

BAB II LANDASAN TEORI

2.1. Tinjauan Pustaka

Jamaludin Achmad (2012) telah melakukan penelitian dengan judul ” Sistem Informasi pada Taman Taman Bacaan Masyarakat Nurul Fatah Semarang”. Pada penelitian tersebut membuat program aplikasi yang sudah terkomputerisasi dan sudah terintegrasi *database* namun aplikasi masih terdapat kekurangan. Berdasarkan penelitian tersebut maka dibuat sistem yang lebih lengkap agar dapat terkomputerisasi dengan baik, sehingga pengelolaan taman bacaan masyarakat lebih arsipis.

2.2. Sistem

Menurut Jogiyanto (2005) sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata. Kesatuan yang nyata adalah suatu obyek nyata, seperti tempat, benda dan orang-orang yang betul-betul ada dan terjadi.

1. Prosedur adalah urutan-urutan operasi klerikal atau tulis menulis yang melibatkan beberapa orang di dalam suatu departemen untuk menjamin penanganan yang seragam.
2. Prosedur adalah urutan-urutan yang tepat dari tahapan-tahapan instruksi yang menerangkan apa (*what*) yang harus dikerjakan, siapa (*who*) yang mengerjakan, kapan (*when*) dikerjakan, dan bagaimana (*how*) mengerjakannya.

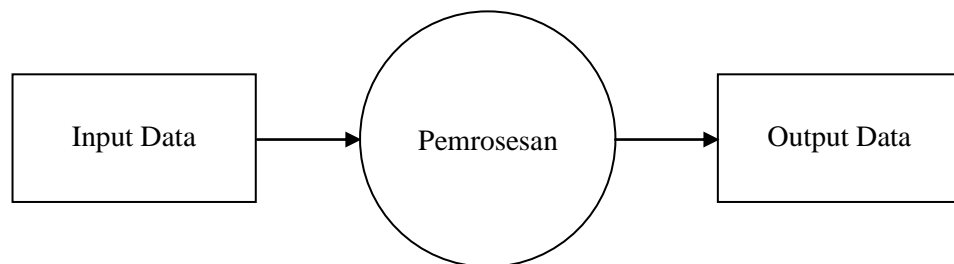
2.3. Informasi

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti lagi bagi yang menerimanya (Jogiyanto, 2000).

Informasi merupakan data yang telah diproses sedemikian rupa sehingga meningkatkan pengetahuan seseorang yang menggunakan data tersebut (Abdul Kadir, 2002).

2.4. Sistem Informasi

Sistem informasi dapat didefinisikan sebagai kumpulan elemen yang saling berhubungan satu sama lain yang membentuk satu kesatuan untuk mengintegrasikan data, memproses, dan menyimpan serta mendistribusikan informasi, atau dengan kata lain sistem informasi adalah kesatuan elemen-elemen yang saling berinteraksi secara sistematis dan teratur untuk menciptakan dan membentuk aliran informasi yang mendukung pembuatan keputusan dan melakukan kontrol terhadap jalannya perusahaan (Teguh Wahyono, 2004). Bagan sistem informasi disajikan pada Gambar 2.1.



Gambar 2.1. Sistem Informasi

2.5. Profil Pusat Kegiatan Belajar Masyarakat (PKBM) Pioneer

Pusat Kegiatan Belajar Masyarakat (PKBM) Pioneer didirikan oleh Bapak W. W. Prayitno, pria kelahiran Boyolali. Awalnya Bapak W. W. Prayitno mempunyai cita-cita menjadi guru. Namun, cita-cita tersebut tidak bisa diwujudkan pasalnya hanya lulusan Sekolah Menengah Umum (SMU). Dengan bekal pendidikan tersebut Bapak Prayitno sempat bekerja pada suatu CV yang bergerak pada bidang bangunan. Jiwanya tidak pernah patah semangat dan dapat menjadikan daya penggerak diri untuk bekerja keras. Bapak Prayitno berusaha menambah pengetahuan dan ketrampilan dengan mengikuti kursus mengetik manual dan tata buku.

Dengan pengalaman mengikuti kursus tersebut, Bapak Prayitno kemudian memberanikan diri untuk mendirikan lembaga kursus yang diberi nama Lembaga Kursus dan Pelatihan (LKP) Pioneer pada tahun 1972. Program yang dikembangkan adalah mengetik, tata buku dan bahasa Inggris.

Dalam perkembangannya sejak tahun 1982, sejalan dengan kebijakan Lembaga Kursus dan Pelatihan (LKP) berubah menjadi Pusat Kegiatan Belajar Masyarakat (PKBM) Pioneer. Perubahan tersebut bertujuan agar lembaga pendidikan ini memiliki dimensi kegiatan dan layanan pembelajaran yang lebih luas. Perubahan LKP menjadi PKBM juga memiliki filosofi yang berbeda. Jika LKP bertujuan untuk mencari profit semata, maka PKBM memiliki tujuan sosial yang sangat mulia. Pusat Kegiatan Belajar Masyarakat (PKBM) Pioneer memiliki visi dan misi sebagai berikut:

1. Visi

Menciptakan masyarakat madani yang berpendidikan atau berdaya, bermanfaat dan mandiri melalui layanan pendidikan yang adil, merata dan bermutu.

2. Misi

- a. Meningkatkan kemandirian masyarakat lapisan bawah yang didasarkan atas prinsip manfaat, kebersamaan, keselarasan, kebutuhan, kesetiakawanan, dan keadilan sosial untuk mengangkat martabatnya.
- b. Mengembangkan sumber daya manusia untuk pendidikan masyarakat yang adil dan martabatnya.
- c. Menjaga eksistensi lembaga PKBM Pioneer agar mandiri dan berkesinambungan.

Awalnya PKBM mengadakan keaksaraan fungsional dan keaksaraan usaha mandiri, kemudian mengadakan program kesetaraan paket A setara SD, paket B setara SMP dan paket C setara SMA. Pada tahun 2007 PKBM Pioneer membuka program Taman Bacaan Masyarakat (TBM) demi meningkatkan minat baca masyarakat yang menyediakan dan memberikan layanan di bidang bahan bacaan, berupa: buku, majalah, tabloid, komik, dan bahan multimedia lain yang

dilengkapi dengan ruangan untuk membaca, diskusi, bedah buku, menulis, dan kegiatan literasi lainnya dan didukung oleh pengelola yang berperan sebagai motivator. Taman Bacaan Masyarakat (TBM) merupakan jantung pendidikan masyarakat, dengan diharapkan mampu memotivasi dan menumbuhkan minat dan kegemaran membaca bagi aksarawan baru, warga belajar, dan masyarakat. Dengan tumbuh kembangnya minat dan kegemaran membaca, maka membaca merupakan suatu kebiasaan yang mesti dilakukan setiap hari sebagaimana memenuhi kebutuhan hidup.

2.6. Taman Bacaan Masyarakat (TBM)

Menurut Buku Pedoman Penyelenggaraan Taman Bacaan Masyarakat (2006), taman bacaan masyarakat adalah sebuah tempat/ wadah yang didirikan dan dikelola baik masyarakat maupun pemerintah untuk memberikan akses layanan bahan bacaan bagi masyarakat sekitar sebagai sarana pembelajaran seumur hidup dalam rangka peningkatan kualitas hidup masyarakat di sekitar Taman Bacaan Masyarakat.

Taman Bacaan Masyarakat yang selanjutnya disebut TBM merupakan peningkatan budaya membaca masyarakat dengan ruang yang disediakan untuk membaca, diskusi, bedah buku, menulis dan kegiatan sejenisnya yang dilengkapi dengan bacaan berupa ; buku, majalah, tabloid, koran, komik, dan bahan multimedia lain, serta pengelola yang berperan sebagai motivator. Dasar hukum taman bacaan masyarakat :

1. Undang –undang nomor 20 tahun 2003 tentang Sistem Pendidikan Nasional
2. Undang-undang nomor 43 tahun 2007 tentang Perpustakaan.
3. Peraturan Presiden nomor 47 tahun 2009, tentang Pembentukan dan Organisasi Kementrian Negara.
4. Peraturan Menteri Pendidikan Nasional nomor 31 tahun 2007 tentang Struktur Organisasi dan Tata Kerja Direktorat Jenderal Pendidikan *Non-Formal* dan *Informal*.

5. Kesepakatan bersama antara Menteri Pendidikan Nasional dengan Menteri Agama nomor 1 tahun 2007 dan nomor 2 tentang Penyelenggaraan Program Pendidikan Luar Sekolah di Lembaga Keagamaan.

2.7. Metode Berorientasi Objek









Metode berorientasi objek adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadapnya. Dalam perancangan metodologi berorientasi objek menggunakan diagram UML (*Unified Modelling Language*) adalah sebuah "bahasa" yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem perangkat lunak. (Adi Nugroho, 2002).

UML (*Unified Modelling Language*) merupakan bahasa untuk menguraikan spesifikasi yang mudah distandarisasikan untuk tujuan pemodelan suatu objek. (Rosa dan Shalahuddin, 2013).

2.7.1. Use Case Diagram

Use case diagram menggambarkan fungsionalitas dari sebuah sistem (apa fungsinya), yang merepresentasikan sebuah interaksi antara aktor dengan sistem (sebuah pekerjaan). Sebuah *use case* dapat meng-include fungsionalitas *use case* lain. Sebuah *use case* dapat di-include oleh lebih dari satu *use case* lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang *common*. Sebuah *use case* juga dapat meng-extend *use case* lain dengan cara *behaviour*-nya sendiri. (Rosa dan Shalahuddin, 2013). Simbol *use case diagram* disajikan dalam Tabel 2.1.

Tabel 2.1. Simbol *Use Case Diagram*.

No	Gambar	Keterangan
1		<i>Actor</i> Menspesifikasi seperangkat peranan yang user sistem dapat diperankan ketika berinteraksi dengan <i>use case</i>
2		<i>Association</i> Menggambarkan interaksi antar <i>actor</i> dan <i>use case</i>
3		<i>Use Case</i> Sebuah deskripsi dari seperangkat aksi-aksi berurutan yang ditampilkan pada sebuah sistem
4		<i>Subsystem</i> Tempat sebuah aktivitas -aktivitas sistem yang sedang berjalan
5		<i>Dependancy</i> Simbol untuk menggambarkan ketergantungan sebuah <i>use case</i> dengan <i>use case</i> lainnya
6		<i>Generalization</i> Relasi antar <i>use case</i> , dimana salah satunya dalam bentuk yang lebih umum dari yang lain
7		<i>Include</i> Menggambarkan bahwa keseluruhan dari sebuah <i>use case</i> merupakan fungsionalitas <i>use case</i> lainnya
8		<i>Extend</i> Menggambarkan hubungan antar <i>use case</i> dimana sebuah <i>use case</i> merupakan fungsionalitas <i>use case</i> lainnya apabila kondisi tertentu terpenuhi







2.7.2. Activity Diagram

Activity diagram menggambarkan *workflow* (aliran kerja) atau aktifitas dari sebuah sistem atau proses bisnis. Perlu diperhatikan bahwa diagram aktifitas menggambarkan aktifitas sistem bukan apa yang dilakukan aktor. Diagram aktifitas juga banyak digunakan untuk mendefinisikan hal-hal sebagai berikut:

1. Rancangan proses bisnis dimana setiap urutan aktifitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
2. Urutan atau pengelompokan tampilan dari sistem atau *user interface* dimana setiap aktifitas dianggap memiliki sebuah rancangan antarmuka tampilan.

3. Rancangan pengujian dimana setiap aktifitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya. (Rosa dan Shalahuddin, 2013). Simbol-simbol *activity diagram* disajikan pada Tabel 2.2.

Tabel 2.2. Simbol *Activity Diagram*





No	Gambar	Nama	Keterangan
1		<i>Initial Node</i>	Mempresentasikan dimulainya alur kerja stau sistem dalam <i>activity diagram</i>
2		<i>Action State</i>	Sebuah <i>state</i> yang menggambarkan eksekusi dari aksi <i>atomic</i> .
3		<i>Transition Between Activities</i>	Mengidentifikasi bahwa suatu objek dari <i>state</i> pertama akan menampilkan aksi-aksi tertentu dan memasuki <i>state</i> kedua ketika peristiwa terjadi pergerakan dari aksi ke aksi lainnya.
4		<i>Join Node</i>	Mengidentifikasi untuk menggabungkan kembali <i>activity</i> atau <i>action</i> yang paralel
5		<i>Fork Node</i>	Mengidentifikasi suatu <i>behavior</i> menjadi <i>activity</i> yang paralel
4		<i>Decision</i>	Menentukan kapan alur dalam aktivitas menjadi bercabang.
5		<i>Final Node</i>	Mempresentasikan bahwa telah diakhirinya alur suatu sistem dalam <i>activity diagram</i>

2.7.3. Sequence Diagram

Sequence diagram menggambarkan interaksi objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Maka dari itu untuk menggambarkan *sequence diagram* harus diketahui objek-objek yang terlibat dalam sebuah *use case* serta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. (Rosa dan Shalahuddin, 2013)

Sequence diagram mendokumentasikan komunikasi/ interaksi antar kelas-kelas. Diagram ini menunjukkan sejumlah obyek dan *message* (pesan) yang diletakkan diantara obyek-obyek didalam *use case*. Perlu diingat bahwa di dalam diagram ini, kelas-kelas dan aktor-aktor diletakkan di bagian atas diagram dengan urutan dari kiri ke kanan dengan garis *lifeline* yang diletakkan secara vertikal terhadap kelas dan aktor. Simbol-simbol *sequence diagram* disajikan pada Tabel 2.3.

Tabel 2.3. Simbol *Sequence Diagram*

No	Gambar	Nama	Keterangan
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
3		<i>Activation</i>	Menyatakan objek dalam keadaan aktif dan berinteraksi dengan pesan
4		<i>Return Message</i>	Menggambarkan pengembalian dari pemanggilan prosedur

2.8. NetBeans

NetBeans diawali dari Xelfi, sebuah proyek mahasiswa tahun 1997 di bawah bimbingan Fakultas Matematika dan Fisika Universitas Charles, Praha. Sebuah perusahaan kemudian dibentuk untuk proyek tersebut dan menghasilkan versi komersial NetBeans IDE hingga kemudian dibeli oleh Sun Microsystems pada tahun 1999. Sun kemudian menjadikan NetBeans *open source* pada bulan Juni tahun 2000. Sejak itu komunitas NetBeans terus berkembang.

NetBeans mengacu pada dua hal, yakni platform untuk pengembangan aplikasi desktop Java, dan sebuah *Integrated Development Environment* (IDE) yang dibangun menggunakan platform NetBeans. Platform NetBeans

memungkinkan aplikasi dibangun dari sekumpulan komponen perangkat lunak modular yang disebut “modul”. Sebuah modul adalah suatu arsip Java (Java Archive) yang memuat kelas-kelas Java untuk berinteraksi dengan NetBeans Open API dan file manifestasi yang mengidentifikasikannya sebagai modul. Karena modul dapat dikembangkan secara independen, aplikasi berbasis platform NetBeans dapat dengan mudah dikembangkan oleh pihak ketiga secara mudah dan *powerfull* (Miftakhul Huda dan Bunafit Komputer, 2010).

2.9. MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. Selain itu MySQL bersifat terbuka atau *open source*.

MySQL adalah sebuah implementasi dari sistem manajemen basis data relational (RDBMS). MySQL merupakan turunan dari SQL (*Structured Query Language*) SQL sendiri mengandung pengetahuan yaitu sebuah bahasa yang digunakan untuk mengakses data dalam basis data relasional. Terdapat beberapa bahasa pemrograman yang biasa dipakai untuk mengakses basis data MySQL antara lain: bahasa pemrograman C, C++, C#, bahasa pemrograman Eiffel, SmartTalk, Java, Lisp, perl, PHP, Phyton, Ruby, REALbasic, dan Tcl. Keistimewaan MySQL antara lain:

1. Portabilitas , yaitu dapat berjalan stabil dalam berbagai macam sistem operasi.
2. Perangkat lunak sumber terbuka, yaitu dapat digunakan secara gratis karena di bawah lisensi GPL.
3. *Performance tuning*, yaitu memiliki kecepatan dalam menangani query sederhana.
4. Ragam tipe data , yaitu memiliki tipe data yang banyak.
5. Perintah dan fungsi, yaitu memiliki operator dan fungsi secara penuh yang mendukung perintah. *SELECT* dan *WHERE* dalam perintah query.
6. Keamanan, yaitu memiliki beberapa level keamanan dan izin akses user dengan sistem perizinan yang mendetail serta sandi terenskripsi.

7. Skalabilitas dan pembatasan, yaitu mampu menangani basis data dalam skala besar.
8. Konektivitas, yaitu dapat menggunakan koneksi dengan klien, menggunakan protocol TCP/ IP Unix soket (UNIX) atau Named Pipes (NT).
9. Lokalisasi, yaitu dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa.
10. Antar muka, yaitu memiliki antar muka terhadap berbagai macam aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Appllication Programmning Interface*).
11. Klien dan peralatan, yaitu dilengkapi dengan berbagai macam peralatan (*tool*) yang dapat digunakan untuk administrasi basis data.
12. Struktur table, yaitu memiliki struktur table yang lebih fleksibel dalam menangani ALTER TABLE (Miftakhul Huda dan Bunafit Komputer, 2010).

2.10. Pengujian Perangkat Lunak

Menurut Iskandaria (2012), Pengujian *blackbox* adalah salah satu metode pengujian perangkat lunak yang berfokus pada sisi fungsionalitas, khususnya pada *input* dan *output* aplikasi, apakah sudah sesuai dengan apa yang diharapkan atau belum). Tahap pengujian merupakan salah satu tahap yang harus ada dalam sebuah siklus pengembangan perangkat lunak.

Menurut Shihab (2011), *Blackbox testing* merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetasan pada spesifikasi fungsional program. Ciri-ciri *blackbox testing*, yaitu:

1. *Blackbox testing* berfokus pada kebutuhan fungsional pada *software*, berdasarkan pada spesifikasi kebutuhan dari *software*.
2. *Blackbox testing* bukan teknik alternatif daripada *whitebox testing*. Lebih dari pada itu, *blackbox* merupakan pendekatan pelengkap dalam mencakup error dengan kelas yang berbeda dari metode *whitebox testing*.

Pengujian ini dimulai dengan membuat suatu uji kasus untuk setiap fungsi pada perangkat lunak untuk merancang sistem informasi pengendalian dan rencana produksi, kemudian dilanjutkan dengan pengujian terhadap modul-modul dan terakhir pada tampilan antar muka untuk memastikan tidak ada kesalahan dan semua berjalan dengan baik dan input yang diberikan hasilnya sesuai dengan yang diinginkan.

Dalam tahap ini merupakan tahap akhir pada pembuatan sistem. Sistem akan dilihat dan diamati apakah hasil dari eksekusi yang ada pada sistem sesuai dengan rencana awal pembuatan. Metode pengujian menggunakan metode *blackbox* yaitu pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Dengan metode pengujian ini dianalogikan seperti kotak hitam yang hanya bisa melihat penampilan luarnya saja tanpa tau ada apa dibalik bungkus hitamnya. Pengujian sistem informasi Taman Bacaan Masyarakat (TBM) di Pusat Kegiatan Belajar Masyarakat (PKBM) Pioneer dilakukan dengan metode *blackbox*. Pengujian dilakukan pada fungsi-fungsi sistem untuk menentukan apakah fungsi tersebut telah berjalan sesuai dengan yang diharapkan.