

Rancang Bangun Sistem Jejaring Klaster Berbasis Web Menggunakan Metode *Model View Controller*

Supriyadi¹
Rini Kartika Hudiono²
Lina Sinatra Wijaya³

*e-mail : supriyadi@staff.uksw.edu, rinihudiono@gmail.com,
linasinatra@yahoo.co.uk.*

Diterima : 19 Maret 2012 / Disetujui: 19 April 2013

ABSTRACT

Cluster in Indonesia particularly in the area of Central Java in 2010 totaled 150. Based on the results of the survey in Demak, Wonosobo and Pemalang, most Clusters does not yet have a website as a media product promotion and Cluster communication. Some Cluster already leverages media Blog and facebook but the media still do not meet the expected needs because media Blog and Facebook are not a media for product promotion. This is because the limitations of cost and the scarcity of skilled human resources information technology owned by the Cluster. Therefore, this research was conducted on the design and implementation of a web application that serves as a Networking System Cluster. System development method's using the Prototype Model. Method of MVC (Model View Controller) is used in the implementation of the system, while technology programming using CodeIgniter Framework. As a result, the system is capable of providing the website for each Cluster are listed, so it can be used as a promotional media, publications and communication between Cluster members.

Keywords: Cluster, MVC, Framework CodeIgniter

ABSTRACT

Klaster di Indonesia khususnya di wilayah Jawa Tengah pada tahun 2010 berjumlah 150. Berdasarkan hasil survei di kabupaten Demak, Wonosobo dan Pemalang, sebagian besar

1. Staf Pengajar Fakultas Teknologi Informasi Universitas Kristen Satya Wacana
Jl. Diponegoro 52-60, Salatiga 50711, Indonesia
2. Staf Pengajar Fakultas Teknologi Informasi Universitas Kristen Satya Wacana
Jl. Diponegoro 52-60, Salatiga 50711, Indonesia
3. Staf Pengajar Fakultas Teknologi Informasi Universitas Kristen Satya Wacana
Jl. Diponegoro 52-60, Salatiga 50711, Indonesia

Klaster belum memiliki website sebagai media promosi produk Klaster dan sarana komunikasi. Sebagian Klaster sudah memanfaatkan media Blog dan facebook tetapi media tersebut masih belum memenuhi kebutuhan yang diharapkan, karena media blog dan facebook memang bukan media untuk promosi produk. Hal ini disebabkan karena keterbatasan biaya dan minimnya sumber daya manusia berkemampuan teknologi informasi yang dimiliki Klaster. Oleh karena itu, pada penelitian ini dilakukan perancangan dan implementasi sebuah aplikasi web yang berfungsi sebagai Sistem Jejaring Klaster. Metode pengembangan sistem menggunakan Prototype Model. Metode MVC (Model View Controller) dipergunakan dalam implementasi sistem, sedangkan teknologi pemrograman menggunakan Framework CodeIgniter. Sebagai hasilnya, sistem ini mampu memberikan website bagi setiap Klaster yang terdaftar, sehingga dapat dipergunakan sebagai media promosi, publikasi dan komunikasi antar anggota Klaster.

Kata Kunci: *Klaster, MVC, Framework Codeigniter*

PENDAHULUAN

Perkembangan teknologi informasi, khususnya teknologi komputer yang kian pesat pada saat ini telah menjadi suatu kebutuhan yang mendasar dalam setiap segi kehidupan manusia maupun masyarakat. Hal ini dapat dilihat dari penggunaan teknologi komputer dalam bidang komunikasi, transportasi, bisnis, bahkan dalam bidang kesehatan. Dalam bidang bisnis, salah satu program pemerintah adalah penguatan kelembagaan klaster, yaitu memberdayakan klaster sebagai pusat bisnis masyarakat guna meningkatkan perekonomian lokal. Sebagai contoh yaitu pada tahun 2007 Bank Indonesia sudah melakukan pengembangan klaster melalui *Pilot Project Klaster*[1]. SIDaklaster.com adalah website Balitbang Prov. Jawa Tengah berfungsi sebagai media informasi, representasi, konsultasi, fasilitasi dan promosi bagi UMKM berbasis klaster.

Berdasarkan hasil pengamatan bahwa jumlah klaster di Jawa Tengah berjumlah 150, terdiri dari 87 klaster industri, 49 klaster pertanian, dan 14 klaster pariwisata[2]. Berdasarkan hasil survei di Kabupaten Demak, Wonosobo dan Pemalang, beberapa klaster belum sepenuhnya memanfaatkan teknologi informasi dipergunakan dalam proses bisnis atau untuk pengembangan bisnis mereka. Hal ini terbukti bahwa klaster-klaster tersebut belum memiliki situs sebagai sarana sistem informasi maupun sarana promosi produk hasil usaha anggota klaster, melainkan mempergunakan sarana atau media sosial yang ada seperti *facebook* dan *twitter* untuk dipergunakan sebagai sarana promosi dan jejaring usaha. Beberapa klaster yang memiliki anggota berkemampuan teknologi informasi, telah memanfaatkan Blog sebagai sarana media promosi bisnis klaster mereka.

Permasalahan-permasalahan tersebut yang mendasari penelitian ini. Salah satu kebutuhan yang mendasar untuk penyelesaian permasalahan tersebut adalah bahwa klaster membutuhkan penerapan teknologi informasi sebagai media promosi, bisnis dan komunikasi bisnis antar anggota klaster, yaitu berupa *Website*. Berdasarkan hal tersebut, maka penelitian ini bertujuan untuk merancang suatu sistem jejaring klaster yang mampu menyediakan *website* bagi klaster anggota dengan memanfaatkan satu alamat domain dengan mengimplementasikan menggunakan metode *Model View Controller* (MVC) dengan teknologi *Framework CodeIgniter*. Dengan demikian, diharapkan bahwa Klaster dapat memiliki *website* yg mampu mengelola anggota klaster dan dapat menampilkan produk anggota klaster serta dapat dipergunakan sebagai sarana komunikasi bisnis. Sedangkan manfaat pada bidang akademik yaitu supaya dapat dipergunakan sebagai sarana pembelajaran membangun sebuah aplikasi

web yang dapat membuat dan menyediakan *website* dengan memanfaatkan satu alamat domain dengan metode *Model View Controller*.

Untuk tidak memperluas pembahasan, perlu adanya batasan-batasan untuk menyederhanakan permasalahan, yaitu tidak membahas keamanan konten *web*, tidak membahas unjuk kerja dari aplikasi yang dibangun, tidak membahas secara detil penerapan teknologi *framework CodeIgniter* pada proses implementasi aplikasi. Penelitian ini berfokus pada penerapan metode *Model View Controller* pada proses implementasi sistem dan tidak membahas dampak implementasi sistem bagi Klaster.

KAJIAN PUSTAKA

Penerapan dari arsitektur MVC (*Model View Controller*) telah banyak digunakan dalam pembuatan aplikasi yang mendukung suatu sistem. Pada penelitian sebelumnya dengan judul Multimedia Pembelajaran Bahasa Inggris Menggunakan Konsep Jejaring Sosial Berbasis *Web* [3] yang membahas tentang bagaimana membangun sebuah aplikasi *web* multimedia pembelajaran bahasa inggris dengan menggabungkan konsep *e-Learning* dan konsep jejaring sosial menggunakan metode MVC dan teknologi *Framework CodeIgniter*.

Judul kedua pada penelitian selanjutnya adalah “Membangun *Website e-Commerce* dengan Teknik MVC Menggunakan *Framework CodeIgniter*”[4] yaitu membuat sebuah cms *e-commerce* secara mandiri dengan memanfaatkan *framework CodeIgniter* yang sudah menggunakan konsep *Model, View, Controller* (MVC).

Berdasarkan kedua penelitian terdahulu tersebut maka penelitian ini mendapatkan masukan tentang penerapan konsep Jejaring Sosial pada sebuah aplikasi *web e-Learning*, serta penerapan metode MVC pada implementasi suatu sistem berbasis pada *web e-Commerce*. Sedangkan perbedaan dengan penelitian ini adalah konsep Jejaring Sosial diterapkan pada aplikasi *e-Commerce*, sedangkan metode MVC diterapkan guna memudahkan dalam perawatan dan pengembangan aplikasi karena MVC memisahkan pengembangan aplikasi berdasarkan komponen utama yang membangun sebuah aplikasi seperti manipulasi data, antarmuka pengguna, dan bagian yang menjadi kontrol dalam sebuah aplikasi *web*.

Sistem Informasi

Data merupakan sekumpulan keterangan atau bukti mengenai suatu kenyataan yang masih mentah, masih berdiri sendiri-sendiri, belum diorganisasikan, dan belum diolah.

Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat bagi pengambilan keputusan saat ini atau saat mendatang.

Sistem adalah sekumpulan unsur / elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan.

Sistem informasi adalah sekumpulan komponen pembentuk sistem yang mempunyai keterkaitan antara satu komponen dengan komponen lainnya yang bertujuan menghasilkan suatu informasi dalam suatu bidang tertentu [5]. Hal tersebut berfungsi untuk membantu perencanaan, pengendalian, dan pengambilan keputusan dengan menyediakan rangkuman rutin dan laporan tertentu.

Klaster

Klaster dapat didefinisikan sebagai grup perusahaan yang berkumpul pada satu lokasi dan bekerja pada sektor yang sama. Dalam hal ini semua yang berhubungan dengan suatu klaster adalah merupakan hubungan yang berfokus pada aktivitas klaster itu sendiri,

mulai dari pengelolaan produk serta aktivitas lainnya yang mana kesemuanya akan terhubung dalam satu jaringan yang saling terintegrasi, dan data dikelola oleh *database* sebagai media penyimpanan data terpusat[6].

Sistem Jejaring Sosial

Sistem jejaring sosial komunitas virtual sebagai perkumpulan sosial yang muncul dari *internet* ketika cukup banyak orang melakukan diskusi publik yang cukup panjang dan perasaan manusia yang cukup untuk membentuk hubungan personal di dunia maya. Biasanya sekelompok orang ini tinggal di daerah yang terpisah satu sama lain, mereka dapat berasal dari desa atau kota yang berbeda tapi memiliki ketertarikan atau tujuan yang sama untuk dibagikan diantara anggota komunitas [7].

PHP (Hypertext Preprocessor) dan SQL (Structure Query Language)

Bahasa pemrograman yang digunakan adalah *Hypertext Preprocessor* (PHP), bahasa pemrograman ini memudahkan dalam membuat aplikasi *web* dengan cepat, dapat digunakan untuk membuat *dynamic website*, baik itu yang memerlukan penggunaan *database* ataupun tidak

Database untuk aplikasi ini menggunakan MySQL karena cukup handal dan gratis. Untuk mengelola *database* tersebut, aplikasi menggunakan SQL dimana SQL adalah sebuah konsep pengoperasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis[8].

Framework CodeIgniter

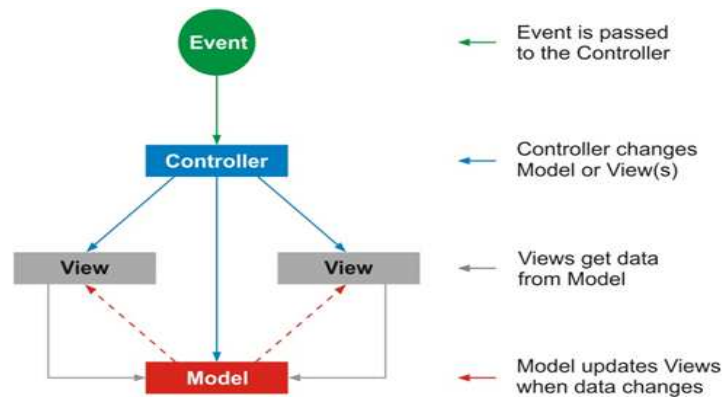
Framework dapat diartikan sebagai kumpulan perintah atau fungsi dasar yang dapat membantu menyelesaikan proses-proses yang kompleks, namun pihak *developer* tetap harus menulis kode sendiri dan harus menyesuaikan dengan lingkungan *framework* yang digunakan [9].

Sebuah *framework* selain menyediakan lingkungan pengembangan sendiri-sendiri juga menyediakan berbagai macam fungsi siap pakai yang dapat digunakan dalam pembuatan situs *web*. Fungsi tersebut merupakan pengembangan atau penyesuaian fungsi asli PHP agar lebih mudah digunakan atau agar lebih sesuai dengan kebutuhan pengguna.

Tujuan dari *framework CodeIgniter* ini menurut panduan penggunaan adalah untuk menghasilkan *framework* yang akan dapat digunakan untuk pengembangan proyek pembuatan situs *web* dengan cara penggunaan kode program secara manual, dengan menyediakan banyak sekali pustaka yang dibutuhkan dalam pembuatan situs *web*, dengan antarmuka yang sederhana dan struktur logika untuk mengakses pustaka yang dibutuhkan. *CodeIgniter* membiarkan *developer* untuk memfokuskan diri pada pembuatan situs *web* dengan meminimalkan pembuatan kode untuk berbagai tujuan pembuatan situs *web* [9]. Beberapa alasan kenapa menggunakan *framework CodeIgniter* diantaranya yaitu gratis, berjalan di PHP versi 4 dan 5, ringan dan cepat, memiliki arsitektur MVC, dokumentasi dan pustaka yang lengkap.

Model View Controller (MVC)

Model View Controller atau yang sering disebut dengan MVC merupakan arsitektur yang sangat berguna dalam melakukan pengembangan sebuah sistem. Terdapat 3 komponen dalam pola MVC dan interaksi yang terjadi [10].

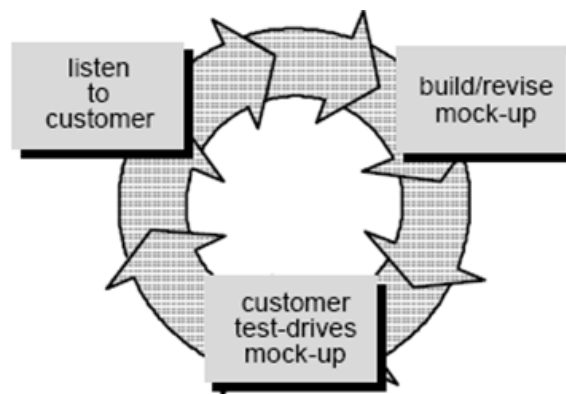


Gambar 1 Arsitektur model, view, controller [10]

Gambar 1 menunjukkan 3 komponen yang terdapat dalam pola MVC (*Model, View, Controller*) dan interaksi yang terjadi.

PERANCANGAN SISTEM

Model pengembangan sistem yang dipergunakan pada penelitian ini adalah model *prototype*. Model *prototype* merupakan suatu teknik untuk mengumpulkan informasi tertentu mengenai kebutuhan-kebutuhan informasi pengguna secara cepat. Dengan metode *prototype* ini, pengembang dan pelanggan dapat saling berinteraksi selama proses pembuatan sistem. Secara lengkap, alur model *prototype* dapat dilihat seperti pada Gambar 2.



Gambar 2 Prototype Model [11]

Gambar 2 merupakan gambaran tahapan umum dari model *prototype*. Penjelasan dari tahap – tahap model *prototype* sebagai berikut:

Listen to Customer

Pada tahapan pertama *listen to customer* adalah tahap mengumpulkan informasi tentang kebutuhan aplikasi yang akan dibangun. Pada tahap ini dilakukan wawancara dengan klaster.

Wawancara tentang kebutuhan klaster dilakukan terhadap klaster-klaster di Kabupaten Pemalang, Demak dan Wonosobo Jawa Tengah. Setelah itu dilakukan studi pustaka, menganalisis kebutuhan sistem yang akan dibangun kemudian melakukan rancangan sistem dengan cepat.

Build / Revise Mock-Up

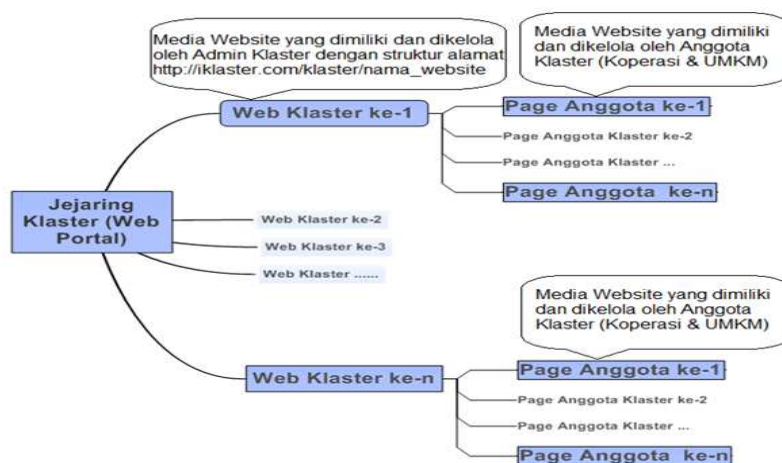
Tahapan selanjutnya dalam metode *prototype* yaitu *build/revise mock-up*. Pada tahap ini dilakukan pembuatan aplikasi secara cepat, yaitu mengimplementasikan hasil rancangan pada tahap sebelumnya menggunakan *framework CodeIgniter* dan arsitektur MVC. Dengan arsitektur MVC, pengembang dapat lebih fokus pada masing-masing bagian aplikasi, yaitu bagian *View* untuk implementasi antarmuka sistem, bagian *Model* untuk implementasi data, dan bagian *Controller* untuk implementasi pengendalian proses sistem.

Hasil pada tahap ini merupakan aplikasi *prototype* yang kemudian akan dipergunakan untuk proses tahapan model *prototype* selanjutnya.

Customer Test-Drives Mock-Up

Pada tahap ini dilakukan uji dan evaluasi aplikasi *prototype* oleh *user* yang bersifat sebagai *customer* yaitu Klaster. Evaluasi *prototype* digunakan untuk mendapatkan umpan balik apakah aplikasi sudah sesuai dengan kebutuhan *customer*. Tahap ini dilakukan kepada Klaster Jawa Tengah di Kabupaten Demak, Pemalang dan Wonosobo. Evaluasi dilakukan dengan cara wawancara dan mengisi kuesioner. Jika hasil uji dan evaluasi *prototype* belum sesuai dengan kebutuhan *customer*, maka dilakukan proses perbaikan dimulai kembali ke tahap awal dan dilanjutkan ke tahap berikutnya.

Perancangan Multi Website pada Sistem Jejaring Klaster

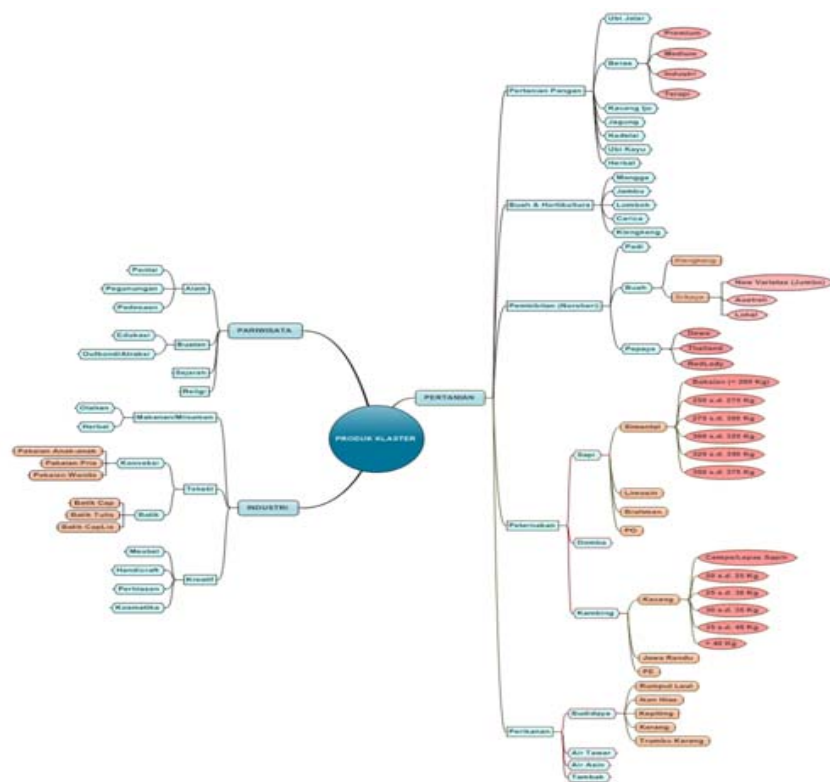


Gambar 3 Mapping Sistem Jejaring Klaster [12]

Pada Gambar 3 dapat dijelaskan bahwa sistem jejaring klaster mampu menyediakan *web* klaster atau alamat/situs *web* klaster berdasarkan klaster yang melakukan registrasi di Sistem Jejaring Klaster dan dapat secara langsung mengakses profil klaster dengan format URL http://domain/klaster/nama_website_klaster. Setelah terdaftar menjadi admin *Web* klaster, maka klaster dapat membuat halaman anggota sesuai dengan kebutuhan.

Perancangan Konsep *Social Network* pada Aplikasi *e-Commerce*

Hal utama pada suatu aplikasi *e-commerce* adalah bagaimana cara menampilkan produk supaya pengguna dapat dengan mudah memperoleh informasi produk yang diinginkan. Dalam penelitian ini, dihasilkan desain *mapping* produk-produk klaster seperti pada Gambar 4. *Mapping* digolongkan atas kelompok klaster berdasarkan ketentuan yang berlaku di Indonesia.

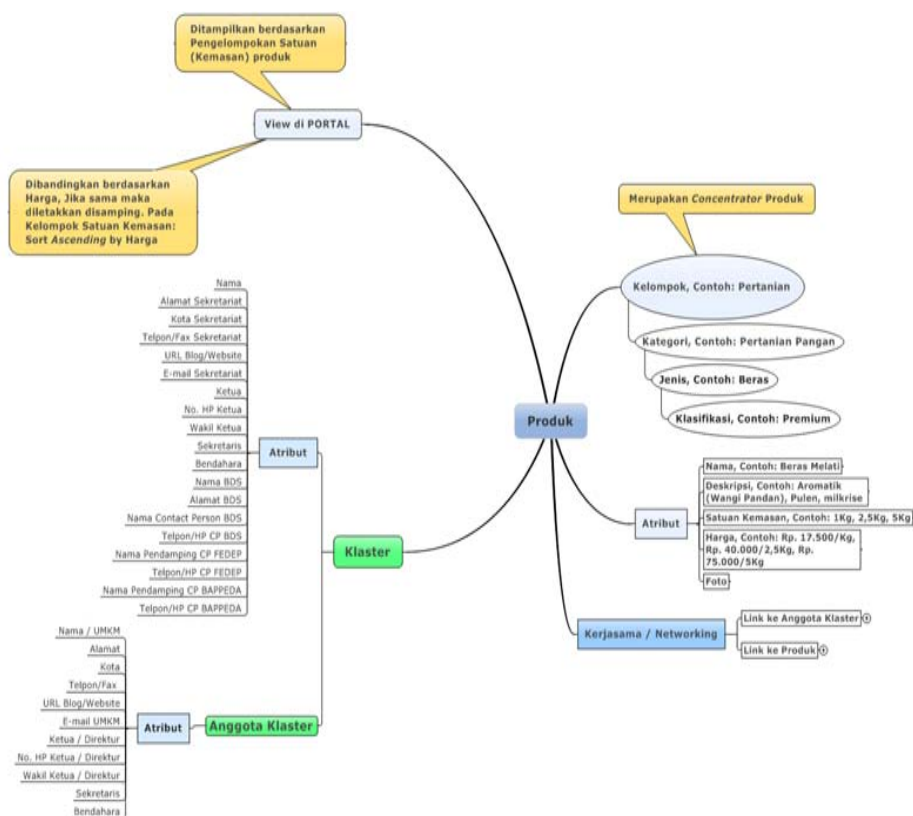


Gambar 4 Mapping Produk Klaster.[12]

Pada Gambar 4 dapat dijelaskan bahwa produk klaster terbagi dalam tiga jenis kelompok, yaitu Pertanian, Pariwisata dan Industri. Kelompok produk klaster memiliki beberapa sub golongan, yaitu Kategori, Jenis dan Klasifikasi. Misal, produk Beras Melati merupakan produk dari klaster kelompok Pertanian, kategori yaitu Pertanian Pangan, jenis yaitu Beras, lalu memiliki

klasifikasi yaitu Premium. Setelah produk masuk kedalam golongan Klasifikasi, maka produk-produk akan dikonsentrasikan berdasarkan Satuan, kemudian dilakukan proses *Sorting Ascending* berdasarkan Harga Produk. Setelah produk-produk klaster dikonsentrasikan berdasarkan proses tersebut, maka produk ditampilkan di halaman *Homepage* portal dalam bentuk *image*, nama produk dan harga.

Setelah diperoleh desain proses konsentrator produk, maka dilakukan desain atribut produk guna mendapatkan kesamaan pola atribut atau *field* tabel untuk semua produk klaster. Pada Gambar 5 merupakan *mapping* atribut produk klaster dan atribut klaster sebagai hasil penelitian.

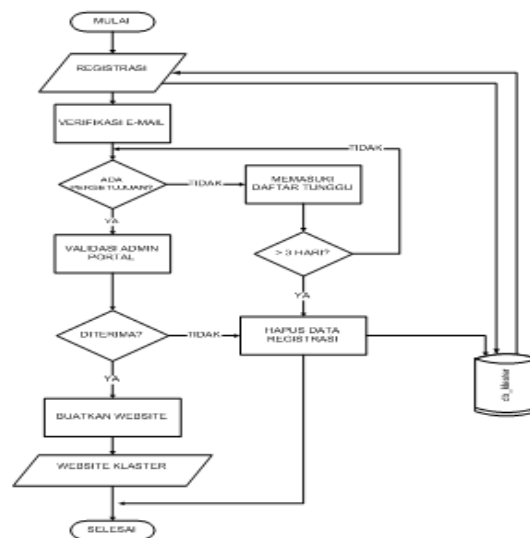


Gambar 5 Mapping Atribut Klaster [12]

Pada Gambar 5 dijelaskan bahwa setiap produk klaster memiliki atribut yaitu kelompok, kategori, jenis, klasifikasi, nama produk, deksripsi, satuan, harga dan foto. Sedangkan klaster memiliki atribut seperti yang terlihat pada Gambar 5. Untuk jejaring produk dibutuhkan tabel

berbeda yaitu tabel kerjasama atau *networking/jejaring* dengan atribut yaitu nama produk, jejaring ke nama produk, jejaring ke anggota klaster.

Diagram Alir Proses Pembuatan Website Klaster



Gambar 6 Diagram Alir Proses Registrasi Klaster[12]

Pada Gambar 6 terlihat bahwa proses registrasi diawali dari seorang pengurus klaster melakukan registrasi dengan cara mengisi *form* registrasi. Setelah mengisi data registrasi, maka pengurus klaster akan menerima verifikasi *e-mail*. Proses verifikasi berfungsi juga sebagai anti *spam* dari sistem jejaring. Pengurus klaster harus menerima verifikasi *e-mail* supaya data klaster dapat tersimpan pada *database* sistem. Jika pengurus klaster tidak melaksanakan verifikasi *e-mail* dalam kurun waktu 3 (tiga) hari, maka secara otomatis data registrasi klaster akan dihapus.

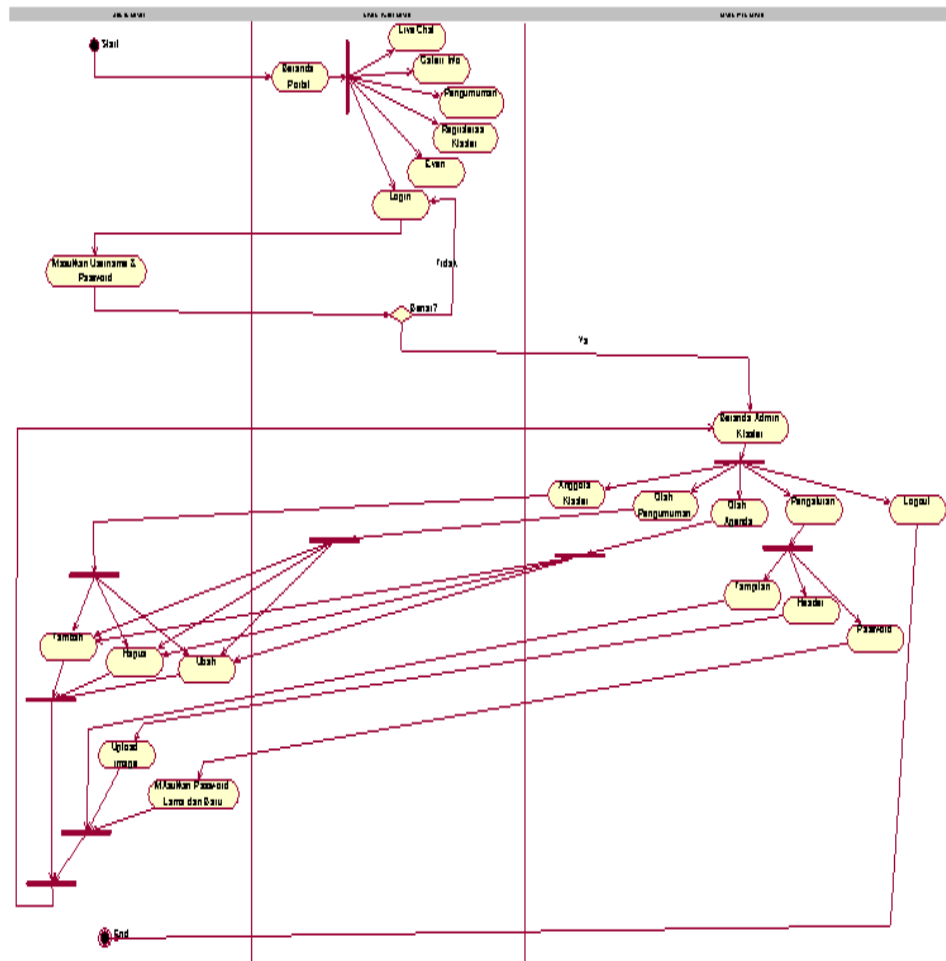
Setelah melaksanakan verifikasi *e-mail* maka data klaster akan masuk kedalam daftar tunggu validasi admin portal. Jika klaster mendapat validasi admin portal berupa ditolak maka data klaster dihapus. Jika mendapat validasi diterima oleh admin portal, maka klaster akan mendapatkan *website* klaster. Setelah mendapat *website* maka pengurus klaster tersebut secara otomatis telah terdaftar sebagai admin klaster. Sebagai admin klaster diijinkan untuk *login* dan dapat mendaftarkan para anggota klaster untuk dapat mengakses *website* klaster. Anggota klaster dapat mempergunakan *website* klaster untuk *upload* produk serta melakukan jejaring bisnis dengan klaster lain atau sesama anggota klaster.

Perancangan Aplikasi

Perancangan aplikasi *web* sistem jejaring klaster ini dirancang menggunakan UML (*Unified Modeling Language*) sebagai pemodelan sistem.

Gambar 7 menjelaskan bahwa admin portal memiliki hak akses secara penuh. Diantaranya adalah mengelola master data produk (kelompok produk, kategori produk, jenis produk, dan klasifikasi produk), *manage profil* klaster dan anggota klaster, *manage info*, *manage* pengumuman, *manage even*, *manage header*. Admin klaster di sini memiliki hak akses untuk *manage* anggota klaster, *manage* agenda, *manage* pengumuman dan *manage* pengaturan (tampilan, *header*, *password*). Anggota klaster yang memiliki hak akses untuk *manage* hasil produk (tambah produk, daftar *request*, tambah jejaring, lihat jejaring, edit produk, hapus produk). *Guest* yang memiliki hak akses hanya untuk input pemesanan, input komentar dan *view* info lengkap (lihat jejaring, lihat agenda klaster, lihat pengumuman klaster).

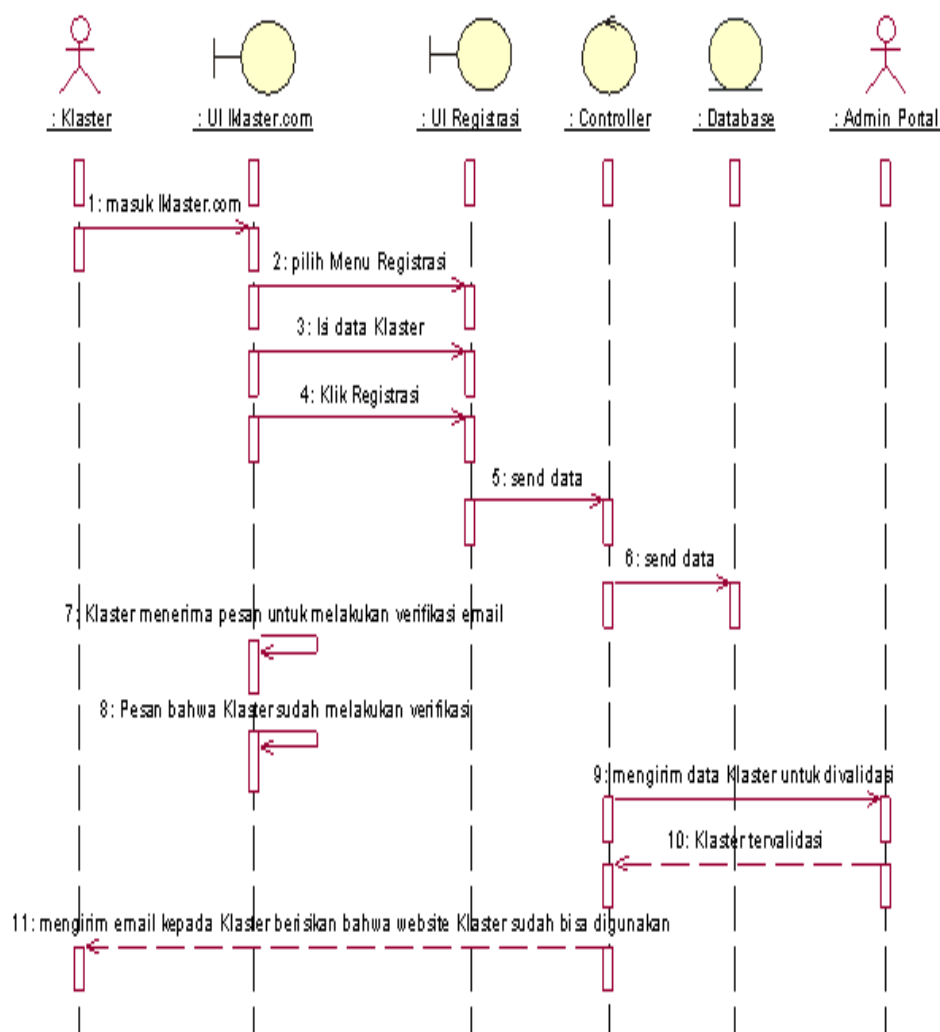
Activity Diagram



Gambar 8 Activity Diagram Admin Klaster[12]

Activity diagram menggambarkan proses-proses yang terjadi dari suatu aktivitas dimulai sampai berhenti.

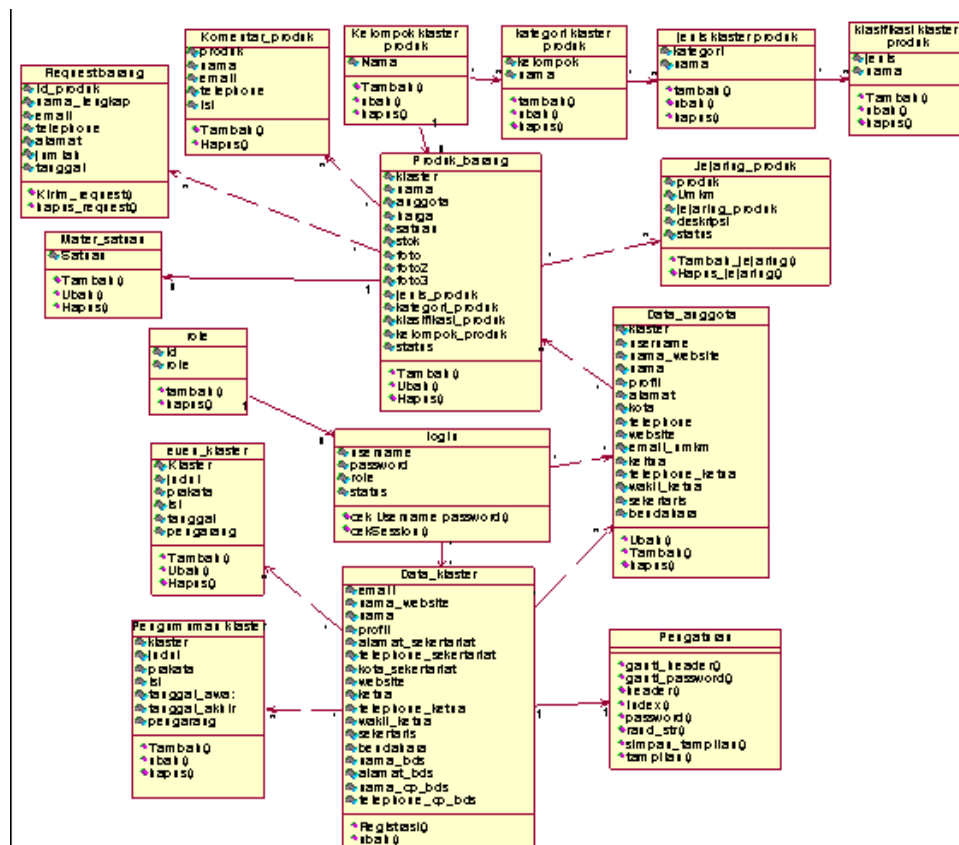
Gambar 8 menggambarkan aktivitas-aktivitas yang bisa dilakukan oleh admin klaster. Aktivitas-aktivitas tersebut diantaranya adalah *manage* anggota klaster, *manage* agenda, *manage* pengumuman dan *manage* pengaturan (tampilan, header, password). Sebelum mengakses menu-menu tersebut harus melakukan *login* terlebih dahulu.

Sequence Diagram

Gambar 9 Sequence Diagram Registrasi[12]

Sequence diagram menggambarkan interaksi antar obyek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. Gambar 9 merupakan *sequence diagram* untuk aktivitas sebagai *Guest* yang melakukan registrasi menjadi admin klaster.

Class Diagram



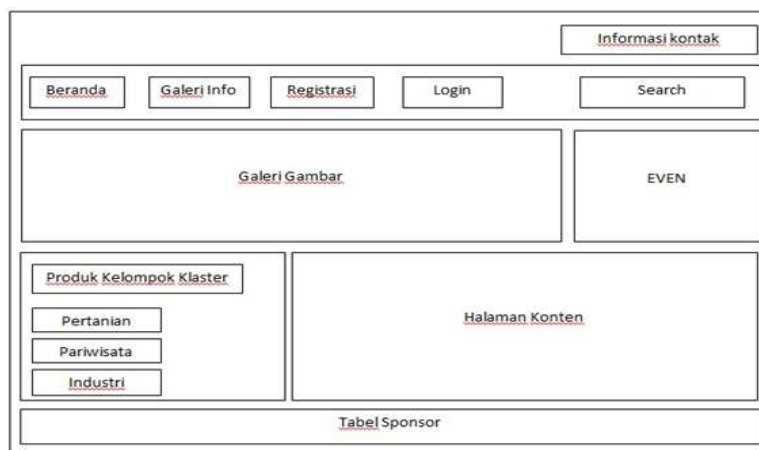
Gambar 10 Class Diagram System[12]

Class diagram system pada Gambar 10 merupakan perancangan *class* yang akan dibuat pada aplikasi tersebut yang terdapat pada *Controller* dan *Model*, meliputi : *class role*, *class login*, *class data_kluster*, *class pengumuman_kluster*, *class even_kluster*, *class pengaturan*, *class data_anggota*, *class jejaring_produk*, *class produk_barang*, *class master_satuan*, *class request_barang*, *class komentar_produk*, *class kategori_kluster_produk*, *class kelompok_kluster_produk*, *class jenis_kluster_produk*, *class klasifikasi_kluster_produk*.

Perancangan User Interface

Dalam perancangan sistem informasi Sistem Jejaring Kluster diperlukan *user interface* dalam penggunaannya. Perancangan *user interface* dilakukan untuk mendasari pembuatan tampilan aplikasi yang akan dibuat. Gambar 11 merupakan rancangan *user interface* Menu

Utama pada Sistem Jejaring Klaster. Sistem yang dibangun berbasis *web*, dengan *header*, sebuah sub menu dan isi *web* di dalamnya.



Gambar 11 User Interface Menu Utama[12]

HASIL DAN PEMBAHASAN

Setelah melakukan tahap perancangan, langkah selanjutnya adalah mengimplementasikannya. Pada tahap implementasi dihasilkan sebuah aplikasi yaitu Sistem Jejaring Klaster berbasis *Web* dan telah di-*hosting* dengan nama domain *iklaster.com*. *iklaster.com* dibuat melalui beberapa tahap perancangan *prototype*, dan tahap perancangan *prototype* dapat dilihat pada Dokumentasi *Prototyping*.

Dokumentasi *Prototyping*

Prototype I : Uji dan Evaluasi Sistem: 23 Feb 2012 di Kab. Demak

Pada *prototype I* pada *web portal* telah terdapat penggolongan produk, verifikasi *email* saat registrasi klaster, lihat jejaring. Dan kekurangan pada *prototype I* di *web portal* adalah belum adanya validasi Registrasi Klaster oleh Admin Portal, agenda dan pengumuman, pemesanan produk oleh *Guest*, komentar produk oleh *Guest*. Pada *prototype I website* admin Klaster telah terdapat menu Pengaturan : *header*, tampilan dan *password*, pengelolaan anggota Klaster. Dan kekurangan pada *prototype I* di *website* Klaster adalah belum adanya belum ada nama *website* Klaster yang diinginkan pada saat proses registrasi, *website* Klaster belum memiliki alamat *URL* sendiri, belum terdapat agenda dan pengumuman. Pada *prototype I* ditemukan khususnya di telah terdapat *add* jejaring dan lihat jejaring. Dan kekurangan pada *prototype I* di *website* anggota adalah belum adanya detail produk, lihat daftar pemesanan, notifikasi terima dan tolak jejaring, *image* produk atau UMKM/Klaster pada lihat jejaring.

Prototype II: Uji dan Evaluasi Sistem: 24 April 2012 : di Bapeda Wonosobo, 26 April 2012 : di Bapeda Pemasang.

Beranda portal sudah memiliki *Header* yang berbeda bertuliskan iKl@s. Pada *prototype* II di *web portal* telah terdapat Penggolongan Produk Verifikasi Email saat registrasi klaster Agenda, Pengumuman Lihat JejaringPemesanan/Request Produk oleh *Guest* dengan Login, adanya penambahan Even. Dan kekurangan pada *prototype* II di *web portal* adalah belum adanya Validasi Registrasi Klaster oleh Admin Portal Komentar Produk oleh *Guest*. Terdapat validasi Klaster oleh Admin Portal pada *prototype* ini, setelah diterima dalam proses validasi maka Klaster akan memiliki halaman *web*-nya, jika ditolak maka data klaster akan dihapus. Pada *prototype* II ditemukan khususnya di *website* Admin Klaster belum ada perubahan dari *prototype* I. Pada *prototype* II ditemukan khususnya di telah terdapat deskripsi detil produk dan stoknya. Dan kekurangan pada *prototype* I di *website* Anggota belum disempurnakan pada *prototype* II ini.

Prototype III: 11 May 2012

Pada Gambar *prototype* III, beranda portal sudah memiliki *Header* yang berbeda bertuliskan iKlaster.com. Pada *prototype* III ditemukan perubahan pada *web portal* telah terdapat penambahan *Contact Information* dan *website* ini sudah memiliki *Domain* dan sudah di *Hosting*. Pada *prototype* III ditemukan perubahan pada *website* Admin Klaster belum ada perubahan dari *prototype* I. Pada *prototype* III ditemukan perubahan pada *Website* Anggota belum ada perubahan dari *prototype* II.

Prototype IV: 3 Juni 2012

Pada *prototype* III ditemukan perubahan pada *web portal* yaitu sudah ada jumlah Produk pada Kelompok Klaster Homepage Portal, Sudah ditambahkan Verifikasi Registrasi Klaster. Pada *prototype* IV ditemukan perubahan pada *website* Admin Klaster yaitu sudah memiliki *URL Website* berdasarkan Registrasi Klaster dengan format *iklaster.com/klaster/nama_website*, sebagai contoh http://iklaster.com/klaster/tani_mahardika. Pada *prototype* IV ditemukan perubahan pada di *Website* Anggota yaitu dengan adanya Pemesanan/Request Produk oleh *Guest* sudah ada tanpa *Login*, Lihat Daftar Pemesanan, Image produk atau UMKM/ Klaster pada Lihat Jejaring, Notifikasi: Terima atau Tolak Jejaring, Fasilitas Komentar Produk oleh *Guest*.

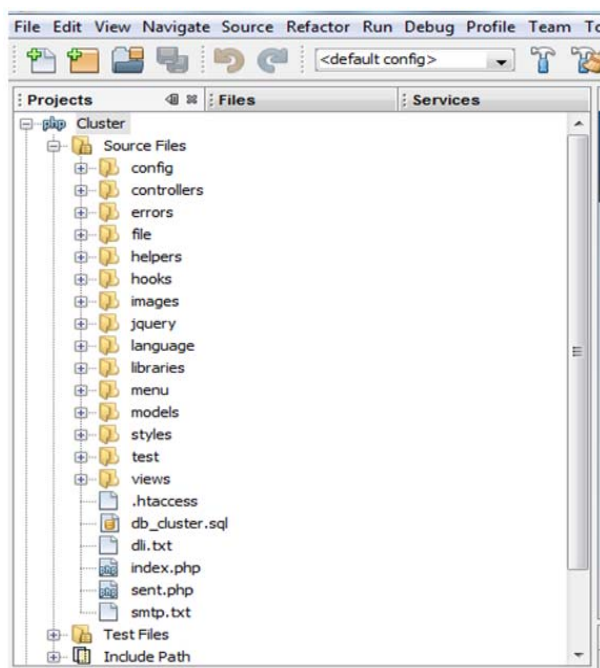
Prototype V: 6 Juli 2012

Perubahan yang terjadi pada Homepage Portal yaitu terjadi perubahan pada beranda portal tentang adanya informasi jumlah antrian Registrasi Klaster dibawah menu Kelompok Produk Klaster. Perubahan yang terjadi pada Admin Portal yaitu disediakannya fasilitas hapus Registrasi Klaster jika melakukan Aktivasi > 3 (lebih dari) hari pada *email* verifikasi yang dikirim oleh Admin Portal. Dari hasil evaluasi *prototype* V, pengguna menyatakan bahwa *prototype* V telah cukup memenuhi kebutuhan. Dengan demikian, proses berhenti karena tahapan dalam metode *prototyping* dinyatakan selesai jika pengguna menyatakan bahwa *prototype* yang dibangun sudah memenuhi kebutuhan pengguna.

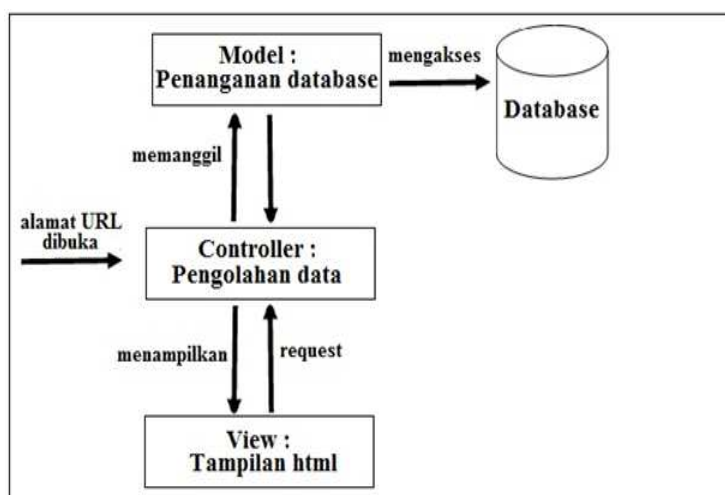
Hasil Implementasi

Implementasi Model View Controller (MVC)

Penerapan konsep MVC pada sistem yang dibuat terlihat pada Gambar 12.



Gambar 12 Penerepan MVC Pada Aplikasi yang Dibuat[12]



Gambar 13 Cara Kerja MVC Pada Aplikasi yang Dibuat

Mekanisme kinerja MVC pada aplikasi yang dibuat, terlihat pada Gambar 13 yaitu ketika suatu alamat URL dibuka maka mengakses *controller* (yang ada dalam *folder controllers*) untuk menjalankan fungsi. Penerapan skema arsitektur MVC pada aplikasi dapat dijelaskan dengan contoh yaitu pada saat proses registrasi. Pertama, *user* masuk ke halaman registrasi yang berada pada halaman depan *iklaster.com*. Pada saat melakukan klik pada registrasi akan dipanggil *controller* *registrasi.php* dan *controller* *registrasi.php* memanggil *view* *registrasi.php*. Setelah *user* mengisi data dan klik *submit*, pada *controller* dilakukan pengecekan *username* dan nama *website* apakah sudah digunakan apa belum, jika belum maka *controller* akan masuk ke bagian *model*. Pada bagian *model* data yang tadi telah di-input-kan akan disimpan pada *database*. Setelah menyimpan lalu kembali lagi ke *controller*. *Controller* akan mengirim *e-mail*, setelah mengirim *e-mail* maka *controller* akan melakukan *load view* yang berisi silahkan cek *e-mail* saudara. Setelah *user* membuka *email*, akan ada *link* yang setelah diklik masuk ke bagian *controller*. Pada bagian *controller* akan diambil *username* dan dicocokkan validasinya pada bagian *model*, jika cocok maka *controller* *load* halaman *view* yang berisi bahwa registrasi anda berhasil. Setelah itu tunggu validasi dari admin portal.

Tampilan halaman muka *iklaster.com* dapat dilihat pada Gambar 14.



Gambar 14 Tampilan Halaman Utama *iklaster.com*

Aplikasi yang dibangun berbasis *web* dan dibuat menggunakan *Framework Codeigniter*. *Framework Codeigniter* adalah *framework* yang berbasis *Model View Controller* yang memisahkan aplikasi logika dengan presentasi pada halaman *web*.

Model berisikan bagian-bagian yang mengelola data dengan sistem *query database*, mengambil dan menyimpan data, menghapus data, mengurutkan data, mencari data dan proses lainnya yang berhubungan dengan pengelolaan data. Bertugas untuk mengelola berbagai *model* yang diperlukan oleh aplikasi. Seperti yang terlihat pada Tabel 8 [12].

Tabel 8 Tabel Model

No	Class	Keterangan
1.	model_cluster.php	Query-query untuk mengambil data klaster
2.	mod_info.php	Query-query untuk mengambil data info
3.	mod_jejaring.php	Query-query untuk mengambil data jejaring
4.	mod_komentar.php	Query-query untuk mengambil data komentar
5.	mod_login.php	Query-query untuk mengambil data login
6.	mod_produk.php	Query-query untuk mengambil data produk
7.	mod_satuan.php	Query-query untuk mengambil data nilai satuan

View bertugas mengelola tampilan aplikasi, sesuai dengan namanya, merupakan bagian yang dapat dilihat dan dikelola oleh *user*. Bagian ini umumnya terdiri dari tombol-tombol, *tabs*, *check list*, *combo box*, teks, *audio*, *button*, *list*, *table* dan lain-lain. Implementasi yang dilakukan pada bagian *View* dikelompokkan dalam *package-package* untuk lebih memudahkan dalam pengembangan, yaitu dapat dilihat pada Gambar 15.

No.	Class	Keterangan
1.	Baca.php	Untuk memuat agenda, even, pengumuman, pengumuman klaster tertentu
2.	Depan.php	Untuk memuat halaman beranda
3.	Klaster.php	Untuk memuat nama website klaster pada URL
4.	Lihat.php	Untuk memuat lihat detail produk, jejaring umkm dan produk, jenis produk, kategori produk, kirim komentar, kirim request, klasifikasi, lihat produk, request produk
5.	Login.php	Untuk proses login
6.	Registrasi.php	Untuk proses registrasi
7.	Folder: Administrator	
8.	Cluster.php	Untuk mengatur klaster, bisa menambah dan menghapus klaster
9.	Depan.php	Untuk beranda depan admin portal
10.	Even.php	Untuk mengelola even portal
11.	info.php	Untuk mengelola info portal
12.	jenis_cluster.php	Untuk menambah, ubah, hapus jenis klaster
13.	kategori_cluster.php	Untuk menambah, ubah, hapus kategori klaster
14.	klasifikasi_cluster.php	Untuk menambah, ubah, hapus klasifikasi klaster
15.	logout.php	Untuk proses logout
16.	pengumuman.php	Untuk mengelola pengumuman portal
17.	satuan.php	Untuk mengelola satuan produk
18.	folder: anggota	
19.	depan.php	Untuk beranda depan anggota
20.	logout.php	Untuk proses logout
21.	produk.php	Untuk pengolahan data produk
22.	folder: cluster	
23.	anggota.php	Untuk mengelola anggota
24.	bantuan.php	Untuk load view bantuan.php
25.	depan.php	Untuk halaman beranda admin klaster
26.	logout.php	Untuk proses logout
27.	pengaturan.php	Untuk ubah password, header dan tampilan

Tabel 10. Tabel Controller

Implementasi *multiweb* pada satu *domain*

Proses pembuatan *website* bagi klaster pada aplikasi *iklaster.com* dilakukan pada saat registrasi klaster. Proses registrasi diatur dalam *Class* *registrasi.php* pada *folder Controller* dan kode program tersebut dapat dilihat pada Kode Program 1 sebagai berikut:

Kode Program 1 Perintah *Controller* pada *class* *registrasi.php*

```

1. function proses() {
2. if (count($_POST) > 0) {
3. $username = $this->input->post('username');
4. $password = $this->input->post('password');
5. $nama = $this->input->post('nama');
6. $profil = $this->input->post('profil');
7. $alamat_sekretariat = $this->input->post('alamat_sekretariat');
8. $kota_sekretariat = $this->input->post('kota_sekretariat');
9. $website = $this->input->post('website');
10. $ketua = $this->input->post('ketua');
11. $telepon_ketua = $this->input->post('telepon_ketua');
12. $wakil_ketua = $this->input->post('wakil_ketua');
13. $sekertaris = $this->input->post('sekertaris');
14. $bendahara = $this->input->post('bendahara');
15. $nama_website = $this->input->post('nama_website');
16. if ($this->login->checkNamaWebsite($nama_website) == FALSE) {
17. $data['kelompok'] = $this->produk->ListKelompokKategoriProduk();

```

```

18. $data['kategori'] = $this->produk->ListKategoriKlasterProduk();
19. $data['jenis'] = $this->produk->ListJenisKlasterProduk();
20. $data['klasifikasi'] = $this->produk->ListKlasifikasiKlasterProduk();
21. $data['error'] = "Nama website kembar, silahkan masukan Nama website yang lain";
22. $this->load->view('header_depan', $data);
23. $this->load->view('registrasi');
24. $this->load->view('footer');
25. if ($this->login->checkUsername($username) == TRUE) {
26. $reg_login = array("username" => $username, "password" => md5($password), "role" => 2, "status" => 2);
27. $tanggal_sekarang = date('Y-m-d');
28. $this->login->TambahLogin($reg_login);
29. $random = $this->rand_str();
30. $reg_data = array("email" => $username, "nama" => $nama, "profil" => $profil, "alamat_sekretariat" => $alamat_sekretariat, "kota_sekretariat" => $kota_sekretariat, "website" => $website, "ketua" => $ketua, "telepon_ketua" => $telepon_ketua, "wakil_ketua" => $wakil_ketua, "sekretaris" => $sekretaris, "bendahara" => $bendahara, "kode_aktivasi" => $random, "nama_website" => $nama_website, "tanggal" => $tanggal_sekarang);
31. $this->login->TambahDataCluster($reg_data);
32. $reg_css = array("css" => 1, "cluster" => $username);
33. $this->login->TambahCssCluster($reg_css);
34. $reg_header = array("cluster" => $username);
35. $this->login->TambahDataHeaderCluster($reg_header);
36. $this-> kirim_email($username, $random);
37. redirect('registrasi/lanjut');
38. } else {
39. $data['kelompok'] = $this->produk->ListKelompokKategoriProduk();
40. $data['kategori'] = $this->produk->ListKategoriKlasterProduk();
41. $data['jenis'] = $this->produk->ListJenisKlasterProduk();
42. $data['klasifikasi'] = $this->produk->ListKlasifikasiKlasterProduk();
43. $data['error'] = "Username kembar, silahkan masukan username yang lain";
44. $this->load->view('header_depan', $data);
45. $this->load->view('registrasi');
46. $this->load->view('footer');
47. } else {
48. redirect('registrasi');
49. }

```

Penjelasan Kode Program 1 sebagai berikut:

Baris 1 nama fungsi

Baris 2 mengecek apakah ada inputan dari user

Baris 3-15 mengambil inputan dari user

Baris 16-24 mengecek apakah nama website dari user masih tersedia

Baris 25 mengecek username masih tersedia atau tidak

Baris 26 membuat variable bertipe array dengan isi username, password yang dienkripsi md5, role = 2 (sebagai klaster), status=2 (belum mengirim verifikasi)

Baris 27 mengambil tanggal dari sistem

Baris 28 menambahkan data dari baris 26 ke dalam tbl_login

Baris 29 membuat variable random dengan isi karakter acak 32 karakter

Baris 30 membuat variable bertipe array yang berisikan data registrasi user

Baris 31 menambah data dari baris 30 kedalam tbl_data_cluster

Baris 32 membuat variable bertipe array yang berisikan data css default klaster

Baris 33 menambah data dari baris 32 kedalam tbl_css_cluster

Baris 35 menambah data yang berisi username pendaftar kedalam tbl_header_cluster

Baris 36 memanggil fungsi kirim e-mail dengan memasang e-mail dan kode aktivasi.

Baris 38-46 jika username sudah ada, tampilkan error dan inputan yang diisikan.

UJISISTEM

Uji Validitas Sistem

Untuk pengujian aplikasi jejaring klaster dilakukan dengan metode uji validitas. Pengujian dilakukan oleh *developer* seperti tercantum pada Tabel 11 sebagai berikut :

- a. Pengujian registrasi yaitu untuk mengecek apakah fungsional dari *form* registrasi telah berjalan dengan baik. Dari pengujian yang dilakukan didapatkan kesimpulan bahwa *form* registrasi telah berfungsi dengan baik.

Tabel 11. Pengujian Validitas Untuk Form Registrasi

No	Aktivitas dan Event	Input	Output	Status Pengujian
1.	Registrasi klaster dan klik tombol Registrasi pada menu utama	Data Klaster	Jika masih terdapat <i>field</i> belum diisi lengkap, maka muncul <i>alert</i> : Terdapat <i>field</i> inputan yang kosong harus diisi	Valid
2.	Penamaan <i>website</i>	Penamaan <i>Website</i>	a. Nama <i>website</i> yang dimasukkan sudah terdaftar sebelumnya akan ada pemberitahuan b. Dalam memberikan nama <i>website</i> tidak sesuai ketentuan tidak akan diijinkan	Valid

- b. Pengujian untuk menu-menu yang dapat diakses admin klaster meliputi *form* tambah anggota dan *form* pengaturan yang berfungsi untuk merubah *header*, *password* dan tampilan. Dari pengujian yang dilakukan didapatkan kesimpulan bahwa *form* tambah anggota dan *form* pengaturan telah berfungsi dengan baik.

Tabel 12. Pengujian Validitas Pengaturan dan Add Anggota

No	Aktivitas dan Event	Input	Output	Status Pengujian
1.	Tambah anggota	Data anggota klaster	Jika masih terdapat <i>field</i> belum diisi lengkap, maka muncul <i>alert</i> : Terdapat <i>field</i> inputan yang kosong harus diisi	Valid
2.	Ubah <i>header</i>	Gambar	Tampilan <i>header</i> berubah, jika tidak mengisi inputan maka tampilan <i>header</i> akan kosong	Valid
3.	Ubah <i>password</i>	Memasukan <i>password</i> lama dan dua kali <i>password</i> baru	a. Jika <i>password</i> lama salah maka muncul peringatan <i>password</i> salah b. Jika <i>password</i> lama salah maka muncul peringatan <i>password</i> salah	Valid
4.	Ubah tampilan	Memilih warna	Warna tampilan berubah	Valid

Uji Responden

Uji sistem ini didasarkan pada pengujian yang dilakukan oleh 25 (duapuluh lima) responden dari beberapa klaster di Kabupaten Pemalang, Wonosobo dan Demak yang merupakan para pengurus dan anggota klaster. Setelah uji sistem, para responden mengisi kuesioner yang berisi pertanyaan seputar penggunaan aplikasi Sistem Jejaring Klaster berbasis Web, yaitu sebagai admin portal *website* klaster dan admin *website* klaster.

Berdasarkan hasil uji sistem pada admin portal *website* klaster, diperoleh hasil bahwa sebanyak 26% responden menyatakan bahwa sistem sangat mudah digunakan dan 61% responden menyatakan bahwa sistem mudah digunakan, sementara sisanya menyatakan sangat sulit, sulit dan tidak tahu.

Sedangkan untuk uji sistem admin *website* klaster didapatkan hasil sebanyak 24% responden menyatakan bahwa sistem sangat mudah digunakan dan 66% responden menyatakan bahwa sistem mudah digunakan, sementara sisanya menyatakan sangat sulit, sulit dan tidak tahu.

SIMPULAN

Berdasarkan hasil perancangan dan implementasi Sistem Jejaring Klaster Berbasis Web Menggunakan Metode *Model View Controller*, maka dapat diperoleh kesimpulan yaitu :

- Desain *multi website* dengan memanfaatkan satu *domain* dengan cara setiap *page* didalam *website* memiliki kerangka tampilan yang sama sangat tepat guna memfasilitasi berbagai kelompok klaster. Setiap *website* klaster memiliki situs dengan format URL: *iklaster.com/klaster/nama_website* berdasarkan *nama_website* yang dimasukkan pada saat proses registrasi klaster.
- Berdasarkan *mapping* produk klaster dapat diperoleh desain atribut produk dan atribut klaster, yaitu dikelompokkan kedalam 4 (empat) golongan yang sama antara lain Kelompok, Kategori, Jenis dan Klasifikasi. Atribut produk dan atribut klaster tersebut dipergunakan sebagai dasar penerapan konsep jejaring sosial pada aplikasi *e-commerce* Sistem Jejaring Klaster. Struktur data yang sama sangatlah tepat jika menggunakan *framework* dengan metode MVC karena lebih efisien dalam proses implementasi *web* klaster ini.
- Sedangkan data klaster dan produk anggota klaster yang terdapat didalamnya dikelola oleh *class-class model*. Manajemen prosesnya dikelola oleh *class-class controller*. Pada *database*, informasi klaster disimpan dalam tabel *data_cluster*, tampilan *website* klaster pada tabel *css_cluster* dan tampilan *header* klaster disimpan dalam tabel *header_cluster*. Tampilan pada setiap *page* dikelola oleh *class-class* pada *folder view*.
- Penggunaan *framework CodeIgniter* membantu memudahkan dalam implementasi aplikasi karena *framework CodeIgniter* mendukung arsitektur *Model View Controller* (MVC) yang memisahkan antara *model*, *view* dan *controller*.

DAFTAR PUSTAKA

- [1] Bank Indonesia, 2007, *Pilot Project Pengembangan Klaster Mebel Rotan Di Trangsan Kec. Gatak Kab. Sukoharjo Jawa Tengah*, <http://www.bi.go.id/NR/rdonlyres/7D54C99C-C1E2-40E4-A428-309F234546DE/10372/Boks3.pdf>, Diakses tanggal: 8 April 2012

- [2] FPESD Jateng, 2010, Rakor FPESD Jawa Tengah, <http://www.fpesd.org/?Home>, Diakses tanggal: 8 April 2012
- [3] Saputra, Irwan, 2011, *Multimedia Pembelajaran Bahasa Inggris Menggunakan Konsep Jejaring Sosial Berbasis Web*, <http://dir.unikom.ac.id/s1-final-project/fakultas-teknik-dan-ilmu-komputer/teknik-informatika/2011/jbptunikompp-gdl-irwansaput-26098/1-1.uniko-r.pdf/pdf/1-1.uniko-r.pdf>. Diakses tanggal: 4 Juni 2012
- [4] Wijaya, I Wayan Gede Suma, Eko Heri Susanto, M. Kom, 2012, *Membangun Website e-Commerce dengan Teknik MVC Menggunakan Framework CodeIgniter*, http://www.general-files.com/download/g56068415h17i0/Jurnal_Jikom_-_Membangun_Website_E-Commerce_dengan_Teknik_MVC_Menggunakan_Framework_CodeIgniter.pdf.html. Diakses tanggal: 6 Juni 2012
- [5] Jogiyanto, 2003, *Pengertian Sistem Informasi*, Jogjakarta : Skripta Media.
- [6] Porter, M.E., 1998, *On Competition*. Boston: Havard Business School Publishing.
- [7] Ridwan, Muhammad, *Jejaring Sosial (Social Networking)*, <http://www.ridwanforge.net/blog/jejaringsosial-social-networking>, Diakses tanggal: 28 Juni 2012
- [8] Upton, David. 2007. *CodeIgniter for Rapid PHP Application Development Improve your PHP coding productivity with the free compact open-source MVC CodeIgniter framework*. Birmingham : Packt Publishing Ltd.
- [9] *User_guide*, <http://Codeigniter.com>, Diakses tanggal: 3 januari 2012
- [10] eNode Inc, 2002, *Arsitektur Model, View, Controller*. Yogyakarta : Skripta Media.
- [11] Pressman, 2001, *Software Engineering: A Practicioner's Approach 5th Edition*, America : Mc. Graw Hill.
- [12] Supriyadi, dkk, 2012, *Rancang Bangun Sistem Jejaring Klaster Berbasis Web Dengan Pendekatan Model E-Commerce Marketplace*, Salatiga: FTI-UKSW