

BAB II

TINJAUAN PUSTAKA

1.1. Sistem Informasi

Menurut Bonnie Soeherman dalam Aasinjery (2020:5) menyatakan bahwa sistem informasi merupakan serangkaian komponen berupa manusia, prosedur, data, dan teknologi (seperti komputer) yang digunakan untuk melakukan sebuah proses untuk pengambilan keputusan guna menunjang keberhasilan bagi setiap organisasi (dalam pencapaian tujuan).

Informasi merupakan kumpulan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerima. Tanpa suatu informasi, suatu sistem tidak akan berjalan dengan lancar dan akhirnya bisa mati. Suatu organisasi tanpa adanya suatu informasi maka organisasi tersebut tidak bisa berjalan dan tidak bisa beroperasi (Kristanto, 2018).

Sistem informasi merupakan suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi (Kristanto, 2018).

1.2. Website

Yeni Susilowati (2019), *website* adalah sejumlah halaman *web* yang memiliki topik saling terkait antar satu halaman dan halaman yang lainnya, yang biasanya ditempatkan pada sebuah *server web* yang dapat di akses melalui jaringan internet maupun jaringan wilayah lokal (*LAN*).

1.3. Biro Kemahasiswaan

Biro Kemahasiswaan adalah bagian yang memfokuskan kegiatannya dengan mendukung dan mengelola kegiatan mahasiswa baik itu kegiatan himpunan, ukm (unit kegiatan mahasiswa) atau kegiatan mahasiswa lainnya (Munazar, A. A. 2019).

1.4. Organisasi Kemahasiswaan

Organisasi kemahasiswaan diartikan sebagai sebuah wadah bagi mahasiswa dalam meningkatkan dan mengembangkan minat, bakat, keahlian, ilmu pengetahuan, dan keterampilan sosial. Organisasi kemahasiswaan adalah suatu kegiatan yang berada di perguruan tinggi dan anggota organisasinya ialah organisasi kemahasiswaan termasuk dalam

bagian organisasi namun ruang lingkupnya berada di perguruan tinggi. (Amalia Dwi Pertiwi, 2021)

1.5. Layanan

Layanan mahasiswa bertujuan untuk memberikan solusi atas permasalahan yang terjadi pada mahasiswa saat perkuliahan. Seluruh data perkuliahan mahasiswa harus di simpan layanan mahasiswa. (Buhori muslim 2018)

1.6. Minat Bakat

Minat merupakan sumber motivasi yang mendorong orang untuk melakukan apa yang mereka inginkan bila mereka bebas memilih. Bakat adalah sejumlah karakter, kondisi atau kualitas yang di miliki seseorang sejak lahir (bawaan) yang berupa potensi yang akan berpengaruh terhadap kemungkinan keberhasilan seseorang dalam suatu bidang pekerjaan. (Noraini 2021)

1.7. Mentor Minat Bakat

Mentor minat bakat adalah mahasiswa semester lanjut yang memiliki pengalaman, pengetahuan, dan keterampilan yang di pilih langsung oleh bagian kemahasiswaan melalui tahapan proses seleksi antara lain; seleksi berkas, microteaching, dan wawancara. Mentor bertugas membimbing atau memonitoring mahasiswa baru selama satu semester dengan goals mahasiswa baru di harapkan dapat menjadi kaderkader yang aktif dalam berorganisasi dan mengembangkan minat bakat nya.

1.8. PHP

Menurut Ani Oktarini Sari, Ari Abdilah, dan Sunarti (2019:23), PHP atau *PHP Hypertext Preprocessor* adalah sebuah bahasa script berbasis *server (server-side)* yang mampu mem-parsing *kode php* dari *kode web* dengan ekstensi *.php*, sehingga menghasilkan tampilan *website* yang dinamis di sisi client (browser). Dengan menambahkan skrip PHP, anda bisa menjadikan halaman HTML menjadi lebih *powerful*, dinamis dan bisa dipakai sebagai aplikasi lengkap, misalnya *web portal, e-learning, elibrary*, dll.

2.9 Alat Implementasi

2.9.1. My SQL

MySQL adalah Relational Database Management System (RDBMS) open-source yang dikembangkan dan didukung oleh Oracle Corporation. MySQL didukung pada sejumlah besar platform, termasuk varian Linux, OS X, dan Windows. Ini juga memiliki API untuk sejumlah besar bahasa, termasuk C, C++, Java, Lua, .Net, Perl, PHP, Python, dan Ruby (Stack Overflow Documentation, 2019).

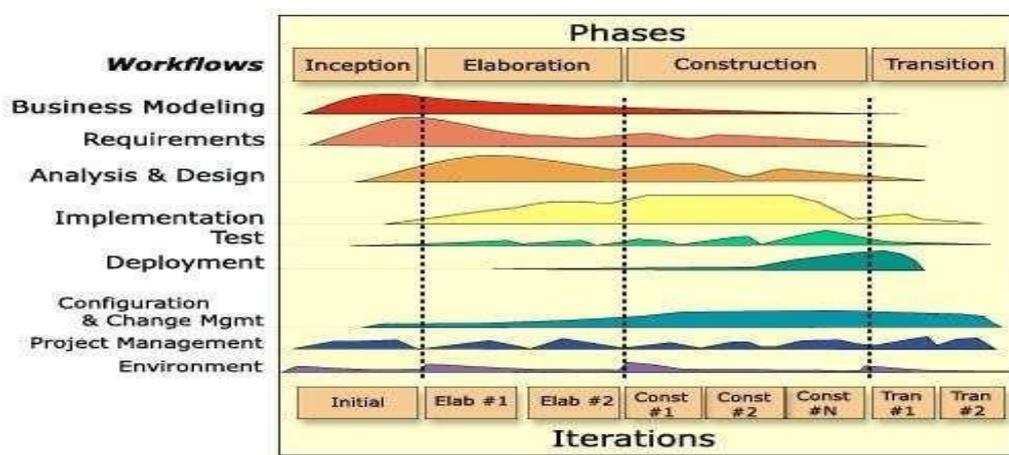
2.9.2. Xampp

Menurut MADCOMS (2016) Xampp adalah sebuah paket kumpulan software yang terdiri dari Apache, MySQL, PhpMyAdmin, PHP, Perl, Filezilla, dan lain.

Xampp berfungsi untuk memudahkan instalasi lingkungan PHP, di mana biasanya lingkungan pengembangan web memerlukan PHP, Apache, MySQL dan PhpMyAdmin (MADCOM, 2016).

2.10. Metode Pengembangan Sistem

Metode RUP (*Rational Unified Process*) adalah suatu kerangka kerja (*framework*) pengembangan perangkat lunak yang telah ada sejak tahun 1990an. Metode ini di kembangkan oleh *Rational Software Corporation* (sekarang merupakan bagian dari IBM) dan sering digunakan untuk mengelola proyek pengembangan perangkat lunak secara lebih terstruktur dan efisien. RUP adalah salah satu pendekatan proses pengembangan perangkat lunak yang bersifat iteratif dan inkremental. Berikut adalah tahapan pengembangan sistem dalam RUP:



Gambar 2.1 metode Rup

1. Fase Insepsi (*inception*)

Fase insepsi adalah untuk membatasi sistem dengan cukup sebagai dasar untuk memvalidasi biaya awal dan penganggaran. Pada fasa ini, ditentukan kasus bisnis yaitu: konteks bisnis, faktor sukses (perkiraan pendapatan, pengenalan ke pasar, dll.) dan perkiraan finansial. Sebagai pelengkap kasus bisnis adalah model penggunaan, perencanaan proyek, penilaian risiko tahap awal, dan deskripsi proyek disusun.

2. Fase Elaborasi (*elaboration*)

Fase Elaborasi adalah untuk memitigasi risiko kunci yang diidentifikasi dari analisis hingga akhir fase. Fasa elaborasi merupakan fase saat proyek mulai terlihat bentuknya. Pada fase ini, masalah analisis domain dibuat dan arsitektur proyek mulai mendapatkan bentuk dasarnya.

3. Fase Konstruksi (*contruction*)

Fase Konstruksi adalah untuk membangun sistem perangkat lunak. Fase ini fokus pada pengembangan komponen dan fitur lain dari sistem. Pada fase inilah saat banyak dilakukan pengkodean. Pada proyek yang lebih besar, beberapa iterasi konstruksi dikembangkan sebagai usaha untuk memecah kasus penggunaan menjadi segmen ter kelola yang menunjukkan purwarupa.

4. Fase Transisi (*transision*)

Fase Transisi adalah sebagai perantara sistem dari pengembangan ke produksi, yang tersedia untuk pengguna akhir. Aktivitas dalam fase ini termasuk pelatihan kepada pengguna akhir dan pengelola sistem dan pengujian beta untuk memvalidasi terhadap harapan pengguna akhir

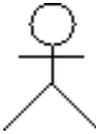
2.11 Bahasa pemodelan pengembangan sistem

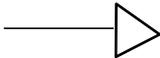
Menurut Booch (2011) “UML adalah bahasa standar untuk membuat rancangan software, UML biasanya digunakan untuk membangun dokumen dari software-intensive system”. UML merupakan bahasa standar yang sering kali dipakai untuk menjelaskan proses analisa dan desain sistem berorientasi objek. Pada taha analisa, UML sendiri mengembangkan model dari aplikasi yang ada di dunia nyata dengan memperlihatkan komponen-komponen penting yang bisa dijadikan prototipe. Pada tahap perancangan adalah bagaimana hasil analisa orientasi aplikasi direalisasikan pada lingkungan implementasi. Menurut Nugroho (2010), “UML adalah bahasa pemodelan untuk sistem untuk sistem berparadigma objek”. UML sendiri menyediakan diagram-diagram yang dapat membantu mendefinisikan dan membagi sebuah aplikasi. Adapun jenis diagramnya sebagai berikut:

2.11.1 Use Case Diagram

Diagram model *use case* menggambarkan interaksi antara sistem dengan sistem eksternal dan pengguna. Sebuah *use case* mempresentasikan satu tujuan tunggal dari sistem dan menggambarkan satu rangkaian kegiatan dan interaksi pengguna untuk mencapai tujuan (Whitten, 2004). Hal-hal yang berkaitan dengan diagram *usecase*, yaitu:simbol-simbol yang akan digunakan dalam menggambarkan *Use Case Diagram*:

Tabel 2.1 Simbol-Simbol *Use case Diagram*

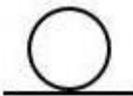
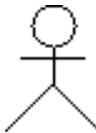
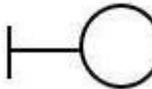
No	Simbol	Namasimbol	Deskripsi
1		<i>Usecase</i>	Menggambarkan fungsionalitas dari suatu sistem, sehingga pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun.
2		Aktor	Menggambarkan orang atau sistem yang menyediakan atau menerima informasi dari sistem atau menggambarkan pengguna <i>software</i> aplikasi(<i>user</i>).
3		Asosiasi/ <i>assosiati</i> <i>on</i>	Komunikasi antara aktor dan usecase yang berpartisipasi pada usecase dan memiliki interaksi dengan actor
4		Ekstend / <i>extend</i>	Relasi usecase tambahan kesebuah usecase, dimana usecase yang ditambahkan dapat berdiri sendiri walaupun tanpa Usecase tambahan

5		Generalisasi	Hubungan generalisasi dengan spesialisasi (umum-khusus) antara dua buah usecase dimana fungsi yang satu merupakan fungsi Yang lebih umum dari lainnya.
---	---	--------------	--

2.11.2 Sequence Diagram

Sequence diagram atau diagram urutan adalah sebuah diagram yang digunakan untuk menjelaskan dan menampilkan interaksi antar objek-objek dalam sebuah sistem secara terperinci. Selain itu *sequence diagram* juga akan menampilkan pesan atau perintah yang dikirim, beserta waktu pelaksanaannya. Objek-objek yang berhubungan dengan berjalannya proses operasi biasanya diurutkan dari kiri ke kanan. Berikut symbol-simbol yang menggambarkan *Sequence Diagram*:

Tabel 2.2 Simbol-Simbol *Sequence Diagram*

No	Simbol	Namasimbol	Deskripsi
1		<i>Entity Class</i>	Menggambarkan hubungan yang akan dilakukan
2		Aktor	Menggambarkan orang atau sistem yang menyediakan atau menerima informasi dari sistem atau menggambarkan pengguna <i>software</i> aplikasi(<i>user</i>).
3		<i>Boundary Class</i>	Menggambarkan sebuah gambaran dari foem

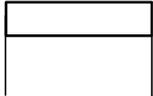
4		<i>Control Class</i>	Menggambarkan penghubung antara boundary dengan tabel
5		<i>A focus of control & a life line</i>	Menggambarkan tempat mulai dan berakhirnya message
6		<i>A message</i>	Menggambarkan pengiriman pesan

2.11.3 Activity Diagram

Activity diagram menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis atau menggambarkan aktivitas sistem bukannya apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem (Rosa dan Shalahuddin, 2019).

Tabel 2.3 Simbol-Simbol Activity Diagram

No	Simbol	Nama simbol	Deskripsi
1		Status awal	Status awal aktifitas sistem, sebuah diagram aktifitas memiliki sebuah status awal
2		Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
3		Percabangan/ <i>decision</i>	Asosiasi percabangan dimana jika ada pilihan aktifitas lebih dari satu
4		Pengabungan/ <i>join</i>	Asosiasi pengabungan dimana lebih dari satu aktivitas digabungkan

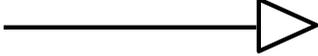
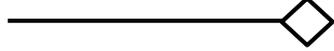
			menjadi satu
5		Status akhir	status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir
6		Swimlane	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktifitas yang terjadi

2.11.4 Class Diagram

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi (Rosa and Shalahudin, 2018). menjelaskan simbol-simbol yang ada pada diagram kelas pada tabel *class diagram 2.4*.

Tabel 2. 4 Simbol Class Diagram

Simbol	Deskripsi
Kelas 	Kelas pada struktur sistem
Antarmuka/ <i>Interface</i>  nama_interface	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek
Asosiasi/ <i>asociation</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i>

Asosiasi berarah/ <i>directed association</i> 	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya disertai dengan <i>multiplicity</i>
Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus)
Kebergantungan/ <i>dependecy</i> 	Relasi antar kelas dengan makna kebergantungan antar kelas
Agregasi/ <i>agregation</i> 	Relasi antar kelas dengan makna semua bagian (<i>whole-part</i>)

2.11.5 Skenario *use case*

Skenario Use Case mendeskripsikan aktor-aktor yang melakukan prosedur dalam sistem, serta menjelaskan respon yang ditanggapi oleh sistem tersebut terhadap prosedur yang dilakukan oleh aktor.

2.12 Pengujian Sistem *Black Box*

Pengujian *black-box* berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian *black-box* memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program (Howden, 2019). Pengujian *black-box* berusaha menemukan kesalahan dalam kategori sebagai berikut :

1. Fungsi – fungsi yang tidak benar atau hilang,
2. Kesalahan *interface*
3. Kesalahan dalam struktur data atau akses eksternal
4. Kesalahan kinerja
5. Inisialisasi dan kesalahan terminasi

Kelebihan *Black Box* adalah *black box testing* dapat menguji keseluruhan fungsionalitas perangkat lunak. *Black box testing* dapat memilih *subset test* yang secara

efektif dan efisien dapat menemukan cacat. Dengan cara ini *black box testing* dapat membantu memaksimalkan *testing investment* (Howden, 2019).

2.13 Penelitian Terdahulu

Nama peneliti	Judul
Andhika Widyanto, 2021	Penerapan Metode Rup Pada Sistem Informasi Unit Kegiatan Mahasiswa STMIK PalComTech,
Nova Indrayana Yusman,2019	Perancangan Sistem Informasi Berbasis Orientasi Objek Menggunakan Star Uml Di Cv Niasa Bandung
David Joshua Hutahaean, 2019	Pengembangan Sistem Informasi Penyewaan Gedung Berbasis Web dengan Metode Rational Unified Process (RUP) (Studi Kasus: Wisma Rata Medan),
DJ Hutahean ,NH Wardhani, 2019	Pengembangan Sistem Informasi Penyewaan Gedung Berbasis Web Dengan Metode Rup (Studi Kasus; Wisma Rata Medan),
I Seprina, E yulianingsih, 2022	Penerapan Metode RUP Untuk Sistem Pengajuan Cuti Pegawai Di SMA AZ-Zahra Palembang Berbasis Web Mobile,