

INTEGRASI DAN AKSES DATA MULTI DATABASE MENGUNAKAN FRAMEWORK YII

Badiyanto¹⁾

¹⁾Teknik Informatika STMIK AKKAKOM Yogyakarta
Jalan Raya Janti 143, Karang Jambe, Yoyakarta 55198
email:badi@akakom.ac.id

ABSTRAK

Penelitian ini membuat aplikasi protipe yang mampu mengintegrasikan database MySQL, PostgreSQL dan SQLite. Aplikasi tersebut akan dibangun menggunakan bahasa pemrograman PHP dengan Framework Yii. Tujuan dari penelitian ini ingin mendapatkan hasil aplikasi pototipe yang digunakan sebagai acuan untuk membantu para pengembang dalam membangun aplikasi web yang terkoneksi ke multi database. Pengerjaan melalui beberapa tahap: mempelajari dan mengumpulkan sub sistem lama, kemudian perancangan, membuat program dan melakukan uji coba. Bahasa pemrograman pemrograman yang digunakan: php native, dan Framework Yii dengan memanfaatkan komponen Data Access Object, Query Builder serta kelas CActiveRecord. Setelah dilakukan pengujian dengan melakukan multi koneksi dan relasi tabel antar database secara langsung, hasil yang terbaik adalah menggunakan Object-relational mapping (ORM) di dalam kelas CActiveRecord.

kata kunci : integrasi, database, relasi, framework

ABSTRACT

This research makes an application prototype that is able to integrate MySQL database, PostgreSQL and SQLite. The application will be built using the PHP programming language with Yii Framework. The purpose of this study want to get the application pototipe used as a reference to help developers to build web applications that connect to multiple databases. The execution through several stages: studying and collecting existing sub-systems, then the design, programming and testing. Programming language used: php native and Yii Framework by utilizing components of the Data Access Object, Query Builder and CActiveRecord class. After testing by performing multi connections and relationships between database tables directly, the best results are using the Object - relational mapping (ORM) in the classroom CActiveRecord .

keywords : integration , database , relation, Framework

I. PENDAHULUAN

Membangun Sistem Informasi berskala menengah hingga besar membutuhkan biaya yang cukup besar. Untuk mewujudkan membutuhkan tahapan sesuai dengan ketersediaan biaya. Maka biasanya dilakukan tarhap demi tahap menurut sekala prioritas. Kendala yang sering terjadi adalah jangka waktunya cukup lama, dan pemilihan perangkat lunak bahasa pemrogram yang tidak seragam, serta penggunaan platform database yang berbeda, karena mengikuti perkembangan perangkat lunak saat itu. Masalah lainnya adalah jika suatu organisasi belum mempunyai perencanaan jangka panjang dan setiap unit kerja sudah dibangun sistem sendiri-sendiri. Sebagai contoh suatu institusi perguruan tinggi telah mempunyai beberapa database, misalnya sistem akademik menggunakan MySQL, sistem kepegawaian menggunakan PosgreSQL dan sistem keuangan menggunakan SQLite. Tentunya tidak mudah untuk memigrasi database yang ada ke database yang baru. Untuk itu perlu adanya suatu cara mengintegrasikan database tersebut.

Teknologi framework adalah metode baru untuk pembuatan perangkat lunak. Salah satu framework yang terbaik adalah Framework Yii. Keuntungan menggunakan Framework Yii adalah penulisan kode secara efisien dan cepat, kode program tidak dimulai dari nol, karena Framework Yii menyediakan template dan utilitas generator kode [1]. Beberapa penelitian sebelumnya telah menerapkan Heterogenous Distributed Database System untuk prototipe sistem informasi akademik yang dapat meningkatkan kapasitas Oracle XE 10g dengan memanfaatkan MySQL ODBC dan database link sebagai pintu gerbang penghubung basis data Oracle XE 10g dengan database MySQL dalam membangun sebuah jaringan database terdistribusi [5]. Model integrasi database digunakan bertukar data antar Sistem Informasi Berbasis Komputer (SIBK) dengan memanfaatkan data primer data kependudukan nasional yang dapat digunakan oleh berbagai instansi lewat identitas penduduk yang unik yang diebut Single Identity Number (SIN) [2]. Integrasi data center epidemiologi dinas kesehatan yang mana sumber data epidemiologinya berasal dari pelaporan data epidemiologi puskesmas dan rumah

sakit, sumber datanya berasal dari aplikasi dan struktur data, DBMS yang berbeda (MySQL dan PostgreSQL) [3]. Teknologi SOAP (Simple Object Access Protocol) berbasis XML dengan pustaka web service (nusoap) atau SOA (Service-Oriented Architecture), digunakan untuk mengintegrasikan multi database serta berbagai macam perbedaan platform aplikasi dan sistem operasi. Hasilnya untuk membantu organisasi antar unit kerja dalam penyediaan data secara real-time yang digunakan untuk mendukung suatu keputusan atau kebijakan yang akan diambil oleh pihak pimpinan [4]. Teknologi web service digunakan untuk akses data induk (master) pada sistem akademik secara otomatis yang mengintegrasikan tiga sub sistem informasi sistem informasi yang ada dalam lingkup fakultas hasilnya memudahkan administrator mengambil data sehingga dapat menjaga konsistensi data antara sistem [7].

Melihat kondisi di atas penelitian ini membuat prototipe bagaimana caranya mengkoneksikan beberapa database dengan platform yang berbeda menggunakan Framework Yii, serta melakukan percobaan relasi tabel antar database, dilanjutkan uji transaksi untuk mengetahui keberhasilan tidaknya integritas data antar tabel.

II. METODE PENELITIAN

Metode penelitian ini terdiri dari beberapa tahap:

Lokasi dan sumber data, data yang digunakan pada penelitian ini mengambil dari 3 (tiga) sub sistem informasi yang ada di lingkungan sistem informasi di STMIK AKAKOM Yogyakarta.

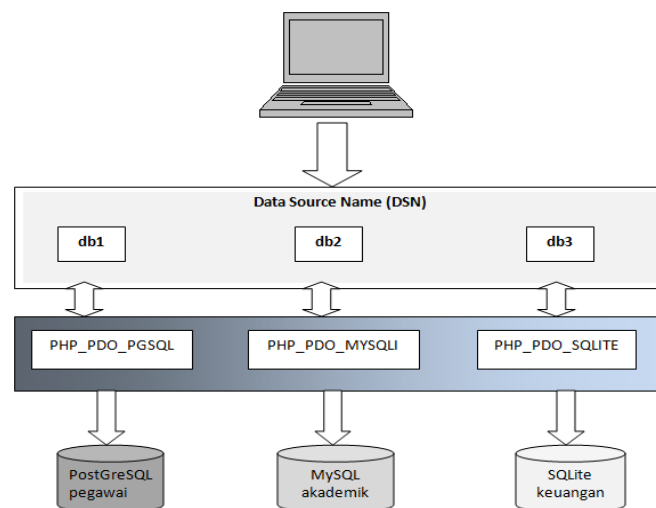
Rancangan, rancangan dilakukan dengan berdasarkan sistem yang sudah ada mengambil sebagian dari 3 (tiga) sub sistem yang telah berjalan antara lain: sub sistem informasi akademik, keuangan, kepegawaian di STMIK AKAKOM Yogyakarta.

Selanjutnya mengumpulkan dan mempelajari struktur data tabel dari masing-masing sub sistem. Dari struktur yang ada akan dibuat suatu model dalam bentuk turunan kelas/objek yang telah disediakan oleh Yii.

Pengujian dilakukan merelasikan tabel antar database dengan membandingkan dari 3 metode, pertama menggunakan php *native*, yang kedua menggunakan Yii, menyediakan 3 tiga cara antara lain: (1) *Data Access Object (DAO)*; (2) *Query Builder*; (3) menggunakan kelas *CActiveRecord*, untuk memperoleh hasil analisis yang sesuai.

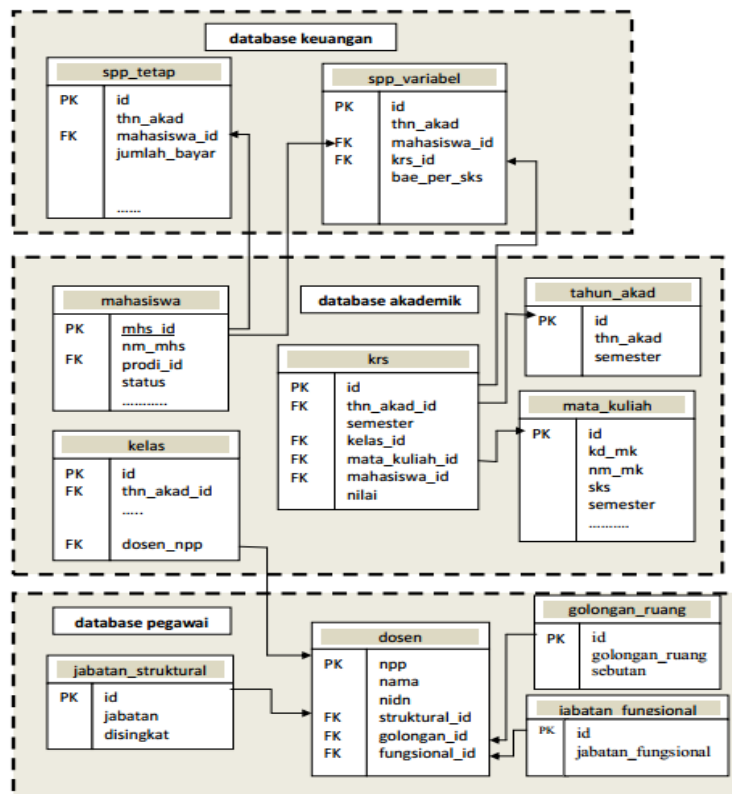
III. HASIL

Untuk akses ke database menggunakan *Data Source Name (DSN)* dikonfigurasi dalam file `config/main.php`, menggunakan parameter `components` diberi nama `db1`, `db2`, `db3`. Koneksi antar sistem ditunjukkan pada gambar 3.



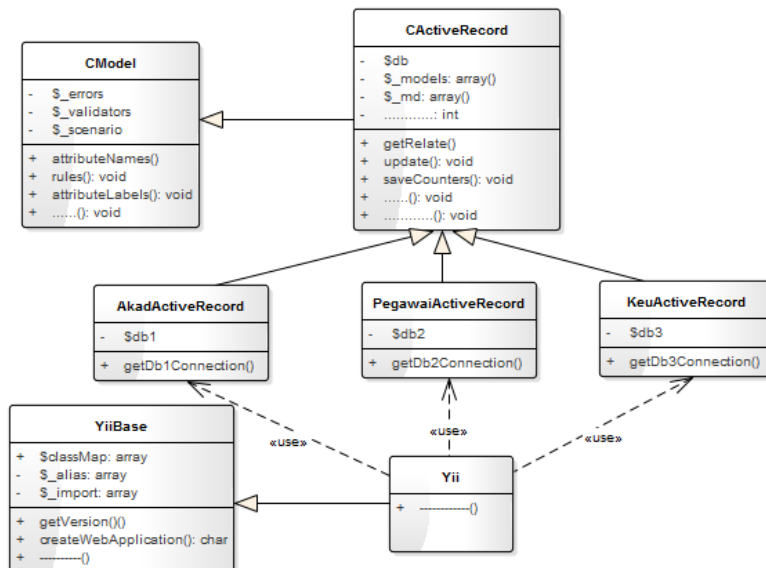
Gambar 1. Bagan koneksi dengan PHP PDO

Relasi tabel dilakukan satu database maupun antar database, misalkan tabel `mahasiswa` dalam database akademik, relasi dengan tabel `spp_variabel` dalam database keuangan, adapun bentuk relasinya seperti ditunjukkan pada gambar 4.



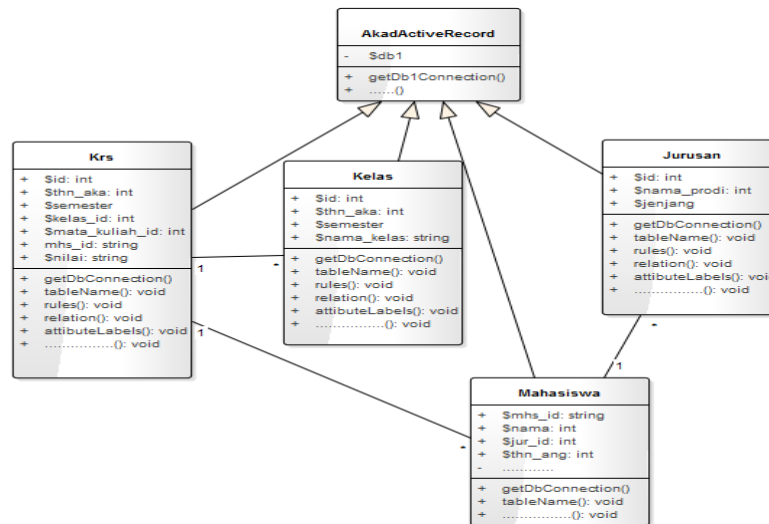
Gambar 2. Relasi tabel antar database

Framework Yii, tabel direpresentasikan dengan kelas `CActiveRecord`. Untuk menghubungkan koneksi ke tabel dengan perintah `class AkadActiveRecord extends CActiveRecord`, digambarkan dalam diagram kelas seperti ditunjukkan pada gambar 5.



Gambar 3. Diagram kelas koneksi

Kelas `AkaCActiveRecord` merupakan perluasan kelas dari `CActiveRecord` yang menggunakan koneksi `db1`, `db1` adalah koneksi ke database MySQL. Untuk mengakses tabel `krs` direpresentasikan dengan kelas/model `Krs`, Jurusan, Kelas Mahasiswa perluasan dari kelas `AkaCActiveRecord`, seperti ditunjukkan pada gambar 6.



Gambar 4. Diagram kelas Akademik

Pengujian program dengan cara memanggil url:<http://ns.akad/>, kemudian aplikasi akan mengarahkan ke <http://ns.akad/index.php?r=site/login>, seperti ditunjukkan pada gambar 7.

Gambar 7. Halaman Login

Untuk membaca transkrip nilai mahasiswa dengan cara memanggil ke url <http://ns.akad/index.php?r=krs/khs>, hasilnya seperti ditunjukkan pada gambar 8.

Gambar 8. Formulir masukan KHS

Ketika pengguna menekan tombol “Proses” dilanjutkan ke link url http://ns.akad/index.php?r=krs/transkripKhs&mhs_id=105610091&thn_akad=2011&smt=1, hasilnya seperti ditunjukkan gambar 9.

TRANSKRIP KARTU HASIL STUDI					
Tahun Akademik	2011/2012				
Semester	Ganjil				
NO	105610091				
Nama	RAHMAD QADRI				
Jurusan	Sistem Informasi				
No	Kode	Mata Kuliah	Nilai	Sks	Skor
1	SI2006T	PENGENALAN UNIFIED MODELLING LANGUAGE	E	2	0
2	SI2007T	PEMROGRAMAN BERORIENTASI OBYEK	A	3	12
3	AK2005P	PRAKT. JARINGAN KOMPUTER	A	1	4
4	AK2004P	PRAKT. PENGENALAN WEB	A	1	4
5	SI2008T	BASIS DATA 2	C	2	4
6	AK3213P	BAHASA INGGRIIS 3	C	3	6
7	SI2008P	PRAKT. BASIS DATA	B	1	3
8	SI2007P	PRAKT. PEMROGRAMAN BERORIENTASI OBYEK	C	1	2
9	AK2005T	JARINGAN KOMPUTER	B	3	9
10	AK2004T	PENGENALAN WEB	B	3	9
			Jumlah	20	53

Indeks Prestasi = 2,65

Gambar 9. Trankrip KHS

IV. PEMBAHASAN

Cara koneksi ke database diletakkan dalam file `config/main.php`, terdapat pada parameter `component`, `db1` koneksi ke database akademik dan `db2` koneksi ke database kepegawaian, serta `db3` koneksi ke database keuangan, seperti ditunjukkan pada kode program `main.php` berikut:

```

.....
'db1'=>array(
    'connectionString' => 'mysql:host=ns.akademik;
                        dbname=akademik',
    'emulatePrepare' => true,
    'username' => 'badi',
    'password' => 'badi',
    'charset' => 'utf8',
    'class'=> 'CDbConnection'
),
'db2'=>array(
    'connectionString' => 'pgsql:host=ns.kepegawian;
                        port=5432;
                        dbname=pegawai',
    'username'=>'badi',
    'password'=>'badi',
    'charset'=>'UTF8',
    'class'=> 'CDbConnection'
),
'db3'=>array(
),
.....

```

Cara mengelompokkan dan mengarahkan koneksi ke database akademik menggunakan kelas `AkadActiveRecord`, koneksiya menggunakan metode `getDb1Connection()`, seperti ditunjukkan pada kode program `AkadActiveRecord.php` berikut:

```

<?php
class AkadActiveRecord extends CActiveRecord
{
    private static $db1 = null;

    protected static function getDb1Connection()
    {
        if (self::$db1 !== null)
            return self::$db1;
        else
        {
            self::$db1 = Yii::app()->db1;
            if (self::$db1 instanceof CDbConnection)

```

```

        {
            self::$db1->setActive(true);
            return self::$db1;
        }
        else
            throw new CDbException(Yii::t('yii','Active Record requires
                "db1" CDbConnection application component.'));
    }
}
?>

```

Untuk mengelompokkan dan mengarahkan koneksi ke database kepegawaian menggunakan kelas PegawaiActiveRecord, koneksi menggunakan metode `getDb2Connection()` seperti pada kode program PegawaiActiveRecord.php berikut:

```

<?php
class PegawaiActiveRecord extends CActiveRecord
{
    private static $db2 = null;
    protected static function getDb2Connection()
    {
        if (self::$db2 !== null)
            return self::$db2;
        else
        {
            self::$db2 = Yii::app()->db2;
            if (self::$db2 instanceof CDbConnection)
            {
                self::$db2->setActive(true);
                return self::$db2;
            }
            else
                throw new CDbException(Yii::t('yii','Active Record requires
                    "db2" CDbConnection application component.'));
        }
    }
}

```

Untuk mengelompokkan dan mengarahkan koneksi ke database keuangan menggunakan kelas KeuActiveRecord, koneksi menggunakan metode `getDb3Connection()`, seperti pada kode program KeuActiveRecord.php berikut:

```

<?php
class KeuanganActiveRecord extends CActiveRecord
{
    private static $db3 = null;
    protected static function getDb2Connection()
    {
        if (self::$db3 !== null)
            return self::$db3;
        else
        {
            self::$db3 = Yii::app()->db3;
            if (self::$db3 instanceof CDbConnection)
            {
                self::$db3->setActive(true);
                return self::$db3;
            }
            else
                throw new CDbException(Yii::t('yii','Active Record requires
                    "db3" CDbConnection application component.'));
        }
    }
}

```

Dalam Framework yii suatu tabel krs direpresentasikan di dalam kelas/model Krs peluasan dari AkadActiveRecord seperti ditunjukkan pada kode program Krs.php berikut:

```

<?php

```

```

class Krs extends AkadActiveRecord
{
    public function getDbConnection(){
        return self::getDb1Connection();
    }

    // memanggil tabel
    public function tableName()
    {
        return 'krs';
    }
}
.....

```

Relasi Objek dengan *Object-relational mapping (ORM)*. Relasi antar objek diletakkan dalam metode `relations()` di masing-masing model. Penulisan relasi objek ditunjukkan seperti pada kode program berikut:

```

public function relations()
{
    return array(
        'mataKuliah' => array(self::BELONGS_TO, 'MataKuliah', 'mata_kuliah_id',
        'kelas' => array(self::BELONGS_TO, 'Kelas', 'kelas_id'));
    }
}

```

'mataKuliah' adalah nama relasi, `self::BELONGS_TO` adalah hubungan Krs dan MataKuliah, banyak dan satu. 'MataKuliah' nama kelas/model, dan 'mata_kuliah_id' kunci FK.

Untuk membaca rekaman krs menggunakan kelas Krs. Cara memanggil menggunakan metode `findAll()`. Pemanggilan kelas terdapat pada kode program berikut :

```

$model=Krs::model()->findAll('mahasiswa_id=:id AND thn_akad=:ta AND
semester=:smt', array(':id'=>$mhs_id,
':ta'=>$thn_akad, ':smt'=>$smt));

```

Bagian controller `KrsController`, pada saat pengguna meminta ke aplikasi, controller akan menjalankan aksi lewat metode yang bernama awalan `action`. Di dalam `KrsController.php` terdapat metode `actionKhs()`, dalam metode `actionKhs()` *me-render* ke view `khs.php`, seperti ditunjukkan pada kode program berikut:

```

class KrsController extends Controller
{
    .....
    public function actionKhs()
    {
        $this->layout='//layouts/column2';
        $model=new Krs;
        $model->unsetAttributes();
        if(isset($_POST['Krs']))
        {
            $model->attributes=$_POST['Krs'];
            $mhs_id      =$_POST['Krs']['mahasiswa_id'];
            $thn_akad    =$_POST['Krs']['thn_akad'];
            $smt         =$_POST['Krs']['semester'];
            $this->redirect(array(
                'transkripKhs', 'mhs_id'=>$mhs_id,
                'thn_akad'=>$thn_akad, 'smt'=>$smt ));
        }
        $this->render('khs', array('model'=>$model));
    }

    public function actionTranskripKhs() {
        $model=new Krs;
        $this->render('transkripKhs', array('model'=>$model));
    }
}

```

Menggunakan *native SQL*

Akses data menggunakan sql akses data langsung hanya mendukung 1 koneksi. Untuk mengakses ke tabel ditunjukkan seperti pada kode program berikut:

```
<?php
.....
$conn = new mysqli($servername, $username, $password, $dbname);
$sql="SELECT krs.thn_akad, krs.semester, mata_kuliah.kd_mk
, mata_kuliah.nm_mk, mata_kuliah.sks, krs.nilai
FROM r_akademik.krs
INNER JOIN r_akademik.mata_kuliah
ON (krs.mata_kuliah_id = mata_kuliah.id)
WHERE krs.mhs_id='105610091' AND
krs.thn_akad='2011' AND krs.smt_id=1";
$result = $conn->query($sql);
.....?>
```

Menggunakan *Data Access Object (DAO)*

Framework Yii, menyediakan akses database bisa menggunakan *DAO*. Model akses menggunakan *DAO* hampir sama dengan *native* hanya bedanya koneksi. Adapun cara mengakses data dengan *DAO*, ditunjukkan pada kode program berikut:

```
<?php
public function actionTampilkan(){
    $sql=" SELECT krs.thn_akad, krs.semester, mata_kuliah.kd_mk
, mata_kuliah.nm_mk, mata_kuliah.sks, krs.nilai
FROM r_akademik.krs
INNER JOIN r_akademik.mata_kuliah
ON (krs.mata_kuliah_id = mata_kuliah.id)
WHERE krs.mhs_id='105610091' AND
krs.thn_akad='2011' AND krs.smt_id=1";
    $cmd = Yii::app()->db->createCommand($sql);
    $rows=$cmd->queryAll();
    $this->render('transkripKhs',array('rows'=>$rows,));
}
?>
```

Menggunakan *Query Builder*

Akses menggunakan *Query Builder* membentuk perintah SQL ke dalam fungsi dari kelas Yii. Setiap satu perintah hanya satu koneksi, adapun perintahnya seperti ditunjukkan pada kode program berikut:

```
<?php
public function actionTampilkan(){
    $hasil = Yii::app()->db1->createCommand()
->select(' SELECT krs.thn_akad,
krs.semester, mata_kuliah.kd_mk
, mata_kuliah.nm_mk, mata_kuliah.sks
, krs.nilai
->from('krs')
->join('mata_kuliah','krs.mata_kuliah_id=mata_kuliah.id')
->where('krs.mhs_id=:mhs_id AND
krs.thn_akad=:thn_akad AND krs.smt_id=:smt_id',
array(':mhs_id'=>'105610091', ':thn_akad'=>'2011',
':smt_id'=>1))
->queryAll();

    $this->render('tampilkan',array('hasil'=>$hasil));
}
<?php
```

Menggunakan *CActiveRecord*

Kelas/model merupakan representasi dari tabel. Akses tabel membutuhkan koneksi maka setiap penggunaan *CActiveRecord*, untuk melakukan koneksi terdapat pada file config/main.php, dengan parameter 'components'=>array('db1'=>array(...)). Untuk parameter db1, db2,

db3 bisa diarahkan ke masing-masing database. Sehingga penggunaan model seperti ditunjukkan pada kode program `Krs.php` berikut:

```
<?php

class Krs extends AkadActiveRecord {

public function getDbConnection(){
    return self::getDb1Connection();
}

public function relations(){
    return array(
        'mataKuliah'=>array(self::BELONGS_TO, 'MataKuliah', 'mata_kuliah_id',
        'kelas' => array(self::BELONGS_TO, 'Kelas', 'kelas_id')),);
    }
    .....
?>
```

Menggunakannya di dalam views seperti ditunjukkan pada kode program berikut:

```
<?php
...
$model=Krs::model()->findAll(
    'mahasiswa_id=:id AND thn_akad=:ta
    AND semester=:smt',
    array(':id'=>$mhs_id, ':ta'=>$thn_akad, ':smt'=>$smt));
.....
?>
```

Hasil Analisis Perbandingan

Setelah melakukan sejumlah uji coba dengan membandingkan sejumlah akses ke tabel, hasil terangkum seperti ditunjukkan pada tabel 1.

TABEL I.
DUKUNGAN RELASI TABEL ANTAR DATABASE

Kelas/Metode	db1 (MySQL)	db2 (PostgreSQL)	db3 (SQLite)	Bisa Relasi tabel Antar Database
native	SQL	SQL	SQL	Tidak
DAO	SQL	SQL	SQL	Tidak
SQL Builder	SQL Builder	SQL Builder	SQL Builder	Tidak
CActiveRecord	ORM	ORM	ORM	Ya

V. SIMPULAN DAN SARAN

Berdasarkan pengujian integrasi dan akses data *multi database* menggunakan Framework Yii maka dapat diambil kesimpulan sebagai berikut :

- 1) Secara umum menggunakan pustaka `php_pdo_mysql`, `php_pdo_pgsql`, `php_pdo_sqlite` dalam satu aplikasi memungkinkan ter koneksi ke beberapa database yang berbeda, tetapi tidak mendukung relasi tabel secara langsung
- 2) Penggunaan `php native`, `DAO`, dan `Query Builder` tidak mendukung relasi tabel antar database secara langsung.
- 3) Penggunaan Framework Yii dalam kelas `CActiveRecord` dengan dukungan `ORM` mampu menghubungkan tabel antar database secara langsung menggunakan relasi antar model/objek.

REFERENSI

- [1] Badiyanto, Buku Pintar Framework YII Cara Mudah Membangun Aplikasi Web PHP, Mediakom, Yogyakarta, 2013:7-8.
- [2] Edhy Sutanta, Model Integrasi Database Penduduk Indonesia dengan Berbagai Sistem Informasi Berbasis Komputer, Program Studi Teknik Informatika UAD Yogyakarta Jurnal Informatika Vol. 5, No. 2, Juli 2011, 2011:542-553
- [3] Fikri B., Slamet S., Muslih, Desain integrasi data antar database epidemiologi untuk Mendukung pusat data kesehatan dengan menggunakan soa webservice, Prosiding SNATIF Ke-2 Tahun 2015, 2015:90-100.
- [4] Mugi Sugiarto, Pelita Fajarhati, Implementasi Integrasi Data Antar Sistem Informasi Untuk Mendukung *Decision Support System*, Konferensi dan Temu Nasional Teknologi Informasi dan Komunikasi untuk Indonesia, 2008.

- [5] M. Didik R. W., Anas A. Q., Rancang Bangun *Heterogenous Distributed Database System* Untuk Meningkatkan Kapasitas Oracle Xe 10g Pada Studi Kasus Sistem Informasi Akademik, Kaunia, Vol. IX, No. 2, Oktober 2013, 2013:60-72
- [6] Pressman S.R., *Software Engineering A Practitioner's Approach Seventh Edition*, McGraw-Hill Companies, NewYork, NY 10020, 2010:43-44.
- [7] Sitti A, Amil A.I Muh. N, Integrasi Multi database menggunakan teknologi web service, Fakultas Teknik Universitas Hasanuddin Makassar, 2015.