

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1. Tinjauan Pustaka

Berdasarkan permasalahan yang ada, maka terdapat referensi-referensi yang dapat membantu dalam berjalannya penelitian ini. Berikut merupakan beberapa penelitian terdahulu

1. Rancang Bangun Sistem Informasi Kurikulum 2013 Tingkat Sekolah Dasar Berbasis Web dengan SDLC *Waterfall*., S Very Yulianto, A Prima Atmaja, *Open Acces Journal of Information System(OAJIS)* 2018.

Penelitian ini membangun sistem informasi pengelolaan nilai pada Kurikulum 2013. Tujuan daripada penelitian ini yaitu untuk memudahkan sekolah dalam menginputkan nilai agar terstruktur dan terhindar dari tumpukan data-data atau file. Metode yang digunakan untuk penyelesaiannya yaitu dengan metode (*System Development Life Cycle*)SDLC dengan model *Waterfall*. Perancangan aplikasi menggunakan *framework codeigniter* dan bahasa pemrograman PHP. Dalam pengujian sistem pada penelitian ini menggunakan pengujian sistem *Black box Testing*. [5]

Hasil penelitian menunjukkan bahwa dengan menggunakan sistem informasi ini, proses penilaian menjadi lebih mudah dan cepat. Dalam penggunaan sistem informasi ini, guru kelas perlu didampingi oleh operator sekolah yang berperan dalam mengelola data guru, siswa, orang tua atau wali, mata pelajaran, ekstrakurikuler, kelas dan user (akun guru kelas). Hasil pengujian sistem menggunakan metode *blackbox* menunjukkan bahwa semua fitur berjalan dengan baik. [5]

Perbedaan penelitian tersebut dengan penelitian yang dilakukan yaitu penelitian tersebut berfokus pada pengelolaan dalam penilaian kurikulum 2013 tingkat sekolah dasar sedangkan penelitian yang dilakukan berfokus pada pengelolaan pemesanan tiket dan penanganan kegiatan pada *Ketenger Adventure*

2. **Pembangunan Sistem Informasi Perpustakaan Berbasis Web menggunakan Metode *Rapid Application Development*(RAD)., Nur Aini, Satrio Agung Wicaksono, Issa Arwani, Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer 2019.**

Penelitian ini bertujuan untuk membangun sistem informasi perpustakaan yang terdapat di SMK Negeri 11 Malang. Dalam penyelesaiannya penelitian ini menggunakan metode *Rapid Application Development*(RAD). Hasil dari pembangunan sistem informasi perpustakaan dapat memudahkan admin dalam pengelolaan data-data perpustakaan disetiap waktu dan memudahkan pengguna dalam melakukan peminjaman buku dipergustakaan dengan hasil pengujian *black box valid* sebanyak 100%. Dan hasil dari uji penerimaan pengguna atau *User Acceptance Testing*(UAT) yang dilakukan kepada 5 penguji tersiri ketua perpustakaan, petugas perpustakaan, 1 guru, 1 staf dan 1 siswa memiliki hasil 84% pengguna setuju bahwa Sistem Informasi Perpustakaan Sekolah (SIPS) Malang telah memenuhi kebutuhan pengguna dalam peminjaman buku dipergustakaan SMKN 11 Malang.[6]

Perbedaan penelitian tersebut dengan penelitian yang dilakukan adalah pada penelitian tersebut dalam pembangunan sistem informasi menggunakan metode *Rapid Acception Development*(RAD) dan berfokus pada pengelolaan sistem perpustakaan sedangkan pada penelitian yang dilakukan dalam pembangunan sistem informasinya menggunakan metode *System Development Life Cycle* (SDLC) berfokus pada pengelolaan pemesanan Tiket Wisata Ketenger Adventure.

3. **Perancangan Sistem Informasi *E-Voting* Pemilihan Ketua RT Dengan Verifikasi Nomor Induk Keluarga Berbasis Web., Adi Hermawansyah, Rosmiati Nur, Jurnal Sistem Informasi 2019.**

Pada penelitian ini merancang sistem informasi *E-voting* pemilihan ketua RT dengan verifikasi Nomor Induk Keluarga berbasis web. Tujuan dari penelitian ini yaitu untuk mempermudah warga dalam memilih ketua RT dimanapun sehingga hasil suara yang diharapkan lebih maksimal agar kegiatan pemilihan ketua RT ini dapat berjalan dengan optimal dan efisien. Metode penelitian ini adalah studi lapangan, studi pustaka dan wawancara. Metode pengumpulan data dalam penelitian ini yaitu metode

kualitatif serta metode perancangan sistem adalah metode *System Development Life Cycle*(SDLC). Perancangan *database* pada penelitian tersebut menggunakan MySQL sebagai *database server* dan bahasa pemrograman PHP.[7]

Perbedaan penelitian tersebut dengan penelitian yang dilakukan yaitu pada penelitian tersebut berfokus terhadap perancangan informasi untuk pengelolaan e-voting sedangkan penelitian yang dilakukan berfokus terhadap pembangunan sistem informasi pemesanan tiket dan pengelolaannya.

4. **Sistem Informasi Penerimaan Pengemudi Berbasis Web Dengan Metode *Prototype* : Studi Kasus PT. Dinamika Makmur Sentosa Cikarang., Diah Wijayanti, R Dwi Asworowati, Y Kety Ananta, Indonesian Jurnal on Computer and Information Technology 2018.**

Pada penelitian ini merancang sistem informasi penerimaan pengemudi berbasis web dengan studi kasus di PT. Dinamika Makmur Sentosa Cikarang. Tujuan penelitian ini untuk menginformasikan lowongan secara cepat dan tepat, memudahkan calon pengemudi melakukan pendaftaran online dan memberikan informasi secara online bagi pelamar yang diterima sebagai pengemudi. Metode yang digunakan pada penelitian tersebut dalam pengembangan sistemnya menggunakan metode *prototype* dimana alur dan tahapan yang digunakan meliputi mengidentifikasi pengguna dan mengembangkan *prototype*. Hasil dari penelitian tersebut adalah sistem informasi penerimaan pengemudi berbasis web yang diharapkan dapat memudahkan proses pengolahan data pengemudi pada PT. Dinamika Makmur Sentosa Cikarang.[8]

Perbedaan penelitian tersebut dengan penelitian yang dilakukan yaitu dari aspek pengembangan sistem yang dilakukan menggunakan metode *prototype* sedangkan pada penelitian yang dilakukan dalam pengembangan sistem menggunakan metode *Waterfall*

5. **Sistem Informasi Pemesanan Tiket Berbasis Web pada Perum Damri Palembang Menggunakan Metode *Waterfall*. Bakhtiar K. Jurnal JUPITER 2018**

Pada penelitian ini membangun sistem informasi *ticketing* untuk pemesanan tiket transportasi umum layanan angkutan bus antar kota, antar provinsi dan bus pariwisata pada Perum Damri Palembang. Tujuan penelitian ini untuk membangun aplikasi

berbasis web yang memanfaatkan teknologi internet, yang dapat menyebarluaskan informasi tentang kegiatan layanan-layanan Perum Damri cabang Palembang kepada masyarakat guna memudahkan pengguna jasa dalam memperoleh informasi, melakukan pemesanan tiket dan mengetahui jadwal keberangkatan secara cepat, akurat dan efisien. Metode yang digunakan dalam penelitian tersebut yaitu metode *waterfall*. Hasil dari penelitian ini adalah sebuah perangkat lunak aplikasi yang dapat melakukan fungsinya sesuai dengan kebutuhan fungsional perangkat yang diharapkan berupa sistem informasi pemesanan tiket berbasis *web*. [9]

Perbedaan penelitian tersebut dengan penelitian yang dilakukan terdapat pada studi kasus, dimana penelitian tersebut berfokus pada pembangunan sistem informasi pemesanan tiket transportasi umum pada Perum Damri Palembang. Sedangkan penelitian yang dilakukan berfokus pada pembangunan sistem informasi pemesanan tiket wisata Ketenger *Adventure*.

Tabel 2. 1 Ringkasan Penelitian Terdahulu

KATEGORI	PENELITIAN SEBELUMNYA					PENULIS Ripta Septiyana (2019)
	S Veri Yulianto <i>et al</i> (2018)	Nur Aini <i>et al</i> (2019)	Adi Hermawansyah <i>et al</i> (2019)	Diah Wijayanti <i>et al</i> (2018)	K Bakhtiar <i>et al</i> (2018)	
Judul	Rancang Bangun Sistem Informasi Kurikulum 2013 Tingkat Sekolah Dasar Berbasis Web dengan metode <i>Waterfall</i>	Pembangunan Sistem Informasi Perpustakaan Berbasis Web menggunakan Metode <i>Rapid Application Development</i> (RAD)	Perancangan Sistem Informasi <i>E-voting</i> Pemilihan Ketua RT Dengan Verifikasi Nomor Induk Keluarga Berbasis Web	Sistem Informasi Penerimaan Pengemudi Berbasis Web Dengan Metode Prototype : Studi kasus PT. Dinamika Makmur Sentosa Cikarang	Sistem Informasi Pemesanan Tiket Berbasis Web pada Perum Damri Palembang Menggunakan Metode <i>Waterfall</i>	Sistem Informasi Pemesanan Tiket Wisata Ketenger Adventure Berbasis Web dengan Metode <i>Waterfall</i>
Metode	<i>Waterfall</i>	<i>Rapid Application Development</i> (RAD)	<i>Waterfall</i>	<i>Prototype</i>	<i>Waterfall</i>	<i>Waterfall</i>
Objek Penelitian	Pengelolaan nilai	Pengelolaan data-data Perpustakaan	Pengelolaan <i>E-voting</i>	Pengelolaan Penerimaan Pengemudi	Akses informasi dan pemesanan tiket	Pengelolaan Pemesanan Tiket dan Kegiatan pada wisata Ketenger Adventure

Berdasarkan Tabel 2. 1 maka dapat disimpulkan bahwa dalam proses penelitian dan membangun sistem informasi terdapat tahapan dan metode pengembangan yang berbeda-beda. Adapun perbedaan tahapan metode pengembangan sistem dapat dijelaskan pada Tabel 2. 2

Tabel 2. 2 Perbedaan Tahapan Metode /Model

Nama Metode/Model	Tahapan
<i>Waterfall</i>	Bersifat sistematis dan skuensial, memiliki tahapan yaitu : <ol style="list-style-type: none"> 1. <i>Requirement analysis and definition</i> 2. <i>System and software design</i> 3. <i>Implementation and unit testing</i> 4. <i>Operation and maintenance</i>
<i>Prototyping</i>	Bersifat iterative, memiliki tahapan : <ol style="list-style-type: none"> 1. <i>Listen to customer</i> 2. <i>Building Mockup</i> 3. <i>Customer test drive</i>
<i>Rapid Application Development(RAD)</i>	Bersifat skuensial dan iterative, memiliki tahapan yaitu : <ol style="list-style-type: none"> 1. <i>Requirements Planning</i> 2. <i>Desain workshop</i> 3. <i>Implementation</i>

Berdasarkan table 2. 2 maka dapat disimpulkan bahwa metode pengembangan *Rapid Application Development(RAD)*, *Prototyping*, dan *Waterfall* terdapat perbedaan dalam tahapannya. Adapun perbedaannya yaitu *Rapid Application Development(RAD)* merupakan sebuah model proses perkembangan perangkat lunak sekuensial linier yang menekankan siklus perkembangan yang sangat pendek dapat ditempuh 60 sampai 90 hari dan memiliki tahapan seperti *Requirements Planning*, *Design Workshop* dan *Implementation*[10]. *Prototype* merupakan sebuah versi awal dari perangkat lunak yang digunakan untuk mendemonstrasikan konsep, mencoba berbagai pilihan desain, dan menggali lebih banyak permasalahan dan solusinya tahapannya meliputi *Listen to Customer*, *Building mockup*, dan *Customer test drive mockup*[11]. *Waterfall* merupakan model pengembangan sistem

informasi yang sistematis dan sekuensial serta memiliki tahapan seperti *Requirement analysis and definition, System and software design, Implementation and unit testing* dan *Operation and maintenance*[12]. Selain itu, pada metode *waterfall* akan menghasilkan kualitas sistem yang baik dikarenakan pelaksanaannya secara bertahap sehingga tidak berfokus terhadap tahapan tertentu. Tujuan metode *waterfall* untuk membangun sebuah sistem dari awal dengan mengumpulkan kebutuhan sistem yang akan dibangun sesuai dengan topik penelitian yang dipilih sampai dengan produk tersebut diuji[13].

Ditinjau dari tujuan penelitian wisata *adventure* ini memiliki tujuan diantaranya meningkatkan proses pelaporan, pengelolaan dan penjadwalan kegiatan dengan menerapkan sistem informasi dan membangun sistem informasi pemesanan tiket. Pada penelitian ini menggunakan metode *Waterfall* sebagai metode pengembangan sistem. Metode *waterfall* memiliki model tahapan yang sesuai kriteria daripada konteks permasalahan dalam penelitian ini. Metode *Waterfall* memiliki tahapan yang lebih intensif dan fleksibel dibandingkan dengan metode yang telah dikaji sebelumnya [14] di Tabel 2.1. Dengan adanya penelitian ini diharapkan mampu menjawab permasalahan yang telah dipaparkan.

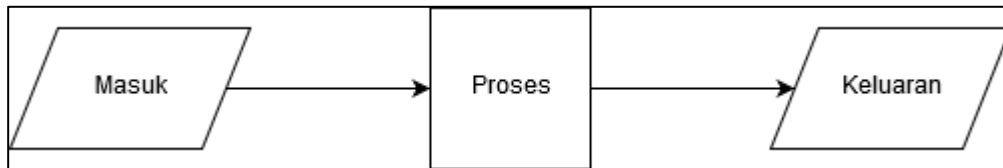
2.2. Landasan Teori

2.2.1. Sistem Informasi

Sistem informasi merupakan elemen-elemen yang saling berinteraksi untuk mencapai tujuan tertentu, mempunyai beberapa komponen yang saling terkait dan membentuk jalinan kerja yang kompak untuk mencapai sasaran. Pengamatan dan kenyataan menunjukkan bahwa perkembangan dan terobosan teknologi informasi akan terus berlanjut di masa depan. Tidak sulit untuk memperkirakan bahwa salah satu ujian bagi kemahiran dan keandalan manajemen dimasa depan ialah kemampuannya memanfaatkan perkembangan teknologi tersebut. Kemampuan manajemen memanfaatkan informasi dalam menjalankan fungsi-fungsi manajerial akan turut menentukan berhasil tidaknya manajemen yang bersangkutan meraih keberhasilan dalam mengelola organisasi yang dipimpinnya[15].

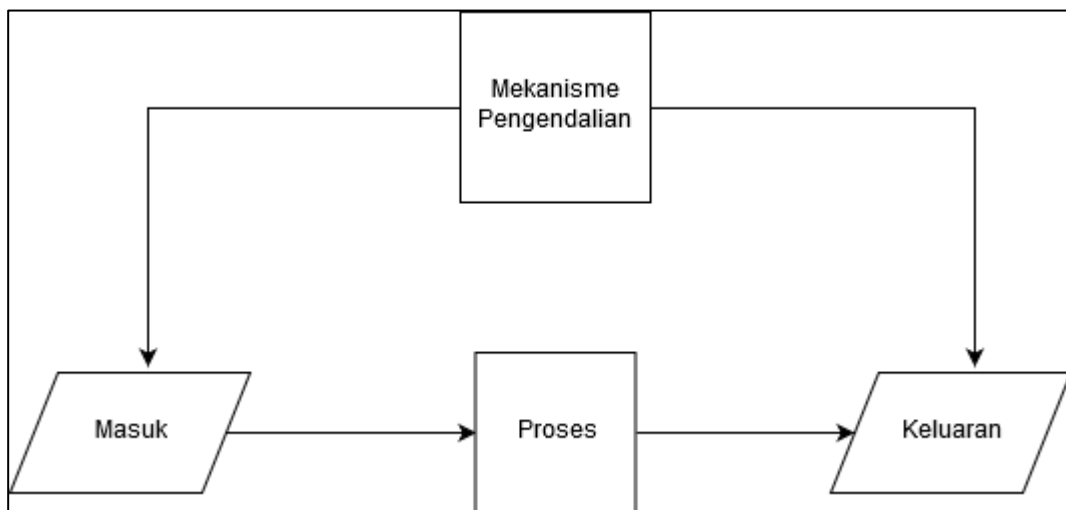
Sistem Informasi manajemen dapat mendukung dalam pengambilan keputusan secara cepat dan tepat. SIM berbasis komputer ini sangat dibutuhkan dalam rangka pengembangan organisasi untuk menghadapi persaingan global yang semakin erat. Adapun prinsip didalam sistem dikelompokkan menjadi 2 yaitu sistem

terbuka dan sistem tertutup. Sistem terbuka adalah sistem yang dihubungkan dengan lingkungannya melalui arus sumber daya. Secara sederhana sistem terbuka dapat dilihat pada Gambar 2. 1



Gambar 2. 1 Sistem Terbuka

Sistem tertutup adalah sistem yang tidak berinteraksi secara langsung dengan lingkungannya melalui arus sumber daya[15]. Skema sistem tertutup dapat dilihat pada Gambar 2. 2



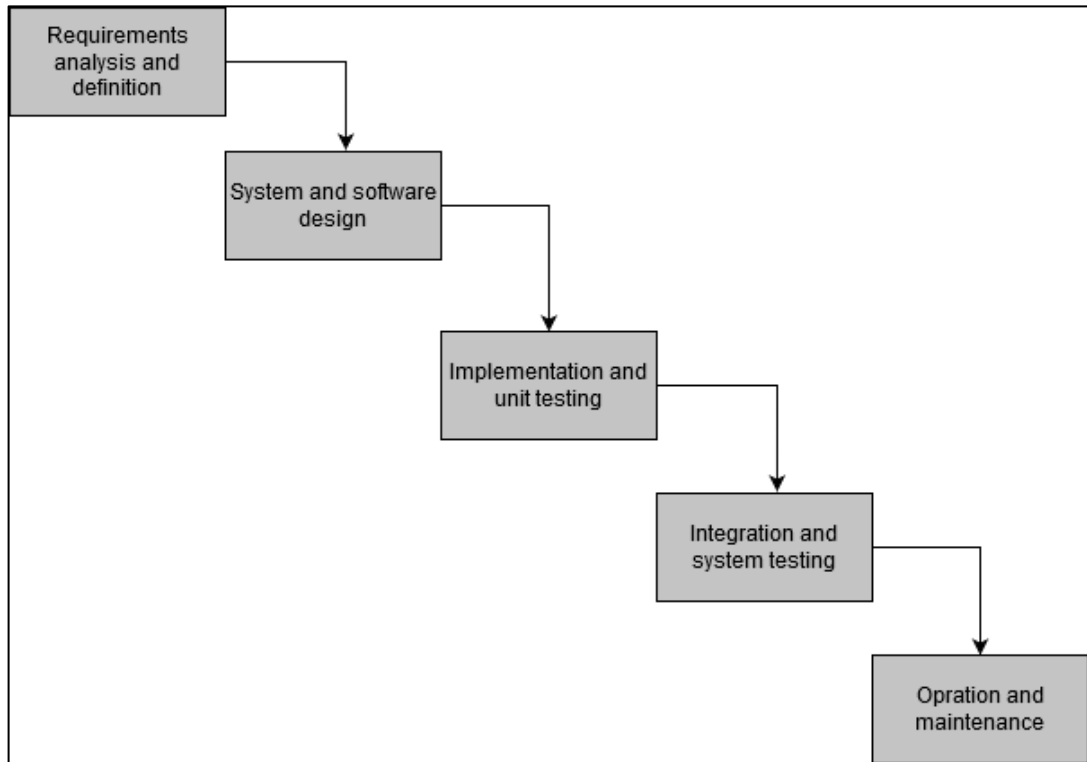
Gambar 2. 2 Sistem Tertutup

Management yang baik sangat diperlukan dalam setiap organisasi untuk menjaga kelangsungan hidup organisasi dan pengembangannya. *System Information Management* (SIM) konsep manajemen memegang peranan yang sangat penting dalam pengembangan sistem informasi yang diterapkan, sebab tanpa adanya *management* yang baik sistem informasi tidak akan dapat berkembang dengan pesat.

2.2.2. Metode Waterfall

Metode penelitian yang diterapkan pada penelitian ini yaitu dengan pengembangan metode *waterfall*. Metode *waterfall* merupakan model pengembangan sistem

informasi yang sistematis dan sekuensial[12]. Metode *waterfall* memiliki tahapan-tahapan sebagai berikut :



Gambar 2. 3 Metode *Waterfall*

a. *Requirements analysis and definition*

Layanan sistem, kendala dan tujuan ditetapkan oleh hasil konsultasi dengan pengguna yang kemudian didefinisikan secara rinci dan berfungsi sebagai spesifikasi sistem

b. *System and software design*

Tahapan perancangan sistem mengalokasikan kebutuhan-kebutuhan sistem baik perangkat keras maupun perangkat lunak dengan membentuk arsitektur sistem secara keseluruhan. Perancangan perangkat lunak melibatkan identifikasi dan penggambaran abstraksi sistem dasar perangkat lunak dan hubungannya.

c. *Implementation and unit testing*

Pada tahap ini, perancangan perangkat lunak direalisasikan sebagai serangkaian program. Pengujian melibatkan verifikasi bahwa setiap unit memenuhi spesifikasinya.

d. Integration and system testing

Unit-unit individu program atau program digabung dan diuji sebagai sebuah sistem lengkap untuk memastikan apakah sesuai dengan kebutuhan perangkat lunak atau tidak. Setelah pengujian, perangkat lunak bisa dikirim ke *customer*

e. Operation and maintenance

Biasanya (walaupun tidak selalu), tahapan ini merupakan tahapan yang paling panjang. Sistem dipasang dan digunakan secara nyata. *Maintenance* melibatkan pembetulan kesalahan yang tidak ditemukan pada tahapan-tahapan sebelumnya, meningkatkan implementasi dari unit sistem, dan meningkatkan layanan sistem sebagai kebutuhan baru

2.2.3. Booking Online

Booking/Pemesanan adalah kegiatan untuk memesan sebuah tempat sesuai waktu dan tanggal perjalanan yang di inginkan. *Online* adalah keadaan disaat seseorang terhubung kedalam suatu jaringan ataupun sistem yang lebih besar. *Online booking* adalah bagian kegiatan dari seseorang yang melakukan pemesanan atau reservasi terhadap suatu produk atau jasa melalui media *online*. Industri pariwisata telah mencapai keunggulan kompetitif dari sektor lain, karena telah mampu *mentransfer* lebih banyak pelanggan secara *online*[16].

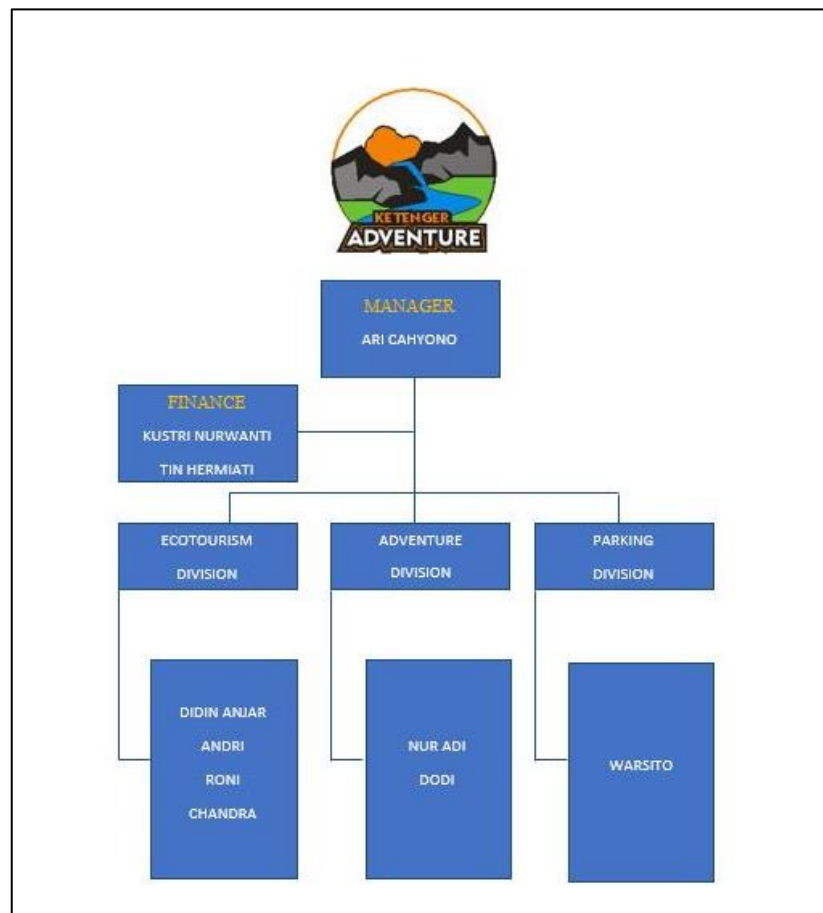
2.2.4. E-ticketing

E-ticketing atau *electronic ticketing* adalah merupakan cara untuk mendokumentasi proses penjualan dari aktivitas perjalanan pelanggan tanpa harus mengeluarkan dokumen berharga secara fisik ataupun paper *ticket*. Semua informasi mengenai *electronic ticketing* disimpan secara digital dalam sistem komputer milik *airline*. Sebagai bukti pengeluaran *E-ticket*, pelanggan akan diberikan *Invoice* yang hanya berlaku sebagai alat untuk dapat melakukan kegiatan wisata. E-ticketing adalah peluang untuk meminimalkan biaya dan mengoptimalkan kenyamanan pengunjung. E-ticketing mengurangi biaya proses tiket, menghilangkan formulir kertas dan

meningkatkan fleksibilitas penumpang dana gen perjalanan dalam membuat perubahan-perubahan dalam jadwal perjalanan.[17]

2.2.5. Struktur Organisasi

Struktur organisasi merupakan sistem yang harus dilaksanakan oleh manajer untuk menggerakkan aktivitas dalam mewujudkan kesatuan tujuan. Struktur organisasi harus selalu dievaluasi untuk memastikan konsistensinya dalam pelaksanaan operasi yang efektif dan efisien untuk memenuhi kebutuhan sekarang. [18] Ketenger Adventure memiliki struktur organisasi sebagai berikut :



Gambar 2. 4 Struktur Organisasi Ketenger Adventure

(Sumber : Bag. Manajer Ketenger Adventure)

Berdasarkan Gambar 2. 4 dapat dilihat bahwa Ketenger Adventure dibawah langsung oleh Bapak Ari Cahyono dengan memiliki dua bendahara yaitu Ibu Kusni Nurwanti dan Ibu Tin Hermiati. Ketenger Adventure juga memiliki tiga divisi diantaranya

ecotourism division, *adventure division* dan *parking division*. Pada struktur organisasi tersebut setiap divisi memiliki tugas masing-masing diantaranya pada divisi *ecotourism division* memiliki tugas untuk mengelola curug bayan, *adventure division* memiliki tugas untuk mengelola wisata paket *adventure* dan *parking division* bertugas mengelola lahan parkir. Pada penelitian ini berfokus pada bisnis proses yang ada di wisata paket *adventure*.

Berdasarkan hasil wawancara dengan pihak manajer, bisnis proses dalam pemesanan paket kegiatan dapat dilakukan dengan menghubungi *contact person* bapak jajat dengan nomor *handphone* +6285726120567 atau datang langsung ke sekretariat Ketenger *Adventure*. Adapun calon pengunjung/*customer* dalam melihat informasi paket dapat dilakukan dengan meminta brosur ke pihak Ketenger *Adventure*. Ketenger *Adventure* dalam penanganannya tidak jarang mendapati pemesanan paket kegiatan lebih dari satu paket dalam satu waktu hal ini dibutuhkannya tim yang lebih banyak untuk meng-*handle* kegiatan. Pesanan paket yang kompleksitasnya tinggi dan membutuhkan alternatif tempat yang banyak biasanya tim hanya mampu melaksanakan satu *event* dalam satu waktu. Sebaliknya jika pesanan tidak terlalu kompleks dan tidak terlalu membutuhkan tempat yang banyak meskipun ada pesanan lebih dari satu *event* dalam satu waktu masih dapat dilaksanakan dengan melakukan *outside sourcing* atau mengambil orang dari luar sebagai tim tambahan.

2.2.6. Black Box Testing

Black Box Testing adalah pengujian yang mengabaikan mekanisme internal dari sistem atau komponen dan hanya berfokus pada *output* yang dihasilkan sebagai *respon* terhadap input yang dipilih dan kondisi eksekusi. Ada 7 (tujuh) jenis level pengujian yang terlibat dalam test. Ada dua hal yang perlu dipertimbangkan dalam jenis pengujian, yang pertama adalah *opacity* yaitu pandangan kode pengujian (*Black Box Testing* dan *White Box Testing*). [19] Salah satu jenis pengujian *Black Box* yang sering digunakan adalah *functional testing*. *Functional testing* menggunakan teknik pengujian *black box*, pengujian menguji desain tingkat tinggi dan pelanggan melakukan spesifikasi untuk merencanakan uji kasus untuk memastikan kode apa yang dimaksud. *Functional testing* memastikan bahwa fungsi yang ditentukan dalam spesifikasi persyaratan dapat bekerja. Pengujian Sistem menempatkan program baru

di banyak lingkungan yang berbeda untuk memastikan program ini bekerja di lingkungan pelanggan dengan berbagai versi dan jenis operasi dana atau aplikasi. Pengujian sistem dilakukan pada sistem yang terintegrasi lengkap untuk mengevaluasi kepatuhan sistem dengan persyaratan yang ditentukan. Oleh karena itu, pengujian sistem dilakukan dengan implementasi sistem lengkap dan lingkungan, beberapa kelas pengujian dilakukan yaitu untuk memeriksa sifat non-fungsional dari sistem.

2.2.7. Bootstrap

Bootstrap adalah paket aplikasi siap pakai untuk membuat *frond-end* sebuah *website*. Bisa dikatakan, Bootstrap adalah template desain web dengan fitur *plus*. Bootstrap diciptakan untuk mempermudah proses desain web bagi berbagai tingkat pengguna, mulai dari level pemula hingga yang sudah berpengalaman. Cukup bermodalkan pengetahuan dasar mengenai HTML dan CSS[20].

Paket Bootstrap berisi sekumpulan file CSS, font, dan JavaScript yang siap diintegrasikan ke sebuah dokumen HTML menggunakan kaidah-kaidah tertentu. Dokumen HTML yang dihasilkan pun secara dinamis akan tampil dalam layout yang disesuaikan dengan ukuran layar piranti pengunjung.

Daya tarik Bootstrap terletak pada kemudahan dan kepraktisan penggunaannya. Tata layout dan pewarnaannya juga bersih, simpel, indah dan berkesan modern. Selain itu, mengingat akses web via piranti mobile semakin meningkat tajam, maka pendekatan desain web responsive dan mobile-friendly yang ditawarkan Bootstrap pun juga menjadi solusi praktis dan murah bagi para pengembang web.

2.2.8. Bahasa Pemrograman PHP

PHP singkatan dari *Hypertext Preprocessor* yang merupakan server-side programming, yaitu bahasa pemrograman yang diproses disisi server. Fungsi utama PHP dalam membangun website adalah untuk melakukan pengolahan data pada database. Data website akan dimasukan ke database, diedit, dihapus dan ditampilkan pada website yang diatur oleh PHP[21].

2.2.9. CSS

CSS singkatan dari *Casecading Style Sheets*, yaitu skrip yang digunakan untuk mengatur desain website. Walaupun HTML mempunyai kemampuan untuk mengatur tampilan website, namun kemampuannya sangat terbatas. Fungsi Ccss adalah

memberikan pengaturan yang lebih lengkap agar struktur website yang dibuat dengan HTML terlihat lebih rapi dan elegan[21].

2.2.10. HTML

HTML singkatan dari *Hyper Text Markup Language*, yaitu skrip yang berupa tag-tag untuk membuat dan mengatur struktur website. Beberapa tugas utama HTML dalam membangun website, diantaranya sebagai berikut[21].

- a. Menentukan layout website.
- b. Memformat text dasar, seperti pengaturan paragraf dan format font.
- c. Membuat list.
- d. Membuat tabel.
- e. Menyisipkan gambar, video, dan audio.
- f. Membuat link.
- g. Membuat formulir.

2.2.11. Use Case View

Use case digunakan untuk memodelkan fungsionalitas-fungsionalitas sistem atau perangkat lunak dilihat dari pengguna yang ada diluar sistem (yang sering dinamakan sebagai aktor). *Use case* pada dasarnya merupakan unit fungsionalitas koheren yang diekspresikan sebagai transaksi-transaksi yang terjadi antara aktor dan sistem. Kegunaan dari *use case view* adalah untuk mendaftarkan aktor-aktor dan *use case-use case* dan memperlihatkan aktor mana yang berpartisipasi dalam masing-masing *use case*. Definisi *use case* didalamnya mencakup semua perilaku yang ada dalam sistem yang sedang kita kembangkan : urutan-urutan pertama, berbagai variasi dari perilaku normal (*exception*), serta tanggapan yang dikehendaki. Dari sudut pandang pengguna (*user*), mungkin saja ada berbagai situasi yang abnormal (*exception*). [22]

Dari sudut sistem atau perangkat lunak, ada berbagai variasi tambahan yang harus dideskripsikan dan ditangani. Dalam model eksekusi masing-masing *use case* harus diupayakan sedemikian rupa sehingga masing-masing *use case* bersifat mandiri satu dengan yang lainnya, meskipun implementasi suatu *use case* mungkin membuat keberuntungan implisit antar objek-objek yang dibagikan. Masing-masing *use case* merepresentasikan fungsionalitas mandiri dimana eksekusinya dapat digabungkan

dengan eksekusi *use case* yang lainnya. Adapun relasi-relasi dalam *use case* yaitu sebagai berikut :

- a. **Asosiasi** : Lintasan komunikasi antara aktor dengan *use case*.
- b. **Extend** : Penambahan perilaku ke suatu *use case* dasar.
- c. **Generalisasi use case** : Menggambarkan hubungan antara *use case* yang bersifat umum dengan *use case* yang bersifat lebih spesifik.
- d. **Include** : Penambahan perilaku ke suatu *use case* dasar yang secara eksplisit mendeskripsikan penambahan tersebut.

2.2.12. Sequence Diagram

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirim dan diterima antar objek[23]. Adapun bagan dari *sequence diagram* adalah sebagai berikut :

- a. **Entity class** : Bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
- b. **Boundary class** : Berisi kumpulan kelas yang menjadi interface atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan *formentry* dan *form* cetak.
- c. **Control class** : Suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
- d. **Message** : Suatu objek untuk mengirim pesan antar *class*.
- e. **Recursive** : Menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
- f. **Activation** : Mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
- g. **Life line** : Garis titik-titik yang terhubung dengan objek, sepanjang *life line* terdapat *activation*.

2.2.13. Activity Diagram

Activity Diagram merupakan kumpulan aksi-aksi. Pada aksi-aksi melakukan langkah satu kali saja tidak boleh dipecah menjadi beberapa langkah lagi. Sebagai contoh fungsi matematika, pemanggilan perilaku, pemrosesan data. Aktivitas dapat mengakses atribut dan operasi dan operasi *classifier*, tiap objek yang terhubung dan parameter-parameter jika aktivitas memiliki hubungan dengan perilaku. Ketika digunakan untuk model bisnis, informasi itu biasanya disebut *process-relevant data*[24]. Penjelasan bagan dalam *activity* diagram adalah sebagai berikut :

- a. **Partition** : Memperlihatkan dimana aliran berawal.
- b. **Relasi** : Merupakan langkah atau aksi-aksi yang terjadi.
- c. **Decision** :Memperhatikan dimana keputusan diambil selama terjadi aliran kerja.
- d. **Fork** : Merupakan percabangan dari sebuah aksi-aksi yang terjadi.
- e. **Final Node** : Memperlihatkan dimana aliran itu berakhir.

2.2.14. Class Diagram

Class Diagram bersifat statis. Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula diagram kelas memuat kelas-kelas aktif[24]. Adapun penjelasan bagan daripada *class* diagram adalah sebagai berikut :

- a. **Generalization** : Menggambarkan relasi generalisasi.
- b. **Realize** : Menggambarkan relasi realisasi.
- c. **Interface** : Menambahkan kelas antarmuka (*interface*) pada diagram.
- d. **Return Message** : Menggambarkan pengembalian dari pemanggilan prosedur.
- e. **Aggregation** : Menggambarkan relasi agresi.
- f. **Actor** : Menggambarkan aktor pada diagram kelas.
- g. **Boundary** : Menggambarkan kelas batasan (*boundary*) pada diagram.