**BAB II**

**LANDASAN TEORI**

1. **Dinas Pekerjaan Umum Bina Marga Kabupaten Mandailing Natal**

Dinas Pekerjaan Umum Bina Marga Kabupaten Mandailing Natal merupakan dinas pemerintahan sebagai unsur pelaksana dan penunjang mempunyai tugas melaksanakan perumusan kebijakan dan program serta pengaturan, pembinaan, pembangunan, pengusaha dan pengawasan bidang bina marga.

Akuntabilitas kinerja Dinas Pekerjaan Umum Bina Marga Kabupaten Mandailing Natal merupakan perwujudan kewajiban organisasi untuk mempertangggung jawabkan keberhasilan dan kegagalan pelaksanaan visi dan misi Dinas Pekerjaan Umum Bina Marga Kabupaten Mandailing Natal dalam mencapai sasaran dan tujuan yang telah ditetapkan secara periodik.

* + 1. **Fungsi dan Tugas**

Dinas Pekerjaan Umum Bina Marga Kabupaten Mandailing Natal memiliki fungsi sebagai berikut:

1. Koordinasi pengumpulan bahan dan data perumusan kebijakan dan program dibidang bina marga.
2. Penyusuan pedoman operasional penyelenggaraan dan penetapan status jalan kabupaten, jalan desa dan jalan perkotaan.
3. Penyusunan perencanaan umum dan pembiayaan jaringan jalan dan jembatan.
4. Pengembangan teknologi terapan dibidang jalan dan jembatan.

Dalam menyelenggarakan tugas dan fungsi sebagaimana yang dimaksudkan dalam pasal 14, Bidang Bina Marga terdiri atas:

1. Seksi Pembangunan Jalan dan Jembatan

Berikut ini adalah penjelasan tugas pokok dan fungsi dari seksi pembangunan jalan dan jembatan yang dapat dilihat pada Tabel 2.1:

Tabel 2.1 Tugas Pokok dan Fungsi Seksi Pembangunan Jalan dan Jembatan

(Sumber : Data Primer, 2019)

|  |  |
| --- | --- |
| **Seksi Pembangunan Jalan dan Jembatan** | |
| **Tugas Pokok** | Seksi pembangunan jalan dan jembatan mempunyai tugas melaksanakan kegiatan pembangunan jalan/ jembatan berdasarkan ketentuan peraturan yang berlaku. |
| **Fungsi** | 1. Penyiapan bahan dan data penyusunan kebijakan dan program dibidang pembangunan jalan dan jembatan. 2. Mempersiapkan rekomendasi (*advice service*) dibidang pembangunan jalan dan jembatan. 3. Mengkoordinasikan spesifikasi teknis dan rencana anggaran biaya pembangunan jalan dan jembatan. 4. Melaksanakan pembinaan kegiatan dibidang pembangunan jalan dan jembatan. |

1. Seksi Peningkatan Jalan dan Jembatan

Berikut ini adalah penjelasan tugas pokok dan fungsi dari seksi peningkatan jalan dan jembatan yang dapat dilihat pada Tabel 2.2:

Tabel 2.2 Tugas Pokok dan Fungsi Seksi Peningkatan Jalan dan Jembatan

|  |  |
| --- | --- |
| **Seksi Peningkatan Jalan dan Jembatan** | |
| **Tugas Pokok** | Seksi peningkatan jalan dan jembatan mempunyai tugas melaksanakan penyusunan dan pelaksanaan kebijakan dan program dibidang peningkatan jalan dan jembatan |
| **Fungsi** | * + - 1. Penyiapan bahan dan penyusunan kebijakan dan program dibidang peningkatan jalan dan jembatan.       2. Mempersiapkan rekomendasi (advis admin) dibidang peningkatan jalan dan jembatan.       3. Mengkoordinasikan spesifikasi teknis dan rencana anggaran biaya peningkatan jalan dan jembatan. |

(Sumber : Data Primer, 2019)

1. Seksi Rehabilitasi dan Pemeliharaan Jalan dan Jembatan

Berikut ini adalah penjelasan tugas pokok dan fungsi dari seksi rehabilitasi dan pemeliharaan jalan dan jembatan yang dapat dilihat pada Tabel 2.3:

Tabel 2.3 Tugas Pokok dan Fungsi Seksi Rehabilitasi dan Pemeliharaan Jalan dan Jembatan

|  |  |
| --- | --- |
| **Seksi Rehabilitasi dan Pemeliharaan Jalan dan Jembatan** | |
| **Tugas Pokok** | Seksi Rehabilitasi dan Pemeliharaan Jalan dan Jembatan mempunyai tugas melaksanakan penyusunan dan pelaksanaan kebijakan dan program dibidang rehabilitasi dan pemeliharaan jalan dan jembatan. |
| **Fungsi** | 1. Penyiapan bahan dan data penyusunan kebijakan dan program dibidang rehabilitasi dan pemeliharaan jalan dan jembatan. 2. Mempersiapkan rekomendasi (advis teknis) dibidang rehabilitasi dan pemeliharaan jalan dan jembatan.Mengkoordinasikan spesifikasi teknis dan rencana anggaran biaya rehabilitasi dan pemeliharaan jalan/ jembatan. 3. Melaksanakan pembinaan kegiatan dibidang rehabilitasi dan pemeliharaan jalan dan jembatan. 4. Pelaksanaan kegiatan rehabilitasi dan pemeliharaan jalan dan jembatan berdasarkan spesifikasi teknis yang telah ditetapkan. |

(Sumber : Data Primer, 2019)

* + 1. **Visi dan Misi**

Berikut visi dan misi Dinas Pekerjaan Umum Bina Marga Kabupaten Mandailing Natal:

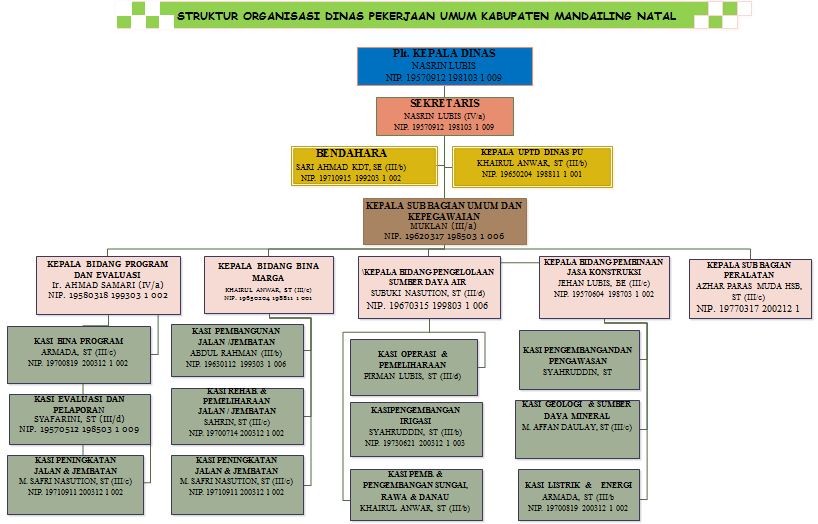
Tabel 2.4 Visi dan Misi Dinas Pekerjaan Umum Kabupaten Mandailing Natal

|  |  |
| --- | --- |
| **Visi** | “Tersedianya infastruktur jalan dan jembatan dalam mewujudkan masyarakat sejahtera dan berdaya saing tinggi.” |
| **Misi** | 1. Memperkuat kapasitas kelembagaan Dinas Bina Marga dan membangun kerja sama antar pemangku kepentingan 2. Melaksanakan pembangunan atau peningkatan jalan dan jembatan untuk memacu laju pertumbuhan ekonomi dan membuka daerah terisolir sesuai dengan pemanfaatan ruang pada Kabupaten Mandailing Natal 3. Memelihara jalan dan jembatan untuk memperlancar arus barang dan jasa 4. Meningkatkan kinerja sarana dan prasarana pendukung kebinamargaan. |

(Sumber: Data Primer, 2019)

* + 1. **Struktur Organisasi**

Struktur organisasi merupakan susunan dan hubungan antara tiap bagian baik secara posisi maupun tugas yanga ada pada perusahaan dalam kegiatan operasional untuk mencapai tujuan perusahaan. Gambar dibawah ini merupakan struktur organisasi yang ada di Dinas Pekerjaan Umum Bina Marga Kabupaten Mandailing Natal:



Gambar 2.1 Struktur Organisasi Dinas Pekerjaan Umum Kab. Mandailing Natal

(Sumber: Data Primer, 2017)

## 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).

* 1. **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 efisiensibiaya, efisiensi beban kerja sumber daya manusia untuk instalasi,pemeliharaan dan dukungan teknis, pemangkasan kurvapembelajaran untuk pengguna akhir dan keunggulan dalam halintegrasi data spasial dan data non spansial menggunakan DBMS, (Wijaya 2012).

* 1. **Jalan**
     1. **Definisi Jalan**

Berdasarkan Undang-undang nomor 38 tahun 2004 tentang jalan, jalan adalah prasarana transportasi darat yang meliputi segala bagian jalan, termasuk bangunan pelengkap dan perlengkapannya yang diperuntukkan bagi lalu lintas, yang berada pada permukaan tanah, di atas permukaan tanah, di bawah permukaan tanah atau air, serta di atas permukaan air, kecuali jalan kereta api, jalan lori, dan jalan kabel.

* + 1. **Klasifikasi Jalan**

Berdasarkan undang-undang nomor 38 tahun 2004 tentang jalan, klasifikasi jalan berdasarkan fungsinya dibedakan atas :

* 1. Jalan arteri, merupakan jalan umum yang berfungsi melayani angkutan utama dengan ciri perjalanan jarak jauh, kecepatan rata-rata tinggi, dan jumlah jalan masuk (akses) dibatasi secara berdaya guna.
  2. Jalan kolektor, merupakan jalan umum yang berfungsi melayani angkutan pengumpul atau pembagi dengan ciri perjalanan jarak sedang, kecepatan rata-rata sedang, dan jumlah jalan masuk dibatasi.
  3. Jalan lokal, merupakan jalan umum yang berfungsi melayani angkutan setempat dengan ciri perjalanan jarak dekat, kecepatan rata-rata rendah, dan jumlah jalan masuk tidak dibatasi.
  4. Jalan lingkungan, merupakan jalan umum yang berfungsi melayani angkutan lingkungan dengan ciri perjalanan jarak dekat, dan kecepatan rata-rata rendah.

Berdasarkan Undang-undang nomor 38 tahun 2004 tentang jalan, klasifikasi jalan berdasarkan status jalan dibagi menurut kewenangan pembinaannya,yaitu :

* + 1. Jalan nasional, merupakan jalan arteri dan jalan kolektor dalam sistem jaringan jalan primer yang menghubungkan antar ibukota provinsi, dan jalan strategis nasional, serta jalan tol.
    2. Jalan provinsi, merupakan jalan kolektor dalam sistem jaringan jalan primer yang menghubungkan ibukota provinsi dengan ibukota kabupaten/kota, atau antar ibukota kabupaten/kota, dan jalan strategis provinsi.
    3. Jalan kabupaten, merupakan jalan lokal dalam sistem jaringan jalan primer yang tidak termasuk jalan yang menghubungkan ibukota kabupaten dengan ibukota kecamatan, antar ibukota kecamatan, ibukota kabupaten dengan pusat kegiatan lokal, antarpusat kegiatan lokal, serta jalan umum dalam sistem jaringan jalan sekunder dalam wilayah kabupaten, dan jalan strategis kabupaten.
    4. Jalan kota, merupakan jalan umum dalam sistem jaringan jalan sekunder yang menghubungkan antar pusat pelayanan dalam kota, menghubungkan pusat pelayanan dengan persil, menghubungkan antarpersil, serta menghubungkan antar pusat permukiman yang berada di dalam kota.
    5. Jalan desa, merupakan jalan umum yang menghubungkan kawasan dan antar permukiman di dalam desa, serta jalan lingkungan.

## *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, memanajemen, mengedit, menganalisis data, dan menyusun peta yang dapat dicetak, (http://qgis.org).

## LeafletJS

LaefletJS 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*, memdukung sebagian besar *mobile* dan *desktop* *platform*. Leaflet memungkinkan seorang tanpa latar belakang SIG mampu menampikan peta *web* *tile* pada *server* publik dengan mudah. Terdapat banyak *plugin* yang dapat digunakan untuk menambahkan fitur-fitur tambahan pada peta.

* 1. **L****eaflet *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 menghapusdata *vector* baikberupa *point*, *polyline* dan *polygon*.

* 1. **Bahasa Pemograman dan Database**
     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***.**

* + 1. **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 pemoraman tingkat tinggi php fungsi utama untuk pembuatan dan pengembangan web secara full power.

* + 1. **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.

* + 1. **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.

* + 1. **Bootleaf**

Template sederhana dan responsif untuk membuat aplikasi pemetaan web dengan [bootstrap](https://translate.googleusercontent.com/translate_c?depth=1&hl=id&prev=search&rurl=translate.google.com.sg&sl=en&sp=nmt4&u=http://getbootstrap.com/&usg=ALkJrhjap2qNz5EQpqJc4n_hUocqXvWb3Q), [leaflet](https://translate.googleusercontent.com/translate_c?depth=1&hl=id&prev=search&rurl=translate.google.com.sg&sl=en&sp=nmt4&u=http://leafletjs.com/&usg=ALkJrhjqSrBcQLIX2hTdFOFtSwcaqutFrQ), dan [typeahead.js](https://translate.googleusercontent.com/translate_c?depth=1&hl=id&prev=search&rurl=translate.google.com.sg&sl=en&sp=nmt4&u=http://twitter.github.io/typeahead.js/&usg=ALkJrhjby_wKUuJzEIFCVr69m0UG3OflzA). Fitur yang terdapat pada bootleaf ini sebagai berikut:

* + - 1. Template peta ramah peta lengkap dengan penanda Navbar dan modal yang responsive.
      2. Memuat jQuery file GeoJSON eksternal.
      3. Penanda marker multiple layer secara logis melalui [plugin client marker leaflet](https://translate.googleusercontent.com/translate_c?depth=1&hl=id&prev=search&rurl=translate.google.com.sg&sl=en&sp=nmt4&u=https://github.com/Leaflet/Leaflet.markercluster&usg=ALkJrhg-DVO2iln5r9MXtldLdhxGrpnIOg).
      4. Penelusuran fitur multi-lapisan sisi client yang elegan dengan pelengkapan otomatis menggunakan [typeahead.js](https://translate.googleusercontent.com/translate_c?depth=1&hl=id&prev=search&rurl=translate.google.com.sg&sl=en&sp=nmt4&u=http://twitter.github.io/typeahead.js/&usg=ALkJrhjby_wKUuJzEIFCVr69m0UG3OflzA).
      5. Daftar fitur sidebar yang responsif dengan penyortiran dan penyaringan via [list.js](https://translate.googleusercontent.com/translate_c?depth=1&hl=id&prev=search&rurl=translate.google.com.sg&sl=en&sp=nmt4&u=http://listjs.com/&usg=ALkJrhgySLkrbtJd7dd6De4Y0w4QStLfVg).
      6. Icon penanda disertakan dalam kontrol lapisan berkelompok melalui [plugin pengontrol lapisan berkelompok](https://translate.googleusercontent.com/translate_c?depth=1&hl=id&prev=search&rurl=translate.google.com.sg&sl=en&sp=nmt4&u=https://github.com/ismyrnow/Leaflet.groupedlayercontrol&usg=ALkJrhieOI4-O5mvmUkN2mEQ_wby_bH0qw).
    1. **MySQL**

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

* + 1. **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.

* 1. **Microsoft Visio**

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

## Balsamiq Mockup

## Balsamiqadalah 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.

* 1. ***Unified Modelling Language* (UML)**

*Unified Modelling Language*  (UML) adalah sebuah standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek (Salahuddin, 2013).

1. Use Case Diagram

Use case atau diagram use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu (Rosa & Salahuddin, 2013).

Syarat penanaman pada *use case* adalah nama didefinisakan menjadi sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*.

* Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem infromasi yang akan dibuat di luar sistem itu sendiri, jadi awalaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
* *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

1. Activity Diagram (diagram aktivitas)

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang pelu diperhatikan disini adalah bahwa diagram aktivitas mengambarkan aktivitas sistem bukan apa yang dilakukan oleh aktor, jadi aktivitas yang dapat di lalukan oleh sistem (Rosa & Salahuddin, 2013).

Diagram aktivitas juga banyak digunakan untuk mendefinisikan hal-hal berikut ini:

* Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
* Urutan atau pengelompokan tampilan dari sistem/user interface dimana setiap akitivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
* Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
* Rancangan menu yang ditampilkan pada perangkat lunak.

Dalam pembuatan activity diagram, penggunaan dan penempatan simbol-simbol harus sesuai dengan deskripsi dari simbol itu sendiri, agar perancangan sistem tergambar dengan jelas.

1. *Class Diagram* (Diagram Kelas)

Diagram kelas atau C*lass Diagram* menggambarkan struktur sistem dari segi pendefinisan kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi. Atribut merupakan variabel-variabel yang dimiliki oleh suatu kelas, sedangkan operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas (Rosa dan Salahuddin, 2013).

Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agat antara dokumentasi perancangan dan perangkat lunak *singkron* (Rosa dan Salahuddin, 2013).

1. *Sequence diagram* (Sekuensial Diagram)

*Sequence diagram* menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirim dan diterima antar objek.

* 1. ***V-Model***

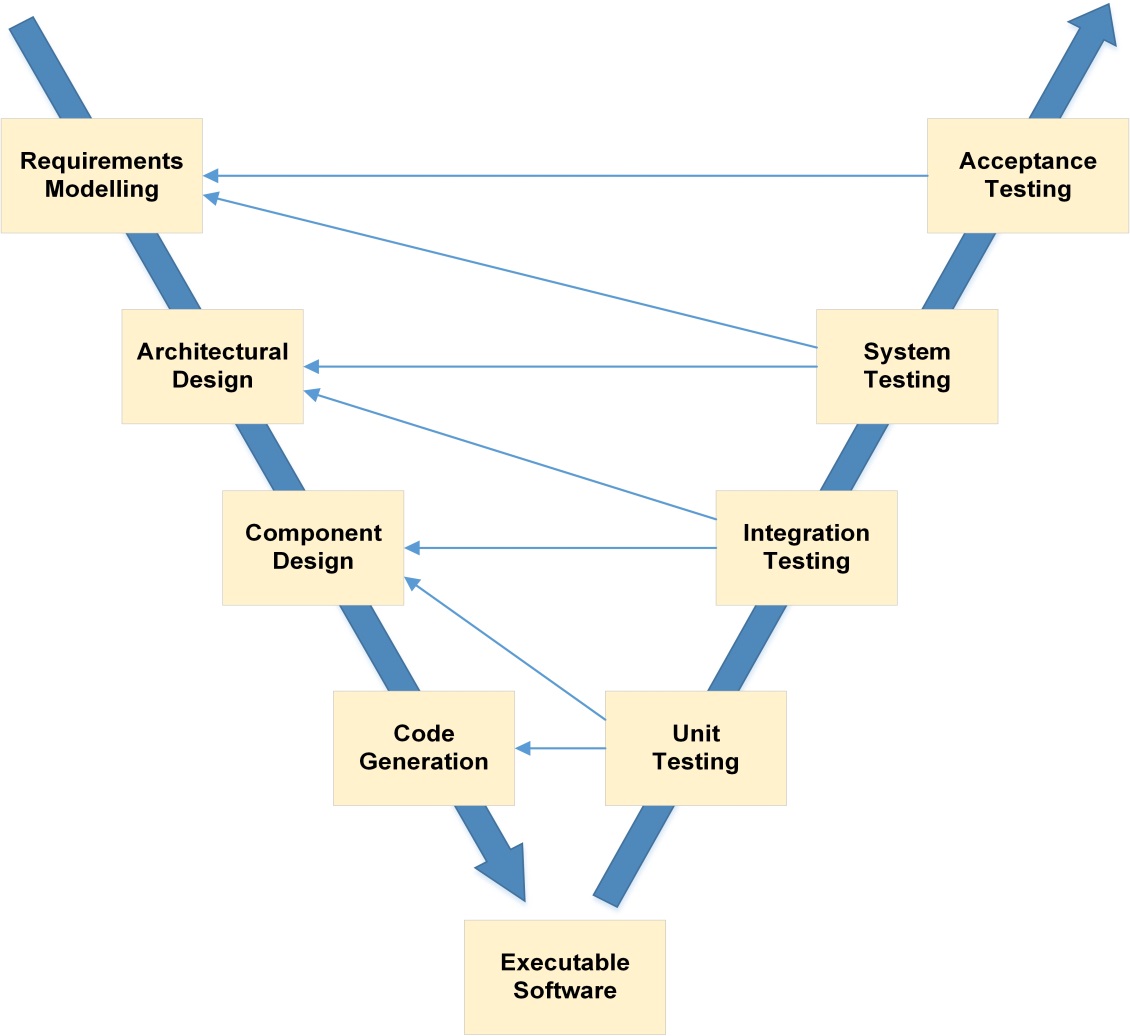
Model ini merupakan perluasan dari model *waterfall*, disebut sebagai perluasan karena tahap-tahapnya mirip dengan yang terdapat dalam model *waterfall*. Jika dalam model *waterfall* proses dijalankan secara linear, maka dalam *V-model* proses dilakukan bercabang (Rifai, 2015).

Model V adalah model yang 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).

Model V menggambarkan relasi aksi-aksi jaminan kualitas pada aksi-aksi yang berkaitan dengan komunikasi, pemodelan, serta aktivitas-aktivitas kontruksi pada tahap awal. Saat tim perangkat lunak bergerak ke bawah pada sisi kiri model-V, 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 model-V 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).

*V-model* memberikan panduan bahwa pengujian perlu dimulai sedini mungkin dalam siklus hidup. Dalam *V-model* ini digambarkan hubungan antara tahap pengembangan software dengan tahap pengujiannya yang dapat dilihat pada Gambar 2.2.

****

Gambar 2.3 V-Model

(Sumber: Pressman, 2010)

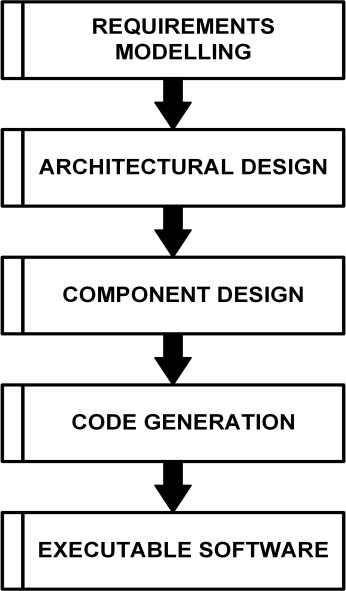
Gambar 2.2 V-Model

(Sumber: Pressman, 2010)

Tahapan yang ada pada V-model dibagi menjadi dua tahap yaitu tahap pengembangan sistem dan tahap pengujian sistem. Tahap pengembangan sistem terdapat empat tahapan yaitu *requirements modeling, architectural design, component design, code generation* dan *executable software*. Sedangkan tahap pengujian sistem terdiri dari empat tahapan yaitu *unit testing, integration sistem, system testing* dan *acceptance system*. Berikut penjelasan masing-masing tahap yang ada pada *V-Model*:

* + 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

1. *Requirements Modelling*

Tahap *requirements modelling* dikenal juga sebagai tahap mendefinisikan rencana aplikasi. Keluaran dari tahap ini adalah dokumentasi kebutuhan pengguna.

1. *Architectural Design*

Sering juga disebut High Level 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.

1. *Component 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.

1. *Code Generation*

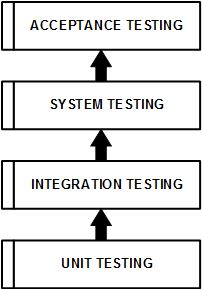
Pada tahap ini akan dilakukan pembuatan kode terhadap perancangan sistem yang telah dilakukan sebelumnya. Tahap ini membutuhkan perangkat lunak dalam pembuatan kode-kode tersebut.

1. *Executable Software* (*Implementation*)

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

* + 1. **Tahap Pengujian Sistem**

Pada tahap pengujian sistem ini terdapat empat tahapan yang dapat dilihat pada Gambar 2.4:



Gambar 2.5 Tahap Pengujian Sistem

* + - * 1. Unit Testing (Pengujian Unit)

Pengujian unit berfokus pada upaya verifikasi terhadap unit terkecil dari perancangan perangkat lunak-komponen atau modul perangkat lunak. Pengujian unit berfokus pada logika pemrosesan internal dan struktur data di dalam batas-batas komponen. Jenis pengujian ini dapat dilakukan secara paralel untuk beberapa komponen (Pressman, 2015).

Pengujian unit biasanya dianggap sebagai tambahan dalam langkah penulisan kode program (*coding*). Perancangan pengujian unit dapat dilakukan sebelum penulisan kode program dimulai atau setelah kode program dihasilkan (Pressman, 2015).

Jenis pengujian ini digerakkan oleh tim arsitektur dan implementasi. Pengujian internal dan unit dapat dilakukan secara otomatis dengan bantuan tool cangkupan (*coverage tool*). Tool cangkupan akan menganalisis kode sumber dan menghasilkan pengujian yang akan melaksanakan setiap urutan eksekusi alternatif. Lalu, pemrograman menggabungkan pengujian tersebut ke dalam kasus untuk memvalidasi hasil dari urutan masing-masing eksekusi (Simarmata, 2010).

Tujuan dilakukannya pengujian ini yaitu untuk memeriksa apakah modul tertentu yang telah dirancang atau kode unit telah berjalan dengan baik. Contoh pengujian yang dilakukan yaitu memeriksa kode program pada *event, procedure*, dan *function*. *Unit testing* ini dilakukan untuk meyakinkan bahwa masing-masing unit dapat berjalan sebagaimana mestinya.

* + - * 1. Integration Testing

Pengujian integrasi adalah teknik sistematik untuk membangun arsitektur perangkat lunak, sementara pada saat yang sama melakukan pengujian untuk menemukan kesalahan-kesalahan yang terkait dengan antarmuka. Tujuannya adalah untuk mengambil komponen yang diuji dan membangun struktur program yang telah ditentukan oleh perancangan (Pressman, 2015).

Pengujian integrasi menjelaskan kecacatan dalam antarmuka dan inteaksi antar komponen terpadu (modul). Semakin besar kelompok komponen perangkat lunak yang diuji terkait dengan elemen-elemen dari desain arsitekturnya akan dipadukan dan diuji sampai perangkat lunak bekerja sebagai sistem (Simarmata, 2010).

Pengujian ini dilakukan untuk memastikan bahwa seluruh kondisi yang terjadi dari hasil interaksi antar unit dapat menghasilkan *output* yang diharapkan. *Integration testing* dapat dilakukan dengan pengujian *blackbox.*

* + - * 1. 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).

*System testing* ini dilakukan terhadap aplikasi yang telah selesai dikembangkan. Pengujian ini dapat dilakukan menggunakan metode *Blackbox, nonfunctional test* (*deployment test, compability test, stress test,* dan lain sebagainya), *thread testing, static testing, automated testing tools*.

* + - * 1. Acceptance Testing

*Aceptance 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.  Tujuan dari pengujian ini adalah untuk mengetahui tingkat kelayakan dari perangakat lunak. Acceptance test dapat dilakukan dengan menggunakan metode *Blackbox*, *usability test*, *thread testing*, *static testing*.

* 1. **Pengujian Perangkat Lunak**

Menurut J.Watkins (2004) dalam bukunya yang berjudul *Testing IT*, menyebutkan bahwa teknik pengujian perangkat lunak dibagi menjadi tiga bagian utama seperti yang ditunjukkan pada Tabel 2.6 berikut:

Tabel 2.5 Teknik Pengujian (Watkins, 2004)

|  |  |  |
| --- | --- | --- |
| ***General Testing*** | ***Functional Testing*** | ***Nonfunctional Testing*** |
| 1. *Positive and negative testing* 2. *Whitebox and blackbox testing* 3. *Error guesing* 4. *Automated software testing* | 1. *Equivalence Partitioning* 2. *Boundary analysis* 3. *Intrusive testing* 4. *Random testing* 5. *State Transition Testing* 6. *State testing* 7. *Thread testing* | 1. *Configuration/Installation testing* 2. *Compability testing* 3. *Documentation and help testing* 4. *Fault recovery testing* 5. *Performance testing* 6. *Reliability testing* 7. *Security testing* 8. *Load testing* 9. *Usability testing* 10. *Volume testing.* |

* + 1. **Metode *Blaxbox 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 *interface* dan *form validation*. Pengujian *interface* adalah pengujian yang dilakukan secara langsung terhadap desain interface yang dibuat pada sistem (Simarmata, 2010).

Tujuan yang diharapkan dalam melakukan Blackbox Testing adalah dapat membuat desain dan fungsi sistem yang sesuai dengan kebutuhan organisasi, lembaga atau perusahaan.

Pengujian black-box berusaha menemukan kesalahan dalam kategori sebagai berikut : (1) fungsi-fungsi yang tidak benar atau hilang (2) kesalahan interface (3) kesalahan pada struktur data atau akses database eksternal, (4) kesalahan kinerja dan performa, dan (5) kesalahan inisialisai dan terminasi.

* + 1. ***Functional Testing***

Pada pengujian ini, perangkat lunak diuji untuk persyaratan fungsional. Pengujian dilakukan dalam bentuk tertulis untuk memeriksa apakah aplikasi berjalan sepertiyang diharapkan. Pengujian fungsional meliputi seberapa baik sistem melaksanakan fungsinya, termasuk perintah-perintah pengguna, manipulasi data, pencarian dan proses bisnis, pengguna layar dan integrasi. Pengujian fungsional juga meliputi permukaan yang jelas dari jenis fungsi-fungsi, serta operasi *back-end* (seperti keamanan dan bagaimana meningkatkan sistem) (Simarmata, 2010).

* + 1. ***Compability Test***

*Compatibility testing* merupakan pengujian yang digunakan untuk memeriksa kemampuan perangkat lunak yang dikembangkan mampu ditransfer dari satu lingkungan ke lingkungan lain. Tujuannya yaitu untuk mengetahui apakah software tersebut dapat berjalan pada hardware, sistem operasi, aplikasi, ataupun lingkungan jaringan yang berbeda.

* + 1. ***User Acceptance Testing***

Pada jenis pengujian ini, perangkat lunak akan diserahkan kepada pengguna untuk mengetahui apakah perangkat lunak memenuhi harapan pengguna dan bekerja seperti yang diharapkan. Padapengembangan perangkat lunak, *User Acceptance Testing* (UAT), juga disebut pengujian beta *(beta testing),* pengujian aplikasi *(application testing),* dan pengujianpengguna akhir *(end user testing)* adalah tahapan pengembangan perangkat lunak ketika perangkat lunak diuji pada ”dunia nyata” yang dimaksudkan oleh pengguna (Simarmata, 2010).

UAT atau uji penerimaan pengguna merupakan suatu proses verifikasi yang dilakukan oleh pengguna terhadap *software* yang dibangun sehingga menghasilkan dokumen yang dapat dijadikan sebagai bukti bahwa *software* yang telah dikembangkan telah dapat diterima oleh pengguna.

* 1. **Penelitian Terdahulu**

Hasil penelitian terdahulu digunakan untuk mengetahui persamaan dan perbedaan dari beberapa penelitian yang ada, Berikut adalah beberapa penelitian terdahulu yang digunakan dalam penelitian ini, yaitu:

1. Penelitian yang pernah dilakukan oleh Gunawan Wibisana (2011) dengan judul Penyediaan Sistem Informasi Geografis Jaringan Jalan di Kabupaten Batang Berbasis Web.

Tujuan dari penelitian ini adalah merancang dan membangun sistem informasi geografis jaringan jalan di Kabupaten Batang secara interaktif yang berbasis web menggunakan Sistem Informasi Geografis (SIG) dengan format html dan pengolahan data dengan Arcview GIS 3.3 dan extensions Mapview SVG.

1. Anggita Ratna (2010) juga melakukan penelitian sebelumnya yaitu Sistem Informasi Geografis Kondisi Jaringan Jalan dan Jembatan Kecamatan Depok, Sleman.

Tujuan dari penelitian ini adalah menghasilkan sistem yang mampu memberikan informasi geografi seperti keadaan suatu jalan dan jembatan, kemampuan sistem untuk mengubah data yang telah ada yang dapat diakses oleh pegawai DPU (Dinas Pekerjaan Umum), dan sistem ini juga dapat membantu mempermudah pekerjaan Pegawai DPU (Dinas Pekerjaan Umum) dalam melakukan survey ke lapangan dengan menggunakan data yang dihasilkan oleh SIG tersebut.

1. Penelitian yang dilakukan oleh Helmi Bahar Alim (2016) tentang Sistem Informasi Geografis Jalan dan Jembatan di Kabupaten Wonogiri.

Tujuan dari penelitian ini adalah menghasilkan suatu sistem informasi geografis jalan dan jembatan yang dapat digunakan untuk memantau jalan dan jembatan di Kabupaten Wonogiri dan dapat di kembangkan lagi menjadi sistem informasi geografis yang lebih komplek. Pengembangan sistem informasi geografis ini diharapkan dapat membantu mempermudah pengelolaan jalan dan jembatan di Kabupaten Wonogiri.

1. Penelitian yang pernah dilakukan oleh Stanlay K. Kimani, Dr. Andrew T. Imwati (2013) yang berjudul A Web Based Geographical Information System For Marketing of Residential House – A Case Study of Karen County Assembly Ward in Nairobi City Country, Kenya.

Tujuan dari penelitian ini adalah untuk menggabungkan penggunaan teknologi GIS dan web untuk mengembangkan Sistem Informasi Geografis berbasis web yang akan berfungsi sebagai platform untuk pemasaran rumah tinggal sehingga mengatasi isu-isu terkait.

1. Penelitian yang dilakukan oleh Hendrika Saputra (2018) yang berjudul Sistem Informasi Geografis Pemetaan Jaringan Pipa Air Berbasis Mapbox Gl Js (Studi Kasus : Dinas Pekerjaan Uumum dan Penataan Ruang Provinsi Riau). Tujuan dari penelitian ini adalah membangun sistem informasi geografis pengelolahan jaringan pipa air DUROLIS dalam upaya meningkatkan pelayanan, perencanaan, pengawasan, perawatan, dan pengaturan operasional jaringan pipa air serta adanya partisipasi peran masyarakat dalam penyampaian informasi ketika terjadi permasalahan pipa di lapangan.

Hasil dari penelitian ini berupa sistem informasi geografis yang menampilkan informasi lokasi jaringan pipa air. Selain itu masyarakat dapat melaporkan permasalahan, selanjutnya laporan masyarakat dapat dijadikan sarana pengambilan keputusan dan tindakan dalam mengatasi permasalahan dengan jaringan pipa air DUROLIS.

1. Penelitian yang dilakukan oleh Febri Mahendra (2014) yang berjudul Sistem Informasi Geografis Berbasis Web Inventarisasi Jalan Kota Pekanbaru Menggunakan Google Maps Api.

Tujuan dari penelitian ini adalah merancang, dan membangun SIG berbasis web yang dapat digunakan sebagai sistem informasi jaringan jalan di Kota Pekanbaru untuk memvisualisasikan data spasial dalam bentuk web-based, sehingga diharapkan dapat terbangun aplikasi geodatabase yang dapat terdistribusi secara baik kepada pengguna yang berkepentingan.