

# SISTEM INFORMASI MANAJEMEN PENGELOLAAN PENDAPATAN ASLI DAERAH (PAD) BERBASIS E-MAIL (STUDI KASUS KABUPATEN TULANG BAWANG – LAMPUNG)

Dwirgo Sahlinal, P. Insap Santoso, Abdul Kadir  
Jurusan Teknik Elektro, Fakultas Teknik, Politeknik Negeri Lampung  
Alamat email : dwirgo\_sahlinal@ymail.com

## ABSTRACT

*The difficulty of mobile operators signal that can be captured in the sub-districts in Tulang Bawang district, makes a problem in the data transmission process payment transactions Land and Building Tax (PBB) from business districts to the Department of Revenue Property and Public Asset (DP2KA), and need testing the type of signal that can be received in each district. This study uses data delivery technique messaging (e-mail) with software support PHP, Javascript and JSON as a form of payment transaction data to be sent. By using PHP, Javascript and JSON in this application, create an application that was developed into a platform-independent code and data both as well as low cost in implementation of e-governement.*

**Keywords :** PBB, DP2KA, JSON, messaging, e-governement

## 1. PENDAHULUAN

Pembangunan e-Government meliputi aspek yang sangat luas mencakup seluruh sistem dan prosedur penyelenggaraan pemerintah. Aplikasi dan layanan e-Government yang berfungsi membantu instansi pemerintah dalam melayani kebutuhan masyarakat, mengatur aset pemerintah, mengorganisasi kegiatan pemerintah secara internal, diantaranya keuangan, kepegawaian dan pelayanan masyarakat dan pelaporan.

Monitoring dan evaluasi penerimaan Pajak Bumi dan Bangunan (PBB) yang dilakukan oleh kepala daerah di Kabupaten Tulang Bawang dilakukan pada triwulan ketiga dan keempat. Prosedur manual dilakukan dengan memanggil Kepala DP2KA untuk memberikan laporan realisasi penerimaan PBB disemua kecamatan di Kabupaten Tulang Bawang. Cara manual ini, rawan terhadap manipulasi data, yang terkesan terlalu memaksakan data pelaporan telah mencapai target penerimaan yang diharapkan. Selain itu, data pelaporan PBB dari kecamatan tidak dapat langsung di *sharing* ke kepala daerah sebagai bahan evaluasi kinerja camat dalam mencapai target penerimaan PBB.

Pembangunan awal infrastruktur jaringan sebagai media transfer data memerlukan biaya yang besar. Selain itu,

untuk jangka waktu yang panjang diperlukan juga biaya perawatan yang besar. Sulitnya sinyal operator seluler yang dapat ditangkap di kecamatan-kecamatan yang ada di Kabupaten Tulang Bawang, menjadikan suatu permasalahan dalam proses pengiriman data transaksi pembayaran PBB dari kecamatan ke DP2KA. Perlu dilakukan pengujian jenis sinyal yang dapat diterima di tiap-tiap kecamatan serta bagaimana menentukan model pengiriman data dengan memanfaatkan keterbatasan sinyal *GSM* yang ada di kecamatan-kecamatan Kabupaten Tulang Bawang dan menentukan format data yang akan dikirim dengan kapasitas kecil.

Menurut Ahmad [5], JSON (*JavaScript Object Notation*) merupakan format pertukaran data berdasarkan notasi JavaScript. Dari beberapa sumber mengatakan bahwa JSON lebih baik dari XML sebagai format pertukaran data. Ahmad mencoba untuk membuat uji untuk membandingkan kedua format data tersebut pada ASP.NET Web Service. Parameter pembandingan yang digunakan yaitu besar stream yang ditransmisikan dan *response time*. Untuk menganalisis kinerja keduanya digunakan *tool* buatan Nikhil Kotari yaitu *Web Development Helper*. Pertama dibuat sebuah *object Employee* dengan data *EmpId, Name, Sex* dan *Title*. *Object* ini yang akan diserialisasi ke *JSON* dan *XML* sebagai

*return value WebService*. Dari hasil uji tersebut, format XML memerlukan 226 byte sedangkan JSON hanya 132 byte. Ini jauh lebih kecil (kurang dari setengah) dibandingkan XML. Dari sisi *response time*, pada *invoke* yang pertama XML membutuhkan waktu 0:0:3430 sedangkan JSON 0:0:3280 dan *invoke* berikutnya untuk XML 0:0:0150 sedangkan JSON 0:0:0000. Dari data ini membuktikan bahwa JSON memang lebih ringan dibandingkan XML baik dari sisi ukuran *stream* yang ditransmisikan lebih kecil maupun waktu proses di server juga lebih singkat.

Majid Khosravi [9], melakukan penelitian dengan mengkoneksikan *device mobile android* ke *web service REST* untuk mengukur *delay time* yang terjadi pada XML dan JSON dari segi jumlah objek data, ukuran data, *loading time*, *parsing time* dengan menyediakan operasi *select*, *update*, *insert* dan *delete*. Dalam penelitian tersebut, JSON mempunyai *respon time* yang lebih cepat jika dibandingkan dengan XML ketika jumlah objek data kecil (dibawah 1300).

Rutecki[4], melakukan penelitian merancang dan mengimplementasikan *Data Acquisition Module (DAM)* pada suatu jaringan secara riil, dengan beberapa kriteria rancangan DAM sebagai berikut :

- a) jenis protokol yang digunakan pada transfer data, yaitu GPRS dan ethernet;
- b) metode enkapsulasi data, dengan menggunakan JSON;
- c) merancang perangkat lunak akuisisi data;
- d) menggunakan basis data mysql.

Hasil penelitian tersebut, menggambarkan modul DAM bertanggungjawab dalam menerima data yang dikumpulkan oleh konsentrator dan selanjutnya mengoreksi secara benar untuk disisipkan pada basis data.

Penelitian ini menggunakan teknik pengiriman data messaging (*e-mail*) dengan perangkat lunak pendukung PHP, Javascript dan JSON sebagai bentuk data yang akan dikirim. PHP sebagai bahasa yang bebas platform dan berorientasi obyek, sedangkan JSON sebagai platform bagi pengiriman data terdistribusi.

## 2. TINJAUAN PUSTAKA

### 2.1 Konsep Pengembangan e-Government

Konsep pengembangan *e-Government* (Kominfo, 2003) di setiap lembaga pemerintah ditentukan oleh:

- a. tugas pokok dan fungsi dari setiap lembaga;
- b. jenis informasi sumberdaya;
- c. jenis layanan yang diberikan oleh masing-masing lembaga.

Hal ini menentukan struktur data dan proses bisnis yang menjadi dasar penyusunan rencana induk *e-Government* di setiap lembaga pemerintah. Jenis layanan yang diberikan dan jenis informasi yang dibutuhkan menentukan pola dan prioritas pengembangan *e-Government* suatu lembaga. Pola tersebut dapat menyangkut hubungan *Government to Government (G2G)*, *Government to Business (G2B)*, dan *Government to Citizen (G2C)*.

Secara sederhana, Heeks (1999) dalam [14] mendefinisikan *e-Government* sebagai berikut : “Kegiatan yang dilakukan oleh pemerintah dengan menggunakan Teknologi Informasi (TI) untuk memberikan layanan kepada masyarakat”. Dari definisi tersebut, dapat kita lihat bahwa tujuan utama *E-Government* adalah untuk meningkatkan efisiensi dan kualitas layanan. Menurut Heeks, hampir semua lembaga pemerintahan di dunia ini mengalami ketidakefisienan, terutama di negara yang sedang berkembang.

Adapun menurut *Center for Democracy and Technology* dan *InfoDev* (Hasibuan, [14]), proses implementasi *e-Government* terbagi menjadi 3 (tiga) tahapan, yang tidak bergantung satu sama lain, atau harus dilakukan secara berurutan, tetapi masing-masing menjelaskan mengenai tujuan dari *e-Government*. Tahapan tersebut antara lain sebagai berikut:

- a. Tahap pertama adalah *Publish*, yaitu tahapan yang menggunakan teknologi informasi untuk meluaskan akses untuk informasi pemerintah. Misalnya dengan cara pembuatan situs informasi di setiap lembaga, penyiapan sumber daya manusia, sosialisasi situs informasi baik untuk internal maupun untuk publik, serta penyiapan sarana akses yang mudah.

- b. Tahap kedua, adalah *Interact*, yaitu meluaskan partisipasi masyarakat dalam pemerintahan. Misalnya dengan cara pembuatan situs yang interaktif dengan publik, serta adanya antarmuka yang terhubung dengan lembaga lain.
- c. Tahap ketiga adalah *Transact*, yaitu menyediakan layanan pemerintah secara *online*, misalnya dengan cara pembuatan situs transaksi pelayanan publik, serta interoperabilitas aplikasi maupun data dengan lembaga lain.

Dalam pengembangan *e-Government*, perlu diperhatikan dan disiapkan aspek kepemimpinan (*e-leadership*), aspek kesadaran akan manfaat *e-Government* (*awareness building*), aspek sumber daya manusia, serta peraturan perundangan dan PERDA yang mendukung (*applied regulation*). Seluruh aspek tersebut berperan dalam menentukan desain arsitektur sistem informasi yang akan dibangun (*enterprise architecture*).

## 2.2 Konsep Pengembangan Infrastruktur e-Government

Pengembangan *e-Government* di suatu lembaga pemerintah, dilandasi oleh empat infrastruktur utama, yaitu sebagai berikut (Kominfo, 2006. Versi 1.0) :

- a. Suprastruktur *e-Government* yang memuat antara lain kepemimpinan manajemen lembaga (*e-leadership*), sumberdaya manusia (*human resources*) dan peraturan di tingkat lembaga yang terkait dengan pengembangan *e-Government* (*regulation*).
- b. Infrastruktur jaringan yang memuat antara lain protokol komunikasi, topologi, teknologi dan keamanan, yang lebih lanjut dapat dilihat pada Panduan Pembangunan Infrastruktur Portal Pemerintah.
- c. Infrastruktur informasi yang memuat antara lain struktur data, format data, metoda berbagi data (*data sharing*), dan sistem pengamanannya, yang lebih lanjut dapat dilihat pada Panduan Sistem Manajemen Dokumen Elektronik.
- d. Infrastruktur aplikasi yang memuat antara lain aplikasi layanan publik, aplikasi antar muka (*interface*), dan aplikasi *back office*.

Seluruh infrastruktur tersebut akan dibangun dalam satu kerangka berpikir yang utuh, yang selanjutnya dikembangkan menjadi *blue print* pengembangan *e-Government* di setiap lembaga pemerintah. Konsep pengembangan infrastruktur diarahkan kepada pemanfaatan semaksimal mungkin sumberdaya informasi yang telah ada sebagai modal utama dalam mengembangkan *e-Government*. Pengembangan *e-Government* pada setiap lembaga, selain akan meningkatkan pemanfaatan sistem informasi yang dimiliki, juga diharapkan meningkatkan layanan publik dan operasional pengelolaan pemerintahan secara lebih efektif dan efisien.

## 2.3 JSON (JavaScript Object Notation)

*JSON* (dilafalkan "Jason"), singkatan dari *JavaScript Object Notation* (bahasa Indonesia: notasi objek *JavaScript*), adalah suatu format ringkas pertukaran data komputer. Formatnya berbasis teks dan terbaca-manusia serta digunakan untuk merepresentasikan struktur data sederhana dan larik asosiatif (disebut objek). Format *JSON* sering digunakan untuk mentransmisikan data terstruktur melalui suatu koneksi jaringan pada suatu proses yang disebut serialisasi. Aplikasi utamanya adalah pada pemrograman aplikasi *web AJAX* dengan berperan sebagai alternatif terhadap penggunaan tradisional format *XML*. Walaupun *JSON* didasarkan pada subset bahasa pemrograman *javascript* (secara spesifik, edisi ketiga standar ECMA-262, Desember 1999) dan umumnya digunakan dengan bahasa tersebut, *JSON* dianggap sebagai format data yang tak tergantung pada suatu bahasa. Kode untuk pengolahan dan pembuatan data *JSON* telah tersedia untuk banyak jenis bahasa pemrograman. Situs [json.org](http://json.org) menyediakan daftar komprehensif pengikatan *JSON* yang tersedia, disusun menurut bahasa. Format *JSON* dispesifikasikan di RFC 4627 oleh Douglas Crockford. Tipe media Internet resmi *JSON* adalah *application/json* sedangkan ekstensi berkasnya adalah *.json*. *JSON* merupakan salah satu format pertukaran data antara browser dengan server. *JSON* memiliki format yang lebih sederhana, mendukung semua *platform*, sistem yang *heterogen*

(*interoperabilitas*) dan elegan dibandingkan dengan pendahulunya, *XML*.

Objek adalah sepasang nama/nilai yang tidak terurutkan. Objek dimulai dengan { (kurung kurawal buka) dan diakhiri dengan } (kurung kurawal tutup). Setiap nama diikuti dengan : (titik dua) dan setiap pasangan nama/nilai dipisahkan oleh, (koma). Sebuah objek dideklarasikan dengan

```
var myObject = {};
```

Terlihat sangat sederhana, didalam pasangan kurung kurawal dapat berisi berbagaimacam informasi dari yang paling sederhana hingga kompleks. *Javascript* menyimpan semua informasi di dalam pasangan kurungkurawal yang dapat berisi semua bentuk tipe data primitive seperti *string*, angka (*number*), *array*, tanggal (*date*) dan *regular expression*.

Cara lama untuk membuat sebuah objek adalah dengan menggunakan kata-kunci *new*.

```
var myJSON = new Object();
```

Metode ini telah diperbaiki. Sekarang untuk membuat objek kosong, dapat digunakan dengan hanya mendefinisikan pasangan kurungkurawal.

```
var myJSON = {};
```

Seperti halnya kebanyakan *object* dasar *Javascript*, format data sangatlah fleksibel yang ditulis sebagai pasangan nama dan nilai. Objek memiliki nama yang terkandung dalam *property object* yang biasanya disebut sebagai nama *object*. *Object* juga memiliki *property* nilai, seperti contoh berikut :

```
var myFirstJSON = { "firstName" : "string",  
  "lastName" : "string", "age" : string};  
document.writeln(myFirstJSON.firstName);  
document.writeln(myFirstJSON.lastName);  
document.writeln(myFirstJSON.age);
```

Objek ini memiliki tiga *property* atau pasangan nama dan nilai. Nama adalah sebuah string (pada contoh di atas *firstName*, *lastName* dan *age*). Nilai dapat diisi dengan semua jenis *object javascript* (semua di *javascript* adalah *object*, nilai dapat diisi dengan string, angka, array, fungsi dan *object* lainnya. Format data ini disebut sebagai *JSON (JavaScript Object Notation)*. Yang menjadikan *JSON* tangguh adalah

kemampuan nilai diisi dengan semua jenis data. Nilai dapat diisi dengan array atau *object* yang bertingkat dengan kedalaman yang tak terhingga.

*Array* adalah kumpulan nilai yang terurutkan. *Array* dimulai dengan [ (kurung kotak buka) dan diakhiri dengan ] (kurung kotak tutup). Setiap nilai dipisahkan oleh , (koma).

Nilai (*value*) dapat berupa sebuah string dalam tanda kutip ganda, atau angka, atau *true* atau *false* atau *null*, atau sebuah objek atau sebuah larik. Struktur-struktur tersebut dapat disusun bertingkat.

*String* adalah kumpulan dari nol atau lebih karakter *Unicode*, yang dibungkus dengan tanda kutip ganda. Di dalam string dapat digunakan *backslash escapes* "\" untuk membentuk karakter khusus. Sebuah karakter mewakili karakter tunggal pada *string*. *String* sangat mirip dengan *string C* atau *Java*.

Angka adalah sangat mirip dengan angka di *C* atau *Java*, kecuali format oktal dan heksadesimal tidak digunakan.

## 2.4 Pengaksesan JSON

Cara yang paling umum untuk mengakses data *JSON* adalah dengan menggunakan notasi titik. Caranya dengan menuliskan nama *object* diikuti dengan titik dan kemudian diikuti dengan nama/*property*.

```
var myObject = { 'color' : 'blue' };  
document.writeln(myObject.color); // output  
blue
```

Jika *object* mengandung *object* lagi di dalamnya, tinggal tambahkan titik dan nama di belakangnya.

```
var myObject = { 'color' : 'blue', 'animal' : {  
  'cat' : 'friendly' } };  
document.writeln(myObject.animal.cat); //  
output friendly
```

## 3. METODOLOGI PENELITIAN

### 3.1 Alat dan Bahan Penelitian

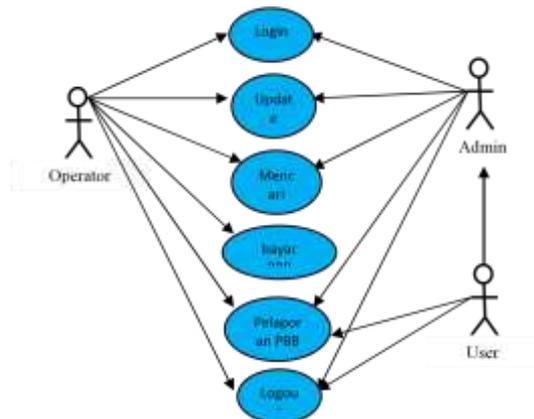
Pada penelitian ini akan dilakukan pengiriman data hasil pembayaran PBB di kecamatan ke DP2KA dengan format data menggunakan *JSON* sebagai pendukung sistem informasi manajemen pengelolaan/pengendalian Pendapatan Asli Daerah Tulang Bawang-Lampung.

Pengiriman data yang diperoleh di tiap-tiap kecamatan ke DP2KA dikirim melalui e-mail sebagai media pengiriman data, yang selanjutnya data di input ke database DP2KA. Selanjutnya, menjadi data monitoring bagi kepala daerah dan instansi terkait melalui web, sehingga kepala daerah dan instansi yang berwenang dapat memantau pemasukan PAD untuk tiap-tiap transaksi yang terjadi setiap hari. Data yang digunakan adalah data setoran pajak bumi dan bangunan dari beberapa kecamatan, yang diolah dengan *Mysql* melalui *PHP*. Untuk menguji model tersebut dibuat suatu *prototype* perangkat lunak menggunakan *PHP*, *Javascript* dan *JSON*, salah satu aspek pemilihan teknologi tersebut adalah karena sesuai dengan lingkungan *web* dan bersifat *open source*. Alat yang dibutuhkan untuk pengolahan dan pengujian dilakukan pada komputer dengan sistem operasi *Windows XP* yang telah mempunyai akses internet dan *LAN*, *account Gmail* untuk kecamatan dan instansi terkait. Dibutuhkan perangkat lunak untuk menguji *prototype*, yaitu:

- DATABASE : *MySQL*
- DEVELOPMENT TOOLS : *JSON*, *JavaScript* dan *PHP*

Materi penelitian yang digunakan adalah data Pajak Bumi dan Bangunan yang ada di kecamatan.

### 3.2 Requirement/Use Case dan Constraint (Secara Umum)



Gambar 1. Use Case Sistem Informasi PBB

Penjelasan model *use case/requirement* yang ingin didukung oleh aplikasi, sebagai berikut:

- Kepala Daerah (*user*) dan dinas pengelola PAD/DP2KA (*admin*) ingin memonitor setiap transaksi pembayaran

- PBB yang menjadi penghasil PAD disetiap kecamatan yang ada di Kabupaten Tulang Bawang.
- Transaksi yang dilakukan di kecamatan melalui operator kecamatan (*petugas*) harus dikirim ke dinas terkait agar bisa dibuatkan laporan keseluruhan kepada kepala daerah.
- Data detail transaksi dan jenis transaksi juga harus dikirim ke dinas pengelola PAD/DP2KA.

### 3.3 Identifikasi Penggunaan Sistem

Pengguna sistem memiliki otorisasi untuk menentukan operasi apa saja yang dapat dilakukan. Biasanya didefinisikan dengan matrik CRUD. Matrik CRUD mendefinikan hak menciptakan (C) *create*, membaca (R) *read*, memperbaharui (U) *update* dan menghapus (D) *delete*. Tabel 3.1 menjelaskan otorisasi dari sistem informasi pajak bumi dan bangunan:

TABEL 3.1  
OTORISASI PENGGUNA SISTEM

Jenis Pengguna	Otorisasi	Keterangan
User (Kep. Daerah)	R	<i>Read</i>
Admin (DP2KA)	R,U,D	<i>Read, Update, Delete</i>
Petugas (kecamatan)	U,R	<i>Update, Read</i>

### 3.4 Penentuan Model Pengiriman Data

Setelah *requirement*, *constraint* dan otorisasi ditentukan, tahap berikutnya adalah membuat daftar alternatif solusi model pengiriman data yang sesuai. Berikut adalah alternatif yang dapat dilakukan untuk memecahkan masalah :

- Remoting*
  - Web Services*
  - Java RMI*
  - HTTPInvoker*
- Messaging*
  - JMS*
  - Email (POP3/IMAP dan SMTP)*
- File
  - Shared Folder*
  - FTP*
  - SSH/SCP*
- Database*

- i. *Shared Database*
- ii. *Replikasi Database*

Dalam permasalahan ini, jaringan publik melalui internet sangat tidak *reliable* dan memiliki banyak keterbatasan. *Port* tidak bisa dibuka secara bebas, karena mungkin saja diblokir pada saat sedang dipergunakan. Penggunaan protokol yang terlalu kompleks (membutuhkan transfer data yang sering). Oleh karena itu, beberapa alternatif berikut bisa dihilangkan:

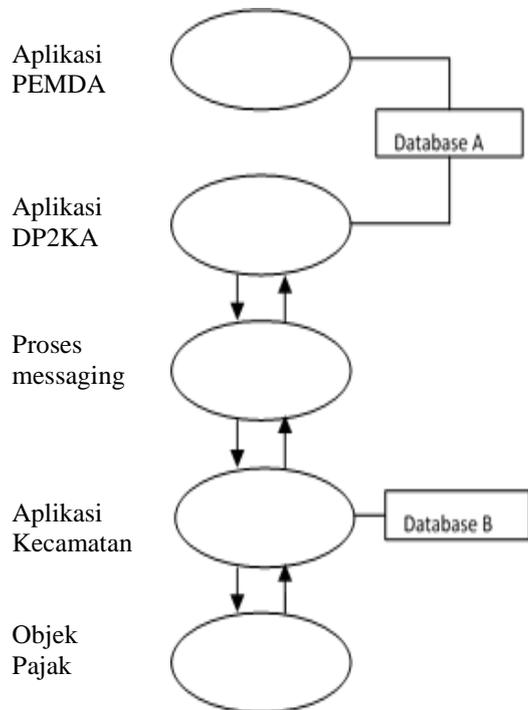
- a. *Remoting* dengan Java *RMI*: port tidak standar
- b. *Messaging* dengan *JMS*: port tidak standar
- c. Replikasi *MySQL*: protokol terlalu kompleks, port tidak standar

Konsumsi *bandwidth* dapat menjadi pertimbangan karena akan menghabiskan biaya yang besar. Protokol yang memerlukan *bandwidth* besar seperti *Remoting* dengan *Web Service* dapat dihindari. Dari penggunaan waktu, harus mempertimbangkan beberapa alternatif yang mengharuskan Dinas dan kecamatan *online* bersamaan, yaitu *Remoting* dengan *Spring HTTPInvoker* dan *File Transfer*, baik *FTP* maupun *SSH/SCP*. Dengan demikian, yang memungkinkan adalah *Messaging* dengan *e-mail*. Pengiriman data dapat dilakukan dengan *attachment*. Untuk menghemat *bandwidth*, sebelum *attachment* bisa dikompresi. Dinas dan kecamatan bisa *online* sesuai dengan pembagian waktu yang telah ditentukan (*time sharing/schedule*), dan akan tetap menerima *message*. Kalau ada pesan yang *error*, pihak dinas bisa menelepon untuk minta dikirim ulang datanya. Memilih *e-mail* sebagai media transfer data, maka tidak perlu disibukkan dengan mengelola infrastruktur jaringan. Dengan menggunakan layanan *GMail* yang gratis, berkapasitas besar, dan memiliki implementasi *POP3*, *IMAP*, dan *SMTP*, maka pengiriman data dapat dilakukan dengan biaya yang murah.

### 3.5 Skema Rancangan e-Government PBB

Skema rancangan e-goverenment PBB, disajikan pada Gambar 2. pada proses *messaging* menggunakan *e-mail* melibatkan dua instansi pemerintah yaitu kecamatan dan DP2KA, dengan masing-masing proses sebagai berikut:

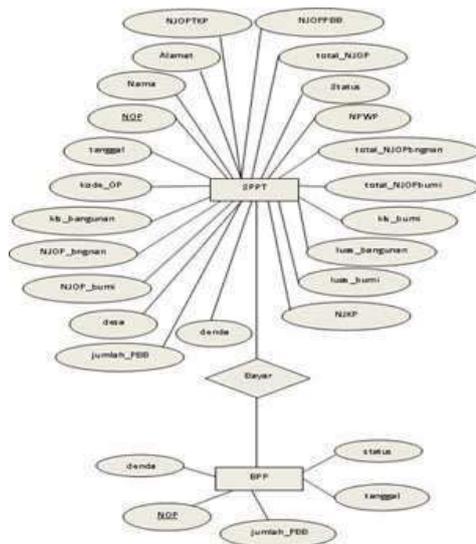
- a. Proses pengiriman data
  - i. Membuat objek pajak yang akan dikirim
  - ii. Membuat *class* untuk mengkonversi objek pajak menjadi sebuah file *JSON*
  - iii. Membuat *class* pengujian pada proses a dan b
  - iv. Membuat *class sender*
  - v. Membuat *gateway* sebagai *interface* untuk pengiriman *message*
- b. Proses penerimaan data
  - i. Membuat *class message receiver*
  - ii. Membuat *class service*, dengan tujuan mengkonversi file menjadi objek pajak.



Gambar 2. Skema Rancangan e-Government PBB

### 3.6 Rancangan Basis Data

*Entity Relationship Diagram (ERD)* yang dirancang pada sistem informasi Pajak Bumi dan Bangunan (PBB) dalam penelitian ini adalah *ERD* untuk DP2KA, Kecamatan dan *user*. *ERD* rancangan disajikan pada Gambar 3.

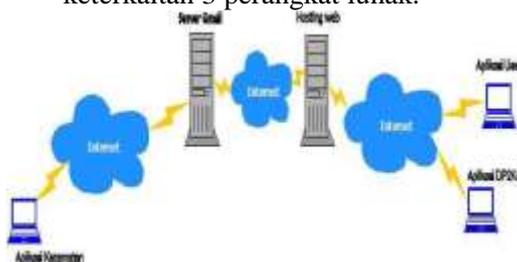


**Gambar 3. Rancangan Entity Relationship Diagram pada DP2KA**

#### 4. HASIL DAN PEMBAHASAN

Penelitian ini merancang dan membuat tiga buah perangkat lunak, yang terdiri dari:

- Perangkat lunak di kecamatan  
Perangkat lunak ini merupakan aplikasi desktop berbasis web, yang bertujuan untuk pelayanan pembayaran PBB, dengan *record* basis data, *username*, *password* berbeda-beda di setiap kecamatan.
- Perangkat lunak di DP2KA  
Aplikasi web ini mengelola data transaksi yang masuk dari 15 kecamatan yang ada di Kabupaten Tulang Bawang dan melakukan *update* basis data setiap data transaksi yang masuk. Aplikasi ini ditempatkan pada hosting web.
- Perangkat lunak untuk user  
Aplikasi untuk *user* ditempatkan pada satu *hosting* web yang sama dengan DP2KA, dengan *login username* dan *password* yang berbeda dengan DP2KA. Fungsi aplikasi ini sebagai pelaporan bagi *user*. Gambar 4 melukiskan keterkaitan 3 perangkat lunak.



**Gambar 4. Hubungan antara aplikasi kecamatan, DP2KA dan user**

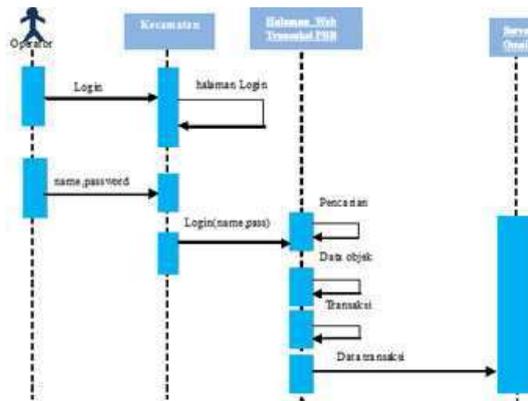
#### 4.1 Pengujian Kekuatan Sinyal GSM di Kecamatan-Kecamatan

Kekuatan sinyal GSM dan CDMA yang diterima di tiap-tiap kecamatan berbeda-beda. Permasalahan ini sangat penting guna mendukung proses pengiriman data dari kecamatan ke DP2KA. Berikut adalah jenis sinyal yang dapat diterima di 15 kecamatan Kabupaten Tulang Bawang berdasarkan pengujian dengan menggunakan modem dan informasi dari staf kecamatan:

**TABEL 4.1  
PENGUJIAN KEKUATAN SINYAL GSM**

No	Kecamatan	Operator	Jenis Sinyal	Metode	Jenis Modem
1	Menggal a	Simpati	3G	Pengujian	T-Mobile MF 626
2	Menggal a Timur	Simpati	3G	Pengujian	T-Mobile MF 626
3	Gedung Aji	Simpati	3G	Pengujian	T-Mobile MF 626
4	Banjar Agung	Simpati	3G/HS DPA	Pengujian	T-Mobile MF 626
5	Penawar Aji	Simpati	3G	Pengujian	T-Mobile MF 626
6	Meraksa Aji	Simpati	3G	Pengujian	T-Mobile MF 626
7	Banjar Margo	Smart fren	-	Interview	-
8	Dente Teladas	simpati	-	Interview	-
9	Gedung Aji Baru	Smart fren /Simpati	-	Interview	-
10	Gedung Meneng	Simpati	-	Interview	-
11	Banjar Baru	Simpati /Smart fren	-	Interview	-
12	Penawar Tama	Simpati	-	Interview	-
13	Rawa Pitu	Simpati	-	Interview	-
14	Rawa Jitu Selatan	Simpati /Smart fren	-	Interview	-
15	Rawa Jitu Timur	Simpati	-	Interview	-

## 4.2 Diagram Skenario Aplikasi Kecamatan

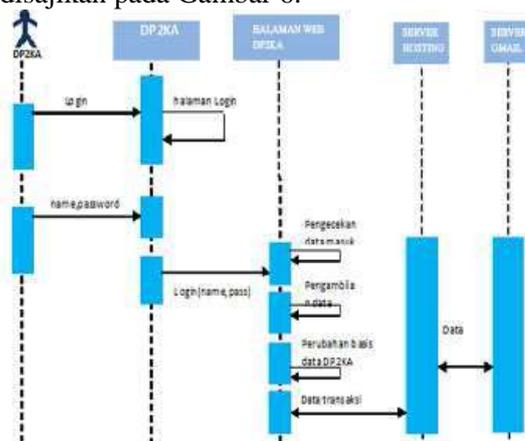


Gambar 5. Diagram Skenario Aplikasi Kecamatan

Pada proses aplikasi pembayaran PBB yang dilakukan di kecamatan, diperlukan alur kerja dari proses pembayaran PBB untuk memudahkan dalam membaca proses transaksi pembayaran PBB di kecamatan, seperti yang disajikan pada Gambar 5.

## 4.3 Diagram Skenario Aplikasi DP2KA

Pada aplikasi PBB di DP2KA mengolah data transaksi yang dikirim dari kecamatan-kecamatan yang ada di Tulang Bawang untuk dijadikan hasil pelaporan. Diperlukan alur proses dari pengambilan data pembayaran PBB dari kecamatan oleh DP2KA yang disajikan pada Gambar 6.



Gambar 6. Diagram Skenario Aplikasi DP2KA

## 4.4 Pengujian Sistem

Pengujian sistem dilakukan dengan menggunakan data simulasi SPPT salah satu kecamatan.

TABEL 4.4  
HASIL PENGUJIAN PENGIRIMAN DATA

Hasil pengujian (data benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan
Mengirim data transaksi dari Kecamatan Menggala ke DP2KA	Tampil perubahan jumlah data masuk pada Kec.Menggala di aplikasi DP2KA	Penambahan jumlah data masuk	[x] Diterima [ ] Ditolak

Pengiriman data hasil transaksi pada aplikasi kecamatan dilakukan pada kekuatan sinyal, *edge*, *3G* dan *HSDPA*. TABEL 4.4 menunjukkan hasil pengujian sistem. Pada pengujian sistem tidak terdapat kendala pengiriman data dari Kecamatan Menggala ke DP2KA. Akan tetapi, terjadi permasalahan pada *hosting* untuk aplikasi DP2KA pada saat memeriksa data masuk dari kecamatan melalui aplikasi DP2KA. Permasalahan ini disebabkan karena proses *looping* pada aplikasi DP2KA memerlukan waktu yang panjang, sehingga harus merubah lamanya waktu akses pada *hosting*.

## 5. KESIMPULAN

Pemilihan penggunaan *e-mail* sebagai media transfer data sangat tepat diimplementasikan pada aplikasi ini, karena tidak semua kecamatan yang ada di Kabupaten Tulang Bawang mendukung teknologi jaringan data (*third-generation technology*) *3G*, (*Code Division Multiple Access*) *CDMA* atau (*High-Speed Downlink Packet Access*) *HSDPA*. Pada jaringan (*General Packet Radio Service*) *GPRS* atau (*Enhanced Data rates for GSM Evolution*) *EDGE*, besarnya kapasitas data yang di transfer dari kecamatan ke DP2KA sangat mempengaruhi keberhasilan pengiriman data. Untuk itu, dalam penelitian ini, menggunakan format data *JSON* sebagai pengiriman data hasil transaksi pembayaran.

Implementasi sistem informasi pengelolaan pendapatan asli daerah Kabupaten Tulang Bawang berbasis *e-mail* merupakan aplikasi *e-Government* dengan biaya yang murah dan *open source* tanpa harus disibukkan dengan pembangunan

infrastruktur jaringan yang mahal dan kemampuan sumber daya manusia yang tinggi sebagai operator pelaksana.

Penempatan aplikasi DP2KA dan *user* akan lebih baik jika menggunakan jasa *Virtual Private Server* (VPS) karena akan lebih mudah mengatur lama akses aplikasi pada saat *looping* pemeriksaan data masuk dari kecamatan ke DP2KA.

## 6. DAFTAR PUSTAKA

- [1] Bo Yang. 2010. "Querying JSON Streams". Institutionen för informat ionstek nologi Uppsala Universitet
- [2] HAN Min, FENG Hao.2010. "Research of GIS data exchange method based on JSON".  
[http://en.cnki.com.cn/Article\\_en/CJFD TOTALCHKD20100 1058.htm](http://en.cnki.com.cn/Article_en/CJFD TOTALCHKD20100 1058.htm)
- [3] Hansen Ute.2006. "How to Manage and Control e-Government".  
<http://www.slideshare.net/Billy82/how-to-manage-and-controlegovernment>. diakses 26 Februari 2011
- [4] RUTECKI KRZYSZTOF.2012. "Practical approach to design and implementa tion of fault-tolerant, continuous-operation data acquisition module". Institute of Telecommunications, Teleinformatics and Acoustics Wroclaw University of Technology Wyb. Wyspianskiego 27, 50-370 Wroclaw POLAND
- [5] Keoduangsine Saysoth, Goodwin Robert.2012." A GPRS-Based Data Collection and Transmission for Flood Warning System: The Case of the Lower Mekong River Basin". International Journal of Innovation, Management and Technology, Vol. 3, No. 3, June 2012
- [6] Ahmad Masykur. 2008. JSON vs XML.  
<http://www.masykur.web.id/post/JSON-vs-XML.aspx>. diakses tanggal 12 April 2010
- [7] Xiaofeng HU.2010." Application Analysis of JSON and XML on Networ ked Data Transmission".  
[http://en.cnki.com.cn/Article\\_en/CJFD TOTAL-DNBC20 1010031.htm](http://en.cnki.com.cn/Article_en/CJFD TOTAL-DNBC20 1010031.htm)
- [8] Maheswari Piyush.2003. "Enterprise Application Integration using a Component-based Architecture" compsoc, pp.557. 27th Annual International Computer Software and Applications Conference
- [9] Khosravi Majid.2012. "XML vs JSON parsing in Android".  
<http://www.majidkhosravi.com/xml-vs-json-android/>. diakses 9 Januari 2013
- [10] Topcu et al. "Web 2.0 for E-Science Environments". Community Grids Laboratory, Indiana University, 501 North Morton Street, Suite 224 Bloomington, IN 47404
- [11] Nurseitov Nurzhan.2009. "Comparison of JSON and XML Data Interchange Formats: A Case Study".[http://www.researchgate.net/publication/220922905\\_Comparison\\_of\\_JSON\\_and\\_XML\\_Data\\_Interchange\\_Formats\\_A\\_Case\\_Study](http://www.researchgate.net/publication/220922905_Comparison_of_JSON_and_XML_Data_Interchange_Formats_A_Case_Study)
- [12] Hidayanto, Ahmad Nizar .2000. Pemrograman dan obyek terdistribusi pada aplikasi piranti bantu bekerja sama. Universitas Indonesia
- [13] Hunlock, Patrick. 2007. "Mastering JSON (JavaScript Object Nota tion)".  
<http://www.scribd.com/doc/15009816/Mastering-in-JSON-JavaScript-Object-Notation->
- [14] Hasibuan, Z.A., 2002, "Electronic Government for Good Governance", Jurnal Sistem Informasi dan Manajemen Teknologi Informasi, Vol. 1, Nomor 1
- [15] JSON official website [Online].  
<http://www.json.org/>. 18 Oktober 2010
- [16] The World Bank Group, "A Definition of E-Government",  
<http://www1.worldbank.org/publicsecto r/egov/definition.htm> ,[online], 4 Juni 2010.

- [17] John E.S., alih bahasa: Dwi Prabantini. 2002. "Just XML". Penerbit Andi, Yogyakarta
- [18] Badan Pusat Statistik -Direktorat Pengembangan Metodologi Sensus dan Survei. 2009. "Sistem Pemutakhiran MFD dan MBS". [www.http://mfdonline.bps.go.id/](http://mfdonline.bps.go.id/). diakses 24 Maret 2012