

MEMBANGUN *WEB SERVICE* PADA SISTEM INFORMASI PERIZINAN ANGKUTAN UMUM DENGAN METODE *EXTREME PROGRAMMING* (STUDI KASUS DINAS PERHUBUNGAN KABUPATEN BANDUNG BARAT)

¹Fathiyyah Nur Azizah, ²Nia Ambarsari, ³Ridha Hanafi

^{1,2,3}Program Studi Sistem Informasi, Fakultas Rekayasa Industri, Universitas Telkom

¹nfathiyyah@gmail.com, ²ambarsarinia@gmail.com, ³ridhanafi@gmail.com

Abstrak—Dinas Perhubungan Kabupaten Bandung Barat merupakan dinas pemerintahan yang bertugas dalam bidang perhubungan, salah satunya dalam pemberian izin di Kabupaten Bandung Barat itu sendiri. Proses perizinan terbilang rumit dikarenakan pemohon harus datang ke loket informasi untuk mendapatkan form perizinan, kemudian mengembalikan form beserta pengumpulan berkas persyaratan administrasi. Saat pengambilan dan pengembalian form terjadi antrian, karena yang mengajukan perizinan tidaklah sedikit. Bahkan selain itu, surat perizinan yang memiliki tanggal *expire-date* kurang menjadi perhatian bagi para pemohonnya. Sehingga terkadang surat perizinan yang dimiliki sudah kadaluarsa. Di sisi lain, ketika staff dishub melakukan pemeriksaan angkutan umum di terminal maupun di beberapa titik pemeriksaan, ada beberapa angkutan umum yang tidak memiliki surat dengan lengkap atau bahkan suratnya tidak berlaku. Berdasarkan hal tersebut maka akan diaplikasikan *web service* dan website sistem informasi perizinan angkutan umum bernama “WS.PAT” yang dapat membantu pemohon dalam melakukan pengajuan secara online, mengetahui informasi perizinan yang sedang diajukan, serta memudahkan mendapat informasi untuk melakukan perpanjangan. Adapun aplikasi *web service* sendiri digunakan sebagai backup database demi menjaga keamanan data Dinas Perhubungan sendiri. Dimana ada beberapa aplikasi yang dapat diakses oleh pihak eksternal (pemohon) untuk mengajukan perizinan, sehingga untuk meminimalisir kejadian yang tidak diharapkan, maka digunakannya aplikasi *web service* ini. Selain itu juga, aplikasi *web service* ini digunakan untuk menghubungkan antara aplikasi berbasis intranet dan android. Sehingga keduanya dapat terhubung dalam satu database. Metode yang digunakan yaitu *extreme programming*, karena *extreme programming* memiliki kelebihan dalam menjalin komunikasi yang baik dengan *client* serta meningkatkan komunikasi dan sifat saling menghargai antar *developer*. Selain itu, *extreme programming* digunakan saat *client* membutuhkan waktu yang cepat dalam pembuatan sistem.

Kata kunci : *web service*, *extreme programming*, sistem informasi perizinan, dinas perhubungan

I. PENDAHULUAN

Kemajuan teknologi informasi menjadi kebutuhan yang penting dalam mendukung kegiatan operasional suatu organisasi. Pada sebuah sistem informasi yang handal dan terintegrasi dibutuhkan untuk menyajikan informasi yang bermanfaat guna mendukung pengambilan keputusan secara tepat dan bermanfaat bagi semua pihak. Sistem informasi juga diharapkan dapat mengadopsi perubahan proses bisnis yang terjadi pada organisasi dengan tepat dan cepat.

Dinas Perhubungan Kabupaten Bandung Barat merupakan dinas pemerintahan yang bertugas dalam bidang perhubungan, yaitu pemberian izin atas pengajuan perizinan angkutan umum di Kabupaten Bandung Barat. Dalam pengajuan perizinan angkutan umum, pemohon diwajibkan untuk memenuhi persyaratan administrasi, yuridis, teknis, serta waktu dan biaya. Setelah itu, pemohon akan mendapatkan surat perizinan jika pengajuan telah mendapat persetujuan.

Proses pengajuan perizinan angkutan umum terbilang rumit, dikarenakan pemohon harus datang ke loket informasi untuk mendapatkan form pengajuan perizinan angkutan umum, kemudian mengembalikan form beserta pengumpulan berkas persyaratan administrasi. Saat pengambilan dan pengembalian form terjadi antrian, karena yang mengajukan perizinan angkutan umum tidak hanya satu atau dua orang, bahkan lebih. Selain itu, surat perizinan harus dilakukan pengecekan tersendiri untuk mengetahui *expired-date* atas surat perizinan tersebut. Di sisi lain, ketika staff dishub akan melakukan pemeriksaan terhadap angkutan umum baik di terminal maupun di beberapa titik pemeriksaan razia, ada beberapa angkutan umum yang tidak memiliki surat dengan lengkap atau bahkan suratnya sudah tidak berlaku lagi.

Oleh karena itu, maka dibuatkan sistem informasi perizinan angkutan umum untuk melakukan pengajuan secara online, yaitu menggunakan website yang berbasis internet. Website ini, selain digunakan untuk melakukan pengajuan secara online, tetapi juga dapat mengetahui mengenai informasi yang berkaitan dengan perizinan. Selain itu, sistem informasi ini juga akan berhubungan dengan proses

pengawasan. Proses pengawasan dilakukan oleh staff dinas perhubungan saat melakukan pemeriksaan angkutan umum di lapangan.

Pada penelitian ini, *web service* digunakan sebagai sistem yang menjembatani antara 3 aplikasi yang berbeda platform, yakni data perizinan Kantor Dinas Perhubungan Kabupaten Bandung Barat berbasis intranet, kemudian sistem informasi perizinan angkutan umum sendiri berbasis internet, serta sistem pengawasan berbasis android. Selain itu, *web service* digunakan untuk melakukan integrasi database sebagai backup data, karena database pada sistem informasi berbasis internet dapat diakses oleh pemohon (bukan staff dinas perhubungan Kabupaten Bandung Barat) sedangkan database pada perizinan berbasis intranet hanya dapat diakses oleh staff dinas perhubungan Kabupaten Bandung Barat. *Web service* dibangun berdasarkan arsitektur yang terstruktur, karena *web service* dapat menyediakan data maupun fungsi tertentu bagi aplikasi lain meskipun berbeda sistem operasi, perangkat keras, maupun bahasa pemrograman yang digunakan untuk membangunnya. *Web service* dapat mempermudah dalam distribusi sekaligus integrasi database yang tersebar di beberapa server sekaligus.

Web service sebagai suatu sistem perangkat lunak yang dirancang untuk mendukung interoperabilitas dan interaksi antar mesin/ sistem pada suatu jaringan [1]. *Web service* digunakan sebagai suatu fasilitas yang disediakan oleh suatu *website* untuk menyediakan layanan (dalam bentuk informasi) kepada sistem lain, sehingga sistem lain dapat berinteraksi dengan sistem tersebut melalui layanan-layanan (*services*) yang disediakan oleh suatu sistem yang menyediakan *web service*. *Web service* menyimpan data informasi dalam format pesan universal (misal: XML dan JSON), sehingga data ini dapat diakses oleh sistem lain walaupun berbeda *platform*, sistem operasi, maupun bahasa *compiler*.

Arsitektur yang ada pada *web service* salah satunya adalah *Simple Oriented Architecture* (SOA), yang memungkinkan kita untuk dapat terhubung berbagai jenis perangkat lunak yang memiliki basis data dan sistem operasi yang berbeda untuk saling bertukar data [2]. Hal ini memungkinkan karena *web service* yang menggunakan arsitektur SOA menggunakan sebuah standar format data yang universal yaitu XML *web service* [3].

Oleh karena itu, maka perancangan dan pembangunan sistem integrasi antar sistem informasi yang memiliki jenis basis data yang berbeda dan platform berbeda dengan cara membuat suatu *service* pada sistem untuk melakukan tugas secara spesifik. Model arsitektur *web service* dengan menggunakan *Simple Oriented Architecture* (SOA) dan protokol *web service* yang digunakan adalah SOAP.

Service Oriented Architecture (SOA) merupakan pendekatan untuk membangun sistem perangkat lunak yang kompleks dengan membentuk satu set layanan yang dapat digunakan kembali serta mematuhi prinsip-prinsip *service-oriented*. Selain itu juga memungkinkan respon yang lebih cepat, lebih efektif sehingga hemat biaya, serta responsif terhadap perubahan kondisi [4].

SOAP merupakan protokol standar yang ringan dan ditujukan untuk pertukaran informasi dalam desentralisasi sistem [5]. Protokol ini melakukan pemanggilan metode dalam bahasa XML (*Extensible Markup Language*). Namun, selain

itu juga kita dapat menggunakan format pertukaran data lainnya yaitu JSON (*JavaScript Object Notation*). Dengan demikian, SOAP adalah suatu mekanisme yang sederhana untuk melakukan pertukaran struktur dan tipe informasi dalam lingkungan yang tersebar dan terdistribusi menggunakan XML/ JSON. Dokumen SOAP yang digunakan untuk melakukan *request* disebut SOAP *request* sedangkan dokumen SOAP yang diperoleh dari *web service* disebut dengan SOAP *response*.

JSON (*JavaScript Object Notation*) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer [6]. Adapun jika menggunakan JSON, hanya butuh akses melalui *javascript* dan jika ingin ditampilkan hanya dengan menggunakan HTML DOM yaitu dengan cara mengambil data dari JSON melalui *array javascript* lalu set datanya ke HTML maka data akan tampil. Keuntungan dari JSON juga terdapat pada saat mengirim data banyak ke *server* atau sebaliknya. Karena JSON menggunakan sintak *javascript* dan lebih mudah diakses oleh *javascript* itu sendiri sehingga JSON mendapat *performance* kecepatan yang lebih dibandingkan dengan xml. Hal ini dikarenakan dalam xml jika mengakses sebuah data maka butuh *request server* dan DOM xml itu sendiri.

Perizinan angkutan umum, terbagi menjadi 4 (empat) jenis [7]. Diantaranya yaitu perizinan usaha (usaha menyelenggarakan angkutan orang atau barang dengan kendaraan bermotor yang berdomisili dalam wilayah Kabupaten Bandung Barat), perizinan operasi (Untuk melakukan kegiatan angkutan tidak dalam trayek), perizinan trayek (izin untuk mengangkut orang dengan mobil dan atau mobil penumpang umum pada jaringan trayek), dan perizinan insidental (izin yang hanya diberikan kepada perusahaan angkutan yang telah memiliki izin trayek untuk menggunakan kendaraan bermotor cadangannya menyimpang dari izin trayek yang dimiliki).

Aplikasi untuk perizinan online tersebut akan memiliki beberapa fitur, diantaranya yaitu pengajuan online, pendaftaran sebagai member, melihat data pemohon dan data kendaraan, serta melakukan penambahan pengajuan dan perpanjangan perizinan. Dimana data tersebut diambil dari data intranet yang diproses menggunakan *scheduler* untuk melakukan *sharing data* dari intranet ke internet dalam suatu kurun waktu yang sudah ditetapkan.

Adapun metode penelitian yang akan digunakan, yaitu metode *extreme programming*. *Extreme programming* (XP) adalah salah satu pengembangan *software* berdasarkan nilai-nilai kesederhanaan, komunikasi, umpan balik, dan keberanian. Bekerja dengan membawa seluruh tim bersamasama pada praktik yang sederhana, cukup dengan umpan balik untuk memungkinkan tim dalam menentukan di mana mereka berada, serta untuk menyesuaikan praktek dalam situasi unik mereka [8].

Pada penelitian yang dilakukan oleh Sutanta (2006), mencoba untuk melakukan sinkronisasi data antar aplikasi. Permasalahan ini dapat diatasi dengan mengembangkan aplikasi *web service* yang memungkinkan untuk melakukan proses pertukaran data untuk sinkronisasi data antar sistem yang telah ada. Sedangkan terkait dengan pengembangan

eGovernment (e-Gov) dalam bentuk aplikasi *website*, saat ini setidaknya telah dikembangkan sebanyak 33 aplikasi yang sebagian besar dapat diakses dari portal webPembkab, dan sebagian yang lain merupakan sub domain dalam portal webPembkab [9].

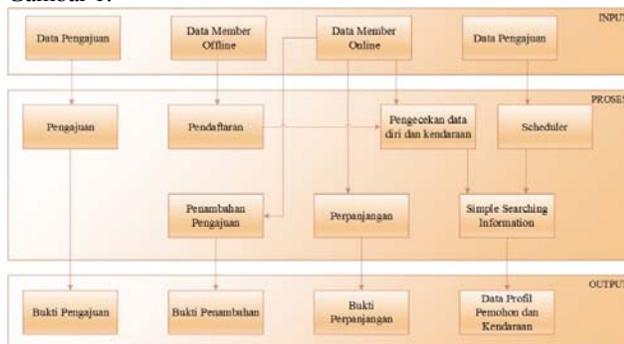
Pada penelitian yang dilakukan oleh Satoto (2014), mencoba untuk menjembatani kebutuhan data kepegawaian dari yang telah dikelola oleh unit sistem informasi tingkat universitas untuk dapat dipergunakan oleh sistem-sistem lain yang akan dikembangkan di tingkat fakultas, yang selanjutnya sistem-sistem ini disebut dengan sub sistem, dengan mengembangkan *web service* [10].

Berdasarkan kedua penelitian terdahulu, dapat disimpulkan bahwa *web service* yang digunakan sebagai sinkronisasi data serta menjembatani beberapa sub sistem yang ada dalam sistem informasi tingkat universitas. Sedangkan pada penelitian ini, yang dilakukan adalah menggunakan *web service* sebagai salah satu *backup database* serta menghubungkan antara 3 aplikasi yang berbeda platform. Dikarenakan aplikasi android tidak dapat berjalan jika tidak menggunakan format JSON dalam berkomunikasi dengan platform lainnya.

II. METODE PENELITIAN

A. Model Konseptual

Model konseptual adalah konsep pemikiran yang membantu peneliti untuk merumuskan pemecahan masalah dan membantu dalam merumuskan solusi permasalahan yang ada. Model konseptual dari penelitian akan diGambarkan pada Gambar 1.



Gambar 9 Model konseptual

Model konseptual pada penelitian ini terbagi menjadi 3 bagian utama, yaitu input, proses, dan output. Dimana input terdiri dari data pengajuan, data member offline, data member online, serta data intranet. Yang kemudian data-data tersebut dilakukan proses untuk dapat melakukan pengajuan, pendaftaran, penambahan pengajuan, perpanjangan, pengecekan data profil pemohon dan kendaraan, *scheduller*, serta *simple searching information*. Serta menghasilkan beberapa output, diantaranya yaitu bukti pengajuan, bukti penambahan, bukti perpanjangan, serta data profil pemohon dan kendaraan. Adapun penggunaan *web service*, terdapat pada proses *scheduller*.

Dimana *scheduller* merupakan proses integrasi dari database sistem intranet dengan database sistem internet. Sehingga database pada setiap sistem akan terupdate dan memiliki isi data yang sama. Selain itu, pihak aplikasi mobile dapat mengakses database internet, dimana data yang dibutuhkan oleh aplikasi mobile terdapat pada database intranet.

B. Sistematika Penelitian

Sistematika penelitian yang harus dilakukan untuk menyelesaikan penelitian. Sistematika penelitian ini, disesuaikan dengan metode penelitian yang digunakan, yaitu metode *extreme programming*. Sistematika dari penelitian ini diGambarkan pada Gambar 2.

Dimana terbagi menjadi 3 tahap utama, yaitu tahap identifikasi, tahap pengembangan sistem, serta tahap kesimpulan dan saran. Tahap identifikasi merupakan tahap yang dilakukan saat perumusan, sedangkan tahap pengembangan sistem merupakan tahapan yang disesuaikan dengan metode *extreme programming*.

Tahap pengembangan sistem terbagi menjadi beberapa tahap diantaranya yaitu tahap *exploration* (mengumpulkan seluruh kebutuhan dan keinginan *user* terhadap pada aplikasi yang akan dibuat), tahap *planning* (menentukan prioritas terhadap *requirement user*, perhitungan estimasi *effort* pada tiap *requirement*, pengaturan jadwal pembuatan aplikasi berdasarkan perhitungan *effort* sebelumnya, dan perencanaan proses iterasi), tahap *iteration to release (prototype)* akan dihasilkan adalah rancangan *web service* yang disesuaikan dengan proses – proses bisnis yang ada, berikut rencana proses iterasi), tahap *productionizing* (aplikasi dirilis untuk versi pertama dengan melakukan *customer approval* untuk disesuaikan dengan *requirement user*), tahap *maintenance* (aplikasi versi pertama diperbaiki dan dirilis kembali dengan versi baru setelah mengevaluasi *feedback* yang diberikan pelanggan terhadap aplikasi versi sebelumnya), dan tahap *death* (aplikasi versi final telah dirilis dan bisa digunakan oleh masyarakat umum).

Namun pada penelitian ini, tidak semua tahap dilakukan. Tahap yang dilakukan diantaranya yaitu tahap *exploration*, *planning*, *iteration to release*, *productionizing*, hingga *testing*. Hal ini dikarenakan untuk sampai ke tahap *maintenance* membutuhkan waktu yang sangat lama.

III. HASIL DAN PEMBAHASAN

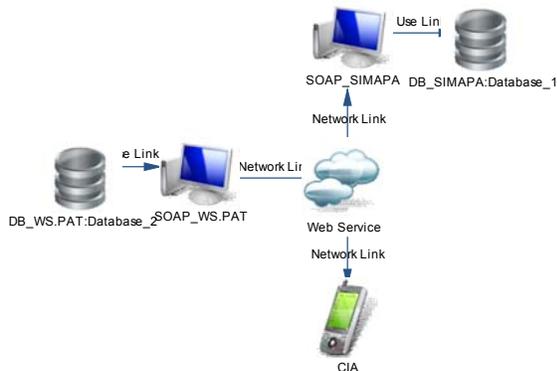
A. Exploration Phase

Pada tahap *exploration*, tahap ini mengumpulkan seluruh kebutuhan dan keinginan *user* terhadap pada aplikasi yang akan dibuat. *Output* yang akan dikeluarkan pada tahap ini adalah rancangan arsitektur sistem, *requirement* dari *user* dan proses bisnis yang akan diterapkan pada *web service* sistem informasi perizinan angkutan umum.

Perancangan model arsitektur sistem yang menerapkan *web service* sebagai media integrasi antar beberapa aplikasi sistem informasi perizinan angkutan umum, diantaranya aplikasi SIMAPA (aplikasi intranet untuk staff dinas perhubungan), aplikasi WS.PAT (aplikasi internet untuk pihak

pemohon yang akan melakukan perizinan online), serta aplikasi CIA (aplikasi *mobile* untuk pihak petugas dishub dalam melakukan pengawasan saat pemeriksaan angkutan umum di lapangan). Dengan memanfaatkan teknologi ini, manajemen data akan lebih terkontrol dan terpusat. Sistem ini akan mensimulasikan interaksi antara entitas yang ada.

Dalam perancangan dan pembuatan sistem ini, akan dimungkinkan penggabungan data dari DB_WS.PAT ke DB_SIMAPA berdasarkan waktu *peng-update-an*. Sehingga data pemohon yang melakukan pengajuan online, dapat dilakukan proses untuk ke tahap selanjutnya sesuai dengan alur, jika melakukan pendaftaran secara offline. Begitupun jika pemohon via offline ingin mengakses WS.PAT, maka hanya tinggal melakukan pendaftaran, dengan data pengajuan yang disesuaikan dengan data yang ada pada offline. Selanjutnya CIA dapat mengakses database yang dimiliki oleh WS.PAT yang dimana database WS.PAT telah disesuaikan dengan database SIMAPA. Pada Gambar 2 akan diGambarkan mengenai arsitektur *web service* sistem informasi perizinan angkutan umum yang akan dibangun.



Gambar 10 Perancangan arsitektur *web service*

TABEL I
IDENTIFIKASI KEBUTUHAN SISTEM

Req ID	Nama Kebutuhan	Deskripsi	User
Req -01	Bukti Pengajuan	Merupakan bukti hasil pengajuan online yang harus dicetak.	Pemohon
Req -02	Bukti Penambahan	Merupakan bukti hasil penambahan pengajuan online yang harus dicetak.	Member
Req -03	Bukti Perpanjangan	Merupakan bukti hasil perpanjangan online yang harus dicetak.	Member
Req -04	Data Profil Pemohon dan Kendaraan	Menampilkan data profil pemohon yang terdiri dari nama, alamat, no telepon, dan email. Serta data kendaraan yang diajukan perizinan atas pemohon tersebut.	Member
Req -05	Database internet terupdate	Merupakan hal yang dilakukan oleh sistem dalam melakukan pengupdate-an data yang berasal dari database intranet dan database internet, sehingga database dapat terbackup dengan baik.	Admin

Requirement yang dibutuhkan didapat dari hasil studi lapangan ke Dinas Perhubungan Kabupaten Bandung Barat. Sehingga dapat disimpulkan, bahwa *requirement* yang dibutuhkan diantaranya, yaitu bukti pengajuan, bukti penambahan, bukti perpanjangan, serta data profil dan data

kendaraan. *Requirement* yang dibutuhkan akan dijelaskan pada Tabel I.

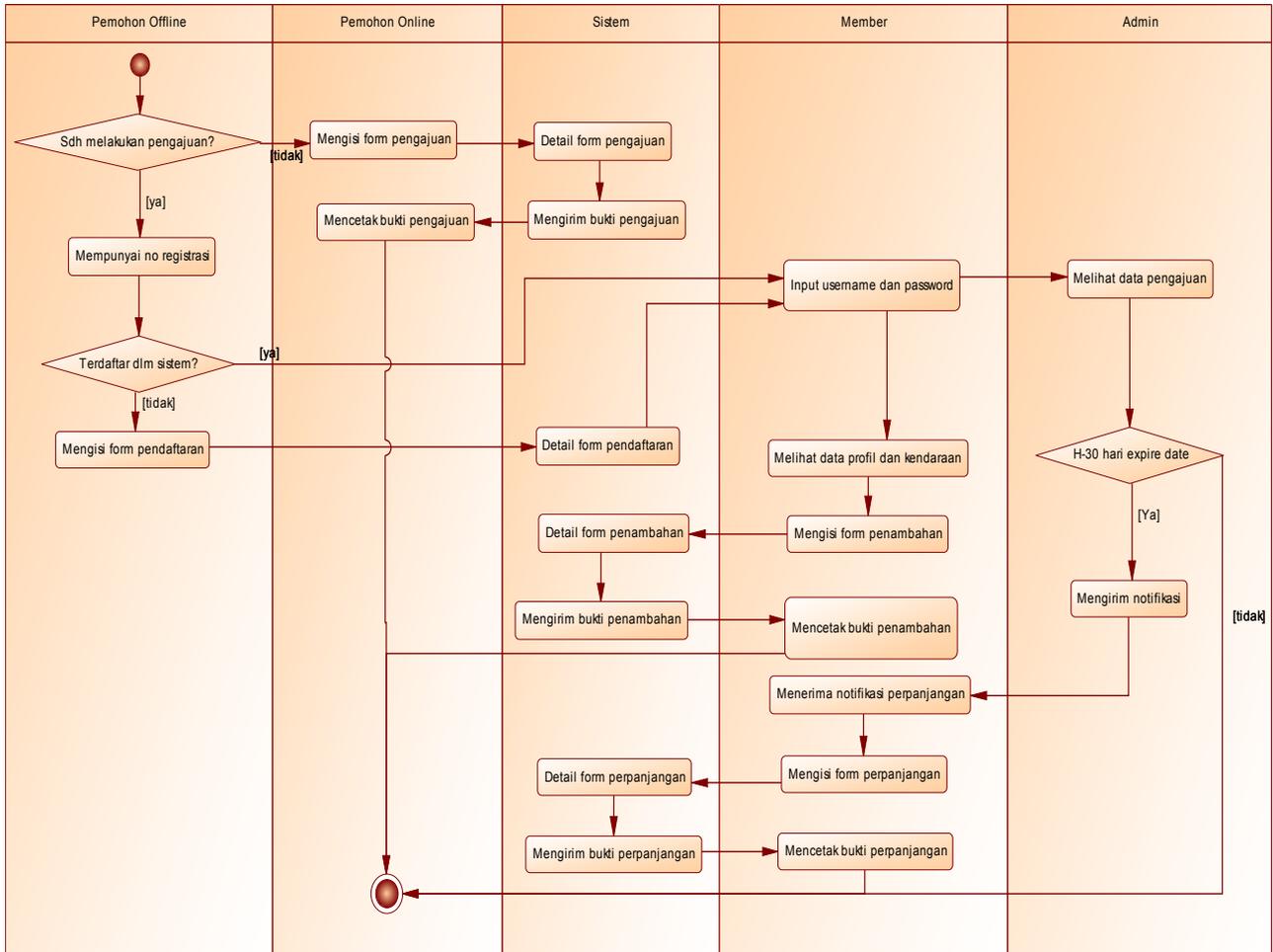
Proses bisnis mengGambarkan keterkaitan objek dengan alur kerja. Berikut terdapat 2 proses bisnis, yaitu proses bisnis web service yang mengGambarkan proses integrasi database terhadap 3 aplikasi, yaitu aplikasi SIMAPA, aplikasi WS.PAT, serta aplikasi CIA. Kemudian terdapat proses bisnis WS.PAT sebagai *website* sistem informasi perizinan online. Dimana user yang berperan adalah pemohon online, pemohon offline, member, sistem, serta admin. Dimana pemohon merupakan salah satu objek yang belum mempunyai username, sehingga diharuskan untuk mengajukan perizinan. Sedangkan member merupakan salah satu objek yang telah mempunyai username, dapat mengakses beberapa fitur yang ada pada aplikasi ini. Sedangkan sistem merupakan salah satu objek yang mengGambarkan kerja dari sistem ini. Adapun admin merupakan salah satu objek yang melakukan pemberian notifikasi jika sudah mendekati *expire-date*. Proses bisnis diGambarkan pada Gambar 3.

B. Planning Phase

Kebutuhan awal *user* atau biasa disebut *user story*, ditulis pada fase ini untuk menjelaskan rincian perkiraan awal untuk mengidentifikasi proses pengembangan dan faktor resiko yang mungkin muncul. *User story* umumnya ditulis pada *index card*. Setelah *user story* dikembangkan, tim pembuat aplikasi melakukan pertemuan formal dengan pengembang untuk mengidentifikasi prioritas dari *user*. Selanjutnya setiap *user story* akan diberi bobot dan nilai prioritas berdasarkan tiga tingkatan, yaitu sangat penting (bernilai 1), penting (bernilai 2), dan kurang penting (bernilai 3). Berikut ini akan dijelaskan mengenai *user stories* pada Tabel II.

TABEL II
USER STORY (INDEX CARD)

Req ID	Nama Kebutuhan	Perkiraan Awal	Kepentingan
Req -01	Bukti Pengajuan	Merupakan bukti hasil pengajuan online, dimana bukti ini harus dibawa ke kantor Dinas Perhubungan Kabupaten Bandung Barat untuk melanjutkan ke proses selanjutnya.	1
Req -02	Bukti Penambahan	Merupakan bukti hasil penambahan pengajuan online, dimana bukti ini harus dibawa sama seperti dengan bukti pengajuan.	1
Req -03	Bukti Perpanjangan	Merupakan bukti hasil perpanjangan online, dimana bukti ini harus dibawa sama seperti dengan bukti pengajuan	1
Req -04	Data Profil Pemohon dan Kendaraan	Menampilkan data profil pemohon yang terdiri dari nama, alamat, no telepon, email, serta data kendaraan yang diajukan perizinan atas pemohon tersebut.	1
Req -05	Database internet terupdate	Merupakan hal yang dilakukan oleh sistem dalam melakukan pengupdate-an data yang berasal dari database intranet dan database internet, sehingga database dapat terbackup dengan baik	1



Gambar 11 Proses bisnis WS.PAT

C. Iteration to Release Phase

Pada tahap *iteration to release, prototype* akan dihasilkan dengan melakukan proses *analysis, design, coding,* dan *testing*. Output yang akan dikeluarkan pada tahap ini adalah rancangan *web service* yang disesuaikan dengan proses – proses bisnis yang ada.

1. Use Case Diagram

Diagram *use case* merupakan pemodelan untuk *behavior* sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Selain itu, *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. *Use case diagram* digambarkan pada Gambar 9. Berdasarkan Gambar 9 dapat diketahui bahwa yang menjadi user, diantaranya yaitu pemohon online, admin, member, serta pemohon *offline*.

2. Class Diagram

Class diagram dibuat untuk menggambarkan relasi antar *class* pada sistem yang akan dibangun. *Class diagram* yang digunakan pada aplikasi *web service* sistem informasi perizinan angkutan umum. Terdapat 4 *class model* yaitu *admin_model*, *pengajuan_model*, *frontend_model*, serta *time_scheduler*. Adapun *class model* mempunyai fungsi fungsi tertentu yang digunakan pada 3 *class* aktor yaitu pemohon, member, dan admin berdasarkan *use case* yang telah dijelaskan sebelumnya. *Class diagram* akan digambarkan pada Gambar 10.

3. Entity Relationship Diagram

Setelah melakukan perancangan *use case diagram* serta *class diagram*, selanjutnya dilakukan perancangan database. Perancangan database ini digunakan sebagai acuan untuk melakukan proses *coding*. Di dalam *entity relationship diagram*, dapat dilihat bahwa suatu entitas memiliki atribut apa saja serta hubungan antar entitas tersebut. Pada penelitian ini, digunakan salah satu konsep dalam merancang *entity relation diagram*, yaitu dengan menggunakan *Physical Data*

Model (PDM). PDM pada penelitian ini, akan diGambarkan pada Gambar 12.

D. Productioning Phase

Tahap terakhir dalam penelitian ini adalah tahap *productionizing*. Tahap ini dilakukan setelah tahap *iteration to release* selesai. Pada tahap ini, peneliti akan mengimplementasi aplikasi ke dalam kondisi yang sebenarnya.

1. Deployment Diagram

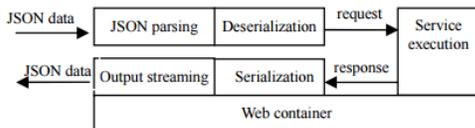
Deployment diagram merupakan diagram yang mengGambarkan arsitektur fisik dari dari *hardware* dan *software* yang terdapat pada sistem. Aplikasi ini terdiri dari sistem WS.PAT, SIMAPA, dan CIA dalam penerapannya. Pada WS.PAT terdapat database DB_WS.PT, *website* WS.PAT, dan SOAP_WS.PAT. Sedangkan SIMAPA terdapat database DB_SIMAPA, *website* SIMAPA, dan SOAP_SIMAPA. Adapun pada CIA terdapat database DB_CIA, aplikasi_CIA, dan SOAP_CIA.

Dimana SOAP antara sistem WS.PAT dan SIMAPA, akan saling berinteraksi sebagai *request* dan *request handler*.

Sedangkan SOAP_WS.PAT akan berinteraksi dengan CIA untuk memberikan data bagi CIA sesuai dengan data yang dibutuhkan atau yang di-*request*. *Deployment diagram* akan diGambarkan pada Gambar 11.

2. Implementasi JSON pada web service SOAP

Berdasarkan Gambar 4, dapat terlihat bahwa untuk mengurai data JSON berawal dengan diterimanya data request dari sistem intranet (SIMAPA) maupun dari sistem mobile (CIA) kemudian data diurai secara *deserialization* menjadi data aplikasi. Kemudian server akan mengambil layanan web untuk mendapatkan hasil. Hasilnya dilakukan *serialization* terhadap hasil yang didapat dari data PHP menjadi data JSON ke dalam output HTTP, kemudian mengirimkannya kembali ke SIMAPA maupun CIA.



Gambar 12 Implementasi JSON pada *web service*

Berikut merupakan XML *web service* yang digunakan dalam melakukan pengupdate-an database serta melakukan response data yang diminta oleh pihak aplikasi android. Pada Gambar 5 (*request*) dan Gambar 6 (*response*) merupakan bahasa markup JSON pada pesan SOAP yang digunakan saat melakukan pengupdate-an database. Sedangkan pada Gambar 7 (*request*) dan Gambar 8 (*response*) merupakan bahasa markup JSON pada pesan SOAP yang digunakan saat melakukan permintaan untuk mengakses data surat perizinan.

```

<?php
include "koneksi.php";
include "model.php";

$mod = new Model();
$private_key=md5("bakkekak");

if(isset($_GET['itrayek'])){
    $json_url="http://localhost/wspatAPI/jsonitrayek.php?key=".$private_key;
    $ch=curl_init($json_url);

    $json_string="";
    $options = array(
        CURLOPT_RETURNTRANSFER => true,
        CURLOPT_HTTPHEADER => array('Content-type: application/json'),
        CURLOPT_POSTFIELDS => $json_string
    );
    curl_setopt_array($ch, $options);

    $sqltrayek= $mod->getIzinTrayek();
    $field= mysqli_fetch_fields($sqltrayek);
    $idx=0;
    foreach($field as $sar){
        $fields[$idx]=$sar->name;
        $idx++;
    }
    $result = curl_exec($ch);
    $sdecode = json_decode($result, true);
    $stot=0;
    if(!empty($sdecode)){
        foreach($sdecode as $srow){
            for($si=0;$si<count($fields);$si++){
                $sdatalitrayek[$fields[$si]]=$srow[$fields[$si]];
            }
            $status=$mod->cekIzinTrayek($sdatalitrayek);
            echo $status." Pemilik dengan no kendaraan ".$sdatalitrayek['NO_KENDARAAN']. "<br />
                Perizinan dengan ID Izin ".$sdatalitrayek['ID_IZIN_TRAYEK']. "<br />";
            $stot++;
        }
    }
    echo "Total terjadi eksekusi ".$stot." record ";
}
  
```

Gambar 13 Pesan *Request* SOAP pengupdate-an database izin trayek yang dilakukan oleh pihak admin sistem informasi perizinan berbasis intranet

```

<?php
include "koneksi.php";

$sqlitrayek="SELECT * FROM 'izin_trayek'
WHERE 'UPDATE_IT'>=DATE_ADD(CURRENT_DATE(),
INTERVAL -1 DAY)";
$sqltrayek=mysqli_query($konek,$sqlitrayek);

$sdatalitrayek="";
$idx=0;
$field= mysqli_fetch_fields($sqltrayek);

foreach($field as $sar){
    $fields[$idx]=$sar->name;
    $idx++;
}
$idx=0;
while($r=mysqli_fetch_assoc($sqltrayek)){
    for($si=0;$si<count($fields);$si++){
        $sdatalitrayek[$fields[$si]]=$r[$fields[$si]];
    }
    $idx++;
}
$jsonitrayek=json_encode($sdatalitrayek);
echo ($jsonitrayek);
}
  
```

Gambar 14 Pesan *Response* SOAP pengupdate-an database izin trayek yang dilakukan oleh pihak admin *web service*

```

<?php
mysql_connect("localhost","petugas","lugasistub") or die(mysql_error());
mysql_select_db("webservice") or die(mysql_error());

$no = $_REQUEST['no'];

$sql = mysql_query("SELECT NO_USI_ID, NO_USI_IT, surat_perizinan.ID_IZIN_TRAYEK AS JENIS_IZIN, NAMA_PEMILIK, " .
    "TAMOGAL_AKHIR, BERLAKU AS PASA, BERLAKU, STATUS_IZIN FROM kartu_pengawasan " .
    "LEFT JOIN surat_perizinan ON (kartu_pengawasan.ID_SURAT = surat_perizinan.ID_SURAT) " .
    "LEFT JOIN izin_operasi ON (surat_perizinan.ID_IZIN_OPERASI = izin_operasi.ID_IZIN_OPERASI) " .
    "LEFT JOIN izin_trayek ON (surat_perizinan.ID_IZIN_TRAYEK = izin_trayek.ID_IZIN_TRAYEK) " .
    "LEFT JOIN pemilik ON (izin_trayek.KTP_PEMILIK = pemilik.KTP_PEMILIK) " .
    "OR (izin_operasi.KTP_PEMILIK = pemilik.KTP_PEMILIK) " .
    "WHERE NO_PENGAWASAN=$no") or die(mysql_error());

while($so = mysql_fetch_assoc($sql)) {
    if($so['JENIS_IZIN'] == "") {
        $so['JENIS_IZIN'] = "OPERASI";
        $so['NO_USI_ID'] = $so['NO_USI_ID'];
    } else {
        $so['JENIS_IZIN'] = "TRAYEK";
        $so['NO_USI_ID'] = $so['NO_USI_ID'];
    }
    $soutput[] = array_slice($so,2);
}

if(mysql_num_rows($so) == 0) {
    $n["NO_USI_ID"] = "0";
    $n["JENIS_IZIN"] = "0";
    $n["NAMA_PEMILIK"] = "0";
    $n["PASA_BERLAKU"] = "0";
    $n["STATUS_IZIN"] = "0";
    $soutput[] = $n;
}

print(json_encode(array("Info"=>$soutput)));
mysql_close();
}
  
```

Gambar 15 Pesan *Request* SOAP permintaan untuk mengakses data surat perizinan yang dilakukan oleh admin aplikasi android

```

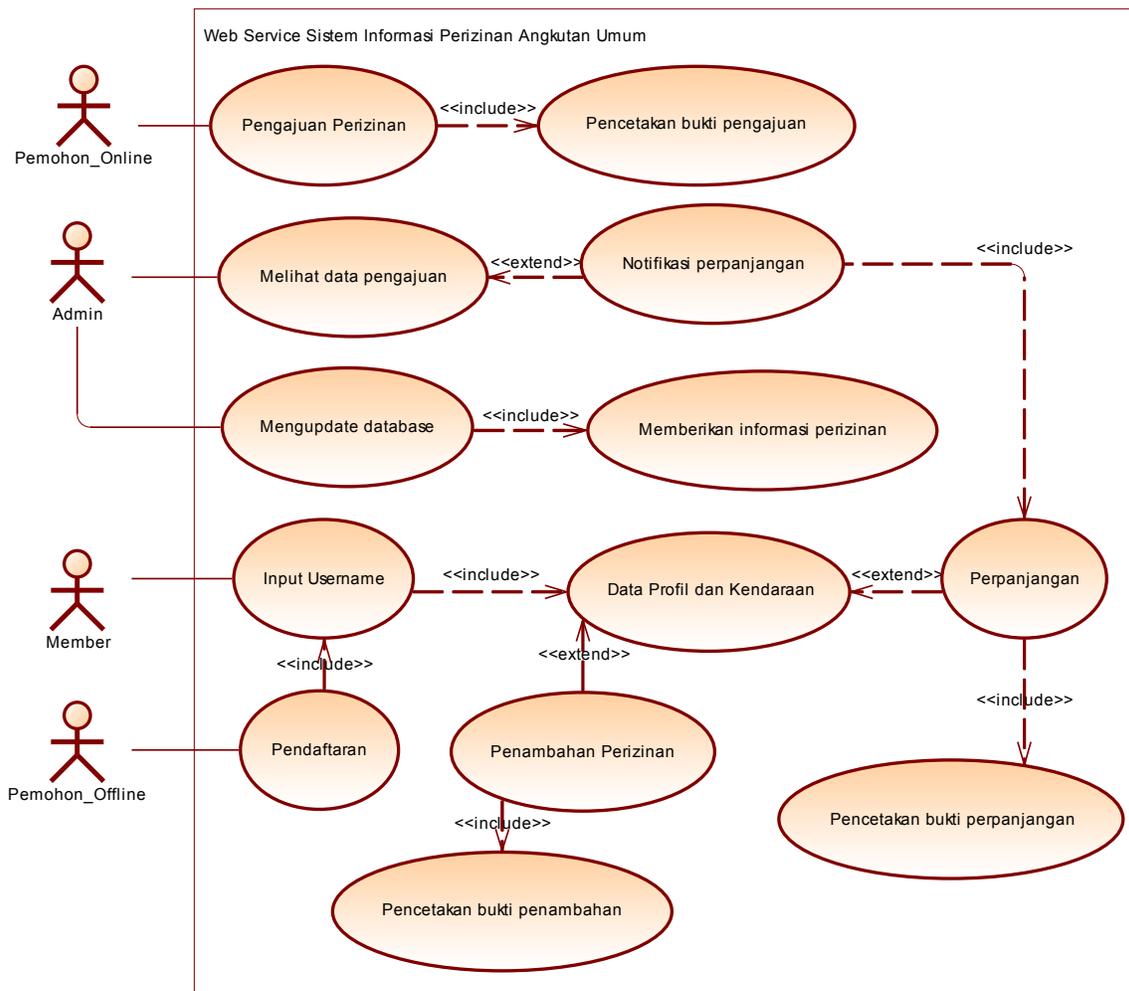
<?php
include "koneksi.php";

$sqlsurat="SELECT * FROM 'surat_perizinan' WHERE 'UPDATE_SP'>=DATE_ADD(CURRENT_DATE(), INTERVAL -1 DAY
$asurat=mysqli_query($koneksi,$sqlsurat);
//echo $asurat;

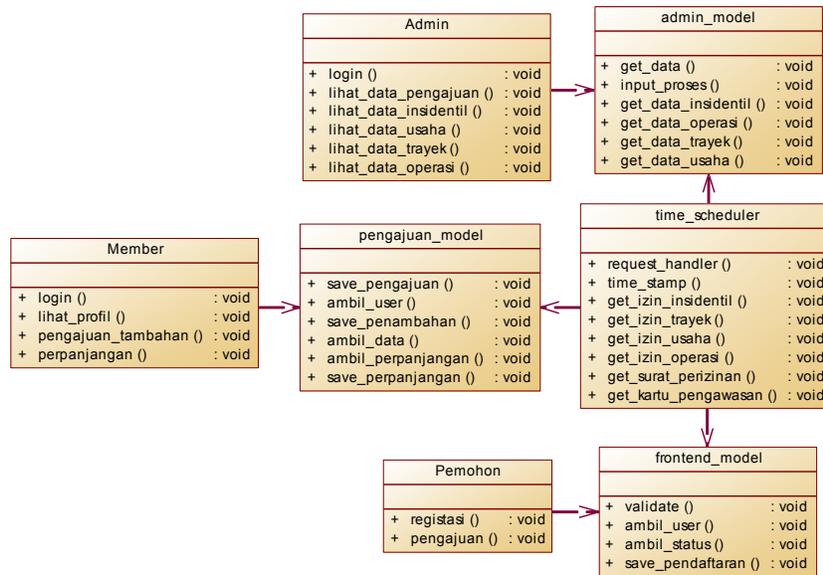
//$jumlahTrayek=mysqli_num_rows($asurat);
$dataSurat="";
$idx=0;
$fields=mysqli_fetch_fields($asurat);
foreach($field as $ar){
    $fields[$idx]=$ar->name;
    $idx++;
}
//print_r($fields);
$idx=0;
while($r=mysqli_fetch_assoc($asurat)){
    for($i=0;$i<count($fields);$i++){
        $dataSurat[$idx][$fields[$i]]=$r[$fields[$i]];
    }
    $idx++;
}
$jsonSurat=json_encode($dataSurat);
echo ($jsonSurat);
}

```

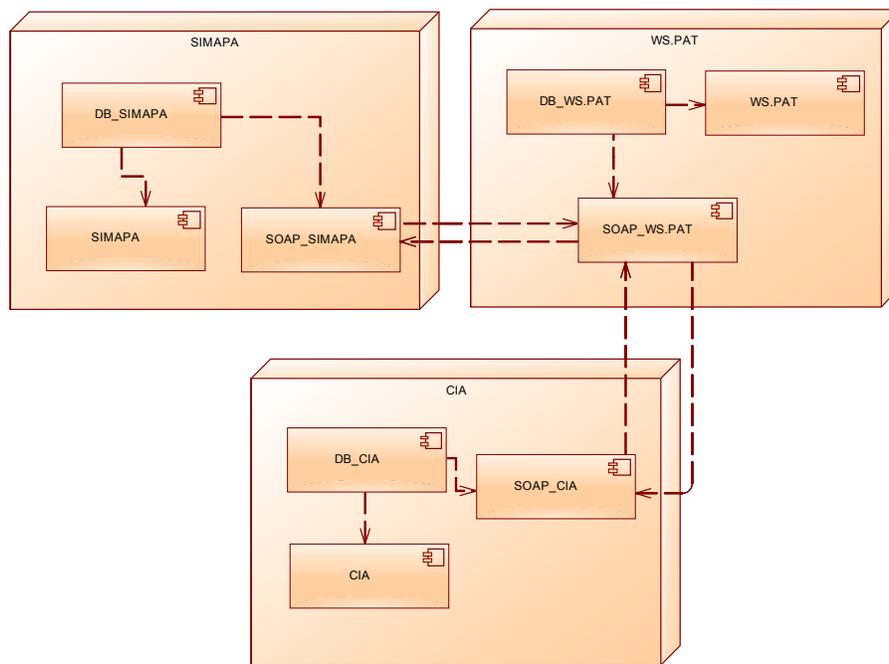
Gambar 16 Pesan *Response SOAP* permintaan untuk mengakses data surat yang dilakukan oleh admin *web service*



Gambar 17 *Use case diagram*



Gambar 18 Class diagram



Gambar 19 Deployment diagram

E. Testing

Testing merupakan proses untuk menguji apakah semua fungsionalitas yang ada pada aplikasi sudah berjalan sesuai dengan yang diinginkan. Pengujian yang dilakukan dengan menggunakan metode *Blackbox Testing*. Proses pengujian dilakukan oleh user.

Usability adalah suatu atribut untuk menilai seberapa mudah interface website digunakan. *Usability* juga sering digunakan untuk meningkatkan kemudahan pengguna selama proses desain. Pengujian dilakukan oleh 30 responden yang telah mencoba aplikasi WS.PT ini, dengan 2 kali iterasi

bertahap menurut metode *extreme programming*. Adapun hasil *usability testing* dapat dilihat pada tabel IV.

Hasil dari *usability testing*, dapat terlihat bahwa aplikasi WS.PAT sudah berjalan dengan baik, mudah dimengerti, memberikan desain yang mempermudah dalam melakukan aktifitas, mudah diingat dalam penggunaan pada waktu selanjutnya, eror yang terjadi tidak mengganggu ketika menggunakan aplikasi, serta tampilan sudah sesuai dan memberikan kepuasan tersendiri.

TABEL III
HASIL USABILITY TESTING

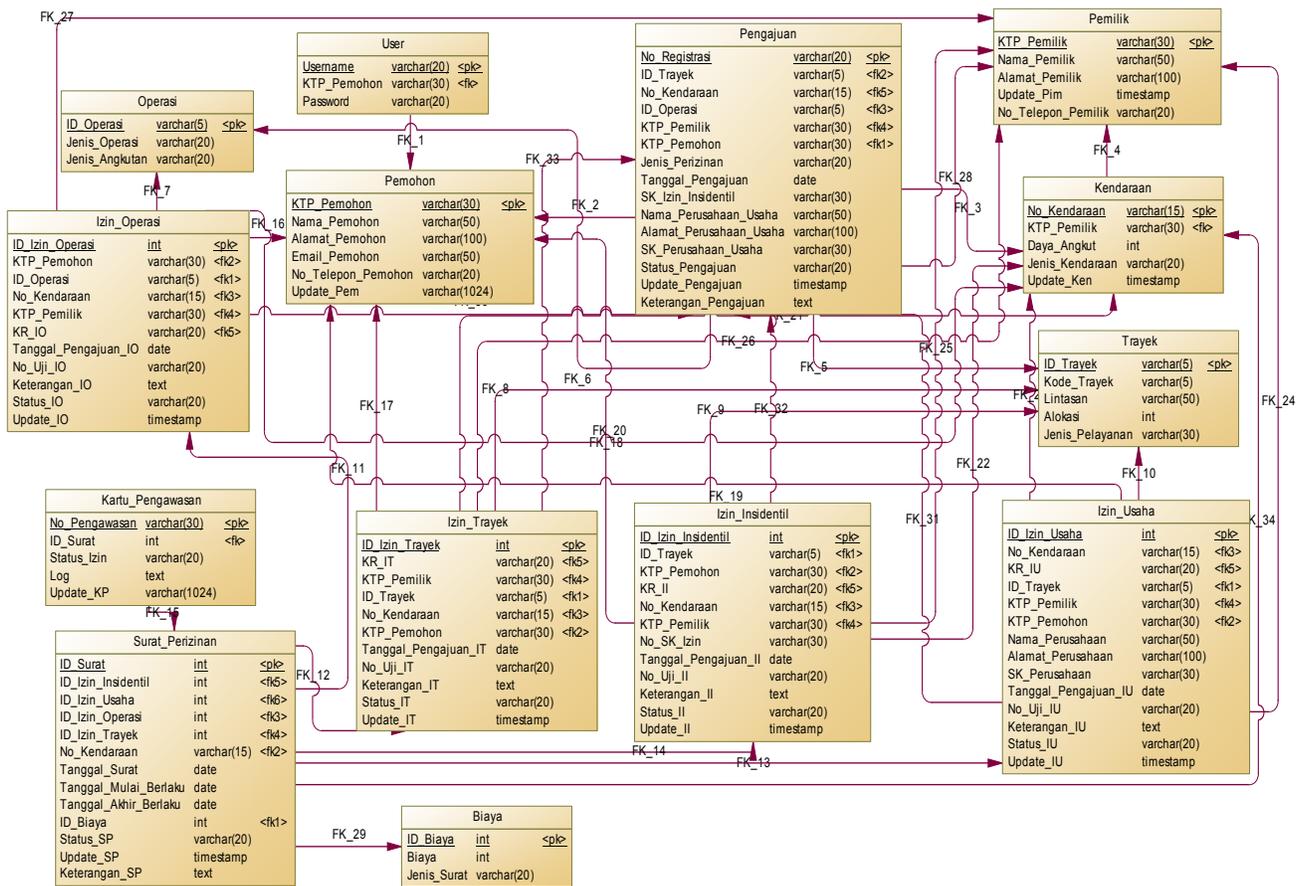
Nama Dimensi	Pertanyaan	Jawaban (responden)	
		Ya	Tidak
Learnability	Apakah menu dan fungsi pada aplikasi WS.PAT dapat dimengerti dengan baik?	93%	7%
Efficiency	Apakah dengan desain seperti itu, memberikan kemudahan pada aktifitasnya?	90%	10%
Memorability	Apakah aplikasi WS.PAT mudah untuk diingat dalam penggunaan selanjutnya?	93%	7%
Errors	Apakah kesalahan sistem sering mengganggu ketika menggunakan aplikasi WS.PAT?	10%	90%
Satisfaction	Apakah tampilan sudah sesuai dan menyenangkan bagi pengguna?	93%	7%

IV. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, dapat diambil kesimpulan yang disesuaikan dengan batasan masalah

yang telah dirancang sebelumnya, diantaranya yaitu terbentuknya rancangan *web service* yang dapat digunakan pada sistem informasi perizinan angkutan umum, yaitu menggunakan protokol SOAP dengan bahasa *markup* JSON. Hal ini dikarenakan disesuaikan dengan kebutuhan yang ingin didapatkan setelah integrasi aplikasi, yaitu sebagai *backup database* serta penggabungan data antara sistem intranet dengan sistem internet. Selain itu, aplikasi *mobile* hanya bisa mengakses database dari aplikasi WS.PAT. Hal ini dikarenakan WS.PAT merupakan aplikasi yang menggunakan jaringan internet, sehingga dapat diakses dimana dan kapan saja.

Terbentuknya *website* yang dapat digunakan oleh pihak eksternal (pemohon) dalam melakukan pengajuan online, mengetahui status terhadap pengajuan yang telah dilakukan sebelumnya, mendapat notifikasi jika surat sudah mendekati *expire-date*. Sehingga dapat dilakukan pengajuan perpanjangan secepatnya. Selain itu juga, bagi pemohon yang melakukan pengajuan offline, dapat melakukan pendaftaran untuk mendapatkan *username* dan *password*, sehingga dapat login ke dalam aplikasi WS.PAT.



Gambar 20 Physical data model (PDM)

DAFTAR PUSTAKA

- [1] <http://www.w3.org/TR/wsSarch/>, diakses Juni 2015.
- [2] S. Potss and M. Kopack, *Sams Teach Yourself Web Services in 24 Hours*, Indiana: Sams Publishing, 2003.
- [3] H. Deviana, "Penerapan XML Web service Pada Sistem Distribusi Barang," *Jurnal Generic*, 2011.
- [4] Rosenberg, Doug, *Modeling Service-Oriented Architectures*, Sparx Systems Pty Ltd, 2010.
- [5] New Comer, E., *Understanding Web Service: XML, WSDL, SOAP, and UDDI*, Independent Technology Guide, 2002.
- [6] <http://www.json.org/>, diakses Juni 2015.
- [7] Peraturan Daerah Kabupaten Bandung Barat Nomor 5 Tahun 2010 Tentang Penyelenggaraan Perhubungan.
- [8] Turban, McLean, & Wetherbe, *Information of Technology Management*, Wiley, 2001.
- [9] Sutanta, Edhy, *Kebutuhan Web Service untuk Sinkronisasi Data Antar Sistem Informasi Dalam E-Govdi Pemkab Bantul Yogyakarta, Jurtik - STMIK Bandung (Edisi Mei)*, 2012.
- [10] Satoto, Kodrat Iman, *Integrasi Data Kepegawaian Aplikasi Sub Sistem di Universitas Diponegoro Melalui Web Service, Jurnal Teknologi Informasi-Aiti, Vol. 11. No.1, Februari, 2014.*