

Pembelajaran PHP dengan Codeigniter Berbasis Project Based Learning

Syahbaniar Rofiah ^{1,*}

¹ Manajemen Informatika; STMIK Bina Insani; Jl. Siliwangi No 6 Rawa Panjang Bekasi Timur 17114 Indonesia, Telp. (021) 824 36 886 / (021) 824 36 996. Fax. (021) 824 009 24; e-mail: rsyahbaniar@gmail.com

* Korespondensi: e-mail: rsyahbaniar@gmail.com

Diterima: 7 Nopember 2018; Review: 21 Nopember 2018; Disetujui: 5 Desember 2018

Cara sitasi: Rofiah S. 2018. Pembelajaran Pemrograman PHP dengan Codeigniter Berbasis Project Based Learning. Bina Insani ICT Journal. 5 (2): 183 – 192.

Abstrak: Pembelajaran merupakan cara untuk memahami bagaimana materi tersampaikan dengan baik diantara kedua pihak sebagai contoh dari dosen kepada mahasiswa. Pembelajaran yang baik ketika mahasiswa memahami materi yang dijelaskan oleh Dosen dapat menjelaskan kembali. Belajar saat ini sangat mudah hanya dengan koneksi dengan internet materi yang dicari akan didapat dengan cepat. Sehingga membutuhkan sumber daya manusia yang kreatif, handal dan penerapan metode penyampaian materi yang efektif. Metode penelitian hanya berfokus pada *project based learning* pembelajaran yang dilakukan dengan mengaplikasikan materi kedalam suatu project yang telah ditentukan. Pembelajaran menggunakan code igniter merupakan framework PHP yang mudah digunakan baik pemula maupun yang mahir. Hasil dari penelitian ini dengan menerapkan pembelajaran menggunakan pemrograman PHP dengan Codeigniter berbasis *project based learning* mampu membuat belajar lebih kreatif dan mudah dipahami.

Kata kunci: codeigniter, pembelajaran, pemrograman, php, project based learning

Abstract: Learning is a way to understand how the material is conveyed well between the two parties as an example from lecturers to students. Good learning when students understand the material explained by lecturers can explain again. Learning now is very easy just by connecting to the internet the material sought will be obtained quickly. So it requires creative, reliable human resources and the application of effective material delivery methods. The research method only focuses on *project based learning* that is done by applying the material into a predetermined project. Learning using a code igniter is a PHP framework that is easy to use for both beginners and advanced. The results of this study by applying learning using PHP programming with Codeigniter based on *project based learning* can make learning more creative and easy to understand.

Keywords: codeigniter, learning, programming, php, project based learning

1. Pendahuluan

Pembelajaran merupakan cara untuk memahami bagaimana materi tersampaikan dengan baik diantara kedua pihak sebagai contoh dari dosen kepada mahasiswa. Pembelajaran yang baik ketika mahasiswa memahami materi yang dijelaskan oleh Dosen. Belajar saat ini sangat mudah hanya dengan koneksi dengan internet materi yang dicari akan didapat dengan cepat. Pembelajaran bertujuan agar memperoleh berbagai pengalaman yang meliputi pengalaman pengetahuan, pengalaman keterampilan, dan pengalaman nilai yang berfungsi sebagai pengendali sikap dan perilaku menjadi bertambah, baik kuantitas maupun kualitasnya [Titu, 2015].

Pemrograman adalah suatu kegiatan menuliskan kumpulan urutan perintah kekomputer untuk mengerjakan sesuatu, dimana instruksi tersebut menggunakan bahasa yang dimengerti oleh komputer atau dikenal dengan bahasa pemrograman [Wibawanto, 2017]. Bahasa komputer yang digunakan dalam menulis program merupakan bahasa pemrograman [Munthe and Suryadi, 2018]. PHP adalah singkatan dari PHP *Hypertext Preprocessing*, merupakan bahasa scripting untuk web yang cukup populer dan dapat ditanam atau disisipkan ke dalam HTML [EMS, 2016]. PHP bukanlah pemrograman murni berbasis objek seperti java. OOP dalam PHP merupakan fitur tambahan yang mulai ada pada versi 4 [Abdulloh, 2017].

Object Oriented Programming (OOP) merupakan teknik pemrograman dengan menggunakan konsep objek. Tujuan dari OOP adalah memudahkan programmer dalam pembuatan program dengan menggunakan objek yang ada dalam kehidupan sehari-hari. Jadi setiap bagian permasalahan seperti objek, dan objek itu sendiri merupakan gabungan dari beberapa objek yang lebih kecil [Abdulloh, 2017].

Tabel 1. Sejarah Codeigniter

Tahun	Kepemilikan
Pertama kali rilis 28 Februari 2006	Ellis Lab
Oktober 2014-Sampai Sekarang	British Columbia Institute of Technology (BCIT)

Sumber: [Subagia, 2018]

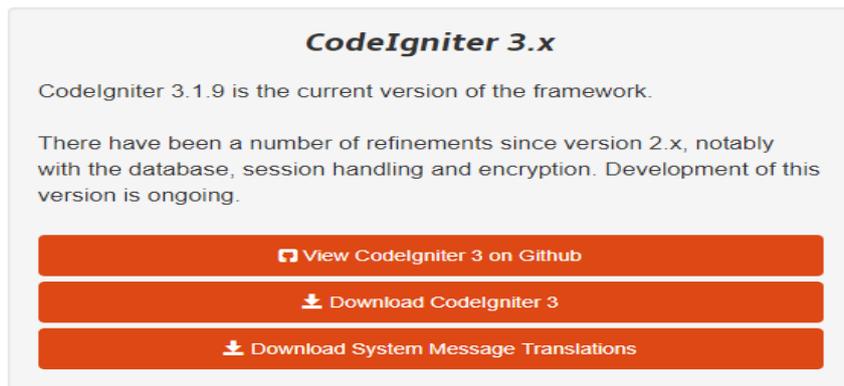
Sejarah codeigniter telah dijelaskan pada tabel 1, pertama kali dirilis tahun 28 Februari 2006 ditulis oleh Ellis Lab versi stabil 2.1.0. Kemudian diambil alih dan dikembangkan British Columbia Institute of Technology (BCIT) yang merupakan Sekolah Tinggi Teknologi di Kanada pada bulan Oktober 2014 sampai dengan sekarang versi 3.1.9. sedangkan keunggulan yang terdapat pada codeigniter disajikan pada tabel 2.

Tabel 2. Keunggulan Codeigniter

Keunggulan	Keterangan
<i>Free</i>	Berada dibawah lisensi <i>open source</i> dapat diakses gratis tanpa harus membayar
<i>Line Weight</i>	Sistem inti codeigniter memerlukan <i>library</i> yang sedikit
<i>Fast</i>	Performa terbukti cepat setelah dibandingkan dengan framework lainnya
Menggunakan Kaidah MVC	Dengan menggunakan <i>Model-View-Controller</i> dapat dipisahkan bagian logic dan presentation dari aplikasi yang dibangun
Menghasilkan <i>URL</i> yang Bersih	<i>URL</i> yang dihasilkan codeigniter bersih dan ramah terhadap <i>search engine</i>
<i>Pack a Punch</i>	Hadir dengan berbagai <i>library</i> yang membantu dalam pengembangan web seperti mengakses database, mengirim email, validasi data dari form, mengelola session dan lain-lain
<i>Extensible</i>	Dapat menambahkan <i>library</i> dan helper yang diciptakan sendiri dalam codeigniter
<i>Thoroughly Documented</i>	Hampir semua fitur, <i>library</i> dan helper terdokumentasi dengan lengkap dan baik
Mempunyai Komunitas yang Ramah	Memudahkan para pengguna baik pemula maupun yang mahir untuk saling berbagi ilmu pengetahuan

Sumber: [Subagia, 2018]

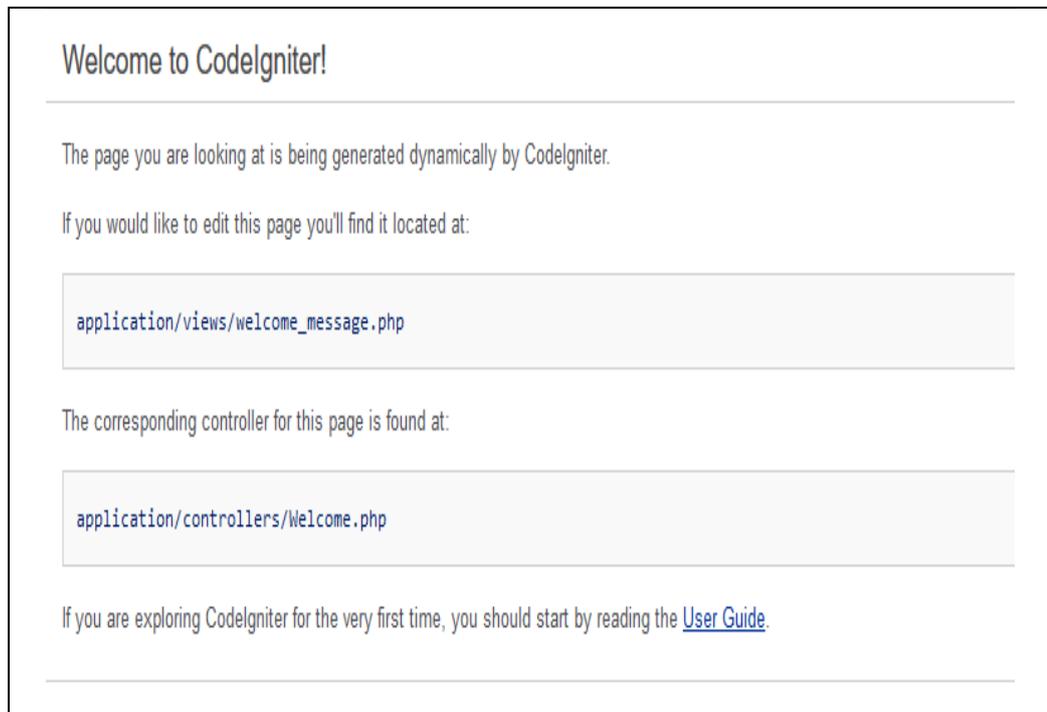
Sedangkan Codeigniter adalah framework PHP yang bisa digunakan untuk menyederhanakan script dan mengoptimalkan fungsi PHP [Enterprise, 2015]. Dalam instalasi codeigniter sangat mudah, sebelumnya kunjungi situs <https://codeigniter.com/download> untuk mendownload *codeigniter* setelah berhasil didownload instalasi *codeigniter* dapat dilakukan. Caranya dengan menyimpan satu folder yang telah didownload kedalam *webserver* yang ada seperti xampp contoh penyimpanan yang benar <http://xampp/htdocs/codeigniter>. Pada saat dijalankan di web browser ketikkan localhost/codeigniter akan tampil sesuai dengan gambar 1.



Sumber: <https://codeigniter.com/download>

Gambar 1. Halaman Download Codeigniter 3

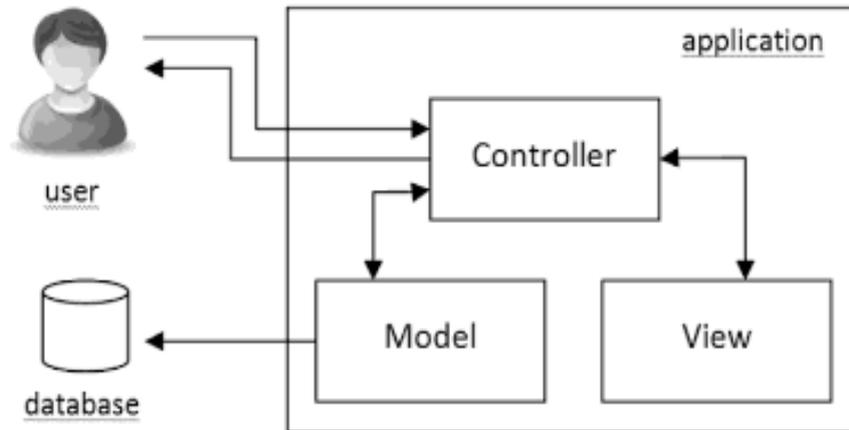
Pada gambar 2 merupakan tampilan awal index.php pada saat pertama kali dijalankan pada *web browser* terdapat keterangan welcome to codeigniter yang berarti instalasi codeigniter telah berhasil dan dapat digunakan.



Sumber: Hasil Penelitian (2018)

Gambar 2. Halaman Index Codeigniter

Pada gambar 3 merupakan cara kerja *Model, View, Controller* (MVC) yang pertama model, menggambarkan struktur data yang berisi kelas model berguna untuk mengambil, menambah dan memperbaharui yang ada didatabase, yang kedua view, informasi yang diperlihatkan kepada user termasuk HTML,CSS dan Javascript, yang ketiga controller, perantara model, view dan resource lainnya yang dibutuhkan untuk menangani HTTP request dan menghasilkan halaman web. Keuntungan dalam menggunakan MVC akan dijelaskan pada tabel 3.



Sumber: [Subagia, 2018]

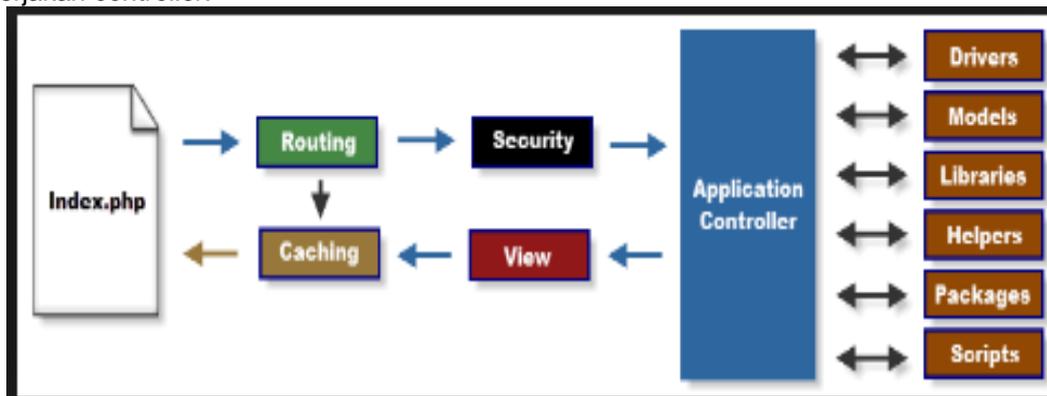
Gambar 3. Cara Kerja MVC

Tabel 3. Keuntungan Menggunakan MVC

Keuntungan
Operasi data hanya dilakukan pada area yang ditentukan (model) tidak tersebar dalam keseluruhan aplikasi, sehingga memudahkan dalam proses pemeliharaan aplikasi
Memudahkan penggabungan divisi desain dalam tim development
Komponen view dapat didesain tanpa harus memperhatikan bagian lain secara berlebihan

Sumber: [Abdulloh, 2017]

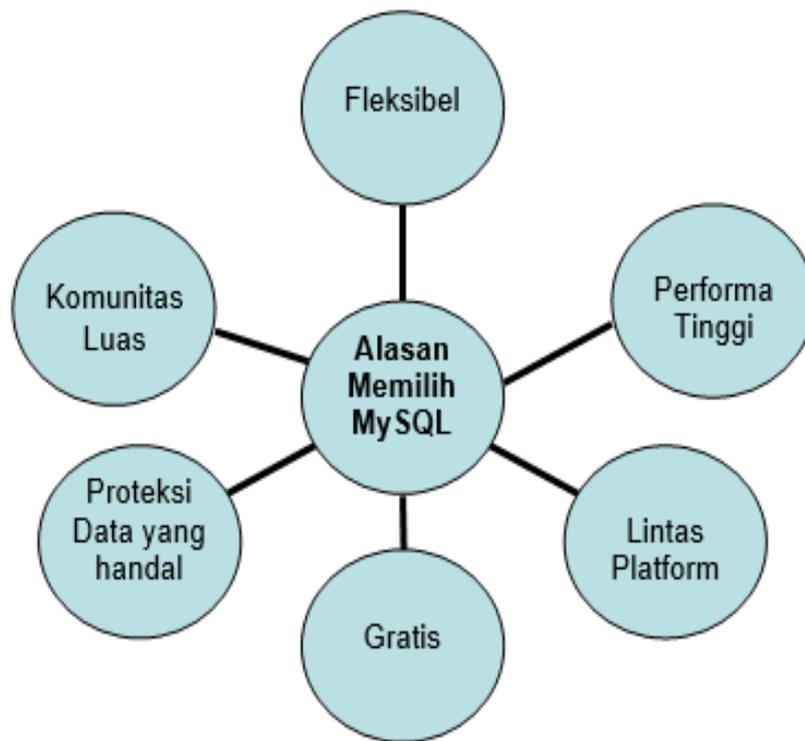
Pada gambar 4 merupakan cara kerja dari codeigniter yang pertama `index.php` yang bertindak sebagai controller dan menginisiasi *resource* yang diperlukan pada codeigniter. Yang kedua *router* bertindak memeriksa HTTP *request* untuk menentukan apa yang harus dikerjakan. Yang ketiga *chace* jika ada akan ditampilkan langsung. Yang keempat sebelum memuat controller HTTP *request* memeriksa yang disubmit *user* dan memfilter untuk keamanan. Yang kelima *controller* memuat model, *core libraries*, *plugin*, *helper* dan *resource* lainnya untuk memproses permintaan tertentu. Yang keenam *view*, tampilan di browser sesuai proses yang dikerjakan *controller*.



Sumber: [Subagia, 2018]

Gambar 4. Cara Kerja Codeigniter

MySQL merupakan *software Database Management System* (RDBMS) yang dapat mengelola *database* yang sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak user (*multiuser*) dan dapat melakukan suatu proses secara sinkronisasi atau berbarengan (*multi-threaded*) [Raharjo, 2011].



Sumber: [Raharjo, 2011]

Gambar 5. Alasan Memilih *MySQL*

Pada gambar 5 menjelaskan alasan mengapa memilih *MySQL* yang pertama fleksibel, *MySQL* dapat digunakan untuk mengembangkan aplikasi *desktop* maupun aplikasi web dengan menggunakan teknologi yang bervariasi seperti PHP, JSP, Java, Delphi dan yang lainnya. Yang kedua performa tinggi, *MySQL* memiliki mesin query dengan performa tinggi dengan demikian proses transaksional dapat dilakukan dengan sangat cepat. Yang ketiga lintas *platform*, *MySQL* dapat digunakan pada *platform* atau lingkungan (dalam hal ini sistem operasi) yang beragam, bisa Microsoft Windows, Linux atau UNIX. Yang keempat gratis, *MySQL* dapat digunakan secara gratis, meskipun demikian ada beberapa yang bersifat komersial seperti ada kemampuan spesifik dan mendapatkan pelayanan dari technical suport. Yang kelima proteksi data yang handal, perlindungan terhadap keamanan data merupakan hal nomor satu yang dilakukan oleh para profesional dibidang *database*, *MySQL* menyediakan mekanisme yang powerful untuk menangani hal tersebut dengan menyediakan fasilitas manajemen *user*, enkripsi data, dan lain sebagainya. Dan yang keenam komunitas yang luas, karena penggunanya banyak maka *MySQL* memiliki komunitas luas sehingga jika memiliki kendala atau permasalahan dalam menggunakan *MySQL* dapat berdiskusi dan berharap mendapatkan solusi akan permasalahannya.

Database didefinisikan sebagai kumpulan data yang terintegrasi dan diatur sedemikian rupa sehingga data tersebut dapat dimanipulasi dan dicari secara cepat [Raharjo, 2011]. Dalam membuat aplikasi dibutuhkan interface yang menghubungkan ke *database*. Aplikasi yang baik yaitu dapat diintegrasikan dan dihubungkan kedalam *database*.

Pada penelitian sebelumnya, Metode *learning based project* menggunakan Alice juga mempermudah pembelajaran karena berbasis animasi dengan sedikit kode pemrograman. Metode belajar berbasis proyek (*project based learning*) juga membuat lebih aktif, kreatif dan berhasil memecahkan masalah dengan algoritma yang baik dan benar [Retnoningsih et al., 2017]. Salah satu macam kerangka kerja pemrograman web berbasis bahasa pemrograman PHP dan menggunakan pendekatan Model-View-Controller (MVC) adalah CodeIgniter yang dikembangkan oleh EllisLab, Inc [Gani, 2018].

Tabel 4. Data Tes Hasil Belajar

No	Tingkat Ketuntasan	Banyak Siswa	Presentasi Jumlah Siswa
1	Tidak Tuntas	7	17,5%
2	Tuntas	33	82,5%
	Jumlah	40	100%

Sumber: [Novita et al., 2016]

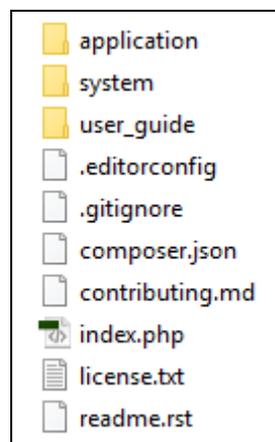
Dari tabel 4 menjelaskan bahwa sebesar 82,5% siswa tuntas dengan KKM 75 sedangkan 17,5% lainnya belum dinyatakan tuntas pada penelitian pengembangan LKS berbasis *project based learning* dan hasil yang didapat dari segi konstruk, sesuai dengan prinsip *project based learning* yakni petunjuk-petunjuk yang mengarahkan siswa melaksanakan proyek untuk mencapai tujuan pembelajaran [Novita et al., 2016].

2. Metode Penelitian

Metode penelitian hanya berfokus pada *project based learning* pembelajaran yang dilakukan dengan mengaplikasikan materi kedalam suatu *project* yang telah ditentukan. Pembelajaran menggunakan code igniter merupakan *framework* PHP yang mudah digunakan untuk pemula. Dan codeigniter merupakan pemrograman berorientasi objek atau *Object Oriented Programming* (OOP) yang memberikan kemudahan dalam pembuatan sebuah program, karena pemrograman OOP sudah menggunakan konsep modularitas object dan class [Retnoningsih et al., 2017].

3. Hasil dan Pembahasan

Pada penelitian ini berfokus pada pembelajaran pemrograman PHP dengan framework codeigniter berbasis *project based learning*. Codeigniter mempunyai konsep MVC yang memudahkan dalam memperbaharui, memodifikasi dan menghapus data sesuai kebutuhan dan dapat menambahkan *header* dan *footer*, *web editor* yang digunakan yaitu dreamweaver CS6 sedangkan *Webserver* menggunakan Xampp versi 3.2.1. Untuk webbrowser menggunakan google chrome dan Mozilla Firefox. Sedangkan folder utama yang terdapat pada codeigniter versi 3.1.9 ada 3 folder disajikan pada gambar 6.



Sumber: Hasil Penelitian (2018)

Gambar 6. Struktur Folder Codeigniter versi 3.1.9

Di folder utama yang terdapat pada codeigniter versi 3 diantaranya *application*, *system* dan *user_guide*. Yang pertama folder *application* berisi *chase*, *config*, *controller*, *core*, *helpers*, *hooks*, *language*, *libraries*, *logs*, *models*, *third_party*, dan *views*. Yang pertama folder *application* merupakan folder yang akan dikembangkan sesuai dengan *project* yang dibuat. Yang kedua folder *system* yang berisi kode-kode helper, library dan lain-lain yang merupakan core dari codeigniter. Sedangkan folder yang ketiga yaitu *user_guide* merupakan informasi untuk cara penggunaan dari setiap folder yang terdapat pada codeigniter. Pada penelitian ini *project* yang

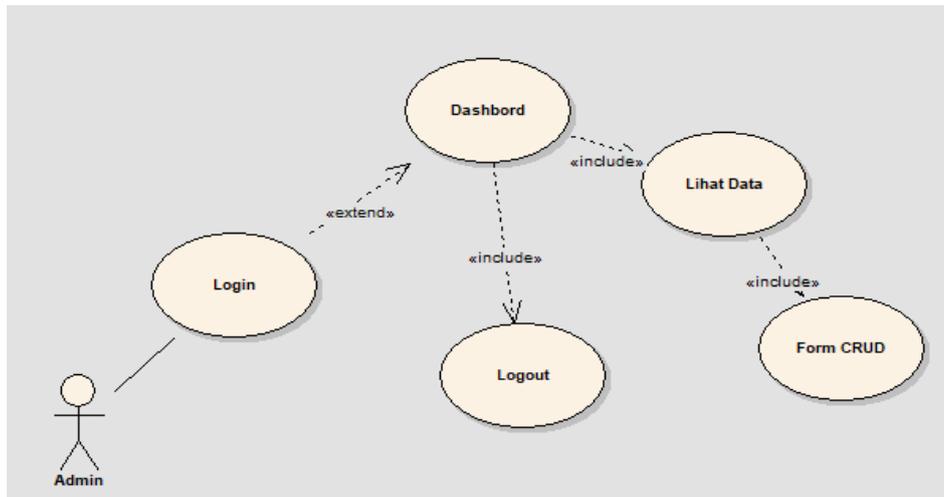
dibuat tentang cara membuat login, menambah data, merubah data dan menghapus data dari database. Nama database yang digunakan yaitu db_sekolah dan terdapat 2 tabel diantaranya tabel admin dan tabel *user*.

Tabel 5. Tabel Siswa

Nama Field	Type Data	Keterangan
no_induk	Int (6)	Auto Increment, Primary Key
nama	Varchar (25)	
alamat	Varchar (50)	

Sumber: Hasil Penelitian (2018)

Pada tabel 5 menjelaskan struktur tabel dari salah satu tabel dalam project yaitu tabel siswa. Tabel tersebut yang menghubungkan antara *form* dengan tabel.



Sumber: Hasil Penelitian (2018)

Gambar 7. Use Case Halaman *Login*

Gambar 7 merupakan *use case* halaman login, yang pertama admin mengisi *username* dan *password* jika tidak berhasil admin tidak dapat mengakses halaman dashbord dan tidak dapat mengelola data siswa.

Sumber: Hasil Penelitian (2018)

Gambar 8. Halaman *Login*

Pada halaman *login* admin dapat mengakses halaman admin jika telah terdaftar sebagai admin didatabase. Admin yang telah terdaftar terlebih dahulu mengisi *username* dan *password* yang telah didaftarkan. Tahapan pertama dalam pembuatan login pada codeigniter dimulai dari membuat tabel admin yang terdiri dari struktur tabel diantaranya *id*, *username* dan *password*. Tahap kedua dengan merubah *code* pada database.php yang ada pada folder *application* dan didalamnya terdapat folder config kemudian buka file database.php seperti pada gambar 7.

```

<!DOCTYPE html>
<html>
<head>
    <title>Login</title>
</head>
<body bgcolor="#66FF33">
    <center><h1>Membuat Login Dengan CodeIgniter <br/> </h1>
    <form action="<?php echo site_url('welcome/aksi_login'); ?>" method="post">
        <table>
            <tr>
                <td>Username</td>
                <td><input type="text" name="username"></td>
            </tr>
            <tr>
                <td>Password</td>
                <td><input type="password" name="password"></td>
            </tr>
            <tr>
                <td></td>
                <td><input type="submit" value="Login"></td>
            </tr>
        </table>
    </form>
</center>
</body>
</html>

```

Sumber: Hasil Penelitian (2018)

Gambar 9. Source Code Login

Pada gambar 9 merupakan *source code* dari login yang ditampilkan pada halaman *browser* dengan konsep *model view control* (MVC) di codeigniter *source code* tersebut disimpan pada folder *application/views* folder *views* merupakan tampilan yang terlihat pada *browser*.

```

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

class App_model extends CI_Model {

    function cek_login($table,$where)
    {
        return $this->db->get_where($table,$where);
    }
}

```

Sumber: Hasil Penelitian (2018)

Gambar 10. Source Code App_Model

Pada gambar 10 merupakan *source code* dari login yang diproses pada folder *application/moduls*, *source code* tersebut disimpan kedalam folder *moduls*. Kemudian dari *source code* *app_modul* tersebut dikirimkan ke folder *views* kemudian diproses oleh folder *controller*.

```

8 $db['default'] = array(
9     'dsn' => '',
10    'hostname' => 'localhost',
11    'username' => 'root',
12    'password' => '',
13    'database' => 'db_sekolah',
14    'dbdriver' => 'mysqli',
15    'dbprefix' => '',
16    'pconnect' => FALSE,
17    'db_debug' => (ENVIRONMENT !== 'production'),
18    'cache_on' => FALSE,
19    'cachedir' => ''

```

Sumber: Hasil Penelitian (2018)

Gambar 11. Code Database.php

Pada gambar 11 berisi *code* untuk menjalankan database dengan mengatur *hostname*, *username*, *password*, dan yang wajib diubah adalah database diisi menyesuaikan dengan nama database yang dibuat, kemudian simpan kembali tanpa merubah nama file tersebut. Tahapan kedua mengatur file *autoload.php* yang berada folder *application/config/autoload.php*. *file* tersebut yang diubah sesuai dengan gambar 12 dapat juga diubah sesuai dengan *libraries* yang dimiliki oleh pengembang web pada *codeigniter*.

```

$autoload['libraries'] = array('database'); = isi sesuai yg ditandai warna merah
$autoload['helper'] = array('url'); = isi sesuai yg ditandai warna merah
$autoload['model'] = array('app_model'); = isi sesuai yg ditandai warna merah

```

Sumber: Hasil Penelitian (2018)

Gambar 12. Pengaturan Autoload.php

Pada gambar 12 menjelaskan *code* yang diubah seperti *libraries*, *helper* dan *model* sesuai dengan *file* yang dibuat. Setiap pengaturan yang telah dibuat akan berpengaruh terhadap halaman pada *codeigniter*. *File* *autoload.php* berfungsi memanggil *model-model* yang sering digunakan secara otomatis akan terpanggil oleh *core system* sehingga memudahkan pengguna. Tidak menutup kemungkinan jika banyak yang diubah sesuai dengan *file* yang dimiliki. Langkah selanjutnya jika *username* dan *password* sesuai maka akan tampil halaman *dashbord* sesuai dengan gambar 13.

Login berhasil !

Hai, niar

- [Lihat Data](#)
- [Logout](#)

Sumber: Hasil Penelitian (2018)

Gambar 13. Halaman *Dashbord*

Pada halaman *dashbord* admin akan tampil keterangan login berhasil dan terlihat nama admin yang login dan admin dapat melihat data dan logout dari halaman *dashbord*.

Membuat CRUD dengan CodeIgniter

Tambah Data

No	No Induk	Nama	Alamat	Action
1	19010801	Rizki Hidayatulloh	Bekasi Utara	Edit Hapus
2	19010802	Nurhasanah	Bekasi Timur	Edit Hapus
3	19010803	Khoirunnisa	Tambun Utara	Edit Hapus
4	19010804	Siti Khodijah	Bekasi	Edit Hapus
5	19010805	Muhammad Ridwan	Jakarta Timur	Edit Hapus
6	19010806	Isyana Saraswati	Jakarta Selatan	Edit Hapus

Sumber: Hasil Penelitian (2018)

Gambar 14. Halaman *CRUD*

Pada halaman *CRUD* admin dapat mengelola data seperti merubah, menghapus dan melihat data yang telah tersimpan didatabase. Setiap *field* dapat diubah kecuali no induk siswa karena nomor induk siswa merupakan kunci utama yang mewakili data setiap tabel.

Sumber: Hasil Penelitian (2018)

Gambar 15. Halaman Edit Data

Pada halaman edit data admin dapat merubah data nama siswa dan alamat siswa kemudian diperbaharui sedangkan nomor induk siswa tidak dapat diubah karena nomor induk merupakan kunci utama.

4. Kesimpulan

Hasil dari penelitian ini dengan menerapkan pembelajaran menggunakan pemrograman PHP dengan Codeigniter berbasis *project based learning* mampu membuat belajar lebih kreatif dan mudah dipahami. Sehingga dalam belajar siswa dapat mengembangkan materi yang telah disampaikan oleh pengajar dan tercipta pembelajaran yang interaktif. Diharapkan untuk penelitian yang selanjutnya dapat dikembangkan dengan codeigniter menggunakan bootstrap atau ajax.

Referensi

- Abdulloh R. 2017. Membuat Toko Online dengan Teknik OOP, MVC dan AJAX. Jakarta: PT. Elex Media Komputindo.
- EMS T. 2016. PHP 5 dari Nol. Jakarta: PT. Elex Media Komputindo.
- Enterprise J. 2015. Membuat Website PHP dengan Codeigniter. Jakarta: PT. Elex Media Komputindo.
- Gani AG. 2018. Analisis Sistem Informasi Pengelolaan Data Alumni Berbasis Codeigniter PHP Framework. J. Sist. Inf. Univ. Suryadarma 5: 76–98.
- Munthe IR, Suryadi S. 2018. Bahasa Pemrograman Pascal, 1e. Surabaya: Uwais Inspirasi Indonesia.
- Novita D, Darmawijoyo, Aisyah N. 2016. Pengembangan LKS Berbasis Project Based Learning untuk Pembelajaran Materi Segitiga di Kelas VII. J. Pendidik. Mat. 10: 1–12.
- Raharjo B. 2011. Belajar Otodidak Membuat Database Menggunakan MySQL. Bandung: Informatika Bandung. 3 p.
- Retnoningsih E, Shadiq J, Oscar D. 2017. Pembelajaran Pemrograman Berorientasi Objek (Object Oriented Programming) Berbasis Project Based Learning. INFORMATICS Educ. Prof. 2: 95–104.
- Subagia A. 2018. Kolaborasi Codeigniter dan Ajax dalam Perancangan CMS. Jakarta: PT. Elex Media Komputindo.
- Titu MA. 2015. Penerapan Model Pembelajaran Berbasis Proyek untuk Meningkatkan Kreatifitas Siswa pada Materi Konsep Masalah Ekonomi. Pros. Semin. Nas.: 176–186.
- Wibawanto W. 2017. Desain Dan Program Multimedia Pembelajaran Interaktif. Jember: Cerdas Ulet Kreatif. 1-5 p.