

**PEMBANGUNAN SISTEM PENGELOLAAN PENDISTRIBUSIAN
DANA BANTUAN SEKOLAH (STUDI KASUS: DINAS
PENDIDIKAN DAN KEBUDAYAAN KOTA SERANG)**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Birald Muhammad Dayan

NIM: 145150201111093



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018

PENGESAHAN

PEMBANGUNAN SISTEM PENGELOLAAN PENDISTRIBUSIAN DANA BANTUAN SEKOLAH (STUDI KASUS: DINAS PENDIDIKAN DAN KEBUDAYAAN KOTA SERANG)

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan memperoleh gelar Sarjana Komputer

Disusun oleh :
Biralda Muhammad Dayan
NIM: 145150201111093

Skripsi ini telah diuji dan dinyatakan lulus pada
26 Desember 2018
Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Bayu Priyambadha, S.Kom, M.Kom
NIP. 19820909 200812 1 004

Denny Sagita Rusdianto, S.Kom, M.Kom
NIP. 19851124 201504 1 001

Mengetahui
Ketua Jurusan Teknik Informatika



Tri Astoto Kurniawan, S.T, M.T, Ph.D
NIP. 19710518 200312 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 oranglain, 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, 26 Desember 2018



Birald Muhammad Dayan

NIM: 145150201111093

KATA PENGANTAR

Rasa syukur penulis panjatkan kepada Allah SWT karena atas segala limpahan kasih dan sayang-Nya penulis dapat menyelesaikan skripsi yang berjudul “PEMBANGUNAN SISTEM PENGELOLAAN PENDISTRIBUSIAN DANA BANTUAN SEKOLAH (STUDI KASUS: DINAS PENDIDIKAN DAN KEBUDAYAAN KOTA SERANG)”. Pada pengantar ini, penulis ingin berterimakasih kepada pihak-pihak yang telah membantu penulis dalam penyusunan skripsi ini, diantaranya kepada:

1. Kedua orang tua tercinta beserta saudara penulis yang telah memberikan semangat, kasih sayang, dan dukungan yang selalu diberikan dengan tulus kepada penulis.
2. Bapak Bayu Priyambadha, S.Kom, M.Kom selaku Dosen Pembimbing I skripsi dan Bapak Denny Sagita Rusdianto, S.Kom, M.Kom selaku Dosen Pembimbing II skripsi, yang telah mengarahkan, membina, dan membimbing saya sehingga dapat menyelesaikan skripsi ini.
3. Para Dosen Fakultas Ilmu Komputer Universitas Brawijaya yang telah bersedia membagi ilmu kepada penulis beserta Staff Akademik yang telah memberikan bantuan akademik selama proses menempuh studi hingga penyelesaian skripsi.
4. Bapak Drs. Akhmad Zubaidilah, M.Si selaku Kepala Dinas Pendidikan dan Kebudayaan Kota Serang dan Bapak Yayan Kosasih, S.Pd, M.Si selaku Kepala Seksi Saran Bidang Sekolah Dasar Dinas Pendidikan dan Kebudayaan Kota Serang, yang sudah mengarahkan dan membantu penelitian skripsi ini.
5. Rekan – rekan Eksekutif Mahasiswa Informatika (EMIF) Fakultas Ilmu Komputer 2017/2018 yang telah mendukung dan memberikan ilmu – ilmu organisasi kepada penulis.
6. Teman – teman dari TPOC yang selalu memberi bantuan, motivasi, dan kebersamaan selama ini.
7. Teman – teman dari MLI yang selalu menghibur dan membantu penulis ketika terjadi permasalahan.
8. Teman – teman teknik informatika 2014 atas bantuan, motivasi, dan kebersamaan selama ini sehingga penulis dapat menyelesaikan skripsi dengan baik.
9. Sahabat perjuangan saya dalam penyelesaian skripsi, Abul A’la Alghifari, Fegreit Rizda Wibowo, Kevin A. R., Muhammad Setyawan, Mikhanael S., Bestralaga R., dan yang teristimewa Atikah Nur Rahimah membantu, memberikan motivasi dan selalu menemani di saat susah maupun senang.

10. Semua pihak yang tidak semuanya bisa dituliskan disini yang terlibat secara langsung maupun tidak langsung dalam proses pengerjaan skripsi maupun sebagai pemberi semangat dan motivasi.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih banyak kekurangan, sehingga saran dan kritik yang membangun sangat penulis harapkan. Akhir kata penulis berharap skripsi ini dapat membawa manfaat bagi semua pihak yang menggunakannya.

Malang, 26 Desember 2018

Penulis

biraldmuhammad@gmail.com



ABSTRAK

Kualitas pendidikan adalah hal yang harus diperhatikan dalam dunia pendidikan, untuk meningkatkan kualitas pendidikan, seluruh satuan pendidikan wajib memiliki sarana dan prasarana sesuai standar. Pada sekolah – sekolah di kota Serang terdapat permasalahan tentang sarana dan prasarana yaitu dalam pengelolaan dana bantuan sarana dan prasarana sekolah. Masalah yang pertama adalah beberapa sekolah merasa kesulitan dalam mengajukan dana bantuan operasional sekolah dikarenakan prosesnya panjang dan rumit. Masalah yang kedua adalah beberapa sekolah juga kesulitan untuk mengajukan proposal pengajuan dana bantuan sarana dan prasarana sekolah dikarenakan jarak sekolah terlalu jauh dari kantor Dinas Pendidikan dan Kebudayaan Kota Serang. Masalah yang ketiga adalah kurang tepatnya alur dana distribusi, masih banyak sekolah yang harus segera dibantu tapi tidak tertutupi semua dikarenakan pengajuan dana bantuan sekolah diberikan kepada sekolah yang mengajukan dana bantuan sarana dan prasarana sekolah terlebih dahulu, bukan sekolah yang keadaannya butuh dibantu terlebih dahulu. Untuk menyelesaikan permasalahan tersebut dilakukan penelitian yang dimaksudkan untuk membangun sistem yang mampu untuk mempermudah dalam pengajuan bantuan sarana dan prasarana sekolah dan dapat memprioritaskan sekolah yang harus dibantu terlebih dahulu. Sistem ini diharapkan mampu mengatasi masalah – masalah yang ada dan membantu meningkatkan efisiensi kerja Dinas Pendidikan dan Kebudayaan Kota Serang. Sistem ini telah melewati pengujian unit dengan menggunakan whitebox testing, pengujian validasi dengan menggunakan blackbox testing yang menghasilkan nilai 100% valid, dan pengujian kesesuaian (*compatibility*) dimana hasilnya sistem dapat berjalan pada 8 browser berbeda.

Kata kunci: *sistem aplikasi, dana, bantuan sekolah.*

ABSTRACT

Education quality is something that must be considered in the world of education, to improve the quality of education, all education units must have facilities and infrastructure in according to the standards. In schools in Serang city, there are problems regarding facilities and infrastructure, namely in the management of funds for school facilities and infrastructure. The first problem is that some schools find it difficult to submit school operational aid funds because the process is long and complicated. The second problem is that some schools also find it difficult to submit proposals for submitting funds for school facilities and infrastructure due to the distance of the school too far from the Serang City Department of Education. The third problem is the inaccurate flow of distribution funds, there are still many schools that must be immediately assisted but not covered all because the submission of school aid funds is given to schools that submit school facilities and infrastructure grants first, not schools whose conditions need help first. To solve these problems, research is carried out which is intended to build a system that is able to facilitate the submission of school facilities and infrastructure assistance and can prioritize schools that must be assisted in advance. This system is expected to be able to overcome existing problems and help improve the work efficiency of the Serang City Department of Education. This system has passed unit testing using whitebox testing, validation testing using blackbox testing which results in a 100% valid value, and compatibility testing where the results of the system can run on 8 different browsers.

Keywords: *application system, fund, school aid.*

DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT	vii
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR.....	xv
DAFTAR LAMPIRAN	xvi
BAB 1 PENDAHULUAN.....	Error! Bookmark not defined.
1.1 Latar Belakang.....	Error! Bookmark not defined.
1.2 Rumusan masalah	Error! Bookmark not defined.
1.3 Tujuan	Error! Bookmark not defined.
1.4 Manfaat.....	Error! Bookmark not defined.
1.5 Batasan masalah	Error! Bookmark not defined.
1.6 Sistematika pembahasan	Error! Bookmark not defined.
BAB 2 LANDASAN KEPUSTAKAAN	Error! Bookmark not defined.
2.1 Kajian Pustaka	Error! Bookmark not defined.
2.2 Pendidikan	Error! Bookmark not defined.
2.3 Sarana dan Prasarana Pendidikan	Error! Bookmark not defined.
2.4 Hypertext PreProcessor (PHP)	Error! Bookmark not defined.
2.5 Laravel.....	Error! Bookmark not defined.
2.6 Rekayasa Perangkat Lunak	Error! Bookmark not defined.
2.7 Kebutuhan Fungsional dan Non-Fungsional.....	Error! Bookmark not defined.
2.8 Pendekatan Berorientasi Objek	Error! Bookmark not defined.
2.9 Permodelan Berorientasi Objek.....	Error! Bookmark not defined.
2.10 UML (<i>Unified Modeling Language</i>).....	Error! Bookmark not defined.
2.11 <i>Waterfall Model</i>	Error! Bookmark not defined.
2.12 Pengujian Sistem.....	Error! Bookmark not defined.

2.12.1 Unit Testing	Error! Bookmark not defined.
2.12.2 Integration Testing	Error! Bookmark not defined.
2.12.3 Validation Testing	Error! Bookmark not defined.
2.12.4 Compatibility Testing	Error! Bookmark not defined.
BAB 3 METODOLOGI	Error! Bookmark not defined.
3.1 Studi Literatur	Error! Bookmark not defined.
3.2 Analisis Kebutuhan Sistem.....	Error! Bookmark not defined.
3.3 Perancangan Sistem.....	Error! Bookmark not defined.
3.4 Implementasi Sistem	Error! Bookmark not defined.
3.5 Pengujian dan Analisis	Error! Bookmark not defined.
3.6 Kesimpulan dan Saran	Error! Bookmark not defined.
BAB 4 ANALISIS KEBUTUHAN	Error! Bookmark not defined.
4.1 Elisitasi Kebutuhan.....	Error! Bookmark not defined.
4.2 Gambaran Umum Aplikasi	Error! Bookmark not defined.
4.3 Identifikasi Aktor.....	Error! Bookmark not defined.
4.4 Kebutuhan Fungsional Sistem	Error! Bookmark not defined.
4.5 Kebutuhan Non Fungsional Sistem.....	Error! Bookmark not defined.
4.6 Validasi dan Verifikasi	Error! Bookmark not defined.
4.7 Use Case Diagram	Error! Bookmark not defined.
4.8 Use Case Scenario	Error! Bookmark not defined.
BAB 5 Perancangan dan implementasi.....	Error! Bookmark not defined.
5.1 Perancangan Sistem.....	Error! Bookmark not defined.
5.1.1 Perancangan Arsitektur.....	Error! Bookmark not defined.
5.1.2 Sequence Diagram.....	Error! Bookmark not defined.
5.1.3 Class Diagram	Error! Bookmark not defined.
5.1.4 Perancangan Komponen	Error! Bookmark not defined.
5.1.5 Perancangan Data	Error! Bookmark not defined.
5.1.6 Perancangan Antarmuka.....	Error! Bookmark not defined.
5.2 Implementasi Sistem	Error! Bookmark not defined.
5.2.1 Spesifikasi Sistem	Error! Bookmark not defined.
5.2.2 Implementasi Kode Program	Error! Bookmark not defined.
5.2.3 Implementasi Data	Error! Bookmark not defined.



5.2.4 Implementasi Antarmuka**Error! Bookmark not defined.**

BAB 6 Pengujian**Error! Bookmark not defined.**

6.1 Pengujian Unit.....**Error! Bookmark not defined.**

6.1.1 Pengujian Unit *Method* “input_proposal_sekolah” class “sekolah_controller”**Error! Bookmark not defined.**

6.1.2 Pengujian Unit *Method* “pengiriman_dana” class “kasi_controller”**Error! Bookmark not defined.**

6.1.3 Pengujian Unit *Method* “input_update_dana” class “sekolah_controller”**Error! Bookmark not defined.**

6.2 Pengujian Integrasi**Error! Bookmark not defined.**

6.3 Pengujian Validasi**Error! Bookmark not defined.**

6.4 Pengujian *Compatibility***Error! Bookmark not defined.**

BAB 7 PENUTUP**Error! Bookmark not defined.**

7.1 Kesimpulan.....**Error! Bookmark not defined.**

7.2 Saran**Error! Bookmark not defined.**

DAFTAR PUSTAKA.....**Error! Bookmark not defined.**

LAMPIRAN A SURAT PERMOHONAN DATA SKRIPSI..**Error! Bookmark not defined.**

LAMPIRAN B KETERANGAN PERSETUJUAN PENELITIAN**Error! Bookmark not defined.**

LAMPIRAN C HASIL WAWANCARA.....**Error! Bookmark not defined.**

.....**Error! Bookmark not defined.**



DAFTAR TABEL

Tabel 2.1 Kajian Pustaka	Error! Bookmark not defined.
Tabel 4.1 Identifikasi Aktor	Error! Bookmark not defined.
Tabel 4.2 Kebutuhan Fungsional Sistem.....	Error! Bookmark not defined.
Tabel 4.3 Kebutuhan Non Fungsional Sistem.....	Error! Bookmark not defined.
Tabel 4.4 <i>Use Case Scenario Login</i>	Error! Bookmark not defined.
Tabel 4.5 <i>Use Case Scenario Logout</i>	Error! Bookmark not defined.
Tabel 4.6 <i>Use Case Scenario Buat Proposal</i>	Error! Bookmark not defined.
Tabel 4.7 <i>Use Case Scenario Ubah Proposal</i>	Error! Bookmark not defined.
Tabel 4.8 <i>Use Case Scenario Hapus Proposal</i>	Error! Bookmark not defined.
Tabel 4.9 <i>Use Case Scenario Lihat Daftar Proposal</i> .	Error! Bookmark not defined.
Tabel 4.10 <i>Use Case Scenario Lihat Detail Proposal</i>	Error! Bookmark not defined.
Tabel 4.11 <i>Use Case Scenario Approve Proposal</i>	Error! Bookmark not defined.
Tabel 4.12 <i>Use Case Scenario Decline Proposal</i>	Error! Bookmark not defined.
Tabel 4.13 <i>Use Case Scenario Kirim Tugas Verifikasi</i>	Error! Bookmark not defined.
Tabel 4.14 <i>Use Case Scenario Verifikasi Proposal</i> ...	Error! Bookmark not defined.
Tabel 4.15 <i>Use Case Scenario Notifikasi</i>	Error! Bookmark not defined.
Tabel 4.16 <i>Use Case Scenario Lihat Daftar Prioritas</i>	Error! Bookmark not defined.
Tabel 4.17 <i>Use Case Scenario Kirim Dana</i>	Error! Bookmark not defined.
Tabel 4.18 <i>Use Case Scenario Input Dana Terpakai</i>	Error! Bookmark not defined.
Tabel 4.19 <i>Use Case Scenario Log Aktivitas Proposal</i>	Error! Bookmark not defined.
Tabel 4.20 <i>Use Case Scenario Daftar Sekolah</i>	Error! Bookmark not defined.
Tabel 4.21 <i>Use Case Scenario Daftar Kepala Seksi</i> ..	Error! Bookmark not defined.
Tabel 4.22 <i>Use Case Scenario Daftar Tim Verifikasi</i>	Error! Bookmark not defined.
Tabel 4.23 <i>Use Case Scenario Tambah Sekolah</i>	Error! Bookmark not defined.
Tabel 4.24 <i>Use Case Scenario Ubah Sekolah</i>	Error! Bookmark not defined.
Tabel 4.25 <i>Use Case Scenario Hapus Sekolah</i>	Error! Bookmark not defined.
Tabel 4.26 <i>Use Case Scenario Tambah Kepala Seksi</i>	Error! Bookmark not defined.
Tabel 4.27 <i>Use Case Scenario Ubah Kepala Seksi</i>	Error! Bookmark not defined.

Tabel 4.28 Use Case Scenario Hapus Kepala Seksi ..Error! Bookmark not defined.

Tabel 4.29 Use Case Scenario Tambah Tim VerifikasiError! Bookmark not defined.

Tabel 4.30 Use Case Scenario Ubah Tim Verifikasi..Error! Bookmark not defined.

Tabel 4.31 Use Case Scenario Hapus Tim Verifikasi Error! Bookmark not defined.

Tabel 4.32 Use Case Scenario Lihat Rekap BantuanError! Bookmark not defined.

Tabel 5.1 Penjelasan antarmuka daftar proposalError! Bookmark not defined.

Tabel 5.2 Penjelasan antarmuka detail proposal.....Error! Bookmark not defined.

Tabel 5.3 Penjelasan antarmuka rekap proposal.....Error! Bookmark not defined.

Tabel 5.4 Spesifikasi Perangkat KerasError! Bookmark not defined.

Tabel 6.1 Hasil pengujian unit method “input_proposal_sekolah” class “sekolah_controller”Error! Bookmark not defined.

Tabel 6.2 Hasil pengujian unit method “pengiriman_dana” class “kasi_controller”Error! Bookmark not defined.

Tabel 6.3 Hasil pengujian unit method “input_update_dana ” class “sekolah_controller”Error! Bookmark not defined.

Tabel 6.4 Identifikasi dan Perancangan Pengujian IntegrasiError! Bookmark not defined.

Tabel 6.5 Hasil Pengujian Integrasi Nomor 1Error! Bookmark not defined.

Tabel 6.6 Hasil Pengujian Integrasi Nomor 2Error! Bookmark not defined.

Tabel 6.7 Hasil Pengujian Integrasi Nomor 3Error! Bookmark not defined.

Tabel 6.8 Hasil Pengujian Validasi LoginError! Bookmark not defined.

Tabel 6.9 Hasil Pengujian Validasi Login Alternatif 1 Error! Bookmark not defined.

Tabel 6.10 Hasil Pengujian Validasi Login Alternatif 2Error! Bookmark not defined.

Tabel 6.11 Hasil Pengujian Validasi Logout.....Error! Bookmark not defined.

Tabel 6.12 Hasil Pengujian Validasi Buat Proposal ...Error! Bookmark not defined.

Tabel 6.13 Hasil Pengujian Validasi Buat Proposal Alternatif 1Error! Bookmark not defined.

Tabel 6.14 Hasil Pengujian Validasi Buat Proposal Alternatif 2Error! Bookmark not defined.

Tabel 6.15 Hasil Pengujian Validasi Buat Proposal Alternatif 3Error! Bookmark not defined.

Tabel 6.16 Hasil Pengujian Validasi Ubah Proposal ..Error! Bookmark not defined.

- Tabel 6.17 Hasil Pengujian Validasi Ubah Proposal Alternatif 1 **Error! Bookmark not defined.**
- Tabel 6.18 Hasil Pengujian Validasi Ubah Proposal Alternatif 2 **Error! Bookmark not defined.**
- Tabel 6.19 Hasil Pengujian Validasi Ubah Proposal Alternatif 3 **Error! Bookmark not defined.**
- Tabel 6.20 Hasil Pengujian Validasi Hapus Proposal. **Error! Bookmark not defined.**
- Tabel 6.21 Hasil Pengujian Validasi Lihat Daftar Proposal **Error! Bookmark not defined.**
- Tabel 6.22 Hasil Pengujian Validasi Lihat Detail Proposal **Error! Bookmark not defined.**
- Tabel 6.23 Hasil Pengujian Validasi Approve Proposal **Error! Bookmark not defined.**
- Tabel 6.24 Hasil Pengujian Validasi Decline Proposal **Error! Bookmark not defined.**
- Tabel 6.25 Hasil Pengujian Validasi Kirim Tugas Verifikasi **Error! Bookmark not defined.**
- Tabel 6.26 Hasil Pengujian Validasi Kirim Tugas Verifikasi Alternatif 1 **Error! Bookmark not defined.**
- Tabel 6.27 Hasil Pengujian Validasi Kirim Tugas Verifikasi Alternatif 2 **Error! Bookmark not defined.**
- Tabel 6.28 Hasil Pengujian Validasi Kirim Tugas Verifikasi Alternatif 3 **Error! Bookmark not defined.**
- Tabel 6.29 Hasil Pengujian Validasi Verifikasi Proposal **Error! Bookmark not defined.**
- Tabel 6.30 Hasil Pengujian Validasi Verifikasi Proposal Alternatif 1 **Error! Bookmark not defined.**
- Tabel 6.31 Hasil Pengujian Validasi Verifikasi Proposal Alternatif 2 **Error! Bookmark not defined.**
- Tabel 6.32 Hasil Pengujian Validasi Notifikasi **Error! Bookmark not defined.**
- Tabel 6.33 Hasil Pengujian Validasi Lihat Daftar Prioritas **Error! Bookmark not defined.**
- Tabel 6.34 Hasil Pengujian Validasi Kirim Dana **Error! Bookmark not defined.**
- Tabel 6.35 Hasil Pengujian Validasi Kirim Dana Alternatif 1 **Error! Bookmark not defined.**
- Tabel 6.36 Hasil Pengujian Validasi Kirim Dana Alternatif 2 **Error! Bookmark not defined.**

Tabel 6.37 Hasil Pengujian Validasi Kirim Dana Alternatif 3 **Error! Bookmark not defined.**

Tabel 6.38 Hasil Pengujian Validasi Kirim Dana Alternatif 4 **Error! Bookmark not defined.**

Tabel 6.39 Hasil Pengujian Validasi Input Dana Terpakai **Error! Bookmark not defined.**

Tabel 6.40 Hasil Pengujian Validasi Input Dana Terpakai Alternatif 1..... **Error! Bookmark not defined.**

Tabel 6.41 Hasil Pengujian Validasi Input Dana Terpakai Alternatif 2..... **Error! Bookmark not defined.**

Tabel 6.42 Hasil Pengujian Validasi Input Dana Terpakai Alternatif 3..... **Error! Bookmark not defined.**

Tabel 6.43 Hasil Pengujian Validasi Input Dana Terpakai Alternatif 4..... **Error! Bookmark not defined.**

Tabel 6.44 Hasil Pengujian Validasi Log Aktivitas Proposal **Error! Bookmark not defined.**

Tabel 6.45 Hasil Pengujian Daftar Sekolah **Error! Bookmark not defined.**

Tabel 6.46 Hasil Pengujian Validasi Daftar Kepala Seksi **Error! Bookmark not defined.**

Tabel 6.47 Hasil Pengujian Validasi Daftar Tim Verifikasi **Error! Bookmark not defined.**

Tabel 6.48 Hasil Pengujian Validasi Tambah Sekolah **Error! Bookmark not defined.**

Tabel 6.49 Hasil Pengujian Validasi Tambah Sekolah Alternatif 1 **Error! Bookmark not defined.**

Tabel 6.50 Hasil Pengujian Validasi Ubah Sekolah.... **Error! Bookmark not defined.**

Tabel 6.51 Hasil Pengujian Validasi Ubah Sekolah Alternatif 1 **Error! Bookmark not defined.**

Tabel 6.52 Hasil Pengujian Validasi Hapus Sekolah. **Error! Bookmark not defined.**

Tabel 6.53 Hasil Pengujian Validasi Tambah Kepala Seksi **Error! Bookmark not defined.**

Tabel 6.54 Hasil Pengujian Validasi Tambah Kepala Seksi Alternatif 1..... **Error! Bookmark not defined.**

Tabel 6.55 Hasil Pengujian Validasi Ubah Kepala Seksi **Error! Bookmark not defined.**

Tabel 6.56 Hasil Pengujian Validasi Ubah Kepala Seksi Alternatif 1 **Error! Bookmark not defined.**

Tabel 6.57 Hasil Pengujian Validasi Hapus Kepala Seksi **Error! Bookmark not defined.**

Tabel 6.58 Hasil Pengujian Validasi Tambah Tim Verifikasi **Error! Bookmark not defined.**

Tabel 6.59 Hasil Pengujian Validasi Tambah Tim Verifikasi Alternatif 1..... **Error! Bookmark not defined.**

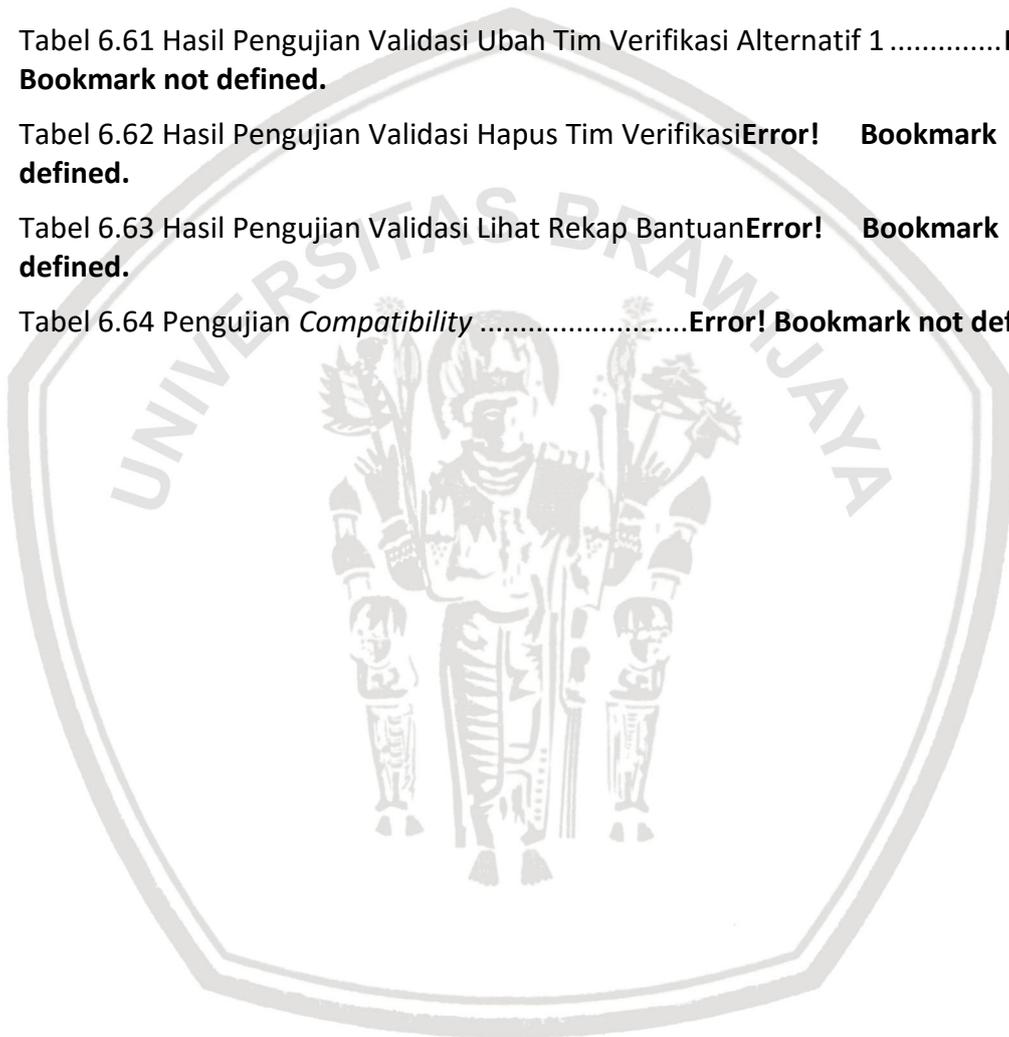
Tabel 6.60 Hasil Pengujian Validasi Ubah Tim Verifikasi **Error! Bookmark not defined.**

Tabel 6.61 Hasil Pengujian Validasi Ubah Tim Verifikasi Alternatif 1 **Error! Bookmark not defined.**

Tabel 6.62 Hasil Pengujian Validasi Hapus Tim Verifikasi **Error! Bookmark not defined.**

Tabel 6.63 Hasil Pengujian Validasi Lihat Rekap Bantuan **Error! Bookmark not defined.**

Tabel 6.64 Pengujian *Compatibility* **Error! Bookmark not defined.**



DAFTAR GAMBAR

Gambar 2.1 <i>Use Case Diagram</i>	Error! Bookmark not defined.
Gambar 2.2 <i>Sequence Diagram</i>	Error! Bookmark not defined.
Gambar 2.3 <i>Class Diagram</i>	Error! Bookmark not defined.
Gambar 3.1 Metodologi Penelitian.....	Error! Bookmark not defined.
Gambar 4.1 <i>Use Case Diagram</i> Sistem.....	Error! Bookmark not defined.
Gambar 5.1 <i>Sequence Diagram</i> Buat Proposal	Error! Bookmark not defined.
Gambar 5.2 <i>Sequence Diagram</i> Kirim Dana	Error! Bookmark not defined.
Gambar 5.3 <i>Sequence Diagram</i> Input Dana Terpakai Error!	Bookmark not defined.
Gambar 5.4 <i>Class Diagram</i>	Error! Bookmark not defined.
Gambar 5.5 <i>Conceptual Data Model</i>	Error! Bookmark not defined.
Gambar 5.6 Perancangan antarmuka Daftar Proposal Error!	Bookmark not defined.
Gambar 5.7 Perancangan antarmuka Detail Proposal Error!	Bookmark not defined.
Gambar 5.8 Perancangan antarmuka rekap proposal Error!	Bookmark not defined.
Gambar 5.9 Implementasi Database	Error! Bookmark not defined.
Gambar 5.10 Implementasi antarmuka daftar proposal Error!	Bookmark not defined.
Gambar 5.11 Implementasi antarmuka Detail Proposal Error!	Bookmark not defined.
Gambar 5.12 Implementasi antarmuka Rekap Data Error!	Bookmark not defined.
Gambar 6.1 Flow graph method “input_proposal_sekolah” class “sekolah_controller”	Error! Bookmark not defined.
Gambar 6.2 Flow graph method “pengiriman_dana” class “kasi_controller”	Error! Bookmark not defined.
Gambar 6.3 Flow graph method “input_update_dana” class “sekolah_controller”	Error! Bookmark not defined.
Gambar 6.4 Hasil Pengujian <i>Compatibility</i>	Error! Bookmark not defined.

DAFTAR LAMPIRAN

LAMPIRAN A SURAT PERMOHONAN DATA SKRIPSI..Error! Bookmark not defined.

LAMPIRAN B KETERANGAN PERSETUJUAN PENELITIANError! Bookmark not defined.

LAMPIRAN C HASIL WAWANCARA.....Error! Bookmark not defined.





BAB 1 PENDAHULUAN

1.1 Latar Belakang

Setiap satuan pendidikan wajib memiliki sarana dan prasarana sesuai standar meliputi perabot, peralatan pendidikan, buku, sumber belajar lainnya, bahan habis pakai, serta perlengkapan lain yang diperlukan untuk menunjang proses pembelajaran yang teratur dan berkelanjutan (Permendiknas no 24 tahun 2007). Kualitas pendidikan adalah hal yang harus diperhatikan dalam dunia pendidikan, salah satu aspek yang berpengaruh dalam meningkatkan kualitas pendidikan adalah sarana dan prasarana di sekolah karena sekolah merupakan tempat untuk mencetak manusia yang memiliki pengetahuan dan keterampilan, bertanggung jawab, demokratis dan berprilaku yang mantap serta mandiri (Soedijarto, 2008). Agar tujuan tersebut dapat tercapai maka dibutuhkan salah satunya sarana dan prasarana yang sesuai dengan standar nasional.

Tujuan pembangunan dan perbaikan ruang kelas dan yang lainnya untuk memenuhi standar sarana dan prasarana, selain itu mendorong tercapainya standar nasional pendidikan dalam hal jumlah murid per kelas, dengan adanya peningkatan sarana dan prasarana pendidikan diharapkan akan meningkatkan mutu pendidikan, baik secara kuantitas maupun kualitas (Diskominfokota Serang, 2017). Sekolah juga harus memiliki predikat SSN (Sekolah Standar Nasional) salah satunya adalah standar sarana dan prasarana (Permendiknas no 24 tahun 2007).

Saat ini, instansi pemerintahan yaitu Dinas Pendidikan dan Kebudayaan yang berada di kota Serang untuk melakukan penyaluran anggaran dana untuk sekolah – sekolah yang butuh bantuan, sekolah yang bangunannya sudah tidak layak pakai, serta sekolah – sekolah yang terkena bencana masih belum maksimal. Masalah yang pertama adalah beberapa sekolah merasa kesulitan dalam mengajukan dana bantuan operasional sekolah dikarenakan prosesnya panjang dan rumit. Masalah yang kedua adalah beberapa sekolah juga kesulitan untuk mengajukan proposal pengajuan dana bantuan sarana dan prasarana sekolah dikarenakan jarak sekolah terlalu jauh dari kantor Dinas Pendidikan dan Kebudayaan Kota Serang. Masalah yang ketiga adalah kurang tepatnya alur dana distribusi, masih banyak sekolah yang harus segera dibantu tapi tidak tertutupi semua dikarenakan pengajuan dana bantuan sekolah diberikan kepada sekolah yang mengajukan dana bantuan sarana dan prasarana sekolah terlebih dahulu, bukan sekolah yang keadaannya butuh dibantu terlebih dahulu. Masalah didapatkan berdasarkan wawancara kepada Bapak Yayan Kosasih, S.Pd, M.Si. selaku Kepala Seksi Sarana Bidan SD Dinas Pendidikan dan Kebudayaan Kota Serang yang tersedia pada Lampiran C. Pada tanggal 25 Oktober 2017 sebanyak 65 siswa SD Batarpanjang di Cikeusal, Serang, Banten terpaksa dievakuasi dan belajar ditenda karena bangunan sekolah hampir roboh (Bhaktiar Rifa'i, DetikNews, 2017). Dengan adanya peristiwa ini Dinas Pendidikan sangat menyayangkan karena tidak ada laporan sebelum peristiwa itu muncul di media.

Untuk menyelesaikan permasalahan tersebut, dibangun sistem aplikasi berbasis *website* untuk mengelola bantuan sekolah yang butuh dibantu yang dimana target dari user sendiri adalah pihak sekolah. Sistem aplikasi ini bertujuan untuk menampung proposal - proposal sekolah yang butuh dibantu kepada Dinas Pendidikan dan Kebudayaan Kota Serang dari instansi sekolah yang ada di Kota Serang tersebut dan dapat memprioritaskan urutan sekolah yang butuh bantuan terlebih dahulu serta memonitoring proses bantuan sekolah. Untuk dapat memprioritaskan urutan sekolah yang butuh bantuan terlebih dahulu terdapat 4(empat) aspek yang diperhitungkan, dari 4 aspek tersebut dihitung dan diurutkan berdasarkan hasil persentase perhitungan. Dengan adanya sistem aplikasi ini diharapkan tidak ada lagi sekolah yang tidak dibantu dan anggaran dana dari pemerintah tepat sasaran dan mengoptimalkan penyaluran anggaran dana bantuan sekolah.

1.2 Rumusan masalah

1. Bagaimana analisis dilakukan untuk mendapatkan kebutuhan?
2. Bagaimana perancangan sistem pengelolaan dana bantuan sekolah dari analisis kebutuhan dilakukan?
3. Bagaimana implementasi pada sistem pengelolaan dana bantuan sekolah dari perancangan yang dibuat?
4. Bagaimana hasil pengujian sistem pengelolaan dana bantuan sekolah?

1.3 Tujuan

Bagian ini berisi tujuan yang ingin dicapai dari skripsi ini. Tujuan yang ditulis harus dapat memberikan arah pada capaian penelitian. Tujuan ini dapat terdiri dari beberapa butir yang masing – masing harus dituliskan dalam kalimat pernyataan yang sederhana dan jelas, sesuai dengan masalah penelitian dan hasil yang ingin di capai.

1. Menganalisis dan menyusun spesifikasi kebutuhan sistem pengelolaan dana bantuan sekolah berbasis web.
2. Merancang sistem pengelolaan dana bantuan sekolah berbasis web.
3. Mengimplementasi rancangan sistem pengelolaan dana bantuan sekolah berbasis web untuk menyelesaikan masalah yang ada.
4. Menguji sistem pengelolaan dana bantuan sekolah berbasis web memenuhi kebutuhan.

1.4 Manfaat

1. Membantu Dinas Pendidikan dan Kebudayaan Kota Serang dalam pengelolaan pendistribusian dana bantuan sekolah.
2. Membantu meningkatkan ketanggapan Dinas Pendidikan dan Kebudayaan Kota Serang dalam menangani sekolah sekolah yang butuh bantuan.

1.5 Batasan masalah

Batasan pada penelitian ini adalah:

1. Sistem ini hanya digunakan oleh Dinas Pendidikan dan Kebudayaan Kota Serang.
2. Sistem aplikasi ini membutuhkan *browser* (aplikasi pendukung) untuk menjalankannya, karena sistem aplikasi ini berbasis website.
3. Sistem aplikasi ini membutuhkan koneksi internet untuk menjalankannya.

1.6 Sistematika pembahasan

Berikut ini merupakan sistematika pembahasan yang ada dalam laporan ini:

BAB I : PENDAHULUAN

Bab ini membahas mengenai latar belakang, identifikasi masalah, rumusan masalah, tujuan, manfaat, dan sistematika pembahasan.

BAB II : LANDASAN KEPUSTAKAAN

Bab ini berisi tentang teori-teori dan referensi yang mendukung dalam perancangan, implementasi dan pengujian dalam metode yang digunakan.

BAB III : METODOLOGI

Bab ini membahas mengenai metode yang akan diimplementasikan dalam penelitian yang meliputi dasar teori, metode perancangan, metode implementasi dan metode pengujian serta pengambilan kesimpulan, saran.

BAB IV : ANALISIS KEBUTUHAN

Pada bab ini akan dijelaskan mengenai analisis kebutuhan untuk membangun sistem pengelolaan dana bantuan sekolah.

BAB V : PERANCANGAN DAN IMPLEMENTASI

Bab ini memuat perancangan yang digunakan untuk pembangunan sistem pengelolaan dana bantuan sekolah dan implementasi yang akan dilakukan untuk memenuhi kebutuhan yang ada.

BAB VI : PENGUJIAN

Bab ini memuat hasil pengujian dari hasil implementasi yang sudah dibuat.

BAB VII : PENUTUP

Bab terakhir memuat kesimpulan dari hasil jawaban rumusan masalah yang didapatkan dari pembahasan serta saran untuk penelitian kedepannya.

Diharapkan bagian ini dapat membantu pembaca dalam memahami sistematika pembahasan isi dalam skripsi ini.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Pada kajian pustaka ini terdapat pembahasan tentang penelitian – penelitian yang sudah ada untuk menjadi referensi penelitian ini, kajian pustaka yang digunakan akan dijelaskan pada Tabel 2.1 yang menjelaskan tentang judul penelitian, nama penulis, dan persamaan serta perbedaan dari penelitian tersebut dengan penelitian ini.

1. Penelitian yang dilakukan oleh Udkhiati Mawaddah dengan judul Otomasi Penilaian Standar Sarana Prasarana Pada Sistem Informasi Supervisi, Monitorin dan Evaluasi Sekolah Standara Nasional Tingkat SLTP. Pada penelitian ini mengembangkan sistem infomasi monitoring untuk sarana dan prasarana sekolah.
2. Penelitian yang dilakukan oleh Awaludin dengan judul Sistem Infomasi Manajemen Sarana Prasarana Sekolah. Pada penelitian ini mengembangkan sistem informasi untuk memudahkan sekolah dalam melakukan pengadaan dan perubahan sarana prasarana sekolah.

Tabel 2.1 Kajian Pustaka

No	Nama Penulis, Tahun, dan Judul	Persamaan	Perbedaan	
			Penelitian Terdahulu	Rencana Penelitian
1	Udkhiati Mawaddah, 2013, Otomasi Penilaian Standar Sarana Prasarana Pada Sistem Informasi Supervisi, Monitorin dan Evaluasi Sekolah Standara Nasional Tingkat SLTP.	Sistem infomasi sarana dan prasarana sekolah untuk memudahkan memonitoring sarana prasarana sekolah.	Cakupan penelitian tentang monitoring sarana prasarana pada tingkat SLTP.	Cakupan penelitian adalah pengajuan bantuan sarana prasarana sekolah dan laporan sekolah pada tingkat paud, sd, dan smp.
2	Awaludin dan Eki Sputra, 2016, Sistem Infomasi Manajemen Sarana Prasarana Sekolah.	Sistem infomasi untuk meningkatkan kualitas sarana prasarana sekolah.	Cakupan penelitan tentang pelaporan pengadaan dan perubahan sarana prasarana sekolah.	Cakupan penelitian tentang pengajuan bantuan sarana prasarana sekolah dengan sistem prioritas untuk menentukan yang terlebih dahulu dibantu.

2.2 Pendidikan

Menurut Undang – Undang SISDIKNAS nomor 23 tahun 2003, pendidikan adalah usaha sadar dan terencana untuk mewujudkan suasana belajar dan proses pembelajaran agar peserta didik secara aktif mengembangkan potensi dirinya untuk memiliki kekuatan spiritual keagamaan, pengendalian diri, kepribadian, kecerdasan, akhlak mulia, serta keterampilan yang diperlukan dirinya, masyarakat, bangsa dan negara.

2.3 Sarana dan Prasarana Pendidikan

Menurut Peraturan Menteri Pendidikan Nasional no 24 tahun 2007, sarana adalah perlengkapan yang diperlukan untuk menyelenggarakan pembelajaran yang dapat dipindah-pindah, sedangkan Prasarana adalah fasilitas dasar yang diperlukan untuk menjalankan fungsi satuan pendidikan.

Setiap satuan pendidikan wajib memiliki sarana dan prasarana sesuai standar meliputi perabot, peralatan pendidikan, buku, sumber belajar lainnya, bahan habis pakai, serta perlengkapan lain yang diperlukan untuk menunjang proses pembelajaran yang teratur dan berkelanjutan.

2.4 Hypertext PreProcessor (PHP)

Menurut Arief (2011), *Hypertext PreProcessor* (PHP) merupakan bahasa pemrograman *open source* yang paling banyak digunakan saat ini dan paling banyak didukung oleh *web server*. Dengan adanya *server-side-scripting* maka sintaks dan perintah – perintah PHP dapat dieksekusi dan kemudian hasilnya dapat dikirimkan ke browser.

Keunggulan PHP ini adalah keamanan halaman website akan terjamin karena kode program yang ditulis dalam PHP tidak akan terlihat oleh pengguna atau *user* lainnya. *PHP* dapat digunakan oleh banyak sistem operasi Windows, Linux, maupun BSD.

2.5 Laravel

Laravel merupakan framework yang menggunakan bahasa pemrograman PHP dengan konsep *model view controller* (MVC). Laravel saat ini berada pada lisensi MIT. Tujuan dibangunnya Laravel ini untuk mengurangi tugas – tugas yang umum dilakukan dalam pengembangan web sehingga dapat mengurangi beban dari developer pengembangan tanpa mengurangi kualitas dari perangkat lunak tersebut (Laravel, 2018). Berikut merupakan beberapa fitur yang disediakan oleh Laravel:

1. Routing
2. Blade Template
3. Auth
4. Database Migration

Fitur – fitur yang ada dalam Laravel dapat mempermudah dalam mengembangkan APL dengan bahasa pemrograman PHP. *Artisan* merupakan *command tool* yang disediakan oleh Laravel untuk membantu dalam pembuatan *database*, membuat *file controller* serta *model*.

2.6 Rekayasa Perangkat Lunak

Menurut Sommerville (2011), rekayasa perangkat lunak adalah suatu ilmu yang mempelajari proses pengembangan perangkat lunak, dimulai dari tahap spesifikasi sistem hingga cara untuk mempertahankan sistem setelah digunakan. Pada pengertian tersebut terdapat dua poin penting yaitu:

1. *Engineering discipline*
Dengan menerapkan teori, metode, dan peralatan pendukung untuk mendapatkan solusi, seorang perrekaayasa harus membuat sistem berjalan sesuai dengan fungsinya.
2. *All aspects of software production*
Proses manajemen, pengembangan alat, metode, dan teori perlu diperhatikan untuk mendukung produksi perangkat lunak karena rekayasa perangkat luna tidak berfokus hanya pada tahapan teknis pengembangan perangkat lunak.

2.7 Kebutuhan Fungsional dan Non-Fungsional

Kebutuhan sistem merupakan deskripsi dari apa yang harus sistem lakukan. Kebutuhan tersebut merefleksikan kebutuhan konsumen dari sistem yang menyediakan tujuan khusus. Proses mencari, menganalisis, mendokumentasikan dan memeriksa ketersediaan layanan yang disebut rekayasa kebutuhan (Sommerville, 2011). mengenai kebutuhan fungsional dan non-fungsional :

- a. Kebutuhan Fungsional
Merupakan pernyataan dari layanan sistem harus disediakan, bagaimana sistem harus bereaksi terhadap input tertentu, dan bagaimana sistem harus berperilaku dalam situasi tertentu. Dalam beberapa kasus, kebutuhan fungsional dapat juga secara eksplisit menyatakan apa yang sistem tidak harus melakukan.
- b. Kebutuhan Non-Fungsional
Kebutuhan non-fungsional merupakan kendala pada layanan atau fungsi yang ditawarkan oleh sistem. Termasuk di dalamnya kendala waktu, kendala proses pembangunan, dan kendala yang pada standardisasi. Kebutuhan non-fungsional sering berlaku untuk sistem secara keseluruhan, bukan fitur sistem individual atau jasa.

2.8 Pendekatan Berorientasi Objek

Pendekatan berorientasi objek atau bisa disebut *Object Oriented (OO)* merupakan salah satu pendekatan dalam pengembangan perangkat lunak. *Object Oriented Analysis (OOA)* merupakan tahap awal proses pembangunan perangkat lunak dalam pendekatan pembangunan perangkat lunak OO dengan tujuan untuk mengidentifikasi kebutuhan – kebutuhan sistem, objek, dan kelas yang didapatkan dari domain masalah. *Object Oriented Design (OOD)* merupakan tahap selanjutnya setelah tahap OOA, pada tahap ini dilakukan perancangan arsitektur, komponen, data, dan antarmuka. Setelah proses OOD dilakukan proses *Object Oriented Programming (OOP)* yang bertujuan mengubah hasil rancangan pada OOD menjadi alur kode program yang dapat dibaca oleh mesin. Tahap terakhir adalah *Object Oriented Testing (OOT)* yang bertujuan untuk menguji hasil perancangan arsitektur, antarmuka, dan komponen (Pressman, 2010).

2.9 Permodelan Berorientasi Objek

Pada OOA dan OOD dapat dimodelkan dengan diagram yang terdapat pada Unified Modelling Language (UML). UML adalah sebuah bahasa yang telah menjadi standar dalam industry untuk visualisasi, merancang dan mendokumentasi sistem perangkat lunak (Pressman, 2010). Terdapat tiga diagram yang akan digunakan pada penelitian ini yaitu:

1. Diagram use case

Diagram use case memberikan gambaran umum mengenai interaksi antara sistem dengan lingkungannya untuk memberikan informasi yang mudah dipahami mengenai apa saja yang terlibat dengan sistem. Diagram use case merepresentasikan sebuah interaksi antara aktor dengan sistem.

2. Diagram sequence

Diagram sequence menggambarkan interaksi antar objek didalam dan diluar sistem berupa pesan yang digambarkan terhadap waktu. Diagram sequence merepresentasikan scenario atau rangkaian langkah-langkah yang dilakukan sebagai tanggapan dari sebuah pesan untuk menghasilkan keluaran tertentu.

3. Diagram class

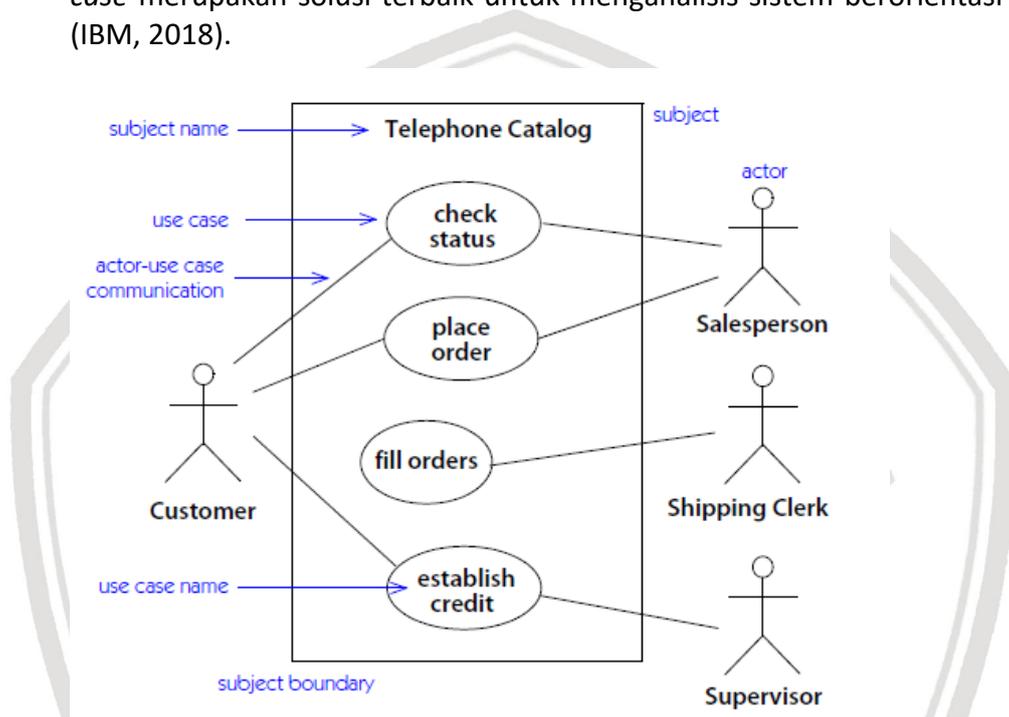
Diagram class menggambarkan struktur dan deskripsi kelas, package, dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi dan lain lain. Kelas adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Kelas menggambarkan atribut suatu sistem, sekaligus menawarkan layanan untuk memanipulasi atribut tersebut dengan method atau fungsi.

2.10 UML (*Unified Modeling Language*)

Unified Modeling Language (UML) adalah sebuah bahasa yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasi sistem perangkat lunak (Pressman, 2010). UML terdiri dari beberapa jenis diagram, yaitu:

a. Use Case Diagram

Use case berguna untuk mengetahui bagaimana sebuah sistem harus bekerja. *Use case* membantu untuk membangun skenario proses dari suatu sistem, *Use case* merupakan solusi terbaik untuk menganalisis sistem berorientasi objek (IBM, 2018).



Gambar 2.1 Use Case Diagram

(Sumber: *Unified Modeling Language Reference Manual, The (2nd Edition)*, 2010)

Use case Diagram memodelkan fungsionalitas sistem terhadap aktor yang berinteraksi dengan sistem dari sudut pandang tertentu. Salah satu tujuan dari *Use case* Diagram adalah untuk mendapatkan aktor dan *use case* apa saja yang ada pada suatu sistem, dan juga interaksi antara aktor dan *use case* yang ada (James Rumbaugh, 2004).

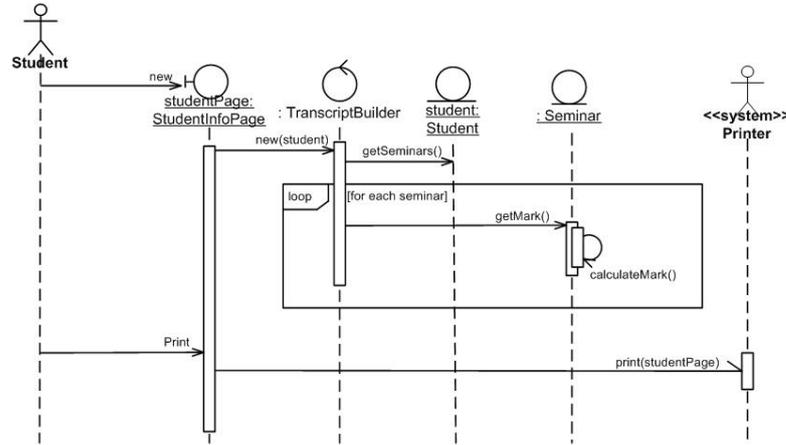
b. Use Case Scenario

Use Case Scenario merupakan deskripsi yang mencatat mengenai detail pemrosesan dari suatu *use case* (Satzinger et al., 2010). *Use Case* ini memiliki urutan yang lengkap dari tahapan – tahapan untuk menyelesaikan bisnis proses.

Scenario merupakan suatu kumpulan unik dari aktivitas internal di dalam *use case* yang menggambarkan langkah unik sepanjang *use case*.

c. Sequence Diagram

Sequence Diagram digunakan untuk memodelkan rincian dari skenario sistem. Diagram ini menjelaskan interaksi pertukaran pesan antar objek untuk menghasilkan hasil yang diinginkan (IBM, 2016).



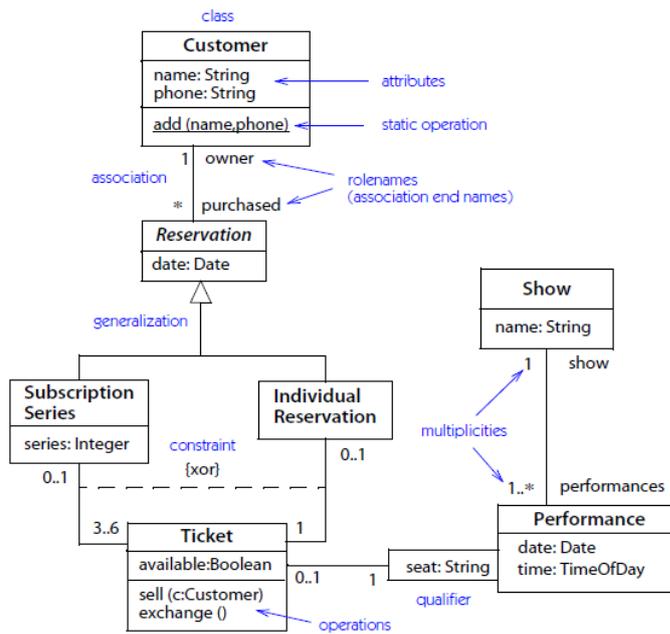
Gambar 2.2 Sequence Diagram

(Sumber: AgileModeling, 2018)

Sequence Diagram menunjukkan objek yang terlibat dalam skenario dengan garis vertikal putus-putus yang disebut *lifelines*. Garis horizontal antar objek menunjukkan pertukaran pesan antar objek, kronologi pertukaran pesan dapat dilihat dari atas diagram ke bawah.

d. Class Diagram

Class Diagram adalah Diagram statis analisis utama untuk sebuah sistem, dimana struktur kelas dari sistem ditentukan dan menentukan hubungan antar kelas dan turunannya (IBM, 2018).

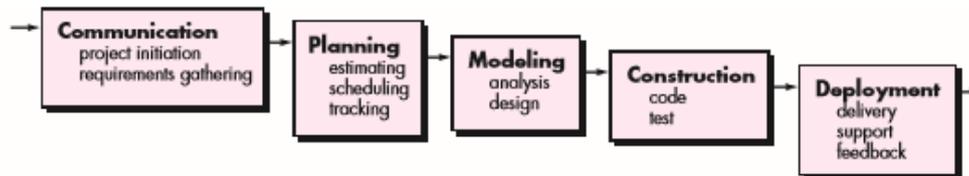


Gambar 2.3 Class Diagram

(Sumber: *Unified Modeling Language Reference Manual, The (2nd Edition)*, 2010)

2.11 Waterfall Model

Waterfall Model adalah atau biasa disebut klasik *Life Cycle* adalah model klasik yang bersifat sistematis, berurutan dalam membangun perangkat lunak (Roger S. Pressman, 2010).



Gambar 2.1 Alur Waterfall

(Sumber: Roger S. Pressman, 2010)

Alur dalam model *waterfall* dapat dilihat dari diagram alir diatas, dengan langkah – langkah sebagai berikut:

1. Communication

Pada tahap ini akan dilakukan inisiasi proyek, seperti menganalisis masalah yang ada dan tujuan yang akan dicapai. Selain itu dilakukan juga pengumpulan kebutuhan, dimana pengumpulan kebutuhan didapatkan dari stakeholder.

2. Planning

Tahap ini merupakan tahap dimana akan dilakukan estimasi mengenai kebutuhan - kebutuhan yang diperlukan untuk membuat sebuah sistem.

Selain itu, penjadwalan dalam proses pengerjaan juga ditentukan pada tahap ini.

3. *Modeling*

Kemudian mulai masuk pada tahap perancangan dimana perancang menerjemahkan kebutuhan sistem kedalam representasi untuk menilai kualitas sebelum tahap selanjutnya dikerjakan. Tahap ini lebih difokuskan pada atribut program, seperti struktur data, arsitektur perangkat lunak, dan detail prosedur perangkat lunak.

4. *Construction*

Proses ini merupakan pembuatan kode. Pengkodean atau *Coding* adalah penerjemah desain dalam bahasa yang bias dikenali oleh computer. Proses ini menterjemahkan kebutuhan dari stakeholder kedalam bahasa yang dimengerti oleh komputer. Setelah tahap *coding* akan dilakukan tahap *testing* terhadap sistem yang telah dibuat. Tujuan dari *testing* adalah menemukan kesalahan-kesalahan terhadap sistem tersebut untuk kemudian bisa diperbaiki.

5. *Deployment*

Tahap ini bisa dikatakan final dalam pembuatan sebuah perangkat lunak. Setelah melakukan analisis, desain dan *coding* maka sistem yang sudah jadi akan digunakan oleh *user*. Kemudian perangkat lunak harus dilakukan pemeliharaan secara berkala.

2.12 Pengujian Sistem

Pada Tahap pengujian sistem, dilakukan pengujian kepada sistem yang telah dibuat menggunakan skenario-skenario yang mungkin dialami oleh sistem untuk menguji fungsionalitasnya, apakah telah sesuai dengan perencanaan atau belum.

2.12.1 Unit Testing

Unit testing merupakan pengujian yang berfokus pada usaha dalam melakukan verifikasi pada unit terkecil dalam suatu perangkat lunak yakni modul atau komponen perangkat lunak. Pada unit testing terdapat jalur dari komponen untuk diuji yang bertujuan untuk mengungkap kesalahan dalam batas komponen. Proses yang dibutuhkan dalam melakukan unit testing adalah membuat test case yang berisi tahapan dalam melakukan pengujian (Pressman, 2010).

Unit Testing menggunakan teknik *White-box testing* yang merupakan pengujian program dengan cara pendekatan dimana tes dilakukan pada pengetahuan tentang struktur program dan komponennya (Pressman, 2010). Pada pengujian ini berfokus pada pencarian jalur independent kode program. Jalur independent kode program dapat diketahui melalui perancangan *flow graph*. Flow graph merupakan diagram yang menggambarkan alur kode program yang terdiri dari *node* dan *edge*. *node* direpresentasikan dalam bulatan berisi data dan *edge* direpresentasikan sebagai penghubung antara node satu dan node yang lainnya yang memiliki arah pada satu simpul tertentu.

2.12.2 Integration Testing

Integration testing yaitu pengujian untuk membangun arsitektur perangkat lunak secara sistematis dan disaat bersamaan melakukan pengujian untuk mendeteksi kesalahan pada sistem. Pengujian ini berfokus pada komponen yang telah diuji pada *white-box testing* dan membangun struktur program sesuai desain (Pressman, 2010).

Teknik yang digunakan pada pengujian integrasi ini adalah *Non-incremental itegration* yang bertujuan untuk membangun perangkat lunak dengan pendekatan “big bang” dimana semua unit/komponen yang sudah diuji digabungkan dahulu lalu bisa dilakukan pengujian. Hasil yang dikeluarkan dalam pendekatan big bang sering kali tidak diharapkan, dikarenakan kesalahan pada suatu unit/komponen akan sangat sulit melakukan koreksi karena batasan yang luas. Teknik *incremental itegration* adalah teknik yang berbeda karena kebalikan dari teknik *non-incremental itegration* karena dalam teknik ini perangkat lunak dibangun dan diuji secara bertahap karena untuk meminimalisir kesalahan yang ditemukan dan lebih mudah untuk dikoreksi (Pressman, 2010).

2.12.3 Validation Testing

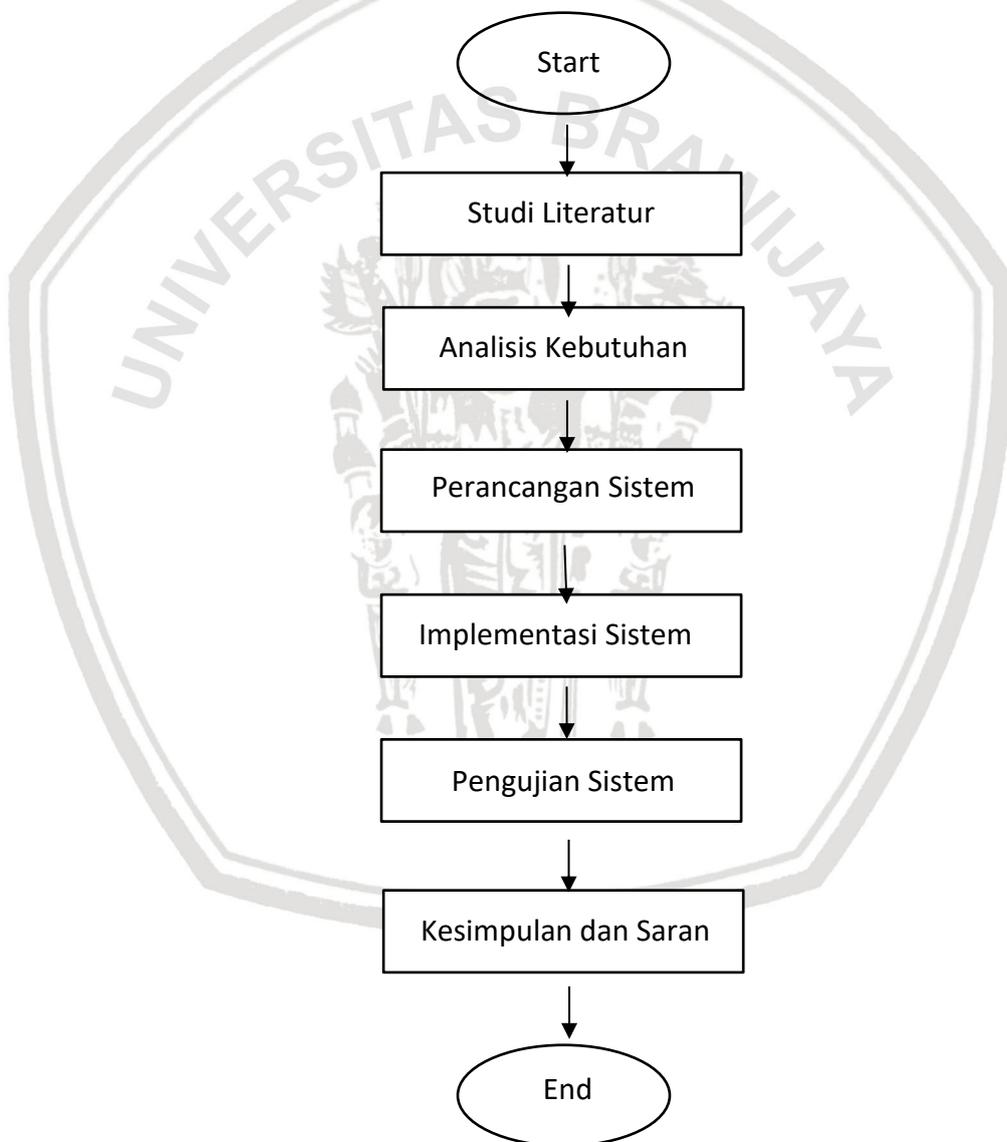
Validation testing merupakan pengujian yang dilakukan pada tahap akhir, untuk melakukan pengujian ini perangkat lunak telah dijadikan satu dan telah dilakukan pengujian itegrasi sehingga kesalahan yang ditemukan dapat diminimalisir. Pengujian validasi berfokus pada keluaran yang dihasilkan oleh sistem dan tindakan pengguna terhadap sistem, pengujian ini tidak ada perbedaan untuk perangkat lunak konvensional dan perangkat lunak berorientasi objek serta webapps. Pengujian validasi dianggap sukses apabila keluaran sistem sesuai dengan harapan pengguna (Pressman, 2010).

2.12.4 Compatibility Testing

Compatibility testing adalah pengujian untuk mengetahui masalah – masalah yang terjadi ketika sistem dijalankan pada sistem operasi, browser atau perangkat layar yang berbeda (Pressman, 2010). Pengujian ini berfokus untuk meminimalisir kesalahan yang terjadi sehingga kenyamanan pengguna terganggu. *Compability testing* digunakan untuk melihat apakah sistem berjalan lancar meskipun dijalankan pada browser yang berbeda. Masalah yang muncul dalam pengujian ini berupa tata letak pada tampilan tidak sesuai, perubahan *font*, atau ada fungsi yang tidak berjalan karena browser tidak menggunakan *library* yang dibutuhkan.

BAB 3 METODOLOGI

Metodologi penelitian, dari kata metode yang berarti cara atau langkah untuk melakukan suatu hal. Logos yang berarti ilmu atau pengetahuan, sehingga arti dari metodologi adalah cara untuk melakukan suatu hal untuk mencapai tujuan menggunakan ilmu pengetahuan. Sedangkan penelitian sendiri merupakan pemikiran yang sistematis untuk memecahkan masalah dengan fakta-fakta yang ada. Penelitian yang akan dilakukan disini merupakan jenis penelitian studi kasus, yang mana pada penelitian ini mempelajari individu atau kelompok tertentu yang mengalami kasus/masalah, dimana hasil dari penelitian ini adalah solusi dari kasus atau masalah yang dihadapi. Tahapan metodologi penelitian ini dijelaskan pada diagram dibawah ini:



Gambar 3.1 Metodologi Penelitian

Guna mengetahui permasalahan yang terdapat pada Dinas Pendidikan Kota Serang terkait dengan penyaluran dana bantuan untuk sekolah-sekolah yang membutuhkan, pada tahapan analisis kebutuhan penulis melakukan wawancara dengan Kepala Seksi Sarana Bidang Sekolah Dasar dan Kepala Dinas Pendidikan dan Kebudayaan Kota Serang sebagai narasumber. Wawancara tersebut membahas bagaimana alur penyaluran dana bantuan sekolah terutama proses penentuan sekolah calon penerima dana bantuan dan permasalahan apa saja yang didapatkan selama proses tersebut.

Setelah proses wawancara selesai dan penulis sudah mendapatkan point permasalahan berikut dengan solusi yang ditawarkan penulis untuk Dinas Pendidikan dan Kebudayaan Kota Serang yaitu Sistem Pengelolaan Pendistribusian Dana Bantuan Sekolah, selanjutnya penulis melakukan observasi mengenai metode dan teori yang digunakan sebagai dasar perancangan dan implementasi penelitian seperti bahasa pemrograman dan metode pengembangan perangkat lunak yang akan digunakan dalam pengembangan Sistem Pengelolaan Pendistribusian Dana Bantuan Sekolah ini.

3.1 Studi Literatur

Penyusunan dasar teori disusun berdasarkan referensi yang diperoleh dari buku, artikel, jurnal, konferensi, serta penelitian – penelitian terkait. Penggunaan studi literatur dalam tahap penelitian adalah untuk referensi sebagai pengetahuan dasar dalam melakukan rekayasa kebutuhan, perancangan, dan implementasi serta pengujian.

3.2 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem dibutuhkan untuk menggali kebutuhan yang akan digunakan dalam proses perancangan. Menggali kebutuhan yang digunakan dalam penelitian ini adalah dengan wawancara. Metode analisis yang digunakan adalah *Object Oriented Analysis (OOA)* dengan menggunakan bahasa permodelan *Unified Modeling Language (UML)*.

1. Elisitasi Kebutuhan

Kebutuhan dari penelitian akan didapatkan dari hasil wawancara terhadap narasumber dari Kepala Seksi Sarana Bidang Sekolah Dasar Dinas Pendidikan dan Kebudayaan Kota Serang Bapak Yayan Kosasih, S.Pd, M.Si. wawancara dilakukan dengan memberikan pertanyaan tentang sistem yang sudah berjalan, masalah yang dihadapi, dan solusi yang akan diberikan untuk mengatasi masalah yang dihadapi. Setelah mendapatkan kebutuhan maka didapatkan kebutuhan yang akan menjadi dasar dalam proses perancangan.

2. Spesifikasi Kebutuhan

Kebutuhan umum yang telah didapatkan dari fase elisitasi kebutuhan dijelaskan secara mendetail. Identifikasi aktor juga dilakukan untuk mengetahui siapa yang akan menggunakan sistem nantinya. Kebutuhan

yang telah dispesifikasi akan dimodelkan menggunakan *use case diagram* dan *use case scenario*.

3. Validasi dan Verifikasi Kebutuhan

Tahap validasi dan verifikasi dilakukan agar kebutuhan yang telah didapatkan dan didefinisikan sudah benar, akurat, dan lengkap. Proses validasi dan verifikasi dilakukan dengan membuat *prototype* agar membantu pengembang dalam memahami kebutuhan dan memastikan bahwa kebutuhan yang telah didefinisikan dan dispesifikasikan adalah benar.

4. Manajemen Kebutuhan

Dalam menjaga konsistensi dari kebutuhan yang didapatkan maka setiap kebutuhan memerlukan kode unik yang membedakan antar satu kebutuhan dan kebutuhan dengan kebutuhan lainnya. Kode tersebut akan dibagi berdasarkan kebutuhan fungsional dan non-fungsional, kode unik yang digunakan pada sistem pengelolaan soal adalah SRS-SP-F-1 untuk kebutuhan fungsional dan SRS-SP-NF-1 untuk kebutuhan non-fungsional.

3.3 Perancangan Sistem

Perancangan sistem dikerjakan berdasarkan kebutuhan yang didapatkan dari analisa kebutuhan. Dalam penelitian ini perancangan dilakukan dengan metode *Object Oriented Design*, dimana suatu masalah akan dimodelkan menurut konsep sekitar dunia nyata dan diorganisir sebagai kumpulan objek yang memiliki struktur data dan perilakunya. Proses perancangan sistem ini dapat dilakukan setelah kebutuhan dari stackholder telah disepakati, sehingga dapat masuk ke proses perancangan.

Proses perancangan selesai apabila seluruh komponen perancangan dan arsitektur sistem telah selesai dan siap untuk diimplementasikan. Hasil dari tahap ini akan menghasilkan rancangan sistem berupa:

1. Perancangan Arsitektur

Pada perancangan arsitektur dilakukan pemodelan berupa sequence diagram dan class diagram. Sequence diagram merupakan pemodelan yang menggambarkan interaksi antar objek. Pada penelitian ini dibuat tiga sequence diagram yang merupakan fungsionalitas utama. Class diagram merupakan pemodelan yang menggambarkan relasi antar class pada sistem, dimana setiap class memiliki atribut dan oprasinya.

2. Perancangan Komponen

Pada perancangan komponen dituliskan algoritme dari tiga fungsionalitas utama dari sistem yang dibangun. Penulisan algoritme dari tiga fungsionalitas tersebut dituliskan dalam pseudocode.

3. Perancangan Basis Data

Pada perancangan data digunakan *Conceptual Data Model* (CDM) untuk memodelkan relasi data pada database yang akan dibuat. CDM ini nantinya akan diimplementasikan menjadi *Physical Data Model* (PDM) pada tahap implementasi.

4. Perancangan Antarmuka

Pada perancangan antarmuka, rancangan antarmuka sistem akan dibuat berupa wireframe dengan komponen-komponen pada antarmuka sistem. Penjelasan pada setiap komponen pada wireframe diberikan untuk menghindari kesalahan pada tahap implementasi.

3.4 Implementasi Sistem

Proses implementasi dapat dilakukan setelah rancangan sistem selesai dibuat. Implementasi dilakukan menggunakan Bahasa pemrograman web yaitu PHP untuk membuat sistemnya. Implementasi menggunakan PHP akan dilakukan sesuai perancangan yang telah dibuat. Penggunaan aplikasi desain seperti photoshop juga diperlukan sebagai media desain untuk menyiapkan aset yang diperlukan dalam pembangunan *website*. Proses yang berada pada tahap implementasi adalah sebagai berikut:

1. Implementasi Kode Program

Implementasi kode program dilakukan berdasarkan perancangan arsitektur dan komponen yang telah dilakukan, yang merujuk pada sequence diagram, class diagram, dan pseudocode.

2. Implementasi Basis Data

Implementasi basis data dilakukan berdasarkan perancangan data yang telah dikerjakan, yang merujuk pada conceptual data model (CDM) diimplementasikan menggunakan MySQL.

3. Implementasi Antarmuka

Implementasi antarmuka dilakukan berdasarkan perancangan antarmuka yang telah dibuat, yang merujuk pada wireframe dan penjelasan komponen antarmuka. Pada implementasinya digunakan bahasa pemrograman HTML, CSS, dan PHP dimana digunakan Codeigniter sebagai framework PHP dan Bootstrap sebagai framework CSS.

3.5 Pengujian

Pengujian dilakukan untuk melihat apakah sistem yang dibuat telah sesuai dengan kebutuhan dan dapat menyelesaikan masalah yang ada. Terdapat 4 pengujian yang akan dilakukan pada penelitian ini.

1. Pengujian Unit

Pengujian unit dilakukan dengan tujuan untuk menguji suatu komponen dari suatu sistem yang telah dibuat. Komponen yang diuji dapat berupa class atau objek berdasarkan perancangan komponen. Metode pengujian basis path testing digunakan untuk melakukan teknik pengujian whitebox.

2. Pengujian Integrasi

Pada pengujian integrasi ini digunakan strategi top-down integration dimana pengujian dilakukan dari modul tertinggi dan turun ke modul-modul dibawahnya. Pengujian integrasi dilakukan menggunakan teknik blackbox testing.

3. Pengujian Validasi

Pada pengujian validasi digunakan teknik blackbox testing dalam pengujiannya. Pengujian ini bertujuan untuk menguji kesesuaian kebutuhan yang telah didefinisikan pada sistem yang telah dibangun. Pengujian blackbox testing dilakukan dengan hanya melihat kondisi masukan dan keluaran yang dihasilkan sistem.

4. Pengujian *Compatibility*

Pengujian kesesuaian atau compability testing adalah pengujian yang menguji kesesuaian sistem untuk dapat berjalan pada beberapa platform, yang mana dalam penelitian ini sistem berjalan pada perambah web sehingga perlu diuji apakah sistem dapat berjalan pada beberapa perambah web yang ada. Pengujian dilakukan dengan bantuan *website powermapping*, untuk mengecek apakah terdapat bagian yang tidak sesuai antar satu perambah web dengan yang lainnya.

3.6 Kesimpulan dan Saran

Kesimpulan dilakukan setelah semua tahapan mulai dari studi literatur, analisis kebutuhan, perancangan sistem, implementasi sistem, dan pengujian sistem yang diterapkan sudah selesai dilakukan. Selanjutnya dapat diperoleh kesimpulan berdasarkan hasil penelitian yang dilakukan penulis berupa Sistem Pengelolaan Pendistribusian Dana Bantuan Sekolah. Sehingga dapat diperoleh penarikan kesimpulan untuk menjawab perumusan masalah yang telah dirumuskan. Tahap terakhir dalam penulisan yaitu saran. Saran berguna untuk memperbaiki sistem atau untuk pengembangan sistem selanjutnya.

BAB 4 ANALISIS KEBUTUHAN

Dalam analisis kebutuhan yang dilakukan pada bab ini menjelaskan gambaran umum sistem yang akan dibangun, aktor yang akan menggunakan aplikasi dan kebutuhan dalam membangun sistem.

4.1 Elisitasi Kebutuhan

Elisitasi kebutuhan sistem dilakukan dengan menentukan kebutuhan umum yang didapatkan ketika elisitasi kebutuhan. Penggalan atau elisitasi kebutuhan dengan cara wawancara di Dinas Pendidikan dan Kebudayaan Kota Serang. Kebutuhan dari penelitian akan didapatkan dari hasil wawancara terhadap narasumber dari Kepala Seksi Sarana Bidang Sekolah Dasar Dinas Pendidikan dan Kebudayaan Kota Serang Bapak Yayan Kosasih, S.Pd, M.Si. Dari hasil wawancara yang dilakukan terdapat masalah yang muncul yaitu beberapa sekolah merasa kesulitan dalam mengajukan dana bantuan sekolah dikarenakan prosesnya panjang dan rumit. Masalah yang kedua adalah beberapa sekolah juga kesulitan untuk mengajukan proposal pengajuan dana bantuan sarana dan prasarana sekolah dikarenakan jarak sekolah terlalu jauh dari kantor Dinas Pendidikan dan Kebudayaan Kota Serang. Masalah yang ketiga adalah kurang tepatnya alur dana distribusi, masih banyak sekolah yang harus segera dibantu tapi tidak tertutupi semua dikarenakan pengajuan dana bantuan sekolah diberikan kepada sekolah yang mengajukan dana bantuan sarana dan prasarana sekolah terlebih dahulu, bukan sekolah yang keadaannya butuh dibantu terlebih dahulu.

Bedasarkan masalah tersebut, maka didapatkan kebutuhan sebagai solusi dari masalah berupa sistem yang mampu mengurangi kerumitan proses pengajuan dana bantuan sekolah dan membuat pengajuan dana bantuan sekolah berupa proposal dilakukan dengan *online*. Dari permasalahan ketiga sistem juga harus mampu menentukan urutan sekolah yang harus dibantu terlebih dahulu. Untuk dapat memprioritaskan urutan sekolah yang butuh bantuan terlebih dahulu terdapat empat aspek yang diperhitungkan, dari empat aspek tersebut dihitung dan diurutkan berdasarkan hasil persentase perhitungan. Adapun empat aspek tersebut adalah jumlah ruangan rusak, jumlah siswa, jumlah rombongan belajar, dan tingkat kerusakan. Setelah menghitung nilai dari empat aspek tersebut, lalu nilai dari masing – masing aspek di hitung persentase dengan bobot jumlah ruangan rusak yaitu 30%, jumlah siswa yaitu 10%, jumlah rombongan belajar yaitu 20%, dan tingkat kerusakan yaitu 40% yang mana dasar perhitungan persentase didapatkan dari Bapak Yayan Kosasih, S.Pd, M.Si. sebagai Kepala Seksi Sarana Bidang SD Dinas Pendidikan dan Kebudayaan Kota Serang yang dapat dilihat pada Lampiran C.

4.2 Gambaran Umum Aplikasi

Sistem Pengelolaan Pendistribusian Dana Bantuan Sekolah ini bertujuan untuk menampung proposal – proposal sekolah tentang pengajuan dana bantuan sarana sekolah berupa proposal pengajuan dana bantuan sarana dan prasarana

sekolah serta dapat memonitoring proses bantuan sarana dan pra sarana sekolah. Proses pertama dari sistem adalah membuat proposal yang nantinya proposal akan diterima oleh pihak Dinas Pendidikan dan Kebudayaan Kota Serang dan data – data proposal akan di verifikasi oleh tim verifikasi Dinas Pendidikan. Setelah data dari proposal sudah diverifikasi, proposal akan diurutkan berdasarkan data – data yang sudah diinputkan didalam proposal. Proposal yang sudah masuk dalam daftar prioritas akan diberikan bantuan dana dan pihak sekolah akan melaporkan proses dari bantuan sekolah berupa laporan terpakainya dana dan proses pembangunan sarana sekolah serta terdapat rekap proses bantuan setelah semua proses pengajuan sampai pengiriman dana selesai.

4.3 Identifikasi Aktor

Aktor merupakan seseorang ataupun sistem yang dapat berinteraksi dengan sistem. Adapun aktor dalam aplikasi ini ditunjukkan pada tabel 4.1.

Tabel 4.1 Identifikasi Aktor

Aktor	Deskripsi
User	Merupakan aktor umum yang belum dikenali oleh sistem.
Admin Diknas	Merupakan Admin IT dari Dinas Pendidikan dan Kebudayaan Kota Serang yang membuat akun Admin Sekolah, Kepala Seksi, dan Tim Verifikasi.
Kepala Seksi	Merupakan Kepala Seksi Sarana Bidang PAUD, SD, dan SMP dari Dinas Pendidikan dan Kebudayaan Kota Serang yang memeriksa, menyetujui dan menolak, dan mengirim dana bantuan proposal - proposal tingkat PAUD, SD, dan SMP dari Admin Sekolah.
Tim Verifikasi	Merupakan tim dari Dinas Pendidikan dan Kebudayaan Kota Serang untuk memverifikasi data dari proposal – proposal bantuan sekolah.
Admin Sekolah	Merupakan pihak sekolah yang ingin mengajukan bantuan perbaikan sekolah dengan cara

	membuat proposal pengajuan perbaikan sekolah.
--	---

4.4 Kebutuhan Fungsional Sistem

Kebutuhan fungsional harus mencakup apa saja yang harus diselesaikan oleh sistem untuk dapat menyelesaikan masalah yang telah ditentukan sebelumnya. Kebutuhan fungsional juga harus diberikan suatu identitas untuk mempermudah proses identifikasi kebutuhan juga untuk menjaga konsistensi yang terkait dengan kebutuhan suatu sistem, mulai dari proses perancangan hingga pengujian sistem selesai dilakukan. Kebutuhan fungsional sistem didapatkan dengan wawancara terhadap Kepala Seksi Sarana Bidang Sekolah Dasar Dinas Pendidikan dan Kebudayaan Kota Serang Bapak Yayan Kosasih, S.Pd, M.Si. Adapun kebutuhan fungsional sistem serta spesifikasinya ditunjukkan pada Tabel 4.2. Setiap Kebutuhan akan diberikan kode SRS-SP-F-X untuk kebutuhan fungsional. SRS merupakan *System Requirement Specification*, SP merupakan Sistem Pengelolaan, F untuk kebutuhan fungsional dan X menunjukkan nomor dari definisi kebutuhan utama.

Tabel 4.2 Kebutuhan Fungsional Sistem

No.	Kode Kebutuhan Sistem	Deskripsi Kebutuhan	Nama Use Case
1	SRS-SP-F-1	Sistem harus mampu <i>login</i> sehingga user dapat masuk kedalam sistem sesuai hak akses yang dimiliki.	Login
1.1	SRS-SP-F-1-1	Sistem menyediakan <i>form login</i> berupa <i>email</i> dan <i>password</i> .	
2	SRS-SP-F-2	Sistem harus mampu <i>logout</i> agar user yang telah masuk dapat keluar dari sistem.	Logout
3	SRS-SP-F-3	Sistem harus mampu membuat proposal untuk meminta bantuan perbaikan sekolah.	Buat Proposal
3.1	SRS-SP-F-3-1	Sistem menyediakan <i>form</i> buat proposal berupa judul proposal, deskripsi proposal, upload gambar, upload berkas proposal, upload berkas permohonan	

		bantuan, upload surat pengantar dinas, jumlah siswa, jumlah rombongan belajar, jumlah rusak, tingkat kerusakan, tingkat sekolah, jenis sekolah, dan jumlah dana diajukan.	
3.2	SRS-SP-F-4-3	Sistem mampu mengidentifikasi format file yang di <i>upload</i> .	
4	SRS-SP-F-4	Sistem harus mampu ubah proposal yang sudah dibuat	Ubah Proposal
4.1	SRS-SP-F-4-1	Sistem menampilkan <i>form</i> buat proposal berupa judul proposal, deskripsi proposal, upload gambar, upload berkas proposal, upload berkas permohonan bantuan, upload surat pengantar dinas, jumlah siswa, jumlah rombongan belajar, jumlah rusak, tingkat kerusakan, tingkat sekolah, jenis sekolah, dan jumlah dana diajukan.	
5	SRS-SP-F-5	Sistem harus mampu hapus laporan yang sudah dibuat.	Hapus Proposal
6	SRS-SP-F-6	Sistem harus mampu untuk melihat daftar proposal yang sudah dibuat.	Lihat Daftar Proposal
6.1	SRS-SP-F-6-1	Sistem menampilkan data proposal berupa judul proposal, tanggal proposal dibuat, status proposal, dan tombol " <i>Detail Proposal</i> " untuk melihat <i>detail</i> proposal.	
7	SRS-SP-F-7	Sistem harus mampu melihat detail laporan yang sudah dibuat.	Lihat Detail Proposal
7.1	SRS-SP-F-7-1	Sistem menampilkan data detail proposal berupa judul	

		proposal, deskripsi proposal, status proposal, gambar kerusakan, berkas proposal, berkas permohonan bantuan, berkas surat pengantar dinas, jumlah siswa, jumlah rombongan belajar, jumlah ruangan rusak, tingkat kerusakan, jumlah dana diajukan, total bantuan, dana diturunkan sementara, dan dana terpakai.	
8	SRS-SP-F-8	Sistem harus mampu untuk <i>approve</i> proposal yang sudah dibuat.	<i>Approve Proposal</i>
9	SRS-SP-F-9	Sistem harus mampu untuk <i>decline</i> proposal yang sudah dibuat.	<i>Decline Proposal</i>
10	SRS-SP-F-10	Sistem harus mampu mengirim tugas untuk verifikasi.	Kirim Tugas Verifikasi
10.1	SRS-SP-F-10-1	Sistem menampilkan <i>form</i> tugas verifikasi berupa batas tanggal awal verifikasi, batas akhir verifikasi, dan berkas surat tugas verifikasi.	
11	SRS-SP-F-11	Sistem harus mampu verifikasi proposal yang sudah diminta diverifikasi.	Verifikasi Proposal
11.1	SRS-SP-F-11-1	Sistem menyediakan <i>form</i> verifikasi berupa checklist data verifikasi dan upload berkas verifikasi.	
12	SRS-SP-F-12	Sistem harus mampu mengirim notifikasi kepada admin sekolah, tim verifikasi, dan kepala seksi.	Notifikasi
12.1	SRS-SP-F-12-1	Sistem mengirimkan data notifikasi berupa tanggal notifikasi dan data notifikasi.	

13	SRS-SP-F-13	Sistem harus mampu untuk melihat daftar proposal yang akan dibantu terlebih dahulu.	Lihat Daftar Prioritas
13.1	SRS-SP-F-13-1	Sistem menampilkan data proposal prioritas berupa judul laporan, status proposal, rate prioritas, dan tombol "Detail Proposal"	
14	SRS-SP-F-14	Sistem harus mampu untuk kirim dana proposal yang sedang dibantu.	Kirim Dana
14.1	SRS-SP-F-14-1	Sistem menyediakan form kirim dana berupa nomor rekening, nama rekening, rekening bank tujuan, jumlah dana, foto bukti, dan upload berkas pengiriman dana	
15	SRS-SP-F-15	Sistem harus mampu input dana terpakai dari proposal.	Input Dana Terpakai
15.1	SRS-SP-F-15-1	Sistem menyediakan form input dana terpakai berupa nama barang yang dibeli, harga, jumlah dana terpakai, dan upload berkas dana terpakai.	
16	SRS-SP-F-16	Sistem harus mampu menampilkan data log aktivitas proposal.	Log Aktivitas Proposal
16.1	SRS-SP-F-16-1	Sistem menampilkan data log aktivitas proposal berupa berkas pengiriman dana dan berkas laporan pemakaian dana.	
17	SRS-SP-F-17	Sistem harus mampu untuk menampilkan halaman daftar sekolah yang sudah ada didalam sistem.	Daftar Sekolah
17.1	SRS-SP-F-17-1	Sistem menampilkan data sekolah berupa nama	

		sekolah, nomor pokok sekolah nasional (NPSN), alamat sekolah, nomor telepon sekolah, nomor rekening sekolah dan <i>email</i> sekolah.	
18	SRS-SP-F-18	Sistem harus mampu untuk menampilkan halaman daftar kasi yang ada didalam sistem	Daftar Kasi
18.1	SRS-SP-F-18	Sistem menampilkan data akun kasi berupa tingkat bidang, username, dan email.	
19	SRS-SP-F-19	Sistem harus mampu untuk menampilkan halaman daftar tim verifikasi yang ada didalam sistem	Daftar Tim Verifikasi
19.1	SRS-SP-F-19-1	Sistem menampilkan data akun tim verifikasi berupa username dan email.	
20	SRS-SP-F-20	Sistem harus mampu menyimpan data user yang didaftarkan.	Tambah Sekolah
20.1	SRS-SP-F-20-1	Sistem menyediakan <i>form</i> tambah sekolah berupa nama sekolah, tingkat sekolah, nomor pokok sekolah nasional (NPSN), alamat sekolah, nomor telepon sekolah, nomor rekening sekolah, nama pemilik rekening sekolah, nama bank, <i>email</i> sekolah dan <i>password</i> .	
21	SRS-SP-F-21	Sistem harus mampu edit sekolah yang sudah ada didalam sistem	Ubah Sekolah
21.1	SRS-SP-F-21-1	Sistem menampilkan <i>form</i> tambah sekolah berupa nama sekolah, tingkat sekolah, nomor pokok	

		sekolah nasional (NPSN), alamat sekolah, nomor telepon sekolah, nomor rekening sekolah, nama pemilik rekening sekolah, nama bank, <i>email</i> sekolah dan <i>password</i> yang telah terisi sesuai data saat itu.	
22	SRS-SP-F-22	Sistem harus mampu menghapus data sekolah yang sudah ada didalam sistem	Hapus Sekolah
23	SRS-SP-F-23	Sistem harus mampu menyimpan data kepala seksi yang didaftarkan.	Tambah Kepala Seksi
23.1	SRS-SP-F-23-1	Sistem menampilkan <i>form tambah</i> akun berupa tingkatan kasi, username, email dan password, NIP, alamat, dan nomor hp.	
24	SRS-SP-F-24	Sistem harus mampu ubah akun kepala seksi yang sudah ada didalam sistem	Ubah Kepala Seksi
24.1	SRS-SP-24-1	Sistem menampilkan <i>form</i> ubah akun berupa tingkatan kasi, username, email dan password, NIP, alamat, dan nomor hp.	
25	SRS-SP-F-25	Sistem harus mampu menghapus data kepala kasi yang sudah ada didalam sistem	Hapus Kepala Seksi
26	SRS-SP-F-26	Sistem harus mampu menyimpan data tim verifikasi yang didaftarkan.	Tambah Tim Verifikasi
26.1	SRS-SP-F-26-1	Sistem menampilkan <i>form</i> ubah akun berupa username, email, password, NIP, alamat, dan nomor hp	

27	SRS-SP-F-27	Sistem harus mampu ubah akun tim verifikasi yang sudah ada didalam sistem	Ubah Tim Verifikasi
27.1	SRS-SP-27-1	Sistem menampilkan <i>form</i> ubah akun berupa username, email, password, NIP, alamat, dan nomor hp.	
28	SRS-SP-F-28	Sistem harus mampu menghapus data tim verifikasi yang sudah ada didalam sistem	Hapus Tim Verifikasi
29	SRS-SP-F-29	Sistem harus mampu menampilkan data rekap bantuan.	Lihat Rekap Bantuan
29.1	SRS-SP-F-29-1	Sistem menampilkan data rekap bantuan berupa data proposal, data verifikasi proposal, data terima proposal, data kirim dana proposal, data update dana proposal.	

4.5 Kebutuhan Non Fungsional Sistem

Setiap Kebutuhan akan diberikan kode SRS-SP-NF-X untuk kebutuhan non-fungsional. SRS merupakan singkatan *System Requirement Spesification*, SP merupakan singkatan dari Sistem Pengelolaan, NF untuk kebutuhan non fungsional dan X menunjukkan nomor dari definisi kebutuhan utama. Kebutuhan non fungsional akan ditunjukkan pada Tabel 4.3.

Tabel 4.3 Kebutuhan Non Fungsional Sistem

No.	Kode Kebutuhan Sistem	Deskripsi Kabutuhan
1	SRS-SP-NF-1	<i>Compatibility</i> , sistem dapat berjalan dengan baik pada beberapa browser seperti Mozilla Firefox, Google Chrome, Opera, Internet Exploler, Microsoft Edge, iOS, Android dan Safari.

4.6 Validasi dan Verifikasi

Pada proses analisis kebutuhan dilakukan proses validasi dan verifikasi. Proses validasi adalah untuk mengetahui apakah sistem yang kita bangun adalah benar. Sedangkan proses verifikasi adalah untuk mengetahui apakah sistem dibangun dengan benar. Proses validasi dan verifikasi dilakukan dengan cara menunjukan

daftar kebutuhan dan spesifikasi kebutuhan yang sudah dibuat kepada *stakeholder* yaitu Bapak Yayan Kosasih, S.Pd, M.Sc selaku Kepala Seksi Sarana Bidang SD Dinas Pendidikan dan Kebudayaan Kota Serang. Dari proses tersebut dihasilkan bahwa daftar kebutuhan dan spesifikasi kebutuhan yang telah disetujui oleh *stakeholder*.

4.7 Use Case Diagram



Gambar 4.1 Use Case Diagram Sistem

Use case diagram berisi sejumlah aksi yang dilakukan oleh aktor kepada sistem untuk mencapai tujuan tertentu. Aktor dalam diagram *use case* bisa berupa



manusia atau sistem eksternal yang berinteraksi dengan sistem yang dimaksud. Diagram *use case* ditunjukkan dalam Gambar 4.1 diatas.

4.8 Use Case Scenario

Use Case Scenario adalah penjabaran dari *use case diagram* yang telah digambarkan. Pada *use case scenario* proses penjabaran dilakukan dengan menjelaskan nama *use case*, kode *use case*, actor sebagai pengguna *use case*, kondisi awal yang harus dipenuhi sebelum menggunakan *use case* (*pre condition*), alur kerja *use case* untuk mencapai tujuan (*main flow*), alur alternatif jika *main flow* tidak sesuai harapan (*alternative flow*, dan hasil akhir setelah actor menjalankan *main flow* (*post condition*). *Use Case Scenario* akan ditunjukkan oleh Tabel 4.4 hingga Tabel 4.32.

Tabel 4.4 Use Case Scenario Login

Nama Use Case	Login
Kode Kebutuhan Terkait	SRS-SP-F-1
Aktor	User
Tujuan	Mengenali User dan memberikan hak akses sesuai dengan otoritas User.
Pre Condition	User berada pada halaman login
Main flow	<ol style="list-style-type: none"> 1. User memasukkan <i>email</i> dan <i>password</i> dan menekan tombol <i>login</i> 2. Sistem melakukan pencocokan terhadap <i>email</i> dan <i>password</i> yang ada di <i>database</i>, membuat <i>cookies</i>, dan menampilkan halaman <i>dashboard</i>
Alternative Flow	<ol style="list-style-type: none"> 1. Jika <i>email</i> atau <i>password</i> tidak diisi proses login tidak dapat dilanjutkan dan sistem meminta untuk mengisi kolom yang kosong 2. Jika <i>email</i> atau <i>password</i> maka proses akan kembali ke halaman login.
Post Condition	User berhasil masuk kedalam sistem

Tabel 4.5 Use Case Scenario Logout

Nama Use Case	Logout
Kode Kebutuhan Terkait	SRS-SP-F-2
Aktor	Kepala Seksi, Admin Diknas, Tim Verifikasi, dan Admin Sekolah

Tujuan	Kepala Seksi, Admin Diknas, Tim Verifikasi, dan Admin Sekolah dapat keluar dari sistem.
<i>Pre Condition</i>	Kepala Seksi, Admin Diknas, Tim Verifikasi, dan Admin Sekolah telah melakukan <i>login</i>
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Aktor menekan tombol logout 2. Sistem akan <i>direct</i> halaman utama kembali ke halaman <i>login</i>
<i>Alternative Flow</i>	-
<i>Post Condition</i>	Kepala Seksi, Admin Diknas, Tim Verifikasi, dan Admin Sekolah berhasil keluar dari sistem

Tabel 4.6 Use Case Scenario Buat Proposal

Nama Use Case	Buat Proposal
Kode Kebutuhan Terkait	SRS-SP-F-3
Aktor	Admin Sekolah
Tujuan	Admin Sekolah dapat membuat proposal bantuan perbaikan sekolah.
<i>Pre Condition</i>	Admin Sekolah telah melakukan login.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Admin Sekolah menekan halaman buat proposal pada halaman utama Admin Sekolah. 2. Sistem menampilkan halaman buat proposal. 3. Admin Sekolah mengisi form berupa judul proposal, deskripsi proposal, upload gambar, upload berkas proposal, upload berkas permohonan bantuan, upload surat pengantar dinas, jumlah siswa, jumlah rombongan belajar, jumlah rusak, tingkat kerusakan, tingkat sekolah, jenis sekolah, dan jumlah dana diajukan. 4. Sistem menyimpan data yang telah dimasukkan oleh Admin Sekolah pada database dan menampilkan halaman sebelumnya. 5. Sistem menampilkan pesan "Buat Proposal Berhasil"

<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Jika salah satu <i>input</i> tidak diisi sistem menampilkan pesan untuk mengisi <i>form</i> dengan lengkap. 2. Jika upload file selain docx/pdf maka akan tampil pesan gagal untuk membuat proposal. 3. Jika upload file melebihi 2mb maka tampil pesan gagal untuk membuat proposal.
<i>Post Condition</i>	Admin Sekolah berhasil membuat proposal.

Tabel 4.7 Use Case Scenario Ubah Proposal

Nama Use Case	Ubah Proposal
Kode Kebutuhan Terkait	SRS-SP-F-4
Aktor	Admin Sekolah
Tujuan	Admin Sekolah dapat mengubah proposal yang sudah dibuat.
<i>Pre Condition</i>	Admin Sekolah sudah login dan berada pada halaman utama.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Admin sekolah menekan halaman Daftar Proposal 2. Sistem menampilkan halaman Daftar Proposal. 3. Admin Sekolah menekan tombol “Lihat Detail Proposal”. 4. Sistem menampilkan detail dari proposal yang sudah dibuat. 5. Admin Sekolah menekan tombol “Ubah”. 6. Sistem Menampilkan form ubah proposal berupa judul proposal, deskripsi proposal, upload gambar, upload berkas proposal, upload berkas permohonan bantuan, upload surat pengantar dinas, jumlah siswa, jumlah rombongan belajar, jumlah rusak, tingkat kerusakan, tingkat sekolah, jenis sekolah, dan jumlah dana diajukan. 7. Admin Sekolah mengubah data laporan berupa judul laporan dan latar belakang.

	8. Sistem akan melakukan update data sesuai isi yang diubah oleh Admin Sekolah dan menampilkan pesan sukses.
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Jika salah satu <i>input</i> tidak diisi sistem menampilkan pesan untuk mengisi <i>form</i> dengan lengkap. 2. Jika input file selain docx/pdf maka akan tampil pesan gagal untuk mengubah proposal. 3. Jika upload file melebihi 2mb maka tampil pesan gagal untuk mengubah proposal.
<i>Post Condition</i>	Admin Sekolah berhasil mengubah data proposal.

Tabel 4.8 Use Case Scenario Hapus Proposal

Nama Use Case	Hapus Proposal
Kode Kebutuhan Terkait	SRS-SP-F-5
Aktor	Admin Sekolah
Tujuan	Admin Sekolah dapat menghapus proposal yang sudah dibuat.
<i>Pre Condition</i>	Admin Sekolah sudah login.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Admin sekolah menekan halaman Daftar Proposal. 2. Sistem menampilkan halaman Daftar Proposal. 3. Admin Sekolah menekan tombol "Detail Proposal". 4. Sistem menampilkan detail dari proposal yang sudah dibuat. 5. Admin Sekolah menekan tombol hapus. 6. Sistem menghapus data proposal dan menampilkan pesan sukses.
<i>Alternative Flow</i>	-
<i>Post Condition</i>	Data proposal berhasil dihapus oleh Admin Sekolah

Tabel 4.9 Use Case Scenario Lihat Daftar Proposal

Nama Use Case	Lihat Daftar Proposal
Kode Kebutuhan Terkait	SRS-SP-F-6
Aktor	Kepala Seksi dan Admin Sekolah
Tujuan	Kepala Seksi dan Admin Sekolah dapat melihat Daftar Proposal yang sudah dibuat oleh Admin Sekolah
Pre Condition	Admin Diknas dan Admin Sekolah sudah login.
Main flow	<ol style="list-style-type: none"> 1. Admin memilih halaman Daftar Proposal. 2. Sistem menampilkan halaman Daftar Proposal berupa judul proposal, tanggal proposal dibuat, status proposal, dan tombol "Detail Proposal" untuk melihat detail proposal.
Alternative Flow	-
Post Condition	Daftar Proposal dilihat oleh Admin Diknas dan Admin Sekolah

Tabel 4.10 Use Case Scenario Lihat Detail Proposal

Nama Use Case	Lihat Detail Proposal
Kode Kebutuhan Terkait	SRS-SP-F-7
Aktor	Kepala Seksi, Tim Verifikasi, dan Admin Sekolah
Tujuan	Kepala Seksi, Tim Verifikasi, dan Admin Sekolah dapat melihat detail proposal yang sudah dibuat oleh Admin Sekolah
Pre Condition	Kepala Seksi, Tim Verifikasi, dan Admin Sekolah sudah login.
Main flow	<ol style="list-style-type: none"> 1. Kepala Seksi, Tim Verifikasi, dan Admin Sekolah memilih halaman Daftar Proposal. 2. Sistem menampilkan halaman Daftar Proposal dan menampilkan Daftar Proposal yang sudah dibuat oleh Admin Sekolah 3. Kepala Seksi, Tim Verifikasi, dan Admin Sekolah menekan tombol Lihat Detail Proposal. 4. Sistem menampilkan detail berupa judul proposal, deskripsi proposal, status proposal, gambar kerusakan, berkas

	proposal, berkas permohonan bantuan, berkas surat pengantar dinas, jumlah siswa, jumlah rombongan belajar, jumlah ruangan rusak, tingkat kerusakan, jumlah dana diajukan, total bantuan, dana diturunkan sementara, dan dana terpakai.
<i>Alternative Flow</i>	-
<i>Post Condition</i>	Detail Proposal dilihat oleh Kepala Seksi, Tim Verifikasi, dan Admin Sekolah

Tabel 4.11 Use Case Scenario Approve Proposal

Nama Use Case	Approve Proposal
Kode Kebutuhan Terkait	SRS-SP-F-8
Aktor	Kepala Seksi
Tujuan	Kepala Seksi dapat <i>approve</i> proposal yang sudah dibuat oleh Admin Sekolah.
<i>Pre Condition</i>	Kepala Seksi sudah login.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Kepala Seksi memilih halaman Daftar Proposal. 2. Sistem menampilkan halaman Daftar Proposal dan menampilkan daftar proposal yang sudah dibuat oleh Admin Sekolah 3. Kepala Seksi menekan tombol Lihat Detail Proposal. 4. Sistem menampilkan detail berupa judul proposal, deskripsi proposal, status proposal, gambar kerusakan, berkas proposal, berkas permohonan bantuan, berkas surat pengantar dinas, jumlah siswa, jumlah rombongan belajar, jumlah ruangan rusak, tingkat kerusakan, jumlah dana diajukan, total bantuan, dana diturunkan sementara, dan dana terpakai, serta tombol approve dan tombol decline. 5. Kepala Seksi menekan tombol <i>approve</i>. 6. Kepala Seksi menekan tombol 'ya' pada pesan konfirmasi.

	7. Sistem menyimpan status <i>approve</i> pada proposal yang sudah di <i>approve</i>
<i>Alternative Flow</i>	-
<i>Post Condition</i>	Laporan berhasil di- <i>approve</i> oleh Kepala Seksi

Tabel 4.12 Use Case Scenario Decline Proposal

Nama Use Case	<i>Decline Proposal</i>
Kode Kebutuhan Terkait	SRS-SP-F-9
Aktor	Kepala Seksi
Tujuan	Kepala Seksi dapat <i>decline</i> proposal yang sudah dibuat oleh Admin Sekolah.
<i>Pre Condition</i>	Kepala Seksi sudah login.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Kepala Seksi memilih halaman Daftar Proposal. 2. Sistem menampilkan halaman Daftar Proposal dan menampilkan daftar proposal yang sudah dibuat oleh Admin Sekolah 3. Kepala Seksi menekan tombol Lihat Detail Proposal. 4. Sistem menampilkan detail berupa judul proposal, deskripsi proposal, status proposal, gambar kerusakan, berkas proposal, berkas permohonan bantuan, berkas surat pengantar dinas, jumlah siswa, jumlah rombongan belajar, jumlah ruangan rusak, tingkat kerusakan, jumlah dana diajukan, total bantuan, dana diturunkan sementara, dan dana terpakai, serta tombol approve dan tombol decline. 5. Kepala Seksi menekan tombol <i>decline</i>. 6. Kepala Seksi menekan tombol 'ya' pada pesan konfirmasi. 7. Sistem menyimpan status <i>decline</i> pada proposal yang sudah di <i>decline</i>
<i>Alternative Flow</i>	-
<i>Post Condition</i>	Proposal berhasil di- <i>decline</i> oleh Admin Diknas

Tabel 4.13 Use Case Scenario Kirim Tugas Verifikasi

Nama Use Case	Kirim Tugas Verifikasi
Kode Kebutuhan Terkait	SRS-SP-F-10
Aktor	Kepala Seksi
Tujuan	Kepala Seksi dapat mengirim tugas verifikasi kepala Tim Verifikasi.
Pre Condition	Kepala Seksi sudah login.
Main flow	<ol style="list-style-type: none"> 1. Kepala Seksi memilih halaman Daftar Proposal. 2. Sistem menampilkan halaman Daftar Proposal dan menampilkan daftar proposal yang sudah dibuat oleh Admin Sekolah 3. Kepala Seksi menekan tombol Lihat Detail Proposal. 4. Sistem menampilkan detail proposal berupa judul proposal, deskripsi proposal, status proposal, gambar kerusakan, berkas proposal, berkas permohonan bantuan, berkas surat pengantar dinas, jumlah siswa, jumlah rombongan belajar, jumlah ruangan rusak, tingkat kerusakan, jumlah dana diajukan, total bantuan, dana diturunkan sementara, dan dana terpakai, serta tombol "Kirim Tugas Verifikasi". 5. Kepala Seksi menekan tombol "Forward Tim Verifikasi". 6. Sistem menampilkan form berupa tanggal awal verifikasi dan batas akhir verifikasi. 7. Kepala seksi mengisi form "Forward Tim Verifikasi" 8. Sistem menyimpan data yang sudah diisi oleh Kepala Seksi.
Alternative Flow	<ol style="list-style-type: none"> 1. Jika salah satu <i>input</i> tidak diisi sistem menampilkan pesan untuk mengisi <i>form</i> dengan lengkap. 2. Jika input file selain docx/pdf maka akan tampil pesan gagal untuk kirim tugas verifikasi.

	3. Jika upload file melebihi 2mb maka tampil pesan gagal untuk kirim tugas verifikasi.
<i>Post Condition</i>	Kirim Tugas Verifikasi sudah dilakukan oleh Kepala Seksi

Tabel 4.14 Use Case Scenario Verifikasi Proposal

Nama Use Case	Lihat Daftar Verifikasi Proposal
Kode Kebutuhan Terkait	SRS-SP-F-11
Aktor	Tim Verifikasi
Tujuan	Tim Verifikasi dapat mem-verifikasi data proposal yang sudah diminta oleh Kepala Seksi.
<i>Pre Condition</i>	Tim Verifikasi sudah login.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Tim Verifikasi memilih halaman Daftar Verifikasi. 2. Sistem menampilkan halaman Daftar Verifikasi dan menampilkan daftar verifikasi yang sudah dibuat oleh Admin Sekolah 3. Tim Verifikasi menekan tombol Lihat Detail Proposal. 4. Sistem menampilkan detail proposal berupa judul proposal, deskripsi proposal, status proposal, gambar kerusakan, berkas proposal, berkas permohonan bantuan, berkas surat pengantar dinas, jumlah siswa, jumlah rombongan belajar, jumlah ruangan rusak, tingkat kerusakan, jumlah dana diajukan, total bantuan, dana diturunkan sementara, dan dana terpakai, serta tombol “Verifikasi Proposal”. 5. Tim Verifikasi menekan tombol “Verifikasi Proposal”. 6. Sistem menampilkan form verifikasi berupa checklist data verifikasi dan upload berkas verifikasi. 7. Tim verifikasi mengisi form verifikasi dan menekan tombol submit. 8. Sistem menyimpan data verifikasi.

<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Jika input file selain docx/pdf maka akan tampil pesan gagal untuk verifikasi proposal. 2. Jika upload file melebihi 2mb maka tampil pesan gagal untuk verifikasi proposal.
<i>Post Condition</i>	Verifikasi proposal berhasil dilakukan oleh Tim Verifikasi.

Tabel 4.15 Use Case Scenario Notifikasi

Nama Use Case	Notifikasi
Kode Kebutuhan Terkait	SRS-SP-F-12
Aktor	Kepala Seksi, Admin Sekolah, dan Tim Verifikasi
Tujuan	Kepala Seksi, Admin Sekolah dan Tim verifikasi dapat melihat notifikasi yang ada.
<i>Pre Condition</i>	Kepala Seksi dan Admin Sekolah sudah login.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Kepala Seksi, Admin Sekolah, dan Tim Verifikasi menekan tombol Notifikasi. 2. Sistem menampilkan halaman notifikasi berupa tanggal notifikasi dan data notifikasi.
<i>Alternative Flow</i>	-
<i>Post Condition</i>	Halaman notifikasi dilihat oleh Kepala Seksi, Admin Sekolah, dan Tim Verifikasi

Tabel 4.16 Use Case Scenario Lihat Daftar Prioritas

Nama Use Case	Lihat Daftar <i>Prioritas</i>
Kode Kebutuhan Terkait	SRS-SP-F-13
Aktor	Kepala Seksi
Tujuan	Kepala Seksi dapat melihat daftar proposal prioritas untuk mengetahui sekolah yang lebih dahulu dibantu.
<i>Pre Condition</i>	Kepala Seksi sudah login.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Admin memilih halaman Daftar <i>Prioritas</i>. 2. Sistem menampilkan halaman Daftar <i>Prioritas</i> dan menampilkan daftar proposal prioritas untuk mengetahui sekolah yang lebih dahulu dibantu.

<i>Alternative Flow</i>	-
<i>Post Condition</i>	Daftar Prioritas dilihat oleh Kepala Seksi.

Tabel 4.17 Use Case Scenario Kirim Dana

Nama Use Case	Kirim Dana
Kode Kebutuhan Terkait	SRS-SP-F-14
Aktor	Kepala Seksi
Tujuan	Mengirimkan dana ke sekolah untuk proposal yang sudah dibuat oleh Admin Sekolah.
<i>Pre Condition</i>	Kepala Seksi berada pada halaman utama
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Kepala Seksi memilih halaman Daftar Prioritas. 2. Sistem menampilkan halaman Daftar Prioritas dan menampilkan daftar proposal yang sudah dibuat oleh Admin Sekolah 3. Kepala Seksi menekan tombol Lihat Detail Proposal. 4. Sistem menampilkan detail proposal berupa judul proposal, deskripsi proposal, status proposal, gambar kerusakan, berkas proposal, berkas permohonan bantuan, berkas surat pengantar dinas, jumlah siswa, jumlah rombongan belajar, jumlah ruangan rusak, tingkat kerusakan, jumlah dana diajukan, total bantuan, dana diturunkan sementara, dan dana terpakai serta tombol "Kirim Dana". 5. Kepala Seksi menekan tombol "Kirim Dana". 6. Sistem menampilkan form berupa nomor rekening, nama rekening, rekening bank tujuan, jumlah dana, foto bukti, dan upload berkas kirim dana. 7. Kepala seksi mengisi form "Kirim Dana" 8. Sistem menyimpan data yang sudah diisi oleh Kepala Seksi.

<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Jika salah satu <i>input</i> tidak diisi sistem menampilkan pesan untuk mengisi <i>form</i> dengan lengkap. 2. Jika kirim dana melebihi total dana yang ditentukan maka pesan gagal untuk kirim dana. 3. Jika input file selain docx/pdf maka akan tampil pesan gagal untuk kirim dana. 4. Jika upload file melebihi 2mb maka tampil pesan gagal untuk kirim dana.
<i>Post Condition</i>	Dana berhasil dikirim oleh Kepala Seksi

Tabel 4.18 Use Case Scenario Input Dana Terpakai

Nama Use Case	<i>Input Dana Terpakai</i>
Kode Kebutuhan Terkait	SRS-SP-F-15
Aktor	Admin Sekolah
Tujuan	Admin Sekolah membuat laporan pemakaian dana.
<i>Pre Condition</i>	Admin Sekolah berada pada halaman utama
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Admin Sekolah memilih halaman Daftar Proposal. 2. Sistem menampilkan halaman Daftar Proposal dan menampilkan daftar proposal yang sudah dibuat oleh Admin Sekolah 3. Admin Sekolah menekan tombol Lihat Detail Proposal. 4. Sistem menampilkan detail proposal berupa judul proposal, deskripsi proposal, upload gambar, upload berkas proposal, upload berkas permohonan bantuan, upload surat pengantar dinas, jumlah siswa, jumlah rombongan belajar, jumlah rusak, tingkat kerusakan, tingkat sekolah, jenis sekolah, dan jumlah dana diajukan serta tombol "<i>Input Dana Terpakai</i>". 5. Admin Sekolah menekan tombol <i>Input Dana Terpakai</i>.

	<ol style="list-style-type: none"> 6. Sistem menampilkan form berupa nama barang yang dibeli, harga, jumlah dana terpakai, dan upload berkas dana terpakai. 7. Admin Sekolah mengisi form <i>Input Dana Terpakai</i>. 8. Sistem menyimpan data <i>Input Dana Terpakai</i>.
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Jika salah satu <i>input</i> tidak diisi sistem menampilkan pesan untuk mengisi <i>form</i> dengan lengkap. 2. Jika kirim dana melebihi total dana yang ditentukan maka pesan gagal untuk input dana terpakai. 3. Jika input file selain docx/pdf maka akan tampil pesan gagal untuk input dana terpakai. 4. Jika upload file melebihi 2mb maka tampil pesan gagal untuk input dana terpakai.
<i>Post Condition</i>	<i>Input Dana Terpakai</i> berhasil dibuat oleh Admin Sekolah.

Tabel 4.19 Use Case Scenario Log Aktivitas Proposal

Nama Use Case	Log Aktivitas Proposal
Kode Kebutuhan Terkait	SRS-SP-F-16
Aktor	Kepala Seksi dan Admin Sekolah
Tujuan	Melihat update dari progress bantuan dari kepala seksi dan admin sekolah
<i>Pre Condition</i>	Kepala Seksi dan Admin Sekolah
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Kepala Seksi dan Admin Sekolah memilih halaman Daftar Proposal. 2. Sistem menampilkan halaman Daftar Proposal 3. Kepala Seksi dan Admin Sekolah Menekan tombol Log Aktivitas Proposal 4. Sistem menampilkan halaman Log aktivitas Proposal berupa berkas pengiriman dana dan berkas laporan pemakaian dana.

<i>Alternative Flow</i>	-
<i>Post Condition</i>	Kepala Seksi dan Admin Sekolah berhasil melihat log aktivitas proposal.

Tabel 4.20 Use Case Scenario Daftar Sekolah

<i>Nama Use Case</i>	Daftar Sekolah
<i>Kode Kebutuhan Terkait</i>	SRS-SP-F-17
<i>Aktor</i>	Admin Diknas
<i>Tujuan</i>	Admin Diknas dapat melihat daftar sekolah yang sudah ditambahkan.
<i>Pre Condition</i>	Admin Diknas sudah login.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Admin memilih halaman Daftar Sekolah. 2. Sistem menampilkan halaman Daftar User dan menampilkan data sekolah berupa nama sekolah, nomor pokok sekolah nasional (NPSN), alamat sekolah, nomor telepon sekolah, dan email sekolah.
<i>Alternative Flow</i>	-
<i>Post Condition</i>	Daftar Sekolah dilihat oleh Admin Diknas.

Tabel 4.21 Use Case Scenario Daftar Kepala Seksi

<i>Nama Use Case</i>	Daftar Kepala Seksi
<i>Kode Kebutuhan Terkait</i>	SRS-SP-F-18
<i>Aktor</i>	Admin Diknas
<i>Tujuan</i>	Admin Diknas dapat melihat daftar kepala seksi yang sudah ditambahkan.
<i>Pre Condition</i>	Admin Diknas sudah login.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Admin memilih halaman Daftar Kepala Seksi. 2. Sistem menampilkan data akun kasi berupa tingkat sekolah, username, dan email.
<i>Alternative Flow</i>	-
<i>Post Condition</i>	Daftar Kasi dilihat oleh Admin Diknas.

Tabel 4.22 Use Case Scenario Daftar Tim Verifikasi

Nama Use Case	Daftar Tim Verifikasi
Kode Kebutuhan Terkait	SRS-SP-F-19
Aktor	Admin Diknas
Tujuan	Admin Diknas dapat melihat daftar tim verifikasi yang sudah ditambahkan.
Pre Condition	Admin Diknas sudah login.
Main flow	<ol style="list-style-type: none"> 1. Admin memilih halaman Daftar User. 2. Sistem harus mampu untuk menampilkan halaman daftar tim verifikasi berupa username dan email.
Alternative Flow	-
Post Condition	Daftar Tim Verifikasi dilihat oleh Admin Diknas.

Tabel 4.23 Use Case Scenario Tambah Sekolah

Nama Use Case	Tambah Sekolah
Kode Kebutuhan Terkait	SRS-SP-F-20
Aktor	Admin Diknas
Tujuan	Admin Diknas dapat mendaftarkan akun untuk dapat menggunakan fungsi sistem.
Pre Condition	Admin Diknas berada pada halaman utama Admin Diknas.
Main flow	<ol style="list-style-type: none"> 1. Admin Diknas menekan halaman tambah sekolah. 2. Sistem menampilkan <i>form</i> tambah sekolah berupa nama sekolah, tingkat sekolah, nomor pokok sekolah nasional(NPSN), alamat sekolah, nomor telepon sekolah, nomor rekening sekolah, nama pemilik rekening sekolah, nama bank, <i>email</i> sekolah dan <i>password</i>. 3. Admin Diknas mengisi <i>form</i> dengan lengkap dan menekan tombol submit.

	4. Sistem menyimpan data yang telah diisi oleh Admin Diknas.
<i>Alternative Flow</i>	1. Jika salah satu <i>input</i> tidak diisi sistem menampilkan pesan untuk mengisi <i>form</i> dengan lengkap.
<i>Post Condition</i>	Tambah Sekolah oleh Admin Diknas berhasil dilakukan

Tabel 4.24 Use Case Scenario Ubah Sekolah

Nama Use Case	Ubah Sekolah
Kode Kebutuhan Terkait	SRS-SP-F-21
Aktor	Admin Diknas
Tujuan	Admin Diknas dapat edit data user yang sudah register.
<i>Pre Condition</i>	Admin Diknas sudah login.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Admin memilih halaman Daftar User. 2. Sistem menampilkan halaman Daftar User. 3. Admin merubah data akun berupa nama sekolah, tingkat sekolah, nomor pokok sekolah nasional(NPSN), alamat sekolah, nomor telepon sekolah, nomor rekening sekolah, nama pemilik rekening sekolah, nama bank, <i>email</i> sekolah dan <i>password</i> 4. Sistem melakukan update data sesuai akun yang ingin dirubah.
<i>Alternative Flow</i>	1. Jika salah satu <i>input</i> tidak diisi sistem menampilkan pesan untuk mengisi <i>form</i> dengan lengkap.
<i>Post Condition</i>	Data akun sekolah berhasil di-ubah oleh Admin Diknas.

Tabel 4.25 Use Case Scenario Hapus Sekolah

Nama Use Case	Hapus Sekolah
Kode Kebutuhan Terkait	SRS-SP-F-22
Aktor	Admin Diknas

Tujuan	Admin Diknas dapat hapus data sekolah yang sudah ditambahkan dalam sistem.
<i>Pre Condition</i>	Admin Diknas sudah login.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Admin memilih halaman Daftar Sekolah. 2. Sistem menampilkan halaman Daftar Sekolah. 3. Admin menekan tombol “hapus”. 4. Sistem menampilkan pesan konfirmasi. 5. Admin menghapus data akun sekolah. 6. Sistem menghapus data sesuai akun yang ingin dihapus.
<i>Alternative Flow</i>	-
<i>Post Condition</i>	Data akun sekolah berhasil dihapus oleh Admin Diknas.

Tabel 4.26 Use Case Scenario Tambah Kepala Seksi

Nama Use Case	Tambah Kepala Seksi
Kode Kebutuhan Terkait	SRS-SP-F-23
Aktor	Admin Diknas
Tujuan	Admin Diknas dapat mendaftarkan akun Kepala Seksi untuk dapat menggunakan fungsi sistem.
<i>Pre Condition</i>	Admin Diknas berada pada halaman utama Admin Diknas.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Admin Diknas menekan halaman tambah kasi. 2. Sistem menampilkan <i>form</i> tambah kepala seksi berupa tingkatan kasi, username, email dan password, NIP, alamat, dan nomor hp. 3. Admin Diknas mengisi <i>form</i> dengan lengkap dan menekan tombol selesai. 4. Sistem menyimpan data yang telah diisi oleh Admin Diknas.
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Jika salah satu <i>input</i> tidak diisi sistem menampilkan pesan untuk mengisi <i>form</i> dengan lengkap.

<i>Post Condition</i>	Tambah Kepala Seksi oleh Admin Diknas berhasil dilakukan.
-----------------------	---

Tabel 4.27 Use Case Scenario Ubah Kepala Seksi

Nama Use Case	Ubah Kepala Seksi
Kode Kebutuhan Terkait	SRS-SP-F-24
Aktor	Admin Diknas
Tujuan	Admin Diknas dapat ubah data akun kasi yang sudah ditambahkan.
<i>Pre Condition</i>	Admin Diknas sudah login.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Admin memilih halaman Daftar Kepala Seksi. 2. Sistem menampilkan halaman Daftar Kepala Seksi. 3. Admin merubah data akun berupa tingkatan kasi, username, email dan password, NIP, alamat, dan nomor hp. 4. Sistem melakukan update data sesuai akun yang ingin dirubah.
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Jika salah satu <i>input</i> tidak diisi sistem menampilkan pesan untuk mengisi <i>form</i> dengan lengkap.
<i>Post Condition</i>	Data akun kepala seksi berhasil di ubah oleh Admin Diknas.

Tabel 4.28 Use Case Scenario Hapus Kepala Seksi

Nama Use Case	Hapus Kepala Seksi
Kode Kebutuhan Terkait	SRS-SP-F-25
Aktor	Admin Diknas
Tujuan	Admin Diknas dapat hapus data akun kepala seksi yang sudah ditambahkan dalam sistem.
<i>Pre Condition</i>	Admin Diknas sudah login.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Admin memilih halaman Daftar Kepala Seksi. 2. Sistem menampilkan halaman Daftar Kepala Seksi.

	<ol style="list-style-type: none"> 3. Admin menekan tombol “hapus”. 4. Sistem menampilkan pesan konfirmasi. 5. Admin menghapus data akun kasi. 6. Sistem menghapus data sesuai akun yang ingin dihapus.
<i>Alternative Flow</i>	-
<i>Post Condition</i>	Data akun kasi berhasil dihapus oleh Admin Diknas.

Tabel 4.29 Use Case Scenario Tambah Tim Verifikasi

Nama Use Case	Tambah Tim Verifikasi
Kode Kebutuhan Terkait	SRS-SP-F-26
Aktor	Admin Diknas
Tujuan	Admin Diknas dapat mendaftarkan akun Tim Verifikasi untuk dapat menggunakan fungsi sistem.
<i>Pre Condition</i>	Admin Diknas berada pada halaman utama Admin Diknas.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Admin Diknas menekan halaman tambah tim verifikasi. 2. Sistem menampilkan <i>form</i> tambah tim verifikasi berupa username, email, password, NIP, alamat, dan nomor hp 3. Admin Diknas mengisi <i>form</i> dengan lengkap dan menekan tombol selesai. 4. Sistem menyimpan data yang telah diisi oleh Admin Diknas.
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Jika salah satu <i>input</i> tidak diisi sistem menampilkan pesan untuk mengisi <i>form</i> dengan lengkap.
<i>Post Condition</i>	Tambah Tim Verifikasi oleh Admin Diknas berhasil dilakukan

Tabel 4.30 Use Case Scenario Ubah Tim Verifikasi

Nama Use Case	Ubah Tim Verifikasi
---------------	---------------------

Kode Kebutuhan Terkait	SRS-SP-F-27
Aktor	Admin Diknas
Tujuan	Admin Diknas dapat ubah data akun tim verifikasi yang sudah ditambahkan.
<i>Pre Condition</i>	Admin Diknas sudah login.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Admin memilih halaman Daftar Tim Verifikasi. 2. Sistem menampilkan halaman Daftar Tim Verifikasi. 3. Admin merubah data akun berupa username, email, password, NIP, alamat, dan nomor hp. 4. Sistem melakukan update data sesuai akun yang ingin dirubah.
<i>Alternative Flow</i>	<ol style="list-style-type: none"> 1. Jika salah satu <i>input</i> tidak diisi sistem menampilkan pesan untuk mengisi <i>form</i> dengan lengkap.
<i>Post Condition</i>	Data akun kasi berhasil di ubah oleh Admin Diknas.

Tabel 4.31 Use Case Scenario Hapus Tim Verifikasi

Nama Use Case	Hapus Tim Verifikasi
Kode Kebutuhan Terkait	SRS-SP-F-28
Aktor	Admin Diknas
Tujuan	Admin Diknas dapat hapus data akun tim verifikasi yang sudah ditambahkan dalam sistem.
<i>Pre Condition</i>	Admin Diknas sudah login.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Admin memilih halaman Daftar Tim Verifikasi. 2. Sistem menampilkan halaman Daftar Tim Verifikasi. 3. Admin menekan tombol “hapus”. 4. Sistem menampilkan pesan konfirmasi. 5. Admin menghapus data akun tim verifikasi. 6. Sistem menghapus data sesuai akun yang ingin dihapus.

<i>Alternative Flow</i>	-
<i>Post Condition</i>	Data akun tim verifikasi berhasil dihapus oleh Admin Diknas.

Tabel 4.32 Use Case Scenario Lihat Rekap Bantuan

Nama Use Case	Lihat Rekap Bantuan
Kode Kebutuhan Terkait	SRS-SP-F-29
Aktor	Kepala Seksi
Tujuan	Kepala Seksi dapat melihat rekap bantuan dari proposal.
<i>Pre Condition</i>	Kepala Seksi sudah login.
<i>Main flow</i>	<ol style="list-style-type: none"> 1. Kepala Seksi memilih halaman Daftar Proposal Selesai. 2. Sistem menampilkan halaman Daftar Proposal Selesai. 3. Kepala Seksi menekan tombol Rekap Bantuan. 4. Sistem menampilkan halaman Rekap Bantuan.
<i>Alternative Flow</i>	-
<i>Post Condition</i>	Halaman Rekap Bantuan dilihat oleh Kepala Seksi.

BAB 5 PERANCANGAN DAN IMPLEMENTASI

Pada bab ini membahas mengenai perancangan sistem pada Sistem Pengelolaan Pendistribusian Dana Bantuan Sekolah studi kasus Dinas Pendidikan dan Kebudayaan Kota Serang. Tahapan yang dilakukan peneliti terdapat alur kerja secara umum. Perancangan umum sistem menjelaskan proses kerja dari pengguna, basis data dan media yang digunakan untuk mengakses basis datanya. Tahap perancangan berikutnya adalah perancangan perangkat lunak, dalam tahap ini terdapat beberapa langkah yaitu perancangan arsitektur sistem, perancangan diagram arsitektur, perancangan basis data dan perancangan antarmuka sistem.

5.1 Perancangan Sistem

Setelah melakukan proses analisis kebutuhan, proses perancangan sistem dilakukan dengan memodelkan hasil dari analisis kebutuhan. Pada proses perancangan sistem pengelolaan dana bantuan sekolah terdapat empat bagian perancangan, yaitu perancangan arsitektur, perancangan komponen, perancangan data, dan perancangan antarmuka.

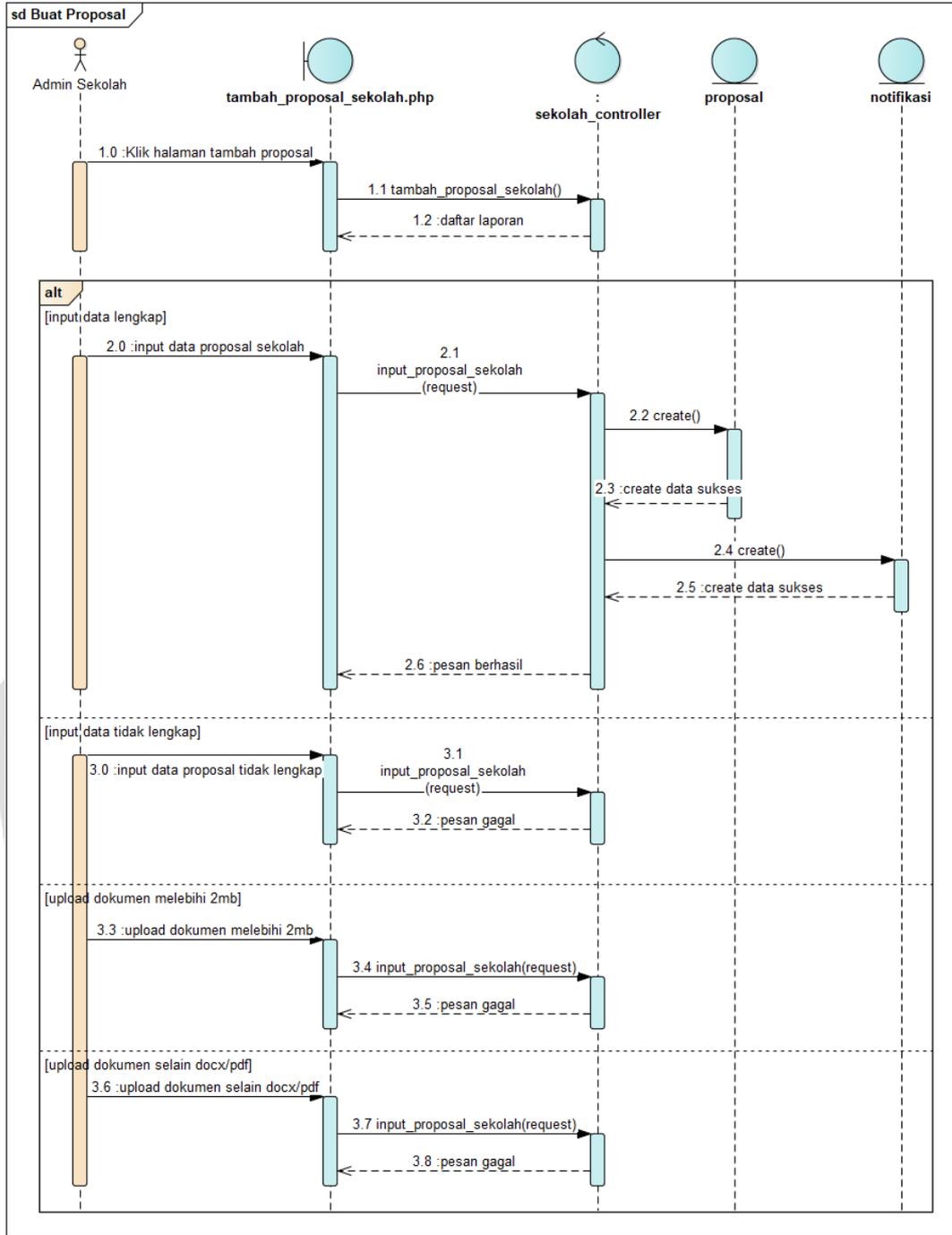
5.1.1 Perancangan Arsitektur

Sistem pengelolaan dana bantuan sekolah ini dibangun dengan menggunakan arsitektur MVC (*Model View Controller*) yang mana *model* sebagai kelas yang berfungsi dalam pengelolaan basis data, *view* merupakan kelas sebagai antarmuka dari sistem, dan *Controller* sebagai kelas yang menjembatani antara *model* dan *view*.

5.1.2 Sequence Diagram

5.1.2.1 Sequence Diagram Buat Proposal

Pada gambar 5.1 merupakan *sequence* diagram buat laporan, pada *sequence* diagram tersebut dijelaskan bagaimana aktor yaitu Admin Sekolah dapat membuat laporan didalam sistem. Terdapat pula objek – objek selain aktor yang berinteraksi pada *sequence* diagram yaitu sebuah *boundary* dengan nama `index.php` dan `tambah_proposal_sekolah.php`, sebuah *controller* dengan nama `sekolah_controller`, dan dengan sebuah *model* bernama `laporan`. Aktor dapat membuat laporan pada sistem dengan mengisi *form* yang ada dengan memanggil `view tambah_proposal_sekolah()`. Pengisian yang perlu dilakukan yaitu, judul laporan, latarbelakang laporan, gambar, dan file pdf. Kemudian aktor dapat menyimpannya dengan menekan tombol submit. Setelah menekan tombol submit, `view tambah_laporan_sekolah.php` memanggil `method input_laporan_sekolah()` pada `controller sekolah_controller`. Setelah semua proses selesai `sekolah_controller` akan return ke halaman sebelumnya.



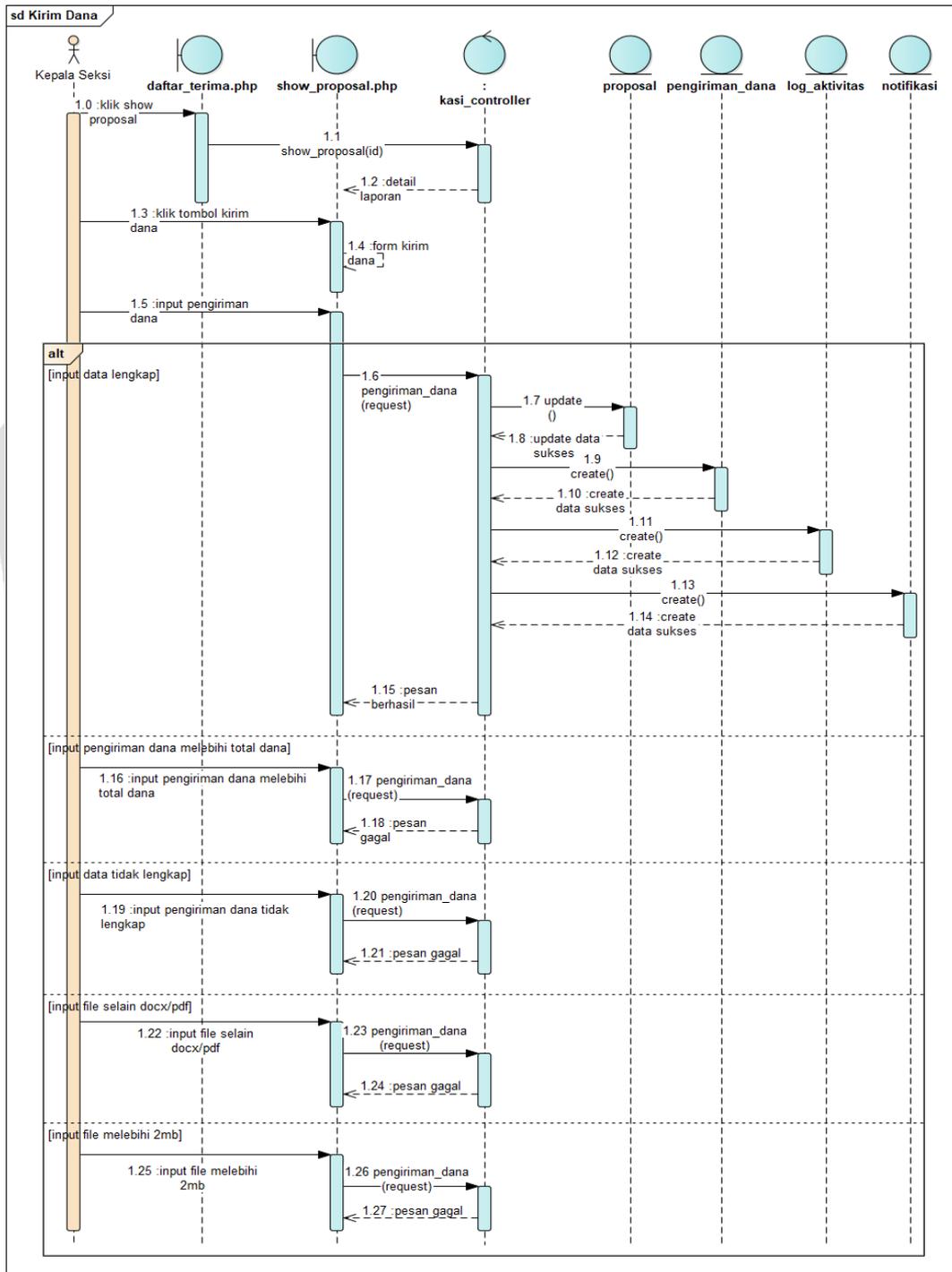
Gambar 5.1 Sequence Diagram Buat Proposal

5.1.2.2 Sequence Diagram Kirim Dana

Pada gambar 5.2 merupakan *sequence* diagram input status laporan, pada *sequence* diagram tersebut dijelaskan bagaimana aktor yaitu Kepala Seksi mengirim dana bantuan terhadap proposal yang sudah dibuat. Terdapat pula objek – objek selain aktor yang berinteraksi pada *sequence* diagram yaitu sebuah *boundary* dengan nama *daftar_terima.php* dan *show_proposal.php* sebuah *controller* dengan nama *kasi_controller* dan sebuah *model* yang bernama



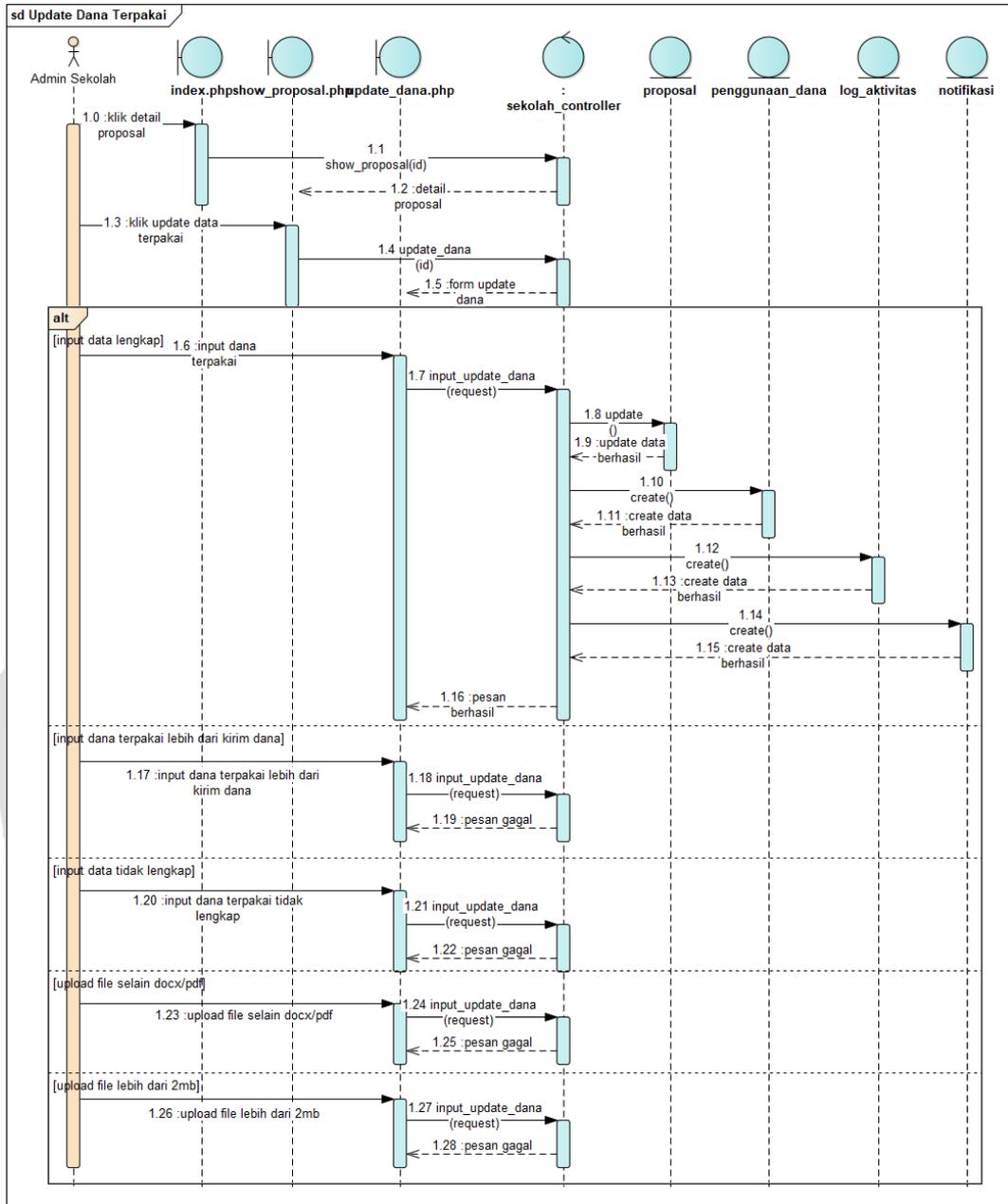
proposal, pengiriman_dana, notifikasi, rekap_data dan log_aktivitas. Kepala seksi dapat mengirimkan dana proposal dengan mengisi form yang sudah. Pengisian form yang perlu dilakukan yaitu jumlah dana yang akan dikirim dan foto bukti pengiriman dana lalu Kepala Seksi menekan tombol submit dan fungsi pengiriman_dana(request) dari controller kasi_controller. Setelah semua proses penyimpanan selesai, maka kasi_controller akan melakukan return ke halaman sebelumnya.



Gambar 5.2 Sequence Diagram Kirim Dana



5.1.2.3 Sequence Diagram Input Dana Terpakai

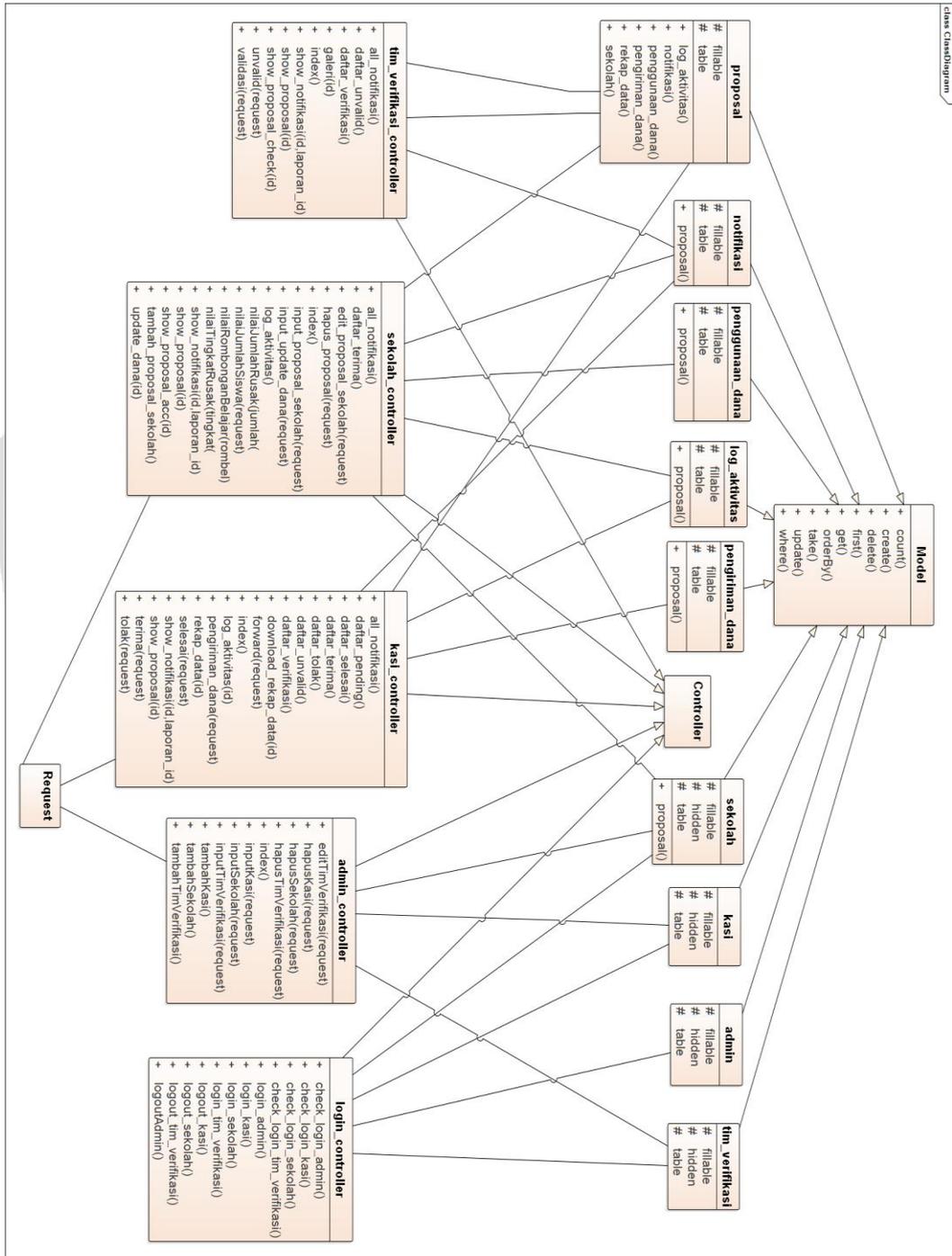


Gambar 5.3 Sequence Diagram Input Dana Terpakai

Pada gambar 5.3 merupakan *sequence* diagram input Dana Terpakai, pada *sequence* diagram tersebut dijelaskan bagaimana aktor yaitu Admin Sekolah dapat meng-inputkan dana bantuan sekolah yang sudah dipakai. Terdapat objek – objek selain aktor yang berinteraksi pada *sequence diagram* yaitu sebuah *boundary* dengan nama index.php, show_proposal.php dan update_dana.php, sebuah controller dengan nama sekolah_controller, dan sebuah model bernama proposal, penggunaan dana, log_aktivitas, dan notifikasi. Admin Sekolah dapat mengisi rincian dana yang sudah terpakai dengan mengisi form input dana terpakai. Untuk mengisi form Admin Sekolah harus berada dalam view update_dana.php mengisi

form. View update_dana.php memanggil method input_update_dana(request) untuk menyimpan data yang sudah diisi oleh Admin Sekolah. Setelah proses penyimpanan selesai, maka controller sekolah_controller return ke halaman sebelumnya.

5.1.3 Class Diagram



Gambar 5.4 Class Diagram

Pada gambar 5.4 diatas terdapat 2 jenis class, yaitu class yang berperan sebagai controller dan class yang berperan sebagai model. Class Controller merupakan general class untuk class-class controller turunannya, yaitu login_controller, kasi_controller, sekolah_controller, tim_verifikasi_controller. Sedangkan class Model merupakan general class untuk class-class model yaitu proposal, notifikasi, penggunaan_dana, log_aktivitas, pengiriman_dana, sekolah, kasi, admin, tim_verifikasi.

5.1.4 Perancangan Komponen

5.1.4.1 Perancangan Komponen Method "input_proposal_sekolah" class "sekolah_controller"

Nama class: sekolah_controller

Nama operasi: input_proposal_sekolah(request)

Algoritma:

```

1      validator = request
2      judul = required
3      deskripsi = required
4      dokumen = required doc,docx,pdf
5      jumlah_rusak = required
6      jumlah_siswa = required
7      rombongan_belajar = required
8      dana_diajukan = required
9      pengantar_dinas = required
10     surat_permohonan = required
11     inialisasi nilai survei=0
12     inialisasi nilai jmlh_rusak = nilaiJumlahRusak
13     inialisasi tngkt_rusak = nilaiTingkatRusak
14     inialisasi jmlh_siswa = nilaiJumlahSiswa
15     inialisasi rombongan_bljr = nilaiRombonganBelajar
16     menghitung      nilai      dari      survei      =
17     ((30*jmlh_rusak)+(40*tngkt_rusak)+(20*jmlh_siswa)+(10*romb
18     ongan_bljr))/100
19     foreach foto sebagai image
20         name = image
21         image dipindahkan dengan format nama foto
22         data = name
23     endfor
24     galeri = data
25     foto = data
26     dokumen = Request dokumen
27     namaDokumen = dokumen
28     data =
29         sekolah_id = sekolah id
30         judul = Request judul
31         deskripsi = request deskripsi
32         dokumen = namaDokumen
33         pengantar_dinas = namaPengantar
34         surat_permohonan = namaPermohonan
35         foto = foto
36         galeri = galeri
37         status = belum
38         status_kerja = 0

```

```

39     dana_diajukan = Request dana_diajukan
40     jumlah_rusak = Request jumlah_rusak
41     tingkat_rusak = Request tingkat_rusak
42     jumlah_siswa = Request jumlah_siswa
43     rombongan_belajar = Request rombongan_belajar
44     jenis_sekolah = Request jenis_sekolah
45     tingkat_sekolah = tingkat sekolah
46     rate => survei
47     memindahkan file proposal dan mengubah nama format
48     proposal
49     membuat data proposal dan menyimpan
50     mengurutkan proposal berdasarkan yang terbaru
51     dataNotif
52         proposal_id = last id,
53         dari = sekolah id
54         tujuan = kasi
55         deskripsi = buat_proposal
56         tingkat = tingkat sekolah
57         status = belum
58     membuat data notifikasi dan menyimpan data notifikasi
59     notif
60     pesan buat proposal berhasil dibuat
61     kembali kehalaman sebelumnya dengan pesan

```

5.1.4.2 Perancangan Komponen *Method* “pengiriman_dana” class “kasi_controller”

Nama class : kasi_controller

Nama operasi: pengiriman_dana(request)

Algoritma:

```

1     proposal = proposal id
2     dana_terkirim_1 = proposal dana_terkirim
3     dana_terkirim_2 = Request jumlah;
4     dana_total = dana_total
5     selisih = dana_total - dana_terkirim_1
6     if dana_terkirim_2 lebih dari selisih
7         maka ada pesan gagal untuk pengiriman
8         kembali ke halaman sebelumnya
9     else
10        dana_terkirim_update_
11    dana_terkirim_1+dana_terkirim_2
12        foto = Request foto
13        namaFoto = foto
14        data_pengiriman
15            proposal_id => Request id
16            jumlah => Request jumlah
17            foto => namaFoto
18        data_proposal
19            dana_terkirim => dana_terkirim_updated
20        dataNotif
21            proposal_id => Request->id
22            dari => kasi
23            tujuan => proposal->sekolah_id
24            deskripsi => kirim_dana
25            status => belum

```



26	deskripsi = jumlah dana terkirim
27	membuat data notifikasi dan menyimpan data notifikasi
28	membuat data pengiriman dana dan menyimpan data
29	pengiriman
30	update data proposal
31	memindahkan data proposal dengan format nama
32	mengurutkan data pengiriman dana berdasarkan yang
33	terbaru
34	dataLog =
35	proposal_id => Request id
36	deskripsi => kasi mengirim dana
37	dari => kasi
38	data_id => data id
39	membuat data log aktivitas dan menyimpan data log
40	aktivitas
41	notif
42	pesan pengiriman dana berhasil
43	kembali kehalaman sebelumnya dengan notifikasi

5.1.4.3 Perancangan Komponen *Method* “input_update_dana” class “sekolah_controller”

Nama class : sekolah_controller

Nama operasi : input_update_dana(request)

Algoritma :

1	validator dokumen persetujuan tidak boleh lebih dari 2mb
2	total =required
3	barang = Request barang
4	harga = Request harga
5	no = 1;
6	total = Request total
7	mengambil data proposal melalui id
8	terkirim = dana_terkirim
9	terpakai = dana_terpakai
10	selisih = terkirim - terpakai
11	total = request total
12	jika total kurang dari selisih
13	maka
14	notif
15	message = gagal, total penggunaan besar
16	dari dana terkirim kasi
17	kembali ke halaman sebelumnya dengan
18	notif
19	else
20	sekarang = tanggal dana terpakai
21	inisialisasi nama file dan format file
22	menyimpan data pdf dengan nama_file
23	dana_terpakai_updated_ = terpakai+ total
24	dana_terpakai_updated =
25	dana_terpakai_updated_
26	dokumen_persetujuan = Request
27	dokumen_persetujuan
28	namaDokumen = dokumen_persetujuan
29	data_proposal
30	dana_terpakai = dana_terpakai_updated

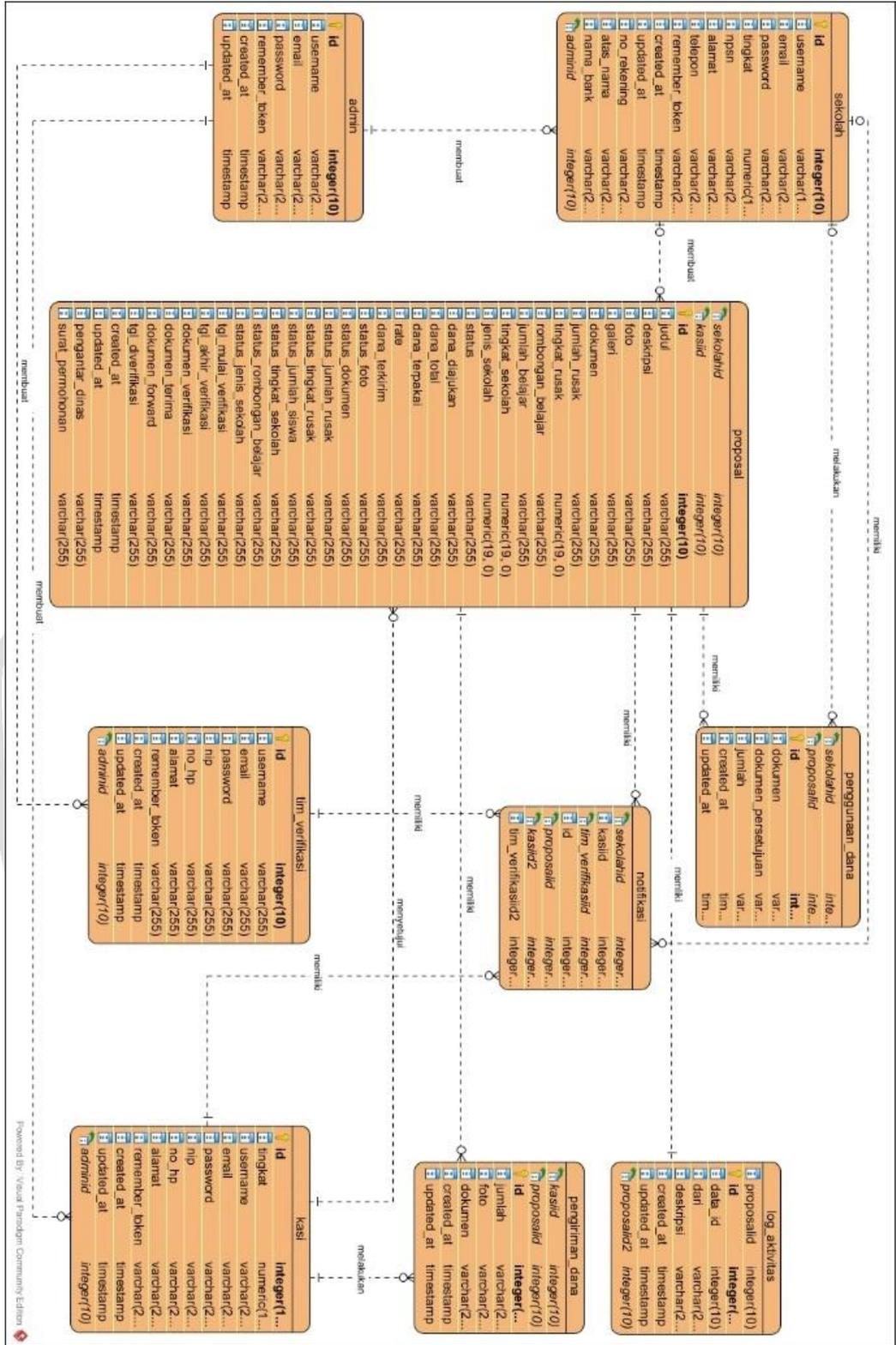
```

31     data_penggunaan_dana
32     proposal_id => Request id
33     jumlah => total
34     dokumen => nama_file
35     dokumen_persetujuan = namaDokumen
36     dataNotif
37     proposal_id => proposal id,
38     dari = sekolah id,
39     tujuan = kasi
40     deskripsi = update_dana
41     tingkat = tingkat sekolah,
42     status = belum
43     deskripsi = sekolah mengupdate pemakaian
44     dana bantuan sebesar Rp total
45     memindahkan dokumen_persetujuan ke folder
46     yang ditentukan
47     membuat data notifikasi dan menyimpan data
48     notifikasi
49     meng-update data proposal bedasarkan id
50     membuat data penggunan dana dan menyimpan
51     data penggunaan dana
52     mengurutkan data penggunaan dana
53     dataLog
54     proposal_id = Request id
55     deskripsi = update penggunaan dana oleh
56     sekolah
57     dari => sekolah
58     data_id => data id
59     membuat data log aktivitas dan menyimpan
60     data log aktivitas
61     notif
62     pesan menyimpan data sukses
63     kembali ke halaman detail proposal

```

5.1.5 Perancangan Data

Hasil dari analisis data dimodelkan dalam bentuk *Conceptual Data Model* (CDM) dari Sistem Pengelolaan Pendistribusian Dana Bantuan Sekolah. Pada gambar 5.5 menunjukkan detail dari entitas relasi user dengan menyertakan atribut yang ada pada entitas user.

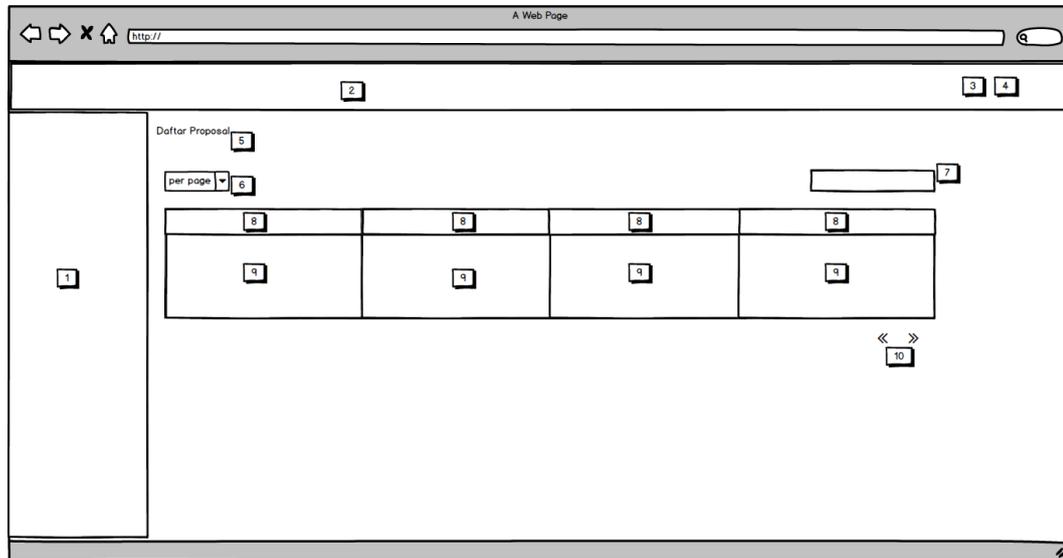


Gambar 5.5 Conceptual Data Model



5.1.6 Perancangan Antarmuka

5.1.6.1 Perancangan Antarmuka Daftar Proposal



Gambar 5.6 Perancangan antarmuka Daftar Proposal

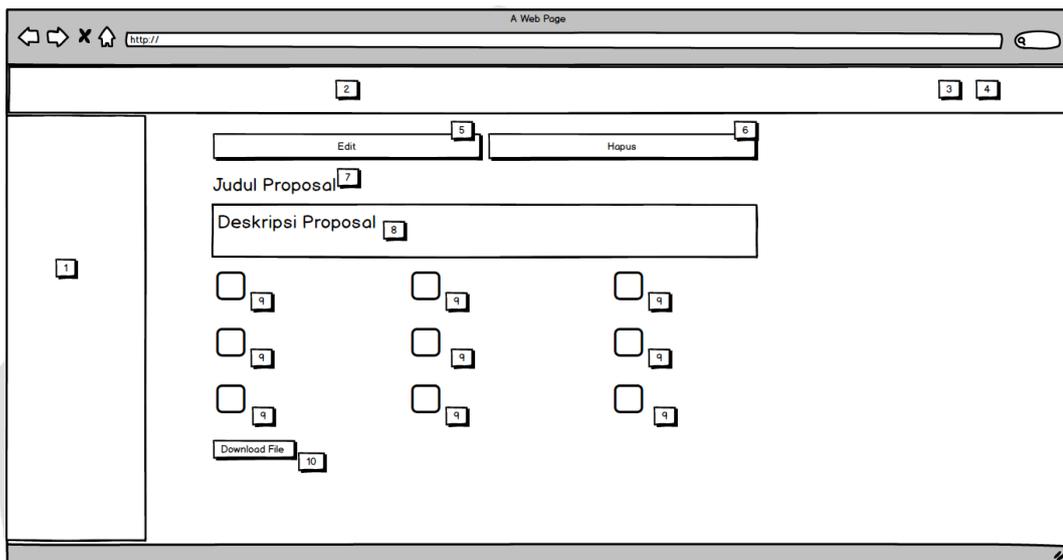
Pada Gambar 5.6 merupakan gambaran perancangan antarmuka dari halaman daftar proposal dan penjelasan terdapat pada Tabel 5.1

Tabel 5.1 Penjelasan antarmuka daftar proposal

No	Nama Objek	Tipe	Keterangan
1	Side Bar	Navbar	Menampilkan daftar menu yang bisa pilih oleh user.
2	Header	Navbar	Sebagai header untuk informasi user dan notifikasi.
3	Notifikasi	Dropdown	Menampilkan notifikasi pada sistem.
4	Informasi Pengguna	Image and text	Menampilkan halaman Daftar Laporan.
5	Judul Halaman	<i>Label</i>	Menampilkan halaman Daftar Proposal.
6	Jumlah Urutan Data	Combo Box	Untuk menampilkan data proposal sesuai data yang diinginkan.
7	Search	<i>Input</i>	Untuk mencari data proposal.

8	Judul tabel	<i>Text</i>	Menampilkan judul dari tabel.
9	Data Proposal	<i>Text</i>	Untuk menampilkan data dari proposal.
10	Pagination	<i>Button</i>	Untuk menampilkan halaman berikutnya.

5.1.6.2 Perancangan Antarmuka *Detail Proposal*



Gambar 5.7 Perancangan antarmuka Detail Proposal

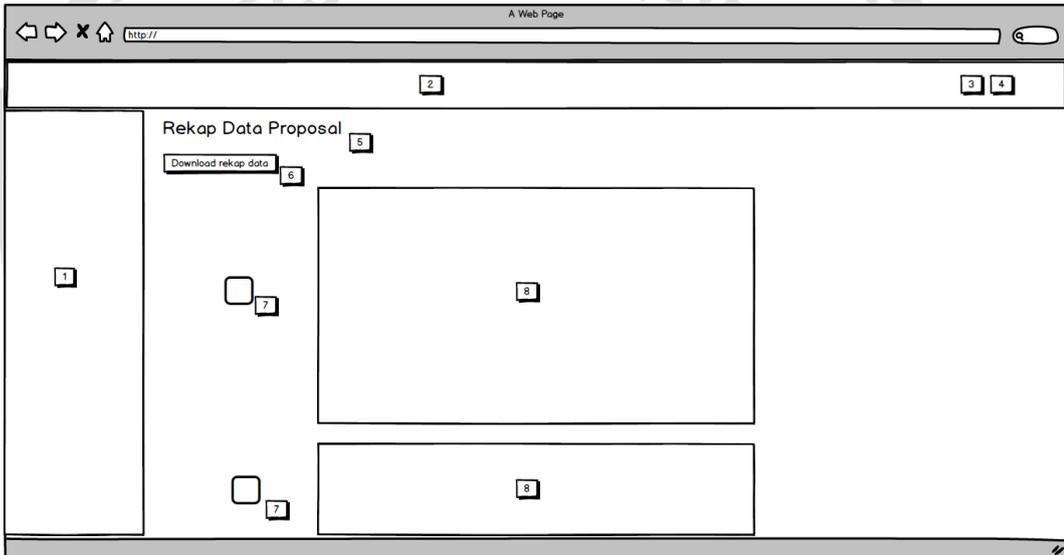
Pada Gambar 5.7 merupakan gambaran perancangan antarmuka dari halaman buat laporan dan penjelasan terdapat pada Tabel 5.2

Tabel 5.2 Penjelasan antarmuka detail proposal

No	Nama Objek	Tipe	Keterangan
1	Side Bar	Navbar	Menampilkan daftar menu yang bisa pilih oleh user.
2	Header	Navbar	Sebagai header untuk informasi user dan notifikasi.
3	Notifikasi	Dropdown	Menampilkan notifikasi pada sistem.
4	Informasi Pengguna	Image and text	Menampilkan halaman Daftar Laporan.
5	Edit	<i>Button</i>	Untuk mengubah data proposal.

6	Delete	Button	Untuk menghapus data proposal.
7	Judul Proposal	Label	Menampilkan judul dari proposal.
8	Deskripsi Proposal	Text	Untuk menampilkan deskripsi dari proposal.
9	Data Proposal	Icon dan text	Menampilkan data dari proposal yang sudah di inputkan.
10	Download	Button	Untuk meng-unduh berkas proposal yang sudah dibuat.

5.1.6.3 Perancangan Antarmuka Rekap Proposal



Gambar 5.8 Perancangan antarmuka rekap proposal

Pada Gambar 5.8 merupakan gambaran perancangan antarmuka dari halaman buat laporan dan penjelasan terdapat pada Tabel 5.3

Tabel 5.3 Penjelasan antarmuka rekap proposal

No	Nama Objek	Tipe	Keterangan
1	Side Bar	Navbar	Menampilkan daftar menu yang bisa pilih oleh user.
2	Header	Navbar	Sebagai header untuk informasi user dan notifikasi.



3	Notifikasi	Dropdown	Menampilkan notifikasi pada sistem.
4	Informasi Pengguna	Image and text	Untuk menampilkan menu logout, dan menampilkan nama pengguna.
5	Nama Halaman	<i>Label</i>	Menampilkan halaman Rekap Data Proposal.
6	Download	Button	Untuk meng-unduh berkas data rekap proposal.
7	Tanggal	<i>Text</i>	Menampilkan tanggal proses bantuan proposal.
8	Data Rekap Proposal	<i>Text, image, dan button</i>	Menampilkan data rekap proposal.

5.2 Implementasi Sistem

Proses implementasi dilakukan setelah selesai proses perancangan. Implementasi dilakukan berdasarkan hasil yang diperoleh dari tahap analisis kebutuhan dan tahap perancangan. Setiap kebutuhan yang telah digali dan didapatkan akan diimplementasikan pada sistem. Implementasi harus memuat struktur dan kode program berdasarkan kelas-kelas dan algoritma yang sudah dibuat pada bagian perancangan. Pada bagian implementasi ini akan dijelaskan mengenai spesifikasi sistem, implementasi kode program, implementasi data, dan implementasi antarmuka.

5.2.1 Spesifikasi Sistem

Pada bagian ini akan dijelaskan mengenai spesifikasi yang digunakan untuk membangun sistem penentuan prioritas kebutuhan perangkat lunak dengan metode MVC. Spesifikasi yang akan dijelaskan adalah spesifikasi perangkat keras, perangkat lunak, sistem operasi, dan arsitektur sistem.

5.2.1.1 Spesifikasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan disajikan dalam tabel 5.

Tabel 5.4 Spesifikasi Perangkat Keras

Nama Komponen	Spesifikasi
Laptop	<i>Processor Intel® Core™ i7-6700HQ</i> <i>Memory 8 GB</i> <i>NVIDIA GeForce 950M</i>

5.2.1.2 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak yang digunakan disajikan dalam tabel 5.5

Tabel 5.5 Spesifikasi Perangkat Lunak

Nama Komponen	Spesifikasi
Editor Perancangan	Draw.io, Visual Paradigm <i>Community Edition</i> , Balsamiq Cloud
Editor Pemrograman	Visual Studio Code
Framework dan Bahasa Pemrograman	Laravel, Bootstrap, PHP, XML, HTML, CSS, Javascript
DBMS	MySQL
Browser	Mozilla Firefox

5.2.1.3 Spesifikasi Sistem Operasi

Tabel 5.6 Spesifikasi Sistem Operasi

Nama Komponen	Spesifikasi
Sistem Operasi	Windows 10 Pro 64-bit (10.0, Build 17134)

5.2.2 Implementasi Kode Program

Implementasi kode program didasarkan pada perancangan komponen yang telah dilakukan sebelumnya. Pada bagian ini akan dirubah algoritma-algoritma yang sudah dibuat dalam bentuk pseudocode pada bagian perancangan komponen bentuk bahasa pemrograman. Algoritma yang akan dituliskan implementasinya yaitu membuat laporan, tambah sekolah, input survey Implementasi menggunakan bahasa pemrograman PHP dengan framework Laravel 5.5.

5.2.2.1 Implementasi Komponen *Method* “input_proposal_sekolah” class “sekolah_controller”

Nama class: sekolah_controller

Nama operasi: input_proposal_sekolah(request)

Algoritma:

1	<code>\$validator = \$this->validate(\$Request, [</code>
2	<code> 'judul' => 'required',</code>
3	<code> 'deskripsi' => 'required max:220',</code>
4	<code> 'dokumen' =></code>
5	<code>'required mimes:doc,docx,pdf file max:2000',</code>
6	<code> 'jumlah_rusak' => 'required',</code>
7	<code> 'jumlah_siswa' => 'required',</code>

```
8         'rombongan_belajar' => 'required',
9         'dana_diajukan' => 'required',
10        'pengantar_dinas' =>
11 'required|mimes:doc,docx,pdf|file|max:2000',
12        'surat_permohonan' =>
13 'required|mimes:doc,docx,pdf|file|max:2000',
14    ];
15    $survei = 0 ;
16    $jmlh_rusak = $this->nilaiJumlahRusak($Request-
17 >jumlah_rusak);
18
19    $tngkt_rusak = $this->nilaiTingkatRusak($Request-
20 >tingkat_rusak);
21
22    $jmlh_siswa = $this->nilaiJumlahSiswa($Request-
23 >jumlah_siswa);
24
25    $rombongan_bljr = $this-
26 >nilaiRombonganBelajar($Request->rombongan_belajar);
27    $survei =
28 ((30*$jmlh_rusak)+(40*$tngkt_rusak)+(20*$jmlh_siswa)+(10*$
29 rombongan_bljr))/100);
30
31    if($Request->hasfile('foto'))
32    {
33
34        foreach($Request->file('foto') as $image)
35        {
36            $name=$image->getClientOriginalName();
37            $image->move('proposal/foto', $name);
38            $data[] = $name;
39        }
40    }
41    $galeri=json_encode($data);
42    $foto = $data[0];
43    $dokumen = $Request->dokumen;
44    $namaDokumen = $dokumen->getClientOriginalName();
45    $pengantar_dinas = $Request->pengantar_dinas;
46    $namaPengantar = $pengantar_dinas-
47 >getClientOriginalName();
48    $surat_permohonan = $Request->surat_permohonan;
49    $namaPermohonan = $surat_permohonan-
50 >getClientOriginalName();
51    $data = [
52        'sekolah_id' => auth()->guard('sekolah')->id(),
53        'judul' => $Request->judul,
54        'deskripsi' => $Request->deskripsi,
55        'dokumen' => $namaDokumen,
56        'pengantar_dinas' => $namaPengantar,
57        'surat_permohonan' => $namaPermohonan,
58        'foto' => $foto,
59        'galeri' => $galeri,
60        'status' => 'belum',
61        'status_kerja' => '0',
62        'dana_diajukan' => $Request->dana_diajukan,
63        'jumlah_rusak' => $Request->jumlah_rusak,
64        'tingkat_rusak' => $Request->tingkat_rusak,
65        'jumlah_siswa' => $Request->jumlah_siswa,
66
```

```

67         'rombongan_belajar' => $Request->rombongan_belajar,
68     >rombongan_belajar,
69         'jenis_sekolah' => $Request->jenis_sekolah,
70         'tingkat_sekolah' => auth()->guard('sekolah')->user()->tingkat,
71     >user()->tingkat,
72         'rate' => $survei
73     ];
74     $dokumen->move('proposal/dokumen', $namaDokumen);
75     $pengantar_dinas->move('proposal/pengantar_dinas', $namaPengantar);
76     $surat_permohonan->move('proposal/surat_permohonan', $namaPermohonan);
77     $proposal = proposal::create($data);
78     $last = proposal::orderBy('id', 'DESC')->take(1)->first();
79     $dataNotif = [
80         'proposal_id' => $last->id,
81         'dari' => auth()->guard('sekolah')->id(),
82         'tujuan' => 'kasi',
83         'deskripsi' => 'buat_proposal',
84         'tingkat' => auth()->guard('sekolah')->user()->tingkat,
85         'status' => 'belum'
86     ];
87     notifikasi::create($dataNotif);
88     $notif = [
89         'message' => 'Buat Proposal Berhasil',
90         'alert-type' => 'success'
91     ];
92     return back()->with($notif);

```

5.2.2.2 Implementasi Komponen Method “pengiriman_dana” class “kasi_controller”

Nama class : kasi_controller

Nama operasi : pengiriman_dana(request)

Algoritma :

```

1     $proposal = proposal::where('id', $Request->id)->first();
2     $dana_terkirim_1 = preg_replace('/[.]/', '', $proposal->dana_terkirim);
3     $dana_terkirim_2 = preg_replace('/[.]/', '', $Request->jumlah);
4     $dana_total = preg_replace('/[.]/', '', $proposal->dana_total);
5
6     $selisih = $dana_total - $dana_terkirim_1;
7
8     if($dana_terkirim_2 > $selisih){
9         $notif = [
10            'message' => 'gagal, pengiriman dana besar dari total bantuan',
11            'alert-type' => 'error'
12        ];
13        return back()->with($notif);

```



```
19         }else{
20             $dana_terkirim_updated_ =
21 $dana_terkirim_1+$dana_terkirim_2;
22             $dana_terkirim_updated =
23 number_format($dana_terkirim_updated_,0,',','.');
24
25             $foto = $Request->foto;
26             $namaFoto = $foto->getClientOriginalName();
27
28             $dokumen = $Request->dokumen;
29             $namaDokumen = $dokumen-
30 >getClientOriginalName();
31
32
33             $data_pengiriman = [
34                 'proposal_id' => $Request->id,
35                 'jumlah' => $Request->jumlah,
36                 'foto' => $namaFoto,
37                 'dokumen' => $namaDokumen,
38             ];
39
40             $data_proposal = [
41                 'dana_terkirim' => $dana_terkirim_updated
42             ];
43             $dataNotif = [
44                 'proposal_id' => $Request->id,
45                 'dari' => 'kasi',
46                 'tujuan' => $proposal->sekolah_id,
47                 'deskripsi' => 'kirim_dana',
48                 'status' => 'belum'
49             ];
50
51             $deskripsi = "Kasi mengirim dana sebanyak Rp.
52 $Request->jumlah ,-";
53
54             notifikasi::create($dataNotif);
55             pengiriman_dana::create($data_pengiriman);
56             proposal::where('id',$Request->id)-
57 >update($data_proposal);
58             $foto->move('proposal/kirim_dana',$namaFoto);
59             $dokumen-
60 >move('proposal/dokumen_kirim_dana',$namaDokumen);
61             $data =
62 pengiriman_dana::orderBy('id','desc')->first();
63
64             $dataLog = [
65                 'proposal_id' => $Request->id,
66                 'deskripsi' => 'kasi mengirim dana',
67                 'dari' => 'kasi',
68                 'data_id' => $data->id
69             ];
70             log_aktivitas::create($dataLog);
71
72             $notif = [
73                 'message' => 'Update Data Success',
74                 'alert-type' => 'success'
75             ];
76
77
78
```

79	<pre> return redirect(url('kasi/show_proposal/'.\$Request->id))- >with(\$notif); </pre>
----	---

5.2.2.3 Implementasi Komponen *Method* “update_dana_terpakai” class “sekolah_controller”

Nama class : sekolah_controller

Nama operasi : update_dana_terpakai(request)

Algoritma :

1	<code>\$validator = \$this->validate(\$Request, [</code>
2	<code> 'dokumen_persetujuan' =></code>
3	<code> 'required file max:2000',</code>
4	<code> 'total' => 'required'</code>
5	<code>]);</code>
6	<code>\$barang = \$Request->barang;</code>
7	<code>\$harga = \$Request->harga;</code>
8	<code>\$no = 1;</code>
9	<code>\$total = \$Request->total;</code>
10	<code>\$proposal = proposal::where('id',\$Request->id)-</code>
11	<code>>first();</code>
12	<code>\$terkirim = preg_replace('/[.]/', '', \$proposal-</code>
13	<code>>dana_terkirim);</code>
14	<code>\$terpakai = preg_replace('/[.]/', '', \$proposal-</code>
15	<code>>dana_terpakai);</code>
16	<code>\$selisih = \$terkirim-\$terpakai;</code>
17	<code>\$total = preg_replace('/[.]/', '', \$Request-</code>
18	<code>>total);</code>
19	
20	<code> if(\$total>\$selisih){</code>
21	<code> \$notif = [</code>
22	<code> 'message' => 'gagal, total penggunaan besar</code>
23	<code>dari dana terkirim kasi',</code>
24	<code> 'alert-type' => 'error'</code>
25	<code>];</code>
26	<code> return back()->with(\$notif);</code>
27	<code> }else{</code>
28	<code> \$sekarang = Carbon::now()->format('d-m-y h-m-</code>
29	<code>s');</code>
30	<code> \$nama_file = 'invoice_'. \$sekarang.'.pdf';</code>
31	
32	<code> \$pdf = PDF::loadView('master',</code>
33	<code>compact('barang','harga','no','total'));</code>
34	<code>\$pdf->save('penggunaan_dana/'.\$nama_file);</code>
35	
36	<code> \$dana_terpakai_updated_ = \$terpakai+\$total;</code>
37	<code> \$dana_terpakai_updated =</code>
38	<code>number_format(\$dana_terpakai_updated_,0,',','.');</code>
39	
40	<code> \$dokumen_persetujuan = \$Request-</code>
41	<code>>dokumen_persetujuan;</code>
42	<code> \$namaDokumen = \$dokumen_persetujuan-</code>
43	<code>>getClientOriginalName();</code>
44	<code> \$data_proposal =[</code>
45	<code> 'dana_terpakai' => \$dana_terpakai_updated</code>
46	<code>];</code>

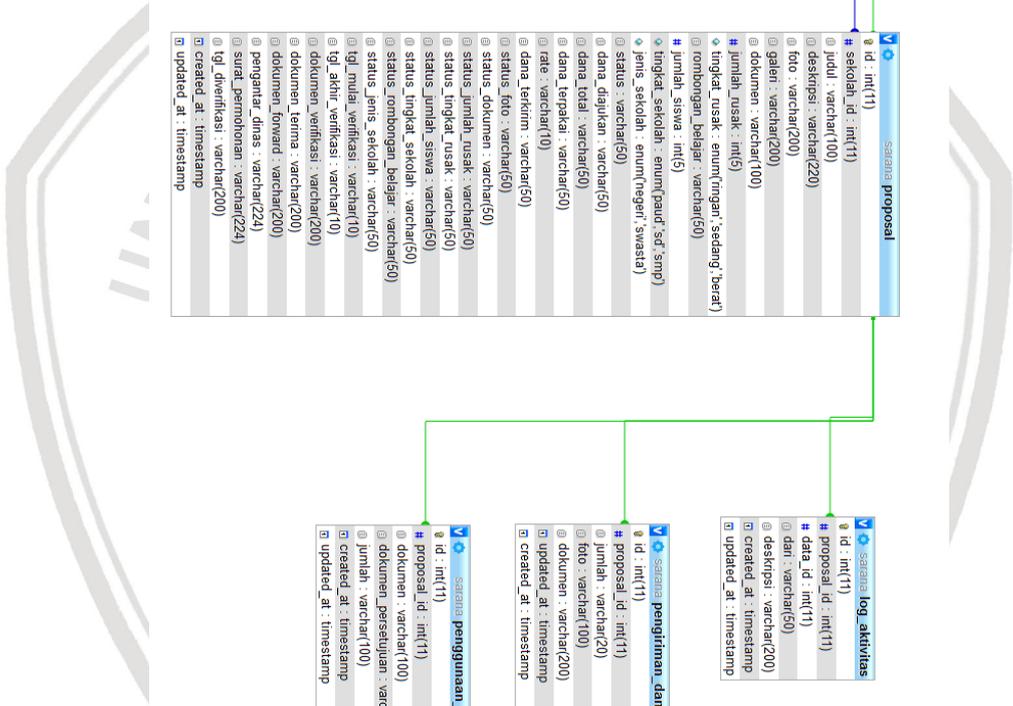
```

47     $data_penggunaan_dana = [
48         'proposal_id' => $Request->id,
49         'jumlah' => $total,
50         'dokumen' => $nama_file,
51         'dokumen_persetujuan' => $namaDokumen
52     ];
53
54     $dataNotif = [
55         'proposal_id' => $proposal->id,
56         'dari' => auth()->guard('sekolah')->id(),
57         'tujuan' => 'kasi',
58         'deskripsi' => 'update_dana',
59         'tingkat' => auth()->guard('sekolah')-
60 >user()->tingkat,
61         'status' => 'belum'
62     ];
63
64     $deskripsi = "sekolah mengupdate pemakaian dana
65 bantuan sebesar Rp $total ,-";
66
67     $dokumen_persetujuan-
68 >move('proposal/dokumen_penggunaan_dana',$namaDokumen);
69     notifikasi::create($dataNotif);
70     proposal::where('id',$Request->id)-
71 >update($data_proposal);
72     penggunaan_dana::create($data_penggunaan_dana);
73     $data = penggunaan_dana::orderBy('id','desc')-
74 >first();
75     $dataLog = [
76         'proposal_id' => $Request->id,
77         'deskripsi' => 'update penggunaan dana oleh
78 sekolah',
79         'dari' => 'sekolah',
80         'data_id' => $data->id
81     ];
82     log_aktivitas::create($dataLog);
83
84     $notif = [
85         'message' => 'Saved Data Success',
86         'alert-type' => 'success'
87     ];
88     return
89     redirect(url('sekolah/show_proposal_acc/'.$Request->id))-
90 >with($notif);

```

5.2.3 Implementasi Data

Implementasi data dibuat berdasarkan perancangan data yang telah dibuat pada *Conceptual Data Model* (CDM). Implementasi data disajikan dalam diagram *Physical Data Model* (PDM) pada gambar 5.9 yang menggambarkan semua tabel dalam basis data dalam sistem berikut dengan kolom dan tipe data untuk masing-masing kolomnya.



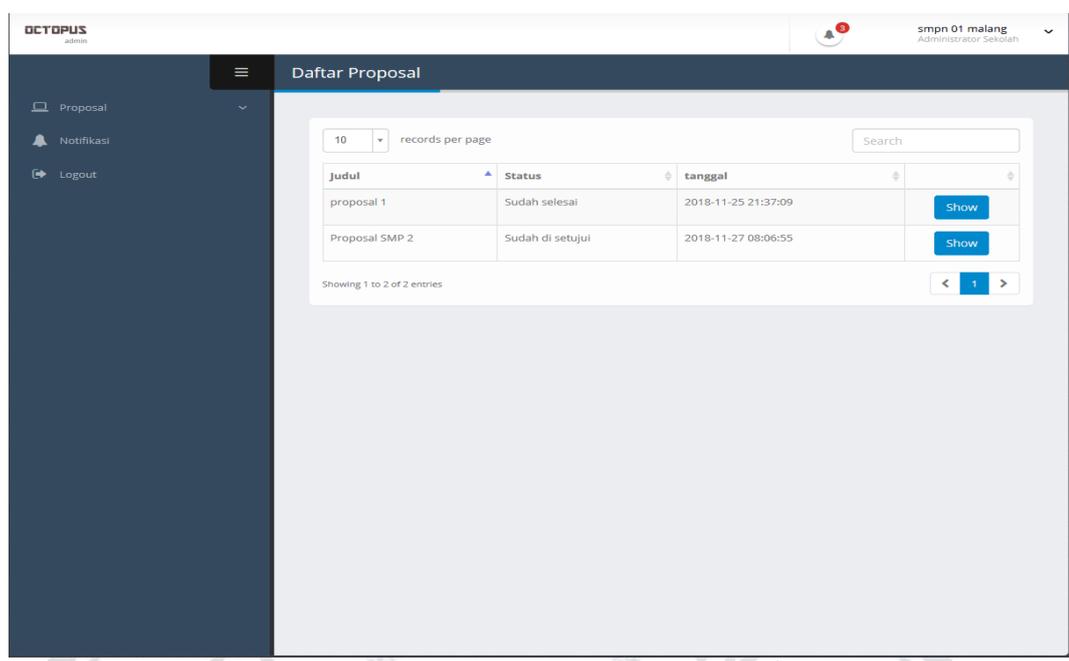
Gambar 5.9 Implementasi Database

5.2.4 Implementasi Antarmuka

Pada bagian ini akan dipaparkan antarmuka hasil implementasi antarmuka sistem berdasarkan rancangan yang telah dibuat pada bagian perancangan antarmuka.

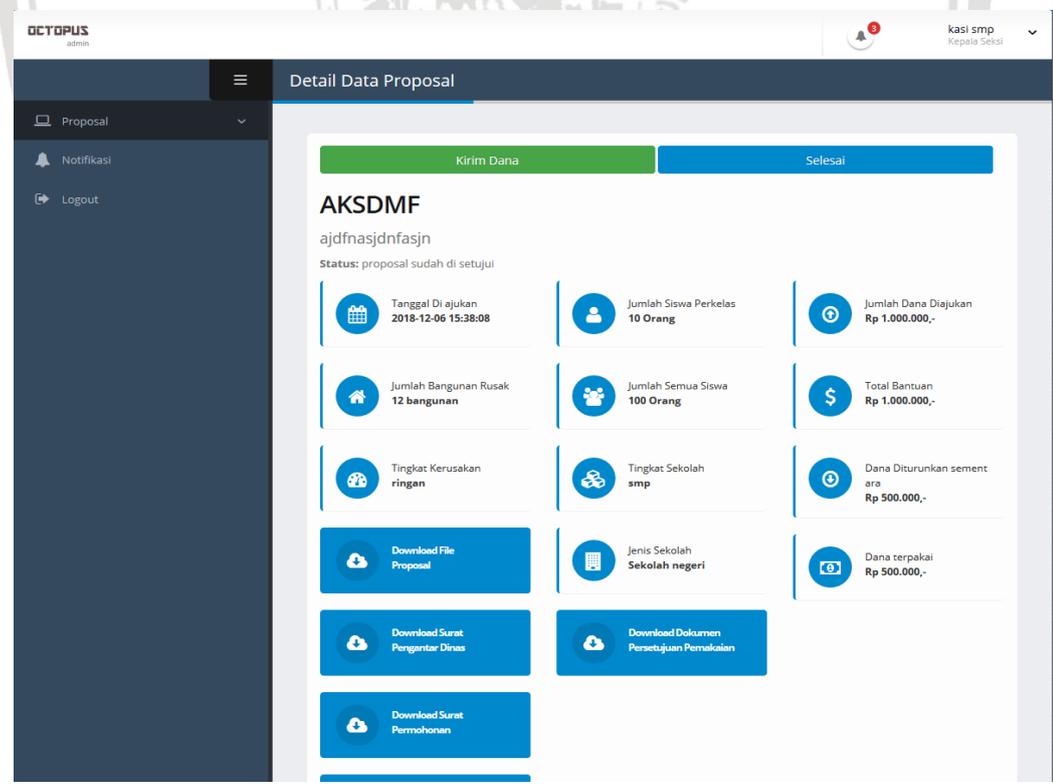


5.2.4.1 Implementasi Antarmuka Daftar Proposal



Gambar 5.10 Implementasi antarmuka daftar proposal
Pada gambar 5.10 merupakan tampilan antarmuka implementasi Daftar Proposal.

5.2.4.2 Implementasi Antarmuka Detail Proposal

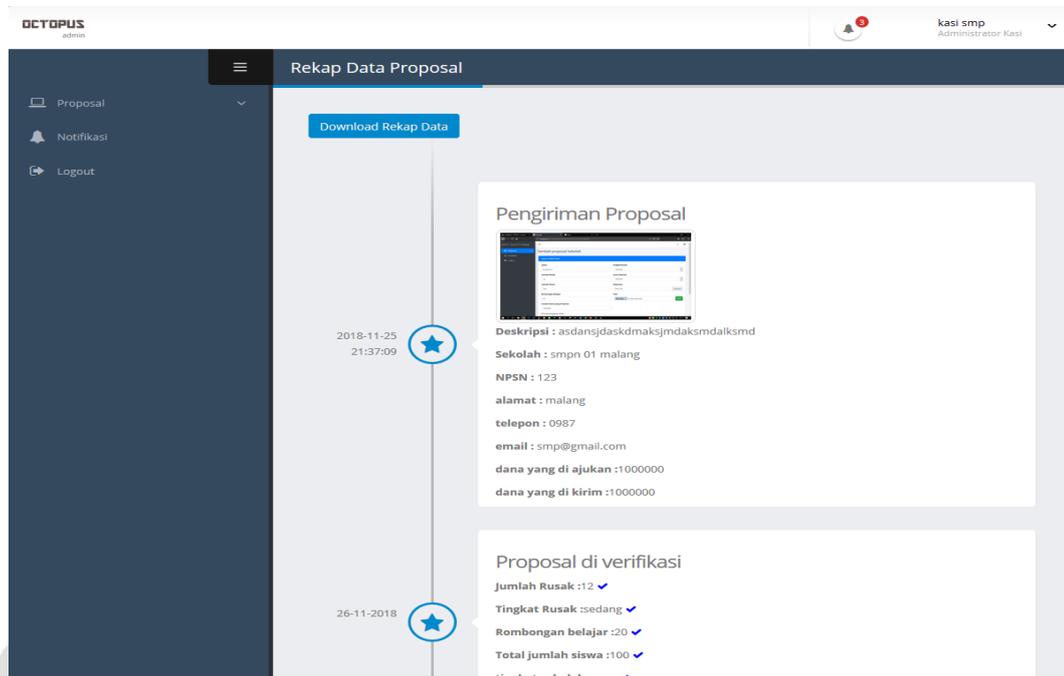


Gambar 5.11 Implementasi antarmuka Detail Proposal



Pada gambar 5.11 merupakan tampilan antarmuka implementasi Detail Proposal.

5.2.4.3 Implementasi Antarmuka Rekap Data



Gambar 5.12 Implementasi antarmuka Rekap Data

Pada gambar 5.12 merupakan tampilan antarmuka implementasi Rekap Data

BAB 6 PENGUJIAN

Tahap pengujian dilakukan setelah sudah melakukan implementasi sistem. Pengujian bertujuan untuk memeriksa apakah implementasi sudah sesuai dengan analisis kebutuhan dan perancangan sistem atau tidak. Tahap pengujian yang akan dilakukan adalah pengujian unit, pengujian integrasi, dan pengujian validasi.

6.1 Pengujian Unit

Pengujian unit ini berkonsentrasi pada setiap unit (misalnya: komponen, klas, atau objek) dari perangkat lunak yang diterapkan. Pada pengujian unit ini menggunakan metode *whitebox testing* dengan teknik pengujian *basis path*. Metode ini memungkinkan perancang *test case* mendapatkan ukuran kompleksitas logika dari perancangan prosedural dan menggunakan ukuran ini sebagai petunjuk untuk mendefinisikan basis set dari jalur pengerjaan. *Test case* didapat digunakan untuk mengerjakan basis set yang menjamin pengerjaan setiap perintah minimal satu kali selama uji coba. Langkah-langkah dalam pembuatan *test case* pada *basis path testing* adalah sebagai berikut:

1. Pembuatan diagram alir dari perancangan prosedural atau *source code*
2. Menentukan *cyclomatic complexity* dari diagram alir
3. Menentukan *independent path* dari diagram alir

6.1.1 Pengujian Unit *Method* “input_proposal_sekolah” class “sekolah_controller”

1. *Pseudocode*

```

1      validator = request
1      judul = required
1      deskripsi = required
1      dokumen = required doc, docx, pdf
1      jumlah_rusak = required
1      jumlah_siswa = required
1      rombongan_belajar = required
1      dana_diajukan = required
1      pengantar_dinas = required
1      surat_permohonan = required
1      inisialisasi nilai survei=0
1      inisialisasi nilai jmlh_rusak = nilaiJumlahRusak
1      inisialisasi tngkt_rusak = nilaiTingkatRusak
1      inisialisasi jmlh_siswa = nilaiJumlahSiswa
1      inisialisasi rombongan_bljr = nilaiRombonganBelajar
1      menghitung      nilai      dari      survei      =
1      (((30*jmlh_rusak)+(40*tngkt_rusak)+(20*jmlh_siswa)+(10*romb
1      ongan_bljr))/100
2      foreach foto sebagai image
3      name = image
3      image dipindahkan dengan format nama foto
3      data = name
4      endfor
5      galeri = data
5      foto = data

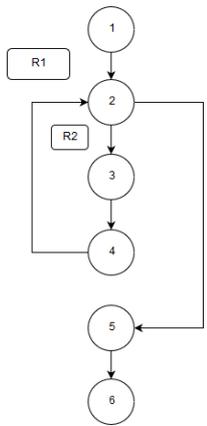
```

```

5     dokumen = Request dokumen
5     namaDokumen = dokumen
5     data =
5         sekolah_id = sekolah id
5         judul = Request judul
5         deskripsi = request deskripsi
5         dokumen = namaDokumen
5         pengantar_dinas = namaPengantar
5         surat_permohonan = namaPermohonan
5         foto = foto
5         galeri = galeri
5         status = belum
5         status_kerja = 0
5         dana_diajukan = Request dana_diajukan
5         jumlah_rusak = Request jumlah_rusak
5         tingkat_rusak = Request tingkat_rusak
5         jumlah_siswa = Request jumlah_siswa
5         rombongan_belajar = Request rombongan_belajar
5         jenis_sekolah = Request jenis_sekolah
5         tingkat_sekolah = tingkat sekolah
5         rate => survei
5     memindahkan file proposal dan mengubah nama format
5     proposal
5     membuat data proposal dan menyimpan
5     mengurutkan proposal berdasarkan yang terbaru
5     dataNotif
5         proposal_id = last id,
5         dari = sekolah id
5         tujuan = kasi
5         deskripsi = buat_proposal
5         tingkat = tingkat sekolah
5         status = belum
5     membuat data notifikasi dan menyimpan data notifikasi
5     notif
5         pesan buat proposal berhasil dibuat
6     kembali kehalaman sebelumnya dengan pesan
    
```

2. Basis Path Testing

1.1 Flow Graph



Gambar 6.1 Flow graph method "input_proposal_sekolah" class "sekolah_controller"



1.2 Cyclomatic Complexity

- $V(G) = 2$, ada *region* R1 dan R2
- $V(G) = 6 \text{ edges} - 6 \text{ nodes} + 2 = 2$
- $V(G) = 1 \text{ predicate nodes} + 1 = 2$

1.3 Independent Path

- Jalur 1: 1-2-5-6
- Jalur 2: 1-2-3-4-2-5-6

Test case dan hasil akan dijelaskan pada tabel 6.1 dibawah ini.

Tabel 6.1 Hasil pengujian unit method “input_proposal_sekolah” class “sekolah_controller”

No	No. Jalur	Prosedur Uji	Expected Result	Result	Status
1	1	Ketika hashfile yang diinputkan bukan 'foto(.png/.jpg)'	Data foto tidak tersimpan dalam database.	Data foto tidak tersimpan dalam database.	Valid
2	2	Ketika hashfile yang inputkan adalah 'foto(.png/.jpg)'	Data foto yang diinputkan tersimpan dalam database.	Data foto yang diinputkan tersimpan dalam database.	Valid

6.1.2 Pengujian Unit Method “pengiriman_dana” class “kasi_controller”

1. Pseudocode

1	proposal = proposal id	
1	dana_terkirim_1 = proposal dana_terkirim	
1	dana_terkirim_2 = Request jumlah;	
1	dana_total = dana_total	
1	selisih = dana_total - dana_terkirim_1	
2	if dana_terkirim_2 lebih dari selisih	
3	maka ada pesan gagal untuk pengiriman	
3	kembali ke halaman sebelumnya	
4	else	
4	dana_terkirim_update_	=
4	dana_terkirim_1+dana_terkirim_2	
4	foto = Request foto	
4	namaFoto = foto	
4	data_pengiriman	
4	proposal_id => Request id	
4	jumlah => Request jumlah	
4	foto => namaFoto	
4	data_proposal	
4	dana_terkirim => dana_terkirim_updated	
4	dataNotif	
4	proposal_id => Request->id	
4	dari => kasi	
4	tujuan => proposal->sekolah_id	

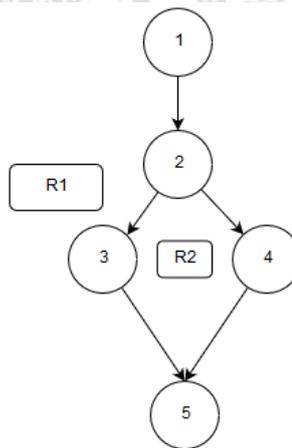


```

4         deskripsi => kirim_dana
4         status => belum
4         deskripsi = jumlah dana terkirim
4         membuat data notifikasi dan menyimpan data notifikasi
4         membuat data pengiriman dana dan menyimpan data
4 pengiriman
4         update data proposal
4         memindahkan data proposal dengan format nama
4         mengurutkan data pengiriman dana berdasarkan yang
4 terbaru
4         dataLog =
4             proposal_id => Request id
4             deskripsi => kasi mengirim dana
4             dari => kasi
4             data_id => data id
4         membuat data log aktivitas dan menyimpan data log
4 aktivitas
4         notif
4             message = Update Data Success
5         kembali kehalaman sebelumnya dengan notifikasi
    
```

2. Basis Path Testing

1.4 Flow Graph



Gambar 6.2 Flow graph method “pengiriman_dana” class “kasi_controller”

1.5 Cyclomatic Complexity

- $V(G) = 2$, ada 2 region R1 dan R2
- $V(G) = 5 \text{ edges} - 5 \text{ nodes} + 2 = 2$
- $V(G) = 1 \text{ predicate nodes} + 1 = 2$

1.6 Independent Path

- Jalur 1: 1-2-3-5
- Jalur 2: 1-2-4-5

Test case dan hasil akan dijelaskan pada tabel 6.2 dibawah ini.



Tabel 6.2 Hasil pengujian unit method “pengiriman_dana” class “kasi_controller”

No	No. Jalur	Prosedur Uji	Expected Result	Result	Status
1	1	Jika dana_terkirim_2 > selisih	Terdapat pesan gagal dan kembali pada halaman sebelumnya.	Terdapat pesan gagal dan kembali pada halaman sebelumnya.	Valid
2	2	Jika dana_terkirim_2 < =selisih	Data pengiriman dana berhasil disimpan dan kembali ke halaman detail proposal.	Data pengiriman dana berhasil disimpan dan kembali ke halaman detail proposal.	Valid

6.1.3 Pengujian Unit Method “input_update_dana” class “sekolah_controller”

1. Pseudocode

```

1 validator dokumen persetujuan tidak boleh lebih dari 2mb
1     total =required
1     barang = Request barang
1     harga = Request harga
1     no = 1;
1     total = Request total
1     mengambil data proposal melalui id
1     terkirim = dana_terkirim
1     terpakai = dana_terpakai
1     selisih = terkirim - terpakai
1     total = request total
2     jika total kurang dari selisih
3         maka
3             notif
3                 message = gagal, total penggunaan besar
dari dana terkirim kasi
3                 kembali ke halaman sebelumnya dengan
notif
4     else
4         sekarang = tanggal dana terpakai
4         inisialisasi nama file dan format file
4         menyimpan data pdf dengan nama_file
4         dana_terpakai_updated_ = terpakai+ total
4         dana_terpakai_updated =
4         dana_terpakai_updated_
4         dokumen_persetujuan = Request
4         dokumen_persetujuan
4         namaDokumen = dokumen_persetujuan
4         data_proposal
4         dana_terpakai = dana_terpakai_updated
4         data_penggunaan_dana
    
```

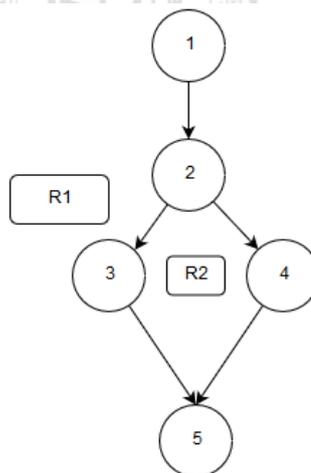


```

4         proposal_id => Request id
4         jumlah => total
4         dokumen => nama_file
4         dokumen_persetujuan = namaDokumen
4         dataNotif
4         proposal_id => proposal id,
4         dari = sekolah id,
4         tujuan = kasi
4         deskripsi = update_dana
4         tingkat = tingkat sekolah,
4         status = belum
4         deskripsi = sekolah mengupdate pemakaian
4         dana bantuan sebesar Rp total
4         memindahkan dokumen_persetujuan ke folder
4         yang ditentukan
4         membuat data notifikasi dan menyimpan data
4         notifikasi
4         meng-update data proposal bedasarkan id
4         membuat data peggunaan dana dan menyimpan
4         data peggunaan dana
4         mengurutkan data peggunaan dana
4         dataLog
4         proposal_id = Request id
4         deskripsi = update peggunaan dana oleh
4         sekolah
4         dari => sekolah
4         data_id => data id
4         membuat data log aktivitas dan menyimpan
4         data log aktivitas
4         notif
4         pesan menyimpan data sukses
5         kembali ke halaman detail proposal
    
```

2. Basis Path Testing

1.7 Flow Graph



Gambar 6.3 Flow graph method “input_update_dana” class “sekolah_controller”



1.8 Cyclomatic Complexity

- $V(G) = 2$, ada 2 region R1 dan R2
- $V(G) = 5 \text{ edges} - 5 \text{ nodes} + 2 = 2$
- $V(G) = 1 \text{ predicate nodes} + 1 = 2$

1.9 Independent Path

- Jalur 1: 1-2-3-5
- Jalur 2: 1-2-4-6

Test case dan hasil akan dijelaskan pada tabel 6.3 dibawah ini.

Tabel 6.3 Hasil pengujian unit method "input_update_dana" class "sekolah_controller"

No	No. Jalur	Prosedur Uji	Expected Result	Result	Status
1	1	Jika total > selisih	Terdapat pesan gagal dan kembali ke halaman sebelumnya	Terdapat pesan gagal dan kembali ke halaman sebelumnya	Valid
2	2	Jika total <= selisih	Data input update dana berhasil disimpan dan kembali ke halaman detail proposal	Data input update dana berhasil disimpan dan kembali ke halaman detail proposal	Valid

6.2 Pengujian Integrasi

Pengujian ini bertujuan untuk menguji desain dan konstruksi dari arsitektur perangkat lunak. Pada Tabel 6.4 menunjukan identifikasi dan rencana uji dalam pengujian itegrasi pada Sistem Pengelolaan Pendistribusian Dana Bantuan Sekolah, identifikasi dan rencana uji pengujjian itegrasi ditunjukkan pada tabel 6.4.

Tabel 6.4 Identifikasi dan Perancangan Pengujian Integrasi

No	Nama Kelas	Nama Method	Tujuan
1	sekolah_controller	input_proposal_sekolah(request)	Menambahkan proposal baru pada database.



	laporan	create()	
	notifikasi	create()	
2	kasi_controller	pengiriman_dana(request)	Mengirimkan dana bantuan terhadap proposal yang sudah disetujui oleh Kepala Seksi.
	pengiriman_dana	create()	
	proposal	update()	
	notifikasi	create()	
3	sekolah_controller	Input_update_dana(request)	Melakukan pendataan penggunaan dana yang sudah dipakai untuk proses bantuan sekolah.
	penggunaan_dana	create()	
	proposal	update()	
	notifikasi	create()	

Berikut adalah hasil pengujian integrasi berdasarkan identifikasi pada tabel 6.4

1. Hasil pengujian integrasi nomor 1 ditunjukkan pada tabel 6.5.

Tabel 6.5 Hasil Pengujian Integrasi Nomor 1

Nomor Uji	1
Input Pertama	"judul": "laporan 1", "deskripsi": "perbaikan", "dokumen": "proposal.pdf", "pengantar_dinas": "pengantar.pdf", "surat_permohonan": "surat.pdf", "foto": "image.png", "galeri": "image.png", "dana_diajukan": "1000000", "jumlah_rusak": "3", "tingkat_rusak": "sedang", "jumlah_siswa": "330", "rombongan_belajar": "30", "jenis_sekolah": "Negeri", "tingkat_sekolah": "SMP"

Method dari kelas sekolah_controller	Input_proposal_sekolah(request)
Output Pertama/ Input Kedua	<pre> "id": "1" "sekolah_id": "1", "judul": "laporan 1", "deskripsi": "perbaikan", "dokumen": "proposal.pdf", "pengantar_dinas": "pengantar.pdf", "surat_permohonan": "surat.pdf" "foto": "image.png", "galeri": "image.png", "status": "belum", "status_kerja": "0", "dana_diajukan": "1000000", "jumlah_rusak": "3", "tingkat_rusak": "sedang", "jumlah_siswa": "330", "rombongan_belajar": "30", "jenis_sekolah": "Negeri", "tingkat_sekolah": "SMP", "rate": "66.4", "created_at": "2018-7-17 20:46:35", "updated_at": "2018-7-17 20:46:35" </pre>
Method dari Kelas proposal	create()
Output kedua/input ketiga	<pre> "id": "1" "proposal_id": "1" "dari": "1", "tujuan": "kasi", "deskripsi": "buat_proposal", "tingkat": "smp", "status": "belum", </pre>

	"created_at": "2018-7-17 20:46:35", "updated_at": "2018-7-17 20:46:35"
Method dari kelas notifikasi	create()
Expected Result	Proposal baru tersimpan dalam database sesuai data yang telah dimasukkan.
Result	Proposal baru tersimpan dalam database sesuai data yang telah dimasukkan.
Status	Valid

2. Hasil pengujian integrasi nomor i ditunjukkan pada tabel 6.6

Tabel 6.6 Hasil Pengujian Integrasi Nomor 2

Nomor Uji	2
Input Pertama	"jumlah": "1000000", "foto": "image.png", "dokumen": "buktikirim.png"
Method dari kelas kasi_controller	pengiriman_dana(request)
Output Pertama/ Input Kedua	"id": "1" "proposal_id": "1", "jumlah": "1000000", "foto": "image.png", "dokumen": "buktikirim.png", "created_at": "2018-7-17 20:46:35", "updated_at": "2018-7-17 20:46:35"
Method dari kelas pengiriman_dana	create()
Output Kedua/ Input Ketiga	"dana_terkirim": "1000000"
Method dari kelas proposal	update()
Output Ketiga/ Input Keempat	"id": "2" "proposal_id": "1" "dari": "kasi", "tujuan": "3",

	<pre> “deskripsi”:"kirim_dana", “tingkat”:"null", “status”:"belum", “created_at”: “2018-7-17 20:46:35”, “updated_at”: “2018-7-17 20:46:35” </pre>
Method dari kelas notifikasi	create()
Expected Result	Mengirimkan dana bantuan terhadap proposal yang sudah disetujui oleh Kepala Seksi.
Result	Mengirimkan dana bantuan terhadap proposal yang sudah disetujui oleh Kepala Seksi.
Status	Valid

3. Hasil pengujian integrasi nomor 3 ditunjukkan pada tabel 6.7

Tabel 6.7 Hasil Pengujian Integrasi Nomor 3

Nomor Uji	3
Input Pertama	<pre> “jumlah”:"1000000" “dokumen”:"penggunaandana.pdf" “dokumen_persetujuan”:"updatedana.pdf" </pre>
Method dari kelas sekolah_controller	Input_update_dana(request)
Output Pertama/ Input Kedua	<pre> “id”:"1" “proposal_id” : “1”, “jumlah”:"1000000" “dokumen”:" penggunaandana.pdf" “dokumen_persetujuan”:" updatedana.pdf" “created_at”: “2018-7-17 20:46:35”, “updated_at”: “2018-7-17 20:46:35” </pre>
Method dari kelas pengiriman_dana	create()

Output Kedua/Input Ketiga	"dana_terpakai": "1000000"
Method dari kelas proposal	update()
Output Ketiga/Input Keempat	"id": "3" "proposal_id": "1" "dari": "3", "tujuan": "kasi", "deskripsi": "update_dana", "tingkat": "smp", "status": "belum", "created_at": "2018-7-17 20:46:35", "updated_at": "2018-7-17 20:46:35"
Method dari kelas notifikasi	create()
Expected Result	Melakukan pendataan penggunaan dana yang sudah dipakai untuk proses bantuan sekolah.
Result	Melakukan pendataan penggunaan dana yang sudah dipakai untuk proses bantuan sekolah.
Status	Valid

6.3 Pengujian Validasi

Pengujian validasi dilakukan ketika kebutuhan yang sudah ditetapkan sebagai bagian dari pemodelan kebutuhan divalidasi terhadap perangkat lunak yang telah dibangun. Pengujian validasi dikatakan berhasil apabila fungsi yang ada pada perangkat lunak sesuai dengan yang diharapkan. Pada pengujian validasi ini digunakan metode *blackbox testing*. Dengan *blackbox testing* memungkinkan pembuat perangkat lunak untuk memperoleh kondisi yang terjadi untuk suatu masukan yang akan menjalankan semua kebutuhan.

Tabel 6.8 Hasil Pengujian Validasi Login

Kode Kebutuhan	SRS-SP-F-1
Nama Kasus Uji	<i>Login</i>
Prosedur	<ol style="list-style-type: none"> 1. Membuka halaman login 2. Mengisi <i>form login</i> dengan password dan username yang benar. 3. Menekan tombol <i>login</i>.

Hasil yang diharapkan	Sistem menampilkan halaman utama.
Hasil	Sistem menampilkan halaman utama.
Status	Valid

Tabel 6.9 Hasil Pengujian Validasi *Login Alternatif 1*

Kode Kebutuhan	SRS- SP-F-1
Nama Kasus Uji	<i>Login</i>
Prosedur	<ol style="list-style-type: none"> 1. Membuka halaman login 2. Mengisi <i>form login</i> dengan password dan username yang salah. 3. Menekan tombol <i>login</i>.
Hasil yang diharapkan	Sistem menampilkan halaman login karena username atau password salah.
Hasil	Sistem menampilkan halaman login karena username atau password salah.
Status	Valid

Tabel 6.10 Hasil Pengujian Validasi *Login Alternatif 2*

Kode Kebutuhan	SRS- SP-F-1
Nama Kasus Uji	<i>Login</i>
Prosedur	<ol style="list-style-type: none"> 1. Membuka halaman login. 2. Mengosongkan atau tidak mengisi form login. 3. Menekan tombol <i>login</i>.
Hasil yang diharapkan	Sistem menampilkan halaman login karena username atau password tidak isi.
Hasil	Sistem menampilkan halaman login karena username dan password salah.
Status	Valid

Tabel 6.11 Hasil Pengujian Validasi *Logout*

Kode Kebutuhan	SRS-SP-F-2
Nama Kasus Uji	<i>Logout</i>
Prosedur	1. Memilih menu " <i>logout</i> "
Hasil yang diharapkan	Sistem menampilkan halaman <i>login</i> .

Hasil	Sistem menampilkan halaman <i>login</i> .
Status	Valid

Tabel 6.12 Hasil Pengujian Validasi Buat Proposal

Kode Kebutuhan	SRS-SP-F-3
Nama Kasus Uji	Buat Proposal
Prosedur	<ol style="list-style-type: none"> 1. Menekan menu Buat Proposal. 2. Mengisi form buatproposal dengan lengkap. 3. Menekan tombol submit.
Hasil yang diharapkan	Sistem menyimpan data proposal dan kembali ke halaman sebelumnya.
Hasil	Sistem menyimpan data proposal dan kembali ke halaman sebelumnya.
Status	Valid

Tabel 6.13 Hasil Pengujian Validasi Buat Proposal Alternatif 1

Kode Kebutuhan	SRS-SP-F-3
Nama Kasus Uji	Buat Proposal
Prosedur	<ol style="list-style-type: none"> 1. Menekan menu Buat Proposal. 2. Mengisi form buat proposal dengan tidak lengkap. 3. Menekan tombol submit.
Hasil yang diharapkan	Sistem menampilkan pesan untuk mengisi lengkap form buat proposal.
Hasil	Sistem menampilkan pesan untuk mengisi lengkap form buat proposal.
Status	Valid

Tabel 6.14 Hasil Pengujian Validasi Buat Proposal Alternatif 2

Kode Kebutuhan	SRS-SP-F-3
Nama Kasus Uji	Buat Proposal
Prosedur	<ol style="list-style-type: none"> 1. Menekan menu Buat Proposal. 2. Mengisi form buat proposal dengan file selain docx/pdf. 3. Menekan tombol submit.

Hasil yang diharapkan	Sistem menampilkan pesan gagal membuat proposal.
Hasil	Sistem menampilkan pesan gagal membuat proposal.
Status	Valid

Tabel 6.15 Hasil Pengujian Validasi Buat Proposal Alternatif 3

Kode Kebutuhan	SRS-SP-F-3
Nama Kasus Uji	Buat Proposal
Prosedur	<ol style="list-style-type: none"> 1. Menekan menu Buat Proposal. 2. Mengisi form buat proposal dengan file melebihi 2bm. 3. Menekan tombol submit.
Hasil yang diharapkan	Sistem menampilkan pesan gagal membuat proposal.
Hasil	Sistem menampilkan pesan gagal membuat proposal.
Status	Valid

Tabel 6.16 Hasil Pengujian Validasi Ubah Proposal

Kode Kebutuhan	SRS-SP-F-4
Nama Kasus Uji	Ubah Proposal
Prosedur	<ol style="list-style-type: none"> 1. Menekan menu Ubah Proposal. 2. Mengisi form ubah proposal dengan lengkap. 3. Menekan tombol submit.
Hasil yang diharapkan	Sistem menyimpan data proposal dan kembali ke halaman sebelumnya.
Hasil	Sistem menyimpan data proposal dan kembali ke halaman sebelumnya.
Status	Valid

Tabel 6.17 Hasil Pengujian Validasi Ubah Proposal Alternatif 1

Kode Kebutuhan	SRS-SP-F-3
Nama Kasus Uji	Ubah Proposal
Prosedur	<ol style="list-style-type: none"> 1. Menekan menu Ubah Proposal. 2. Mengisi form ubah proposal dengan tidak lengkap. 3. Menekan tombol submit.

Hasil yang diharapkan	Sistem menampilkan pesan untuk mengisi lengkap form ubah proposal.
Hasil	Sistem menampilkan pesan untuk mengisi lengkap form ubah proposal.
Status	Valid

Tabel 6.18 Hasil Pengujian Validasi Ubah Proposal Alternatif 2

Kode Kebutuhan	SRS-SP-F-3
Nama Kasus Uji	Ubah Proposal
Prosedur	<ol style="list-style-type: none"> 1. Menekan menu Ubah Proposal. 2. Mengisi form ubah laporan dengan file selain docx/pdf. 3. Menekan tombol submit.
Hasil yang diharapkan	Sistem menampilkan pesan gagal mengubah proposal.
Hasil	Sistem menampilkan pesan gagal mengubah proposal.
Status	Valid

Tabel 6.19 Hasil Pengujian Validasi Ubah Proposal Alternatif 3

Kode Kebutuhan	SRS-SP-F-3
Nama Kasus Uji	Ubah Proposal
Prosedur	<ol style="list-style-type: none"> 1. Menekan menu Ubah Proposal. 2. Mengisi form ubah laporan dengan file melebihi 2mb. 3. Menekan tombol submit.
Hasil yang diharapkan	Sistem menampilkan pesan gagal mengubah proposal.
Hasil	Sistem menampilkan pesan gagal mengubah proposal.
Status	Valid

Tabel 6.20 Hasil Pengujian Validasi Hapus Proposal

Kode Kebutuhan	SRS-SP-F-4
Nama Kasus Uji	Hapus Proposal

Prosedur	<ol style="list-style-type: none"> 1. Menekan menu Detail Proposal. 2. Menekan tombol Hapus. 3. Menekan tombol “ya” pada pesan konfirmasi.
Hasil yang diharapkan	Sistem menghapus data proposal.
Hasil	Sistem menghapus data proposal.
Status	Valid

Tabel 6.21 Hasil Pengujian Validasi Lihat Daftar Proposal

Kode Kebutuhan	SRS-SRS-F-6
Nama Kasus Uji	Lihat Daftar Proposal
Prosedur	1. Memilih menu Daftar Proposal
Hasil yang diharapkan	Sistem menampilkan halaman daftar proposal.
Hasil	Sistem menampilkan halaman daftar proposal.
Status	Valid

Tabel 6.22 Hasil Pengujian Validasi Lihat Detail Proposal

Kode Kebutuhan	SRS- SP-F-7
Nama Kasus Uji	Lihat Detail Proposal
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol ‘Detail Proposal’.
Hasil yang diharapkan	Sistem menampilkan detail proposal yang dipilih.
Hasil	Sistem menampilkan detail proposal yang dipilih.
Status	Valid

Tabel 6.23 Hasil Pengujian Validasi Approve Proposal

Kode Kebutuhan	SRS- SP-F-8
Nama Kasus Uji	Approve Proposal
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol approve. 3. Menekan tombol “ya” pada pesan konfirmasi.
Hasil yang diharapkan	Sistem menyimpan data status proposal yang sudah di approve.

Hasil	Sistem menyimpan data status proposal yang sudah di approve.
Status	Valid

Tabel 6.24 Hasil Pengujian Validasi Decline Proposal

Kode Kebutuhan	SRS-SP-F-9
Nama Kasus Uji	Decline Proposal
Prosedur	<ol style="list-style-type: none"> 1. Memilih halaman Daftar Laporan 2. Menekan tombol decline 3. Menekan tombol “ya” pada pesan konfirmasi.
Hasil yang diharapkan	Sistem menyimpan data status proposal yang sudah di decline.
Hasil	Sistem menyimpan data status proposal yang sudah di decline.
Status	Valid

Tabel 6.25 Hasil Pengujian Validasi Kirim Tugas Verifikasi

Kode Kebutuhan	SRS-SP-F-10
Nama Kasus Uji	Kirim Tugas Verifikasi
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol “Detail Proposal”. 3. Menekan tombol “Kirim Tugas Verifikasi”. 4. Mengisi form kirim tugas verifikasi dengan lengkap. 5. Menekan tombol submit.
Hasil yang diharapkan	Sistem menyimpan data kirim tugas verifikasi dan kembali kehalaman sebelumnya.
Hasil	Sistem menyimpan data kirim tugas verifikasi dan kembali kehalaman sebelumnya.
Status	Valid

Tabel 6.26 Hasil Pengujian Validasi Kirim Tugas Verifikasi Alternatif 1

Kode Kebutuhan	SRS-SP-F-10
Nama Kasus Uji	Kirim Tugas Verifikasi
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal.

	<ol style="list-style-type: none"> 2. Menekan tombol "Detail Proposal". 3. Menekan tombol "Kirim Tugas Verifikasi". 4. Mengisi form kirim tugas verifikasi dengan dengan tidak lengkap. 5. Menekan tombol submit.
Hasil yang diharapkan	Terdapat pemberitahuan untuk mengisi form dengan lengkap.
Hasil	Terdapat pemberitahuan untuk mengisi form dengan lengkap.
Status	Valid

Tabel 6.27 Hasil Pengujian Validasi Kirim Tugas Verifikasi Alternatif 2

Kode Kebutuhan	SRS-SP-F-10
Nama Kasus Uji	Kirim Tugas Verifikasi
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol "Detail Proposal". 3. Menekan tombol "Kirim Tugas Verifikasi". 4. Mengisi form dengan dengan file selain docx/pdf. 5. Menekan tombol submit.
Hasil yang diharapkan	Sistem gagal menyimpan data kirim tugas verifikasi.
Hasil	Sistem gagal menyimpan data kirim tugas verifikasi.
Status	Valid

Tabel 6.28 Hasil Pengujian Validasi Kirim Tugas Verifikasi Alternatif 3

Kode Kebutuhan	SRS-SP-F-10
Nama Kasus Uji	Kirim Tugas Verifikasi
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol "Detail Proposal". 3. Menekan tombol "Kirim Tugas Verifikasi". 4. Mengisi form dengan dengan file melebihi 2mb. 5. Menekan tombol submit.
Hasil yang diharapkan	Sistem gagal menyimpan data kirim tugas verifikasi.
Hasil	Sistem gagal menyimpan data kirim tugas verifikasi.
Status	Valid

Tabel 6.29 Hasil Pengujian Validasi Verifikasi Proposal

Kode Kebutuhan	SRS-SP-F-11
Nama Kasus Uji	Verifikasi Proposal
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol "Detail Proposal". 3. Menekan tombol "Verifikasi Proposal". 4. Mengisi form dengan lengkap. 5. Menekan tombol submit.
Hasil yang diharapkan	Sistem menyimpan data verifikasi proposal dan kembali kehalaman sebelumnya.
Hasil	Sistem menyimpan data verifikasi proposal dan kembali kehalaman sebelumnya.
Status	Valid

Tabel 6.30 Hasil Pengujian Validasi Verifikasi Proposal Alternatif 1

Kode Kebutuhan	SRS-SP-F-11
Nama Kasus Uji	Verifikasi Proposal
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol "Detail Proposal". 3. Menekan tombol "Verifikasi Proposal". 4. Mengisi form dengan dengan file selain docx/pdf. 5. Menekan tombol submit.
Hasil yang diharapkan	Sistem gagal menyimpan data verifikasi proposal.
Hasil	Sistem gagal menyimpan data verifikasi proposal.
Status	Valid

Tabel 6.31 Hasil Pengujian Validasi Verifikasi Proposal Alternatif 2

Kode Kebutuhan	SRS-SP-F-11
Nama Kasus Uji	Verifikasi Proposal
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol "Detail Proposal". 3. Menekan tombol "Verifikasi Proposal". 4. Mengisi form dengan dengan file melebihi 2mb. 5. Menekan tombol submit.
Hasil yang diharapkan	Sistem gagal menyimpan data verifikasi proposal.

Hasil	Sistem gagal menyimpan data verifikasi proposal.
Status	Valid

Tabel 6.32 Hasil Pengujian Validasi Notifikasi

Kode Kebutuhan	SRS-SP-F-12
Nama Kasus Uji	Notifikasi
Prosedur	1. Memilih menu “Notifikasi”.
Hasil yang diharapkan	Sistem menampilkan data notifikasi.
Hasil	Sistem menampilkan data notifikasi.
Status	Valid

Tabel 6.33 Hasil Pengujian Validasi Lihat Daftar Prioritas

Kode Kebutuhan	SRS-SRS-F-13
Nama Kasus Uji	Lihat Daftar Prioritas
Prosedur	1. Memilih menu Daftar Prioritas
Hasil yang diharapkan	Sistem menampilkan halaman daftar prioritas.
Hasil	Sistem menampilkan halaman daftar prioritas.
Status	Valid

Tabel 6.34 Hasil Pengujian Validasi Kirim Dana

Kode Kebutuhan	SRS-SP-F-14
Nama Kasus Uji	Kirim Dana
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol “Detail Proposal”. 3. Menekan tombol “Kirim Dana”. 4. Mengisi form kirim dana dengan lengkap. 5. Menekan tombol submit.
Hasil yang diharapkan	Sistem menyimpan data kirim dana dan kembali kehalaman sebelumnya.
Hasil	Sistem menyimpan data kirim dana dan kembali kehalaman sebelumnya.
Status	Valid

Tabel 6.35 Hasil Pengujian Validasi Kirim Dana Alternatif 1

Kode Kebutuhan	SRS-SP-F-14
Nama Kasus Uji	Kirim Dana
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol "Detail Proposal". 3. Menekan tombol "Kirim Dana". 4. Mengisi form kirim dana dengan tidak lengkap. 5. Menekan tombol submit.
Hasil yang diharapkan	Terdapat pemberitahuan untuk mengisi form.
Hasil	Terdapat pemberitahuan untuk mengisi form.
Status	Valid

Tabel 6.36 Hasil Pengujian Validasi Kirim Dana Alternatif 2

Kode Kebutuhan	SRS-SP-F-14
Nama Kasus Uji	Kirim Dana
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol "Detail Proposal". 3. Menekan tombol "Kirim Dana". 4. Mengisi form kirim dana dengan file selain docx/pdf. 5. Menekan tombol submit.
Hasil yang diharapkan	Sistem gagal untuk menyimpan data kirim dana.
Hasil	Sistem gagal untuk menyimpan data kirim dana.
Status	Valid

Tabel 6.37 Hasil Pengujian Validasi Kirim Dana Alternatif 3

Kode Kebutuhan	SRS-SP-F-14
Nama Kasus Uji	Kirim Dana
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol "Detail Proposal". 3. Menekan tombol "Kirim Dana". 4. Mengisi form kirim dana dengan file melebihi 2mb. 5. Menekan tombol submit.
Hasil yang diharapkan	Sistem gagal untuk menyimpan data kirim dana.

Hasil	Sistem gagal untuk menyimpan data kirim dana.
Status	Valid

Tabel 6.38 Hasil Pengujian Validasi Kirim Dana Alternatif 4

Kode Kebutuhan	SRS-SP-F-14
Nama Kasus Uji	Kirim Dana
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol "Detail Proposal". 3. Menekan tombol "Kirim Dana". 4. Mengisi form kirim melebihi total dana yang ditetapkan. 5. Menekan tombol submit.
Hasil yang diharapkan	Sistem gagal untuk menyimpan data kirim dana.
Hasil	Sistem gagal untuk menyimpan data kirim dana.
Status	Valid

Tabel 6.39 Hasil Pengujian Validasi Input Dana Terpakai

Kode Kebutuhan	SRS-SP-F-15
Nama Kasus Uji	Input Dana Terpakai
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol "Detail Proposal". 3. Menekan tombol "Input Dana Terpakai". 4. Mengisi form input dana terpakai dengan lengkap. 5. Menekan tombol submit.
Hasil yang diharapkan	Sistem menyimpan data input dana terpakai dan kembali ke halaman sebelumnya.
Hasil	Sistem menyimpan data input dana terpakai dan kembali ke halaman sebelumnya.
Status	Valid

Tabel 6.40 Hasil Pengujian Validasi Input Dana Terpakai Alternatif 1

Kode Kebutuhan	SRS-SP-F-15
Nama Kasus Uji	Input Dana Terpakai
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal.

	<ol style="list-style-type: none"> 2. Menekan tombol "Detail Proposal". 3. Menekan tombol "Input Dana Terpakai". 4. Mengisi form input dana terpakai dengan tidak lengkap. 5. Menekan tombol submit.
Hasil yang diharapkan	Terdapat pemberitahuan untuk mengisi form dengan lengkap.
Hasil	Terdapat pemberitahuan untuk mengisi form dengan lengkap.
Status	Valid

Tabel 6.41 Hasil Pengujian Validasi Input Dana Terpakai Alternatif 2

Kode Kebutuhan	SRS-SP-F-15
Nama Kasus Uji	Input Dana Terpakai
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol "Detail Proposal". 3. Menekan tombol "Input Dana Terpakai". 4. Mengisi form input dana terpakai dengan file selain docx/pdf. 5. Menekan tombol submit.
Hasil yang diharapkan	Sistem gagal menyimpan data input dana terpakai.
Hasil	Sistem gagal menyimpan data input dana terpakai.
Status	Valid

Tabel 6.42 Hasil Pengujian Validasi Input Dana Terpakai Alternatif 3

Kode Kebutuhan	SRS-SP-F-15
Nama Kasus Uji	Input Dana Terpakai
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol "Detail Proposal". 3. Menekan tombol "Input Dana Terpakai". 4. Mengisi form input dana terpakai dengan file melebihi 2mb. 5. Menekan tombol submit.
Hasil yang diharapkan	Sistem gagal menyimpan data input dana terpakai.
Hasil	Sistem gagal menyimpan data input dana terpakai.
Status	Valid

Tabel 6.43 Hasil Pengujian Validasi Input Dana Terpakai Alternatif 4

Kode Kebutuhan	SRS-SP-F-15
Nama Kasus Uji	Input Dana Terpakai
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol "Detail Proposal". 3. Menekan tombol "Input Dana Terpakai". 4. Mengisi form input dana terpakai dengan melebihi dana yang dikirimkan. 5. Menekan tombol submit.
Hasil yang diharapkan	Sistem gagal menyimpan data input dana terpakai.
Hasil	Sistem gagal menyimpan data input dana terpakai.
Status	Valid

Tabel 6.44 Hasil Pengujian Validasi Log Aktivitas Proposal

Kode Kebutuhan	SRS-SP-F-16
Nama Kasus Uji	Log Aktivitas Proposal
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Proposal. 2. Menekan tombol "Log Aktivitas".
Hasil yang diharapkan	Sistem menampilkan data log aktivitas proposal
Hasil	Sistem menampilkan data log aktivitas proposal
Status	Valid

Tabel 6.45 Hasil Pengujian Daftar Sekolah

Kode Kebutuhan	SRS-SP-F-17
Nama Kasus Uji	Daftar Sekolah
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu halaman Daftar Sekolah.
Hasil yang diharapkan	Sistem menampilkan halaman daftar sekolah.
Hasil	Sistem menampilkan halaman daftar sekolah.
Status	Valid

Tabel 6.46 Hasil Pengujian Validasi Daftar Kepala Seksi

Kode Kebutuhan	SRS-SP-F-18
-----------------------	-------------

Nama Kasus Uji	Daftar Kepala Seksi
Prosedur	1. Memilih menu halaman Daftar Kepala Seksi.
Hasil yang diharapkan	Sistem menampilkan halaman daftar kepala seksi.
Hasil	Sistem menampilkan halaman daftar kepala seksi.
Status	Valid

Tabel 6.47 Hasil Pengujian Validasi Daftar Tim Verifikasi

Kode Kebutuhan	SRS-SP-F-19
Nama Kasus Uji	Daftar Tim Verifikasi
Prosedur	1. Memilih menu halaman Daftar Tim Verifikasi
Hasil yang diharapkan	Sistem menampilkan halaman daftar tim verifikasi
Hasil	Sistem menampilkan halaman daftar tim verifikasi
Status	Valid

Tabel 6.48 Hasil Pengujian Validasi Tambah Sekolah

Kode Kebutuhan	SRS-SP-F-20
Nama Kasus Uji	Tambah Sekolah
Prosedur	1. Memilih menu halaman Tambah Sekolah. 2. Mengisi form tambah sekolah dengan lengkap.
Hasil yang diharapkan	Sistem menyimpan data sekolah dan kembali ke halaman sebelumnya.
Hasil	Sistem menyimpan data sekolah dan kembali ke halaman sebelumnya.
Status	Valid

Tabel 6.49 Hasil Pengujian Validasi Tambah Sekolah Alternatif 1

Kode Kebutuhan	SRS-SP-F-20
Nama Kasus Uji	Tambah Sekolah
Prosedur	1. Memilih menu halaman Tambah Sekolah. 2. Mengisi form tambah sekolah dengan tidak lengkap.
Hasil yang diharapkan	Terdapat pemberitahuan untuk mengisi form dengan lengkap.

Hasil	Terdapat pemberitahuan untuk mengisi form dengan lengkap.
Status	Valid

Tabel 6.50 Hasil Pengujian Validasi Ubah Sekolah

Kode Kebutuhan	SRS-SP-F-21
Nama Kasus Uji	Ubah Sekolah
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu halaman Daftar Sekolah 2. Menekan tombol 'ubah'. 3. Mengisi form ubah sekolah dengan lengkap. 4. Menekan tombol submit.
Hasil yang diharapkan	Sistem menyimpan data ubah data sekolah dan kembali ke halaman sebelumnya.
Hasil	Sistem menyimpan data ubah data sekolah dan kembali ke halaman sebelumnya.
Status	Valid

Tabel 6.51 Hasil Pengujian Validasi Ubah Sekolah Alternatif 1

Kode Kebutuhan	SRS-SP-F-21
Nama Kasus Uji	Ubah Sekolah
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu halaman Daftar Sekolah 2. Menekan tombol 'ubah'. 3. Mengisi form ubah sekolah dengan tidak lengkap. 4. Menekan tombol submit.
Hasil yang diharapkan	Terdapat pemberitahuan untuk mengisi form dengan lengkap.
Hasil	Terdapat pemberitahuan untuk mengisi form dengan lengkap.
Status	Valid

Tabel 6.52 Hasil Pengujian Validasi Hapus Sekolah.

Kode Kebutuhan	SRS-SP-F-22
Nama Kasus Uji	Hapus Sekolah
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu halaman Daftar Sekolah

	<ol style="list-style-type: none"> 2. Menekan tombol 'hapus'. 3. Menekan tombol 'ya' pada pesan konfirmasi
Hasil yang diharapkan	Sistem menghapus data sekolah dan kembali pada halaman sebelumnya.
Hasil	Sistem menghapus data sekolah dan kembali pada halaman sebelumnya.
Status	Valid

Tabel 6.53 Hasil Pengujian Validasi Tambah Kepala Seksi

Kode Kebutuhan	SRS-SP-F-23
Nama Kasus Uji	Tambah Kepala Seksi
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu halaman Tambah Kepala Seksi. 2. Mengisi form tambah kepala seksi dengan lengkap.
Hasil yang diharapkan	Sistem menyimpan data kepala seksi dan kembali ke halaman sebelumnya.
Hasil	Sistem menyimpan data kepala seksi dan kembali ke halaman sebelumnya.
Status	Valid

Tabel 6.54 Hasil Pengujian Validasi Tambah Kepala Seksi Alternatif 1

Kode Kebutuhan	SRS-SP-F-23
Nama Kasus Uji	Tambah Kepala Seksi
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu halaman Tambah Kepala Seksi. 2. Mengisi form tambah kepala seksi dengan tidak lengkap.
Hasil yang diharapkan	Terdapat pemberitahuan untuk mengisi form dengan lengkap.
Hasil	Terdapat pemberitahuan untuk mengisi form dengan lengkap.
Status	Valid

Tabel 6.55 Hasil Pengujian Validasi Ubah Kepala Seksi

Kode Kebutuhan	SRS-SP-F-24
-----------------------	-------------

Nama Kasus Uji	Ubah Kepala Seksi
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Kepala Seksi. 2. Menekan tombol 'ubah'. 3. Mengisi form ubah kepala seksi dengan lengkap.
Hasil yang diharapkan	Sistem menyimpan data ubah kepala seksi dan kembali ke halaman sebelumnya.
Hasil	Sistem menyimpan data ubah kepala seksi dan kembali ke halaman sebelumnya.
Status	Valid

Tabel 6.56 Hasil Pengujian Validasi Ubah Kepala Seksi Alternatif 1

Kode Kebutuhan	SRS-SP-F-24
Nama Kasus Uji	Ubah Kepala Seksi
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Kepala Seksi. 2. Menekan tombol 'ubah'. 3. Mengisi form ubah kepala seksi dengan tidak lengkap.
Hasil yang diharapkan	Terdapat pemberitahuan untuk mengisi form dengan lengkap.
Hasil	Terdapat pemberitahuan untuk mengisi form dengan lengkap.
Status	Valid

Tabel 6.57 Hasil Pengujian Validasi Hapus Kepala Seksi

Kode Kebutuhan	SRS-SP-F-25
Nama Kasus Uji	Mengirim Notifikasi
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu Daftar Kepala Seksi. 2. Menekan tombol 'hapus'. 3. Menekan tombol 'ya' pada pesan konfirmasi
Hasil yang diharapkan	Sistem berhasil menghapus data kepala seksi dan kembali ke halaman sebelumnya.
Hasil	Sistem berhasil menghapus data kepala seksi dan kembali ke halaman sebelumnya.
Status	Valid

Tabel 6.58 Hasil Pengujian Validasi Tambah Tim Verifikasi

Kode Kebutuhan	SRS-SP-F-26
Nama Kasus Uji	Tambah Tim Verifikasi
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu halaman Tambah Tim Verifikasi 2. Mengisi form tambah tim verifikasi dengan lengkap. 3. Menekan tombol submit.
Hasil yang diharapkan	Sistem menyimpan data tim verifikasi dan kembali ke halaman sebelumnya.
Hasil	Sistem menyimpan data tim verifikasi dan kembali ke halaman sebelumnya.
Status	Valid

Tabel 6.59 Hasil Pengujian Validasi Tambah Tim Verifikasi Alternatif 1

Kode Kebutuhan	SRS-SP-F-26
Nama Kasus Uji	Tambah Tim Verifikasi
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu halaman Tambah Tim Verifikasi 2. Mengisi form tambah tim verifikasi dengan tidak lengkap. 3. Menekan tombol submit.
Hasil yang diharapkan	Terdapat pemberitahuan untuk mengisi form dengan lengkap.
Hasil	Terdapat pemberitahuan untuk mengisi form dengan lengkap.
Status	Valid

Tabel 6.60 Hasil Pengujian Validasi Ubah Tim Verifikasi

Kode Kebutuhan	SRS-SP-F-27
Nama Kasus Uji	Ubah Tim Verifikasi
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu halaman Daftar Tim Verifikasi 2. Menekan tombol 'ubah' 3. Mengisi form ubah tim verifikasi dengan lengkap. 4. Menekan tombol submit.
Hasil yang diharapkan	Sistem berhasil menyimpan data ubah tim verifikasi dan kembali ke halaman sebelumnya.

Hasil	Sistem berhasil menyimpan data ubah tim verifikasi dan kembali ke halaman sebelumnya.
Status	Valid

Tabel 6.61 Hasil Pengujian Validasi Ubah Tim Verifikasi Alternatif 1

Kode Kebutuhan	SRS-SP-F-27
Nama Kasus Uji	Ubah Tim Verifikasi
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu halaman Daftar Tim Verifikasi 2. Menekan tombol 'ubah' 3. Mengisi form ubah tim verifikasi dengan tidak lengkap. 4. Menekan tombol submit.
Hasil yang diharapkan	Terdapat pemberitahuan untuk mengisi form dengan lengkap.
Hasil	Terdapat pemberitahuan untuk mengisi form dengan lengkap.
Status	Valid

Tabel 6.62 Hasil Pengujian Validasi Hapus Tim Verifikasi

Kode Kebutuhan	SRS-SP-F-28
Nama Kasus Uji	Ubah Tim Verifikasi
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu halaman Daftar Tim Verifikasi 2. Menekan tombol 'hapus' 3. Menekan tombol 'ya' pada pesan konfirmasi.
Hasil yang diharapkan	Sistem berhasil menghapus data tim verifikasi dan kembali ke halaman sebelumnya
Hasil	Sistem berhasil menghapus data tim verifikasi dan kembali ke halaman sebelumnya
Status	Valid

Tabel 6.63 Hasil Pengujian Validasi Lihat Rekap Bantuan

Kode Kebutuhan	SRS-SP-F-29
Nama Kasus Uji	Lihat Rekap Bantuan
Prosedur	<ol style="list-style-type: none"> 1. Memilih menu halaman Daftar Proposal. 2. Menekan tombol "Rekap Bantuan"

Hasil yang diharapkan	Sistem menampilkan halaman rekap bantuan.
Hasil	Sistem menampilkan halaman rekap bantuan.
Status	Valid

6.4 Pengujian *Compatibility*

Pengujian *compatibility* dapat dilihat pada Tabel 6.64 dan hasil pengujian *compatibility* dapat dilihat pada Gambar 6.4, pengujian menggunakan aplikasi SortSite 5.

Tabel 6.64 Pengujian *Compatibility*

Kode Kebutuhan	SRS-SP-NF-1
Prosedur	Menjalankan sistem pada browser yang berbeda
Hasil yang diharapkan	Fungsi pada sistem dapat berjalan dengan baik pada 8 browser yang berbeda
Hasil	Fungsi pada sistem dapat berjalan dengan baik pada 8 browser yang berbeda
Status	Valid

The screenshot shows the SortSite 5 interface for a compatibility test. The URL is https://uiiskripsi.000webhostapp.com/. The 'Compatibility' tab is active, showing a table of browser support and a list of issues. The table indicates that the system is compatible with all tested browsers (IE, Edge, Firefox, Safari, Opera, Chrome, iOS, and Android) across various versions. The 'Issues' section shows 0 critical, major, or minor issues.

Browser	IE			Edge		Firefox		Safari		Opera		Chrome		iOS			Android	
Version	9	10	11	17	63	≤ 11	12	55	70	≤ 10	11	12	≤ 3	4*				
Critical Issues	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Major Issues	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Minor Issues	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Key
 ● Missing content or functionality
 ● Major layout or performance problems
 ● Minor layout or performance problems

* Most Android devices from 4.4 onwards use Chrome as the default browser, older versions use the original Android stock browser

Priority	Description and URL	Guideline and Line#	Count
Expand all 0 issues			

Gambar 6.4 Hasil Pengujian *Compatibility*

Pada gambar 6.4 menunjukkan hasil pengujian dari pengujian *compatibility* dimana dihasilkan bahwa sistem ini berjalan pada 8 *browser* yang berbeda.



BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan hasil analisis kebutuhan, perancangan, implementasi, dan pengujian yang telah dilakukan, maka dapat disimpulkan sebagai berikut:

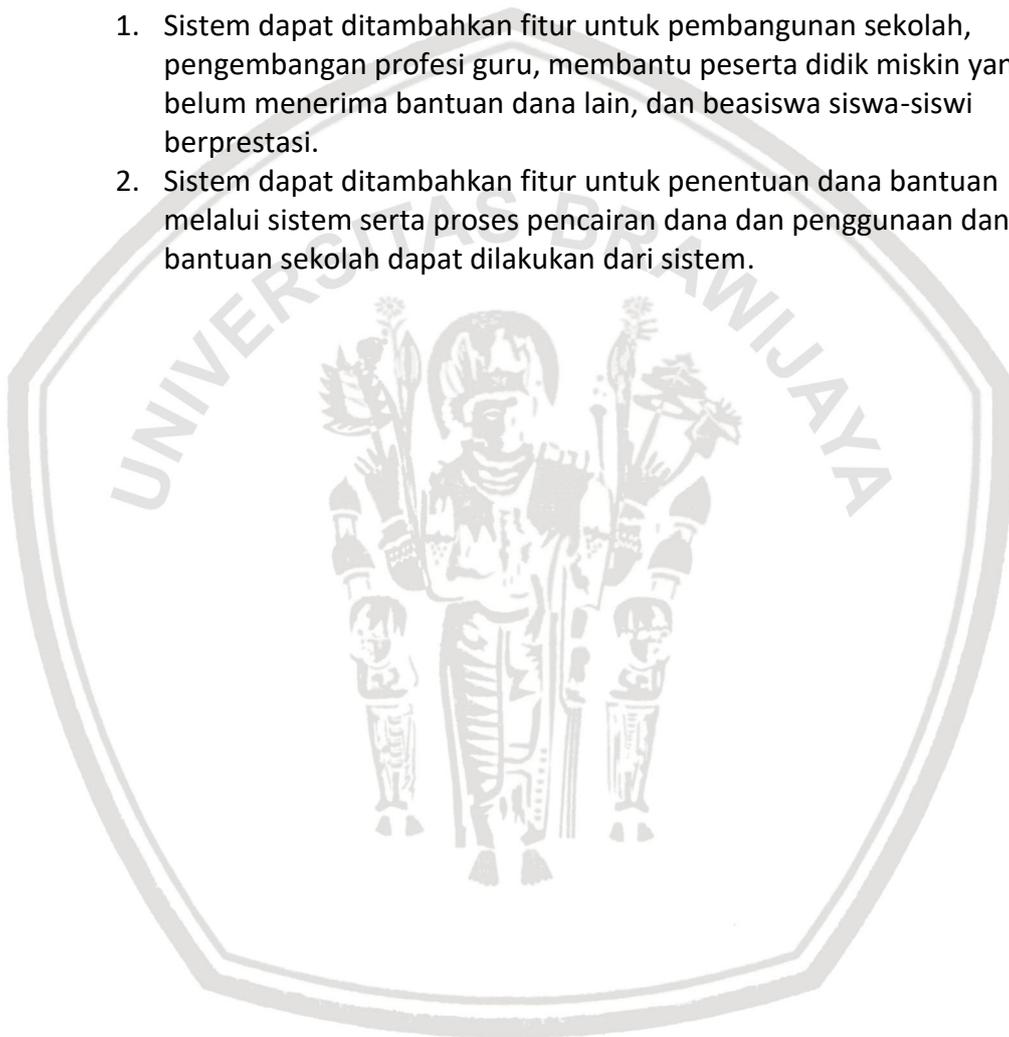
1. Berdasarkan hasil dari analisis kebutuhan dan analisis data, Sistem Pengelolaan Pendistribusian Dana Bantuan Sekolah memiliki 29 kebutuhan fungsional yang dapat membantu pihak Dinas Pendidikan dan Kebudayaan Kota Serang membantu meningkatkan sarana dan prasarana sekolah. Kebutuhan fungsional yang didapat telah mencakup permasalahan yang ada yaitu menyediakan form online untuk menginputkan proposal bagi pihak sekolah yang nantinya akan diperiksa oleh pihak Dinas Pendidikan dan Kebudayaan Kota Serang dan akan ditentukan prioritas sekolah yang akan dibantu lebih dulu. Kebutuhan- kebutuhan tersebut nantinya dapat memberikan solusi untuk masalah dalam proses penggalian dan analisis kebutuhan. Kemudian dari hasil analisis yang di dapatkan di modelkan dengan *use case diagram dan usecase scenario*.
2. Berdasarkan perancangan yang telah dilakukan, didapatkan hasil dari perancangan arsitektur, perancangan komponen, perancangan basis data, dan perancangan antarmuka berdasarkan tahap analisis kebutuhan. Perancangan arsitektur meliputi *sequence diagram dan class diagram*, sedangkan pada perancangan komponen mendapatkan algoritma utama untuk diterapkan dalam sistem. Perancangan basis data mendapatkan *Conceptual Data Model (CDM)*. Perancangan antar muka mendapatkan antar rancangan antarmuka yang nantinya dipakai dalam implementasi antarmuka sistem. Pada implementasi data diperoleh query untuk mengimplementasikan perancangan database. Dan pada implementasi antarmuka diperoleh tampilan web yang digunakan untuk interaksi antara sistem dan pengguna sistem.
3. Berdasarkan implementasi yang telah dilakukan, dihasilkan spesifikasi sistem, kode program, basis data, dan antarmuka. Pada spesifikasi sistem terdapat spesifikasi perangkat lunak, perangkat keras, sistem operasi, dan arsitektur sistem. Pada implementasi kode program dilakukan implementasi kelas-kelas dan algoritma-algoritma yang ada pada tahap perancangan. Implementasi basis data menghasilkan diagram *Physical Data Model (PDM)*. Pada implementasi antar muka dihasilkan antarmuka untuk sistem.
4. Berdasarkan hasil pengujian yang telah dilakukan, Sistem Pengelolaan Pendistribusian Dana Bantuan Sekolah ini dapat melakukan pengelolaan terhadap data bantuan sekolah. Dalam pengujian unit terdapat 3 method utama yang valid. Pada pengujian integrasi pada 3 pasang method dari class controller dan class model didapatkan valid,

dan berjalan sesuai dengan fungsinya. Pada pengujian validasi dari 29 usecase adalah valid, semua fitur bisa berjalan sesuai dengan kebutuhan. Sedangkan pada pengujian *compatibility* menghasilkan bahwa sistem ini dapat berjalan atau digunakan pada 8 *web browser* yang berbeda.

7.2 Saran

Saran yang diberikan untuk pengembangan sistem aplikasi pengelolaan soal sebagai aplikasi untuk mengelola soal-soal ini selanjutnya antara lain:

1. Sistem dapat ditambahkan fitur untuk pembangunan sekolah, pengembangan profesi guru, membantu peserta didik miskin yang belum menerima bantuan dana lain, dan beasiswa siswa-siswi berprestasi.
2. Sistem dapat ditambahkan fitur untuk penentuan dana bantuan melalui sistem serta proses pencairan dana dan penggunaan dana bantuan sekolah dapat dilakukan dari sistem.



DAFTAR PUSTAKA

- Arief, M. Rudianto. 2011. *Pemrograman Web Dinamis Menggunakan PHP dan Mysql*. Yogyakarta: ANDI.
- Awaludin dan Eki Saputra. 2016. *Sistem Informasi Manajemen Sarana Prasarana Sekolah*. Riau: UIN Suska Riau.
- Pressman, R.S. 2010. *Software Engineering : A Practitioner's Approach*. 7th Ed. New York: McGraw-Hill.
- Menteri Pendidikan Nasional. 2007. *Standar Sarana dan Prasarana Sekolah/Madrasah Pendidikan Umum*. Jakarta.
- Republik Indonesia. 2003. *Undang – Undang No. 20 Tentang Sistem Pendidikan Nasional*. Jakarta: Sekretariat Negara.
- Rumbaugh, James. 2010. *Unified Modeling Language Reference Manual, The (2nd Edition)*. Austria: Graz.
- "Sarana dan Prasarana Pendidikan tahun 2017 Dinas Pendidikan dan Kebudayaan Kota Serang", <https://ppid.serangkota.go.id/sarana-dan-prasarana-pendidikan-tahun-2017-dinas-pendidikan-dan-kebudayaan-kota-serang> (diakses pada tanggal 27 November 2018)
- Satzinger, J. W., Jackson, R. B., and Burd. 2005. *Object-Oriented Analysis and Design with the Unified Process*. Boston: Course Technology.
- "Sekolah Mau Ambruk, Siswa SD di Banten Dievakuasi ke Tenda", <https://news.detik.com/berita/d-3700589/sekolah-mau-ambruk-siswa-sd-di-banten-dievakuasi-ke-tenda> (Diakses pada tanggal 11 Januari 2018)
- Sommerville, I. 2011. *Software Engineering(9th Edition)*. Pearson.
- "UML Use Case Diagram" *Use Case Diagram Description*, <http://www.uml-diagrams.org/use-case-diagrams.html> (diakses pada tanggal 11 Januari 2018).
- "UML 2 Class Diagrams: An Agile Introduction" *Agile Modeling*, <http://www.agilemodeling.com/artifacts/ClassDiagram.htm> (diakses 11 Juli 2018).
- "Use-case diagram in UML modeling", https://www.ibm.com/support/knowledgecenter/SS8PJ7_9.1.1/com.ibm.xtools.modeler.doc/topics/cucd.html (diakses pada tanggal 8 November 2018).