BAB III

OBJEK DAN METODE PENELITIAN

3.1. Objek Penelitian

Penelitian yang dilakukan di Padang sumatera barat bertujuan untuk membuat suatu sistem informasi. Sistem informasi yang akan dibuat adalah sistem informasi Geographich search parawisata kota padang.

3.1.1. Sejarah Singkat Kota Padang

Kota Padang adalah salah satu Kota tertua di pantai barat Sumatera di Lautan Hindia. Menurut sumber sejarah pada awalnya (sebelum abad ke-17) Kota Padang dihuni oleh para nelayan, petani garam dan pedagang. Ketika itu Padang belum begitu penting karena arus perdagangan orang Minang mengarah ke pantai timur melalui sungai-sungai besar. Namun sejak Selat Malaka tidak lagi aman dari persaingan dagang yang keras oleh bangsa asing serta banyaknya peperangan dan pembajakan, maka arus perdagangan berpindah ke pantai barat Pulau Sumatera.

Suku Aceh adalah kelompok pertama yang datang setelah Malaka ditaklukkan oleh Portugis pada akhir abad ke XVI. Sejak saat itu Pantai Tiku, Pariaman dan Inderapura yang dikuasai oleh raja-raja muda wakil Pagaruyung berubah menjadi pelabuhan-pelabuhan penting karena posisinya dekat dengan sumber-sumber komoditi seperti lada, cengkeh, pala dan emas.

Kemudian Belanda datang mengincar Padang karena muaranya yang bagus dan cukup besar serta udaranya yang nyaman dan berhasil menguasainya pada Tahun

1660 melalui perjanjian dengan raja-raja muda wakil dari Pagaruyung. Tahun 1667 membuat Loji yang berfungsi sebagai gudang sekaligus tangsi dan daerah sekitarnya dikuasai pula demi alasan keamanan.

Selanjutnya :

7 Agustus 1669 : Puncak pergolakan masyarakat Pauh dan Koto Tangah melawan Belanda dengan menguasai Loji-Loji Belanda di Muaro, Padang. Peristiwa tersebut diabadikan sebagai tahun lahir kota Padang.

20 Mei 1784 : Belanda menetapkan Padang sebagai pusat kedudukan dan perdagangannya di Sumatera Barat. Padang menjadi lebih ramai setelah adanya Pelabuhan Teluk Bayur.

31 Desember 1799 : Seluruh kekuasaan VOC diambil alih pemerintah Belanda dengan membentuk pemerintah kolonial dan Padang dijadikan pusat kedudukan Residen.

1 Maret 1906 : Lahir ordonansi yang menetapkan Padang sebagai daerah Cremente (STAL 1906 No.151) yang berlaku 1 April 1906.

9 Maret 1950 : Padang dikembalikan ke tangan RI yang merupakan negara bagian melalui SK. Presiden RI Serikat (RIS), No.111 tanggal 9 Maret 1950.

15 Agustus 1950 : SK. Gubernur Sumatera Tengah No. 65/GP-50, tanggal 15 Agustus 1950 menetapkan Pemerintahan Kota Padang sebagai suatu daerah otonom sementara menunggu penetapannya sesuai UU No. 225 tahun 1948. Saat itu kota Padang diperluas, kewedanaan Padang dihapus dan urusannya pindah ke Walikota Padang.

29 Mei 1958 : SK. Gubernur Sumatera Barat No. 1/g/PD/1958, tanggal 29 Mai 1958 secara de facto menetapkan kota Padang menjadi ibukota propinsi Sumatera Barat.

Tahun 1975 : Secara de jure Padang menjadi ibukota Sumatera Barat, yang ditandai dengan keluarnya UU No.5 tahun 1974 tentang Pokok-Pokok Pemerintahan di Daerah, dengan Kotamadya Padang dijadikan daerah otonom dan wilayah administratif yang dikepalai oleh seorang Walikota.

Pada awalnya luas Kota Padang adalah 33 Km2, yang terdiri dari 3 Kecamatan dan 13 buah Kampung, yaitu Kecamatan Padang Barat, Padang Selatan dan Padang Timur. Dengan Undang-undang Nomor 5 Tahun 1979 dan Peraturan Pemerintah Nomor 17 Tahun 1980 tanggal 21 Maret 1980 wilayah Kota Padang menjadi 694,96 Km2, yang terdiri dari 11 Kecamatan dan 193 Kelurahan. Dengan dicanangkannya pelaksanaan otonomi daerah sejak Tanggal 1 Januari 2001, maka wilayah administratif Kota Padang dibagi dalam 11 Kecamatan dan 103 Kelurahan. Dengan Keluarnya Peraturan Daerah Kota Padang Nomor 16 Tahun 2004 tentang Pembentukan organisasi Kelurahan Maka jumlah Kelurahan di Kota Padang menjadi 104 Kelurahan.

3.2. Metode Penelitian

Pada subbab ini penulis akan menerangkan metode penelitian yang akan digunakan dalam penelitian ini. Metode penelitian ini meliputi desain penelitian, jenis dan metode pengumpulan data, serta metode pendekatan dan pengembangan sistem. Desain penelitian yang digunakan oleh penulis adalah Deskriptif dan

Action Research. Sedangkan jenis pengumpulan data menggunakan data primer, dan data sekunder. Metode pendekatan sistem yang penulis pilih untuk perancangan aplikasi ini adalah pendekatan berorientasi objek dan metode pengembangan sistem yang digunakan adalah *Rational Unified Process (RUP).*

3.2.1. Desain Penelitian

Penelitian ini menggunakan metode Deskriptif dan *Action Research*. Penelitian deskriptif dapat diartikan sebagai proses pemecahan masalah yang diselidiki dengan melukiskan keadaan subyek dan obyek pada saat sekarang berdasarkan fakta-fakta yang ada. Dimana dalam metode ini menggambarkan semua data yang kemudian dianalisis dan dibandingkan berdasarkan kenyataan yang sedang berlangsung dan selanjutnya mencoba untuk memberikan pemecahan masalahnya.

Pelaksanaan metode penelitian deskriptif tidak terbatas sampai pada pengumpulan dan penyusunan data, tetapi meliputi analisis dan interpretasi tentang data tersebut, selain itu semua yang dikumpulkan memungkinkan menjadi kunci terhadap apa yang diteliti.

Penelitian ini menggunakan metode *Action Research* (penelitian tindakan). Metode *Action Research* atau tindakan merupakan penelitian langsung, disertai dengan praktek di lapangan. Membuat suatu program yang akan dilaksanakan secara sistematis dan terencana, serta mempunyai nilai perbaikan yang signifikan. Penelitian tindakan ini lebih efektif, karena akan terlihat langsung hasilnya. Salah satu syarat dalam melakukan penelitian tindakan adalah adanya keinginan dari

orang yang memilki masalah untuk mengidentifikasi masalah yang ada dan mempunyai keinginan untuk memecahkannya.

3.2.2. Jenis dan Metode Pengumpulan Data

Metode pengumpulan data merupakan suatu teknik atau cara untuk memperoleh, mencari, mengumpulkan dan mencatat data yang di gunakan untuk menyusun karya . Adapun pengumpulan data itu sendiri terbagi menjadi dua yaitu data primer dan data sekunder.

3.2.2.1. Sumber Data Primer

Data primer adalah data yang hanya dapat kita peroleh dari sumber asli (nara sumber) / responden penelitian baik melalui pengamatan maupun pencatatan terhadap objek penelitian.

Dalam penelitian ini penulis mengumpulkan data primer dengan menggunakan teknik :

1. Pengamatan Langsung (Observasi)

Observasi merupakan suatu cara pengumpulan data melalui pengamatan dan pencatatan oleh pengumpul data terhadap gejala atau peristiwa yang diselidiki pada obyek penelitian secara langsung tetapi tidak ada interaksi dengan objek yang diteliti dengan cara melakukan tinjauan langsung ke lapangan.

Penulis melakukan tinjauan langsung ke lapangan untuk mencari data-data yang diperlukan terhadap objek yang diteliti di bagian tata usaha di kota padang.

2. Wawancara

Wawancara merupakan proses memperoleh keterangan untuk tujuan penelitian dengan cara tanya jawab, sambil bertatap muka antara si penanya atau pewawancara dengan si penjawab atau responden. Dalam penelitian ini penulis melakukan wawancara secara langsung dengan mengajukan berbagai pertanyaan kepada pihak yang ikut terlibat langsung.

3.2.2.2. Sumber Data Sekunder

Data sekunder adalah data yang digunakan untuk mendukung data primer, merupakan jenis data yang sudah diolah terlebih dahulu oleh pihak pertama, data sekunder diambil secara tidak langsung dari objek penelitian misalnya data ini diperoleh dari buku-buku, jurnal, tutorial, internet dan lain-lain.

Dokumentasi adalah teknik pengumpulan data dengan cara mengumpulkan dokumen-dokumen yang berhubungan dengan objek penelitian. Dalam hal ini, dokumen yang diperoleh akan dianalisis agar diperoleh data yang sesuai dengan penelitian.

3.2.3. Metode Pendekatan dan Pengembangan Sistem

Dalam merancang suatu sistem dibutuhkan sebuah metode pendekatan dan pengembangan sistem untuk membantu dalam proses perancangan dan pembuatannya. Berikut adalah penjelasan metode pendekatan dan pengembangan sistem yang akan digunakan.

3.2.3.1. Metode Pendekatan Sistem

Pada dasarnya saat ini pengembangan sistem dapat dikategorikan dalam 2 pendekatan pengembangan, yaitu pengembangan secara terstruktur daan pengembangan secara *object oriented* (berorientasi objek). Metode pendekatan yang penulis ambil adalah pendekatan berorientasi objek yang divisualisasikan dengan UML.

Dalam jurnal yang ditulis oleh Hendrik, Fathul Wahid yang berjudul "Pengembangan Learning Management System" menerangkan bahwa :

"Metode berorientasi obyek (*object oriented method*) merupakan suatu cara pengembangan perangkat lunak dan sistem informasi berdasarkan abstraksi obyek-obyek yang ada di dunia nyata. Brooks (1987) – dalam Nugroho (2002) – menyatakan bahwa bagian tersulit dari pengembangan perangkat lunak dan / atau sistem informasi adalah tahap analisis, yang didalamnya abstraksi dilakukan. Dalam metode berorientasi obyek kejadian atau hubungan antar entitas dalam dunia nyata direpresentasikan dalam obyek-obyek. "

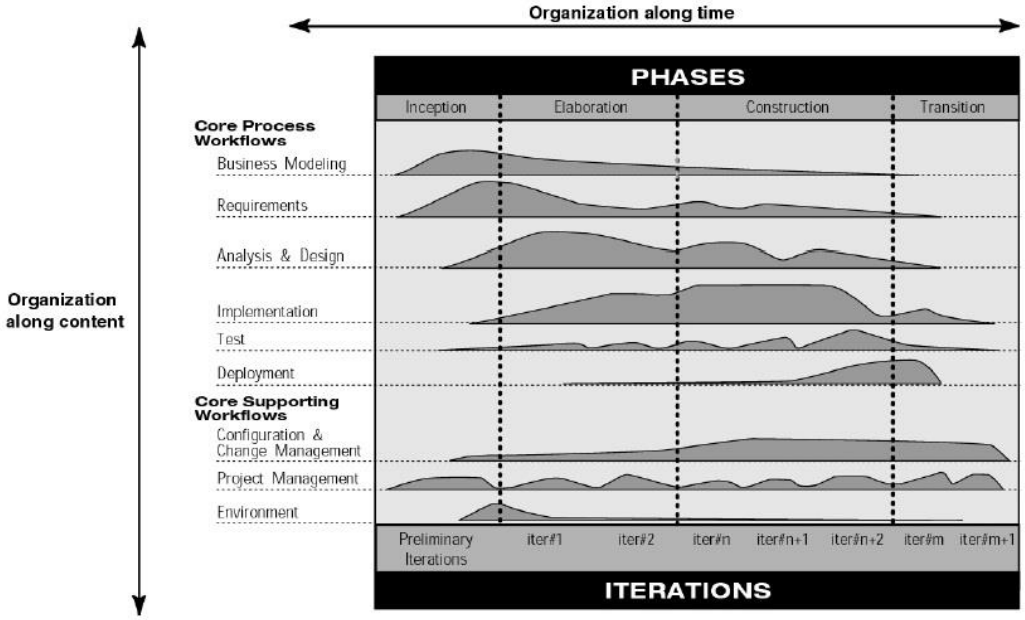
3.2.3.2. Metode Pengembangan Sistem

Metode pengembangan yang penulis pakai adalah metode *Rational Unified Process* (RUP). *Rational Unified Process* (RUP) merupakan suatu metode rekayasa perangkat lunak yang dikembangkan dengan mengumpulkan berbagai *best practises* yang terdapat dalam industri pengembangan perangkat lunak. Ciri utama metode ini adalah menggunakan *use-case driven* dan pendekatan iteratif untuk siklus pengembangan perangkat lunak.

RUP menggunakan konsep *object oriented*, dengan aktifitas yang berfokus

pada pengembangan model dengan menggunakan *Unified Model Language* (UML). Melalui gambar dibawah dapat dilihat bahwa RUP memiliki, yaitu:

1. **Dimensi pertama** digambarkan secara horizontal. Dimensi ini mewakili aspek-aspek dinamis dari pengembangan perangkat lunak. Aspek ini dijabarkan dalam tahapan pengembangan atau fase. Setiap fase akan memiliki suatu *major milestone* yang menandakan akhir dari awal dari phase selanjutnya. Setiap phase dapat berdiri dari satu atau beberapa iterasi. Dimensi ini terdiri atas *Inception, Elaboration, Construction,* dan *Transition.*
2. **Dimensi kedua** digambarkan secara vertikal. Dimensi ini mewakili aspek-aspek statis dari proses pengembangan perangkat lunak yang dikelompokkan ke dalam beberapa disiplin. Proses pengembangan perangkat lunak yang dijelaskan kedalam beberapa disiplin terdiri dari empat elemen penting, yakni *who is doing*, *what, how* dan *when*. Dimensi ini terdiri atas *Business Modeling, Requirement, Analysis and Design, Implementation, Test, Deployment, Configuration* dan *Change Manegement, Project Management, Environtment.*



Gambar 3.1. Arsitektur *Rational Unified Process* (RUP)

(Sumber : *Rational-The Software Development Company, Rational Unified Process: Best Practices for Software Development Teams*)

3.2.3.3 Alat Bantu Analisis dan Perancangan

Pada perancangan sistem informasi parawisata, penulis menggunakan metode pendekatan berorientasi objek, terdapat banyak alat bantu yang dapat digunakan dalam analisis dan desain berorientasi obyek menyebabkan munculnya ketidakjelasan alat bantu apa yang paling unggul. Para pengguna harus memilih diantara bahasa pemodelan dan alat bantu pemodelan yang beragam. Untuk mengatasi masalah tersebut maka *Object Management Group* (OMG) kemudian mengeluarkan UML, dimana dengan adanya UML ini diharapkan dapat mengurangi kekacauan dalam bahasa pemodelan yang selama ini terjadi dalam lingkugan industri perangkat lunak.

1. *Use Case Diagram*

Diagram *use case* menjelaskan manfaat sistem jika dilihat menurut pandangan orang yang berada diluar sistem (aktor). Diagram ini menunjukkan fungsionalitas suatu sistem atau kelas dan bagaimana sistem berinteraksi dengan dunia luar.

Use case diagram dapat digunakan selama proses analisis untuk menangkap *requirements* sistem dan untuk memahami bagaimana sistem seharusnya bekerja. Selama tahap desain, *use case* diagram menetapkan perilaku (*behavior*) sistem saat diimplementasikan. Dalam sebuah model mungkin terdapat satu atau beberapa *use case* diagram.

Dalam diagram *use-case* terdapat beberapa hubungan antar *use-case* antara lain: *Extend* dan *Include*. Hubungan *include* menunjukkan bahwa sebuah *business use-case* selalu termasuk dalam alur kerja dari *business use-case* lainnya. Hubungan *extend* menunjukkan bahwa sebuah *use-case* merupakan kelanjutan dari *business use-case* lainnya jika suatu syarat tertentu telah dipenuhi.

2. *Activity Diagram*

Activity diagram (diagram aktivitas) adalah diagram *flowchart* yang disempurnakan. Diagram aktivitas menggambarkan operasi pada suatu obyek atau proses pada sebuah organisasi. Kelebihan diagram aktivitas dibandingkan dengan diagram *flowchart* adalah adanya dukungan konkurensi (pelaksanaan aktivitas secara bersamaan), pengiriman pesan dan *swimlane* (pelaku/penanggung jawab aktivitas).

Diagram aktivitas sangat berguna ketika kita ingin menggambarkan perilaku paralel atau menjelaskan bagaimana perilaku dalam berbagai *use case* berinteraksi. Diagram aktivitas diawali dengan lingkaran hitam, dan diakhiri dengan lingkaran hitam bertepi putih. Aktivitas digambarkan dengan kotak persegi panjang bersudut lengkung. Setiap aktivitas dihubungkan dengan panah dari awal hingga akhir diagram aktivitas.

Sama halnya dengan diagram *flowchart*, diagram aktivitas pun memiliki simbol yang sama untuk menggambarkan keputusan. Keputusan digambarkan dengan intan, namun deskripsi kondisi yang menyertai keputusan diletakkan diluar simbol intan. Diagram aktivitas dapat menggambarkan konkurensi, yaitu satu atau lebih aktivitas yang berjalan secara bersamaan. Konkurensi diawali dengan sebuah garis tebal horizontal yang menjadi tempat keluarnya garis aktivitas. Konkurensi juga diakhiri dengan garis tebal horizontal.

3. *Sequence Diagram*

Sequence diagram terdiri antar dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait). Sebuah *Sequence diagram*, secara khusus, menjabarkan *behavior* sebuah skenario tunggal. Diagram tersebut menunjukkan sebuah objek contoh dan pesan-pesan yang melewati objek-objek ini di dalam *use case*.

Sequence diagram biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai *respons* dari sebuah *event* untuk menghasilkan *output* tertentu.

4. *Class Diagram*

Diagram *class* menggambarkan seluruh *class* yang terlibat dalam sebuah perangkat lunak. Diagram ini menggambarkan seluruh atribut dan operasi suatu *class* beserta hubungannya dengan *class* lainnya.

Class Diagram mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat diantara mereka. *Class Diagram* juga menunjukkan *property* dan operasi sebuah *class* dan batasan-batasan yang terdapat dalam hubungan-hubungan objek tersebut.

5. *Object Diagram*

Object diagrams memodelkan pandangan statik terhadap sistem untuk memodelkan struktur objek.

Menurut Bambang Heriyanto (2004:291) *Object Diagram* adalah diagram instan (*instance diagram*) yang mendeskripsikan instan-instan kelas.

Pemodelan struktur objek melibatkan *snapshot* dari objek-objek sistem pada suatu waktu. *Object diagrams* merepresentasikan satu *frame* statik papan cerita dinamis dari diagram interaksi.

Kegunaan *object diagram* adalah mendeskripsikan bagaimana kumpulan objek tertentu saling berhubungan. *Object diagrams* adalah diagram instan yang mendeskripsikan instan-instan kelas. Diagram instan berguna untuk dokumentasi skenario, serta kasus pengujian dan mendiskusikan contoh-contoh instanisasi *class diagrams*.

6. *Deployment Diagram*

Diagram *deployment* menggambarkan bagaimana konfigurasi perangkat keras dimana sistem akan diinstalasikan. Diagram ini memberi simbol untuk segala bentuk perangkat keras, seperti: komputer, printer, router, hub, jaringan, dan sebagainya.

Diagram *deployment* menunjukkan tata-letak perangkat keras secara fisik, dan komponen yang terdapat didalamnya. Diagram ini menunjukkan secara lengkap arsitektur dari sistem dan perangkat kerasnya. Diagram ini akan memperjelas pengguna bagaimana hasil akhir dari sistem yang dikembangkan, dan mempermudah teknisi dalam pemeliharaan sistem.

Deployment diagram bisa juga berisi komponen-komponen, masing-masing komponen itu berada di suatu node. Juga dapat berisi paket atau subsistem yang digunakan mengelompokkan elemen-elemen di model menjadi potongan-potongan.

3.3 Pengujian Software

Penulis akan menggunakan teknik *black box testing* dalam pengujian *software*. Pengujian *black box* adalah pengujian aspek fundamental sistem tanpa memperhatikan struktur logika internal perangkat lunak. Metode ini digunakan untuk mengetahui apakah perangkat lunak berfungsi dengan benar. Pengujian *black box* merupakan metode perancang data uji yang didasarkan pada spesifikasi perangkat lunak. Data uji dijalankan, dieksekusi pada perangkat lunak dan

kemudian keluaran dari perangkat lunak dicek apakah telah sesuai dengan yang diharapkan.

Pada pengujian *black box,* mencoba beragam masukan dan memeriksa keluaran yang dihasilkan. Teknik pengujian *black box* juga dapat digunakan untuk pengujian berbasis skenario, dimana isi dalam sistem mungkin tidak tersedia untuk diinspeksi tapi masukan dan keluaran yang didefinisikan dengan *use case* dan informasi analisis yang lain.

Beberapa jenis kesalahan yang dapat di identifikasi :

1. Fungsi tidak benar atau hilang
2. Kesalahan antar muka
3. Kesalahan inisialisasi dan akhir program
4. Kesalahan performasi.