

BAB III

TINJAUAN PUSTAKA

3.1. Landasan Teori

3.1.1. Website

Menurut Yuhefizar (2009:2), *Website* adalah keseluruhan halaman-halaman *web* yang terdapat dalam sebuah domain yang mengandung informasi sebuah *website* biasanya dibangun atas banyak halaman *web* yang saling berhubungan. Hubungan antara satu halaman *web* dengan yang lain disebut *hyperlink*, sedangkan *text* yang dijadikan media penghubung disebut *hypertext*.

Menurut Hidayat (2010:2), *Website* atau situs dapat diartikan sebagai kumpulan halaman-halaman yang digunakan untuk menampilkan informasi teks, gambar diam atau gerak, animasi, suara, dan atau gabungan dari semuanya, baik bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait, yang masing-masing di hubungkan dengan jaringan-jaringan halaman. Hubungan antara satu halaman dengan halaman web yang lainnya disebut *Hyperlink*, sedangkan teks yang dijadikan media penghubung disebut *Hypertext*.

Ada beberapa hal yang di persiapkan untuk membangun website gratis, maka harus tersedia unsur-unsur pendukungnya sebagai berikut:

- Nama domain (*Domain name/URL – Uniform Resource Locator*)
- Rumah website (*website Hosting*).

- Contact Management System (cms).

Perkembangan dunia website pada saat ini lebih menekankan pada pengelolaan contact sebuah website. Pengguna yang tidak biasa bahasa pemrograman website pada saat ini bisa membuat dengan memanfaatkan CMS tersebut.

A. Jenis – Jenis *Website*

Andi (2010:2) berdasarkan mengoprasisnnya secara website dibagi menjadi dua jenis yaitu website statik dan *website Dynamic* :

1. *Website Static*

Adalah website yang memiliki halaman front end, yaitu halaman yang dapat dilihat oleh penmgunjung website. Karena fasilitas yang sangat terbatas, isi dari halaman website static bersipat tetap atau tidak berubah. Untuk menganti sebuah halaman webstatic harus dibutuhkan secara manual dan harus mengganti semua kode-kode HTML yang merupakan unsur utama dari website tersebut.

Website static biasa digunakan untuk *company profil* (*profil perusahaan*), yaitu jenis *website* pengumuman berup brosur online yang sangat sederhana dan tidak bisa diubah atau di modifikasi.

2. *Website Dynamic*

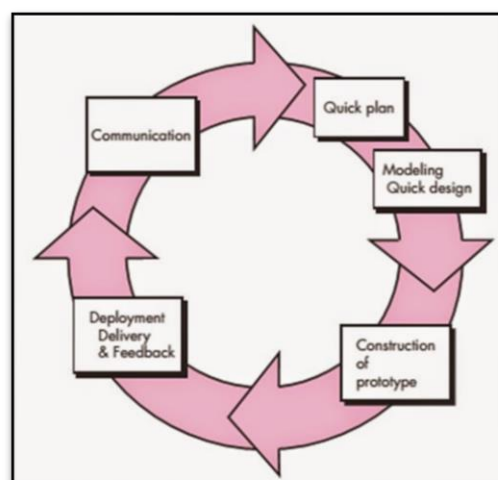
Adalah website yang dapat diubah atau di-update. Dalam website dynamic biasanya terdapat dua halaman front end dan back end. Halaman Front end merupakan halaman yang dapat diakses semua user, sedangkan

halaman blok end merupakan halaman yang hanya dapat diakses oleh admin yang bersangkutan. Back end biasa disebut dengan CMS (*Content Management System*) atau dalam *WordPress* biasa disebut dengan halaman *Dashboard*.

Fungsi dari halaman back end adalah untuk mengatur front end, sebagai contoh untuk pengaturan isi artikel, pengaturan tampilan front end, hingga tambahan untuk menghias front end. Beberapa contoh jenis CMS yang biasa digunakan adalah *WordPress*, *Joomla*, *PHP Nuke*.

3.1.2 *Prototype*

Menurut Pressman (2012;51) *Prototype* merupakan salah satu model SDLC (*system development lyfe cycle*) yang mempunyai ciri khas sebagai model proses *evolusioner*. *Prototype* sendiri bertujuan agar pengguna dapat memahami alur proses aplikasi dengan tampilan dan simulasi yang terlihat siap digunakan. Gambar model *Prototype* dapat dilihat dibawah ini ;



(Sumber : Pressman 2012:51)

Gambar 5.1 Metode *Prototype*

Pada gambar 5.1 menampilkan serangkaian tahapan pengembangan dengan penjelasan mengenai tahapan pada metode yang digunakan untuk penelitian ini adalah sebagai berikut;

a. Identifikasi kebutuhan pemakai

Tahap ini adalah tahap awal melakukan komunikasi dengan admin untuk mengidentifikasi kebutuhan yang diperlukan untuk membangun sistem. Kebutuhan ini berupa data lowongan kerja, data rekrutmen, data penempatan .

b. Membangun *Prototyping*

Setelah mendapatkan data, langkah selanjutnya membuat *UML, USE CASE, DAN DIAGRAM AKTIVITY* sebagai gambaran sistem yang akan dibangun.

c. Menguji *Prototyping*

Setelah membangun prototyping, maka gambaran sistem yang akan dibangun dievaluasi dengan tujuan untuk mendapatkan saran dan masukan dari pemakai. Jika pada tahap ini masih ada revisi maka harus dilakukan perbaikan.

d. Pengkodean system

Dalam tahap ini prototyping yang sudah disepakati selanjutnya diterjemahkan ke dalam bahasa pemrograman yang sesuai yaitu PHP dan database menggunakan MySQL.

e. Pengujian system

Setelah selesai di bangun dengan pemrograman, selanjutnya dilakukan proses pengujian sistem menggunakan metode *blackbox* dilakukan oleh admin.

3.1.3 UML (*Unified Model Language*)

Unified Modeling Language (UML), merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung.

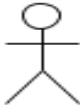
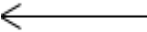




UML hanya berfungsi untuk melakukan pemodelan jadi penggunaan UML tidak terbatas pada metodologi tertentu meskipun pada kenyataannya UML paling digunakan pada metodologi berorientasi objek. Rosa dan Shalahuddin (2011:118).

1) *Diagram Use Case (Use case Diagram)*

Menurut Rosa A.S (2011 : 130), *use case* atau diagram *use case* merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah intraksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu.

Adapun keterangan dari simbol – simbol *use case* dapat dilihat pada Tabel 4.2 sebagai berikut:

Tabel 5.2. Simbol Use Case Diagram

| NO | GAMBAR | NAMA | KETERANGAN |
|----|---|-----------------------|---|
| 1. |  | <i>Actor</i> | Orang, proses, atau sistem lain yang berintraksi 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 fase nama aktor. |
| 2. |  | <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>). |
| 3. |  | <i>Include</i> | Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini. |
| 4. |  | <i>Extend</i> | Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> . |
| 5. |  | <i>Association</i> | Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki intraksi dengan aktor. |
| 6. |  | <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> . |

(Sumber : Rosa, 2011)

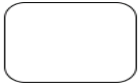
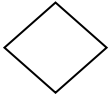



2). Diagram Aktivitas (*Activity Diagram*)

Menurut Rosa A.S (2011 : 134), diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah

sistem atau proses sebuah bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

Adapun keterangan dari simbol – simbol *activitydiagram* dapat dilihat pada Tabel 3.4 sebagai berikut.

Tabel 3.4. Simbol Activity Diagram

| NO | GAMBAR | NAMA | KETERANGAN |
|----|---|--|--|
| 1. |  | <i>Activity</i> | Aktivitas yang dilakukan sistem, aktivitas biasanya diawal dengan kata kerja. |
| 2. |  | <i>Decision</i> | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu. |
| 3. |  | <i>Initial Node</i> | Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal |
| 4. |  | <i>Activity Final Node</i> | Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir. |
| 5. |  | <i>Pengabungan</i> / <i>Join</i> | Asosiasi pengabungan dimana lebih dari satu aktivitas digabungkan menjadi satu. |

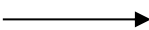



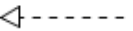
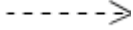

(Sumber : Rosa, 2011)

3). Diagram kelas (*Class Diagram*)

Menurut Rosa A.S (2011:122), diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem.

Adapun keterangan dari symbol-simbol class diagram dapat dilihat pada Tabel 3.1 sebagai berikut :

Tabel 3.1. Simbol *Class Diagram*

| NO | GAMBAR | NAMA | KETERANGAN |
|----|---|------------------------|---|
| 1. |  | <i>Generalization</i> | Relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus) |
| 2. |  | <i>NaryAssociation</i> | Upaya untuk menghindari asosiasi dengan lebih dari 2 objek. |
| 3. |  | <i>Class</i> | Kelas pada struktur sistem |
| 4. |  | <i>Collaboration</i> | Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor. |
| 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 memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri |
| 7. |  | <i>Association</i> | Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> |

(Sumber : Rosa, 2011)

3.2. Penelitian Terdahulu

Penelitian terdahulu adalah penelitian yang sudah dibuat oleh peneliti lain yang digunakan sebagai referensi penulis. Adapun peneliti terdahulu dapat dilihat pada Tabel 5.5 sebagai berikut.

Tabel 5.5. Penelitian Terdahulu

| No | Judul | Penulis | Hasil |
|----|--|---|---|
| 1 | Pengembangan Modul UKM Web Portal Crowdfunding Dengan Metode Prototype Dan Framework Codeigniter ISSN : 2355-9365 e-Proceeding of Engineering : Vol.2, No.2 Agustus 2015 Page 5125 | HARPAND I WIBOWO1 , YULI ADAM PRASETY O,S.T.,M.T. 2, FAISHAL MOFIED AL- ANSHARY ,S.Kom,M. Kom,M.Sc3 | Menghasilkan pengembangan modul UKM web portal crowdfunding dengan metode prototype. Metode yang digunakan peneliti pada perancangan sistem menggunakan metode Prototype. Prototype Model merupakan salah satu metode pengembangan perangkat lunak. Penerapan sistem berbasis objek oriented dengan pemodelan UML |
| 2 | Pembangunan Sistem Informasi Penerima calon Tenaga Kerja Secara Online Berbasis Web pada Bursa kerja Khusus SMK Ganesha Taman Boyolali. | Farid Hamzah Habibie | Menghasilkan pembangunan sistem informasi penerimaan calon tenaga kerja berbasis web dengan bahasa pemrograman PHP Hipertext Pree Processor (PHP), dan sistem database MySQL. Penerapan sistem menggunakan Pemrograman terstruktur dengan Menerapkan ERD,DFD,dan flowchat |
| 3 | Sistem informasi Job Fair Pada Dinas Sosial Tenaga Kerja Dan Transmigasi Kota Pekalongan Berbasis Web | Aslam Fatkhudin1, Imam Rosyadi2, Nur Hesti Sulistyowati 3 | Menghasilkan Sistem informasi Job Fair Pada Dinas Sosial Tenaga Kerja Berbasis Web metode yang di gunakan waterfall dengan bahasa pemrograman PHP Hipertext Pree Processor (PHP), dan sistem database MySQL. Pemrograman terstruktur menggunakan ERD,DFD,dan menggunakan flowchat |