



BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Beberapa penelitian yang memiliki keterkaitan dengan Penyeberangan Kapal dengan hasil yang relevan antara lain : Tantri Hidayat Sinaga [4], melakukan sebuah penelitian pada Kawasan Pariwisata Danau Toba dan Pulau Samosir, dimana pada Kawasan Pariwisata Danau Toba dan Pulau Samosir memiliki sebuah masalah pada proses pencatatan data manifes penumpang penyeberangan di Danau Toba. Pada proses pencatatan tidak mencatat nama penumpang serta dalam laporan masih berupa rekapitulasi daftar perjalanan kapal dan muatan secara umum, sehingga tingkat keakuratan data akan sangat diragukan kebenarannya. Sehingga dibutuhkan sebuah aplikasi untuk mengelola data manifes penumpang kapal dan mempermudah pemantauan kegiatan penyeberangan. Penelitian ini menggunakan metode *PIECES* dan *UML*. Hasil dari penelitian ini berupa model *UML* dan *prototype* aplikasi berbasis web. Kekurangan dari penelitian ini yaitu belum adanya implementasi *website* dari hasil *prototype*.

Penelitian lain yang dilakukan oleh Khairul Imtihan, Muhamad Hasyim Basri [5], melakukan sebuah penelitian pada PT. Mentari Sejati Perkasa, dimana pada PT. Mentari Sejati Perkasa memiliki sebuah masalah pada proses pembuatan data manifes bongkar muat barang. Pada proses pencatatan pembuatan manifes atau data muatan kapal masih dengan cara tulis tangan yaitu dengan mengisi formulir muatan, hasil isian tersebut kemudian direkap dan ketik ulang menggunakan *Microsoft Excel* dalam proses rekapitulasi dan pembuatan laporan terkadang membutuhkan waktu yang lama karna menunggu hasil isian formulir muatan, sehingga berdampak pada proses validasi dan pengiriman laporan yang mengakibatkan pembengkakan biaya dalam setiap kegiatan bongkar muat. Sehingga dibutuhkan sebuah aplikasi untuk mempermudah dan mempercepat proses pendataan muatan dan data manifes muatan, serta pembuatan laporan dan pengiriman laporan. Penelitian ini menggunakan metode analisis *PIECES* serta menggunakan metode *SDLC* dalam perancangan

sistemnya. Hasil dari penelitian adalah sistem informasi manifes berbasis *desktop* dan *android*. Saran untuk penelitian berikutnya yaitu menggunakan *barcode* dalam pencatatan transaksi, tanpa perlu *input* manual ke sistem.

Penelitian lain telah dilakukan oleh Basten Handali, Munawir, Susmanto [6], melakukan sebuah penelitian pada PT ASDP, dimana pada PT ASDP memiliki sebuah masalah pada minimnya informasi keberangkatan kapal serta proses pembelian tiket yang mana membutuhkan waktu yang lama dalam mengantri pembelian tiket. Pada proses pembelian tiket masih secara langsung yang mana menyebabkan antrian yang panjang serta minimnya informasi menjadikan masyarakat yang berada di pulau terpencil tidak dapat dengan mudah melihat jadwal keberangkatan kapal. Sehingga dibutuhkan sebuah sistem untuk memudahkan penumpang melihat jadwal keberangkatan kapal serta pemesanan tiket. Penelitian ini menggunakan bahasa pemrograman *php*, *css*, *html* dan *javascript*. Hasil dari penelitian adalah perancangan sistem informasi pemesanan e-tiket kapal feri Pelabuhan Sinabung Kabupaten Simeulue pada PT ASDP yang dapat memudahkan penumpang dalam memesan tiket serta melihat jadwal keberangkatan kapal. Kekurangan pada penelitian ini adalah tidak adanya metode dalam perancangan sistemnya, serta belum adanya integrasi antara rekening pembayaran dengan aplikasi *e-booking*.

Penelitian berikutnya telah dilakukan oleh Masriadi, Muhammad Ikhlas, Romi Wijaya [7], melakukan sebuah penelitian di Pelabuhan Teluk Bungus Mentawai, dimana pada Pelabuhan Teluk Bungus Mentawai memiliki masalah pada penjualan tiket. Pada proses penjualan tiket masih dilakukan secara manual serta dalam laporan penjualan tiket masih ditulis dalam buku besar, serta penjualan tiket masih kurang terorganisir. Sehingga dibutuhkan sebuah aplikasi untuk mengelola data penjualan tiket. Penelitian ini berbasis *desktop* dalam pengimplementasian sistemnya. Kekurangan pada penelitian ini adalah tidak adanya penjelasan terkait metode pengembangan sistem yang digunakan serta penggunaan *framework* maupun bahasa pemrograman apa dalam proses pengembangan sistem ini.

Penelitian selanjutnya dilakukan oleh Nurul Afni, Roida Pakpahan, Astri Rezky Jumarah [8], melakukan suatu penelitian pada Sekretariat Nasional Ikatan Arsitek Indonesia, dimana pada Sekretariat Nasional Ikatan Arsitek Indonesia terdapat permasalahan pada proses penggajian karyawan. Pada proses penggajian karyawan bagian keuangan harus membaca rekapitulasi absensi karyawan, kasbon serta menghitung gaji net karyawan dan meminta persetujuan kepada ketua umum. Pada proses tersebut masih belum efisien karena data masih berbentuk kertas dan kurangnya tempat penyimpanan laporan. Sehingga dibutuhkan sebuah sistem untuk mempermudah proses penggajian karyawan dan rekap data laporan penggajian. Penelitian ini menggunakan sistem berbasis *website*. Hasil dari penelitian ini adalah *prototype* sistem penggajian serta *website* penggajian karyawan yang dapat memudahkan dalam proses rekap data serta laporan penggajian karyawan. Kekurangan dari penelitian ini adalah tidak adanya penjelasan terkait bahasa pemrograman yang digunakan dalam pengembangan sistem serta metode pengembangan sistem.

Dari penelitian yang akan dilakukan memiliki suatu perbedaan dengan penelitian-penelitian sebelumnya. Perbedaan tersebut diantaranya yaitu penggunaan *Framework Laravel* dalam pembangunan sistem ini, yang mana menggunakan bahasa pemrograman PHP (*Hypertext Preprocessor*) dan *MySQL* sebagai aplikasi *database*. Tahap pengembangan sistem ini menggunakan metode *Agile* model *SCRUM*. Serta perbedaan lain yang terletak pada fitur pengelolaan penggajian dan statistik pendapatan mingguan dan bulanan, adanya pengelolaan penggajian bertujuan untuk mempermudah kasir dalam perhitungan hasil akhir gaji yang harus diberikan kepada sopir kapal, serta statistik pendapatan bertujuan untuk mengetahui detail pendapatan serta kesesuaian dari gaji yang diberikan kepada sopir kapal.

Dari penelitian penelitian diatas dirancanglah tabel perbandingan agar menjadi sebuah titik banding atau perbedaan yang jelas. Serta memiliki pengembangan yang konkrit. Berikut disajikan tabel perbandingan antara penelitian sebelumnya dengan penelitian yang sekarang dilakukan.

Tabel 2.1 Tabel Perbandingan penelitian sebelumnya dengan sekarang

No	Item	Penelitian		Perbedaan
		Sebelumnya	Sekarang	
1	Nama Peneliti	Tantri Hidayat Sinaga	Imas Nurdianto	<ul style="list-style-type: none"> • Pada tahap pengembangan metode yang digunakan berbeda • Hasil penelitian yang dihasilkan juga terdapat perbedaan
	Judul	Analisis dan Desain Sistem Pencatatan Data Manifes Kapal Penyeberangan Menggunakan Metode PIECES dan UML	Sistem Informasi Transaksi Dan Penggajian Penyeberangan Kapal Pada Dermaga Alasmalang Cilacap Berbasis Website	
	Metode	PIECES dan UML	<i>Agile model SCRUM</i>	
	Hasil Penelitian	Hasil dari penelitian ini berupa model <i>UML</i> dan <i>prototype</i> aplikasi berbasis web	Hasil dari penelitian ini berupa Website sistem informasi untuk mengelola transaksi dan rekapitulasi yang terdapat pada dermaga alasmalang	
2	Nama Peneliti	Khairul Imtihan, Muhamad Hasyim Basri	Imas Nurdianto	<ul style="list-style-type: none"> • Metode nya yang masih menggunakan SDLC • Dan platform yang digunakan
	Judul	Sistem Informasi Pembuatan Manifest Muatan Kapal	Sistem Informasi Transaksi Dan Penggajian	

		Berbasis Dekstop dan Android	Penyeberangan Kapal Pada Dermaga Alasmalang Cilacap Berbasis Website	adalah berbasis desktop dan android
	Metode	metode analisis <i>PIECES</i> serta menggunakan metode <i>SDLC</i>	<i>Agile</i> model <i>SCRUM</i>	
	Hasil Penelitian	Hasil dari penelitian adalah sistem informasi manifes berbasis <i>desktop</i> dan <i>android</i> .	Hasil dari penelitian ini berupa Website sistem informasi untuk mengelola transaksi dan rekapitulasi yang terdapat pada dermaga alasmalang	
3	Nama Peneliti	Basten Handali, Munawir, Susmanto	Imas Nurdianto	<ul style="list-style-type: none"> •Bahasa pemrograman yang masih menggunakan native belum menggunakan Framework
	Judul	Sistem Informasi Pemesanan E-Tiket Kapal Feri Pelabuhan Sinabang Kabupaten Simeulue	Sistem Informasi Transaksi Dan Penggajian Penyeberangan Kapal Pada Dermaga Alasmalang Cilacap Berbasis Website	
	Metode	(tidak tecantum dalam jurnal)	<i>Agile</i> model <i>SCRUM</i>	

	Hasil Penelitian	perancangan sistem informasi pemesanan e-tiket kapal feri Pelabuhan Sinabung Kabupaten Simeulue pada PT ASDP yang dapat memudahkan penumpang dalam memesan tiket serta melihat jadwal keberangkatan kapal	Hasil dari penelitian ini berupa Website sistem informasi untuk mengelola transaksi dan rekapitulasi yang terdapat pada dermaga alasmalang	
4	Nama Peneliti	Masriadi, Muhammad Ikhlas, Romi Wijaya	Imas Nurdianto	<ul style="list-style-type: none"> • Hasil dari penelitian yang berbasis dekstop • Dan secara bahasa pemrograman berbeda
	Judul	Sistem Informasi Penjualan Tiket Kapal Di Pelabuhan Teluk Bungus Padang Berbasis Desktop	Sistem Informasi Transaksi Dan Penggajian Penyeberangan Kapal Pada Dermaga Alasmalang Cilacap Berbasis Website	
	Metode	(tidak tecantum dalam jurnal)	<i>Agile</i> model <i>SCRUM</i>	
	Hasil Penelitian	Penelitian ini berbasis desktop dalam pengimplementasian sistemnya.	Hasil dari penelitian ini berupa Website sistem informasi untuk mengelola transaksi dan rekapitulasi	

			yang terdapat pada dermaga alasmalang	
5	Nama Peneliti	Nurul Afni, Roida Pakpahan, Astri Rezky Jumarah	Imas Nurdianto	<ul style="list-style-type: none"> • Metode perancangan yang masih menggunakan waterfall
	Judul	Rancang Bangun Sistem Informasi Penggajian Dengan Implementasi Metode Waterfall	Sistem Informasi Transaksi Dan Penggajian Penyeberangan Kapal Pada Dermaga Alasmalang Cilacap Berbasis Website	
	Metode	Waterfall	Agile model SCRUM	
	Hasil Penelitian	Hasil dari penelitian ini adalah <i>prototype</i> sistem penggajian serta <i>website</i> penggajian karyawan yang dapat memudahkan dalam proses rekap data serta laporan penggajian karyawan	Hasil dari penelitian ini berupa Website sistem informasi untuk mengelola transaksi dan rekapitulasi yang terdapat pada dermaga alasmalang	

2.2 Landasan Teori

2.2.1 Sistem Informasi

Sistem informasi adalah kumpulan subsistem yang terintegrasi dan bekerja sama untuk memecahkan masalah tertentu dengan memanipulasi data dengan alat yang disebut komputer, sehingga menambah nilai dan manfaat bagi pengguna serta digunakan

dalam pengumpulan, pemasukan, dan pemrosesan data serta penyimpanan, pengelolaan, mengontrol, dan melaporkannya untuk digunakan sebagai sarana mencapai tujuan. Jadi pada kesimpulannya sistem informasi adalah serangkaian langkah-langkah penggabungan subsistem untuk menyatukan kebutuhan suatu pengguna dengan laporan yang dibutuhkan [9].

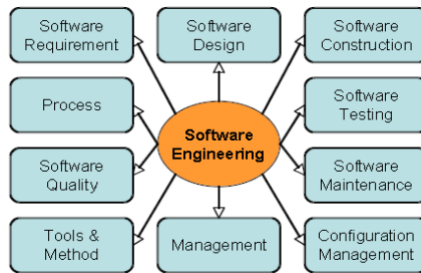
Adapun definisi sistem informasi menurut beberapa ahli adalah sebagai berikut :

1. Menurut Hutahean menyatakan bahwa sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengelolaan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategis dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang dibutuhkan [10].
2. Menurut Laudon menyatakan bahwa sistem informasi adalah komponen- komponen yang saling berhubungan dan bekerja sama untuk mengumpulkan, memproses, menyimpan dan menyebarkan informasi untuk mendukung pengambilan keputusan, koordinasi, pengendalian dan untuk memberikan gambaran aktivitas di dalam perusahaan [11].
3. Menurut John F Sistem Informasi adalah kombinasi dari manusia, fasilitas, atau alat teknologi, media, prosedur, dan pengendalian yang bermaksud menata jaringan komunikasi yang penting, proses atas transaksi-transaksi tertentu, rutin membantu manajemen, pemakai intern, ekstern serta menyediakan dasar pengambilan keputusan yang tepat [12].

2.2.2 Rekayasa Perangkat Lunak

Rekayasa perangkat lunak (*software*) adalah Ilmu yang membahas semua aspek produksi perangkat lunak, mulai dari tahap awal yaitu analisa kebutuhan pengguna, menentukan spesifikasi dari kebutuhan pengguna, desain, pengkodean, pengujian sampai memelihara sistem setelah di gunakan [13].

Ruang lingkup dari RPL dapat digambarkan sebagai berikut [13]:



Gambar 2.1 Ruang Lingkup RPL

1. *Software Requirement* merupakan kegiatan yang dilakukan untuk mengidentifikasi dan menganalisis kebutuhan perangkat lunak. Hasil akhir tahapan ini adalah spesifikasi dan model perangkat lunak.
2. *Software Design* adalah tahapan perancangan arsitektur, komponen, antar muka, dan karakteristik lain dari perangkat lunak.
3. *Software Construction* berhubungan dengan detail pengembangan perangkat lunak, termasuk algoritma, pengkodean, pengujian dan pencarian kesalahan.
4. *Software Testing* meliputi pengujian pada keseluruhan perilaku perangkat lunak.
5. *Software Maintenance* mencakup upaya-upaya perawatan ketika perangkat lunak telah dioperasikan.
6. *Software Configuration Management* berhubungan dengan usaha perubahan konfigurasi perangkat lunak untuk memenuhi kebutuhan tertentu.
7. *Software Engineering Management* berkaitan dengan pengelolaan dan pengukuran RPL, termasuk perencanaan proyek perangkat lunak.
8. *Software Engineering Tools and Methods* mencakup kajian teoritis tentang alat bantu dan metode RPL.
9. *Software Engineering Process* berhubungan dengan definisi, implementasi pengukuran, pengelolaan, perubahan dan perbaikan proses RPL.
10. *Software Quality* menitik beratkan pada kualitas dan daur hidup perangkat lunak.

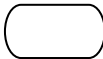
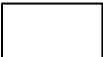
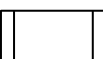
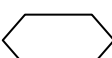
2.2.3 Pemrograman Berorientasi Objek

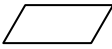
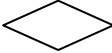
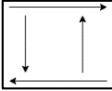
Metodologi berorientasi objek merupakan suatu cara bagaimana sistem perangkat lunak dibangun melalui pendekatan objek secara sistematis. Konsep *object oriented* atau berorientasi objek memfokuskan pada penciptaan *class* yang merupakan *blueprint* dari suatu *object*. Konsep ini dapat membagi perangkat lunak menjadi beberapa *object* yang saling berhubungan dan berinteraksi satu sama lainnya [14].

2.2.4 Flowchart




Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urutan-urutan dari suatu program. *Flowchart* adalah bentuk gambar/diagram yang mempunyai aliran satu atau dua arah secara sekuensial. Berikut Tabel 2.1 merupakan simbol-simbol yang terdapat pada *flowchart* [15]:



Tabel 2.2 Simbol-Simbol Flowchart [15]

No.	Simbol	Nama	Keterangan
1.		<i>Terminal</i>	Memulai dan mengakhiri suatu program
2.		<i>Proses</i>	Proses perhitungan atau proses pengolahan data
3.		<i>Predefined Process</i> (sub program)	Permulaan sub program atau proses pengolahan data
4.		<i>Preparation</i>	Proses inisialisasi atau pemberian harga awal

No.	Simbol	Nama	Keterangan
5.		<i>Input – Output</i>	Memasukan data maupun menunjukkan hasil dari suatu <i>process</i> tanpa tergantung dengan jenis peralatannya
6		<i>Decision</i>	Memilih proses berdasarkan kondisi yang ada.
7.		<i>Flow</i>	Menghubungkan antara simbol satu dengan simbol yang lain atau menyatakan jalannya arus dalam suatu proses. Simbol arus ini sering disebut juga dengan <i>connecting line</i> .

Tabel 2.3 Simbol-Simbol Flowchart [15]

No	Simbol	Nama	Keterangan
1.		<i>Document</i>	Merupakan simbol untuk data yang terbentuk informasi.
2.		<i>On Page Connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada satu halaman
3.		<i>Off Page Connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang

			berada pada halaman berbeda
4.		<i>Manual Operation</i>	Menunjukkan pengolahan yang tidak dilakukan oleh komputer/pc.
5.		<i>Manual Input</i>	Memasukan data secara manual <i>on-line keyboard</i> .


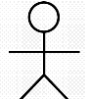




2.2.5 UML (*Unified Modeling Language*)

UML (Unified Modelling Language) merupakan alat perancangan sistem yang berorientasi pada objek. Kemunculan *UML* diilhami oleh konsep yang telah ada yaitu konsep permodelan *Object Oriented (OO)*, konsep ini menganalogikan sistem seperti kehidupan nyata yang didominasi oleh obyek dan digambarkan atau dinotasikan dalam simbol-simbol yang cukup spesifik [16]. *UML* memiliki banyak diagram, berikut penjelasan mengenai diagram yang ada di *UML*.

a. *Use Case Diagram*

Use case diagram merupakan diagram yang menjelaskan secara rinci menggunakan dokumen yaitu *use case scenario*. *Use case scenario* ini mendeskripsikan secara tekstual antara aktor dengan sistem, berbeda hal dengan *use case diagram* yang merupakan gambaran representasi dari sistem yang dikembangkan. *Use case diagram* digambarkan dalam visualisasi hubungan antara aktor dengan sistem. Elemen-elemen pada *use case diagram* yaitu aktor, *use case*, asosiasi, *include*, *extend* dan hubungan generalisasi [17]. Berikut tabel 2.3 adalah simbol-simbol yang ada pada diagram *use case diagram* :







Tabel 2.4 Simbol-Simbol *Use Case Diagram* [18]

No.	Simbol	Nama	Keterangan
1.		<i>Use case</i>	Deskripsi dari urutan aksi – aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .
2.		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
3.		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
4.		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara eksplisit.
5.		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6.		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas

b. Sequence Diagram

Sequence diagram digunakan untuk menggambarkan interaksi antara sejumlah objek dalam urutan waktu. Banyaknya diagram sekuen yang akan dibangun sesuai dengan pendefinisian *usecase* yang memiliki proses sendiri [19]. Berikut merupakan tabel 2.4 adalah simbol-simbol yang ada pada *Sequence Diagram* :

Tabel 2.5 Simbol -Simbol *Sequence Diagram* [18]

No.	Simbol	Nama	Keterangan
1.		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2.		<i>Actor</i>	Menggambarkan <i>user</i> atau pengguna.
3.		<i>Message</i>	Spesifikasi dari komunikasi antar <i>objek</i> yang memuat informasi – informasi tentang aktivitas yang terjadi.
4.		<i>Boundary</i>	Menggambarkan sebuah <i>form</i> .
5.		<i>Control Class</i>	Menghubungkan <i>boundary</i> dengan Tabel.
6.		<i>Entity Class</i>	Menggambarkan hubungan kegiatan yang akan dilakukan.

2.2.6 Basis Data (*Database*)

Database merupakan kumpulan informasi yang disimpan secara sistematis di dalam komputer sehingga dapat dikendalikan oleh program komputer untuk mengambil informasi dari *database*. *Database* adalah sistem penyimpanan yang dapat menyimpan kumpulan informasi terkait yang dibutuhkan sistem sehingga mudah untuk diakses [20]. *Database* mewakili entitas, atribut, dan hubungan logis antara entitas. Basis data terdiri dari kumpulan data yang terorganisir, relasi antar data, dan objektifnya.

a. *DBMS (Database Management System)*

Database management System (DBMS). *DBMS* adalah kumpulan program yang digunakan untuk mendefinisikan, mengatur, dan memproses *database* [21]. *DBMS* menyediakan berbagai fasilitas yaitu [22]:

1. *DDL (Data Definition Language)* merupakan perintah-perintah yang biasa digunakan administrator *database* untuk mendefinisikan skema dan subskema *database*. Perintah yang termasuk di dalamnya yaitu :
 - a. *CREATE* : Digunakan untuk membuat termasuk diantaranya membuat *database* dan tabel baru.
 - b. *ALTER* : Digunakan untuk mengubah struktur tabel yang ada.
 - c. *DROP* : Digunakan untuk menghapus *database* dan tabel.
2. *DML (Data Manipulation Language)* merupakan perintah-perintah yang memungkinkan pengguna melakukan akses dan manipulasi data sebagaimana yang telah diorganisasikan sebelumnya dalam model data yang tepat, *DML* digunakan untuk memanipulasi *database* yang telah didefinisikan dengan *DDL*. Perintah yang termasuk *DML* adalah sebagai berikut :
 - a. *INSERT* : Digunakan untuk menyisipkan atau memasukkan data dalam tabel.
 - b. *SELECT* : Digunakan untuk mengambil data atau menampilkan data dari suatu tabel atau beberapa tabel.

- c. *UPDATE* : Digunakan untuk memperbarui data lama menjadi data terkini.
- d. *DELETE* : Digunakan untuk menghapus data dari tabel.

b. SQL (Structured Query Language)

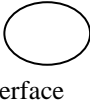

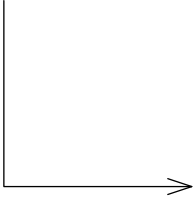
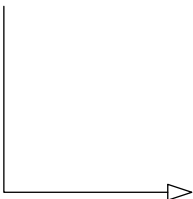
SQL (Structured Query Language) merupakan salah satu produk RDBMS. RDBMS memiliki kepanjangan *Relational Database Management System*. Merupakan salah satu produk andalan yang dibuat oleh Microsoft yang berfungsi sebagai *reational database*. yaitu hubungan antar table yang berisi data data pada suatu *database* [23].

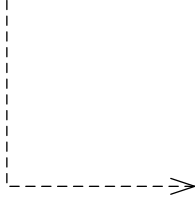
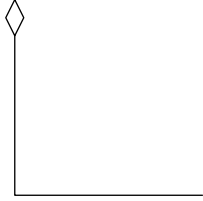
c. Class Diagram

Class diagram memperlihatkan struktur sistem dalam hal pendefinisian *class-class* yang akan digunakan untuk membangun sistem. *Class* memiliki atribut dan metode, yang juga disebut sebagai operasi. Dalam tabel 2.5 di bawah ini, terdapat simbol-simbol yang digunakan dalam *class diagram* [24].

Tabel 2.6 Simbol-Simbol Class Diagram [24]

No	Simbol	Keterangan
1.	<div style="border: 1px solid black; padding: 10px; width: fit-content; margin: auto;"> <div style="border-bottom: 1px solid black; padding-bottom: 5px; text-align: center;">Nama_kelas</div> <div style="border-bottom: 1px solid black; padding: 5px 0 5px 20px; text-align: center;">+atribut</div> <div style="padding: 5px 0 5px 20px; text-align: center;">+operasi()</div> </div>	Kelas pada struktur sistem.

2.	Antar muka / <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3.	Asosiasi / <i>association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4.	Asosiasi berarah / <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5.	Generalisasi 	Relasi antar kelas dengan makna generalisasi- generalisasi- spesialisasi (umum khusus).

6.	Kebergantungan <i>dependency</i> 	/ Relasi antar kelas dengan makna kebergantungan antar kelas.
7.	Agregasi / <i>aggregation</i> 	Relasi antar kelas dengan makna semua bagian (<i>whole part</i>).

2.2.7 Website

Secara terminologi, web atau *website* adalah kumpulan dari halaman situs dan dokumen yang tersebar di beberapa komputer server yang berada di seluruh penjuru dunia dan terhubung menjadi satu jaringan melalui jaringan yang disebut internet. *website* atau disingkat web, dapat diartikan sekumpulan halaman yang terdiri dari atas beberapa laman yang berisi informasi dalam bentuk data digital, audio, dan animasi lainnya yang disediakan melalui jalur koneksi internet [25].

2.2.8 PHP

PHP (Hypertext Preprocessor) merupakan bahasa pemrograman yang digunakan secara luas untuk pembuatan dan pengembangan sebuah web dan biasa digunakan pada HTML. PHP adalah bahasa yang biasanya digunakan untuk komunikasi sisi server

dan digunakan oleh seorang programmer untuk mengembangkan sebuah *website* yang dinamis dan interaktif [26].

2.2.9 Framework

Framework merupakan sebuah *software* atau aplikasi yang bisa dibilang seperti kerangka kerja yang fungsinya untuk memudahkan *developer* dalam mengembangkan aplikasi *website* yang ada. Ada beragam jenis *framework* yang sering digunakan untuk membuat *website*. Setiap *framework* memiliki fungsinya masing-masing dan menggunakan bahasa pemrograman yang berbeda [27].

a. Laravel

Laravel adalah *framework* PHP yang dirilis di bawah lisensi MIT, berdasarkan konsep MVC (*Model View Controller*). *Laravel* adalah pengembangan web berbasis MVP yang ditulis dalam PHP. Struktur pola MVC pada *laravel* sedikit berbeda pada struktur pola MVC pada umumnya. Di *laravel* terdapat *routing* yang berfungsi sebagai penghubung antara *request* dari *user* dan *controller* [28].

2.2.10 Angkutan Penyeberangan

Angkutan penyeberangan adalah angkutan yang berfungsi sebagai jembatan yang menghubungkan jaringan jalan untuk mengangkut penumpang dan kendaraan beserta muatannya [29]. Angkutan penyeberangan memungkinkan orang dan kendaraan untuk mencapai tujuan mereka dengan melintasi perairan yang tidak dapat dilalui dengan cara lain. Hal ini membantu meningkatkan konektivitas antara pulau-pulau, wilayah pesisir, atau area terpencil dengan daratan utama atau pusat transportasi lainnya. Angkutan penyeberangan juga berperan penting dalam memfasilitasi industri pariwisata. Banyak destinasi wisata terkenal terletak di pulau-pulau atau wilayah pesisir yang memerlukan transportasi melintasi perairan. Dengan adanya angkutan penyeberangan yang handal, wisatawan dapat dengan mudah mengakses tempat-tempat tujuan tersebut, sehingga mendorong pertumbuhan ekonomi di sektor pariwisata. Angkutan penyeberangan berkaitan dengan sebuah titik pemberhentian yang disebut dermaga. Sedangkan Dermaga memiliki definisi konstruksi yang terletak di pelabuhan yang berfungsi sebagai tempat kapal bersandar, melakukan kegiatan pemuatan dan pembongkaran barang, serta menaik-turunkan penumpang [30]. Selain itu, dermaga juga berperan dalam keamanan dan keselamatan

angkutan penyeberangan. Dengan menyediakan platform yang tidak mudah rusak dan fasilitas seperti tangga, jembatan, dan pijakan yang tepat, dermaga membantu mengurangi risiko cedera atau kecelakaan saat penumpang dan muatan berpindah antara kapal dan darat.