

**SISTEM ONLINE UNTUK DOKUMEN PERANCANGAN  
PERANGKAT LUNAK BAGI PERUSAHAAN ATAU  
TIM PENGEMBANG**



Disusun Oleh:

N a m a : Muhammad Luthfi Fadholi  
NIM : 14523033

**PROGRAM STUDI TEKNIK INFORMATIKA – PROGRAM SARJANA  
FAKULTAS TEKNOLOGI INDUSTRI  
UNIVERSITAS ISLAM INDONESIA  
2018**

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**SISTEM ONLINE UNTUK DOKUMEN PERANCANGAN  
PERANGKAT LUNAK BAGI PERUSAHAAN ATAU  
TIM PENGEMBANG**

**TUGAS AKHIR**



الجامعة الإسلامية  
Yogyakarta, 18 September 2018  
Pembimbing,

( Beni Suranto, S.T. M.Soft.Eng. )

**HALAMAN PENGESAHAN DOSEN PENGUJI**

**SISTEM ONLINE UNTUK DOKUMEN PERANCANGAN  
PERANGKAT LUNAK BAGI PERUSAHAAN ATAU  
TIM PENGEMBANG  
TUGAS AKHIR**

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk  
memperoleh gelar Sarjana Komputer dari Program Studi Teknik Informatika  
di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 18 Oktober 2018

Tim Penguji

Beni Suranto, S.T., M.Soft.Eng.

**Anggota 1**

Andhik Budi Cahyono, S.T., M.T.

**Anggota 2**

Erika Ramadhani, S.T., M.Eng.

Mengetahui,

Ketua Program Studi Teknik Informatika – Program Sarjana  
Fakultas Teknologi Industri  
Universitas Islam Indonesia



( Dr. Raden Teduh Dirgahayu, S.T., M.Sc. )

## HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Muhammad Luthfi Fadholi

NIM : 14523033

Tugas akhir dengan judul:

### **SISTEM ONLINE UNTUK DOKUMEN PERANCANGAN PERANGKAT LUNAK BAGI PERUSAHAAN ATAU TIM PENGEMBANG**

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila dikemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung resiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 4 Oktober 2018



( Muhammad Luthfi Fadholi )

## HALAMAN PERSEMBAHAN

Segala puji bagi Allah SWT, yang telah melimpahkan ridho, rahmat, nikmat, hidayah, serta kesempatan kepada saya untuk menuntut ilmu, menambah pengalaman dan pengetahuan sehingga penulis tiada henti-hentinya mengucap syukur atas apa yang telah Allah berikan. Tidak lupa penulis memohon ampun atas kesalahan yang telah dilakukan baik disadari maupun tidak. Shalawat serta salam semoga tercurah kepada Nabi Muhammad SAW, yang telah memimpin umat manusia dari zaman kegelapan menuju zaman terang benderang. Dengan mengucap hamdallah, tugas akhir ini penulis persembahkan kepada:

1. Kedua orang tua saya, Bapak Supriyono dan Ibu Eni Rahayuningsih yang telah sabar mendidik saya, selalu memberikan dukungan moral dan selalu mendo'akan yang terbaik untuk saya.
2. Keluarga besar, serta teman-teman yang selalu berbagi suka maupun duka, semoga kita bisa tetap menjaga tali silaturahmi.
3. Dosen pembimbing, Bapak Beni Suranto yang telah meluangkan waktu, serta tenaga, dan selalu sabar dalam membimbing saya selama mengerjakan Tugas Akhir ini.

Penulis menyadari bahwa ini saja tidak akan cukup untuk membalas semua kebaikan mereka semua. Akan tetapi, semoga prestasi kecil ini bisa membuat bangga semua pihak terutama kedua orang tua tercinta. Aamiin.

## HALAMAN MOTO

*“Tidak ada kata terlambat untuk belajar, asal ada kemauan pasti bisa”*

**(M. Luthfi Fadholi)**

*Man Jadda Wa Jadda*

***“Barangsiapa yang bersungguh-sungguh pasti akan mendapatkan hasil”***

*Innama'al 'Usri Yusroo*

***“Sesungguhnya sesudah kesulitan pasti ada kemudahan”***

## KATA PENGANTAR

### *Assalamu'alaykum Warahmatullahi Wabarakatuh*

Alhamdulillah penulis haturkan kepada Allah SWT Tuhan semesta alam, yang telah melimpahkan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan laporan Tugas Akhir yang berjudul “Sistem Online Untuk Dokumen Perancangan Perangkat Lunak Bagi Perusahaan atau Tim Pengembang” dengan baik. Sholawat serta salam kita haturkan kepada junjungan besar kita Nabi Muhammad SAW karena berkat perjuangan beliau kita dapat mengecap indahny ilmu pengetahuan.

Laporan Tugas Akhir ini disusun sebagai salah satu syarat untuk memperoleh gelar sarjana pada Jurusan Teknik Informatika di Universitas Islam Indonesia dan sebagai salah satu sarana bagi penulis untuk mempraktekkan secara langsung ilmu yangtelah diperoleh selama masa studi di Jurusan Teknik Informatika Fakultas Teknologi Industri Universitas Islam Indonesia.

Penulis menyadari bahwa dalam pelaksanaan Tugas Akhir dan penyusunan laporan ini tidak bisa lepas dari bimbingan, dorongan, serta bantuan baik material maupun spiritual dari berbagai pihak. Oleh karena itu, pada kesempatan ini dengan kerendahan hati penulis menyampaikan ucapan terima kasih dan penghargaan setinggi-tingginya kepada:

1. Allah SWT atas karunia dan rahmat-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir ini dengan baik.
2. Kedua Orang Tua (Supriyono dan Eni Rahayuningsih) atas segala doa dan dukungan selama penulis menyelesaikan masa studi di Teknik Informatika dan selama penulis menyelesaikan Tugas Akhir.
3. Fathul Wahid, S.T., M.Sc., Ph.D., selaku Rektor Universitas Islam Indonesia.
4. Prof. Dr. Ir. Hari Purnomo, M.T., selaku Dekan Fakultas Teknologi Industri Universitas Islam Indonesia.
5. Hendrik, S.T., M.Eng, selaku Ketua Jurusan Teknik Informatika.
6. Beni Suranto, S.T, M.Soft.Eng., selaku dosen pembimbing tugas akhir yang telah memberikan waktu untuk membantu penulis dalam menyelesaikan tugas akhir
7. Seluruh Dosen Jurusan Teknik Informatika UII yang telah memberikan ilmu yang sangat bermanfaat.
8. Adysta Galang yang sudah mengajari saya *framework Laravel* sehingga saya dapat menyelesaikan Tugas Akhir ini dengan baik.

9. Teman-teman Asisten Praktikum Pemrograman Web dan Asisten Lab Terpadu Informatika yang telah mengajarkan banyak hal dan menjadi keluarga saya selama saya menjadi asisten praktikum.
10. Teman-teman Himpunan Mahasiswa Teknik Informatika (HMTF) yang telah mengajarkan saya banyak hal tentang organisasi dan *self development*.
11. Teman-teman Persatuan Bulutangkis Mahasiswa Tua (PBMT) yang telah mengisi hari-hari saya dengan olahraga sehingga tubuh saya tetap fit selama mengerjakan Tugas Akhir.
12. Teman-teman Komeng Ranger yaitu Agusta, Galih, Fandi, Fauzan, Goudy, Salsa, Irgi, dan Iqbal yang telah menjadi bagian dari pertemanan selama saya kuliah.
13. Teman-teman Teknik Informatika angkatan 2014 (MAGNIFICO) untuk semua bantuan, dukungan, serta kenangan indah selama saya kuliah.
14. Semua pihak yang telah banyak membantu dalam pelaksanaan Tugas Akhir yang tidak dapat penulis sebutkan satu persatu.

Penulis menyadari bahwa laporan Tugas Akhir ini masih belum sempurna. Oleh karena itu penulis mengharapkan kritik dan saran yang membangun demi kesempurnaan Laporan Tugas Akhir ini. Semoga laporan ini dapat bermanfaat bagi semua pihak dan atas segala bantuan yang telah diberikan dari semua pihak semoga mendapat imbalan yang setimpal dari Allah SWT, aamiin.

***Wassalamu'alaykum Warahmatullahi Wabarakatuh***

Yogyakarta, 17 September 2018

( Muhammad Luthfi Fadholi )



## SARI

Perangkat lunak yang baik merupakan perangkat lunak yang memiliki perancangan dan dokumentasi yang baik dalam proses pembuatannya. Dokumentasi perancangan perangkat lunak biasa disebut *Software Design Document (SDD)*. SDD adalah representasi dari sistem perangkat lunak yang digunakan sebagai media untuk mengkomunikasikan informasi desain perangkat lunak. Pembuatan SDD biasanya masih dibuat secara konvensional menggunakan aplikasi pengolah kata seperti Microsoft Word, Open Office, dan lain sebagainya. Sehingga apabila anggota tim lain ingin menambahkan ataupun mengubah isi dari SDD, anggota tim yang terakhir mengerjakan harus mengirim data terbaru agar tidak terjadi kesalahan data. Maka dari itu dibutuhkan sebuah sistem yang dapat membantu tim pengembang untuk berkolaborasi dalam pembuatan SDD.

Penelitian ini dilakukan dalam beberapa tahapan. Tahapannya dimulai dari analisis kebutuhan sistem yang dilakukan dengan mengacu pada standar IEEE 1016-2009. Tahapan perancangan sistem yang meliputi rancangan fungsionalitas sistem, rancangan interaksi antara pengguna dengan sistem, rancangan basisdata di dalam sistem, dan rancangan desain antarmuka dari sistem. Tahap implementasi sistem dengan menggunakan framework Laravel versi 5.5. Tahap yang terakhir adalah pengujian sistem yang dilakukan dengan cara menguji coba sistem yang telah dibuat pada para calon pengguna, kemudian calon pengguna diminta untuk mengisi kuesioner.

Kesimpulan dari penelitian ini adalah sistem yang telah dibuat memiliki manfaat bagi para tim pengembang perangkat lunak untuk membantu kolaborasi dalam pembuatan dan penggunaan SDD. Selain itu, sistem ini juga telah hampir memenuhi seluruh kebutuhan tim pengembang dalam pembuatan SDD dengan perolehan nilai pengujian sebesar 83% , sehingga dapat digolongkan dalam kategori baik.

Kata kunci: *Software Design Document (SDD)*, Kolaborasi, Tim Pengembang Perangkat Lunak.

## GLOSARIUM

SDD	Dokumen perancangan perangkat lunak yang berisi rancangan suatu perangkat lunak mulai dari rancangan fungsionalitas sistem sampai rancangan
Role	peran pengguna dalam sistem yang berguna untuk menentukan hak akses pengguna didalam sistem.
Log	pencatatan aktivitas pengguna didalam sistem.

## DAFTAR ISI

HALAMAN JUDUL .....	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING .....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI .....	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR.....	iv
HALAMAN PERSEMBAHAN .....	v
HALAMAN MOTO .....	vi
KATA PENGANTAR.....	vii
SARI.....	ix
GLOSARIUM .....	x
DAFTAR ISI .....	xi
DAFTAR TABEL .....	xiii
DAFTAR GAMBAR.....	xv
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	1
1.3 Batasan Masalah .....	2
1.4 Tujuan Penelitian .....	2
1.5 Manfaat Penelitian .....	3
1.6 Metode Penelitian .....	3
1.7 Sistematika Penulisan .....	4
<b>BAB II LANDASAN TEORI .....</b>	<b>5</b>
2.1 Manajemen Proyek Perangkat Lunak .....	5
2.2 Perancangan Perangkat Lunak .....	5
2.2.1 Langkah-Langkah Dalam Perancangan Proyek Perangkat Lunak .....	5
2.2.2 Langkah-Langkah Dalam Perancangan Proyek Perangkat Lunak .....	7
2.3 Kolaborasi Dalam Pengembangan Perangkat Lunak.....	9
2.4 Penelitian Terdahulu .....	10
<b>BAB III METODOLOGI .....</b>	<b>11</b>
3.1 Analisis Kebutuhan Sistem .....	11
3.1.1 Analisis Kebutuhan Perangkat Lunak .....	11
3.1.2 Analisis Kebutuhan Perangkat Keras .....	11
3.1.3 Analisis Kebutuhan Masukan.....	11

3.1.4	Analisis Kebutuhan Proses .....	12
3.1.5	Analisis Kebutuhan Keluaran .....	13
3.2	Perancangan Fungsionalitas .....	14
3.2.1	Rancangan <i>Use Case Diagram</i> .....	17
3.2.2	Rancangan <i>Activity Diagram</i> .....	24
3.3	Perancangan Basisdata .....	71
3.3.1	<i>Entity Relationship Diagram</i> .....	71
3.3.2	Struktur Tabel Basisdata .....	72
3.4	Perancangan Antarmuka .....	78
3.5	Perancangan Pengujian .....	82
BAB IV HASIL DAN PEMBAHASAN .....		85
4.1	Implementasi Sistem .....	85
4.2	Pengujian .....	109
BAB V KESIMPULAN DAN SARAN .....		111
5.1	Kesimpulan .....	111
5.2	Saran .....	111
DAFTAR PUSTAKA .....		112
LAMPIRAN .....		113

## DAFTAR TABEL

Tabel 3.1 Deskripsi setiap <i>use case</i> .....	21
Tabel 3.2 Penjelasan alur kegiatan <i>login</i> .....	25
Tabel 3.3 Penjelasan alur kegiatan melihat daftar <i>project</i> .....	25
Tabel 3.4 Penjelasan alur kegiatan menambah <i>project</i> .....	26
Tabel 3.5 Penjelasan alur kegiatan melihat detail <i>project</i> .....	27
Tabel 3.6 Penjelasan alur kegiatan mengedit detail <i>project</i> .....	27
Tabel 3.7 Penjelasan alur kegiatan menghapus <i>project</i> .....	28
Tabel 3.8 Penjelasan alur kegiatan melihat anggota.....	29
Tabel 3.9 Penjelasan alur kegiatan melihat anggota.....	30
Tabel 3.10 Penjelasan alur kegiatan mengedit anggota.....	31
Tabel 3.11 Penjelasan alur kegiatan menghapus anggota.....	32
Tabel 3.12 Penjelasan alur kegiatan melihat daftar versi .....	33
Tabel 3.13 Penjelasan alur kegiatan menambah versi .....	34
Tabel 3.14 Penjelasan alur kegiatan mengedit versi.....	35
Tabel 3.15 Penjelasan alur kegiatan menghapus versi.....	36
Tabel 3.16 Penjelasan alur kegiatan melihat detail versi.....	37
Tabel 3.17 Penjelasan alur kegiatan melihat pengantar.....	38
Tabel 3.18 Penjelasan alur kegiatan melihat daftar fungsionalitas sistem.....	39
Tabel 3.19 Penjelasan alur kegiatan melihat daftar jenis rancangan diagram.....	40
Tabel 3.20 Penjelasan alur kegiatan melihat daftar desain antarmuka .....	42
Tabel 3.21 Penjelasan alur kegiatan konversi SDD menjadi PDF.....	43
Tabel 3.22 Penjelasan alur kegiatan menambah pengantar .....	44
Tabel 3.23 Penjelasan alur kegiatan mengedit pengantar .....	45
Tabel 3.24 Penjelasan alur kegiatan menambah fungsionalitas sistem .....	47
Tabel 3.25 Penjelasan alur kegiatan mengedit fungsionalitas sistem .....	48
Tabel 3.26 Penjelasan alur kegiatan menghapus fungsionalitas sistem.....	50
Tabel 3.27 Penjelasan alur kegiatan melihat diagram fungsionalitas sistem.....	51
Tabel 3.28 Penjelasan alur kegiatan melihat daftar <i>parent</i> diagram.....	53
Tabel 3.29 Penjelasan alur kegiatan menambah <i>parent</i> diagram .....	54
Tabel 3.30 Penjelasan alur kegiatan melihat daftar <i>child</i> diagram .....	56
Tabel 3.31 Penjelasan alur kegiatan menambah <i>child</i> diagram.....	57
Tabel 3.32 Penjelasan alur kegiatan melihat detail diagram.....	59

Tabel 3.33 Penjelasan alur kegiatan mengedit diagram.....	60
Tabel 3.34 Penjelasan alur kegiatan menghapus diagram .....	62
Tabel 3.35 Penjelasan alur kegiatan menambah desain antarmuka.....	64
Tabel 3.36 Penjelasan alur kegiatan menghapus desain antarmuka .....	66
Tabel 3.37 Penjelasan alur kegiatan melihat detail desain antarmuka.....	67
Tabel 3.38 Penjelasan alur kegiatan mengedit desain antarmuka.....	69
Tabel 3.39 Penjelasan alur kegiatan mengunduh desain antarmuka.....	70
Tabel 3.40 Struktur Tabel <i>Users</i> .....	72
Tabel 3.41 Struktur Tabel <i>Projects</i> .....	72
Tabel 3.42 Struktur Tabel <i>Versions</i> .....	73
Tabel 3.43 Struktur Tabel <i>Introductions</i> .....	73
Tabel 3.44 Struktur Tabel <i>Functionalities</i> .....	74
Tabel 3.45 Struktur Tabel <i>Diagrams</i> .....	74
Tabel 3.46 Struktur Tabel <i>Diagram_types</i> .....	75
Tabel 3.47 Struktur Tabel <i>User_interfaces</i> .....	75
Tabel 3.48 Struktur Tabel <i>Comments</i> .....	76
Tabel 3.49 Struktur Tabel <i>Logs</i> .....	76
Tabel 3.50 Struktur Tabel <i>Glossaries</i> .....	77
Tabel 3.51 Struktur Tabel <i>References</i> .....	77
Tabel 3.52 Struktur Tabel <i>Stakeholders</i> .....	77
Tabel 3.53 Kuesioner Pengujian .....	83
Tabel 4.1 Hasil Kuesioner Pengujian.....	109

## DAFTAR GAMBAR

Gambar 2.1 Langkah-langkah dalam mengelola sebuah proyek .....	6
Gambar 3.1 <i>Use case diagram</i> .....	17
Gambar 3.2 <i>Use case diagram Sub Package: Use case melihat detail project</i> .....	18
Gambar 3.3 <i>Use case diagram Sub Package: Use case melihat detail versi</i> .....	18
Gambar 3.4 <i>Use case diagram Sub Package: Use case melihat penganter</i> .....	19
Gambar 3.5 <i>Use case diagram Sub Package: Use case melihat daftar fungsionalitas sistem</i> . 19	19
Gambar 3.6 <i>Use case diagram Sub Package: Use case melihat jenis rancangan diagram</i> .....	20
Gambar 3.7 <i>Use case diagram Sub Package: Use case melihat daftar parent diagram</i> .....	20
Gambar 3.8 <i>Use case diagram Sub Package: Use case melihat detail parent diagram</i> .....	21
Gambar 3.9 <i>Use case diagram Sub Package: Use case melihat daftar desain antarmuka</i> .....	21
Gambar 3.10 <i>Activity diagram login</i> .....	24
Gambar 3.11 <i>Activity diagram melihat daftar project</i> .....	25
Gambar 3.12 <i>Activity diagram menambah project</i> .....	26
Gambar 3.13 <i>Activity diagram melihat detail project</i> .....	26
Gambar 3.14 <i>Activity diagram mengedit detail project</i> .....	27
Gambar 3.15 <i>Activity diagram menghapus project</i> .....	28
Gambar 3.16 <i>Activity diagram melihat anggota</i> .....	29
Gambar 3.17 <i>Activity diagram menambah anggota</i> .....	30
Gambar 3.18 <i>Activity diagram mengedit anggota</i> .....	31
Gambar 3.19 <i>Activity diagram menghapus anggota</i> .....	32
Gambar 3.20 <i>Activity diagram melihat daftar versi</i> .....	33
Gambar 3.21 <i>Activity diagram menambah versi</i> .....	34
Gambar 3.22 <i>Activity diagram mengedit versi</i> .....	35
Gambar 3.23 <i>Activity diagram menghapus versi</i> .....	36
Gambar 3.24 <i>Activity diagram melihat detail versi</i> .....	37
Gambar 3.25 <i>Activity diagram melihat pengantar</i> .....	38
Gambar 3.26 <i>Activity diagram melihat daftar fungsionalitas sistem</i> .....	39
Gambar 3.27 <i>Activity diagram melihat daftar jenis rancangan diagram</i> .....	40
Gambar 3.28 <i>Activity diagram melihat daftar desain antarmuka</i> .....	41
Gambar 3.29 <i>Activity diagram konversi SDD menjadi PDF</i> .....	42
Gambar 3.30 <i>Activity diagram menambah pengantar</i> .....	43
Gambar 3.31 <i>Activity diagram mengedit pengantar</i> .....	45

Gambar 3.32 <i>Activity diagram</i> menambah fungsionalitas sistem.....	46
Gambar 3.33 <i>Activity diagram</i> mengedit fungsionalitas sistem .....	48
Gambar 3.34 <i>Activity diagram</i> menghapus fungsionalitas sistem.....	49
Gambar 3.35 <i>Activity diagram</i> melihat diagram fungsionalitas sistem .....	51
Gambar 3.36 <i>Activity diagram</i> melihat daftar <i>parent</i> diagram .....	52
Gambar 3.37 <i>Activity diagram</i> menambah <i>parent</i> diagram.....	54
Gambar 3.38 <i>Activity diagram</i> melihat daftar <i>child</i> diagram .....	55
Gambar 3.39 <i>Activity diagram</i> menambah <i>child</i> diagram .....	57
Gambar 3.40 <i>Activity diagram</i> melihat detail diagram.....	58
Gambar 3.41 <i>Activity diagram</i> mengedit diagram.....	60
Gambar 3.42 <i>Activity diagram</i> menghapus diagram.....	62
Gambar 3.43 <i>Activity diagram</i> menambah desain antarmuka .....	64
Gambar 3.44 <i>Activity diagram</i> menghapus desain antarmuka.....	65
Gambar 3.45 <i>Activity diagram</i> melihat detail desain antarmuka.....	67
Gambar 3.46 <i>Activity diagram</i> mengedit desain antarmuka.....	68
Gambar 3.47 <i>Activity diagram</i> mengunduh desain antarmuka.....	70
Gambar 3.48 <i>Entity Relationship Diagram</i> .....	71
Gambar 3.49 Desain antarmuka halaman daftar nama <i>project</i> .....	78
Gambar 3.50 Halaman daftar versi .....	79
Gambar 3.51 Halaman tambah versi.....	79
Gambar 3.52 Halaman menu pengantar.....	80
Gambar 3.53 Halaman menu fungsionalitas sistem.....	80
Gambar 3.54 Halaman menu rancangan diagram.....	81
Gambar 3.55 Halaman desain antarmuka .....	81
Gambar 3.56 Halaman lihat detail desain antarmuka .....	82
Gambar 3.57 Halaman tambah desain antarmuka .....	82
Gambar 4.1 Halaman <i>login</i> .....	85
Gambar 4.2 Halaman <i>register</i> .....	86
Gambar 4.3 Halaman daftar <i>project</i> .....	87
Gambar 4.4 Halaman tambah <i>project</i> .....	87
Gambar 4.5 Halaman <i>dashboard project</i> .....	88
Gambar 4.6 Halaman <i>log project</i> .....	88
Gambar 4.7 Halaman daftar versi .....	89
Gambar 4.8 Halaman tambah versi.....	90



Gambar 4.9 Halaman edit versi .....	90
Gambar 4.10 Konfirmasi hapus versi .....	91
Gambar 4.11 Halaman daftar anggota .....	91
Gambar 4.12 Halaman edit anggota.....	92
Gambar 4.13 Konfirmasi hapus anggota .....	92
Gambar 4.14 Halaman pengaturan .....	93
Gambar 4.15 Form edit detail <i>project</i> .....	93
Gambar 4.16 Konfirmasi hapus <i>project</i> .....	94
Gambar 4.17 Halaman <i>dashboard</i> versi .....	94
Gambar 4.18 Halaman <i>log</i> versi .....	95
Gambar 4.19 Halaman menu pengantar.....	96
Gambar 4.20 Halaman menu fungsionalitas sistem.....	96
Gambar 4.21 Halaman tambah fungsionalitas sistem.....	97
Gambar 4.22 Halaman edit fungsionalitas sistem.....	97
Gambar 4.23 Konfirmasi hapus fungsionalitas sistem .....	98
Gambar 4.24 Halaman menu diagram .....	98
Gambar 4.25 Halaman tambah jenis diagram.....	99
Gambar 4.26 Halaman edit tipe diagram .....	99
Gambar 4.27 Konfirmasi hapus jenis diagram .....	100
Gambar 4.28 Halaman daftar <i>parent</i> diagram .....	101
Gambar 4.29 Halaman tambah <i>parent</i> diagram.....	101
Gambar 4.30 Halaman tambah <i>parent</i> diagram ketika memilih pilihan <i>yes</i> .....	102
Gambar 4.31 Halaman daftar <i>sub</i> diagram .....	102
Gambar 4.32 Halaman tambah <i>sub</i> diagram.....	103
Gambar 4.33 Halaman detail diagram .....	104
Gambar 4.34 Halaman edit diagram .....	104
Gambar 4.35 Konfirmasi hapus diagram .....	105
Gambar 4.36 Halaman daftar desain antarmuka.....	105
Gambar 4.37 Halaman tambah desain antarmuka .....	106
Gambar 4.38 Konfirmasi hapus desain antarmuka .....	106
Gambar 4.39 Halaman detail desain antarmuka .....	107
Gambar 4.40 Halaman edit desain antarmuka .....	107
Gambar 4.41 Halaman profil .....	108
Gambar 4.42 Halaman edit profil .....	108

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Perangkat lunak yang baik adalah perangkat lunak yang memiliki perancangan dan dokumentasi yang baik dalam proses pembuatannya. Dokumentasi perangkat lunak merupakan sebuah artefak yang tujuannya untuk menyampaikan informasi tentang sistem perangkat lunak yang menyertainya (Forward, 2002). Pembuatan dokumentasi dapat memberikan banyak manfaat bagi para pengembang perangkat lunak. Menurut Sommerville dalam (Najwaini, 2012), dokumentasi memiliki fungsi sebagai berikut:

- a. Bertindak sebagai media komunikasi antar anggota pengembang tim
- b. Penyimpanan sistem informasi untuk digunakan oleh *maintenance engineers*
- c. Membantu manajer proyek dalam merencanakan, mengatur anggaran, dan penjadwalan dalam proses pembangunan perangkat lunak
- d. Memberi penjelasan kepada pengguna bagaimana cara menggunakan dan mengelola sistem yang dibangun

Sebagai tempat penyimpanan informasi, dokumen harus berisi informasi yang lengkap, valid, mudah dimengerti, dan selalu diperbaharui. Beberapa *software engineer* beranggapan bahwa dokumentasi cukup dengan *source code*, sehingga tidak dibutuhkan dokumen tambahan. Hal ini mungkin dapat berlaku jika program yang dibuat untuk dirinya sendiri. Namun akan menjadi sulit apabila program tersebut digunakan oleh orang lain atau program tersebut merupakan bagian dari suatu perangkat lunak yang dikerjakan oleh banyak orang.

Dalam pembuatan perangkat lunak, biasanya perangkat lunak dibuat oleh sebuah tim yang terdiri dari beberapa orang yang dimana masing-masing anggota dari tim tersebut memiliki tugas yang berbeda-beda. Setiap anggota tim berkolaborasi untuk menghasilkan sebuah perangkat lunak yang baik. Sebelum dilakukan pembuatan oleh *programmer*, akan dilakukan perancangan dan analisis kebutuhan sistem yang dilakukan oleh bagian analisis sistem. Hasil dari analisis yang dilakukan tim analisis sistem diberikan kepada tim desain untuk dibuatkan desain antarmuka dari sistem yang dibuat. Kemudian hasil desain antarmuka dan analisis kebutuhan sistem dituliskan dalam sebuah dokumen *Software Design Document* (SDD). *Software Design Document* (SDD) adalah representasi dari sistem perangkat lunak yang digunakan sebagai media untuk mengkomunikasikan informasi desain perangkat lunak (IEEE, 2009). Tujuan dari dibuatnya *Software Design Document* (SDD) sendiri adalah untuk

memberikan deskripsi tentang desain dari suatu sistem yang memungkinkan untuk digunakan mengembangkan perangkat lunak dengan pemahaman tentang apa yang akan dibangun dan bagaimana perangkat lunak tersebut diharapkan untuk dibangun (Mcelrath, 2007).

Pada proses pembuatannya, *Software Design Document* (SDD) masih dibuat secara konvensional menggunakan aplikasi pengolah kata seperti Microsoft Word, Open Office, dan lain sebagainya. Sehingga apabila anggota tim lain ingin menambahkan ataupun mengubah isi dari *Software Design Document*, anggota tim yang terakhir mengerjakan harus mengirim *Software Design Document* terbaru agar tidak terjadi kesalahan data. Dengan cara saling mengirim dokumen tersebut ditambah lagi apabila salah satu yang memiliki dokumen terbaru dari dokumen perancangan *software* tersebut sedang tidak menggunakan komputer, tentu akan menghambat anggota tim lain dalam perancangan maupun pembuatan suatu sistem. Selain itu, pembuatan secara manual dapat berakibat hilangnya data pada salah satu anggota tim dengan berbagai penyebab seperti salah menghapus file dan lain lain.

Berdasarkan penjelasan diatas, penulis ingin membantu tim pengembang *software* dengan membuat Sistem Online Untuk Dokumen Perancangan Perangkat Lunak Bagi Perusahaan/Tim Pengembang berbasis web yang dapat digunakan tim pengembang *software* untuk berkolaborasi dalam membuat perancangan maupun dalam pembuatan suatu sistem. Sistem ini dapat menyimpan seluruh *requirement* suatu *software* hingga rancangan seperti *Data Flow Diagram* (DFD), *Use Case Diagram* (UCD), *Activity Diagram*, dan lain lain.

## 1.2 Rumusan Masalah

Bagaimana membuat sistem yang dapat digunakan untuk mengelola dokumen perancangan perangkat lunak sekaligus meningkatkan kualitas kolaborasi pada tim pengembang?

## 1.3 Batasan Masalah

Adapun batasan pekerjaan dalam penelitian tugas akhir ini adalah sebagai berikut:

- a. Sistem yang dikembangkan merupakan aplikasi berbasis web
- b. Standar SDD yang digunakan dalam sistem yang dikembangkan menggunakan standar IEEE

## 1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah menghasilkan sebuah sistem online untuk pembuatan dokumen perancangan perangkat lunak dalam bentuk aplikasi berbasis web yang dapat

memudahkan perusahaan ataupun tim pengembang perangkat lunak dalam bekerjasama membuat sebuah dokumen perancangan perangkat lunak.

### 1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut:

- a. Bagi Penulis
  1. Menambah pengetahuan penulis mengenai *Software Design Document*
  2. Mengasah kemampuan penulis dalam pengembangan sistem online pembuatan SDD
  3. Menambah pengetahuan penulis mengenai pengembangan perangkat lunak yang baik
- b. Bagi Tim Pengembang
  1. Memudahkan dalam pembuatan SDD karena dapat dilakukan secara online
  2. Memudahkan tim pengembang berkolaborasi dalam pembuatan dan penggunaan SDD
  3. Memudahkan tim pengembang dalam mengembangkan SDD pada perangkat lunak sejenis di masa mendatang

### 1.6 Metode Penelitian

Adapun metode penelitian yang digunakan dalam penelitian tugas akhir ini adalah sebagai berikut :

- a. Analisis kebutuhan

Terdapat beberapa tahapan analisis diantaranya adalah analisis kebutuhan perangkat lunak, analisis kebutuhan perangkat keras, analisis kebutuhan masukan, analisis kebutuhan proses, dan analisis kebutuhan keluaran. Adapun rincian dari tahapan ini akan dijelaskan pada bab tiga analisis sistem.
- b. Perancangan sistem

Pada tahapan ini dilakukan perancangan yang nantinya akan dipakai pada proses implementasi. Rancangan sistem yang disiapkan meliputi kebutuhan fungsionalitas, ERD (*Entity Relationship Diagram*), *Use Case Diagram*, dan lain-lain. Adapun rincian dari tahapan ini akan dijelaskan pada bab empat perancangan sistem.
- c. Implementasi

Pada tahapan ini dilakukan implementasi dari hasil perancangan sistem yang telah dibuat sebelumnya. Aplikasi pembuatan dokumen perancangan perangkat lunak ini akan diimplementasikan dalam bentuk aplikasi web. Adapun rincian dari tahapan ini akan dijelaskan pada bab lima implementasi dan pengujian sistem.

d. Pengujian

Pada tahapan ini dilakukan pengujian terhadap sistem yang telah dibuat untuk mengukur seberapa efektif hasil dari sistem untuk membuat dokumen perancangan perangkat lunak. Adapun rincian dari tahapan ini akan dijelaskan pada bab lima implementasi dan pengujian sistem.

## 1.7 Sistematika Penulisan

Sistematika penulisan ini bertujuan untuk memudahkan dalam melakukan pembahasan pada tugas akhir ini. Secara garis besar sistematika penulisan dalam laporan ini dibagi menjadi enam bab. Adapun rincian dari enam bab tersebut adalah sebagai berikut:

a. Bab I Pendahuluan

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

b. Bab II Landasan Teori

Pada bab ini membahas mengenai manajemen proyek perancangan perangkat lunak, tahapan perancangan perangkat lunak, kolaborasi tim perangkat lunak dan review sistem sejenis yang akan menjadi dasar dalam pengembangan sistem online untuk kolaborasi tim dalam pembuatan dokumen perancangan perangkat lunak.

c. Bab III Metodologi

Pada bab ini berisi identifikasi masalah, gambaran umum sistem, solusi penyelesaian masalah, analisis kebutuhan, perancangan sistem, dan desain pengujian yang diperlukan dalam pengembangan sistem.

d. Bab IV Hasil dan Pembahasan

Pada bab ini berisi hasil implementasi dan rancangan yang telah dibuat serta hasil pengujian dari sistem yang telah dibuat terhadap calon pengguna.

e. Bab V Kesimpulan dan Saran

Pada bab ini berisi kesimpulan dari penelitian yang telah dilakukan dan saran untuk pengembangan sistem ini kedepannya.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Manajemen Proyek Perangkat Lunak**

Manajemen proyek adalah perencanaan, penjadwalan, dan pengendalian aktivitas proyek untuk mencapai tujuan proyek (Simarmata, 2010). Manajemen proyek perangkat lunak merupakan manajemen proyek yang berfokus pada kegiatan mengembangkan perangkat lunak. Menurut Arifin (2011), sifat manajemen proyek harus memenuhi hal-hal berikut:

- a. Menyelesaikan masalah
- b. Mengerjakan sesuatu hingga selesai
- c. Memiliki batas waktu mulai dan selesainya
- d. Membutuhkan resource/sumber daya dan waktu
- e. Bagi beberapa orang merupakan kesempatan dan menarik

Manajemen proyek sendiri dibuat dengan beberapa tujuan. Menurut Arifin (2011), manajemen perangkat lunak memiliki tujuan sebagai berikut:

- a. Bagi *Project Manager*

Menggambarkan status proyek kepada manajer senior dan *Stakeholder* dan merencanakan aktivitas tim proyek

- b. Bagi anggota tim proyek

Memahami konteks pekerjaan

- c. Bagi manajer senior

Memastikan apakah biaya dan waktu yang dialokasikan masuk akal dan terkendali serta melihat apakah proyek dilaksanakan secara efisien dan *cost effective*

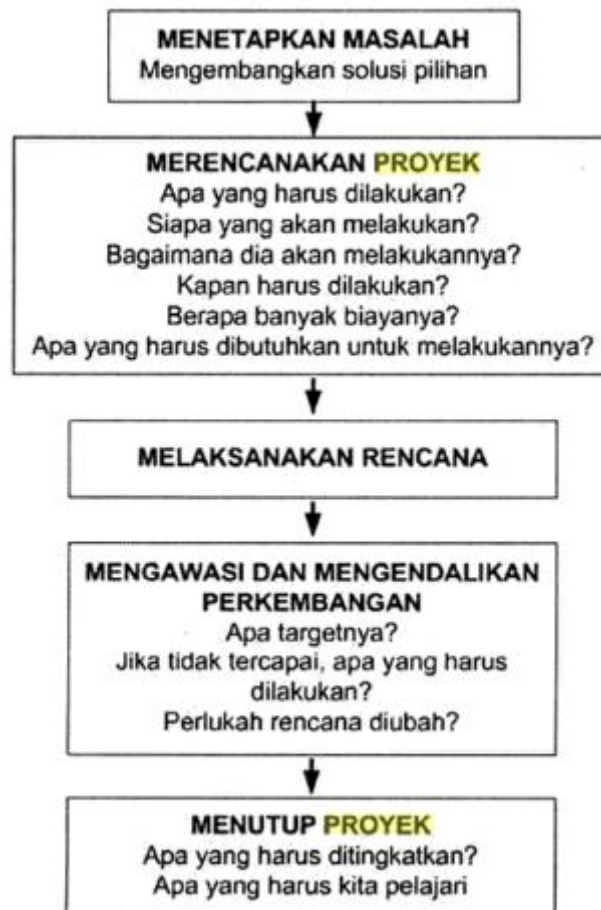
- d. Bagi *Stakeholder*

Memastikan apakah proyek masih berada pada jalurnya dan memastikan kebutuhan mereka sedang diakomodir oleh proyek

#### **2.2 Perancangan Perangkat Lunak**

##### **2.2.1 Langkah-Langkah Dalam Perancangan Proyek Perangkat Lunak**

Dalam membuat perancangan sebuah proyek perangkat lunak dibutuhkan langkah-langkah yang tepat agar dapat berjalan dengan baik. Langkah-langkah manajemen proyek dapat dilihat pada Gambar 2.1.



Gambar 2.1 Langkah-langkah dalam mengelola sebuah proyek

Sumber: Simarmata (2010)

Berikut penjelasan dari langkah-langkah yang terdapat dalam pengelolaan sebuah proyek menurut Simarmata (2010):

a. Menetapkan masalah

Langkah ini akan mengidentifikasi masalah yang akan diselesaikan oleh proyek. Langkah ini membantu dalam memvisualisasikan hasil akhir yang diinginkan.

b. Mengembangkan solusi pilihan

Langkah ini digunakan untuk mengembangkan beberapa cara yang berbeda yang dapat digunakan menyelesaikan suatu masalah.

c. Merencanakan proyek

Langkah ini digunakan untuk menentukan apa saja yang harus dilakukan, siapa sasarannya, dan lain-lain. Dalam konteks perangkat lunak perencanaan dapat dibuat dengan dokumen perancangan perangkat lunak atau yang biasa disebut *Software Design Document* (SDD)

d. Melaksanakan rencana

Langkah ini merupakan realisasi dari perencanaan yang sudah dibuat sebelumnya.

e. Mengawasi dan mengendalikan perkembangan

Langkah ini digunakan untuk mengawasi jalannya proyek agar jika ditemukan penyimpangan dari rencana dapat segera menentukan langkah apa yang harus diambil dengan mempertimbangkan kondisi terbaru.

f. Menutup proyek

Pada langkah terakhir dilakukan audit, yaitu mempelajari sesuatu apa yang baru saja dilakukan untuk evaluasi agar dapat ditingkatkan lagi di proyek selanjutnya.

## 2.2.2 Langkah-Langkah Dalam Perancangan Proyek Perangkat Lunak

### Definisi *Software Design Document*

Dalam proses pembuatan perangkat lunak, dibutuhkan perancangan yang baik agar menghasilkan perangkat lunak yang berkualitas, sehingga dibuatlah *Software Design Document*. *Software Design Document* adalah representasi dari perancangan perangkat lunak yang digunakan sebagai media untuk mengkomunikasikan informasi desain perangkat lunak kepada *stakeholder* (IEEE, 2009).

### Konten *Software Design Document*

Menurut IEEE (2009), sebagai dokumen yang berguna untuk menjelaskan rancangan dari sebuah perangkat lunak, sebuah SDD harus memuat beberapa poin penting, diantaranya:

a. Identifikasi *Software Design Document*

Sebuah SDD harus menyertakan informasi deskriptif berikut:

1. Tanggal terbit dan status
2. Cakupan
3. Organisasi Penerbit
4. Kepenulisan (informasi dapat dipertanggung jawabkan atau memiliki hak cipta)
5. Referensi
6. Konteks
7. Satu atau lebih bahasa desain untuk setiap sudut pandang desain yang digunakan
8. Isi
9. Ringkasan
10. Glosarium
11. Sejarah perubahan



b. *Stakeholder* yang teridentifikasi

Sebuah SDD harus mengidentifikasi *stakeholder* desain untuk *subjek* desain. *Stakeholder* merupakan semua pihak baik internal maupun eksternal yang mempunyai hubungan yang bersifat mempengaruhi maupun dipengaruhi, bersifat langsung maupun tidak langsung terhadap sebuah proyek.

c. Masalah desain yang teridentifikasi

SDD harus mengidentifikasi masalah desain dari masing-masing *stakeholder* yang teridentifikasi dan juga harus bisa mengatasi setiap masalah desain yang diidentifikasi

d. Sudut pandang desain yang dipilih, masing-masing dengan definisi jenis elemen desain yang diizinkan dan bahasa desaini

Untuk setiap tampilan desain dalam SDD, akan ada tepat satu sudut pandang desain yang mengaturnya. Setiap sudut pandang desain harus ditentukan oleh:

1. Nama sudut pandang
2. Masalah desain yang merupakan topik dari sudut pandang
3. Elemen desain yang didefinisikan oleh sudut pandang
4. Metode analisis atau operasi lain yang akan digunakan dalam membangun pandangan desain berdasarkan sudut pandang dan kriteria untuk menafsirkan dan mengevaluasi desain
5. Sumber sudut pandang (misalnya kepenulisan atau kutipan), bila berlaku

e. Tampilan desain

SDD harus diatur ke dalam satu atau beberapa tampilan desain. Setiap tampilan desain dalam SDD harus sesuai dengan sudut pandang desain yang mengaturnya. Tujuan dari pandangan desain adalah untuk mengatasi masalah desain yang berkaitan dengan *subjek* desain yang memungkinkan *stakeholder* untuk focus pada detail desain dari perspektif tertentu dan memenuhi persyaratan yang relevan secara efektif.

f. Overlay desain

Overlay desain digunakan untuk menyajikan informasi tambahan sehubungan dengan tampilan desain yang sudah ditentukan. Setiap overlay desain harus diberi nama dan ditandai secara unik sebagai overlay. Selain itu, setiap overlay desain harus jelas terkait dengan satu sudut pandang

g. Rasionalisasi desain

Rasionalisasi digunakan untuk menjelaskan alasan desainer membuat suatu desain dan membenaran atas keputusan tersebut.

### 2.3 Kolaborasi Dalam Pengembangan Perangkat Lunak

Kolaborasi adalah dasar untuk menyatukan pengetahuan, pengalaman dan keterampilan dari beberapa anggota tim untuk berkontribusi pada pengembangan produk baru secara lebih efektif daripada anggota tim individu yang melakukan tugas sempit mereka untuk mendukung pengembangan produk (Kusumasari, Supriana, Surendro, & Sastramihardja, 2011).

Pengembangan perangkat lunak dapat dinyatakan sebagai aktivitas penyelesaian persoalan secara kooperatif dan kolaborasi tim dimana kesuksesannya tergantung pada akuisisi pengetahuan, pertukaran dan integrasi informasi, dan meminimalisasi kegagalan komunikasi (Kusumasari, Sastramihardja, Surendro, & Supriana, 2013). Aktivitas pengembangan perangkat lunak seperti analisis, desain, pengujian, dan *coding* dapat dilakukan oleh berbagai kombinasi individu. Oleh karena itu kolaborasi dan koordinasi dalam proyek pengembangan perangkat lunak memainkan peran penting dalam menentukan keberhasilan proyek perangkat lunak.

Ada berbagai model kolaborasi tim. Model-model tersebut berfokus pada berbagai aspek dan juga menjelaskan aspek-aspek tersebut pada tingkat detail yang berbeda. Salah satu model kolaborasi tim pengembang perangkat lunak yang sedang populer yaitu *metodologi agile*. Kolaborasi yang dilakukan oleh tim pengembang dalam *metodologi agile* adalah sebagai berikut:

- a. Orang-orang bisnis dan developer harus bekerja bersama setiap hari di seluruh proyek.
- b. Metode yang paling efisien dan efektif dalam menyampaikan informasi kepada dan di dalam tim pengembangan adalah percakapan dengan bertemu secara langsung.
- c. Bekerja individu dan interaksi terhadap proses dan alat

Sedangkan faktor yang menentukan keberhasilan proyek perangkat lunak menggunakan *metodologi agile* adalah sebagai berikut:

- a. Komunikasi langsung dan komunitas kecil
- b. Pakar penggunaan berada di tempat pengerjaan
- c. Penambahan singkat
- d. Uji regresi otomatis
- e. *Developer* yang berpengalaman

## 2.4 Penelitian Terdahulu

Penelitian terdahulu merupakan salah satu acuan yang digunakan dalam melakukan penelitian Sistem Online untuk Dokumen Perancangan Perangkat Lunak Bagi Perusahaan atau Tim Pengembang. Penelitian ini memiliki topik mengenai perencanaan Software Design Document, sehingga acuan yang digunakan yaitu mengenai penelitian yang berkaitan dengan topik utama penelitian ini juga.

Dalam tugas akhir yang berjudul Aplikasi Manajemen Dokumen Perancangan Perangkat Lunak (Ferriana, 2015), memaparkan tentang pembangunan sebuah sistem untuk pembuatan dokumen perancangan perangkat lunak agar pembuatan dokumen perancangan perangkat lunak tidak dilakukan secara manual. Hasil akhir dari penelitian ini berupa sebuah dokumen perancangan perangkat lunak yang dapat dilihat dalam bentuk dokumen maupun didalam sistem secara langsung. Namun, penelitian ini masih memiliki beberapa kekurangan, diantaranya:

- a. Alur sistem kurang simpel
- b. Belum ada manajemen versi setiap proyeknya
- c. Antarmuka terlalu kaku dan belum *responsive*
- d. *Role user* yang disediakan sistem terlalu sedikit untuk mendukung kolaborasi tim pengembang
- e. Hanya berfokus pada pembuatan SDD namun kurang memperhatikan dari sisi kolaborasi tim pengembang

Kesimpulan dari hasil penelitian terdahulu menunjukkan persamaan tujuan yaitu bagaimana membuat dokumen perancangan perangkat lunak dengan menggunakan sistem agar tidak dibuat secara manual. Namun selain berfokus pada pembuatan dokumen perancangan perangkat lunak dengan menggunakan sistem, penelitian kali ini juga berfokus pada bagaimana memudahkan kolaborasi antar tim pengembang perangkat lunak agar dapat menghasilkan dokumen perancangan perangkat lunak yang lebih baik.

## **BAB III**

### **METODOLOGI**

#### **3.1 Analisis Kebutuhan Sistem**

Dalam pembuatan “Sistem Online Untuk Dokumen Perancangan Perangkat Lunak Bagi Perusahaan atau Tim Pengembang” terdapat beberapa analisis kebutuhan, yaitu analisis kebutuhan perangkat lunak, analisis kebutuhan perangkat keras, analisis kebutuhan masukan, analisis kebutuhan proses, dan analisis kebutuhan keluaran.

##### **3.1.1 Analisis Kebutuhan Perangkat Lunak**

Untuk menjalankan “Sistem Online Untuk Dokumen Perancangan Perangkat Lunak Bagi Perusahaan atau Tim Pengembang” dibutuhkan perangkat lunak yang dapat mendukung agar dapat berjalan dengan baik, diantaranya sebagai berikut:

- a. Browser yang dapat digunakan adalah Google Chrome dan Mozilla Firefox. Untuk Google Chrome minimum versi 67.0.3396.99 atau lebih tinggi dan untuk Mozilla Firefox minimum versi 59.0.2 atau lebih tinggi.
- b. Untuk sistem operasi Windows, disarankan menggunakan minimal Windows XP atau lebih tinggi.

##### **3.1.2 Analisis Kebutuhan Perangkat Keras**

Dalam pengerjaan pengembangan “Sistem Online Untuk Dokumen Perancangan Perangkat Lunak Bagi Perusahaan atau Tim Pengembang”, selain menggunakan perangkat lunak tentunya juga dibutuhkan perangkat keras untuk mendukung kinerja dari perangkat lunak tersebut. Adapun perangkat keras / komputer yang mendukung untuk menjalankan sistem ini yaitu memiliki processor Intel Pentium 4 atau lebih tinggi.

##### **3.1.3 Analisis Kebutuhan Masukan**

Analisis kebutuhan masukan dilakukan untuk mengidentifikasi masukan-masukan apa saja yang harus dimasukkan oleh pengguna ke sistem untuk menjalankan fungsionalitas sistem sebagaimana mestinya. Berikut analisis kebutuhan masukan dari “Sistem Online Untuk Dokumen Perancangan Perangkat Lunak Bagi Perusahaan atau Tim Pengembang”:

- a. Data Proyek

Data proyek berisi nama proyek yang hanya bisa diedit oleh *owner* proyek tersebut

- b. Data Versi  
Data versi berisi nama versi yang dibuat sesuai nomor urut release versi proyek tersebut. Setiap proyek dapat memiliki banyak versi.
- c. Data Pengantar  
Data pengantar berisi 5 data, yaitu tujuan, ruang lingkup, glosarium, referensi, dan *stakeholder* dari sebuah proyek
- d. Data Fungsionalitas Sistem  
Data fungsionalitas sistem berisi nama fungsionalitas. Setiap fungsionalitas sistem dapat berelasi dengan rancangan diagram yang ada.
- e. Data Rancangan Diagram  
Data rancangan diagram berupa gambar diagram beserta judulnya. Data diagram dapat dihubungkan dengan fungsionalitas sistem yang ada. Setiap proyek dapat memiliki banyak rancangan diagram.
- f. Data Desain Antarmuka  
Data desain antarmuka berupa gambar desain antarmuka sistem. Setiap proyek dapat memiliki banyak desain antarmuka.

#### **3.1.4 Analisis Kebutuhan Proses**

Analisis kebutuhan proses dilakukan untuk mengidentifikasi proses-proses apa saja yang harus dapat ditangani oleh sistem. Berikut analisis kebutuhan proses dari “Sistem Online Untuk Dokumen Perancangan Perangkat Lunak Bagi Perusahaan atau Tim Pengembang” sesuai *role user* pada sebuah proyek:

- a. Proses yang dapat dilakukan *user* dengan *role Project Manager*
  - 1. Proses mengelola data proyek
  - 2. Proses mengelola data versi
  - 3. Proses mengelola data pengantar
  - 4. Proses mengelola data fungsionalitas sistem
  - 5. Proses mengelola data rancangan diagram
  - 6. Proses mengelola data desain antarmuka
  - 7. Proses konversi SDD menjadi PDF
- b. Proses yang dapat dilakukan *user* dengan *role System Analyst*
  - 1. Proses mengelola data pengantar
  - 2. Proses mengelola data fungsionalitas sistem

3. Proses mengelola data rancangan diagram
  4. Proses konversi SDD menjadi PDF
- c. Proses yang dapat dilakukan *user* dengan *role UI designer*
1. Proses mengelola data desain antarmuka
  2. Proses konversi SDD menjadi PDF
- d. Proses yang dapat dilakukan *user* dengan *role Programmer*
1. Proses konversi SDD menjadi PDF

### 3.1.5 Analisis Kebutuhan Keluaran

Analisis kebutuhan keluaran dilakukan untuk mengidentifikasi keluaran-keluaran apa saja yang akan ditampilkan kepada pengguna. Berikut analisis kebutuhan keluaran dari “Sistem Online Untuk Dokumen Perancangan Perangkat Lunak Bagi Perusahaan atau Tim Pengembang”:

- a. Informasi daftar proyek
 

Setiap pengguna dapat melihat daftar proyek yang ia miliki. Daftar proyek yang tampil berupa nama proyek beserta waktu terakhir proyek tersebut diedit. Dan apabila pengguna memilih salah satu proyek, pengguna dapat melihat detail dari proyek yang dipilih.
- b. Informasi daftar versi
 

Setiap pengguna dapat melihat daftar versi dari suatu proyek. Daftar versi yang tampil berupa nama versi beserta waktu terakhir data versi tersebut diedit.
- c. Informasi pengantar
 

Informasi pengantar terdiri dari 5 jenis data, yaitu data tujuan, data ruang lingkup, data glossarium, data referensi, dan data *stakeholder*. Setiap versi proyek dapat memiliki banyak glossarium, referensi, dan *stakeholder*.
- d. Informasi fungsionalitas sistem
 

Informasi fungsionalitas sistem disajikan dalam bentuk daftar nama fungsionalitas sistem. Setiap fungsionalitas sistem dapat memiliki relasi dengan diagram dan desain antarmuka. Jika pengguna memilih salah satu fungsionalitas sistem, maka akan tampil diagram dan desain antarmuka yang berelasi dengan fungsionalitas sistem tersebut.
- e. Informasi rancangan diagram
 

Informasi rancangan diagram berisi daftar tipe diagram deskripsi diagram, dan gambar dari diagram. Setiap versi dapat memiliki banyak diagram dan setiap diagram dapat direlasikan dengan fungsionalitas sistem.

f. Informasi desain antarmuka

Informasi desain antarmuka berisi nama desain, deskripsi, serta gambar dari desain antarmuka. Setiap versi dapat memiliki banyak desain antarmuka.

### 3.2 Perancangan Konten *Software Design Document*

Pada penelitian ini, standar SDD yang digunakan dalam sistem menggunakan standar SDD dari IEEE 1016-2009, namun tidak semua atau hanya sebagian dari konten pada standar IEEE 1016-2009 yang digunakan. Berikut konten SDD yang digunakan didalam sistem ini:

a. Identifikasi *Software Design Document*

1. Konteks/Tujuan
2. Cakupan
3. Glossarium
4. Referensi
5. Stakeholder

b. Tampilan Desain

1. Desain Diagram
2. Desain Antarmuka

### 3.3 Perancangan Fungsionalitas

“Sistem Online Untuk Dokumen Perancangan Perangkat Lunak Bagi Perusahaan atau Tim Pengembang” memiliki banyak fungsionalitas yang dapat digunakan oleh penggunanya. Dari semua fungsionalitas yang ada, tidak semua fungsionalitas dapat digunakan oleh setiap *role* pengguna. Setiap *role* memiliki batasan dalam penggunaan fungsionalitas. Berikut pembagian fungsionalitas setiap *role user*:

a. *Project Manager*

1. Melihat Daftar Proyek
2. Menambah Daftar Proyek
3. Mengedit Daftar Proyek
4. Menghapus Daftar Proyek
5. Melihat List Versi
6. Menambah List Versi
7. Mengedit List Versi

8. Menghapus List Versi
9. Melihat Pengantar
10. Menambah Pengantar
11. Mengedit Pengantar
12. Melihat Fungsionalitas Sistem
13. Menambah Fungsionalitas Sistem
14. Mengedit Fungsionalitas Sistem
15. Melihat Rancangan Diagram
16. Menambah Rancangan Diagram
17. Mengedit Rancangan Diagram
18. Melihat Desain Antarmuka
19. Menambah Desain Antarmuka
20. Mengedit Desain Antarmuka
21. Menghapus Desain Antarmuka
22. Mengunduh Desain Antarmuka
23. Memberi Komentar
24. Convert SDD menjadi PDF
25. Mengelola anggota proyek

b. *System Analyst*

1. Melihat Daftar Proyek
2. Menambah Daftar Proyek
3. Mengedit Daftar Proyek
4. Menghapus Daftar Proyek
5. Melihat List Versi
6. Menambah List Versi
7. Mengedit List Versi
8. Menghapus List Versi
9. Melihat Pengantar
10. Menambah Pengantar
11. Mengedit Pengantar
12. Melihat Fungsionalitas Sistem
13. Menambah Fungsionalitas Sistem
14. Mengedit Fungsionalitas Sistem
15. Melihat Rancangan Diagram



16. Menambah Rancangan Diagram
17. Mengedit Rancangan Diagram
18. Melihat Desain Antarmuka
19. Mengunduh Desain Antarmuka
20. Memberi Komentar
21. Convert SDD menjadi PDF

c. *UI Designer*

1. Melihat Daftar Proyek
2. Melihat Daftar Versi
3. Melihat Pengantar
4. Melihat Fungsionalitas Sistem
5. Melihat Use Case Diagram
6. Melihat *Activity Diagram*
7. Melihat Entity Relationship Diagram
8. Melihat Desain Antarmuka
9. Menambah Desain Antarmuka
10. Mengedit Desain Antarmuka
11. Menghapus Desain Antarmuka
12. Mengunduh Desain Antarmuka
13. Memberi Komentar
14. Convert SDD menjadi PDF

d. *Programmer*

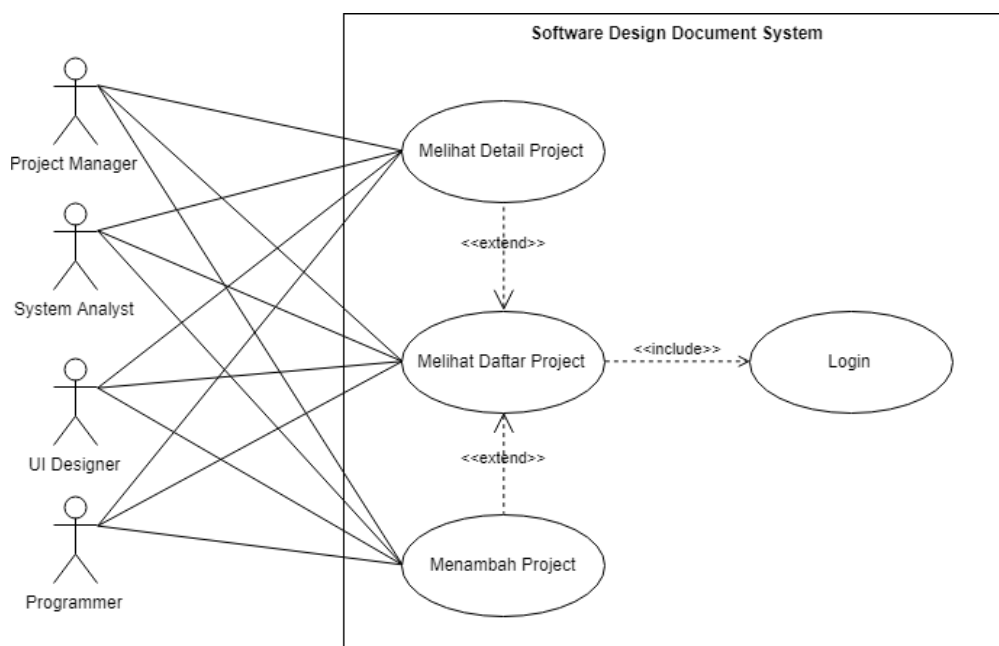
1. Melihat Daftar Proyek
2. Melihat Daftar Versi
3. Melihat Pengantar
4. Melihat Fungsionalitas Sistem
5. Melihat Rancangan Diagram
6. Melihat Desain Antarmuka
7. Mengunduh Desain Antarmuka
8. Memberi Komentar
9. Convert SDD menjadi PDF

Untuk setiap *user* biasa dapat menambahkan proyek baru dan akan otomatis memiliki *role* owner dalam proyek yang *user* tersebut buat. Pada bagian pengantar sendiri berisi 5 poin, yaitu:

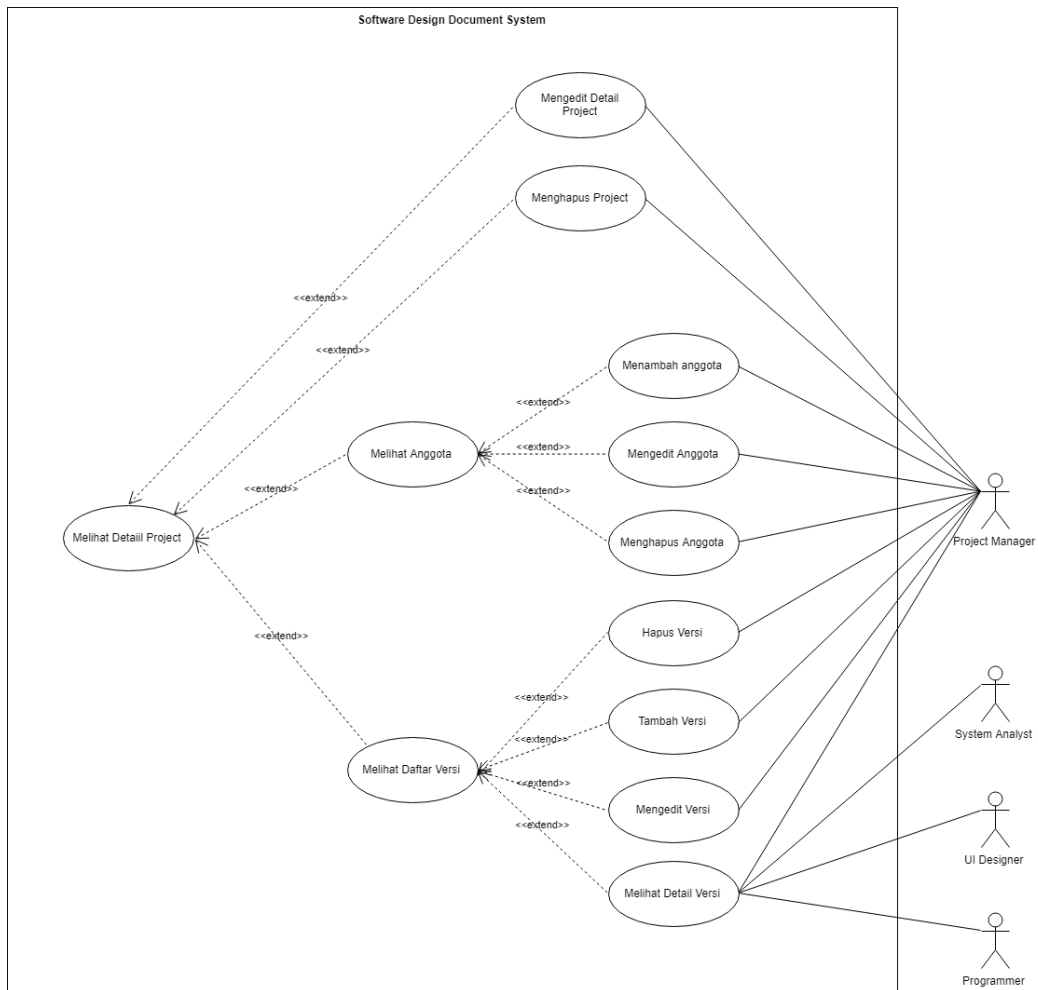
- a. Tujuan
- b. Ruang lingkup
- c. Glosarium
- d. Referensi
- e. *Stakeholder*

### 3.3.1 Rancangan *Use Case Diagram*

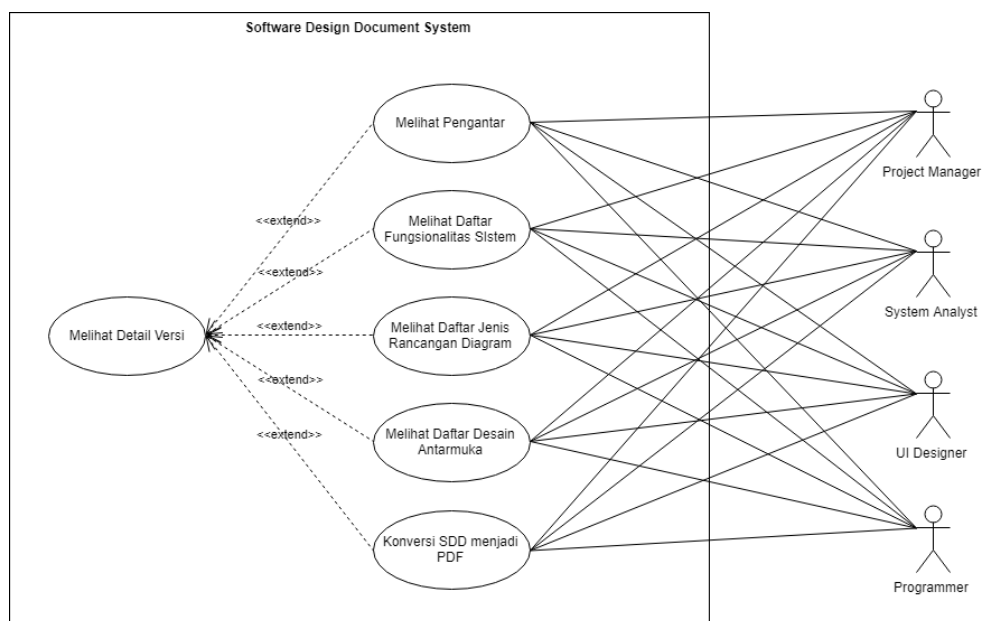
*Use Case Diagram* digunakan untuk mendefinisikan interaksi-interaksi dari satu atau lebih aktor dari sebuah sistem. *Use Case Diagram* juga turut mendefinisikan fungsi-fungsi yang akan berada di dalam sebuah sistem dan actor-aktor apa saja yang berhak menggunakan fungsi-fungsi tersebut. Adapun *use case diagram* pada sistem ini dapat dilihat pada Gambar 3.1 hingga Gambar 3.9



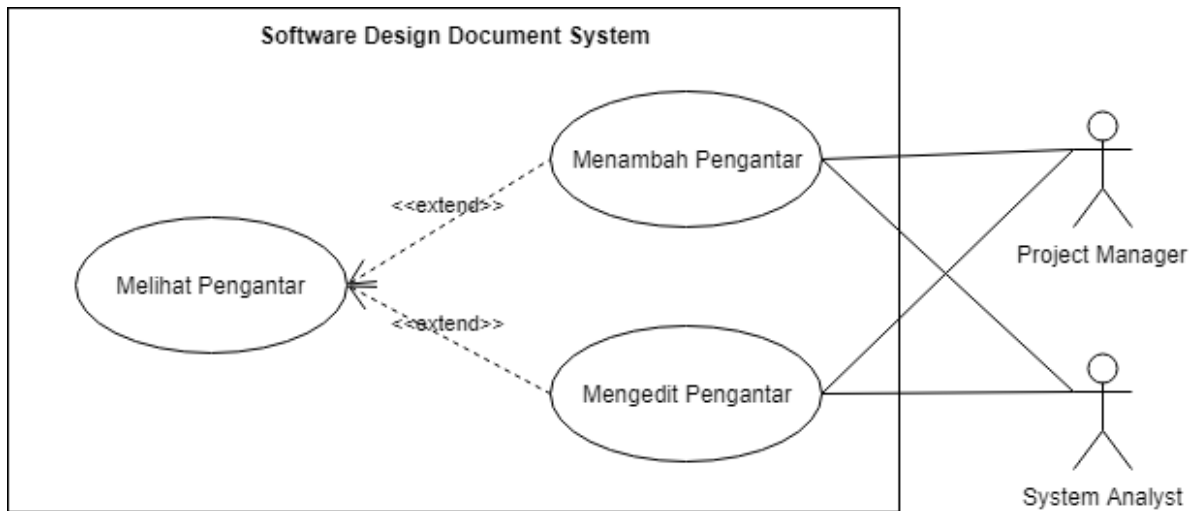
Gambar 3.1 *Use case diagram*



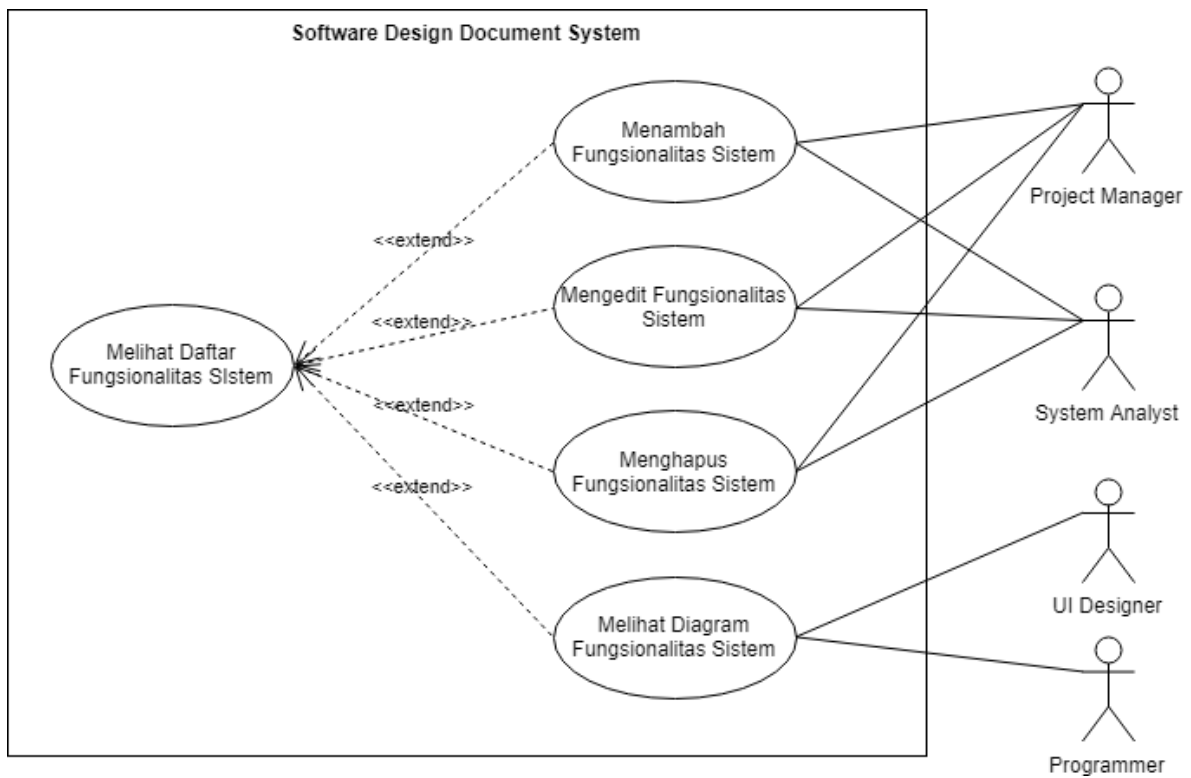
Gambar 3.2 Use case diagram Sub Package: Use case melihat detail project



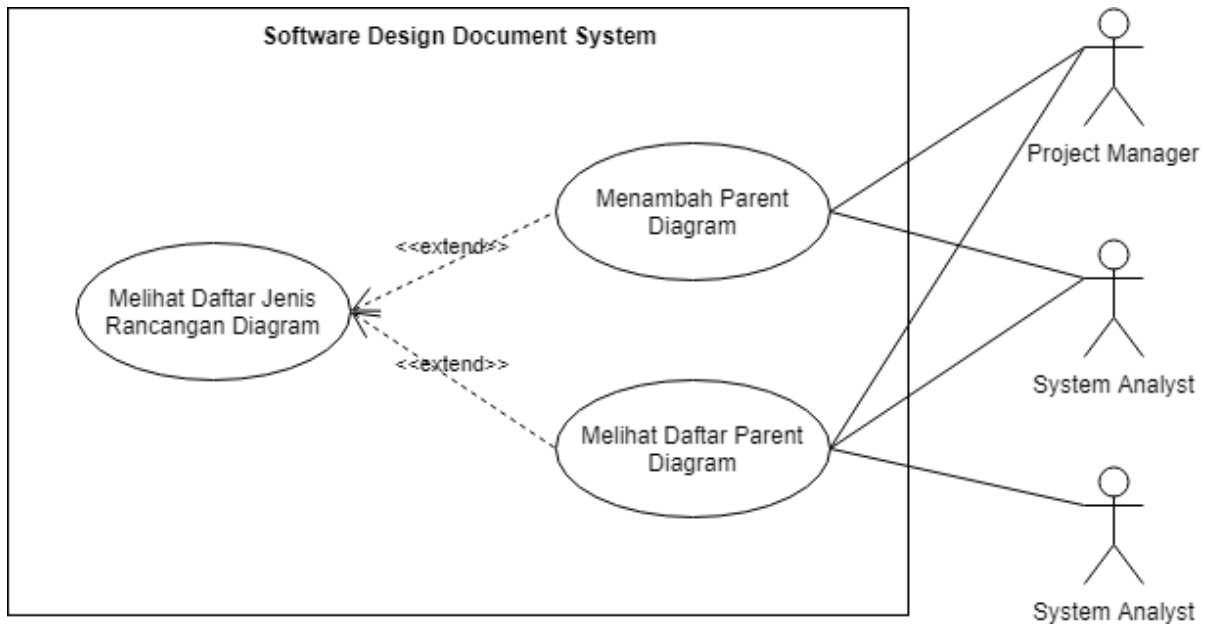
Gambar 3.3 Use case diagram Sub Package: Use case melihat detail versi



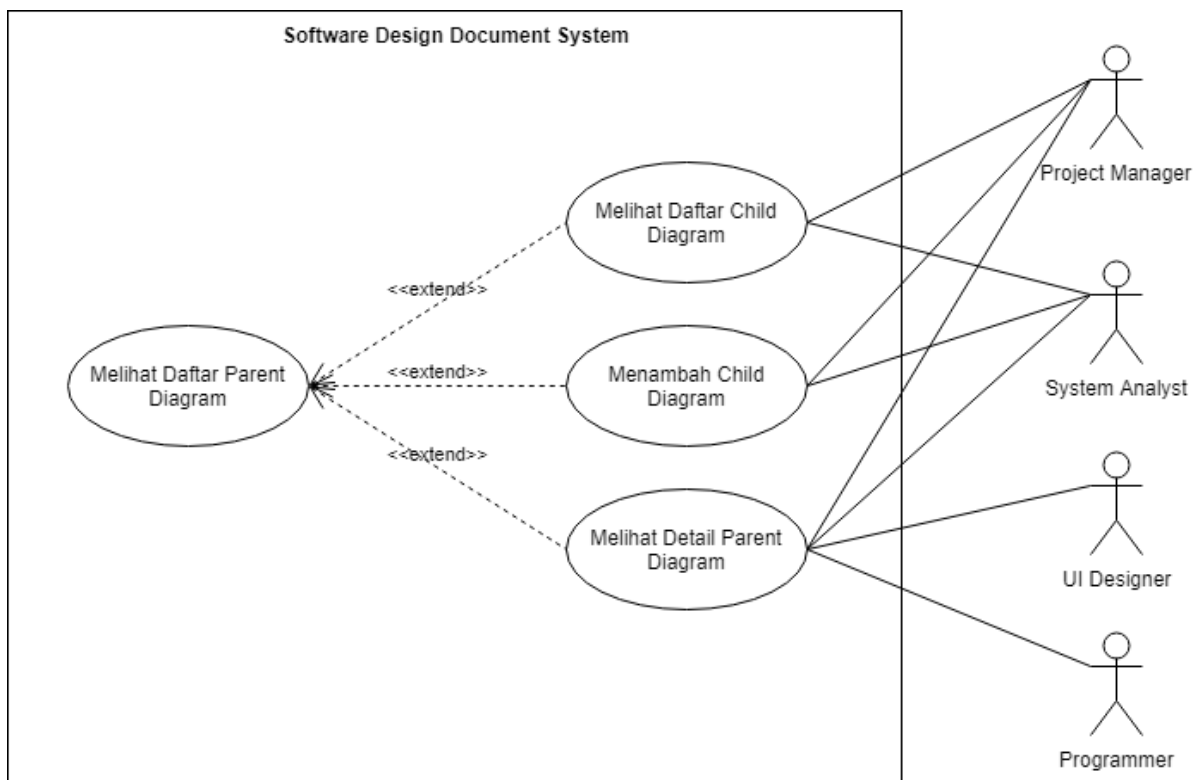
Gambar 3.4 Use case diagram Sub Package: Use case melihat pengantar



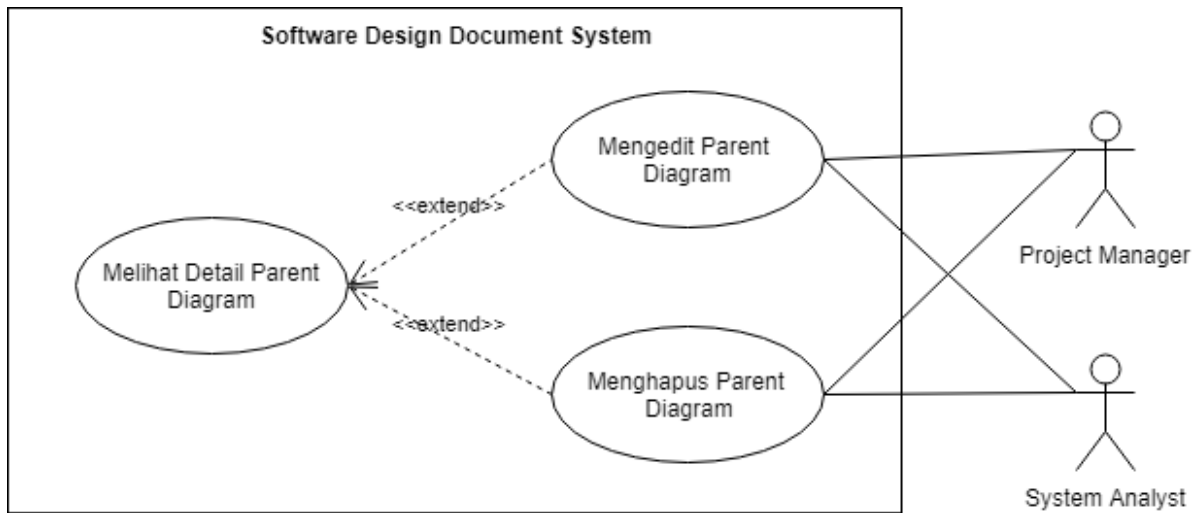
Gambar 3.5 Use case diagram Sub Package: Use case melihat daftar fungsionalitas sistem



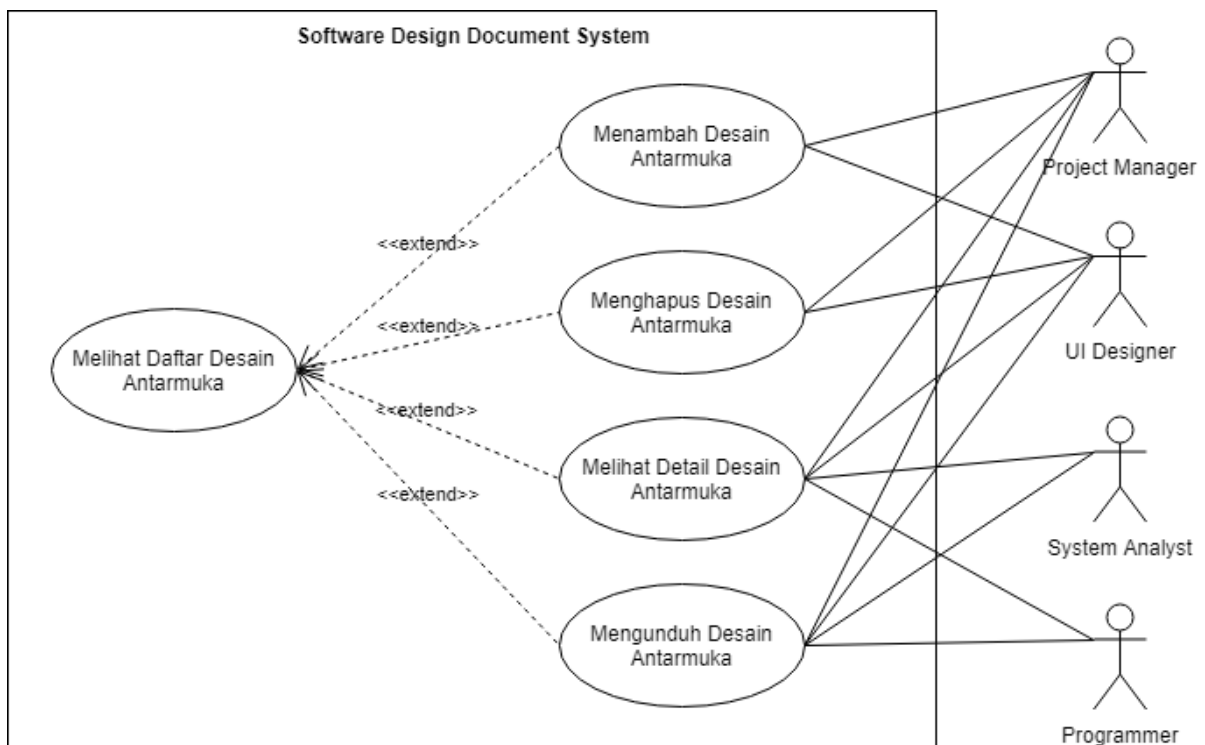
Gambar 3.6 Use case diagram Sub Package: Use case melihat jenis rancangan diagram



Gambar 3.7 Use case diagram Sub Package: Use case melihat daftar parent diagram



Gambar 3.8 Use case diagram Sub Package: Use case melihat detail parent diagram



Gambar 3.9 Use case diagram Sub Package: Use case melihat daftar desain antarmuka

Berikut penjelasan dari setiap use case yang terdapat pada use case diagram diatas.

Tabel 3.1 Deskripsi setiap use case

No	Use Case	Deskripsi
UC01	Login	Semua user yang akan menggunakan sistem wajib melakukan login terlebih dahulu

No	Use Case	Deskripsi
UC02	Melihat Daftar <i>Project</i>	<i>User</i> dapat melihat daftar <i>project</i> setelah melakukan <i>login</i>
UC03	Menambah <i>Project</i>	Semua <i>user</i> dapat menambah <i>project</i> baru dan <i>user</i> yang menambahkan <i>project</i> akan otomatis memiliki <i>role</i> sebagai <i>Project Manager</i>
UC04	Melihat Detail <i>Project</i>	<i>User</i> dapat melihat detail <i>project</i> dengan mengklik nama <i>project</i> pada daftar <i>project</i>
UC05	Mengedit Detail <i>Project</i>	<i>User</i> dengan <i>role Project Manager</i> dapat mengedit detail <i>project</i>
UC06	Menghapus <i>Project</i>	<i>User</i> dapat menghapus <i>project</i> yang telah dibuat sebelumnya. Apabila <i>project</i> dihapus maka seluruh data maupun gambar yang ada didalamnya akan ikut hilang
UC07	Melihat Anggota	<i>User</i> dapat melihat anggota tim yang tergabung dalam suatu <i>project</i>
UC08	Menambah Anggota	<i>User</i> dengan <i>role Project Manager</i> dapat menambahkan anggota baru
UC09	Mengedit Anggota	<i>User</i> dengan <i>role Project Manager</i> dapat mengedit <i>role</i> anggota
UC10	Menghapus Anggota	<i>User</i> dengan <i>role Project Manager</i> dapat menghapus anggota suatu <i>project</i>
UC11	Melihat Daftar Versi	Semua <i>role user</i> dapat melihat daftar versi pada sebuah <i>project</i>
UC12	Menambah Versi	<i>User</i> dengan <i>role Project Manager</i> dapat menambah versi
UC13	Mengedit Versi	<i>User</i> dengan <i>role Project Manager</i> dapat mengedit versi
UC14	Menghapus Versi	<i>User</i> dengan <i>role Project Manager</i> dapat menghapus versi
UC15	Melihat Detail Versi	Semua <i>role user</i> dapat melihat detail versi pada sebuah <i>project</i>
UC16	Melihat Pengantar	Semua <i>role user</i> dapat melihat informasi pengantar pada sebuah <i>project</i>
UC17	Melihat Daftar Fungsionalitas Sistem	Semua <i>role user</i> dapat melihat daftar fungsionalitas sistem dari sebuah <i>project</i>

No	Use Case	Deskripsi
UC18	Melihat Daftar Jenis Rancangan Diagram	Semua <i>role user</i> dapat melihat daftar jenis rancangan diagram dari sebuah <i>project</i>
UC19	Melihat Daftar Desain Antarmuka	Semua <i>role user</i> dapat melihat daftar desain antarmuka dari sebuah <i>project</i>
UC20	Konversi SDD menjadi PDF	Semua <i>role user</i> dapat mengkonversi SDD menjadi PDF
UC21	Menambah Pengantar	<i>User</i> dengan <i>role Project Manager</i> dan <i>System Analyst</i> dapat menambah pengantar
UC22	Mengedit Pengantar	<i>User</i> dengan <i>role Project Manager</i> dan <i>System Analyst</i> dapat mengedit pengantar
UC23	Menambah Fungsionalitas Sistem	<i>User</i> dengan <i>role Project Manager</i> dan <i>System Analyst</i> dapat menambah fungsionalitas sistem
UC24	Mengedit Fungsionalitas Sistem	<i>User</i> dengan <i>role Project Manager</i> dan <i>System Analyst</i> dapat mengedit fungsionalitas sistem
UC25	Menghapus Fungsionalitas Sistem	<i>User</i> dengan <i>role Project Manager</i> dan <i>System Analyst</i> dapat menghapus fungsionalitas sistem
UC26	Melihat Diagram Fungsionalitas Sistem	Semua <i>role user</i> dapat melihat diagram fungsionalitas sistem
UC27	Menambah <i>Parent</i> Diagram	<i>User</i> dengan <i>role Project Manager</i> dan <i>System Analyst</i> dapat menambah <i>parent</i> diagram
UC28	Melihat Daftar <i>Parent</i> Diagram	Semua <i>role user</i> dapat melihat daftar <i>parent</i> diagram
UC29	Melihat Daftar <i>Child</i> Diagram	Semua <i>role user</i> dapat melihat daftar <i>child</i> diagram
UC30	Menambah <i>Child</i> Diagram	<i>User</i> dengan <i>role Project Manager</i> dan <i>System Analyst</i> dapat menambah <i>child</i> diagram
UC31	Melihat Detail <i>Parent</i> Diagram	Semua <i>role user</i> dapat melihat detail <i>parent</i> diagram
UC32	Mengedit <i>Parent</i> Diagram	<i>User</i> dengan <i>role Project Manager</i> dan <i>System Analyst</i> dapat mengedit <i>parent</i> diagram
UC33	Menghapus <i>Parent</i> Diagram	<i>User</i> dengan <i>role Project Manager</i> dan <i>System Analyst</i> dapat menghapus <i>parent</i> diagram
UC34	Menambah Desain Antarmuka	<i>User</i> dengan <i>role Project Manager</i> dan <i>UI Designer</i> dapat menambah desain antarmuka



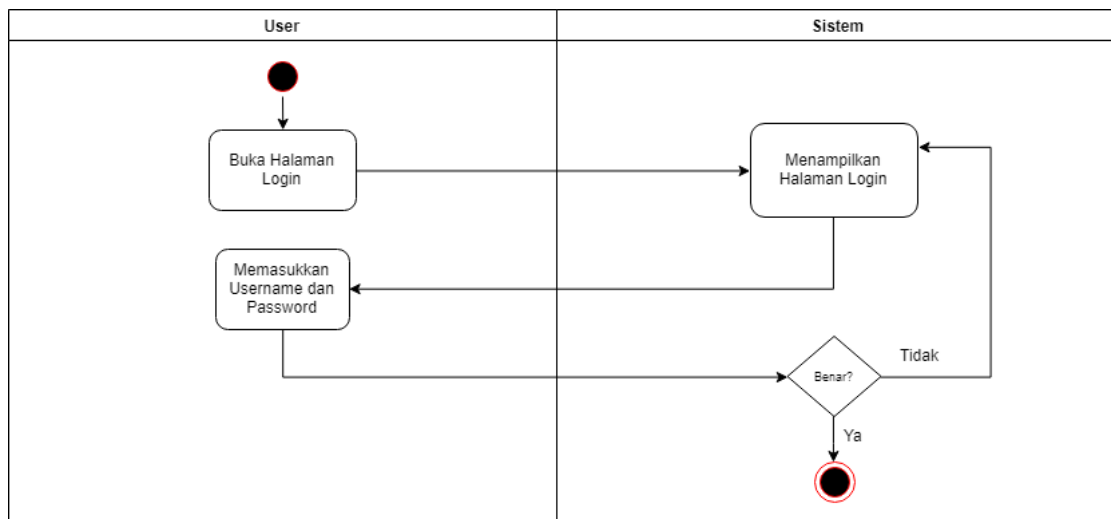
No	Use Case	Deskripsi
UC35	Menghapus Desain Antarmuka	User dengan role Project Manager dan UI Designer dapat menghapus desain antarmuka
UC36	Melihat Detail Desain Antarmuka	Semua role user dapat melihat detail desain antarmuka
UC37	Mengedit Desain Antarmuka	User dengan role Project Manager dan UI Designer dapat mengedit desain antarmuka
UC38	Mengunduh Desain Antarmuka	Semua role user dapat mengunduh desain antarmuka

### 3.3.2 Rancangan Activity Diagram

Activity diagram digunakan oleh penulis untuk mendeskripsikan alur kegiatan antara pengguna dengan sistem yang sebelumnya juga telah dijelaskan pada bagian *use case diagram*. Penulis membagi jumlah *activity diagram* berdasarkan *use case* yang telah dibuat sebelumnya, diantaranya adalah:

#### 6. Activity Diagram Login

Alur kerja sistem pada *use case login* dapat dilihat pada Gambar 3.10



Gambar 3.10 Activity diagram login

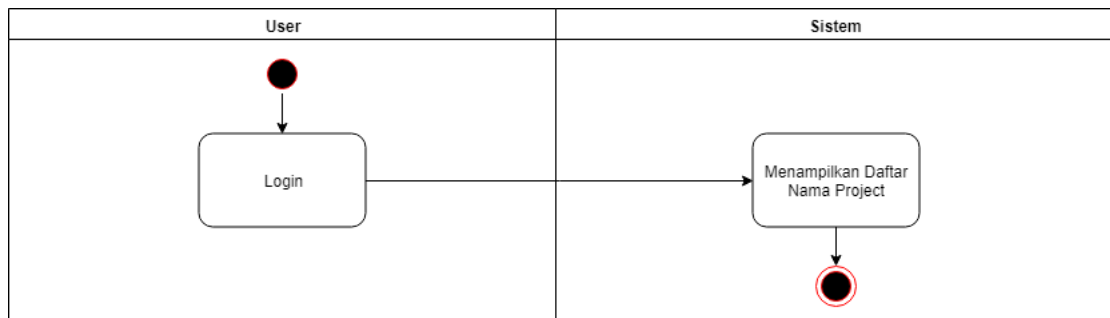
Penjelasan *activity diagram* pada Gambar 3.10 dapat dilihat pada Tabel 3.2

Tabel 3.2 Penjelasan alur kegiatan *login*

No	Aktor	Aktivitas
1	User	User membuka halaman <i>login</i>
2	Sistem	Sistem menampilkan halaman <i>login</i>
3	User	User memasukkan <i>username</i> dan <i>password</i>
4	Sistem	Sistem akan mengecek apakah <i>username</i> dan <i>password</i> yang dimasukkan <i>user</i> sudah sesuai dan tersedia dalam basisdata. Jika benar maka sistem akan menampilkan halaman daftar <i>project</i> dan jika salah maka sistem akan kembali menampilkan halaman <i>login</i>

### 7. Activity Diagram Melihat Daftar Project

Alur kerja sistem pada *use case* melihat daftar *project* dapat dilihat pada Gambar 3.11

Gambar 3.11 Activity diagram melihat daftar *project*

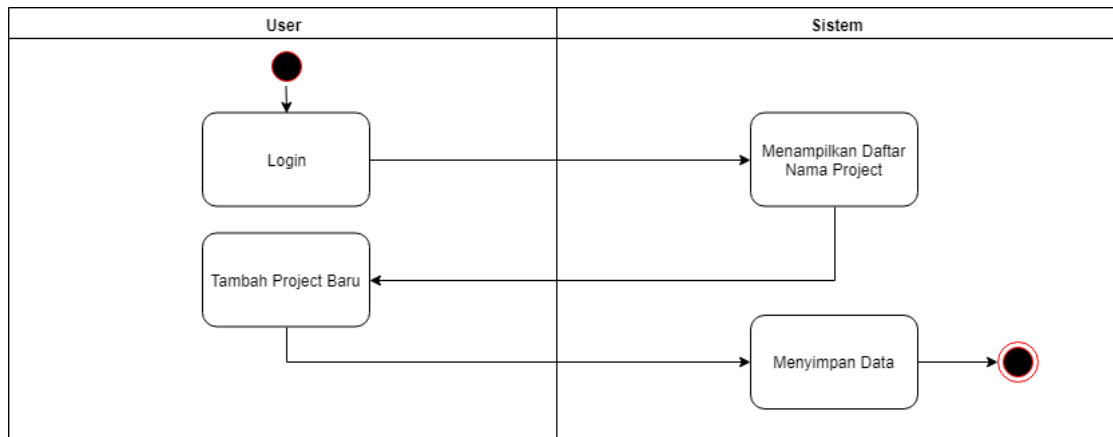
Penjelasan *activity diagram* pada Gambar 3.11 dapat dilihat pada Tabel 3.3

Tabel 3.3 Penjelasan alur kegiatan melihat daftar *project*

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>

### 8. Activity Diagram Menambah Project

Alur kerja sistem pada *use case* menambah *project* dapat dilihat pada Gambar 3.12



Gambar 3.12 Activity diagram menambah *project*

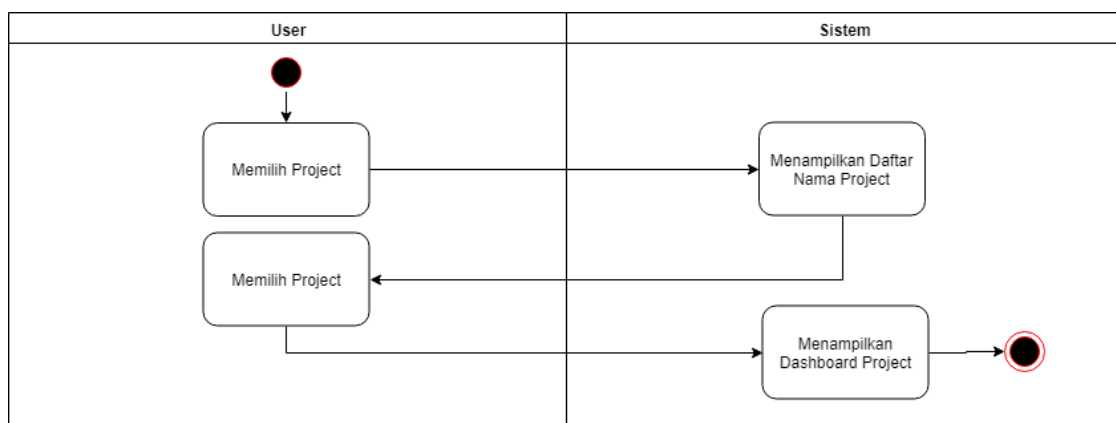
Penjelasan *activity diagram* pada Gambar 3.12 dapat dilihat pada Tabel 3.4

Tabel 3.4 Penjelasan alur kegiatan menambah *project*

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User menambah <i>project</i> baru
4	Sistem	Sistem menyimpan data yang dimasukkan oleh <i>user</i> dan kembali menampilkan halaman daftar nama <i>project</i>

### 9. Activity Diagram Melihat Detail Project

Alur kerja sistem pada *use case* melihat detail *project* dapat dilihat pada Gambar 3.13



Gambar 3.13 Activity diagram melihat detail *project*

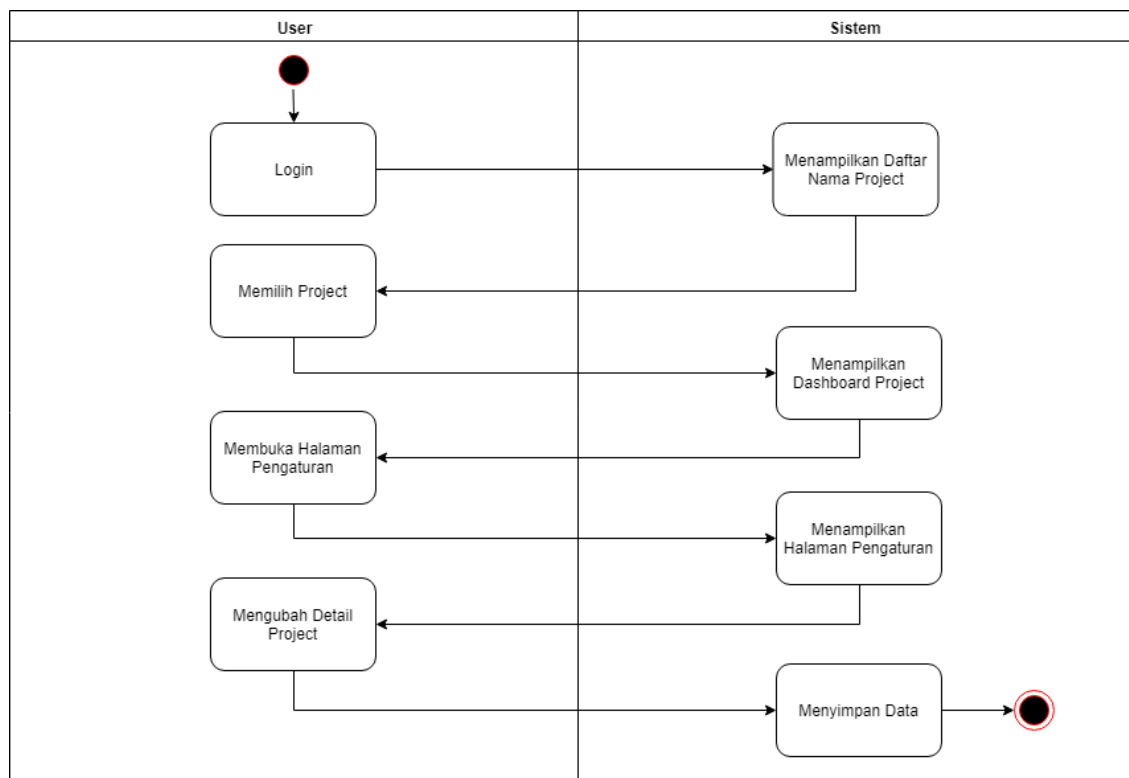
Penjelasan *activity diagram* pada Gambar 3.13 dapat dilihat pada Tabel 3.5

Tabel 3.5 Penjelasan alur kegiatan melihat detail *project*

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>

#### 10. Activity Diagram Mengedit Detail Project

Alur kerja sistem pada *use case* mengedit detail *project* dapat dilihat pada Gambar 3.14

Gambar 3.14 Activity diagram mengedit detail *project*

Penjelasan *activity diagram* pada Gambar 3.14 dapat dilihat pada Tabel 3.6

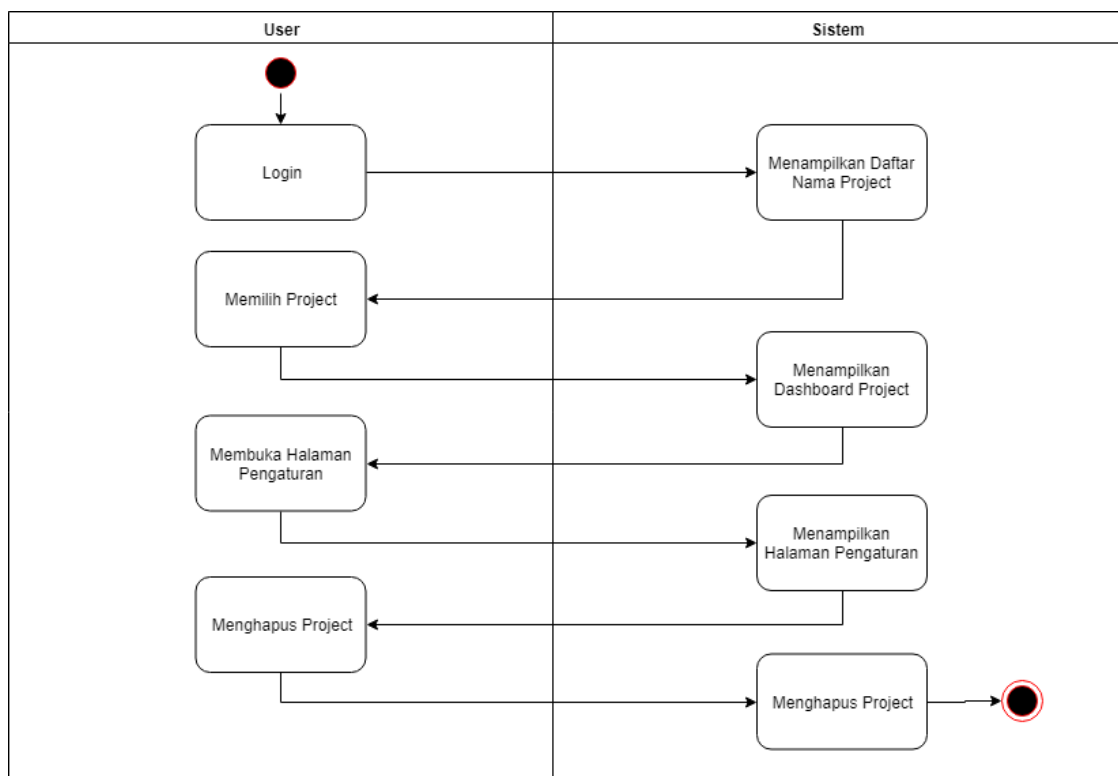
Tabel 3.6 Penjelasan alur kegiatan mengedit detail *project*

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>

No	Aktor	Aktivitas
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka halaman pengaturan
6	Sistem	Sistem menampilkan halaman pengaturan
8	User	User mengubah data <i>project</i>
9	Sistem	Sistem menyimpan data <i>project</i> yang telah diubah dan kembali menampilkan halaman pengaturan

### 11. Activity Diagram Menghapus Project

Alur kerja sistem pada *use case* Menghapus Project dapat dilihat pada Gambar 3.15



Gambar 3.15 Activity diagram menghapus project

Penjelasan *activity diagram* pada Gambar 3.15 dapat dilihat pada Tabel 3.7

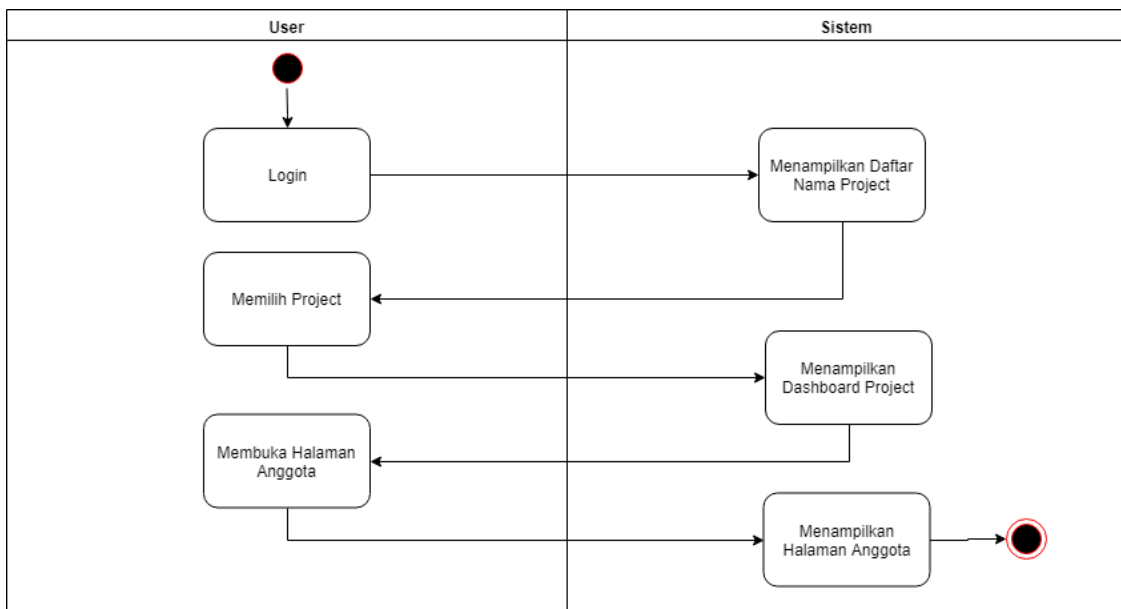
Tabel 3.7 Penjelasan alur kegiatan menghapus *project*

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>

No	Aktor	Aktivitas
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu pengaturan
6	Sistem	Sistem menampilkan halaman pengaturan
7	User	User memilih hapus <i>project</i>
8	Sistem	Sistem menghapus <i>project</i> dan kembali menampilkan halaman daftar nama <i>project</i>

## 12. Activity Diagram Melihat Anggota

Alur kerja sistem pada *use case* Melihat Anggota dapat dilihat pada Gambar 3.16



Gambar 3.16 Activity diagram melihat anggota

Penjelasan *activity diagram* pada Gambar 3.16 dapat dilihat pada Tabel 3.8

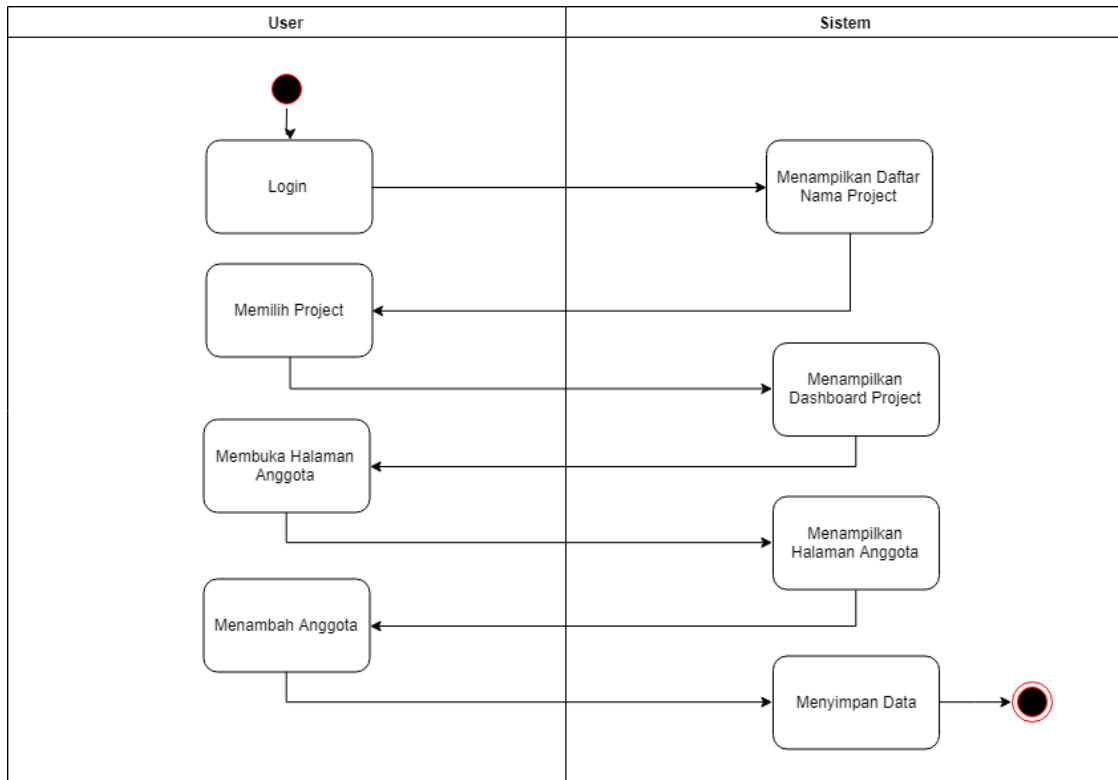
Tabel 3.8 Penjelasan alur kegiatan melihat anggota

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu anggota

6	Sistem	Sistem menampilkan halaman anggota
---	--------	------------------------------------

### 13. Activity Diagram Menambah Anggota

Alur kerja sistem pada *use case* Menambah Anggota dapat dilihat pada Gambar 3.17



Gambar 3.17 Activity diagram menambah anggota

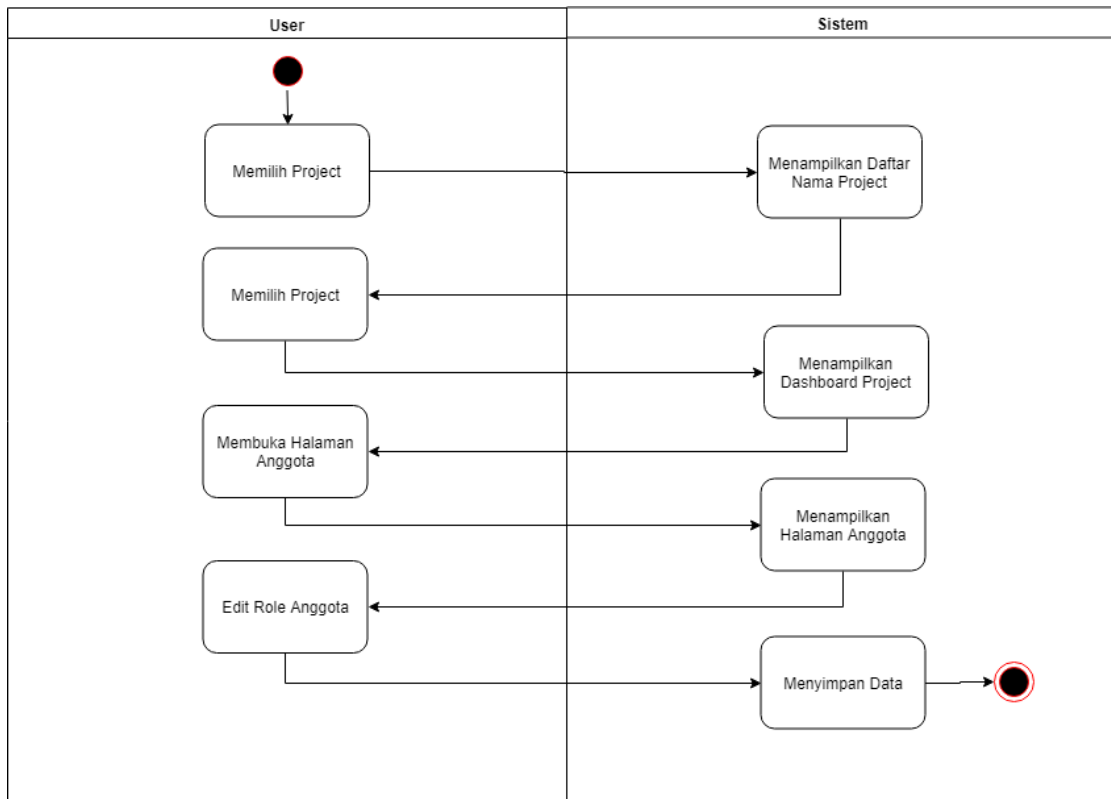
Penjelasan *activity diagram* pada Gambar 3.17 dapat dilihat pada Tabel 3.9

Tabel 3.9 Penjelasan alur kegiatan melihat anggota

No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu anggota
6	Sistem	Sistem menampilkan halaman anggota

### 14. Activity Diagram Mengedit Anggota

Alur kerja sistem pada *use case* Mengedit Anggota dapat dilihat pada Gambar 3.18



Gambar 3.18 Activity diagram mengedit anggota

Penjelasan activity diagram pada Gambar 3.18 dapat dilihat pada Tabel 3.10

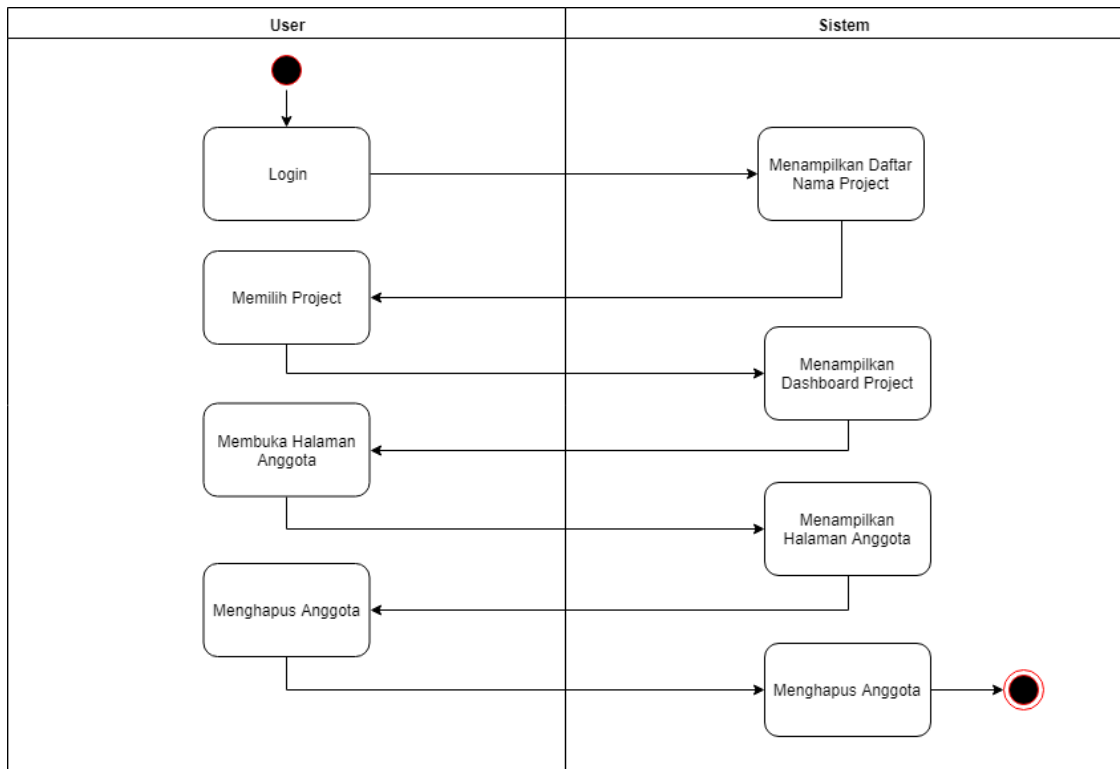
Tabel 3.10 Penjelasan alur kegiatan mengedit anggota

No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu anggota
6	Sistem	Sistem menampilkan halaman anggota
7	User	User memilih edit anggota yang ingin diedit <i>role</i> -nya
8	Sistem	Sistem menampilkan form edit anggota
9	User	User mengubah role sesuai yang diinginkan
10	Sistem	Sistem menyimpan data dan kembali menampilkan halaman anggota

## 15. Activity Diagram Menghapus Anggota



Alur kerja sistem pada *use case* Menghapus Anggota dapat dilihat pada Gambar 3.19



Gambar 3.19 *Activity diagram* menghapus anggota

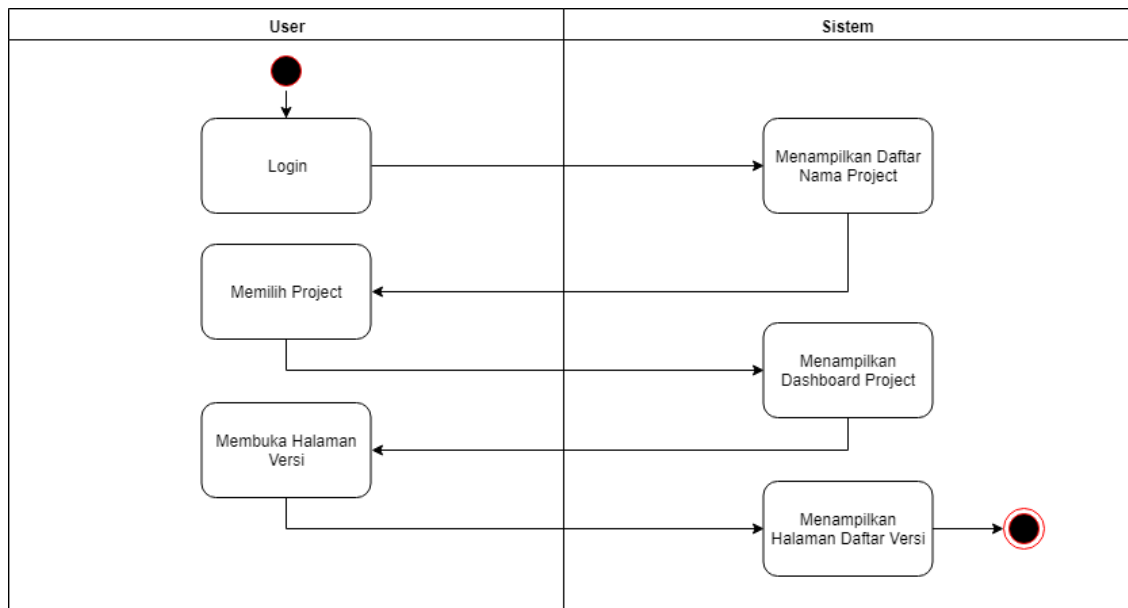
Penjelasan *activity diagram* pada Gambar 3.19 dapat dilihat pada Tabel 3.11

Tabel 3.11 Penjelasan alur kegiatan menghapus anggota

No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu anggota
6	Sistem	Sistem menampilkan halaman anggota
7	User	User memilih hapus anggota yang ingin dihapus dari daftar anggota <i>project</i>
8	Sistem	Sistem menghapus data dan kembali menampilkan halaman anggota

#### 16. *Activity Diagram* Melihat Daftar Versi

Alur kerja sistem pada *use case* Melihat Daftar Versi dapat dilihat pada Gambar 3.20



Gambar 3.20 Activity diagram melihat daftar versi

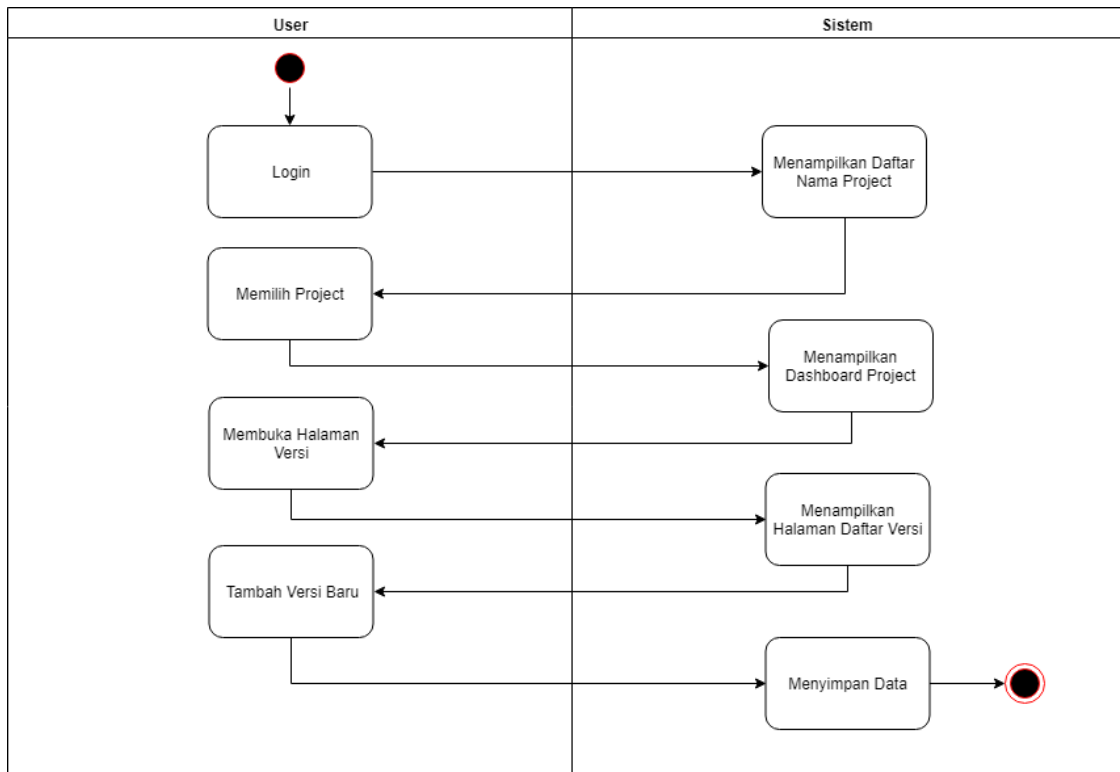
Penjelasan *activity diagram* pada Gambar 3.20 dapat dilihat pada Tabel 3.12

Tabel 3.12 Penjelasan alur kegiatan melihat daftar versi

No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi

## 17. Activity Diagram Menambah Versi

Alur kerja sistem pada *use case* Menambah Versi dapat dilihat pada Gambar 3.21



Gambar 3.21 *Activity diagram* menambah versi

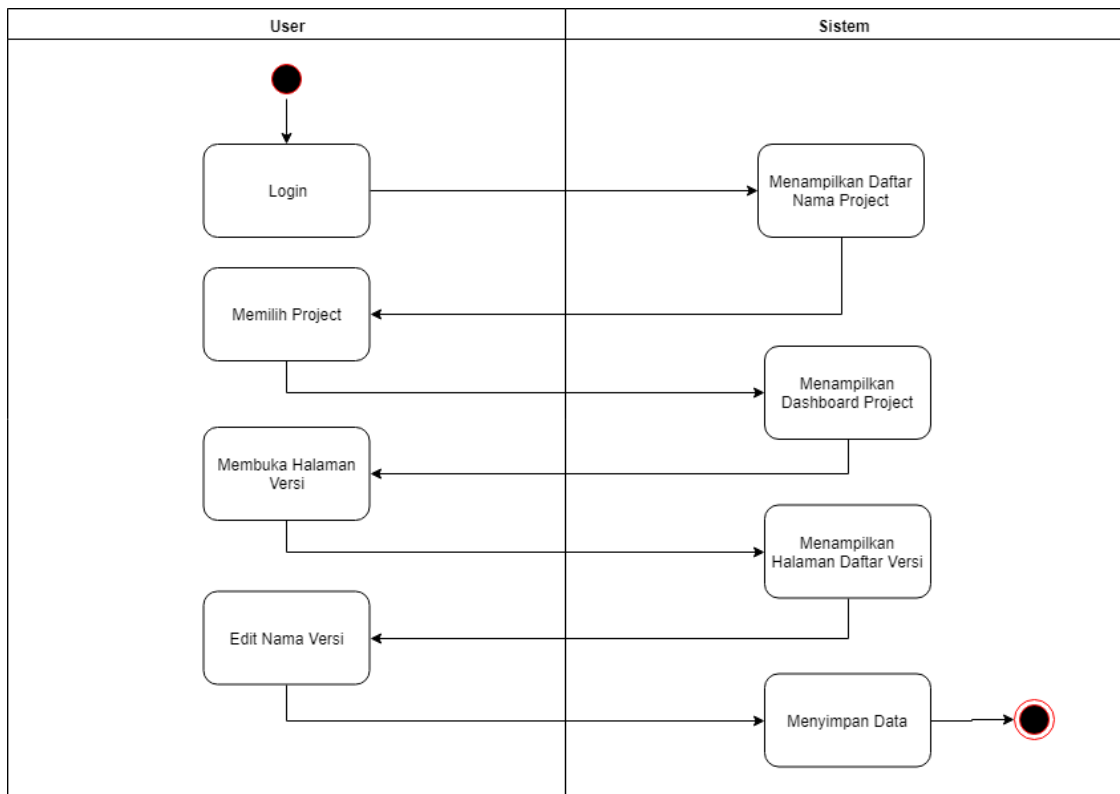
Penjelasan *activity diagram* pada Gambar 3.21 dapat dilihat pada Tabel 3.13

Tabel 3.13 Penjelasan alur kegiatan menambah versi

No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih tambah versi
8	Sistem	Sistem menampilkan form tambah versi baru
9	User	User mengisi form tambah versi baru
10	Sistem	Sistem menyimpan data dan kembali menampilkan halaman daftar versi

#### 18. *Activity Diagram* Mengedit Versi

Alur kerja sistem pada *use case* Mengedit Versi dapat dilihat pada Gambar 3.22



Gambar 3.22 Activity diagram mengedit versi

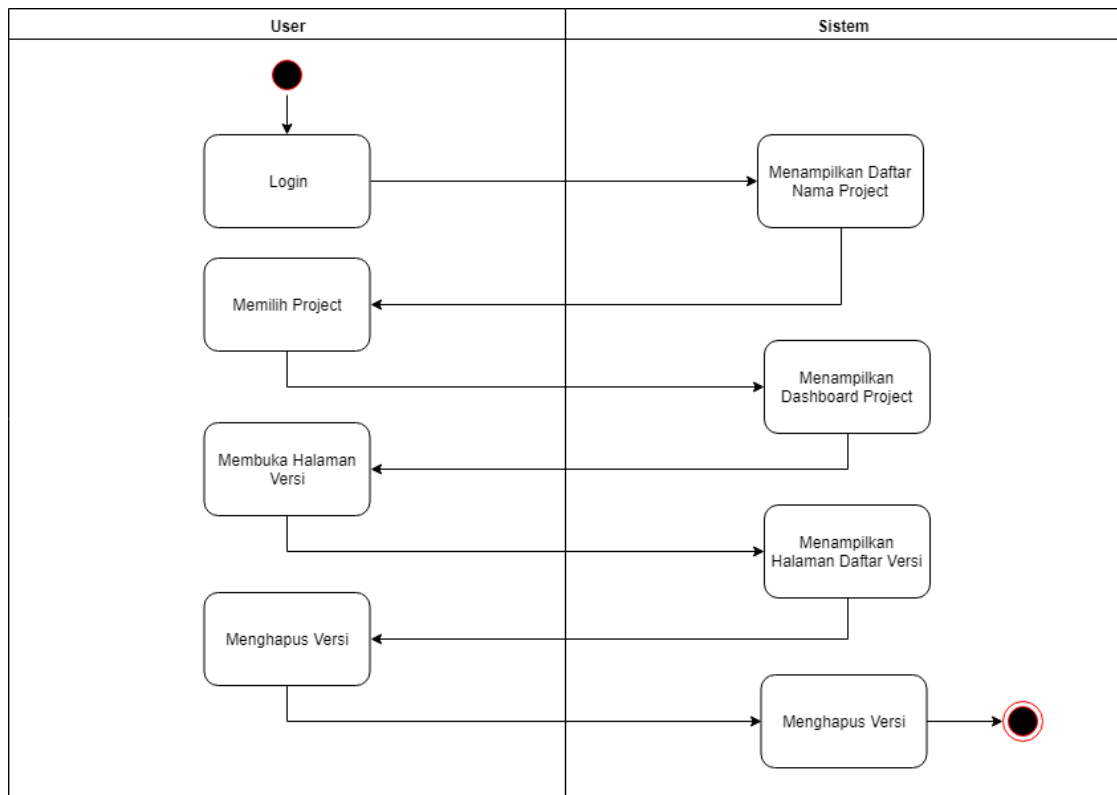
Penjelasan *activity diagram* pada Gambar 3.22 dapat dilihat pada Tabel 3.14

Tabel 3.14 Penjelasan alur kegiatan mengedit versi

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi yang ingin diedit
8	Sistem	Sistem menampilkan form edit versi
9	User	User mengubah data versi yang ingin diubah
10	Sistem	Sistem menyimpan data dan kembali menampilkan halaman daftar versi

### 19. Activity Diagram Menghapus Versi

Alur kerja sistem pada *use case* Menghapus Versi dapat dilihat pada Gambar 3.23



Gambar 3.23 Activity diagram menghapus versi

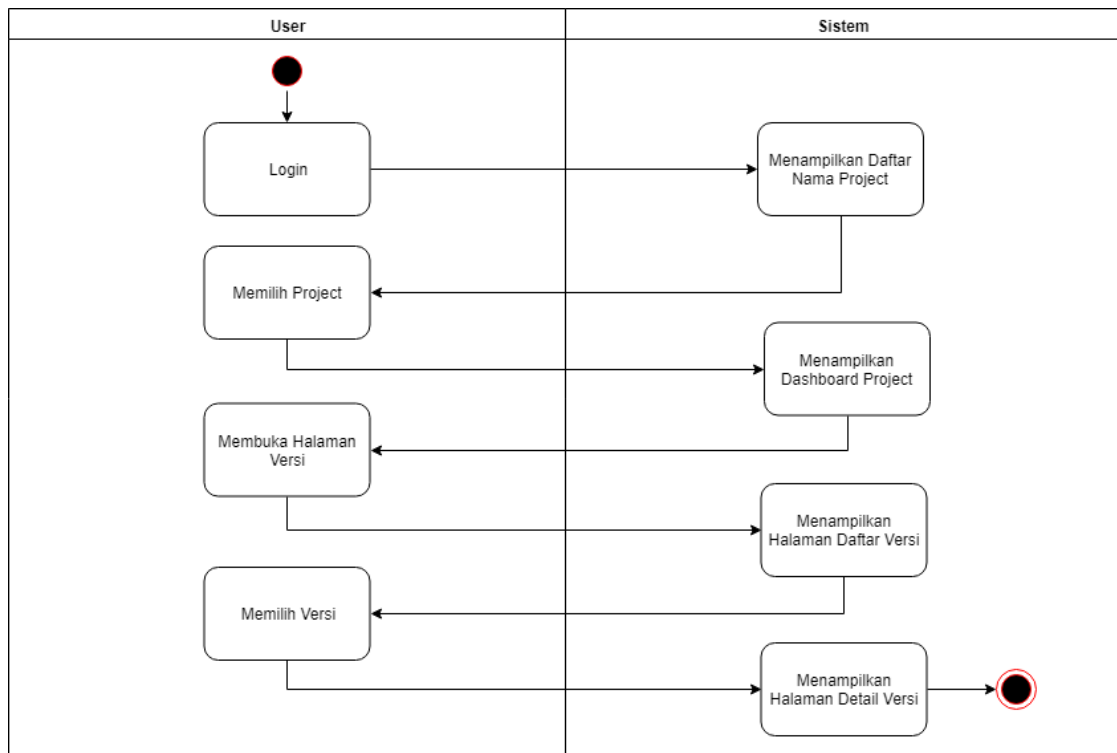
Penjelasan *activity diagram* pada Gambar 3.23 dapat dilihat pada Tabel 3.15

Tabel 3.15 Penjelasan alur kegiatan menghapus versi

No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi yang ingin dihapus
8	Sistem	Sistem menghapus data dan kembali menampilkan halaman daftar versi

## 20. Activity Diagram Melihat Detail Versi

Alur kerja sistem pada *use case* Melihat Detail Versi dapat dilihat pada Gambar 3.24



Gambar 3.24 Activity diagram melihat detail versi

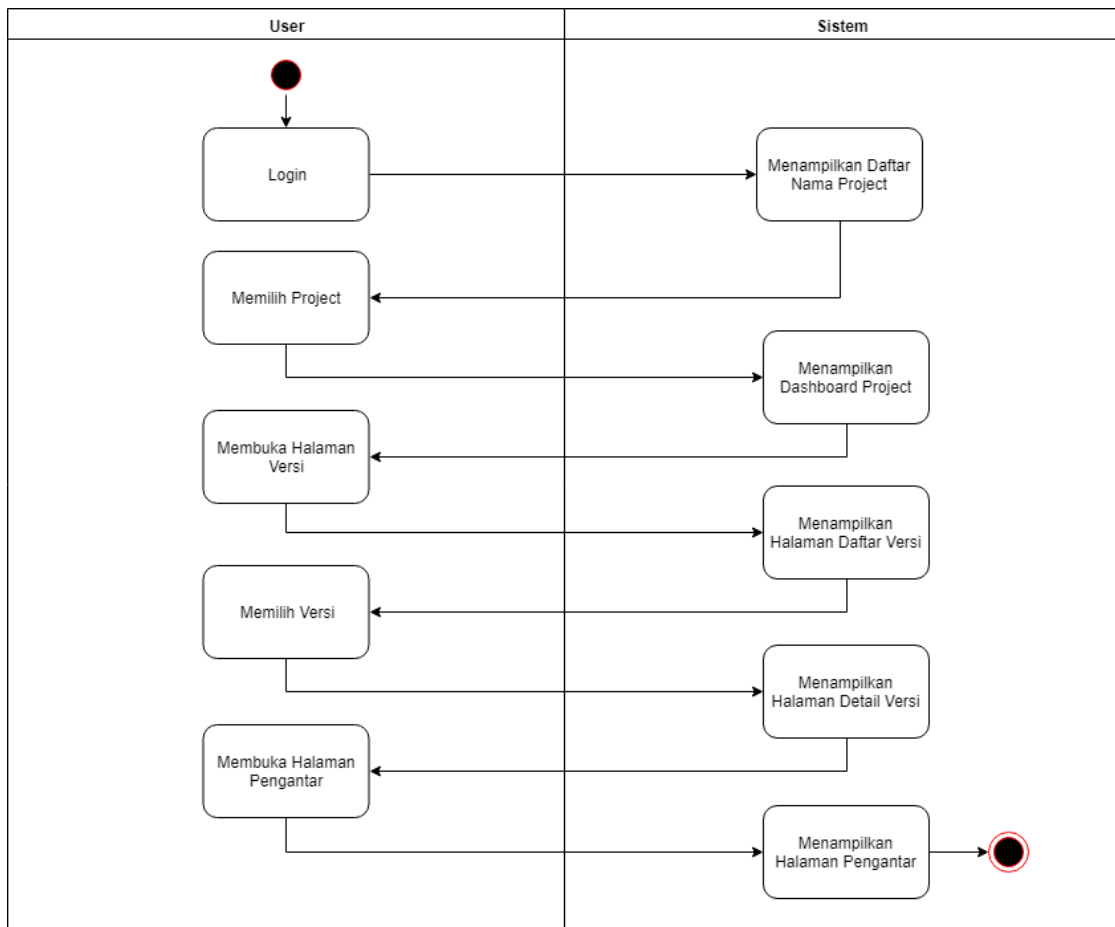
Penjelasan *activity diagram* pada Gambar 3.24 dapat dilihat pada Tabel 3.16

Tabel 3.16 Penjelasan alur kegiatan melihat detail versi

No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi

## 21. Activity Diagram Melihat Pengantar

Alur kerja sistem pada *use case* Melihat Pengantar dapat dilihat pada Gambar 3.25



Gambar 3.25 Activity diagram melihat pengantar

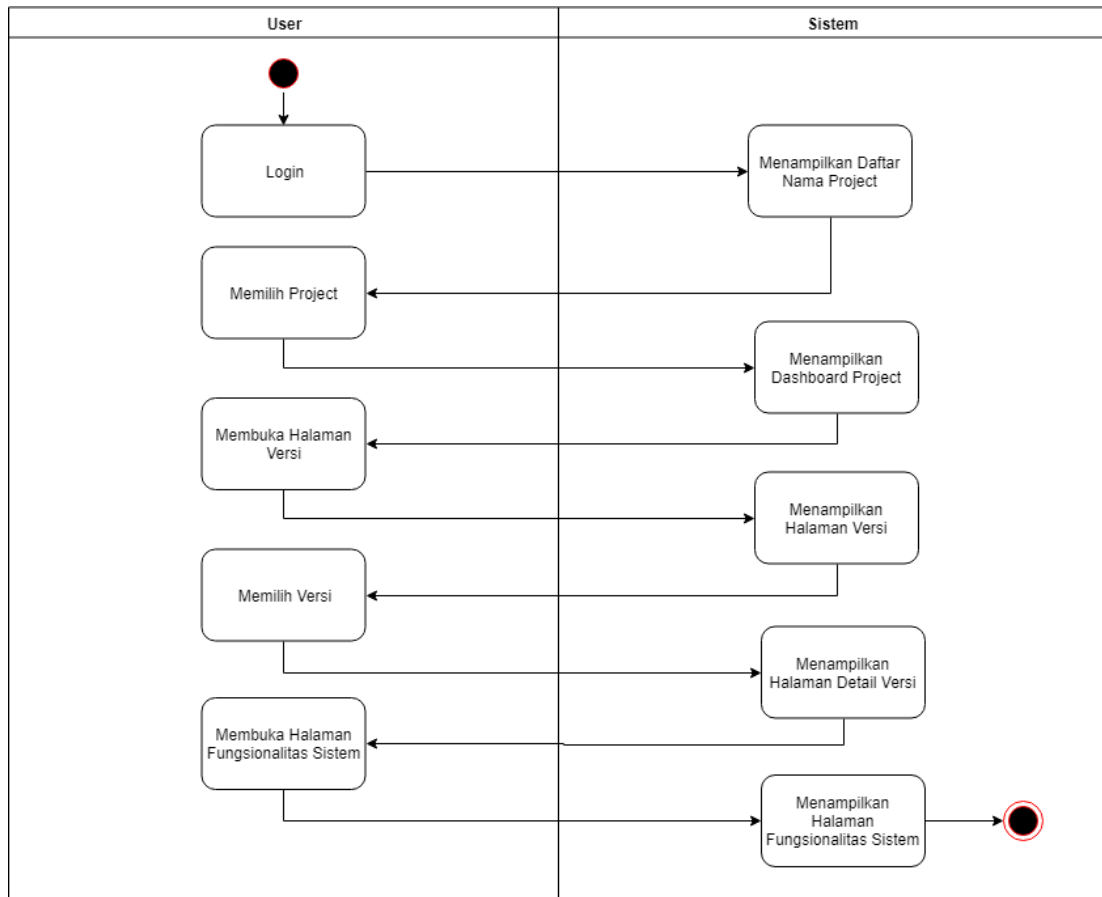
Penjelasan *activity diagram* pada Gambar 3.25 dapat dilihat pada Tabel 3.17

Tabel 3.17 Penjelasan alur kegiatan melihat pengantar

No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu pengantar
10	Sistem	Sistem menampilkan halaman pengantar

## 22. Activity Diagram Melihat Daftar Fungsionalitas Sistem

Alur kerja sistem pada *use case* Melihat Daftar Fungsionalitas Sistem dapat dilihat pada Gambar 3.26



Gambar 3.26 Activity diagram melihat daftar fungsionalitas sistem

Penjelasan *activity diagram* pada Gambar 3.26 dapat dilihat pada Tabel 3.18

Tabel 3.18 Penjelasan alur kegiatan melihat daftar fungsionalitas sistem

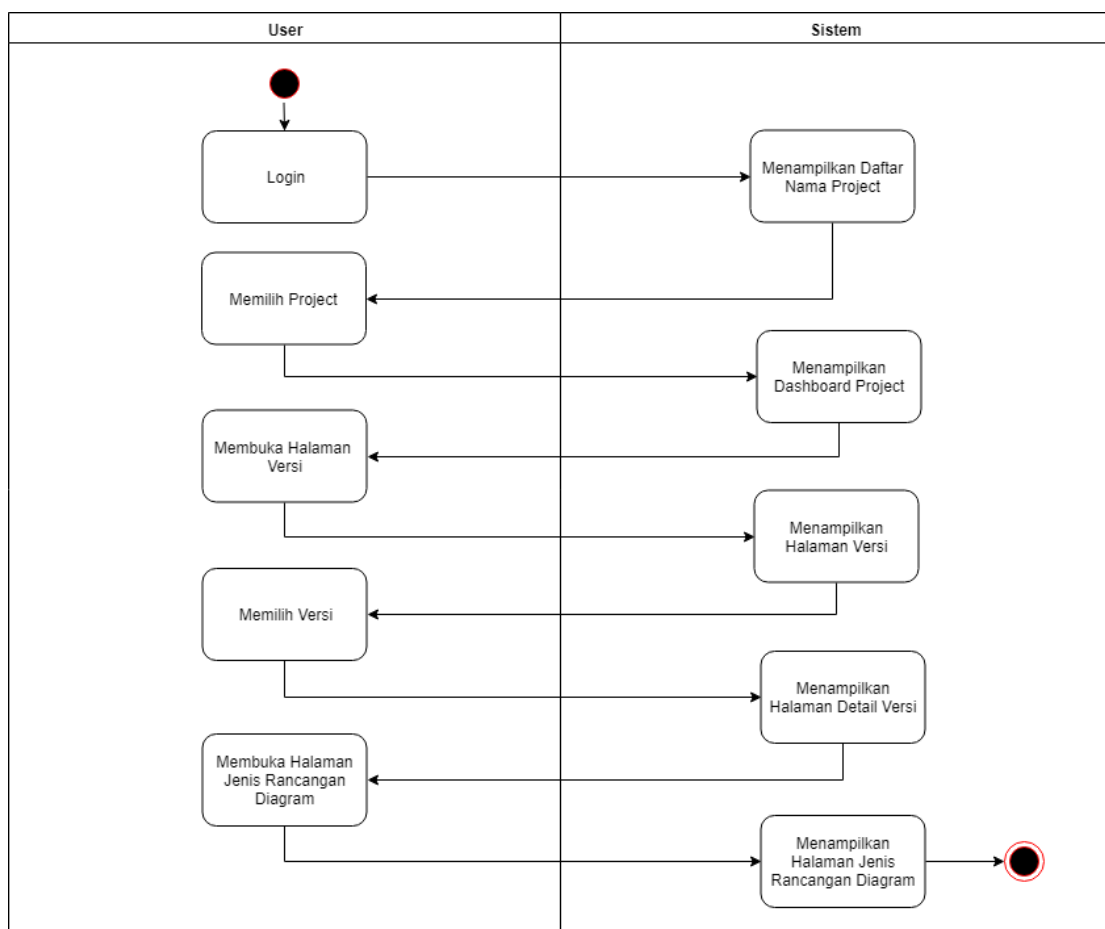
No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi



No	Aktor	Aktivitas
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu fungsionalitas sistem
10	Sistem	Sistem menampilkan halaman fungsionalitas sistem

### 23. Activity Diagram Melihat Daftar Jenis Rancangan Diagram

Alur kerja sistem pada *use case* Melihat Daftar Jenis Rancangan Diagram dapat dilihat pada Gambar 3.27



Gambar 3.27 Activity diagram melihat daftar jenis rancangan diagram

Penjelasan *activity diagram* pada Gambar 3.27 dapat dilihat pada Tabel 3.19

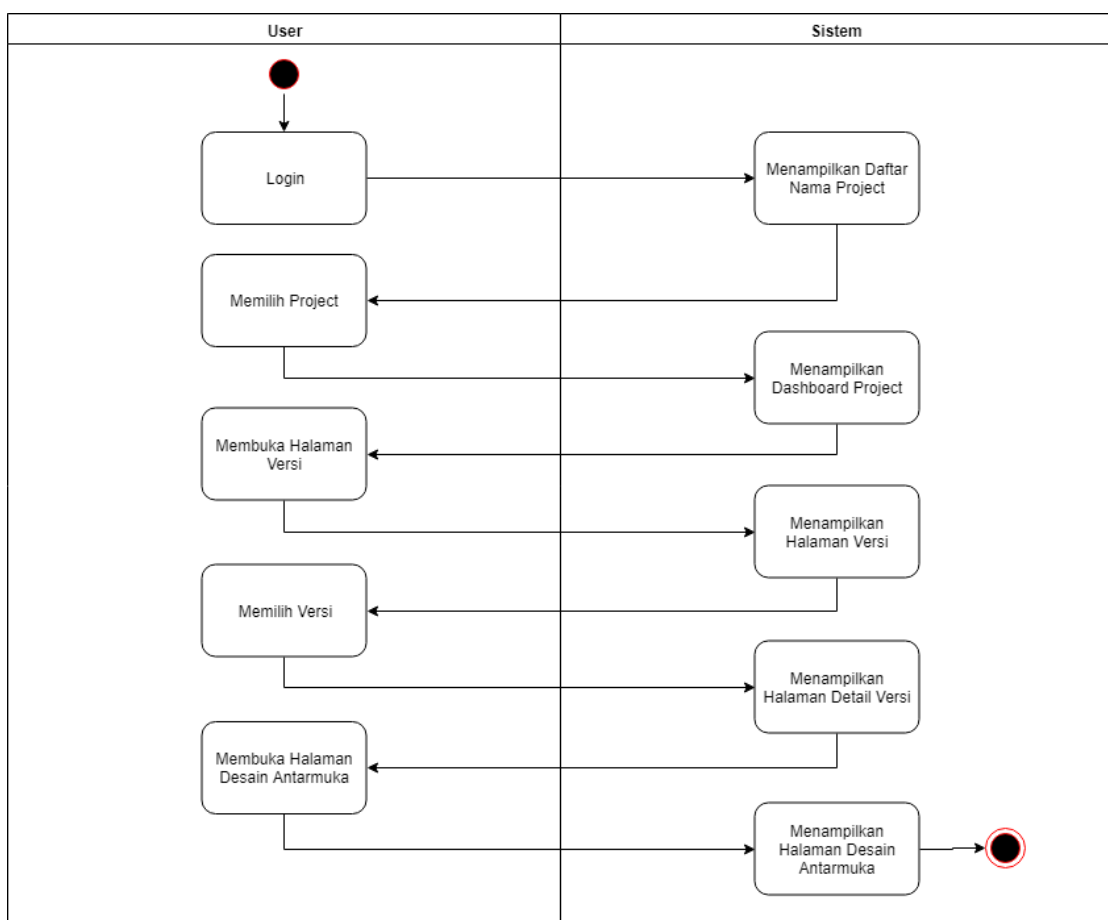
Tabel 3.19 Penjelasan alur kegiatan melihat daftar jenis rancangan diagram

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>

2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu rancangan diagram
10	Sistem	Sistem menampilkan halaman rancangan diagram

#### 24. Activity Diagram Melihat Daftar Desain Antarmuka

Alur kerja sistem pada *use case* Melihat Daftar Desain Antarmuka dapat dilihat pada Gambar 3.28



Gambar 3.28 Activity diagram melihat daftar desain antarmuka

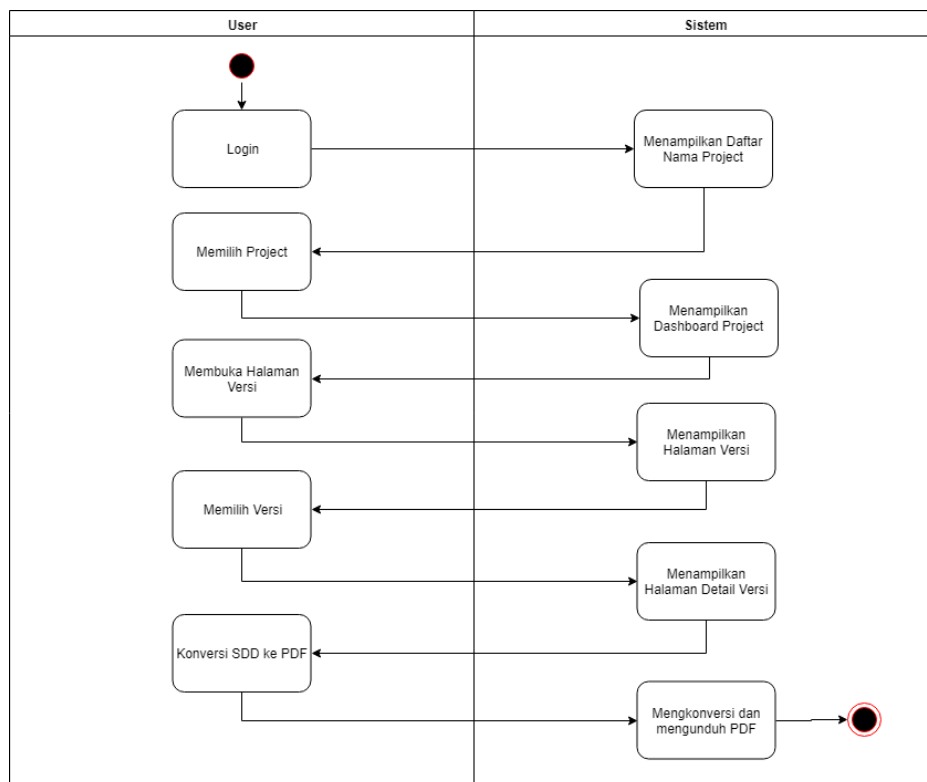
Penjelasan activity diagram pada Gambar 3.28 dapat dilihat pada Tabel 3.20

Tabel 3.20 Penjelasan alur kegiatan melihat daftar desain antarmuka

No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu desain antarmuka
10	Sistem	Sistem menampilkan halaman desain antarmuka

## 25. Activity Diagram Konversi SDD menjadi PDF

Alur kerja sistem pada *use case* Konversi SDD menjadi PDF dapat dilihat pada Gambar 3.29



Gambar 3.29 Activity diagram konversi SDD menjadi PDF

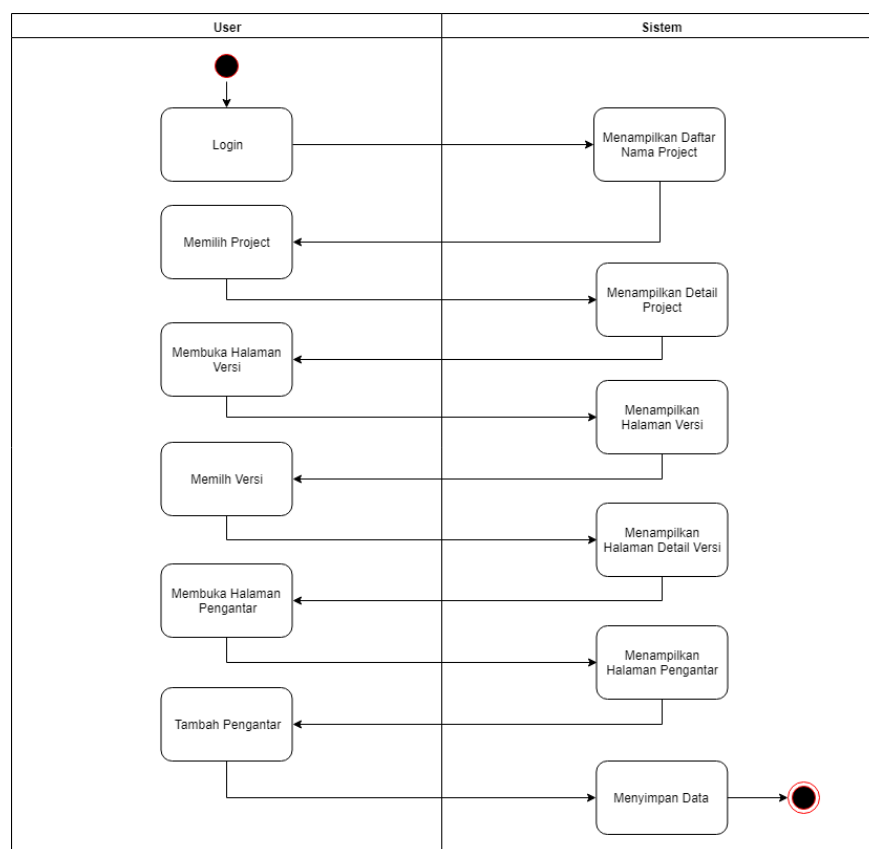
Penjelasan *activity diagram* pada Gambar 3.29 dapat dilihat pada Tabel 3.21

Tabel 3.21 Penjelasan alur kegiatan konversi SDD menjadi PDF

No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User memilih konversi SDD menjadi PDF
10	Sistem	Sistem mengkonversi dan mengunduh SDD menjadi PDF

## 26. Activity Diagram Menambah Pengantar

Alur kerja sistem pada *use case* Menambah Pengantar dapat dilihat pada Gambar 3.30



Gambar 3.30 Activity diagram menambah pengantar

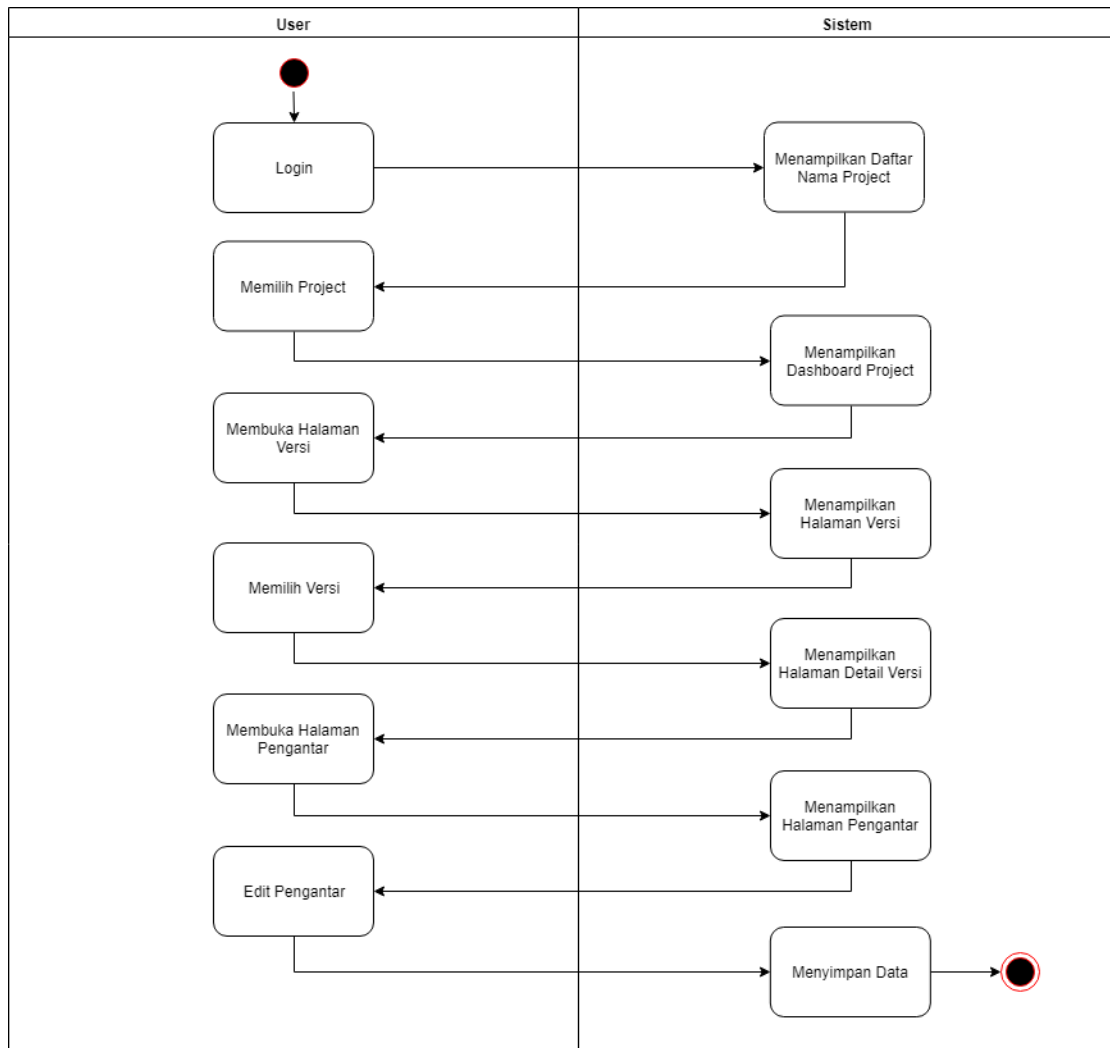
Penjelasan *activity diagram* pada Gambar 3.30 dapat dilihat pada Tabel 3.22

Tabel 3.22 Penjelasan alur kegiatan menambah pengantar

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu pengantar
10	Sistem	Sistem menampilkan halaman pengantar
11	User	User memilih tambah pengantar. Pilihan tambah pengantar hanya muncul jika belum ada data pengantar yang dimasukkan sebelumnya
12	Sistem	Sistem menampilkan form tambah pengantar
13	User	User memasukkan data pengantar
14	Sistem	Sistem menyimpan data dan kembali menampilkan halaman pengantar

## 27. Activity Diagram Mengedit Pengantar

Alur kerja sistem pada *use case* Mengedit Pengantar dapat dilihat pada Gambar 3.31



Gambar 3.31 Activity diagram mengedit pengantar

Penjelasan *activity diagram* pada Gambar 3.31 dapat dilihat pada Tabel 3.23

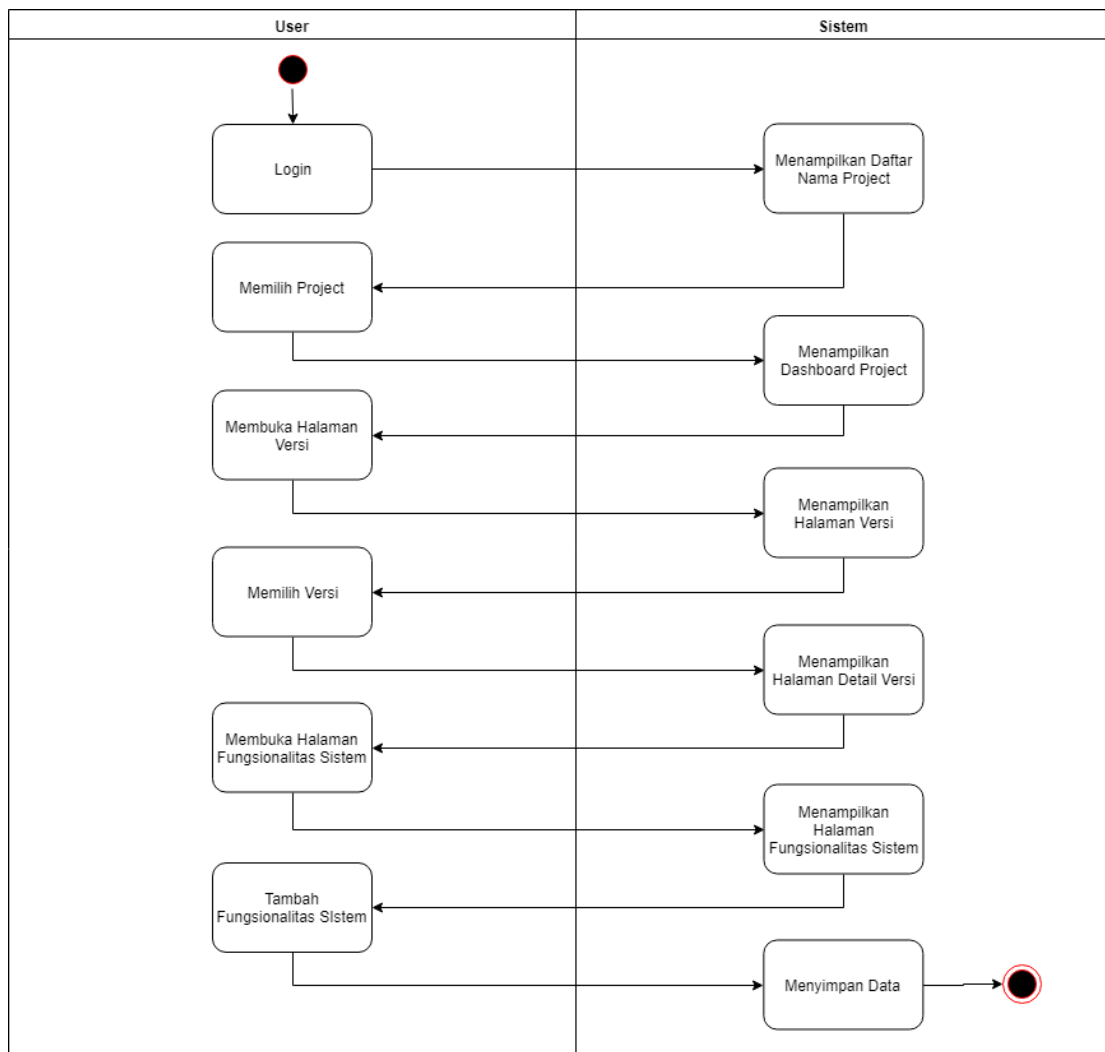
Tabel 3.23 Penjelasan alur kegiatan mengedit pengantar

No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu pengantar

No	Aktor	Aktivitas
10	Sistem	Sistem menampilkan halaman pengantar
11	User	User memilih edit pengantar. Pilihan edit pengantar hanya muncul jika sudah ada data pengantar yang dimasukkan sebelumnya
12	Sistem	Sistem menampilkan form tambah pengantar
13	User	User mengubah data pengantar
14	Sistem	Sistem menyimpan data dan kembali menampilkan halaman pengantar

## 28. Activity Diagram Menambah Fungsionalitas Sistem

Alur kerja sistem pada *use case* Menambah Fungsionalitas Sistem dapat dilihat pada Gambar 3.32



Gambar 3.32 Activity diagram menambah fungsionalitas sistem

Penjelasan *activity diagram* pada Gambar 3.32 dapat dilihat pada Tabel 3.24

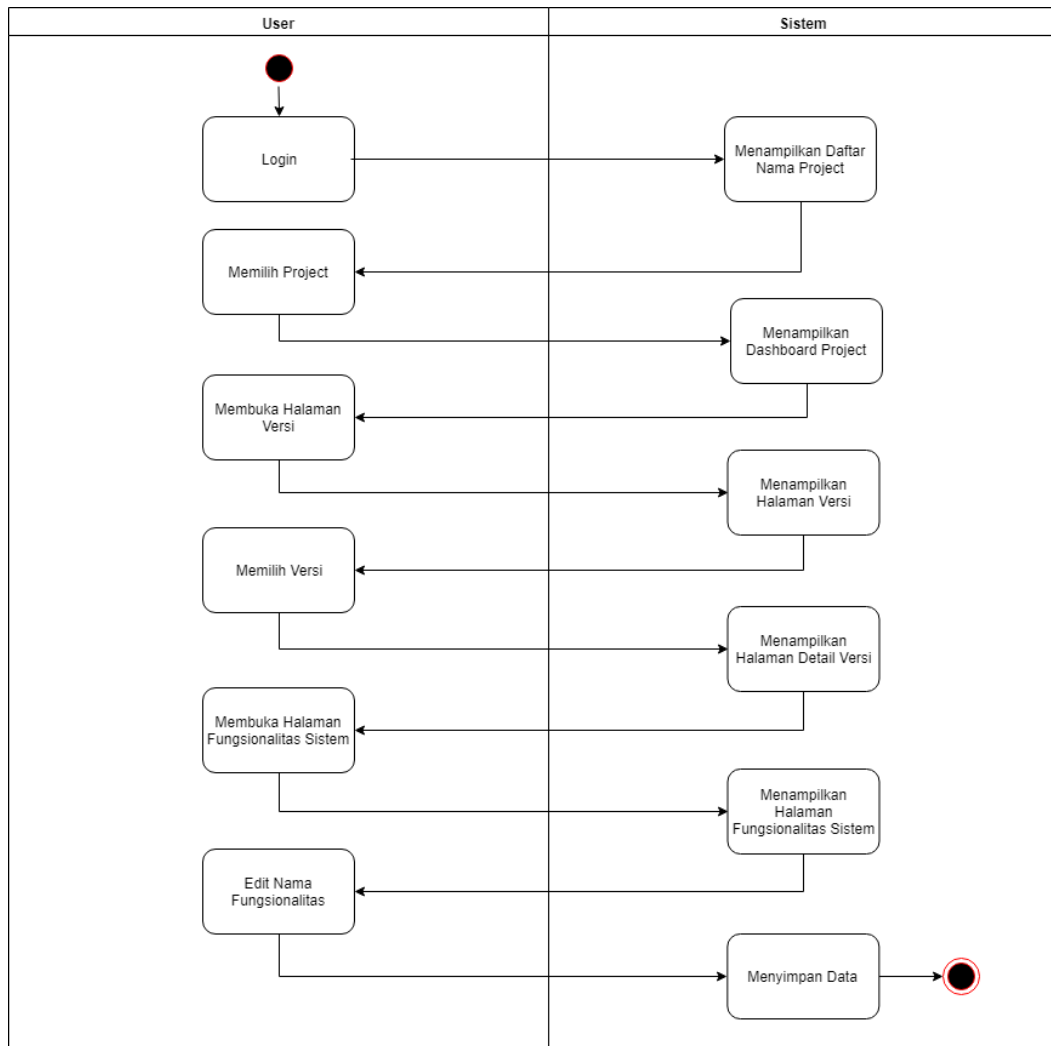
Tabel 3.24 Penjelasan alur kegiatan menambah fungsionalitas sistem

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu fungsionalitas sistem
10	Sistem	Sistem menampilkan halaman fungsionalitas sistem
11	User	User memilih tambah fungsionalitas sistem
12	Sistem	Sistem menampilkan form tambah fungsionalitas sistem
13	User	User menambah data fungsionalitas sistem
14	Sistem	Sistem menyimpan data dan kembali menampilkan halaman fungsionalitas sistem

## 29. *Activity Diagram* Mengedit Fungsionalitas Sistem

Alur kerja sistem pada *use case* Mengedit Fungsionalitas Sistem dapat dilihat pada Gambar 3.33





Gambar 3.33 Activity diagram mengedit fungsionalitas sistem

Penjelasan activity diagram pada Gambar 3.33 dapat dilihat pada Tabel 3.25

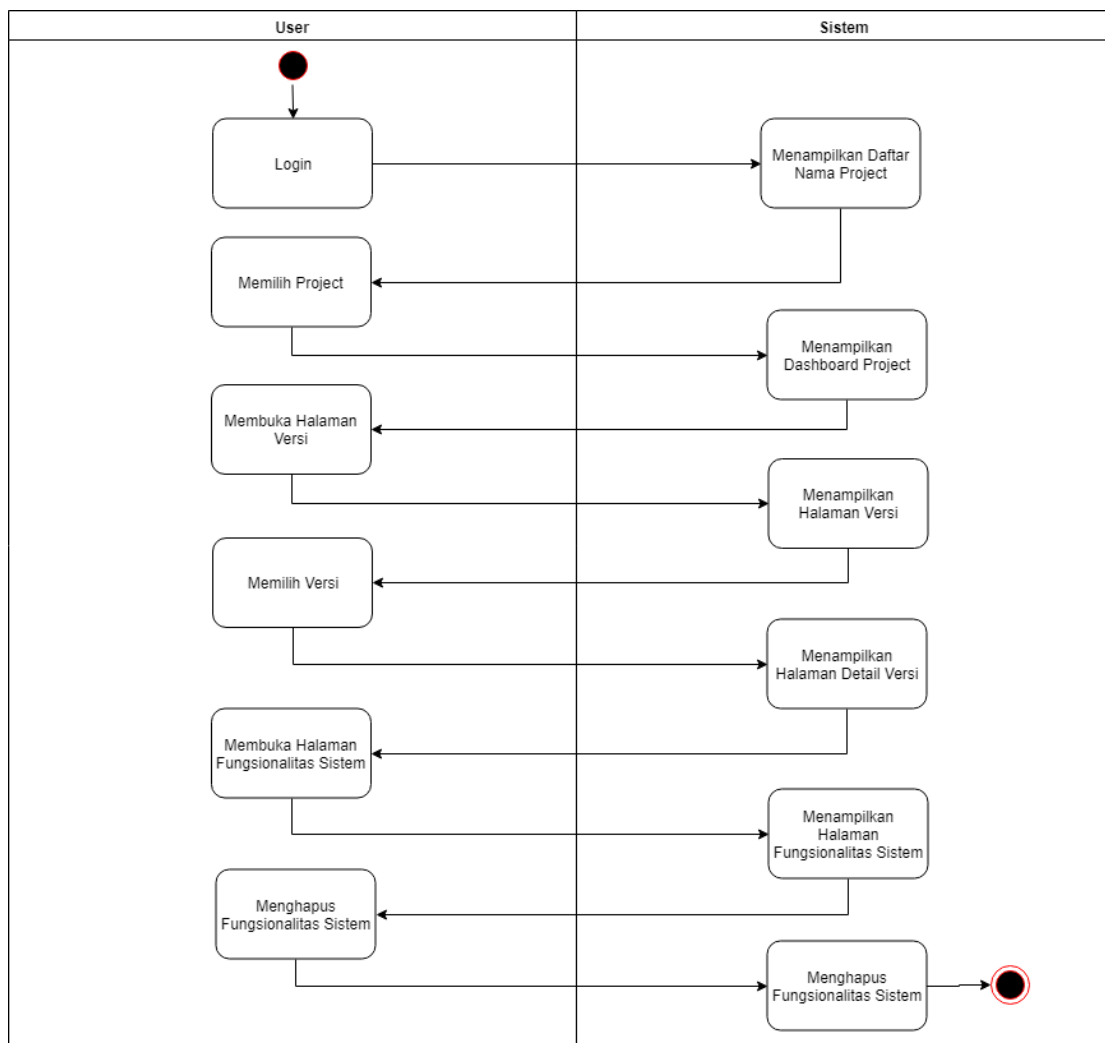
Tabel 3.25 Penjelasan alur kegiatan mengedit fungsionalitas sistem

No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama project
3	User	User memilih project yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail project
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu fungsionalitas sistem

No	Aktor	Aktivitas
10	Sistem	Sistem menampilkan halaman fungsionalitas sistem
11	User	User memilih edit fungsionalitas sistem yang ingin diedit
12	Sistem	Sistem menampilkan form edit fungsionalitas sistem
13	User	User mengubah data fungsionalitas sistem
14	Sistem	Sistem menyimpan data dan kembali menampilkan halaman fungsionalitas sistem

### 30. Activity Diagram Menghapus Fungsionalitas Sistem

Alur kerja sistem pada *use case* Menghapus Fungsionalitas Sistem dapat dilihat pada Gambar 3.34



Gambar 3.34 Activity diagram menghapus fungsionalitas sistem

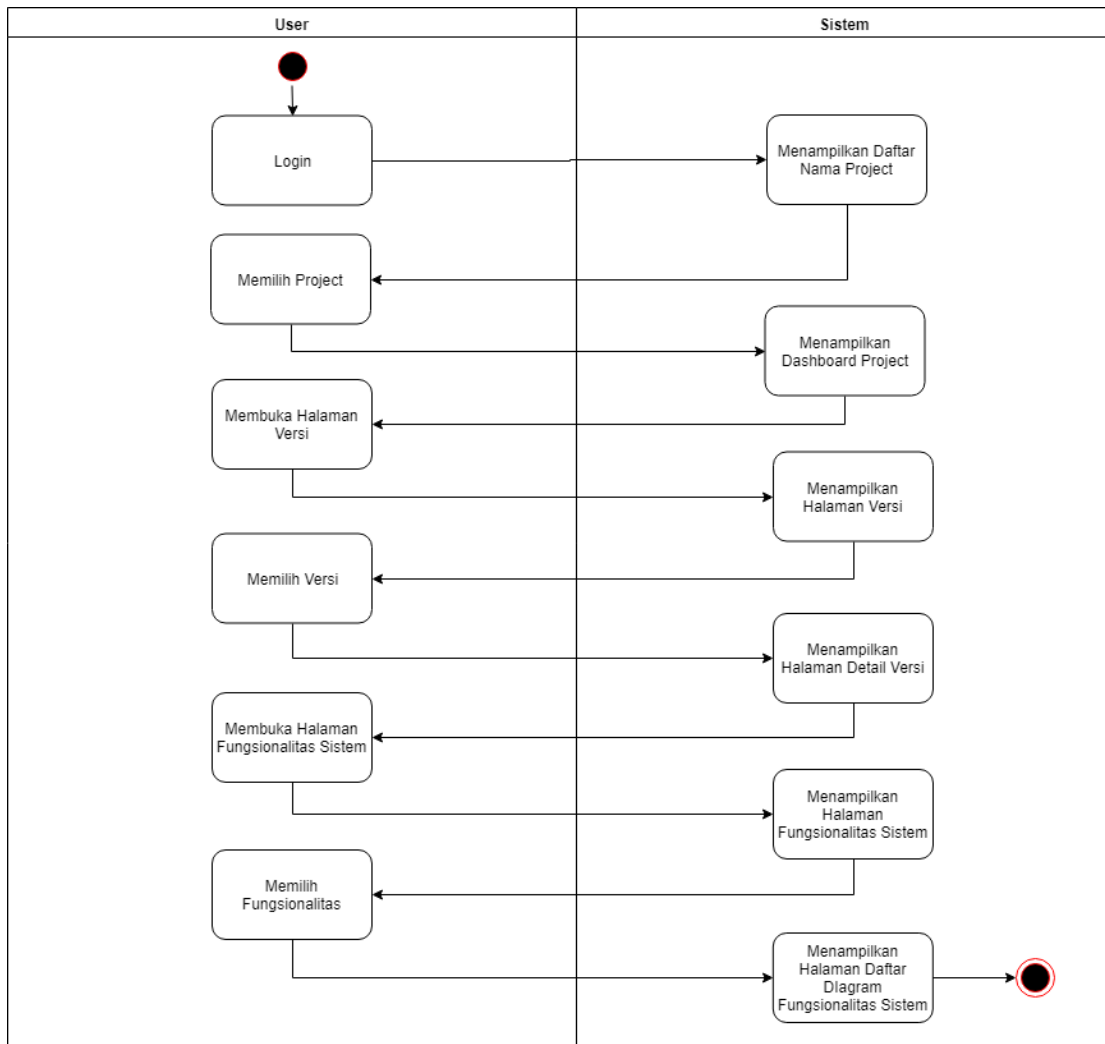
Penjelasan *activity diagram* pada Gambar 3.34 dapat dilihat pada Tabel 3.26

Tabel 3.26 Penjelasan alur kegiatan menghapus fungsionalitas sistem

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu fungsionalitas sistem
10	Sistem	Sistem menampilkan halaman fungsionalitas sistem
11	User	User memilih hapus fungsionalitas sistem yang ingin dihapus
12	Sistem	Sistem menghapus data dan kembali menampilkan halaman fungsionalitas sistem

### 31. Activity Diagram Melihat Diagram Fungsionalitas Sistem

Alur kerja sistem pada *use case* Melihat Diagram Fungsionalitas Sistem dapat dilihat pada Gambar 3.35



Gambar 3.35 Activity diagram melihat diagram fungsionalitas sistem

Penjelasan *activity diagram* pada Gambar 3.35 dapat dilihat pada Tabel 3.27

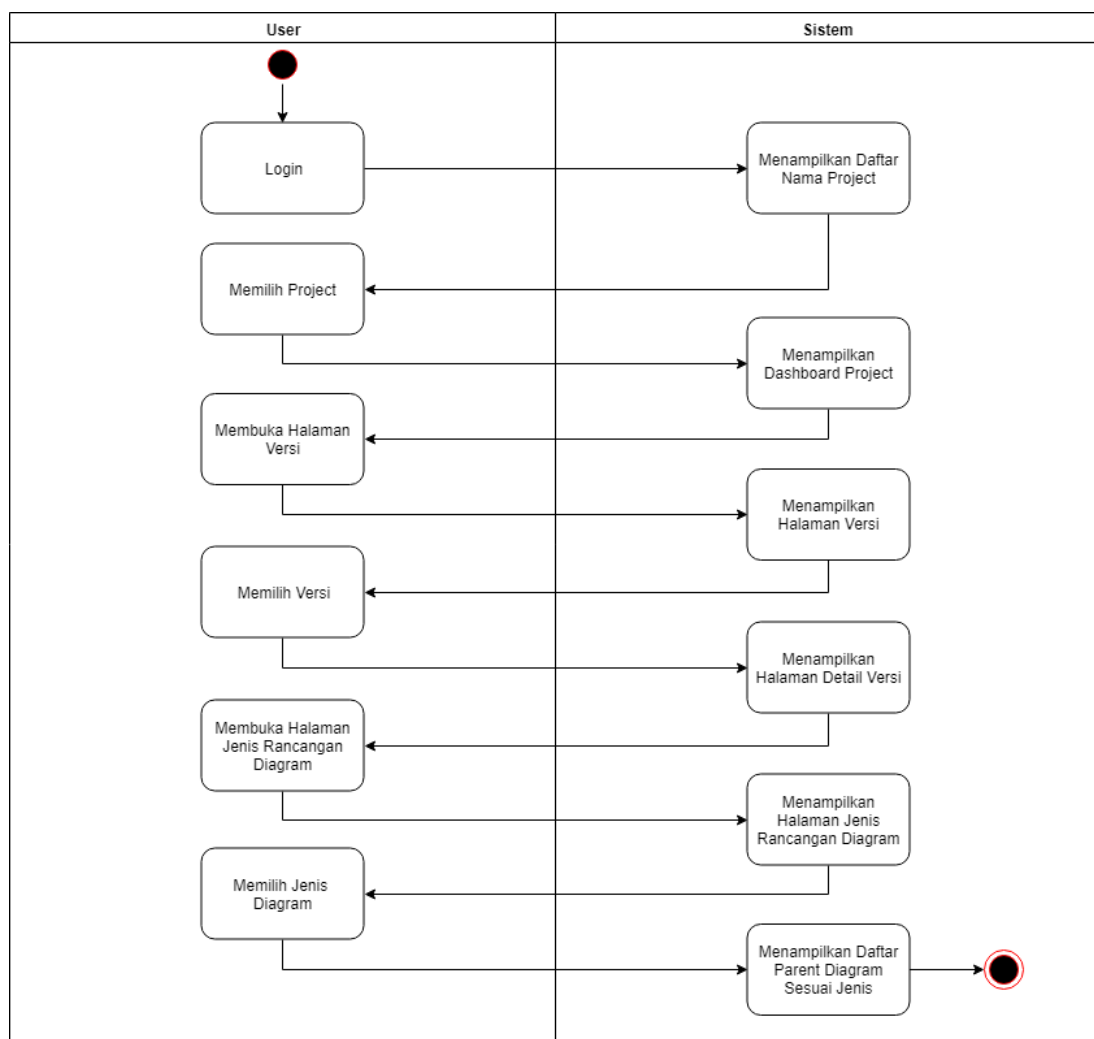
Tabel 3.27 Penjelasan alur kegiatan melihat diagram fungsionalitas sistem

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu fungsionalitas sistem

No	Aktor	Aktivitas
10	Sistem	Sistem menampilkan halaman fungsionalitas sistem
11	User	User memilih fungsionalitas sistem yang ingin dilihat diagramnya
12	Sistem	Sistem menampilkan diagram yang sesuai dengan fungsionalitas sistem yang dipilih

### 32. Activity Diagram Melihat Daftar Parent Diagram

Alur kerja sistem pada *use case* Melihat Daftar Parent Diagram dapat dilihat pada Gambar 3.36



Gambar 3.36 Activity diagram melihat daftar parent diagram

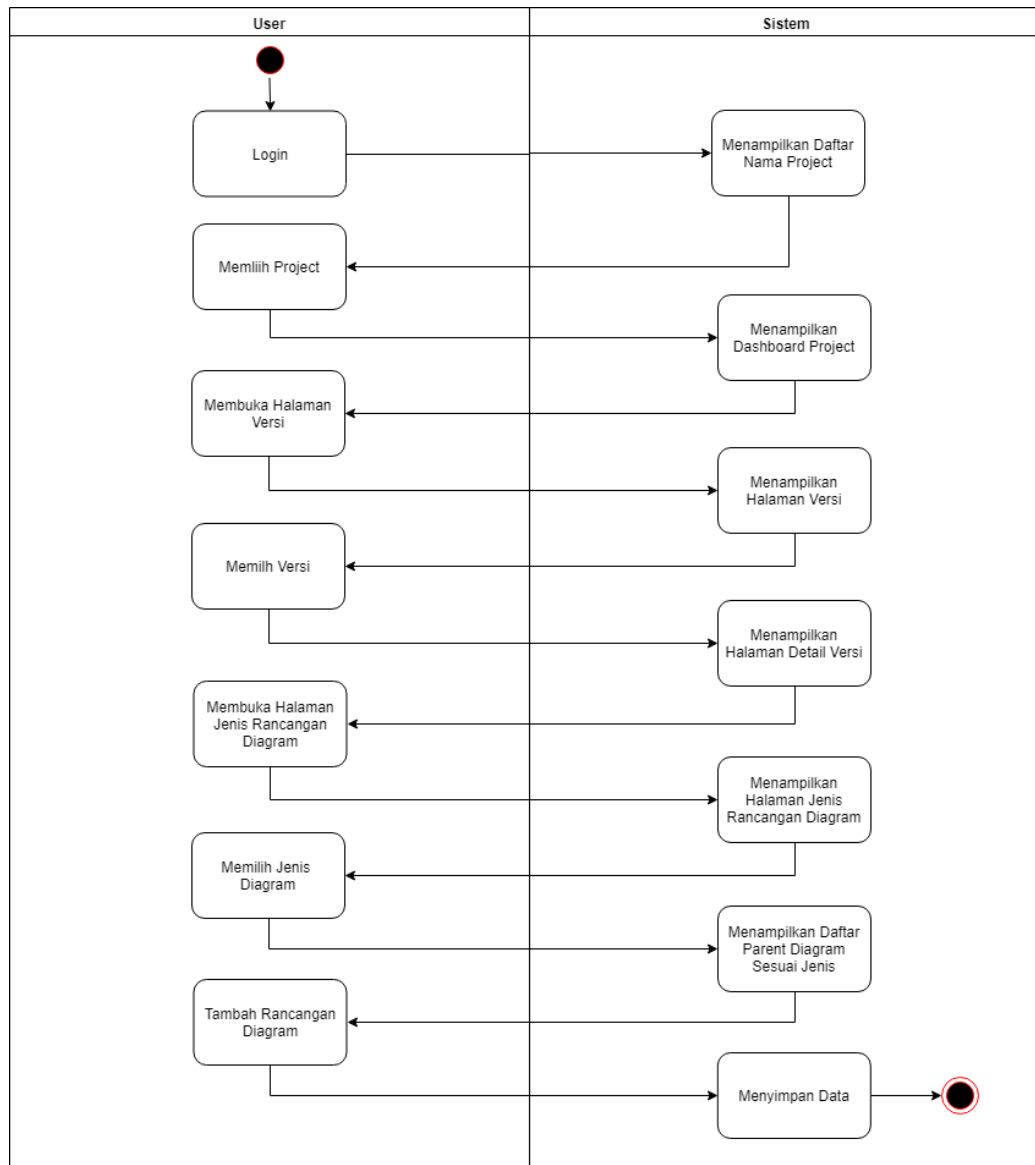
Penjelasan *activity diagram* pada Gambar 3.36 dapat dilihat pada Tabel 3.28

Tabel 3.28 Penjelasan alur kegiatan melihat daftar *parent* diagram

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu rancangan diagram
10	Sistem	Sistem menampilkan halaman jenis rancangan diagram
11	User	User memilih jenis diagram yang ingin dilihat diagramnya
12	Sistem	Sistem menampilkan halaman daftar <i>parent</i> diagram sesuai jenis diagram yang dipilih

### 33. Activity Diagram Menambah Parent Diagram

Alur kerja sistem pada *use case* Menambah Parent Diagram dapat dilihat pada Gambar 3.37



Gambar 3.37 Activity diagram menambah *parent* diagram

Penjelasan *activity diagram* pada Gambar 3.37 dapat dilihat pada Tabel 3.29

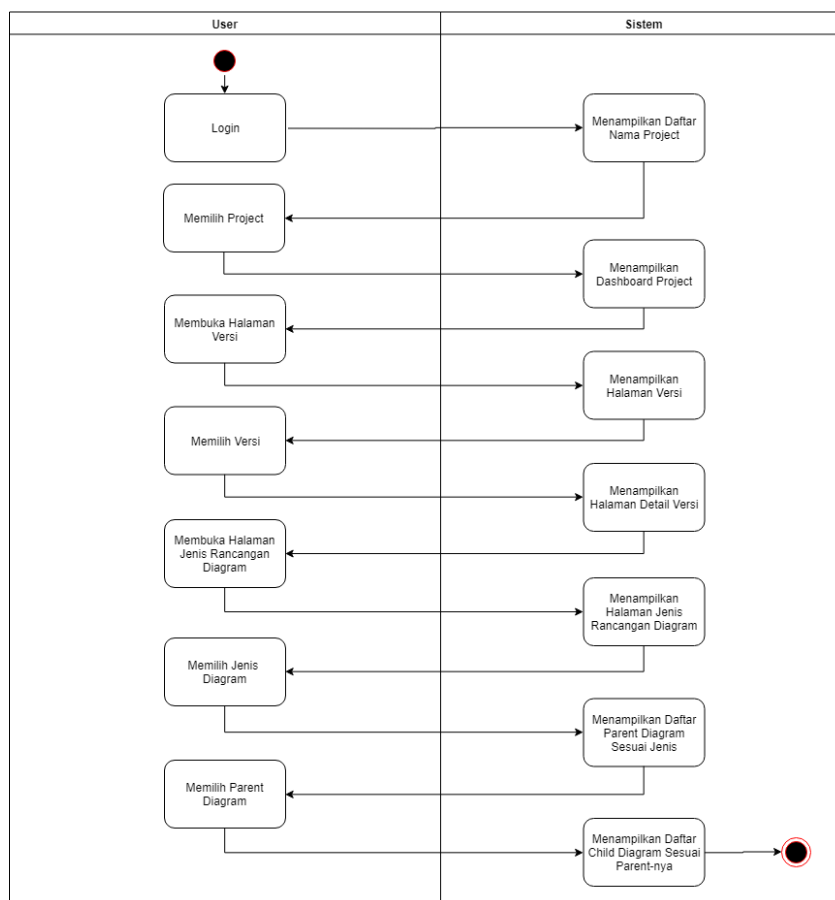
Tabel 3.29 Penjelasan alur kegiatan menambah *parent* diagram

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi

No	Aktor	Aktivitas
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu rancangan diagram
10	Sistem	Sistem menampilkan halaman jenis rancangan diagram
11	User	User memilih jenis diagram yang ingin dilihat diagramnya
12	Sistem	Sistem menampilkan halaman daftar <i>parent</i> diagram sesuai jenis diagram yang dipilih
13	User	User memasukkan data diagram baru
14	Sistem	Sistem menyimpan data dan kembali menampilkan halaman daftar <i>parent</i> diagram

### 34. Activity Diagram Melihat Daftar Child Diagram

Alur kerja sistem pada *use case* Melihat Daftar Child Diagram dapat dilihat pada Gambar 3.38



Gambar 3.38 Activity diagram melihat daftar child diagram



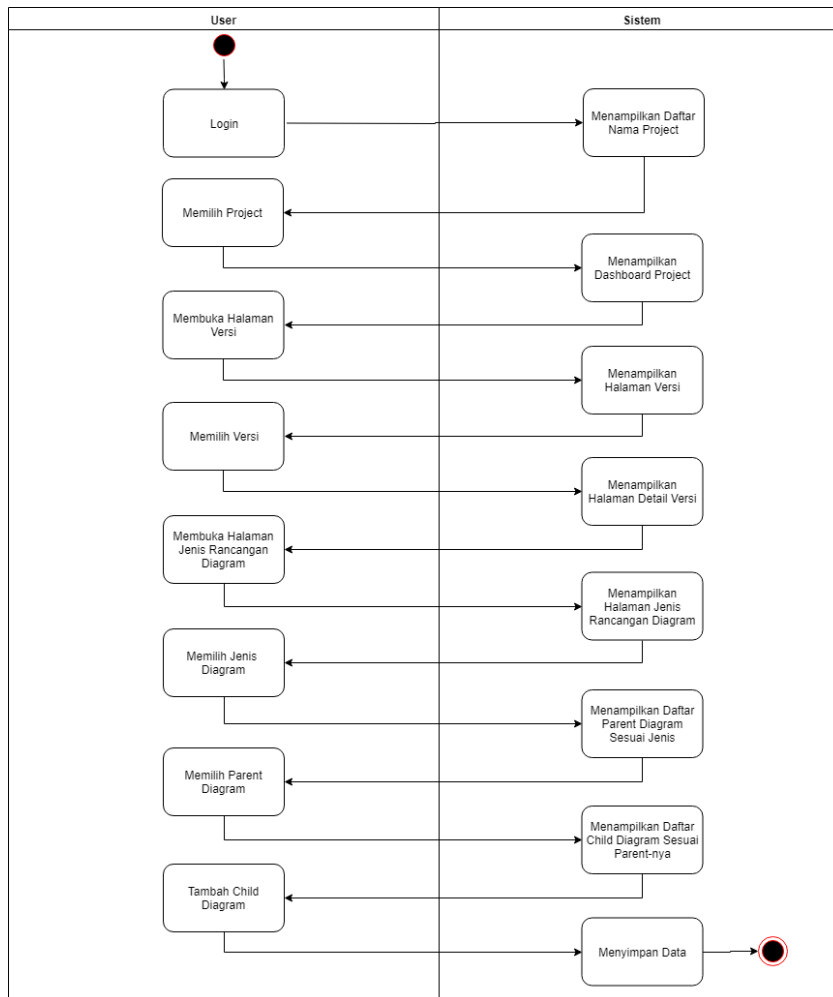
Penjelasan *activity diagram* pada Gambar 3.38 dapat dilihat pada Tabel 3.30

Tabel 3.30 Penjelasan alur kegiatan melihat daftar *child* diagram

No	Aktor	Aktivitas
1	<i>User</i>	<i>User</i> melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	<i>User</i>	<i>User</i> memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	<i>User</i>	<i>User</i> membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	<i>User</i>	<i>User</i> memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	<i>User</i>	<i>User</i> membuka menu rancangan diagram
10	Sistem	Sistem menampilkan halaman jenis rancangan diagram
11	<i>User</i>	<i>User</i> memilih jenis diagram yang ingin dilihat diagramnya
12	Sistem	Sistem menampilkan halaman daftar <i>parent</i> diagram sesuai jenis diagram yang dipilih
13	<i>User</i>	<i>User</i> memilih <i>parent</i> diagram
14	Sistem	Sistem akan menampilkan halaman <i>child</i> diagram beserta <i>parent</i> diagramnya

### 35. *Activity Diagram* Menambah *Child* Diagram

Alur kerja sistem pada *use case* Menambah *Child* Diagram dapat dilihat pada Gambar 3.39



Gambar 3.39 Activity diagram menambah *child* diagram

Penjelasan *activity diagram* pada Gambar 3.39 dapat dilihat pada Tabel 3.31

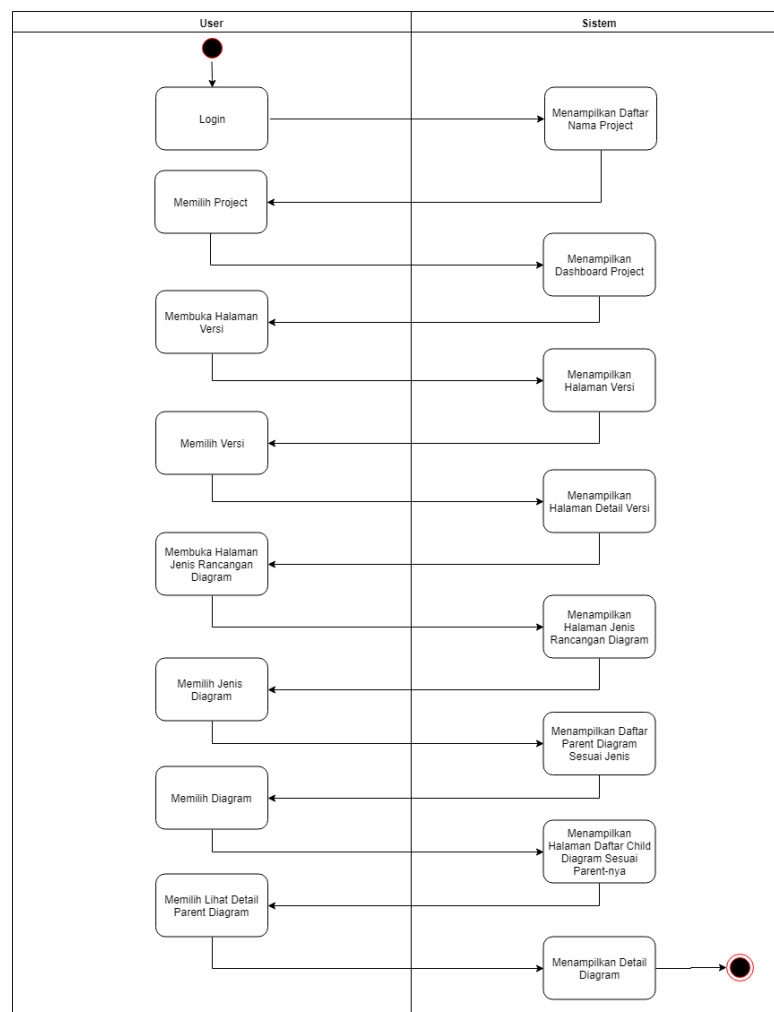
Tabel 3.31 Penjelasan alur kegiatan menambah *child* diagram

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu rancangan diagram

10	Sistem	Sistem menampilkan halaman jenis rancangan diagram
11	User	User memilih jenis diagram yang ingin dilihat diagramnya
12	Sistem	Sistem menampilkan halaman daftar <i>parent</i> diagram sesuai jenis diagram yang dipilih
13	User	User memilih <i>parent</i> diagram
14	Sistem	Sistem akan menampilkan halaman <i>child</i> diagram beserta <i>parent</i> diagramnya
15	User	User menambahkan data child diagram baru
16	Sistem	Sistem menyimpan data dan kembali menampilkan daftar child diagram dengan diagram <i>parent</i> -nya

### 36. Activity Diagram Melihat Detail Diagram

Alur kerja sistem pada *use case* Melihat Detail Diagram dapat dilihat pada Gambar 3.40



Gambar 3.40 Activity diagram melihat detail diagram

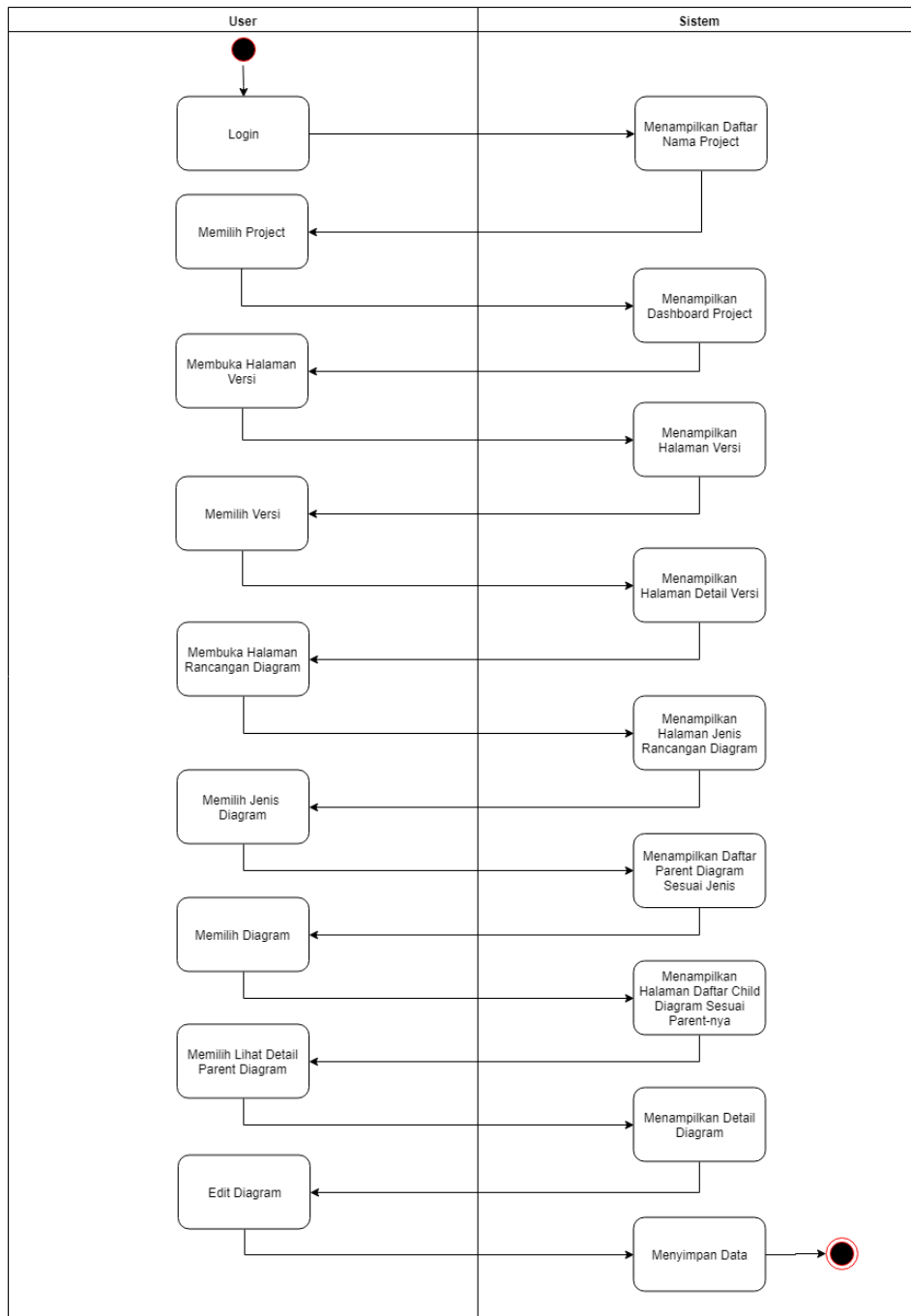
Penjelasan *activity diagram* pada Gambar 3.40 dapat dilihat pada Tabel 3.32

Tabel 3.32 Penjelasan alur kegiatan melihat detail diagram

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu rancangan diagram
10	Sistem	Sistem menampilkan halaman jenis rancangan diagram
11	User	User memilih jenis diagram yang ingin dilihat diagramnya
12	Sistem	Sistem menampilkan halaman daftar <i>parent</i> diagram sesuai jenis diagram yang dipilih
13	User	User memilih <i>parent</i> diagram
14	Sistem	Sistem akan menampilkan halaman <i>child</i> diagram beserta <i>parent</i> diagramnya
15	User	User memilih lihat detail <i>parent</i> diagram
16	Sistem	Sistem menampilkan halaman detail diagram

### 37. Activity Diagram Mengedit Diagram

Alur kerja sistem pada *use case* Mengedit Diagram dapat dilihat pada Gambar 3.41



Gambar 3.41 Activity diagram mengedit diagram

Penjelasan *activity diagram* pada Gambar 3.41 dapat dilihat pada Tabel 3.33

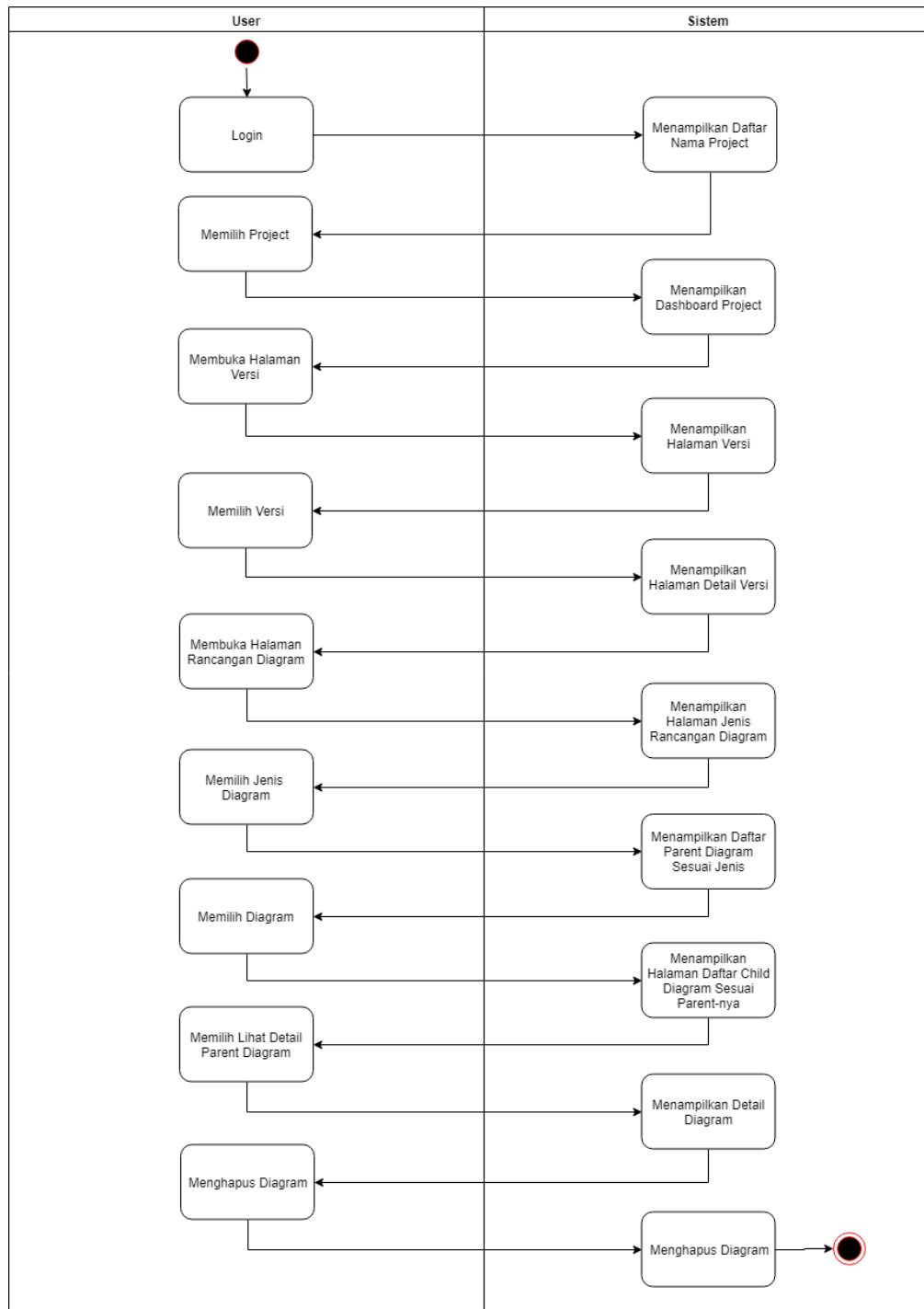
Tabel 3.33 Penjelasan alur kegiatan mengedit diagram

No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>

3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu rancangan diagram
10	Sistem	Sistem menampilkan halaman jenis rancangan diagram
11	User	User memilih jenis diagram yang ingin dilihat diagramnya
12	Sistem	Sistem menampilkan halaman daftar <i>parent</i> diagram sesuai jenis diagram yang dipilih
13	User	User memilih <i>parent</i> diagram
14	Sistem	Sistem akan menampilkan halaman <i>child</i> diagram beserta <i>parent</i> diagramnya
15	User	User memilih <i>parent</i> diagramnya
16	Sistem	Sistem menampilkan halaman detail <i>parent</i> diagram
17	User	User mengubah data diagram
18	Sistem	Sistem menyimpan data dan kembali menampilkan halaman detail diagram

### 38. Activity Diagram Menghapus Diagram

Alur kerja sistem pada *use case* Menghapus Diagram dapat dilihat pada Gambar 3.42



Gambar 3.42 Activity diagram menghapus diagram

Penjelasan activity diagram pada Gambar 3.42 dapat dilihat pada Tabel 3.34

Tabel 3.34 Penjelasan alur kegiatan menghapus diagram

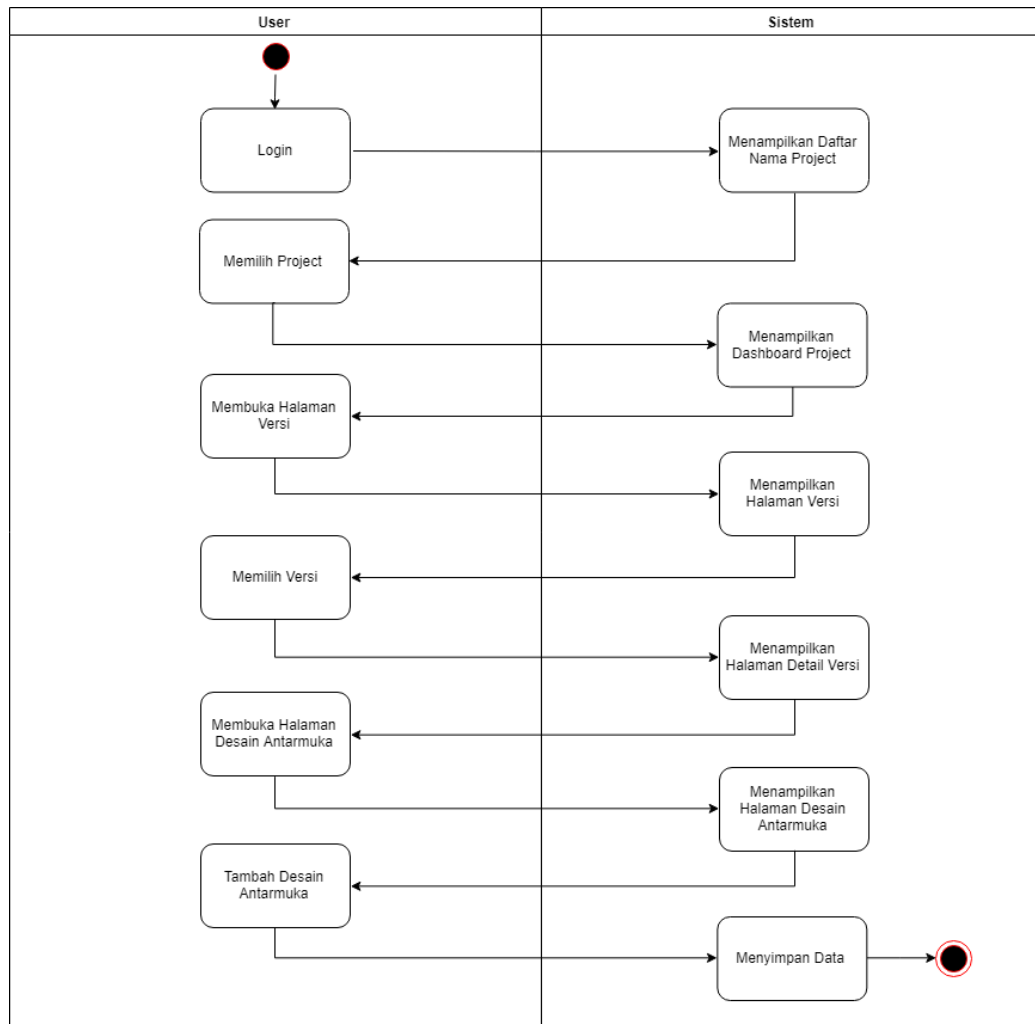
No	Aktor	Aktivitas
1	User	User melakukan login

2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu rancangan diagram
10	Sistem	Sistem menampilkan halaman jenis rancangan diagram
11	User	User memilih jenis diagram yang ingin dilihat diagramnya
12	Sistem	Sistem menampilkan halaman daftar <i>parent</i> diagram sesuai jenis diagram yang dipilih
13	User	User memilih <i>parent</i> diagram
14	Sistem	Sistem akan menampilkan halaman <i>child</i> diagram beserta <i>parent</i> diagramnya
15	User	User memilih <i>parent</i> diagramnya
16	Sistem	Sistem menampilkan halaman detail <i>parent</i> diagram
17	User	User memilih hapus diagram
18	Sistem	Sistem menghapus diagram dan menampilkan halaman daftar <i>parent</i> diagram

### 39. Activity Diagram Menambah Desain Antarmuka

Alur kerja sistem pada *use case* Menambah Desain Antarmuka dapat dilihat pada Gambar 3.43





Gambar 3.43 Activity diagram menambah desain antarmuka

Penjelasan activity diagram pada Gambar 3.43 dapat dilihat pada Tabel 3.35

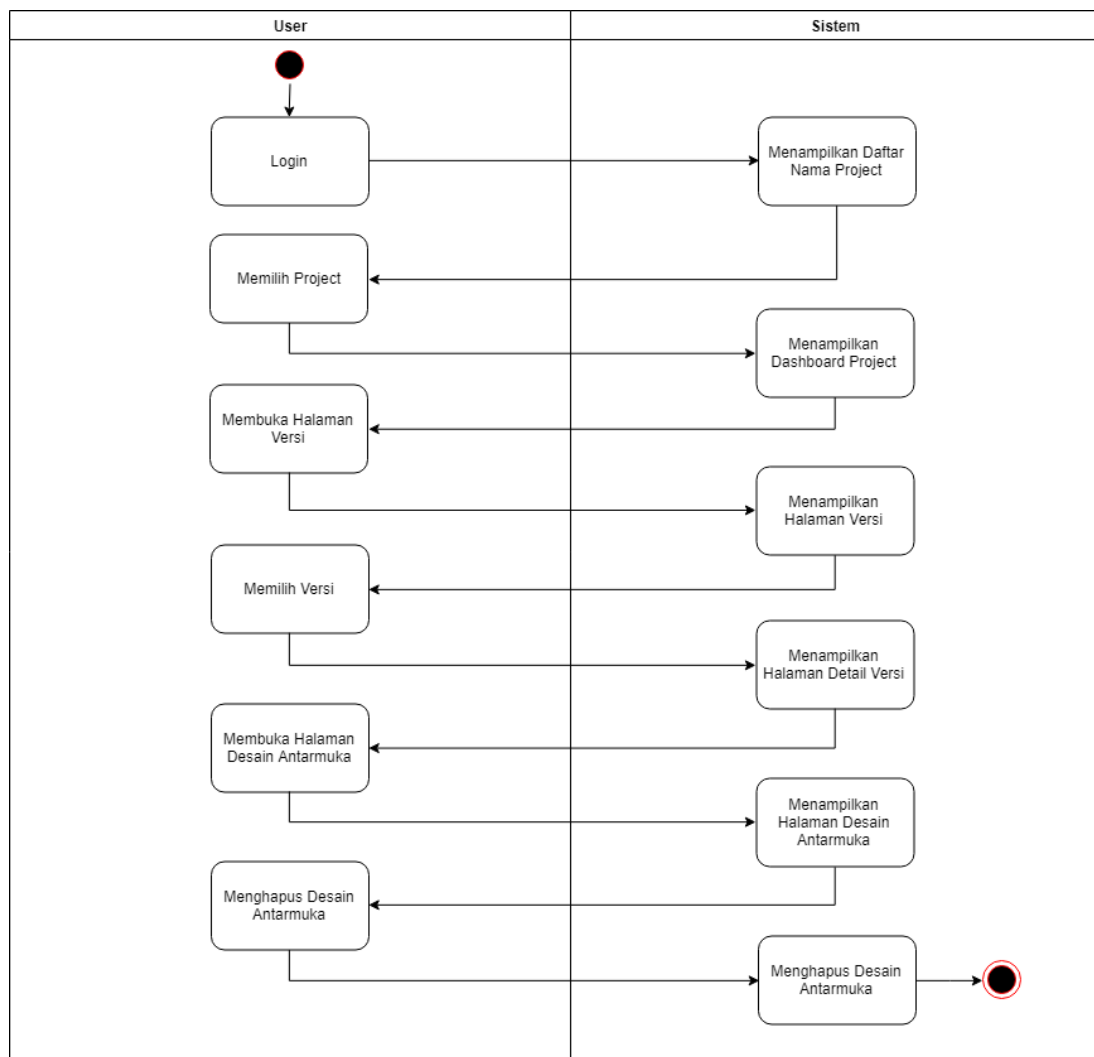
Tabel 3.35 Penjelasan alur kegiatan menambah desain antarmuka

No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu desain antarmuka

No	Aktor	Aktivitas
10	Sistem	Sistem menampilkan halaman desain antarmuka
11	User	User memilih tambah desain antarmuka
12	Sistem	Sistem menampilkan form tambah desain antarmuka
13	User	User menambahkan data desain antarmuka baru
14	Sistem	Sistem menyimpan data dan kembali menampilkan halaman daftar desain antarmuka

#### 40. Activity Diagram Menghapus Desain Antarmuka

Alur kerja sistem pada *use case* Menghapus Desain Antarmuka dapat dilihat pada Gambar 3.44



Gambar 3.44 Activity diagram menghapus desain antarmuka

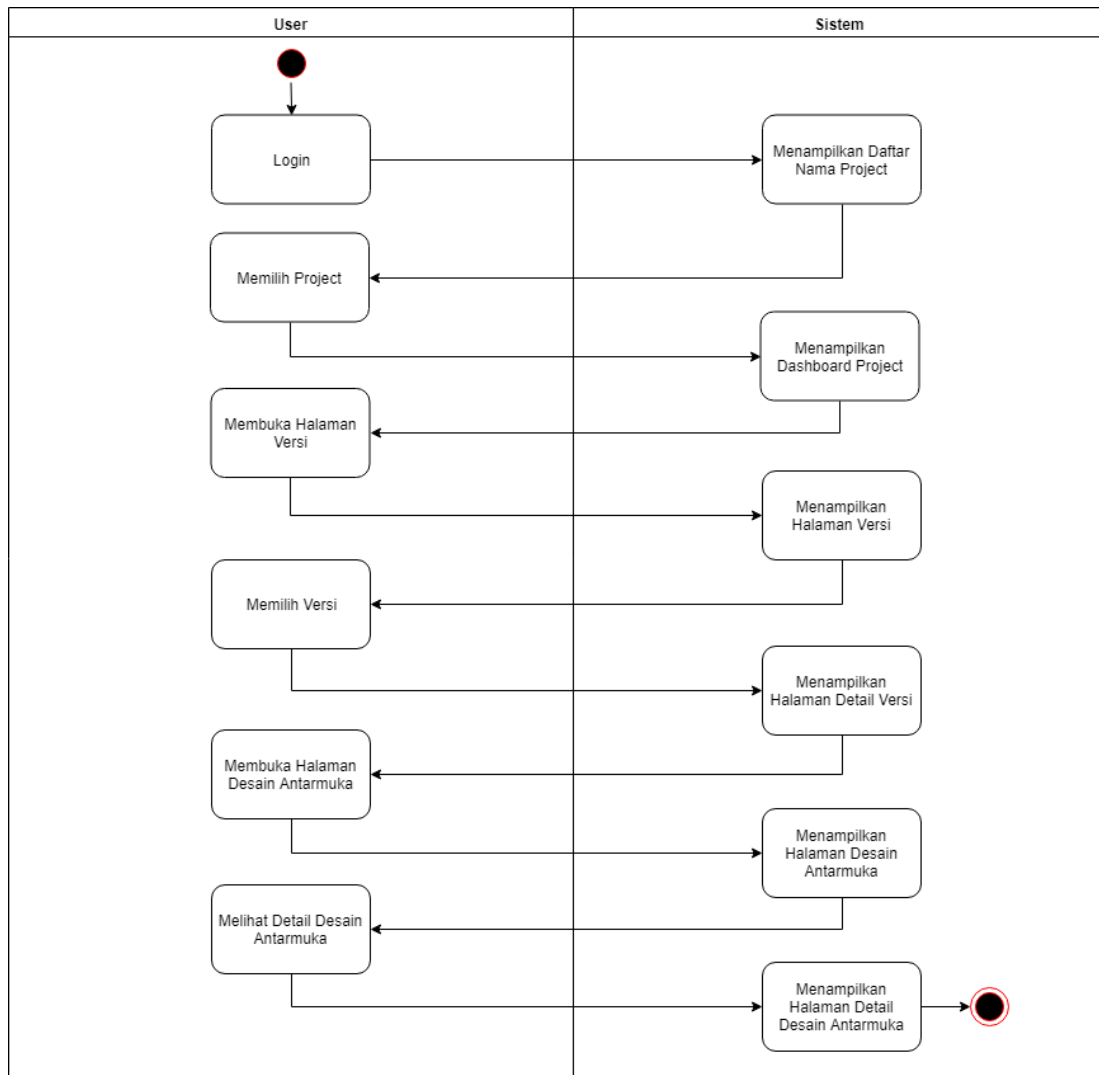
Penjelasan *activity diagram* pada Gambar 3.44 dapat dilihat pada Tabel 3.36

Tabel 3.36 Penjelasan alur kegiatan menghapus desain antarmuka

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu desain antarmuka
10	Sistem	Sistem menampilkan halaman desain antarmuka
11	User	User memilih hapus desain antarmuka yang ingin dihapus
14	Sistem	Sistem menghapus data dan kembali menampilkan halaman daftar desain antarmuka

#### 41. Activity Diagram Melihat Detail Desain Antarmuka

Alur kerja sistem pada *use case* Melihat Detail Desain Antarmuka dapat dilihat pada Gambar 3.45



Gambar 3.45 Activity diagram melihat detail desain antarmuka

Penjelasan *activity diagram* pada Gambar 3.45 dapat dilihat pada Tabel 3.37

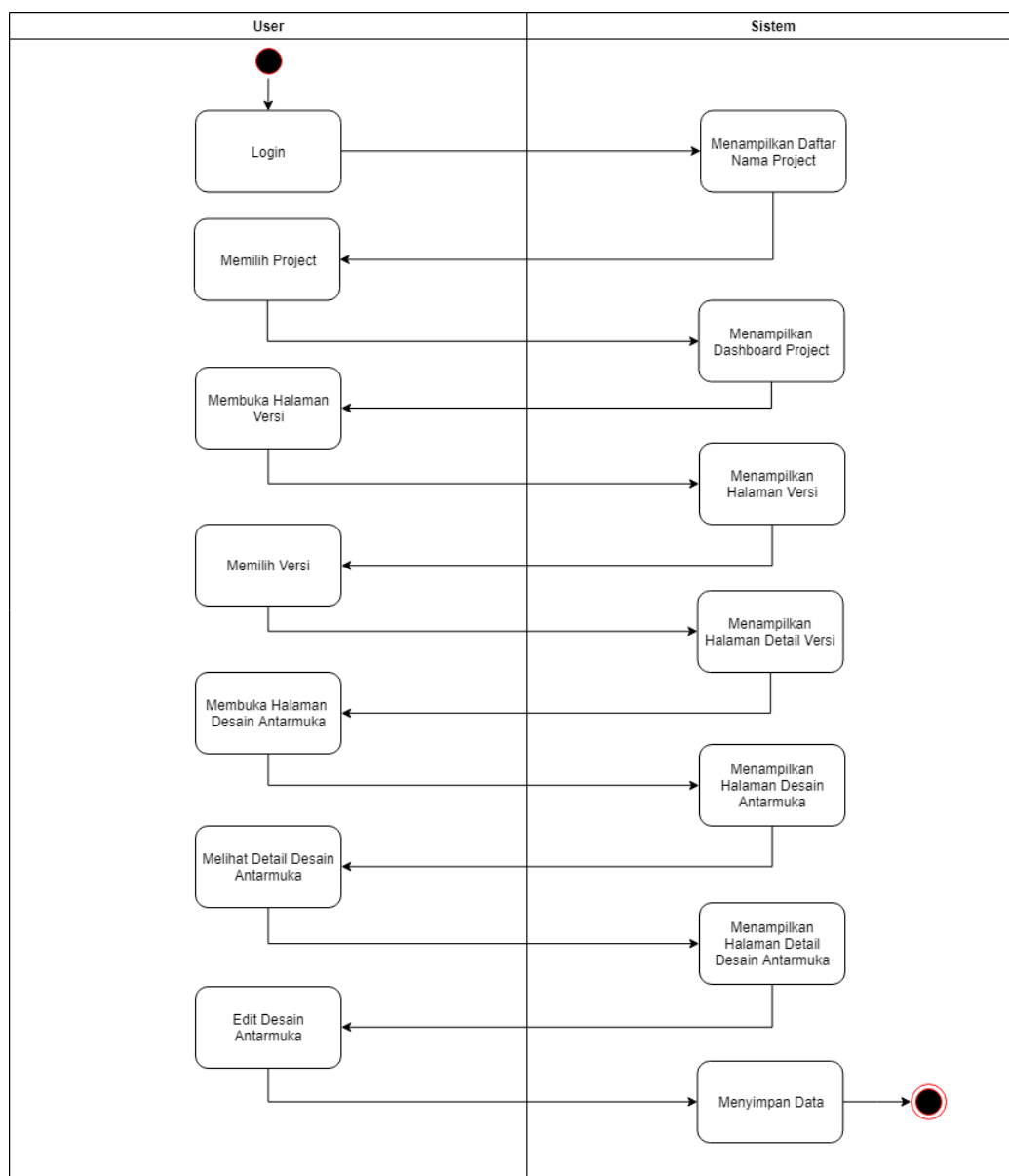
Tabel 3.37 Penjelasan alur kegiatan melihat detail desain antarmuka

No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi

No	Aktor	Aktivitas
9	User	User membuka menu desain antarmuka
10	Sistem	Sistem menampilkan halaman desain antarmuka
11	User	User memilih lihat desain antarmuka yang ingin dilihat detailnya
12	Sistem	Sistem menampilkan halaman detail desain antarmuka yang dipilih

#### 42. Activity Diagram Mengedit Desain Antarmuka

Alur kerja sistem pada *use case* Mengedit Desain Antarmuka dapat dilihat pada Gambar 3.46



Gambar 3.46 Activity diagram mengedit desain antarmuka

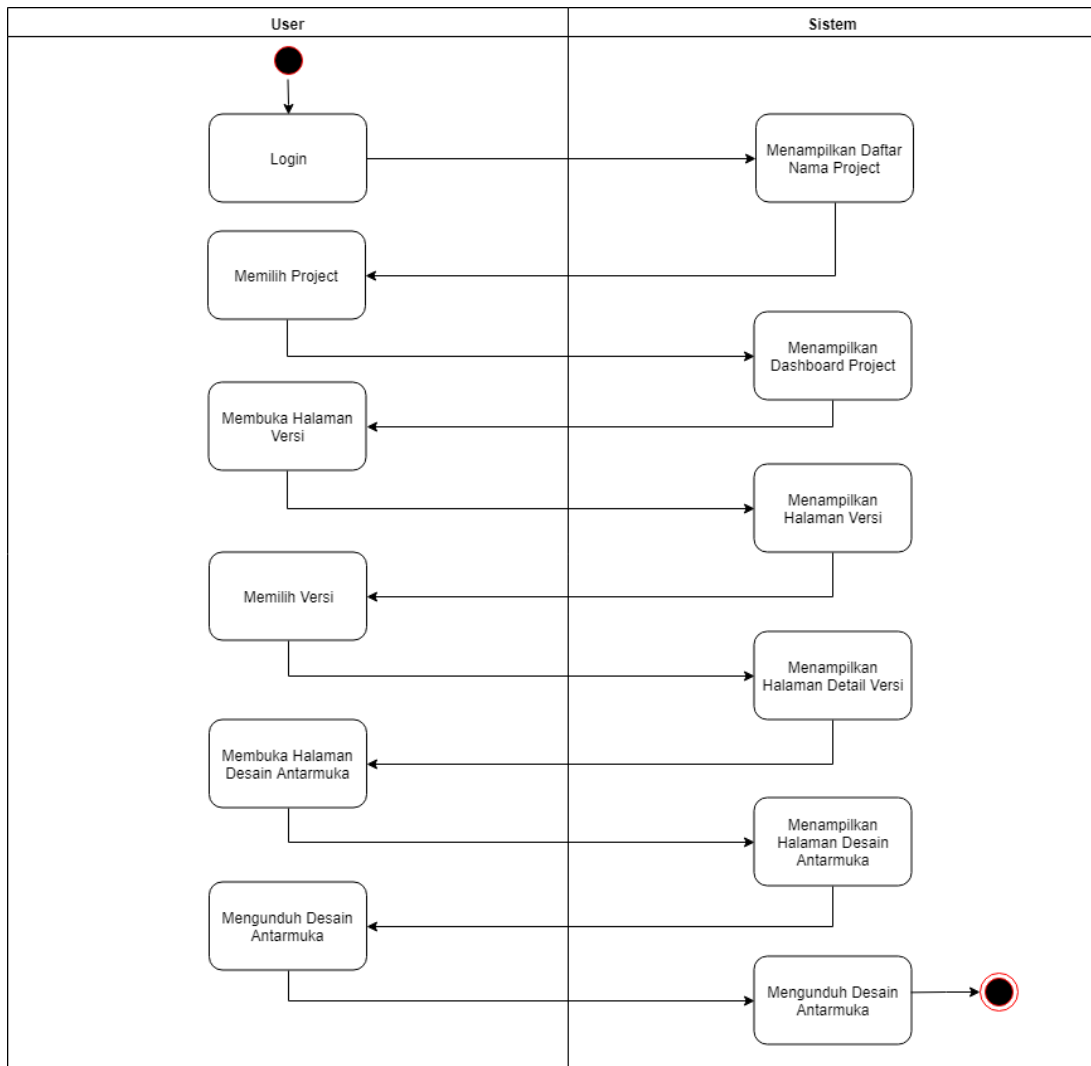
Penjelasan *activity diagram* pada Gambar 3.46 dapat dilihat pada Tabel 3.38

Tabel 3.38 Penjelasan alur kegiatan mengedit desain antarmuka

No	Aktor	Aktivitas
1	User	User melakukan <i>login</i>
2	Sistem	Sistem menampilkan halaman daftar nama <i>project</i>
3	User	User memilih <i>project</i> yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail <i>project</i>
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi
9	User	User membuka menu desain antarmuka
10	Sistem	Sistem menampilkan halaman desain antarmuka
11	User	User memilih lihat desain antarmuka yang ingin dilihat detailnya
12	Sistem	Sistem menampilkan halaman detail desain antarmuka yang dipilih
13	User	User memilih edit desain antarmuka
14	Sistem	Sistem menampilkan form edit desain antarmuka
15	User	User mengubah data desain antarmuka
16	Sistem	Sistem menyimpan data dan kembali menampilkan halaman detail desain antarmuka

#### 43. *Activity Diagram* Mengunduh Desain Antarmuka

Alur kerja sistem pada *use case* mengunduh desain antarmuka dapat dilihat pada Gambar 3.47



Gambar 3.47 Activity diagram mengunduh desain antarmuka

Penjelasan activity diagram pada Gambar 3.47 dapat dilihat pada Tabel 3.39

Tabel 3.39 Penjelasan alur kegiatan mengunduh desain antarmuka

No	Aktor	Aktivitas
1	User	User melakukan login
2	Sistem	Sistem menampilkan halaman daftar nama project
3	User	User memilih project yang ingin dilihat detailnya
4	Sistem	Sistem menampilkan halaman detail project
5	User	User membuka menu versi
6	Sistem	Sistem menampilkan halaman versi
7	User	User memilih versi
8	Sistem	Sistem menampilkan halaman detail versi

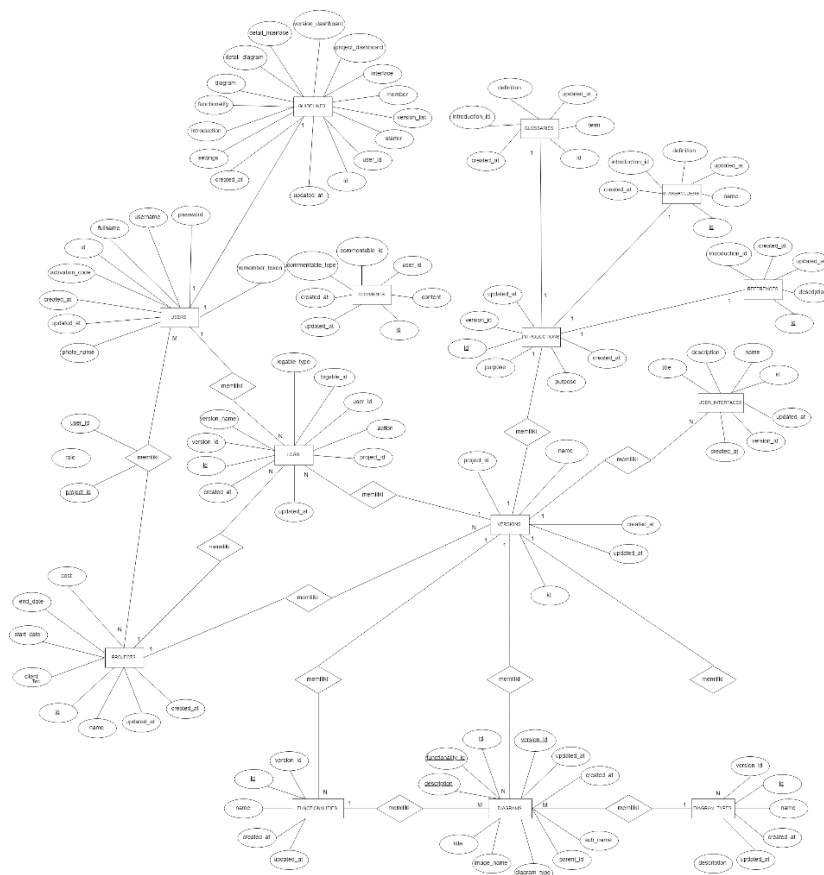
No	Aktor	Aktivitas
9	User	User membuka menu desain antarmuka
10	Sistem	Sistem menampilkan halaman desain antarmuka
11	User	User memilih unduh desain antarmuka
12	Sistem	Sistem mengunduh desain antarmuka

### 3.4 Perancangan Basisdata

Perancangan basisdata sistem ini berguna untuk memberikan gambaran rancangan basisdata baik diagram maupun struktur setiap tabel yang ada di dalam basisdata yang digunakan

#### 3.4.1 Entity Relationship Diagram

Entity Relationship Diagram (ERD) adalah suatu model yang digunakan untuk menjelaskan hubungan antar data dalam basisdata berdasarkan objek-objek data yang mempunyai hubungan antar relasi. Berikut rancangan ERD dari sistem ini.



Gambar 3.48 Entity Relationship Diagram



### 3.4.2 Struktur Tabel Basisdata

Struktur tabel pada sistem ini menjelaskan mengenai struktur detail dari setiap *table* yang terdapat dalam *database*. *Database* pada sistem ini terdiri dari 15 tabel. Berikut rincian setiap *table* yang terdapat pada sistem ini.

#### a. Tabel *Users*

Tabel *users* menyimpan data *user*/pengguna yang terdiri dari beberapa kolom, yaitu *id*, *fullname*, *email*, *username*, *password*, *photo\_name*, *activation\_code*, *remember\_token*, *created\_at*, dan *updated\_at*. Struktur tabel selengkapnya dapat dilihat pada Tabel 3.40

Tabel 3.40 Struktur Tabel *Users*

Nama Kolom	Tipe Data	Panjang	Keterangan
<i>id</i>	<i>integer</i>	10	Primary Key
<i>fullname</i>	<i>varchar</i>	100	
<i>email</i>	<i>varchar</i>	100	
<i>username</i>	<i>varchar</i>	100	
<i>password</i>	<i>varchar</i>	100	
<i>photo_name</i>	<i>varchar</i>	100	
<i>activation_code</i>	<i>varchar</i>	100	
<i>remember_token</i>	<i>varchar</i>	100	
<i>created_at</i>	<i>timestamp</i>		
<i>updated_at</i>	<i>timestamp</i>		

#### b. Tabel *Projects*

Tabel *projects* menyimpan data *project* yang terdiri dari beberapa kolom, yaitu *id*, *name*, *client*, *start\_date*, *end\_date*, *cost*, *created\_at*, dan *updated\_at*. Struktur tabel selengkapnya dapat dilihat pada Tabel 3.41

Tabel 3.41 Struktur Tabel *Projects*

Nama Kolom	Tipe Data	Panjang	Keterangan
<i>id</i>	<i>integer</i>	10	Primary Key
<i>name</i>	<i>varchar</i>	100	
<i>client</i>	<i>varchar</i>	100	
<i>start_date</i>	<i>date</i>		
<i>end_date</i>	<i>date</i>		

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang</b>	<b>Keterangan</b>
<i>cost</i>	<i>integer</i>	11	
<i>created_at</i>	<i>timestamp</i>		
<i>updated_at</i>	<i>timestamp</i>		

c. Tabel *Versions*

Tabel *versions* menyimpan data *version* yang terdiri dari beberapa kolom, yaitu *id*, *name*, *project\_id*, *created\_at*, dan *updated\_at*. Struktur tabel selengkapnya dapat dilihat pada Tabel 3.42

Tabel 3.42 Struktur Tabel *Versions*

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang</b>	<b>Keterangan</b>
<i>id</i>	<i>integer</i>	10	Primary Key
<i>name</i>	<i>varchar</i>	100	
<i>project_id</i>	<i>integer</i>	10	Foreign Key
<i>created_at</i>	<i>timestamp</i>		
<i>updated_at</i>	<i>timestamp</i>		

d. Tabel *Introductions*

Tabel *introductions* menyimpan data menu pengantar yang terdiri dari beberapa kolom, yaitu *id*, *purpose*, *scope*, *version\_id*, *created\_at*, dan *updated\_at*. Struktur tabel selengkapnya dapat dilihat pada Tabel 3.43

Tabel 3.43 Struktur Tabel *Introductions*

<b>Nama Kolom</b>	<b>Tipe Data</b>	<b>Panjang</b>	<b>Keterangan</b>
<i>id</i>	<i>integer</i>	10	Primary Key
<i>purpose</i>	<i>text</i>		
<i>scope</i>	<i>text</i>		
<i>version_id</i>	<i>integer</i>	10	Foreign Key
<i>created_at</i>	<i>timestamp</i>		
<i>updated_at</i>	<i>timestamp</i>		

e. Tabel *Functionalities*

Tabel *functionalities* menyimpan data menu fungsionalitas sistem yang terdiri dari beberapa kolom, yaitu *id*, *name*, *version\_id*, *created\_at*, dan *updated\_at*. Struktur tabel selengkapnya dapat dilihat pada Tabel 3.44

Tabel 3.44 Struktur Tabel *Functionalities*

Nama Kolom	Tipe Data	Panjang	Keterangan
<i>id</i>	<i>integer</i>	10	Primary Key
<i>name</i>	<i>varchar</i>	100	
<i>version_id</i>	<i>integer</i>	10	Foreign Key
<i>created_at</i>	<i>timestamp</i>		
<i>updated_at</i>	<i>timestamp</i>		

f. Tabel *Diagrams*

Tabel *diagrams* menyimpan data seluruh diagram yang ada di sistem. *Table* ini terdiri dari beberapa kolom, yaitu *id*, *title*, *image\_name*, *diagram\_type*, *parent\_id*, *sub\_name*, *functionality\_id*, *version\_id*, *created\_at*, dan *updated\_at*. Struktur tabel selengkapnya dapat dilihat pada Tabel 3.45

Tabel 3.45 Struktur Tabel *Diagrams*

Nama Kolom	Tipe Data	Panjang	Keterangan
<i>id</i>	<i>integer</i>	10	Primary Key
<i>title</i>	<i>varchar</i>	100	
<i>description</i>	<i>text</i>		
<i>image_name</i>	<i>varchar</i>	100	
<i>diagram_type</i>	<i>varchar</i>	100	
<i>parent_id</i>	<i>integer</i>	10	
<i>sub_name</i>	<i>varchar</i>	100	
<i>functionality_id</i>	<i>integer</i>	10	Foreign Key
<i>version_id</i>	<i>integer</i>	10	Foreign Key
<i>created_at</i>	<i>timestamp</i>		
<i>updated_at</i>	<i>timestamp</i>		

g. *Table Diagram\_types*

Tabel *diagram\_types* menyimpan data tipe diagram yang nantinya akan digunakan pada *table* diagram. *Table* ini terdiri dari beberapa kolom, yaitu *id*, *name*, *version\_id*, *created\_at*, dan *updated\_at*. Struktur tabel selengkapnya dapat dilihat pada Tabel 3.46

Tabel 3.46 Struktur Tabel *Diagram\_types*

Nama Kolom	Tipe Data	Panjang	Keterangan
<i>id</i>	<i>integer</i>	10	Primary Key
<i>name</i>	<i>varchar</i>	100	
<i>description</i>	<i>text</i>		
<i>version_id</i>	<i>integer</i>	10	Foreign Key
<i>created_at</i>	<i>timestamp</i>		
<i>updated_at</i>	<i>timestamp</i>		

h. *Tabel User\_interfaces*

Tabel *user\_interfaces* menyimpan data desain antarmuka yang terdiri dari beberapa kolom, yaitu *id*, *name*, *title*, *version\_id*, *created\_at*, dan *updated\_at*. Struktur tabel selengkapnya dapat dilihat pada Tabel 3.47

Tabel 3.47 Struktur Tabel *User\_interfaces*

Nama Kolom	Tipe Data	Panjang	Keterangan
<i>id</i>	<i>integer</i>	10	Primary Key
<i>name</i>	<i>varchar</i>	100	
<i>title</i>	<i>varchar</i>	100	
<i>description</i>	<i>text</i>		
<i>version_id</i>	<i>integer</i>	10	Foreign Key
<i>created_at</i>	<i>timestamp</i>		
<i>updated_at</i>	<i>timestamp</i>		

i. *Tabel Comments*

Tabel *comments* menyimpan data komentar yang terdiri dari beberapa kolom, yaitu *id*, *content*, *user\_id*, *commentable\_id*, *commentable\_type*, *created\_at*, dan *updated\_at*. Struktur tabel selengkapnya dapat dilihat pada Tabel 3.48

Tabel 3.48 Struktur Tabel *Comments*

<b>Nama Kolom</b>	<b>Type Data</b>	<b>Panjang</b>	<b>Keterangan</b>
<i>id</i>	<i>integer</i>	10	Primary Key
<i>content</i>	<i>varchar</i>	100	
<i>user_id</i>	<i>varchar</i>	100	
<i>commentable_id</i>	<i>integer</i>	10	
<i>commentable_type</i>	<i>varchar</i>	100	
<i>created_at</i>	<i>timestamp</i>		
<i>updated_at</i>	<i>timestamp</i>		

j. Tabel *Logs*

Tabel *logs* menyimpan data aktivitas yang terdiri dari beberapa kolom, yaitu *id*, *action*, *logable\_id*, *logable\_type*, *user\_id*, *project\_id*, *version\_id*, *created\_at*, dan *updated\_at*. Struktur tabel selengkapnya dapat dilihat pada Tabel 3.49

Tabel 3.49 Struktur Tabel *Logs*

<b>Nama Kolom</b>	<b>Type Data</b>	<b>Panjang</b>	<b>Keterangan</b>
<i>id</i>	<i>integer</i>	10	Primary Key
<i>action</i>	<i>varchar</i>	100	
<i>logable_id</i>	<i>integer</i>	10	
<i>logable_type</i>	<i>varchar</i>	100	
<i>user_id</i>	<i>integer</i>	10	Foreign Key
<i>project_id</i>	<i>integer</i>	10	Foreign Key
<i>version_id</i>	<i>integer</i>	10	
<i>version_name</i>	<i>varchar</i>	100	
<i>created_at</i>	<i>timestamp</i>		
<i>updated_at</i>	<i>timestamp</i>		

k. Tabel *Glossaries*

Tabel *glossaries* menyimpan data glosarium yang terdiri dari beberapa kolom, yaitu *id*, *term*, *definition*, *introduction\_id*, *project\_id*, *version\_id*, *created\_at*, dan *updated\_at*. Struktur tabel selengkapnya dapat dilihat pada Tabel 3.50

Tabel 3.50 Struktur Tabel *Glossaries*

Nama Kolom	Tipe Data	Panjang	Keterangan
<i>id</i>	<i>integer</i>	10	Primary Key
<i>term</i>	<i>varchar</i>	100	
<i>definition</i>	<i>text</i>	10	
<i>introduction_id</i>	<i>integer</i>	10	Foreign Key
<i>created_at</i>	<i>timestamp</i>		
<i>updated_at</i>	<i>timestamp</i>		

l. Tabel *References*

Tabel *references* menyimpan data referensi *project* yang terdiri dari beberapa kolom, yaitu *id*, *description*, *introduction\_id*, *project\_id*, *version\_id*, *created\_at*, dan *updated\_at*. Struktur tabel selengkapnya dapat dilihat pada Tabel 3.50

Tabel 3.51 Struktur Tabel *References*

Nama Kolom	Tipe Data	Panjang	Keterangan
<i>id</i>	<i>integer</i>	10	Primary Key
<i>description</i>	<i>text</i>		
<i>introduction_id</i>	<i>integer</i>	10	Foreign Key
<i>created_at</i>	<i>timestamp</i>		
<i>updated_at</i>	<i>timestamp</i>		

m. Tabel *Stakeholders*

Tabel *stakeholders* menyimpan data pengguna yang akan menggunakan sistem yang terdiri dari beberapa kolom, yaitu *id*, *name*, *description*, *introduction\_id*, *project\_id*, *version\_id*, *created\_at*, dan *updated\_at*. Struktur tabel selengkapnya dapat dilihat pada Tabel 3.50

Tabel 3.52 Struktur Tabel *Stakeholders*

Nama Kolom	Tipe Data	Panjang	Keterangan
<i>id</i>	<i>integer</i>	10	Primary Key
<i>name</i>	<i>varchar</i>	100	
<i>description</i>	<i>text</i>	10	
<i>introduction_id</i>	<i>integer</i>	10	Foreign Key

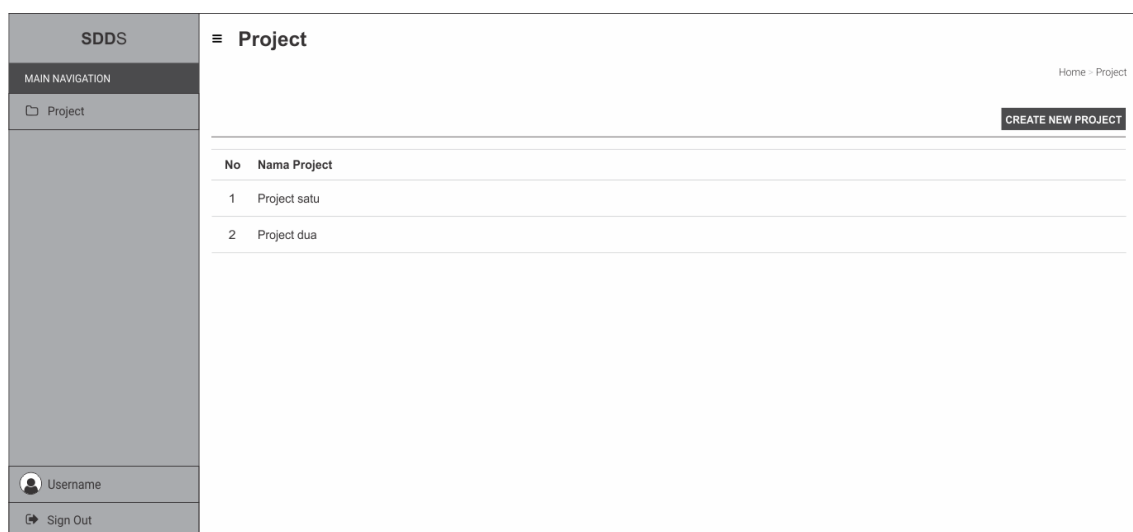
<i>created_at</i>	<i>timestamp</i>		
<i>updated_at</i>	<i>timestamp</i>		

### 3.5 Perancangan Antarmuka

Perancangan antarmuka sistem berguna untuk memberikan gambaran terkait sistem yang akan dibuat. Berikut rancangan antarmuka dari sistem ini.

#### a. Halaman Daftar Nama *Project*

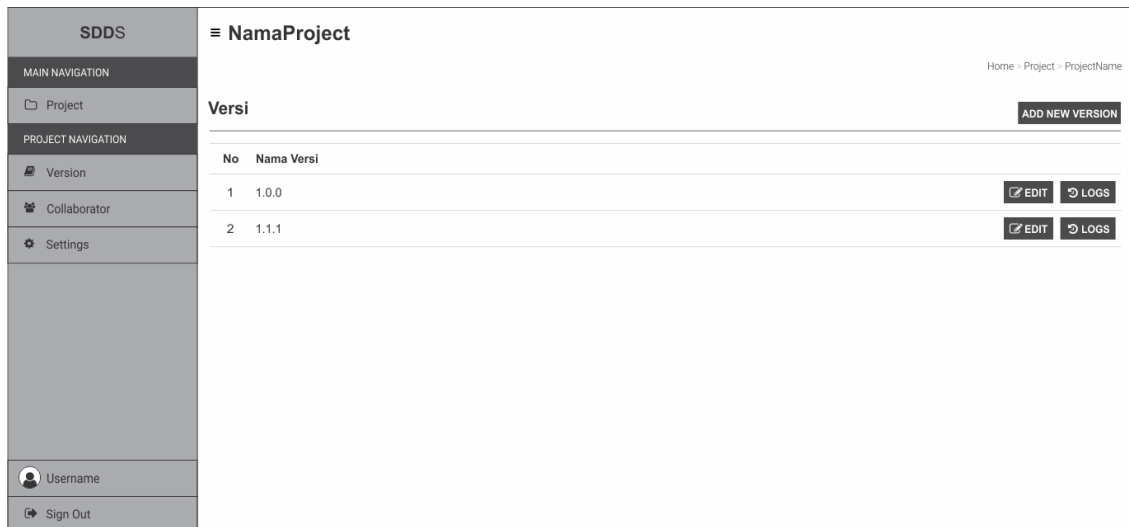
Halaman daftar nama *project* menampilkan daftar *project* yang sudah pernah ditambahkan sebelumnya.



Gambar 3.49 Desain antarmuka halaman daftar nama *project*

#### b. Halaman Daftar Versi

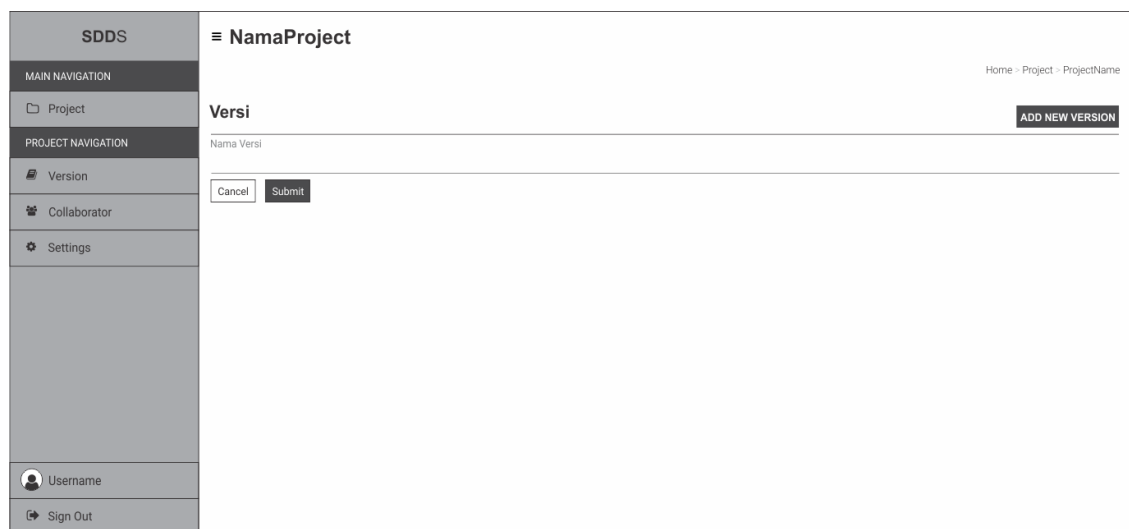
Halaman daftar nama versi menampilkan daftar versi yang sudah pernah ditambahkan sebelumnya.



Gambar 3.50 Halaman daftar versi

c. Halaman Tambah Versi

Halaman tambah versi menampilkan form untuk menambah versi baru.

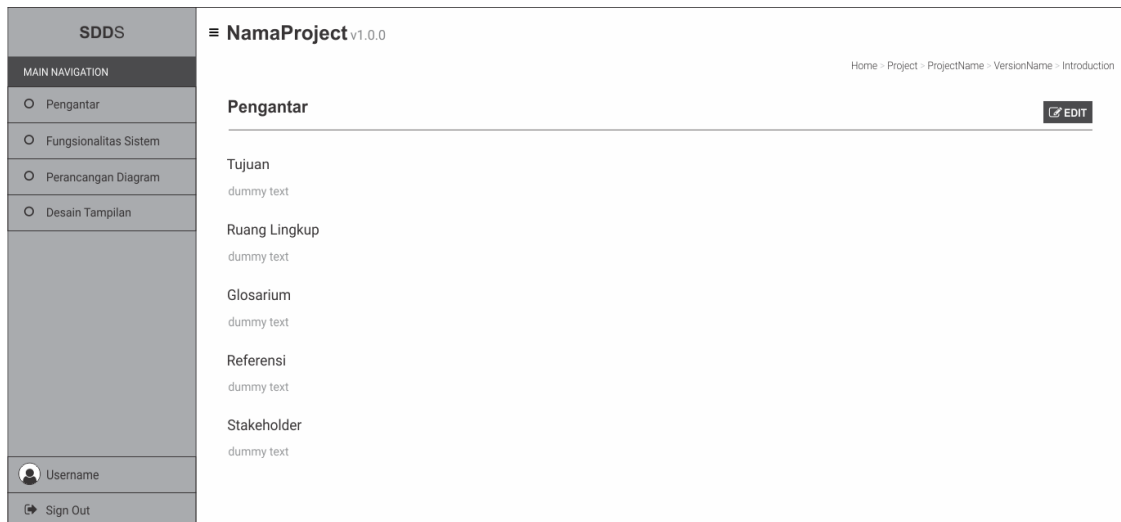


Gambar 3.51 Halaman tambah versi

d. Halaman Menu Pengantar

Halaman menu pengantar menampilkan data pengantar suatu *project* yaitu tujuan, ruang lingkup, glosarium, referensi, dan *stakeholder*.

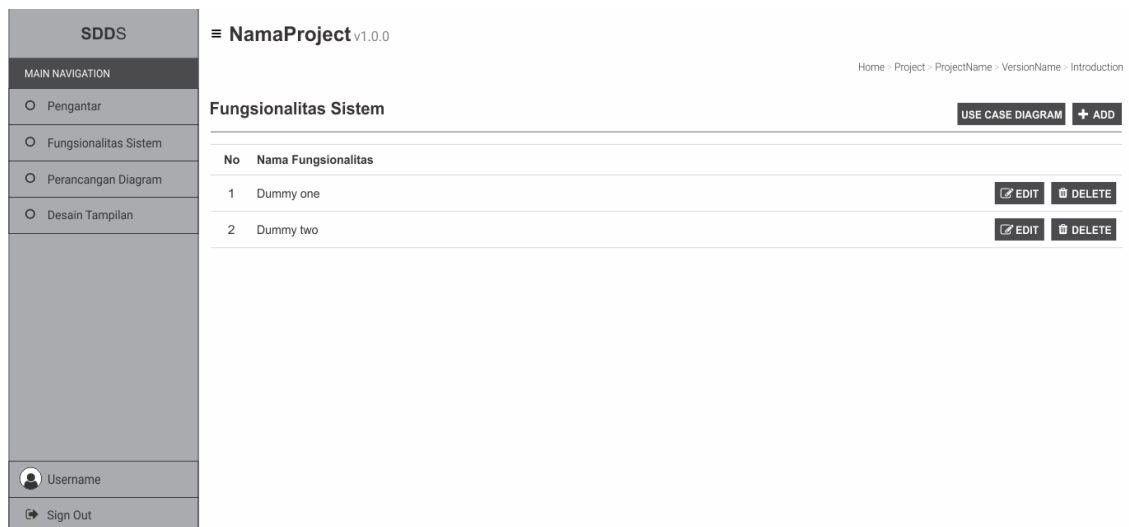




Gambar 3.52 Halaman menu pengantar

e. Halaman Menu Fungsionalitas Sistem

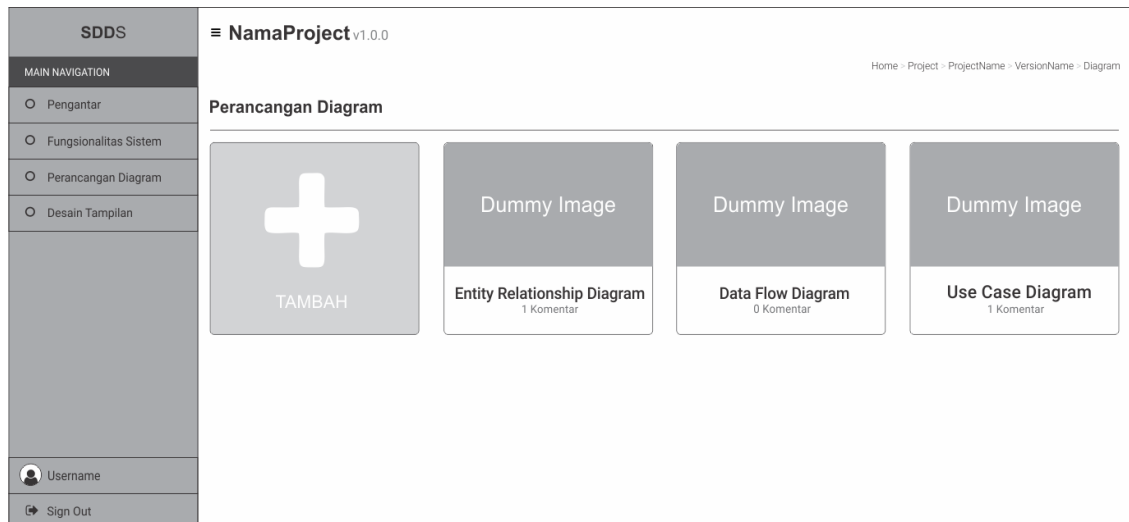
Halaman menu fungsionalitas sistem menampilkan daftar fungsionalitas sistem yang sudah pernah ditambahkan sebelumnya.



Gambar 3.53 Halaman menu fungsionalitas sistem

f. Halaman Menu Rancangan Diagram

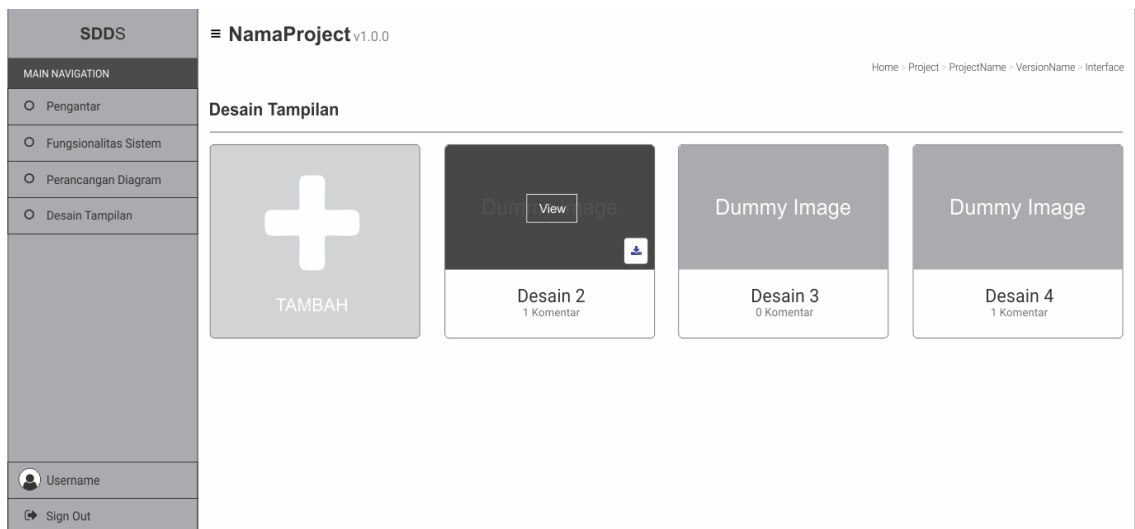
Halaman menu rancangan diagram menampilkan daftar jenis diagram yang sudah pernah ditambahkan sebelumnya.



Gambar 3.54 Halaman menu rancangan diagram

g. Halaman Menu Desain Antarmuka

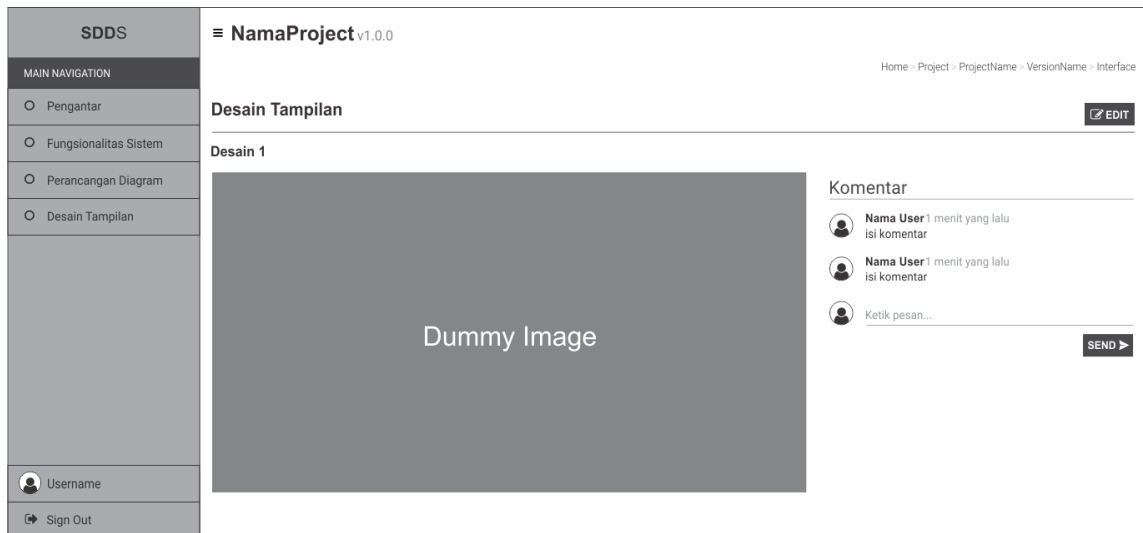
Halaman menu desain antarmuka menampilkan daftar desain antarmuka yang sudah pernah ditambahkan sebelumnya.



Gambar 3.55 Halaman desain antarmuka

h. Halaman Lihat Detail Desain Antarmuka

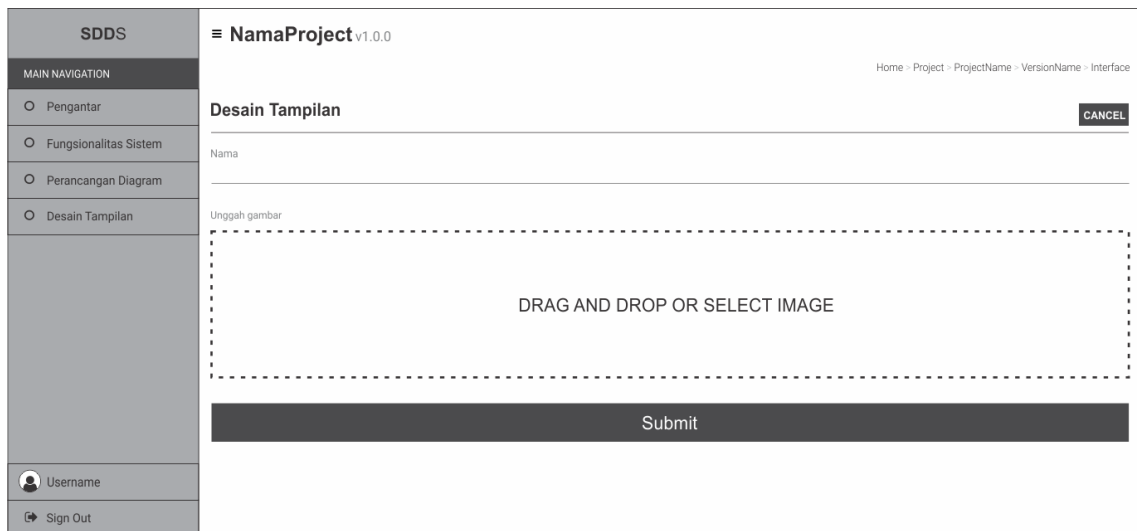
Halaman lihat detail desain antarmuka menampilkan detail dari desain antarmuka yang dipilih dari halaman daftar antarmuka. Pada halaman ini pengguna dapat memberikan komentar.



Gambar 3.56 Halaman lihat detail desain antarmuka

i. Halaman Tambah Desain Antarmuka

Halaman tambah desain antarmuka menampilkan form untuk menambah desain antarmuka baru.



Gambar 3.57 Halaman tambah desain antarmuka

### 3.6 Perancangan Pengujian

Rancangan pengujian merupakan bentuk pengujian sistem yang akan dilakukan setelah sistem selesai dibuat sesuai rancangan yang telah dibuat. Pengujian dilakukan dengan memberikan kuesioner kepada responden dan responden akan mengisi kuesioner yang telah disediakan sehingga mendapatkan persentase kelayakan sistem manajemen dokumen perancangan perangkat lunak. Berikut desain kuesioner yang akan digunakan dalam pengujian.

Tabel 3.53 Kuesioner Pengujian

No	Keterangan	STS	TS	N	S	SS
1	Fitur komentar pada sistem bermanfaat dalam kolaborasi antar tim pengembang untuk memberi saran maupun kritik terhadap perancangan sistem yang telah dibuat					
2	Fitur <i>log</i> /pencatatan aktivitas pada sistem bermanfaat dalam kolaborasi antar tim pengembang untuk mendapatkan informasi riwayat perubahan yang terjadi pada sistem.					
3	Sistem dapat menghasilkan dokumen perancangan perangkat lunak yang baik dan rapi sehingga mudah dipahami oleh pembacanya					
4	Sistem memiliki potensi dan manfaat yang besar sebagai aplikasi dokumentasi rekayasa perangkat lunak pada tahap perancangan					
5	Sistem memiliki tampilan yang menarik					
6	Sistem menyediakan mekanisme yang jelas dan mudah dimengerti bagi tim pengembang dalam membuat maupun mendapatkan informasi perancangan sistem					
7	Sistem memiliki tata letak elemen antarmuka yang memudahkan bagi pengguna					
8	Sistem memiliki ukuran dan warna elemen antarmuka yang nyaman bagi pengguna					

Berdasarkan kuesioner pengujian pada Tabel 3.50, berikut penjelasan dari pilihan jawaban pada kuesioner tersebut:

- a. STS : Sangat Tidak Setuju (1 Poin)
- b. TS : Tidak Setuju (2 Poin)
- c. N : Netral (3 Poin)
- d. S : Setuju (4 Poin)
- e. SS : Sangat Setuju (5 Poin)

Setelah pengujian selesai, dibutuhkan perhitungan untuk mendapatkan presentase kelayakan sistem yang dibuat untuk digunakan sebagai perangkat lunak untuk membuat dokumen perancangan perangkat lunak serta meningkatkan kualitas kolaborasi tim

pengembang. Rumus perhitungan hasil pengujian sistem dapat dilihat pada persamaan ( 3.1 ) dan persamaan ( 3.2)

$$\text{Rumus} = \text{Nilai Ideal} \times \text{Jumlah Sampel} \times \text{Jumlah Pertanyaan} \quad (3.1)$$

$$\text{Presentase} = \frac{\text{Skor Pengujian}}{\text{Nilai Maksimum}} \times 100\% \quad (3.2)$$

## BAB IV

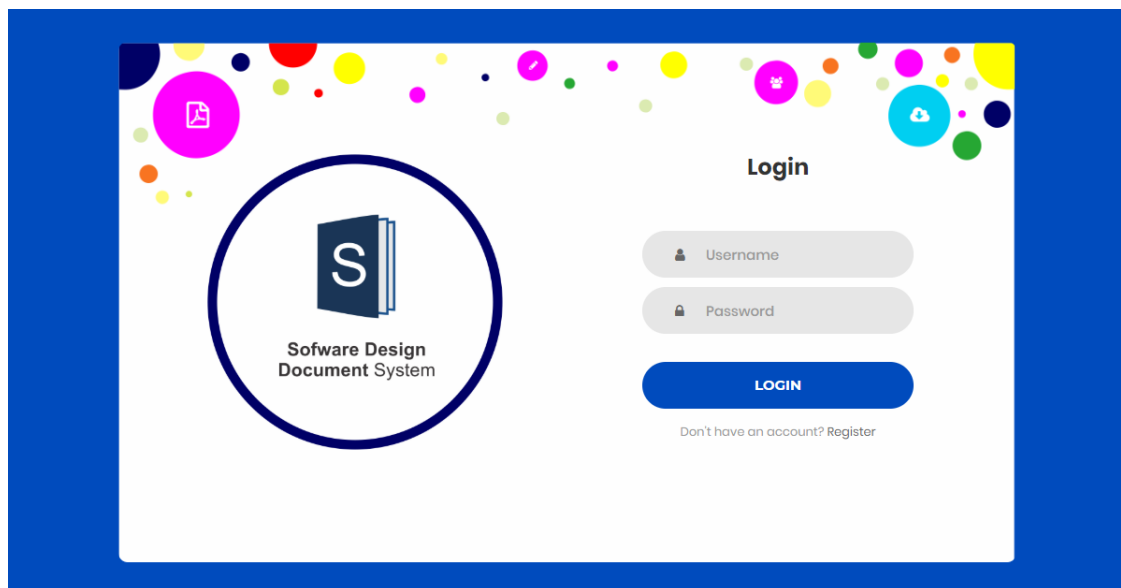
### HASIL DAN PEMBAHASAN

#### 4.1 Implementasi Sistem

Tahap implementasi merupakan tahapan mengimplementasikan semua perancangan yang telah dibuat pada tahap sebelumnya sehingga sistem siap untuk dijalankan dan diuji kelayakannya dari sisi pengguna. Pada pengembangan “Sistem Online Untuk Dokumen Perancangan Perangkat Lunak Bagi Perusahaan atau Tim Pengembang”, sistem dibuat menggunakan *Framework Laravel*. Berikut implementasi “Sistem Online Untuk Dokumen Perancangan Perangkat Lunak Bagi Perusahaan atau Tim Pengembang”.

##### 1. Halaman *Login*

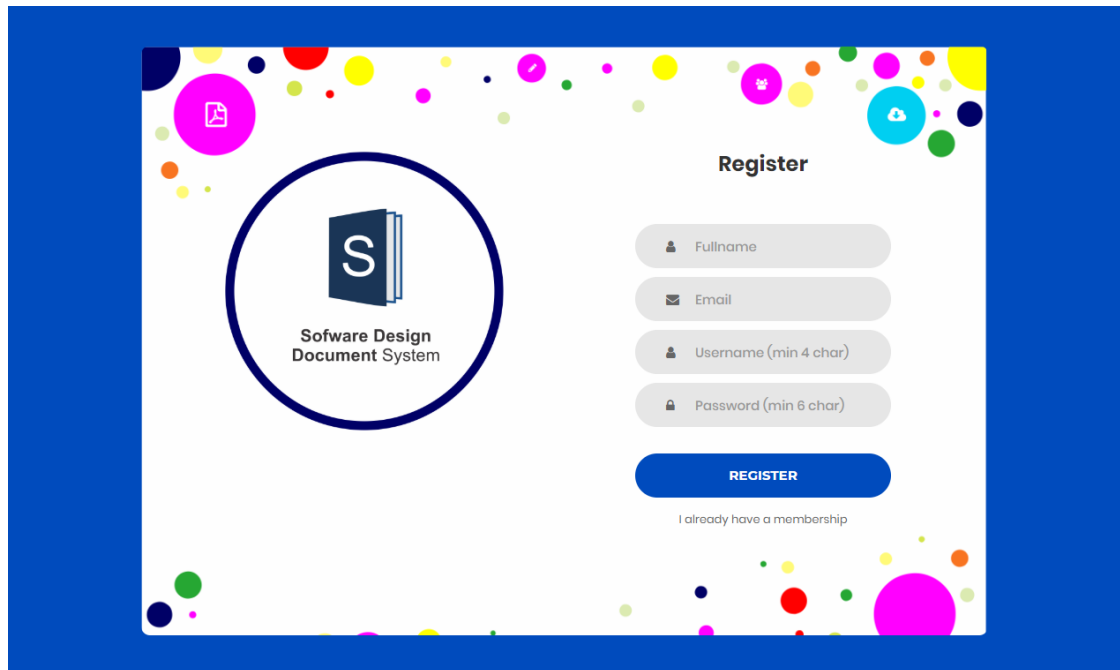
Halaman ini merupakan antarmuka yang muncul ketika pertama kali pengguna mengakses sistem dan juga sebagai form *login* untuk masuk agar dapat menggunakan fitur yang ada pada sistem. Pada proses *login*, pengguna diminta untuk memasukkan *username* dan *password*. Ketika pengguna menekan tombol *Sign In*, maka sistem akan melakukan autentifikasi apakah *username* dan *password* yang dimasukkan sudah sesuai atau belum. Apabila sesuai, maka sistem akan otomatis membuka halaman daftar *project* dan apabila salah, sistem akan otomatis kembali ke halaman *login*. Jika pengguna belum punya akun sebelumnya, maka pengguna dapat memilih pilihan *register*/daftar.



Gambar 4.1 Halaman *login*

## 2. Halaman *Register*

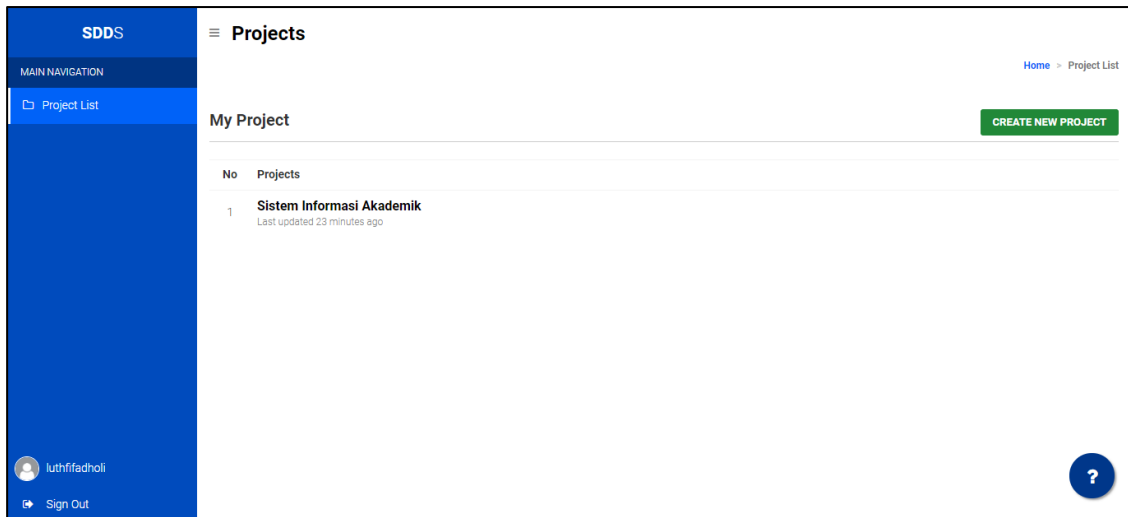
Pada halaman *register*, terdapat form yang harus diisi lengkap untuk mendaftar akun pada sistem. Adapun *form register* yang harus diisi yaitu nama lengkap, email, *username*, dan kata sandi. Untuk data *username* maksimal harus terdiri dari 10 karakter dan untuk data *password* minimal 6 karakter. Jika pengguna sudah memiliki akun sebelumnya, pengguna dapat memilih pilihan “*I already have a membership*” untuk beralih ke halaman *login*.



Gambar 4.2 Halaman *register*

## 3. Halaman Daftar *Project*

Pada halaman daftar *project*, terdapat daftar nama-nama *project* yang dimiliki oleh pengguna yang sudah berhasil *login*. Pada menu di sebelah kiri, terdapat menu daftar *project*, profil dan menu untuk *Sign Out*/keluar. Pada bagian kanan atas terdapat tombol tambah *project* baru untuk menambahkan *project* baru. Setiap pengguna yang menambahkan *project* baru, maka ia akan otomatis menjadi *project manager* pada *project* tersebut.



Gambar 4.3 Halaman daftar *project*

#### 4. Halaman Tambah *Project*

Pada halaman tambah *project* terdapat form yang harus diisi oleh pengguna untuk menambahkan *project* baru. Pada form tambah *project* terdapat 5 data, yaitu nama *project*, klien, tanggal mulai *project*, batas waktu *project*, dan biaya. Pada bagian kanan atas form, terdapat tombol *cancel* untuk membatalkan tambah form dan jika pengguna menekan tombol *cancel* maka akan otomatis membuka halaman daftar *project*.

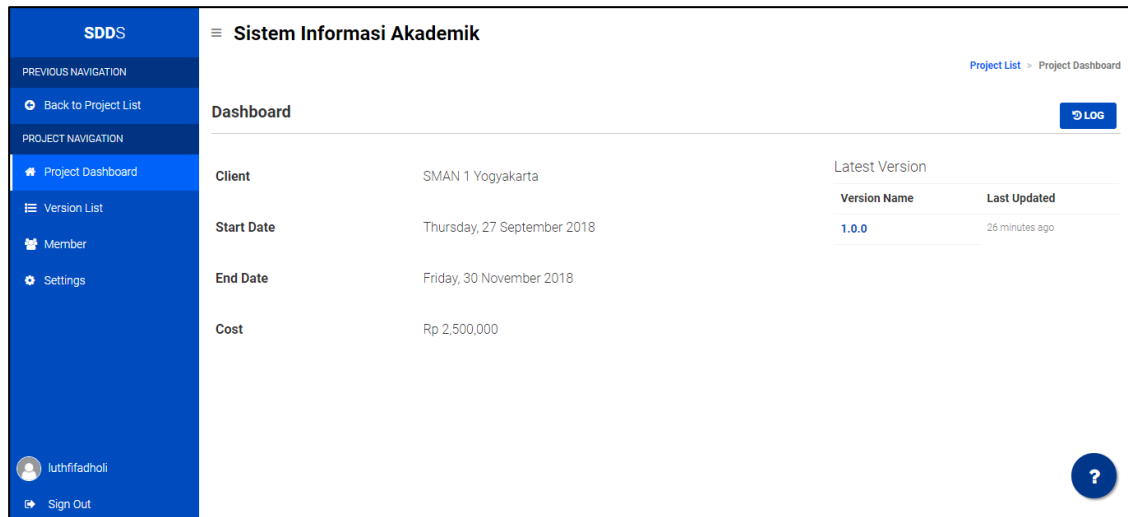
Gambar 4.4 Halaman tambah *project*

#### 5. Halaman *Dashboard Project*

Pada halaman *dashboard* akan tampil detail dari data *project* yang sudah dimasukkan pada form tambah *project* dan 5 versi terakhir dari *project* tersebut. Selain itu pilihan menu pada



sisi samping halaman juga berbeda dari halaman daftar *project*. Pada halaman ini terdapat 4 menu tambahan, yaitu menu halaman *dashboard*, menu daftar versi, menu anggota *project*, dan menu pengaturan. Dan disisi kanan atas terdapat tombol *log* untuk membuka halaman *log project*.



Gambar 4.5 Halaman *dashboard project*

## 6. Halaman *Log Project*

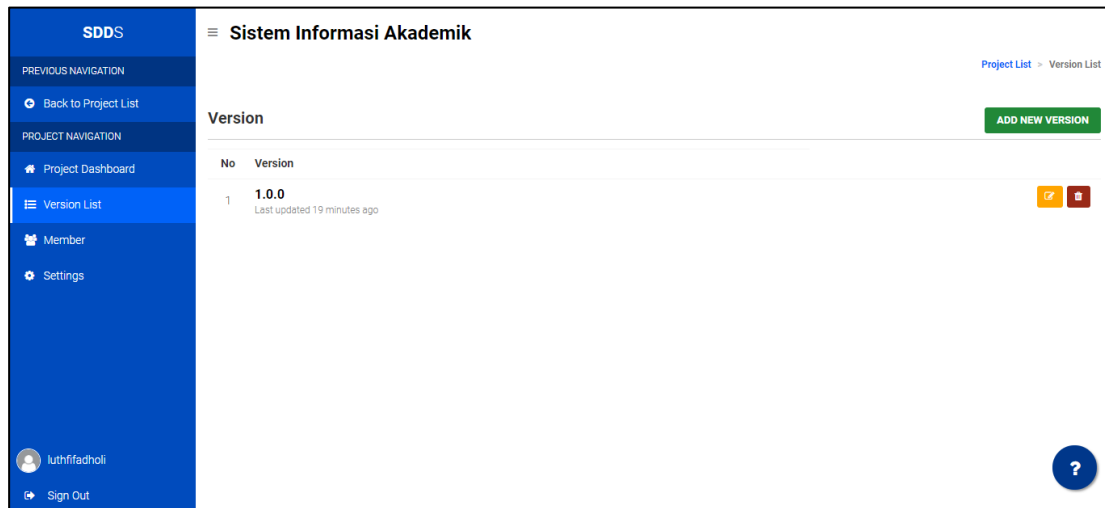
Halaman ini merupakan halaman yang berisi aktivitas yang dilakukan pengguna didalam suatu *project*. Pada halaman ini terdapat 4 data yang diambil dari *database*, yaitu aksi yang dilakukan pengguna, versi *project*, actor yang melakukan aktivitas, dan tanggal aktivitas tersebut.

No	Action	Version	Actor	Created
1	Add new interface design		luthffadholi	2018-09-16 16:43:22
2	Delete interface design		luthffadholi	2018-09-16 16:43:04
3	Add new parent diagram		luthffadholi	2018-09-16 16:38:42
4	Add new diagram type		luthffadholi	2018-09-16 16:38:12
5	Add new diagram type		luthffadholi	2018-09-16 16:37:16
6	Edit functionality		luthffadholi	2018-09-16 16:36:18
7	Add new functionality		luthffadholi	2018-09-16 16:36:09
8	Add stakeholder in introduction		luthffadholi	2018-09-16 16:35:11
9	Add stakeholder in introduction		luthffadholi	2018-09-16 16:34:40
10	Add reference in introduction		luthffadholi	2018-09-16 16:33:51

Gambar 4.6 Halaman *log project*

## 7. Halaman Daftar Versi

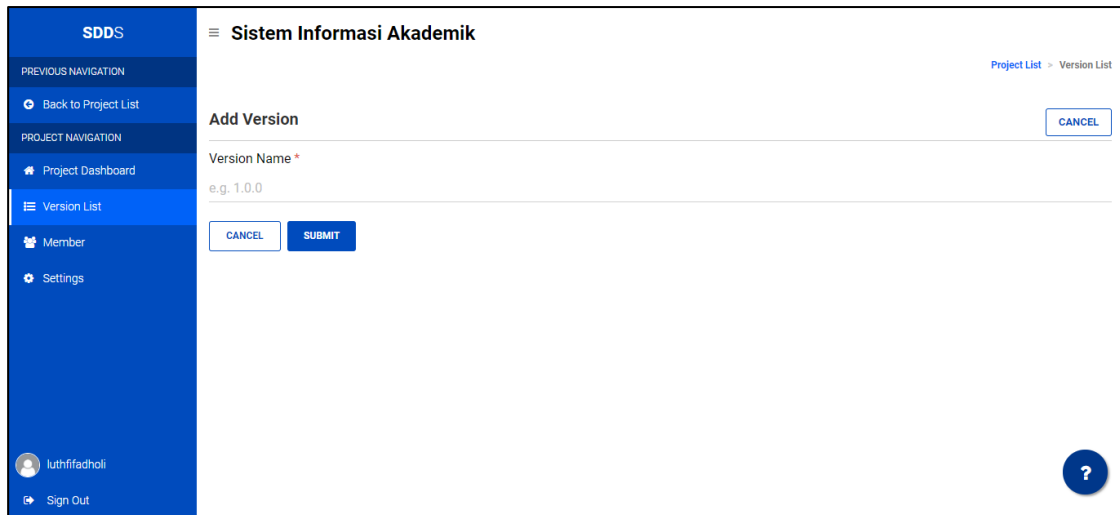
Halaman ini merupakan halaman yang berisi daftar versi dari suatu *project*. Setiap *project* dapat memiliki banyak versi. Pada bagian kanan atas halaman terdapat tombol tambah versi baru. Jika tombol ini di klik akan membuka form tambah versi baru dan akan menyembunyikan halaman daftar versi. Pada daftar versi, di setiap barisnya terdapat 3 tombol yaitu edit, hapus, dan *log* versi.



Gambar 4.7 Halaman daftar versi

## 8. Halaman Tambah Versi

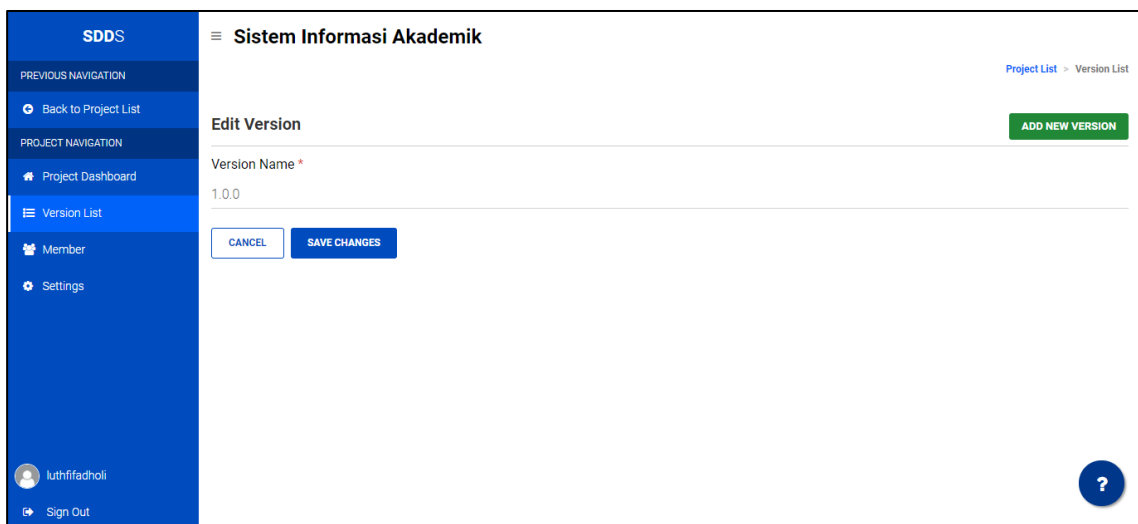
Halaman ini tampil ketika pengguna menekan tombol “*Add New Version*” pada halaman daftar versi. Pada halaman form tambah versi, pengguna diminta untuk memasukkan nama versi baru yang ingin dibuat. Apabila belum ada versi sebelumnya, maka sistem hanya menyimpan nama versi saja dengan detail yang masih kosong, tetapi jika saat pengguna menambahkan versi baru sudah ada versi sebelumnya, maka detail dari versi baru akan berisi sama persis seperti versi sebelumnya sehingga pengguna tidak perlu menambahkan detail dari awal lagi.



Gambar 4.8 Halaman tambah versi

## 9. Halaman Edit Versi

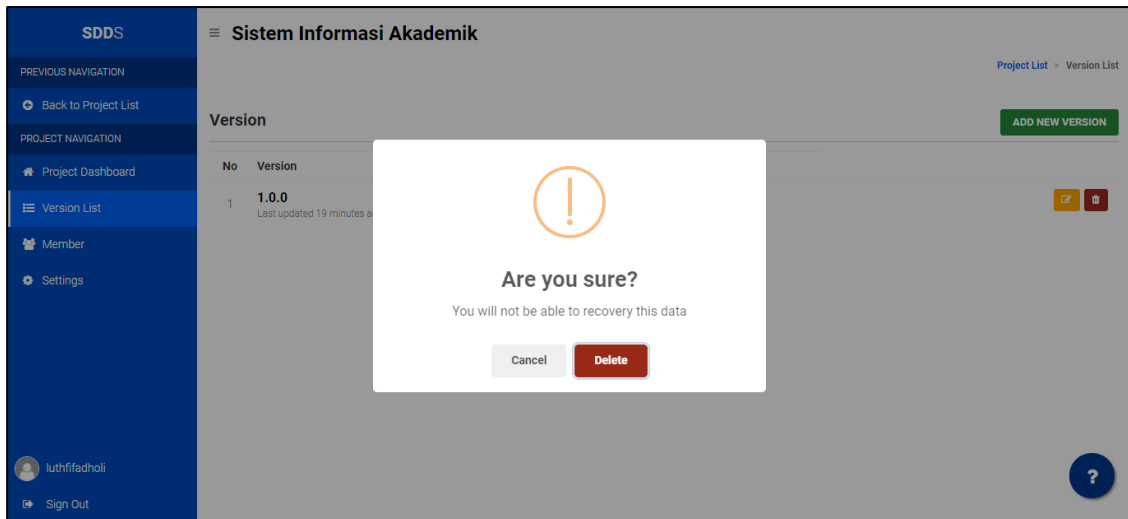
Halaman ini tampil ketika pengguna menekan tombol edit pada halaman daftar versi. Pada halaman ini pengguna dapat mengubah nama versi sesuai data yang diinginkan.



Gambar 4.9 Halaman edit versi

## 10. Konfirmasi Hapus Versi

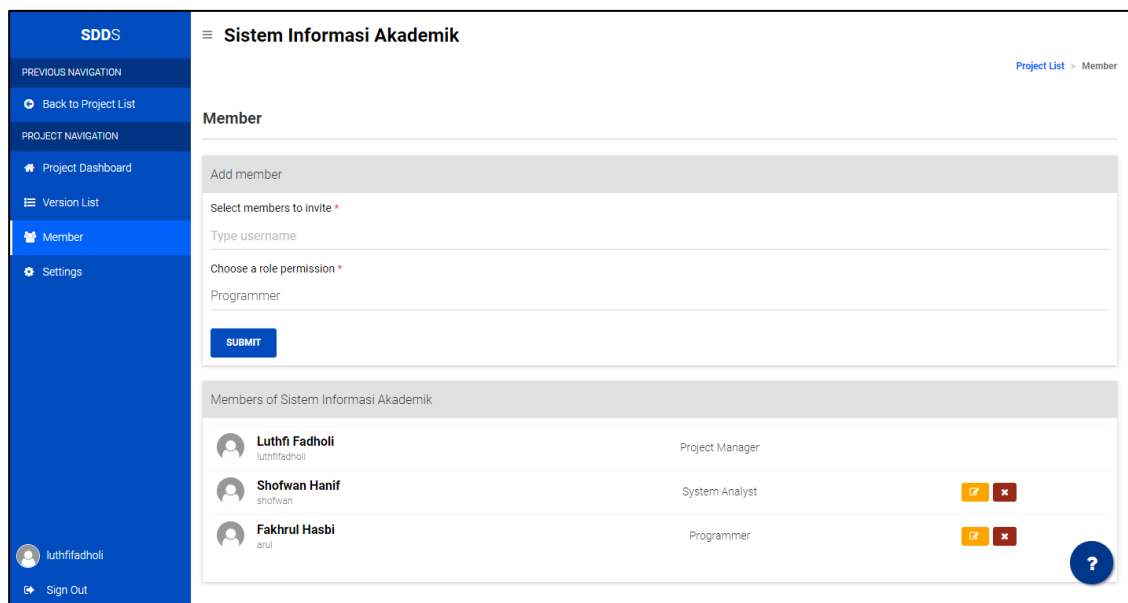
Apabila pengguna menekan tombol hapus versi, maka akan muncul peringatan untuk konfirmasi apakah pengguna benar-benar ingin menghapus versi yang dipilih atau tidak. Fitur ini bertujuan untuk mencegah kesalahan pengguna dalam menghapus data.



Gambar 4.10 Konfirmasi hapus versi

## 11. Halaman Daftar Anggota

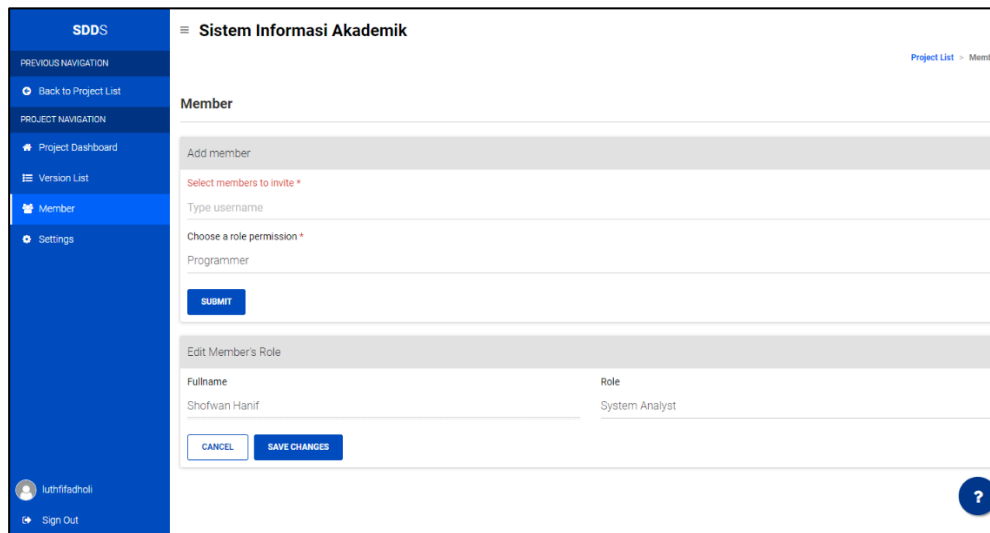
Halaman daftar anggota merupakan halaman yang berisi daftar anggota suatu *project*. Pada daftar anggota terdapat 2 tombol aksi yaitu edit dan hapus. Tombol edit digunakan untuk mengedit role *user* yang dipilih. Pada halaman ini juga terdapat form tambah anggota yang hanya dapat digunakan oleh *project manager*. Terdapat 3 pilihan jenis role *user* pada form tambah yang dapat digunakan yaitu *system analyst*, *UI designer*, dan *programmer*.



Gambar 4.11 Halaman daftar anggota

## 12. Halaman Edit Anggota

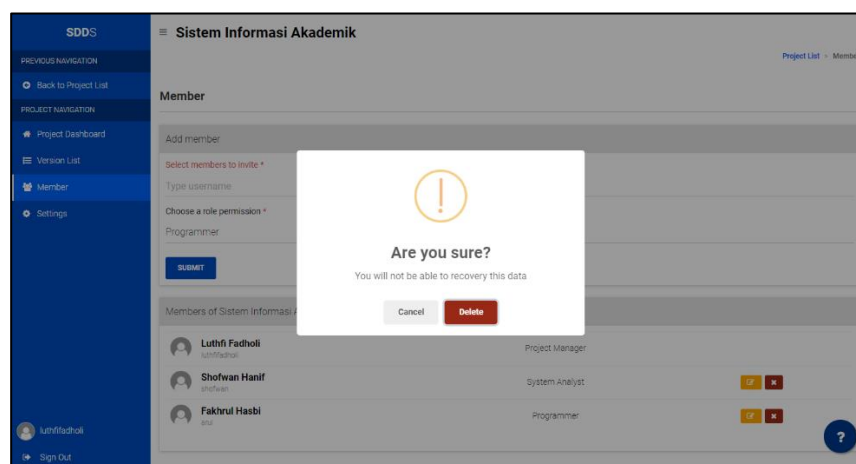
Form edit anggota muncul ketika pengguna menekan tombol edit pada baris *user* yang ingin diedit. Pada form edit anggota akan tampil nama *user* dan role-nya tetapi hanya role saja yang dapat diedit. Apabila menekan tombol *cancel*, sistem akan menyembunyikan form edit dan menampilkan daftar anggota *project*.



Gambar 4.12 Halaman edit anggota

## 13. Konfirmasi Hapus Anggota

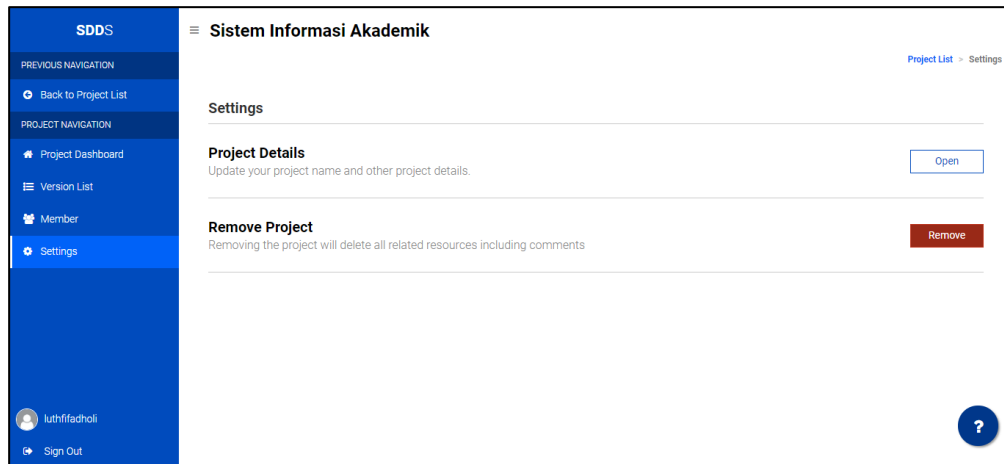
Apabila pengguna menekan tombol hapus anggota, maka akan muncul peringatan untuk konfirmasi apakah pengguna benar-benar ingin menghapus anggota yang dipilih atau tidak. Fitur ini bertujuan untuk mencegah kesalahan pengguna dalam menghapus data.



Gambar 4.13 Konfirmasi hapus anggota

#### 14. Halaman Pengaturan

Halaman ini berfungsi untuk mengatur detail *project* dan menghapus *project*. Apabila pengguna menekan tombol *Open* maka akan tampil form edit detail *project*. Halaman pengaturan hanya bisa diakses oleh pengguna dengan role *project manager*



Gambar 4.14 Halaman pengaturan

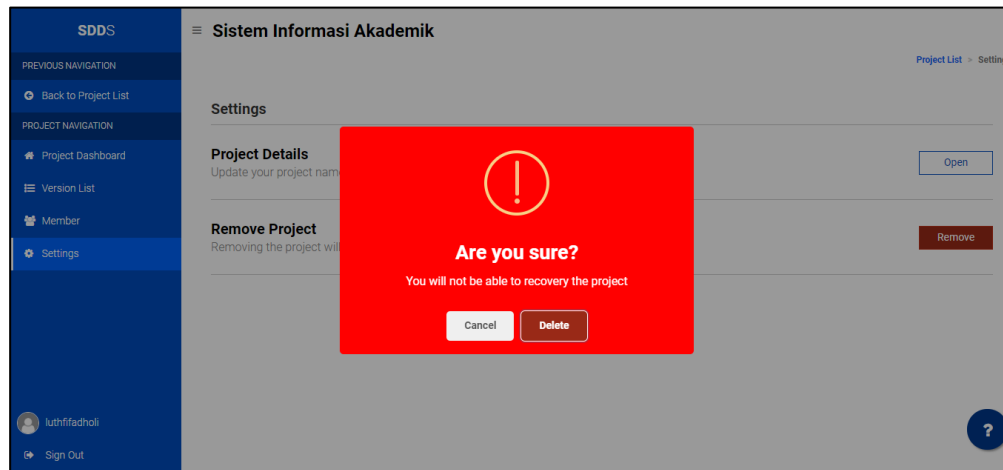
#### 15. Form Edit Detail *Project*

Form edit detail *project* muncul ketika pengguna menekan tombol *open*. Pada form edit detail *project* akan tampil detail dari *project* yaitu nama *project*, klien *project*, tanggal mulai *project*, tenggat waktu *project*, dan biaya *project*.

Gambar 4.15 Form edit detail *project*

## 16. Konfirmasi Hapus *Project*

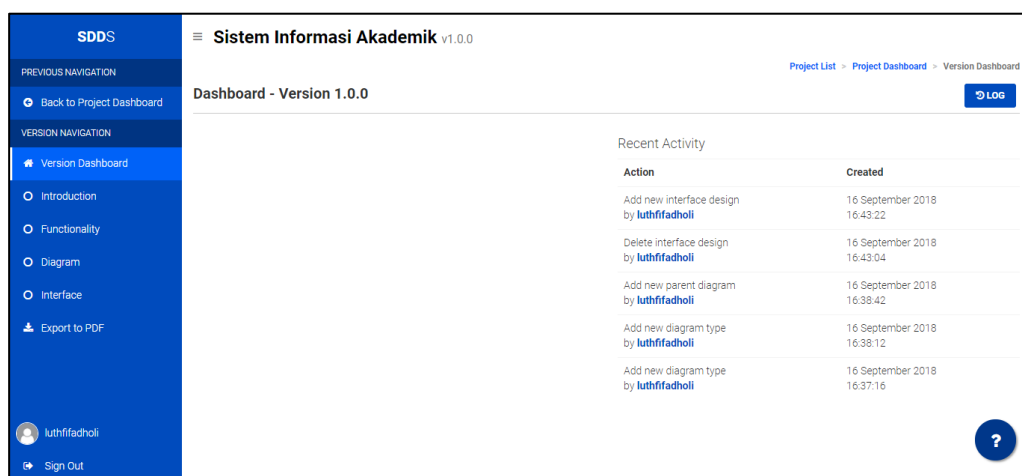
Apabila pengguna menekan tombol hapus project, maka akan muncul peringatan untuk konfirmasi apakah pengguna benar-benar ingin menghapus project atau tidak. Apabila pengguna memilih tombol hapus pada peringatan hapus, maka menghapus project beserta seluruh data didalamnya seperti versi, pengantar, fungsionalitas sistem, dan lain-lain. Fitur ini bertujuan untuk mencegah kesalahan pengguna dalam menghapus data.



Gambar 4.16 Konfirmasi hapus *project*

## 17. Halaman *Dashboard* Versi

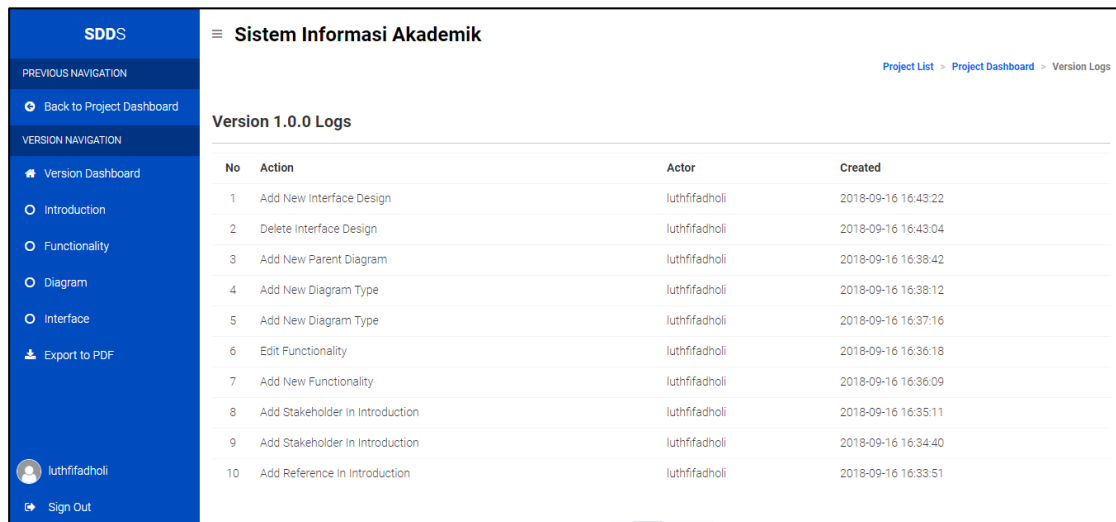
Halaman ini muncul ketika pengguna menekan salah satu nama versi pada daftar versi. Pada halaman dashboard versi akan tampil 5 aktivitas terakhir yang dilakukan anggota didalam versi tersebut. Selain itu ada juga tombol untuk membuka halaman *log* versi.



Gambar 4.17 Halaman *dashboard* versi

## 18. Halaman *Log* Versi

Halaman ini merupakan halaman yang berisi aktivitas yang dilakukan anggota *project* dalam suatu versi *project*. Pada halaman ini terdapat 3 data yang diambil dari *database*, yaitu aksi yang dilakukan pengguna, aktor yang melakukan aktivitas, dan waktu aktivitas tersebut.



The screenshot shows a web application interface for 'Sistem Informasi Akademik'. The left sidebar contains navigation options: 'Back to Project Dashboard', 'Version Dashboard', 'Introduction', 'Functionality', 'Diagram', 'Interface', 'Export to PDF', and 'Sign Out'. The main content area displays 'Version 1.0.0 Logs' with a table of activities.

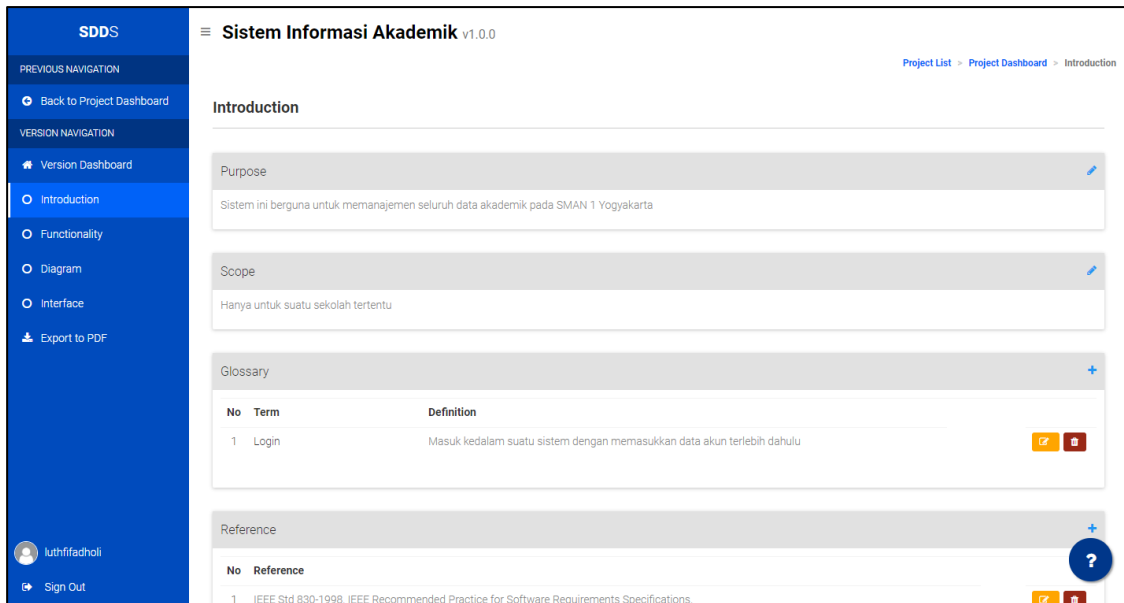
No	Action	Actor	Created
1	Add New Interface Design	luthiffadholi	2018-09-16 16:43:22
2	Delete Interface Design	luthiffadholi	2018-09-16 16:43:04
3	Add New Parent Diagram	luthiffadholi	2018-09-16 16:38:42
4	Add New Diagram Type	luthiffadholi	2018-09-16 16:38:12
5	Add New Diagram Type	luthiffadholi	2018-09-16 16:37:16
6	Edit Functionality	luthiffadholi	2018-09-16 16:36:18
7	Add New Functionality	luthiffadholi	2018-09-16 16:36:09
8	Add Stakeholder In Introduction	luthiffadholi	2018-09-16 16:35:11
9	Add Stakeholder In Introduction	luthiffadholi	2018-09-16 16:34:40
10	Add Reference In Introduction	luthiffadholi	2018-09-16 16:33:51

Gambar 4.18 Halaman *log* versi

## 19. Halaman Menu Pengantar

Halaman ini merupakan implementasi dari standar SDD IEEE 1016-2009 yaitu deskripsi SDD. Pada halaman ini terdapat 5 jenis data yaitu tujuan, ruang lingkup, glossarium, referensi, dan *stakeholder*. Untuk menambah data, pengguna dapat menekan tombol (+) pada sudut kanan atas di setiap *header* data. Selain menambah, pengguna juga dapat mengedit dan menghapus data yang ada. Hanya pengguna dengan role project manager dan system analyst yang dapat menambah, mengedit, dan menghapus data yang ada pada halaman menu pengantar.

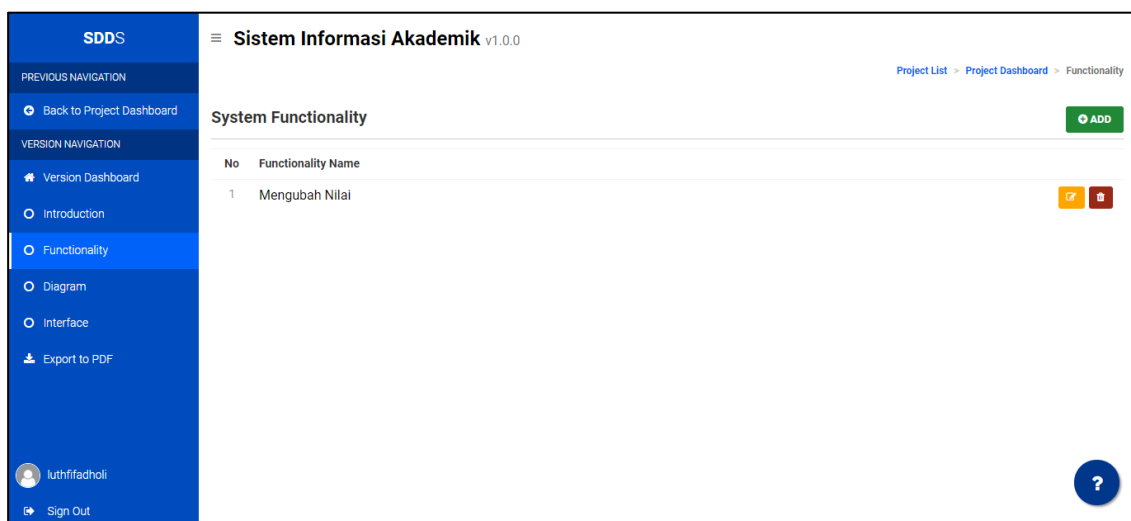




Gambar 4.19 Halaman menu pengantar

## 20. Halaman Menu Fungsionalitas Sistem

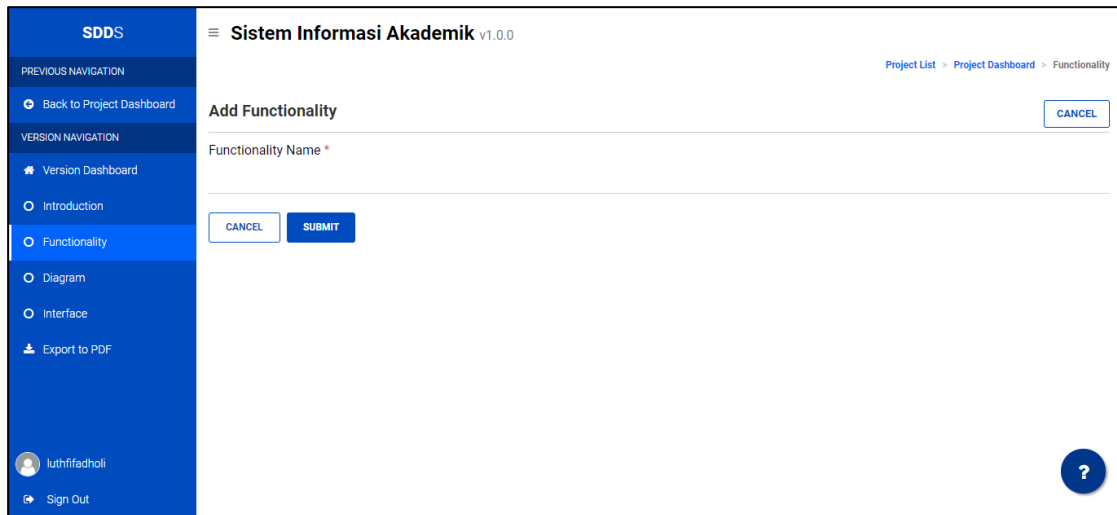
Pada halaman menu fungsionalitas sistem menampilkan daftar fungsionalitas sistem dari suatu project. Pada halaman ini juga terdapat 3 tombol yaitu tombol tambah, edit, dan hapus. Jika pengguna menekan salah satu nama fungsionalitas sistem yang ada, maka sistem akan membuka halaman daftar diagram yang berelasi dengan fungsionalitas yang dipilih.



Gambar 4.20 Halaman menu fungsionalitas sistem

## 21. Halaman Tambah Fungsionalitas Sistem

Halaman ini muncul jika pengguna menekan tombol tambah pada halaman daftar fungsionalitas sistem. Pada form tambah fungsionalitas pengguna diminta memasukkan nama fungsionalitas baru yang ingin dibuat.

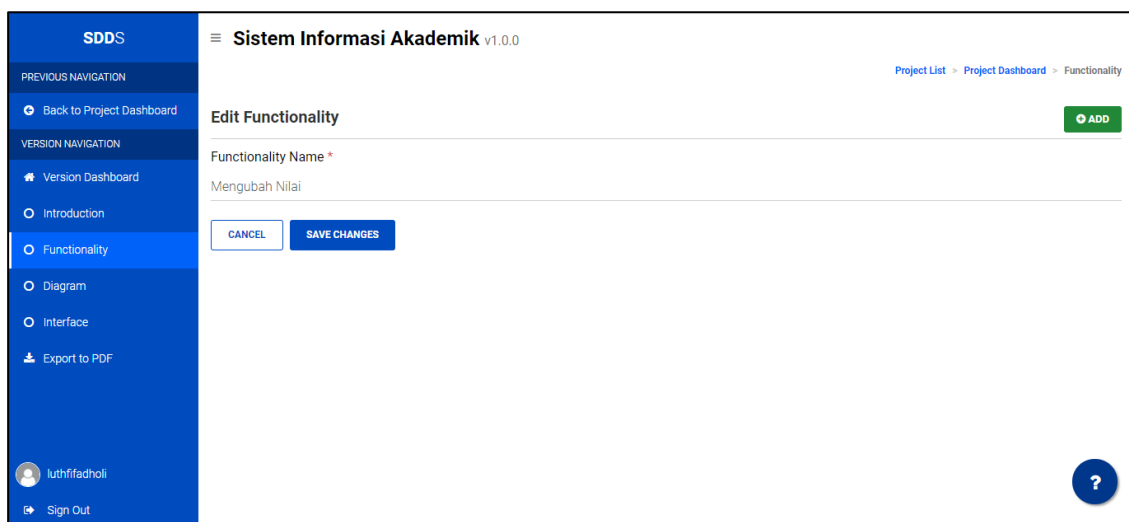


The screenshot shows the 'Add Functionality' page in the SDDS system. The page has a blue sidebar on the left with navigation options: 'Back to Project Dashboard', 'Version Dashboard', 'Introduction', 'Functionality' (selected), 'Diagram', 'Interface', and 'Export to PDF'. The main content area is titled 'Add Functionality' and contains a form with a 'Functionality Name \*' input field. Below the input field are 'CANCEL' and 'SUBMIT' buttons. The top right corner shows the breadcrumb 'Project List > Project Dashboard > Functionality' and a 'CANCEL' button. The bottom right corner has a blue circular help icon with a question mark.

Gambar 4.21 Halaman tambah fungsionalitas sistem

## 22. Halaman Edit Fungsionalitas

Halaman ini muncul ketika pengguna menekan tombol edit pada salah satu fungsionalitas sistem. Apabila ingin membatalkan mengedit, pengguna cukup memilih tombol cancel pada form edit.

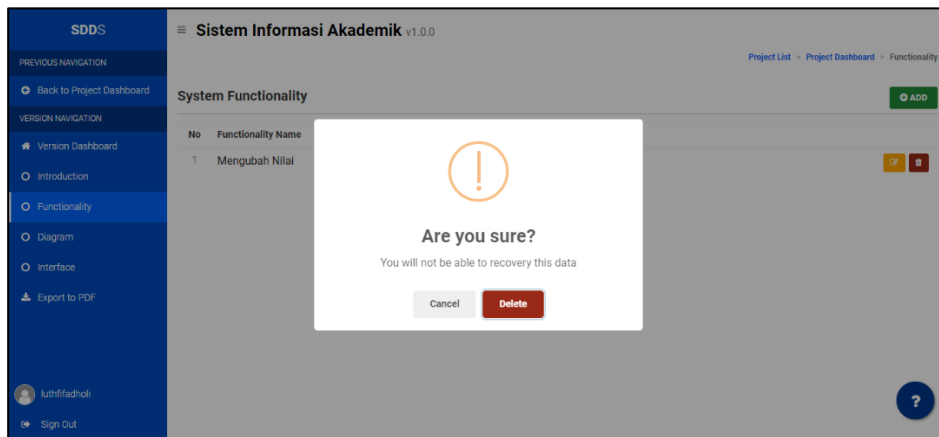


The screenshot shows the 'Edit Functionality' page in the SDDS system. The page has a blue sidebar on the left with navigation options: 'Back to Project Dashboard', 'Version Dashboard', 'Introduction', 'Functionality' (selected), 'Diagram', 'Interface', and 'Export to PDF'. The main content area is titled 'Edit Functionality' and contains a form with a 'Functionality Name \*' input field. Below the input field are 'CANCEL' and 'SAVE CHANGES' buttons. The top right corner shows the breadcrumb 'Project List > Project Dashboard > Functionality' and an 'ADD' button. The bottom right corner has a blue circular help icon with a question mark.

Gambar 4.22 Halaman edit fungsionalitas sistem

### 23. Konfirmasi Hapus Fungsionalitas Sistem

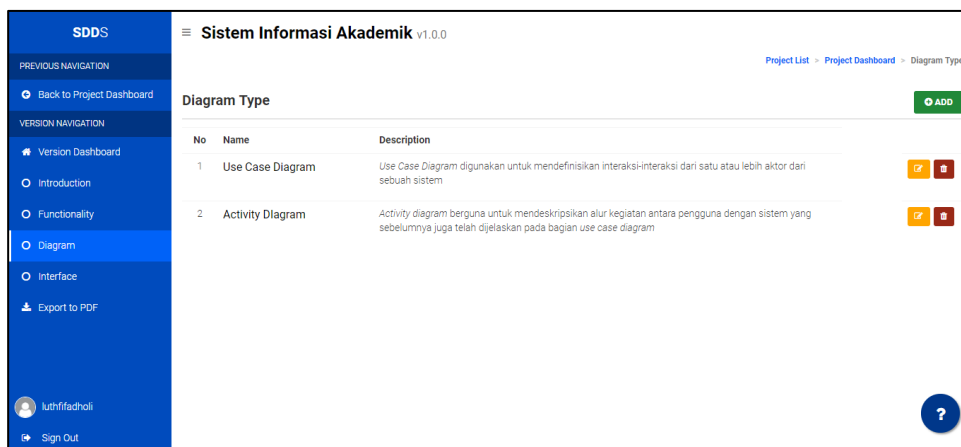
Apabila pengguna menekan tombol hapus fungsionalitas sistem, maka akan muncul peringatan untuk konfirmasi apakah pengguna benar-benar ingin menghapus fungsionalitas yang dipilih atau tidak. Fitur ini bertujuan untuk mencegah kesalahan pengguna dalam menghapus data.



Gambar 4.23 Konfirmasi hapus fungsionalitas sistem

### 24. Halaman Daftar Tipe Diagram

Pada halaman utama menu diagram akan tampil daftar jenis diagram yang nantinya jenis diagram ini akan dipakai saat pengguna akan menambah diagram baru. Pada halaman ini juga terdapat 3 tombol yaitu tombol tambah, edit, dan hapus. Jika pengguna menekan salah satu jenis diagram yang ada, maka sistem akan membuka halaman *parent* diagram dengan jenis diagram sesuai jenis diagram yang dipilih.



Gambar 4.24 Halaman menu diagram

## 25. Halaman Tambah Jenis Diagram

Halaman ini muncul ketika pengguna memilih tombol tambah pada bagian kanan atas halaman daftar jenis diagram. Pada form ini pengguna diminta memasukkan nama jenis diagram dan deskripsi.

Gambar 4.25 Halaman tambah jenis diagram

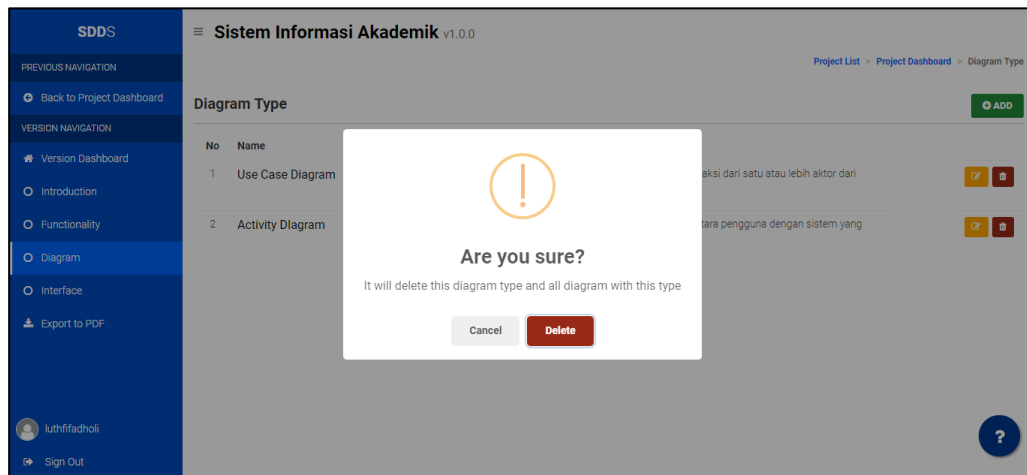
## 26. Halaman Edit Jenis Diagram

Halaman ini muncul ketika pengguna memilih tombol edit pada salah satu tipe diagram. Apabila pengguna ingin membatalkan edit, pengguna cukup memilih tombol *cancel*.

Gambar 4.26 Halaman edit tipe diagram

## 27. Konfirmasi Hapus Jenis Diagram

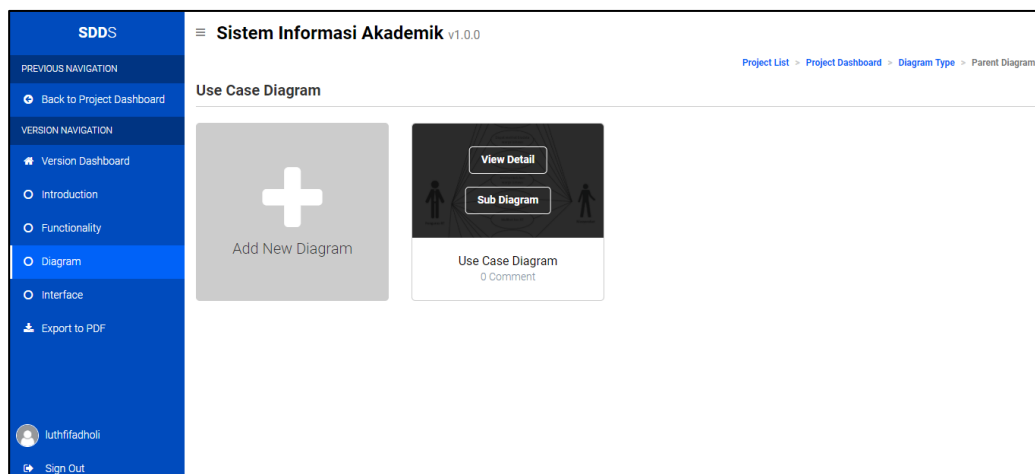
Apabila pengguna menekan tombol hapus tipe diagram, maka akan muncul peringatan untuk konfirmasi apakah pengguna benar-benar ingin menghapus tipe diagram yang dipilih atau tidak. Fitur ini bertujuan untuk mencegah kesalahan pengguna dalam menghapus data.



Gambar 4.27 Konfirmasi hapus jenis diagram

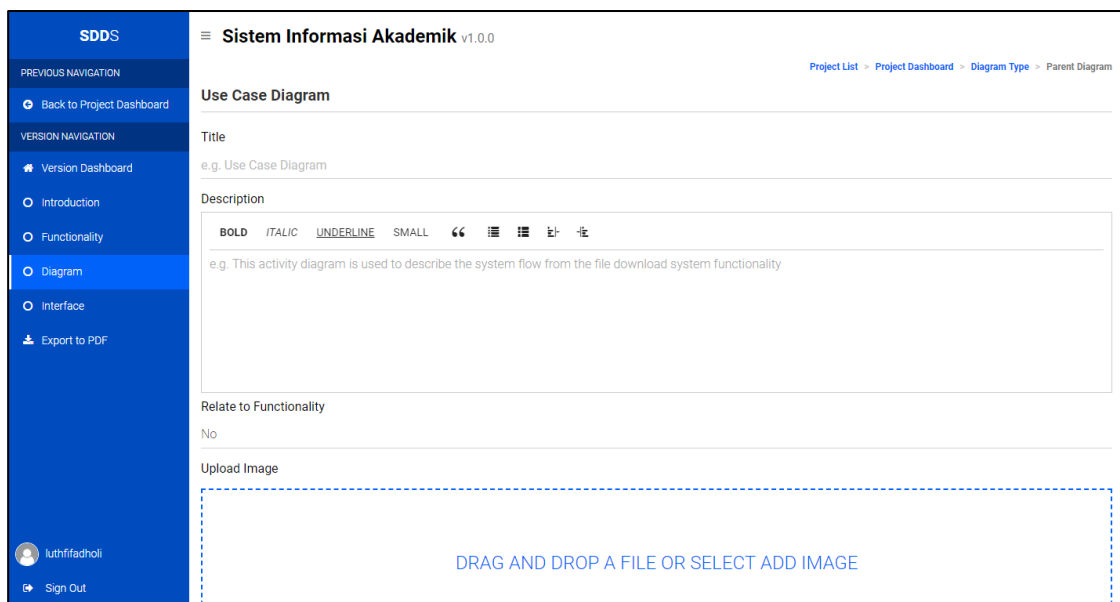
## 28. Halaman Daftar *Parent* Diagram

Halaman ini merupakan implementasi dari standar SDD IEEE 1016-2009 yaitu tampilan desain diagram. Diagram pada sistem ini dapat meng-*handle* baik *single* diagram maupun *multilevel* diagram sehingga dibagi menjadi 2 jenis diagram untuk memudahkan yaitu *parent* diagram dan *sub* diagram. Halaman ini muncul ketika pengguna memilih salah satu jenis diagram pada halaman daftar jenis diagram.



Gambar 4.28 Halaman daftar *parent* diagram29. Halaman Tambah *Parent* Diagram

Ketika pengguna memilih tombol tambah pada halaman daftar *parent* diagram maka akan menyembunyikan halaman daftar *parent* diagram dan akan menampilkan halaman form tambah *parent* diagram. Pada form tambah *parent* diagram pengguna diminta memasukkan 4 jenis data, yaitu judul diagram, deskripsi diagram, relasi dengan fungsionalitas mana, dan gambar diagram.



The screenshot shows a web application interface for adding a parent diagram. The left sidebar contains navigation options: 'Back to Project Dashboard', 'Version Dashboard', 'Introduction', 'Functionality', 'Diagram' (selected), 'Interface', and 'Export to PDF'. The main content area is titled 'Sistem Informasi Akademik v1.0.0' and 'Use Case Diagram'. It features a 'Title' field with the example 'e.g. Use Case Diagram', a 'Description' field with a rich text editor (including Bold, Italic, Underline, and Small options) and the example 'e.g. This activity diagram is used to describe the system flow from the file download system functionality'. Below the description is a 'Relate to Functionality' section with a radio button for 'No'. At the bottom is an 'Upload Image' section with a dashed box and the text 'DRAG AND DROP A FILE OR SELECT ADD IMAGE'. The user's name 'luthiffadholi' and a 'Sign Out' button are visible in the bottom left corner.

Gambar 4.29 Halaman tambah *parent* diagram

Pada form ini pengguna juga dapat memilih apakah diagram yang dibuat ingin direlasikan dengan fungsionalitas sistem yang ada atau tidak. Jika pengguna memilih pilihan *yes* mana akan tampil pilihan masukan untuk memilih fungsionalitas sistem. Berikut dapat dilihat pada Gambar 4.30

**SDDS** Sistem Informasi Akademik v1.0.0

Project List > Project Dashboard > Diagram Type > Parent Diagram

**Use Case Diagram**

Title  
e.g. Use Case Diagram

Description  
e.g. This activity diagram is used to describe the system flow from the file download system functionality

Relate to Functionality  
Yes

Choose the functionality  
Mengubah Nilai

Upload Image

luthiffadholi Sign Out

Gambar 4.30 Halaman tambah *parent* diagram ketika memilih pilihan *yes*

### 30. Halaman Daftar *Sub* Diagram

Halaman ini dapat diakses dengan memilih tombol *sub* diagram pada salah satu *parent* diagram pada halaman daftar *parent* diagram. Pada halaman ini akan tampil daftar *sub* diagram atau diagram turunan dari *parent* diagram yang sudah dipilih sebelumnya. Selain itu pengguna juga bisa menambahkan *sub* diagram baru.

**SDDS** Sistem Informasi Akademik v1.0.0

Project List > Project Dashboard > Diagram Type > Parent Diagram > Sub Diagram

**Use Case Diagram**

Use Case Diagram  
0 Comments

**Sub Diagram**

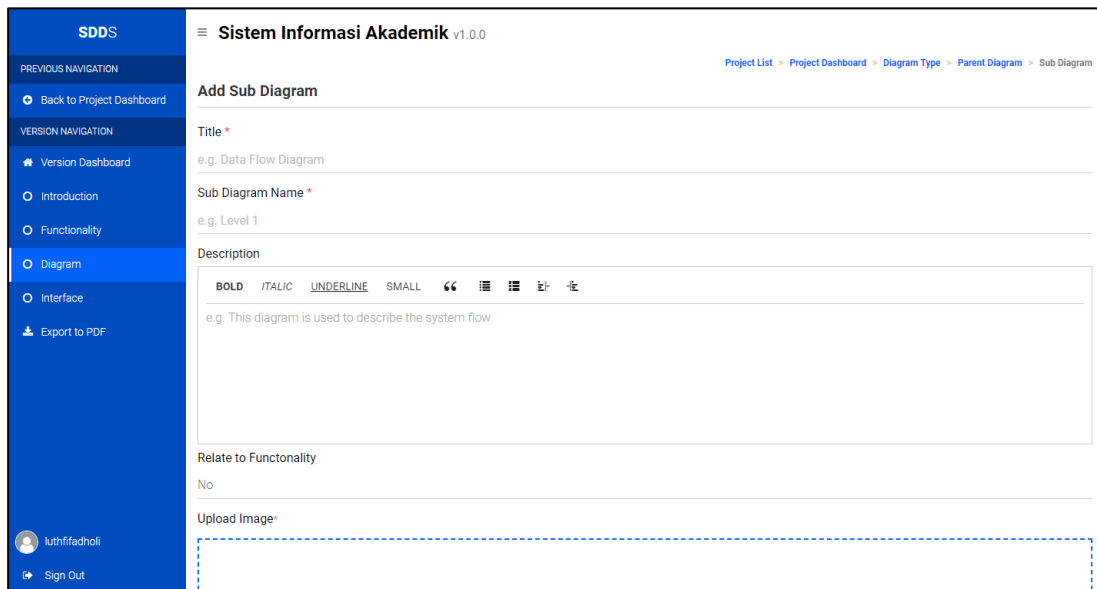
Add New Sub Diagram

luthiffadholi Sign Out

Gambar 4.31 Halaman daftar *sub* diagram

### 31. Halaman Tambah *Sub* Diagram

Halaman tambah *sub* diagram dapat diakses dengan menekan tombol tambah pada halaman daftar *sub* diagram. Pada halaman form tambah ini pengguna diminta memasukkan 5 data yaitu judul diagram, nama *sub* diagram, deskripsi diagram, relasi dengan fungsionalitas mana, dan gambar diagram.



The screenshot shows the 'Add Sub Diagram' form in the SDDS system. The form is titled 'Add Sub Diagram' and is part of the 'Sistem Informasi Akademik v1.0.0' interface. The form includes the following fields:

- Title \***: e.g. Data Flow Diagram
- Sub Diagram Name \***: e.g. Level 1
- Description**: A rich text editor with a toolbar containing options for BOLD, ITALIC, UNDERLINE, SMALL, and various alignment and list options. The description field contains the text: e.g. This diagram is used to describe the system flow.
- Relate to Functionality**: A dropdown menu with the value 'No' selected.
- Upload Image\***: A dashed blue box indicating where to upload an image.

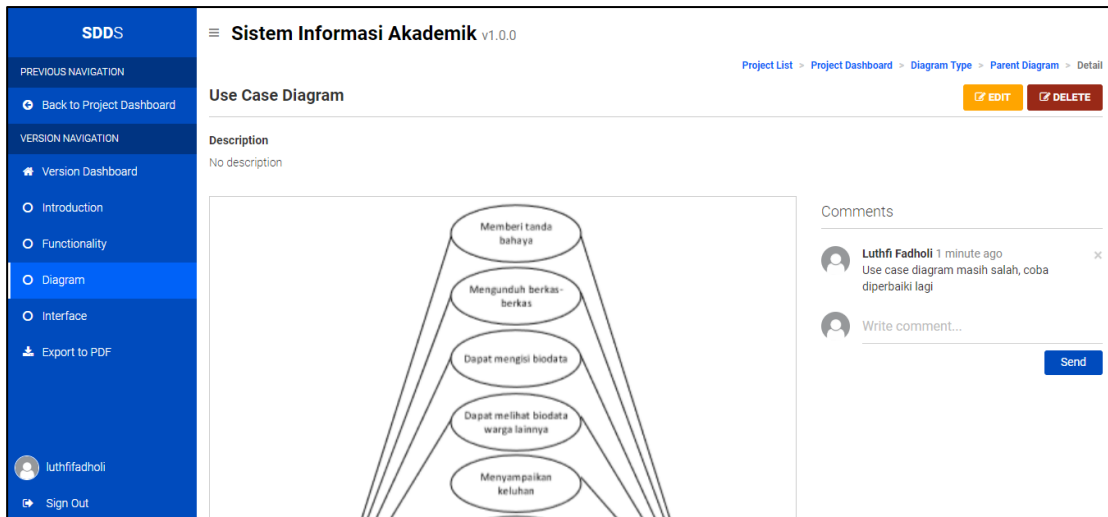
The left sidebar shows the navigation menu with 'Diagram' selected. The top right corner shows the breadcrumb trail: Project List > Project Dashboard > Diagram Type > Parent Diagram > Sub Diagram.

Gambar 4.32 Halaman tambah *sub* diagram

### 32. Halaman Detail Diagram

Jika pengguna mengklik tombol lihat detail pada salah satu *parent* diagram pada halaman daftar *parent* diagram, maka pengguna akan diarahkan ke halaman detail diagram. Pada halaman ini akan tampil gambar dengan ukuran yang cukup besar sehingga cukup jelas dilihat oleh pengguna. Selain itu pengguna juga dapat meninggalkan komentar untuk gambar diagram yang ada sehingga kolaborasi antar sesama anggota project dapat menjadi lebih baik. Pada halaman ini pengguna juga dapat melakukan edit dan hapus diagram.

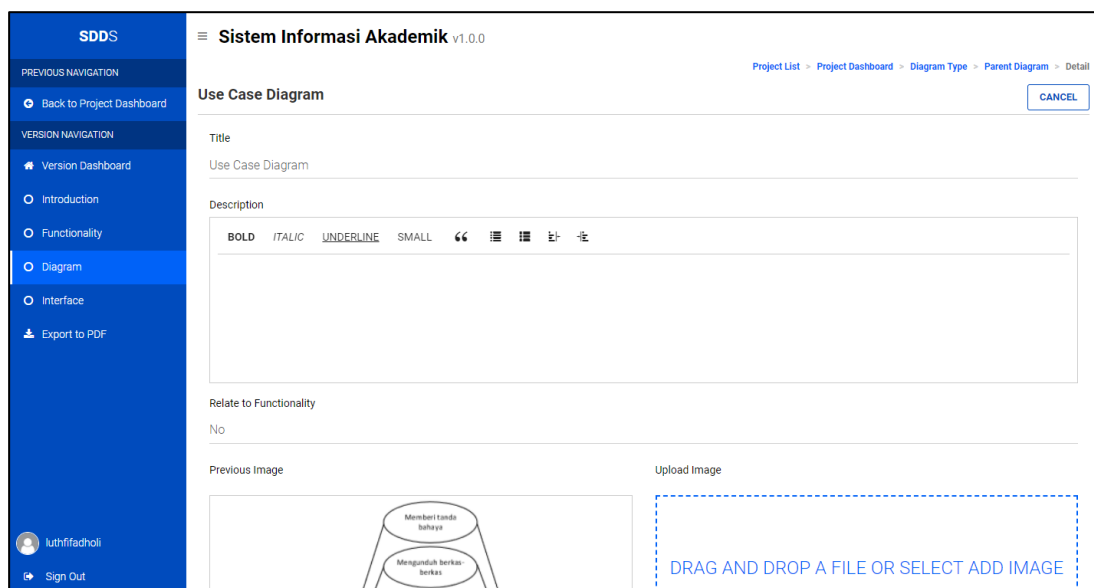




Gambar 4.33 Halaman detail diagram

### 33. Halaman Edit Diagram

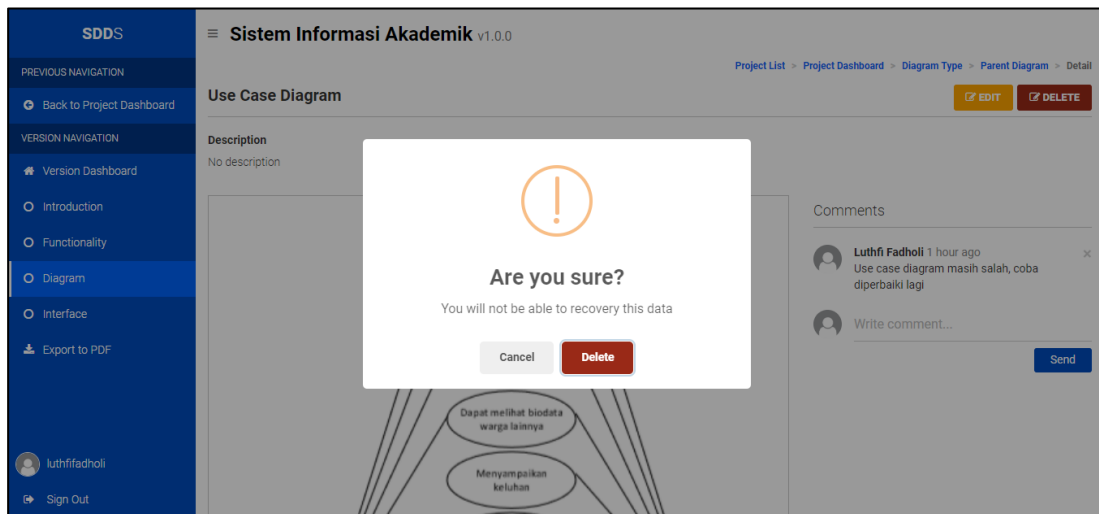
Halaman form edit bisa diakses dengan menekan tombol edit pada halaman detail diagram. Pada halaman ini pengguna dapat mengedit judul, relasi dengan fungsionalitas, maupun mengganti gambar diagram.



Gambar 4.34 Halaman edit diagram

### 34. Konfirmasi Hapus Diagram

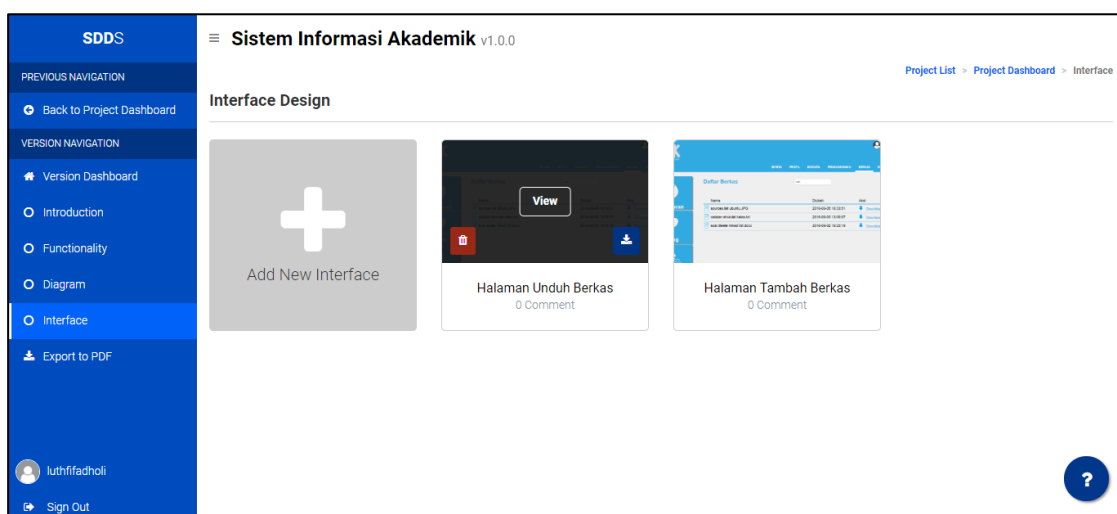
Apabila pengguna menekan tombol hapus, maka akan muncul peringatan untuk konfirmasi apakah pengguna benar-benar ingin menghapus diagram atau tidak. Fitur ini bertujuan untuk mencegah kesalahan pengguna dalam menghapus data.



Gambar 4.35 Konfirmasi hapus diagram

### 35. Halaman Daftar Desain Antarmuka

Halaman ini merupakan penerapan dari standar SDD IEEE 1016-2009 yaitu tampilan desain antarmuka. Pada halaman ini akan tampil daftar desain antarmuka dan tombol tambah untuk menambahkan desain antarmuka baru. Jika pengguna mengarahkan *cursor mouse* diatas salah satu desain, maka akan muncul 3 tombol yaitu tombol *view* untuk melihat detail desain antarmuka, tombol dengan *icon* tempat sampah untuk menghapus, dan tombol *download* untuk mengunduh *file*.



Gambar 4.36 Halaman daftar desain antarmuka

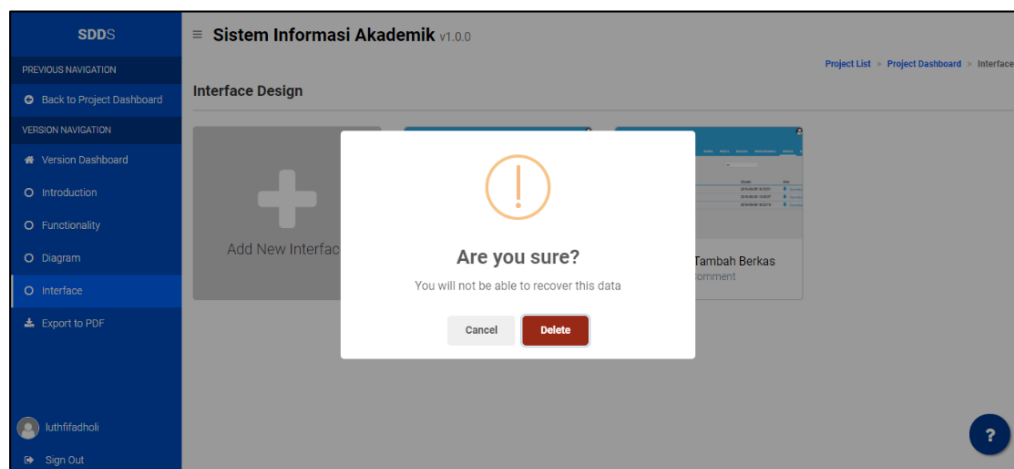
### 36. Halaman Tambah Desain Antarmuka

Halaman ini dapat dibuka dengan memilih tombol tambah pada halaman daftar desain antarmuka. Pada halaman ini pengguna diminta memasukkan 3 data, yaitu judul, deskripsi, dan file gambar yang ingin diunggah.

Gambar 4.37 Halaman tambah desain antarmuka

### 37. Konfirmasi Hapus Desain Antarmuka

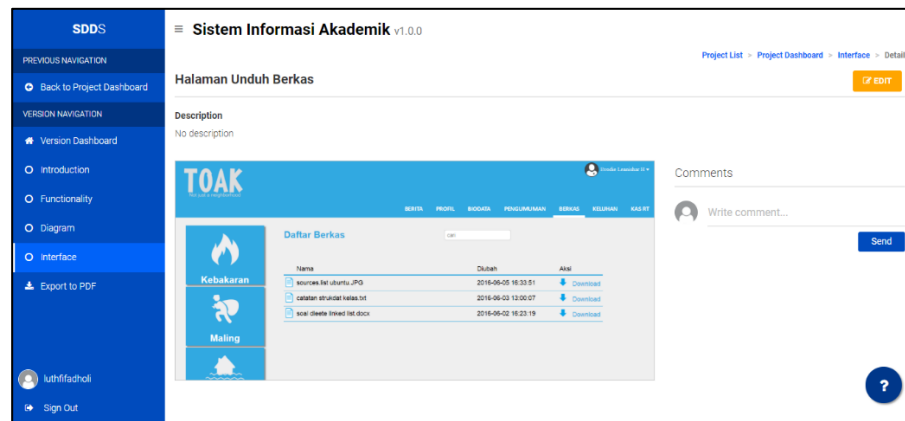
Apabila pengguna menekan tombol hapus, maka akan muncul peringatan untuk konfirmasi apakah pengguna benar-benar ingin menghapus desain antarmuka yang dipilih atau tidak. Fitur ini bertujuan untuk mencegah kesalahan pengguna dalam menghapus data.



Gambar 4.38 Konfirmasi hapus desain antarmuka

### 38. Halaman Detail Desain Antarmuka

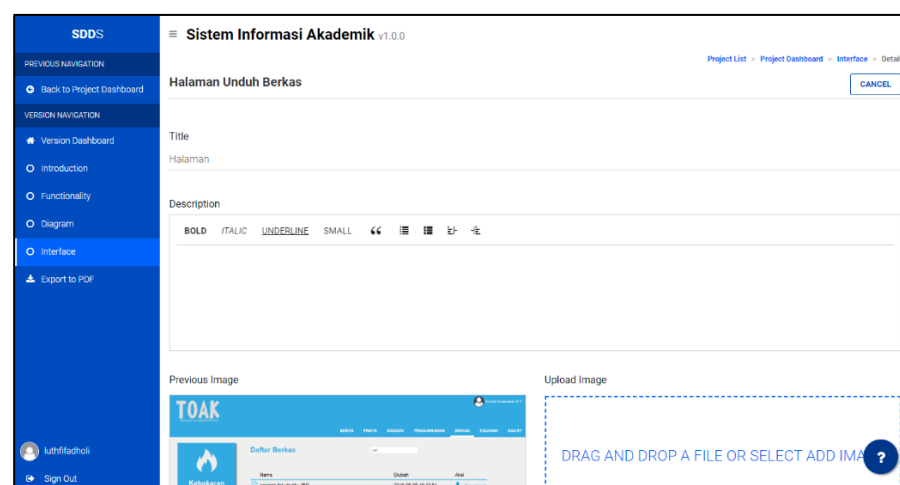
Apabila pengguna menekan tombol *view* maka pengguna akan diarahkan ke halaman detail desain antarmuka. Pada halaman ini akan tampil gambar desain antarmuka dengan ukuran yang cukup besar sehingga cukup jelas dilihat oleh pengguna. Selain itu pengguna juga dapat meninggalkan komentar untuk gambar desain yang ada sehingga kolaborasi antar sesama anggota project dapat menjadi lebih baik. Pada halaman ini pengguna juga dapat melakukan edit desain antarmuka.



Gambar 4.39 Halaman detail desain antarmuka

### 39. Halaman Edit Desain Antarmuka

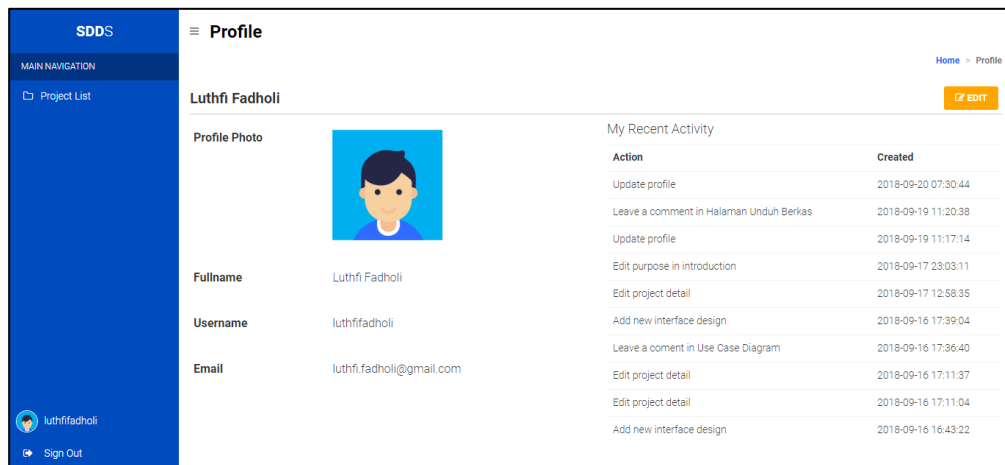
Apabila pengguna menekan tombol edit, maka sistem akan menyembunyikan halaman detail desain antarmuka dan menampilkan form edit desain antarmuka. Pada halaman ini pengguna dapat mengedit judul dan mengganti gambar desain antarmuka



Gambar 4.40 Halaman edit desain antarmuka

#### 40. Halaman Profil

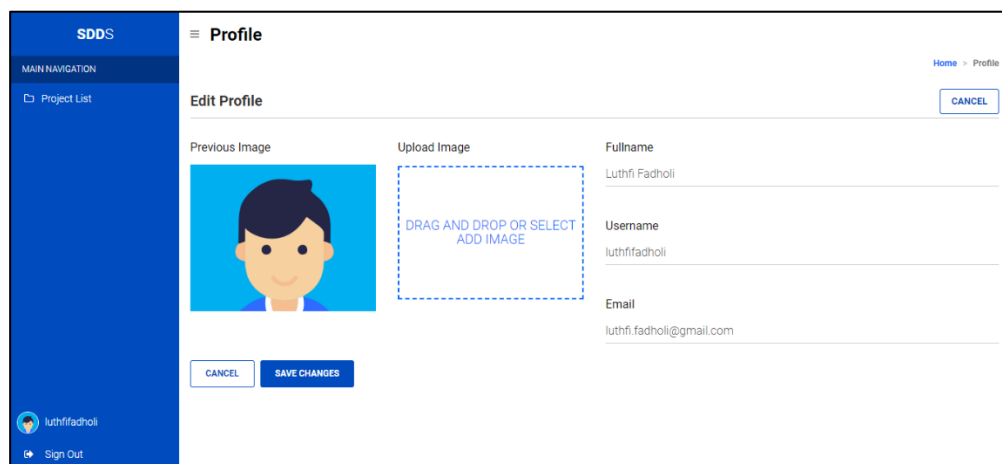
Halaman ini dapat diakses dengan mengklik menu dengan tulisan *username* dan foto pengguna yang sedang *login*. Pada halaman ini akan tampil detail data pengguna yang sedang login, yaitu nama lengkap, *username*, *password*, dan juga foto pengguna. Selain itu akan tampil juga 10 aktivitas terakhir yang dilakukan pengguna, sehingga pengguna bisa mengontrol apa yang baru saja pengguna lakukan. Pengguna juga dapat mengedit data pribadinya dengan mengklik tombol edit yang ada pada sisi kanan atas halaman.



Gambar 4.41 Halaman profil

#### 41. Halaman Edit Profil

Apabila pengguna mengklik tombol edit, maka akan tampil form edit profil. Pada halaman ini pengguna dapat mengedit data pribadinya dan juga mengganti foto profilnya.



Gambar 4.42 Halaman edit profil

## 4.2 Pengujian

Setelah dilakukan implementasi sistem, dilakukan juga pengujian kepuasan pengguna terhadap sistem yang sudah dibuat. Pengujian dilakukan oleh 20 orang secara acak yang terdiri dari beberapa alumni Teknik Informatika UII yang bekerja dibidang TI dan beberapa mahasiswa Teknik Informatika UII. Pengujian dilakukan dengan cara mengisi kuesioner yang telah disediakan, sehingga didapatkan presentase kelayakan dari sistem yang telah dibuat. Berikut hasil dari pengujian yang telah dilakukan.

Tabel 4.1 Hasil Kuesioner Pengujian

No	Keterangan	STS	TS	N	S	SS
1	Fitur komentar pada sistem bermanfaat dalam kolaborasi antar tim pengembang untuk memberi saran maupun kritik terhadap perancangan sistem yang telah dibuat			1	10	9
2	Fitur <i>log</i> /pencatatan aktivitas pada sistem bermanfaat dalam kolaborasi antar tim pengembang untuk mendapatkan informasi riwayat perubahan yang terjadi pada sistem.			3	9	8
3	Sistem dapat menghasilkan dokumen perancangan perangkat lunak yang baik dan rapi sehingga mudah dipahami oleh pembacanya			3	10	7
4	Sistem memiliki potensi dan manfaat yang besar sebagai aplikasi dokumentasi rekayasa perangkat lunak pada tahap perancangan		1	3	11	5
5	Sistem memiliki tampilan yang menarik		2	4	10	4
6	Sistem menyediakan mekanisme yang jelas dan mudah dimengerti bagi tim pengembang dalam membuat maupun mendapatkan informasi perancangan sistem			6	8	6
7	Sistem memiliki tata letak elemen antarmuka yang memudahkan bagi pengguna			3	8	9
8	Sistem memiliki ukuran dan warna elemen antarmuka yang nyaman bagi pengguna			3	10	7

Berdasarkan rumus perhitungan yang telah ditentukan sebelumnya, berikut perhitungan dari hasil pengujian yang telah dilakukan:

$$\text{Nilai maksimal (NM)} : (5 \times 20 \times 8) = 800$$

$$\text{Skor pengujian (SP)} : (2 \times 3) + (3 \times 26) + (4 \times 76) + (5 \times 55) = (6+78+304+275) = 663$$

$$\text{Persentase} : \frac{SP}{NM} \times 100 \% = \frac{663}{800} \times 100\% = 82.875\% = 83\% \text{ (dibulatkan)}$$

Dari hasil perhitungan diatas, didapatkan presentase pengujian sebesar 83% sehingga dapat disimpulkan bahwa sistem yang telah dibuat sudah baik dari segi fungsionalitas maupun tampilan. Dan jika dilihat dari sisi kolaborasi tim pengembang, didapatkan hasil sebagai berikut:

$$\text{Nilai maksimal (NM)} : (5 \times 20 \times 2) = 200$$

$$\text{Skor pengujian (SP)} : (3 \times 4) + (4 \times 19) + (5 \times 17) = (12+76+85) = 173$$

$$\text{Persentase} : \frac{SP}{NM} \times 100 \% = \frac{173}{200} \times 100\% = 86.5\% = 87\% \text{ (dibulatkan)}$$

Dari hasil perhitungan diatas, didapatkan presentase pengujian sebesar 87% sehingga dapat disimpulkan bahwa sistem yang telah dibuat memiliki manfaat bagi para tim pengembang perangkat lunak untuk membantu kolaborasi dalam pembuatan dan penggunaan SDD.

## **BAB V**

### **KESIMPULAN DAN SARAN**

#### **5.1 Kesimpulan**

Berdasarkan hasil analisis, perancangan, dan pembuatan sistem yang telah dilakukan, dapat diambil kesimpulan dari penelitian sebagai berikut.

- a. Sistem ini dapat menghasilkan dokumen perancangan perangkat lunak yang baik dan sesuai dengan standar IEEE
- b. Sistem ini memiliki manfaat bagi para tim pengembang perangkat lunak untuk membantu kolaborasi dalam pembuatan dan penggunaan SDD.

#### **5.2 Saran**

Berdasarkan analisis kelemahan yang ada pada sistem dan kesimpulan di atas, dapat diambil beberapa saran yang dapat dipertimbangkan apabila akan dilakukan pengembangan terhadap sistem ini. Adapun saran untuk pengembangan sistem yang akan datang diantaranya:

- a. Mengembangkan fitur perancangan diagram secara langsung sehingga dapat memudahkan dan mempercepat proses perancangan sistem
- b. Menambahkan lagi konten SDD yang ada sesuai dengan standar IEEE agar SDD yang dihasilkan lebih baik lagi kedepannya



## DAFTAR PUSTAKA

- Arifin, F. (2011). Manajemen Proyek Perangkat Lunak. *ComTech*, 2(1), 284–295.
- Ferriana, R. W. (2015). Aplikasi Manajemen Dokumen Perancangan Perangkat Lunak.
- Forward, A. (2002). Software documentation –Building and maintaining artefacts of communication. Retrieved from [https://www.site.uottawa.ca/~tcl/gradtheses/afoward/afoward\\_thesis.pdf](https://www.site.uottawa.ca/~tcl/gradtheses/afoward/afoward_thesis.pdf)
- IEEE. (2009). *IEEE Std 1016-2009 (Revision of IEEE Std 1016-1998), IEEE Standard for Information Technology—Systems Design—Software Design Descriptions. Middle East* (Vol. 2009). <https://doi.org/10.1109/IEEESTD.2009.5167255>
- Kusumasari, T. F., Sastramihardja, H., Surendro, K., & Supriana, I. (2013). FAKTOR PERILAKU MANUSIA DALAM KOLABORASI PENGEMBANGAN PERANGKAT LUNAK, (February).
- Kusumasari, T. F., Supriana, I., Surendro, K., & Sastramihardja, H. (2011). Collaboration model of software development. *Proceedings of the 2011 International Conference on Electrical Engineering and Informatics*, (July), 1–6. <https://doi.org/10.1109/ICEEI.2011.6021769>
- Mcelrath, R. (2007). Example XML Legal Document Utility Software Design Document, 1–48.
- Najwaini, E. (2012). Dokumentasi sebagai bagian dari perangkat lunak, *2012(Semantik)*, 499–505.
- Simarmata, J. (2010). *Rekayasa Perangkat Lunak - Janner Simarmata - Google Buku*. Retrieved April 20, 2018, from [https://books.google.co.id/books?hl=id&lr=&id=QP1BjG\\_VIsoC&oi=fnd&pg=PA1&dq=perangkat+lunak&ots=Tz-FoRP3uA&sig=4mrUKQ2tYX4CfrTwAI-b4VbPVOk&redir\\_esc=y#v=onepage&q=perangkat+lunak&f=false](https://books.google.co.id/books?hl=id&lr=&id=QP1BjG_VIsoC&oi=fnd&pg=PA1&dq=perangkat+lunak&ots=Tz-FoRP3uA&sig=4mrUKQ2tYX4CfrTwAI-b4VbPVOk&redir_esc=y#v=onepage&q=perangkat+lunak&f=false)

## LAMPIRAN