

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Penelitian sistem informasi yang berhubungan dengan rental mobil telah banyak dilakukan oleh peneliti-peneliti sebelumnya salah satunya dengan judul “Sistem Informasi Manajemen Rental pada Well Trans Transport Service Yogyakarta” [1]. Penelitian sebelumnya adalah “Sistem Informasi Berbasis Web pada Car Rental Prabu Transport Yogyakarta” [2].

Yosha Zein Immawan (2012) melakukan penelitian dengan judul “Sistem Informasi Rental Mobil pada WellTrans Transportation Service Yogyakarta”. pengolahan data yang tersusun secara sistematis merupakan media yang tepat untuk mencapai keberhasilan suatu perusahaan, karena kemampuan yang tinggi dalam mencapai ketelitian, kecepatan proses yang mendukung efisiensi waktu dan tenaga. Perencanaan seperti ini diharapkan mampu mendukung sistem sewa yang lebih efektif untuk mempermudah pencatatan dalam transaksi sewa mobil. Melihat kenyataan ini maka penulis mencoba untuk memberikan solusi yang tepat untuk kelancaran sistem persewaan secara optimal, dengan mengembangkan sistem informasi yang telah ada dengan bahasa pemrograman Visual Basic 6.0 dan Microsoft SQL Server 2010 sebagai *databasenya*.

Pada penelitian Washlina Marisha Sarah (2012) yang berjudul “Sistem Informasi Berbasis *Web* pada Car rental Prabu Transport Yogyakarta” dengan latar belakang masalah yaitu: selama ini proses transaksi pemesanan sewa mobil dilakukan oleh pelanggan yang bersangkutan datang langsung ke tempat rental atau jika mereka sudah mempunyai nomor telepon, mereka menggunakan media telepon. Permasalahan ini muncul ketika pemesanan sewa mobil yang menggunakan telepon atau datang langsung ke tempat rental akan membutuhkan waktu yang cukup lama untuk memperoleh informasi mengenai ketersediaan mobil yang belum disewa. Selain itu permasalahan muncul pada proses pencatatan pemesanan sewa mobil yang

memakan banyak waktu karena masih menginput data pemesan melalui aplikasi *desktop*. Serta pemesanan melalui telepon yang tidak memuaskan pelanggan karena tidak bisa melihat mobil yang akan disewa. Untuk menanggulangi masalah tersebut maka dibangun sebuah aplikasi sistem pemesanan dan pembayaran sewa mobil secara online berbasis *web*. Dengan *web* ini maka pemesanan mobil akan lebih terstruktur dengan baik dan dapat melakukan pembayaran melalui transfer rekening

Diantara penelitian tersebut terdapat perbedaan. Penelitian yang pertama memfokuskan pada kemudahan dalam transaksi pelayanan terhadap pelanggan, pengelolaan data transaksi yang dari manual menjadi terkomputerisasi, sehingga lebih menghemat waktu. *Output* dari pengelolaan data tersebut juga menjadi acuan pengelola rental mengambil keputusan penyewaan. Transaksi menjadi lebih aman dan meminimalkan kesalahan dalam pencatatan transaksi dan mempercepat dalam pembuatan laporan keuangan sewa. Bahasa pemrograman yang digunakan yaitu *Visual Basic* dan *SQL Server 2010* sebagai *databasenya*.

Pada penelitian kedua memfokuskan pada mempercepat dan mempermudah dalam penyewaan mobil karena sistemnya menggunakan *web* sehingga pelanggan dapat menyewa mobil dimana saja karena sistem *web* ini dapat digunakan tidak hanya di tempat sewa, *web* ini dapat diakses oleh publik. Pelanggan dapat terlebih dulu memesan mobil yang diinginkan sesuai dengan ketersediaan mobil yang ada pada rental mobil karena di *web* tersebut menampilkan mobil apa saja yang tersedia dan harga yang sudah tercantum. Perancangannya menggunakan *Macromedia Dreamweaver MX 2004*, *Adobe Photoshop CS*, dan *Xampp*

Pada penelitian ini, peneliti mengambil judul "Sistem Informasi Manajemen Rental Mobil pada Nandi Trans Yogyakarta". Rental mobil merupakan salah satu bisnis yang menguntungkan dan sangat berhubungan dengan jasa karena banyaknya orang yang menginginkan cara yang cepat dan aman untuk bepergian. Guna untuk mengutamakan pelayanan kepada pelanggan maka ketepatan dan kecepatan dalam proses sewa harus lebih ditingkatkan untuk memperlihatkan efisien dan efektifitas usahanya. Untuk itu peneliti membuat perbedaan dengan kedua penelitian tersebut

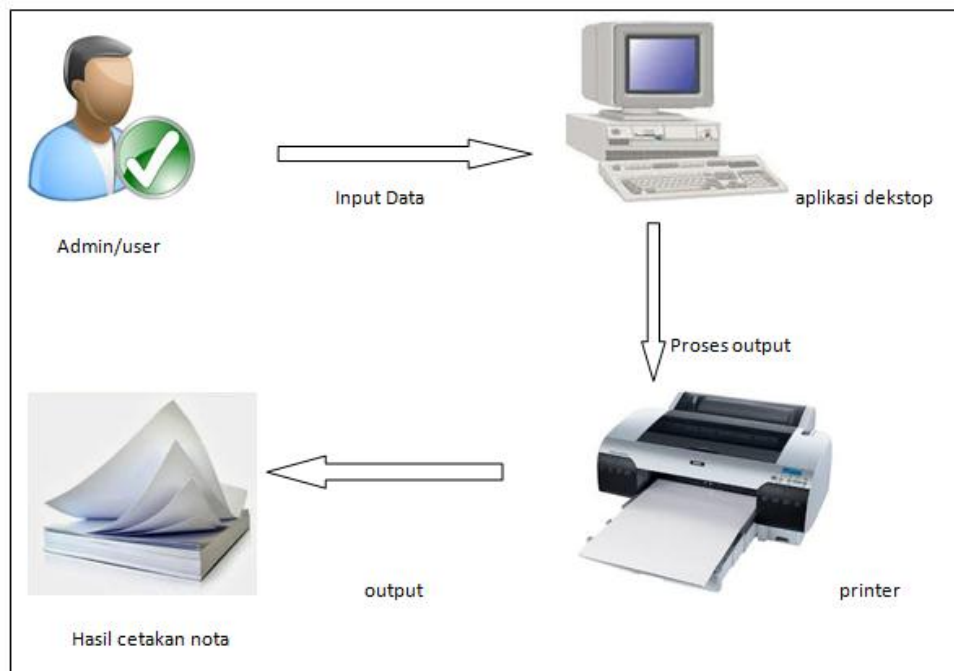
dari segi bahasa pemrograman *Visual Studio C# 2010* dan *MySQL* sebagai *databasenya* yang bertujuan untuk memberikan kemudahan bagi Nandi Trans Yogyakarta dalam menjalankan usaha rental mobil sehingga dapat menghemat biaya, waktu dan tenaga , dapat juga meminimalkan kesalahan dalam pencatatan transaksi penyewaan serta memberikan kepuasan bagi pelanggan.

2.2 Landasan Teori

2.2.1. Arsitektur

Arsitektur merupakan wujud hasil penerapan pengetahuan, ilmu, teknologi, dan seni secara utuh dalam mengubah ruang dan lingkungan binaan, sebagaibagian dari kebudayaan dan peradaban manusia, sehingga dapat menyatu dengan keseluruhan lingkungan ruang dari tingkat makro sampai dengan tingkat mikro.

Untuk membangun sebuah sistem, diperlukan rancangan penyusun sistem tersebut. Karena itu arsitektur ini bertujuan untuk menggambarkan bagaimana sistem ini akan dibangun dan dijalankan. Arsitektur *software* dapat dilihat pada Gambar 2.1



Gambar 2.1 : Arsitektur *Software*

2.2.2. Unsur Sewa Menyewa

Pada dasarnya sewa menyewa dilakukan untuk waktu tertentu, sedangkan sewa-menyewa tanpa waktu tertentu tidak diperkenankan. Persewaan tidak berakhir dengan meninggalnya orang yang menyewakan atau penyewa. Begitu juga karena barang yang disewakan dipindahtangankan. Disini berlaku asas jual beli tidak memutuskan sewa menyewa (Salim, dkk, 2006).

Dari uraian di atas, dapatlah dikemukakan unsur-unsur yang tercantum dalam perjanjian sewa-menyewa adalah :

1. Adanya pihak yang menyewa dan pihak penyewa.
2. Adanya konsensus antara kedua belah pihak yang melakukan sewa
3. Adanya objek sewa-menyewa, yaitu barang, baik barang bergerak maupun tidak bergerak
4. Adanya kewajiban dari pihak yang menyewakan untuk menyerahkan kemikmatan kepada pihak penyewa atas suatu benda.
5. Adanya kewajiban dari penyewa untuk menyerahkan uang pembayaran kepada yang menyewakan

2.2.3. Subyek dan Objek Sewa-Menyewa

Pihak yang terlibat dalam perjanjian sewa menyewa adalah pihak yang menyewakan dan pihak penyewa. Pihak yang menyewakan adalah orang atau badan hukum yang menyewakan barang atau benda kepada pihak penyewa, sedangkan pihak penyewa adalah orang atau badan hukum yang menyewa barang atau benda dari pihak yang menyewakan. Yang menjadi objek dalam perjanjian sewa-menyewa adalah barang dan harga. Dengan syarat yang disewakan adalah barang yang halal, artinya tidak bertentangan dengan undang-undang, ketertiban, dan kesusilaan.

2.3 Unified Modeling Language (UML)

Unified Modelling Language (UML) adalah sebuah "bahasa" yang telah menjadi standar dalam industri untuk *visualisasi*, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. Dengan menggunakan UML kita dapat membuat model untuk semua jenis aplikasi piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena *UML* juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka ia lebih cocok untuk penulisan piranti lunak dalam bahasa-bahasa berorientasi objek seperti *C++*, *Java*, *C#* atau *VB.NET*. Walaupun demikian, *UML* tetap dapat digunakan untuk modeling aplikasi prosedural dalam *VB* atau *C*.






2.3.1. Bangunan Dasar UML

1. Use-Case Diagram

Use case diagram merupakan diagram yang menggambarkan interaksi antara sistem dengan sistem eksternal dan pengguna. Secara grafis menggambarkan siapa yang akan menggunakan sistem dan dengan cara apa pengguna mengharapkan untuk berinteraksi dengan sistem.

Tabel 2.1 : *Use Case Diagram*

| Gambar | Keterangan |
|-----------------|---|
| <i>Use Case</i> | Fungsionalitas yang disediakan sistem sebagai unit-unit saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja diawal <i>frase</i> nama <i>use case</i> |
| <i>Actor</i> | Orang, proses, atau sistem lain yang berinteraksi |

| | |
|--|---|
|  | <p>dengan sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol aktor adalah gambar orang, tapi aktor belum tentu orang, biasanya dinyatakan menggunakan kata benda si awal <i>frase</i> nama aktor</p> |
| <p><i>Association</i></p>  | <p>Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.</p> |
| <p><<extends>></p>  | <p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.</p> |
| <p>Generalization</p>  | <p>Hubungan (umum atau khusus) antara dua buah <i>use case</i> Dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p> |
| <p><<include>></p>  | <p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya</p> |


Sumber: Rosa A.S –M. Salahuddin 2011

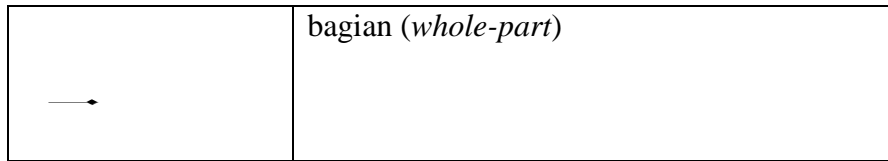
2. Class Diagram

Class Diagram adalah sebuah spesifikasi yang jika *diinstansiasi* akan menghasilkan sebuah obyek dan merupakan inti dari pengembangan dan desain berorientasi obyek. Class menggambarkan keadaan (atribut atau properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda atau fungsi). Sebuah *Class* memiliki tiga area pokok:

- a. Nama, merupakan nama dari sebuah kelas
- b. Atribut, merupakan properti sebuah kelas. Atribut melambangkan batas nilai yang mungkin ada pada obyek dari *class*.
- c. Operasi, adalah sesuatu yang bisa dilakukan oleh sebuah *class* atau yang dapat dilakukan oleh *class* lain terhadap sebuah *class*.

Tabel 2.2 : *Class Diagram*

| Gambar | Keterangan |
|---|--|
|  | Kelas pada struktur sistem |
| <i>Interface</i> | Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek |
|  | Relasi antar kelas dengan makna umum, asosiasi biasanya disertai disertai <i>Multiplicity</i> . |
|  | Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya disertai disertai multiplicity |
|  | Relasi antar kelas dengan makna generalisasi-spesialisasi (umum-khusus) |
| <i>Aggregation</i> | Relasi antar kelas dengan makna semua- |



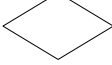



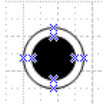
(Sumber: Rosa A.S –M. Salahuddin 2011)

3. Activity Diagram

Activity Diagram yaitu memodelkan alur kerja (*workflow*) sebuah proses bisnis dan urutan aktivitas dalam suatu proses. Diagram ini sangat mirip dengan sebuah *flowchart* karena kita dapat memodelkan sebuah alur kerja dari satu aktivitas ke aktivitas lainnya atau dari satu aktivitas ke keadaan sesaat (*state*). Juga sangat berguna ketika ingin menggambarkan perilaku paralel atau menjelaskan bagaimana perilaku dalam berbagai *use case* berinteraksi.

Tabel 2.3 : *Activity Diagram*

| Gambar | Keterangan |
|---|--|
| Status Awal  | Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal |
| Aktivitas  | Aktivitas yang dilakukan sistem, aktivitasnya biasanya diawali kata kerja |
| Percabangan  | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu |
| Penggabungan atau Join  | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu |

| | |
|---|------------------------------------|
| Status Akhir  | Status akhir yang dilakukan sistem |
|---|------------------------------------|

(Sumber: Rosa A.S –M. Salahuddin, 2011)

2.4 Bahasa Pemrograman C#

Bahasa C# adalah sebuah bahasa pemrograman modern yang bersifat *general-purpose*, berorientasi objek, yang dapat digunakan untuk membuat program di atas arsitektur *Microsoft .NET Framework*. Bahasa C# ini memiliki kemiripan dengan bahasa *Java*, *C* dan *C++*. *Visual C#* tidak diragukan lagi dalam proses pengembangan aplikasi berbasis *.NET Framework*, dimana C# bebas dari masalah kompatibilitas dilengkapi dengan berbagai fitur yang sebagian besar merupakan fitur baru, menarik, dan tentu saja menjanjikan.

C# (baca : *See-Sharp*) adalah bahasa pemrograman baru yang diciptakan Microsoft yang digunakan oleh banyak developer *.NET* untuk mengembangkan applikas dengan *platform .NET*.

2.5 MySQL

MySQL adalah sebuah *database server* yang mampu menerima dan mengirimkan datanya dengan sangat cepat, *multi user* serta menggunakan perintah standar *SQL (Structured Query Language)*. *MySQL* memiliki dua bentuk lisensi yaitu *freeware* dan *shareware*. *MySQL* yang biasa digunakan adalah *MySQL freeware* yang berada di bawah lisensi *GNU/GPL (general public license)* *MySQL* merupakan sebuah *database server* yang *free*, artinya kita bebas menggunakan *database* ini untuk keperluan pribadi atau usaha tanpa harus membeli atau membayar lisensinya. *MySQL* pertama kali di rintis oleh seorang programmer bernama Michael Widenius (Bunafit Nugroho,2005,h1).

2.6 Xampp

Kepanjangan dari XAMPP yaitu Apache, PHP, MySQL dan phpMyAdmin. XAMPP merupakan tool yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dengan menginstal XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi *web server Apache*, PHP dan MySQL secara manual. XAMPP akan menginstalasi dan mengkonfigurasikannya secara otomatis untuk anda atau auto konfigurasi. (Siswoutomo, Wiwit, 2008 : 15). Software XAMPP versi ini terdiri atas:

2.6.1. Apache

Apache sudah berkembang sejak versi pertamanya. Sampai saat ditulisnya artikel ini versi terakhirnya yang ada yaitu *Apache ver 3.2.1*. Apache bersifat *open source*, artinya setiap orang boleh menggunakannya, mengambil dan bahkan mengubah kode programnya.

Tugas utama *apache* adalah menghasilkan halaman *web* yang benar kepada peminta, berdasarkan kode PHP yang dituliskan oleh pembuat halaman *web*. Jika diperlukan juga berdasarkan kode PHP yang dituliskan, maka dapat saja suatu *database* diakses terlebih dahulu (misalnya dalam *MySQL*) untuk mendukung halaman *web* yang dihasilkan.

2.6.2. phpMyAdmin

Pengelolaan *database* dengan *MySQL* harus dilakukan dengan mengetikkan baris-baris perintah yang sesuai (*command line*) untuk setiap maksud tertentu. Jika anda ingin membuat *database*, ketikkan baris perintah yang sesuai untuk membuat *database*. Jika kita ingin menghapus tabel, ketikkan baris perintah yang sesuai untuk menghapus tabel. Hal tersebut tentu cukup menyulitkan karena kita harus hafal dan mengetikkan perintahnya satu persatu.

Banyak sekali perangkat lunak yang dapat dimanfaatkan untuk mengelola *database* dalam *MySQL*, salah satunya adalah *phpMyAdmin*. Dengan *phpMyAdmin* kita dapat membuat tabel, mengisi data dan lain-lain dengan mudah tanpa harus hafal perintahnya. Untuk mengaktifkan *phpMyAdmin* langkah-langkahnya adalah:

1. Mengaktifkan *web server apache* dan *MySQL* dari *control panel XAMPP*
2. Membuka browser, contohnya : *IE, Mozilla Firefox, Google Chrome*
3. Memasukkan alamat *web* *http://localhost/phpmyadmin/* pada *adres bar* lalu tekan enter. Setelah itu akan muncul tampilan *phpMyadmin*. Di sana dapat membuat baris data baru seperti *database*.

2.7 Crystal Report

Crystal Reports merupakan salah satu paket program yang digunakan untuk membuat, menganalisa, dan menterjemahkan informasi yang terkandung dalam *database* ke dalam berbagai jenis laporan. *Crystal Reports* dirancang untuk membuat laporan yang dapat digunakan dengan berbagai bahasa pemrograman berbasis *Windows*, seperti *Visual Basic, Visual C/C++, Visual Interdev, dan Borland Delphi*. Beberapa kelebihan yang dimiliki program *Crystal Reports*, antara lain:

- Pembuatan laporan dengan *Crystal Reports* tidak terlalu rumit dan banyak melibatkan kode program.
- Program *Crystal Reports* banyak digunakan karena mudah terintegrasi dengan bahasa lain.
- Fasilitas impor hasil laporan yang mendukung format-format paket program lain, seperti *Microsoft Office, Adobe Acrobat Reader, HTML, dan sebagainya*.