

RANCANG BANGUN SISTEM INFORMASI ABSENSI PERKULIAHAN PJJ

Suryadi,¹ Wiratmoko Yuwono², ST

¹ Mahasiswa, ² 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:acong_surya@yahoo.com

Abstrak

Sering dengan perkembangan Teknologi Informasi yang semakin pesat, kebutuhan akan konsep dan mekanisme registrasi atau daftar ulang berbasis Teknologi Informasi tidak dapat dipungkiri lagi. Salah satu konsep registrasi dalam bidang akademik sangat dibutuhkan untuk memperlancar proses belajar mengajar yang salah satunya adalah daftar ulang.

Proses daftar ulang adalah proses mendaftarkan ulang kembali mahasiswa sebagai mahasiswa aktif dari suatu perguruan tinggi atau sekolah.

Proses pengisian FRS PJJ EEPIS-ITS saat ini masih menggunakan cara manual dan belum bisa diakses oleh mahasiswa melalui internet, sehingga mahasiswa merasa kesulitan dalam mengisi FRS apabila berada diluar kota. Melihat hal tersebut pengisian FRS dapat dikembangkan kedalam sistem terkomputerisasi yang berbasis web, sehingga dapat diakses oleh mahasiswa dimana saja dan kapan saja melalui fasilitas internet.

ABSTRACT

Often with growth of Information Technology which is fast progressively, requirement will conception and registration mechanism or list repeat to base on Information Technology cannot dipungkiri again. One of the registration concept in the field of akademik very required for the fluent of process learn to teach which one of them [is] list repeat.

List process repeat is process register to redo student as active student from an school or college. By conducting payment of lecturing registrsi and admission filling of form plan study to go through lecturing in one next semester

Process admission filling of FRS PJJ EEPIS-ITS in this time still use the way of manual and not yet can be accessed by student [pass/through] internet, so that student feel difficulty in filling FRS if residing in outside town. See the mentioned admission filling of FRS can be developed into computerized system which base on web, so that can be accessed by just student where and any time thought facility of internet.

KEYWORD : FRS, registration

1. Pendahuluan

1.1 Latar Belakang

Sering dengan perkembangan Teknologi Informasi yang semakin pesat, kebutuhan akan konsep dan mekanisme registrasi atau daftar ulang berbasis Teknologi Informasi tidak dapat dipungkiri lagi. Salah satu konsep registrasi dalam bidang akademik sangat dibutuhkan untuk memperlancar proses belajar mengajar yang salah satunya adalah daftar ulang.

Proses daftar ulang adalah proses mendaftarkan ulang kembali mahasiswa sebagai mahasiswa aktif dari suatu perguruan tinggi atau sekolah. Dengan melakukan pembayaran registrsi perkuliahan dan pengisian formulir rencana studi untuk menempuh perkuliahan dalam satu semester berikutnya.

Proses daftar ulang adalah melakukan registrasi pembayaran spp dan mengisi FRS (Form Rencana Studi) yang digunakan untuk mengetahui mata kuliah apa saja yang diambil oleh mahasiswa/i selama satu semester. Pengisian FRS ini dilakukan oleh mahasiswa yang kemudian disetujui oleh dosen wali dan bagian akademik. Apabila terjadi perubahan rencana studi pada semester tersebut mahasiswa dapat melakukan Perubahan Rencana Studi (PRS) atas persetujuan dari dosen wali dan bagian akademik dengan ketentuan setelah satu minggu perkuliahan semester baru berlangsung.

Proses pengisian FRS PJJ EEPIS-ITS saat ini masih menggunakan cara manual dan belum bisa diakses oleh mahasiswa melalui internet, sehingga mahasiswa merasa kesulitan dalam mengisi FRS apabila berada diluar kota. Melihat hal tersebut pengisian FRS dapat dikembangkan kedalam sistem terkomputerisasi yang berbasis web, sehingga dapat diakses oleh mahasiswa dimana saja dan kapan saja melalui fasilitas internet.

1.2 Tujuan

Tujuan dari pembuatan Rancang Bangun Sistem Informasi Daftar Ulang adalah sebagai berikut :

1. Membuat sistem informasi daftar ulang secara online sehingga memberikan kecepatan, keakuratan dan efisiensi waktu dalam proses pengolahan data dan informasi mengenai formulir rencana studi mahasiswa.
2. Memberikan pelayanan yang optimal kepada mahasiswa, sehingga mahasiswa tidak perlu selalu datang ke kampus untuk melakukan daftar ulang.

1.3 Perumusan Masalah

Berdasarkan uraian diatas, maka permasalahan yang timbul dalam pengerjaan proyek akhir ini adalah:

1. Bagaimana melakukan daftar ulang dengan melakukan pengisian FRS untuk mengetahui jumlah mata kuliah yang diambil dalam satu semester.
2. Bagaimana mengetahui tiap-tiap mata pelajaran yang diikuti mahasiswa yang mengambil mata kuliah tersebut.

1.4 Batasan Masalah

Batasan masalah dalam proyek akhir ini diantaranya:

1. Hanya untuk mengetahui jumlah mata kuliah yang diambil dalam satu semester.
2. Mahasiswa mengetahui nilai batas maksimal dan minimal SKS dalam pengisian FRS.
3. Mahasiswa mengetahui mana mata kuliah wajib diambil, belum diambil dan mengulang.
4. Pembuatan aplikasi ini menggunakan Web Design Macromedia Dreamweaver 8, Pemrograman Web JSP dan database ORACLE 9i.

2 Teori Penunjang

2.1 Pengenalan JSP

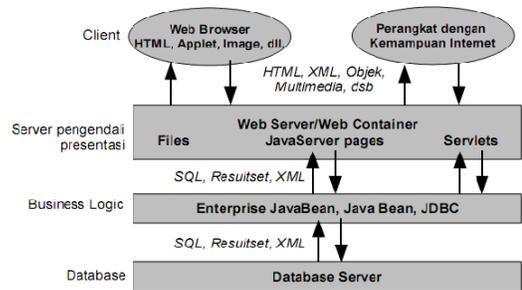
Java Server Pages (JSP) adalah bahasa scripting untuk web programming yang bersifat server side seperti halnya PHP dan ASP. JSP dapat berupa gabungan antara baris HTML dan fungsi-fungsi dari JSP yang disisipkan. JSP merupakan suatu teknologi web berbasis bahasa pemrograman Java dan berjalan di platform java, serta merupakan bagian teknologi J2EE (Java 2 Enterprise Edition). JSP sangat tangguh dan sesuai untuk menangani presentasi di web.

Berbeda dengan Servlet yang harus dikompilasi oleh USER menjadi class sebelum dijalankan, JSP tidak perlu dikompilasi oleh USER tapi SERVER yang akan melakukan tugas tersebut. Makanya pada saat user membuat pertama kali atau melakukan modifikasi halaman dan mengeksekusinya pada web browser akan memakan sedikit waktu sebelum ditampilkan.

JSP dan Servlet bekerja dalam suatu lingkungan web container yang menyediakan koneksi dan service: sedangkan dan Servlet mengontrol presentasi data yang dihasilkan dari suatu proses untuk dikirimkan ke client.

2.1.1 Arsitektur JSP

JSP merupakan bagian dari J2EE dan khususnya merupakan komponen web dari aplikasi J2EE secara keseluruhan. JSP juga memerlukan JVM (Java Virtual Machine) supaya dapat berjalan, yang berarti juga mengisyaratkan keharusan menginstal JVM di server, dimana JSP akan dijalankan. Selain JVM, JSP memerlukan web server yang disebut Web Container.



Gambar 2.1 Arsitektur JSP

Tujuan dari adanya spesifikasi JSP dalam J2EE API adalah untuk menyederhanakan pembuatan halaman web yang bersifat dinamis, dengan cara memisahkan content dan presentasi dari halaman web tersebut. Pada saat JSP dipanggil oleh client melalui web browser, JSP tersebut akan di compile terlebih dahulu menjadi sebuah Servlet, kemudian hasilnya dikembalikan kepada client.

2.1.2 Web Container

Web container adalah service yang dijalankan oleh suatu Java Application Server khususnya untuk service yang kompatibel dengan dengan Servlet dan JSP. Selain menjadi service oleh Java Application Server, Web Container dapat berdiri sendiri. Contoh Web Container adalah Tomcat, ServletExec, Resin, Jrun, Blazix. Web Container juga dapat bekerja sama dengan web server, misalnya Tomcat dengan Apache, JRUN dengan IIS.

2.1.3 Java Virtual Machine (JVM)

Sebelum menginstal Web Container sebagai prasyarat untuk menjalankan JSP, maka kita terlebih dahulu harus menginstal Java Virtual Machine. Java Virtual Machine adalah software yang berfungsi untuk menterjemahkan program Java supaya dapat dimengerti oleh Komputer. Untuk menginstal JVM kita memerlukan JDK (Java Development Kit) karena kita memerlukan class-class API untuk development.

2.2 JDBC

JDBC adalah suatu nama trademark, bukan sebuah singkatan. Walaupun demikian, JDBC sering dikira singkatan dari "Java Database Connectivity". JDBC API terdiri atas sejumlah class dan standart sebagai alat bantu bagi pembuatan program (developer) dan memberikan kemungkinan untuk menulis aplikasi database dengan menggunakan semua JAVA API.

JDBC API memudahkan untuk mengirim statement SQL ke sistem database relasional dan mendukung bermacam macam dialek SQL. JDBC 2.0 lebih dari sekedar SQL, ia memiliki kemampuan untuk berinteraksi dengan sumber data yang lain, seperti file yang berisi data tabular.

Keuntungan menggunakan teknologi JDBC diantaranya adalah sebagai berikut :

- **Mempertahankan data enterprise yang telah ada**
Dengan teknologi JDBC, aplikasi tidak terpaku pada arsitektur tertentu, dapat terus menggunakan database yang telah terpasang dan dapat mengakses informasi dengan mudah.
- **Menyederhanakan development enterprise**
Kombinasi Java API dan JDBC API menjadikan pembuatan aplikasi lebih mudah dan ekonomis. JDBC menyembunyikan kompleksitas tugas-tugas akses data sehingga programmer tidak perlu mengetahui apa yang terjadi di belakang layar. JDBC API mudah untuk dipelajari, dan tidak mahal dalam pemeliharaan.
- **Tidak perlu konfigurasi pada jaringan komputer**
Dengan JDBC API, tidak ada konfigurasi yang diperlukan pada sisi klien. Dengan driver yang ditulis dalam bahasa pemrograman Java, semua informasi yang diperlukan untuk membuat koneksi didefinisikan oleh JDBC URL atau oleh sebuah objek DataSource yang diregistrasi dengan JNDI (Java Naming and Directory Interface).
- **Akses penuh ke metadata**
Dengan JDBC API menyediakan akses metadata sehingga aplikasi yang memerlukan fasilitas dan kapabilitas dari koneksi database tertentu dapat dibangun dengan lebih baik.
- **Koneksi database menggunakan URL**
Teknologi JDBC memakai keunggulan standart internet yaitu URL (*Uniform Resource Locator*) untuk mengidentifikasi koneksi database.

2.3 Oracle

Oracle merupakan salah satu database yang lengkap dan memiliki banyak kelebihan serta keistimewaan dibandingkan database lainnya. Kelebihan dan keistimewaannya antara lain :

- Oracle mendukung ukuran database yang besar hingga jumlahnya terabyte (1000 gb) serta mendukung pengaturan penggunaan space di harddisk, sehingga ruangan harddisk termanfaatkan secara efisien.
- Oracle dapat melayani banyak user yang terkoneksi dalam waktu yang sama dan mengakses data yang sama.
- Oracle dapat mengatur sistem agar pemrosesan database berjalan cepat walaupun jumlah transaksi sangat banyak.
- Oracle dapat menjalankan database selama 24 jam. Pemisahan sistem komputer dan proses backup dapat dilakukan secara on-line tanpa mematikan database.
- Oracle diterima oleh berbagai standart industri untuk pengaksesan data, sistem operasi, user interface, dan protokol jaringan.
- Oracle memiliki fitur untuk membatasi dan memonitor akses data sehingga dapat menghindari pengaksesan dari pihak yang tidak diinginkan.

- Oracle dapat berjalan diberbagai sistem operasi baik linux, windows, unix, dan masih banyak lagi.
- Oracle mampu menduplikasikan database object dalam lokasi server yang berbeda sehingga sangat membantu jika salah satu server lain langsung menggantungkan fungsinya.
- Oracle bisa berjalan di jaringan komputer dan oracle memisahkan proses antara database.

Instalasi Database Oracle 9i

- Minimal System Yang Dibutuhkan

Database oracle terinstall dalam jaringan komputer, Sebelum dapat menginstal database oracle, pastikan bahwa konfigurasi software dan jaringan secara fisik harus sudah terpasang dan berjalan dengan baik pada komputer.

Selanjutnya harus di perhatikan minimal system yang dibutuhkan saat menginstal database oracle. Penulis menggunakan database oracle 9i enterprise edition dimana membutuhkan minimal system sebagai berikut :

1. Oracle Database Enterprise Edition (Server)
 - Pentium 233 or Pentium 266
 - Memory RAM 512 MB
 - Hardisk Minimal
 - FAT File System : 4,5 GB untuk Oracle Home, 140 MB untuk system drive.
 - NTFS File System : 2.75 GB untuk Oracle Home drive, 140 MB untuk system drive.
 - Operating System : Windows NT Workstation 4.0, Windows NT server 4.0, Windows 2000 professional, Windows 2000 server, Windows XP, Windows 2003.
 - Virtual Memory : Minimal 200 MB, Maksimal 400 MB.
 - Temporary Space : 400 MB (alokasi ruang hardisk pada direktori temp)
2. Oracle Client Enterprise Edition.
 - Pentium 233 or Pentium 266
 - Memory RAM 512 MB
 - Hardisk Minimal
 - FAT File System : 1 GB untuk Oracle Home, 51 MB untuk system drive.
 - NTFS File System : 650 MB untuk Oracle Home drive, 51 MB untuk system drive.
 - Operating System : Windows 98, Windows ME, Windows NT Workstation 4.0, Windows NT server 4.0, Windows 2000 professional, Windows 2000 server, Windows XP, Windows 2003.

ICON DATABASE

Setelah melakukan instalasi, beberapa icon database akan muncul. Icon ini berfungsi untuk berhubungan dengan database server. Namun yang terpenting adalah sebagai berikut:

1. Net Manager

Icon ini digunakan untuk konfigurasi atau penghubung antara komputer client

dengan server. Konfigurasi ini harus dilakukan pertama kali bila belum punya konfigurasi ke server. Adapun langkah-langkahnya sebagai berikut:

- a) Klik start, menu, program, orahome9i, configuration and migration tools, dan klik Net Manager.
- b) Bila muncul gambar Oracle Net Manager, klik local dan aktifkan service naming, lalu klik tanda plus. Isikan nama net service name (sesuai yang diinginkan), kemudian klik next.
- c) Pilih protokol koneksi yang digunakan. Dalam hal ini pilih TCP/IP. Kemudian klik next.
- d) Isikan nama komputer server dibagian hostname atau bisa diisi dengan IP nya. Kemudian klik next.
- e) Pada dialog berikut isikan dengan nama global database nya dibagian service name. Kemudian klik next .
- f) Klik test untuk melihat hasil koneksinya. Bila berhasil akan muncul pesan succesfully. Klik tombol close untuk keluar program.

2. Enterprise Manager Console.

Di dalam aplikasi ini terdapat empat buah komponen penting. Diantaranya adalah:

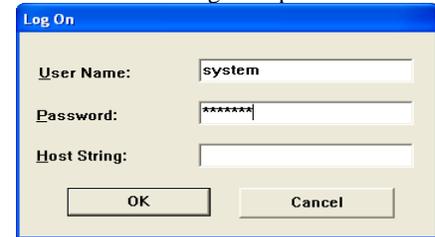
- 1) Security, dipakai untuk membuat user baru.
- 2) Storage, dipakai untuk membuat tablespace atau ruang penyimpanan untuk penggunaan database.
- 3) Schema, dipakai untuk membuat table atau berisi objek-objek yang dimiliki oleh suatu user. Untuk menggunakan fungsi ini, anda harus login ulang dengan user dan password yang baru anda buat. Agar semua perubahan yang terjadi dapat disimpan dilokasi pemakai atau user.
- 4) Instance, merupakan service yang mewakili kondisi kerja dari server database.

3. SQL Plus

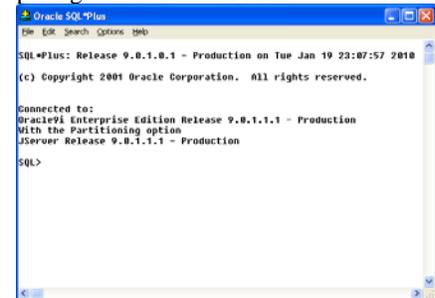
SQL Plus merupakan suatu aplikasi yang dipakai untuk melakukan administrasi database. Dengan aplikasi ini kita bisa membuat user beserta hak atau privillagenya, tablespaces, table dan lain-lain. Namun, dalam mengoperasikan aplikasi ini, kita harus menguasai query sql atau bahasa sql. Karena aplikasi ini dioperasikan dengan manual.

Adapun langkah untuk menjalankan SQL Plus adalah Start, All Program, Oracle-OraHome90, Application

Development, SQL Plus. Maka akan muncul halaman log on seperti berikut:



Masukkan username dan passwordnya untuk masuk ke halaman SQL. Misal menggunakan username “system” dan password “manager” lalu tekan “ok”, maka akan masuk ke lembar SQL, seperti pada gambar dibawah ini:



3.1 SQL (Structured query language)

Sql merupakan bahasa yang digunakan untuk mengakses dan memanipulasi database. Sql ini sudah menjadi bahasa standard, sehingga hampir semua sistem database dapat memahaminya. Bahkan sekarang sql diterapkan pada internet dan intranet agar konsep pembuatan web dinamis menjadi terpenuhi.

3.2 KELOMPOK PERNYATAAN SQL

Pernyataan sql dikelompokkan menjadi 5, yaitu :

1. DDL (Data Definition Language)

Berfungsi untuk mendefinisikan atribut-atribut database, table, dan lain-lain. Yang termasuk DDL antara lain:

 - Create, untuk membuat table atau index
 - Alter, untuk merubah struktur table
 - Drop, untuk menghapus table atau index
2. DML (Data Manipulation Language)

Berfungsi untuk memanipulasi data. Yang termasuk DML adalah:

 - Select, untuk memilih data
 - Insert, untuk menambah data
 - Delete, untuk menghapus data
 - Update, untuk mengubah data
3. DCL (Data Control Language)

Berfungsi untuk mengendalikan pengaksesan data. Yang termasuk DCL adalah :

- Grant, untuk memberikan kendali pengaksesan data.
 - Revoke, untuk mencabut kemampuan pengaksesan data
 - Lock Table, untuk mengunci table
4. Pengendali Transaksi
Berfungsi untuk mengendalikan pengeksesian transaksi. Yang termasuk ini adalah :
- Commit, untuk menyetujui rangkaian perintah yang telah dijalankan.
 - Rollback, untuk membatalkan transaksi
5. Pengendali Programatik
Mencakup pernyataan-pernyataan yang berhubungan dengan pemanfaatan SQL dalam bahasa lain (SQL yang dilekatkan).
Yang termasuk dalam kelompok ini adalah :
- OPEN untuk membuka kursor
 - CLOSE untuk menutup kursor
 - DECLARE untuk mendeklarasikan kursor
 - FETCH untuk mengambil nilai baris berikutnya.

2.3.1 Membuat dan Menghapus Database

Hal pertama yang harus dimengerti dalam mengatur database adalah bagaimana membuat database. Database inilah yang digunakan untuk menampung data. Perintah yang digunakan adalah :

```
> CREATE DATABASE nama_database;
```

Perlu diperhatikan bahwa penulisan perintah adalah 'case sensitif' artinya dalam menulis perintah harus persis dengan menggunakan huruf besar. Jika tidak persis maka perintah tersebut akan dikenali sebagai perintah lain atau bahkan tidak dikenali. Tanda ';' digunakan untuk mengakhiri baris perintah.

Untuk menghapus database cukup digunakan perintah sebagai berikut :

```
> DROP DATABASE nama_database;
```

2.3.2. Menggunakan Database

Untuk mengarahkan database mana yang ditempati untuk bekerja, digunakan perintah :

```
> USE nama_database;  
Database changed
```

Perintah tersebut harus selalu dikerjakan untuk memberitahukan bahwa database tersebut akan dimanipulasi data yang ada didalamnya.

2.3.3. Membuat Tabel

Setelah berhasil membuat database dan menggunakannya, langkah selanjutnya adalah mengisi database tersebut dengan tabel – tabel. Karena tabel – tabel tersebut baru dan belum terisi maka digunakan perintah USE nama_tabel; untuk mengarahkan di database mana tabel akan dibuat.

Untuk membuat tabel, maka perlu dirancang terlebih dahulu, yang meliputi jumlah field, tipe field serta atribut pelengkap yang lainnya.

2.3.4. Mengisi Tabel

Sekarang tabel yang telah dibuat dapat diisikan data. Cara untuk mengisi dapat dibedakan dalam dua cara, pertama dengan memasukkan data satu – persatu untuk masing – masing baris, cara kedua adalah dengan mengubah file .txt yang berisi data ke dalam tabel yang diinginkan. File .txt yang akan dimasukkan sebaiknya ditempatkan pada direktori yang sama dengan program yang dijalankan agar mempermudah pemindahan ke tabel. Perintah yang digunakan untuk memasukkan file adalah sebagai berikut :

```
> LOAD DATA LOCAL INFILE  
"nama_file.txt" INTO TABLE nama_tabel;
```

Sedangkan untuk menghapus data yang terdapat dalam tabel tersebut adalah dengan perintah :

```
> DELETE FROM nama_tabel WHERE  
(kondisi);
```

2.3.5. Membaca dan Mengatur Isi Tabel

Untuk membaca semua data dari sebuah tabel digunakan perintah SELECT * FROM nama_tabel;. Selain dapat dibaca secara keseluruhan, juga dapat dilakukan pembacaan data tertentu pada tabel. Untuk menampilkan data di dalam tabel yang sesuai yang diinginkan di sebuah kolom, digunakan perintah sebagai berikut :

```
> SELECT * FROM nama_tabel WHERE  
(kondisi);
```

Perintah tersebut berarti program diperintahkan untuk menampilkan semua data dari tabel yang diinginkan yang memenuhi kondisi yang dimasukkan.

Selain menampilkan data dari sebuah tabel berdasarkan karakter tertentu yang terdapat di dalam sebuah kolom data tersebut, untuk tipe data DATE, dapat juga dipilih data berdasarkan data tanggal yang ada. Sebagai contoh, untuk menampilkan data yang bertanggal sebelum tanggal 17 Juli 2003 atau kalau dituliskan dalam format DATE adalah

“2003-07-17”, maka digunakan perintah berikut :

```
> SELECT * FROM nama_tabel WHERE
tanggal <="2003-03-17";
```

Di samping menampilkan semua data, MySQL juga memberikan kemudahan untuk menampilkan data berdasarkan kolom yang diinginkan.

2.4 INTEGRASI ANTARA JSP DAN ORACLE

Beragam-macam aplikasi web yang dapat dibuat dengan menggunakan JSP dan Oracle.

Untuk mengintegrasikan JSP dengan database ORACLE, pertanyaan pertama yang timbul adalah bagaimana mengakses database ORACLE menggunakan web atau JSP. Sebuah file yang berekstensi .jsp dapat dieksekusi langsung lewat sebuah browser dengan menyembunyikan kode-kode pemrograman JSP, dan menampilkan kode – kode html yang dimengerti oleh browser. JSP inilah yang digunakan sebagai antar muka ke web atau user sekaligus penghubung dengan database. Namun, ini terjadi jika file .jsp diakses secara langsung dari browser. Jika file tersebut dieksekusi setelah aksi dari sebuah form html dijalankan, JSP disini sering berfungsi sebagai penghubung ke database dan baru menjadi antar muka ke user setelah aksi tadi diproses dan ditampilkan.

Untuk menampilkan data dari database yang pertama harus terdapat databasenya terlebih dahulu. Dan kemudian untuk mengakses database tersebut dari program JSP digunakan listing sebagai berikut :

```
<% @ page import="java.sql.*" %>
<%
Connection Conn=null;
String dbname="orcl";
String host="localhost";
String port="1521";
ResultSet rs = null;
Statement statement = null;
String oracleURL
=jdbc:oracle:thin:@'+host+'+'+port+'+'+dbname;
```

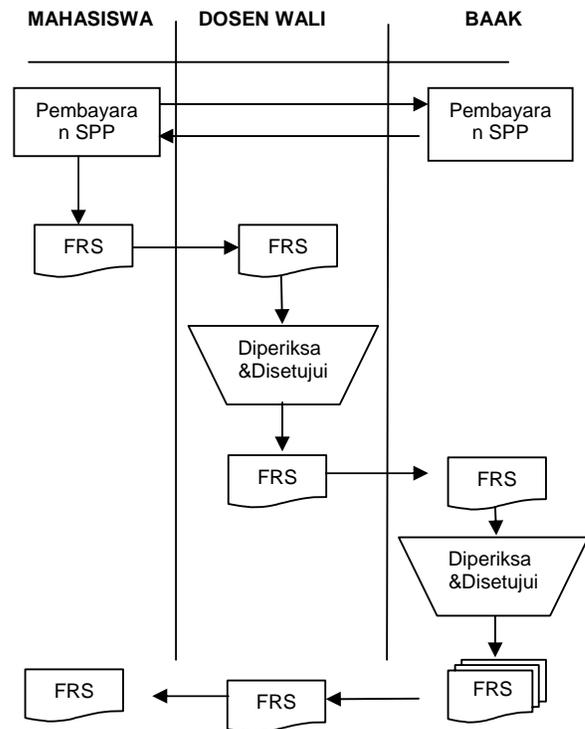
```
Class.forName("oracle.jdbc.driver.OracleDriver");
```

```
Conn=DriverManager.getConnection(oracleURL,"penilaian","penilaian");
statement =Conn.createStatement();
//Conn.close();
```

%>

3. Perancangan dan Pembuatan Sistem

3.1.1 Blok Diagram Sistem

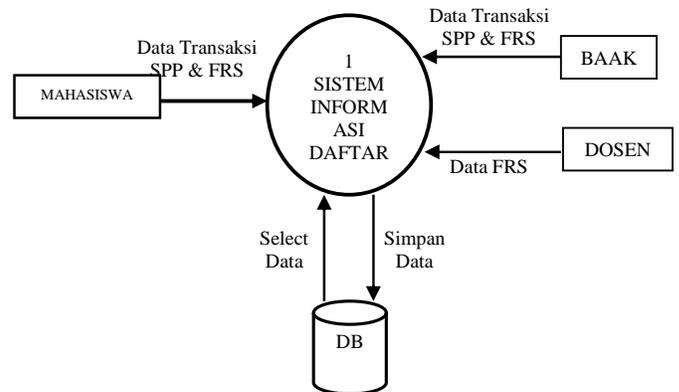


Gambar 3.1 Blok Diagram Sistem

- Mahasiswa melakukan pembayaran SPP.
- Bukti pembayaran diserahkan kepada BAAK.
- Mahasiswa melakukan registrasi FRS online.
- Hasil FRS akan diperiksa dan disetujui oleh dosen wali.
- Hasil FRS akan diperiksa dan disetujui oleh BAAK.

3.1.2 Data Flow Diagram (DFD)

- Berikut Gambar DFD level 0:



Gambar 3.2 Blok Diagram Sistem

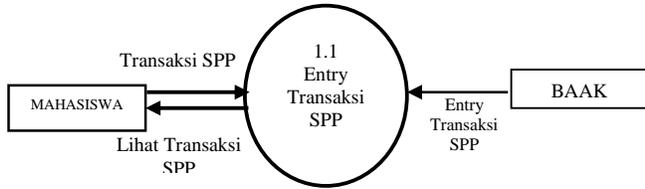
Pada DFD level 0 Mahasiswa melakukan login dan mengisi data FRS. Dosen melakukan login dan aktifasi data FRS yang telah diisi oleh mahasiswa. Sedangkan pegawai melakukan login dan mengisi data transaksi SPP mahasiswa.

14	Staff
15	Transaksi SPP

Tabel 3.5 daftar table yang digunakan

Table-table tersebut berelasi seperti gambar di bawah ini :

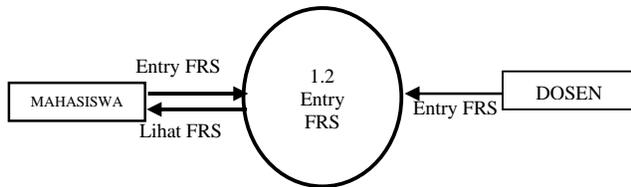
b. Berikut Gambar DFD level 1.1:



Gambar 3.3 DFD level 1.1

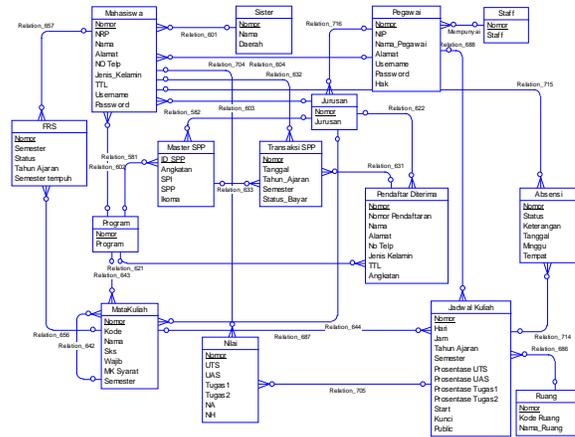
Pada DFD level ini Mahasiswa melakukan transaksi SPP yang kemudian data transaksi SPP akan di entry oleh pegawai administrasi. Selanjutnya Mahasiswa melakukan proses pengisian FRS.

c. Berikut Gambar DFD level 1.2:

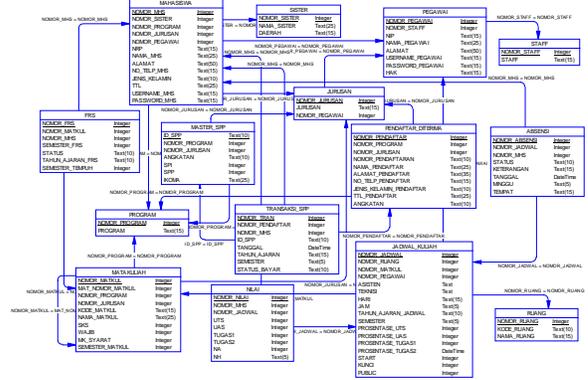


Gambar 3.4 DFD level 1.2

Pada DFD Level ini Mahasiswa melakukan login dan melakukan entry FRS. Selanjutnya dosen akan memeriksa untuk menyetujuinya atau tidak hasil pengisian FRS dari mahasiswa tersebut



Gambar 3.6 Diagram relasi antar entitas (Conceptual ERD)



Gambar 3.7 Diagram relasi antar table (Physical ERD)

3.2 Perancangan dan Pembuatan Sistem

3.2.1 Perancangan Basisdata

Dalam tahap ini dilakukan perancangan dan pembuatan database pada database ORACLE dengan table-table sebagai berikut :

Nomor	Nama Tabel
1	Absen Mahasiswa
2	FRS
3	Jadwal_Kuliah
4	Jurusan
5	Mahasiswa
6	Master_SPP
7	Matakuliah
8	Nilai
9	Pegawai
10	Pendaftar_diterima
11	Program
12	Ruang
13	Sister

Daftar table yang digunakan :

Nama Field	Type Data	Lebar	Keterangan
Nomor*	Varchar	10	No Pegawai
NIP	Varchar	20	ID Pegawai
Nama	Varchar	35	Nama Pegawai
Alamat	Varchar	50	Alamat Pgw
Staff	Varchar	5	Kode Staff
Username	Varchar	20	Username pgw
Password	Varchar	20	Pass Pegawai
Hak	Varchar	10	Hak Pegawai

Table 3.8 Pegawai

Nama Field	Type Data	Lebar	Keterangan
Nomor*	Varchar	5	No Staff
Staff	Varchar	15	Ket Staff

Table 3.9 Staf

Nama Field	Type Data	Lebar	Keterangan
Nomor*	Varchar	10	No Mhs
NRP	Varchar	10	ID Mhs
Nama	Varchar	20	Nama Mhs
Alamat	Varchar	50	Nama Mhs
No Telp	Varchar	15	No Telp Mhs
Jenis_Kelamin	Varchar	10	JK Mhs
TTL	Varchar	25	TTL Mhs
Program	Varchar	5	Program Mhs
Jurusan	Varchar	5	Jurusan Mhs
Dosen_Wali	Varchar	5	Wali Mhs
Username	Varchar	20	Usernme mhs
Password	Varchar	20	Passwd Mhs
Sister	Varchar	5	Sister Mhs
Angkatan	Varchar	10	Angktan Mhs

Table 3.10 Mahasiswa

Nama Field	Type Data	Lebar	Keterangan
Nomor*	Varchar	5	No Matkul
Kode	Varchar	10	Kode Matkul
Nama	Varchar	50	Nama Matkul
SKS	Varchar	5	Jumlah Sks
Wajib	Varchar	10	Ket Matkul
MK_Syarat	Varchar	10	Matkul Syarat
Semester	Varchar	5	Ket Semester
Program	Varchar	5	Program pd Matakuliah
Jurusan	Varchar	5	Jur pd Matkul

Tabel 3.11 Matakuliah

Nama Field	Type Data	Lebar	Keterangan
Nomor*	Varchar	10	No Jadwal
Matakuliah	Varchar	10	Ket Matkul pd Jadwal
Dosen	Varchar	5	Dosen yg mengajar matkul
Asisten	Varchar	5	Asisten yg membantu
Teknisi	Varchar	5	Teknisi yg membantu
Hari	Varchar	10	Hari perkuliahan berlangsung
Ruang	Varchar	10	Ruang yg digunakan
Jam	Varchar	15	Jam perkuliahan berlangsung
Kelas	Varchar	5	Ket Kelas
Tahun_Ajaran	Varchar	10	Thn_Ajaran matakuliah
Semester	Varchar	5	Smt matkul
Prosentase_Tugas1	Number	5	Persen tgs1

Prosentase_Tugas2	Number	5	Persen tgs2
Prosentase_UTS	Number	5	Persen uts
Prosentase_UAS	Number	5	Persen uas
Kunci	Number	5	Kunci nilai
Publik	Number	5	Publik nilai

Tabel 3.12 Jadwal_kuliah

Nama Field	Type Data	Lebar	Keterangan
Nomor*	Varchar	5	No Ruang
Kode_Ruang	Varchar	10	Kode Ruang
Nama_Ruang	Varchar	10	Nama Ruang

Table 3.13 Ruang

Nama Field	Type Data	Lebar	Keterangan
Nomor*	Varchar	5	No Program
Program	Varchar	10	Nama Progrm

Tabel 3.14 Program

Nama Field	Type Data	Lebar	Keterangan
Nomor*	Varchar	5	No Jurusan
Jurusan	Varchar	30	Nama Jurusan
Kajur	Varchar	5	Kajur

Tabel 3.15 Jurusan

Nama Field	Type Data	Lebar	Keterangan
Nomor	Varchar	5	No Frs
Mahasiswa	Varchar	5	Mhs yg mengisi FRS
Matakuliah	Varchar	5	Matkul_yg diambil
Semester	Number	5	Smt FRS
Status	Varchar	10	Status matkul yg diambil
Tahun_Ajaran	Varchar	10	Tahun_ajaran
Semeter_tempuh	Number	5	Semester yg ditempuh

Table 3.16 FRS

Nama Field	Type Data	Lebar	Keterangan
Nomor	Varchar	5	No Nilai
Jadwal_Kuliah	Varchar	10	Jadwal yang diberikan nilai
Mahasiswa	Varchar	5	Mhs yg diberi nilai
Tugas1	Number	5	Nilai Tugas1
Tugas2	Number	5	Nilai Tugas2
UTS	Number	5	Nilai UTS
UAS	Number	5	Nilai UAS
NA	Number	5	Nilai Angka
NH	Varchar	5	Nilai Huruf

Table 3.17 Nilai

Nama Field	Type Data	Lebar	Keterangan
Id_spp*	Varchar	5	No SPP
Program	Varchar	5	No Program
Jurusan	Varchar	5	No Jurusan
Angkatan	Varchar	10	Th angkatan
SPI	Number	10	Besar SPI
SPP	Number	10	Besar SPP
Ikoma	Number	10	Besar Ikoma

Table 3.18 Master_SPP

Nama Field	Type Data	Lebar	Keterangan
Nomor*	Varchar	5	No Transaksi
Mahasiswa	Varchar	5	No mahasiswa
ID_SPP	Varchar	5	ID_SPP
Tanggal	Date		Tanggal transaksi
Tahun_ajaran	Varchar	15	Tahun ajaran yang berlangsung
Semester	Varchar	5	Semester yang ada
Status_bayar	Varchar	5	Status bayar

Tabel 3.19 Transaksi SPP

Nama Field	Type Data	Lebar	Keterangan
Nomor*	Varchar	5	No Sister
Nama	Varchar	15	Nama sister
Daerah	Varchar	20	Daerah

Tabel 3.20 Sister

Nama Field	Type Data	Lebar	Keterangan
Nomor	Varchar	5	No sister
Jadwal_kuliah	Varchar	5	No jadwal kuliah
Mahasiswa	Varchar	5	No mahasiswa
Status	Varchar	5	Hadir / Ijin / Sakit / Alasan
Keterangan	Varchar	10	Keterangan
Tanggal	Date		Tanggal saat Absen
Minggu	Number	5	Minggu
Tempat	Number	5	1 : PENS 2 : Sister

Tabel 3.20 Absen_mahasiswa

Keterangan :
* primary key

4. Pengujian dan Analisa

4.1 Pengujian Sistem

Pada bab pengujian dan analisa ini akan dibahas mengenai pengujian dari perangkat lunak (software) yang dibuat. Hal ini bertujuan untuk mengetahui sejauh mana ketepatan eksekusi perangkat lunak yang telah dibuat serta tidak menutup kemungkinan mengetahui kelemahannya. Sehingga dari sini nantinya dapat disimpulkan apakah perangkat lunak yang dibuat dapat berjalan secara benar dan sesuai dengan kriteria yang diharapkan.

Untuk pengujian terhadap aplikasi yang telah direncanakan akan dilakukan dimulai langkah-langkah berikut :

- Kategori Login
- Kategori Administrasi
- Kategori Mahasiswa
- Kategori Dosen

4.1.1 Kategori Login

Untuk masuk ke dalam system harus login terlebih dahulu. Setiap pengguna memiliki otoritas dan hak user yang berbeda-beda antara administrator, mahasiswa dan dosen.



Gambar 4.1 Halaman Login



Gambar 4.2 Halaman Login Administrasi



Gambar 4.3 Halaman Login Mahasiswa



Gambar 4.4 Halaman Login Dosen

4.1.2 Kategori Administrasi

Setelah melakukan login sebagai admin maka muncul menu pilihan menu *Data Program*, *Data Jurusan*, *Data Staff*, *Data Pegawai*, *Data Mahasiswa*, *Data Mata Kuliah*, *Data Master SPP*, *Data Transaksi SPP*, *Data Jadwal Kuliah*, *Data Ruang* dan *Data Sister*.

Untuk lebih jelasnya dapat dilihat pada gambar dibawah ini :



Gambar 4.5 Halaman Menu Administrasi

Fungsi dari masing-masing menu itu adalah :

- **Data Program**
Admin dapat melihat seluruh data program keahlian yang ada. Selain itu juga dapat mengentry, mengubah dan menghapus data program.



Gambar 4.6 Halaman Menu Administrasi Program

- **Data Jurusan**
Admin dapat melihat seluruh data jurusan yang ada. Selain itu juga dapat mengentry, mengubah dan menghapus data jurusan.



Gambar 4.7 Halaman Menu Administrasi Jurusan

- **Data staff**
Admin dapat melihat seluruh data jurusan yang ada. Selain itu juga dapat mengentry, mengubah dan menghapus data staff.



Gambar 4.8 Halaman Menu Administrasi Staff

- Data Pegawai
Admin dapat melihat seluruh data pegawai yang ada. Selain itu juga dapat mengentry, mengubah dan menghapus data pegawai.



Gambar 4.9 Halaman Menu Administrasi Pegawai

- Data Mahasiswa
Admin dapat melihat seluruh data mahasiswa yang ada. Selain itu juga dapat mengentry, mengubah dan menghapus data mahasiswa.



Gambar 4.10 Halaman Menu Administrasi

Mahasiswa

- Data Mata Kuliah
Admin dapat melihat seluruh data mata kuliah yang ada. Selain itu juga dapat mengentry, mengubah dan menghapus data mata kuliah.



Gambar 4.11 Halaman Menu Administrasi Mata

Kuliah

- Data Master SPP
Admin dapat melihat seluruh data master SPP yang ada. Selain itu juga dapat mengentry, mengubah dan menghapus data master SPP.



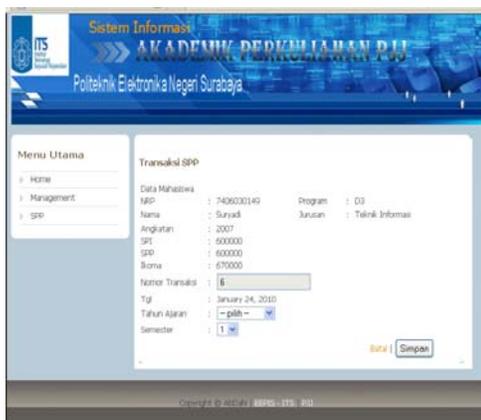
Gambar 4.12 Halaman Menu Administrasi

Master SPP

- Data Transaksi SPP
Admin mengentry transaksi SPP yang telah dilakukan oleh mahasiswa. Selain itu juga dapat mengubah dan menghapusnya.
Admin memasukkan nrp mahasiswa terlebih untuk memastikan mahasiswa ada atau tidak dalam database. Setelah itu jika mahasiswa ada maka akan muncul data mahasiswa dan informasi jumlah pembayaran SPP yang harus dibayarkan, masukkan tanggal dan semester kemudian simpan.



Gambar 4.13. Halaman Menu Administrasi Transaksi SPP



Gambar 4.14.. Halaman Menu Administrasi Entry Transaksi SPP

- **Data Ruang**
Admin dapat melihat seluruh data ruang yang ada. Selain itu juga dapat mengentry, mengubah dan menghapus data ruang.



Gambar 4.15. Halaman Menu Administrasi Ruang

- **Data Suster**
Admin dapat melihat seluruh data suster yang ada. Selain itu juga dapat mengentry, mengubah dan menghapus data suster.



Gambar 4.16. Halaman Menu Administrasi Suster

4.1.3 Kategori Mahasiswa

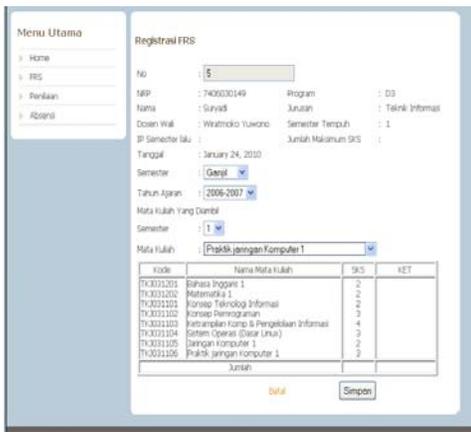
Setelah melakukan login sebagai mahasiswa yang sebelumnya sudah melakukan transaksi SPP terlebih dahulu maka mahasiswa dapat melakukan FRS. Jika belum melakukan transaksi SPP maka mahasiswa tidak melakukan FRS. Mahasiswa juga dapat melihat hasil dari FRS apakah sudah disetujui atau belum oleh dosen wali.



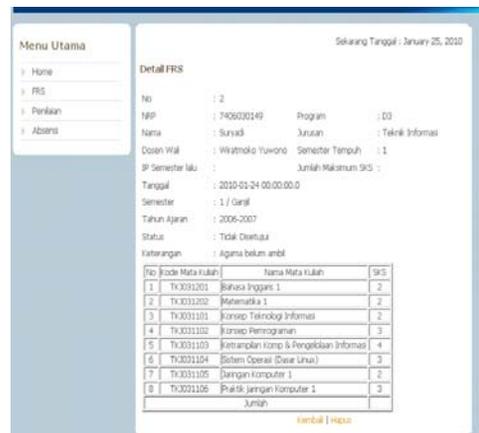
Gambar 4.17. Halaman Menu Mahasiswa



Gambar 4.18. Halaman Menu Mahasiswa FRS



Gambar 4.19. Halaman Menu Mahasiswa Entry FRS



Gambar 4.20. Halaman Menu Mahasiswa Detail FRS

4.1.4 Kategori Dosen

Setelah melakukan login sebagai dosen, maka selanjutnya akan muncul mahasiswa yang sudah melakukan registrasi FRS yang menjadi wali dosennya. Dengan memilih tahun ajaran dan semester terlebih dahulu. Kemudian dosen akan memeriksa FRS mahasiswanya apakah disetujui atau tidak disetujui.



Gambar 4.21. Halaman Menu Dosen



Gambar 4.22. Halaman Menu Dosen

5. Kesimpulan dan saran

5.1 Kesimpulan

Dari hasil uji coba Proyek Akhir dengan judul : “Rancang Bangun Sistem Daftar Ulang PJJ “ini, dapat diperoleh beberapa kesimpulan antara lain :

- Memberikan pelayanan yang optimal kepada mahasiswa, sehingga mahasiswa tidak perlu selalu datang ke kampus untuk melakukan daftar ulang.
- Membuat sistem informasi daftar ulang secara online sehingga memberikan kecepatan, keakuratan dan efisiensi waktu dalam proses pengolahan data dan informasi mengenai formulir rencana studi mahasiswa.
- Mahasiswa tidak dapat melakukan FRS apabila belum melakukan transaksi SPP terlebih dahulu.

5.2 Saran

Proyek akhir ini masih bisa dikembangkan lebih lanjut , beberapa saran untuk pengembangan diberikan berikut ini :

1. Aplikasi yang dibuat pada proyek akhir ini masih perlu untuk disempurnakan lagi. Misal pada proses autentikasi user maupun proses pendaftaran.
2. Ditambah lagi ruang lingkupnya bukan untuk PJJ saja.

6. Daftar Pustaka

- [1] Sano Alb. V. Dian, "24 Jam Menguasai HTML, JSP dan MySQL", CV. Andi Offset, Yogyakarta , 2005
- [2] Rickyanto Isak, "*Belajar Sendiri Java Server Pages*", PT Elex Media Komputindo, Jakarta, 2002
- [3] Heryanto Imam, "Pemrograman PL/SQL Oracle", Informatika Bandung, Bandung, 2003
- [4] Rickyanto Isak, "Java Server Pages", PT Elex Media Komputindo, Jakarta, 2002
- [5] Sampurna, "*Menguasai Aplikasi Web tanpa Pemrograman*", Elex Media Komputindo, Jakarta, 2002
- [6] Referensi Bahasa Pemrograman JSP :
<http://www.echo.net>
- [7] Referensi Bahasa Pemrograman JSP :
<http://www.ilmukomputer.com>
- [8] Referensi Database Oracle :
<http://www.oracle.com>