

Makalah Seminar Tugas Akhir

SISTEM INFORMASI DEMOGRAFI PENDUDUK KOTA SEMARANG BERBASIS FRAMEWORK CODE IGNITER

R.Kibar Aji Seno¹⁾, Ir. Kodrat Iman Satoto, M.T.²⁾, Rinta Kridalukmana, S.Kom., M.T.²⁾

Program Studi Sistem Komputer, Fakultas Teknik, Universitas Diponegoro

Jln. Prof. Sudharto, Tembalang, Semarang, Indonesia

Email: kibar.seno@gmail.com

Abstrak — *Data kependudukan ialah salah satu data yang rumit dalam pengolahannya, karena meliputi data cukup banyak yaitu data matapencaharian, data kepadatan penduduk, data jenis kelamin penduduk, data jumlah keluarga, data pendidikan penduduk dan sebagainya yang meliputi sebagai data monografi. Sering kali terjadi kesulitan dalam pencatatan dan pengolahan data tersebut. Bahkan terkadang data penduduk tersebut tidak tersimpan dan diolah dengan baik sehingga setiap membutuhkan data tersebut sering dilakukannya pendataan ulang yang tentu itu akan membuat pemborosan dalam segi waktu, tenaga dan biaya.*

Oleh karena itu dibutuhkan suatu teknologi informasi yang dapat membantu dalam pengolahan data tersebut. Sistem informasi dengan berbasis web ialah salah satu aplikasi yang cocok untuk membantu masalah ini, karena dengan berbasis web pengolahan data dapat di lakukan dimana saja tanpa harus melakukan pemasangan aplikasi tersebut. Tetapi karena berbasis web ini pula, perlu adanya perhatian khusus dalam pengembangannya yaitu pada bagian keamanan aplikasi tersebut. Sehingga dalam pengembangan aplikasi ini digunakan framework codeigniter yang dikenal cukup baik dalam penanganan keamanannya serta memiliki libraries yang cukup lengkap dan performance yang handal.

Sistem informasi kependudukan republik indonesia ialah sistem informasi yang dikembangkan dengan menggunakan framework codeigniter dan database mysql. dengan begitu Sistem informasi diharapkan dapat membantu pengolahan data kependudukan sehingga data-data tersebut menjadi optimal dan terstruktur dengan baik.

Kata Kunci : *aplikasi, teknologi informasi, sistem informasi, framework, codeigniter, PHP, mysql,*

I. Pendahuluan

Perkembangan ilmu pengetahuan dan teknologi yang semakin kompleks mampu mempengaruhi pola pikir manusia. Kemajuan ini telah mendorong manusia untuk berusaha mengatasi segala permasalahan yang timbul di sekitarnya. Terutama dalam bidang pendataan, peranan basis data sangat penting. Pemrosesan basis data menjadi perangkat andalan yang kehadirannya sangat diperlukan oleh berbagai institusi dan perusahaan. Basis data tidak hanya mempercepat dalam memperoleh informasi, tetapi

juga dapat meningkatkan pelayanan kepada masyarakat sehingga efektifitas waktu dapat dimaksimalkan.

Pengolahan data kependudukan ialah salah satu permasalahan yang perlunya teknologi informasi karena dengan jumlah data yang begitu besar dan perbaruan data yang terus menerus sehingga perlu pengolahan yang maksimal. Data kependudukan selama ini hanyalah sebagai data semu yang menunggu untuk diolah, banyak data kependudukan hanya di tampung saja tanpa ada bentuk pengolahan, sehingga saat bila membutuhkan data kependudukan untuk keperluan lain maka akan melakukan pengambilan data kembali data kependudukan, dan itu sangatlah pemborosan dari segi waktu, tenaga dan uang. Oleh karena itu di butuhkan sebuah teknologi informasi untuk membantu menyelesaikan masalah ini, dan sistem informasi adalah teknologi yang tepat dalam penyelesaiannya.

Tujuan dari tugas akhir ini adalah untuk merancang dan membuat perangkat lunak sistem informasi kependudukan se-tingkat Kota Semarang. Sistem Informasi Kependudukan Kota Semarang dengan *Framework Codeigniter* dapat memberikan manfaat untuk mempermudah dalam mengelolah data kependudukan yang ada.

II. Dasar Teori

2.1 Pengertian Sistem Informasi

Sistem adalah sekelompok bagian-bagian (alat dan sebagainya) yang bekerja bersama-sama untuk melakukan suatu maksud. Informasi adalah makna atau pengertian yang dapat diambil dari suatu data dengan menggunakan konversi-konversi yang umum digunakan di dalam representasinya. Suatu sistem adalah jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu. Prosedur adalah suatu urutan-urutan operasi tulis-menulis dan biasanya melibatkan beberapa orang dalam satu atau lebih departemen^[4].

Informasi adalah kumpulan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerima. Banyak atribut dalam konsep informasi yang membantu kita dalam mengidentifikasi dan mendeskripsikan kebutuhan-kebutuhan informasi yang spesifik.

1) Mahasiswa Sistem Komputer UNDIP

2) Dosen Sistem Komputer UNDIP

2.2 Framework Code Igniter

Framework merupakan kumpulan dari *library* (*Class*) yang bisa diturunkan atau bisa langsung dipakai fungsinya oleh modul-modul atau fungsi yang akan dikembangkan.

Dalam perkembangannya, *framework* memiliki banyak jenis salah satunya adalah PHP *Framework*. PHP *Framework* merupakan sebuah lingkungan pengembangan aplikasi berbasis PHP yang berisi sejumlah *Class* yang telah dibuat agar digunakan kembali untuk membuat aplikasi. Berbagai jenis *Class* ini tergantung tujuan dari PHP *framework* tersebut dan PHP *framework* yang satu dengan yang lain kemungkinan berbeda. Sebagian besar PHP *framework* yang tersedia saat ini berbasis MVC (*model view controller*) yang merupakan model PHP *framework* modern. Jika menulis aplikasi dengan suatu *framework*, maka harus mengikuti aturan dari *framework* tersebut dan tidak boleh asal. Karena inilah biasanya perusahaan menggunakan *framework* daripada membiarkan seorang *programmer* dengan gaya pemrogramannya sendiri.

Code Igniter adalah aplikasi *open source* yang berupa *framework* dengan model MVC (*model view controller*) untuk membangun website dinamis. Dengan menggunakan PHP Code Igniter akan memudahkan *developer* untuk membuat aplikasi web dengan cepat dan mudah dibandingkan dengan membuatnya dari awal.

2.3 PHP

PHP merupakan singkatan dari *PHP Hypertext Preprocessor* yang merupakan bahasa berbentuk kode yang ditempatkan pada *server* dan diproses di *server*. Hasilnya kemudian dikirimkan ke *browser* klien.

Secara khusus, PHP dirancang untuk membentuk *web* dinamis. Misalnya, Anda dapat menampilkan isi basisdata ke halaman *web*. Pada prinsipnya, PHP memiliki fungsi yang sama dengan skrip-skrip seperti ASP (*Active Server Page*), *Cold Fusion*, ataupun *Perl*^[6].

Kelahiran PHP bermula saat Rasmus Lerdorf membuat sejumlah kode *Perl* yang dapat mengamati siapa saja yang melihat-lihat daftar riwayat hidupnya, yakni pada tahun 1994. Kode-kode ini selanjutnya dikemas menjadi *tool* yang disebut "*Personal Home Page*". Paket inilah yang menjadi cikal bakal PHP. Pada tahun 1995, Rasmus menciptakan PHP/FI Versi 2.0. Pada rilis ini *interpreter* PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan. Pada versi inilah pemrograman dapat menempelkan kode terstruktur di dalam tag HTML. Yang menarik, kode PHP juga dapat berkomunikasi dengan basisdata dan melakukan perhitungan-perhitungan yang kompleks sambil jalan^[6].

2.4 MySQL

Dewasa ini peranan teknologi pengolahan basisdata sangatlah menonjol. Basisdata memiliki peranan sentral dalam sistem informasi. Mengingat peranannya yang cukup besar, banyak sistem basisdata ditawarkan. Salah satunya adalah *MySQL*. *MySQL*

dikembangkan sekitar tahun 1994 oleh sebuah perusahaan pengembang perangkat lunak dan konsultan database *MySQL AB* yang bertempat di Swedia. Waktu itu perusahaan tersebut masih bernama *TcX DataKonsult AB*, dan tujuan awal dikembangkannya *MySQL* adalah untuk mengembangkan aplikasi berbasis web pada klien^[2].

MySQL merupakan basisdata yang menggunakan konsep RDBMS, sebuah tabel merupakan struktur penyimpanan dasar. Satu tabel atau lebih membentuk sebuah *relational database*^[2].

MySQL adalah *Relational Database Management System* (RDBMS) yang didistribusikan secara gratis di bawah lisensi GPL (*General Public Licence*). Dimana setiap orang bebas untuk menggunakan *MySQL*, namun tidak boleh dijadikan produk turunan yang bersifat *closed source* atau komersial^[2].

Keandalan suatu sistem basisdata dapat diketahui dari cara kerja *optimizer*-nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh user maupun program-program aplikasinya. Sebagai basisdata *server* lainnya dalam *query* data. Hal ini terbukti untuk *query* yang dilakukan oleh *single user*, kecepatan *query MySQL* bisa sepuluh kali cepat dari *PostgreSQL* dan lima kali lebih cepat dibandingkan *Interbase*. Kemampuan yang cukup menakjubkan untuk sebuah perangkat lunak gratisan^[10].

Keistimewaan dari *MySQL* adalah *Portabilitas* yaitu *MySQL* dapat berjalan stabil pada berbagai sistem operasi diantaranya seperti *Windows*, *Linux*, *FreeBSD*, *Mac OS X Server*, *Solaris*, *Amiga*, *HP-UX* dan masih banyak lagi.

1. *Open Source* yaitu *MySQL* didistribusikan secara gratis, dibawah *lisensi GPL* sehingga kita dapat menggunakannya secara cuma-cuma^[10].
2. *Multiuser* yaitu *MySQL* dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah atau konflik. Hal ini memungkinkan sebuah basisdata *server MySQL* dapat diakses klien secara bersamaan^[10].
3. *Skalabilitas* dan *Limit* yaitu *MySQL* mampu menangani basisdata dalam skala besar, dengan jumlah rekord lebih dari 50 juta dan 60 ribu tabel serta 5 miliar baris. Selain itu, batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya^[10].

III. PERANCANGAN SISTEM

3.1 Tahap Perancangan Sistem

Sebuah sistem informasi perlu dilakukannya perancangan yang matang agar dapat beroperasi dengan baik. Layaknya membuat sebuah bangunan tentulah dibutuhkan sebuah cetak biru agar bangunan yang akan dibuat nantinya dapat tepat sasaran dan tepat guna. Penelitian sistem informasi ini, penulis menggunakan metode *waterfall* yang terdiri dari beberapa tahap yaitu analisa kebutuhan (*requirements*), analisa (*analysis*), desain (*design*), implementasi (*implementation*) dan pengujian (*testing*).

1) Mahasiswa Sistem Komputer UNDIP

2) Dosen Sistem Komputer UNDIP

3.1.1. Analisis Kebutuhan

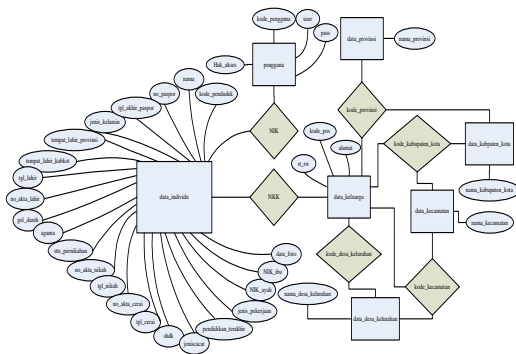
Pada analisis kebutuhan bertujuan untuk mengumpulkan kebutuhan-kebutuhan pengguna dan kemudian mentransformasikan ke dalam sebuah deskripsi yang jelas dan lengkap. Pada tahap ini berisi tahapan kerja dari instansi dan juga tabel yang berisi kesimpulan dari tahapan, disertai dengan kendala yang dialami. Pada bagian akhir tahap ini pula terdapat keinginan pengguna yang berdasar pada kendala yang ada.

3.1.2. Analisis Sistem

Tahap analisis sistem bertujuan untuk menjabarkan segala sesuatu yang nantinya akan ditangani oleh perangkat lunak. Tahapan ini berisi analisa dari permasalahan yang ada pada tahap sebelumnya, meliputi aktor yang terlibat dalam sistem, kebutuhan fungsional dan kebutuhan aplikasi yang muncul berdasarkan kendala yang ada.

3.1.3. Rancangan / Desain

Tahap perancangan perangkat lunak (*Design*) merupakan proses yang berfokus pada beberapa atribut yang ada di perangkat lunak. Proses ini menerjemahkan kebutuhan ke dalam sebuah model perangkat lunak dan berdasarkan analisis sistem sebelumnya menghasilkan rancangan yang berisi ERD (*Entity Relationship Diagram*), *Use Case*, desain antar muka, dan juga diagram alir dari algoritma berjalannya perangkat lunak. Berikut adalah bentuk ERD seluruh relasi entitas data.



Gambar 1. Relasi seluruh entitas data

Tahap selanjutnya ialah membuat UML dengan pertama melakukan definisi aktor, kemudian definisi use case serta membuat diagram use case, objek dan sekuen. Berikut ini adalah deskripsi pendefinisian aktor pada aplikasi Sistem Informasi Statistik Kependudukan Kota Semarang. Deskripsi pendefinisian tugas dari masing-masing aktor akan dijelaskan pada Tabel 1.

Tabel 1. Deskripsi aktor

No.	Aktor	Deskripsi
1.	Super Admin	Orang yang bertugas dan memiliki hak akses untuk melakukan pengelolaan akun pengguna sistem.

2.	Admin	Orang yang bertugas dan memiliki hak akses untuk melakukan pengelolaan data kependudukan. Meliputi penambahan, pembaruan dan penghapusan data
----	-------	---

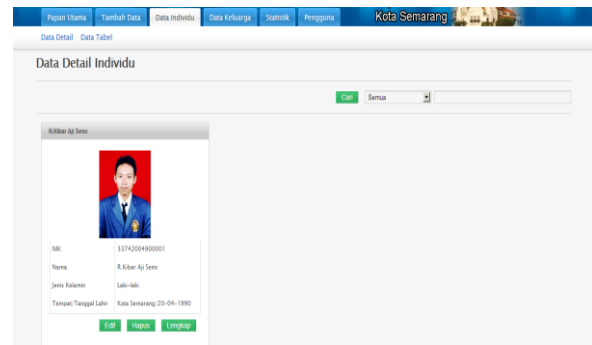
Kemudian setelah mendefinisikan aktor yang terlibat, maka mulailah penyusunan definisi use case. Berikut ialah deskripsi use case pada aplikasi.

Tabel 2. Deskripsi use case

No.	Use Case	Deskripsi
1.	Login	Merupakan proses validasi hak akses dari pengguna yang ingin menggunakan fasilitas sistem.
2.	Melihat semua data pengguna	Merupakan proses untuk melihat semua akun pada data pengguna
3.	Menambah data pengguna	Merupakan proses menambah akun pada data pengguna.
4.	Mengubah data pengguna	Merupakan proses mengubah akun data pengguna
5.	Menghapus data pengguna	Merupakan proses penghapusan salah satu akun pada data pengguna
6.	Menambah data wilayah	Merupakan proses penambahan data wilayah yang meliputi data provinsi, kabupaten/kota dan kecamatan
7.	Pengubahan data wilayah	Merupakan proses pengubahan atau pembaruan salah satu data wilayah
8.	Penghapusan data wilayah	Merupakan proses penghapusan salah satu data wilayah yang ada
9.	Menambah data individu	Merupakan proses menambah data individu baru
10.	Menambah data keluarga	Merupakan proses menambah data keluarga baru
11.	Melihat semua data individu	Merupakan proses untuk melihat semua data individu dengan bentuk detail atau tabel
12.	Pencarian data individu	Merupakan proses untuk melakukan pencarian data individu yang diinginkan
13.	Melihat semua data keluarga	Merupakan proses untuk melihat semua data keluarga dengan bentuk detail atau table
14.	Pencarian data keluarga	Merupakan proses untuk melakukan pencarian data

		keluarga yang diinginkan
15	Melihat detail data individu	Merupakan proses untuk melihat salah satu data individu secara lengkap
16	Mencetak data detail individu	Merupakan proses untuk mencetak detail dari data individu menjadi PDF
17	Melihat detail data keluarga	Merupakan proses untuk melihat salah satu data keluarga secara lengkap
18	Mencetak data detail keluarga	Merupakan proses untuk mencetak detail dari data keluarga menjadi PDF
19	Mengubah data individu	Merupakan proses perubahan salah satu data individu
20	Mengubah data keluarga	Merupakan proses perubahan salah satu data keluarga
21	Menghapus data individu	Merupakan proses menghapus salah satu data individu.
22	Melihat statistik data	Merupakan proses untuk melihat statistik data dari data individu dan data keluarga menjadi data kependudukan
23	Mensotir statistik data	Merupakan proses untuk pemyotiran statistic data yang diinginkan
24	Mencetak statistic data	Merupakan proses untuk mencetak hasil statistik data ke PDF

akan berinteraksi langsung ke pengguna aplikasi ini. Berikut ialah salah satu bentuk antarmuka yang ada dalam aplikasi.

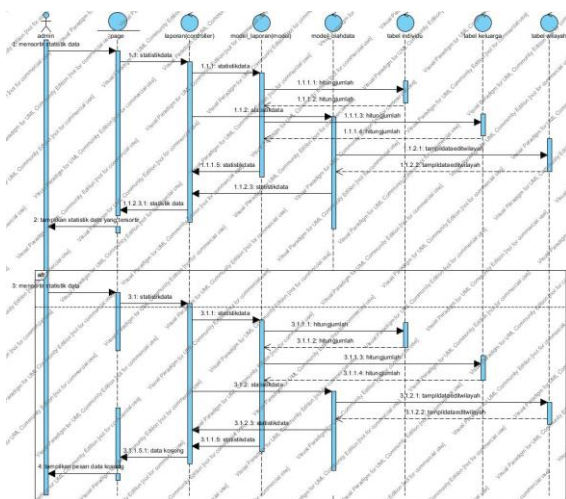


Gambar 3. Halaman data detail individu

Code igniter merupakan *framework* PHP yang penulis gunakan untuk melakukan pengembangan aplikasi dengan metode MVC (*Model, View, Controller*). MVC ini sekarang sering digunakan dalam beberapa *framework*, tidak hanya pada code igniter. Secara sederhana, MVC yang telah terbagi menjadi tiga bagian memiliki fungsi masing-masing.

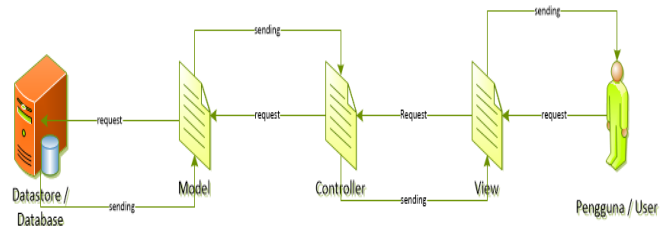
1. *Model* merupakan bagian yang bertindak sebagai komunikator ke *database* yang pada aplikasi ini adalah *database* MySQL.
2. *View* merupakan bagian yang bertindak sebagai antarmuka kepada pengguna. Dengan kata lain, *view* adalah tampilan dari aplikasi.
3. *Controller* merupakan bagian yang bertindak untuk mengatur komunikasi antara *view* dan *model*.

Use case tersebut selanjutnya dideskripsikan ke dalam diagram sekuen yaitu alur / tingkah laku yang akan dilakukan oleh aktor pada sistem. Berikut adalah salah satu contoh diagram sekuen yang ada dalam aplikasi..



Gambar 2. Diagram sekuen mensotir statistik data

Antarmuka merupakan salah satu hal yang penting dalam perancangan, karena bagian inilah yang kemudian



Gambar 4. Cara kerja MVC Code Igniter

IV. Implementasi dan Pengujian

4.1. Implementasi

Berdasarkan rancangan dan kebutuhan sistem, proses implementasi dibagi menjadi implementasi tabel dan implementasi program. Implementasi tabel yaitu mengimplementasikan rancangan tabel ke dalam *database* sesuai dengan analisa sebelumnya menggunakan MySQL. Sedangkan implementasi program berisi potongan kode program yang dibuat agar program berjalan sesuai dengan rancangan analisa di BAB III.

4.1.1 Implementasi Tabel

Aplikasi sistem informasi kependudukan kota semarang ini menggunakan basis data MySQL untuk media penyimpanan datanya. Dalam pengembangan aplikasi berbasis web, biasanya pada *developer* menggunakan aplikasi local server seperti XAMPP, Appserv, WampServer dan lainnya yang merupakan aplikasi yang di dalamnya ada beberapa aplikasi yang dibutuhkan untuk membuat server kecil (di komputer pribadi). Untuk implementasi ini, penulis menggunakan aplikasi XAMPP yang di dalamnya ada Apache, PHP dan MySQL. XAMPP secara *default* akan dimatikan, oleh karena itu harus dijalankan terlebih dahulu. Pembuatan basis data dengan aplikasi ini dapat dilakukan dengan menggunakan *browser* seperti google chrome dan Mozilla firefox. Pada *address bar* di *browser*, tuliskan alamat <http://localhost/phpmyadmin> dan menu PHPmyadmin sebagai antarmuka basis data akan muncul. Langkah selanjutnya sebelum membuat tabel yaitu membuat nama *database*, untuk hal ini penulis memberikan nama dengan “siks” yang terdiri dari 6 tabel.

Setelah *database* berhasil dibuat, langkah selanjutnya adalah membuat tabel dengan atribut yang sudah dirancang sebelumnya. Dalam *phpmyadmin*, seorang *developer* tidak harus menggunakan kode untuk membuat tabel-tabel tersebut, tetapi hanya perlu mengklik dari menu-menu yang sudah disediakan.

Berikut ialah salah satu tabel dalam aplikasi

Tabel 3. Tabel data individu

#	Name	Type	Collation	Attributes	Null	Default
1	kode_penduduk	varchar(50)	latin1_swedish_ci		No	None
2	NIK	varchar(20)	latin1_swedish_ci		No	None
3	NKK	varchar(20)	latin1_swedish_ci		No	None
4	nama	varchar(200)	latin1_swedish_ci		No	None
5	no_paspor	int(11)			No	None
6	tgl_akhir_paspor	date			No	None
7	jenis_kelamin	varchar(10)	latin1_swedish_ci		No	None
8	tempat_lahir_provinsi	varchar(2)	latin1_swedish_ci		No	None
9	tempat_lahir_kabkot	varchar(2)	latin1_swedish_ci		No	None
10	tgl_lahir	date			No	None
11	no_akta_lahir	int(11)			No	None
12	gol_darah	varchar(3)	latin1_swedish_ci		No	None
13	agama	varchar(10)	latin1_swedish_ci		No	None
14	stts_pernikahan	varchar(15)	latin1_swedish_ci		Yes	NULL
15	no_akta_nikah	int(11)			No	None
16	tgl_nikah	date			No	None
17	no_akta_cerai	int(11)			No	None
18	tgl_cerai	date			No	None
19	shdk	varchar(20)	latin1_swedish_ci		No	None
20	jeniscacat	varchar(100)	latin1_swedish_ci		No	None
21	pendidikan_terakhir	varchar(200)	latin1_swedish_ci		No	None
22	jenis_pekerjaan	varchar(200)	latin1_swedish_ci		No	None
23	NIK_ayah	int(16)			No	None
24	NIK_ibu	int(16)			No	None
25	data_foto	text	latin1_swedish_ci		No	None

4.1.2 Implementasi Program

Setelah *database* selesai dibentuk beserta tabel-tabel di dalamnya, selanjutnya yaitu implementasi program. Pada implementasi program, dibagi menjadi dua tahap yaitu membentuk tampilan / *user interface* dan sistem kerja aplikasi. Karena code igniter menggunakan prinsip MVC, maka sistem kerja aplikasi dikelola oleh *model* dan *controller* sedangkan *user interface* dikelola oleh *view*.

Di bawah ini beberapa tampilan *source code* dari penggalan *controller*, *model* dan *view*.

1. Membuat *model* *model_olahdata.php*
Model *model_olahdata.php* merupakan *file* yang mengatur pengolahan data yang ada.

```

1 <?php
2 class Model_olahdata extends CI_Model{
3     function __construct(){
4         parent::__construct();
5     }
6     function hitungjmlrow($datahitung)
7     {
8         return $this->db->count_all($datahitung);
9     }
10
11     function tampildatasemua($limit, $offset){
12         $this->db->from('data_individu');
13         $this->db->limit($limit, $offset);
14         $query = $this->db->get();
15         return $query->result();
16     }
17
18     function tampildataindividu($limit, $offset, $cari,$faktorcari){
19         $this->db->from('data_individu');
20         $this->db->join('data_provinsi','data_provinsi.kode_provinsi = data_individu.tempat_lahir_provinsi','left');
21         $this->db->join('data_kabupatenkota','data_kabupatenkota.kode_kabupatenkota = data_individu.tempat_lahir_kabkot','left');
22         if($faktorcari == 'Semua' or $faktorcari == ''){
23             $this->db->like('NIK',$cari);
24             $this->db->or_like('nama',$cari);
25             $this->db->or_like('no_paspor',$cari);
26             $this->db->or_like('tgl_akhir_paspor',$cari);
27             $this->db->or_like('jenis_kelamin',$cari);
28             $this->db->or_like('tempat_lahir_provinsi',$cari);
29             $this->db->or_like('tempat_lahir_kabkot',$cari);
30             $this->db->or_like('tgl_lahir',$cari);
31             $this->db->or_like('no_akta_lahir',$cari);
32             $this->db->or_like('gol_darah',$cari);
33             $this->db->or_like('agama',$cari);
34         }

```

Gambar 5. Tampilan *source code* *model_olahdata.php* dalam *dreamweaver*

2. Membuat *controller* *tambahdata.php*
Controller *tambahdata.php* merupakan *file* yang mengatur penambahan data, meliputi data individu maupun data keluarga.

```

1 <?php
2 class Tambahdata extends CI_Controller {
3     function __construct(){
4         parent::__construct();
5         $this->load->model('model_olahdata');
6         $this->load->helper('file');
7     }
8
9     function ceksesi(){
10        if ($this->session->userdata('login') != TRUE){
11            redirect('login');
12        }
13    }
14
15    function dataindividu(){
16        $this->ceksesi();
17        $data['title'] = 'Tambah Data';
18        $data['submenu'] = 'konfigurasi/submenu/tambahdata';
19        $data['isiutama'] = 'isi_utama/tambahdataindividu';
20        $data['javascript'] = 'javascript/tambahdataindividu';
21        $data['jstambahan'] = 'konfigurasi/js/tambahdataindividu';
22        $data['kodeprovinsi'] = $this->model_olahdata->cekkodeprovinsi();
23        $this->load->view('view_utama', $data);
24    }
25
26    function datakeluarga(){
27        $this->ceksesi();
28        $data['title'] = 'Tambah Data';
29        $data['submenu'] = 'konfigurasi/submenu/tambahdata';
30        $data['isiutama'] = 'isi_utama/tambahdatakeluarga';
31        $data['javascript'] = 'javascript/tambahdatakeluarga';
32        $data['jstambahan'] = 'konfigurasi/js/tambahdatakeluarga';
33        $data['kodeprovinsi'] = $this->model_olahdata->cekkodeprovinsi();
34        $this->load->view('view_utama', $data);

```

Gambar 6. Tampilan *source code* *tambahdata.php* di *dreamweaver*

3. Membuat *view* *tampilanutama.php*

View tampilanutama.php merupakan file yang bertindak untuk tampilan dasar dari setiap tampilan yang ada.

```

1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
2 <html xmlns="http://www.w3.org/1999/xhtml">
3 <head>
4 <meta http-equiv="content-type" content="text/html; charset=utf-8" />
5 <title><?php echo $title; ?></title>
6
7 <?php $this->load->view('konfigurasi/css'); ?>
8 <?php $this->load->view('konfigurasi/js/js_default'); ?>
9 <?php $this->load->view($jstambahan); ?>
10
11
12 <script type="text/javascript">
13 <?php $this->load->view($javascript); ?>
14 </script>
15 </head>
16 <body>
17 <!-- Header -->
18 <div id="header">
19 <!-- Header. Status part -->
20 <div id="header-status">
21 <div class="container_12">
22 <div class="grid_8">
23 &nbsp;
24 </div>
25 <div class="grid_4">
26 <?php $this->load->view('login/user_login'); ?>
27 </div>
28 </div>
29 <div style="clear:both;"></div>
30 </div> <!-- End $header-status -->
31
32 <!-- Header. Main part -->
33 <div id="header-main">
34 <div class="container_12">

```

Gambar 7. Tampilan source code view_utama.php di dreamweaver

4.2. Pengujian

Proses pengujian menggunakan metode *black-box* dibagi menurut fungsi dari masing-masing menu sesuai dengan kegunaannya. Bentuk pengujian berupa cara pengguna menggunakan aplikasi.

Tabel 4. Tabel pengujian seluruh fungsi

No.	Fungsi	Deskripsi	Hasil Uji
1.	Login	Merupakan proses validasi hak akses dari pengguna yang ingin menggunakan fasilitas sistem.	Berhasil
2.	Melihat semua data pengguna	Merupakan proses untuk melihat semua akun pada data pengguna	Berhasil
3.	Menambah data pengguna	Merupakan proses menambah akun data pengguna.	Berhasil
4.	Mengubah data pengguna	Merupakan proses mengubah akun data pengguna	Berhasil
5.	Menghapus data pengguna	Merupakan proses penhapusan salah satu akun pada data pengguna	Berhasil
6.	Menambah data wilayah	Merupakan proses tambah data wilayah	Berhasil
7.	Pengubahan	Merupakan proses	Berhasil

	data wilayah	pengubahan atau pembaruan salah satu data wilayah	
8	Penghapusan data wilayah	Merupakan proses penghapusan salah satu data wilayah yang ada	Berhasil
9.	Menambah data individu	Merupakan proses menambah data individu baru	Berhasil
10.	Menambah data keluarga	Merupakan proses menambah data keluarga baru	Berhasil
11	Melihat semua data individu	Merupakan proses untuk melihat semua data individu dengan bentuk detail atau tabel	Berhasil
12	Pencarian data individu	Merupakan proses untuk melakukan pencarian data individu yang diinginkan	Berhasil
13	Melihat semua data keluarga	Merupakan proses untuk melihat semua data keluarga dengan bentuk detail atau table	Berhasil
14	Pencarian data keluarga	Merupakan proses untuk melakukan pencarian data keluarga yang diinginkan	Berhasil
15	Melihat detail data individu	Merupakan proses untuk melihat salah satu data individu secara lengkap	Berhasil
16	Mencetak data detail individu	Merupakan proses untuk mencetak detail dari data individu menjadi PDF	Berhasil
17	Melihat detail data keluarga	Merupakan proses untuk melihat salah satu data keluarga secara lengkap	Berhasil
18	Mencetak data detail keluarga	Merupakan proses untuk mencetak detail dari data keluarga menjadi PDF	Berhasil
19	Mengubah data individu	Merupakan proses pengubahan salah satu data individu	Berhasil
20	Mengubah	Merupakan proses	Berhasil

	data keluarga	pengubahan salah satu data keluarga	
--	---------------	-------------------------------------	--

KESIMPULAN

Dari hasil pengujian dan analisis aplikasi Sistem Informasi Kependudukan Kota Semarang dengan Framework Codeigniter, maka dapat disimpulkan hal-hal sebagai berikut.

1. Aplikasi dibangun dengan basis web yang dapat dijalankan oleh pengguna dimana saja selama masih dalam satu jaringan.
2. Berdasarkan sampel yang ada dapat menampilkan semua data dengan baik dan detail.
3. Aplikasi ini dapat menghasilkan statistik data sesuai dengan data yang tersimpan dalam basis data.
4. Aplikasi ini walaupun berjalan pada tingkat kota Semarang namun juga dapat digunakan sampai wilayah provinsi

Saran

Berdasarkan pengujian terhadap aplikasi Sistem Informasi Demografi Penduduk Kota Semarang dengan Framework Codeigniter yang telah dibuat, dapat diberikan beberapa saran sebagai berikut.

1. Aplikasi ini dapat dikembangkan lebih lanjut dengan menambahkan beberapa fitur mencatat *history* seluruh kegiatan pengguna aplikasi yang belum dimasukkan ke dalam aplikasi.
2. Melakukan *backup* data secara berkala untuk menanggulangi jika terjadi kehilangan maupun kerusakan data.

Daftar Pustaka

- [1] Whitekom, Mark dan Mark, Bill, Seluk Beluk Database Relasional, Erlangga, Jakarta, 2003.
- [2] Madcoms. Aplikasi Program PHP dan MySQL, Penerbit Andi, Yogyakarta, 2004.
- [3] M, David, Dasar-Dasar Desain Dan Implementasi Database Processing, Erlangga, Jakarta, 2005.
- [4] Ladjamudin, Al Bahra Bin, Analisis dan Desain Sistem Informasi, Penerbit Graha Ilmu, Yogyakarta, 2007.
- [5] Mata, A.Ramon, dkk, Dasar-Dasar Database Relasional, Schaum's Outline Erlangga, Jakarta, 2007.
- [6] Hakim, Lukmanul, Membakar trik Rahasia Para Master PHP, Loko media, Yogyakarta, 2008.
- [7] Kadir, Abdul, Dasar Perancangan dan Implementasi Database Relasional, Andi Offset, Yogyakarta, 2008.

- [8] Kadir, Abdul. Dasar Pemrograman Web Dinamis Menggunakan PHP. Penerbit Andi, Yogyakarta, 2008.
- [9] Kristanto, Andri, Perancangan Sistem Informasi dan Aplikasinya, Gaya Media, Yogyakarta, 2008.
- [10] Nugroho, Bunafit. Latihan Membuat Aplikasi Web PHP dan MySQL dengan Dreamweaver, Gaya Media, Yogyakarta, 2008.
- [11] Hakim, Lukmanul, *Jalan pintas menjadi master PHP*, Loko media, Yogyakarta, 2009.
- [12] Hermawan, C.Widya, *ShortCourse : PHP Programming*, Wahana Komputer, Semarang, 2009.
- [13] Kadir, Abdul, *Form Zero to A Pro Membuat aplikasi Web dengan PHP dan Database MySQL*, Penerbit Andi, Yogyakarta, 2009.
- [14] Kadir, Abdul, *Mastering AJAX dan PHP*, Andi Offset, Yogyakarta, 2009.
- [15] Kadir, Abdul, *Mudah Mempelajari Database MySQL*, Penerbit Andi, Yogyakarta, 2009.

BIODATA PENULIS



R. Kibar Aji Seno Lahir di Semarang pada 20 April 1990 Telah menempuh pendidikan di TK *Setyo Siwi*, SDN Sompok Semarang, SMPN 02 Semarang, SMAN 03 Semarang, dan sekarang tengah menempuh pendidikan Strata satu di Program Studi Sistem Komputer

Fakultas Teknik Universitas Diponegoro, Semarang, Indonesia Angkatan Tahun 2009.

Menyetujui,
Dosen Pembimbing I

Ir. Kodrat Iman Satoto, M.T.
NIP. 196310281993031002

Dosen Pembimbing II

Rinta Kridalukmana, S.Kom., M.T.
NIP. 197706152008011011