

BAB II

TINJAUAN PUSATAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Jati Sasongko dan Dwi Agus Diartono (2009), melakukan penelitian yang berjudul Rancang Bangun Sistem Manajemen Surat. Sistem administrasi manajemen surat yang ada masih dilakukan secara manual, dengan banyaknya surat yang dibuat dan diterima, hal ini menyebabkan proses menjadi tidak efisien dalam hal waktu dan tenaga pada pencarian data. Jati Sasongko dan Dwi Agus Diartono mengatasi hal tersebut dengan merancang sistem administrasi manajemen surat yang lebih terstruktur.

Budi Siswanto dan Moh. Sulhan (2014), melakukan penelitian yang berjudul Sistem Pengolahan Data Surat Masuk Dan Keluar Dengan Menggunakan Metode *Index Field* Berbasis *Web* Pada Smk Islam Batu. Penelitian bertujuan untuk memperbaiki sistem penginputan surat masuk dan keluar yang dilakukan berulang-ulang pada saat diperlukan sehingga mengakibatkan *Inkonsistensi* data terhadap arsip surat masuk dan keluar, penumpukan catatan atau rekapan data, surat yang tercecer sehingga ketika data surat keluar dan surat masuk diperlukan maka harus dicari kembali, proses pencarian tidak bisa dilakukan dengan cepat dan tepat, dan secara fisik media penyimpanan arsip surat masuk dan surat keluar memerlukan media yang luas dan besar akibatnya terjadi penumpukan arsip. Budi dan Sulhan mengatasi hal tersebut dengan membuat *web* sistem pengolahan data surat masuk dan keluar dengan menggunakan metode *index field*.

Abu Yanto (2015), pada jurnalnya yang berjudul Sistem Informasi Arsip Elektronik Berbasis *Web* Pada Kantor Cabang Jember, merancang sebuah *web* sistem informasi arsip elektronik yang mampu menyimpan data surat kabar yang didapat dari wartawan agar diarsipkan sebagai bukti adanya berita dari percetakan. Dengan sistem tersebut diharapkan dapat mempermudah tim

redaksi untuk menyimpan data berita yang diberikan oleh wartawan pada kantor tersebut agar jika sewaktu-waktu dibutuhkan dapat diambil kembali atau digunakan sebagai laporan untuk menghitung gaji karyawan.

Yoshepin Dwi Tatalia dan Nina Setiyawati (2016), melakukan penelitian yang berjudul Perancangan Sistem Informasi Arsip Salatiga Berbasis Web dengan *CodeIgniter* dan *Materialize*. Sistem penyimpanan arsip Salatiga masih terpisah dan pendeskripsian arsip tersebut disimpan dalam buku atau bentuk dokumen *word* atau *excel* menyebabkan proses pengelolaan dan penemuan kembali arsip mengalami kesulitan serta informasi yang dihasilkan kurang cepat dan tepat. Yosephin dan Nina mengatasi hal tersebut dengan merancang sistem informasi kearsipan berbasis web untuk arsip salatiga sehingga mempermudah dalam penyajian arsip, pengelolaan arsip, pengelompokan arsip dan pencetakan laporan, serta mempercepat proses pencarian dan penemuan kembali arsip ketika dibutuhkan.

Berdasarkan beberapa analisis yang telah dilakukan oleh peneliti sebelumnya, aplikasi yang dibuat hanya berupa pencatatan data, pencarian data, dan pembuatan laporan saja. Dari kekurangan tersebut maka pada skripsi Rancang Bangun Sistem Manajemen Arsip Undangan Berbasis Web ini penulis tidak hanya berfokus pada proses pencatatan, pencarian, dan pembuatan laporan pada data undangan saja akan tetapi penulis juga menambahkan fitur-fitur sebagai berikut:

1. Fitur kirim undangan melalui *email* yang dilakukan oleh *admin*.
2. *User* dapat menerima jadwal undangan melalui *email* dan dapat melihat daftar undangan masing-masing *user* pada halaman *website*.
3. Fitur kalender untuk menandai tanggal jadwal undangan masing-masing *user*.
4. Fitur *download* data undangan yang berupa dokumen dengan *format pdf* oleh *user* pada halaman *website*.

2.2 Landasan Teori

2.2.1 Pengertian Undangan

Menurut Susan Azzahra (2016), undangan merupakan alat yang digunakan sebagai media untuk menyampaikan informasi kepada orang-orang tertentu tentang sebuah kegiatan atau acara. Undangan berasal dari kata dasar undang dan menggunakan imbuhan an. Undang memiliki arti panggil, mengundang memiliki arti memanggil atau mempersilakan datang. Undangan merupakan kata benda yang artinya adalah subjek yang dipanggil atau dipersilakan untuk hadir dalam sebuah acara atau kegiatan yang sudah ditetapkan pada undangan.

Menurut Dot Productions (2011), undangan dibagi menjadi 3 jenis jika dibedakan dari kepentingan acaranya antara lain:

1. Undangan Formal yaitu undangan yang berasal dari sebuah Instansi, organisasi, atau kedinasan. Biasanya digunakan untuk kepentingan-kepentingan kedinasan ataupun bersifat struktural.
2. Undangan Semi Formal yaitu undangan yang berasal dari perorangan untuk kepentingan perorangan. Sebagai contoh undangan acara adat yang diberikan oleh seseorang kepada pemangku adat untuk melakukan acara yang bersifat sakral dan menjadi budaya di dalam masyarakat setempat.
3. Undangan Informal yaitu undangan yang berasal perorangan untuk kepentingan perorangan atau golongan. Sebagai contoh undangan ulang tahun.

2.2.2 Pengertian Arsip

Menurut Achmad Maulidi (2016), arsip merupakan sumber informasi atau catatan kegiatan dengan berbagai macam bentuk seperti berupa surat, akta, warkat, buku, dan sebagainya yang dibuat oleh suatu lembaga, organisasi atau perseorangan yang dapat dijadikan bukti dari suatu tindakan dan keputusan dalam pelaksanaan kegiatan. Seiring dengan teknologi yang semakin maju, arsip kini dapat berbentuk audio, video dan digital. Dengan perkembangan jumlah arsip yang semakin banyak dibuat dan diterima oleh lembaga,

organisasi, maupun perseorangan maka diperlukan manajemen pengelolaan arsip yang lebih dikenal dengan sistem kearsipan melalui beberapa pekerjaan atau kegiatan untuk mengelola arsip yang ada.

Sistem kearsipan adalah kegiatan yang dilakukan dalam mengelola sumber informasi atau catatan rekaman kegiatan yang memiliki nilai kegunaan secara teratur dan terencana baik itu arsip yang dibuat maupun diterima, yang memiliki tujuan agar mudah dalam proses pencarian kembali ketika diperlukan. Sebuah sistem kearsipan yang baik dan benar akan memperlancar kegiatan dan tujuan dari suatu lembaga, organisasi, atau perseorangan.

2.2.3 Pengertian Sistem Manajemen

Menurut Sora (2016), sistem manajemen adalah sebuah sistem yang digunakan dalam sebuah lembaga atau organisasi yang berfungsi untuk mengelola semua transaksi yang mendukung fungsi manajemen yang memiliki tujuan sebagai pendukung dalam pelaksanaan tugas guna pengambilan keputusan dalam suatu lembaga atau organisasi. Atau dengan kata lain, sistem manajemen merupakan sebuah sistem yang menghasilkan *Output* dengan masukan *Input* dan berbagai proses lainnya yang hasilnya dibutuhkan untuk tujuan tertentu dalam kegiatan manajemen.

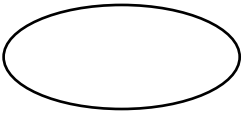



2.2.4 *Unified Modelling Language (UML)*

Menurut Sri Dharwiyanti dan Romi Satria Wahono (2003), *Unified Modelling Language (UML)* merupakan bahasa pemodelan yang menjadi standar dan digunakan untuk merancang, visualisasi, dan mendokumentasikan sistem perangkat lunak. Penggunaan dengan model UML berfungsi untuk mengidentifikasi bagian-bagian yang termasuk dalam lingkup sistem didalam aplikasi dan merupakan sebuah standar yang digunakan untuk merancang model dari sebuah sistem. Untuk mengembangkan aplikasi ini penulis menggunakan beberapa pemodelan UML yaitu *Use Case Diagram*, *Class Diagram*, dan *Activity Diagram*.

a. *Use Case Diagram*

Use Case Diagram menggambarkan aktivitas-aktivitas dan interaksi yang dilakukan oleh suatu sistem dari sudut pandang pengamatan luar. Simbol-simbol yang terdapat pada *Use Case Diagram* antara lain sebagai berikut:

Tabel 2.1 Simbol-Simbol *Use Case Diagram*

No.	Gambar	Nama Gambar	Keterangan
1.		<i>Use case</i>	Merupakan fungsionalitas yang disediakan sistem sebagai unit yang bertukar pesan dengan aktor.
2.		<i>Actor</i>	Merupakan penggambaran abstrak dari orang yang mengaktifkan fungsi dari target sistem dan merupakan orang yang berinteraksi dengan <i>use case</i> .
3.		<i>Association</i>	Digambarkan dengan garis tanpa panah mengindikasikan siapa yang berinteraksi secara langsung dengan sistem.
4.		<i>Include</i>	Mengidentifikasi hubungan antar dua <i>use case</i> dimana satu <i>use case</i> memanggil <i>use case</i> yang lain.

No.	Gambar	Nama Gambar	Keterangan
5.	-- <<extend>> ->	<i>Extend</i>	Merupakan perluasan dari <i>use case</i> jika kondisi atau syarat terpenuhi.

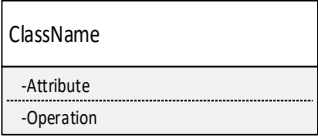
b. Class Diagram



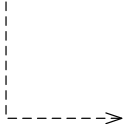
Class Diagram memberikan pandangan secara luas tentang sebuah sistem dengan cara menunjukkan kelas-kelas dan juga hubungan antar kelas tersebut. *Class Diagram* memiliki 3 komponen penting antara lain:

1. Nama, merupakan nama dari sebuah kelas.
2. Variabel, merupakan properti dari sebuah kelas.
3. Operasi atau *Method*, merupakan hal yang bisa dilakukan oleh sebuah kelas terhadap kelas lain.

Simbol-simbol yang terdapat pada *Class Diagram* antara lain sebagai berikut:

Tabel 2.2 Simbol-Simbol Class Diagram




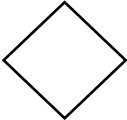

No.	Gambar	Nama Gambar	Keterangan
1.		<i>Class</i>	Merupakan simbol dalam sebuah <i>class</i> dimana dibagi menjadi 3 yaitu nama kelas, atribut, dan operasi.

No.	Gambar	Nama Gambar	Keterangan
2.		<i>Association</i>	Garis yang menghubungkan <i>class</i> satu dengan <i>class</i> lainnya dengan makna umum.
3.		<i>Aggregation</i>	Menghubungkan antar <i>class</i> dengan makna untuk semua bagian.
4.		<i>Dependency</i>	Menunjukkan operasi pada suatu <i>class</i> yang menggunakan <i>class</i> lain.

c. Activity Diagram

Activity diagram menggambarkan aliran aktivitas dalam rancangan sebuah sistem, dari awal aliran pada sebuah sistem, *decision* yang mungkin terjadi, dan akhir dari aliran pada sistem tersebut. *Activity* Diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa aktivitas. Simbol-simbol yang terdapat pada *Activity* Diagram antara lain sebagai berikut:


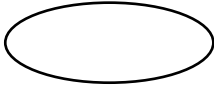
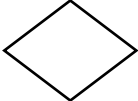
Tabel 2.3 Simbol-Simbol Activity Diagram

No.	Gambar	Nama Gambar	Keterangan
1.		<i>Initial Node</i>	Merupakan awal dari aktivitas dalam <i>activity diagram</i> .
2.		<i>Final Node</i>	Merupakan akhir dari aktivitas dalam <i>activity diagram</i> .
3.		<i>Action</i>	Menggambarkan suatu proses atau kegiatan dalam <i>activity diagram</i> .
4.		<i>Decision</i>	Pengambilan pilihan atau keputusan dalam <i>activity diagram</i> .
5.		<i>Fork Node</i>	Menggambarkan percabangan dari aktivitas dalam <i>activity diagram</i> .

2.2.5 Entity Relationship Diagram (ERD)

Menurut Riki dan Sutopo Hadi (2016), *Entity Relationship Diagram* merupakan suatu rancangan sistem yang berhubungan pada suatu kegiatan yang digambarkan dengan hubungan antar entitas dimana setiap entitas terdiri dari beberapa atribut yang memiliki fungsi dalam proses tersebut. Komponen-komponen dalam ER-Diagram dapat dilihat pada tabel berikut:

Tabel 2.4 Simbol-Simbol Entity Relationship Diagram

No.	Gambar	Nama Gambar	Keterangan
1.		<i>Entity</i>	Objek yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain.
2.		<i>Attribute</i>	Atribut yang dimiliki entitas, yang mendeskripsikan karakteristik dari entitas tersebut.
3.		<i>Relationship</i>	Hubungan yang terjadi antar entitas yang berbeda.

2.2.6 *Software Development Life Cycle (SLDC)*

Menurut Riki dan Sutopo Hadi (2016), *SDLC (Software Development Life Cycle)* merupakan tahapan-tahapan pekerjaan yang dilakukan oleh analis sistem dan *programmer* dalam kegiatan membangun sebuah sistem informasi. SDLC memiliki beberapa metode yang dapat digunakan dalam mengembangkan sebuah *software* seperti *Waterfall*, *Prototype*, *RAD (Rapid Application Development)*, dan *Agile Software Development*. Dalam proses pengembangan aplikasi ini penulis menggunakan metode *Waterfall*.

Disebut dengan *Waterfall* karena tahap demi tahap berjalan berurutan atau dengan kata lain tahapan yang akan dilalui harus menunggu selesainya tahap sebelumnya. Tahap-tahap yang digunakan adalah sebagai berikut:

1. *Software Requirements Analysis*

Tahap pengumpulan data dan analisis kebutuhan yang harus dipenuhi oleh aplikasi yang akan dibuat. Fase ini merupakan fase yang penting

karena merupakan tahap awal dalam pembuatan aplikasi dan agar bisa menghasilkan desain sistem yang baik.

2. *Design*

Setelah menentukan kebutuhan sistem yang akan dibuat, pada fase ini proses desain sistem dikerjakan.

3. *Coding*

Desain dari aplikasi yang akan dibuat diterjemahkan kedalam kode-kode dengan menggunakan bahasa pemrograman yang telah ditentukan agar menjadi bentuk visual aplikasi.

4. *Testing*

Pada tahap ini merupakan penyatuan unit-unit program untuk kemudian dilakukan pengujian dari fungsi-fungsi yang terdapat dalam aplikasi.

5. *Maintenance*

Pada tahap ini aplikasi yang sudah lolos tahap pengujian diimplementasikan di lapangan dan dilakukan pemeliharaan serta penyesuaian sistem pada aplikasi.

2.2.7 Pengertian Aplikasi *Web*

Menurut Ahmad Ghifary (2015), aplikasi *web* merupakan jenis aplikasi yang hanya bisa diakses dengan menggunakan *web browser* dalam suatu jaringan *internet* atau *intranet*. Aplikasi *web* juga merupakan salah satu jenis dari perangkat lunak komputer yang dikembangkan dengan menggunakan bahasa pemrograman yang mendukung seperti HTML, JavaScript, CSS, Php, C#, Java dan berbagai bahasa pemrograman lainnya.

Aplikasi *web* memiliki beberapa keunggulan dan kekurangan. Berikut ini merupakan beberapa keunggulan dari aplikasi web sebagai berikut:

1. Dapat diakses dimanapun dan kapanpun karena tidak perlu melakukan penginstalan.
2. Tidak memerlukan lisensi ketika menggunakan aplikasi *web*, karena lisensi merupakan tanggung jawab dari web yang menyediakan aplikasi.

3. Dapat dijalankan pada semua sistem operasi seperti *linux*, *windows*, dan sistem operasi lainnya yang dijalankan dengan menggunakan *browser* serta akses internet.
4. Dapat diakses melalui media seperti komputer dan *handphone* yang sesuai dengan standar WAP.
5. Tidak memerlukan spesifikasi komputer yang tinggi untuk menggunakan aplikasi berbasis *web*, karena proses dilakukan didalam *web server* yang menyediakan aplikasi berbasis *web*.

Dari beberapa keunggulan diatas, aplikasi web juga memiliki kekurangan sebagai berikut:

1. Membutuhkan koneksi *internet* atau *intranet* yang stabil, hal ini bertujuan agar aplikasi dapat berjalan dengan baik.
2. Membutuhkan sistem keamanan yang baik karena aplikasi dijalankan secara terpusat, sehingga apabila *server down* maka sistem aplikasi tidak akan berjalan.

2.2.8 *Microsoft SQL Server*

Menurut Teori Komputer (2015), *Microsoft SQL Server* adalah sistem manajemen basis data relasional yang menggunakan arsitektur *client server*. Beberapa fitur pada *Microsoft SQL Server* memiliki kelebihan dalam membuat *database mirroring* dan *clustering*.

Untuk memproses *query* ke dalam *database*, *Microsoft SQL Server* menggunakan bahasa SQL yang telah digunakan secara umum pada semua produk *database server*. *Microsoft SQL Server* memiliki beberapa kelebihan sebagai berikut:

1. Dapat digunakan untuk mengolah data yang besar, jenis *database* ini sangat cocok diterapkan pada perusahaan mikro, menengah maupun perusahaan yang besar sekalipun.

2. DBMS jenis ini memiliki kelebihan mengatur user data serta hak aksesnya.
3. Mempunyai tingkat keamanan pada data yang sangat baik.
4. Dapat melakukan *back up*, *recovery* dan *rollback* pada data dengan mudah.
5. Mempunyai keunggulan dalam membuat *database mirroring* dan *clustering*.

Selain beberapa kelebihan diatas, SQL Server juga mempunyai kelemahan sebagai berikut:

1. DBMS jenis ini hanya dapat digunakan pada sistem operasi atau *platform windows*.
2. Software ini mempunyai lisensi dari *Microsoft* sehingga untuk menggunakannya membutuhkan pembelian dengan biaya yang relatif mahal.

2.2.9 *Microsoft Visual Studio 2015*

Menurut Sam (2018), *Microsoft Visual Studio* adalah salah satu *Integrated Development Environment (IDE)* yang banyak digunakan dalam pengembangan program komputer, situs *web*, aplikasi *web*, layanan *web*, dan aplikasi *mobile*. *Visual Studio* juga menggunakan *platform* pengembangan perangkat lunak dari *Microsoft* seperti *Windows API*, *Windows Forms*, *Windows Presentation Foundation*, *Windows Store*, dan *Microsoft Silverlight*.

2.2.10 ASP.NET

Menurut Umayah Maya (2013), ASP.NET adalah suatu teknologi *server side* yang baru dan handal untuk membuat halaman web dinamis yang merupakan pengembangan terbaru dari *Microsoft Active Server Pages (ASP)*. ASP.NET merupakan tahapan pembuatan aplikasi *web* yang menyediakan fasilitas-fasilitas bagi pengembang untuk membangun aplikasi *web* karena menyatu dengan *.NET Framework* sehingga dapat membuat aplikasi dalam

berbagai macam bahasa yang kompatibel dengan .NET, termasuk *Visual Studio.NET*, *Visual Basic .NET*, *Jscript .NET*.

2.2.11 Bahasa Pemrograman C#

Menurut Teo Filus (2017), C# atau yang dibaca *C sharp* adalah sebuah bahasa pemrograman digunakan untuk berbagai fungsi antara lain untuk membangun aplikasi berbasis *desktop* atau *mobile*, pemrograman *server-side* pada *website*, pembuatan *game* dan sebagainya. Selain itu, C# juga mengukung konsep pemrograman berorientasi objek seperti *inheritance*, *class*, *polymorphism* dan *encapsulation*.

Bahasa pemrograman C# juga menggunakan *framework* yang disebut *.NET Framework* yang digunakan untuk menjalankan kode C#. C# dibuat dengan tujuan sebagai bahasa pemrograman utama dalam lingkungan *.NET Framework*. Sebelum adanya C#, *Microsoft* mengembangkan J++ yang bertujuan membuat Java agar dapat berjalan pada *platform Windows*, namun dikarenakan adanya masalah dari pihak luar maka *Microsoft* menghentikan proyek J++ dan beralih untuk mengembangkan bahasa baru yaitu C#.

2.2.12 *Hyper Text Markup Language* (HTML)

Menurut Constantianus dan Suteja (2005), HTML yang merupakan singkatan dari *Hyper Text Markup Language* merupakan rangkaian kode program yang digunakan dalam pengembangan visual sebuah halaman web. Secara umum, situs web yang di internet masih menggunakan HTML sebagai bahasa pemrograman utama mereka dan dalam sebuah halaman HTML berisi kumpulan informasi yang disimpan dalam tag-tag tertentu, dimana tag-tag tersebut digunakan untuk melakukan format terhadap informasi yang dimaksud. Berbagai pengembangan telah dilakukan terhadap kode HTML dan telah melahirkan teknologi-teknologi baru di dalam dunia pemrograman web dan HTML juga merupakan dasar dari bahasa web seperti PHP, ASP, JSP dan lainnya.

2.2.13 *Metode Black Box Testing*

Menurut M. Sidi Mustaqbal, Roeri Fajri Firdaus, dan Hendra Rahmadi (2016), *Black Box Testing* merupakan salah satu metode *testing* yang bertujuan untuk mengetahui spesifikasi fungsional dari sebuah perangkat lunak. Penguji dapat mendefinisikan kumpulan kondisi *input* dan melakukan pengujian pada fungsi-fungsi program. *Black Box Testing* cenderung untuk menemukan hal-hal berikut:

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (*interface errors*).
3. Kesalahan pada struktur data dan akses basis data.
4. Kesalahan performansi (*performance errors*).
5. Kesalahan inisialisasi dan terminasi.