

RANCANGAN PENGUJIAN VERIFIKASI ATAU VALIDASI  
UNTUK SISTEM INFORMASI PEMBAYARAN PAJAK BUMI  
DAN BANGUNAN PERDESAAN DAN PERKOTAAN DI  
KABUPATEN BREBES

---

PERIODE PENILAIAN TAHUN 2018



Oleh :

Priyanto Tamami, S.Kom.

NIP 19840409 201001 1 025

Fungsional Pranata Komputer

Badan Pengelolaan Pendapatan, Keuangan dan Aset Daerah

Pemerintah Daerah Kabupaten Brebes

Brebes, 23 Maret 2018

# Lembar Pengesahan

Nama Kegiatan : Merancang Pengujian Verifikasi atau Validasi Program  
Judul : RANCANGAN PENGUJIAN VERIFIKASI ATAU VALIDASI UNTUK SISTEM INFORMASI PEMBAYARAN PAJAK BUMI DAN BANGUNAN PERDESAAN DAN PERKOTAAN DI KABUPATEN BREBES

Disetujui oleh :	Disusun Oleh
Kepala Sub Bidang Keberatan	Pranata Komputer
Pada tanggal 26 Maret 2018	Selesai tanggal : 23 Maret 2018

M.L. Setiyawan, S.E.Ak	Priyanto Tamami, S.Kom
NIP 19790530 200604 1 006	NIP 19840409 201001 1 025

# Daftar Isi

<b>1 PROSEDUR VERIFIKASI / VALIDASI SPESIFIKASI PRO-</b>	
<b>GRAM</b>	<b>1</b>
1.1 Pengujian Permintaan Terhadap Objek Pajak . . . . .	2
1.1.1 Nomor Objek Pajak . . . . .	2
1.1.2 Luas Bumi Objek Pajak . . . . .	2
1.1.3 Luas Bangunan Objek Pajak . . . . .	2
1.1.4 NJOP Bumi Objek Pajak . . . . .	3
1.1.5 NJOP Bangunan Objek Pajak . . . . .	3
1.1.6 Nama Jalan Objek Pajak . . . . .	3
1.1.7 RT dan RW Objek Pajak . . . . .	3
1.1.8 Nama Kelurahan Objek Pajak . . . . .	4
1.1.9 Nama Kecamatan Objek Pajak . . . . .	4
1.1.10 Nomor Identitas Subjek Pajak . . . . .	4
1.2 Pengujian Permintaan Terhadap Data Subjek Pajak . . . . .	5
1.2.1 Nama Subjek Pajak . . . . .	5
1.2.2 Nama Jalan Alamat Subjek Pajak . . . . .	5
1.2.3 RT dan RW dari Alamat Subjek Pajak . . . . .	5
1.2.4 Nama Kelurahan dari Alamat Subjek Pajak . . . . .	6
1.2.5 Nama Kota atau Kabupaten dari Alamat Subjek Pajak . . . .	6

1.3	Pengujian Permintaan Data Ketetapan . . . . .	6
1.3.1	Pengujian Atas Daftar Ketetapan . . . . .	6
1.3.2	Pengujian Besarnya Nilai Pajak Terhutang Tiap Tahun Pajak	7
1.3.3	Pengujian Status Pembayaran Untuk Tiap Tahun Pajak . .	8
1.4	Pengujian Modul <i>Service</i> . . . . .	8
1.4.1	Pengujian Terhadap Kelas <code>DatObjekPajakService</code> . . . . .	8
1.4.2	Pengujian Terhadap Kelas <code>DatSubjekPajakService</code> . . . . .	9
1.4.3	Pengujian Terhadap Kelas <code>SpptService</code> . . . . .	10
<b>2</b>	<b>PETUGAS VERIFIKASI ATAU VALIDASI PROGRAM</b>	<b>11</b>

# Daftar Gambar

## Bab 1

# PROSEDUR VERIFIKASI / VALIDASI SPESIFIKASI PROGRAM

Prosedur verifikasi atas spesifikasi program akan dilakukan saat pengembangan aplikasi / program dengan *unit testing* dan *integration testing*.

Data yang digunakan untuk pengembangan dan pengujian dilakukan dengan data *dummy* menggunakan struktur data asli seperti pada spesifikasi sistem basis data.

*Unit testing* dan *integration testing* yang dilakukan akan terbagi menjadi beberapa kelompok seperti spesifikasi berikut :

## 1.1 Pengujian Permintaan Terhadap Objek Pajak

### 1.1.1 Nomor Objek Pajak

Pada saat klien melakukan permintaan data ke peladen, hal pertama yang dilakukan adalah memeriksa kebenaran Nomor Objek Pajak (NOP) yang sampai ke peladen, apakah Nomor Objek Pajak yang dikirimkan oleh klien, telah diterima benar susunannya pada peladen.

Pemeriksaan ini dilakukan pada objek `JsonObject`. Apabila *method* `setNop()` berisi parameter berupa Nomor Objek Pajak (NOP) 332901000100500390, maka *method* `getNop()` harus memberikan hasil yang sama.

### 1.1.2 Luas Bumi Objek Pajak

Pada bagian ini, klien akan melakukan *request* atau permintaan data ke peladen dengan membawa parameter `keyword` berisi `op`, dan `nop` berisi 332901000100500390.

Hasil dari *request* atau permintaan terhadap API (*application programmable interface*) ini berupa data bertipe `BigDecimal` dengan nilai 80 untuk luas bumi objek pajaknya.

### 1.1.3 Luas Bangunan Objek Pajak

Pada bagian ini, klien masih melakukan *request* atau permintaan data ke peladen dengan membawa parameter `keyword` berisi `op`, dan `nop` berisi 332901000100500390.

Hasil dari *request* atau permintaan terhadap API (*Application Programmable Interface*) ini berupa data bertipe `BigDecimal` dengan nilai 40 untuk luas bangunan objek pajaknya.

#### 1.1.4 NJOP Bumi Objek Pajak

Pada bagian ini, klien masih melakukan *request* atau permintaan data ke peladen dengan parameter yang sama, yaitu **keyword** yang berisi **op**, dan **nop** yang berisi 332901000100500390.

Hasil keluaran untuk Nilai Jual Objek Pajak (NJOP) Bumi dari *request* atau permintaan data tersebut akan bertipe **BigDecimal** dengan nilai 1600000 (Satu juta enam ratus) rupiah per meter persegi.

#### 1.1.5 NJOP Bangunan Objek Pajak

*Request* untuk bagian ini masih dengan parameter yang sama, yaitu parameter **keyword** yang berisi **op**, dan **nop** yang berisi 332901000100500390.

Hasil Nilai Jual Objek Pajak (NJOP) Bangunan yang didapat dari *request* atau permintaan data tersebut bertipe **BigDecimal** dengan nilai 9000000 (Sembilan juta) rupiah per meter persegi.

#### 1.1.6 Nama Jalan Objek Pajak

Untuk pengambilan nama jalan dari objek pajak, masih menggunakan parameter yang sama untuk *request* atau permintaan data ke peladen, yaitu dengan parameter **keyword** yang berisi **op**, dan **nop** yang berisi 332901000100500390.

Hasil yang seharusnya didapat dari *request* atau permintaan data di atas akan memiliki tipe data **String** dengan nilai KP RW 01.

#### 1.1.7 RT dan RW Objek Pajak

Untuk pengambilan data RT dan RW objek pajak, masih menggunakan parameter yang sama untuk *request* atau permintaan data ke peladen, yaitu dengan parameter **keyword** yang berisi **op**, dan **nop** yang berisi 332901000100500390.



Hasil yang seharusnya didapat dari *request* atau permintaan data tersebut akan memiliki tipe data **string** dengan nilai RT. 002 RW. 01.

### 1.1.8 Nama Kelurahan Objek Pajak

Pengambilan data nama Kelurahan atau nama Desa untuk objek pajak masih menggunakan parameter yang sama untuk *request* atau permintaan data ke peladen, yaitu dengan parameter **keyword** yang berisi **op**, dan **nop** yang berisi 332901000100500390.

Hasil yang seharusnya didapatkan dari *request* atau permintaan data tersebut akan memiliki tipe data **string** dengan nilai GUNUNGJAYA.

### 1.1.9 Nama Kecamatan Objek Pajak

Tidak berbeda dengan pengujian sebelumnya, untuk pengambilan data nama Kecamatan sebuah objek pajak masih menggunakan parameter yang sama untuk *request* atau permintaan data ke peladen, yaitu dengan parameter **keyword** yang berisi **op**, dan **nop** yang berisi 332901000100500390.

Hasil yang seharusnya didapatkan dari *request* atau permintaan data tersebut akan memiliki tipe data **string** dengan nilai SALEM.

### 1.1.10 Nomor Identitas Subjek Pajak

Untuk pengambilan data nomor identitas subjek pajak pun masih menggunakan parameter *request* atau permintaan data ke peladen berupa parameter *keyword* yang berisi **op** dan parameter **nop** yang berisi 332901000100500390.

Hasil yang seharusnya didapatkan dari *request* atau permintaan data tersebut akan memiliki tipe data **string** dengan nilai 332901000100500390

## 1.2 Pengujian Permintaan Terhadap Data Subjek Pajak

Untuk pengujian permintaan terhadap data subjek pajak akan terdiri dari beberapa bagian berikut :

### 1.2.1 Nama Subjek Pajak

Permintaan data nama subjek pajak dilakukan oleh klien ke peladen API (*Application Programmable Interface*) dengan parameter **keyword** yang isinya **wp** dan parameter **subjek\_pajak\_id** yang berisi nomor identitas dari subjek pajak tersebut, yang diambil sebagai contoh pengujian adalah 332901000100500390.

Hasil yang seharusnya didapatkan dari *request* atau permintaan data tersebut berupa nama KARSO dengan tipe data **string**.

### 1.2.2 Nama Jalan Alamat Subjek Pajak

Untuk pengujian nama jalan dari alamat subjek pajak, akan dilakukan oleh klien ke peladen dengan *request* parameter berupa **keyword** dengan nilai **wp** dan parameter **subjek\_pajak\_id** dengan nilai 332901000100500390.

Hasil yang seharusnya didapatkan dari *request* atau permintaan data tersebut adalah nilai KP RW 01 dengan tipe data **string**.

### 1.2.3 RT dan RW dari Alamat Subjek Pajak

Untuk pengujian permintaan data RT dan RW dari alamat subjek pajak yang akan dilakukan klien ke peladen adalah menggunakan parameter **keyword** yang berisi **wp** dan parameter **subjek\_pajak\_id** dengan isi 332901000100500390.

Hasil yang didapat untuk *request* atau permintaan tersebut seharusnya adalah berupa data RT. 002 RW. 01 dengan tipe data `string`

#### 1.2.4 Nama Kelurahan dari Alamat Subjek Pajak

Untuk pengujian permintaan data nama Kelurahan / Desadari alamat subjek pajak yang akan dilakukan klien ke peladen adalah menggunakan parameter `keyword` yang berisi `wp`, dan parameter `subjek_pajak_id` yang berisi 332901000100500390.

Hasil yang seharusnya didapat untuk parameter tersebut adalah data GUNUNGJAYA dengan tipe data `string`.

#### 1.2.5 Nama Kota atau Kabupaten dari Alamat Subjek Pajak

Untuk pengujian permintaan data nama Kota atau Kabupaten dari alamat subjek pajak yang dilakukan klien ke peladen adalah menggunakan parameter `keyword` yang berisi `wp`, dan parameter `subjek_pajak_id` yang berisi 332901000100500390.

Hasil yang seharusnya didapat untuk parameter tersebut adalah data KAB BREBES dengan tipe data `string`.

### 1.3 Pengujian Permintaan Data Ketetapan

Untuk pengujian permintaan data ketetapan tiap tahun pajak, hal yang perlu dilakukan pengujian adalah seperti berikut ini :

#### 1.3.1 Pengujian Atas Daftar Ketetapan

Pengujian atas daftar ketetapan dilakukan klien dengan cara melakukan *request* atau permintaan data ke peladen dengan parameter `keyword` yang berisi `sppt`,

dan parameter `nop` yang berisi 332901000100500390.

Hasil yang seharusnya dikembalikan oleh peladen adalah berupa daftar tahun pajak dimana Nomor Objek Pajak tersebut telah terbit sebagai Surat Pemberitahuan Pajak Terhutang (SPPT) untuk tiap tahun pajak, adapun banyaknya data yang seharusnya ditampilkan untuk Nomor Objek Pajak (NOP) tersebut adalah 8 (delapan data).

### 1.3.2 Pengujian Besarnya Nilai Pajak Terhutang Tiap Tahun Pajak

Untuk pengujian besarnya nilai pajak, akan menggunakan 3 (tiga) contoh data seperti berikut :

- *Request* pertama menggunakan parameter `keyword` yang berisi `bayar`, parameter `nop` yang berisi 332911090400000967, dan parameter `tahun` yang berisi 1994. Hasil yang diharapkan dari *request* ini adalah data 2088 dengan tipe data `BigDecimal`.
- *Request* yang kedua menggunakan parameter `keyword` yang berisi `bayar`, parameter `nop` yang berisi 332912090500003797, dan parameter `tahun` yang berisi 1994. Hasil yang diharapkan dari *request* ini adalah data 2720 yang bertipe `BigDecimal`.
- *Request* yang ketiga menggunakan parameter `keyword` yang berisi `bayar`, parameter `nop` yang berisi 332912090500003657. Hasil yang diharapkan dari *request* ini adalah data 3920 dengan tipe data `BigDecimal`.

### 1.3.3 Pengujian Status Pembayaran Untuk Tiap Tahun Pajak

Untuk pengujian status pembayaran untuk tiap tahun pajak akan menggunakan 1 (satu) contoh data dalam 1 (satu) kali *request*, dan akan dicocokkan datanya dengan 3 (tiga) tahun pertama terbit, apakah statusnya sudah LUNAS atau BELUM LUNAS.

*Request* yang dilakukan klien ke peladen menggunakan parameter **keyword** yang berisi **sppt** dan parameter **nop** dengan isi 332901000100500390. Hasil yang diharapkan untuk data pertama, data kedua, dan data ketiga dari daftar yang diberikan oleh peladen harus bernilai LUNAS.

## 1.4 Pengujian Modul *Service*

Pengujian atas *service* yang terbentuk akan terbagi menjadi 3 (tiga) bagian seperti jumlah kelas yang terdapat pada paket **service** seperti berikut :

### 1.4.1 Pengujian Terhadap Kelas **DatObjekPajakService**

Pada pengujian kali ini, yang digunakan adalah 1 (satu) buah contoh data berdasarkan Nomor Objek Pajak (NOP), nantinya data yang dikembalikan dari sistem basis data harus memenuhi nilai yang ditentukan. *Request* yang dilakukan adalah terhadap Nomor Objek Pajak (NOP) 332901000100500390, hasil yang diharapkan adalah sebuah objek **DatObjekPajakModif** dengan nilai berikut :

- Atribut **op\_kec** seharusnya bernilai **SALEM** dengan tipe data **string**
- Atribut **op\_kel** seharusnya bernilai **GUNUNGJAYA** dengan tipe data **string**
- Atribut **op\_jalan** seharusnya bernilai **KP RW 01** dengan tipe data **string**

- Atribut `op_rtrw` seharusnya bernilai RT. 002 RW. 01 dengan tipe data `string`
- Atribut `op_luas_bumi` seharusnya bernilai 80 dengan tipe data `BigDecimal`
- Atribut `op_luas_bng` seharusnya bernilai 40 dengan tipe data `BigDecimal`
- Atribut `op_njop_bumi` seharusnya bernilai 1600000 dengan tipe data `BigDecimal`
- Atribut `op_njop_bng` seharusnya bernilai 9000000 dengan tipe data `BigDecimal`
- Atribut `op_wp_id` seharusnya bernilai 332901000100500390 dengan tipe data `string`

#### **1.4.2 Pengujian Terhadap Kelas `DatSubjekPajakService`**

Pengujian terhadap kelas `DatSubjekPajak` pun akan dilakukan dengan sebuah parameter berupa nomor identitas subjek pajak dengan nilai 332901000100500390. Hasil yang seharusnya didapatkan adalah objek dari kelas `DatSubjekPajakModif` dengan rincian nilai atas atribut seperti berikut ini :

- Atribut `wp_nama` dengan isi KARSO dengan tipe data `string`
- Atribut `wp_jalan` dengan isi KP RW 01 dengan tipe data `string`
- Atribut `wp_rtrw` dengan isi RT. 002 RW. 01 dengan tipe data `string`
- Atribut `wp_kel` dengan isi GUNUNGJAYA dengan tipe data `string`
- Atribut `wp_kota` dengan isi KAB BREBES dengan tipe data `string`

### 1.4.3 Pengujian Terhadap Kelas SpptService

Untuk pengujian kelas `SpptService`, yang akan diuji adalah hasil dari *method* `getSppt()` dan `getPiutang()`

Pada saat pemanggilan *method* `getSppt()` dengan menggunakan parameter Nomor Objek Pajak (NOP) 332901000100500390, maka *method* tersebut akan mengembalikan nilai berupa daftar dengan tipe data `List`, jumlah data yang terdapat pada `List` ini seharusnya ada 8 (delapan) buah data.

Kemudian pada saat pemanggilan *method* `getPiutang()` untuk dilakukan uji coba, parameter yang dilewatkan adalah berupa Nomor Objek Pajak (NOP) dengan isi 332911090400000967, kemudian parameter kedua diisi dengan tahun pajak, yaitu 1994.

Diharapkan hasil dari operasi di atas adalah data 2088 dengan tipe data berupa `BigDecimal`.

## **Bab 2**

# **PETUGAS VERIFIKASI ATAU VALIDASI PROGRAM**

Karena pengujian dilakukan secara paralel pada saat pengembangan aplikasi atau program, dan dilakukan secara otomatis dengan kode program, maka kebutuhan untuk petugas verifikasi atau validasi program tidak diperlukan lagi untuk pengembangan sistem informasi atau aplikasi ini.



## Bab 3

# JADWAL PELAKSANAAN

Jadwal pelaksanaan pengujian akan dilakukan bersamaan pada saat melakukan *coding* program, sehingga dalam siklusnya, saat selesai mengerjakan sebuah fitur dalam program, kode otomasi untuk pengujian akan dibuatkan juga untuk memastikan bahwa program atau aplikasi yang akan di kompilasi telah lulus dari pengujian.