

# RANCANG BANGUN PERANGKAT LUNAK DAILY ACTIVITY BERBASIS J2ME

Ghea Ardy P. W<sup>1</sup>, Wiratmoko Yuwono<sup>2</sup>, ST

<sup>1</sup> Mahasiswa, <sup>2</sup> Dosen Pembimbing

Politeknik Elektronika Negeri Surabaya

Institut Teknologi Sepuluh Nopember Kampus ITS Keputih Sukolilo Surabaya 60111, Indonesia

Telp:+62-31-5947280 Fax:+62-31-5946114

Email:livelyc@student.eepis-its.edu

## Abstrak

*Aktivitas sehari – hari yang dilakukan oleh pegawai pada suatu universitas menjadi suatu proses yang rumit ketika pegawai tersebut melakukan kegiatannya di luar universitas. Hal ini dikarenakan pegawai tersebut akan merasa kesulitan dalam melakukan pendataan terhadap aktivitas tersebut yang biasa dilakukan di dalam universitas. Kemudian akan sulit pula bagi atasan untuk mengetahui apakah aktivitas pegawainya telah selesai ataupun belum serta sulitnya atasan untuk memberikan konfirmasi atas selesainya aktivitas pegawai tersebut.*

*Oleh karena itu dibuatlah suatu sistem daily activity yang dapat membantu para pegawai tersebut untuk tetap dapat melakukan aktivitas di luar universitas dan data aktivitas yang dilakukan dapat dikirim melalui ponsel kedalam server, dan atasan juga dapat melakukan konfirmasi terhadap aktivitas yang telah diselesaikan. Sehingga proses pendataan aktivitas menjadi lebih fleksibel dan efisien.*

*Hasil dari sistem ini berupa output dari server pada user yang mengirimkan inputan data berupa format SMS yang telah ditentukan. Software ini dibuat dengan menggunakan database Oracle dan bahasa pemrograman JAVA dan J2ME.*

**Kata Kunci :** *Daily Activity, Aktivitas, Pegawai, JAVA, J2ME, Oracle*

## I. Pendahuluan

### Latar Belakang

Aktivitas sehari – hari merupakan suatu keharusan bagi setiap manusia. Mulai dari anak – sampai orang dewasa sekalipun, pasti memiliki daftar aktivitas mereka masing – masing. Hal ini bertujuan untuk mempermudah mereka untuk mengingat kegiatan yang belum dan sudah dilakukan. Bahkan untuk aktivitas yang masih dalam rencana sekalipun.

Pada universitas, khususnya PENS – ITS terdapat banyak pegawai. Sehingga data kegiatan merekapun berjumlah lebih dari satu pada setiap harinya. Untuk itulah dibuat sebuah system untuk mendata kegiatan para dosen tersebut berupa sebuah buku aktivitas. Data kegiatan tersebut diinputkan manual kedalamnya lalu data tersebut disimpan oleh pihak BAAK untuk kemudian dapat dilihat oleh atasan. Setiap universitas memiliki cara – cara sendiri untuk manage data aktivitas para pegawai dan dosen tersebut agar dapat dilihat kembali apabila diperlukan.

Akan tetapi di zaman yang telah maju ini, perlu dibuat system yang lebih baik lagi. Hal ini dikarenakan banyaknya kekurangan yang ditemukan pada system diatas. Diantaranya ialah proses penginputan datanya masih menggunakan sytem manual pada buku absensi. Kemudian data tersebut dapat hilang ataupun rusak dengan mudah serta kurang efisien. Apalagi bagi para pegawai yang sedang ditugaskan diluar kampus, hal ini tentu akan sangat menyulitkan pihak kampus dalam mendata kegiatan apa saja yang dilakukan oleh pegawai tersebut.

Untuk itulah muncul ide untuk merubah proses pendataan itu melalui sms dengan basis SMS Gateway. Perancangan system yang dilakukan ini dipergunakan untuk menjembatani agar proses data aktivitas pegawai dapat sampai ke tujuan (SMS Center). Selain itu, user

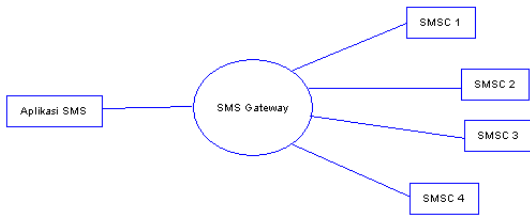
akan semakin dimudahkan dengan tampilan / interface yang dapat diakses melalui ponsel yang berisikan form untuk pengisian data aktivitas. Dalam system yang dibuat, terdapat dua peran yaitu user dan admin. User mengirimkan sms dalam format tertentu yang kemudian dikirimkan ke SMS Center yang kemudian diakses oleh admin. Sedangkan Admin bertugas untuk manage data yang ada.

Dari perancangan system yang dibuat dapat dibuktikan bahwa dengan adanya system tersebut, maka proses pendataan aktivitas para pegawai di PENS – ITS dapat lebih termanage dan lebih efisien

## II. Teori Penunjang.

### a. SMS Gateway

SMS Gateway merupakan pintu gerbang bagi penyebaran Informasi dengan menggunakan SMS. Anda dapat menyebarkan pesan ke ratusan nomor secara otomatis dan cepat yang langsung terhubung dengan database nomor-nomor ponsel saja tanpa harus mengetik ratusan nomor dan pesan di ponsel anda karena semua nomor akan diambil secara otomatis dari database tersebut. Selain itu , dengan adanya SMS Gateway anda dapat mengustomisasi pesan-pesan yang ingin dikirim. Dengan menggunakan program tambahan yang dapat dibuat sendiri, pengirim pesan dapat lebih fleksibel dalam mengirim berita karena biasanya pesan yang ingin dikirim berbeda-beda untuk masing-masing penerimanya.



**Gambar 2.1 SMS Gateway**

SMS Gateway bekerja dengan cara menghubungkan handphone yang memiliki fasilitas SMS dengan computer (PC) selaku operator otomatisnya. Keduanya akan dihubungkan oleh suatu kabel data dari handphone-nya yang sesuai dengan seial port yang ada di PC. Komunikasi antara hanphone dengan PC akan menggunakan bahasa pemrograman serial port.

**b. JAVA**

Java adalah bahasa pemrograman berorientasi objek yang diciptakan untuk mempermudah pengembangan program dengan cara mengikuti model yang telah ada dalam kehidupan nyata. Dalam paradig ini, sesuai dengan model kehidupan nyata, segala bagian (entity) dari suatu permasalahan adalah objek. Objek-objek ini kemudian juga dapat berupa gabungan dari beberapa objek yang lebih kecil.

**a. Objek**

Sebuah objek memiliki dua karakteristik, yaitu state dan behavior. **State** adalah keadaan dari sebuah objek yang biasa disimpan dalam suatu variable. Sedang behavior adalah kelakuan dari objek tersebut, yang diimplementasikan dengan metode. Dengan penjelasan di atas, dapat disimpulkan bahwa objek adalah bagian software yang dibentuk dengan variable-variabel dan metode-metode yang berhubungan dengan variable tersebut.

**b. Kelas**

Kelas adalah semacam cetakan, atau template, untuk membuat objek. Dengan kata lain, kelas dapat kita definisikan sebagai sebuah blueprint, atau prototype, yang mendefinisikan variable dan metode yang sama untuk semua objek sejenis.

**c. Pewarisan**

Secara sederhana, pewarisan(inheritance) berarti sebuah kelas mewarisi state dan behaviour dari kelas lain.

**d. Interface**

Interface digunakan oleh berbagai objek yang tidak terhubung untuk saling berinteraksi. Jadi interface dapat didefinisikan sebagai koleksi definisi metode-metode dan variable-variabel konstan, namun tanpa implementasi.

Implementasi akan dilakukan oleh kelas-kelas yang mengimplements interface ini.

Java adalah bahasa pemrograman berorientasi obyek yang dikembangkan pertama kali oleh Sun Microsystem pada tahun 1995. Source code berakhir .java, setelah dikompilasi menjadi .class. Keunggulan utama bahasa Java adalah multiplatform, yaitu dapat dijalankan pada bermacam-macam system operasi, seperti UNIX, Macintosh dan Windows, karena di Java terdapat Java Virtual Machine(JVM).

- **Koneksi database dengan JDBC**

JDBC atau Java Database Connectivity adalah API yang digunakan Java untuk melakukan koneksi dengan aplikasi lain atau dengan berbagai macam database. JDBC memungkinkan kita untuk membuat aplikasi Java yang melakukan tiga hal, yaitu koneksi ke database, mengirimkan query dan statement ke database, menerima dan mengolah resultset yang diperoleh dari database.

- **Empat komponen JDBC**

1. JDBC API

JDBC API menyediakan metode akses yang sederhana ke sumber data relational (RDBMS) menggunakan pemrograman Java dengan menggunakan JDBC API, kita bisa membuat program yang dapat mengeksekusi SQL, menerima hasil ResultSet, mengubah data dalam database. JDBC API juga mempunyai kemampuan untuk berinteraksi dengan lingkungan terdistribusi dari jenis sumber data yang berbeda-beda.

2. JDBC Driver Manager

Class Driver Manager dari JDBC bertugas untuk mendefinisikan object-object yang dapat digunakan untuk melakukan koneksi ke sebuah sumber data.

3. JDBC Test Suite

JDBC Test Suite membantu kita untuk mencari driver mana yang cocok digunakan untuk melakukan sebuah koneksi ke sumber data tertentu.

4. JDBC-ODBC Bridge

Bridge ini menyediakan fasilitas JDBC untuk melakukan koneksi ke sumber data menggunakan ODBC(Open Database Connectivity) driver.

Dengan keempat komponen yang dimiliki, JDBC menjadi tools yang dapat diandalkan untuk melakukan koneksi, mengambil data dan merubah data dari berbagai database.

- **J2ME**

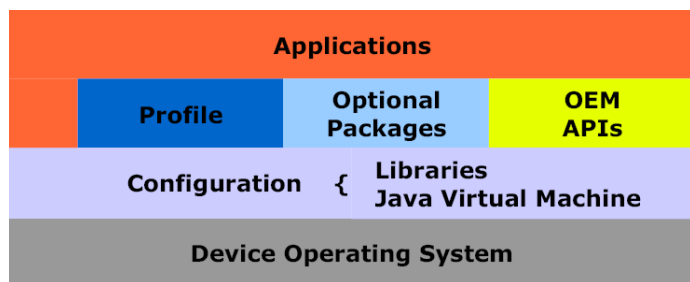
J2ME adalah satu set spesifikasi dan teknologi yang fokus kepada perangkat konsumen.Perangkat ini

memiliki jumlah memori yang terbatas, menghabiskan sedikit daya dari baterai, layar yang kecil dan bandwidth jaringan yang rendah.

Dengan perkembangbiakan perangkat mobile konsumen dari telepon, PDA, kotak permainan ke peralatan-peralatan rumah, Java menyediakan suatu lingkungan yang portable untuk mengembangkan dan menjalankan aplikasi pada perangkat ini.

Program J2ME, seperti semua program JAVA adalah diterjemahkan oleh VM. Program-program tersebut dikompil ke dalam bytecode dan diterjemahkan dengan Java Virtual Machine(JVM).Ini berarti bahwa program-program tersebut tidak berhubungan langsung dengan perangkat

J2ME menyediakan suatu interface yang sesuai dengan perangkat. Aplikasi-aplikasi tersebut tidak harus dikompil ulang supaya mampu dijalankan pada mesin yang berbeda. Inti dari J2ME terletak pada configuration dan profile-profile. Suatu configuration menggambarkan lingkungan runtime dasar dari suatu sistem J2ME. Ia menggambarkan core library, virtual machine, fitur keamanan dan jaringan



Gambar 2.2 Arsitektur J2ME

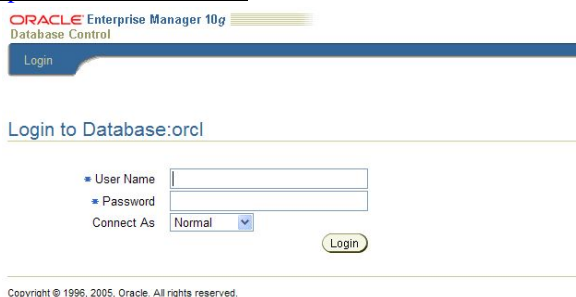
**d. ORACLE**

Oracle adalah basis data relasional yang terdiri dari kumpulan data dalam suatu sistem manajemen basis data RDBMS. Perusahaan perangkat lunak Oracle memasarkan jenis basis data ini untuk bermacam-macam aplikasi yang bisa berjalan pada banyak jenis dan merk perangkat keras komputer (*platform*).

Ada 2 macam tool yang dapat digunakan untuk melakukan administrasi database :

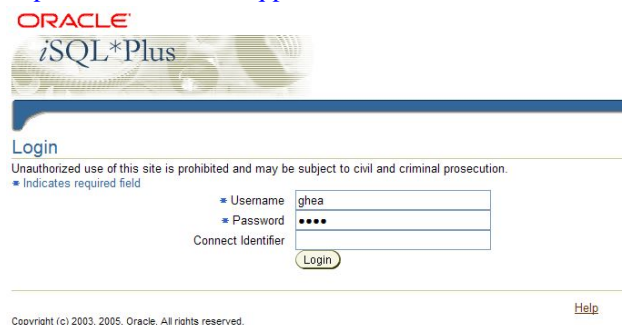
- a. Oracle Enterprise Manager
- b. iSQL \*PLUS

Menjalankan Oracle Enterprise Manager menggunakan Web Browser dengan mengetik alamat: <http://localhost:1158/em>.



Gambar 2.3 Enterprise Manager Oracle

Menjalankan iSQL \*PLUS juga menggunakan Web Browser dengan mengetik alamat <http://localhost:5560/isqlplus>



Gambar 2.4 iSQL \*Plus Oracle

Dasar Administrasi Oracle ada 3, yaitu:

1. Pembuatan TableSpace  
TableSpace adalah ruang penyimpanan untuk pengguna database. Dengan kebebasan untuk mengatur kapasitas dan lokasi penyimpanan database, maka akan dengan mempermudah kita dalam melakukan administrasi database.
2. Pembuatan User  
User adalah sekumpulan orang yang memakai TableSpaces. Pembuatan user hanya dilakukan oleh administrator Database. Pembuatan user bisa dilakukan dengan menggunakan Enterprise Manager atau iSQL Plus.
3. Pembuatan Tabel  
Setiap user memiliki skema yang berisi kumpulan tabel-tabel yang saling berkoneksi.

Structure query language (SQL) adalah satu set pernyataan yang mana semua program dan pengguna mengakses data pada database Oracle.

### III. Perancangan dan Pembuatan Sistem

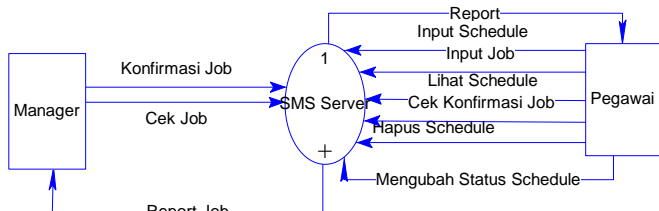
#### Perancangan Sistem

##### Diagram Konteks Sistem

Penggunaan Diagram Arus Data di sini bertujuan untuk memudahkan dalam melihat arus data dalam sistem.

##### a. Diagram Konteks Sistem

Diagram konteks sistem seperti terlihat pada gambar di bawah ini :

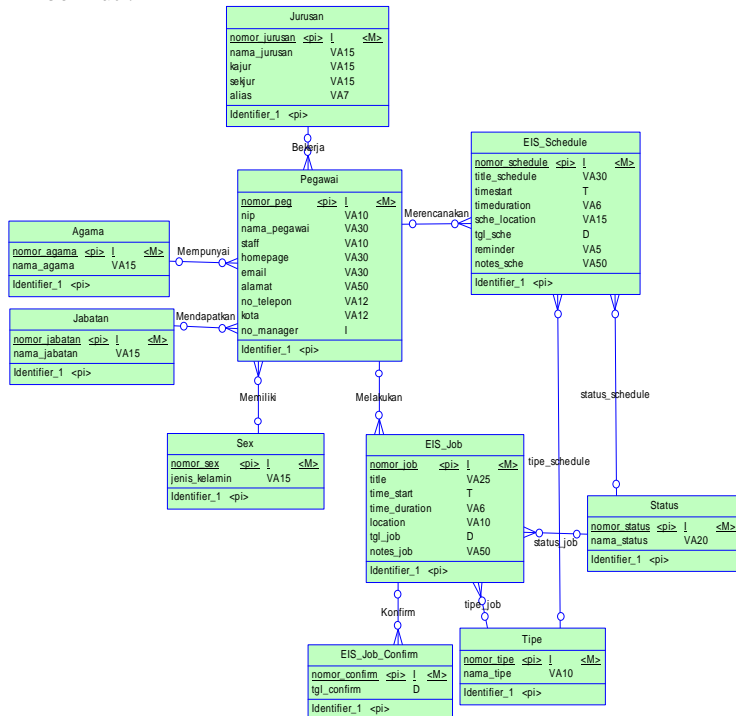


Gambar 3.1 Diagram Konteks Sistem

Ada dua jenis user pada system ini, yaitu pegawai dan manager. Keduanya memiliki prioritas masing – masing dalam melakukan transaksi yang kemudian dikirim ke gateway. Dari gambar diagram diatas, maka dapat diketahui jenis dari transaksi apa saja yang dapat dilakukan oleh user tersebut. Selain itu kedua user tersebut akan menerima balasan dari setiap transaksi atau request yang mereka lakukan tergantung dari jenis transaksi yang dipilih oleh user tersebut.

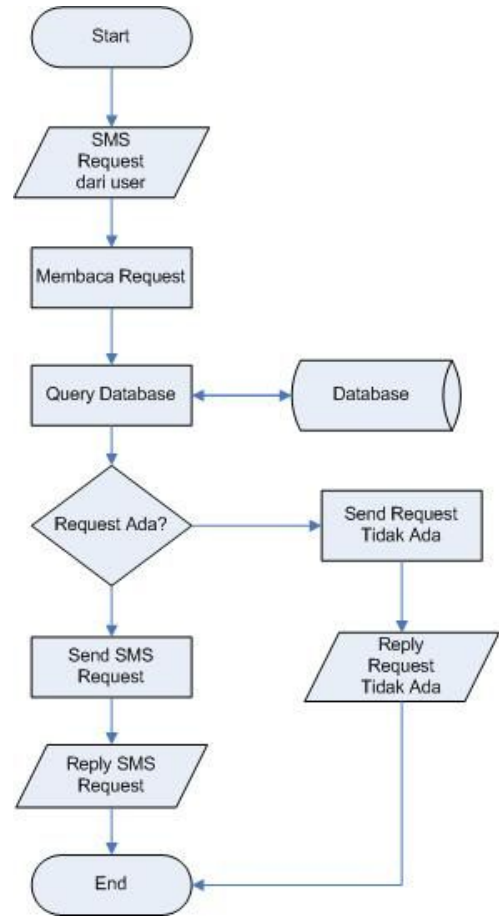
#### Perancangan Basisdata

Diagram relasi entitas ini berguna untuk memberikan gambaran hubungan antara relasi sehingga dapat diimplementasikan pada aplikasi yang akan dibuat. Relasi yang menjadi tabel adalah sebagai berikut :



Gambar 3.2 Relasi antar Tabel

### Flowchart Aplikasi Sistem

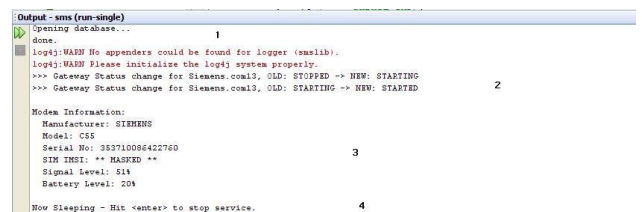


Gambar 3.3 Flowchart Server

### IV. Pengujian dan Analisa

#### Uji Coba Program

Dalam system ini, terdapat dua jenis, yaitu system dari program server dan siste, dari program client. Berikut ini beberapa contoh dan penjelasan dari kedua jenis system tersebut :



Gambar 4.1 Tampilan Server



Gambar 4.2 Server Detect Request

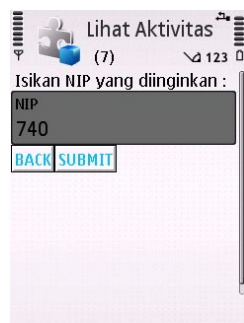
Dari kedua gambar diatas, dapat dijelaskan bahwa Gambar 4.1 adalah tampilan server saat pertama kali dijalankan. Server menunggu adanya

request yang masuk yang dikirimkan oleh user. Dan ketika ada request masuk dari user, maka tampilan server akan menjadi seperti pada Gambar 4.2. Server akan mencetak format sms yang dikirimkan oleh user, lalu akan melakukan proses pengiriman pesan sesuai request yang dikirimkan. Apabila sms balasan telah dikirim, maka sms request tadi akan dihapus agar tidak diproses lagi.

- [5] [http://lecturer.eepis-its.edu/~tessy/PrakDB2/31Dasar\\_Administrasi\\_Oracle.pdf](http://lecturer.eepis-its.edu/~tessy/PrakDB2/31Dasar_Administrasi_Oracle.pdf)
- [6] <http://java.sun.com/products/javacomm/>
- [7] <http://smslib.org/download>



Gambar 4.3 Client



Gambar 4.4 Input Parameter

Gambar 4.3 adalah tampilan dari client. Client dapat memilih beberapa option yang disediakan oleh system. Pada Gambar 4.4, user menginputkan parameter yang dibutuhkan. Tiap option memiliki parameter yang berbeda. Dan pilih tombol submit apabila pengisian parameter selesai.



Gambar 4.5 Kirim



Gambar 4.6 Output

Setelah pengisian parameter selesai, dan tombol submit ditekan, maka akan muncul tampilan seperti pada Gambar 4.5. Disini user harus menginputkan nomor dari server. Kemudian pengiriman pesanpun dilakukan. Apabila request benar, maka akan tampil output seperti pada Gambar 4.6.

## V. Daftar Pustaka

- [1] Gunawan, Ferry. 2003. *Membuat Aplikasi SMS Gateway Server dan Client dengan JAVA dan PHP*. Yogyakarta : Graha Ilmu
- [2] Ben Forta. 2000. *Belajar Sendiri dalam 10 Menit SQL*. Yogyakarta : Andi
- [3] M. Shalahuddin, Rossa A.S. 2008. *Pemrograman J2ME (Belajar Cepat Pemrograman Perangkat Telekomunikasi Mobile)*. Bandung : Informatika
- [4] <http://poss.ipb.ac.id/files/JENI-Intro1-Bab02-Pengenalan%20Bahasa%20JAVA.pdf>