

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1. Penelitian Terdahulu**

Penelitian terdahulu ini menjadi satu acuan penulis dalam membuat laporan akhir sehingga dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Berikut merupakan penelitian terdahulu berupa beberapa jurnal yang terkait dengan judul laporan akhir penulis.

Pada penelitian sebelumnya yang dilakukan (Murtikasari, dkk, 2020) dalam jurnal yg berjudul “**Efektivitas Pengelolaan Arsip Elektronik di Indonesia**” berdasarkan hasil dan pembahasan di atas, pengelolaan arsip elektronik di Indonesia terdiri atas beberapa tahap, yaitu penciptaan dan penyimpanan, distribusi dan penggunaan, pemeliharaan dan disposisi. Namun dalam pelaksanaannya belum maksimal karena tidak semua organisasi yang menerapkan sistem pengelolaan arsip elektronik memberikan pelatihan kearsipan kepada pengelola arsip elektronik, sehingga pengelola kurang memiliki pengetahuan dan keterampilan untuk mengelola arsip elektronik. Organisasi yang menerapkan sistem pengelolaan arsip elektronik tidak semuanya memiliki sarana dan prasarana yang memadai, sehingga menjadi kendala dalam kegiatan kearsipan. Oleh karena itu, dapat disimpulkan bahwa efektivitas pengelolaan arsip elektronik di Indonesia belum sepenuhnya tercapai.

Pada penelitian sebelumnya yang dilakukan (Listiyani, 2019) dalam jurnal yang berjudul “**Analisis Autentikasi dan Pengelolaan Arsip Elektronik di Dinas Kearsipan dan Perpustakaan Kabupaten Semarang**” Penelitian ini bertujuan untuk mengetahui usaha dan upaya yang dilakukan oleh Dinas Kearsipan dan Perpustakaan Kabupaten Semarang dalam menjaga keaslian arsip hasil digitalisasi yang dimilikinya.

Melalui pengelolaan arsip secara elektronik, diharapkan akan meningkatkan kualitas pelayanan arsip kepada semua pihak, termasuk pimpinan dalam rangka pengambilan keputusan.

## 2.2. Manajemen

(Follet dalam Sulastri, 2012) manajemen adalah seni dalam menyelesaikan tugas melalui perantara. Dalam hal ini, manajemen dapat diartikan sebagai suatu kegiatan yang dilakukan oleh seorang manager untuk mengarahkan bawahan atau orang lain dalam menyelesaikan pekerjaan demi tercapainya sebuah tujuan.

Menurut (Griffin dalam Sulastri, 2012) manajemen adalah proses perencanaan, organisasi, koordinasi, dan kontrol pada sumber daya agar tujuan tercapai secara efektif dan efisien. Efektif di sini maksudnya tujuan tercapai sesuai rencana, dan efisien berarti bahwa manajemen dilakukan secara cermat, terorganisir, dan tepat waktu.

Jadi, manajemen adalah proses merencanakan, mengorganisasi, mengarahkan, dan mengendalikan kegiatan untuk mencapai tujuan organisasi secara efektif dan efisien dengan menggunakan sumber daya organisasi.

## 2.3. Arsip

### 2.3.1 Pengertian Arsip

Pengertian arsip menurut Kamus Besar Bahasa Indonesia (arsip, n.d.) adalah dokumen tertulis (surat, akta, dan sebagainya), lisan (pidato, ceramah, dan sebagainya), atau bergambar (foto, film, dan sebagainya) dari waktu yang lampau, disimpan dalam media tulis (kertas), elektronik (pita kaset, pita video, disket komputer, dan sebagainya), biasanya dikeluarkan oleh instansi resmi, disimpan dan dipelihara di tempat khusus untuk referensi.

Sedangkan menurut UU No. 43 tahun 2009 bahwa arsip adalah rekaman kegiatan atau peristiwa dalam berbagai bentuk dan media. Sementara istilah arsip dalam Bahasa Inggris disebut "*record*". Kata "*record*" diartikan juga sebagai catatan (*Record*, n.d.). ISO 15489 mendefinisikan *record* (catatan) adalah catatan informasi yang direkam yang dibuat, diterima, dan dipelihara sebagai bukti oleh organisasi atau orang dalam menjalankan kewajiban hukum atau dalam transaksi bisnis. Berdasarkan berbagai pengertian tersebut maka kata arsip sering digunakan secara silih berganti dengan kata catatan atau rekaman atau rekod. Arsip dalam pengelolaannya dibedakan menjadi dua yaitu arsip dinamis dan arsip statis.

(Barthos, 2016) mengemukakan tentang jenis-jenis arsip, yakni (1) arsip dinamis, dilihat dari kegunaannya dibedakan atas arsip aktif dan arsip inaktif, dan (2) arsip statis. Hal yang perlu dipahami oleh petugas kearsipan suatu organisasi adalah memperhatikan setiap pengelolaan dokumen-dokumen penting agar tertib dan teratur, sehingga arsip tersebut dapat digunakan seoptimal mungkin (Wijaya, Wiyono, & Bafadal, 2018).

(Sugiarto dan Wahyono, 2015) mengatakan bahwa kearsipan merupakan dasar dari pemeliharaan surat: kearsipan mengandung proses penyusunan dan penyimpanan surat-surat sedemikian rupa, sehingga surat/berkas dapat diketemukan kembali bila diperlukan. Sifat yang paling penting yang harus dimiliki oleh suatu sistem kearsipan adalah keterpercayaan dan accessibility, disamping dari sifat lainnya seperti kerapian, kebersihan dan lainnya.

Penjelasan diatas dapat disimpulkan dari istilah bahasa dan beberapa pengertian diatas bahwa arsip berperan sebagai pusat ingatan atau sumber informasi dan sebagai alat pengawasan yang sangat diperlukan oleh setiap organisasi dalam rangka kegiatan. Arsip sangat membantu organisasi dalam menjalankan suatu kegiatan seperti kegiatan perencanaan, penganalisaan, pengembangan, perumusan kebijaksanaan, pengambilan keputusan, pembuatan laporan, pertanggung jawaban, penilaian dan pengendalian setepat-tepatnya. Sedangkan,

Kearsipan merupakan salah satu jenis pekerjaan kantor atau pekerjaan tata usaha yang banyak dilakukan oleh badan pemerintahan, maupun badan swasta. Kearsipan menyangkut pekerjaan yang berhubungan dengan penyimpanan arsip atau surat-surat, dan dokumen kantor lainnya mulai dari penciptaan, penerimaan, pengumpulan, pengaturan, pengendalian, pemeliharaan dan perawatan serta penyimpanan dokumen menurut sistem tertentu yang saat dibutuhkan dapat ditemukan dengan cepat dan tepat ditemukan.

### **2.3.2 Arsip Elektronik**

(Rifauddin dalam Almahdi dan Pahlevi, 2020) menyimpulkan pada dasarnya

arsip elektronik merupakan catatan yang dibuat atau disimpan dalam bentuk elektronik, baik analog atau digital. Arsip-arsip yang disimpan dan diolah dalam suatu format dimana hanya mesin komputer yang dapat memprosesnya.

(Mulyadi dalam Almafidi dan Pahlevi, 2020) menyatakan arsip elektronik merupakan kumpulan informasi yang telah direkam menggunakan teknologi komputer sebagai dokumen elektronik agar dapat dilihat dan dipergunakan kembali.

Jadi, arsip elektronik adalah catatan dari kumpulan data menggunakan teknologi yang disimpan dalam bentuk data pindaian(scan) pada mesin komputer agar dokumen mudah dilihat, dikelola, ditemukan dan dipergunakan kembali.

## **2.4. Website**

### **2.4.1 Pengertian Website**

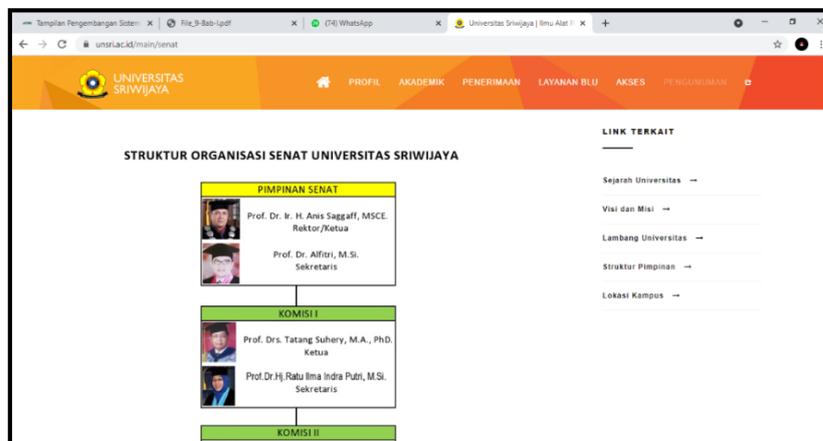
*Website* atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan/atau gabungan dari semuanya baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing dihubungkan dengan jaringan-jaringan halaman. Hubungan antara satu halaman *web* dengan halaman *web* yang lainnya disebut *hyperlink*, sedangkan teks yang dijadikan media penghubung disebut *hypertext* (Batubara, 2015).

### **2.4.2 Jenis – Jenis Website**

Seiring dengan perkembangan teknologi informasi yang begitu cepat, *website* juga mengalami perkembangan yang sangat berarti. Dalam pengelompokan jenis *web*, lebih diarahkan berdasarkan kepada fungsi, sifat atau *style* dan bahasa pemrograman yang digunakan.

1. Adapun jenis – jenis *web* berdasarkan sifat atau *style*-nya yaitu :
  - a. *Website* Dinamis, merupakan sebuah *website* yang menyediakan *content* atau isi yang selalu berubah-ubah setiap saat. Bahasa pemrograman yang digunakan antara lain PHP, ASP, NET dan memanfaatkan *database* MySQL atau MS SQL.

- b. *Website Statis*, merupakan *website* yang *content*-nya sangat jarang diubah. Bahasa pemrograman yang digunakan adalah HTML dan belum memanfaatkan *database*.
2. Fungsinya, *website* terbagi atas :
- Personal website**, *website* yang berisi informasi pribadi seorang.
  - Commercial website**, *website* yang dimiliki oleh sebuah perusahaan yang bersifat bisnis.
  - Government website**, *website* yang dimiliki oleh instansi pemerintahan, pendidikan yang bertujuan memberikan pelayanan kepada pengguna.
  - Non-Profit Organization website**, dimiliki oleh organisasi yang bersifat *non-profit* atau tidak bersifat bisnis.
3. Segi bahasa pemrograman yang digunakan, *website* terbagi atas :
- Server Side*, merupakan *website* yang menggunakan bahasa pemrograman yang tergantung kepada tersedianya *server*. Seperti PHP, ASP, NET dan lain sebagainya.
  - Client Side*, adalah *website* yang tidak membutuhkan *server* dalam menjalankannya, cukup diakses melalui browser saja (Batubara, 2015).

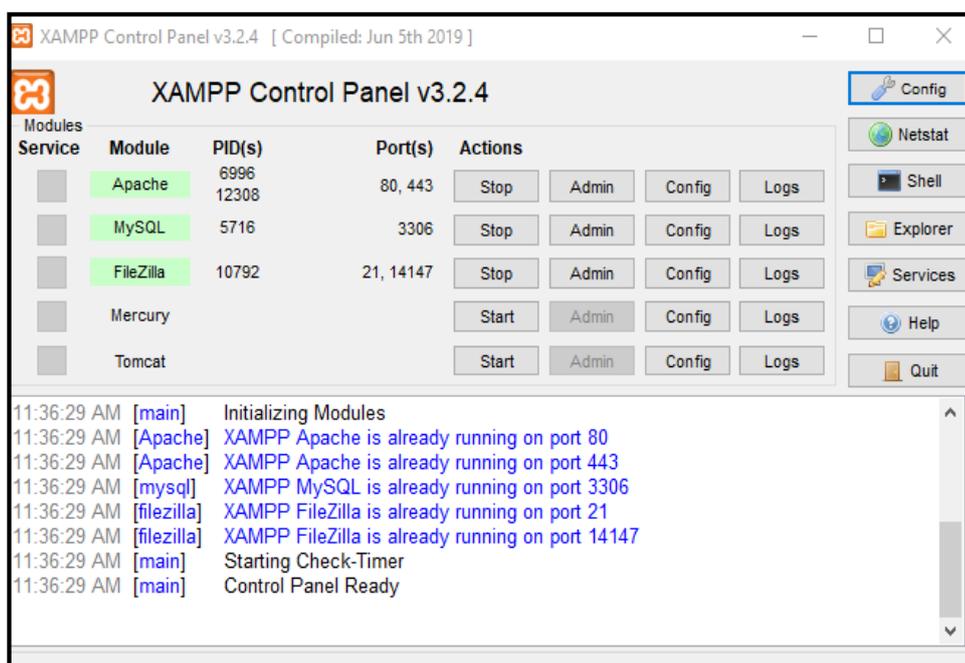


**Gambar 2.1 Contoh Tampilan Website**  
(Sumber : unsri.ac.id)

## 2.5. Aplikasi XAMPP

XAMPP adalah salah satu paket instalasi apache, PHP, dan MySQL secara instan yang dapat digunakan untuk membantu proses instalasi ketiga produk tersebut (Sihotang, 2018).

XAMPP adalah sebuah paket kumpulan *software* yang terdiri dari *apache*, *mysql*, *phpMyAdmin*, *PHP*, *Perl*, *Freetype2*, dan lainnya. XAMPP berfungsi untuk memudahkan instalasi lingkungan PHP, di mana biasanya lingkungan pengembangan *web* memerlukan PHP, apache, MySQL dan *phpMyAdmin* serta *software-software* yang terkait dengan pengembangan *web*. Dengan menggunakan XAMPP, kita tidak perlu meng-*install* aplikasi-aplikasi tersebut satu persatu (Sofwan, 2017).



**Gambar 2.2** Tampilan XAMPP  
(Sumber : Aplikasi XAMPP)

### 2.5.1 Apache

Server HTTP *Apache* atau *Server Web/WWW Apache* adalah *server web* yang dapat dijalankan di banyak sistem operasi (*Unix*, *BSD*, *Linux*, *Microsoft Windows*, serta *platform* lainnya) yang berguna untuk melayani dan memfungsikan situs *web*. Protokol yang digunakan untuk melayani fasilitas *web/www* ini

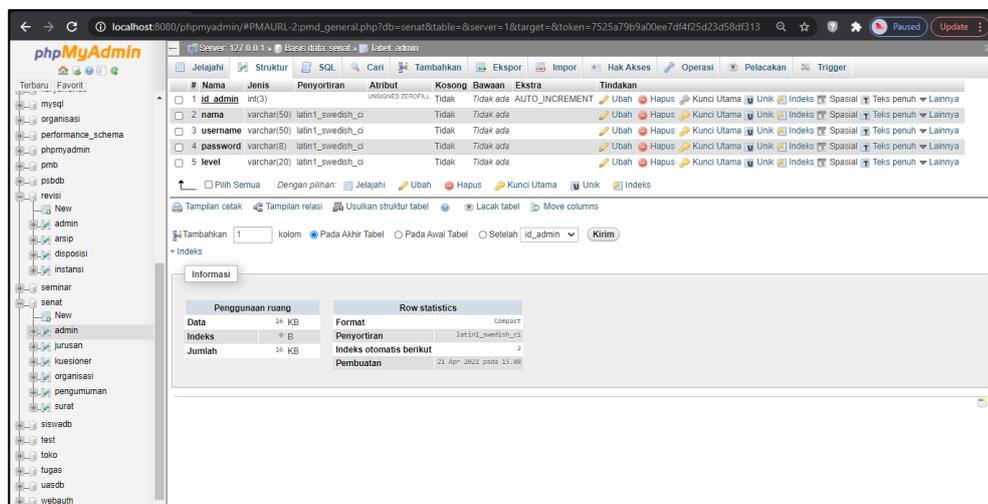
menggunakan HTTP. *Apache* memiliki fitur-fitur canggih seperti pesan kesalahan yang dapat dikonfigurasi, autentikasi berbasis basis data dan lain-lain. *Apache* juga didukung oleh sejumlah antarmuka pengguna berbasis grafik (GUI) yang memungkinkan penanganan *server* menjadi mudah (Wardani, 2013).

## 2.5.2 PHPMyAdmin

Pengertian *PHPMyAdmin* sebagaimana dikemukakan (Madcoms dalam Soca, 2018) *PHPMyAdmin* adalah sebuah aplikasi open source yang berfungsi untuk memudahkan manajemen MySQL. Dengan menggunakan *PHPMyAdmin*, Anda dapat membuat database, membuat tabel, meng-insert, menghapus dan meng-update data dengan GUI dan terasa lebih mudah, tanpa perlu mengetikkan perintah SQL secara manual.

*PHPMyAdmin* adalah sebuah aplikasi pemrograman yang digunakan untuk *management database* melalui *browser* (web) untuk mengontrol data mereka dan isi web yang akan ditampilkan dalam sebuah *website* yang mereka buat tanpa harus menggunakan perintah (*command*) SQL (Nurmalasari, dkk, 2019).

Berdasarkan penjelasan diatas maka penulis menyimpulkan bahwa *PHPMyAdmin* adalah perangkat lunak *open source* yang digunakan untuk menggunakan *MySQL*.



**Gambar 2.3** Tampilan *PHP MyAdmin*  
(Sumber : localhost/phpmyadmin)

### 2.5.3 PHP (Hypertext Preprocessor)

PHP singkatan dari PHP *Hypertext Preprocessor* yaitu bahasa pemrograman *web server-side* yang bersifat *open source*. PHP merupakan *script* yang terintegrasi dengan HTML dan berada pada *server* (*server side HTML embedded scripting*). PHP adalah *script* yang digunakan untuk membuat halaman *website* yang dinamis (Batubara, 2015).

Yang menjadikan PHP berbeda dengan HTML adalah proses dari PHP itu sendiri. HTML merupakan bahasa statis yang apabila kita ingin merubah konten/isinya maka yang harus dilakukan pertama kali nya adalah, membuka *file*-nya terlebih dahulu, kemudian menambahkan isi kedalam *file* tersebut. Beda halnya dengan PHP. Bagi anda yang pernah menggunakan CMS seperti *wordpress* atau *joomla* yang dibangun dengan PHP tentunya, ketika akan menambahkan konten kedalam *website*, anda tinggal masuk kedalam halaman admin, kemudian pilih *new* artikel untuk membuat halaman/*content* baru. Artinya hal ini, seorang *user* tidak berhubungan langsung dengan *script*nya. Sehingga seorang pemula sekalipun dapat menggunakan aplikasi seperti itu.

Keunggulan PHP :

1. Gratis

Yang membuat PHP begitu berkembang sangat pesat hingga jutaan domain menggunakan PHP, karena PHP itu gratis.

2. *Cross platform*

Artinya dapat di gunakan di berbagai sistem operasi, mulai dari *linux*, *windows*, *mac os* dan *os* yang lain.

3. Mendukung banyak *database*

PHP telah mendukung banyak *database*, banyak *developer web* menggunakan PHP Adabas D, dBase, Empress, FilePro (*read-only*) Hyperwave, IBM DB2, Informix, Ingres, InterBase, FrontBase mSQL, Direct MS-SQL, MySQL, ODBC, Oracle (OCI7 dan OCI8), Ovrimos, PostgreSQL SQLite, Solid, Sybase, Velocis, Unix dbm.

4. *On The Fly*

PHP sudah mendukung *on the fly*, artinya dengan PHP anda dapat membuat

*document* (Dwiartara, 2012).

#### **2.5.4 HTML (Hypertext Markup Language)**

(Rerung, 2018) mengungkapkan bahwa HTML adalah singkatan dari HyperText Markup Language. Disebut hypertext karena di dalam HTML sebuah text biasa dapat berfungsi lain, kita dapat membuatnya menjadi link yang dapat berpindah dari satu halaman ke halaman lainnya hanya dengan meng-klik text tersebut.

Pendapat (Jubilee Enterprise, 2014) HTML adalah script pemrograman yang mengatur bagaimana kita menyajikan informasi di dunia internet dan bagaimana informasi itu membawa kita melompat dari satu tempat ke tempat lainnya.

Jadi, HTML(*HyperText Markup Language*) merupakan bahasa pemrograman *web* yang memiliki sintak atau aturan tertentu dalam menuliskan *script* atau kode-kode, sehingga *browser* dapat menampilkan informasi dengan membaca kode-kode HTML(*HyperText Markup Language*) yang disajikan pada browser internet dengan kemampuan menyediakan link (hubungan ke dokumen lain), teks, grafik, dan suara.

#### **2.6. Flowchart**

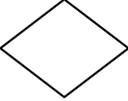
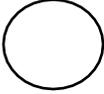
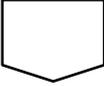
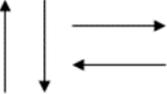
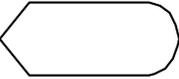
Menurut (Rusmawan, 2019), *Flowchart* (bagan alir) merupakan sebuah gambaran dalam bentuk diagram alir dari algoritma-algoritma dalam suatu program yang menyatakan arah alur program tersebut.

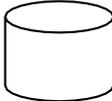
Sedangkan pendapat (Kadir, 2017) Diagram alir (*flowchart*) merupakan cara lain untuk menuangkan algoritma. Pendekatan yang dilakukan menggunakan gambar.

Berdasarkan pengertian diatas maka penulis menyimpulkan bahwa *flowchart* merupakan bagan alir yang berisikan alur dari program.

Berikut adalah tabel simbol-simbol *flowchart* :

Tabel 2.1 Simbol-Simbol *Flow Chart*

No	Simbol	Keterangan
1.		Simbol <i>Start</i> atau <i>End</i> yang mendefinisikan awal atau akhir dari sebuah <i>flowchart</i> .
2.		Simbol pemrosesan yang terjadi pada sebuah alur kerja
3.		Simbol <i>Input/Output</i> yang mendefinisikan masukan dan keluaran proses
4.		Simbol untuk memutuskan proses lanjutan dari kondisi tertentu
5.		Simbol konektor untuk menyambung proses pada lembar kerja yang sama
6.		Simbol konektor untuk menyambung proses pada lembar kerja yang berbeda
7.		Simbol untuk menghubungkan antar proses atau antar simbol.
8.		Simbol yang menyatakan piranti keluaran, seperti monitor, printer, dll

9.		Simbol yang mendefinisikan proses yang dilakukan secara manual
10.		Simbol masukan atau keluaran dari atau ke sebuah dokumen
11.		Simbol yang menyatakan bagian dari program (subprogram)
12.		Simbol masukan atau keluaran dari atau ke sebuah pita magnetic
13.		Simbol database atau basis data

(Sumber : Rusmawan, 2019)

## 2.7. Database

*Database* adalah sebuah *system* yang dibuat untuk mengorganisasi, menyimpan dan menarik data dengan mudah. *Database* terdiri dari kumpulan data yang terorganisir untuk satu atau lebih penggunaan, dalam bentuk *digital*. *Database digital* di-*manage* menggunakan *Database Management System* (DBMS), yang menyimpan isi *database*, mengizinkan pembuatan dan *maintenance data* dan pencarian dan akses yang lain. Beberapa *Database* yang ada saat ini adalah : MySQL, SQL Server, Ms.Access, Oracle, dan PostgreSQL (Sofwan, 2017).

Dalam konsep *database*, urutan atau hierarki *database* sangatlah penting (Solichin, 2010).

Istilah-istilah dalam basis data, yang didefinisikan sebagai berikut:

(Ladjamudin, 2015)

1. Bit

Bit merupakan bagian data yang terkecil; yang bisa diwakili dengan *numeric*, *symbol* khusus, gambar-gambar dan *alphabets*.

2. Byte

*Byte* adalah kumpulan dari pada bit-bit yang sejenis. Satu *byte* identik dengan satu karakter.

3. Field

*Field* merupakan sekumpulan *byte-byte* yang sejenis akan membentuk suatu *field*.

4. Atribut

Atribut merupakan relasi fungsional dari satu *object set* ke *object set* yang lain. Tiap tipe entitas memiliki sekumpulan atribut yang berkaitan dengannya.

5. Tuple/ Record

Dalam basis data istilah yang lebih tepat untuk menyatakan suatu baris data dalam suatu relasi adalah *tuple*, sebenarnya pengertian *tuple* bisa diidentikkan dengan *record*. *Tuple* terdiri dari kumpulan atribut-atribut dan atribut atribut tersebut saling berkaitan dalam menginformasikan tentang suatu entitas/relasi secara lengkap.

6. Entitas/ File

*File* merupakan kumpulan dari *record-record* yang sejenis dan mempunyai elemen yang sama, atribut yang sama, namun berbeda-beda data dan *value*-nya. *Database* terbentuk dari kumpulan *file*.

7. Domain

Domain adalah kumpulan dari nilai-nilai yang diperbolehkan untuk berada dalam satu atau lebih atribut. Setiap atribut dalam suatu basis data relasional didefinisikan sebagai suatu domain.

8. Kunci Elemen Data (*Key*)

*Key* adalah elemen record yang dipakai untuk menemukan *record* tersebut pada waktu akses, atau bisa juga digunakan untuk mengidentifikasi setiap *entity/record*/baris. Jenis-jenis *key*, yaitu :

a. Superkey

*Superkey* merupakan satu atau lebih atribut (kumpulan atribut) dari suatu table yang dapat digunakan untuk mengidentifikasi *entity/record* dari tabel tersebut secara unik. (tidak semua atribut dapat menjadi *superkey*).

b. *Candidate Key*

*Superkey* dengan jumlah atribut minimal, disebut *candidate key*.

c. *Primary Key*

Salah satu atribut dari *candidate key* dapat dipilih/ditentukan menjadi *primary key* dengan tiga kriteria sebagai berikut :

- a) *Key* tersebut lebih natural untuk digunakan sebagai acuan.
- b) *Key* tersebut lebih sederhana.
- c) *Key* tersebut terjamin keunikannya.

d. *Foreign Key*

*Foreign key* merupakan sembarangan atribut yang menunjuk kepada *primary key* pada table lain (Yanto, 2016).

Komponen penting dalam sistem basis data adalah :

1. Data

Merupakan informasi yang disimpan dalam suatu struktur tertentu yang terintegrasi.

2. *Hardware*

Merupakan perangkat keras berupa komputer dengan media penyimpanan yang digunakan untuk menyimpan data karena pada umumnya basis data memiliki ukuran yang besar.

3. Sistem Operasi

Program yang mengaktifkan dan memfungsikan sistem komputer, mengendalikan seluruh sumber daya dalam komputer serta melakukan operasi dasar dalam komputer meliputi *input*, proses dan *output*.

#### 4. Basis Data

Basis data sebagai inti dari sistem basis data. Basis data menyimpan data serta struktur sistem basis data baik untuk entitas maupun objek-objek secara detail.

#### 5. Database Management System

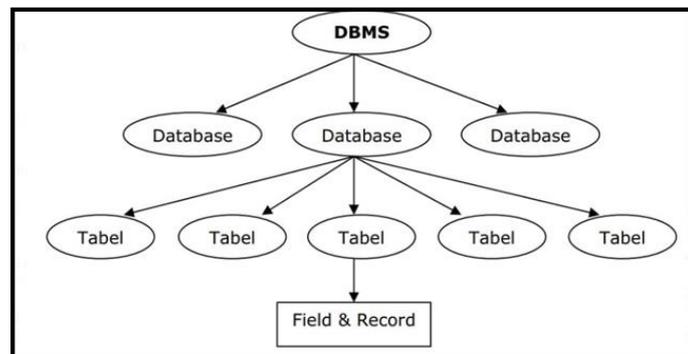
Merupakan perangkat lunak yang digunakan untuk melakukan pengelolaan basis data.

#### 6. User

Merupakan Pengguna yang menggunakan data yang tersimpan dan terkelola dapat berupa seseorang yang mengelola basis data yang disebut *database administrator (DBA)*, bisa juga disebut *end user*.

#### 7. Aplikasi Lainnya

Program yang dibuat untuk memberikan *interface* kepada *user* sehingga lebih mudah dan terkontrol dalam mengakses basis data.



**Gambar 2.4** Urutan atau Hierarki Database.

(Sumber : <https://queryitumudah.blogspot.com/2017/10/hirarki-database.html>)

### 2.7.1 MariaDB

MariaDB merupakan *Relational Database Management System (RDBMS)* yang cukup populer dikarenakan diciptakan oleh orang yang sama dengan yang menciptakan *MySQL*. Secara umum baik perintah, fungsi, maupun tampilan sangat mirip antara *MySQL* dengan MariaDB. MariaDB memiliki struktur *database* berupa relasi antara tabel yang satu dengan yang lain. Dalam artian bahwa sebelum menggunakan MariaDB, terlebih dahulu harus dilakukan inisialisasi terhadap setiap tabel dan setiap *field* didalam masing-masing tabel. Setelah

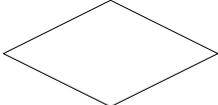
dilakukan inialisasi, baru kemudian dilakukan normalisasi tabel untuk memaksimalkan performa *database*. MariaDB hanya berfokus terhadap penyimpanan data, sementara untuk proses *input* data dan menampilkan data membutuhkan bantuan *custom-program* dengan berbagai bahasa pemrograman yang cukup banyak dan memberikan pilihan sesuai dengan kebutuhan sistem (Hendra dan Andriyani, 2020).

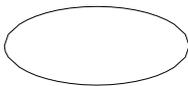
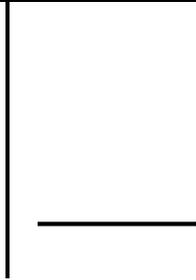
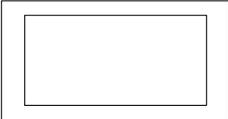
## 2.8. ERD (Entity Relationship Diagram)

(Fathansyah, 2018) mengatakan *Entity Relationship Diagram*(ERD) adalah komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang mempresentasikan seluruh fakta di dunia nyata.

ERD (*Entiy Relationship Diagram*) merupakan notasi grafis dalam pemodelan data konseptual yang mendeskripsikan hubungan antara penyimpanan. ERD digunakan untuk memodelkan struktur data dan hubungan antar data, karena hal ini relatif kompleks. Dengan ERD kita dapat menguji *model* dengan mengabaikan proses yang harus dilakukan.

**Tabel 2.2** Simbol dalam *Entity Relationship Diagram*

No	Simbol	Nama	Keterangan
1		Entitas	Jenis entitas dapat berupa suatu elemen lingkungan, sumber nya atau transaksi yang <i>field-field</i> -nya dipergunakan dalam aplikasi program
2		Hubungan atau Relasi	Menunjukkan nama relasi antar satu entitas dengan entitas lainnya

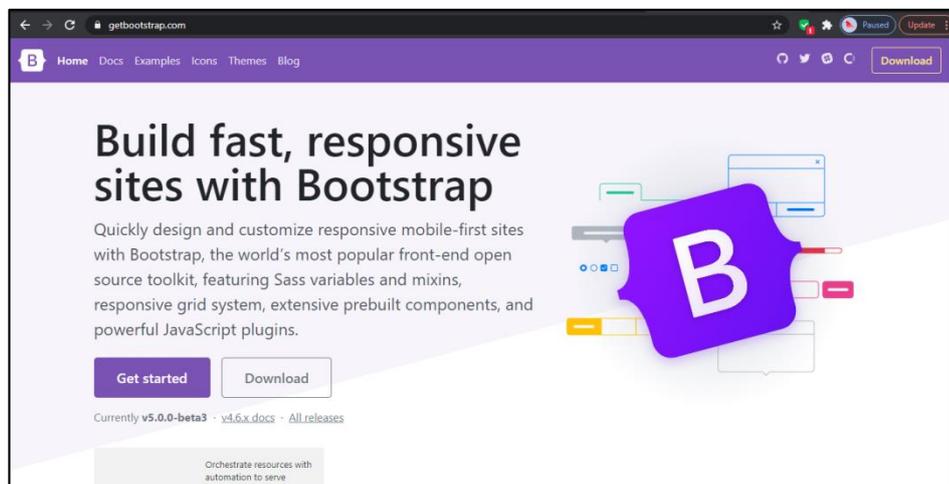
3		Atribut	Atribut adalah karakteristik dari sebuah entitas
4		Garis Relasi	Menunjukkan hubungan (keterkaitan) antar entitas
5		Entitas Lemah	Entitas yang kemunculannya tergantung dari entitas lain yang lebih kuat

## 2.9. Bootstrap

*Bootstrap* merupakan sebuah *framework* yang dapat menyelesaikan permasalahan dalam mendesain *web*. Slogan dari *framework* ini adalah “*Sleek, intuitive, and powerful front-end framework for faster and easier web development*”, yang berarti kita dapat mendesain sebuah *website* dengan lebih rapi, cepat dan mudah. Selain itu *Bootstrap* juga *responsive* terhadap banyak *platform*, artinya tampilan halaman *website* yang menggunakan *Bootstrap* ini akan tampak tetap rapi, baik versi *mobile* maupun *desktop* (Zakir, 2016).

*Bootstrap* merupakan salah satu jenis *framework* untuk CSS (*Cascading Style Sheet*) yang digunakan untuk perancangan situs *website*. Penggunaan *bootstrap* sangatlah membantu *programmer* dalam membangun tampilan sebuah *website*. Hal tersebut tertulis di buku *Bootstrap: Responsive Web Development* karangan Jake Spurlock. Menurut Jake Spurlock keunggulan dari penggunaan *Bootstrap* adalah dapat menyesuaikan dengan kebutuhan *website*, dimana dapat memilih fitur CSS dan *JavaScript* sesuai dengan kebutuhan. Contohnya CSS pada *Bootstrap* juga menyediakan *fitur form*, tombol, navigasi dan komponen-komponen lainnya dan *JavaScript* yang membantu pembuatan antarmuka lebih mudah dan stabil.

*Bootstrap* juga menyediakan banyak sekali *class-class* CSS dan *plugin* yang siap dipakai untuk membantu dalam membuat tampilan sebuah *website*. Karena sangat membantu, maka *Bootstrap* menjadi salah satu *front-end framework* yang paling banyak digunakan.



**Gambar 2.5** Tampilan *Bootstrap*

(Sumber : <https://blog.getbootstrap.com/2020/06/16/bootstrap-5-alpha/>)

```
<head>
<title>Bootstrap Example</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"></script>
</head>
```

**Gambar 2.6** Contoh Tampilan Kode *Bootstrap*

(Sumber : *Framework Bootstrap*)

### 2.9.1 CSS (Cascading Style Sheet)

(Andika, 2018) mengemukakan CSS (*Cascading Style Sheet*) adalah salah satu bahasa desain web (*style sheet language*) yang mengontrol format tampilan sebuah halaman web yang ditulis dengan menggunakan penanda (*markup language*). Biasanya CSS digunakan untuk mendesain sebuah halaman HTML dan XHTML, tetapi sekarang CSS bisa diaplikasikan untuk segala dokumen XML, termasuk SVG dan XUL bahkan ANDROID.

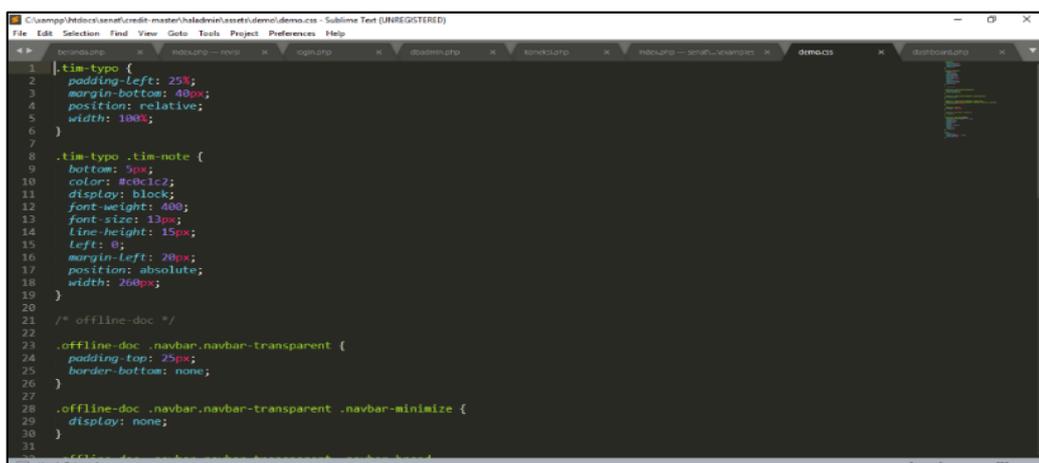
CSS dibuat untuk memisahkan konten utama dengan tampilan dokumen yang meliputi *layout*, warna dan *font*. Pemisahan ini dapat meningkatkan daya

akses konten pada web, menyediakan lebih banyak fleksibilitas dan kontrol dalam spesifikasi dari sebuah karakteristik dari sebuah tampilan, memungkinkan untuk membagi halaman untuk sebuah formatting dan mengurangi kerumitan dalam penulisan kode dan struktur dari konten, contohnya teknik *tableless* pada desain web.

CSS juga memungkinkan sebuah halaman untuk ditampilkan dalam berbagai *style* dengan menggunakan metode pembawaan yang berbeda pula, seperti *on-screen*, *in-print*, *by voice*, dan lain-lain. Sementara itu, pemilik konten web bisa menentukan *link* yang menghubungkan konten dengan *file* CSS.

Tujuan utama CSS diciptakan untuk membedakan konten dari dokumen dan dari tampilan dokumen, dengan itu pembuatan ataupun pemrograman ulang web akan lebih mudah dilakukan. Hal yang termasuk dalam desain web diantaranya adalah warna, ukura dan formatting. Dengan adanya CSS, konten dan desain web akan mudah dibedakan, jadi memungkinkan untuk melakukan pengulangan pada tampilan-tampilan tertentu dalam suatu web, sehingga akan memudahkan dalam membuat halaman web yang banyak, yang pada akhirnya dapat memangkas waktu pembuatan web.

Fungsi utama CSS adalah merancang, merubah, mendisain, membentuk halaman *website* (*blog* juga *website*) dan isi dari halaman *website* adalah *tag-tag html*, logikanya CSS itu dapat merubah *tag-tag html* (yang sederhana) sehingga menjadi lebih fungsional dan menarik.



```

1  .tim-typo {
2  padding-left: 25%;
3  margin-bottom: 40px;
4  position: relative;
5  width: 100%;
6  }
7
8  .tim-typo .tim-note {
9  bottom: 5px;
10 color: #ccc1c2;
11 display: block;
12 font-weight: 400;
13 font-size: 13px;
14 line-height: 15px;
15 left: 0;
16 margin-left: 20px;
17 position: absolute;
18 width: 260px;
19 }
20
21 /* offline-doc */
22
23 .offline-doc .navbar.navbar-transparent {
24 padding-top: 25px;
25 border-bottom: none;
26 }
27
28 .offline-doc .navbar.navbar-transparent .navbar-minimize {
29 display: none;
30 }
31

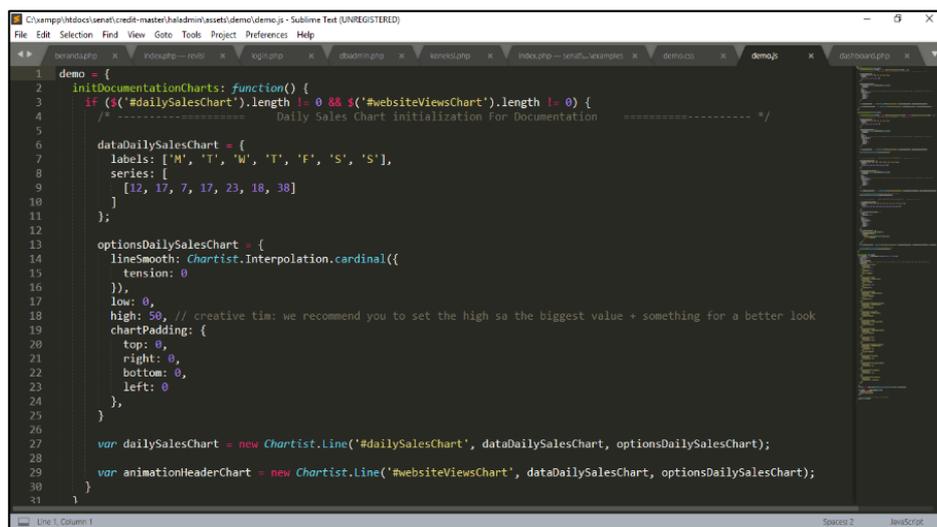
```

**Gambar 2.7** Contoh *Script* CSS  
(Sumber : Aplikasi *Sublime Text*)

## 2.10 JavaScript

(Ariata C, 2021) *JavaScript* adalah salah satu bahasa pemrograman yang paling banyak digunakan dalam kurun waktu dua puluh tahun ini. Bahkan, bahasa pemrograman ini merupakan salah satu yang paling utama bagi web *developer*, di antaranya:

- HTML : Memungkinkan Anda menambahkan konten ke halaman web.
- CSS : Menentukan *layout, style*, serta keselarasan halaman *website*.
- JavaScript* : Menyempurnakan tampilan dan sistem halaman web.



```

1 demo = {
2   initDocumentationCharts: function() {
3     if ($('#dailySalesChart').length != 0 && $('#websiteViewsChart').length != 0) {
4       /* ----- Daily Sales Chart Initialization For Documentation ----- */
5
6       dataDailySalesChart = {
7         labels: ['M', 'T', 'W', 'T', 'F', 'S', 'S'],
8         series: [
9           [12, 17, 7, 17, 23, 18, 38]
10        ]
11      };
12
13      optionsDailySalesChart = {
14        lineSmooth: Chartist.Interpolation.cardinal({
15          tension: 0
16        }),
17        low: 0,
18        high: 50, // creative tim: we recommend you to set the high sa the biggest value + something for a better look
19        chartPadding: {
20          top: 0,
21          right: 0,
22          bottom: 0,
23          left: 0
24        },
25      },
26
27      var dailySalesChart = new Chartist.Line('#dailySalesChart', dataDailySalesChart, optionsDailySalesChart);
28
29      var animationHeaderChart = new Chartist.Line('#websiteViewsChart', dataDailySalesChart, optionsDailySalesChart);
30    }
31  }

```

**Gambar 2.8** Contoh *Script JavaScript*  
(Sumber : Aplikasi *Sublime Text*)

### 2.10.1 jQuery

*jQuery* adalah *library JavaScript* yang populer. Bahasa pemrograman ini dibuat oleh John Resig, tepatnya pada tahun 2006, untuk memudahkan para *developer* dalam menggunakan dan menerapkan *JavaScript* di *website*. *jQuery* bukanlah bahasa pemrograman yang berdiri sendiri, melainkan bekerja sama dengan *JavaScript* (Ariata C, 2019).

Menurut (Sigit W, 2011) *jQuery* juga adalah kumpulan kode *JavaScript* siap pakai. Keunggulan menggunakan *jQuery* dibandingkan dengan *JavaScript* standar, yaitu menyederhanakan kode *JavaScript* dengan cara memanggil fungsi-fungsi yang disediakan oleh *jQuery*. *JavaScript* sendiri merupakan bahasa *Scripting* yang bekerja disisi *Client/Browser* sehingga *website* bisa lebih interaktif.

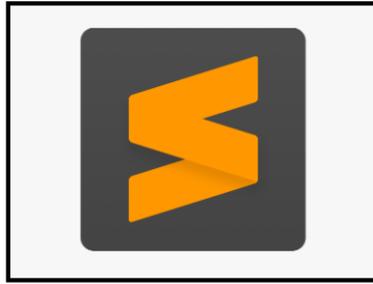
## 2.11 Text Editor

Menurut (Febrian, 2019) *Text editor* adalah sebuah *software* yang digunakan *programmer* untuk membuat sebuah program. Selain itu setiap *programmer* pasti membutuhkan kenyamanan dan kemudahan saat mengoding, oleh karena itu pilihan *text editor* terbaik dan juga tepat dapat menjadi kunci para *programmer* untuk membuat sebuah program yang bagus.

Salah satu *text editor* adalah *sublime text*. *Sublime Text*, menurut (Supono, 2016) *Sublime Text* merupakan perangkat lunak *text editor* yang digunakan untuk membuat atau meng-*edit* suatu aplikasi. *Sublime Text* mempunyai fitur *plugin* tambahan yang memudahkan *programmer*. Selain itu, *Sublime Text* terkesan elegan untuk sebuah *syntaxeditor*. Selain ringan, IDE ini memiliki kecepatan proses simpan dan buka file. Tidak heran kalau IDE ini paling banyak digunakan terutama dikalangan *programmer* berbasis *web*.

Pendapat (Haughee, 2013) *Sublime Text* adalah aplikasi *editor* untuk kode dan teks yang dapat berjalan di berbagai *platform operating system* dengan menggunakan teknologi *Phyton API*. Terciptanya aplikasi ini terinspirasi dari aplikasi Vim. Aplikasi ini sangatlah fleksibel dan *powerfull*. Fungsionalitas dari aplikasi ini dapat dikembangkan dengan menggunakan *sublime-packages*. *Sublime Text* bukanlah aplikasi *open source*, yang artinya aplikasi ini membutuhkan lisensi (*license*) yang harus dibeli. Akan tetapi beberapa fitur pengembangan fungsionalitas (*packages*) dari aplikasi ini merupakan hasil dari temuan dan mendapat dukungan penuh dari komunitas serta memiliki lisensi (*license*) aplikasi gratis.

Berdasarkan penjelasan diatas dapat disimpulkan *sublime text* merupakan aplikasi *editor* yang digunakan untuk membuat dan meng-*edit* suatu aplikasi dengan adanya kode dan teks yang dapat berbagai *platform operating system* dengan menggunakan teknologi *Phyton API*.



**Gambar 2.9** Logo *Sublime Text*  
(Sumber : Aplikasi *Sublime Text*)

```

30 <link href="../../assets/demo/demo.css" rel="stylesheet" />
31 </head>
32
33 <body class="">
34 <div class="wrapper">
35 <div class="sidebar" data-color="purple" data-background-color="white" data-image="../../assets/img/sidebar-1.jpg">
36 <!--
37 Tip 1: You can change the color of the sidebar using: data-color="purple | azure | green | orange | danger"
38
39 Tip 2: you can also add an image using data-image tag
40 -->
41 <div class="logo"><a href="../../index.html" class="simple-text logo-normal">
42 SENAT POLSRI
43 </a></div>
44 <div class="sidebar-wrapper">
45 <ul class="nav">
46 <li class="nav-item active">
47 <a class="nav-link" href="/dashboard.html">
48 <i class="material-icons">dashboard</i>
49 <p>Dashboard</p>
50 </a>
51 </li>
52 </ul>
53 <li class="nav-item">
54 <a class="nav-link" href="index.php?hal=pengumuman">
55 <i class="material-icons">upload</i>
56 <p>Upload Arsip</p>
57 </a>
58 </li>
59 <li class="nav-item">
60 <a class="nav-link" href="">

```

**Gambar 2.10** Tampilan *Sublime Text 3*  
(Sumber : Aplikasi *Sublime Text*)

## 2.12. DFD

DFD adalah sebuah teknik grafis yang menggambarkan aliran informasi dan tranformasi yang diaplikasikan pada saat data bergerak dari input menjadi output. Data Flow Diagram (DFD) adalah alat pembuatan model yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun komputerisasi. DFD ini sering disebut juga dengan nama bubble chart, bubble diagram, model proses, diagram alur kerja, atau model fungsi. Terdapat 2 bentuk DFD, yaitu DFD fisik (Physical Data Flow Diagram) dan DFD logika (Logical Data Flow Diagram). DFD fisik lebih menekankan pada bagaimana proses dari sistem diterapkan sedang DFD logika lebih menekankan proses-proses apa

yang terdapat di sistem. DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut akan disimpan (Herlambang dan Setyawati, 2015).