

APLIKASI PENGELOLAAN SURAT IZIN GANGGUAN HO (HINDER ORDONANSI) DI DINAS PENANAMAN MODAL DAN PELAYANAN TERPADU

Siti Nazilah
Program Studi Teknik Informatika
Universitas Suryakancana
Zilah.nazilah@gmail.com

Yuli Yuliani
Program Studi Teknik Informatika
Universitas Suryakancana
Yuliyuliani398@gmail.com

Abstrak

Dinas Penanaman Modal dan Pelayanan Terpadu (DPMPT) merupakan Unit Pelayanan Terpadu (UPT) yang mengelola penanaman modal dan perusahaan perijinan, salah satunya yaitu izin gangguan / HO (*Hinder Ordonansi*) yang bertujuan sebagai sebuah perlindungan hukum dari pemerintah terhadap tempat usaha yang dikelola. Pada saat ini dalam melakukan pengelolaan surat izin gangguan sudah terkomputerisasi dengan menggunakan *microsoft office* dan perhitungan retribusi sederhana dengan menggunakan kalkulator namun belum maksimal karena data juga masih bersifat fisik atau kertas. Hal itu menyebabkan masalah diantaranya proses pencarian data yang lama, sering terjadi kesalahan dalam pengelolaan surat dan ketidakakuratan perhitungan. Dalam pembuatan perangkat lunak ini menggunakan metode pengembangan sistem berdasarkan paradigma *waterfall* yang terdiri dari *Communication, Planning, Modlling, Contruction, Deployment* sedangkan untuk metode perancangan sistem menggunakan UML. Bahasa Pemrograman yang digunakan adalah *IDE Delphi XE 8* dan *MySQL* sebagai DBMS. Pembuatan aplikasi berbasis dekstop dibuat sedemikian rupa agar dapat memenuhi kebutuhan pengguna dan menghasilkan sistem yang membantu pelaksanaan kinerja di instansi, sehingga memudahkan dalam pendataan data, mempermudah dalam pembuatan surat serta pelaporan kegiatan surat izin gangguan / HO.

Kata kunci: Aplikasi, Retribusi, Pengelolaan Surat Izin Gangguan / HO

1. Pendahuluan

Kebutuhan manusia akan sebuah informasi yang akurat mendorong perkembangan teknologi dan informasi semakin berkembang dengan pesat. Hal ini dapat dibuktikan dengan banyaknya inovasi – inovasi yang telah dibuat. Sebuah informasi ini sangat dibutuhkan untuk melakukan proses pengelolaan data. Salah satunya yaitu pengelolaan surat izin gangguan yang semula hanya menggunakan catatan – catatan yang dibuatkan sebuah buku, namun saat ini semua proses pengelolaan surat sudah terkomputerisasi dengan baik. Dinas Penanaman Modal Dan Pelayanan Terpadu (DPMPT) merupakan salah satu Kantor Dinas yang mengelola penanaman modal dan perusahaan perijinan berlokasi di Jl. Raya Bandung KM.03 Cianjur. Izin gangguan atau biasa didefinisikan sebagai izin HO (*Hinder Ordonansi*) adalah salah satu dokumen wajib yang dimiliki oleh setiap badan usaha / perorangan di lokasi tertentu yang berpotensi menimbulkan dampak bahaya, kerugian, serta gangguan bagi wilayah sekitar kegiatan usaha. Pembuatan surat izin gangguan / HO ini bukan tanpa tujuan tetapi dimaksudkan untuk pembinaan, pengaturan, pengendalian, dan pengawasan atas kegiatan pemanfaatan ruang, penggunaan sumber daya alam, barang, prasarana atau fasilitas tertentu guna melindungi kepentingan umum dan menjaga kelestarian lingkungan. Proses pengelolaan surat izin gangguan

sudah menggunakan komputer berupa Microsoft office dan perhitungan sederhana dengan menggunakan kalkulator. Hal tersebut yang menyebabkan proses pengelolaan surat izin gangguan menjadi tidak efektif dan efisien, maka dari itu sering terjadi kesalahan dalam pengelolaan surat dan ketidakakuratan perhitungan, saat pihak pemohon datang untuk membuat surat izin gangguan, belum adanya sistem yang bisa melakukan perhitungan retribusi izin gangguan, kemudian belum adanya sistem yang mampu mengelola atau membuat laporan surat izin gangguan. Sebaiknya hal tersebut harus sudah terkomputerisasi sehingga memudahkan admin untuk melakukan semua proses pengelolaan surat izin gangguan / HO. Oleh karena itu diperlukan aplikasi yang tepat, sehingga diharapkan aplikasi tersebut dapat membantu menyelesaikan permasalahan dengan cepat dan akurat. Kemudian aplikasi tersebut juga mampu mengelola pendaftaran, melakukan pengelolaan surat izin gangguan, dan melakukan perhitungan retribusi izin gangguan.

2. Rumusan Masalah

Berdasarkan latar belakang masalah tersebut maka rumusan masalah dari penelitian ini adalah bagaimana membangun sebuah aplikasi pengelolaan surat izin gangguan HO (*Hinder Ordonansi*).

3. Maksud dan Tujuan

A. Maksud

Maksud dari penelitian ini adalah untuk membuat aplikasi yang mampu melakukan proses pengelolaan surat izin gangguan / HO secara cepat dan akurat, serta mampu melakukan perhitungan retribusi izin gangguan.

B. Tujuan

Adapun tujuan yang didapat dari kerja praktek ini adalah :

1. Memudahkan pendaftaran surat izin gangguan / HO
2. Memudahkan perhitungan retribusi izin gangguan / HO.
3. Memudahkan Instansi dalam pembuatan laporan surat izin gangguan / HO.

4. Batasan Masalah

Agar pembahasan masalah Aplikasi Pengelolaan Surat Izin Gangguan HO (Hinder Ordonansi) Di Dinas Penanaman Modal Dan Pelayanan Terpadu terarah dengan baik maka, dalam penelitian ini hanya akan membahas:

- a. Pendaftaran surat izin gangguan / HO.
- b. Data hasil perhitungan retribusi izin.
- c. Pembuatan surat izin gangguan / HO.
- d. Pembuatan laporan surat izin gangguan / HO.

5. Landasan Teori

5.1 Pengelolaan

Pengelolaan adalah kegiatan pemanfaatan dan pengendalian atas semua sumber daya yang diperlukan untuk mencapai ataupun menyelesaikan tujuan tertentu (Prajudi Atmosudirjo, 1982).

Pengelolaan adalah soft skill / keterampilan untuk mencapai suatu hasil tertentu dengan menggunakan tenaga / bantuan orang lain (Sondang P.Siagian, 1997).

5.2 Surat Izin Gangguan / HO

Surat izin gangguan atau HO (Hinder Ordonansi) adalah surat izin kegiatan usaha kepada orang pribadi / badan di lokasi tertentu yang berpotensi menimbulkan dampak bahaya, kerugian, serta gangguan, ketentraman dan ketertiban umum bagi wilayah sekitar kegiatan usaha (Undang-Undang No.28 Tahun 2009).

Izin gangguan adalah pemberian izin tempat usaha/kegiatan kepada orang pribadi/badan di lokasi tertentu yang dapat menimbulkan bahaya, kerugian, dan gangguan, tidak termasuk/kegiatan yang telah ditentukan oleh pemerintah pusat atau pemerintah daerah (pasal 1 angka 3 Permendagri No.27 tahun 2009).

5.3 Retribusi

Retribusi adalah pungutan daerah sebagai pembayaran atas jasa atau pemberian izin tertentu yang khusus disediakan dan / atau diberikan oleh pemerintah daerah untuk kepentingan pribadi atau badan (Undang-Undang No.28 tahun 2009).

Berikut merupakan rumus perhitungan retribusi izin gangguan / HO (Hinder Ordonansi) :

Total Retribusi = (Luas Ruang x Indeks jalan x indeks gangguan x 750) x 50%

Keterangan :

Indek jalan

5 = Jalan Nasional / Provinsi

4 = Jalan Kabupaten

3 = Jalan Desa / Lingkungan

Indek gangguan

5 = Gangguan Berat

4 = Gangguan Sedang

3 = Gangguan Kecil / Ringan

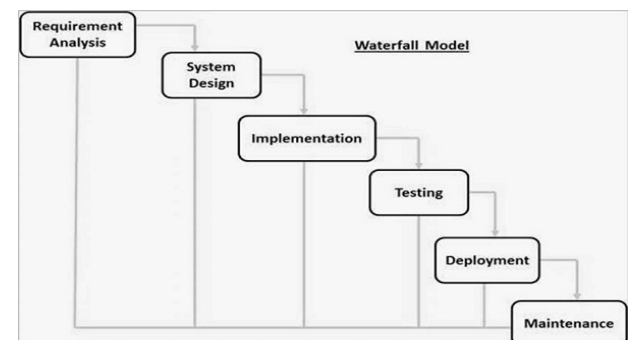
5.4 Unified Modelling Language

UML (Unified Modelling Language) adalah bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasikan artifacts (bagian dari informasi yang digunakan atau dihasilkan oleh proses pembuatan perangkat lunak, artifact tersebut dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya.

Selain itu UML adalah bahasa pemodelan yang menggunakan konsep orientasi object. UML dibuat oleh *Grady Booch*, James Rumbaugh, dan Ivar Jacobson di bawah bendera Rational Software Corp. UML menyediakan notasi-notasi yang membantu memodelkan sistem dari berbagai perspektif, UML tidak hanya digunakan dalam pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan. Terdapat beberapa bagian utama dari UML yaitu *View*, *Diagram*, *Model element*, dan *general mechanism*.

5.5 Metode Rekayasa Perangkat Lunak

Metode Rekayasa Perangkat Lunak yang digunakan adalah metode *waterfall*. Menurut Pressman metode *waterfall* adalah model klasik yang bersifat sistematis, berurutan dalam membangun software, model ini mengusulkan sebuah pendekatan kepada pengembangan software yang sistematis dan sekuensial yang mulai dari tingkat kemajuan sistem pada seluruh analisis, desain, kode, pengujian dan pemeliharaan. Model ini melingkupi aktivitas-aktivitas sebagai berikut : Rekayasa dan pemodelan sistem informasi, analisis kebutuhan, desain, coding, pengujian dan pemeliharaan. Tahapan Metode *Waterfall* terdapat pada gambar 5.1



Gambar Tahapan Metode *Waterfall* 5.1

Dalam pengembangan model *Waterfall* memiliki beberapa tahapan yang berurut yaitu : *requirement* (analisis kebutuhan), *design system* (desain sistem), *Coding* (pengkodean), *testing* (pengujian), penerapan program, pemeliharaan, tahapan-tahapan dari metode *waterfall* adalah sebagai berikut:

a. *Requiremen Analisis*

Tahap ini pengembang sistem diperlukan komunikasi yang bertujuan untuk memahami

perangkat lunak yang diharapkan oleh penggunaan batasan perangkat lunak tersebut. Informasi ini biasanya dapat diperoleh melalui wawancara, diskusi atau survey langsung. Informasi dianalisis untuk mendapatkan data yang dibutuhkan oleh pengguna.

b. *System Design*

Spesifikasi kebutuhan dari tahap sebelumnya akan dipelajari dalam fase ini dan desain sistem disiapkan. Desain sistem membantu dalam menentukan perangkat keras (*hardware*) dan sistem persyaratan dan juga membantu dalam mendefinisikan arsitektur secara keseluruhan.

c. *Implementation*

Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut unit, yang terintegrasi dalam tahap selanjutnya. Setiap unit dikembangkan dan diuji untuk fungsionalitas yang disebut sebagai *unit testing*.

d. *Integration & Testing*

Seluruh *unit* yang dikembangkan dalam tahap implementasi diintegrasikan kedalam sistem setelah pengujian yang dilakukan masing-masing *unit*. Setelah integrasi seluruh sistem diuji untuk mengecek setiap kegagalan maupun kesalahan.

e. *Operation & Maintenance*

Tahapan akhir dalam model *waterfall*. Perangkat lunak yang sudah jadi, dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang ditemukan pada langkah sebelumnya. Perbaikan implementasi *unit* sistem dan peningkatan jasa sistem sebagai kebutuhan baru.

Kelebihan dari menggunakan metode *waterfall* ini memungkinkan untuk departementalisasi dan kontrol. Proses pengembangan model *fase one by one*, sehingga meminimalis kesalahan yang mungkin akan terjadi. Pengembangan bergerak dari konsep, yaitu melalui desain, implementasi, pengujian, instalasi, penyelesaian masalah dan berakhir di operasi pemeliharaan.

Kekurangan Metode *Waterfall* adalah metode ini tidak memungkinkan untuk banyak revisi jika terjadi kesalahan dalam prosesnya. Karena setelah aplikasi ini dalam tahap pengujian, sulit untuk kembali lagi dan mengubah sesuatu yang tidak terdokumentasi dengan baik dalam tahap konsep sebelumnya.

5.6. Tool-tool yang digunakan

Agar memudahkan dalam pembuatan sistem, maka diperlukan beberapa alat atau *tool-tool* yang digunakan dalam pembuatan aplikasi dan teknik untuk proses pelaksanaan pembuatan sistem baru

a. Diagram UML

Diagram berbentuk grafik yang Menunjukkan simbol elemen model yang disusun untuk mengilustrasikan bagian atau aspek tertentu dari sistem. Sebuah diagram merupakan bagian dari suatu *view* tertentu dan ketika digambarkan biasanya dialokasikan untuk *view* tertentu. Adapun jenis-jenis diagram antara lain:

1) *Use case Diagram*

Use case Diagram menggambarkan sejumlah *external actors* dan hubungannya ke *Use case* yang diberikan oleh sistem. *Use case* adalah deskripsi fungsi yang disediakan oleh sistem dalam bentuk teks sebagai dokumentasi dari *Use case symbol* namun dapat juga dilakukan dalam *activity diagrams*.

2) *Class diagram*

Class diagram adalah pandangan aplikasi yang bersifat statis. *Class diagram* tidak hanya menggambarkan visualisasi, tetapi juga menggambarkan dan mendokumentasikan aspek yang berbeda dalam sistem, tetapi juga untuk konstruksi eksekusi kode dalam *software* aplikasi. *Class diagram* digunakan untuk mengelompokkan hal-hal inti dari setiap proses yang ingin dilakukan. Semua proses dimasukkan ke dalam tiap-tiap *Class* dan saling dihubungkan pada *Class-Class* lainnya yang saling berhubungan.

3) *Activity Diagram*

Diagram ini menggambarkan rangkaian aliran dari aktivitas didalam sebuah system yang bersifat dinamis, diagram ini digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti *Use case* atau interaksi.

4) *Sequence diagram*

Diagram ini menggambarkan kolaborasi dinamis antara sejumlah *object* yang bersifat dinamis. Kegunaanya untuk Menunjukkan rangkaian pesan yang dikirim terhadap *object* juga interaksi antara *object*, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.

5) *Communication Diagram*

Diagram ini berfungsi sebagai pengganti diagram kolaborasi yang menekankan organisasi structural dari objek-objek yang menerima serta mengirim pesan.

6) *Component Diagram*

Diagram komponen ini memperlihatkan organisasi serta ketergantungan sistem/perangkat lunak pada komponen-komponen yang telah ada sebelumnya. Diagram ini berhubungan dengan diagram kelas dimana komponen secara tipikal dipetakan kedalam satu atau lebih kelas-kelas, antarmuka-antarmuka serta kolaborasi-kolaborasi.

7) *Deployment Diagram*

Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan (*run-time*) yang bersifat statis. Memuat simpul-simpul beserta komponen-komponen yang ada di dalamnya. Diagram *deployment* berhubungan erat dengan diagram komponen dimana diagram ini memuat satu atau lebih komponen-komponen. Diagram ini sangat berguna saat aplikasi kita berlaku sebagai aplikasi yang dijalankan pada banyak mesin (*distributed computing*).

8) *Package Diagram*

Diagram ini memperlihatkan kumpulan kelas-kelas, merupakan bagian diagram komponen yang bersifat statis.

9) *Statechart Diagram*

Menggambarkan semua *state* (kondisi) yang dimiliki oleh suatu *object* dari suatu *Class* dan keadaan yang menyebabkan *state* berubah. Kejadian dapat berupa *object* lain yang mengirim pesan.

b. Delphi

Delphi adalah sebuah Lingkungan Pengembangan Terpadu (IDE) untuk mengembangkan aplikasi konsol, desktop, website ataupun perangkat *mobile*. Produk ini pada awalnya dikembangkan oleh *Code Gear* sebagai divisi pengembangan perangkat lunak milik, divisi tersebut sebelumnya adalah milik Borland. Bahasa Delphi, atau dikenal pula sebagai object pascal (pascal dengan ekstensi Pemrograman Berorientasi Objek (PBO/OOP) pada mulanya ditujukan hanya untuk *Microsoft Windows*, namun saat ini telah mampu digunakan untuk mengembangkan aplikasi untuk Mac Os X, iOS, Microsoft .Net framework.

Keuntungan :

Komunitas pengguna yang besar pada usenet maupun web.

1. Dapat mengkompilasi menjadi single executable (aplikasi portable), memudahkan distribusi dan meminimalisir masalah yang terkait dengan versioning.
2. Banyaknya dukungan dari pihak ketiga terhadap VCL (biasanya tersedia berikut *source* codenya) ataupun tools pendukung lainnya (dokumentasi, tool debugging).
3. Optimasi kompilasi yang cukup cepat.
4. Mendukung *multiple platform* dari *source code* yang sama.

Kerugian :

1. *Partial single vendor lock-in* (Borland dapat menetapkan standar bahasa, kompatibilitas yang harus mengikutinya).
2. Akses pada platform dan library pihak ketiga membutuhkan file-file header yang diterjemahkan ke dalam bahasa pascal.
3. Dokumentasi atas *platform* dan teknik-teknik yang menyertainya sulit ditemukan dalam bahasa pascal (contoh akses COM dan Win32).

c. XAMPP

Xampp menurut Adibowo (2010, 5) singkatan dari X (empat *system* operasi apapun), *Apache*, *MySQL*, *PHP*, *Perl*. XAMPP merupakan tool yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dalam paketnya sudah terdapat *Apache (web server)*, *MySQL (database)*, *PHP (server side scripting)*, *Perl*, *FTP server.phpMyAdmin* dan berbagai pustaka bantu lainnya. Dengan menginstall XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi web server *Apache*, *PHP* dan *MySQL* secara manual. XAMPP akan menginstallasi dan mengkonfigurasikannya secara otomatis.

d. MySQL

MySQL merupakan salah satu *software* untuk *database server* yang banyak digunakan, MySQL

bersifat *Open Source* dan menggunakan SQL. MySQL memiliki beberapa kelebihan, antara lain :

1. Kapabilitas dan pembatasan. MySQL mampu menangani basis data dalam skala besar, dengan jumlah rekaman (*records*) lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu, batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.
2. Konektivitas. MySQL dapat melakukan koneksi dengan *Client* menggunakan protokol TCP/IP, *Unix* socket (UNIX), atau *Named Pipes* (NT).
3. Lokalisasi. MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa.

Antar Muka. MySQL memiliki *interface* (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*). Portabilitas. MySQL dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi

e. Visual Paradigm

Visual Paradigm merupakan aplikasi untuk merancang sebuah aplikasi atau biasa disebut aplikasi rekayasa perangkat lunak. Dengan Visual Paradigm sebuah aplikasi dapat digambarkan dalam sebuah rancangan simbol dan gambar tanpa koding yang menjelaskan bagaimana aplikasi tersebut akan berjalan setelah selesai nantinya. Visual Paradigm adalah salah satu alat bantu *Unified Modelling Language* (UML) yang digunakan untuk membuat *Use case Diagram*, *Activity Diagram*, *Class diagram*, dan *Sequence diagram* [1].

6. Analisis dan Perancangan

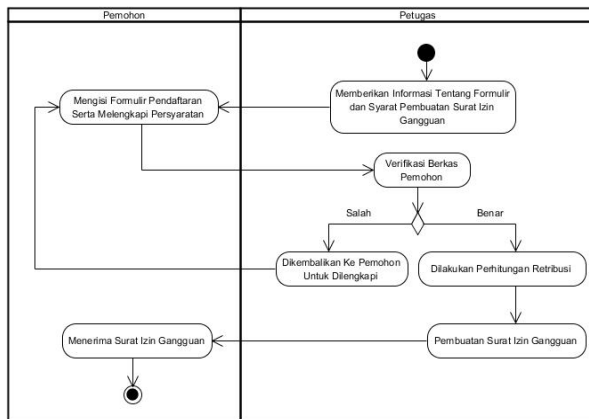
Analisis perancang dalam pembuatan Aplikasi Pengelolaan Surat Izin Gangguan Ho (Hinder Ordonansi) Di Dinas Penanaman Modal Dan Pelayanan Terpadu

6.1 Analisis Sistem yang Sedang Berjalan

Dinas Penanaman Modal Dan Pelayanan Terpadu merupakan Unit Pelayanan Terpadu (UPT) yang memberikan surat dalam bidang perizinan dengan tugas melaksanakan sebagian fungsi dinas di bidang pelayanan salah satunya izin gangguan / HO yang di ajukan pemohon dalam proses pembuatan surat izin gangguan / HO. Pemohon mendatangi Dinas Penanaman Modal Dan Pelayanan Terpadu untuk melakukan pembuatan surat izin gangguan / HO. Pemohon melengkapi berkas permohonan diantaranya formulir pendaftan, dan berkas persyaratan lainnya yang selanjutnya di lakukan verifikasi berkas permohonan, jika masih kurang berkas di kembalikan kepada pemohon untuk melengkapi ulang berkas permohonan. Jika telah terdaftar sebagai pemohon dan persyaratan telah terpenuhi maka akan dilakukan proses perhitungan retribusi izin gangguan / HO. Kemudian akan dibuatkan surat izin gangguan / HO. Setelah surat izin gangguan / HO selesai maka diserahkan kepada pemohon

6.2 Swimlane Sistem Yang Sedang Berjalan

Berikut gambar 6.1 *swimlane diagram* dari sistem yang sedang berjalan di Dinas Penanaman Modal Dan Pelayanan Terpadu bagian pengelolaan surat izin gangguan / HO :



gambar 6.1 *swimlane diagram*

6.3 Use Case Diagram

Use case diagram digunakan untuk memperlihatkan 3 aspek sistem : aktor, use case, dan lingkup sistem.

Memperlihatkan hubungan aktor dan fungsi yang dapat dilakukan oleh aktor dalam ruang lingkup sistem.

1. Daftar aktor

Aktor merupakan representasi orang-orang atau perangkat yang menggunakan sistem dan memiliki sifat serta fungsi yang dijelaskan dalam sebuah konteks yang memiliki peran terhadap sistem. Aktor yang terlibat disini yaitu admin.

2. Daftar Use Case

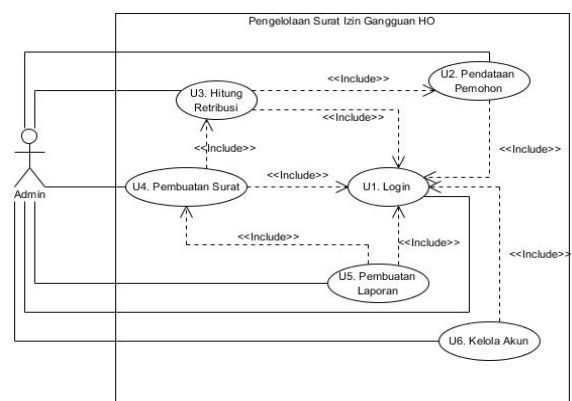
Daftar Use case menjelaskan tentang daftar Use case yang ada didalam sistem, daftar Use case berisi kode, nama Use case dan keterangan dari tiap masing-masing Use case.

Kode	Sistem Use case	Keterangan
U1	Login	Menggambarkan Login ke aplikasi yang dilakukan oleh Admin.
U2	Pendataan Pemohon	Digunakan untuk melakukan pendataan data pemohon dan verifikasi persyaratan surat izin.

Kode	Sistem Use case	Keterangan
U3	Hitung Retribusi	Digunakan untuk melakukan perhitungan biaya retribusi izin gangguan / HO.
U4	Pembuatan Surat	Digunakan untuk melakukan pengelolaan surat izin gangguan / HO.
U5	Pembuatan Laporan	Digunakan untuk memberikan hasil laporan dari setiap data yang diinputkan.
U6	Kelola Akun	Digunakan untuk pengelolaan data akun.

3. Use Case Diagram

Use case diagram mendeskripsikan kelakuan sistem dari sudut pandang pengguna, berguna untuk membantu memahami kebutuhan. Use case adalah dasar dari diagram lain.



Keterangan :

Berdasarkan Gambar 3.2 Use case Diagram Aplikasi Pengelolaan Surat Izin Gangguan / HO, terdapat 1 aktor dan 6 use case yaitu admin. Admin harus terlebih dahulu melakukan Login kemudian admin mendapatkan hak akses penuh terhadap aplikasi pengelolaan surat izin / HO. Mulai dari mengelola pendataan data pemohon, menghitung retribusi, mengelola surat izin gangguan / HO, pembuatan laporan dan pengelolaan data akun.

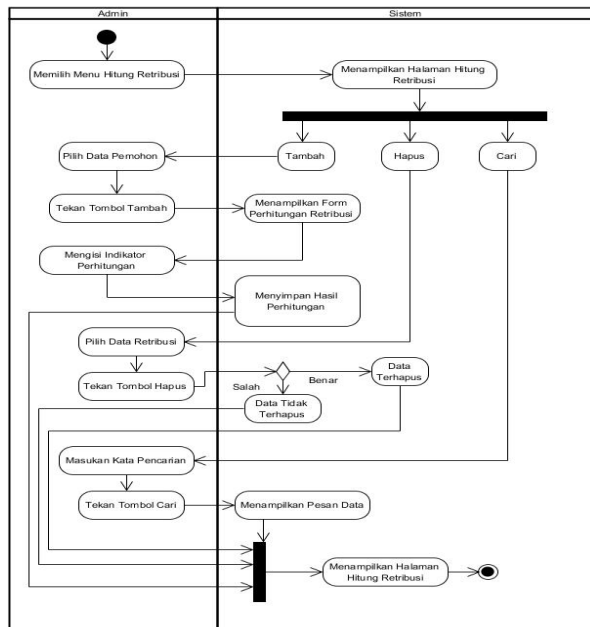
6.4 Swimlane diagram

Swimlane diagram merupakan diagram yang memperlihatkan aliran aktivitas-aktivitas yang dideskripsikan oleh Use case dan pada saat yang

bersamaan memperlihatkan aktor mana atau kelas analisa mana yang bertanggung jawab untuk aksi tertentu yang dideskripsikan oleh kotak aktivitas

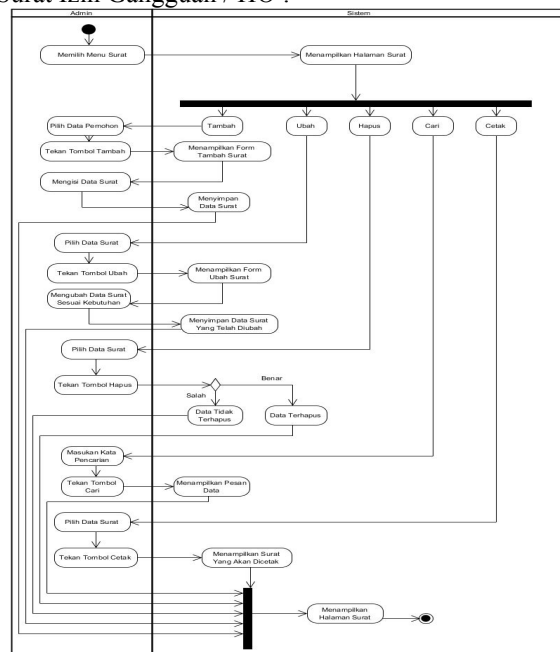
a. *Swimlane diagram* Hitung Retribusi

Didalam *Swimlane diagram* Hitung Retribusi, hitung retribusi digunakan untuk menghitung data retribusi pemohon, admin akan mengisi indikator perhitungan retribusi kemudian sistem akan menghitung retribusi dan menampilkan hasil perhitungan retribusi. Berikut gambar 3.5 *Swimlane diagram* Hitung Retribusi dari Aplikasi Pengelolaan Surat Izin Gangguan / HO :



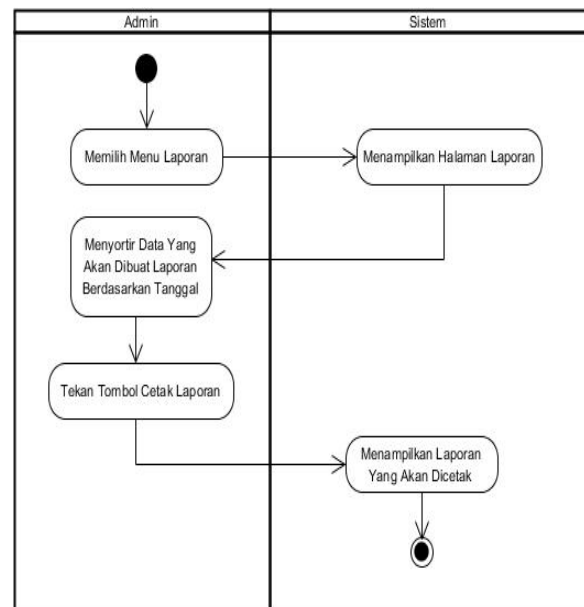
b. *Swimlane diagram* Pembuatan Surat

Didalam *Swimlane diagram* Pembuatan Surat, pembuatan surat digunakan untuk mengelola surat yang akan dibuat dan menghasilkan hasil akhir surat izin gangguan / HO. Berikut gambar 3.6 *Swimlane diagram* Pembuatan Surat dari Aplikasi Pengelolaan Surat Izin Gangguan / HO :



c. *Swimlane diagram* Pembuatan Laporan

Didalam *Swimlane diagram* Pembuatan Laporan, pembuatan laporan digunakan untuk mendata banyak surat yang terdaftar setiap bulannya dan juga mencetak hasil laporan tersebut. Berikut gambar 3.7 *Swimlane diagram* Pembuatan Laporan dari Aplikasi Pengelolaan Surat Izin Gangguan / HO :



6.5 Data Modeling

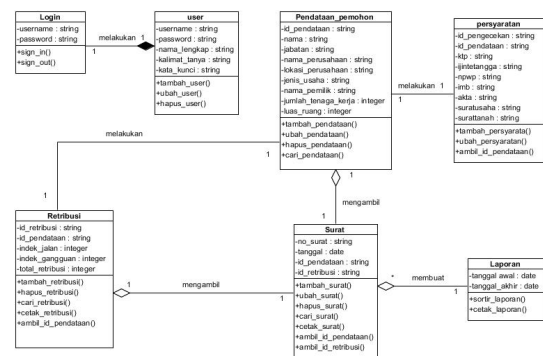
Data Modeling adalah object data yang diproses didalam sistem beserta relasi antar object data. Object data merupakan representasi dari gabungan informasi yang harus dipahami oleh *software* atau sistem. (Pressman, 2010).

A. Class diagram

Class diagram adalah pandangan aplikasi yang statis. *Class diagram* tidak hanya menggambarkan visualisasi, menggambarkan dan mendokumentasikan aspek yang berbeda dalam sistem, tetapi juga untuk konstruksi eksekusi kode dalam *software* aplikasi.

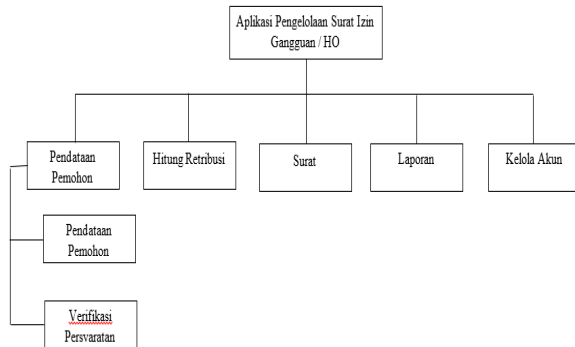
Class diagram digunakan untuk mengelompokkan hal-hal inti dari setiap proses yang ingin dilakukan. Semua proses dimasukkan ke dalam tiap-tiap *Class* dan saling dihubungkan pada *Class-Class* lainnya yang saling berhubungan.

Berikut gambar 6.2 *Class diagram* dari Aplikasi Pengelolaan Surat Izin Gangguan / HO :



6.6 Perancangan Stuktur Menu

Perancangan struktur menu merupakan gambaran jalur pemakaian aplikasi pengelolaan surat izin gangguan / HO yang akan dibangun. Berikut merupakan Struktur Menu dari Aplikasi Pengelolaan Surat Izin Gangguan / HO :



A. Perancangan Antar Muka

Menggambarkan bagaimana *software* berkomunikasi dengan sistem dan manusia yang menggunakannya. Antarmuka menggambarkan aliran informasi dan prilakunya. (Pressman, 2010).

1. Antar Muka Halaman Login

Halaman login merupakan tampilan awal dari aplikasi pengelolaan surat izin gangguan / HO. Admin harus melakukan login terlebih dahulu untuk mendapatkan akses terhadap sistem dengan memasukkan *username* dan *password*.

2. Antar Muka Lupa Password

Halaman Lupa Password merupakan halaman yang digunakan untuk membantu menampilkan *password user* ketika *user* tidak mengingat *password* mereka. *Password* dapat diketahui ketika *user* memasukkan *username*, memilih kalimat tanya dan memasukkan kata kunci. Ketika ketiga data cocok dengan yang ada di database maka sistem akan menampilkan *password user* tersebut

3. Antar Muka Halaman Utama

Halaman Utama merupakan halaman ketika *user* berhasil login dan mempunyai hak akses penuh terhadap aplikasi.

6.7 Implementasi Dan Pengujian Sistem

Tahap implementasi dapat diterapkan jika sistem perangkat lunak dan perangkat kerasnya telah dipersiapkan. Kegiatan implementasi sistem ini meliputi kebutuhan perangkat lunak, perangkat keras, instalasi aplikasi dan implementasi antarmuka, serta pengujian aplikasi sistem yang telah dirancang sesuai dengan kebutuhan organisasi.

7. Simpulan

Berdasarkan hasil analisis selama melaksanakan penelitian di Dinas Penanaman Modal dan Pelayanan Terpadu yang berjudul Aplikasi Pengelolaan Surat Izin Gangguan / HO , dapat disimpulkan bahwa :

1. Memudahkan admin dalam melakukan proses pengelolaan data surat izin gangguan dengan cepat dan tepat;
2. Memudahkan admin melakukan perhitungan retribusi menjadi lebih cepat;
3. Membantu admin dalam membuat laporan menjadi lebih efektif dan efisien.

Saran

Untuk meningkatkan kinerja Aplikasi Pengelolaan Surat Izin Gangguan / HO, maka penulis mengajukan beberapa saran antara lain :

1. Untuk pengembangan selanjutnya dapat dibuatkan aplikasi berbasis web sehingga pemohon dalam melakukan proses pendaftaran tidak perlu datang ke kantor dinas secara langsung.
2. Perlu ada pemeliharaan sistem yang baik menyangkut pemeliharaan data, pemeliharaan perangkat pengelolaan data, maupun pemeliharaan aplikasi.

9. Daftar pustaka

- [2] Hoffer, Prescott and McFadden., 2005:5.
- [3] Yulianto, Ardhian Agung, dkk., 2009:p2.
- [4] Yulianto, Ardhian Agung, dkk., 2009:2.
- [5] Yulianto, Ardhian Agung, dkk., 2009:2.
- [1] Parsimin, *Membangun Sistem Informasi Berbasis Website Pada SMP Negeri 4 Purwanegara*. Yogyakarta: AMIKOM, 2013.
- [6] Jogiyanto, *Analisis & Desain Sistem Informasi : Pendekatan terstruktur teori dan praktek bisnis*. Yogyakarta: Andi, 1999.
- [7] Prajudi Atmosudirdjo, *Administrasi dan Manajemen Umum*. Jakarta: Ghalia Indonesia, 1982.
- [8] Sondang P Siagian, *Manajemen Sumber Daya Manusia*. Jakarta: PT. Toko Gunung Agung, 1997.
- [9] Adibowo. (2010) Definisi Google Body. [Online]. <http://adibowo.com/tag/definisi-google-body-browser/>
- [10] Mary B. Prescott, Fred R. McFadden Jeffrey A. Hoffer, *Modern Database Management.*: Prentice Hall, 2005.
- [11] Yulianto, Dkk Ardian Agung, *Analisis dan Desain Sistem Informasi*. Bandung: Politeknik Telkom, 2009.
- [12] Roger S Presman, *Rekayasa Perangkat Lunak : Pendekatan Praktisi*. Yogyakarta: Andi Yogyakarta, 2010.