

STUDI KELAYAKAN RINCI PEMBANGUNAN LAYANAN SISTEM OTENTIKASI DI BADAN PENGELOLAAN PENDAPATAN, KEUANGAN DAN ASET DAERAH KABUPATEN BREBES

22 Februari 2019

Priyanto Tamami, S.Kom.

1 RUANG LINGKUP PEKERJAAN

Ruang lingkup dari pekerjaan membangun sebuah sistem informasi otentikasi di Badan Pengelolaan Pendapatan, Keuangan dan Aset Daerah Kabupaten Brebes yaitu membangun sebuah sistem yang mampu melayani otentikasi termasuk otorisasi di dalamnya, untuk seluruh aplikasi yang berkaitan dengan Pajak Daerah di Kabupaten Brebes, adapun aplikasi atau sistem informasi yang akan dibangun berbasis *web*, artinya untuk menggunakan aplikasi atau sistem informasi ini dibutuhkan *browser* untuk menampilkan halaman sistem informasi.

Sistem informasi yang akan dibangun akan memiliki arsitektur dimana aplikasi yang dibangun akan dipisahkan menjadi 3 (tiga) bagian, yaitu bagian *front-end* dan 2 (dua) bagian *back-end*, bagian *back-end* akan terdiri dari layanan sistem yang menyediakan otentikasi dan otorisasi itu sendiri, yang disebut *Auth Server*, dan bagian lain akan melayani data yang digunakan untuk melakukan otentikasi dan otorisasi seperti informasi pengguna, informasi hak akses, dan informasi mengenai aplikasi klien yang dapat terhubung. Bagian *front-end* dan bagian *back-end* akan berkomunikasi melalui cara *web service* dengan model REST. Data yang dikirim dari peladen *web service* berupa JSON.

Pemilihan arsitektur ini karena nantinya aplikasi klien yang dapat terhubung

dan menggunakan fasilitas otorisasi dan otentikasi bukan hanya aplikasi yang berbasis *web*, namun termasuk juga aplikasi *mobile* yang terpasang dalam ponsel pintar dan aplikasi yang berbasis *desktop*.

Komunikasi data yang sifatnya universal dan mampu menghubungkan ketiga perangkat tersebut, yaitu peladen, klien berbasis aplikasi *mobile* dan aplikasi berbasis *desktop* adalah menggunakan arsitektur REST.

Bagian *front-end* nantinya akan memiliki fungsi sebagai tampilan tatap muka untuk melakukan konfigurasi untuk tiap pengguna dan konfigurasi untuk tiap klien yang nantinya akan dihubungkan. Bagian *front-end* nantinya akan terhubung ke 2 (dua) layanan di bagian *back-end*, layanan yang pertama (yang disebut dengan *OAuth Server*) tentu saja akan melakukan otorisasi dan otentikasi terhadap pengguna yang akan melakukan akses, layanan ini hanya menyediakan halaman masuk (*login*) saja, yang kemudian, setiap aplikasi klien akan mendapatkan sebuah token yang digunakan untuk melakukan akses ke layanan yang kedua (yang diistilahkan sebagai *resource server*). *Resource server* inilah yang nantinya akan berkomunikasi dengan bagian *front-end* (diistilahkan juga sebagai aplikasi klien) dalam penyediaan datanya.

Untuk memastikan bahwa layanan pada *Resource server* terlindungi, tiap permintaan (*request*) yang dilakukan oleh aplikasi klien harus menyertakan token yang didapat dari proses otentikasi dengan *OAuth Server*, nantinya token yang diterima oleh *Resource server* dari aplikasi klien akan diverifikasi dengan melakukan verifikasi token.

Cara yang dilakukan *Resource server* untuk memverifikasi token dari aplikasi klien adalah dengan melakukan dekripsi token menggunakan *public key* yang disediakan oleh *OAuth Server* yang dapat diakses secara bebas.

Token yang diimplementasikan sebagai tanda pengenal bahwa pengguna memiliki hak akses tertentu akan berbentuk JWT (*JSON Web Token*) yang dijabarkan

pada dokumen IETF (*Internet Engineering Task Force*) dengan nomor RFC (*Request for Comment*) 7519 yang telah diperbarui dengan dokumen IETF dengan nomor RFC 7797.

Sebagaimana dijelaskan dalam dokumen tersebut (IETF RFC No. 7519), bahwa JWT sebetulnya hanyalah sebuah teks (*string*) yang mewakili beberapa pernyataan dari objek JSON yang terkode. Pernyataan dalam objek JSON sendiri adalah pasangan antara *key* dan *value*, dimana *key* harus berupa teks (*string*) dan *value* dapat berupa nilai apapun yang dapat didefinisikan dalam JSON.

JSON sendiri sebetulnya adalah singkatan dari *JavaScript Object Notation*, sebagaimana didefinisikan pada dokumen IETF (*Internet Engineering Task Force*) dengan nomor RFC (*Request for Comment*) 8259, bahwa JSON adalah sebuah format pertukaran data yang ringan, berbasis teks, dan independen terhadap bahasa pemrograman.

JSON sendiri dapat merepresentasikan 4 (empat) tipe data primitif seperti *string*, *number*, *boolean*, dan *null*, serta 2 (dua) tipe struktur, yaitu objek dan larik (*array*).

String sendiri adalah barisan dari 0 (nol) atau lebih karakter *Unicode* yang teruang pada laman <http://www.unicode.org/versions/latest/>. Harap dicatat bahwa petikan tersebut mengacu pada versi paling akhir dari *Unicode* tanpa menyebutkan secara spesifik versinya. Karena hal ini, sehingga tidak menjamin bahwa perubahan yang terjadi pada spesifikasi *Unicode* dimasa mendatang akan memiliki pengaruh terhadap sintak dari JSON.

Bagian *Back-end* atau bisa disebut *server web services* yang nantinya akan melayani *request* data baik yang datang dari aplikasi *mobile* atau dari bagian *front-end*. Bagian *back-end* ini akan dibangun dengan menggunakan bahasa Java dengan penggunaan *framework* Spring Boot agar lebih memudahkan proses pembangunannya.

Hal-hal yang mendukung studi kelayakan dibangunnya sistem informasi untuk Pajak Bumi dan Bangunan sektor Perdesaan dan Perkotaan ini yaitu dengan menggunakan metode analisis kelayakan TELOS (*Technical, Economic, Legal, Operational, Schedule*).

Detail dari masing-masing faktor pada metode analisis kelayakan TELOS ini adalah seperti berikut :

1. Faktor Kelayakan Teknis (*Technical*)

Kelayakan teknis menyoroti kebutuhan sistem yang telah disusun dari aspek teknologi yang digunakan, jika teknologi yang dikehendaki untuk pengembangan sistem merupakan teknologi yang mudah didapat, murah, dan tingkat pemakaiannya mudah, maka secara teknis usulan kebutuhan sistem bisa dinyatakan layak

2. Faktor Kelayakan Ekonomi (*Economic*)

Aspek yang paling dominan dari aspek kelayakan yang lain adalah kelayakan ekonomi. Tidak dapat disangkal lagi, motivasi pengembangan sistem informasi pada perusahaan atau organisasi adalah motif keuntungan. Dengan demikian aspek untung rugi jadi pertimbangan utama dalam pengembangan sistem. Kelayakan ekonomi berhubungan dengan *return investmen* atau berapa lama biaya investasi dapat kembali, namun pada instansi pemerintah, yang secara umum bertugas melayani masyarakat, maka hubungannya bukan lagi untung rugi, tetapi lebih kepada imbas administrasi data yang tercatat dengan baik dan sedikit banyak mungkin berpengaruh pada peningkatan kepatuhan wajib pajak dalam memenuhi kewajiban pajaknya.

3. Faktor Kelayakan Hukum (*Legal*)

Menguraikan secara hukum apakah sistem informasi yang akan dikembangkan tidak menyimpang dari hukum yang berlaku (tidak melanggar hukum jika

diterapkan di objek penelitian). Misal: bagaimana kelayakan perangkat lunak yang digunakan, bagaimana kelayakan hukum informasi yang dihasilkan oleh program aplikasi yang dibuat. Apakah melanggar hukum atau tidak.

4. Faktor Kelayakan Operasional (*Operational*)

Penilaian terhadap kelayakan operasional digunakan untuk mengukur apakah sistem yang akan dikembangkan nantinya dapat dioperasikan dengan baik atau tidak untuk melayani masyarakat wajib pajak.

5. Faktor Kelayakan Jadwal *Schedule*)

Penilaian kelayakan jadwal ini digunakan untuk menentukan bahwa pengembangan sistem akan dapat dilakukan dalam batas waktu yang telah ditetapkan.

Adapun penilaian untuk masing-masing faktor kelayakan TELOS ini adalah sebagai berikut :

1. Menilai kelayakan Teknik (*Technical*)

Kelayakan teknik menyoroti kebutuhan sistem yang telah disusun dari teknologi yang akan digunakan, untuk penerapan sistem informasi pembayaran untuk jenis pajak bumi dan bangunan sektor perdesaan dan perkotaan. Sistem informasi ini merupakan sistem berbasis *web* yang digunakan oleh masyarakat wajib pajak melakukan verifikasi atau pengecekan pembayaran atas pajak terhutang. Sistem informasi dibangun berbasis *web*, dengan tujuan agar seluruh wajib pajak dapat mengakses sistem informasi ini hanya dengan berbekal koneksi jaringan internet dan aplikasi *browser* seperti Chrome, Firefox, Internet Explorer, Safari, atau jenis *web browser* lainnya.

Sistem ini pun tidak memerlukan banyak *input*, pengguna cukup memberikan informasi Nomor Objek Pajak pada kolom yang telah disediakan, sehingga

target pengguna yang diharapkan yaitu pengguna yang tidak memiliki keahlian khusus komputer pun dapat mengoperasikan sistem ini.

(a) Kebutuhan Perangkat Keras, Perangkat Lunak, dan Perangkat Jaringan

i. Perangkat Keras

Kebutuhan akan perangkat keras, sebetulnya tidak diperlukan pengadaan alat baru, melainkan dengan menggunakan perangkat yang sudah ada, sistem informasi ini akan dapat berjalan sebagaimana mestinya. Adapun perangkat keras yang digunakan adalah seperti berikut :

No	Perangkat keras	Keterangan
<i>Server Aplikasi</i>		
1	Prosesor	Intel Xeon 2,4G 8 Core
2	Memori	8 GB
3	Harddisk	
4	Jaringan	4 ethernet slot
<i>Server Basis Data</i>		
1	Prosesor	Intel Xeon 2,4G 8 Core
2	Memori	32 GB
3	Harddisk	
4	Jaringan	4 ethernet slot

ii. Perangkat Lunak

Kebutuhan akan perangkat lunak baru tentunya ada, tetapi tidak memerlukan biaya tambahan, karena beberapa perangkat lunak ada yang sudah terpasang, dan perangkat lunak yang akan diinstall tidak memerlukan biaya tambahan.

No	Perangkat Lunak	Kegunaan
1	Windows Server 2008	Sistem Operasi
2	Linux Ubuntu	Sistem Operasi
3	IntelliJ IDEA	IDE
4	Visual Studio Code	Editor
5	Chrome, Firefox	Web Browser

iii. Perangkat Jaringan

Kebutuhan akan perangkat jaringan pun tidak memerlukan pengadaan kembali, karena kondisi jaringan sudah terhubung dengan baik dengan peralatan seperti berikut :

No	Perangkat	Kegunaan
1	Switch	Penghubung antar kabel jaringan
2	Kabel UTP	Media penghubung antar perangkat
3	Konektor RJ-45	Media penghubung antara kabel dengan <i>socket</i> perangkat
4	Modem	Sebagai perangkat penghubung jaringan lokal dan internet
5	Router	Penghubung antar 2 atau lebih jaringan (berdasarkan alamat jaringan)

(b) Arsitektur Jaringan Komputer

Arsitektur jaringan sistem informasi ini adalah seperti pada gambar

1. Bila ada seorang pengguna yang melakukan akses dengan *browser*, maka akan ditampilkan halaman utama dari sistem informasi ini. Akses ini oleh *router* akan diarahkan ke *server* aplikasi.

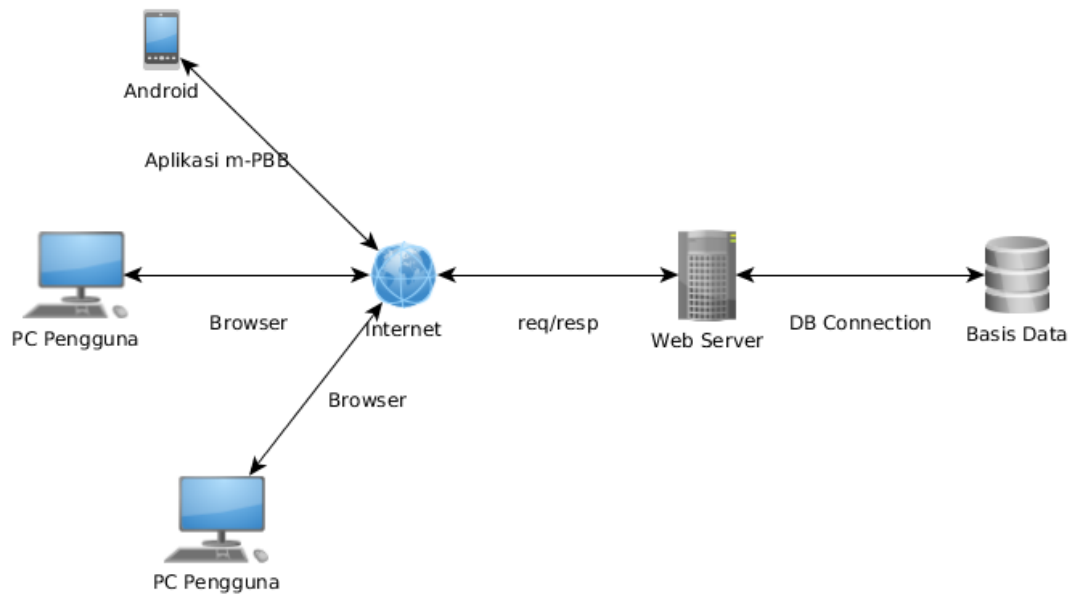


Figure 1: Desain Arsitektur Sistem

Kemudian saat pengguna melakukan *view* data terhadap nomor objek pajak tertentu, maka aplikasi akan melakukan *request* data ke bagian *back-end* yang selanjutnya bagian *back-end* akan melakukan *query* data ke sistem basis data. Setelah itu hasil yang didapatkan dari sistem basis data akan dikembalikan ke bagian *front-end* untuk di olah dalam tampilan halaman *web*.

Adapun kondisi sistem informasi yang digunakan saat ini adalah seperti berikut :

(a) Sistem Aplikasi

Aplikasi adalah sebuah *tools* untuk membantu dan mempermudah suatu aktifitas administrasi maupun yang lainnya. Tanpa adanya aplikasi, tentu suatu aktifitas tidak bisa diselesaikan dengan cepat, segala sesu-

atunya dilakukan dengan pencatatan manual. Aplikasi atau sistem informasi yang digunakan di Bidang Pengelolaan Pendapatan, Keuangan dan Aset Daerah Kabupaten Brebes saat ini adalah :

No	Aplikasi	Keterangan
1	SISMIOP	Melakukan manajemen objek pajak PBB-P2
2	Microsoft Excel	Digunakan untuk mengolah data peralihan hak

Dengan adanya sistem informasi atau aplikasi tersebut, sudah membantu dari mulai manajemen pelayanan pajak bumi dan bangunan sektor perdesaan dan perkotaan, penetapan, penagihan, hingga ke prosedur pembayaran, namun demikian, sistem yang ada tidak memungkinkan informasi pembayarannya diakses oleh publik, maka dari itu dibutuhkan sistem atau aplikasi lain agar masyarakat wajib pajak mengetahui status pembayarannya, dengan tanpa melakukan prosedur instalasi apapun.

(b) Sistem Basis Data

Basis data merupakan suatu tempat penyimpanan data-data yang biasanya dari sebuah aplikasi, adapun basis data yang sudah ada pada Badan Pengelolaan Pendapatan, Keuangan dan Aset Daerah adalah seperti berikut :

No	Basis data	Keterangan
1	Oracle DB	Digunakan untuk menyimpan data dari aplikasi SISMIOP
2	SQL Server	Digunakan untuk menyimpan data dari aplikasi Simda Pendapatan

Basis data yang digunakan masih belum terintegrasi dengan baik, karena penerimaan pembayaran pada SISMIOP belum dapat secara otomatis tercatat pada Simda Pendapatan.

Sebetulnya sumber data yang digunakan pada pengelolaan Pajak Bumi dan Bangunan sektor Perdesaan dan Perkotaan ada pada 2 (dua) tempat, yang pertama berada di Badan Pengelolaan Pendapatan, Keuangan dan Aset Daerah Kabupaten Brebes, yang kedua berada di BPD Jateng.

Kondisi data yang berada di Badan Pengelolaan Pendapatan, Keuangan dan Aset Daerah berfungsi sebagai pendataan dan penetapan, namun realisasi pembayaran menggunakan data dari BPD Jawa Tengah. Selisih nilai antara 2 (dua) sumber data ini pasti terjadi, maka seharusnya dibutuhkan personil untuk melakukan pencocokan / rekonsiliasi data antara 2 (dua) sumber data tersebut agar data yang ditampilkan pada sistem informasi yang akan dibangun menjadi lebih lengkap.

(c) Infrastruktur

Infrastruktur merupakan sarana yang sangat dibutuhkan dalam sebuah aktifitas apapun termasuk aktifitas pengelolaan Pajak Daerah. Adapun infrastruktur yang ada pada Badan Pengelolaan Pendapatan, Keuangan dan Aset Daerah Bidang Pendataan dan Penetapan yaitu :

No	Infrastruktur	Keterangan
1	Komputer	Memiliki 13 komputer dan 4 <i>server</i>
2	Printer	Memiliki 4 <i>printer multi-purpose</i> , 4 <i>printer dot-matrix</i> , 3 <i>printer line-matrix</i>
3	Jaringan Internet	Memiliki 2 (dua) sumber internet, dengan 1 (satu) internet <i>shared bandwidth</i> dan 1 (satu) internet <i>dedicated</i>

Dari kondisi infrastruktur di atas, sebetulnya Badan Pengelolaan Pendapatan, Keuangan dan Aset Daerah cukup untuk membuka atau melayani

sebuah sistem informasi status pembayaran Pajak Bumi dan Bangunan Sektor Perdesaan dan Perkotaan yang akan dibangun.

Melihat kondisi teknis demikian, dimana sistem belum pernah dibuat, teknologi yang digunakan terbilang cukup baru, sehingga nilai kelayakan teknik yang dapat diberikan adalah 9.0.

2. Menilai kelayakan Ekonomi (*Economic*)

Pembangunan sistem baru tentunya membutuhkan investasi ataupun dana yang tidak sedikit, untuk mendapatkan manfaat dimasa yang akan datang, sumber daya dan sumber dana diperlukan dalam pembangunan sistem baru sebagai bentuk investasi.

Untuk menganalisis kelayakan ekonomi digunakan kalkulasi analisis biaya dan manfaat (*cost benefit analysis*). Adapun tujuan dari analisis biaya dan manfaat adalah untuk memberikan gambaran kepada internal Badan Pengelolaan Pendapatan, Keuangan dan Aset Daerah, apakah manfaat yang diperoleh dari sistem baru "lebih besar" dibandingkan dengan biaya yang dikeluarkan.

Pada analisis biaya dan manfaat ada beberapa metode kuantitatif yang digunakan untuk menemukan standar kelayakan proyek.

Analisis Biaya dan Manfaat

Untuk melakukan analisa biaya dan manfaat diperlukan dua komponen, yaitu komponen biaya dan komponen manfaat.

(a) Komponen Biaya

Biaya yang berhubungan dengan pembuatan sistem ini dapat diklasifikasikan ke dalam 3 (tiga) kategori utama, yaitu :

- i. Biaya pengadaan, yaitu biaya pembelian perangkat keras, yang digunakan pada awal pembuatan sistem, sebelum sistem dioperasikan.
- ii. Biaya Pengembangan, yaitu biaya pembuatan perangkat lunak sistem yang meliputi biaya konsultasi, biaya tahap analisa sistem, biaya tahap desain sistem dan biaya tahap penerapan sistem.
- iii. Biaya operasi dan biaya perawatan, yaitu biaya yang dikeluarkan untuk menjalankan sistem, yaitu biaya *overhead*, biaya perawatan terhadap perangkat keras dan perangkat lunak.

(b) Komponen Manfaat.

Manfaat yang didapat dari sistem informasi diklasifikasi sebagai berikut :

- i. Keuntungan berwujud adalah keuntungan yang berupa penghematan atau peningkatan di dalam pengelolaan pajak daerah yang dapat diukur dalam bentuk satuan nilai uang. Keuntungan berwujud antara lain :
 - A. Pengurangan biaya cetak
 - B. Pengurangan biaya operasi
 - C. Pengurangan biaya perlengkapan
- ii. Keuntungan tak berwujud, adalah keuntungan yang sulit atau tidak mungkin diukur dalam bentuk satuan uang. Keuntungan tersebut antara lain :
 - A. Peningkatan kepatuhan wajib pajak
 - B. Peningkatan efektifitas pegawai
 - C. Peningkatan kepuasan masyarakat

Adapun metode untuk melakukan analisis biaya dan manfaat adalah :

i. Metode Periode Pengembalian

Metode ini adalah uji kuantitatif yang digunakan untuk menghitung jangka waktu yang diperlukan untuk membayar kembali biaya investasi dalam pembuatan aplikasi yang telah dikeluarkan. Penilaian kelayakan untuk pengembalian

$$periode = \frac{investasi}{proceed} \times tahun$$

- A. Layak jika waktu pengembalian lebih kecil dari umur investasi
- B. Tidak layak jika waktu pengembalian lebih besar dari umur investasi.

Perhitungan Periode Pengembalian adalah seperti berikut :

Nilai Investasi = Rp0,- Proses Th 1 = Rp30.000.000.000,-

Nilai investasi memang bernilai 0 (nol) rupiah karena seluruh sarana dan prasarana telah tersedia, hanya tinggal membangun sebuah sistem informasi untuk digunakan dan dijalankan. Sedangkan nilai Rp30.000.000.000,- (Tiga puluh milyar) didapat dari nilai target penerimaan jenis pajak bumi dan bangunan sektor perdesaan dan perkotaan.

$$PP = \frac{0}{30.000.000.000}$$

$$PP = 0 \text{ tahun}$$

Artinya, karena nilai investasi yang dikeluarkan nihil sama sekali, atau dengan kata lain tidak memerlukan nilai investasi, namun sistem informasi secara tidak langsung memberikan andil terhadap realisasi sebesar 30 Milyar Rupiah kepada Kas Daerah. Artinya sistem ini sangat layak untuk dikembangkan karena pengembalian tidak membutuhkan waktu untuk mencapai titik impas.

ii. Metode Pengembalian Investasi

Metode pengembalian investasi digunakan untuk mengukur persentase manfaat yang dihasilkan proyek dibandingkan dengan biaya yang dikeluarkan.

Return On Investmen (ROI) dari suatu proyek dapat dihitung dengan rumus penilaian kelayakan untuk ROI :

A. Layak jika $ROI > 0$

B. Tidak layak jika $ROI < 0$

$$ROI = \frac{TotalManfaat - TotalBiaya}{TotalBiaya} \times 100\%$$

Karena nilai biaya tidak akan pernah muncul, maka bisa disebut bahwa Total Biaya = 0 (nol), Total Manfaat adalah penerimaan realisasi Pajak Bumi dan Bangunan sektor Perdesaan dan Perkotaan dalam satu tahun pajak, yaitu senilai 30 Milyar rupiah. Maka hasil yang didapatkan adalah seperti berikut :

$$ROI = \frac{30M - 0}{0} \times 100\%$$

$$ROI = \infty$$

Ternyata karena kondisi pembagiya adalah 0 (nol), maka hasil yang didapatkan menghasilkan Tak Terhingga. Ini dikarenakan bilangan pembagi adalah 0 (nol) sehingga menghasilkan nilai demikian, artinya proyek apapun itu bila tidak memerlukan biaya sama sekali, namun bisa mendapatkan manfaat maksimal tentunya ini proyek dengan ROI tak terhingga.

Jadi, karena nilai ∞ itu lebih dari 0 (nol) maka proyek ini layak untuk dikerjakan.

iii. Metode Nilai Sekarang Bersih

Metode nilai sekarang bersih merupakan metode yang memperhatikan nilai waktu dari uang. Suku bunga diskonto mempengaruhi arus dari uangnya (*proceed*). Metode Nilai Sekarang Bersih atau lebih dikenal dengan *Net Present Value* (NPV) dapat dihitung dari selisih nilai proyek pada awal tahun dikurangi dengan *proceed* tiap tahun yang dinilai uangkan ketahun awal dengan tingkat bunga diskonto.

Kriteria NPV ini adalah seperti berikut :

- $NPV > 0$: *Feasible*
- $NPV = 0$: *Indifferent*
- $NPV < 0$: *Unfesible*

Rumus untuk menghitung NPV ini adalah seperti berikut :

$$NPV = -NilaiProyek + \frac{proceed1}{(1+i)^1} + \frac{proceed2}{(1+i)^2} + \frac{proceedn}{(1+i)^n}$$

dimana :

- i adalah tingkat bunga diskonto diperhitungkan
- n adalah umur proyek investasi
- *proceed* adalah selisih biaya dan manfaat

Sehingga perhitungannya dapat kita cari seperti berikut :

$$NPV = -0 + \frac{30M}{(1+6.5\%)^1}$$

$$NPV = -0 + \frac{30M}{1.065}$$

$$NPV = 28.169.014.084.51$$

Pada perhitungan di atas, nilai waktu dari bunga uang yang ditanamkan adalah 6,5% berdasarkan suku bunga dari www.bi.go.id pada tanggal 21 Juli 2016. Karena nilai NPV > 0 berarti investasi dapat diterima dan layak untuk dikembangkan.

Dengan melihat ketiga metode untuk kelayakan ekonomi maka dapat dikatakan bahwa sistem informasi yang akan dibangun layak dan dapat dikerjakan sebagaimana mestinya. Nilai yang diberikan untuk kelayakan ekonomi ini adalah 9,0.

3. Menilai kelayakan Hukum (*Legal*)

Kelayakan hukum adalah kelayakan yang berkaitan dengan legalitas atau kekuatan hukum. Berarti bahwa sistem informasi yang akan dibangun tidak boleh melanggar hukum yang berlaku, baik hukum yang ditetapkan oleh pemerintah maupun hukum yang ditetapkan berdasarkan peraturan-peraturan organisasi.

Proyek pembangunan sistem informasi yang akan dikembangkan secara hukum dinilai layak karena perangkat lunak (*software*) yang digunakan resmi sesuai dengan perijinan yang ada, karena sebagian besar menggunakan perangkat lunak yang bersifat *free* atau gratis.

Adapun rincian perangkat lunak secara hukum adalah sebagai berikut :

No	<i>Software</i>	Lisensi
1	Windows Server 2008 R2	Lisensi tersedia
2	Linux Ubuntu	Gratis
3	IntelliJ IDEA Community Edition	Gratis, Apache 2.0
4	Visual Studio Code	Gratis, MIT

Karena sistem informasi yang akan dibangun tidak meliputi data sensitif, seperti data detail wajib pajak, dan pada saat melakukan akses diperlukan Nomor Objek Pajak yang hanya wajib pajak saja yang memiliki, sehingga nilai untuk kelayakan hukum ini adalah 9,0.

4. Menilai kelayakan Operasional (*Operational*)

Kelayakan operasional dinilai dengan menggunakan kerangka kerja PIECES yang merupakan singkatan dari *Performance, Information, Economy, Control, Efficiency*, dan *Services*. Kerangka kerja PIECES ini dikembangkan oleh James Wetherbe yang bertujuan mengukur apakah sistem yang akan dikembangkan dapat dioperasikan dengan baik atau tidak di dalam organisasi.

Penjelasan untuk masing-masing poin dalam kerangka kerja PIECES ini adalah seperti berikut ini :

- *Performance* atau kinerja adalah untuk mengetahui apakah sistem menyediakan *throughput* dan *response time* yang cukup. Perbandingannya adalah seperti berikut ini :

Sistem Lama	Sistem Baru
Waktu yang dibutuhkan untuk melakukan pengambilan data pembayaran untuk tiap Nomor Objek Pajak membutuhkan waktu cukup lama karena banyaknya menu	Waktu yang dibutuhkan untuk mencari informasi objek pajak berdasarkan Nomor Objek Pajak relatif singkat.

- *Information* atau informasi adalah untuk mengetahui apakah sistem menyediakan informasi yang berkualitas bagi pengguna.

Sistem Lama	Sistem Baru
Informasi yang disajikan hanya dapat diakses oleh pegawai internal dan melalui aplikasi <i>desktop</i>	Informasi dapat diakses oleh siapa saja pemegang Nomor Objek Pajak

Sistem Lama	Sistem Baru
Biaya yang dikeluarkan tinggi, karena terkadang wajib pajak meminta versi cetak-nya	Biaya yang dikeluarkan relatif rendah, karena wajib pajak dapat melakukan akses mandiri terhadap datanya.

- *Economy* atau ekonomi adalah untuk mengetahui apakah sistem menawarkan tingkat dan kapasitas pelayanan yang memadai untuk mengurangi biaya dan meningkatkan keuntungan. Berikut adalah perbandingan dari sistem lama dan baru :
- *Control* atau pengendalian adalah untuk mengetahui apakah sistem informasi yang dibangun menawarkan kontrol atau pengendalian untuk mengatasi kecurangan-kecurangan dan untuk menjamin keakuratan dan keamanan data. Berikut adalah perbandingan sistem yang lama dan sistem yang baru :

Sistem Lama	Sistem Baru
Hanya dapat diakses oleh pengguna yang terdaftar pada sistem informasi SISMIOP	Data dapat diakses oleh siapapun yang mengetahui Nomor Objek Pajak yang akan dicari

Namun akses datanya hanya dapat dilihat saja, tidak dapat merubah data yang ditampilkan.

- *Efficiency* atau efisiensi adalah untuk mengetahui apakah sistem meng-

gunakan secara maksimum sumber yang tersedia termasuk orang, waktu aliran *form*, meminimalkan penundaan proses. Berikut adalah perbandingan sistem informasi yang lama dan sistem informasi yang baru :

Sistem Lama	Sistem Baru
Wajib pajak harus datang ke kantor hanya untuk mengetahui status pembayaran atau perubahan nama pada Surat Pemberitahuan Pajak Terhutangnya	Wajib pajak dapat melakukan pemeriksaan secara mandiri terhadap data miliknya berdasarkan Nomor Objek Pajak.

- *Services* atau pelayanan adalah untuk mengetahui apakah sistem menyediakan layanan yang diinginkan dan handal pada siapa saja yang menginginkannya, dan apakah sistem fleksibel dan dapat dikembangkan. Berikut adalah perbandingannya antara sistem lama dan sistem baru :

Sistem Lama	Sistem Baru
Pelayanan sangat membebani wajib pajak yang harus datang ke kantor dan mengantri bila ada antrian hanya untuk mengetahui data pembayarannya	Wajib pajak cukup melakukan akses dari mana pun berdasarkan Nomor Objek Pajak untuk mengetahui data pembayarannya

Karena sistem berbasis *web* dan banyak pengguna, serta tidak memerlukan keahlian khusus untuk melakukan operasinya. Maka nilainya kita berikan 7,0.

5. Menilai kelayakan Jadwal (*Schedule*)

Kelayakan jadwal digunakan untuk menentukan bahwa pengembangan sistem dapat dilakukan dalam batas waktu yang telah ditetapkan. Pengembangan sistem direncanakan selesai dalam waktu maksimal ± 9 minggu. Adapun perkiraan tahap-tahap pengembangan sistem dijadwalkan sebagai berikut :

Dalam proyek pengembangan sistem informasi Pembayaran Pajak Bumi dan Bangunan Sektor Perdesaan dan Perkotaan dilakukan dalam 9 (sembilan) tahap, yaitu :

- (a) Tahap analisa kebutuhan sistem
- (b) Tahap studi kelayakan
- (c) Tahap desain *user interface*
- (d) Tahap desain data
- (e) Tahap desain *proses*
- (f) Tahap Pengodean
- (g) Tahap persiapan tempat *instalasi*
- (h) Tahap *instalasi* perangkat keras dan perangkat lunak
- (i) Tahap uji sistem
- (j) Tahap dokumentasi

Grafik *Gantt* untuk tahapan-tahapan tersebut adalah seperti pada gambar 2 berikut ini :

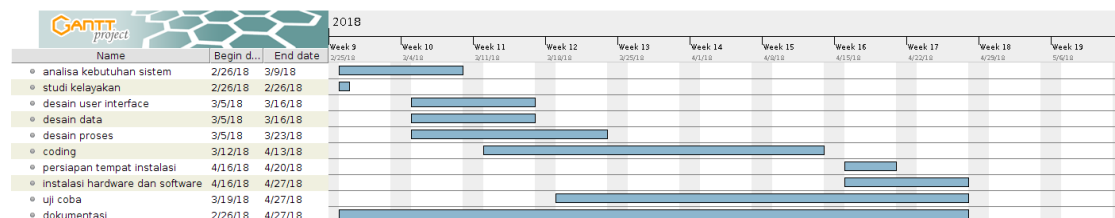


Figure 2: Grafik *Gantt*

Karena pengembangan diukur dalam jam, hari, minggu, dan bulan maka kesalahan perkiraan (*error estimation*) yang dibutuhkan untuk perancangan dan implementasi menjadi kecil. Maka nilainya 8.0

Dari keseluruhan faktor kelayakan TELOS tersebut, jumlah dari semua faktornya adalah

$$Total = 9 + 9 + 9 + 7 + 8$$

$$Total = 44$$

Total score dari kelayakan ini adalah :

$$score = 44/5 = 8,8$$

Artinya perancangan pengembangan sistem informasi yang dievaluasi adalah LAYAK dengan resiko pengembangan sistem informasi yang cukup rendah.

2 SARANA DAN PRASARANA YANG MELIPUTI PERANGKAT KERAS DAN PERANGKAT LUNAK YANG DIPERLUKAN

Kebutuhan akan sarana dan prasarana untuk jenis perangkat keras yaitu :

1. Peladen (*server*) Basis Data

Peladen atau *server* ini berfungsi untuk menyimpan basis data hasil proses penetapan yang menghasilkan nilai pajak terhutang untuk pajak bumi dan bangunan sektor perdesaan dan perkotaan, untuk nantinya dapat diakses

oleh peladen (*server*) aplikasi *web* yang menyediakan sebuah API (*Application Programming Interface*) untuk diakses oleh aplikasi *web* dan aplikasi *mobile* yang juga akan dibangun.

Kebutuhan perangkat keras peladen ini tidak perlu membeli karena nantinya akan menggunakan peladen yang sudah ada sebagai peladen basis data untuk aplikasi SISMIOP (Sistem Manajemen Informasi Objek Pajak) yang sudah beroperasi.

2. Peladen (*server*) *web* aplikasi

Peladen ini akan digunakan sebagai tempatnya aplikasi *web* yang menyediakan API (*Application Programming Interface*) untuk aplikasi *web* Angular dan aplikasi *mobile*.

Peladen ini juga akan menjadi tempat aplikasi *web* bagian ujung depan (*front-end*) berada. Aplikasi ujung depan (*front-end*) dengan aplikasi ujung belakang (*back-end*) yang bisa kita sebut peladen API (*Application Programming Interface*) adalah 2 (dua) aplikasi yang berbeda fungsi, berbeda bahasa pemrograman, yang dibangun secara mandiri tetapi saling berkomunikasi dalam sebuah sistem.

Peladen ini pun tidak perlu membeli karena nantinya akan menggunakan peladen yang sudah disediakan sebagai peladen aplikasi *web*.

3. Kartu Jaringan

Kartu jaringan digunakan atau terpasang pada peladen agar peladen dapat terhubung dengan jaringan, baik jaringan LAN (*Local Area Network*) maupun jaringan *internet*.

Kartu jaringan ini tidak perlu membeli karena sudah terpasang lebih dari 1 (satu) pada tiap peladen yang dapat dikonfigurasi secara mandiri seluruhnya.

4. Kabel *Unshielded Twisted Pair* (UTP)

Kabel ini terdiri dari 4 (empat) pasang warna yang digunakan sebagai penghubung atau media komunikasi antar kartu jaringan yang terdapat pada tiap peladen agar aplikasi atau data pada peladen dapat diakses melalui jaringan.

Kabel ini juga tidak perlu membeli karena telah terpasang dan saling terhubung dari tiap peladen ke area jaringan.

5. Konektor RJ45

Konektor ini digunakan atau dipasangkan pada kabel UTP agar kabel UTP dapat dipasangkan pada kartu jaringan di tiap peladen dan tiap perangkat jaringan seperti *switch*, *router*, ataupun *modem*.

Masing-masing helai kabel UTP akan terpasang pada tiap pin yang ada pada konektor RJ45, yang kemudian apabila konektor RJ45 dipasangkan pada kartu jaringan, atau slot RJ45 yang terdapat pada *switch*, *router* ataupun *modem*, akan terhubung dengan pin pada RJ45 sehingga aliran data dapat disampaikan dari 1 (satu) peladen ke perangkat jaringan yang lain.

6. *Switch*

Masing-masing komponen yang terkoneksi / terhubung ke jaringan dapat diistilahkan sebagai *host*, tentunya untuk membuat agar 1 (satu) *host* terhubung dengan 1 (satu) *host* yang lain maka diperlukan sebuah kartu di masing-masing *host* dan sebuah kabel jaringan yang sudah terpasang konektor RJ45. Akan menjadi masalah apabila ada lebih dari 2 (dua) *host* yang perlu terhubung. Rumusnya akan menjadi demikian :

- (a) Untuk mengetahui jumlah kebutuhan kartu jaringan yaitu menggunakan rumus :

$$JumlahKartuJaringan = n(n - 1)$$

dimana n adalah jumlah *host*

- (b) Sedangkan untuk mengetahui jumlah kabel jaringan yaitu menggunakan rumus :

$$JumlahKabelJaringan = \frac{n(n - 1)}{2}$$

dimana n adalah jumlah *host*

Ini akan lebih efektif dan efisien apabila digunakannya *switch* sebagai penghubung atau pengatur lalu lintas data antar *host*. Sebagai contoh bila ada 2 (dua) *host* yang akan dihubungkan dengan jaringan menggunakan *switch* ini, maka diperlukan 1 (satu) buah *switch*, 2 (dua) buah kabel jaringan (UTP), dan 2 (dua) buah kartu jaringan.

Bagaimana bila ada 3 (tiga) atau lebih *host* yang akan dihubungkan dalam sebuah jaringan yang sama, maka cukup menggunakan 1 (satu) buah *switch*, n buah kabel jaringan (UTP) dan n buah kartu jaringan sebanyak jumlah *host* yang akan dihubungkan. Tentunya model *switch* perlu disesuaikan jumlah *port* dengan jumlah *host* yang akan dihubungkan.

Switch ini selain menjadikan jaringan komputer lebih efektif dan efisien, juga mengatur lalu lintas data secara lebih baik (dibandingkan *hub*). Salah satu kelebihan yang dimiliki *switch* dibanding *hub* adalah data akan disampaikan tepat kepada alamat yang dituju, sehingga pertukaran data nyaris aman dalam jalur komunikasinya, sedangkan *hub* lebih rentan karena setiap data yang data yang terkirim dan dilewatkan melalui *hub*, maka alat ini akan

menyebarkan berita ke setiap *host* yang terhubung dengan *hub* yang kemudian diserahkan kepada kartu jaringan apakah data yang diterima sesuai dengan alamat yang dimiliki atau tidak. Dalam kondisi ini, keamanan data akan terlalu beresiko saat terjadi komunikasi.

Switch ini pun tidak perlu membeli karena sudah terpasang pada sistem yang lama sebagai penghubung antar 1 (satu) *host* dengan *host* yang lain.

7. Router

Alat ini dalam istilah lain juga disebut sebagai gerbang (*gateway*), yaitu sebuah jembatan yang menghubungkan 2 (dua) atau lebih jaringan yang berbeda alamat, cara kerja alat ini yaitu dengan memiliki peta *routing* yang diprogram, peta inilah yang dibaca oleh *router* kemana paket data yang diterima harus dikirimkan, apabila tujuan paket data hanya dikirimkan untuk *host* yang berada di jaringan lokal, maka *router* tidak akan mengirimkan paket datanya keluar, melainkan akan dikirim ke alamat yang berada di dalam jaringan LAN (*Local Area Network*), tetapi bila tujuan paket data berada di luar jaringan, maka paket tersebut akan dikirimkan ke jaringan lain dalam lingkupnya, atau diteruskan ke *router* lain yang menghubungkan jaringan di luar lingkupnya.

Tentunya cara kerja tersebut di atas dengan konfigurasi peta jaringan yang sebelumnya sudah diatur dalam *router*. Dan biasanya, pada setiap *router* terdapat *firewall* yang akan memberikan proteksi atau aturan-aturan komunikasi, dimana tidak semua data dapat diteruskan ke jaringan lain di luarnya, tetapi harus memenuhi persyaratan yang ditetapkan dalam tabel *firewall*.

Jaringan internal atau LAN pada bidang Pendataan dan Penetapan memiliki sebuah alamat jaringan lokal baik untuk jaringan nirkabel atau jaringan dengan kabel (UTP) di alamat 192.168.2.0/24, sedangkan untuk terhubung den-

gan internet digunakan sebuah *router* yang akan menjembatani atau menjadi gerbang antara jaringan internet dengan jaringan pada LAN. *Router* ini pula yang nantinya akan meneruskan data dan melakukan filter data baik yang masuk dari jaringan internet yang aksesnya atau *request*-nya akan diteruskan ke peladen aplikasi *web*.

Dengan kondisi demikian, maka biaya pengadaan untuk *router* sendiri tidak diperlukan, karena sudah terpasang dan sudah digunakan untuk kegiatan komunikasi data pada jaringan yang sudah ada sebelumnya.

8. *Modem*

Fungsi dari *modem* yaitu untuk menghubungkan 2 (dua) atau lebih media jaringan yang berbeda sehingga data dapat dikomunikasikan pada jaringan yang berbeda media tersebut. Sebagai contoh, karena bidang Pendataan dan Penetapan menggunakan akses serat-optik (*fiber optik*) dari jaringan penyedia layanan internet, dan jaringan internal menggunakan kabel UTP, maka dibutuhkan *modem* untuk menerjemahkan data yang melintas melalui 2 (dua) media jaringan tersebut.

Karena *modem* ini sudah digunakan pada sistem jaringan sebelumnya, maka tidak perlu dibeli.

Sehingga kebutuhan perangkat keras secara keseluruhan sebetulnya tidak perlu mengeluarkan biaya tambahan lain karena perangkat-perangkat tersebut sudah tersedia dan cukup untuk mengakomodir berjalannya sistem informasi yang nantinya akan dibangun.

Kemudian, untuk sarana dan prasarana jenis perangkat lunak yang dibutuhkan yaitu :

1. Perangkat Lunak Sistem Operasi

Perangkat lunak sistem operasi digunakan sebagai dasar dari seluruh sistem yang akan berjalan di atasnya. Pemilihan sistem operasi ini pun harus dapat memenuhi kriteria stabil dalam melakukan tugasnya sebagai peladen untuk rentang waktu 24 jam selama 7 hari berturut-turut.

Tentunya sistem operasi ini sudah terpasang baik pada peladen-peladen yang nantinya akan menjadi peladen sistem basis data dan peladen aplikasi *web*. Pada peladen sistem basis data terpasang sistem operasi Windows Server 2008 R2 dengan lisensi legal yang sah tersedia karena hasil pengadaan di tahun sebelumnya. Sedangkan pada peladen aplikasi *web* telah terpasang sistem operasi Ubuntu Server 14.04 yang tentu saja, untuk sistem operasi ini berlisensi gratis sehingga kebutuhan akan kedua peladen ini tidak perlu mengeluarkan biaya tambahan baik untuk lisensinya maupun pemasangannya.

2. Perangkat Lunak Sistem Basis Data

Perangkat lunak basis data digunakan sebagai sistem yang nantinya akan mengatur dan menyimpan data-data hasil perekaman yang dilakukan oleh petugas pelayanan dan petugas entri data, yang kemudian hasil perekaman ini dapat diolah untuk menyajikan berbagai macam laporan yang nantinya diproduksi dan dibutuhkan oleh aplikasi *web*, termasuk di antaranya adalah aplikasi yang akan dibangun ini.

Pemilihan perangkat lunak basis data ini pun dikarenakan pada saat pengelolaan pajak bumi dan bangunan sektor perdesaan dan perkotaan di kelola oleh Pemerintah Pusat, aplikasi di bangun dan datanya di simpan dalam sistem basis data Oracle, sehingga pada saat persiapan pengalihan pajak bumi dan bangunan sektor perdesaan dan perkotaan menjadi pajak daerah, maka dibeli juga lisensi sistem basis data Oracle agar migrasi data ke Pemerintah

Daerah Kabupaten Brebes berjalan dengan lebih mudah dan tanpa kendala. Oleh sebab itu sistem basis data yang digunakan untuk membangun aplikasi sistem informasi pajak bumi dan bangunan sektor perdesaan dan perkotaan ini pun menggunakan sistem yang sama dan telah terpasang, sehingga tidak perlu mengeluarkan biaya kembali untuk pengadaan perangkat lunak basis data ini.

3. Perangkat Lunak Peladen *Web*

Perangkat lunak peladen *web* digunakan untuk menjalankan atau penyediaan aplikasi *web* yang telah dibangun yaitu sistem informasi PBB, perangkat lunak ini berbeda dengan peladen *web* yang digunakan untuk aplikasi *web* berbasis bahasa pemrograman PHP. Perangkat lunak peladen *web* yang digunakan harus mendukung *servlet* atau *web container*.

Alasan kenapa dipilih peladen yang mendukung *servlet* atau *web container* karena memiliki beberapa kelebihan seperti berikut ini :

(a) Efisien dan baik dalam *performance*

Performance servlet dapat dikatakan efisien dan baik karena tidak ada proses pembuatan berulang untuk tiap *request* dari klien (*client*). Setiap *request* ditangani oleh proses *servlet container*. *Servlet* tidak dibuat dan dihancurkan secara berulang-ulang, melainkan tetap tersimpan pada memori untuk menangani *request* yang datang selanjutnya.

(b) *Powerful*

Servlet memiliki kemampuan yang lengkap antara lain mampu melakukan penanganan *request*, *session*, *cookie*, akses ke basis data dengan *Java Database Connection* (JDBC) dan *caching*, serta pustaka yang lengkap untuk pembuatan aplikasi *web*.

(c) Aman

Servlet memiliki fasilitas *security* atau keamanan yang baik dan merupakan bagian dari teknologi Java yang sudah dari asalnya didesain dengan *security* atau keamanan yang baik.

(d) Portabilitas

Teknologi Java *servlet* dapat dijalankan di berbagai *servlet container*, *application server*, maupun sistem operasi.

(e) Proses *development* yang lebih cepat

Dengan menggunakan *servlet*, kita dapat menggunakan pustaka Java yang lengkap dan menggunakan komponen yang sudah ada tanpa membangun dari awal kembali.

(f) Tangguh

Servlet merupakan teknologi Java yang memiliki penanganan *memory* yang baik dan *garbage collector* sehingga menjadikan aplikasi *web* yang tangguh dan stabil.

(g) Telah digunakan dan diakui di dunia

Servlet merupakan teknologi Java yang telah digunakan di berbagai belahan dunia. Dapat ditemukan berbagai komponen, solusi, dan dukungan yang ditawarkan baik secara gratis maupun komersial.

(h) Murah

Dikatakan murah karena JDK Java dapat diunduh secara gratis, begitupun dengan *servlet container*.

Pemilihan perangkat lunak peladen *web* yang mendukung *servlet* ini pun tidak perlu mengeluarkan biaya, cukup menggunakan Apache Tomcat yang memberikan lisensi gratis dalam penggunaannya, dengan konfigurasi yang

tepat, maka peladen *web* Apache Tomcat ini cukup untuk melayani permintaan aplikasi dari beberapa klien sekaligus. Sehingga untuk pengadaan perangkat lunak peladen *web* ini pun tidak memerlukan tambahan biaya.

4. Perangkat Lunak Desain Aplikasi

Perangkat lunak desain aplikasi digunakan untuk membuat desain pembangunan aplikasi, mulai dari desain struktur basis data, desain *Unified Modeling Language* (UML) yang digunakan untuk membuat spesifikasi standar untuk mendokumentasikan, menspesifikan, dan membangun sistem perangkat lunak, desain UML sendiri dapat menggambarkan atau memodelkan diagram struktur aplikasi dan diagram sifat atau cara kerja aplikasi.

Perangkat lunak atau desain aplikasi ini pun tidak perlu mengeluarkan biaya tambahan, ada perangkat lunak untuk desain aplikasi yang gratis dan disediakan oleh *repository* Ubuntu yang bernama Dia.

Aplikasi Dia ini memiliki fasilitas bukan hanya untuk membuat desain UML atau struktur basis data, melainkan dapat pula untuk membuat desain jaringan komputer, desain rangkaian elektronik, dan beberapa desain lain sebagaimana terdapat pada halaman *repository* Dia di alamat <http://dia-installer.de/shapes/index.html.en>.

5. Perangkat Lunak *Integrated Development Environment* (IDE)

Perangkat lunak IDE ini digunakan untuk membangun aplikasi *web*, baik dari sisi tampilan tatap muka pengguna (*user interface*) maupun dari sisi kode. Pengujian *unit testing* pun dapat dilakukan dengan menggunakan perangkat lunak ini.

Beberapa pilihan untuk perangkat lunak IDE ini pun beragam dan banyak yang menyediakan secara gratis. Sebagai contoh adalah Netbeans dan Eclipse. Untuk pengembangan aplikasi ini dipilih IntelliJ IDEA karena tersedianya

beragam *plugins* sebagai pendukung dalam membangun sebuah aplikasi.

Dengan kata lain, untuk pengadaan perangkat lunak IDE ini pun tidak perlu mengeluarkan biaya tambahan.

6. Perangkat Lunak Manajemen Basis Data

Perangkat lunak manajemen basis data digunakan untuk membangun struktur basis data dengan lebih mudah, biasanya disertai dengan cekatan (*wizard*) yang memandu pengguna untuk membentuk sebuah sistem basis data. Perangkat lunak ini pun dapat digunakan untuk menguji kebenaran kode *Structured Query Language* (SQL) yaitu kode yang digunakan untuk mengolah data yang berada pada sistem basis data.

Karena basis data yang digunakan adalah Oracle, maka perangkat lunak untuk manajemen basis data yang digunakan adalah SQL Developer yang merupakan aplikasi yang dapat di unduh secara gratis, sehingga tidak perlu lagi mengeluarkan biaya tambahan untuk pengadaannya.

7. *Framework* Spring

Framework Spring adalah perangkat yang dapat membantu untuk membangun aplikasi Java berskala besar dengan mudah. *Framework* Spring sendiri sebetulnya terbagi menjadi beberapa modul. Aplikasi yang akan dibangun dapat memilih modul mana saja yang akan digunakan. Modul inti dari *Framework* Spring ini adalah modul *core container*, yang termasuk di dalamnya adalah model untuk konfigurasi dan mekanisme *dependency injection*.

Dependency Injection adalah salah satu fasilitas yang memudahkan kita untuk tidak perlu membentuk tiap objek baru secara manual, melainkan cukup dideklarasikan *bean*, kemudian *framework* Spring yang akan melakukan pembentukan objeknya secara otomatis.

Selain itu *framework* Spring pun memberikan dukungan dasar terhadap beberapa arsitektur aplikasi seperti *messaging*, transaksi data dan *persistence*, serta *web*. Spring pun menyertakan *servlet-based* Spring MVC *web framework*.

Beberapa spesifikasi yang didukung oleh *Framework* Spring ini adalah seperti berikut ini :

- Servlet API (JSR 340)
- WebSocket API (JSR 356)
- Concurrency Utilities (JSR 236)
- JSON Binding API (JSR 367)
- Bean Validation (JSR 303)
- JPA (JSR 338)
- JMS (JSR 914)
- Common Annotations (JSR 250)

Beberapa waktu tata aturan pada pengembangan aplikasi Java EE telah mengalami perubahan. Pada awal hadirnya Java EE dan Spring, aplikasi *web* yang telah dibangun akan di *deploy* ke peladen aplikasi. Namun sekarang dengan hadirnya Spring Boot, aplikasi dibangun dalam lingkungan *devops* dan lebih ramah dengan teknologi *cloud* dengan adanya *servlet container* yang dipendam dalam paket *framework*. Bahkan pada *Framework* Spring 5, aplikasi WebFlux tidak perlu menggunakan API *servlet* secara langsung dan dapat menggunakan peladen *web* seperti Netty yang tidak memiliki *servlet containers*.

8. Spring Data JPA

Spring Data JPA sebetulnya adalah bagian dari *framework* Spring yang tujuan dibentuknya adalah untuk mengurangi jumlah kode *boilerplate* yang dibutuhkan untuk mengimplementasikan lapisan akses data ke berbagai sistem basis data.

Kelebihan dari penyederhanaan ini adalah mengurangi jumlah *artifact* yang harus dideklarasikan dan di *maintained*, konsistensi pada pola data akses, dan konsistensi pada konfigurasi. Satu-satunya yang perlu dideklarasikan pada Spring Data JPA adalah *interface* DAO.

Untuk dapat mengimplementasikan Spring Data JPA, *interface* DAO terlebih dahulu harus melakukan *extends* terhadap salah satu antar-muka *repository* yang disediakan oleh Spring Data JPA, nantinya Spring Data akan mencari antar-muka ini dalam pustakanya, kemudian membuatkan implementasinya secara otomatis.

9. Pustaka *Java Database Connection* (JDBC)

Pustaka *driver* JDBC diperlukan sebagai jembatan antara kode aplikasi yang dibangun dengan data yang berada dalam sistem basis data. *Driver* JDBC ini spesifik dan berbeda untuk tiap sistem basis data, karena sistem basis data yang digunakan adalah Oracle, maka *driver* yang diperlukan adalah *driver* untuk basis data Oracle yang dapat diunduh di laman <http://www.oracle.com/technetwork/apps-tech/jdbc-112010-090769.html>, karena kita menggunakan *Maven* sebagai *build-tools*, sehingga kita cukup mendeklarasikan skrip XML seperti berikut di berkas konfigurasi untuk Maven :

```
<dependency>
  <groupId>ojdbc</groupId>
  <artifactId>ojdbc</artifactId>
  <version>14</version>
```

</dependency>

Biasanya tiap-tiap sistem basis data akan menyediakan *driver* JDBC masing-masing sebagai salah satu bentuk dukungan bahwa sistem basis data tersebut mampu ikut terintegrasi bersama aplikasi yang dibangun dengan bahasa pemrograman Java.

Lisensi untuk pustaka *driver* JDBC ini pun gratis sehingga tidak perlu mengeluarkan biaya untuk pengadaannya.

10. Pustaka Lombok

Pustaka ini digunakan untuk mengurangi *boilerplate* saat kita membuat sebuah kelas entitas atau saat kita ingin membuat beberapa fungsi generik seperti *get* dan *set*. Pustaka ini pun menyediakan beberapa fungsi atau *method* dan konstruktor yang umum biasa dibentuk dan digantikan hanya dengan sebuah anotasi. Kondisi ini akan sangat meringkas jumlah kode yang harus di ketikkan.

11. Pustaka Jackson

Pustaka ini sangat terkenal dan efisien untuk mengubah objek Java ke dalam format JSON atau sebaliknya. Pustaka ini akan digunakan untuk layanan REST nantinya, ketika API akan mengembalikan data sebuah objek Java, maka Jackson akan mengubah secara otomatis objek tersebut menjadi JSON, begitu pula saat menerima data dari aplikasi klien yang berbentuk JSON, akan mengubahnya secara otomatis menjadi bentuk objek Java.

3 SUMBER DAYA MANUSIA YANG TERLIBAT DALAM SISTEM APLIKASI

Karena sistem aplikasi ini hanya melakukan akses informasi terhadap pencatatan pembayaran pajak bumi dan bangunan untuk sektor perdesaan dan perkotaan, maka sumber daya manusia yang terlibat dalam sistem aplikasi ini nantinya adalah masyarakat umum wajib pajak yang dapat mengoperasikan atau menggunakan *browser* untuk melakukan akses aplikasi.

Adapun untuk mendukung penyampaian data yang valid, maka dibutuhkan sumber daya manusia pada sisi internal untuk memastikan bahwa data yang telah tercatat pembayarannya pada BPD Jawa Tengah, dapat tercatat pula secara periodik pada SISMIOP, kondisi ini diperlukan karena mengingat bahwa untuk pengelolaan atau manajemen data ketetapan pajak bumi dan bangunan sektor perdesaan dan perkotaan berada pada peladen di BPPKAD Kabupaten Brebes, sedangkan basis data tempat melakukan pencatatan pembayaran berada di peladen BPD Jawa Tengah, data yang tercatat pembayarannya di hari ini, perlu dicatatkan dan diverifikasi pencatatannya pada peladen di BPPKAD Kabupaten Brebes pada H+1.

Dengan demikian sistem informasi pembayaran pajak bumi dan bangunan sektor perdesaan dan perkotaan akan menampilkan informasi yang valid kepada masyarakat wajib pajak.

4 ORGANISASI SISTEM APLIKASI

Organisasi sistem informasi pencatatan pembayaran sebetulnya cukup sederhana, karena hanya menampilkan informasi pencatatan pembayaran, namun untuk menghasilkan informasi pencatatan pembayaran yang valid, perlu di dukung dengan

data yang akurat, yang perlu ditingkatkan adalah menjaga tingkat akurat pencatatan pembayaran pada SISMIOP yang telah dihasilkan oleh peladen BPD Jawa Tengah.

Alur yang didapat dari awalnya adalah pada saat proses penetapan untuk pertama kali di tahun pajak berjalan, proses penetapan ini akan menghasilkan nilai awal ketetapan dari tiap objek pajak yang memenuhi kriteria, nilai hasil ketetapan ini akan diduplikasikan ke peladen BPD Jawa Tengah agar pihak BPD Jawa Tengah dapat melakukan verifikasi data apabila ada penyetoran pajak bumi dan bangunan perdesaan dan perkotaan yang akan masuk. Data awal ini akan dipastikan sama antara kondisi di peladen BPPKAD Kabupaten Brebes, dengan peladen milik BPD Jawa Tengah.

Seiring berjalannya waktu, kondisi pencatatan pembayaran pada peladen BPD Jawa Tengah akan berjalan, bahkan pada kondisi dimana hari sedang libur atau jam kantor kas masih dalam kondisi tutup, karena transaksi yang dilakukan dapat melalui mesin ATM. Kondisi untuk objek-objek yang telah tercatat terbayar tidak dapat lagi diubah pada peladen BPD Jawa Tengah, dan pada H+1 data dapat diunduh untuk kepentingan pencatatan pembayaran pada peladen di BPPKAD Kabupaten Brebes.

Pada kondisi yang bersamaan (di hari yang sama) dapat pula terjadi perubahan data pada peladen di BPPKAD Kabupaten Brebes akibat mutasi, penggabungan, pengurangan, perubahan data objek mengenai luasan, zona, atau hal yang lain. Idealnya, kondisi perubahan yang terjadi akibat pengajuan perubahan data pada BPPKAD Kabupaten Brebes, tidak diperbolehkan melakukan pembayaran pajak bumi dan bangunan perdesaan dan perkotaan di BPD Jawa Tengah, karena akan mengakibatkan ketidak-sesuaian data antara BPD Jawa Tengah dan peladen BPPKAD Kabupaten Brebes. Hal inilah yang sering terjadi dan perlu petugas khusus untuk melakukan penyesuaian data yang terjadi akibat perubahan-perubahan data

tersebut.

Kegiatan inilah yang harus dilakukan setiap saat agar hasil atau informasi yang disampaikan kepada masyarakat wajib pajak pada sistem informasi ini menjadi lebih valid, akurat dan dapat dipertanggungjawabkan.

Solusi lain adalah dengan melakukan penghentian penetapan ulang atas perubahan objek pajak, sehingga apabila ada pengajuan perubahan data pada tahun berjalan, data akan langsung diubah, namun tidak ditetapkan, sehingga tidak menimbulkan tagihan baru pada peladen BPPKAD Kabupaten Brebes yang mungkin akan tidak bersesuaian dengan data yang ada pada peladen pada BPD Jawa Tengah yang telah tercatat terbayar.

5 WAKTU DAN BIAYA YANG DIBUTUHKAN DALAM PEMBUATAN / PENGEMBANGAN SISTEM APLIKASI SECARA MENYELURUH

Dari sisi biaya, maka tidak diperlukan pengeluaran biaya untuk pembuatan / pengembangan ini, karena sebetulnya seluruh perangkat keras sudah terpasang dan berjalan normal, dan dari pengadaan perangkat lunaknya pun semuanya menggunakan versi gratis.

Sedangkan dari waktu yang dibutuhkan, akan menghabiskan waktu sekitar 9 minggu yang tentu saja ini waktu yang tidak terlalu lama untuk membangun sebuah aplikasi.

6 MANFAAT DAN DAMPAK DARI PENGEMBANGAN SISTEM APLIKASI

Beberapa manfaat dan dampak yang didapat dari pengolahan data nantinya adalah sebagai berikut :

1. Masyarakat wajib pajak dapat dengan mudah mengakses data objek-pajaknya, berapa besaran yang harus dibayarkan, dan bagaimana status pembayarannya, apakah sudah lunas atau belum, tanpa harus datang ke BPPKAD Kabupaten Brebes.
2. Masyarakat wajib pajak dapat melakukan kontrol terhadap petugas pemungut tingkat Desa/Kelurahan dalam melaksanakan kewajibannya meneruskan uang setoran pajak bumi dan bangunan perdesaan dan perkotaan ke BPD Jawa Tengah.
3. Bagi petugas pemungut tingkat Desa/Kelurahan pun dapat melakukan pemeriksaan ulang, objek-objek mana saja yang telah terbayarkan dan mana yang belum.