

I. PENDAHULUAN

1.1 Latar Belakang

Pengolahan data adalah suatu sistem yang memberikan informasi laporan berupa laporan nilai serta informasi yang berkaitan dengan berbasis *web* sehingga dapat membantu kecepatan dan kualitas dalam penyampaian informasi. Pengolahan data ialah proses (*processing*) atau manipulasi data ke dalam bentuk yang lebih informatif (Rosidah, 2018).

Badan Pengelolaan Pajak dan Retribusi Daerah (BPPRD) Kota Metro adalah salah satu badan publik yang memiliki tugas penyelenggaraan berkaitan dengan pendapatan daerah khususnya pendapatan Kota Metro. BPPRD Kota Metro mempunyai tugas dan kewajiban membantu Walikota dalam menyelenggarakan urusan bidang pengelolaan pajak dan retribusi. Bidang-bidang di BPPRD Kota Metro yaitu bidang umum dan kepegawaian, bidang keuangan, bidang pendaftaran dan pendataan, bidang penetapan dan penagihan, bidang pembukuan dan pengendalian. di setiap bidang terdapat bagian yang memiliki fungsi dan tugasnya masing-masing dan memiliki 52 pegawai. Kantor BPPRD Kota Metro berlokasi di Jl. JEND. A.H.Nasution, No. 05, Kota Metro yang terdiri dari 2 lantai.

Piutang adalah tuntutan perusahaan kepada pihak perusahaan lain dikarenakan adanya berhutang di waktu sebelumnya dalam bentuk uang, barang, jasa atau dalam bentuk lainnya penangih harus melakukan penagihan pada pihak terhutang sesuai dengan tanggal jatuh temponya (Rompas dkk., 2018).

Pajak PBB merupakan pajak yang dikenakan atas bumi dan bangunan dan dimanfaatkan oleh badan perorangan atau pribadi, kecuali kawasan yang digunakan untuk kegiatan perhutanan, perkebunan dan pertambangan. Sesuai dengan sektor usaha perhutanan, perkebunan dan pertambangan. Sedangkan untuk sektor usaha tersebut, usaha lainnya masih dipungut oleh pemerintahan pusat (Tambingon dkk., 2019) Sejak tahun 2014, ruang pelayanan PBB telah terhubung secara *online* dengan PT. Bank Lampung selaku tempat pembayaran PBB sehingga data realisasi dapat *termonitor* dan *terupdate* secara *real time*.

Dalam pengolahan data piutang pajak PBB di bidang pembukuan BPPRD Kota Metro terdapat laporan penerimaan data piutang pajak PBB. Pegawai terlebih dahulu melihat laporan harian penerimaan kemudian diinputkan menurut tahun dan mengelompokkannya untuk mengetahui piutang. Setelah itu data jumlah dan denda akan diinputkan ke dalam *file* laporan penerimaan pertahun dengan menggunakan *Microsoft Excel* hal ini sering terjadi kesalahan pada saat penginputan penerimaan data piutang pajak PBB yang mengakibatkan selisih data piutang dan dapat terjadi redudansi data maupun kehilangan data. Pada saat pembuatan laporan di akhir bulan data akan dicocokkan dengan bidang lain.

Adanya masalah tersebut BPPRD Kota Metro memerlukan aplikasi yang memanfaatkan *database* pajak piutang agar dapat membantu mempermudah pegawai di bagian pembukuan dan pengendalian. *Database* adalah kumpulan informasi yang di simpan ke dalam komputer secara sistematis. Data tersebut akan diolah agar dapat digunakan dengan mudah dan terorganisasi ke dalam sebuah skema dan struktur tertentu dengan *software* untuk melakukan manipulasi kegunaan tertentu. keunggulan *database* ialah berkurangnya redudansi data, hubungan data yang dapat ditingkatkan, kualitas informasi yang akurat, tepat waktu dan relevan, efisiensi dalam penggunaan ruang penyimpanan, manipulasi terhadap data dengan mudah dan cepat (Puriwigati & Buana, 2020).

Framework CodeIgniter dipilih dalam perancangan ini karena memiliki kelebihan yaitu performa yang cepat, konfigurasi yang sederhana, fungsi keamanan yang baik (Rofifah, 2020).

Membuat suatu konfigurasi pada sebuah aplikasi membutuhkan metodologi atau metode pengembangan sistem dengan tujuan untuk memenuhi kebutuhan kepada pengguna, serta memberikan gambaran yang jelas dan rancang bangun kepada programmer. Metode pengembangan sistem yang digunakan dalam perancangan Aplikasi pengolahan data piutang pajak PBB ini menggunakan metode *Rapid Application Development (RAD)*. Metode RAD dipilih karena aplikasi sederhana sehingga tidak memerlukan waktu yang lama untuk pengembangan ulang setelah tahap implementasi serta menghemat biaya.

Berdasarkan permasalahan yang ada maka dibutuhkan suatu sistem yang dapat digunakan untuk mengelolah data serta memonitoring data penerimaan

piutang pajak PBB Kota Metro. Oleh karena itu, penulis mengangkat judul **“Aplikasi Pengolahan Data Piutang Pajak PBB di BPPRD Kota Metro Berbasis Web”** yang diharapkan dapat membantu petugas bidang pembukuan dalam mengelola data penerimaan piutang pajak PBB.

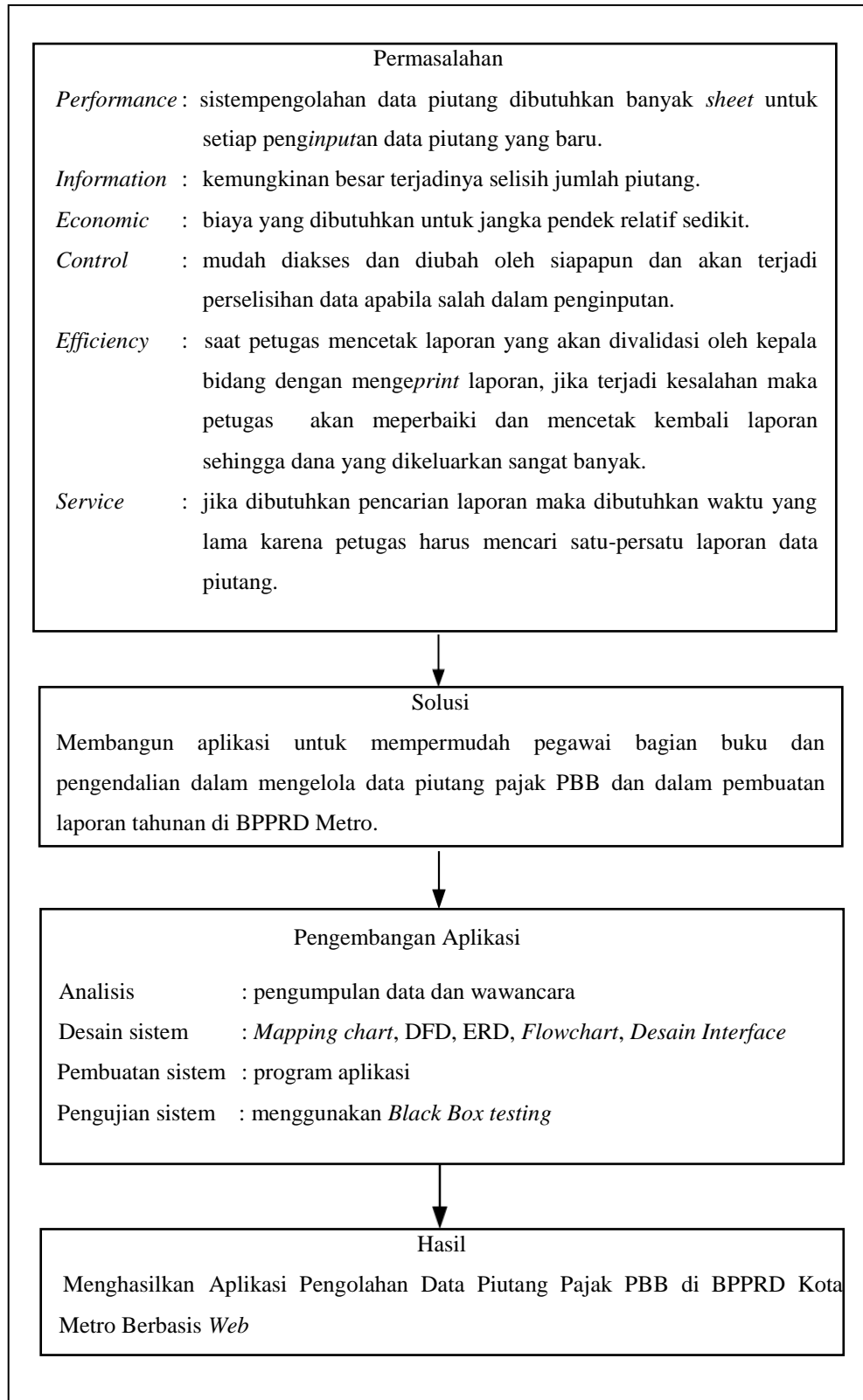
1.2 Tujuan

Tujuan pembuatan Tugas Akhir ini adalah membangun aplikasi yang diharapkan mempermudah petugas bidang pembukuan mengolah data piutang pajak PBB dan pembuatan laporan tahunan PBB di BPPRD Metro.

1.3 Kerangka Pemikiran

Setiap ruang kerja di kantor pengelolaan pajak dan retribusi daerah (BPPRD) Kota metro banyak bagian di dalamnya salah satunya yaitu bagian di bidang pembukuan dan pengendalian. Saat ini sistem pelaporan data piutang pajak PBB masih dilakukan secara manual dengan menggunakan *Microsoft Excel*. Masalah lain yang muncul adalah terjadi kesalahan dalam penginputan jumlah data piutang sehingga mengakibatkan selisih jumlah piutang dan hal tersebut membuat kesalahan dalam membuat laporan.

Berdasarkan masalah tersebut, diperlukan solusi dalam pelaporan data piutang Pajak PBB untuk memudahkan dalam pengelolaanya, maka dibuatlah **“Aplikasi Pelaporan Data Piutang Pajak PBB di BPPRD Kota Metro Berbasis Web”**. Sistem ini dapat membantu pegawai dalam mengelola dan memonitoring data piutang. Sistem ini dibangun menggunakan teknik pengembangan *System Rapid Application Development (RAD)*. Kerangka pemikiran dalam pembuatan sistem ini disajikan pada Gambar 1.



Gambar 1. Kerangka Pemikiran

1.4 Kontribusi

Aplikasi Pengolahan Data Piutang Pajak PBB di BPPRD Kota Metro Berbasis *Web* Menggunakan *Framework CodeIgniter* ini diharapkan dapat memberikan kontribusi kepada beberapa pihak antara lain:

1. Mempermudah pegawai BPPRD dalam melakukan pengolahan data piutang pajak PBB.
2. Mempermudah pegawai BPPRD dalam *penginputan* pokok, jumlah, denda data piutang dan pengecekan data piutang pajak PBB.
3. Mempermudah pegawai BPPRD dalam mendapatkan informasi pelaporan data piutang pajak PBB.
4. Membantu pegawai BPPRD dan Kepala Bidang dalam mengelola dan memonitoring pelaporan data piutang pajak PBB .
5. Membantu pegawai BPPRD dalam pembuatan laporan tahunan PBB.

II. TINJAUAN PUSTAKA

2.1 Aplikasi

Aplikasi ialah program maupun perangkat komputer yang siap dipakai oleh pengguna dan berisi sebuah *coding* atau perintah yang dapat diubah sesuai dengan keinginan. Aplikasi memiliki makna ialah pemecahan permasalahan yang memakai salah satu metode pemrosesan informasi aplikasi yang berpacu pada suatu komputasi (Setiawan, Nita, 2019).

2.2 Pengolahan data

Pengolahan data adalah suatu sistem yang memberikan informasi laporan berupa laporan nilai serta informasi yang berkaitan dengan berbasis *web* sehingga dapat membantu kecepatan dan kualitas dalam penyampaian informasi. Pengolahan data ialah proses (*processing*) atau manipulasi data ke dalam bentuk yang lebih *informative* (Rosidah, 2018)

2.3 Pajak PBB

Pajak PBB merupakan pajak yang dikenakan atas bumi dan bangunan dan dimanfaatkan oleh badan perorangan atau pribadi, kecuali kawasan yang digunakan untuk kegiatan perhutanan, perkebunan dan pertambangan. Sesuai dengan sektor usaha perhutanan, perkebunan dan pertambangan. Sedangkan untuk sektor usaha tersebut, usaha lainnya masih dipungut oleh pemerintahan pusat (Tambingon dkk., 2019).

2.4 Piutang

Piutang adalah tuntutan perusahaan kepada pihak perusahaan lain dikarenakan adanya berhutang di waktu sebelumnya dalam bentuk uang, barang, jasa atau dalam bentuk lainnya penangih harus melakukan penagihan pada pihak terhutang sesuai dengan tanggal jatuh temponya (Rompas dkk., 2018).

2.5 Pegawai

Pegawai ialah seseorang yang ditugaskan sebagai pekerja dalam sebuah perusahaan tertentu untuk melakukan operasional perusahaan dan bekerja untuk digaji dan bergerak di masing-masing bidang perusahaan, pegawai sangat

berpengaruh terhadap produktivitas kerja serta loyalitas terhadap perusahaan tersebut (Fadli, 2017)

2.6 Aspek Teknis

Aspek Teknis merupakan bagian dari rincian pelaksanaan yang lebih berorientasi pada teknologi, topik HTML, PHP, *CodeIgniter*, *Bootstrap*, CSS, Basis Data, *Xampp*, *PHPMyAdmin*, *Framework*, *Sublime Text*, *JavaScript*, *web server*, MVC, DFD, ERD, *Flowchart*, *Mapping Chart*, RAD, *Black-Box Testing*.

2.6.1 Web

Web adalah media penyampaian informasi yang membuat pengguna bisa mengakses internet lewat aplikasi yang terkoneksi dengan internet dan juga *web* menyediakan data yang diperlukan untuk pengguna yang tersambung ke internet. Data yang ditampilkan kepada pengguna berbentuk data interaktif, gambar akan ditampilkan secara lebih terperinci (Nurmalasari, dkk., 2019).

2.6.2 HTML

HTML ialah bahasa standard yang digunakan untuk menampilkan halaman *web* yang dapat dilakukan dengan HTML yaitu, mengatur tampilan pada halaman *web* beserta isinya, membuat tabel dalam halaman *web*, dapat mempublikasikan halaman *web* secara *online*, pembuatan *form* yang bisa digunakan untuk menangani registrasi dan transaksi *web*. halaman *web* juga bisa digunakan buat Menulis bacaan, memasukkan foto dan sebagainya (Ayu, Permatasari, 2018).

2.6.3 PHP

PHP kepanjangan dari *Hypertext Preprocessor* ialah salah satu bahasa pemrograman *open source* yang sangat sesuai ataupun dikhususkan buat pengembangan *web* serta bisa ditanamkan pada suatu skripsi HTML. Bahasa PHP Dapat menggambarkan sebagian bahasa pemrograman semacam C, *Java*, serta *Pearl* dan mudah untuk dipelajari (Rahmat, 2018).

2.6.4 CodeIgniter

CodeIgniter merupakan sebuah framework untuk membuat sebuah website yang bersifat *open source*. pengembangan proyek menjadi lebih mudah dan cepat *CodeIgniter* pula telah memakai konsep MVC (*Models View-Controller*) yang memisahkan antara informasi serta presentasi sehingga membolehkan

pengembangan suatu *web* dengan kilat dan mempermudah proses pengelolaan *web* tersebut (Sitinjak, Suwita, 2020).

Ada pula fitur-fitur yang terdapat pada CodeIgniter:

1. *Model-View-Controller*, yang digunakan buat pengembangan aplikasi *web* Dengan memisahkan antara logika bisnis serta etika.
2. *Support* terhadap PHP 4 serta PHP 5.
3. *Template Engine Class*, buat memisahkan informasi serta presentasi.
4. *Scaffolding*, membolehkan untuk membangun aplikasi dengan kilat serta Mudah.

2.6.5 Bootstrap

Bootstrap ialah sesuatu *framework* yang bisa menyelesaikan kasus dalam mendesain *web*. *Bootstrap* bersifat *responsive* yang artinya *bootstrap* dapat digunakan di banyak *platform*, maksudnya tampilan halaman *web* yang memakai *bootstrap* ini hendak nampak senantiasa apik, baik *tipemobile* ataupun *dekstop* (Zakir, 2016).

2.6.6 Cascading Style Sheet (CSS)

CSS ialah suatu *style sheet* yang berisi rangkaian intruksi digunakan untuk menentukan bagaimana suatu text akan tampil pada halaman *web*. Dengan menggunakan CSS, pengformatan *web* bisa dicoba secara kilat serta efektif (Raymond, dkk., 2016).

2.6.7 Basis Data

Basis Data adalah sistem penyimpanan dan kumpulan data yang berelasi dan terdapat informasi yang disusun sehingga mudah untuk dapat diakses . Basis data dapat diartikan juga sebagai suatu pengorganisasian data dengan bantuan komputer (Sitinjak, Suwita, 2020) .

2.6.8 XAMPP

Xampp ialah paket *web server* berbasis *open source* yang menunjang banyak sistem pengoperasian, kompilasi dari sebagian program yang mempunyai guna sebagai *Server* yang berdiri sendiri (*localhost*), dan berdiri atas program *Apache*, *PHP*, *MySQL database* secara manual (Nurmalasari, dkk., 2019).

2.6.9 PhpMyAdmin

PhpMyAdmin ialah suatu aplikasi ataupun perangkat lunak bebas (*open source*) yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk menanggulangi administrasi *database MySQL* lewat jaringan lokal ataupun internet. *PhpMyAdmin* menunjang bermacam pembedahan *MySQL*, antara lain (mengelola basis data, tabel-tabel, bidang (*fields*), kedekatan (*relations*), indeks, pengguna (*users*), perijinan (*permissions*), dll) (Standsyah, Restu, 2017).

2.6.10 Framework

Framework ialah suatu kerangka kerja atau kumpulan *script* dengan fungsi masing-masing untuk pengembangan aplikasi dalam menanggulangi bermacam permasalahan pemrograman semacam koneksi ke *database*, pemanggilan *variable*, serta *file* sehingga dapat menghemat waktu dalam membangun sebuah aplikasi (Destiningrum, Adrian, 2017).

2.6.11 Sublime Text

Sublime Text Merupakan aplikasi editor berbasis *python*, buat kode serta bacaan API yang bisa berjalan diberbagai *platform operating system*, sebuah *text editor* yang elegan dan kaya fitur. *Sublime Text* mendukung berbagai bahasa pemrograman dan sanggup menyajikan fitur *syntax highlight* (Pradiatiningtyas, 2017).

2.6.12 JavaScript

JavaScript merupakan bahasa yang dipakai untuk membuat sebuah program yang digunakan agar dokumen HTML yang ditampilkan dalam sebuah browser menjadi lebih interaktif, *javascript* memberikan beberapa fungsionalitas ke *page web*. sehingga menghasilkan aplikasi yang disajikan dengan menggunakan interface dan *web* (Yani, Saputra, 2018).

2.6.13 Web Server

Web Server yakni perangkat lunak *server client* (*web browser*) serta mempunyai protokol sendiri ialah *Hyper Text Transfer Protocol*. Dengan protokol ini, *client* yang dikenal dengan *web browser* dan mengirimkan kembali hasil dalam bentuk halaman-halaman *web* yang umumnya akan berbentuk sebuah dokumen HTML (Josi, 2017).





2.6.14 Model-View-Controller (MVC)

MVC ialah pola konsep yang digunakan untuk membolehkan seseorang pengembang aplikasi memisahkan ataupun mengenkapsulasi informasi pada pemrosesan (*model*), mengisolasi dari sesi manipulasi (*controller*) serta tampilan (*view*) buat pengguna (Pastima, Kasnady, 2016). Salah satu *framework* yang menggunakan konsep MVC ini yaitu *CodeIgniter*.

2.6.15 Data Flow Diagram (DFD)

Penyajian grafik yang menggambarkan aliran data serta transformasi yang diterapkan sebagai informasi yang mengalir dari *input* serta *output* merupakan pengertian dari DFD . DFD terbuat memakai notasi- notasi untuk menggambarkan alur sistem dari proses kerja sesuatu sistem sehingga bisa membantu dalam menguasai sistem secara logika, terstruktur serta jelas. Ada pula simbol- simbol *Data Flow Diagram* tertera pada Tabel 1.

Tabel 1. Simbol-simbol DFD

Simbol	Nama	Keterangan
(1)	(2)	(3)
	<i>Terinator/ External Entity</i>	Proses menampilkan transformasi dari masukan jadi keluaran.
	Proses	Entitas eksternal dimana entitas tersebut berbicara dengan sistem.
	<i>Informasi store</i>	Penyimpanan menampilkan penyimpanan dalam suatu <i>database</i> .
	Alir data	Aliran menggambarkan aliran informasi yang masuk ke proses ataupun keluar dari sesuatu proses.



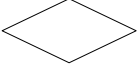
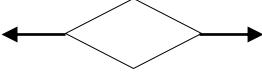
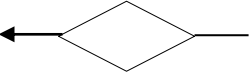
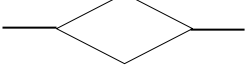
Sumber : (Shalahuddin, 2018)

2.6.16 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) ialah sekumpulan metode untuk mendeskripsikan data-data ataupun objek-objek yang terbuat bersumber pada serta berasal dari dunia nyata yang disebut entitas (*entity*) dan ikatan (*relationship*) yang menunjukkan informasi, di buat, di simpan dan di gunakan

dalam sistem bisnis (Syahrul, Desmulyati, 2019). Simbol-simbol pembuat ERD dapat dilihat pada Tabel 2.

Tabel 2. Simbol-simbol ERD



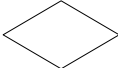





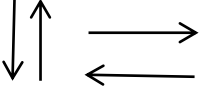
Simbol	Nama	Keterangan
(1)	(2)	(3)
	Entitas	Individu yang mewakili sesuatu objek serta bisa dibedakan dengan objek yang lain.
	Atribut	<i>Property</i> yang dimiliki oleh sesuatu entitas, dimana bisa mendeskripsikan ciri dari entitas tersebut.
	Relasi	Menunjukkan ikatan diantara beberapa entitas yang berbeda.
	Relasi 1: 1	Relasi yang menampilkan kalau tiap entitas pada himpunan entitas awal berhubungan dengan sangat banyak satu entitas pada himpunan entitas kedua
	Relasi 1: N	Relasi yang menampilkan kalau ikatan antara entitas awal dengan entitas kedua merupakan satu banding banyak ataupun kebalikannya. Tiap entitas bisa berelasi dengan banyak entitas pada himpunan entitas yang lain.
	Relasi N: N	Hubungan ini menampilkan kalau tiap entitas pada himpunan entitas yang awal bisa berhubungan dengan banyak entitas pada himpunan entitas yang kedua, demikian pula kebalikannya.

Sumber: (Edi, Betshani, 2016).


2.6.17 Flowchart

Menurut (Ridlo, 2017) *Flowchart* merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. *Flowchart* membantu analisis dalam membongkar permasalahan kedalam segmen-segmen yang lebih kecil serta menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian. *Flowchart* memudahkan penyelesaian sesuatu permasalahan khususnya kasus yang perlu dipelajari dan dievaluasi lebih lanjut. Simbol-simbol *Flowchart* dapat dilihat pada Tabel 3.

Tabel 3. Simbol-simbol *Flowchart*

Simbol	Nama	Fungsi
(1)	(2)	(3)
	Terminal	Menyatakan awal serta akhir suatu program.
	<i>Input/ Output</i>	Merepresentasikan <i>input</i> data ataupun <i>output</i> data yang diproses ataupun informasi.
	<i>Decision</i>	Menyatakan sesuatu keadaan yang menciptakan 2 kemungkinan: ya/ tidak
	<i>Connector</i>	Keluar ke ataupun masuk dari bagian lain <i>Flowchart</i> khususnya halaman yang sama
	<i>Offline Connector</i>	Digunakan sebagai beda halaman
	Masukan Manual	Input yang dimasukkan secara manual lewat keyboard
	Keluaran	Menyatakan hasil dari pemrosesan data
	<i>Document</i>	I/ O dalam format yang dicetak
	<i>Flow</i>	Menyatakan aliran arus sesuatu proses

Tabel 3. Lanjutan






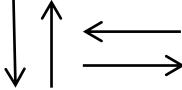



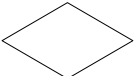
	<i>Predifined Process</i>	Menggambarkan proses yang masih berisi proses lain di dalamnya
---	---------------------------	--

Sumber: (Ridlo, 2017)

2.6.18 Mapping Chart

Suatu aliran dokumen dalam suatu program yang menampilkan alur dokumen berbentuk data laporan serta formulir merupakan pengertian dari *flowmap*. (Mawardah, 2017). Simbol-simbol *Mapping chart* dapat dilihat pada Tabel 4.

Tabel 4. Simbol-simbol *Mapping Chart*

Simbol	Nama	Keterangan
(1)	(2)	(3)
	Dokumen	Menunjukkan I/ O baik buat proses manual, mekanik, serta komputer
	Manual	Kegiatan secara manual
	Proses	Kegiatan proses dari pembedahan program Komputer
	<i>Keyboard</i>	<i>Input</i> yang terkomputerisasi
	<i>Hardisk/ Penyimpanan</i>	Digunakan sebagai tempat penyimpanan data
	Garis Alir	Menunjukkan aliran proses
	<i>Display</i>	Untuk melihat hasil keluaran ouput pada monitor
	<i>Offline Connector</i>	Penghubung untuk beda halaman.
	Arsip	Penyimpanan informasi secara manual
	<i>Decision</i>	Kondisi yang menciptakan 2 kemungkinan: ya/ tidak

Sumber: (Mawardah, 2017)

2.6.19 Rapid Application Development (RAD)

Rapid Application Development (RAD) ialah salah metode cara pengembangan perangkat lunak yang bersifat *incremental*. pengembangan aplikasi dengan menggunakan RAD dapat dilakukan dalam waktu yang relatif cepat. sistem yang dibesarkan bisa memenuhi kemauan dari para pemakai sehingga bisa dikurangi waktu buat pengembangan ulang setelah tahap implementasi (Puteri, Effendi, 2018)

Tata cara RAD memiliki 4 tahapan, ialah:

1. *Recuirements Planning*

Tahap ini dicoba analisa sehingga diperoleh informasi serta data proses bisnis yang terpaut, dan diperoleh informasi berbentuk sistem yang lagi berjalan serta dapat ditafsirkan dalam desain alir dokumen ataupun *mapping chart*.

2. *User Design*

Tahap ini ialah proses merancang alur aplikasi bersumber pada DFD (*Data Flow Diagram*) serta ERD (*Entity Relationship Diagram*).

3. *Contruction*

Tahap ini berfokus pada pembuatan program serta aplikasi yang memakai bahasa pemrograman dan perangkat lunak pendukung yang lain dalam pembangun aplikasi.

4. *Cutover*

Tahapan ini ialah tahapan terakhir dalam pengembangan aplikasi. Aplikasi yang sudah selesai dibangun, tiap komponen diuji memakai sistem pengujian *black box system*.

2.6.20 Black-Box Testing

Black-Box Testing berfokus pada spesifikasi fungsional dari fitur lunak. Tester bisa mendefinisikan kumpulan keadaan input serta melaksanakan pengetesan pada spesifikasi fungsional program *Black-Box testing* digunakan untuk menemukan kesalahan pada fungsi yang hilang ataupun tidak benar (Jaya, 2018).

2.7 Penelitian Terkait

Penelitian terkait ialah teori dari banyak sumber riset yang dijadikan pedoman penyusunan tugas akhir ini. Dalam penataan kajian pustaka ini, penulis menjadikan sebagian tulisan sebagai rujukan dalam pembuatan “Aplikasi Pelaporan Data Piutang Pajak PBB di BPPRD Kota Metro Berbasis *Web*” oleh antara lain merupakan jurnal yang berjudul “Aplikasi Perhitungan Pajak Bumi dan Bangunan Berbasis *Web*” oleh Azhar, dkk., (2016) yang bertujuan untuk mempermudah petugas pajak dalam proses menghitung pajak yang sebelumnya masih menggunakan *Microsoft Excel*, dan mempermudah wajib pajak untuk mengurus pajak PBB. Maka dispenda memerlukan sebuah aplikasi yang dapat menyediakan laporan penerimaan pembayaran PBB dan laporan Akhir tahunan.

Karya ilmiah berikutnya berjudul “Sistem Informasi Pelaporan Data Pajak PBB-P2 berbasis *web* Pada Dispenda Kota Bandar Lampung” oleh Karim, (2017) Sistem Informasi ini bertujuan untuk membantu petugas pajak di kecamatan dan kelurahan agar mempermudah dalam melakukan penagihan PBB-P2 berdasarkan data yang tertera pada *website*. Adapun manfaat dari penelitian ini adalah menghasilkan sebuah sistem pelaporan data yang mudah untuk digunakan bukan hanya bagi petugas penagih pajak di kecamatan melainkan kelurahan, akan tetapi pimpinan akan menjadi bahan evaluasi terhadap realisasi pembayaran PBB-P2.

Karya ilmiah lain dalam jurnal berjudul “Sistem Informasi monitoring Pajak Bumi dan Bangunan Berbasis *web*” oleh Sri, dkk., (2020) Tujuan dari penelitian ini adalah pembuatan sistem informasi serta monitoring Pajak Bumi dan Bangunan Berbasis *web* ini agar mempermudah dalam mengelola data pembayaran pajak bumi dan bangunan, dapat mengelola total pajak terhutang baik yang telah lunas maupun yang belum lunas pertahun serta dapat menyajikan informasi data pajak bumi dan bangunan serta dalam pembuatan laporan yang tepat yaitu laporan PBB setiap kepala desa dan laporan keterlambatan berdasarkan status pembayaran wajib pajak tersebut.

Berdasarkan jurnal terkait untuk dapat melihat perbandingannya bisa ditunjukkan pada Tabel 5.

Tabel 5. Jurnal terkait

No	Nama dan Tahun	Judul	Metode	
			Pengembangan Sistem	Hasil
1.	Azhar, (2016)	Aplikasi Pelaporan Data Piutang Pajak PBB di BPPRD Kota Metro Berbasis <i>Web</i>	Metode SDLC <i>Waterfall</i>	Aplikasi berbasis <i>Web</i> yang memudahkan petugas pajak dalam proses menghitung pajak, dan mengelola laporan penerimaan pembayaran pajak PBB.
2.	Karim, (2017)	Sistem Informasi Pelaporan Data Pajak PBB-P2 berbasis web Pada Dispenda Kota Bandar Lampung	Metode <i>Waterfall</i>	Sistem Informasi Berbasis <i>Web</i> yang memudahkan dalam membantu petugas penagih pajak dalam melakukan penagihan pajak PBB.
3.	Sri, dkk., (2020)	Sistem Informasi monitoring Pajak Bumi dan Bangunan Berbasis web	Metode <i>Prototype</i>	sistem informasi yang memudahkan serta monitoring Pajak Bumi dan Bangunan Berbasis web ini agar mempermudah dalam mengelola data pembayaran pajak bumi dan bangunan