

## Sistem Informasi Pajak Bumi dan Bangunan Berbasis *Stand Alone*

Kristina Natonis<sup>1</sup>, Maria Adelvin Londa<sup>2</sup>, Melky Radja<sup>3</sup>

<sup>1,2,3</sup>Sistem Informasi, Teknologi Informasi, Universitas Flores, Ende, Indonesia

Email: <sup>1</sup>cristhynatonis@gmail.com, <sup>2</sup>adelvinmaria@gmail.com, <sup>3</sup>melkyradjastudy@gmail.com

### Abstrack

*The problem in this study is that the Land and Building Tax Service at the Potulando Village Office is still manual or conventional, so it requires a lot of place and time in storing data or processing data, besides that the reports produced are inaccurate. The purpose of this research is to design a Stand Alone -Based Land and Building Tax Information System. Sources of data in this study are primary data sources and secondary data sources. This type of research is qualitative while the data collection techniques in this study use three methods, namely, observation, interviews and library studies. The construction of a Stand Alone-Based Land and Building Tax Information System in this study uses Visual Basic.Net . The system development method used in this research is System Development. Life Cycle (SDLC), while The system testing technique in this study uses the Blackbox testing method. The system design in this study uses a use case diagrams, context diagrams, and data flow diagrams. With the existence of a Land and Building Tax Information System , it can improve the tax service process quickly and practically compared to the manual process , facilitate the process of recording taxpayer identity , able to obtain the desired taxpayer data quickly and easily. Facilitate the process of making tax reports , both monthly and annual reports .*

**Keywords:** Information System, Land and Building Tax, Vb .Net, MySQL, Blackbox Testing.

### Abstrak

Masalah dalam penelitian ini adalah Pelayanan PBB (Pajak Bumi dan Bangunan) pada Kantor Kelurahan Potulando masih manual atau konvensional, sehingga membutuhkan banyak tempat dan waktu dalam menyimpan data atau pengolahan data, selain itu laporan yang dihasilkan menjadi tidak akurat. Tujuan dalam penelitian ini adalah untuk merancang Sistem Informasi Pajak Bumi dan Bangunan Berbasis *Stand Alone*. Sumber data dalam penelitian ini adalah sumber data primer dan sumber data sekunder. Jenis penelitian ini adalah kualitatif sedangkan teknik pengumpulan data pada penelitian ini menggunakan tiga metode yaitu, observasi, wawancara dan studi Pustaka. Pembangunan Sstem Informasi Pajak Bumi dan Bangunan Berbasis Stand Alone pada penelitian ini menggunakan Visual Basic.Net. Metode pengembangan sistem yang digunakan dalam penelitian ini adalah System Development. Life Cycle (SDLC), sedangkan teknik pengujian sistem pada penelitian ini menggunakan metode Blackbox testing. Desain sistem pada penelitian ini menggunakan usecase diagram, context diagram, dan data flow diagram. Dengan adanya Sistem Informasi Pajak Bumi dan Bangunan, dapat meningkatkan proses pelayanan perpajakan dengan cepat dan praktis dibandingkan dengan proses manual, memudahkan proses pencatatan identitas wajib pajak, mampu memperoleh data wajib pajak yang diinginkan dengan cepat dan memudahkan proses pembuatan laporan pajak, baik laporan bulanan dan tahunan.

**Kata Kunci:** Sistem Informasi, Pajak Bumi dan Bangunan, Vb .Net, MySQL, Blackbox Testing.

### 1. Pendahuluan

#### 1.1 Latar Belakang Masalah

Seperti yang kita ketahui bahwa pajak memiliki peranann penting dalam sebuah negara, dimana pajak merupakan sumber pendapatan dalam pembangunan suatu negara, sehingga sangat dibutuhkanya sebuah

sistem informasi yang memiliki peranan penting dalam pengolahan dan pembuatan laporan pemungutan pajak.

Namun hingga saat ini masih banyak instansi-instansi pemerintahan yang masih menggunakan cara lama atau manual dalam mengolah data pajak. Salah satunya adalah Kantor Kelurahan Potulando, dimana proses pelayanan perpajakan di instansi tersebut membutuhkan waktu yang cukup banyak dan

menguras tenaga, dapat dilihat dari proses pencarian data wajib pajak. Pada kelurahan tersebut terdapat 16 RT dan 6 RW dengan jumlah jiwa yang terdaftar wajib pajak sebanyak 985 KK (Kepala Keluarga).

Banyaknya warga Kelurahan Potulando yang terdaftar wajib pajak membuat petugas pajak mengalami kewalahan dalam mencari data wajib pajak yang diinginkan. Begitu juga dengan proses penginputan data dan pembuatan laporan keuangan yang dilakukan secara berulang dan membutuhkan banyak tempat penyimpanan. Misalnya seseorang yang terdaftar wajib pajak telah melakukan pembayaran pada tahun 2021 kembali melakukan pembayaran pada tahun 2022, maka petugas yang bersangkutan akan membuat file baru untuk menginput data wajib pajak, karena data wajib pajak yang diperoleh sebelumnya telah diarsipkan. Selain itu untuk pertanggungjawaban dalam dunia kerja dibutuhkan laporan, baik laporan harian, mingguan, bulanan, hingga tahunan, namun dalam membuat laporan keuangan terkait pajak bumi dan bangunan pada instansi tersebut juga memerlukan waktu yang cukup banyak dan rumit, dimana petugas yang bersangkutan harus membuat laporan dengan membuat file baru dan berpatokan pada data wajib pajak dari tahun, bulan, minggu dan hari sebelumnya sebagai pedoman. Tentu hal ini dapat menyebabkan laporan keuangan menjadi tidak akurat jika orang yang bersangkutan lelah atau kehilangan konsentrasi.

Oleh karena itu penulis membuat sebuah sistem yang berjudul Perancangan Sistem Informasi Pajak Bumi dan Bangunan Berbasis *Stand Alone* di Kantor Kelurahan Potulando, dimana sistem tersebut dapat bekerja sesuai perintah admin dan user agar dapat dengan mudah mengolah data pajak, salah satunya melakukan pencatatan mulai dari input data wajib pajak hingga menghasilkan laporan harian, mingguan, bulanan dan tahunan, sesuai prosedur atau cara kerja yang berlaku dalam instansi tersebut. Hal ini dikarenakan pajak mempunyai peranan penting dalam kehidupan bernegara.

## 1.2 Identifikasi Masalah

Berdasarkan latar belakang diatas, maka identifikasi masalah dalam penelitian ini adalah :

1. Pelayanan PBB (Pajak Bumi dan Bangunan) pada Kantor Kelurahan Potulando masih manual atau konvensional.
2. Membutuhkan banyak tempat penyimpanan dalam mengolah data wajib pajak.
3. Membutuhkan banyak waktu dalam pelayanan wajib pajak dan pembuatan laporan.
4. Memperoleh data yang tidak akurat.

## 1.3 Batasan Masalah

Agar penyusunan skripsi ini tetap berada pada tujuan yang telah di rumuskan, maka pembahasan di batasi pada proses pembuatan Sistem Informasi Pajak Bumi dan Bangunan Berbasis *Stand Alone*.

## 1.4 Rumusan Masalah

Berdasarkan Batasan masalah diatas, maka rumusan masalah dalam penelitian ini adalah bagaimanakah merancang Sistem Informasi Pajak Bumi dan Bangunan Berbasis *Stand Alone* ?.

## 1.5 Tujuan Penelitian

Dari rumusan masalah diatas maka tujuan dalam penelitian ini adalah untuk merancang Sistem Informasi Pajak Bumi dan Bangunan Berbasis *Stand Alone*.

## 1.6 Manfaat Penelitian

Adapun manfaat yang diharapkan dari penelitian ini adalah 1. Sebagai sarana pengaplikasian kemampuan dan pengetahuan yang di peroleh selama mengikuti perkuliahan, 2. Sebagai sarana menggali informasi dan mempersiapkan diri untuk terjun dalam dunia kerja, 3. Melatih tanggung jawab dan disiplin dalam hal pengolahan sistem informasi pembayaran pajak dan manajemen waktu dalam menjalankan tugas yang diberikan, 4. Sebagai sarana pembinaan hubungan baik terhadap instansi pemerintahan dengan calon tenaga kerja dari lulusan-lulusan mahasiswa nantinya, 5. Meningkatkan dan menciptakan mahasiswa dengan kemampuan berkualitas, 6. Mengukur kemampuan tenaga pendidik dalam memberi materi perkuliahan kepada mahasiswa dengan perkembangan dunia kerja, 7. Meringankan beban kerja instansi dimana mahasiswa di tempatkan, 8. Instansi dapat melakukan tanggung jawab sosialnya karena telah memberikan kesempatan mahasiswa untuk melakukan kegiatan penelitian, 9. Dapat menjalin hubungan baik, harmonis, dan saling menguntungkan dengan pihak universitas, 10. Menjalinkan kontribusi terhadap dunia pendidikan.

## 2. Landasan Teori

### 2.1 Teori Penelitian

Ada beberapa teori yang akan penulis paparkan dalam bab ini dimana teori-teori ini menjadi penunjang dalam pembuatan Sistem Informasi Pajak Bumi dan Bangunan Berbasis *Stand Alone* di Kantor Kelurahan Potulando.

#### 2.1.1 Sistem Informasi

Sistem Informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan. Sistem informasi adalah cara-cara yang diorganisasi untuk mengumpulkan, memasukkan dan mengolah serta menyimpan data, dan cara-cara yang diorganisasi untuk menyimpan, mengelola, mengendalikan, dan melaporkan informasi sedemikian rupa sehingga sebuah organisasi dapat mencapai tujuan yang telah

ditetapkan. Berdasarkan uraian di atas, penulis menyimpulkan bahwa sistem informasi adalah suatu sistem yang digunakan untuk mengolah data dalam suatu organisasi yang bersifat manajerial.

#### 2.1.2 Pajak Bumi dan Bangunan

Pajak Bumi dan Bangunan (PBB) adalah Pajak Negara yang dikenakan terhadap bumi dan atau bangunan berdasarkan Undang-undang nomor 12 Tahun 1985 tentang Pajak Bumi dan Bangunan sebagaimana telah diubah dengan Undang - Undang nomor 12 Tahun 1994. PBB adalah pajak yang bersifat kebendaan dalam arti besarnya pajak terutang ditentukan oleh keadaan objek yaitu bumi/tanah dan atau bangunan. Keadaan subjek (siapa yang membayar) tidak ikut menentukan besarnya pajak.

#### 2.1.3 Stand Alone (Desktop)

Aplikasi *desktop* adalah aplikasi yang dapat berjalan secara sendiri atau independen dalam sistem *desktop* komputer atau laptop dan dapat menjalankan serangkaian aktivitas dengan diatur oleh pengguna. Pada penelitian ini, yang dimaksud dengan aplikasi berbasis *stand alone* (berdiri sendiri) atau yang lebih dikenal dengan sebutan *desktop* merupakan sistem yang dirancang hanya berfokus pada Sistem Informasi Pajak Bumi dan Bangunan dan hanya dapat digunakan untuk kebutuhan pelayanan Pajak Bumi dan Bangunan pada Kantor Kelurahan Potulando.

#### 2.1.4 Visual Basic.Net

*Visual Basic.Net* adalah sebuah alat untuk mengembangkan dan membangun aplikasi yang bergerak diatas sistem *.NET Framework*, dengan menggunakan Bahasa *BASIC*. *Visual Basic .Net* adalah generasi penerus *visual basic 6* dari microsoft. *Visual Basic* merupakan bahasa yang mudah dipelajari dan digunakan sehingga *visual basic* telah menjadi bahasa pemrograman pilihan oleh banyak sekali developer di dunia selama dekade terakhir. *Visual Basic .Net* memiliki perbedaan dengan versi sebelumnya sehingga memerlukan usaha dan waktu bagi programmer *visual basic* untuk berpindah. Namun karena sebagian besar konsep dan sintaks masih sama, tentunya waktu dan usaha yang diperlukan tidaklah menyulitkan.


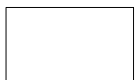

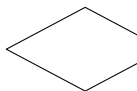
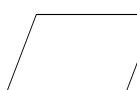
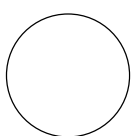
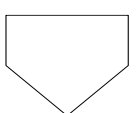

#### 2.1.5 MySQL

*MySQL* adalah RDBMS yang cepat dan mudah digunakan, serta sudah banyak digunakan untuk berbagai kebutuhan. *MySQL* dikembangkan oleh *MySQL AB Swedia*. *MySQL* adalah database yang paling populer diantara database-database yang lain. *MySQL* pertama kali dirintis oleh seorang programmer database bernama Michael Widenius. *MySQL* adalah program database yang mampu mengirim dan menerima data dengan sangat cepat dan multi user.

#### 2.16 Flowchart

*Flowchart* adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. *Flowchart* menolong analis dalam untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian.

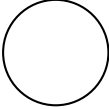

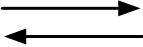

Tabel 1. Simbol *Flowchart*

Simbol	Deskripsi
	<i>Terminator</i> , Awal atau akhir konsep prosedur
	<i>Process</i> , Proses operasional
	<i>Document</i> , Dokumen atau laporan berupa <i>print out</i>
	<i>Decision</i> , Keputusan atau sub-point, garis yang terhubung dengan bentuk <i>decision</i> merujuk pada situasi-situasi yang berbeda sesuai dengan keputusan yang digambarkan
	Data, <i>Input</i> atau <i>Output</i> (contohnya, <i>Input</i> : <i>feedback</i> dari pelanggan, <i>Output</i> : desain produk baru)
	<i>On-Page Reference/Connector</i> , Penghubung alur dalam halaman yang sama
	<i>Off-Page Reference/Off-Page Connector</i> , Penghubung alur dalam halaman yang berbeda
	<i>Flow</i> , Arah atau alur dalam konsep ( <i>Prosedur</i> )

#### 2.1.7 Data Flow Diagram (DFD)

*Data Flow Diagram (DFD)* adalah representasi grafik dari sebuah system. DFD menggambarkan komponen-komponen sebuah system, aliran-aliran data diantara komponen-komponen tersebut, asal, tujuan dan penyimpanan dari data tersebut. Berikut simbol-simbol dari *Data Flow Diagram (DFD)*.



Tabel 2. Simbol *Data Flow Diagram*

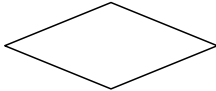

No	Simbol	Keterangan
		<i>Process</i> , simbol ini digunakan untuk proses pengolahan atau transformasi data.
		<i>External entity</i> , simbol ini digunakan untuk menggambarkan asal atau tujuan data.
		<i>Data flow</i> , simbol ini digunakan untuk menggambarkan aliran data yang berjalan
		<i>Data store</i> , simbol ini digunakan untuk menggambarkan data <i>flow</i> yang sudah disimpan atau diarsipkan

### 2.1.8 Entity Relationship Diagram ( ERD)

ERD (*Entity Relationship Diagram*) adalah model teknik pendekatan yang menyatakan atau menggambarkan hubungan suatu model. Di Dalam hubungan ini tersebut dinyatakan yang utama dari ERD adalah menunjukan objek data (*Entity*) dan hubungan (*Relationship*), yang ada pada *Entity* berikutnya.

Tabel 1. Simbol *Entity Relation Diagram*

Simbol	Deskripsi
Entitas/ <i>Entity</i> 	Entitas merupakan data inti yang akan disimpan, bakal tabel pada basis data, benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer, penamaan entitas biasanya lebih kekata benda dan belum merupakan nama tabel.
<i>Atribut Primary Key</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan, biasanya berupa <i>id</i> , kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom

	tersebut dapat bersifat unik (berbeda tanpa ada yang sama).
Relasi 	Relasi yang menghubungkan antar entitas, biasa di awal dengan kata kerja.
Asosiasi 	Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B.

### 2.2 Kajian Penelitian Relevan

Kajian penelitian relevan dapat di jelaskan sebagai berikut:

Tabel 2. Kajian Penelitian Relevan

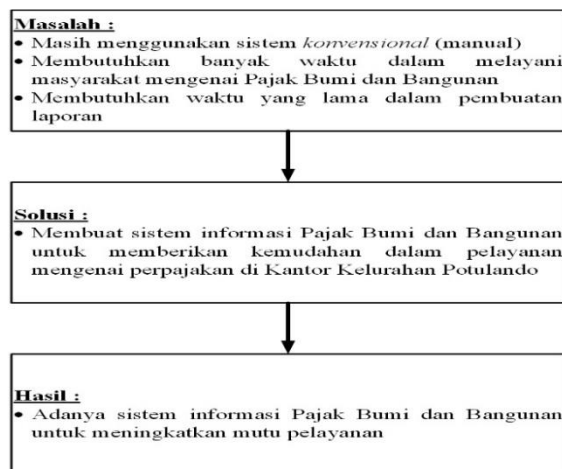
No	Nama Peneliti (Tahun Penelitian)	Judul Penelitian	Kesimpulan
1	Trinomika Takke Palabuan (2017)[12]	Pembuatan Sistem Informasi Untuk Pajak Bumi dan Bangunan Berbasis WEB dengan Menggunakan Bahasa Pemrograman PHP dan <i>Postgresql</i>	Dalam membuat sistem informasi pada penelitian ini menggunakan metode <i>SIG</i> berbasis web
2.	penelitian Yani Parti Astuti, Egia Rosi Subhiyako(2017)[13]	Pengembangan Sistem Informasi Dengan Metode Waterfall Untuk Pengarsipan Data Wajib Pajak	Analisis dan perancangan pada penelitian ini menggunakan metode berorientasi objek. Metode berorientasi objek mempunyai Bahasa pemodelan

			standar yakni UML ( <i>Unified Modeling Language</i> )
3.	Fauziyah, Dominique Navia Jasmine, Alexius Ulan Bani(2021)[14]	Rancang Bangun Sistem Pelayanan Pengurusan Pajak Pada kantor Konsultan Pajak R. Nursalim & Rekan Berbasis	Metode yang dipakai dalam merancang sistem informasi ini adalah metode <i>System Development Life Cycle (SDLC)</i>

Dilihat dari tabel diatas, penulis menyimpulkan bahwa yang membedakan antara penelitian penulis dengan ketiga penelitian relevan diatas adalah dalam membuat Sistem Informasi Pajak Bumi dan Bangunan *Berbasis Stand Alone*, penulis menggunakan bahasa pemrograman *Microsoft Visual Studio*, dan metode pengujian yang digunakan dalam penelitian ini adalah metode *black box testing*.

### 2.3 Kerangka Pikir

Kerangka pikir dalam penelitian ini dijelaskan secara garis besar alur logika dalam suatu diagram sebagai berikut :



Gambar 1. Kerangka Pikir Penelitian

### 2.4 Pertanyaan Penelitian

Pertanyaan yang penulis gunakan dalam penelitian ini adalah, 1. Kendala apa saja yang dihadapi saat belum menggunakan Sistem Informasi Pajak Bumi dan Bangunan?, 2. Sistem seperti apa yang di inginkan petugas atau staff yang bersangkutan dalam menyelesaikan persoalan yang ada?.

## 3. Metode Penelitian

### 3.1 Jenis Penelitian

Jenis penelitian ini adalah kualitatif dimana dalam mendukung proses pelaksanaan penelitian, penulis melakukan pengamatan yang mendalam tentang ucapan, tulisan maupun perilaku yang dapat diamati dalam suatu keadaan yang dikaji dari sudut pandang yang menyeluruh.

### 3.2 Lokasi dan Waktu Penelitian

Lokasi dalam penelitian di Kantor Kelurahan Potulando, jalan Patimura, Kecamatan Ende Tengah, Kabupaten Ende, sedangkan waktu dalam penelitian ini dilaksanakan selama 3 bulan mulai dari tanggal 1 mei 2022 sampai tanggal 1 Juli 2022.

### 3.3 Profil Kelurahan Potulando

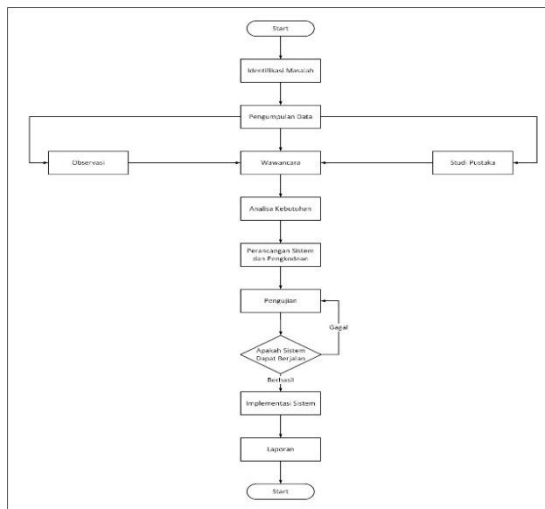
Kelurahan Potulando dibentuk berawal dari Pembentukan sebuah desa pada tahun 1965 yang pada saat itu diberi nama Desa Potulando berada dalam naungan wilayah Kecamatan Ende bersama dengan 7 desa lainnya yang saat itu disebut dengan desa gaya baru yaitu Desa Paupanda, Desa Rukun Lima, Desa Mbongawani, Desa Pancasila, Desa Onekore, Desa Ekasila, dan Desa Mautapaga. Namun Seiring dengan tuntutan kebutuhan dan perkembangan sistem pemerintahan secara nasional, maka pada Tahun 1981 status Desa Potulando dan 7 desa lainnya berubah status menjadi Kelurahan.

### 3.4 Subyek/Populasi dan Sampel Penelitian

Populasi dalam penelitian ini adalah pegawai atau staff yang bertugas dalam pelayanan perpajakan pada kantor Kelurahan Potulando yang berjumlah 15 orang, sedangkan sampel dalam penelitian ini adalah Seluruh staff atau pegawai Kantor Kelurahan Potulando dengan Jumlah sampel yang dapat dijadikan subjek penelitian berjumlah 15 orang. Jenis sampel yang digunakan dalam penelitian ini adalah sampling jenuh.

### 3.5 Tahap-tahap Penelitian

Tahapan penelitian merupakan gambaran jalannya penelitian sebagai pendukung kelancaran proses penelitian.



Gambar 2. Tahap-Tahap Penelitian

1. Identifikasi masalah, tahapan ini merupakan tahapan awal dimana, penulis melakukan studi lapangan agar mendapatkan informasi yang jelas tentang masalah-masalah yang dialami pada objek penelitian. Studi kasusnya di Kantor Kelurahan Potulando.
2. Tahap pengumpulan data, tahap pengumpulan data dalam penelitian ini dilakukan dengan tiga metode yaitu, observasi, wawancara dan studi Pustaka.
3. Tahapan analisis kebutuhan, tahapan analisis kebutuhan merupakan tahapan persiapan alat yang digunakan dalam perancangan sebuah sistem, baik perangkat keras seperti laptop dan beberapa perangkat lunak penunjang perancangan system.
4. Perancangan sistem dan pengkodean, tahapan ini merupakan tahapan perancangan sistem dan pengkodean dimana dilakukan menggunakan bahasa pemrograman *Microsoft Visual Basic.Net*.
5. Pengujian, pada tahapan ini pengujian dilakukan menggunakan metode *blackbox testing*, yang merupakan salah satu metode pengujian perangkat lunak khususnya pada *input* dan *output* aplikasi (apakah sudah sesuai dengan apa yang diharapkan atau belum).
6. Implementasi system, tahapan ini merupakan tahap penerapan aplikasi yang dilakukan saat sistem yang dirancang sudah sempurna.
7. Laporan tahapan ini merupakan tahapan akhir, dimana peneliti melakukan rekam layar saat demo aplikasi, sebagai bukti bahwa peneliti telah selesai membuat sebuah System Informasi Pajak Bumi dan Bangunan berbasis *Stand Alone* dengan sempurna.

### 3.6 Jenis dan Sumber Data

Sumber data yang sering digunakan dalam sebuah penelitian adalah sumber data primer dan sumber data sekunder yang juga akan penulis gunakan dalam proses pembuatan Sistem Informasi Pajak Bumi dan Bangunan Berbasis *Stand Alone*. Yang dimaksud sumber data primer dalam penelitian ini adalah penulis melakukan observasi pada kantor Kelurahan Potulando sebagai tahap awal pembuatan Sistem informasi Pajak Bumi dan Bangunan Berbasis *Stand Alone*. Sedangkan yang

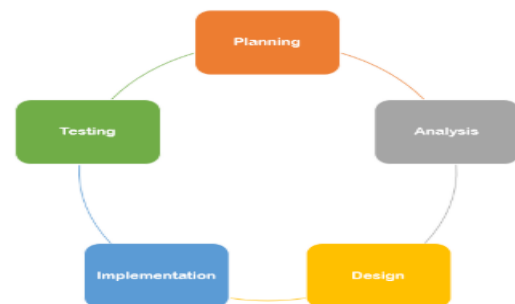
dimaksud sumber data sekunder dalam penelitian ini adalah penulis mencari referensi dari jurnal-jurnal yang mirip atau sama sebagai landasan teori.

### 3.7 Prosedur Pengumpulan Data

Tahap pengumpulan data dalam penelitian ini dilakukan dengan tiga metode yaitu, 1. Observasi, dalam mendukung proses pelaksanaan penelitian, penulis terjun langsung ke Kantor Kelurahan Potulando untuk memperoleh informasi yang jelas mengenai sistem pembayaran pajak bumi dan bangunan pada Kantor Kelurahan Potulando. 2. Wawancara, penulis melakukan wawancara langsung dengan salah satu staff yang bertugas melayani warga yang terdaftar wajib pajak. 3. Studi Pustaka, pada tahapan ini dilakukan pengumpulan data dan informasi melalui refrensi berbentuk buku-buku, makalah, serta *browsing* internet yang dapat dijadikan landasan teori.

### 3.8 Penerapan Metode *System Development Life Cycle* (SDLC)

Metode *System Development Life Cycle* (SDLC) merupakan metode pengembangan *system* yang penulis gunakan dalam penelitian ini. *System Development Life Cycle* (SDLC) adalah salah satu bentuk pengembangan sistem yang memiliki fungsi untuk menggambarkan tingkatan utama dan proses-proses yang terjadi dalam tahapan untuk pengembangan.



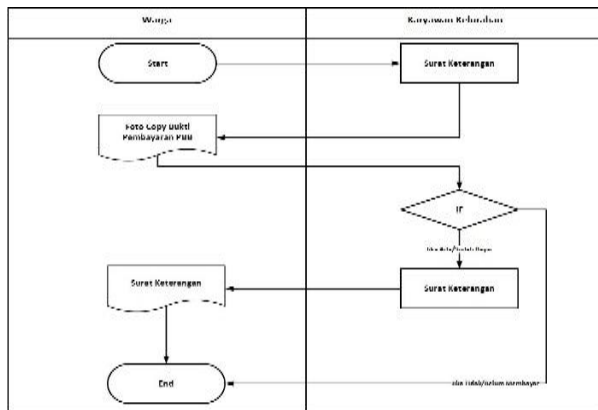
Gambar 3. *System Development Life Cycle*

Tahapan *System Development Life Cycle* (SDLC) pada sistem usulan pada tiap tahapannya adalah 1. Tahapan Perencanaan (*Planning*), tahapan perancangan merupakan tahapan awal dimana penulis mulai melakukan observasi pada Kelurahan potulando, lalu mengidentifikasi dan merumuskan masalah terkait dengan proses pelayanan pajak. 2. Tahapan Analisis (*Analysis*), tahapan analisis merupakan tahapan kedua dimana setelah mengidentifikasi dan merumuskan masalah terkait pelayanan pajak, penulis menganalisa proses pelayanan dan konsultasi pajak pada Kantor Kelurahan Potulando, dengan melakukan pengamatan dan melakukan wawancara terhadap pihak terkait. 3. Tahapan Desain (*Design*), tahapan desain merupakan tahapan dimana penulis mulai merancang system dengan prosedur pada sistem usulan, dengan menggunakan perancangan *Usecase Diagram*, *Data Flow Diagram*, *Entity Relationship Diagram*. 4.



Implementasi (*Implementation*), tahapan implementasi merupakan tahapan dimana penulis melakukan pengkodean perangkat lunak Sistem Informasi Pajak Bumi dan Bangunan Berbasis *Stand Alone*. 5. Pengujian (*Testing*), tahapan pengujian merupakan tahapan dimana penulis mulai melakukan demo program untuk mengetahui apakah sistem yang dibangun sudah sesuai yang diharapkan atau belum menggunakan metode *blackbox testing*.

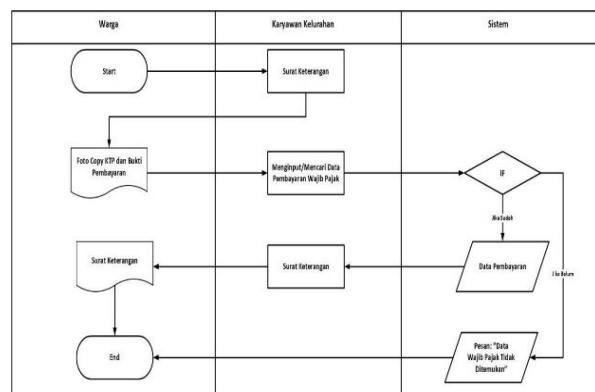
### 3.9 Analisis Sistem yang Berjalan



Gambar 4. Flowchart Analisis Sistem Yang Berjalan

Dilihat dari gambar diatas dapat dijelaskan bahwa untuk dapat mengambil surat keterangan pada karyawan kelurahan warga perlu membawa *fotocopy* KTP dan bukti pembayaran pajak bumi dan bangunan. Jika ada atau sudah dibayar maka karyawan kelurahan dapat memberikan surat keterangan pada warga. Jika tidak atau belum dibayar maka karyawan kelurahan tidak dapat memberikan surat keterangan pada warga.

### 3.10 Analisis Sistem yang Diusulkan



Gambar 5. Flowchart Sistem yang diusulkan

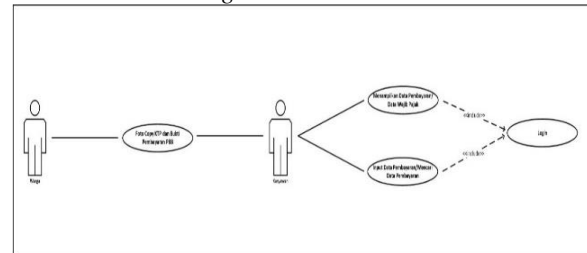
Dilihat dari gambar diatas dapat dijelaskan bahwa untuk dapat mengambil surat keterangan pada karyawan kelurahan warga perlu membawa *fotocopy* KTP dan bukti pembayaran pajak bumi dan bangunan. Kemudian karyawan kelurahan menginput atau mencari data pembayaran wajib pajak pada sistem, jika sudah maka sistem akan menampilkan data pembayaran dan karyawan kelurahan dapat memberikan surat keterangan pada warga. Jika belum maka sistem akan menampilkan

pesan “Data Wajib Pajak Tidak Dapat Ditemukan” maka karyawan kelurahan tidak dapat memberikan surat keterangan pada warga.

### 3.11 Desain Sistem

Desain sistem pada penelitian ini menggunakan *usecase diagram*, *context diagram*, dan *data flow diagram*.

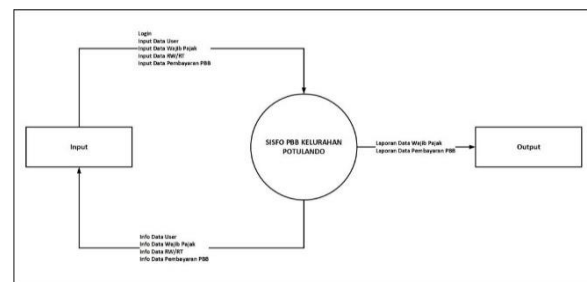
#### 3.11.1 Usecase Diagram



Gambar 6. Usecase Diagram

Dilihat dari gambar diatas dapat dijelaskan bahwa warga memberikan *fotocopy* KTP dan bukti pembayaran pajak bumi dan bangunan pada karyawan. Untuk dapat menampilkan data pembayaran atau data wajib pajak dan *input* data pembayaran atau mencari data pembayaran, karyawan perlu melakukan login untuk masuk kehalaman utama sistem.

#### 3.11.2 Context Diagram

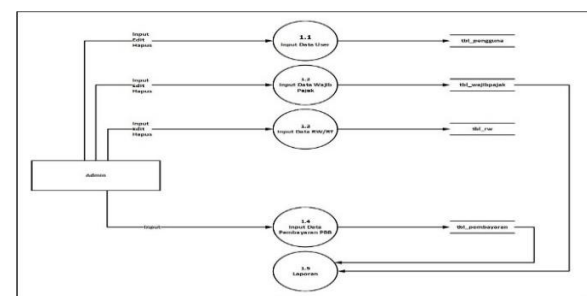


Gambar 7. Context Diagram

Dilihat dari gambar diatas dapat dijelaskan bahwa context diagram meliputi *Input*, Proses dan *Output*. Admin melakukan *input* data User, data wajib pajak, data RT/RW dan data Pembayaran Pajak Bumi dan Bangunan dan aplikasi melakukan proses simpan data ke *database*, sehingga aplikasi menghasilkan informasi dan laporan.

#### 3.11.3 Data Flow Diagram

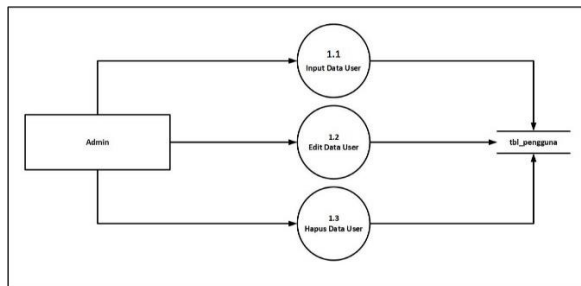
##### Data Flow Diagram Level 1



Gambar 8. Data Flow Diagram Level 1

Dilihat dari gambar diatas dapat dijelaskan sebagai berikut : 1. Admin melakukan proses *input*, edit dan hapus data *user* lalu data yang telah diinput tersimpan pada tabel pengguna. 2. Admin melakukan proses *input*, edit dan hapus data wajib pajak. Lalu data yang telah diinput tersimpan pada tabel wajib pajak dan menghasilkan laporan sebagai *output*. 3. Admin melakukan proses *input*, edit dan hapus data RW/RT. Lalu data yang telah diinput tersimpan pada tabel tabel RW/RT. 4. Admin melakukan proses *input* data pembayaran, Lalu data yang telah diinput tersimpan pada tabel tabel pembayaran dan menghasilkan laporan sebagai *output*.

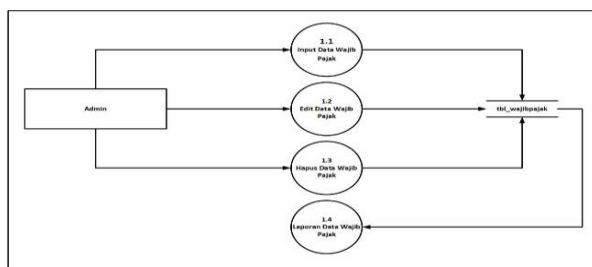
Data Flow Diagram Level 1 proses 1



Gambar 9. Data Flow Diagram Level 1 Proses 1

Dilihat dari gambar diatas dapat dijelaskan sebagai berikut : 1. Admin melakukan proses *input* data *user*, data yang telah diinput tersimpan pada tabel pengguna. 2. Admin melakukan proses edit data *user*, data yang telah diedit tersimpan pada tabel pengguna. 3. Admin melakukan proses hapus data *user*, data yang telah dihapus pada tabel pengguna akan terhapus juga dari *database*

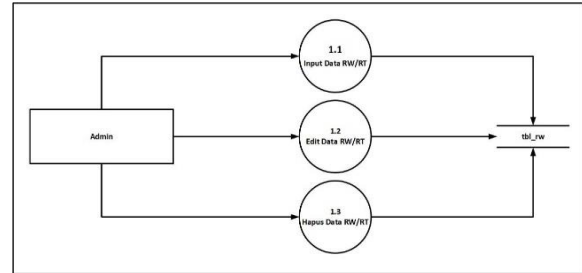
Data Flow Diagram Level 1 proses 2



Gambar 10. Data Flow Diagram Level 1 Proses 2

Dilihat dari gambar diatas dapat dijelaskan sebagai berikut : 1. Admin melakukan proses *input* data wajib pajak, data yang telah diinput tersimpan pada tabel wajib pajak dan menghasilkan laporan sebagai *otput*. 2. Admin melakukan proses edit data data wajib pajak, data yang telah diedit tersimpan kembali pada tabel wajib pajak dan menghasilkan laporan sebagai *output*. 3. Admin melakukan proses hapus data wajib pajak, data yang telah diinput tersimpan pada tabel wajib pajak dan menghasilkan laporan sebagai *output*.

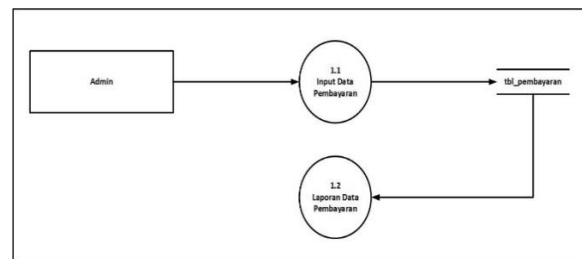
Data Flow Diagram Level 1 proses 3



Gambar 11. Data Flow Diagram Level 1 Proses 3

Dilihat dari gambar diatas dapat dijelaskan sebagai berikut : 1. Admin melakukan proses *input* data RT/RW, data yang telah diinput tersimpan pada tabel RT/RW. 2. Admin melakukan proses edit data RT/RW, data yang telah diedit tersimpan pada tabel RT/RW. 3. Admin melakukan proses hapus data RT/RW, data yang telah diinput atau diedit terhapus dari tabel RT/RW.

Data Flow Diagram Level 1 proses 4



Gambar 12. Data Flow Diagram Level 1 Proses 4

Dilihat dari gambar diatas dapat dijelaskan sebagai berikut: Admin melakukan proses *input* data pembayaran, data yang telah diinput tersimpan pada tabel pembayaran dan menghasilkan *output* berupa laporan pembayaran bulanan dan tahunan.

### 3.12 Desain Database

Desain *database* pada penelitian ini terdiri dari beberapa tabel yang mempunyai beberapa struktur dan fungsi yang berbeda.

#### 3.12.1 Normalisasi

##### Tabel Pengguna

Nama Tabe : tbl\_pengguna  
Primary Key : nip  
Fungsi : Berfungsi untuk menyimpan data pengguna

Tabel 5. Tabel Pengguna

	NAMA FIELD	TIPE DATA	KETERANGAN
1	Nip	Varchar (16)	Primary Key
2	Nama_pengguna	Varchar (30)	
3	Username	Varchar (15)	
4	Password	Varchar (15)	



5	Level	Varchar (6)
---	-------	-------------

#### Tabel Wajib Pajak

Nama Tabel : tbl\_wajibpajak

Primary Key : nop

Fungsi : Berfungsi untuk menyimpan data wajib pajak.

Tabel 6. Tabel wajib Pajak

NO	NAMA FIELD	TIPE DATA	KETERANGAN
1	Nop	Varchar (25)	Primary Key
2	Nm_wajibpajak	Varchar (30)	
3	Almt_objekpajak	Varchar (65)	
4	Rw_rt	Varchar (7)	
5	Kd_rw_rt	Varchar (3)	Foreign Key
6	Almt_wajibpajak	Varchar (65)	
7	L_bumi	Integer	
8	Luas_bangunan	Integer	
9	Pajak_terutang	Integer	

#### Tabel RT dan Rw

Nama Tabel : tbl\_rw\_rt

Primary Key : kd\_rw\_rt

Fungsi : Berfungsi untuk menyimpan data RW dan RT

Tabel 7. Tabel RW/RT

NO	NAMA FIELD	TIPE DATA	KETERANGAN
1	Kd_rw_rt	Varchar (7)	Primary Key
2	Rw_rt	Varchar (15)	

#### Tabel Pembayaran

Nama Tabel : tbl\_pembayaran

Primary Key : no\_kuitansi

Fungsi : Berfungsi untuk menyimpan data wajib pajak yang sudah melakukan pembayaran di kantor Bapenda

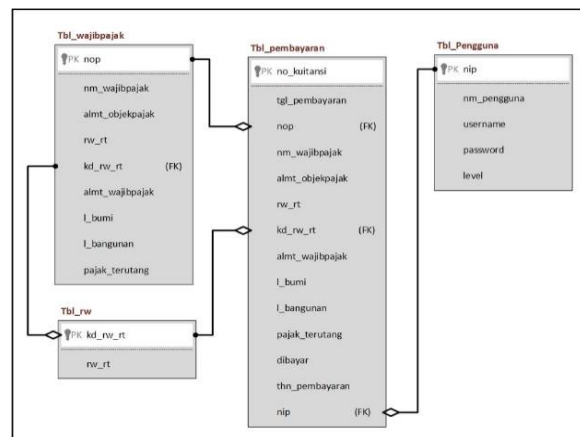
Tabel 8. Pembayaran

NO	NAMA FIELD	TIPE DATA	KETERANGAN
1	No_kuitansi	Varchar (7)	Primary Key
2	Tgl_pembayaran	Date	
3	Nop	Varchar (25)	Foreign Key

4	Nm_wajibpajak	Varchar (30)	
5	Almt_objekpajak	Varchar (30)	
6	Rw_rt	Varchar (30)	
7	Kd_rw_rt	Varchar (3)	Foreign Key
8	Almt_wajibpajak	Varchar (30)	
9	L_bumi	Integer	
10	L_bangunan	Integer	
11	Pajak_terutang	Integer	
12	Thn_pembayaran	Integer	
13	Nip	Varchar (16)	Foreign Key

### 3.12.2 Desain Tabel

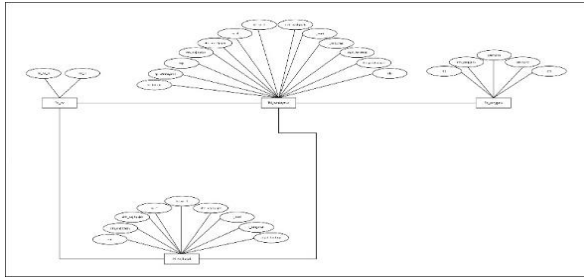
Desain tabel diterapkan sebagai berikut :



Gambar 13. Desain Tabel

Gambar diatas merupakan gambar hubungan relasi antar tabel pada sistem informasi pajak bumi dan bangunan yang dapat dijelaskan sebagai berikut; 1. Tabel wajib pajak memiliki hubungan dengan tabel pembayaran. *Primary key* pada tabel wajib pajak adalah nop, yang dihubungkan dengan nop yang ada pada tabel pembayaran sebagai *foreign key*. 2. Tabel wajib pajak juga memiliki hubungan dengan tabel RW/RT. *Primary key* pada tabel RW/RT adalah Kode RW/RT, yang dihubungkan dengan kode RW/RT yang ada pada tabel wajib pajak sebagai *foreign key*. 3. Tabel RW/RT memiliki hubungan dengan tabel pembayaran. *Primary key* pada tabel RW/RT adalah kode RW/RT, yang dihubungkan dengan kode RW/RT pada tabel pembayaran sebagai *foreign key*. 4. Tabel pembayaran juga memiliki hubungan dengan tabel pengguna. *Primary key* pada tabel pengguna adalah NIP, yang dihubungkan dengan NIP yang ada pada tabel pembayaran sebagai *foreign key*.

### 3.12.3 Entity Relation Diagram (ERD)



Gambar 14. Entity Relation Diagram (ERD)

Gambar 17. Antarmuka Master Pengguna

### 3.13 Desain User Interface

#### 3.13.1 Antarmuka Login

Desain Antarmuka Form Login digunakan untuk menginput username dan password agar dapat mengakses system. Bentuk rancangan seperti gambar dibawah ini :

Gambar 15. Antarmuka Login

#### 3.13.2 Antarmuka Menu Utama

Desain Antarmuka Form Menu Utama digunakan untuk menampilkan menu-menu utama setelah melakukan login. Bentuk rancangan seperti gambar dibawah ini :

Gambar 16 Antarmuka Menu Utama

#### 3.13.3 Antarmuka Master Pengguna

Desain Antarmuka Form Master Pengguna digunakan untuk menampilkan data-data pengguna. Selain itu form ini juga dapat menginput data pengguna, mengedit data pengguna dan menghapus data pengguna. Bentuk rancangan seperti gambar dibawah ini :

#### 3.13.4 Antarmuka Master Wajib Pajak

Desain Antarmuka *Form* Master Wajib Pajak digunakan untuk menampilkan data-data wajib pajak. Selain itu *form* ini juga dapat menginput data wajib pajak, mengedit data wajib pajak dan menghapus data wajib pajak. Bentuk rancangan seperti gambar dibawah ini :

Gambar 18. Antarmuka Master Wajib Pajak

#### 3.13.5 Antarmuka Master RT/RW

Desain Antarmuka *Form* Master RT/RW digunakan untuk menampilkan data-data RT/RW. Selain itu *form* ini juga dapat menginput data RT/RW, mengedit data RT/RW dan menghapus data RT/RW. Bentuk rancangan seperti gambar dibawah ini :

Gambar 19 Antarmuka Master RT/RW

#### 3.13.6 Antarmuka Pembayaran

Desain Antarmuka *Form* Pembayaran dapat menampilkan data-data wajib pajak. Selain itu fungsi utama dari form ini adalah untuk melakukan pencatatan pembayaran pajak. Bentuk rancangan seperti gambar dibawah ini :

Gambar 20 Antarmuka Pembayaran

### 3.13.7 Antarmuka Data Pembayaran

Desain Antarmuka Form Data Pembayaran digunakan untuk menampilkan data-data pembayaran. Bentuk rancangan seperti gambar dibawah ini :

Gambar 21 Antarmuka data Pembayaran

### 3.13.8 Antarmuka Laporan Data Wajib Pajak

Desain Antarmuka Form Laporan Data Wajib Pajak digunakan untuk menampilkan laporan data wajib pajak. Bentuk rancangan seperti gambar dibawah ini :

Gambar 22 Antarmuka Laporan Data Wajib Pajak

### 3.13.9 Antarmuka Laporan Bulanan Pembayaran

Desain Antarmuka Form Laporan Bulanan Pembayaran digunakan untuk menampilkan laporan bulanan pembayaran. Bentuk rancangan seperti gambar dibawah ini :

Gambar 23 Antarmuka Laporan Bulanan Pembayaran

### 3.13.10 Antarmuka Laporan Tahunan Pembayaran

Desain Antarmuka Form Laporan Tahunan Pembayaran digunakan untuk menampilkan laporan tahunan pembayaran. Bentuk rancangan seperti gambar dibawah ini :

Gambar 24 Antarmuka Laporan Tahunan Pembayaran

### 3.13.11 Antarmuka Ganti Password

Desain Antarmuka Form Ganti Password digunakan untuk mengganti password pengguna. Bentuk rancangan seperti gambardibawah ini :

Gambar 25 Antarmuka Ganti Password

### 3.13.12 Antarmuka Backup Restore Database

Desain Antarmuka Form Backup Restore Database digunakan untuk menampilkan kembali data yang telah dihapus. Bentuk rancangan seperti gambar dibawah ini :

Gambar AntarMuka Backup Restore Database

## 4. Hasil dan Pembahasan

### 3.1. Deskripsi Hasil Penelitian

Pembangunan form pada tulisan ini dibuat dengan bahasa pemrograman Visual Basic.Net, yang berisikan hasil atau tampilan sebagai berikut :

### 3.1.1 Tampilan *Form Menu Login*



Gambar 3.1 Tampilan *Form Menu Login*

Pada gambar tampilan diatas, terdapat jejeran menu yang bisa digunakan untuk pengolahan data, namun menu-menu tersebut belum bisa di akses. Untuk dapat mengakses menu-menu tersebut diperlukan terlebih dahulu melakukan validasi pada *form login* dengan cara 1. masukan *username*, 2. masukan *password*, 3. klik ikon “lihat *password*” tepat disebelah kotak kecil hingga menampilkan tanda centang untuk meyakinkan *password* yang di *input* sudah benar atau masih salah, 4. klik “login” jika merasa *username* dan *password* yang dimasukan sudah benar, agar dapat mengakses program ke tahap selanjutnya, klik batal jika berubah pikiran, atau tidak ingin mengakses program selanjutnya.

### 3.1.2 Tampilan *Form Menu Utama*

Tampilan ini merupakan tampilan pertama atau halaman menu utama pada program, setelah melewati tahap validasi atau *login*. Pada halaman utama ini, menu-menu yang terkunci yang terlihat pada halaman *login*, sudah dapat di akses. Berikut adalah gambar dari tampilan menu utama.



Gambar 3.2 Tampilan *Form Menu Utama*

Setiap menu yang berada ditampilan menu utama ini, terdapat fungsi dan kegunaanya masing-masing. Disinilah peran penting sebuah program melakukan setiap perintah-perintah *user*, karena *form* menu utama ini merupakan induk dari segala *form*, sehingga tampilanya pun terlihat sangat sederhana.

### 3.1.3 Tampilan *Form Master Pengguna*

*Form* master pengguna merupakan *form* pertama yang ada pada halaman master/*form* master. Pada *system* ini terdapat dua tingkatan pengguna, yakni pengguna berlevel *admin* dan pengguna berlevel *user*. Berbeda dari pengguna berlevel *user*, pada *system* ini pengguna berlevel *admin* memiliki hak penuh dalam mengakses

semua *form* yang terdapat pada *system* ini, salah satunya adalah *form* master pengguna. Peranan *admin* dalam halaman ini sangat berguna dalam menentukan level pengguna. Pada halaman ini, *admin* dapat menginput data *user*, mengedit data *user* dan menghapus data *user*. Oleh karena itu hanya pengguna berlevel *admin*lah yang dapat mengakses menu ini, agar data yang diperoleh lebih akurat. Jika terdapat kesalahan data, kelebihan data atau kekurangan data, pengguna berlevel *admin*lah yang bertanggung jawab penuh dalam hal ini.

Berikut adalah gambar dari tampilan menu master pengguna, dimana terdapat perintah-perintah yang dapat mendukung *user* berlevel *admin* dalam melakukan tugas dan tanggung jawabnya.



Gambar 3.3. Tampilan *Form Master Pengguna*

### 3.1.4 *Form* Master RT/RW

*Form* master RT/RW merupakan *form* kedua yang ada pada halaman master/*form* master. *Form* ini digunakan sebagai tempat penyimpanan data RT/RW, khusus pada wilayah Kelurahan Potulando. Pada *form* ini *admin* juga dapat mengedit dan menghapus data RT/RW jika ada perubahan ataupun pemekaran wilayah. Oleh karena itu hanya pengguna berlevel *admin*lah yang dapat mengakses menu ini, agar data yang diperoleh lebih akurat. Jika terdapat kesalahan, kelebihan data atau kekurangan data, pengguna berlevel *admin*lah yang bertanggung jawab penuh dalam hal ini.



Gambar 3.4 Tampilan *Form Master RT/RW*

### 3.1.5 *Form* Master Wajib Pajak

*Form* master wajib pajak merupakan *form* ketiga yang ada pada halaman master/*form* master. *Form* ini berfungsi sebagai tempat penyimpanan data wajib pajak. Pada *form* ini *admin* juga dapat mengedit, dan menghapus data wajib pajak, jika terjadi perubahan



atau terhapusnya wajib pajak atas pemindahan ahli waris dan lain sebagainya. *Form* ini memiliki peranan yang sangat penting, karena data yang di *input* pada *form* ini juga akan ditampilkan pada *form* pembayaran, yang bertujuan untuk menghemat waktu pelayanan pencatatan pembayaran. Oleh karena itu *form* ini hanya dapat diakses oleh pengguna berlevel *admin* saja. Jika terdapat kesalahan data, kelebihan data atau kekurangan data, pengguna berlevel *admin*lah yang bertanggung jawab penuh dalam hal ini.

Berikut adalah gambar dari tampilan menu wajib pajak yang berisi data wajib pajak, dan perintah-perintah dalam menginput, mengedit dan menghapus data wajib pajak.



Gambar 4.5 Tampilan *Form* Master Wajib Pajak

### 3.1.6 *Form* Pembayaran

*Form* pembayaran adalah *form* yang berfungsi sebagai tempat pencatatan pembayaran pajak. Seperti yang telah dijelaskan pada *form* data wajib pajak, bahwa data yang telah di *input* pada *form* data wajib pajak akan terekam atau tersimpan otomatis pada *form* pembayaran. Jadi untuk melakukan pencatatan pembayaran, *admin* atau *user* hanya perlu mengklik nama wajib pajak pada data yang telah disiapkan, maka akan menampilkan *form* data pembayaran beserta data diri wajib pajak. Sehingga *admin* atau *user* hanya perlu menginput data yang dapat berubah-ubah, seperti waktu pencatatan pembayaran atau nomor kuitansinya. Selain pengguna berlevel *admin* yang dapat mengakses *form* ini, pengguna berlevel *user* juga dapat mengaksesnya. Hal ini dikarenakan pada halaman ini, tidak memiliki perintah-perintah seperti edit data, ataupun hapus data. Jika terjadi kesalahan yang bersifat merugikan, tentu *user* berlevel *admin*lah orang yang bertanggung jawab penuh dalam hal ini.

Berikut adalah gambar dari tampilan *Form* Pembayaran.



Gambar 4.6 Tampilan *Form* Pembayaran

### 3.1.7 *Form* Data Pembayaran

*Form* data pembayaran merupakan *form* yang berisi data-data wajib pajak yang telah melakukan pembayaran. Selain pengguna berlevel *admin* yang dapat mengakses *form* ini, pengguna berlevel *user* juga dapat mengaksesnya karena pada halaman ini tidak terdapat perintah-perintah seperti hapus data, edit data ataupun simpan data. Data yang ada pada halaman ini merupakan salinan otomatis dari *form* master pembayaran. Hal ini tentu mempunyai tujuan yakni, agar orang yang bertanggung jawab dalam hal ini jelas dan data yang di *input* terhindar dari kecurangan, penipuan ataupun kesalahan. Jika terjadi kesalahan yang bersifat merugikan, tentu *user* berlevel *admin*lah orang yang bertanggung jawab penuh dalam hal ini.

Berikut adalah gambar dari tampilan *Form* data pembayaran.



Gambar 4.7 Tampilan *Form* Master Data Pembayaran

### 3.1.8 *Form* Laporan Data Wajib Pajak

*Form* laporan data wajib pajak merupakan *form* yang berfungsi sebagai tempat penyimpanan laporan data wajib pajak. Laporan data wajib pajak pada *form* ini dapat dicetak/print dan dapat di unduh. Selain pengguna berlevel *admin* yang dapat mengakses *form* ini, pengguna berlevel *user* juga dapat mengaksesnya.

Berikut adalah gambar dari tampilan *Form* Laporan Data Wajib Pajak.



Gambar 4.8 Tampilan *Form* Laporan Data Wajib Pajak

### 3.1.9 *Form* Laporan Bulanan Pembayaran

*Form* pada gambar dibawah ini merupakan *form* laporan tahunan. *Form* ini berfungsi membuat laporan tahunan otomatis. Laporan pada *form* ini dapat dicetak/print dan dapat di unduh. Selain pengguna berlevel *admin* yang dapat mengakses *form* ini, pengguna berlevel *user* juga dapat mengaksesnya.

Berikut adalah gambar dari tampilan *Form* Laporan Bulanan Pembayaran.

Gambar 4.9 Tampilan *Form* Laporan Bulanan Pembayaran

### 3.1.10 *Form* Laporan Tahunan Pembayaran

*Form* pada gambar dibawah ini merupakan *form* laporan tahunan. *Form* ini berfungsi membuat laporan tahunan otomatis. Laporan pada *form* ini dapat dicetak/print dan dapat di unduh. Selain pengguna berlevel *admin* yang dapat mengakses *form* ini, pengguna berlevel *user* juga dapat mengaksesnya.

Berikut adalah gambar dari tampilan *Form* Laporan Tahunan Pembayaran.

Gambar 4.10 Tampilan *Form* Laporan Tahunan Pembayaran

### 3.1.11 *Form* Ganti Password

Gambar dibawah ini merupakan *form* yang berfungsi untuk mengganti *password*. *Form* ini dapat diakses oleh pengguna berlevel *admin* maupun pengguna berlevel *user*.

Gambar 4.11 Ganti *Password*

### 3.1.12 *Form* Backup Restore Database

Gambar dibawah ini merupakan *form* yang berfungsi untuk memanggil kembali data yang telah dihapus, baik sengaja maupun tidak sengaja. Dan hanya pengguna berlevel *admin* saja yang dapat mengakses halaman ini.

Gambar 4.12 Tampilan *Form* Backup Database

## 3.2 Uji Coba Hasil Penelitian

Teknik pengujian yang digunakan pada penelitian ini menggunakan *Blackbox Testing*.

Tabel 4.1 Uji Coba Hasil Penelitian

N	Menu yang di uji	Skenario	Hasil yang diharapkan	Keterangan
1	<i>Form</i> Menu Login	User melakukan login dengan menginput <i>username</i> dan <i>password</i> di menu <i>login</i> , kemudian klik "Login"	User berhasil masuk ke menu utama	Berhasil
	<i>Form</i> Menu Login	User melakukan login dengan menginput <i>password</i> atau <i>username</i> yang salah	Menampilkan pesan "Username /Password Salah!" dan tidak dapat menampilkan menu utama	Berhasil
	<i>Form</i> Menu Login	User melakukan pengecekan <i>password</i> , apakah <i>password</i> yang diinput benar atau salah dengan memberikan centang pada perintah <i>system</i> "Lihat Password"	<i>Password</i> berhasil ditampilkan	Berhasil
2	<i>Form</i> Master Pengguna	Admin menginput data user kedalam sistem untuk mendapatkan hak akses ke <i>system</i>	Data berhasil tersimpan ke <i>database</i> dan menampilkan pesan "Data Pengguna berhasil Disimpan"	Berhasil
	<i>Form</i> Master Pengguna	Admin mengedit data user, jika terjadi kesalahan penginputan data user	Data Kembali tersimpan dan menampilkan pesan "Data pengguna"	Berhasil



			Berhasil Diedit”	
	Form Master Pengguna	Admin menghapus data user tertentu, agar user tersebut tidak dapat lagi mengakses system	Data user berhasil dihapus, serta menampilkan pesan “Data Pengguna Berhasil Dihapus”	Berhasil
3	Form Master Wajib Pajak	Admin menginput data Wajib pajak kedalam system	Data berhasil tersimpan ke data base dan menampilkan pesan “Data Wajib Pajak berhasil Disimpan”	Berhasil
	Form Master Wajib Pajak	Admin mengedit data wajib pajak jika terjadi kesalahan penginputan data wajib pajak	Data Kembali tersimpan dan menampilkan pesan “Data Wajib Pajak Berhasil Diedit”	Berhasil
	Form Master Wajib Pajak	Admin menghapus data wajib pajak tertentu	Menampilkan pesan “Yakin Anda Ingin Menghapus Pengguna ini?”” Jika admin mengklik “Yes”, maka akan menampilkan pesan “Data Wajib Pajak Berhasil Dihapus” Jika admin mengklik “No” maka akan Kembali ketampilan wajib pajak semula.	Berhasil
4	Form Master RT/RW	Admin menginput data RT/RW	Data berhasil tersimpan ke data base dan menampilkan pesan “Data RT/RW berhasil DiSimpan”	Berhasil
	Form Master RT/RW	Admin mengedit data RT/RW, jika terjadi kesalahan	Data Kembali tersimpan dan menampilkan	Berhasil

		penginputan data RT/RW	an pesan “Data RT/RW Berhasil Diedit”	
	Form Master RT/RW	Admin menghapus data RT/RW tertentu	Data RT/RW berhasil dihapus, serta menampilkan pesan “Data Berhasil Dihapus”	Berhasil
5	Form Master Pembayar an	Admin menginput data pembayaran wajib pajak kedalam system	Data berhasil tersimpan ke database dan menampilkan pesan “Data Pembayara n berhasil Disimpan”	Berhasil
6	Form Master Data Pembayar an	Data yang di input pada Master pembayaran akan ditampilkan pada master Data Pembayaran	Data pembayaran yang dinput pada Master pembayaran berhasil ditampilkan pada Master Data Pembayara n	Berhasil
7	Form Laporan Data Wajib Pajak	Admin mengecek laporan data wajib pajak	Laporan data wajib pajak berhasil ditampilkan	Berhasil
	Form Laporan Data Wajib Pajak	Admin mencetak laporan data wajib pajak	Laporan data wajib pajak berhasil dicetak / print	Berhasil
	Form Laporan Data Wajib Pajak	Admin mengunduh laporan data wajib pajak	Laporan data wajib pajak berhasil diunduh	Berhasil
8	Form Laporan Bulanan Pembayar an	Admin mengecek laporan pembayaran bulanan	Laporan bulanan pembayaran berhasil ditampilkan	Berhasil
	Form Laporan Bulanan Pembayar an	Admin mencetak laporan Bulanan Pembayaran	Laporan bulanan pembayaran berhasil dicetak / print	Berhasil
	Form Laporan Bulanan Pembayar an	Admin mengunduh laporan bulanan pembayaran	Laporan bulanan pembayaran berhasil diunduh	Berhasil
9	Form Laporan Tahunan Pembayar an	Admin mengecek laporan pembayaran Tahunan	Laporan tahunan pembayaran berhasil ditampilkan	Berhasil

	Form Laporan Tahunan Pembayaran	Admin mengecek laporan pembayaran Tahunan	Laporan tahunan pembayaran berhasil ditampilkan	Berhasil
	Form Laporan Tahunan Pembayaran	Admin mengecek laporan pembayaran Tahunan	Laporan tahunan pembayaran berhasil ditampilkan	Berhasil
10	Form Ganti Password	Admin mengganti password lama dengan password yang baru	Password berhasil dirubah	Berhasil
	Form Ganti Password	Admin batal mengganti password	Tampilan berubah menjadi ketampilan awal form ganti password	Berhasil
11	Form Backup Restore Database	Admin dapat mengkoneksikan ke database	Koneksi database berhasil	Berhasil
	Form Backup Restore Database	Admin dapat membackup database	Database berhasil dibackup	Berhasil
	Form Backup Restore Database	Admin dapat merestore database	Database berhasil direstore	Berhasil

Dilihat dari tabel diatas, penulis menyimpulkan bahwa Sistem Informasi Pajak Bumi dan Bangunan siap diimplementasikan pada Kantor Kantor Kelurahan Potulando. Karna telah selesai dilakukan uji coba dengan hasil yang memuaskan.

#### 4. Kesimpulan

Berdasarkan pembahasan yang telah diuraikan dari bab-bab sebelumnya, dapat diambil beberapa kesimpulan yang berkaitan dengan sistem informasi pajak bumi dan bangunan yaitu, dengan adanya sistem informasi pajak bumi dan bangunan, dapat meningkatkan proses pelayanan perpajakan dengan cepat dan praktis

dibandingkan dengan proses manual. Memudahkan proses pencatatan identitas wajib pajak. Mampu memperoleh data wajib pajak yang diinginkan dengan cepat. Memudahkan proses pembuatan laporan pajak baik laporan bulanan maupun tahunan.

#### Referensi

- [1] J. Jayinudin, "Perancangan Sistem Informasi Akuntansi Pajak Bumi Dan Bangunan (PBB) Pada Kantor Kecamatan Cileunyi Bandung Dengan Menggunakan Microsoft Visual Basic 6.0 Dan Microsoft SQL Server 2000 Berbasis Client Server," *is Best Account. Inf. Syst. Inf. Technol. Bus. Enterp. this is link OJS us*, vol. 2, no. 2, pp. 288–300, 2017, doi: 10.34010/aisthebest.v2i2.1825.
- [2] R. R. Fatturohman and R. Ilyasa, "Sistem Informasi Pendataan Pajak Daerah Pada Badan Keuangan Daerah Kota Cirebon Berbasis Web Menggunakan Metode Self Assessment System," *J. Digit.*, vol. 10, no. 2, p. 161, 2020, doi: 10.51920/jd.v10i2.167.
- [3] S. Adiman, M. Si, D. Fakultas, S. Sains, U. Pembangunan, and P. Budi, "Jurnal Perpajakan Page 13," vol. 1, no. 2, pp. 12–28, 2019.
- [4] A. H. Hendri and Mochammad Arief Sutisna, "Article Desktop Based National Police Commission Activities Information System," *J. CoSciTech (Computer Sci. Inf. Technol.)*, vol. 2, no. 1, pp. 14–23, 2021, doi: 10.37859/coscitech.v2i1.2393.
- [5] Ali Blazing, *PEMROGRAMAN WINDOWS DENGAN VISUAL BASIC.NET: Praktikum Pemrograman Vb.Net*. 2018.
- [6] I. Rickyanto, *Membuat Aplikasi Windows dengan Visual Basic .NET*. 2003.
- [7] J. Enterprise, *MySQL untuk Pemula*. 2014.
- [8] W. Komputer, *Panduan Belajar MySQL Database Server*. 2010.
- [9] B. I. Pendahuluan, "Panduan pembuatan flowchart," 2017.
- [10] B. Pemeliharaan and S. Aplikasi, "DATA FLOW DIAGRAM: sebagai alat bantu desain sistem Ninuk Budiani," no. April, 2000.
- [11] P. D. F. Pack, "RISKI Entity Relationship Diagram & Praktik DBMS."
- [12] B. Pemrograman, "Dosen Pembimbing I: Silcester Sari Sai, ST., MT."
- [13] Y. P. Astuti *et al.*, "Pengembangan Sistem Informasi Dengan Metode Waterfall Untuk Pengarsipan Data Wajib Pajak," vol. 16, no. 2, pp. 106–113, 2017.
- [14] D. N. Jasmine and A. U. Bani, "Rancang Bangun Sistem Pelayanan Pengurusan Pajak Pada Kantor Konsultan Pajak R. Nursalim & Rekan," vol. 9, no. 3, 2021.
- [15] A. Wahyudi, "Perancangan Sistem Menggunakan Metode Sdlc," *J. Din. Inform.*, vol. 4, no. 2, pp. 1–11, 2018.