

BAB II

LANDASAN TEORI

2.1 Tinjauan Penelitian Terdahulu

Penulis melihat hasil dari penelitian-penelitian yang terdahulu dan mempelajari penelitian yang sudah dilakukan yang terkait dengan penelitian penulis agar dapat di gunakan sebagai penunjuk dan pertimbangan pada penelitian juga hasil yang dicapai pada penelitian ini.

Penelitian yang dibuat oleh Putu Agus Pradana Adi Putra (2020) yang berjudul “Rancang Bangun Sistem Informasi Penyewaan Kendaraan (Sewadisini.com) Berbasis Web”. Penelitian ini mengulas tentang sistem informasi rental mobil menggunakan sistem informasi Website. Pada sistem informasi ini penyewa harus mempunyai akun pada web sewadisini.com dengan cara mendaftar pada halaman *register*. Setelah itu pengguna mendaftarkan diri dengan mengisi data diri dihalaman tersebut, jika sudah mendaftar dan mempunyai akun pengguna login pada halaman ini dengan menginput *username*, *password* dan kode *captcha*. Setelah *login* maka sistem menampilkan Halaman Beranda, dalam halaman pengguna dapat mengetahui beberapa kendaraan yang dipromosikan dan penyewa yang akan mencari kendaraan yang diinginkan pada kolom pencarian bisa menginput nama dan lokasi setelah itu hasil pencarian muncul gambar kendaraan beserta detail kendaraan. Selain itu terdapat juga *link rating* untuk melihat *rating* dari penyedia jasa. Setelah membooking dan menggenapi rincian pemesanan di form *booking* maka sistem akan mengirimkan

rincian *booking* yang sudah terkirim ke email penyewa. Setelah penyewa menggenapi, penyewa menuju ke halaman pembayaran. Di tampilan rincian penyewaan yang telah di isi terdapat tombol mengunggah tanda pembayaran. lalu penyewa mengunggah tanda pembayaran, lalu sistem akan menyampaikan pesan pemberitahuan bahwa pembayaran akan segera dikonfirmasi. Setelah dikonfirmasi tampil halaman daftar kendaraan yang dipromosikan oleh pemilik di *sewadisini.com*. Penyewa dapat menambah, merubah rincian kendaraan yang dipromosikan serta menerbitkan *invoice* promosi kendaraan.

Sistem informasi rental kendaraan (*sewadisini.com*) ini dibuat dengan memanfaatkan *framework* codeigniter sama dengan rancangan yang dilakukan sebelumnya. Percobaan sistem dilakukan memakai metode *blackbox* beserta sistem sudah beroperasi sesuai dengan fungsi yang sudah dirancang. Namun Pada sistem informasi ini tidak ada tata cara untuk memesan mobil yang akan dirental dan tidak ada menu layanan menghubungi perusahaan (bantuan/masalah).

Penelitian selanjutnya dibuat oleh Johan Wahyudi (2019) yang berjudul “Pembangunan Sistem Aplikasi Penyewaan Mobil Berbasis Android”. Penelitian ini mengulas tentang sistem informasi rental kendaraan menggunakan sistem informasi rental kendaraan dengan memanfaatkan sistem operasi Android. Pada saat pengguna akan memasuki kehalaman utama aplikasi, pengguna harus login terlebih dahulu lalu mengisi email dan *password*. Setelah login berhasil pengguna dapat melihat tampilan daftar rental mobil dan dimenu ini pengguna dapat melihat rincian pemesanan. Lalu pengguna memilih kendaraan yang mau disewa, setelah itu pengguna akan mengisi formulir penyewaan, di formulir ini pengguna akan

mengisi tanggal penyewaan, jangka penyewaan, memilih meyewa sama sopirnya atau tidak, dan mengisi alamat ambil yang bertujuan untuk bertemu dan menyerahkan mobil sebagai alamat pengembalian mobil.

Aplikasi Rental kendaraan ini dapat membantu penyewa di CV. Amanah Kalimantan Rent *Car* untuk transaksi penyewaan mobil dan mengetahui seluk beluk tentang kendaraan dengan internet tanpa harus pergi ke rental mobil. Aplikasi ini dapat memberi informasi kepada penyewa dan pengelola dapat mengelompokkan data mobil, penyewa dan supir dengan mudah sehingga proses perentalan lebih cepat dan efisien. Namun memiliki kelemahan yaitu tidak ada tata cara penggunaan aplikasi dalam pemesanan pembayaran, harus mendownload apk terlebih dahulu dan tidak ada menu layanan menghubungi perusahaan (bantuan/masalah).

Penelitian selanjutnya dilakukan oleh Derian Pratama (2019) yang berjudul “Rancang Bangun Sistem Informasi Penyewaan Kendaraan Berbasis Web”. Penelitian ini mengulas tentang rental kendaraan berbasis web dimana pengguna yang akan melakukan penyewaan diharuskan melakukan pendaftaran akun terlebih dahulu sebelum memesan, dan masuk keweb tersebut dengan cara memasukan nama dan sandi yang sebelumnya sudah mendaftarkan diri untuk memulai memesan kendaraan, lalu pengguna dapat memilih kendaraan yang akan dipesan dengan mengisi *form* penyewaan yang sudah tersedia lalu mengkonfirmasi pembayaran di dalam aplikasi. Dan untuk admin dapat melakukan pengecekan penyewaan dan konfirmasi atas pembayaran yang sudah diajukan oleh pelanggan.

Diuraikan diatas dapat disimpulkan bahwa aplikasi yang akan dibangun dengan yang ada dalam penelitian terkait terdapat perbedaan dan juga ada kesamaan. Persamaannya adalah tidak ada menu menghubungi perusahaan (bantuan/masalah), kedua tidak ada tata cara untuk memesan dalam aplikasi.

Tabel 2.1 Rangkuman Hasil Penelitian Terdahulu

No	Peneliti	Judul Aplikasi	Kelebihan Aplikasi
1.	Johan Wahyudi (2019)	Pembangunan Sistem Aplikasi Penyewaan Mobil Berbasis Android	- Lebih simple untuk memesan kembali
2.	Putu Agus Pradana (2020)	Rancang Bangun Sistem Informasi Penyewaan Kendaraan(Sewadisin.com) Berbasis Website	- Terdapat kode <i>captcha</i>
3.	Derian Pratama (2019)	Rancang Bangun Sistem Informasi Penyewaan Kendaraan (Sewadisin.com) Berbasis Website	- Tedapat laporan penyewaan untuk pelanggan maupun admin
4.	Peneliti Saat Ini (2020)	Rancang Bangun Sistem Informasi <i>Booking Jeep</i> Berbasis Web Di Kabupaten Probolinggo menggunakan <i>framework</i> <i>codeigniter</i>	- Ada cara untuk memesan - Ada menu pelayanan kusus melewati whatsapp bisnis

2.2 Pengertian Sistem Informasi

Sistem yaitu jaringan kerja dalam prosedur-prosedur yang satu sama lain berhubungan, kelompok untuk melakukan suatu aksi atau untuk menyelesaikan suatu tujuan tertentu.

Sistem informasi yaitu sarana untuk menampilkan informasi sedemikian rupa agar bermanfaat bagi penerimanya. Bertujuan untuk menampilkan informasi untuk mengambil keputusan pada persiapan, pengorganisasian, mengedalikan kegiatan. Suatu sistem mempunyai cirri-ciri tertentu. (Ardiant, 2018).

2.3 Pengertian Penyewaan/Rental

sewa yaitu menggunakan objek dengan membayar, uang yang dibayar karena menggunakan atau meminjam sesuatu, yang dapat dipakai dengan membayar uang dengan uang.

Sewa-menyewa pada pasal 1548 sampai pasal 1600 KUH Perdata sewa menyewa yaitu suatu persetujuan dimana sisi yang satu berkomitmen untuk memberi kepuasan suatu barang kepada sisi yang lain selama durasi tertetu, dengan membayarkan suatu harga yang disanggupi oleh pihak terakhir itu (Pasal 1548 KUH Perdata) (Johan Wahyudi, 2019).

2.4 Website

Website yaitu sebuah sistem dengan pemberitahuan yang ditampilkan dalam bentuk teks, gambar, suara dan lainnya yang tersimpan di *server* Web. Internet yang ditampilkan dalam bentuk *hiperteks*. Isi dari Web dalam berbentuk teks umumnya dicatat dalam bentuk HTML (*Hypertext Markup Language*). Tampilan lainnya ditampilkan dalam bentuk ilustrasi (bentuk GIF, JPG, PNG),

suara (dalam *bentuk* AU, WAV), dan objek *multimedia* lainnya (seperti MIDI, *Shockwave, Quicktime Movie, 3D World*)(Putu Agus Pradana Adi Putra).

2.5 MySql

MySQL yaitu suatu *Database Management System* (DBMS). MySQL beroperasi untuk mengelola *database* memanfaatkan bahasa SQL. MySQL bersifat terbuka sehingga kita dapat memakainya secara gratis. bahasa PHP juga sangat mendukung dengan MySQL.

2.6 Codeigniter

CodeIgniter yaitu kerangka kerja aplikasi web yang bersifat terbuka untuk bahasa pemrograman PHP. CodeIgniter mempunyai banyak fitur yang berbeda dengan kerangka kerja lainnya. Tidak sama dengan kerangka kerja PHP lainnya, dokumentasi untuk kerangka kerja ini begitu lengkap, mencakup semua aspek dikerangka kerja. CodeIgniter juga mampu berkerja dilingkungan *shared hosting* karena mempunyai ukuran yang sangat kecil, tetapi mempunyai kinerja yang begitu luar biasa. Dari sisi pemrograman, CodeIgniter cocok dengan PHP4 dan PHP5, sehingga akan berkerja dengan baik pada web *host* yang banyak digunakan saat ini. CodeIgniter menggunakan pola (MVC), berupa cara untuk mengoprasian aplikasi web di 3 bagian yang berbeda, yaitu Model merupakan lapisan abstraksi penyimpanan data, *Views* berisikan file-file tema tampilan depan, dan *Controller* merupakan logika dan proses dari aplikasi. Pada intinya, CodeIgniter juga membuat pemakai ekstensif dari pola desain *Single ton*. Maksudnya adalah cara untuk meload *class* sehingga jika *class* itu disebut dalam beberapa kali, terjadi kesamaan pada *class* tersebut akan digunakan kembali. Hal berguna di koneksi

penyimpanan data, karena kita menginginkan menggunakan satu koneksi setiap kali *class* itu digunakan.

2.7 Framework

Framework merupakan kumpulan kelas yang menandakan bentuk abstrak dalam pemecahan beberapa masalah yang berhubungan. *Framework* yaitu kelompok perintah atau tugas dasar yang membuat aturan-aturan tertentu dan saling berinteraksi dengan lainnya sehingga saat membuat aplikasi web kita harus mengikuti aturan dari kerangka kerja tersebut. Dalam hal ini *framework* PHP kita tidak perlu berfikir tentang kode perintah dasar dari aplikasi website kita. Pada pemrograman web, kerangka kerja telah dikembangkan untuk bahasa pemrograman antara lain PHP dan Java. Untuk PHP, kerangka kerja yang banyak dimanfaatkan yaitu *Zend framework*, *CodeIgniter* dan *Symfony Framework*.

2.8 FlowChart

Flowchart yaitu menggambar secara grafik dengan langkah yang berurutan dengan prosedur dalam program. *Flowchart* membantu untuk menemukan masalah ke dalam bagian yang lebih kecil dalam menganalisis alternatif-alternatif lainnya dalam pengoperasian dan mempermudah menyelesaikan suatu masalah, khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut.

Ada beberapa petunjuk yang harus diperhatikan, seperti :

1. *Flowchart* diilustrasikan dari halaman atas ke bawah dan kiri kekanan.
2. Aktifitas yang diilustrasikan harus diterangkan secara hati-hati dan keterangan ini harus dimengerti oleh pembacanya.

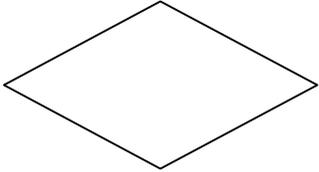
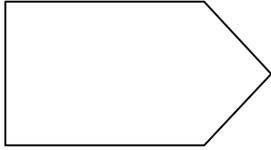
3. Kapan aktifitas dimulai dan berakhir harus ditentukan secara jelas.
4. Setiap langkah dari kegiatan harus dijabarkan dengan memakai deksripsi kata kerja.
5. Setiap langkah dari kegiatan harus berada di urutan yang benar.
6. Lingkup, *range* dan kegiatan yang sedang diilustrasikan harus diamati dengan hati-hati. Percabangan-percabangan yang memotong kegiatan yang sedang diilustrasikan tidak harus dipakai dan percabangannya disimpan pada halaman yang terpisah atau hapus semuanya bila percabangan tidak berkaitan dengan sistem.
7. Pakai lambang *flowchart* yang standar.

Berikut ini adalah gambar lambang standar dalam *flowchart* beserta dengan arti dari masing – masing lambing :

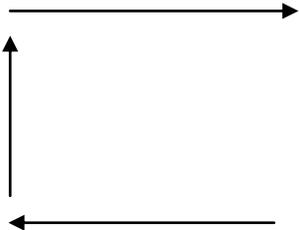
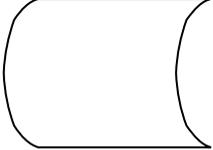
Tabel 2.2 Simbol – Simbol *FlowChart*

Simbol	Keterangan
 <p data-bbox="427 1570 660 1693"><i>Terminal/Interrupt</i> (Mulai/Berhenti)</p>	<p data-bbox="759 1547 1348 1727">Simbol ini dipergunakan untuk menunjukkan awal kegiatan atau akhir kegiatan atau berhentinya suatu program.</p>

Tabel 2.2 (Lanjutan)

Simbol	Keterangan
 <p data-bbox="395 701 687 734"><i>Process</i> (Pengolahan)</p>	<p data-bbox="751 566 1356 674">Suatu simbol yang melambangkan diprosesnya suatu data.</p>
 <p data-bbox="395 1014 687 1048"><i>Decision</i> (Keputusan)</p>	<p data-bbox="751 891 1356 999">Dipakai untuk menuliskan jika adanya percabangan, seperti <i>if</i>, <i>case</i>.</p>
 <p data-bbox="448 1491 635 1525"><i>Prefendefined</i></p>	<p data-bbox="751 1155 1356 1559">Untuk program-program yang sering dipergunakan sebuah program berulang kali, biasanya dibuat program terpisah dengan sebuah sub program (<i>subroutine</i>). Untuk menghubungkan program utama dengan <i>subroutine</i> dipergunakan simbol ini.</p>
 <p data-bbox="376 1839 703 1872"><i>Connector</i> (Penghubung)</p>	<p data-bbox="751 1648 1356 1906">Bila suatu <i>flowchart</i> sangat panjang dan diputus di tengah sebelum selesai, jika disambung dalam halaman yang sama lagi, maka digunakan simbol ini.</p>

Tabel 2.2 (Lanjutan)

Simbol	Keterangan
 <p data-bbox="376 846 667 880"><i>Flowlines (Garis Alir)</i></p>	<p data-bbox="715 483 1359 880">Bila suatu <i>flowchart</i> dihubungkan dengan garis-garis ini. Garis-garis ini menunjukkan akar selanjutnya yang akan dituju. Bila arahnya ke bawah atau ke kanan tidak perlu memakai tanda panah. Bila ke atas atau ke kiri, tanda panah harus dipakai, untuk membedakannya.</p>
 <p data-bbox="451 1070 587 1104"><i>Data base</i></p>	<p data-bbox="715 958 1359 1059">Digunakan untuk menunjukkan penyimpanan data.</p>
 <p data-bbox="432 1361 603 1395"><i>Data Storage</i></p>	<p data-bbox="715 1216 1359 1328">Digunakan untuk menunjukkan penyimpanan data dengan berbagai format.</p>
 <p data-bbox="440 1585 603 1619"><i>Input-output</i></p>	<p data-bbox="715 1440 1359 1619">Digunakan untuk menyatakan proses <i>input</i> dan <i>output</i> tanpa tergantung dengan jenis peralatannya.</p>

2.9 Unified Model Language (UML)

Unified Modeling Language yaitu bahasa perincian standar untuk pengumpulan bukti, perancangan, dan membangun sistem perangkat lunak. UML

juga merupakan suatu cara agar memudahkan pengembang aplikasi yang berkelanjutan. sistem yang tidak mengumpulkan bukti juga dapat menghambat pengembangan dikarenakan pembuat harus membuat pelacakan dan mempelajari kode program. UML juga bisa berperan sebagai alat bantu untuk mengirim ilmu yang berhubungan dengan sistem atau aplikasi yang akan dikembangkan dari satu pembuat ke pembuat lainnya. Tidak hanya antara pembuat dengan pembisnis dan siapapun juga bisa mengerti sebuah sistem dengan adanya UML.

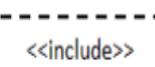
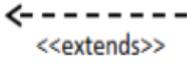
Perlengkapan yang dipakai dalam perancangan berbasis UML adalah sebagai berikut:

- a. *Use Case Diagram* yaitu contoh untuk kelakuan perilaku sistem pemberitahuan yang akan dibuat. *Use case* dipakai agar mengetahui semua tugas yang terdapat pada sistem pemberitahuan dan siapa saja yang bisa memakai tugas tersebut. lambang yang dipakai di *use case diagram* yaitu:
(Hendini. 2016).

Tabel 2.3 *Use Case Diagram*

Simbol	Keterangan
	<p><i>Use Case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktif, yang dinyatakan dengan menggunakan kata kerja.</p>

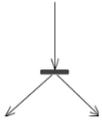
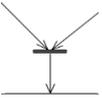
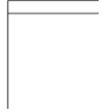
Tabel 2.3 (Lanjutan)

Simbol	Keterangan
	<p>Aktor adalah <i>Abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem.</p>
	<p>Asosiasi antara aktor dan <i>usecase</i>, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan data.</p>
	<p>Asosiasi antara aktor dan usecase yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.</p>
	<p><i>Include</i>, merupakan pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.</p>
	<p><i>Extend</i> merupakan perluasan dari <i>usecase</i> lain jika kondisi atau syarat terpenuhi.</p>

- b. Diagram Aktivitas yaitu mengilustrasikan sebuah aliran kerja atau kegiatan dari suatu sistem atau proses bisnis.

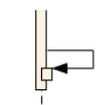
simbol yang dipakai dalam diagram aktifitas yaitu: (Hendini. 2016).

Tabel 2.4 *Activity Diagram*

Gambar	Keterangan
	<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktivitas.
	<i>End Point</i> , akhir aktivitas.
	<i>Activities</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> /percabangan, digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	Join (penggabungan) atau <i>rake</i> , digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i> , menggambar kan pilihan untuk pengambilan keputusan, <i>true</i> atau <i>false</i> .
	<i>Swimlane</i> , pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa.

- c. Diagram urutan yaitu mengilustrasikan perilaku objek kepada *usecase* dengan mendefinisikan waktu hidup objek dan pesan yang dikirimkan dan diterima antar objek. Lamabang yang dipakai di *Sequence Diagram* yaitu: (Hendini. 2016).

Tabel 2.5 *Sequence Diagram*

Gambar	Keterangan
	<p><i>Entity Class</i>, merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.</p>
	<p><i>Boundary Class</i>, berisi kumpulan kelas yang menjadi interfaces atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan <i>formentry</i> dan <i>form cetak</i></p>
	<p><i>Control class</i>, suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas.</p>
	<p><i>Message</i>, simbol mengirim pesan antar<i>class</i>.</p>
	<p><i>Recursive</i>, menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.</p>
	<p><i>Activation</i>, mewakili sebuah <i>eksekusi</i> operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivasi sebuah operasi.</p>
	<p><i>Lifeline</i>, garis titik-titik yang terhubung dengan objek, sepanjang lifelineter dapat <i>activation</i>.</p>