# SKRIPSI

**SISTEM INFORMASI PENGOLAHAN DATA PERTUMBUHAN EKONOMI DAN KETIMPANGAN ANTAR KECAMATAN DI KABUPATEN KLATEN (TAHUN 2003-2012) MENGGUNAKAN FRAMEWORK CODEIGNITER**

**DATA PROCESSING INFORMATION OF ECONOMY GROWTH AND LAMENESS INTER DISTRICT AT KLATEN REGENCY (PERIOD 2003-2012) USING CODEIGNITER FRAMEWORK**

****

**Disusun Oleh :**

**Koespradono**

**091051092**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INDUSTRI**

**INSTITUT SAINS & TEKNOLOGI AKPRIND**

**YOGYAKARTA**

**2014**

# SKRIPSI

**SISTEM INFORMASI PENGOLAHAN DATA PERTUMBUHAN EKONOMI DAN KETIMPANGAN ANTAR KECAMATAN DI KABUPATEN KLATEN (TAHUN 2003-2012) MENGGUNAKAN FRAMEWORK CODEIGNITER**

**DATA PROCESSING INFORMATION OF ECONOMY GROWTH AND LAMENESS INTER DISTRICT AT KLATEN REGENCY (PERIOD 2003-2012) USING CODEIGNITER FRAMEWORK**

****

**Disusun Oleh :**

**Koespradono**

**091051092**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INDUSTRI**

**INSTITUT SAINS & TEKNOLOGI AKPRIND**

**YOGYAKARTA**

**2014**

**SISTEM INFORMASI PENGOLAHAN DATA PERTUMBUHAN EKONOMI DAN KETIMPANGAN ANTAR KECAMATAN DI KABUPATEN KLATEN (TAHUN 2003-2012) MENGGUNAKAN FRAMEWORK CODEIGNITER**

**DATA PROCESSING INFORMATION OF ECONOMY GROWTH AND LAMENESS INTER DISTRICT AT KLATEN REGENCY (PERIOD 2003-2012) USING CODEIGNITER FRAMEWORK**

Laporan Penelitian Skripsi ini

Disusun sebagai Salah Satu Syarat Kelengkapan Kurikulum

pada Jurusan Teknik Informatika Fakultas Teknologi Industri

Institut Sains & Teknologi AKPRIND Yogyakarta

**A description...**

**Disusun Oleh:**

**Koespradono**

**091051092**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INDUSTRI**

**INSTITUT SAINS & TEKNOLOGI AKPRIND**

**YOGYAKARTA**

**2014**

# HALAMAN PERSETUJUAN

**SISTEM INFORMASI PENGOLAHAN DATA PERTUMBUHAN EKONOMI DAN KETIMPANGAN ANTAR KECAMATAN DI KABUPATEN KLATEN (TAHUN 2003-2012) MENGGUNAKAN FRAMEWORK CODEIGNITER**

**DATA PROCESSING INFORMATION OF ECONOMY GROWTH AND LAMENESS INTER DISTRICT AT KLATEN REGENCY (PERIOD 2003-2012) USING CODEIGNITER FRAMEWORK**

Disusun Oleh:

Koespradono

091051092

Skripsi Mahasiswa tersebut

Dinyatakan telah Memenuhi Syarat

untuk Diujikan dalam Ujian Pendadaran

Telah Disetujui:

Di : Yogyakarta

Tanggal :

Jurusan Teknik Informatika

|  |  |
| --- | --- |
| Dosen Pembimbing I,  Catur Iswahyudi, S.Kom., S.E., M.Cs.  NIK 03.0775.579.E | Dosen Pembimbing II,  Muhammad Sholeh, S.T., M.T.  NIK 94.1269.498.E |

Mengetahui,

Ketua Jurusan

Uning Lestari, S.T., M.Kom.

NIK 96.0870.520.E

**HALAMAN PENGESAHAN**

**SISTEM INFORMASI PENGOLAHAN DATA PERTUMBUHAN EKONOMI DAN KETIMPANGAN ANTAR KECAMATAN DI KABUPATEN KLATEN (TAHUN 2003-2012) MENGGUNAKAN FRAMEWORK CODEIGNITER**

**DATA PROCESSING INFORMATION OF ECONOMY GROWTH AND LAMENESS INTER DISTRICT AT KLATEN REGENCY (PERIOD 2003-2012) USING CODEIGNITER FRAMEWORK**

Telah Diujikan dan Dipertahankan

dalam Sidang Ujian Pendadaran Skripsi

Jurusan Teknik Informatika Fakultas Teknologi Industri

Institut Sains & Teknologi AKPRIND Yogyakarta

Pada:

Hari :

Tanggal :

Disetujui:

Di :

Tanggal :

|  |  |
| --- | --- |
| Dosen Penguji | Tanda Tangan |
| 1. Catur Iswahyudi, S.Kom., S.E., M.Cs. | 1. |
| 1. Muhammad Sholeh, S.T., M.T. | 2. |
|  | 3. |

Mengetahui,

Kepala Jurusan Teknik Informatika

Uning Lestari, S.T., M.Kom.

NIK 96.0870.520.E

**SURAT PERNYATAAN**

Yang bertanda tangan dibawah ini :

Nama : Koespradono

NIM : 091051092

Jurusan : Teknik Informatika

Fakultas : Teknologi Industri

Perguruan Tinggi : Institut Sains & Teknologi AKPRIND Yogyakarta

Menyatakan bahwa Skripsi sebagai berikut:

Judul Bahasa Indonesia : Sistem Informasi Pengolahan Pertumbuhan Ekonomi Dan Ketimpangan Antar Kecamatan Di Kabupaten Klaten (Tahun 2003-2012) Menggunkan Framework Codeigniter

Judul Bahasa Inggris : Data Processing Information System Of Economy Growth And Lameness Inter District At Klaten Regency (Period 2003-2012) Using CodeIgniter Framework

Dosen Pembimbing I : Catur Iswahyudi, S.Kom., S.E., M.Cs.

Dosen Pembimbing II : Mohammad Sholeh, S.T., M.T.

Adalah benar-benar **ASLI** dan **BELUM PERNAH** dibuat oleh orang lain, kecuali yang diacukan dalam daftar pustaka **SKRIPSI** ini.

Demikian pernyataan ini **SAYA** buat, apabila di kemudian hari terbukti bahwa **SAYA** menjiplak karya orang lain, **SAYA** bersedia menerima **SANKSI AKADEMIK**.

Yogyakarta, 24 Maret 2013

Koespradono

**HALAMAN PERSEMBAHAN**

**Kupersembahkan Untuk :**

Allah SWT Tuhan Semesta Alam dan Muhammad SAW Sang Rasul atas semua berkah dan bimbingan hidup yang diberikan,

Kedua Orang Tuaku, sahabat-sahabatku, saudariku-saudariku dan diriku,

Semua Makhluk ALLAH SWT, Teman-temanku dan Almamaterku IST AKPRIND Yogyakarta.

# MOTTO

”Allah tidak akan merubah nasib suatu kaum hingga ia berusaha untuk mengubahnya sendiri”

**Qur’an 13:11**

”Apabila seorang keturunan Adam meninggal dunia maka terputuslah amalnya kecuali dari tiga hal: shadaqah jariyyah, atau ilmu yang bermanfaat, atau seorang anak shalih yang mendo'akannya”

**HR. Muslim**

”Learn from yesterday, live for today, hope for tomorrow. The important thing is not to stop questioning”

**Albert Einstein**

”Waktu akan terasa lambat bagi mereka yg menunggu, terlalu panjang bagi mereka yg gelisah, dan terlalu singkat bagi mereka yg bahagia, tetapi waktu akan terasa abadi bagi mereka yg mampu bersyukur”

**Anonim**

# KATA PENGANTAR

*Assalamu’alaikum Wr, Wb.*

Puji syukur penulis panjatkan kehadirat Allah SWT, atas segala Rahmat dan Anugerah-Nya sehingga penulis dapat menyelesaikan penyusunan laporan Skripsi dengan judul **“SISTEM INFORMASI PENGOLAHAN DATA PERTUMBUHAN EKONOMI DAN KETIMPANGAN ANTAR KECAMATAN DI KABUPATEN KLATEN (TAHUN 2003-2012) MENGGUNAKAN FRAMEWORK CODEIGNITER”**. Skripsi ini disusun guna memenuhi syarat kelengkapan kurikulum bagi mahasiswa Jurusan Teknik Informatika, Fakultas Teknologi Industri di Institut Sains & Teknologi AKPRIND Yogyakarta dalam menyelesaikan program studi Strata (S-1).

Penyusunan laporan skripsi ini tidak akan berhasil dengan baik tanpa adanya bantuan dari berbagai pihak. Dalam penyusunan laporan ini penulis mengambil data-data dari berbagai sumber, baik dengan melakukan percobaan-percobaan yang mungkin bisa dilakukan, maupun studi literatur dengan membaca dan mempelajari berbagai pustaka yang ada yang berkaitan dengan masalah yang diteliti.

Pada kesempatan ini penulis menghaturkan banyak terima kasih kepada :

1. Bapak Ir Sudarsono selaku Rektor IST AKPRIND Yogyakarta.
2. Bapak Muhammad Sholeh, ST., MT. selaku Dekan Fakultas Teknologi Industri.
3. Ibu Uning Lestari, ST., M.Kom. selaku ketua jurusan Teknik Informatika.
4. Bapak Catur Iswahyudi, S.Kom, SE. selaku Dosen Pembimbing I Skripsi.
5. Bapak Muhammad Sholeh, ST.,MT. selaku Dosen Pembimbing II Skripsi.
6. Kedua Orang Tuaku tercinta, sebagai motivator terbesar dalam hidupku atas semua pengorbanan dan kesabaran mengantarku sampai saat ini.
7. Saudara ku Sidiq Purnomo dan Muhammad Ikhsan Triyanto beserta keluarga yang kucintai.
8. Eriza, Tomi, Enggar, Riyan, Syaiful, Salman, Alfan, Wasis, Jehan, Yoga, Vian, Aji, Dimas dan teman-teman seperjuangan ”*Cyber Pengok”*. *for u all I miss u forever.*
9. Semua teman-teman angkatan 2009 Teknik Informatika yang tidak mungkin penulis sebutkan satu-persatu, atas dukungan dan kesempatan berbagi ilmu.

Penulis menyadari dalam penyusunan Skripsi ini masih terdapat banyak kekurangan, untuk itu penulis mengharapkan kritik dan saran yang sifatnya membangun demi kesempurnaan Skripsi ini. Akhir kata semoga Skripsi ini dapat bermanfaat bagi pembaca dan khususnya dapat berguna dalam kemajuan bidang Informatika.

*Wassalamu’alaikum, Wr, Wb.*

Yogyakarta, 24 Maret 2013

Penyusun,

Koespradono

# DAFTAR ISI

**HALAMAN SAMPUL ii**

**HALAMAN JUDUL. iii**

**HALAMAN PERSETUJUAN. iv**

**HALAMAN PENGESAHAN. v**

**SURAT PERNYATAAN. vi**

**HALAMAN PERSEMBAHAN. vii**

**MOTTO. viii**

**KATA PENGANTAR. ix**

**DAFTAR ISI. xi**

**DAFTAR GAMBAR. xv**

**DAFTAR TABEL. xvii**

**INTISARI xviii**

**ABSTRACT. xix**

**BAB I. PENDAHULUAN**

* 1. Latar Belakang Permasalahan. 1
  2. Rumusan Masalah. 5
  3. Batasan Masalah. 6
  4. Tujuan Penelitian. 6
  5. Manfaat Penelitian. 7

**BAB II. LANDASAN TEORI**

* 1. Tinjauan Pustaka. 8
  2. Landasan Teori 10

1. Konsep Sistem Informasi. 10
2. Geograpic Information System (GIS). 11
3. Peta. 14
4. Shape File. 15
5. Konsep Bsis Data 16
6. PostgreSQL 20
7. *Internet* 22
8. Server 23
9. WWW (World Wide *Web*) 23
10. *Web* Browser. 25
11. UML (Unified Modeling Language). 27

**BAB III. METODOLOGI PENELITIAN**

* 1. Obyek Penelitian 25
  2. Kebutuhan Sistem 25
     1. Hardware 25
     2. Software dan Aplikasi 25
  3. Metode Pengumpulan Data 26
  4. Metode Analisis Sistem 26
  5. Langkah-langkah Penelitian 27
  6. Proses Simulasi 27
  7. Proses Pengujian 28
  8. Skenario Pengujian Serangan 28
     1. Identify 25
     2. Understand 25
     3. Analyze 26
     4. Report 26
        1. Data Flow Diagram (DFD) Level 2 30
        2. Data Flow Diagram (DFD) Level 3 30
  9. Rancangan Basis Data 31
     1. Struktur Tabel 32
     2. Relasi Antar Tabel 37
  10. Rancangan Menu Sistem 38
  11. Rancangan Antar Muka Sistem 39

**BAB IV. HASIL DAN PEMBAHASAN**

1. Desain Tampilan Untuk Admin Pengelola Kecamatan 41
2. Tampilan Login Admin Pengelola Kecamatan 41
3. Tampilan Halaman Utama Admin pengelola Kecamatan 42
4. Tampilan Album 43
5. Tampilan Tambah Album 44
6. Tampilan Edit Album 45
7. Tampilan Gallery Foto 46
8. Tampilan Tambah Gallery Foto 47
9. Tampilan Edit Gallery Foto 48
10. Tampilan Industri 49
11. Tampilan Tambah Industri 50
12. Tampilan Edit Industri 51
13. Desain Tampilan Untuk Admin Pusat 52
    1. Tampilan Login Admin Pusat 52
    2. Tampilan Halaman Admin Pusat 53
    3. Tampilan Halaman Statis (About) 54
    4. Tampilan Hubungi Kami 55
    5. Tampilan Konfirmasi Hubungi Kami 56
    6. Tampilan Kategori 57
    7. Tampilan Tambah Kategori 58
    8. Tampilan Edit Kategori 59
    9. Tampilan Kecamatan 60
    10. Tampilan Management Setting 61
    11. Tampilan Management User 62
    12. Tampilan Tambah User 63
14. Desain Tampilan Halaman *Web*site 63
    * 1. Tampilan Halaman Utama (Home Page) 63
      2. Tampilan Menu Kecamatan Kabupaten Bantul 66
      3. Tampilan Menu Tentang Kami 67
      4. Tampilan Menu Hubungi Kami 68
      5. Tampilan Pencarian Lokasi Industri Terdekat 69
      6. Tampilan Lokasi Industri Kecamatan 71
      7. Tampilan Detail Lokasi Industri Kecamatan 72
15. Kelebihan dan Kekurangan Sistem 74
    * + 1. Kelebihan 74
        2. Kekurangan 74

**BAB V. PENUTUP**

* 1. Kesimpulan 75
  2. Saran 75

**DAFTAR PUSTAKA**

**LAMPIRAN**

# DAFTAR GAMBAR

[Gambar II.1 Tampilan *Google Maps* 13](#_Toc378918800)

[Gambar II.2 Konsep MVC Framework 36](#_Toc378918801)

[Gambar II.3 Tampilan default Framework Codeigniter 37](#_Toc378918802)

[Gambar III.1 Diagram Alir Penelitian **Error! Bookmark not defined.**](#_Toc378921140)

[Gambar III.2 Use Case Diagram Admin 48](#_Toc378921141)

[Gambar III.3 Use Case Diagram User 49](#_Toc378921142)

[Gambar III.4 Class Diagram 50](#_Toc378921143)

[Gambar III. 5 Sequence Diagram Admin 51](#_Toc378921144)

[Gambar III.6 Sequence Diagram User 52](#_Toc378921145)

[Gambar III. 7 Activity Diagram Admin 53](#_Toc378921146)

[Gambar III. 8 Activity Diagram User 54](#_Toc378921147)

[Gambar III.9 Relasi Antar Tabel 61](#_Toc378921148)

[Gambar III. 10 Tampilan Rancangan Halaman Beranda 62](#_Toc378921149)

[Gambar III. 11 Tampilan Rancangan Halaman About **Error! Bookmark not defined.**](#_Toc378921150)

# DAFTAR TABEL

[Tabel III.1 level user 55](#_Toc379174174)

[Tabel III. 2 user 55](#_Toc379174175)

[Tabel III.3 kategori 56](#_Toc379174176)

[Tabel III.4 berita 57](#_Toc379174177)

[Tabel III.5 tautan 58](#_Toc379174178)

[Tabel III.6 setting\_*web* 58](#_Toc379174179)

[Tabel III.7 kecamatan 59](#_Toc379174180)

[Tabel III.8 data 60](#_Toc379174181)

[Tabel III. 9 sector 60](#_Toc379174182)

# INTISARI

Sektor industri yang tersebar di Kabupaten Bantul sangat beragam. Secara umum industri yang terdapat di Kabupaten Bantul merupakan industri kecil, sedangkan untuk industri besar jumlahnya tidak banyak. Namun potensi industri ini belum banyak diketahui oleh masyarakat luas. Salah satu kendalanya adalah kurangnya informasi mengenai lokasi industri. Maka dari itu sangat diperlukan *web*site industri kabupaten Bantul berbasis Geographic Information System (GIS) yang akan membantu dan mempermudah baik dari sisi pengguna dimanapun berada agar dapat memperoleh informasi dan pemerintah daerah Kabupaten Bantul dari sisi promosi daerah.

GIS dapat menjadi salah satu alat bantu utama yang interaktif, karena dengan adanya GIS akan digambarkan juga letak lokasi industri pada kondisi sesungguhnya dalam bentuk peta. Proses tersebut perlu adanya pembuatan *web*site agar dapat lebih menarik dan tersampaikan informasi secara cepat, akurat dan efisien.

Dengan adanya sistem informasi industri Kabupaten Bantul dapat menampilkan informasi mengenai industri tingkat Kabupaten Bantul berupa galeri foto, deskripsi industri, kontak komunikasi pihak industri serta rute dan jarak terdekat lokasi industri.

# ABSTRACT

The industrial sector is scattered in Bantul district is very diverse. In general, the industry is located in Bantul district is a small industry, while for large industries is small in number. But the potential of this industry has not been widely known by the public. One of the obstacles is lack of information about the location of industry. Therefore, it is necessary to build Bantul industry *web*site based on Geographic Information System (GIS) that would assist and facilitate both the users wherever they are in order to obtain information and the local government of Bantul Regency of the promotion of the region.

GIS can be one of the main tools are interactive, because with the GIS will be described also the location where the actual conditions in the industry in the form of a map. The process of making a *web*site need to be more interesting and conveyed information quickly, accurately and efficiently.

The information system of industry Bantul can display information about the industry level in the form of photo galleries Bantul, industry descriptions, contact to the communications industry as well as the shortest distance route and the location of industries

# BAB I

# PENDAHULUAN

## Latar Belakang

Perkembangan ilmu pengetahuan dan teknologi khususnya bidang komputer saat ini menunjukkan grafik perkembangan yang sangat pesat. Perkembangan ini telah mempengaruhi banyak bidang kehidupan termasuk bidang informasi. Penggunaan komputer yang semakin canggih menumbuhkan daya kreasi baru bagi para *user*. Salah satu kreasi yang berkembang terus-menerus adalah teknologi informasi. Teknologi informasi telah menduduki peringkat teratas sebagai kebutuhan dasar manusia sejak berabad-abad lamanya. Kini teknologi informasi itu mendapat bentuk baru, yakni sistem informasi yang dapat diakses tanpa batas ruang dan waktu. Sistem informasi mutakhir yang menjadi kegemaran adalah sistem informasi *online*. Sistem informasi *online* dikemas dalam berbagai bentuk seperti *website*, forum, *survey online* dan lain-lain.

*Website* merupakan salah satu media yang dapat digunakan untuk memberikan informasi kepada masyarakat umum secara cepat dan mudah melalui

*internet*. Biaya yang murah, kemudahan akses dan efisiensi menjadi alasan semakin luasnya penggunaan *website*. Kebutuhan akan sebuah informasi juga menjadi salah satu faktor pendorong perkembangan *website*.

Dalam pengembangan sistem informasi pun khususnya berbasis *website* saat ini sudah menggunakan konsep MVC (*Model View Controller*) yang membuat para *developer*  (pembangun aplikasi) menjadi mudah dan penulisan *source code* yang menjadi lebih rapi sehingga mudah untuk kegiatan maintenance dan pengembangan. Konsep tersebut dikembangkan atas dasar OOP (*Object Oriented Programming*) atau Pemrograman Berorientasi Objek dan kemudian dikembangkan lagi dalam bentuk MVC *Framework*, Salah satu MVC *Framework* yang populer digunakan saat ini adalah *Framework CodeIgniter* yang dikembangkan oleh *Eliss Lab*.

Upaya pemerintah pusat untuk meningkatkan peran pemerintah daerah dalam pembangunan daerah sendiri, yang didukung oleh adanya undang-undang Pemerintah Daerah (Lembaga Negara Republik Indonesia Tahun 2004 No 125, Tambahan Lembaran Negara Republik Indonesia Nomor 4437) dan UU RI No.33 tahun 2004 tentang perimbangan keuangan antara pemerintah pusat dan pemerintah daerah, mendorong pemerintah daerah untuk mempercepat proses pembangunan dengan suatu model percepatan pembangunan yaitu Daerah Otonom yang disertai dengan kemandirian pengelolaan daerahnya.

Daerah Otonom adalah kesatuan masyarakat hukum dengan batas tertentu berwenang mengatur dan mengurus kepentingan masyarakat setempat menurut prakarsa sendiri berdasarkan aspirasi masyarakat. Tolak ukur dari keberhasilan pembangunan dapat dilihat dari pertumbuhan ekonomi, struktur ekonomi, dan semakin minimnya antar penduduk, antar daerah, dan antar sector.

Agar pembangunan ekonomi di daerah dapat selaras dan seimbang dengan pembangunan diskala nasional, maka perlu dilakukan berbagai kegiatan pembangunan sektoral yang harus disesuaikan dengan kondisi, prioritas, dan potensi yang dimiliki tiap daerah.

Beberapa daerah mengalami pertumbuhan yang sangat cepat namun beberapa daerah sangat lambat, disinilah peran serta pemerintah daerah untuk mengembangkan daerah yang tertinggal ini untuk maju. Pemerintah daerah memiliki beberapa permasalahan dalam penyelenggaraan proses pembangunan daerahnya, baik itu berupa pengambilan kebijakan atau mengusulkan strategi pembangunan daerahnya, memperbaiki lingkungan didaerahnya agar mampu mendukung perkembangan perekonomian.

Adanya pengembangan wilayah pada pusat-pusat pertumbuhan akan merangsang pertumbuhan ekonomi yang juga akan diikuti oleh pembangunan wilayah. Kondisi ini akan menciptakan hubungan timbal balik yang akan menjadi generator bagi pertumbuhan perekonomian daerah. Tujuan dari pembangunan adalah naiknya pertmbuhan ekonomi melalui peningkatan Produk Domestik Bruto (PDRB) dengan adanya peningkatan PDRB berarti meningkat juga kemakmuran dan kesejahteraan masyarakat. Salah satu faktor penting yang dapat menentukan keberhasilan proses pembangunan di Indonesia adalah faktor perbedaan tingkat pertumbuhan ekonomi dan pendapatan perkapita regional. Pertumbuhan ekonomi yang dilihat berdasarkan nilai Produk Domestik Regional Bruto (PDRB) dari tahun ke tahun merupakan salah satu indikator dari keberhasilan pembangunan.

Dengan adanya peran Pemerintah Daerah untuk menjaga stabilitas perekonomian, serta mengembangankan dan mempercepat perekonomian daerah yang ada, membuat pemerintah daerah harus dapat melihat dan menentukan wilayah-wilayah mana yang secara ekonomi, sosial, dan kultural memiliki potensi untuk dikembangkan, baik itu potensi yang ada secara alami maupun potensi yang telah tumbuh dikarenakan adanya pembangunan. Hal ini penting bagi pemerintah agar para pengambil kebijakan tersebut, dapat lebih menempatkan pembangunan infrastruktur dan fasilitas-fasilitas lainnya pada lokasi tepat yang akan memberikan dampak-dampak yang positif terhadap perkembangan dan pertumbuhan pembangunan ekonomi.

Ciri-ciri dari daerah yang perekonomiannya meningkat secara terus menerus adalah memiliki industri yang kuat dan maju, pertanian yang tangguh serta memiliki basis-basis pertumbuhan sektoral yang berpotensi besar. Selain itu, pertumbuhan juga diperlukan untuk menggerakan dan memacu pembangunan dibidang lainnya yang akan meningkatkan pendapatan masyarakat. Pertumbuhan yang ada jika diarahkan pada daerah-daerah yang telah memiliki potensi atau fasilitas wilayah juga akan mempercepat terjadinya kemajuan ekonomi, karena secara tidak langsung kemajuan daerah akan membuat masyarakat untuk mencari kehidupan yang lebih baik di daerah tersebut.

Sistem ini dikembangkan untuk mempermudah pemerintah dalam bidang bidang statistik khususnya pertumbuhan ekonomi dan mengetahui besar tingkat ketimpangan di tiap-tiap kecamatan yang berada di Kabupaten Klaten. Dalam bentuk statistik dan dapat mempermudah pencarian daerah mana saja yang mengalami pertumbuhan ekonominya tinggi atau rendah di setiap wilayah Kabupaten Klaten, dapat mengakses sistem ini menggunakan media apa saja yang bisa terhubung dengan *internet*

## Rumusan Masalah

Berdasarkan latar belakang masalah di atas maka dapat diambil rumusan masalah sebagai berikut:

1. Bagaimana menyediakan sistem informasi pengolahan data dalam bentuk statistik yang memudahkan bagi pemerintah dalam menganalisis pertumbuhan ekonomi di Kabupaten Klaten?
2. Berapa besar tingkat ketimpangan antar kecamatan di kabupaten Klaten pada periode 2003-2012?

## **Batasan Masalah**

Batasan masalah dibuat agari sistem ini lebih terfokus sesuai dengan tugas dan fungsinya secara maksimal adalah sebagai berikut:

1. Data yang di ambil meliputi PDRB, jumlah penduduk, dan mata pencaharian kabupaten klaten per kecamatan tahun2003 – 2012.
2. Bahasa pemograman yang dipakai adalah PHP dengan Framework CodeIgniter dan menggunakan basis data MySQL.
3. Penggunaan Goggle Maps sebagai penunjuk arah (route) dan pencarian

lokasi terdekat (near me).

## Tujuan Penelitian

Tujuan dari dibuatnya sistem informasi ini adalah sebagai berikut:

1. Menghasilkan Sistem Informasi Pengolahan Data Pertumbuhan Ekonomi dan Ketimpangan Antar Kecamatan di Kabupaten Klaten yang dapat digunakan oleh pihak lembaga pemerintah pengkajian dan penyusunan kebijakan nasional dibidang kegiatan statistik.
2. Untuk menganalisis besarnya ketimpangan antar kecamatan di Kabupaten Klaten.

## Manfaat Penelitian

1. Referensi penulisan karya ilmiah dalam bentuk laporan Tugas Akhir (Skripsi) bagi mahasiswa.
2. Memudahkan Pemerintah untuk memperoleh informasi letak obyek besar tingkat ketimpangan antar kecamatan di Kabupaten Klaten.
3. Memberi gambaran penerapan Sistem Informasi Pengolahan Data Pertumbuhan Ekonomi dan Ketimpangan Antar Kecamatan Menggunakan Framework CodeIgniter, Goggle Maps dan statistik di Kabupaten Klaten

# BAB II

# TINJAUAN PUSTAKA

## Tinjauan Pustaka

Pesatnya kemajuan dan kecanggihan teknologi saat ini, tekad untuk dapat menjadi bagian dari kemajuan dan kecanggihan teknologi tersebut terus berlangsung. Dari hasil analisis, kemajuan dan kecanggihan teknologi mengakibatkan perubahan guna pembaharuan dan integrasi sistem. Berbekal ilmu yang diraih pada lingkungan akademik dan juga referensi yang tersedia di perpustakaan maupun *internet*, maka diimplementasikan sebuah sistem yang memanfatkan pembaharuan dan kemajuan teknologi tersebut.

Dalam penelitian ini digunakan beberapa referensi yang berhubungan dengan objek penelitian. Beberapa referensi itu dapat diambil dari sumber yang berhubungan dengan komputerisasi, serta dari hasil penelitian yang telah dilakukan sebelumnya, antara lain:

1. (Tanamah, 2008), melakukan penelitian mengenai perancangan dan implementasi *Web* GIS Pariwisata Kabupaten Sumba Timur, Sistem ini mampu memberikan informasi posisi daerah wisata dan fasilitas *search* untuk membantu penggunaan dalam mendapat informasi lokasi pada *Web* GIS Pariwisata Sumba Timur ini belum secara *detail*. Kelemahan *Web* GIS Pariwissata Sumba Timur ini belum bisa menampilkan lokasi pengguna saat mengakses *web* pada peta, memang dikarenakan belum adanya fitur GPS.
2. (Alhadi, 2013), Sistem Penggajian dan Pengupahan Karyawan Menggunakan *Framework Codeigniter*, Penelitian ini berhasil mengimplementasikan hasil analis dan perancangan sistem pengajian ke dalam sistem yang sudah ada, dari sistem yang sudah di buat mempunyai kelemahan belum adanya enkripsi *password* sehingga tingkat keamanan sistem masih kurang dan sistem ini perlu dikembangkan agar *user friendly*.
3. (Pratono, 2009), Pertumbuhan Ekonomi dan Ketimpangan di Kabupaten Klaten dapat dijadikan sebagai menganalisis besarnya ketimpangan untuk mengetahui tingkat pertumbuhan ekonomi perkecamatan di Kabupaten klaten. Dari penelitian ini dilihat salah satu faktor penyebab ketimpangan pada perbedaan letak geografis daerah kecamatan yang menyebabkan hanya dapat dilakukan di daerah-daerah yang mudah dijangkau. Pada referensi ini pengembang menerapkan apa yang sudah diteliti dalam bentuk statistik dan pemetaan agar mudah dalam memperoleh informasi.

Untuk melengkapi penelitian yang telah dilakukan sebelumnya, dalam penelitian ini terdapat perbedaan materi yaitu Sistem ini mampu memberikan informasi posisi daerah yang mengalami ketimpangan dengan fasilitas *near me* (pencarian lokasi terdekat) dan hasil dari pengolahan data ditampilkan dalam bentuk statistik diagram agar memudahkan dalam menganalisa pertumbuhan ekonomi dan ketimpangan.

## Landasan Teori

### Konsep sistem informasi

Sistem yaitu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul, bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu.

Suatu sistem yang baik harus mempunyai tujuan dan sasaran yang tepat karena hal ini akan sangat menentukan dalam mendefinisikan masukan yang dibutuhkan sistem dan juga keluaran yang dihasilkan .

Informasi adalah kumpulan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerima. Tanpa suatu informasi, suatu sistem tidak akan berjalan dan tidak bisa beroperasi.

Dengan kata lain informasi adalah data. Data menggambarkan suatu kejadian yang sedang terjadi, dimana data tersebut akan diolah dan diterapkan dalam sistem menjadi *input* yang berguna dalam suatu sistem .

Sistem informasi merupakan kumpulan dari perangkat keras dan perangkat lunak komputer serta perangkat manusia yang akan mengolah data menggunakan perangkat keras dan perangkat lunak tersebut. Selain itu sistem informasi dapat didefinisikan sebagai berikut:

1. Suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi.
2. Sekumpulan prosedur organisasi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan dan atau untuk mengendalikan informasi.
3. Suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan (Kristanto, 2008).

### Geograpic information system (GIS)

*Geographic Information System* (GIS) adalah suatu sistem informasi yang dirancang untuk bekerja dengan data yang bereferensi spasial atau berkoordinat geografi atau dengan kata lain suatu GIS adalah suatu sistem basis data dengan kemampuan khusus untuk menangani data yang bereferensi keruangan (spasial) bersamaan dengan seperangkat operasi kerja . Sedangkan menurut Sistem Informasi geografi adalah suatu sistem Informasi yang dapat memadukan antara data grafis (spasial) dengan data teks (atribut) objek yang dihubungkan secara geogrfis di bumi (*georeference*). Disamping itu, GIS juga dapat menggabungkan data, mengatur data dan melakukan analisis data yang akhirnya akan menghasilkan keluaran yang dapat dijadikan acuan dalam pengambilan keputusan pada masalah yang berhubungan dengan geografi.

Sistem Informasi Geografis dibagi menjadi dua kelompok yaitu sistem manual (analog), dan sistem otomatis (yang berbasis digital komputer). Perbedaan yang paling mendasar terletak pada cara pengelolaannya. Sistem Informasi manual biasanya menggabungkan beberapa data seperti peta, lembar transparansi untuk tumpang susun *(overlay)*, foto udara, laporan statistik dan laporan survey lapangan. Kesemua data tersebut dikompilasi dan dianalisis secara manual dengan alat tanpa komputer. Sedangkan Sistem Informasi Geografis otomatis telah menggunakan komputer sebagai sistem pengolah data melalui proses digitasi. Sumber data digital dapat berupa citra satelit atau foto udara digital serta foto udara yang terdigitasi. Data lain dapat berupa peta dasar terdigitasi.

Ada beberapa alasan mengapa perlu menggunakan Sistem Informasi Geografis, diantaranya adalah :

1. GIS menggunakan data spasial maupun atribut secara terintegrasi,
2. GIS dapat digunakan sebagai alat bantu interaktif yang menarik dalam usaha meningkatkan pemahaman mengenai konsep lokasi, ruang, kependudukan, dan unsur-unsur geografi yang ada dipermukaan bumi,
3. GIS dapat memisahkan antara bentuk presentasi dan basis data,
4. Semua operasi GIS dapat dilakukan secara interaktif,
5. GIS memiliki kemampuan menguraikan unsur-unsur yang ada di permukaan bumi ke dalam beberapa *layer* atau *coverage* data spasial.
6. GIS memiliki kemampuan yang sangat baik dalam memvisualisasikan data spasial berikut atributnya.
7. GIS dengan mudah menghsilkan peta-peta tematik,
8. Semua operasi GIS dapat di *costumize* dengan menggunakan perintah-perintah dalam bahasa *script*,
9. Perangkat lunak GIS menyediakan fasilitas untuk berkomunikasi dengan perangkat lunak lain,
10. GIS sangat membantu pekerjaan yang erat kaitannya dengan bidang spasial dan geoinformatika.

Komponen-komponen GIS

1. *Hardware*

Sistem informasi geografis memerlukan spesifikasi komponen *hardware* yang sedikit lebih tinggi dibanding spesifikasi komponen sistem informasi lainnya.

1. *Software*

Sebuah software GIS haruslah menyediakan fungsi dan tool yang mampu melakukan penyimpanan data, analisis dan menampilkan informasi geografis.

1. Data

Hal yang merupakan komponen penting dalam GIS adalah data. Secara *fundamental*, GIS bekerja dengan 2 tipe model data geografis, yaitu model data *vector* dan model data raster.

1. Manusia

Komponen manusia memegang peranan yang sangat menentukan, karena tanpa manusia maka sistem tersebut tidak dapat diaplikasikan dengan baik. Jadi manusia menjadi komponen yang mengendalikan suatu sistem sehingga menghasilkan suatu analisa yang dibutuhkan.

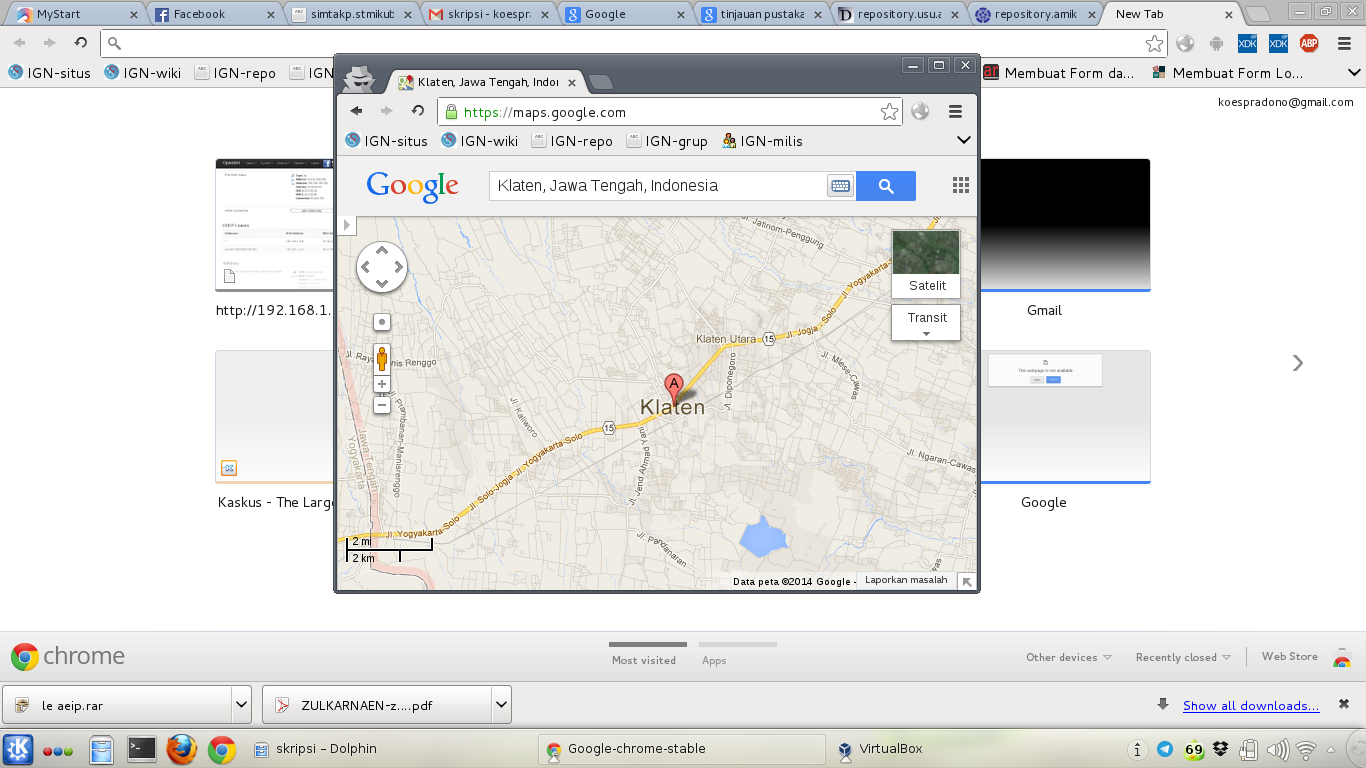
1. Metode

GIS yang baik memiliki keserasian antara rencana desain yang baik dan aturan dunia nyata, dimana metode, model dan implementasi akan berbeda untuk setiap permasalahan.

### Google maps

Google Maps adalah sebuah jasa peta *globe virtual* gratis dan *online* disediakan oleh Google dapat ditemukan di <http://maps.google.com/>. Google Map menawarkan peta yang dapat diseret dan gambar satelit untuk seluruh dunia, menampilkan citra satelit resolusi tinggi yang disediakan *DigitalGlobe* dengan satelitnya *QuickBird*. Serta data dari *Geographic Information System* (GIS) buatan Tele Atlas, *NAVTEQ* dan *MapABC* (Susrini, 2009)

Salah satu contoh tampilan Google Maps dapat dilihat pada Gambar II.1 di bawah ini :



Gambar II.1 Tampilan google maps

(2013, <https://maps.google.com>)

### Konsep dasar basis data

Basis data (*database*) adalah kumpulan informasi yang disimpan di dalam komputer secara sistematik sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Perangkat lunak atau *software* yang digunakan untuk mengelola dan memanggil *query* basis data disebut sistem manajemen basis data (*database management system, DBMS*) .

Sesungguhnya ada beberapa macam *Database*  yaitu *Database* rasioanal merupakan *database* yang popular saat ini dan telah diterapkan pada berbagai *platform*, dari PC hingga mini komputer.

Tujuan utama dari *Database* adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Kebutuhan basis data dalam sistem informasi meliputi:

1. Memasukkan, menyimpan, dan mengambil.
2. Membuat laporan berdasarkan data yang telah disimpan  
   (Shalahudin, 2008).

Dalam sebuah *database*, pengguna dapat menempatkan 1 atau lebih file atau tabel. Pada *file* atau tabel inilah sesungguhnya data data disimpan atau ditempatkan. Misalnya dalam *file* mahasiswa dapat di tempatkan nomor mahasiswa, nama, pembimbing kerja praktek, dan pembimbing tugas akhir.

Keuntungan dari *database* adalah:

1. Mengurangi duplikasi data
2. Meningkatkan intensitas data
3. Memelihara indenpendensi data
4. Meningkatkan keamanan data
5. Memelihara konsistensi data
6. Memanipulasi data lebih canggih
7. Mudah untuk digunakan

Kekurangan dari *database* adalah:

1. Sistem lebih rumit, jadi memerlukan tenaga ahli dalam desain, pemrograman, dan implementasi.
2. Lebih mahal.
3. Karena semua data di tempatkan terpusat, maka dapat terjadi kerusakan software dan hardware.
4. Bila ada akses yang tidak benar, dapat terjadi kerusakan.
5. Proses pemeliharaan dapat memakan waktu yang lama karena ukurannya yang besar.
6. Proses back up data memakan waktu.

Operasi-operasi dasar yang dapat dilakukan berkenan dengan *database* meliputi:

1. *Create database*

Perintah ini digunakan untuk membuat basis data baru, yang identik dengan pembuatan lemari arsip baru.

1. *Drop database*

Perintah ini digunakan untuk menghapus basis data. Perintah ini identik dengan perusakan lemari arsip (sekaligus beserta isinya, jika memang ada isinya).

1. *Create table*

Perintah ini digunakan untuk membuat *file* atau tabel baru ke suatu basis data. Perintah ini identik dengan penambahan map arsip baru ke sebuah lemari arsip yang telah ada.

1. *Drop table*

Perintah ini digunakan untuk menghapus *file* atau tabel dari suatu basis data. Perintah ini identik dengan perusakan map arsip lama yang ada di sebuah lemari arsip.

1. *Insert*

Perintah ini digunakan untuk menambahkan atau mengisi data baru ke sebuah *file* atau tabel di sebuah basis data yang identik dengan penambahan lembar arsip ke sebuah map arsip.

1. *Retrieve/search*

Perintah ini digunakan untuk mengambil data dari sebuah *file* atau tabel yang identik dengan pencarian lembaran arsip dari sebuah map arsip.

1. *Update*

Perintah ini berguna untuk mengubah data dari sebuah *file* atau tabel yang identik dengan perbaikan isi lembaran arsip yang ada di sebuah map arsip.

1. *Delete*

Perintah ini digunakan untuk menghapus data dari sebuah *file* atau table yang identik dengan penghapusan sebuah lembaran arsip yang ada disebuah lembaran arsip.

### MySQL

MySQL di kembangkan sekitar tahun 1994 oleh sebuah perusahaan pengembang perangkat lunak dan konsultan database bernama MySQL AB yang bertempat di Swedia. Pada waktu itu perusahaan tersebut masih bernama TcX Data Konsult AB, dan tujuan awal dikembangkan MySQL untuk mengembangkan aplikasi berbasis *web* pada *client* (Saputra, 2008).

MySQL merupakan salah satu *database server* yang berkembang di lingkungan *open source* dan didistribusikan secara *free* (gratis) di bawah lisensi GPL (*General Public Licence*), Lisensi GPL ini di umumkan sejak meluncurnya versi 3.23.19. Dengan Lisensi ini pengguna MySQL dapat melihat *source code* MySQL tanpa harus mengeluarkan biaya.

MySQL merupakan *Relational Database Management System* (RDBMS) *server*. RDBMS adalah program yang memungkinkan pengguna database untuk membuat, mengelola dan menggunakan data pada suatu model *relational*.Dengan demikian, tabel-tabel yang ada pada database memiliki relasi (hubungan) antara satu tabel dengan lainnya. RDBMS di buat dengan menggunakan bahasa *standart* dalam *database*, yaitu *Structured Query Language* (SQL) yang di masukkan oleh pengguna ke dalam data tersebut.

### Server

Secara global *server* dapat diartikan sebagai pusat dan difungsikan sebagai “pelayan” yang berguna untuk pengiriman data dan penerimaan data serta mengatur pengiriman dan penerimaan data diantara komputer-komputer yang terhubung atau dengan kata lain *server* berfungsi menyediakan pelayanan terhadap *client*.

*Web server* adalah sebuah perangkat lunak yang dikenal dengan *server* *web* dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman *web* yang umummya berbentuk dokumen. *Web server* yang terkenal diantaranya adalah *Apache2*. *Apache2* merupakan *web server* antar-*platform* perangkat lunak yang dapat digunakan dibeberapa sistem operasi yang berbeda.

### WWW (World Wide *Web*)

*World Wide Web* (biasa disingkat WWW) atau *web* adalah salah satu dari sekian banyak layanan yang ada di *internet*. Layanan ini paling banyak digunakan di *internet* untuk menyampaikan informasi karena sifatnya mendukung multimedia. Artinya informasi tidak hanya disampaikan melalui teks, tapi juga gambar, video dan suara. Berikut ini beberapa jenis layanan *World Wide Web* yang perlu diketahui, antara lain:

1. *Web Mail Service*

Salah satu layanan yang paling banyak digunakan orang di *internet* adalah *e-mail*, dan untuk bisa saling berkirim *e-mail* kita harus memiliki *e-mail* *account* tertentu. Saat ini sudah cukup banyak situs *web* yang memberikan *email account* secara gratis tanpa bayar. Kita bisa langsung memesan sebuah *e-mail account* dengan nama tertentu sesuai dengan keinginan kita. Contoh: *mail.yahoo.com, www.hotmail.com*, dll.

1. *Search Engine*

Sering kali kita memerlukan untuk mencari sesuatu dari *web*, akan tetapi kita belum mengetahui dimanakah alamat situs *web* yang memberikan layanan atau informasi tersebut. Padahal jumlah situs *web* saat ini sudah mencapai ratusan juta situs dengan berbagai layanan yang ditawarkan.

Untuk itu kita memerlukan sebuah aplikasi bantu yang akan mencari apa yang kita butuhkan di situs-situs *web*. Alat bantu itu disebut *Search Engine*. Dengan *search engine* kita akan cepat menemukan alamat situs yang berisi informasi yang kita butuhkan. Contoh situs *search enngine : www.google.com, www.yahoo.com,* dll.

1. *Web Hosting*

Setelah proses *design* dan proses penyusunan sebuah situs *web* selesai, maka selanjutnya kita akan *upload* situs tersebut di *internet*. Untuk itu kita memerlukan *web space* untuk meletakkan situs yang telah jadi itu. Bila kita berlangganan pada sebuah ISP, biasanya kita diberikan *space* khusus untuk keperluan ini. Akan tetapi bila tidak demikian, kita dapat menggunakan layanan *web hosting* gratis yang ditawarkan di *internet*. Contoh situs yang menawarkan *web* hosting gratis yaitu: *www.geocities.com,www.coolfreepages.com, www.lonex.com,* dll.

1. *Portal*

Kalau dalam dunia nyata kita kenal *mall* atau *plaza*, yang menyediakan semua yang kita butuhkan saat belanja, maka di dunia maya pun ada fasilitas seperti ini. Di dalam satu situs *web*, tidak hanya menawarkan satu layanan saja, tetapi bermacam-macam layanan sekaligus. Situs seperti ini, disebut sebagai portal. Contoh portal: *www.yahoo.com,* www.astaga.com, dll (Rizal, 2013).

### *Web* browser

Pengertian *browser* adalah suatu aplikasi *and user*, yang digunakan untuk membuka *website* di *internet*. Dimana *website* bisa berupa *file*, informasi, dokumen, dan *graphics*, bahkan *file* multimedia lainnya, atau dengan kata lain *web browser* adalah suatu perangkat lunak yang digunakan untuk mengalokasikan tampilan halaman *web* atau memperbolehkan seseorang pengguna *web* untuk menampilkan dan berinteraksi dengan dokumen-dokumen yang di-*host* oleh *server-server* *web*. Ada banyak *browser* dipasaran seperti *Internet Explorer*, *Chrome*, *Netscape Navigator*, *Opera*, dan terakhir yang lagi naik daun adalah *Mozilla Firefox*. Setiap *browser* mempunyai karateristik masing-masing dan yang pasti mempunyai kelebihan dan kekurangannya (Kadir, 2001).

*Internet Explorer* dan *Netscape Navigator* merupakan *Graphical Browser* yang berarti keduanya dapat menampilkan *graphic* sebagus teks. Sebagai tambahan, kebanyakan *browser modern* saat ini dapat menampilkan informasi multimedia termasuk video dan suara, meskipun keduanya membutuhkan *plug-in* untuk beberapa format. Beberapa fitur *web browser*:

1. *Activex*

*Activex* adalah pertumbuhan luar dari *Microsoft technologies* yang disebut OLE (*Object Linking and Embedding*) dan COM (*Component Object Mode*).

1. DHTML

*Dynamic* HTML atau DHTML adalah terminal yang digunakan untuk kumpulan teknologi, digunakan bersama untuk membuat *website* interaktif dan animasi *website* dengan menggunakan kombinasi dari *static markup language* (seperti HTML), *bahasa client-side scripting language* (seperti *JavaScrip*), bahasa definisi presentasi (*Cascading Style Sheets*), dan *Document Object Model.*

1. *Web Cache*

*Web cache* adalah *cache* dari dokumen *web* (contoh HTML *pages, image*) dalam perintah mengurangi penggunaan *bandwidth*, *server load*, dan *perceived* “*lag*”.

1. CSS

*Cascading Style Sheets* (CSS) adalah suatu bahasa *style sheet* yang digunakan untuk mengatur tampilan suatu dokumen yang ditulis dalam bahasa *markup*.

1. *Autocompletion*

Autocompletion adalah fitur provider dengan banyak source code text editors, word processors, dan *web* browser.

### UML (Unified Modeling Language)

*Unified Modeling Language* (UML) adalah sebuah bahasa untuk menetukan, visualisasi, kontruksi, dan mendokumentasikan *artifact* (bagian dari informasi yang digunakan atau dihasilkan dalam suatu proses pembuatan perangkat lunak. *Artifact* dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya (Ashary, 2013).

UML merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan sistem yang besar dan kompleks. UML tidak hanya digunakan dalam proses pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan.

Bagian-bagian utama dari UML adalah *view*, diagram, model *element*, dan *general mechanism*.

1. *View*

*View*digunakan untuk melihat sistem yang dimodelkan dari beberapa aspek yang berbeda. *View* bukan melihat grafik, tapi merupakan suatu abstraksi yang berisi sejumlah diagram.

Beberapa jenis *view* dalam UML antara lain: *use case view, logical view, component view, concurrency view, dan deployment view.*

1. *Use case view*

Mendeskripsikan fungsionalitas sistem yang seharusnya dilakukan sesuai yang diinginkan *external actors*. *Actor* yang berinteraksi dengan sistem dapat berupa *user* atau sistem lainnya.

*View* ini digambarkan dalam *use case* diagrams dan kadang-kadang dengan *activity diagrams*. *View* ini digunakan terutama untuk pelanggan, perancang (*designer*), pengembang (*developer*), dan penguji sistem (*tester*).

1. *Logical view*

Mendeskripsikan bagaimana fungsionalitas dari sistem, struktur statis (*class*, *object*, dan *relationship*) dan kolaborasi dinamis yang terjadi ketika object mengirim pesan ke object lain dalam suatu fungsi tertentu.

View ini digambarkan dalam *class diagrams* untuk struktur statis dan dalam *state*, *sequence*, *collaboration*, dan *activity* diagram untuk model dinamisnya. *View* ini digunakan untuk perancang (*designer*) dan pengembang (*developer*).

1. *Component view*

Mendeskripsikan implementasi dan ketergantungan modul. Komponen yang merupakan tipe lainnya dari *code module* diperlihatkan dengan struktur dan ketergantungannya juga alokasi sumber daya komponen dan informasi *administrative* lainnya. View ini digambarkan dalam *component view* dan digunakan untuk pengembang (*developer*).

1. *Concurrency view*

Membagi sistem ke dalam proses dan prosesor. *View* ini digambarkan dalam diagram dinamis (*state*, *sequence*, *collaboration*, dan *activity diagrams*) dan diagram implementasi (*component* dan *deployment diagrams*) serta digunakan untuk pengembang (*developer*), pengintegrasi (*integrator*), dan penguji (*tester*).

1. *Deployment view*

Mendeskripsikan fisik dari sistem seperti komputer dan perangkat (*nodes*) dan bagaimana hubungannya dengan lainnya. *View* ini digambarkan dalam *deployment diagrams* dan digunakan untuk pengembang (*developer*), pengintegrasi (*integrator*), dan penguji (*tester*).

1. Diagram

Diagram berbentuk grafik yang menunjukkan simbol elemen model yang disusun untuk mengilustrasikan bagian atau aspek tertentu dari sistem. Sebuah diagram merupakan bagian dari suatu *view* tertentu dan ketika digambarkan biasanya dialokasikan untuk *view* tertentu. Adapun jenis diagram antara lain :

1. *Use Case Diagram*

*Use case* adalah abstraksi dari interaksi antara sistem dan *actor*. *Use case* bekerja dengan cara mendeskripsikan tipe interaksi antara *user* sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. *Use case* merupakan konstruksi untuk mendeskripsikan bagaimana sistem akan terlihat di mata *use*r. Sedangkan *use case* diagram memfasilitasi komunikasi diantara analis dan pengguna serta antara analis dan *client*.

1. *Class Diagram*

*Class* adalah dekripsi kelompok obyek-obyek dengan *property*, perilaku (operasi) dan relasi yang sama. Sehingga dengan adanya *class* diagram dapat memberikan pandangan global atas sebuah sistem. Hal tersebut tercermin dari *class-class* yang ada dan relasinya satu dengan yang lainnya. Sebuah sistem biasanya mempunyai beberapa *class* diagram. *Class* diagram sangat membantu dalam visualisasi struktur kelas dari suatu sistem.

1. *Component* Diagram

*Component software* merupakan bagian fisik dari sebuah sistem, karena menetap di komputer tidak berada di benak para analis. Komponen merupakan implementasi *software* dari sebuah atau lebih *class*. Komponen dapat berupa *source code*, komponen biner, atau *executable component*. Sebuah komponen berisi informasi tentang *logic class* atau *class* yang diimplementasikan sehingga membuat pemetaan dari *logical view* ke *component view*. Sehingga *component* diagram merepresentasikan dunia riil yaitu *component software* yang mengandung *component*, *interface* dan *relationship*.

1. *Deployment* Diagram

Menggambarkan tata letak sebuah sistem secara fisik, menampakkan bagian-bagian *software* yang berjalan pada bagian-bagian *hardware*, menunjukkan hubungan komputer dengan perangkat (*nodes*) satu sama lain dan jenis hubungannya. Di dalam *nodes*, *executeable component* dan *object* yang dialokasikan untuk memperlihatkan unit perangkat lunak yang dieksekusi oleh *node* tertentu dan ketergantungan komponen.

1. *State* Diagram

Menggambarkan semua *state* (kondisi) yang dimiliki oleh suatu *object*dari suatu *class* dan keadaan yang menyebabkan *state* berubah. Kejadian dapat berupa *object* lain yang mengirim pesan. *State class* tidak digambarkan untuk semua *class*, hanya yang mempunyai sejumlah *state* yang terdefinisi dengan baik dan kondisi *class* berubah oleh *state* yang berbeda.

1. *Sequence* Diagram

*Sequence* Diagram digunakan untuk menggambarkan perilaku pada sebuah *scenario*. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara *objec*  juga interaksi antara *object*, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.

1. *Collaboration* Diagram

Menggambarkan kolaborasi dinamis seperti *sequence diagrams*. Dalam menunjukkan pertukaran pesan, *collaboration diagrams* menggambarkan *object* dan hubungannya (mengacu ke konteks). Jika penekannya pada waktu atau urutan gunakan *sequence diagrams*, tapi jika penekanannya pada konteks gunakan *collaboration* diagram.

1. *Activity* Diagram

Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti *use case* atau interaksi.

Adapun tujuan dari penggunaan UMLyaitu:

1. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
2. Menyatukan praktek-praktek terbaik yang terdapat dalam pemodelan.
3. Memberikan model yang siap pakai, bahasa pemodelan visual yang ekspresif untuk mengembangkan dan saling menukar model dengan mudah dan dimengerti secara umum.
4. UML bisa juga berfungsi sebagai sebuah (*blue print*) cetak biru karena sangat lengkap dan detail. Dengan cetak biru ini maka akan bisa diketahui informasi secara detail tentang *coding* program atau bahkan membaca program dan menginterpretasikan kembali ke dalam bentuk diagram (*reserve enginering*).

Perangkat lunak yang mendukung pembuatan diagaram UML:

1. *StarUML*

*StarUML* adalah sebuah proyek *open source* untuk mengembangkan cepat, fleksibel, *extensible*, *featureful*, dan bebas-tersedia UML/*platform* MDA berjalan pada *platform* Win32. Tujuan dari proyek *StarUML* adalah untuk membangun sebuah alat pemodelan perangkat lunak dan juga *platform* yang menarik adalah pengganti alat UML komersial seperti *Rational Rose*, dan sebagainya.

1. *Acceleo*

*Acceleo* adalah generator kode yang mengubah model menjadi kode. *Acceleo* mudah digunakan dan menyediakan “dari rak” generator (Jee,. Bersih, Php …) dan *template* editor untuke *eclipse*.

1. *Argo*UML

*Argo*UML adalah *open source* UML *modeling tool* terkemuka dan termasuk dukungan untuk semua diagram UML standar 1,4. Ini berjalan pada setiap *platform Java* dan tersedia dalam bahasa sepuluh. *Argo*UML ditulis seluruhnya di Jawa dan menggunakan *Java* Kelas *Foundation*. Hal ini memungkinkan *Argo*UML untuk berjalan di hampir semua *platform*.

### PHP (Hypertext Preprocessor)

PHP adalah bahasa pemrograman berbasis *server side*. Artinya, untuk dapat menjalankan *script* PHP harus membutuhkan *web* *server*. PHP sendiri sebenarnya merupakan singkatan dari “Hypertext Preprocessor”, yang merupakan sebuah bahasa *scripting* tingkat tinggi yang dipasang pada dokumen HTML. Sebagian besar sintaks dalam PHP mirip dengan bahasa C, Java dan Perl, namun pada PHP ada beberapa fungsi yang lebih spesifik. Sedangkan tujuan utama dari penggunaan bahasa ini adalah untuk memungkinkan perancangan *web* dinamis dan dapat bekerja secara otomatis. PHP pertama kali dibuat oleh Rasmus Lerdroft, seorang *programmer* C. Pada waktu itu PHP masih bernama FI (*Form Interpreted*)*,* yang wujudnya berupa sekumpulan *script* yang digunakan untuk mengolah data form dari *web*. Jadi semula PHP digunakan untuk menghitung jumlah pengunjung di dalam *web* nya (Kadir, 2003).

### HTML *(HyperText Markup Language)*

Bermula dari sebuah bahasa yang sebelumnya banyak digunakan di dunia penerbitan dan percetakan yang disebut dengan SGML (Standard Generalized Markup Language), HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan halaman *web* . HTML saat ini merupakan standar *Internet*yang didefinisikan dan dikendalikan penggunaannya oleh World Wide *Web* Consortium (W3C). HTML5 adalah revisi kelima dari HTML (yang pertama kali diciptakan pada tahun 1990 dan versi keempatnya, HTML4, pada tahun 1997) dan hingga bulan Juni 2011 masih dalam pengembangan. Pada HTML 5 diperkenalkan beberapa elemen baru dan atribut yang merefleksikan tipikal penggunaan *web*site modern. Tidak semua teknologi di atas dimasukkan pada spesifikasi HTML5 W3C, meski teknologi tersebut telah termasuk dalam spesifikasi milik WHATWG HTML.

### CSS (*Cascading Style Sheet*)

CSS adalah salah satu [bahasa pemrograman](http://id.wikipedia.org/wiki/Bahasa_pemrograman) [*web*](http://id.wikipedia.org/wiki/Web) untuk mengendalikan beberapa komponen dalam sebuah *web* sehingga akan lebih terstruktur dan seragam.

Sama halnya *styles* dalam [aplikasi](http://id.wikipedia.org/wiki/Aplikasi) pengolahan kata seperti [Microsoft Word](http://id.wikipedia.org/wiki/Microsoft_Word) yang dapat mengatur beberapa *style*, misalnya *heading*, *subbab*, *bodytext*, *footer*, *images*, dan *style* lainnya untuk dapat digunakan bersama-sama dalam beberapa berkas ([*file*](http://id.wikipedia.org/wiki/File)). Pada umumnya CSS dipakai untuk memformat tampilan halaman *web* yang dibuat dengan [bahasa](http://id.wikipedia.org/wiki/Bahasa) [HTML](http://id.wikipedia.org/wiki/HTML) dan [XHTML](http://id.wikipedia.org/wiki/XHTML). CSS dapat mengendalikan [ukuran](http://id.wikipedia.org/wiki/Ukuran) [gambar](http://id.wikipedia.org/wiki/Gambar), [warna](http://id.wikipedia.org/wiki/Warna) bagian tubuh pada teks, warna [tabel](http://id.wikipedia.org/wiki/Tabel), ukuran *border*, warna *border*, warna [*hyperlink*](http://id.wikipedia.org/wiki/Hyperlink), warna [*mouse*](http://id.wikipedia.org/wiki/Mouse) *over*, spasi antar paragraf, spasi antar teks, *margin* kiri, kanan, atas, bawah, dan parameter lainnya. CSS adalah bahasa [*style* *sheet*](http://id.wikipedia.org/wiki/Stylesheet) yang digunakan untuk mengatur tampilan dokumen.Dengan adanya CSS memungkinkan kita untuk menampilkan halaman yang sama dengan format yang berbeda. CSS3 adalah versi terbaru dari CSS yang mampu melakukan banyak hal dalam desain *web*site. CSS2 mendukung penentuan posisi konten, *downloadable*, huruf *font*, tampilan pada tabel / *table layout* dan media tipe untuk printer.Kehadiran versi CSS yang kedua diharapkan lebih baik dari versi pertama (Sugiri, 2007).

CSS3 juga dapat melakukan animasi pada halaman *web*site, diantaranya animasi warna hingga animasi 3D. Dengan CSS3 desainer lebih dimudahkan dalam hal kompatibilitas *web*sitenya pada *smartphone* dengan dukungan fitur baru yakni media *query*. Selain itu, banyak fitur baru pada CSS3 seperti: *multiple background*, *border-radius*, *drop-shadow*, *border-image*, CSS *Math*, dan CSS *Object Model*. Ada dua sifat CSS yaitu internal dan eksternal :

1. Sifat yang pertama adalahinternal, maka [skrip](http://id.wikipedia.org/wiki/Script) itu dimasukkan secara langsung ke halaman [*web*site](http://id.wikipedia.org/wiki/Website) yang akan didesain Kalau halaman *web* yang lain akan didesain dengan model yang sama, maka skrip CSS itu harus dimasukkan lagi ke dalam halaman *web* yang lain itu.
2. Sifat yang kedua adalah eksternal, di mana skrip CSS dipisahkan dan diletakkan dalam berkas khusus.Nanti, cukup gunakan semacam tautan menuju berkas CSS itu jika halaman *web* yang didesain akan dibuat seperti [model](http://id.wikipedia.org/wiki/Model) yang ada di skrip tersebut.

Fakta Menggunakan CSS diantaranya :

1. Telah didukung oleh kebanyakan browser versi terbaru, tetapi tidak didukung oleh browser-browser lama.
2. Lebih fleksibel dalam penempatan posisi *layout*. Dalam *layouting* CSS, kita mengenal *Z-Index* untuk menempatkan objek dalam posisi yang sama.
3. Menjaga HTML dalam penggunaan tag yang minimal, hal ini berpengaruh terhadap ukuran berkas dan kecepatan pengunduhan.
4. Dapat menampilkan konten utama terlebih dahulu, sementara gambar dapat ditampilkan sesudahnya.
5. Penerjemahan CSS setiap browser berbeda, tata letak akan berubah jika dilihat di berbagai browser.
6. CSS adalah layouting "Masa Depan" dengan penggabungan bersama [XHTML](http://id.wikipedia.org/wiki/XHTML).

### JavaScript

JavaScript adalah bahasa pemrograman berbasis prototipe yang berjalan disisi klien. Jika kita berbicara dalam konteks *web*, sederhananya, kita dapat memahami JavaScript sebagai bahasa pemrograman yang berjalan khusus untuk di browser atau halaman *web* agar halaman *web* menjadi lebih hidup. Kalau dilihat dari suku katanya terdiri dari dua suku kata, yaitu Java dan Script. Java adalah Bahasa pemrograman berorientasi objek, sedangkan Script adalah serangkaian instruksi program.

Secara fungsional, JavaScript digunakan untuk menyediakan akses script pada objek yang dibenamkan (*embedded*). Contoh sederhana dari penggunaan JavaScript adalah membuka halaman *pop up*, fungsi validasi pada *form* sebelum data dikirimkan ke *server*, merubah *image* kursor ketika melewati objek tertentu, dan lain lain.

**Dalam** pengelolaan pemrograman JavaScript, diantaranya JavaScript adalah ‘case sensitive’, yang artinya JavaScript membedakan huruf besar dan huruf kecil. Jika Anda pernah belajar bahasa pemrograman seperti Turbo C atau C++, maka sama seperti bahasa pemrograman tersebut, dimana huruf T tidak sama dengan huruf t.  
Dalam bahasa pemrograman JavaScript juga, sebagai contoh fungsi perintah var tidak boleh ditulis Var dan juga tidak boleh ditulis VAR (huruf besar semua), yang benar adalah var (huruf kecil semua). Perintah lain adalah *new Date* tidak boleh *ditulis new date* (huruf kecil semua), dsb (Kadir, 2003).

### AJAX

*Asynchronous JavaScript and XMLHTTP* atau disingkat AJAX, adalah suatu teknik pemrograman berbasis *web* untuk menciptakan aplikasi *web* interaktif. Tujuannya adalah untuk memindahkan sebagian besar interaksi pada komputer *web* *surfer,* melakukan pertukaran data dengan *server* di belakang layar *background*, sehingga halaman *web* tidak harus dibaca ulang secara keseluruhan setiap kali seorang pengguna melakukan perubahan. Hal ini akan meningkatkan interaktivitas, kecepatan, dan usability. Ajax merupakan kombinasi dari:

1. DOM yang diakses dengan *client side scripting language*, seperti *VBScript* dan implementasi *ECMAScript* seperti JavaScript dan JScript, untuk menampilkan secara dinamis dan berinteraksi dengan informasi yang ditampilkan.
2. Objek XML HTTP dari *Microsoft* atau XML Http *Request* yang lebih umum di implementasikan pada beberapa browser. Objek ini berguna sebagai kendaraan pertukaran data asinkronus dengan *web* server. Pada beberapa *framework* AJAX, *element HTML IFrame* lebih dipilih daripada XMLHTTP atau XMLHttpRequest untuk melakukan pertukaran data dengan *web* *server*.
3. XML umumnya digunakan sebagai dokumen transfer, walaupun format lain juga memungkinkan, seperti HTML, plain text. XML dianjurkan dalam pemakaian teknik AJaX karena kemudahan akses penanganannya dengan memakai DOM.Seperti halnya DHTML, LAMP, atau SPA, Ajax bukanlah teknologi spesifik, melainkan merupakan gabungan dari teknologi yang dipakai bersamaan.

Pada penerapannya cukup rumit jika harus membangun *web*site dengan konsep AJAX, saat ini banyak library khusus yang berusaha mempermudahnya. JQuery merupakan salah satunya. Menyederhanakan penulisan Javascript biasa. Semboyan JQuery adalah “*Write less, do more*” (Indonesia ; Menulis kurang, berbuat lebih banyak) atau dengan kata lain kesederhanaan dalam penulisan code, tetapi menghasilnya tampilan yang lebih. Sebenarnya inilah yang menjadi daya tarik tersendiri buat para pengembang *web* untuk menggunakan JQuery (Komputer, 2007)

### JQuery

JQuery adalah *library* JavaScript yang memungkinkan kita untuk membuat program *web* pada suatu halaman *web*, tanpa harus secara eksplisit kita menambahkan *event* atau properti pada halaman *web* tersebut (Saputra, 2011).

Dengan JQuery, suatu halaman *web* yang menjadi aplikasi *web*, jika dilihat *source*nya, akan terlihat seperti dokumen HTML biasa, tidak ada kode JavaScript yang terlihat langsung. Teknik pemrograman *web* seperti ini disebut sebagai *unobstrusive* JavaScript *programming*.

JQuery dikembangkan pertama kali oleh John Resig, yang dibuat lebih ramping dari *library Prototype* yang menjadi inspirasi dari *libarary* JQuery ini. Secara pemrograman, JQuery memiliki kemiripan seperti *Prototype*. JQuery, merupakan *library* yang sangat ramping, core dari *library* ini dalam keadaan terkompres hanya berukuran sekitar 19KB.

### J’SON

Bahasan tentang Java di *WEB*. J’SON (JavaScript *Object* *Notation*) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari [Bahasa Pemprograman JavaScript](http://javascript.crockford.com/), [Standar ECMA-262 Edisi ke-3 - Desember 1999](http://www.ecma-international.org/publications/files/ecma-st/ECMA-262.pdf). J’SON merupakan format teks yang tidak bergantung pada bahasa pemprograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan J’SON ideal sebagai bahasa pertukaran-data. (Shalahudin, 2011). J’SON terbuat dari dua struktur:

1. Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (*object*), rekaman (*record*), struktur (*struct*), kamus (*dictionary*), tabel *hash* (*hash table*), daftar berkunci (*keyed list*), atau [*associative array*](http://en.wikipedia.org/wiki/Associative_array).
2. Daftar nilai terurutkan (*an ordered list of values*). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (*array*), vektor (*vector*), daftar (*list*), atau urutan (*sequence*).

Struktur-struktur data ini disebut sebagai struktur data *universal*. Pada dasarnya, semua bahasa pemprograman moderen mendukung struktur data ini dalam bentuk yang sama maupun berlainan. Hal ini pantas disebut demikian karena format data mudah dipertukarkan dengan bahasa-bahasa pemprograman yang juga berdasarkan pada struktur data ini.

### Framework

*Framework* secara sederhana dapat diartikan kumpulan dari fungsi-fungsi/ prosedur-prosedur dan class-class untuk tujuan tertentu yang sudah siap digunakan sehingga bisa lebih mempermudah dan mempercepat pekerjaan seorang programmer, tanpa harus membuat fungsi atau class dari awal (Sidik, 2012).

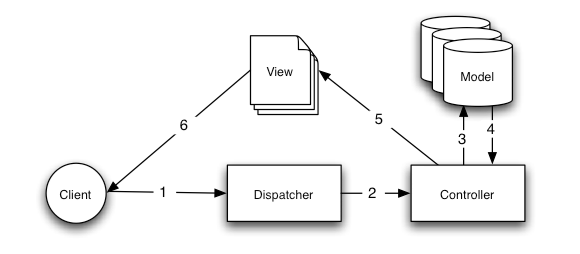
Ada beberapa alasan mengapa menggunakan *Framework*:

1. Mempercepat dan mempermudah pembangunan sebuah aplikasi *web*.
2. Relatif memudahkan dalam proses *maintenance* karena sudah ada pola tertentu dalam sebuah *framework* (dengan syarat programmer mengikuti pola standar yang ada). Umumnya *framework* menyediakan fasilitas-fasilitas yang umum dipakai sehingga kita tidak perlu membangun dari awal (misalnya validasi, ORM, *pagination*, *multiple database*, *scaffolding*, pengaturan *session*, *error*

*handling*, dll

1. Lebih bebas dalam pengembangan jika dibandingkan CMS.

Konsep MVC *Framework* dapat dilihat pada gambar II.2 dibawah ini :



**Gambar II.2** Konsep mvc framework(Ribkhan, 2012, http://buggzilla.files.wordpress.com/2012/05/basic\_mvc.png?w=812)

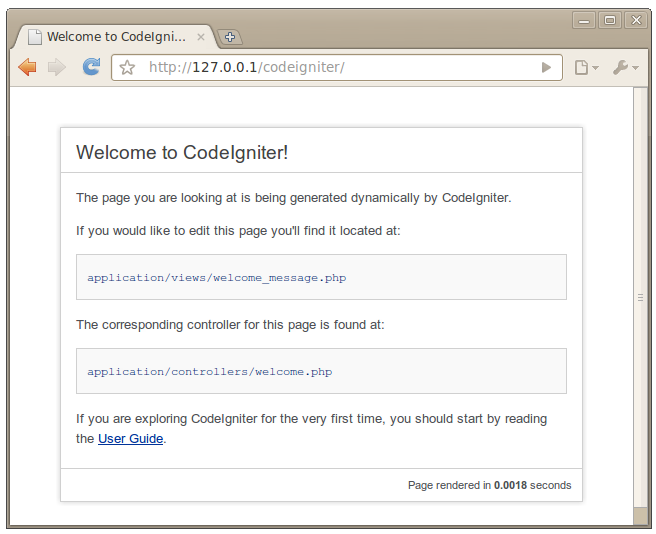
### CodeIgniter

*CodeIgniter* adalah sebuah *framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi *web* berbasis PHP dibandingkan jika menulis semua kode program dari awal.

*CodeIgniter* menyediakan banyak *library* untuk mengerjakan tugas-tugas yang umumnya ada pada sebuah aplikasi berbasis *web*. Selain itu, struktur dan susunan logis dari C*odeIgniter* membuat aplikasi yang dibuat menjadi semakin teratur dan rapi.

*CodeIgniter* pertama kali dibuat oleh Rick Ellis, CEO Ellislab, Inc. (http://ellislab.com), sebuah perusahaan yang memproduksi sebuah CMS (*content management system*) yang cukup handal, yaitu *expression engine* (http://www.expressionengine.com). Saat ini CodeIgniter dikembangkan dan dimaintain oleh *expression engine development team* (Sidik, 2012).

Tampilan *default* *Framework CodeIgniter* dapat dilihat pada gambar II.3.



Gambar II. 3 Tampilan default framework codeigniter

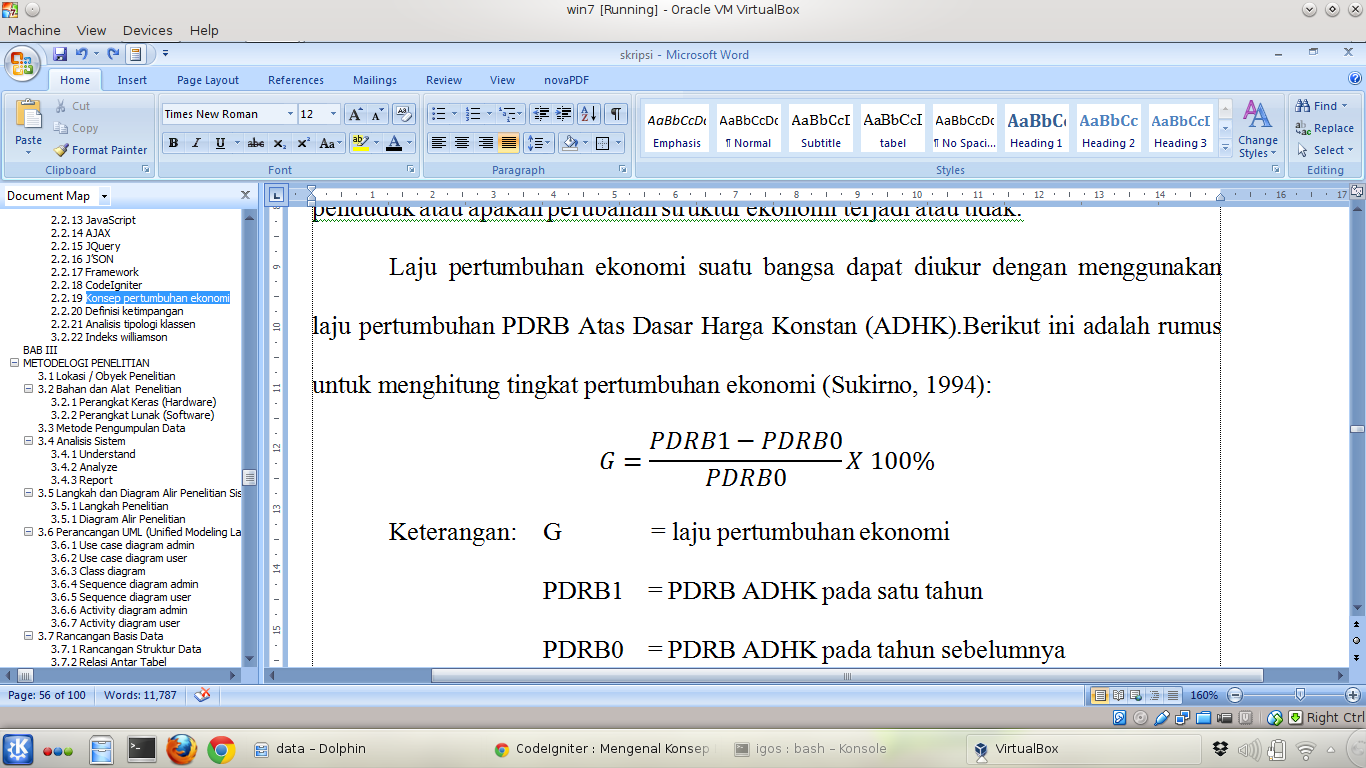
(Jon, 2013, http://wiki.bitnami.com/@api/deki/files/290/=codeigniter.png)

### Konsep pertumbuhan ekonomi

Pertumbuhan ekonomi adalah proses kenaikan output perkapita dalam jangka panjang. Pertumbuhan ekonomi berkaitan dengan kenaikan output perkapita dengan adanya dua sisi yang perlu diperhatikan, yaitu sisi *output* total dan sisi jumlah penduduknya. *Output* perkapita adalah output total dibagi jumlah penduduk (Boediono, 1999).

Pertumbuhan ekonomi diartikan sebagai kenaikan produk domestik bruto tanpa memandang apakah kenaikan itu lebih besar atau lebih kecil dari tingkat pertumbuhan penduduk atau apakah perubahan struktur ekonomi terjadi atau tidak.

Laju pertumbuhan ekonomi suatu bangsa dapat diukur dengan menggunakan laju pertumbuhan PDRB Atas Dasar Harga Konstan (ADHK). Berikut ini adalah rumus untuk menghitung tingkat pertumbuhan ekonomi (Sukirno, 1994) dapat dilihat pada gambar II.4.



Gambar II.4 Rumus menghitung tingkat pertumbuhan ekonomi

Keterangan: G = laju pertumbuhan ekonomi

PDRB1 = PDRB ADHK pada satu tahun

PDRB0 = PDRB ADHK pada tahun sebelumnya

Faktor-faktor yang mempengaruhi pertumbuhan ekonomi suatu masyarakat adalah :

1. Akumulasi modal, termasuk investasi baru yang berbentuk tanah atau lahan, peralatan fiskal, dan sumber daya manusia (*human resources*). Akumulasi modal terjadi apabila adanya proporsi tertentu dari pendapatan sekarang yang ditabung dan kemudian di investasikan untuk memperbesar output pada masa yang akan dating. Pabrik-pabrik, mesin-mesin, peralatan-peralatan dan barang-barang baru akan meningkatkan stok modal (*capital stock*) fiskal suatu Negara yaitu jumlah riil bersih dari semua barang-barang modal produktif secara fiskal sehingga pada gilirannya akan memungkinkan Negara tersebut mencapai tingkat output yang lebih besar.
2. Pertumbuhan penduduk

Pertumbuhan penduduk dan hal-hal yang berhubungan dengan jumlah angkatan kerja (*labor force*) secara tradisional telah diangkap sebagai faktor positif dalam merangsang pertumbuhan ekonomi. Artinya semakin banyak angkatan kerja berarti semakin produktif tenaga kerja, sedangkan semakin banyak penduduk maka akan meningkatkan potensi pasar domestik.

1. Kemajuan Teknologi

Kemajuan teknologi disebabkan oleh cara-cara baru dan cara-cara lama yang diperbaiki dalam melakukan pekerjaan-pekerjaan tradisional

### Definisi ketimpangan

Ketimpangan mengacu pada standar hidup relatif dari seluruh masyarakat. Sebab ketimpangan antar wilayah yaitu adanya perbedaan faktor anugrah awal (*Endowment Factor*). Perbedaan inilah yang menyebabkan tingkat pembangunan di berbagai wilayah dan daerah berbeda-beda, sehingga menimbulkan gap atau jurang kesejahteraan di berbagai wilayah tersebut (Sukirno, 2006).

Ketimpangan adalah mengacu pada standar hidup relatif dari seluruh masyarakat. Pada tingkat ketimpangan yang maksimum, kekayaan dimiliki hanya satu orang saja, dan tingkat kemiskinan sangat tinggi (Kuncoro, 2004).

Faktor-faktor yang menyebabkan ketimpangan :

1. Konsentrasi kegiatan ekonomi yang lemah.
2. Alokasi mobilitas faktor produksi yang rendah antar daerah.
3. Perbedaan sumber daya alam.
4. Perbedaan kondisi demografis antar daerah.
5. Kurang lancarnya perdagangan antar daerah.

### Analisis tipologi klassen

Alat analisis Tipologi Klassen digunakan untuk mengetahui gambaran tentang pola dan struktur celah fiskal masing-masing daerah. Tipologi Klassen pada dasarnya mengelompokkan daerah berdasarkan dua indikator utama, yaitu pertumbuhan ekonomi daerah dan pendapatan perkapita daerah (Sjafrizal, 2009). Dalam penelitian ini, pengelompokkan daerah berdasarkan dua indikator utama, yaitu laju pertumbuhan PDRB dan celah fiskal perkapita. Dengan menentukan rata-rata pertumbuhan PDRB sebagai sumbu vertikal dan rata-rata celah fiskal perkapita sebagai sumbu horizontal.

Pengelompokan daerah ini selanjutnya dapat digambarkan pada matriks Tipologi Klassen berikut ini :

Tabel II 1 Tipologi daerah

|  |  |  |
| --- | --- | --- |
| PDRB per  Kapita (Y)  Laju Pertumbuhan (r) | Yi > Y | Yi < Y |
| Ri > r | Daerah maju tumbuh cepat | Daerah bertumbuh cepat |
| Ri < r | Daerah maju tapi tertekan | Daerah relative tertinggal |

Keterangan: ri = laju pertumbuhan PDRB kecamatan i

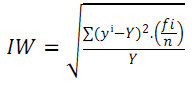
r = laju pertumbuhan PDRB kabupaten

Yi = PDRB kecamatan i

Y = PDRB kabupaten

### Indeks williamson

Indeks Williamson menunjukkan ketimpangan kemampuan keuangan daerah Kab/Kota di Provinsi Sumatera Barat. Jika Indeks Williamson keuangan suatu daerah mendekati 0, berarti ketimpangan kemampuan keuangan semakin kecil. Sedangkan jika Indeks Williamson keuangan suatu daerah mendekati 1, berarti ketimpangan kemampuan keuangan semakin besar (Sjafrizal, 1997). Indeks Williamson dapat didefinisikan sebagai berikut (Naskah Akademik Revisi UU No. 32 Bab V Tentang Keuangan Daerah) dapat dilihat pada gambar II.5.



Gambar II.5 Rumus indeks wiliamson

Keterangan : IW = Indeks Williamson

yⁱ = PDRB perkapita Kecamatan i

Y = PDRB perkapita rata-rata Kabupaten

fi = jumlah penduduk Kecamatan i

n = jumlah penduduk Kabupaten

# BAB III

# METODELOGI PENELITIAN

## Lokasi / Obyek Penelitian

Lokasi penelitian ini dilakukaan di Badan Pusat Statistik Kabupaten Klaten Jawa Tengah.

## Bahan dan Alat Penelitian

### Perangkat Keras (Hardware)

1. Prosesor Intel (R) Pentium Dual Core CPU B960 @2.20 GHz
2. Memori 6 GB DDR3 PC10600 dan Hard Disk 320 GB

### Perangkat Lunak (Software)

1. Sistem Operasi IGOS NUSANTARA 9.0 sebagai server dan client
2. MySQL sebagai perancangan basis data.
3. Apache2 sebagai aplikasi *web* server.
4. *Web browser* seperti : Mozilla Firefox, Google Chrome
5. *Text Editor* Geany digunakan untuk penulisan script program
6. PHP sebagai bahasa pemrograman
7. Pencil digunakan untuk perancangan sistem
8. Microsoft *Office*  digunakan untuk penulisan laporan
9. *Visual Paradigm for UML* digunakan untuk perancangan alur sistem

## Metode Pengumpulan Data

Untuk mendapatkan suatu uraian yang nyata dan luas, diperlukan data-data yang dapat mendukung dalam proses pengumpulan data. Metode pengumpulan data dalam penelitian ini diperoleh dari:

1. Metode Wawancara

Merupakan suatu metode pengumpulan data dengan cara mengadakan tanya jawab kepada pihak-pihak yang terkait dan mengetahui masalah yang ada.

1. Metode Observasi

Merupakan suatu metode untuk memperoleh data dengan cara melakukan pengamatan langsung pada obyek yang akan diteliti.

1. Metode Studi Pustaka

Metode studi pustaka merupakan sebuah cara dalam pengumpulan data dengan mempelajari bahan pustaka berupa dokumentertulis ataupun berupa gambar yang berkaitan dengan penelitian yang diteliti oleh penulis.

## Analisis Sistem

1. **Identify**

Mengidentifikasi masalah yang ada, agar sistem yang dibangun dapat mencapai sasaran atau target dari masalah-masalah tersebut.

### Understand

Memahami kerja dari sistem yang ada. Langkah ini dapat dilakukan dengan mempelajari secara terinci dan bagaimana operasi dari sistem informasi yang ada sebelumnya dan mencoba untuk menganalisa pernasalahan-permasalahan, kelemahan-kelemahan dan kebutuhan-kebutuhan pengguna untuk dapat memberikan rekomendasi pemecahan masalah dari sistem yang sudah ada.

### Analyze

Menganalisa keseluruhan sistem informasi. Dalam analisis ini meliputi analisis kelemahan sistem dan analisis kebutuhan informasi.

### Report

Membuat laporan dari hasil analisa yang diperoleh. Dari survey dan penelitian sampai pembuatan aplikasinya.

## Langkah dan Diagram Alir Penelitian Sistem Informasi

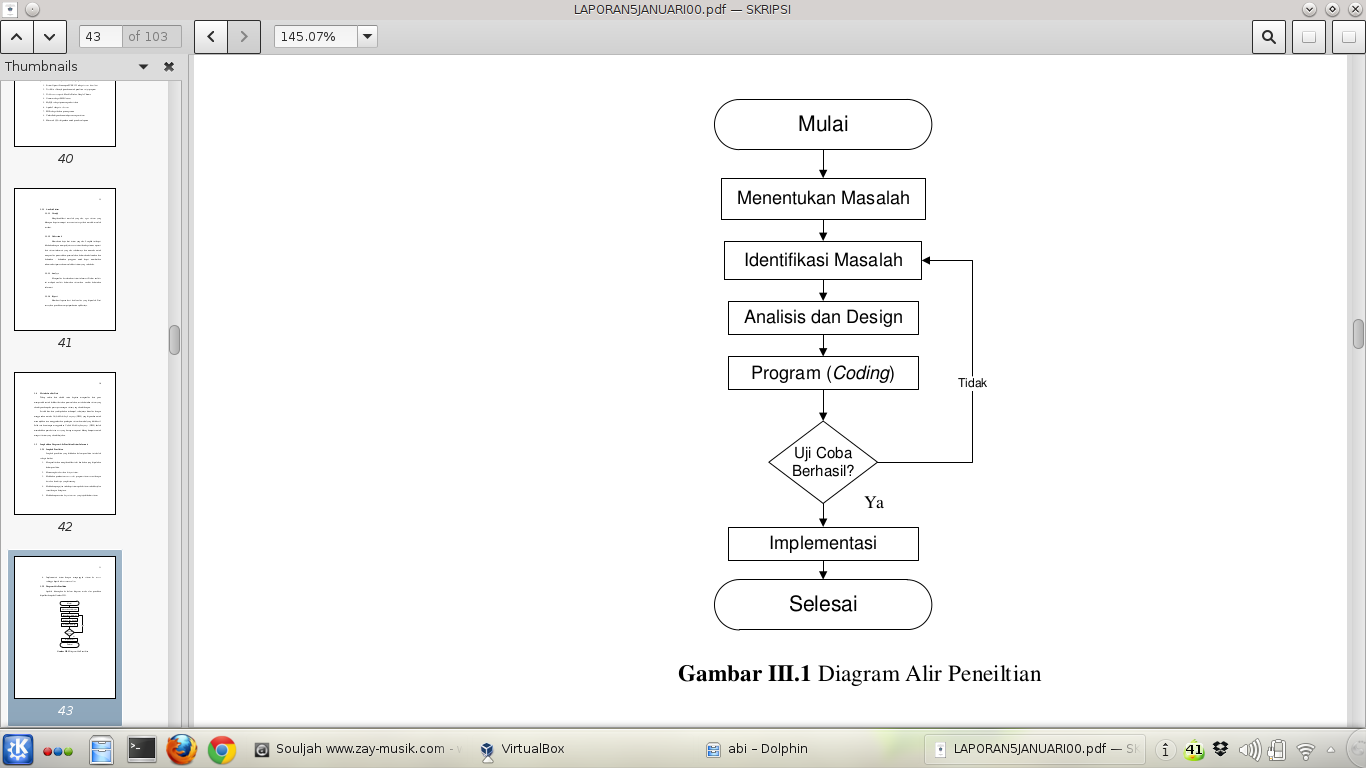
### Langkah Penelitian

Langkah penelitian yang dilakukan dalam penelitian ini adalah sebagai berikut:

1. Menganalisis dan mengidentifikasi alat dan bahan yang diperlukan dalam penelitian.
2. Merancang database dan design sistem.
3. Melakukan pembuatan *source code* program sistem sesuai dengan database dan design yang dirancang.
4. Melakukan pengujian terhadap sistem apakah sistem sudah berjalan sesuai dengan keinginan.
5. Melakukan pencarian bugs atau error yang terjadi dalam sistem.
6. Implementasi sistem dengan mengunggah sistem ke server sehingga dapat diakses secara online.

### Diagram Alir Penelitian

Apabila dituangkan ke dalam diagram, maka alur penelitian diperlihatkan pada Gambar III.1.



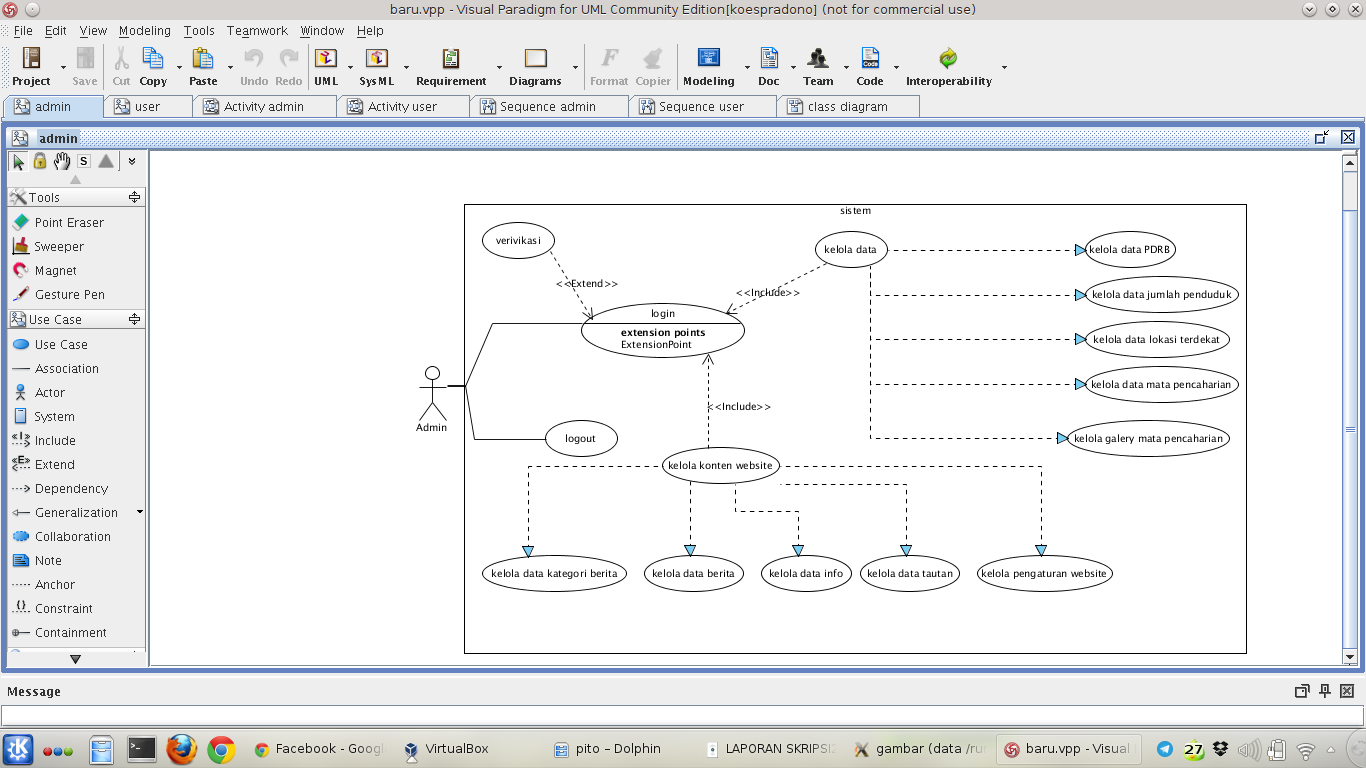
Gambar III.1 Diagram alir penelitian

## Perancangan UML (*Unified Modeling Language*)

Tahap analisa data adalah suatu kegiatan menganalisa data guna memperoleh suatu klasifikasi data dan permasalahan, serta kebutuhan sistem yang akan digunakan pada proses perancangan sistem yang akan dibangun. Setelah data-data yan diperlukan terkumpul, selanjutnya dianalisa dengan menggunakan metode *Unified Modeling Language* (UML)yang digunakan untuk menunjukkan atau menggambarkan pembagian sistem ke modul yang lebih kecil. Salah satu keuntungan menggunakan *Unified Modeling Language* (UML) adalah memudahkan pemakai atau *user* yang kurang menguasai bidang komputer untuk mengerti sistem yang akan dikerjakan.

### Use case diagram admin

*Use case* diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Sebuah *use case* mempresentasikan sebuah intraksi antara admin dengan sistem. *Use case* merupakan sebeuah pekerjaan tertentu, misalnya *login* ke sistem. Seorang/sebuah admin adalah sebagai entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. Proses *login* admin dan aktivitas yang dilakukan oleh admin dapat dilihat pada gambar III.2.

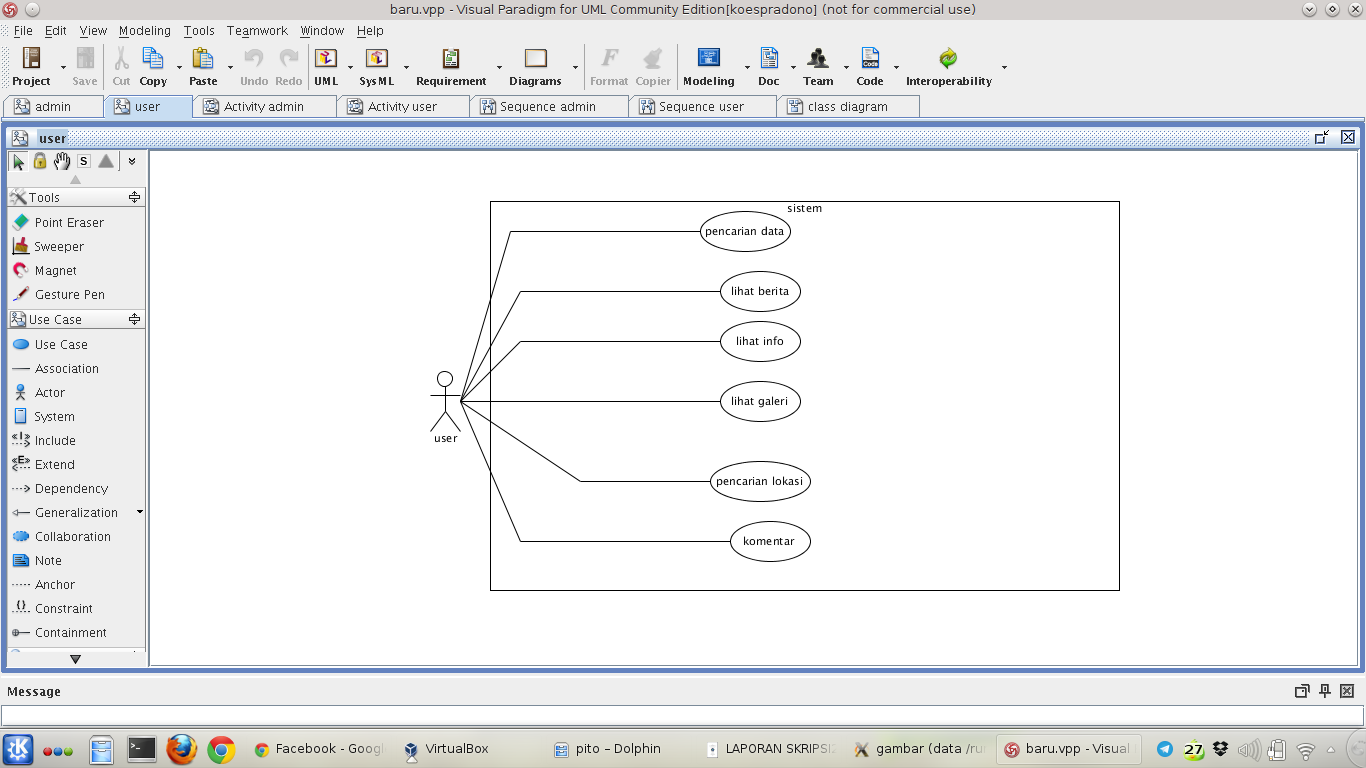


Gambar III.2 Use case diagram admin

Dari design pada gambar III.2 use case diagram admin dapat disimpulkan, admin dapat melakukan beberapa aktivitas di antaranya yaitu admin dapat login ke sistem, admin dapat kelola konten *web*site dan kelola data.

### Use case diagram user

Selain admin, maka user pun dapat melakukan aktivitas. Didalam sistem ini seseorang dikatakan user hanya dapat mengunjungi halaman *web*site dan tidak mempunyai hak akses login ke sistem. Aktivitas-aktivitas yang dilakukan oleh seorang *user* dapat dilihat pada gambar III.3.

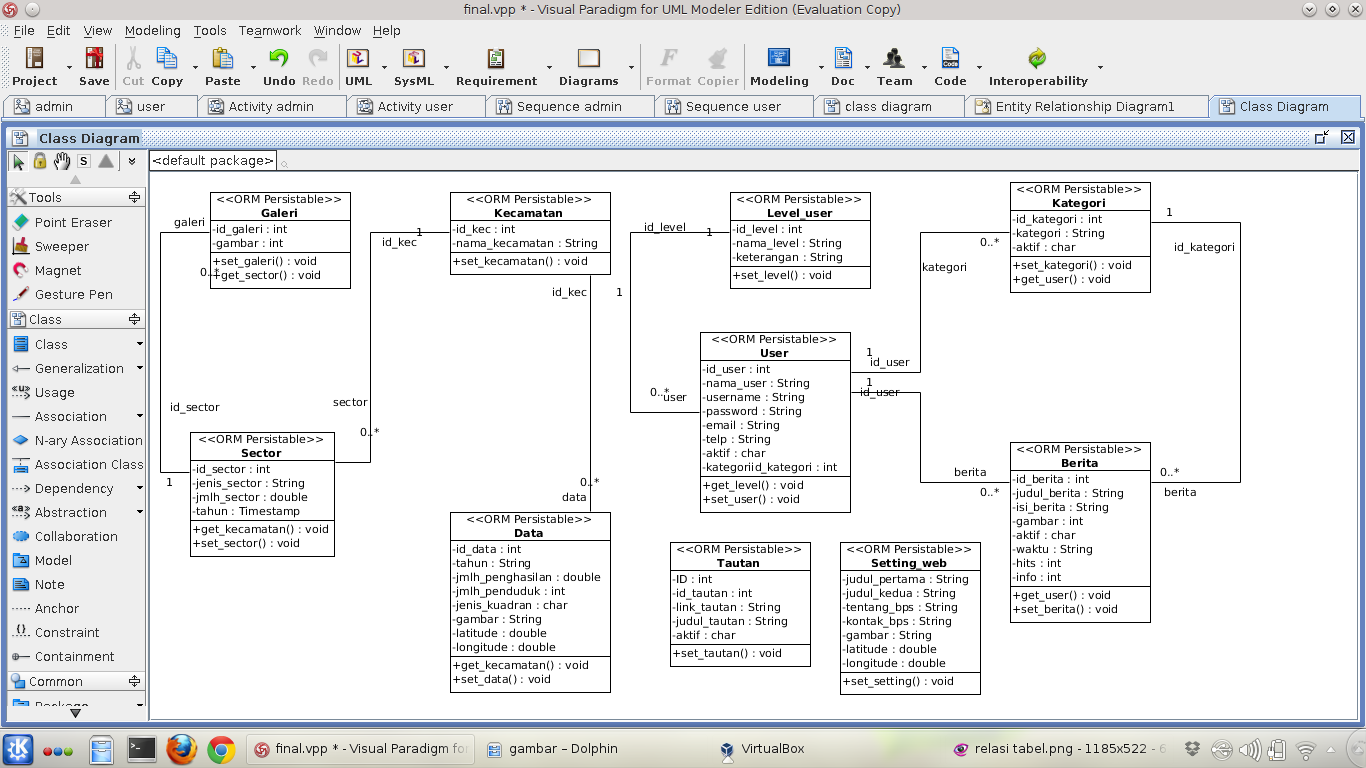


Gambar III.3 Use Case Diagram User

Dari design pada gambar III.3 *use case* diagram *user* di atas, dapat melakukan aktivitas seperti melakukan pencariian suatu data dalam sistem, melihat berita, melihat info, pencarian lokasi, melihat *gallery* foto, memberikan komentar.

### Classdiagram

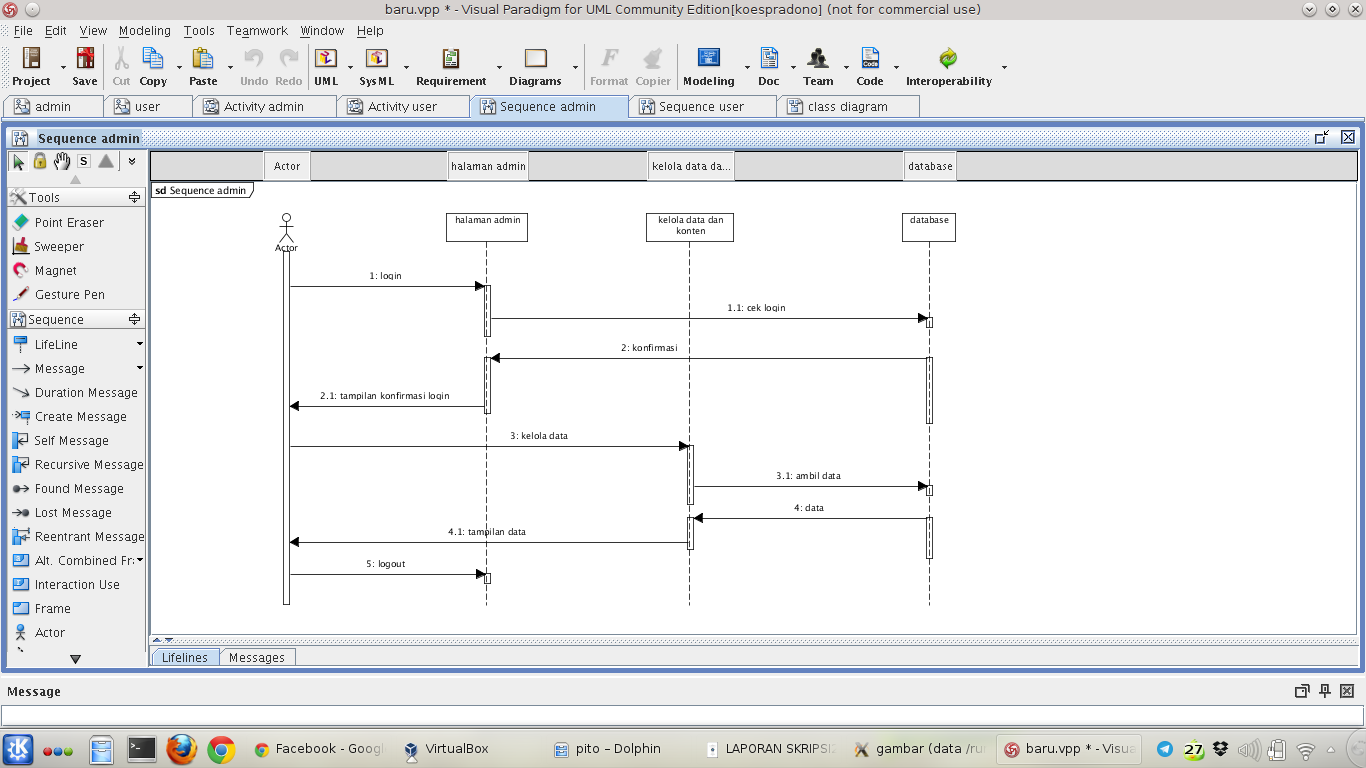
Berikut adalah *class* diagram yang memberikan gambaran dan interaksi secara terstruktur dari sistem yang dibuat. Class diagram di lihat pada gambar III.4.



Gambar III.4 Class diagram

### Sequence diagram admin

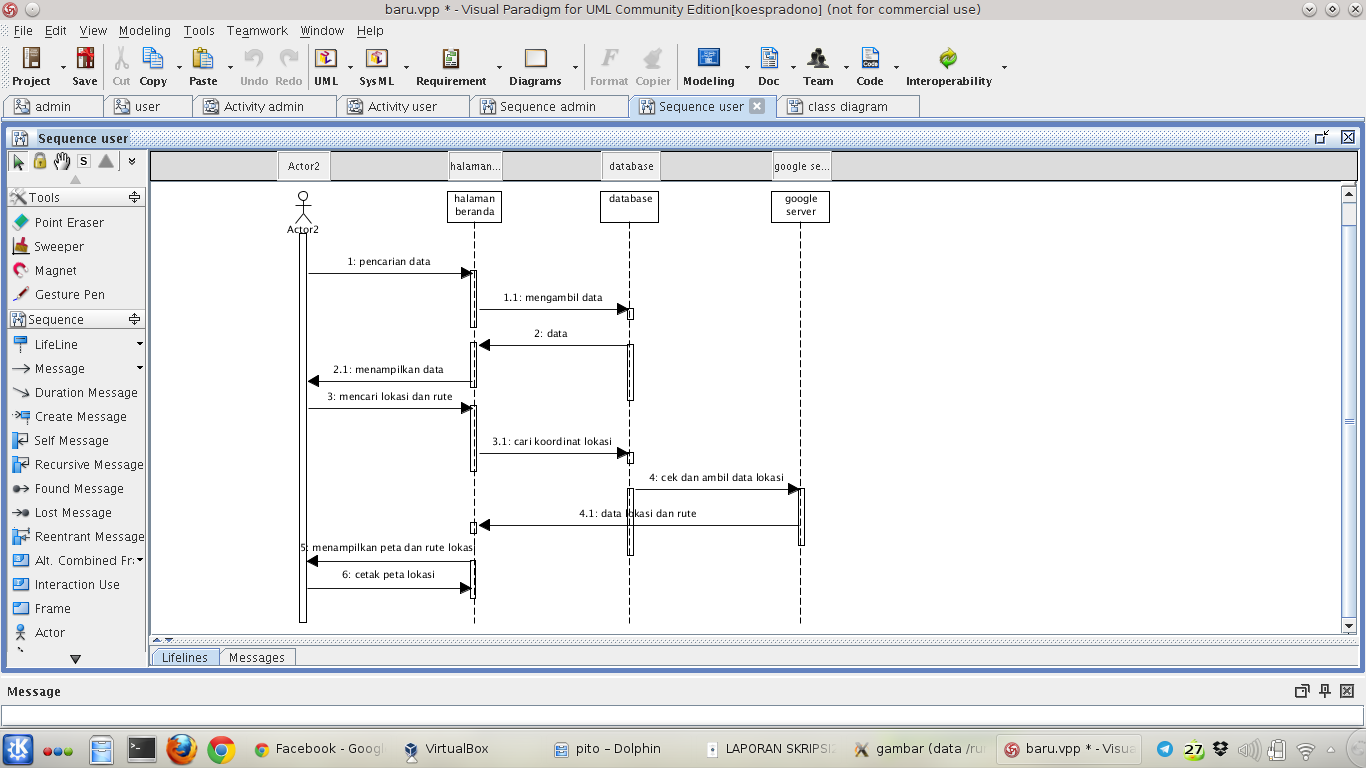
Berikut ini adalah *sequence* diagram yang menggambarkan proses aktivitas admin untuk login dan melakukan aktivitas dalam sistem. Sequence diagram ini menggambarkan aktivitas pertama kali yang dilakukan admin adalah login kemudian akan di validasi oleh sistem dengan diproses pada *database* yang jika proses *login* berhasil, akan masuk pada halaman admin. Setelah *login*, admin dapat mengelola data yang berkaitan dengan kos dan konten dari sistem. Pengelolaan data ini yang akan disimpan kedalam *database*. Sequence diagram admin dapat dilihat pada Gambar III.5.



Gambar III. 5 Sequence Diagram Admin

### Sequence diagram user

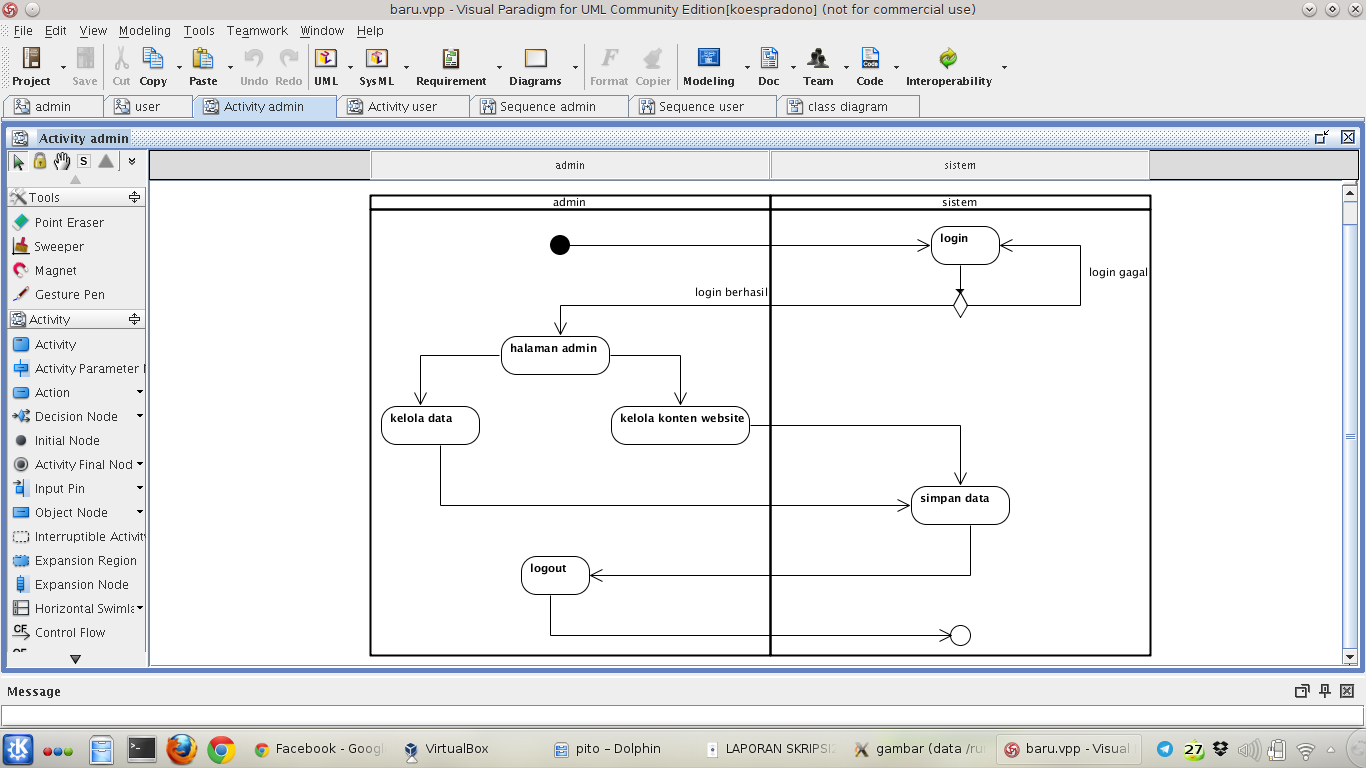
Berikut ini adalah sequence diagram yang menggambarkan proses aktivitas dari user untuk melakukan aktivitas dalam sistem. Aktivitas tersebut diantaranya melakukan mencari data dan mencari lokasi dan rute. Sequence diagram *user* dapat dilihat pada Gambar III.6



Gambar III.6 Sequence diagram user

### Activity diagram admin

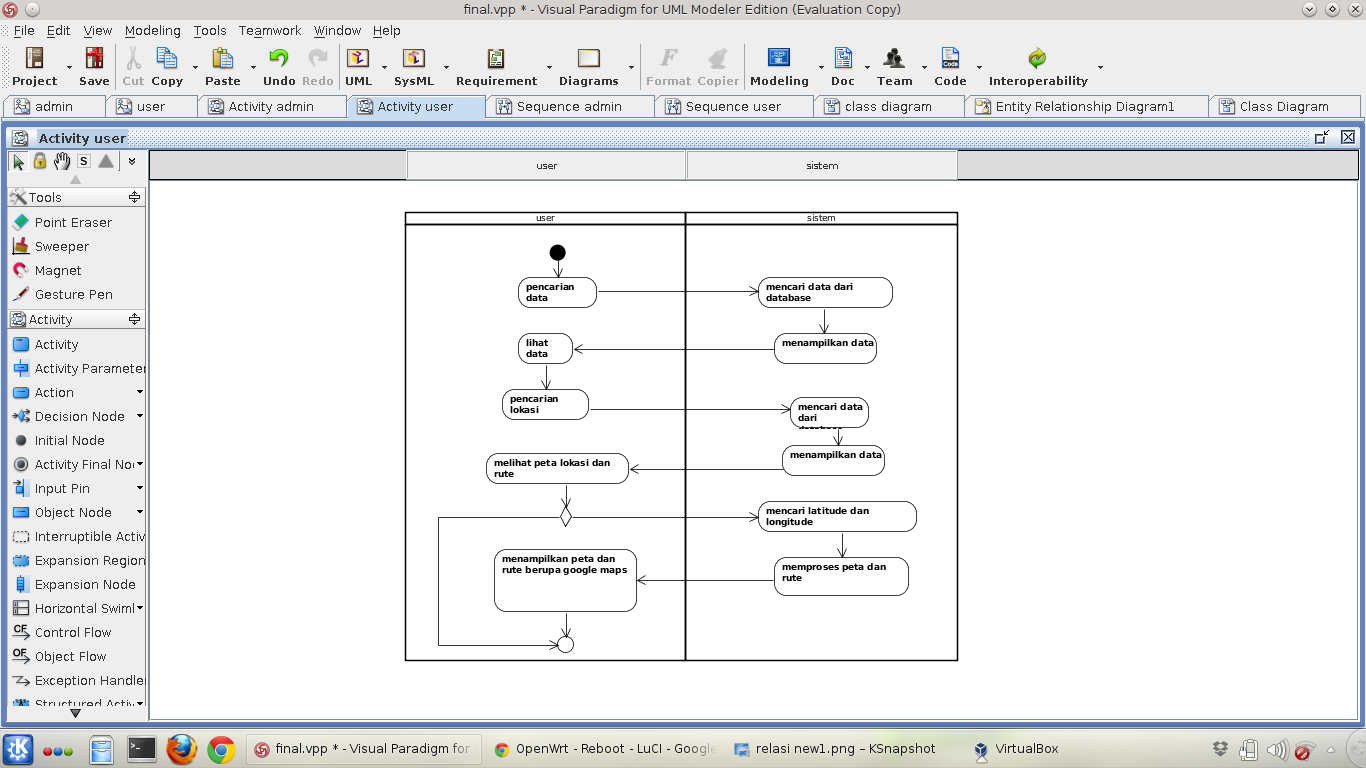
Berikut ini adalah activity diagram yang menggambarkan aktifitas admin. Activity diagram admin ditampilkan pada Gambar III.7.



Gambar III.7 Activity Diagram Admin

### Activity diagram user

Berikut ini adalah activity diagram yang menggambarkan aktifitas user. Activity diagram admin ditampilkan pada Gambar III.8.



Gambar III.8 Activity Diagram User

## Rancangan Basis Data

Dalam membuat suatu sistem informasi yang mampu menyimpan atau merekam data dibutuhkan suatu media penyimpanan yang disebut database. Dalam hal ini perancangan Basis Data menggunakan penyimpanan dengan aplikasi MySQL versi 5.5.34

### Rancangan Struktur Data

1. Tabel level\_user

Nama table : level\_user

*Primary key* : id\_level

*Foreign key* : -

Tabel III.1 level user

|  |  |  |
| --- | --- | --- |
| Field | Type | Null |
| id\_level | int (11) | No |
| nama\_level | varchar (20) | No |
| keterangan | varchar (100) | No |

Pada table III.1 di atas, berfungsi untuk mengatur hak akses atau level user Tabel ini terdiri 3 fileld antara lain: id\_level, nama\_level, dan keterangan.

1. Tabel user

Nama tabel : user

*Primary key* : id\_user

*Foreign key* : id\_level

Tabel III.2 user

|  |  |  |
| --- | --- | --- |
| Field | Type | Null |
| id\_user | int (11) | No |
| id\_level | int (11) | No |
| nama\_user | varchar (50) | No |
| username | varchar(30) | No |
| password | varchar(100) | No |
| email | varchar(50) | No |
| telp | varchar(15) | Yes |
| aktif | enum(‘Y,’N’) | No |

Pada tabel III.2 di atas, berfungsi untuk menyimpan data User login menurut levelnya dan mengatur user yang aktif maupun tidak aktif yang digunakan untuk login ke sistem baik itu admin, dan karyawan. Tabel ini terdiri dari 8 filed antara lain: id\_user, id\_level, nama\_user, username, password, email, telp, dan aktif.

1. Tabel kategori

Nama table : kategori

*Primary key* : id\_kategori

*Foreign key* : id\_user

Tabel III.3 kategori

|  |  |  |
| --- | --- | --- |
| Field | Type | Null |
| id\_kategori | int (11) | No |
| id\_user | int (11) | No |
| kategori | varchar (50) | No |
| aktif | enum (‘Y,’N’) | No |

Pada tabel III.3 di atas, digunakan untuk menyimpan data kategori berita yang nanti dipakai untuk menentukan berita yang tergolong dalam kategori tertentu. Tabel ini terdiri dari 4 filed antara lain: id\_kategori, id\_user, kategori, dan aktif.

1. Tabel berita

Nama table : berita

*Primary key* : id\_berita

*Foreign key* : id\_kategori, id\_user

Tabel III.4 berita

|  |  |  |
| --- | --- | --- |
| Field | Type | Null |
| id\_berita | int (11) | No |
| id\_kategori | int (11) | No |
| id\_user | int (11) | No |
| judul\_berita | varchar (100) | No |
| isi\_berita | text | No |
| gambar | varchar (100) | YES |
| aktif | enum (‘Y,’N’) | No |
| waktu | timestamp | No |
| hits | int (11) | No |
| info | int (1) | No |

Pada tabel III.4 di atas, digunakan untuk menyimpan data isi berita tentang badan pusat statistik. Tabel ini terdiri dari 7 filed antara lain: id\_berita, id\_kategori, id\_user, judul\_berita, isi\_berita, gambar, dan aktif.

1. Tabel tautan

Nama tabel : tautan

*Primary key* : id\_tautan

*Foreign key* : -

Tabel III.5 tautan

|  |  |  |
| --- | --- | --- |
| Field | Type | Null |
| id\_tautan | int (11) | No |
| link\_tautan | varchar (50) | No |
| judul\_tautan | varchar (40) | No |
| aktif | enum (‘Y,’N’) | No |

Pada tabel III.5 di atas, digunakan untuk menyimpan data link / tautan yang akan ditampilkan di halaman front end. Tabel ini terdiri dari 4 filed antara lain: id\_tautan, link\_tautan, judul\_tautan, dan aktif.

1. Tabel setting *web*

Nama tabel : setting\_*web*

*Primary key* : -

*Foreign key* : -

Tabel III.6 setting\_*web*

|  |  |  |
| --- | --- | --- |
| Field | Type | Null |
| judul\_pertama | varchar (100) | No |
| judul\_kedua | varchar (100) | No |
| tentang\_bps | text | No |
| kontak\_bps | text | No |
| gambar\_bps | varchar (100) | Yes |
| latitude | double | No |
| longitude | double | No |

Pada tabel III.6 di atas, digunakan untuk menyimpan data yang berhubungan dengan pengaturan *web*site seperti nama sistem informasi, Tentang kami, kontak, dan letak alamat pada *Google Maps*. Tabel ini terdiri dari 7 filed antara lain: judul\_pertama, judul\_kedua, tentang\_bps, kontak\_bps, gambar\_bps, latitude, dan longitude.

1. Tabel kecamatan

Nama tabel : kecamatan

*Primary key* : id\_kec

*Foreign key* : -

Tabel III.7 kecamatan

|  |  |  |
| --- | --- | --- |
| Field | Type | Null |
| id\_kec | int (11) | No |
| nama\_kecamatan | varchar (30) | No |

Pada tabel III.7 di atas, digunakan untuk menyimpan id kecamatan dan nama kecamatan di Kabupaten Klaten. Tabel ini terdiri dari 2 filed antara lain: id\_kec, dan nama\_kecamatan.

1. Table data

Nama tabel : data

*Primary key* : id\_data

*Foreign key* : id\_kec

Tabel III.8 data

|  |  |  |
| --- | --- | --- |
| Field | Type | Null |
| id\_data | int (11) | No |
| id\_kec | int (11) | No |
| tahun | year | No |
| jmlh\_penghasilan | double | No |
| jmlh\_penduduk | int (6) | No |
| jenis\_kuadran | enum(‘I’,’II’,’III’,’IV’) | No |
| gambar | varchar (100) | No |
| latitude | double | No |
| longitude | double | No |

Pada tabel III.8 di atas, digunakan untuk menyimpan data, tahun, penghasilan jumlah penghasilan pertahun, jumlah penduduk\_jenis, kuadran, dan menyimpan letak Kecamatan di Kabupaten Klaten. Tabel ini terdiri dari 9 filed antara lain: id\_data, id\_kec, tahun, jmlh\_penghasilan, jmlh\_penduduk, jenis\_kuadran, gambar, latitude, dan longitude.

1. Table sector

Nama tabel : sector

*Primary key* : id\_sector

*Foreign key* : -

Tabel III.9 sector

|  |  |  |
| --- | --- | --- |
| Field | Type | Null |
| id\_sector | int (11) | No |
| jenis\_sector | varchar (30) | No |
| jumlah | double | No |
| tahun | Year | No |

Pada tabel III.9 di atas, digunakan untuk menyimpan jenis penghasil dan jumlah penghasilan pertahun tiap kecamatan. Tabel ini terdiri dari 4 filed antara lain: id\_sector, jenis\_sector, jumlah, dan tahun.

1. Table galeri

Nama tabel : galeri

*Primary key* : id\_galeri

*Foreign key* : id\_sector

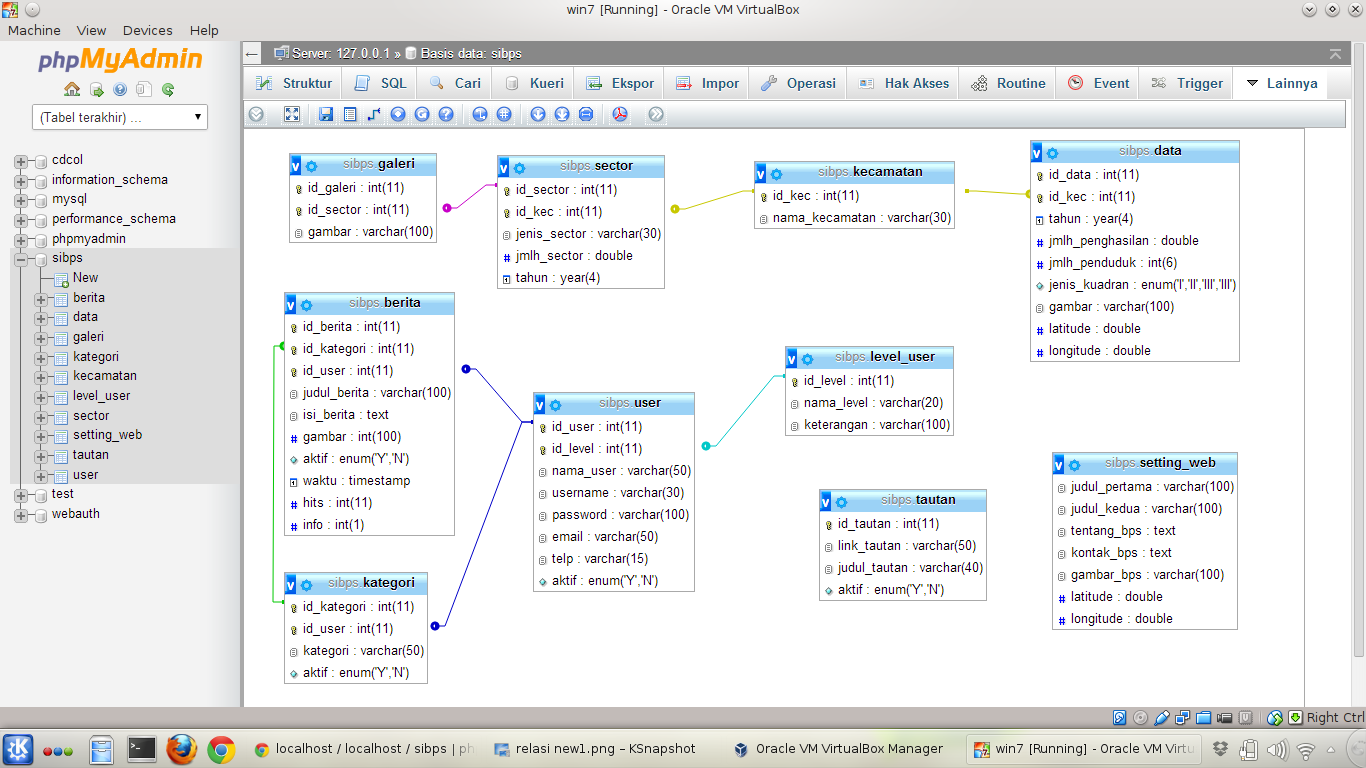
Tabel III.10 galeri

|  |  |  |
| --- | --- | --- |
| Field | Type | Null |
| id\_galeri | int(11) | No |
| id\_sector | int(11) | No |
| gambar | varchar(11) | No |

Pada tabel III.10 di atas, digunakan untuk menyimpan gambar galeri berdasarkan mata pencaharian di Kabupaten Klaten. Tabel ini terdiri dari 3 filed antara lain: id\_galeri, id\_sector, dan gambar.

### Relasi Antar Tabel

Relasi dimaksudkan sebagai saling keterkaitan antara satu kesatuan data dalam satu atau beberapa tabel. Hubungan antara tabel dapat dilihat pada Gambar III.9.



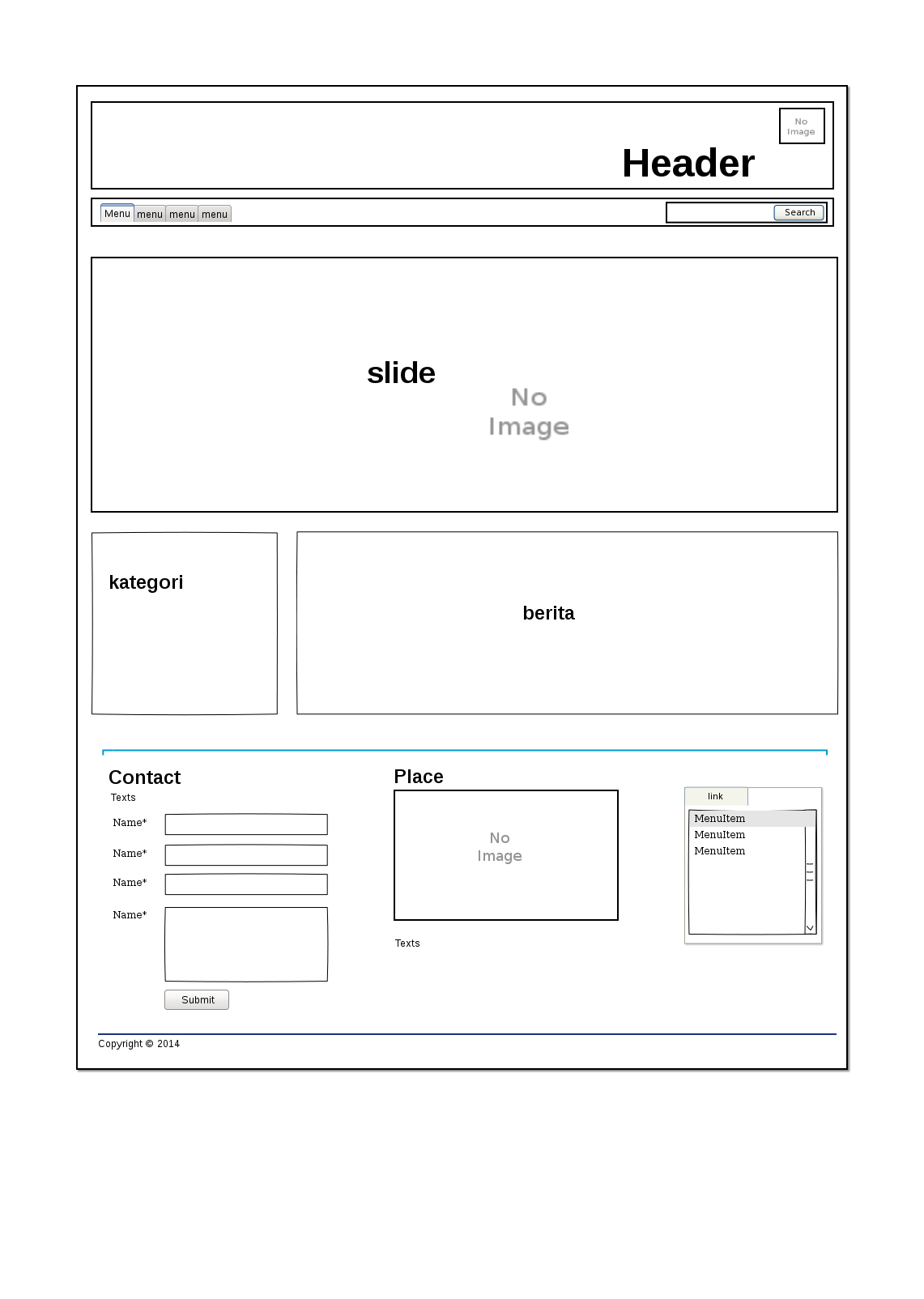
Gambar III.9 Relasi antar tabel

## Rancangan Antarmuka

Rancangan Antarmuka ini dibuat bertujuan untuk membuat desain tampilan agar bentuk program yang dibuat dapat mempermudah user saat mengoperasikan program yang sebenarnya.

### Rancangan halaman beranda

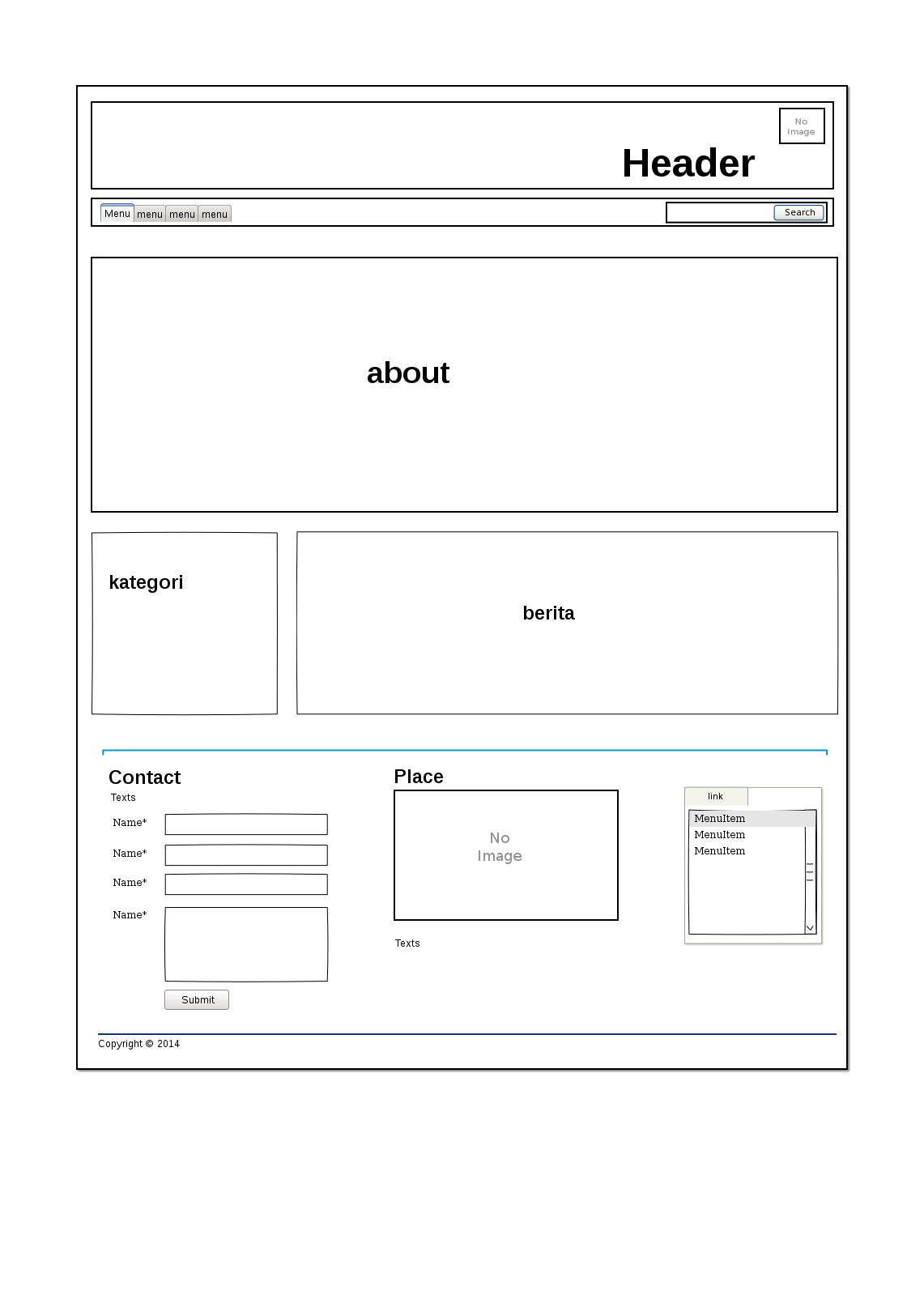
Pada rancangan tampilan halaman beranda, terdapat beberapa menu yang memberikan informasi dan pencarian lokasi pertumbuhan ekonomi yang dicari di Kabupaten Klaten. User dapat meng-klik setiap menu tersebut untuk melihat informasi-informasi yang ada dalam menu tersebut. Rancangan halaman beranda dapat dilihat pada Gambar III.10.



Gambar III.10 Tampilan rancangan halaman beranda

### Rancangan halaman tentang

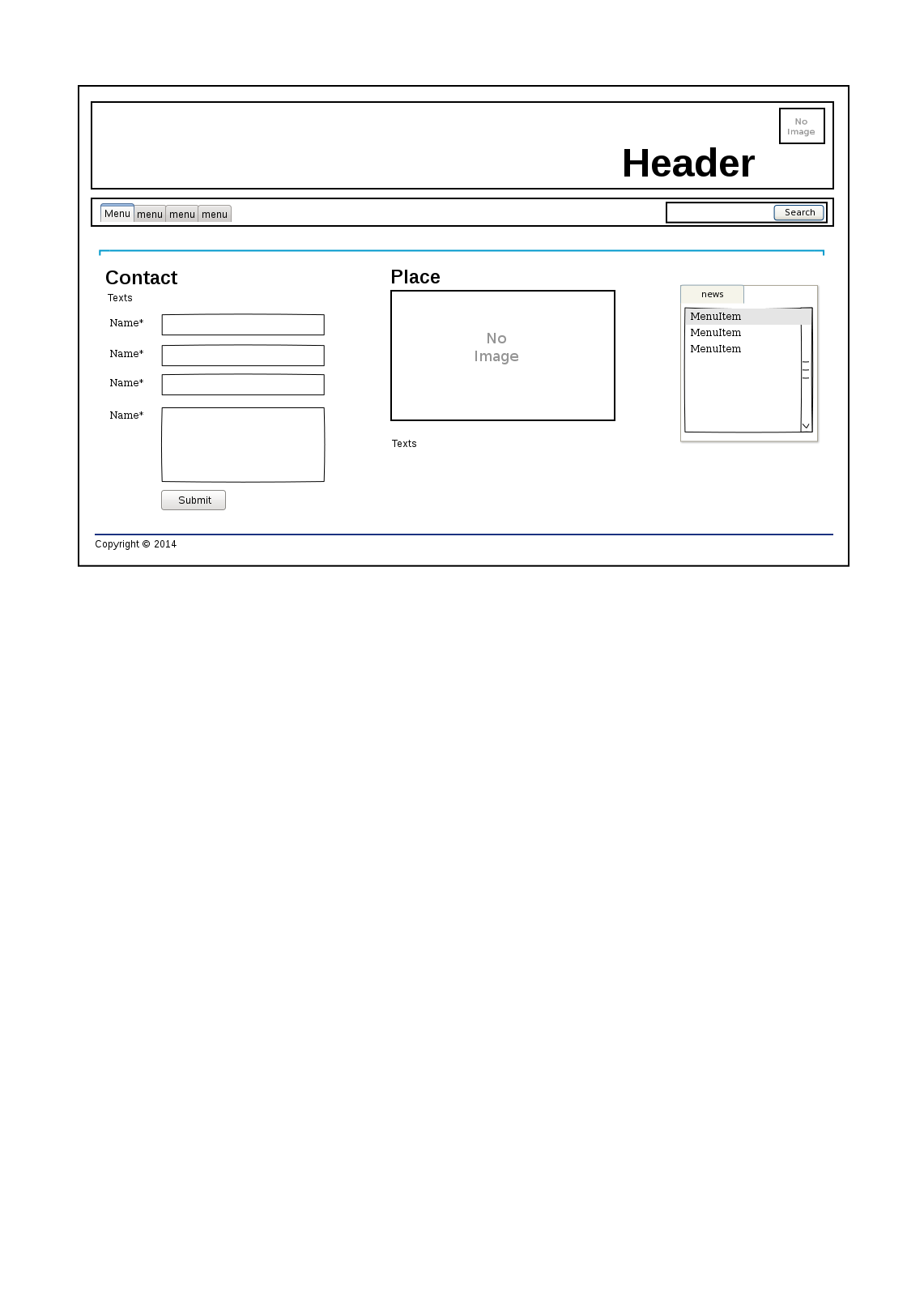
Pada rancangan tampilan halaman tentang, berisi tentang apa itu sistem informasi Badan Pusat Statistik. Rancangan halaman tentang BPS dapat dilihat pada Gambar III.11.



Gambar III.11 Tampilan rancangan halaman tentang

### Rancangan halaman kontak

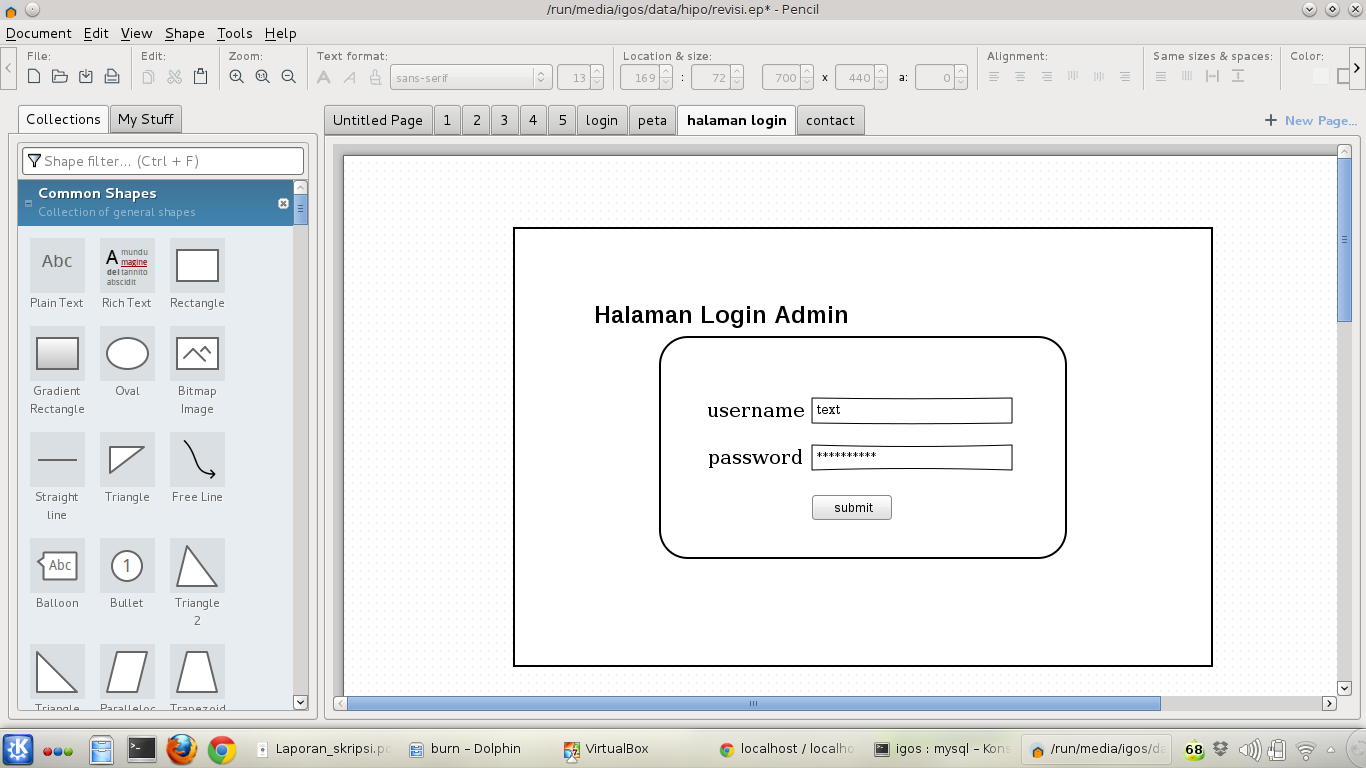
Pada rancangan tampilan halaman kontak, berisi tentang data-data yang digunakan user untuk menghubungi admin dari sistem. Rancangan halaman kontak dapat dilihat pada Gambar III.12.



Gambar III.12 Tampilan rancangan halaman kontak

### Rancangan halaman login

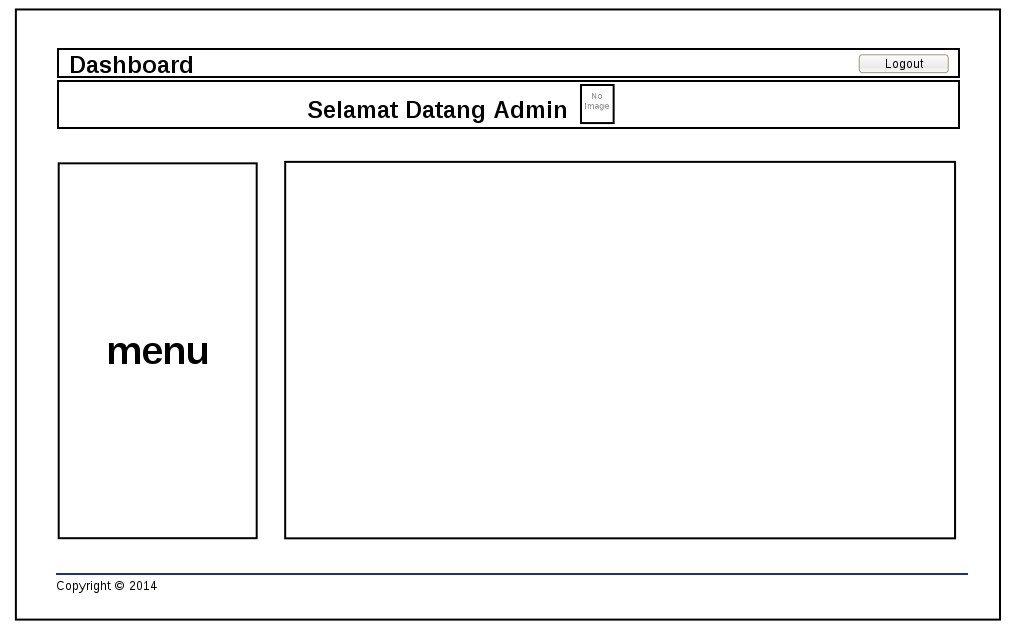
Halaman *login*  merupakan halaman yang digunakan oleh Administrator untuk memvalidasi akun sebelum memasuki sistem. Rancangan halaman login dapat dilihat pada Gambar III.13.



Gambar III.13 Tampilan rancangan halaman login

### Rancangan halaman Admin

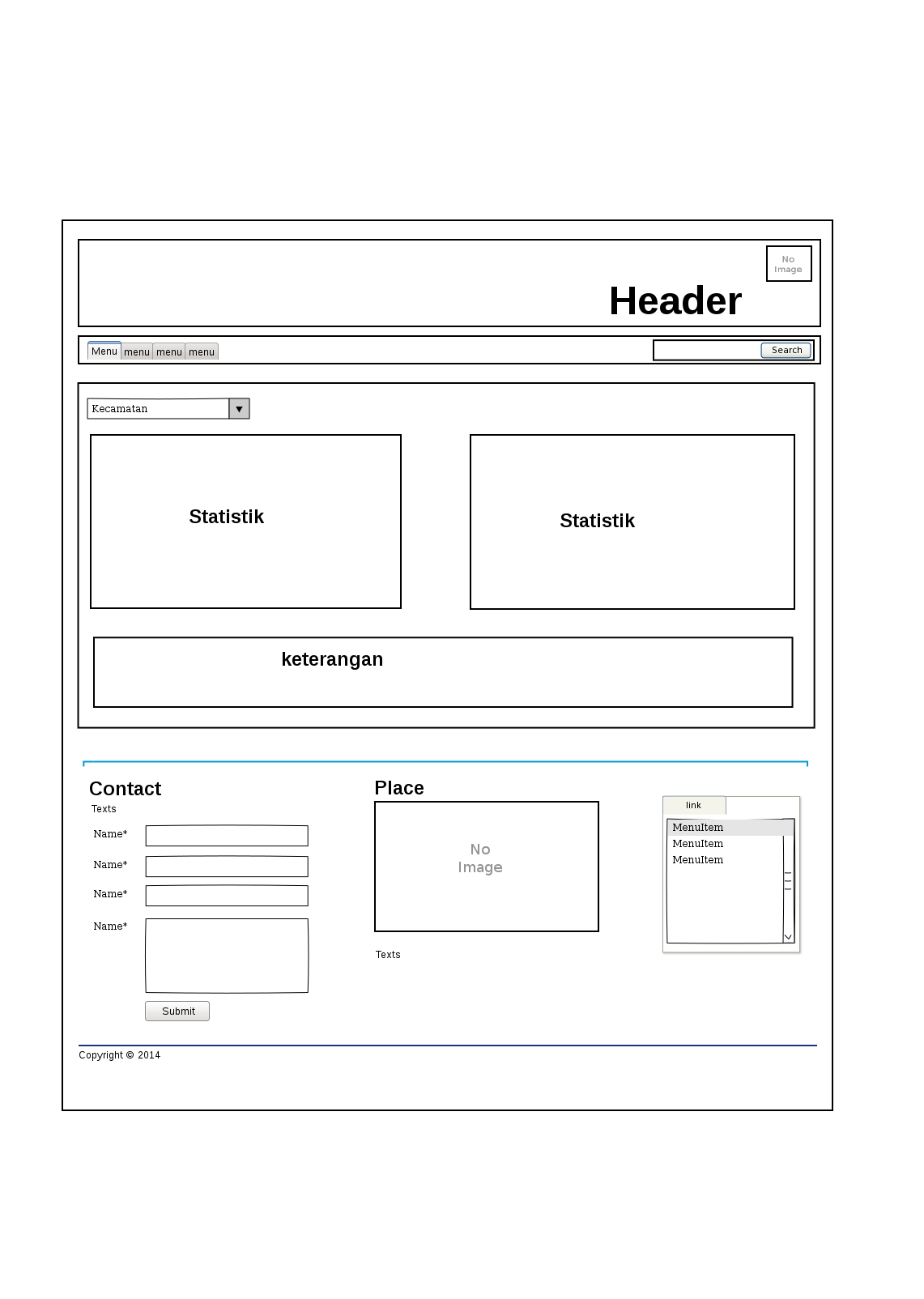
Halaman admin merupakan halaman yang digunakan oleh Administrator untuk menginputkan data, mengatur user admin,menampilkan data di halaman beranda. Rancangan halaman login dapat dilihat pada Gambar III.14.



Gambar III.14 Rancangan halaman admin

### Rancangan halaman statistik pdrb kecamatan

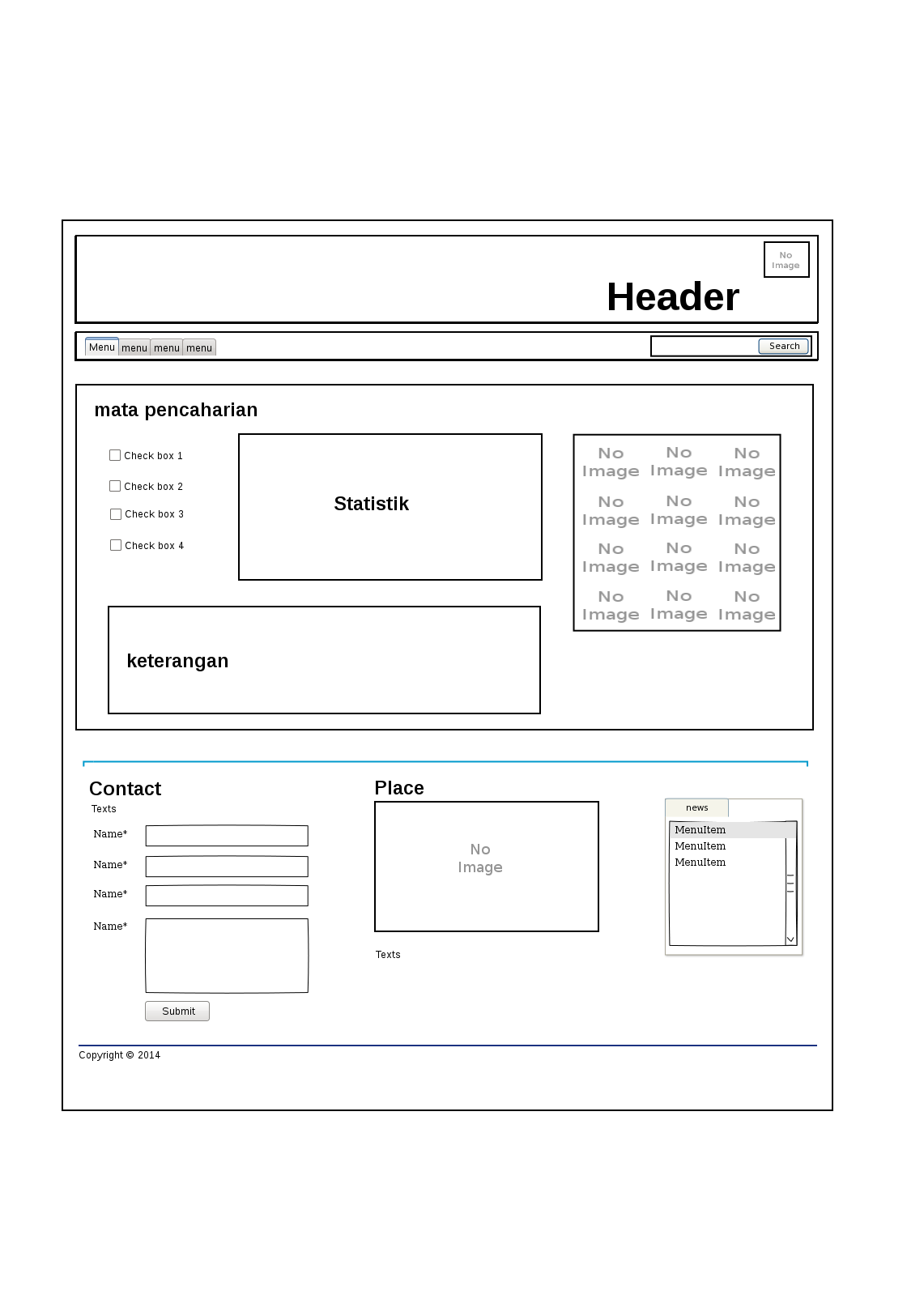
Halaman statistic pdrb kecamatan merupakan halaman yang digunakan oleh Administrator untuk menampilkan data pdrb perkecamatan dalam bentuk diagram. Rancangan halaman login dapat dilihat pada Gambar III.15.



Gambar III.15 Tampilan rancangan halaman statistic pdrb kecamatan

### Rancangan halaman statistik mata pencaharian

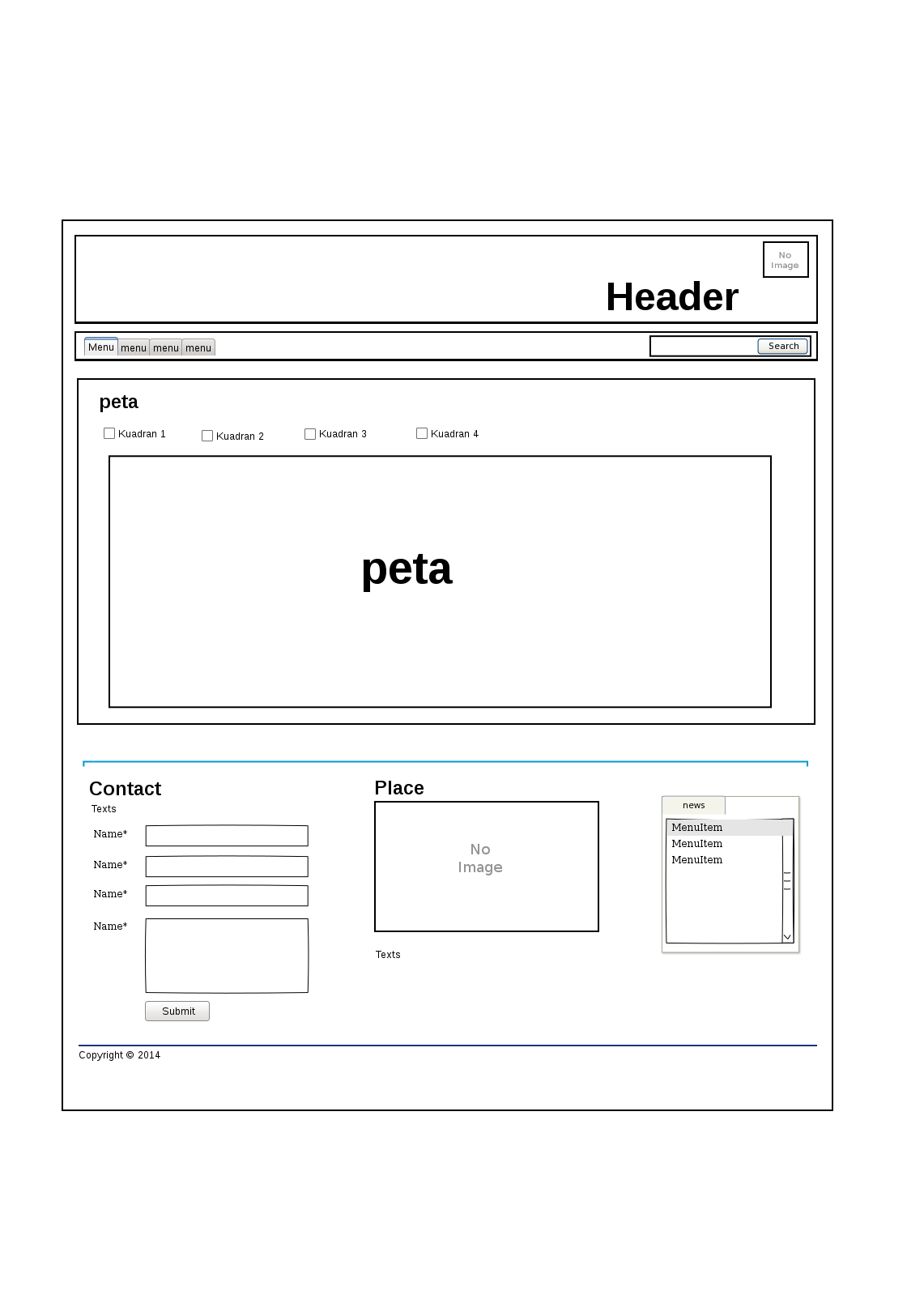
Halaman statistik mata pencaharian merupakan halaman yang digunakan oleh Administrator untuk menampilkan hasil data mata pencaharian dalam bentuk diagram. Rancangan halaman login dapat dilihat pada Gambar III.16.



Gambar III.16 Tampilan rancangan halaman statistik mata pencaharian

### Rancangan halaman peta lokasi

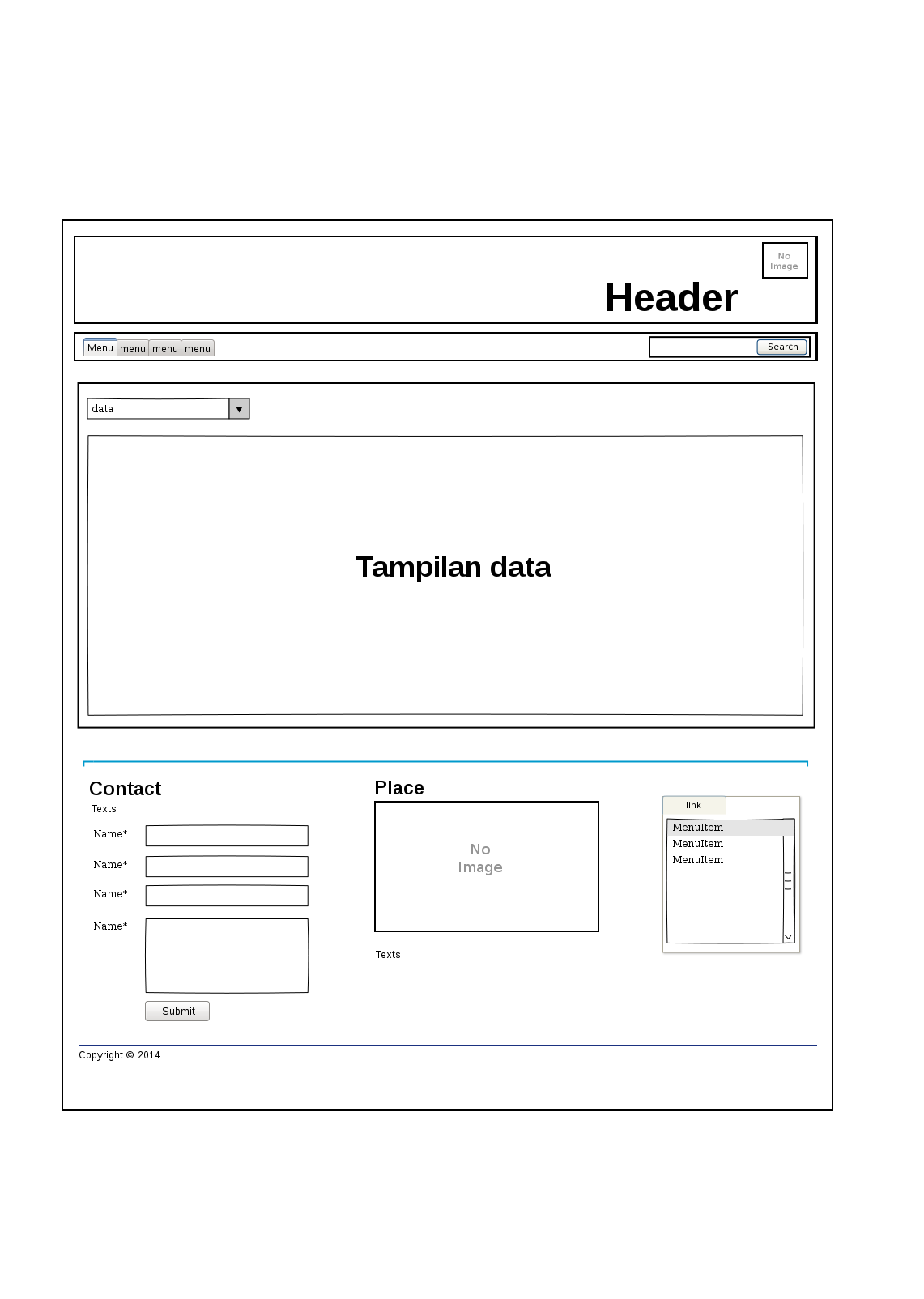
Halaman ini digunakan untuk mecari daerah kecamatan terdekat dari tempat yang berada pada kuadran tertentu. Seperti pencarian daerah kecamatan yang termasuk pada daerah kuadran I dari tempat berada saat ini sehingga memudahkan dalam pencarian berdasarkan tempat yang ditentukan berdasarkan kuadran. Rancangan halaman Peta Lokasi dilihat pada Gambar III.17.



Gambar III.17 Tampilan rancangan halaman peta lokasi

### Rancangan halaman data

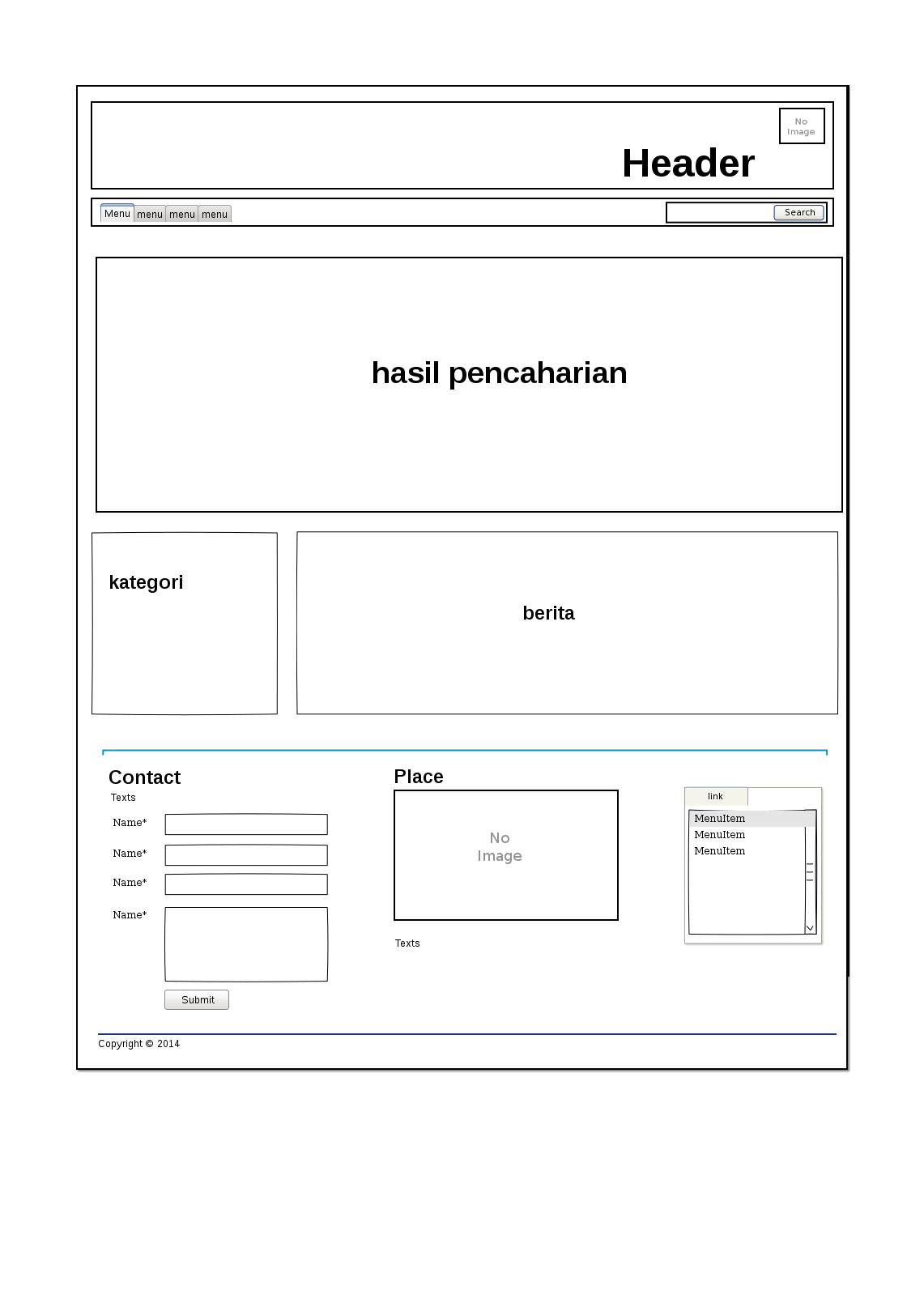
Halaman data merupakan halaman data yang telah di *input* di halaman administrator kemudian di tampilkan di halaman beranda untuk melihat data ini tanpa harus memasuki halaman admin. Rancangan halaman data dapat dilihat pada Gambar III.13.



Gambar III.18 Tampilan halaman data

### Rancangan halaman pencaharian data

Halaman pencaharian data merupakan halaman untuk pencaharian data oleh user untuk mendapatkan data yang di cari. Rancangan halaman pencaharian data dapat dilihat pada Gambar III.13.



Gambar III. 19 Rancangan halaman pencaharian data

# BAB IV

# HASIL DAN PEMBAHASAN

## Pelaksanaan Implementasi Sistem

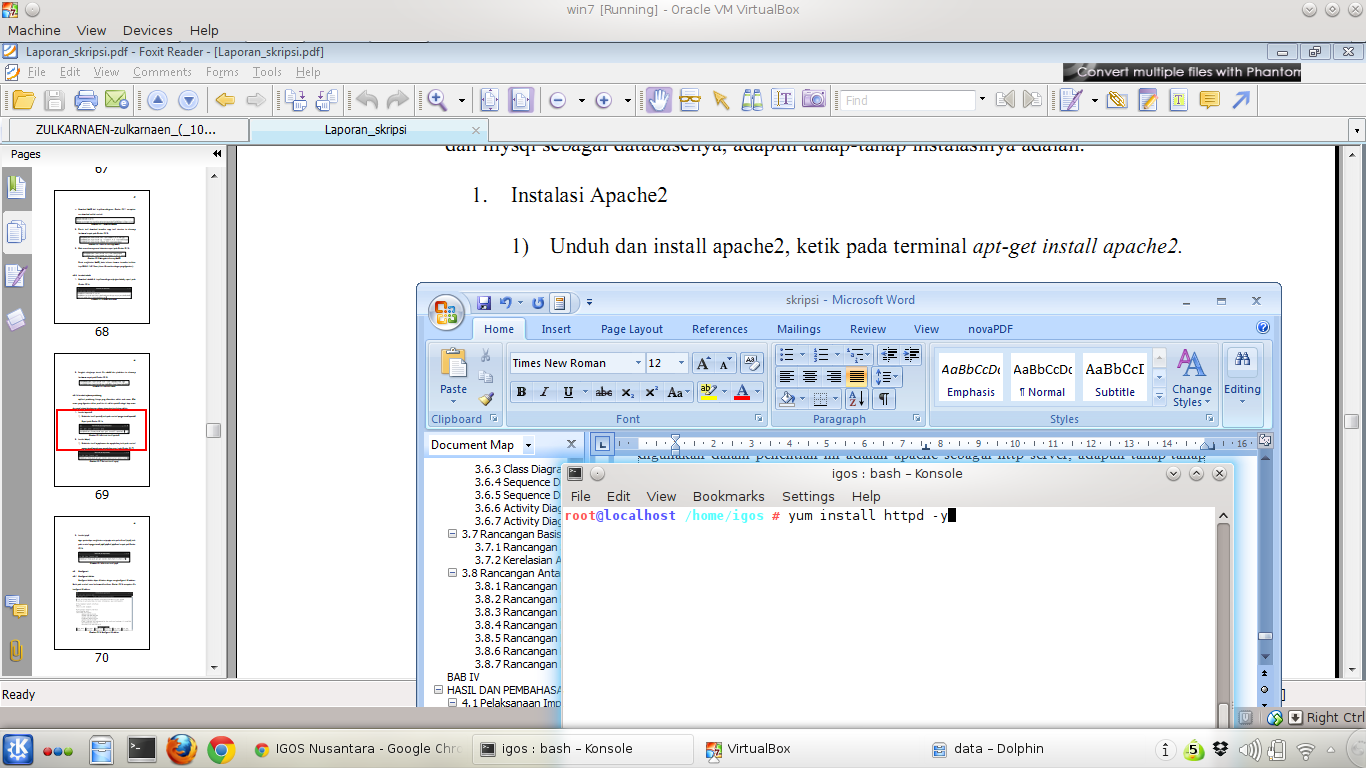
Setelah sistem analisa dirancang dan didesain secara rinci maka sistem diimplementasikan. Tujuannya adalah menerapkan sistem yang telah dirancang untuk diterapkan pada keadaan sebenarnya dan dievaluasi apakah menghasilkan sistem yang baik atau sebaliknya. Dikatakan akan menghasilkan sistem yang baik jika sistem yang digunakan dapat mencapai tujuan dengan prinsip efisien dan efektif. Dikatakan sistem tidak berhasil jika sistem tersebut, ternyata banyak kelemahan dan kekurangan sehingga tidak efisien.

### Instalasi *Software* Pendukung

Aplikasi pendukung lainnya yang dibutuhkan adalah *web server*. *Web server* yang digunakan dalam penelitian ini adalah apache2 sebagai http server, adapun tahap-tahap instalasinya adalah :

1. Instalasi Apache2

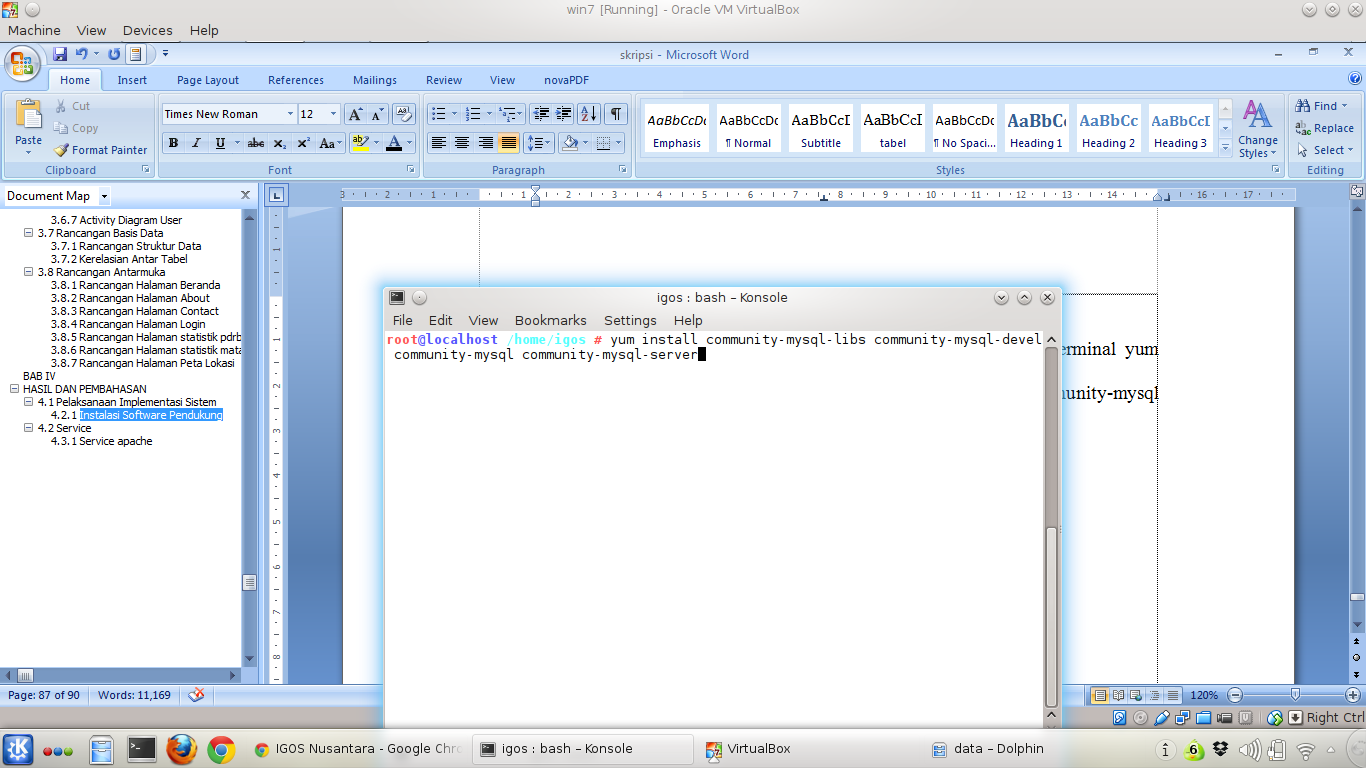
Unduh dan install apache2, ketik pada terminal yum install httpd –y. Seperti pada Gambar IV.1.



Gambar IV.1 perintah install apache2

1. Instalasi Mysql

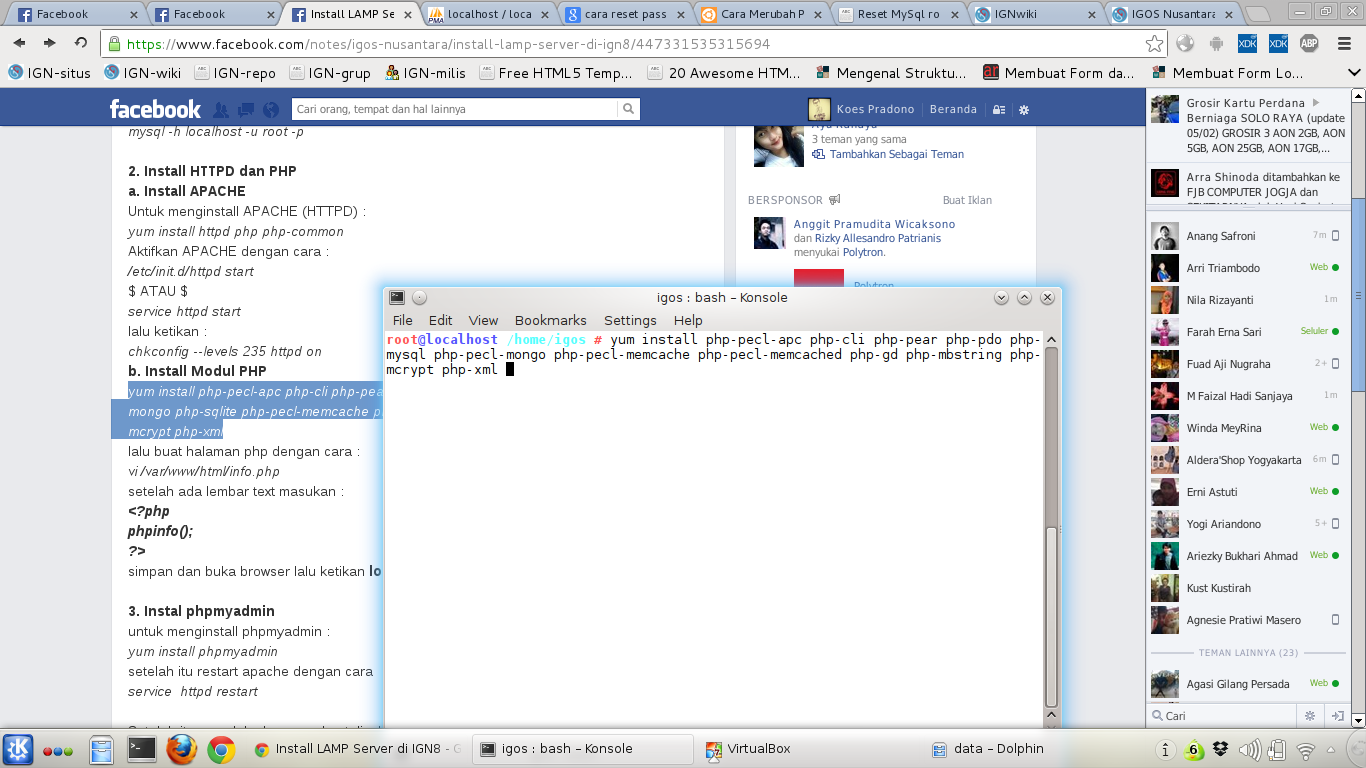
Unduh dan install *mysql*, ketik pada terminal yum install community-mysql-libs community-mysql-devel community-mysql community-mysql-server. Seperti Gambar IV.2.



Gambar IV.2 perintah install mysql

1. Instalasi php5

Agar apache2 dapat menjalankan *script* *php* maka perlu diinstall php5, ketik pada terminal yum install php-pecl-apc php-cli php-pear php-pdo php-mysql php-pecl-mongo php-pecl-memcache php-pecl-memcached php-gd php-mbstring php-mcrypt php-xml tampak pada Gambar IV.3.



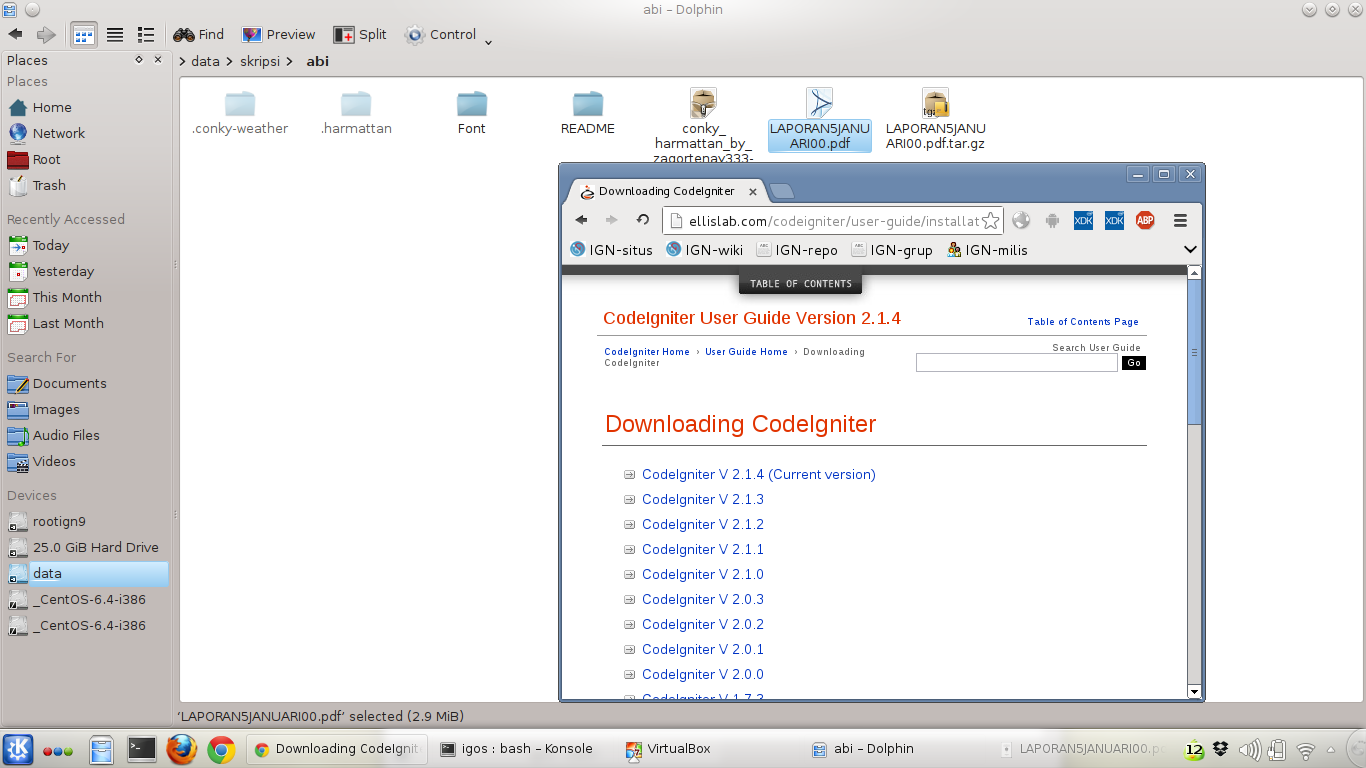
Gambar IV.3 Perintah Instalasi php5

### Instalasi *CodeIgniter*

*Codeigniter* merupakan Framework yang dipakai dalam pembuatan sistem ini, untuk menggunakannya download terlebih dahulu *Codeigniter* :

http://ellislab.com/codeigniter/user-guide/installation/downloads.html

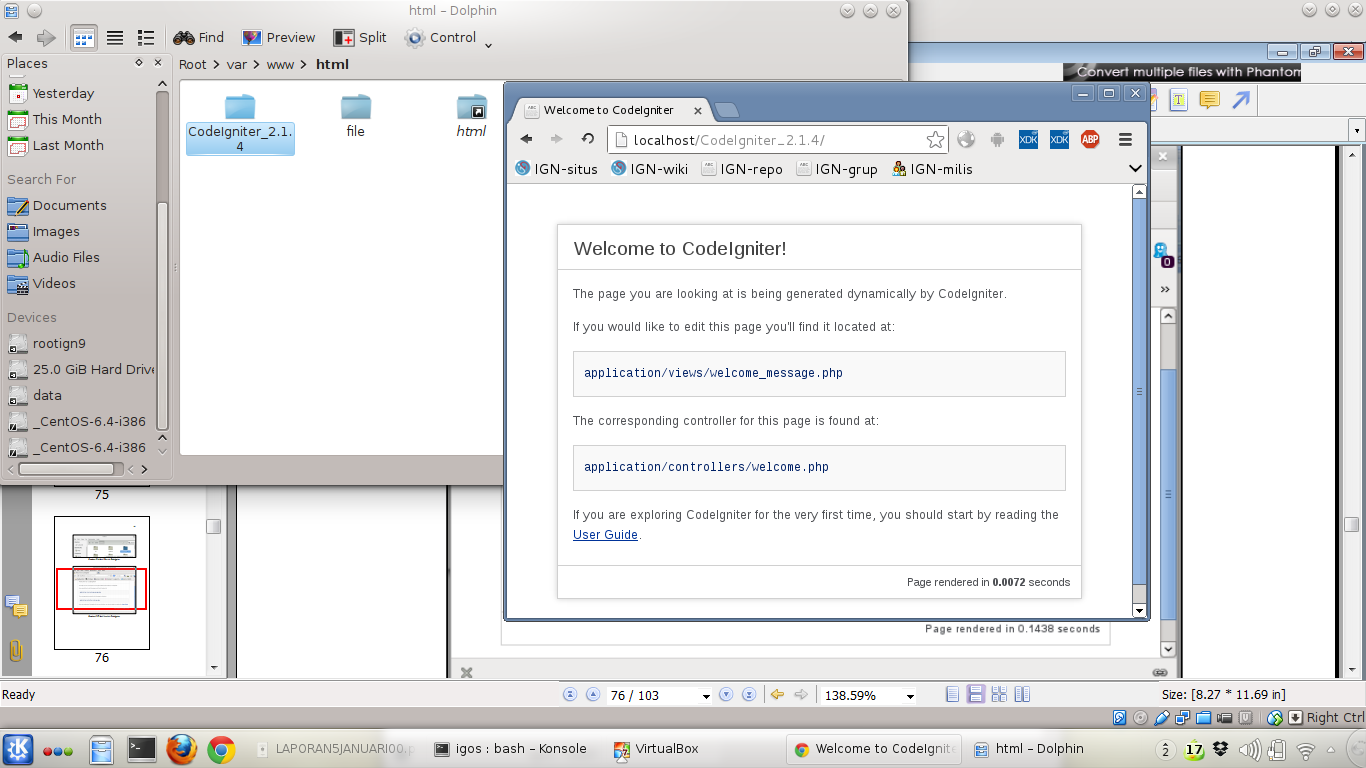
Download *Codeigniter* dengan versi terakhir dirilis yaitu versi 2.1.4 seperti pada Gambar IV.4 . Setelah proses *download* selesai kemudian *install* dengan cara mengekstrak ke dalam direktori /var/www/html/. Hasil *extract*  dapat dilihat pada Gambar IV.5 dan hasil instalasi *Codeigniter* dapat dilihat pada *browser* seperti pada gambar IV.6.



Gambar IV.4 *Download CodeIgniter*



Gambar IV.5 Hasil *Extract CodeIgniter*



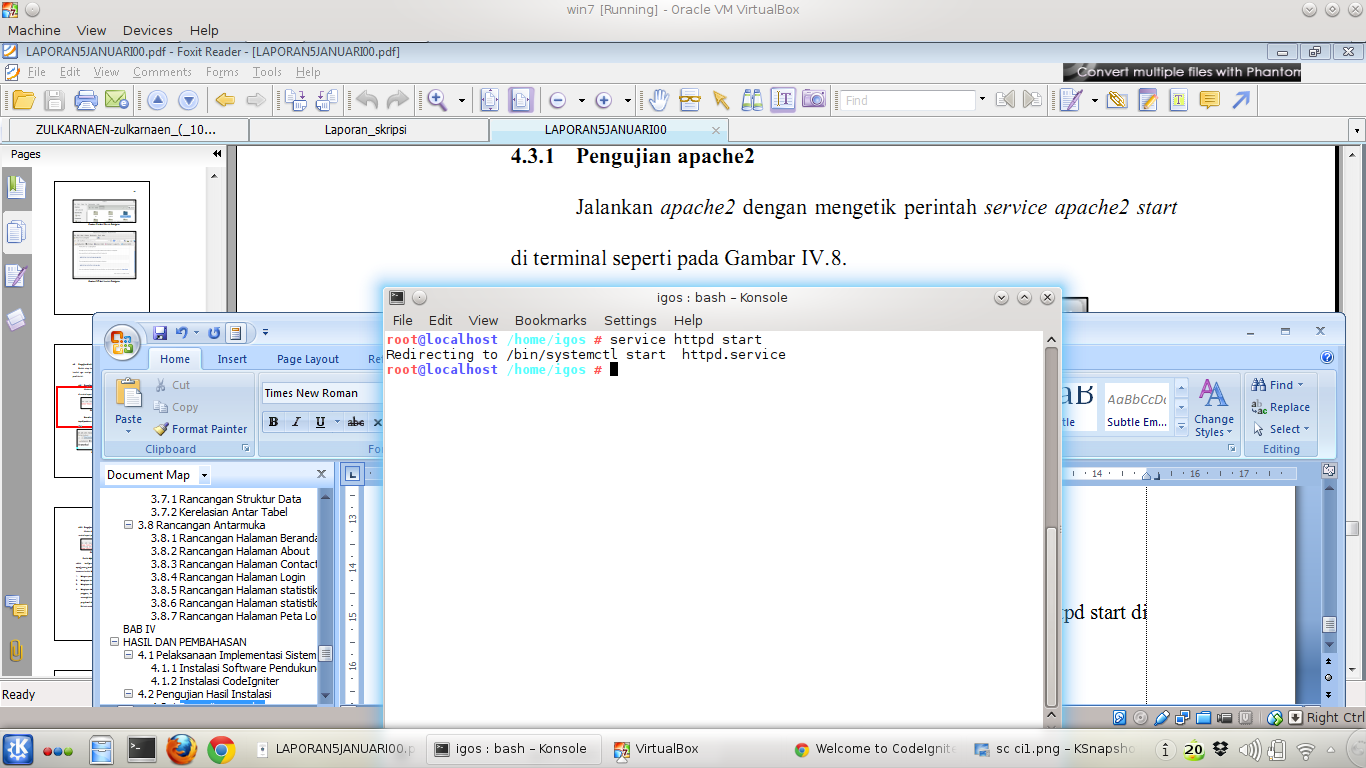
Gambar IV.6 Hasil Instalasi *CodeIgniter*

## Pengujian Hasil Instalasi

Setelah tahap instalasi selesai, selanjutnya adalah tahap uji coba hasil instalasi agar nantinya tidak ada kendala dalam menjalankan sistem hasil penelitian ini.

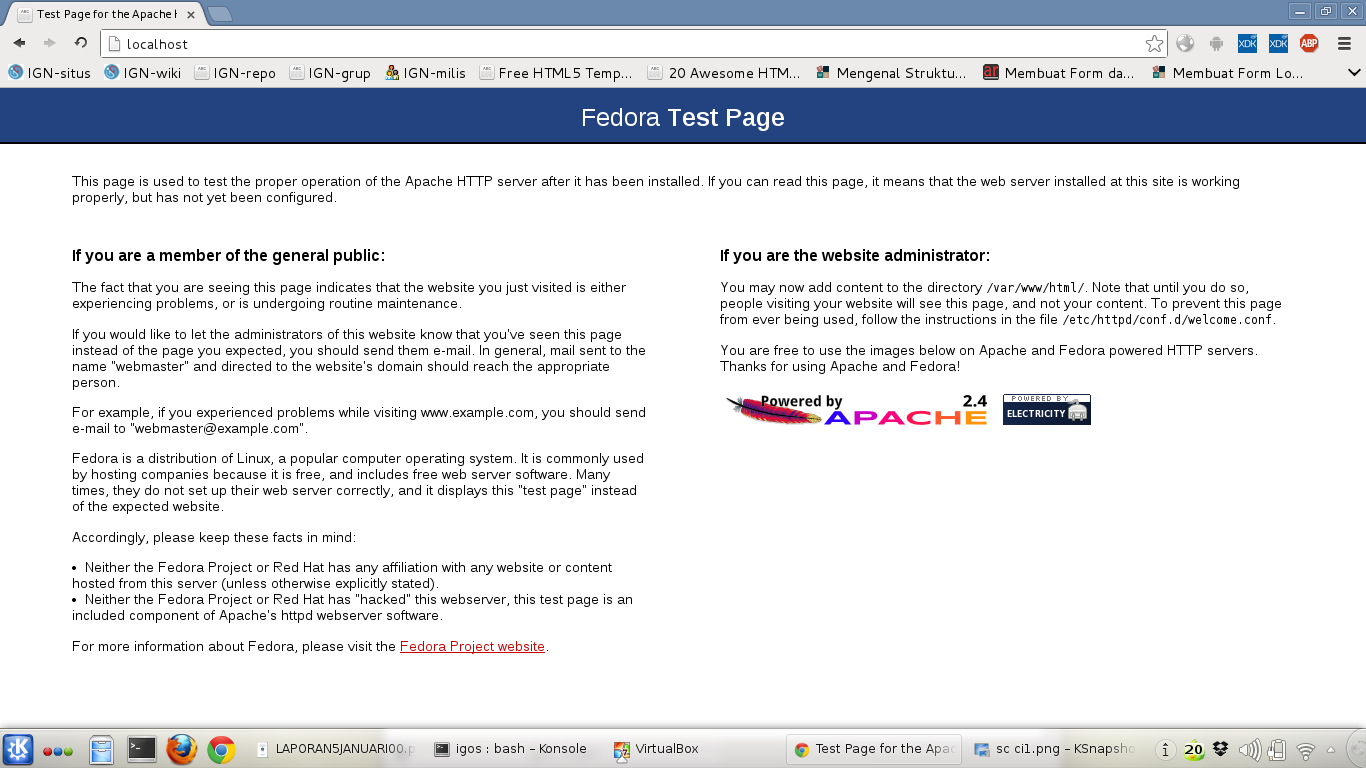
### Pengujian apache2

Untuk menjalankan service apache2 dengan cara mengetik perintah service httpd start di terminal seperti pada Gambar IV.7.



Gambar IV.7 *Service* apache2

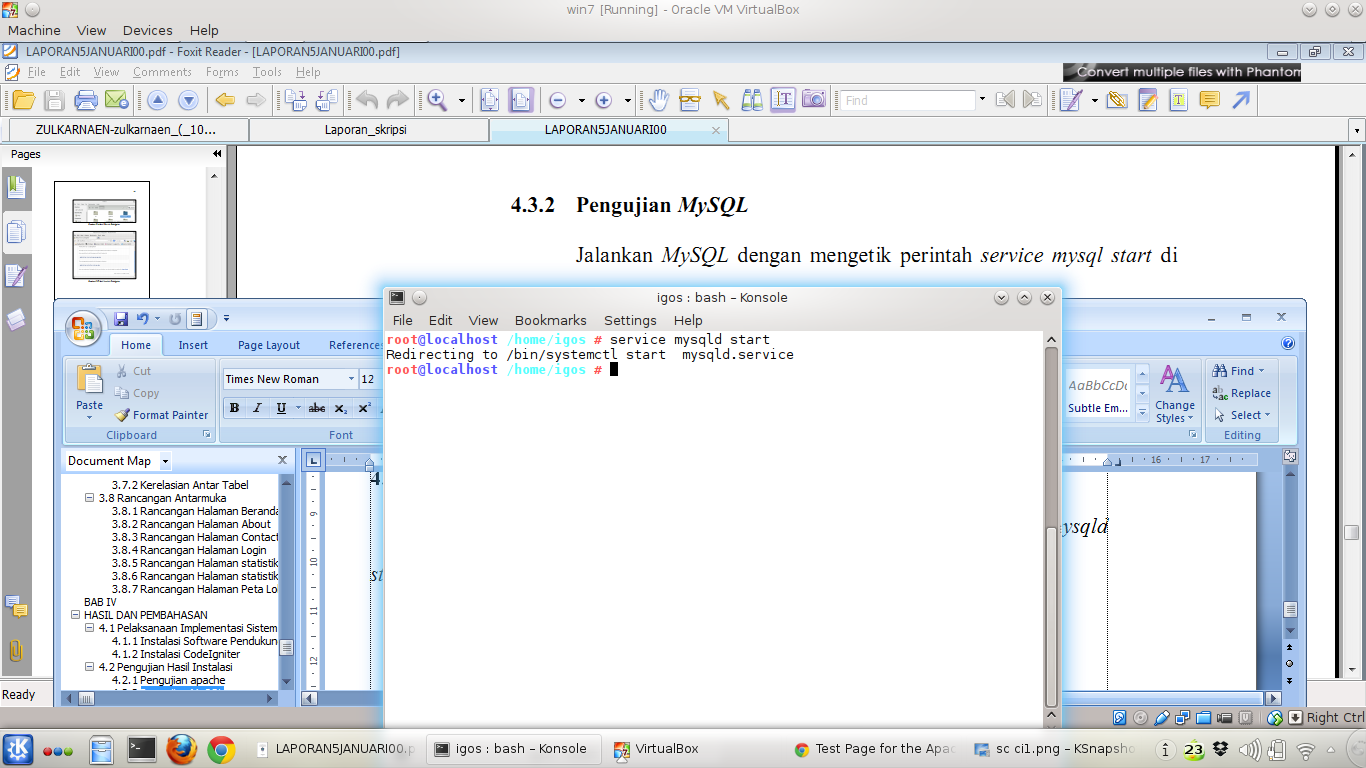
Kemudian buka *web* browser dan ketikan http://localhost atau http:ip-address dan jika berhasil maka akan terlihat seperti pada Gambar IV.8.



Gambar IV.8 Pengujian apache2

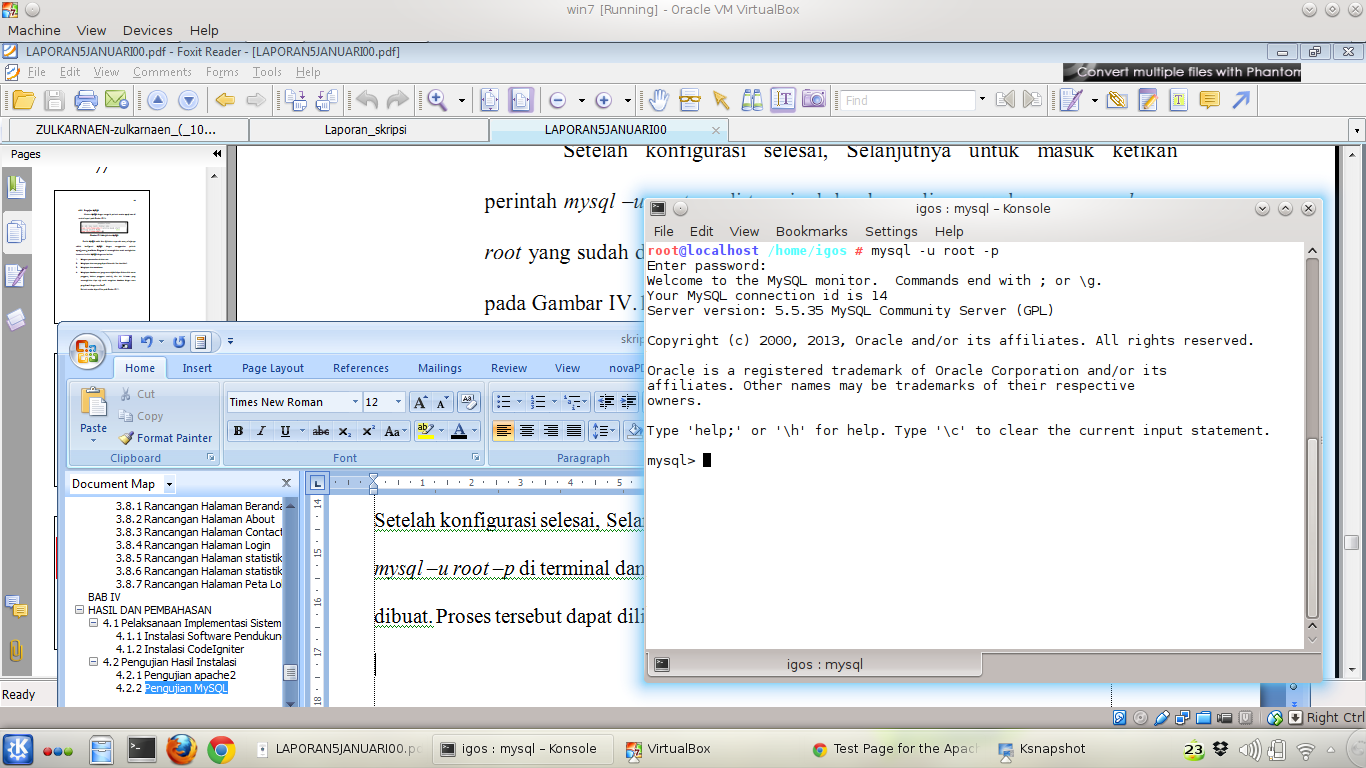
### Pengujian *MySQL*

Untuk menjalankan *service MySQL* dengan cara mengetik perintah *service mysqld start* di terminal seperti pada Gambar IV.9.



Gambar IV.9 Pengujian *MySQL*

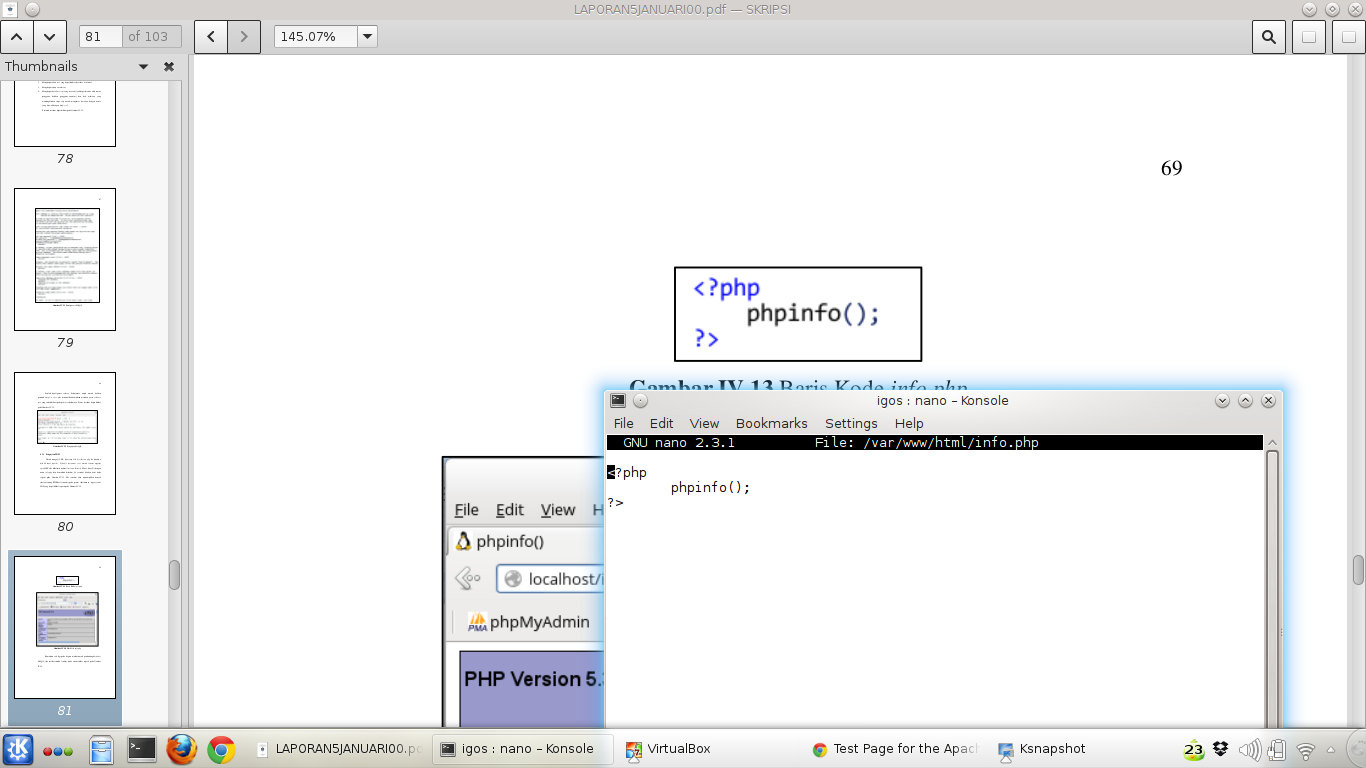
Setelah konfigurasi selesai, Selanjutnya untuk masuk ke dalam MySQL dengan perintah *mysql –u root –p* di terminal dan kemudian masukan *password user root* yang sudah dibuat. Proses tersebut dapat dilihat pada gambar IV.10.



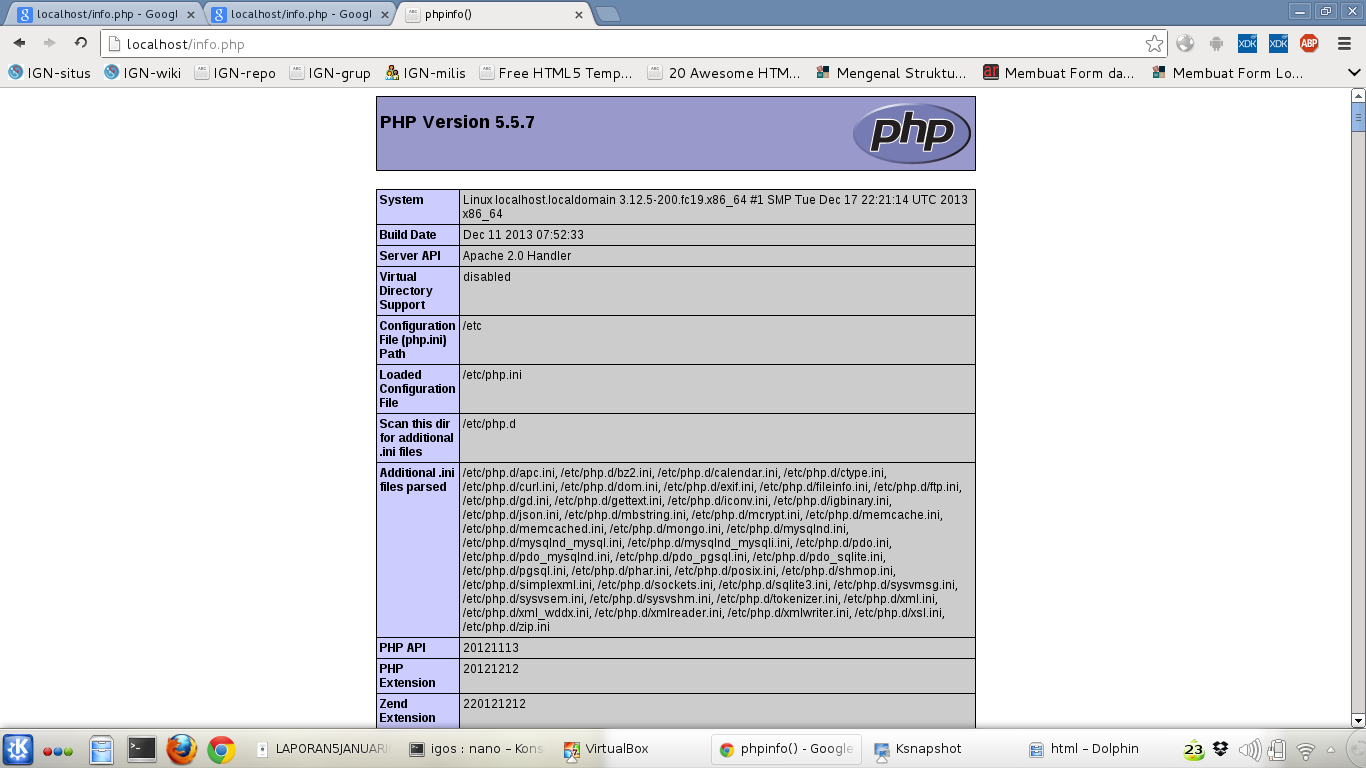
Gambar IV.10 Menjalankan *MySQL*

### Pengujian PHP5

Untuk menguji PHP, buat satu file berekstensi php ke direktori default dari Apache. Default document root untuk sistem operasi IGOS Nusantara ada didalam direktori /var/www/html/. Misal, buat file dengan nama info.php dan kemudian didalam file tersebut ketikan baris kode seperti pada Gambar IV.11. File tersebut akan menampilkan banyak rincian tentang PHP hasil instalasi pada proses sebelumnya, seperti versi PHP yang dapat dilihat seperti pada Gambar IV.12.

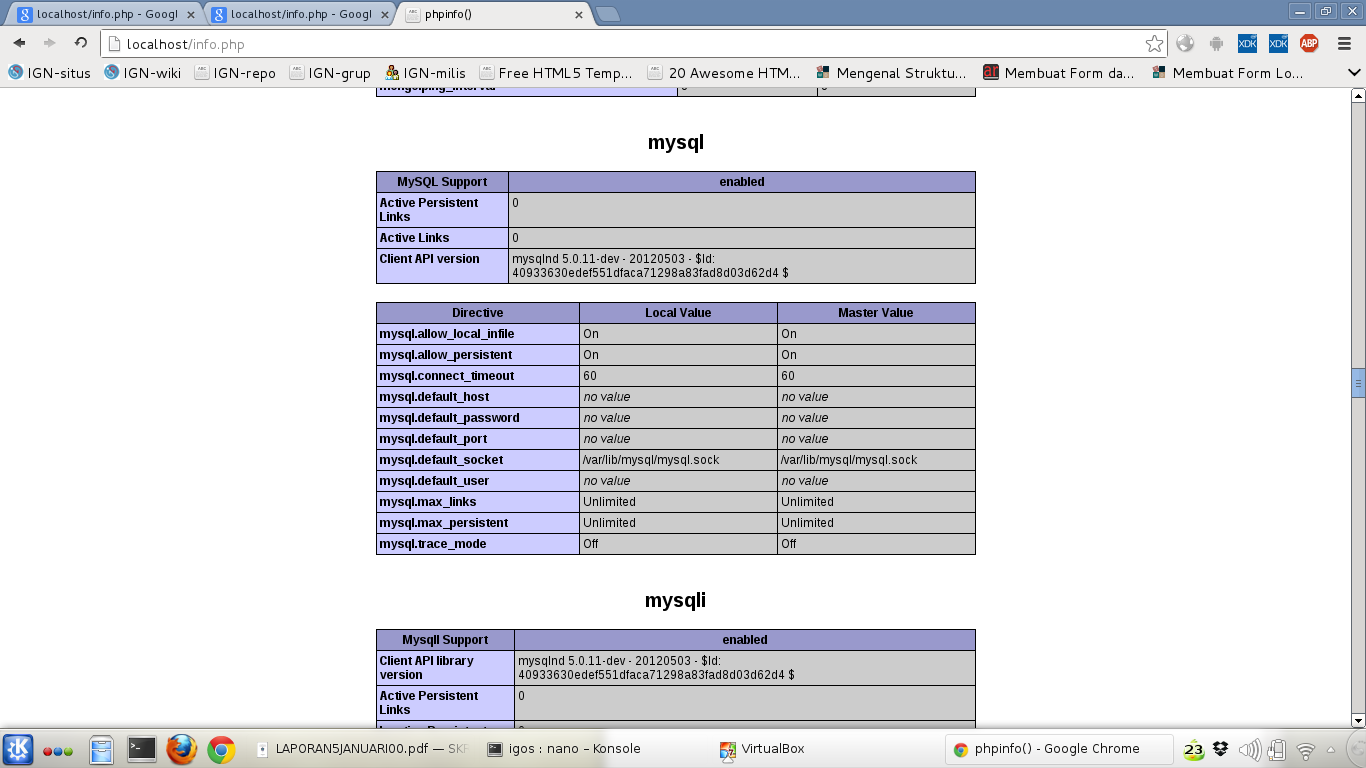


Gambar IV.11 Baris Kode *info.php*



Gambar IV.12 Hasil *file* *info.php*

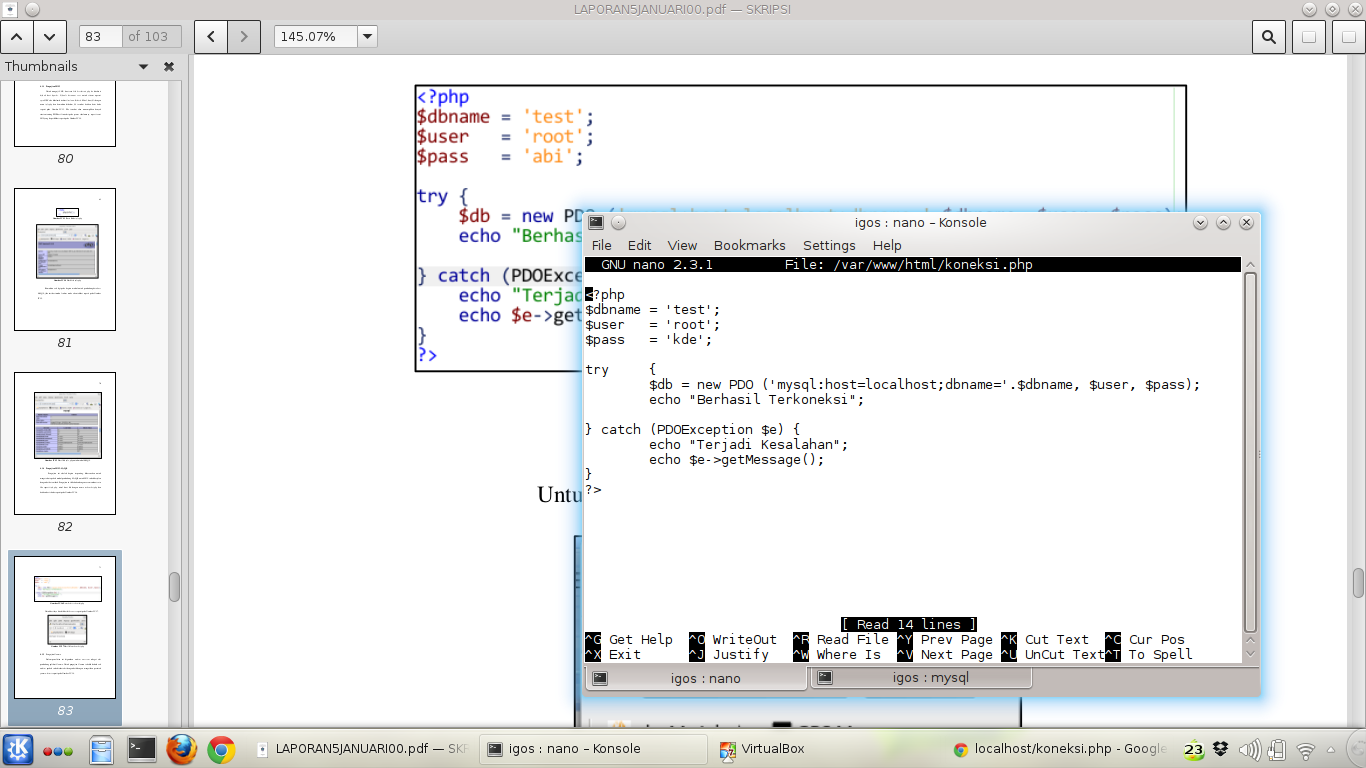
Kemudian cek lagi pada bagian modul untuk pendukung *database MySQL*, jika instalasi modul sukses maka akan terlihat seperti pada Gambar IV.13.



Gambar IV.13 Hasil *file info.php* untuk modul *MySQL*

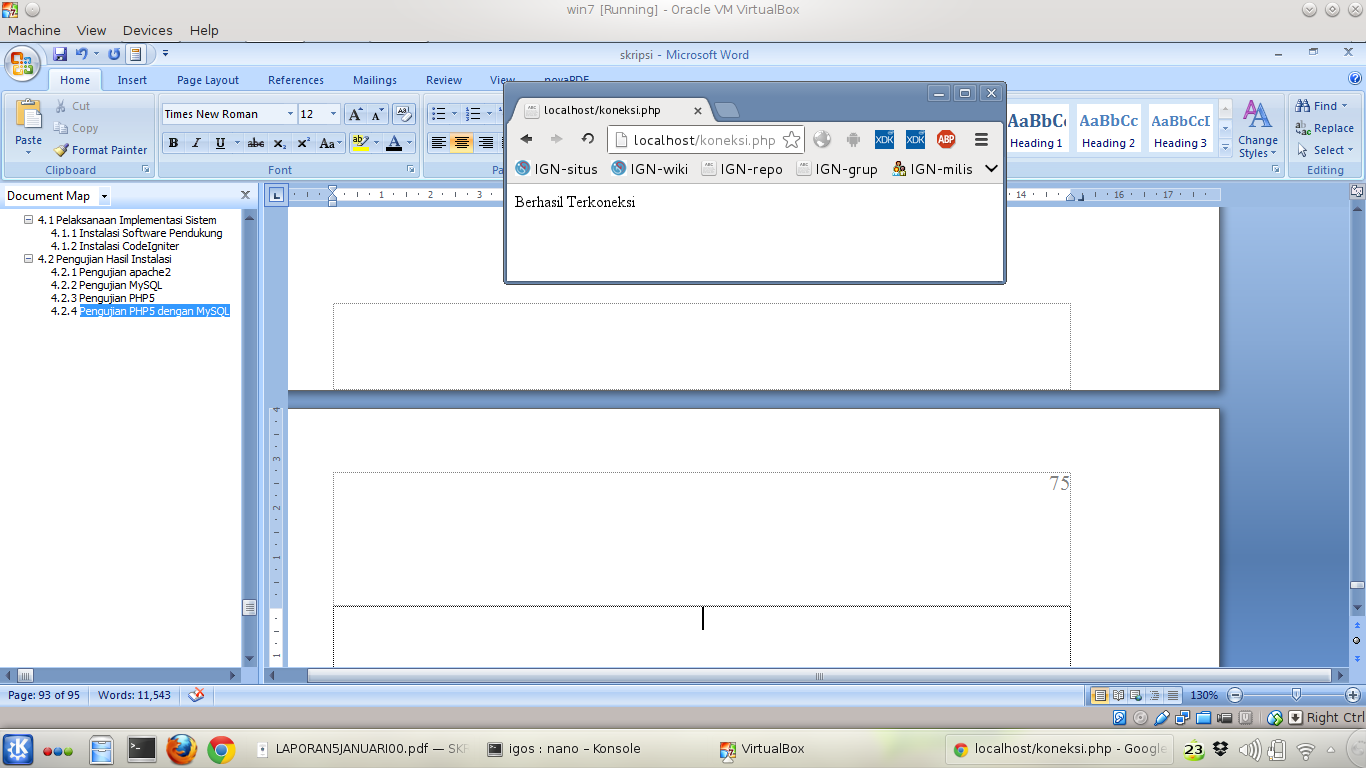
### Pengujian PHP5 dengan MySQL

Pengujian ini adalah bagian terpenting, dikarenakan untuk mengetahui apakah modul pendukung MySQL untuk PHP sudah berjalan dengan baik atau tidak. Pengujian ini dilakukan dengan cara membuat satu file seperti info.php, misal buat file dengan nama koneksi.php dan ketikan baris kode seperti pada Gambar IV.14.



Gambar IV.14 Baris kode koneksi.php

Untuk hasilnya bisa dilihat di browser seperti pada Gambar IV.15.



Gambar IV. 15 Hasil file terkoneksi.php

# DAFTAR PUSTAKA

Alhadi, A., (2013), *Sistem Informasi Penggajian dan Pengupahan Karyawan Menggunakan Arsitektur Hierarchical-Model-View-Control Framework Php Codeigniter,* Skripsi Sarjana, Universitas Islam Negri Sunan Kalijaga, Yogyakarta.

Ashary, F., (2013), *Pengertian Uml Unified Modeling*. http://fadhlyashary.blogspot.com/2012/06/pengertian-uml-unified- modeling.html, Diakses 7 September 2013.

Boediono., (1999), *Ekonomi Moneter: seri sinopsis Pengantar Ilmu Ekonomi no.5 BPFE.* Yogyakarta.

Google Inc.,(2013), *Tampilan Google Map*, http://maps.google.com, Diakses 24 Desember 2013

Jon., (2013), *Tampilan Default Framework CodeIgniter*. http://wiki.bitnami.com/@api/deki/files/290/=codeigniter.png, Diakses 1 Februari 2014.

Kadir, A., (2008), *Belajar Database Mengunakan MySQL,* Andi Offset, Yogyakarta.

Kadir, A., (2003), *Pemograman Web dan PHP,* Andi Offset, Yogyakarta.

Kadir, A., (2001), *Pemrograman WEB Mencakup: HTML, CSS, JAVA SCRIPT, dan PHP,* Andi Offset, Yogyakarta.

Komputer, I., (2007), *Web Scripting Dengan AJAX,* Ilmu Komputer, Jakarta.

Kristanto, A., (2008), *Perancangan Sistem Informasi Dan Aplikasinya.* Gava Media, Yogyakarta.

Kuncoro, M., (2004), *Otonomi dan Pembangunan Daerah,* Erlangga, Jakarta.

Prahasta, E., (2002), *System informasi Geografi tutorial ArcView,* Informatika, Bandung.

Pratono, A., (2009), *Pertumbuhan Ekonomi dan Ketimpangan Antar Kecamatan di Kabupaten Klaten (Tahun 2003-2007),* Skripsi Sarjana, Universitas Islam Indonesia Yogyakarta.

Ribkhan., 2012, *Konsep MVC Framework*. http://buggzilla.files.wordpress.com/2012/05/basic\_mvc.png?w=812, Diakses 13 Februari 2013

Rizal, L., (2013), *Pengertian Internet dan Pengertian www* http://www.dq.blogspot.com/2011/04/pengertian-*internet*-dan-pengertian- www.html, Diakses 15 juni 2013.

Saputra, A., (2008), *Database Server Mysql,* Elex Media Komputindo, Yogyakarta.

Saputra, A., (2011), *Slide show Dengan JQuery,* Elex Media Komputindo, Yogyakarta.

Shalahudin, M., (2008), *Java Di WEB,* Informatika, Bandung.

Shalahudin, M., (2011), *Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek), Informatika,* Bandung.

Sidik, B., (2012), *Framework Codeigniter,* Informatika Bandung.

Sjafrizal., (1997), *Pertumbuhan Ekonomi dan Ketimpangan Regional Wilayah Indonesia Bagian Barat,* Prisma Padang.

Sjafrizal., (2009), *Teknik Praktis Penyusunan Rencana Pembangunan Daerah,* Baduose Media Padang.

Sugiri, A. S., (2007), *Desain Web Menggunakan HTML Dan CSS,* Andi Offset Yogyakarta.

Sukirno, S., (2006), *Ekonomi Pembangunan : Proses, Masalah, dan Dasar Kebijakan. Edisi kedua,* Kencana Jakarta.

Sukirno, S., (1994), *Pengantar Teori Makro Ekonomi,* PT. Raja Grafindo Persada Jakarta.

Susrini, N., (2009), *Google : Mesin Pencari yang Ditakuti Raksasa Microsoft,* BFIRST, Yogyakarta.

Tanamah, R., (2008), *Perancangan dan Implementasi Web GIS Parawisata,* Skripsi Sarjana, IST AKPRIND, Yogyakarta.

Utami, E., (2006), *RDMS Using MS SQL Server 2000,* NRAR.NET Publisher Yogyakarta.

Wiradisastra, B. D., (2000), *Sistem Informasi Geografi,* ISBN.