

MODEL OBJECT RELATIONAL DATABASE PADA APLIKASI NOTIFIKASI SMS

Indrajani

Information Systems Department, School of Information Systems, Binus University
Jl. K.H. Syahdan No. 9, Palmerah, Jakarta Barat 11480
indrajani@binus.ac.id

ABSTRACT

The purpose of this research is to analyse the database using object oriented approach. In addition, this research develops and designs an object relational database structure to make easier the process of searching for the required information. The benefit of this research is to provide a data structure that can be reused for SMS notification application developer team, so that the developers can use existing objects for future use if there are changes in the current business process. The methods used are data collection, analysis, and design. The result obtained from this research is a database structure of sms notification application, which can be used in various programming languages for using object relational database approach. For conclusion, by using the object relational database structure, application developers will be made easier in developing the current database structure. Programmer can add a new data type simply by creating a new object. Data is accessed or processed as object-oriented.

Keywords: SMS notification, database, model object relational database

ABSTRAK

Tujuan penelitian ini adalah untuk menganalisis struktur database menggunakan pendekatan objek. Selain itu, penelitian ini juga menyusun dan merancang struktur database object relational database yang lebih mudah untuk pencarian informasi yang dibutuhkan. Adapun manfaat penelitian ini adalah memberikan struktur data yang dapat digunakan kembali bagi tim pengembang aplikasi notifikasi sms, sehingga para pengembang tersebut dapat menggunakan objek-objek yang telah ada untuk digunakan kembali jika proses bisnis yang berjalan mengalami perubahan. Metode yang digunakan pengumpulan data, metode analisis, dan metode perancangan. Hasil yang diperoleh dari penelitian ini adalah struktur database aplikasi notifikasi sms, di mana database tersebut dapat digunakan dengan berbagai bahasa pemrograman karena menggunakan pendekatan object relational database. Simpulan yang didapat, dengan menggunakan struktur database object relational database, pengembang aplikasi akan lebih mudah dalam pengembangan struktur database yang berjalan. Programmer hanya menambahkan tipe data baru hanya dengan membuat sebuah obyek baru. Data pun diakses atau diolah dengan cara orientasi objek.

Kata kunci: notifikasi SMS, database, model database relasional objek

PENDAHULUAN

Saat ini penggunaan teknologi informasi tidak dapat dipisahkan lagi dari segala aspek dalam kehidupan sehari-hari, karena penggunaan teknologi informasi dapat membuat berbagai kegiatan menjadi lebih mudah, cepat, dan tepat serta mengurangi kesalahan yang dilakukan oleh manusia. Perusahaan di bidang apapun harus lebih pandai dalam melihat dan memanfaatkan peluang yang disediakan oleh teknologi informasi sekarang ini dalam menunjang serta memaksimalkan fungsi bisnis yang ada. Penerapan suatu teknologi informasi sangat erat kaitannya dengan aplikasi. Suatu aplikasi pasti memiliki hubungan yang sangat erat dengan database. Tanpa sistem database yang benar, maka implementasi suatu teknologi informasi akan menjadi kurang bermanfaat atau dapat dikatakan tidak maksimal penggunaannya. Penerapan sistem database mempunyai beberapa keuntungan antara lain dapat mengontrol redundansi data, data menjadi konsisten, meningkatkan integritas data, adanya standarisasi data, dan meningkatkan produktifitas kerja. Pada setiap aplikasi yang diciptakan memerlukan database. Database merupakan salah satu komponen penting untuk setiap aplikasi, pada perkembangannya, database telah mengalami perubahan yang signifikan dalam pengolahannya.

Tujuan dari penelitian ini adalah menganalisis sistem basis yang berjalan dan memberikan penjelasan mengenai database object relational database. Selain itu juga membangun database object relational database aplikasi Notifikasi SMS.

Ruang lingkup penelitian ini dibatasi pada proses transformasi dari penggunaan model database relasional menjadi model database object relational database. Pembatasan tersebut mencakup database relasional sesuai dengan contoh ERD yang dibuat, proses pembuatan rancangan database object relational database dengan pendekatan pembuatan UML, *Functional Data Model*, dan *Object Data Model*. Yang terakhir adalah komparasi kelebihan dan kekurangan dari bentuk relasional dan object relational database.

Agar didapatkan struktur data yang *reusable* bagi tim developer, sehingga para pengembang aplikasi dapat menggunakan objek-objek yang telah ada untuk digunakan kembali jika proses bisnis yang berjalan mengalami perubahan dan kemudahan bagi pengembang aplikasi dari tahap analisa sistem, perancangan struktur database, dan perancangan struktur dalam program. Dikarenakan tiga tahapan tersebut memiliki struktur yang sama, sehingga waktu pengembangan menjadi lebih singkat. Juga memudahkan *programmer* dalam mengakses data lebih cepat karena tidak perlu memanggil tabel lain. Serta memudahkan penambahan fungsi pada strukturnya dengan hanya menurunkan *classnya*. *Programmer* tidak mengalami kesulitan untuk pemeliharaan data. Karena data yang disajikan berupa objek. Proses pemeliharaan data akan memelihara objek yang perlu saja, tidak perlu objek secara keseluruhan, maka dilakukanlah penelitian *database object relational database* pada aplikasi Notifikasi SMS.

METODE

Metode penelitian yang digunakan adalah metode pengumpulan data, metode analisis, dan metode perancangan database. Teknik yang digunakan dalam metode pengumpulan data ini mencakup antara lain wawancara, mempelajari dokumen, observasi, dan studi kepustakaan (Indrajani, 2011). Wawancara dilakukan pada bagian-bagian yang akan menggunakan sistem database yang diusulkan. Pertanyaan-pertanyaan yang akan ditanyakan akan disiapkan terlebih dahulu sebelum wawancara berlangsung. Sifat yang akan digunakan adalah gabungan antara pertanyaan tertutup dan terbuka. Untuk mendapatkan data dan informasi mengenai kebutuhan pengguna secara lengkap, maka

dilakukan juga pengumpulan dokumen-dokumen. Selain itu juga dilakukan observasi langsung ke lapangan (Indrajani, 2011).

Objek relasional database dalam penerapannya memerlukan beberapa langkah atau proses untuk pengimplemensiannya. Untuk implementasi object relational database model, biasanya harus disertai dengan penerapan pemrograman berbasis objek (Connolly & Begg, 2010). Dalam sebuah projek peranti lunak, pembuatan *Unified Modelling Language* (UML) sangat penting untuk proses pendekatan kepada objek relasional database. Diagram-diagram yang termasuk dalam UML adalah *use case diagram*, *conceptual diagram*, *sequence diagram*, *collaboration diagram*, *state diagram*, *class diagram*, *object diagram*, *component diagram*, dan *deployment diagram*.

Berkenaan dengan konsep objek murni, ada banyak kesamaan antara objek dengan instance dari entitas dan antara kelas dengan entitas. Objek pada kelas biasanya memetakan ke row di tabel dengan sangat erat dalam desain fisik. Namun ada yang perbedaan penting antara pendekatan relasional dan pendekatan objek enkapsulasi dan *inheritance*. Enkapsulasi adalah konsep dasar dalam teori objek, yang berarti bahwa tidak ada aplikasi yang harus memiliki akses langsung ke tabel. Sebagai contoh, daripada memasukkan record langsung ke tabel, aplikasi harus memanggil method untuk proses insert. *Inheritance* mirip dengan *subtyping* relasional. Ada produk database objek murni di pasar yang mendukung objek persisten dan sepenuhnya mendukung enkapsulasi dan pewarisan sifat, tetapi mereka tidak menawarkan model hubungan kinerja sistem relasional dalam database objek.

Database objek relasional yang didefinisikan pada model objek data relasional adalah kumpulan dari bentuk normal pertama (1NF) atau relasi non-1NF dengan atribut atomic maupun non-atomic dari sudut pandang relasional atau koleksi objek-objek dengan atribut yang tidak dienkapsulasi dari sudut pandang objek. Dalam hubungannya dengan tabel, OR database adalah kumpulan tabel yang masing-masing dapat merupakan tabel 1NF atau non-1NF yang tidak berasal dari tipe terstruktur atau typed table yang berasal dari tipe terstruktur (Vo Thi Ngoc Chau & Chittayasothorn, 2008).

Kecenderungan saat ini dalam komunitas database adalah fokus pada DBMS objek relasional (OR), yang merupakan perpaduan antara teknologi berorientasi objek dan database relasional. Teknologi database berorientasi objek berasal dari bidang bahasa pemrograman berorientasi objek di mana kebutuhan akan fasilitas database berkembang. Produk-produk awal pada kategori ini memiliki keterbatasan kemampuan dalam melakukan query dan juga tidak memiliki fasilitas DBMS umum lainnya. Namun, mereka bisa diintegrasikan dengan baik dengan aplikasi-aplikasi yang diimplementasikan dalam bahasa pemrograman berorientasi objek yang sama dengan mereka.

Produk DBMS berorientasi objek (OO) saat ini antara lain Gemstone, Objectivity, ObjectStore, Ontos, O2, Poet dan Versant. Dahulu, komunitas database relasional pada saat yang sama mengembangkan apa yang sekarang disebut OR DBMS (sebelumnya disebut extended relational DBMS) di mana teknologi database relasional diperluas dengan model data yang lebih kaya untuk dapat menangani tipe data baru seperti teks, gambar, audio, dan video. Semua pemain utama di pasar database relasional telah merilis produk dalam area ini, antara lain IBM, Oracle, Informix, dan CA.

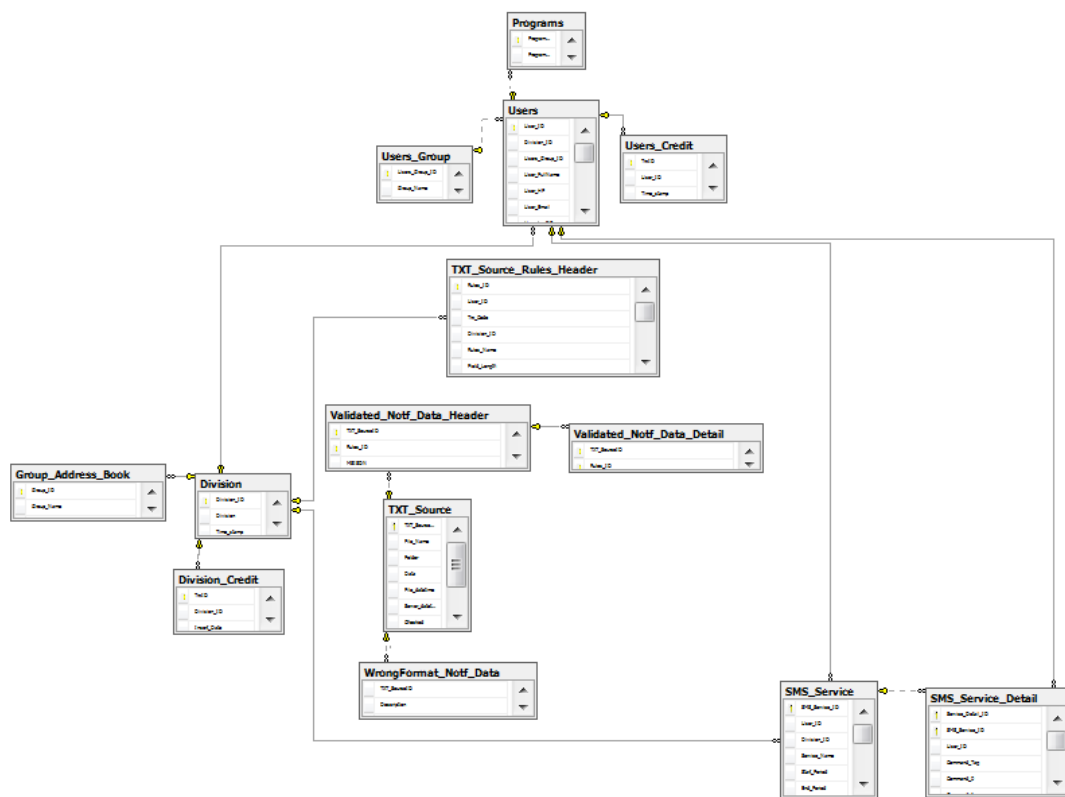
Namun, penting untuk dicatat bahwa perkembangan teknologi database tidak berakhir di sini. Misalnya, untuk aplikasi yang menuntut dan mahal secara komputasi, diperlukan solusi database yang extensible dan lebih ringan, seperti teknologi database main-memory.

Objek membentuk konsep dasar metodologi berorientasi objek dan teknologi objek database. Dalam data model berorientasi objek, objek merepresentasikan entitas secara fisik atau abstrak yang memiliki karakteristik tertentu. Objek yang mirip yang memiliki karakteristik yang sama dapat didefinisikan dan direferensikan dalam sebuah grup, yang berarti memperkenalkan struktur dan mengurangi kompleksitas dalam sebuah konsep domain. Berbeda dengan objek yang

diimplementasikan dengan bahasa pemrograman berorientasi objek, database berorientasi objek menyediakan fasilitas untuk menangani persistent object, misalnya objek-objek yang tetap ada setelah program selesai dijalankan. Objek database juga dapat dipakai bersama oleh beberapa aplikasi yang berbeda.

HASIL DAN PEMBAHASAN

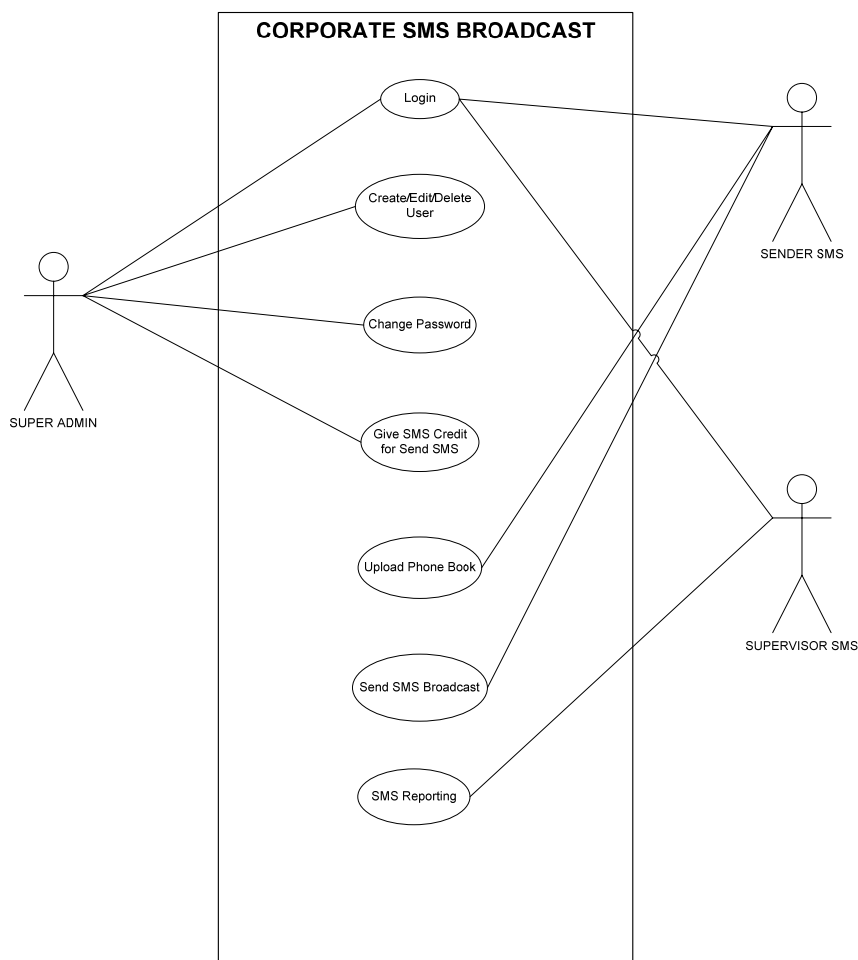
Sistem pengiriman SMS ke HP yang tersedia saat ini Aplikasi Notifikasi SMS mencakup pengiriman pesan dengan *bulk file* diletakkan di server internal kemudian akan dikirimkan oleh aplikasi di server tersebut secara http post ke server eksternal (Gambar 1). *Sender SMS* (alfanumeric, bisa dikustomisasi) hanya melayani pengiriman SMS ke provider: XL, Indosat, Telkomsel, Telkom Flexi.



Gambar 1 ERD pada Notifikasi SMS

Pada ERD diatas terlihat jelas bahwa terdapat relasi untuk setiap tabelnya. Hubungan atau relasi pada tabel dihubungkan oleh *primary key* dari masing-masing tabel. Misalnya: User Group dengan User Credit digeneralisasi dalam Users.

Dalam ERD ini terdapat programs, divisions yang merupakan generalisasi dari Group_Address_Book dan Division_Credit, Txt_Source_Ruler_Header, Txt_Source yang terhubung dengan Validate_Notif_Data_Header dan Validate_Notif_Data_Detail serta WrongFormat_Notif_Data, dan terakhir SMS_Service yang terhubung dengan SMS_Service_Detail. Berikut adalah penerapan UML pada aplikasi Notifikasi SMS (Gambar 2).



Gambar 2 Use case proses bisnis notifikasi SMS

Use case diagram tersebut merupakan penggambaran actor dalam fungsionalitasnya pada aplikasi yang dibuat. Terdapat actor Super Admin, Sender SMS, dan Supervisor SMS. Tiap-tiap actor tersebut memiliki fungsi yang berbeda. Misalnya Supervisor SMS berfungsi untuk mengelola SMS Reporting. Kemudian actor Sender SMS berfungsi untuk upload phone book dan send SMS broadcast. Terakhir Super Admin berfungsi untuk Create, Edit, dan Delete User, Change password, dan Give SMS credit for Send SMS.

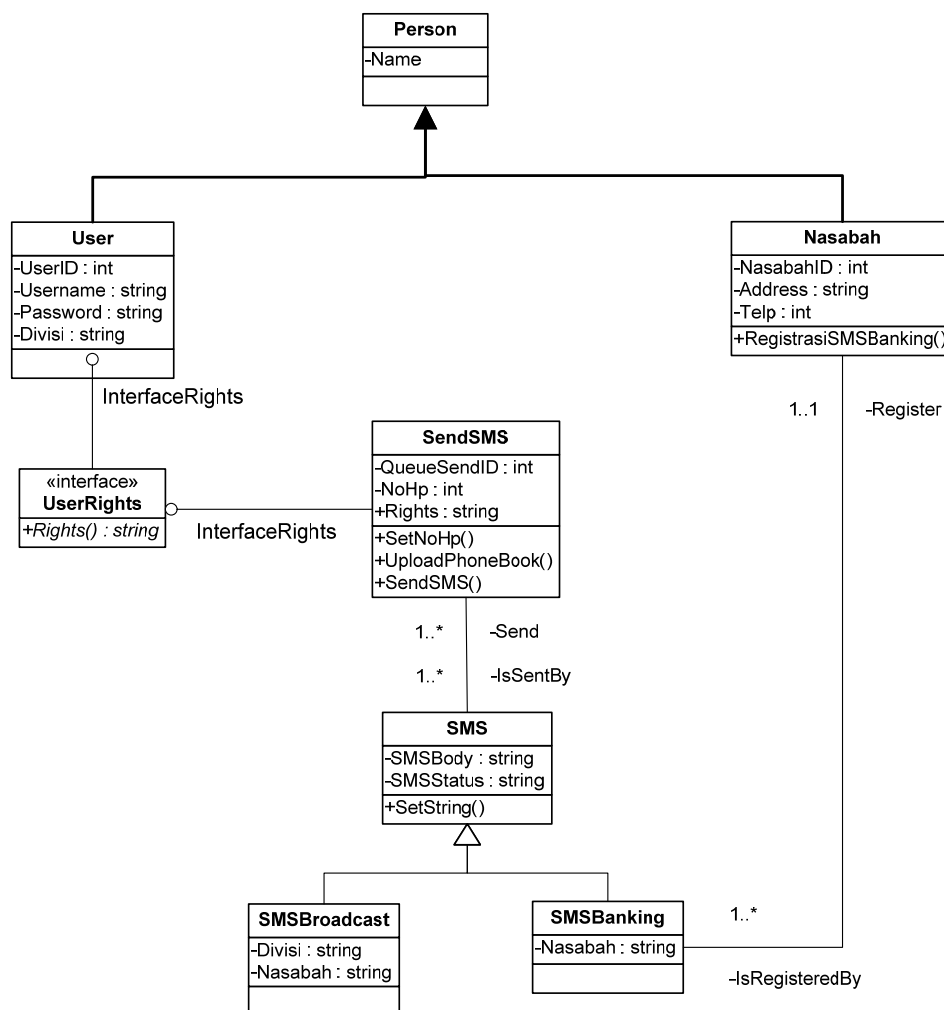
Class diagram (Gambar 3) memaparkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain.

Sequence diagram (Gambar 5) memetakan bagaimana cara object berkomunikasi dengan satu sama lain dengan messages yang terstruktur. Terbagi atas 4 bagian yaitu Sequence Diagram Login, create/edit/delete user, Notifikasi SMS, dan SMS Reporting.

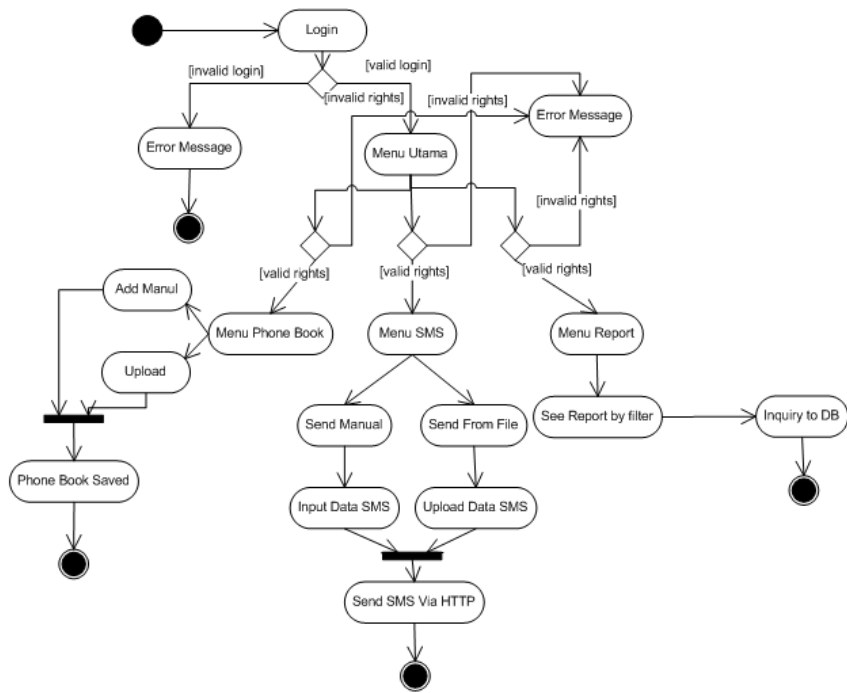
Seluruh user (Super Admin, Sender SMS, dan Supervisor SMS) dapat melakukan login. Setelah user login, dilakukan verifikasi class users. Jika user id dan password valid, menu utama akan ditampilkan. Dari menu utama itulah para user akan melakukan fungsinya masing-masing. Actor Super Admin melakukan login. Setelah itu ke menu utama dan melakukan create, edit atau delete user (Gambar 6). Pada proses notifikasi, actor Sender SMS melakukan login terlebih dahulu. Kemudian masuk ke menu utama. Lalu upload phone book untuk send SMS (Gambar 7).

Setelah UML dibuat, untuk tahap selanjutnya adalah *mapping* dari *class diagram* menjadi *object relational database model*. Pada tahap ini dapat dibuat dengan pendekatan membuat *Functional Data Model* (FDM) terlebih dahulu. FDM menggambarkan entity dan fungsionalitas berdasarkan relationship antar entity. FDM juga lebih memetakan entity dengan konsep OOP di mana dengan FDM dapat terlihat jelas relasi entity. FDM untuk Notifikasi SMS adalah sebagai berikut (Gambar 9).

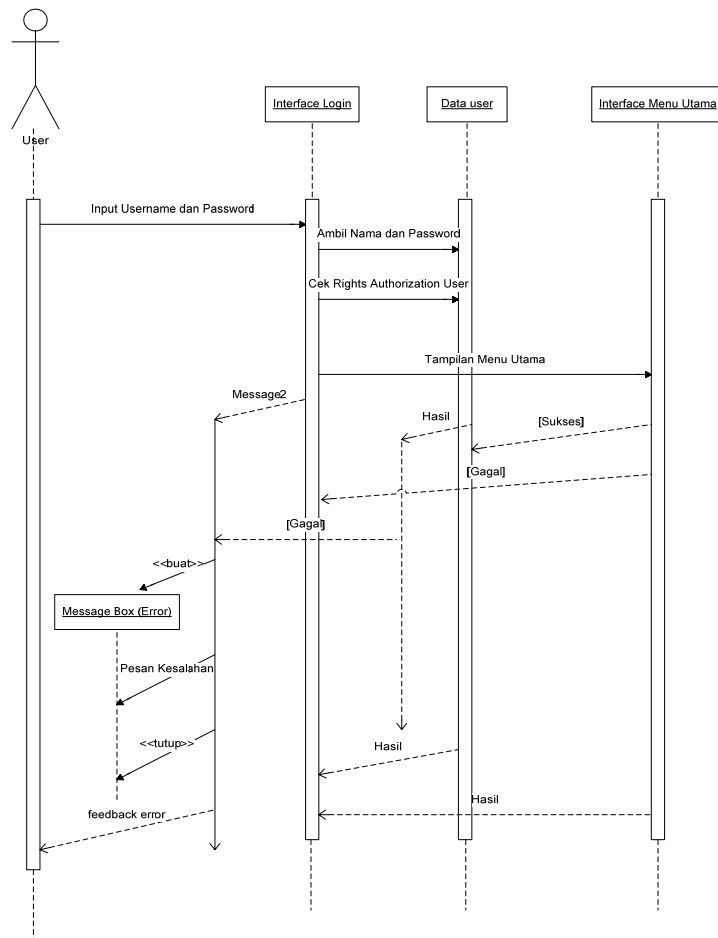
Dengan adanya FDM ini, proses pemetaan ke *object data model* akan semakin jelas. Langkah berikutnya adalah perancangan design objek data model. *Mapping* dari ke dalam objek data model disebut ORM (*object relational mapping*). *Mapping* untuk Notifikasi SMS ke dalam bentuk Object Relational Data Model yaitu *Object Mapping Staff* (Gambar 10) dan *Object Mapping SMS* (Gambar 11).



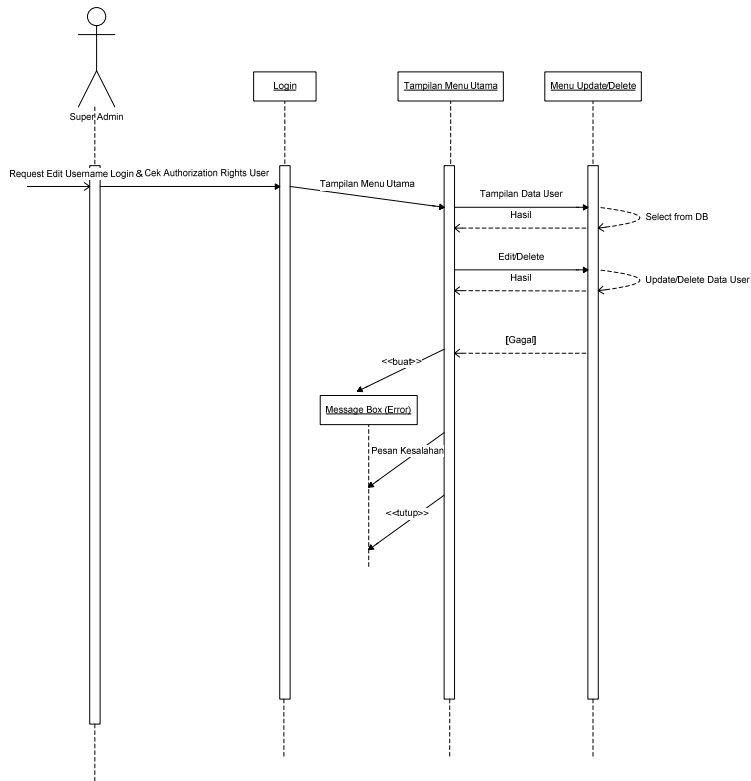
Gambar 3 Class diagram notifikasi SMS



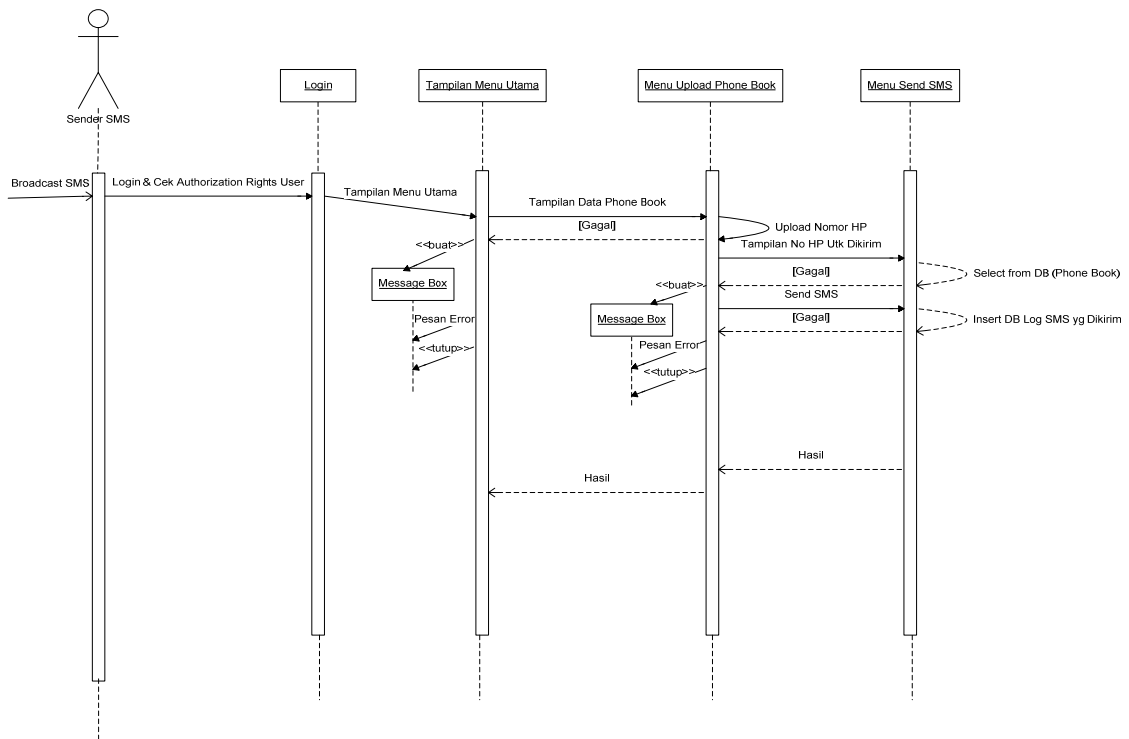
Gambar 4 Activity Diagram Notifikasi SMS



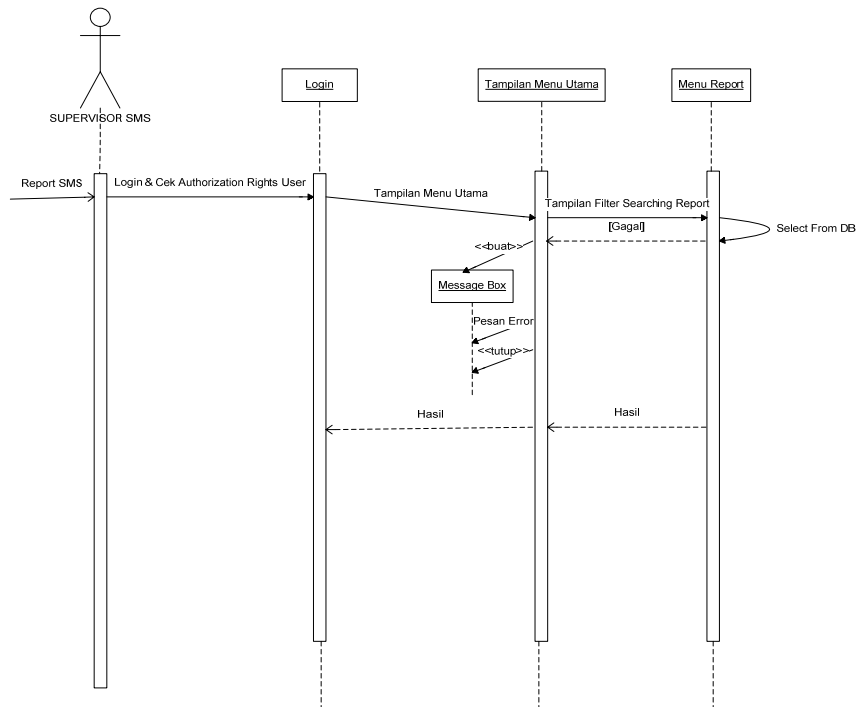
Gambar 5 Sequence diagram login



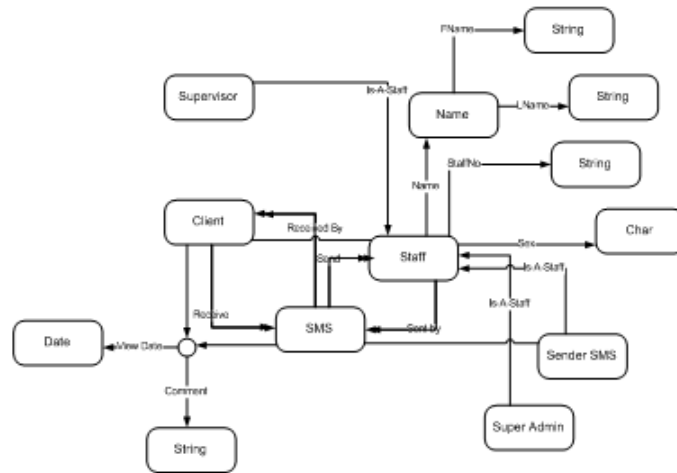
Gambar 6 Sequence diagram create/edit/delete user



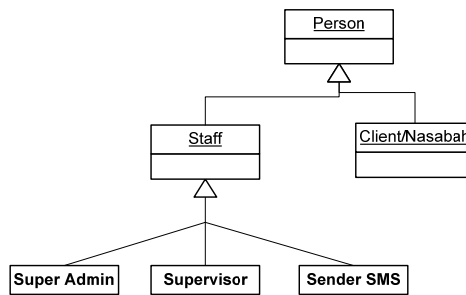
Gambar 7 Sequence diagram notifikasi SMS



Gambar 8 Sequence diagram SMS reporting

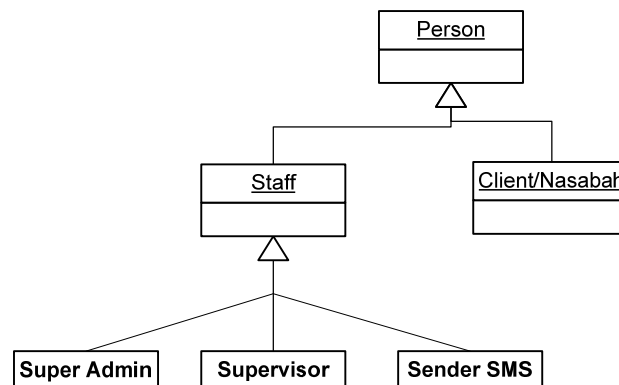


Gambar 9 FDM notifikasi SMS



Gambar 10 Object mapping staff

Object Person memiliki inheritance staff dan client/nasabah, sedangkan staff sendiri memiliki sub class superadmin, supervisor, dan sender SMS.



Gambar 11 Object mapping SMS

Object SMS memiliki *inheritance broadcast* dan *notification* di mana masing-masing *inheritance* tersebut *me-reference class send*. Dua *mapping* di atas menjelaskan bahwa akan dibentuk 2 buah objek yang mewakili keseluruhan system. Sedangkan untuk tablenya hanya diperlukan tabel transaksi yang selalu diinsert setiap kali adanya pengiriman sms.

Setelah membahas proses perancangan arsitektur *object relational database*, terlihat kelebihan dan kekurangan dari model objek relasional data model ini. Kelebihan dan kekurangan model objek relasional. Kelebihannya antara lain lebih sederhana dalam konsep, mendukung kompleks data type seperti images, audio/video dan digital, dan reliablity dan stability sangat bagus karena semua tipe data dapat disimpan dan diproses. Adapun kekurangannya adalah tidak adanya standard yang pasti untuk penggunaan.

PENUTUP

Simpulan yang dapat ditarik dari penelitian ini adalah *database* relasional kurang dapat mendukung tipe data kompleks dan manipulasi data pada relasional hanya dengan menggunakan sql yang fungsinya terbatas. Dalam penggunaannya, database relasional lebih common tetapi kadang normalisasi menyulitkan ketika meretrived data dari database. Object Relational Database mendukung semua tipe data. Penggunaan *object relational database* sangat terpengaruh dari UML yang dibuat saat memetakan dan perancangan object model. *Object Relational Database* memiliki stabilitas, reliability, dan reusability yang lebih tinggi dibandingkan dengan relational database.

DAFTAR PUSTAKA

- Connolly, T., & Begg, C. (2010). *Database System; a Practical Approach to design, Implementation and Management* (5 ed.). England: Pearson Education.
- Indrajani. (2011). *Bedah Kilat 1 Jam: Pengantar dan Sistem Database*. Jakarta: Elex Media Computindo.

Indrajani. (2011). *Perancangan Database Dalam All In 1*. Jakarta: Elex Media Computindo.

Vo Thi Ngoc Chau, & Chittayasothorn, S. (2008). A temporal object relational SQL language with attribute timestamping. *Data & Knowledge Engineering*, 331–361.