses bisnis terkait laporan keuangan bulanan menggunakan sistem informasi keuangan

4.1.3.1 Proses Bisnis To-Be Pengajuan Dana



Gambar 4.5 Diagram BPMN Proses Bisnis To-Be Pengajuan Dana

Gambar 4.5 merupakan proses bisnis *to-be* pengajuan dana, berikut merupakan penjelasan alur proses bisnis *to-be* pengajuan dana :

1. Peminta mengisi *form* pengajuan dana, kemudian submit *form* pengajuan dana ke sistem.
2. Kepala sekolah melihat daftar pengajuan dana dan melakukan verifikasi *form* pengajuan dana.
3. Bendahara melihat daftar pengajuan dana yang telah disetujui dan melakukan verifikasi pengajuan dana.
4. Bendahara menyerahkan dana kepada pihak yang mengajukan.
5. Pihak yang mengajukan dana menyerahkan nota penggunaan dana.
6. Bendahara melakukan pencatatan dana berdasarkan nota yang di serahkan oleh peminta dana.

**Tabel 4.5 Perubahan Aktivitas Proses Bisnis Pengajuan Dana
Beserta Perubahan Waktu**

No.	Aktor	Task pada Proses Bisnis As-Is	Task pada Proses Bisnis To-Be	Waktu Sebelum	Waktu Sesudah
1	Peminta Dana	Mengisi <i>form</i> pengajuan dana	Mengisi <i>form</i> pengajuan dana	10 menit	5 menit
2	Peminta Dana	Menyerahkan <i>form</i> pengajuan dana	-	5 menit	-
3	Kepala sekolah	-	Melihat daftar pengajuan dana	-	20 detik
4	Kepala sekolah	Verifikasi <i>form</i> pengajuan dana	Verifikasi <i>form</i> pengajuan dana	5 menit	5 menit
5	Kepala sekolah	Menyerahkan <i>form</i> pengajuan dana yang telah disetujui	Submit menyetujui pengajuan dana	30 menit	2 detik
6	Kepala sekolah	-	Submit tidak menyetujui pengajuan dana	-	2 detik
7	Bendahara	-	Melihat daftar pengajuan dana yang disetujui	-	20 detik

**Tabel 4.5 Perubahan Aktivitas Proses Bisnis Pengajuan Dana
Beserta Perubahan Waktu**

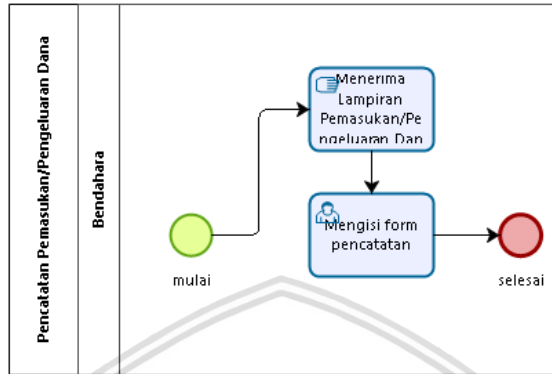
No.	Aktor	Task pada Proses Bisnis As-Is	Task pada Proses Bisnis To-Be	Waktu Sebelum (lanjutan)	Waktu Sesudah
8	Bendahara	-	Verifikasi pengeluaran dana	-	5 detik
9	Bendahara	Menyerahkan uang	Menyerahkan dana	5 menit	3 menit
10	Peminta Dana	Menyerahkan nota pengeluaran	Menyerahkan nota pengeluaran dan uang kembalian jika ada	5 menit	5 menit
11	Bendahara	Mencatat pengeluaran	Mencatat pengeluaran dan uang kembalian	10 menit	1 menit 4 detik
Total Waktu Sebelum				<i>70 menit</i>	
Total Waktu Sesudah					<i>19 menit 53 detik</i>
Selisih Waktu Sebelum dan Sesudah					<i>50 menit 7 detik</i>

Pada Tabel 4.5 merupakan perubahan aktivitas dan perubahan waktu pada proses bisnis *as-is* dan *to-be* pengajuan dana. Aktor peminta dana memiliki perubahan aktivitas diantaranya mengisi *form* pengajuan dana pada sistem, *submit form* pengajuan dana. Aktor kepala sekolah memiliki penambahan aktivitas diantaranya melihat daftar pengajuan dana melalui sistem informasi keuangan, verifikasi menyetujui atau tidak menyetujui pengajuan dana yang di minta oleh peminta dana. Selain penambahan aktivitas kepala sekolah memiliki perubahan aktivitas dari proses bisnis sebelumnya diantaranya verifikasi di lakukan di dalam sistem. Aktor bendahara memiliki perubahan aktivitas dimana pencatatan pengeluaran dan uang kembalian yang sebelumnya di lakukan melalui buku keuangan, sekarang di lakukan di dalam sistem informasi keuangan. Bendahara juga memiliki penambahan aktivitas yakni melihat daftar pengajuan dana yang telah di setujui dan melakukan verifikasi pencairan.

Total waktu pada proses bisnis *as-is* pengajuan dana membutuhkan sekitar *70 menit* untuk melakukan segala aktivitas hingga selesai. Dan untuk proses bisnis *to-*

be pengajuan dana membutuhkan sekitar 19 menit 53 detik untuk melakukan segala aktivitas hingga selesai. Total selisih waktu untuk proses bisnis *as-is* dan *to-be* pengajuan dana ialah 50 menit 7 detik.

4.1.3.2 Proses Bisnis *To-Be* Pencatatan Pemasukan atau Pengeluaran Dana



Gambar 4.6 Diagram BPMN To-Be Proses Bisnis Pencatatan Pemasukan atau Pengeluaran Dana

Gambar 4.6 merupakan *diagram* BPMN *to-be*. Berikut merupakan alur proses bisnis pencatatan pemasukan atau pengeluaran dana :

1. Bendahara menerima lampiran pemasukan/pengeluaran dana.
2. Bendahara mengisi *form* pencatatan melalui sistem informasi keuangan.

Tabel 4.6 Perubahan Aktivitas Proses Bisnis Pencatatan Dana Beserta Perubahan Waktu

No.	Aktor	Task pada Proses Bisnis As-Is	Task pada Proses Bisnis To-Be	Waktu Sebelum	Waktu Sesudah
1	Bendahara	Menerima lampiran keterangan dana	Menerima lampiran keterangan dana	2 menit	2 menit
2	Bendahara	Mencatat pemasukan/pengeluaran dana sesuai dengan kategori kebutuhan	Mengisi <i>form</i> pencatatan pada sistem informasi keuangan	5 menit	1 menit
3	Bendahara	Menghitung jumlah dana terakhir setelah ada pemasukan/pengeluaran	-	5 menit	-
5	Bendahara	Mencatat jumlah dana terakhir	-	3 menit	-
Total Waktu Sebelum				15 menit	



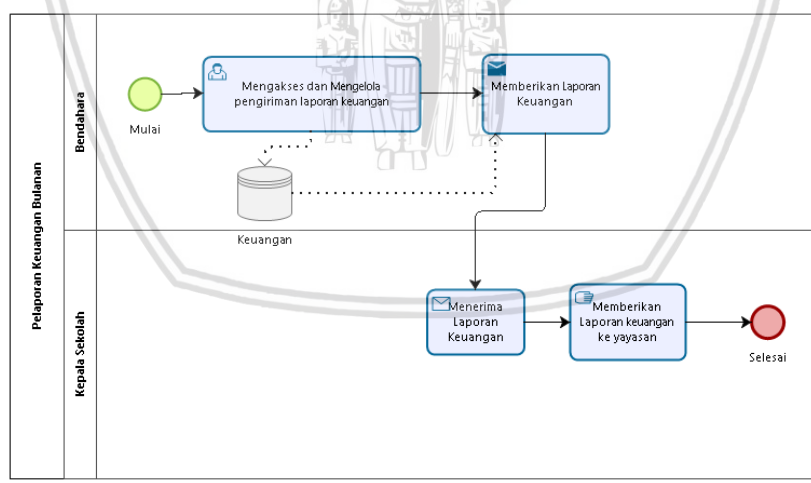
Tabel 4.6 Perubahan Aktivitas Proses Bisnis Pencatatan Dana Beserta Perubahan Waktu

No.	Aktor	Task pada Proses Bisnis As-Is	Task pada Proses Bisnis To-Be	Waktu Sebelum	Waktu Sesudah
Total Waktu Sesudah					3 menit
Selisih Waktu Sebelum dan Sesudah					12 menit

Pada Tabel 4.6 merupakan perubahan aktivitas proses bisnis pencatatan dana beserta perubahan waktu. Aktor bendahara memiliki perubahan aktivitas yang sebelumnya mencatat pemasukan / pengeluaran melalui buku lalu di ganti dengan mengelola pencatatan dana melalui sistem informasi keuangan. Aktivitas menghitung jumlah dana setelah ada pemasukan/pengeluaran dan mencatat jumlah dana terakhir pada proses bisnis *as-is* memiliki perubahan aktor yakni di ganti oleh sistem informasi keuangan.

Total waktu yang di habiskan oleh semua aktivitas proses bisnis *as-is* pencatatan dana ialah sekitar *15 menit*. Total waktu yang di habiskan oleh semua aktivitas proses bisnis *to-be* pencatatan dana ialah *3 menit*. Selisih perubahan waktu yang di habiskan semua aktivitas oleh proses bisnis *to-be* adalah *12 menit*.

4.1.3.3 Proses Bisnis To-Be Laporan Keuangan Perbulan



Gambar 4.7 Diagram BPMN To-Be Proses Bisnis Laporan Keuangan Bulanan

Gambar 4.7 merupakan *diagram BPMN to-be* proses bisnis laporan keuangan bulanan. Berikut merupakan alur rekomendasi proses bisnis Laporan Keuangan Perbulan dengan menerapkan automasi pengiriman setiap waktu pada bulannya :

1. Sistem Informasi Keuangan mengolah data berdasarkan pencatatan dana dan membuat laporan keuangan.
2. Sistem Informasi Keuangan Mengirim Laporan Keuangan.
3. Kepala sekolah menerima laporan keuangan bulanan.
4. Kepala sekolah memberikan serta mempresentasikan laporan keuangan bulanan tersebut ke yayasan.

Tabel 4.7 Perubahan Aktivitas Proses Bisnis Laporan Keuangan Bulanan Beserta Perubahan Waktu

No.	Aktor	Task pada Proses Bisnis As-Is	Task pada Proses Bisnis To-Be	Waktu Sebelum	Waktu Sesudah
1	Bendahara	Merekap seluruh catatan pengeluaran dan pemasukan setiap bulan	Mengakses dan mengelola pengiriman laporan keuangan	3 jam	10 detik
2	Bendahara	Melaporkan laporan keuangan ke kepala sekolah	-	10 menit	-
3	Sistem informasi keuangan	-	Mengirim Laporan Keuangan	-	5 menit
4	Kepala sekolah	Memeriksa laporan keuangan	Menerima laporan keuangan dari <i>email</i>	30 menit	5 menit
5	Kepala sekolah	Memberikan laporan keuangan ke yayasan	Memberikan laporan keuangan ke yayasan	5 menit	-
Total Waktu Sebelum				3 jam 45 menit	
Total Waktu Sesudah					10 menit 10 detik
Selisih Waktu Sebelum dan Sesudah					3 jam 34 menit 50 detik

Pada Tabel 4.7 merupakan perubahan aktivitas proses bisnis laporan keuangan bulanan beserta perubahan waktu. Aktor bendahara memiliki perubahan aktivitas



yang sebelumnya merekap seluruh catatan pengeluaran dan pemasukan, melaporkan laporan keuangan kepada kepala sekolah di ganti dengan pengolahan laporan keuangan dan pengiriman terjadwal melalui sistem informasi keuangan. Kepala sekolah memiliki perubahan aktivitas yakni menerima notifikasi laporan keuangan yang di kirim langsung ke *e-mail* melauai sistem.

Total waktu yang di habiskan oleh semua aktivitas proses bisnis *as-is* laporan keuangan bulanan ialah sekitar *3 jam 45 menit*. Total waktu yang di habiskan oleh semua aktivitas proses bisnis *to-be* laporan keuangan bulanan ialah *10 menit 10 detik*. Selisis perubahan waktu yang di habiskan semua aktivitas oleh proses bisnis *to-be* adalah *3 jam 34 menit 50 detik*.

4.2 Pemetaan Proses Bisnis

Didalam melakukan pemetaan proses bisnis usulan penulis memberikan kode aktivitas untuk setiap proses bisnis yang nantinya akan difasilitasi oleh sistem. Kode aktivitas ini bertujuan untuk mengenali proses bisnis yang nantinya akan digunakan untuk menelusuri proses bisnis terhadap fitur sistem. Gambar 4.8 merupakan aturan penomoran kode aktivitas dan Tabel 4.8 merupakan penjelasan aturan penomoran kode aktivitas.



ABP-SIK-XX

Gambar 4.8 Aturan Penomoran Aktivitas Proses Bisnis

Gambar 4.8 merupakan aturan penomoran aktivitas proses bisnis yang di butuhkan agar mempermudah *tracebilty*. Dan setiap elemennya akan dijelaskan pada Tabel 4.8.

Tabel 4.8 Penjelasan aturan penomoran kode aktivitas

Aturan Penomoran	Deskripsi
ABP	Aktivitas proses bisnis
SIK	Sistem Informasi Keuangan
XX	Nomor Urut aktivitas

Berdasarkan hasil dari analisis proses bisnis tersebut maka didapatkan pemetaan proses bisnis seperti yang di tunjukkan pada Tabel 4.9 di bawah ini :

Tabel 4.9 Pemetaan Proses Bisnis

Kode Proses Bisnis	Kode Aktivitas	Nama	Keterangan
BP-01	ABP-SIK-01	Membuat dan melihat <i>Form</i> Pengajuan Dana	Sistem harus mampu menyediakan fasilitas untuk membuat dan melihat <i>form</i> pengajuan dana.
	ABP-SIK-02	Verifikasi <i>Form</i> Pengajuan dana	Sistem harus mampu menyediakan fasilitas untuk verifikasi <i>form</i> pengajuan dana.
	ABP-SIK-03	Verifikasi Pencairan Dana	Sistem harus mampu menyediakan fasilitas untuk verifikasi pencairan dana.
BP-02	ABP-SIK-03	Verifikasi Pencairan Dana	Sistem harus mampu menyediakan fasilitas untuk verifikasi pencairan dana.
	ABP-SIK-04	Pencatatan Pengeluaran dan Pemasukan dana	Sistem harus mampu menyediakan fasilitas untuk melakukan pencatatan pengeluaran dan pemasukan dana.
BP-03	ABP-SIK-05	Membuat Pelaporan keuangan setiap bulan	Sistem harus mampu menyediakan fasilitas untuk melakukan pembuatan pelaporan keuangan setiap bulannya.
	ABP-SIK-06	Mengirim Pelaporan Keuangan setiap bulan	Sistem harus mampu menyediakan fasilitas untuk melakukan pengelolaan pelaporan keuangan setiap bulannya.

4.3 Analisis Kebutuhan Pemangku Kepentingan

Analisis kebutuhan pemangku kepentingan dilakukan untuk mengetahui apa saja permasalahan yang dihadapi oleh pemangku kepentingan sebelum sistem dibuat, dan apa yang di inginkan oleh pemangku kepentingan untuk penyelesaian masalah serta mendukung bisnis agar target bisnis dapat tercapai.

4.3.1 Tipe Pemangku Kepentingan

Tipe pemangku kepentingan merupakan kumpulan pemangku kepentingan yang memiliki karakteristik dan hubungan yang sama dengan sistem atau proyek yang menghasilkan sistem. Tabel 4.10 merupakan hasil analisis dari tipe pemangku kepentingan:

Tabel 4.10 Identifikasi Pemangku Kepentingan

No	Tipe Pemangku Kepentingan	Deskripsi	Perwakilan Pemangku Kepentingan
1	Pengguna	Orang-orang yang memiliki peran langsung untuk menggunakan atau menjalankan sistem.	Kepala Sekolah, Bendahara, Bagian Kesiswaan, Bagian Kurikulum, Bagian Humas, Bagian Sarana prasarana
2	Pengembang	Orang – orang yang mengambil andil dalam mengembangkan sistem informasi keuangan	Analisis, Programmer
3	Pemegang Wewenang	Orang – orang yang ahli dalam suatu aspek pada domain permasalahan maupun solusi. Selain itu juga instansi/orang yang memiliki wewenang	Kepala Sekolah, Anggota Yayasan

4.3.2 Identifikasi Aktor

Identifikasi aktor adalah seseorang atau sistem lain yang terlibat secara langsung dengan sistem. Tahap ini dilakukan agar lebih mudah mengetahui pengguna yang terlibat dalam proyek ini. Tabel 4.11 merupakan penjelasan tentang aktor yang terkait dengan sistem

Tabel 4.11 Identifikasi Aktor

No	Aktor	Deskripsi
1	Kepala Sekolah	Orang yang dapat menggunakan sistem dan memiliki otorisasi untuk dapat mengelola data pengguna sistem. Melihat daftar pengajuan dana. Mengelola pemasukan dan pengeluaran dana. Serta mengelola laporan keuangan bulanan.
2	Bendahara	Orang yang dapat menggunakan sistem dan memiliki otorisasi untuk melihat, memverifikasi daftar pengajuan dana. Mengelola pemasukan dan pengeluaran dana. Mengelola laporan keuangan bulanan.
3	Peminta Dana	Orang yang dapat mengajukan anggaran dana yaitu Ketua Bagian Kesiswaan, Bagian Kurikulum, Bagian Humas, Bagian Sarana Prasarana TK Permata Iman

Tabel 4.11 Identifikasi Aktor (lanjutan)

No	Aktor	Deskripsi
4	Pengguna	Orang yang belum mendapatkan otentifikasi untuk menggunakan sistem secara sepenuhnya dan yang hanya dapat melihat halaman login.

4.3.3 Kebutuhan Pemangku Kepentingan dan Aktor



Gambar 4.9 Aturan Penomoran Kebutuhan

Gambar 4.9 merupakan aturan penomoran kebutuhan pemangku kepentingan dan aktor yang di butuhkan agar mempermudah *tracebilty*. Dan setiap elemennya akan dijelaskan pada Tabel 4.12.

Tabel 4.12 Penjelasan aturan penomoran kode kebutuhan

Aturan Penomoran	Deskripsi
KB	Kebutuhan pemangku kepentingan dan aktor
SIK	Sistem Informasi Keuangan
XX	Nomor Urut aktivitas

Analisis kebutuhan pemangku kepentingan dan aktor dilakukan agar mengetahui apa saja yang di inginkan oleh pemangku kepentingan dan aktor dalam sistem dan memetakan dari setiap kebutuhan. Kebutuhan dipetakan pada kondisi saat ini dan solusi yang di tawarkan untuk memenuhi kebutuhan tersebut. Kebutuhan pemangku kepentingan dipetakan dan di sesuaikan dengan aktivitas proses bisnis yang telah di analisis. Berikut pada Tabel 4.13 merupakan hasil analisis kebutuhan pemangku kepentingan.

Tabel 4.13 Kebutuhan Pemangku Kepentingan dan Aktor

Kode Kebutuhan	Kode Aktivitas	Kebutuhan	Pemangku Kepentingan	Situasi Saat Ini	Solusi yang Diajukan
KB-SIK-01	BP-01	Kemudahan melakukan verifikasi pengajuan dana	Kepala Sekolah	Kepala sekolah harus berada di sekolah agar dapat memberikan verifikasi terkait pengajuan dana	Proses verifikasi dapat di lakukan dengan menggunakan sistem sehingga tidak perlu menunggu

Tabel 4.13 Kebutuhan Pemangku Kepentingan dan Aktor

Kode Kebutuhan	Kode Aktivitas	Kebutuhan	Pemangku Kepentingan	Situasi Saat Ini	Solusi yang Diajukan
					kehadiran kepala sekolah untuk verifikasi
KB-SIK-02	BP-01	Membuat Fasilitas pembuatan <i>form</i> pengajuan dana	Bagian Kesiswaan, Bagian Kurikulum, Bagian Humas, Bagian Sarana Prasarana	Pengajuan dana dibuat dengan <i>form</i> manual yang nantinya di isi oleh pengaju dan mendatangi kepala sekolah untuk proses verifikasi pengajuan dana.	<i>Form</i> pengajuan dana dapat dibuat pada sistem dan langsung mempresentasikan kebutuhan pengaju dengan rincian dana yang di butuhkan dengan <i>format</i> yang konsisten demi mempermudah untuk verifikasi dan proses lainnya
KB-SIK-03	BP-01	Kemudahan verifikasi pencairan dana	Bendahara	Pengajuan dana yang telah di verifikasi oleh kepala sekolah di kembalikan lagi kepada bendahara untuk di lakukan validasi serta proses pencairan dana kepada pengaju	Proses validasi oleh bendahara dilakukan dengan menggunakan sistem sehingga proses pengajuan dana sampai ke tahap pencairan dana tanpa harus menunggu lagi verifikasi dari kepala sekolah
KB-SIK-04	BP-02	Kemudahan melakukan proses pencatatan pemasukan dan pengeluaran dana berdasarkan	Bendahara	Pencatatan pemasukan dan pengeluaran dana di lakukan dengan mencatat dari nota pengeluaran dana, kartu SPP,	Proses pencatatan pemasukan dan pengeluaran dana dapat di lakukan oleh sistem dan menyimpannya di dalam database. mempermudah dan meringankan

Tabel 4.13 Kebutuhan Pemangku Kepentingan dan Aktor

Kode Kebutuhan	Kode Aktivitas	Kebutuhan	Pemangku Kepentingan	Situasi Saat Ini	Solusi yang Diajukan
		kategori tertentu untuk mempermudah rekap pencatatan dana		atauapun sumbangan dari yayasan ke dalam buku bendahara	beban bendahara serta menyediakan fitur pengelolaan kategori keuangan
KB-SIK-05	BP-03	Menyediakan fasilitas untuk laporan keuangan bulanan	Bendahara	Laporan keuangan di lakukan dengan merekap seluruh pemasukan dan pengeluaran dan dilakukannya pembukuan setiap bulannya untuk di laporkan kepada kepala sekolah dan pihak yayasan	Pelaporan keuangan perbulannya dapat di lakukan dengan sistem berdasarkan pencatatan dan pemasukan yang berada di database sistem informasi keuangan
KB-SIK-06	BP-03	Menyediakan fasilitas untuk pengiriman laporan keuangan bulanan secara otomatis	Kepala Sekolah	Laporan keuangan di berikan dengan melaporkan buku rekap hasil pencatatan	Pengiriman laporan ke kepala sekolah menggunakan sistem setiap bulan secara otomatis
KB-SIK-07		Menyediakan fasilitas untuk autentifikasi dan pengelolaan pengguna yang dapat menggunakan sistem	Kepala Sekolah	-	Sistem autentifikasi dengan menggunakan <i>log in</i> dan <i>log out</i> serta sistem yang dapat mengelola pengguna

4.4 Analisis Spesifikasi Kebutuhan

Pada sub bab ini membahas analisis spesifikasi kebutuhan terdiri dari identifikasi fitur, kebutuhan fungsional dan kebutuhan non fungsional dari sistem informasi keuangan.

4.4.1 Fitur Sistem

Berdasarkan hasil analisis pemangku kepentingan maka didapatkan fitur yang harus disediakan oleh sistem. Fitur merupakan hasil solusi yang di tawarkan berdasarkan kebutuhan pemangku kepentingan dan aktor. Fitur di sediakan agar mempermudah pengembang dalam menentukan fungsi fungsi yang akan di rancang dan nantinya akan di implementasi pada penelitian.

FEAT-XX

Gambar 4.10 Aturan Penomoran Fitur

Gambar 4.10 merupakan aturan penomoran fitur di butuhkan agar mempermudah *tracebilty*. Dan setiap elemennya akan dijelaskan pada Tabel 4.14

Tabel 4.14 Penjelasan Aturan Penomoran Fitur

Aturan Penomoran	Deskripsi
FEAT	Fitur sistem informasi keuangan
XX	Nomor urut fitur

Fitur sistem informasi keuangan dipetakan berdasarkan kebutuhan pemangku kepentingan dan aktor. Fitur sistem juga dipetakan berdasarkan kebutuhan pemangku kepentingan dan aktor. Tabel 4.15 berikut merupakan fitur yang disediakan oleh sistem informasi keuangan KB & TK Permata Iman.

Tabel 4.15 Fitur Sistem

Kode Fitur	Kode kebutuhan	Nama Fitur	Dekripsi
FEAT-01	KB-SIK-07	Autentifikasi Pengguna	Sistem dapat digunakan untuk melakukan login dan <i>log out</i>

Tabel 4.15 Fitur Sistem

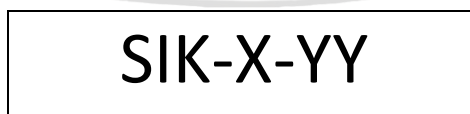
Kode Fitur	Kode kebutuhan	Nama Fitur	Dekripsi
FEAT-02	KB-SIK-07	Mengelola Pengguna	Sistem dapat digunakan untuk mengelola pengguna serta otorisasinya. Sistem dapat digunakan untuk dapat melihat, menambahkan, mengubah data pengguna
FEAT-03	KB-SIK-02	Mengelola <i>Form</i> Pengajuan Dana	Sistem dapat digunakan untuk melakukan pengelolaan terhadap <i>form</i> pengajuan dana, pengguna dapat menambah, mengubah dan menghapus <i>form</i> pengajuan dana
FEAT-04	KB-SIK-01	Verifikasi Pengajuan Dana	Sistem dapat di gunakan untuk verifikasi pengajuan dana
FEAT-05	KB-SIK-03	Verifikasi Pencairan Dana	Sistem dapat di gunakan untuk verifikasi pencairan dana
FEAT-06	KB-SIK-04	Mengelola Data Pemasukan Dan Pengeluaran	Sistem dapat di gunakan untuk melihat, mengubah, dan memasukkan data pemasukan dan pengeluaran keuangan
FEAT-07	KB-SIK-04	Mengelola Kategori Keuangan	Sistem dapat di gunakan untuk melihat, membuat, mengubah dan

Tabel 4.15 Fitur Sistem

Kode Fitur	Kode kebutuhan	Nama Fitur	Dekripsi
FEAT-08	KB-SIK-05	Menampilkan Laporan Keuangan	Sistem dapat di gunakan untuk menampilkan laporan keuangan perbulan
FEAT-09	KB-SIK-06	Mengelola Pengiriman Laporan Keuangan	Sistem dapat digunakan oleh pengguna untuk mengubah tanggal memberikan pemberitahuan otomatis laporan keuangan pada tanggal tertentu
FEAT-10	SIK-F- 06	Pengiriman Laporan Keuangan	Sistem dapat mengirimkan laporan keuangan perbulan pada tanggal tertentu
FEAT-11	KB-SIK-01	Dapat dijalankan di banyak platform	Sistem dapat di akses pada banyak platform dan OS yang berbeda

4.4.2 Kebutuhan Fungsional

Penjabaran kebutuhan fungsional dan non fungsional didapatkan dari fitur yang telah diidentifikasi. Kebutuhan fungsional dan non fungsional merupakan fungsi spesifik yang harus ada pada sistem dan nantinya akan menjadi dasar dari perancangan sistem.



Gambar 4.11 Aturan Penomoran Kebutuhan Fungsional dan Non-Fungsional

Gambar 4.11 merupakan aturan penomoran fungsional dan non-fungsional yang di butuhkan agar mempermudah *tracebilty*. Dan setiap elemennya akan dijelaskan pada Tabel 4.16



Tabel 4.16 Penjelasan Aturan Penomoran Kebutuhan

Aturan Penomoran	Deskripsi
SIK	Sistem Informasi Keuangan
X	Merupakan jenis kebutuhan, F untuk kebutuhan fungsional dan NF untuk kebutuhan non fungsional
YY	Nomor urut kebutuhan

Kebutuhan fungsional sistem informasi keuangan dipetakan berdasarkan fitur sistem informasi keuangan. Kebutuhan fungsional merupakan kondisi atau kemampuan yang harus dipenuhi oleh sistem agar fitur sistem dapat berjalan dan dapat memenuhi kebutuhan pengguna. Tabel 4.17 berikut merupakan kebutuhan fungsional yang disediakan sistem informasi keuangan.

Tabel 4.17 Kebutuhan Fungsional

Kode Fitur	Kode Kebutuhan	Deskripsi
FEAT-01	SIK-F-01	Sistem menyediakan fasilitas <i>log in</i> agar dapat membedakan antara kepala sekolah, bendahara maupun orang yang mengajukan dana.
	SIK-F-02	Sistem menyediakan fasilitas <i>log out</i> bagi <i>user</i> agar dapat keluar dari sistem.
FEAT-02	SIK-F-03	Sistem menyediakan fasilitas melihat pengguna agar dapat melihat siapa saja pengguna yang mendapatkan otorisasi masuk ke dalam sistem.
	SIK-F-04	Sistem menyediakan fasilitas menambah pengguna agar dapat menambahkan siapa saja pengguna yang mendapatkan otorisasi masuk ke dalam sistem.
	SIK-F-05	Sistem menyediakan fasilitas mengubah pengguna agar dapat mengubah siapa saja pengguna yang mendapatkan otorisasi masuk ke dalam sistem.
	SIK-F-06	Sistem menyediakan fasilitas menghapus pengguna agar dapat menghapus siapa saja pengguna yang mendapatkan otorisasi masuk ke dalam sistem.

Tabel 4.17 Kebutuhan Fungsional (lanjutan)

Kode Fitur	Kode Kebutuhan	Deskripsi
FEAT-03	SIK-F-07	Sistem menyediakan fasilitas membuat <i>form</i> pengajuan dana.
	SIK-F-08	Sistem menyediakan fasilitas melihat <i>form</i> pengajuan dana
	SIK-F-09	Sistem menyediakan fasilitas mengubah <i>form</i> pengajuan dana.
	SIK-F-10	Sistem menyediakan fasilitas hapus <i>form</i> pengajuan dana.
FEAT-04	SIK-F-11	Sistem menyediakan fasilitas verifikasi <i>form</i> pengajuan dana.
FEAT-05	SIK-F-12	Sistem menyediakan fasilitas verifikasi pencairan dana.
FEAT-06	SIK-F-13	Sistem menyediakan fasilitas membuat <i>form</i> pemasukan/pengeluaran dana.
	SIK-F-14	Sistem menyediakan fasilitas mengubah <i>form</i> pemasukan/pengeuaran dana yang telah dibuat.
	SIK-F-15	Sistem menyediakan fasilitas menghapus <i>form</i> pemasukan/pengeuaran dana yang telah dibuat.
	SIK-F-16	Sistem menyediakan fasilitas melihat data keuangan.
FEAT-07	SIK-F-17	Sistem menyediakan fasilitas melihat kategori keuangan.
	SIK-F-18	Sistem menyediakan fasilitas membuat kategori keuangan.
	SIK-F-19	Sistem menyediakan fasilitas mengubah kategori keuangan yang telah dibuat sebelumnya.
	SIK-F-20	Sistem menyediakan fasilitas menghapus kategori keuangan yang telah dibuat sebelumnya.
FEAT-08	SIK-F-21	Sistem menyediakan fasilitas melihat laporan keuangan.
FEAT-09	SIK-F-22	Sistem menyediakan fasilitas mengubah alamat tujuan pengiriman laporan keuangan.

Tabel 4.17 Kebutuhan Fungsional (lanjutan)

Kode Fitur	Kode Kebutuhan	Deskripsi
FEAT-10	SIK-F-23	Sistem menyediakan fasilitas pengiriman laporan secara otomatis berdasarkan tanggal tertentu

4.4.3 Kebutuhan Non-Fungsional

Penjabaran kebutuhan fungsional didapatkan dari hasil wawancara, data penelitian sebelumnya dan di sesuaikan dengan proses bisnis usulan. Tabel 4.18 di bawah menjelaskan tentang non-fungsional apa saja yang harus sistem penuhi.

Tabel 4.18 Kebutuhan Non-Fungsional

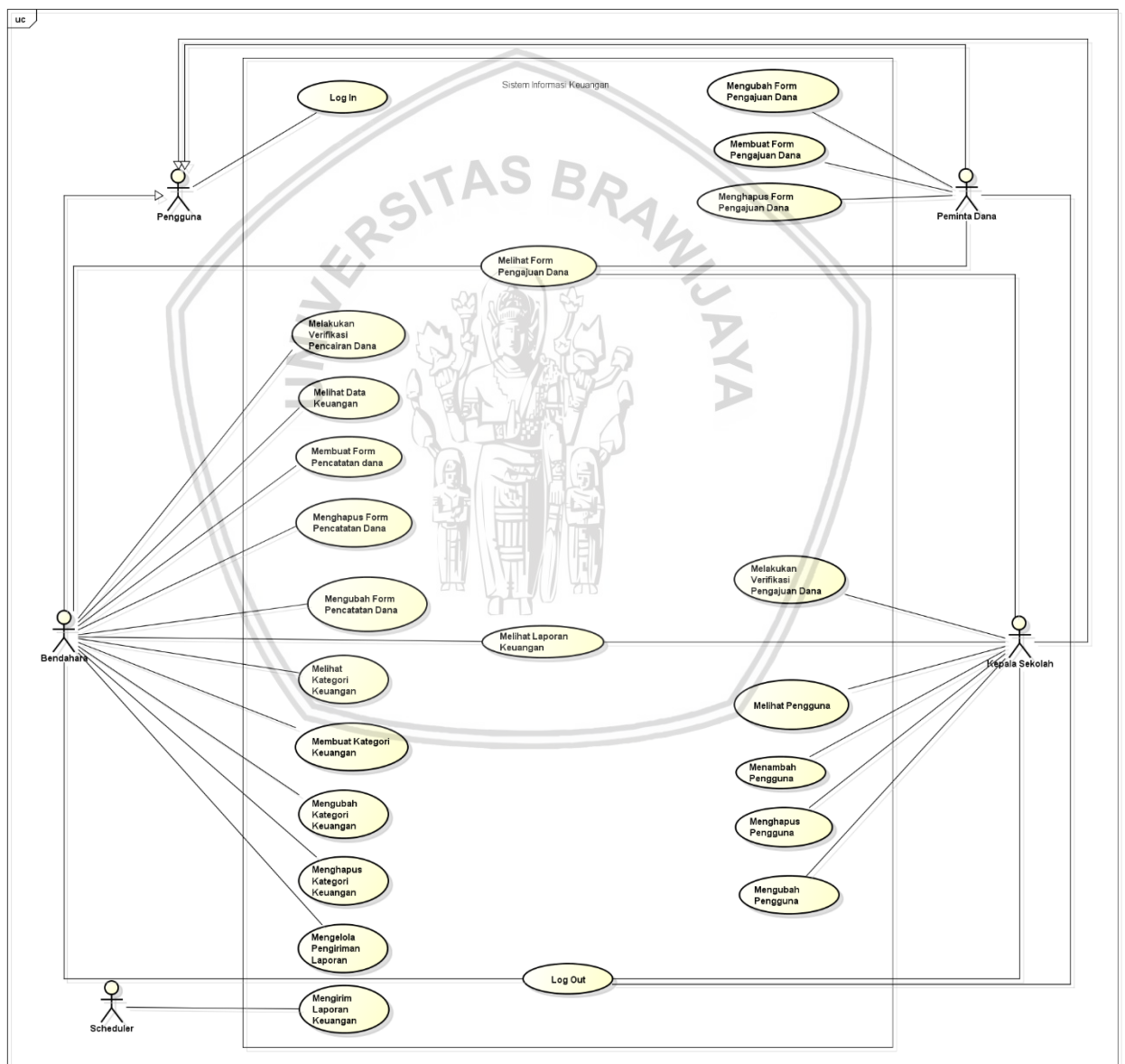
Kode Bisnis	Proses	Kode Kebutuhan	Deskripsi
FEAT-11		SIK-NF-01	Sistem dapat dijalankan di <i>Mozilla firefox</i> , <i>Google Chrome</i> dan <i>microsoft edge</i>



BAB 5 PERANCANGAN

Pada bab perancangan berisi pembahasan mengenai hasil pemodelan dengan sistem dengan menggunakan *Unified Modelling Language* (UML) sebagai notasi pemodelan. Hasil pemodelan yang dijelaskan dalam bab ini antara lain sequence diagram, class diagram, perancangan basis data yang didokumentasikan ke dalam Physical Data Model, perancangan algoritme yang didokumentasikan ke dalam *pseudocode*, perancangan antarmuka pengguna dan perancangan pengujian.

5.1 Pemodelan *Use case Diagram*



Gambar 5.1 *Use case Diagram* Sistem Informasi Keuangan

Gambar 5.1 merupakan model *use case diagram* sistem informasi keuangan yang di peroleh dari analisis kebutuhan. *Use case diagram* merupakan diagram UML yang merepresentasikan aktor-aktor yang terlibat dalam sistem beserta peran yang bisa dijalankan melalui sistem tersebut, *use case diagram* mengacu pada fitur dan kebutuhan fungsional dari sistem. Hubungan antar *use case* dengan hasil analisis kebutuhan dapat di lihat pada Tabel 5.1.

**Tabel 5.1 Hubungan *Use case Diagram*
Dengan Hasil Analisis Kebutuhan**

Kode Fitur	Kode Kebutuhan	<i>Use case</i>
FEAT-01	SIK-F-01	<i>Log in</i>
	SIK-F-02	<i>Log out</i>
FEAT-02	SIK-F-03	Melihat Pengguna
	SIK-F-04	Menambah Pengguna
	SIK-F-05	Mengubah Pengguna
	SIK-F-06	Menghapus Pengguna
FEAT-03	SIK-F-07	Membuat <i>form</i> pengajuan dana
	SIK-F-08	Melihat <i>form</i> pengajuan dana
	SIK-F-09	Mengubah <i>form</i> pengajuan dana
	SIK-F-10	Menghapus <i>form</i> pengajuan dana
FEAT-04	SIK-F-11	Melakukan verifikasi pengajuan dana
FEAT-05	SIK-F-12	Melakukan verifikasi pencairan dana
FEAT-06	SIK-F-13	Membuat <i>form</i> pencatatan dana
	SIK-F-14	Mengubah <i>form</i> pencatatan dana
	SIK-F-15	Menghapus <i>form</i> pencatatan dana
	SIK-F-16	Melihat data keuangan
FEAT-07	SIK-F-17	Melihat kategori keuangan
	SIK-F-18	Membuat kategori keuangan
	SIK-F-19	Mengubah kategori keuangan yang telah dibuat sebelumnya
	SIK-F-20	Menghapus kategori keuangan yang telah dibuat sebelumnya
FEAT-08	SIK-F-21	Melihat laporan keuangan

**Tabel 5.1 Hubungan *Use case Diagram*
Dengan Hasil Analisis Kebutuhan (lanjutan)**

Kode Fitur	Kode Kebutuhan	<i>Use case</i>
FEAT-09	SIK-F-22	Mengelola Pengiriman Laporan
FEAT-10	SIK-F-23	Pengiriman Laporan

5.1.1 *Use case Scenario*

Use case scenario berisi penjelasan mengenai setiap *use case* yang teridentifikasi. *Scenario* juga memuat informasi mengenai aktor yang menjalankan *use case*, kondisi sebelum dan sesudah *use case* dijalankan, serta tahap-tahap yang dilakukan ketika *use case* dilakukan. Tahap-tahap pada spesifikasi *use case* digunakan sebagai panduan memodelkan aktivitas ke dalam *activity diagram*.

5.1.1.1 Skenario *Use case Log in*

Tabel 5.2 Skenario *Use case Log in*

<i>Use case Name</i>	<i>Log in</i>
<i>Use case Description</i>	Proses autentifikasi oleh sistem yang berjalan saat pengguna ingin mengakses sistem
<i>Actor</i>	Pengguna
<i>Pre-Condition</i>	<ol style="list-style-type: none"> 1. Aktor telah memiliki akun dan terdaftar di dalam sistem. 2. Mengakses halaman <i>login</i> sistem informasi keuangan.
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor mengunjungi halaman login sistem informasi keuangan. 2. Aktor memasukkan <i>id</i> dan <i>password</i>. 3. Aktor menjalankan fungsi login. 4. Sistem melakukan autentifikasi dan menampilkan halaman awal sistem informasi keuangan.
<i>Alternate Flow</i>	<ul style="list-style-type: none"> • <i>Id</i> atau <i>password</i> salah <p>Jika <i>id</i> atau <i>password</i> salah maka akan muncul notifikasi <i>login gagal</i> dan aktor kembali ke halaman <i>login</i>.</p>
<i>Post-Condition</i>	Aktor berhasil <i>login</i> ke dalam sistem sesi aktor di simpan oleh sistem.

5.1.1.2 Skenario *Use case Log out*

Tabel 5.3 Skenario *Use case Log out*

<i>Use case Name</i>	<i>Log out</i>
<i>Use case Description</i>	Proses autentifikasi oleh sistem yang berjalan saat pengguna ingin mengakses sistem
<i>Actor</i>	Kepala sekolah, Bendahara, dan Pemita Dana
<i>Pre-Condition</i>	<ol style="list-style-type: none"> Berhasil melakukan <i>log in</i>. Aktor berada di dalam sistem.
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> Aktor memilih fungsi <i>log out</i>. Sistem menampilkan halaman <i>log in</i>.
<i>Alternate Course</i>	-
<i>Post-Condition</i>	Aktor berhasil <i>log out</i> .

5.1.1.3 Skenario *Use case Melihat Pengguna*

Tabel 5.4 Skenario *Use case Melihat Pengguna*

<i>Use case Name</i>	Melihat Pengguna
<i>Use case Description</i>	Proses melihat pengguna yang mendapatkan otorisasi masuk ke dalam sistem
<i>Actor</i>	Kepala Sekolah
<i>Pre-Condition</i>	<ol style="list-style-type: none"> Berhasil melakukan <i>log in</i>. Aktor berada di dalam sistem.
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> Aktor memilih menu fungsi kelola pengguna. Sistem menampilkan data daftar pengguna.
<i>Alternate Flow</i>	-
<i>Post-Condition</i>	Aktor berhasil melihat daftar pengguna.

5.1.1.4 Skenario *Use case Menambah Pengguna*

Tabel 5.5 Skenario *Use case Menambah Pengguna*

<i>Use case Name</i>	Menambah Pengguna.
<i>Use case Description</i>	Proses penambahan pengguna yang mendapatkan otorisasi masuk ke dalam sistem.
<i>Actor</i>	Kepala Sekolah.
<i>Pre-Condition</i>	<ol style="list-style-type: none"> Berhasil melakukan <i>log in</i>.

Tabel 5.5 Skenario *Use case* Menambah Pengguna (lanjutan)

	2. Aktor berada di dalam sistem.
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu fungsi kelola pengguna. 2. Sistem menampilkan halaman daftar pengguna. 3. Aktor memilih fungsi tambah pengguna 4. Sistem menampilkan <i>form</i> penambahan pengguna. 5. Aktor mengisi <i>form</i> penambahan pengguna dengan dan memilih fungsi simpan. 6. Sistem menyimpan pengguna baru dan memberikan notifikasi pengguna berhasil ditambahkan.
<i>Alternate Flow</i>	<ul style="list-style-type: none"> • Data yang di masukkan aktor tidak lengkap <p>Pada penambahan pengguna jika aktor tidak mengisi salah satu data yang harus terisi maka sistem akan memberikan pesan bahwa terdapat salah satu data yang belum terisi.</p>
<i>Post-Condition</i>	Aktor berhasil menambah pengguna.

5.1.1.5 Skenario *Use case* Mengubah Pengguna

Tabel 5.6 Skenario *Use case* Mengubah Pengguna

<i>Use case Name</i>	Mengubah Pengguna
<i>Use case Description</i>	Proses mengubah data pengguna yang mendapatkan otorisasi masuk ke dalam sistem
<i>Actor</i>	Kepala Sekolah
<i>Pre-Condition</i>	<ol style="list-style-type: none"> 1. Berhasil melakukan <i>log in</i>. 2. Aktor berada di dalam sistem. 3. Data pengguna telah tersimpan ke dalam sistem.
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu fungsi kelola pengguna 2. Sistem menampilkan halaman daftar pengguna 3. Aktor memilih salah satu pengguna yang telah berada pada daftar pengguna dan memilih fungsi mengubah pengguna



Tabel 5.6 Skenario *Use case* Mengubah Pengguna (lanjutan)

	<ol style="list-style-type: none"> 4. Sistem menampilkan <i>form</i> mengubah pengguna. 5. Aktor mengisi <i>form</i> mengubah pengguna dengan dan memilih fungsi simpan. 6. Sistem menyimpan data baru dan memberikan notifikasi data berhasil di simpan.
<i>Alternate Flow</i>	<ul style="list-style-type: none"> • Data yang di masukkan aktor tidak lengkap <p>Pada mengubah data pengguna jika aktor tidak mengisi salah satu data yang harus terisi maka sistem akan memberikan pesan bahwa terdapat salah satu data yang belum terisi.</p>
<i>Post-Condition</i>	Aktor berhasil mengubah data pengguna

5.1.1.6 Skenario *Use case* Menghapus Pengguna

Tabel 5.7 Skenario *Use case* Menghapus Pengguna

<i>Use case Name</i>	Menghapus Pengguna
<i>Use case Description</i>	Proses menghapus data pengguna yang mendapatkan otorisasi masuk ke dalam sistem
<i>Actor</i>	Kepala Sekolah
<i>Pre-Condition</i>	<ol style="list-style-type: none"> 1. Berhasil melakukan <i>log in</i>. 2. Aktor berada di dalam sistem. 3. Data pengguna telah tersimpan ke dalam sistem.
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu fungsi kelola pengguna. 2. Sistem menampilkan halaman daftar pengguna. 3. Aktor memilih salah satu pengguna yang telah berada pada daftar pengguna dan memilih fungsi menghapus pengguna. 4. Sistem memberikan pesan konfirmasi menghapus pengguna. 5. Aktor memilih fungsi hapus. 6. Sistem menghapus pengguna dan memberikan notifikasi pengguna berhasil dihapus.
<i>Alternate Flow</i>	<ul style="list-style-type: none"> • Aktor memilih fungsi batal



Tabel 5.7 Skenario Use case Menghapus Pengguna(lanjutan)

	Pada menghapus pengguna jika aktor memilih fungsi batal maka data pengguna tidak di hapus oleh sistem dan kembali ke halaman daftar pengguna
<i>Post-Condition</i>	Aktor berhasil menghapus data pengguna

5.1.1.7 Skenario Use case Melihat Form Pengajuan Dana

Tabel 5.8 Skenario Use case Melihat Form Pengajuan Dana

<i>Use case Name</i>	Melihat <i>Form</i> Pengajuan Dana
<i>Use case Description</i>	Proses melihat <i>form</i> pengajuan dana yang dilakukan oleh aktor
<i>Actor</i>	Kepala sekolah dan Bendahara, Peminta dana
<i>Pre-Condition</i>	<ol style="list-style-type: none"> Berhasil melakukan <i>log in</i> Aktor berada di dalam sistem
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> Aktor memilih menu fungsi pengajuan dana Sistem menampilkan daftar pengajuan dana
<i>Alternate Flow</i>	-
<i>Post-Condition</i>	Aktor berhasil melihat <i>form</i> pengajuan dana

5.1.1.8 Skenario Use case Membuat Form Pengajuan Dana

Tabel 5.9 Skenario Use case Membuat Form Pengajuan Dana

<i>Use case Name</i>	Membuat <i>form</i> pengajuan dana
<i>Use case Description</i>	Proses membuat <i>form</i> pengajuan dana yang dilakukan oleh aktor
<i>Actor</i>	Peminta Dana
<i>Pre-Condition</i>	<ol style="list-style-type: none"> Berhasil melakukan <i>log in</i> Aktor berada di dalam sistem
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> Aktor memilih menu fungsi pengajuan dana Sistem menampilkan halaman pengajuan dana Aktor memilih fungsi tambah <i>form</i> pengajuan dana Sistem menampilkan <i>form</i> pengajuan dana Aktor mengisi data <i>form</i> pengajuan dana dan memilih fungsi simpan



Tabel 5.9 Skenario Use case Membuat Form Pengajuan Dana(lanjutan)

	6. Sistem menyimpan data <i>form</i> pengajuan yang di tambahkan oleh aktor dan memberikan notifikasi <i>form</i> pengajuan berhasil di simpan
<i>Alternate Flow</i>	<ul style="list-style-type: none"> Data yang di masukkan aktor tidak lengkap <p>Jika aktor tidak mengisi salah satu data yang harus terisi maka sistem akan memberikan pesan bahwa terdapat salah satu data yang belum terisi.</p>
<i>Post-Condition</i>	Aktor berhasil membuat <i>form</i> pengajuan dana.

5.1.1.9 Skenario Use case Mengubah Form Pengajuan Dana

Tabel 5.10 Skenario Use case Mengubah Form Pengajuan Dana

<i>Use case Name</i>	Membuat <i>Form</i> Pencatatan Dana
<i>Use case Description</i>	Proses mengubah <i>form</i> Pengajuan oleh aktor
<i>Actor</i>	Peminta Dana
<i>Pre-Condition</i>	<ol style="list-style-type: none"> Berhasil melakukan <i>log in</i>. Aktor berada di dalam sistem. Data pengajuan dana telah tersimpan. Data pengajuan dana status belum terverifikasi.
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> Aktor memilih menu fungsi pengajuan dana. Sistem menampilkan halaman pengajuan dana. Aktor memilih salah satu <i>form</i> pengajuan dana yang telah dibuat sebelumnya dan memilih fungsi ubah. Sistem menampilkan <i>form</i> ubah pengajuan dana. Aktor mengisi <i>form</i> ubah pengajuan dana dan memilih fungsi simpan. Sistem menyimpan <i>form</i> pengajuan dana.
<i>Alternate Flow</i>	<ul style="list-style-type: none"> Data yang di masukkan aktor tidak lengkap <p>Jika aktor tidak mengisi salah satu data yang harus terisi maka sistem akan memberikan pesan bahwa terdapat salah satu data yang belum terisi.</p>



<i>Post-Condition</i>	Aktor berhasil mengubah pengajuan dana
-----------------------	--

5.1.1.10 Skenario *Use case* Menghapus *Form* Pengajuan Dana

Tabel 5.11 Skenario *Use case* Menghapus *form* pengajuan dana

<i>Use case Name</i>	Menghapus <i>form</i> pengajuan dana
<i>Use case Description</i>	Proses menghapus <i>form</i> pengajuan dana yang dilakukan oleh aktor
<i>Actor</i>	Peminta Dana
<i>Pre-Condition</i>	<ol style="list-style-type: none"> 1. Berhasil melakukan <i>log in</i>. 2. Aktor berada di dalam sistem. 3. Data pengajuan dana telah tersimpan. 4. Data pengajuan dana status belum terverifikasi.
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu fungsi pengajuan dana. 2. Sistem menampilkan halaman pengajuan dana. 3. Aktor memilih salah satu <i>form</i> pengajuan dana yang telah dibuat sebelumnya dan memilih fungsi hapus. 4. Sistem memberikan konfirmasi hapus <i>form</i> pengajuan dana. 5. Aktor mengisi fungsi hapus. 6. Sistem menghapus <i>form</i> pengajuan dana dan memberikan notifikasi bahwa data berhasil dihapus.
<i>Alternate Flow</i>	<ul style="list-style-type: none"> • Aktor memilih fungsi batal <p>Jika aktor memilih fungsi batal saat konfirmasi maka sistem membatalkan penghapusan <i>form</i> pengajuan dana dan kembali ke halaman daftar pengajuan dana</p>
<i>Post-Condition</i>	Aktor berhasil menghapus <i>form</i> pengajuan dana

5.1.1.11 Skenario *Use case* Verifikasi *Form* Pengajuan Dana

Tabel 5.12 Skenario *Use case* Verifikasi *Form* Pengajuan Dana

<i>Use case Name</i>	Verifikasi <i>Form</i> Pengajuan Dana
<i>Use case Description</i>	Proses verifikasi <i>form</i> pengajuan dana yang dilakukan oleh aktor



Tabel 5.12 Skenario *Use case* Verifikasi *Form* Pengajuan Dana(lanjutan)

<i>Actor</i>	Kepala Sekolah
<i>Pre-Condition</i>	<ol style="list-style-type: none"> Berhasil melakukan <i>log in</i>. Aktor berada di dalam sistem.
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> Aktor memilih menu fungsi pengajuan dana. Sistem menampilkan halaman pengajuan dana. Aktor memilih salah satu <i>form</i> pengajuan dana yang telah dibuat sebelumnya dan memilih fungsi verifikasi. Sistem memberikan konfirmasi verifikasi <i>form</i> pengajuan dana. Aktor memilih status verifikasi dan memilih fitur simpan.
<i>Alternate Flow</i>	<ul style="list-style-type: none"> Aktor memilih fungsi batal <p>Jika aktor memilih fungsi batal saat konfirmasi maka sistem membatalkan verifikasi <i>form</i> pengajuan dana dan kembali ke halaman daftar pengajuan dana.</p>
<i>Post-Condition</i>	Aktor berhasil melakukan verifikasi <i>form</i> pengajuan dana

5.1.1.12 Skenario *Use case* Verifikasi Pencairan DanaTabel 5.13 Skenario *Use case* Verifikasi Pencairan Dana

<i>Use case Name</i>	Verifikasi Pencairan Dana
<i>Use case Description</i>	Proses verifikasi <i>form</i> pengajuan dana yang dilakukan oleh aktor
<i>Actor</i>	Bendahara
<i>Pre-Condition</i>	<ol style="list-style-type: none"> Berhasil melakukan <i>log in</i>. Aktor berada di dalam sistem. <i>Form</i> Pengajuan dana telah di verifikasi oleh Kepala Sekolah.
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> Aktor memilih menu fungsi pengajuan dana. Sistem menampilkan daftar pengajuan dana. Aktor memilih salah satu <i>form</i> pengajuan dana yang telah di verifikasi oleh kepala sekolah.

Tabel 5.13 Skenario *Use case* Verifikasi Pencairan Dana (lanjutan)

	<ol style="list-style-type: none"> 4. Sistem memberikan konfirmasi verifikasi pencairan dana. 5. Aktor memilih status verifikasi dan memilih fitur simpan.
<i>Alternate Flow</i>	<ul style="list-style-type: none"> • Aktor memilih fungsi batal <p>Jika aktor memilih fungsi batal saat konfirmasi maka sistem membatalkan verifikasi <i>form</i> pengajuan dana dan kembali ke halaman daftar pengajuan dana</p>
<i>Post-Condition</i>	Aktor berhasil melakukan verifikasi pencairan dana

5.1.1.13 Skenario *Use case* Melihat Data KeuanganTabel 5.14 Skenario *Use case* Melihat Data Keuangan

<i>Use case Name</i>	Melihat data keuangan
<i>Use case Description</i>	Proses melihat data keuangan dimana aktor dapat melihat data keuangan seperti saldo, <i>history</i> pencatatan, dan melihat kategori keuangan
<i>Actor</i>	Bendahara
<i>Pre-Condition</i>	<ol style="list-style-type: none"> 1. Berhasil melakukan <i>log in</i> 2. Aktor berada di dalam sistem
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu Pencatatan Pencatatan dana. 2. Sistem menampilkan halaman data keuangan.
<i>Alternate Flow</i>	-
<i>Post-Condition</i>	Aktor berhasil melihat kategori keuangan

5.1.1.14 Skenario *Use case* Membuat *Form* Pencatatan DanaTabel 5.15 Skenario *Use case* Membuat *Form* Pencatatan Dana

<i>Use case Name</i>	Membuat <i>Form</i> Pencatatan Dana
<i>Use case Description</i>	Proses membuat <i>form</i> Pencatatan oleh aktor untuk melakukan pencatatan rekap dana
<i>Actor</i>	Bendahara
<i>Pre-Condition</i>	<ol style="list-style-type: none"> 1. Berhasil melakukan <i>log in</i> 2. Aktor berada di dalam sistem
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu Pencatatan dana.

Tabel 5.15 Skenario *Use case* Membuat *Form* Pencatatan Dana (lanjutan)

	<ol style="list-style-type: none"> 2. Sistem menampilkan halaman data keuangan. 3. Aktor memilih fungsi tambah Pencatatan dana. 4. Sistem menampilkan <i>form</i> pencatatan Pencatatan dana. 5. Aktor mengisi <i>form</i> dengan ketentuan yang di tampilkan oleh sistem lalu memilih fungsi simpan. 6. Sistem menyimpan data yang telah di masukkan oleh aktor dan memberikan notifikasi data telah di simpan.
<i>Alternate Flow</i>	<ul style="list-style-type: none"> • Data yang di masukkan aktor tidak lengkap <p>Jika aktor tidak mengisi salah satu data yang harus terisi maka sistem akan memberikan pesan bahwa terdapat salah satu data yang belum terisi</p>
<i>Post-Condition</i>	Aktor berhasil melakukan pencatatan Pencatatan dana

5.1.1.15 Skenario *Use case* Mengubah *Form* Pencatatan Dana

Tabel 5.16 Skenario *Use case* Mengubah *Form* Pencatatan Dana

<i>Use case Name</i>	Membuat <i>Form</i> Pencatatan Dana
<i>Use case Description</i>	Proses mengubah <i>form</i> Pencatatan oleh aktor yang telah di masukkan sebelumnya untuk melakukan pencatatan rekap dana
<i>Actor</i>	Bendahara
<i>Pre-Condition</i>	<ol style="list-style-type: none"> 1. Berhasil melakukan <i>log in</i>. 2. Data pencatatan dana telah tersimpan. 3. Aktor berada di dalam sistem.
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu Pencatatan dana. 2. Sistem menampilkan halaman data keuangan. 3. Aktor memilih salah satu data pencatatan yang telah tersimpan dan memilih fungsi ubah. 4. Sistem menampilkan <i>form</i> pencatatan dana. 5. Aktor mengisi <i>form</i> dengan ketentuan yang di tampilkan oleh sistem lalu memilih fungsi simpan.



Tabel 5.16 Skenario *Use case* Mengubah *Form* Pencatatan Dana (lanjutan)

	6. Sistem menyimpan data yang telah di masukkan oleh aktor dan memberikan notifikasi data telah di ubah.
<i>Alternate Flow</i>	<ul style="list-style-type: none"> Data yang di masukkan aktor tidak lengkap <p>Jika aktor tidak mengisi salah satu data yang harus terisi maka sistem akan memberikan pesan bahwa terdapat salah satu data yang belum terisi.</p>
<i>Post-Condition</i>	Aktor berhasil mengubah pencatatan pencatatan dana

5.1.1.16 Skenario *Use case* Menghapus *Form* Pencatatan DanaTabel 5.17 Skenario *Use case* Menghapus *Form* Pencatatan Dana

<i>Use case Name</i>	Membuat <i>Form</i> Pencatatan Dana
<i>Use case Description</i>	Proses mengubah <i>form</i> pencatatan oleh aktor yang telah di masukkan sebelumnya untuk melakukan pencatatan rekap dana
<i>Actor</i>	Bendahara
<i>Pre-Condition</i>	<ol style="list-style-type: none"> Berhasil melakukan <i>log in</i>. Aktor berada di dalam sistem.
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> Aktor memilih menu Pencatatan Pencatatan dana. Sistem menampilkan halaman data keuangan. Aktor memilih salah satu data pencatatan yang sebelumnya telah di masukkan oleh aktor dan memilih fungsi hapus. Sistem menampilkan konfirmasi hapus <i>form</i> pencatatan dana. Aktor memilih fungsi hapus. Sistem menghapus data yang telah di masukkan oleh aktor dan memberikan notifikasi data telah di hapus.
<i>Alternate Flow</i>	<ul style="list-style-type: none"> Aktor memilih fungsi batal <p>Jika aktor memilih fungsi batal saat konfirmasi maka sistem membatalkan penghapusan <i>form</i> pengajuan dana dan kembali ke halaman daftar pengajuan dana.</p>

Tabel 5.17 Skenario *Use case* Menghapus *Form* Pencatatan Dana (lanjutan)

<i>Post-Condition</i>	Aktor berhasil menghapus pencatatan pencatatan dana
-----------------------	---

5.1.1.17 Skenario *Use case* Melihat Kategori KeuanganTabel 5.18 Skenario *Use case* Melihat Kategori Keuangan

<i>Use case Name</i>	Melihat Kategori Keuangan
<i>Use case Description</i>	Proses melihat kategori keuangan yang di lakukan oleh aktor untuk melihat kategori apa saja yang dapat di <i>input</i> berguna untuk struktur laporan keuangan
<i>Actor</i>	Bendahara
<i>Pre-Condition</i>	<ol style="list-style-type: none"> Berhasil melakukan <i>log in</i> Aktor berada di dalam sistem
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> Aktor memilih menu Pencatatan Pencatatan dana. Sistem menampilkan halaman data keuangan. Aktor memilih fungsi kategori keuangan. Sistem menampilkan data kategori keunagan.
<i>Alternate Flow</i>	-
<i>Post-Condition</i>	Aktor berhasil melihat kategori keuangan

5.1.1.18 Skenario *Use case* Membuat Kategori KeuanganTabel 5.19 Skenario *Use case* Membuat Kategori Keuangan

<i>Use case Name</i>	Membuat Kategori Keuangan
<i>Use case Description</i>	Proses membuat kategori keuangan yang di lakukan oleh aktor untuk membuat kategori apa saja yang dapat di <i>input</i> berguna untuk struktur laporan keuangan.
<i>Actor</i>	Bendahara
<i>Pre-Condition</i>	<ol style="list-style-type: none"> Berhasil melakukan <i>log in</i>. Aktor berada di dalam sistem.
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> Aktor memilih menu Pencatatan Pencatatan dana. Sistem menampilkan halaman data keuangan. Aktor memilih fungsi kategori keuangan.

Tabel 5.19 Skenario *Use case* Membuat Kategori Keuangan (lanjutan)

	<ol style="list-style-type: none"> 4. Sistem menampilkan data kategori keuangan. 5. Aktor memilih fungsi tambah kategori keuangan. 6. Sistem menampilkan <i>form</i> penambahan kategori keuangan. 7. Aktor mengisi <i>form</i> penambahan kategori keuangan dan memilih fungsi simpan. 8. Sistem menyimpan data yang di masukkan oleh aktor dan memberikan notifikasi data berhasil di simpan.
<i>Alternate Flow</i>	<ul style="list-style-type: none"> • Data yang di masukkan aktor tidak lengkap <p>Jika aktor tidak mengisi salah satu data yang harus terisi maka sistem akan memberikan pesan bahwa terdapat salah satu data yang belum terisi.</p>
<i>Post-Condition</i>	Aktor berhasil membuat kategori keuangan

5.1.1.19 Skenario *Use case* Mengubah Kategori KeuanganTabel 5.20 Skenario *Use case* Mengubah Kategori Keuangan

<i>Use case Name</i>	Mengubah Kategori Keuangan
<i>Use case Description</i>	Proses mengubah kategori keuangan yang di lakukan oleh aktor untuk mengubah kategori apa saja yang dapat di <i>input</i> berguna untuk struktur laporan keuangan
<i>Actor</i>	Bendahara
<i>Pre-Condition</i>	<ol style="list-style-type: none"> 1. Berhasil melakukan <i>log in</i> 2. Aktor berada di dalam sistem
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu Pencatatan Pencatatan dana. 2. Sistem menampilkan halaman data keuangan. 3. Aktor memilih fungsi kategori keuangan. 4. Sistem menampilkan data kategori keunagan. 5. Aktor memilih fungsi kategori yang telah dibuat sebelumnya dan memilih fungsi ubah kategori. 6. Sistem menampilkan <i>form</i> perubahan kategori keuangan.

Tabel 5.20 Skenario *Use case* Mengubah Kategori Keuangan (lanjutan)

	<ol style="list-style-type: none"> 7. Aktor mengisi <i>form</i> perubahan kategori keuangan dan memilih fungsi simpan 8. Sistem menyimpan data yang di masukkan oleh aktor dan memberikan notifikasi data berhasil di simpan.
<i>Alternate Flow</i>	<ul style="list-style-type: none"> • Data yang di masukkan aktor tidak lengkap <p>Jika aktor tidak mengisi salah satu data yang harus terisi maka sistem akan memberikan pesan bahwa terdapat salah satu data yang belum terisi.</p>
<i>Post-Condition</i>	Aktor berhasil mengubah kategori keuangan.

5.1.1.20 Skenario *Use case* Menghapus Kategori KeuanganTabel 5.21 Skenario *Use case* Menghapus Kategori Keuangan

<i>Use case Name</i>	Menghapus Kategori Keuangan
<i>Use case Description</i>	Proses menghapus kategori keuangan yang di lakukan oleh aktor untuk menghapus kategori apa saja yang dapat di <i>input</i> berguna untuk struktur laporan keuangan
<i>Actor</i>	Bendahara
<i>Pre-Condition</i>	<ol style="list-style-type: none"> 1. Berhasil melakukan <i>log in</i>. 2. Aktor berada di dalam sistem..
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu Pencatatan Pemasukan dan Pengeluaran dana. 2. Sistem menampilkan halaman data keuangan. 3. Aktor memilih fungsi kategori keuangan. 4. Sistem menampilkan data kategori keunagan. 5. Aktor memilih fungsi kategori yang telah dibuat sebelumnya dan memilih fungsi hapus kategori. 6. Sistem menampilkan konfirmasi hapus kategori. 7. Aktor memilih fungsi hapus. 8. Sistem menghapus kategori dan memberikan notifikasi data telah di hapus.
<i>Alternate Flow</i>	<ul style="list-style-type: none"> • Aktor memilih fungsi batal.

Tabel 5.21 Skenario *Use case* Menghapus Kategori Keuangan (lanjutan)

	Jika aktor memilih fungsi batal saat konfirmasi maka sistem membatalkan penghapusan <i>form</i> pengajuan dana dan kembali ke halaman daftar pengajuan dana.
<i>Post-Condition</i>	Aktor berhasil menghapus kategori keuangan.

5.1.1.21 Skenario *Use case* Melihat Laporan KeuanganTabel 5.22 Skenario *Use case* Melihat Laporan Keuangan

<i>Use case Name</i>	Melihat Laporan Keuangan
<i>Use case Description</i>	Proses melihat laporan keuangan yang di lakukan oleh aktor berdasarkan pencatatan pencatatan yang telah di masukkan oleh aktor
<i>Actor</i>	Bendahara, Kepala sekolah
<i>Pre-Condition</i>	<ol style="list-style-type: none"> Berhasil melakukan <i>log in</i>. Aktor berada di dalam sistem.
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> Aktor memilih menu Laporan Keuangan. Aktor memilih laporan berdasarkan bulan tertentu. Sistem menampilkan halaman laporan keuangan.
<i>Alternate Flow</i>	-
<i>Post-Condition</i>	Aktor berhasil melihat laporan keuangan keuangan

5.1.1.22 Skenario *Use case* Mengelola Pengiriman LaporanTabel 5.23 Skenario *Use case* Mengelola Pengiriman Laporan

<i>Use case Name</i>	Mengelola Pengiriman Laporan
<i>Use case Description</i>	Proses mengelola pengiriman laporan keuangan yang di lakukan oleh aktor berdasarkan pencatatan pencatatan yang telah di masukkan oleh aktor
<i>Actor</i>	Bendahara, Kepala Sekolah
<i>Pre-Condition</i>	<ol style="list-style-type: none"> Berhasil melakukan <i>log in</i>. Aktor berada di dalam sistem.
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> Aktor memilih menu Laporan Keuangan. Aktor memilih sub menu pengiriman laporan. Sistem menampilkan <i>form</i> pengiriman .

Tabel 5.23 Skenario *Use case* Mengelola Pengiriman Laporan (lanjutan)

	<ol style="list-style-type: none"> 4. Aktor mengisi <i>form</i> pengiriman dan memilih fungsi kirim. 5. Sistem menampilkan notifikasi email telah di kirim.
<i>Alternate Flow</i>	-
<i>Post-Condition</i>	Aktor berhasil mengirim laporan keuangan ke email yang telah di tentukan

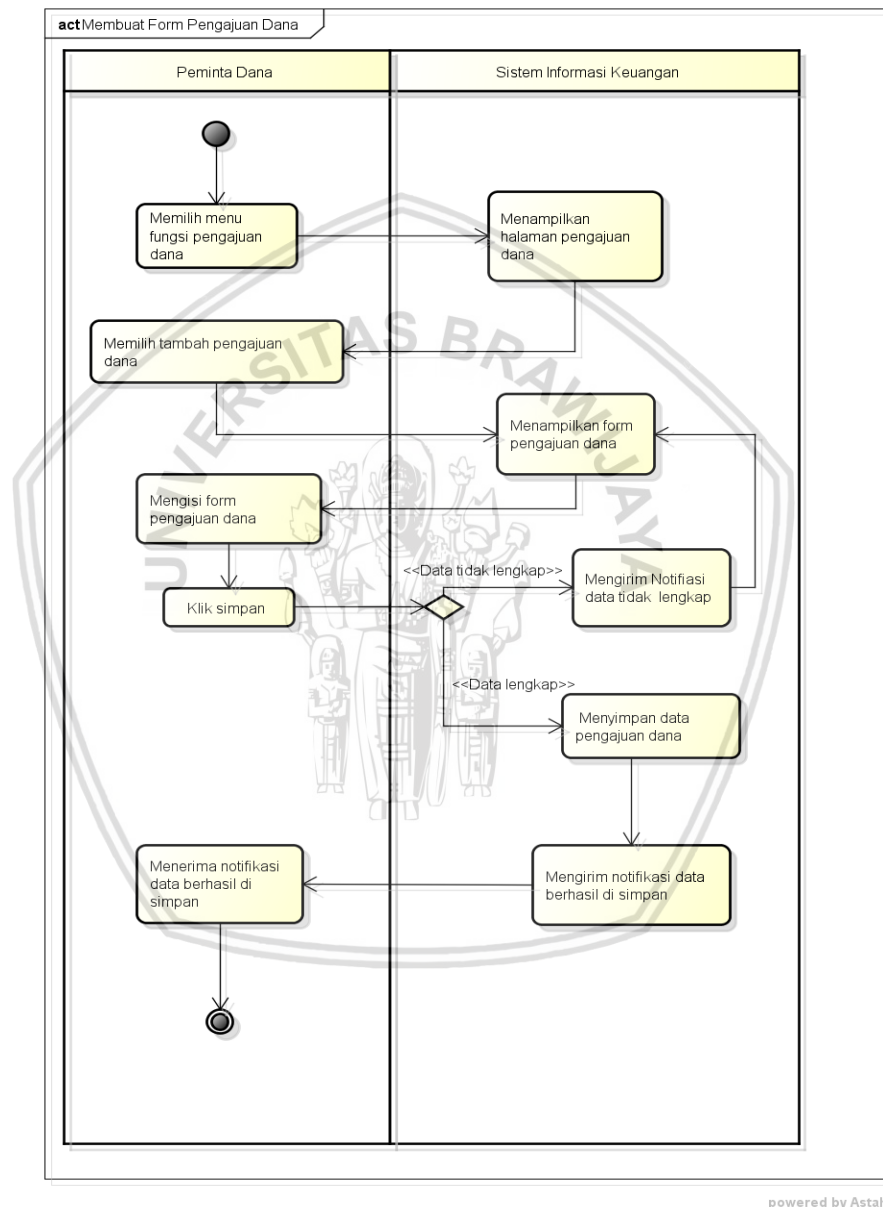
5.1.1.23 Skenario *Use case* Mengirim Laporan KeuanganTabel 5.24 Skenario *Use case* Mengirim Laporan Keuangan

<i>Use case Name</i>	Mengelola Pengiriman Laporan
<i>Use case Description</i>	Proses pengiriman laporan yang dilakukan oleh sistem secara otomatis berdasarkan tanggal tertentu.
<i>Actor</i>	<i>Scheduler</i>
<i>Pre-Condition</i>	-
<i>Basic Flow of Events</i>	<ol style="list-style-type: none"> 1. Aktor mengecek waktu sekarang dengan waktu yang telah di tentukan sebelumnya. 2. Aktor mengakses fungsi pengiriman laporan. 3. Aktor mengirimkan laporan keuangan.
<i>Alternate Course</i>	<ul style="list-style-type: none"> • Waktu belum sesuai <p>Jika waktu belum sesuai maka aktor akan kembali pada <i>basic flow of events</i> nomor 1.</p>
<i>Post-Condition</i>	Aktor berhasil mengirim laporan keuangan.

5.2 Pemodelan Activity Diagram

Pada bagian ini menjelaskan dengan menggunakan notasi *activity diagram* berdasarkan kepada *use case scenario* yang telah dibuat pada bagian *Use case Scenario*. *Activity diagram* menjelaskan dalam bentuk gambar aktor dan aktifitasnya pada skenario yang telah dibuat. Berikut adalah *activity diagram*nya:

5.2.1 Membuat *Form* Pengajuan Dana



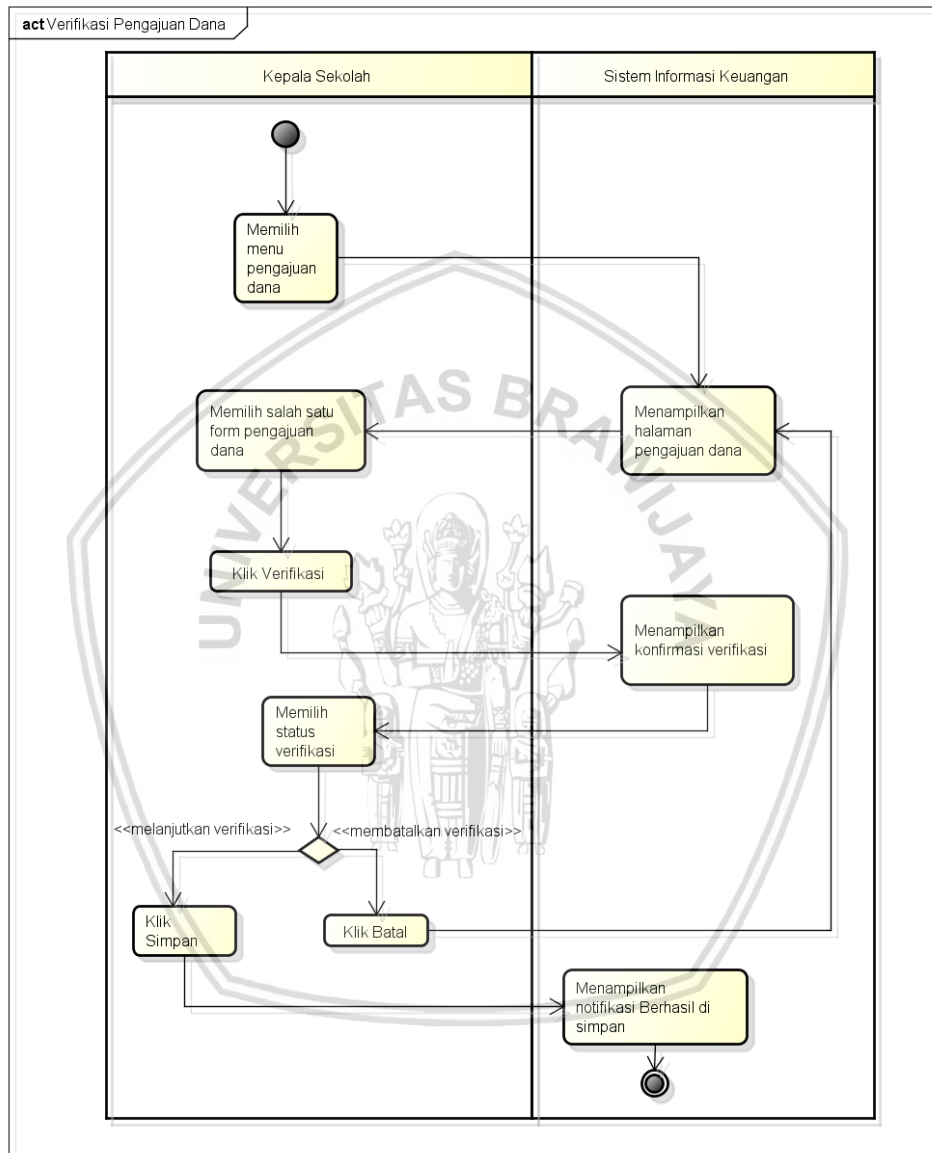
Gambar 5.2 Activity Diagram Membuat *Form* Pengajuan Dana

Pada gambar 5.2 adalah gambar aktifitas diagram untuk membuat *form* pengajuan dana. Aktifitas di mulai dengan dengan peminta dana memilih menu pengajuan dana lalu sistem merespon dengan menampilkan halaman pengajuan dana. Peminta dana memilih tambah *form* pengajuan dana dan sistem merespon



dengan memberikan *form* pengajuan dana yang nantinya akan di isi oleh aktor peminta dana dan mengklik simpan. Sistem memberikan notifikasi bahwa data berhasil disimpan yaitu merupakan akhir dari aktifitas diagram membuat *form* pengajuan dana.

5.2.2 Activity Diagram Verifikasi Pengajuan Dana



powered by Astah

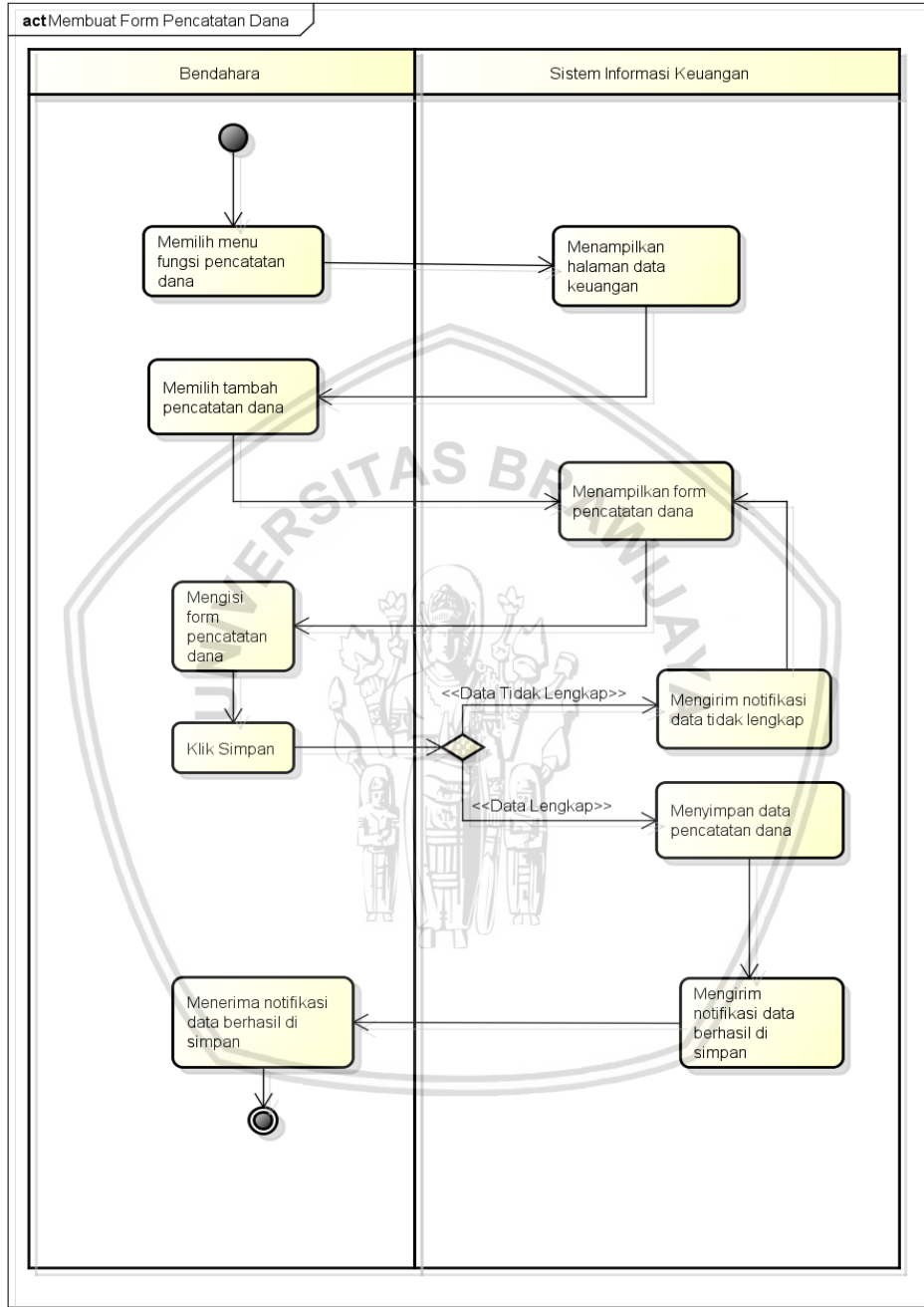
Gambar 5.3 Activity Diagram Verifikasi Pengajuan Dana

Pada gambar 5.3 adalah gambar aktifitas diagram untuk verifikasi pengajuan dana yang di lakukan oleh aktor kepala sekolah. Sistem menampilkan halaman pengajuan dana lalu kepala sekolah dapat memilih pengajuan dana yang telah dibuat oleh peminta dana sebelumnya. Kepala sekolah memilih salah satu dan memilih fungsi verifikasi maka sistem akan merespon dengan memberikan konfirmasi. Lalu kepala sekolah dapat memilih status pada *form* pengajuan dana



dan klik tombol simpan, maka status akan berubah di akhiri dengan sistem menampilkan pesan notifikasi status berhasil di simpan.

5.2.3 Activity Diagram Membuat Form Pencatatan Dana



powered by Astah

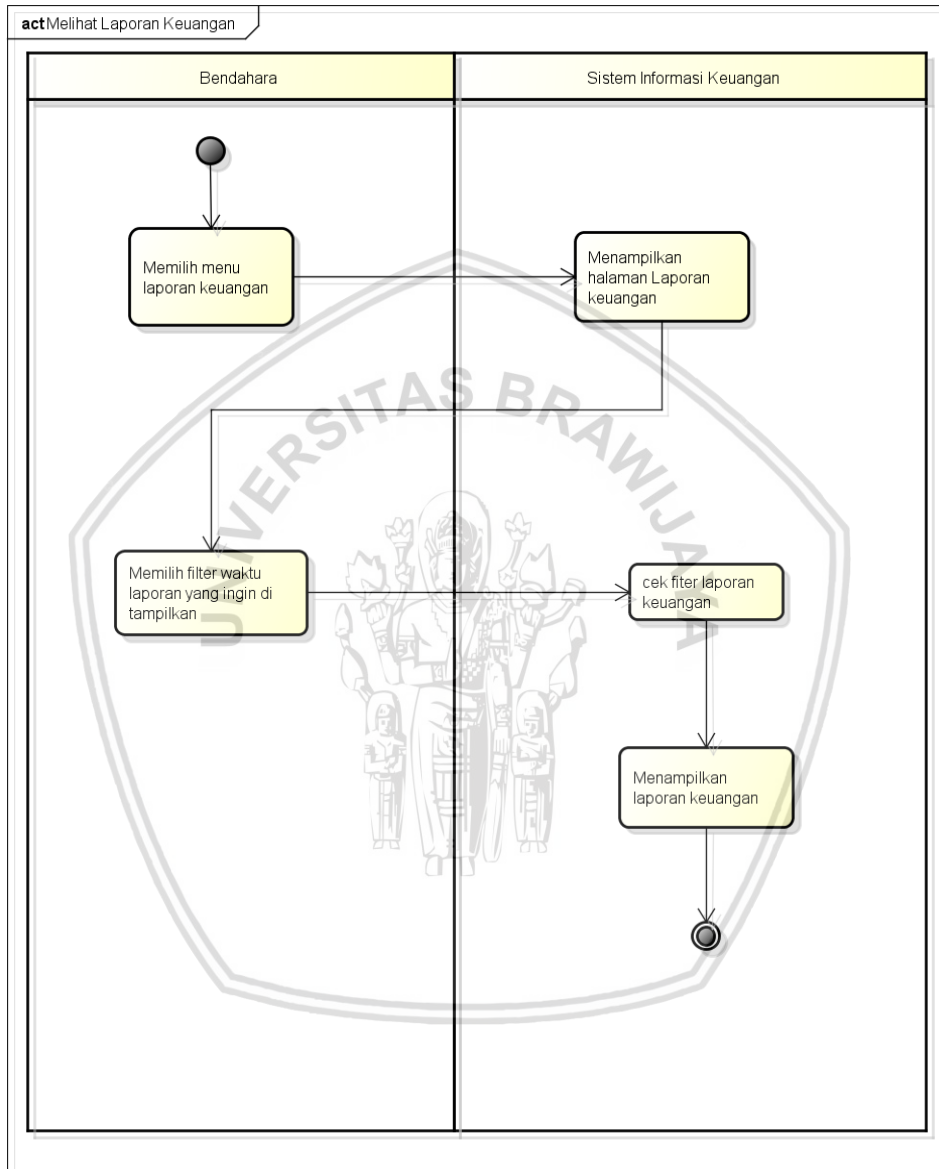
Gambar 5.4 Activity Diagram Membuat Form Pencatatan Dana

Pada gambar 5.4 menjelaskan aktifitas diagram untuk membuat form pencatatan dana. Aktifitas diagram dimulai dengan bendahara mengakses memilih menu pencatatan dana dan sistem merespon dengan menampilkan halaman data keuangan. Lalu bendahara memilih fungsi tambah form pencatatan



mengisi data dan memilih tomol simpan. Jika data ada yang tidak lengkap maka sistem akan memberikan notifikasi data tidak lengkap, jika data sudah lengkap maka akan di teruskan ke halaman data keuangan dan di berikan notifikasi data berhasil di simpan.

5.2.4 Activity Diagram Melihat Laporan Keuangan



powered by Astah

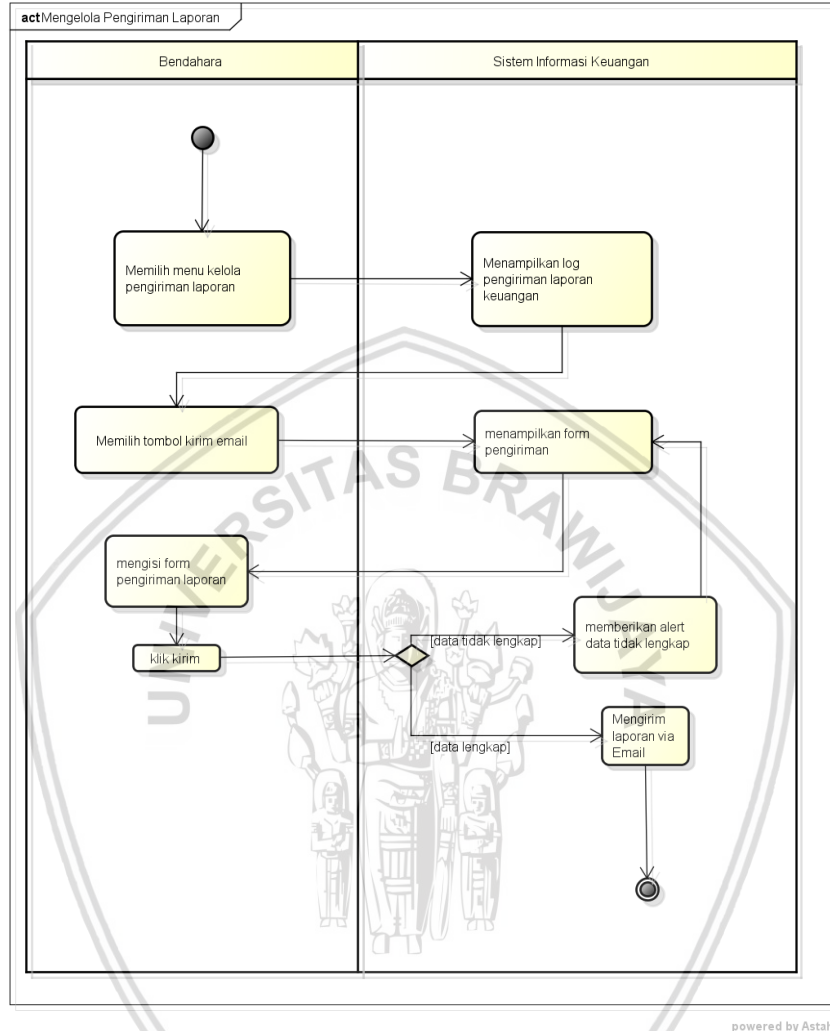
Gambar 5.5 Activity Diagram Melihat Laporan Keuangan

Pada gambar 5.5 merupakan aktifitas diagram melihat laporan keuangan di mulai dengan bendahara memilih menu laporan pada halaman utama dan sistem merespon dengan menampilkan halaman laporan keuangan. Sebelum dapat melihat laporan keuangan bendahara harus memilih waktu yang ingin di



tampilkan . Setelah memilih waktu sistem akan menampilkan laporan keuangan berdasarkan waktu yang telah di pilih.

5.2.5 Activity Diagram Mengelola Pengiriman Laporan



Gambar 5.6 Activity Diagram Pengiriman Laporan

Pada gambar 5.6 merupakan aktifitas diagram pengiriman laporan. Aktifitas di mulai dengan bendahara memilih menu laporan dan memilih sub menu kelola pengiriman laporan keuangan. Bendahara memilih tombol kirim laporan lalu sistem menampilkan *form* yang harus di isi. Jika data tidak lengkap dan tidak memenuhi standar pengiriman maka sistem memberikan notifikasi jika harus diisi dan di sesuaikan. Jika data sudah benar maka sistem akan mengirimkan laporan keuangan.

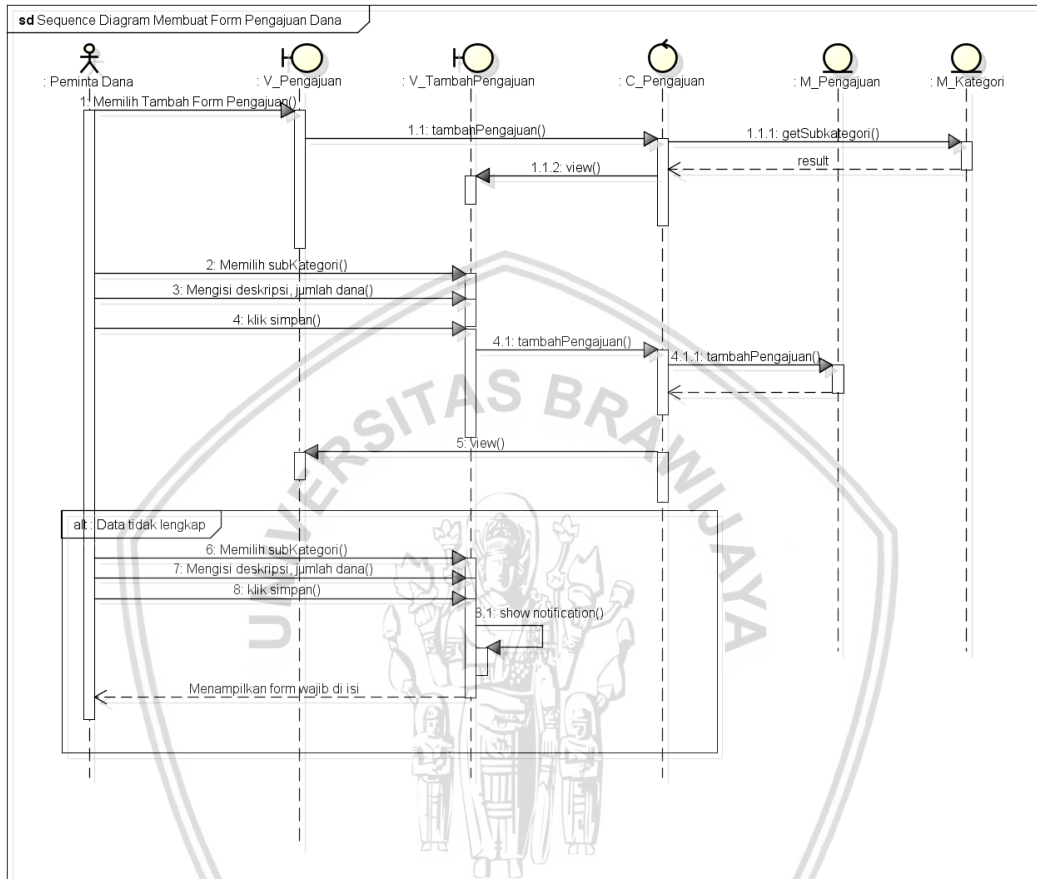
5.3 Perancangan Sequence Diagram

Sequence Diagram menggambarkan interaksi antar masing-masing objek pada setiap *use case* dalam urutan waktu. Interaksi ini berupa pengiriman serangkaian



data antar objek yang saling berinteraksi. Pada perancangan *sequence diagram* ini penulis hanya menggambarkan 5 *sequence diagram* yang mewakili fungsi utama dari sistem, berikut ini merupakan beberapa *sequence diagram* dari sistem informasi keuangan:

5.3.1 *Sequence diagram* menambah pengajuan dana

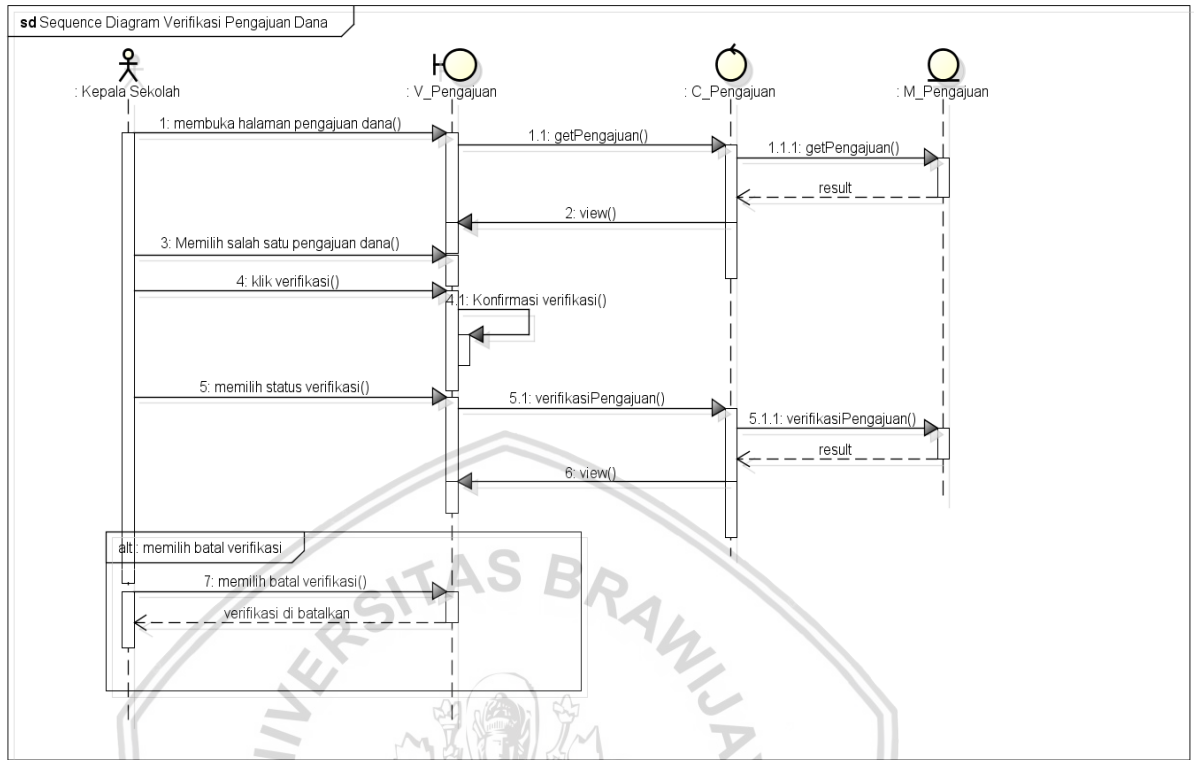


Gambar 5.7 *Sequence diagram* membuat form pengajuan dana

Pada gambar 5.7 merupakan *sequence diagram* yang di rancang untuk menggambarkan interaksi peminta dana terhadap sistem informasi keuangan dalam aktifitas menambah pengajuan dana. Peminta dana melakukan interaksi dengan tombol tambah pada V_Pengajuan lalu di teruskan ke *controller* C_Pengajuan. C_Pengajuan berinteraksi dengan V_Pengajuan lalu *model* M_Kategori agar mendapatkan subkategori dan di kembalikan ke view. Peminta dana lalu mengisi *form* sesuai dengan subkategori yang tersedia lalu klik simpan. C_Pengajuan meneruskan dengan fungsi tambahPengajuan() lalu di simpan dengan fungsi tambah Pengajuan pada model M_Pengajuan. Lalu di kembalikan ke peminta dana dengan notifikasi data berhasil di simpan. Jika data yang di masukkan tidak lengkap maka akan muncul notifikasi data tidak lengkap.



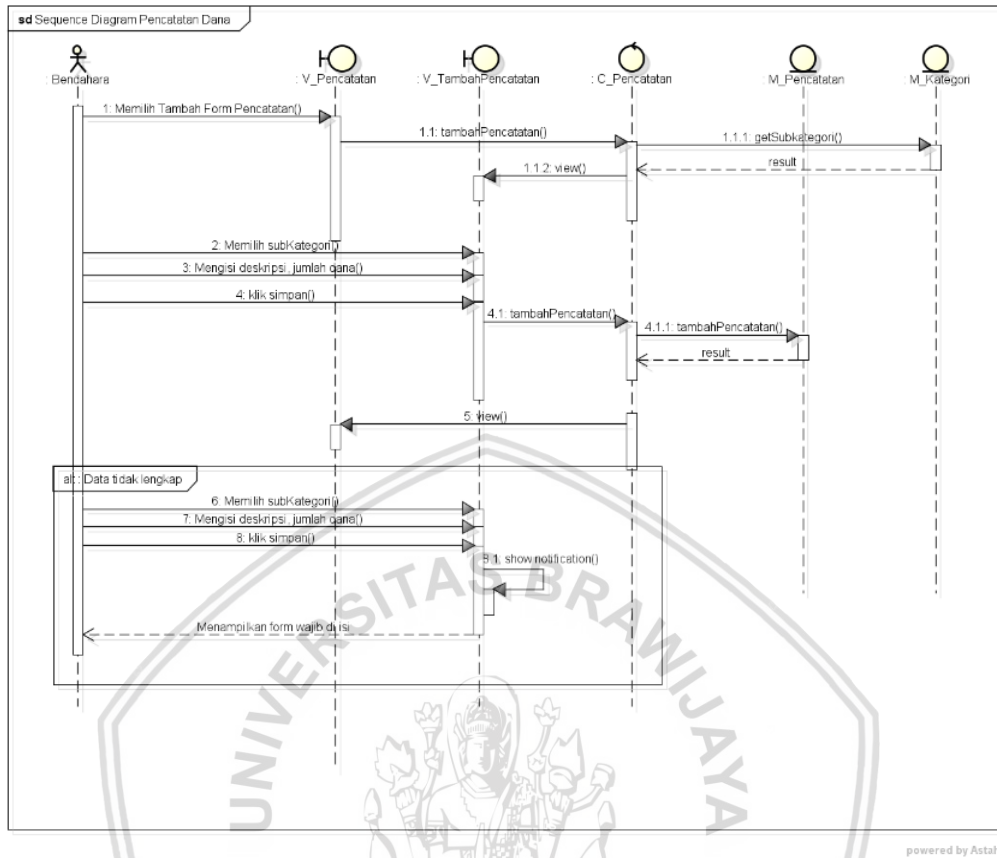
5.3.2 Sequence diagram verifikasi pengajuan dana



Gambar 5.8 Sequence Diagram verifikasi pengajuan dana

Gambar 5.8 merupakan *sequence diagram* yang di rancang untuk menggambarkan komunikasi antar objek dan aktor yang di bangun untuk aktifitas verifikasi pengajuan dana. Dimana aktor yang berperan adalah kepala sekolah melakukan interaksi dengan *view* V_Pengajuan lalu di teruskan dengan fungsi *getPengajuan* pada *controller* C_Pengajuan. Lalu di teruskan dengan *getPengajuan()* pada *model* M_Pengajuan. Data di kembalikan dengan daftar pengajuan. Lalu kepala sekolah melakukan klik verifikasi maka muncul notifikasi konfirmasi dan di lanjutkan dengan *verifikasiPengajuan()* pada C_Pengajuan. Dan *verifikasiPengajuan()* pada M_Pengajuan. Maka di kembalikan notifikasi pengajuan berhasil di verifikasi.

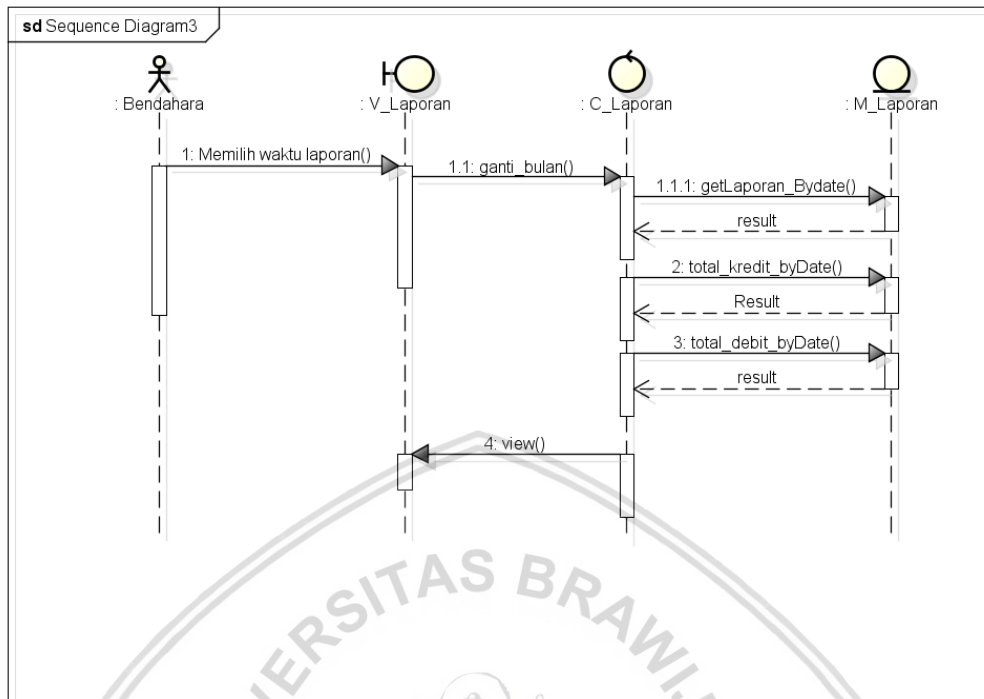
5.3.3 Sequence diagram menambah pencatatan dana



Gambar 5.9 Sequence diagram menambah pencatatan dana

Pada gambar 5.9 merupakan *sequence diagram* interaksi bendahara dengan sistem dan dijelaskan alur komunikasi antar objek. Bendahara melakukan interaksi dengan *view* V_Pencatatan lalu mengakses fungsi `pencatatan()` pada C_Pencatatan. Lalu di teruskan dengan `getKategori()` pada *model* M_Kategori. lalu mengembalikan nilai kategori dan subkategori. Bendahara lalu memilih sub kategori lalu mengisi jenis pemasukan, nilai uang lalu klik simpan. Lalu memanggil `tambahPencatatan()` yang ada pada C_Pencatatan. lalu di teruskan dengan memanggil `tambahPencatatan()` pada *model* M_Pencatatan. lalu *return* nilai dengan `showNotification()` dan menotifikasi bendahara bahwa data berhasil di simpan. Lalu alternatif jika data tidak lengkap maka akan di kembalikan dengan notifikasi data tidak lengkap kepada bendahara.

5.3.4 Sequence diagram melihat laporan

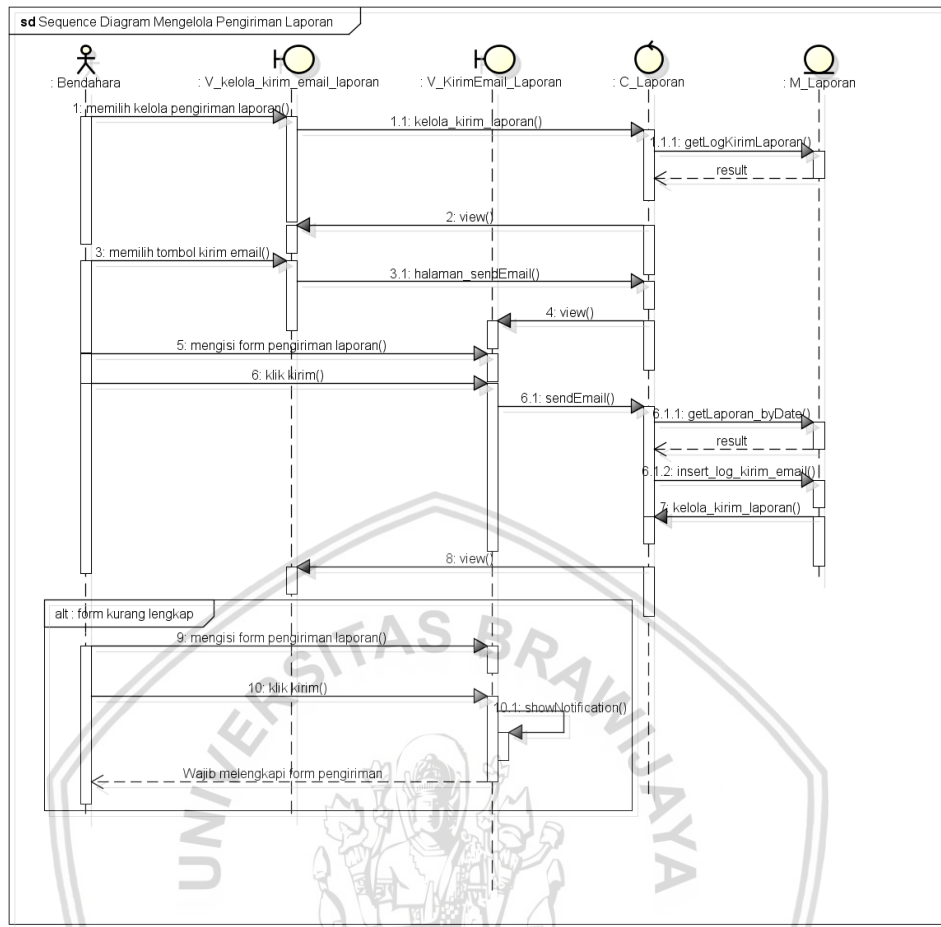


Gambar 5.10 Sequence diagram melihat laporan

Gambar 5.10 menjelaskan *sequence diagram* interaksi aktor bendahara dengan sistem dan komunikasi antar objeknya. Bendahara melakukan memilih laporan perBulan lalu *view* V_Laporan meneruskan dengan memanggil fungsi *ganti_bulan()* pada *controller* C_Laporan. Lalu *controller* C_Laporan memanggil fungsi yang ada pada M_Laporan dengan *getLaporanbyDate()* lalu *return* value kepada C_Laporan. C_Laporan memanggil fungsi *total_kredit_byDate* dan fungsi *total_debit_byDate* dari model M_Laporan lalu di kembalikan kepada bendahara dengan laporan per bulan.

5.3.5 Sequence diagram mengelola pengiriman laporan

Sequence diagram mengelola pengiriman laporan merupakan diagram yang menjelaskan bagaimana interaksi antar aktor bendahara terhadap objek terkhususkan pada operasi mengelola pengiriman laporan melalui sistem informasi keuangan. Pada rancangan *sequence diagram* mengelola pengiriman laporan terdapat aktor bendahara, boundary V_kelola_kirim_laporan dan V_KirimEmail_Laporan, *controller* C_Laporan dan entity M_Laporan. Rancangan *sequence diagram* terdapat pada Gambar 5.11.



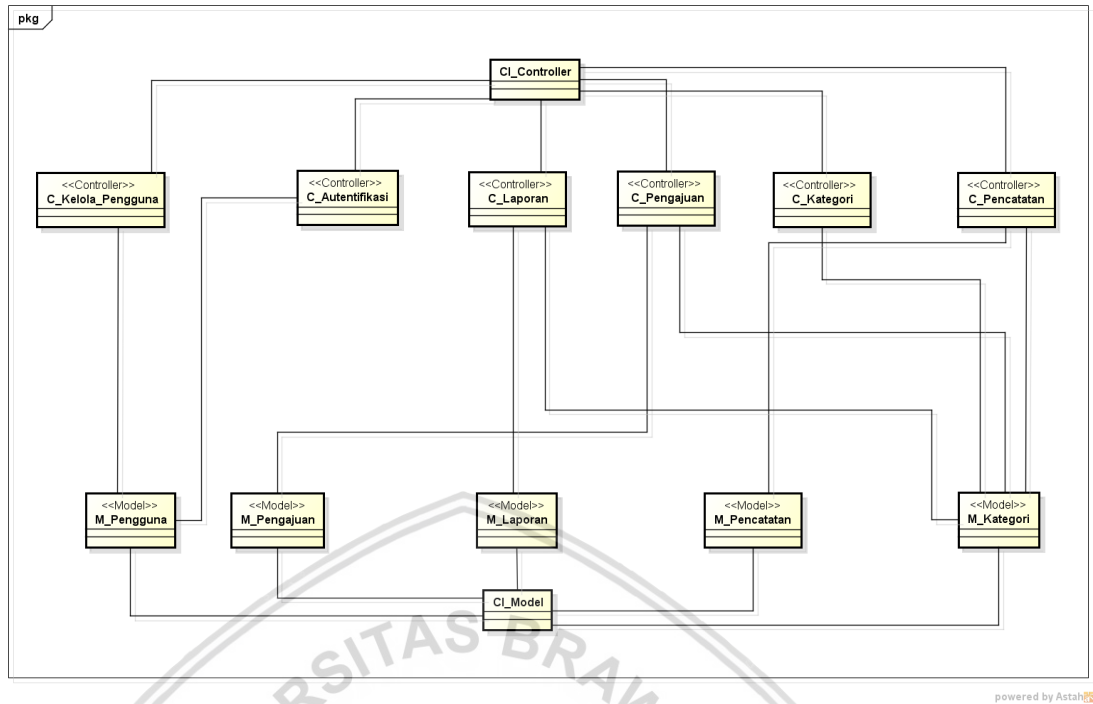
powered by Astah

Gambar 5.11 sequence diagram Mengelola Pengiriman Laporan

Gambar 5.11 merupakan penjelasan komunikasi antar objek pada aktifitas mengelola pengiriman laporan. Bendahara memilih kelola pengiriman laporan lalu memanggil kelola_kirim_laporan() pada C_Laporan lalu diteruskan dengan getKirimLaporan() lalu bendahara memilih tombol kirim email lalu diteruskan dengan pemanggilan halaman_sendEmail() pada controller C_Laporan lalu di kembalikan ke bendahara lalu bendahara mengisi data lalu memanggil sendEmail(). Alternatif terjadi jika *form* tidak valid untuk di kirim maka sistem memberikan notifikasi.

5.4 Perancangan Class Diagram

Pada perancangan class diagram mempresentasikan class, atribut, fungsi dan hubungan antar objek pada perangkat lunak. Dalam penggambaran *class diagram* di lakukan analisa objek. Method di analisa berdasarkan kebutuhan fungsional dari Tahap Analisa Kebutuhan. Hasil pengelompokan method maka di temukan class objek dan atribut dari ciri khas objek tersebut. Hasil penggambaran *class diagram* di muat pada gambar 5.12.



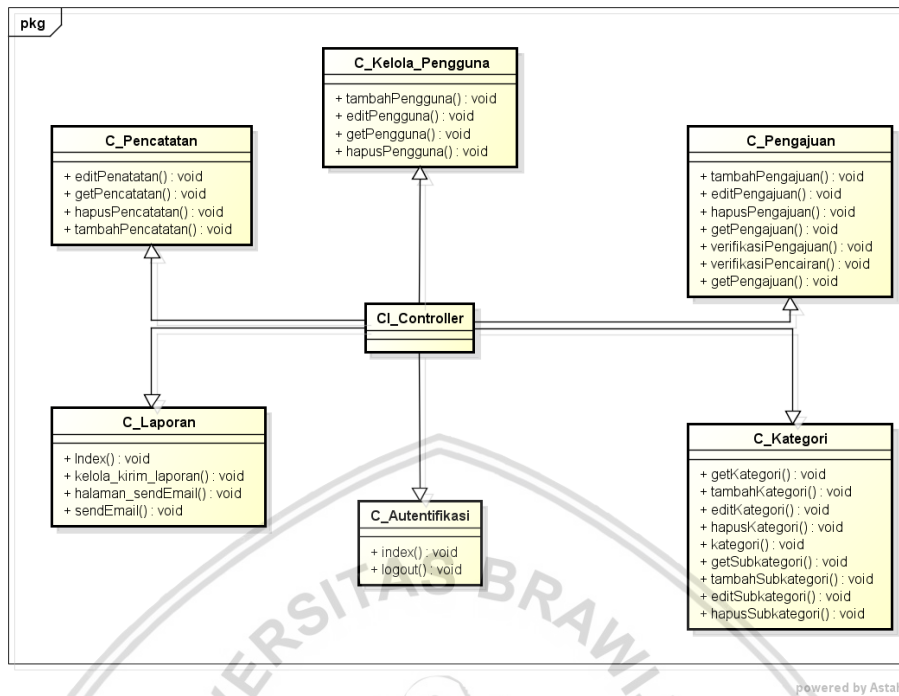
Gambar 5.12 Diagram Sistem Informasi Keuangan

Pada gambar 5.12 merupakan class diagram pada sistem informasi keuangan yang siap untuk di implementasi. *Class diagram* di sesuaikan dengan *framework codeigniter* yaitu konsep MVC (*Model-View-Controller*). Kelas *CI_Controller* di extend oleh kelas *C_Autentifikasi*, *C_Dashboard*, *C_Kelola_Pengguna*, *C_Laporan*, *C_Pengajuan*, *C_Kategori*, *C_Pencatatan*. Sedangkan kelas *CI_Model* di extend oleh *M_Pengguna*, *M_Pengajuan*, *M_Laporan*, *M_Pencatatan*, *M_Kategori*.

Pada kelas *C_Autentifikasi* berfungsi utama untuk melakukan autentifikasi dengan fungsi login dan logout berelasi dengan kelas *M_Pengguna* untuk melakukan proses pengecekan pengguna. Kelas *C_Kelola_pengguna* memiliki fungsi yang bertujuan untuk mengelola pengguna berelasi dengan kelas *M_Pengguna*. Pada kelas *C_Pengajuan* yang fungsinya secara garis besar untuk melakukan proses pengajuan dana berelasi dengan kelas *M_Pengajuan* dan *M_Kategori*. Kelas *C_Kategori* yang berfungsi untuk mengelola kategori keuangan berelasi dengan kelas *M_Kategori*. Pada kelas *C_Pencatatan* yang fungsinya untuk melakukan proses pencatatan dana berelasi dengan kelas *M_Kategori* dan *M_Pencatatan*. Kelas *C_Laporan* yang secara garis besar untuk menampilkan laporan berelasi dengan kelas *M_Laporan*. Dan kelas



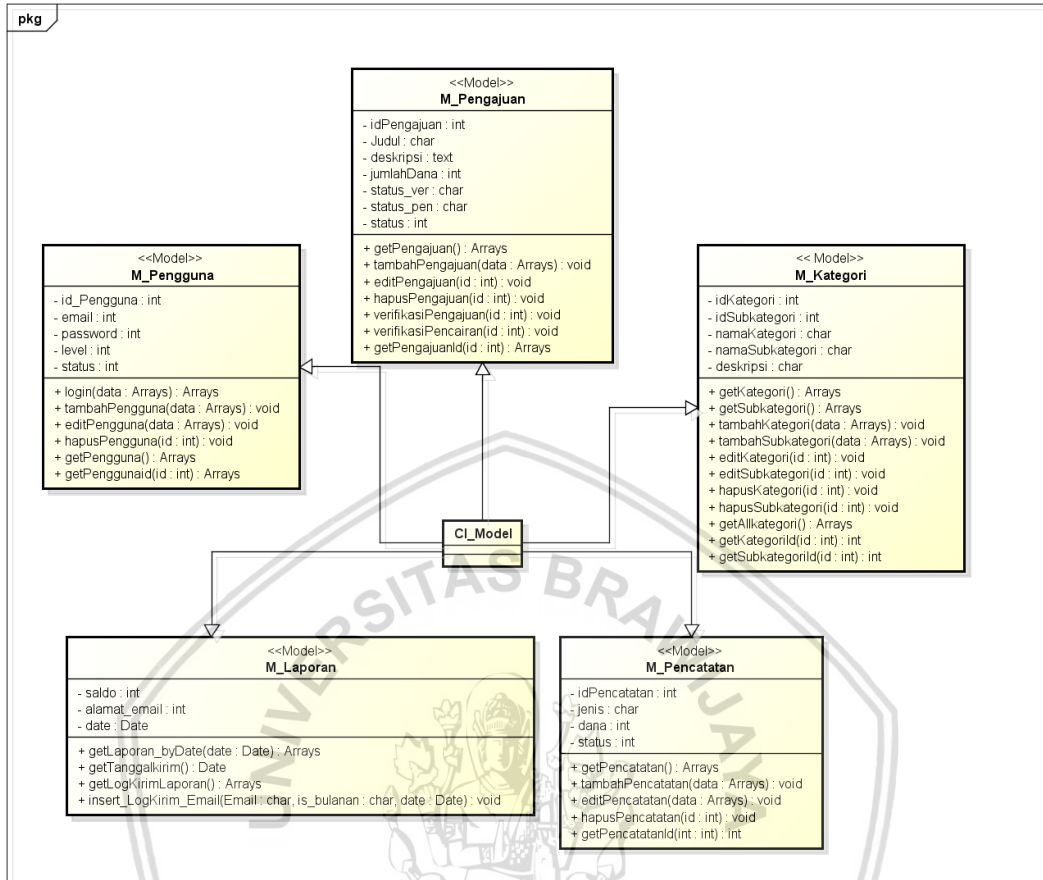
5.4.1 Class Diagram Relasi Antar Controller



Gambar 5.13 Class Diagram Relasi Antar Controller

Pada gambar 5.13 merupakan hasil rancangan class diagram yang menjelaskan relasi antar controller. Hasil rancangan memuat 6 objek class yang extend kepada kelas CI_Controller. 6 objek tersebut yakni C_Autentifikasi, C_Kategori, C_Pengajuan, C_Kelola_Pengguna, C_Pencatatan, C_Laporan. C_Autentifikasi memiliki atribut dan fungsi yang bertujuan agar sistem dapat menjalankan fungsi *Log in* dan *Log out*. C_Kategori memiliki atribut dan fungsi yang bertujuan agar sistem dapat menjalankan fungsi tambah, ubah dan hapus pada kategori keuangan. C_Kelola_Pengguna memiliki atribut dan fungsi yang bertujuan agar sistem dapat menjalankan fungsi tambah, kurang dan hapus pengguna. C_Pengajuan memiliki atribut dan fungsi yang bertujuan agar sistem dapat menjalankan fungsi tambah, ubah hapus pengajuan dana, verifikasi dan verifikasi pencairan dana. C_Pencatatan memiliki atribut dan fungsi yang bertujuan agar sistem dapat menjalankan fungsi tambah, ubah, hapus pencatatan dana. C_Laporan memiliki atribut dan fungsi yang bertujuan agar sistem dapat menjalankan fungsi menampilkan laporan dan mengelola pengiriman laporan.

5.4.2 Class Diagram Relasi Antar Model

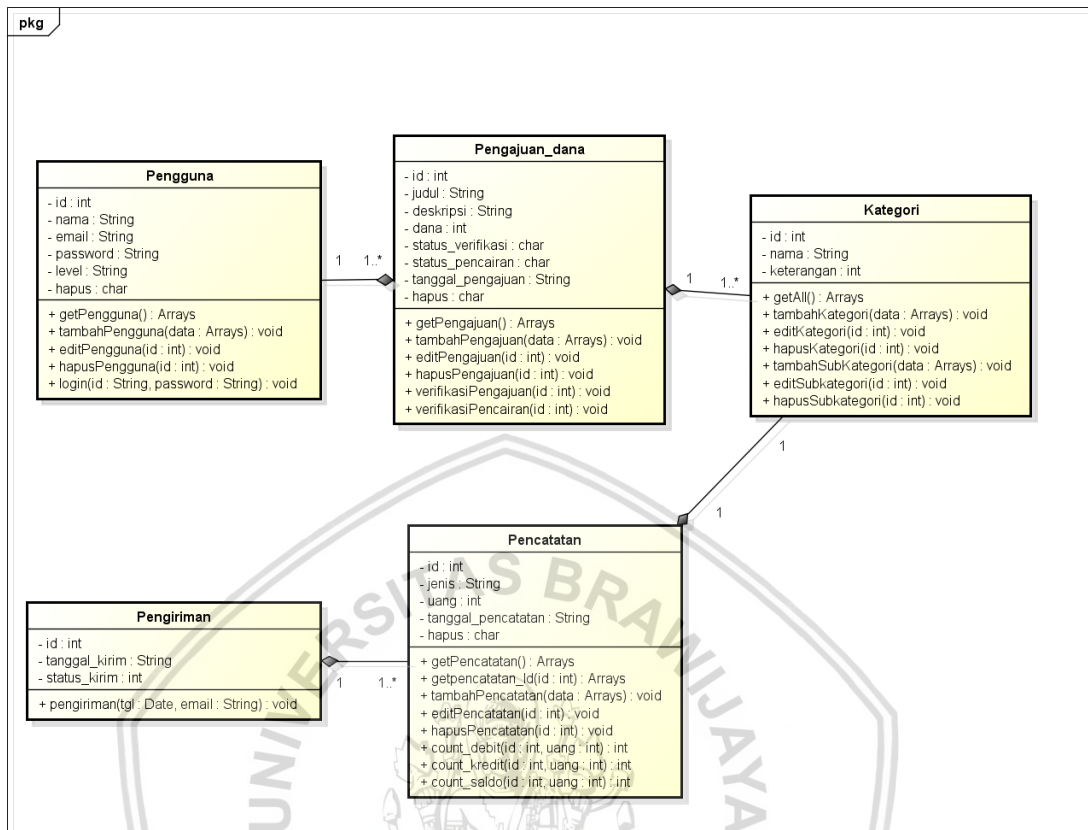


powered by Astah

Pada gambar 5.13 merupakan hasil rancangan class diagram yang menjelaskan relasi antar class model. Hasil rancangan memuat 5 objek class yang extend kepada kelas CI_Model. 5 objek tersebut yakni M_Kategori, M_Pengajuan, M_Kelola_Pengguna, M_Pencatatan, M_Laporan. M_Kategori memiliki atribut dan fungsi yang bertujuan agar sistem dapat mengakses basis data menjalankan fungsi tambah, ubah dan hapus pada kategori keuangan . M_Kelola_Pengguna memiliki atribut dan fungsi yang bertujuan agar sistem dapat mengakses basis data menjalankan fungsi tambah, kurang dan hapus pengguna. M_Pengajuan memiliki atribut dan fungsi yang bertujuan agar sistem dapat mengakses basis data menjalankan fungsi tambah, ubah hapus pengajuan dana, verifikasi dan verifikasi pencairan dana. M_Pencatatan memiliki atribut dan fungsi yang bertujuan agar sistem dapat mengakses basis data menjalankan fungsi tambah, ubah, hapus pencatatan dana. M_Laporan memiliki atribut dan fungsi yang bertujuan agar sistem mengakses basis data dapat menjalankan fungsi menampilkan laporan dan mengelola pengiriman laporan.



5.4.3 Class Diagram Domain Model



Gambar 5.14 Rancangan *Class Diagram Domain Model*

Gambar 5.13 merupakan rancangan *class diagram domain model*. *Class diagram domain model* dirancang berdasarkan class diagram relasi antar model dimana relasi antar model di pecah lagi dan di beri agresi. Class diagram tersebut menjadi model rancangan yang di gunakan untuk di implementasikan di dalam basis data. Terdapat 5 kelas yakni pengguna, pengajuan_dana, kategori, pencatatan serta pengiriman.

5.5 Pemetaan *Class Diagram Domain Model* ke *Physical Data Model*

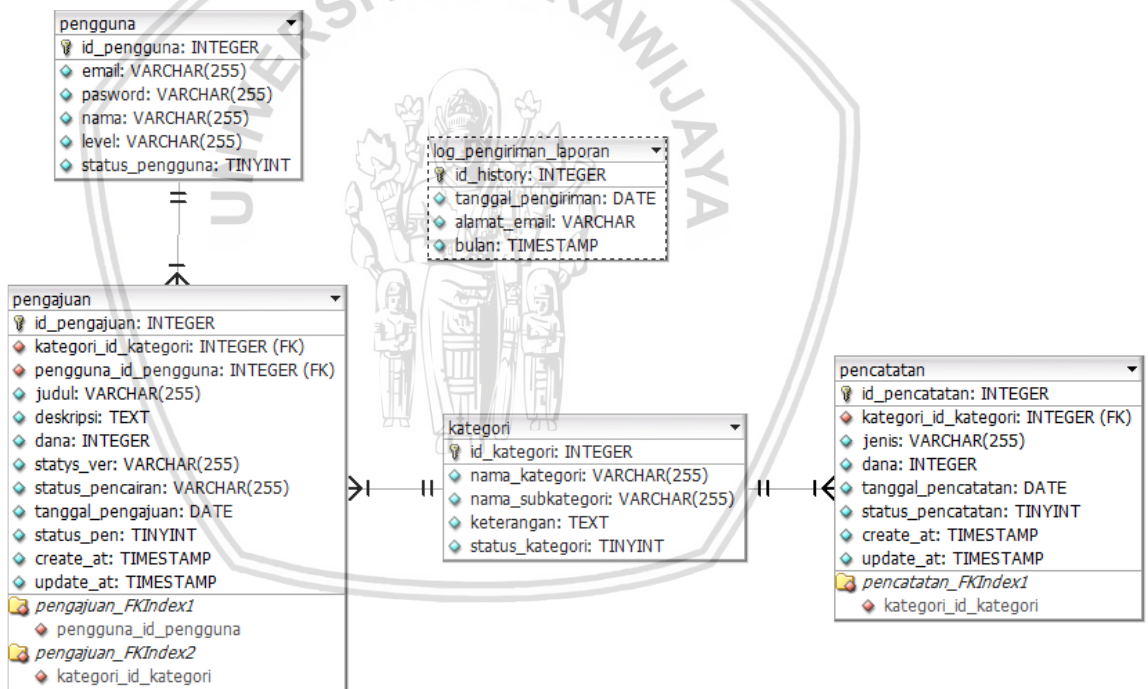
Perancangan *Physical Data Model diagram* menggambarkan basis data, relasi antar entitas dan variabel di dalamnya yang mengacu pada *domain model class diagram*. Entitas itu sendiri menggambarkan tabel yang berisi kolom dan baris yang dapat di simpan data di dalamnya.



Tabel 5.25 Pemetaan Class Diagram Domain Model ke Physical Data Model

No	Class pada class diagram domain model	Tabel pada Physical Data Model
1	Pengguna	Pengguna
2	Pengajuan_dana	Pengajuan
3	Kategori	Kategori
4	Pencatatan	Pencatatan
5	Pengiriman	Log_pengiriman_laporan

Tabel 5.25 merupakan tabel pemetaan setiap class pada domain model yang dipetakan menjadi tabel pada Physical Data Model. Gambar 5.14 merupakan gambar pemetaan class diagram domain model ke Physical Data Model dalam bentuk diagram.



Gambar 5.15 Physical Data Model sistem informasi keuangan

Pada gambar 5.8 merupakan rancangan Physical Data Model diagram pada sistem informasi keuangan. Terdapat 5 tabel yang masing-masing mempunyai atribut di dalamnya. Tabel pengguna berguna untuk menyimpan data pengguna yaitu atribut di dalamnya ialah id_pengguna, email, password, nama, level dan status. Tabel kategori berisikan tabel untuk menyimpan data kategori yang memiliki atribut id_kategori, nama_kategori, keterangan, status.



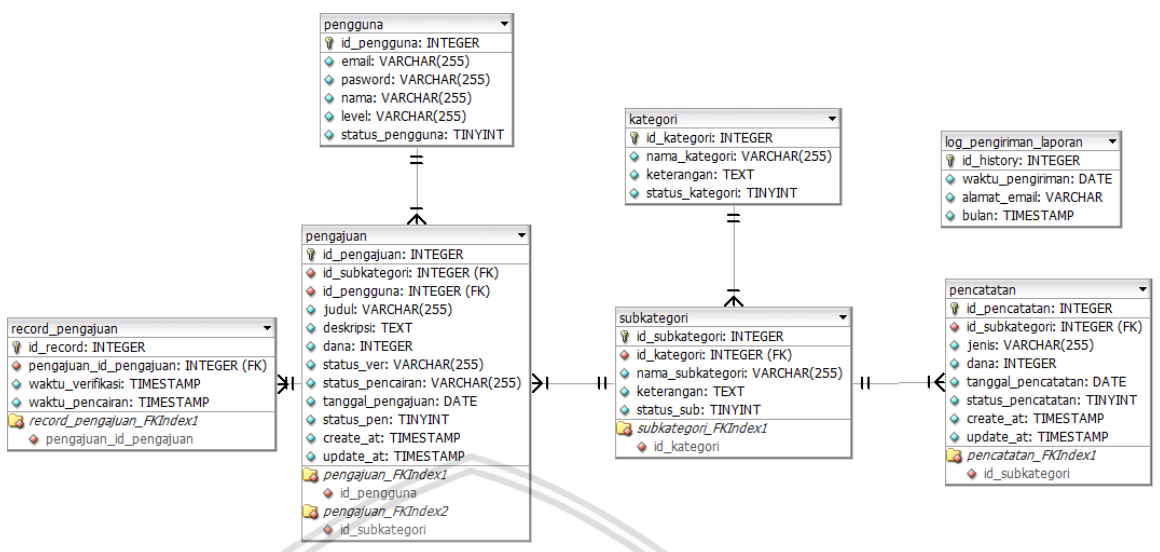
Tabel pengajuan merupakan tabel yang di gunakan untuk menyimpan data pengajuan dana dengan berelasi dengan tabel pengguna dan subkategori. Pada tabel pengajuan terdapat atribut `id_pengajuan`, `id_kategori`, `id_pengguna`, `judul`, `deskripsi`, `dana`, `status_ver`, `status_pencairan`, `tanggal_pengajuan`, `status`, `create_at` dan `update_at`. Tabel pencatatan merupakan tabel yang berfungsi untuk menyimpan data pencatatan dana yang memiliki atribut `id_pencatatan`, `id_kategori`, `jenis`, `dana`, `tanggal_pencatatan`, `status`, `create_at`, `update_at`. Tabel `Log_pengiriman_laporan` berisikan `id_history`, `tanggal_pengiriman`, `alamat_tujuan`, `bulan`.

5.6 Normalisasi *Physical Data Model*

Normalisasi *Physical Data Model diagram* dilakukan agar menghilangkan dan mengurangi redudansi data dan memastikan akan dependensi data tepat pada tempatnya. Pada Tabel 5.24 merupakan tabel sesudah dan sebelum proses normalisasi mengacu kepada *Physical Data Model* yang telah dipetakan dari *class domain diagram*.

Tabel 5.26 Normalisasi *Physical Data Model*

No	Tabel	Normalisasi Tabel
1	Pengguna	Pengguna
2	Pengajuan	Pengajuan
3	Pengajuan	Record_pengajuan
4	Kategori	Kategori
5	Kategori	Subkategori
6	Pencatatan	Pencatatan
7	Log_pengiriman_laporan	Log_pengiriman_laporan



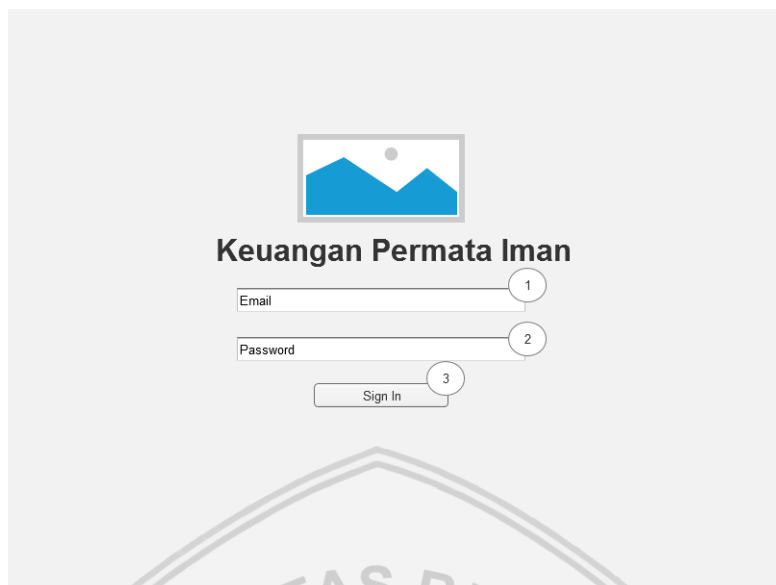
Gambar 5.16 Normalisasi Physical Data Model sistem informasi keuangan

Pada gambar 5.15 merupakan gambaran diagram *Physical Data Model*. Setelah dilakukannya normalisasi basis data pada sistem informasi keuangan di tambahnya 2 tabel yaitu subkategori dan record_pengajuan dimana subkategori tersebut memiliki atribut id_subkategori, nama_subkategori, keterangan, status_sub. Disini tabel subkategori di pisahkan dari tabel kategori agar menambah dependensi dan menghindari penambahan data yang tidak perlu pada tabel kategori saat belum di normalisasi. Tabel subkategori bereleasi dengan 4 tabel yaitu tabel pengajuan, tabel kategori, dan tabel pencatatan. Tabel record_pengajuan memiliki atribut id_record, id_pengajuan, waktu_verifikasi, waktu_pencairan.

5.7 Perancangan Antar Muka

Pada sub bab ini merupakan gambaran dan penjelasan mengenai rancangan antar muka pada sistem informasi keuangan KB & TK Permata Iman. Antar muka sistem informasi keuangan KB & TK Permata Iman di rancang agar memudahkan dalam pengembangan sistem terutama dalam permasalahan tampilan yang akan di bangun.

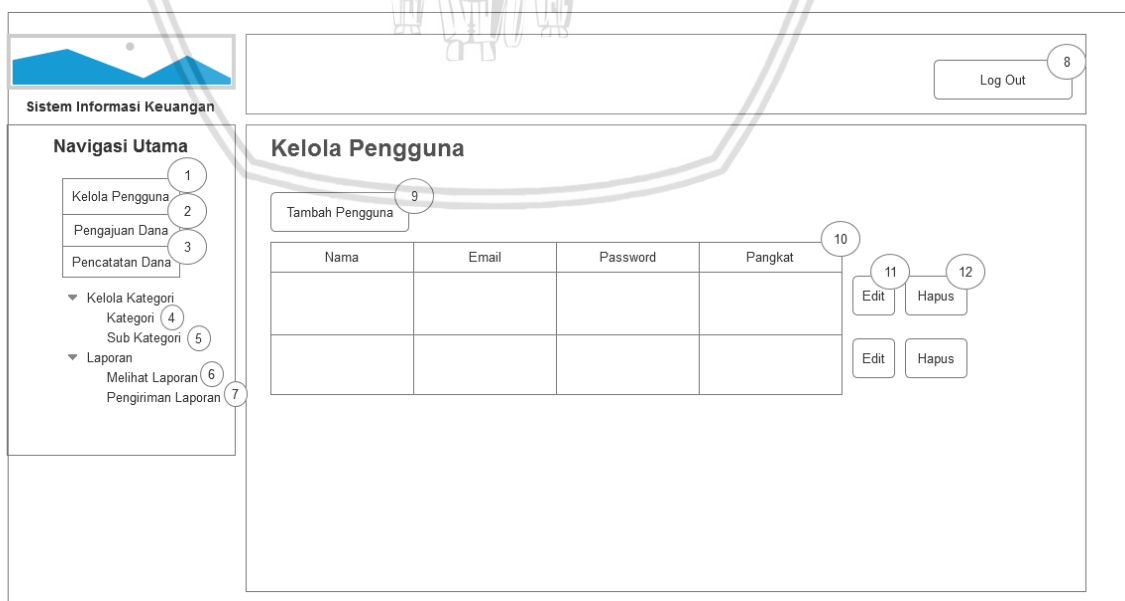
5.7.1 Halaman Log in



Gambar 5.17 Rancangan Antar Muka Login

Pada gambar 5.16 merupakan rancangan antar muka *login* pada sistem informasi keuangan KB & TK Permata Iman. Antar muka login merupakan tampilan awal untuk berinteraksi dengan sistem. (1) merupakan kolom untuk pengisian *email* sebagai tanda pengguna untuk memakai sistem. (2) merupakan kolom *password* untuk melihat apakah pengguna punya hak akses akan sistem. Dan (3) merupakan tombol *Sign In* untuk submit data yang telah di masukkan oleh pengguna dan melakukan proses autentifikasi.

5.7.2 Halaman Kelola Pengguna



Gambar 5.18 Rancangan antar muka kelola pengguna



pada gambar 5.17 menjelaskan rancangan antar muka pada halaman kelola pengguna. (1) merupakan menu untuk kelola pengguna. Pengguna dapat melakukan tambah pengguna, *edit* pengguna, lihat pengguna serta menghapus pengguna. (2) merupakan menu pengajuan dana untuk melihat pengajuan dana, tambah pengajuan dana, *edit* pengajuan dana dan hapus pengajuan dana. (3) merupakan menu pencatatan dana, pengguna dapat melihat *history* pencatatan dana, tambah pencatatan dana, *edit* pencatatan dana dan hapus pencatatan dana. (4) merupakan menu untuk mengelola kategori. (5) merupakan menu untuk mengelola sub kategori. (6) merupakan menu untuk melihat laporan keuangan. (7) merupakan menu untuk pengiriman laporan ke *e-mail* kepala sekolah per tanggal tertentu. (8) merupakan tombol *log out* untuk keluar dari sistem. (9) merupakan tombol untuk memanggil fungsi penambahan pengguna. (10) merupakan tabel keluaran dari basis data yang berisi data pengguna. (11) merupakan tombol *edit* dan (12) merupakan tombol hapus.

5.7.3 Form tambah pengguna

The screenshot shows the 'Tambah Pengguna' form within the 'Sistem Informasi Keuangan' application. On the left, there is a 'Navigasi Utama' menu with options: 'Kelola Pengguna', 'Pengajuan Dana', 'Pencatatan Dana', 'Kelola Kategori', 'Sub Kategori', and 'Laporan'. The main form area contains the following fields and controls:

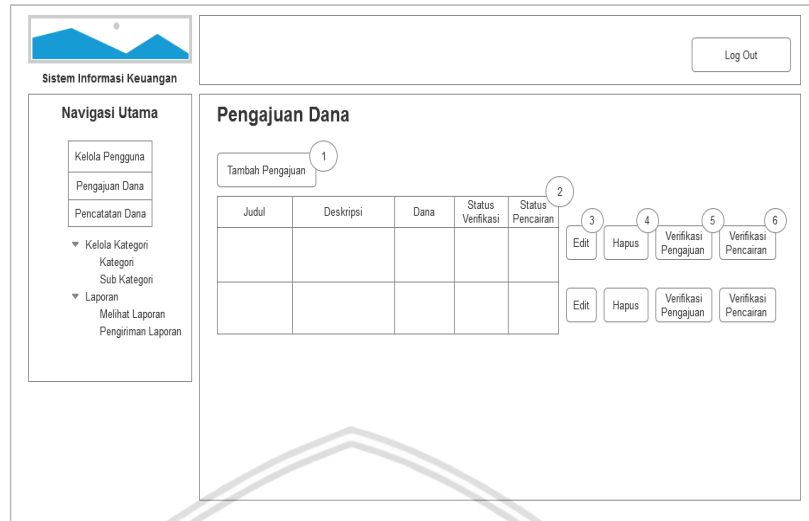
- 1**: Input field for 'Nama Pengguna'.
- 2**: Input field for 'Email'.
- 3**: Input field for 'Password'.
- 4**: Dropdown menu for 'Pangkat'.
- 5**: 'Simpan' button.

A 'Log Out' button is located in the top right corner of the application window.

Gambar 5.19 Rancangan antar muka form tambah pengguna

Gambar 5.18 merupakan rancangan antar muka form tambah pengguna. (1) adalah kolom untuk mengisi nama pengguna. (2) merupakan kolom untuk mengisi e-mail pengguna. (3) merupakan kolom untuk mengisi *password* pengguna dan (4) merupakan selektor memilih pangkat / jabatan pengguna. (5) merupakan tombol untuk simpan.

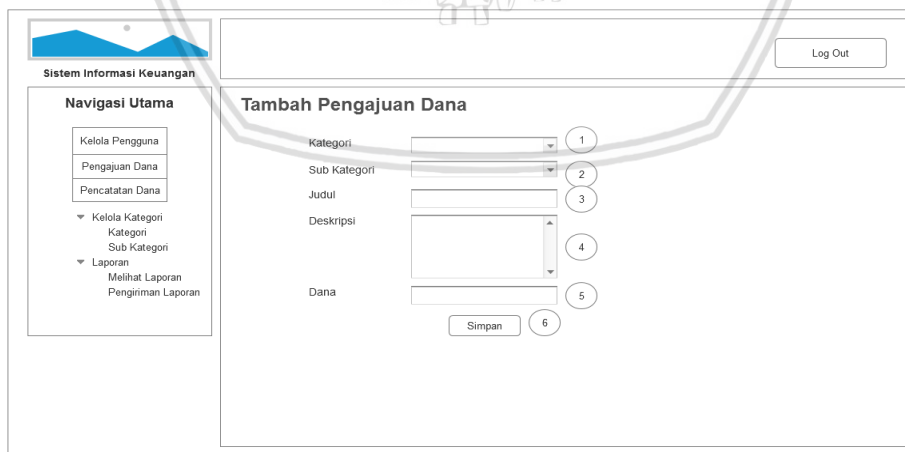
5.7.4 Melihat pengajuan dana



Gambar 5.20 Rancangan antar muka menu pengajuan dana

Gambar 5.19 merupakan rancangan antar muka pada menu pengajuan dana. (1) merupakan tombol tambah mengguna untuk memanggil *form* penambahan pengajuan dana dan menyimpan pengajuan dana. (2) merupakan tabel daftar pengajuan dana yang berisi kolom judul, deskripsi, dana, status verifikasi dan status pencairan. (3) merupakan tombol edit untuk memanggil halaman *form* edit. (4) merupakan tombol hapus untuk menghapus. (5) merupakan tombol verifikasi pengajuan yang di lakukan oleh jabatan kepala sekolah. (6) merupakan tombol untuk verifikasi pencairan dana dimana apakah bendahara sudah menyerahkan uang tersebut kepada peminta dana.

5.7.5 Form tambah pengajuan dana



Gambar 5.21 Rancangan Antarmuka Form tambah pengajuan dana

Gambar 5.20 merupakan rancangan antar muka *form* pengajuan dana. (1) merupakan pengguna untuk memilih kategori yang tersedia. (2) merupakan selektor dimana pengguna memilih sub kategori yang tersedia. (3) merupakan

kolom dimana pengguna mengisi dengan judul pengajuan dana. (4) merupakan kolom deskripsi yang diisi dengan kebutuhan peminta dan urgensi kebutuhan dalam pengajuan dana. (5) merupakan kolom yang diisi dengan total dana yang di butuhkan untuk pengajuan dana. (6) tombol simpan untuk menyimpan pengajuan dana.

5.7.6 Form pencatatan dana

Gambar 5.22 Rancangan Antarmuka Form Pencatatan Dana

Gambar 5.21 merupakan gambar rancangan antar muka tambah pencatatan dana. (1) merupakan kolom untuk memilih kategori yang tersedia. (2) merupakan kolom untuk memilih sub kategori yang tersedia. (3) merupakan pilihan yang di isi pengguna apakah pencatatan jenis debit atau kredit. (4) merupakan kolom total dana yang di catat dalam rupiah. (5) merupakan kolom tanggal yang diisi saat pencatatan. (6) merupakan tombol simpan yang di gunakan pengguna untuk menyimpan pencatatan dana.

5.7.7 Laporan keuangan

No	Tanggal	Kategori	Sub Kategori	Debet	Kredit	Saldo

Gambar 5.23 Rancangan Antarmuka Laporan Keuangan



Gambar 5.22 merupakan rancangan antar muka laporan keuangan. (1) merupakan kolom untuk memilih laporan pada bulan apa saja. (2) merupakan keluaran dari laporan yang berisi kolom untuk nomor, tanggal, kategori, sub kategori, debit, kredit, dan saldo. (3) merupakan kolom untuk total dari tabel (2) yaitu total debit, total kredit dan total saldo selama sebulan.

5.8 Perancangan Algoritme

Pada perancangan algoritme dijelaskan algoritme yang dipakai untuk diimplementasikan. Terdapat beberapa algoritme program yang akan dijelaskan dalam sub bab ini yakni algoritme menambah pengajuan dana, verifikasi pengajuan dana, menambah pencatatan, mengirim laporan.

5.8.1 Algoritme Menambah Pengajuan Dana

Pada sub bab ini menjelaskan rancangan algoritme dari fungsi yang di sediakan oleh sistem kepada pengguna saat pengguna menggunakan sistem untuk menambah pengajuan dana. Tabel 5.27 merupakan rancangan algoritme untuk menambah pengajuan dana.

Tabel 5.27 Pseudocode Menambah Pengajuan Dana

No	Pseudocode
1	Mulai
2	Jika isi level dan id pengguna adalah ` ` maka
3	Kembali ke halaman <i>log in</i>
4	Lainnya
5	Jalankan fungsi menambah pengajuan dana
6	Jalankan method subkategori dari model kategori
7	Mengambil data input dari <i>form</i> pengajuan dana
8	Jika input salah
9	Kembali ke halaman sebelumnya dengan pesan input yang salah
10	Lainnya
11	Jalankan fungsi menambah pengajuan dana dari model pengajuan
12	Memberikan notifikasi data berhasil di tambahkan
13	Selesai

5.8.2 Algoritme Verifikasi Pengajuan Dana

Pada sub bab ini menjelaskan rancangan algoritme dari fungsi yang di sediakan oleh sistem kepada pengguna saat pengguna menggunakan sistem untuk verifikasi pengajuan dana. Tabel 5.28 merupakan rancangan algoritme untuk verifikasi pengajuan dana.

Tabel 5.28 Pseudocode Verifikasi Pengajuan Dana

No	Pseudocode
1	Mulai
2	Jika isi level adalah kepek maka
3	Jalankan fungsi get Pengajuan dana
4	Memilih Pengajuan dana
5	Jalankan fungsi verifikasi pengajuan dana
6	Jalankan method verifikasi dari model pengajuan dana
7	Lainnya
8	Kembali ke halaman pengajuan dana
9	selesai

5.8.3 Algoritme Pencatatan Dana

Pada sub bab ini menjelaskan rancangan algoritme dari fungsi yang di sediakan oleh sistem kepada pengguna saat pengguna menggunakan sistem untuk menambah pencatatan dana. Tabel 5.29 merupakan rancangan algoritme untuk menambah pencatatan dana.

Tabel 5.29 Pseudocode Pencatatan Dana

No	Pseudocode
1	Mulai
2	Set Variabel Saldo sama dengan nilai 0
3	Jika isi level dan id pengguna adalah ` ` maka
4	Kembali ke halaman <i>log in</i>
5	Lainnya
6	Jalankan fungsi menambah pencatatan dana
7	Jalankan method subkategori dari model kategori
8	Menggambil data <i>form</i> pencatatan
9	Jika input kurang
10	Kembali ke halaman sebelumnya dengan pesan input yang kurang
11	Lainnya
12	Jalankan method tambahPencatatan dari model pencatatan
13	Memberikan notifikasi data berhasil di simpan
14	selesai

5.8.4 Algoritme Melihat Laporan Keuangan

Pada sub bab ini menjelaskan rancangan algoritme dari fungsi yang di sediakan oleh sistem kepada pengguna saat pengguna menggunakan sistem untuk melihat laporan keuangan. Tabel 5.30 merupakan rancangan algoritme untuk menambah pengajuan dana.

Tabel 5.30 Pseudocode Melihat Laporan Keuangan

No	Pseudocode
1	Mulai
2	Set Variabel Saldo sama dengan nilai 0
3	Jika isi level dan id pengguna adalah ` ` maka
4	Kembali ke halaman <i>log in</i>
5	Lainnya
6	Jalankan fungsi melihat laporan
7	Jalankan method total saldo
8	Menggambil data bulan dari pilihan bulan
9	Menjalankan fungsi menampilkan laporan berdasarkan bulan
10	selesai

5.8.5 Algoritme Mengelola Pengiriman Laporan Keuangan

Pada sub bab ini menjelaskan rancangan algoritme dari fungsi yang di sediakan oleh sistem kepada pengguna saat pengguna menggunakan sistem untuk menambah pencatatan dana. Tabel 5.31 merupakan rancangan algoritme untuk menambah pengajuan dana.

Tabel 5.31 Pseudocode Mengelola Pengiriman Laporan Keuangan

No	Pseudocode
1	Mulai
2	Jika isi level dan id pengguna adalah ` ` maka
3	Kembali ke halaman <i>log in</i>
4	Lainnya
5	Jalankan fungsi mendapatkan log pengiriman laporan
6	Mengambil data bulan dari pilihan bulan
7	Mengambil data input alamat dari <i>form</i> alamat email tujuan
8	Jika input alamat tujuan kurang dan salah
9	Kembali ke halaman sebelumnya dengan pesan input yang salah
10	Lainnya
11	Memanggil fungsi pengiriman email di controller laporan
12	Selesai

5.9 Perancangan Pengujian *Black-Box* Validasi

Perancangan pengujian validasi merupakan perancangan rencana uji untuk pengujian sistem setelah sistem berhasil di implementasikan. Pengujian pada bagian ini dilakukan dengan mendefinisikan kasus uji yang merujuk pada *use case scenario* yang telah didefinisikan. Kemudian setiap hasil dari pengujian yang dilakukan akan dibandingkan dengan hasil yang telah didefinisikan pada *use case scenario*. Berikut pada Gambar 5.24 merupakan aturan penomoran yang diterapkan untuk pengujian validasi.

PV – XX

Gambar 5.24 Aturan Penomoran Pengujian Validasi

Gambar 5.32 merupakan aturan penomoran fitur di butuhkan agar mempermudah *tracebilty*. Dan setiap elemennya akan dijelaskan pada Tabel 5.30

Tabel 5.32 Aturan Penomoran Pengujian Validasi

Aturan Penomoran	Deskripsi
PV	Pengujian Validasi
XX	Nomor Pengujian

5.9.1 Pengujian Validasi Membuat *Form* Pengajuan Dana

Bagian ini menjelaskan pengujian validasi pada beberapa fungsi yang dapat digunakan oleh pengguna untuk membuat *form* pengajuan dana melalui sistem yang telah dikembangkan. Pada Tabel 5.33 menunjukkan beberapa skenario yang bersumber dari alur *use-case* membuat *form* pengajuan dana yang akan digunakan untuk mengidentifikasi kasus uji.



Tabel 5.33 Matriks Skenario Pengujian Validasi Menambah *Form* Pengajuan Dana

No. Skenario	Alur Awal	Alur Alternatif
Skenario 1	<i>Basic Flow</i>	-
Skenario 2	<i>Basic Flow</i>	Menambah <i>form</i> pengajuan dana data tidak lengkap

5.9.1.1 Pengujian Validasi Membuat *Form* Pengajuan Dana

Pada Tabel 5.34 berisi informasi rencana pengujian dan kasus uji validasi dengan kode PV-01. Kasus uji PV-01 dilakukan untuk memastikan bahwa sistem dapat menambah *form* pengajuan dana. Kasus uji diidentifikasi berdasarkan skenario 1 pada Tabel 5.33.

Tabel 5.34 Rencana Pengujian dan Kasus Uji *Black-Box* Fungsi Menambah *Form* Pengajuan Dana

Kode Pengujian	PV-01
Nomor Skenario	Skenario 1
Kode Persyaratan	SIK-F-07
Tujuan Pengujian	Pengujian Validasi
Prosedur Uji	<ol style="list-style-type: none"> 1. Penguji teridentifikasi sebagai kepala bagian kurikulum. 2. Penguji memilih menu pengajuan dana. 3. Penguji memilih submenu tambah pengajuan dana. 4. Penguji mengisi <i>form</i> pengajuan dana. 5. Penguji memilih tombol simpan.
Hasil yang Diharapkan	Sistem berhasil menyimpan data pada <i>database</i> dan menampilkan pesan berhasil menyimpan.

5.9.1.2 Pengujian Validasi Membuat *Form* Pengajuan Dana Data Tidak Lengkap

Pada Tabel 5.35 berisi informasi rencana pengujian dan kasus uji validasi dengan kode PV-02. Kasus uji PV-02 dilakukan untuk memastikan bahwa sistem dapat menangani *form* pengajuan dana jika terdapat data yang kurang lengkap. Kasus uji diidentifikasi berdasarkan skenario 2 pada Tabel 5.33.

Tabel 5.35 Rencana Pengujian dan Kasus Uji *Black-Box* Fungsi Menambah *Form* Pengajuan Dana *Data Tidak Lengkap*

Kode Pengujian	PV-02
-----------------------	-------

Tabel 5.35 Rencana Pengujian dan Kasus Uji *Black-Box* Fungsi Menambah *Form* Pengajuan Dana *Data Tidak Lengkap* (lanjutan)

Nomor Skenario	Skenario 2
Kode Persyaratan	SIK-F-07
Tujuan Pengujian	Pengujian Validasi
Prosedur Uji	<ol style="list-style-type: none"> 1. Penguji teridentifikasi sebagai kepala bagian kurikulum. 2. Penguji memilih menu pengajuan dana. 3. Penguji memilih submenu tambah pengajuan dana. 4. Penguji mengisi <i>form</i> pengajuan dana menyisakan 1 kolom tidak terisi 5. Penguji memilih tombol simpan.
Hasil yang Diharapkan	Sistem menampilkan pesan notifikasi beberapa data harus diisi.

5.9.2 Pengujian Validasi Verifikasi Pengajuan Dana

Bagian ini menjelaskan pengujian validasi pada beberapa fungsi yang dapat digunakan oleh pengguna untuk memverifikasi pengajuan dana melalui sistem yang telah dikembangkan. Pada Tabel 5.36 menunjukkan beberapa skenario yang bersumber dari alur *use-case* verifikasi pengajuan dana yang akan digunakan untuk mengidentifikasi kasus uji.

Tabel 5.36 Matriks Skenario Pengujian Validasi Verifikasi Pengajuan Dana

No. Skenario	Alur Awal	Alur Alternatif
Skenario 1	<i>Basic Flow</i>	-
Skenario 2	<i>Basic Flow</i>	Membatalkan verifikasi pengajuan dana

5.9.2.1 Pengujian Validasi Verifikasi Pengajuan Dana

Pada 5.37 berisi informasi rencana pengujian dan kasus uji validasi dengan kode PV-03. Kasus uji PV-03 dilakukan untuk memastikan bahwa sistem dapat menambah *form* pengajuan dana. Kasus uji diidentifikasi berdasarkan skenario 1 pada Tabel 5.36.

Tabel 5.37 Rencana Pengujian dan Kasus Uji Validasi Fungsi Verifikasi Pengajuan Dana

Kode Pengujian	PV-03
Nomor Skenario	Skenario 1

Tabel 5.37 Rencana Pengujian dan Kasus Uji Validasi Fungsi Verifikasi Pengajuan Dana (lanjutan)

Kode Persyaratan	SIK-F-10
Tujuan Pengujian	Pengujian Validasi
Prosedur Uji	<ol style="list-style-type: none"> 1. Penguji teridentifikasi sebagai kepala sekolah 2. Penguji memilih menu pengajuan dana. 3. Penguji memilih pengajuan dana yang telah ada dan memilih tombol verifikasi 4. Penguji memilih status verifikasi 5. Penguji menekan tombol simpan
Hasil yang Diharapkan	Sistem berhasil menyimpan data pada <i>database</i> dan menampilkan pesan berhasil menyimpan.

5.9.2.2 Pengujian Validasi Fungsi Membatalkan Verifikasi Pengajuan Dana

Pada Tabel 5.38 berisi informasi rencana pengujian dan kasus uji validasi dengan kode PV-04. Kasus uji PV-04 dilakukan untuk memastikan bahwa sistem dapat melakukan penanganan jika pengguna ingin membatalkan proses verifikasi pengajuan. Kasus uji diidentifikasi berdasarkan skenario 2 pada Tabel 5.38.

Tabel 5.38 Rencana Pengujian dan Kasus Uji Validasi Fungsi Menambah Form Pengajuan Dana Data Tidak Lengkap

Kode Pengujian	PV-04
Nomor Skenario	Skenario 2
Kode Persyaratan	SIK-F-10
Tujuan Pengujian	Pengujian Validasi
Prosedur Uji	<ol style="list-style-type: none"> 1. Penguji teridentifikasi sebagai kepala bagian kurikulum. 2. Penguji memilih menu pengajuan dana. 3. Penguji memilih submenu tambah pengajuan dana. 4. Penguji memilih pengajuan dana yang telah ada dan memilih tombol verifikasi 5. Penguji memilih tombol batal
Hasil yang Diharapkan	Sistem menampilkan pesan notifikasi proses verifikasi di batalkan

5.9.3 Pengujian Validasi Membuat *Form* Pencatatan Dana

Bagian ini menjelaskan pengujian validasi pada beberapa fungsi yang dapat digunakan oleh pengguna untuk membuat *form* pencatatan dana melalui sistem yang telah dikembangkan. Pada Tabel 5.39 menunjukkan beberapa skenario yang bersumber dari alur *use-case* membuat *form* pencatatan dana yang akan digunakan untuk mengidentifikasi kasus uji.

Tabel 5.39 Matriks Skenario Pengujian Validasi Membuat *Form* Pencatatan Dana

No. Skenario	Alur Awal	Alur Alternatif
Skenario 1	<i>Basic Flow</i>	-
Skenario 2	<i>Basic Flow</i>	Menambah <i>form</i> pencatatan dana data tidak lengkap

5.9.3.1 Pengujian Validasi Membuat *Form* Pencatatan Dana

Pada Tabel 5.40 berisi informasi rencana pengujian dan kasus uji validasi dengan kode PV-05. Kasus uji PV-05 dilakukan untuk memastikan bahwa sistem dapat menambah *form* pencatatan dana. Kasus uji diidentifikasi berdasarkan skenario 1 pada Tabel 5.39.

Tabel 5.40 Rencana Pengujian dan Kasus Uji Validasi Fungsi Membuat *Form* Pencatatan Dana

Kode Pengujian	PV-05
Nomor Skenario	Skenario 1
Kode Persyaratan	SIK-F-10
Tujuan Pengujian	Pengujian Validasi
Prosedur Uji	<ol style="list-style-type: none"> 1. Penguji teridentifikasi sebagai bendahara 2. Penguji memilih menu pencatatan dana. 3. Penguji mengisi data pencatatan dana 4. Penguji memilih fungsi simpan
Hasil yang Diharapkan	Sistem berhasil menyimpan data pada <i>database</i> dan menampilkan pesan berhasil menyimpan.

5.9.3.2 Pengujian Validasi Fungsi Membuat Form Pencatatan Dana Data Tidak Lengkap

Pada Tabel 5.41 berisi informasi rencana pengujian dan kasus uji validasi dengan kode PV-06. Kasus uji PV-06 dilakukan untuk memastikan bahwa sistem dapat melakukan penanganan jika data yang di tambahkan oleh pengguna kurang lengkap. Kasus uji diidentifikasi berdasarkan skenario 2 pada Tabel 5.39.

Tabel 5.41 Rencana Pengujian dan Kasus Uji Validasi Fungsi Menambah Form Pengajuan Dana Data Tidak Lengkap

Kode Pengujian	PV-06
Nomor Skenario	Skenario 2
Kode Persyaratan	SIK-F-10
Tujuan Pengujian	Pengujian Validasi
Prosedur Uji	<ol style="list-style-type: none"> 1. Penguji teridentifikasi sebagai bendahara 2. Penguji memilih menu pencatatan dana. 3. Penguji mengisi <i>form</i> pencatatan dana tidak lengkap 4. Penguji memilih fungsi simpan
Hasil yang Diharapkan	Sistem menampilkan pesan notifikasi beberapa data harus diisi.

5.9.4 Pengujian Validasi Melihat Laporan Keuangan

Pada Tabel 5.42 berisi informasi rencana pengujian dan kasus uji validasi dengan kode PV-07. Kasus uji PV-07 dilakukan untuk memastikan bahwa sistem dapat menampilkan laporan keuangan.

Tabel 5.42 Rencana Pengujian dan Kasus Uji Validasi Melihat Laporan Keuangan

Kode Pengujian	PV-07
Nomor Skenario	Skenario 1
Kode Persyaratan	SIK-F-10
Tujuan Pengujian	Pengujian Validasi
Prosedur Uji	<ol style="list-style-type: none"> 1. Penguji teridentifikasi sebagai bendahara. 2. Penguji memilih menu laporan keuangan.
Hasil yang Diharapkan	Sistem berhasil menampilkan laporan keuangan.

5.9.5 Pengujian Validasi Mengelola Pengiriman Laporan

Bagian ini menjelaskan pengujian validasi pada beberapa fungsi yang dapat digunakan oleh pengguna untuk mengelola pengiriman laporan keuangan melalui sistem yang telah dikembangkan. Pada Tabel 5.43 menunjukkan beberapa skenario yang bersumber dari alur *use-case scenario* mengelola pengiriman laporan yang akan digunakan untuk mengidentifikasi kasus uji.

Tabel 5.43 Matriks Skenario Pengujian Validasi Mengelola Pengiriman Laporan Keuangan.

No. Skenario	Alur Awal	Alur Alternatif
Skenario 1	<i>Basic Flow</i>	-
Skenario 2	<i>Basic Flow</i>	Pengiriman laporan keuangan <i>form</i> tidak lengkap

5.9.5.1 Pengujian Validasi Mengelola Pengiriman Laporan Keuangan

Pada Tabel 5.43 berisi informasi rencana pengujian dan kasus uji validasi dengan kode PV-08. Kasus uji PV-08 dilakukan untuk memastikan bahwa sistem dapat menampilkan laporan keuangan. Kasus uji PV-08 di sesuaikan dengan skenario 1 pada Tabel 5.44.

Tabel 5.44 Rencana Pengujian dan Kasus Uji Validasi Mengelola Pengiriman Laporan Keuangan

Kode Pengujian	PV-08
Nomor Skenario	Skenario 1
Kode Persyaratan	SIK-F-21
Tujuan Pengujian	Pengujian Validasi
Prosedur Uji	<ol style="list-style-type: none"> 1. Penguji teridentifikasi sebagai bendahara. 2. Penguji memilih menu laporan keuangan. 3. Penguji memilih sub menu pengiriman laporan keuangan. 4. Penguji memilih alamat tujuan dan laporan bulan tertentu 5. Penguji memilih fungsi kirim
Hasil yang Diharapkan	Sistem berhasil mengirimkan laporan keuangan bulanan ke alamat email tertentu

5.9.5.2 Pengujian Validasi Mengelola Pengiriman Laporan Keuangan

Pada Tabel 5.45 berisi informasi rencana pengujian dan kasus uji validasi dengan kode PV-09. Kasus uji PV-09 dilakukan untuk memastikan bahwa sistem

dapat menampilkan *handler* pada pengiriman laporan keuangan jika alamat tujuan tidak tersedia. Kasus uji PV-09 di sesuaikan dengan skenario 2 pada Tabel 5.43.

Tabel 5.45 Rencana Pengujian dan Kasus Uji Validasi Mengelola Pengiriman Laporan Keuangan *Form* tidak lengkap

Kode Pengujian	PV-09
Nomor Skenario	Skenario 2
Kode Persyaratan	SIK-F-21
Tujuan Pengujian	Pengujian Validasi
Prosedur Uji	<ol style="list-style-type: none"> 1. Penguji teridentifikasi sebagai bendahara. 2. Penguji memilih menu laporan keuangan. 3. Penguji memilih sub menu pengiriman laporan keuangan. 4. Penguji mengosongkan kolom alamat email 5. Penguji memilih fungsi kirim 6. Sistem memberikan notifikasi bahwa alamat email harus di lengkapi dengan harus di isi
Hasil yang Diharapkan	Sistem berhasil memberikan <i>alert</i> bahwa alamat email harus diisi.

5.10 Perancangan Pengujian Penerimaan Pengguna

Perancangan Pengujian penerimaan pengguna dilakukan oleh penulis untuk memastikan bahwa sistem sudah memenuhi kriteria spesifikasi kebutuhan yang sudah dijelaskan sebelumnya dan dapat diterima oleh pengguna. Pengujian penerimaan pengguna akan dilakukan kepada *stakeholder* ataupun yang bertanggung jawab terhadap sistem informasi keuangan. Tabel 5.46 menjelaskan *stakeholder* yang terlibat beserta alasan dalam pengujian penerimaan pengguna.

Tabel 5.46 Stakeholder yang terlibat pada pengujian penerimaan pengguna

No	Stakeholders	Alasan
1	Kepala Sekolah	Supaya dapat mengelola data pengguna, dan mengoperasikan sistem yang akan digunakan
2	Bendahara	Supaya dapat mengoperasikan sistem yang akan digunakan

Tabel 5.46 Stakeholder yang terlibat pada pengujian penerimaan pengguna (lanjutan)

3	Peminta Dana	Supaya dapat mengoperasikan sistem yang akan digunakan
---	--------------	--

Kriteria user acceptance testing yang digunakan menyesuaikan dengan kebutuhan spesifikasi pengguna yaitu, *functional correctness and completeness*, *usability* dan *confidentiality and availability*. Pengujian akan dilaksanakan dengan menghampiri *tester*(penguji) untuk melakukan pengujian terhadap sistem yang telah dibuat. Berikut pada Gambar 5.25 merupakan aturan penomoran yang diterapkan untuk tugas pengujian penerimaan pengguna.

T-XX

Gambar 5.25 Aturan Penomoran Pengujian Penerimaan Pengguna

Gambar 5.25 merupakan aturan penomoran tugas di butuhkan agar mempermudah *tracebilty*. Dan setiap elemennya akan dijelaskan pada Tabel 5.47

Tabel 5.47 Aturan Penomoran Pengujian Validasi

Aturan Penomoran	Deskripsi
T	Tugas Penguji
XX	Nomor Pengujian

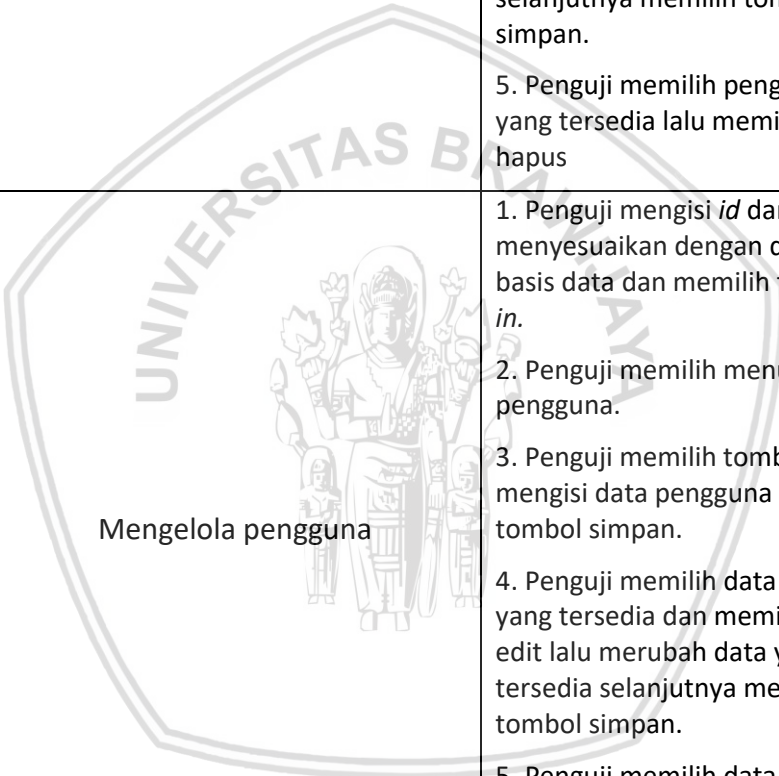
Selanjutnya masing-masing tester akan diberikan tugas sesuai dengan fungsi yang diberikan kepada masing-masing penguji. Sebelum penguji diberikan pertanyaan, penulis akan membantu penguji dalam menjalankan prosedur uji program menyesuaikan dengan tugas yang akan diberikan pada Tabel 5.48.

Tabel 5.48 Tugas Pengujian Penerimaan pengguna

Kode Tugas	Nama tugas	Prosedur Uji
T-01	Autentifikasi Pengguna	1. Penguji mengisi <i>id</i> dan <i>password</i> menyesuaikan dengan data dari basis data dan memilih tombol <i>sign in</i> . 2. Penguji memilih tombol <i>log out</i> .
T-02	Mengelola <i>form</i> pengajuan dana	1. Penguji mengisi <i>id</i> dan <i>password</i> menyesuaikan dengan data dari basis data dan memilih tombol <i>log in</i> .



Tabel 5.48 Tugas Pengujian Penerimaan pengguna (lanjutan)

Kode Tugas	Nama tugas	Prosedur Uji
		<p>2. Penguji memilih menu pengajuan dana.</p> <p>3. Penguji memilih tombol tambah, mengisi <i>form</i> pengajuan dana lalu memilih tombol simpan.</p> <p>4. Penguji memilih pengajuan yang tersedia dan memilih tombol edit lalu merubah data yang tersedia selanjutnya memilih tombol simpan.</p> <p>5. Penguji memilih pengajuan dana yang tersedia lalu memilih tombol hapus</p>
T-03	 <p>Mengelola pengguna</p>	<p>1. Penguji mengisi <i>id</i> dan <i>password</i> menyesuaikan dengan data dari basis data dan memilih tombol <i>log in</i>.</p> <p>2. Penguji memilih menu kelola pengguna.</p> <p>3. Penguji memilih tombol tambah, mengisi data pengguna lalu memilih tombol simpan.</p> <p>4. Penguji memilih data pengguna yang tersedia dan memilih tombol edit lalu merubah data yang tersedia selanjutnya memilih tombol simpan.</p> <p>5. Penguji memilih data pengguna yang tersedia lalu memilih tombol hapus</p>
T-04	Verifikasi pengajuan dana	<p>1. Penguji mengisi <i>id</i> dan <i>password</i> menyesuaikan dengan data dari basis data dan memilih tombol <i>log in</i>.</p> <p>2. Penguji memilih menu pengajuan dana.</p> <p>3. Penguji memilih pengajuan dana yang tersedia lalu memilih tombol verifikasi</p>

Tabel 5.48 Tugas Pengujian Penerimaan pengguna (lanjutan)

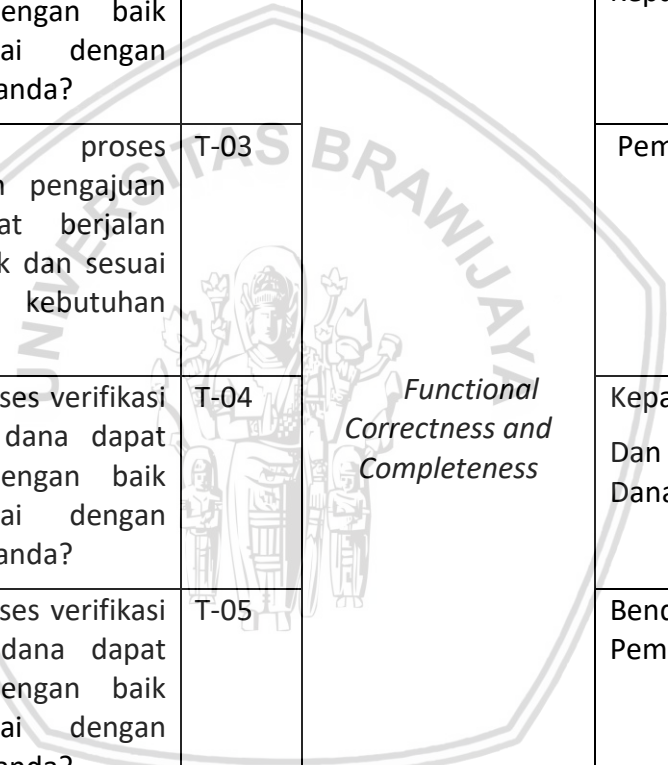
Kode Tugas	Nama tugas	Prosedur Uji
		4. Penguji memilih status verifikasi
T-05	Verifikasi pencairan dana	<ol style="list-style-type: none"> 1. Penguji mengisi <i>id</i> dan <i>password</i> menyesuaikan dengan data dari basis data dan memilih tombol <i>log in</i>. 2. Penguji memilih menu pengajuan dana. 3. Penguji memilih pengajuan dana yang tersedia lalu memilih tombol pencairan 4. Penguji memilih status pencairan
T-06	Mengelola pencatatan dana	<ol style="list-style-type: none"> 1. Penguji mengisi <i>id</i> dan <i>password</i> menyesuaikan dengan data dari basis data dan memilih tombol <i>log in</i>. 2. Penguji memilih menu pencatatan dana 3. Penguji memilih tombol tambah, mengisi data pencatatan lalu memilih tombol simpan. 4. Penguji memilih data pencatatan yang tersedia dan memilih tombol edit lalu merubah data yang tersedia selanjutnya memilih tombol simpan. 5. Penguji memilih data pencatatan yang tersedia lalu memilih tombol hapus
T-07	Mengelola kategori keuangan	<ol style="list-style-type: none"> 1. Penguji mengisi <i>id</i> dan <i>password</i> menyesuaikan dengan data dari basis data dan memilih tombol <i>log in</i>. 2. Penguji memilih menu kelola kategori. 3. Penguji memilih tombol tambah, mengisi data kategori lalu memilih tombol simpan.

Tabel 5.48 Tugas Pengujian Penerimaan pengguna (lanjutan)

Kode Tugas	Nama tugas	Prosedur Uji
		<p>4. Penguji memilih data kategori yang tersedia dan memilih tombol edit lalu merubah data yang tersedia selanjutnya memilih tombol simpan.</p> <p>5. Penguji memilih data kategori yang tersedia lalu memilih tombol hapus</p>
T-08	Melihat laporan keuangan	<p>1. Penguji mengisi <i>id</i> dan <i>password</i> menyesuaikan dengan data dari basis data dan memilih tombol <i>log in</i>.</p> <p>2. Penguji memilih menu laporan dan memilih submenu melihat laporan</p>
T-09	Mengelola pengiriman laporan keuangan	<p>1. Penguji mengisi <i>id</i> dan <i>password</i> menyesuaikan dengan data dari basis data dan memilih tombol <i>log in</i>.</p> <p>2. Penguji memilih menu laporan dan memilih sub menu kelola pengiriman laporan</p> <p>3. Penguji memilih tombol kirim email dan memilih bulan laporan selanjutnya mengisi ke alamat email tujuan mengakhiri dengan memilih tombol kirim.</p>
T-10	Pengiriman Laporan	<p>1. Penguji mengakses task <i>scheduler</i> dan menyetting task <i>task scheduler</i> dengan waktu yang di tentukan</p> <p>2. Penguji memilih file SIK.bat untuk di akses</p>

Tabel 5.49 berikut menjelaskan daftar pertanyaan pengujian penerimaan pengguna beserta *penguji* selaku yang menjawab pertanyaan setelah mengerjakan semua tugas pada Tabel 5.48.

Tabel 5.49 Tabel Pertanyaan Pengujian Penerimaan Pengguna

No	Pertanyaan	Kode Tugas	Kriteria	Responden
1	Apakah proses autentifikasi pengguna dapat berjalan dengan baik dan sesuai dengan kebutuhan anda?	T-01	 <p>Functional Correctness and Completeness</p>	Kepala Sekolah, Bendahara Peminta Dana
2	Apakah proses pengelolaan data pengguna dapat berjalan dengan baik dan sesuai dengan kebutuhan anda?	T-02		Kepala Sekolah
3	Apakah proses pengelolaan pengajuan dana dapat berjalan dengan baik dan sesuai dengan kebutuhan anda?	T-03		Peminta Dana
4	Apakah proses verifikasi pengajuan dana dapat berjalan dengan baik dan sesuai dengan kebutuhan anda?	T-04		Kepala sekolah Dan Peminta Dana
5	Apakah proses verifikasi pencairan dana dapat berjalan dengan baik dan sesuai dengan kebutuhan anda?	T-05		Bendahara dan Peminta Dana
6	Apakah proses pengelolaan pencatatan dana dapat berjalan dengan baik dan sesuai dengan kebutuhan anda?	T-06		Bendahara
7	Apakah proses pengelolaan data kategori keuangan dana dapat berjalan dengan	T-07		Bendahara



Tabel 5.49 Tabel Pertanyaan Pengujian Penerimaan Pengguna (lanjutan)

No	Pertanyaan	Kode Tugas	Kriteria	Responden
	baik dan sesuai dengan kebutuhan anda?			
8	Apakah proses melihat laporan keuangan dapat berjalan dengan baik dan sesuai dengan kebutuhan anda?	T-08		Bendahara dan Kepala sekolah
9	Apakah proses pengelolaan pengiriman laporan dapat berjalan dengan baik dan sesuai dengan kebutuhan anda?	T-09		Bendahara
10	Apakah proses pengiriman laporan dapat berjalan dengan baik dan sesuai dengan kebutuhan anda?	T-10		Bendahara
11	Apakah proses verifikasi pengajuan dana cepat dan mudah untuk dilakukan?			Kepala sekolah dan Peminta Dana
12	Apakah informasi yang diberikan kepada kepala sekolah sudah sesuai dengan kebutuhan kepala sekolah?	T-08	<i>Usability</i>	Kepala sekolah
13	Apakah anda sebagai pengguna senang dengan tampilan web yang ada saat ini?	T-01		Kepala sekolah, bendahara dan Peminta Dana
14	Apakah sistem informasi keuangan menyampaikan data akurat perihal penyampaian informasi	T-06 T-08		<i>Confidentiality and Availability</i>

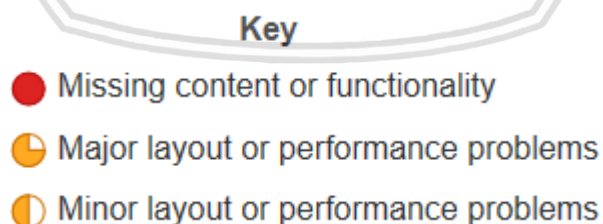


Tabel 5.49 Tabel Pertanyaan Pengujian Penerimaan Pengguna (lanjutan)

No	Pertanyaan	Kode Tugas	Kriteria	Responden
	terkait dengan data keuangan ?			
15	Apakah anda sebagai kepala sekolah merasa terjamin dengan keamanan data dari sistem informasi keuangan?	T-01		Kepala sekolah
16	Apakah anda sebagai bendahara merasa terjamin dengan keamanan data dari sistem informasi keuangan?	T-01		Bendahara

5.11 Perancangan Pengujian Compatibilty

Compatibility testing merupakan sebuah pengujian yang bertujuan untuk mengetahui kemampuan sistem dalam menunjukkan kesesuaian terhadap *web browser* yang digunakan. *Compability testing* di butuhkan agar dapat menguji kebutuhan non-fungsional yang telah di definisikan pada Tahap Analisis Kebutuhan. *Compatibility testing* dilakukan dengan menggunakan software SortSite Trial versi 5.29.818.0. Terdapat beberapa kategori yang ditunjukkan dalam bentuk simbol untuk mengetahui tingkat kompatibilitas yang dijelaskan dalam gambar 5.26.



Gambar 5.26 Simbol Tingkat Kompabilitas pada Compatibility Testing SortSite

Gambar 5.26 merupakan jenis tingkat masalah compability yang di sediakan oleh sortsite. Jenis permasalahan Tabel 5.50 merupakan kasus uji yang akan di lakukan dalam pengujian *compability*.

Tabel 5.50 Rencana Pengujian dan Kasus Uji *Compability* menggunakan *Sortsite*

Kode Pengujian	-
Software pengujian yang di gunakan	SortSite Trial versi 5.29.818.0
Software Pendukung	Xampp v3.2.1
Kode Persyaratan	SIK-NF-01
Tujuan Pengujian	Pengujian <i>Compability</i>
Prosedur Uji	<ol style="list-style-type: none"> 1. Penguji menyalakan Xampp dengan mencentang Apache server dan basis data Mysql 2. Penguji membuka <i>software sortsite</i> 3. Penguji memilih menu file new site check 4. Penguji memilih alamat sistem yakni localhost/SIK/index.php 5. Penguji mencentang pengecekan via <i>entire site</i> 6. Penguji memilih tombol <i>check</i>
Hasil yang Diharapkan	Hasil pengujian pada menu <i>compability</i> sistem dapat berjalan tanpa masalah pada browser/sistem operasi yang di sediakan oleh <i>sortsite</i> .

BAB 6 IMPLEMENTASI SISTEM

Pada bab ini berisikan penjelasan tentang spesifikasi pengembangan sistem, batasan-batasan implementasi, implementasi algoritme, implementasi basis data, dan implementasi antarmuka pengguna.

6.1 Spesifikasi Sistem

Spesifikasi sistem merupakan lingkungan implementasi dalam pembuatan sistem ini yang terdiri dari spesifikasi perangkat keras dan perangkat lunak.

6.1.1 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan dalam pengembangan sistem informasi keuangan KB & TK Permata Iman dapat dilihat di Tabel 6.1.

Tabel 6.1 Spesifikasi Perangkat Lunak

Nama Komponen	Spesifikasi
<i>Operating System</i>	Windows 10 64-bit
Bahasa Pemrograman	PHP 5
<i>Framework</i>	Codeigniter
<i>Text Editor</i>	Sublime Text
<i>Web server</i>	Xampp v.3.2.1
<i>Database</i>	MySQL
<i>Task Scheduler</i>	Task Scheduler Windows

6.2 Batasan Implementasi

Batasan implementasi pada pembangunan sistem ini adalah sebagai berikut:

1. *Database Management System* yang digunakan pada sistem ini adalah mysql Ver 14.14 Distrib 5.7.17, for Windows (x86_64).
2. Versi bahasa pemrograman PHP yang digunakan dalam membangun sistem ini adalah versi 7.0.8.
3. Fungsi pengiriman oleh *scheduler*, alamat pengirim dan alamat yang dituju bersifat statis(tidak dapat di rubah melalui sistem).

6.3 Implementasi Algoritme

Subbab implementasi algoritme berisi beberapa kode program yang dibuat berdasarkan algoritme dalam *pseudocode* yang telah dibuat pada tahap perancangan. Hasil implementasi algoritme dikelompokkan ke dalam beberapa bagian yang berkaitan dengan proses menambah pengajuan dana, verifikasi pengajuan dana, menambah pencatatan dana, melihat laporan keuangan, dan pengiriman laporan keuangan.

6.3.1 Implementasi Menambah Pengajuan Dana

Implementasi menambah pengajuan dana bernama `tambahPengajuan` yang ada pada controller `C_Pengajuan`. Pada proses menambah pengajuan dana terdapat beberapa langkah yang harus di penuhi sebelum data berhasil di simpan di dalam basis data. Sistem melakukan pengecekan apakah pengguna dapat menggunakan penambahan pengajuan dana karena hal ini hanya dapat di lakukan oleh peminta dana.

Setelah melakukan pengecekan dan pengguna *valid* maka akan di lanjutkan ke halaman penambahan pengajuan dana. Pada proses penambahan pengajuan dana pengguna memilih subkategori yang cocok untuk jenis pengajuan dananya lalu mengisi judul serta kebutuhan data untuk pengajuan dana tersebut. Fungsi tersebut terdapat pada implementasi Tabel 6.2.

Tabel 6.2 Implementasi Menambah Pengajuan Dana

```

1 public function tambahPengajuan()
2 {
3     $session_data = $this->session->userdata('penggunasistem');
4     $id_pengguna= $session_data['id_pengguna'];
5     $level=$session_data['level'];
6     $data['id_pengguna']=$id_pengguna;
7     $data['level']=$level;
8     if ($level == null) {
9         redirect();
10    }
11    $data['subkategori']= $this->M_Kategori->getSubkategori()-
12 >result();
13    $this->load->view('V_Header',$data);
14    $this->load->view('V_TambahPengajuan',$data);
15    $this->load->view('V_Footer',$data);
16    if ($this->input->post()) {
17        $id_pengguna = $this->input->post('id_pengguna');
18        $status_ver = $this->input->post('status_ver');
19        $status_pencairan = $this->input->post('status_pencairan');
20        $id_subkategori = $this->input->post('id_subkategori');
21        $judul = $this->input->post('judul');
22        $deskripsi = $this->input->post('deskripsi');
23        $dana = $this->input->post('dana');
24        $tanggal_pengajuan = $this->input-
25 >post('tanggal_pengajuan');
26        $result = $this->M_Pengajuan-
27 >tambahPengajuan($id_subkategori, $id_pengguna, $judul, $deskripsi,
28 $dana, $status_ver, $status_pencairan, $tanggal_pengajuan);
29        if ($result) {
30            $this->Sys_notif->success('Pengajuan berhasil
31 ditambahkan');

```

Tabel 6.2 Implementasi Menambah Pengajuan Dana (lanjutan)

32	}
33	redirect('C_Pengajuan/getPengajuan');
34	
35	} else {
36	
37	}
38	}

Penjelasan hasil *Source Code* implemtasi pada Tabel 6.2 akan dijelaskan pada Tabel 6.3.

Tabel 6.3 Penjelasan Implementasi Menambah Pengajuan Dana

Line of Code	Penjelasan
1	Insiasi fungsi tambahPengajuan
3-10	Insiasi pemanggilan session dan menyimpan pada array data. Proses pengecekan session data jika level pengguna null maka akan kembali ke halaman <i>log in</i> .
11-12	Pemanggilan fungsi getSubkategori pada model kateogri dan menyimpannya pada attribut data yang bertipe array
13-15	Pemanggilan view V_header, V_TambahPengajuan, V_Footer dengan parsing attribut data.
19-25	Pengecekan jika ada nilai post maka akan menyimpan nilai post tersebut ke nilai atribut id_pengguna, status_ver, status_pencairan, id_subkategori, judul, deskripsi, dana, tanggal_pengajuan.
26-28	Menyimpan nilai atribut id_pengguna, status_ver, status_pencairan, id_subkategori, judul, deskripsi, dana, tanggal_pengajuan pada attribut result lalu memanggil fungsi tambahPengajuan pada model Pengajuan.
29-32	Pengecekan jika nilai result ada maka memanggil Sys_notif success yakni memberikan alert bahwa penambahan pengajuan dana telah berhasil.
33-35	Jika nilai result tidak ada makan akan di redirect ke halaman pengajuan dana dengan pemanggilan fungsi getPengajuan pada controller C_Pengajuan.
37	<i>Bracket</i> akhir dari fungsi post.
38	<i>Bracket</i> akhir dari fungsi tambahPengajuan.

6.3.2 Implementasi Verifikasi Pengajuan Dana

Implementasi fungsi verikasi bernama verifikasiPengajuan yang ada pada controller C_Pengajuan. Pada proses verifikasi pengajuan dana terdapat beberapa

langkah yang harus di penuhi sebelum data berhasil di simpan di dalam basis data. Sistem melakukan pengecekan apakah pengguna dapat melakukan verifikasi pengajuan dana karena hanya dapat dilakukan oleh kepala sekolah.

Setelah melakukan pengecekan dan pengguna *valid* maka akan di lanjutkan ke halaman pengajuan dana. Pada proses penambahan verifikasi pengajuan dana pengguna memilih pengajuan dana lalu mengisi status verifikasi. Fungsi tersebut terdapat pada implementasi Tabel 6.4.

Tabel 6.4 Implementasi Verifikasi Pengajuan Dana

```

1 public function verifikasiPengajuan($id =null, $status_ver){
2
3     if ($this->input->post('verifikasi')) {
4         $status_ver = "Belum Terverifikasi";
5     }
6     elseif ($this->input->post('ditolak')) {
7         $status_ver = "Ditolak";
8     }
9
10    $session_data = $this->session->userdata('penggunasistem');
11    $id_pengguna= $session_data['id_pengguna'];
12    $level=$session_data['level'];
13    $data['id_pengguna']=$id_pengguna;
14    $data['level']=$level;
15    if ($level == null) {
16        redirect();
17    }
18
19    $result = $this->M_Pengajuan-
20 >verifikasiPengajuan($id,$status_ver);
21    redirect('C_Pengajuan/getPengajuan');
22 }
23

```

Penjelasan hasil *Source Code* implementasi pada Tabel 6.4 akan dijelaskan pada Tabel 6.5.

Tabel 6.5 Penjelasan Implementasi Verifikasi Pengajuan Dana

Line of Code	Penjelasan
1	Insiasi fungsi verifikasiPengajuan dengan paramater id dan status_ver.
4-10	Melakukan pengecekan jika terdapat nilai post verifikasi maka merubah atribut status_ver menjadi terverifikasi. Jika nilai post ditolak maka merubah atribut status_ver menjadi ditolak.
11-17	Insiasi pemanggilan session dan menyimpan pada array data. Proses pengecekan session data jika level pengguna null maka akan kembali ke halaman <i>log in</i> .
18-22	Menyimpan atribut id dan status_ver pada result dan memanggil fungsi verifikasiPengajuan pada Model M_Pengajuan lalu redirect ke halaman pengajuan dana.



Tabel 6.5 Penjelasan Implementasi Verifikasi Pengajuan Dana (lanjutan)

Line of Code	Penjelasan
23	<i>Bracket</i> akhir dari fungsi verifikasiPengajuan.

6.3.3 Implementasi Menambah Pencatatan Dana

Implementasi fungsi menambah pencatatan dana bernama `tambahPencatatan` yang ada pada controller `C_Pencatatan`. Pada proses menambah pencatatan dana terdapat beberapa langkah yang harus di penuhi sebelum data berhasil di simpan di dalam basis data. Sistem melakukan pengecekan apakah pengguna dapat menggunakan penambahan pencatatan dana karena hal ini hanya dapat di lakukan oleh bendahara.

Setelah melakukan pengecekan dan pengguna *valid* maka akan di lanjutkan ke halaman penambahan pencatatan dana. Pada proses penambahan pencatatan dana pengguna memilih subkategori yang cocok untuk jenis pencatatan dananya lalu mengisi jenis pencatatan yakni debit atau kredit serta data kebutuhan untuk pencatatan dana tersebut. Fungsi tersebut terdapat pada implementasi Tabel 6.6.

Tabel 6.6 Implementasi Menambah Pencatatan Dana

```

1 public function tambahPencatatan()
2 {
3     $session_data = $this->session->userdata('penggunasistem');
4     $id_pengguna= $session_data['id_pengguna'];
5     $level=$session_data['level'];
6     $data['id_pengguna']=$id_pengguna;
7     $data['level']=$level;
8     if ($level == null) {
9         redirect();
10    }
11    $data['subkategori']= $this->M_Kategori->getSubkategori();
12    $this->load->view('V_Header',$data);
13    $this->load->view('V_TambahPencatatan',$data);
14    $this->load->view('V_Footer',$data);
15    if ($this->input->post()) {
16        $id_subkategori    = $this->input->post('id_subkategori');
17        $jenis            = $this->input->post('jenis');
18        $dana             = $this->input->post('dana');
19        // karena tidak ada di view yang dikirimkan sebelumnya
20        // $deskripsi      = $this->input->post('deskripsi');
21        // end
22        $dana             = $this->input->post('dana');
23        $tanggal_pencatatan = date('Y-m-d',      strtotime($this->input->
24        >post('tanggal_pencatatan')));
25        $result          = $this->M_Pencatatan->
26        >tambahPencatatan($id_subkategori,      $jenis,      $dana,
27        $tanggal_pencatatan);
28
29        if ($result == 'success') {
30            $this->Sys_notif->success('Pencatatan dana berhasil ditambahkan');
31        } else {
32            $this->Sys_notif->danger('Pencatatan dana gagal ditambahkan karena
33            tidak sesuai preferensi sistem');
34        }

```



Tabel 6.6 Implementasi Menambah Pencatatan Dana (lanjutan)

35	
36	
37	<code>redirect('C_Pencatatan/getPencatatan');</code>
38	
39	<code>}</code>
40	<code>}</code>

Penjelasan hasil *Source Code* implementasi pada Tabel 6.6 akan dijelaskan pada Tabel 6.7.

Tabel 6.7 Penjelasan Implementasi Menambah Pencatatan Dana

Line of Code	Penjelasan
1	Insiasi fungsi tambahPencatatan
3-10	Insiasi pemanggilan session dan menyimpan pada array data. Proses pengecekan session data jika level pengguna null maka akan kembali ke halaman <i>log in</i> .
11-12	Pemanggilan fungsi getSubkategori pada model kategori dan menyimpannya pada atribut data yang bertipe array
13-14	Pemanggilan view V_header, V_TambahPencatatan, V_Footer dengan parsing atribut data.
15-24	Pengecekan jika ada nilai post maka akan menyimpan nilai post tersebut ke nilai atribut id_subkategori, jenis, dana.
26-28	Menyimpan nilai atribut id_pengguna, status_ver, status_pencairan, id_subkategori, judul, deskripsi, dana, tanggal_pengajuan pada atribut result lalu memanggil fungsi tambahPengajuan pada model Pengajuan.
29-31	Pengecekan jika nilai result success ada maka memanggil Sys_notif danger yakni memberikan alert bahwa penambahan pengajuan dana telah gagal.
32-34	Pengecekan jika nilai result tidak success maka memanggil Sys_notif success yakni memberikan alert bahwa penambahan pengajuan dana telah berhasil.
35-40	<i>Bracket</i> akhir dari fungsi post, mengarahkan ke halaman getPecatatan dan <i>Bracket</i> akhir dari fungsi tambahPencatatan

6.3.4 Implementasi Melihat Laporan Keuangan

Implementasi fungsi melihat laporan bernama index yang ada pada controller C_Laporan. Pada proses melihat laporan keuangan terdapat beberapa langkah yang harus di penuhi. Sistem melakukan pengecekan apakah pengguna dapat

menggunakan penambahan pengajuan dana karena hal ini hanya dapat dilakukan bendahara dan kepala sekolah.

Setelah melakukan pengecekan dan pengguna *valid* maka akan di lanjutkan ke halaman laporan keuangan. Pada proses melihat laporan keuangan pengguna memilih filter waktu yang diinginkan dan sistem akan menampilkan laporan keuangan. Fungsi tersebut terdapat pada implementasi Tabel 6.8.

Tabel 6.8 Implementasi Melihat Laporan Keuangan

```

1 public function index()
2 {
3     $session_data = $this->session->userdata('penggunasistem');
4     $level=$session_data['level'];
5     $data['level']=$level;
6
7     if ($level == null) {
8         redirect();
9     }
10
11    if ($this->input->post('bulan')) {
12        $data_bulan = $this->input->post('bulan');
13    } else {
14        $data_bulan = $this->ganti_bulan(date('m'));
15    }
16
17    $data['bulan'] = $data_bulan;
18    $data['get_data_laporan_filtered'] = $this->M_Laporan-
19 >getLaporan_byDate($data_bulan);
20
21    $this->load->view('V_Header', $data);
22    $this->load->view('V_Laporan', $data);
23    $this->load->view('V_Footer', $data);
24 }

```

Penjelasan hasil *Source Code* implementasi pada Tabel 6.8 akan dijelaskan pada Tabel 6.9.

Tabel 6.9 Penjelasan Implementasi Melihat Laporan Keuangan

Line of Code	Penjelasan
1	Insiasi fungsi index
2-10	Insiasi pemanggilan session dan menyimpan pada array data. Proses pengecekan session data jika level pengguna null maka akan kembali ke halaman <i>log in</i> .
11-16	Pengecekan jika terdapat input post bulan maka menyimpannya pada attribut data_bulan jika tidak maka memanggil fungsi ganti_bulan pada C_Laporan dengan paramater m.
17-20	Menyimpan data_bulan pada array data llalu memanggil fungsi getLaporan_byDate dari model M_Laporan dengan parameter data_bulan.



Tabel 6.9 Penjelasan Implementasi Melihat Laporan Keuangan (lanjutan)

Line of Code	Penjelasan
17-20	Pengecekan jika ada nilai post maka akan menyimpan nilai post tersebut ke nilai atribut <code>id_pengguna</code> , <code>status_ver</code> , <code>status_pencairan</code> , <code>id_subkategori</code> , <code>judul</code> , <code>deskripsi</code> , <code>dana</code> , <code>tanggal_pengajuan</code> .
21-24	Pemanggilan view <code>V_Header</code> , <code>V_Laporan</code> , <code>V_Footer</code> dengan parsing atribut data dan di akhiri dengan <i>Bracket</i> akhir dari fungsi <code>index</code>

6.3.5 Implementasi Mengelola Pengiriman Laporan Keuangan

Proses pengiriman laporan keuangan di muat dalam fungsi `sendEmail`. Fungsi tersebut memanggil library email lalu melakukan pengisian data yang dibutuhkan dalam pengiriman seperti email penerima dan laporan tertentu. Fungsi tersebut terdapat pada source code Tabel 6.10.

Tabel 6.10 Implementasi Mengelola Pengiriman Laporan

1	<code>public function kelola_kirim_laporan()</code>
2	<code>{</code>
3	<code> \$session_data = \$this->session->userdata('penggunasistem');</code>
4	<code> \$id_pengguna= \$session_data['id_pengguna'];</code>
5	<code> \$level=\$session_data['level'];</code>
6	<code> \$data['level']=\$level;</code>
7	
8	<code> if (\$level == null) {</code>
9	<code> redirect();</code>
10	<code> }</code>
11	<code> \$data['log_pengiriman']= \$this->M_Laporan-</code>
12	<code>>getLogKirimLaporan();</code>
13	<code> \$this->load->view('V_Header',\$data);</code>
14	<code> \$this->load->view('V_kelola_kirim_email_laporan', \$data);</code>
15	<code> \$this->load->view('V_Footer',\$data);</code>
16	<code>}</code>

Penjelasan hasil *Source Code* implementasi pada Tabel 6.10 akan dijelaskan pada Tabel 6.11.

Tabel 6.11 Penjelasan Implementasi Pengiriman Laporan

Line of Code	Penjelasan
1	Insiasi fungsi <code>kelola_kirim_laporan</code>
3-10	Insiasi pemanggilan session dan menyimpan pada array data. Proses pengecekan session data jika level pengguna null maka akan kembali ke halaman <i>log in</i> .
11-12	Pemanggilan fungsi <code>getLogKirimLaporan</code> pada model <code>M_Laporan</code>



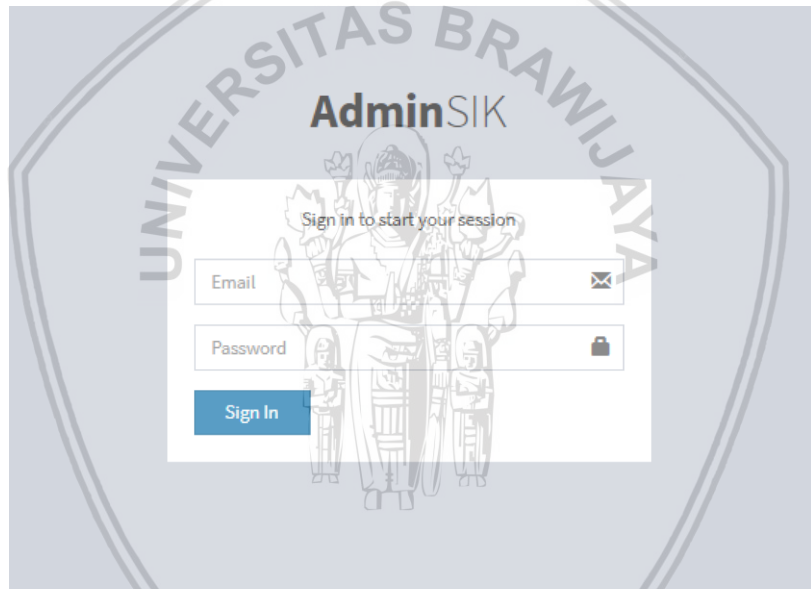
Tabel 6.11 Penjelasan Implementasi Pengiriman Laporan (lanjutan)

Line of Code	Penjelasan
13-16	Pemanggilan view V_header, V_kelola_kirim_email_laporan, V_Footer dengan parsing atribut data. <i>Bracket</i> akhir dari fungsi <i>kelola_kirim_laporan</i> .

6.4 Implementasi Antarmuka Pengguna

Subbab implementasi antarmuka berisi gambar hasil dari implementasi sistem informasi keuangan. Hasil implementasi antar muka diantaranya antarmuka *login*, antarmuka kelola pengguna, antarmuka pengajuan dana, antarmuka tambah pengajuan dana, antarmuka pencatatan dana, antar muka laporan keuangan.

6.4.1 Antarmuka *Login*



Gambar 6.1 Hasil implementasi *login*

Gambar 6.1 merupakan gambar hasil implementasi *login* yang dijalankan pada *browser* pada *personal computer* saat menjalankan sistem informasi keuangan untuk menjalankan fungsi otentikasi identitas pengguna. Kolom *Email* di isi menurut email berdasarkan kelola pengguna dan *password* untuk di cocokan. Tombol *Sign In* untuk menjalankan fungsi otentikasi pengguna sistem.

6.4.2 Antarmuka Kelola Pengguna



Gambar 6.2 Hasil implementasi antarmuka kelola pengguna

Gambar 6.2 merupakan hasil implementasi antarmuka kelola pengguna yang hanya dapat di akses oleh jabatan kepala sekolah. Hasil dari otentikasi yang sukses akan menampilkan komponen antarmuka seperti *sidebar* di sisi kiri, lalu tampilan isi dari setiap menu dari *sidebar* dan tombol *logout*.

Antarmuka kelola pengguna terdapat judul menu, tombol tambah pengguna, kolom nama, email, *password*, pangkat lalu tombol *edit* dan hapus. Tombol tambah pengguna untuk menjalankan fungsi menambah pengguna, tombol *edit* untuk menjalankan mengubah pengguna dan tombol hapus untuk menjalankan fungsi menghapus pengguna.

6.4.3 Antarmuka Pengajuan Dana



Gambar 6.3 Hasil implementasi antarmuka pengajuan dana

Gambar 6.3 merupakan hasil implementasi antarmuka kelola pengguna yang hanya dapat di akses oleh pengguna yang telah terdaftar. Hasil dari otentikasi yang sukses akan menampilkan komponen antarmuka seperti *sidebar* di sisi kiri, lalu tampilan isi menu dari *sidebar* dan tombol *logout*.

Antarmuka pengajuan dana berisikan komponen judul dari menu, tombol tambah untuk menjalankan fungsi menambah *form* pengajuan dana, tombol *edit* untuk menjalankan fungsi mengubah *form* pengajuan dana dan tombol hapus

untuk menjalankan fungsi menghapus pengajuan dana. Antarmuka pengajuan dana juga menampilkan judul pengajuan dana, sub kategori, deskripsi, dana, tanggal dibutuhkan, status verifikasi, status pencairan dari setiap pengajuan dana yang telah dibuat.

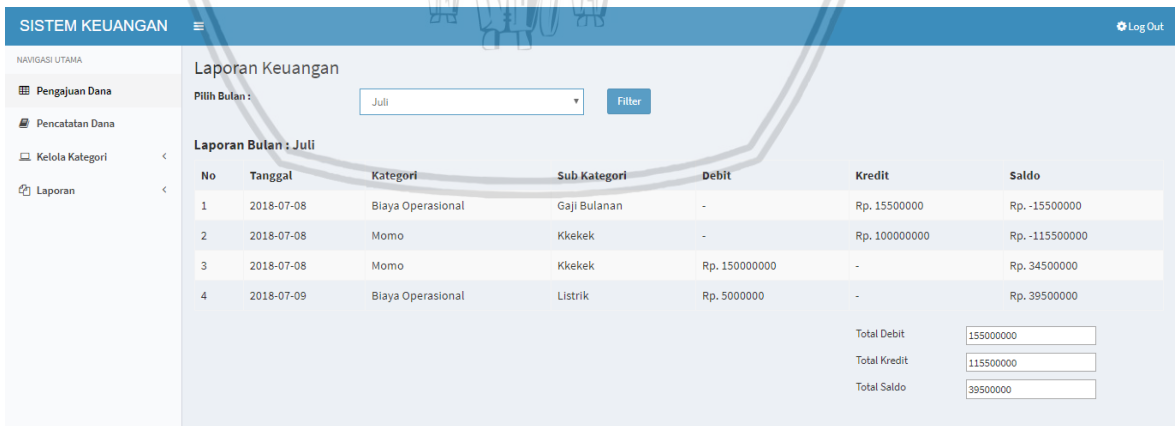
6.4.4 Antarmuka Pencatatan Dana



Gambar 6.4 Antarmuka Pencatatan Dana

Gambar 6.4 merupakan hasil implementasi antarmuka pencatatan dana pada sistem informasi keuangan. Tombol tambah untuk menjalankan fungsi tambah pencatatan, tombol edit untuk menjalankan fungsi ubah pencatatan. Tampilan utama menampilkan tabel tanggal pencatatan, kategori, subkategori jenis pencatatan lalu jumlah dana.

6.4.5 Antarmuka Mengelola Laporan Keuangan



Gambar 6.5 Antarmuka Laporan Keuangan

Gambar 6.5 merupakan antarmuka laporan keuangan. Tampilan laporan keuangan menampilkan hasil query dari pencatatan dana berdasarkan filter waktu yang di tentukan pengguna. Laporan keuangan melampirkan keterangan pencatatan berdasarkan tanggal dan menghasilkan total perhitungan.



BAB 7 PENGUJIAN DAN ANALISIS HASIL

Pengujian merupakan representasi dari Tahap Pengujian pada *Waterfall Model* yang mana merupakan proses untuk memeriksa apakah solusi yang ditawarkan perangkat lunak tersebut telah memenuhi persyaratan dan spesifikasi yang telah disepakati di awal serta memenuhi tujuannya. Pembahasan pada bab pengujian dan analisis hasil ini diuraikan menjadi pengujian *black-box* validasi, pengujian penerimaan pengguna dan pengujian *compatibility*.

7.1 Pengujian *Black-box*

Pengujian *black-box* merupakan pengujian yang dilakukan pada antarmuka dari perangkat lunak dan berfokus untuk mencari kesalahan mengenai fungsi yang salah atau hilang, kesalahan antarmuka, kesalahan dalam struktur data serta kesalahan perilaku atau kinerja sistem. Pengujian pada bagian ini dilakukan dengan mendefinisikan kasus uji yang merujuk pada Tahap Perancangan terkait rancangan kasus uji. Metode pengujian *Black-box* yang di gunakan pada penelitian ini adalah pengujian validasi.

7.1.1 Hasil Pengujian Validasi Sistem Informasi Keuangan

Pengujian validasi merupakan pengujian yang dilakukan pada antarmuka dari perangkat lunak dan berfokus kepada kebutuhan fungsional dari perangkat lunak tersebut. Pengujian ini berusaha untuk mencari kesalahan mengenai fungsi yang salah atau hilang, kesalahan antarmuka, kesalahan dalam struktur data serta kesalahan perilaku atau kinerja sistem. Pengujian pada bagian ini dilakukan dengan mendefinisikan kasus uji yang merujuk pada *use case scenario* yang telah didefinisikan pada Tahap Analisis Kebutuhan. Kemudian setiap hasil dari pengujian yang dilakukan akan dibandingkan dengan hasil yang telah didefinisikan pada *use case scenario*. Pengujian validasi dilakukan oleh pengembang untuk memastikan sistem siap untuk di *delivery* kepada *stackholder*.

Tabel 7.1 Hasil pengujian validasi sistem informasi keuangan

No	Kode Pengujian	Hasil yang di harapkan	Hasil Pengujian	Status Uji
1	PV-01	Sistem berhasil menyimpan data pada <i>database</i> dan menampilkan pesan berhasil menyimpan.	Sistem berhasil menyimpan data pada <i>database</i> dan menampilkan pesan berhasil menyimpan.	Valid
2	PV-02	Sistem menampilkan pesan notifikasi beberapa data harus diisi.	Sistem menampilkan pesan notifikasi beberapa data harus diisi.	Valid

Tabel 7.1 Hasil pengujian validasi sistem informasi keuangan (lanjutan)

No	Kode Pengujian	Hasil yang di harapkan	Hasil Pengujian	Status Uji
3	PV-03	Sistem berhasil menyimpan data pada <i>database</i> dan menampilkan pesan berhasil menyimpan.	Sistem berhasil menyimpan data pada <i>database</i> dan menampilkan pesan berhasil menyimpan.	Valid
4	PV-04	Sistem menampilkan pesan notifikasi proses verifikasi di batalkan	Sistem menampilkan pesan notifikasi proses verifikasi di batalkan	Valid
5	PV-05	Sistem berhasil menyimpan data pada <i>database</i> dan menampilkan pesan berhasil menyimpan.	Sistem berhasil menyimpan data pada <i>database</i> dan menampilkan pesan berhasil menyimpan.	Valid
6	PV-06	Sistem menampilkan pesan notifikasi beberapa data harus diisi.	Sistem menampilkan pesan notifikasi beberapa data harus diisi.	Valid
7	PV-07	Sistem berhasil menampilkan laporan keuangan.	Sistem berhasil menampilkan laporan keuangan.	Valid
8	PV-08	Sistem berhasil mengirimkan laporan keuangan bulanan ke alamat email tertentu.	Sistem berhasil mengirimkan laporan keuangan bulanan ke alamat email tertentu.	Valid
9	PV-09	Sistem berhasil memberikan <i>alert</i> bahwa alamat email harus diisi.	Sistem berhasil memberikan <i>alert</i> bahwa alamat email harus diisi.	Valid

7.2 Hasil Pengujian Penerimaan Pengguna

Subbab ini merupakan hasil pengujian penerimaan pengguna untuk melihat *feedback* dari pengguna dan *stackholder* terkait sistem yang telah di implementasikan. Merujuk pada Tahap Perancangan Pengujian Penerimaan Pengguna didapatkan kriteria dan pertanyaan yang sesuai dengan kriteria yang dibutuhkan. Tahap selanjutnya adalah dengan menganalisa hasil wawancara yang terlampir pada Lampiran B Hasil Pengujian Penerimaan Pengguna. Pada Tabel 7.2 menunjukkan rangkuman hasil wawancara pengujian penerimaan pengguna.

Tabel 7.2 Rangkuman Hasil Wawancara Pengujian Penerimaan Pengguna

R	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	Total
R1	1	1	1	1	1	1	1	1	1	1	10
R2	1	1	1	1	1	1	1	0	1	1	9
R3	1	1	1	1	1	1	x	x	x	x	5

Tabel 7.2 merupakan hasil wawancara dari pengujian penerimaan pengguna elemennya akan dijelaskan Pada tabel 7.3.

Tabel 7.3 Penjelasan Elemen Rangkuman Hasil Wawancara Pengujian Penerimaan Pengguna

Kode	Deskripsi
R	Responden
P	Pertanyaan
x	Tidak memiliki pertanyaan
1	Jawaban Setuju
0	Jawaban Tidak Setuju

7.3 Hasil Pengujian *Compatibility*

Subbab ini merupakan hasil pengujian *Compatibility* dengan kasus uji merujuk pada Tahap Perancangan Pengujian *Compatibility*. Pengujian di lakukan dengan menginput halaman akses pengguna pada *software SortSite Trial* versi 5.29.818.0. Hasil pengujian di tampilkan pada Gambar 7.1.

		http://localhost/SIK/																		
		Compatibility																		
		This tab shows pages that exhibit browser-specific behavior, or trigger browser bugs.																		
Browser	Version	Internet Explorer		Edge	Firefox	Safari		Opera	Chrome	iOS		Android	BlackBerry							
		6	7	8	9	10	11	16	60	≤ 10	11	51	66	≤ 9	10	11	≤ 3	4*	≤ 7.1	10.0
Critical Issues		✓	✓	✓	✓	✓	✓	✓	●	✓	✓	✓	●	✓	✓	✓	✓	✓	✓	✓
Major Issues		⚠	⚠	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Minor Issues		⚠	⚠	⚠	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	⚠	✓

* Most Android devices from 4.1 onwards use Chrome as the default browser, older versions use the original Android browser

Gambar 7.1 Hasil Pengujian *Compatibility* Menggunakan *SortSite Trial* versi 5.29.818.0

7.4 Analisis Hasil Pengujian

Sub bab ini menjelaskan kesimpulan berdasarkan hasil pengujian yang telah dilakukan. Hasil analisa pengujian yang terkait adalah pengujian validasi, pengujian penerimaan pengguna dan Pengujian *Compability*.

7.4.1 Analisis Hasil Pengujian Validasi

Pengujian berhasil dilakukan dengan 8 kasus uji yakni dengan kode uji PV-01 hingga PV-9. Semua pengujian yang dilakukan didapatkan status uji *valid*. Berdasarkan hasil yang di dapatkan maka dapat disimpulkan bahwa perangkat lunak sistem informasi keuangan dapat berjalan dengan baik sesuai dengan yang di harapkan pengguna.

7.4.2 Analisis Hasil Pengujian Penerimaan Pengguna

Berdasarkan data hasil pengujian penerimaan pengguna pada tabel 7.2 maka hasil analisis pengujian penerimaan pengguna akan disesuaikan dengan kriteria dan menghitung persentase jawaban pengguna.

Tabel 7.4 Hasil Analisis Pengujian Penerimaan Pengguna

No	Kriteria	Jumlah Pertanyaan Diajukan	Jawaban "SETUJU"	Jawaban "TIDAK SETUJU"	Presentase Jawaban "SETUJU"
1	<i>Functional Correctness and Completeness</i>	15	15	0	100%
2	<i>Usability</i>	6	5	1	83%
3	<i>Confidentiality and Availability</i>	5	5	0	100%

Berdasarkan hasil wawancara UAT pada Tabel 7.2 menunjukkan hasil rata-rata dari setiap kriteria menjadi 94,3% dari keseluruhan pertanyaan yang diberikan memberikan jawaban "SETUJU". Maka dapat diambil kesimpulan bahwa sistem informasi keuangan dapat diterima dengan baik dan sudah sesuai dengan kebutuhan yang disepakati diawal.

7.4.3 Analisis Hasil Pengujian *Compatibilty*

Berdasarkan hasil pengujian *compability* pada Gambar7.1 menghasilkan 2 *critical error* pada *firefox 60* dan *chrome 66* di karenakan tidak di implementasikannya fitur *ssl/https*. 2 major issues dan 7 minor issues karena tidak support pada css yang di implementasikan.

BAB 8 PENUTUP

8.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat diambil kesimpulan sebagai berikut:

- 1) A. Hasil analisis proses bisnis menghasilkan 3 proses bisnis rekomendasi (*to-be*) yang merupakan usulan dari proses bisnis saat ini (*as-is*) berdasarkan analisis permasalahan. Proses bisnis *to-be* memberikan selisih waktu 5 jam 8 menit 59 detik lebih cepat dari proses bisnis *as-is*.

B. Hasil analisis kebutuhan sistem informasi keuangan menghasilkan 23 kebutuhan fungsional, dan 1 kebutuhan non-fungsional memberikan spesifikasi kemampuan yang dapat dilakukan oleh sistem informasi keuangan.
- 2) Hasil perancangan yang dilakukan berdasarkan hasil analisis kebutuhan menghasilkan 1 *use case diagram* dimana terdapat 23 *use case*, 5 aktor dan dijelaskan pada 23 *use case scenario*, 5 *activity diagram*, 5 *sequence diagram*, 6 *class controller* dan 5 *class model* pada rancangan *class diagram*, *class diagram domain model* menghasilkan 5 kelas yang dipetakan menjadi 5 tabel *Physical Data Model diagram* dan dinormalisasi menjadi 7 tabel. Hasil rancangan pengujian validasi menghasilkan 9 *test case*, dan rancangan pengujian penerimaan pengguna menghasilkan 26 pertanyaan terhadap 3 kriteria penerimaan pengguna dan 1 *test case* pengujian *compability*.
- 3) Hasil implementasi sistem Informasi keuangan merupakan sistem informasi berbasis *website* yang merupakan hasil pengembangan berdasarkan perancangan. SIK dikembangkan dengan menggunakan kerangka kerja CodeIgniter dengan bahasa pemrograman PHP, HTML dan SQL. SIK memberikan fitur autentikasi pengguna, mengelola pengguna, mengelola pengajuan dana, verifikasi pengajuan dan pencairan dana, mengelola pencatatan dana, mengelola kategori keuangan, menyediakan laporan keuangan, mengelola pengiriman laporan keuangan dan pengiriman laporan bulanan otomatis.
- 4) Hasil pengujian sistem informasi keuangan menggunakan metode *black-box* validasi pada beberapa fungsi yang berkaitan dengan pengelolaan pengajuan dana, pengelolaan pencatatan dana, laporan keuangan dan pengiriman laporan keuangan dinyatakan valid. Hasil pengujian penerimaan pengguna menunjukkan bahwa sistem dapat diterima 94,3% dari sisi *stackholder* sebagai kepala sekolah, bendahara dan Peminta Dana dengan total 25 jawaban "SETUJU" pada kuisisioner yang dibagikan. Hasil pengujian *compability* sistem menghasilkan 2 critical error pada firefox 60 dan chrome 66 dikarenakan tidak diimplementasikannya fitur *ssl/https*. 2 major issues dan 7 minor issues karena tidak support pada *css* yang diimplementasikan.

8.2 Saran

Saran yang dapat diberikan sebagai bahan pertimbangan untuk melaksanakan pengembangan lanjut pada Sistem Informasi Keuangan KB & TK Permata Iman di antaranya adalah:

1. Perlu dilakukan penerapan lapangan Sistem Informasi Keuangan pada lokasi studi kasus KB & TK Permata Iman, kemudian dilakukan kegiatan monitoring dan evaluasi penerapan lapangan sehingga memungkinkan hasil pengembangan lanjut Sistem Informasi Keuangan yang lebih baik.
2. Perlu dilakukan evaluasi pada antarmuka pengguna sistem informasi keuangan untuk meningkatkan pengalaman dan kenyamanan pengguna dalam menggunakan sistem.
3. Ditambahnya fitur notifikasi setelah di lakukannya verifikasi pengajuan dana terhadap kepala sekolah agar dapat mempercepat proses verifikasi jika kepala sekolah berada di luar sekolah.



DAFTAR PUSTAKA

- A.S, R. and Shalahuddin, M. 2016. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika
- Becker, J., Kugeler, M. and Rosemann, M. 2003. *Preparation of Process Modeling A Guide for the Design of Business Process, Process Management: A Guide for the Design of Business Processes*. Berlin: Springer
- Bittner, K. & Spence, I., 2002. *Use Case Modeling. first edition* Boston: Addison-Wesley.
- Booch, G., 1998. *Unified Modeling Language User Guide*. Boston: Addison-Wesley.
- Booch, G. James, R. Ivar, J, 2005. *The Unified Modeling Language User Guide Second Edition*. Boston: Addison Wesley Professional.
- Ellislab, 2006. CodeIgniter User Guide Version 2.2.6. [Online] Available at: <https://www.codeigniter.com/userguide2/> [Accessed 17 March 2017].
- Fowler, M. (2005) *UML DISTILLED* Edisi 3. Panduan Singkat Bahasa Pemodelan Objek Standar. Yogyakarta: Andi
- Hansen, T. B. & Lengstorf, J., 2014. *PHP for Absolute Beginners*. New York: Apress.
- Monk, E. & Wagner, B., 2013. *Concepts in Enterprise Resource Planning*. 4th penyunt. Boston: Course Technology.
- Naik, K. & Tripathy, P., 2008. *Software Testing And Quality Assurance. In: Theory and Practice*. Hoboken: John Wiley & Sons
- O'Brien, J. a. (2012) *Management Information System*. New York: McGraw-Hill.
- Object Management Group, 2005. *OMG Unified Modeling Language Specification*. [Online] Available at: <http://www.omg.org/spec/UML/2.0/> [Diakses 31 August 2017].
- Pitt, C., 2012. *Pro PHP MVC*. New York: Apress.
- Pressman, R. S., 2010. *Software Engineering A Practitioner's Approach*. 7th ed. New York: McGraw-Hill.
- Rosing, M. von, Scheel, H. and Scheer, A.-W. (2012) *Business Process Model and Notation, The Complete Business Process Handbook*. Burlington : Morgan Kaufmann.
- Sommerville, I., 2011. *Software Engineering. 11th ed*. London : Pearson Education.
- Sugiono. 2006. *Statistika Untuk Penelitian*. Bandung : Alfabeta
- Windana, Valeria M. 2014. *Pengembangan Sistem Informasi Akuntansi Penjualan dan Penerimaan Kas Berbasis Komputer Pada Perusahaan Kecil (Studi Kasus*

pada PT. *Trust Technology*). Palembang: Jurnal Ekonomi dan Bisnis Sriwijaya.
Vol 12 No. 1.

