BAB III

OBJEK DAN METODOLOGI PENELITIAN

3.1. Objek Penelitian

Penelitian ini dilakukan di Kebun Binatang Bandung yang berlokasi di Jl. Kebun Binatang No. 6, Bandung.

3.1.1. Sejarah Singkat Kebun Binatang Bandung

Kebun Binatang Bandung ini pada awalnya dikenal dengan nama Derenten (dalam bahasa sunda, dierentuin) yang artinya kebun binatang. Kebun Binatang Bandung didirikan pada tahun 1930 oleh Bandung Zoological Park (BZP), yang dipelopori oleh Direktur Bank Dennis, Hoogland. Pengesahan pendirian Kebun Binatang ini diwenangi oleh Gubernur Jenderal Hindia Belanda dan pengesahannya dituangkan pada keputusan 12 April 1933 No.32. Pada saat Jepang menguasai daerah ini, tempat wisata ini kurang terkelola, hingga pada tahun 1948, dilakukan rehabilitasi untuk mengembalikan fungsi tempat wisata ini.

Pada tahun 1956, atas inisiatif dari R. Ema Bratakoesoema, Bandung Zoological Park dibubarkan dan berganti menjadi Yayasan Marga Satwa Tamansari pada tahun 1957.

Kebun Binatang Bandung mempunyai luas lahan sekitar 13.5 Ha dengan topologi bergelombang, menjadikan tempat ini begitu unik dan tidak membosankan. Dari luas 13.5 Ha itu, dibuat beberapa wahana dengan penggunaan sekitar 18,25% untuk area kandang-kandang hewan, sekitar 55,20% untuk area pertamanan dan lesehan, 4,7% digunakan untuk taman ria dan juga kolam perahu,

dan sekitar 2,4% untuk tempat pengolahan sampah, sisanya digunakan untuk berbagai bangunan seperti kantor, masjid, museum, teater, aquarium, dan jalan.

3.1.2. Visi dan Misi Kebun Binatang Bandung

Visi dan Misi Kebun Binatang Bandung adalah sebagai berikut :

3.1.2.1. Visi

Mewujudkan Terpeliharanya :

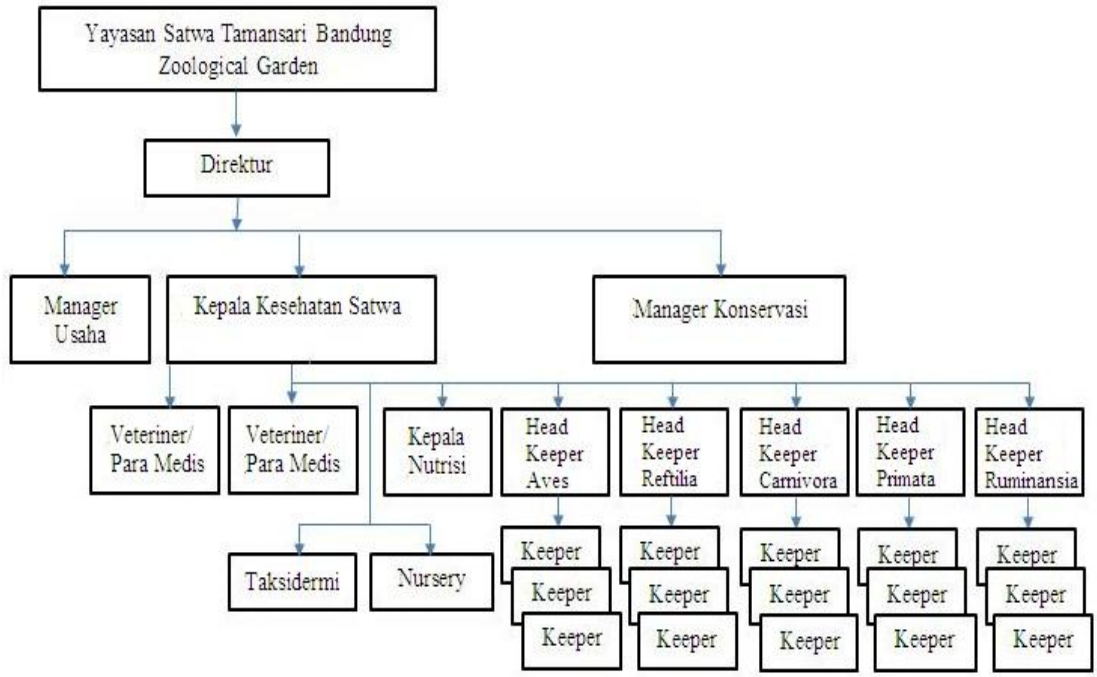
1. Kelestarian keanekaragaman jenis satwa dan tumbuhan dari kepunahan;
2. Kemurnian genetik jenis satwa dan tumbuhan;
3. Keseimbangan ekosistem yang ada.

3.1.2.2. Misi

1. Melaksanakan upaya pengelolaan satwa liar sesuai dengan kaidah-kaidah konservasi dan teknologi yang benar.
2. Mengembangkan ilmu pengetahuan dan teknologi bidang kedokteran hewan, biologis, lingkungan dan disiplin ilmu terkait dalam mensejahterakan dan mengembangbiakan satwa liar, terutama satwa langka dan terancam punah.
3. Melaksanakan prosedur rehabilitasi dan pelatihan satwa liar dalam upaya pelepasan kembali ke habitatnya (*in-situ)*.
4. Melaksanakan peragaan satwa koleksi dengan benar dan beretika dengan tujuan pendidikan dan penyuluhan tentang konservasi serta menyediakan sarana rekreasi edukatif yang sehat untuk masyarakat.

5. Melakukan kerja sama dengan lembaga-lembaga konservasi, baik dalam skala lokal maupun nasional, baik regional maupun internasional dalam bentuk alih ilmu pengetahuan dan teknologi, tukar menukar satwa, serta bantuan-bantuan teknis dan finansial.

3.1.3. Struktur Organisasi Kebun Binatang Bandung



Gambar 3.1 Struktur Organisasi Kebun Binatang Bandung

3.1.4. Job Deskripsi

Terlampir

3.2. Metode Penelitian

Metodologi penelitian adalah sekumpulan peraturan, kegiatan, dan prosedur yang digunakan oleh pelaku suatu disiplin ilmu. Metodologi juga

merupakan analisis teoritis mengenai suatu cara atau metode. Penelitian merupakan suatu penyelidikan yang sistematis untuk meningkatkan sejumlah pengetahuan, juga merupakan suatu usaha yang sistematis dan terorganisasi untuk menyelidiki masalah tertentu yang memerlukan jawaban.

3.2.1. Desain Penelitian

Metode penelitian yang digunakan oleh penulis adalah metode penelitian deskriptif dan *Action.* Menurut Erlina (2008:20) penelitian deskriptif adalah penelitian terhadap fenomena atau populasi tertentu yang diperoleh oleh peneliti dari subjek berupa individu, organisasional, industri atau perspektif yang lain. Penelitian deskriptif dilakukan untuk menjawab pertanyaan tentang: apa, siapa, kapan, dimana,dan bagaimana yang berkaitan dengan karakteristik populasi atau fenomena tersebut.

Metode *Action* atau tindakan merupakan penelitian dimana penulis berupaya

untuk memecahkan masalah dalam dunia nyata dengan mengkaji pengalaman-pengalaman dalam memecahkan masalah tersebut.

3.2.2. Jenis dan Metode Pengumpulan Data

Data yang diperlukan dalam penelitian ini terdiri dari data primer dan data sekunder. Data primer yang diperlukan dalam penelitian ini dikumpulkan dengan melakukan penelitian lapangan yaitu datang langsung ke Kebun Binatang Bandung. Data sekunder dikumpulkan melalui penelitian kepustakaan.

3.2.2.1. Sumber Data Primer

Sumber data primer yaitu sumber data yang berasal dari perorangan, kelompok, panel atau sumber terselubung. Dalam memperoleh data primer penulis melakukan wawancara dan observasi di tempat penelitian.

1. Pengamatan Langsung (Observasi)

   Proses pengumpulan data dengan mengamati objek yang akan diteliti secara langsung di tempat penelitian yaitu di Kebun Binatang Bandung. Meliputi bagaimana proses pembelian tiket dan pemanduan wisata yang berjalan di Kebun Binatang Bandung.

2. Wawancara (Interview)

   Proses pengumpulan data dengan melakukan dialog langsung kepada orang yang dianggap dapat memberikan informasi yang benar mengenai Kebun Binatang Bandung yang dibutuhkan oleh penulis yaitu kepada salah satu staff Kebun Binatang Bandung.

3.2.2.2. Sumber Data Sekunder

Sumber data sekunder adalah data yang digunakan untuk mendukung data primer. Jenis data primer merupakan jenis data yang sudah diolah terlebih dahulu oleh pihak pertama. Pengumpulan data yang dilakukan penulis dengan cara membaca dan mempelajari buku-buku yang berhubungan dengan masalah yang diteliti untuk melengkapi dan menunjang data-data yang telah dipelajari dalam laporan. Data sekunder yang diperoleh dari intansi terkait seperti struktur organisasi, uraian tugas dan fungsi dari struktur organisasi (*job Deskription*).

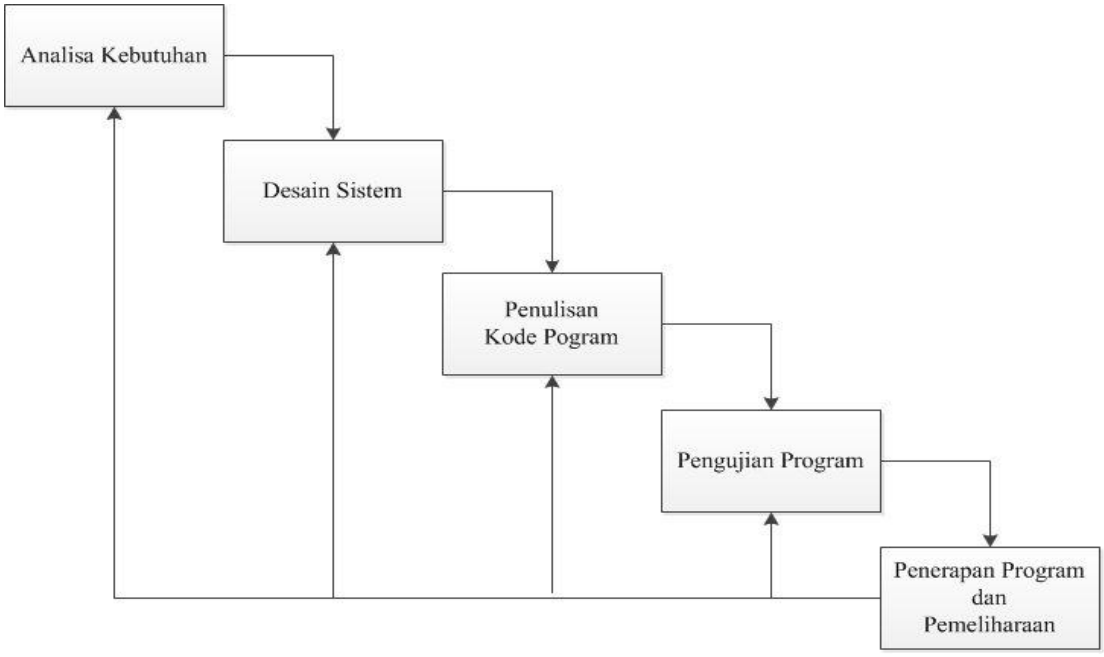
3.2.3. Metode Pendekatan Sistem dan Pengembangan Sistem

Metode pendekatan dan pengembangan sistem yang penulis gunakan dalam "**Membangun Aplikasi Mobile Pemandu Wisata Kebun Binatang Bandung Berbasis Android**" ini adalah sebagai berikut:

3.2.3.1. Metode Pendekatan Sistem

Metode pendekatan yang digunakan adalah metode pendekatan berorientasi objek dengan menggunakan metode pemodelan UML, yang mencakup analisis dan desain yang disebut OOAD (*Object Oriented Analysis and Design*). *Object Oriented Analysis* merupakan metode analisa yang memeriksa *requirement* (syarat atau keperluan yang harus dipenuhi suatu sistem) dari sudut pandang kelas-kelas dan objek-objek yang ditemui dalam ruang lingkup permasalahan. Sedangkan *objek Oriented Design* adalah metode untuk mengarahkan arsitektur *software* yang didasarkan pada manipulasi objek-objek sistem atau subsistem.

3.2.3.2. Metode Pengembangan Sistem



Gambar 3.2 Metode Waterfall

[Sudarman,Dony Ariyus, 2007, *Interaksi manusia dan komputer*, Andi Yogyakarta, Yogyakarta.]

Metode pengembangan sistem yang penulis gunakan yaitu metode *waterfall* sebagai alternatif pengembangan sistem informasi dengan tujuan untuk mendapatkan kualitas sistem yang baik.

Metode *waterfall* merupakan metode yang sering digunakan oleh analis sistem pada umumnya. Inti dati metode *waterfall* adalah pengerjaan dari suatu sistem dilakukan secara berurutan secara linear. Jadi jika langkah satu belum dikerjakan maka tidak akan dapat melakukan pengerjaan ke langkah selanjutnya.

Secara garis besar metode *waterfall* mempunyai langkah-langkah sebagai berikut:

1. **Analisa**

   Langkah ini merupakan analisa terhadap kebutuhan sistem. Pengumpulan data dalam tahap ini bisa dengan melakukan sebuah

penelitian, wawancara atau studi literatur. Seorang sistem analis akan menggali informasi sebanyak-banyaknya dari *user* sehingga akan tercipta sebuah sistem yang dapat melakukan tugas-tugas yang diinginkan oleh *user* tersebut. Tahapan ini akan menghasilkan dokumen *user requirement* atau keinginan pengguna dalam pembuatan aplikasi ini. Dokumen ini lah yang akan menjadi acuan sistem analis untuk menterjemahkan kedalam bahasa pemograman.

2. Design

Proses desain akan menterjemahkan syarat kebutuhan ke sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat *coding.* Proses ini berfokus pada : struktur data, arsitektur perangkat lunak, representasi *interface,* dan detail (algoritma) prosedural. Tahapan ini akan menghasilkan dokumen yang disebut *software requirement.* Dokumen inialh yang akan digunakan *programer* untuk melakukan aktifitas pembuatan sistemnya.

3. Coding & Testing

Coding merupakan penerjemahan *design* dalam bahasa yang bisa dikenali oleh komputer. Dilakukan oleh *programer* yang akan menterjemahkan fungsi yang diminta oleh *user.* Tahapan ini lah yang merupakan tahapan secara nyata dalam mengerjakan suatu sistem.

4. Pengujian Black Box

Metode pengujian yang digunakan oleh penulis adalah black-box testing. Black-box testing digunakan untuk menguji fungsi-fungsi dari

perangkat lunak yang dirancang. Pengujian black-box berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian black-box memungkinkan perekayasaan perangkat lunak mendapat serangkaian kondisi input yang sepenuhnya menggunakan semua pendekatan komlementer yang kemungkinan besar mampu mengungkapkan kelas kesalahan. Pengujian black-box berusaha menemukan kesalahan-kesalahan berikut:

1. fungsi yang tidak sesuai, salah, atau hilang.
2. Kesalahan antar muka.
3. Kesalahan struktur data atau akses database eksternal.
4. Kesalahan kinerja dan inisialisasi kesalahan terminasi.

5. Penerapan

Tahapan ini adalah tahapan *final* dalam pembuatan sebuah sistem atau rancangan. Setelah melakukan analisa, design dan pengkodean maka sistem atau rancangan yang sudah jadi akan digunakan oleh *user.*

3.2.3.3. Alat Bantu dan Pengembangan Sistem

1. Use Case Diagram

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah apa yang dapat diperbuat sistem. Sebuah *use case* mempresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu misalnya login sistem, menampilkan informasi binatang, dan sebagainya. Seorang

aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu.

Use case diagram dapat sangat membantu bila kita sedang menyusun *requitement* sebuah sistem, mengkomunikasikan rancangan klien, dan merancang *test case* untuk semua fitur yang ada pada sistem.

2. Activity Diagram

Activity diagram atau yang biasa disebut diagram aktivitas digunakan untuk mendokumentasikan alur kerja pada sebuah *sistem,* yang di mulai dari pandangan *business level* hingga ke *operational level,* adapun penjelasanya sebagai berikut:

1. *Activity* diagram merupakan cara lain memodelkan aliran kejadian.
2. *Activity* diagram menggambarkan berbagai alir aktivitas dalam sistem yang di rancang bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir.
3. *Activity* diagram juga dapat menggambarkan proses parallel yang mungkin terjadi pada beberapa eksekusi.

3. Class Diagram

Class diagram adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek.

Class diagram menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi). *Class* diagram menggambarkan struktur dan deskripsi *class, package* dan objek beserta hubungan antara *class.*

4. Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display,* dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence* diagram terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait). *Sequence* diagram biasa digunakan untuk menggambarkan sekenario atau rangkaian langkah-langkah yang dilakukan sebagai *respons* dari sebuah *event* untuk menghasilkan output tertentu. Diawali dari apa yang mem-*triger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan output apa yang dihasilkan. Masing-masing objek, termasuk aktor, memiliki *lifeline* vertikal.

Message digambarkan sebagai garis berpanah dari satu objek ke objek lainya. Pada *fase* desain berikutnya, *message* akan dipertakan menjadi operasi/metoda dari class. *Activation bar* menunjukan lamanya eksekusi sebuah proses, biasanya diawali dengan diterimanya sebuah message. Untuk objek-objek yang memiliki sifat khusus, standar UML mendefinisikan icon khusus untuk objek *boundary, controller dan persistent entity.*

5. Component Diagram

Component diagram menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan (*dependency*) di antaranya. Komponen piranti lunak adalah modul berisi *code,* baik berisi *source code* maupun *binary code,* baik *library* maupun *executable,* baik yang muncul pada *compile time, link time,* maupun *run time.* Umumnya komponen terbentuk dari beberapa *class* dan/atau *package,* tapi dapat juga dari komponen-komponen yang lebih kecil. Komponen dapat juga berupa *interface,* yaitu kumpulan layanan yang disediakan sebuah komponen untuk komponen lain.

6. Deployment Diagram

Deployment/physical diagram menggambarkan detail bagaimana komponen di-*deploy* dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, *server* atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi *server,* dan hal-hal lain yang bersifat fisikal. Sebuah *node* adalah *server, wirkstation,* atau piranti keras lain yang digunakan untuk men-*deploy* komponen dalam lingkungan sebenarnya, hubungan antar *node* (misalnya TCP/IP) dan *requitement* dapat juga didefinisikan dalam diagram ini.

3.2.4. Pengujian Software

Pengujian software adalah elemen kritis dari jaminan kualitas perangkat lunak dan mempersentasikan kajian pokok dari spesifikasi, desain dan pengkodean.

3.2.4.1. Pengujian Black Box

Metode pengujian yang digunakan oleh penulis adalah black-box testing. Black-box testing digunakan untuk menguji fungsi-fungsi dari perangkat lunak yang dirancang. Pengujian black-box berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian black-box memungkinkan perekayasaan perangkat lunak mendapat serangkaian kondisi input yang sepenuhnya menggunakan semua pendekatan komlementer yang kemungkinan besar mampu mengungkapkan kelas kesalahan. Pengujian black-box berusaha menemukan kesalahan-kesalahan berikut:

1. fungsi yang tidak sesuai, salah, atau hilang.
2. Kesalahan antar muka.
3. Kesalahan struktur data atau akses database eksternal.
4. Kesalahan kinerja dan inisialisasi kesalahan terminasi.

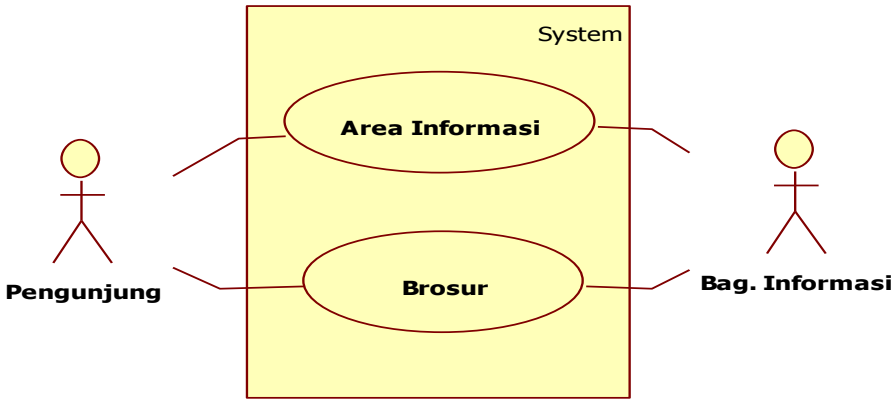
3.3. Analisis Sistem yang Berjalan

Analisis sistem yang sedang berjalan adalah penguraian dari suatu sistem informasi yang utuh dari suatu organisasi atau perusahaan menjadi bagian-bagian yang lebih kecil dengan tujuan untuk mengevaluasi sistem yang sedang berjalan sehingga teridentifikasi masalah-masalah yang ada pada sistem tersebut serta menentukan kebutuhan-kebutuhan dari sistem yang akan dibangun, sehingga dapat diusulkan perbaikan-perbaikan.

3.3.1. Analisis Prosedur yang Sedang Berjalan

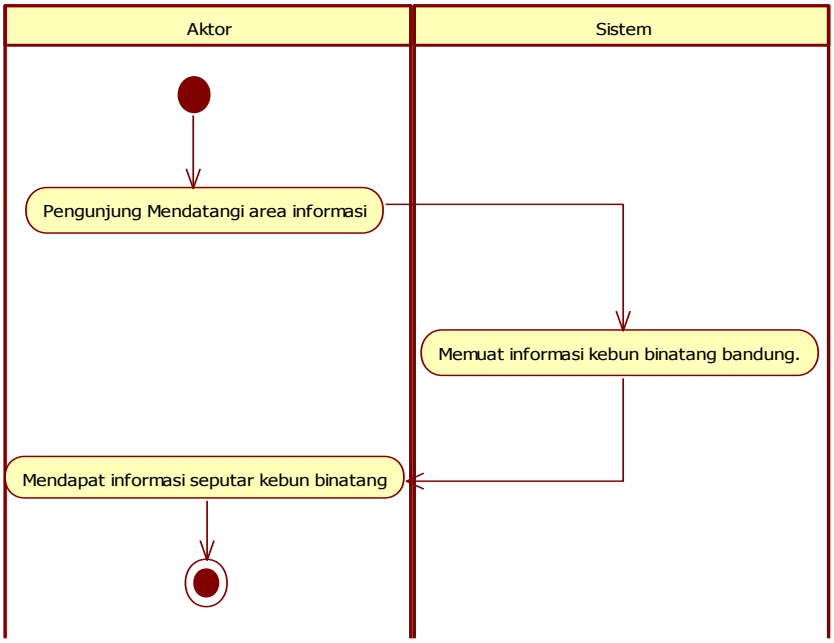
Berdasarkan hasil penelitian yang dilakukan, prosedur pada proses media yang sedang berjalan sekarang adalah sebagai berikut:

3.3.1.1. Use Case Sistem yang sedang berjalan



Gambar 3.3 Use Case Sistem yang sedang berjalan

3.3.1.2 Activity Diagram Sistem yang sedang berjala



Gambar 3.4 Activity Diagram yang sedang berjalan

3.3.1.3 Sekenario Use Case Diagram Sistem yang sedang berjalan

Adapun sekenario use case sebagai berikut :

Tabel 3.1 Sekenario Use Case Mading

No	Aktor	Sistem
1	Pengunjung Mendatangi area informasi	
2		Memuat informasi kebun binatang bandung.
3	Mendapat informasi seputar kebun binatang bandung.	

Tabel 3.2 Sekenario Use Case Brosur

No	Aktor	Sistem
1	Pengunjung Mendatangi loket informasi	
2		Staff memberikan brosur informasi kepada pengunjung.
3	Mendapat brosur yang berisi informasi seputar kebun binatang bandung	

3.3.2. Evaluasi Sistem

Berdasarkan hasil analisis sistem yang sedang berjalan diatas terlihat sudah banyak media yang menginformasikan mengenai Kebun Binatang Bandung, namun penulis menemukan beberapa permasalahan yang perlu, dimana permasalahan-permasalahan tersebut adalah sebagai berikut:

1. Media mading, informasi hanya dapat menampilkan lokasi binatang-binatang dikarenakan media penyimpanan informasi sangatlah tebatas

dan informasi yang disampaikan tidak dapat di perbaharui secara *realtime*.

2. Media brosur juga demikian hanya menampilkan informasi mengenai tata letak binatang, karena media informasi yang terbatas serta membutuhkan biaya tambahan untuk percetakan dan informasi bersifat permanen tidak dapat diperbaharui terkecuali mencetak ulang brosur.
3. *Tour guide/*pemandu juga sangatlah terbatas sehingga tidak dapat melayani setiap pelanggan yang berkunjung.

Dari permasalahan-permasalah diatas maka solusi yang tepat untuk memecahkan masalah tersebut yaitu :

1. Membuat Sistem Informasi yang dapat diakses dimanapun dan kapanpun sehingga pengunjung tidak perlu lagi mencari Mading untuk mendapatkan informasi mengenai Kebun Binatang Bandung.
2. Membangun Sistem Informasi yang dapat mengolah data mengenai tataletak dan informasi seputar binatang yang portable, dapat dibawa kemana-mana tanpa memerlukan perangkat komputer.
3. Membangun sistem informasi yang memiliki interasi data, sehingga data/informasi dapat dengan mudah diperbaharui secara langsung / *realtime*.