

BAB 2

DASAR TEORI

2.1 Sistem Informasi Akademik

Sistem informasi akademik adalah suatu sistem informasi untuk menangani pengolahan data- data akademik yang dibutuhkan siswa, guru, wali murid dan pengguna yang membutuhkan sistem informasi akademik. Sistem informasi ini dibangun untuk mengelola data-data akademik dengan mudah. Untuk mendukung hal ini maka digunakan teknologi pemrograman dan database sehingga penyimpanan yang digunakan instansi saat ini dapat membantu dalam mengklasifikasikan data dan informasi. (Indrayasa, 2015)

2.2 Monitoring

Kegiatan monitoring lebih terfokus pada kegiatan yang akan dilaksanakan. Monitoring dilakukan dengan cara menggali untuk mendapatkan informasi secara regular berdasarkan indikator tertentu, dengan maksud mengetahui apakah kegiatan yang sedang berlangsung sesuai dengan perencanaan dan prosedur yang telah disepakati. Indikator monitoring mencakup esensi aktivitas dan target yang ditetapkan pada perencanaan program. Apabila monitoring dilakukan dengan baik akan bermanfaat dalam memastikan pelaksanaan kegiatan tetap pada jalurnya (sesuai pedoman dan perencanaan program). Juga memberikan informasi kepada pengelola program apabila terjadi hambatan dan penyimpangan, serta sebagai masukan dalam melakukan evaluasi. Secara prinsip, monitoring dilakukan sementara kegiatan sedang berlangsung guna memastikan kesesuaian proses dan capaian sesuai rencana, tercapai atau tidak. Bila ditemukan penyimpangan atau kelambanan maka segera dibenahi sehingga kegiatan dapat berjalan sesuai rencana dan targetnya. Jadi, hasil monitoring menjadi *Input* bagi kepentingan proses selanjutnya. (Legowo, 2016)

2.3 Web Browser

Web Browser adalah program untuk menampilkan halaman yang berbentuk kode HTML. Semua halaman web ditulis dengan bahasa HTML (*Hypertext Mark Up Language*). Walaupun beberapa file mempunyai ekstensi yang berbeda (contoh: .html, .php, .php3), *output* file-file tersebut tetap HTML. HTML adalah medium yang selalu dikirimkan ke

web browser baik halaman itu berupa halaman statis, sebuah *script* seperti PHP, ataupun yang dibuat oleh program CGI (*Common Gateway Interface*). (Peranginangin, 2006)

2.4 PHP

PHP Adalah bahasa *scripting server-side*, Bahasa pemrograman yang digunakan untuk mengembangkan situs web statis atau situs web dinamis atau aplikasi web. PHP singkatan dari *Hypertext Pre-processor*, yang sebelumnya disebut *Personal Home Pages*. Karena php merupakan *scripting server-side* maka jenis bahasa pemrograman ini nantinya *script/program* tersebut akan dijalankan/diproses oleh server. Berbeda dengan *javascript* yang *client-side*. PHP adalah bahasa pemrograman umum yang berarti php dapat disematkan ke dalam kode HTML, atau dapat digunakan dalam kombinasi dengan berbagai sistem *template* web, sistem manajemen konten web, dan kerangka kerja web. Fungsi PHP adalah membuat atau mengembangkan situs web statis atau situs web dinamis atau aplikasi Web. Walaupun sebenarnya bukan hanya PHP bahasa pemrograman yang bisa digunakan untuk memuat website. PHP digunakan karena untuk membuat website dinamis bisa digunakan untuk menyimpan data ke dalam database, membuat halaman yang dapat berubah-ubah sesuai dengan *Input* user, memproses form, dll. (Intermedia, 2019)

2.5 MySQL

MySQL adalah sebuah DBMS (*Database Management System*) menggunakan perintah SQL (*Structured Query Language*) yang banyak digunakan saat ini dalam pembuatan aplikasi berbasis website. MySQL dibagi menjadi dua lisensi, pertama adalah *Free Software* dimana perangkat lunak dapat diakses oleh siapa saja. Dan kedua adalah *Shareware* dimana perangkat lunak berpemilik memiliki batasan dalam penggunaannya.

MySQL termasuk ke dalam RDBMS (*Relational Database Management System*). Sehingga, menggunakan tabel, kolom, baris, di dalam struktur *database*-nya. Jadi, dalam proses pengambilan data menggunakan metode *relational database*. Dan juga menjadi penghubung antara perangkat lunak dan *database server*. (Adani, 2020)

2.6 Framework

Framework adalah sebuah kerangka program yang digunakan untuk membantu developer untuk mengembangkan kode secara konsisten. Framework memiliki fungsi utama untuk mempermudah para developer mengembangkan software dan aplikasi. Karena framework memiliki program yang tersusun dengan struktur MVC (*Model View Controller*). Framework memiliki tiga jenis platform, yaitu *Ajax framework*, *web application framework* dan *application framework*. Dari ketiga platform tersebut memiliki perbedaan penggunaannya. Salah satu contoh *web application* adalah CodeIgniter (Kurniawan, 2020).

2.7 Laravel

Laravel diluncurkan sejak tahun 2011 dan mengalami pertumbuhan yang cukup eksponensial. Di tahun 2015, Laravel adalah framework yang paling banyak mendapatkan bintang di Github. Sekarang framework ini menjadi salah satu yang populer di dunia, tidak terkecuali di Indonesia. Laravel fokus di bagian end-user, yang berarti fokus pada kejelasan dan kesederhanaan, baik penulisan maupun tampilan, serta menghasilkan fungsionalitas aplikasi web yang bekerja sebagaimana mestinya. Hal ini membuat developer maupun perusahaan menggunakan framework ini untuk membangun apa pun, mulai dari proyek kecil hingga skala perusahaan kelas atas.

Laravel mengubah pengembangan website menjadi lebih elegan, ekspresif, dan menyenangkan, sesuai dengan jargonnya “The PHP Framework For Web Artisans”. Selain itu, Laravel juga mempermudah proses pengembangan website dengan bantuan beberapa fitur unggulan, seperti Template Engine, Routing, dan Modularity. (Kurniawan Y. , 2019)

2.8 PHP Mailer

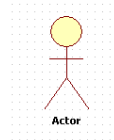
PHP Mailer adalah fungsi PHP yang digunakan untuk mengirim *e-mail* pada server localhost. PHP Mailer dapat menjalankan fungsinya sebagai pengirim *e-mail* jika kita mensupportnya dengan Simple Mail Transfer Protocol (SMTP). SMTP adalah suatu protokol yang diperlukan untuk mengirim dan menerima *e-mail*. Karena itulah kita harus menggunakan SMTP sebagai layanan mengirim *e-mail*. Layanan ini dapat kita gunakan untuk keperluan seperti memverifikasi *e-mail*, contohnya ketika kita mendaftar di Twitter

atau Facebook. Setelah mendaftar maka kita diharuskan membuka *e-mail* dan memverifikasinya. (Romansa, 2015)

2.9 Use Case

Use case merupakan sebuah teknik yang digunakan dalam pengembangan sebuah software atau sistem informasi untuk menangkap kebutuhan fungsional dari sistem yang bersangkutan, *use case* menjelaskan interaksi yang terjadi antara ‘aktor’ — inisiator dari interaksi sistem itu sendiri dengan sistem yang ada, sebuah *use case* direpresentasikan dengan urutan langkah yang sederhana. *Use case* memiliki beberapa komponen yaitu :

2.9.1 Actor

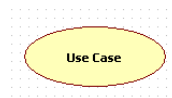


Gambar 2. 1 Actor Use Case

Sumber : (Rahardjo, 2018)

Menggambarkan seseorang yang berinteraksi dengan sistem, di mana hanya bisa meng*Inputkan* informasi dan menerima informasi dari sistem dan tidak memegang kendali pada *use case*. Dan biasa *actor* di gambarkan dengan *stickman*.

2.9.2 Use Case



Gambar 2. 2 Use Case

Sumber : (Rahardjo, 2018)

Gambaran fungsional sistem yang akan di buat, agar pengguna lebih mengerti penggunaan sistem.

2.9.3 Relasi Dalam Use Case

Ada beberapa relasi yang terdapat pada *use case diagram*:

1. *Association*, menghubungkan link antar element.

2. *Generalization*, sebuah elemen yang menjadi spesialisasi dari elemen yang lain.
3. *Dependency*, sebuah elemen yang bergantung beberapa cara kepada elemen lainnya.
4. *Aggregation*, bentuk association yang di mana sebuah elemen berisi elemen lainnya.




Tipe relasi/ *stereotype* yang mungkin terjadi pada *use case diagram*:


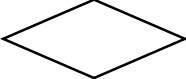
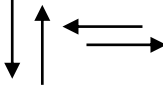
1. <<*include*>> , yaitu kelakuan yang harus terpenuhi agar sebuah *event* dapat terjadi, dimana pada kondisi ini sebuah *use case* adalah bagian dari *use case* lainnya.
2. <<*extends*>>, kelakuan yang hanya berjalan di bawah kondisi tertentu seperti menggerakkan alarm.
3. <<*communicates*>>, mungkin ditambahkan untuk asosiasi yang menunjukkan asosiasinya adalah *communicates association* . Ini merupakan pilihan selama asosiasi hanya tipe *relationship* yang dibolehkan antara *actor* dan *use case*.(Rahardjo, 2018)

2.10 Activity Diagram

Activity diagram pada dasarnya menggambarkan macam-macam alir aktivitas yang akan dirancang dalam sebuah sistem. Dimana masing-masing alir memiliki awal, decision yang mungkin terjadi pada sistem, dan akhir dalam sistem tersebut. *Activity* diagram pada dasarnya memiliki struktur yang hampir mirip dengan *flowchart* atau diagram alir dalam perancangan sistem secara terstruktur. *Activity* diagram ini dibuat berdasarkan sebuah *use case* atau beberapa *use case* dalam *use case diagram*. (Rahardjo, 2018)

Tabel 2. 1 Simbol *Activity* Diagram

No.	Gambar	Nama	Keterangan
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari system yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali

No.	Gambar	Nama	Keterangan
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan diakhiri
5		<i>Decision</i>	Digunakan untuk menggambarkan suatu keputusan/tindakan yang harus diambil pada kondisi tertentu
6		<i>Line Connector</i>	Digunakan untuk menghubungkan satu simbol dengan simbol lainnya

Sumber : (Rahardjo, 2018)

2.11 *Conceptual Data Model(CDM)*

CDM adalah model yang dibuat berdasarkan anggapan bahwa dunia nyata terdiri dari koleksi obyek-obyek dasar yang dinamakan entitas (Entity) serta hubungan (relationship) antara entitas-entitas tersebut. CDM memodelkan struktur logis dari keseluruhan aplikasi data, tidak tergantung pada software atau pertimbangan model struktur data. CDM yang valid dapat dikonversi ke PDM atau OOM. dalam penerapannya dapat di samakan dengan ERD yang fungsinya memang sama yaitu memodelkan struktur logik dari basis data. (Zulhaidir, 2019)

2.12 *Physical Data Model(PDM)*

PDM merupakan model yang menggunakan sejumlah tabel untuk menggambarkan data serta hubungan antara data-data tersebut. Setiap tabel mempunyai sejumlah kolom di mana setiap kolom memiliki namayang unik. PDM merupakan gambaran secara detail basis data dalam bentuk fisik. Penggambaran rancangan PDM memperlihatkan struktur penyimpanan data yang benar pada basis data yang digunakan sesungguhnya. (Zulhaidir, 2019)

2.13 **Class Diagram**

Class diagram adalah visual dari struktur sistem program pada jenis-jenis yang di bentuk. Class diagram merupakan alur jalannya database pada sebuah sistem. Class diagram merupakan penjelasan proses database dalam suatu program. Dalam sebuah laporan sistem

maka class diagram ini wajib ada. Fungsi dan manfaat dari class diagram adalah sebagai berikut:

1. Menjelaskan suatu model data untuk program informasi, tidak peduli apakah model data tersebut sederhana maupun kompleks.
2. Dengan menguasai class diagram maka akan meningkatkan pemahaman mengenai gambaran umum skema dari suatu program.
3. Mampu menyatakan secara visual akan kebutuhan spesifik suatu informasi serta dapat berbagi informasi tersebut ke seluruh bisnis.
4. Dengan Class Diagram dapat dibuat bagan secara terperinci dan jelas, dengan cara memperhatikan kode spesifik apa saja yang dibutuhkan oleh program. Hal ini mampu mengimplementasikan ke struktur yang dijelaskan.
5. Class Diagram mampu memberikan penjelasan implementasi-independen dari suatu jenis program yang digunakan, kemudian dilewatkan diantara berbagai komponennya. (Ansori, 2020)