KATA PENGANTAR

Bismillahirrahmaanirrahim,

Puji dan syukur penulis panjatkan kepada Illahi Rabbi karena atas rahmat dan hidayah Nya, penulis dapat menyelesaikan tugas akhir dengan judul "**SISTEM INFORMASI AKADEMIK DI SD NEGERI 2 PARAKANSALAK"** dapat diselesaikan walaupun terdapat banyak kekurangan dan hambatan.

Penulis menyadari sepenuhnya penyusunan skripsi ini masih jauh dari kesempurnaan, dikarenakan masih terbatasnya ilmu pengetahuan, pengalaman serta kemampuan yang penulis miliki.

Dalam usaha penyusunan tugas akhir ini, penulis tidak dapat melupakan `bantuan dari beberapa pihak yang telah meluangkan waktunya yang sangat berharga untuk memberi bimbingan dan nasihat-nasihat yang berguna bagi penulis.

Oleh karena itu dengan rasa hormat, penulis ingin menyampaikan rasa terima kasih yang sebesar-besarnya kepada :

1. Bapak Ir. Eddy Suryanto Soegoto, M.Sc, selaku Rektor Universitas Komputer Indonesia.
2. Bapak Dr.H.Denny Kurniadie,Ir,M.Sc selaku Dekan Teknik dan Ilmu Komputer Universitas Komputer Indonesia.
3. Bapak Dadang Munandar, S.E., M.Si., selaku Ketua Program studi Sistem Informasi

4. Bapak Syahrul Mauludin., S.kom.selaku dosen pembimbing yang telah meluangkan waktu, tenaga, dan pikiran untuk memberikan pengarahan dan bimbingan kepada penulis dalam menyelesaikan skripsi ini.
5. Kepada seluruh dosen jurusan sistem informasi yang telah memberikan bimbingan kepada penulis selama menempuh pendidikan di universitas komputer indonesia.
6. Kepada seluruh teman teman yang tak henti hentinya memberikan semangat dan motivasi.

Secara khusus penulis mengucapkan terima kasih dan rasa hormat kepada yang tercinta ibu dan bapak, terutama ibu yang tiada henti hentinya menaburkan doa untuk penulis (bisakah “air mata “ itu menjadi kebahagian) dan untuk bapak mudah mudahan semua perlakuan dapat menjadi cambuk bagi penulis agar lebih baik. Penulis juga ucapkan terima kasih kepada kakak tercinta yang telah memberikan dorongan, semangat, batuan, dan doa, sehingga dengan kesabaran dan rasa tawakal segala rintangan dapat terlampaui.

Mudah mudahan skripsi ini dapat bermanfaat bagi pembacanya dan bagi peningkatan pembelajaran. Akhirnya kepada allah swt jualah penulis kembalikan segala sesuatunya. Semoga mal baik yang telah dicurahkan mendapat imbalan dari allah swt. Amin.

DAFTAR ISI

ABSTRAK ..... **i**

ABSTRACT ..... ***ii***

KATA PENGANTAR ..... **iii**

DAFTAR ISI ..... **v**

DAFTAR GAMBAR ..... **x**

DAFTAR TABEL ..... **xii**

DAFTAR SIMBOL ..... **xiv**

BAB I PENDAHULUAN

1.1 Latar Belakang 1

1.2 Identifikasi dan Rumusan Masalah 2

1.2.1 Identifikasi Masalah 2

1.2.2 Rumusan Masalah 3

1.3 Maksud dan Tujuan Penelitian 3

1.4 Kegunaan Penelitian 4

1.4.1 Kegunaan Praktis 4

1.4.2 Kegunaan Akademis 4

1.5 Batasan Masalah 4

1.6 Lokasi Dan Waktu Penelitian 6

BAB II LANDASAN TEORI

2.1 Konsep Dasar Sistem 7

2.1.1 Definisi Sistem 7

2.1.2 Elemen Sistem 7

2.1.3 Karakteristik Sistem 9

2.2 Konsep Dasar Informasi 11

2.2.1 Siklus Informasi 11

2.2.2 Kualitas dan Nilai Informasi 12

2.3 Konsep Dasar Sistem Informasi 12

2.4 Mengenal Java Netbeans 6.8.1 13

2.4.1 IDE NetBeans 13

2.4.2 Keunggulan NetBeans 13

2.4.3 Kelemahan NetBeans 14
2.5 Mengenal Database MySQL 14
2.6 Sistem Informasi Akademik 16
BAB III OBJEK DAN METODE PENELITIAN
3.1 Objek Penelitian 17
3.1.1 Visi dan Misi di SD Negeri 2 Parakansalak 18
3.1.1.1 Visi SD Negeri 2 Parakansalak 18
3.1.1.2 Misi SD Negeri 2 Parakansalak 18
3.1.2 Struktur Organisasi 18
3.1.3 Deskripsi Kerja 19
3.2 Metode Penelitian 21
3.2.1 Desain Penelitian 21
3.2.2 Jenis dan Metode Pengumpulan Data 22
3.2.2.1 Sumber Data Primer 22
3.2.2.2 Sumber Data Sekunder 22
3.2.3 Metode Pendekatan dan Pengembangan Sistem 23
3.2.3.1 Metode Pendekatan Sistem 23
3.2.3.2 Metode Pengembangan Sistem 23
3.2.3.3 Alat Bantu Analisis dan Perancangan 26
3.2.4 Pengujian Software 28
BAB IV ANALISIS DAN PERANCANGAN SISTEM
4.1 Analisis Sistem yang Berjalan 30
4.1.1 Analisis Dokumen 30
4.1.2 *Use Case* 33
4.1.2.1 *Use Case Diagram* yang sedang Berjalan 33
4.1.2.2 Skenario *Use Case* Pendaftaran yang Berjalan 33
4.1.2.3 Skenario *Use Case* Penjadwalan yang Berjalan 35
4.1.2.4 Skenario *Use Case* Penilian yang Berjalan 36
4.1.2.5 Skenario *Use Case* Data guru yang Berjalan 37

4.1.3 *Activity Diagram* 38

4.1.3.1 *Activity Diagram* untuk Pendaftaran yang Berjalan 38

4.1.3.2 *Activity Diagram* untuk Penjadwalan yang Berjalan 39

4.1.3.3 *Activity Diagram* untuk Penilaian yang Berjalan 40

4.1.3.4 *Activity Diagram* untuk Data guru yang Berjalan 41

4.2 Perancangan Sistem 41

4.2.1 Tujuan Perancangan Sistem 41

4.2.2 Gambaran Umum Sistem yang Diusulkan 42

4.2.3 Diagram *Use Case* yang Diusulkan 43

4.2.3.1 Skenario *Use Case* Pendaftaran yang Diusulkan 44

4.2.3.2 Skenario *Use Case* Penjadwalan yang Diusulkan 45

4.2.3.3 Skenario *Use Case* Penilaian yang Diusulkan 46

4.2.3.4 Skenario *Use Case* Data guru yang Diusulkan 47

4.2.4 Diagram Aktivitas *(Activity Diagram)* yang Diusulkan 48

4.2.4.1 Diagram Aktivitas Login yang Diusulkan 48

4.2.4.2 Diagram Aktivitas Pendaftaran yang Diusulkan 49

4.2.4.3 Diagram Aktivitas Penjadwalan yang Diusulkan 50

4.2.4.4 Diagram Aktivitas Penilaian yang Diusulkan 51

4.2.4.5 Diagram Aktivitas data guru yang Diusulkan 52

4.2.5 Diagram Sequence (*Sequence Diagram*) 53

4.2.5.1 Sequence Diagram Login yang Diusulkan..... 54

4.2.5.2 Sequence Diagram Pendaftaran yang Diusulkan 55

4.2.5.3 Sequence Diagram Penjadwalan yang Diusulkan 56
4.2.5.4 Sequence Diagram penilaian yang Diusulkan..... 57
4.2.5.5 Sequence Diagram data guru yang Diusulkan 58
4.2.6 Class diagaram 59
4.2.7 Component diagram 60
4.2.8 Deployment diagram 61
4.2.9 Perancangan antar muka 62
4.2.9.1 perancangan struktur menu 62
4.2.10 Perancangan input 63
4.2.11 Perancangan output 68
4.2.12 Perancangan arsitektur jaringan 71

BAB V IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 Implementasi 72
5.1.1 Batasan Implementasi 72
5.1.2 Implementasi perangkat lunak 73
5.1.3 Implementasi perangkat keras 73
5.1.4 Implementasi basis data 74
5.1.5 Implementasi antar muka 75
5.1.5.1 Implementasi halaman utama 76
5.1.5.2 Implementasi Menu file 76
5.1.5.3 Implementasi Menu master 77
5.1.5.4 Implementasi Menu laporan 77
5.1.6 penggunaan program 77

5.1.6.1 form menu login 77

5.1.6.2 form menu utama 78

5.1.6.3 form pendaftaran 79

5.1.6.4 form penjadawalan 80

5.1.6.5 form penilaian 81

5.1.6.6 laporan data siswa 81

5.1.6.7 laporan nilai siswa 82

5.1.6.8 laporan jadwal pelajaran 82

5.2 Pengujian 83

5.2.1 kasus hasil pengujian 83

5.2.2 kesimpulan hasil pengujian 85

BAB VI KESIMPULAN DAN SARAN

6.1 kesimpulan 86

6.2 saran 86

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

GAMBAR 2.1	Transformasi Data Menjadi Informasi	11
GAMBAR 3.1	Struktur Organisasi SD Negeri 2 Parakansalak	19
GAMBAR 3.2	Pengembangan *Prototype*	25
GAMBAR 4.1	*Use Case* Diagram Yang Sedang Berjalan	33
GAMBAR 4.2	*Activity Diagram* Pendaftaran yang Berjalan	38
GAMBAR 4.3	*Activity Diagram* Penjadwalan yang Berjalan	39
GAMBAR 4.4	*Activity Diagram* Penilaian yang Berjalan	40
GAMBAR 4.5	*Activity Diagram* Data Guru yang Berjalan	41
GAMBAR 4.6	*Use Case* Diagram yang Diusulkan	43
GAMBAR 4.7	*Activity Diagram* Login yang Diusulkan	48
GAMBAR 4.8	*Activity Diagram* Pendaftaranyang Diusulkan	49
GAMBAR 4.9	*Activity Diagram* penjadwalan yang Diusulkan	50
GAMBAR 4.10	*Activity Diagram* penilaian yang Diusulkan	51
GAMBAR 4.11	*Activity Diagram* data guru yang Diusulkan	52
GAMBAR 4.12	*Sequence Diagram* Login yang Di usulkan	54
GAMBAR 4.13	*Sequence Diagram* Pendaftaran yang Diusulkan	55
GAMBAR 4.14	*Sequence Diagram* Penjadwalan yang Diusulkan	56
GAMBAR 4.15	*Sequence Diagram* Penilaian yang Diusulkan	57
GAMBAR 4.16	*Sequence Diagram* data guru yang Diusulkan	58
GAMBAR 4.17	class diagram yang Diusulkan	59

GAMBAR 4.18 component diagram yang Diusulkan 61

GAMBAR 4.19 deployment diagram yang Diusulkan 61

GAMBAR 4.20 Perancangan Struktur Menu yang Diusulkan 62

GAMBAR 4.21 Tampilan Login 63

GAMBAR 4.22 Rancangan Pendaftaran 64

GAMBAR 4.23 Rancangan Penjadwalan 65

GAMBAR 4.24 Rancangan Data Guru 66

GAMBAR 4.25 Rancangan Input Nilai 67

GAMBAR 4.26 Perancangan output laporan nilai 68

GAMBAR 4.27 Perancangan output laporan penjadwalan 69

GAMBAR 4.28 Perancangan output laporan pendaftaran 70

GAMBAR 4.29 Arsitektur jaringan yang diusulkan 71

GAMBAR 5.1 Menu login 78

GAMBAR 5.2 Pesan kesalahan pada form login 78

GAMBAR 5.3 Form menu utama 79

GAMBAR 5.4 form pendaftaran 80

GAMBAR 5.5 form penjadwalan 80

GAMBAR 5.6 form penilaian 81

GAMBAR 5.7 laporan data siswa 81

GAMBAR 5.8 laporan penilaian 82

GAMBAR 5.9 laporan penjadwalan 82

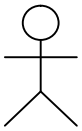
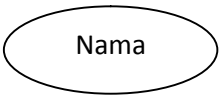
DAFTAR TABEL

TABEL 1.1	Jadwal Penyelesaian Tugas Akhir	6
TABEL 4.1	Dokumen Formulir Pendaftaran	31
TABEL 4.2	Dokumen Lampiran Jadwal Pelajaran	31
TABEL 4.3	Dokumen Lampiran Data Penilaian	32
TABEL 4.4	Dokumen Lampiran Data Guru	32
TABEL 4.5	Skenario *Use Case* Pendaftaran yang Berjalan	34
TABEL 4.6	Skenario *Use Case* Penjadwalan yang Berjalan	35
TABEL 4.7	Skenario *Use Case* Penilaian yang Berjalan	36
TABEL 4.8	Skenario *Use Case* Data Guru yang Berjalan	37
TABEL 4.9	Skenario *Use Case* Pendaftaran yang Diusulkan	44
TABEL 4.10	Skenario *Use Case* Penjadwalan yang Diusulkan	45
TABEL 4.11	Skenario *Use Case* Penilaian yang Diusulkan	46
TABEL 4.12	Skenario *Use Case* lap pendaftaran yang Diusulkan	47
TABEL 4.11	Skenario *Use Case* lap penjadwalan yang Diusulkan	46
TABEL 4.12	Skenario *Use Case* lap penilaian yang Diusulkan	47
TABEL 5.1	Implementasi halaman utama	76
TABEL 5.2	Implementasi menu file	76
TABEL 5.3	Implementasi menu master	77
TABEL 5.4	Implementasi menu laporan	77
TABEL 5.5	kasus hasil pengujian login benar	83
TABEL 5.6	kasus hasil pengujian login salah	83

TABEL 5.7	kasus hasil pengujian pendaftaran benar	84
TABEL 5.8	kasus hasil pengujian pendaftaran salah	84
TABEL 5.9	kasus hasil pengujian penjadwalan benar	84
TABEL 5.10	kasus hasil pengujian penjadwalan salah	84

DAFTAR SIMBOL

USE CASE DIAGRAM

Notasi	Keterangan	Simbol
Actor	*Actor* adalah pengguna sistem. *Actor* tidak terbatas hanya manusia saja, jika sebuah sistem berkomunikasi dengan aplikasi lain dan membutuhkan input atau memberikan output, maka aplikasi tersebut juga bisa dianggap sebagai *actor.*	 Aktor
Use Case	*Use case* digambarkan sebagai lingkaran elips dengan nama *use case* dituliskan didalam elips tersebut.	
Association	Asosiasi digunakan untuk menghubungkan *actor* dengan *use case*. Asosiasi digambarkan dengan sebuah garis yang menghubungkan antara *actor* dengan *use Case*.	

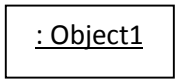
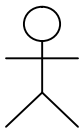
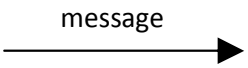
(Sumber : Modul *Workshop* UML Bab 2)

ACTIVITY DIAGRAM

Notasi	Keterangan	Simbol
Initial	Titik awal, untuk memulai suatu aktivitas.	
Final	Titik akhir, untuk mengakhiri aktivitas.	
Activity	*Activity*	
Decision	Pilihan untuk mengambil keputusan	

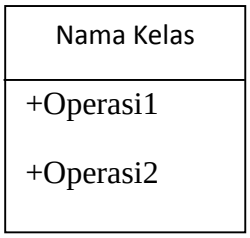
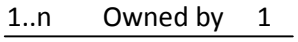
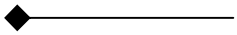
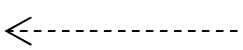
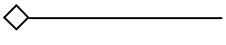
(Sumber : *User Guide Enterprise Architect* 7.0, Modul *Workshop* UML Bab 2)

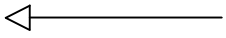
SEQUENCE DIAGRAM

Notasi	Keterangan	Simbol
***Object* (Partisipan)**	*Object* atau biasa juga disebut partisipan merupakan *instance* dari sebuah *class* dan dituliskan tersusun secara horizontal. Digambarkan sebagai sebuah *class* (kotak) dengan nama objek didalamnya yang diawali dengan sebuah titik koma.	
Actor	*Actor* juga dapat berkomunikasi dengan *object*, maka *actor* juga dapat diurutkan sebagai kolom.	
Lifeline	*Lifeline* mengindikasikan keberadaan sebuah *object* dalam basis waktu. Notasi untuk *lifeline* adalah garis putus-putus vertikal yang ditarik dari sebuah objek.	
Activation	*Activation* dinotasikan sebagai sebuah kotak segi empat yang digambar pada sebuah *lifeline*. *Activation* mengindikasikan sebuah objek yang akan melakukan sebuah aksi.	
Massage	*Message*, digambarkan dengan anak panah horizontal antara *activation*. *Message* mengindikasikan komunikasi antara *object-object*.	
Self-Message	*Self-message* atau panggilan mandiri mengindikasikan komunikasi kembali kedalam sebuah objek itu sendiri.	

(Sumber : *Martin Fowler* UML *Distilled* 2005, Modul *Workshop* UML Bab 2)

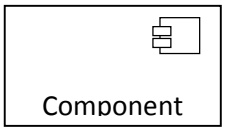
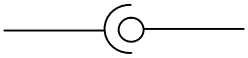
CLASS DIAGRAM

Notasi	Keterangan	Simbol
Class	*Class* adalah blok - blok pembangun pada pemrograman berorientasi objek. Sebuah class digambarkan sebagai sebuah kotak yang terbagi atas 3 bagian. Bagian atas adalah bagian nama dari class. Bagian tengah mendefinisikan property/atribut *class*. Bagian akhir mendefinisikan method-method dari sebuah *class*.	
Assosiation	Sebuah asosiasi merupakan sebuah *relationship* paling umum antara 2 *class*, dan dilambangkan oleh sebuah garis yang menghubungkan antara 2 *class*. Garis ini bisa melambangkan tipe-tipe relationship dan juga dapat menampilkan hukum-hukum multiplisitas pada sebuah *relationship* (Contoh: *One-to-one, one-to-many, many-to-many*).	
Composition	Jika sebuah class tidak bisa berdiri sendiri dan harus merupakan bagian dari *class* yang lain, maka *class* tersebut memiliki relasi *composition* terhadap class tempat dia bergantung tersebut. Sebuah *relationship composition* digambarkan sebagai garis dengan ujung berbentuk jajaran genjang berisi/solid.	
Dependency	Kadangkala sebuah *class* menggunakan *class* yang lain. Hal ini disebut *dependency*. Umumnya penggunaan *dependency* digunakan untuk menunjukkan operasi pada suatu *class* yang menggunakan *class* yang lain. Sebuah *dependency* dilambangkan sebagai sebuah panah bertitik-titik.	
Aggregation	*Aggregation* mengindikasikan keseluruhan bagian *relationship* dan biasanya disebut sebagai relasi "mempunyai sebuah" atau "bagian dari". Sebuah *aggregation*	

	digambarkan sebagai sebuah garis dengan sebuah jajaran genjang yang tidak berisi/tidak solid.	
Generalization	Sebuah relasi *generalization* sepadan dengan sebuah relasi *inheritance* pada konsep berorientasi objek. Sebuah *generalization* dilambangkan dengan sebuah panah dengan kepala panah yang tidak solid yang mengarah ke kelas "*parent*"-nya/induknya.	

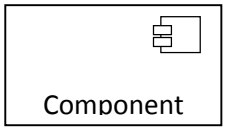
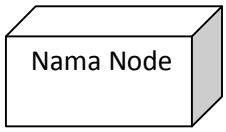
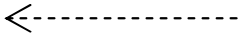
(Sumber : Modul *Workshop* UML Bab 2)

COMPONENT DIAGRAM

Notasi	Keterangan	Simbol
Component	Sebuah komponen melambangkan sebuah entitas *software* dalam sebuah sistem. Sebuah komponen dinotasikan sebagai sebuah kotak segiempat dengan dua kotak kecil tambahan yang menempel disebelah kirinya.	
Assembly	Sebuah konektor diantara dua komponen yang salah satunya menyediakan layanan ke komponen lain yang memerlukannya.	
Association	Sebuah *association* digambarkan sebagai sebuah garis yang menghubungkan dua komponen yang mengindikasikan jalur komunikasi antara element-elemen software.	
Depedency	Sebuah *dependency* digunakan untuk menotasikan relasi antara dua komponen. Notasinya adalah tanda panah putus-putus yang diarahkan kepada komponen tempat sebuah komponen itu bergantung.	

(Sumber : *User Guide Enterprise Architect* 7.0, Modul *Workshop* UML Bab 2)

DIAGRAM DEPLOYMENT

Notasi	Keterangan	Simbol
Component	Pada *deployment* diagram, komponen-komponen yang ada diletakkan didalam node untuk memastikan keberadaan posisi mereka.	
Node	Node menggambarkan bagian-bagian hardware dalam sebuah sistem. Notasi untuk node digambarkan sebagai sebuah kubus 3 dimensi.	
Association	Sebuah *association* digambarkan sebagai sebuah garis yang menghubungkan dua node yang mengindikasikan jalur komunikasi antara element-elemen hardware.	
Depedency	Sebuah *dependency* digunakan untuk menotasikan relasi antara dua *node*. Notasinya adalah tanda panah putus-putus yang diarahkan kepada *node* tempat sebuah *node* itu bergantung.	

(Sumber : *User Guide Enterprise Architect* 7.0, Modul *Workshop* UML Bab 2)