

BAB II

LANDASAN TEORI

2.1 Sistem

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

Sistem adalah seperangkat elemen yang membentuk kumpulan atau prosedur-prosedur atau bagan-bagan pengolahan yang mencari suatu tujuan bagian atau tujuan bersama dengan mengoperasikan data dan/atau barang pada waktu rujukan tertentu untuk menghasilkan informasi dan/atau energi dan/atau barang (Murdick R. G, 1991).

Sistem digunakan dalam membentuk beberapa prosedur yang digunakan dalam mengolah data yang dimasukkan oleh bagian-bagian yang ada pada SMA Khadijah Surabaya hingga tersampainya informasi kepada wali murid SMA Khadijah Surabaya. Prosedur yang dibuat termasuk proses pengolahan data hingga menjadi informasi yang berguna terhadap *end-user*.

2.2 Informasi

Informasi dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian (*event*) yang nyata (*fact*) yang digunakan untuk pengambilan keputusan (Jogiyanto, 1999).

Informasi didapatkan dengan megolah data-data yang terdapat pada sistem *back-end* untuk dijadikan informasi yang dapat ditampilkan pada bagian *front-end*. Informasi berguna bagi wali murid dikarenakan melalui informasi yang baik proses evaluasi dan pemantauan anak walinya akan berjalan dengan baik juga. Beberapa informasi yang dibutuhkan wali murid meliputi informasi nilai siswa, presensi siswa, jadwal pelajaran siswa, jadwal kegiatan siswa, pembayaran SPP siswa, prestasi siswa, dan catatan guru.

2.3 Sistem Informasi

Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengelolaan transaksi harian, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang dibutuhkan (Hutahaean J, 2014).

Sistem informasi digunakan dalam melakukan proses pengolahan data hingga menjadi informasi melalui data-data yang dimasukkan melalui bagian belakang sistem yang dikelola dan diolah menjadi informasi yang di tampilkan pada bagian depan sistem.

Menurut Yakub (2012: 20) Sistem informasi merupakan sebuah susunan yang terdiri dari beberapa komponen atau elemen. Komponen-komponen dari sistem informasi ini dapat digambarkan sebagai berikut ini:

1. Blok Masukan (*Input Block*), Input memiliki data yang masuk ke dalam sistem informasi, juga metode-metode untuk menangkap data yang dimasukkan.

2. Blok Model (*Model Block*), blok ini terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data.
3. Blok Keluaran (*Output Block*), produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.
4. Blok Teknologi (*Technology Block*), blok teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dari sistem secara keseluruhan. Teknologi terdiri dari tiga bagian utama, yaitu; teknisi (*brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).
5. Basis Data (*Database Block*), basis data merupakan kumpulan dari data yang saling berhubungan satu sama lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

2.4 Sistem Informasi Akademik

Sistem Informasi Akademik adalah Sistem yang memberikan layanan informasi yang berupa data dalam hal yang berhubungan dengan akademik. Dimana dalam hal ini pelayanan yang diberikan yaitu seperti : penyimpanan data untuk siswa baru, penentuan kelas, penentuan jadwal pelajaran, pembuatan jadwal mengajar, pembagian wali kelas, dan proses penilaian. (Imelda & Erik, 2004).

Sistem informasi akademik dapat dikembangkan pada sekolah menengah atas seiring dengan adanya perkembangan teknologi saat ini. Salah satu manfaat dari sistem informasi akademik ini adalah kualitas layanan sekolah akan menjadi

lebih baik melalui pemberian fasilitas kepada wali murid dalam melakukan pemantauan sekaligus evaluasi terhadap anak walinya.

Beberapa data yang dapat diolah menjadi layanan akademik adalah sebagai berikut:

1. Data siswa digunakan dalam menyimpan profil siswa mulai dari nama, alamat, tempat/tanggal lahir, kelas dan nomor induk.
2. Penentuan kelas digunakan dalam menentukan kelas siswa termasuk jenjang dan jurusan yang diambil oleh siswa pada saat periode ajaran tertentu.
3. Penentuan jadwal pelajaran digunakan oleh bagian kurikulum dalam menentukan jadwal pelajaran pada setiap kelas pada periode ajaran tertentu.
4. Pembuatan jadwal mengajar digunakan untuk menentukan pengajar pada jadwal pelajaran yang telah ditentukan pada periode ajaran tertentu.
5. Pembagian wali kelas digunakan oleh bagian kesiswaan dalam menentukan wali kelas pada suatu kelas pada periode ajaran tertentu.
6. Proses penilaian dilakukan oleh guru pada sesuai dengan pembagian jadwal mengajar terhadap siswa hingga tahapan penghitungan nilai akhir yang telah ditentukan oleh sekolah.

2.5 Analisis sistem

Analisis sistem ialah teknik pemecahan masalah yang menguraikan sistem menjadi kepingan-kepingan komponen, dalam keperluan mempelajari cara kerja bagian-bagian komponen itu, dan caranya dalam berinteraksi untuk mencapai tujuannya (Jeffery L. Whitten, 2004).

Analisis sistem dilakukan dengan cara melakukan penguraian terhadap masalah yang ditemukan untuk diketahui kebutuhannya. Analisis sistem dibangun

dengan melakukan identifikasi dan penguraian proses bisnis yang ada pada SMA Khadijah Surabaya saat ini. Penguraian dilakukan hingga tertulis secara deskriptif guna mendapatkan komponen-komponen yang dibutuhkan secara jelas sehingga dapat membantu dalam mencapai tujuan yang sesuai. Analisis sistem dapat digunakan dalam penguraian sistem sesuai dengan permasalahan yang didapatkan dengan mempelajari proses bisnis pada SMA Khadijah saat ini.

2.6 Perancangan berorientasi objek

Object oriented merupakan sebuah paradigma baru dalam rekayasa perangkat lunak. Paradigma ini memandang sistem sebagai kumpulan objek-objek diskrit yang saling berinteraksi satu dengan lainnya. Berorientasi objek bisa juga bermakna sebagai kumpulan objek-objek diskrit yang bekerja sama antara informasi (atau struktur data) dan perilaku (*behaviour*) yang mengaturnya. Pada paradigma berorientasi objek, setiap objek mempunyai dua pengenalan, yaitu informasi tentang objek tersebut dan perilaku yang mengaturnya. Informasi tentang objek dapat disebut dengan *attribut* sedangkan perilaku pada sebuah objek merupakan operasi yang menjadikan suatu objek memiliki perilaku khusus baik pada objek itu sendiri atau berpengaruh pada objek lainnya (Sholih, 2010).

2.7 Konsep Berorientasi Objek

Pada perancangan berorientasi objek terdapat beberapa paradigma orientasi objek (Sholih, 2010), antara lain:

1. Abstraksi

Sebuah objek harus memiliki atribut sebagai informasi atas objek tersebut dan operasi sebagai perilaku yang mengatur objek. Abstraksi merupakan cara

paling dasar guna mengelola kompleksitas dimana tahap mengelola tersebut dilakukan dengan cara mengabaikan sesuatu yang tidak terlalu penting dalam hubungan objek dan hanya berkonsentrasi pada sesuatu yang signifikan pada objek. Abstraksi juga merupakan proses memilah beberapa atribut dan beberapa operasi suatu obyek hanya sampai pada bagian yang benar-benar diperlukan.

2. Pewarisan

Pewarisan merupakan keadaan dimana suatu *class* dapat diturunkan ke *class* lainnya. Terdapat dua jenis pewarisan yaitu kelas ke objek-objeknya dan dari super kelas ke sub kelas. Fungsi dari pewarisan adalah mewariskan sifat-sifat kelas induk ke kelas turunannya berupa atribut dan operasi.

3. Banyak bentuk (*Polymorfism*)

Banyak bentuk (*Polymorfism*) merupakan beberapa operasi dengan nama yang sama namun jika diberikan pada objek yang berbeda akan mengakibatkan operasi yang berbeda.

4. Pengakpsulan (*Encapsulation*)

Pengakpsulan (*Encapsulation*) merupakan kegiatan memisahkan aspek-aspek eksternal objek yang dapat diakses objek-objek lain dari rincian implementasi internal pada objek itu sendiri.

5. Pengiriman pesan

Pengiriman pesan merupakan proses dimana terjadinya komunikasi antar objek. Suatu objek mengirimkan pesan ke objek lain untuk melakukan sebuah operasi. Juga dalam rangka, suatu objek menerima pesan dari objek lain untuk melakukan operasi lainnya.

6. Asosiasi

Asosiasi merupakan hubungan antar objek-objek dalam sistem. Sebuah objek diasosiasikan dengan objek lainnya dengan menggunakan lebih dari satu cara.

7. Agregasi

Agregasi merupakan bentuk khusus dari asosiasi dengan hubungan lebih kuat, ketika asosiasi terjadi adalah part of antara objek yang satu dengan beberapa objek lainnya, atau asosiasi antara keseluruhan dengan bagian.

2.8 UML

UML (*Unified Modeling Language*) adalah Metodologi kolaborasi antara metoda-metoda Booch, OMT (*Object Modeling Technique*), serta OOSE (*Object Oriented Software Engineering*) dan beberapa metoda lainnya, merupakan metodologi yang paling sering digunakan saat ini untuk analisa dan perancangan sistem dengan metodologi berorientasi objek mengadaptasi maraknya penggunaan bahasa “pemrograman berorientasi objek” (OOP) (Nugroho, 2009).

UML digunakan sebagai metodologi untuk melakukan analisa dan perancangan berorientasi objek dalam pembuatan sistem informasi yang akan dibangun pada SMA Khadijah Surabaya melalui beberapa tahapan penggambaran sistem yang terintegrasi.

UML menyediakan beberapa diagram visual yang menunjukkan berbagai aspek dalam sistem. Banyaknya diagram tersebut dimaksudkan untuk memberikan gambaran yang lebih terintegrasi terhadap sistem yang akan dibangun. Beberapa diagram yang terdapat dalam UML (Sholiq, 2013), antara lain:

1. Diagram *Use Case* Bisnis

Use case bisnis digunakan untuk mempresentasikan bisnis yang dilakukan oleh organisasi. Selain itu *use case* bisnis digunakan untuk memodelkan aktivitas bisnis organisasi sebagai landasan pembuatan *Use Case System*.

2. Diagram *Use Case System*

Use Case System menyajikan interaksi antara *use case* dan aktor dalam sistem yang akan dikembangkan. Sedangkan *use case* sendiri merupakan fungsionalitas atau persyaratan-persyaratan sistem yang harus dipenuhi oleh sistem yang akan dikembangkan tersebut menurut pandangan pemakai sistem.

3. *Flow of Events*

Flow of Events merupakan dokumen yang bertujuan guna mendokumentasikan alur logika dalam *use case* yang menjelaskan secara rinci apa yang pemakai akan lakukan dan apa yang sistem itu sendiri lakukan.

4. Diagram Sekuensial

Diagram sekuensial atau *sequence diagram* merupakan diagram yang digunakan untuk menunjukkan alur (*flows*) fungsionalitas yang melalui sebuah *use case* yang disusun dalam urutan waktu.

5. Diagram Kolaborasi

Diagram kolaborasi merupakan diagram yang digunakan untuk menunjukkan informasi yang sama persis dengan diagram sekuensial, tetapi dalam bentuk dan tujuan yang berbeda. Pada diagram sekuensial keseluruhan interaksi berdasarkan urutan waktu sedangkan dalam diagram kolaborasi interaksi antar objek atau faktor ditunjukkan dengan arah panah tanpa keterangan waktu.

6. Diagram Kelas

Diagram kelas merupakan diagram yang menunjukkan interaksi antar kelas-kelas dalam sistem. Sebuah kelas mengandung informasi (atribut) dan tingkah laku (*behavior*) yang berkaitan dengan informasi tersebut.

7. Diagram Statechart

Diagram statechart merupakan diagram yang digunakan untuk memodelkan bermacam-macam *state* yang mungkin dialami oleh objek tunggal. *Diagram statechart* digunakan untuk menggambarkan perilaku dinamik sebuah objek tunggal.

8. Diagram Komponen

Diagram komponen merupakan diagram yang menunjukkan model secara fisik atau komponen perangkat lunak pada sistem dan hubungannya antar mereka. Diagram komponen menunjukkan komponen apa yang dibutuhkan saat proses kompilasi dan menampilkan komponen runtime apa saja yang dihasilkan sebagai hasil proses kompilasi.

9. Diagram Deployment

Diagram deployment merupakan diagram yang menampilkan rancangan fisik jaringan tempat berbagai komponen akan diletakkan. Dalam pengembangan suatu sistem hanya digunakan satu buah diagram deployment untuk satu sistem.

2.9 Waterfall

Model Air Terjun (*Waterfall*) sering juga disebut model sekuensial linier (*Sequential Linear*) atau alur hidup klasik (*Classic Life Cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut

dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*) (Rosa, salahuddin, 2011).

1. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

2. Desain

Desain perangkat lunak adalah proses multistep yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

3. Pembuatan kode program

Desain harus ditranslasikan kedalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan kita sudah dikirim ke user. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

