

BAB II

KAJIAN LITERATUR

Pada kajian literatur ini berisi tentang teori pendukung dalam pengembangan modul evaluasi mahasiswa pada aplikasi web bimbingan akademik pada penelitian ini.

2.1 Bimbingan Akademik

Bimbingan akademik adalah kegiatan konsultasi antara pembimbing akademik dan mahasiswa dalam merencanakan studi dan membantu menyelesaikan masalah studi yang dialami, agar mahasiswa yang bersangkutan dapat menyelesaikan studinya dengan baik sesuai dengan minat dan kemampuannya pelaksanaan bimbingan akademik diatur sesuai oleh tiap fakultas (UAJY, 2016).

Dalam buku pedoman akademik (STT, 2016) dijelaskan, bimbingan akademik setiap mahasiswa ditetapkan seorang dosen pembimbing akademik yang berperan dalam memberikan bimbingan kepada mahasiswa bimbingannya agar lancar dalam perancangan studi tiap semester. Dosen pembimbing akademik bertugas memberikan pertimbangan kepada mahasiswa dalam menentukan jumlah SKS dan mata kuliah yang akan diambil tiap semesternya. Selain itu tugas pembimbing akademik untuk memantau dan membantu perkembangan akademik mahasiswa bimbingannya. Dan membantu memecahkan masalah akademik dan non-akademik yang dihadapi mahasiswanya. Serta melaporkan kepada ketua program studi jika mahasalah mahasiswa yang membutuhkan penanganan khusus.

2.2 Aplikasi Web Bimbingan Akademik

Berikut adalah tabel modul dan sub modul aplikasi bimbingan akademik yang telah dikembangkan pada penelitian sebelumnya. Modul dan sub modul di bawah merupakan gabungan dari beberapa *role* yang ada dalam aplikasi.

TABEL 2. 1 MENU DAN SUB MENU APLIKASI YANG TELAH DIKEMBANGKAN

MENU	SUB MENU
Home	
Dashboard	
Artikel	Dosen
Profil	Mahasiswa
	Dosen
Evaluasi Mahasiswa	Keuangan
	Akademik
	Presensi
Bimbingan Akademik	
User	

Pada tabel 2.1 menunjukkan menu dan sub menu yang ada pada Web Bimbingan Akademik. Kolom yang berwarna biru menunjukkan modul yang telah selesai dikembangkan pada penelitian sebelumnya, yaitu modul home, artikel, profil, bimbingan akademik, dan user. Dan tabel yang berwarna oranye menunjukkan modul yang belum terselesaikan pada penelitian sebelumnya, yaitu modul dashboard dan evaluasi mahasiswa.

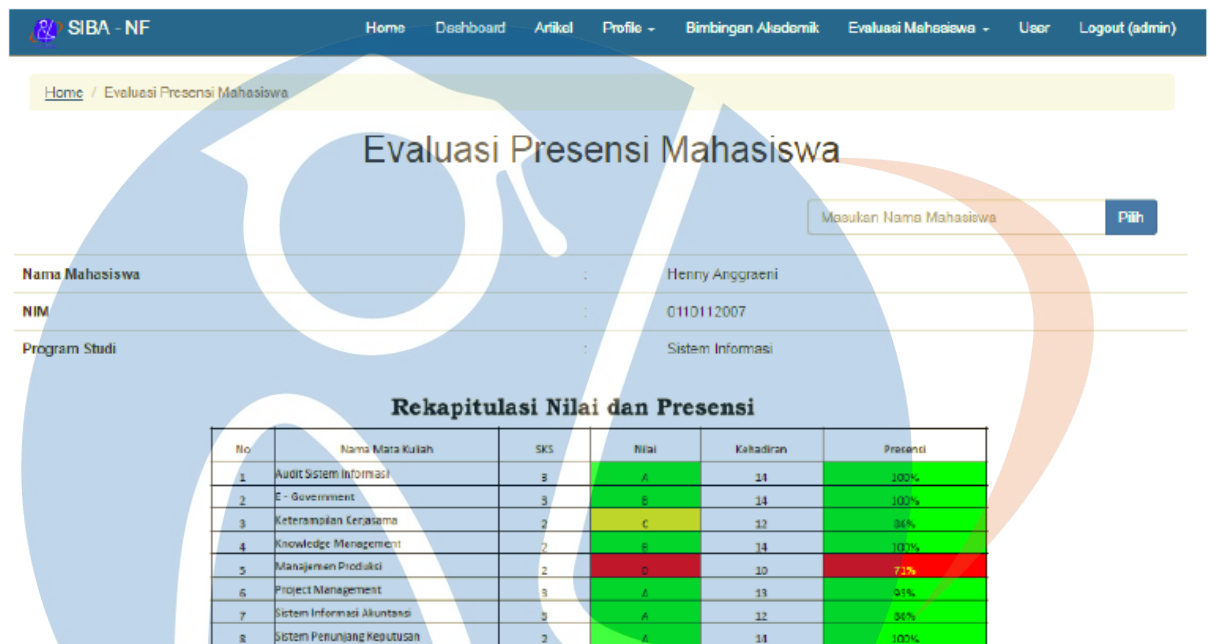
2.2 Modul Evaluasi Mahasiswa

Modul evaluasi mahasiswa dibuat untuk membantu dosen untuk membantu proses berlangsung bimbingan akademik. Modul evaluasi mahasiswa memiliki sub modul akademik, presensi, dan keuangan. User yang bisa mengakses modul evaluasi mahasiswa ini adalah admin, dosen pembimbing, dan mahasiswa.

User admin dapat melihat ke tiga sub modul dari modul evaluasi mahasiswa. User dosen pembimbing dapat melihat grafik perkembangan IP mahasiswa bimbingannya dari semester awal hingga semester akhir. Dosen pembimbing juga dapat melihat status presensi dan keuangan mahasiswanya, serta dapat melakukan eskalasi masalah ke ketua kaprodi. User mahasiswa dapat melihat perkembangan akademiknya (Anggraeni, 2015).

2.2.1 Sub Modul Evaluasi Presensi Mahasiswa

Pada halaman ini informasi presensi mahasiswa di tiap perkuliahan yang diambil. User harus meng-*input* nama mahasiswa pada bagian kanan atas agar mengetahui presentase kehadiran dalam kelas pada tiap mata kuliahnya.

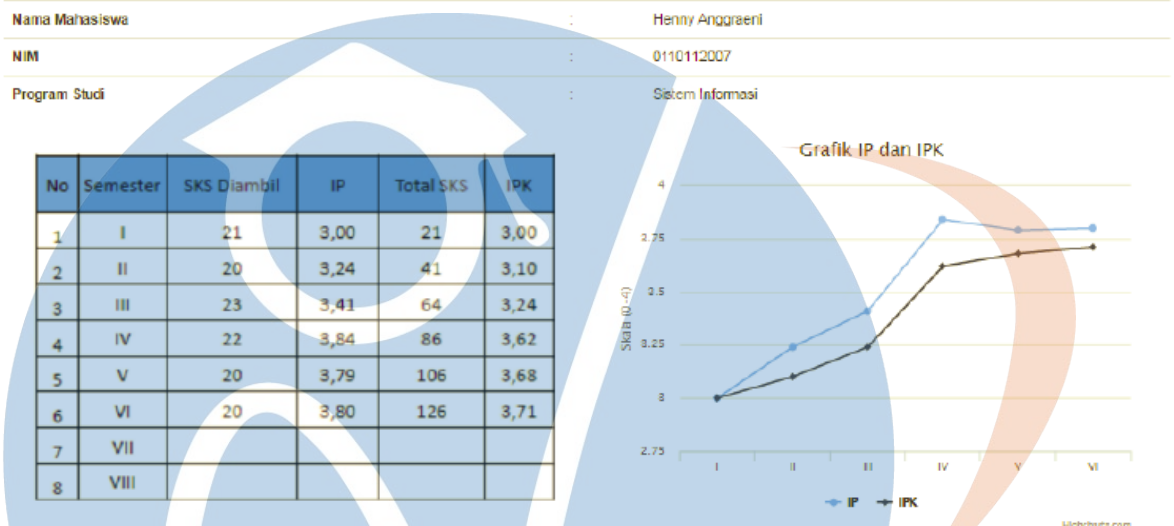


GAMBAR 2. 1 SUB MODUL EVALUASI PRESENSI MAHASISWA

2.2.2 Sub Modul Evaluasi Akademik Mahasiswa

Dihalaman ini memberikan informasi perkembangan akademik mahasiswa berupa grafik IP di setiap semester, sehingga dosen pembimbing dan mahasiswa dapat melihat perkembangan akademiknya dari awal memiliki IP hingga lulus.

Evaluasi Akademik



GAMBAR 2. 2 SUB MODUL EVALUASI AKADEMIK MAHASISWA

2.2.3 Sub Modul Evaluasi Keuangan Mahasiswa

Pada sub modul ini dapat memberikan informasi kepada dosen pembimbing terkait status keuangan mahasiswa bimbingannya.



GAMBAR 2. 3 SUB MODUL EVALUASI KEUANGAN MAHASISWA

2.3 Software Requirement Specifications (SRS)

Software Requirement Specifications (SRS) dokumen yang berisi berbagai kebutuhan *software* yang akan dibangun. Dokumen dibuat oleh *developer* setelah menggali informasi dari calon pemakai *software*. Pembuatannya pun sesuai standar dari IEEE. Agar *developer* memahami keinginan calon user terkait *software* yang diinginkan. Dokumen SRS selanjutnya sebagai bahan dalam pembuatan desain UML *software* yang akan dibuat (Irawanto, 2014).

a. Fungsional

Pernyataan layanan tentang bagaimana sistem harus bereaksi terhadap input, sistem harus berlaku pada situasi-situasi tertentu. Secara khusus menyatakan apa yang tidak boleh dilakukan sistem. Merupakan penjelasan tentang layanan yang perlu disediakan oleh sistem, bagaimana sistem menerima dan mengolah masukan, dan bagaimana sistem mengatasi situasi – situasi tertentu. Selain itu kadang – kadang juga secara jelas menentukan apa yang tidak dikerjakan oleh sistem. *Functional Requirement* menggambarkan sistem requirement secara detail seperti input, output dan pengecualian yang berlaku

b. Non Fungsional

Pernyataan tentang batasan layanan dan fungsi yang diberikan sistem. Karena berkaitan dengan kebutuhan sistem secara keseluruhan, maka kegagalan memenuhi kebutuhan jenis ini berakibat pada sistem secara keseluruhan. Contoh kebutuhan jenis ini adalah kecepatan akses, keamanan data, besarnya kapasitas penyimpanan yang diperlukan, privasi masing – masing profil/account, bahasa pemrograman yang digunakan, sistem operasi yang digunakan.

2.4 UML (*Unified Modelling Language*)

UML adalah singkatan dari *Unified Modelling Language*, yaitu sebuah metode permodelan secara visual untuk sarana perancangan sistem berorientasi objek. UML merupakan suatu bahasa standar pada visualisasi, perancangan dan

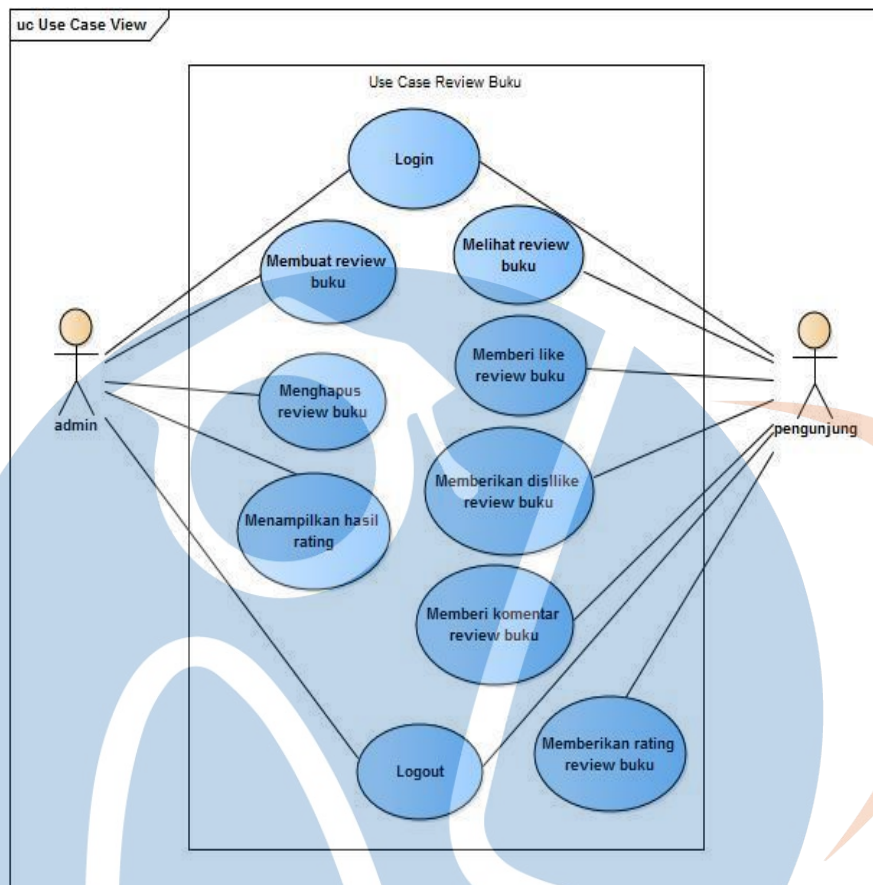
juga pendokumentasian sistem software. Saat ini UML menjadi bahasa standar dalam penulisan *blue print software*.

Dengan menggunakan UML kita dapat membuat model untuk semua jenis *software*, dimana aplikasi tersebut dapat berjalan pada *hardware*, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML juga menggunakan *class* dan *operation* dalam konsep dasarnya, maka ia lebih cocok untuk penulisan *software* dalam Bahasa pemrograman berorientasi objek seperti C++, Java, C# atau VB.NET. Walaupun demikian, UML tetap dapat digunakan untuk modeling aplikasi prosedural dalam VB atau C.

1. Usecase Diagram

Menggambarkan fungsionalitas yang diharapkan dari sebuah sistem, yang ditekankan adalah “apa” yang diperbuat sistem dan bukan “bagaimana”. Sebuah *usecase* merepresentasikan sebuah interaksi antar aktor dengan sistem. *Usecase* merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-*create* sebuah daftar belanja, dan sebagainya. Aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan tertentu. *Usecase* akan sangat membantu bila kita sedang menyusun *requirement system*, mengomunikasikan rancangan dengan *client*, dan merancang *test case* untuk semua fitur yang ada pada sistem. (Wahono, 2003).

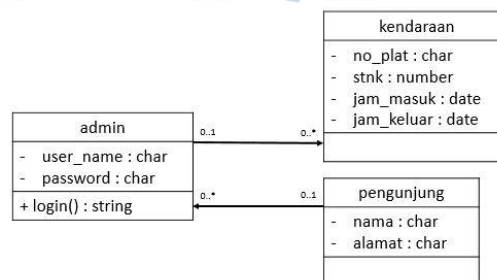
STT - NF



GAMBAR 2. 4 USE CASE DIAGRAM

2. Class Diagram

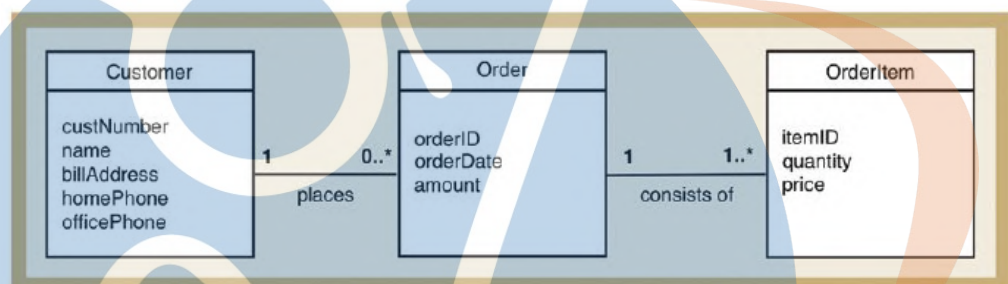
Memberikan pandangan secara luas dari suatu sistem dengan menunjukkan kelas-kelasnya dan hubungan mereka. Diagram Class bersifat statis; menggambarkan hubungan apa yang terjadi bukan apa yang terjadi jika mereka berhubungan.



GAMBAR 2. 5 CONTOH CLASS DIAGRAM

3. Domain Model

Sebuah domain model mengilustrasikan konsep-konsep dalam *problem* domain, dimana diagram ini mewakili entitas (objek atau konsep) di dunia nyata, dan bukan komponen *software* melainkan kumpulan dari diagram yang *static*, dimana tidak operasi yang didefinisikan didalamnya. Domain model menunjukkan konsep-konsep, keterhubungan antara konsep dan atribut-atribut dari konsep.

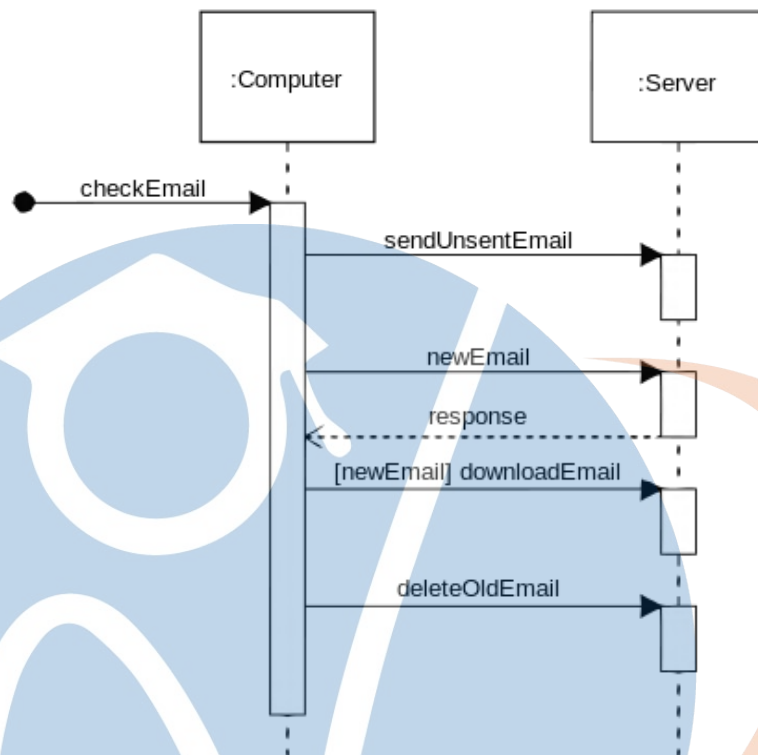


GAMBAR 2. 6 DOMAIN MODEL

4. Sequence Diagram

Sequence diagram merupakan salah satu diagram *interaction* yang menjelaskan bagaimana suatu operasi itu dilakukan *message* (pesan) apa yang dikirim dan kapan pelaksanaannya. Diagram ini mengatur berdasariakn waktu. Objek-objek yang berkaitan dengan proses berjalannya operasi diurutkan dari kiri ke kanan berdasarkan waktu terjadinya dalam pesan yang terurut.

STT - NF



GAMBAR 2. 7 SEQUENCE DIAGRAM

2.5 Framework Yii 2

Yii adalah kerangka kerja (Framework) open source berbasis php, berupa sekumpulan kode dan library PHP yang membentuk sebuah kerangka dasar untuk membangun berbagai benjenis aplikasi berbasis web, misalnya: web portal, forum, CMS, e-commerce, web sevice, dan lain-lain. Nama Yii singkatan dari “Yes It Is”. Sama seperti framework PHP pada umumnya, Yii juga telah mengadopsi konsep MVC (Model, View, Controller) dalam stuktur pemogramannya. Proyek pengembangan Yii dimulai sejak 1 Januari 2008 oleh Qiang Xue programmer asal Cina. Pada awalnya Yii dikembangkan menggunakan bahasa kerja PRADO framework, setelah mengalami beberpa beberapa kodisi dan penyempurnaan kurang lebih setahun, pada 3 Desember 2008, Yii 1.0 secara resmi dirilis ke publik.

Untuk menjalankan aplikasi web berbasis Yii, maka diperlukan web server yang mendukung PHP 5.1.0. Bagi para developer yang menggunakan Yii, maka wajib mengerti pemograman berorientasi objek (OOP) sehingga akan sangat

membantu dalam membuat project berbasis web, karena Yii merupakan framework OPP murni. Yii adalah framework pemograman umum web yang bisa dipakai untuk mengembangkan semua jenis aplikasi web. Dikarenakan sangat ringan dan dilengkapi dengan mekanisme caching yang canggih.

Yii adalah MCV framework. Kekayaan fitur Yii dan kejelasan dokumentasi. Yii di desain dengan cermat dari awal agar sesuai pengembangan aplikasi web secara serius. Yii adalah hasil dari pengalaman para pembuat pada pengembangan aplikasi web dan investigasi framework pemograman web dan aplikasi yang paling populer.

Yii mengimplementasi pola desain Model – View – Controller (MVC), yang diadopsi secara luas dalam pemograman web. Model – View – Controller bertujuan untuk memisahkan logika bisnis dari pertimbangan antar muka pengguna agar para developer bisa lebih mudah mengubah setiap bagian tanpa mempengaruhi yang lain. Dalam Model – View – Controller, model model menggambarkan informasi (data) dan aturan bisnis; view (tampilan) berisi elemen antar muka pengguna seperti teks, input form; controller mengatur komunikasi antar model dengan view.

Selain implementasi Model – View – Controller, Yii juga memperkenalkan front-controller, aplikasi mengumpulkan beberapa informasi mengenai request pengguna dan kemudian mengirimnya ke controller yang sesuai untuk penganan selanjutnya.

Dalam yii2 juga terdapat fitur Role – Base – Access – Control (RBAC) yaitu untuk membatasi web dari penggunaan tertentu. Autentikasi adalah meverifikasi sebelum mengakses sesuatu. Autorisasi merupakan pencarian apakah user tersebut mempunyai peran yang memiliki hak akses ke sumber daya.

2.6 PostgreSQL

PostgreSQL adalah Object-Relational Database Management System (ORDBMS) yang dikembangkan pada Universitas California pada Berkeley

Computer Science Departement. PostgreSQL memiliki lisensi GPL (General Public License) oleh karena itu PostgreSQL dapat digunakan, dimodifikasi dan didistribusikan oleh setiap orang dengan gratis tanpa harus membayar lisensi.

PostgreSQL dirancang dapat berjalan pada platform UNIX-Like, kemudian dirancang menjadi portable sehingga dapat berjalan diberbagai platform seperti Windows dan MacOS X. Database server yang mendukung sistem besar, berkinerja tinggi, dan mudah perawatan dan pengembangan.

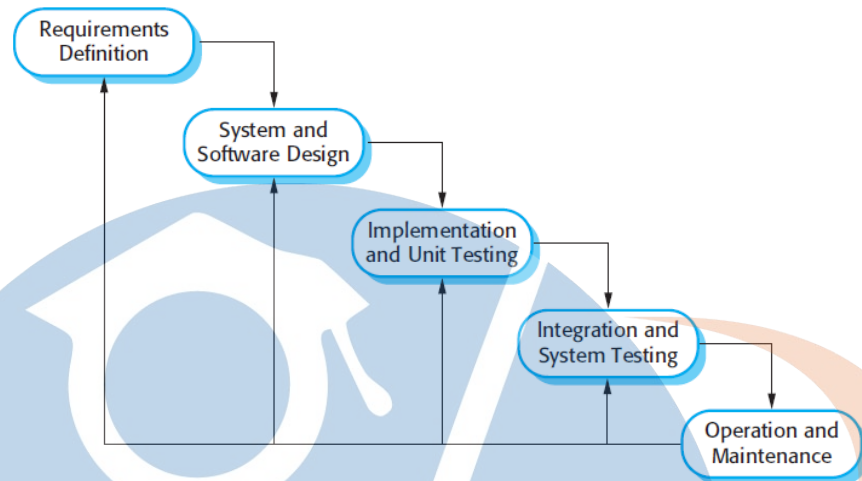
Fitur yang dimiliki Postgresql ialah *Multi-Version Concurrency Control* (MVCC) dimana user diberi data *snapshot* ketika suatu perubahan dilakukan sampai *commit*, *table inheritance* yaitu dimana satu tabel dapat diturunkan model kebeberapa karakteristik table lain, *rules* dimana suatu *query* DML yang dikirimkan ke server akan mengalami kesulitan penulisan ulang (*rewrite*), ini terjadi sebelum diproses oleh *query planner*, titik waktu pemulihan, *tablespace*, permintaan perencanaan/ optimizer canggih menulis depan logging untuk toleransi kesalahan dan lain-lain. Mendukung set karakter internasional, karakter multybyte ,pengkodean, unicode, case sensitive, dan format.

PostgreSQL dirancang untuk dikembangkan, dapat menentukan sendiri jenis data jenis index, bahasa fungsional, dan lain-lain. Dapat juga mengembangkan custom plugin untuk memenuhi kebutuhan pemakainya.

2.7 Waterfall

Waterfall model merupakan sebuah metode pengembangan *software* yang pengerjaannya dilakukan secara berurutan atau sekuensial. Jika tahapan sebelumnya belum selesai atau belum sempurna apalagi belum dilakukan, maka tahapan selanjutnya belum bisa dikerjakan. Secara otomatis tahapan ke 4 belum bisa dikerjakan bila tahapan ke 3 belum dilakukan.

Tahapan metode waterfall menurut Pressman



GAMBAR 2. 8 BAGAN MOTODE WATERFALL MENURUT PRESSMEN

Tiap tahapan di diatas akan dijelaskan di bawah ini:

1. *Requirement Definition*

Metode ini diawali dengan proses pendefinisian kebutuhan dan pengumpulan kebutuhan dari sistem yang akan dibangun. Mencari data dan informasi sebanyak-banyaknya dari berbagai sumber. Tahapan ini akan menghasilkan sebuah dokumen *user requirement* dimana keinginan user dalam pembuatan sistem.

2. *System and Software Design*

Pada tahapan ini berfokus pada sebuah proses perancangan stuktur data, arsitektur perangkat lunak, representasi *interface*, dan detail (algoritma) prosedural untuk persiapan sebelum memulai *coding*.

3. *Implementation and Unit Testing*

Disini proses coding dilakukan untuk dapat membuat program untuk dapat dimengerti *device*, dengan menggunakan bahasa pemograman. Setelah itu dilakukanlan proses *testing*. Pada tahap testing ini dilakukan per *unit* untuk menemukan kesalahan-kesalahan pada sistem dan kemudian bisa diperbaiki.

4. *Integration and System Testing*

Pada tahapan ini bisa juga dikatakan tahapan final. Sebelumnya proses analisis kebutuhan *user*, *design*, *coding*, *unit testing* dan penyelesaiannya. Setelah itu *system testing* dimana keseluruhan sistem yang telah dibangun diuji.

5. *Operation and Maintenance*

Sistem siap dioperasikan lalu setelah itu proses *maintenance* dilakukan. Proses pengembangan juga bisa dilakukan pada tahapan ini, karena sistem yang dibangun sekarang tidak selamanya hanya seperti ini.

2.8 *User Acceptance Test (UAT)*

Proses pengujian oleh *user* sebagai bukti dokumentasi bahwa sistem yang telah dikembangkan dapat diterima oleh *user*. Apabila hasil *testing* sudah bisa dianggap memenuhi kebutuhan *user*. Tahapan testing dengan UAT ini didasarkan pada dokumen *requirement* yang sebelumnya telah dibuat.

Pengujian dilakukan terhadap hasil pekerjaan. Diperiksa apakah *item-item* yang ada dalam dokumen *requirement* sudah ada dalam sistem yang diuji atau belum. Dan diuji apakah semua item dapat memenuhi kebutuhan *user*.

2.9 *Black Box Testing*

Pengujian yang dilakukan oleh pengembang berdasarkan apa yang harus sistem lakukan. Black box testing juga bisa disebut pengujian fungsional. Pengujian sistem tanpa memperhatikan struktur logika internal software. Pengujian ini digunakan untuk mengetahui apakah software yang dikembangkan berfungsi dengan benar.

