

## BAB II. TINJAUAN PUSTAKA DAN LANDASAN TEORI

### 2.1 Tinjauan Pustaka

Dalam penelitian ini digunakan beberapa sumber pustaka. Penelitian sebelumnya telah dilakukan oleh (Asridina 2013), untuk pembuatan aplikasi pendaftaran penjadwalan pertandingan olah raga yang diselenggarakan oleh KONI Kabupaten Asahan Sumatra Utara. Aplikasi ini dibangun menggunakan Framework Yii untuk membantu pencarian informasi jadwal dan hasil skor pertandingan.

Kemudian penelitian lainnya dilakukan oleh (Hartadi,2014), dalam penelitiannya membahas tentang aplikasi manajemen dari Tabungan Kurban, lingkup yang dikerjakan pembukuan klien termasuk kredit, debit dan saldo rekening tabungan, maka manajemen data pelanggan, dan laporan keuangan.

Pengembangan perangkat lunak menggunakan metodologi *Rapid Application Development*, serta bahasa pemrograman PHP dengan Framework Yii. Framework Yii juga digunakan oleh (Lily, 2015), untuk mengembangkan aplikasi perpustakaan di Pusat Pembelajaran “TRAMPIL”, yang membantu pustakawan untuk mengelola dan mendistribusikan informasi fisik dan koleksi digital.

(Mugi Sugiarto, 2008) telah mengintegrasikan multi database menggunakan teknologi *SOAP (Simple Object Access Protocol)* berbasis XML, dengan pustaka web service (nusoap) atau *SOA (Service-Oriented Architecture)*. Hasilnya untuk membantu organisasi dalam mendukung suatu keputusan atau kebijakan yang akan diambil oleh pihak pimpinan, ketersediaan data.

(Sitti dkk, 2013) telah mengintegrasikan tiga sistem informasi sistem informasi yang ada pada lingkup fakultas teknik menggunakan web service yang memudahkan admin mengambil data sehingga dapat menjaga konsistensi data antara sistem informasi yang telah diintegrasikan.

(Zelika,2014) juga membahas tentang pengembangan perangkat lunak rekrutmen online di PT Inti Persero menggunakan Framework Yii dengan tools pemodelan diagram *UML (Unified Modelling Language)*, yang dapat memberikan solusi untuk setiap masalah yang timbul, mempermudah sumber daya atau panitia

rekrutmen dalam pengelolaan dari segi proses filtering, validasi dan verifikasi data lamaran, untuk mendapatkan hasil seleksi data yang lolos.

Tabel 2.1. Tinjauan Pustaka

No	Nama Peneliti	Judul	Arsitektur	Hasil
1.	Mugi Sugiarto, Pelita Fajarhati, 2008	Implementasi Integrasi Data Antar Sistem Informasi Untuk Mendukung Decision Support System	SOAP, SOA, nusoap, XML	dapat meningkatkan performansi integrasi data.
2.	Asridina 2013	Aplikasi Pendaftaran dan Penjadwalan Pertandingan olahraga studi kasus : KONI Kabupaten Asahan Sumatera Utara	Framework Yii	Membantu pencarian informasi jadwal dan hasil skor pertandingan.
3.	Sitti dkk, 2013	Integrasi Multi database menggunakan teknologi web service,	Web Service	memudahkan admin mengambil data sehingga dapat menjaga konsistensi data antara sistem informasi yang telah diintegrasikan
4.	Hartadi, 2014	<i>Development of web-based savings Kurban management application with Yii framework case study: CV Almann</i>	Framework Yii	pembukuan klien termasuk kredit, debit dan saldo rekening tabungan, dan manajemen data pelanggan, dan laporan keuangan
5	Zelika, 2014	Perangkat Lunak Rekrutmen Online di PT Inti Persero Menggunakan Framework Yii	Framework Yii	validasi dan verifikasi data lamaran, untuk mendapatkan hasil seleksi data yang lolos
6.	Lily, 2015	<i>Development of Integrated Library System: A Case Study of TRAMPIL Foundation 's Learning Center</i>	Framework Yii	membantu pustakawan untuk mengelola dan mendistribusikan informasi fisik dan koleksi digital.

## 2.2 Landasan Teori

### 2.2.1 Apa itu Yii

Yii adalah *framework* (kerangka kerja) PHP berbasis-komponen, berkinerja tinggi untuk pengembangan aplikasi Web berskala-besar. Yii menyediakan reusability

maksimum dalam pemrograman Web dan mampu meningkatkan kecepatan pengembangan secara signifikan. Nama Yii (dieja sebagai /i:/) singkatan dari "**Yes It Is!**".

### **2.2.2 Bagaimana Yii Dibandingkan dengan Framework Lain?**

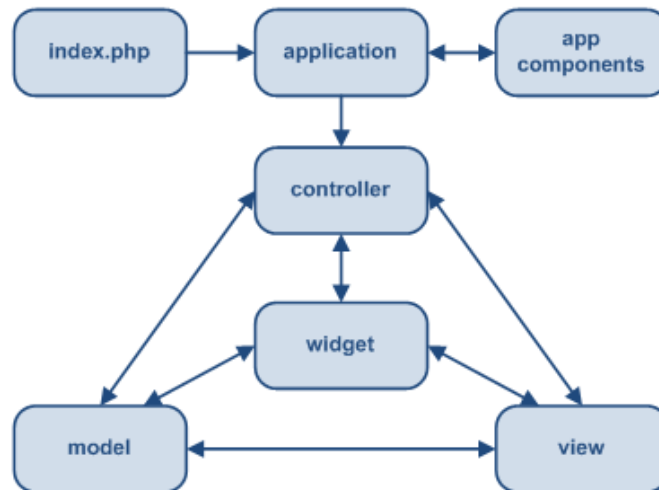
Seperti kebanyakan PHP *framework*, Yii adalah *MVC framework*. Yii melampaui framework PHP lain dalam hal efisiensi, kekayaan fitur, dan kejelasan dokumentasi. Yii didesain dengan hati-hati dari awal agar sesuai untuk pengembangan aplikasi Web secara serius. Yii bukan berasal dari produk pada beberapa proyek maupun konglomerasi pekerjaan pihak-ketiga. Yii adalah hasil dari pengalaman kaya para pembuat pada pengembangan aplikasi Web dan investigasi framework pemrograman Web dan aplikasi yang paling populer.

### **2.2.3 Model-View-Controller (MVC)**

Yii mengimplementasikan pola desain *model-view-controller (MVC)*, yang diadopsi secara luas dalam pemrograman Web. *MVC* bertujuan untuk memisahkan logika bisnis dari pertimbangan antar muka pengguna agar para pengembang bisa lebih mudah mengubah setiap bagian tanpa mempengaruhi yang lain. Dalam *MVC*, model menggambarkan informasi (data) dan aturan bisnis; view(tampilan) berisi elemen antar muka pengguna seperti teks, input form; sementara controller mengatur komunikasi antar model dan view.

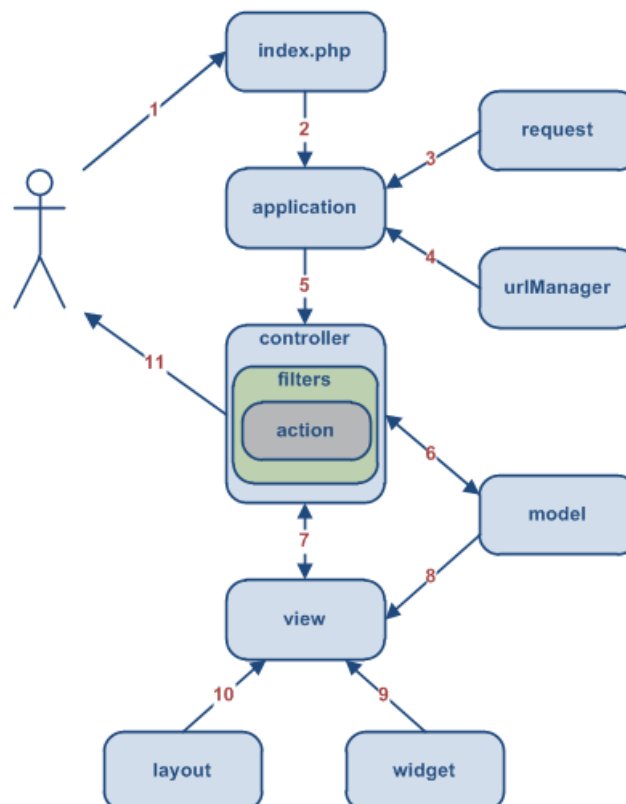
Selain implementasi *MVC*, Yii juga memperkenalkan *front-controller* (controller-depan), yang disebut Aplikasi, yang mengenkapsulasi konteks eksekusi untuk memproses sebuah permintaan. Aplikasi mengumpulkan beberapa informasi mengenai permintaan pengguna dan kemudian mengirimnya ke *controller* yang sesuai untuk penanganan selanjutnya.

Gambar 2.1 berikut memperlihatkan struktur statis sebuah aplikasi Yii:



Gambar 2.1 Struktur statis aplikasi Yii

Gambar diagram 2.1 berikut memperlihatkan alur kerja umum sebuah aplikasi Yii saat menangani permintaan pengguna



Gambar 2.2 Alur kerja umum aplikasi Yii

1. Pengguna membuat permintaan dengan

URL `http://www.example.com/index.php?r=post/show&id=1` dan

server Web menangani permintaan dengan menjalankan skrip `bootstrapindex.php`.

2. Skrip bootstrap membuat sebuah instance Aplikasi dan menjalankannya.
3. Aplikasi mendapatkan rincian informasi permintaan pengguna dari komponen aplikasi bernama request.
4. Aplikasi menentukan controller dan aksi yang diminta dengan bantuan komponen aplikasi bernama `urlManager`. Dalam contoh ini, controller adalah `post` yang merujuk pada kelas `PostController`; dan aksi adalah `show` yang arti sebenarnya ditentukan oleh controller.
5. Aplikasi membuat instance controller yang diminta untuk selanjutnya menangani permintaan pengguna. Controller menentukan aksi `show` merujuk pada sebuah metode bernama `actionShow` dalam kelas controller. Kemudian membuat dan menjalankan filter (contoh kontrol akses, pengukuran) terkait dengan aksi ini. Aksi dijalankan jika diizinkan oleh filter.
6. Aksi membaca **Post** model di mana ID adalah 1 dari database.
7. Aksi menyiapkan **view(tampilan)** bernama `show` dengan model `Post`.
8. View membaca dan menampilkan atribut model `Post`.
9. View menjalankan beberapa **widget**.
10. View menyiapkan hasil yang dipasangkan dalam **layout(tata letak)**.
11. Aksi mengakhiri pembuatan **view** dan menampilkan hasil akhir kepada pengguna.

#### 2.2.4 Model

Kelas Model dalam aplikasi Yii-based diletakkan dalam folder `protected/models/`. File-file model tersebut disimpan dengan nama file berekstensi `.php`.

Framework Yii model menggunakan kelas-kelas perluasan dari kelas **CModel**, juga digunakan oleh sub kelas **CFormModel**, dan **CActiveRecord**. Kelas **CActiveRecord** digunakan operasi rekaman dalam tabel, dan **CFormModel** hanya digunakan untuk proses yang menghasilkan tampilan dalam format HTML.

contoh:

```
class NamaKelas extend CFormModel {
//atribut
```

```

public $atribut_1;
public $atribut_2;
public $atribut_3;
.....
//metode
public function rules()
{
    .....
}

public function attributeLabels()
{
    .....
}
}

```

### 2.2.5 View

View (tampilan) adalah skrip PHP yang berisi terutama elemen antar muka pengguna. View bisa saja berisi code PHP, tetapi direkomendasikan code ini tidak mengubah model data dan harus tetap relatif sederhana. Guna mempertahankan semangat pemisahan logika dan penampilan, bagian besar dari logika harus ditempatkan dalam controller ataupun model alih-alih view.

View memiliki nama yang dipakai untuk mengidentifikasi file skrip tampilan saat me-render. Nama view sama seperti nama file skrip view-nya. Sebagai contoh, file view edit merujuk pada file skrip bernama edit.php. Untuk me-render view, menggunakan metode **CController::render()** dengan menyertakan nama view. Metode tersebut akan mencari file view terkait di dalam direktori **protected/views/ControllerID**.

Di dalam skrip view, kita dapat mengakses instance controller menggunakan **\$this**. Selanjutnya kita bisa menarik setiap properti controller dengan mengevaluasi **\$this->propertyName** dalam tampilan.

Cara pemanggilan view seperti pada perintah berikut:

```

$this->render('namaView', array(
    'var1'=>$value1,
    'var2'=>$value2,
));

```

Dalam contoh di atas, metode **render ()** akan mengurai parameter array kedua ke dalam variabel. Hasilnya, dalam skrip view kita dapat mengakses variabel lokal `$var1` dan `$var2`.

### 2.2.6 Controller

Sebuah controller adalah instance dari **CController** atau kelas turunan **CController**. **Controller** dibuat oleh aplikasi saat pengguna meminta. Ketika berjalan, controller melakukan action yang diminta yang biasanya memerlukan model dan membuat view yang sesuai. Sebuah action, dalam bentuk paling sederhana sebenarnya hanyalah metode kelas controller yang namanya dimulai dengan kata **action**.

Controller memiliki action standar. Ketika permintaan pengguna tidak menetapkan action mana yang dijalankan, action standar yang akan dijalankan. Biasanya action default dinamai sebagai `index`. Action default bisa diubah dengan mengeset variabel instance publik, **CController::defaultAction**.

Berikut merupakan code untuk mendefinisikan controller **site**, sebuah action `index` (**action default**), dan sebuah action `contact`:

```
class SiteController extends CController
{
    public function actionIndex()
    {
        // ...
    }

    public function actionContact()
    {
        // ...
    }
}
```

### 2.2.7 Bekerja dengan Database

Yii menyediakan dukungan berkemampuan untuk pemrograman database.

Dibangun di atas *extension PHP Data Objects (PDO)*, *Yii Data Access Objects (DAO)* memungkinkan pengaksesan ke sistem manajemen database (*DBMS*) yang berbeda dalam satu antar muka tunggal yang seragam. Aplikasi yang dikembangkan

menggunakan Yii DAO dapat dialihkan dengan mudah ke *DBMS* berbeda tanpa perlu memodifikasi data pengaksesan code.

Yii Query Builder menyediakan sebuah method berorientasi objek untuk membuat query SQL, yang bisa mengurangi resiko terserang *SQL injection*.

Kelas ActiveRecord Yii, diimplementasikan sebagai pendekatan Pemetaan Relasional-Obyek / *Object-Relational Mapping (ORM)* yang diadopsi secara luas, mempermudah pemrograman database. Tabel direpresentasikan dalam bentuk kelas dan baris dalam bentuk instance, Yii ActiveRecord mengeliminasi tugas berulang pada penulisan SQL statement terutama yang berkaitan dengan operasi *CRUD (create, read, update dan delete)*.

Meskipun Yii menyertakan fitur-fitur database yang dapat menangani hampir semua tugas-tugas terkait-database, Anda masih bisa menggunakan pustaka database Anda sendiri dalam aplikasi Yii Anda. Bahkan, Yii framework didesain secara hati-hati agar bisa dipakai bersamaan dengan pustaka pihak ketiga lainnya.

## 2.2.8 Koneksi ke Database

ActiveRecord bergantung pada koneksi database untuk melaksanakan operasi terkait database. Secara default, ActiveRecord menganggap bahwa komponen aplikasi db adalah turunan `CDbConnection` yang dibutuhkan untuk bertindak sebagai koneksi database. Konfigurasi aplikasi berikut memperlihatkan sebuah contoh:

protected/config/main.php

```
.....
'db'=>array(
'connectionString' => 'mysql:host=localhost;dbname=namaDB',
    'emulatePrepare' => true,
    'username' => 'root',
    'password' => '',
    'charset' => 'utf8',
),
...

```

Selanjutnya dapat mengakses koneksi database menggunakan perintah `Yii::app()->db` yang sudah diaktifkan secara otomatis, kecuali dikonfigurasi secara eksplisit `CDbConnection::autoConnect` menjadi false.

```
$connection=Yii::app()->db;
```

asumsi menggunakan koneksi menggunakan nama konfigurasi "db". Jika tidak, untuk membuat sebuah koneksi secara eksplisit seperti ditunjukkan pada kode berikut:



```
$connection=new CDbConnection($dsn,$username,$password);
```

atau

```
$connection = Yii::app()->db->createCommand();
$connection->select  ='judul, isi';
$connection->from    ='page';
$connection->where   ='live=1';
$connection->order  ='tanggal_publicasi DESC';
$connection->limit  ='1';
$hasil=$$connection->query();
```

**Menjalankan Query Builder :**

Yii Query Builder menyediakan cara berorientasi objek dalam menulis statement SQL

```
$user = Yii::app()->db->createCommand()
->select('id, username, profile')
->from('tbl_user u')
->join('tbl_profile p', 'u.id=p.user_id')
->where('id=:id', array(':id'=>$id))
->queryRow();
```