

BAB II

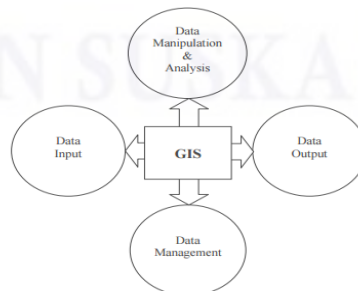
LANDASAN TEORI

2.1 Sistem Informasi Geografis

Sistem Informasi Geografis (SIG) adalah sistem komputer yang dirancang untuk menangkap, menyimpan, memanipulasi, menganalisa, mengelola, dan menyajikan semua jenis data geografis. Penjelasan paling sederhana adalah bahwa SIG adalah penggabungan kartografi dan teknologi basis data SIG. Sistem SIG digunakan dalam bentuk kartografi, penginderaan jarak jauh, survei, utilitas pengelolaan, fotogrametri, geografi, perencanaan kota, pengelolaan keadaan darurat, navigasi dan mesin pencari lokal. Aplikasi SIG adalah alat yang memungkinkan pengguna membuat *query interactive* (permintaan pengguna), menganalisis informasi spasial, mengedit data, memetakan, dan menyajikan hasil dari semua operasi ini, (Baros & Stojanovic, 2015).

2.1.1 Sub Sistem Informasi Geografis

Menurut Prahasta (2009), Sistem Informasi Geografis dapat diuraikan menjadi empat subsistem, yaitu: (1) data *input*; (2) data *output*; (3) data *management*; dan (4) data *manipulation*. Data input mengumpulkan dan mempersiapkan data spasial. Data output berhubungan dengan menampilkan atau menghasilkan keluaran. Data *management* mengorganisasikan data spasial dan atribut. Data *manipulation* dan *analysis* menentukan informasi yang dapat dihasilkan oleh SIG dan melakukan manipulasi serta pemodelan data untuk menghasilkan informasi yang diharapkan. Subsistem SIG menurut Prahasta (2009) dapat di lihat pada Gambar 2.1.



Gambar 2.1 Subsistem SIG

(Sumber: Prahasta dalam Wijaya, 2012)

2.1.2 Model Data Sistem Informasi Geografis

Dalam SIG, sebuah konstruksi matematis untuk mewakili objek geografis atau permukaan sebagai data. Sebagai contoh, model data vektor mewakili geografi sebagai kumpulan *point*, *polyline* dan *polygon*. Model data raster mewakili geografi sebagai matriks sel yang menyimpan nilai numerik dan model data TIN (*Triangulated Irregular Network*) mewakili geografi sebagai kumpulan segitiga bersebelahan dan tidak tumpang tindih, (ESRI).

1. Model Data Spasial

Data spasial merupakan data yang berkaitan dengan lokasi keruangan yang umumnya berbentuk peta, sedangkan data atribut merupakan data tabel yang berfungsi menjelaskan keberadaan berbagai objek sebagai data spasial (Ruhimat, 2010).

2. Model Data Vektor

Model data vektor terletak pada titik dengan nilai koordinat (x,y) untuk membangun objek spasial. Objek yang dibangun terbagi menjadi tiga bagian lagi, yaitu: (1) titik (*point*); (2) garis (*line*); dan (3) area (*polygon*), (Prahasta, 2009). Dari data titik (*point*) representasi grafis sederhana pada suatu objek. Titik tidak mempunyai dimensi tetapi dapat ditampilkan dalam bentuk simbol baik pada peta. Garis (*line*) berupa bentuk *linear* yang menghubungkan dua atau lebih titik dan merepresentasikan objek. Contoh: jalan, sungai dan lain-lain. Area (*polygon*) representasi objek dalam dua dimensi, format vektor dapat diperoleh melalui metode terestis dengan alat-alat *survey digital*.

- Metode terestis dengan *Global Positioning System* (GPS).
- Metode digitasi terhadap peta *hardcopy* yang sudah ada.
- Metode vektorisasi terhadap peta digital (format *raster*) yang sudah ada.

3. Model Data Raster

Pendekatan paling sederhana untuk menyusun data spasial adalah dengan menggunakan sel *grid* untuk mewakili bagian bumi yang diukur yang disebut SIG *grid* atau *raster* SIG. Dalam SIG *raster*, berbagai metode yang

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

berbeda digunakan untuk mengkodekan entitas spasial untuk penyimpanan dan representasi di komputer.

Sel-sel di setiap garis gambar dicerminkan oleh baris angka yang setara dalam struktur. Garis struktur *file* menunjukkan jumlah baris, jumlah kolom dan nilai sel maksimum pada gambar. Representasi *raster* mengasumsikan bahwa ruang geografis dapat diperlakukan seolah-olah itu adalah permukaan *cartesian* yang rata. Setiap *pixel* atau sel *grid* kemudian diasosiasikan dengan bidang tanah persegi. Resolusi atau skala data *raster* adalah hubungan antara ukuran sel dalam *database* dan ukuran sel di tanah (Reddy, 2008).

Data raster digunakan sebagai tampilan latar belakang (*background*) untuk suatu *layer* dari objek yang lain (*vector*). format data *raster* dapat diperoleh melalui:

- a. *Digital photogrametry*.
- b. *Remote sensing satellite* (NOAA, *Landsat*, *Quickbird*).
- c. Metode *scanning* dengan alat *scanner*.

4. Data Atribut

Data atribut merupakan data tabel yang berfungsi menjelaskan keberadaan berbagai objek sebagai data spasial (Wijaya, 2010). Data yang mempresentasikan aspek-aspek deskriptif dari fenomena yang dimodelkan. Aspek deskriptif ini mencakup *item* atau *properties* dari fenomena yang bersangkutan hingga dimensi waktunya. Jenis data ini sering juga disebut sebagai data non-spasial. Jenis data atribut atau non-spasial digunakan oleh sistem-sistem manajemen basis data.

5. Komponen SIG

Menurut (Prahasta, 2009), SIG sebagai sistem terdiri dari beberapa komponen dengan berbagai karakteristik, yaitu: (1) perangkat keras; (2) perangkat lunak; (3) Data dan Informasi Geografis; dan (4) Manajemen. Perangkat keras sudah tersedia bagi berbagai *platform* perangkat keras, mulai dari kelas PC *desktop*, *workstations*, hingga *multi-user host* dapat digunakan banyak pengguna bersama dalam jaringan komputer yang

2.2 WebGIS

WebGIS (*Web-based GIS*) adalah SIG yang terdistribusi dalam suatu jaringan komputer untuk mengintegrasikan dan menyebarluaskan informasi geografi secara visual pada *World Wide Web*. WebGIS dibandingkan dengan *desktop* SIG menawarkan efisiensi biaya, efisiensi beban kerja sumber daya manusia untuk instalasi, pemeliharaan dan dukungan teknis, pemangkasan kurva pembelajaran untuk pengguna akhir dan keunggulan dalam hal integrasi data spasial dan data non spasial menggunakan DBMS, (Wijaya 2012).

2.3 Air

Air adalah suatu kebutuhan sangat penting dari makhluk hidup di bumi ini baik pula manusia membutuhkan air khususnya air bersih dalam kehidupan sehari-hari. Untuk memenuhi kebutuhan konsumsi dan lainnya manusia dapat menentukan jumlah air yang di butuhkan bagi kehidupan sehari-hari.

tersebar luas, berkemampuan tinggi, memiliki ruang penyimpanan (*harddisk*) yang besar dan mempunyai kapasitas memori RAM (*Random Access Memory*) yang besar. Perangkat lunak SIG bisa juga merupakan sistem perangkat lunak yang tersusun secara modular dimana sistem basis datanya memegang peranan kunci. Data dan Informasi geografis dalam SIG dapat mengumpulkan dan menyimpan data atau informasi yang di perlukan dengan melakukan digitasi data spasialnya dari peta analog dan kemudian memasukkan data atributnya dari tabel-tabel atau laporan. Manajemen, jika di kelola dengan baik dan di kerjakan oleh orang-orang yang memiliki keahlian yang tepat pada semua tingkatan.

6. Pemetaan

Peta merupakan suatu gambaran yang ada dari permukaan bumi ini yang digambarkan di bidang datar dalam proyeksi tertentu. Pemetaan merupakan proses yang dilakukan berupa pengukuran, perhitungan dan penggambaran permukaan bumi dengan menggunakan cara atau metode tertentu sehingga didapatkan hasil berupa *softcopy* dan *hardcopy*.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Air yang digunakan untuk keperluan sehari-hari yang kualitasnya memenuhi syarat kesehatan dan akan menjadi air minum setelah dimasak terlebih dahulu. Air minum adalah air yang kualitasnya memenuhi syarat kesehatan dan dapat langsung diminum, (Triatmadja, 2008). Air bersih sama dengan halnya air yang digunakan sehari-hari namun belum tentu di kategorikan untuk langsung di minum melainkan harus di masak terlebih dahulu untuk di konsumsi.

Faktor Penyebab Kekurangan Air Bersih Kehilangan air pada umumnya disebabkan karena adanya kebocoran air pada pipa transmisi dan distribusi serta kesalahan dalam pembacaan meter.

2.4 Jaringan Pipa

Pipeline atau jaringan pipa adalah sistem jaringan penghubung untuk sarana transportasi fluida produksi dari satu tempat ke tempat lainnya, dimana pipa-pipa tersebut biasanya dipendam didalam tanah, ditempatkan di atas permukaan atau ditempatkan di *sea floor*.

2.5 Kota Dumai, Rokan Hilir dan Bengkalis (DUROLIS)

Durolis adalah program dari pemerintah Dinas Pekerjaan Umum dan Penataan Ruang Provinsi Riau yang bekerjasama dengan Kota Dumai, Kabupaten Rokan Hilir dan Kabupaten Bengkalis untuk memenuhi kebutuhan air bersih di tiga daerah tersebut. Sebelumnya program durolis ini di mulai pada awal tahun 2017 terus berlanjut hingga sekarang. Program air bersih telah di lakukan oleh PDAM di setiap Kabupaten, namun mengingat PDAM sudah berdiri lama membuat penurunan ketersediaan air bersih sedangkan kebutuhan air bersih semakin bertambah.

Program Dinas PUPR kerjasama dengan tiga daerah Dumai, Rokan Hilir dan Bengkalis untuk meningkatkan air bersih dalam memenuhi kebutuhan masyarakat berdasarkan data tiap daerah:

1. Dumai

Kebutuhan Air minum ataupun air bersih mempunyai peranan yang penting dalam kehidupan masyarakat. PDAM Dumai memiliki pelanggan 1.387 pelanggan dengan produksi air minum pada tahun 2014 berjumlah 288.833m³, sejumlah 798m³ didistribusikan ke pelanggan sosial, 72.430m³

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

ke pelanggan non niaga, 22.394m³ ke pelanggan niaga, 183.626m³ ke perusahaan industri. Sedangkan di tahun 2013 berjumlah 1.386 pelanggan dengan produksi air 445.205m³ dan di tahun 2012 berjumlah 1.384 pelanggan dengan kebutuhan air 461.656m³. Data diatas menunjukan penambahan pelanggan namun produksi air bersih semakin menurun tiap tahunnya, (Data BPS Kota Dumai 2015).

2.

Rokan Hilir

Kebutuhan rumah tangga yang menggunakan air bersih sebagai sumber air minum sebanyak 53,52 persen. Sumber air minum rumah tangga di wilayah perkotaan relatif lebih baik dari pada mereka yang di perdesaan, yakni sebesar 55,41 persen untuk daerah perkotaan menggunakan air bersih, sedangkan daerah perdesaan 53 persen. Pada tahun 2013 persentase rumahtangga yang mempunyai fasilitas air minum sendiri dari 88,37 persen, fasilitas bersama 7,92 persen, fasilitas umum 1,48 persen dan sisanya tidak ada fasilitas sebanyak 2,23 persen, (Data BPS Rokan Hilir 2014).

3.

Bengkalis

Kebutuhan air bersih di kabupaten bengkalis terus meningkat setiap tahunnya hal ini berdasakan data PDAM Bengkalis. Pada tahun 2012 jumlah pelanggan PDAM bengkalis 11.557 pelanggan dan meningkat di tahun 2013 mencapai 13.149 pelanggan. Pada tahun 2014 berjumlah 13.64 pelanggan dan tahun 2015 jumlah pelanggan PDAM Bengkalis mencapai 14.246 angka ini terus meningkat setiap tahunnya, (BPS Bengkalis 2016).

2.6 Quantum GIS (QGIS)

QGIS adalah perangkat SIG *Open Source* yang *user friendly* dengan lisensi di bawah *General Public License* (GNU). QGIS merupakan proyek tidak resmi dari *Open Source Geospatial Foundation* (OSGeo). QGIS dapat dijalankan pada *Linux*, *Unix*, *Mac OSX*, *Windows* dan *Android*, serta mendukung banyak format dan fungsionalitas data *vector*, *raster*, dan *database*. QGIS menyediakan sejumlah besar kemampuan yang terus tumbuh dari fungsi inti QGIS dan *plugin* tambahan. Sehingga dapat menampilkan, manajemen, mengedit, menganalisis data, dan menyusun peta yang dapat dicetak, (<http://qgis.org>).

2.7 MapBox

Mapbox suatu penyedia peta terbesar dari situs-situs ternama yang di khususkan bagi pegembang untuk sebuah aplikasi baik itu *desktop* dan *mobile* yang terintegrasi dengan peta. Mapbox GL JS adalah pustaka *JavaScript* yang menggunakan WebGL untuk membuat peta interaktif dari *vector tile* dan *Mapbox style*. Mapbox GL adalah kumpulan perpustakaan *OpenSource* untuk menyematkan peta sisi klien yang dapat disesuaikan dan responsif di aplikasi *Web*, seluler, dan *desktop*. Dalam mapbox mempresentasikan *vector* kedalam bentuk *tile*, yaitu: (1) *vector tile*, dengan menerapkan konsep ubin dan di kembangkan untuk penyimpanan *caching*, penskalaan dan penyajian peta dengan cepat ke sebuah data *vector*. Dari data *vector tile* ini menawarkan pengolahan data *vector* untuk menampilkan data *raster* pada umumnya namun bukan berupa gambar yang di tampilkan melainkan berisi *geometry* metadata nama jalan, nama tempat, nomor rumah dan format ringkas serta terstruktur. (2) Spesifikasi *vector tile*, Spesifikasi *vector tile* sama dengan aplikasi yang membuat informasi apa yang ada dalam *database* dan alat mapbox *vector tile* menyimpan informasi geografis. Mapbox *vector tile specification* secara eksplisit menyediakan informasi tentang format dan ekstensi *file*, proyeksi dan batasan, dan struktur internal *vector tile*.

2.8 Mapbox Studio

Sebuah *tool* perancang dan membangun peta khusus dengan mapbox. aplikasi mapbox untuk mengelola data geospasial merancang peta dengan gaya khusus tersendiri. Mapbox Studio di gunakan membuat dan merancang peta sesuai dengan spesifikasi yang di inginkan untuk mengunggah dan mengedit data, dengan menggunakan *tile sets* yang disediakan mapbox, menambahkan *font* dan *icon* khusus, atau hanya memperbaiki *style* peta *template* bawaan.

2.9 Leaflet

LeafletJS singkatan Leaflet JavaScript merupakan perpustakaan javascript yang bersifat *open source*. LeafletJS pertama kali dirilis oleh Vladimir Agafonkin pada tahun 2011. *Library* ini khusus digunakan untuk membangun aplikasi pemetaan berbasis *web*, mendukung sebagian besar *mobile* dan *desktop platform*. Leaflet memungkinkan seorang tanpa latar belakang SIG mampu menampilkan peta

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

web tile pada *server* publik dengan mudah. Terdapat banyak *plugin* yang dapat digunakan untuk menambahkan fitur-fitur tambahan pada peta

2.9.1 Leaflet Draw API Reference

Leaflet *Draw* untuk menambahkan fitur *draw toolbar* mengatur sebuah *drawControl*, untuk pilihan benar pada pilihan peta. Leaflet *draw* biasanya membuat, mengedit, dan menghapus data *vector* baik berupa *point*, *polyline* dan *polygon*.

2.9.2 Leaflet Plugins

Leaflet *Plugins* sebuah pihak ketiga untuk mendukung fungsinya secara luas dengan dimaksud memperkecil sedemikian mungkin dan berfokus serangkaian fitur intinya.

2.10 Bahasa Pemrograman dan Database

Beberapa bahasa pemrograman dan *database* yang di gunakan dalam membangun *system* ini,

2.10.1 HTML dan CSS

Hypertext Markup Language (HTML) dan *Cascading Style Sheets* (CSS) adalah dua teknologi inti untuk membangun halaman *web*. HTML menyediakan struktur halaman, CSS (visual dan aural) *layout*, untuk berbagai perangkat. Seiring dengan grafis dan *scripting*, HTML dan CSS adalah dasar membangun halaman *web* dan aplikasi *web*.

2.10.2 PHP

PHP (*Hypertext Preprocessor*) adalah bahasa *scripting* tujuan umum *open source* yang umum digunakan yang sangat sesuai untuk pengembangan *web* dan dapat disematkan ke dalam HTML. Selain bahasa pemrograman tingkat tinggi php fungsi utama untuk pembuatan dan pengembangan *web* secara *full power*.

2.10.3 Javascript

Javascript adalah bahasa *scripting* kecil, ringan, berorientasi objek yang ditempelkan pada kode HTML dan di proses di sisi *client*. Javascript digunakan dalam pembuatan *website* agar lebih interaktif dengan memberikan kemampuan tambahan terhadap HTML melalui eksekusi perintah di sisi *browser*. Javascript dapat merespon perintah *user* dengan cepat dan menjadikan halaman *web* menjadi

responsive. Javascript memiliki struktur sederhana, kodenya dapat disisipkan pada dokumen HTML atau berdiri sebagai satu kesatuan aplikasi.

2.10.4 Bootstraap

Bootstrap adalah *toolkit open source* untuk pengembangan dengan HTML, CSS, dan JS. Dengan cepat prototipe yang di buat atau membuat keseluruhan aplikasi dengan variabel *Sass* dan *Mixin*, sistem grid responsif, komponen bebas membuat yang luas, dan plugin hebat yang terdapat di jQuery.

2.10.5 Bootleaf

Template sederhana dan responsif untuk membuat aplikasi pemetaan *web* dengan bootstrap, leaflet, dan *typeahead.js*. Fitur yang terdapat pada bootleaf ini, yaitu: (a) *Template* peta ramah peta lengkap dengan penanda *Navbar* dan modal yang *responsive*. (b) Memuat *jQuery file GeoJSON* eksternal. (c) Penanda *marker multiple layer* secara logis melalui *plugin client marker leaflet*. (d) Penelusuran fitur multi-lapisan sisi *client* yang elegan dengan pelengkapan otomatis menggunakan *typeahead.js*. (e) Daftar fitur *sidebar* yang responsif dengan penyortiran dan penyaringan via *list.js*. (f) *Icon* penanda disertakan dalam kontrol lapisan berkelompok melalui *plugin* pengontrol lapisan berkelompok.

2.10.6 Fontawesome

Fontawesome adalah *toolkit font* dan *icon* berdasarkan CSS dan LESS. Fontawesome dibuat oleh bernama Dave Gandy untuk digunakan dengan Twitter Bootstrap, dan kemudian dimasukkan ke dalam BootstrapCDN. Fontawesemo *framework* dengan banyak jenis *icon* di gunakan untuk mempercantik tampilan pada aplikasi *web*. Fontawesome menempatkannya di posisi kedua setelah Google Fonts. Beberapa fontawesome yang di gunakan, yaitu: (a) `<i class="fa fa-home"></i>`; (b) `<i class="far fa-id-card"></i>`; (c) `<i class="fa fa-globe white"></i>`; (d) `<i class="fa fa-list white"></i>`. Fontawesome *fa-home* di gunakan untuk menu *home*, *fa-id-card* digunakan untuk menu profil, *fa-globe* untuk menu peta dan *fa-list white* untuk menu legenda.

2.10.7 MySQL

Database dapat di artikan sebuah penyimpanan data berupa data dokumen file. *Database* didefinisikan kumpulan data yang dihubungkan secara bersama-

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

sama, dan gambaran dari data yang dirancang untuk memenuhi kebutuhan informasi dari suatu organisasi.

MySQL adalah *database open source* terpopuler di dunia. MySQL adalah properti *web* yang berkembang pesat, teknologi ISV atau perusahaan besar, MySQL dapat menghemat biaya untuk membantu Anda menjalankan aplikasi *database* berkinerja tinggi dan terukur.

2.10.8 Xampp

Xampp merupakan *tools* yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan menginstall XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi *web server* Apache, PHP dan MySQL secara manual.

2.11 CI Bonfire

CI Bonfire menyediakan area admin bertenaga bootstrap yang siap untuk dirubah dan di modifikasi kodenya. Mudah dalam menata ulang tema, dan mengintegrasikan area admin secara otomatis ke dalam modul aplikasi yang dibuat.

2.12 Visio

Visio aplikasi yang menghasilkan bagan alur, diagram jaringan, bagan organisasi, denah lantai, desain teknik, dan lainnya menggunakan bentuk dan *templete modern*.

2.13 Balsamiq Mockup

Balsamiq adalah alat *wireframing* yang cepat yang membantu membuat sketsa menggunakan komputer. Membuat *wireframes* lebih cepat dan menghasilkan banyak gagasan sehingga pengguna dapat mudah dan lebih cepat dalam mengganti atau memperbaiki desain dengan baik.

2.14 Metode V-Model

V-Model merupakan model menggambarkan bagaimana aktivitas pengujian (verifikasi dan validasi) dapat diintegrasikan ke dalam setiap fase siklus hidup. Dalam *V-Model*, pengujian validasi berlangsung terutama pada tahap awal (Graham et.al, 2006). Menurut Pressman dalam Faid (2017), *V-Model* merupakan model pengembangan dari metode *waterfall*, tahapan yang ada pada *V-Model* mirip dengan yang terdapat pada metode *waterfall*. Pada metode *waterfall* proses dijalankan secara berurutan yang dimulai dari puncak dan mengalir turun melalui

Hak Cipta Dilindungi Undang-Undang

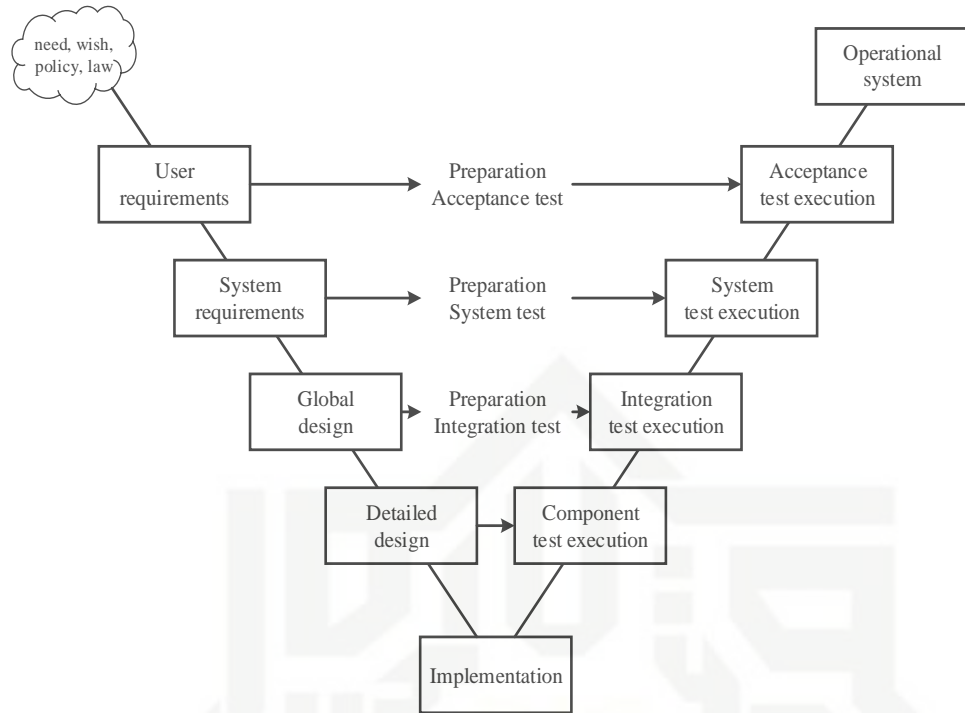
1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

berbagai tugas sampai implementasi sistem. Pada metode *waterfall* sulit untuk melakukan banyak pengulangan untuk fase tertentu. *V-Model* dikembangkan untuk mengatasi beberapa masalah pada metode *waterfall*. *V-Model* adalah model yang menggambarkan bagaimana aktifitas pengujian verifikasi dan validasi yang dapat diintegrasikan ketika membangun sebuah sistem informasi.

V-Model menggambarkan relasi aksi-aksi jaminan kualitas pada aksi-aksi yang berkaitan dengan komunikasi, pemodelan, serta aktivitas-aktivitas konstruksi pada tahap awal. Saat tim perangkat lunak bergerak ke bawah pada sisi kiri *V-Model*, spesifikasi-spesifikasi kebutuhan, permasalahan-permasalahan dasar dan solusi-solusinya diperhalus secara progresif menjadi semakin rinci dan semakin teknis. Saat kode program telah terbentuk, selanjutnya tim perangkat lunak bergerak ke atas melalui sisi kanan *V-Model* dan secara terus-menerus melaksanakan sejumlah pengujian yang berbeda (tindakan-tindakan jaminan kualitas), yang pada dasarnya melakukan validasi pada masing-masing tahapan model yang telah diselesaikan saat tim perangkat lunak bergerak ke bawah pada sisi kiri model, (Pressman, 2010).

Tahapan pada *V-Model* dibagi menjadi dua tahap yaitu tahap pengembangan sistem dan tahap pengujian sistem. Tahap pengembangan sistem terdapat lima tahapan yaitu *user requirements*, *system requirements*, *global design*, *detailed design* dan *implementation*. Sedangkan tahap pengujian sistem terdiri dari empat tahapan yaitu *component testing*, *integration testing*, *system testing* dan *acceptance testing*.

Dalam *V-Model* ini digambarkan hubungan antara tahap pengembangan *software* dengan tahap pengujiannya yang dapat dilihat pada Gambar 2.2.

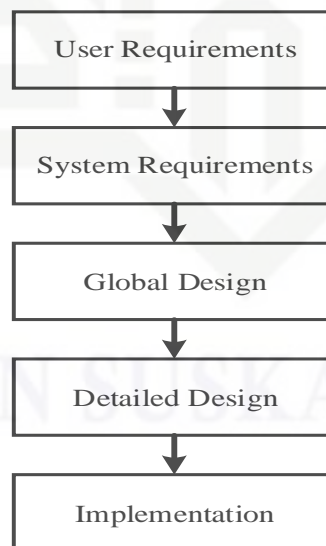


Gambar 2.2 V-Model

(Sumber: Graham et.al, 2006)

2.14.1 Tahap Pengembangan Sistem

Pada tahap pengembangan sistem ini terdapat lima tahapan yang dapat dilihat pada Gambar 2.3.



Gambar 2.3. Tahap Pengembangan Sistem

(Sumber: Graham et.al, 2006)

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

1. *User Requirements*

Tahap *user requirements* dikenal juga sebagai tahap mendefinisikan kebutuhan pengguna. Keluaran dari tahap ini adalah dokumentasi kebutuhan pengguna. Tahap ini juga menggambarkan kebutuhan pengguna terhadap sistem yang akan dikembangkan.

2. *System Requirements*

Tahap *system requirements* yaitu tahap mendefinisikan kebutuhan sistem. Setelah mengetahui permasalahan pada sistem yang berjalan, selanjutnya mencari solusi pada masalah tersebut.

3. *Global Design*

Pada tahap ini dilakukan perancangan (*design*) yang dimaksudkan untuk membuat pemodelan sistem baru yang dapat mewakili sistem yang berjalan saat ini. Tujuan dari tahap adalah untuk menentukan bagaimana komponen *software* dari sistem informasi akan dijalankan pada *hardware* yang ada di sistem. Dasar dari pemilihan arsitektur yang akan digunakan berdasar kepada beberapa hal seperti pemakaian kembali tiap modul, ketergantungan tabel dalam basis data, hubungan antar *interface*, *detail* teknologi yang dipakai.

4. *Detailed Design*

Pada tahap ini perancangan yang telah dilakukan sebelumnya dipecah menjadi modul-modul yang lebih kecil. Setiap modul akan diberi penjelasan yang cukup untuk memudahkan programmer melakukan *coding*.

5. *Implementation*

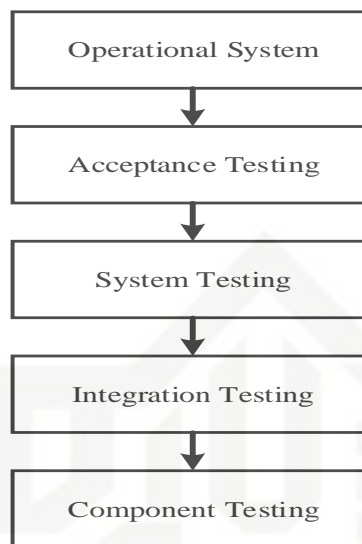
Pada tahap ini dilakukan implementasi terhadap keseluruhan aplikasi setelah tahap pengkodean selesai.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

2.14.2 Tahap Pengujian Sistem

Terdapat lima fase pengujian yang dapat dilihat pada Gambar 2.4.



Gambar 2.4. Tahap Pengujian Sistem
(Sumber: Graham et.al, 2006)

1. *Component Testing*

Component testing juga dikenal sebagai unit, pengujian program yang mencari kesalahan serta memverifikasi fungsi perangkat lunak. Biasanya *component testing* terjadi dengan akses kode yang sedang diuji dan dengan dukungan pengembangan, seperti alat uji kerangka atau debug. Setelah ditemukan kesalahan, biasanya dilakukan perbaikan pada kodingan tersebut (Graham et.al, 2006).

2. *Integration Testing*

Integration testing yaitu menguji antarmuka antar komponen. Pengujian integrasi harus dibedakan dari kegiatan integrasi lainnya. Idealnya pengujian harus memahami arsitektur dan mempengaruhi perencanaan integrasi. Jika *integration testing* direncanakan sebelum komponen atau sistem dibangun, mereka dapat dikembangkan dalam urutan yang diperlukan untuk pengujian yang paling efisien, (Graham et.al, 2006). *Integration testing* dapat dilakukan dengan pengujian *blackbox* seperti *spesification based testing*.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:
 - a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
 - b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.
2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

3. *System Testing*

Pada tahap ini akan dilakukan pengujian sistem secara keseluruhan untuk memastikan bahwa sistem telah memenuhi persyaratan. Pengujian sistem sebenarnya adalah serangkaian pengujian yang berbeda-beda yang tujuan utamanya adalah untuk sepenuhnya mewujudkan sistem berbasis komputer, (Pressman, 2015). Meskipun masing-masing pengujian memiliki tujuan yang berbeda, semua pengujian yang dilakukan untuk memverifikasi bahwa semua elemen sistem telah terintegrasi dengan baik dan menjalankan fungsi yang telah ditetapkan (Pressman, 2015).

Pengujian sistem berkaitan dengan kegiatan seluruh sistem sebagaimana ditentukan oleh ruang lingkup proyek atau produk yang dikembangkan, (Graham et.al, 2006).

System testing ini dilakukan terhadap aplikasi yang telah selesai dikembangkan. Pengujian ini dapat dilakukan menggunakan metode *blackbox*.

4. *Acceptance Testing*

Acceptance testing atau uji penerimaan merupakan pengujian yang dilakukan untuk menentukan apakah sistem sudah diterima oleh pengguna dan sudah memenuhi kebutuhan yang diminta oleh pengguna. Pada tahap ini, *end-user* yang terpilih melakukan *testing* terhadap fungsi-fungsi aplikasi dan melaporkan permasalahan yang ditemukan. Hasil *output* pengujian ini adalah sebuah dokumen hasil uji yang dapat dijadikan bukti bahwa *software* sudah diterima dan sudah memenuhi kebutuhan yang diminta.

Ketika pengembang telah melakukan pengujian sistemnya dan telah memperbaiki semua atau sebagian besar kesalahan, sistem akan dikirimkan kepada pengguna untuk pengujian penerimaan. *Acceptance testing* paling sering difokuskan pada jenis pengujian validasi, dimana peneliti mencoba menentukan apakah sistem sesuai dengan tujuan yang diharapkan (Graham et.al, 2006). Tujuan dari pengujian ini adalah untuk mengetahui tingkat kelayakan dari perangkat lunak. *Acceptance test* dapat dilakukan dengan menggunakan metode *blackbox*.

5. *Operational System*
Operational system adalah pemeliharaan sistem setelah digunakan selama beberapa tahun. Pengembangan dan proses uji yang dilakukan sama dengan sebelumnya, yaitu dilakukan pengujian ulang *component testing*, *integration testing*, *system testing*, dan *acceptance testing*. Proses pemeliharaan biasanya dilakukan untuk memperbaharui sistem yang sebelumnya (Graham et.al, 2006).




2.15 Unified Modeling Language (UML)

Tools yang bisa digunakan untuk menganalisis dan mendesain sistem menggunakan OOAD adalah UML. UML menyediakan beberapa diagram visual yang menunjukkan berbagai aspek dalam sistem, (Sholih, 2006).

2.15.1 Diagram Use case (Use Case Diagram)

Menyajikan interaksi antara *use case* dan aktor. *Actor* dapat berupa orang, peralatan, atau sistem lain yang berinteraksi dengan sistem yang sedang dibangun. *Use case* menggambarkan fungsionalitas sistem atau persyaratan-persyaratan yang harus dipenuhi sistem dari pandangan pemakai atau pengguna. Pelajari setiap penjelasan simbol *use case diagram* yang dijelaskan dalam Tabel 2.1. berikut ini.

Tabel 2.1. Deskripsi *Use Case Diagram*.

SIMBOL	NAMA	KETERANGAN
	<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
	<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
	<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .








(Sumber: Sholih, 2006).

2.15.2 Diagram Aktivitas (Activity Diagram)

Menggambarkan aliran fungsionalitas sistem. Digunakan untuk menggambarkan aliran kejadian (*flow of events*) dalam *use case*. Aktivitas dalam digram dipresentasikan dengan bentuk bujur sangkar bersudut tidak lancip, yang didalamnya berisi langkah-langkah apa saja yang terjadi dalam aliran kerja. Ada

sebuah keadaan mulai (*start state*) yang menunjukkan dimulainya aliran kerja, dan sebuah keadaan selesai (*end state*) yang menunjukkan akhir diagram, titik keputusan dipresentasikan dengan *diamond*. Diagram aktivitas tidak perlu dibuat untuk setiap aliran kerja, tetapi diagram ini akan sangat berguna untuk aliran kerja yang kompleks dan melebar, (Sholiq, 2006). Pelajari setiap penjelasan simbol *activity diagram* yang dijelaskan dalam Tabel 2.2. berikut ini.

Tabel 2.2 Deskripsi *Activity Diagram*

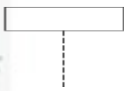

GAMBAR	NAMA	KETERANGAN
	<i>Actifity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
	<i>Decision</i>	State dari sistem yang mencerminkan pengambilan keputusan
	<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
	<i>Actifity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
	<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran
	<i>Control Flow</i>	Arus aktivitas
	<i>Receive</i>	Tanda Penerimaan

(Sumber: Sholiq, 2006)

2.15.3 Diagram Sekuensial (*Sequence Diagram*)



Digunakan untuk menunjukkan aliran fungsionalitas dalam *use case*, (Sholiq, 2006). Pelajari setiap penjelasan simbol *sequence diagram diagram* yang dijelaskan dalam Tabel 2.3 berikut ini.

Tabel 2.3 Deskripsi *Sequence Diagram*

GAMBAR	NAMA	KETERANGAN
	<i>Lifeline</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
	<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

(Sumber: Sholiq, 2006)

Tabel 2.3 Deskripsi *Sequence Diagram* (Lanjutan)

GAMBAR	NAMA	KETERANGAN
	<i>Self-message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
	<i>Activation</i>	Indikasi dari sebuah objek yang melakukan suatu aksi


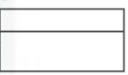


(Sumber: Sholiq, 2006)

2.15.4 Diagram Kelas (*Class Diagram*)

Menunjukkan interaksi antara kelas dalam sistem. Kelas mengandung informasi dan tingkah laku (*behavior*) yang berkaitan dengan informasi tersebut. Sebuah kelas pada diagram kelas dibuat untuk setiap tipe objek pada diagram sekuensial atau diagram kolaborasi.

Programmer menggunakan diagram ini untuk mengembangkan kelas. *Case tool* tertentu seperti *rational rose* membangkitkan struktur kode sumber untuk kelas-kelas, kemudian para *programmer* menyempurnakan dengan bahasa pemrograman yang dipilih pada saat *coding*. *Analyst* menggunakan *diagram* ini untuk menunjukkan detail sistem, sedangkan arsitek sistem mempergunakan diagram ini untuk melihat rancangan sistem (Sholiq, 2006). Pelajari setiap penjelasan simbol *class diagram diagram* yang dijelaskan dalam Tabel 2.4 berikut.

Tabel 2.4. Deskripsi *Class Diagram*

GAMBAR	NAMA	KETERANGAN
	<i>Generalization</i>	Hubungan objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atas objek induk (<i>ancestor</i>)
	<i>Class</i>	Himpunan dari objek yang berbagi atribut serta operasi yang sama
	<i>Dependency</i>	Hubungan perubahan yang terjadi pada elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung pada elemen yang tidak mandiri
	<i>Association</i>	Menghubungkan antara objek satu dengan objek lainnya

(Sumber: Sholiq, 2006)

2.16 Pengujian Sistem

Pengujian aplikasi sangat diperlukan untuk memastikan aplikasi yang sudah atau sedang dibuat dapat berjalan sesuai dengan fungsionalitas yang diharapkan. Pengembang atau penguji aplikasi harus menyiapkan sesi khusus untuk menguji program yang sudah dibuat agar kesalahan ataupun kekurangan dapat dideteksi sejak awal dan dikoreksi secepatnya. Pengujian atau testing sendiri merupakan elemen kritis dari jaminan kualitas perangkat lunak dan merupakan bagian yang tidak terpisah dari siklus hidup pengembangan *software* seperti halnya analisis, desain, dan pengkodean (Shidiq dkk, 2010).

Pengujian software haruslah dilakukan dalam proses rekayasa perangkat lunak atau *software engineering*. Sejumlah strategi pengujian *software* telah diusulkan dalam literatur. Langkah-langkah pengujian *software* ada 4, yaitu:

1. *Unit testing-testing* per unit yaitu mencoba alur yang spesifik pada struktur modul kontrol untuk memastikan pelengkapan secara penuh dan pendeteksian *error* secara maksimum.
2. *Integration testing – testing* per penggabungan *unit* yaitu pengalamanan dari isu-isu yang diasosiasikan dengan masalah ganda pada verifikasi dan konstruksi program.
3. *High-order test* yaitu terjadi ketika *software* telah selesai diintegrasikan atau dibangun menjadi satu tidak terpisah-pisah.
4. *Validation test* yaitu menyediakan jaminan akhir bahwa *software* memenuhi semua kebutuhan fungsional, kepribadian dan performa.

Menurut Lewis, *software testing* adalah aktivitas menjalankan serangkaian eksekusi yang dinamis pada program *software* setelah *source code software* tersebut telah dikembangkan. *Software testing* dilakukan untuk menemukan dan memperbaiki sebanyak mungkin potensi kesalahan sebelum *software* tersebut digunakan oleh pelanggan atau *end user*.

2.17 Blackbox Testing

Merupakan metode pengujian perangkat lunak yang dilakukan hanya mengamati hasil eksekusi data uji dan memeriksa fungsional dari perangkat lunak. Cakupan pengujian yang dilakukan pada *blackbox testing* adalah perihal pengujian

Tabel 2.5. Penelitian Terdahulu

No.	Peneliti / Tahun	Hasil / Keterangan
1	Avhad Priyanka T (2016) <i>Mapping and Management of Water Pipeline in Aurangabad City.</i>	Teknologi geospasial seperti penginderaan jarak jauh, pemetaan mobile sangat berguna dalam pembuatan <i>database</i> utilitas air geospasial dan SIG membantu berbagai tahap pengelolaan jaringan utilitas air terutama dalam pengelolaan aset, mengidentifikasi risiko aset dan sub-pemodelan 3D permukaan dari jaringan utilitas.
2	Mar' Atul Karimah (2012). Sistem Informasi Geografis Pemetaan Sumber Air Dan Reservoir Kabupaten Sleman.	Pengembangan system ini adalah tata letak sarna air bersih berupa sumber air, reservoir dan dalam bentuk peta kertas.
3	Arief Laila Nugraha, Hani'ah (2011). Desain Aplikasi SIG Untuk Pelayanan Jaringan Pipa Pdam.	Hasil penelitian ini adalah SIG Jaringan Pipa PDAM Kota Demak. Pemetaan jaringan pipa air PDAM Demak menggunakan <i>Pmapper</i> .
4	Singh dan Gambhir (2014). <i>An Open Source Approach to Build a Web GIS Application.</i>	Dalam aplikasi ini <i>pmapper</i> telah digunakan untuk membuat GUI di dalamnya gambar peta yang dibuat oleh <i>MapServer</i> ditampilkan setiap pan, <i>zoom</i> , <i>query</i> dll <i>events</i> . <i>Shapefiles</i> telah digunakan dalam aplikasi ini tapi database PostGIS lapisan juga bisa digunakan karena <i>software open source, framework</i> inti <i>pmapper</i> dapat digunakan oleh pengembang perangkat lunak untuk membangun <i>web</i> yang kompleks Aplikasi SIG
5	Awad, Ganawa, Salih, Hilmi dan Mustafa (2016). <i>Pipeline Operations & Integrity Management Using GIS & Remote Sensing Technologies (Applying on PDOC'S Export pipeline).</i>	Dalam penelitian ini SIG digunakan untuk menyediakan ruang kompak dimana segala macam data yang relevan dengan Area dapat disimpan format digital, termasuk gambar, peta, dokumen, foto. Data sudah diatur sedemikian rupa sehingga dapat dimasukkan ke dalam display seperti peta, grafik, dan meja, dan bisa dalam pelayanan analitis yang canggih Prosedur.
6	Ardiansyah dan Kardono (2017). Sistem informasi geografis (SIG) pemetaan jaringan Pipa dan titik properti pelanggan di PT.Aetra air Tangerang.	Menampilkan peta secara <i>visual</i> dalam cakupan skala wilayah yang cukup luas jaringan pipa dan titik properti. Peta <i>digital</i> yang ada dalam sistem dapat menyimpan informasi yang ada, meliputi: informasi pemetaan Jaringan pipa, titik properti pelanggan, batas-batas wilayah konsesi dan jumlah perumahan dan jumlah pelanggan.

(Sumber: Data Olahan, 2018)

Tabel 2.5. Penelitian Terdahulu (Lanjutan)

No.	Peneliti / Tahun	Hasil / Keterangan
7	Susila, Piarsa dan Buana (2014). Sistem Informasi Geografis Pemetaan Jaringan Pipa PDAM Tirta Mangutama	Sistem Informasi Geografis Pemetaan Jaringan Pipa PDAM Tirta Mangutama Berbasis Web. Sistem dapat menampilkan jaringan pipa primer, jaringan pipa sekunder, data objek sumber air, reservoar dan aksesoris dengan menggunakan fitur <i>marker</i> dan <i>polyline</i> di wilayah distribusi Badung Utara. Sistem dapat memberikan laporan/ <i>report</i> jaringan pipa mengenai jaringan pipa yang saling terkoneksi dengan sumber air, reservoar, aksesoris dan jaringan pipa lain dengan format HTML dan pdf.
8	Shunfu dan Dai (2013). Online Map Application Development Using Google Maps API, SQL Database, and ASP.NET	Aplikasi pemetaan online itu berhasil dikembangkan menggunakan Google Maps API v3, Google Geocoding, Microsoft SQL Server Express database, dan Spry Framework untuk Ajax.

(Sumber: Data Olahan, 2018)

2.20 Profil Instansi

Profil instansi yaitu sebuah tempat penelitian yang di laukan oleh penulis untuk melakukan penelitian dan pengolahan data yang di perlukan oleh peneliti.

2.20.1 Sejarah Dinas PUPR Provinsi Riau

Departemen Pekerjaan Umum adalah lembaga pemerintah yang bertanggung jawab langsung kepada presiden. Sebelumnya, Departemen Pekerjaan Umum sempat bernama Departemen Pemukiman dan Pengembangan Wilayah (2000-2004), adalah kementerian dalam pemerintah Indonesia yang membidangi urusan pekerjaan umum. Istilah “Pekerjaan Umum” adalah terjemahan dari istilah bahasa Belanda *Openbare Werken* yang pada zaman Hindia belanda disebut *Waterstaat Swerken*. Setelah belanda menyerah dalam perang pasifik pada tahun 1942 kepada jepang, maka daerah Indonesia ini dibagi oleh Jepang dalam 3 wilayah pemerintah, yaitu Jawa, Madura, Sumatra dan Indonesia Timur dan tidak ada pusat Pemerintah tertinggi di Indonesia yang menguasai ke-3 wilayah tersebut.

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.

b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

Dinas Pekerjaan Umum Provinsi Riau terbentuk setelah ditetapkan undang-undang darurat No 19/1957 kemudian diundangkan dengan Undang-Undang No.61 tahun 1958.

Pekerjaan umum adalah unsur pelaksana dan penunjang dibidang Pekerjaan Umum yang mempunyai tugas melaksanakan sebagai tugas pemerintah dan pembangunan dibidang Pekerjaan Umum, Permukiman dan Prasarana Wilayah diserahkan (desentralisasi) dan dilimpahkan (dekonsentrasi) kepada Daerah Provinsi Riau, sesuai ketentuan perundang undangan yang berlaku.

2.20.2 Visi dan Misi

1. Visi

"Terwujudnya Provinsi Riau yang maju, masyarakat sejahtera, berbudaya Melayu dan berdaya saing tinggi, menurunnya kemiskinan, tersedianya lapangan kerja serta pemantapan aparatur".

2. Misi

- a. Meningkatkan pembangunan insfrastruktur.
- b. Meningkatkan pelayanan pendidikan.
- c. Meningkatkan pelayanan kesehatan.
- d. Mewujudkan pemerintahan yang handal dan terpercaya serta pemantapan kehidupan politik.
- e. Pembangunan masyarakat yang berbudaya melayu, beriman dan bertaqwa.
- f. Memperkuat pembangunan pertanian dan perkebunan.
- g. Meningkatkan perlindungan dan pengelolaan lingkungan hidup serta pariwisata.
- h. Meningkatkan peran swasta dalam pembangunan.

2.20.3 Tugas Pokok dan Fungsi

1. Tugas Pokok

"Melaksanakan urusan pemerintahan daerah di Bidang KePUPRan berdasarkan asas otonomi daerah dan tugas Pembantuan".

Hak Cipta Dilindungi Undang-Undang

1. Dilarang mengutip sebagian atau seluruh karya tulis ini tanpa mencantumkan dan menyebutkan sumber:

- a. Pengutipan hanya untuk kepentingan pendidikan, penelitian, penulisan karya ilmiah, penyusunan laporan, penulisan kritik atau tinjauan suatu masalah.
- b. Pengutipan tidak merugikan kepentingan yang wajar UIN Suska Riau.

2. Dilarang mengumumkan dan memperbanyak sebagian atau seluruh karya tulis ini dalam bentuk apapun tanpa izin UIN Suska Riau.

2. Fungsi

- a. Perumusan kebijaksanaan teknis bidang jalan dan jembatan.
- b. Perancangan program kerja Kebinamargaan.
- c. Penyelenggaraan urusan pemerintahan dan pelayanan umum bidang jalan dan jembatan.
- d. Pembinaan dan fasilitasi bidang jalan dan jembatan lingkup provinsi kabupaten/kota.
- e. Pelaksanaan tugas bidang bina teknik, pembangunan dan peningkatan jalan dan jembatan serta pemeliharaan jalan dan jembatan.
- f. Pelaksanaan Tugas Pembuatan di Bidang jalan dan jembatan.
- g. Pembuatan, evaluasi dan pelaporan bidang jalan dan jembatan.
- h. Pelaksanaan kesekretariatan Dinas.
- i. Penyelenggaraan pelaksanaan penanganan darurat akibat bencana dalam bidang ke Binamargaan.
- j. Melaksanakan tugas lain yang diberikan oleh Gubernur sesuai dengan tugas dan fungsinya.

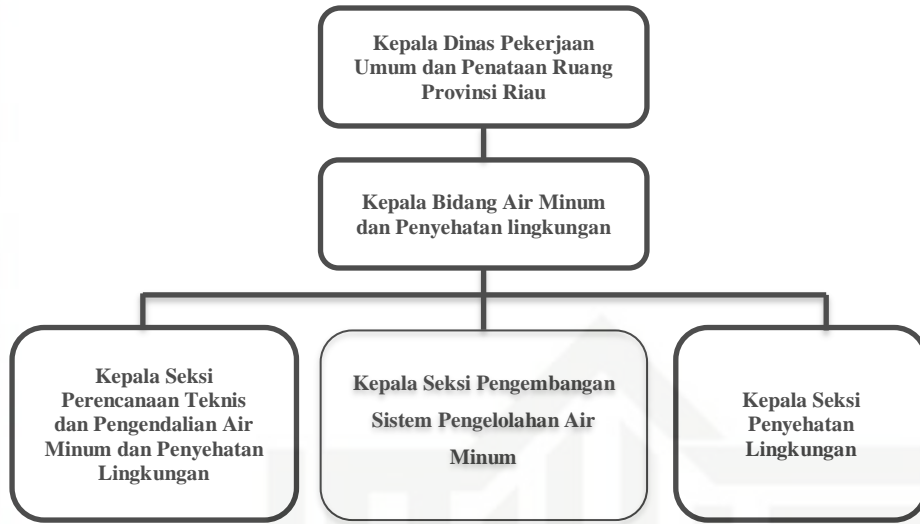
2.20.4 Struktur Organisasi

Struktur organisasi Dinas Pekerjaan Umum dan Penataan Ruang (PUPR) Provinsi Riau dapat dilihat pada Lampiran B.

2.20.5 Bidang Air Minum dan Penyehatan Lingkungan

1. Nama jabatan : Kepala Seksi Pengembangan Sistem Pengelolaan Air Minum.
2. Kode Jabatan : -
3. Unit Kerja
Eselon II : Dinas Pekerjaan Umum dan Penataan Ruang
Eselon III : Bidang Air Minum dan Penyehatan Lingkungan
Eselon IV : Seksi Pengembangan Sistem Pengelolaan Air Minum

4. Kedudukan Dalam Struktur Organisasi :



Gambar 2.5 Struktur Organisasi Bidang Air Minum dan Penyehatan Lingkungan

(Sumber: Data Primer, 2018)

5. Ikhtisar Jabatan :

Memimpin dan melaksanakan Seksi Pengembangan Sistem Pengolahan Air Minum dalam menyelenggarakan kegiatan Pengembangan Sistem Pengolahan Air Minum serta tugas lainnya yang diberikan oleh atasan sesuai dengan ketentuan yang berlaku agar dapat tercapai pelaksanaan tugas yang berdaya guna dan berhasil guna.

6. Uraian Tugas:

Merencanakan dan melaksanakan Seksi Pengembangan Sistem Pengolahan Air Minum berdasarkan rencana kegiatan Seksi Pengembangan Sistem Pengolahan Air Minum dan ketentuan berlaku sebagai pedoman pelaksanaan tugas.

- a. Mempelajari rencana kegiatan Seksi Pengembangan Sistem Pengolahan Air Minum tahun lalu.
- b. Menyusun konsep rencana kegiatan Seksi Pengembangan Pengolahan Air Minum.
- c. Mengunsultasikan konsep rencana kegiatan dengan pimpinan untuk mendapatkan pengarahan.