



FINAL PROJECT - KS09 1336

**DEVELOPMENT OF STATE FINANCIAL
MANAGEMENT INFORMATION SYSTEM
PAYMENT MODULE USING JAVA SERVER PAGES
AND POSTGRESQL TECHNOLOGY**

NOVAN ANDRE VALEN
NRP 5209 108 709

Supervisor
Dr.Eng. Febriliyan Samopa, S.Kom., M.Kom
Alwi, S.E

INFORMATION SYSTEM DEPARTMENT
Information Technology Faculty
Institut Teknologi Sepuluh Nopember
Surabaya 2012

**PENGEMBANGAN SISTEM INFORMASI
PENGELOLAAN KEUANGAN NEGARA MODUL
PEMBAYARAN MENGGUNAKAN TEKNOLOGI
JAVASERVER PAGES DAN POSTGRESQL**

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Jurusan Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

NOVAN ANDRE VALEN
5209 108 709

Surabaya, 2012

**KETUA
JURUSAN SISTEM INFORMASI**

Dr.Eng. Febriliyan Samopa, S.Kom., M.Kom
NIP. 197302191998021001



**PENGEMBANGAN SISTEM INFORMASI
PENGELOLAAN KEUANGAN NEGARA MODUL
PEMBAYARAN MENGGUNAKAN TEKNOLOGI
JAVASERVER PAGES DAN POSTGRESQL**

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Jurusan Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh :

NOVAN ANDRE VALEN

5209 108 709

Disetujui Tim Penguji :

Tanggal Ujian : 16 Juli 2012
Periode Wisuda : September 2012

Dr. Eng. Febriliyan Samopa, S.Kom, M.Kom

(Pembimbing I)

Alwi, S.E

(Pembimbing II)

Mudjahiddin, ST, MT

(Penguji 1)

Radityo Prasetyanto W, S.Kom, M.Kom

(Penguji 2)

**PENGEMBANGAN SISTEM INFORMASI
PENGELOLAAN KEUANGAN NEGARA MODUL
PEMBAYARAN MENGGUNAKAN TEKNOLOGI
JAVA SERVER PAGES DAN POSTGRESQL**

Nama Mahasiswa : Novan Andre Valen
NRP : 5209 108109
Jurusan : Sistem Informasi FTIf-ITS
Dosen Pembimbing : Dr. Eng. Febriliyan Samopa, S.Kom,
M.Kom
Alwi, SE.

Abstrak

Sistem informasi merupakan salah satu pendukung dalam mencapai tujuan organisasi. Tidak terkecuali dengan Direktorat Jenderal Perbendaharaan (DJPB) yang mempunyai fungsi sebagai pelaksana kebijakan perbendaharaan. Dalam mewujudkan pelaksanaan anggaran yang berbasis kinerja, maka DJPB menerapkan sistem informasi untuk membantu dalam menatausahakan Anggaran Pendapatan dan Belanja Negara (APBN). Selain itu DJPB juga mendukung pelaksanaan Satuan Kerja sebagai Kuasa Pengguna Anggaran dengan mengembangkan berbagai aplikasi yang mempermudah pelaksanaan APBN. Saat ini terdapat banyak aplikasi yang dikembangkan oleh DJPB. Namun, aplikasi tersebut dikembangkan dengan arsitektur yang terpisah meskipun mempunyai keterkaitan satu sama lain. Hal tersebut berakibat pada duplikasi proses bisnis yang ada sehingga mengurangi efisiensi dan efektivitas kinerja Satuan Kerja. Karena itu dibutuhkan suatu sistem informasi yang terintegrasi untuk menghasilkan kinerja yang lebih baik dan transparan.

Melalui Tugas Akhir ini penulis mengembangkan Sistem Informasi Pengelolaan Keuangan Negara Modul Pembayaran dengan menggunakan teknologi JavaServer Pages dan

PostgreSQL. Modul pembayaran mencakup proses penerbitan Surat Perintah Membayar (SPM). Sedangkan *JavaServer Pages* merupakan platform yang berguna dalam mengembangkan aplikasi berbasis web yang dapat menampilkan konten dinamis dengan baik dan *PostgreSQL* sebagai pendukung sistem manajemen basis data agar informasi yang dihasilkan dapat ditampilkan dengan baik.

Hasil dari penelitian ini merupakan aplikasi yang berguna bagi Satuan Kerja untuk menerbitkan Surat Perintah Membayar sebagai dasar pembayaran dan pencairan dana APBN.

Kata Kunci : Surat Perintah Membayar, APBN, DJPB, *JavaServer Pages*, *PostgreSQL*

**DEVELOPMENT OF STATE FINANCIAL
MANAGEMENT INFORMATION SYSTEM PAYMENT
MODULE USING JAVA SERVER PAGES AND
POSTGRESQL TECHNOLOGY**

Name : Novan Andre Valen
NRP : 5209 108 709
Department : Information System FTIf-ITS
Supervisor : Dr. Eng. Febriliyan Samopa, S.Kom,
M.Kom
Alwi, SE.

Abstract

Information system is one of the proponents in achieving organizational goals. No exception to the Directorate General of Treasury (DJPB) which has the function as the executor of the treasury policy. In realizing the implementation of performance-based budgeting, the DJPB implement information systems to assist in administering the State Budget (APBN). In addition DJPB also supports the implementation of the Budget User Authority (Satker) to develop a variety of applications that simplify the implementation of APBN. Currently there are many applications developed by DJPB. However, the application is developed with separate architectures despite having relation to one another. This results in duplication of existing business processes, thereby reducing the efficiency and effectiveness of the performance of Satker. Therefore it needs an integrated information system to produce better performance and transparency.

Through this Final Project, author would make the State Financial Management Information System Payment module using JavaServer Pages technology and PostgreSQL. Payment module covers the process of issuance of Payment Order (SPM). While Java is a platform that is useful in developing web-based

application that can display dynamic content as well and the PostgreSQL database management system support so that the resulting information can be displayed properly.

The results of this study is a useful application for the Unit to issue a Letter of Payment Order as the basis for payment and disbursement of budget funds.

Keywords : Surat Perintah Membayar, APBN, DJPB, JavaServer Pages, PostgreSQL

KATA PENGANTAR

Segala puji dan syukur penulis ucapkan pada Allah SWT yang telah memberikan kekuatan bagi penulis sehingga dapat menyelesaikan laporan tugas akhir dengan judul “PENGEMBANGAN SISTEM INFORMASI PENGELOLAAN KEUANGAN NEGARA MODUL PEMBAYARAN MENGGUNAKAN TEKNOLOGI JAVA SERVER PAGES DAN POSTGRESQL”.

Tugas akhir ini tidak akan pernah terselesaikan tanpa dukungan dan bantuan dari berbagai pihak yang telah meluangkan waktu, jiwa dan pikirannya bagi penulis untuk menyelesaikan tugas akhir ini. Semoga karya tulis ini dapat memberikan manfaat bagi para pembaca dan memberikan kontribusi pada berbagai pihak. Khususnya Direktorat Jenderal Perbendaharaan. Pada kesempatan ini penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada :

- Bapak, ibu, serta keluarga penulis yang selalu memberikan *dukungan dan kepercayaan*.Terima kasih atas doa dan dukungannya yang tiada henti.
- Bapak Febriliyan Samopa, selaku dosen pembimbing I, dan Bapak Alwi, selaku dosen pembimbing II, yang telah meluangkan waktu dan pikiran di tengah kesibukan beliau untuk membimbing dan mengarahkan penulis dalam mengerjakan tugas akhir ini.
- Seluruh anggota tim tugas akhir SIAPKAN yang telah berjuang bersama, Alfie, Hari, Rizal, Yoga, Wahyudiono serta Mas ali.
- Siska Sania Putri, atas segala doa dan dukungan yang telah diberikan selama ini.
- Segenap keluarga Bendits yang telah berjuang bersama-sama serta saling mendukung selama masa tugas belajar ini.

- Direktorat Jenderal Perbendaharaan yang telah memberikan penulis kepercayaan besar dalam menjalankan tugas belajar selama ini.
- Jurusan Sistem Informasi ITS Surabaya yang selalu memberikan dukungan serta pembelajaran.
- Manchester United 1998/1999, yang mengajarkan penulis untuk tidak pernah menyerah hingga akhir.
- Jihad, Miftah serta Yohanes #teamgembel, yang telah memberikan inspirasi serta semangat melalui perjalanan yang mengesankan.
- Pihak-pihak lain yang telah mendukung adanya tugas akhir ini.

Penulis menyadari bahwa masih banyak kekurangan pada tugas tugas akhir ini. Karena itu penulis mohon maaf atas segala kekurangan dan kekeliruan yang ada di dalam tugas akhir ini. Penulis membuka pintu selebar-lebarnya bagi pihak-pihak yang ingin memberikan kritik dan saran bagi penulis untuk menyempurnakan tugas akhir ini. Semoga tugas akhir ini dapat bermanfaat bagi seluruh pembaca.

Surabaya, Juli 2012

Penulis

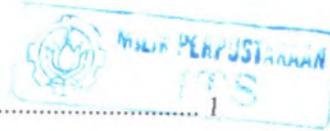
DAFTAR ISI

Abstrak.....	iii
Kata Pengantar.....	vii
Daftar Isi	ix
Daftar Gambar	xv
Daftar Tabel.....	xvii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang Masalah.....	1
1.2. Perumusan Masalah.....	6
1.3. Batasan Masalah.....	6
1.4. Tujuan	7
1.5. Relevansi atau Manfaat	7
1.6. Sistematika Penulisan.....	8
BAB II TINJAUAN PUSTAKA	9
2.1. Pembayaran APBN.....	9
2.2. Prosedur Penerbitan SPM.....	10
2.3. JavaServer Pages	12
2.4. Java Database Connectivity	14
2.5. Servlet.....	14
2.6. PosgtreSQL.....	17
BAB III METODE PENELITIAN	19
3.1. Studi Literatur.....	20
3.2. Identifikasi Kebutuhan.....	20
3.3. Desain Aplikasi.....	20
3.4. Pembuatan Aplikasi	20
3.5. Integrasi Aplikasi.....	21
3.6. Uji Coba Aplikasi	21
3.7. Migrasi.....	21
3.8. Pembuatan laporan.....	21
BAB IV ANALISIS DAN DESAIN SISTEM	23

4.1. Analisis Kebutuhan Sistem	23
4.1.1. Kondisi Sistem Saat Ini	24
4.1.2. Karakteristik Pengguna.....	24
4.1.3. Kebutuhan Modul Pembayaran	25
4.2. Arsitektur Aplikasi SIAPKAN	26
4.2.1. Arsitektur Desain	26
4.2.2. Arsitektur Teknis	30
4.3. Alur Kerja Modul Pembayaran.....	32
4.3.1. Perekaman Data Kontrak.....	33
4.3.2. Perekaman Data Gaji	33
4.3.3. Perekaman Data SPP	34
4.3.4. Cetak SPP	35
4.3.5. Persetujuan dan Pembatalan SPM	35
4.3.6. Cetak SPM.....	35
4.3.7. Pengiriman Arsip Data Komputer SPM	36
4.4. Domain Model.....	36
4.5. Use Case Diagram	38
4.6. Robustness Diagram	39
4.7. Sequence Diagram	40
4.8. Class Diagram.....	41
4.9. Test Case.....	41
4.10. GUI Story Board.....	42
BAB V IMPLEMENTASI DAN UJI COBA SISTEM.....	59
5.1. Lingkungan Implementasi	59
5.2. Konfigurasi	60
5.2.1. Pembuatan Struktur Direktori Project.....	60
5.2.2. Konfigurasi File.....	61

5.3. Penulisan Kode Program	64
5.3.1. Package 'pembayaran.entity'	64
5.3.2. Package 'pembayaran.model'	65
5.3.3. Package 'pembayaran.servlet'	66
5.3.4. Pembuatan View	67
5.4. Integrasi Modul	71
5.5. Uji Coba	72
5.6. Analisis Uji Coba	73
BAB VI KESIMPULAN DAN SARAN	75
6.1. Kesimpulan	75
6.2. Saran	75
DAFTAR PUSTAKA	77
Lampiran A DIAGRAM DAN DESKRIPSI USE CASE	1
A.1. Diagram Use Case	2
A.2. Deskripsi Use Case Rekam Kontrak	3
A.3. Deskripsi Use Case Ubah Kontrak	4
A.4. Deskripsi Use Case Hapus Kontrak	5
A.5. Deskripsi Use Case Rekam Gaji	6
A.6. Deskripsi Use Case Ubah Gaji	7
A.7. Deskripsi Use Case Hapus Gaji	9
A.8. Deskripsi Use Case Rekam SPP	10
A.9. Deskripsi Use Case Ubah SPP	11
A.10. Deskripsi Use Case Hapus SPP	12
A.11. Deskripsi Use Case Setuju SPM	13
A.12. Deskripsi Use Case Batal SPM	14
A.13. Deskripsi Use Case Cetak SPP	15
A.14. Deskripsi Use Case Cetak SPM	16
A.15. Deskripsi Use Case Rekam Nomor SP2D	17
A.16. Deskripsi Use Case Ubah Nomor SP2D	18
A.17. Deskripsi Use Case Hapus Nomor SP2D	19
A.18. Deskripsi Use Case Kirim SPM	20
Lampiran B DIAGRAM ROBUSTNESS ANALYSIS	1

B.1. Diagram Rekam Kontrak Robustness	3
B.2. Diagram Ubah Kontrak Robustness.....	4
B.3. Diagram Hapus Kontrak Robustness	5
B.4. Diagram Rekam Gaji Robustness	6
B.5. Diagram Ubah Gaji Robustness	7
B.6. Diagram Hapus Gaji Robustness	8
B.7. Diagram Rekam SPP Robustness	9
B.8. Diagram Ubah SPP Robustness	10
B.9. Diagram Hapus SPP Robustness.....	11
B.10. Diagram Setuju SPM Robustness	11
B.11. Diagram Batal SPM Robustness	12
B.12. Diagram Cetak SPP Robustness	12
B.13. Diagram Cetak SPM Robustness	13
B.14. Diagram Rekam Nomor SP2D Robustness.....	13
B.15. Diagram Ubah Nomor SP2D Robustness	14
B.16. Diagram Hapus Nomor SP2D Robustness.....	14
B.17. Diagram Kirim SPM Robustness	15
Lampiran C SEQUENCE DIAGRAM	1
C.1. Diagram Rekam Kontrak Sequence	3
C.2. Diagram Ubah Kontrak Sequence.....	4
C.3. Diagram Hapus Kontrak Sequence	5
C.4. Diagram Rekam Gaji Sequence	6
C.5. Diagram Ubah Gaji Sequence	7
C.6. Diagram Hapus Gaji Sequence	8
C.7. Diagram Rekam SPP Sequence	9
C.8. Diagram Ubah SPP Sequence	10
C.9. Diagram Hapus SPP Sequence.....	11
C.10. Diagram Setuju SPM Sequence	12
C.11. Diagram Batal SPM Sequence	13
C.12. Diagram Cetak SPP Robustness	14
C.13. Diagram Cetak SPM Sequence	15
C.14. Diagram Rekam Nomor SP2D Sequence	16
C.15. Diagram Ubah Nomor SP2D Sequence	17
C.16. Diagram Hapus Nomor SP2D Sequence.....	18
C.17. Diagram Kirim SPM Sequence.....	19



Lampiran D CLASS DIAGRAM	1
Lampiran E TEST CASE.....	1
E.1. Test Case Rekam Kontrak.....	3
E.2. Test Case Ubah Kontrak	4
E.3. Test Case Hapus Kontrak.....	5
E.4. Test Case Rekam Gaji.....	6
E.5. Test Case Ubah Gaji.....	7
E.6. Test Case Hapus Gaji	8
E.7. Test Case Rekam SPP	9
E.8. Test Case Ubah SPP.....	10
E.9. Test Case Hapus SPP	11
E.10. Test Case Setuju SPM.....	12
E.11. Test Case Batal SPM.....	13
E.12. Test Case Cetak SPP	14
E.13. Test Case Cetak SPM.....	14
E.14. Test Case Rekam Nomor SP2D	15
E.15. Test Case Ubah Nomor SP2D.....	16
E.16. Test Case Hapus Nomor SP2D	16
E.17. Test Case Kirim SPM.....	17

DAFTAR GAMBAR

Gambar 2.1 Alur Penerbitan SPM.....	12
Gambar 2.2 Arsitektur Model 2.....	13
Gambar 3.1 Diagram Alur Metodologi.....	19
Gambar 4.1 ICONIX Process	23
Gambar 4.2 Arsitektur Global Aplikasi SIAPKAN.....	26
Gambar 4.3 Hubungan Antar Modul Aplikasi SIAPKAN	29
Gambar 4.4 Alur Kerja Modul Pembayaran.....	33
Gambar 4.5 Arsitektur Desain aplikasi SIAPKAN	31
Gambar 4.6 Domain Model SIAPKAN Modul Pembayaran.....	37
Gambar 4.7 Use Case Diagram SIAPKAN Modul Pembayaran	38
Gambar 4.8 Robustness Diagram Rekam Kontrak.....	40
Gambar 4.9 Robustness Diagram Rekam Kontrak.....	41
Gambar 4.10 RUH Kontrak.....	42
Gambar 4.11 Rekam Kontrak	43
Gambar 4.12 Rekam Kontrak – Peringatan Error.....	43
Gambar 4.13 Ubah Kontrak.....	44
Gambar 4.14 Konfirmasi Hapus	44
Gambar 4.15 RUH Gaji.....	45
Gambar 4.16 Rekam Gaji	45
Gambar 4.17 Rekam Gaji – Peringatan Error.....	46
Gambar 4.18 Ubah Gaji.....	46
Gambar 4.19 RUH SPP	47
Gambar 4.20 Rekam SPP	48
Gambar 4.21 Rekam SPP – Peringatan Error	48
Gambar 4.22 Ubah SPP.....	49
Gambar 4.23 Setuju SPM	49
Gambar 4.24 Konfirmasi Setuju SPM.....	50
Gambar 4.25 Batal SPM.....	51
Gambar 4.26 Konfirmasi Batal SPM.....	51
Gambar 4.27 Cetak SPP	52
Gambar 4.28 Konfirmasi Cetak SPP	52
Gambar 4.29 Cetak SPM.....	53
Gambar 4.30 Konfirmasi Cetak SPM.....	53

Gambar 4.31 RUH Nomor SP2D	54
Gambar 4.32 Rekam Nomor SP2D.....	55
Gambar 4.33 Rekam Nomor SP2D – Peringatan Error	55
Gambar 4.34 Ubah Nomor SP2D	56
Gambar 4.35 Kirim SPM.....	56
Gambar 4.36 Konfirmasi Kirim SPM.....	57
Gambar 5.1 Struktur Direktori.....	61
Gambar 5.2 Konfigurasi file build.xml.....	62
Gambar 5.3 Konfigurasi file project.properties	62
Gambar 5.4 Konfigurasi file web.xml	63
Gambar 5.5 Konfigurasi file persistence.xml	63
Gambar 5.6 Konfigurasi file context.xml	63
Gambar 5.7 Potongan kode Gaji.java	64
Gambar 5.8 Potongan kode DaftarGaji.java.....	65
Gambar 5.9 Potongan kode GajiController.java.....	66
Gambar 5.10 Potongan kode Gajiservlet.java	67
Gambar 5.11 Potongan kode rekam_spp.jsp	68
Gambar 5.12 Potongan kode validasi rekam_spp.jsp	68
Gambar 5.13 Halaman Daftar SPP	69
Gambar 5.14 Halaman rekam SPP	69
Gambar 5.15 Halaman ubah SPP.....	70
Gambar 5.16 Pesan konfirmasi hapus data.....	70

DAFTAR TABEL

Table 5.1 Spesifikasi Perangkat Keras dan Lunak	59
Table 5.2 Teknologi yang digunakan	59
Table 5.3 Konfigurasi File.....	61
Table 5.4 Hasil Uji Coba Fungsionalitas	72

BAB I

PENDAHULUAN

Pada bab ini, akan dijelaskan tentang Latar Belakang Masalah, Perumusan Masalah, Batasan Masalah, Tujuan Tugas Akhir, dan Relevansi atau Manfaat Tugas Akhir.

1.1.Latar Belakang Masalah

Saat ini hampir seluruh instansi pemerintah di Indonesia telah menerapkan sistem informasi sebagai pendukung dalam mencapai tujuan utama organisasi. Sistem informasi digunakan sebagai sarana untuk meningkatkan kinerja sumber daya yang ada di setiap instansi pemerintah. Selain itu sistem informasi juga digunakan dalam memperoleh data, serta mengolahnya agar menjadi suatu informasi yang dapat didistribusikan sebagai penunjang kegiatan operasional serta dasar pengambilan keputusan strategis pada organisasi.

Direktorat Jenderal Perbendaharaan (DJPB) merupakan salah satu jajaran dalam Departemen Keuangan yang dalam tugasnya mempunyai fungsi untuk melaksanakan kebijakan di bidang Perbendaharaan Negara sesuai dengan peraturan perundangan yang berlaku, dalam hal ini adalah Anggaran Pendapatan Belanja Negara (APBN). Dengan visi untuk menjadi pengelola perbendaharaan negara yang profesional, transparan guna mewujudkan manajemen keuangan pemerintah yang efektif dan efisien, DJPB mempunyai beberapa misi, diantaranya adalah :

- Menciptakan fungsi pelaksanaan anggaran yang efektif
- Mewujudkan pengelolaan kas negara yang efisien dan optimal
- Menciptakan sistem manajemen investasi yang tepat sasaran
- Mewujudkan pengelolaan keuangan Badan Layanan Umum yang fleksibel, efektif dan akuntabel



- Mewujudkan pengelolaan keuangan negara yang akuntabel, transparan, tepat waktu dan akurat
- Mewujudkan dukungan teknis perbendaharaan yang handal, terintegrasi, terotomatisasi, dan mudah diterapkan
- Menyempurnakan proses bisnis sistem perbendaharaan sesuai best practice
- Melaksanakan pemberdayaan dan integrasi seluruh sumber daya organisasi secara optimal.

Dalam menjalankan tugasnya DJPB sudah menerapkan sistem informasi sebagai pendukung dalam menatausahakan APBN. Sistem informasi yang ada pada DJPB membantu kegiatan operasional 181 unit Kantor Pelayanan Perbendaharaan Negara serta 30 Kantor Wilayah Direktorat Jenderal Perbendaharaan untuk melayani lebih dari tiga puluh ribu Satuan Kerja dalam melaksanakan pencairan dana serta pengelolaan penerimaan negara. Data transaksi pencairan dana serta penerimaan negara tersebut tersimpan dan diolah di setiap KPPN sebagai unit terdepan dalam menjalankan tugas sebagai Bendahara Umum Negara.

Untuk mendukung pelaksanaannya, DJPB mengembangkan aplikasi yang berguna sebagai alat bantu Satuan Kerja dalam menatausahakan pencairan dana APBN. Sejalan dengan Undang Undang Nomor 17 tahun 2003 tentang Keuangan Negara yaitu mewujudkan pelaksanaan APBN secara akuntabel dan transparan, maka DJPB mengembangkan aplikasi yang digunakan oleh Satuan Kerja sebagai alat bantu pelaksanaan pengelolaan keuangan negara yang memenuhi prinsip tepat waktu serta disusun dengan mengikuti standar akuntansi pemerintah yang telah diterima secara umum. Saat ini terdapat beberapa aplikasi yang telah dikembangkan, yaitu:

1. Aplikasi Rencana Kerja Anggaran Kementerian/Lembaga (RKAKL), digunakan sebagai pendukung Satuan Kerja dalam penyusunan rencana kerja dan anggaran tingkat kementerian/lembaga.

2. Aplikasi Daftar Isian Pelaksanaan Anggaran (DIPA), digunakan sebagai pendukung Satuan Kerja dalam penyusunan Daftar Isian Pelaksanaan Anggaran untuk pelaksanaan pencairan dana.
3. Aplikasi Surat Perintah Membayar (SPM) digunakan sebagai pendukung Satuan Kerja dalam pembuatan Surat Perintah Membayar sebagai alat pencairan dan pembayaran kegiatan yang dilakukan selama tahun anggaran berjalan.
4. Aplikasi Sistem Informasi Pembukuan Bendahara (SISKA) digunakan sebagai pendukung Satuan Kerja dalam proses pelaksanaan teknis pencairan dana serta pembukuan dan penatausahaan anggaran.
5. Aplikasi Persediaan digunakan sebagai pendukung Satuan Kerja dalam proses pengaturan dan pengelolaan Persediaan pada masing-masing instansi.
6. Aplikasi Sistem Informasi Manajemen Akuntansi Barang Milik Negara (SIMAK BMN) digunakan sebagai pendukung Satuan Kerja dalam proses pengaturan dan pengelolaan Aset Tetap dan Barang Milik Negara.
7. Aplikasi Sistem Akuntansi Kuasa Pengguna Anggaran (SAKPA) digunakan sebagai pendukung Satuan Kerja dalam proses pelaporan keuangan dalam tahun anggaran berjalan.

Seluruh aplikasi tersebut saling terkait namun dibangun dan dikembangkan dengan arsitektur yang terpisah. Dengan demikian, masih banyak terdapat proses bisnis yang berulang sehingga mengurangi efisiensi dan efektivitas dalam pelaksanaannya. Selain itu dengan sistem yang terpisah memberikan implikasi terjadinya perbedaan data antar aplikasi yang ada serta dimungkinkan untuk melakukan manipulasi data untuk kepentingan tertentu. Karena alasan itulah dibutuhkan pembuatan *sistem informasi yang terintegrasi sebagai upaya memodernisasi pengelolaan keuangan negara di bidang pelaksanaan anggaran, pada pelaksanaan anggaran akan dilakukan modernisasi*

mekanisme pembayaran untuk meningkatkan transparansi, efektivitas dan efisiensi penggunaan anggaran dan perbaikan pelayanan publik. Integrasi sistem tersebut mencakup prinsip sebagai berikut :

- Arsitektur teknologi yang mempunyai standar yang baik
- Mengoptimalkan penggunaan sumber daya yang terbatas agar lebih efisien.

Sistem Informasi Pengelolaan Keuangan Negara (SIAPKAN) merupakan suatu solusi untuk menjawab permasalahan dengan menyatukan seluruh aplikasi yang terkait dengan proses pelaksanaan anggaran. SIAPKAN akan menggunakan *single database* agar data yang diolah dapat disajikan dengan cepat, tepat dan akurat serta meningkatkan efisiensi dan efektivitas dalam proses bisnis yang ada. SIAPKAN akan menggunakan teknologi JavaServer Pages yang menawarkan beberapa keunggulan seperti paradigma *Object Oriented Programming* (OOP) yang menawarkan efisiensi pemrograman, pengembangan serta pemeliharannya. Untuk basis data menggunakan PostgreSQL karena mendukung kinerja yang baik pada kondisi jumlah klien yang banyak. Keuntungan ini sesuai dengan kondisi Satuan Kerja yang dilayani oleh DJPB. Terdapat tujuh modul pada SIAPKAN, yaitu :

1. Modul Penganggaran, merupakan representasi dari aplikasi RKA-KL dan DIPA.
2. Modul Pembayaran, merupakan representasi dari aplikasi SPM.
3. Modul Bendahara, merupakan representasi dari aplikasi SISKA.
4. Modul Aset Tetap, merupakan representasi dari aplikasi SIMAK-BMN
5. Modul Persediaan, merupakan representasi dari aplikasi Persediaan.
6. Modul Pelaporan, merupakan representasi dari aplikasi SAKPA.

7. *Modul Administrasi*, merupakan modul yang digunakan untuk mengelola dan mengatur akses pada modul-modul yang lain sesuai dengan tingkat kewenangannya. Selain itu juga akan memelihara seluruh referensi yang digunakan dalam setiap modul.

Modul Pembayaran merupakan modul yang digunakan Satuan Kerja untuk melaksanakan pencairan serta pembayaran dana APBN. Dalam integrasinya, Modul Pembayaran membutuhkan masukan dari Modul Anggaran berupa Daftar Isian Pelaksanaan Anggaran (DIPA) atau dokumen pelaksanaan anggaran lain yang dipersamakan. Selain itu juga mendapatkan input dari Modul Bendahara berupa informasi transaksi Uang Persediaan (UP), informasi Posting dan Closing dari Modul Pelaporan serta informasi referensi dari Modul Administrasi. Dengan demikian, Satuan Kerja dapat dengan mudah menggunakan Modul Pembayaran untuk menghasilkan output berupa Surat Perintah Membayar. Sesuai dengan Peraturan Direktorat Jenderal Perbendaharaan Nomor PER-66/PB/2005 tentang Mekanisme Pelaksanaan Pembayaran Atas Beban Anggaran Pendapatan dan Belanja Negara, dokumen anggaran tersebut merupakan media / alat bayar yang diterbitkan oleh Kuasa Pengguna Anggaran dalam hal ini Satuan Kerja untuk mencairkan dana yang bersumber dari Daftar Isian Pelaksanaan Anggaran.

Selama ini, Satuan Kerja mengalami permasalahan dalam menggunakan aplikasi SPM, yaitu perekaman / transfer data DIPA, ketidakcocokan data realisasi dan pelaporan yang ada. Untuk perekaman / transfer data DIPA biasanya disebabkan banyaknya data yang akan dimasukkan serta terkadang kesulitan dalam melakukan transfer data yang berasal dari Aplikasi DIPA. Selain itu perekaman data secara manual dapat memicu manipulasi yang berujung pada tindakan kriminal. Sedangkan ketidakcocokan data realisasi dan pelaporan sering terjadi karena *output daftar realisasi Surat Perintah Membayar* merupakan masukan bagi aplikasi SAKPA. Pada aplikasi SAKPA, Satuan Kerja juga melakukan perekaman data realisasi SPM. Tingkat

kesalahan serta inkonsistensi data yang ada dapat menyebabkan akurasi pelaporan dan pertanggungjawaban keuangan berkurang. Dengan adanya aplikasi SIAPKAN yang terintegrasi maka dapat dimanfaatkan untuk menghasilkan kinerja dan kualitas layanan yang mempunyai standar tinggi di lingkungan DJPB. Selain itu dengan adanya sistem yang terintegrasi diharapkan sebagai fasilitator dalam pembangunan proses bisnis yang efektif dan efisien dalam sistem perbendaharaan yang ada. Lebih lanjut untuk Modul Pembayaran, diharapkan dapat mempermudah Satuan Kerja dalam melaksanakan pencairan dana APBN sesuai dengan Service Level Agreement yang ada (SLA) yaitu proses pencairan dana yang cepat, tepat dan akurat. Saat ini proses pencairan dana berlangsung hanya dengan waktu maksimal satu jam. Diharapkan dengan adanya SIAPKAN dapat terus memaksimalkan proses tersebut dan menjadi lebih baik.

1.2.Perumusan Masalah

Permasalahan yang akan terselesaikan dalam tugas akhir ini adalah:

1. Bagaimana membangun aplikasi yang terintegrasi untuk Sistem Informasi Pengelolaan Keuangan Negara Modul Pembayaran dengan menggunakan teknologi JavaServer Pages dan PostgreSQL?
2. Bagaimana membangun dan mendesain modul Pembayaran agar sesuai dengan proses bisnis serta dapat memudahkan pengguna?

1.3.Batasan Masalah

Dari permasalahan yang telah disebutkan diatas, yang menjadi batasan masalah dalam tugas akhir ini adalah sebagai berikut :

1. Surat Perintah Membayar yang dimaksud hanya untuk pembayaran yang meliputi Belanja Pegawai, Belanja

Barang, Belanja Modal, Belanja Sosial dan Belanja Lain-Lain.

2. Surat Perintah Membayar hanya untuk jenis pembayaran permintaan uang muka Persediaan, penggantian Uang Persediaan dan Langsung.
3. Satuan Kerja yang dimaksud merupakan Satuan Kerja instansi pemerintah pusat, bukan pemerintah daerah.
4. Modul Pembayaran akan digunakan oleh Satuan Kerja sebagai mitra kerja Direktorat Jenderal Perbendaharaan.

1.4.Tujuan

Tujuan dari tugas akhir ini adalah membangun aplikasi integrasi Sistem Informasi Pengelolaan Keuangan Negara Modul Pembayaran.

1.5.Relevansi atau Manfaat

Manfaat yang diberikan oleh tugas akhir ini adalah sebagai berikut :

1. Modul Pembayaran dapat berguna dalam mempermudah Satuan Kerja untuk memproses pencairan dana dengan menerbitkan Surat Perintah Membayar serta melakukan penatausahaan Kontrak.
2. Modul Pembayaran dapat mengurangi duplikasi serta ketidakcocokan
3. Sistem informasi yang akan dibangun berguna untuk memfasilitasi proses bisnis yang efektif dan efisien dalam sistem perbendaharaan yang ada saat ini.
4. Meningkatkan akurasi data dan akuntabilitas laporan pertanggungjawaban.

1.6.Sistematika Penulisan

Sistematika penulisan buku tugas akhir dibagi menjadi 6 bab sebagai berikut.

BAB IPENDAHULUAN

Bab ini berisi latar belakang masalah, rumusan permasalahan, batasan masalah/ruang lingkup, tujuan, manfaat dan sistematika penulisan.

BAB IITINJAUAN PUSTAKA

Bab ini menjelaskan teori-teori dari berbagai pustaka yang digunakan sebagai acuan dalam penulisan buku tugas akhir ini.

BAB III

METODOLOGI

Bab ini membahas alur dan pengerjaan tugas akhir dari awal sampai selesainya tugas akhir.

BAB IV

PERANCANGAN DESAIN APLIKASI

Bab ini menjelaskan rancangan desain aplikasi yang dibuat berdasarkan kebutuhan sistem. Desain tersebut digunakan untuk pembangunan aplikasi pada tugas akhir ini.

BAB V

IMPLEMENTASI DAN UJI COBA SISTEM

Bab ini menjelaskan pembangunan aplikasi yang sesuai dengan desain. Selain itu, dijelaskan pula uji coba sistem dalam menjaga performa aplikasi.

BAB VI

KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan yang dapat diambil dari tugas akhir ini dan saran untuk kelanjutan sistem.

BAB II

TINJAUAN PUSTAKA

Untuk memudahkan pemahaman tentang apa yang akan dilakukan pada tugas akhir ini, berikut ini akan di paparkan tentang konsep dan teknologi apa saja yang akan digunakan atau di terapkan. Adapun penerapan teknologi yang akan dilakukan pada tugas akhir ini adalah sebagai berikut.

2.1. Pembayaran APBN

Perihal pelaksanaan APBN telah tertuang dalam Undang Undang Nomor 1 Tahun 2004 tentang Perbendaharaan Negara Pasal 17 ayat 1 dan 2, yaitu :

- Pengguna Anggaran/Kuasa Pengguna Anggaran, dalam hal ini Satuan Kerja melaksanakan kegiatan sebagaimana tersebut dalam dokumen pelaksanaan anggaran yang telah disahkan.
- Untuk keperluan pelaksanaan kegiatan sebagaimana tersebut dalam dokumen pelaksanaan anggaran, Pengguna Anggaran/Kuasa Pengguna Anggaran dalam hal ini Satuan Kerja berwenang mengadakan ikatan/perjanjian dengan pihak lain dalam batas anggaran yang telah ditetapkan.

Kemudian lebih lanjut dituangkan pada pasal 18 ayat 1, yaitu :

- Pengguna Anggaran/Kuasa Pengguna Anggaran berhak untuk menguji, membebaskan pada mata anggaran yang telah disediakan, dan memerintahkan pembayaran tagihan-tagihan atas beban APBN.

Untuk memerintahkan pembayaran terhadap tagihan-tagihan atas beban APBN dibutuhkan suatu dokumen pelaksanaan anggaran, yaitu Surat Perintah Membayar. Sesuai dengan Keputusan Menteri Keuangan Nomor 134 tahun 2005 tentang Pedoman Pembayaran Dalam Pelaksanaan Anggaran Pendapatan dan Belanja Negara Pasal 1, yang dimaksud dengan Surat Perintah

Membayar (SPM) adalah dokumen yang diterbitkan/digunakan oleh Pengguna Anggaran/Kuasa Pengguna Anggaran untuk mencairkan alokasi dana yang sumber dananya dari DIPA. SPM dapat dikategorikan menjadi beberapa jenis, yaitu :

- Surat Perintah Membayar Langsung (SPM-LS) adalah surat perintah membayar yang dikeluarkan oleh Pengguna Anggaran/Kuasa Pengguna Anggaran kepada :
 - a. Pihak Ketiga atas dasar perikatan atau surat keputusan
 - b. Bendahara Pengeluaran untuk Belanja Pegawai/Perjalanan.
- Surat Perintah Membayar Uang Persediaan (SPM-UP) adalah surat perintah membayar yang diterbitkan oleh Pengguna Anggaran/Kuasa Pengguna Anggaran, yang dananya digunakan sebagai uang Persediaan untuk membiayai kegiatan operasional kantor sehari-hari.
- Surat Perintah Membayar Penggantian Uang Persediaan (SPM-GU) adalah surat perintah membayar yang diterbitkan oleh Pengguna Anggaran/Kuasa Pengguna Anggaran dengan membebani DIPA, yang dananya dipergunakan untuk menggantikan Uang Persediaan yang telah dipakai.

2.2. Prosedur Penerbitan SPM

Prosedur penerbitan SPM telah tertuang dalam Peraturan Direktorat Jenderal Perbendaharaan Nomor PER-66/PB/2005 tentang Mekanisme Pelaksanaan Pembayaran Atas Beban Anggaran Pendapatan dan Belanja Negara Pasal 5. Setelah menerima Surat Permintaan Pembayaran (SPP), Satuan Kerja dalam hal ini diberikan kewenangan pada Pejabat Penerbit SPM menerbitkan SPM dengan rincian sebagai berikut :

1. Petugas penerima SPP memeriksa kelengkapan berkas SPP, mengisi check list kelengkapan berkas SPP, mencatatnya dalam buku pengawasan penerimaan SPP

dan membuat/ menandatangani tanda terima SPP berkenaan. Selanjutnya petugas penerima SPP menyampaikan SPP dimaksud kepada pejabat penerbit SPM.

2. Pejabat penerbit SPM melakukan pengujian atas SPP sebagai berikut :
 - a. Memeriksa secara rinci dokumen pendukung SPP sesuai dengan ketentuan yang berlaku.
 - b. Memeriksa ketersediaan pagu anggaran dalam DIPA untuk memperoleh keyakinan bahwa tagihan tidak melampaui batas pagu anggaran.
 - c. Memeriksa kesesuaian rencana kerja dan/atau kelayakan hasil kerja yang dicapai dengan indikator keluaran.
 - d. Memeriksa kebenaran atas hak tagih yang menyangkut antara lain :
 - i. Pihak yang ditunjuk untuk menerima pembayaran (nama orang/ perusahaan, alamat, nomor rekening dan nama bank)
 - ii. Nilai tagihan yang harus dibayar (kesesuaian dan/atau kelayakannya dengan prestasi kerja yang dicapai sesuai spesifikasi teknis yang tercantum dalam kontrak)
 - iii. Jadwal waktu pembayaran.
 - e. Memeriksa pencapaian tujuan dan/atau sasaran kegiatan sesuai dengan indikator keluaran yang tercantum dalam DIPA berkenaan dan/atau spesifikasi teknis yang sudah ditetapkan dalam kontrak

Setelah dilakukan pengujian terhadap SPP, Pejabat Penguji SPP dan Penandatangan SPM menerbitkan SPM-UP/ SPM-GUP/SPM-LS. Gambar 2.1 menunjukkan alur singkat penerbitan SPM.



Gambar 2.1 Alur Penerbitan SPM

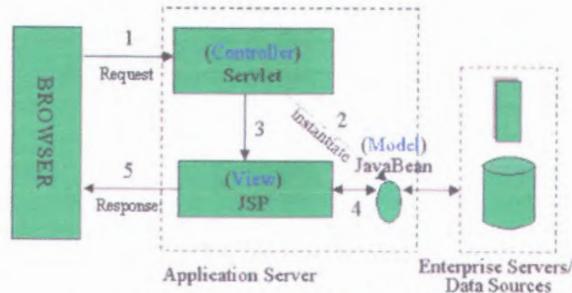
2.3. JavaServer Pages

JavaServer Pages (JSP) merupakan teknologi untuk mengembangkan halaman web yang berisi konten dinamik (Bergsten, 2004). JSP biasa digunakan untuk membangun aplikasi web. Sebuah halaman JSP dapat terdiri dari elemen seperti HTML dan *markup language* lainnya, sehingga dapat dikatakan sama dengan halaman web umumnya. JSP biasa digunakan untuk membangun aplikasi web. Sebuah halaman JSP dapat terdiri dari elemen seperti HTML dan *markup language* lainnya, sehingga dapat dikatakan sama dengan halaman web umumnya. Tetapi, sebuah halaman JSP dapat terdiri dari elemen khusus yang memungkinkan *server* memasukkan konten dinamis pada halaman tersebut. Aplikasi yang dikembangkan menggunakan arsitektur *client/server* Model 2 yang memisahkan antara konten dengan logika yang digunakan untuk memanipulasi konten tersebut. Model 2 terdiri dari tiga proses yang terpisah, yaitu (Seshadri, 1999):

1. *Browser*, berada pada sisi *client* yang menampilkan konten dinamis.
2. *Application Server*, menggunakan Servlet sebagai *controller* yang bekerja untuk menangani proses permintaan dari JSP lalu berkomunikasi dengan sumber

data lalu menyatukannya sebagai konten dinamis agar dapat diakses oleh JSP dan ditampilkan pada *Browser*.

3. *Data Source*, merupakan sistem manajemen data yang diperlukan untuk mengakses data dari *Application Server*.



Gambar 2.2 Arsitektur Model 2

Dengan menggunakan JavaServer Pages terdapat beberapa keuntungan, yaitu :

1. JSP bersifat *precompiled* sehingga proses pada server menjadi lebih efisien
2. Menggunakan paradigma HTTP sebagai standar dalam proses *request/response* sehingga baik untuk digunakan sebagai dasar aplikasi berbasis web
3. Dapat dijalankan pada semua *platform* (independen) dan dapat dibangun serta dijalankan dari semua *platform* server (*write once, run anywhere*)
4. Bersifat *free*, sehingga menekan biaya pengembangan aplikasi
5. Keamanan serta perlindungan terhadap aset teknologi informasi masa depan terjamin

Dapat dilakukan perbaikan atau perubahan secara cepat apabila dibutuhkan.

2.4. Java Database Connectivity

Java Database Connectivity (JDBC) merupakan kelas dari Java yang berfungsi untuk mengirimkan pernyataan SQL menuju Database Management System (DBMS) dan mengembalikannya pada suatu sistem aplikasi. JDBC memungkinkan suatu aplikasi dikembangkan tanpa menargetkan pada DBMS tertentu (interoperabilitas). Pengguna dapat memilih JDBC secara spesifik untuk menentukan target database yang diinginkan (Van Haecke, 2002).

JDBC menyediakan seperangkat metode yang memungkinkan koneksi pada database, *query* dan mengembalikan hasil. *Interface* JDBC merupakan interface antara suatu driver database tertentu dengan pengguna aplikasi berbasis Java. JDBC mempunyai beberapa komponen, yaitu :

- Aplikasi : pengguna memanggil metode JDBC untuk mengirimkan *SQL statement* menuju database lalu mengambil (*retrieve*) hasilnya.
- Driver Manager : berfungsi untuk memuat driver khusus pada aplikasi pengguna.
- Driver : berfungsi memproses pemanggilan metode JDBC, mengirimkan *SQL statement* menuju sumber data tertentu dan mengembalikan hasilnya pada aplikasi.

2.5. Servlet

Servlet merupakan kelas dari Java yang di desain untuk berinteraksi dengan konten dinamis yang dibutuhkan klien melalui suatu jaringan komunikasi (Perry, 2004). Servlet meneruskan permintaan dari suatu web *browser* atau HTTP

client menuju aplikasi atau database yang ada pada server. Beberapa fungsi yang dimiliki servlet yaitu (Hall, 2000) :

1. Membaca data/informasi yang dimasukkan/dikirimkan oleh klien. Data tersebut biasanya dimasukkan oleh user melalui *form* HTML pada halaman web.
2. Membaca data/informasi sisipan yang dikirimkan melalui HTTP *request*. Informasi tersebut biasanya berupa skema kompresi, *cookies*, *hostname* dan informasi lainnya.
3. Menghasilkan output. Servlet dapat berkomunikasi dengan database atau aplikasi pada server kemudian mengirimkan serta mengekstrak data yang berasal dari klien.
4. Mengirimkan data/informasi pada klien. Data yang semisal dokumen dapat dikirimkan dengan format yang bervariasi, termasuk teks, binari atau berupa dokumen yang terkompresi dalam format yang ditentukan.
5. Menentukan parameter HTTP *response*. Hal ini berguna bagi browser agar dapat mengetahui jenis dokumen yang dikirimkan, *caching*, manajemen cookies serta informasi lainnya.
6. Mengatur hasil yang ada pada sebuah dokumen. Hal ini dapat berupa informasi tambahan pada suatu halaman HTML.

Dengan fungsi-fungsi yang dimiliki oleh servlet, maka output yang dihasilkan bersifat dinamis. Selain itu terdapat beberapa kelebihan servlet dibandingkan dengan teknologi sebelumnya, yaitu :

1. *Handal*

Servlet mampu untuk berkomunikasi secara langsung dengan web server. Teknologi sebelumnya seperti CGI membutuhkan sebuah API server agar dapat berkomunikasi. Selain itu servlet membuat teknik *session tracking* menjadi lebih sederhana.

2. *Portabel*

Servlet ditulis dengan bahasa pemrograman Java yang mengikuti standar API sehingga dapat dijalankan oleh banyak web server.

3. *Aman*

Salah satu permasalahan yang dimiliki program CGI tradisional adalah program yang dijalankan melalui *shell* sebuah sistem operasi. Hal ini mengharuskan programmer untuk ekstra hati-hati dalam mengimplementasikan kode program terhadap karakter yang diperlakukan spesial oleh shell. Servlet terbebas dari permasalahan tersebut.

4. *Lengkap*

Servlet memiliki infrastruktur yang lengkap, baik dari sisi *session tracking*, *decoding* dan *parsing HTML*, pengaturan header, serta utilitas lainnya.

5. *Murah*

Servlet kompatibel dengan banyak web server yang murah bahkan tanpa biaya atau gratis.

6. Efisien

Servlet memproses HTTP request dalam jumlah banyak, namun hanya satu servlet class yang ditempatkan pada memori sehingga sangat efisien.

2.6. PostgreSQL

PostgreSQL merupakan salah satu sistem manajemen basis data yang berlisensi *open source*. PostgreSQL menggabungkan model relasional sebagai basis data dan mendukung penggunaan SQL *query language*. PostgreSQL mempunyai karakteristik performa yang baik dan dapat dijalankan pada berbagai *platform* serta mendukung berbagai penggunaan bahasa pemrograman (Matthew, et al., 2005).

PostgreSQL mempunyai sistem otorisasi yang baik serta fleksibel sehingga membuatnya aman untuk digunakan. Selain itu dengan dukungan teknis serta komunitas yang besar membuat PostgreSQL stabil dan minim dari *bug*. PostgreSQL dapat diandalkan untuk menampung data biner yang besar seperti foto atau video.

3.1.Studi Literatur

Studi literatur dalam pembuatan tugas akhir ini adalah dengan melakukan pembelajaran serta pemahaman pada literatur yang berkaitan dengan permasalahan yang ada. Literatur yang dipelajari berkaitan dengan Undang Undang serta Peraturan yang terkait dalam pembuatan aplikasi. Selain itu juga dipelajari bahasa pemrograman serta perangkat lain yang mendukung pengembangan aplikasi. Literatur didapatkan melalui media buku, halaman web, serta kumpulan peraturan perundangan.

3.2.Identifikasi Kebutuhan

Setelah memahami konsep dan teori yang mendukung pengembangan aplikasi SIAPKAN, maka pada tahap ini dilakukan identifikasi kebutuhan sistem. Identifikasi ini dilakukan dengan mempelajari sistem yang telah ada sebelumnya. Dari identifikasi tersebut akan didapatkan gambaran jelas tentang aplikasi SIAPKAN.

3.3.Desain Aplikasi

Pada tahap ini dilakukan desain dari sistem aplikasi yang akan dikembangkan. Perancangan dan desain sistem mencakup proses bisnis serta mengacu pada integrasi dengan modul lain. Perancangan dan desain sistem dilakukan dengan menggunakan *Unified Modeling Language (UML)*. *UML* adalah bahasa standar yang digunakan untuk proses dokumentasi, pembuatan spesifikasi dan membangun sebuah sistem perangkat lunak.

3.4.Pembuatan Aplikasi

Setelah proses desain selesai dengan baik maka dilanjutkan dengan tahap pembuatan aplikasi, khususnya Modul Pembayaran.

Sistem dikembangkan dengan menggunakan platform JavaServer Pages dan PostgreSQL sebagai basis data.

3.5.Integrasi Aplikasi

Pada tahap ini akan dilakukan integrasi terhadap Modul Pembayaran dengan keseluruhan modul yang telah dibangun dalam aplikasi SIAPKAN.

3.6.Uji Coba Aplikasi

Pada tahap ini akan dilakukan uji coba terhadap aplikasi yang telah dibangun. Hal ini bertujuan untuk melihat kesesuaian dengan desain yang telah dibuat dengan fungsionalitasnya serta mengetahui apakah terdapat kesalahan-kesalahan, baik secara tampilan informasi maupun kode program. Apabila ditemukan, maka akan dilakukan perbaikan pada aplikasi tersebut. Proses ini akan dilakukan sampai aplikasi telah sesuai dengan desain yang telah dibuat pada proses sebelumnya.

3.7.Migrasi

Pada tahap ini dilakukan migrasi atau perpindahan data dari aplikasi yang telah digunakan sebelumnya. Untuk Modul Pembayaran dilakukan migrasi data dari Aplikasi Surat Perintah Membayar.

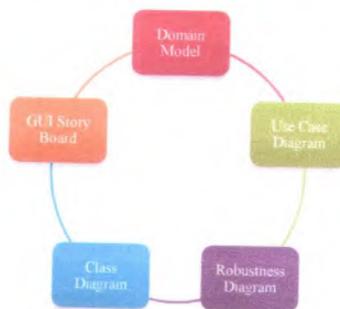
3.8.Pembuatan laporan

Pembuatan laporan merupakan sarana dokumentasi agar seluruh kegiatan yang pembuatan tugas akhir ini dapat dijabarkan secara lengkap sehingga mampu memberikan informasi yang berguna bagi individu yang membacanya. Pada laporan ini terdapat *kesimpulan dan saran berupa pengembangan atau perbaikan* untuk penelitian selanjutnya. Laporan dikerjakan menggunakan

format pengerjaan tugas akhir sehingga menghasilkan sebuah buku tugas akhir.

BAB IV ANALISIS DAN DESAIN SISTEM

Bab ini akan menjelaskan rancangandari Sistem Informasi Pengelolaan Keuangan Negara (SIAPKAN) Modul Pembayaran. Rancangan sistem ini mengacu pada *ICONIX process* dengan alur pembuatan secara berurutan, yaitu perancangan *GUI (Graphical User Interface) story board*, *domain model*, *use case diagram* serta deskripsi, *robustness diagram*, *sequence diagram* dan *test case*. Hal ini tergambar dalam Gambar 4.1. Selain itu akan dijelaskan mengenai analisa kebutuhan serta alur aplikasi khususnya Modul Pembayaran.



Gambar 4.1 ICONIX Process

4.1. Analisis Kebutuhan Sistem

Analisa dibutuhkan untuk memahami karakteristik yang ada dalam Modul Pembayaran. Berdasarkan informasi yang didapatkan dari pengguna Aplikasi SPM dan dosen pembimbing teknis, maka didapatkan beberapa analisa yang diperlukan untuk Modul Pembayaran, yaitu analisa aplikasi pengguna dan analisa kebutuhan.

4.1.1.Kondisi Sistem Saat Ini

Modul Pembayaran disusun berdasarkan aplikasi yang sudah ada, yaitu Aplikasi Surat Perintah Membayar (SPM). Aplikasi SPM mempunyai fungsi utama yaitu menerbitkan Surat Perintah Membayar sebagai dasar pencairan dana APBN. Aplikasi SPM juga mencakup perekaman data Kontrak dari pihak ketiga serta komponen gaji yang berasal dari input data gaji.

Sebelum SPM diterbitkan, Pejabat Pembuat Komitmen di masing-masing Satuan Kerja (Satker) membuat Surat Permintaan Pembayaran (SPP). SPP yang diterbitkan merupakan rangkuman dari seluruh transaksi yang telah dicatat oleh Bendahara Pengeluaran. Transaksi tersebut disahkan dalam bentuk Surat Permintaan Tanggung Jawab Belanja (SPTB) agar dapat dijadikan dasar untuk penerbitan SPP.

Setelah SPP diterbitkan, Pejabat Penerbit SPM menyetujui untuk menerbitkan SPM. Kemudian SPM yang diterbitkan kemudian diajukan ke Kantor Pelayanan Perbendaharaan Negara bersama Arsip Data Komputer yang dihasilkan dari Aplikasi SPM.

4.1.2.Karakteristik Pengguna

Modul Pembayaran merupakan modul yang digunakan untuk memproses serta mengesahkan permintaan pembayaran yang dilakukan oleh Satuan Kerja sebagai Kuasa Pengguna Anggaran. Dalam hal ini, kewenangan untuk menerbitkan SPP serta SPM ada pada Pejabat Penerbit SPM. Selain itu dibutuhkan operator sebagai penanggungjawab dalam mengelola penatausahaan Modul Pembayaran. Karena itulah pengguna Modul Pembayaran dibagi menjadi :

1. *Operator Modul Pembayaran*, merupakan pengguna yang bertanggung jawab dalam proses penatausahaan, mulai dari perekaman input serta menghasilkan output Surat Permintaan Pembayaran (SPP) dan Surat Perintah Membayar (SPM).

2. *Pejabat Penerbit SPM*, merupakan pengguna yang melaksanakan kewenangan persetujuan dan pengesahan atas terbitnya SPP dan SPM.

4.1.3. Kebutuhan Modul Pembayaran

Modul Pembayaran membutuhkan proses untuk menerbitkan Surat Permintaan Pembayaran (SPP) dan Surat Perintah Membayar (SPM), serta komponen yang menjadi bagian dari SPP dan SPM. Kebutuhan yang terdapat pada Modul Pembayaran telah dituangkan pada *Use Case* Modul Pembayaran. *Use Case* tersebut mencakup kebutuhan sebagai berikut :

- Proses perekaman data Resume Kontrak
Proses ini adalah perekaman data resume atas Kontrak yang dilakukan Satuan Kerja dengan Pihak Ketiga. Data resume kontrak digunakan sebagai bagian dari penerbitan SPP dan SPM.
- Proses perekaman data Gaji
Proses ini adalah perekaman data resume Gaji Induk yang berasal dari daftar gaji untuk Satuan Kerja bersangkutan. Data resume Gaji ini juga digunakan sebagai bagian dari penerbitan SPP dan SPM.
- Proses perekaman data SPP
Proses ini adalah perekaman Surat Permintaan Pembayaran yang merupakan dasar penerbitan Surat Perintah Membayar untuk dibebankan pada Anggaran Pendapatan Belanja Negara (APBN). Proses ini memerlukan data transaksi dari Kontrak, Gaji, serta Transaksi yang berasal dari Modul Bendahara.
- Proses Persetujuan dan Pembatalan SPP-SPM
Proses ini adalah pengesahan dari SPP yang telah direkam agar SPP dapat diubah statusnya menjadi SPM.
- Proses perekaman data SP2D
Proses ini adalah perekaman atau pencatatan nomor SP2D yang telah diterima dari Kantor Pelayanan

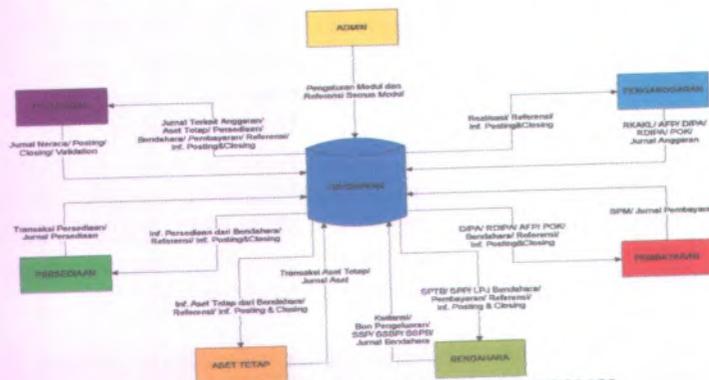
Perbendaharaan Negara (KPPN). Proses ini diperlukan agar SPM yang telah terbit dapat digunakan sebagai input untuk Modul Pelaporan.

- Pencetakan SPP dan SPM
Proses ini menyajikan output dari Modul Pembayaran yang digunakan sebagai dasar pembayaran APBN.
- Pengiriman ADK SPM
Proses ini adalah pengiriman Arsip Data Komputer yang berupa file untuk diterima di KPPN.

4.2. Arsitektur Aplikasi SIAPKAN

4.2.1. Arsitektur Desain

Aplikasi Sistem Informasi Pengelolaan Keuangan Negara (SIAPKAN) yang akan dikembangkan merupakan integrasi dari beberapa aplikasi yang sebelumnya telah digunakan pada Satuan Kerja di lingkungan Direktorat Jenderal Perbendaharaan. Pengembangan aplikasi ini terbagi menjadi tujuh modul, yaitu Modul Administrasi, Modul Anggaran, Modul Bendahara, Modul Pembayaran, Modul Aset Tetap dan Modul Persediaan. Gambar 4.2 menunjukkan Arsitektur global pada aplikasi SIAPKAN.



Gambar 4.2 Arsitektur Desain Aplikasi SIAPKAN

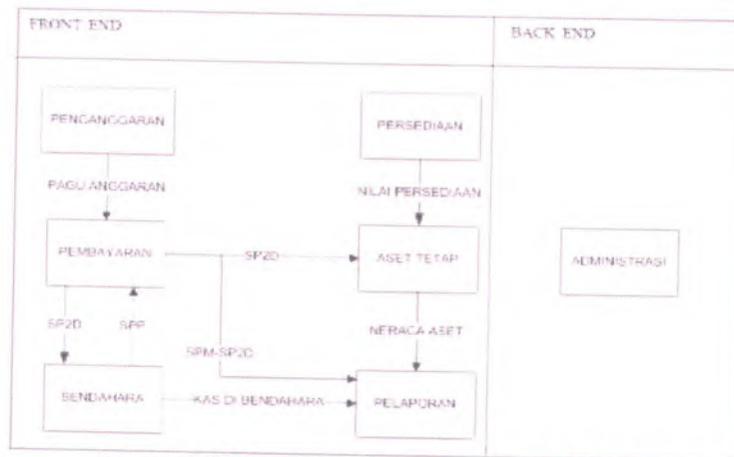
Hubungan serta keterkaitan antar modul SIAPKAN dapat dijelaskan sebagai berikut:

1. Modul Administrasi berfungsi untuk menangani kebutuhan referensi data untuk modul-modul lainnya. Selain itu Modul Administrasi berfungsi untuk menatausahakan pengguna aplikasi sesuai dengan kebutuhan masing-masing Satuan Kerja. Output dari Modul Administrasi berupa referensi Anggaran, Aset, Persediaan, Bendahara dan Pembayaran serta manajemen pengguna.
2. Modul Penganggaran berfungsi untuk menyusun kertas kerja Rencana Kerja Anggaran Kementerian Lembaga (RKA-KL) di setiap Satuan Kerja (Satker). Modul Penganggaran mendapatkan referensi dari Modul Administrasi sebagai dasar untuk pembuatan Kertas Kerja Pedoman Operasional Kinerja (POK) dan Daftar Isian Pelaksanaan Anggaran (DIPA). Output yang dihasilkan yaitu POK digunakan oleh Modul Bendahara sebagai dasar untuk menyusun transaksi pengeluaran. Sedangkan DIPA digunakan oleh Modul Pembayaran sebagai dasar pencairan dana APBN.
3. Modul Bendahara berfungsi untuk menatausahakan transaksi pengeluaran serta penerimaan. Dasar penyusunan transaksi berasal dari POK yang dihasilkan oleh Modul Penganggaran. Output dari Modul Bendahara berupa Kuitansi, Surat Pernyataan Tanggung Jawab Belanja (SPTB) dan Pembukuan Bendahara. Output tersebut digunakan oleh Modul Pembayaran sebagai dasar pembuatan Surat Permintaan Pembayaran.
4. Modul Pembayaran berfungsi untuk memproses dokumen pembayaran yang berupa Surat Permintaan Pembayaran (SPP) dan Surat Perintah Membayar (SPM) sebagai dasar permintaan pembayaran APBN masing-masing Satuan

Kerja. Dasar pencairan dana berupa DIPA berasal dari Modul Penganggaran, sementara input yang digunakan untuk membuat SPP dan SPM berasal dari Modul Bendahara yang berupa Kuitansi dan Surat Pernyataan Tanggung Jawab Belanja. Selain itu Modul Pembayaran menggunakan input referensi yang berasal dari Modul Administrasi. Output SPP dan SPM digunakan oleh Modul Aset Tetap sebagai dasar pencatatan transaksi pembelian aset. Sedangkan untuk Modul Pelaporan digunakan sebagai dasar untuk menyusun jurnal, buku besar serta laporan keuangan tingkat Satuan Kerja.

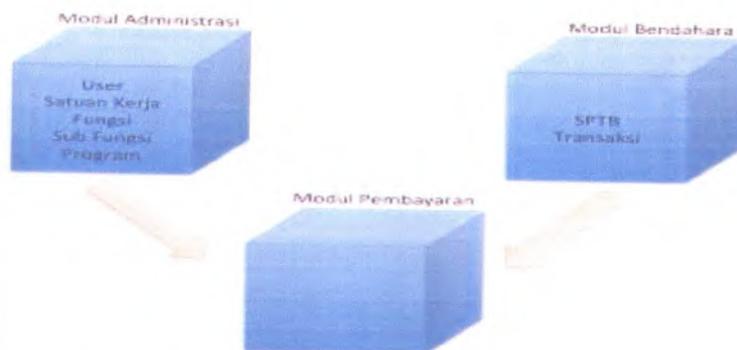
5. Modul Aset Tetap berfungsi untuk menatausahakan aset tetap pada masing-masing Satuan Kerja (Satker). Modul Aset Tetap menggunakan data SPM yang telah tercatat nomor Surat Perintah Pencairan Dana (SP2D) untuk pencatatan transaksi pembelian serta mengambil nilai saldo persediaan dari Modul Persediaan untuk menyusun laporan posisi Barang Milik Negara (BMN).
6. Modul Persediaan berfungsi untuk menatausahakan barang habis pakai (persediaan) pada setiap satuan kerja. Modul persediaan menggunakan referensi kode barang, jenis transaksi, dan unit pengguna dari Modul Administrasi serta menyediakan data nilai saldo persediaan bagi Modul Aset Tetap dalam penyusunan laporan posisi BMN.
7. Modul Pelaporan menerima posting jurnal dari modul-modul lain dan menampungnya dalam general ledger untuk digunakan menyusun laporan keuangan dengan didukung data referensi dari Modul Administrasi. Setiap periode, Modul Laporan mengirim feedback kepada modul lain setelah melakukan closing, bahwa data tidak boleh di edit lagi.

Gambar 4.3 menunjukkan keterkaitan antar modul pada aplikasi Sistem Informasi Pengelolaan Keuangan Negara.



Gambar 4.3 Hubungan Antar Modul Aplikasi SIAPKAN

Pada Modul Pembayaran, integrasi yang dilakukan terkait dengan Modul Administrasi serta Modul Bendahara. Gambar 4.4 menunjukkan hubungan antar komponen yang ada pada setiap modul.



Gambar 4.4 Hubungan Antar Modul Aplikasi SIAPKAN

Komponen yang diperlukan dari Modul Administrasi adalah sebagai berikut :

- a. Tabel User
- b. Tabel Satuan Kerja
- c. Tabel Fungsi
- d. Tabel Sub Fungsi
- e. Tabel Program

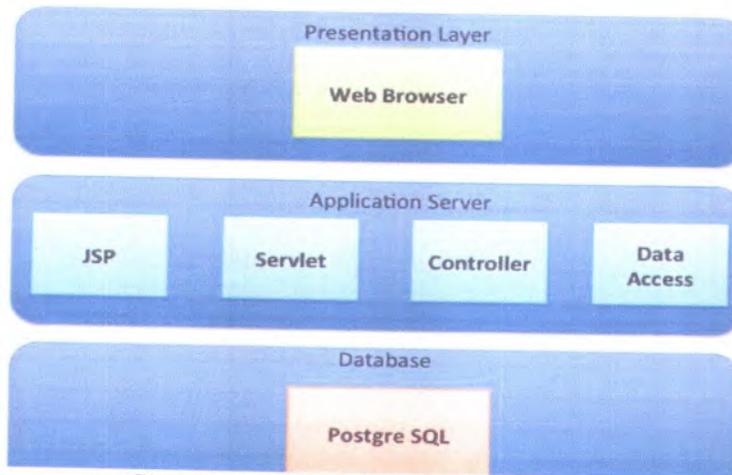
Sedangkan komponen yang dibutuhkan dari Modul Bendahara adalah sebagai berikut:

- a. Tabel SPTB
- b. Tabel Transaksi

Modul Pembayaran membutuhkan tabel *User* yang diambil melalui *session* ketika pengguna masuk ke dalam aplikasi melalui halaman Login. Untuk melakukan perekaman data, dibutuhkan komponen tabel SPTB dan Transaksi yang berasal dari Modul Bendahara. Modul Pembayaran akan memberikan umpan balik yang berupa penanda apabila komponen tabel yang diperlukan telah digunakan.

4.2.2. Arsitektur Teknis

Sistem Informasi Pengelolaan Keuangan Negara dibangun dengan menggunakan teknologi JavaServer Pages serta database PostgreSQL. Gambar 4.5 menunjukkan arsitektur teknis yang digunakan pada aplikasi.



Gambar 4.5 Arsitektur Teknis aplikasi SIAPKAN

Aplikasi SIAPKAN menggunakan tiga *layer* yaitu *Presentation Layer*, *Application Server* dan *Database*. Berikut merupakan poin-poin pada arsitektur teknologi yang digunakan.

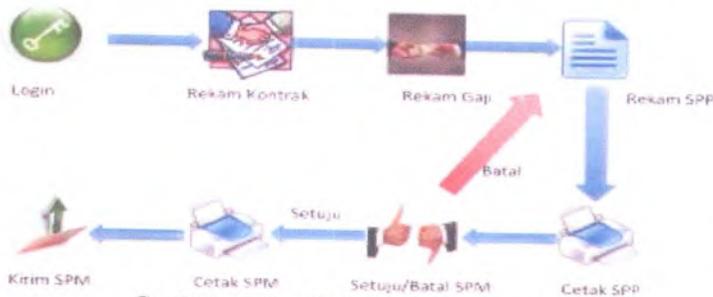
1. *Presentation Layer* merupakan hasil yang dipresentasikan kepada pengguna. Aplikasi menggunakan *Web Browser* sebagai media untuk menampilkan output yang dihasilkan oleh aplikasi. Output tersebut berasal dari *Application Server*.
2. *Application Server* merupakan layer yang terdiri dari beberapa komponen dan merupakan inti dari aplikasi. *Application Server* dijalankan dengan *Tomcat Server*. Berikut merupakan komponen yang terdapat pada *Application Server*.
 - a. *JSP* : pada arsitektur tersebut, halaman *JSP* digunakan sebagai tampilan yang berinteraksi langsung dengan pengguna aplikasi SIAPKAN. Interaksi dapat berupa :
 - i. *Form isian* yang mengharuskan pengguna untuk menginput data

- ii. Daftar yang berisi informasi yang dilihat oleh pengguna dan dapat digunakan untuk penghubung untuk mengubah atau menghapus data yang ada pada daftar tersebut.
 - b. Servlet : pada arsitektur tersebut, Servlet berfungsi untuk menerima serta meneruskan perintah yang berasal dari pengguna maupun dari Controller. Setelah pengguna berinteraksi dengan halaman JSP, maka sistem akan memproses informasi yang dikirimkan melalui servlet menuju Controller untuk diproses lebih lanjut.
 - c. Controller : pada arsitektur tersebut, Controller berfungsi untuk menerima serta meneruskan perintah yang berasal dari Servlet maupun dari Model. Perintah yang diterima atau diteruskan akan dikelola terlebih dahulu agar sesuai dengan permintaan yang dikirim baik oleh Servlet ataupun Model. Perintah yang diolah umumnya berupa pengolahan data seperti perekaman, perubahan serta penghapusan.
 - d. Data Access : pada arsitektur tersebut, Data Access merupakan representasi dari pengolahan data. Model menggunakan *Java Persistence API* sehingga *Query* yang dikelola lebih mudah. Hal ini mempercepat permintaan yang diterima atau dikirim melalui Controller.
3. Database merupakan layer tempat menampung data pada aplikasi. Data tersebut yang ditransmisikan ketika ada permintaan dari model yang berupa *Query*.

4.3. Alur Kerja Modul Pembayaran

Pada Modul Pembayaran, alur yang dirancang tidak berbeda dengan Aplikasi SPM. Perbedaan hanya terdapat pada media

komunikasi, dimana Aplikasi SPM berbasis *Stand Alone*, sedangkan Modul Pembayaran berbasis web. Gambar 4.6 menunjukkan alur kerja Modul Pembayaran. Berikut dijelaskan mengenai alur kerja pada Modul Pembayaran.



Gambar 4.6 Alur Kerja Modul Pembayaran

4.3.1. Perekaman Data Kontrak

Proses yang dilakukan pertama kali pada Modul Pembayaran adalah perekaman data Kontrak. Data Kontrak dibutuhkan sebagai pengawasan dalam belanja yang membutuhkan kerjasama dengan Pihak Ketiga (PFK). Pada saat Surat Perintah Membayar (SPM) diterbitkan berdasarkan kontrak yang telah disepakati, maka SPM akan mengurangi total nilai kontrak agar tidak terjadi kelebihan pembayaran pada kontrak tersebut.

4.3.2. Perekaman Data Gaji

Salah satu komponen Surat Perintah Membayar (SPM) merupakan data gaji. Untuk melakukan pembayaran Gaji Induk diperlukan data gaji yang berasal dari Daftar Gaji masing-masing Satuan Kerja (Satker). Daftar gaji tersebut berisi rincian gaji masing-masing pegawai serta potongan yang dibebankan dan biaya-biaya lain yang mengurangi komponen gaji induk pegawai.

Pada Modul Pembayaran, data gaji yang diinput berupa rekapitulasi gaji serta potongan serta biaya lainnya.

4.3.3.Perekaman Data SPP

Transaksi yang telah dicatat pada Modul Bendahara dan telah dibuat menjadi SPTB akan dijadikan input untuk merekam data SPP. Ada beberapa jenis SPP yang diterbitkan, yaitu:

8. SPP Gaji Induk. SPP tersebut diterbitkan berdasarkan pada data gaji yang telah diinput sebelumnya
9. SPP Langsung Kontrak. SPP tersebut diterbitkan berdasarkan SPTB yang telah diterbitkan pada Modul Bendahara dan data kontrak yang telah diinput.
10. SPP Langsung Non Kontrak. SPP tersebut diterbitkan berdasarkan SPTB yang telah diterbitkan pada Modul Bendahara, tetapi bukan merupakan transaksi yang berhubungan dengan Pihak Ketiga (PFK).
11. SPP Ganti Uang Persediaan, SPP tersebut diterbitkan berdasarkan SPTB yang telah diterbitkan pada Modul Bendahara. Transaksi yang dicatat telah dibayar sebelumnya menggunakan Uang Muka Persediaan (UP). UP akan dikembalikan setelah Surat Perintah Pencairan Dana (SP2D) diterbitkan oleh Kantor Pelayanan Perbendaharaan Negara.
12. SPP Uang Persediaan. SPP tersebut diterbitkan hanya satu kali pada masa tahun anggaran berjalan. SPP Uang Persediaan diterbitkan untuk mencairkan dana Uang Muka sebagai dana operasional Satuan Kerja dalam melaksanakan kegiatan yang bersifat rutin.
13. SPP Gaji Induk. SPP tersebut diterbitkan berdasarkan data gaji yang telah diinput sebelumnya. SPP gaji

dibayarkan setiap awal bulan untuk mengakomodir kebutuhan belanja pegawai di masing-masing Satker.

4.3.4.Cetak SPP

Setelah data SPP direkam, diperlukan pencetakan SPP sebagai dasar untuk penerbitan Surat Perintah Membayar. SPP yang telah dicetak beserta dengan lampiran cetak SPTB yang berasal dari Modul Bendahara diserahkan pada Pejabat Penerbit SPM untuk melakukan persetujuan SPM. Pejabat Penerbit SPM melakukan pengujian substantif dan formal, yaitu menguji kelengkapan dokumen serta ketepatan yang dihasilkan pada output SPP.

4.3.5.Persetujuan dan Pembatalan SPM

Setelah melakukan pengujian terhadap SPP, Pejabat Penerbit SPM melakukan persetujuan SPM pada Modul Pembayaran. Apabila Pejabat telah menyetujui, maka sistem akan menghasilkan SPM. Data SPM tersebut merupakan data SPP sebelumnya yang telah disahkan atau disetujui. Hal ini sedikit berbeda dengan Aplikasi SPM dimana data SPM harus kembali direkam. Apabila SPM telah disetujui tetapi terdapat kesalahan yang menyebabkan SPM harus dibatalkan, Pejabat berhak untuk melakukan pembatalan SPM. Pembatalan ini hanya dapat dilakukan sebelum SPM diajukan ke Kantor Pelayanan Perbendaharaan Negara dan diterbitkan Surat Perintah Pencairan Dana.

4.3.6.Cetak SPM

Cetak SPM dilakukan setelah SPM disetujui oleh Pejabat Penerbit SPM. Hasil cetak SPM tersebut ditandatangani oleh Pejabat

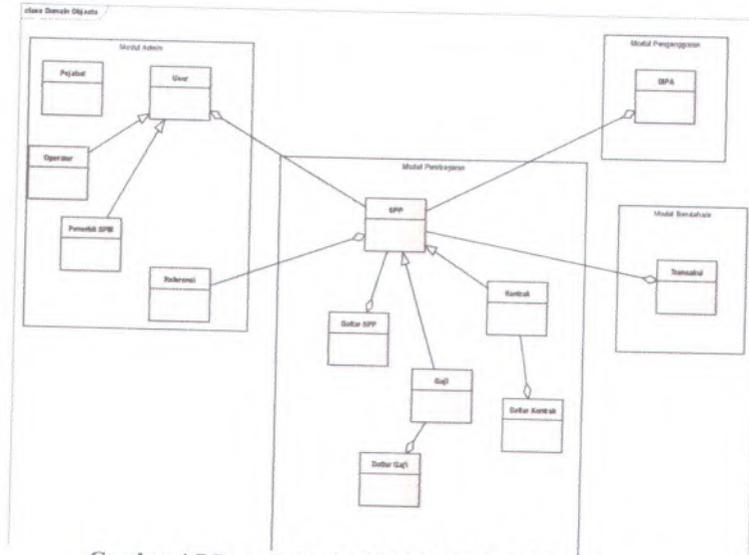
Penerbit SPM kemudian dilengkapi dengan dokumen pendukung berupa SPP serta dokumen lain yang berkaitan dengan SPP. Dokumen-dokumen tersebut kemudian disampaikan kepada Kantor Pelayanan Perbendaharaan Negara bersama dengan Arsip Data Komputer.

4.3.7. Pengiriman Arsip Data Komputer SPM

Data SPM yang telah dicetak dapat dihasilkan Arsip Data Komputer sebagai pendukung dalam pengajuan SPM pada Kantor Pelayanan Perbendaharaan Negara. Arsip Data Komputer berisi seluruh data SPM serta transaksi yang dihasilkan dan dicatat pada Modul Bendahara.

4.4. Domain Model

Sesuai dengan proses pada ICONIX, *domain model* digunakan untuk menggambarkan obyek-obyek utama yang akan digunakan serta menginisialisasi cakupan dari sistem dan hal-hal yang terlibat di dalamnya. *Domain model* dapat berubah seiring dengan pengembangan desain dan aplikasi, sehingga obyek-obyek yang digambarkan pada *domain model* akan semakin lengkap dan akurat sesuai dengan alur sistem. Pada aplikasi SIAPKAN yang dikembangkan pada Tugas Akhir ini, penulis menginisialisasi *domain model* yang diperlihatkan Gambar 4.7 sebagai berikut.



Gambar 4.7 Domain Model SIAPKAN Modul Pembayaran

Berdasarkan *domain model* pada Gambar 4.6 terdapat beberapa *class* yang diinisialisasi, yaitu :

1. **Kontrak**, merupakan salah satu jenis komponen pembayaran yang berisi tentang kesepakatan antara Satuan Kerja dengan Pihak Ketiga (PFK) untuk suatu kegiatan tertentu dengan nilai pembayaran tertentu. Kontrak pada aplikasi ini merupakan input resume dari kontrak yang sesungguhnya dan akan menjadi komponen dari SPP.
2. **Daftar Kontrak**, merupakan kumpulan dari Kontrak yang ada.
3. **Gaji**, merupakan salah satu jenis komponen pembayaran yang berisi rincian komponen pembayaran gaji induk bagi PNS atau TNI/POLRI dengan nilai pembayaran tertentu setiap bulan nya. Gaji pada aplikasi ini merupakan input resume dari daftar gaji yang

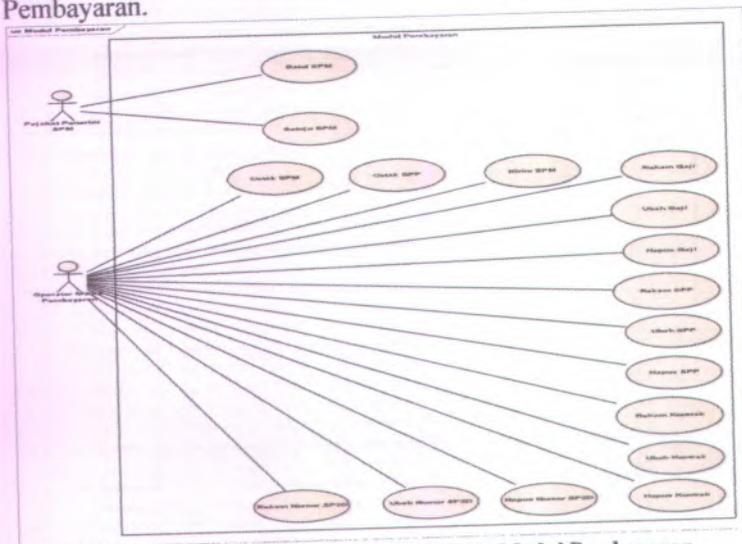
sesungguhnya dan akan menjadi komponen pembayaran dari SPP.

4. **Daftar Gaji**, merupakan kumpulan dari Gaji yang ada.
5. **SPP**, merupakan Surat Permintaan Pembayaran yang berisi permintaan pembayaran yang didapat dari Transaksi maupun Kontrak dan Gaji.
6. **Daftar SPP**, merupakan kumpulan dari SPP yang ada.

Terdapat juga beberapa *domain* yang berasal dari modul lain yang terintegrasi pada aplikasi SIAPKAN. *Domain* tersebut diperlukan untuk melengkapi kebutuhan yang diperlukan pada *domain* Modul Pembayaran.

4.5. Use Case Diagram

Use Case Diagram menunjukkan hal-hal yang dapat dilakukan oleh setiap aktor dalam Modul Pembayaran. Gambar 4.8 menunjukkan keseluruhan *Use Case Diagram* pada Modul Pembayaran.



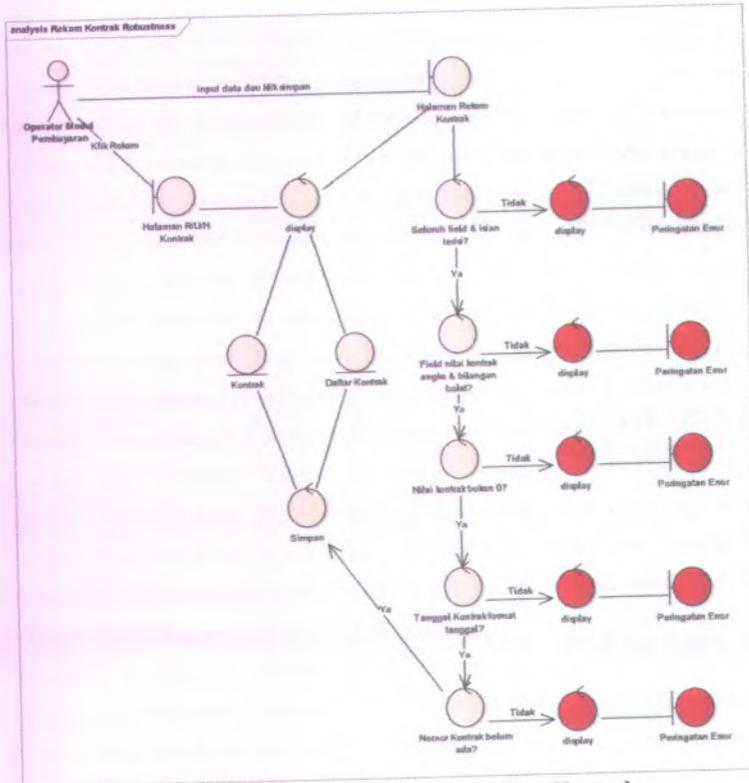
Gambar 4.8 Use Case Diagram SIAPKAN Modul Pembayaran

Use case diagram serta detail deskripsi masing-masing *use case* dapat dilihat pada Lampiran A. Berikut ini merupakan *use case* yang ada pada Modul Pembayaran :

1. Use Case Rekam Kontrak
2. Use Case Ubah Kontrak
3. Use Case Hapus Kontrak
4. Use Case Rekam Gaji
5. Use Case Ubah Gaji
6. Use Case Hapus Gaji
7. Use Case Rekam SPP
8. Use Case Ubah SPP
9. Use Case Hapus SPP
10. Use Case Setuju SPM
11. Use Case Batal SPM
12. Use Case Cetak SPP
13. Use Case Cetak SPM
14. Use Case Rekam SP2D
15. Use Case Ubah SP2D
16. Use Case Hapus SP2D
17. Use Case Kirim SPM

4.6. Robustness Diagram

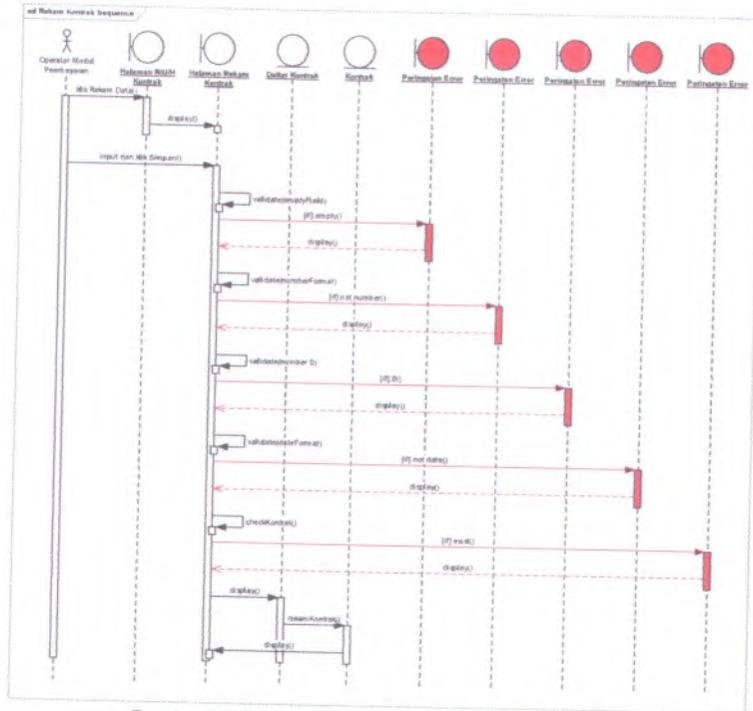
Robustness diagram merupakan penjelasan alur dari *use case* yang dirancang. *Robustness diagram* dibuat di dalam seluruh *usecase* yang ada dan mengacu pada diagram *use case* sebelumnya. Gambar 4.9 menunjukkan tampilan *Robustness Diagram* Rekam Kontrak. Detail dari seluruh *Robustness diagram* dapat dilihat pada Lampiran B.



Gambar 4.9 Robustness Diagram Rekam Kontrak

4.7. Sequence Diagram

Sequence diagram merupakan penjelasan dari proses yang terjadi pada sistem dan memuat alur dalam *usecase* yang telah didefinisikan sebelumnya. Gambar 4.10 menunjukkan Sequence Diagram Rekam Kontrak. Detil perancangan *sequence diagram* dapat dilihat pada lampiran C.



Gambar 4.10 Robustness Diagram Rekam Kontrak

4.8. Class Diagram

Class Diagram merupakan penjabaran dari class yang ada pada Java dan digunakan pada aplikasi ini. Class diagram meliputi atribut serta operasi yang ada di dalam setiap class tersebut. Detil Class Diagram dapat dilihat pada lampiran D.

4.9. Test Case

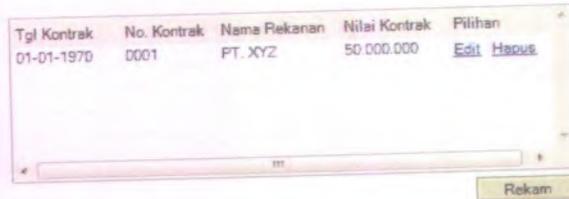
Test case merupakan proses yang dirancang untuk menjaga performa aplikasi agar sesuai dengan desain telah yang dibuat. *Test case* akan dijalankan dengan beberapa skenario yang sesuai

dengan rancangan pada diagram *use case*. Perancangan skenario dan *test* dapat dilihat pada lampiran D.

4.10. GUI Story Board

GUI Story Board digunakan sebagai gambaran awal atau bentuk dari setiap form yang ada pada sistem yang dikembangkan. Pada perancangan GUI untuk aplikasi SIAPKAN meliputi rancangan desain tampilan form input untuk masing-masing Use Case serta desain tampilan error apabila terjadi kesalahan dalam pemrosesan input data. Gambar 4.11 menunjukkan tampilan untuk halaman RUH Kontrak. Halaman tersebut berisi daftar Kontrak yang ada pada database. Pada halaman ini beberapa tombol, yaitu tombol Rekam untuk menuju halaman Rekam Kontrak (Gambar 4.12), tombol Edit untuk menuju ke halaman form Ubah Kontrak (Gambar 4.14) serta tombol Hapus yang digunakan untuk menghapus data Kontrak.

RUH Kontrak



Tgl Kontrak	No. Kontrak	Nama Rekanan	Nilai Kontrak	Pilihan
01-01-1970	0001	PT. XYZ	50.000.000	Edit Hapus

Gambar 4.11 RUH Kontrak

Halaman Rekam Kontrak merupakan form untuk menambah data Kontrak pada aplikasi SIAPKAN. Tampilan form Rekam Kontrak dapat dilihat pada Gambar 4.12.

Rekam Kontrak

Tanggal Kontrak

Nomor Kontrak

Rincian DPB

Nama Pakan

Jenis Rekrut

Tipe Kontrak

Uraian

Indikator HSDM Pemasangan

Detail

Gambar 4.12 Rekam Kontrak

Sedangkan pada Gambar 4.13 memperlihatkan tampilan form pada halaman Rekam Kontrak dengan tampilan kesalahan pada saat menginputkan data.

Rekam Kontrak

Tanggal Kontrak

Nomor Kontrak

Rincian DPB

Nama Pakan

Jenis Rekrut

Tipe Kontrak

Uraian

Indikator HSDM Pemasangan

Detail

*Nilai Harus Terdiri
*Jumlah Tanggal Harus

Gambar 4.13 Rekam Kontrak – Peringatan Error

Gambar 4.14 menunjukkan tampilan form pada halaman Ubah Kontrak. Tampilan form Ubah Kontrak sama dengan tampilan form Rekam Kontrak. Form ini digunakan untuk edit data Kontrak yang sudah pernah direkam sebelumnya.

Ubah Kontrak

Tanggal Kontrak	01-01-1970
Nomor Kontrak	0001-DPK2012
Proyek DPK	0001.0001.0001.521119
Nama Pakan	PT. XNC
Jenjang Rekrutan	JR. AEC
Nilai Kontrak	15.000.000
Uraian	Pembelian ATK
Jangka Waktu Pelaksanaan	3 Hari
Sanksi	Potang Tandang

Gambar 4.14 Ubah Kontrak

Gambar 4.15 merupakan tampilan konfirmasi apabila data Kontrak akan dihapus dengan menekan tombol Hapus pada halaman RUH Kontrak (Gambar 4.11). Terdapat pilihan untuk kembali ke halaman RUH Kontrak atau untuk menghapus data Kontrak dari database. Tampilan konfirmasi penghapusan digunakan di setiap proses penghapusan data.



Gambar 4.15 Konfirmasi Hapus

Gambar 4.16 menunjukkan tampilan halaman RUH Gaji. Halaman RUH Gaji menampilkan data Gaji yang telah direkam. Pada halaman tersebut terdapat tombol Rekam untuk menuju ke halaman Rekam Gaji (Gambar 4.17), tombol Edit untuk menuju

ke halaman Ubah Gaji (Gambar 4.19) dan tombol Hapus yang digunakan untuk menghapus data Gaji.

RUH Gaji

Bulan Bayar	Total Gaji	Total Potongan	Gaji Bersih	Pilihan
1	50.000.000	500.000	49.500.000	Edit Hapus

Rekam

Gambar 4.16 RUH Gaji

Halaman Rekam Gaji merupakan form untuk menambahkan data Gaji pada aplikasi SIAPKAN. Tampilan form Rekam Gaji dapat dilihat pada Gambar 4.17.

Rekam Gaji

Revisi Data <input type="text" value="1000000000"/>	Jml. Pegawai <input type="text" value="100"/>	Jml. Jaga <input type="text" value="100"/>	Jml. Jaga <input type="text" value="100"/>	Jml. Jaga <input type="text" value="100"/>
Bulan Bayar <input type="text" value="1"/>	Gaji Pokok <input type="text" value="100000000"/>	Tunj. Jamin <input type="text" value="100000000"/>	Tunj. Jarak <input type="text" value="100000000"/>	Gaji Kotor <input type="text" value="100000000"/>
	Tunj. PP <input type="text" value="100000000"/>	Tunj. Dibent <input type="text" value="100000000"/>	Tunj. Terpens <input type="text" value="100000000"/>	Potongan <input type="text" value="100000000"/>
	Tunj. Struktural <input type="text" value="100000000"/>	Tunj. Fungsional <input type="text" value="100000000"/>	Pertambahan <input type="text" value="100000000"/>	Gaji Bersih <input type="text" value="100000000"/>
	Tunj. Belas <input type="text" value="100000000"/>	Tunj. Lain <input type="text" value="100000000"/>	Tunj. PPh <input type="text" value="100000000"/>	
	Tunj. Senda <input type="text" value="100000000"/>	Tunj. Mecha <input type="text" value="100000000"/>	Tunj. Kompenasi <input type="text" value="100000000"/>	
	Tunj. Batin <input type="text" value="100000000"/>	Tunj. Uang Pajak <input type="text" value="100000000"/>	Tunj. Pungsi <input type="text" value="100000000"/>	
	Tunj. Uang Pns <input type="text" value="100000000"/>	PKK 12% <input type="text" value="100000000"/>	PKK 2% <input type="text" value="100000000"/>	
	PKK 50% <input type="text" value="100000000"/>	Potongan PKK 2% <input type="text" value="100000000"/>	Bil. Pegawai Murni <input type="text" value="100000000"/>	
Pst. Sisa Nomor <input type="text" value="100000000"/>	Pst. Knt. Bekerja <input type="text" value="100000000"/>	Pst. Lulusan <input type="text" value="100000000"/>	<input type="button" value="Simpan"/>	<input type="button" value="Kembali"/>

Gambar 4.17 Rekam Gaji

Sedangkan Gambar 4.18 menunjukkan tampilan form Rekam Gaji dengan tampilan kesalahan pada saat menginputkan data.

The screenshot shows a web form titled "Rekam Gaji". It contains several input fields for employee data and salary components. On the left, there are dropdown menus for "Revisi DPR" (set to "0000,0000,0000") and "Bulan Bayar". The main form is organized into columns for different salary components: "Jml. Pegawai", "Jml. Istim", "Jml. Anak", "Gaji Pokok", "Tunj. Istri", "Tunj. Anak", "Tunj. PP", "Tunj. Daerah", "Tunj. Terpenuhi", "Tunj. Struktural", "Tunj. Fungsional", "Pembulatan", "Tunj. Belas", "Tunj. Lain", "Tunj. PPH", "Tunj. Senda", "Tunj. Medis", "Tunj. Kompensasi", "Tunj. Dabat", "Tunj. Lemb. Pns", "Tunj. Pns/Non", "Tunj. Umum Pns", "PPh 10%", "PPh 2%", "PPh Bulang", "Potongan PPh 2%", "Pot. Tabungan Runtah", "Pot. Sewa Rumah", "Pot. Hal. Batawa", "Pot. Lain-lain". At the bottom right, there are "Simpan" and "Kembali" buttons. A red error message "Validasi harus diisi" is displayed next to the "Gaji Bersih" field.

Gambar 4.18 Rekam Gaji – Peringatan Error

Gambar 4.19 memperlihatkan form pada halaman Ubah Gaji. Tampilan form Ubah Gaji sama persis dengan form pada halaman Rekam Gaji. Form tersebut digunakan untuk mengedit data Gaji yang sudah ada pada database.

The screenshot shows a web form titled "Ubah Gaji". It is identical in layout to the "Rekam Gaji" form but contains pre-filled data. The "Revisi DPR" dropdown is set to "0000,0000,0000" and "Bulan Bayar" is empty. The data entered in the fields is as follows:

Jml. Pegawai	10	Jml. Istim	5	Jml. Anak	3	Gaji Kotor	30.000.000
Gaji Pokok	20.000.000	Tunj. Istri	2.000.000	Tunj. Anak	499.800	Potongan	5.000.000
Tunj. PP	10.000	Tunj. Daerah	10.000	Tunj. Terpenuhi	110.000	Gaji Bersih	
Tunj. Struktural	200.000	Tunj. Fungsional	100.000	Pembulatan	200		
Tunj. Belas	50.000	Tunj. Lain	10.000	Tunj. PPH	1.000.000		
Tunj. Senda	50.000	Tunj. Medis	10.000	Tunj. Kompensasi	50.000		
Tunj. Dabat	500.000	Tunj. Lemb. Pns	500.000	Tunj. Pns/Non	400.000		
Tunj. Umum Pns	1.000.000	PPh 10%	2.250.000	PPh 2%	450.000		
PPh Bulang	10.000	Potongan PPh 2%	10.000	Pot. Tabungan Runtah	50.000		
Pot. Sewa Rumah	50.000	Pot. Hal. Batawa	50.000	Pot. Lain-lain	50.000		

 At the bottom right, there are "Simpan" and "Kembali" buttons.

Gambar 4.19 Ubah Gaji

Tampilan untuk konfirmasi penghapusan Gaji pada halaman RUH Gaji (Gambar 4.16) sama persis dengan konfirmasi penghapusan Kontrak (Gambar 4.15). Gambar 4.20 menunjukkan halaman RUH SPP yang menampilkan data SPP yang telah direkam. Pada halaman tersebut terdapat tombol Rekam untuk menuju ke halaman Rekam SPP (Gambar 4.21), tombol Edit untuk menuju ke halaman Ubah SPP (Gambar 4.23) dan tombol Hapus yang digunakan untuk menghapus data SPP.

RUH SPP

Tgl SPP	No. SPP	Jenis SPP	Nilai SPP	Pilihan
01-01-1970	0001	LS	50.000.000	Edit Hapus

Rekam

Gambar 4.20 RUH SPP

Gambar 4.20 memperlihatkan tampilan form pada halaman Rekam SPP. Halaman tersebut digunakan untuk menambah data SPP pada aplikasi SIAPKAN. Sedangkan Gambar 4.21 menunjukkan tampilan form Rekam SPP dengan tampilan kesalahan saat menginputkan data.

Rekam SPP

Tanggal SPP	<input type="text"/>
Nomor SPP	<input type="text"/>
Revisi DPP	<input type="text" value="100.0000.100"/>
Jenis SPP	<input type="text" value="Langung"/>
TipeSaksi	<input type="text" value="SPTB/GAJI/KONTRAK"/>
Potongan	<input type="text" value="PPH Pasal 21(2)(2)"/>
Uraian	<input type="text"/>
Pengeluaran	<input type="text"/>
Potongan	<input type="text"/>

Gambar 4.21 Rekam SPP

Rekam SPP

Tanggal SPP	<input type="text"/>	
Nomor SPP	<input type="text"/>	Field harus diisi
Revisi DPP	<input type="text" value="100.0000.100"/>	
Jenis SPP	<input type="text" value="Langung"/>	
TipeSaksi	<input type="text" value="SPTB/GAJI/KONTRAK"/>	
Potongan	<input type="text" value="PPH Pasal 21(2)(2)"/>	
Uraian	<input type="text"/>	
Pengeluaran	<input type="text"/>	
Potongan	<input type="text"/>	

Gambar 4.22 Rekam SPP – Peringatan Error

Gambar 4.22 menunjukkan tampilan form pada halaman Ubah SPP. Halaman tersebut digunakan untuk mengedit data SPP yang sudah ada pada database. Tampilan form Ubah SPP sama persis dengan form pada halaman Rekam SPP.

Ubah SPP

Tanggal SPP	01-01-1970
Nomor SPP	
Rincian DPJ	0001 0001 0001
Jenis SPP	Lampung
Tipe SPP	SPTB/GAUKONTRAK
Polygon	PPH Pasal 21.22.23
Isi SPP	Pembayar ATK
Pengukuran	7.000.000
Pinsangan	100.000

Gambar 4.23 Ubah SPP

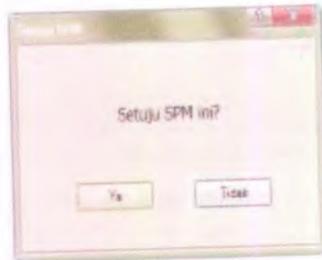
Sedangkan untuk konfirmasi penghapusan SPP pada halaman RUH SPP (Gambar 4.20) sama persis dengan konfirmasi penghapusan Kontrak (Gambar 4.15). Gambar 4.24 menunjukkan tampilan halaman Setuju SPM. Halaman ini menampilkan data SPP yang siap untuk disetujui menjadi SPM. Terdapat tombol Setuju yang digunakan untuk proses persetujuan SPP menjadi SPM.

Setuju SPM

Tgl SPP	No. SPP	Jenis SPP	Nilai SPP	Pilihan
01-01-1970	0001	LS	50.000.000	<input type="button" value="Setuju"/>

Gambar 4.24 Setuju SPM

Gambar 4.25 menunjukkan tampilan konfirmasi apabila SPP akan disetujui menjadi SPM dengan menekan tombol Setuju pada halaman Setuju SPM (Gambar 4.24). Terdapat pilihan untuk kembali ke halaman Setuju SPM atau untuk menyetujui data SPP menjadi SPM.



Gambar 4.25 Konfirmasi Setuju SPM

Gambar 4.26 memperlihatkan tampilan halaman Batal SPM. Halaman ini menampilkan data SPM yang akan dibatalkan prosesnya untuk kembali menjadi SPP. Terdapat tombol Batal yang digunakan untuk proses pembatalan SPM menjadi SPP. Gambar 4.27 menunjukkan tampilan konfirmasi apabila SPM akan dibatalkan menjadi SPP dengan menekan tombol Batal pada halaman Batal SPM. Terdapat pilihan untuk kembali ke halaman Batal SPM atau untuk membatalkan data SPM menjadi SPP.

Batal SPM

Tgl SPM	No. SPM	Jenis SPM	Nilai SPM	Pilihan
01-01-1970	0001	LS	50.000.000	<input type="button" value="Batal"/>

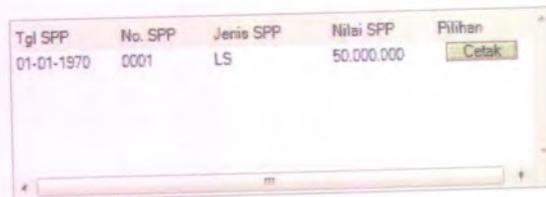
Gambar 4.26 Batal SPM



Gambar 4.27 Konfirmasi Batal SPM

Gambar 4.28 menunjukkan tampilan halaman Cetak SPP. Halaman ini menampilkan data SPP yang akan dicetak ke printer. Terdapat tombol Cetak yang berfungsi untuk proses cetak SPP.

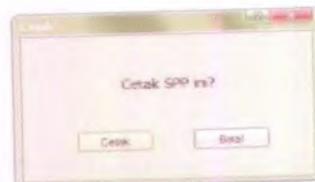
Cetak SPP



Tgl SPP	No. SPP	Jenis SPP	Nilai SPP	Pilihan
01-01-1970	0001	LS	50.000.000	<input type="button" value="Cetak"/>

Gambar 4.28 Cetak SPP

Gambar 4.29 menunjukkan tampilan konfirmasi apabila SPP akan dicetak ke printer dengan menekan tombol Cetak. Terdapat pilihan untuk kembali ke halaman Cetak SPP atau untuk mencetak data SPM ke printer.



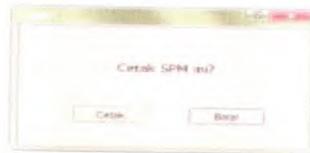
Gambar 4.29 Konfirmasi Cetak SPP

Gambar 4.30 memperlihatkan tampilan halaman Cetak SPM. Halaman ini menampilkan data SPM yang akan dicetak ke printer. Terdapat tombol Cetak yang berfungsi untuk proses cetak SPP. Sedangkan gambar 4.31 memperlihatkan tampilan konfirmasi apabila SPM akan dicetak ke printer dengan menekan tombol Cetak. Terdapat pilihan untuk kembali ke halaman Cetak SPM atau untuk mencetak data SPM ke printer.

Cetak SPM

Tgl SPM	No. SPM	Jenis SPM	Nilai SPM	Pilihan
01-01-1970	0001	LS	50.000.000	<input type="button" value="Cetak"/>

Gambar 4.30 Cetak SPM



Gambar 4.31 Konfirmasi Cetak SPM

Gambar 4.32 menunjukkan tampilan halaman RUH SP2D yang menampilkan data SPM yang siap untuk dicatat nomor SP2D. terdapat tombol Rekam untuk menuju ke halaman Rekam Nomor SP2D (Gambar 4.33), tombol Edit untuk menuju ke halaman Ubah Nomor SP2D (Gambar 4.35) dan tombol Hapus yang digunakan untuk menghapus nomor SP2D pada data SPM.

RUH Nomor SP2D

Tgl SPM	No. SPM	Jenis SPM	Nilai SPM	Pilihan
01-01-1970	0001	LS	50.000.000	Rekam Edit Hapus

Gambar 4.32 RUH Nomor SP2D

Gambar 4.33 memperlihatkan tampilan form pada halaman Rekam Nomor SP2D. Halaman tersebut digunakan untuk menambah nomor SP2D pada data SPM. Sedangkan gambar 4.34 menunjukkan tampilan form Rekam Nomor SP2D dengan tampilan kesalahan pada saat penginputan data.

Rekam No. SP2D

Tanggal SPM	01-01-1970
Nomor SPM	00001
Kata SPM	50.000.000
Tanggal SP2D	
Nomor SP2D	

Gambar 4.33 Rekam Nomor SP2D

Rekam No. SP2D

Tanggal SPM	01-01-1970
Nomor SPM	00001
Kata SPM	50.000.000
Tanggal SP2D	
Nomor SP2D	

Perintah Tanggal Tidak
 Field Harus Terisi

Gambar 4.34 Rekam Nomor SP2D – Peringatan Error

Gambar 4.35 menunjukkan tampilan form pada halaman Ubah Nomor SP2D. Halaman tersebut digunakan untuk mengedit nomor SP2D yang telah direkam pada data SPM.



Ubah No. SP2D

Tanggal SPM	01-01-1970
Nomor SPM	00001
Nilai SPM	50.000.000
Tanggal SP2D	01-01-1970
Nomor SP2D	2257084

Gambar 4.35 Ubah Nomor SP2D

Tampilan untuk konfirmasi penghapusan nomor SP2D pada halaman RUH Nomor SP2D (Gambar 4.32) sama dengan konfirmasi penghapusan Kontrak (Gambar 4.15). Gambar 4.36 menunjukkan halaman Kirim SPM yang menampilkan data SPM yang siap untuk dikirimkan ke KPPN. SPM akan dikirimkan dengan format berupa Arsip Data Komputer (ADK). Pada halaman tersebut terdapat tombol Kirim untuk mengirimkan dan menghasilkan ADK SPM yang dipilih.

Kirim SPM

Tgl SPM	No. SPM	Jenis SPM	Nilai SPM	Pilihan
01-01-1970	0001	LS	50.000.000	<input type="button" value="Kirim"/>

Gambar 4.36 Kirim SPM

Gambar 4.37 memperlihatkan tampilan konfirmasi apabila SPM akan dikirim dengan menekan tombol Kirim. Terdapat pilihan

untuk kembali ke halaman Kirim SPM atau untuk mengirimkan dan menghasilkan ADK SPM.



Gambar 4.37 Konfirmasi Kirim SPM

BAB V

IMPLEMENTASI DAN UJI COBA SISTEM

Bab ini menjelaskan tentang proses pada tahap implementasi serta uji coba aplikasi SIAPKAN. Penjelasan pada bab ini dimulai dari lingkungan implementasi, tahapan implementasi yang berupa kode program serta uji coba sistem.

5.1. Lingkungan Implementasi

Aplikasi SIAPKAN Modul Pembayaran dikembangkan dengan menggunakan *Personal Computer* (PC) yang memiliki spesifikasi perangkat keras dan lunak yang dapat dilihat pada Tabel 5.1 sebagai berikut.

Table 5.1 Spesifikasi Perangkat Keras dan Lunak

Perangkat	Spesifikasi
Personal Computer	Prosesor : AMD Phenom™ II X4 955 @3.2GHz
	Memori : 2 GB RAM
	Sistem Operasi : Windows 7

Selain itu untuk mengembangkan sistem menggunakan bahasa pemrograman berbasis Java dengan menggunakan JavaServer Pages dan Servlet. Piranti lunak (editor) yang digunakan adalah Netbeans IDE. Teknologi lain yang diimplementasikan dalam proses pengembangan aplikasi SIAPKAN dapat dilihat pada Tabel 5.2.

Table 5.2 Teknologi yang digunakan

Tools	
Websserver	Apache Tomcat 7.0.22
Bahasa Pemrograman	Java, JSP
Database	PostgreSQL 9.1

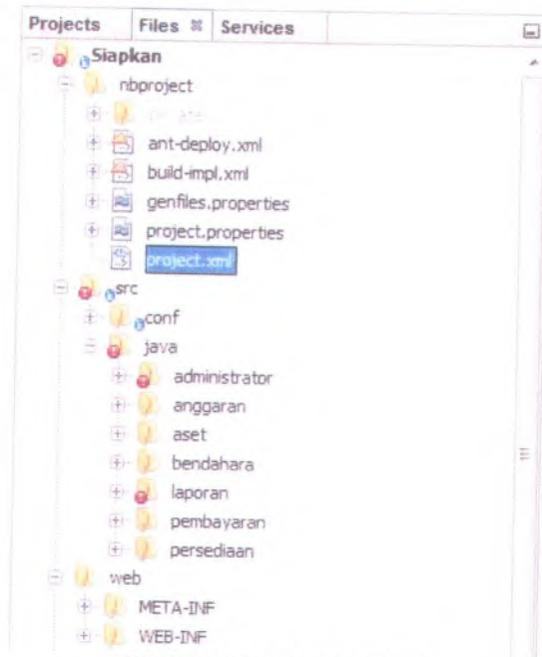
Database Management	PhpPgAdmin 5.0.2
Development Kit	JDK 1.6.0
Editor	Netbeans IDE 7.1.2
Library Pendukung	
Java Persistence	JPA Eclipselink 2.0
Subversion	Tortoise SVN 1.7.7

5.2. Konfigurasi

Pengembangan aplikasi SIAPKAN Modul Pembayaran memiliki tahapan meliputi meliputi pembuatan struktur direktori project, setting file konfigurasi serta setting package yang berisi file-file Java.

5.2.1. Pembuatan Struktur Direktori Project

Pembuatan struktur direktori project aplikasi SIAPKAN mengikuti standar pembuatan Web Application Java. Pada awalnya dibuat nama project terlebih dahulu. Karena Modul Pembayaran merupakan bagian dari project aplikasi secara keseluruhan, maka project dinamakan "SIAPKAN". Kemudian membuat sub-direktori 'src' sebagai tempat source file java yang akan dibuat. Sub-direktori kemudian dibagi sesuai masing-masing modul pada aplikasi. Selain itu juga dibuat sub-direktori 'build' sebagai tempat menyimpan file-file yang akan di-build menjadi file WAR. Selain itu juga dibuat sub-direktori 'web' sebagai tempat menyimpan file JSP. Sub-direktori ini kemudian dibagi sesuai dengan masing-masing modul. Struktur direktori yang dibuat diperlihatkan pada Gambar 5.1.



Gambar 5.1 Struktur Direktori

5.2.2. Konfigurasi File

Pada pengembangan aplikasi berbasis Java Web dibutuhkan beberapa file konfigurasi yang memiliki kegunaan masing-masing. File tersebut umumnya berupa file XML serta file dengan ekstensi “.properties”. beberapa file konfigurasi yang digunakan beserta fungsinya dapat dilihat pada Tabel. 5.3.

Table 5.3 Konfigurasi File

Nama File	Fungsi
build.xml	File yang berisi script untuk melakukan tugas build, deploy,


```

<servlet>
  <servlet-name>GajiServlet</servlet-name>
  <servlet-class>penjualan.servlet.GajiServlet</servlet-class>
</servlet>
<servlet>
  <servlet-name>EPPServlet</servlet-name>
  <servlet-class>penjualan.servlet.EPPServlet</servlet-class>
</servlet>
<servlet>
  <servlet-name>HibahMasukServlet</servlet-name>
  <servlet-class>persediaan.servlet.HibahMasukServlet</servlet-class>
</servlet>
<servlet>
  <servlet-name>KampasanServlet</servlet-name>
  <servlet-class>persediaan.servlet.KampasanServlet</servlet-class>
</servlet>
<servlet>
  <servlet-name>PerolehanLainServlet</servlet-name>
  <servlet-class>persediaan.servlet.PerolehanLainServlet</servlet-class>
</servlet>
<servlet>
  <servlet-name>FokServlet</servlet-name>
  <servlet-class>anagaram.servlet.FokServlet</servlet-class>
</servlet>
<servlet>
  <servlet-name>FeldsAvalAT</servlet-name>
  <servlet-class>aset.servlet.PedolehanATServlet</servlet-class>

```

Gambar 5.4 Konfigurasi file web.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<persistence xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://xmlns.jcp.org/xml/beans/1.0" xsi:schemaLocation="http://xmlns.jcp.org/xml/beans/1.0 http://xmlns.jcp.org/xml/beans/1.0/beans-1.0.xsd">
  <provider>org.eclipse.persistence.jpa.PersistenceProvider</provider>
  <class>kendahara.entity.Truck</class>
  <class>kendahara.entity.Mop</class>
  <class>kendahara.entity.Dip</class>
  <class>kendahara.entity.BBakti</class>
  <class>kendahara.entity.DSvrt</class>
  <class>kendahara.entity.DSlpp</class>
  <class>kendahara.entity.DSoktipe</class>
  <class>aset.entity.Kmp</class>
  <class>entity.Truck</class>
  <class>aset.entity.Ruang</class>
  <class>aset.entity.Dip</class>
  <class>aset.entity.Dik</class>
  <class>aset.entity.Ptsh</class>
  <class>aset.entity.Kang</class>
  <class>aset.entity.Xbal</class>

```

Gambar 5.5 Konfigurasi file persistence.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<Context antiJARLocking="true" path="/Siapkan"/>
</Context>

```

Gambar 5.6 Konfigurasi file context.xml

5.3. Penulisan Kode Program

Pada tahap ini merupakan tahapan penulisan kode program. Penulisan dilakukan pada sub-direktori 'src'. Penulisan kode dipisahkan oleh beberapa package sesuai dengan fungsi masing-masing agar lebih terstruktur dan rapi.

5.3.1. Package 'pembayaran.entity'

Package Entity ini berisikan file-file Java yang merupakan class untuk digunakan serta merepresentasikan seluruh data yang digunakan dalam aplikasi ini. Class ini memuat tabel yang terdapat pada database yang nantinya akan digunakan untuk menampung seluruh data yang diinput menggunakan aplikasi, dan juga seluruh data yang akan ditampilkan pada view. Gambar 5.7 memperlihatkan potongan kode program salah satu file java yang ada dalam package, yaitu Gaji.java.

```

public class Gaji implements Serializable {
    private static final long serialVersionUID = 1L;
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Id
    private Long id;
    @Column(name = "namauser")
    private String username;
    @Column(name = "password")
    private Long password;
    @ManyToMany
    private List<User> users;
    @ManyToMany
    private List<User> users;
    @OneToMany(mappedBy = "Gaji")
    private List<Transaksi> transaksi;
    public Gaji() {
    }
    public String getUsername() {
        return username;
    }
}

```

Gambar 5.7 Potongan kode Gaji.java

5.3.2. Package 'pembayaran.model'

Package model ini berisikan file-file java yang merupakan class dan digunakan sebagai penghubung antara class yang terdapat pada package 'pembayaran.entity' dengan class controller yang terdapat pada package 'pembayaran.servlet'. Class ini memuat method-method yang digunakan untuk memanggil data yang berasal dari class entity untuk diproses menuju class controller agar dapat ditampilkan dalam view. Gambar 5.8 menunjukkan potongan kode program dari file DaftarGaji.java. Class tersebut berisi method yang digunakan untuk menampilkan data gaji yang terdapat dalam entity Gaji.java.

```

public class DaftarGaji {

    public DaftarGaji() {
        sw = DriverManager.getConnection("jdbc:mysql://localhost:3306/");
    }

    private EntityManagerFactory emf = null;

    public EntityManager getEntityManager() {
        return emf.createEntityManager();
    }

    public boolean checkGaji(String nikays) {
        boolean result = false;
        EntityManager em = getEntityManager();
        try {
            Query q = em.createQuery("select * from gaji where nikays = :nikays");
            q.setParameter("nikays", nikays);
            int result = (int) q.getResultList().size();
            if (result > 0) {
                result = true;
            }
        } finally {
            if (em != null) {
                em.close();
            }
        }
        return result;
    }

    public boolean getGaji() {
        List<Gaji> ganseksi = new ArrayList<Gaji>();

        EntityManager em = getEntityManager();
        try {
            Query q = em.createQuery("select * from gaji");
            ganseksi = q.getResultList();
        } finally {
            em.close();
        }
        return ganseksi;
    }
}

```

Gambar 5.8 Potongan kode DaftarGaji.java

5.3.3.Package 'pembayaran.servlet'

Package ini berisikan file-file java yang merupakan class dan berfungsi sebagai controller pada aplikasi. Controller berfungsi untuk menangani *request* serta mengembalikan *response* ke user client. Isi dari controller pada aplikasi ini adalah method yang berfungsi untuk merekam, membaca, memperbarui serta menghapus data. Selain itu pada package ini juga terdapat file yang merupakan class servlet dan digunakan untuk menghubungkan antara class controller dengan view. Gambar 5.9 menunjukkan potongan kode program pada file GajiController.java

```
public class GajiController {
    HttpServletRequest request;

    public GajiController(HttpServletRequest request) {
        this.request = request;
    }
    int offset;
    int length;
    DaftarGaji daftarGaji = new DaftarGaji();
    List<Gaji> listGaji = daftarGaji.getGaji();

    public void TestList(int offset, int length) {
        this.length = length;
        this.offset = offset;
    }

    public ArrayList getListByOffsetAndLength() {
        DaftarGaji daftarGaji = new DaftarGaji();
        List<Gaji> listGaji = daftarGaji.getGaji();
        ArrayList arrayList = new ArrayList();
        int to = this.offset + this.length;
        if (this.offset > listGaji.size()) {
            this.offset = listGaji.size();
        }
        if (to > listGaji.size()) {
            to = listGaji.size();
        }
        for (int i = this.offset; i < to; i++) {
            arrayList.add(listGaji.get(i));
        }
        return arrayList;
    }
}
```

Gambar 5.9 Potongan kode GajiController.java

Gambar 5.10 menunjukkan potongan kode program pada file GajiServlet.java. File ini digunakan sebagai class untuk mengarahkan method dari controller untuk dikirimkan ke view.

```

public class GajiServlet extends HttpServlet {

    // ...

    request = request;
    response = response;

    protected void processRequest(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException, HttpSessionBindingException {
        String pathInfo = request.getPathInfo();
        HttpSession session = request.getSession();

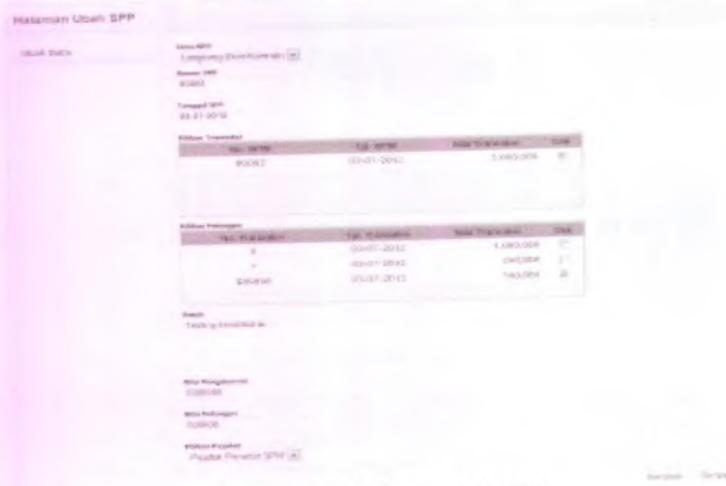
        if (pathInfo.equals("/home")) {
            session.removeAttribute("username");
            session.removeAttribute("password");
            session.removeAttribute("password2");
            request.getRequestDispatcher("/home").forward(request, response);
        } else if (pathInfo.equals("/login")) {
            session.removeAttribute("username");
            session.removeAttribute("password");
            GajiController gajiController = new GajiController(request);
            request.getRequestDispatcher(gajiController.processLogin()).forward(request, response);
        } else if (pathInfo.equals("/logout")) {
            session.removeAttribute("username");
            GajiController gajiController = new GajiController(request);
            request.getRequestDispatcher(gajiController.logout()).forward(request, response);
        } else if (pathInfo.equals("/register")) {
            session.removeAttribute("username");
            session.removeAttribute("password");
            GajiController gajiController = new GajiController(request);
            request.getRequestDispatcher(gajiController.register()).forward(request, response);
        } else if (pathInfo.equals("/update")) {
            session.removeAttribute("username");
            session.removeAttribute("password");
            GajiController gajiController = new GajiController(request);
            request.getRequestDispatcher(gajiController.update()).forward(request, response);
        }
    }
}

```

Gambar 5.10 Potongan kode Gajiservlet.java

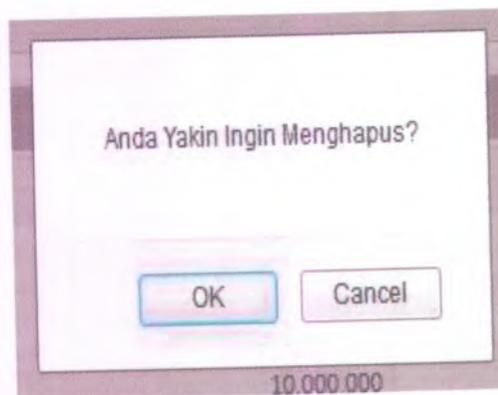
5.3.4. Pembuatan View

Pembuatan view untuk aplikasi SIAPKAN ini menggunakan JavaServer Pages (JSP) yang didukung dengan menggunakan Cascading Style Sheet (CSS) untuk mendukung tampilan serta Javascript dan JQuery untuk memperkuat validasi pada sisi client. Gambar 5.11 menunjukkan potongan kode program pada halaman rekam_spp.jsp. Sedangkan pada Gambar 5.12 menunjukkan validasi untuk sisi client pada halaman rekam_spp.jsp.



Gambar 5.15 Halaman ubah SPP

Gambar 5.16 memperlihatkan pesan konfirmasi 'hapus' yang merupakan pesan yang muncul apabila user meng-klik tombol hapus pada data yang ada di halaman daftar.



Gambar 5.16 Pesan konfirmasi hapus data



5.4.Integrasi Modul

Tahapan ini dilakukan untuk mengintegrasikan seluruh modul yang telah dikembangkan dalam aplikasi SIAPKAN menjadi satu aplikasi yang utuh. Hal ini dilakukan dengan mengintegrasikan modul-modul yang saling berkaitan, sesuai dengan proses bisnis organisasi. Modul Pembayaran melakukan proses integrasi dengan modul Administrasi, Anggaran serta Bendahara. Ketiga modul tersebut mempunyai proses bisnis yang berkesinambungan dalam organisasi.

Integrasi yang dilakukan dengan modul Administrasi mencakup use case 'Login', 'Logout' serta 'Referensi'. User harus melakukan login terlebih dahulu sebelum masuk ke dalam seluruh proses yang ada pada Modul Pembayaran. Begitu juga ketika user melakukan logout, maka tidak dapat masuk Modul Pembayaran sebelum melakukan login ulang. Pada Modul Pembayaran, seluruh referensi yang dibutuhkan terdapat pada Modul Administrasi, misalnya kode satker, pejabat penandatanganan, dan lain-lain. Untuk mengakses data referensi tersebut, maka Modul Pembayaran harus mengambil data yang terdapat pada modul Administrasi.

Untuk integrasi yang dilakukan dengan modul Anggaran mencakup data DIPA serta POK yang merupakan komponen dari SPP dan SPM pada Modul Pembayaran. Untuk dapat mengakses data DIPA dan POK maka Modul Pembayaran harus mengambil data yang terdapat pada modul Anggaran. Sedangkan untuk integrasi pada modul Bendahara mencakup data Transaksi yang juga merupakan komponen dari SPP dan SPM. Sama halnya dengan modul Anggaran dan Administrasi, Modul Pembayaran harus mengambil data yang terdapat pada Modul Bendahara.

5.5. Uji Coba

Uji coba yang dilakukan pada aplikasi SIAPKAN terdiri atas uji coba fungsional dan uji coba non fungsional. Uji coba fungsional dilakukan dengan mengacu pada test case yang dibuat pada tahap desain dan kemudian membandingkan apakah aplikasi yang dikembangkan telah sesuai dengan test case. Uji coba non fungsional dilakukan dengan cara menguji keandalan aplikasi terhadap pemakaian dan jumlah request yang ada. Berikut ini dipaparkan hasil uji coba fungsional.

Table 5.4 Hasil Uji Coba Fungsionalitas

No.	Test Case	Status
TC01	Rekam Kontrak	Terpenuhi
TC02	Ubah Kontrak	Terpenuhi
TC03	Hapus Kontrak	Terpenuhi
TC04	Rekam Gaji	Terpenuhi
TC05	Ubah Gaji	Terpenuhi
TC06	Hapus Gaji	Terpenuhi
TC07	Rekam SPP	Terpenuhi
TC08	Ubah SPP	Terpenuhi
TC09	Hapus SPP	Terpenuhi
TC10	Setuju SPM	Terpenuhi
TC11	Batal SPM	Terpenuhi
TC12	Cetak SPP	Terpenuhi
TC13	Cetak SPM	Terpenuhi
TC14	Rekam Nomor SP2D	Terpenuhi
TC15	Ubah Nomor SP2D	Terpenuhi
TC16	Hapus Nomor SP2D	Terpenuhi
TC17	Kirim SPM	Terpenuhi

5.6. Analisis Uji Coba

Pada tahapan ini analisis uji coba yang dilakukan hanya berdasarkan uji coba fungsionalitas yang telah dilakukan. Hasil uji coba fungsional menunjukkan bahwa aplikasi ini telah sesuai dengan seluruh test case yang telah dibuat pada tahapan desain. Hal ini menunjukkan bahwa aplikasi SIAPKAN Modul Pembayaran secara fungsional telah berjalan dengan baik.

BAB VI

KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dan saran dari seluruh proses pengerjaan tugas akhir. Kesimpulan dan saran diharapkan berguna untuk proses pengembangan selanjutnya.

6.1. Kesimpulan

Berdasarkan hasil penelitian tugas akhir ini, maka dapat disimpulkan sebagai berikut:

1. Aplikasi Sistem Informasi Pengelolaan Keuangan Negara (SIAPKAN) Modul Pembayaran telah berhasil dikembangkan dengan baik. Hal ini terbukti dengan terpenuhinya seluruh test case yang telah dirancang untuk uji coba fungsionalitas dari aplikasi.
2. Aplikasi SIAPKAN mempermudah Satuan Kerja dalam menerbitkan Surat Permintaan Pembayaran dan Surat Perintah Membayar sebagai dasar pencairan dana APBN.

6.2. Saran

Beberapa saran yang dapat dipertimbangkan untuk penelitian lebih lanjut adalah sebagai berikut :

1. Perlunya pengembangan lebih lanjut pada aplikasi, khususnya untuk penerbitan SPP dan SPM Gaji Induk, karena belum mencakup pembayaran langsung kepada masing-masing pegawai.
2. Dibutuhkannya modul Komitmen yang berfungsi sebagai modul manajemen rekanan / pihak ketiga, agar proses pencairan dana serta manajemen data pihak ketiga lebih terpenuhi
3. Proses manajemen data untuk pembayaran Gaji perlu dibuatkan modul tersendiri, atau mengikuti modul

Bendahara agar proses penerbitan daftar gaji serta komponen pendukung gaji lainnya menjadi lebih baik.

4. Pada proses bisnis yang ada saat ini, pencatatan transaksi pencairan dana masih berbasis kas. Perlu dikembangkan proses bisnis agar pencatatan transaksi pengeluaran serta penerimaan berbasis akrual.

DAFTAR PUSTAKA

- Bergsten, Hans. 2004.** *JavaServer Pages*. Sebastopol : O'Reilly Media, Inc., 2004.
- Direktur Jenderal Perbendaharaan. 2005.** Peraturan Direktur Jenderal Perbendaharaan Nomor PER-66/PB/2005 Tentang Mekanisme Pelaksanaan Pembayaran Atas Beban Anggaran Pendapatan dan Belanja Negara. Jakarta : s.n., 2005.
- Hall, Marty. 2000.** *Core Servlets and JavaServer Pages*. Upper Saddle River : Prentice-Hall, Inc, 2000.
- Kementerian Keuangan. 2005.** Peraturan Kementerian Keuangan Nomor 134 tentang Pedoman Pembayaran dan Pelaksanaan Anggaran Pendapatan dan Belanja Negara. Jakarta : s.n., 2005.
- Matthew, Neil and Stones, Richard. 2005.** *Beginning Databases with PostgreSQL: From Novice to Professional, Second Edition*. Berkeley : Apress, 2005.
- Perry, Bruce W. 2004.** *Java Servlet and JSP Cookbook*. Sebastopol : O'Reilly Media, Inc, 2004.
- Presiden Republik Indonesia. 2004.** Undang-Undang Republik Indonesia Nomor 1 tentang Perbendaharaan Negara. Jakarta : s.n., 2004.
- .**2003. Undang-Undang Republik Indonesia Nomor 17 tentang Keuangan Negara. Jakarta : s.n., 2003.

Rosenberg, Matt. 2007. *Use Case Driven Object Modelling With UML*. Berkeley : Apress, 2007.

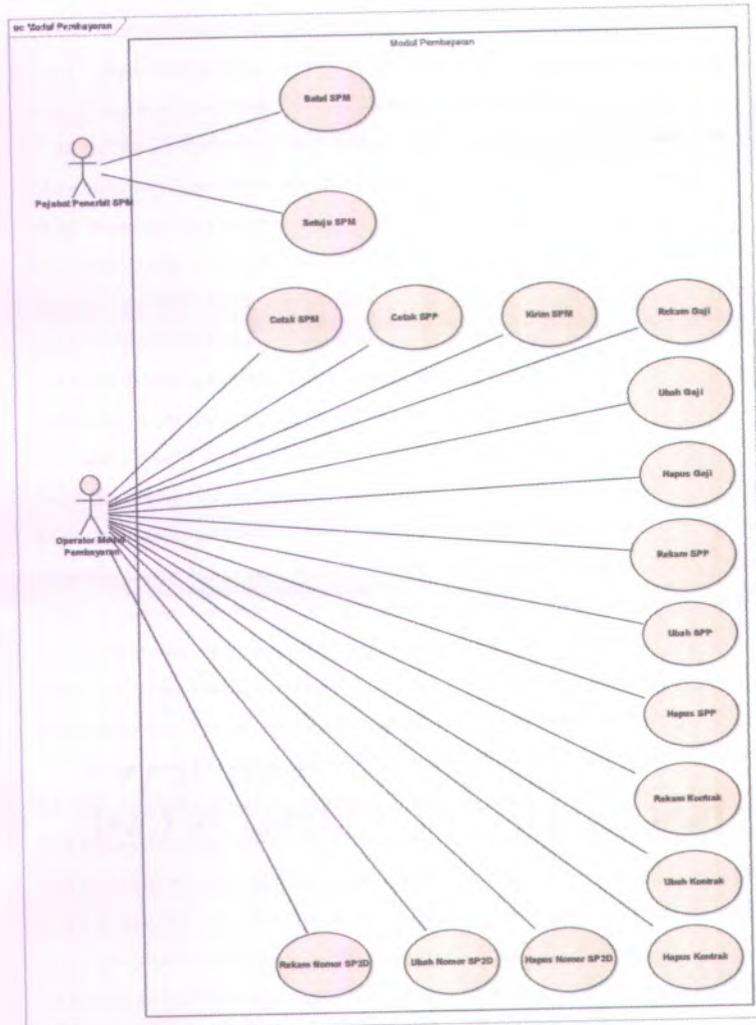
Seshadri, Govind. 1999. Server-side Java: Understanding JavaServer Pages Model 2 architecture. Javaworld. [Online] 1999.

Sun Microsystems. JavaServer Pages. [Online]
<http://www.kirkdorffer.com/jspspecs/jsp092.html#model>.

Van Haecke, Bernard. 2002. *JDBC3.0 Java Database Connectivity*. New York : Hungry Minds, Inc, 2002.

LAMPIRAN A
DIAGRAM DAN DESKRIPSI USE CASE

A.1. Diagram Use Case



Gambar A.1. Diagram Use Case

A.2. Deskripsi Use Case Rekam Kontrak

Tabel A.1. Deskripsi Use Case Rekam Kontrak

<i>UC01</i> – Rekam Kontrak	
Primary Actor: Operator Modul Pembayaran	Level: User Goal
Pre-conditions: Operator Modul Pembayaran berada di halaman RUH Kontrak.	
Triggers: <ul style="list-style-type: none"> • Operator Modul Pembayaran mengklik tombol Rekam Data. 	
Basic course: Sistem menampilkan halaman Rekam Kontrak. Operator Modul Pembayaran melakukan input serta memilih pada field dan pilihan yang ada lalu menekan tombol Simpan.	
Post-conditions: Sistem akan memastikan bahwa seluruh field dan pilihan telah terisi lalu menyimpan pada database.	
Alternate courses: <p>Jika seluruh field belum terisi: sistem akan menampilkan peringatan bahwa field tidak boleh kosong</p> <p>Jika isian field Nilai Kontrak bukan angka dan bilangan bulat: sistem akan menampilkan peringatan bahwa isian harus angka dan bilangan bulat</p> <p>Jika isian field Nilai Kontrak adalah 0: sistem akan</p>	

menampilkan peringatan bahwa isian tidak boleh 0
Jika isian field Tanggal Kontrak bukan berupa format tanggal: sistem akan menampilkan peringatan bahwa isian harus format tanggal.
Jika Nomor Kontrak sudah ada pada database: sistem akan menampilkan peringatan bahwa kontrak sudah pernah direkam

A.3. Deskripsi Use Case Ubah Kontrak

Tabel A.2. Deskripsi Use Case Ubah Kontrak

<i>UC02</i> –Ubah Kontrak	
Primary Actor: Operator Modul Pembayaran	Level: User Goal
Pre-conditions: Operator Modul Pembayaran berada di halaman RUH Kontrak	
Triggers: <ul style="list-style-type: none"> Operator Modul Pembayaran memilih Kontrak yang akan diubah lalu mengklik tombol Ubah 	
Basic course: Sistem menampilkan halaman Ubah Kontrak. Operator Modul Pembayaran melakukan edit data pada field yang ada lalu menekan tombol Simpan.	
Post-conditions: Sistem akan memastikan bahwa seluruh field telah terisi lalu menyimpan pada database.	

Alternate courses:

Jika seluruh field belum terisi: sistem akan menampilkan peringatan bahwa field tidak boleh kosong

Jika isian field Nilai Kontrak bukan angka dan bilangan bulat: sistem akan menampilkan peringatan bahwa isian harus angka dan bilangan bulat

Jika isian field Nilai Kontrak adalah 0: sistem akan menampilkan peringatan bahwa isian tidak boleh 0.

Jika isi field Tanggal Kontrak bukan berupa format tanggal: sistem akan menampilkan peringatan bahwa isian harus format tanggal.

Jika Nomor Kontrak sudah ada pada database: sistem akan menampilkan peringatan bahwa kontrak sudah pernah direkam

A.4. Deskripsi Use Case Hapus KontrakTabel A.3. Deskripsi *Use Case* Hapus Kontrak

<i>UC03</i> – Hapus Kontrak	
Primary Actor: Operator Modul Pembayaran	Level: User Goal
Pre-conditions: Operator Modul Pembayaran berada di halaman RUH Kontrak.	
Triggers: <ul style="list-style-type: none"> • Operator Modul Pembayaran memilih data Kontrak yang akan dihapus lalu mengklik tombol Hapus. 	
Basic course:	

Sistem menampilkan konfirmasi penghapusan Kontrak. Operator Modul Pembayaran memilih Hapus.
<p>Post-conditions:</p> <p>Sistem akan menghapus Kontrak pada database dan me-redirect ke halaman R/U/H Kontrak.</p>
<p>Alternate courses:</p> <p>Jika Kontrak telah terhubung dengan SPP: sistem akan menampilkan peringatan bahwa Kontrak sudah terhubung dengan SPP</p> <p>Jika pada konfirmasi penghapusan menekan tombol Batal: sistem akan me-redirect ke halaman R/U/H Kontrak tanpa menghapus Kontrak pada database.</p>

A.5. Deskripsi Use Case Rekam Gaji

Tabel A.4. Deskripsi Use Case Rekam Gaji

<i>UC04</i> – Rekam Gaji	
<p>Primary Actor: Operator Modul Pembayaran</p>	<p>Level: User Goal</p>
<p>Pre-conditions:</p> <p>Operator Modul Pembayaran berada di halaman RUH Gaji.</p>	
<p>Triggers:</p> <ul style="list-style-type: none"> • Operator Modul Pembayaran mengklik tombol Rekam Data. 	

<p>Basic course:</p> <p>Operator Modul Pembayaran melakukan input serta memilih pada field dan pilihan yang ada lalu menekan tombol Simpan.</p>
<p>Post-conditions:</p> <p>Sistem akan memastikan bahwa seluruh field telah terisi lalu menyimpan pada database.</p>
<p>Alternate courses:</p> <p>Jika seluruh field belum terisi: sistem akan menampilkan peringatan bahwa field tidak boleh kosong</p> <p>Jika isian field bukan angka dan bilangan bulat: sistem akan menampilkan peringatan bahwa isian harus angka dan bilangan bulat</p> <p>Jika isian field adalah 0: sistem akan menampilkan peringatan bahwa isian tidak boleh 0</p> <p>Jika field Potongan lebih besar dari field Gaji Kotor: sistem akan menampilkan peringatan bahwa potongan lebih besar</p> <p>Jika Bulan Pembayaran sudah ada pada database: sistem akan menampilkan peringatan bahwa Gaji sudah pernah direkam</p>

A.6. Deskripsi Use Ubah Gaji

Tabel A.5. Deskripsi Use case Ubah Gaji

UC05 – Ubah Gaji	
<p>Primary Actor: Operator Modul Pembayaran</p>	<p>Level: User Goal</p>
Pre-conditions:	

Operator Modul Pembayaran berada di halaman RUH Gaji.
Triggers: <ul style="list-style-type: none">• Operator Modul Pembayaran memilih data Gaji yang akan diubah lalu mengklik tombol Ubah.
Basic course: <p>Sistem menampilkan halaman Ubah Gaji. Operator Modul Pembayaran melakukan edit data pada field yang ada lalu menekan tombol Simpan.</p>
Post-conditions: <p>Sistem akan memastikan bahwa seluruh field telah terisi lalu menyimpan pada database.</p>
Alternate courses: <p>Jika seluruh field belum terisi: sistem akan menampilkan peringatan bahwa field tidak boleh kosong</p> <p>Jika isian field bukan angka dan bilangan bulat: sistem akan menampilkan peringatan bahwa isian harus angka dan bilangan bulat</p> <p>Jika isian field adalah 0: sistem akan menampilkan peringatan bahwa isian tidak boleh 0</p> <p>Jika field Potongan lebih besar dari field Gaji Kotor: sistem akan menampilkan peringatan bahwa potongan lebih besar</p> <p>Jika Bulan Pembayaran sudah ada pada database: sistem akan menampilkan peringatan bahwa Gaji sudah pernah direkam</p>

A.7. Deskripsi Use Case Hapus Gaji

Tabel A.6. Deskripsi Use Case Hapus Gaji

<i>UC06</i> – Hapus Gaji	
Primary Actor: Operator Modul Pembayaran	Level: User Goal
Pre-conditions: Operator Modul Pembayaran berada di halaman RUH Gaji.	
Triggers: <ul style="list-style-type: none"> • Operator Modul Pembayaran memilih data Gaji yang akan dihapus lalu mengklik tombol Hapus. 	
Basic course: Sistem menampilkan konfirmasi penghapusan Gaji. Operator Modul Pembayaran memilih Hapus.	
Post-conditions: Sistem akan menghapus Gaji pada database dan me-redirect ke halaman R/U/H Gaji.	
Alternate courses: <p>Jika Gaji telah terhubung dengan SPP: sistem akan menampilkan peringatan bahwa Gaji sudah terhubung dengan SPP</p> <p>Jika pada konfirmasi penghapusan menekan tombol Batal: sistem akan me-redirect ke halaman R/U/H Gaji tanpa menghapus Gaji pada database</p>	



A.8. Deskripsi Use Case Rekam SPP

Tabel A.7. Deskripsi Use Case Rekam SPP

<i>UC07</i> – Rekam SPP	
Primary Actor: Operator Modul Pembayaran	Level: User Goal
Pre-conditions: Operator Modul Pembayaran berada di halaman RUH SPP.	
Triggers: <ul style="list-style-type: none"> • Operator Modul Pembayaran mengklik tombol Rekam Data. 	
Basic course: Sistem menampilkan halaman Rekam SPP. Operator Modul Pembayaran melakukan input serta memilih pada field dan pilihan yang ada lalu menekan tombol Simpan.	
Post-conditions: Sistem akan memastikan bahwa seluruh field dan pilihan telah terisi lalu menyimpan pada database.	
Alternate courses: <p>Jika seluruh field dan pilihan belum terisi: sistem akan menampilkan peringatan bahwa field tidak boleh kosong</p> <p>Jika isian field Pengeluaran dan Potongan bukan angka dan</p>	

bilangan bulat: sistem akan menampilkan peringatan bahwa isian harus angka dan bilangan bulat

Jika isian field Pengeluaran dan Potongan tidak sama dengan nilai pilihan Transaksi dan Potongan: sistem akan menampilkan peringatan bahwa nilai field Pengeluaran dan Potongan tidak sama dengan pilihan

Jika isian field Potongan lebih besar dari Pengeluaran: sistem akan menampilkan peringatan bahwa nilai field potongan tidak boleh lebih besar dari field pengeluaran

Jika Nomor SPP sudah ada pada database: sistem akan menampilkan peringatan bahwa SPP sudah pernah direkam

A.9. Deskripsi Use Case Ubah SPP

Tabel A.8. Deskripsi Use Case Ubah SPP

<i>UC08</i> – Ubah SPP	
Primary Actor: Operator Modul Pembayaran	Level: User Goal
Pre-conditions: Operator Modul Pembayaran berada di halaman RUH SPP.	
Triggers: <ul style="list-style-type: none"> • Operator Modul Pembayaran memilih data SPP yang akan diubah lalu mengklik tombol Ubah. 	
Basic course: Sistem kemudian menampilkan halaman Ubah SPP. Operator Modul Pembayaran melakukan edit data pada field yang ada lalu menekan tombol Simpan.	

<p>Post-conditions: Sistem akan memastikan bahwa seluruh field telah terisi lalu menyimpan pada database</p>
<p>Alternate courses:</p> <p>Jika seluruh field dan pilihan belum terisi: sistem akan menampilkan peringatan bahwa field tidak boleh kosong</p> <p>Jika isian field Pengeluaran dan Potongan bukan angka dan bilangan bulat: sistem akan menampilkan peringatan bahwa isian harus angka dan bilangan bulat</p> <p>Jika isian field Pengeluaran dan Potongan tidak sama dengan nilai pilihan Transaksi dan Potongan: sistem akan menampilkan peringatan bahwa nilai field Pengeluaran dan Potongan tidak sama dengan pilihan</p> <p>Jika isian field Potongan lebih besar dari Pengeluaran: sistem akan menampilkan peringatan bahwa nilai field potongan tidak boleh lebih besar dari field pengeluaran</p> <p>Jika Nomor SPP sudah ada pada database: sistem akan menampilkan peringatan bahwa SPP sudah pernah direkam</p>

A.10. Deskripsi Use Case Hapus SPP

Tabel A.9. Deskripsi Use Case Hapus SPP

<i>UC09</i> – Hapus SPP	
<p>Primary Actor: Operator Modul Pembayaran</p>	<p>Level: User Goal</p>
<p>Pre-conditions:</p> <p>Operator Modul Pembayaran berada di halaman RUH SPP.</p>	

<p>Triggers:</p> <ul style="list-style-type: none"> • Operator Modul Pembayaran memilih SPP yang akan dihapus lalu mengklik tombol Hapus.
<p>Basic course:</p> <p>Sistem kemudian menampilkan konfirmasi penghapusan SPP. Operator Modul Pembayaran memilih Hapus.</p>
<p>Post-conditions:</p> <p>Sistem akan menghapus SPP pada database dan me-redirect ke halaman R/U/H SPP</p>
<p>Alternate courses:</p> <p>Jika SPP telah disetujui menjadi SPM: sistem akan menampilkan peringatan bahwa SPP telah disetujui menjadi SPM Jika SPP telah dicatat SP2D: sistem akan menampilkan peringatan bahwa SPP telah dicatat SP2D Jika pada konfirmasi penghapusan menekan tombol Batal: sistem akan me-redirect ke halaman R/U/H SPP tanpa menghapus SPP pada database</p>

A.11. Deskripsi Use Case Setuju SPM

Tabel A.10. Deskripsi Use Case Setuju SPM

UC10 – Setuju SPM	
<p>Primary Actor: Pejabat Penerbit SPM</p>	<p>Level: User Goal</p>
Pre-conditions:	

Pejabat Penerbit SPM berada di halaman Setuju SPM.
<p>Triggers:</p> <ul style="list-style-type: none"> • Pejabat Penerbit SPM memilih salah satu SPP lalu menekan tombol Proses
<p>Basic course:</p> <p>Sistem akan menampilkan konfirmasi persetujuan SPM. Operator Modul Pembayaran memilih Proses.</p>
<p>Post-conditions:</p> <p>Sistem memproses database SPP yang dipilih dan me-redirect ke halaman Setuju SPM.</p>
<p>Alternate courses:</p> <p>Jika menekan tombol Batal pada konfirmasi persetujuan: sistem akan menampilkan kembali halaman Setuju SPM tanpa memproses SPM yang dipilih pada database.</p>

A.12. Deskripsi Use Case Batal SPM

Tabel A.11. Deskripsi Use Case Batal SPM

<i>UC11</i> – Batal SPM	
Primary Actor: Pejabat Penerbit SPM	Level: User Goal
Pre-conditions: Pejabat Penerbit SPM berada di halaman Batal SPM.	

<p>Triggers:</p> <ul style="list-style-type: none"> • Pejabat Penerbit SPM memilih salah satu SPM lalu menekan tombol Proses
<p>Basic course:</p> <p>Sistem akan menampilkan konfirmasi pembatalan SPM. Operator Modul Pembayaran memilih Proses.</p>
<p>Post-conditions:</p> <p>Sistem memproses database SPP yang dipilih dan me-redirect ke halaman Batal SPM.</p>
<p>Alternate courses:</p> <p>Jika menekan tombol Batal pada konfirmasi pembatalan: sistem akan menampilkan kembali halaman Batal SPM tanpa memproses Konsep SPM yang dipilih pada database.</p>

A.13. Deskripsi Use Case Cetak SPP

Tabel A.12. Deskripsi Use Case Cetak SPP

UC12 – Cetak SPP	
<p>Primary Actor: Operator Modul Pembayaran</p>	<p>Level: User Goal</p>
<p>Pre-conditions:</p> <p>Operator Modul Pembayaran berada di halaman Cetak SPP.</p>	

<p>Triggers:</p> <ul style="list-style-type: none"> • Operator Modul Pembayaran memilih data SPP yang akan dicetak lalu mengklik tombol Cetak.
<p>Basic course:</p> <p>Sistem akan menampilkan konfirmasi cetak SPP. Operator Modul Pembayaran memilih Proses.</p>
<p>Post-conditions:</p> <p>Sistem mengirimkan paket data ke printer untuk mencetak SPP.</p>
<p>Alternate courses:</p> <p>Jika pada konfirmasi menekan Batal: sistem akan menampilkan kembali halaman Cetak SPP tanpa mencetak SPP terlebih dahulu.</p>

A.14. Deskripsi Use Case Cetak SPM

Tabel A.13. Deskripsi Use Case Cetak SPM

UC13 – Cetak SPM	
<p>Primary Actor: Operator Modul Pembayaran</p>	<p>Level: User Goal</p>
<p>Pre-conditions:</p> <p>Operator Modul Pembayaran berada di halaman Cetak SPM.</p>	
<p>Triggers:</p>	

- Operator Modul Pembayaran memilih data SPM yang akan dicetak lalu mengklik tombol Cetak.

Basic course:

Sistem akan menampilkan konfirmasi cetak SPM. Operator Modul Pembayaran memilih Proses.

Post-conditions:

Sistem mengirimkan paket data ke printer untuk mencetak SPM.

Alternate courses:

Jika pada konfirmasi menekan Batal: sistem akan menampilkan kembali halaman Cetak SPP tanpa mencetak SPP terlebih dahulu.

A.15. Deskripsi Use Case Rekam Nomor SP2D

Tabel A.14. Deskripsi Use Case Rekam Nomor SP2D

<i>UC14</i> – Rekam Nomor SP2D	
Primary Actor: Operator Modul Pembayaran	Level: User Goal
Pre-conditions: Operator Modul Pembayaran berada di halaman RUH Nomor SP2D.	
Triggers:	

<ul style="list-style-type: none"> • Operator Modul Pembayaran memilih data SPM yang akan dicatat nomor SP2D lalu menekan tombol Rekam.
<p>Basic course:</p> <p>Sistem menampilkan halaman Rekam Nomor SP2D. Operator Modul Pembayaran melakukan input pada field yang ada lalu menekan tombol Simpan.</p>
<p>Post-conditions:</p> <p>Sistem akan memastikan bahwa seluruh field telah terisi lalu menyimpan pada database</p>
<p>Alternate courses:</p> <p>Jika seluruh field belum terisi: sistem akan menampilkan peringatan bahwa field tidak boleh kosong Jika Nomor SP2D sudah ada: sistem akan menampilkan peringatan bahwa Nomor SP2D sudah pernah dicatat</p>

A.16. Deskripsi Use Case Ubah Nomor SP2D

Tabel A.15. Deskripsi Use Case Ubah SP2D

UC15 – Ubah Nomor SP2D	
<p>Primary Actor: Operator Modul Pembayaran</p>	<p>Level: User Goal</p>
<p>Pre-conditions:</p> <p>Operator Modul Pembayaran berada di halaman RUH Nomor SP2D.</p>	
<p>Triggers:</p>	

- Operator Modul Pembayaran memilih data SPM yang akan diubah nomor SP2D lalu menekan tombol Ubah.

Basic course:

Sistem menampilkan halaman Ubah SP2D. Operator Modul Pembayaran melakukan edit data pada field yang ada lalu menekan tombol Simpan.

Post-conditions:

Sistem akan memastikan bahwa seluruh field telah terisi lalu menyimpan pada database.

Alternate courses:

Jika seluruh field belum terisi: sistem akan menampilkan peringatan bahwa field tidak boleh kosong

Jika Nomor SP2D sudah ada: sistem akan menampilkan peringatan bahwa Nomor SP2D sudah pernah dicatat

Jika Status SPM sudah Closing pada Modul Pelaporan: sistem akan menampilkan peringatan bahwa SPM sudah Closing

A.17. Deskripsi Use Case Hapus Nomor SP2D

Tabel A.16. Deskripsi Use Case Hapus Nomor SP2D

UC16 – Hapus Nomor SP2D	
Primary Actor: Operator Modul Pembayaran	Level: User Goal
Pre-conditions:	

<i>Operator Modul Pembayaran berada di halaman RUH Nomor SP2D.</i>
<p>Triggers:</p> <ul style="list-style-type: none"> • Operator Modul Pembayaran memilih data SPM yang akan dihapus nomor SP2D lalu menekan tombol Hapus.
<p>Basic course:</p> <p>Sistem kemudian menampilkan konfirmasi penghapusan nomor SP2D. Operator Modul Pembayaran memilih Hapus.</p>
<p>Post-conditions:</p> <p>Sistem akan menghapus nomor SP2D di SPP pada database dan me-redirect ke halaman R/U/H Nomor SP2D.</p>
<p>Alternate courses:</p> <p>Jika Status SPM sudah Closing pada Modul Pelaporan: sistem akan menampilkan peringatan bahwa SPM sudah Closing Jika pada konfirmasi penghapusan menekan tombol Kembali: sistem akan me-redirect ke halaman R/U/H Nomor SP2D tanpa menyimpan nomor SP2D pada database</p>

A.18. Deskripsi Use Case Kirim SPM

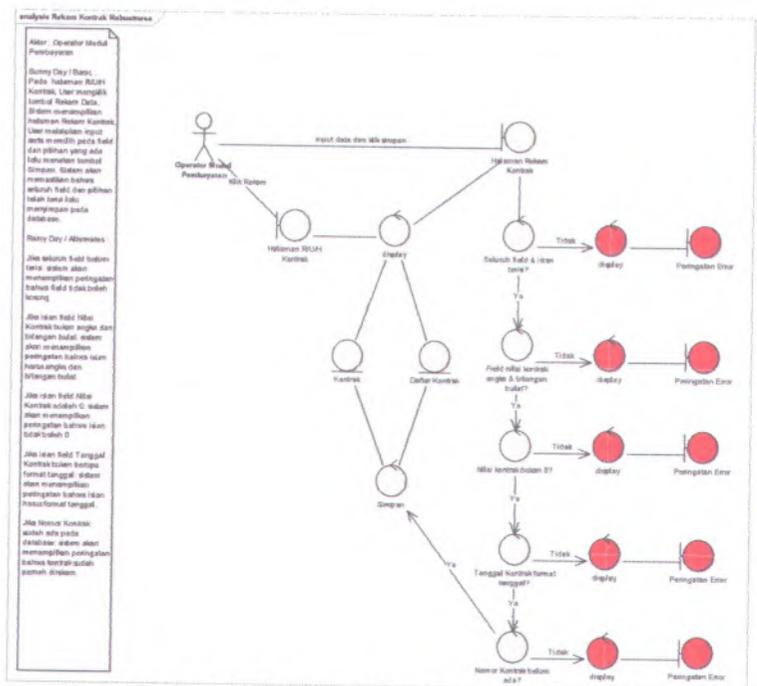
Tabel A.17. Deskripsi Use Case Kirim SPM

<i>UC17 – Kirim SPM</i>	
Primary Actor:	Level:

Operator Modul Pembayaran	User Goal
Pre-conditions: Operator Modul Pembayaran berada pada halaman Kirim SPM.	
Triggers: <ul style="list-style-type: none"> • Operator Modul Pembayaran memilih data SPM yang akan dikirim lalu menekan tombol Kirim. 	
Basic course: Sistem kemudian menampilkan konfirmasi pengiriman SPM. Operator Modul Pembayaran memilih Kirim.	
Post-conditions: Sistem akan menghasilkan file kirim SPP dari database.	
Alternate courses: Jika pada konfirmasi pengiriman menekan tombol Batal: sistem akan me-redirect ke halaman Kirim SPM tanpa menghasilkan file kirim SPP.	

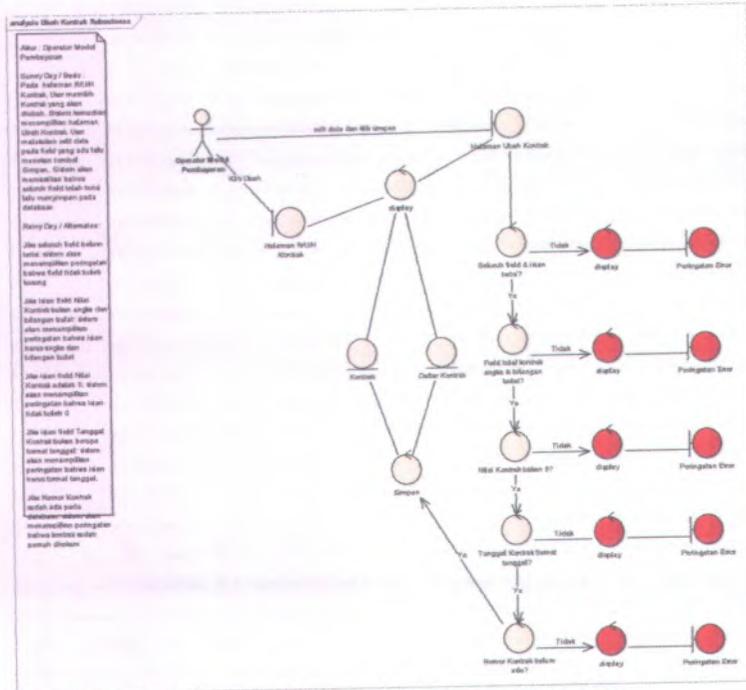
LAMPIRAN B
DIAGRAM ROBUSTNESS ANALYSIS

B.1. Diagram Rekam Kontrak Robustness



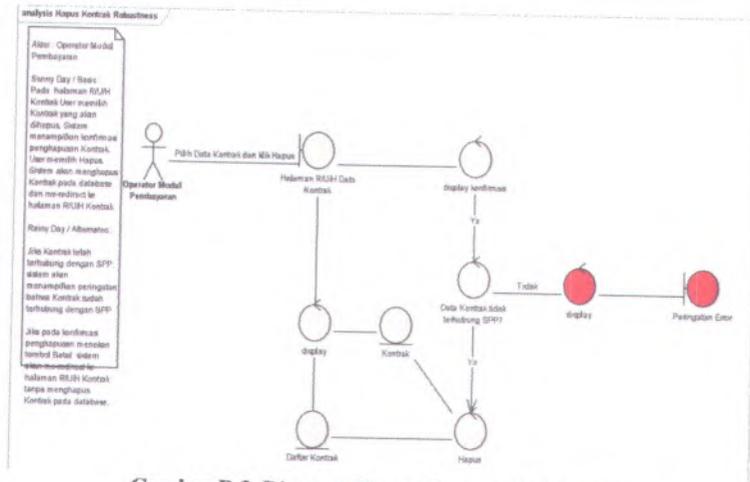
Gambar B.1. Diagram Rekam Kontrak Robustness

B.2. Diagram Ubah Kontrak Robustness



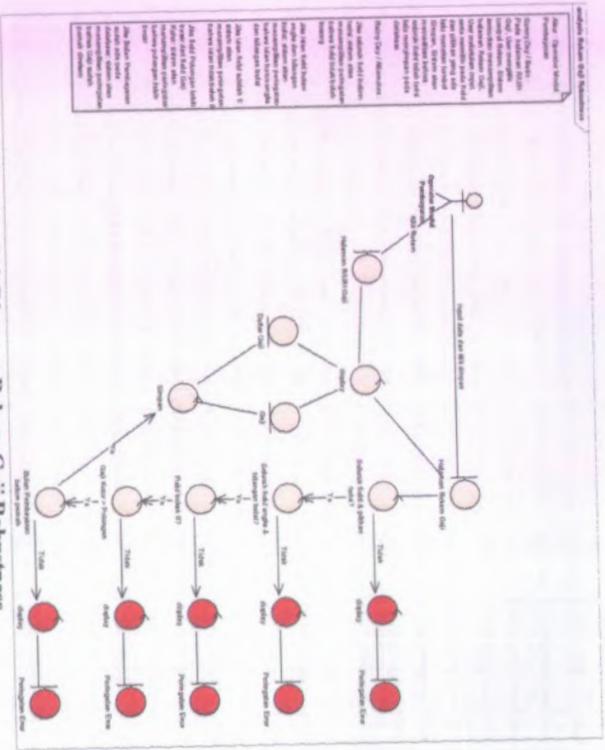
Gambar B.2. Diagram Ubah Kontrak Robustness

B.3. Diagram Hapus Kontrak Robustness



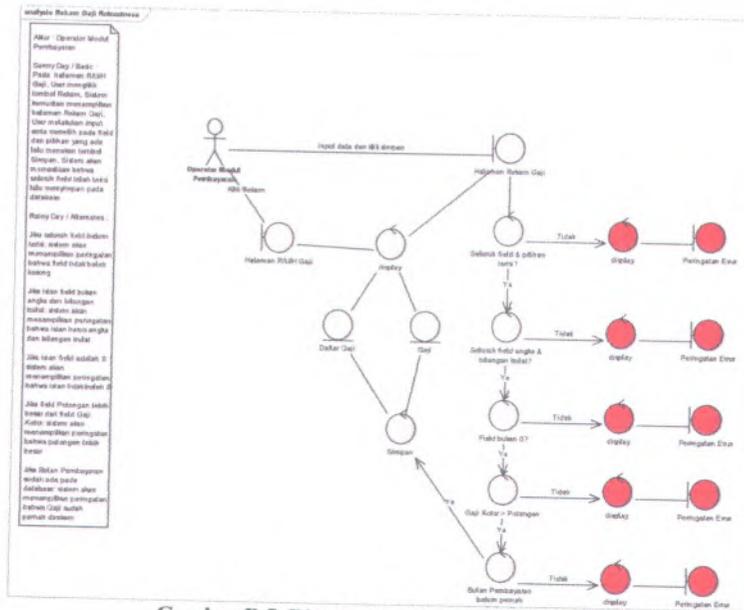
Gambar B.3. Diagram Hapus Kontrak Robustness

B.4. Diagram Rekam Gaji Robustness



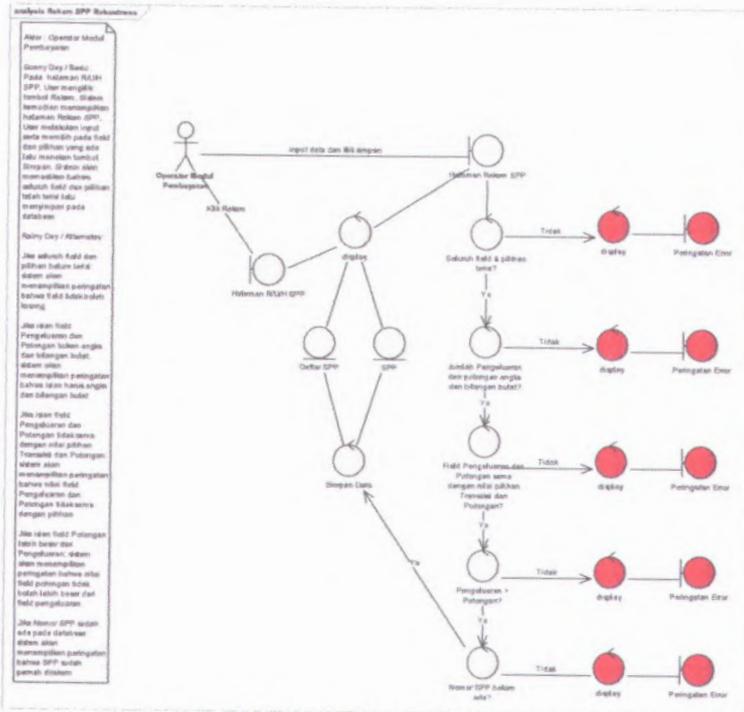
Gambar B.4. Diagram Rekam Gaji Robustness

B.5. Diagram Ubah Gaji Robustness



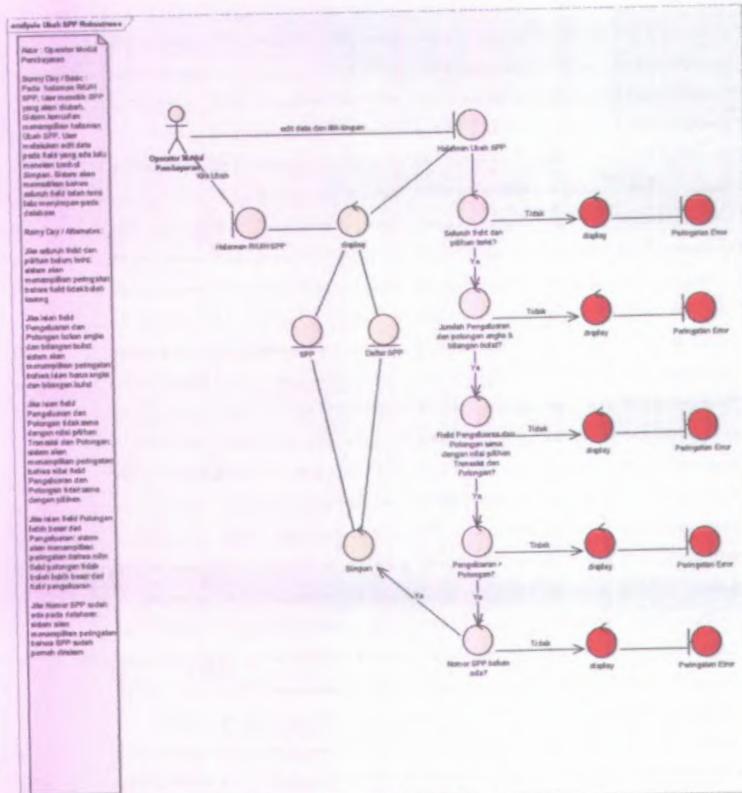
Gambar B.5. Diagram Ubah Gaji Robustness

B.7. Diagram Rekam SPP Robustness



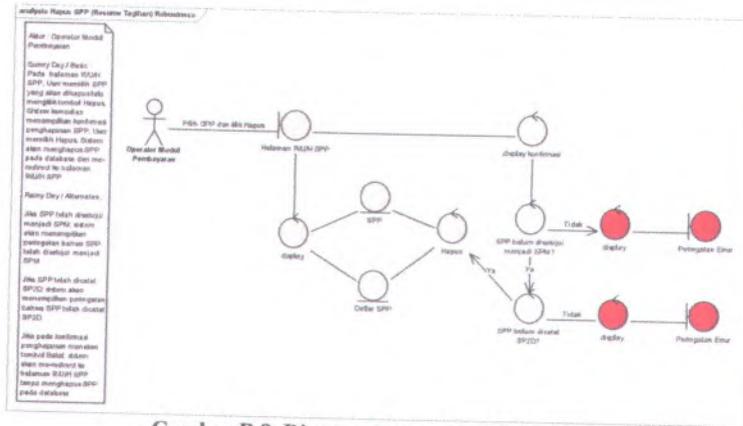
Gambar B.7. Diagram Rekam SPP Robustness

B.8. Diagram Ubah SPP Robustness



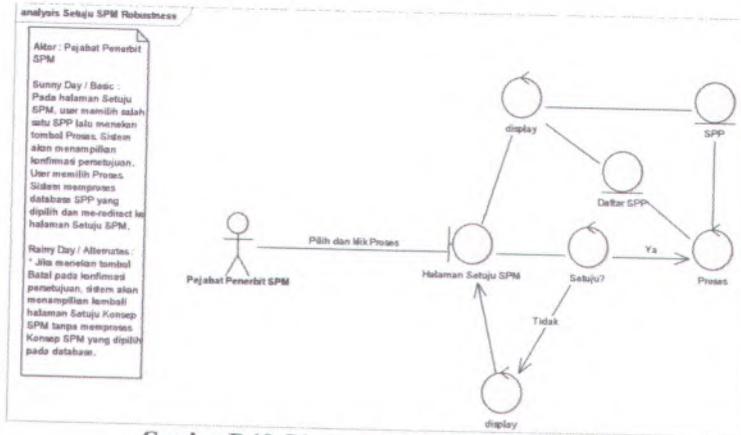
Gambar B.8. Diagram Ubah SPP Robustness

B.9. Diagram Hapus SPP Robustness



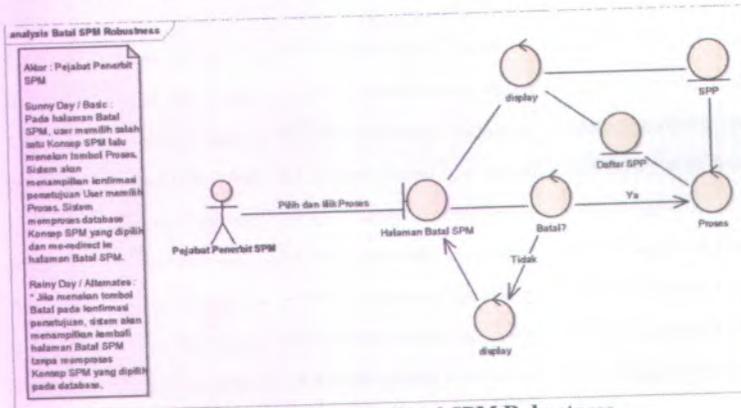
Gambar B.9. Diagram Hapus SPP Robustness

B.10. Diagram Setuju SPM Robustness



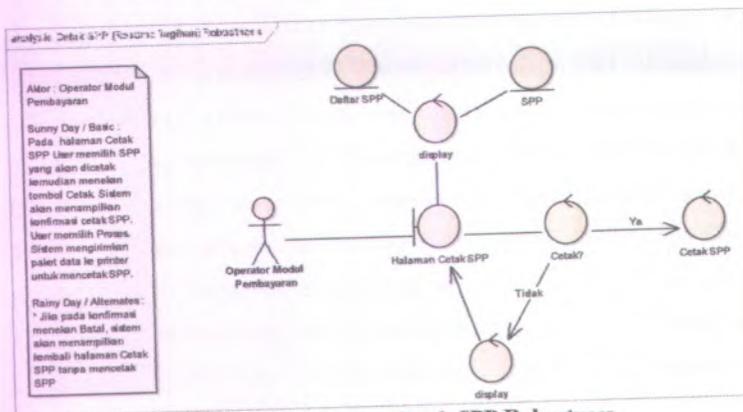
Gambar B.10. Diagram Setuju SPM Robustness

B.11. Diagram Batal SPM Robustness



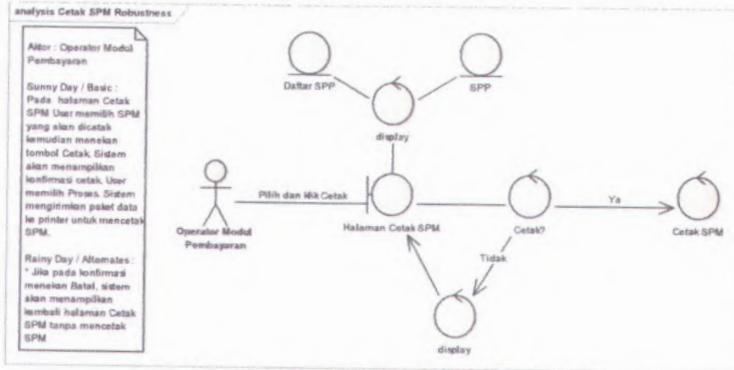
Gambar B.11. Diagram Batal SPM Robustness

B.12. Diagram Cetak SPP Robustness



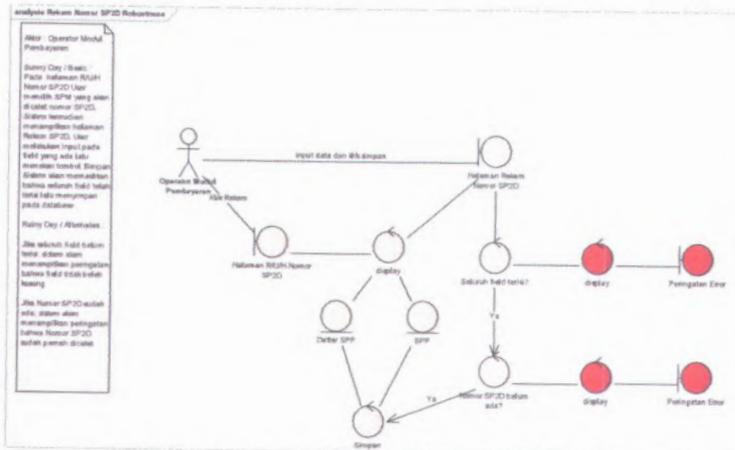
Gambar B.12. Diagram Cetak SPP Robustness

B.13. Diagram Cetak SPM Robustness



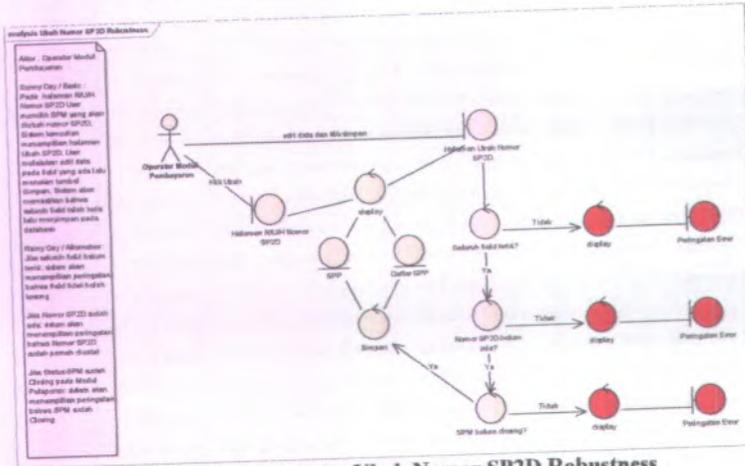
Gambar B.13. Diagram Cetak SPM Robustness

B.14. Diagram Rekam Nomor SP2D Robustness



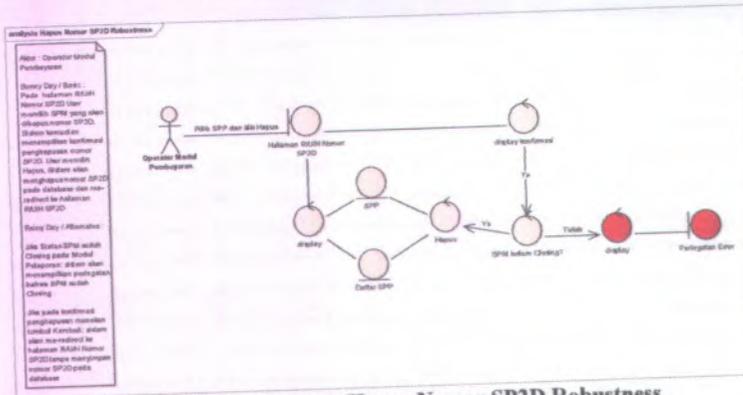
Gambar B.14. Diagram Rekam Nomor SP2D Robustness

B.15. Diagram Ubah Nomor SP2D Robustness



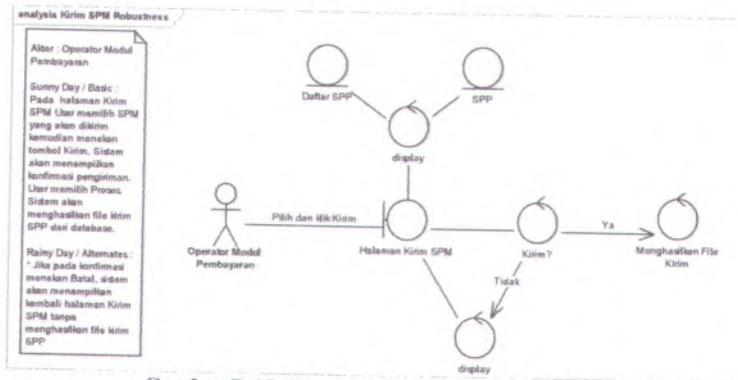
Gambar B.15. Diagram Ubah Nomor SP2D Robustness

B.16. Diagram Hapus Nomor SP2D Robustness



Gambar B.16. Diagram Hapus Nomor SP2D Robustness

B.17. Diagram Kirim SPM Robustness



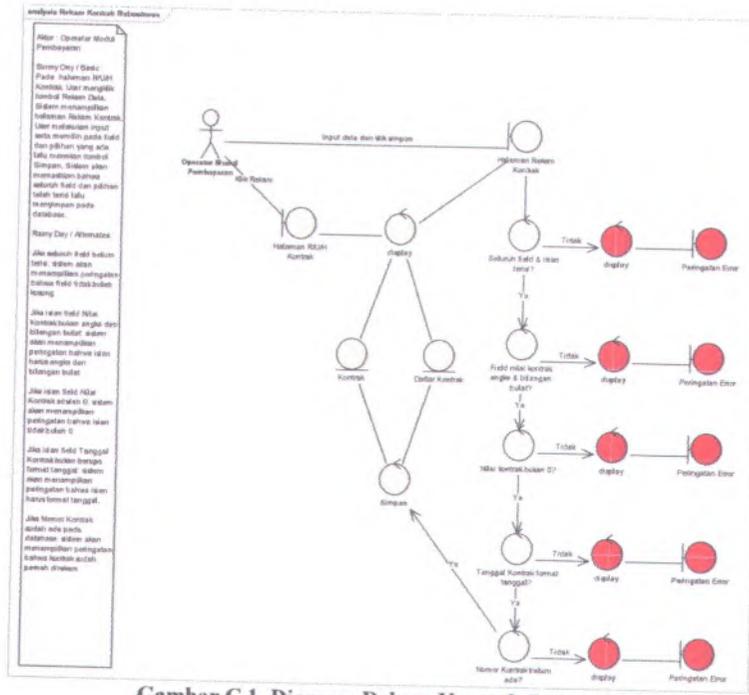
Gambar B.17. Diagram Kirim SPM Robustness

LAMPIRAN C
SEQUENCE DIAGRAM

D-1

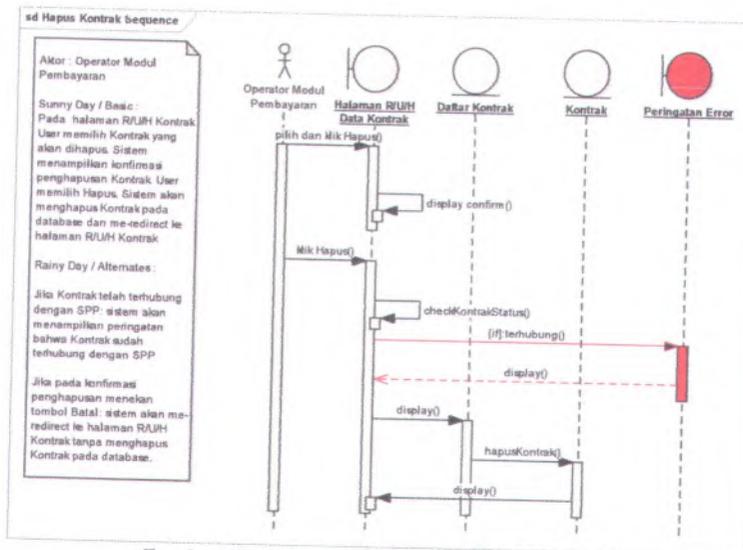


C.1. Diagram Rekam Kontrak Sequence



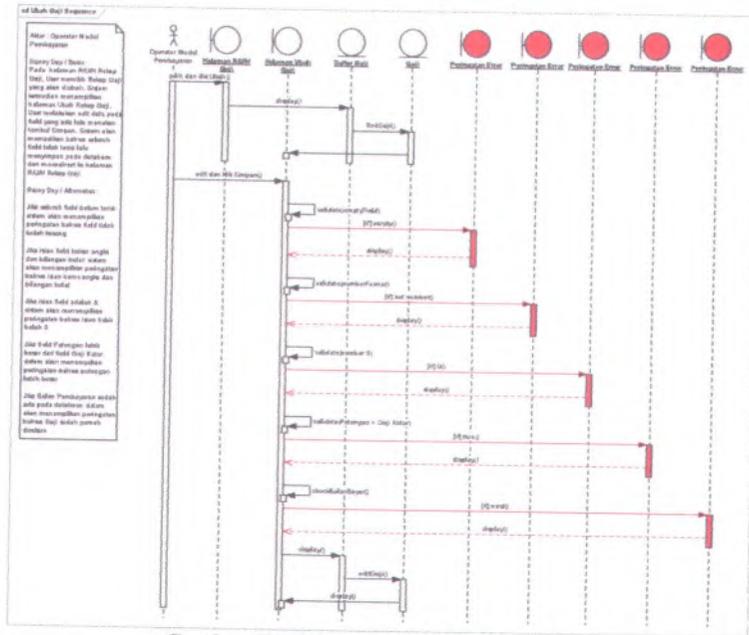
Gambar C.1. Diagram Rekam Kontrak Sequence

C.3. Diagram Hapus Kontrak Sequence



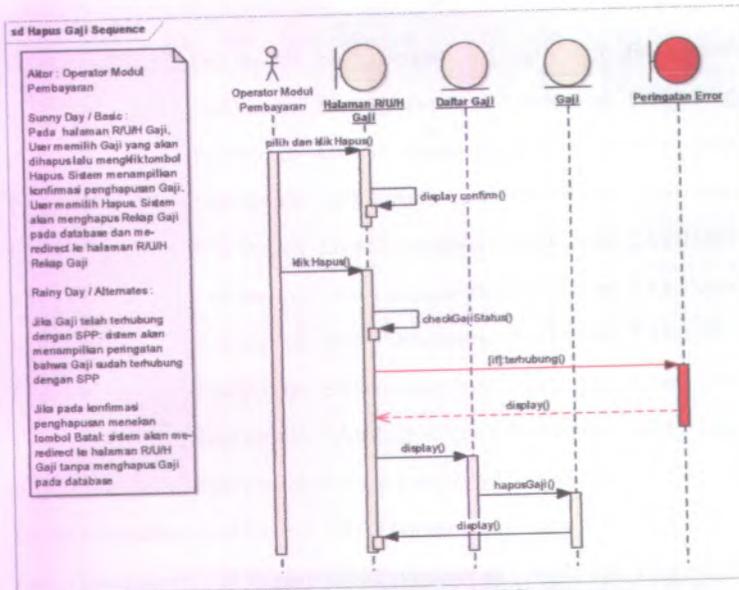
Gambar C.3. Diagram Hapus Kontrak Sequence

C.5. Diagram Ubah Gaji Sequence



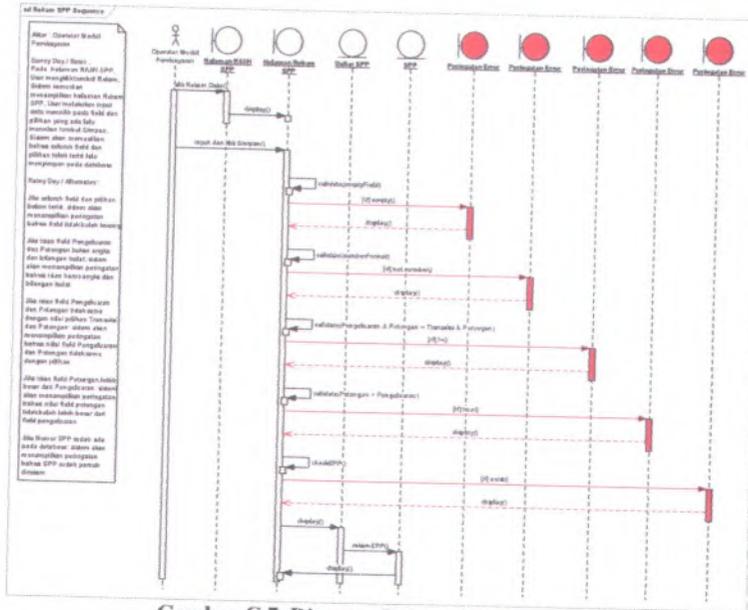
Gambar C.5. Diagram Ubah Gaji Sequence

C.6. Diagram Hapus Gaji Sequence



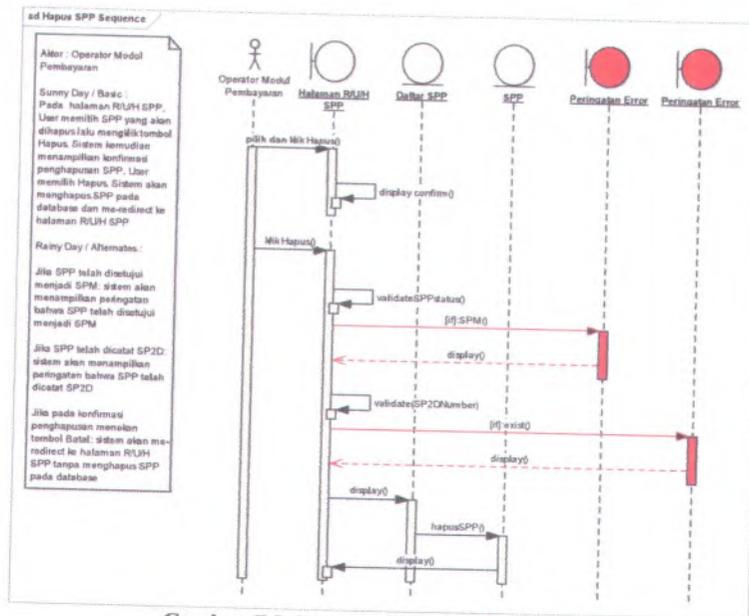
Gambar C.6. Diagram Hapus Gaji Sequence

C.7. Diagram Rekam SPP Sequence



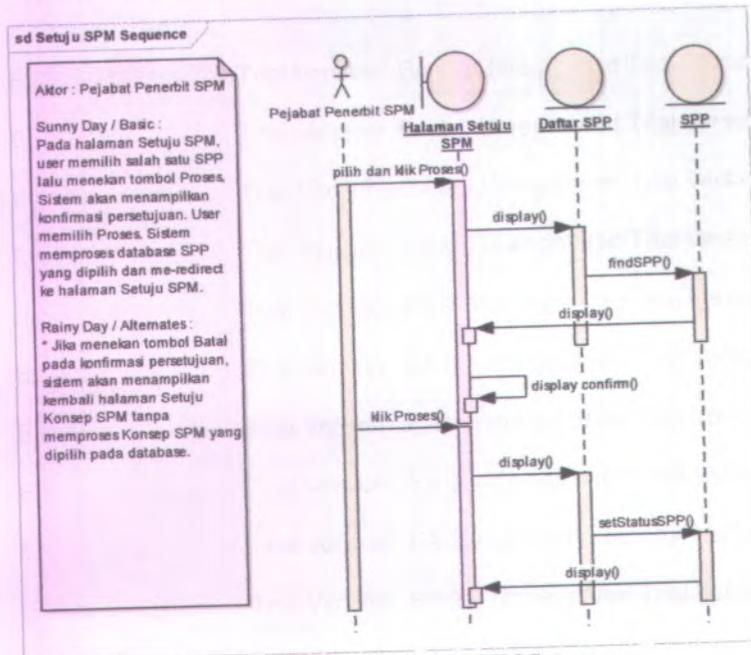
Gambar C.7. Diagram Rekam SPP Sequence

C.9. Diagram Hapus SPP Sequence



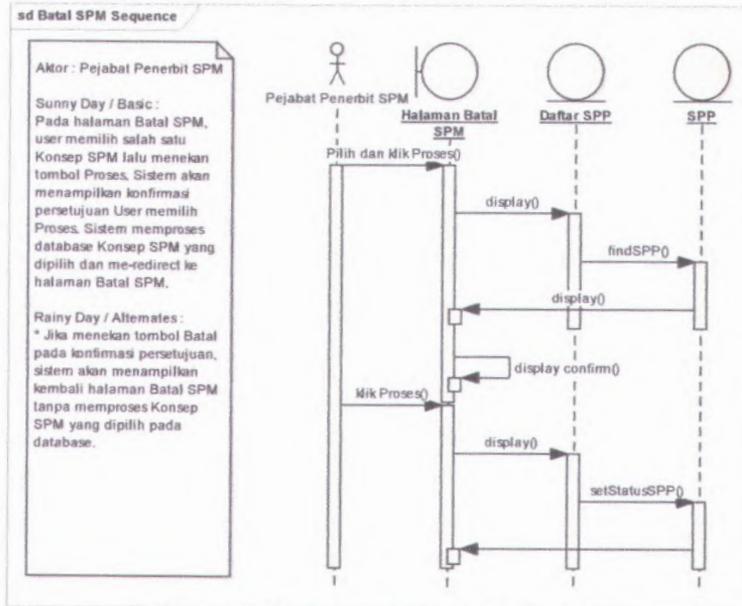
Gambar C.9. Diagram Hapus SPP Sequence

C.10. Diagram Setuju SPM Sequence



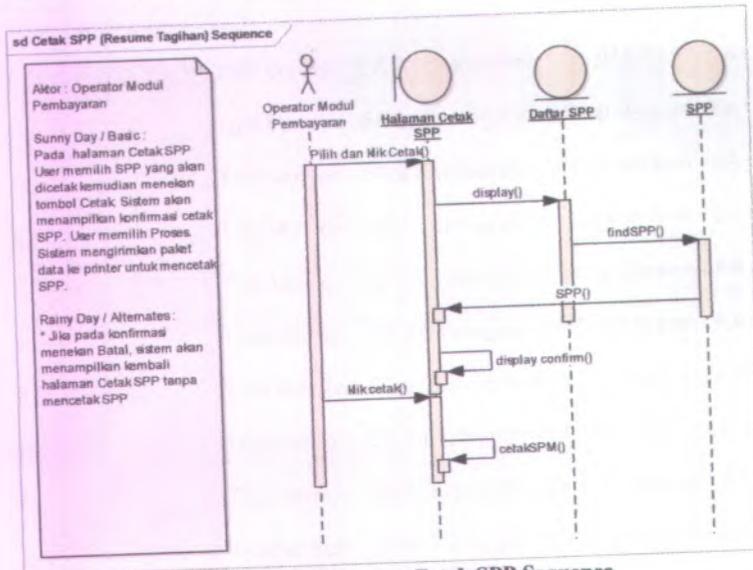
Gambar C.10. Diagram Setuju SPM Sequence

C.11. Diagram Batal SPM Sequence



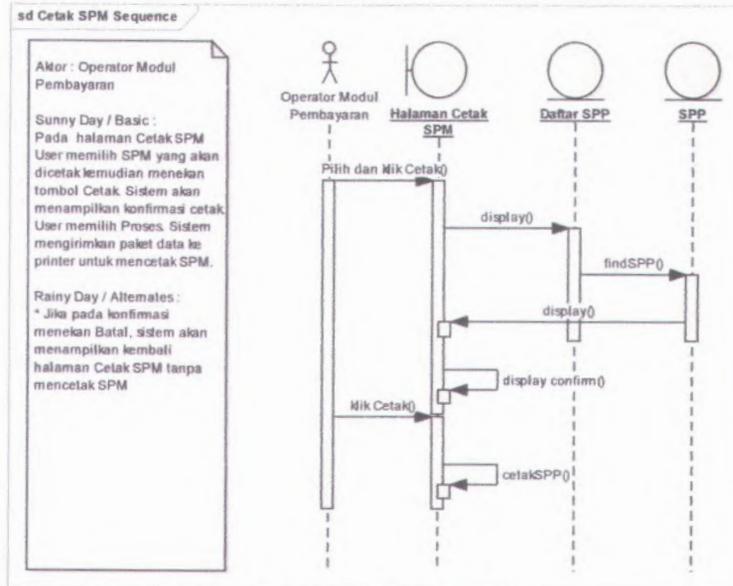
Gambar C.11. Diagram Batal SPM Sequence

C.12. Diagram Cetak SPP Robustness



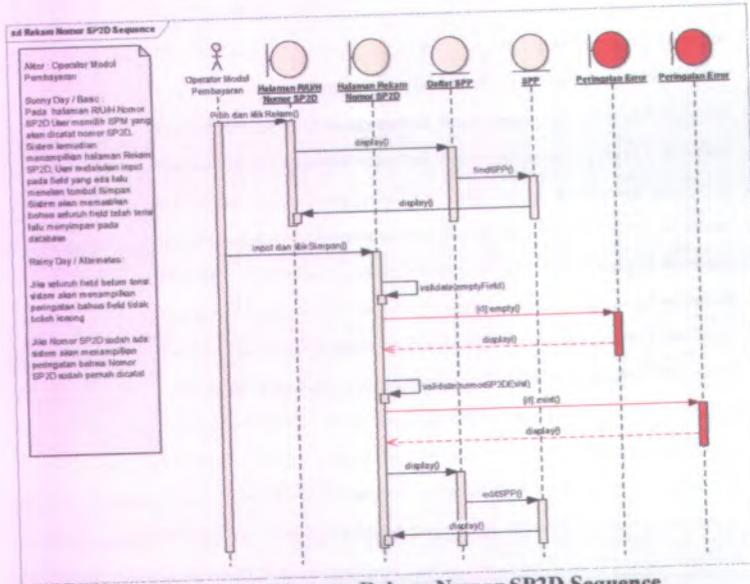
Gambar C.12. Diagram Cetak SPP Sequence

C.13. Diagram Cetak SPM Sequence

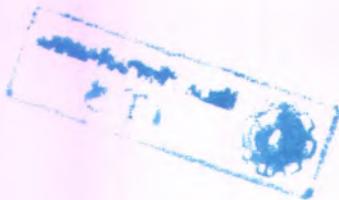


Gambar C.13. Diagram Cetak SPM Sequence

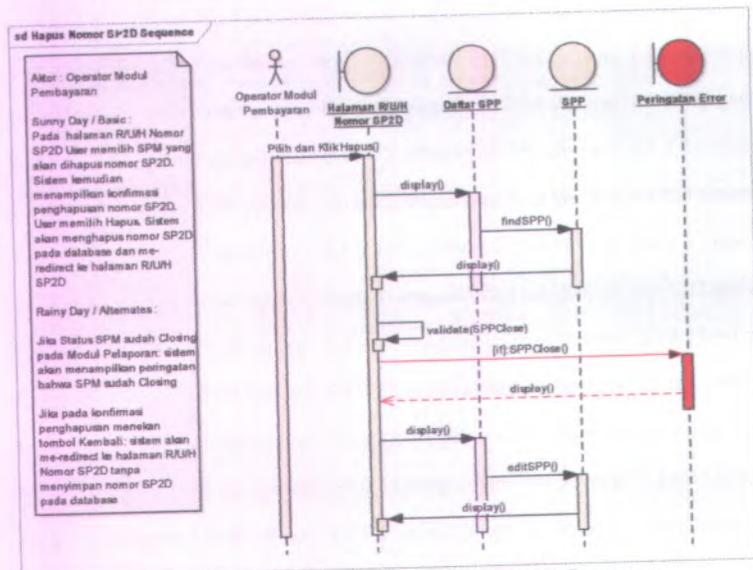
C.14. Diagram Rekam Nomor SP2D Sequence



Gambar C.14. Diagram Rekam Nomor SP2D Sequence

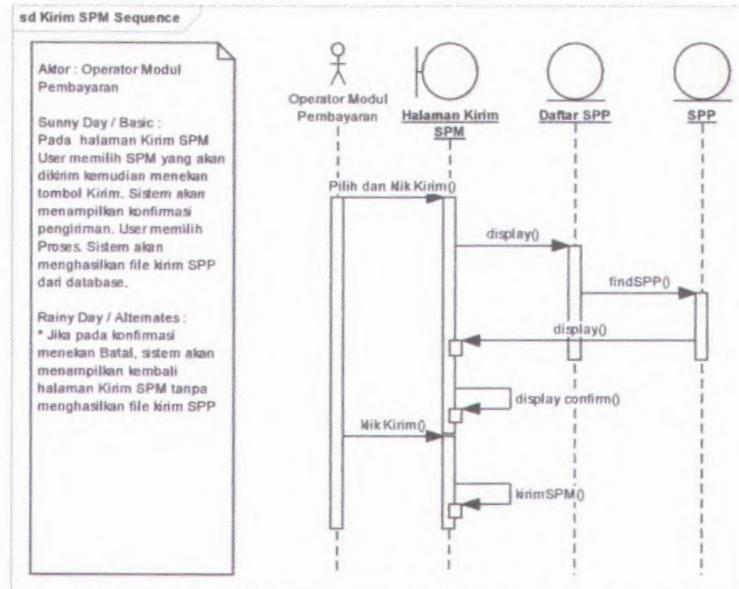


C.16. Diagram Hapus Nomor SP2D Sequence



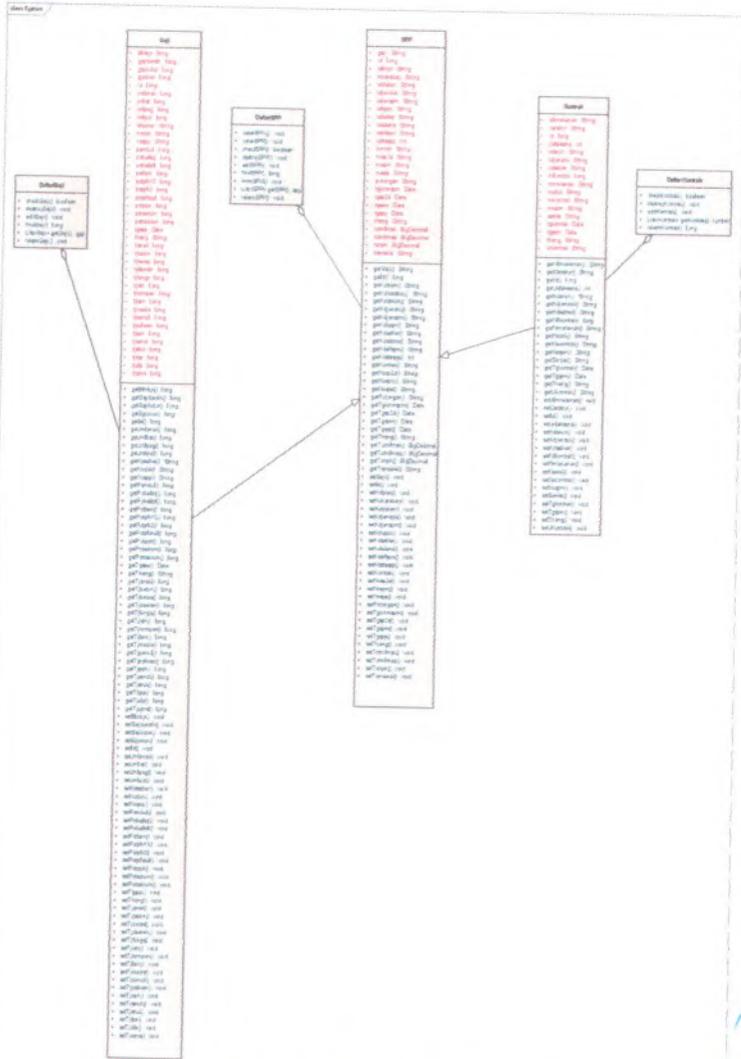
Gambar C.16. Diagram Hapus Nomor SP2D Sequence

C.17. Diagram Kirim SPM Sequence



Gambar C.17. Diagram Kirim SPM Robustness

LAMPIRAN D
CLASS DIAGRAM



Gambar D.1. Class Diagram

LAMPIRAN ETEST CASE

E.1. Test Case Rekam Kontrak

Tabel D.1. Test Case Rekam Kontrak

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	User mengisi seluruh field isian dengan benar	Menekan Tombol Simpan	Data Kontrak tersimpan pada database	Sukses
TC02	Salah satu field tidak terisi	Menekan Tombol Simpan	Muncul pesan error	Sukses
TC03	Mengisi field Nilai Kontrak dengan isian bukan angka	Menekan Tombol Simpan	Muncul pesan error	Sukses
TC04	Mengisi field Nilai Kontrak dengan isian angka 0	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC05	Mengisi field Tanggal Kontrak bukan dengan format tanggal	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC06	Mengisi field Nomor Kontrak dengan isian yang sebelumnya pernah direkam	Menekan Tombol Simpan	Muncul Pesan error	Sukses

E.2. Test Case Ubah Kontrak

Tabel D.2. Test Case Ubah Kontrak

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	User mengisi seluruh field isian dengan benar	Menekan Tombol Simpan	Data Kontrak tersimpan pada database	Sukses
TC02	Salah satu field tidak terisi	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC03	Mengisi field Nilai Kontrak dengan isian bukan angka	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC04	Mengisi field Nilai Kontrak dengan isian angka 0	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC05	Mengisi field Tanggal Kontrak bukan dengan format tanggal	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC06	Mengisi field Nomor Kontrak dengan isian yang sebelumnya pernah direkam	Menekan Tombol Simpan	Muncul Pesan error	Sukses

E.3. Test Case Hapus Kontrak

Tabel D.3. Test Case Hapus Kontrak

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	Pada halaman RUH Kontrak, User memilih data lalu menekan tombol Hapus	Menekan Tombol Hapus	Sistem akan menampilkan konfirmasi penghapusan	Sukses
TC02	User menekan tombol Yes pada konfirmasi penghapusan	Menekan Tombol Yes pada Konfirmasi	Data kontrak terhapus dari database	Sukses
TC03	User menekan tombol Yes pada konfirmasi penghapusan, tetapi Kontrak terhubung dengan SPP	Menekan Tombol Yes pada Konfirmasi	Muncul Pesan error	Sukses
TC04	User menekan tombol No pada konfirmasi penghapusan	Menekan Tombol No pada Konfirmasi	Kembali ke halaman RUH Kontrak	Sukses

E.4. Test Case Rekam Gaji

Tabel D.4. Test Case Rekam Gaji

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	User mengisi seluruh field isian dengan benar	Menekan Tombol Simpan	Data gaji tersimpan pada database	Sukses
TC02	Salah satu field tidak terisi	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC03	Mengisi salah satu field diisi dengan isian bukan angka	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC04	Mengisi field Gaji Pokok dengan isian angka 0	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC05	Seluruh field telah terisi, namun jumlah total Gaji Kotor kurang dari total Potongan	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC06	Memilih Bulan Pembayaran dengan pilihan yang sudah pernah direkam	Menekan Tombol Simpan	Muncul Pesan error	Sukses

E.5. Test Case Ubah Gaji

Tabel D.5. Test Case Ubah Gaji

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	User mengisi seluruh field isian dengan benar	Menekan Tombol Simpan	Data gaji tersimpan pada database	Sukses
TC02	Salah satu field tidak terisi	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC03	Mengisi salah satu field diisi dengan isian bukan angka	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC04	Mengisi field Gaji Pokok dengan isian angka 0	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC05	Seluruh field telah terisi, namun jumlah total Gaji Kotor kurang dari total Potongan	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC06	Memilih Bulan Pembayaran dengan pilihan yang sudah pernah direkam	Menekan Tombol Simpan	Muncul Pesan error	Sukses

E.6. Test Case Hapus Gaji

Tabel D.6. Test Case Hapus Gaji

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	Pada halaman RUH Gaji, User memilih data lalu menekan tombol Hapus	Menekan Tombol Hapus	Sistem akan menampilkan konfirmasi penghapusan	Sukses
TC02	User menekan tombol Yes pada konfirmasi penghapusan	Menekan Tombol Yes pada Konfirmasi	Gaji terhapus dari database	Sukses
TC03	User menekan tombol Yes pada konfirmasi penghapusan, tetapi Gaji terhubung dengan SPP	Menekan Tombol Yes pada Konfirmasi	Muncul Pesan error	Sukses
TC04	User menekan tombol No pada konfirmasi penghapusan	Menekan Tombol No pada Konfirmasi	Kembali ke halaman RUH Gaji	Sukses

E.7. Test Case Rekam SPP

Tabel D.7. Test Case Rekam SPP

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	User mengisi seluruh field isian dengan benar	Menekan Tombol Simpan	Data SPP tersimpan pada database	Sukses
TC02	Salah satu field tidak terisi	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC03	Mengisi field Jumlah Pengeluaran dan Potongan bukan dengan angka	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC04	Mengisi field Jumlah Pengeluaran dan Potongan tidak sama dengan pilihan Transaksi dan Potongan	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC05	Seluruh field telah terisi, namun jumlah total Pengeluaran kurang dari total Potongan	Menekan Tombol Simpan	Muncul Pesan error	Sukses

E-10

TC06	Mengisi field Nomor SPP dengan isian yang sebelumnya pernah direkam	Menekan Tombol Simpan	Muncul Pesan error	Sukses
------	---	-----------------------	--------------------	--------

E.8. Test Case Ubah SPP

Tabel D.8. Test Case Ubah SPP

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	User mengisi seluruh field isian dengan benar	Menekan Tombol Simpan	Data SPP tersimpan pada database	Sukses
TC02	Salah satu field tidak terisi	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC03	Mengisi field Jumlah Pengeluaran dan Potongan bukan dengan angka	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC04	Mengisi field Jumlah Pengeluaran dan Potongan tidak sama dengan pilihan Transaksi dan Potongan	Menekan Tombol Simpan	Muncul Pesan error	Sukses

TC05	Seluruh field telah terisi, namun jumlah total Pengeluaran kurang dari total Potongan	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC06	Mengisi field Nomor SPP dengan isian yang sebelumnya pernah direkam	Menekan Tombol Simpan	Muncul Pesan error	Sukses

E.9. Test Case Hapus SPP

Tabel D.9. Test Case Hapus SPP

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	Pada halaman RUH SPP, User memilih data lalu menekan tombol Hapus	Menekan Tombol Hapus	Sistem akan menampilkan konfirmasi penghapusan	Sukses
TC02	User menekan tombol Yes pada konfirmasi penghapusan	Menekan Tombol Yes pada Konfirmasi	Data SPP terhapus dari database	Sukses
TC03	User menekan tombol	Menekan	Muncul Pesan error	Sukses

E-12

	Yes pada konfirmasi penghapusan, tetapi SPP sudah disetujui menjadi SPM	Tombol Yes pada Konfirmasi		
TC05	User menekan tombol Yes pada konfirmasi penghapusan, tetapi SPP sudah dicatat nomor SP2D	Menekan Tombol Yes pada Konfirmasi	Muncul Pesan error	Sukses
TC06	User menekan tombol No pada konfirmasi penghapusan	Menekan Tombol No pada Konfirmasi	Kembali ke halaman RUH SPP	Sukses

E.10. Test Case Setuju SPM

Tabel D.10. Test Case Setuju SPM

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	Pada halaman Setuju SPM, User memilih data lalu menekan tombol Proses	Menekan Tombol Proses	Sistem akan menampilkan konfirmasi persetujuan	Sukses
TC02	User menekan tombol	Menekan	SPM tersimpan pada	Sukses

	Yes pada konfirmasi persetujuan	Tombol Yes pada Konfirmasi	database	
TC04	User menekan tombol No pada konfirmasi persetujuan	Menekan Tombol No pada Konfirmasi	Kembali ke halaman Setuju SPM	Sukses

E.11. Test Case Batal SPM

Tabel D.11. Test Case Batal SPM

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	Pada halaman Batal SPM, User memilih data lalu menekan tombol Proses	Menekan Tombol Proses	Sistem akan menampilkan konfirmasi pembatalan	Sukses
TC02	User menekan tombol Yes pada konfirmasi pembatalan	Menekan Tombol Yes pada Konfirmasi	SPM dibatalkan dari database	Sukses
TC03	User menekan tombol No pada konfirmasi pembatalan	Menekan Tombol No pada Konfirmasi	Kembali ke halaman Batal SPM	Sukses

E.12. Test Case Cetak SPP

Tabel D.12. Test Case Cetak SPP

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	Pada halaman Cetak SPP, User memilih data lalu menekan tombol Cetak	Menekan Tombol Cetak	Sistem akan menampilkan konfirmasi persetujuan	Sukses
TC02	User menekan tombol Yes pada konfirmasi persetujuan	Menekan Tombol Yes pada Konfirmasi	Menampilkan data SPP untuk dicetak	Sukses
TC04	User menekan tombol No pada konfirmasi persetujuan	Menekan Tombol No pada Konfirmasi	Kembali ke halaman Cetak SPP	Sukses

E.13. Test Case Cetak SPM

Tabel D.13. Test Case Cetak SPM

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	Pada halaman Cetak SPM, User memilih data lalu menekan	Menekan Tombol Cetak	Sistem akan menampilkan konfirmasi persetujuan	Sukses

	tombol Cetak			
TC02	User menekan tombol Yes pada konfirmasi persetujuan	Menekan Tombol Yes pada Konfirmasi	Menampilkan data SPP untuk dicetak	Sukses
TC04	User menekan tombol No pada konfirmasi persetujuan	Menekan Tombol No pada Konfirmasi	Kembali ke halaman Cetak SPM	Sukses

E.14. Test Case Rekam Nomor SP2D

Tabel D.14. Test Case Rekam Nomor SP2D

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	User mengisi seluruh field isian dengan benar	Menekan Tombol Simpan	Data SP2D tersimpan pada database	Sukses
TC02	Salah satu field tidak terisi	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC03	Mengisi field Nomor SP2D dengan isian yang sebelumnya pernah direkam	Menekan Tombol Simpan	Muncul Pesan error	Sukses

E-16

E.15. Test Case Ubah Nomor SP2D

Tabel D.15. Test Case Ubah Nomor SP2D

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	User mengisi seluruh field isian dengan benar	Menekan Tombol Simpan	Data SP2D tersimpan pada database	Sukses
TC02	Salah satu field tidak terisi	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC03	Mengisi field Nomor SP2D dengan isian yang sebelumnya pernah direkam	Menekan Tombol Simpan	Muncul Pesan error	Sukses
TC04	Seluruh field isian benar tetapi SPM sudah closing	Menekan Tombol Simpan	Muncul Pesan error	Sukses

E.16. Test Case Hapus Nomor SP2D

Tabel D.16. Test Case Hapus Nomor SP2D

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	Pada halaman RUH SP2D, User memilih	Menekan Tombol Hapus	Sistem akan menampilkan konfirmasi penghapusan	Sukses

	data lalu menekan tombol Hapus			
TC02	User menekan tombol Yes pada konfirmasi penghapusan	Menekan Tombol Yes pada Konfirmasi	Data SP2D terhapus dari database	Sukses
TC03	User menekan tombol Yes pada konfirmasi penghapusan, tetapi SPM sudah closing	Menekan Tombol Yes pada Konfirmasi	Muncul Pesan error	Sukses
TC05	User menekan tombol No pada konfirmasi penghapusan	Menekan Tombol No pada Konfirmasi	Kembali ke halaman RUH SP2D	

E.17. Test Case Kirim SPM

Tabel D.17. Test Case Kirim SPM

ID	Skenario	Aksi	Hasil	Sukses/Gagal
TC01	Pada halaman Kirim SPM, User memilih data lalu menekan tombol Kirim	Menekan Tombol Kirim	Sistem akan menampilkan konfirmasi pengiriman	Sukses
TC02	User menekan tombol	Menekan	Menghasilkan file kirim	Sukses

E-18

	Yes pada konfirmasi pengiriman	Tombol Yes pada Konfirmasi	SPM	
--	--------------------------------	----------------------------	-----	--

BIODATA PENULIS



Penulis dilahirkan di Jakarta, 27 November 1984, merupakan anak pertama dari lima bersaudara. Penulis telah menempuh pendidikan formal di SD Hang Tuah 1 Jakarta, lalu melanjutkan di SMP N 95 Jakarta, dilanjutkan di SMUN 13 Jakarta dan Sekolah Tinggi

Akuntansi Negara. Pada tahun 2009, penulis mengajukan diri untuk mengikuti Program Beasiswa Internal Direktorat Jenderal Perbendaharaan, Departemen Keuangan, dan diterima di Jurusan Sistem Informasi pada tahun 2009 dengan NRP 5209108709.

Selama melaksanakan tugas belajar di Jurusan Sistem Informasi, penulis juga aktif membantu instansi bernaung dalam kegiatan pengembangan website untuk kantor-kantor daerah, dan juga aktif dalam menekuni hobi travelling. Untuk pengerjaan Tugas Akhir, penulis mengambil bidang minat e-Bisnis dengan topik tentang Integrasi Aplikasi bagi instansi Direktorat Jenderal Perbendaharaan.