

**LAPORAN AKHIR PROGRAM RESPONSI
WEBSITE BERITA KEBAKARAN LAHAN SAWIT**



DISUSUN OLEH :

- 1. Loudry (193030503069) (CEO)**
- 2. Michael Isachar (193030503043) (CTO)**
- 3. Robertus Manserat (DBC 117035) (Data Analyst)**
- 4. Okki Andaresta (DBC118068) (Tester dan Dokumentasi)**
- 5. Andreas Aji Setioko (193030503061) (Back-end)**
- 6. Gebri Meizeri cordias (193030503068) (Front-end)**

JURUSAN TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS PALANGKA RAYA

2021

BAB I

LANDASAN TEORI

1.1 Kebakaran Hutan

Kebakaran Hutan Kebakaran hutan dapat di artikan sebagai pembakaran yang dilakukan secara sengaja dan dilakukan pada area yang direncanakan. Sedangkan pembakaran hutan adalah dimana terjadi akibat kegiatan yang dilakukan oleh manusia seperti pembukaan lahan dengan cara pembakaran. Ada tiga tipe bentuk kebakaran hutan , yaitu :

1. Kebakaran bawah (ground fire)

Kebakaran yang biasanya terjadi pada hutan yang bertanah gambut serta pada tanah yang mengandung mineral seperti batu bara, karena adanya bahan-bahan organik di bawah lapisan serasah yang mudah terbakar. Api dimulai dari membakar serasah dan kemudian membakar bahan-bahan organik yang berada pada lapisan di bawahnya. Kebakaran bentuk ini tidak menampilkan nyala api sehingga sulit dideteksi.

2. Kebakaran permukaan (surface fire)

Kebakaran yang terjadi pada permukaan tanah. Api membakar serasah, semak-semak dan pohon. Kebakaran ini tidak sampai membakar tajuk pohon karena pohon-pohonnya jarang atau dari jenis yang sulit terbakar.

3. Kebakaran tajuk (crown fire)

Kebakaran yang terjadi pada tajuk-tajuk pohon. Api berawal dari serasah (kebakaran permukaan), kemudian merambat ke tajuk pohon karena adanya tajuk, seperti tumbuhan liar atau cabang dan ranting-ranting kering yang menyentuh serasah hutan.

Kebakaran seperti ini yang paling sulit dikendalikan karena menjalar sangat cepat searah dengan arah angin.

1.2 Lahan Sawit

Lahan sawit memiliki mulifungsi yang memberi multi manfaat bagi masyarakat dunia. Lahan sawit merupakan bagian dari paru-paru, mata rantai energi, konservasi tanah dan air serta mesin cetak kue ekonomi dalam ekosistem planet bumi.

Lahan sawit bukan hanya berfungsi sebagai ekonomi saja sebagaimana yang diketahui masyarakat, melainkan juga memiliki fungsi ekologis dalam ekosistem. Oleh

karena itu, melihat dan memperlakukan lahan sawit jangan hanya melihat fungsi ekonomi tetapi juga fungsi ekologisnya. Mulifungsi lahan sawit yang dimaksud terdiri atas: Pertama, lahan sawit merupakan bagian dari “paru-paru” ekosistem seperti hutan.

Pada manusia, paru-paru berfungsi membersihkan dan membuang karbondioksida dari dalam tubuh dan memasukkan oksigen kedalam tubuh manusia

1.3 INTERNET

1.3.1. Pengertian Internet

Internet (*interconnection-networking*) yang pada awalnya dikenal dengan sebutan *Advanced Research Project Agency Network* (ARPANET) yang merupakan jaringan komputer yang dibuat oleh *Advanced Research Project Agency* (ARPA) dari Departemen Pertahanan Amerika Serikat pada tahun 1969. Internet sendiri merupakan sebuah sistem global dari seluruh jaringan komputer yang saling terhubung menggunakan standar *Internet Protocol Suite* (TCP/ IP) sebagai protokol pertukaran paket data untuk melayani miliaran pengguna di seluruh dunia (*Meloni, 2012*).

Internet adalah sebuah solusi jaringan yang dapat menghubungkan beberapa jaringan lokal yang ada pada suatu daerah, kota atau bahkan negara. *Internet* dapat menghubungkan beberapa jaringan lokal yang ada pada setiap tempat (*Kurniawan, 2004*).

Menghubungkan beberapa komputer sehingga dapat menjadi sebuah jaringan membutuhkan suatu media penghubung yang bernama *TCP/IP*, yaitu sebuah protokol yang mengidentifikasi sebuah komputer yang terhubung di dalam sebuah jaringan. TCP / IP memiliki teknik mengidentifikasi dengan menggunakan penomoran yang dinamakan nomor *IP/IP address* (*Internet Protokol Address*). Nomor *IP* sebuah komputer dapat terhubung dengan komputer lain dalam sebuah jaringan atau dalam jaringan global yang disebut *internet*.

1.4 WEBSITE

1.4.1 Pengertian Website

Website atau sering disingkat dengan istilah situs adalah sejumlah halaman web yang memiliki topik saling terkait, terkadang disertai pula dengan berkas-berkas gambar, video, atau jenis-jenis berkas lainnya. Sebuah *website* biasanya ditempatkan setidaknya pada sebuah *server web* yang dapat diakses melalui jaringan seperti internet, ataupun jaringan wilayah lokal (LAN) melalui alamat internet yang dikenali

sebagai URL. Gabungan atas semua situs yang dapat diakses publik di internet disebut pula sebagai *Waring Wera Wanua* atau lebih dikenal dengan singkatan *WWW*.

Meskipun setidaknya halaman beranda situs internet umumnya dapat diakses publik secara bebas, pada prakteknya tidak semua situs memberikan kebebasan bagi publik untuk mengaksesnya, beberapa situs *web* mewajibkan pengunjung untuk melakukan pendaftaran sebagai anggota, atau bahkan meminta pembayaran untuk dapat menjadi anggota untuk dapat mengakses isi yang terdapat dalam situs web tersebut, misalnya situs-situs yang menampilkan pornografi, situs-situs berita, layanan surel (*e-mail*), dan lain-lain. Pembatasan-pembatasan ini umumnya dilakukan karena alasan keamanan, menghormati privasi, atau karena tujuan komersil tertentu.

1.4.1 Sejarah Website

Web dikembangkan pertama kali oleh Sir Timothy John Tim Berners-Lee, hanya saja pada saat itu *web* masih berjalan tanpa terhubung jaringan. *Web* semakin populer ketika mulai terhubung jaringan *internet*, yaitu pada akhir tahun 80-an. Saat itu di laboratorium CERN berlokasi di kota Geneva, Swiss menyatakan bahwa *web* bisa diakses melalui jaringan dan dimiliki oleh siapa saja.

Sejarah *web* juga berkaitan dengan sejarah perkembangan teknologi komputer. Karena pada awalnya tampilan *web* masih sangatlah sederhana, hanya menampilkan teks, lalu untuk *hyperlink (link)* pada saat itu masih menggunakan tampilan nomor yang menghubungkan antara satu halaman ke halaman lainnya.

Pada saat itu pun, teknologi *web* dikembangkan dan berjalan pada sistem operasi Unix, masih sangat jarang yang menggunakan teknologi windows. andapun ada, teknologi windows masih sangat sederhana.

Lalu dengan semakin beragamnya tampilan dan penerapan teknologi berbasis *desktop* mendorong perkembangan teknologi sejarah *web*, baik berupa teknologi tampilan (*GUI Graphical User Interface*), teknologi *browser*, teknologi bahasa yang digunakan untuk mengembangkan *web*, platform web dan beragam teknologi *web* lainnya. Seiring waktu keberadaan *web* pun meledak.

Dari hanya berjumlah ribuan, hingga mencapai jumlah jutaan bahkan milyaran. Fungsinya pun berkembang, bukan hanya sebagai media bertukar informasi, tapi juga mewujudkan berupa aplikasi sistem informasi berbasis *web*.

Dalam sejarah *web*, *web browser* digunakan sebagai media untuk berselancar, dan browser yang pertama kali populer digunakan yaitu Internet Explorer. Internet Explorer digunakan sebagai aplikasi untuk mengakses web melalui komputer. Seiring perkembangan saat ini sudah tersedia beragam web browser selain Internet Explorer, seperti mozilla firefox, google chrome, safari, opera dan sebagainya. Untuk bahasa pemrograman yang pada awalnya hanya terdapat beberapa bahasa, saat ini sudah beragam.

Salah satu bentuk bagian perkembangan sejarah web, lahir definisi web pada akhir tahun 90-an, yaitu yang disebut web 2.0. Web 2.0 ini didefinisikan sebagai web yang berfungsi kolaboratif, informatif, dengan tujuan membuat pengguna semakin dekat, salah satu cirinya ditandai dengan hadirnya wiki, blog, *social network*. Bertolak dari sejarah *web* yang masih sangat sederhana digunakan, saat ini akses web bukan hanya dapat digunakan melalui komputer melainkan juga sudah menjangkau hingga berbasis *mobile*. Baik berupa *laptop*, *netbook*, *smartphone* hingga *handphone*. Dengan begitu kemudahan akses web melalui beragam media mana saja, semakin membuat web bagian dari kebutuhan hidup masyarakat secara luas.

1.4.3 World Wide Web (WWW)

World Wide Web yang juga disebut web didefinisikan sebagai sistem interkoneksi komputer internet (disebut server) yang mendukung dokumen-dokumen berformat multimedia (Williams & Sawyer, 2007).

Menurut Nugroho (2004), *World Wide Web* yang sering disingkat WWW adalah layanan yang disediakan server untuk komputer *client* dalam mengakses data di server. Komputer *client* merupakan suatu komputer yang digunakan untuk mengakses layanan server seperti web. Dengan adanya WWW seorang pengguna dapat menampilkan halaman virtual yang disebut *website*.

Pada tahun 1990 *World Wide Web* mulai dikembangkan oleh CERN (Laboratorium Fisika Partikel di Swiss) berdasarkan proposal yang dibuat oleh Tim Bernes-lee. Namun demikian WWW browser yang baru lahir dua tahun kemudian, tepatnya tahun 1992 dengan nama Viola. Viola diluncurkan oleh Pei Wei dan didistribusikan bersama CERN.

Jika dilihat dari proses kerjanya WWW dapat dibagi menjadi beberapa komponen seperti berikut:

1. *Protocol* adalah sebuah media yang distandarkan untuk dapat mengakses komputer ke dalam jaringan. WWW memiliki standar *protocol* yang bernama *Hypertext Transfer Protocol* (HTTP).
2. *Address* merupakan alamat yang berkaitan dengan penamaan sebuah komputer didalam jaringan alamat ini merupakan sebuah nomor yang dimiliki sebuah komputer yang sering disebut nomor IP, akan tetapi dengan perkembangan zaman dibentuklah metode baru yang bernama *domain name*, sehingga nomor IP tersebut digantikan dengan sebuah alamat yang dinamakan *Uniform Resource Locator* (URL).
3. *Hypertext Markup Language* (HTML) yaitu salah satu bahasa *scripting* yang dapat menghasilkan halaman *website* sehingga halaman tersebut dapat diakses pada setiap komputer pengakses (*client*).

Pertama-tama yang harus diketahui bahwa *world wide web*(www) atau yang biasa disebut *web* bukanlah *internet*, demikian pula sebaliknya. Namun demikian, internet adalah jaringan komputer global, sedangkan *web* bukan sekedar jaringan tetapi di dalamnya terdapat suatu set aplikasi komunikasi dan sistem perangkat lunak yang memiliki karakteristik sebagai berikut:

1. Umumnya terletak pada internet *host* dan *client*.
2. Umumnya menggunakan *protocol TCP/IP*.
3. Mengerti *HTML*.
4. Mengikuti model *client/server* untuk komunikasi data dua arah.
5. Memungkinkan client untuk mengakses *server* dengan berbagai *protocol* seperti *HTTP*, *FTP*, *Telnet*, dan *Ghosper*.
6. Memungkinkan *client* untuk mengakses informasi dalam berbagai media seperti teks, audio dan video.
7. Menggunakan model alamat *URL* (*Uniform Resource Locator*).

Konstitusi yang terdapat pada *web* sekarang ini berkembang dari ide dan konsep yang ditelurkan oleh Tim Berners Lee, seorang peneliti pada CERN Particle Physics Lab di Jenewa, Swiss. Pada tahun 1989, Berner Lee merumuskan suatu proposal tentang sebuah sistem *hypertext* yang memiliki tiga komponen yaitu:

1. Akses informasi yang universal. Setiap pengguna harus dapat mengakses seluruh informasi yang tersedia.
2. Antarmuka yang menyediakan akses terhadap berbagai jenis dokumen dan protocol.
3. Antarmuka yang konsisten untuk semua platform. Antarmuka ini harus menyediakan akses yang dapat digunakan oleh berbagai jenis informasi.

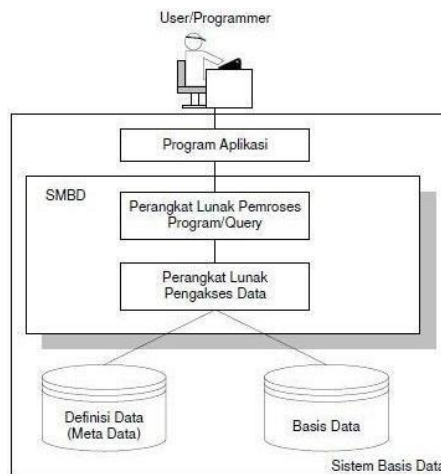
1.4.4 Website Profil

Website profile adalah website yang berisi rangkuman atau penjelasan singkat mengenai suatu perusahaan, sejarah berdirinya perusahaan tersebut, tujuan dan pencapaian yang diinginkan dalam masa depan, visi dan misi, produk yang ditawarkan, klien-klien yang telah menjalankan kerja sama, serta berbagai informasi lainnya terkait dengan perusahaan itu sendiri.(Solution, 2010:171).

1.5 Database

Basis data (*data base*) adalah kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

Sistem basis data pada dasarnya adalah sistem terkomputerisasi yang tujuan utamanya untuk memelihara informasi dan membuat informasi tersebut tersedia pada saat diperlukan.



Gambar 2.6 Konsep Sistem Basis data
(Sumber :Kompilasi Ramez Elmasri. dkk (1994))

Operasi-Operasi Yang Dapat Dilakukan Pada Basis Data :

- a. menambahkan file ke sistem database,
- b. menyisipkan data ke dalam suatu file,
- c. mengambil data dari suatu file,
- d. mengubah data pada suatu file,
- e. menghapus data dari suatu file,
- f. menyajikan informasi dari suatu atau sejumlah file,
- g. mengosongkan file.

Manfaat Sistem Basis Data :

1. Mengurangi duplikasi data (kemubaziran).
2. Menghindari inkonsistenan data.
3. Meningkatkan integritas data, dimana data selalu dalam keadaan valid; hal ini hanya dapat terjadi bila data hanya berada dalam satu tempat.
4. Adanya independensi data, sifat yang memungkinkan perubahan struktur file tidak mempengaruhi program dan sebaliknya; juga dapat berarti bahwa data bersifat tidak bergantung pada data lain.
5. Adanya sekuritas data, yang bermanfaat untuk menghindari pengaksesan data oleh yang tidak berhak.

6. Penggunaan data menjadi lebih mudah, dimana pada umumnya sistem manajemen database (*DataBase Management System = DBMS*) menyediakan fasilitas query yang memudahkan user untuk memperoleh informasi.
7. Salah satu komponen penting dalam sistem informasi, karena merupakan dasar dalam menyediakan informasi.
8. Menentukan kualitas informasi : akurat, tepat pada waktunya dan relevan. Informasi dapat dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya.

Sistem manajemen database dapat diartikan sebagai suatu program komputer yang digunakan untuk memasukkan, mengubah, menghapus, memanipulasi dan memperoleh data dengan praktis dan efisien.

1.5.1 Perancangan Basis Data

Permasalahan dalam perancangan basis data adalah bagaimana merancang struktur logikal dan fisik dari satu atau lebih basis data untuk memenuhi kebutuhan informasi yang diperlukan oleh pengguna sesuai dengan aplikasi-aplikasi yang ditentukan. Menurut Waliyanto (2000) Dengan permasalahan tersebut dapat ditentukan beberapa tujuan utama perancangan basis data, yaitu:

- a. Memenuhi kebutuhan informasi sesuai dengan yang diperlukan oleh pengguna untuk aplikasi tertentu.
- b. Mempermudah pemahaman terhadap struktur informasi yang tersedia dalam basis data
- c. Memberikan keterangan tentang persyaratan pemrosesan dan kemampuan sistem, seperti lamatidaknya mengakses data, kapasitas memori yang tersedia dan sebagainya.

Tahapan-tahapan proses perancangan untuk memenuhi tujuan tersebut adalah:

1. Mengumpulkan dan menganalisis persyaratan
2. Merancang konseptual basis data
3. Memilih Sistem Manajemen Basis Data
4. Merancang logikal basis data
5. Merancang fisik basis data (pemetaan model data)

6. Implementasi sistem basis data

Dalam pelaksanaan perancangan tersebut terdapat dua kegiatan yang dapat dilakukan secara paralel, yaitu perancangan struktur dan isi data (analisis data) dan perancangan pemrosesan data serta program aplikasi (analisis fungsional).

1.6 MySQL

My Structur Query (MySQL) adalah salah satu jenis basis data *server* yang populer. Kepopulerannya disebabkan MySQL menggunakan SQL (*Structured Query Language*) sebagai bahasa dasar untuk mengakses basis datanya.

MySQL merupakan *Software* yang bersifat *open source*, sesuai dengan namanya, bahasa standar MySQL adalah SQL. SQL adalah singkatan dari *Structured Query Language* dan sering disebut *sql*. SQL mulai dikembangkan pada akhir tahun 70-an di Laboratorium IBM, Sanjose, California. Sedangkan *MySQL Front* merupakan *Software* yang digunakan untuk memudahkan dalam mengatur *database* yang dibuat baik dalam penambahan Tabel, *record* dan *field* maupun menghapus dan mengedit *database* yang ada. SQL adalah bahasa yang digunakan untuk berkomunikasi dengan *database*. (Nugroho, 2004)

MySQL memiliki beberapa keunggulan yaitu sebagai berikut :

1. *Multiplatform*, MySQL tersedia pada berbagai *platform* seperti Windows, Linux, Unix, dan lain-lain.
2. Jaminan keamanan akses, MySQL mendukung pengamanan *database* dengan berbagai kriteria pengaksesan. Pembagian pengaturan kriteria hak akses pengguna tertentu dalam hal pengaksesan data tertentu. Misalnya ada pengguna yang hanya dapat melakukan penambahan data, dan ada pengguna lainnya yang dapat memanipulasi keseluruhan data, dan ada pengguna lainnya yang dapat memanipulasi keseluruhan data tertentu seperti menambah data, menghapus data dan mengubah data.

3. MySQL mendukung perintah SQL (*Structured Query Language*). SQL merupakan standar dalam pengaksesan *database relational*. Statemen SQL terdiri dari *reserved word* dan *user-defined word*. *Reserved word* adalah bagian yang tetap (*fixed*) dalam bahasa SQL dan memiliki makna yang tetap. *User-defined word* adalah bagian yang ditetapkan oleh pemakai untuk mewakili bermacam-macam objek sebuah *database*, seperti relasi, kolom, indeks, dan sebagainya. Statemen SQL adalah *case insensitive*, yaitu boleh ditulis dengan huruf besar atau huruf kecil.

Jenis perintah SQL:

- a. DDL (*Data Definition Language*)

Merupakan bahasa yang mendefinisikan data, atau kelompok perintah yang digunakan untuk melakukan pendefinisian *database* dan tabel. Kita dapat membuat tabel, mengubah strukturnya, menghapus tabel, membuat indeks untuk tabel dan sebagainya.

1. Membuat *Database*

Database dibuat melalui pernyataan *CREATE DATABASE*.

CREATE DATABASE nama database;

Nama *database* sebaiknya diawali dengan huruf besar, lalu dikombinasikan dengan huruf dan karakter. Sebaiknya nama *database* tidak mengandung spasi dan tanda baca.

2. Membuat Tabel

Setiap tabel dalam *database* harus didefinisikan dan dibuat. Hal ini dapat dilakukan dengan menggunakan perintah *CREATE TABLE*.

CREATE TABLE nama_tabel(nama_kolom_1 tipe_data([ukuran]),
nama_kolom_2 tipe_data([ukuran]),
nama_kolom_3 tipe_data([ukuran]),
...,
nama_kolom_n tipe_data([ukuran]));

Supaya tabel yang kita buat dapat berguna, kita perlu menambah atau mengisi informasi ke dalamnya. Kita bisa melakukan dengan perintah *INSERT*.

Bentuk umum 1 :

INSERT INTO namatabel (nama_kolom_1, nama_kolom_2, ...) *VALUES*
(nilai1, nilai2, ...);

Bentuk umum 2 :

INSERT INTO namatabel SET nama_kolom_1 = nilai1, nama_kolom_2 =
nilai2, ...;

Dibanding dengan cara pertama, maka cara kedua ini lebih jelas posisi data akan diletakkan pada kolom (*field*) mana saja. Hanya saja, bila jumlah data yang akan kita masukkan banyak, maka dengan cara kedua ini sangat merepotkan sekali.

Bentuk umum 3 :

INSERT INTO namatabel *VALUES* (nilai1, nilai2, ...);

Syntax perintah *INSERT INTO* diikuti nama tabel yang akan diisi, nama *field* yang akan diisi, dan *values* / isi data yang akan dimasukkan. Memilih *record* dalam *database MySQL* menggunakan perintah *select*. Ketika perintah ini dieksekusi dari *shell*, *MySQL* mencetak semua *record* yang cocok dengan query yang dimasukkan. Perintah yang paling gampang adalah *SELECT * FROM* namatabel;

Tanda * berarti “menampilkan isi dari semua *field* pada tabel” *from* tabel yang akan dibuka.

3. Mengubah Struktur Tabel

Ada saatnya kita sadar kalau ternyata struktur tabel yang pernah kita buat perlu penyempurnaan. Penyempurnaan bisa dalam hal penambahan kolom, pengubahan lebar kolom, penghapusan kolom, dan sebagainya. *ALTER TABLE* nama tabel jenis_pengubahan.

Jenis pengubahan yang dapat dilakukan antara lain penambahan kolom(*ADD*), penghapusan kolom dan indeks (*DROP*), penggantian nama kolom sekaligus jenis kolomnya (*RENAME*), dan pengubahan jenis kolom(*MODIFY*).

4. Melihat Struktur Tabel

Untuk melihat struktur tabel: *DESC* namatabel; atau dapat ditulis dengan *DESCRIBE* namatabel;

5. Mengganti Struktur Tabel

Untuk menambah kolom: *ALTER TABLE* namatabel *ADD* nama_kolom tipe_data([ukuran]);

Untuk menghapus kolom: *ALTER TABLE* namatabel *DROP COLUMN* nama_kolom;

Untuk mengubah nama tabel: *ALTER TABLE* namatabel_lama *RENAME* nama tabel_baru;

Untuk mengubah tipe data: *ALTER TABLE* namatabel *MODIFY* nama_kolom tipe_data_baru([ukuran_baru]);

Untuk mengubah nama kolom dan tipe data sekaligus: *ALTER TABLE* namatabel *CHANGE* nama_kolom_lama nama_kolom_baru tipe_data_baru;

b. DML (*Data Manipulation Language*)

DML (*Data Manipulation Language*) merupakan bagian dari *MySQL* yang digunakan untuk melakukan manipulasi dalam *database*. Manipulasi data pada tabel meliputi pemasukan, perubahan dan penghapusan data. Contohnya: *INSERT*, *UPDATE*, *DELETE*, dan *QUERY*.

1. Memasukkan Data dengan *INSERT*

Perintah *INSERT* digunakan untuk memasukkan data ke dalam tabel. Data yang dimasukkan pada kolom suatu tabel harus sesuai dengan struktur tabel yang dibuat pada perintah *CREATE TABLE*. Perintahnya adalah sebagai berikut:

```
INSERT INTO namatabel(nama_kolom1,nama_kolom2,...)VALUES (isi_kolom1,isi_kolom2,...);
```

Nama_kolom1, 2, dst adalah *field* atau kolom pada tabel yang akan diisi. Isi_kolom1, 2, dst merupakan data yang akan dimasukkan ke dalam *field* tersebut. Penulisan nilai disesuaikan dengan tipe datanya. Jika tipe datanya merupakan tipe data karakter maka data harus diapit dengan tanda petik. Untuk memasukkan data pada semua kolom, tanpa menulis nama_kolom dapat ditulis seperti di bawah ini:

INSERT INTO namatabel *VALUES* (isi_kolom1,isi_kolom2,...);

2. Penggunaan *SELECT*

Penulisan tanpa menyebutkan nama kolom dapat diberikan apabila akan mengisi ke seluruh kolom yang ada dalam tabel, urutan data dalam bagian *values* menunjukkan urutan kolom yang akan diisi. Jadi harus hati-hati, agar tidak tertukar urutan datanya. Untuk menampilkan data, perintah yang digunakan adalah *SELECT* yang diikuti beberapa pernyataan khusus berkenaan dengan tabel yang diseleksi.

a) Menampilkan data tanpa kriteria tertentu

Menampilkan semua data pada sebuah tabel

Bentuk umum :

*SELECT * FROM* nama_tabel;

b) Menampilkan data untuk kolom tertentu

Bentuk umum :

SELECT nama_kolom1,nama_kolom2,nama_kolom3,...

FROM nama_tabel;

c) Menampilkan data yang unik

Digunakan untuk menghindari redundan data pada kolom yang diinginkan.

Bentuk umum :

SELECT DISTINCT nama_kolom *FROM* nama_tabel;

d) Menampilkan data berdasarkan kriteria tertentu

KLAUSA WHERE

Klausula *WHERE* digunakan untuk menampilkan data dengan kondisi tertentu sehingga hasil yang dikeluarkan hanya sesuai dengan kriteria yang diinginkan.

e) *UPDATE* (Mengubah Data)

Perintah *UPDATE* digunakan untuk mengubah (memperbaiki) dalam suatu baris dalam suatu tabel).

Perintah :

UPDATE nama_tabel SET nama_kolom1 = nilai_baru1,
nama_kolom2 = nilai_baru2,... *WHERE* kondisi;

Catatan: klausula *WHERE* bersifat opsional. Bila klausula *WHERE* disebutkan, maka semua baris akan diubah.

f) *DELETE* (Menghapus Data)

Perintah *DELETE* digunakan untuk melakukan penghapusan baris dari suatu tabel yang memiliki kondisi yang dinyatakan dalam pernyataan kondisi.

Perintah :

DELETE FROM nama_tabel *WHERE* kondisi;

1.7 Flowchart

Flowchart merupakan gambar atau bagan yang memperlihatkan urutan dan hubungan antar proses beserta instruksinya. Gambaran ini dinyatakan dengan simbol. Dengan demikian setiap simbol menggambarkan proses tertentu. Sedangkan hubungan antar proses digambarkan dengan garis penghubung.

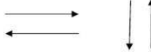



Flowchart ini merupakan langkah awal pembuatan program. Dengan adanya *flowchart* urutan proses kegiatan menjadi lebih jelas.

Simbol-simbol yang dipakai dalam flowchart dibagi menjadi 3 kelompok:

1.7.1 Flow Direction Symbols

1.7.1.1 Digunakan untuk menghubungkan simbol satu dengan yang lain









1.7.1.2 Disebut juga connecting line

	Simbol arus / <i>flow</i> , yaitu menyatakan jalannya arus suatu proses
	Simbol <i>communication link</i> , yaitu menyatakan transmisi data dari satu lokasi ke lokasi lain
	Simbol <i>connector</i> , berfungsi menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama
	Simbol <i>offline connector</i> , menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda

Gambar 2.7 Flow Direction Symbols
(Sumber : Rekayasa Perangkat Lunak, Roger S. Pressman(2002))

1.7.2 Processing symbols







Menunjukkan jenis operasi pengolahan dalam suatu proses / prosedur.

	Simbol <i>process</i> , yaitu menyatakan suatu tindakan (proses) yang dilakukan oleh komputer
	Simbol <i>manual</i> , yaitu menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer
	Simbol <i>decision</i> , yaitu menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban : ya / tidak
	Simbol <i>predefined process</i> , yaitu menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal
	Simbol <i>terminal</i> , yaitu menyatakan permulaan atau akhir suatu program
	Simbol <i>keying operation</i> , Menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai keyboard
	Simbol <i>offline-storage</i> , menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu
	Simbol <i>manual input</i> , memasukkan data secara manual dengan menggunakan online keyboard

Gambar 2.8 Processing Symbols
(Sumber : Rekayasa Perangkat Lunak, Roger S. Pressman(2002))

1.7.3 Input / Output symbols

Menunjukkan jenis peralatan yang digunakan sebagai media input atau output.

	Simbol <i>input/output</i> , menyatakan proses input atau output tanpa tergantung jenis peralatannya
	Simbol <i>punched card</i> , menyatakan input berasal dari kartu atau output ditulis ke kartu
	Simbol <i>magnetic tape</i> , menyatakan input berasal dari pita magnetis atau output disimpan ke pita magnetis
	Simbol <i>disk storage</i> , menyatakan input berasal dari disk atau output disimpan ke disk
	Simbol <i>document</i> , mencetak keluaran dalam bentuk dokumen (melalui printer)
	Simbol <i>display</i> , mencetak keluaran dalam layar monitor

Gambar 2.9 Input/Output Symbols
(Sumber : Rekayasa Perangkat Lunak, Roger S. Pressman(2002))

1.8 Data Flow Diagram (DFD)

Menurut Yourdon (1989, p139), *data flow diagram* adalah model atau alat yang digunakan untuk menggambarkan sistem sebagai jaringan dari sekumpulan proses fungsional, yang dihubungkan satu dengan lainnya oleh suatu aliran data dan meneruskannya menjadi data.

Ada tiga tingkatan dalam diagram aliran data, yaitu:

1. Diagram Konteks

Menggambarkan ruang lingkup sistem dari sistem yang digunakan. Diagram ini hanya memiliki satu proses yang menggambarkan sistem secara keseluruhan dan hubungan antara sistem dengan unit-unit di luar sistem tersebut.

2. Diagram Nol

Diagram yang menggambarkan proses-proses dan aliran data yang terjadi di dalam suatu sistem. Proses-proses ini dapat dipecah menjadi proses-proses dan aliran data yang lebih terperinci.

3. Diagram Rinci

Diagram yang menggambarkan rincian proses-proses yang ada pada diagram nol dan rincian proses-proses ini dapat dipecah menjadi proses-proses yang lebih terperinci.

Simbol-simbol yang ada pada diagram aliran data adalah sebagai berikut:

- Terminator

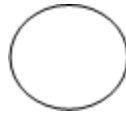
Merupakan simbol yang menggambarkan entitas yang dapat berupa orang, barang, kelompok, atau organisasi yang berhubungan dengan sistem.



Gambar 2.10 Simbol Terminator Dalam DFD

- Proses (*Bubble* atau *function* atau *transformation*)

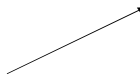
Proses menggambarkan bagian dari sistem yang mengolah masukan menjadi keluaran.



Gambar 2.11 Simbol Proses Dalam DFD

- Aliran (*flow*)

Aliran menggambarkan perpindahan informasi dari satu bagian ke bagian lain dari sistem. Awal panah menggambarkan asal data sedangkan arah panah menggambarkan tujuan.



Gambar 2.12 Simbol Aliran Data Dalam DFD

- Penyimpanan (*store*)

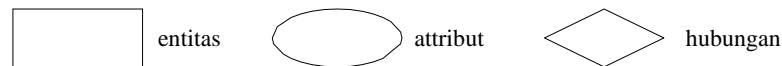
Penyimpanan data yang ada pada sistem dilambangkan dengan simbol ini.



Gambar 2.13 Simbol Penyimpanan Data Dalam DFD

1.9 Entity Relationship Diagram (ERD)

Perancangan konsep hubungan antar data memerlukan suatu pendekatan yang digunakan untuk menggambarkan hubungan antar data tersebut. Hubungan tersebut dapat dinyatakan dalam bentuk diagram ER (*Entity Relationship Diagram*). Diagram ER adalah suatu model yang digunakan untuk menggambarkan data dalam bentuk entitas, atribut dan hubungan antar entitas. Huruf “E” sendiri menyatakan entitas dan huruf “R” menyatakan hubungan. Beberapa notasi dasar yang digunakan dalam diagram ER ditunjukkan pada gambar 2.14 berikut.

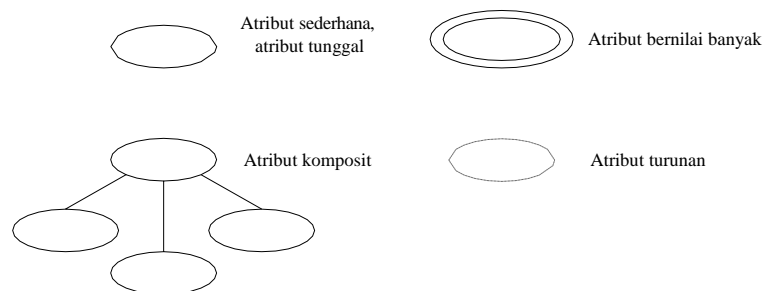


Gambar 2.14 Notasi Dasar Diagram ERD
(Sumber : Pressman,R.S(2002), <http://eprints.undip.ac.id>)

Diagram ERD memiliki beberapa komponen yaitu sebagai berikut :

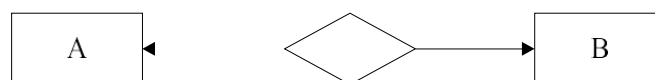
1.9.1 Entitas, merupakan sesuatu dalam dunia nyata yang keberadaannya tidak bergantung pada yang lain. Sebagai contoh, pegawai dalam sebuah organisasi adalah sebuah entitas. Entitas dapat berupa sesuatu yang nyata ataupun abstrak(berupa suatu konsep). Untuk lebih jelasnya, entitas dapat berupa seseorang, sebuah tempat, sebuah objek, sebuah kejadian atau sebuah konsep.

1.9.2 Atribut, setiap entitas dinyatakan oleh sejumlah atribut. Atribut adalah properti atau karakteristik yang terdapat pada setiap entitas. Atribut memiliki banyak jenis. Berikut ini adalah jenis-jenis atribut :



Gambar 2.15 Notasi Jenis-jenis Atribut
(Sumber : Pressman,R.S(2002))

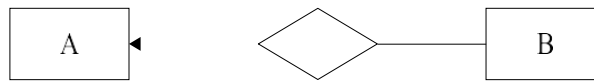
- 1.9.2.1 Atribut sederhana, atribut yang tidak dapat dipecah menjadi bagian-bagian yang lebih kecil yang masih memiliki makna.
- 1.9.2.2 Atribut komposit, dalam suatu kondisi tertentu atribut dapat dipecah menjadi bagian-bagian atribut yang lebih kecil dan tetap bermakna. Contoh, atribut Nama dapat dipecah menjadi Nama_Depan, Nama_Tengah, dan Nama_Belakang.
- 1.9.2.3 Atribut bernilai tunggal, atribut yang nilainya hanya satu untuk setiap kejadian dalam entitas.
- 1.9.2.4 Atribut bernilai banyak, atribut yang nilai atributnya dapat lebih dari satu dalam satu kejadian entitas.
- 1.9.2.5 Atribut turunan, nilai atribut dalam suatu tipe entitas dapat dihitung atau diturunkan dari nilai suatu atribut atau sejumlah atribut yang tersimpan dalam database atau dari nilai lain (misalnya jam sistem atau tanggal sistem). Atribut yang demikian dinamakan atribut turunan.
- 1.9.3 Relasi, menunjukkan adanya hubungan di antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Contoh: antara Dosen_Wali dan Mahasiswa terdapat hubungan berupa Bimbingan, antara Nasabah dan Pinjaman_Bank terdapat hubungan Peminjam. Relasi dinyatakan dengan kata kerja.
- 1.9.4 Kardinalitas relasi, menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain. Paling berguna dalam menggambarkan relasi antara dua himpunan entitas (*binary relationship sets*). (Imbar dkk., 2006) Kardinalitas relasi dapat berupa :
- 1.9.4.1 Satu ke satu (*one to one*), menyatakan bahwa setiap entitas pada tipe entitas A paling banyak berpasangan dengan satu entitas pada tipe entitas B dan begitu pula sebaliknya. Adapun contoh notasinya adalah sebagai berikut.



Gambar 2.16 Relasi Satu ke Satu
(Sumber : Imbar dkk., (2006))

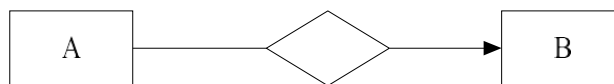
- 1.9.4.2 Satu ke banyak (*one to many*), menyatakan bahwa setiap entitas pada tipe entitas A dapat berpasangan dengan banyak entitas pada tipe entitas B.

Sedangkan setiap entitas pada B hanya dapat berpasangan dengan satu entitas pada tipe entitas A. Adapun contoh notasinya adalah sebagai berikut.



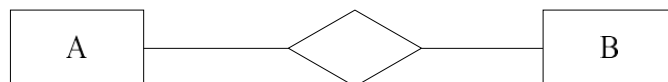
Gambar 2.17 Relasi Satu ke Banyak
(Sumber : Imbar dkk., (2006))

1.9.4.3 Banyak ke satu (*many to one*), menyatakan bahwa setiap entitas pada tipe A paling banyak berpasangan dengan satu entitas pada tipe entitas B dan setiap entitas pada entitas B dapat berpasangan dengan banyak entitas pada tipe entitas A. Adapun contoh notasinya adalah sebagai berikut.



Gambar 2.18 Relasi Banyak ke Satu
(Sumber : Imbar dkk., (2006))

1.9.4.4 Banyak ke banyak (*many to many*), menyatakan bahwa setiap entitas pada suatu tipe entitas A dapat berpasangan dengan banyak entitas pada tipe entitas B dan begitu juga sebaliknya. Adapun contoh notasinya adalah sebagai berikut.



Gambar 2.19 Relasi Banyak ke Banyak
(Sumber : Imbar dkk., (2006))

1.10 HTML

Menurut Sidik (2002), *Hypertext Markup Language* (HTML) yaitu salah satu bahasa *scripting* yang dapat menghasilkan halaman *website* sehingga halaman tersebut dapat diakses pada setiap komputer pengakses (*client*). Dokumen HTML merupakan dokumen yang disajikan dalam *browser web surfer*. Dokumen ini umumnya berisi informasi ataupun *interface* aplikasi dalam *internet*.

1.10.1 PHP (HYPERTEXT PREPROCESSOR)

PHP adalah kependekan dari PHP *Hypertext Preprocessor*, bahasa interpreter yang mempunyai kemiripan dengan bahasa C dan Perl yang mempunyai kesederhanaan dalam perintah, yang digunakan untuk pembuatan aplikasi web. (Sidik, 2004: 3)

PHP/F1 merupakan nama awal dari PHP (Personal Home Page / Form Interface). Dibataptama kali oleh Rasmus Lerdorf. PHP awalnya merupakan program CGI yang dikhususkan untuk menerima input melalui form yang ditampilkan dalam browser web. Dengan menggunakan PHP maka maintenance suatu situs web menjadi lebih mudah. Proses update dapat dilakukan dengan menggunakan aplikasi yang dibuat dengan menggunakan script PHP. PHP merupakan script untuk pemrograman script web server-side, script yang membuat dokumen HTML secara *on the fly*, dokumen HTML yang dihasilkan dari suatu aplikasi bukan dokumen HTML yang dibuat dengan menggunakan editor teks atau editor HTML.

1.10.1 Sintaks Program PHP

PHP adalah bahasa yang dirancang untuk mudah diletakkan di dalam kode HTML. Banyak dijumpai kode PHP yang menyatu dengan kode HTML. Kode PHP diawali dengan tag `<?php` dan diakhiri dengan tag `?>`.

Berikut ini contoh kode PHP yang sederhana.

```
<?php
    Echo "hello world"
?>
```

Perintah echo di dalam PHP berguna untuk mencetak nilai, baik teks maupun numerik ke layar *web browser*. Selain echo, kita juga dapat menggunakan perintah print yang mempunyai fungsi sama dengan echo.

1.10.2 Variable Pada PHP

Variabel digunakan untuk menyimpan suatu nilai, seperti text, angka atau array. Ketika sebuah variabel dibuat, variabel tersebut dapat dipakai berulang-ulang. Pada PHP semua variabel harus dimulai dengan karakter '\$'. Variabel PHP tidak perlu dideklarasikan dan ditetapkan jenis datanya sebelum kita menggunakan variabel tersebut. Hal itu berarti pula bahwa tipe data dari variabel dapat berubah sesuai dengan perubahan konteks yang dilakukan oleh user. Secara tipikal, variabel PHP cukup diinisialisasikan dengan memberikan nilai kepada variabel tersebut.

Contoh berikut akan mencetak "PHP" :

```
<?php
    Echo "hello world"
?>

$text = "PHP";
print "$text";
```

Identifier dalam PHP adalah *case-sensitive*, sehingga \$text dengan \$Text merupakan *variabel* yang berbeda. *Built-in function* dan *structure* tidak *case sensitive*, sehingga *echo* dengan ECHO akan mengerjakan perintah yang sama. Identifier dapat berupa sejumlah huruf, digit/angka, *underscore*, atau tanda dollar tetapi *identifier* tidak dapat dimulai dengan digit/angka. Adapun aturan penamaan variabel :

1.10.2.1 Nama variabel harus diawali dengan sebuah huruf atau garis bawah (*underscore*) “_”

1.10.2.2 Nama variabel hanya boleh mengandung karakter *alpha-numeric* dan *underscore* (a-Z, 0-9, dan _)

1.10.2.3 Nama variabel tidak boleh mengandung spasi.

1.11 XAMPP

XAMPP adalah sebuah paket instalasi mudah Apache Server yang sudah termasuk di dalamnya MySQL, PHP, dan Perl (Adhi dan Andi, 2006). XAMPP dikembangkan dari sebuah tim proyek bernama Apache Friends, yang terdiri dari Tim Inti (*Core Team*).

XAMPP merupakan paket PHP yang berbasis *open source* yang dikembangkan oleh sebuah komunitas Open Source. Dengan menggunakan XAMPP tidak dibingungkan dengan penginstalan program-program lain, karena semua kebutuhan telah tersedia oleh XAMPP. Yang terdapat pada XAMPP di antaranya : Apache, MySQL, PHP, FileZilla FTP Server, PHPmyAdmin dll.

Xampp merupakan tools yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan meng-install Xampp maka tidak perlu lagi melakukan instalasi dan konfigurasi web server Apache, PHP dan MySQL secara manual. Xampp akan menginstalasi dan mengkonfigurasinya secara otomatis.



Gambar : 2.21 XAMPP

Fungsi XAMPP adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, MySQL database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU General Public License dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis.

1.12 BLACK BOX TESTING

Black-Box Testing merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program.

Ciri-Ciri Black Box Testing

1.12.1 Black box testing berfokus pada kebutuhan fungsional pada *software*, berdasarkan pada spesifikasi kebutuhan dari *software*.

1.12.2 *Black box testing* bukan teknik alternatif daripada *white box testing*.

Lebih daripada itu, ia merupakan pendekatan pelengkap dalam mencakup *error* dengan kelas yang berbeda dari metode *white box testing*.

1.12.3 *Black box testing* melakukan pengujian tanpa pengetahuan detil struktur internal dari sistem atau komponen yang dites. juga disebut sebagai *behavioral testing*, *specification-based testing*, *input/output testing* atau *functional testing*.

Kategori *error* yang akan diketahui melalui *black box testing* :

1. Fungsi yang hilang atau tak benar.
2. *Error* dari antar-muka.
3. *Error* dari struktur data atau akses eksternal *database*.
4. *Error* dari kinerja atau tingkah laku.
5. *Error* dari inisialisasi dan terminas

BAB II

PEMBAHASAN

4.1. Implementasi

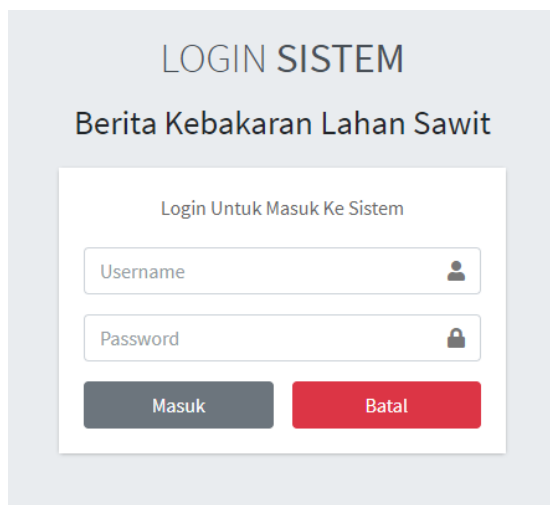
Tahap implementasi merupakan tahap di mana desain basis data ataupun desain *interface* yang telah dibuat sebelumnya diterjemahkan ke dalam kode-kode dengan menggunakan bahasa pemrograman. Tahap implementasi merupakan tahapan untuk membuktikan bahwa website yang dibuat telah berfungsi dengan baik.

4.1.1. Implementasi Program

4.1.1.1. Tampilan Website untuk Admin

1. Halaman Login Admin

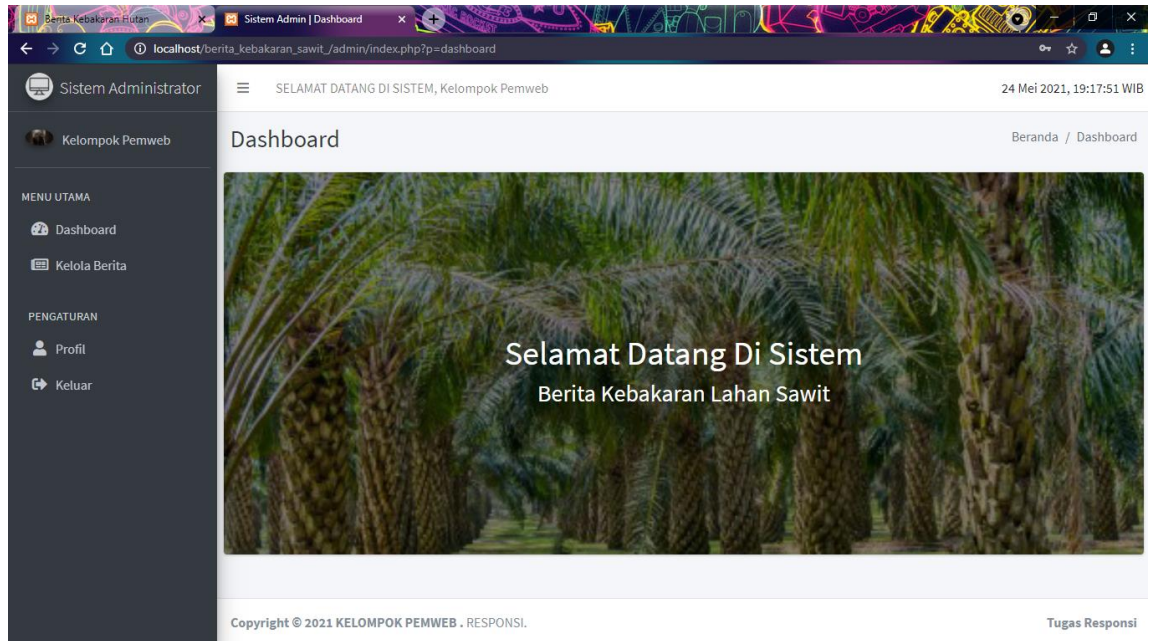
Gambar 4.1 merupakan halaman login. Halaman ini harus dilakukan sebelum seorang admin masuk ke dalam sistem. Seorang admin harus login dengan menggunakan username dan password yang telah diberikan. Terlihat pada halaman ini terdapat button Login yang digunakan sebagai button aksi saat admin telah selesai menginputkan data. Apabila admin belum memiliki akun, maka admin harus melakukan Registrasi terlebih dulu.



Gambar 4.1 Halaman Login Admin

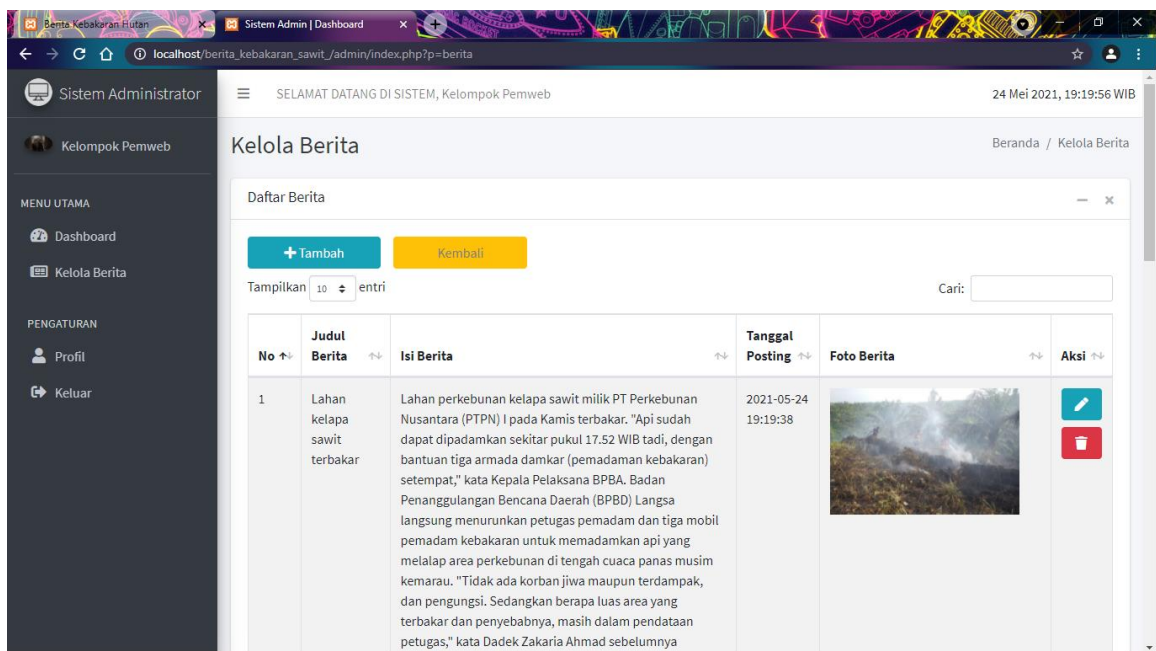
2. Halaman Beranda Admin

Gambar 4.2 merupakan implementasi halaman beranda Admin saat setelah admin berhasil login dengan akun penggunaanya.



Gambar 4.2 Halaman Beranda Admin

3. Halaman Kelola Berita

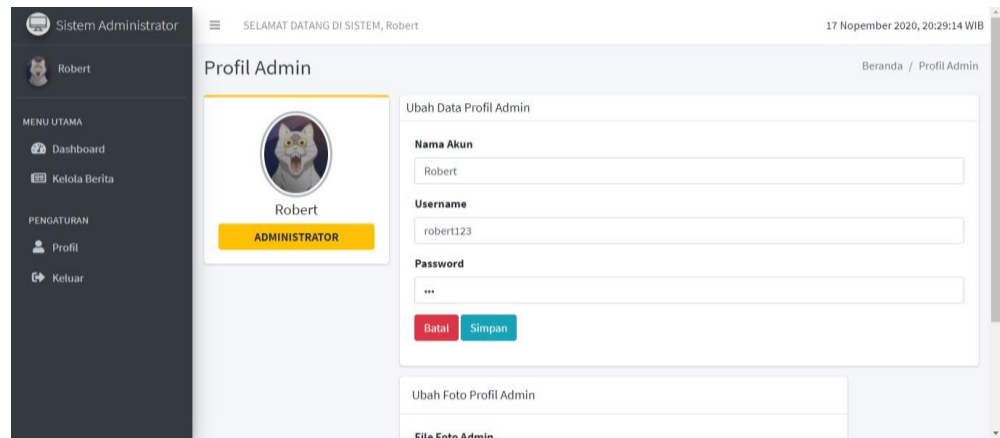


Gambar 4.3 merupakan implementasi website untuk halaman kelola berita.

Gambar 4.3 Halaman Kelola Berita

4. Halaman Profil Admin

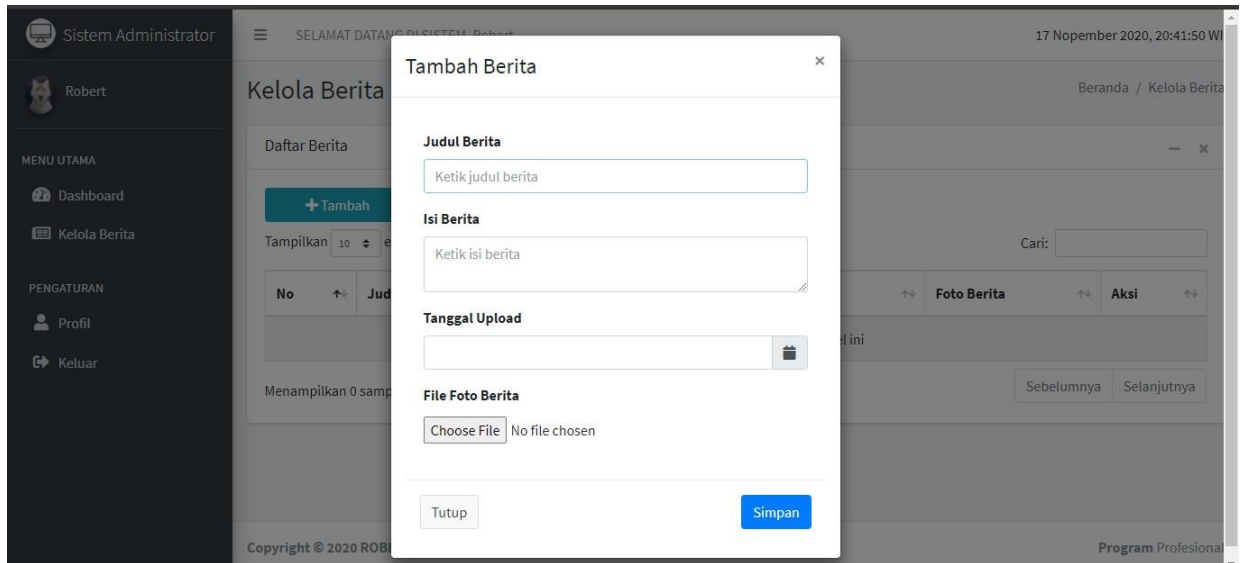
Gambar 4.4 merupakan implementasi website untuk halaman profil admin.



Gambar 4.4 Halaman Profil Admin

5. Halaman Tambah Berita

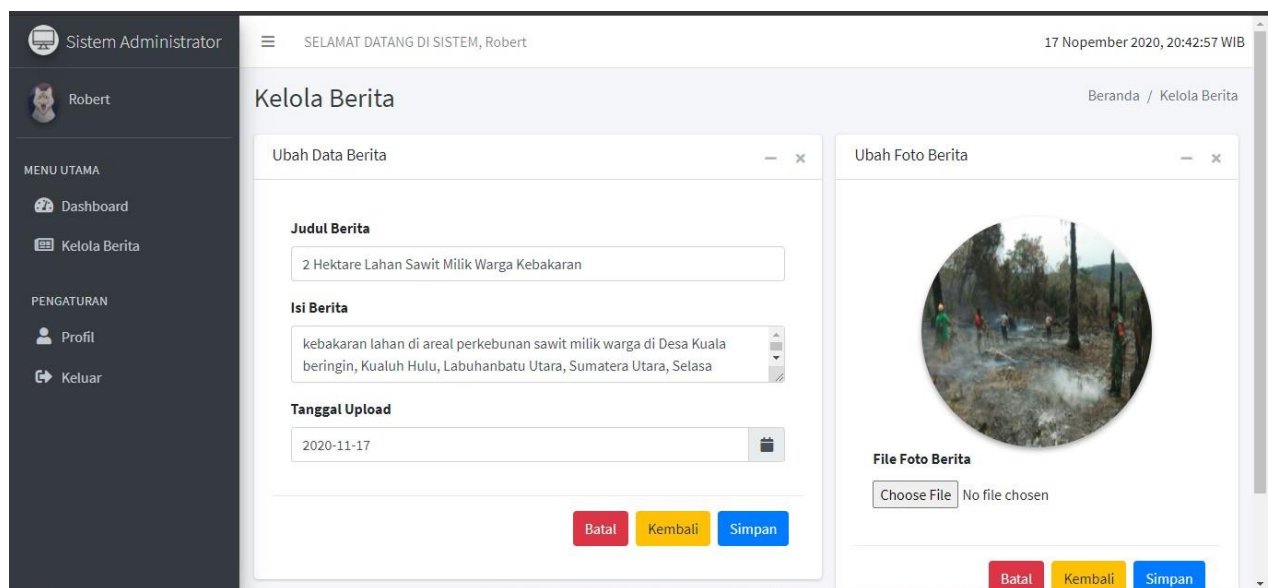
Gambar 4.5 merupakan implementasi website untuk halaman tambah data berita.



Gambar 4.5 Halaman Tambah Berita

6. Halaman Ubah Berita

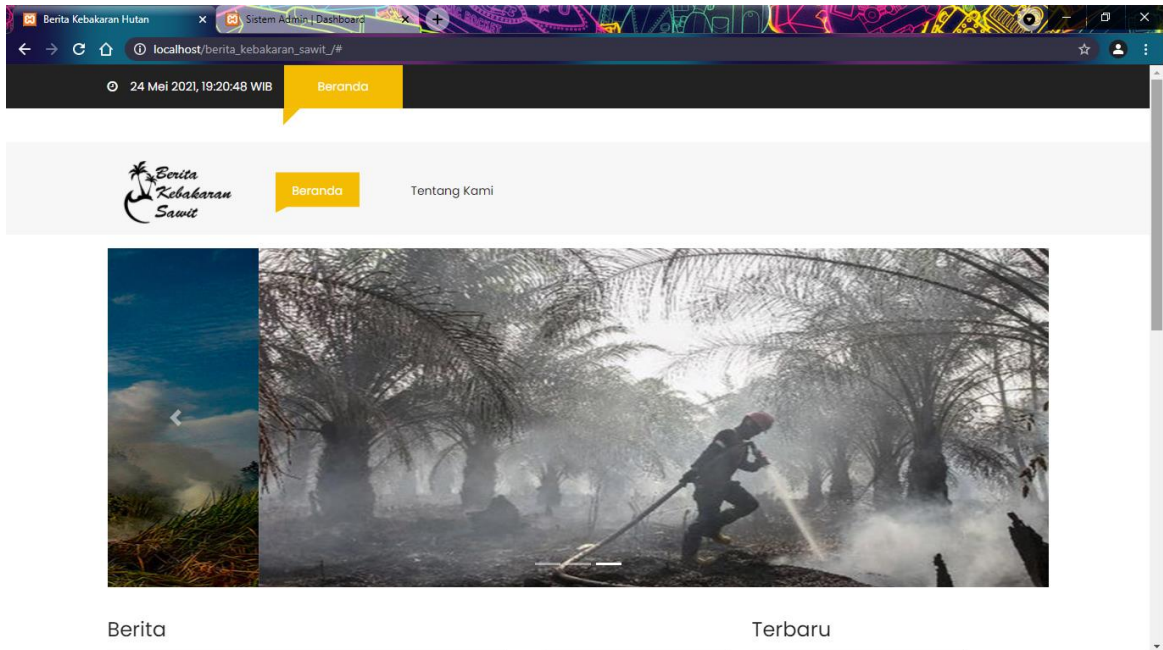
Gambar 4.6 merupakan implementasi website untuk halaman ubah data berita.



Gambar 4.6 Halaman Ubah Berita

4.1.1.2. Tampilan Website untuk Pengunjung

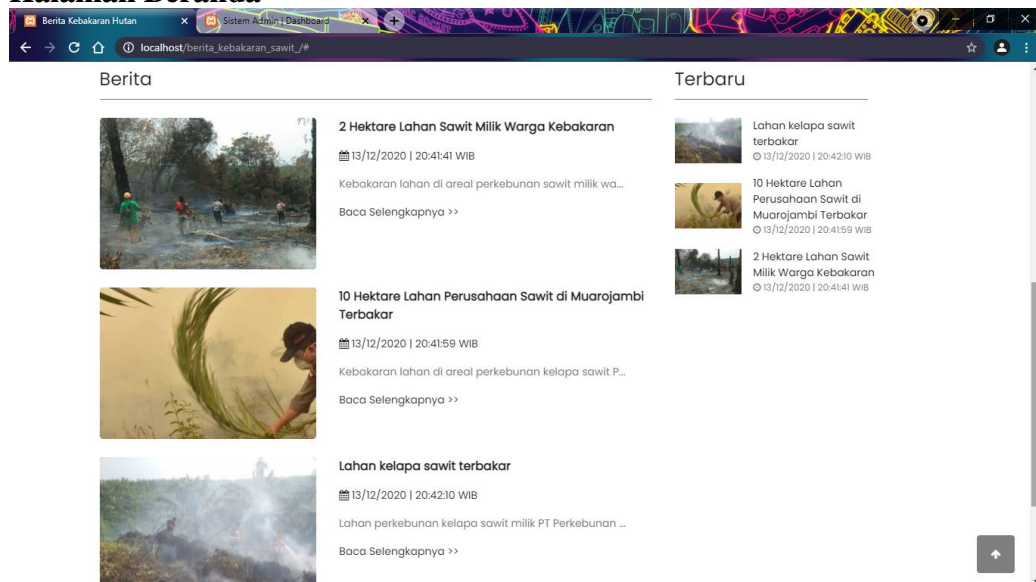
1. Halaman Beranda



Gambar 4.7 merupakan implementasi website untuk halaman beranda.

Gambar 4.7 Halaman Berita

2. Halaman Beranda

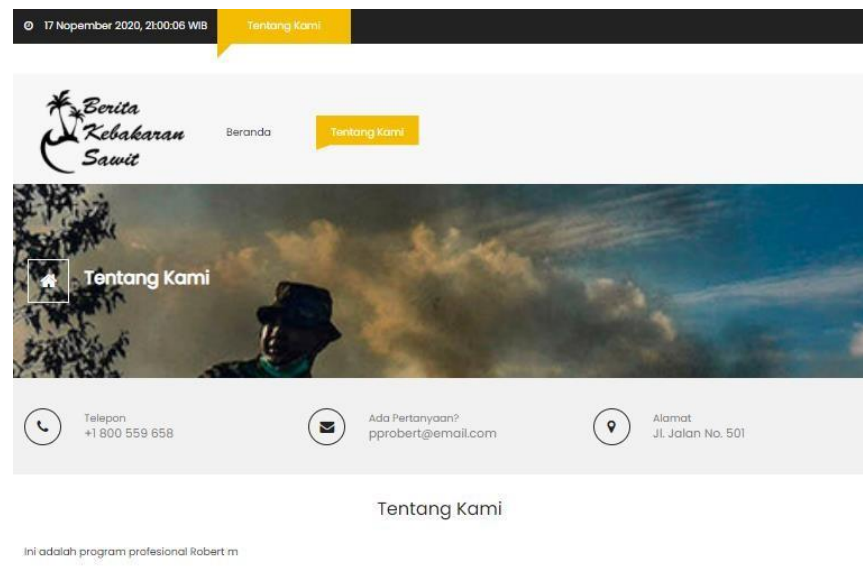


Gamabar 4.8 merupakan implementasi website untuk halaman beranda

Gambar 4.8 Halaman Berita

3. Halaman Tentang Kami

Gambar 4.8 merupakan implementasi website untuk halaman tentang kami.



Gambar 4.9 Halaman Tentang Kami

4.2. Integration and System Testing (Pengintegrasian dan Pengujian Sistem)

Penyatuan unit program kemudian uji secara keseluruhan. Pengujian yang dilakukan untuk website ini adalah **Black Box Testing**. Pengujian blackbox digunakan untuk mengetahui seluruh fungsi secara spesifik dari website ini, apakah sudah berjalan dengan baik atau tidak.

4.2.1. Pengujian pada halaman website

1. Admin

Tabel 4.1 Tabel *Blackbox Testing* Halaman Admin

No	Kondisi Awal	Kondisi Yang Diinginkan	Kondisi Hasil	Hasil
1.	Input alamat website halaman admin	nampilkan halaman login admin	nampilkan halaman login admin	√
2.	Login	masuk ke halaman beranda admin	masuk ke halaman beranda admin	√
3	Klik menu kelola berita	masuk ke halaman kelola berita	masuk ke halaman kelola berita	√
3	Klik tombol Tambah	masuk ke halaman tambah berita	masuk ke halaman tambah berita	√
4	Klik tombol Ubah	masuk ke halaman Ubah berita	masuk ke halaman Ubah berita	√
5	Klik tombol Hapus	Data Terhapus	Data Terhapus	√
6	Klik menu Profil	masuk ke halaman profil admin	masuk ke halaman profil admin	√

2. Pengunjung

4.2 Tabel *Blackbox Testing* Halaman Pengunjung

No	Kondisi Awal	Kondisi Yang Diinginkan	Kondisi Hasil	Hasil
1	Klik menu beranda	masuk ke halaman beranda	masuk ke halaman beranda	√
2	Klik menu Tentang Kami	masuk ke halaman tentang kami	masuk ke halaman tentang kami	√

BAB III

KESIMPULAN

5.1 Kesimpulan

Adapun kesimpulan yang didapat dari pembuatan “*Website Berita Kebakaran Lahan Sawit*” ini yaitu:

Website ini dibangun menggunakan bahasa pemrograman *web PHP*, *database MySQL*, *HTML*, *CSS* dan *Javascript*. Dalam pengembangan perangkat lunak *website* ini menggunakan metodologi *waterfall* dengan tahapan yaitu *Requirement Analysis And Definition* yaitu melakukan analisis kebutuhan yang akan digunakan dalam *website*.

Selanjutnya tahapan *System And Software Design* yaitu melakukan analisis kebutuhan *website berita*, membuat *flowchart* bagaimana proses data dibuat, membuat desain *database* dengan *ERD (Entity Relationship Diagram)*, desain alur data pada *website* dengan *DFD (Data Flow Diagram)* dan desain *interface website* berita.

Setelah tahapan Analisis dan Desain dilakukan tahapan selanjutnya adalah tahapan *Implementation And Unit Testing* dimana pada tahap ini penulisan program dengan menggunakan *software SublimeText* untuk penulisan baris program, *Xampp* dan *MySQL* sebagai perangkat untuk pembuatan *database* dan untuk pengujian unit melibatkan verifikasi bahwa setiap unit telah memenuhi spesifikasinya. Tahapan selanjutnya adalah *Integration And System Testing* yaitu melakukan integrasi aplikasi yang sudah di analisis dan dibuat untuk di lakukan uji coba apakah sudah sesuai dengan analisis yang dibuat. Dan untuk pengujian sistem dilakukan dengan menggunakan *blackbox testing*.

DAFTAR PUSTAKA

Simarmata , Janner & paryudi, iman. *Basis Data edisi II*.Yogyakarta: ANDI,2010.

Nugroho, Adi.*Perancangan & implementasi sistem Basis Data*, Yogyakarta: ANDI,2011.

Simarmata, Janner dan Paryudi, Iman. 2010. *Basis Data*. Yogyakarta : Andi

Simarmata, Janner. 2011. *Perancangan Basis Data*. Yogyakarta : Andi

Politeknik Telkom, Tim Dosen. 2012. *Analisa Desain Sistem Indormasi (e-book)*.
Politeknik Telkom Bandung, Bandung.

Narbuko, Cholid dan Abu Achmadi. 2012. *Metodologi Penelitian*. Jakarta : Bumi
Aksara

Burch, J.G.2012. *System, Analysis, Design, and Implementation*, Boyd & Fraser
Publishing Company.

Asosiasi Digital Marketing, 2020. *Fungsi User Interface*.

