

BAB 2

DASAR TEORI

2.1 Yayasan Sosial Senyum Desa

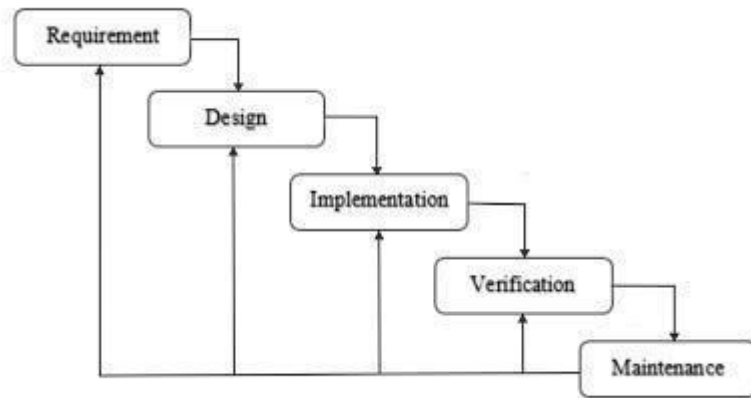
Senyum Desa dicetuskan pada tanggal 27 Oktober 2017 di Kabupaten Sampang oleh Abdul Rozak salah satu mahasiswa Fakultas Hukum Universitas Airlangga Surabaya. Senyum Desa merupakan komunitas pengabdian masyarakat yang independen tanpa berafiliasi dengan pihak manapun, sumber pendanaan kegiatan murni dari para donatur serta pihak lain yang tidak mengikat.

Senyum Desa berawal dari terbentuk di Kabupaten Sampang Provinsi Jawa Timur, komunitas Senyum Desa sudah beranggotakan 50-150 orang yang tersebar di beberapa daerah se-indonesia. Serta, dia juga membentuk koordinator wilayah (korwil) di Madura seperti Sampang, Bangkalan, Pamekasan, Sumenep. Komunitas tersebut juga sudah membangun koordinator wilayah sejumlah daerah luar pulau Madura.

Seperti Surabaya, Malang, Kediri, Lamongan, Bondowoso, Lombok Tengah, Kotawaringin (Kalimantan Tengah), Kalimantan Barat, NTB, dan Sungai Ambawang Selain sudah tersebar di daerah tersebut, pria yang mendirikan komunitas itu di usianya yang 20 tahun juga sudah berpartisipasi membangun di desa pelosok Sulawesi.

2.2 Metode Pengembangan Perangkat Lunak

Menurut Rosa A.S. dan M. Shalahuddin “*Waterfall* adalah model SDLC paling sederhana. Model ini hanya cocok untuk pengembangan perangkat lunak dengan spesifikasi yang tidak berubah-ubah”. Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (sequential linear) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau berurutan dimulai dari analisis, desain, pengkodean, pengujian, dan tahap pendukung (*support*). [1]



Gambar 2. 1 Gambar Metode Waterfall

(Sumber: (Rosa A.S. dan M.Shalahuddin, 2017))

a. Analisis Kebutuhan Perangkat Lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

b. Desain

Desain perangkat lunak adalah proses multistep yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya. Desain perangkat lunak yang dihasilkan pada tahap ini juga perlu didokumentasikan.

c. Pembuatan Kode Program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

d. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logika dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

e. Pendukung (*support*) atau Pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

2.3 Rancang Bangun

Rancang merupakan serangkaian prosedur untuk menerjemahkan hasil analisa dari sebuah sistem ke dalam bahasa pemrograman untuk mendeskripsikan dengan detail bagaimana komponen-komponen sistem diimplementasikan.

Perancangan 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 [2]. Sedangkan pengertian bangun atau pembangunan sistem adalah kegiatan menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang telah ada baik secara keseluruhan maupun sebagian [3].

Dengan demikian pengertian rancang bangun merupakan kegiatan menerjemahkan hasil analisa dari sebuah sistem ke dalam bahasa pemrograman untuk menciptakan sistem baru maupun mengganti atau memperbaiki sistem yang telah ada.

2.4 Sistem Informasi

Sistem informasi adalah suatu sistem dalam organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat terstruktur, kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan [4].

Adapun definisi sistem informasi menurut Kristanto (2008) yaitu kumpulan dari perangkat keras dan perangkat lunak komputer serta perangkat manusia yang akan mengolah data menggunakan perangkat keras dan perangkat lunak tersebut. Menurut Jogiyanto Hartono (2005) untuk menghasilkan informasi, suatu sistem informasi harus mempunyai lima komponen, yaitu:

1. Komponen *input*, komponen ini merupakan bahan dasar pengolahan informasi karena *input* merupakan data yang masuk ke dalam sistem.
2. Komponen *output*, merupakan produk sistem informasi. *Output* sistem informasi harus berupa informasi yang berguna bagi pemakainya.
3. Komponen basis data, yaitu kumpulan data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.
4. Komponen model, komponen ini menunjukkan pengolahan data lewat suatu model-model tertentu untuk menghasilkan informasi yang dibutuhkan.
5. Komponen teknologi, komponen ini berfungsi untuk mempercepat pengolahan data.

Secara umum sistem informasi merupakan kombinasi dari orang (*people*), perangkat keras (*hardware*), jaringan komunikasi (*communications network*) dan sumber data yang dihimpun, ditransformasi dan mengalami proses pengaliran dalam suatu organisasi

2.5 Website

Website (situs *web*) 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 halaman *web* yang lainnya disebut dengan *hyperlink*.

Sedangkan *website* Menurut Sibero (2013:11) "web adalah suatu sistem yang berkaitan dengan dokumen digunakan sebagai media untuk menampilkan teks, gambar, multimedia, dan lainnya pada jaringan internet".[5]

Dari penjelasan diatas penulis menyimpulkan *website* merupakan halaman situs yang menampilkan informasi di *internet* yang dapat diakses melalui sebuah *browser* yang nantinya akan menampilkan teks, gambar, multimedia dan lainnya pada jaringan internet.

2.6 PHP

PHP (*Hypertext PreProcessor*) merupakan bahasa pemrograman *script* yang membuat dokumen HTML secara *on the fly* yang dieksekusi di *server web*, dokumen HTML yang dihasilkan dari suatu aplikasi bukan dari dokumen HTML yang dibuat dengan menggunakan *editor* teks atau *editor* HTML, dikenal juga sebagai bahasa pemrograman *server side*[5]

PHP pada awalnya diciptakan oleh Rasmus Lerdorf pada tahun 1994 sebagai sebuah aplikasi kecil (berbentuk makro) untuk melengkapi situs personalnya di *internet*. Itu sebabnya pada saat itu, PHP merupakan singkatan dari *Personal Homepage*. Kemudian, dikembangkan lagi oleh masyarakat *internet* sukarelawan pendukung gerakan *Public License* menjadi sebuah bahasa yang lebih sempurna seperti saat ini.[6]

PHP Merupakan bahasa berbentuk *script* yang ditempatkan dalam *server* dan diproses di *server*. Secara khusus PHP dirancang untuk membentuk aplikasi dinamis. Artinya, dapat membentuk suatu tampilan berdasarkan permintaan.

2.7 CodeIgniter

CodeIgniter adalah sebuah *web application network* yang bersifat *open source* yang digunakan untuk membangun aplikasi php dinamis. CodeIgniter menjadi sebuah *framework* PHP dengan model MVC (*Model, View, Controller*) untuk membangun *website* dinamis dengan menggunakan PHP yang dapat mempercepat pengembang untuk membuat sebuah aplikasi *web*.

Selain ringan dan cepat, CodeIgniter juga memiliki dokumentasi yang super lengkap disertai dengan contoh implementasi kodenya. Dokumentasi yang lengkap inilah yang menjadi salah satu alasan kuat mengapa banyak orang memilih CodeIgniter sebagai *framework* pilihannya.

2.8 XAMPP

XAMPP merupakan paket *web server* PHP dan *database* MySQL yang paling populer di kalangan pengembang *web* dengan menggunakan PHP dan MySQL sebagai *datasenya*. XAMPP merupakan proyek yang dikerjakan oleh dua orang bernama Kai Oswald dan Kay Vogelgesang. Mereka menciptakan suatu pengembangan yang sempurna seperti diantaranya Apache, MySQL, PHP, Perl, dan berbagai komponen dalam beberapa tahun ini.

Bagian penting XAMPP yang digunakan pada umumnya:

- XAMPP *Control Panel Application*, berfungsi mengelola layanan (*service*) XAMPP. Seperti mengaktifkan layanan (*start*) dan menghentikan (*stop*) layanan.
- *Htdocs*, yaitu folder tempat meletakkan berkas-berkas yang akan dijalankan. Di *Windows*, folder ini berada di *C:/xampp*.
- *PHPMysqlAdmin* merupakan bagian untuk mengelola *database*

2.9 MY SQL

MySQL (*My Structure Query Language*) adalah sebuah program berbasis *DOS* yang bersifat *open Source*. MySQL adalah produk yang berjalan pada *platform* baik *windows* maupun *Linux*. Selain itu, MySQL merupakan program pengakses *database* yang bersifat jaringan sehingga dapat digunakan untuk *multi-user* (banyak pengguna).

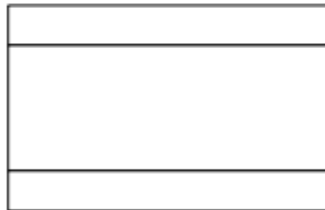
Kelebihan lain dari MySQL adalah menggunakan bahasa *query* standar yang dimiliki SQL (*Structure Query Language*). SQL adalah suatu bahasa permintaan yang terstruktur yang telah distandarkan untuk semua program pengakses *database* seperti *Oracle*, *Postgresql*, dan *SQL Server*. MySQL dapat didukung oleh hampir semua program aplikasi baik yang *open source* maupun yang tidak, yang ada pada *platform windows* (Syafii, 2005).

2.10 Javascript

JavaScript adalah bahasa scripting kecil, ringan, berorientasi objek yang ditempelkan pada kode HTML. JavaScript digunakan dalam pembuatan *website* agar lebih interaktif dengan memberikan kemampuan tambahan terhadap HTML melalui eksekusi perintah di sisi browser. JavaScript dapat merespon perintah *user* dengan cepat dan menjadikan halaman *web* menjadi *responsive*. JavaScript memiliki struktur sederhana, kodenya dapat disisipkan pada dokumen HTML atau berdiri sebagai satu kesatuan aplikasi.

2.11 Class Diagram

Class adalah cetak biru objek. *Class* mempresentasikan atribut-atribut. Atribut-atribut berkorespondensi dengan variabel-variabel instan di dalam kelas. (Heriyanto, 2004 : 340). Adapun simbol *class* diagram sebagai berikut:



Gambar 2. 2 Simbol *Class* Diagram

2.12 Efisiensi

Menurut Kamus Besar Bahasa Indonesia, efisiensi merupakan ketepatan cara (usaha kerja) dalam menjalankan sesuatu (dengan tidak membuang waktu , tenaga, dan biaya), atau kemampuan menjalankan tugas dengan baik dan tepat. Semakin sedikit dana atau sumber daya yang digunakan dalam mencapai suatu hasil yang diinginkan . maka dapat dikatakan semakin efisiensi berkaitan dengan penggunaan sumber daya yang terbatas, namun dapat menghasilkan sesuatu yang diharapkan atau direncanakan. Suatu kegiatan bisa dikatakan efisien jika prosesnya berjalan dengan baik, misalnya prosesnya berjalan lebih cepat atau lebih murah