**PENGAMANAN DATABASE SERVER SISTEM INFORMASI KENDARAAN BEMOTOR DI UNIT KEPOLISIAN MENGGUNAKAN TEKNOLOGI WEB SERVICES**

**(STUDI KASUS SAMSAT KOTA PALEMBANG)**

Ahmad Mutakin Bakti 1,Izman Herdiansyah2 ,Muhammd Akbar3

Software Engineering, Ilmu Komputer, Universitas Bina Darma

email: mutakin@mail.binadarma.ac.id 1, izman@yahoo.com 2 , akbartea@yahoo.com3

ABSTRAK

Penggunaan teknologi client server mengharuskan pengembang mengakses langsung database server. Cara akses dengan metode ini mempunyai resiko lemahnya keamanan data karena developer mengetahui alamat komputer serta user id dan password database server. Penelitian ini bertujuan untuk membuat aplikasi layanan untuk menjembatani aplikasi berbagai instansi atau unit dengan database server Sistem Informasi Kendaraan Bermotor di Unit Kepolisian pada Kantor Samsat Palembang dan dapat mengamankan database server dari akses langsung. Teknologi saat ini yang tengah berkembang pesat adalah sebuah teknologi yang memungkinkan bentuk akhir dari sebuah program atau aplikasi komputer adalah layanan atau fungsi yang melakukan sebuah tugas atau proses yang spesifik dan dikenal dengan istilah Web Services yang menggunakan XML, sebagai format dokumennya. Web Services juga mampu memberi layanan pada bahasa pemerograman tanpa mengharuskan bahasa pemrograman tersebut membuat koneksi langsung ke Database Server akan mengurangi perasaan was-was para Database Administrator, akan ketakutan kehilangan data, karena akses ke database cukup dilakukan oleh aplikasi layanan berupa web services. Aplikasi layanan ini ternyata mampu mengatasi masalah keamanan database server dan mengurangi biaya pengembangan karena tidak harus menyeragamkan bahasa pemerograman

**Kata Kunci:** *Web Services, XML, Database Server*

1. **PENDAHULUAN**

Seiring dengan tuntutan demokrasi tentang transparansi dan kemudahan serta kecepatan pelayanan publik kepada masyarakat, kantor SAMSAT (Sistem Administrasi Manunggal di bawah Satu Atap) Palembang sedang giatnya untuk mewujudkan tatakelola pelayanan pemerintahan yang baik, diantaranya sedang membangun sistem komputerisasi pelayanan yang saling berintegerasi antar instansi didalamnya.

Meskipun beberapa kantor SAMSAT (Sistem Administrasi Manunggal di bawah Satu Atap) sudah ada yang menggunakan fasilitas *on-line* yaitu pada proses pendaftaran kendaraan baru hal tersebut masih dirasa kurang dan terbatas karena dalam pembuatan aplikasinya masih harus dibatasi menggunakan program tertentu saja dan sebagian besar aplikasinya berdiri sendiri. Untuk aplikasi yang sudah ada, pembuatan antara aplikasi server dengan aplikasi client harus menggunakan bahasa pemrograman yang sama, hal ini dirasa kurang efektif karena di beberapa instansi yang ada di kantor SAMSAT yang terdiri dari Kepolisian, Dinas Pendapatan Daerah, Jasaraharja dan BANK sudah mengembangkan aplikasi dengan bahasa pemerograman yang berbeda seperti *INFORMIX .ASP.Net VB 6.0, VB.Net* dan lain-lain.

Saat ini aplikasi yang ada di kantor SAMSAT Palembang yang sudah terintegrasi antar instansi adalah aplikasi yang dimiliki oleh Kepolisian dengan aplikasi yang dimiliki oleh Dinas Pendapatan Daerah (DISPENDA) Sumatera Selatan dengan bentuk integrasi aplikasi Penerimaan Pajak Kendaraan Bermotor yang dimiliki DISPENDA dengan cara langsung mengakses *Database Server* yang dimiliki oleh Kepolisian. Cara akses langsung ini memiliki resiko keamanan data, karena pengembang aplikasi DISPENDA mengetahui *userid* dan *password* untuk *login* ke *Database Server* yang dimiliki kepolisian.

Berdasarkan permasalahan di atas, maka dibutuhkan sebuah aplikasi layanan yang mampu menghubungkan aplikasi *client* baik berbasis *Desktop* maupun berbasis *web* yang dibangun oleh berbagai bahasa pemrograman dengan *Database Server* terutama *Database Server* yang dimiliki kepolisian. Salah satu teknologi Saat ini telah berkembang sebuah teknologi yang memungkinkan bentuk akhir dari sebuah program atau aplikasi komputer adalah berupa sebuah service atau fungsi yang melakukan sebuah tugas atau proses yang spesifik dan dikenal dengan istilah W*eb Services*. Konsep ini sebenarnya bukanlah sebuah konsep yang baru, berbagai teknologi telah dikembangkan untuk mewujudkan kebutuhan akan koneksi atau hubungan antar aplikasi, seperti teknologi *DCOM*  dan *CORBA*. Namun teknologi tersebut dianggap masih memiliki keterbatasan, yaitu: adanya kesulitan untuk dilakukan silang teknologi antara sistem operasi yang satu dengan sistem operasi yang lain, maupun antara satu bahasa pemrograman dengan bahasa pemrograman yang lain. *XML Web Service*s adalah layanan yang dirasa mampu mengatasi permasalahan tersebut*. XML Web Service* itu sendiri merupakan jenis layanan yang menggunakan *XML* sebagai format dokumen dalam pertukaran data dan menggunakan protokol http untuk komunikasi datanya. Dengan menggunakan *XML* sebagai format dokumennya akan memungkinkan *Web Service*s dalam berkomunikasi antar aplikasi dan platform yang berbeda.

Dengan adanya *Web Service*s yang mampu memberi layanan pada bahasa pemerograman tanpa mengharuskan bahasa pemrograman tersebut membuat koneksi langsung ke *Database Server* akan mengurangi perasaan was-was para *Database Administrator* Kepolisian, akan ketakutan kehilangan data kendaraan bermotor. Tentunya resiko kehilangan data ini adalah resiko dari akses *database,* bukannya resiko yang disebabkan oleh hal yang bukan bersifat koneksi seperti kebakaran, membuka *server* langsung dan lain-lain

1. **MODEL, ANALISIS, DESAIN DAN IMPLEMENTASI**
   1. **MODEL PENELITIAN**

Pada penelitian ini menerapkan metode penelitian tindakan *(action research).* Menggunakan metode penelitian inikarena merupakan metode penelitian yang bertujuan mengembangkan pendekatan baru untuk memecahkan masalah atau memperbaiki sistem pemecahan masalah

* 1. **ANALISIS**

Metode analisis menggunakan metode *unified.* Adapun tahapan -tahapannya menurut Haryanto (2004 :380) yang digunakan adalah:

* 1. Berpedoman pada kebutuhan pemakai sistem.
  2. Mengidentifikasikan skenario pemakaian atau *use-case.*
  3. Memilih kelas-kelas dan objek-objek menggunakan kebutuhan sebagai penuntun.
  4. Mengidentifikasi atribut dan operasi untuk masing-masing kelas objek.
  5. Mengidentifikasi struktur dan hirarki kelas-kelas.
  6. Membangun model keterhubungan kelas dan objek.
  7. Melakukan *review* model yang dihasilkan dengan skenario atau *use-case.*
     1. **Analisis Masalah**

Salah satu masalah yang dihadapi oleh pembuat sistem di samsat Palembang karena banyaknya instansi di dalamnya seperti Kepolisian Republik Indonesia, Dinas Pendapatan Daerah, Jasa Raharja dan Bank. Instansi-instansi itu semuanya membuat sistem sendiri-sendiri. Dimulai Dinas Pendapatan Daerah mengembangkan sistem penerimaan pajak menggunakan bahasa pemerograman dan database Informix Base. Kepolisian mengembangkan Sistem Informasi Kendaraan (SIKB) menggunakan bahasa pemerograman Visual Basic.Net dan *database management system* menggunakan Microsoft SQLServer 2005. Sedangkan Jasa Raharja menggunakan ASP.Net dan pengelola database menggunkan Oracle.

Dengan perbedaan ini terjadinya kesulitan menghubungkan (*link)* data antar instansi tersebut. Saat ini di kantor pelayanan samsat Palembang yang sudah terhubung aplikasinya adalah aplikasi yang dimiliki oleh kepolisian dengan aplikasi yang dimiliki oleh Dinas Pendapatan Daerah dengan bentuk *link* aplikasi pajak yang dimiliki oleh Dinas Pendapatan Daerah langsung mengakses database yang dimiliki oleh Kepolisian. Bentuk ini mempunyai kelemahan keamanan data yang disimpan di *database server* karena pengembang dispenda mengetahui *userid* dan *password* untuk *login* ke database server, meskipun hak akses sudah dikunfigurasi.

* + 1. **Analisis Kebutuhan**

Spesifikasi kebutuhan dapat digambarkan dengan diagram *Uses Case*.Diagram ini adalah diagram yang menggambarkan fungsionalitas yang diharapkan dari sistem Administrasi teliti ulang kendaraan di Kepolisian dan pembayaran pajak di Dinas Pendapatan Daerah Sumatera Selatan yang tergabung dalam satu kantor yaitu Samsat. Selain itu desain *Use Case* ini juga untuk menggambarkan kebutuhan sistem dari sudut pandang *user* dengan mengfokuskan pada proses komputerisasi (*automated processes*).

Pada gambar ini memperlihatkan Wajib Pajak (pemilik kendaraan atau yang diwakilkan) mendaftarkan dengan menyerahkan berkas kelengkapan seperti KTP, Surat Tanda Nomor Kendaraan (STNK) Lama dan form isian kepada bagian Pendaftaran dan dilakukan verifikasi. Selanjutnya diinput datanya dengan memasukan nomor polisi oleh orang (Aktor) yang sama yaitu bagian pendaftaran. Itulah sebabnya maka gambar verifikasi termasuk (*include*) atau mempunyai syarat mendaftar dan entry nopolisi harus ada proses verifikasi dahulu. Selanjutnya cetak IDBerkas yang akan merupakan *key input*  untuk proses selanjtnya yaitu penetapan pajak oleh petugas penetapan. Terlihat bahwa entry IDberkas merupakan bagian dari penetapan pajak.

Selanjutnya pembayaran pajak bisa dilakukakn jika status penetapan pajak dan asuransi sudah dilakukan. Terakhir adalah proses cetak notes pajak dan cetak STNK yang mempunyai syarat harus entry pembayaran terlebih dahulu.

****

Gambar 1. Use Case Diagram

* 1. **Perancangan**

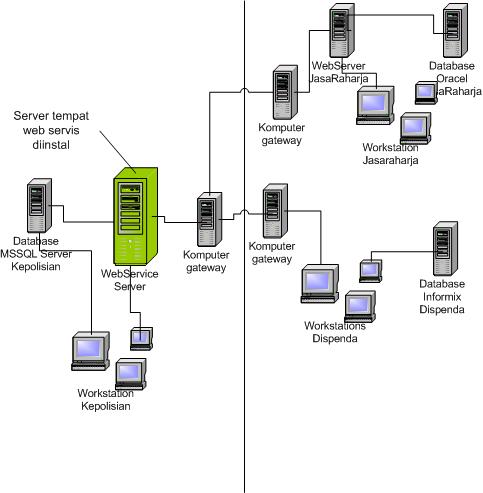
Pada tahapan ini adalah tahapan mendesain Perangkat Lunak *web services*.

* + 1. **Desain Arsitektur Sistem *Web Services***

Desain ini berguna untuk memberi gambaran dan memudahkan kita dalam menempatkan perangkat keras dan lunak yang merupakan sumber daya yang dibutuhkan aplikasi webservices.

Pada *web services* yang nantinya hanya difokuskan pada instansi Kepolisian yang servicenya dapat diberikan Dinas Pendapatan Daerah dan Jasaraharja.

Pada gambar di bawah ini dijelaskan aplikasi *web services* nantinya akan di install di komputer web server yang ada di Kepolisian. sedangkan aplikasi web bisa di install di web server atau aplikasi desktop di computer *client****.*** Semua kegiatan akses database di kerjakan oleh aplikasi *web services*, sehingga sisi keamanan database dapat dijaga. Karena *programmer, user* tidak mengetahui di komputer apa dan dimana letak database server. Sedangkan di Kepolisian dibuat dengan dua kemungkinan bisa akses langsung ke database ataupun aplikasi yang harus mengakses webservice karena dikepolisian sendiri ada aplikasi yang berbeda bahasa pemerogramannya dan dibuat oleh *Programmer* yang berbeda.



Gambar 2. Arsitektur Sistem Web Services

* + 1. **Desain *Database***

Dalam Desain database adalah tahapan mendesain Tabel, View, Strored Procedure serta Trigger. Karena hasil dari penelitian ini hanya model maka tidak semua table akan di desain.

Nama database adalah DBSIKBermotor sedangkan nama table terdiri atas:

1. Tabel TabelIndukKendaraan

Tabel TabelIndukKendaraan merupakan tabel *Secondary* dari tabel TabelIdentitasPemilik dengan *field* relasi kodePemilik. Tabel ini mempunyai field *foriegnKey* yaitu KodeTipe dari tabel TabelMasterTipe, KodeWarna dari Tabel TabelMasterWarna, KodePemilik dari tabel TabelIdentitasPemilik, KodeATPM dari Tabel TabelMasterATPM, KodePendaftaran berasal dari tabel TabelMasterPendaftaran.

Tabel1. Desain Tabel IndukKendaraan

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Field | Type | Ukuran |
| 1 | NIK | Nvarchar | 35 |
| 2 | KodeTipe | int |  |
| 3 | TahunPembuatan | int |  |
| 4 | KodeWarna | Int |  |
| 5 | NoRangka | Nvarchar | 30 |
| 6 | NoMesin | Nvarchar | 30 |
| 7 | KodePemilik | Int |  |
| 8 | NoFaktur | Nvarchar | 35 |
| 9 | KodeATPM | Int |  |
| 10 | KodePendaftaranPertama | Int |  |
| 11 | NoBPKB | Nvarchar | 20 |

1. Tabel TabelIdentitasPemilik

Tabel TabelIdentitasPemilik merupakan tabel *primary* dari Tabel TabelIndukKendaraan. Maksudnya adalah tabel yang pertama kali harus mempunyai nilai dari tabel *secondary* nya yaitu tabel TabelIndukKendaraan. Pada Tabel ini mempunyai *foriegnKey* KodePekerjaan dari TabelMasterPekerjaan.

Tabel2. Desain Tabel TabelIdentitasPemilik

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Field | Tipe | Ukuran |
| 1 | KodePemilik | int |  |
| 2 | NamaPemilik | Nvarchar |  |
| 3 | Alamat | Nvarchar | 100 |
| 4 | NoKTP | Nvarchar | 35 |
| 6 | KodePekerjaan | int |  |

1. Tabel TabelMasterPekerjaan

Tabel TabelMasterPekerjaan adalah tabel *primary* untuk mengisi nilai pada tabel TabelIdentitasPemilik.

Tabel 3. Desain Tabel TabelMasterPekerjaan

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Field | Tipe | Ukuran |
| 1 | KodePekerjaan | Int |  |
| 2 | NamaPekerjaan | Nvarchar | 35 |

1. TabelRegistrasi

Sedangkan Tabel TabelRegistrasi adalah tabel yang digunakan untuk menampung nilai Nopolisi kendaraan tersebut tabel ini merupakan tabel *secondary* dari tabel TabelIndukKendaraan. Tabel ini mempunyai primary key dua *field* yaitu nopol+seriakhir.

Tabel4.Desain Tabel TabelRegistrasi

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Field | Tipe | Ukuran |
| 1 | Nopol | Int |  |
| 2 | SeriAkhir | Nvarchar | 3 |
| 3 | NIK | Nvarchar | 35 |

1. Tabel TabelMasteTipe

Tabel ini adalah tabel yang digunakan untuk menampung nilai tipe kendaraan. Tabel ini adalah tabel *primary* untuk tabel TabelIndukKendaraan dan merupakan tabel *secondary* dari tabel MasterModel yang dihubungkan dengan *field* KodeModel dan *field* KodeMerek dari TabelMasterMerek

Tabel 5. Desain Tabel TabelMasterTipe

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Field | Tipe | Ukuran |
| 1 | KodeTipe | Int |  |
| 2 | Tipe | Nvarchar | 35 |
| 3 | KodeModel | Int |  |
| 4 | KodeMerek | Int |  |

1. Tabel TabelMasterPendaftaran

Tabel TabelMasterpendaftaran adalah Tabel master untuk tabel TabelIndukKendaraan

Tabel 6. Desain Tabel MasterPendaftaran

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Field | Tipe | Ukuran |
| 1 | KodePendaftaran | int |  |
| 2 | NamaPendaftaran | Nvarchar | 35 |

1. Tabel TabelMasterATPM

Tabel TabelMasterATPM adalah tabel *primary* untuk tabel TabelIndukKendaraan yang direlasikan dengan *field* KodeATPM

Tabel 7. Desain Tabel TabelMasterATPM

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Field | Tipe | Ukuran |
| 1 | KodeATPM | int |  |
| 2 | NamaATPM | Nvarchar | 35 |

1. TabelMasterJenis

Tabel TabelMasterJenis adalah tabel untuk menampung data jenis kendaraan yang berelasi dengan tabel TabelMasterTipe

Tabel 8. Desain Tabel TabelMasterJenis

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Field | Tipe | Ukuran |
| 1 | KodeModel | Int |  |
| 2 | NamaModel | Nvarchar | 35 |

1. Tabel TabelMasterModel

Tabel TabelMasterModelKendaraan adalah tabel *primary* untuk tabel TabelMasterTipe

Tabel 9. Desain Tabel TabelMasterModel

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Field | Tipe | Ukuran |
| 1 | KodeModel | Int |  |
| 2 | NamaModel | Nvarchar | 35 |

1. Tabel TabelMasterMerek

Tabel TabelMasteMerekKendaraan adalah tabel *primary* untuk tabel TabelMasterTipe

Tabel 10. Desain Tabel TabelMasterMerek

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Field | Tipe | Ukuran |
| 1 | KodeMerek | Int |  |
| 2 | NamaMerek | Nvarchar | 35 |

1. Tabel TabelPendaftaranSTNK

Tabel TabelPendaftaranSTNK adalah tabel yang digunakan untuk kegiatan transaksi pendaftaran STNK. Pada tabel ini mempunyai *Primary key* IDBerkas sekaligus *Foreign Key* untuk tabel TabelBerkas.

Tabel 11. Desain Tabel TabelPendaftaranSTNK

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Field | Tipe | Ukuran |
| 1 | IDBerkas | Nvarchar | 20 |
| 2 | TglDaftar | DateTime | 35 |
| 3 | TglAwalSTNK | DateTime |  |
| 4 | TglAkhirSTNK | DateTime |  |
| 5 | FlagStatusBayar | Bit |  |

1. Tabel TabelMasterProses

Tabel ini merupakan tabel master dari tabel TabelBerkas.

Tabel 12. Desain Tabel TabelMasterProses

|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Field | Tipe | Ukuran |
| 1 | KodeProses | Int |  |
| 2 | NamaProses | Nvarchar | 35 |

1. Tabel TabelBerkas

Tabel TabelBerkas adalah adalah tabel yang pertama kali diisi sebelum kegiatan transaksi pelayanan di samsat dilakukan.

Tabel 13. Desain Tabel TabelBerkas

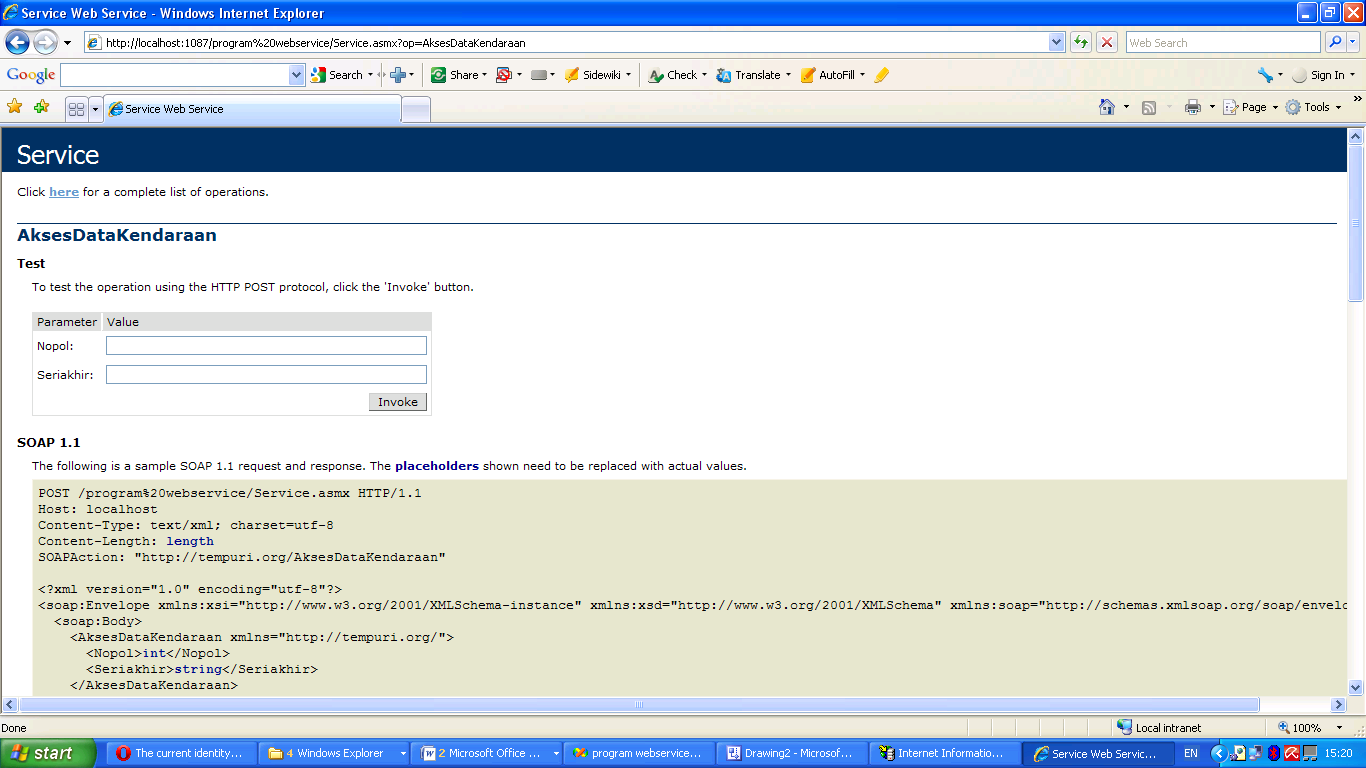
|  |  |  |  |
| --- | --- | --- | --- |
| No | Nama Field | Tipe | Ukuran |
| 1 | IDBerkas | Nvarchar | 20 |
| 2 | NIK | NVarchar | 35 |
| 3 | KodeProses | Int |  |

Pada dasarnya masih banyak lagi tabel yang ada di Sistem Informasi Kendaraan Bermotor. Tetapi yang terlibat langsung ke sistem instansi lain dalam hal ini Dispenda, Jasaraharja dan Bank hanya tabel-tabel di atas. Jadi peneliti hanya fokus pada tabel-tabel di atas.

* + 1. **Desain *Interface* atau Tampilan *Web services***

Sebenarnya *web services* itu adalah layanan tetapi jika kita menggunakan fungsi yang membutuhkan parameter maka tampilannya dapat kita lihat. Dengan demikian tampilan *web services* dapat dibuat. Desain tampilan *web services*s dapat dibuat menyesuaikan dengan *Function* yang memerlukan parameter imputan.

1. Desain Fungsi AksesDataKendaraan

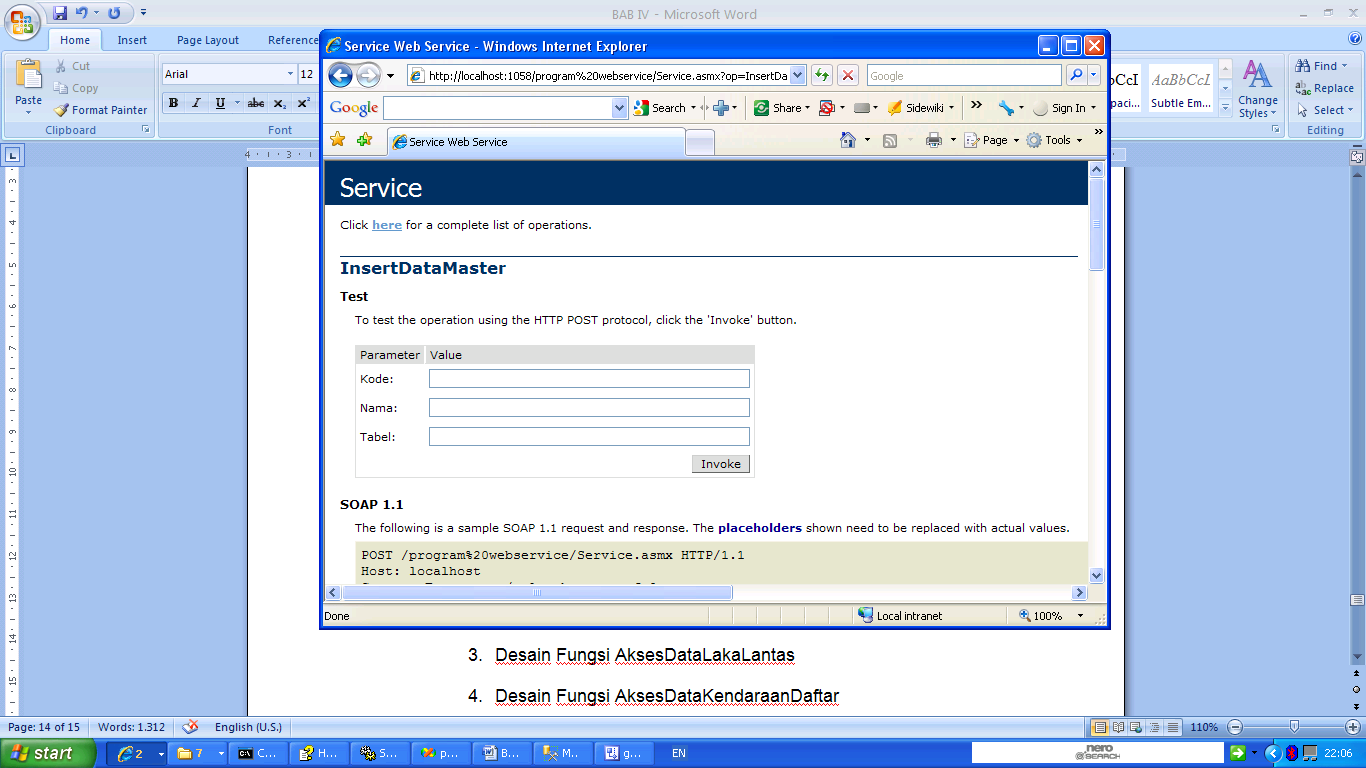


Gambar 3. Desain Tampilan WebService untuk fungsi akses data kendaraan

Tampilan dari webservice ini merupakan tampilan yang akan berupa halaman web (*web page) .* Nopol dan seriakhir adalah parameter yang nantinya akan diproses oleh program *web services*nya ke database server. Sedangkan tombol Imvoke untuk menampilkan hasilnya dalam format XML.

1. Desain Fungsi InsertDataMaster

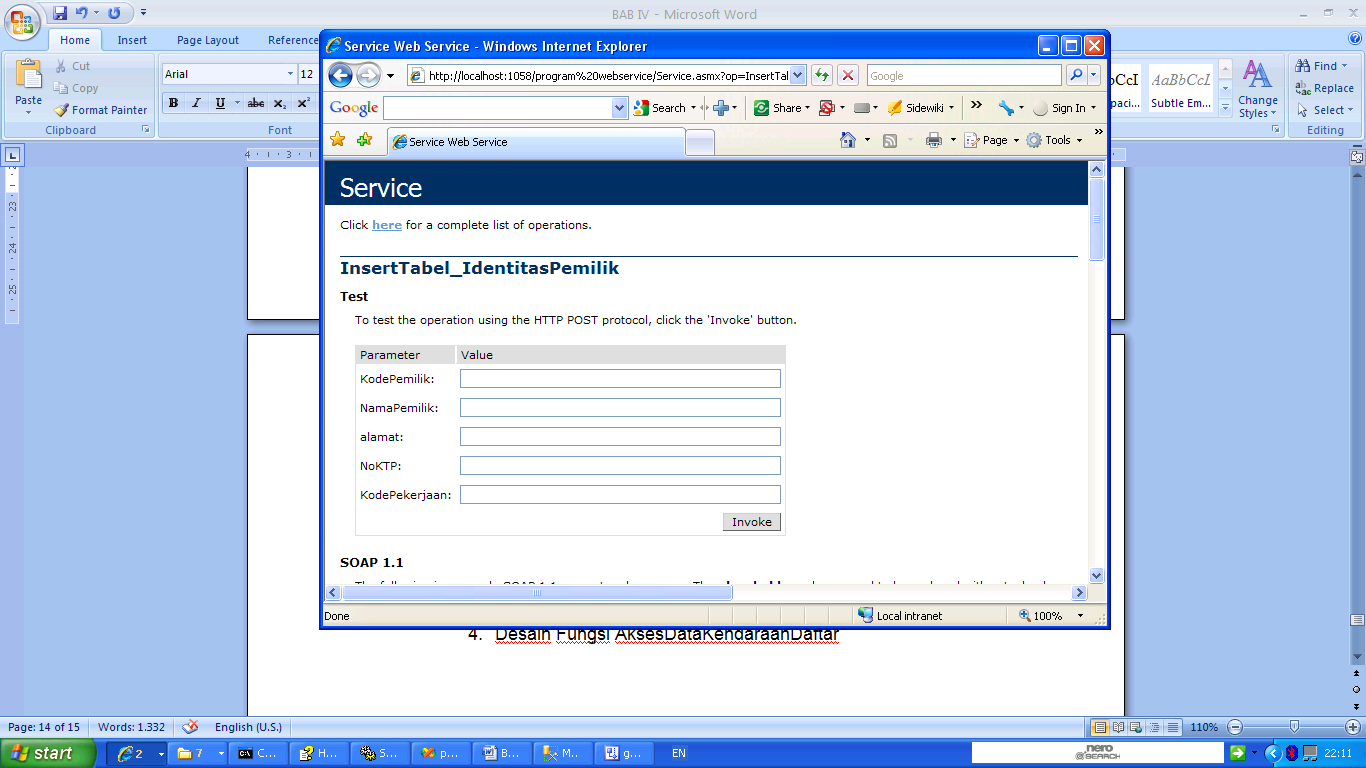
Fungsi ini berguna untuk menyimpan baris data pada tabel -tabel master seperti Tabel TabelMasterMerek, TabelMasterJenis, TabelMasterModel dan tabel master lainnya



Gambar 4. Desain Tampilan WebService untuk fungsi menyimpan data master

1. Desain Fungsi InsertTabel\_IdentitasPemilik

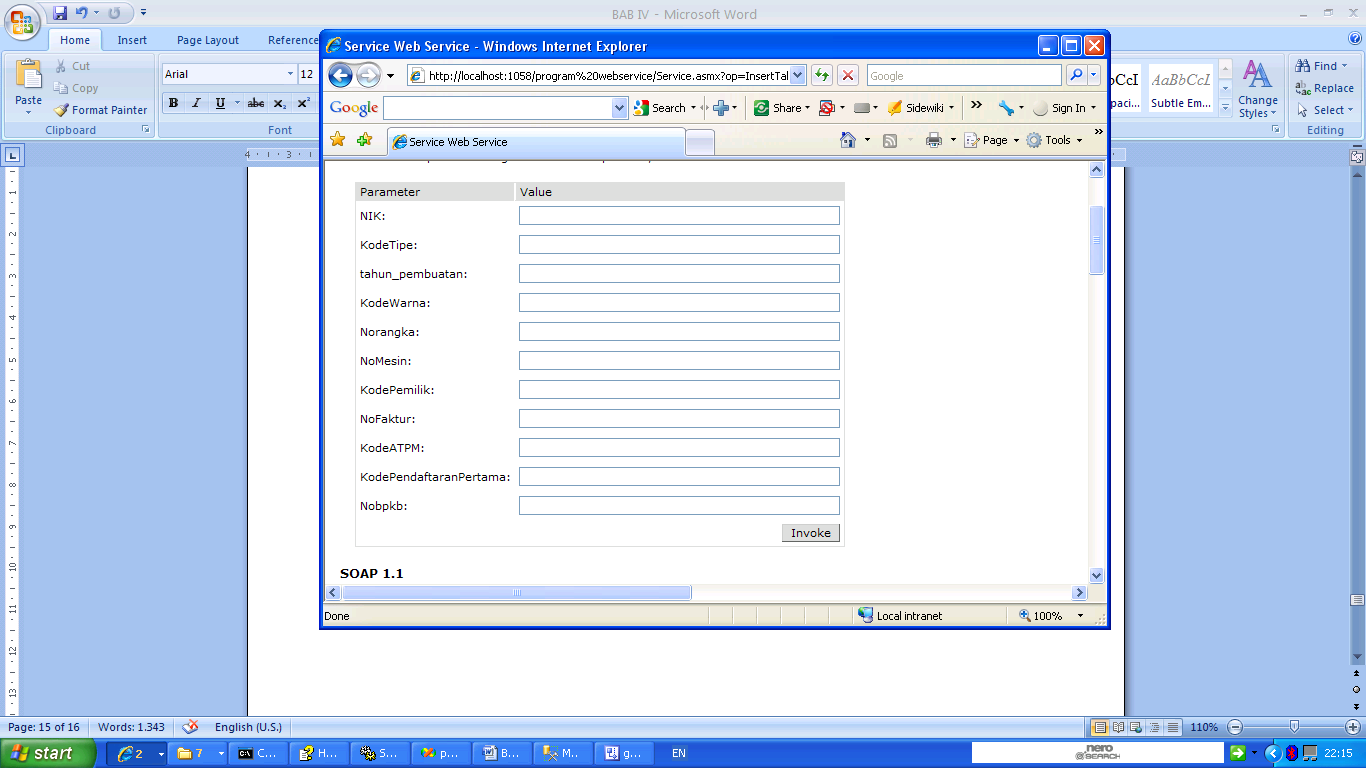
Fungsi ini digunakan untuk menyimpan identitas pemilik yang mempunyai parameter seperti yang terlihat pada gambar 4.7. Prosesnya akan menjalanakan *stored procedure* “InsertTabelIdentitasPemilik”. Hasil file XML nya hanya berupa pesan tersimpan atau tidak tersimpan.



Gambar 5. Desain Tampilan WebService untuk fungsi menyimpan data Identitas Pemilik

1. Desain Fungsi InsertTabel\_Kendaraan

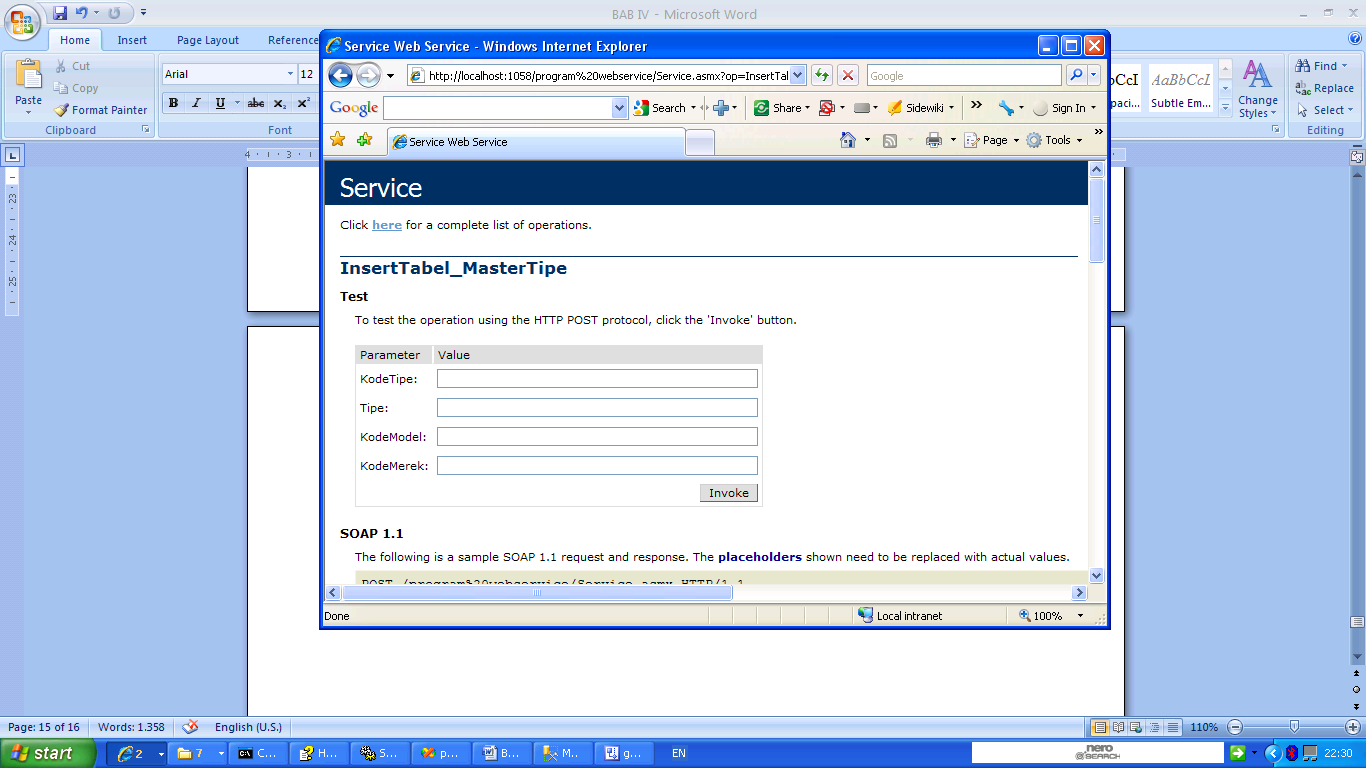
Fungsi ini berfungsi untuk menyimpan data induk kendaraan yang mempunyai parameter nik, kodeTipe, Tahun\_Pembuatan, KodeWarna, Norangka, NoMesin, KodePemilik, NoFaktur, KodeATPM dan Nobpkb. Hasil file xmlnya hanya berupa pesan tersimpan.



Gambar 6. Desain Tampilan WebService untuk fungsi menyimpan data induk kendaraan

1. Desain Fungsi InsertTabelMasterTipe

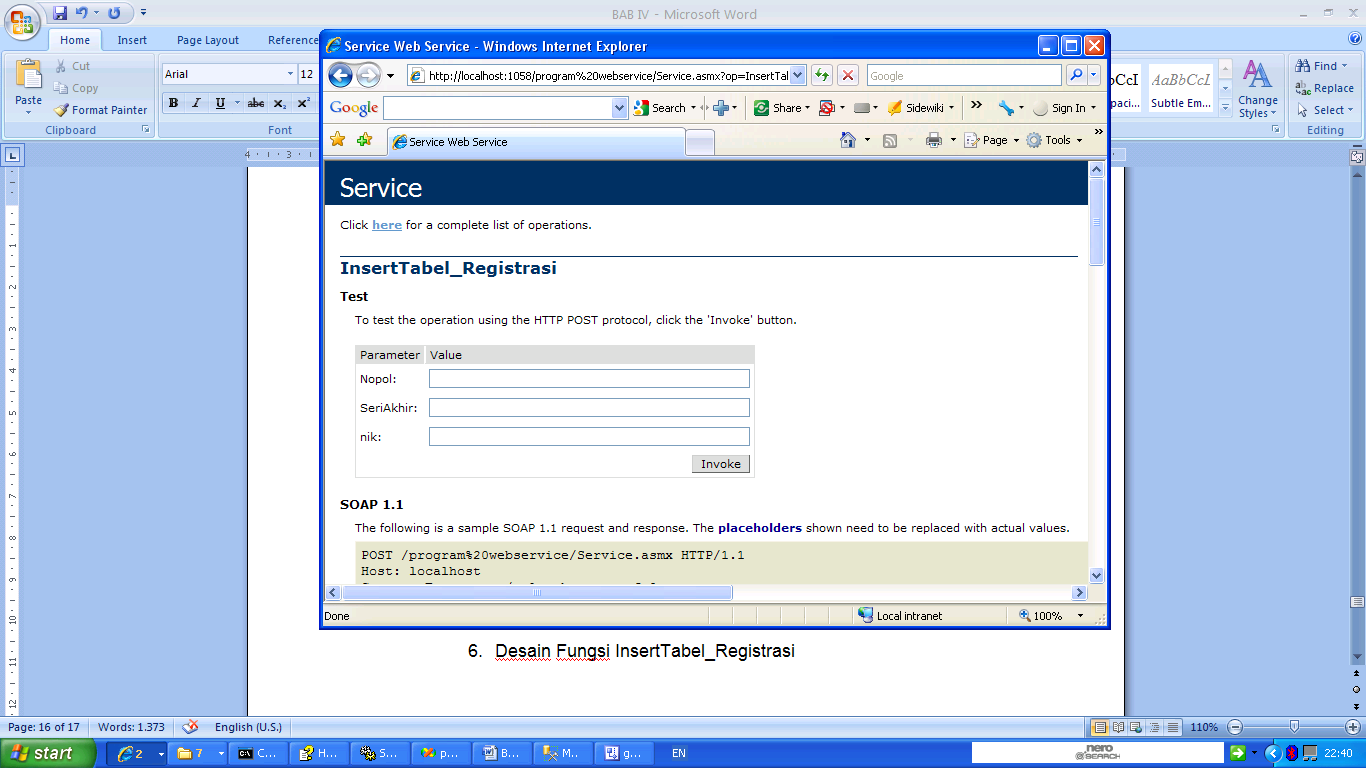
Pada desain fungsi ini terlihat empat parameter untuk menyimpan data kodetipe, tipe, KodeModel, KodeMerek dan akan tersimpan di tabel TabelMasterTipe.



Gambar 7. Desain Tampilan WebService untuk fungsi menyimpan data Master Tipe

1. Desain Fungsi InsertTabel\_Registrasi

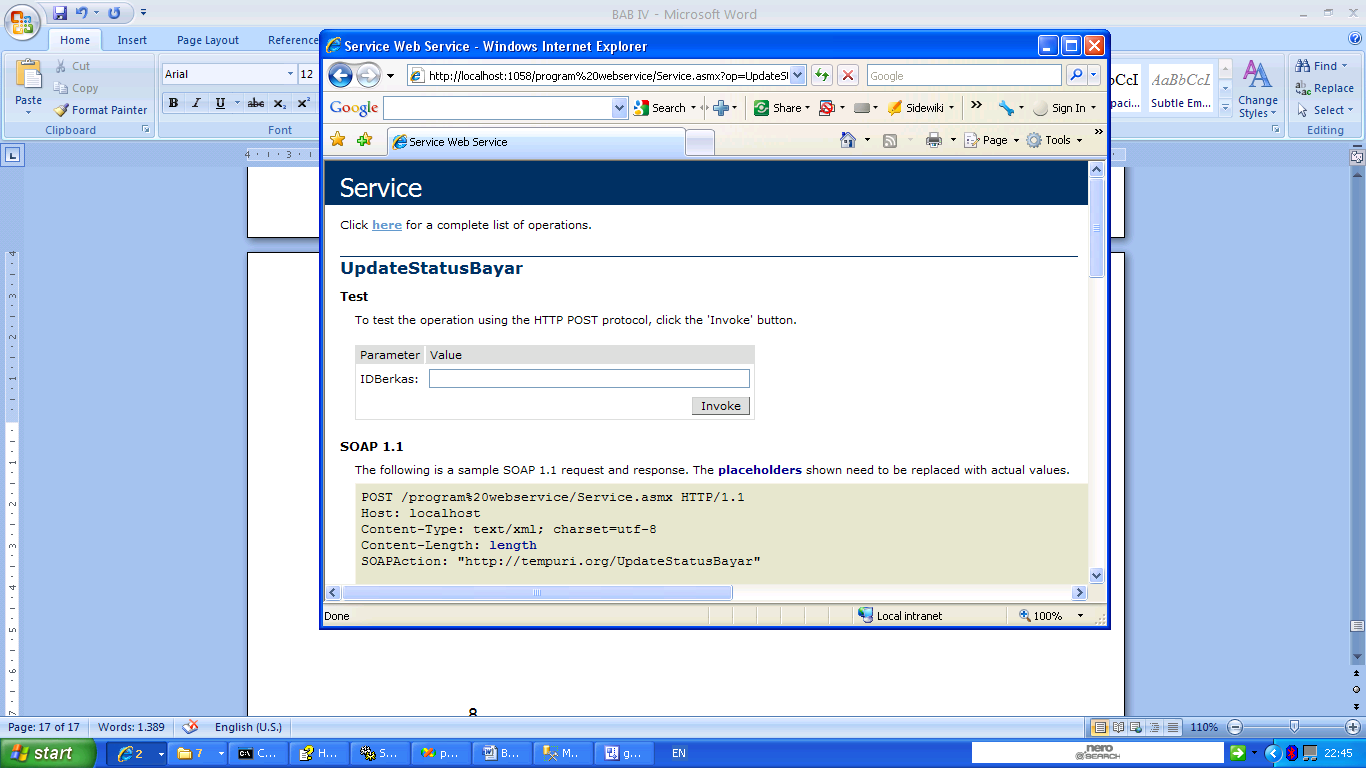
Sama dengan fungsi-fungsi sebelumnya fungsi ini untuk menyimpan data registrasi nopol. Fungsi ini mempunyai parameter Nopol, SeriAkhir dan nik.



Gambar 8. Desain Tampilan WebService untuk fungsi menyimpan data Registasi Nopolisi

1. Desain Fungsi UpdateStatusBayar

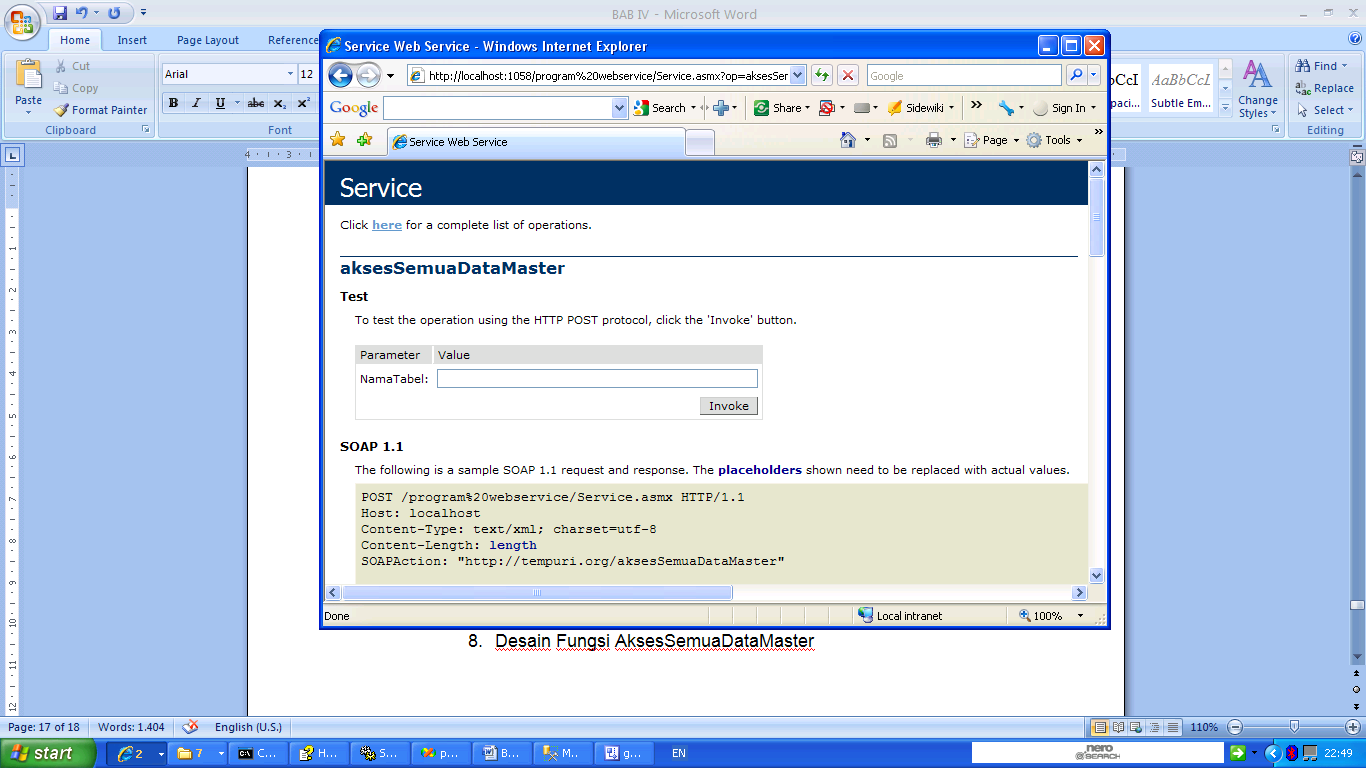
Fungsi ini berguna untuk mengupdate statusbayar menjadi *true* pada tabel tabelpendaftaranstnk. We*bservice* ini nantinya akan diakses oleh aplikasi pembayaran pajak yang dimiliki oleh dinas pendapatan daerah.



Gambar 9. Desain Tampilan WebService untuk fungsi menyimpan Status STNK sudah dibayar

1. Desain Fungsi AksesSemuaDataMaster

Fungsi ini mengakses semua data master terutama data master yang hanya mempunyai dua kolom.



Gambar 10. Desain Tampilan WebService untuk fungsi menampilkan seluruh data master

* + 1. **Desain *Class Diagram***

Desain *Class Diagram* adalah desain blue print dari program. Pada diagram di bawah ini terdiri atas empat class yaitu DataKendaraan, DataMaster, Koneksi, Service,Pesan dan Security serta MyHeader.

1. *Class* Service

*Class* utama pada aplikasi ini adalah service. *Class*  service mempunyai method :

1. “AksesDataKendaraan” yang merupakan *method* untuk menampilkan nilai dari objek yang merupakan implementasi dari class “DataKendaraan”. Karena fungsi atau method ini merupakan fungsi yang bertipe *class* DataKendaraan
2. “InsertDataMaster”

*Method* ini merupakan implementasi dari *class* pesanyang mempunyai atribut atau *field* NilaiPesan. Maka File XMLnya mempunyai atribut sama dengan *class* pesan. File XML inilah yang bias dimanfaatkan *Programmer*  pada aplikasinya untuk memberitahu pengguna bahwa kondisi baris data sudah tersimpan atau tidak.

1. “InsertTabel\_IdentitasPemilik”

*Method* ini merupakan implementasi dari *class pesan* yang mempunyai atribut atau *field* NilaiPesan

1. “InsertTabel\_Registrasi

*Method ini merupakan implementasi* dari *class pesan* yang mempunyai atribut atau *field* NilaiPesan.

1. “InsertTabel\_Kendaraan”

*Method* ini merupakan implementasi dari *class pesan* yang mempunyai atribut atau *field* NilaiPesan.

1. “InsertTabel\_MasterTipe
2. “UpdateStatusBayar”
3. “AksesSemuaDataMaster”

*Method* ini merupakan implementasi dari *class* DataMaster yang mempunyai atribut atau *field* Kode dan Nama yang bertipe *Array Integer* dan *Array String*. *File XML* dari fungsi ini dapat dimanfaatkan oleh pengembang untuk ditampilkan misalnya di *combo box* dan lain-lain

1. *Class* DataKendaraan

Kelas ini adalah *Base Class*  untuk kelas *Service*. Pada kelas DataKendaraan hanya mempunyai atribut.

1. *Class*  DataMaster

Kelas ini adalah *Base Class*  untuk kelas service. Hubungan satu ke banyak karena kelas ini banyak di implementasikan di kelas *service*.

1. *Class* Koneksi

Pada kelas ini mempunyai dua *method* yaitu Konek dan diskonek. Sedangkan atribut yang dimiliki hanya satu yaitu kon yang implementasi dari *base* *class SqlConnection* yang sudah disediahkan *Viusal Basic Net.2010.* Kelas diimplementasikan pada kelas *service*

1. *Class* Myheader

Kelas ini dibuat untuk menjaga keamanan aplikasi *web service*. Diimplementasikan pada kelas *Security.* Atribut yang dimilik adalah adalah *password* dan *username* yang keduanya bertipe string.

1. *Class Security*

*Security* adalah kelas yang digunakan untuk menjaga keamanan *web services.* Kelas ini mempunyai *method*  cekpassword yang memiliki parameter *userid* dan *password* bertipe string. Sedangkan atribut yang dimiliki *passwordok* yang bertipe *Boolean* dan *sHeader* yang bertipe kelas *MyHeader* . jadi pada kelas *security* mempunyai atribut yang merupakan implementasi dari *base class MyHeader* yaitu sHeader.

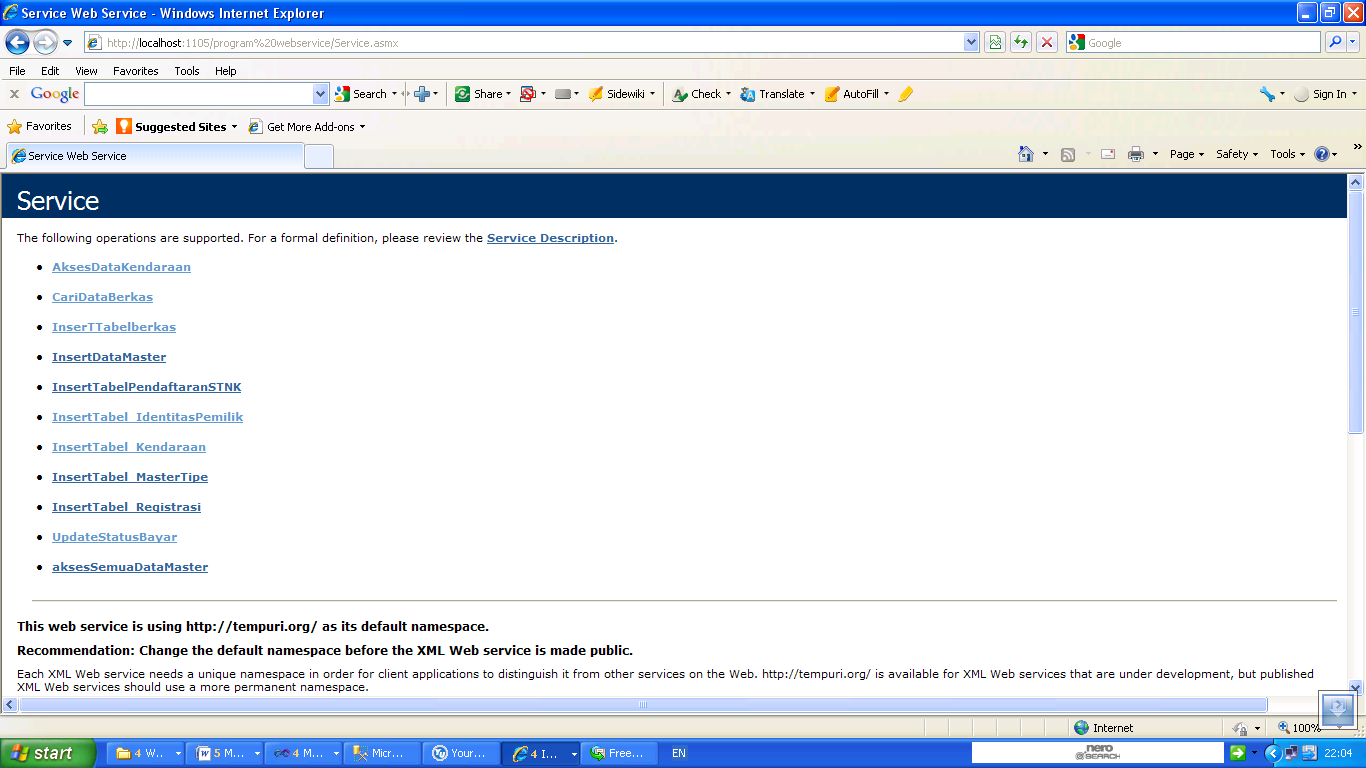


Gambar 11*. Class Diagram*

* 1. **HASIL**

Hasil dari penelitian adalah sebuah perangkat lunak layanan yang menggunakan teknologi *web services*. Aplikasi ini ini adalah aplikasi yang berbasis *web*, maka tentunya membutuhkan perangkat lunak *web server*. Perangkat Lunak *web server* yang digunakan adalah *Internet Information Services (IIS)*  yang sudah melekat di sistem operasi *windows*.

Aplikasi *web services* ini diletakan di *web server* untuk pembahasan ini diletakan di lingkungan jaringan local dengan alamat [*http://localhost/webservice/Service.asmx*](http://localhost/webservice/Service.asmx). *Service* ini mempunyai 11 layanan utama seperti yang terlihat pada gambar di bawah ini :

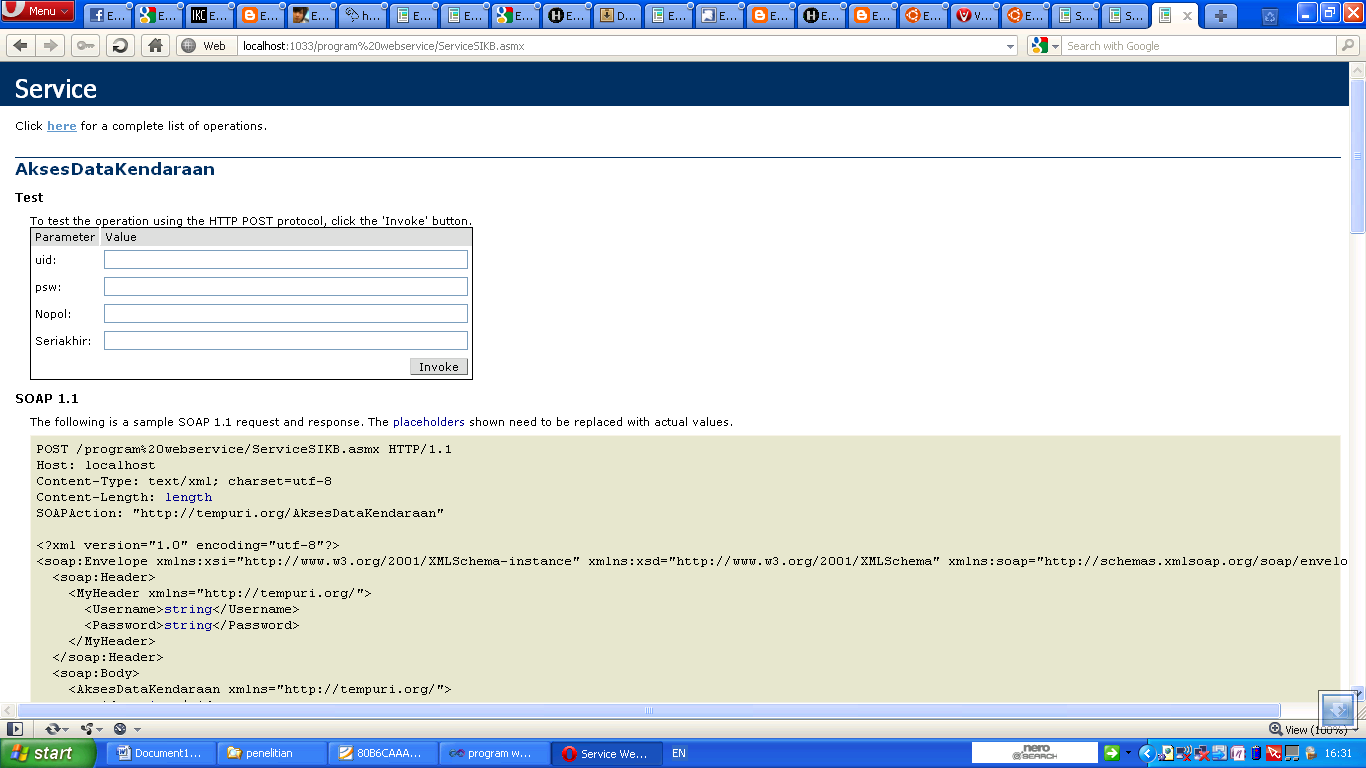


Gambar 12. Tampilan Layanan Web services

Kesepuluh layanan tersebut adalah AksesDataKendaraan, CariDataBerkas, InsertTabelBerkas, Insert DataMaster, InsertTabelPendaftaranSTNK, InsertPemilik, InsertMasterKendaraan, InsertMasterTipe, InsertRegistrasi, UpdateStatusBayar, AksesSemuaDataMaster. Detail dari layanan-layanan ini sebagai berikut

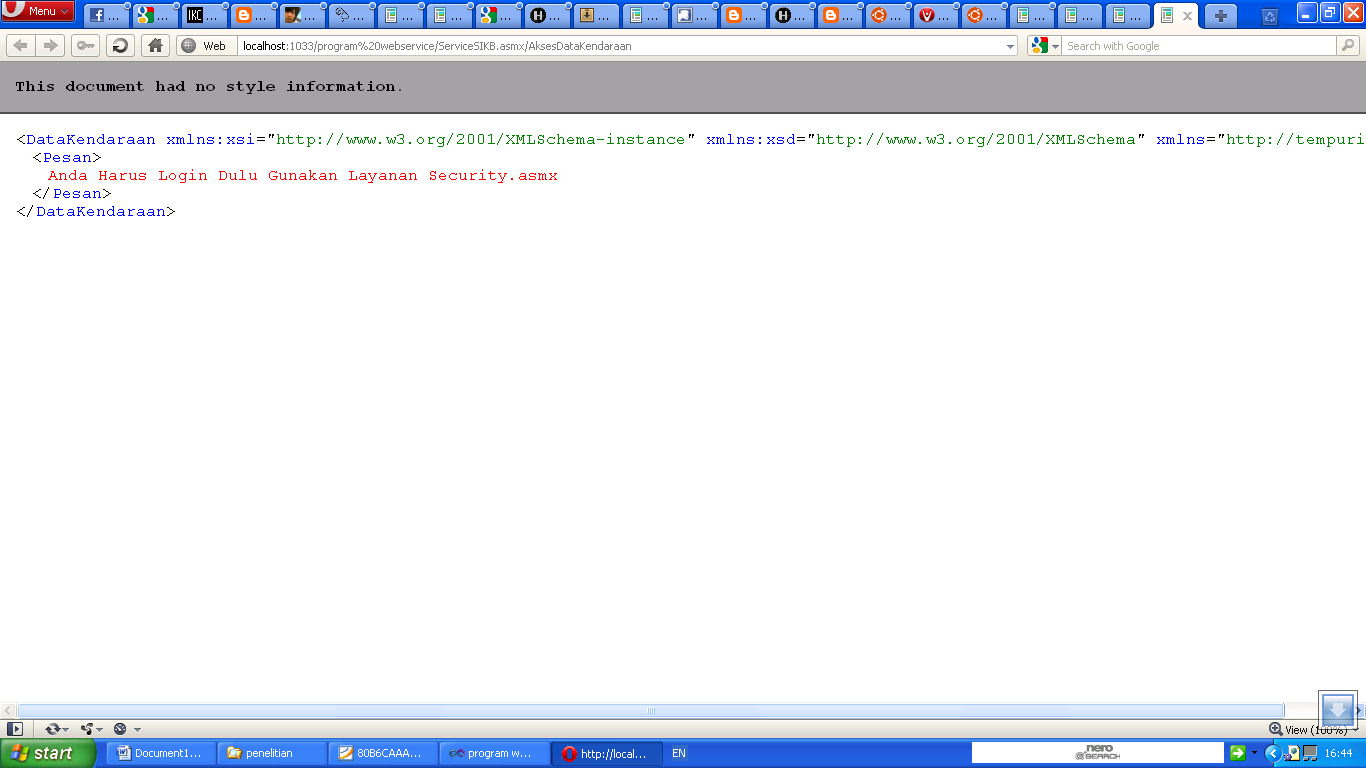
1. Layanan AksesDataKendaraan

Service ini digunakan untuk menampilkan data kendaraan termasuk pemiliknya. Nantinya service ini salah satunya digunakan oleh instansi dinas pendapatan daerah untuk digunakan didalam aplikasinya. Layanan ini mempunyai 4 paramater, yaitu userid, password nomor polisi dan seri nomor polisi. Bentuk tampilan dari layanan atau *service* ini adalah :



Gambar 13 Layanan AksesDataKendaraan ServiceSIKB

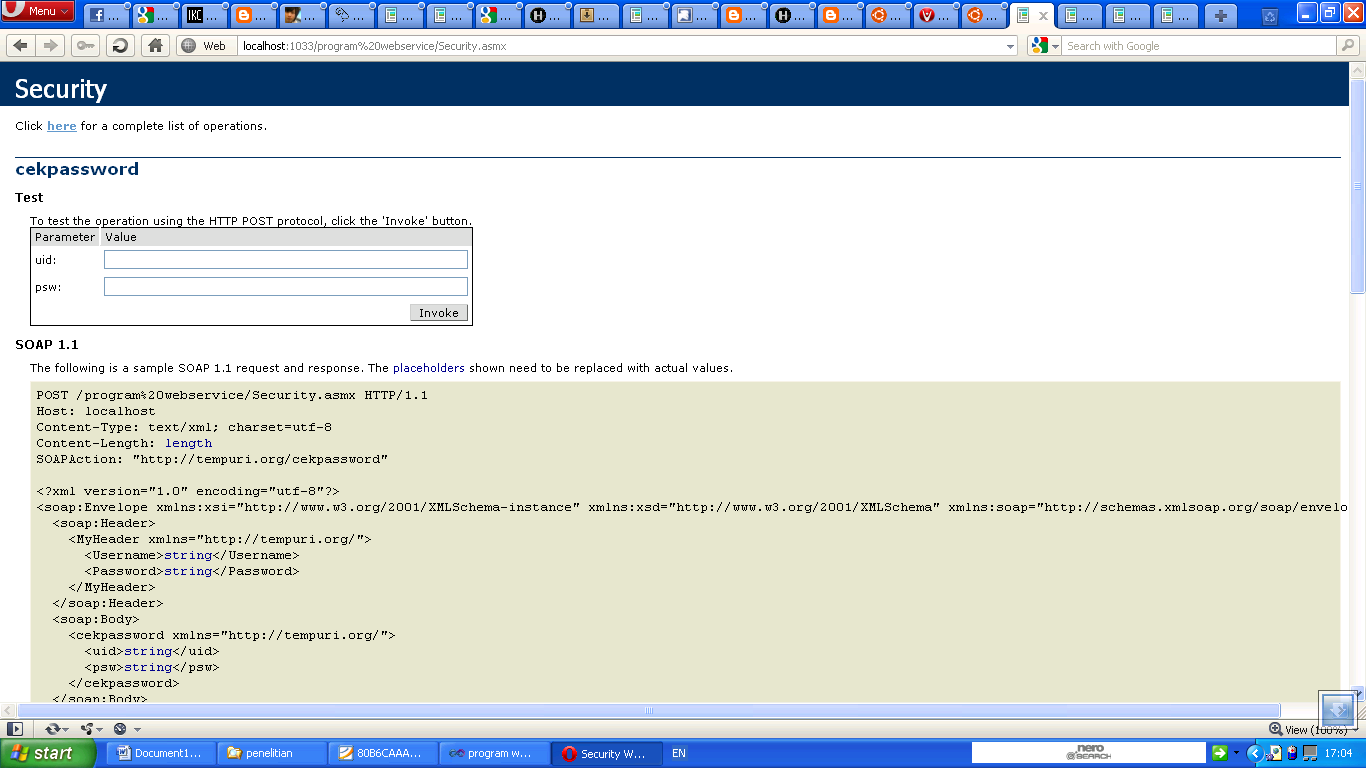
Kita akan mencoba mengisi nilai empat parameter yang diminta. Maka *file XML* yang dihasilkan dari layanan ini seperti terlihat dibawah ini :



Layanan ini sengaja dibuat dengan memaksa pengguna menggunakan bahasa pemerograman untuk mengakses data *XML* nya. Yaitu dengan cara memasukan user id dan password. Nilai kedua parameter ini diambil dari *web services* yang berbeda yaitu *Security*.

1. Layanan “CekPassword” pada *Web Services Security*

Pada *web services* ini mempunyai layanan yaitu cekpassword. Layanan ini mempunyai parameter yaitu *user id* dan *password*



Gambar 14. Layanan cekpassword pada web services security

* 1. **KESIMPULAN**

Setelah melalui tahap perancangan dan pembahasan, maka dapat ditarik kesimpulan bahwa *Web Service*  Penyedia Layanan ini mempunyai keunggulan

1. Mampu mensinergikan berbagai bahasa pemerograman karena komunikasi datanya menggunakan data dengan format *XML*.
2. Bahasa pemerograman dapat berkomunikasi dengan *database server* tanpa harus koneksi langsung
3. Dengan tidak koneksi langsung ke *database server*  maka keamanan data di *database server* dapat terjamin
4. Menggunakan protokol *http* sebagai komunikasi data, sehingga tidak perlu lagi untuk membangun jaringan pribadi dalam implementasinya
5. Karena menggunakan protocol *http* maka bahasa pemerograman yang berbasis *desktop* dapat mengakses data meskipun melalui jaringan *internet*.

**REFERENSI**

1. Santosa, Budi. 2008. ”*ANALISA DAN PERANCANGAN WEB SERVICES UNTUK SISTEM INFORMASI UNIVERSITAS*”. <http://yudiagusta.files.wordpress.com/2009/11/267-272-knsi08-047-analisa-dan-perancangan-web-services-untuk-sistem-informasi-universitas.pdf>. Diakses tanggal 18 Februari 2011 .
2. Berlilana. 2008. “*Web Service Sebagai Penyedia Layanan Administrasi Pada Penjualan Tiket Bus*”. <http://p3m.amikom.ac.id/p3m/dasi/maret08/>. Diakses tanggal 3 Maret 2010
3. Bennet, Simon., McRobb, Steve., Farmer, Ray. 2006. “*Object Oriented System Analysis and Design*” . McGraw-Hill Education. New York
4. Duthie G, Andrew. 2002. ”*Microsoft ASP.Net Step by Step*”. PT. Elexmedia Komputindo. Jakarta
5. Hadiwinata, Mario. 2003. “*Solusi Pemrograman XML Web Service Dengan Visual Basic.Net*”. PT Elex Media Komputindo, Jakarta
6. Hariyadi, Efendi. 2009. ” *OOP (Object Oriented Programming)*”. <http://www.bongkar.co.id/informasi-teknologi/hard-coding/817-oop-object-oriented-programming-part-1.html>. Diakses tanggal 18 Februari 2011
7. Hariyanto, Bambang. 2004. ”*Sistem Manajemen Basis Data*”.
8. Kuswandi, Riana., Fahmi, Husni., Faidah, Haret., Setiadi, Herald., Hasibuan Zainal. 2009. “*Konsolidasi Database Kependudukan Nasional*”. <http://restama.com/ebook/konsolidasi-database-kependudukan-nasional/> . Diakses 18 februari 2011
9. Leong, Marlon. 2004. “*Pemerograman Dasar Microsoft Visual Basic.NET*”. Andi Yogyakarta. Yogyakarta
10. Sukrisno,Mardiayanto. 2006. ”*Validasi Perangkat Lunak dengan metode Hybrid berbasis UML*”, Proseding Konferensi Nasional Teknologi Informasi & Komunikasi untuk Indonesia”, Mei 2006
11. Mutakin, Bakti. 2010. ”Buku Petunjuk Penggunaan Sistem Informasi Kendaraan Bermotor SIKB Versi. 1.0”. Direktorat Lalu Lintas Polda Sumsel
12. NIIT. 2009. *Introduction to Web Content Development*. India
13. Nugroho, Adi. 2005. ”*Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek”*. Edisi Revisi. Informatika: Bandung.
14. Nugroho, Adi. 2004. ”*Konsep Pengembangan Sistem Basis Data”.* Cetakan Pertama.INFORMATIKA. Bandung
15. Priyambodo, Tri Kuntoro. 2005. “*IMPLEMENTASI WEB - SERVICE UNTUK PENGEMBANGAN LAYANAN PARIWISATA TERPADU*”. TEKNOIN, Vol. 10, No. 2, Juni 2005, 105-118
16. Rizky Wicaksono, Soetam. 2008. “*IMPLEMENTASI EXTENSIBLE MARKUP LANGUAGE WEB SERVICE PADA SECURE REMOTE FORM FIELD FILLIN DALAM DOKUMEN MICROSOFT WORD*”. 104 GEMATIKA JURNAL MANAJEMEN INFORMATIKA, VOLUME 9 NOMOR 2, JUNI 2008
17. Wicaksono, Narenda., Primadhanty, Audi. 2006. ”*Sistem Layanan Pesan Pada Telepon Genggam GSM dengan Memanfaatkan GPRS dan Web Services*”. Sekolah Teknik Elektro dan Informatika ITB. Bandung
18. Wulandari , Lily dan I Wayan Simri Wicaksana. 2008. ”*Toward Web Service*”. Proceeding, Seminar Ilmiah Nasional Komputer dan Sistem Intelijen (KOMMIT 2006) Auditorium Universitas Gunadarma, Depok, 23-24 Agustus 2006. Depok