



UNIVERSITAS INDONESIA

**IMPLEMENTASI SISTEM INFORMASI *BIOLOGICAL
DIVERSITY* UNIVERSITAS INDONESIA**

TESIS

**IRWAN KURNIA ANDRIANTO
0806424472**

**FAKULTAS TEKNIK
PROGRAM PASCA SARJANA
DEPOK
JANUARI 2011**



UNIVERSITAS INDONESIA

**IMPLEMENTASI SISTEM INFORMASI *BIOLOGICAL
DIVERSITY* UNIVERSITAS INDONESIA**

TESIS

Diajukan sebagai salah satu syarat untuk memperoleh gelar Magister Teknik

**IRWAN KURNIA ANDRIANTO
0806424472**

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
BIDANG KEKHUSUSAN MULTIMEDIA DAN JARINGAN INFORMASI
DEPOK
JANUARI 2011**

HALAMAN PERNYATAAN ORISINALITAS

Tesis ini adalah hasil karya saya sendiri,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.

Nama : Irwan Kurnia Andrianto

NPM : 0806424472

Tanda Tangan : 

Tanggal : 5 Januari 2011


HALAMAN PENGESAHAN


Tesis ini diajukan oleh :
Nama : Irwan Kurnia Andrianto
NPM : 0806424472
Program Studi : Multimedia dan Jaringan Informasi
Judul Tesis :

IMPLEMENTASI SISTEM INFORMASI *BIOLOGICAL DIVERSITY* UNIVERSITAS INDONESIA

Telah berhasil dipertahankan dihadapan Dewan penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Magister Teknik pada Program Studi Teknik Elektro, Kekhususan Multimedia dan Jaringan Informasi, Departemen Teknik Elektro, Fakultas Teknik, Universitas Indonesia.

DEWAN PENGUJI

Pembimbing : Prof. Dr. Ir. Riri Fitri Sari, MSc, MM (.....) 

Penguji : Prof. Dr. Ir. Bagio Budiardjo, MSc (.....) 

Penguji : Prof. Dr. Ir. Kalamullah Ramli, M.Eng (.....) 

Penguji : Dr. Ir. Anak Agung Putri Ratna, M.Eng (.....) 

Ditetapkan di : Depok

Tanggal : 5 Januari 2011

KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, saya dapat menyelesaikan tesis ini. Penulisan tesis ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Magister Teknik jurusan Multimedia dan Jaringan Informasi pada Fakultas Teknik Universitas Indonesia. Saya menyadari bahwa tanpa bantuan dan bimbingan dari berbagai pihak, dari masa seminar sampai pada penyusunan tesis ini, sangatlah sulit bagi saya untuk menyelesaikan tesis ini. Oleh karena itu, saya mengucapkan terima kasih kepada:

- (1) Prof. Dr. Ir. Riri Fitri Sari, MSc, MM, selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan saya dalam penyusunan tesis ini;
- (2) Dra. Mega Atria, Msi, selaku Kepala Laboratorium Taksonomi Tumbuhan Departemen Biologi Fakultas MIPA Universitas Indonesia yang telah memberi kesempatan untuk bekerja mencatat data-data tanaman pada Laboratorium Taksonomi Tumbuhan.

Semoga tesis ini membawa manfaat bagi pengembangan ilmu.

Depok, 5 Januari 2011

Penulis

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai civitas akademik Universitas Indonesia, saya yang bertanda tangan dibawah ini:

Nama : Irwan Kurnia Andrianto
NPM : 0806424472
Program Studi : Multimedia dan Jaringan Informasi
Departemen : Teknik Elektro
Fakultas : Teknik
Jenis Karya : Tesis

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul :


IMPLEMENTASI SISTEM INFORMASI *BIOLOGICAL DIVERSITY*
UNIVERSITAS INDONESIA

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tesis saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok
Pada tanggal : 5 Januari 2010

Yang menyatakan



(Irwan Kurnia Andrianto)

ABSTRAK

Nama : Irwan Kurnia Andrianto
Program Studi : Teknik Elektro
Judul : Implementasi Sistem Informasi *Biological Diversity* Universitas Indonesia

Sistem Informasi *Biological Diversity* Universitas Indonesia adalah suatu sistem untuk mensurvei seluruh tanaman/tumbuhan yang berada di Kampus Baru Universitas Indonesia di Depok. Proses pencatatan informasi dapat dilakukan dengan melibatkan mahasiswa baru Universitas Indonesia untuk membuat daftar tanaman yang ada diseluruh kawasan kampus yang luanya mencapai 320 hektar. Setiap mahasiswa mendapatkan tugas untuk mensurvei tanaman dengan luas 100 m². Pembagian letak koleksi untuk survei yang dilakukan oleh mahasiswa dapat dimasukkan ke sistem dan dilihat dari peta tanaman. Sistem dapat digunakan untuk menentukan koordinat X dan Y untuk posisi survei setiap mahasiswa. Dalam sistem informasi *biological diversity* ini menggunakan *Google Earth* untuk memvisualisasikan peta. Metodologi yang digunakan dalam pelaksanaan sistem informasi ini adalah Proses Model *Waterfall* dan menggunakan *Unified Modeling System* (UML) untuk pemodelannya. Untuk implementasi dilakukan dengan menggunakan *script language* PHP versi 5.2.9 + PEAR dan basis datanya menggunakan MySQL-phpMyAdmin versi 3.1.3.1. Sistem informasi *biological diversity* ini berjalan dengan baik (rata-rata skala 3 dari skala 4). Untuk mahasiswa berpendapat bahwa tampilan untuk koleksi, *family*, *genus*, nama tanaman, detail koleksi, peta tanaman menilai menilai sistem informasi baik (rata-rata skala 3 dari skala 4).

Kata kunci: *Biological Diversity*, Koleksi, Koordinat X dan Y, *Google Earth*, *Waterfall*, *Unified Modeling Language*, PHP, MySQL, *family*, *genus*.

ABSTRACT

Name : Irwan Kurnia Andrianto
Study Program : Electrical Engineering
Title : Implementation of Biological Diversity Information System of
University of Indonesia

Biological Diversity Information system of University of Indonesia is a system for surveying the whole plants in the New Campus of University of Indonesia in Depok. The process of information record can be done by involving the new students of University of Indonesia to create a list of plants in a particular area. For example, every student got a duty to survey the plant with an area of 100 m². The distribution of location assign to students can be seen done use the system and can be inputted the system and can be viewed using the map of plant. This system determines X and Y coordinates for each student. In this Biodiversity Information System we use Google Earth to visualize the map. The methodology used in the implementation of information system is the waterfall process model and use Unified Modeling Language (UML) for modeling. The system is implemented using the scripting language PHP version 5.2.9 + PEAR and MySQL-database using phpMyAdmin version 3.1.3.1.

The testing of the system by the users record the data collections of the plant based on the coordinates in the campus area generate some response. Of the common visitor response to the biological diversity of information systems that the average user rate is good enough (the average scale of 3 on the scale 4). For students argued that the display for the collection, family, genus, plants name, details of collections, maps crop information system to assess rate is good enough (the average scale of 3 on the scale 4).

Key words: Biological Diversity, Collection, X and Y coordinates, Google Earth, Waterfall, Unified Modeling Language (UML), PHP, MySQL, family, genus.

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERNYATAAN ORISINALITAS	ii
LEMBAR PENGESAHAN	iii
KATA PENGANTAR	iv
LEMBAR PERSETUJUAN PUBLIKASI	v
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR GAMBAR	xi
DAFTAR TABEL	xiii
1. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Perumusan Masalah	2
1.3 Tujuan Penelitian	2
1.4 Batasan Masalah	3
1.5 Metodologi Penelitian	3
1.6 Sistematika Penulisan	4
2. LANDASAN TEORI	6
2.1 Sejarah Internet	6
2.1.1 Pengertian Internet	6
2.1.2 Fasilitas Internet	7
2.2 Metodologi Pengembangan: <i>Waterfall</i>	7
2.3 <i>Unified Modelling Language</i> (UML)	8
2.3.1 Dasar-dasar UML	9
2.4 Pengertian PHP	11
2.4.1 Struktur Penulisan Script PHP	12
2.4.2 Variabel	13
2.4.3 Tipe Data	13
2.5 Pengertian MySQL	13
2.5.1 Mengenal Bahasa MySQL	14
2.6 <i>User Interface Design</i>	15
2.6.1 Mengatur/ <i>Organize</i>	15
2.6.1.1 Konsistensi	16
2.6.1.2 Tata Letak Layar	17
2.6.1.3 Hubungan/ <i>Relationship</i>	17
2.6.1.4 Navigasi	17
2.6.2 Hemat/ <i>Economize</i>	18
2.6.2.1 Kesederhanaan/ <i>Simplicity</i>	18
2.6.2.2 Kejelasan/ <i>Clarity</i>	18
2.6.2.3 Kekhususan/ <i>Distinctiveness</i>	18
2.6.2.4 Tekanan/ <i>Emphasis</i>	18
2.6.3 Komunikasi/ <i>Communicate</i>	19

3. PERANCANGAN SISTEM	20
3.1 Unified Modeling Language	20
3.1.1 <i>Use Case Diagram</i>	20
3.1.2 <i>Class Diagram</i>	21
3.1.3 <i>Object Diagram</i>	23
3.1.4 <i>Collaboration Diagram</i>	24
3.1.4.1 Administrator <i>Entry</i> Koordinat	24
3.1.4.2 Administrator <i>Entry</i> Mahasiswa dan Menset Kode Koordinat	24
3.1.4.3 Administrator <i>Entry</i> Family	25
3.1.4.4 Administrator <i>Entry</i> Genus berdasarkan Family ..	25
3.1.4.5 Administrator <i>Entry</i> Tanaman berdasarkan Genus	26
3.1.4.6 Mahasiswa <i>Entry</i> Koleksi berdasarkan Tanaman ..	26
3.1.4.7 Pengunjung Melihat Data Tanaman	27
3.1.4.8 Pengunjung Melihat Data Koleksi	27
3.1.5 <i>Sequence Diagram</i>	28
3.1.6 <i>State Diagram</i>	29
3.1.6.1 Administrator <i>Entry</i> Koordinat	29
3.1.6.2 Administrator <i>Entry</i> Mahasiswa	30
3.1.6.3 Administrator <i>Entry</i> Family	30
3.1.6.4 Administrator <i>Entry</i> Genus	31
3.1.6.5 Administrator <i>Entry</i> Tanaman	31
3.1.6.6 Mahasiswa <i>Entry</i> Koleksi	32
3.1.6.7 Pengunjung Cari Koleksi	33
3.1.7 <i>Activity Diagram</i>	33
3.1.7.1 Administrator Menambahkan Koordinat	33
3.1.7.2 Administrator Menambahkan Mahasiswa	34
3.1.7.3 Administrator Menambahkan Family	35
3.1.7.4 Administrator Menambahkan Genus	35
3.1.7.5 Administrator Menambahkan Tanaman	36
3.1.7.6 Mahasiswa Menambahkan Koleksi	37
3.1.8 <i>Deployment Diagram</i>	38
3.1.9 <i>Component Diagram</i>	38
3.1.9.1 Administrator	38
3.1.9.2 Mahasiswa	39
3.2 Perancangan Database Aplikasi	40
3.2.1 Membuat Tabel	40
3.3 Desain Tampilan Secara Umum	44

4. IMPLEMENTASI DAN ANALISA PROGRAM	47
4.1 Implementasi Program	47
4.1.1 Halaman Administrator	47
4.1.1.1 Halaman Login	47
4.1.1.2 Halaman Index Administrator	48
4.1.1.3 Halaman Form Koordinat	48
4.1.1.4 Halaman Form Mahasiswa	48
4.1.1.5 Halaman Form <i>Family</i>	49
4.1.1.6 Halaman Form <i>Genus</i>	49
4.1.1.7 Halaman Form Tanaman	49
4.1.2 Halaman Mahasiswa	49
4.1.2.1 Halaman Login	49
4.1.2.2 Halaman Index Mahasiswa	49
4.1.2.3 Halaman Tanaman	49
4.1.2.4 Halaman Koleksi	50
4.1.2.5 Halaman Peta Tanaman	51
4.1.2.6 Halaman Pencarian	52
4.1.2.7 Halaman Download	52
4.2 Pengujian Oleh Pengguna	53
4.3 Pengembangan Masa Depan	56
5. PENUTUP	57
DAFTAR REFERENSI	58
DAFTAR LAMPIRAN	
Lampiran 1. Survey Penggunaan Sistem Informasi <i>Biological Diversity</i> Universitas Indonesia	60

DAFTAR GAMBAR

Gambar 2.1	<i>Waterfall Model</i>	8
Gambar 2.2	Tampilan Tidak Teratur	16
Gambar 2.3	Tampilan Teratur	16
Gambar 2.4	Konsistensi Internal – Dialog Box	16
Gambar 2.5	Konsistensi Eksternal untuk <i>Text Tool Icons</i>	17
Gambar 2.6	Konsistensi <i>Real-world</i>	17
Gambar 2.7	Desain Navigasi Buruk	18
Gambar 2.8	Desain Navigasi Perbaikan	18
Gambar 3.1	<i>Use Case Diagram</i>	21
Gambar 3.2	<i>Class Diagram</i>	22
Gambar 3.3	<i>Object Diagram</i>	23
Gambar 3.4	<i>Collaboration Diagram</i> Administrator Entry Koordinat .	24
Gambar 3.5	<i>Collaboration Diagram</i> Administrator Entry Mahasiswa dan Menset Kode Koordinat	24
Gambar 3.6	<i>Collaboration Diagram</i> Administrator Entry Family	25
Gambar 3.7	<i>Collaboration Diagram</i> Administrator Entry Genus berdasarkan Family	25
Gambar 3.8	<i>Collaboration Diagram</i> Administrator Entry Tanaman berdasarkan Genus	26
Gambar 3.9	<i>Collaboration Diagram</i> Administrator Entry Koleksi berdasarkan Tanaman	26
Gambar 3.10	<i>Collaboration Diagram</i> Pengunjung Melihat Data Tanaman	27
Gambar 3.11	<i>Collaboration Diagram</i> Pengunjung Melihat Data Koleksi	27
Gambar 3.12	<i>Sequence Diagram</i>	28
Gambar 3.13	<i>State Diagram</i> Administrator Entry Koordinat	29
Gambar 3.14	<i>State Diagram</i> Administrator Entry Mahasiswa	30
Gambar 3.15	<i>State Diagram</i> Administrator Entry Family	30
Gambar 3.16	<i>State Diagram</i> Administrator Entry Genus	31
Gambar 3.17	<i>State Diagram</i> Administrator Entry Tanaman	32
Gambar 3.18	<i>State Diagram</i> Mahasiswa Entry Koleksi	32
Gambar 3.19	<i>State Diagram</i> Pengunjung Cari Koleksi	33
Gambar 3.20	<i>Activity Diagram</i> Administrator Menambahkan Koleksi .	34
Gambar 3.21	<i>Activity Diagram</i> Administrator Menambahkan Mahasiswa	34
Gambar 3.22	<i>Activity Diagram</i> Administrator Menambahkan Family ..	35
Gambar 3.23	<i>Activity Diagram</i> Administrator Menambahkan Genus ...	35

Gambar 3.24	<i>Activity Diagram</i> Administrator Menambahkan Tanaman	36
Gambar 3.25	<i>Activity Diagram</i> Mahasiswa Menambahkan Koleksi	37
Gambar 3.26	<i>Deployment Diagram</i>	38
Gambar 3.27	<i>Component Diagram</i> Administator	39
Gambar 3.28	<i>Component Diagram</i> Mahasiswa	40
Gambar 4.1	Halaman Login Administrator	47
Gambar 4.2	Halaman Index Administrator	48
Gambar 4.3	Halaman Detail Tanaman	50
Gambar 4.4	Halaman Detail Koleksi	50
Gambar 4.5	Halaman Peta Tanaman	51
Gambar 4.6	Halaman Peta FT	51
Gambar 4.7	Halaman Pencarian	52
Gambar 4.8	Halaman Laporan	52
Gambar 4.9	Laporan Koleksi Tanaman	53
Gambar 4.10	Grafik tanggapan terhadap tingkat familiar dengan bahasa pemrograman dan database	55
Gambar 4.11	Grafik tanggapan sistem informasi <i>biological diversity</i> secara umum	55
Gambar 4.12	Grafik tanggapan sistem informasi <i>biological diversity</i> dari mahasiswa	56

DAFTAR TABEL

Tabel 2.1	Konsepsi Dasar UML	9
Tabel 3.1	Tabel Admin	40
Tabel 3.2	Tabel Koordinat	41
Tabel 3.3	Tabel Mahasiswa	42
Tabel 3.4	Tabel <i>Family</i>	42
Tabel 3.5	Tabel <i>Genus</i>	43
Tabel 3.6	Tabel Tanaman	43
Tabel 3.7	Tabel Koleksi	44
Tabel 4.1	Data tanggapan terhadap kuesioner	53
Tabel 4.2	Hasil perhitungan terhadap tanggapan kuesioner	54



BAB 1

PENDAHULUAN

1.1. Latar Belakang

Saat ini status biodiversity dan hutan di Indonesia dengan 17.000 (tujuh belas ribu) pulau merupakan salah satu negara mega *biodiversity* yang menduduki peringkat pertama di dunia untuk jumlah mamalia, palem, kupu-kupu *swallowtail*, dan jenis burung nuri [1]. Kemudian, Indonesia merupakan pusat dari *diversity* spesies tumbuhan untuk sejumlah genus dan merupakan salah satu pusat dunia *diversity* spesies terumbu karang dan banyak kelompok terkait flora dan fauna.

Informasi *biodiversity* sangat penting untuk berbagai keperluan ilmiah, pendidikan dan pemerintahan, dan penting untuk pengambilan keputusan. Sebagian besar informasi *biodiversity* tidak tersedia atau tidak dapat diakses. Perkembangan terkini dalam teknologi informasi dan komunikasi yang memungkinkan pengalaman baru dalam analisis, integrasi dan visualisasi [2].

Ahli biologi mengklarifikasi semua organisma hidup ke dalam jenis. Jenis diidentifikasi dengan memanfaatkan penampakan dan strukturnya, hal yang nampaknya sederhana namun memerlukan waktu bertahun-tahun untuk mempelajarinya. Seorang ahli Biologi bangsa Swedia, Carl Linnaeus, menciptakan metoda modern untuk penamaan jenis ini pada abad ke-18 [3]. Metoda Linnaeus ini memberikan dua kata bahasa Latin untuk setiap jenis. Nama pertama mengacu pada kelompok jenis yang berkerabat dekat yang disebut marga, sedangkan nama kedua membedakan jenis yang satu dengan jenis lainnya dalam marga yang sama.

Lemahnya database flora fauna Indonesia menyebabkan tingginya potensi kehilangan biodiversitas yang sangat bernilai baik secara ekonomi maupun ekologis, bahkan seringkali tanpa sempat kita tahu persis spesies apa yang telah hilang tersebut. Saat ini peran pendokumentasian dan preservasi biodiversitas di Indonesia lebih banyak dilakukan oleh museum Botani dan Zoologi yang dikelola oleh Lembaga Penelitian dan Pengembangan Biologi, LIPI [4]. Banyak pemangku kepentingan (*stake holder*) di Indonesia yang sangat potensial untuk ikut berperan dalam pendokumentasian ini, antara lain: libang di departemen teknis, perguruan

tinggi, maupun LSM diseluruh Indonesia. Hanya saja peran serta *stake holder* tersebut terkendala oleh lemahnya kemampuan SDM maupun kurang memadainya sistem pendokumentasian.

Secara geografis, posisi kampus Universitas Indonesia berada di dua area berjauhan, kampus Salemba dan kampus Depok. Mayoritas fakultas berada di Depok dengan luas lahan mencapai 320 hektar dengan atmosfer *green campus* karena hanya 25% lahan digunakan digunakan sebagai sarana akademik, riset dan kemahasiswaan [5]. Lebih kurang 75% wilayah Universitas Indonesia bisa dikatakan adalah area hijau berwujud hutan kota dimana di dalamnya terdapat 8 (delapan) danau alam. Kampus Universitas Indonesia Depok memberi suasana yang tenang dan asri untuk belajar dan *echo tourism*.

Hal ini yang mendorong penulis ingin mengembangkan atau membuat suatu sistem informasi *biodiversity* untuk mensurvei semua tanaman/tumbuhan yang berada di Kampus Baru Universitas Indonesia di Depok.

1.2. Perumusan Masalah

Perumusan masalah antara lain:

1. Sebagian besar informasi *biodiversity* tidak tersedia atau tidak dapat diakses.
2. Bagaimana cara pencatatan tanaman berdasarkan jenis, nama tanaman dan nama latin.
3. Bagaimana membangun Sistem Informasi *Biodiversity* agar dapat dipakai sebagai website yang *user-friendly*.

1.3. Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk membuat sebuah sistem informasi yang berfungsi mencatat dan membantu survei data tanaman berdasarkan *family*, *genus*, *spesies*/nama latin, dan nama tanaman yang berada di Kampus Baru Universitas Indonesia. Pembagian letak koleksi untuk survei yang dilakukan oleh mahasiswa dapat dilihat dari peta tanaman. Sistem ini menentukan dengan koordinat X dan Y untuk setiap mahasiswa. Dalam membangun Sistem Informasi *Biodiversity* dilakukan dengan menggunakan *Unified Modeling Language* (UML) untuk pemodelannya dan implementasi dilakukan dengan menggunakan *script*

Universitas Indonesia

language PHP dan untuk basis datanya menggunakan MySQL agar dapat dipakai sebagai website yang *user-friendly*. Serta implementasi dan pengujian program terhadap 20 mahasiswa yang menguji coba sistem ini.

1.4. Batasan Masalah

Di dalam pembahasan ini, ada beberapa batasan masalah, antara lain:

1. Database tanaman dicatat dari Laboratorium Taksonomi Tumbuhan FMIPA Universitas Indonesia.
2. Tanaman yang didata yaitu golongan herba dan perdu.
 - a. Herba = batang basah, berukuran kecil, dan berumur pendek
 - b. Perdu = Tinggi tanaman sampai dengan 5 (lima) meter, dan berkayu.
3. Foto atau gambar tanaman yang akan diunggah ke sistem besarnya tidak lebih dari 4 Mb.
4. Dalam sistem Informasi *Biodiversity* menggunakan *Google Earth* untuk memvisualisasikan peta.

1.5. Metodologi Penelitian

Pada penelitian ini penulis akan melakukan penelitian dengan melakukan beberapa tahapan diantaranya adalah:

1. Studi Literatur
Mengumpulkan dan mempelajari buku, artikel, jurnal dan referensi lainnya yang berkaitan dengan sistem Informasi dan taksonomi tumbuhan.
2. Pengumpulan Data
Mengumpulkan data-data yang berkenaan dengan data-data tanaman yang berada di Laboratorium Taksonomi Tumbuhan FMIPA Universitas Indonesia.
3. Perancangan Sistem
Pada tahap ini dilakukan skenario perancangan sistem informasi yaitu:
 - a. *Unified Modeling Language* (UML)
 - b. Perancangan *Database*
 - c. Desain Tampilan Secara Umum
4. Implementasi Program

Universitas Indonesia

Pada tahap ini dilakukan perancangan halaman *website* dan pengujian sistem.

5. Analisa Sistem

Setelah dilakukan implementasi, pada tahap analisa sistem ini dilakukan survey untuk penggunaan sistem Informasi terhadap mahasiswa.

6. Penarikan Kesimpulan

Berdasarkan hasil yang diharapkan dari Sistem Informasi Biodiversity ini.

7. Penulisan Buku Laporan

Penulisan laporan ini mengacu pada pedoman penulisan ilmiah dalam hal ini penulisan tesis yang bentuk bakunya telah diatur oleh pihak Universitas Indonesia.

1.6. Sistematika Penulisan

Sistematika pembahasan pada penelitian ini seperti yang tercantum berikut ini:

BAB I : PENDAHULUAN

Bab ini menjelaskan tentang latar belakang, perumusan masalah, batasan masalah, tujuan penulisan, metodologi penulisan, dan sistematika penulisan.

BAB II : LATAR BELAKANG

Bab ini menjelaskan tentang landasan teori yang akan menunjang proses perancangan sistem khususnya yang berkaitan dengan sejarah Internet, model waterfall, *Unified Modelling Language* (UML), pengertian dan struktur penulisan PHP, pengertian MySQL.

BAB III : PERANCANGAN SISTEM

Bab ini menjelaskan tentang skenario perancangan pembuatan sistem informasi dengan beberapa tahap antara lain: *Unified Modeling Language* (UML), perancangan database dan tampilan desain secara umum.

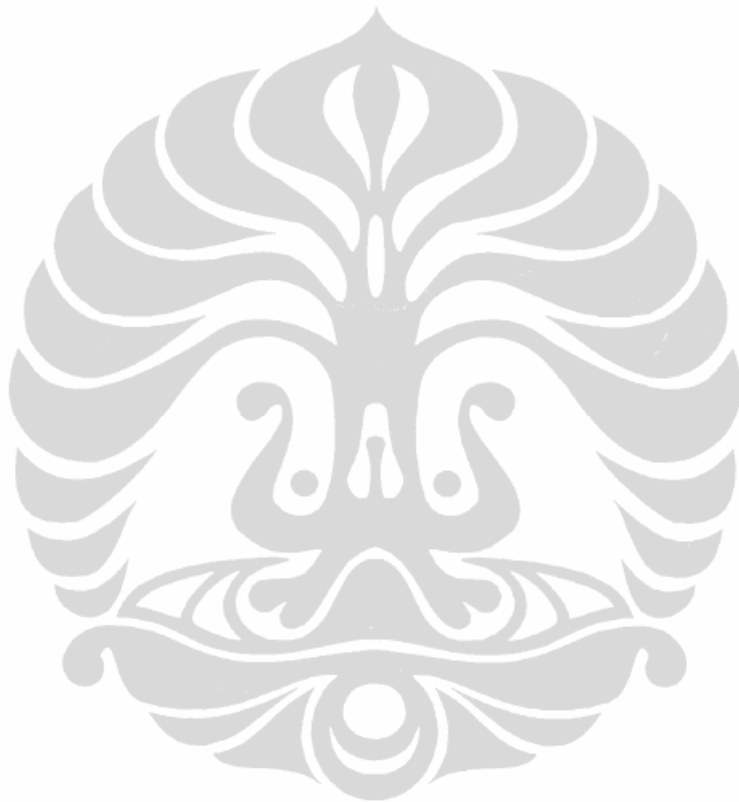
BAB IV : IMPLEMENTASI DAN ANALISA PROGRAM

Bab ini menjelaskan implementasi dan analisa program terhadap survey untuk penggunaan sistem Informasi oleh pengguna.

Universitas Indonesia

BAB V : PENUTUP

Dalam bab ini berisi kesimpulan yang diharapkan dari Implementasi Sistem Informasi Biodiversity.



BAB 2

LANDASAN TEORI

2.1. Sejarah Internet

Sebelum adanya Internet, ARPAnet (US Defense Advanced Research Projects Agency) atau Departemen Pertahanan Amerika pada tahun 1969 membuat jaringan komputer yang tersebar untuk menghindarkan terjadinya informasi terpusat, yang apabila terjadi perang dapat mudah dihancurkan. Jadi bila satu bagian dari sambungan *network* terganggu dari serangan musuh, jalur yang melalui sambungan itu secara otomatis dipindahkan ke sambungan lainnya [6]. Setelah itu Internet digunakan oleh kalangan akademis (UCLA) untuk keperluan penelitian dan pengembangan teknologi. Dan baru pada awal tahun 1990 Pemerintah Amerika Serikat memberikan ijin untuk penggunaan komersial.

2.1.1. Pengertian Internet

Internet berasal dari kata *Interconnection Networking* yang mempunyai arti hubungan komputer dengan berbagai tipe yang membentuk sistem jaringan yang mencakup seluruh dunia (jaringan komputer global) dengan melalui jalur telekomunikasi seperti telepon, radio link, satelit dan lainnya [7].

Dalam mengatur integrasi dan komunikasi jaringan komputer ini digunakan protokol yaitu TCP/IP. *Transmission Control Protocol* (TCP) bertugas memastikan bahwa semua hubungan bekerja dengan benar, sedangkan *Internet Protocol* (IP) yang mentransmisikan data dari satu komputer ke komputer lain.

TCP/IP secara umum berfungsi memilih rute terbaik transmisi data, memilih rute alternatif jika suatu rute tidak dapat digunakan, mengatur dan mengirimkan paket-paket pengiriman data.

Untuk dapat ikut serta menggunakan fasilitas Internet, pengguna harus berlangganan ke salah satu *Internet Service Provider* (ISP) yang ada di kota tempat tinggalnya. ISP ini biasanya disebut penyelenggara jasa Internet misalnya menggunakan fasilitas dari Telkom yakni Telkomnet Instan.

Dengan memanfaatkan Internet, pemakaian komputer di seluruh dunia dimungkinkan untuk saling berkomunikasi dan pemakaian bersama informasi

dengan cara saling kirim e-mail, menghubungkan ke komputer lain, mengirim dan menerima file, membahas topik tertentu pada *newsgroup* dan lain-lain.

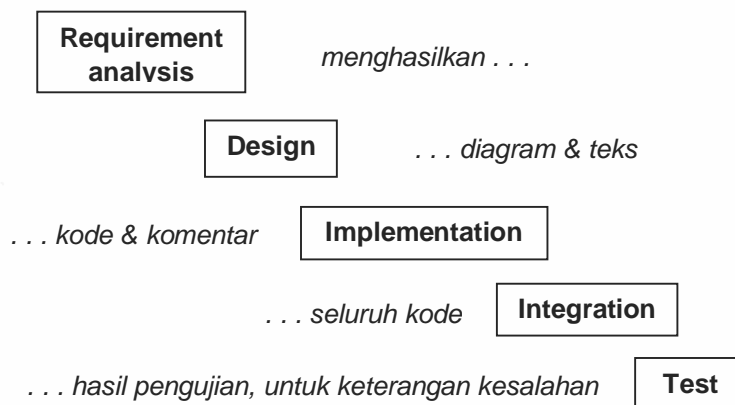
2.1.2. Fasilitas Internet

Fasilitas-fasilitas yang dapat Anda manfaatkan dengan menggunakan internet, diantaranya:

- **Web**, adalah fasilitas *hypertext* untuk menampilkan data berupa teks, gambar, bunyi, animasi dan data multimedia lainnya, yang diantara data tersebut saling berhubungan satu sama lain. Untuk memudahkan Anda membaca data dan informasi tersebut Anda dapat memprgunakan web browser seperti Internet Explorer (IE) ataupun Mozilla Firefox.
- **E-mail (*Electronic Mail*)**, dengan fasilitas ini Anda dapat mengirim dan menerima surat elektrik (e-mail) pada/dari pemakai komputer lain yang terhubung di internet, dan dapat menyertakan file sebagai lampiran (*attachement*).
- **File Transfer Protocol (FTP)**, fasilitas ini digunakan untuk menghubungkan ke server komputer tertentu dan bila perlu menyalin (*download*) file yang Anda butuhkan dari server tersebut dan menyimpannya di komputer Anda.

2.2. Metodologi Pengembangan: *Waterfall*

Proses pengembangan perangkat lunak model klasik adalah model *waterfall* (air terjun). Ini merupakan urutan kegiatan atau fase yang terdiri dari persyaratan analisis (*requirements analysis*), perancangan (*design*), implementasi (*implementation*), dan integrasi (*integration*). Fase tersebut dapat dilihat pada gambar 2.1 dibawah ini [8]:



Gambar 2.1. *Waterfall* Model [8]

- Persyaratan analisis (*requirements analysis*) terdiri dari pengumpulan persyaratan untuk produk, dan outputnya biasanya teks.
- Perancangan (*design*) menggambarkan bagaimana produk akan terstruktur secara internal, dan biasanya digambarkan oleh diagram dan teks.
- Implementasi (*Implementation*) berarti pemrograman. Hasil dari tahap ini adalah kode di tingkat manapun, termasuk yang dihasilkan oleh sistem high-level drop-and-drag, bahasa generasi keempat, dan seterusnya.
- Integrasi (*Integration*) adalah proses perakitan bagian-bagian untuk menyelesaikan produk.

2.3. *Unified Modelling Language (UML)*

Unified Modeling Language (UML) adalah sebuah “bahasa” yang telah menjadi standar dalam industri untuk visualisasi, merancang, dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem.

Seperti bahasa-bahasa lainnya, UML mendefinisikan notasi dan *syntax*/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan sintaks UML mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Notasi UML terutama diturunkan dari 3 notasi yang telah ada sebelumnya: Grady Booch OOD (*Object-Oriented Design*), Jim Rumbaugh OMT (*Object Modeling Technique*), dan Ivar Jocabson OOSE (*Object-Oriented Software Engineering*).

2.3.1. Dasar-dasar UML

Berikut adalah konsepsi dasar-dasar UML [9]:

Tabel 2.1. Konsepsi Dasar UML [9]

Area Utama	Tampilan	Diagram	Bagan Utama
struktural	tampilan statik	<i>class diagram</i>	kelas, asosiasi, penyamarataan, gabungan, realisasi, antarmuka
	tampilan use case	<i>use case diagram</i>	<i>use case</i> , aktor, gabungan, menyampaikan, memasukkan, penyamarataan <i>use case</i>
	tampilan implementasi	<i>component diagram</i>	komponen, antarmuka, ketergantungan, realisasi
	tampilan menyebarkan	<i>deployment diagram</i>	<i>node</i> , komponen, gabungan, lokasi
dinamik	tampilan keadaan mesin	<i>statechart diagram</i>	keadaan, kejadian, transisi, tindakan
	tampilan aktifitas	<i>activity diagram</i>	keadaan, aktifitas, penyelesaian transisi, percabangan, penggabungan
	tampilan interaksi	<i>sequence diagram</i>	interaksi, objek, pesan, aktifitas
		<i>collaboration diagram</i>	kolaborasi, interaksi, peran kolaborasi, pesan
manajemen model	tampilan manajemen model	<i>class diagram</i>	paket, subsistem, model
kemungkinan diperpanjang	semua	semua	pembatas, klise, memasang nilai

Seperti juga tercantum pada gambar diatas UML mendefinisikan diagram-diagram sebagai berikut [9]:

- **Use Case Diagram**

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* mempresentasikan sebuah interaksi antara aktor dengan sistem. *Use case diagram* dapat sangat membantu bila

kita sedang menyusun *requirement* sebuah sistem, mengkomunikasikan rancangan dengan klien, dan merancang *test case* untuk semua *feature* yang ada pada sistem.

- **Class Diagram**

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).

- **Statechart Diagram**

Statechart diagram menggambarkan transisi dan perubahan keadaan (dari satu *state* ke *state* lainnya) suatu objek pada sistem sebagai akibat dari *stimuli* yang diterima. Pada umumnya *statechart diagram* menggambarkan *class* tertentu (satu *class* dapat memiliki lebih dari satu *statechart diagram*).

- **Activity Diagram**

Activity diagram menggambarkan berbagai alir aktifitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *state* sebelumnya (*internal processing*).

- **Sequence Diagram**

Sequence diagram menggambarkan interaksi antar objek di dalam dan 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 skenario atau rangkain langkah-langkah yang dilakukan sebagai tanggapan dari sebuah *event* untuk menghasilkan *output* tertentu.

- **Collaboration Diagram**

Collaboration diagram juga menggambarkan interaksi antar objek seperti *sequence diagram*, tetapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian *message*. Setiap *message* memiliki *sequence number*, di mana *message* dari level tertinggi memiliki nomor 1. *Message* dari level yang sama memiliki prefiks yang sama.

- **Component Diagram**

Component diagram menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan (*dependency*). 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.

- **Deployment Diagram**

Deployment/physical diagram menggambarkan terlalu rinci 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 fisik.

2.4. Pengertain PHP

PHP adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membentuk web dinamis, yang merupakan singkatan dari *PHP (Hypertext Preprocessor)*. Ia merupakan bahasa berbentuk skrip yang ditempatkan dalam server dan proses di server. Hasilnya yang dikirimkan ke klien, tempat pemakai menggunakan *browser*.

Maksud dari *server-side scripting* adalah sintaks dan perintah-perintah yang diberikan akan sepenuhnya dijalankan di server tetapi disertakan pada

dokumen HTML. Pembuatan web ini merupakan kombinasi antara PHP sendiri sebagai bahasa pemrograman dan HTML sebagai pembangun halaman web.

Ketika seorang pengguna internet akan membuka suatu situs yang menggunakan fasilitas *server-side scripting* PHP, maka terlebih dahulu server yang bersangkutan akan memproses semua perintah PHP di server lalu mengirimkan hasilnya dalam format HTML ke web browser.

PHP merupakan software yang *Open Source* (gratis) dan mampu lintas *platform*, yaitu dapat digunakan dengan sistem operasi dan web server apapun. PHP mampu berjalan di Windows dan beberapa versi Linux. PHP dapat juga dibangun sebagai modul pada web server Xampp dan sebagai *binary* yang berjalan sebagai CGI.

PHP dapat mengirim HTTP *header*, dapat mengeset *cookies*, mengatur *authentication* dan *redirect users*. PHP menawarkan koneksitas yang baik dengan beberapa basis data, antara lain Oracle, mSQL, MySQL, Sybase dan tak terkecuali semua database ber-*interface* ODBC. Juga dapat berintegrasi dengan beberapa *library* eksternal yang dapat melakukan segalanya mulai dari membuat dokumen PDF hingga mem-*parse* XML.

2.4.1. Struktur Penulisan Script PHP

Ada beberapa cara untuk mulai menuliskan script PHP, yaitu [6]:

1. `<?php`
 Script PHP anda
 `?>`
2. `<?`
 Script PHP anda
 `?>`
3. `<script language="php">`
 Script PHP anda
 `</script>`
4. `<%`
 Script PHP anda
 `%>`

Cara pertama merupakan format yang dianjurkan tetapi mungkin cara kedua lebih sering digunakan karena lebih ringkas. Cara yang ketiga digunakan

untuk mengantisipasi editor dan web server *tag* tidak dapat menerima kedua cara sebelumnya, selain itu juga bisa menggunakan cara penulisan ASP.

Kode PHP diawali dengan `<?php....?>`. Pasangan kedua kode inilah yang berfungsi sebagai tag kode PHP. Berdasarkan *tag* inilah, pihak server dapat memahami kode PHP dan kemudian memprosesnya, dan hasilnya dikirimkan ke browser.

2.4.2. Variabel

Variabel adalah sebuah tempat di memori untuk menyimpan data yang nilainya dapat berubah-ubah. Namun tidak seperti pada bahasa pemrograman lain yang mengharuskan kita untuk mendeklarasikan variabel terlebih dahulu, variabel dalam PHP tidak harus dideklarasikan sebelum variabel tersebut digunakan. Variabel diwakili oleh kata tertentu dengan aturan penulisan sebagai berikut:[7]

1. Variabel dimulai dengan tanda dollar (\$).
2. Harus dimulai dengan huruf atau *underscore* (_).
3. Tidak boleh menggunakan tanda baca.
4. *Case sensitive* atau huruf besar dan huruf kecil dibedakan.

2.4.3. Tipe Data

PHP mengenal beberapa macam tipe data, antara lain *integer*, *floating point* dan *string*. Floating point lebih dikenal dengan nama *double* atau *decimal*. Penulisan *string* selalu diawali dengan tanda petik ganda (") atau tanda petik tunggal (').

Tabel 2.2 Contoh penulisan tipe data [10]

Tipe Data	Contoh	Keterangan
Integer	<code>\$jumlah = 10;</code> <code>\$nilai = -5;</code>	Bilangan Bulat
Doubel	<code>\$ekor = 90.00;</code> <code>\$bunga = 12.50;</code>	Bilangan Real
String	<code>\$kota = "Yogyakarta"</code>	Karakter, kalimat

2.5. Pengertian MySQL

MySQL adalah database multiuser yang menggunakan bahasa *Structured Query Language* (SQL). MySQL dalam operasi client-server melibatkan *server*

daemon MySQL diisi server dan berbagai macam program serta *library* yang berjalan disisi *client*.

SQL adalah bahasa standar yang digunakan untuk mengakses database server. Dengan menggunakan SQL, proses akses database menjadi lebih *user-friendly* dibandingkan dengan menggunakan perintah-perintah pemrograman.

2.5.1. Mengenal Bahasa SQL

Dalam pengenalan bahasa SQL terdapat beberapa cara:

a. Membuat *Database*

Sebelum kita membuat tabel-tabel ataupun memasukkan data, kita terlebih dahulu kita harus membuat databasenya. Untuk membuat suatu database, gunakan perintah berikut ini:

```
CREATE DATABASE nama_database;
```

b. Membuka *Database*

Setelah membuat suatu database, bila ingin membuat suatu tabel atau melakukan operasi pada suatu database, terlebih dahulu kita harus membuka database tersebut. Untuk membuka suatu database, gunakan perintah berikut ini:

```
USE nama_database;
```

c. Membuat Tabel

Setelah database terbentuk dan juga telah terbuka, untuk membuat tabel yang akan menyimpan data yang akan kita butuhkan dapat dilakukan dengan perintah berikut ini:

```
CREATE TABLE nama_tabel  
(  
nama_field1tipe_data1,  
nama_field2tipe_data2,  
..... );
```

d. Memasukkan Data ke Tabel

Untuk memasukkan data pada tabel yang telah dibuat kita menggunakan perintah INSERT. Sintaks penulisannya adalah sebagai berikut:

```
INSERT INTO nama_tabel (field1,field2,...)  
values (nilai_field1,nilai_field2,...);
```

e. Menampilkan Data dari Tabel

Perintah ini digunakan untuk melihat data pada field-field tertentu dari suatu tabel. Sintaks penulisannya adalah sebagai berikut:

```
SELECT (field1,field2,...) FROM nama_tabel;
```

f. Menghapus Data dalam Tabel

Dalam pengelolaan suatu data tentunya juga diperlukan suatu perintah untuk menghapus data. Dalam SQL digunakan perintah DELETE. Sintaks penulisannya adalah sebagai berikut:

```
DELETE FROM nama_tabel;
```

g. Mengubah Data dalam Tabel

Selain menghapus data, kadangkala perlu pula mengubah suatu data. Disinilah peran dari perintah UPDATE yang berfungsi untuk memodifikasi nilai kolom (field) dari suatu record. Sintaks penulisannya adalah sebagai berikut:

```
UPDATE nama_tabel SET  
nama_field1=nilai_baru1,  
nama_field2=nilai_baru2,  
....  
WHERE kriteria;
```

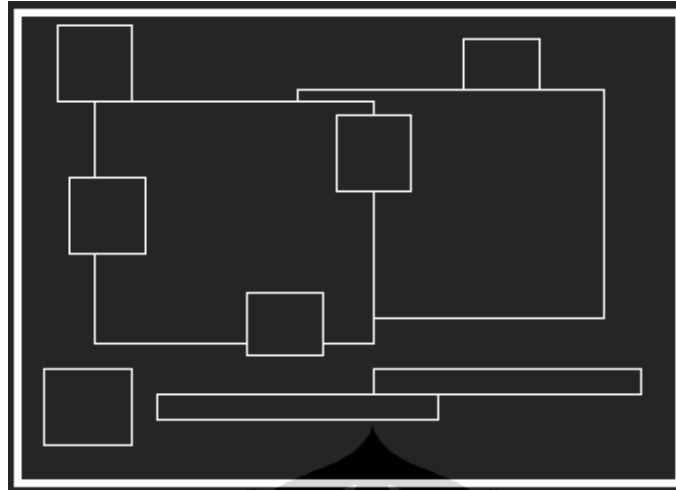
2.6. User Interface Design

Ada 3 (tiga) prinsip dasar yang terlibat dalam penggunaan bahasa pemrograman, yaitu [20]:

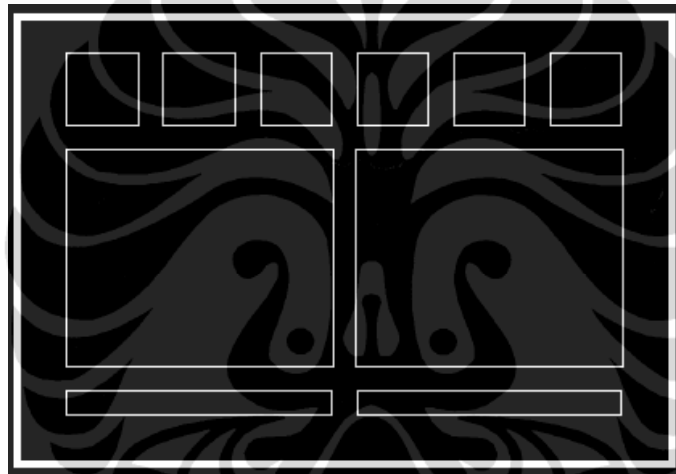
1. Mengatur/*Organize*: menyediakan pengguna dengan struktur konseptual yang jelas dan konsisten.
2. Hemat/*Economize*: melakukan yang paling sedikit isyarat.
3. Komunikasi/*Communicate*: presentasi sesuai dengan kemampuan pengguna.

2.6.1. Mengatur/*Organize*

Konsistensi, tata letak layar, hubungan, dan navigasi merupakan konsep penting dari mengatur tampilan. Lihat Gambar 2.2. dan Gambar 2.3. dibawah ini.



Gambar 2.2. Tampilan Tidak Teratur

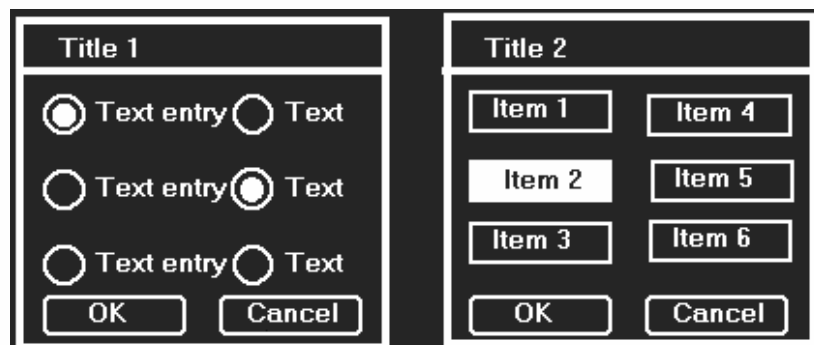


Gambar 2.3. Tampilan Teratur

2.6.1.1. Konsistensi

Ada empat pandangan konsistensi: konsistensi internal, konsistensi eksternal, konsistensi *real-world*, dan kapan tidak konsistensi.

Poin pertama, konsistensi internal menyatakan konvensi yang sama dan aturan harus diterapkan pada seluruh elemen GUI. Lihat Gambar 2.4. dibawah ini.



Gambar 2.4. Konsistensi Internal – Dialog Box

Poin kedua, konsistensi eksternal menyatakan *platform* yang ada dan konvensi tampilan yang harus diikuti di seluruh user interface. Lihat Gambar 2.5. dibawah ini.



Gambar 2.5. Konsistensi Eksternal untuk *Text Tool Icons*

Konsistensi *Real-world* berarti konvensi harus dibuat konsisten dengan *real-world experiences*, pengamatan (*observations*) dan persepsi pengguna. Lihat Gambar 2.6. dibawah ini.



Gambar 2.6. Konsistensi *Real-world*

Poin terakhir, inovasi berhubungan dengan kapan tidak konsistensi. Menyimpang dari konvensi yang ada hanya boleh dilakukan jika hal itu memberikan manfaat yang jelas kepada pengguna.

2.6.1.2.Tata Letak Layar

Tiga cara untuk merancang tata letak tampilan ruang, yaitu: mrengunakan struktur *grid*, standarisasi tata letak layar, dan elemen kelompok terkait.

Struktur *grid* dapat membantu menentukan menu, kotak dialog atau panel kontrol. Umumnya 7 menu merupakan adalah jumlah maksimum untuk tampilan utama horizontal dan vertikal. Ini akan membantu membuat layar teratur dan mudah dimengerti.

2.6.1.3.Hubungan/Relationship

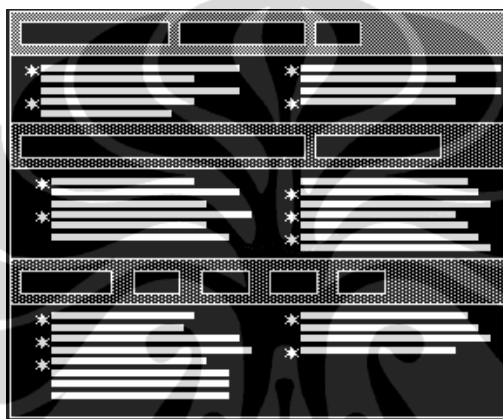
Menghubungkan item terkait dan tidak dikaitkan item terkait dapat membantu tampilan terorganisasi.

2.6.1.4.Navigasi

Ada 3 (tiga) teknik navigasi yang penting, yaitu: memberikan fokus awal atas perhatian pengguna, perhatian langsung sangat penting, sekunder atau item sekelilingnya. Lihat Gambar 2.7. dan Gambar 2.8. dibawah ini.



Gambar 2.7. Desain Navigasi Buruk



Gambar 2.8. Desain Navigasi Perbaikan

2.6.2. Hemat/*Economize*

Empat poin utama yang harus dipertimbangkan: kesederhanaan (*simplicity*), kejelasan (*clarity*), kekhususan (*distinctiveness*), dan tekanan (*emphasis*).

2.6.2.1. Kesederhanaan/*Simplicity*

Kesederhanaan hanya mencakup unsur-unsur yang penting untuk komunikasi.

2.6.2.2. Kejelasan/*Clarity*

Semua komponen harus dirancang sehingga maknanya tidak ambigu.

2.6.2.3. Kekhususan/*Distinctiveness*

Sifat penting dari elemen-elemen yang diperlukan harus dibedakan.

2.6.2.4. Tekanan/*Emphasis*

Elemen-elemen yang paling penting harus mudah dirasakan. Unsur tidak penting yang harus ditekankan dan kekacauan harus diminimalkan sehingga tidak menyembunyikan informasi yang penting.

2.6.3. Komunikasi/*Communicate*

GUI harus tetap menjaga keseimbangan, *readability*, tipografi/*typography*, simbolisme, tampilan ganda dan warna atau tekstur untuk berkomunikasi dengan sukses.



BAB 3

PERANCANGAN SISTEM

Dalam pembuatan aplikasi tentang Sistem Informasi *Biological Diversity* Universitas Indonesia, maka pada bab ini penulis akan menjelaskan beberapa tahap dalam perancangan dan pembuatan aplikasi. Tahap-tahap ini terdiri dari 3 (tiga) tahap yang dikerjakan secara berurutan, yaitu:

- *Unified Modeling Language* (UML),
- Perancangan database,
- Desain tampilan secara umum.

Untuk dapat mengatur, mengendalikan dan mengelola *website* Sistem Informasi *Biological Diversity* Universitas Indonesia ini, dibuatlah suatu kumpulan halaman web yang berfungsi sebagai ruang kerja bagi *sysadmin*, dimana halaman web ini hanya dapat diakses oleh *user admin*.

Dalam penyajian sebuah *website* di *Internet* diperlukan perancangan yang tepat, *user-friendly* (mudah dalam penggunaan), dan berbobot. Syarat utama sebuah situs yang baik adalah situs yang mengerti kebutuhan dan keinginan para user. *Loading time* (waktu tampil) juga harus diperhatikan secara seksama.

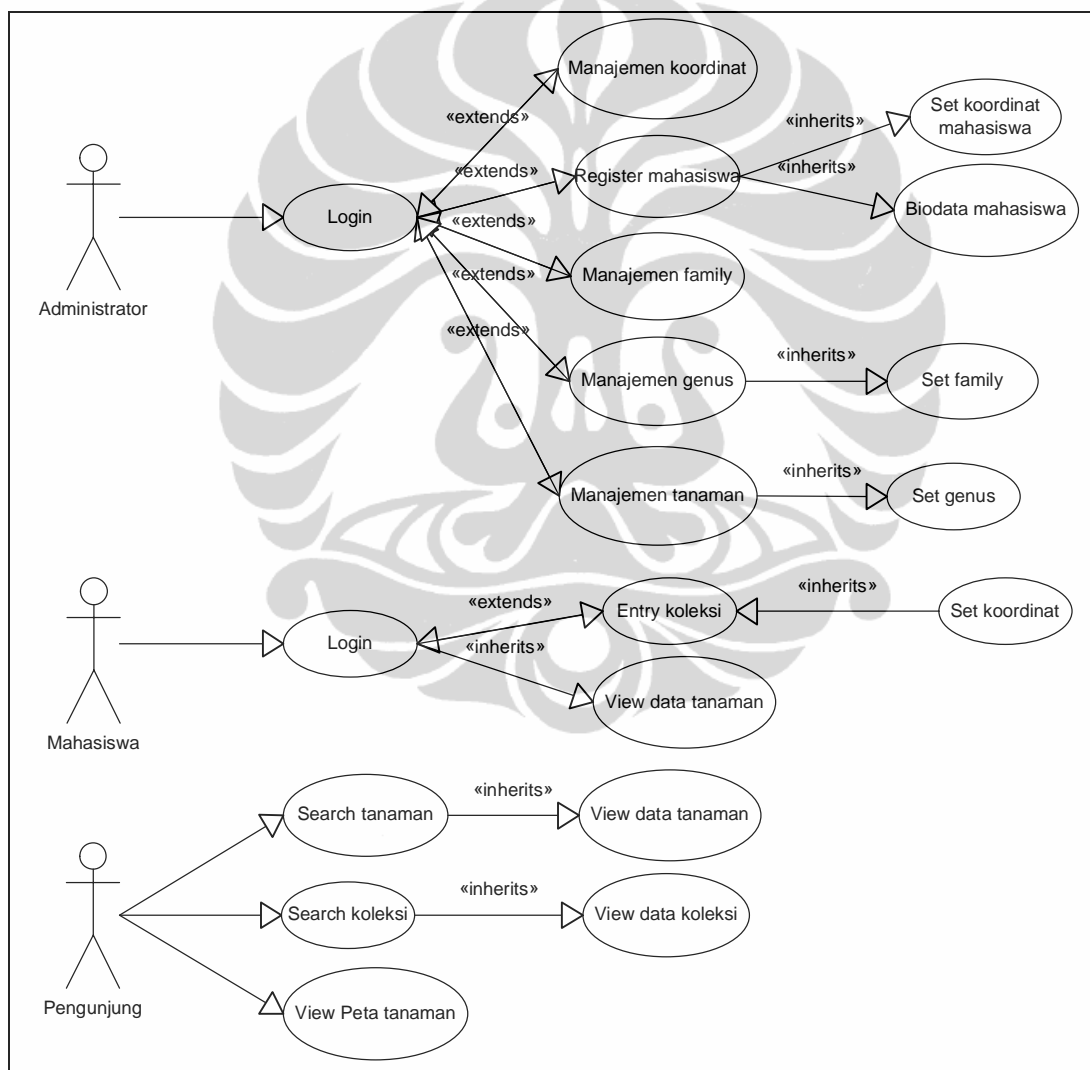
3.1. *Unified Modeling Language*

3.1.1. *Use Case Diagram*

Pada Gambar 3.1. terdapat *use case diagram* dengan keterangan, sebagai berikut:

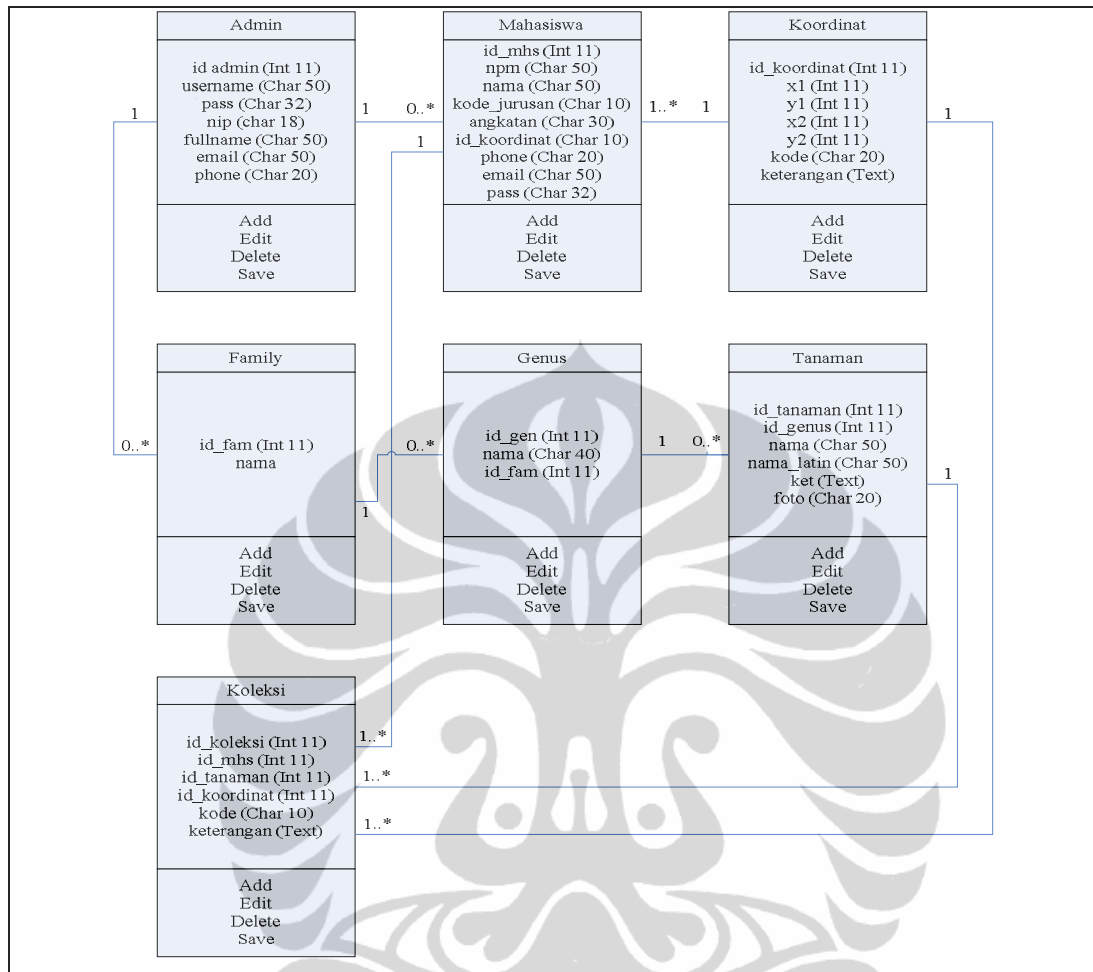
1. Aktor “Administrator” pertama kali memasukkan user “login”.
 - a. “Manajemen koordinat” berdasarkan satu mahasiswa satu koordinat.
 - b. “Register mahasiswa” berdasarkan biodata mahasiswa dengan memberikan sebuah koordinat.
 - c. “Manajemen family”, administrator mendaftarkan family.
 - d. “Manajemen genus”, administrator mendaftarkan genus dengan memilih family yang telah didaftarkan.
 - e. “Manajemen tanaman”, administrator mendaftarkan tanaman dengan memilih family dan genus terlebih dahulu.

2. Aktor “Mahasiswa” memasukkan login terlebih dahulu.
 - a. “View data tanaman”. Melihat data-data tanaman yang telah tercatat di sistem.
 - b. “Entry tanaman”. Mengentry koleksi yang didapatkan mahasiswa, dengan memilih family, genus dan koordinat yang telah ditentukan oleh administrator.
3. Aktor “Pengunjung” hanya dapat melihat daftar tanaman, daftar koleksi, pencarian nama atau nama latin tanaman dan peta tanaman.



Gambar 3.1. Use Case Diagram

3.1.2. Class Diagram



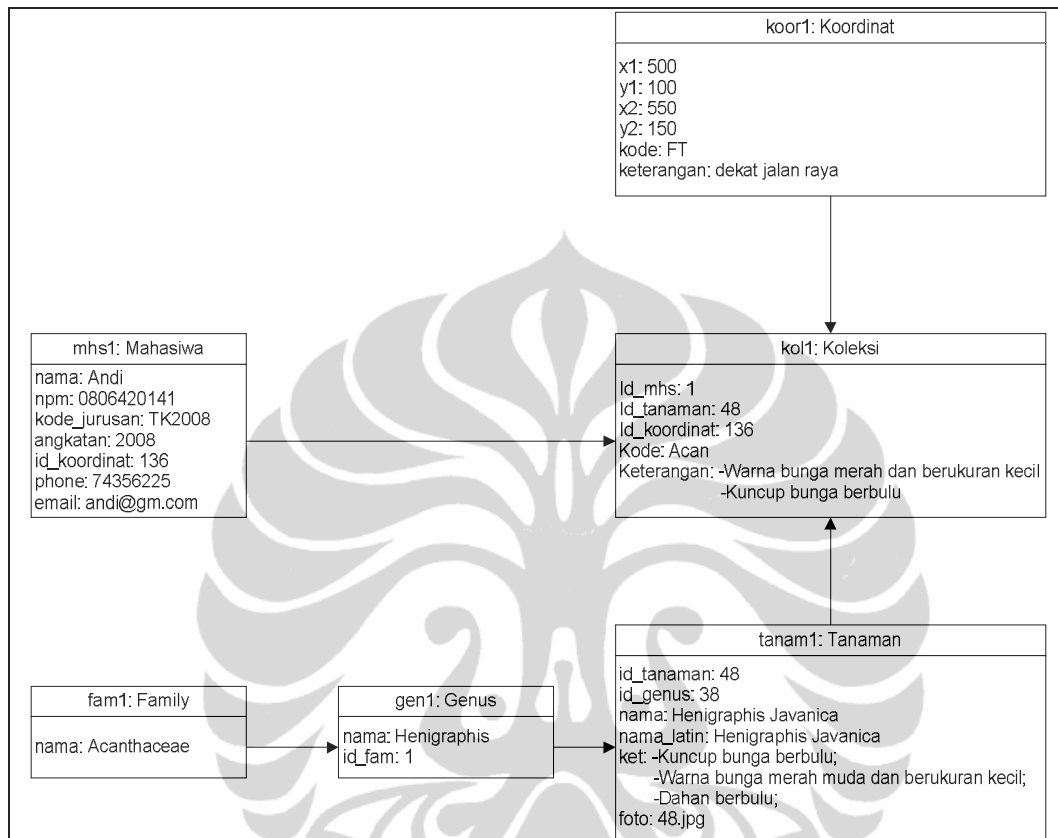
Gambar 3.2. Class Diagram

Pada Gambar 3.2. merupakan *class diagram* dengan keterangan sebagai berikut:

1. Admin dapat tidak mendaftarkan mahasiswa atau banyak mahasiswa, dan Admin dapat tidak mendaftarkan *family* atau banyak *family*.
2. Seorang mahasiswa memiliki suatu koleksi atau banyak koleksi.
3. Sebuah koordinat dimiliki seorang mahasiswa atau banyak mahasiswa pada tahun angkatan berikutnya. Dan sebuah koordinat dapat memiliki sebuah koleksi atau banyak koleksi.
4. Sebuah *family* dapat tidak mempunyai genus atau lebih dari satu genus.
5. Sebuah *genus* dapat tidak mempunyai tanaman atau lebih dari satu tanaman.

6. Sebuah tanaman memiliki satu koleksi atau banyak koleksi.

3.1.3. Object Diagram



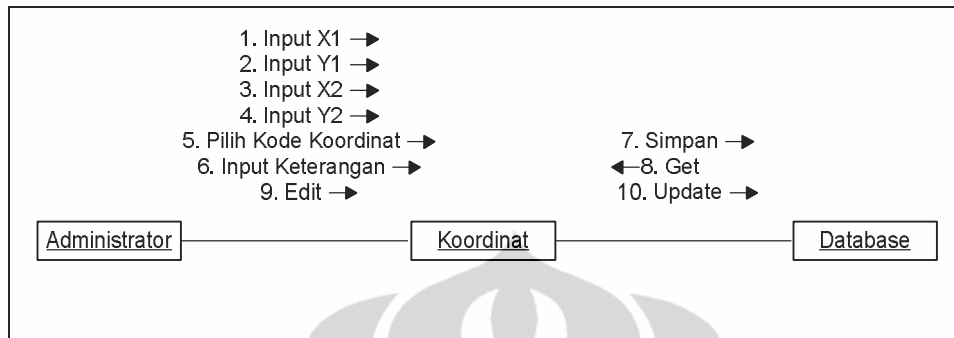
Gambar 3.3. Object Diagram

Pada Gambar 3.3. merupakan *object diagram* dengan keterangan sebagai berikut:

Mahasiswa “Andi” mendapatkan koordinat “koor1”, kemudian mengisi sebuah koleksi “kol1” dengan memilih tanaman “tanam1” dari *family* “Acanthaceae” dan *genus* “*Henigraphis*”. Kemudian menambahkan ‘kode’ dan ‘keterangan’.

3.1.4. Collaboration Diagram

3.1.4.1. Administrator Entry Koordinat

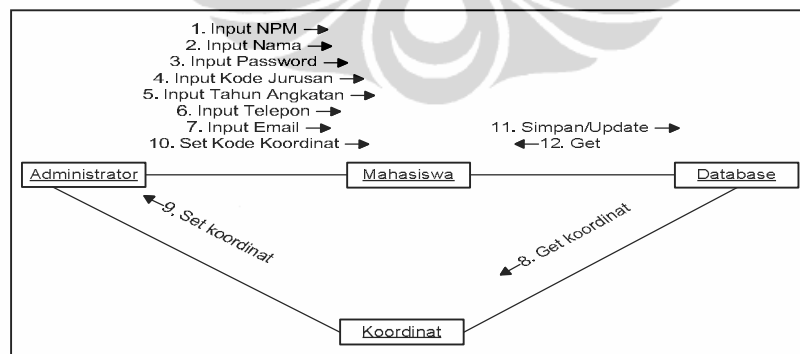


Gambar 3.4. Collaboration Diagram Administrator Entry Koordinat

Pada Gambar 3.4. merupakan *collaboration diagram* administrator entry koordinat dengan keterangan sebagai berikut:

Administrator melakukan input X1, X2, Y1, Y2, memilih kode koordinat dan input keterangan. Kemudian simpan koordinat ke database. Apabila akan melakukan perubahan koordinat, dari database memberikan id_koordinat kepada administrator dan administrator melakukan pengeditan setelah itu, kemudian di update ke database.

3.1.4.2. Administrator Entry Mahasiswa dan Menset Kode Koordinat

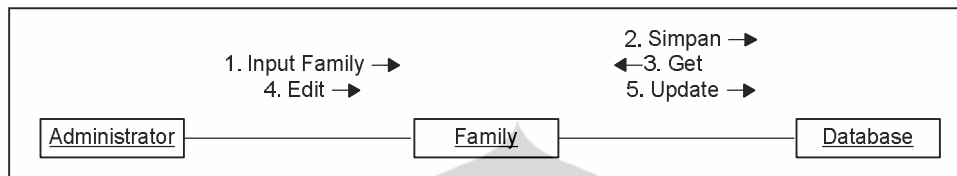


Gambar 3.5. Collaboration Diagram Administrator Entry Mahasiswa dan Menset Kode Koordinat

Pada Gambar 3.5. merupakan *collaboration diagram* administrator entry mahasiswa dan menset kode koordinat dengan keterangan sebagai berikut:

Administrator melakukan input NPM, nama, password, kode jurusan, tahun angkatan, telepon, email, kemudian dari database memberikan data koordinat dan administrator menentukan kode koordinat. Setelah itu data mahasiswa disimpan ke database.

3.1.4.3. Administrator Entry Family

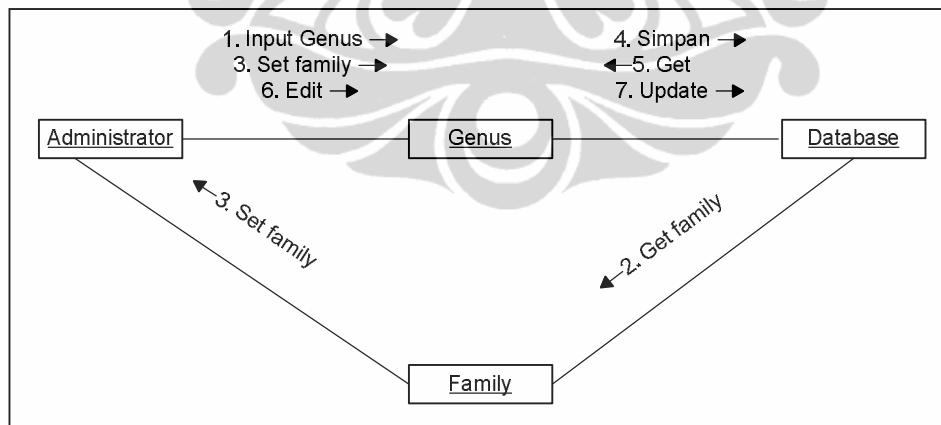


Gambar 3.6. *Collaboration Diagram Administrator Entry Family*

Pada Gambar 3.6. merupakan *collaboration diagram* administrator entry family dengan keterangan sebagai berikut:

Administrator melakukan input *family*, kemudian menyimpan ke dalam database. Apabila akan melakukan perubahan *family*, dari database memberikan *id_family* kepada administrator dan administrator melakukan pengeditan kemudian di update ke database.

3.1.4.4. Administrator Entry Genus berdasarkan Family

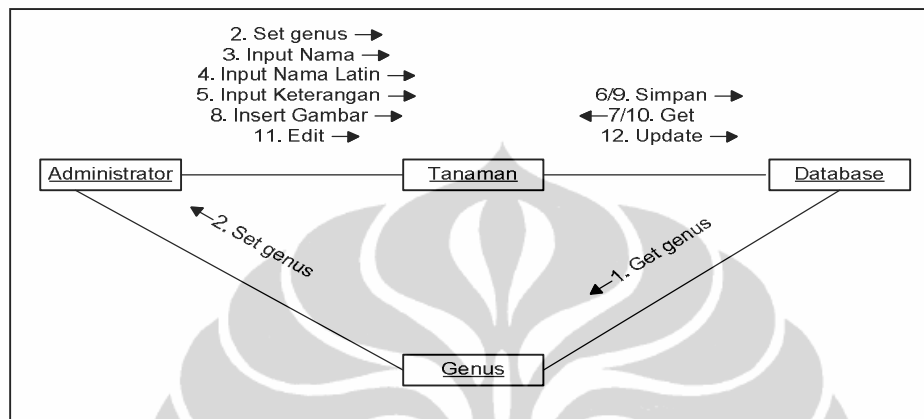


Gambar 3.7. *Collaboration Diagram Administrator Entry Genus berdasarkan Family*

Pada Gambar 3.7. merupakan *collaboration diagram* administrator entry genus berdasarkan *family* dengan keterangan sebagai berikut:

Admin input *genus* setelah itu memilih *family* kemudian disimpan dalam database. Apabila akan melakukan perubahan *genus*, dari database memberikan *id_genus* kepada administrator dan administrator melakukan pengeditan kemudian di update ke database.

3.1.4.5. Administrator Entry Tanaman berdasarkan Genus

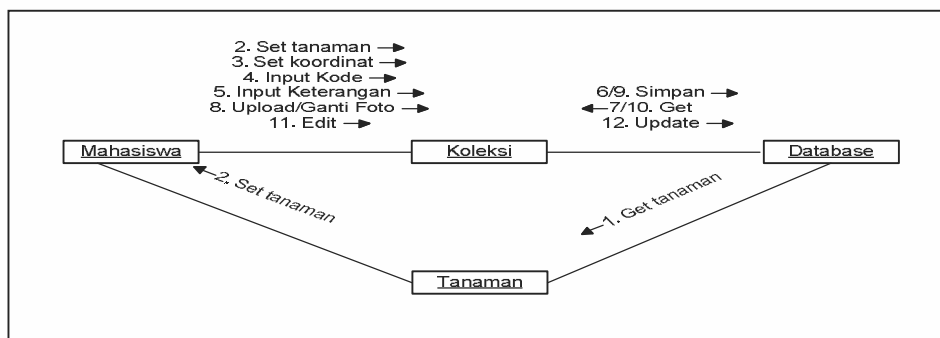


Gambar 3.8. Collaboration Diagram Administrator Entry Tanaman berdasarkan Genus

Pada Gambar 3.8. merupakan *collaboration diagram* administrator entry tanaman berdasarkan *genus* dengan keterangan sebagai berikut:

Pertama administrator memilih *family* dan *genus*, input nama, keterangan kemudian disimpan dalam database. Apabila akan memasukkan gambar, dari database akan memberikan *id_tanaman* yang dipilih kemudian *insert* gambar dan disimpan kedalam server. Apabila akan melakukan perubahan tanaman, dari database memberikan *id_tanaman* kepada administrator dan administrator melakukan pengeditan kemudian di update ke database.

3.1.4.6. Mahasiswa Entry Koleksi berdasarkan Tanaman

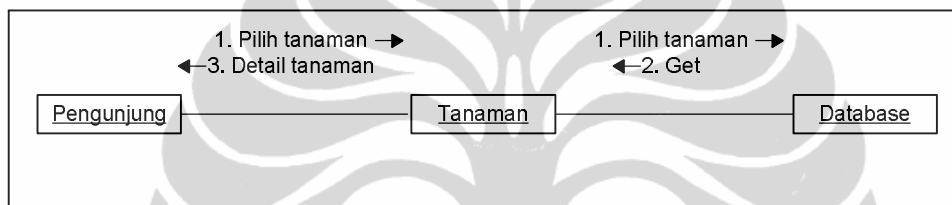


Gambar 3.9. Collaboration Diagram Mahasiswa Entry Koleksi berdasarkan Tanaman

Pada Gambar 3.9. merupakan *collaboration diagram* mahasiswa *entry* koleksi berdasarkan tanaman dengan keterangan sebagai berikut:

Mahasiswa pertama kali menset tanaman, koordinat telah ditentukan oleh administrator, input kode, input keterangan kemudian disimpan kedalam database. Apabila akan meng-*upload* atau ganti foto, dari database akan memberikan *id_koleksi* yang dipilih kemudian *insert* gambar dan disimpan kedalam server. Apabila akan melakukan perubahan koleksi, dari database memberikan *id_koleksi* kepada mahasiswa dan mahasiswa melakukan pengeditan kemudian di update ke database.

3.1.4.7. Pengunjung Melihat Data Tanaman

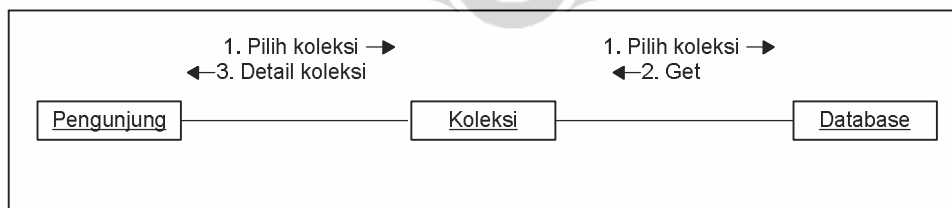


Gambar 3.10. *Collaboration Diagram* Pengunjung Melihat Data Tanaman

Pada Gambar 3.10. merupakan *collaboration diagram* pengunjung melihat data tanaman dengan keterangan sebagai berikut:

Pengunjung memilih tanaman yang berada di database, kemudian akan menampilkan detail tanaman.

3.1.4.8. Pengunjung Melihat Data Koleksi

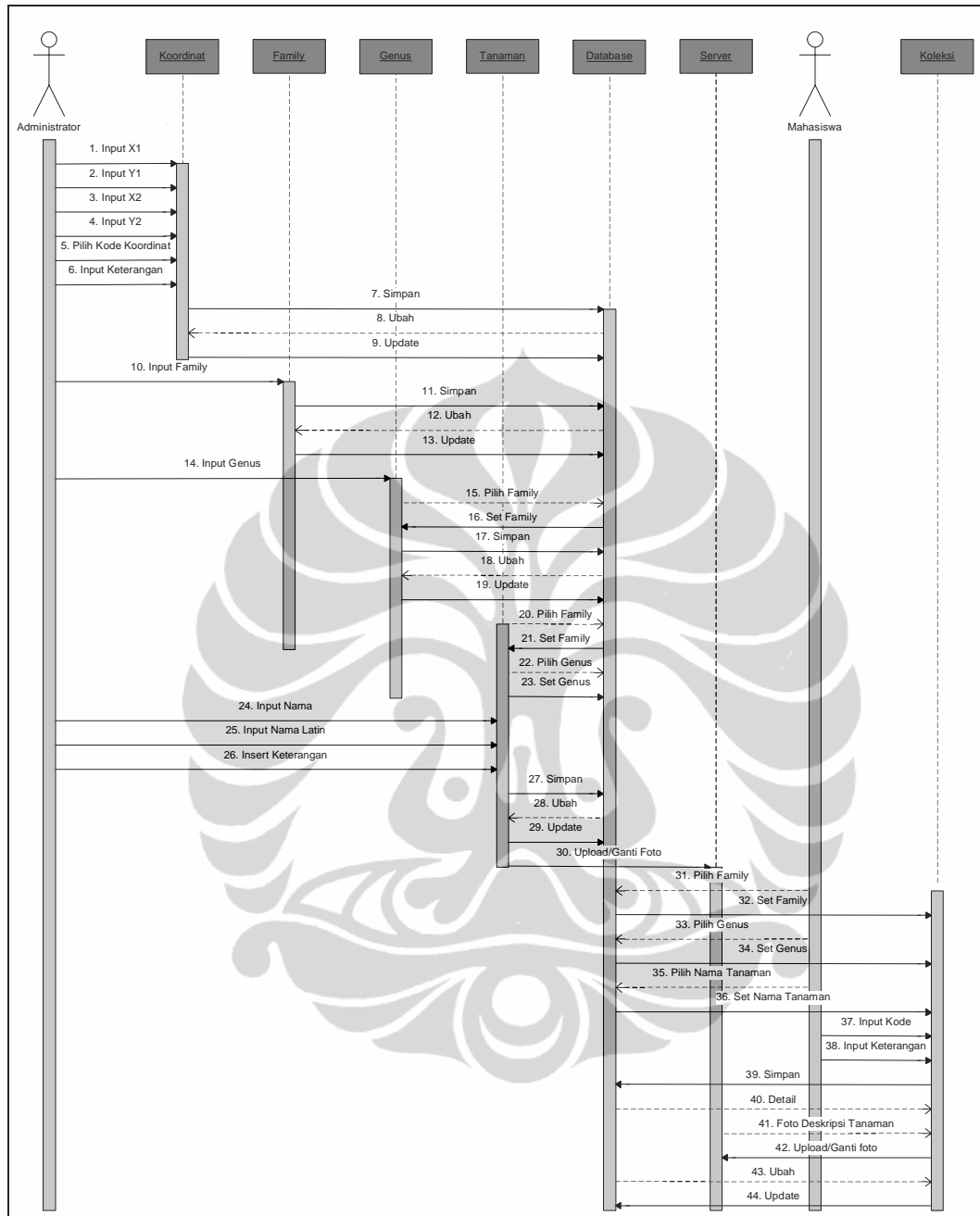


Gambar 3.11. *Collaboration Diagram* Pengunjung Melihat Data Koleksi

Pada Gambar 3.11. merupakan *collaboration diagram* pengunjung melihat data koleksi dengan keterangan sebagai berikut:

Pengunjung memilih koleksi dari mahasiswa yang berada di database, kemudian akan menampilkan detail koleksi dari mahasiswa yang dipilih.

3.1.5. Sequence Diagram



Gambar 3.12. Sequence Diagram

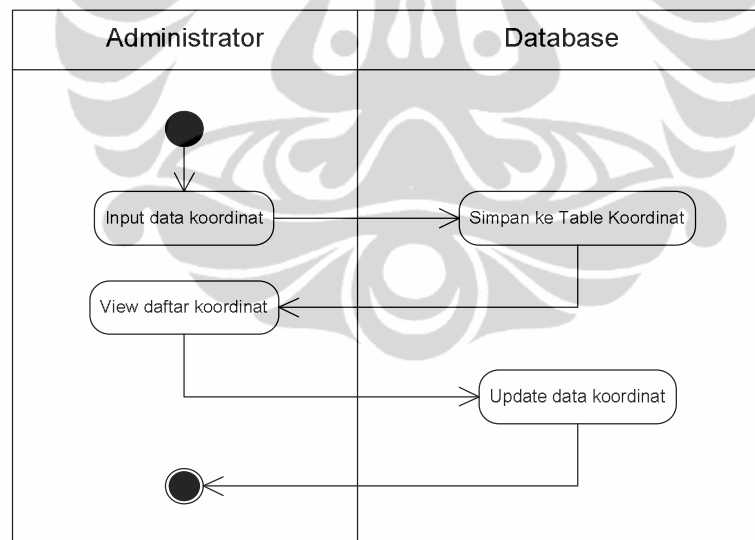
Pada Gambar 3.4. merupakan *sequence diagram* dengan keterangan sebagai berikut:

Proses pertama administrator menginput X1, Y1, X2, Y2, pilih kode koordinat dan input keterangan ke form koordinat, kemudian simpan, ubah dan update dari form koordinat ke database koordinat. Proses kedua input *family* ke form *family*, kemudian simpan, ubah dan update dari form *family* ke database

family. Proses ketiga input *genus* ke form *genus*, pilih *family* dari database *family*, set *family* yang dipilih ke form *genus*, kemudian simpan, ubah dan update dari form *genus* ke database *genus*. Proses keempat pilih *family* dan set *family* pada form tanaman, pilih *genus* dan set *genus* pada form tanaman, input nama, nama latin, keterangan ke form tanaman, kemudian simpan, ubah dan update dari form tanaman ke database tanaman, simpan upload/ganti server dari form koordinat ke server. Proses kelima pilih *family* dan set *family* pada form koleksi, pilih *genus* dan set *genus* ke form koleksi, pilih tanaman pada database tanaman dan set nama tanaman, input kode dan keterangan ke form koleksi, kemudian simpan form koleksi ke database koleksi. pada detail koleksi, menampilkan foto deskripsi tanaman ke detail koleksi, kemudian simpan upload/ganti foto dari detail koleksi ke server, ubah dan update dari form koleksi ke database koleksi.

3.1.6. State Diagram

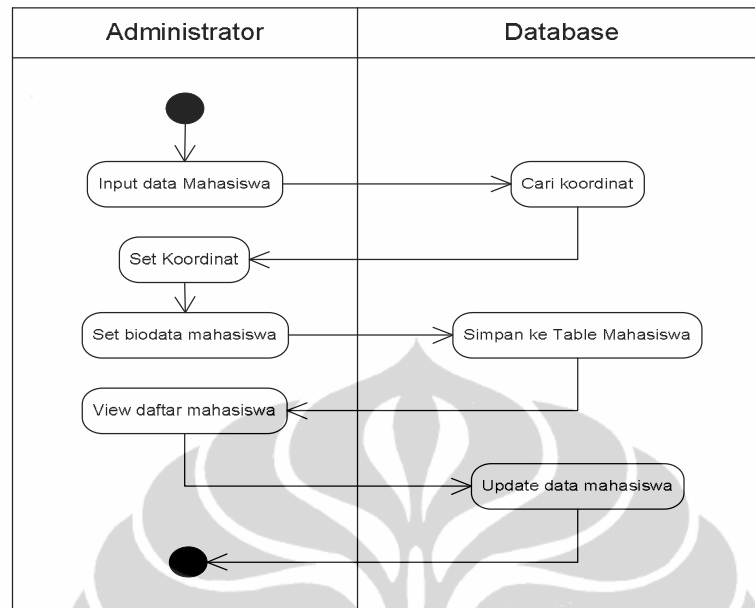
3.1.6.1. Administrator Entry Koordinat



Gambar 3.13. State Diagram Administrator Entry Koordinat

Pada Gambar 3.13. terdapat *state diagram* administrator menginput data koordinat dan disimpan ke database koordinat kemudian administrator melihat daftar koordinat dan mengupdate data koordinat.

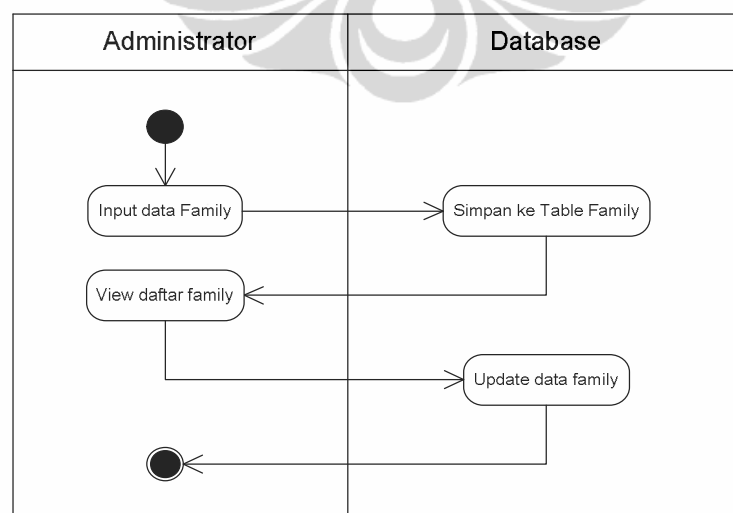
3.1.6.2. Administrator *Entry* Mahasiswa



Gambar 3.14. *State Diagram* Administrator *Entry* Mahasiswa

Pada Gambar 3.14. terdapat *state diagram* administrator menginput data mahasiswa dan cari koordinat pada database koordinat lalu menset koordinat kepada mahasiswa kemudian disimpan dalam database. Setelah itu administrator melihat daftar mahasiswa dan mengupdate data mahasiswa.

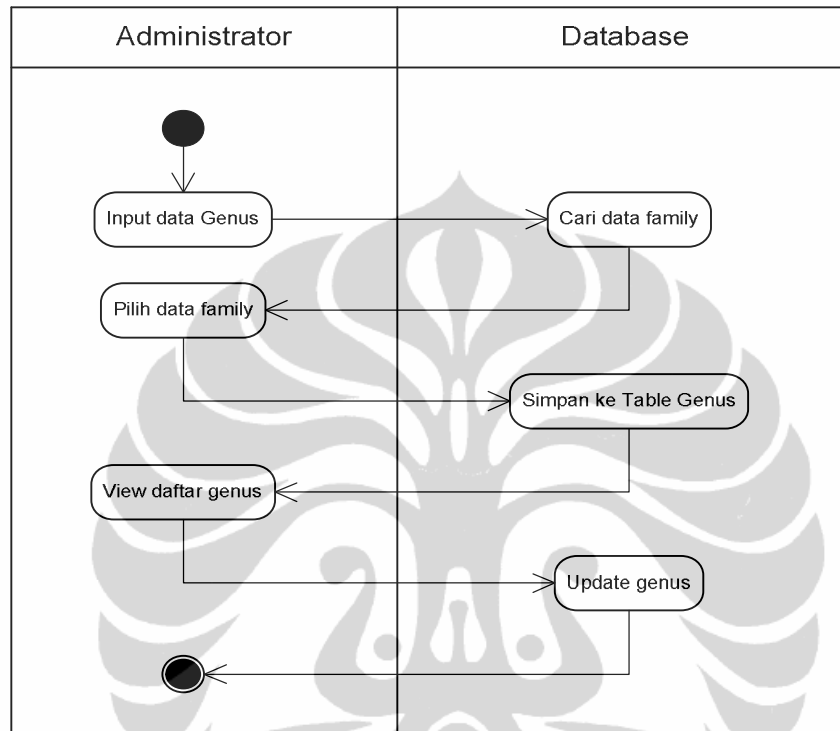
3.1.6.3. Administrator *Entry* Family



Gambar 3.15. *State Diagram* Administrator *Entry* Family

Pada Gambar 3.15. terdapat *state diagram* administrator menginput data *family* dan disimpan ke database *family* kemudian administrator melihat daftar *family* dan mengupdate data *family*.

3.1.6.4. Administrator Entry Genus

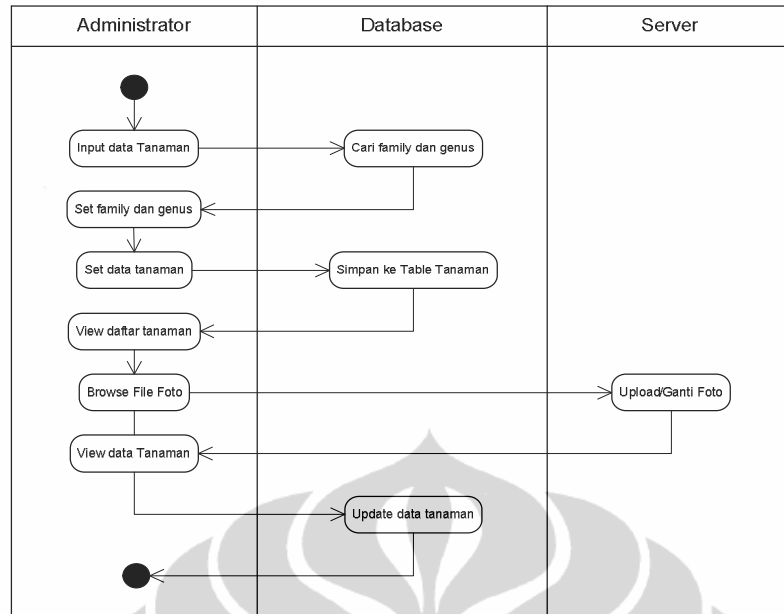


Gambar 3.16. *State Diagram Administrator Entry Genus*

Pada Gambar 3.16. terdapat *state diagram* administrator menginput data *genus* dan cari data *family* pada database *family* lalu memilih data *family* dan disimpan dalam database *genus*. Setelah itu administrator melihat daftar *genus* dan mengupdate data *genus*.

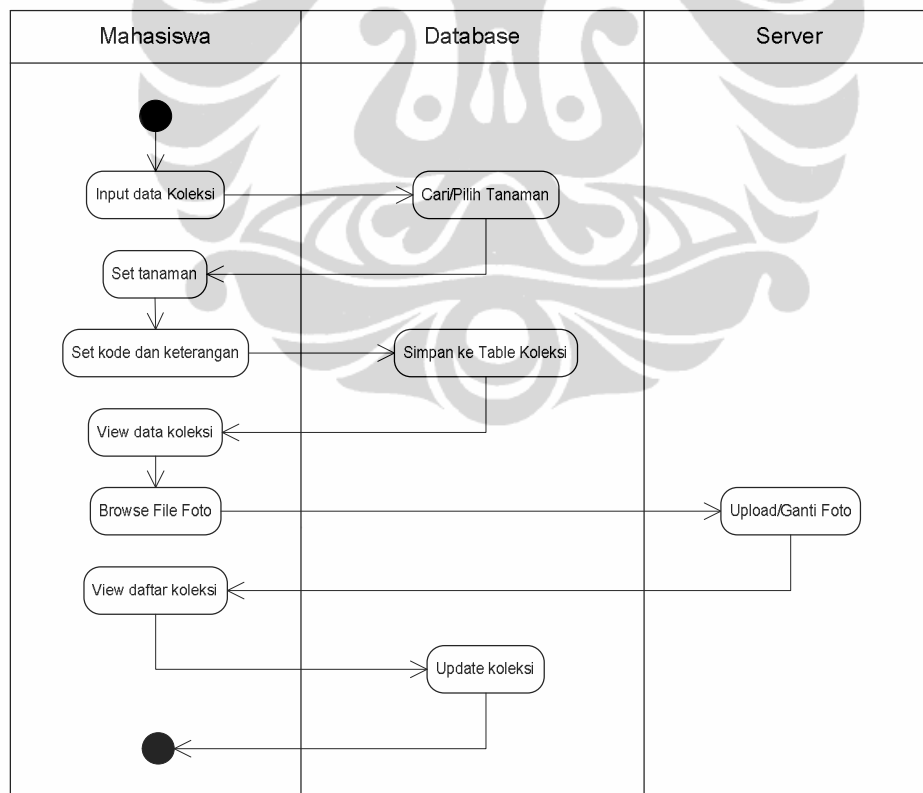
3.1.6.5. Administrator Entry Tanaman

Pada Gambar 3.17. terdapat *state diagram* administrator menginput data tanaman dan cari data *family* dan *genus* dari database *genus* lalu menset data tanaman dan disimpan ke dalam database tanaman. Setelah itu administrator melihat daftar tanaman dan mencari file foto dan mengupload/ganti foto ke server. Dan melihat data tanaman dan mengupdate data tanaman.



Gambar 3.17. *State Diagram* Administrator *Entry* Tanaman

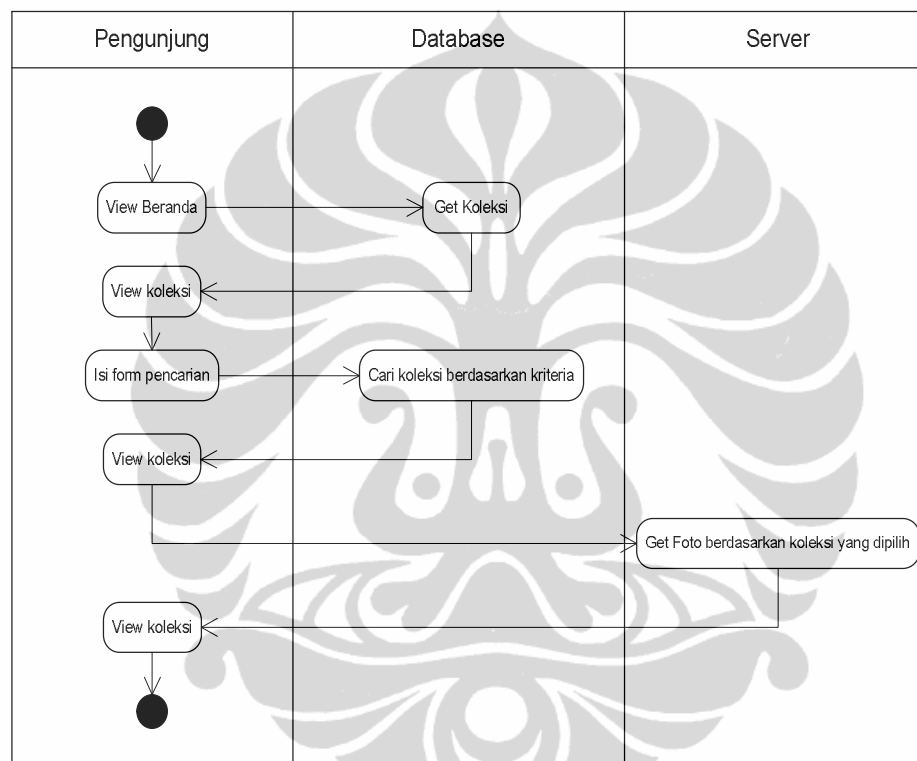
3.1.6.6. Mahasiswa *Entry* Koleksi



Gambar 3.18. *State Diagram* Mahasiswa *Entry* Koleksi

Pada Gambar 3.18. terdapat *state diagram* mahasiswa menginput data koleksi dan cari/pilih tanaman dari database tanaman lalu menset tanaman yang dipilih lalu menset data kode dan keterangan dan disimpan ke dalam database koleksi. Setelah itu mahasiswa melihat daftar koleksi dan mencari file foto dan mengupload/ganti foto ke server. Dan melihat data koleksi dan mengupdate data koleksi.

3.1.6.7. Pengunjung Cari Koleksi

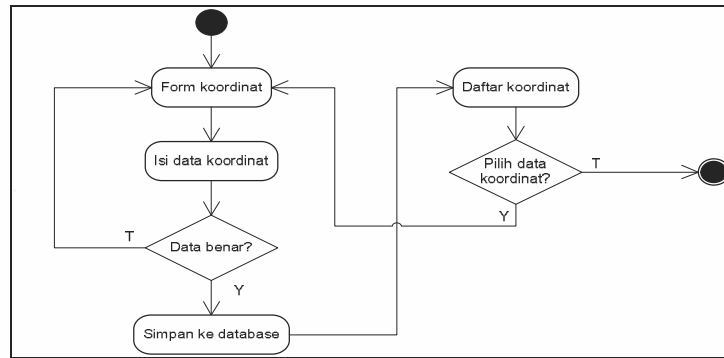


Gambar 3.19. *State Diagram* Pengunjung Cari Koleksi

Pada Gambar 3.19. terdapat *state diagram* pengunjung melihat beranda, kemudian dari data database memberikan database dan pengunjung melihat koleksi. Untuk pencarian pengunjung mengisi nama/nama latin kemudian pada database cari koleksi berdasarkan kriteria dan menampilkan koleksi dan foto berdasarkan koleksi yang dipilih dari server.

3.1.7. Activity Diagram

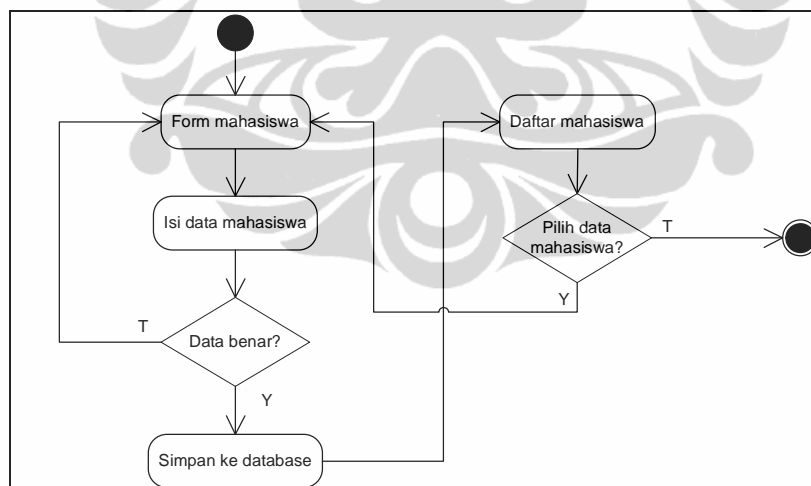
3.1.7.1. Administrator Menambahkan Koordinat



Gambar 3.20. *Activity Diagram* Administrator Menambahkan Koordinat

Pada Gambar 3.20. terdapat *activity diagram* administrator memilih form koordinat kemudian isi data koordinat, setelah itu pilihan apakah data yang diisi benar, apabila salah akan kembali ke form koordinat apabila benar akan simpan ke database koordinat, kemudian lihat daftar koordinat dan pilih data koordinat, apabila ingin memilih untuk merubah maka akan kembali ke form koordinat yang dipilih, apabila tidak maka akan selesai aktifitas administrator menambahkan koordinat.

3.1.7.2. Administrator Menambahkan Mahasiswa

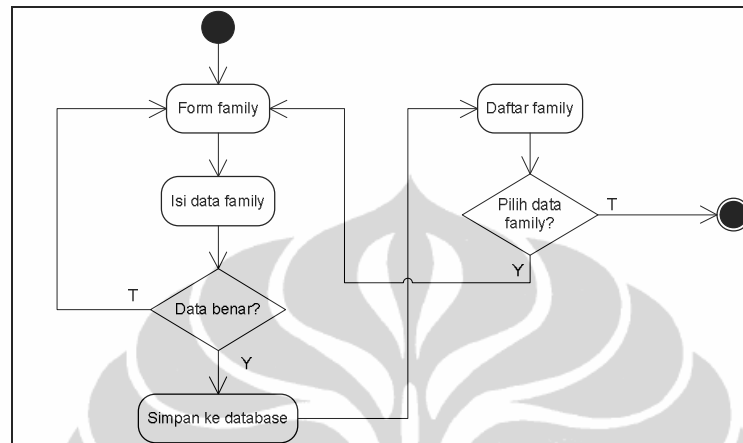


Gambar 3.21. *Activity Diagram* Administrator Menambahkan Mahasiswa

Pada Gambar 3.21. terdapat *activity diagram* administrator memilih form mahasiswa kemudian isi data koordinat, setelah itu pilihan apakah data yang diisi benar, apabila salah akan kembali ke form mahasiswa apabila benar akan simpan ke database mahasiswa, kemudian lihat daftar mahasiswa dan pilih data

mahasiswa, apabila ingin memilih untuk merubah maka akan kembali ke form mahasiswa yang dipilih, apabila tidak maka akan selesai aktifitas administrator menambahkan mahasiswa.

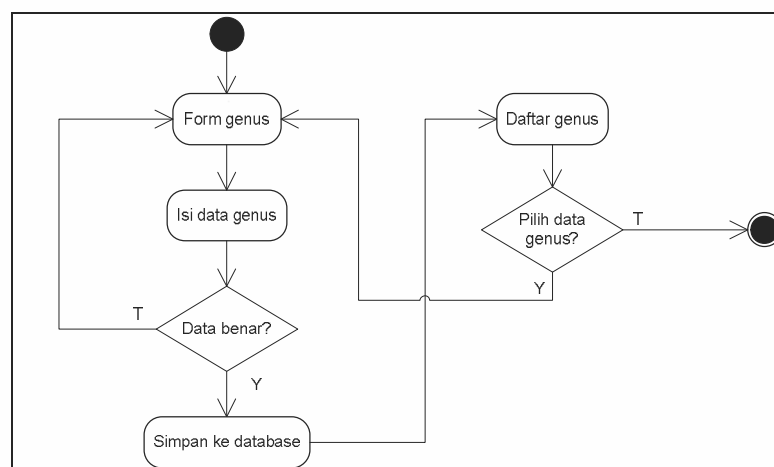
3.1.7.3. Administrator Menambahkan *Family*



Gambar 3.22. *Activity Diagram* Administrator Menambahkan *Family*

Pada Gambar 3.22. terdapat *activity diagram* administrator memilih form *family* kemudian isi data *family*, setelah itu pilihan apakah data yang diisi benar, apabila salah akan kembali ke form *family* apabila benar akan simpan ke database *family*, kemudian lihat daftar *family* dan pilih data *family*, apabila ingin memilih untuk merubah maka akan kembali ke form *family* yang dipilih, apabila tidak maka akan selesai aktifitas administrator menambahkan *family*.

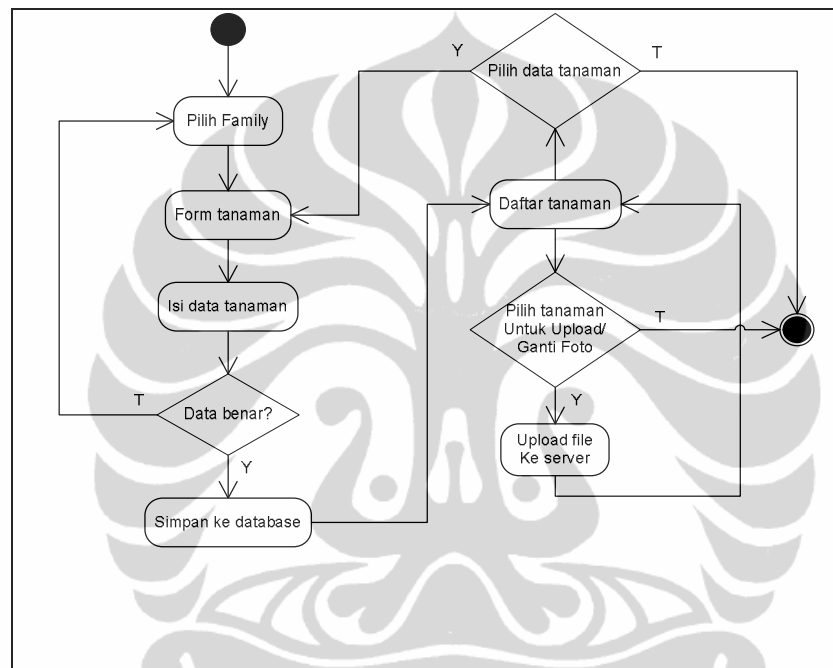
3.1.7.4. Administrator Menambahkan *Genus*



Gambar 3.23. *Activity Diagram* Administrator Menambahkan *Genus*

Pada Gambar 3.23. terdapat *activity diagram* administrator memilih form *genus* kemudian isi data *family*, setelah itu pilihan apakah data yang diisi benar, apabila salah akan kembali ke form *genus* apabila benar akan simpan ke database *genus*, kemudian lihat daftar *genus* dan pilih data *genus*, apabila ingin memilih untuk merubah maka akan kembali ke form *genus* yang dipilih, apabila tidak maka akan selesai aktifitas administrator menambahkan genus.

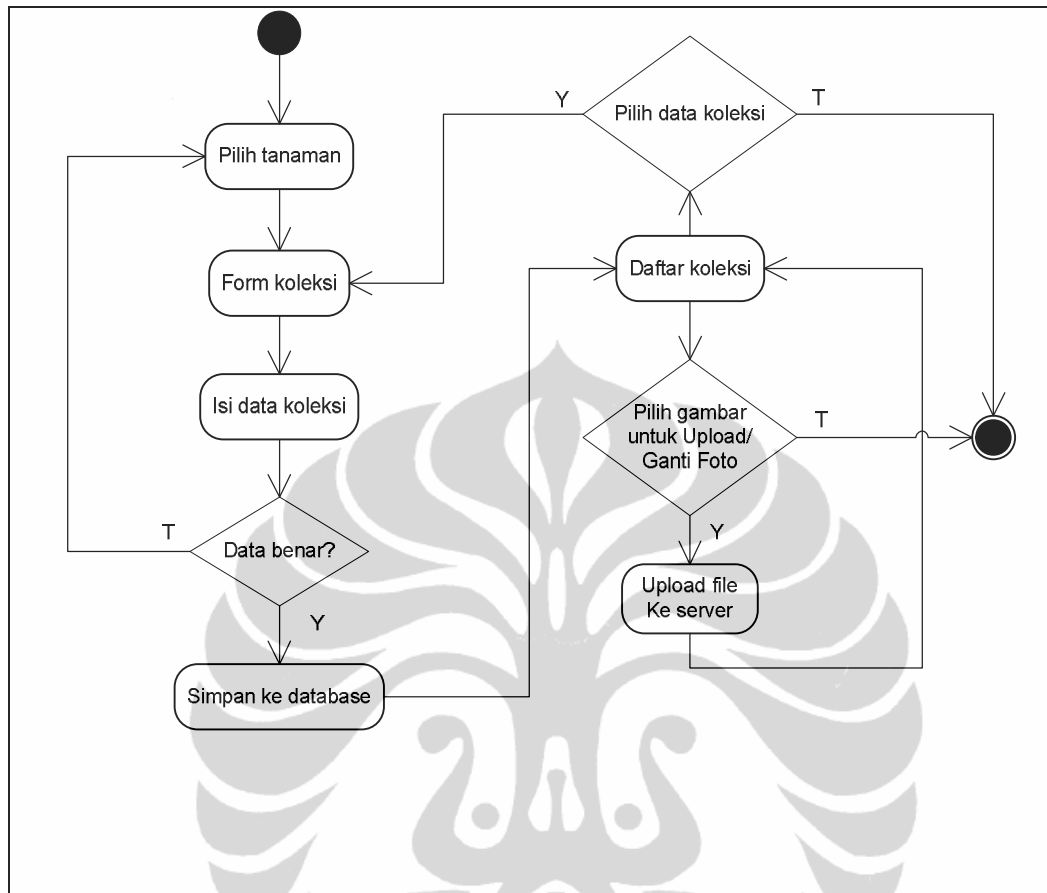
3.1.7.5. Administrator Menambahkan Tanaman



Gambar 3.24. *Activity Diagram* Administrator Menambahkan Tanaman

Pada Gambar 3.24. terdapat *activity diagram* administrator memilih *family* pada form tanaman kemudian isi data tanaman, setelah itu pilihan apakah data yang diisi benar, apabila salah akan kembali pilih *family* pada form tanaman apabila benar akan simpan ke database tanaman, kemudian lihat daftar tanaman dan pilih data tanaman, apabila ingin memilih untuk merubah maka akan kembali ke form tanaman yang dipilih, apabila tidak maka akan selesai aktifitas administrator, apabila dari daftar tanaman akan memasukkan foto pilih tanaman untuk upload/ganti foto kemudian akan diupload ke file server.

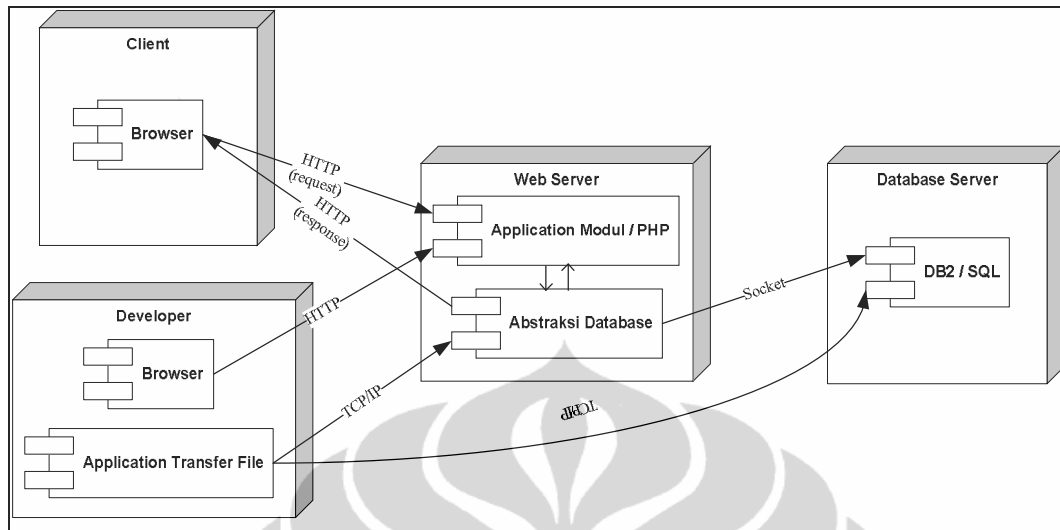
3.1.7.6. Mahasiswa Menambahkan Koleksi



Gambar 3.25. *Activity Diagram* Mahasiswa Menambahkan Koleksi

Pada Gambar 3.25. terdapat *activity diagram* mahasiswa memilih tanaman dengan menentukan *family* dan *genus* pada form koleksi terlebih dahulu kemudian isi data koleksi, setelah itu pilihan apakah data yang diisi benar, apabila salah akan kembali pilih tanaman pada form koleksi apabila benar akan disimpan ke database koleksi, kemudian lihat daftar koleksi dan pilih data koleksi, apabila ingin memilih untuk merubah maka akan kembali ke form koleksi yang dipilih, apabila tidak maka akan selesai aktifitas mahasiswa, apabila dari daftar koleksi akan memasukkan foto pilih gambar untuk upload/ganti foto kemudian akan diupload ke file server.

3.1.8. Deployment Diagram



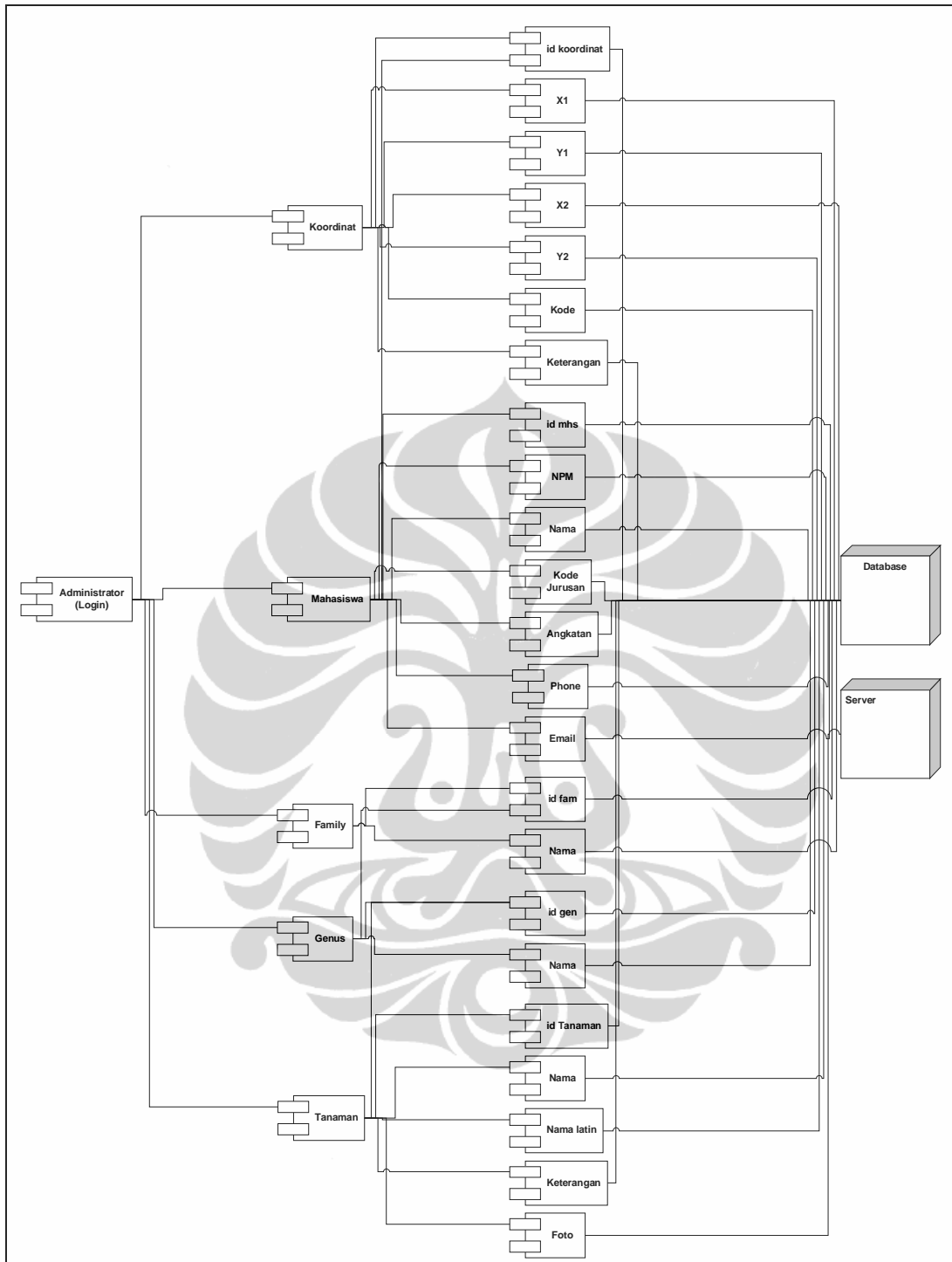
Gambar 3.26. Deployment Diagram

Pada Gambar 3.26. terdapat *deployment diagram* pada sistem client menggunakan browser dengan jalur komunikasi HTTP (*request*) kepada PHP dan dari abtraksi database memberikan *response* dengan jalur komunikasi HTTP. Pada sistem developer menggunakan browser dengan jalur komunikasi HTTP kepada Application Modul/PHP dan aplikasi transfer file dengan jalur komunikasi TCP/IP kepada Abstraksi databas dan Database Server.

3.1.9. Component Diagram

3.1.9.1. Administrator

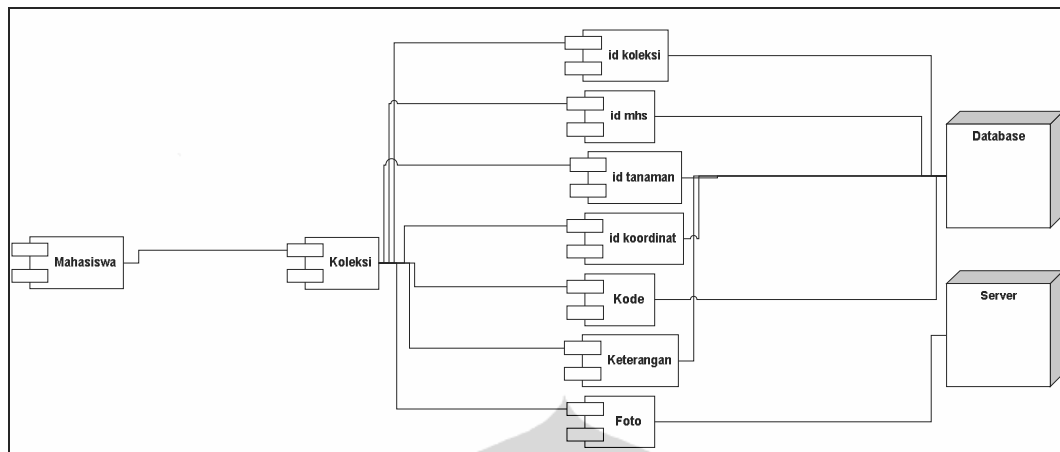
Pada Gambar 3.27. terdapat *component diagram* pada administrator memiliki 5 (lima) node, yaitu: koordinat, mahasiswa, *family*, *genus*, tanaman. Koordinat terdiri dari: id_koordinat, x1, y1, x2, y2, kode, dan keterangan disimpan ke database. Mahasiswa terdiri dari: id_mhs, id_koordinat, NPM, nama, kode_jurusan, angkatan, phone, dan email disimpan ke database. *Family* terdiri dari: id_fam dan nama disimpan ke database. *Genus* terdiri dari: id_gen, id_fam, dan nama disimpan ke database. Tanaman terdiri dari: id_tanaman, id_gen, nama, nama_latin, keterangan disimpan ke database, dan foto disimpan ke server.



Gambar 3.27. *Component Diagram Administrator*

3.1.9.2. Mahasiswa

Pada Gambar 3.28. terdapat *component diagram* pada mahasiswa memiliki sebuah node yaitu: koleksi. Yang terdiri dari id_koleksi, id_mhs, id_tanaman, id_koordinat, kode, keterangan disimpan ke database dan foto disimpan ke server.



Gambar 3.28. *Component Diagram Mahasiswa*

3.2. Perancangan Database Aplikasi

Sebelum masuk ke dalam tahapan perancangan halaman, perlu dipersiapkan tempat untuk menampung data (database) di server yang berisikan tabel-tabel yang akan digunakan pada Aplikasi. Pembuatan database dengan menggunakan MySQL phpmyadmin versi 3.1.3.1.

3.2.1. Membuat Tabel

Langkah selanjutnya adalah pembuatan tabel. Tabel-tabel dibuat Penulis, dimaksudkan untuk membedakan letak penyimpanan data. Penulis membuat 7 (tujuh) buah tabel untuk Aplikasi yaitu:

1. Tabel Admin

Tabel admin ini berfungsi sebagai tempat penampungan data administrator. Struktur data tabel admin adalah sebagai berikut. Lihat Tabel 3.1.

Tabel 3.1. Tabel Admin

Field	Jenis	Ukuran	Keterangan
id_admin	Integer	11	Primary key, Auto_increment
Username	Varchar	50	
Pass	Varchar	32	Md5
Nip	Varchar	18	
Fullname	Varchar	50	
Email	Varchar	50	

Phone	Varchar	20	
-------	---------	----	--

Tabel admin ini username digunakan untuk menyimpan data dari nama admin. Pass merupakan tempat penyimpanan password admin dengan function MD5. NIP merupakan nomor induk pegawai admin. *Fullname* merupakan nama lengkap admin. Email digunakan sebagai tempat penyimpanan data alamat *e-mail* dari admin. *Phone* merupakan nomer telepon admin.

2. Tabel Koordinat

Tabel koordinat ini dibuat untuk menentukan posisi letak/daerah koordinat dengan satu koordinat satu mahasiswa. Struktur dari tabel koordinat adalah sebagai berikut. Lihat Tabel 3.2.

Tabel 3.2. Tabel Koordinat

Field	Jenis	Ukuran	Keterangan
id_koordinat	Integer	11	Primary key, Auto_increment
X1	Integer	11	
Y1	Integer	11	
X2	Integer	11	
Y2	Integer	11	
Kode	Varchar	20	
Keterangan	Text		

Kode digunakan untuk menentukan tempat atau daerah yang dipilih. Keterangan merupakan keterangan dari daerah koordinat yang ditentukan. Pada field keterangan mempunyai tipe data *Text*, tipe data ini digunakan untuk menyimpan data string dengan jumlah karakter tidak terbatas.

3. Tabel Mahasiswa

Tabel mahasiswa ini berfungsi sebagai tempat penampungan data dan informasi mahasiswa, dimana koordinat telah ditentukan oleh admin. Struktur dari tabel mahasiswa adalah sebagai berikut. Lihat Tabel 3.3.

Tabel 3.3. Tabel Mahasiswa

Field	Jenis	Ukuran	Keterangan
id_mhs	Integer	11	Primary key, Auto_increment
id_koordinat	Integer	11	
Npm	Varchar	10	
Nama	Varchar	50	
Pass	Varchar	32	Md5
kode_jurusan	Varchar	10	
Angkatan	Integer	11	
Phone	Varchar	20	
Email	Varchar	50	

Tabel mahasiswa ini nama digunakan untuk menyimpan data dari nama admin. Pass merupakan tempat penyimpanan password mahasiswa dengan function MD5. NPM merupakan nomor pokok mahasiswa. Kode_jurusan merupakan kode dari jurusan mahasiswa. Angkatan merupakan tahun masuk mahasiswa. Phone merupakan nomer telepon mahasiswa. Email digunakan sebagai tempat penyimpanan data alamat *e-mail* dari mahasiswa dan digunakan sebagai user login.

4. Tabel *Family*

Tabel *family* ini berfungsi sebagai tempat penampungan data family tanaman. Struktur data tabel *family* adalah sebagai berikut. Lihat Tabel 3.4.

Tabel 3.4. Tabel *Family*

Field	Jenis	Ukuran	Keterangan
id_fam	integer	11	Primary key, Auto_increment
Nama	varchar	40	

Tabel *family* ini nama digunakan untuk menyimpan data dari nama family tanaman.

5. Tabel *Genus*

Tabel *genus* ini berfungsi sebagai tempat penampungan data genus tanaman. Dan memilih data *family* tanaman. Struktur data tabel *genus* adalah sebagai berikut. Lihat Tabel 3.5.

Tabel 3.5. Tabel Genus

Field	Jenis	Ukuran	Keterangan
id_gen	integer	11	Primary key, Auto_increment
Nama	varchar	40	
id_fam	integer	11	

Tabel *genus* ini nama digunakan untuk menyimpan data dari nama *genus* tanaman. Id_fam digunakan untuk menyimpan *family* yang dipilih.

6. Tabel Tanaman

Tabel tanaman ini berfungsi sebagai tempat penampungan data tanaman. Struktur data tabel tanaman adalah sebagai berikut. Lihat tabel 3.6.

Tabel 3.6. Tabel Tanaman

Field	Jenis	Ukuran	Keterangan
id_tanaman	integer	11	Primary key, Auto_increment
id_genus	integer	11	
Nama	varchar	50	
Nama_latin	varchar	50	
Ket	text		

Tabel tanaman ini id_genus sebagai *foreign key* dari tabel *genus*. Nama digunakan untuk menyimpan data nama tanaman. Nama latin digunakan untuk menyimpan data nama latin tanaman. Ket merupakan keterangan dari tanaman yang diinput administrator. Pada field keterangan mempunyai tipe data *text*, tipe data ini digunakan untuk menyimpan data string dengan jumlah karakter tidak terbatas.

7. Tabel Koleksi

Tabel koleksi ini berfungsi sebagai tempat penampungan data koleksi. Struktur data tabel koleksi adalah sebagai berikut. Lihat tabel 3.7.

Tabel 3.7. Tabel Koleksi

Field	Jenis	Ukuran	Keterangan
id_koleksi	integer	11	Primary key, Auto_increment
id_mhs	integer	11	
id_tanaman	integer	11	
id_koordinat	integer	11	
Kode	varchar	20	
Keterangan	text		
Jumlah	integer	11	

Pada tabel koleksi ini id_mhs dan id_koordinat telah ditentukan oleh administrator, sehingga mahasiswa hanya bisa memilih id_tanaman, kode dan keterangan. Kode merupakan tempat untuk memasukkan kode yang diinginkan mahasiswa. Keterangan merupakan keterangan dari tanaman yang diinput mahasiswa. Pada field keterangan mempunyai tipe data *text*, tipe data ini digunakan untuk menyimpan data string dengan jumlah karakter tidak terbatas. Jumlah merupakan banyaknya jumlah suatu tanaman yang didapatkan mahasiswa.

3.3. Desain Tampilan Secara Umum

Pemahaman tentang bagaimana cara mengadopsi suatu pendekatan dalam pembuatan tampilan Sistem Informasi *Biological Diversity* Universitas Indonesia. Berikut adalah pertimbangan yang diambil dalam membuat desain tampilan sistem secara umum:

1. Memiliki halaman user publik (*front end*) dan user administrator (*back end*). Halaman user publik (*front end*) dapat digunakan oleh mahasiswa untuk mendaftarkan koleksi dan pengunjung sistem untuk melihat data tanaman, data koleksi, peta tanaman, mencari dan mengunduh (*download*) data koleksi. Sedangkan halaman user administrator hanya dapat

digunakan oleh admin dan berfungsi untuk mendaftarkan koordinat posisi tanaman, mahasiswa pengentri data, *family*, *genus*, dan tanaman.

2. Setiap halaman yang memerlukan otentikasi pengguna menampilkan informasi pengguna tersebut. Pada halaman pemakai mahasiswa, akan tampil nama mahasiswa, NPM, kode lokasi, koordinat dan kode koordinat. Sementara pada halaman administrator akan tampil nama user administrator tersebut.
3. Tampilan sistem memanfaatkan maksimal halaman sistem untuk user dengan resolusi minimal 800 x 600 piksel. Layar tampilan penuh memaksimalkan kemampuan komunikasi informasi tampilan, memungkinkan sistem untuk menjadi fokus perhatian pengguna, dan menyembunyikan alat-alat navigasi dari sistem infrastruktur *windows browser*.
4. Meminimalkan penggunaan *window* untuk memanfaatkan lebar sistem dan implementasi CSS dan *Ajax* (tidak menggunakan *window pop-up*).
5. Menggunakan pembatas (*border*) untuk setiap konten sistem seperti tabel, *font*, gambar/*image*, dan teks.
6. Menggunakan *font*, tabel, ukuran *image*, dan elemen tampilan yang konsisten. Tata letak yang konsisten di seluruh browser dan platform akan meningkatkan kegunaan sistem dengan mempertahankan interface (antar muka) pengguna yang stabil yang bisa diandalkan.
7. Menggunakan *rollovers* dan *ALT-overs*. *Rollovers* umumnya digunakan pada *web* dan pengguna untuk menaik turunkan layar dan biasanya terdapat di ujung/tepi *web browser*. *ALT-overs* digunakan ketika kedekatan *rollovers* tersebut tidak diperlukan. *ALT-overs* hadir dalam banyak aplikasi klien dan karena itu sudah akrab bagi pengguna.
8. Memperhatikan estetika tampilan sistem dengan menggunakan penamaan. Estetika tampilan dapat memberikan keakraban pengguna, rasa kepercayaan dan profesionalisme.
9. Menggunakan standar link sistem yang mudah dimengerti. Pengguna *web* menggunakan ini untuk eksplorasi dan akan menggerakkan mouse sampai area diklik muncul.

10. Halaman sistem memiliki struktur menu yang konsisten berdasarkan aturan maksimal 7 menu [20]. Dalam sistem ini user publik (*front end*) menggunakan 6 menu, yaitu: Beranda, Tanaman, Koleksi, Peta Tanaman, Pencarian dan *Download*. Sementara user administrator (*back end*) menggunakan 6 menu, yaitu: Beranda, Tanaman, Koordinat, Mahasiswa, *Family*, *Genus*.



BAB 4

IMPLEMENTASI DAN ANALISA PROGRAM

4.1. Implementasi Program

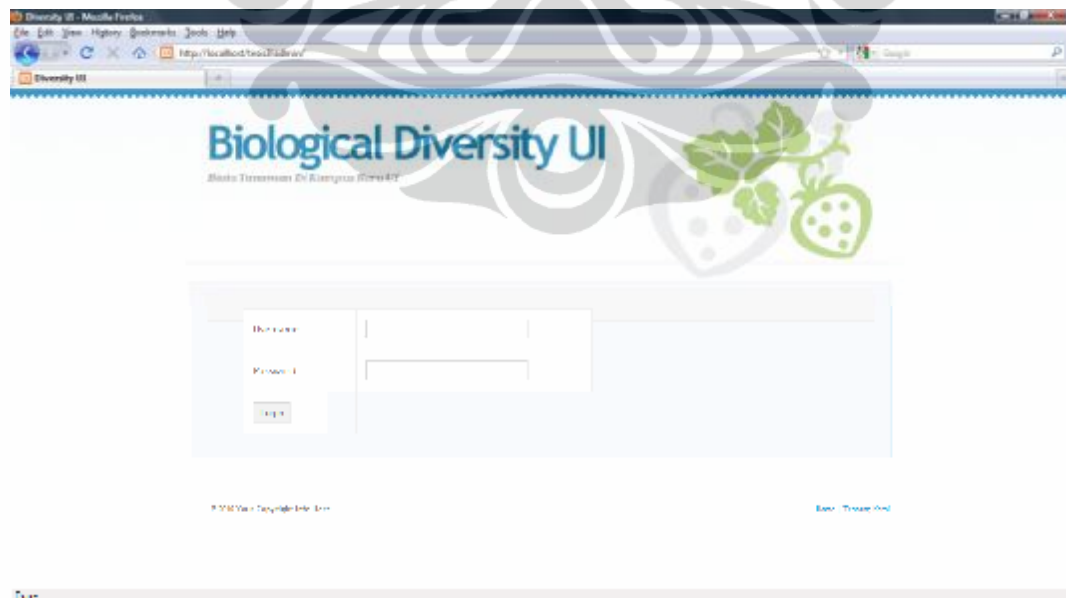
Pada bab ini akan dilakukan mengenai implementasi program Sistem Informasi *Biological Diversity* Universitas Indonesia yang telah dirancang. Dalam lingkup uji coba sistem ini adalah menggunakan komputer laptop (notebook) dengan spesifikasi CPU Intel® Core™2 Solo processor SU3500(1.4GHz, 800 MHz FSB, VGA Intel® GMA 4500MHD Up to 1244 MB, Memory 4GB DDR3 dan dalam pembuatan Sistem Informasi *Biological Diversity* dibuat dengan bahasa pemrograman PHP dan database MySQL agar tampilan lebih *user-friendly*.

Dalam implementasi sistem membagi halaman menjadi 2 (dua) bagian utama yaitu Halaman Administrator dan Halaman Mahasiswa.

4.1.1. Halaman Administrator

Pada halaman administrator ini halaman dibagi menjadi 7 (tujuh) halaman, yaitu:

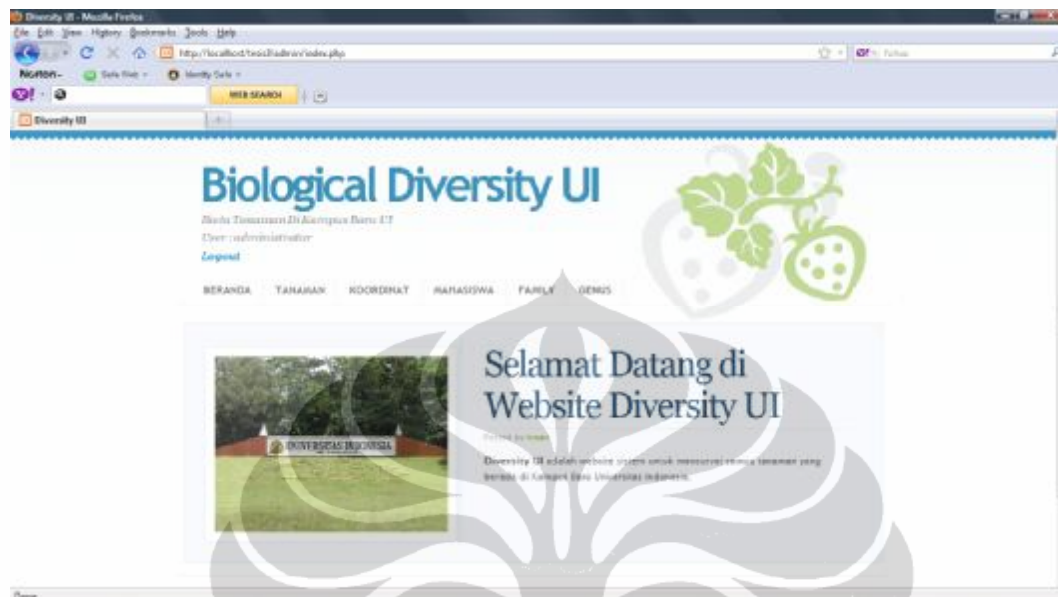
4.1.1.1. Halaman Login



Gambar 4.1. Halaman Login Administrator

Pada Gambar 4. 1. halaman login ini hanya bisa administrator yang bisa mengakses dengan memasukkan *username* dan *password*.

4.1.1.2. Halaman Index Administrator



Gambar 4.2. Halaman Index Administrator

Setelah memasukan *username* dan *password*, pada halaman index administrator terdapat beberapa bagian diantaranya beranda, tanaman, koordinat, mahasiswa, *family*, dan *genus*. Lihat pada Gambar 4.2 diatas.

4.1.1.3. Halaman Form Koordinat

Pada halaman form koordinat *administrator* memasukkan koordinat x1, y1, x2, y2 dengan menggunakan ukuran piksel gambar. Kode koordinat dengan memilih daerah koordinat, seperti FT, FE, FMIPA, dll. Dan keterangan dituliskan dengan keterangan letak, seperti: di seberang FT.

Setelah disimpan maka akan tampil Daftar Koordinat untuk menampilkan seluruh data koordinat.

4.1.1.4. Halaman Form Mahasiswa

Pada halaman form mahasiswa masukkan NPM, Nama, Password, Kode Jurusan, Tahun Angkatan, Telepon, Email dan pilih Kode Koordinat.

Setelah disimpan maka akan tampil Daftar Mahasiswa untuk menampilkan seluruh data mahasiswa.

4.1.1.5. Halaman Form *Family*

Pada halaman form *family* ini masukkan nama *family* tanaman.

Setelah disimpan maka akan tampil Daftar *Family* untuk menampilkan seluruh data *family*.

4.1.1.6. Halaman Form *Genus*

Pada halaman form *genus* masukkan nama *genus* makanan dan pilih *family* tanaman.

Setelah disimpan maka akan tampil Daftar *Genus* untuk menampilkan seluruh data *genus*.

4.1.1.7. Halaman Form Tanaman

Pada halaman form tanaman *administrator* memilih *family*, kemudian memilih *genus*, dan mengisi Nama, Nama Latin, Keterangan. Apabila ingin memasukkan foto, silahkan pilih Gambar pada kotak Daftar Tanaman.

Setelah disimpan maka akan tampil Daftar Tanaman untuk menampilkan seluruh data tanaman.

4.1.2. Halaman Mahasiswa

Pada halaman administrator ini halaman dibagi menjadi 7 (tujuh) halaman, yaitu:

4.1.2.1. Halaman Login

Pada halaman login ini mahasiswa diharapkan mengisi *e-mail* dan *password* sesuai dengan lampiran yang diisi oleh mahasiswa.

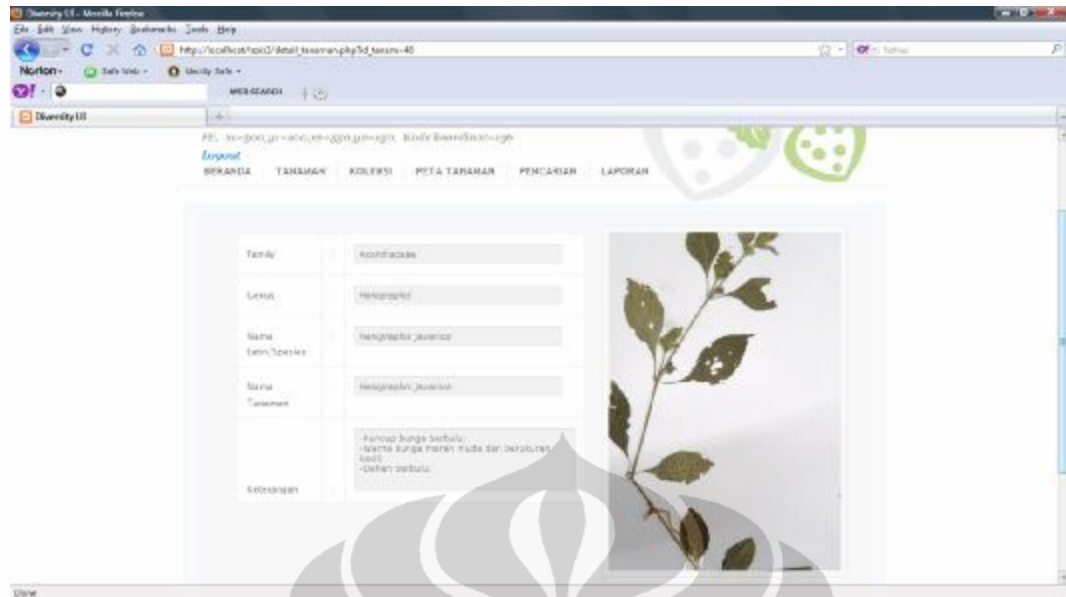
4.1.2.2. Halaman Index Mahasiswa

Pada halaman index atau beranda mahasiswa dapat memilih tanaman, koleksi, peta tanaman, pencarian dan laporan.

4.1.2.3. Halaman Tanaman

Pada halaman tanaman ini berfungsi untuk menampilkan data-data tanaman yang terdapat dalam sistem. Pertama mahasiswa memilih *family* terlebih dahulu, kemudian akan tampil daftar tanaman dari *family* yang telah dipilih.

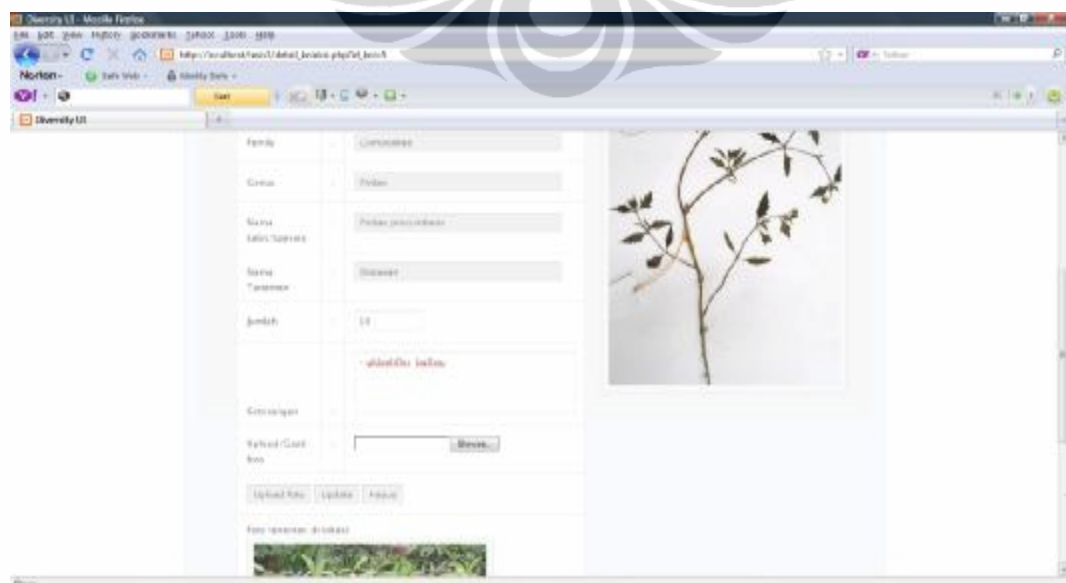
Apabila ingin menampilkan nama, nama latin, keterangan silahkan tekan link Detail. Lihat Gambar 4.3. dibawah.



Gambar 4.3. Halaman Detail Tanaman

4.1.2.4. Halaman Koleksi

Pada halaman koleksi ini merupakan halaman yang berfungsi untuk mahasiswa memasukkan data-data koleksi yang didapatkan dari lapangan/area koordinat yang telah ditetapkan. Cara pengisiannya, yaitu: pada tabel tanaman terlebih dahulu pilih *family*, *genus* dan tanaman, kemudian mengisikan kode tanaman, jumlah koleksi yang sama, dan keterangan. Pada halaman detail koleksi berfungsi untuk mengupload/ganti foto dan mengupdate koleksi tanaman. Lihat Gambar 4.4 dibawah ini.



Gambar 4.4. Halaman Detail Koleksi

Universitas Indonesia

4.1.2.5. Halaman Peta Tanaman



Gambar 4.5. Halaman Peta Tanaman

Pada Gambar 4.5. halaman peta tanaman menampilkan gambar dari Kampus Universitas Indonesia yang diambil dari www.ui.ac.id/id/campus/page/peta-kampus-depok. Halaman Peta FT diambil dari Google Earth. Pada halaman Peta Tanaman dan Peta FT ini dibagi menjadi beberapa koordinat. Untuk memetakan koordinat menggunakan ukuran piksel [13]. Lihat Gambar 4.6 dibawah ini.



Gambar 4.6. Halaman Peta FT

4.1.2.6. Halaman Pencarian



Gambar 4.7. Halaman Pencarian

Pada Gambar 4.7. halaman pencarian ini berfungsi untuk mencari nama/nama latin dari data-data tanaman koleksi yang dimasukkan oleh mahasiswa.

4.1.2.7. Halaman Laporan



Gambar 4.8. Halaman Laporan

Pada Gambar 4.8. halaman laporan ini berfungsi untuk melihat daftar tanaman yang berada di Kampus Baru Universitas Indonesia dan laporan jumlah koleksi tanaman dari mahasiswa berdasarkan tahun dan lokasi. Lihat Gambar 4.9 dibawah ini.

Nomor	NPM	Nama Mahasiswa	Family	Genus	Spesies/Nama Latin	Nama Tanaman	Kode	Lokasi	Keterangan	Jumlah	Tanggal Entry
1	0996420141	Andi	Compositae	Zizania	Zizania procumbens	Gebagan	AG005	BE	gibokhin bglup	10	2010-11-03
2	0996424472	Irwana Kurnia Andrianto	Asteraceae	Alopecurus	Alopecurus maculatus	(Hibiscus swartzii)	FT-3500-AM	FT	Dasar tangkap, Petalip dan berwana hijau dengan garis kuning-kuning.	13	2010-11-03
3	0996424472	Irwana Kurnia Andrianto	Compositae	Agave	Agave americana	Bundaran	FT-3500-BM	FT	-Lihat di halaman dan petalip -Maka berwana putih kekuning-kuningan. Bunga-bunga putih dengan petalip. Bunga-bunga dan kelopak. Daun berwana hijau, tebal. Daunnya, ada beberapa ada berwana putih. Daun rampai.	7	2010-11-03
4	0996424472	Irwana Kurnia Andrianto	Malvaceae	Malva	Malva sylvestris	Malva	FT-3500-SM	FT	Petalip. Maka berwana merah muda kelopak. Bunga-bunga putih. Daunnya dan kelopak. Bunga rampai.	10	2010-11-03
5	0996424472	Irwana Kurnia Andrianto	Malvaceae	Alcea	Alcea rosea	Alcea	FT-3500-SM	FT	Petalip. Maka berwana merah muda kelopak. Bunga-bunga putih. Daunnya dan kelopak. Bunga rampai.	14	2010-11-03
6	0996424472	Irwana Kurnia Andrianto	Compositae	Helianthus	Helianthus annuus	Helianthus	FT-3500-SM	FT	Daun rampai dan petalip. Bunga-bunga putih.	10	2010-11-03
7	0996424472	Irwana Kurnia Andrianto	Compositae	Helianthus	Helianthus annuus	Helianthus	FT-3500-SM	FT	Daun rampai dan petalip. Bunga-bunga putih.	14	2010-11-03
8	0996424472	Irwana Kurnia Andrianto	Malvaceae	Alcea	Alcea rosea	Alcea	FT-3500-SM	FT	Daun rampai dan petalip. Bunga-bunga putih.	14	2010-11-03

Gambar 4.9. Laporan Koleksi.Tanaman

4.2. Pengujian Oleh Pengguna

Umpan balik dari pengguna didapatkan dengan mempersilahkan 20 responden mencoba sistem, kemudian meminta mereka mengisi kuesioner yang dapat dilihat pada lampiran 1. Dari kuesioner tersebut didapatkan data seperti yang ditampilkan oleh Tabel 4.1:

Tabel 4.1. Data tanggapan terhadap kuesioner

Pertanyaan	Jumlah Tanggapan Terhadap Nilai			
	1	2	3	4
1. Anda familiar dengan PHP	8	4	6	2
2. Anda familiar dengan MySQL	8	3	7	2
Keterangan: 1 = sangat tidak familiar (tidak pernah mendengar) 2 = kurang familiar (pernah mendengar namun sama sekali tidak tahu) 3 = familiar (pernah mendengar, tahu, dan pernah menggunakan) 4 = sangat familiar (pernah mendengar, tahu, dan sering menggunakan)				
3. Desain tampilan sistem baik	0	2	17	1
4. Sistem <i>user-friendly</i>	1	2	16	1
5. Database tanaman tersimpan baik	0	3	16	1

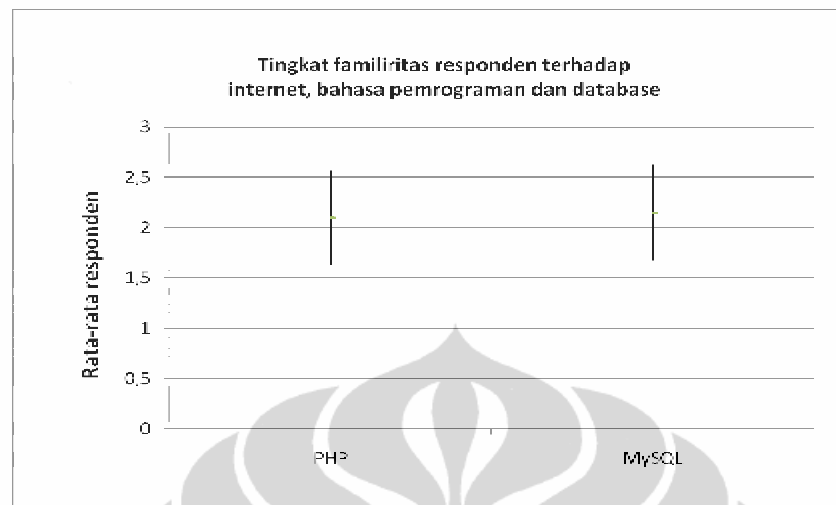
6. Tampilan detail tanaman pada sistem baik	1	3	14	2
7. Data koleksi dari mahasiswa baik	0	4	14	2
8. Detail koleksi dari mahasiswa baik	1	2	15	2
9. Tampilan peta tanaman secara umum baik	1	1	15	3
10. Pencarian nama/nama latin baik	0	3	15	2
11. Data yang dilaporkan sudah baik	1	2	15	2
12. Pengetahuan anda mengenai <i>tanaman</i> baik	1	8	9	2
13. Desain tampilan koleksi baik	2	1	14	3
14. Pemilihan <i>family, genus</i> , nama tanaman pada koleksi	1	1	14	4
15. Tampilan detail koleksi baik	1	2	16	1
16. Penampilan peta tanaman untuk pengguna sistem baik	0	2	15	3
17. Penampilan peta berdasarkan koordinat baik	2	1	14	3
Keterangan: 1 = sangat tidak baik 2 = tidak baik 3 = baik 4 = sangat baik				

Dari data tersebut dihitung rata-rata dan standar deviasi-nya, yang hasilnya dapat dilihat pada Tabel 4.2. Dalam perhitungan ini, digunakan tingkat keyakinan 95%, dengan rumus $Rata - rata \pm \frac{1,96 \times Standar\ Deviasi}{\sqrt{Populasi}}$.

Tabel 4.2. Hasil perhitungan terhadap tanggapan kuesioner

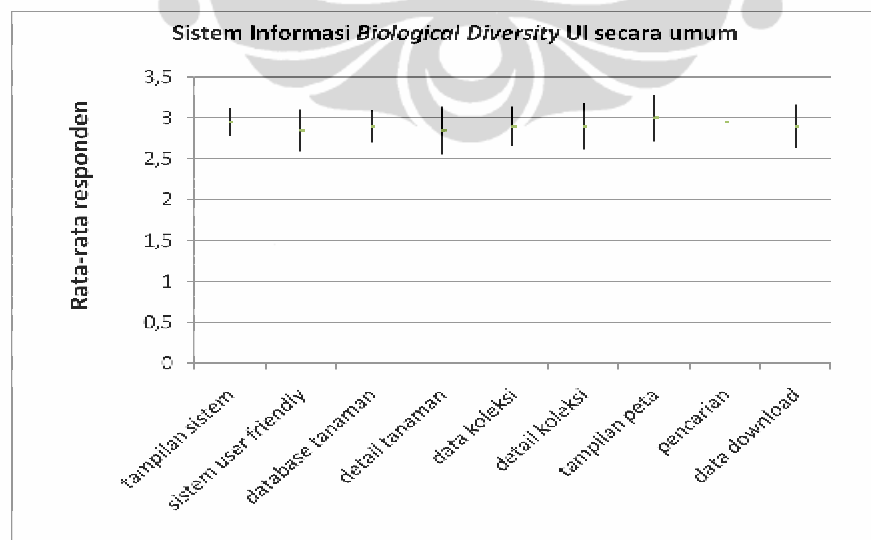
Pertanyaan	Rata-rata
1. Anda familiar dengan PHP	2,1 ± 0,469
2. Anda familiar dengan MySQL	2,15 ± 0,477
3. Desain tampilan sistem baik	2,95 ± 0,173
4. Sistem <i>user friendly</i>	2,85 ± 0,257
5. Database tanaman tersimpan baik	2,9 ± 0,196
6. Tampilan detail tanaman pada sistem baik	2,85 ± 0,294
7. Data koleksi dari mahasiswa baik	2,9 ± 0,242
8. Detail koleksi dari mahasiswa baik	2,9 ± 0,281
9. Tampilan peta tanaman secara umum baik	3 ± 0,284
10. Pencarian nama/nama latin baik	2,95 ± 0,224
11. Data yang dilaporkan sudah baik	2,9 ± 0,265
12. Pengetahuan anda mengenai <i>tanaman</i> baik	2,6 ± 0,33
13. Desain tampilan koleksi baik	2,9 ± 0,345
14. Pemilihan <i>family, genus</i> , nama tanaman pada koleksi	3,05 ± 0,301
15. Tampilan detail koleksi baik	2,85 ± 0,257
16. Penampilan peta tanaman untuk pengguna sistem baik	3,05 ± 0,224
17. Penampilan peta berdasarkan koordinat baik	2,9 ± 0,345

Hasil perhitungan di atas digambarkan ke dalam tiga buah grafik. Masing-masing grafik memuat hasil untuk satu kelompok pertanyaan.



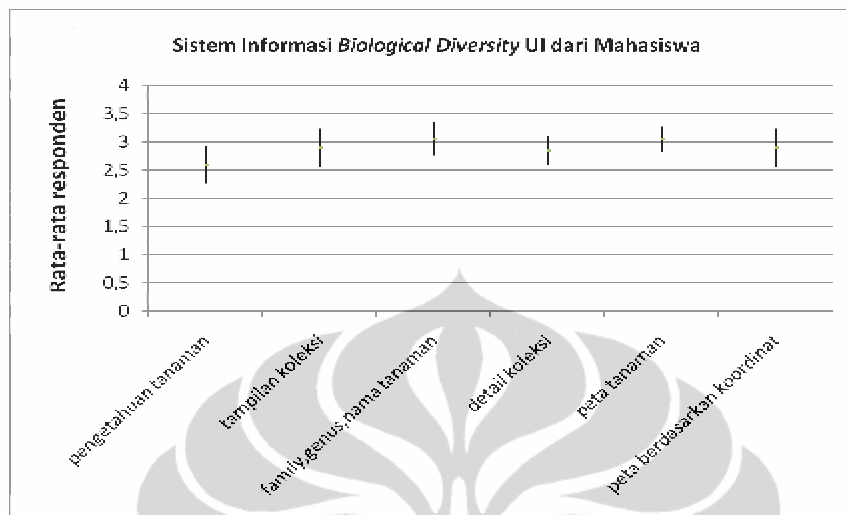
Gambar 4.10. Grafik tanggapan terhadap tingkat familiar dengan bahasa pemrograman dan database

Dari Gambar 4.10. data tanggapan terhadap PHP dan MySQL yang digunakan, terlihat bahwa rata pengguna kurang familiar dengan PHP dan MySQL (rata-rata skala 2 dari skala 4). PHP yang merupakan bahasa pemrograman yang banyak digunakan untuk membuat aplikasi dan sistem informasi yang berhubungan dengan user. MySQL digunakan untuk database sistem.



Gambar 4.11. Grafik tanggapan sistem informasi *biological diversity* secara umum

Dari Gambar 4.11. data tanggapan terhadap sistem informasi *biological diversity* secara umum, terlihat bahwa rata-rata pengguna menilai sistem informasi baik (rata-rata skala 3 dari skala 4).



Gambar 4.12. Grafik tanggapan sistem informasi *biological diversity* dari mahasiswa

Dari Gambar 4.12. data tanggapan terhadap sistem informasi *biological diversity* dari mahasiswa mengenai pengetahuan tanaman tidak baik (rata-rata skala 2 dari skala 4), untuk rata-rata dari mahasiswa terhadap tampilan koleksi, *family*, *genus*, nama tanaman, detail koleksi, peta tanaman, peta berdasarkan koordinat menilai sistem informasi baik (rata-rata skala 3 dari skala 4).

4.3. Pengembangan Masa Depan

Dari hasil pengujian dan analisa, dapat disimpulkan bahwa dalam sistem informasi yang dibuat masih dapat dikembangkan dan dioptimasi di masa yang akan datang. Untuk data tanaman, data tanaman masih perlu ditambahkan baik untuk *family*, *genus*, nama latin/*spesies*, nama tanaman dan keterangan. Untuk peta tanaman juga perlu ditambahkan dan diperbaiki dari sisi tampilan dan agar dapat dipakai menggunakan GPS untuk menentukan koordinat masing-masing mahasiswa.

BAB 5

PENUTUP

Berikut beberapa kesimpulan yang dapat diambil dari implementasi dan analisa program adalah sebagai berikut:

1. Perancangan dan implementasi sistem informasi *biological diversity* Universitas Indonesia ini diharapkan dapat digunakan oleh para mahasiswa, dosen, dan masyarakat umum untuk membuat sistem yang dapat mencatat semua tanaman yang berada di kampus baru Universitas Indonesia di Depok berdasar perolehan ilmu taksonomi tumbuhan.
2. Pengujian program dilakukan oleh administrator sistem dalam memasukkan koordinat peta lokasi yang ditentukan untuk setiap mahasiswa, mendaftarkan data tanaman pada lokasi dengan koordinat tertentu, mendaftarkan klasifikasi *family*, *genus* tanaman dan nama tanaman. Selanjutnya Pengujian program yang dilakukan oleh mahasiswa hanya untuk melihat data tanaman, dan memasukkan data koleksi tanaman yang didapat pada lokasi koordinat yang telah ditentukan.
3. Pengujian program oleh pengguna dengan mempersilahkan 20 pengguna yang mencoba Sistem memperoleh bahwa sebagian besar tidak familiar dengan bahasa pemrograman PHP dan MySQL (rata-rata skala 2 dari skala 4). Secara umum pengguna menilai sistem informasi baik (rata-rata skala 3 dari skala 4). Mahasiswa berpendapat tampilan untuk koleksi, *family*, *genus*, nama tanaman, detail koleksi, peta tanaman, peta berdasarkan koordinat menilai sistem informasi baik (rata-rata skala 3 dari skala 4).

DAFTAR REFERENSI

- [1] Rhee, S., Kitchener, D., Brown, T., Merrill, R., Dilts, R., & Tighe, S. (February 20, 2004). *Report on Biodiversity and Tropical Forest in Indonesia*. Submitted in accordance with Foreign Assistance Act Section 118/119.
- [2] Vanderlei Perez Canhos, Dora Canhos, & Sidnei Souza. (2004). *Networking Biological Diversity Information: Linking Local to Global Effort*. Centro de Referência em Informação Ambiental-CRIA. ICCC 8th International Conference on Electronic.
- [3] Levelink, Jose., Mawdsley, Amanda., & Rijnberg, Theo. (1997). *For Guided Walks BOGOR BOTANIC GARDEN* (Melani Abdulkadir-Sunito, Penerjemah). Jakarta: SMT Grafika Desa Putera.
- [4] *Workshop on Managing Biodiversity Database: Solution toward Sustainable Global Biodiversity*, <http://undip.ac.id/index.php/workshop-on-managing-biodiversity-database-solution-toward-sustainable-global-biodiversity.html>, akses terakhir tanggal 6 Oktober 2010.
- [5] <http://www.ui.ac.id/id/profile/page/pengantar>, akses terakhir tanggal 5 Desember 2010.
- [6] Nugroho, Bunafit. (2004). *PHP & mySQL dengan Editor Dreamweaver MX*. Yogyakarta: Andi Offset.
- [7] Sutarman. (2003). *Membangun Aplikasi Web dengan PHP dan MySQL*. Yogyakarta: Graha Ilmu.
- [8] Braude, Eric J. (2001). *Software Engineering: An Object-Oriented Perspective*. United States of America: Phoenix Color Corporation.
- [9] Dharmawiyanti, Sri., & Satria Wibowo, Romi. (2003) *Pengantar Unified Modeling Language (UML)*. Kuliah Umum IlmuKomputer.Com. Copyright ©2003 IlmuKomputer.Com
- [10] Hakim, Lukmanul., & Musalini, Uus. (2003). *Cara Mudah Memadukan Web Design dan Web Programing*. Jakarta: Elex Media Komputindo.
- [11] Purwanto, Yudhi. (2001). *Pemrograman Web dengan PHP*. Jakarta: Elex Media Komputindo.
- [12] Munir, Sirojul. (2009). *Diktat PHP & MySQL*. Nurul Fikri.

- [13] <http://www.elated.com/articles/creating-image-maps/>, akses terakhir 23 November 2010.
- [14] Sarinder KKS, Lim LHS, Merican AF & Dimiyati K. (2010). *Information Retrieval from Distributed Relational Biodiversity Databases*. Second International Conference on Computer Research and Development.
- [15] Mohd Taib Wahid, M. Kamruzzaman. (2009). *Implementing and Testing of Spatio-temporal and Object-relational Biodiversity Data Model*. Dhaka, Bangladesh: International Conference on Information and Communication Technology.
- [16] Batini, Carlo., & Scannapieco, Monica. (2006). *Data Quality*. © Springer-Verlag Berlin Heidelberg.
- [17] W. Katsarapong, W. Srisang, K. Jaroensutasinee, and M. Jaroensutasinee. (November 30, 2006). *Thailand National Biodiversity Database System with webMathematica and Google Earth*. World Academy of Science, Engineering and Technology 33 2007.
- [18] *Class Diagram dan Objek Diagram*, <http://cheuw.com/class-diagram-dan-objek-diagram/#more-1002>, akses terakhir tanggal 21 Desember 2010.
- [19] *Desain Consideration for Web-based Application*, <http://www.lukew.com/ff/entry.asp?976>, akses terakhir tanggal 27 Desember 2010.
- [20] Martin, Suzanne. *Effective Visual Communication for Graphical User Interfaces*, http://web.cs.wpi.edu/~matt/courses/cs563/talks/smartin/int_design.html, akses terakhir tanggal 7 Januari 2011.

SURVEY PENGGUNAAN SISTEM INFORMASI BIOLOGICAL DIVERSITY UNIVERSITAS INDONESIA

Dalam rangka mendapat masukan tentang penggunaan Sistem Informasi *Biological Diversity* Universitas Indonesia yang dibangun di atas platform PHP dan database MySQL, penulis dengan rendah hati memohon untuk dapat memberikan masukan dengan mengisi survey dibawah ini.

Bahasa pemrograman PHP adalah bahasa *server-side scripting* yang menyatu dengan HTML untuk membentuk web dinamis. PHP merupakan software yang *Open Source* (gratis) dan mampu lintas *platform*, yaitu dapat digunakan dengan sistem operasi dan web server apapun. MySQL adalah multiuser database yang menggunakan bahasa *Structured Query Language (SQL)*. SQL adalah bahasa standar yang digunakan untuk mengakses database server.

Sistem yang penulis buat adalah sistem untuk mensurvei semua tanaman/tumbuhan yang ada di Kampus Baru Universitas Indonesia dengan mencatat *family*, *genus*, nama, nama latin dan keterangan tanaman dengan tanggung jawab 1 (satu) mahasiswa seluas satu ruangan / 100 m².

Mohon untuk dipilih jawaban yang menurut pendapat anda paling tepat:

Tingkat Familiaritas dengan bahasa Pemrograman				
Pertanyaan	1	2	3	4
1. Anda familiar dengan PHP				
2. Anda familiar dengan MySQL				
Keterangan: 1 = sangat tidak familiar (tidak pernah mendengar) 2 = tidak familiar (pernah mendengar namun sama sekali tidak tahu) 3 = cukup familiar (pernah mendengar, tahu, namun tidak pernah menggunakan) 4 = familiar (pernah mendengar, tahu, dan pernah menggunakan) 5 = sangat familiar (pernah mendengar, tahu, dan sering menggunakan)				

Sistem Informasi Biodiversity UI secara Umum				
Pertanyaan	1	2	3	4
3. Desain tampilan sistem baik				
4. Sistem <i>user friendly</i>				
5. Database tanaman tersimpan baik				
6. Tampilan detail tanaman pada sistem baik				
7. Data koleksi dari mahasiswa baik				
8. Detail koleksi dari mahasiswa baik				
9. Tampilan peta tanaman secara umum baik				
10. Pencarian nama/nama latin baik				

11. Data yang dilaporkan baik				
Keterangan: 1 = sangat tidak baik 2 = tidak baik 3 = cukup baik 4 = baik 5 = sangat baik				

Sistem Informasi Biodiversity UI dari Mahasiswa				
Pertanyaan	1	2	3	4
12. Pengetahuan anda mengenai <i>tanaman</i> baik				
13. Desain tampilan koleksi baik				
14. Pemilihan <i>family</i> , <i>genus</i> , nama tanaman pada koleksi				
15. Tampilan detail koleksi baik				
16. Penampilan peta tanaman untuk pengentri baik				
17. Penampilan peta berdasarkan koordinat baik?				
Keterangan: 1 = sangat tidak baik 2 = tidak baik 3 = cukup baik 4 = baik 5 = sangat baik				

Catatan:

- Lokasi koordinat mahasiswa telah ditentukan oleh Sistem.