

BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

Teori Umum sebagai landasan dasar pengetahuan dalam perancangan *website* Sekolah Bodhi Dharma.

2.1.1. Definisi Sistem Informasi Manajemen

Sebelum sampai kepada sistem informasi manajemen, akan diuraikan terlebih dahulu pengertian sistem. Sistem merupakan komponen-komponen yang saling berinteraksi atau berhubungan secara teratur satu dengan lainnya atau saling tergantung membentuk satu kesatuan yang utuh untuk mencapai tujuan tertentu. Ciri-ciri pada sebuah sistem adalah: digunakan sebagai alat untuk mencapai tujuan, merupakan kesatuan usaha, adanya unsur fungsional (*input, process, output* dan *feedback*), saling berhubungan, terstruktur dan berjenjang (Rustiyanto, 2011).

Data dapat didefinisikan sebagai representasi fakta yang bisa diproses oleh manusia maupun oleh mesin elektronik. Informasi adalah data terstruktur atau data yang terorganisir, yang memiliki nilai bagi penerimanya. Informasi merupakan data yang diproses yang menjadi dasar pengambilan keputusan dan tindakan. Ketika informasi dimasukkan ke dalam komputer, biasanya disebut sebagai data, setelah diproses (dicetak), hasil dari data tersebut kembali menjadi informasi.

Agar keputusan menjadi bermakna, data yang diproses harus memenuhi syarat:

1. Tepat waktu, informasi harus tersedia saat diperlukan
2. Akurasi, informasi harus akurat
3. Kelengkapan, informasi harus lengkap.

Pemrosesan data adalah penataan ulang data oleh orang atau mesin untuk meningkatkan kegunaannya dan menambah nilai untuk tujuan tertentu. Pemrosesan data terdiri dari langkah-langkah yaitu *Input*, *Process*, dan *Output* (IPO). Tiga langkah ini merupakan siklus pemrosesan data.

1. *Input*, data input berupa masukan kedalam sistem komputer.
2. *Process*, input data diubah untuk menghasilkan data dalam bentuk yang lebih bermanfaat.
3. *Output*, hasil dari pemrosesan data.

(Griffin, 2012, p.142) mendefinisikan manajemen sebagai sebuah proses perencanaan, pengorganisasian, pengkoordinasian, dan pengontrolan sumber daya untuk mencapai sasaran secara efektif dan efisien.

Dapat disimpulkan bahwa sistem informasi manajemen adalah penerapan sistem teknologi informasi pada organisasi bisnis dengan perencanaan yang terorganisir untuk mencapai tujuan secara efektif dan efisien. Sistem informasi manajemen digunakan untuk melakukan fungsi pengolahan data, menerima masukan (*input*) berupa data-data, kemudian mengolahnya (*processing*), dan menghasilkan keluaran (*output*) berupa informasi untuk pengambilan keputusan yang berguna pada saat itu maupun di masa mendatang. (Hartono, 2013)

2.1.2. Definisi Website

Website atau disingkat *web* merupakan beberapa halaman berisi informasi atau konten multimedia berupa data gambar, *audio*, *video* maupun animasi lainnya yang diakses melalui internet. (Rohi, 2016). Situs *web* dapat berfungsi untuk berbagai kebutuhan, situs *web* dapat berupa situs *web* pribadi, situs *web* perusahaan untuk perusahaan, situs *web* pemerintah, situs *web* organisasi, situs *web* sekolah, dll. Situs *web* biasanya didedikasikan untuk tujuan tertentu, mulai dari hiburan dan jejaring sosial hingga menyediakan berita dan pendidikan.

2.1.3. Definisi Database

Basis data adalah kumpulan data-data yang terkait dan tersimpan dalam suatu penyimpanan yang terorganisir sehingga dapat dengan mudah dimanipulasi (dikelola, diakses dan diperbarui) dengan tujuan untuk

1. Mudah menampilkan data kembali
2. Dapat digunakan oleh lebih dari satu program secara optimal
3. Mempermudah proses pengolahan data seperti CRUD (*Create, Read, Update, Delete*) atau penambahan, pengambilan dan modifikasi data dengan cara yang terkontrol dalam *database*. (Sutanta, 2011)

2.1.4. Definisi *Software Development Life Cycle* (SDLC)

SDLC (siklus hidup pengembangan sistem) merupakan serangkaian kegiatan atau suatu metode yang direncanakan untuk mengembangkan atau mengubah produk perangkat lunak, merancang, dan menguji perangkat lunak berkualitas tinggi dengan tujuan untuk menghasilkan perangkat lunak yang memenuhi kebutuhan pelanggan dan mencapai penyelesaian dalam waktu dan perkiraan biaya.

Tahapan SDLC

1. Perencanaan Sistem (*Sistem planning*), aspek studi kelayakan

- Aktivitas dan pembentukan tim pengembang
- Mendefinisikan tujuan dan ruang lingkup pengembangan
- Mengidentifikasi masalah yang bisa diselesaikan melalui pengembangan sistem.
- Menentukan dan evaluasi strategi dalam pengembangan sistem
- Penentuan prioritas dan pemilihan aplikasi.

2. Analisis Sistem (*Sistem Analysis*), analisis sistem meliputi

- Menemukan kasus yang bisa ditangani sistem dengan melakukan studi literatur
- Pada tim pengembang tentang kasus yang paling tepat dimodelkan dengan sistem.
- Pada kasus yang mungkin diterapkan dengan mengklasifikasikan masalah, peluang dan solusi.
- Membuat batasan sistem dengan analisis kebutuhan pada sistem.
- Kebutuhan sistem didefinisikan.

3. Perancangan Sistem (*Sistem Design*), fitur dan operasi pada sistem dideskripsikan secara detail.
 - Interaksi objek dan fungsi sistem dianalisa.
 - Menganalisa dan membuat skema *database*.
 - *User Interface* dirancang.
4. Implementasi Sistem (*Sistem Implementation*), mengimplementasikan rancangan dari tahap-tahap sebelumnya.
 - Pembuatan *database* sesuai skema rancangan.
 - Pembuatan aplikasi sesuai skema rancangan.
 - Pengujian dan perbaikan aplikasi.
5. Pengujian Sistem (*Sistem Testing*), menguji sistem yang sedang berjalan apakah telah sesuai dengan keinginan pengguna serta untuk mengetahui apakah terjadi kesalahan pada sistem sehingga bisa dilakukan proses meminimalisir dan menghilangkan kesalahan agar sistem berfungsi dengan maksimal.
6. Pemeliharaan Sistem (*Sistem Maintenance*), menjaga sistem tetap mampu beroperasi secara benar melalui kemampuan sistem dalam mengadapatasikan sesuai kebutuhan yang dilakukan oleh admin.

Salah satu model SDLC yang paling sering digunakan adalah Model *Waterfall* dimana merupakan proses pengembangan perangkat lunak tradisional yang digunakan dalam proyek perangkat lunak dengan model sekuensial, sering juga disebut *classic life cycle*. (Fahrurrozi, 2016).

2.2. Teori Khusus

Teori Khusus adalah teori-teori yang digunakan penulis dan berhubungan dengan topik yang dibahas dalam skripsi ini.

2.2.1 Pengertian HTML

HTML merupakan format untuk membuat dokumen yang berjalan di halaman *web* atau yang dikenal dengan *web page*, yang mendeskripsikan struktur dari sebuah halaman *web* menggunakan *markup* yang direpresentasikan dengan *tag*. Komponen dari HTML terdiri dari

1. **Tag**, merupakan pengapit suatu elemen yang diawali dengan tanda < dan diakhiri dengan tanda > Contohnya <head> merupakan *tag* pembuka *head* dan </head> *tag* penutup *head*.
2. **Elemen**, merupakan nama penanda pada dokumen HTML yang diapit oleh *tag* dan memiliki fungsi dan tujuan tertentu. Contohnya <title></title> merupakan elemen yang menandakan sebagai judul.
3. **Atribut**, merupakan suatu elemen pada HTML sebagai properti dengan tujuan tertentu. Contohnya

```
<body id="mybody" class="mycontent">
```

```
<p>
```

pada elemen *body* dengan atribut *id* dan *class*.

```
</p>
```

```
</body>
```

2.2.2 Pengertian PHP

PHP merupakan bahasa pemrograman yang bersifat *interpreter* dimana proses penerjemahan dimengerti oleh komputer secara langsung baris kode sumber menjadi kode mesin saat baris dijalankan. PHP adalah bahasa dengan hak cipta terbuka (*Open Source*) dimana pengguna boleh mengembangkan kode-kode fungsi PHP sesuai dengan kebutuhan dan PHP juga disebut sebagai *Server Side Programming* (semua proses dijalankan pada server.) (Sibero, 2013)

2.2.3 Pengertian CSS

CSS dikembangkan untuk menata gaya pengaturan halaman web. CSS terdiri dari *selector*, *property* dan *value*.

1. *Selector* atau *pattern* merupakan identitas pendefinisian suatu *style* dengan suatu bentuk pola.

Tabel 2.1 Tabel penjelasan *selector*

<i>Selector/Pattern</i>	Contoh Penulisan	Keterangan	Versi CSS
:hover	a:hover	Berlaku bagi elemen <a> saat mouse diatas elemen.	1
[atribut=nilai]	[align="left"]	Berlaku bagi semua elemen yang mendefinisikan atribut [align]	2
:enabled	input:enabled	Berlaku bagi semua elemen <i>input</i> yang statusnya <i>enable</i>	3

2. *Property* atau properti pada CSS merupakan jenis *style* pada *selector* yang ditentukan, misalnya property *background*, *border*, *color*, *font-size*, *margin*, *padding* dan lainnya yang dapat digunakan pada untuk selector yang diinginkan.
3. *Value* merupakan nilai yang diberikan pada suatu *property*. Contoh *font-size:20px*; menandakan ukuran *font* adalah 20px.

a {background-color: blue;}

a → sebagai *selector*

background-color → sebagai **property**

blue → sebagai **value**

2.2.4 Pengertian MySQL

MySQL merupakan aplikasi sistem dengan fungsi untuk pengolahan data seperti CRUD (*Create, Read, Update, Delete*) dalam *database* yang terdiri dari

1. *Data Definition Language* (DDL) merupakan bahasa yang digunakan untuk mendefinisikan *database* dan *table*. DDL secara umum digunakan untuk mendefinisikan suatu *record* yang terdiri dari *alter*, *create*, *drop* dan *rename*.
2. *Data Manipulation Language* (DML) merupakan tata bahasa untuk memanipulasi data yang terdapat dalam MySQL, digunakan untuk manipulasi data pada *table database* yang terdiri dari *select*, *insert*, *update*, *delete*.(Sibero, 2013).

2.2.5 Pengertian phpMyAdmin

phpMyAdmin adalah sebuah aplikasi (*open source*) yang ditulis menggunakan PHP dalam penanganan administrasi *database* MySQL melalui jaringan lokal maupun internet. phpMyAdmin mendukung berbagai operasi manajemen MySQL, diantaranya (mengelola basis data, tabel-tabel, relasi (*relations*), indeks, pengguna (*users*), dan lain-lain) dengan lebih mudah tanpa perlu mengetikkan SQL secara manual. (Akhmad, 2011).

2.2.6 Pengertian XAMPP

XAMPP merupakan alat yang menyediakan paket seperti Apache, PHP dan MySQL dalam sebuah konfigurasi sehingga ketika XAMPP diinstalasi maka secara otomatis menjadi satu kesatuan tanpa harus menkonfigurasi *web server* secara manual. XAMPP dari beberapa bagian.

1. **X**, disebut sebagai sistem operasi karena bisa dijalankan di 4 operasi sistem yang berbeda seperti *Windows, Max OS, Linux* dan *Solaris*
2. **A** mewakili Apache, merupakan *web server* yang bersifat *open source* dengan tujuan menghasilkan halaman web sesuai kode program yang dituliskan pengguna.
3. **M** mewakili MySQL, yaitu bahasa yang terstruktur untuk pengolahan database.
4. **P** mewakili PHP, yaitu bahasa pemrograman *server side programming* dan bersifat *open source*.
5. **P** mewakili Perl (*Practical Extraction and Report Language*)

2.2.7 Pengertian JavaScript

JavaScript adalah bahasa skrip (*scripting language*), yaitu kumpulan instruksi perintah yang digunakan untuk mengendalikan beberapa bagian dari sistem operasi. Sebagai bahasa skrip yang berjalan pada *web browser* atau sisi klien. Sebagai bahasa *script* yang berjalan pada sisi klien, *Javascript* tidak memiliki perintah untuk menjalankan perintah pada server (*server side*). Supaya *JavaScript* dapat berinteraksi pada *server* para pengembang kemudian menambahkan satu mekanisme yang disebut dengan *AJAX* (*Asynchronous Javascript dan XML*) dengan prinsip kerja menjalankan alamat pada *server* dan menerima data yang dikembalikan oleh *server*. (Sibero, 2013)

2.2.8 Pengertian jQuery

JavaScript framework adalah kumpulan pustaka fungsi yang dibentuk menggunakan *JavaScript*, dengan menggunakan *framework JavaScript* dapat menggunakan pustakanya yang memudahkan proses pembuatan program menjadi lebih cepat dan singkat. Salah satu *framework JavaScript* yang populer digunakan yaitu jQuery. jQuery adalah kumpulan fungsi-fungsi *JavaScript* yang siap pakai sehingga memudahkan penulisan kode *JavaScript*, dan merupakan salah satu *framework JavaScript* terbaik saat ini. jQuery dikembangkan oleh John Resig pada tahun 2006 di BarCamp NYC, jQuery memiliki ciri khas pada penggunaan perintahnya, *prefix* untuk jQuery dengan tanda \$ kemudian dilanjutkan dengan fungsi atau perintah. (Sibero, 2013.)

2.2.9 Pengertian *Bootstrap*

Bootstrap adalah sebuah *framework* CSS yang dapat digunakan untuk mempermudah membangun tampilan web, dibuat khusus untuk bagian pengembangan *front-end website*. *Bootstrap* merupakan salah satu *framework* HTML, CSS, dan JavaScript yang paling populer di kalangan *web developer*.

Bootstrap telah menyediakan kumpulan komponen *class interface* dasar yang dirancang sedemikian rupa untuk menciptakan tampilan yang menarik bersih dan ringan, *bootstrap* juga memiliki fitur *grid* yang berfungsi untuk mengatur *layout* pada halaman *website* yang dapat digunakan dengan mudah dan cepat. Komponen pada *bootstrap* terdiri dari *buttons*, *dropdown*, *modal*, *tooltip*, *pagination*, dll.

1. *Buttons* atau tombol pada *Bootstrap* yang telah disediakan mencakup beberapa gaya tombol yang telah ditentukan, masing-masing melayani tujuan yang berbeda-beda terdiri dari
 - a. *Primary*, tombol utama dengan bawaan warna biru
 - b. *Secondary*, tombol sekunder dengan bawaan warna abu-abu
 - c. *Success*, tombol yang menandakan sukses dengan bawaan warna hijau
 - d. *Danger*, tombol yang menandakan bahaya atau *error* dengan bawaan warna merah
 - e. *Warning*, tombol yang menandakan peringatan dengan bawaan warna kuning
 - f. *Info*, tombol yang menandakan informasi dengan bawaan warna biru muda
 - g. *Light*, tombol dengan bawaan warna terang
 - h. *Dark*, tombol dengan bawaan warna gelap/hitam.

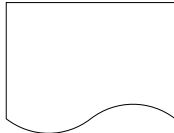
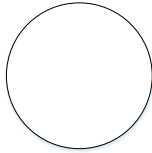
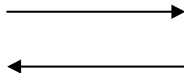
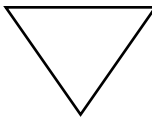
Tombol dari *Bootstrap* dapat digunakan sesuai dengan kebutuhan.

2. *Dropdown* adalah menu pada *Bootstrap* yang terdiri dari *list* (beberapa nilai) dalam suatu daftar, yang memungkinkan pengguna memilih salah satu nilai dari daftar.
3. *Modal* adalah dialog yang muncul dengan fungsional yang fleksibel dan tampilan yang disederhanakan, biasanya digunakan sebagai peringatan (*alert*) konfirmasi, maupun memunculkan data dalam dialog tanpa harus menuju halaman lain.
4. *Tooltips* merupakan tampilan informasi berupa teks maupun gambar yang tampil saat *cursor* diarahkan, biasanya digunakan untuk memberikan sekilas informasi singkat.
5. *Pagination* merupakan fitur yang digunakan untuk membatasi tampilan data agar tidak terlalu panjang dan lebih rapi dan membatasi menjadi beberapa halaman.

2.2.10 Pengertian Aliran Sistem Informasi

Aliran Sistem Informasi (ASI) merupakan alat bantu yang digunakan dalam merancang suatu sistem yang berguna untuk menunjukkan urutan prosedur yang ada pada sebuah sistem. Diagram alir sistem ditunjukkan menggunakan simbol-simbol seperti tabel berikut.

Tabel 2.2 Simbol Aliran Sistem Informasi

Simbol	Keterangan
	Simbol Proses, melambangkan proses pada komputer
	Simbol Dokumen, melambangkan dokumen input dan output baik manual maupun komputer.
	Simbol Penghubung, melambangkan penghubung dalam satu halaman.
	Garis Alur, melambangkan arus dari proses.
	Proses Manual, melambangkan proses yang dilakukan secara manual
	<i>Offline Storage</i> , Digunakan untuk menyimpan data secara manual dan sementara.
	<i>Input output</i> menggunakan disket.
	<i>Database</i> , melabangkan hubungan ke <i>database</i> .

2.2.11 Pengertian *Unified Modeling Language* (UML)

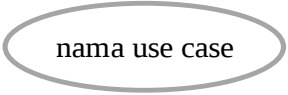
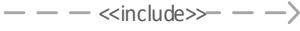
Unified Modeling Language (UML) merupakan bahasa untuk menspesifikasikan, mendokumentasikan rancangan sistem berorientasi pada objek dengan menentukan, memvisualisasikan, membangun dan mendokumentasikan suatu sistem informasi.

Berikut adalah beberapa diagram yang paling sering digunakan dalam pembangunan aplikasi berorientasi objek, yaitu

2.2.11.1. *Use Case Diagram*

Use Case Diagram adalah diagram yang digunakan untuk memodelkan semua proses bisnis berdasarkan perspektif pengguna sistem, yang memperlihatkan hubungan yang terjadi antara *actors* dan *use case* dalam sistem. *Use Case* merupakan sebuah kegiatan atau juga interaksi yang saling berkesinambungan antara actor dan juga sistem, digunakan untuk mengembangkan *software* atau sistem informasi untuk memperoleh kebutuhan fungsional dari sistem yang ada.

Tabel 2.3 Simbol *Use Case*

Simbol	Deskripsi
Use Case 	Fungsional sistem yang akan dibuat, agar pengguna sistem mengerti sistem yang akan dibangun.
Aktor / <i>actor</i> 	Menggambarkan orang atau aktor yang berinteraksi dengan sistem.
Asosiasi / <i>association</i> 	Menghubungkan <i>link</i> antar elemen.
Ekstensi / <i>extend</i> 	Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang dapat berdiri sendiri meskipun tanpa <i>use case</i> tambahan tersebut.
Generalisasi / <i>generalization</i> 	Penghubung antara <i>actor</i> dengan <i>use case</i>
Menggunakan / <i>include</i> 	Relasi <i>use case</i> tambahan harus terpenuhi agar suatu kejadian dapat terjadi.

2.2.11.2. Activity Diagram

Activity Diagram menggambarkan aktivitas untuk menunjukkan aliran pesan dari satu aktivitas ke aktivitas lainnya. Tujuan dari diagram aktivitas adalah sebagai gambaran aliran aktivitas suatu sistem dan menjelaskan urutan dari satu aktivitas ke aktivitas lainnya.

Tabel 2.4 Simbol *Activity Diagram*

Simbol	Deskripsi
<i>Initial Activity</i> 	Sebagai awal dari aktivitas dari sistem yang berjalan.
Aktivitas / <i>action</i> 	Memperlihatkan aktivitas yang saling berinteraksi satu sama lain.
Percabangan / <i>decision</i> 	Digunakan untuk menggambarkan suatu keputusan yang harus diambil dalam kondisi tertentu.
Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Status akhir / <i>Final node</i> 	Status akhir yang dilakukan sistem, yang menandakan aktivitas berakhir.

2.2.11.3. *Class Diagram*

Class Diagram adalah model yang menggambarkan struktur dan deskripsi *class* beserta hubungannya antara *class* untuk membangun sistem, *class* dibagi menjadi 3 bagian utama yaitu, *attribute*, *operation*, dan *name*.

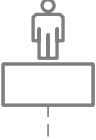
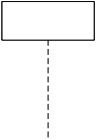
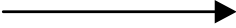
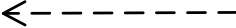
Tabel 2.5 Simbol *Class Diagram*

Simbol	Deskripsi
Asosiasi / <i>Assosiation</i> 	Relasi antar objek yang menghubungkan objek satu dengan objek lainnya.
Asosiasi berarah / <i>Directed Assosiation</i> 	Asosiasi dengan makna kelas yang satu digunakan oleh kelas yang lain.
Kebergantungan / <i>Dependency</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi / <i>Aggregation</i> 	Hubungan yang menyatakan bahwa suatu kelas menjadi atribut bagi kelas lain.
Komposisi / <i>Composition</i> 	Bentuk khusus dari agregasi dimana kelas yang menjadi bagian diciptakan setelah kelas menjadi <i>whole</i> dibuat.
Realisasi / <i>Realization</i> 	Hubungan antar kelas dimana sebuah kelas memiliki keharusan untuk mengikuti aturan yang ditetapkan oleh kelas lainnya.

2.2.11.4. Sequence Diagram

Sequence Diagram merupakan diagram yang menggambarkan interaksi antar objek dalam perintah yang berurut dalam sebuah sistem.

Tabel 2.6 Simbol *Sequence Diagram*

Simbol	Deskripsi
Aktor / Actor 	Menggambarkan orang yang sedang berinteraksi dengan sistem
Garis hidup / lifeline 	Menyatakan kehidupan suatu objek.
Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi pesan.
Pesan tipe create 	Menyatakan suatu objek membuat objek yang lain, arah panah menghadap pada objek yang akan dibuat.
Pesan Return Message 	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi menghasilkan sesuatu kembalian ke objek.

2.3. Penelitian Terdahulu

Penelitian terdahulu ini menjadi salah satu acuan penulis dalam melakukan penelitian sehingga penulis dapat melakukan penelitian dan dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Dari penelitian terdahulu penulis mengangkat beberapa penelitian sebagai referensi dalam memperkaya bahan kajian pada penelitian penulis. Berikut merupakan penelitian terdahulu berupa beberapa jurnal terkait dengan penelitian yang dilakukan penulis.

No	Nama Peneliti	Judul	ISSN	Kesimpulan
1.	Isa Budi Kahano	Pembuatan Website Sekolah Dasar Negeri Sragen Empat Belas	2302-5700	<i>Website</i> merupakan suatu media informasi yang menawarkan berbagai kemudahan dalam menyajikan informasi. Kecepatan dan kenyamanan merupakan nilai positif adanya internet. Sekolah Dasar Negeri Sragen Empat Belas adalah salah satu instansi pemerintah dalam dunia pendidikan penyampaian informasi melalui website kepada masyarakat maupun warga sekolah (Kahono, 2013)
2.	Sugiyanto	Pembuatan Website Profil	2302-1136	Sekolah Menengah Kejuruan Pembangunan

		Sekolah Menengah Kejuruan Pembangunan Nusantara Gabus Grobogan		Nusantara Gabus Grobogan (SMK PEMNUS GABUS) merupakan salah satu sekolah yang didirikan pada tahun 2008, Dalam menyampaikan informasi kepada siswa, staff, guru, dan masyarakat umum untuk mengetahui informasi mengenai SMK PEMNUS GABUS masih bersifat konvensional, yaitu masih menggunakan spanduk/baliho, brosur, papan pengumuman, dengan cara seperti ini sehingga penyampaian informasi terbatas, tidak efektif dalam segi biaya, waktu dan tenaga. Untuk itu perlu dibuatnya <i>website</i> sebagai sarana penyampaian informasi kepada warga sekolah maupun masyarakat luas mengenai SMK PEMNUS GABUS. (Sugiyanto, 2013)
3.	Abner B. Tupas	<i>Design and Development of LPU-B</i>	2350-7756	Situs web adalah cara terbaik untuk mempublikasikan dan menyebarluaskan informasi

		<i>High School Website</i>		kepada jumlah individu terbanyak di dunia. Internet adalah teknologi untuk organisasi yang menyediakan sarana untuk berkomunikasi dengan publik. Ia juga memiliki satu-satunya karakteristik tanpa batas geografis atau batas waktu, semua orang Interaksi melalui jejaring pertemuan satu ke banyak atau banyak ke banyak cocok untuk melakukan bisnis dan menyediakan atau meminta layanan yang mereka inginkan. (Tupas, 2015)
4.	Madiha Shah	<i>Impact of management information systems (MIS) on school administration</i>	2799 – 2804	Sistem informasi manajemen (SIM) digunakan oleh sekolah untuk mendukung berbagai kegiatan administrasi Termasuk pemantauan kehadiran, catatan penilaian, pelaporan, manajemen keuangan, dan sumber daya dan staf SIM memberikan informasi yang dibutuhkan manajer

				untuk mengelola organisasi secara efisien dan Sistem yang efektif Sistem ini berbeda dari sistem informasi lain yang dirancang untuk digunakan Menganalisis dan memfasilitasi kegiatan strategis dan operasional dalam organisasi. (Shah, 2014)
--	--	--	--	--