



LAPORAN PRAKTEK KERJA LAPANGAN

**ANALISA DAN PEMBUATAN MODUL ADMIN PADA
SISTEM INFORMASI ALUMNI TRACERSTUDY
FAKULTAS PETERNAKAN UNIVERSITAS UDAYANA**

Oleh:

GEDE ARIESTA KRISNAYANA

NIM : 1308605065

Pembimbing:

Dra. Luh Gede Astuti, M.Kom

Program Studi Teknik Informatika

Jurusan Ilmu Komputer

Fakultas Matematika Dan Ilmu Pengetahuan Alam

Universitas Udayana

2016

HALAMAN PENGESAHAN

PEMBUATAN BACKEND PADA SISTEM INFORMASI ALUMNI TRACERSTUDY FAKULTAS PETERNAKAN UNIVERSITAS UDAYANA

Oleh:

Gede Ariesta Krisnayana

NIM : 1308605065

Bukit Jimbaran, 06 Juni 2016

Menyetujui,

Dosen Pembimbing

Pembimbing Lapangan

Dra. Luh Gede Astuti, M.Kom.

NIP. 196401141994022001

Dr.Drh. I Gusti Agung Arta P., Msi

NIP. 196211301989031003

Penguji

Mengetahui,

Ketua Jurusan Ilmu Komputer

FMIPA Universitas Udayana

Agus Muliantara, S.Kom., M.Kom.

NIP. 19800616 200501 1 001

KATA PENGANTAR

Puji syukur penulis panjatkan kehadapan Tuhan Yang Maha Esa, karena berkat rahmat dan karunia-Nya, Laporan Praktek Kerja Lapangan yang berjudul “Analisa dan Pembuatan Modul Admin pada Sistem Informasi Alumni Tracerstudy Fakultas Peternakan Universitas Udayana” ini dapat diselesaikan tepat pada waktunya. Secara khusus penulis mengucapkan terima kasih dan penghargaan kepada berbagai pihak yang telah membantu laporan ini, yaitu :

1. Bapak Agus Muliantara, S.Kom, M.Kom., selaku Ketua Jurusan Ilmu Komputer yang telah memberikan dukungan dalam pelaksanaan Praktek Kerja Lapangan ini.
2. Dra. Luh Gede Astuti, M.Kom., selaku dosen pembimbing yang telah membantu dalam pembuatan program dan penyusunan laporan Praktek Kerja Lapangan ini.
3. Dr.Drh. I Gusti Agung Arta P., Msi selaku dosen pembimbing lapangan yang telah membimbing selama kegiatan Praktek Kerja Lapangan berlangsung.
4. Teman-teman di Jurusan Ilmu Komputer yang telah memberikan dukungan moral dalam penyelesaian laporan ini.
5. Semua pihak yang telah memberi dukungan sehingga laporan ini dapat diselesaikan sesuai dengan waktu yang ditentukan.

Pada akhirnya penulis berharap agar adanya perbaikan pada Laporan, Penulis menyadari laporan ini masih jauh dari kesempurnaan. Untuk itu dengan segala kerendahan hati, penulis mengharapakan kritik dan saran dari pembaca sehingga nantinya dapat memperbaiki laporan ini dan mengembangkannya di kemudian hari.

Jimbaran, 05 Desember 2016

Penulis

DAFTAR ISI

| | Halaman |
|--|---------|
| HALAMAN PENGESAHAN | ii |
| KATA PENGANTAR..... | ii |
| DAFTAR ISI | iv |
| DAFTAR TABEL | vi |
| DAFTAR GAMBAR..... | vii |
| DAFTAR LAMPIRAN | viii |
| BAB I PENDAHULUAN | 1 |
| 1.1 Latar Belakang..... | 1 |
| 1.2 Tujuan..... | 2 |
| 1.3 Manfaat..... | 2 |
| 1.3.1 Manfaat Bagi Penulis..... | 2 |
| 1.3.2 Manfaat Bagi Instansi PKL | 2 |
| 1.4 Waktu Dan Tempat Pelaksanaan | 2 |
| BAB II GAMBARAN UMUM | 4 |
| 2.1 Sejarah Fakultas Peternakan | 4 |
| 2.2 Visi Fakultas Peternakan | 5 |
| 2.3 Misi Fakultas Peternakan..... | 5 |
| 2.4 Tujuan Fakultas Peternakan..... | 6 |
| BAB III KAJIAN PUSTAKA | 8 |
| 3.1 Sistem Informasi | 8 |
| 3.2 Komponen Sistem Informasi | 9 |
| 3.3 Elemen Sistem Informasi..... | 11 |
| 3.4 Metode System Development Life Cycle (SDLC) | 12 |
| 3.5 UML (Unified Modeling Language) | 12 |
| 3.5.1 Use Case Diagram | 14 |
| 3.5.2 Activity Diagram | 15 |
| 3.5.3 Class Diagram..... | 16 |
| 3.6 Entity Relationship Diagram (ERD)..... | 17 |
| 3.7 HTML (Hyper Text Markup Language)..... | 20 |
| 3.8 CSS (Cascading Style Sheet)..... | 20 |
| 3.9 PHP : Hypertext Preprocessor | 21 |
| 3.10 JavaScript | 22 |
| 3.11 MySQL | 22 |

| | | |
|---|--|-----------|
| 3.12 | SQL (Structured Query Language) | 23 |
| 3.13 | CodeIgniter..... | 23 |
| 3.13.1 | Konsep MV..... | 23 |
| 3.14 | Framework..... | 23 |
| 3.15 | XAMPP..... | 23 |
| 3.16 | Sublime Text Editor..... | 23 |
| BAB IV PELAKSANAAN PKL | | 28 |
| 4.1 | Gambaran Umum Sistem Informasi Alumni..... | 28 |
| 4.2 | Pengembangan Sistem..... | 28 |
| 4.3 | Analisis Kebutuhan Sistem | 29 |
| 4.4 | Perancangan Sistem..... | 29 |
| 4.4.1 | Use Case Diagram | 29 |
| 4.4.2 | Activity Diagram | 30 |
| 4.4.3 | Class Diagram | 32 |
| 4.5 | ERD (Entitiy Relationship Diagram) | 30 |
| 4.6 | Implementasi | 33 |
| 4.7 | Testing | 334 |
| 4.8 | Pengujian Sistem | 41 |
| BAB V KESIMPULAN DAN SARAN | | 43 |
| 5.1 | Kesimpulan..... | 43 |
| 5.2 | Saran..... | 43 |
| DAFTAR PUSTAKA..... | | 44 |
| LAMPIRAN | | 45 |

DAFTAR TABEL

Halaman

| | |
|---|-----|
| Tabel 3.1 Simbol – simbol <i>use case diagram</i> | 14 |
| Tabel 3.2 Simbol – simbol diagram aktivitas | 166 |
| Tabel 3.3 Simbol – simbol diagram kelas | 17 |
| Tabek 3.4 Simbol – simbol ERD | 178 |
| Tabel 4.1 Tabel Pengujian Sistem Dengan Metode Black Box..... | 41 |

DAFTAR GAMBAR

| | Halaman |
|--|---------|
| Gambar 3.1 Konsep MVC..... | 16 |
| Gambar 3.2 Tahapan model <i>waterfall</i> | 27 |
| Gambar 4.1 <i>Use Case Diagram</i> Admin | 29 |
| Gambar 4.2 <i>Use Case Diagram</i> login Admin | 30 |
| Gambar 4.3 Proses Manajemen Alumni..... | 30 |
| Gambar 4.4 <i>Class Diagram</i> Sistem Informasi Alumni | 32 |
| Gambar 4.5 <i>Entity Relationship Diagram</i> Sistem Informasi alumni | 33 |
| Gambar 4.6 <i>Phpmyadmin</i> | 386 |
| Gambar 4.7 tampilan home | 38 |
| Gambar 4.8 login daftar alumni | 39 |
| Gambar 4.9 <i>Form sign up alumni</i> | 39 |
| Gambar 4.10 form login admin | 40 |
| Gambar 4.11 Lihat Data Alumni | 39 |
| Gambar 4.12 Tambah Data Alumni | 40 |
| Gambar 4.13 Edit Data Alumni..... | 41 |
| Gambar 4.14 Hapus Data Alumni | 41 |

DAFTAR LAMPIRAN

| | Halaman |
|---|---------|
| Lampiran 1. Form aktivitas harian | 45 |

BAB I

PENDAHULUAN

1.1 Latar Belakang

Setiap tahunnya Fakultas Peternakan Universitas Udayana menamatkan ratusan mahasiswa dari berbagai jurusan. Setelah tamat dari kuliah, data atau informasi mengenai alumni sulit di dapatkan. Dan komunikasi di antara alumni tidak berjalan dengan baik, sehingga dengan adanya sistem informasi alumni berbasis web ini, alumni dapat berkomunikasi dengan baik dan dapat bertukar informasi.

Penerapan sistem informasi menjadi begitu penting untuk menunjang kegiatan kerja. Dengan perkembangan teknologi informasi kita dapat membangun sebuah sistem informasi yang bertujuan untuk membantu meningkatkan pekerjaan lebih mudah untuk mengakses data. Data alumni merupakan salah satu contoh dari suatu sistem informasi yang dirancang untuk dapat membantu pekerjaan dari suatu instansi atau perusahaan baik dalam mengolah data sampai memberikan data secara lengkap lewat tersedianya layanan informasi berbasis web. Namun demikian, sejauh ini belum ada penerapan sistem informasi pengolahan data alumni berbasis web di Fakultas Peternakan yang tentunya dapat membantu pekerjaan lebih mudah dalam mengolah, memberikan dan menampilkan data. Sehubungan dengan hal ini, untuk lebih mempermudah memperoleh informasi dari data alumni Fakultas Peternakan Universitas Udayana. Diperlukan suatu sistem informasi yang dapat menginformasikan data secara akurat dari para alumni yang ada. Guna mendapatkan informasi yang akurat dan mudah dalam pengaksesan datanya, maka informasi dapat dirangkum dalam sebuah sistem informasi berbasis web dan perlu juga dibangun sebuah sistem informasi yang tentunya dapat mempermudah proses akses data tersebut pada database.

Fakultas Peternakan Universitas Udayana sudah memiliki system informasi alumni yang bernama tracerstudy, tetapi pada system tersebut belum memiliki modul admin, sehingga data yang telah ada dalam system tidak bisa di edit. Berdasarkan paparan

diatas maka penulis berkeinginan untuk membuat suatu modul admin untuk melengkapi system informasi tracerstudy dengan mengangkat judul “Analisa dan Pembuatan Modul Admin pada Sistem Informasi Alumni Tracerstudy Fakultas Peternakan Universitas Udayana”

1.2 Tujuan

Adapun tujuan yang ingin dicapai dari pelaksanaan kegiatan praktek kerja lapangan ini adalah menambah suatu modul admin ke dalam Sistem Informasi Alumni yang dapat mengelola data alumni mahasiswa,yang ada di Fakultas Peternakan Universitas Udayana.

1.3 Manfaat

Adapun manfaat yang didapatkan dari pelaksanaan kegiatan praktek kerja lapangan ini dibagi menjadi dua bagian, yaitu manfaat bagi penulis dan manfaat bagi instansi.

1.3.1 Manfaat Bagi Penulis

Adapun manfaat yang didapatkan bagi penulis dari pelaksanaan kegiatan praktek kerja lapangan ini adalah :

1. Mengetahui alur perancangan dan pengelolaan Sistem Informasi Alumni Tracerstudy Fakultas Peternakan Universitas Udayana.
2. Menambah ilmu dan pengetahuan terkait dunia kerja.

1.3.2 Manfaat Bagi Instansi PKL

Adapun manfaat yang didapatkan bagi instansi dari pelaksanaan kegiatan praktek kerja lapangan ini adalah :

1. Mendapatkan data pasti dari Alumni mahasiswa Fakultas Peternakan Universitas Udayana.
2. Mempermudah proses pencarian data dan informasi alumni mahasiswa Fakultas Peternakan Universitas Udayana..

1.4 Waktu Dan Tempat Pelaksanaan

Waktu pelaksanaan Praktek Kerja Lapangan ini dilakukan selama tiga bulan yaitu dimulai dari 8 september 2016 hingga 25 November 2016. Pelaksanaan Praktek Kerja Lapangan disesuaikan dengan jam kuliah penulis yaitu pukul 10.00 – 16.00 WITA (Senin – Kamis) dan 10.00 – 15.00 WITA (Jumat) Pelaksanaan Praktek Kerja Lapangan yang penulis laksanakan yatu berlokasi di Fakultas Peternakan Universitas Udayana yang beralamat di Bukit Jimbaran, Badung.

BAB II

GAMBARAN UMUM

2.1 Sejarah Fakultas Peternakan

Fakultas Peternakan Universitas Udayana yang pada awalnya bernama Fakultas Kedokteran Hewan dan Peternakan didirikan tahun 1962 (SK Mendikbud No. 104, tanggal 19 Agustus 1962), dan berdasarkan Surat Keputusan Presiden Republik Indonesia No. 8, tanggal 31 Januari 1963, yang sekaligus sebagai SK pendirian Universitas Udayana. Pada saat itu Universitas Udayana baru menaungi empat fakultas yaitu Fakultas Sastra, Fakultas Kedokteran, Fakultas Kedokteran Hewan dan Peternakan, serta Fakultas Keguruan dan Ilmu Pendidikan. Fakultas terakhir ini telah memisahkan diri dari Unud dan bernama Sekolah Tinggi Keguruan Ilmu Pendidikan berkedudukan di kota Singaraja, Kabupaten Buleleng.

Berdasarkan Keputusan Presiden No. 62 tahun 1982 tentang Organisasi Universitas Udayana, nama Fakultas Kedokteran Hewan dan Peternakan tidak ada lagi, yang muncul adalah nama Fakultas Peternakan. Dengan demikian sejak tahun 1982 nama Fakultas Peternakan ditetapkan sebagai pengganti nama Fakultas Kedokteran Hewan dan Peternakan.

Sejak berdirinya FKHP Unud, jurusan yang ada hanyalah jurusan Peternakan. Pada tahun 1980 dikembangkan Jurusan Kedokteran Hewan yang kemudian berdiri sendiri menjadi Program Studi Kedokteran Hewan (PSKH) dan sekarang Fakultas Kedokteran Hewan (FKH).

Selanjutnya, melalui SK Menteri Pendidikan dan Kebudayaan RI No. 0562/O/1983, tentang jenis dan jumlah jurusan pada fakultas-fakultas di lingkungan Unud, dan SK. Dirjen. Pendidikan Tinggi No. 55/Dikti/Kep/1984, tanggal 31 Juli 1984 tentang jenis dan jumlah program studi di lingkungan Unud, maka Fakultas Peternakan memiliki dua jurusan dan sembilan bidang studi yang didukung oleh laboratorium-laboratorium. Mengacu kepada SK Menteri Pendidikan dan Kebudayaan RI No. 0311/U/1994 tentang kurikulum yang berlaku secara nasional Program Sarjana Bidang Ilmu-Ilmu Pertanian, Fakultas

memiliki dua jurusan (Produksi Ternak serta Nutrisi dan Makanan Ternak). Keduanya telah diakreditasi oleh Badan Akreditasi Nasional Perguruan Tinggi Depdiknas RI. Untuk Jurusan Nutrisi dan Makanan Ternak dengan Nomor 12/BAN-PT/Ak-VII/S1/IV/2004, tanggal 16 April 2004 dan untuk Jurusan Produksi Ternak dengan nomor 017/BAN-PT/Ak-VII/S1/V/2004 tanggal 07 Mei 2004, masing-masing memperoleh peringkat A.

Melalui Surat Ijin Dikti tentang penyelenggaraan Program studi Sosial Ekonomi Peternakan No. 1871/D/T/2002 tanggal 3 September 2002, telah terbentuk Program Studi Sosial Ekonomi Peternakan pada jenjang pendidikan S1. PS ini masih berada di bawah naungan Jurusan Produksi Ternak. Sedangkan Lab.Teknologi Hasil Ternak masih sebagai embrio Program Studi Teknologi Hasil Ternak.

Selanjutnya, dengan menindaklanjuti hasil tracer studi pada alumni dan stakeholder fakultas peternakan, maka dipandang perlu untuk menggabungkan semua jurusan yang ada di Fakultas Peternakan Unud menjadi satu Program Studi. Setelah melalui berbagai tahap maka akhirnya penggabungan Jurusan dan Program Studi di Fakultas Peternakan Universitas Udayana dapat terwujud, dengan Surat Dirjen Dikti Nomor : 113/Dikti/Kep/2007 tanggal 31 Agustus 2007 dengan nama Program Studi yang baru yaitu "Program Studi Peternakan". Terhitung mulai 17 Juni 2016, Program Studi Peternakan memperoleh nilai akreditasi A dari BAN PT

2.2 Visi Fakultas Peternakan

1. Terwujudnya Fakultas Peternakan berkualitas yang mampu menghasilkan lulusan bermoral dan berakhlak, mandiri, menguasai IPTEKS peternakan, berbudaya, dan berjiwa kewirausahaan.
2. Guna mewujudkan keadaan yang dicita-citakan sebagaimana disebutkan dalam visi Fapet UNUD.

2.3 Misi Fakultas Peternakan

Sesuai dengan visi tersebut di atas, misi dari Fakultas Peternakan Unud adalah:

1. Menyelenggarakan pendidikan akademis yang berbasis manajemen mutu terpadu dalam empat cabang ilmu, yakni : Nutrisi dan Makanan Ternak, Produksi Ternak, Teknologi Hasil Ternak dan Sosial Ekonomi Peternakan.
2. Meningkatkan kualitas pendidikan berkelanjutan, berorientasi kepada paradigma baru penataan pendidikan tinggi (otonomi, akuntabilitas, kualitas, evaluasi dan akreditasi) untuk menghasilkan lulusan yang memiliki relevansi dan kompetensi tinggi terhadap perkembangan IPTEKS dan berjiwa kewirausahaan.
3. Menyelenggarakan penelitian dan pengabdian pada masyarakat yang menyentuh kepentingan stakeholders, sehingga IPTEKS yang dikembangkan tetap tergayut dengan pertumbuhan dan kebutuhan lapangan kerja.
4. Mengembangkan infrastruktur pendidikan yang andal untuk menyelenggarakan fungsi Tri Dharma Perguruan Tinggi.
5. Meningkatkan kemampuan manajemen organisasi dan kepemimpinan yang berorientasi kepada pelayanan berkualitas, profesional, demokratis dan berjiwa kewirausahaan.
6. Menyelenggarakan kerjasama di berbagai bidang dengan berbagai pihak, baik dalam maupun luar negeri untuk meningkatkan mutu penyelenggaraan Tri Dharma Perguruan Tinggi.

2.4 Tujuan Fakultas Peternakan Universitas Udayana

Mengacu kepada visi dan misi Fakultas Peternakan Universitas Udayana dan berpedoman pada Peraturan Pemerintah Republik Indonesia No. 60 Tahun 1999 Bab. II ps.2, SK. Mendikbud RI No. 0311/U/1991 dan Statuta Universitas Udayana maka Fakultas Peternakan Universitas Udayana memiliki tujuan untuk menghasilkan sarjana strata satu yang memiliki kualifikasi sebagai berikut:

1. Menghasilkan SDM yang memiliki kompetensi tinggi dalam penguasaan IPTEKS sehingga mampu memikirkan, merencanakan dan mengembangkan IPTEKS; berjiwa wirausaha yang tangguh; terampil dan professional dalam bidang agribisnis peternakan sehingga mampu menciptakan lapangan kerja untuk meningkatkan kesejahteraan masyarakat.
2. Menghasilkan IPTEKS untuk memanfaatkan sumber daya secara optimal dan berkelanjutan melalui kegiatan riset dan pengembangannya.
3. Mewujudkan kondisi masyarakat akademis yang kondusif, berkualitas, professional dan mandiri melalui penerapan sistem manajemen pendidikan yang bermutu, sehat, transparan dan demokratis.

Menghasilkan SDM yang mampu berkarya secara profesional, memiliki sifat dan sikap yang baik dalam berkarya dan bermasyar

BAB III

KAJIAN PUSTAKA

3.1 Sistem Informasi

Sistem informasi yaitu suatu sistem yang menyediakan informasi untuk manajemen dalam mengambil keputusan dan juga untuk menjalankan operasional perusahaan, di mana sistem tersebut merupakan kombinasi dari orang-orang, teknologi informasi dan prosedur-prosedur yang terorganisasi. Biasanya suatu perusahaan atau badan usaha menyediakan semacam informasi yang berguna bagi manajemen. Ada beragam definisi sistem informasi, sebagaimana tercantum di bawah ini.

1. Menurut Alter (1992)

Sistem informasi adalah kombinasi antara prosedur kerja, informasi, orang, dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi.

2. Menurut Gelinis, Oram, dan Wiggins (1990)

Sistem informasi adalah suatu sistem buatan manusia yang secara umum terdiri atas sekumpulan komponen berbasis komputer dan manual yang dibuat untuk menghimpun, menyimpan, dan mengelola data serta menyediakan informasi keluaran kepada pemakai.

3. Menurut Turban, McLean, dan Wetherbe (1999)

Sebuah sistem informasi mengumpulkan, memproses, menyimpan, menganalisis, dan menyebarkan informasi untuk tujuan yang spesifik.

Dari berbagai definisi tersebut, dapat disimpulkan bahwa sistem informasi mencakup sejumlah komponen (manusia, komputer, teknologi informasi, dan prosedur kerja), ada sesuatu yang diproses (data menjadi informasi), dan dimaksudkan untuk mencapai suatu sasaran atau tujuan. Informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian

yang nyata yang digunakan untuk pengambilan keputusan. Informasi merupakan data yang telah diklasifikasikan atau diolah atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan.

Sistem informasi dalam suatu organisasi dapat dikatakan sebagai suatu sistem yang menyediakan informasi bagi semua tingkatan dalam organisasi tersebut kapan saja diperlukan. Sistem ini menyimpan, mengambil, mengubah, mengolah dan mengkomunikasikan informasi yang diterima. Sebagai contoh : Perusahaan toko buku mempunyai sistem informasi yang menyediakan informasi penjualan buku-buku setiap harinya, serta stok buku-buku yang tersedia, dengan informasi tersebut, seorang manajer bisa membuat keputusan, stok buku apa yang harus segera mereka sediakan untuk toko buku mereka, manajer juga bisa tahu buku apa yang paling laris dibeli konsumen, sehingga mereka bisa memutuskan buku tersebut jumlah stoknya lebih banyak dari buku lainnya.

3.2 Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (*building block*), yang terdiri dari komponen *input*, komponen model, komponen *output*, komponen teknologi, komponen *hardware*, komponen *software*, komponen basis data, dan komponen kontrol. Semua komponen tersebut saling berinteraksi satu dengan yang lain membentuk suatu kesatuan untuk mencapai sasaran.

1. Komponen *input*

Input mewakili data yang masuk kedalam sistem informasi. *Input* disini termasuk metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Komponen model

Komponen ini terdiri dari kombinasi prosedur, logika, dan model matematika yang akan memanipulasi data *input* dan data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.

3. Komponen *output*

Hasil dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua pemakai sistem.

4. Komponen teknologi

Teknologi merupakan “*Tool Box*” dalam sistem informasi, Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran, dan membantu pengendalian dari sistem secara keseluruhan.

5. Komponen *hardware*

Hardware berperan penting sebagai suatu media penyimpanan *vital* bagi sistem informasi berfungsi sebagai tempat untuk menampung *database* atau lebih mudah dikatakan sebagai sumber data dan informasi untuk memperlancar dan mempermudah kerja dari sistem informasi.

6. Komponen *software*

Software berfungsi sebagai tempat untuk mengolah, menghitung dan memanipulasi data yang diambil dari *hardware* untuk menciptakan suatu informasi.

7. Komponen basis data

Basis data (*database*) merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya.

8. Komponen kontrol

Banyak hal yang dapat merusak sistem informasi, seperti bencana alam, api, temperatur, air, debu, kegagalan-kegagalan sistem itu sendiri, ketidak efisienan, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

3.3 Elemen Sistem Informasi

Sistem informasi merupakan kumpulan *elemen-elemen* yang terdiri dari orang, prosedur, perangkat keras, perangkat lunak, basis data, jaringan komputer dan komunikasi data. Semua elemen ini merupakan komponen fisik.

1. Orang

Orang yang di maksudkan yaitu *operator* komputer, analis sistem, *programmer*, *personal data entry*, dan manajer sistem informasi/EDP

2. Prosedur

Prosedur merupakan elemen fisik. Hal ini di sebabkan karena prosedur disediakan dalam bentuk fisik seperti buku panduan dan instruksi. Ada 3 jenis prosedur yang dibutuhkan, yaitu instruksi untuk pemakai, instruksi untuk penyiapan masukan, instruksi pengoperasian untuk karyawan pusat komputer.

3. Perangkat keras

Perangkat keras bagi suatu sistem informasi terdiri atas komputer (pusat pengolah, unit masukan/keluaran), peralatan penyiapan data, dan terminal masukan/keluaran.

4. Perangkat lunak

Perangkat lunak dapat dibagi dalam 3 jenis utama:

- a. Sistem perangkat lunak umum, seperti sistem pengoperasian dan sistem *manajemen* data yang memungkinkan pengoperasian sistem komputer.
- b. Aplikasi perangkat lunak umum, seperti model analisis dan keputusan.

- c. Aplikasi perangkat lunak yang terdiri atas program yang secara spesifik dibuat untuk setiap aplikasi.
- 5. Basis data

File yang berisi program dan data dibuktikan dengan adanya media penyimpanan secara fisik seperti *diskette*, *harddisk*, *magnetictape*, dan sebagainya. *File* juga meliputi keluaran tercetak dan catatan lain diatas kertas, *mikro film*, dan lain sebagainya.
- 6. Jaringan komputer

Jaringan komputer adalah sebuah kumpulan komputer, printer dan peralatan lainnya yang terhubung dalam satu kesatuan. Informasi dan data bergerak melalui kabel atau tanpa kabel sehingga memungkinkan pengguna jaringan komputer dapat saling bertukar dokumen dan data.
- 7. Komunikasi data

Komunikasi data adalah merupakan bagian dari telekomunikasi yang secara khusus berkenaan dengan *transmisi* atau pemindahan data dan informasi diantara komputer-komputer dan piranti-piranti yang lain dalam bentuk digital yang dikirimkan melalui media komunikasi data. Data berarti informasi yang disajikan oleh isyarat *digital*. Komunikasi data merupakan bagian *vital* dari suatu sistem informasi karena sistem ini menyediakan infrastruktur yang memungkinkan komputer dapat berkomunikasi satu sama lain.

3.4 Metode Metode System Development Life Cycle (SDLC)

SDLC (Systems Development Life Cycle, Siklus Hidup Pengembangan Sistem) atau Systems Life Cycle (Siklus Hidup Sistem), dalam rekayasa sistem dan rekayasa perangkat lunak, adalah proses pembuatan dan perubahan sistem serta model dan metodologi yang digunakan untuk mengembangkan sistem-sistem tersebut. Konsep ini umumnya merujuk pada sistem komputer atau informasi. SDLC juga merupakan pola yang diambil untuk

mengembangkan sistem perangkat lunak, yang terdiri dari tahap-tahap: rencana(planning), analisis (analysis), desain (design), implementasi (implementation), uji coba (testing) dan pengelolaan (maintenance).[1] Dalam rekayasa perangkat lunak , konsep SDLC mendasari berbagai jenis metodologi pengembangan perangkat lunak. Metodologi-metodologi ini membentuk suatu kerangka kerja untuk perencanaan dan pengendalian pembuatan sistem informasi, yaitu proses pengembangan perangkat lunak. Terdapat 3 jenis metode siklus hidup sistem yang paling banyak digunakan, yakni: siklus hidup sistem tradisional(traditional system life cycle), siklus hidup menggunakan prototyping (life cycle using prototyping), dan siklus hidup sistem orientasi objek (object-oriented system life cycle).

Adapun kegunaan utama dari SDLC adalah mengakomodasi beberapa kebutuhan. Kebutuhan-kebutuhan itu biasanya berasal dari kebutuhan pengguna akhir dan juga pengadaan perbaikan sejumlah masalah yang terkait dengan pengembangan perangkat lunak. Kesemua itu dirangkum pada proses SDLC yang dapat berupa penambahan fitur baru baik itu secara modular maupun dengan proses instalasi baru. Dari proses SDLC juga berapa lama umur sebuah perangkat lunak dapat diperkirakan untuk dipergunakan yang dapat diukur atau disesuaikan dengan kebijakan dukungan dari pengembang perangkat lunak terkait. Setiap pengembang mempunyai strategi yang berlainan, namun demikian pada dasarnya siklus hidup pengembangan sistem informasi terdapat 5 tahapan, yaitu :

1. Perencanaan Sistem (Systems Planning)
2. Analisis Sistem (System Analysis)
3. Perancangan Sistem (System Design)
4. Implementasi Sistem (System Implementation)
5. Penggunaan sistem (System Utilization)

3.5 UML (Unified Modeling Language)

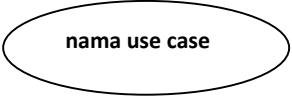
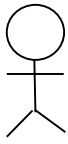

Menurut Nugroho (2010:6), "*UML (Unified Modeling Language)* adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma (berorientasi objek)." Pemodelan (*modeling*) sesungguhnya digunakan untuk

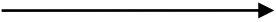
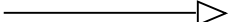

penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami. Menurut Nugroho (2010:10), Sesungguhnya tidak ada batasan yang tegas diantara berbagai konsep dan konstruksi dalam *UML*, tetapi untuk menyederhanakannya, kita membagi sejumlah besar konsep dan dalam *UML* menjadi beberapa *view*. Suatu *view* sendiri pada dasarnya merupakan sejumlah konstruksi pemodelan *UML* yang merepresentasikan suatu aspek tertentu dari sistem atau perangkat lunak yang sedang kita kembangkan. Pada peringkat paling atas, *view* sesungguhnya dapat dibagi menjadi tiga area utama, yaitu: klasifikasi struktural (*structural classification*), perilaku dinamis (*dynamic behaviour*), serta pengolahan atau manajemen model (*model management*).

3.5.1 Use Case Diagram

Use case diagram merupakan pemodelan untuk kelakuan atau *behavior* sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat.

Tabel 3.1 Simbol – simbol *use case diagram*.

| Simbol | Deskripsi |
|---|---|
| <p><i>Use Case</i></p>  <p>nama use case</p> | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau sektor. |
| <p>Aktor / <i>actor</i></p>  <p>nama aktor</p> | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang dibuat di luar sistem. Jadi, walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang. |
| <p>Asosiasi / <i>association</i></p>  | Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> lainnya atau <i>use case</i> memiliki interaksi dengan aktor. |





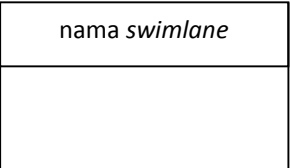
| | |
|---|---|
| <p>Ekstensi / <i>extend</i></p> <p><<extend>></p>  | <p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> lainnya, dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan tersebut, mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek.</p> |
| <p>Generalisasi / <i>generalization</i></p>  | <p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p> |
| <p><i>Include</i></p> <p><<include>></p>  | <p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.</p> |

3.5.2 Activity Diagram

Diagram aktivitas atau *activity diagram* menggambarkan aliran kerja (*workflow*) atau aktivitas dari sebuah sistem atau proses bisnis yang ada pada perangkat lunak. Diagram aktivitas banyak digunakan untuk mendefinisikan hal – hal berikut :

1. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokkan tampilan dari sistem / *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
3. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.

Tabel 3.2 Simbol – simbol diagram aktivitas

| Simbol | Deskripsi |
|--|--|
| Status awal  | Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal. |
| Aktivitas  | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja. |
| Percabangan / <i>decision</i>  | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu. |
| Status akhir  | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir. |
| <i>Swimlane</i>  | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi. |



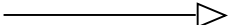

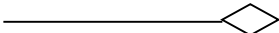
3.5.3 Class Diagram

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas – kelas yang akan dibuat untuk membangun sistem. Pada suatu kelas memiliki beberapa komponen yaitu sebagai berikut.

1. Atribut merupakan variabel – variabel yang bersifat global pada kelas tersebut.
2. Method adalah operasi atau fungsi yang dimiliki oleh suatu kelas.

Untuk membangun sebuah rancangan sistem dengan diagram kelas, maka perlu diperhatikan bahwa diagram kelas memiliki beberapa simbol – simbol, yaitu sebagai berikut.

Tabel 3.3 Simbol – simbol diagram kelas

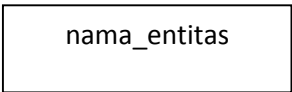
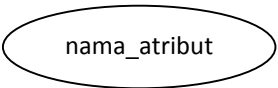
| Simbol | Deskripsi |
|---|---|
| <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Kelas</div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 2px;">nama_kelas</div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 2px;">+atribut</div> <div style="border: 1px solid black; padding: 2px;">+operasi</div> | Kelas pada struktur sistem. Suatu kelas memiliki atribut dan operasi. Masing – masing atribut dan operasi memiliki jenis akses yang berbeda – beda, yaitu public, protected, dan private. |
| Asosiasi / <i>association</i>  | Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> . |
| Asosiasi berarah / <i>directed association</i>  | Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> . |
| Generalisasi  | Relasi antarkelas dengan makna generalisasi-spesialisasi (umum-khusus). |
| Kebergantungan / <i>dependency</i>  | Relasi antarkelas dengan makna kebergantungan antar kelas. |
| Agregasi / <i>aggregation</i>  | Relasi antar kelas dengan makna semua bagian (<i>whole-part</i>). |

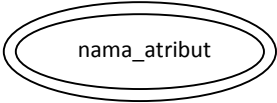
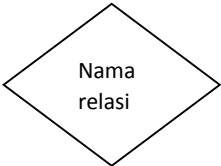
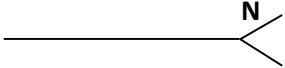
3.6 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak. Sehingga jelas bahwa ERD berbeda dengan DFD yang merupakan suatu model jaringan fungsi yang akan dilaksanakan oleh sistem, sedangkan ERD merupakan model jaringan data yang menekankan pada struktur-struktur relationship

data. Entity Relationship Diagram adalah notasi grafik dari sebuah model data atau sebuah model jaringan yang menjelaskan tentang data yang tersimpan (*storage data*) dalam sistem secara abstrak. Diagram hubungan entitas tidak menyatakan bagaimana memanfaatkan data, membuat data, mengubah data dan menghapus data.

Tabel 3.4 Simbol – Simbol *Entity Relationship Diagram (ERD)*

| Nama | Simbol | Deskripsi |
|-------------------------|---|--|
| Entitas / <i>Entity</i> |  | Entitas merupakan data inti yang akan disimpan. Penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel |
| Atribut |  | <i>Field</i> atau kolom data yang diperlukan dalam suatu entitas. |
| Atribut kunci primer |  | <i>Field</i> atau kolom data yang diperlukan dalam suatu entitas dan digunakan sebagai kunci akses record yang |

| | | |
|--|---|--|
| | | diinginkan. Kunci primer dapat lebih dari satu kolom, tetapi dengan syarat kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama) |
| Atribut multivalai / <i>multivalue</i> |  | <i>Field</i> atau kolom data dalam suatu entitas yang dapat memiliki nilai lebih dari satu. |
| Relasi |  | Relasi yang menghubungkan antar entitas, biasanya diawali dengan kata kerja. |
| Asosiasi / <i>Association</i> |  | Penghubung antar relasi dan entitas dimana di kedua ujungnya memiliki kardinalitas. Jenis – jenis kardinalitas yaitu <i>one to</i> |

| | | |
|--|--|---|
| | | <i>many, many to one, many to many, one to one.</i> |
|--|--|---|

3.7 HTML (Hyper Text Markup Language)

HTML adalah singkatan dari *HyperText Markup Language* yaitu bahasa pemrograman standar yang digunakan untuk membuat sebuah halaman web, yang kemudian dapat diakses untuk menampilkan berbagai informasi di dalam sebuah penjelajah *web Internet (Browser)*. HTML dapat juga digunakan sebagai *link link* antara file-file dalam situs atau dalam komputer dengan menggunakan *localhost*, atau *link* yang menghubungkan antar situs dalam dunia internet. Supaya dapat menghasilkan tampilan wujud yang terintegrasi Pemformatan hiperteks sederhana ditulis dalam berkas format ASCII sehingga menjadi halaman web dengan perintah-perintah HTML. Berikut adalah contoh dari sintak dasar HTML :

```
<html>
<head>
<title>Nama Dokumen Yang Muncul Dalam Judul, Bukan Pada
Halaman </ title>
</ head>
<body>Ketik Informasi Yang Ingin Ditampilkan Pada
Halaman Di Sini
</ body>
</ html>
```

3.8 CSS (Cascading Style Sheet)

Cascading Style Sheets (CSS) adalah suatu bahasa pengaturan tampilan yang digunakan untuk mengatur tampilan dan bentuk dari sebuah dokumen yang ditulis dalam *markup language*. Pengaplikasian CSS paling umum adalah digunakan untuk mengatur tampilan halaman web yang ditulis dalam HTML/XHTML. CSS dirancang terutama untuk memungkinkan pemisahan terhadap

konten/isi dokumen (yang ditulis dalam HTML atau bahasa *markup* sejenis) dengan pengaturan tampilan dokumen, termasuk *layout*, warna dan huruf. Pemisahan ini dapat meningkatkan aksesibilitas konten, memberikan fleksibilitas dalam pengaturan tampilan, memungkinkan untuk beberapa halaman berbagi tampilan yang sama dan mengurangi kompleksitas dan pengulangan dalam struktur konten. CSS juga memungkinkan untuk menampilkan suatu halaman dengan tampilan berbeda sesuai dengan rendering method yang digunakan seperti *on-screen*, *print*, atau dengan suara (bila menggunakan browser khusus yang berbasis suara). Berikut adalah contoh dari sintaks CSS :

```
<style Type="text/css">
    h1{font-family:tahoma}
    h2{color: blue}
    p{font-size:11pt ; font-style: italic}
</style>
```

3.9 PHP : Hypertext Preprocessor

PHP adalah sebuah bahasa pemrograman yang didesain agar dapat disisipkan dengan mudah ke halaman HTML. PHP memberikan solusi sangat murah (karena gratis digunakan) dan dapat berjalan di berbagai jenis platform. Pada awalnya memang PHP berjalan di sistem UNIX dan variannya, namun kini dapat berjalan dengan lancar di lingkungan sistem operasi Windows. Suatu nilai tambah yang luar biasa karena proses pengembangan program berbasis web dapat dilakukan lintas sistem operasi. Dengan luasnya cakupan sistem operasi yang mampu menjalankan PHP dan ditambah begitu lengkapnya function yang dimilikinya (tersedia lebih dari 400 function di PHP yang sangat berguna) tidak heran jika PHP semakin menjadi tren di kalangan programmer web. Untuk dapat menjalankan script-script PHP, sebuah sistem harus mempunyai Apache Web Server, PHP 4/PHP 5, dan database MySQL. Ketiganya adalah program open source yang tersedia secara gratis di Internet dan dapat berjalan di berbagai platform (Windows maupun UNIX / Linux). Berikut merupakan contoh dari *script* PHP :

```
<?php
    echo "Hello World";
?>
```

Untuk mengetikkan baris kode php, maka diperlukan untuk mengapitnya dengan tag php `<?php` dan diakhiri dengan tutup tag php `?>`, ini disebabkan PHP *engine* tidak akan mengeksekusi skrip php diluar daripada tag php tersebut.

3.10 JavaScript

JavaScript adalah bahasa pemrograman web yang bersifat *Client Side Programming Language*. *Client Side Programming Language* adalah tipe bahasa pemrograman yang pemrosesannya dilakukan oleh *client*. Aplikasi *client* yang dimaksud merujuk kepada web browser seperti *Google Chrome* dan *Mozilla Firefox*. Jenis bahasa pemrograman *Client Side* berbeda dengan bahasa pemrograman *Server Side* seperti PHP, dimana untuk *server side* seluruh kode program dijalankan di sisi *server*. Untuk menjalankan *JavaScript*, kita hanya membutuhkan aplikasi text editor, dan web browser. JavaScript memiliki fitur: *high-level programming language*, *client-side*, *loosely typed*, dan berorientasi objek. Berikut merupakan contoh dari *script JavaScript* :

```
<script language="javascript">
Program javascript
</script>
```

Dalam mengawali perintah javascript yaitu di mulai dengan perintah `<script language="javascript">` dan diakhiri dengan perintah `</script>`.

3.11 MySQL

MySQL merupakan *database* yang dikembangkan dari bahasa SQL (*Structure Query Language*). SQL sendiri merupakan bahasa yang terstruktur yang digunakan untuk interaksi antara *script* program dengan *database server* dalam hal pengolahan data. Dengan SQL, kita dapat membuat tabel yang nantinya akan diisi dengan data, memanipulasi data (misalnya menambah data, menghapus data dan

memperbaharui data), serta membuat suatu perhitungan dengan berdasarkan data yang ditemukan. MySQL merupakan *software* resmi yang dikembangkan oleh perusahaan Swedia bernama MySQL AB, yang waktu itu bernama TcX Data Konsult AB. Pada awalnya MySQL memakai nama mSQL atau “*mini SQL*” sebagai antarmuka yang digunakan, ternyata dengan menggunakan mSQL itu mengalami banyak hambatan, yaitu sangat lambat dan tidak *fleksibel*. Oleh karena itu, Michael Widenius berusaha mengembangkan *interface* yang tersebut hingga ditemukan MySQL. Kala itu, MySQL didistribusikan secara khusus, yakni untuk keperluan nonkomersial bersifat gratis, sedangkan untuk kebutuhan komersial diharuskan membayar lisensi. Barulah sejak versi 3.23.19, MySQL dikategorikan *software* berlisensi GPL, yakni dapat dipakai tanpa biaya untuk kebutuhan apapun.

3.12 SQL (Structured Query Language)

SQL (*Structured Query Language*) merupakan bahasa *query* yang digunakan untuk mengakses *database relasional*. SQL sudah menjadi bahasa *database* standar dan hampir semua sistem *database* memahaminya. SQL terdiri dari berbagai jenis statement. Semuanya didesain agar memungkinkan untuk dapat secara interaktif berhubungan dengan *database*. Penggunaan SQL pada DBMS (*Database Management System*) sudah cukup luas. SQL dapat dipakai oleh berbagai kalangan, misalnya DBA (*Database Administrator*), programmer ataupun pengguna. Hal ini disebabkan karena:

1. SQL sebagai bahasa administrasi *database* Dalam hal ini SQL dipakai oleh DBA untuk menciptakan serta mengendalikan pengaksesan *database*.
2. SQL sebagai bahasa *query interaktif* Pengguna dapat memberikan perintah-perintah untuk mengakses *database* yang sesuai dengan kebutuhannya.
3. SQL sebagai bahasa pemrograman *database*. Pemrogram dapat menggunakan perintah-perintah SQL dalam program aplikasi yang dibuat.
4. SQL sebagai bahasa *client/server* SQL juga digunakan untuk mengimplementasikan sistem *client/ server*. Sebuah *client*

dapat menjalankan suatu aplikasi yang mengakses *database*. Dalam hal ini sistem operasi antara *server* dan *client* bisa berbeda. Di samping hal tersebut di atas SQL juga diterapkan pada internet atau intranet untuk mengakses *database* melalui halaman-halaman web untuk mendukung konsep web dinamis.

Pernyataan SQL dapat dikelompokkan menjadi 2 kelompok yaitu Data Definition Language atau disingkat DDL dan Data Manipulation Language atau disingkat DML.

1. DDL (*Data Definition Language*)

DDL merupakan kelompok perintah yang berfungsi untuk mendefinisikan atribut-atribut database, tabel, atribut (kolom), batasan-batasan terhadap suatu atribut serta hubungan antar tabel. Yang termasuk kelompok DDL ini adalah:

- a. CREATE untuk menciptakan tabel atau indeks
- b. ALTER untuk mengubah struktur tabel
- c. DROP untuk menghapus tabel atau indeks

2. DML (*Data Manipulation Language*)

DML adalah kelompok perintah yang berfungsi untuk memanipulasi data, misalnya untuk pengambilan, penyisipan perubahan dan penghapusan data. Yang termasuk DML adalah:

- a. SELECT untuk memilih data
- b. INSERT untuk menambah data
- c. DELETE untuk menghapus data
- d. UPDATE untuk mengubah data

3.13 CodeIgniter

CodeIgniter adalah sebuah framework php opensource dengan menggunakan konsep MVC (Model, View, Controller) untuk membangun website dinamis atau aplikasi berbasis web. CodeIgniter ini dikembangkan oleh Rick Ellis yang di rilis pertama kali pada tanggal 28 Februari 2006.

Seiring berjalan waktu framework CodeIgniter berkembang pesat karena sangat friendly bagi para programmer web.

Hingga saat ini versi terakhir dari framework yang satu ini adalah CodeIgniter 3.0 yang resmi di rilis pada tanggal 30 Maret 2015.

Perkembangan framework CodeIgniter ini sangat pesat dibanding dengan framework-framework php lainnya. CodeIgniter sudah familiar bagi para pengembang aplikasi berbasis website karena framework ini mudah dipahami dan mudah dipelajari sehingga banyak programmer yang menggunakan framework yang satu ini. tetapi tidak sebatas itu saja yang menjadi alasan bagi mereka kenapa menggunakan framework CodeIgniter, ada beberapa alasan ataupun pertimbangan mengapa kebanyakan para programmer web menggunakan framework CodeIgniter sebagai tools dalam mengembangkan websitenya. Berikut beberapa alasannya :

- Framework CodeIgniter adalah aplikasi yang bersifat opensource.
- CodeIgniter mudah dipahami dan dipelajari.
- CodeIgniter merupakan framework yang mempunyai fungsi-fungsi yang lengkap.
- Banyak library yang bisa digunakan pada framework yang satu ini
- Dokumentasinya sangat lengkap sehingga memudahkan para pemula untuk mempelajari framework ini.
- Komunitas yang sudah berkembang di seluruh dunia sehingga memudahkan untuk saling berbagi ilmu tentang CodeIgniter.

3.13.1 Konsep MVC (Model, View, Controller)

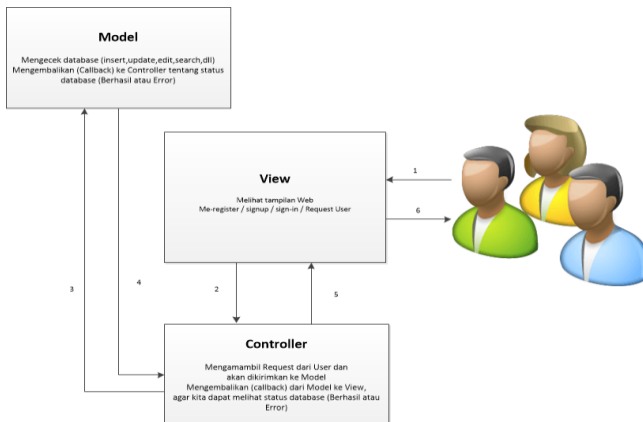
Model View Controller merupakan suatu konsep yang cukup populer dalam pembangunan aplikasi web. MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, user interface, dan bagian yang menjadi kontrol aplikasi. Terdapat 3 jenis komponen yang membangun suatu MVC pattern dalam suatu aplikasi yaitu :

1. View, merupakan bagian yang menangani presentation logic. Pada suatu aplikasi web bagian ini biasanya berupa file template HTML, yang diatur oleh controller. View berfungsi untuk menerima dan merepresentasikan data kepada user.

Bagian ini tidak memiliki akses langsung terhadap bagian model.

2. Model, biasanya berhubungan langsung dengan database untuk memanipulasi data (insert, update, delete, search), menangani validasi dari bagian controller, namun tidak dapat berhubungan langsung dengan bagian view.
3. Controller, merupakan bagian yang mengatur hubungan antara bagian model dan bagian view, controller berfungsi untuk menerima request dan data dari user kemudian menentukan apa yang akan diproses oleh aplikasi.

Model untuk mengatur alur database, View untuk menampilkan web, sedangkan Controller untuk mengatur alur kerja antara Model dan View.



Gambar 3.1. Konsep MCV

3.14 Framework

Framework secara sederhana dapat diartikan kumpulan dari fungsi-fungsi/prosedur-prosedur dan class-class untuk tujuan tertentu yang sudah siap digunakan sehingga bisa lebih mempermudah dan mempercepat pekerjaan seorang programmer, tanpa harus membuat fungsi atau class dari awal

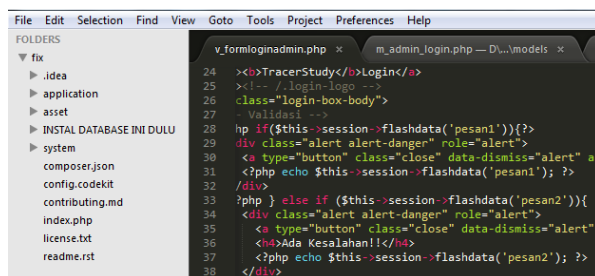
3.15 XAMPP

Menurut Yogi wicaksono (2008:7) “XAMPP adalah sebuah software yang berfungsi untuk menjalankan website berbasis PHP dan menggunakan pengolah data MySQL dikomputer local”. XAMPP berperan sebagai server web pada komputer anda. XAMPP juga dapat disebut sebuah CPanel server virtual, yang dapat membantu anda melakukan preview sehingga dapat memodifikasi website tanpa harus online atau terakses dengan internet.

3.16 Sublime Text Editor

Sublime Text Editor adalah editor teks untuk berbagai bahasa pemrograman termasuk pemrograman PHP. Sublime Text Editor merupakan editor text lintas-platform dengan Python application programming interface (API). Sublime Text Editor juga mendukung banyak bahasa pemrograman dan bahasa markup, dan fungsinya dapat ditambah dengan plugin, dan Sublime Text Editor tanpa lisensi perangkat lunak.

Sublime Text Editor pertama kali dirilis pada tanggal 18 januari 2008, dan sekarang versi Sublime Text Editor sudah mencapai versi 3 yang dirilis pada tanggal 29 januari 2013. Sublime Text mendukung operation system seperti Linux, Mac Os X, dan juga windows. Sangat Banyak fitur yang tersedia pada Sublime Text Editor diantaranya minimap, membuka script secara side by side, bracket highlight sehingga tidak bingung mencari pasangannya, kode snippets, drag and drop direktori ke sidebar terasa mirip dengan TextMate untuk Mac OS.



gambar 3.2. Tampilan Siblime editor

BAB IV

PELAKSANAAN PKL

4.1 Gambaran Umum Sistem Informasi Alumni Tracersrudy

Sistem Informasi Alumni Tracerstudy merupakan suatu sistem untuk mempermudah kebutuhan akan informasi baik berhubungan dengan akadenis atau sebagai arsip data alumni dari Fakultas Peternakan. Kebutuhan informasi tersebut dapat berupa data dari alumni yang ada. Sistem informasi alumni ini dapat mempermudah mendapatkan informasi alumni secara lengkap.

Dalam sistem informasi alumnni mahasiswa sebagai *user* dapat memberikan informasi yang lengkap dan memberikan pesan pesan di dalamnya. Sedangkan untuk pegawai atau dosen Fakultas Peternakan sebagai admin tersebut. Admin tidak hanya mengatur pengawas data alumni, namun admin juga mengelola data alumni yang ada dalam Fakultas Peternakan.

Pada penyampaian laporan ini, yang akan dibahas lebih dalam adalah mengenai pembuatan modul admin system informasi tracerstudy Fakultas Peternakan. Sistem ini digunakan untuk membantu mengelola data para alumni yang ada di Fakultas Peternakan.

4.2 Pengembangan Sistem

Model proses yang di gunakan dalam pengembangan Sistem Informasi Alumni Tracerstudy ini adalah model *SDLC (Systems Development Life Cycle)*. Dipilihnya model ini dikarenakan dalam proses aplikasinya cukup mudah, semua kebutuhan sistem juga dapat didefinisikan secara utuh dan pengadaan perbaikan sejumlah masalah yang terkait dengan pengembangan perangkat lunak. Dalam model *SDLC (Systems Development Life Cycle)* terdapat beberapa tahapan yang digunakan untuk proses pengembangan sistem ini, yaitu analisis kebutuhan sistem, perancangan sistem sesuai definisi kebutuhan sistem, implementasi rancangan sistem dan pengujian sistem.

4.3 Analisis Kebutuhan Sistem

Dari penjabaran umum yang telah dibahas maka dilakukan proses analisis kebutuhan dari sistem informasi alumni yang akan dirancang. Adapun analisis kebutuhan sistem dijabarkan sebagai berikut :

1. Sistem mampu menambah, mengedit, menampilkan dan menghapus data alumni.
2. Sistem mampu menambah data user sebagai alumni.

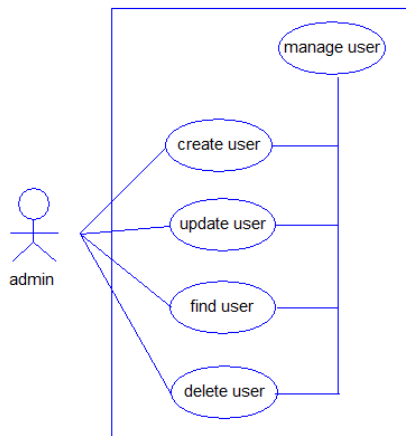
4.4 Perancangan Sistem

Pada bagian perancangan akan dijabarkan desain sistem yang akan dibuat. Pada perancangan ini digunakan desain UML yang terdiri dari *Diagram Use Case*, *Activity Diagram*, *Class Diagram*, ERD, dan rancangan antarmuka sistem informasi lab baca.

4.4.1 Use Case Diagram

Berikut ini rancangan diagram Use Case dari Sistem Informasi Alumni Tracerstudy Fakultas Peternakan Universitas Udayana.

1. Use Case Diagram Admin

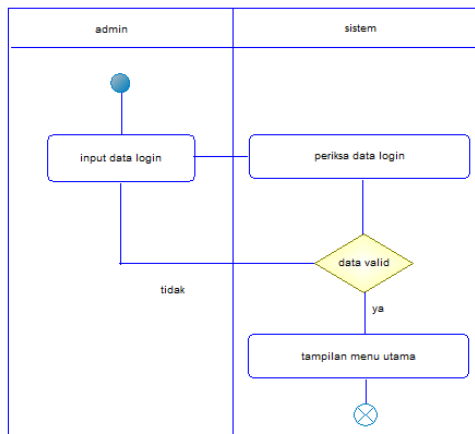


Gambar 4.1 Use Case Diagram Admin

Pada diagram *use case* diatas, yang menjadi aktor adalah admin Sistem Informasi. Pegawai sebagai mengelola data alumni. Dimana sebelum melakukan itu admin harus login terlebih dahulu.

4.4.2 Activity Diagram

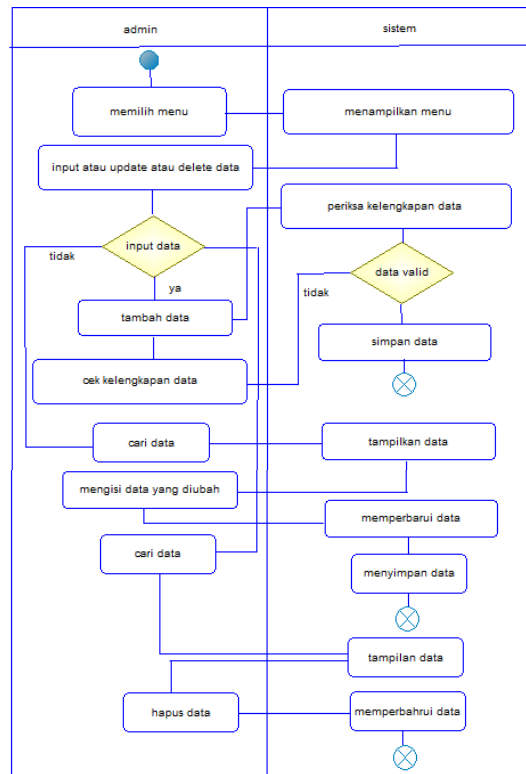
1. Proses Login Admin



Gambar 4.2 Proses Login Admin

Gambar diatas merupakan proses login admin, dimana pertama yang dilakukan admin adalah menginput data untuk login admin. Data yang diinputkan adalah username dan password. Setelah data diinputkan sistem akan mengecek data login, jika data login valid maka admin akan ditampilkan menu utama sistem, jika tidak valid maka admin akan melakukan proses login lagi.

2. Manajemen Data Alumni

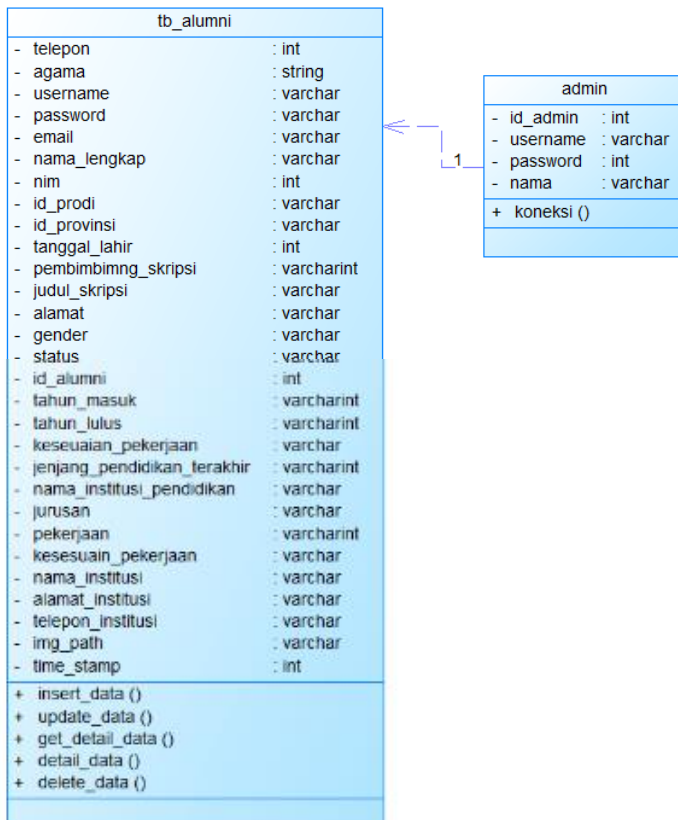


Gambar 4.3 Proses manajemen Admin

merupakan proses manajemen data anggota yang dilakukan petugas. Proses pertama petugas akan memilih menu dan sistem akan menampilkan menu master. Setelah menu ditampilkan, petugas akan

memilih untuk input atau update atau hapus data. Jika petugas memilih input data, setelah petugas menambahkan data anggota maka sistem akan mengecek kelengkapan data anggota. Jika valid maka data anggota disimpan, jika tidak maka petugas akan menginputkan data anggota lagi. Jika petugas memilih untuk update data maka petugas harus mencari data yang akan diupdate, dan sistem akan menampilkan data yang sudah dipilih. Jika data sudah ditampilkan, petugas akan mengisi data yang akan diubah, sistem akan memperbaharui data tersebut dan menyimpannya. Jika petugas memilih hapus data maka petugas harus mencari data yang akan dihapus dan sistem akan menghapus data tersebut dari database.

4.4.2 Class Diagram



Gambar 4.4 *Class Diagram* Sistem Informasi Alumni

Gambar diatas merupakan class diagram dari sistem informasi Alumni. Berikut merupakan penjelasan dari class diagram diatas :

1. Kelas admin merupakan induk dari kelas alumni , dimana sifat – sifat yang dimiliki oleh kelas admin diturunkan juga kepada kelas alumni.

4.5 ERD (Entitiy Relationship Diagram)



Gambar 4.5 *Entity Relationship Diagram* Sistem Informasi Tracerstudy

4.6 Implementasi

Setelah tahap perancangan maka tahapan berikutnya adalah mengimplementasikan rancangan tersebut kedalam bentuk kode program HTML,CSS, PHP, dan JavaScript serta basis data MySQL

menggunakan text editor Sublime dan PHPMyAdmin untuk memanajemen basis data MySQL.

i. source code HTML

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>Login Tracer Study Fakultas
Pernakan Universitas Udayana</title>
    <!-- Bootstrap Core CSS -->
    <link rel="stylesheet"
href="<?php echo
base_url('asset/css/bootstrap.css
'); ?>">
    <linkrel="stylesheet"
href="https://code.ionicframework
.com/ionicons/2.0.1/css/ionicons.
min.css">
    <link rel="stylesheet"
href="<?php echo
base_url('asset/css/AdminLTE.css'); ?
>">
    <!-- Custom Fonts -->
    <link rel="stylesheet"
href="<?php echo
base_url('asset/font-
awesome/css/font-awesome.css'); ?
?>">
</head>
<body class="hold-transition
login-page">
    <?php echo
form_open('admin_auth2/proseslogi
nadmin'); ?>
        <div class="login-
box">
            <div class="login-logo">
```

```

        <a
        ><b>TracerStudy</b>Login</a>
        </div><!-- /.login-logo -->
        <div class="login-box-
        body">
            <!-- Validasi -->
            <?php if($this->session
            >flashdata('pesan1')){?>
            <div class="alert alert-danger"
            role="alert">
                <a type="button" class="close"
                data-dismiss="alert" aria-
                label="close">&times</a>

```

penerapan html pada view from login admin

ii. CSS

```

<title>Login Tracer Study Fakultas
Paternakan Universitas Udayana</title>
<!-- Bootstrap Core CSS -->
<link rel="stylesheet" href="<?php
echo
base_url('asset/css/bootstrap.css');
?>">
<link rel="stylesheet"
href="https://code.ionicframework.co
m
/ionicons/2.0.1/css/ionicons.min.css
">

<link rel="stylesheet" href="<?php
echo
base_url('asset/css/AdminLTE.css');?>">
<!-- Custom Fonts -->
<link rel="stylesheet" href="
<?php echo base_url('asset/font-
awesome
/css/font-awesome.css'); ?>">

```

iii. source code PHP

```
<?php echo $this->session-
>flashdata('pesan1'); ?>
</div>
        <?php } else if ($this-
>session->flashdata('pesan2')){ ?>
            <div class="alert alert-
danger" role="alert">
                <a type="button"
class="close" data-dismiss="alert"
aria-label="close">&times</a>
                <h4>Ada
Kesalahan!!</h4>
                <?php echo $this-
>session->flashdata('pesan2'); ?>
            </div>
```

penerapan html pada view from login admin

iv. Java Script

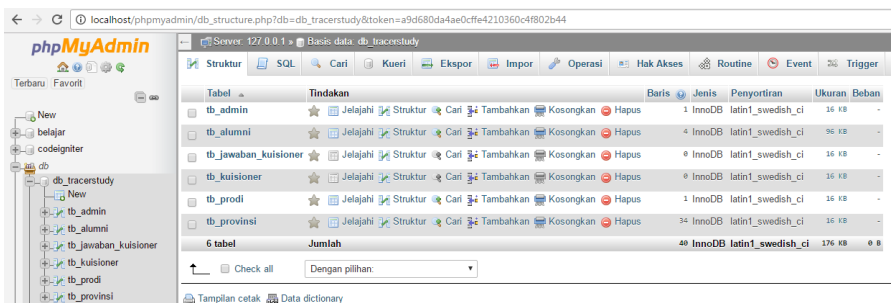
```
<script src="<?php echo
base_url('asset/js/jquery
2.1.4.min.js');?>"></script>
<!-- Bootstrap 3.3.5 -->
<script src="<?php echo
base_url('asset/js/bootstrap.min.js');?
"></script>
<script src="<?php echo
base_url('asset/css/plugins/datatables/
jquery.dataTables.min.js');?>"></script
>
<script src="<?php echo
base_url('asset/css/plugins/datatables/
dataTables.bootstrap.min.js');?>"></scr
ipt>
<script src="<?php echo
base_url('asset/css/plugins/slimScroll/
jquery.slimscroll.min.js');?>"></script
>
```

```

<script src="<?php echo
base_url('asset/css/plugins/fastclick/
fastclick.js');?>"></script>
<!-- AdminLTE App -->
<script src="<?php echo
base_url('asset/js/app.min.js');?>">
</script>
<!-- AdminLTE for demo purposes
-->
<script src="<?php echo
base_url('asset/js/demo.js');?>"></s
cript>
<script>

```

penerapan html pada view from login admin
v. phpmyadmin



gambar 4.6 penggunaan phpmyadmin sebagai DBMS

4.5 Testing

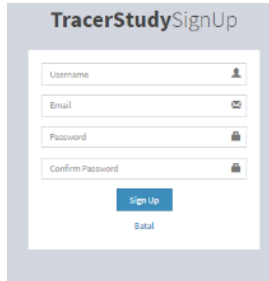
a. Tampilan Home



Gambar 4.7 *Tampilan home*

Gambar 4.7 merupakan tampilan home dari system informasi TracerStudy Fakultas Peternakan Unversitas Udayana.

b. Form Daftar Alumni

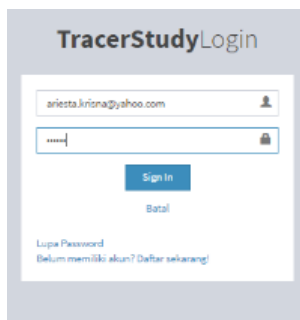


The image shows a web form titled "TracerStudySignUp". It contains four input fields: "Username" with a person icon, "Email" with an envelope icon, "Password" with a lock icon, and "Confirm Password" with a lock icon. Below the fields are two buttons: a blue "Sign Up" button and a smaller "Batal" (Cancel) link.

Gambar 4.8 *form daftar alumni*

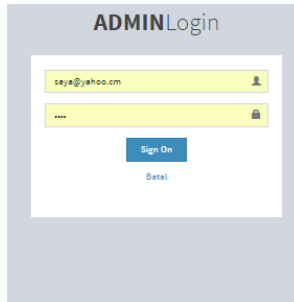
Pada gambar 4.8 merupakan tampilan untuk melakukan pendaftaran bagi alumni Fakultas Peternakan Universitas Udayana.dalam form ini alumni harus menginputkan username,email,password dan confirm password.

c. Form SignUp



The image shows a web form titled "TracerStudyLogin". It contains two input fields: a text field for the username containing "ariesta.liricna@yahoo.com" with a person icon, and a password field with a lock icon. Below the fields are two buttons: a blue "Sign In" button and a smaller "Batal" (Cancel) link. At the bottom, there is a link "Lupa Password" and a text prompt "Belum memiliki akun? Daftar sekarang!" (Don't have an account? Register now!).

- Gambar 4.9 tampilan sign up
- Pada gambar 4.9 merupakan tampilan untuk login alumni
- d. *Form Login Admin*

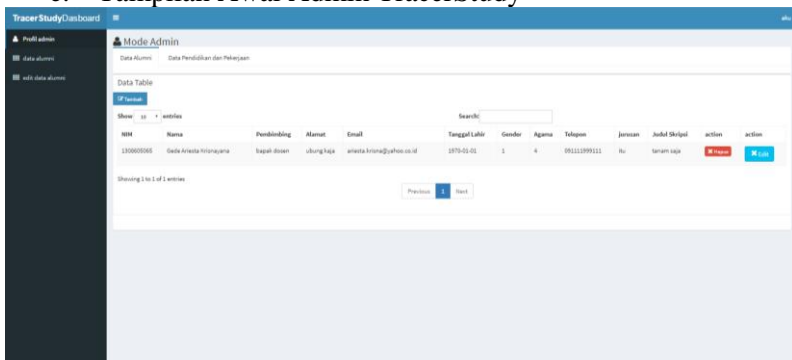


The image shows a web form titled "ADMIN Login". It has two input fields: one for email (containing "saye@yahoo.com") and one for password (containing "****"). Below the fields is a blue "Sign On" button and a link labeled "Batal" (Cancel).

Gambar 4.10 *Form Login Admin*

Gambar 4.10 merupakan tampilan untuk dapat *login* ke sistem, dengan cara menginputkan *username* dan *password* yang ada di *database*.

- e. *Tampilan Awal Admin TracerStudy*



The image shows the "TracerStudy Admin" dashboard. It features a sidebar with navigation links: "Profil admin", "data alumni", and "info data alumni". The main content area is titled "Mode Admin" and includes a "Data Alumni" section with a "Data Table" and a "Data Table" button. Below this is a table with columns: NIM, Nama, Pendidikan, Alamat, Email, Tanggal Lahir, Gender, Agama, Telepon, Jurusan, Jumlah Skripsi, action, and action. The table contains one row of data for a student named Gada Arista Hironjano.

| NIM | Nama | Pendidikan | Alamat | Email | Tanggal Lahir | Gender | Agama | Telepon | Jurusan | Jumlah Skripsi | action | action |
|-----------|-----------------------|------------|------------|------------------------------|---------------|--------|-------|-------------|---------|----------------|----------------------|------------------------|
| 150000505 | Gada Arista Hironjano | bagi dosen | Ubung Tiga | arista.hironjano@yahoo.co.id | 1970-05-01 | S | A | 08111099111 | Iti | terjemah | Edit | Delete |

Showing 1 to 1 of 1 entries

Previous 1 Next

Gambar 4.11 Tampilan Awal TracerStudy

Tampilan awal *admin* setelah *login* adalah laporan dari jumlah semua data yang ada yaitu data alumni

f. Tambah Data Alumni

The screenshot displays the 'Tambah Data Alumni' form in the TracerStudy dashboard. The form is divided into several sections with labels like 'Nama', 'NISN', 'Nama Pendidikan', 'Alamat', 'Email', 'Telepon', 'Gender', 'Agensi', 'No. Telepon', 'Nama Supervisor', and 'Alamat Supervisor'. Each section contains a text input field. On the left side, there is a table titled 'Data Table' with columns 'NISN', 'Nama', and 'Pendidikan'. The table contains one row of data: '1210000000', 'Data Kriteria Institusi', and 'Kategori'. Below the table, there is a button 'Tambah Data Alumni'. At the bottom right, there are buttons 'Kembali' and 'Simpan'.

Gambar 4.12 Tambah Data Alumni

Gambar 4.18 merupakan *form* untuk tambah alumni, semua data harus *diinputkan* untuk menambahkan data alumni. Jika satu data tidak ada maka data tersebut tidak akan bisa dimasukkan ke *database*.

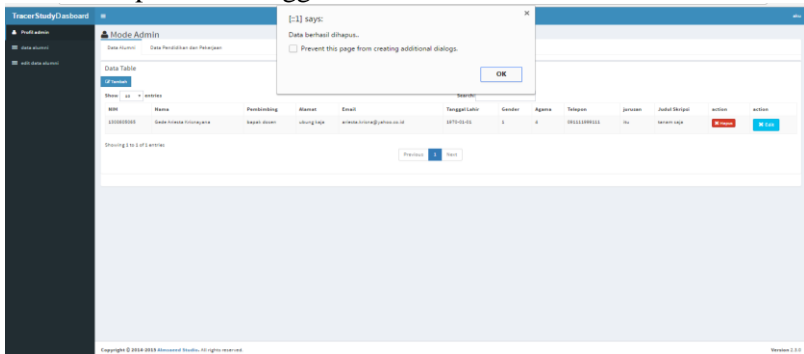
g. Edit Data Anggota

The screenshot displays the 'Detail Alumni' form in the TracerStudy dashboard. The form is divided into several sections with labels like 'NISN', 'Nama', 'Nama Pendidikan', 'Alamat', 'Email', 'Telepon', 'Gender', 'Agensi', 'No. Telepon', 'Nama Supervisor', and 'Alamat Supervisor'. Each section contains a text input field. On the left side, there is a table titled 'Data Table' with columns 'NISN', 'Nama', and 'Pendidikan'. The table contains one row of data: '1210000000', 'Data Kriteria Institusi', and 'Kategori'. Below the table, there is a button 'Tambah Data Alumni'. At the bottom right, there are buttons 'Kembali' and 'Simpan'.

Gambar 4.13 Edit Data Alumni

Gambar 4.13 merupakan *form* untuk *edit* alumni. Admin hanya mengklik tombol biru untuk data yang akan *diedit*, maka data akan ditampilkan lagi untuk *diedit* kembali.

h. Hapus Data Anggota



Gambar 4.14 Hapus Data Alumni

Untuk menghapus data anggota, admin hanya mengklik tombol hapus dan *notifikasi* untuk menghapus data akan keluar.

4.8 Pengujian Sistem

Setelah proses implementasi dilakukan maka dilakukan pengujian pada sistem. Dalam kasus ini pengujian sistem dilakukan dengan metode BlackBox.

Tabel 4.1 Tabel Pengujian Sistem Dengan Metode Black Box

| No | Skenario | Keluaran | Hasil | Kesimpulan |
|----|------------------------|--------------------------------------|-----------------------|------------|
| 1 | Alumni login ke system | Sistem menerima dan menampilkan data | Sesuai dengan harapan | VALID |

| | | | | |
|---|--------------------------|---|-----------------------|-------|
| 2 | Alumni daftar ke system | Sistem menerima dan menambah user pada databasse | Sesuai dengan harapan | VALID |
| 3 | logout dari system | Sistem keluar dan tidak menampilkan home | Sesuai dengan harapan | VALID |
| 4 | Admin login ke system | Sistem menerima dan menampilkan data | Sesuai dengan harapan | VALID |
| 5 | Manajemen data Alumni | Sistem menampilkan form manajemen Alumni | Sesuai dengan harapan | VALID |
| 6 | Menambah data Alumni | Sistem menampilkan Alumni yang sudah ditambahkan | Sesuai dengan harapan | VALID |
| 7 | Mengedit data Alumni | Sistem menampilkan data alumni yang sudah diperbahurui | Sesuai dengan harapan | VALID |
| 8 | Admin logout dari system | Sistem keluar dan tidak menampilkan Sistem Informasi Alumni TracerStudy | Sesuai dengan harapan | VALID |

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil dari kegiatan praktek kerja lapangan di Jurusan Biologi, maka dapat ditarik sebuah kesimpulan yaitu penulis dapat merancang dan mengembangkan sistem informasi sesuai dengan kebutuhan yang didefinisikan dan mengimplementasikannya menjadi suatu sistem informasi yang dapat digunakan untuk menampilkan informasi alumni yang terdapat di Fakultas Peternakan Universitas Udayana. Sehingga sistem informasi yang telah dirancang ini dapat mempermudah fakultas dalam memperoleh informasi alumni, manajemen data alumni dan pelacakan alumni Fakultas Peternakan Universitas Udayana.

5.2 Saran

Berdasarkan hasil perancangan dan implementasi yang dilakukan, adapun saran terkait hasil perancangan Sistem Informasi Alumni TracerStudy Fakultas Peternakan Universitas Udayana ini adalah memperbaharui desain tampilan sistem supaya lebih nyaman dilihat user. Karena sistem informasi yang dirancang pada saat pelaksanaan Praktek Kerja Lapangan ini masih dengan desain yang sangat simple. kemudian perlu juga ditambahkan sebuah halaman dimana data dari alumni yang masih belum mendapatkan perkerjaan dapat di berikan kepada perusahaan, sehingga alumni dan perusahaan langsung mendapatkan informasi yang dibutuhkan Dan menambahkan *fitur-fitur* baru untuk memudahkan *administrator*.

DAFTAR PUSTAKA

DOSEN PENDIDIKAN.12 Pengertian Sistem dan Fungsi Sistem Informasi Menurut Para Ahli. www.dosenpendidikan.com/12-pengertian-dan-fungsi-sistem-informasi-menurut-para-ahli/ . Diakses tanggal 2 Mei 2016

Dewi Santika. Konsep dasar uml. <http://santika.ilearning.me/2-2-1-konsep-dasaruml-unified-modeling-language/>. Diakses tanggal 2 Mei 2016

Kompasiana. Pengertian SI (Sistem Informasi). http://www.kompasiana.com/dimasosd/pengertian-si-sistem-informasi_55291077f17e6126268b48b6. Diakses tanggal 2 Mei 2016

Burung Internet. Pengertian HTML, Fungsi HTML serta Sejarah HTML. <http://www.burung-net.com/2015/05/pengertian-html-fungsi-html-sejarah-html.html>. Diakses tanggal 2 Mei 2016

AgusDar.com. Metode Pengembangan Sistem Waterfall. <https://agusdar.wordpress.com/2013/04/13/metode-pengembangan-sistem-waterfall/> . Diakses tanggal 2 Mei 2016

ttaufikhidayat.com. ERD(Entity Relationship Diagram). [http://www.ttaufikhidayat.com/berita64-ERD\(Entity-Relationship-Diagram\).html](http://www.ttaufikhidayat.com/berita64-ERD(Entity-Relationship-Diagram).html) . Diakses tanggal 2 Mei 2016

<http://monica.ilearning.me/tugas-akhir/bab-ii/>

LAMPIRAN

Lampiran 1. Form aktivitas harian

AKTIVITAS HARIAN PKL

Nama : Gede Ariesta Krisnayana
 NIM : 1308605065
 Lokasi PKL : Fakultas Peternakan Universitas Udayana
 Waktu Pelaksanaan : 05 September 2016 - 25 November 2016

| No. | Nama Penanggung Jawab/Jabatan | Pelaksanaan PKL | | | Keterangan |
|-----|-------------------------------|-----------------|-------------------------|--|------------|
| | | Tanggal | Lokasi | Aktivitas | |
| 1 | pembantu dekan 1 | 05-09-2016 | kantor pembantu dekan 1 | penempatan serta pembagian tugas yang di berikan oleh pembantu dekan 1 | |
| 2 | pembantu dekan 1 | 06-09-2016 | kantor pembantu dekan 1 | libur galungan | |
| 3 | pembantu dekan 1 | 07-09-2016 | kantor pembantu dekan 1 | libur galungan | |
| 4 | pembantu dekan 1 | 08-09-2016 | fakultas peternakan | mendapatkan hak akses admin untuk update website fapet.unud.ac.id | |
| 5 | pembantu dekan 1 | 09-09-2016 | di rumah | libur | |
| 6 | pembantu dekan 1 | 10-09-2016 | di rumah | libur | |
| 7 | pembantu dekan 1 | 11-09-2016 | di rumah | libur | |

1/9

Lampiran 1. Form Aktivitas PKL Teknik Informatika FMIPA Universitas Udayana

| | | | | | |
|----|------------------|------------|---------------------|--|--|
| 8 | pembantu dekan 1 | 12-09-2016 | di rumah | libur idul adha | |
| 9 | pembantu dekan 1 | 13-09-2016 | fakultas peternakan | koordinasi malikah update dengan pak pd 3 dan data yang akan di update kedalam web fapet | |
| 10 | pembantu dekan 1 | 14-09-2016 | fakultas peternakan | mengedit foto yang akan diupdate kedalam web fapet | |
| 11 | | 15-09-2016 | | | |
| 12 | PD 1 | 16-09-2016 | fakultas peternakan | libur kuningan | |
| 13 | PD 1 | 17-09-2016 | fakultas peternakan | libur | |
| 14 | PD 1 | 18-09-2016 | fakultas peternakan | libur | |
| 15 | | 19-09-2016 | | | |
| 16 | PD1 | 20-09-2016 | fakultas peternakan | melakukan pembuatan sim tracer study | |
| 17 | PD1 | 21-09-2016 | fakultas peternakan | melakukan pembuatan sim tracer study | |
| | | | | | |

| | | | | | |
|----|-----|------------|---------------------|--------------------------------------|--|
| 18 | PD1 | 22-09-2016 | fakultas peternakan | melakukan pembuatan sim tracer study | |
| 19 | PD1 | 23-09-2016 | fakultas peternakan | melakukan pembuatan sim tracer study | |
| 20 | PD1 | 24-09-2016 | di rumah | libur | |
| 21 | PD1 | 25-09-2016 | di rumah | libur | |
| 22 | | 26-09-2016 | | | |
| 23 | | 27-09-2016 | | | |
| 24 | PD1 | 28-09-2016 | fakultas peternakan | pembuatan sim tracer study | |
| 25 | PD1 | 29-09-2016 | fakultas peternakan | pembuatan sim tracer study | |
| 26 | PD1 | 30-09-2016 | fakultas peternakan | pembuatan sim tracer study | |
| 27 | PD1 | 01-10-2016 | di rumah | libur | |
| 28 | PD1 | 02-10-2016 | di rumah | libur | |
| 29 | PD1 | 03-10-2016 | fakultas peternakan | pembuatan sim tracer study | |

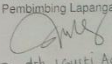
| | | | | | |
|----|-----|------------|---------------------|----------------------------|--|
| 30 | PD1 | 04-10-2016 | fakultas peternakan | pembuatan sim tracer study | |
| 31 | | 05-10-2016 | | | |
| 32 | PD1 | 06-10-2016 | fakultas peternakan | pembuatan sim tracer study | |
| 33 | PD1 | 07-10-2016 | fakultas peternakan | pembuatan sim tracer study | |
| 34 | PD1 | 08-10-2016 | di rumah | libur | |
| 35 | | 09-10-2016 | | | |
| 36 | PD1 | 10-10-2016 | fakultas peternakan | pembuatan sim tracer study | |
| 37 | PD1 | 11-10-2016 | fakultas peternakan | pembuatan sim tracer study | |
| 38 | PD1 | 12-10-2016 | fakultas peternakan | pembuatan sim tracer study | |
| 39 | PD1 | 13-10-2016 | - | izin sembelhyng ke pura | |
| 40 | PD1 | 14-10-2016 | - | izin sembelhyng ke pura | |

| | | | | | |
|----|-----|------------|---------------------|---|--|
| 41 | PD1 | 15-10-2016 | - | libur | |
| 42 | PD1 | 16-10-2016 | - | libur | |
| 43 | PD1 | 17-10-2016 | fakultas peternakan | pembuatan login admin pada sim tracer study | |
| 44 | PD1 | 18-10-2016 | fakultas peternakan | pembuatan login admin pada sim tracer study | |
| 45 | PD1 | 19-10-2016 | fakultas peternakan | pembuatan login admin pada sim tracer study | |
| 46 | PD1 | 20-10-2016 | - | ijin melasyat | |
| 47 | PD1 | 21-10-2016 | fakultas peternakan | pembuatan login admin pada sim tracer study | |
| 48 | PD1 | 22-10-2016 | - | libur | |
| 49 | PD1 | 23-10-2016 | - | libur | |
| 50 | | 24-10-2016 | | | |
| 51 | PD1 | 25-10-2016 | fakultas peternakan | pembuatan login admin pada sim tracer study | |

| | | | | | |
|----|------|------------|---------------------|---|--|
| 52 | PD 1 | 26-10-2016 | Fakultas Peternakan | pembuatan login admin pada sim tracer study | |
| 53 | PD 1 | 27-10-2016 | fakultas peternakan | pembuatan page admin | |
| 54 | PD 1 | 28-10-2016 | fakultas peternakan | pembuatan page admin | |
| 55 | PD1 | 29-10-2016 | libur | libur | |
| 56 | PD1 | 30-10-2016 | - | libur | |
| 57 | PD1 | 31-10-2016 | fakultas peternakan | pembuatan page admin | |
| 58 | PD1 | 01-11-2016 | fakultas peternakan | pembuatan page admin | |
| 59 | PD1 | 02-11-2016 | fakultas peternakan | pembuatan page admin | |
| 60 | PD1 | 03-11-2016 | fakultas peternakan | pembuatan page admin | |
| 61 | PD1 | 04-11-2016 | fakultas peternakan | pembuatan page admin | |
| 62 | PD1 | 05-11-2016 | - | libur | |

| | | | | | |
|----|-----|------------|---------------------|---|--|
| 63 | PD1 | 08-11-2016 | - | libur | |
| 64 | PD1 | 07-11-2016 | fakultas peternakan | pembuatan CRUD admin | |
| 65 | PD1 | 08-11-2016 | fakultas peternakan | pembuatan CRUD admin | |
| 66 | PD1 | 09-11-2016 | fakultas peternakan | pembuatan CRUD admin | |
| 67 | PD1 | 10-11-2016 | fakultas peternakan | pembuatan CRUD admin | |
| 68 | PD1 | 11-11-2016 | fakultas peternakan | pembuatan CRUD admin | |
| 69 | PD1 | 12-11-2016 | - | libur | |
| 70 | | 13-11-2016 | | | |
| 71 | | 14-11-2016 | | | |
| 72 | PD1 | 15-11-2016 | fakultas peternakan | melengkapi data admin Sistem Informasi Alumni fakultas peternakan | |
| 73 | PD1 | 16-11-2016 | fakultas peternakan | melengkapi data admin Sistem Informasi Alumni fakultas peternakan | |

| | | | | | |
|----|-----|------------|---------------------|---|--|
| 74 | PD1 | 17-11-2016 | fakultas peternakan | melengkapi data admin Sistem Informasi Alumni fakultas peternakan | |
| 75 | PD1 | 18-11-2016 | fakultas peternakan | melengkapi data admin Sistem Informasi Alumni fakultas peternakan | |
| 76 | PD1 | 19-11-2016 | - | libur | |
| 77 | PD1 | 20-11-2016 | - | libur | |
| 78 | PD1 | 21-11-2016 | fakultas peternakan | koordinasi sistem informasi dengan USDI | |
| 79 | PD1 | 22-11-2016 | fakultas peternakan | koordinasi sistem informasi dengan USDI | |
| 80 | PD1 | 23-11-2016 | fakultas peternakan | koordinasi sistem informasi dengan USDI | |
| 81 | PD1 | 24-11-2016 | fakultas peternakan | koordinasi sistem informasi dengan USDI | |
| 82 | PD1 | 25-11-2016 | fakultas peternakan | koordinasi sistem informasi dengan USDI | |

Denpasar 7 Desember 2016
 Pembimbing Lapangan,

 Dr. drh. Iquti Agung Arta Ruma, M.Si.