

BAB II

TINJAUAN PUSTAKA

21 Teori Dasar

2.1.1 UML (Unified Modeling Language)

Pada era digitalisasi teknologi perangkat lunak, dibutuhkan adanya bahasa yang dapat digunakan untuk memodelkan perangkat lunak software yang akan dikembangkan dan dibutuhkan adanya standarisasi agar orang-orang di seluruh negara akan mengerti tentang permodelan perangkat lunak. Seperti apa yang kita ketahui yaitu menyatukan beberapa kepala untuk menceritakan sebuah gagasan dengan tujuan untuk mengerti hal yang sama bukanlah perkara mudah, oleh karena itu sangat diperlukan sebuah bahasa pemodelan perangkat lunak software yang dapat dipahami oleh seluruh orang (Shalahuddin, 2013).

Menurut Nugroho(2010:10), Sesungguhnya tidak ada batasan yang signifikan diantara berbagai konsep dan konstruksi yang ada pada UML, tetapi dibutuhkan untuk menyederhanakannya, kita membagi cukup banyak konsep dalam UML menjadi beberapa view.

View sendiri adalah sejumlah konstruksi pemodelan UML yang menggambarkan suatu aspek tertentu dari sebuah perangkat lunak yang sedang kita buat.

Pada peringkat tertinggi, view-view pasalnya dapat dibagi menjadi beberapa bagian area utama, yaitu adalah: klasifikasi struktural (structural classification), perilaku dinamis (dynamic behaviour), serta pembuatan atau manajemen model (model management).

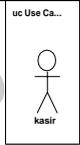
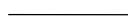
2.1.2 Use Case Diagram

Usecase diagram adalah pemodelan untuk tingkah laku (behavior) untuk sistem yang akan dikembangkan. Usecase diagram menrincikan sebuah interaksi antara satu atau lebih aktor pada sistem yang akan dikembangkan. Secara kasar, usecase difungsikan untuk mengetahui fungsi apa saja yang berada didalam sistem dan menjelaskan siapa saja yang perlu untuk menggunakan fungsi-fungsi yang berada didalamnya. Syarat untuk penamaan pada sebuah usecase diagram adalah nama yang didefinisikan semudah mungkin dan mudah dipahami.

Ada beberapa hal utama pada sebuah usecase diagram yaitu pendefinisian yang disebut aktor dan juga usecase (Shalahuddin, 2013).

1. Aktor adalah orang, proses, atau juga sistem yang berhubungan dengan sistem yang sedang dikembangkan diluar dari sistem yang akan dikembangkan itu sendiri, jadi meskipun simbol pada aktor adalah gambar orang, tetapi aktor belum tentu adalah orang.
2. UseCase adalah fungsionalitas yang telah tersedia pada sistem sebagai unit-unit yang saling mentransfer pesan antar unit atau lainnya.

Table 2.1 Use Case Diagram

NO	SIMBOL	NAMA	KETERANGAN
1.		<i>Use Case</i>	Fungsionalitas yang disediakan oleh sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i> .
2.		Aktor (<i>Actor</i>)	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun Simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal frase nama aktor.
3.		Asosiasi (<i>Association</i>)	Komunikasi antara aktor dan use case yang berpartisipasi pada use case memiliki interaksi dengan

			aktor.
4.	<pre><<extend>> ←</pre>	Ekstensi (Extend)	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walaupun tanpa use case tambahan itu mirip dengan prinsip inheritance pada pemrograman berorientasi objek biasanya use case tambahan memiliki nama depan yang sama dengan use case yang ditambahkan.
5.	<pre>←</pre>	Generalisasi	Hubungan dengan generalisasi dan spesialisasi (umum -khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6.	<pre><<include>> - - - - -></pre>	Menggunakan (Include)	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya atau sebagai syarat dijalankan use case ini

2.1.3 Activity Diagram

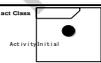
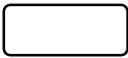
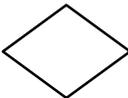
Diagram aktivitas atau juga disebut activity diagram mendefinisikan work flow (aliran kerja) dari sebuah sistem atau proses bisnis yang ada pada perangkat lunak. Yang perlu dipahami disini adalah diagram aktivitas menggambarkan aktivitas sistem yang bukan dikerjakan aktor, jadi aktivitas yang dapat dikerjakan oleh sistem (Shalahuddin, 2013). Diagram aktivitas juga banyak digunakan untuk menggambarkan hal-hal berikut :

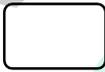
1. Rancangan proses bisnis yang dimana setiap urutan dari aktivitas yang digambarkan merupakan proses bisnis sistem yang telah terdefiniskan.

2. Urutan atau juga pengelompokan tampilan dari system atau *user interface* yaitu pada setiap aktivitas dianggap mempunyai sebuah rancangan interface tampilan.
3. Rancangan testing dimana seluruh aktivitas dianggap membutuhkan sebuah testing yang perlu didefinisikan kasus ujinya.
4. Rancangan menu yang digambarkan pada perangkat lunak.

Berikut adalah simbol-simbol yang ada pada diagram aktivitas (Shalahuddin,2013) :

Tabel 2.1 Activity Diagram

NO	SIMBOL	NAMA	DESKRIPSI
1.		Status awal (<i>Start Point</i>)	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.		Aktivitas (<i>Activities</i>)	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.		Percabangan (<i>Decision</i>)	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.		Penggabungan (<i>Join</i>)	Menunjukkan adanya penggabungan aktivitas.
5.		Percabangan (<i>Fork</i>)	Menunjukkan adanya percabangan paralel dari aktivitas.
6.		Status akhir(<i>End Poin</i>)	Status akhir yang dilakukan system sebuah diagram aktivitas memiliki sebuah status akhir

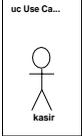
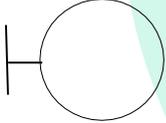
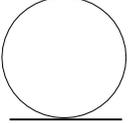
7.		<i>Swimlane</i>	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.
8.		<i>Black hole activity</i>	Ada masukan dan tidak ada keluaran, biasanya digunakan jika dikehendaki ada 1 atau lebih transisi.
9.		<i>Miracle activity</i>	Tidak ada masukan dan ada keluaran, biasanya dipakai pada waktu start point dan dikehendaki ada 1 atau lebih transisi.

2.1.4 Sequence Diagram

Sequence diagram menjelaskan tentang tingkah laku objek pada sebuah usecase dengan menjelaskan waktu hidup sebuah objek dan pesan yang dikirimkan dan diterima antar sebuah objek. Maka dari itu untuk membuat diagram sequence maka wajib diketahui objek-objek yang termasuk dalam sebuah usecase beserta seluruh metodenya yang mempunyai kelas yang diinstansiasi menjadi objek itu. Membuat diagram sequence juga membutuhkan untuk melihat skenario yang dalam usecase (Shalahuddin, 2013).

Banyaknya sequence diagram yang perlu untuk dibuat adalah sebanyak jumlah definisian usecase yang dimiliki dari proses itu sendiri atau yang mencakup seluruh usecase yang pernah didefinisikan interaksinya jalannya pesan telah dicakup pada sebuah sequence diagram sehingga membuat semakin banyak use case yang dibuat, maka sequence diagram perlu juga semakin banyak. Berikut adalah simbol-simbol yang terdapat diagram sequence (Shalahuddin, 2013):

Tabel 2. 3 Sequence Diagram

1.		Aktor (<i>Actor</i>)	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, Tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor.
2.		<i>Boundary</i>	<i>Boundary</i> digunakan Dalam tahap yang Konseptual untuk menangkap para pemakai Yang saling berinteraksi dengan sistem pada suatu Tingkatan layar atau beberapa alat penghubung jenis lain
3.		<i>Control</i>	<i>Control</i> adalah element yang mengatur aliran dari Informasi untuk sebuah scenario. Perilaku dan Prilaku bisnis umumnya diatur oleh objek ini.
4.		<i>Entity</i>	<i>Entity</i> biasanya element Yang bertanggung jawab Menyimpan informasi. Dapat berupa beans atau object.
5.		Garis kehidupan <i>lifeline</i>	Menyatakan kehidupan suatu objek

6.		Objek	Menyatakan objek yang berinteraksi pesan.
	nama_objek : nama_kelas		
7.		Waktu aktif	Menyatakan objek dalam Keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya.
8.	<<create>>	Pesan tipe <i>create</i>	Menyatakan suatu objek membuat objek lain, arah panah mengarah pada objek yang dibuat.
9.	1:nama_metode(:)	Pesan tipe <i>call</i>	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.
10.	1 : masukan	Pesan tipe <i>send</i>	Menyatakan suatu objek mengirimkan data atau masukan atau informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
11.	1 : keluaran	Pesan tipe <i>return</i>	Menyatakan suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah mengarah pada objek yang menerima kembalian.

12.	<<destroy>>	Pesan tipe destroy	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada destroy.
-----	-------------	-----------------------	--

2.1.5 Website

Website adalah fasilitas yang ada di internet yang menyambungkan data dalam ruang lingkup lokal ataupun jarak jauh. Data pada website disebut dengan web page lalu link didalam website memungkinkan user bisa berpindah dari halaman ke halaman lain (hyper text), baik diantara halaman yang disave dalam sebuah server yang sama ataupun server yang berbeda diseluruh dunia. Halaman diakses ataupun dibaca melalui web browser contohnya Mozila Firefox, Google Chrome dan browser lainnya

2.1.6 Bahasa Pemrograman

Menurut (Dipraja, 2013) “Programming language (bahasa pemrograman) merupakan program dalam bentuk-bentuk perintah untuk berkomunikasi dengan komputer, bagaimana data akan diolah, disimpan, dijalankan. Beberapa contoh bahasa pemrograman yaitu, PHP, Python, SQL, Javascript, Kotlin, Go, Ruby, dan lain sebagainya.

2.1.7 Database

Menurut (Setiyowati; Siswanti, 2021)dalam bukunya Perancangan Basis Data & Pengenalan SQL Server Managemen Studio (2021), “database (basis data) adalah media penyimpanan berbentuk kumpulan tabel-tabel teratur secara srtuktural, sehingga dapat dipanggil memakai suatu program komputer terhadap kumpulan data-data tersebut”.

2.1.8 Manajemen Proyek

Manajemen proyek merupakan proses perencanaan, mengatur, memimpin, dan mengontrol sumber daya perusahaan untuk dapat mencapai sasaran telah ditentukan. Manajemen proyek berkembang karena dorongan mencari pengelolaan yang sesuai dengan keharusan dan sifat kegiatan proyek, suatu kegiatan akan dinamis dan berbeda dengan adanya kegiatan operasional rutin (Rani, 2016)

2.1.9 Black Box Testing

Metode black box testing adalah pengujian program yang dianggap sebagai suatu yang disebut “black box” (kotak hitam). Program test case berbasiskan sebuah spesifikasi test perencanaan dapat dilakukan sejak awal pembuatan sistem.

Black box testing bertujuan pada prasyarat fungsional perangkat lunak software. Metode ini berkesempatan seorang software developer untuk menghasilkan serangkaian kondisi input yang menggunakan semua persyaratan fungsional pada sebuah program. Black box testing bukanlah alternatif dari metode white box testing, namun black box testing adalah pelengkap yang bisa mengungkap suatu kesalahan, jika diukur dengan metode white box testing.

Pengujian black box testing berusaha mencari kesalahan dalam kategori :

1. Fungsi yang error atau missing.
2. Kesalahan pada interface.
3. Kesalahan pada struktur data atau pada akses ke sebuah database eksternal.
4. Kesalahan pada kinerja program.
5. Kesalahan inisialisasi dan terminasi.

Black box testing lebih sering dilakukan selama tahap final pengujian. Pengujian black box testing memperlihatkan struktur kontrol, sehingga berfokus pada domain data. Black box testing dikembangkan untuk bisa menjawab pertanyaan berikut :

1. Bagaimana validitas fungsional dapat diuji dengan maksimal?
2. Kelas input bagaimana yang dapat menghasilkan *test case* jadi lebih baik?

22 TINJAUAN STUDI

Peneliti dalam membuat tulisan ini melakukan riset dari beberapa penulis sebelumnya untuk dijadikan bahan acuan dalam menyusun laporan ini. Riset yang telah dijalankan oleh penulis adalah dengan membaca beberapa sumber referensi yang terkait dengan judul laporan yang dibuat. Berikut beberapa referensi yang dijadikan acuan untuk mendukung dalam penelitian ini, yaitu ;

- Oleh Andi Megawati, dengan judul *“Membangun Sistem Informasi Monitoring Kegiatan Proyek Pemancar Sinyal BTS Berbasis Web Pada PT.Swatama Mega Teknik”*. Pada sebuah jurnal Seminar Nasional Teknologi Informasi dan Komunikasi, ISSN: 2085-4315. Jakarta, 2018

Sistem Monitoring Proyek yang dibuat sebelumnya pada perusahaan PT. Swatama Mega Teknik yang masih dengan cara manual menggunakan Excel dan dengan dibuatnya sistem informasi monitoring berbasis web ini diharapkan mampu membantu penginputan data suatu proyek dan pemantauan suatu proyek BTS sehingga dapat mempersingkat waktu dan meminimalisir biaya. Pembangunan sistem informasi monitoring berbasis web ini dibuat dengan menggunakan metode pengembangan Web Development Life Cycle (WDLC) dan di implementasikan dengan cara menggunakan template framework .Sistem monitoring proyek berbasis web ini diharapkan dapat membantu dalam pengolahan data proyek BTS menjadi lebih baik.

- Oleh GEVRIE ARCHRONANTHA, dengan judul *“Sistem Monitoring Proyek Dalam Bidang Konstruksi Berbasis Web Di PT.WARYCORP”* Universitas Komputer Indonesia Bandung , 2017

Sistem Monitoring Proyek sebelumnya yang dibuat pada perusahaan PT.WARYCORP masih dilakukan manual tidak tertata dengan baik mengenai kegiatan suatu proyek maupun data-data dari setiap monitoring suatu proyek dikarena data-data masih dalam media kertas sehingga sulit untuk dicari dan juga data-data tersebut rentan mengalami kehilangan dan kerusakan data. Dengan dibuatnya sistem monitoring proyek berbasis web ini dapat menjadi media alternatif tambahan dalam penginputan data

Dari suatu proyek serta pemantauan kegiatan proyek dan penyimpanan data menjadi lebih cepat dan lebih aman. Dalam pengembangan sistem informasi monitoring proyek berbasis web dalam bidang konstruksi ini pada PT.WARYCORP metode yang penulis gunakan adalah metode pendekatan deskriptif. Maka dengan menggunakan metode pendekatan deskriptif akan menggambarkan proses sistem informasi monitoring proyek berbasis web ini dalam bidang konstruksi pada perusahaan PT.WARYCORP,

- Oleh Moh Ikhsan Saputro, Toni Sukendar, Aziz Setyawan H dengan judul “ *Sistem Informasi Monitoring Pengembangan Proyek Berbasis Web Pada PT.Wahana Reka Tekindo, Jakarta*” Pada Jurnal Teknologi Informatika dan Komputer Vol 4, No.1, 2018

Sistem Monitoring proyek yang dibuat sebelumnya pada PT.Wahana Reka Tekindo belum menggunakan Sistem informasi berbasis website yang aman dan dinamis sehingga sang manager proyek dibatasi dengan ketersediaan waktu untuk memonitoring suatu proyek setiap harinya hal ini dapat membuat pelaksanaan pemantauan dan koordinasi dalam proyek tidak berjalan dengan maksimal. Pembuatan sistem monitoring berbasis web ini dibuat dengan menggunakan metode SDLC waterfall dan di implementasikan untuk dapat berjalan dengan semestinya.

- Oleh Yoseph Tajul Arifin dengan judul “*Rancang Bangun Sistem Informasi Monitoring Perkembangan Proyek Konstruksi Pada Perum Perumnas Jakarta Berbasis Web*” pada Seminar Nasional Inovasi dan Tren,2018

Permasalahan yang dihadapi oleh monitoring yang terjadi pada saat ini adalah sulitnya untuk perkembangan proyek secara real time sehingga membuat data tidak bisa diberikan tepat waktu, proses penulisan data dan laporan proyek masih harus melalui sederet proses yang panjang.Sama halnya dengan bukti-bukti data perkembangan dari suatu proyek, berupa surat, maupun bukti-bukti yang berhubungan lain yang tidak terorganisir dengan baik karena tidak adanya suatu sistem informasi untuk memproses dan menyimpan data dari perkembangan suatu proyek Perum Perumnas. Pembangunan sistem monitoring berbasis web ini dikembangkan dengan menggunakan metode SDLC waterfall.

- Oleh Muhammad Wahyu Firmansyah dengan judul “*PEMBUATAN SISTEM INFORMASI MONITORING PROYEK BERBASIS WEB BAGI MANAJEMEN PERUSAHAAN PENGEMBANGAN SOFTWARE*” Tugas akhir pada Universitas Islam Indonesia, 2019

Perusahaan software house yang biasa menangani pembuatan, pengembangan, maupun perbaikan sistem, tidak selalu dikepalai atau di pimpin oleh orang yang berlatar belakang IT. Sehingga penyampaian informasi tentang perkembangan proyek susah dimengerti dengan penjelasan teknis. Informasi yang dibutuhkan oleh pimpinan berhubungan dengan keputusan yang akan dibuat demi keberlangsungan perusahaan maupun sumber daya manusia yang terlibat didalamnya. Belum lagi keterbatasan jarak dan waktu yang tidak selalu memungkinkan pemimpin untuk setiap hari dapat melakukan meeting dengan pihak teknis terkait perkembangan proyek yang menjadi salah satu kendala pimpinan dalam membuat keputusan. Hal tersebut juga menjadi kendala bagi divisi marketing untuk menjawab pertanyaan client terkait perkembangan sistem yang mereka pesan, dan juga menjadi kendala bagi marketing untuk mendapatkan informasi budget untuk proyek yang diminta client ataupun proyek yang akan ditawarkan oleh marketing perusahaan. Pembuatan sistem monitoring berbasis web ini dibuat dengan menggunakan paradigma waterfall sebagai metodenya.