

BAB II

LANDASAN TEORI

2.1 Kajian Pustaka

2.1.1 Rancangan Aplikasi E-Recruitment Karyawan Dengan Metode Waterfall

Permasalahan tentang proses rekrutmen dan seleksi karyawan di PT BFI Finance Indonesia Tbk Cabang Meruya Jakarta masih menggunakan metode konvensional atau manual dan menggunakan kertas. Hal tersebut menimbulkan beberapa masalah seperti data pelamar yang mudah hilang dan rusak, butuh banyak tempat penyimpanan dokumen, diperlukan waktu yang lama dalam melakukan proses rekrutmen dan seleksi karyawan, dan juga adanya kesulitan dalam pengadaan kelas dan pengawas untuk seleksi.

Penerimaan merupakan sebuah metode perekrutan para calon tenaga kerja baru pada perusahaan dengan melewati segala tahapan-tahapan yang telah diberikan oleh perusahaan dan menggunakan media komunikasi elektronik modern seperti internet, sehingga perekrutan dapat dilaksanakan dengan lebih menghemat waktu dan tenaga guna mendapatkan tenaga kerja yang sesuai dengan kebutuhan perusahaan.

Metode penelitian yang digunakan adalah Model SDLC (*System Development Life Cycle*) air terjun (*waterfall*) sering disebut juga 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*). Menjelaskan kronologis penelitian, termasuk desain penelitian, prosedur penelitian (dalam bentuk algoritma, *Pseudocode* atau lainnya), bagaimana untuk menguji dan akuisisi data.

Dalam jurnal tersebut menjelaskan bahwa deskripsi dari program penelitian harus didukung referensi, sehingga penjelasan tersebut dapat diterima secara ilmiah (Syahriani dan Jaya, 2017). Adapun kesimpulan yang dijelaskan dalam jurnal tersebut adalah sebagai berikut:

- a. Memudahkan PT. BFI Finance Indonesia, Tbk Cabang Meruya Jakarta dalam menyeleksi karyawan melalui website.
- b. Memudahkan pelamar mendaftarkan diri dengan cara online tanpa harus datang ke perusahaannya.
- c. Memberikan Informasi Lowongan Pekerjaan melalui website kepada pelamar
- d. Perusahaan bisa mencari pelamar baru dengan cepat dan tepat memanfaatkan teknologi website secara online.

2.1.2 Rancang Bangun Sistem Informasi Penerimaan Siswa Baru

Penerimaan siswa baru yang merupakan kegiatan bisnis yang dilakukan oleh lembaga pendidikan setiap tahun. Dengan semakin majunya dunia teknologi informasi maka mendorong penggunaan teknologi dalam setiap aspek kehidupan. Bagi institusi pendidikan, penggunaan teknologi informasi untuk media dan sarana untuk meningkatkan proses penerimaan siswa baru menjadi penting.

Model pengembangan sistem yang digunakan adalah metode RAD karena rancang bangun sistem tersebut harus diselesaikan dalam waktu yang cukup singkat, lamanya suatu pengembangan sistem akan berpengaruh terhadap besarnya biaya proyek sistem informasi yang dibangun. Berdasarkan pembahasan dalam jurnal tersebut dari hasil penelitian diketahui bahwa sistem penerimaan siswa baru dapat meningkatkan kinerja sistem penerimaan siswa baru.

Pengembangan sistem penerimaan siswa baru berbasis web dengan menggunakan metode RAD dapat meningkatkan kinerja penerimaan siswa baru. dari hasil penelitian diketahui bahwa sistem penerimaan baru merupakan sistem yang penting bagi setiap institusi atau lembaga pendidikan. Sistem penerimaan yang baik, efisien dan efektif perlu dibuatkan sistem informasi yang mampu meningkatkan kinerja sistem penerimaan siswa baru.

Dengan adanya Sistem penerimaan siswa baru dapat meningkatkan pula pelayanan dan jumlah siswa baru yang diterima dan mempermudah pembuatan laporan siswa baru kepada yayasan. Laporan penerimaan siswa baru dapat dihasilkan lebih cepat, tepat dan akurat sehingga mampu memberikan saran

pendukung terhadap masalah penerimaan baru yang dihadapi (Yunita, dkk., 2018).

2.1.3 Rancangan Sistem Informasi E-Recruitment Berbasis WEB pada PT.GEOSERVICES

Web E-Recruitment dibuat untuk memudahkan calon pelamar untuk memilih pekerjaan yang sesuai dengan kompetensi yang dimiliki dan tanpa harus mendatangi perusahaan untuk mengirimkan berkas maupun melakukan test. Sistem *e-recruitment* secara tidak langsung akan memangkas waktu dan biaya pengeluaran untuk proses rekrutmen karyawan baru guna memudahkan setiap divisi jika ingin mengajukan permintaan karyawan baru dan mempermudah pihak HRD untuk mengolah data pelamar yang masuk tanpa perlu mendata secara manual dalam bentuk berkas.

Black Box adalah tipe pengujian perangkat lunak yang tidak diketahui kinerja internalnya. Sehingga para pengguna memandang perangkat lunak seperti layaknya sebuah kotak hitam yang tidak penting untuk dilihat isinya, tetapi cukup dikenai proses testing bagian luar. Pada jenis *black box* testing, perangkat lunak tersebut akan dieksekusi kemudian berusaha ditest apakah telah memenuhi kebutuhan pengguna yang didefinisikan pada saat awal tanpa harus membongkar *listing* atau kode programnya

Perancangan E-recruitment pada PT. GEOSERVICES menggunakan model RAD yang merupakan adaptasi dari model air terjun versi kecepatan tinggi untuk pengembangan setiap komponen perangkat lunak. Model RAD membagi tim pengembang menjadi beberapa tim untuk mengerjakan beberapa komponen masing-masing tim pengerjaan dapat dilakukan secara paralel.

Sistem Informasi *E-recruitment* berbasis *Web* PT. Geoservices dibuat untuk memudahkan HRD dalam mengelola data pelamar dan juga memudahkan *user* masyarakat umum untuk mendaftar menjadi pelamar dan memudahkan pelamar dalam memilih lowongan pekerjaan yang ada serta mengirimkan berkas data lamaran yang dibutuhkan (Anisah, dkk., 2016).

2.1.4 Pembuatan Sistem Pendaftaran Siswa Baru di SDN Pucangan 6 Kartasura Berbasis *Website*

Organisasi/instansi atau perusahaan dapat menyelesaikan masalah manajemen dengan menunjang efektifitas, produktifitas dan efisiensi terutama memberikan pelayanan kepada calon siswa baru di SDN Pucangan 6 Kartasura, maka perlu menyusun suatu sistem pengelolaan data terkomputerisasi yang baik dan berdaya guna. Sistem tersebut diharapkan dapat menyelesaikan tugas-tugas rutin yang berguna dalam pengambilan keputusan oleh organisasi/instansi atau perusahaan tersebut terutama pada SDN Pucangan 6 Kartasura.

Sistem yang digunakan sebelumnya pada SDN Pucangan 6 Kartasura yaitu para calon siswa datang ke lokasi pendaftaran kemudian meminta formulir pendaftaran dengan melengkapi semua berkas yang diperlukan dan persyaratan administrasi lainnya seperti biaya pendaftaran, setelah proses pendaftaran tersebut, calon siswa mendapatkan kwitansi tanda bukti melakukan pembayaran biaya pendaftaran sedangkan sistem yang baru para calon siswa datang ke lokasi pendaftaran kemudian meminta formulir pendaftaran dengan melengkapi semua berkas yang diperlukan dan persyaratan administrasi lainnya seperti biaya pendaftaran, setelah proses pendaftaran tersebut, panitia atau admin memasukkan data siswa ke *database*.

Website ini memiliki 2 user yaitu admin dan pengguna yang mana memiliki hak akses yang berbeda. Hak akses admin meliputi memasukkan berita, memasukkan gambar ke galeri, pendaftaran siswa, dan memeriksa buku tamu, sedangkan pengguna hanya dapat melihat-lihat seperti home, profil sekolah, galeri, contact, dan buku tamu.

Web ini memberikan informasi tentang pendaftaran kepada pengunjung yang ingin mendaftar atau mencari tentang pendaftaran (Mantiri, dkk., 2018)

2.1.5 Sistem Pelayanan Pendaftaran Mahasiswa Baru Universitas Sahid Surakarta

Kegiatan pendaftaran mahasiswa baru di Universitas Sahid Surakarta baik pendaftaran secara *online* maupun pendaftaran secara langsung belum dilengkapi dengan tahap ujian seleksi. Sehingga pihak Universitas harus menentukan calon mahasiswa yang akan diterima secara manual, karena belum ada batas standart untuk calon mahasiswa yang akan diterima. Didukung adanya kecanggihan teknologi di jaman sekarang ini, maka tata cara penerimaan mahasiswa baru tersebut harus dibenahi mengikuti seiring berkembangnya ilmu pengetahuan dan teknologi. Berkat adanya internet, maka calon mahasiswa dapat melakukan pendaftaran secara mudah dan *online* tanpa dibatasi tempat dan waktu.

Sistem Pelayanan Pendaftaran Mahasiswa Baru Universitas Sahid Surakarta dapat membantu pengolahan data penerimaan mahasiswa baru menjadi lebih mudah, karena data pendaftaran sudah tersimpan rapi di dalam *database*. Sistem ini dilengkapi dengan fasilitas ujian seleksi secara *online*, sehingga calon mahasiswa dapat melakukan ujian seleksi darimana saja. Sistem ini dilengkapi dengan tampilan grafik yang menunjukkan jumlah calon mahasiswa yang telah mendaftar di Universitas Sahid Surakarta

Pada Sistem ini, Dir.Kehumasan dan PMB, Dir. Keuangan, Dir. Akademik dan Wakil Rektor III dapat melihat dan menyimpan laporan hasil pendaftaran calon mahasiswa baru dan hasil ujian seleksi. Laporan tersebut dapat disimpan dalam bentuk *file excel*.

Sistem ini telah diuji dengan metode *WebQual* kepada 30 responden dan menghasilkan nilai kalkulasi akhir sebesar 85,8%. Berdasarkan hasil tersebut dapat disimpulkan bahwa sistem ini sudah baik dan layak untuk diimplementasikan di Universitas Sahid Surakarta (Safitri, 2016)

2.1.6 Analisis Tinjauan Pustaka

Analisis tinjauan pustaka dengan aplikasi yang dibangun ditampilkan pada

Tabel 2.1:

Table 2.1 Perbedaan sistem yang ditinjau dengan sistem yang dibangun

| Diskripsi Perbedaan | Sistem yang ditinjau | Sistem yang dibangun |
|---|---|---|
| Alur sistem secara keseluruhan | Alur sistem sesuai pada perusahaan masing-masing | Alur sistem sesuai dengan CV. RNK |
| Cara akses aplikasi | Pengaksesan laman dapat dilakukan dimana saja secara online. | Aplikasi hanya dapat di akses di CV. RNK dengan koneksi jaringan internal perusahaan. |
| Pengiriman Berkas Lamaran | Dikirim melalui Website perusahaan dengan mengisi data pribadi. | Dikirim melalui email atau datang langsung ke CV.RNK. Pengisian data pribadi dilakukan pada saat pelamar lolos seleksi berkas dan dipanggil HRD untuk psikotest. |
| Proses untuk mendapatkan user name dan password | Semua pelamar yang dapat akses halaman website perusahaan tersebut dapat langsung mendaftarkan diri untuk mendapatkan user name dan password. | Hanya pelamar yang lolos seleksi berkas yang dipanggil perusahaan untuk melengkapi data pribadi dan mengikuti test seleksi berikutnya diberikan user name & password saat test di CV.RNK. |
| Proses Test & Interview | Dilakukan manual di perusahaan masing-masing | Dilakukan melalui aplikasi test masuk di CV.RNK |
| Hasil Test | Dilakukan manual di perusahaan masing-masing. | Otomatis nilai hasil test sudah terbentuk sebagai acuan untuk lanjut ke tahap berikutnya. |
| Data Pribadi | Diinput oleh pelamar ketika mengakses <i>website</i> perusahaan. | Data pribadi diinput oleh pelamar masing masing setelah pelamar mendapatkan userid dan password oleh adin setelah dinyatakan lulus sleksi berkas oleh HRD. |

2.2 Kerangka Berfikir

Kerangka pemikiran dari Tugas Akhir ini dijelaskan pada Gambar 2.1 dan didefinisikan sebagai berikut:

1. Latar belakang masalah

Pokok permasalahan yang mendasari perlunya Rancang Bangun Aplikasi Test Masuk dan Penerimaan Calon Karyawan Baru di CV. RNK Sukoharjo.

2. Rumusan masalah

Perumusan masalah merupakan inti permasalahan dan jalan keluar menyelesaikan permasalahan.

3. Judul tugas akhir

Gambaran umum yang dapat mewakili sistem yang akan dikembangkan dalam penyusunan Tugas Akhir.

4. Pengumpulan data

Pada penelitian dilakukan pengumpulan data pengumpulan penelitian ini menggunakan metode observasi, dokumentasi, dan wawancara.

5. Penguasaan dasar

Pada penelitian dilakukan percobaan membuat sistem agar lebih menguasai.

6. Observasi sistem

Pada penelitian dilakukan pengamatan pada sistem yang sudah ada agar dapat menjadi referensi dalam mengembangkan aplikasi ini.

7. Analisis dan perancangan sistem

Pada penelitian dilakukan menganalisa dan merancang bagaimana sistem nantinya akan dibuat untuk membantu memecahkan permasalahan yang ada.

8. Implementasi sistem

Pada penelitian dilakukan implementasi apa yang sudah dirancang untuk Rancang Bangun Aplikasi Test Masuk Dan Penerimaan Calon Karyawan Baru di CV. RNK Sukoharjo.

9. Pengujian sistem

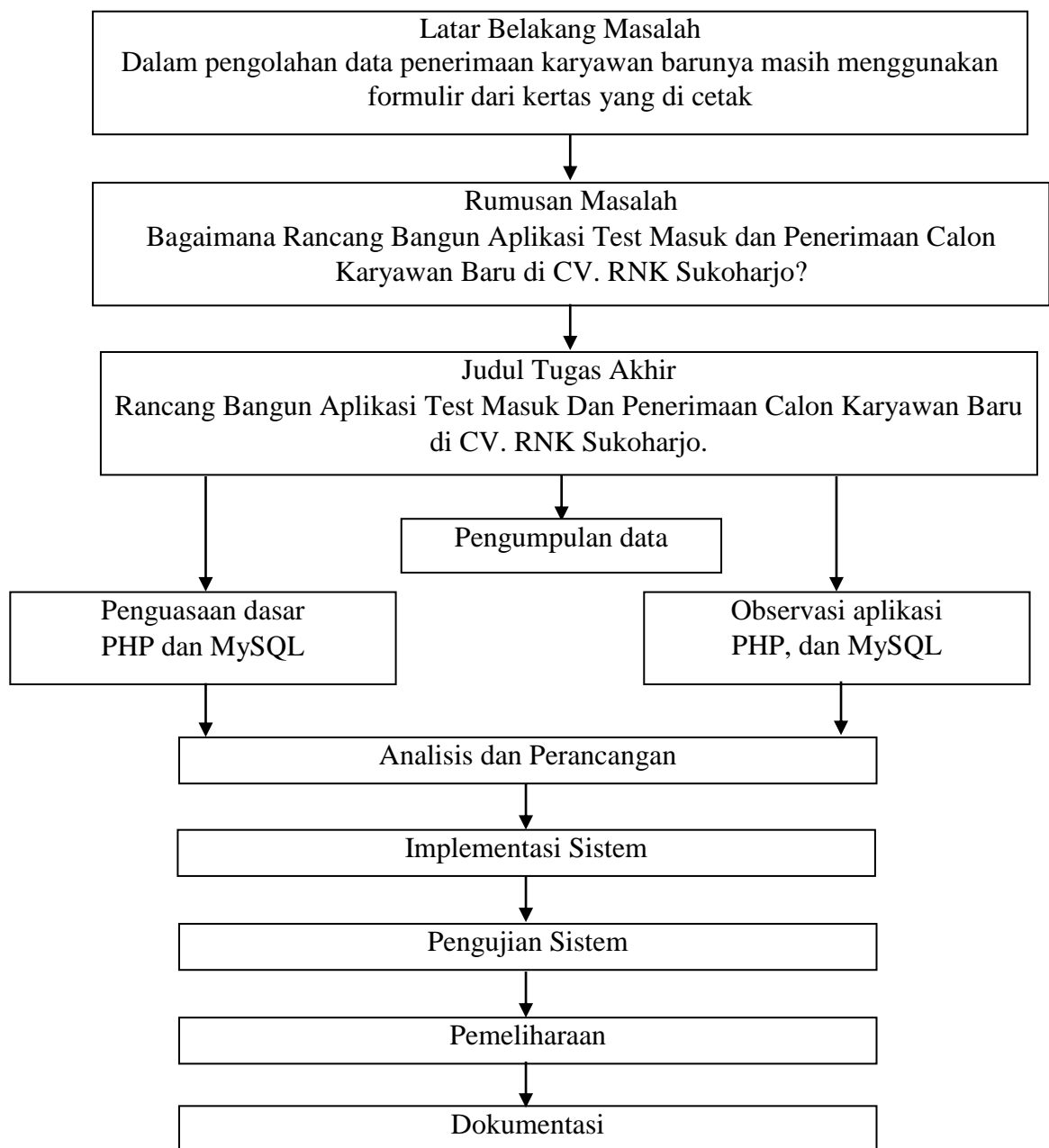
Pada penelitian dilakukan uji coba aplikasi apakah masih terjadi kesalahan ataupun kekurangan pada sistem.

10. Pemeliharaan

Sistem yang sudah diimplementasikan dan diuji coba sebaiknya untuk dilakukan pemeliharaan.

11. Dokumentasi

Dibuatnya dokumentasi dari keseluruhan kegiatan penyusunan Tugas Akhir.



Gambar 2.1 Kerangka Pemikiran

2.3 Teori Pendukung

2.3.1 Pengertian Aplikasi

Aplikasi adalah program siap pakai yang dapat digunakan untuk menjalankan perintah-perintah dari pengguna aplikasi tersebut dengan tujuan mendapatkan hasil yang lebih akurat sesuai dengan tujuan pembuatan aplikasi tersebut. Aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputansi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan (Sari, 2017).

Aplikasi adalah penggunaan dalam suatu komputer, intruksi (instruction) atau pernyataan yang disusun sedemikian rupa sehingga komputer dapat memproses input menjadi output, aplikasi merupakan merupakan software yang ber-fungsi untuk melakukan berbagai bentuk pekerjaan atau tugas-tugas tertentu seperti penerapan, penggunaan dan penambahan data .

2.3.2 Pengertian Rancang

Rancang adalah kegiatan yang memiliki tujuan untuk mendesain sistem baru yang dapat menyelesaikan masalah-masalah yang dihadapi perusahaan yang diperoleh dari pemilihan alternatif sistem yang terbaik (Sari, 2017).

2.3.3 Pengertian Bangun

Bangun adalah kegiatan menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang telah ada baik secara keseluruhan maupun sebagian (Sari, 2017).

2.3.4 Pengertian Rancang Bangun

Rancang bangun adalah proses pembangunan sistem untuk menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang telah ada baik secara keseluruhan maupun hanya sebagian (Sari, 2017).

2.3.5 Pengertian Perancangan Aplikasi

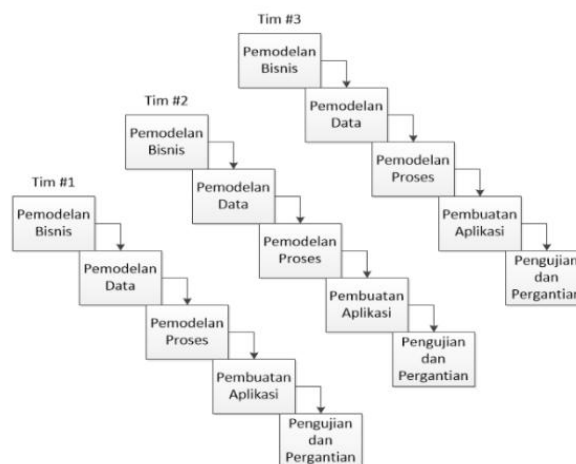
Perancangan aplikasi adalah sekumpulan aktivitas yang menggambarkan secara rinci bagaimana aplikasi akan berjalan. Hal itu bertujuan untuk menghasilkan produk perangkat lunak yang sesuai dengan kebutuhan user (Satzinger, dkk., 2012).

2.3.6 Pengertian Penerimaan Karyawan Baru/Rekrutmen

Menurut Mardianto (2014:8) diartikan sebagai suatu proses untuk mendapatkan calon karyawan yang memiliki kemampuan yang sesuai dengan kualifikasi dan kebutuhan suatu organisasi/perusahaan.

2.3.7 Metode Rapid Application Development (RAD)

Rapid Application Development (RAD) adalah model proses pengembangan perangkat lunak yang bersifat inkremental terutama untuk waktu pengerjaan yang pendek. Model ini merupakan adaptasi dari model air terjun (*Waterfall*) versi cepat dengan menggunakan model air terjun untuk mengembangkan setiap komponen perangkat lunak. (Shalahuddin, 2014). Pada gambar 2.2 dijelaskan mengenai ilustrasi model RAD.



Gambar 2.2. Ilustrasi model RAD.

1. Pemodelan bisnis, dilakukan untuk memodelkan fungsi bisnis guna mengetahui informasi yang terkait dengan proses bisnis, seperti informasi apa saja

yang harus dibuat, siapa yang harus membuat informasi tersebut, bagaimana alur informasi, dan proses apa saja yang terkait dengan informasi tersebut.

2. Pemodelan data, memodelkan data apa saja yang dibutuhkan berdasarkan pemodelan bisnis dan mendefinisikan atribut-atribut beserta relasinya dengan data-data yang lain.
3. Pemodelan proses, mengimplementasikan fungsi bisnis yang sudah didefinisikan terkait dengan pendefinisian data.
4. Pembuatan aplikasi, mengimplementasikan pemodelan proses dan data menjadi program. Model RAD sangat menganjurkan pemakaian komponen yang sudah ada jika dimungkinkan.
5. Pengujian dan pergantian, menguji komponen-komponen yang dibuat dan dilakukan uji penerimaan pengguna. Jika sudah teruji maka tim pengembang dapat beranjak untuk mengembangkan komponen berikutnya.

RAD adalah suatu pendekatan berorientasi objek terhadap pengembangan sistem yang mencakup suatu metode pengembangan serta perangkat-perangkat lunak. RAD bertujuan mempersingkat waktu yang biasanya diperlukan dalam siklus hidup pengembangan sistem tradisional antara perancangan dan penerapan suatu sistem informasi. Pada akhirnya, RAD sama-sama berusaha memenuhi syarat-syarat bisnis yang berubah secara cepat (Kendall, 2010).

Metode pengembangan perangkat lunak *Rapid Application Development* (RAD) adalah model proses pengembangan perangkat lunak yang bersifat incremental terutama untuk waktu pengerjaan yang pendek (Sukanto & Shalahudin, 2016).

Berdasarkan pengertian diatas dapat disimpulkan bahwa *Rapid Application Development* (RAD) merupakan adaptasi dari *warterfall* dan waktu pengerjaannyapun lebih pendek dengan tujuan mempersingkat waktu.

2.3.8 Pengertian Sistem Rekrutmen (*E-Recruitment*)

E-recruitment adalah penggunaan teknologi untuk membantu proses perekrutan. Mereka mengiklankan lowongan pekerjaan melalui internet. Kemudian para pencari pekerjaan akan mengirinkan aplikasi mereka atau

curriculum vitae yaitu CV melalui e-mail menggunakan internet. Pencari tenaga kerja menempatkan CV mereka di website yang dapat ditarik oleh calon karyawan tergantung pada kebutuhannya (Anisah, dkk., 2016).

E-recruitment merupakan proses penerimaan atau perkrutan karyawan baru yang menggunakan media elektronik dan memungkinkan juga untuk pengguna mengirimkan resume mereka secara elektronik seperti *e-mail* atau mengunggah resume mereka di website perusahaan yang bersangkutan bahkan calon pelamar dapat dengan leluasa menemukan pekerjaan yang cocok sesuai dengan keahlian yang dimiliki.

2.3.9 Basis Data

Database atau yang biasa disebut dengan basis data merupakan kumpulan data yang saling berhubungan, Data tersebut biasanya terdapat dalam tabel – tabel yang salibng berhubungan antara tabel satu dengan tabel lain dengan menggunakan *field* atau kolom pada setiap tabel (Prayitno dan Safitri, 2015).

2.3.10 PHP

PHP adalah kependekan dari PHP: HyperText Preprocessor (suatu akronim rekursif) yang dibangun oleh Rasmus Lerdorf pada tahun 1994. Dahulu, pada awal pengembangannya PHP di sebut sebagai kependekan dari Personal Home Page. PHP merupakan suatu bahas pemrograman sisi server yang dapat digunakan untuk membuat halaman Web dinamis (Prayitno dan Safitri, 2015).

PHP adalah singkatan dari *Hypertext Preprocessor*. Saat pertama kali dikembangkan oleh Programmer bernama Rasmus Lerdorf. PHP awalnya singkatan dari *Personal Home Page Tools* (Sari, 2007).

PHP merupakan bahasa untuk pengembangan aplikasi berbasis website yang mendukung metode pemrograman terstruktur dan berorientasi objek (Anisah, dkk., 2016)

2.3.11 HTML

HTML merupakan singkatan dari Hypertext Markup Language dan berguna untuk menampilkan halaman *web* (Prayitno dan Safitri, 2015).

2.3.12 My SQL

MySQL (My Structured Query Language) adalah sebuah perangkat lunak sistem manajemen basis data SQL Database Management System atau DBMS dari sekian banyak DBMS seperti oracle, MS SQL, Postagre SQL dan lainnya (Prayitno dan Safitri, 2015).

MySQL (My Structured Query Language) adalah suatu sistem basis data relation atau Relational Database managemnt System (RDBMS) yang mampu bekerja secara cepat dan mudah digunakan MySQL juga merupakan program pengakses database yang bersifat jaringan, sehingga sapat digunakan untuk aplikasi multi user (banyak pengguna). MySQL didistribusikan gratis dibawah lisensi GPL (General Public License). Dimana setiap program bebas menggunakan MySQL namun tidak bisa dijadikan produk turunan yang dijadikan closed source atau komersial (Destiningrum dan Ardian, 2017).

Mysql adalah salah satu jenis database server yang sangat terkenal, kepopulerannya disebabkan MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya. MySQL bersifat freedom dengan lisensi GNU General Public License (GPL). Dengan adanya keadaan ini maka anda dapat menggunakan software ini dengan bebas tanpa perlu harus takut dengan lisensi yang ada. MySQL termasuk jenis RDBMS (Relational Database Management System). Itulah sebabnya istilah table, baris, kolom digunakan pada MySQL. Pada MySQL sebuah database mengandung satu atau sejumlah tabel (Prasetyo, dkk., 2015).

2.3.13 Apache

Apache adalah server web yang handal dan paling banyak digunakan oleh para *administrator* yang menggunakan sistem operasi *Unix*. Walaupun banyak digunakan pada sistem operasi *unix*, *apache* ini juga dapat digunakan pada operasi

Windows NT/9x, Windows 2000, Netware 5.x, dan OS/2. Selain handal, *Apache* adalah server web yang fleksibel dan mengimplementasikan protokol-protokolweb terbaru seperti HTTP/1.1 (RFC 2616). Salah satu sebab kenapa *Apache* banyak digunakan karena sifat dari software *Apache* sendiri yang *opensource* dan tidak menggunakan lisensi dalam pemakaian *software* tersebut (Taufandan Riza, 2002).

2.3.14 Web Server

Menurut Supardi (2010) “*Web server* merupakan perangkat lunak yang mengelola (mengatur) permintaan *user* dari *browser* dan hasilnya dikembalikan ke *browser*. Contoh *web server*, adalah IIS (*internet information services*) produk *microsoft corp*”.

Web Server atau Server Web merujuk pada perangkat keras (server) dan perangkat lunak yang menyediakan layanan akses kepada pengguna melalui protokol komunikasi HTTP ataupun variannya (seperti FTP dan HTTPS) atas berkas-berkas yang terdapat pada suatu URL ke pemakai (Prayitno dan Safitri, 2015).

2.3.15 Pengujian Alpha dan Beta

Pengujian dilakukan pada program yang telah dibangun, tujuannya adalah untuk menguji kelayakan program dan mengecek kembali apabila ada kesalahan maka program akan dibetulkan. Pengujian *alpha* dilakukan pada tahap *construction* menggunakan teknik *black box* sedangkan tahap pengujian *beta* dilakukan menggunakan kuisioner sebagai alat ukur kepuasan pengguna aplikasi (Rosano, 2019).

Menurut (Hidayat dan Muttaqin, 2018) *Black-Box Testing* merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program.

Menurut (Hanifah dan Alit, 2016) Pengujian sistem bertujuan untuk melihat apakah sistem yang telah dibuat sudah sesuai dengan tujuan awal

pembuatan dan layak untuk dipergunakan. Pengujian pada sistem menggunakan metode Black Box, tujuannya untuk mengetahui bahwa bagian-bagian dalam sistem aplikasi telah benar menampilkan pesan-pesan kesalahan jika terjadi kesalahan dalam penginputan data. Black Box Testing sendiri merupakan pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Pengujian black box ini menitik beratkan pada fungsi system.

Berikut adalah 10 tipe pengujian dari metode Black Box menurut :

1. *Equivalence Partitioning*

Membagi inputan menjadi kelas data yang dapat digunakan untuk menggenerasi kasus uji.

2. *Boundary Value Analysis / Limit Testing*

Mengijinkan untuk menyeleksi kasus uji yang menguji batasan nilai input, Merupakan komplemen dari *Equivalence Partitioning*.

3. *Comparison Testing*

Uji setiap versi dengan data yang sama untuk memastikan semua versi menghasilkan keluaran yang sama.

4. *Sample Testing*

Melibatkan beberapa nilai yang terpilih dari sebuah kelas ekivalen.

5. *Robustness Testing*

Data input dipilih diluar spesifikasi yang telah didefinisikan, Tujuan dari pengujian ini adalah membuktikan bahwa tidak ada kesalahan jika masukan tidak valid

6. *Behavior Testing*

Hasil uji tidak dapat dievaluasi jika hanya melakukan pengujian sekali, tapi dapat dievaluasi jika pengujian dilakukan beberapa kali, misalnya pada pengujian struktur data stack.

7. *Performance Testing*

Mengevaluasi kemampuan program untuk beroperasi dengan benar dipandang dari sisi acuan kebutuhan misalnya : aliran data, ukuran pemakaian memori, kecepatan eksekusi.

8. *Requirement Testing*

Spesifikasi kebutuhan yang terasosiasi dengan perangkat lunak diidentifikasi pada tahap spesifikasi kebutuhan dan desain.

9. *Endurance Testing*

Melibatkan kasus uji yang diulang-ulang dengan jumlah tertentu.

10. *Cause – Effect Relationship Testing*

Bagi-bagi spesifikasi kebutuhan menjadi bagian yang memiliki kemungkinan kerja.

2.3.16 Pengertian *Unified Modeling Language (UML)*

Unified modeling language (UML) adalah metode yang banyak digunakan untuk memvisualisasikan dan mendokumentasikan perangkat lunak dalam mendesain sistem (Shelly, dkk. 2012).





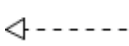
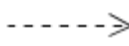

Unified Modelling Language (UML) merupakan suatu set standar konstruksi model dan notasi yang ditetapkan oleh *Object Management Group (OMG)*, yang merupakan suatu standar organisasi untuk pengembangan sistem (Satzinger, dkk., 2012).

Jadi, kesimpulan dari pengertian *unified modeling language (UML)* adalah metode yang banyak yang digunakan untuk memvisualisasikan dan mendokumentasikan perangkat lunak dalam melakukan pendesainan sistem dan juga merupakan dasar dari *object oriented method*.

2.3.16.1 Class Diagram

Class diagram adalah sebuah diagram UML yang merepresentasikan kelas-kelas domain, atribut, pekerjaan pengguna serta hubungan antar kelas tersebut. Pada *class diagram*, bentuk kotak menggambarkan *classes* dan garis menunjukkan hubungan antar class tersebut. *Domain class diagram* digunakan untuk memahami hubungan antar class yang terdiri dari beberapa objek di dalam pengembangan dan perancangan sistem nantinya (Satzinger, dkk., 2012). Tabel *class diagram* ditunjukkan pada Tabel 2.2.

Tabel 2.2 Simbol *Class Diagram*





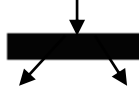
| NO | GAMBAR | NAMA | KETERANGAN |
|----|---|-------------------------|---|
| 1 |  | <i>Generalization</i> | Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>). |
| 2 |  | <i>Nary Association</i> | Upaya untuk menghindari asosiasi dengan lebih dari 2 objek. |
| 3 |  | <i>Class</i> | Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama. |
| 4 |  | <i>Collaboration</i> | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor |
| 5 |  | <i>Realization</i> | Operasi yang benar-benar dilakukan oleh suatu objek. |
| 6 |  | <i>Dependency</i> | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri |
| 7 |  | <i>Association</i> | Apa yang menghubungkan antara objek satu dengan objek lainnya |

2.3.16.2 Activity Diagram

Activity diagram merupakan sebuah tipe dari *diagram workflow* yang menggambarkan tentang aktivitas dari pengguna ketika melakukan setiap kegiatan

dan aliran sekuensial (Satzinger, dkk. 2012). Tabel *activity* diagram ditunjukkan pada Tabel 2.3.

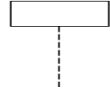
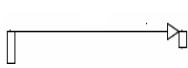

Tabel 2.3 Simbol *Activity* Diagram

| 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 sistem yang mencerminkan eksekusi dari suatu aksi |
| 3 |  | <i>Initial Node</i> | Bagaimana objek dibentuk atau diawali. |
| 4 |  | <i>Activity Final Node</i> | Bagaimana objek dibentuk dan dihancurkan |
| 5 |  | <i>Fork Node</i> | Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran |

2.3.16.3 Sequence Diagram

Sequence diagram digunakan untuk menggambarkan aliran dari informasi yang masuk dan keluar dari sistem yang terotomatisasi. *System sequence* diagram merupakan tipe dari *interaction* diagram yaitu *communication* diagram atau *sequence* diagram yang menunjukkan interaksi diantara objek (Satzinger, dkk., 2012). Tabel *sequence* diagram ditunjukkan pada Tabel 2.4.

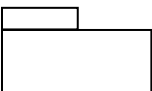
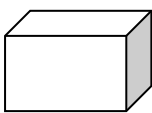
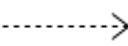

Tabel 2.4 Simbol *Sequence* Diagram

| NO | GAMBAR | NAMA | KETERANGAN |
|----|---|-----------------|--|
| 1 |  | <i>LifeLine</i> | Objek <i>entity</i> , antarmuka yang saling berinteraksi. |
| 2 |  | <i>Message</i> | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi |
| 3 |  | <i>Message</i> | Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi |

2.3.16.4 Deployment Diagram

Deployment diagram merupakan gambaran proses-proses berbeda pada suatu sistem yang berjalan dan bagaimana relasi di dalamnya. Hal inilah yang mempermudah user dalam pemakaian sistem yang telah dibuat dan diagram tersebut merupakan diagram yang statis (Satzinger, dkk., 2012). Tabel *Deployment* diagram ditunjukkan pada Tabel 2.5.


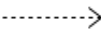








Tabel 2.5 Simbol *Deployment* Diagram

| NO | GAMBAR | NAMA | KETERANGAN |
|----|---|------------------|---|
| 1 |  | <i>Package</i> | Package merupakan sebuah bungkus dari satu atau lebih <i>node</i> |
| 2 |  | <i>Node</i> | Biasanya mengacu pada perangkat keras (<i>hardware</i>) . Perangkat lunak yang tidak biat sendri (<i>software</i>) jika didalam node disertakan komponen untuk mengkonsistenkan rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen. |
| 3 |  | <i>Depedency</i> | Kebergantungan antara node, arah panah mengarah pada node yang dipakai. |
| 4 |  | <i>Link</i> | Relasi antar <i>node</i> |

2.3.16.5 Usecase Diagram

Use case Diagram merupakan suatu aktivitas yang dilakukan oleh sistem, biasanya merupakan sebuah respon untuk permintaan dari pengguna sistem. Aktor tidak selalu sama dengan sumber dari peristiwa di event tabel karena aktor di use case merupakan orang yang berinteraksi dengan sistem yang mana sistem harus meresponnya (Satzinger, dkk. 2012). Tabel *Usecase* diagram ditunjukkan pada Tabel 2.6.

Tabel 2.6 Simbol *Use Case Diagram*

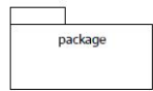
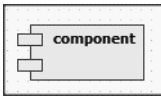

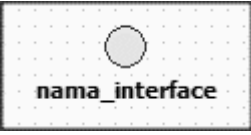

| NO | GAMBAR | NAMA | KETERANGAN |
|----|---|-----------------------|--|
| 1 |  | <i>Actor</i> | Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> . |
| 2 |  | <i>Dependency</i> | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>). |
| 3 |  | <i>Generalization</i> | Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>). |
| 4 |  | <i>Include</i> | Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> . |
| 5 |  | <i>Extend</i> | Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan. |
| 6 |  | <i>Association</i> | Apa yang menghubungkan antara objek satu dengan objek lainnya. |
| 7 |  | <i>System</i> | Menspesifikasikan paket yang menampilkan sistem secara terbatas. |
| 8 |  | <i>Use Case</i> | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor |
| 9 |  | <i>Collaboration</i> | Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi). |
| 10 |  | <i>Note</i> | Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi |

2.3.16.6 Component Diagram

Component diagram adalah suatu bentuk implementasi dari jenis diagram yang menunjukkan sistem arsitektur dan komponen logis secara

keseluruhan di dalamnya (Satzinger, dkk., 2012). Tabel *component* diagram ditunjukkan pada Tabel 2.7.

Tabel 2.7 Simbol *Component* Diagram

| No | GAMBAR | NAMA | KETERANGAN |
|----|---|-----------------------------|--|
| 1 |  | Package | Package merupakan sebuah bungkusan dari satu atau lebih komponen. |
| 2 |  | Komponen | Komponen sistem. |
| 3 |  | Ketergantungan (Dependency) | Kebergantungan antar komponen, arah panah mengarah pada komponen yang dipakai. |
| 4 |  | Antarmuka (Interface) | Sama dengan interface pada pemrograman berbasis objek, yaitu sebagai antarmuka komponen agar tidak mengakses langsung komponen |
| 5 |  | Link | Relasi antar komponen. |