

SISTEM INFORMASI PENGOLAHAN DATA RUMAH KOS MENGGUNAKAN FRAMEWORK CODEIGNITER DI KELURAHAN DEMANGAN KECAMATAN GONDOKUSUMAN YOGYAKARTA

Guridno Adi Saputro¹, Amir Hamzah², Joko Triyono³

^{1, 2, 3} Teknik Informatika, FTI, IST AKPRIND

¹ abbyacakadud@gmail.com, ² amir@akprind.ac.id, ³ zain.jack@gmail.com

ABSTRACT

Information systems is that information technology is growing so rapidly. Technologies that can help more easily, quickly, safely and effectively clear it is helpful to anyone who use it. Including to assist students in the boarding house search. While conducting a study / lecture will need a boarding house is important, since the students who undertake studies in a long time and needed a place to stay in the area. And it makes the issue of how students can find suitable boarding house, given some constraints such as do not know the area, or do not have friends who can demonstrate appropriate boarding house for students. This information systems which will facilitate for students who feel confused in finding suitable boarding houses and rooming will assist owners in his boarding house online marketing. The system is focused in finding the appropriate criteria to boarding reservations for a dorm room can be done with this system. Success in implementing this system is that it can work well to help students in the search for an appropriate location rooming house. Of search, directions, up to ordering online and the response of the system facilities using SMS (Short Message Service) to the owner of the boarding house that facilitate the ordering process.

Keywords: Information System, Boarding House, GIS, SMS Gateway

INTISARI

Sistem informasi merupakan teknologi informasi yang saat ini berkembang begitu pesat. Teknologi yang dapat membantu lebih mudah, cepat, aman dan efektif jelas sangatlah membantu siapa saja yang memanfaatkannya. Termasuk didalamnya untuk membantu mahasiswa dalam pencarian rumah kos. Disaat sedang melakukan studi/kuliah kebutuhan akan rumah kos sangatlah penting, mengingat para mahasiswa yang melakukan studi dalam waktu yang cukup lama dan membutuhkan tempat tinggal di area tersebut. Dan itu membuat masalah bagaimana mahasiswa tersebut dapat mencari rumah kos yang sesuai, mengingat beberapa kendala seperti tidak tahu daerah tersebut, atau tidak memiliki teman yang dapat menunjukkan rumah kos yang sesuai untuk mahasiswa tersebut. Sistem informasi ini yang nantinya akan mewedahi bagi para mahasiswa yang merasa kebingungan dalam mencari rumah kos yang sesuai serta akan membantu pemilik kos dalam memasarkan rumah kosnya secara online. Sistem yang difokuskan dalam mencari kriteria kos yang sesuai sampai dengan pemesanan untuk kamar kos tersebut dapat dilakukan dengan sistem ini. Keberhasilan dari implementasi sistem ini adalah dapat berjalan dengan baik untuk membantu para mahasiswa didalam mencari lokasi rumah kos yang sesuai. Dari pencarian, penunjuk arah, sampai dengan pemesanan secara *online* dan fasilitas respon dari sistem menggunakan SMS (*Short Message Service*) bagi pemilik kos yang memudahkan dalam proses pemesanan.

Kata kunci: Sistem Informasi, Rumah Kos, GIS, SMS Gateway.

PENDAHULUAN

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*.

Informasi tentunya bermacam-macam, salah satunya adalah informasi tentang rumah kos untuk para mahasiswa yang akan menjalani studi/kuliah ditempat yang jauh dari

tempat tinggal. Untuk itu jelas pada saat tersebut membuat mahasiswa tersebut harus segera mencari tempat tinggal sementara untuk dijadikan domisili mereka sampai studi selesai. Kebutuhan tersebut yang saat ini masih belum banyak digunakan di berbagai daerah di Kota Yogyakarta.

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*.

Penelitian ini bertujuan untuk mengimplementasikan sebuah sistem informasi dengan menggunakan *framework codeigniter* sebagai basis program, *Google Maps* sebagai penunjuk lokasi dan *SMS Gateway* sebagai media notifikasi.

TINJAUAN PUSTAKA

(Abidin, 2012) melakukan penelitian mengenai perancangan dan implementasi Sistem Informasi Kos Online di Surabaya Selatan. Kelebihan dari sistem ini adalah sistem mampu memberikan informasi tempat dan lokasi tempat kos yang dapat membantu pengguna mendapat tempat atau kamar kos yang sesuai. Sedangkan kekurangan dari sistem tersebut adalah sistem belum dapat menampilkan informasi dimana letak tempat lokasi berada yang berupa Google Maps.

(Astari, 2013) melakukan penelitian mengenai perancangan dan implementasi Sistem Informasi Percetakan dengan Fasilitas SMS Gateway menggunakan Framework CodeIgniter. Sistem ini mempunyai kelebihan yang mampu menyampaikan dengan cepat informasi yang berkaitan dengan pengambilan pemesanan hasil percetakan yang telah selesai. Tetapi sistem ini tidak terdapat fungsi penerimaan SMS dari pengguna.

(Husen, 2013) melakukan penelitian mengenai perancangan dan implementasi Web GIS Pariwisata Kabupaten Bantul. Sistem ini mampu memberikan informasi posisi daerah wisata dan fasilitas pencarian untuk membantu pengguna dalam mendapatkan informasi lokasi secara detail. Tetapi dalam fasilitas *near me* (pencarian lokasi terdekat) masih menggunakan inputan *start point* (lokasi awal) manual atau tidak mendeteksi secara otomatis.

Penelitian lain (Dantes, 2007) mengembangkan Sistem Informasi Akademik Berbasis SMS. Sistem tersebut dikembangkan dari sisi client (mahasiswa) dan dari sisi admin (administrator). Adapun tujuan pengembangan sistem ini adalah untuk meningkatkan kualitas layanan pendidikan di lingkungan Universitas Pendidikan Ganesha Singaraja, khususnya bagi mahasiswa jurusan D3 Manajemen Informatika.

METODOLOGI PENELITIAN

Langkah penelitian yang dilakukan dalam penelitian ini adalah sebagai berikut:

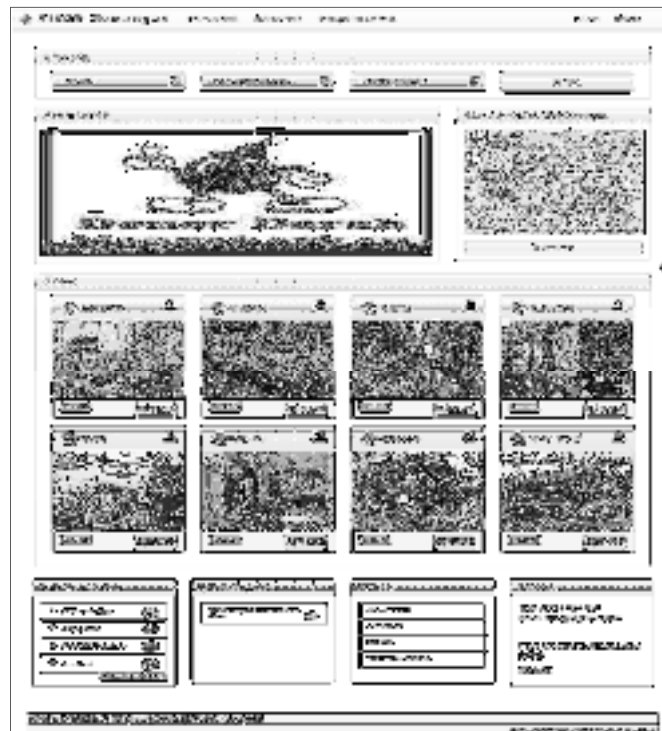
1. Menganalisis dan mengidentifikasi alat dan bahan yang diperlukan dalam penelitian.
2. Merancang *database* dan *design* sistem.
3. Melakukan pembuatan *source code* program sistem sesuai dengan *database* dan *design* yang dirancang.
4. Melakukan pengujian terhadap sistem apakah sistem sudah berjalan sesuai dengan keinginan.
5. Melakukan pencarian *bugs* atau *error* yang terjadi dalam sistem.

PEMBAHASAN

Tampilan halaman beranda

Dalam pembuatan template, sistem ini menggunakan template yang berbasis responsive dengan memanfaatkan Bootstrap yang bisa di-download melalui alamat

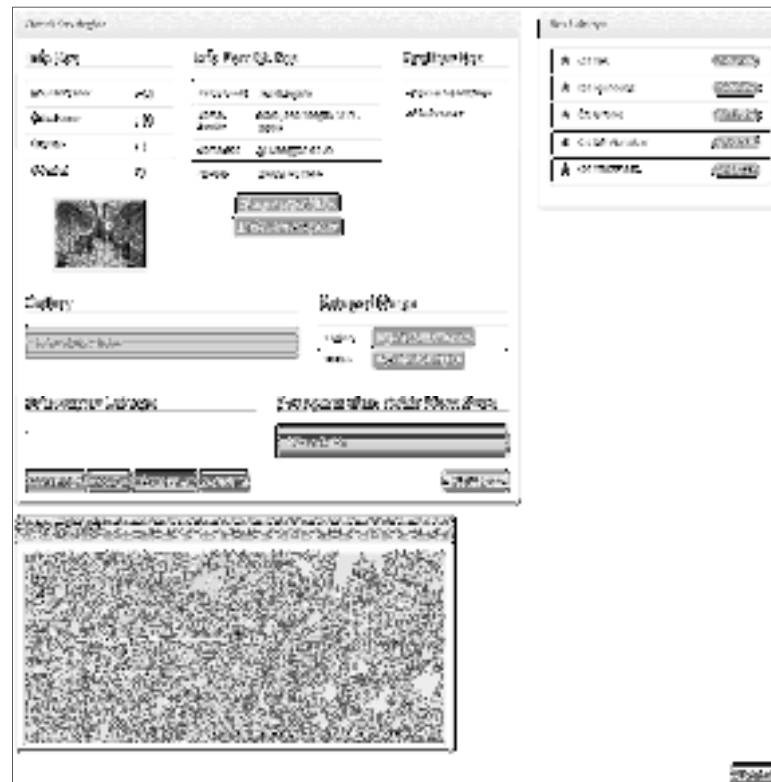
<http://getbootstrap.com/>. Responsive berarti design web dapat mengakomodasi berbagai lebar resolusi dan jenis media yang digunakan user. Sehingga layout dari sistem ini dapat menyesuaikan ketika diakses dengan media laptop, tablet atau handphone. Halaman beranda menampilkan beberapa menu yang bisa diakses oleh seorang guest/tamu. Fasilitas yang sangat mudah digunakan adalah dalam hal pencarian kos dengan disediakan menu pilihan seperti pencarian berdasarkan jenis laki-laki atau perempuan, berdasarkan waktu lama sewa dan berdasarkan harga per waktu lama sewa. Selain itu pemanfaatan Google Maps juga digunakan untuk melihat lokasi kos yang dipilih dan bahkan ada fasilitas yang dapat menampilkan dimana saja kos yang terdekat dari tempat-tempat umum seperti kos yang terdekat dari kampus IST AKPRIND Yogyakarta. Tampilan halaman beranda dapat dilihat pada Gambar 1.



Gambar 1 Tampilan halaman beranda *desktop*

Tampilan halaman detail kos

Gambar 2 merupakan tampilan halaman detail kos yang berisi tentang informasi dari suatu kos. Informasi tersebut berupa status, fasilitas, harga, informasi pemilik, *user* yang akan habis masa sewa dan *gallery*.



Gambar 2 Tampilan halaman detail kos

Tampilan halaman tempat umum terdekat

Gambar 3 merupakan tampilan halaman yang berupa fasilitas untuk mempermudah dalam filtering pencarian kos, sehingga sistem dapat menampilkan lokasi kos berdasarkan tempat umum terdekat tersebut. Fasilitas ini juga dilengkapi dengan kategori dari tempat umum tersebut, seperti kampus, warung makan, dan kategori lainnya.



Gambar 3 Tampilan halaman tempat umum terdekat

Tampilan halaman kos terdekat

Gambar 4 merupakan merupakan hasil dari proses pencarian dengan menggunakan fasilitas pencarian terdekat dari tempat umum. Halaman ini menampilkan data rumah kos terdekat yang berupa map dengan informasi jarak dari tempat umum tersebut.



Gambar 4 Tampilan halaman kos terdekat

Tampilan halaman cek masa berlaku kos

Gambar 5 merupakan tampilan halaman untuk melihat masa berlaku kos yang ditempati. Halaman ini akan mempermudah bagi user karena terkadang penghuni kos sering lupa kapan masa kos yang mereka tempati itu habis. Tidak hanya mempermudah user, pemilik kos pun dapat dimudahkan dalam mencari siapa saja yang telah atau akan habis masa berlaku kos yang ditempati.

| NO | NAMA KOS | NO. HP | NO. WA | NO. TEL | NO. FAX | STATUS |
|----|----------|--------|--------|---------|---------|--------|
| 1 | ... | ... | ... | ... | ... | ... |
| 2 | ... | ... | ... | ... | ... | ... |
| 3 | ... | ... | ... | ... | ... | ... |

Gambar 5 Tampilan halaman cek masa berlaku kos

Tampilan halaman hasil pencarian

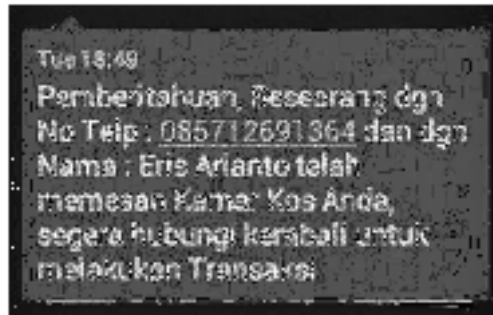
Gambar 6 merupakan tampilan halaman tentang hasil pencarian rumah kos berdasarkan kategori yang telah ditentukan. Dengan halaman ini dapat langsung terlihat beberapa rumah kos dengan keterangan status penuh atau tidak.

| NO | NAMA KOS | NO. HP | NO. WA | NO. TEL | NO. FAX | STATUS |
|----|----------|--------|--------|---------|---------|--------|
| 1 | ... | ... | ... | ... | ... | ... |
| 2 | ... | ... | ... | ... | ... | ... |
| 3 | ... | ... | ... | ... | ... | ... |
| 4 | ... | ... | ... | ... | ... | ... |
| 5 | ... | ... | ... | ... | ... | ... |

Gambar 6 Tampilan halaman cek masa berlaku kos

Tampilan notifikasi SMS kepada pemilik kos

Gambar 7 merupakan tampilan notifikasi yang dikirim oleh sistem berupa SMS kepada pemilik kos dan potongan *script* pengiriman dapat dilihat pada Gambar 8.



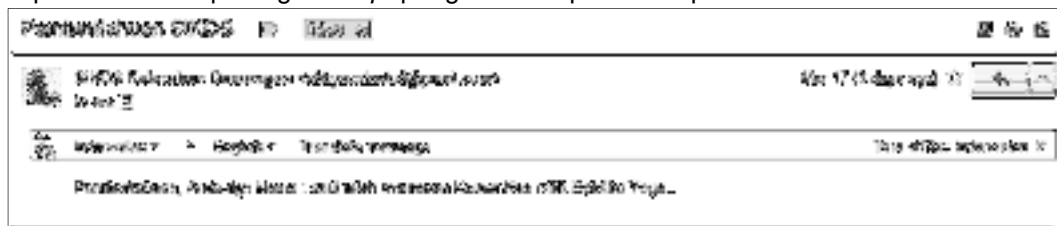
Gambar 7 Tampilan notifikasi SMS kepada pemilik kos



Gambar 8 Potongan script pengiriman notifikasi SMS

Tampilan notifikasi Email kepada user

Gambar 9 merupakan tampilan notifikasi yang dikirim oleh sistem berupa Email kepada user dan potongan script pengiriman dapat dilihat pada Gambar 10.



Gambar 9 Tampilan notifikasi Email kepada user


```

public function sendEmail($email,$pesan,$subjek) {
    $SMTP = 'smtp.gmail.com:587';
    $smtp = array(
        'host' => 'smtp.gmail.com',
        'port' => '587',
        'username' => 'santia@siaga.com',
        'password' => 'santia@siaga.com',
        'SMTPSecure' => 'tls',
        'SMTPAuth' => true,
        'isHTML' => true
    );
    $to = $email;
    $from = 'santia@siaga.com';
    $subject = 'Siaga Informasi';
    $message = $pesan;
    $headers = array(
        'From' => $from,
        'Subject' => $subject,
        'Content-Type' => 'text/html'
    );
    mail($to, $subject, $message, $headers, $SMTP);
}
    
```

Gambar 10 Potongan script pengiriman notifikasi Email

Tampilan halaman beranda admin

Gambar 11 merupakan tampilan halaman beranda admin untuk pengelolaan data.



Gambar 11 Tampilan halaman beranda admin

Tampilan halaman beranda pemilik kos

Gambar 12 merupakan tampilan halaman beranda pemilik kos. Selain admin, pemilik kos juga dapat mengelola data kos dan informasi mengenai reservasi kos mereka.



Gambar 12 Tampilan halaman beranda pemilik kos

Trigger untuk SMS

Fasilitas konfirmasi via SMS tersebut dilakukan dengan memasang *trigger* pada tabel *inbox*. *Script trigger* dapat dilihat pada Gambar 13.

dilakukan 5x *reload* dengan rentang waktu 1 detik. Hasil estimasi waktu dapat dilihat pada Gambar 15 dan Hasil dari 5x *reload* dapat dilihat pada Tabel 2.



Gambar 15 Hasil estimasi waktu *PHP Native*

Tabel 2 Hasil Reload 5x *PHP Native*

| Percobaan Ke- | Estimasi Waktu (Detik) |
|------------------|------------------------|
| 1. | 0.005051 |
| 2. | 0.005493 |
| 3. | 0.004850 |
| 4. | 0.004661 |
| 5. | 0.005853 |
| Rata-rata | 0.0042494 |

Framework CodeIgniter

Seperti pada program berbasis *PHP native*, pengujian dengan *Framework CodeIgniter* juga dibuat beberapa *file* dengan konsep *MVC (Model View Controller)* diantaranya *file controller*, *model* dan *view*. *File* untuk *controller* dapat dilihat pada Gambar 16, *file* untuk *model* dapat dilihat pada Gambar 17 dan *file controller* dapat dilihat pada Gambar 18.

```

<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');

class Welcome extends CI_Controller {

    public function index()
    {
        $this->load->helper('url', 'text');
        $data['url'] = $this->url->base_url();
        $data['text'] = 'Welcome to CodeIgniter!';
    }
}
    
```

Gambar 16 File controller Framework CodeIgniter

```

<?php

class Mhs extends CI_Model {

    function mahasiswa()
    {
        return $this->db->select('Mhs')->get();
    }
}
    
```

Gambar 17 File model Framework CodeIgniter

```

<?php
class Controller {
    public function __construct() {
        $this->model = new Model();
        $this->view = new View();
    }
}

class Model {
    public function __construct() {
        $this->data = array();
    }
}

class View {
    public function __construct() {
        $this->data = array();
    }
}

class Controller {
    public function __construct() {
        $this->model = new Model();
        $this->view = new View();
    }
}

class Model {
    public function __construct() {
        $this->data = array();
    }
}

class View {
    public function __construct() {
        $this->data = array();
    }
}
    
```

Gambar 18 File view Framework CodeIgniter

Dari program diatas kemudian dijalankan dengan *web browser* untuk mengetahui berapa estimasi waktu yang diperlukan untuk membaca 1000 buah data. Pengujian ini dilakukan 5x *reload* dengan rentang waktu 1 detik. Hasil estimasi waktu dapat dilihat pada Gambar 19 dan Hasil dari 5x *reload* dapat dilihat pada Tabel 3.



Gambar 19 Hasil estimasi waktu Framework CodeIgniter

Tabel 3 Hasil Reload 5x Framework CodeIgniter

| Percobaan Ke- | Estimasi Waktu (Detik) |
|------------------|------------------------|
| 1. | 0.000812 |
| 2. | 0.000859 |
| 3. | 0.000912 |
| 4. | 0.000840 |
| 5. | 0.000896 |
| Rata-rata | 0.0008638 |

Framework CakePHP

Pengujian terakhir adalah pengujian dengan *Framework CakePHP*. Seperti pada pengujian *Framework CodeIgniter*, pengujian ini dibuat beberapa file dengan konsep MVC (*Model View Controller*) diantaranya *file controller*, *model* dan *view*. *File* untuk *controller* dapat dilihat pada Gambar 20, *file* untuk *model* dapat dilihat pada Gambar 21 dan *file controller* dapat dilihat pada Gambar 22.

```

<?php
class MahasiswaController extends AppController {

    function index() {
        $data = $this->loadModel('Mahasiswa');
        $pageNo = $this->paginate($data, 'all');
        $data = $this->paginate($data, $pageNo);
    }
}
    
```

Gambar 20 File controller Framework CakePHP

```

<?php
class Mahasiswa extends AppModel {
    var $useTable = 'mahasiswa';
}
    
```

Gambar 21 File model Framework CakePHP

```

<?php
<!--
    Copyright 2005-2010 by CakePHP Foundation, Inc.
    Released under the MIT license
-->
<?php
require_once dirname(__FILE__) . '/AppController.php';
require_once dirname(__FILE__) . '/AppModel.php';
require_once dirname(__FILE__) . '/AppView.php';

class MahasiswaView extends AppView {
    var $helpers = array('Form');
    var $uses = array('Mahasiswa');
    function index() {
        $data = $this->paginate($this->loadModel());
        $pageNo = $this->paginate($data, 'all');
        $data = $this->paginate($data, $pageNo);
        $this->render('index');
    }
}
    
```

Gambar 22 File view Framework CakePHP

Dari program diatas kemudian dijalankan dengan *web browser* untuk mengetahui berapa estimasi waktu yang diperlukan untuk membaca 1000 buah data. Pengujian ini dilakukan 5x *reload* dengan rentang waktu 1 detik. Hasil estimasi waktu dapat dilihat pada Gambar 23 dan Hasil dari 5x *reload* dapat dilihat pada Tabel 4.

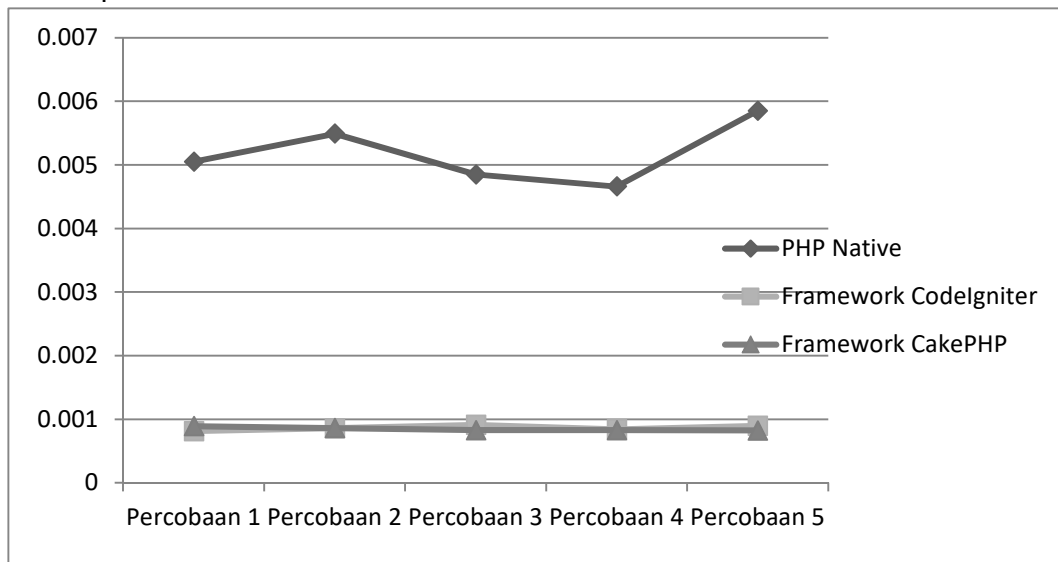


Gambar 23 Hasil estimasi waktu Framework CakePHP

| Percobaan Ke- | Estimasi Waktu (Detik) |
|---------------|------------------------|
| 1. | 0.000892 |
| 2. | 0.000862 |
| 3. | 0.000892 |
| 4. | 0.000828 |
| 5. | 0.000823 |

| | |
|------------------|------------------|
| Rata-rata | 0.0008594 |
|------------------|------------------|

Dari beberapa pengujian diatas dapat disimpulkan dalam Gambar 24 yang dapat dilihat bahwa terdapat perbedaan yang sangat signifikan antara program berbasis *PHP native* dan berbasis *Framework* dan dapat dikatakan bahwa program berbasis *Framework* mempunyai kecepatan 4 sampai 5 kali lebih cepat dari program berbasis *PHP native*



Gambar 24 Grafik kesimpulan estimasi waktu

KESIMPULAN

Berdasarkan penelitian di atas, dapat ditarik beberapa kesimpulan antara lain:

1. Telah berhasil dilakukan implementasi sistem informasi pengolahan data rumah kos di Kelurahan Demangan dengan menggunakan *Framework CodeIgniter*, *Google Maps* sebagai penunjuk lokasi dan *SMS Gateway* sebagai media notifikasi.
2. Sistem ini dapat diakses dari berbagai media seperti *notebook*, *mobile* dan *tablet*
3. Sistem ini memiliki fitur pencarian yang mudah dan proses pemesanan yang cepat.
4. Pembuatan sistem berbasis *framework* mempunyai tingkat kecepatan yang cepat dalam pemrosesan data.

DAFTAR PUSTAKA

Abidin, B. (2012). *Sistem Informasi Rumah Kos Online Berbasis Web dan Messaging*. Surabaya: Institut Teknologi Sepuluh Nopember.

Astari, D. (2013). *Sistem Informasi Percetakan dengan Fasilitas SMS Gateway menggunakan Framework CodeIgniter*. Yogyakarta: TA UGM.

Bootstrap. (2013, November 11). *Bootstrap*. Retrieved November 15, 2013, from Bootstrap: <http://getbootstrap.com>

CakePHP. (2014, Januari 28). *CakePHP*. Retrieved Januari 29, 2014, from CakePHP: <http://cakephp.org/>

Dantes, G. R. (2007). Rancangan Dan Implementasi Sistem Informasi Akademik Berbasis SMS. *Jurnal Penelitian dan Pengembangan Sains & Humaniora* , 11.

Ellislab. (2013, Oktober 16). *CodeIgniter*. Retrieved Oktober 18, 2013, from CodeIgniter: <http://ellislab.com/codeigniter/>

Google. (2013, Desember 11). *Google Maps*. Retrieved Desember 15, 2013, from Google

Maps: <https://maps.google.com>

Husen, M. S. (2013). *Sistem Informasi Lokasi Industri di Tingkat Kabupaten Bantul Berbasis Geographic Information System (GIS)*. Yogyakarta: IST AKPRIND Yogyakarta.

Saputra, A (a). (2013). *Step By Step Membangun Aplikasi Bioskop Dan SMS*. Yogyakarta: Lokomedia.

Saputra, A (b). (2013). *Trik Kolaborasi CodeIgniter & jQuery*. Yogyakarta: Lokomedia.

Sidik, B. (2013). *Framework Codeigniter*. Bandung: Informatika.

Sirenden, B. H. (2013). *Buat Sendiri Aplikasi Petamu menggunakan CodeIgniter dan Google Maps API*. Yogyakarta: Andi Offset.

Westriningsih. (2013). *Javascript Source Code Paling dicari*. Yogyakarta: Andi Offset.