

PERANCANGAN PERANGKAT LUNAK SISTEM INFORMASI KEPANGKATAN PEGAWAI PERUMNAS REGIONAL IV BANDUNG

Sri Lestari¹, Ripan Dapi²

^{1,2}Jurusan Sistem Informasi, Fakultas Teknik, Universitas Widyatama Bandung
Jl. Cikutra No. 204 A Bandung Telp 022 7278860
Email: sri.lestari@widyatama.ac.id

Abstrak

Tenaga kerja merupakan faktor yang sangat penting dalam menentukan kelancaran dan peningkatan kegiatan usaha pada sebuah perusahaan. Dalam perkembangan teknologi dan informasi, komputer sangat berperan sekali dalam pengolahan data suatu informasi. Informasi yang dihasilkan dapat terjamin kebenaran, ketepatan, dan keakuratannya. Sistem informasi kenaikan pangkat pegawai yang sedang berjalan dibagian kepegawaian Perum PERUMNAS Regional IV Bandung masih dilakukan secara manual, sehingga proses pengolahan data kenaikan pangkat membutuhkan waktu yang lama, oleh karena itu untuk meningkatkan kinerja perusahaan dalam bidang pelayanan kepada karyawannya perlu dibuat perangkat lunak sistem informasi kepangkatan pegawai PERUMNAS regional IV Bandung. Metodologi yang digunakan dalam pembangunan perangkat lunak ini adalah sistem Object Oriented Analysis & Design (OOAD) yang diawali dengan mengidentifikasi kebutuhan atau fungsi-fungsi yang harus tersedia dalam sistem perangkat lunak. Kebutuhan tersebut dimodelkan dalam usecase-usecase menggunakan usecase diagram. Masing-masing usecase dijelaskan dalam Use Case Specification. Selanjutnya melakukan realisasi kebutuhan (requirement) di dalam object-object. Relasi dan interaksi antar objek harus ditentukan agar diperoleh realisasi fungsi-fungsi yang dibutuhkan dalam sistem yang digambarkan melalui Sequence Diagram dan Class Diagram. Class Diagram ini akan memberikan arah taktis mengimplementasikan objek-objek beserta atributnya serta perilakunya pada class-class yang sesuai dengan bahasa pemrograman yang digunakan. Di akhir proses pengembangan ini terkumpul dokumentasi konsep pengembangan sistem hingga implementasi teknis ke dalam code serta manual untuk menjelaskan tata cara penggunaan dan pengembangan lanjutan. Bahasa yang digunakan dalam pengembangan perangkat lunak ini adalah Visual Basic

Kata kunci: class diagram ; sequence diagram; OOAD ;Visual Basic

Pendahuluan

Rekayasa Perangkat Lunak adalah suatu disiplin ilmu yang membahas semua aspek produksi perangkat lunak, mulai dari tahap awal *requirement capturing* (analisa kebutuhan pengguna), *specification* (menentukan spesifikasi dari kebutuhan pengguna), *desain*, *coding*, *testing* sampai pemeliharaan sistem setelah digunakan. Ada dua paradigma yaitu perangkat lunak menggunakan pendekatan terstruktur (*Data Oriented Approach*) dan perangkat lunak menggunakan pendekatan objek (*Object Oriented Approach*).

Analisis dan Design Berbasis Objek(OOAD) adalah salah satu pendekatan dalam rekayasa atau pengembangan perangkat lunak yang memodelkan sebuah system sebagai kemampuan dari banyak objek yang saling berinteraksi. Setiap objek merupakan representasi dari entitas-entitas yang ada pada system tersebut dan dapat dibedakan berdasarkan atribut (elemen data) dan perilaku entitas [5].

Jadi OOA adalah metode analisis yang memeriksa *requirements* dari sudut pandang kelas-kelas dan objek-objek yang ditemui dalam ruang lingkup permasalahan. Sasaran OOA adalah mengembangkan sederetan model yang menggambarkan perangkat lunak komputer pada saat perangkat lunak tersebut berkerja untuk memenuhi serangkaian persyaratan yang ditentukan oleh pelanggan, sementara OOD adalah metode untuk mengarahkan arsitektur software yang didasarkan pada manipulasi objek-objek sistem atau subsistem. OOD mentransformasikan model analisis yang dibuat menggunakan OOA kedalam suatu model desain yang berfungsi sebagai cetak biru bangunan perangkat lunak. OOD menghasilkan desain yang mencapai sejumlah tingkatan yang berbeda dari modularitasnya

Banyak metode/konsep OOAD yang dapat digunakan, dan pada paparan ini metode yang digunakan adalah mengacu pada pemodelan UML yaitu RUP (Rational Unified Process), dan untuk dokumen pembangunan perangkat lunak, pada paper ini mengacu pada template Dokumen RUP.

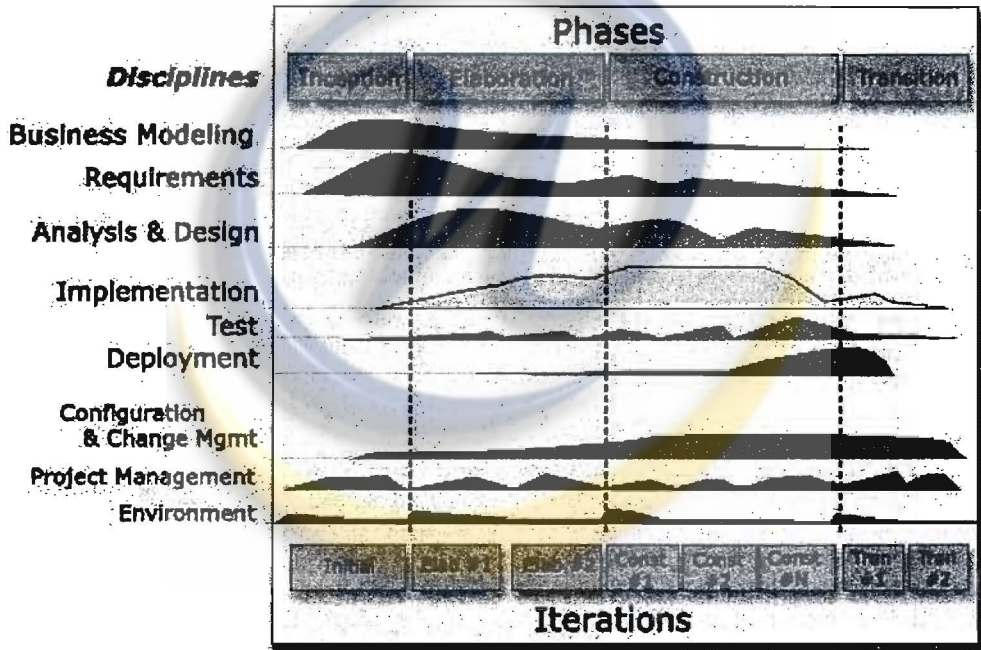
Pengembangan Model

Metode *Rational Unified Process* (RUP) merupakan suatu metode rekayasa perangkat lunak yang dikembangkan dengan mengumpulkan berbagai best practises yang terdapat dalam industri pengembangan perangkat lunak. Ciri utama metode ini adalah menggunakan use-case driven dan pendekatan iteratif untuk siklus pengembangan perangkat lunak. Gambar dibawah menunjukkan secara keseluruhan arsitektur yang dimiliki RUP. RUP menggunakan konsep *object oriented*, dengan aktifitas yang berfokus pada pengembangan model dengan menggunakan *Unified Model Language* (UML). [4]

Melalui gambar dibawah dapat dilihat bahwa RUP memiliki, yaitu:

1. Dimensi pertama digambarkan secara horizontal. Dimensi ini mewakili aspek-aspek dinamis dari pengembangan perangkat lunak. Aspek ini dijabarkan dalam tahapan pengembangan atau *fase*. Setiap *fase* akan memiliki suatu major milestone yang menandakan akhir dari awal dari phase selanjutnya. Setiap *phase* dapat berdiri dari satu beberapa iterasi. Dimensi ini terdiri atas *Inception, Elaboration, Construction, dan Transition*.
2. Dimensi kedua digambarkan secara vertikal. Dimensi ini mewakili aspek-aspek statis dari proses pengembangan perangkat lunak yang dikelompokkan ke dalam beberapa disiplin. Proses pengembangan perangkat lunak yang dijelaskan kedalam beberapa disiplin terdiri dari empat elemen penting, yakni *who is doing, what, how dan when*.

Dimensi diatas terdiri atas : *Business Modeling, Requirement, Analysis and Design, Implementation, Test, Deployment, Configuration dan Change Management, Project Management, Environment*.



Gambar 1 Arsitektur Rational Unified Process

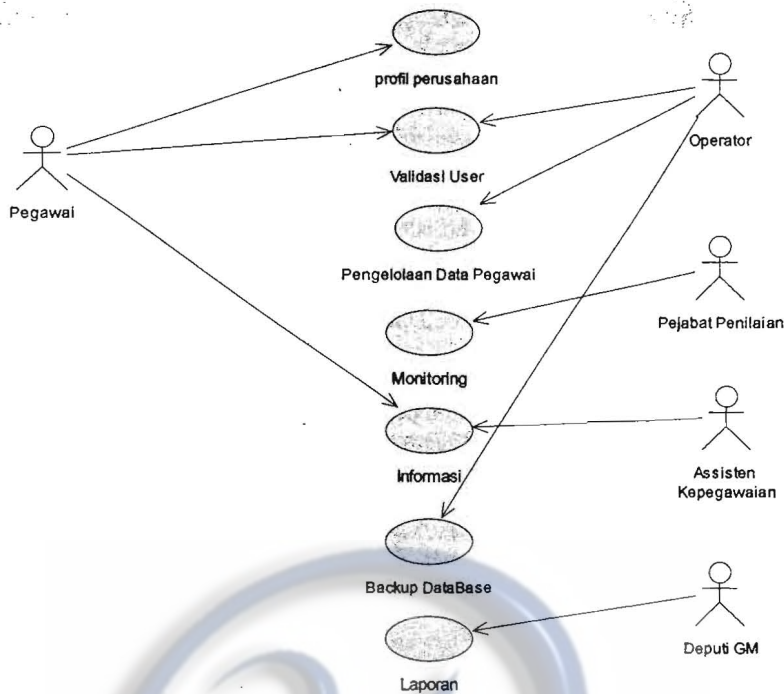
Fase Dalam RUP [3]

Terdapat empat tahap pengembangan perangkat lunak yaitu:

- *Fase Inception*, Pada tahap ini pengembang mendefinisikan batasan kegiatan, melakukan analisis kebutuhan user, dan melakukan perancangan awal perangkat lunak (perancangan arsitektural dan *use case*). Pada akhir fase ini, prototipe perangkat lunak versi *Alpha* harus sudah dirilis
- *Fase Elaboration*, Pada tahap ini dilakukan perancangan perangkat lunak mulai dari menspesifikasikan fitur perangkat lunak hingga perilsan prototipe versi *Betha* dari perangkat lunak.
- *Fase Construction*, Pengimplementasian rancangan perangkat lunak yang telah dibuat dilakukan pada tahap ini. Pada akhir tahap ini, perangkat lunak versi akhir yang sudah disetujui administrator dirilis beserta dokumentasi perangkat lunak.
- *Fase Transition*, Instalasi, *deployment* dan sosialisasi perangkat lunak dilakukan pada tahap ini

Hasil dan Pembahasan

Model Use Case



Gambar 2 Use Case Diagram

Definisi Use Case

No	Use Case	Deskripsi
1	ValidasiUser	Validasi untuk semua user sistem sebelum masuk ke dalam sistem
2	KelolaDataPegawai	Melakukan pengelolaan (insert, update, delete, cari) terhadap data Pegawai
3	monitor	Melakukan monitoring (garfik Pegawai dan nilai rata-rata pegawai)
4	informasi	Melihat informasi pegawai
5	Backup database	Melakukan backup data Base
6	Laporan	Melakukan pengelolaan terhadap data Laporan

Skenario Use Casse

Skenario (flow of event) untuk masing-masing use case dari Sistem Informasi Kenaikan Pangkat Pegawai terdiri dari skenario use case validasi user, kelola data pegawai, data monitor, informasi, backup database, dan laporan

Tabel 1 Skenario Use Case ValidasiUser

Name	ValidasiUser
Level	Level Pengguna
Aktors	Oprator
Goal	Untuk menjaga keamanan data (security)
Precondition	User belum valid dan belum mendapatkan hak akses
Postcondition	User telah valid dan mendapatkan hak akses untuk masuk sistem
Steps	1. User memasukkan ID dan password 2. User menekan tombol login 3. Sistem memberikan pesan bahwa user telah login dan memunculkan page atau menu sesuai hak akses user tersebut.
Variation	#1 Sistem memberikan pesan bahwa username atau password tidak terdaftar

Tabel 2 Skenario Use Case KelolaDataPegawai

Name	KelolaDataPegawai
Level	Level Pengguna
Aktors	Oprator
Goal	Untuk mengelola data-data pegawai
Precondition	Oprator telah valid sebagai user system
Postcondition	Oprator berhasil dimanipulasi (<i>insert, update, delete, cari</i>)
Steps	<ol style="list-style-type: none"> 1. oprator memilih menu kelola data pegawai 2. oprator menginputkan, meng-update, menghapus atau melakukan pencarian data pegawai 3. Sistem memunculkan pesan bahwa data pegawai telah berhasil ditambahkan, di-update, dihapus atau ditemukan.
Variation	#2 Data pegawai yang diinputkan tidak lengkap, sistem memunculkan pesan bahwa data yang diinput belum lengkap #2 Data pegawai yang akan dicari belum terdaftar, sistem memunculkan pesan

Tabel 3 Skenario Use Case Laporan

Name	Laporan
Level	Level Pengguna
Aktors	Oprator
Goal	Untuk mengelola laporan
Precondition	Oprator telah valid sebagai user sistem, data pegawai , data Dp3,laporan KPP,laporan kenaikan pangkat reguler,kenaikan pangkat istimewa,kenaikan pangkat akhir bakti sudah berada di database.
Postcondition	Laporan sudah terimpan di database
Steps	<ol style="list-style-type: none"> 1. Oprator memilih menu laporan 2. Oprator melakukan pencarian data pegawai 3. Sistem memunculkan pesan bahwa laporan telah berhasil disimpan
Variation	#2 laporan yang diinputkan tidak lengkap, sistem memunculkan pesan bahwa data yang diinput belum lengkap #2 laporan yang akan dicari belum ada, sistem memunculkan pesan bahwa laporan belum terdaftar

Deskripsi Arsitektur Tahap Analisis

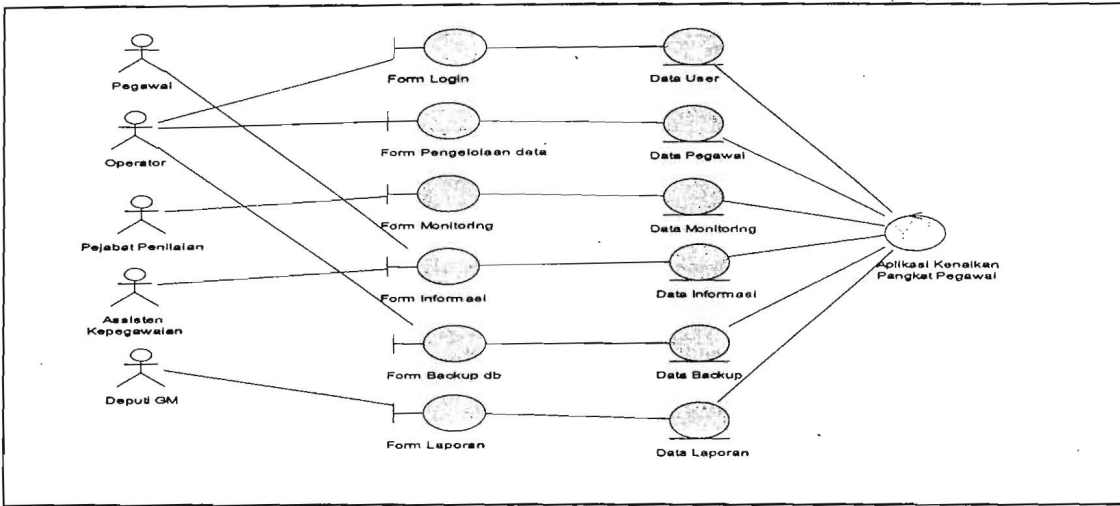
Tabel 7 Tabel identifikasi Paket Analisis

No	Nama Paket	Use Case Terkait	Keterangan
1	Paket SI Kenaikan Pangkat Pegawai	<ol style="list-style-type: none"> 1. Use case validasi 2. Use case kelola data pegawai 3. Use case monitoring 4. Use case informasi 5. Use case backup database 6. Use case Laporan 	

Tabel 8 Class Analisis

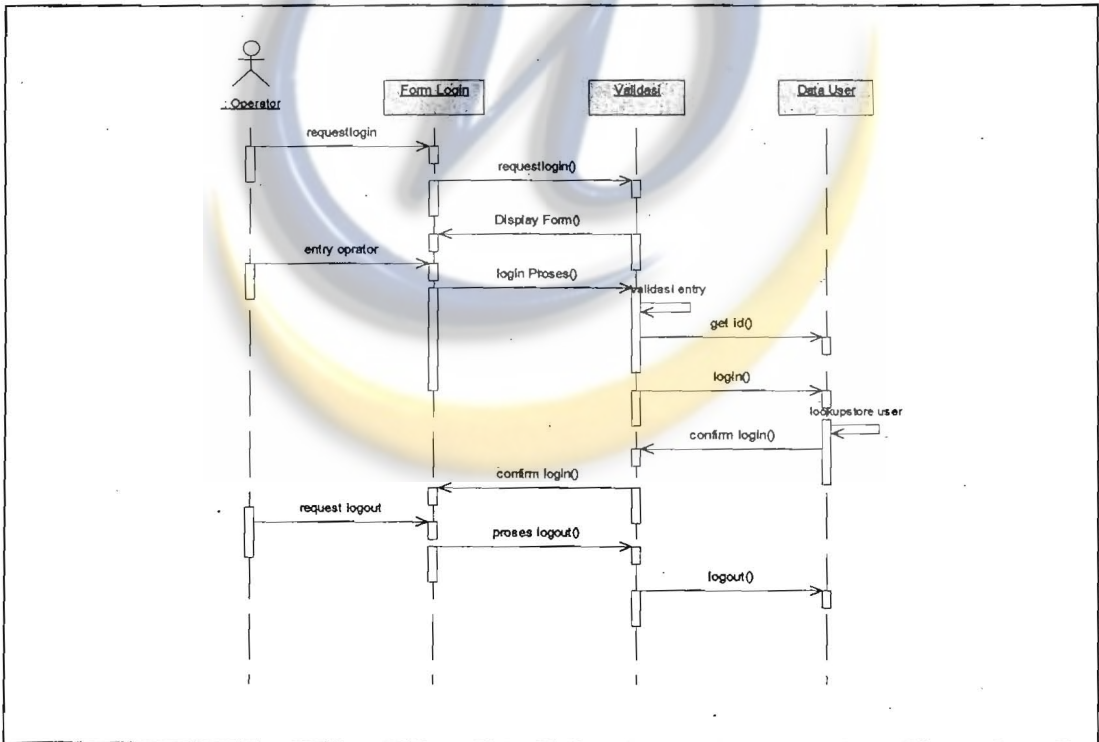
No	Nama Kelas	Jenis	Keterangan
1	Paket SI Kenaikan Pangkat Pegawai	<ol style="list-style-type: none"> 1. Validasi 2. Form Kelola data pegawai 3. Form pengelolaan data 4. Form monitoring 5. Form laporan 6. data pegawai 7. data monitor 8. data informasi 9. data backup database 10. data user 11. laporan 	Control Boundery Boundery Boundery Boundery Entity Entity Entity Entity Entity Entity

Realisasi Use Case Tahap Analisis



Gambar 3 Class Analisis Diagram

Sistem Sequence Diagram terdiri dari sistem sequence diagram validasi, pengelolaan data pegawai, monitor pegawai, diagram informasi, backup database dan pembuatan laporan



Gambar 4 Sistem Sequence Diagram Validasi

Prototype Antar muka terdiri dari desain tampilan login, data pegawai, pengolahan DP3, pengolahan SK, pengolahan KPP, rata-rata nilai pegawai, informasi pegawai, informasi DP3, informasi SK, Kenaikan pangkat, backup database, lagout

Gambar 5 Desain Tampilan Login

Gambar 6 Desain Tampilan Pengolahan DP3

Implementasi

Implementasi meliputi tampilan login, data pegawai, pengolahan DP3, pengolahan SK, pengolahan KPP, rata-rata nilai pegawai, informasi pegawai, informasi DP3, informasi SK, Kenaikan pangkat, backup database, layout

Gambar 7 Tampilan Login

Pengolahan Nilai DP3

Data Pegawai

NPP: 131261000
 Nama: ADMIN
 Golongan: IV / A
 Pangkat: PEMBINA
 Jabatan: ADMINISTRATOR

NPP Pegawai: 131261002
 Nama: ACEP YADI
 Golongan: II / C
 Pangkat: PENGATUR
 Jabatan: STAF PENGADAAN BARANG

Data Nilai

No:
 Keseluruhan: 90 Tanggung Jawab: 86 Tgl. Penyerahan: 10/3/2007 Grady: A
 Prestasi: 98 Kerjasama: 98 Tgl. Pembuatan: 10/3/2007 Keterangan:
 Kejujuran: 98 Prakarsa: 65 Jumlah: 748 **Lulus**
 Ketepatan: 98 Kepemimpinan: 95 Kata-rata: 93

PENGOLAHAN NILAI DP3 PERUMNAS REGIONAL IV BANDUNG									
NO Penilai	NPP	Keseluruhan	Prestasi	Tanggung Jawab	Ketepatan	Kejujuran	Kerjasama	Prakarsa	Kepemimpin.
3	131261006	80	99	97	78	98	90	77	
5	131261007	98	90	99	90	89	98	99	
6	131261003	90	90	97	95	89	98	99	

Gambar 8 Tampilan Pengolahan Data Pegawai

Daftar Pustaka

Benneth, Simon, "Object-Oriented Systems Analysis and Design Using UML", Mc Graw-Hill(2002), pp 119-195

Booch, Grady, "Object-Oriented Analysis and Design with application", Addison-Westley (2001) ,pp 35-39

ITB, "Template Dokumen RUP"

Hariman gunadi, Suhendar,, Visual Modelling menggunakan UML dan rational Rose, Informatika, Bandung, (2002)

Adi Nugroho, Rational Rose Untuk Pemodelan Berorientasi Objek, Penerbit Informatika, Tahun 2005.

http://en.wikipedia.org/wiki/Object-oriented_analysis_and_design diakses pada tanggal 11 november 2010

Pressman, Roger S, "Rekayasa Perangkat Lunak Pendekatan Praktis", Mc Graww Hill, (2003),pp 651-717

Schmuller, Joseph, "Teach Yourself UML in 24 hours", Sams, Indianapolis,(2004),pp 7-26