

**BUKU TUTORIAL PEMBUATAN APLIKASI REGISTRASI ONLINE
MENGGUNAKAN ALGORITMA BOYER MOORE**

**“IMPLEMENTASI BOYER MOORE PADA APLIKASI
PENDAFTARAN KUNJUNGAN ONLINE WARGA BINA
PEMASYARAKATAN GUNA UNTUK MENINGKATKAN
KEAMANAN RUTAN KLAS 1 BANDUNG”**

Buku ini dibuat untuk memenuhi persyaratan kelulusan
matakuliah Program Internship I



**Dibuat Oleh,
1.16.4.058 Wildan Khaustara W**

**PROGRAM DIPLOMA IV TEKNIK INFORMATIKA
POLITEKNIK POS INDONESIA
BANDUNG
2019**

TUTORIAL APLIKASI REGISTRASI ONLINE MENGGUNAKAN ALGORITMA BOYER MOORE



KATA PENGANTAR

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan laporan praktek kerja lapangan dengan judul ” Implementasi *Boyer Moore* Pada Aplikasi Pendaftaran Kunjungan Online WBP Guna Untuk Meningkatkan Keamanan Rutan Klas 1 Bandung”. Penulisan laporan Internship I ini dan penulis menyadari bahwa penyusunan laporan ini masih belum sempurna. Ini mengingat keterbatasan pengetahuan, pengalaman serta kemampuan penulis. Penulis megharapkan kritik dan saran yang sifatnya membangun dari pembaca. Untuk itu penulis mengucapkan terima kasih kepada:

1. Allah S.W.T yang telah melimpahkan karunia-Nya sehingga kami dapat menyelesaikan tugas ini.
2. Kedua orang tua dan keluarga saya yang telah memberi motivasi kepada saya.
3. Bapak Syaeful Bachri selaku pembimbing eksternal di Rutan Klas 1 Bandung.
4. Ibu Woro Isti Rahayu, S.T., M.T. selaku dosen pembimbing internship 1.
5. Ibu Nisa Hanum Harani, S.T., M.T.I selaku Koordinator Internship I.
6. Bapak M. Yusril Helmi Setyawan, S.Kom., M.Kom. selaku Ketua Prodi D4 Teknik Informatika.
7. pihak Rumah Tahanan Negara Klas 1 Bandung Kebonwaru yang telah banyak membantu dalam usaha memperoleh data yang saya perlukan.

Akhir kata, penulis berharap Tuhan Yang Maha Esa berkenan membala segala kebaikan semua pihak yang telah membantu. Semoga buku ini membawa manfaat bagi pengembangan ilmu.

Bandung, 17 Januari 2020

Penulis

DAFTAR ISI

DAFTAR ISI.....	4
BAB I DASAR	1
1.1 PENGENALAN WEB PROGRAMMING	1
1.1.1 Pengantar Web Programming	1
1.1.2 HTML	2
1.1.3 BASIS DATA	9
3.1.2 SEJARAH	14
3.1.3 BAHASA PEMROGRAMAN.....	18
3.1.4 Macam Macam Framework	46
BAB II PENGENALAN ALGORITMA.....	53
1.1 PENGENALAN ALGORITMA BOYER MOORE STRING MATHCING	53
1.1.1 Pengertian String Matching	53
1.1.2 Pengantar Algoritma Boyer Moore.....	57
BAB III PENJELASAN TOOLS DAN BAHASA PEMROGRAMAN.....	65
1.1 PENJELASAN TOOLS DAN BAHASA PEMOGRAMAN YANG DIGUNAKAN	65
1.1.1 Tools Yang Digunakan	65
1.1.2 Framework codeigniter	69
BAB IV INSTALASI TOOLS YANG DIGUNAKAN.....	73
4.1 INSTALASI TOOLS YANG DIGUNAKAN	73
4.1.1 Tools Yang Digunakan	73
BAB V PERANCANGAN DAN PEMBUATAN APLIKASI	86
5.1 PERANCAGAN APLIKASI	86
5.1.1 Analisis Yang Sedang Berjalan.....	86
5.1.2 Analisis Registrasi Yang Akan Dibangun.....	87
5.1.3 Flow Chart Prosedure Kunjungan Yang Akan Dibangun.....	88
5.1.4 Use Case Diagram.....	89

5.1.5	Class Diagram.....	89
5.1.6	Component Diagram.....	90
5.1.7	Component Diagram	90
5.2	PERANCAGAN DATABASE	91
5.2.1	<i>Conceptual Data Model (CDM)</i>	91
5.2.2	<i>Physical Data Model (PDM)</i>	92
5.3	MEMPERSIAPKAN TOOLS YANG DIBUTUHKAN	92
5.3.1	Mempersiapkan Tools.....	93
5.3.2	Membuat database.....	93
5.3.3	Konfigurasi Codeigniter.....	98
5.4	KONFIGURASI CSS BOOTSRAP.....	103
5.5	MEMBANGUN HALAMAN LOGIN	147
5.5.1	Source Code MODEL Login	147
5.5.2	Source Code VIEW Login	147
5.5.3	Source Code CONTROLLER Login	150
5.6	MEMBANGUN HALAMAN DAFTAR AKUN	152
5.6.1	Source Code MODEL Daftar.....	152
5.6.2	Source Code VIEW Daftar	153
5.6.3	Source Code CONTROLLER Daftar.....	156
5.7	HALAMAN PENGISIAN FORMULIR KUNJUNGAN	158
5.7.1	Source Code MODEL Formulir Kunjungan	159
5.7.2	Source Code VIEW Formulir Kunjungan.....	159
5.7.3	Source Code CONTROLLER Formulir Kunjungan.....	172
5.8	SEND NOTIFIKASI KUNJUNGAN VIA EMAIL	190
5.8.1	Source Code MODEL Notifikasi Email	190
5.8.2	Source Code VIEW Notifikasi Email	190
5.8.3	Source Code CONTROLLER Notifikasi Email	192
5.9	HISTORY KUNJUNGAN.....	194
5.9.1	Source Code MODEL History Kunjungan	194
5.9.2	Source Code VIEW History Kunjungan	195

5.9.3	Source Code CONTROLLER History Kunjungan	196
5.10	PESAN	196
5.10.1	Source Code MODEL Pesan.....	197
5.10.2	Source Code VIEW Pesan	197
5.10.3	Source Code CONTROLLER Pesan	198
5.11	DAFTAR PENGUNJUNG	200
5.11.1	Source Code MODEL Daftar Pengunjung.....	200
5.11.2	Source Code VIEW Daftar Pengunjung	200
5.11.3	Source Code CONTROLLER Daftar Pengunjung	202
5.12	EDIT DATA PENGUNJUNG	203
5.12.1	Source Code MODEL Edit Pengunjung	203
5.12.2	Source Code VIEW Edit Pengunjung	204
5.12.3	Source Code CONTROLLER Edit Pengunjung	208
5.13	DAFTAR WBP	209
5.13.1	Source Code MODEL Daftar Data WBP.....	209
5.13.2	Source Code VIEW Daftart Data WBP	210
5.13.3	Source Code CONTROLLER Daftart Data WBP	211
5.14	EDIT DATA WBP.....	213
5.14.1	Source Code MODEL Edit Data WBP	213
5.14.2	Source Code VIEW Edit Data WBP	213
5.14.3	Source Code CONTROLLER Edit Data WBP	213
5.15	TAMBAH DATA JADWAL.....	213
5.15.1	Source Code MODEL Tambah Jadwal.....	213
5.15.2	Source Code VIEW Tambah Jadwal.....	213
5.15.3	Source Code CONTROLLER Tambah Data Jadwal	215
5.16	PENCARIAN WBP (<i>BOYER MOORE</i>)	216
5.16.1	Source Code MODEL Pencarian WBP.....	216
5.16.2	Source Code VIEW Pencarian WBP	216
5.16.3	Source Code CONTROLLER Pencarian WBP (<i>BOYER MOORE</i>) 221	

5.17 PENCARIAN KODE PENDAFTARAN (<i>BOYER MOORE</i>)	222
5.17.1 Source Code MODEL Pencarian Kode Pendaftaran.....	222
5.17.2 Source Code VIEW Pencarian Kode Pendaftaran	222
5.17.3 Source Code CONTROLLER Pencarian Kode Pendaftaran	232
5.18 PENCARIAN BERDASARKAN TANGGAL PENDAFTARAN	233
5.18.1 Source Code MODEL Pencarian tanggal pendaftaran.....	233
5.18.2 Source Code VIEW Pencarian tanggal pendaftaran	233
5.18.3 Source Code CONTROLLER Pencarian tanggal pendaftaran	242
5.19 PESAN	243
5.19.1 Source Code MODEL Pesan.....	243
5.19.2 Source Code VIEW Pesan	244
5.19.3 Source Code CONTROLLER Pesan	245
BAB VI PENERAPAN <i>BOYER MOORE</i>	247
6.1 PENERAPAN ALGORITMA PADA APLIKASI	247
6.2.2 Alasan Menggunakan Algoritma <i>Boyer moore</i>	247
Algortima Pencocokan Data WBP.....	247
6.1.2 Algoritma Pencarian Data (WBP, REGISTRASI, TANGGAL)	250
6.1.3 Hasil Pengujian	253
6.2 Kelebihan dan kekurangan Algoritma <i>Boyer moore</i>	254
6.2.1 Kelebihan	254
6.2.2 Kekurangan	255
BAB VII HASIL ANTAR MUKA APLIKASI	256
7.1 Halaman Antarmuka Admin	256
7.4.1 Halaman Login.....	256
7.4.2 Utama Admin.....	256
7.4.3 Kelola Pengunjung.....	257
7.4.4 Halaman Kelola WBP	258
7.4.5 Kelola Jadwal.....	258
7.4.6 Kelola Daftar Verifikasi.....	259
7.4.7 Cari NamaWBP	259

7.4.8 Cari No Registrasi.....	260
7.4.2 Cetak	260
7.4.2 Pemberitahuan Email	261
6.4.2 Pesan	262
7.5 Halaman Antarmuka Pengunjung	262
7.5.1 Login	262
7.5.2 Pendaftaran	262
7.5.3 Prosedur Pendaftaran	263
7.5.4 Daftar Akun	264
7.5.5 Dashboard	265
7.5.5 Formulir Kunjungan.....	265
7.5.6 Cek Pesan.....	266
BAB VIII KESIMPULAN.....	267
8.1 KESIMPULAN MASALAH.....	267
8.2 KESIMPULAN METODE	267
8.3 KESIMPULAN PENGUJIAN SISTEM.....	268

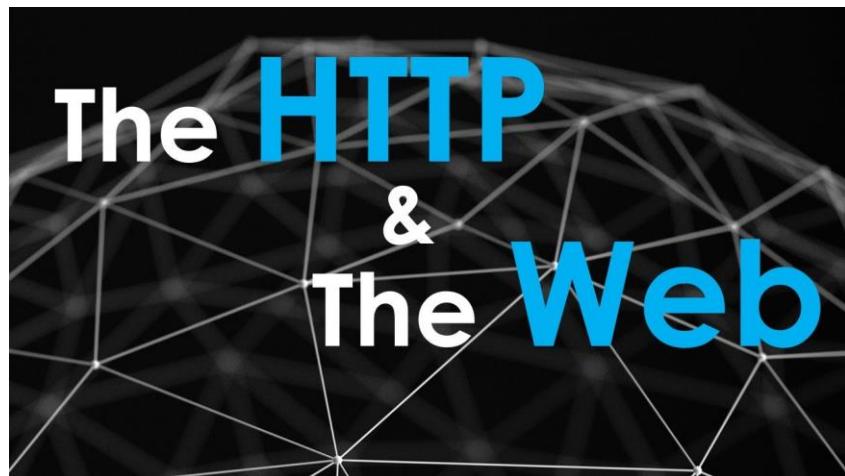
BAB I DASAR

1.1 PENGENALAN WEB PROGRAMMING

Web programming, Pada buku ini penulis akan memaparkan materi tentang teknologi tersebut dengan penerapan teknologi pembuatan aplikasi berbasis Web Programming.

1.1.1 Pengantar Web Programming

1.1.1.1 HTTP & WEB



Web Programming atau biasa disebut Pemrograman Web, Web programming terdiri dari 2 kata yaitu web dan programming, programming dapat di artikan sebagai proses pembuatan suatu program. Sedangkan web dapat di artika sebagai jaringan computer atau biasa disebut *website* yang terdiri dari situs jaringan internet yang menawarkan fitur dengan berbagai jenis seperti text, grafik, suara, Serta memelihara kode untuk membuat suatu pemrograman computer, kode ini ditulis dalam berbagai Bahasa pemrograman yang bertujuan untuk membuat suatu program yang dapat melakukan suatu perhitungan atau proses sesuai dengan keinginan pemrogram yang di akses melalui protocol HTTP.

Apa aitu HTTP? HTTP Merupakan protocol lapisan aplikasi (application layer) yang dikembangkan untuk membantu suatu proses transfer antar

computer. Protokol ini berfungsi untuk melakukan transfer dokumen, file, gambar, dan video antar computer, Protokol HTTP menyediakan kumpulan suatu perintah didalam komunikasi antar suatu jaringan. Komunikasi tersebut

Fungsi HTTP yaitu mengatur Format dan bagaimana suatu data di transmisikan. HTTP juga berfungsi untuk mengatur bagaimana web server dan web browser saling terhubung dan memproses berbagai suatu perintah yang masuk. Fungsi lain dari HTTP ialah mengamankan data dari suatu pencurian dan hacker. Hal ini ditandai dengan munculnya HTTPS (Hypertext transfer Protocol Secure).

Jadi secara singkat untuk Web Programming adalah suatu proses pembuatan program yang memiliki output disajikan oleh web browser yang diakses melalui HTTP dan Bahasa yang digunakan adalah Bahasa HTML.

1.1.2 HTML



HTML merupakan turunan atau pengembangan dari SGML (Standar Generalized Markup Language). HTML sendiri dikembangkan oleh Tim Berners-Lee sewaktu masih bekerja di CERN yang pertama kali dipopulerkan oleh browser Mosaic yang dikembangkan NCSA. Selama awal tahun 1990an, HTML semakin memiliki perkembangan yang sangat cepat. Akan tetapi pengembangan resmi HTML baru dikeluarkan oleh Internet Engineering Task Force (IETF) yang dikeluarkan pada tahun 1995. HTML2 yang dikembangkan ini merupakan turunan dari HTML+ pada tahun 1993. HTML3 yang juga

dirilis pada tahun 1995 mempunyai kemampuan yang jauh lebih bagus dari versi sebelumnya. Merupakan hasil dari usaha yang dikembangkan oleh World Wide Web Consortium's (W3C) yang kemudian menghasilkan HTML3 di tahun 1996 dan rilislah HTML4 pada akhir tahun tersebut yaitu 1997 dan 1998.

1.1.2.1 Perkembangan HTML

1. HTML Versi 1.0 HTML Versi 1.0 merupakan pionir yang di dalamnya masih terdapat banyak sekali kelemahan hingga wajar jika tampilan yang dihasilkan sangat sederhana. Kemampuan yang dimiliki versi 1.0 ini antara lain heading, paragraf, hypertext, list, serta cetak tebal dan miring pada teks. Versi ini juga mendukung peletakan image pada dokumennya tanpa memperbolehkan teks di sekelilingnya (wrapping).
2. HTML Versi 2.0 Pada HTML Versi ini, penambahan kualitas HTML terletak pada kemampuannya untuk menampilkan suatu form pada dokumen. Dengan adanya form ini, kita dapat memasukkan nama, alamat, serta saran dan kritik. HTML versi 2.0 ini merupakan pionir dari adanya web interaktif.
3. HTML Versi 3.0

Versi HTML 3.0 menambahkan beberapa fasilitas baru seperti FIGURE yang merupakan perkembangan dari IMAGE untuk meletakkan gambar dan tabel. Selain itu, HTML ini juga mendukung adanya rumus-rumus matematika dalam dokumennya. Versi ini yang disebut HTML+- tidak bertahan lama dan segera digantikan dengan versi 3.2.

4. HTML Versi 3.2

HTML versi ini merupakan HTML yang sering digunakan. Di dalamnya terdapat suatu teknologi untuk meletakkan teks di sekeliling gambar, gambar sebagai latar belakang, tabel, frame, style sheet dan lain-lain. Selain itu pada HTML versi ini Kita bisa menggunakan script di luar HTML untuk

mendukung kinerja HTML kita tersebut, seperti Javascript, VBScript dan lain-lain.

5. HTML Versi 4

HTML ini memuat banyak sekali perubahan dan revisi dari pendahulunya yaitu HTML 3.2. Perubahan ini hampir terjadi di segala perintah HTML seperti tabel, image, link, text, meta, imagemaps, form dan lain-lain.

6. HTML Versi 4.01

HTML versi 4.01 merupakan revisi dari HTML 4.0. Versi terbaru ini memperbaiki kesalahan-kesalahan kecil (minor errors) pada versi terdahulunya. HTML 4.01 ini juga menjadi standarisasi untuk elemen dan atribut dari script XHTML 1.0. 7.

7. HTML Versi 5.0

Teknologi ini mulai diluncurkan pada tahun 2009, tetapi pada tanggal 4 Maret 2010 terdapat sebuah informasi bahwa W3C (World Wide Web Consortium) dan IETF (Internet Engineering Task Force) yaitusebuah organisasi yang menangani HTML sejak versi 2.0 telah mengmbangkan versi HTML terbaru, yaitu versi 5.0. HTML 5 adalah sebuah prosedur pembuatan tampilan web baru yang merupakan penggabungan antara CSS, HTML itu sendiri dengan JavaScript.

Secara Umum Pemrograman Web dibagi menjadi 2 bagian yaitu :

1. Client Side Scripting (CSS)

Client Side Scripting digunakan ketika browser (pengguna) klien memiliki semua kode dan halaman tersebut diubah berdasarkan informasi klien (pengguna). Browser Web mengeksekusi skrip sisi klien yang terletak di dalam komputer pengguna. Skrip sisi klien juga dikenal sebagai skrip tertanam (karena mereka sering disematkan dalam dokumen HTML atau XHTML).

Browser mendapatkan halaman yang dikirim oleh server & mengeksekusi skrip sisi klien. Skrip sisi klien tidak dapat digunakan untuk bergabung dengan database di server web. Skrip sisi klien tidak bisa mendapatkan sistem file yang terletak di server web.

Catatan dan pengaturan yang bersifat lokal di komputer klien (pengguna) dapat didekati menggunakan bahasa skrip sisi Klien. Secara umum diamati bahwa respons dari skrip sisi klien lebih cepat bila dibandingkan dengan bahasa skrip sisi server saat skrip disiapkan di komputer lokal.

Contoh Script Paling Populer

1.Java Script

2.XML

3.CSS

2. Server Side Scripting (SSS)

Server Side Scripting adalah teknik yang digunakan dalam pengembangan web yang melibatkan penggunaan skrip pada server web yang menghasilkan respons yang disesuaikan untuk permintaan setiap pengguna (klien) ke situs web. Alternatifnya adalah untuk server web itu sendiri untuk memberikan halaman web statis. Skrip dapat ditulis dalam salah satu dari sejumlah bahasa skrip sisi server yang tersedia (lihat di bawah). Skrip sisi server dibedakan dari skrip sisi klien di mana skrip tertanam, seperti JavaScript, dijalankan sisi klien dalam browser web, tetapi kedua teknik ini sering digunakan bersama.

Server Side Scripting sering digunakan untuk menyediakan antarmuka yang disesuaikan untuk pengguna. Skrip ini dapat mengumpulkan karakteristik klien untuk digunakan dalam menyesuaikan respons berdasarkan karakteristik tersebut, persyaratan pengguna, hak akses, dll. Skrip sisi server juga memungkinkan pemilik situs web untuk menyembunyikan kode sumber yang menghasilkan antarmuka, sedangkan dengan sisi klien scripting, pengguna memiliki akses ke semua kode yang diterima oleh klien. Sisi buruk

dari penggunaan skrip sisi server adalah bahwa klien harus membuat permintaan lebih lanjut melalui jaringan ke server untuk menunjukkan informasi baru kepada pengguna melalui browser web. Permintaan ini dapat memperlambat pengalaman bagi pengguna, menempatkan lebih banyak beban di server, dan mencegah penggunaan aplikasi saat pengguna terputus dari server.

Ketika server menyajikan data dengan cara yang umum digunakan, misalnya sesuai dengan protokol HTTP atau FTP, pengguna dapat memilih sejumlah program klien (sebagian besar browser web modern dapat meminta dan menerima data menggunakan kedua protokol tersebut). Dalam hal aplikasi yang lebih khusus, pemrogram dapat menulis sendiri server, klien, dan protokol komunikasi mereka, yang hanya dapat digunakan satu sama lain.

Contoh Script Paling Populer:

- 1.PHP
- 2.ASP
- 3.JSP

1.1.2.2 Usur HTML

1 Tag

Tag adalah simbol khusus (markup) berupa 2 karakter “<” dan “>” yang mengapit suatu teks sebagai nama tag. Contohnya tag <body> adalah tag dengan nama body.

2 Atribut

Atribut yaitu property yang mengatur bagaimana elemen dari suatu tag akan ditampilkan. Atribut ada yang memiliki nilai dan ada yang tidak memiliki nilai. Nilai suatu atribut ditulis di dalam tanda petik ganda (“...”), dipisahkan dengan symbol sama dengan (=) dari nama. Contohnya, <p align=”center”>.

3 ELEMENT

Element adalah bagian dari skrip HTML yang terdiri dari tag pembuka, isi element dan tag tutup. Jika element ditampilkan di browser maka hanya isi element saja yang akan tampil. Contoh nya, <p> Politeknik Pos Indonesia </p> maka jika ditampilkan pada browser akan tampil hanya tulisa Politeknik Pos Indonesia.

1.1.2.3 Struktur dasar HTML

Dalam penulisan HTML ada beberapa tag yang wajib dituliskan dengan struktur yang telah ditentukan. Tag tersebut adalah sebagai berikut :

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Poltekpos</title>
  <head>
  <body>
    Politeknik Pos Indonesia
  </body>
</html>
```

- <!DOCTYPE HTML> adalah tag awal dari setiap dokumen HTML yang berfungsi untuk menginformasikan pada browser bahwa dokumen yang sedang dibuka adalah dokumen HTML. □
- <html>....</html> adalah tag yang menunjukkan pembuka dan penutup dokumen HTML.
- <head>....</head> adalah tag yang digunakan untuk menyimpan berbagai informasi tentang dokumen HTML.
- <title>....</title> adalah tag yang digunakan untuk membuat judul website yang nanti akan muncul di browser.
- <body>....</body>? Adalah tag yang menunjukkan bagian utama website. Semua yang akan ditampilkan pada halaman browser dituliskan di tag ini.

3.1.1.3 HUBUNGAN PHP DAN HTML

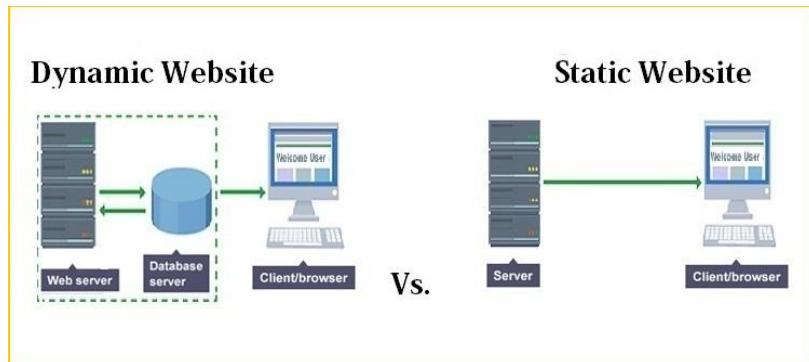
Halaman web biasanya disusun dari kode-kode html yang disimpan dalam sebuah file berekstensi .html. File html ini dikirimkan oleh server (atau file) ke browser, Kemudian browser menerjemahkan kode-kode tersebut sehingga menghasilkan suatu tampilan yang indah. Lain halnya dengan program php, program ini harus diterjemahkan oleh web-server sehingga menghasilkan kode html yang dikirim ke browser agar dapat ditampilkan. Program ini dapat berdiri sendiri ataupun disisipkan di antara kode-kode html sehingga dapat langsung ditampilkan bersama dengan kode-kode html tersebut.

Program php dapat ditambahkan dengan mengapit program tersebut diantara tanda. Tanda-tanda tersebut biasanya disebut tanda untuk escaping (kabur) dari kode html. File html yang telah dibubuhinya program php harus diganti ekstensi-nya menjadi .php3 atau php. Php merupakan bahasa pemrograman web yang bersifat server-side HTML=embedded scripting, di mana script-nya menyatu dengan HTML dan berada si server. Artinya adalah sintaks dan perintah-perintah yang kita berikan akan sepenuhnya dijalankan di server tetapi disertakan HTML biasa. PHP dikenal sebagai bahasa scripting yang menyatu dengan tag HTML, dieksekusi di server dan digunakan untuk membuat halaman web yang dinamis seperti ASP (Active Server Pages) dan JSP (Java Server Pages).

1.1.1.3 *Maksud Permrograman Web Dan Jenisnya*

Web adalah fasilitas dari Hypertext yang memiliki fungsi untuk menampilkan data berupa text, gambar, suara, animasi dan data multimedia dan jika ingin dapat menguasai web maka diperlukan mengenal Bahasa pemrograman Web yaitu HTML dan PHP. HTML termasuk kedalam kategori Script Client Side sedangkan PHP termasuk Kedalam Script Server Side yang akan dimasukan perintah perintah didalam suatu pemrograman web.

Web mengalami perkembangan yang sangat pesat mulai dari situs web E-commerce sampai dengan non profit situs. Dan dapat dikategorikan menjadi 2 yaitu:



1. Web Statis

Web statis ialah web yang berisi tentang informasi informasi yang memiliki sifat statis (tetap) atau pengguna tidak dapat berinteraksi dengan website tersebut, web statis dapat dilihat dari tampilan website tersebut jika suatu web hanya berhubungan dengan halaman web lain yang berisi informasi tetap maka web tersebut termasuk kedalam kategori web statis, pada web statis pengguna hanya dapat melihat isi web tersebut dan jika di klik hanya akan berpindah pada halaman lainnya. Dalam web statis interaksi pengguna sangatlah terbatas.

2. Web Dinamis

Web dinamis adalah web yang dapat menampilkan informasi serta dapat membuat pengguna berinteraksi seperti dengan form input, button sehingga dapat mengolah informasi yang di tampilkan pada web tersebut, web dinamis bersifat tidak kaku dan terlihat lebih enak dipandang.

1.1.3 BASIS DATA

1. Pengertian Basis Data

Basis Data terdiri dari kata basis dan data. Basis dapat diartikan sebagai markas atau gudang. Sedangkan data adalah catatan atas kumpulan fakta dunia

nyata yang mewakili objek seperti manusia, barang, hewan, konsep, peristiwa dan sebagainya yang diwujudkan dalam bentuk huruf, angka, simbol, gambar, teks, bunyi atau kombinasinya.

Sebagai suatu kesatuan maka pengertian basis data atau biasa disebut *database* adalah sebagai berikut:

Pengertian Basis Data atau *Database*:

1. Himpunan kelompok data yang saling terhubung dan diorganisasi sedemikian rupa supaya kelak dapat dimanfaatkan kembali secara cepat dan mudah.
2. Kumpulan data dalam bentuk file/tabel/arsip yang saling berhubungan dan tersimpan dalam media penyimpanan elektronis, untuk kemudahan dalam pengaturan, pemilahan, pengelompokan dan pengorganisasian data sesuai tujuan. Dengan basis data seseorang dapat menyimpan sebuah informasi, seperti data mahasiswa, kepegawaian atau produk ke dalam media penyimpanan elektronis seperti cakram magnetis (*disk*) melalui perangkat komputer, Untuk kemudian data tersebut dapat kita gunakan sesuai keperluan. Database mempunyai 8 operasi dasar diantaranya adalah *Create database, Drop database, create table, Drop table, Insert, Read, Update dan Delete*.

2. Pengertian Sistem Basis Data

Sistem basis data adalah sistem yang terdiri atas kumpulan tabel data yang saling berhubungan dan kumpulan program yang memungkinkan beberapa pemakai atau program lain untuk mengakses dan memanipulasi tabel tabel data tersebut.

3. Komponen Sistem Basis Data

- a. Perangkat Keras

Perangkat keras atau hardware yang umumnya terdapat dalam sistem basis data adalah komputer, hard disk, memori sekunder offline (removable disk, fd), perangkat komunikasi jaringan.

b. Sistem Operasi

Sistem operasi adalah program yang dirancang untuk mengaktifkan sistem komputer dan mengendalikan seluruh sumber daya yang ada di dalamnya termasuk operasi- operasi dasar komputer. seperti Windows, Unix dan Linux.

c. Basis Data

Komponen adalah sekumpulan data yang terorganisir dengan baik sehingga data tersebut mudah disimpan, diakses, dan juga dapat dimanipulasi. Sistem basis data dapat terdiri dari beberapa basis data yang memiliki data masing- masing.

d. Database Management System (DBMS)

DBMS atau database management system adalah program aplikasi khusus yang dirancang untuk membuat dan juga mengelola database yang tersedia. Sistem ini berisi koleksi data dan set program yang digunakan untuk mengakses database tersebut.

DBMS adalah software yang berperan dalam mengelola, menyimpan, dan mengambil data kembali. Adapun mekanisme yang digunakan sebagai pelengkap adalah pengaman data, konsistensi data dan pengguna data bersama.

Contoh dari DBMS adalah *Microsoft Access, MySql, Oracle databse, Sybase, Borland-Interbasi, PostgreSQL dll.*

e. Pemakai atau User

User adalah salah satu komponen database yang berinteraksi secara langsung dengan database. Ada beberapa tipe user, diantaranya,

programmer aplikasi, User mahir (casual user), user umum (end user) dan user khusus (specialized user)

f. Aplikasi atau Perangkat Lain

Aplikasi ini tergantung kebutuhan, pemakai basis data bisa dibuatkan program khusus untuk melakukan pengisian, pengubahan atau pengambilan data yang mudah dalam pemakaianya. Program tersebut ada yang tersedia langsung dalam DBMS atau dibuat menggunakan aplikasi lain seperti misalnya Visual Basic.

4. Bahasa Basis Data (Database Language)

Bahasa database merupakan bahasa data yang dapat ditempelkan kedalam bahasa pemrograman yang lain, sebut saja Java, Pascal, Fortran dst. Bahasa dimana instruksi data base menempel disebut inang. Beberapa komponen Bahasa data base menurut fungsinya dibagi menjadi:

a. *Data Definition Language*

Data definition language adalah sekumpulan definisi yang disimpan di dalam data dictionary.

b. *Data Manipulation Language*

Data Manipulation Language berisi akumulasi dari operasi manipulasi basis data yang dilakukan. Ini biasa disebut dengan bahasa query sebab biasanya digunakan untuk meminta informasi yang ada dari basis data tersebut.

5. Fungsi dan Tujuan Basis Data

Fungsi basis data cukup banyak dan cakupannya pun luas dalam mendukung keberadaan lembaga atau organisasi, diantaranya adalah:

a. Ketersediaan/ *Availability*

Fungsi basis data yang pertama adalah untuk menyediakan data-data penting saat sedang diperlukan. Ya, ini adalah fungsi penting dari basis data yang meskipun tidak terletak dalam satu lokasi, dan tersimpan dalam bentuk

disk, akan tetapi dengan cara penyimpanan yang sistematik, informasi tersebut mudah untuk didapatkan.

b. Mudah dan Cepat/ Speed

Selanjutnya, fungsi dari basis data ini adalah agar Anda sebagai pengguna bisa dengan mudah mengaksesnya saat sedang membutuhkan. Tidak perlu tunggu nanti, apalagi harus mengalokasikan waktu tertentu untuk memanggilnya.

c. Kelengkapan/ Completeness

Basis data harus menyimpan data yang lengkap, yang bisa melayani keperluan penggunanya secara keseluruhan. Meski kata lengkap yang dipakai disini sifatnya relatif, namun setidaknya data tersebut membantu memudahkan untuk menambah koleksi data, dan menjamin mudahnya pengguna untuk memodifikasi struktur data yang ada, sebut saja field-field data yang tersedia.

d. Accuracy dan Security

Fungsi data base selanjutnya adalah untuk accuracy atau keakuratan. Jadi, agar kesalahan dapat ditekan semaksimal mungkin, Anda bisa lakukan pengorganisasian file-file database dengan baik untuk menghindari kesalahan pada proses data entry dan juga dalam proses penyimpanan atau datastore. Selain itu, fungsi database adalah untuk security atau keamanan. Ada fasilitas pengaman data yang disediakan oleh sistem basis data yang baik sehingga data tidak bisa dimodifikasi, diakses, diubah maupun dihapus oleh yang tidak mendapatkan hak untuk melakukannya.

e. Storage Efficiency

Pengorganisasian data dilakukan dengan baik dengan tujuan untuk menghindari duplikasi data yang berpengaruh pada bertambahnya ruang penyimpanan dari basis data tersebut. pengkodean dan juga relasi data bermanfaat untuk menghemat space penyimpanan dalam basis data.

3.1.2 SEJARAH

Cabang ilmu Pemrograman cukup luas, dan erat kaitannya dengan disiplin ilmu yang lainnya. Hal ini bisa dilihat dari berbagai aplikasi yang merupakan hasil kombinasi dari berbagai ilmu.

3.1.2.1 Sejarah MySql



MySQL:Machine Learning bermula di awal abad 20, seorang penemu Spanyol, Torres y Quevedo, membuat sebuah mesin learning setelah ditemukannya komputer digital. Machine Learning pada dasarnya merupakan proses komputer untuk belajar dari data (Learn from data). Sejak pertama kali komputer diciptakan manusia sudah memikirkan bagaimana caranya agar komputer dapat belajar dari pengalaman. Gagasan ini terbukti yaitu pada tahun 1952, Arthur Samuel menciptakan sebuah program, game of checkers, pada sebuah komputer IBM.

Program tersebut dapat mempelajari gerakan untuk memenangkan permainan checkers dan menyimpan gerakan tersebut kedalam memorinya. Istilah teknologi machine learning pada dasarnya adalah proses komputer untuk belajar dari data (learn from data). Jika tidak ada data, maka komputer tidak akan bisa belajar apa-apa. Maka dari itu ketika kita ingin belajar machine learning, pasti akan terus berinteraksi dengan data. Semua pengetahuan machine learning pasti akan melibatkan data. Data bisa saja sama, akan tetapi

algoritma dan pendekatan nya berbeda-beda untuk mendapatkan hasil yang optimal.

3.1.2.2 Sejarah PHP



PHP adalah singkatan dari “PHP: Hypertext Preprocessor”, yaitu bahasa pemrograman disisi server yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah situs web dan bisa digunakan bersamaan dengan HTML. Ketika Anda mengakses sebuah URL, maka web browser akan melakukan request ke sebuah web server.

Pada awalnya PHP merupakan kependekan dari Personal Home Page (Situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama Form Interpreted (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI. Dengan perilisan kode sumber ini menjadi sumber terbuka, maka banyak pemrogram yang tertarik untuk ikut mengembangkan PHP.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini, interpreter PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0 dan singkatan PHP diubah menjadi akronim berulang PHP: Hypertext Preprocessing.

Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis PHP 5.0. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek ke dalam PHP untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek. Server web bawaan ditambahkan pada versi 5.4 untuk mempermudah pengembang menjalankan kode PHP tanpa menginstall software server.

Versi terbaru dan stabil dari bahasa pemograman PHP saat ini adalah versi 7.0.16 dan 7.1.2 yang resmi dirilis pada tanggal 17 Februari 2017.

3.1.2.3 Sejarah CSS



Pada tanggal 17 Agustus 1996, World Wide Web Consortium (W3C) menjadikan CSS sebagai bahasa pemrograman standart dalam pembuatan dokumen web. Tujuannya adalah mengurangi pembuatan tag-tag baru oleh Netscape dan Internet Explorer, karena kedua browser pada saat itu bersaing mengembangkan TAG sendiri untuk mengatur tampilan web.

CSS level 1 mendukung pengaturan tampilan dalam hal:

1. Font (jenis, ketebalan)
2. Warna, teks, backgrounf dan elemen lainnya
3. Text attributes, misalnya spasi antar baris, kata dan huruf
4. Posisi teks, gambar, table dan elemen lainnya
5. Marjin, border dan padding.

Selanjutnya di tahun 1998, W3c menyempurnakan CSS awal dengan menciptakan standar CSS2 (CSS level 2) yang menjadi standar hingga saat ini. Pada CSS level 2 ini, di masukkan semua atribut dari CSS1, serta diperluas dengan penekanan pada Internasional accessibility and capability khususnya

media-specific CSS. Bahkan pada tahun 2000, tidak lama setelah CSS2 diimplementasikan. CSS2 ini sampai sekarang masih terus dikembangkan, spesifikasinya dibagi pada beberapa topik atau modul.

3.1.3 BAHASA PEMROGRAMAN

3.1.3.1 Pemrograman Web



Seperti yang sudah disebutkan bahwa ada 2 kategori dalam pemrograman web, yaitu pemrograman Server Side dan Client side. Dalam pemrograman server side terdapat perintah program yang dijalankan pada server web lalu hasil akan dikirimkan dalam bentuk HTML biasa. Adapun pada Client Side program dijalankan pada browser web dari sever kemudian akan dijalankan oleh browser yang bersangkutan

A. Mengenal Script Client Side

Program web yang tergolong dalam Client Side Seperti Java Script, VB , HTML dll. Hasil dari parshing script pemrograman Client Side yang berupa Html dari web server dapat dilihat dengan menu view > source code dapat

terlihat bahwa script program yang ditulis ditampilkan pada halaman source code.

B. Java Script

JavaScript diperkenalkan pertama kali oleh Netscape di tahun 1995.pada mulanya Bahasa ini memiliki nama Livescript dan memiliki fungsi sebagai Bahasa sederhana untuk browser NetScope Navigator2. Pada masa pertama kali rilis Bahasa ini memiliki sangat banyak kritikan karena kurang aman, tidak ada pesan kesalahan dari setiap script program yang ditampilkan. Kemudian dengan kerja sama dengan Netscape dengan SUN (pengembang Bahasa pemrograman) maka Netscape merubah nama menjadi javascript pada tanggal 4 desember 1995.

Java Script adalah Bahasa pemrograman yang sederhana karena Bahasa yang digunakan tidak dapat membuat aplikasi, java script dapat dengan mudah kita temukan dalam suatu program yang sudah interaktif . program java script dituliskan pada file HTML. Dengan kata lain tidak perlu menuliskan program java script pada file terpisah. Bahasa ini adalah Bahasa pemrograman untuk memberikan kemampuan tambahan terhadap Bahasa HTML. Dengan mengizinkan menjalankan perintah dari sisi Klien. Yang artinya disisi browser bukan disisi server web javascript bergantung kepada browser memanggil pada halaman web yang berisi script java script tidak memerlukan penerjemah khusus untuk menjalankan script tersebut.

3.1.3.2 Keistimewaan MySql

Sebagai database server yang memiliki konsep database modern, MySQL memiliki banyak sekali keistimewaan. Berikut ini beberapa keistimewaan yang dimiliki oleh :

a. Portability

MySQL dapat berjalan stabil pada berbagai OS seperti Windows, Linux, Unix, Mac OS, Solaris, Unix, Amiga, HP-UX, Symbian.

b. Open Source "limited

Dahulu MySQL didistribusikan secara open source (gratis), dibawah lisensi GPL sehingga kita dapat menggunakannya secara cuma-cuma tanpa dipungut biaya. Namun, saat ini karena MySQL telah dibeli oleh SUN, maka kita tidak dapat lagi menikmati fitur-fitur baru yang ada di MySQL, karena SUN akan membatasi fitur-fitur baru ini hanya untuk user yang membeli lisensinya. Sehingga MySQL tidak lagi sebuah opensource yang benar-benar gratis lagi. MySQL sekarang hanya menyediakan fitur-fitur "dasar" saja yang saat ini sudah menggunakan versi 5.1. Untuk mendownloadnya silahkan download di sini dan dicari versi MySQL dengan OS kita.

c. Multiuser

MySQL dapat digunakan oleh beberapa user dalam waktu yang bersamaan tanpa mengalami konflik. Hal ini memungkinkan sebuah database server MySQL dapat diakses klien secara bersamaan.

d. Performance Tuning

MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.

e. Column Types

MySQL memiliki tipe kolom yang sangat kompleks, seperti signed/unsigned integer, float, double, char, varchar, text, blob, date, time, datetime, timestamp, year, set serta enum.

f. Command dan Functions

MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah SELECT dan WHERE dalam query.

g. Security

MySQL memiliki beberapa lapisan sekuritas seperti level subnetmask, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta password terenkripsi.

h. Scalability dan Limits

MySQL mampu menangani database dalam skala besar dengan jumlah records lebih dari 50 juta dan 60 ribu tabel serta 5 miliar baris. Selain itu, batas index yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

i. Connectivity

MySQL dapat melakukan koneksi dengan klien menggunakan TCP/IP, Unix soket (Unix), atau Named Pipes (NT).

j. Localisation

MySQL dapat mendeteksi pesan kesalahan (error code) pada klien dengan menggunakan lebih dari dua puluh bahasa

k. Interface

MySQL memiliki interface terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (Application Programming Interface).

l. Clients dan Tools

MySQL dilengkapi dengan berbagai tool yang dapat digunakan untuk administrasi database, dan pada setiap tool yang ada disertakan petunjuk online.

m. Struktur Tabel

MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE dibandingkan database lainnya.

3.1.3.3 PHP

PHP Merupakan singkatan dari "PHP: Hypertext Preprocessor", yang merupakan sebuah bahasa scripting (kode untuk membangun suatu program)

yang dikombinasikan pada HyperText Markup Language (HTML). Tujuan utama penggunaan bahasa ini adalah untuk memungkinkan perancang web menulis halaman web dinamik dengan cepat

1. Memiliki Community yang besar

Programmer Web mana yang tidak mengetahui PHP, semua web programmer paling tidak pasti pernah mencoba PHP. Banyak sekali website yang menggunakan PHP sebagai bahasa pemrograman untuk membuat aplikasi web atau website nya. Facebook, Yahoo, Wikipedia, WordPress adalah contoh website terkenal yang menggunakan PHP. Forum untuk membahas dan juga saling bertukar pikiran dalam pemrograman PHP juga telah banyak muncul di berbagai situs. Kebanyakan kuliah di bidang IT mengajarkan PHP sebagai bahasa pemrograman awal untuk mahasiswanya yang berkuliah di jurusan website development (baca juga: Daftar Mata Kuliah Teknik Informatika)

2. Mudah Dipelajari

PHP mudah di install dan dikonfigurasi. membuatnya menjadi bahasa pemrograman tingkat entry level yang mudah dipelajari bagi seseorang yang baru memulai belajar pengembangan web. Tutorial untuk memulai belajar pemrograman PHP dapat diperoleh dengan mudah secara online, di toko buku, ataupun di lembaga bimbingan kursus pengembangan website.

3. Pengembangan Cepat

Membuat Aplikasi menggunakan PHP jauh lebih cepat daripada mengembangkan aplikasi web menggunakan bahasa pemrograman lain. banyak sekali tools, boiler yang tersedia secara open source untuk bahasa pemrograman PHP. hal ini mempercepat proses dari start sampai dengan finish sebuah projek pembuatan aplikasi web.

4. Ringkas

Bagi Programmer web yang pernah mencoba bahasa ASP maupun java pasti mengetahui betul satu kelebihan ini. Mulai dari proses install yang tidak perlu setting berlebihan, konfigurasi dengan database yang mudah. hingga proses pengembangan yang tidak memerlukan waktu kompilasi. membuat PHP terasa sangat ringkas dan praktis berbeda dengan bahasa pemrograman lain yang membutuhkan proses kompilasi untuk dapat melihat website yang telah diselesaikan pembuatan kodennya. Bahkan, bahasa pemrograman php dapat digunakan didalam dokumen html, hmm ringkas bukan. (baca juga: Dasar Dasar HTML)

5. Maintenance mudah

Sekali web yang menggunakan PHP berjalan, programmer dapat dengan mudah melakukan update dari software PHP dengan mudah jika memang diperlukan. karena sifat PHP yang merupakan interpreter. Aplikasi web yang dibuat dengan menggunakan PHP dapat dengan mudah diupgrade versi PHP tanpa harus melakukan kompilasi ulang source code. berbeda sekali dengan bahasa pemrograman lain yang membutuhkan kompilasi ulang jika melakukan upgrade versi dari bahasa pemrograman. PHP juga dapat berjalan pada berbagai macam web server seperti apache, nginx, dan IIS. (baca juga : Pengertian Web Server Menurut Para Ahli)

6. Open Source

PHP merupakan sebuah projek Open source dengan license yang dikeluarkan oleh PHP group yaitu PHP license V3.01. Inti dari license ini adalah setiap pengguna program PHP bebas menggunakan PHP secara gratis tanpa harus memberikan royalty apapun ke PHP group namun tetap wajib mencantumkan lisensi atas PHP yang dimiliki PHP Group. Dengan kata lain

selama pemakai program PHP tidak mengakui produk PHP adalah buatannya maka perjual belian program yang menggunakan PHP diperbolehkan tanpa harus membayar lisensi apapun.

7. Perkembangan Pesa

Karena sifat PHP yang open source, banyak sekali bermunculan projek projek open source besar yang menggunakan PHP seperti Prestashop, WordPress, Drupal, dan lain lain. Hal ini menjadi keunggulan yang sangat besar bagi orang yang menguasai pemrograman PHP. Dengan sangat luasnya perkembangan PHP, maka kesempatan untuk bisnis ataupun kerja pada bidang pemrograman PHP sangatlah luas

3.1.3.4 Definisi CSS

CSS adalah singkatan dari Cascading Style-Sheet, sebuah pengembangan atas kode HTML yang sudah ada sebelumnya. Dengan CSS, kita bisa menentukan sebuah struktur dasar halaman web secara lebih mudah dan cepat, serta irit size. CSS juga membantu kita untuk menyeragamkan seluruh halaman website dengan tampilan yang konsisten. Misalnya, kita mau seluruh font yang ada di website kita adalah font "Tahoma", maka dengan bantuan CSS kita bisa bikin proses itu menjadi otomatis tanpa harus mengganti-ganti font secara manual di setiap halaman.

Dahulu kala, sebelum CSS menjadi standar untuk mendesain halaman web seperti sekarang, halaman web di desain menggunakan <table>. jadi dibuat dulu desainnya, dalam format .psd atau jpeg, lalu di slice atau di potong potong menjadi bagian – bagian terpisah. setelah itu dibuat table dengan ukuran yang sesuai, lalu desain tadi di “tempel” pada table sebagai image yang melekat pada tabel, atau sebagai background. Kelemahan dari cara seperti ini adalah, halaman web menjadi berat karena kolom – kolom penyusun table <tr>

dan <td> diberi tambahan atribut image source. Side effect dari hal ini adalah halaman web menjadi padat dan tidak SEO friendly.

Pada perkembangannya CSS sudah masuk level 3 untuk sekarang, dimana dimulai CSS level 1 atau yang sering disebut CSS saja, kemudian level 2 yang merupakan penyempurnaan dari CSS level sebelumnya, yaitu CSS level 1. CSS merupakan alternatif bahasa pemrograman web masa yang akan datang, dimana mempunyai banyak keuntungan, diantaranya :

- a. Ukuran file lebih kecil
- b. Load file lebih cepat
- c. Dapat berkolaborasi dengan Javascript
- d. Pasangan setia XHTML
- e. Menghemat pekerjaan tentunya, dimana hanya membuat 1 halaman CSS.
- f. Mudah mengganti-ganti tampilan dengan hanya merubah file CSS nya saja.

3.1.3.5 Definisi Bahasa Pemrograman C

Bahasa Pemrograman C adalah sebuah bahasa pemrograman komputer yang bisa digunakan untuk membuat berbagai aplikasi (general-purpose programming language), mulai dari sistem operasi (seperti Windows atau Linux), antivirus, software pengolah gambar (image processing), hingga compiler untuk bahasa pemrograman, dimana C banyak digunakan untuk membuat bahasa pemrograman lain yang salah satunya adalah PHP.

Meskipun termasuk general-purpose programming language, yakni bahasa pemrograman yang bisa membuat berbagai aplikasi, bahasa pemrograman C paling cocok merancang aplikasi yang berhubungan langsung dengan Sistem Operasi dan hardware. Ini tidak terlepas dari tujuan awal bahasa C dikembangkan.

Bahasa pemrograman C dibuat pertama kali oleh Dennis M. Ritchie pada tahun 1972. Saat itu Ritchie bekerja di Bell Labs, sebuah pusat penelitian yang berlokasi di Murray Hill, New Jersey, Amerika Serikat. Ritchie membuat bahasa pemrograman C untuk mengembangkan sistem operasi UNIX. Sebelumnya, sistem operasi UNIX dibuat menggunakan bahasa assembly (assembly language). Akan tetapi bahasa assembly sendiri sangat rumit dan susah untuk dikembangkan.

Dengan tujuan mengganti bahasa assembly, peneliti di Bell Labs membuat bahasa pemrograman B. Namun bahasa pemrograman B juga memiliki beberapa kekurangan, yang akhirnya di lengkapi oleh bahasa pemrograman C. Dengan bahasa C inilah sistem operasi UNIX ditulis ulang. Pada gilirannya, UNIX menjadi dasar dari banyak sistem operasi modern saat ini, termasuk Linux, Mac OS (iOS), hingga sistem operasi Android.

3.1.3.5.1 Fitur dan Keunggulan Bahasa Pemrograman C

Berikut beberapa fitur serta keunggulan bahasa pemrograman C jika dibandingkan dengan bahasa pemrograman lain:

1. C Sebagai Bahasa pemrograman procedural.

Konsep pemrograman prosedural adalah sebuah metode pemrograman yang setiap baris perintah diproses secara berurutan dari baris paling atas hingga baris paling bawah. Selain itu bisa terdapat fungsi tambahan (*function*) yang digunakan untuk menyelesaikan berbagai tugas. Bahasa pemrograman C termasuk ke dalam kelompok ini.

Selain konsep prosedural, terdapat juga konsep pemrograman object (*object-oriented programming*). Di dalam bahasa pemrograman object, setiap tugas akan dijalankan menggunakan *class* dan *object*. Contoh bahasa pemrograman object adalah JAVA.

Bagi pemula, sangat disarankan untuk mempelajari bahasa pemrograman prosedural terlebih dahulu baru kemudian masuk ke dalam bahasa pemrograman object. Ini juga menjadi alasan untuk belajar bahasa C sebelum masuk ke bahasa pemrograman object seperti JAVA. Beberapa bahasa pemrograman ada yang mendukung konsep prosedural dan object sekaligus, contohnya bahasa pemrograman C++, Python dan PHP.

2. Bahasa C sangat efisien.

Aplikasi yang dibuat menggunakan bahasa C bisa dieksekusi dengan sangat cepat serta berukuran kecil. Ini karena C bisa langsung berkomunikasi dengan hardware, sebuah fitur yang jarang tersedia di bahasa pemrograman modern seperti JAVA, PHP, maupun Python. Akan tetapi, hal ini juga memiliki kelemahan. Bahasa C relatif sederhana dan tidak memiliki fitur-fitur modern seperti garbage collection dan dynamic typing.

3. C adalah portable language.

Maksudnya, bahasa pemrograman C bisa di-compile ulang supaya berjalan di berbagai sistem operasi tanpa perlu mengubah kode-kode yang ada. Aplikasi yang dibuat di Windows dengan bahasa C, bisa dipindahkan ke Linux dengan sedikit atau tanpa modifikasi.

4. C merupakan induk dari Bahasa pemrograman modern.

Bahasa pemrograman C banyak menginspirasi bahasa pemrograman lain, seperti C++, C#, Objective C, PHP, JAVA, JavaScript dan masih banyak lagi. Dengan mempelajari bahasa C, anda akan familiar dan lebih mudah saat berpindah ke bahasa pemrograman lain yang merupakan turunan dari bahasa C.

3.1.3.6 Definisi Bahasa Pemrograman Java

Java adalah bahasa pemrograman tingkat tinggi yang berorientasi objek dan program java tersusun dari bagian yang disebut kelas. Kelas terdiri atas

metode-metode yang melakukan pekerjaan dan mengembalikan informasi setelah melakukan tugasnya. Para pemrogram Java banyak mengambil keuntungan dari kumpulan kelas di pustaka kelas Java, yang disebut dengan *Java Application Programming Interface* (API).

Kelas-kelas ini diorganisasikan menjadi sekelompok yang disebut paket (package). Java API telah menyediakan fungsionalitas yang memadai untuk menciptakan applet dan aplikasi canggih. Jadi ada dua hal yang harus dipelajari dalam Java, yaitu mempelajari bahasa Java dan bagaimana mempergunakan kelas pada Java API. Kelas merupakan satu-satunya cara menyatakan bagian eksekusi program, tidak ada cara lain. Pada Java program javac untuk mengkompilasi file kode sumber Java menjadi kelas-kelas bytecode. File kode sumber mempunyai ekstensi *.java. Kompilator javac menghasilkan file bytecode kelas dengan ekstensi *.class. Interpreter merupakan modul utama sistem Java yang digunakan aplikasi Java dan menjalankan program bytecode Java.

Beberapa keunggulan java yaitu java merupakan bahasa yang sederhana. Java dirancang agar mudah dipelajari dan digunakan secara efektif. Java tidak menyediakan fitur-fitur rumit bahasa pemrograman tingkat tinggi, serta banyak pekerjaan pemrograman yang mulanya harus dilakukan manual, sekarang digantikan dikerjakan Java secara otomatis seperti dealokasi memori. Bagi pemrogram yang sudah mengenal bahasa C++ akan cepat belajar susunan bahasa Java namun harus waspada karena mungkin Java mengambil arah (semantiks) yang berbeda dibanding C++.

Java merupakan bahasa berorientasi objek (OOP) yaitu cara ampuh dalam pengorganisasian dan pengembangan perangkat lunak. Pada OOP, program komputer sebagai kelompok objek yang saling berinteraksi. Deskripsi ringkas OOP adalah mengorganisasikan program sebagai kumpulan komponen,

disebut objek. Objek-objek ini ada secara independen, mempunyai aturan-aturan berkomunikasi dengan objek lain dan untuk memerintahkan objek lain guna meminta informasi tertentu atau meminta objek lain mengerjakan sesuatu. Kelas bertindak sebagai modul sekaligus tipe. Sebagai tipe maka pada saat jalan, program menciptakan objek-objek yang merupakan instan-instan kelas. Kelas dapat mewarisi kelas lain. Java tidak mengijinkan pewarisan jamak namun menyelesaikan kebutuhan pewarisan jamak dengan fasilitas antarmuka yang lebih elegan.

Seluruh objek diprogram harus dideklarasikan lebih dulu sebelum digunakan. Ini merupakan keunggulan Java yaitu Statically Typed. Pemakaian ini memungkinkan kompilator Java menentukan dan melaporkan terjadinya pertentangan (ketidakkompatibelan) tipe yang merupakan barikade awal untuk mencegah kesalahan yang tidak perlu (seperti mengurangkan variabel bertipe integer dengan variabel bertipe string). Pencegahan sedini mungkin diharapkan menghasilkan program yang bersih. Kebaikan lain fitur ini adalah kode program lebih dapat dioptimasi untuk menghasilkan program berkinerja tinggi.

Java menggunakan model pengamanan tiga lapis (three-layer security model) untuk melindungi sistem dari untrusted Java code. Pertama, bytecode verifier membaca bytecode sebelum dijalankan dan menjamin bytecode memenuhi aturan-aturan dasar bahasa Java. Kedua, class loader menangani pemuatan kelas Java ke runtime interpreter. Ketiga, manajer keamanan menangani keamanan tingkat aplikasi dengan mengendalikan apakah program berhak mengakses sumber daya seperti sistem file, port jaringan, proses eksternal dan sistem window.

Platform independence adalah kemampuan program bekerja di sistem operasi yang berbeda. Bahasa Java merupakan bahasa yang secara sempurna

tidak bergantung platform. Tipe variabel Java mempunyai ukuran sama di semua platform sehingga variabel bertipe integer berukuran sama tidak peduli dimana program java dikompilasi. Begitu telah tercipta file .class dengan menggunakan kompilator Java di platform manapun, maka file .class tersebut dapat dijalankan di platform manapun. Jadi “dimanapun dibuat, dimanapun dapat dijalankan”. Slogan ini biasa diringkas sebagai Write Once, Run Anywhere (WORA).

Java termasuk bahasa Multithreading. Thread adalah untuk menyatakan program komputer melakukan lebih dari satu tugas di satu waktu yang sama. Java menyediakan kelas untuk menulis program multithreaded, program mempunyai lebih dari satu thread eksekusi pada saat yang sama sehingga memungkinkan program menangani beberapa tugas secara konkuren. Program Java melakukan garbage collection yang berarti program tidak perlu menghapus sendiri objek-objek yang tidak digunakan lagi. Fasilitas ini mengurangi beban pengelolaan memori oleh pemrogram dan mengurangi atau mengeliminasi sumber kesalahan terbesar yang terdapat di bahasa yang memungkinkan alokasi dinamis.

Java mempunyai mekanisme exception-handling yang ampuh. Exception-handling menyediakan cara untuk memisahkan antara bagian penanganan kesalahan dengan bagian kode normal sehingga menuntun ke struktur kode program yang lebih bersih dan menjadikan aplikasi lebih tegar. Ketika kesalahan yang serius ditemukan, program Java menciptakan exception. Exception dapat ditangkap dan dikelola program tanpa resiko membuat sistem menjadi turun. Program Java mendukung native method yaitu fungsi ditulis di bahasa lain, biasanya C/C++. Dukungan native method memungkinkan pemrogram menulis fungsi yang dapat dieksekusi lebih cepat dibanding fungsi

ekivalen di java. Native method secara dinamis akan di-link ke program java, yaitu diasosiasikan dengan program saat berjalan.

Selain itu keuntungan menggunakan bahasa pemrograman Java antara lain. Memori pada Java secara otomatis dilengkapi garbage collector yang berfungsi mendealokasi memori yang tidak diperlukan. Tidak ada lagi upaya yang dilakukan pemrogram untuk melakukan dispose(). Kita tidak lagi dibebani urusan korupsi memori. Java menerapkan array sebenarnya, menghilangkan keperluan aritmatika pointer yang berbahaya dan mudah menjadi salah. Menghilangkan pewarisan jamak (multiple inheritance) diganti fasilitas antarmuka. Dan mudah dijalankan diberbagai platform.

Grafical User Interface (GUI) adalah salah satu kemampuan Java dalam mendukung dan manajemen antarmuka berbasis grafis. Tampilan grafis yang akan ditampilkan terhubung dengan program serta tempat penyimpanan data. Elemen dasar di Java untuk penciptaan tampilan berbasis grafis adalah dua paket yaitu AWT dan Swing. Abstract Windowing Toolkit (AWT), atau disebut juga “Another Windowing Toolkit”, adalah pustaka windowing bertujuan umum dan multiplatform serta menyediakan sejumlah kelas untuk membuat GUI di Java. Dengan AWT, dapat membuat window, menggambar, manipulasi gambar, dan komponen seperti Button, Scrollbar, Checkbox, TextField, dan menu pull-down.

Penggunaan komponen AWT ditandai dengan adanya instruksi : import java.awt.*; Swing merupakan perbaikan kelemahan di AWT. Banyak kelas swing menyediakan komponen alternatif terhadap AWT. Contohnya kelas JButton swing menyediakan fungsionalitas lebih banyak dibanding kelas Button. Selain itu komponen swing umumnya diawali dengan huruf “J”, misalnya JButton, JTextField, JFrame, JLabel, JTextArea, JPanel, dan sebagainya. Teknologi swing menggunakan dan memperluas gagasan-

gagasan AWT. Sementara, penggunaan komponen Swing ditandai dengan adanya instruksi : import javax.swing.*;

Beberapa perbedaan AWT dan Swing, AWT merupakan komponen heavyweight (kelas berat) sedangkan Swing lightweight (kelas ringan). Swing memiliki lebih banyak komponen. Fasilitas Swing Look and Feel : Metal, Windows, Motif. Komponen Swing berdasar model-view, yaitu suatu cara pengembangan komponen dengan pemisahan penyimpanan dan penanganan data dari representasi visual data. Bahasa pemrograman Java merupakan salah satu bahasa pemrograman yang umum digunakan untuk mengembangkan aplikasi basis data yang dibuat menggunakan MySQL.

3.1.3.7 Definisi Bahasa Pemrograman Python

Python adalah bahasa pemrograman yang populer. Bahasa pemrograman ini dibuat oleh Guido van Rossum dan dikenalkan sejak tahun 1991. Sebelum memulai untuk belajar Python dasar, akan lebih baik untuk memahami dulu apa itu Python dan bagaimana cara kerjanya. Python termasuk bahasa pemrograman yang mudah untuk dipelajari. Sampai saat ini bahasa pemrograman Python hampir dipakai di segala bidang seperti game, sistem berbasis web, dan bahkan dapat membuat mesin pencari sendiri. Jadi secara umum, bahasa pemrograman ini dipakai dalam pengembangan website, pengembangan software, matematika, dan system scripting.

3.1.3.7.1 Hal-hal yang dapat dilakukan dengan python.

Sebelum belajar Phyton lebih jauh, Anda harus mengetahui apa saja yang bisa dilakukan dengan bahasa pemrograman ini.

Berikut ini beberapa hal yang dapat Anda lakukan menggunakan Python:

1. Python dapat menjadi salah satu bahasa pemrograman untuk membangun server ketika Anda membuat website.

2. Ketika Anda membutuhkan proses pembuatan prototipe atau pengembangan perangkat lunak siap produksi, Python dapat Anda andalkan.
3. Python dapat digunakan untuk membuat workflow di dalam pengembangan perangkat lunak.
4. Python dimanfaatkan untuk membaca dan memodifikasi sebuah file di dalam pembangunan sistem database.
5. Python memungkinkan Anda untuk menangani big data dan menjalankan pemrosesan matematika yang kompleks.

3.1.3.7.2 Manfaat belajar python

Meskipun ada banyak sekali bahasa pemrograman di luar sana, akan tetapi saya merekomendasikan Anda untuk belajar Python. Tentu juga ada beberapa kelebihan yang perlu Anda ketahui. Jika dibandingkan dengan bahasa pemrograman lain, berikut kelebihan bahasa pemrograman Python:

1. Python memiliki sintaksis yang sederhana dan lebih mirip dengan Bahasa Inggris.
2. Python dapat berjalan di berbagai macam sistem operasi.
3. Python berjalan di dalam sistem interpreter, artinya bahasa baris kode bahasa pemrograman ini akan segera dieksekusi setelah ditulis.
4. Python dapat diperlakukan dengan cara prosedural, cara berorientasi objek atau cara fungsional.
5. Python memiliki sintaks yang memungkinkan pengembang untuk menulis program dengan ringkas daripada bahasa pemrograman lain.

3.1.3.7.3 Komponen Python

Sesudah Anda memastikan Python sudah terinstall dengan baik di perangkat. Langkah selanjutnya adalah melakukan percobaan beberapa eksekusi program Python. Namun sebelum itu akan lebih baik jika mengetahui terlebih dahulu apa saja komponen yang terdapat di dalam Python.

1. Sintaks

Python sintaks dapat dieksekusi langsung dengan mengetikkannya di Command Line. Selain itu, Anda dapat membuat file Python di dalam server menggunakan ekstensi .py dan menjalankannya menggunakan Command Line.

```
>>> print("Hello, World!")Hello, World!
```

2. Komentar

Sama seperti bahasa pemrograman lainnya, Python juga memiliki kode untuk menjadikan baris program menjadi komentar. Anda dapat menggunakan tanda pagar '#' untuk menjadikan baris kode di Python menjadi komentar.

```
# Ini adalah baris komentar di Python.print("Hello, World!")
```

3. Python Identitations

Berbeda dengan bahasa pemrograman lainnya, jika Anda menulis dalam bahasa Python, indentasi –penempatan kalimat atau baris kode– sangat diperhatikan. Python menggunakan indentasi untuk mengindikasikan baris kode.

```
if 9 > 2:    print("Sembilan lebih besar daripada dua!")
```

Namun ketika baris kode dituliskan menjadi satu kolom atau dalam tab yang sama, maka program akan menjadi error. Di bawah ini adalah contoh penulisan yang menghasilkan error.

```
if 9 > 2:print("Sembilan lebih besar daripada dua!")
```

4. Variable

Python juga memiliki Variabel, tidak berbeda dengan bahasa pemrograman lainnya. Variabel ini digunakan untuk proses penyimpanan dan bekerja dengan berbagai tipe data.

Python sendiri punya standar pendeklarasian variabel. Variabel di Python dapat berupa nama singkat (seperti x dan y tadi) atau nama yang lebih mendeskripsikan seperti umur, nama, alamat, dan lain sebagainya. Aturan penamaan variabel di Python seperti:

- Variabel tidak bisa diawali dengan angka,
- Variabel harus diawali dengan huruf, atau karakter garis bawah (underscore),
- Variabel hanya bisa mengandung karakter alfa-numerik dan karakter garis bawah,
- Variabel di *Python case-sensitive*

Namun berbeda dengan bahasa pemrograman lainnya, Python tidak memerlukan inisiasi variabel untuk mendeklarasikan variabel. Ini berarti sebuah variabel terbuat ketika pertama kali Anda menambahkan nilai ke dalamnya.

Contohnya ketika Anda ingin membuat variabel ‘x’ dan ‘y’, Anda tinggal memasukkan nilainya langsung seperti di bawah ini:

```
x = 10  
y = "Budi"  
print(x)  
print(y)
```

Perintah di atas akan mengisi variabel ‘x’ dengan nilai ‘5’ dan ‘y’ dengan nilai ‘Budi’. Jadi proses penyusunan baris kode lebih ringkas.

Kemudahan lainnya, Anda tidak perlu mendefinisikan tipe variabel. Python secara otomatis akan memberikan tipe variabel sesuai dengan nilai yang diberikan pada variabel tersebut. Misalnya pada contoh di bawah ini:

```
x = 10 # x bertipe integery = "Budi" # y bertipe string
```

5. Booleans

Setelah mempelajari variabel bekerja, di bagian ini Anda akan belajar tentang Booleans. Jika Variabel dapat menyimpan bilangan dengan satu tipe data, booleans juga digunakan untuk menyimpan sebuah tipe data, tapi tipe data yang berbeda.

Tipe data di Booleans hanya ‘benar’ atau ‘salah’. Jadi ini mirip dengan saklar lampu, hanya memiliki dua nilai. Anda dapat menggunakan booleans seperti contoh di bawah ini:

```
a = Trueb = False
```

6. Number

Ketika Anda belajar Python number, ada tiga tipe numerik variabel di Python, yaitu int, float, dan complex. Anda mungkin tidak akan pernah menuliskan tipe variabel di setiap pendeklarasiannya, karena (seperti yang sudah dijelaskan di atas) Python sudah menginisiasi tipe variabel ketika Anda menambahkan nilai ke dalamnya.

Int, float, dan complex mempunyai range yang berbeda. Int atau bilangan integer adalah bilangan bulat positif atau negatif, tanpa desimal, dengan panjang tak terbatas. Float atau ‘angka floating point’ adalah angka, positif atau negatif, yang mengandung satu atau lebih desimal. Sedangkan complex adalah bilangan kompleks yang ditulis dengan “j” sebagai bagian dari imajiner.

Sebagai contoh, di bawah ini adalah tiga tipe numerik variabel yang berbeda:

```
x = 1 # inty = 2.8 # floatz = 1j # complex
```

Jadi Python akan mengenali dan membedakan tipe setiap variabel pada saat Anda mengisinya dengan sebuah nilai. Sedangkan untuk mengetahui tipe setiap variabel Anda dapat menggunakan fungsi type():

```
print(type(x))print(type(y))print(type(z))
```

7. String

Ketika ingin belajar Python string, Anda hanya perlu menambahkan tanda kutip tunggal atau tanda kutip ganda di antara nilai variabel yang ingin ditambahkan. Misalnya saja ketika Anda ingin menambahkan string “Budi” ke dalam variabel x maka yang perlu Anda lakukan adalah mendeklarasikannya seperti di bawah ini:

```
x = "Budi"# Atau x = 'Budi'  
# "Budi" sama artinya dengan 'Budi'.
```

Sama seperti bahasa pemrograman lainnya, string dalam Python adalah array byte yang mewakili karakter unicode. Namun, Python tidak memiliki tipe data karakter sehingga satu karakter hanyalah string dengan panjang 1. Ketika Anda ingin mengakses satu karakter di dalam string, yang perlu Anda gunakan adalah menggunakan tanda kurung kotak.

Di bawah ini adalah contoh mengambil karakter kedua pada sebuah string.

```
a = "Hello, World!"print(a[1])
```

8. Operator

Bagian terakhir dari pengenalan komponen Python adalah Operator. Selama melakukan proses coding Anda pasti akan membutuhkan operator untuk membuat sebuah alur logika, penghitungan angka, atau yang lainnya. Operator ini bekerja untuk melakukan operasi pada variabel dan nilai. Dalam bahasa pemrograman Python, terdapat beberapa grup dari operator,

seperti operator aritmatika, penugasan (assignment), pembanding (comparison), logika (logical), identitas (identity), keanggotaan (membership), dan bitwise.

Di antara operator lainnya, operator aritmatika sering digunakan. Operator aritmatika ini mengandung beberapa operator. Di bawah ini adalah daftar operator aritmatika secara lengkapnya:

Operator	Nama	Contoh
+	Penambahan	$x + y$
-	Pengurangan	$x - y$
*	Perkalian	$x * y$
/	Pembagian	x / y
%	Modulus	$x \% y$
**	Exponensian (Pangkat)	$x ** y$
//	Floor division	$x // y$

Selain itu, beberapa proses pengembangan juga sering membutuhkan operator yang dapat menetapkan suatu nilai ke dalam variabel. Berikut ini adalah daftar operator assignment yang dapat Anda gunakan di dalam Python:

Operator	Contoh	Persamaan
=	x = 5	x = 5
+=	x += 3	x = x + 3
-=	x -= 3	x = x - 3
*=	x *= 3	x = x * 3
/=	x /= 3	x = x / 3
%=	x %= 3	x = x % 3
//=	x //= 3	x = x // 3
**=	x **= 3	x = x ** 3
&=	x &= 3	x = x & 3
=	x = 3	x = x 3
^=	x ^= 3	x = x ^ 3
>>=	x >>= 3	x = x >> 3
<<=	x <<= 3	x = x << 3

Sedangkan ketika Anda ingin membandingkan antara satu atau beberapa variabel, biasanya membutuhkan operator perbandingan di antaranya. Operator ini terdiri dari enam jenis. Berikut ini adalah operator perbandingan yang digunakan untuk membandingkan dua nilai:

Operator	Nama	Contoh
==	Sama dengan	x == y
!=	Tidak sama dengan	x != y
>	Lebih besar dari	x > y
<	Kurang dari	x < y
>=	Lebih besar atau sama dengan	x >= y
<=	Lebih kecil atau sama dengan	x <= y

9. Python if else

Ketika Anda membutuhkan perbandingan antara kondisi satu dengan yang lain, Python dapat Anda gunakan untuk mendukung kondisi logis dari

matematika. Aturan logika ini biasanya digunakan untuk memberikan syarat sebelum sebuah baris program diambil.

Sesuai dengan tabel perbandingan yang sudah dibahas pada bagian sebelumnya, ada enam kondisi logis yang dapat digunakan di Python; sama dengan ($a == b$), tidak sama dengan ($a != b$), kurang dari ($a < b$), kurang dari atau sama dengan ($a \leq b$), lebih besar dari ($a > b$), lebih besar atau sama dengan ($a \geq b$).

Kondisi ini dapat digunakan dengan beberapa modifikasi, lebih sering digunakan untuk “pernyataan If” dan perulangan. Ketika digunakan untuk “pernyataan If”, contohnya seperti di bawah ini:

```
a = 8b = 10if a > a: print("b lebih besar dari a")
```

Selain penggunaan “If”, Anda dapat menggunakan “Elif dan Else”. Elif berarti “jika kondisi sebelumnya tidak benar, maka coba kondisi ini. Sedangkan Else menyatakan “lakukan perintah berikut jika semua kondisi tidak sesuai”.

Di bawah ini adalah contoh penggunaan If, Elif, dan Else dalam satu logika.

```
a = 8b = 10if a > a: print("b lebih besar dari a")elif a == b: print("a sama dengan b")Else: print("lebih besar dari b")
```

3.1.3.8 Definisi Bahasa Pemrograman Ruby

Bahasa Ruby merupakan sebuah bahasa pemrograman yang dinamis, reflektif, general-purposed, dan berbasis objek. Bahasa ini dirancang dan dikembangkan oleh Yukihiro “Matz” Matsumoto di Jepang. Sebelum menentukan namanya, Yukihiro memiliki dua nama untuk digunakan bahasa ini yaitu Rubi dan Coral. Akan tetapi, Yukihiro lebih memilih menggunakan Ruby karena sama dengan batu kelahiran dari salah satu koleganya.

Meskipun mudah dipelajari, bahasa ini memiliki fungsi yang sangat luar biasa diantaranya yaitu dapat digunakan untuk membangun Desktop GUI (Graphic User Interfaces), membuat aplikasi web, atau mengembangkan web itu sendiri. Telah banyak platform yang dibangun menggunakan bahasa ini mulai dari Hulu, Groupon, Airbnb, dan bahkan Twitter. Hal ini tentunya disebabkan karena kelebihan bahasa pemrograman Ruby tersebut.

Seperti yang dijelaskan sebelumnya, salah satu kelebihan bahasa pemrograman Ruby adalah mudah untuk dipelajari. Bahkan, dalam infografis bahasa pemrograman ini, bahasa pemrograman Ruby dinyatakan sebagai bahasa pemrograman awal terbaik ketika kamu pertama kali belajar pemrograman.

Oleh Yukihiro, bahasa pemrograman ini dikembangkan agar dapat digunakan secara natural dan mudah untuk dimengerti oleh penggunanya. Selain itu, telah terdapat berbagai macam situs yang telah membahas Ruby, yang dapat digunakan sebagai referensi saat mempelajari bahasa ini. Selain mudah untuk dipelajari, kelebihan bahasa pemrograman Ruby lainnya juga memiliki kesamaan dengan kelebihan yang dimiliki bahasa Phyton. Bahasa pemrograman Ruby juga menawarkan berbagai macam library perlengkapan yang menakjubkan serta fungsionalitas yang sangat luas.

Dengan kelebihan ini, para developer dapat menggunakan bahasa pemrograman Ruby dengan sangat mudah dalam membangun berbagai macam hal. Para developer tidak harus membuat segala sesuatunya dari awal, mereka dapat menggunakan beberapa library yang sudah ada sebelumnya dalam membangun sebuah kode.

3.1.3.9 Definisi Bahasa Pemrograman COBOL

COBOL (Common Business Oriented Language) adalah suatu bahasa computer awam (High Level language) yang berorientasi langsung kepada

permasalahan bisnis. Cobol diciptakan pada tahun 1959. pengembangan bahasa cobol selanjutnya dilakukan oleh suatu group yang disebut CODASYL, Singkatan dari Conference On Data System Langguage.

Bahasa Cobol pertama – kali diperkenalkan secara formal pada bulan Januari tahun 1960, versi dari bahasa COBOL ini disebut dengan COBOL - 60, dan diperbarui pada tahun 1965 untuk mengatasi hal ini, pada tahun 1968 dan 1974, bahasa cobol dikembangkan dan disempurnakan lebih lanjut dan standarisasi dengan nama Ansi Cobol (American National Standards Institute).

Program bahasa COBOL Merupakan Program terstruktur, yaitu program yang strukturnya jelas, mudah dibaca, dan mudah dipelajari, dan baik untuk didokumentasikan , Struktur Utama dari suatu program Cobol terdiri dari 4 divisi yaitu:

- IDENTIFICATION DIVISION
- ENVIRONMENT DIVISION
- DATA DIVISION
- PROCEDURE DIVISION

Kalau diinginkan Informal mengenai Identitas program (nama Program, Si pembuat, tanggal dibuat, tanggal dikompilasi, dan lainnya) dapat dilihat pada *identification division* Informasi mengenai Keadaan Komputer dan alat – alat lain yang dipergunakan, dapat dilihat pada *environment division*, Informasi mengenai bentuk, jenis dari data apa saja yang dipergunakan dalam program bersangkutan dapat terlihat pada *data division*, *procedure division* memuat prosedur pemrosesan data yang datanya tampak pada data division untuk dihasilkan outputnya.

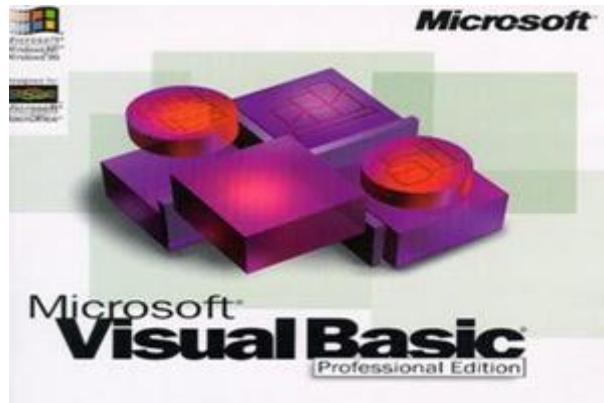
3.1.3.9.1 Yang dapat dikerjakan COBOL

- COBOL dibuat untuk operasi yang mencakup langkah dasar Pengolahan
- data, yaitu membaca data menghasilkan output informasi didalam program COBOL, dua bagian yang utama adalah data division dan Procedure division
- Dari apa yang dapat dikerjakan oleh cobol, konsep COBOL Orientasinya pada permasalahan yang berhubungan dengan pengolahan data.

3.1.3.9.2 Keuntungan Bahasa pemrograman cobol

- Program COBOL dibuat dalam instruksi bahasa inggris, sehingga lebih mudah dipelajari dan dibuat .
- program COBOL sesuai untuk pengolahan data, yang banyak diterapkan pada permasalahan bisnis.
- program COBOL sifatnya standard, sehingga dapat dipergunakan pada computer – computer yang berbeda, tanpa banyak perbedaan.
- struktur program COBOL jelas, sehingga dapat dimengerti oleh orang seperti akuntan, auditor atau manajer – manajer yang hanya mempunyai pengetahuan pengolahan data yang sedikit.
- COBOL menyediakan fasilitas listing program , bila mana perlu dapat diperiksa oleh orang lain selain programmernya.
- mudah didokumentasikan dan dikembangkan bilamana perlu.

3.1.3.10 Definisi Bahasa Pemrograman Microsoft visual basic



Microsoft Visual Basic adalah suatu paket Bahasa Pemrograman tingkat tinggi yang berbasis Under Windows dengan Orientasi Objek Programming (OOP). Maksudnya, program dapat aktif bila ada respon dari pemakai berupa event atau kejadian tertentu.

Setiap Bahasa Pemrograman yang berbasis objek mempunyai IDE, yaitu suatu tampilan Visual tempat bekerja atau membuat program dalam menyelesaikan suatu proyek.

Visual Basic dikembangkan oleh Microsoft sejak tahun 1991, merupakan pengembangan dari pendahulunya yaitu Bahasa Pemrograman Basic (Beginner of All Purpose Symbol) sebagai alat bantu untuk membuat berbagai macam program komputer, khususnya menggunakan sistem operasi Windows.

3.1.3.10.1 Komponen *Microsoft Visual Basic*

Ketika program Visual Basic telah terbuka, maka akan ditemui banyak komponen yang terdapat di dalamnya, yaitu sebagai berikut :

1. Control Menu

Menu yang digunakan untuk memanipulasi jendela Bahasa Pemrograman, dari menu ini kita bisa mengubah ukuran-ukuran, memudahkan menutup jendela Bahasa Pemrograman.

2. Toolbar

Tombol-tombol yang mewakili suatu perintah tertentu.

3. Toolbox

Komponen-komponen yang akan digunakan dalam perancangan program, terdiri dari :

- a. Label, digunakan untuk menampilkan teks yang tidak dapat diedit oleh user.
- b. Text Box, digunakan untuk memasukan dan mengedit teks.
- c. Option button, merupakan bagian dari Optiongroup, digunakan untuk menampilkan beberapa pilhan dimana user hanya dapat memilih satu.
- d. Image, digunakan untuk menampilkan bitmap, Ikon,file JPG atau GIF.
- e. Chek box, digunakan untuk memilih satu pilhan atau lebih secara bersamaan yang disediakan pada program.
- f. ComboBox, digunakan untuk memilih satu pilhan yang disediakan pada program yang menampilkan seluruh pilihan yang tersedia saat objek tersebut diklik.
- g. Command Button, digunakan untuk membuat suatu aksi saat objek tersebut diklik.
- h. List box,digunakan untuk menampilkan dafatar item dimana user dapat memilih dirinya.
- i. Frame, digunakan sebagai wadah Control atau Toolbox yang lain.
- j. Timer, digunakan untuk mengeksekusi event timer dalam interval waktu tertentu.
- k. Data, digunakan untuk menghubungkan dengan sebuah database dan menampilkan informasinya pada form.

4. Form Windows

Daerah kerja utama pada saat membuat suatu program aplikasi akan otomatis tersedia.

5. Properties Windows

Jendela yang digunakan untuk melihat semua informasi objek yang terdapat pada aplikasi Bahasa Pemrograman Berbasis Objek. Properties adalah sifat dari sebuah objek.

6. Project Explorer

Jendela yang digunakan untuk melihat sebuah informasi tentang tampilan form atau tampilan jendela kode.

7. Code Windows

Jendela tempat menulis perintah yang akan dilaksanakan jika suatu objek dijalankan, jendela berbasis kode-kode yang merupakan instruksi Bahasa Pemrograman Visual Basic.

3.1.4 Macam Macam Framework

3.1.4.1 Framework

Framework disebut sebagai kerangka kerja. Para developer menggunakan framework untuk memudahkan mereka dalam membuat dan mengembangkan aplikasi atau software. Pada framework itu sendiri berisi kumpulan fungsi-fungsi dasar atau perintah yang biasa digunakan dalam mengembangkan suatu software, dengan harapan agar software yang dibangun menjadi lebih cepat dan terstruktur.

Ada juga yang mengatakan bahwa framework adalah komponen-komponen yang sudah siap dipakai oleh developer untuk menangani berbagai permasalahan dalam pemrograman, seperti pemanggilan variabel, file, koneksi ke database dan sebagainya. Dengan begitu, developer akan lebih fokus dan mampu menyelesaikan software menjadi lebih cepat dan efektif.

3.1.4.1 Fungsi Framework

4 Menghemat Waktu Pengembangan

Penggunaan framework dalam pengembangan suatu software akan mengurangi beban kerja developer, sehingga tidak ada waktu yang terbuang untuk memikirkan fungsi-fungsi umum yang akan digunakan. Selain itu, developer akan lebih fokus pada alur cerita pada aplikasi seperti yang dibutuhkan oleh pengguna. Dengan begitu, waktu pengembangan software akan berjalan lebih cepat dan dapat diserahkan kepada pengguna sesuai dengan waktu yang telah disepakati bersama.

5 Pemrograman menjadi Lebih Terstruktur

Ketika developer menghadapi suatu proyek pengembangan software yang besar, maka akan terdapat banyak program yang ditulis didalamnya. Terkadang, semakin banyaknya program akan menjadikan proses debugging semakin lambat. Selain itu, untuk mengecek kode program yang menjadi penyebab terjadinya error akan semakin sulit karena program tidak terstruktur dengan baik. ada framework yang menerapkan konsep MVC (Model View Control) yang memudahkan developer untuk memisahkan dan menyusun program berdasarkan bagianya, yaitu Model, View ataupun Control.

6 Pengulangan Kode

Seperti yang sudah disinggung pada poin pertama bahwa pembuatan software membutuhkan waktu yang lama, apalagi Anda berperan sebagai programmer tunggal. Maka Anda dapat menggunakan framework untuk mengurangi beban tersebut.

Sebab, framework sudah menanamkan berbagai fungsi-fungsi umum yang bisa Anda gunakan tanpa harus mengulangi pembuatan kode dari awal. Pastinya, akan memudahkan Anda juga untuk menggunakan kembali di proyek-proyek berikutnya.

7 Meningkatkan Keamanan

Keamanan menjadi suatu hal yang sangat vital dalam pengembangan software atau aplikasi. Apalagi, software yang memuat data pengguna yang privasinya harus dilindungi. Disini, framework terus memperbarui versinya yaitu menawarkan fitur yang handal dalam menangani berbagai jenis ancaman yang menyerang sistem keamanan.

7.1.1.1 Jenis – Jenis Framework

Terdapat dua jenis *framework* yaitu *Desktop Framework* dan *Web Framework*. Kedua jenis framework tersebut memiliki kegunaannya masing-masing. *Desktop framework* merupakan *framework* yang digunakan untuk membangun aplikasi berbasis desktop, sedangkan *web framework* merupakan *framework* yang digunakan untuk membangun aplikasi berbasis web.

1. *Desktop Framework*

a. *Electron*



Gambar 1. 1 Logo Framework Electron

Electron merupakan suatu *framework JavaScript* yang digunakan untuk mengembangkan aplikasi berbasis desktop. Pada umumnya, *JavaScript* identik dengan pembuatan website. Oleh karena itu, *JavaScript* yang

digunakan electron ini dirancang dengan menggunakan teknologi HTML, CSS dan *JavaScript*. Padahal sebenarnya, *electron* juga dapat ditulis dengan menggunakan bahasa C++. Beberapa aplikasi desktop populer yang dibangun oleh *Electron* seperti *Atom*, *Slack*, *WordPress*, *Skype* dan *Github* desktop.

b. *Proton Native*



Gambar 1. 2 Logo Framework Proton Native

Proton Native merupakan *framework JavaScript* yang dikembangkan oleh Gustav Hansen untuk merancang aplikasi berbasis desktop. *Proton Native* memiliki fitur *native tools* yang mempunyai ukuran yang kecil dan penggunaan sumber kekuatan yang lebih sedikit.

2. *Web Framework*

a. Django



Gambar 1. 3 Logo Framework Django

Django merupakan suatu framework yang berbasis Python dengan menggunakan pola MTV, yaitu Model, Template dan View. Django menawarkan berbagai fitur untuk mengembangkan aplikasi web yang berkualitas, terutama bagian keamanan yang digunakan untuk mencegah terjadinya eksekusi program dilapisan Template. Web framework ini sudah banyak digunakan oleh perusahaan besar seperti Disqus, Pinterest, Instagram dan Quora.

b. Angular JS



Gambar 1. 4 Logo Framework Angular JS

Angular JS merupakan framework JavaScript yang sengaja dirilis oleh Google dalam mengembangkan aplikasi web. Bahkan, angular dapat membangun halaman web di sisi client dengan kinerja yang tinggi. Framework ini memiliki konsep MVC (Model, View, Controller). Web framework ini sudah banyak digunakan oleh perusahaan besar seperti Youtube on PS3, Weather dan Netflix.

c. Laravel



Gambar 1. 5 Logo Framework Laravel

Laravel merupakan *framework* MVC yang dibuat oleh Taylor Otwell pada 2011 dengan menggunakan PHP. Framework ini adalah pengembangan *website* berbasis MVP yang dirancang untuk meningkatkan kualitas perangkat lunak, yaitu dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan, meningkatkan pengalaman bekerja dengan aplikasi yang menyediakan sintaks ekspresif, jelas dan menghemat waktu. *Web framework* ini sudah banyak digunakan oleh perusahaan seperti *Deltanet Travel* dan *Neighbourhood Lender*.

d. Rails



Gambar 1. 6 Logo Framework Rails

Rails adalah web framework Model-View-Controller yang ditulis oleh David Heinemeier Hansson. Selain itu, Rails mencakup semua yang diperlukan untuk membuat aplikasi web berbasis database menggunakan

Ruby. Web framework ini sudah banyak digunakan oleh perusahaan seperti AirBnB, GitHub, UrbanDictionary, GroupOn dan Shopify.

e. Spring



Gambar 1. 7 Logo *Framework Spring*

Spring adalah web framework Model-View-Controller yang dikembangkan oleh Pivotal Software. Framework ini dirancang untuk menciptakan aplikasi web berbasis JVM yang sederhana, portabel, cepat dan fleksibel. Web framework ini sudah banyak digunakan oleh perusahaan seperti Mascus dan Allocine.

f. CodeIgniter



Gambar 1. 8 Logo *Framework CodeIgniter*

CodeIgniter merupakan framework yang memiliki konsep MVC (Model, View, Controller) untuk membangun sebuah website yang dinamis menggunakan PHP. Dengan menggunakan codeIgniter, developer akan semakin cepat dalam membangun aplikasi web, walaupun memulainya dari awal. Web framework ini sudah banyak digunakan oleh perusahaan seperti Bufferapp dan The Mail and Guardian.

BAB II PENGENALAN ALGORITMA

1.1 PENGENALAN ALGORITMA BOYER MOORE STRING MATCHING

1.1.1 Pengertian String Matching

String matching adalah pencarian sebuah pattern pada sebuah teks (Cormen, T.H. et al. 1994). String matching digunakan untuk menemukan suatu string yang disebut dengan pattern dalam string yang disebut dengan teks (Charras, C. & Lecroq, T. 1997). Prinsip kerja algoritma string matching (Effendi, D. et al. 2013) adalah sebagai berikut:

1. Memindai teks dengan bantuan sebuah window yang ukurannya sama dengan panjang pattern.
2. Menempatkan window pada awal teks.
3. Membandingkan karakter pada window dengan karakter dari pattern. Setelah pencocokan (baik hasilnya cocok atau tidak cocok) dilakukan pergeseran ke kanan pada window. Prosedur ini dilakukan berulang-ulang sampai window berada pada akhir teks. Mekanisme ini disebut mekanisme sliding window.

Algoritma string matching mempunyai tiga komponen utama (Effendi, D. et al. 2013), yaitu:

1. Pattern, yaitu deretan karakter yang akan dicocokkan dengan teks, dinyatakan dengan $x[0\dots m - 1]$, panjang pattern dinyatakan dengan m .
2. Teks, yaitu tempat pencocokan pattern dilakukan. Dinyatakan dengan $y[0\dots n - 1]$, panjang teks dinyatakan dengan n .
3. Alfabet, berisi semua simbol yang digunakan oleh bahasa pada teks dan pattern, dinyatakan dengan Σ dengan ukuran dinyatakan ASIZE

1.1.1.1 Cara kerja String Matching

Cara yang jelas untuk mencari pattern yang cocok dengan teks adalah dengan mencoba mencari di setiap posisi awal dari teks dan mengabaikan pencarian secepat mungkin jika karakter yang salah ditemukan (Knuth, D.E. et al. 1977). Proses pertama adalah menyelaraskan bagian paling kiri dari pattern dengan teks. Kemudian dibandingkan karakter yang sesuai dari teks dan pattern. Setelah seluruhnya cocok maupun tidak cocok dari pattern, window digeser ke kanan sampai posisi $(n - m + 1)$ pada teks. Menurut Singh, R. & Verma, H.N. (2011), efisiensi dari algoritma terletak pada dua tahap:

1. Tahap praproses, tahap ini mengumpulkan informasi penuh tentang pattern dan menggunakan informasi ini pada tahap pencarian.
2. Tahap pencarian, pattern dibandingkan dengan window dari kanan ke kiri atau kiri ke kanan sampai kecocokan atau ketidakcocokan terjadi.

Dengan sebuah nilai karakter ($m < n$) yang akan dicari dalam teks. Dalam algoritma pencocokan string, teks diasumsikan berada di dalam memori, sehingga bila kita mencari string di dalam sebuah arsip, maka semua isi arsip perlu dibaca terlebih dahulu kemudian disimpan di dalam memori. Jika pattern muncul lebih dari sekali di dalam teks, maka pencarian hanya akan memberikan keluaran berupa lokasi pattern ditemukan pertama kali (Wulan, 2011)

1.1.1.2 Klasifikasi String Matching

Algoritma string matching dapat diklasifikasikan menjadi tiga bagian menurut arah pencarinya (Charras, C. & Lecroq, T. 1997), yaitu:

1. From left to right

Dari arah yang paling alami, dari kiri ke kanan, yang merupakan arah untuk membaca. Algoritma yang termasuk dalam kategori ini adalah algoritma Brute Force, algoritma Morris dan Pratt yang kemudian dikembangkan menjadi algoritma Knuth-Morris-Pratt.

2. From right to left

Dari arah kanan ke kiri, arah yang biasanya menghasilkan hasil terbaik secara partikular. Contoh algoritma ini adalah algoritma Boyer-Moore, yang kemudian banyak dikembangkan menjadi algoritma Tuned Boyer-Moore, algoritma Turbo Boyer-Moore, algoritma Zhu Takaoka dan algoritma Horspool.

3. In a specific order

Dari arah yang ditentukan secara spesifik oleh algoritma tersebut, arah ini menghasilkan hasil terbaik secara teoritis. Algoritma yang termasuk kategori ini adalah algoritma Colussi dan algoritma Crochemore-perrin.

1.1.1.3 Teknik String Matching

Menurut Singla, N. & Garg, D. (2012), ada dua teknik utama dalam algoritma string matching, yaitu:

1. Exact string matching.

Exact string matching, merupakan pencocokan string secara tepat dengan susunan karakter dalam string yang dicocokkan memiliki jumlah maupun urutan karakter dalam string yang sama. Bagian algoritma ini bermanfaat jika pengguna ingin mencari string dalam dokumen yang sama persis

dengan string masukan. Beberapa algoritma exact string matching antara lain:

A. Knuth-Morris-Pratt

Metode ini mencari kehadiran sebuah kata dalam teks dengan melakukan observasi awal (preprocessing) dengan cara mengecek ulang kata sebelumnya. Algoritma ini melakukan pencocokan dari kiri ke kanan.

B. Boyer-Moore Algoritma

Boyer-Moore adalah algoritma string matching yang paling efisien dibandingkan algoritma string matching lainnya. Sebelum melakukan pencarian string, algoritma melakukan proses terlebih dahulu pada pattern, bukan pada string pada teks tempat pencarian. Algoritma ini melakukan pencocokan karakter yang dimulai dari kanan ke kiri. Karena sifatnya yang sangat efisien, Boyer-Moore memiliki banyak variasi penyederhanaannya. Salah satunya adalah algoritma Horspool yang akan digunakan dalam penelitian ini dan akan dijelaskan pada poin berikutnya

C. Approximate string matching atau Fuzzy string matching.

Fuzzy string matching merupakan pencocokan string secara samar, maksudnya pencocokan string dimana string yang dicocokkan memiliki kemiripan memiliki susunan karakter yang berbeda (mungkin jumlah atau urutannya), tetapi string tersebut memiliki kemiripan baik kemiripan tekstual/penulisan (approximate string matching) atau kemiripan ucapan (phonetic string matching).

1.1.2 Pengantar Algoritma Boyer Moore

Algoritma Boyer-Moore diperkenalkan oleh Bob Boyer dan J.S. Moore pada tahun 1977. Pada algoritma ini pencocokan kata dimulai dari karakter terakhir kata kunci menuju karakter awalnya. Jika terjadi perbedaan antara karakter terakhir kata kunci dengan kata yang dicocokkan maka karakter-karakter dalam potongan kata yang dicocokkan tadi akan diperiksa satu per satu. Hal ini dimaksudkan untuk mendeteksi apakah ada karakter dalam potongan kata tersebut yang sama dengan karakter yang ada pada kata kunci. Apabila terdapat kesamaan, maka kata kunci akan digeser sedemikian rupa sehingga posisi karakter yang sama terletak sejajar, dan kemudian dilakukan kembali pencocokan karakter terakhir dari kata kunci.

Sebaliknya jika tidak terdapat kesamaan karakter, maka seluruh karakter kata kunci akan bergeser ke kanan sebanyak m karakter, di mana m adalah panjang karakter dari kata kunci. Boyer-Moore merupakan salah satu Algoritma Pattern Matching yang cukup terkenal. Algoritma ini menggunakan beberapa kasus pengecekan teks (input karakter yang akan dibaca) dengan Pattern (pola yang akan disaring). Algoritma Boyer-Moore adalah algoritma pencarian string yang mencari dengan cara membandingkan sebuah huruf dengan huruf yang ada di pattern yang dicari, dan menggeser pattern tersebut hingga posisinya sama dengan teks yang dicari dan membandingkan kata tersebut. Cara ini disebut character jump. Algoritma Pattern Matching Boyer-Moore ini berbasis pada 2 metode yaitu :

1. The Looking-Glass Technique

The Looking-Glass Technique melakukan perbandingan suatu karakter akhir pada kata w dengan suatu karakter pada teks s . Jika karakter tersebut sama maka jendela karakter akan berjalan mundur pada kedua string dan mengecek kembali kedua karakter. Mencari Suatu kecocokan String pada

Teks dengan pola yang akan dicari dengan cara memindahkan atau menggesernya sampai Teks string selesai.

2. The Character-Jump Technique

Character-jump Technique melakukan suatu aksi ketika perbandingan antara dua karakter yang berbeda. Ada dua aksi yang tergantung pada teks s dan kata w yang dimiliki; jika p yaitu karakter pada s yang sedang diproses yang tidak cocok maka ada dua kemungkinan aksi. Mencari karakter yang sesuai dan cara penggeseran sebuah karakter perbandingan terakhir.

Algoritma Boyer-Moore ini adalah pembandingan karakter dalam sebuah string yang dilakukan dari belakang ke depan atau dari kanan ke kiri karakter. Jika algoritma Boyer Moore membandingkan teks “MAKALAH” misalnya, algoritma ini melakukan pengecekan apakah karakter ke tujuh dari teks yang dibandingkan adalah karakter ‘H’. Jika karakter ketujuh adalah ‘H’ maka ia akan melakukan pengecekan apakah karakter sebelumnya (ke-6) adalah ‘A’. Demikian seterusnya hingga menemukan bahwa karakter pertama adalah ‘M’.

Alasan kenapa Boyer-Moore melakukan pengecekan dari belakang akan lebih jelas jika kita mengamati apa yang terjadi jika pengecekan menghasilkan nilai yang tidak sama. Misalnya, algoritma mendapati karakter ke-8 dengan karakter ‘X’ bukannya ‘H’. ‘H’ tidak muncul sama sekali pada “MAKALAH”, dan ini berarti tidak ada kesamaan sama sekali karakter ‘X’ dengan semua karakter dalam “MAKALAH”. Sehingga, dengan hanya melakukan sekali pengecekan pada karakter ke-8 kita dapat mengabaikan pengecekan karakter ke-1 sampai ke-7 dan langsung melanjutkan pengecekan karakter dimulai dari karakter ke-9, tepat setelah ‘X’.

Algoritma ini pada awalnya melakukan perhitungan sebuah tabel untuk menentukan banyaknya ‘loncatan’ karakter yang akan dilakukan setelah

mendapati sebuah perbandingan yang tidak cocok. Karakteristik utama dari algoritma Boyer-Moore adalah algoritma ini melakukan pencocokan string mulai dari kanan (belakang). dengan karakteristik tersebut, ketidakcocokan saat terjadi perbandingan string akan membuat pergerakan pattern melompat lebih jauh untuk menghindari perbandingan karakter pada string yang diperkirakan gagal. Sehingga proses pencarian string akan lebih optimal.

1.1.2.1 Rumus Algoritma Boyer Moore

Misalnya ada sebuah usaha pencocokan yang terjadi pada **Teks** $[i..i+n-1]$ dan anggap ketidakcocokan pertama terjadi di antara **Teks** $[i+j]$ dan **Pattern** $[j]$, dengan $0 < j < n$. Berarti, **Teks** $[i+j+1..i+n-1] = \text{Pattern } [j+1..-1]$ dan $a = \text{teks } [i+j]$ tidak sama dengan $b = \text{pattern } [j]$, jika u adalah akhiran dari pattern sebelum b dan v adalah sebuah awalan dari pattern maka pergeseran – pergeseran[13] yang mungkin adalah :

1. Penggeseran good-suffix yang terdiri dari menyejajarkan potongan **Teks** $[i+j+1..i+n-1] = \text{Pattern } [j+1..-1]$ dengan kemunculan paling kanan di pattern yang didahului oleh karakter yang berbeda dengan **pattern** $[j]$. Jika tidak ada potongan seperti itu, maka algoritme akan menyejajarkan akhiran u dari **Teks** $[i+j+1..i+n-1]$ dengan awalan dari pattern yang sama.
2. Pergeseran Bad-Character yang terdiri dari menyejajarkan **teks** $[i+j]$ dengan kemunculan paling kanan karakter tersebut di pattern. Bila karakter tersebut tidak ada di pattern, maka pattern akan disejajarkan dengan **Teks** $[i+n-1]$

Contoh:

Pattern yang akan dicocokan adalah: TOOTH

Dari kata TOOTH memiliki 5 panjang karakter maka dapat dihitung bahwa

$$\text{NILAI T} = 5 - 0 - 1 = 4$$

↓
T O O T H Length = 5
 0 1 2 3 4

Letter	T	O	H	*
Value	4			

$$\text{NILAI O} = 5 - 1 - 1 = 3$$

↓
T O O T H Length = 5
 0 1 2 3 4

Letter	T	O	H	*
Value	4	3		

$$\text{NILAI O ke } 2 = 5 - 2 - 1 = 2$$

↓
T O O T H Length = 5
 0 1 2 3 4

Letter	T	O	H	*
Value	4	2		

NILAI T Berubah menjadi 1 karena mengambil perhitungan terbaru dari huruf yang sama

NILAI H = 5 karena merupakan Kata Terakhir maka ketentuan kata terakhir yaitu Panjang kata yaitu 5

Dan nilai * Merupakan nilai pergeseran seterusnya jika text yang di cari tidak di temukan

T O O T H ↓ 0 1 2 3 4	Length = 5			
Letter Value	T 1	o 2	H 5	* 5

H = 5 Last letter = length if not already defined, else, leave

1.1.2.2 Proses Pergeseran Algoritma Boyer Moore

Contoh cara kerja algoritma Boyer Moore ini adalah sebagai berikut :

Teks (Y) : TRUSTHARDTOOTHBRUSHES

Pattern (X): TOOTH

Letter Value	T 1	o 2	H 5	* 5
-------------------------------	----------------------	----------------------	----------------------	----------------------



Pada langkah pertama penyesuaian karakter akan dimulai pada karakter (X) H dimana karakter (Y) T tidak sama dengan karakter di atas (good-suffix) yaitu T dan T memiliki Nilai pergeseran 1 maka akan di geser sebanyak 1 kali Sesuai dengan Nilai Letter Diatas.

Letter	T	O	H	*
Value	1	2	5	5

T R U S T H A R D T O O T H B R U S H E S
 ↑ ↑ ↑
 T O O T H

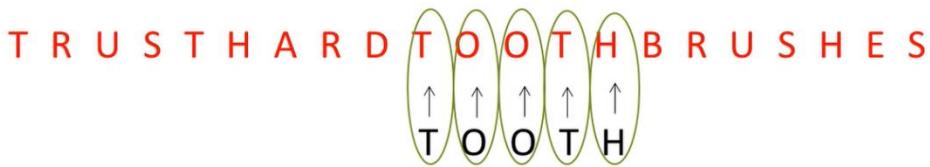
Setelah melakukan pergeseran 1 kali dapat dilihat bahwa karakter (X) H dengan karakter (Y) H maka ada kecocokan(good-suffix), lalu geser dengan karakter (x) T dan karakter (Y) T maka ada kecocokan (good-suffix), lalu geser kembali dengan karakter (x) O dan karakter (Y) S , dapat dilihat tidak ada kecocokan antara (X) dengan (Y) (Bad-suffix) dan pergeseran akan di ambil dari Karakter H karena pada tabel letter tidak terdapat karakter S maka akan dilakukan pergeseran yaitu sebanyak 5 Pergeresan.

Letter	T	O	H	*
Value	1	2	5	5

T R U S T H A R D T O O T H B R U S H E S
 ↑
 T O O T H

Setelah melakukan pergeseran 5 kali dapat dilihat bahwa karakter (X) H dengan karakter (Y) T maka ada tidak ada kecocokan (Bad-suffix), maka akan dilakukan pergeseran Sebanyak 1 kali karena karakter T Memiliki nilai pergeseran yaitu sebanyak 1.

Letter	T	O	H	*
Value	1	2	5	5



Setelah melakukan pergeseran 1 kali dapat dilihat bahwa karakter (X) H dengan karakter (Y)H maka ada kecocokan(Good-suffix), karakter (X) T dengan karakter (Y) T maka ada kecocokan (Good-suffix), karakter (X) O dengan karakter (Y) O maka ada kecocokan (Good-suffix), karakter (X) O dengan karakter (Y) O maka ada kecocokan (Good-suffix), karakter (X) T dengan karakter (Y) T maka ada kecocokan (Good-suffix), Maka dinyatakan Karakter (X) dan Karakter (Y) sesuai dan dapat dinyatakan *String Matching*.

Algoritma Boyer Moore banyak digunakan untuk penyelesaikan masalah pencarian string pada dokumen bahkan pencarian menggunakan search engine di internet telah menjadi sangat populer. Beberapa bidang diantaranya manufaktur, control process, dan ekonomi. Berikut adalah penelitian yang sudah dilakukan dengan menggunakan algoritma Boyer Moore :

1. Pengenalan Sidik Jari dengan menggunakan Algoritma Pencocokan String Boyer Moore. Pada penelitian ini algoritma Boyer Moore digunakan untuk menangani pengenalan sidik jari dalam biometric untuk proses identifikasi seseorang. Algoritma pencocokan string Boyer-Moore dengan pattern dicocokkan dari kanan ke kiri memungkinkan informasi yang didapat akan lebih banyak. Pemanfaatkan algoritma Boyer-Moore dalam struktur sidik jari sebagai string biner yang dicocokkan untuk menemukan suatu

keakuratan. Image sidik jari diubah menjadi bilangan biner untuk selanjutnya diproses dengan algoritma Boyer Moore.

2. Kinerja algoritma paralel untuk Pencarian Kata dengan metode Boyer Moore menggunakan PVM [2]. Pada penelitian ini algoritma Boyer Moore digunakan untuk menangani permasalahan dalam kegiatan penting dalam pemrosesan data mengenai optimasi komputasi parallel dengan bantuan parallel virtual machine dalam mengkombinasikan pencarian string dengan metode Boyer Moore akan didapat peningkatan kinerja komputasi untuk kata kunci yang terdiri dari satu huruf saja.
3. Pencarian nama yang memiliki kesamaan fonetik menggunakan Algoritma Kesamaan String Boyer Moore[3]. Pada penelitian ini, kesamaan fonetik yang dibahas adalah merupakan ukuran kesamaan yang sangat tergantung pada bahasa yang digunakan dalam penerapannya pada perkembangan ejaan bahasa Indonesia. Perkembangan ejaan yang terjadi di Indonesia yaitu terdapat beberapa ejaan bahasa Indonesia, diantaranya
 - 1). Ejaan Van Ophuysen, Ejaan ini digunakan untuk menuliskan kata-kata Melayu menurut model yang dimengerti oleh orang Belanda, yaitu menggunakan huruf Latin dan bunyi yang mirip dengan tuturan Belanda, diantaranya adalah kata jang, pajah, sajang, goeroe, itoe, oemoer.
 - 2). Ejaan Republik (Ejaan Soewandi), adalah ketentuan ejaan dalam Bahasa Indonesia yang berlaku sejak 17 Maret 1947. Ejaan ini kemudian juga disebut dengan nama edjaan Soewandi, Menteri Pendidikan dan Kebudayaan kala itu. Ejaan inimengganti ejaan sebelumnya, yaitu Ejaan Van Ophuijsen yang mulai berlaku sejak tahun 1901. Perbedaan antara ejaan ini dengan ejaan Van Ophuijsen ialah huruf 'oe' menjadi 'u', seperti pada goeroe → guru,
 - 3). Ejaan Melindo adalah sistem ejaan Latin yang termuat dalam Pengumuman Bersama Edjaan Bahasa Melaju-Indonesia (Melindo) (1959)

sebagai hasil usaha penyatuan sistem ejaan dengan huruf Latin di Indonesia dan Persekutuan Tanah Melayu, sebagai contoh kata “tj” dalam bahasa Indonesia dan “ch” dari bahasa Melayu berubah menjadi “c”,

4). Ejaan Baru (Ejaan LBK), Merupakan lanjutan dari rintisan panitia ejaan melindo, sebagai contohnya adalah kata “remadja” menjadi “remaja”, dan

5). Ejaan Bahasa Indonesia Yang di sempurnakan (EYD) sebagai contoh “wadjar” menjadi “wajar”. Jika hal ini terjadi pada penulisan nama, maka pendeksiyan akan mengalami kesulitan dalam fonetik. Oleh sebab itu algoritma Boyer Moore digunakan untuk menentukan varian huruf berdasarkan ejaan yang baru. Jadi huruf-huruf yang tadinya masih menggunakan ejaan yang lama akan langsung diproses menjadi huruf dengan ejaan yang baru, yang sesuai dengan ejaan baru Bahasa Indonesia untuk kemudian diproses pencarian string-nya dengan cara kerja algoritma Boyer Moore, yaitu mengecheck pattern dari kanan ke kiri.

BAB III PENJELASAN TOOLS DAN BAHASA PEMROGRAMAN

1.1 PENJELASAN TOOLS DAN BAHASA PEMOGRAMAN YANG DIGUNAKAN

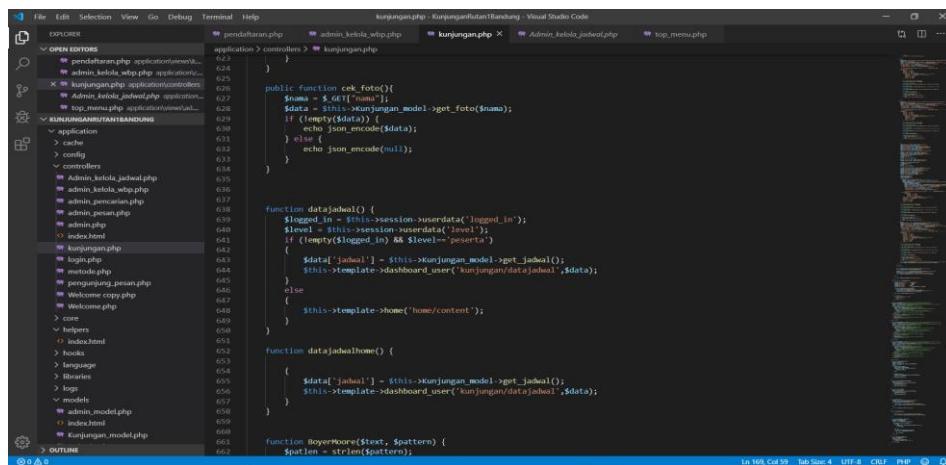
1.1.1 Tools Yang Digunakan

1.1.1.1 Visual Studio Code

Aplikasi Visual Studio Code Text merupakan aplikasi editor yang digunakan untuk kode dan teks yang dapat berjalan diberbagai platform operating system dengan menggunakan teknologi Phyton API. Aplikasi ini diciptakan karena terinspirasi dari aplikasi Vim. Aplikasi ini sangatlah fleksibel dan powerfull. Aplikasi ini dapat dikembangkan dengan menggunakan Visual Studio Code-packages.

Visual Studio Code Text bukanlah aplikasi opensource dan juga aplikasi yang dapat digunakan dan didapatkan secara gratis, namun beberapa fitur

pengembangan fungsionalitas (packages) dari aplikasi ini merupakan hasil dari temuan dan memperoleh dukungan penuh dari komunitas serta memiliki lisensi aplikasi gratis. Aplikasi ini mendukung berbagai bahasa pemrograman dan mampu menyajikan fitur syntax highlight hampir di semua bahasa pemrogramman yang didukung ataupun dikembangkan oleh komunitas.



Gambar 3. 1 Tampilan Visual Studio Code

Berikut beberapa fitur yang diunggulkan dari aplikasi Visual Studio Code

Text:

- Goto Anything

Fitur yang sangat membantu dalam membuka file ataupun menjelajahi isi dari file hanya dengan beberapa keystrokes.

- Multiple Selections

Fitur ini memungkinkan user untuk mengubah secara interaktif banyak baris sekaligus, mengubah nama variabel dengan mudah, dan memanipulasi file lebih cepat dari sebelumnya.

- Command Pallete

Dengan hanya beberapa keystrokes, user dapat dengan cepat mencari fungsi yang diinginkan, tanpa harus menavigasi melalui menu.

- Distraction Free Mode
- Bila user memerlukan fokus penuh pada aplikasi ini, fitur ini dapat membantu user dengan memberikan tampilan layar penuh.
- Split Editing
- Dapatkan hasil yang maksimal dari monitor layar lebar dengan dukungan editing perpecahan. Melakukan editing di sisi file dengan sisi, atau mengedit dua lokasi di satu file. Anda dapat mengedit dengan banyak baris dan kolom yang user inginkan.
- Instant Project Switch

Mengambil seluruh file yang dimasukkan kedalam project pada aplikasi ini terhubung dengan fitur Goto Anything untuk menjelajahi semua file yang ada ataupun untuk beralih ke file dalam project lainnya dengan cepat.
- Plugin API

Aplikasi ini memiliki plugin API berbasis Phyton sehingga membuat aplikasi ini sangat tangguh.
- Customize Anything

Aplikasi Visual Studio Code mempunyai dan memberikan user fleksibilitas dalam hal pengaturan fungsional dalam aplkasi ini.
- Cross Platform

Aplikasi ini dapat berjalan hampir disemua operating system modern seperti Windows, OS X, dan Linux based operating system.

1.1.1.2 Xampp

XAMPP adalah sebuah paket perangkat lunak (software) komputer yang sistem penamaannya diambil dari akronim kata Apache, MySQL (dulu) / MariaDB (sekarang), PHP, dan Perl. Sementara imbuhan huruf “X” yang terdapat pada awal kata berasal dari istilah cross platform sebagai simbol

bahwa aplikasi ini bisa dijalankan di empat sistem operasi berbeda, seperti OS Linux, OS Windows, Mac OS, dan juga Solaris.

Sejarah mencatat, software XAMPP pertama kali dikembangkan oleh tim proyek bernama Apache Friends dan sampai saat ini sudah masuk dalam rilis versi 7.3.9 yang bisa didapatkan secara gratis dengan label GNU (General Public License).

Jika dijabarkan secara gamblang, masing-masing huruf yang ada di dalam nama XAMPP memiliki arti sebagai berikut ini:

X = Cross Platform

Merupakan kode penanda untuk software cross platform atau yang bisa berjalan di banyak sistem operasi.

A = Apache

Apache adalah aplikasi web server yang bersifat gratis dan bisa dikembangkan oleh banyak orang (open source).

M = MySQL / MariaDB

MySQL atau MariaDB merupakan aplikasi database server yang dikembangkan oleh orang yang sama. MySQL berperan dalam mengolah, mengedit, dan menghapus daftar melalui database.

P = PHP

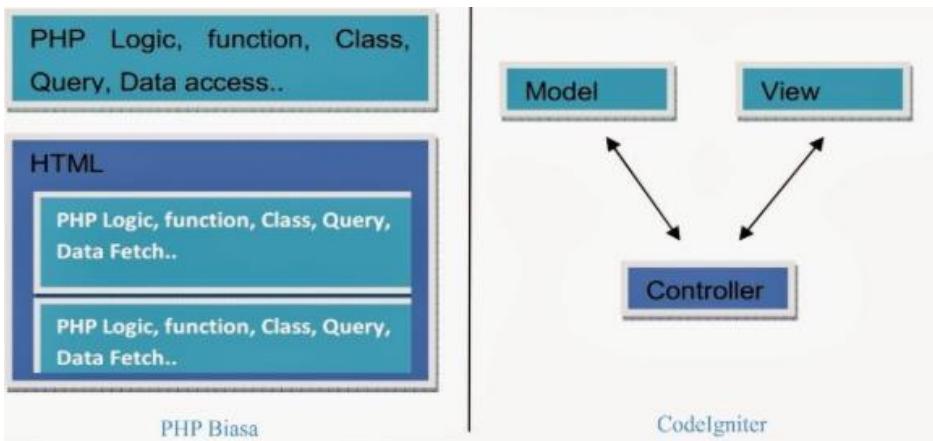
Huruf “P” yang pertama dari akronim kata XAMPP adalah inisial untuk menunjukkan eksistensi bahasa pemrograman PHP. Bahasa pemrograman ini biasanya digunakan untuk membuat website dinamis, contohnya dalam website berbasis CMS WordPress.

Program aplikasi XAMPP berfungsi sebagai server lokal untuk mengampu berbagai jenis data website yang sedang dalam proses pengembangan. Dalam prakteknya, XAMPP bisa digunakan untuk menguji kinerja fitur ataupun menampilkan konten yang ada didalam website kepada orang lain tanpa harus terkoneksi dengan internet, atau istilahnya website offline. XAMPP bekerja secara offline layaknya web hosting biasa namun tidak bisa diakses oleh banyak orang. Maka dari itu, XAMPP biasanya banyak digunakan oleh para mahasiswa maupun pelajar untuk melihat hasil desain website sebelum akhirnya dibuat online menggunakan web hosting yang biasa dijual dipasaran.

1.1.2 Framework codeigniter

CodeIgniter adalah sebuah web application network yang bersifat open source yang digunakan untuk membangun aplikasi php dinamis. CodeIgniter menjadi sebuah framework PHP dengan model MVC (Model, View, Controller) untuk membangun website dinamis dengan menggunakan PHP yang dapat mempercepat pengembang untuk membuat sebuah aplikasi web. Selain ringan dan cepat, CodeIgniter juga memiliki dokumentasi yang super lengkap disertai dengan contoh implementasi kodennya. Dokumentasi yang lengkap inilah yang menjadi salah satu alasan kuat mengapa banyak orang memilih CodeIgniter sebagai framework pilihannya. Karena kelebihan-kelebihan yang dimiliki oleh CodeIgniter, pembuat PHP Rasmus Lerdorf memuji CodeIgniter di frOSCon (Agustus 2008) dengan mengatakan bahwa dia menyukai CodeIgniter karena “it is faster, lighter and the least like a framework.”

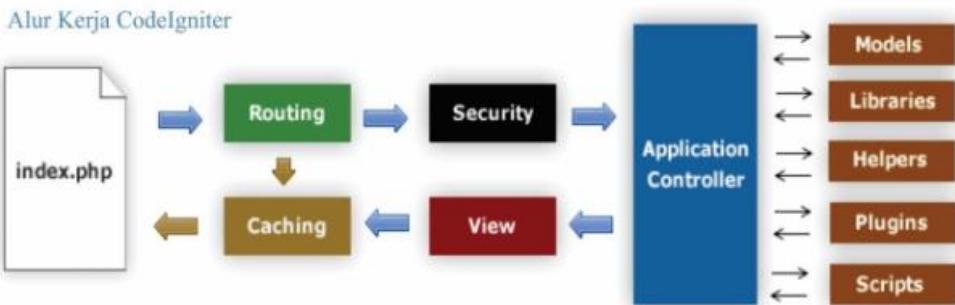
CodeIgniter pertamakali dikembangkan pada tahun 2006 oleh Rick Ellis. Dengan logo api yang menyala, CodeIgniter dengan cepat “membakar” semangat para web developer untuk mengembangkan web dinamis dengan cepat dan mudah menggunakan framework PHP yang satu ini.



Gambar 3. 2 MVC

1.1.2.1 Cara Kerja

Alur Kerja Framework Codeigniter



Gambar 3. 3 Alur kerja Codeigniter

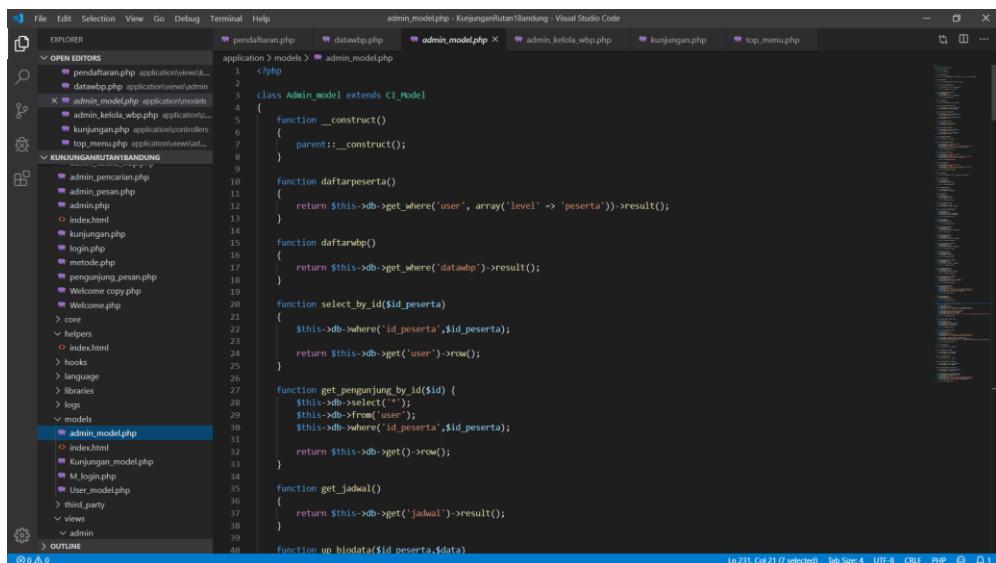
- Index.php: Index.php disini berfungsi sebagai file pertama dalam program yang akan dibaca oleh program.
- The Router: Router akan memeriksa HTTP request untuk menentukan hal apa yang harus dilakukan oleh program.
- Cache File: Apabila dalam program sudah terdapat “cache file” maka file tersebut akan langsung dikirim ke browser. File cache inilah yang dapat membuat sebuah website dapat di buka dengan lebih cepat. Cache file

dapat melewati proses yang sebenarnya harus dilakukan oleh program codeigniter.

- Security: Sebelum file controller di load keseluruhan, HTTP request dan data yang disubmit oleh user akan disaring terlebih dahulu melalui fasilitas security yang dimiliki oleh codeigniter.
- Controller: Controller akan membuka file model, core libraries, helper dan semua resources yang dibutuhkan dalam program tersebut.
- View: Hal yang terakhir akan dilakukan adalah membaca semua program yang ada dalam view file dan mengirimkannya ke browser supaya dapat dilihat. Apabila file view sudah ada yang di “cache” maka file view baru yang belum ter-cache akan mengupdate file view yang sudah ada.

Contoh File untuk Model, View dan Controller:

1. Model/Admin



The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "KUNJUNGANRUTIN". It includes files like pendaftran.php, datawbp.php, admin_model.php, admin_kelela_wbp.php, kunjungan.php, top_menu.php, and various view files (index.php, login.php, metode.php, pengunjung_pesan.php, Welcome.php).
- Code Editor:** The active file is "admin_model.php". The code is as follows:

```
<?php
class Admin_model extends CI_Model
{
    function __construct()
    {
        parent::__construct();
    }

    function daftarpeserta()
    {
        return $this->db->get_where('user', array('level' => 'peserta'))->result();
    }

    function daftarwbp()
    {
        return $this->db->get_where('datawbp')->result();
    }

    function select_by_id($id_peserta)
    {
        $this->db->where('id_peserta',$id_peserta);
        return $this->db->get('user')->row();
    }

    function get_pengunjung_by_id($id) {
        $this->db->select('*');
        $this->db->from('user');
        $this->db->where('id_peserta',$id_peserta);
        return $this->db->get()->row();
    }

    function get_jadwal()
    {
        return $this->db->get('jadwal')->result();
    }

    function update_userdata($id_peserta,$data)
}
```

The status bar at the bottom indicates "In 231, Col 21 (7 selected)" and "Tab Size: 4 - UTF-8 - OLF - PHP - Q 1".

Gambar 3. 4 Alur kerja Codeigniter

2. View/daftar_peserta

File Edit Selection View Go Debug Terminal Help

dattarpeserta.php - KunjunganRutanBandung - Visual Studio Code

EXPLORER

OPEN EDITORS

dattarpalih.php dattarwsp.php dattarpeserta.php admin_kelola_wbp.php kunjungan.php top_menu.php

application > views > admin > dattarpeserta.php

```
<div class="container">
    <h1 class="header smaller lighter blue">Daftar Keseluruhan Pengunjung : <?php echo $jumlah;?></h1>
    <table class="table table-striped table-bordered data" style="width:100%">
        <thead>
            <tr>
                <th>NIK</th>
                <th>USERNAME</th>
                <th>Nama Lengkap</th>
                <th>Jenis Kelamin</th>
                <th>Status Hubungan</th>
                <th>Email</th>
                <th>Foto</th>
                <th>Download Foto</th>
                <th align="center"><a href=<?php echo site_url('admin/tambahpeserta');?>>Tambah Peserta</a></th>
            </tr>
        </thead>
        <tbody>
            <?php foreach ($peserta as $peserta) >
            <tr>
                <td><?php echo $peserta->nik;?></td>
                <td><?php echo $peserta->username;?></td>
                <td><?php echo $peserta->namalengkap;?></td>
                <td><?php echo $peserta->jenis_kelamin;?></td>
                <td><?php echo $peserta->status_hubungan;?></td>
                <td><?php echo $peserta->email;?></td>
                <td class="text-center">
                    <a href=<?php echo base_url();?>/uploads/kk/<?php echo $peserta->dokumen; ?> target="_blank"><img src=<?php echo base_url();?>/uploads/kk/<?php echo $peserta->dokumen; ?>?>></a>
                </td>
                <td>Output one or more strings:</td>
                <td><a href=<?php echo string_1, $string_2;?> void setra->dokumen; ?>" download=Download Foto</a></td>
                <td><a href=<?php echo site_url('admin/biodatapeserta');?>>Selengkapnya | <a href=<?php echo site_url('admin/editpeserta');?>>Edit | <a href=<?php echo site_url('admin/hapus_peserta');?>>Hapus </a></td>
                <td><a href=<?php echo site_url('admin/kunjungan');?>>Data Kunjungan</a></td>
            </tr>
        </tbody>
    </table>

```

Gambar 3. 5 View

3. Controller/admin

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** On the left, it lists the project structure:
 - EXPLORER**
 - OPENED EDITORS**
 - pendaftaran.php (application/view)
 - datawbp.php (application/view/admin)
 - admin.php** (application/controllers)
 - admin_belia.wbp.php (application/controllers)
 - kunjungan.php (application/controllers)
 - top_menu.php (application/news/ad...)
 - KUNJUNGANAKTIFITBANDUNG**
 - application
 - > cache
 - > config
 - > controllers
 - Admin_belia_jadwal.php
 - admin_kelola.wbp.php
 - admin_pencarian.php
 - admin_pesan.php
 - admin.php**
 - index.html
 - kunjungan.php
 - login.php
 - metode.php
 - pengunjung_pesan.php
 - Welcome_copy.php
 - Welcome.php
 - > core
 - > helpers
 - > index.html
 - > hooks
 - > language
 - > libraries
 - > logs
 - > models
 - admin_model.php
 - index.html
 - > OUTLINE
- Code Editor:** The main area displays the **admin.php** file content, which is a PHP script for managing users. It includes imports for Admin, CI_Controller, and various models like Admin_model, Kunjungan_model, and User_model. It defines functions for listing users (daftarpeserta) and adding new users (daftaverifikasi).
- Status Bar:** At the bottom, it shows "Line 369 Col 84 14 selected" and icons for Tab Size 4, UTF-8, CMF, PHP, and a search icon.

Gambar 3. 6 controller

BAB IV INSTALASI TOOLS YANG DIGUNAKAN

4.1 INSTALASI TOOLS YANG DIGUNAKAN

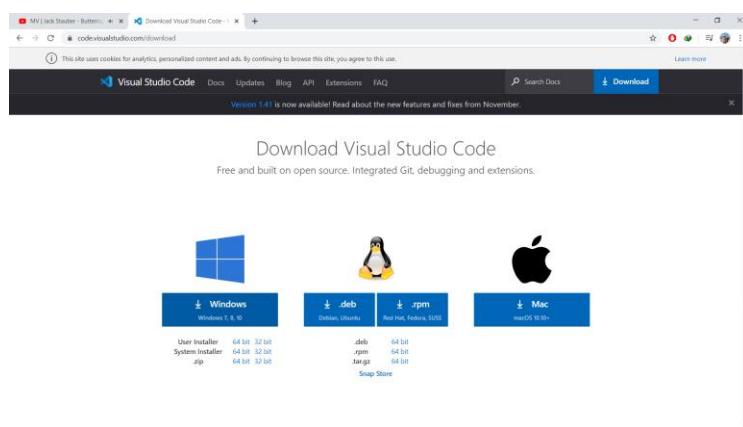
4.1.1 Tools Yang Digunakan

4.1.1.1 Instalasi Visual Studio Code

Berikut adalah langkah-langkah untuk melakukan instalasi Visual Studio Code :

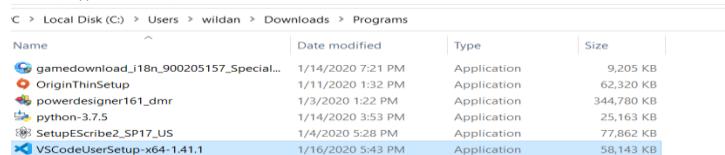
1. Tahapan yang pertama untuk melakukan instalasi maka download dahulu file .exe pada link berikut :

<https://code.visualstudio.com/download>



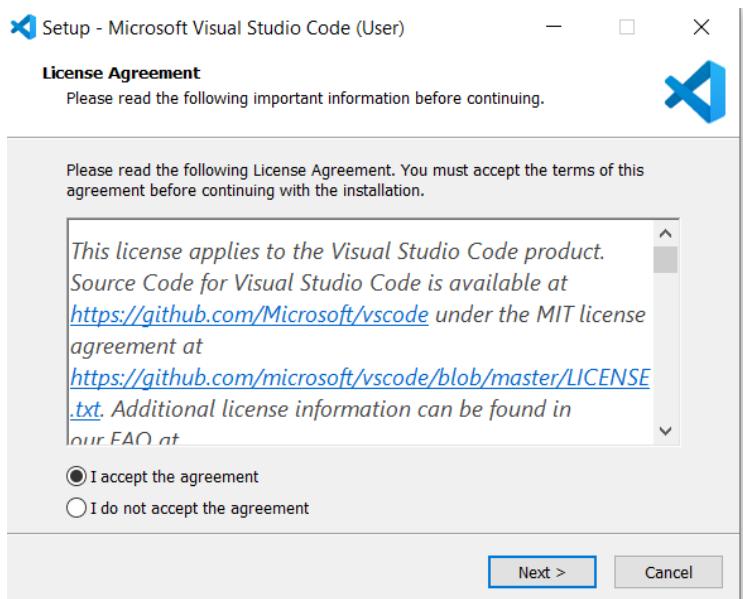
Gambar 4. 1 Downlod file .exe

2. Tahapan yang kedua setelah selesai mendownload maka buka directory dimana file .exe tersimpan, kemudian Double klik file .exe hasil download tadi :



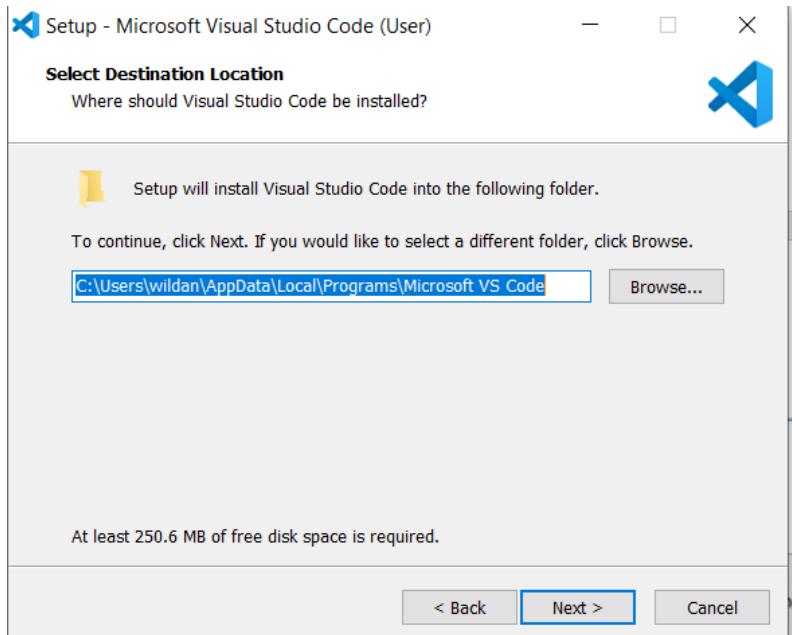
Gambar 4. 2 Double Klik File .exe

3. Tahapan ketiga jika sudah double klik file .exe maka akan muncul tampilan berikut kemudian klik next :



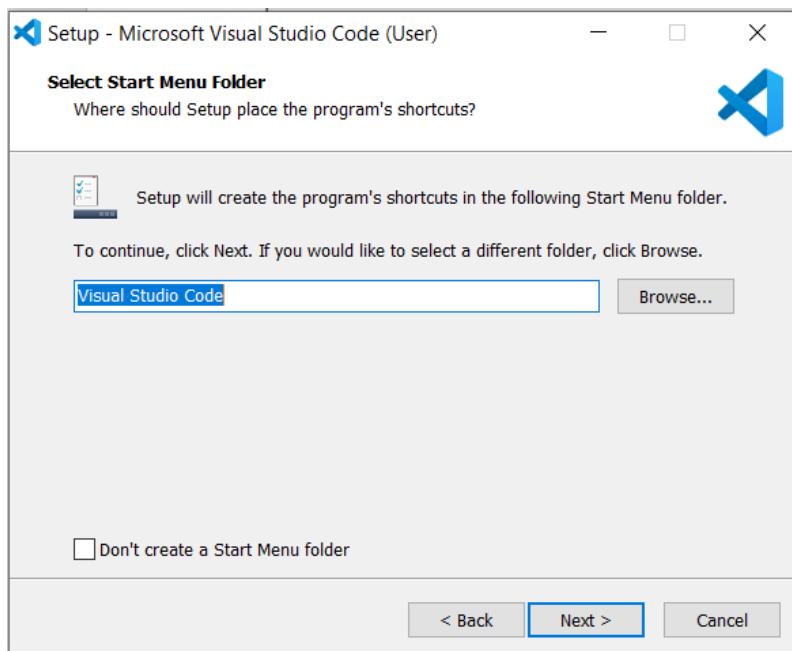
Gambar 4. 3 Proses Instalasi Visual Studio Code

4. Selanjutnya klik next



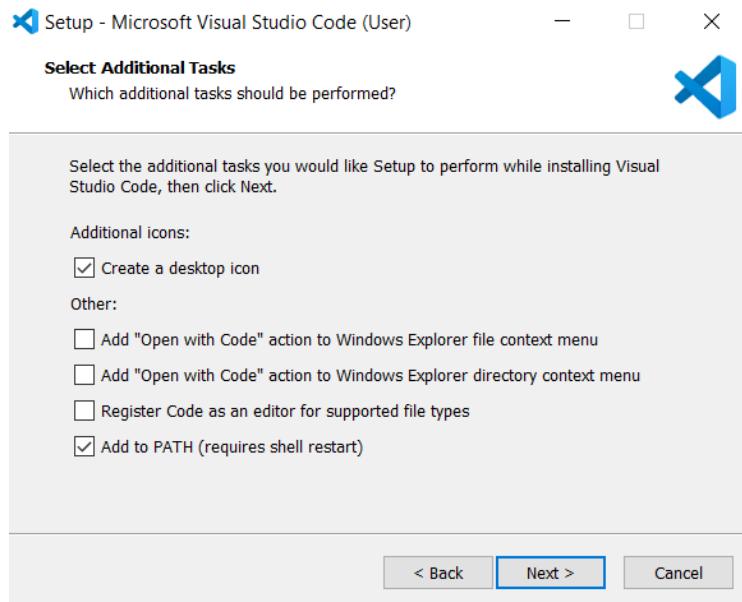
Gambar 4. 4 Proses Instalasi Visual Studio Code

5. Selanjutnya klik next



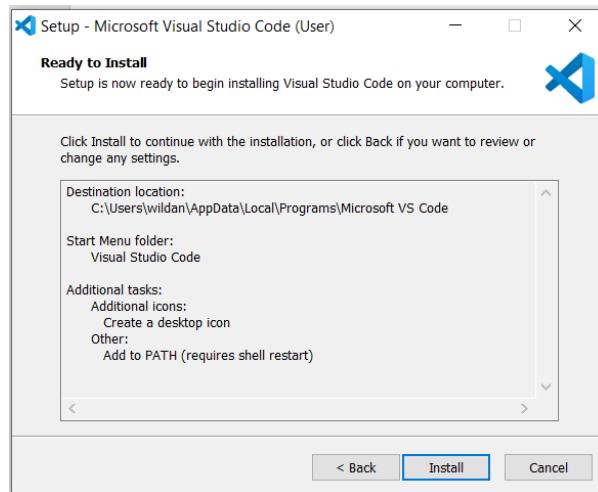
Gambar 4. 5 Proses Instalasi Visual Studio Code

6. Selanjutnya klik next



Gambar 4. 6 Proses Instalasi Visual Studio Code

7. Tekan Install lalu Tunggu sampai proses instalasi selesai, Setelah proses ini selesai maka Visual Studio Code telah bisa digunakan.



Gambar 4. 7 Proses Instalasi Visual Studio Code

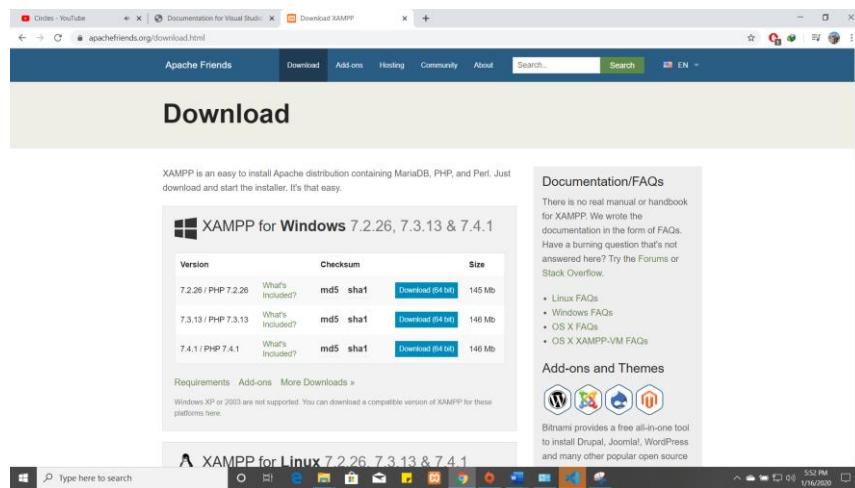
4.1.1.2 Xampp

Berikut adalah langkah-langkah untuk melakukan instalasi Xampp :

1. Untuk melakukan instalasi Xampp maka pertama-tama melakukan download file .exe pada link berikut ini :

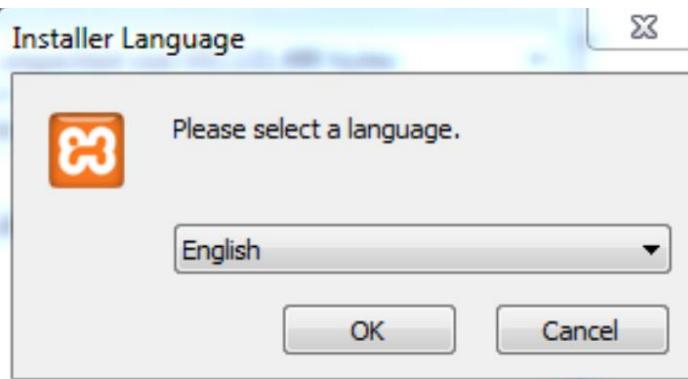
<https://www.apachefriends.org/download.html>

Sesuai kan dengan system operasi yang anda gunakan .



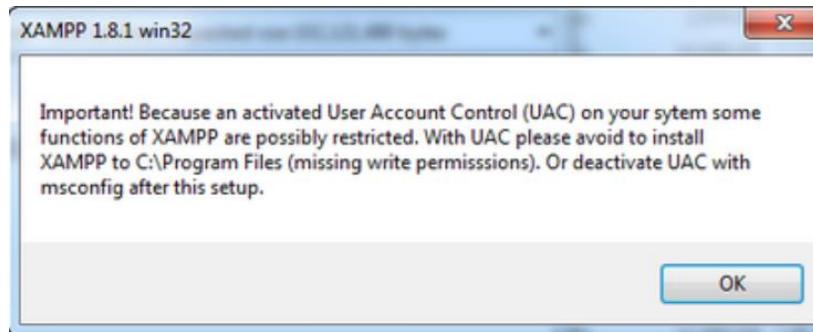
Gambar 4. 8 Instalasi Xampp

2. Dobel klik file XAMPP yang baru saja Anda download, nanti selanjutnya akan muncul jendela “installer language” seperti di bawah ini lalu pilih English dan tekan ok:



Gambar 4. 9 Instalasi Xampp

3. Kadang pada proses ini muncul pesan error. Jika ada, abaikan saja dan lanjutkan dengan klik OK dan YES.



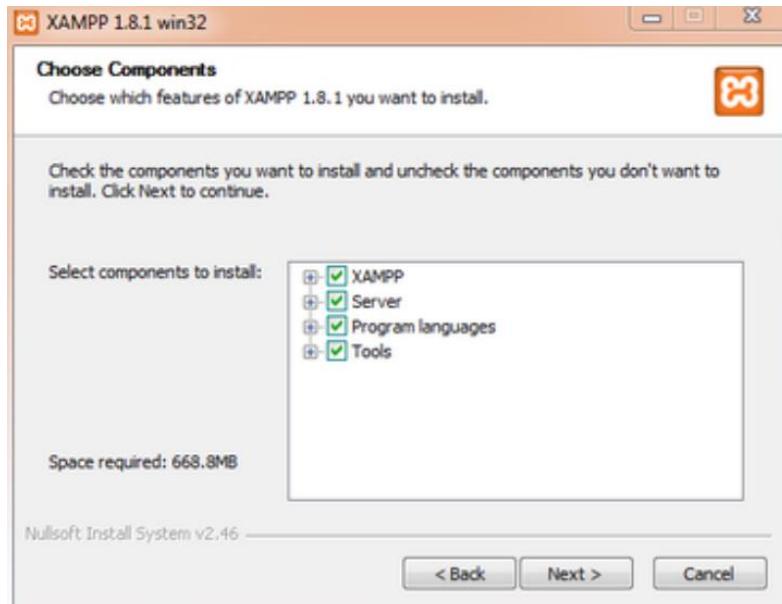
Gambar 4. 10 Instalasi Xampp

4. Berikutnya akan muncul jendela yang isinya meminta Anda menutup semua aplikasi yang sedang berjalan. Jika semua aplikasi sudah ditutup, maka klik tombol Next.



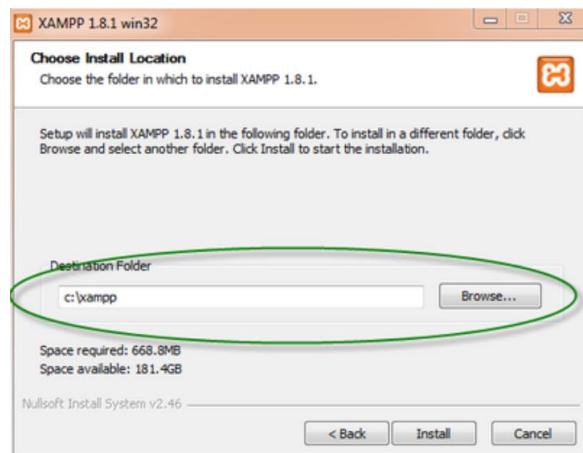
Gambar 4. 11 Instalasi Xampp

5. Selanjutnya Anda akan diminta untuk memilih aplikasi yang mau diinstal. Centang saja semua pilihan dan klik tombol Next.



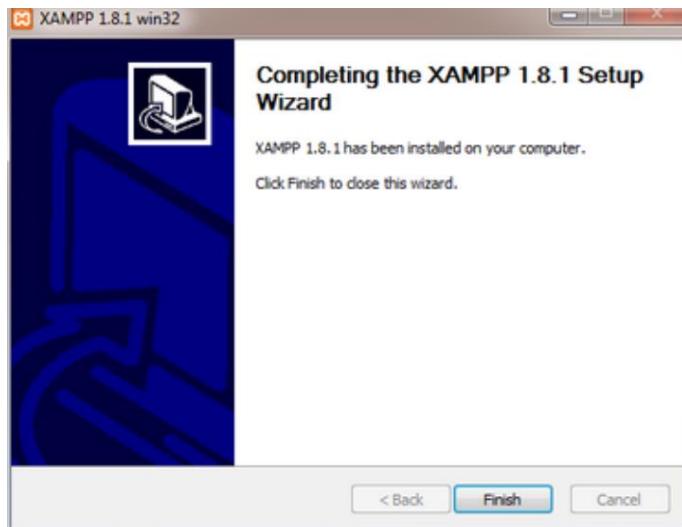
Gambar 4. 12 Instalasi Xampp

6. Kemudian Anda akan diminta untuk menentukan lokasi folder penyimpanan file-file dan folder XAMPP. Secara default akan diarahkan ke lokasi c:\xampp. Namun jika Anda ingin menyimpannya di folder lain bisa klik browse dan tentukan secara manual folder yang ingin digunakan. Jika sudah selesai, lanjutkan dan klik tombol Install.



Gambar 4. 13 Instalasi Xampp

7. Tunggu beberapa menit hingga proses instalasi selesai. Jika sudah muncul jendela seperti di bawah ini, klik tombol Finish untuk menyelesaiakannya.



Gambar 4. 14 Instalasi Xampp

8. Berikutnya, akan muncul jendela dialog seperti gambar di bawah ini yang menanyakan Anda apakah mau langsung menjalankan aplikasi XAMPP atau tidak. Jika ya, maka klik YES.

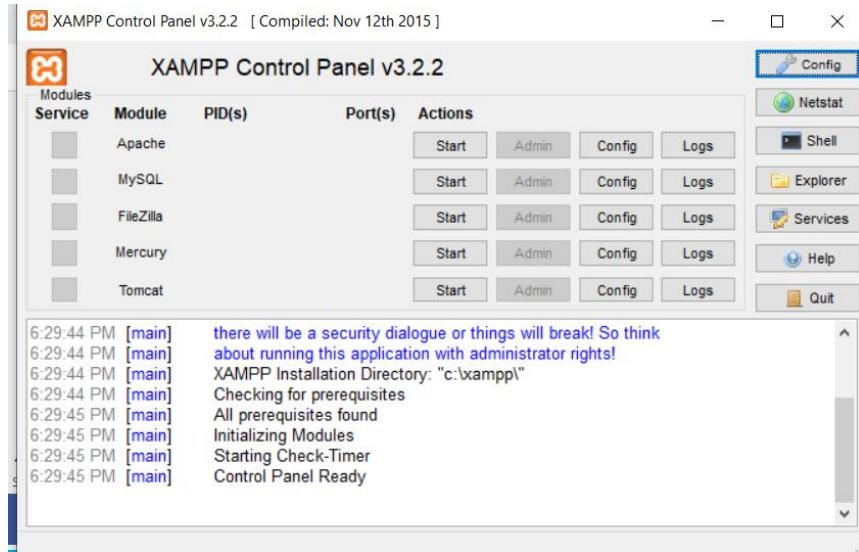


Gambar 4. 15 Instalasi Xampp

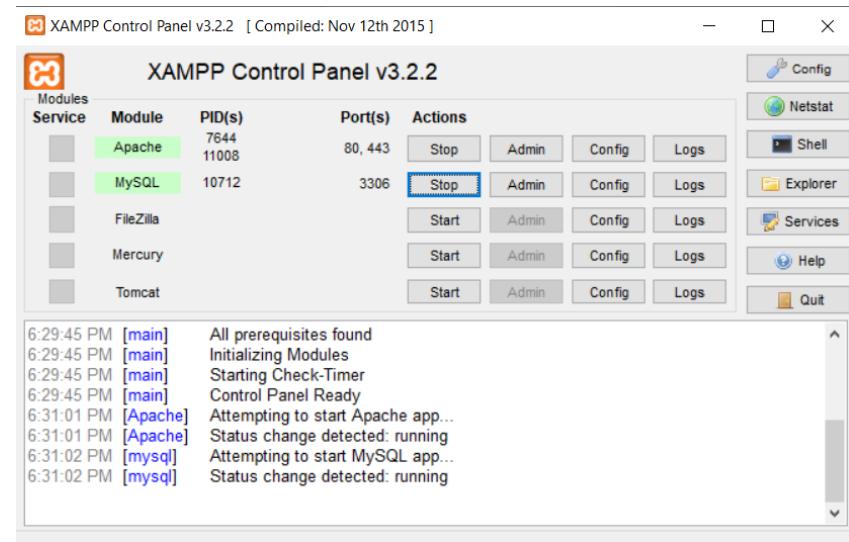
4.1.1.3 Cara Menjalankan Xampp

1. Bukalah aplikasi XAMPP, bisa melalui Start Menu atau Desktop, dan klik icon XAMPP. Atau, jika Anda membukanya begitu proses instalasi selesai maka klik Yes seperti yang terlihat pada gambar di atas.
2. Setelah terbuka, silahkan klik tombol Start pada kolom Action sehingga tombol tersebut berubah menjadi Stop. Dengan mengklik tombol tersebut,

artinya itulah aplikasi yang dijalankan. Biasanya jika saya menggunakan XAMPP, yang saya start hanyalah aplikasi Apache dan MySQL, karena saya tidak memerlukan aplikasi seperti Filezilla, dan lain-lain.



Gambar 4. 16 Menjalankan Xampp



Gambar 4. 17 Menjalankan Xampp

3. Sekarang bukalah browser kesukaan Anda, dan coba ketikkan <http://localhost/dashboard> di address bar. Jika muncul tampilan seperti gambar di bawah ini, instalasi telah berhasil.



Gambar 4. 18 Menjalankan Xampp

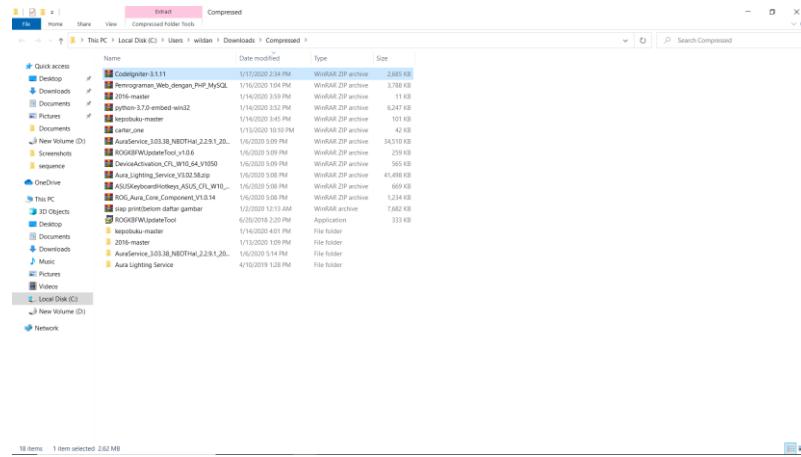
4.1.1.4 Instalasi CodeIgniter

- 1 Bukalah website <https://codeigniter.com/>.



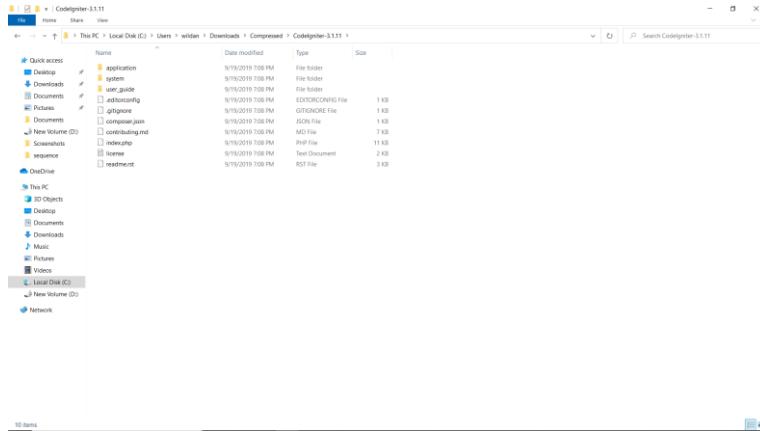
Gambar 4. 19 Instal Codeigniter

- 2 Setelah download, extract file codeigniter.



Gambar 4. 20 Extract file Codeigniter

3 Hasil dari extract file codeigniter



Gambar 4. 21 Extract file Codeigniter

4 Pindahkan file tersebut kedalam folder C:\xampp\htdocs

Name	Date modified	Type	Size
KunjunganRutan1Bandung	1/6/2020 5:23 PM	WinRAR archive	68,739 KB
vellariza	12/30/2019 12:44 AM	WinRAR ZIP archive	34,677 KB
surat kecil untuk midun(revisi)	12/27/2019 12:02 AM	Microsoft Word Document	218 KB
vellariza(1)	12/26/2019 10:15 PM	WinRAR archive	32,649 KB
KunjunganRutan1Bandung (10)	12/26/2019 7:51 AM	WinRAR ZIP archive	67,116 KB
KunjunganRutan1Bandung (H-1 SIDANG ...	12/25/2019 10:09 PM	WinRAR ZIP archive	66,691 KB
CONTROLLER SUDAH DI PISAH	12/24/2019 3:20 PM	WinRAR archive	63,899 KB
sapsidang	12/24/2019 12:39 AM	WinRAR archive	63,657 KB
KunjunganRutan1Bandung (minggu sida...	12/23/2019 12:05 AM	WinRAR ZIP archive	63,522 KB
vellariza	12/18/2019 2:14 PM	WinRAR archive	30,090 KB
KunjunganRutan1Bandung etek terbaru 1...	12/16/2019 12:06 AM	WinRAR ZIP archive	52,179 KB
paling baru db kunjungan.sql	12/16/2019 12:05 AM	SQL File	10 KB
applications	2/27/2017 4:36 PM	HTML File	4 KB
bitnami	2/27/2017 4:36 PM	Cascading Style Sheet	1 KB
favicon	7/16/2015 10:32 PM	Icon	31 KB
index.php	7/16/2015 10:32 PM	PHP File	1 KB
vellariza	12/30/2019 12:45 AM	File folder	
KunjunganRutan1Bandung	12/26/2019 10:37 PM	File folder	
img	12/26/2019 7:57 AM	File folder	
h-1	12/25/2019 10:09 PM	File folder	
UIJCOBA	12/23/2019 12:06 AM	File folder	
dashboard	12/15/2019 4:32 PM	File folder	
xampp	12/15/2019 4:32 PM	File folder	
webalizer	12/15/2019 4:32 PM	File folder	
Codeigniter-3.1.11	9/19/2019 7:08 PM	File folder	

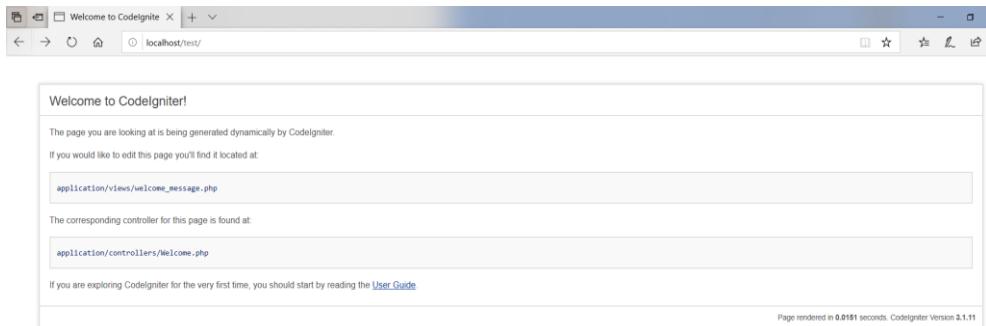
Gambar 4. 22 Extract file Codeigniter

- 5 Lalu rename dengan sesuai kebutuhan, disini saya merubah menjadi “test”

Name	Date modified	Type	Size
KunjunganRutan1Bandung	1/6/2020 5:23 PM	WinRAR archive	68,739 KB
vellariza	12/30/2019 12:44 AM	WinRAR ZIP archive	34,677 KB
surat kecil untuk midun(revisi)	12/27/2019 12:02 AM	Microsoft Word Document	218 KB
vellariza(1)	12/26/2019 10:15 PM	WinRAR archive	32,649 KB
KunjunganRutan1Bandung (10)	12/26/2019 7:51 AM	WinRAR ZIP archive	67,116 KB
KunjunganRutan1Bandung (H-1 SIDANG ...	12/25/2019 10:09 PM	WinRAR ZIP archive	66,691 KB
CONTROLLER SUDAH DI PISAH	12/24/2019 3:20 PM	WinRAR archive	63,899 KB
sapsidang	12/24/2019 12:39 AM	WinRAR archive	63,657 KB
KunjunganRutan1Bandung (minggu sida...	12/23/2019 12:05 AM	WinRAR ZIP archive	63,522 KB
vellariza	12/18/2019 2:14 PM	WinRAR archive	30,090 KB
KunjunganRutan1Bandung etek terbaru 1...	12/16/2019 12:06 AM	WinRAR ZIP archive	52,179 KB
paling baru db kunjungan.sql	12/16/2019 12:05 AM	SQL File	10 KB
applications	2/27/2017 4:36 PM	HTML File	4 KB
bitnami	2/27/2017 4:36 PM	Cascading Style Sheet	1 KB
favicon	7/16/2015 10:32 PM	Icon	31 KB
index.php	7/16/2015 10:32 PM	PHP File	1 KB
vellariza	12/30/2019 12:45 AM	File folder	
KunjunganRutan1Bandung	12/26/2019 10:37 PM	File folder	
img	12/26/2019 7:57 AM	File folder	
h-1	12/25/2019 10:09 PM	File folder	
UIJCOBA	12/23/2019 12:06 AM	File folder	
dashboard	12/15/2019 4:32 PM	File folder	
xampp	12/15/2019 4:32 PM	File folder	
webalizer	12/15/2019 4:32 PM	File folder	
test	9/19/2019 7:08 PM	File folder	

Gambar 4. 23 Extract file Codeigniter

- 6 Lalu buka browser dan masukan <http://localhost/test/>



Gambar 4. 23 Extract file Codeigniter

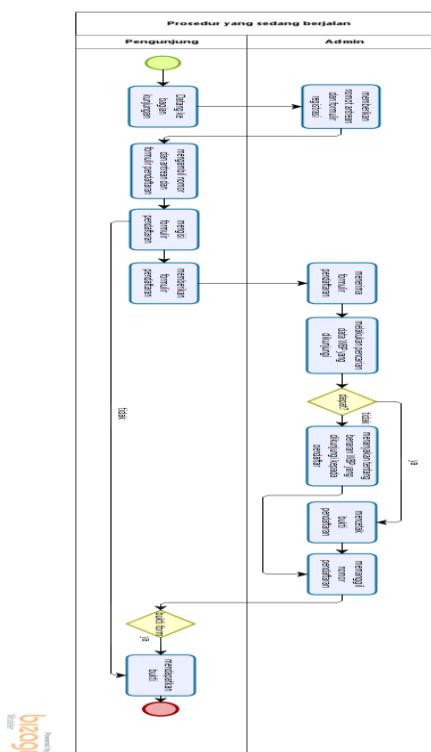
BAB V PERANCANGAN DAN PEMBUATAN APLIKASI

5.1 PERANCAGAN APLIKASI

Dalam membangun suatu program sangat diperlukan suatu perancangan aplikasi, supaya aplikasi yang di buat sesuai dengan yang di inginkan, maka dari itu diperlukan suatu perancangan system seperti berikut :

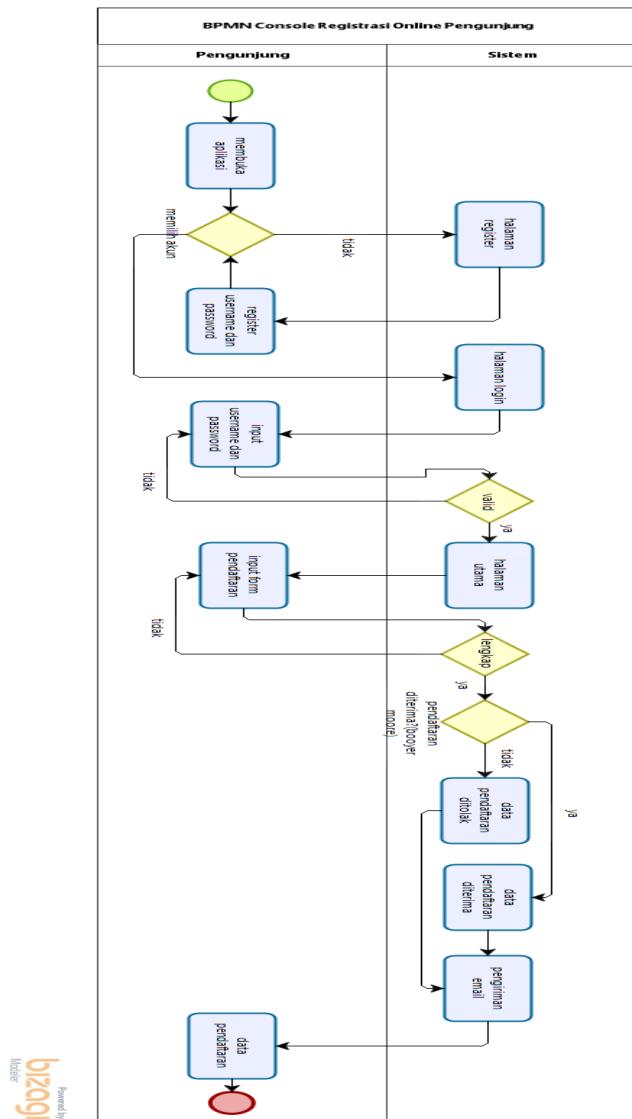
5.1.1 Analisis Yang Sedang Berjalan

Tahapan yang diperlukan dalam pembuatan suatu program yaitu menganalisa sistem yang telah ada, dimana analisa sistem akan memberikan gambaran tentang sistem yang saat ini sedang berjalan dan bertujuan mengetahui lebih jelas bagaimana cara kerja atau rancangan sistem tersebut serta untuk mendefinisikan dan mengevaluasi permasalahan terjadi dan kebutuhan-kebutuhan yang diharapkan dapat diusulkan suatu perbaikan



Gambar 5.1 *Flowchart* Prosedur Pendaftaran Yang Sedang Berjalan

5.1.2 Analisis Registrasi Yang Akan Dibangun

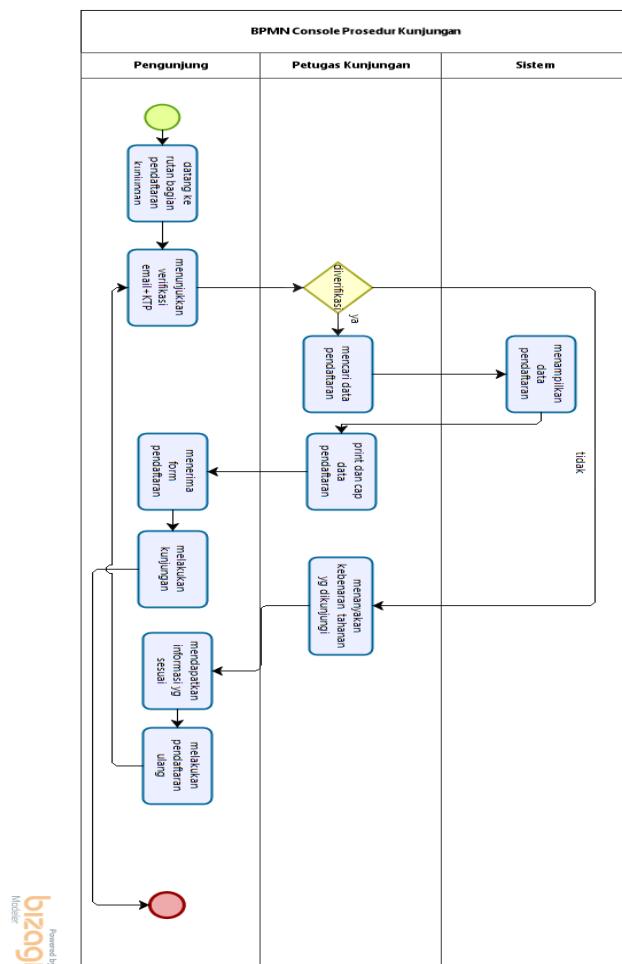


Gambar 5.9 Flowchart Proses Pendaftaran Online

Dapat dilihat gambar 5.9 pendaftar masuk ke dalam aplikasi dengan *login* jika memiliki akun, dan jika tidak memiliki akun akan melakukan *register* terlebih dahulu. Setelah itu di halaman utama atau form pendaftaran, pengunjung melakukan *input* data form pendaftaran tersebut dengan lengkap,

lalu pendaftar selesai. lalu akan dilakukan verifikasi oleh sistem menggunakan metode boyer moore apakah pendaftaran tersebut diverifikasi atau ditolak, lalu sistem akan mengirimkan pendaftaran melalui via email pendaftar kunjungan, kemudian pendaftar akan mendapatkan bukti pendaftaran di terima atau ditolak.

5.1.3 Flow Chart Prosedure Kunjungan Yang Akan Dibangun



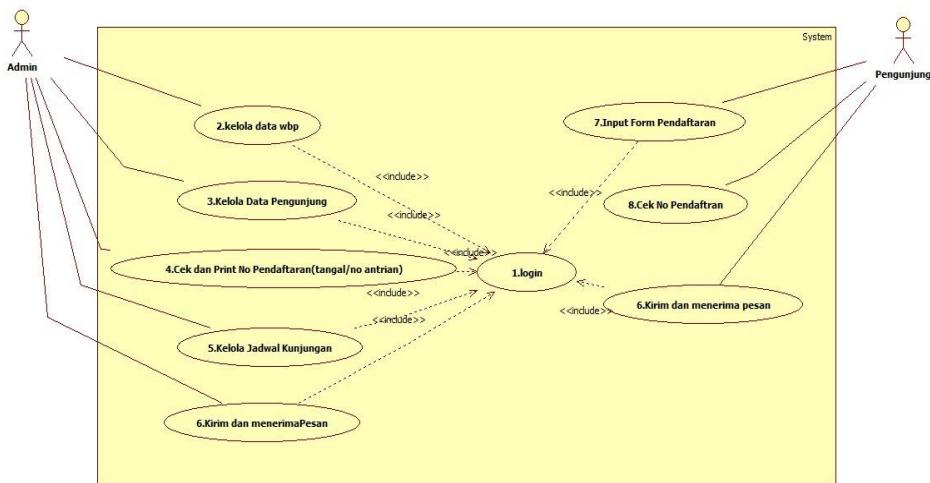
Gambar 5.10 Flowchart Proses Cek Registrasi Yang Akan Dibangun

Dapat dilihat dari alur diatas Gambar 5.10 adalah proses procedure kunjungan jika menggunakan registrasi online yang akan dibangun, prosesnya

yaitu pengunjung mendatangi rutan klas 1 bandung bagian kunjungan lalu pengunjung memperlihatkan hasil verifikasi via email kepada pedugas kunjungan rutan klas 1 bandung, jika status pada email diverifikasi maka pengunjung akan mendapatkan form pendaftaran yang sudah dilakukan melalui registrasi via online dan dapat masuk kerutan klas 1 Bandung untuk melakukan kunjungan, jika status email tidak diverifikasi admin akan menanyakan tentang kebenaran WBP yang akan dikunjungi lalu jika sudah layak mendapatkan kebenaran informasi maka pengunjung dapat melakukan registrasi ulang.

5.1.4 Use Case Diagram

Diagram *use case* menggambarkan interaksi antar *use case* dan *actor* dalam suatu sistem.

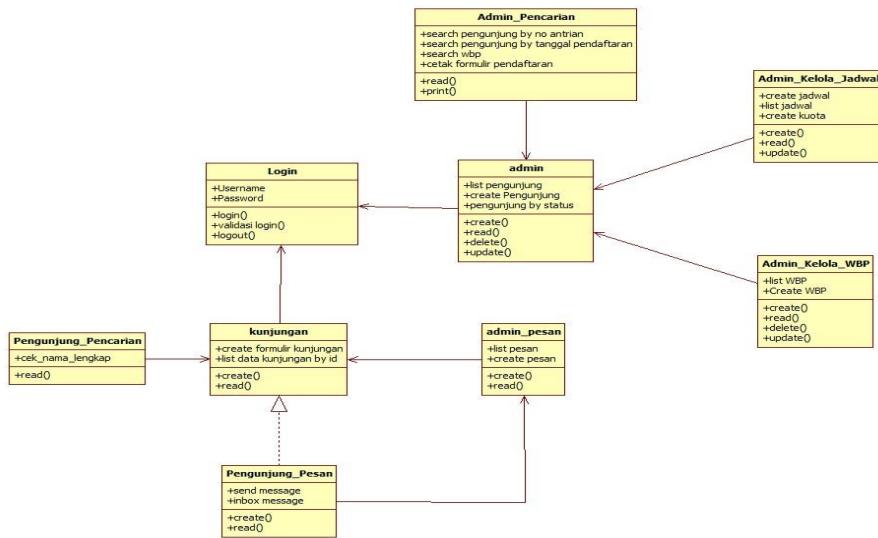


Gambar 5.11 Use Case Diagram Pendaftaran kunjungan

5.1.5 Class Diagram

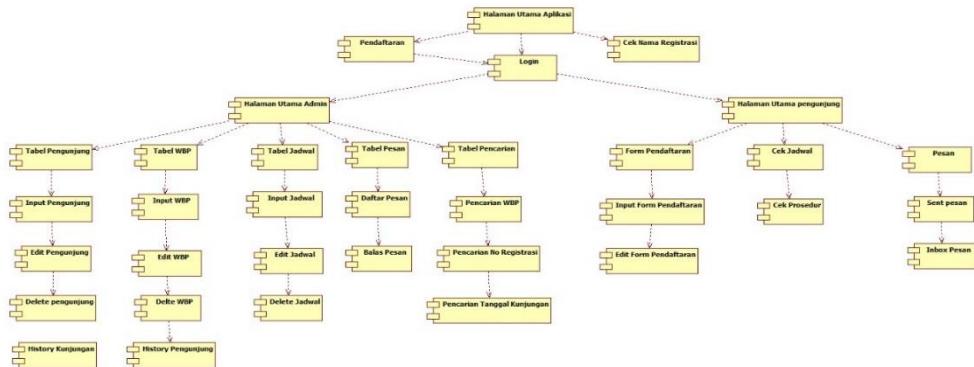
Class diagram berfungsi untuk menggambarkan suatu struktur dan hubungan objek-objek yang ada pada sistem. Struktur itu meliputi atribut-atribut dan method-method yang ada pada masing-masing kelas. Adapun

Class Diagram pada aplikasi pendaftaran kunjungan Online ini yaitu sebagai berikut :



Gambar 5.12 Class Diagram Aplikasi Pendaftaran Kunjungan Online

5.1.6 Component Diagram

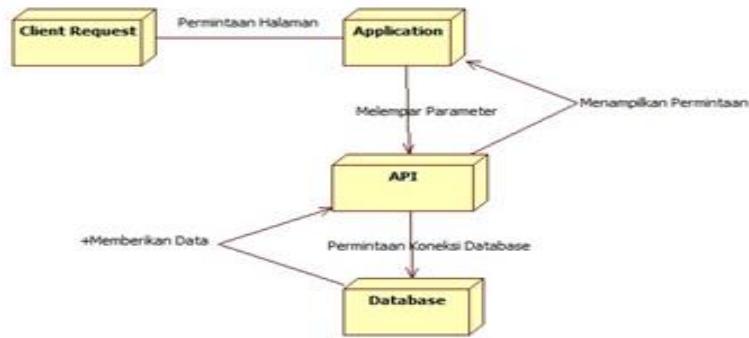


Gambar 5.43 Component Diagram Pendaftaran Kunjungan Online

5.1.7 Component Diagram

Deployment diagram menunjukkan tata letak sebuah sistem secara fisik, menampilkan bagian-bagian *software* yang berjalan pada bagian-bagian

hardware yang digunakan untuk mengimplementasikan sebuah sistem dan keterhubungan antara komponen-komponen *hardware* tersebut. *Deployment diagram* dapat digunakan pada bagian-bagian awal proses perancangan sistem untuk mendokumentasikan arsitektur fisik sebuah sistem. Berikut *deployment diagram* yang menggambarkan susunan fisik dan perangkat lunak dan sistem



Gambar 5.44 Deployment Diagram Aplikasi

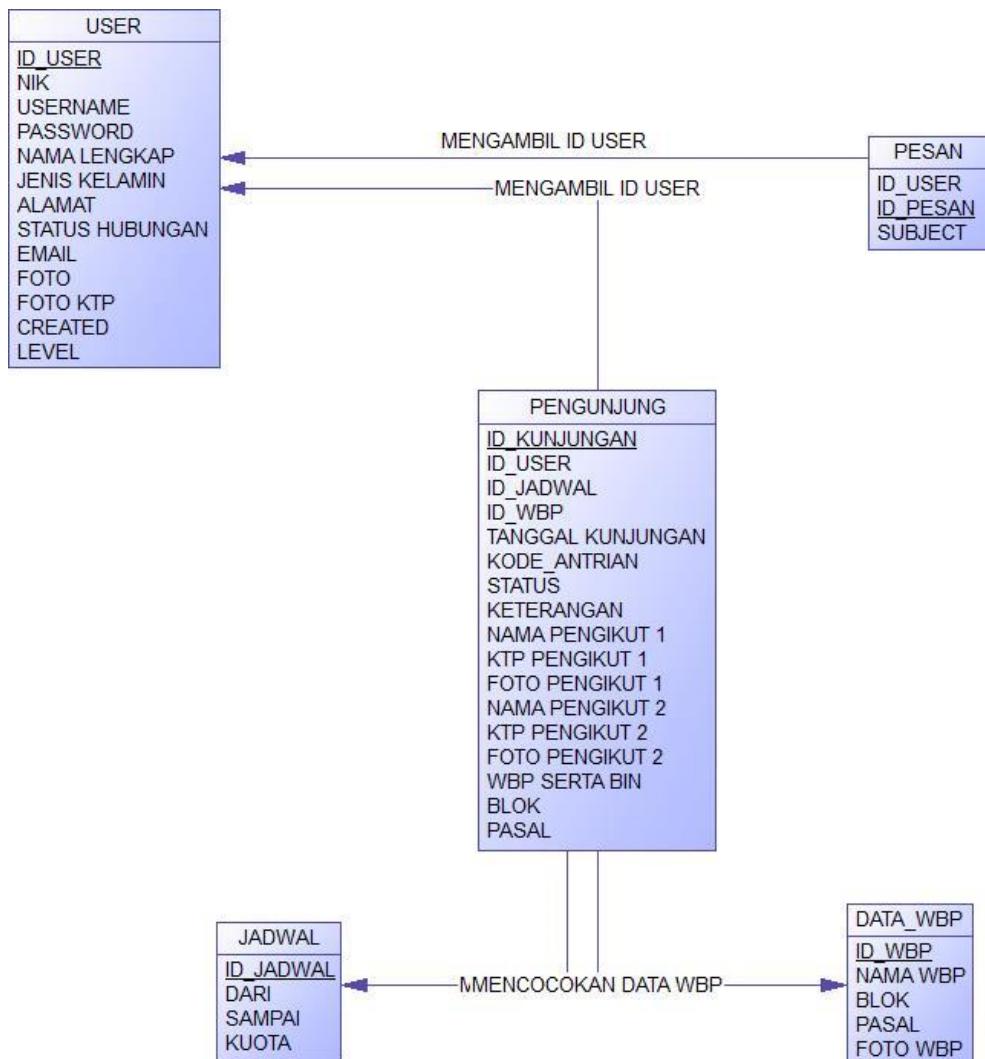
5.2 PERANCAGAN DATABASE

5.2.1 Conceptual Data Model (CDM)



Gambar 5.46 CDM

5.2.2 Physical Data Model (PDM)



Gambar 5.46 PDM

5.3 MEMPERSIAPKAN TOOLS YANG DIBUTUHKAN

Sebelum melakukan tutorial penulis akan menjelaskan sedikit pemaparan latar belakang mengapa Aplikasi Pendaftaran Online Lapas menggunakan algoritma *boyer moore* ini dibuat dan hasil yang diperoleh dari aplikasi yang sudah berhasil dibuat dan diuji. Salah satu permasalahan pada lapas yaitu dengan adanya pendaftaran via online keamanan lapas tentunya

tidak bisa dilupakan, oleh karena itu dibuatlah aplikasi yang menggunakan metode boyer moore untuk meningkatkan kemanan dan melakukan penyaringan kembali terhadap pendaftar yang ingin melakukan kunjungan.

Boyer moore menjadi salah satu metode yang sedang banyak diterapkan dalam String Matching, dengan beberapa contoh seperti pencocokan Fingger Print.

Berdasarkan uraian diatas, maka peneliti membuat sebuah sistem Aplikasi Pendaftaran Kunjungan Online Menggunakan Algoritma Boyer Moore. Bahasa pemograman yang digunakan adalah Framework Codeigniter dan Database MYSQL. Hasil penelitian ini Validasi berdasarkan String Matching yang dibuat dapat Melakukan verifikasi kunjungan secara otomatis berdasarkan rentan kesamaan string, video dan secara real time di webcam. Hasil penelitian menunjukkan bahwa dengan adanya aplikasi tersebut dapat mempermudah dan mempercepat pendaftaran via online dan tidak melupakan tingkat kemanan rutan.

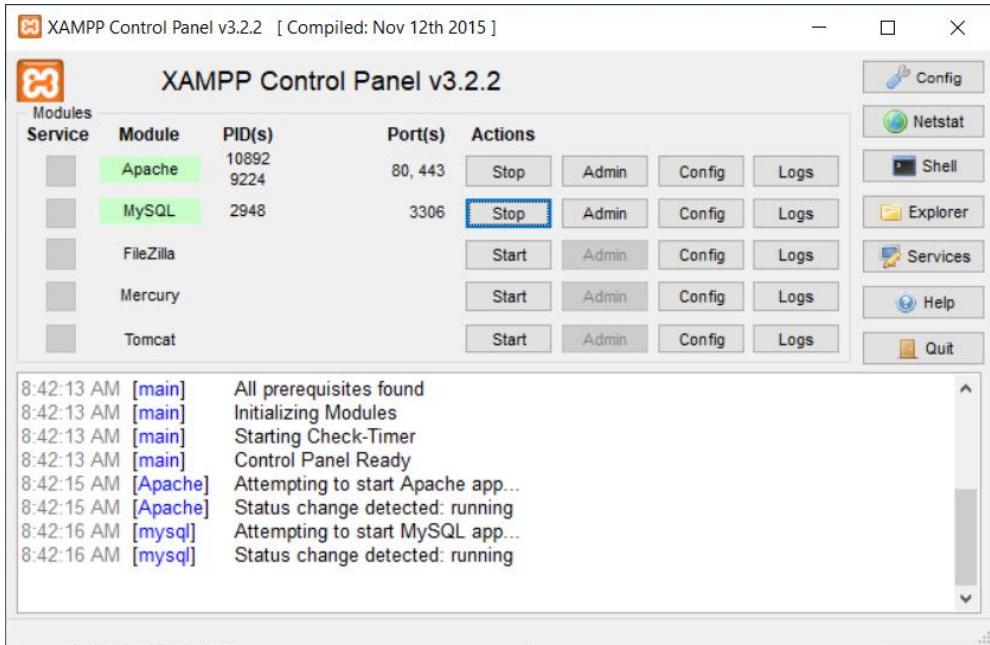
5.3.1 Mempersiapkan Tools

Untuk membuat Aplikasi pendaftaran Online kunjungan rutan maka diperlukan beberapa tools yang dapat mendukung dalam pembuatannya dan tools ini sudah dibahas pada bab sebelumnya, berikut tools yang harus dipersiapkan dalam pembuatan system object deteksi :

- XAMPP
- VISUAL STUDIO CODE
- CODEIGNITER

5.3.2 Membuat database

Langkah awal ialah membuka aplikasi Xampp dan menjalankan APACHE dan MySQL.



Gambar?

Setelah itu buka web browser lalu ketik <http://localhost/phpmyadmin/>

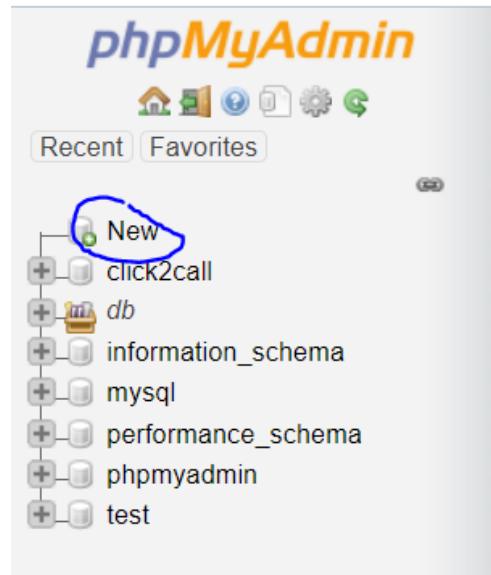
The screenshot shows the phpMyAdmin interface. At the top, it displays 'Server: 127.0.0.1' and the URL 'localhost/phpmyadmin'. The left sidebar shows a tree view of databases: New, click2call, information_schema, mysql, performance_schema, phpmyadmin, and test.

The main area is divided into several sections:

- General settings:** Server connection collation: utf8mb4_unicode_ci.
- Appearance settings:** Language: English, Theme: pmahomme, Font size: 82%.
- Database server:**
 - Server: 127.0.0.1 via TCP/IP
 - Server type: MariaDB
 - Server connection: SSL is not being used
 - Server version: 10.1.38-MariaDB - mariadb.org binary distribution
 - Protocol version: 10
 - User: root@localhost
 - Server charset: UTF-8 Unicode (utf8)
- Web server:**
 - Apache/2.4.38 (Win64) OpenSSL/1.0.2q PHP/5.6.40
 - Databases client version: libmysql - mysqlnd 5.0.11-dev - 20120503 - \$Id: 766082459e12d4553bd11c93cc05be2f97a \$
 - PHP extension: mysqli curl mbstring
 - PHP version: 5.6.40
- phpMyAdmin:**
 - Version information: 4.8.5, latest stable version: 4.9.4
 - Documentation
 - Official Homepage
 - Contribute
 - Get support
 - List of changes
 - License

gambar

Lalu klik new untuk membuat database baru



gambar

Untuk kasus pada kali ini saya memberikan nama database dengan db_kunjungan

A screenshot of the phpMyAdmin 'Databases' page. At the top, there is a "Create database" button with a "Create" icon. Below it is a search bar containing "db_kunjungan" and a dropdown menu set to "latin1_swedish_ci". To the right of these are "Create" and "Check privileges" buttons. The main area is a table with three columns: "Database", "Collation", and "Action". The table lists eight databases: "click2call", "db_kunjungan" (which is highlighted with a gray background), "db_pegadaian", "information_schema", "mysql", "performance_schema", "phpmyadmin", and "test". Each row has a checkbox in the first column and a "Check privileges" link in the "Action" column. At the bottom of the table, it says "Total: 8" and "latin1_swedish_ci". Below the table are buttons for "Check all" and "Drop".

gambar

Pada db_kunjungan terdapat 5 tabel yaitu :

- Datawbp
- Jadwal
- Pengunjung
- Pesan
- user

Table	Action	Rows	Type	Collation	Size	Overhead
datawbp		2	InnoDB	latin1_swedish_ci	16 Kib	-
jadwal		6	InnoDB	latin1_swedish_ci	16 Kib	-
pengunjung		2	MyISAM	latin1_swedish_ci	7.2 Kib	4.8 Kib
pesan		0	InnoDB	latin1_swedish_ci	16 Kib	-
user		5	MyISAM	latin1_swedish_ci	3.7 Kib	148 B
5 tables	Sum	15	InnoDB	latin1_swedish_ci	58.8 Kib	5 Kib

Gambar?

Struktur data table “datawbp“

Table: datawbp										
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	
1	id_wbp	int(11)	latin1_swedish_ci		No	None		AUTO_INCREMENT	Drop	
2	nama_dan_bin	varchar(50)	latin1_swedish_ci		No	None			Drop	
3	pasal	varchar(100)	latin1_swedish_ci		No	None			Drop	
4	blok_kamar	varchar(100)	latin1_swedish_ci		No	None			Drop	
5	foto	varchar(50)	latin1_swedish_ci		No	None			Drop	

Gambar?

Struktur data table “jadwal“

Table: jadwal										
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action	
1	id_jadwal	int(4)	latin1_swedish_ci		No	None		AUTO_INCREMENT	Drop	
2	dari	time	latin1_swedish_ci		No	None			Drop	
3	sampai	time	latin1_swedish_ci		No	None			Drop	
4	kuota	int(4)	latin1_swedish_ci		No	None			Drop	

Gambar?

Struktur data table “Pengunjung”

The screenshot shows the 'Structure' tab for the 'pengunjung' table in MySQL Workbench. The table has 22 columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_pengunjung	int(4)			No	None		AUTO_INCREMENT	Change Drop ▾ More
2	id_peserta	int(4)			No	None			Change Drop ▾ More
3	id_jadwal	int(4)			No	None			Change Drop ▾ More
4	tanggal_kunjungan	date			No	None			Change Drop ▾ More
5	kode_antrian	varchar(255)	latin1_swedish_ci		No	None			Change Drop ▾ More
6	status	varchar(255)	latin1_swedish_ci		No	belum divertifikasi			Change Drop ▾ More
7	Keterangan	varchar(255)	latin1_swedish_ci		No	None			Change Drop ▾ More
8	Iaki_laki	varchar(50)	latin1_swedish_ci		No	None			Change Drop ▾ More
9	perempuan	varchar(50)	latin1_swedish_ci		No	None			Change Drop ▾ More
10	anak_anak	int(100)			No	None			Change Drop ▾ More
11	fotopengikut1	varchar(100)	latin1_swedish_ci		No	None			Change Drop ▾ More
12	Ktpengikut1	varchar(255)	latin1_swedish_ci		No	None			Change Drop ▾ More
13	Jumlah	varchar(30)	latin1_swedish_ci		No	None			Change Drop ▾ More
14	JenisKelamin2	varchar(30)	latin1_swedish_ci		No	None			Change Drop ▾ More
15	noktp2	int(30)			No	None			Change Drop ▾ More
16	fotopengikut2	varchar(100)	latin1_swedish_ci		No	None			Change Drop ▾ More
17	Ktpengikut2	varchar(255)	latin1_swedish_ci		No	None			Change Drop ▾ More
18	wbbsertabin	varchar(100)	latin1_swedish_ci		No	None			Change Drop ▾ More
19	pasal	varchar(30)	latin1_swedish_ci		No	None			Change Drop ▾ More
20	blok	varchar(5)	latin1_swedish_ci		No	None			Change Drop ▾ More
21	fotowbppengunjung	text	latin1_swedish_ci		No	None			Change Drop ▾ More
22	created	timestamp			Yes	CURRENT_TIMESTAMP			Change Drop ▾ More

Gambar?

Struktur data table “pesan“

The screenshot shows the 'Structure' tab for the 'pesan' table in MySQL Workbench. The table has 9 columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_pesan	int(11)			No	None		AUTO_INCREMENT	Change Drop ▾ More
2	id_peserta	int(11)			No	None			Change Drop ▾ More
3	nama_lengkap	varchar(255)	latin1_swedish_ci		No	None			Change Drop ▾ More
4	email	varchar(255)	latin1_swedish_ci		Yes	NULL			Change Drop ▾ More
5	tujuan	int(11)			No	None			Change Drop ▾ More
6	subject	varchar(255)	latin1_swedish_ci		No	None			Change Drop ▾ More
7	no_hp	varchar(20)	latin1_swedish_ci		Yes	NULL			Change Drop ▾ More
8	isi	text	latin1_swedish_ci		No	None			Change Drop ▾ More
9	created	timestamp			No	CURRENT_TIMESTAMP			Change Drop ▾ More

Gambar?

Struktur data table “user“

The screenshot shows the 'Structure' tab for the 'user' table in MySQL Workbench. The table has 15 columns:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_peserta	int(4)		UNSIGNED ZEROFILL	No	None		AUTO_INCREMENT	Change Drop ▾ More
2	nik	int(20)			No	None			Change Drop ▾ More
3	username	varchar(10)	latin1_swedish_ci		No				Change Drop ▾ More
4	password	varchar(10)	latin1_swedish_ci		No	peserta			Change Drop ▾ More
5	nama_lengkap	varchar(30)	latin1_swedish_ci		No	None			Change Drop ▾ More
6	nama_panggilan	varchar(20)	latin1_swedish_ci		No	None			Change Drop ▾ More
7	level	varchar(10)	latin1_swedish_ci		No	peserta			Change Drop ▾ More
8	jenis_kelamin	varchar(12)	latin1_swedish_ci		No	None			Change Drop ▾ More
9	alamat	varchar(100)	latin1_swedish_ci		No	None			Change Drop ▾ More
10	status_hubungan	varchar(50)	latin1_swedish_ci		No	None			Change Drop ▾ More
11	email	varchar(50)	latin1_swedish_ci		No	None			Change Drop ▾ More
12	dokumen	text	latin1_swedish_ci		No	None			Change Drop ▾ More
13	kk	text	latin1_swedish_ci		No	None			Change Drop ▾ More
14	created	timestamp			No	CURRENT_TIMESTAMP			Change Drop ▾ More
15	updated	timestamp		on update CURRENT_TIMESTAMP	No	CURRENT_TIMESTAMP		ON UPDATE CURRENT_TIMESTAMP	Change Drop ▾ More

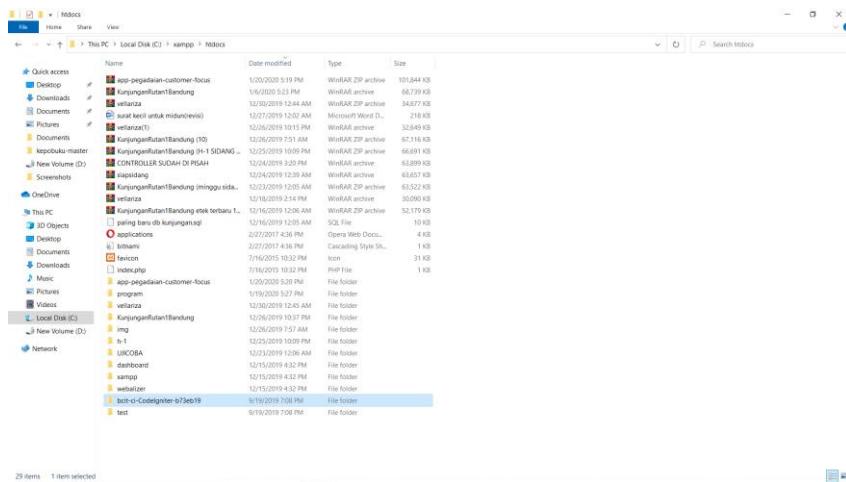
gambar

Pada gambar gambar diatas merupakan struktur database yang terdapat pada aplikasi pendaftaran Kunjungan Online rutan dan akan dilanjutkan kedalam tahap selanjutnya yaitu configurasi Codeigniter sesuai dengan kebutuhan program yang akan dibuat.

5.3.3 Konfigurasi Codeigniter

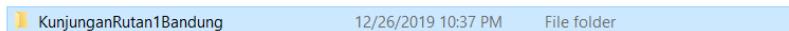
Pada setiap pembuatan aplikasi harus melakukan konfigurasi terlebih dahulu supaya program yang kita buat dapat terhubung dengan database dari aplikasi tersebut, berikut merupakan konfigurasi Codeigniter:

1. Copykan folder Codeigniter yang sudah di download di <https://codeigniter.com/en/download> kedalam folder C:\xampp\htdocs



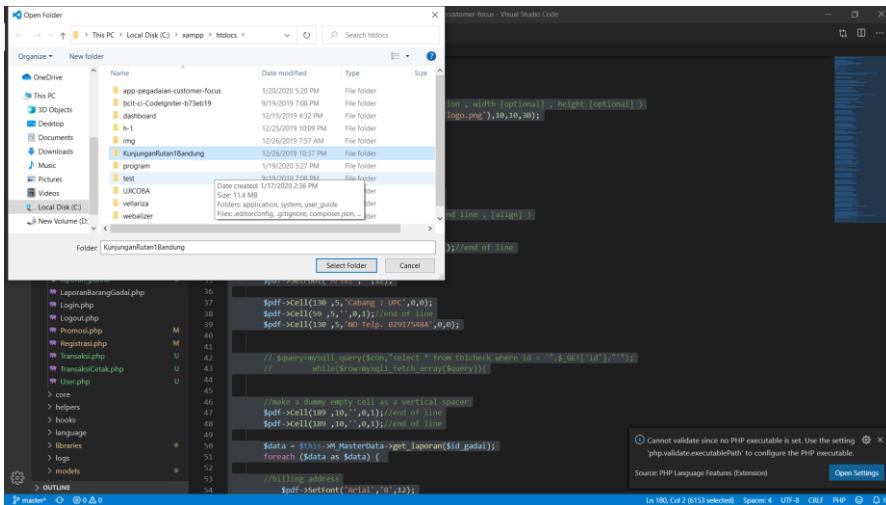
gambar

2. Rename folder codeigniter yang sudah di download menjadi sesuai dengan nama aplikasi yang akan dibangun.



gambar

3. Buka tools visual Studio Code, open folder codeigniter yang sudah di rename tadi



gambar

- Lalu buka config>config.php, pada bagian :

`$config['base_url'] =`

isikan menjadi sesuai nama folder codeigniter

`$config['base_url'] = 'http://localhost/kunjunganrutan1bandung/';`

Yang berfungsi sebagai base url yang akan di panggil pada web browser

```

        $config['base_url'] = 'http://localhost/kunjunganrutan1bandung/';
    
```

gambar

- Dan hasilnya jika kita memanggil localhost/kunjunganrutan1bandung/ pada aplikasi web browser akan muncul tampilan program tersebut



gambar

6. Buka config>config.php, pada bagian:

`autoload['libraries']`

isikan menjadi :

`$autoload['libraries'] = array('database', 'session', 'pagination');`

Konfigurasi pada file ini bertujuan untuk mementukan sumber daya apa yang akan diload secara otomatis.

```

File Edit Selection View Go Debug Terminal Help
autoload.php - Rutan Klas 1 Bandung - Visual Studio Code
application/config/autoload.php
37 | Prototype:
38 | $autoload['packages'] = array(APPPATH.'third_party', '/usr/local/shared');
39 |
40 |
41 |
42 | $autoload['packages'] = array();
43 |
44 | -----
45 | Auto-load Libraries
46 |
47 | These are the classes located in system/libraries/ or your
48 | application/libraries/ directory, with the addition of the
49 | 'database' library, which is somewhat of a special case.
50 |
51 | Prototype:
52 |
53 | $autoload['libraries'] = array('database', 'email', 'session');
54 |
55 | You can also supply an alternative library name to be assigned
56 | in the controller:
57 |
58 | $autoload['libraries'] = array('user_agent' => 'ua');
59 |
60 | $autoload['libraries'] = array('database', 'session', 'pagination');
61 |
62 | -----
63 | Auto-load Drivers
64 |
65 | These classes are located in system/libraries/ or in your
66 | application/libraries/ directory, but are also placed inside their
67 | own subdirectory and they extend the CI_Driver_Library class. They
68 | offer multiple interchangeable driver options.
69 |
70 | Prototype:
71 |
72 | $autoload['drivers'] = array('cache');
73 |
74 | You can also supply an alternative memory name to be assigned in
75

```

Gambar

7. Buka config>config.php, pada bagian:

`$autoload['helper'] =`

isikan menjadi :

```
$autoload['helper'] =  
array('url','language','date','cookie','form','file','text');
```

Konfigurasi pada file ini bertujuan untuk mementukan sumber daya apa yang akan diload secara otomatis.

```
File Edit Selection View Go Debug Terminal Help  
autoload.php - Keep my cursor blurring Visual Studio Code  
File Edit Selection View Go Debug Terminal Help  
application> config > autoload.php  
application> config > autoload.php  
| these classes are located in system/libraries or in your  
| own subdirectory but we also placed inside their  
| own subdirectory and they extend the CI_Driver_Library class. They  
| offer multiple interchangeable driver options.  
|  
| Prototype:  
|  
| $autoload['drivers'] = array('cache');  
| You can also supply an alternative property name to be assigned in  
| the controller:  
|  
| $autoload['drivers'] = array('cache' => 'cch');  
|  
| Prototype:  
|  
| $autoload['drivers'] = array();  
|  
| Auto-load Helper files  
|-----  
| Prototype:  
|  
| $autoload['helper'] = array('url', 'file');  
|  
| Prototype:  
|  
| $autoload['helper'] = array('url', 'language', 'date', 'cookie', 'form', 'file', 'text');  
|  
| Auto-load Config files  
|-----  
| Prototype:  
|  
| $autoload['config'] = array('config1', 'config2');  
|  
| NOTE: This item is intended for use ONLY if you have created custom  
| config files. Otherwise, leave it blank.  
|  
| config1, config2  
|  
101 |  
102 |  
103 |  
104 |  
105 |
```

gambar

- Setelah itu pergi ke config>database, pada bagian :

```
$db['default'] = array(
```

```
'dsn' => "",  
'hostname' => 'localhost',  
'username' => "",  
'password' => "",  
'database' => "",  
'dbdriver' => 'mysqli',
```

Rubah menjadi :

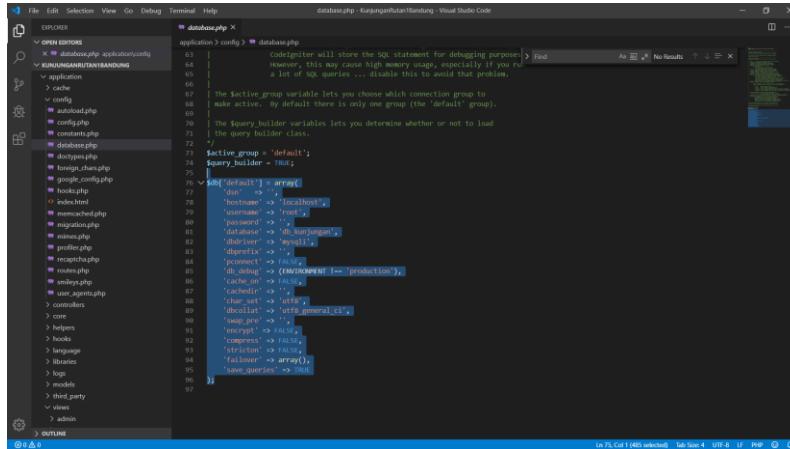
```
$db['default'] = array(
```

```
'dsn' => "",  
'hostname' => 'localhost',  
'username' => 'root',  
'password' => "",
```

```
'database' => 'db_kunjungan',
```

```
'dbdriver' => 'mysql',
```

Yang berfungsi untuk menghubungkan koneksi database yang sudah dibuat kedalam aplikasi web yang akan dibuat



```
File Edit Selection View Go Debug Terminal Help
databases.php X database.php
databases.php
application / config > database.php
43 | Code editor will store the SQL statement for debugging purposes, however, this may cause high memory usage, especially if you run
44 | a lot of SQL queries ... disable this to avoid that problem.
45 |
46 | The $active_group variable lets you choose which connection group to
47 | make active. By default there is only one group (the default group).
48 |
49 | The $query_builder variables lets you determine whether or not to load
50 | the query builder class.
51 |
52 | $active_group = 'default';
53 | $query_builder = TRUE;
54 |
55 | $db['default'] = array(
56 |     'dsn' => '',
57 |     'hostname' => 'localhost',
58 |     'username' => 'root',
59 |     'password' => '',
60 |     'database' => 'db_kunjungan',
61 |     'dbdriver' => 'mysql',
62 |     'dbprefix' => '',
63 |     'pconnect' => FALSE,
64 |     'db_debug' => ($ENVIRONMENT != 'production'),
65 |     'cache_on' => FALSE,
66 |     'cache_limit' => 20,
67 |     'char_set' => 'utf8',
68 |     'dbcollat' => 'utf8_general_ci',
69 |     'swap_pre' => '',
70 |     'encrypt' => FALSE,
71 |     'compress' => FALSE,
72 |     'stricton' => FALSE,
73 |     'failover' => array(),
74 |     'save_queries' => TRUE
75 |
76 );
77 |
78 
```

gambar

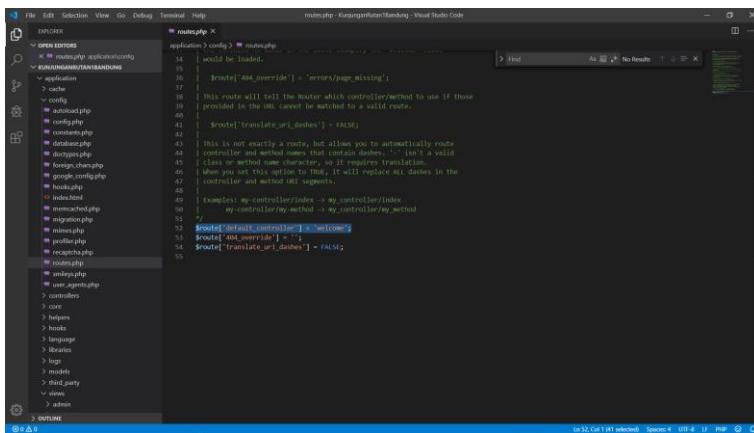
9. Buka config>routes.php, pada bagian:

```
$route['default_controller'] = 'welcome';
```

isikan menjadi :

```
$route['default_controller'] = 'welcome';
```

Konfigurasi pada file ini bertujuan untuk mementukan default controller yang nantinya akan dipanggil.



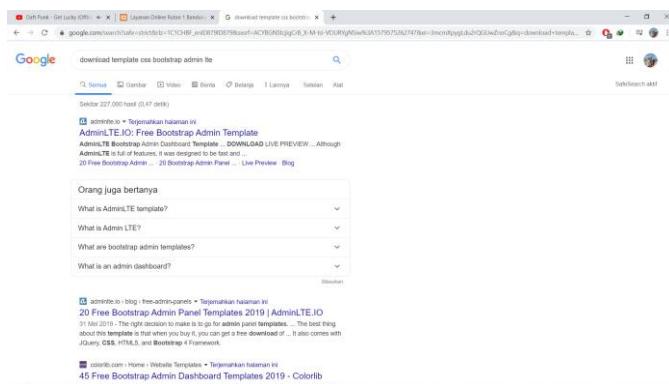
```
File Edit Selection View Go Debug Terminal Help
routes.php X routes.php
routes.php
application / config > routes.php
34 | would be loaded.
35 |
36 | $route['404_override'] = 'errors/page_missing';
37 |
38 | This route will tell the router which controller/method to use if those
39 | provided in the URL cannot be matched to a valid route.
40 |
41 | $route['translate_uri_dashes'] = TRUE;
42 |
43 | This is not exactly a route, but allows you to automatically route
44 | controller and method names that contain dashes to a valid
45 | class name and no underscores, so it requires translation.
46 | When you set this option to TRUE, it will replace all dashes in the
47 | controller and method URL segments.
48 |
49 | Examples: my-controller/index -> my_controller/index
50 | my-controller/my-method -> my_controller/my_method
51 |
52 | $route['default_controller'] = 'welcome';
53 | $route['404_override'] = '';
54 | $route['translate_uri_dashes'] = TRUE;
55 
```

Gambar?

5.4 KONFIGURASI CSS BOOTSRAP

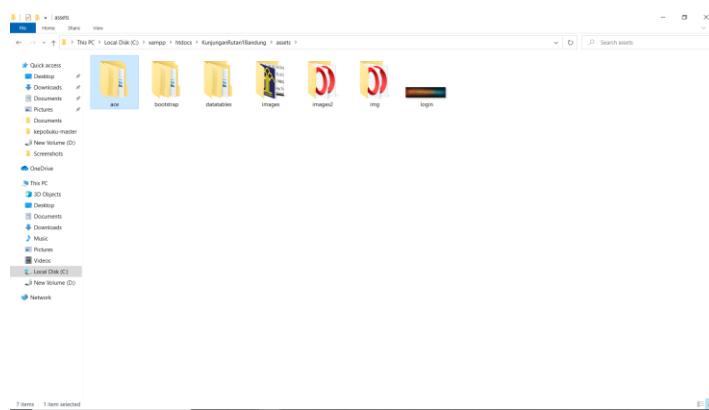
Pada setiap pembuatan aplikasi Tentunya harus memiliki tampilan yang menarik. Tujuan dari dibuatnya template berfungsi untuk jika melakukan pemanggilan view menjadi lebih mudah, maka dari itu diwajibkan untuk melakukan setting template pada codeigniter untuk mempermudah pemanggilan VIEW yaitu seperti:

1. Pilih template yang sesuai dengan keinginan atau sesuai dengan kebutuhan aplikasi yang dapat di unduh melalui web browser dengan mengetikan katakunci download template css bootsrap



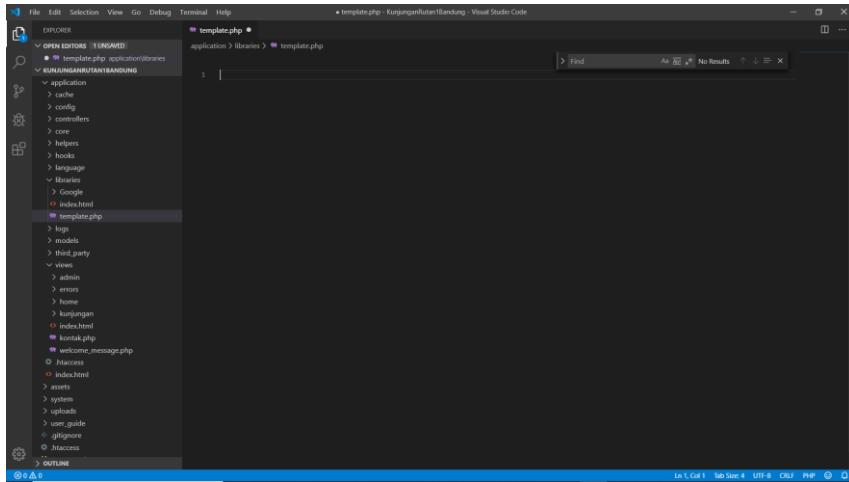
gambar

2. Jika sudah di download extract bootsrap tersebut dan masukan kedalam folder assets pada codeigniter projek.



gambar

3. Buka visual studio code pergi ke bagian liblaries dan buat 1 file bernama template.php



gambar

4. Lalu buka Isikan Source code template untuk memanggil perbagian dari template pada aplikasi

Terdapat template untuk :

1. Home
2. Kunjungan
3. Dashboard user
4. Dan admin

```
Template.php

<?php

class Template
{
```

```

protected $_ci;
function __construct()
{
    $this->_ci = &get_instance();
}

function home($template,$data=NULL)
{
    $data['_content']=$this->_ci->load->view($template,$data,true);
    $data['_header']=$this->_ci->load->view('home/header',$data,true);
    $data['_top_menu']=$this->_ci->load-
>view('home/top_menu',$data,true);
    $data['_menu']=$this->_ci->load->view('home/menu',$data,true);
    $data['_menu2']=$this->_ci->load-
>view('home/menu2',$data,true);
    $data['_kontak']=$this->_ci->load-
>view('home/kontak',$data,true);
    $data['_footer']=$this->_ci->load->view('home/footer',$data,true);
    $data['_alamat']=$this->_ci->load->view('home/alamat',$data,true);
    $this->_ci->load->view('home/template.php',$data);
}

function Kunjungan($content,$data=null)
{
    $data['_content']=$this->_ci->load->view($content,$data,true);
    $data['_top_menu']=$this->_ci->load-
>view('Kunjungan/top_menu',$data,true);
    $this->_ci->load->view('Kunjungan/template_Kunjungan',$data);
}

```

```

}

function dashboard_user($content,$data=null)
{
    $data['_content']=$this->_ci->load->view($content,$data,true);
    $data['_top_menu']=$this->_ci->load-
>view('Kunjungan/dashboard/top_menu',$data,true);
    $this->_ci->load->view('Kunjungan/dashboard/template',$data);
}

function admin($content,$data=null)
{
    $data['_content']=$this->_ci->load->view($content,$data,true);
    $data['_top_menu']=$this->_ci->load-
>view('admin/top_menu',$data,true);
    $data['_sidebar']=$this->_ci->load-
>view('admin/sidebar',$data,true);
    $this->_ci->load->view('admin/template',$data);
}
}

```

5.4.4.1 View ADMIN

Untuk dapat menampilkan hasil dari CSS bootsrap harus dimasukan kedalam kodingan view pertemplate yang akan dipanggil, supaya dapat dengan mudah untuk memenggil tampilan tersebut ketika membangun tampilan view pada program yaitu dengan cara seperti berikut:

1. BOOTSRAP TOP MENU ADMIN

Membuat file dalam folder view>Admin>Top_menu.php

gambar

Top_Menu.php

```
<div class="navbar">

    <div class="navbar-inner">

        <div class="container-fluid">

            <a href="#" class="brand">

                <small>

                    <i class="icon-user"></i>

                    Admin Layanan Kunjungan

                </small>

                    <i class="icon-user"></i>

                </a><!--/.brand-->

            <ul class="nav ace-nav pull-right">

                <li class="grey">

                    <ul class="pull-right dropdown-navbar dropdown-menu dropdown-caret dropdown-closer">
```

```
</ul>

</li>

<li class="purple">
    </a>
    <ul class="pull-right dropdown-navbar navbar-
pink dropdown-menu dropdown-caret dropdown-closer">
        <li>
            <li>
                <a href="#">
                    
                    <span class="msg-body">
                        <span class="msg-title">
                            <span class="blue">Bob:</span>
                            Nullam quis risus eget urna mollis ornare .
                        ..
                    </span>
                </a>
            </li>
        </ul>
    </li>
</ul>
```

```
<span class="msg-time">  
    <i class="icon-time"></i>  
    <span>3:15 pm</span>  
    </span>  
</span>  
</a>  
</li>  
  
<li>  
    <a href="#">  
        See all messages  
        <i class="icon-arrow-right"></i>  
    </a>  
</li>  
</ul>  
</li>  
  
<li class="light-blue">  
    <a data-  
        toggle="dropdown" href="#" class="dropdown-toggle">
```

```


<span class="user-info">
    <small>Welcome,</small>
    <?= $this->session->userdata('nama')?>
</span>

<i class="icon-caret-down"></i>
</a>

<ul class="user-menu pull-right dropdown-
menu dropdown-yellow dropdown-caret dropdown-closer">
    <li>
        <a href="#">
            </a>
    </li>
    <li>
        <a href="#">
            <i class="icon-user"></i>
            </a>
    </li>
```

```
<li class="divider"></li>

<li>

    <a href=<?php echo site_url('login/logout');?>" o
nclick="return confirm('Apakah anda yakin ingin keluar ?')">

        <i class="icon-off"></i>

        [Logout]

    </a>

</li>

</ul>

</li>

</ul><!--/.ace-nav-->

</div><!--/.container-fluid-->

</div><!--/.navbar-inner-->

</div>
```

2. BOOTSRAP SIDEBAR ADMIN

Membuat file dalam folder view>Admin>sidebar.php



gambar

Sidebar.php

```
<ul class="nav nav-list">  
    <li>  
        <a href="php echo site_url('admin/daftarpeserta');?&gt;"&gt;<br/            <i class="icon-user"></i>  
            <span class=""> Daftar Pengunjung </span>  
        </a>  
    </li>  
  
    <li>  
        <a href="php echo site_url('Admin_kelola_wbp/daftarwbp');?&gt;"&gt;<br/            <i class="icon-user"></i>  
            <span class="menu-text"> Daftar WBP </span>  
        </a>  
    </li>
```

```
<li>
    <a href="php echo site_url('Admin_kelola_jadwal/datajadwal');?&gt;"
">
    <i class="icon-list"></i>
    <span class="menu-text"> Data Jadwal </span>
</a>
</li>

<li>
    <a href="php echo site_url('admin/daftarverifpeserta');?&gt;"&gt;
        &lt;i class="icon-text-width"&gt;&lt;/i&gt;
        &lt;span class="menu-text"&gt; Verifikasi Pengunjung &lt;/span&gt;
    &lt;/a&gt;
&lt;/li&gt;

&lt;li&gt;
    &lt;a href="<?php echo site_url('admin_pencarian/pencarian');?&gt;"&gt;
        &lt;i class="icon-search"&gt;&lt;/i&gt;
        &lt;span class="menu-text"&gt; Cari Napi &lt;/span&gt;
    &lt;/a&gt;
&lt;/li&gt;

&lt;li&gt;
    &lt;a href="<?php echo site_url('admin_pencarian/cari_kode');?&gt;"&gt;
        &lt;i class="icon-search"&gt;&lt;/i&gt;
        &lt;span class="menu-text"&gt; Cari Kode Antrian &lt;/span&gt;
    &lt;/a&gt;
&lt;/li&gt;</pre
```

```
</li>

<li>
    <a href=<?php echo site_url('admin_pencarian/cari_tanggal');?>">
        <i class="icon-time"></i>
        <span class="menu-text"> Cari Tanggal </span>
    </a>
</li>

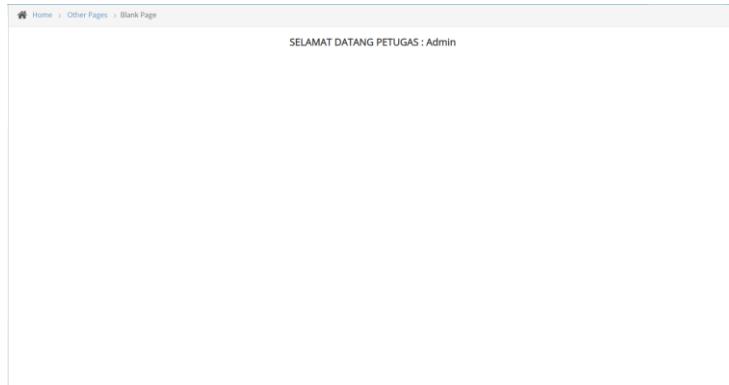
<li>
    <a href=<?php echo site_url('admin/daftarpesan');?>">
        <i class="icon-list-alt"></i>
        <span class="menu-text"> Pesan </span>
    </a>
</li>

</ul>
```

3. BOOTSRAP TEMPLATE ADMIN

Membuat file dalam folder view>Admin>TEMPLATE.php

Berfungsi sebagai main content yang akan menampilkan data data berupa data gambar,table dll yang akan di panggil dari database



Gambar

TEMPLATE.PHP

```
<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="utf-8" />
        <title>Blank Page - Ace Admin</title>

        <meta name="description" content="" />
        <meta name="viewport" content="width=device-width, initial-
scale=1.0" />

        <!--basic styles-->
        <link href=<?php echo base_url('assets/ace/css/bootstrap.min.css') ;?>" rel="stylesheet" />
        <link href=<?php echo base_url('assets/ace/css/bootstrap-
responsive.min.css');?>" rel="stylesheet" />

        <link rel="stylesheet" href=<?php echo base_url('assets/ace/css/fo
nt-awesome.min.css');?>" />
```

```
<!--[if IE 7]>
<link rel="stylesheet" href="assets/css/font-awesome-
ie7.min.css" />
<![endif]-->

<!--page specific plugin styles-->

<!--fonts-->

<link rel="stylesheet" href="http://fonts.googleapis.com/css?family
=Open+Sans:400,300" />

<!--ace styles-->

<link rel="stylesheet" href="<?php echo base_url('assets/ace/css/ac
e.min.css');?>" />
<link rel="stylesheet" href="<?php echo base_url('assets/ace/css/ac
e-responsive.min.css');?>" />
<link rel="stylesheet" href="<?php echo base_url('assets/ace/css/ac
e-skins.min.css');?>" />

<!--[if lte IE 8]>
<link rel="stylesheet" href="assets/css/ace-ie.min.css" />
<![endif]-->
```

```

<!--inline styles related to this page-->
<meta http-equiv="Content-Type" content="text/html; charset=utf-
8" />

<link href=<?php echo base_url('assets/datatables/css/jquery.dataTables.min.css');?>" rel="stylesheet">
<link href=<?php echo base_url('assets/datatables/css/dataTables.bootstrap.min.css');?>" rel="stylesheet">

</head>

<body>

<?php echo $_top_menu;?>

<div class="main-container container-fluid">
<a class="menu-toggler" id="menu-toggler" href="#">
    <span class="menu-text"></span>
</a>

<div class="sidebar" id="sidebar">
<br>
<img src=<?php echo base_url('assets/images/logo.png');?>" wi-
dth="200" height="100">
<br>
<div class="sidebar-shortcuts" id="sidebar-shortcuts">

```

```
<div class="sidebar-shortcuts-large" id="sidebar-shortcuts-large">

    <button class="btn btn-small btn-success">

        </button>

    <button class="btn btn-small btn-info">

        </button>

    <button class="btn btn-small btn-warning">

        </button>

    <button class="btn btn-small btn-danger">

        </button>

</div>

<div class="sidebar-shortcuts-mini" id="sidebar-shortcuts-mini">

    <span class="btn btn-success"></span>

    <span class="btn btn-info"></span>

    <span class="btn btn-warning"></span>

    <span class="btn btn-danger"></span>
```

```
</div>

</div><!--#sidebar-shortcuts-->

<?php echo $_sidebar;?>
<!--/.nav-list-->

<div class="sidebar-collapse" id="sidebar-collapse">
    <i class="icon-double-angle-left"></i>
</div>
</div>

<div class="main-content">
    <div class="breadcrumbs" id="breadcrumbs">
        <ul class="breadcrumb">
            <li>
                <i class="icon-home home-icon"></i>
                <a href="#">Home</a>

                <span class="divider">
                    <i class="icon-angle-right arrow-icon"></i>
                </span>
            </li>
            <li>
                <a href="#">Other Pages</a>

                <span class="divider">
                    <i class="icon-angle-right arrow-icon"></i>
                </span>
            </li>
        </ul>
    </div>
</div>
```

```

        </span>
    </li>
    <li class="active">Blank Page</li>
</ul><!--.breadcrumb-->

</div>

<div class="page-content">
    <div class="row-fluid">
        <div class="span12">
            <!--PAGE CONTENT BEGINS-->

            <!--PAGE CONTENT ENDS-->
            <?php echo $_content;?></div><!--.span-->
        </div><!--.row-fluid-->
    </div><!--.page-content-->

    <!--#ace-settings-container-->
</div><!--.main-content-->
</div><!--.main-container-->

<a href="#" id="btn-scroll-up" class="btn-scroll-up btn btn-small btn-inverse">
    <i class="icon-double-angle-up icon-only bigger-110"></i>
</a>

<!--basic scripts-->

```

```
<!--[if !IE]>-->
<script src="http://ajax.googleapis.com/ajax/libs/jquery/2.0.3/jquery.min.js"></script>

<!--<![endif]-->

<!--[if IE]>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js"></script>
<![endif]-->

<!--[if !IE]>-->

<script type="text/javascript">
    window.jQuery || document.write("<script src='<?php echo base_url('assets/ace/js/jquery-2.0.3.min.js');?>'>"+"<"+"</script>");
</script>

<!--<![endif]-->

<!--[if IE]>
<script type="text/javascript">
    window.jQuery || document.write("<script src='<?php echo base_url('assets/ace/js/jquery-1.10.2.min.js');?>'>"+"<"+"</script>");
</script>
<![endif]-->
```

```

<script type="text/javascript">
    if("ontouchend" in document) document.write("<script src='<?ph
p echo base_url('assets/ace/js/jquery.mobile.custom.min.js');?>'>"+"<"+
"/script>");

</script>
<script src="<?php echo base_url('assets/ace/js/bootstrap.min.js');?>"></script>

<!--page specific plugin scripts-->

<!--ace scripts-->

<script src="<?php echo base_url('assets/ace/js/ace-
elements.min.js');?>"></script>
<script src="<?php echo base_url('assets/ace/js/ace.min.js');?>"></s
cript>

<script type="text/javascript" src="<?php echo base_url('assets/dat
atables/js/jquery.dataTables.min.js');?>"></script>

<script type="text/javascript">
    $(document).ready(function(){
        $('.data').DataTable();
    });
</script>
<!--inline scripts related to this page-->
</body>
</html>

```

5.4.4.2 View HOME

Untuk dapat menampilkan hasil dari CSS bootsrap harus dimasukan kedalam kodingan view pertemplate yang akan dipanggil, supaya dapat dengan mudah untuk memenggil tampilan tersebut ketika membangun tampilan view pada program yaitu dengan cara seperti berikut:

1. BOOTSRAP TEMPLATE HOME

Membuat file dalam folder view>home>Template.php

Berfungsi untuk menyusun tampilan mana yang terlebih dahulu ditampilkan

- HEADER
- TOPMENU
- MENU2
- ALAMAT
- KONTAK
- FOOTER

Template.php

```
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,initial-scale=1">

    <title>Layanan Online Rutan 1 Bandung</title>
```

```
<link href="<?php echo base_url('assets/bootstrap/css/bootstrap.min.css');?  
?>" rel="stylesheet">  
    <link href="<?php echo base_url('assets/bootstrap/css/custom.css');?>" rel  
="stylesheet">  
  
<!--[if lt IE 9]>  
    <script src="<?php echo base_url('assets/html5shiv.min.js');?>"></sc  
ript>  
    <script src="<?php echo base_url('assets/respond.min.js');?>"></scri  
pt>  
    <![endif]-->  
  
</head>  
<style>.navbar-default {  
    background-color: #ff000000;  
    border-color: #e7e7e7;  
</style>  
<body id="myPage" data-spy="scroll" data-target=".navbar" data-  
offset="60">  
  
<div class="row">  
    <div class="col-md-12">  
        <?php echo $_header;?>  
    </div>  
</div>
```

```
<div class="row">
  <div class="col-md-12">
    <?php echo $_top_menu;?>
  </div>
</div>
```

```
<div class="row">
  <div class="col-md-12">
    <?php echo $_menu2;?>
  </div>
```

```
<div class="row">
  <div class="col-md-12">
    <?php echo $_alamat;?>
  </div>
```

```
</div>
```

```
<div class="row">
  <div class="col-md-12">
    <?php echo $_kontak;?>
  </div>
```

```
</div>

<div class="row">
    <div class="col-md-12">
        <?php echo $_footer;?>
    </div>

</div>
</div>

<script type="text/javascript" src="<?php echo base_url('assets/bootstrap/js/jquery.min.js');?>"></script>
<script type="text/javascript" src="<?php echo base_url('assets/bootstrap/js/bootstrap.min.js');?>"></script>

<script>
$(document).ready(function(){
    // Add smooth scrolling to all links in navbar + footer link
    $(".navbar a, footer a[href='#myPage']").on('click', function(event) {

        // Prevent default anchor click behavior
        event.preventDefault();

        // Store hash
        var hash = this.hash;

        // Using jQuery's animate() method to add smooth page scroll

```

```
// The optional number (900) specifies the number of milliseconds it takes to scroll to the specified area
$('html, body').animate({
  scrollTop: $(hash).offset().top
}, 900, function(){
  // Add hash (#) to URL when done scrolling (default click behavior)
  window.location.hash = hash;
});
});

})
</script>

</body>
</html>
```

2. BOOTSRAP HEADER HOME

Membuat file dalam folder view>home>header.php



Gambar

Header.php

```
<div class="container-fluid bg-1 text-center">
    <h1>Rutan Klas 1 Bandung</h1>
    

    <h3>Layanan Online Rutan Klas 1 Bandung</h3>
</div>
```

3. BOOTSRAP TOP MENU HOME

Membuat file dalam folder view>home>Top_menu.php

LOGIN KONTAK

DAFTAR DISINI

gambar

TOP_MENU.PHP

```
<nav class="navbar navbar-default navbar-fixed-top">
    <div class="container">
        <div class="navbar-header">
            <button type="button" class="navbar-toggle" data-
toggle="collapse" data-target="#myNavbar">
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
                <span class="icon-bar"></span>
```

```
</button>

<a class="navbar-brand" data-toggle="modal" data-
target="#myModal" href="#">LOGIN</a>
<a class="navbar-brand" href="#kontak">KONTAK</a>
</div>

<div class="collapse navbar-collapse" id="myNavbar">
<ul class="nav navbar-nav navbar-right">
<li><a href="#menu">DAFTAR DISINI</a></li>
</ul>
</div>
</div>
</nav>

<script type="text/javascript">
function cekform()
{
if (!$("#username").val())
{
alert('Maaf Username tidak boleh kosong');
$("#username").focus();
return false;
}
if (!$("#password").val())
{
alert('Maaf Password tidak boleh kosong');
$("#password").focus();
return false;
}
}
```

```

        }

</script>

<!-- Modal -->
<div class="modal fade" id="myModal" role="dialog">
  <div class="modal-dialog">

    <!-- Modal content-->
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close" data-
dismiss="modal">&times;</button>
        <div class="modalh4"><h4><span class="glyphicon glyphicon-
lock"></span> LOGIN</h4></div>
      </div>
      <div class="modal-body">
        <form role="form" action=<?php echo site_url('login/login');?>" meth-
od="post" onSubmit="return cekform();">
          <div class="form-group">
            <label for="psw"><span class="glyphicon glyphicon-
user"></span> Username</label>
            <input type="text" class="form-
control" id="username" name="username" placeholder="Username">
          </div>
          <div class="form-group">
            <label for="usrname"><span class="glyphicon glyphicon-
lock"></span> Password</label>

```

```

<input type="password" class="form-
control" id="password" name="password" placeholder="Password">
</div>

<?php
$info = $this->session->flashdata('info');
if (!empty($info))
{
    echo $info;
}
?>

<button type="submit" class="btn btn-block">Login
    <span class="glyphicon glyphicon-log-in"></span>
</button>
</form>
</div>
<div class="modal-footer">
    <button type="submit" class="btn btn-danger btn-default pull-left" data-
dismiss="modal">
        <span class="glyphicon glyphicon-remove"></span> Cancel
    </button>
    <p>Need <a href="#">help?</a></p>
</div>
</div>
</div>
</div>

```

4. BOOTSRAP MENU2 HOME

Membuat file dalam folder view>Admin>Menu2.php



gambar

MENU2.PHP

```
<div id="menu" class="container-fluid text-center">
    <h2>MENU</h2>
    <h4></h4>
    <br>
    <div class="row">
        <div class="col-sm-4 col4">
```

```

<a href=<?php echo site_url('Kunjungan/prosedur');?>><span class="gl
yphicon glyphicon-list-alt logo-small"></span>
<h4>PROSEDUR KUNJUNGAN </h4></a>
<p>Cek jadwal Kunjungan Rutan Klas 1 Bandung</p>
</div>
<div class="col-sm-4 col4">
<a data-toggle="modal" data-
target="#cekModal" href="#"><span class="glyphicon glyphicon-user logo-
small"></span>
<h4>CEK KODE PENDAFTARAN</h4></a>
<p>Cek Nomer Kode Pendaftaran Antrian Rutan Klas 1 Bandung</p>
</div>

<div class="col-sm-4 col4">
<a href=<?php echo site_url('Kunjungan');?>><span class="glyp
hic on glyphicon-user logo-small"></span>
<h4>PENDAFTARAN KUNJUNGAN</h4></a>
<p>Pendaftaran Kunjungan Rutan Klas 1 Bandung</p>
</div>
</div>

<div class="modal fade" id="cekModal" role="dialog">
<div class="modal-dialog">
```

```

<!-- Modal content-->
<div class="modal-content">
    <div class="modal-header">
        <button type="button" class="close" data-
dismiss="modal">&times;</button>
        <div class="modalh3"><h4>CEK PENDAFTARAN</h4></div>
    </div>
    <div class="modal-body">
        <div class="form-group">
            <label for="psw"><span class="glyphicon glyphicon-
user"></span> Nama</label>
            <input type="text" class="form-
control" id="nama" name="nama" placeholder="Masukkan Nama Pendaftar
Kunjungan">
        </div>
        <button id="search" onclick="getInputValue();" class="btn btn-
block">Cek
            <span class="glyphicon glyphicon-log-in"></span>
        </button>
        <hr>
        <h4 id="header4" style="display: none;"><center>Nama Tidak Dite
mukan</center></h4>
        <table class="table table-
bordered" id="table" style="display: none;">
            <thead>
                <tr>
                    <th colspan="2" id="header2"><center><b>List Pencarian Nam
a</b></center></th>

```

```

        </tr>
    </thead>
    <tbody class="nama_pendaftar">
        <tr>
            <td id="nama_lengkapi">kakaka</td>
        </tr>
    </tbody>
</table>
<table class="table table-
bordered" id="table1" style="display: none;">
    <thead>
        <tr>
            <th colspan="2" id="header2"><center><b>Data Pendaftaran T
erakhir</b></center></th>
        </tr>
        <tr>
            <th>Nama Pengunjung</th>
            <td id="nama_pendaftar"></td>
        </tr>
        <tr>
            <th>Nomor Antrian</th>
            <td id="kode_antrian"></td>
        </tr>
        <tr>
            <th>Jam Kunjungan</th>
            <td id="jam_kunjungan"></td>
        </tr>
        <th>Status</th>

```

```
<td id="status"></td>
</tr>
</tr>
<th>Keterangan</th>
<td id="keterangan"></td>
</tr>
</thead>
</table>
</div>
<div class="modal-footer">
    <button type="submit" class="btn btn-danger btn-default pull-left" data-dismiss="modal">
        <span class="glyphicon glyphicon-remove"></span> Cancel
    </button>
</div>
</div>
</div>
</div>
```

5. BOOTSRAP ALAMAT HOME

Membuat file dalam folder view>home>alamat.php



Gambar

ALAMAT.PHP

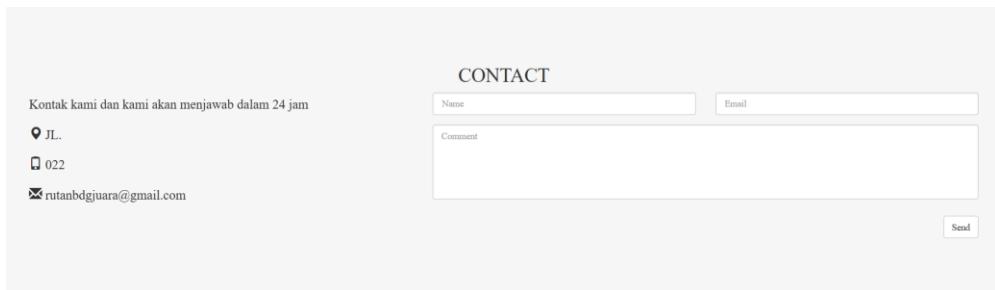
```
<div class="container-fluid bg-2 text-center">

<div class="row">
<div class="col-sm-8">
<h2>Jl. Jakarta, Kebonwaru, NO.29 Kec. Batununggal, Kota Bandung</h2>
>
<h2>Jawa Barat 40272</h2>
</div>
<div class="col-sm-4 col4">
<span class="glyphicon glyphicon-home logo"></span>
</div>
</div>

</div>
```

6. BOOTSRAP KONTAK HOME

7. Membuat file dalam folder view>home>kontak.php



Gambar

KONTAK.PHP

```
<div id="kontak" class="container-fluid abu">
<h2 class="text-center">CONTACT</h2>
<div class="row">
<div class="col-sm-5">
<p>Kontak kami dan kami akan menjawab dalam 24 jam</p>
<p><span class="glyphicon glyphicon-map-marker"></span> JL.</p>
<p><span class="glyphicon glyphicon-phone"></span> 022</p>
<p><span class="glyphicon glyphicon-
envelope"></span> rutanbdgjuara@gmail.com</p>
</div>
<div class="col-sm-7">
<div class="row">
<div class="col-sm-6 form-group">
```

```
<input class="form-
control" id="name" name="name" placeholder="Name" type="text" required
>
</div>
<div class="col-sm-6 form-group">
<input class="form-
control" id="email" name="email" placeholder="Email" type="email" requir-
ed>
</div>
</div>
<textarea class="form-
control" id="comments" name="comments" placeholder="Comment" rows=
"5"></textarea><br>
<div class="row">
<div class="col-sm-12 form-group">
<button class="btn btn-default pull-
right" type="submit">Send</button>
</div>
</div>
</div>
</div>
```

5.4.4.3 View Kunjungan

Untuk dapat menampilkan hasil dari CSS bootstrap harus dimasukan kedalam kodingan view pertemplate yang akan dipanggil, supaya dapat

dengan mudah untuk memanggil tampilan tersebut ketika membangun tampilan view pada program yaitu dengan cara seperti berikut:

1. BOOTSRAP TEMPLATE HOME

Membuat file dalam folder view>home>Template.php

Berfungsi untuk menyusun tampilan mana yang terlebih dahulu ditampilkan

- TOPMENU
- CONTENT

TEMPLATE.PHP

```
<!DOCTYPE html>
<html lang="id">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width,initial-scale=1">

    <title>Layanan Online Rutan Klas 1 Bandung</title>

    <link href=<?php echo base_url('assets/bootstrap/css/bootstrap.min.css');?>" rel="stylesheet">
    <link href=<?php echo base_url('assets/bootstrap/css/custom2.css');?>" rel="stylesheet">
    <link href=<?php echo base_url('assets/datatables/css/jquery.dataTables.min.css');?>" rel="stylesheet">
    <link href=<?php echo base_url('assets/datatables/css/dataTables.bootstrap.min.css');?>" rel="stylesheet">
```

```
<!--[if lt IE 9]>
    <script src=<?php echo base_url('assets/html5shiv.min.js');?>"></sc
ript>
    <script src=<?php echo base_url('assets/respond.min.js');?>"></scri
pt>
<![endif]-->

</head>
<style>
.navbar-inverse {
    background-color: #222;
    border-color: #080808;
    position: fixed;
    right: 0;
    left: 0;
    z-index: 1030;
    margin-bottom: 0;
}</style>
<body>
    <div class="container">
        <div class="row">

            </div>
        </div>

        <div class="row">
            <div class="col-md-12">
                <?php echo $_top_menu;?>
            </div>
        </div>
    </div>
</body>
```

```

        </div>
    </div>

<div class="container">
    <div class="row">
        <div class="col-md-12"><br><br><br>
            <?php echo $_content;?>
        </div>
    </div>

</div>

<script type="text/javascript" src="<?php echo base_url('assets/bootstrap/js/jquery.min.js');?>"></script>
<script type="text/javascript" src="<?php echo base_url('assets/bootstrap/js/bootstrap.min.js');?>"></script>
<script type="text/javascript" src="<?php echo base_url('assets/datatables/js/jquery.dataTables.min.js');?>"></script>
<script type="text/javascript">
    $(document).ready(function(){
        $('.data').DataTable();
    });
</script>
</body>
</html>

```

1. BOOTSRAP TEMPLATE HOME

Membuat file dalam folder view>Kunjungan>Top_menu.php



Gambar

TOP_MENU.PHP

```
<nav class="navbar navbar-inverse">
<div class="container">
    <!-- Brand and toggle get grouped for better mobile display -->
    <div class="navbar-header">
        <button type="button" class="navbar-toggle collapsed" data-
toggle="collapse" data-target="#bs-example-navbar-collapse-1" aria-
expanded="false">
            <span class="sr-only">Toggle navigation</span>
            <span class="icon-bar"></span>
            <span class="icon-bar"></span>
            <span class="icon-bar"></span>
        </button>
        <a class="navbar-
brand" href=<?php echo base_url();?>><span class="glyphicon glyphicon-
home" aria-hidden="true"></span> Beranda</a>
    </div>

    <!-- Collect the nav links, forms, and other content for toggling -->
    <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-
1">
        <ul class="nav navbar-nav">
            <li><a href=<?php echo site_url('kunjungan/daftar');?>><span class=
"glyphicon glyphicon-pencil" aria-
hidden="true"></span> Pendaftaran</a></li>
```

```
<li><a href=<?php echo site_url('kunjungan/prosedur');?>"><span clas
s="glyphicon glyphicon-list-alt" aria-
hidden="true"></span> Prosedur Pendaftaran</a></li>
<li class="dropdown">

<ul class="dropdown-menu">
<li><a href=<?php echo site_url('kunjungan/prosedur');?>"><span c
lass="glyphicon glyphicon-bookmark" aria-
hidden="true"></span> Prosedur</a></li>
<li role="separator" class="divider"></li>

</ul>
</li>
<li class="dropdown">

<ul class="dropdown-menu">
</ul>
</li>
</ul>

<form class="navbar-form navbar-
right" role="search" action=<?php echo site_url('login/login');?>" method="post">
<div class="form-group">
```

```

<input type="text" class="form-control" placeholder="Username" name="username" value=<?php echo set_value('username')?>>

<input type="password" class="form-control" placeholder="Password" name="password" value=<?php echo set_value('password')?>>

</div>

<button type="submit" class="btn btn-default">Masuk</button>

</form>

</div><!-- /.navbar-collapse -->
</div><!-- /.container-fluid -->
</nav>

```

2. BOOTSRAP TEMPLATE HOME

Membuat file dalam folder view>Kunjungan>Content.php



Gambar

CONTENT.PHP

```

<div class="jumbotron">
<h1>Selamat Datang!</h1>

```

<p>- Selamat Datang di Layanan Online Rutan Klas 1 Bandung.

- Sebelum melakukan pendaftaran,
sebaiknya anda memahami prosedur pendaftaran terlebih dahulu dihalaman

<?php echo anchor('Kunjungan/prosedur','Prosedur Pendaftaran');?>

></p>

<p>- Pastikan juga bahwa anda telah mengetahui Jadwal Kunjungan Rutan Klas 1 Bandung yang berada dihalaman

<?php echo anchor('#','Jadwal');?>.

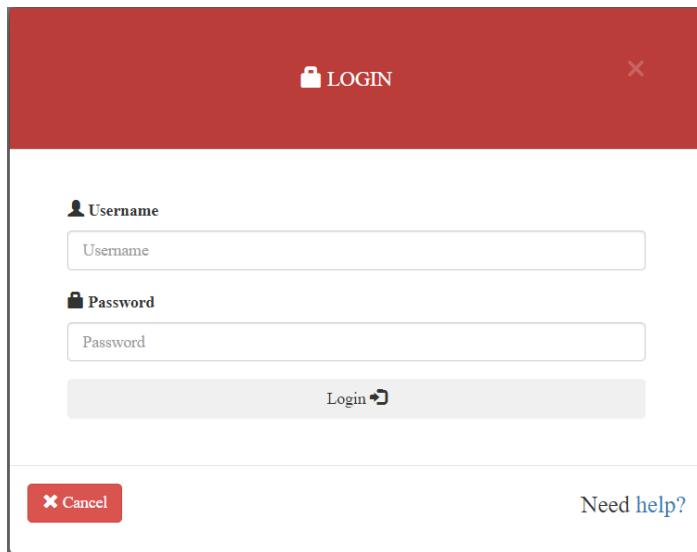
- Untuk mengetahui informasi terbaru mengenai Verifikasi kunjungan anda dapat melihatnya
melalui Email masing masing</p>

<p>- Jika anda sudah memahami prosedur pendaftaran, silahkan klik tombol Daftar dibawah ini untuk melakukan Pendaftaran!</p>

<p><a class="btn btn-primary btn-lg" href="<?php echo site_url('kunjungan/daftar');?>" role="button">Daftar</p>

</div>

5.5 MEMBANGUN HALAMAN LOGIN



Gambar

5.5.1 Source Code MODEL Login

```
function cek_akun($username,$password)
{
    $this->db->select('*');
    $this->db->from('user');
    $this->db->where('username',$username);
    $this->db->where('password',$password);

    return $this->db->get()->row();
}
```

Berfungsi untuk mengkoneksikan kedalam table database yaitu table User dimana yang di ambil adalah Username dan password.

5.5.2 Source Code VIEW Login

```
<script type="text/javascript">
```

```

function cekform()
{
    if (!$("#username").val())
    {
        alert('Maaf Username tidak boleh kosong');
        $("#username").focus();
        return false;
    }
    if (!$("#password").val())
    {
        alert('Maaf Password tidak boleh kosong');
        $("#password").focus();
        return false;
    }
}

</script>

<!-- Modal -->
<div class="modal fade" id="myModal" role="dialog">
    <div class="modal-dialog">

        <!-- Modal content-->
        <div class="modal-content">
            <div class="modal-header">
                <button type="button" class="close" data-
dismiss="modal">&times;</button>
                <div class="modalh4"><h4><span class="glyphicon glyphicon-
lock"></span> LOGIN</h4></div>

```

```

    </div>
    <div class="modal-body">
        <form role="form" action="php echo site_url('login/login');?&gt;" method="post" onSubmit="return cekform();"&gt;
            &lt;div class="form-group"&gt;
                &lt;label for="psw"&gt;&lt;span class="glyphicon glyphicon-user"&gt;&lt;/span&gt; Username&lt;/label&gt;
                &lt;input type="text" class="form-control" id="username" name="username" placeholder="Username"&gt;
            &lt;/div&gt;
            &lt;div class="form-group"&gt;
                &lt;label for="usrname"&gt;&lt;span class="glyphicon glyphicon-lock"&gt;&lt;/span&gt; Password&lt;/label&gt;
                &lt;input type="password" class="form-control" id="password" name="password" placeholder="Password"&gt;
            &lt;/div&gt;

        &lt;?php
        $info = $this-&gt;session-&gt;flashdata('info');
        if (!empty($info))
        {
            echo $info;
        }
        ?&gt;

        &lt;button type="submit" class="btn btn-block"&gt;Login
            &lt;span class="glyphicon glyphicon-log-in"&gt;&lt;/span&gt;
        &lt;/button&gt;
    </pre

```

```

</form>
</div>
<div class="modal-footer">
    <button type="submit" class="btn btn-danger btn-default pull-left" data-
dismiss="modal">
        <span class="glyphicon glyphicon-remove"></span> Cancel
    </button>
    <p>Need <a href="#">help?</a></p>
</div>
</div>

```

5.5.3 Source Code CONTROLLER Login

```

function login()
{
    $username = $this->input->post('username',true);
    $password = $this->input->post('password',true);

    $akun = $this->admin_model->cek_akun($username,$password);
    $temp_account = count($akun);

    if ($temp_account > 0)
    {
        if($akun->level == 'admin') {
            $data = array(
                'level'=>$akun->level,
                'logged_in'=>true,
                'nama'=>$akun->nama_lengkap
            );
        }
    }
}

```

```

    $this->session->set_userdata($data);
    $this->template->admin('admin/beranda');

} else if ($akun->level == 'peserta') {
    $data_session = array(
        'level'=>$akun->level,
        'nama_panggilan'=>$akun->nama_panggilan,
        'id_peserta'=>$akun->id_peserta,
        'logged_in'=>true
    );

    $this->session->set_userdata($data_session);
    $data['history'] = $this->Kunjungan_model-
>get_history_kunjungan($akun->id_peserta);
    $this->template-
>dashboard_user('Kunjungan/dashboard/home', $data);

}else if($akun->level=="kecamatan"){
    $data_session = array(
        'level'=>$akun->level,
        'nama_panggilan'=>$akun->nama_panggilan,
        'status'=>$akun->status,
        'logged_in'=>true
    );

    $this->session->set_userdata($data_session);
    $this->template->kecamatan('kecamatan/beranda');
}

```

```

    }

}

else

{

echo "<script>alert('Username & Password Salah')</script>";

$this->template->home('home/content');

}

}

```

Berfungsi sebagai fungsi untuk melakukan login, setelah login berhasil akan masuk kedalam masing masing role yang sudah dibuat, terdapat 2 role yaitu :

- Admin
- Pengunjung

5.6 MEMBANGUN HALAMAN DAFTAR AKUN

Gambar

5.6.1 Source Code MODEL Daftar

```

function cek_username($username)
{
    $this->db->select('*');
    $this->db->from('user');
}

```

```
$this->db->where('username',$username);

return $this->db->get()->row();

}
```

5.6.2 Source Code VIEW Daftar

```
<div class="panel panel-info">
<div class="panel-heading">Form Pendaftaran</div>
<div class="panel-body">
<p>
<form class="form-
horizontal" action=<?php echo site_url('kunjungan/proses_daftar');?>" meth-
od="post">
<div class="col-sm-10 col-sm-offset-2">

<?php $info=$this->session->flashdata('error');
if (!empty($info))
{
echo $info;
}
?>

<?php echo validation_errors();?>

</div>
```

```
<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">Username</label>
    <div class="col-sm-10">
        <input required type="text" class="form-
control" name="username" value="<?php echo set_value('username')?>" id=
"inputPassword3" placeholder="Username untuk login" onkeyup="this.valu
e = this.value.toUpperCase()">
    </div>
</div>

<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">Password</label>
    <div class="col-sm-10">
        <input required type="password" class="form-
control" name="password" id="inputPassword3" placeholder="Password unt
uk login" >
    </div>
</div>

<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">Nama Lengkap</label>
    <div class="col-sm-10">
        <input required type="text" class="form-
control" name="nama_lengkap" value="<?php echo set_value('nama_lengka
```

```
|p')?>" id="inputPassword3" placeholder="Password" onkeyup="this.value =  
this.value.toUpperCase()">  
    </div>  
    </div>  
  
<div class="form-group">  
    <label for="inputPassword3" class="col-sm-2 control-  
label">Nama Panggilan</label>  
    <div class="col-sm-10">  
        <input required type="text" class="form-  
control" name="nama_panggilan" value=<?php echo set_value('nama_pang  
gilan')?>" id="inputPassword3" placeholder="Password" onkeyup="this.val  
ue = this.value.toUpperCase()">  
        </div>  
    </div>  
  
<div class="form-group">  
    <div class="col-sm-offset-2 col-sm-10">  
        <button type="submit" class="btn btn-success">Daftar</button>  
    </div>  
    </div>  
  
</form>  
</p>  
</div>  
  
</div>
```

5.6.3 Source Code CONTROLLER Daftar

```
function proses_daftar()
{
    $data['username'] = $this->input->post('username',true);
    $data['password'] = $this->input->post('password',true);
    $data['nama_lengkap'] = $this->input->post('nama_lengkap',true);
    $data['nama_panggilan'] = $this->input->post('nama_panggilan',true);

    $username = $this->input->post('username',true);
    $cek_username = $this->Kunjungan_model->cek_username($username);
    $num_account = count($cek_username);

    $this->form_validation->set_rules('username','Username','required');
    $this->form_validation->set_rules('password','Password','required');
    $this->form_validation->set_rules('nama_lengkap','Nama Lengkap','required');
    $this->form_validation->set_rules('nama_panggilan','Nama Panggilan','required');

    if ($this->form_validation->run() == FALSE)
    {
        $this->template->kunjungan('kunjungan/pendaftaran');
    }
    else
    {
```

```
if ($num_account > 0)
{
    $this->session->set_flashdata('error','<div class="alert alert-warning" role="alert">Maaf username sudah ada yang menggunakan</div>');
;
    $this->template->kunjungan('kunjungan/pendaftaran');
}
else
{
    $this->Kunjungan_model->daftar_peserta($data);
    $this->session->set_flashdata('error','<div class="alert alert-success" role="alert">Pendaftaran berhasil. Silahkan Login dengan menggunakan Akun yang sudah didaftarkan</div>');
    $this->template->kunjungan('kunjungan/pendaftaran');
}

}
```

Berfungsi sebagai fungsi untuk melakukan login

5.7 HALAMAN PENGISIAN FORMULIR KUNJUNGAN

Formulir Pendaftaran Kunjungan

DATA KUNJUNGAN

No pendaftaran	0118
No Identitas	336555
Nama Lengkap	WILDAN
Nama Panggilan	wildan
Jenis kelamin	laki-laki
Alamat Tempat Tinggal	adadas
Status Hubungan	keluarga - Suami
Email	wldhaustar@gmail.com
Upload foto pengunjung	<input type="button" value="Choose File"/> No file chosen
Upload foto KTP	<input type="button" value="Choose File"/> No file chosen

Gambar?

PENGUNJUNG PENGIKUT MAXIMAL 2

Nama Pengunjung 1	nama pengunjung pengikut 1
Jenis kelamin	-
No KTP	No KTP
Upload FOTO Pengikut 1	<input type="button" value="Choose File"/> No file chosen
Upload FOTO KTP Pengikut 1	<input type="button" value="Choose File"/> No file chosen
Nama Pengunjung 2	nama pengunjung pengikut 1
Jenis kelamin	-
No KTP	No KTP
Upload FOTO Pengikut 2	<input type="button" value="Choose File"/> No file chosen
Upload FOTO KTP Pengikut 2	<input type="button" value="Choose File"/> No file chosen

Gambar?

WBP YANG DIKUNJUNGI

Pilih Tanggal	mm/dd/yyyy
Pilih Jadwal	09:00:00 - 10:00:00
Nama WBP	Nama WBP berserta BIN
pasal	pasal
Blok Sel tahanan	blok
Upload foto WBP	<input type="button" value="Choose File"/> No file chosen



Kirim

Gambar

5.7.1 Source Code MODEL Formulir Kunjungan

```
function select_by_id($id_peserta)
{
    $this->db->select('*');
    $this->db->from('user');
    $this->db->where('id_peserta',$id_peserta);

    return $this->db->get()->row();
```

5.7.2 Source Code VIEW Formulir Kunjungan

```
<div class="panel panel-success">
    <div class="panel-heading">
        <h3 class="panel-title">Formulir Pendaftaran Kunjungan</h3>
    </div>
    <div class="panel-body">

        <?php
        $info = $this->session->flashdata('info');
        if (!empty($info))
        {
            echo $info;
        }
        ?>

        <h1>DATA KUNJUNGAN</h1>
        <div class="progress">
```

```

<div class="progress-bar progress-bar-success" role="progressbar" aria-
valuenum="60" aria-valuemin="0" aria-
valuemax="100" style="width: 100%;">
    <span class="sr-only">100% Complete</span>
</div>
</div>

<?= form_open_multipart('kunjungan/up_biodata', ['class'=>'form-
horizontal']) ?>
<div class="form-group">
    <label for="inputEmail3" class="col-sm-2 control-
label">No pendaftaran</label>
    <div class="col-sm-10">
        <input type="text" class="form-
control" id="inputEmail3" placeholder="KTP/PASSPORT" name="nik" rea
donly value="<?php echo $peserta->id_peserta;?>">
    </div>
</div>
<div class="form-group">
    <label for="inputEmail3" class="col-sm-2 control-
label">No Identitas</label>
    <div class="col-sm-10">
        <input type="number" class="form-
control" id="inputEmail3" placeholder="KTP/PASSPORT" name="nik" val
ue="<?php echo $peserta->nik;?>">
    </div>
</div>

```

```
<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">Nama Lengkap</label>
    <div class="col-sm-10">
        <input type="text" onkeyup="this.value = this.value.toUpperCase()" cla
ss="form-
control" id="inputPassword3" placeholder="Nama" name="nama_lengkap"
value="<?php echo $peserta->nama_lengkap;?>">
    </div>
</div>

<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">Nama Panggilan</label>
    <div class="col-sm-10">
        <input type="text" onkeyup="this.value = this.value.toUpperCase()" cla
ss="form-
control" id="inputPassword3" placeholder="Nama" name="nama_panggilan"
" value="<?php echo $peserta->nama_panggilan;?>">
    </div>
</div>

<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">jenis kelamin</label>
    <div class="col-sm-10">
        <select class="form-control" name="jenis kelamin">
            <option><?php echo $peserta->jenis_kelamin;?></option>
```

```

<option>LAKI-LAKI</option>
<option>PEREMPUAN</option>
</select>
</div>
</div>

<div class="form-group">
<label for="inputPassword3" class="col-sm-2 control-
label">Alamat Tempat Tinggal</label>
<div class="col-sm-10">
<textarea row="5" onkeyup="this.value = this.value.toUpperCase()" clas-
s="form-
control" id="inputPassword3" name="alamat" value=<?php echo $peserta-
>alamat;?>" placeholder="Alamat"><?php echo $peserta-
>alamat;?></textarea>
</div>
</div>

<div class="form-group">
<label for="inputPassword3" class="col-sm-2 control-
label">Status Hubungan</label>
<div class="col-sm-10">
<select class="form-control" name="status_hubungan">
<option><?php echo $peserta->status_hubungan;?></option>
<option>keluarga - Suami</option>
<option>Keluarga - Istri</option>

```

```
<option>Keluarga - Ayah</option>
<option>Keluarga - Ibu</option>
<option>Keluarga - Adik</option>
<option>Keluarga - Saudara jauh</option>
<option>Teman</option>
</select>
</div>
</div>

<div class="form-group">
<label for="inputPassword3" class="col-sm-2 control-label">Email</label>
<div class="col-sm-10">
<input type="email" class="form-
control" id="inputPassword3" placeholder="email" name="email" value="<?
php echo $peserta->email;?>">
</div>
</div>

<div class="form-group">
<label for="inputPassword3" class="col-sm-2 control-
label">Upload foto pengunjung</label>
<div class="col-sm-10">
<input type="file" class="form-control" name="photo">
</div>
</div>

<div class="form-group">
```

```
<label for="inputPassword3" class="col-sm-2 control-
label">Upload foto KTP</label>
<div class="col-sm-10">
    <input type="file" class="form-control" name="fotokk">
</div>
</div>

<h1>PENGUNJUNG PENGIKUT MAXIMAL 2</h1>
<div class="progress">
    <div class="progress-bar progress-bar-success" role="progressbar" aria-
value="60" aria-valuemin="0" aria-
valuemax="100" style="width: 100%;">
        <span class="sr-only">100% Complete</span>
    </div>
</div>

<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">Nama Pengunjung 1 </label>
    <div class="col-sm-4">
        <input class="form-
control" id="inputPassword3" name="laki_laki" value="" placeholder="nam
a pengunjung pengikut 1" onkeyup="this.value = this.value.toUpperCase()">
    </div>
</div>

<div class="form-group">
```

```
<label for="inputPassword3" class="col-sm-2 control-
label">jenis kelamin</label>
<div class="col-sm-10">
<select class="form-control" name="perempuan">
<option>-</option>
<option>laki-laki</option>
<option>Perempuan</option>
</select>
</div>
</div>
<div class="form-group">
<label for="inputPassword3" class="col-sm-2 control-
label">No Ktp </label>
<div class="col-sm-4">
<input type="number" class="form-
control" id="inputPassword3" name="anak_anak" value="" placeholder="N
o KTP">
</div>
</div>
<div class="form-group">
<label for="inputPassword3" class="col-sm-2 control-
label">Upload FOTO Pengikut 1</label>
<div class="col-sm-10">
<input type="file" class="form-control" name="fotopengikut1">
</div>
</div>
<div class="form-group">
```

```
<label for="inputPassword3" class="col-sm-2 control-
label">Upload FOTO KTP Pengikut 1</label>
<div class="col-sm-10">
    <input type="file" class="form-control" name="ktppengikut1">
</div>
</div>

<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">Nama Pengunjung 2 </label>
    <div class="col-sm-3">
        <input class="form-
control" id="inputPassword3" name="jumlah" value="" placeholder="nama
pengunjung pengikut 1" onkeyup="this.value = this.value.toUpperCase()">
    </div>
</div>

<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">jenis kelamin</label>
    <div class="col-sm-10">
        <select class="form-control" name="jeniskelamin2">
            <option>-</option>
            <option>laki-laki</option>
            <option>Perempuan</option>
        </select>
    </div>
```

```
</div>

<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">No Ktp </label>
    <div class="col-sm-4">
        <input type="number" class="form-
control" id="inputPassword3" name="noktp2" value="" placeholder="No K
TP">
    </div>
</div>
<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">Upload FOTO Pengikut 2</label>
    <div class="col-sm-10">
        <input type="file" class="form-control" name="fotopengikut2">
    </div>
</div>
<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">Upload FOTO KTP Pengikut 2</label>
    <div class="col-sm-10">
        <input type="file" class="form-control" name="ktp pengikut2">
    </div>
</div>

<h1>WBP YANG DIKUNJUNGI</h1>
<div class="progress">
```

```

<div class="progress-bar progress-bar-success" role="progressbar" aria-
valueNow="60" aria-valuemin="0" aria-
valueMax="100" style="width: 100%;">
    <span class="sr-only">100% Complete</span>
</div>
</div>

<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">Pilih Tanggal </label>
    <div class="col-sm-4">
        <input type="date" min="<?=DATE('Y-m-d')?>" class="form-
control" id="inputPassword3" name="tanggal" value="" placeholder="Pilih
Tanggal">
    </div>
</div>

<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">Pilih Jadwal </label>
    <div class="col-sm-4">
        <select name="id_jadwal" class="form-control" id="inputPassword3" >
            <?php
                foreach ($jadwal as $data) {
                    ?>
                    <option value="<?php echo $data->id_jadwal; ?>">
                        <?php echo $data->dari; ?> - <?php echo $data->sampai; ?>
                    </option>
            <?php
        }
    </div>
</div>

```

```
<?php } ?>
</select>
</div>
</div>

<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">Nama WBP </label>
    <div class="col-sm-4">
        <input type="text" required class="form-
control" id="inputPassword3" name="wbpsertabin" value="" placeholder="Nama WBP beserta BIN" onkeyup="fotoFunction(this.value); this.value = th
is.value.toUpperCase()" >
    </div>
</div>
<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-label">pasal </label>
    <div class="col-sm-4">
        <input class="form-
control" required id="inputPassword3" name="pasal" value="" placeholder=
"pasal" onkeyup="this.value = this.value.toUpperCase()" >
    </div>
</div>
<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">Blok Sel tahanan </label>
    <div class="col-sm-4">
```

```
<input class="form-control" required id="inputPassword3" name="blok" value="" placeholder="blok" onkeyup="this.value = this.value.toUpperCase()">
</div>
</div>
<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-label">Upload foto WBP</label>
    <div class="col-sm-10">
        <input type="file" class="form-control" name="fotowbppengunjung">
    </div>
</div>
<div class="col-sm-4">
    <right>
        <img id="fotoWbp" src="" width="200" height="200" onclick="fotoClick(this.value)">
    </right>
</div>
</div>
<div class="progress">
    <div class="progress-bar progress-bar-success" role="progressbar" aria-valuenow="60" aria-valuemin="0" aria-valuemax="100" style="width: 100%;">
        <span class="sr-only">100% Complete</span>
    </div>
</div>
```

```

<input type="hidden" name="id_peserta" value="<?php echo $peserta-
>id_peserta;?>"/>
<input type="hidden" name="old_foto" value="<?php echo $peserta-
>dokumen;?>"/>
<input type="hidden" name="old_kk" value="<?php echo $peserta-
>kk;?>"/>
<button type="submit" class="btn btn-primary btn-lg btn-
block">Kirim</button>
<?= form_close() ?>
</div>
</div>

<script type="text/javascript">
req = null;
function fotoFunction(value)
{
if (req != null) req.abort();

req = $.ajax({
type: "GET",
url : "<?php echo base_url();?>index.php/kunjungan/cek_foto",
data: {nama : value},
dataType: "json",
success: function(data){
if (data) {
document.getElementById("fotoWbp").src = "<?=base_url('uploads/fo
towbp')?>" + '/' +data.foto;
} else {
}
}
}

```

```

        document.getElementById("fotoWbp").src = "";
    }
}
});
};

function fotoClick(value)
{
    var x = document.getElementById("fotoWbp").src;
    window.open(x, '_blank');
}
</script>

```

5.7.3 Source Code CONTROLLER Formulir Kunjungan

```

function up_biodata()
{
    $logged_in = $this->session->userdata('logged_in');
    $level = $this->session->userdata('level');
    if (!empty($logged_in) && $level == 'peserta')
    {
        $config['upload_path'] = './uploads/kk/';
        $config['allowed_types'] = 'jpg|jpeg|png';
        $config['max_size'] = 12000;
        $config['max_width'] = 12000;
    }
}

```

```

$config['max_height'] = 12000;

$this->load->library('upload',$config);

$now = time();
if (strtotime($this->input->post('tanggal')) <= DATE('Y-m-d'))
{
    $id_peserta = $this->session->userdata('id_peserta');
    $this->session-
>set_flashdata('info','TANGGAL TIDAK VALID !');
    redirect('Kunjungan/update_biodata/'.$id_peserta);
}
if (empty($_FILES['photo']['name']))
{
    if ($this->input->post('old_foto') == "") {
        $id_peserta = $this->session->userdata('id_peserta');
        $this->session-
>set_flashdata('info','FOTO PENGUNJUNG WAJIB DI ISI !');
        redirect('Kunjungan/update_biodata/'.$id_peserta);
    } else {
        $data['dokumen'] = $this->input->post('old_foto');
    }
}
if (empty($_FILES['fotokk']['name']))

```

```

{
    if ($this->input->post('old_kk') == "") {
        $id_peserta = $this->session->userdata('id_peserta');
        $this->session-
>set_flashdata('info','FOTO KK WAJIB DI ISI !');
        redirect('Kunjungan/update_biodata/'.$id_peserta);
    } else {
        $data['kk'] = $this->input->post('old_kk');
    }
} else {
    $kk = $this->_do_upload('fotokk', $this->input->post('kk'));
    $data['kk'] = $kk;
}
if (empty($_FILES['fotowbppengunjung']['name']))
{
    $id_peserta = $this->session->userdata('id_peserta');
    $this->session-
>set_flashdata('info','FOTO WBP WAJIB DI ISI !');
    redirect('Kunjungan/update_biodata/'.$id_peserta);
}
if (empty($_FILES['fotopengikut1']['name'])) {
    $datap['fotopengikut1'] = "-";
}
else {
    $foto1 = $this->_do_upload('fotopengikut1');
    $datap['fotopengikut1'] = $foto1;
}
if (empty($_FILES['ktppengikut1']['name'])) {
    $datap['ktppengikut1'] = "-";
}

```

```

} else {
    $ktp1 = $this->_do_upload('ktppengikut1');
    $datap['ktppengikut1'] = $ktp1;
}

if (empty($_FILES['fotopengikut2']['name'])) {
    $datap['fotopengikut2'] = "-";
} else {
    $foto2 = $this->_do_upload('fotopengikut2');
    $datap['fotopengikut2'] = $foto2;
}

if (empty($_FILES['ktppengikut2']['name'])) {
    $datap['ktppengikut2'] = "-";
} else {
    $ktp2 = $this->_do_upload('ktppengikut2');
    $datap['ktppengikut2'] = $ktp2;
}

if ($this->cari_napi($this->input->post('wbpsertabin')) == 1) {
    if ($this->cari_pasal($this->input->post('pasal'), $this->input->post('wbpsertabin')) == 1) {
        if ($this->cari_blok($this->input->post('blok'), $this->input->post('wbpsertabin')) == 1) {
            $keterangan = 'Info napi yang diinputkan terdapat di database';
            ;
            $datap['status'] = "diverifikasi";
        } else {
            $keterangan = 'Ditolak, Blok NAMA Napi yang Anda inputkan tidak cocok, silahkan hubungi Admin';
            $datap['status'] = "ditolak";
        }
    }
}

```

```

$asdf = $this->Kunjungan_model->select_by_id($this->session->userdata('id_peserta'));

$pesan = '<font color="RED"><strong>-----
- PENDAFTARAN DITOLAK ----- </strong></font>
<font color="black"><br><br> No Registrasi : Nomor Antrian tidak didapatkan'.
'<br>No Identitas : '.$this->input->post('nik').
'<br>Nama Lengkap : '.$this->input->post('nama_lengkap').
'<br>Nama Wbp Yang Dikunjungi : '.$this->input->post('wbpertabin').

'<br><br></font><font color="RED"><strong>-----
- PENDAFTARAN DITOLAK ----- </strong></font>';
// Konfigurasi email
$config2 = [
    'mailtype' => 'html',
    'charset' => 'utf-8',
    'protocol' => 'smtp',
    'smtp_host' => 'smtp.gmail.com',
    'smtp_user' => 'internship1wildan@gmail.com', // Email
    'smtp_pass' => 'npm1164058', // Password gmail
    'smtp_crypto' => 'ssl',
    'smtp_port' => 465,
    'crlf' => "\r\n",
    'newline' => "\r\n"
];

```

```
// Load library email dan konfigurasinya
$this->load->library('email', $config2);

// Email dan nama pengirim
$this->email-
>from('internship1wildan@gmail.com', 'Rutan Klas 1 Bandung');

// Email penerima
$this->email->to($this->input-
>post('email'));// Ganti dengan email tujuan

// Subject email
$this->email-
>subject('Verifikasi Email Kunjungan Rutan Klas 1 Bandung');

// Isi email
$this->email->message($pesan);

// Tampilkan pesan sukses atau error
if ($this->email->send()) {
    echo 'Sukses! email berhasil dikirim.';
} else {
    echo 'Error! email tidak dapat dikirim.';
}

}

} else {
```

```

    $keterangan = 'Ditolak, blok dari Napi yang Anda inputkan tidak cocok, silahkan hubungi Admin';

    $datap['status'] = "ditolak";
    $asdf = $this->Kunjungan_model->select_by_id($this->session->userdata('id_peserta'));
    $pesan = '<font color="RED"><strong>-----  

- PENDAFTARAN DITOLAK ----- </strong></font>  

<font color="black"><br><br> No Registrasi : Nomor Antrian tidak didapatkan'.  

'<br>No Identitas : '.$this->input->post('nik').  

'<br>Nama Lengkap : '.$this->input->post('nama_lengkap').  

'<br>Nama Wbp Yang Dikunjungi : '.$this->input->post('wbpsertabin').  

'<br><br></font><font color="RED"><strong>-----  

- PENDAFTARAN DITOLAK ----- </strong></font>';  

    // Konfigurasi email  

    $config3 = [  

        'mailtype' => 'html',  

        'charset' => 'utf-8',  

        'protocol' => 'smtp',  

        'smtp_host' => 'smtp.gmail.com',  

        'smtp_user' => 'internship1wildan@gmail.com', // Email  

        'smtp_pass' => 'npm1164058', // Password gmail  

        'smtp_crypto' => 'ssl',  

        'smtp_port' => 465,  

        'crlf' => "\r\n",

```

```
'newline' => "\r\n"
];

// Load library email dan konfigurasinya
$this->load->library('email', $config3);

// Email dan nama pengirim
$this->email-
>from('internship1wildan@gmail.com', 'Rutan Klas 1 Bandung');

// Email penerima
$this->email->to($this->input-
>post('email'));// Ganti dengan email tujuan

// Subject email
$this->email-
>subject('Verifikasi Email Kunjungan Rutan Klas 1 Bandung');

// Isi email
$this->email->message($pesan);

// Tampilkan pesan sukses atau error
if ($this->email->send()) {
    echo 'Sukses! email berhasil dikirim.';
} else {
    echo 'Error! email tidak dapat dikirim.';
}
```

```

    }

} else {
    $keterangan = 'Ditolak, NAMA Napi yang Anda inputkan tidak cocok, silahkan hubungi Admin';
    $datap['status'] = "ditolak";
    $asdf = $this->Kunjungan_model->select_by_id($this->session->userdata('id_peserta'));
    $pesan = '<font color="RED"><strong>-----  

- PENDAFTARAN DITOLAK ----- </strong></font>  

<font color="black"><br><br> No Registrasi : Nomor Antrian tidak didapatkan'.  

'<br>No Identitas : '.$this->input->post('nik').  

'<br>Nama Lengkap : '.$this->input->post('nama_lengkap').  

'<br>Nama Wbp Yang Dikunjungi : '.$this->input->post('wbpsertabin').  

'  

'<br><br></font><font color="RED"><strong>-----  

- PENDAFTARAN DITOLAK ----- </strong></font>';
    // Konfigurasi email
    $config4 = [
        'mailtype' => 'html',
        'charset' => 'utf-8',
        'protocol' => 'smtp',
        'smtp_host' => 'smtp.gmail.com',
        'smtp_user' => 'internship1wildan@gmail.com', // Email
        'gmail
        'smtp_pass' => 'npm1164058', // Password gmail
    ]
}

```

```

'smtp_crypto' => 'ssl',
'smtp_port'   => 465,
'crlf'        => "\r\n",
'newline'     => "\r\n"

];

// Load library email dan konfigurasinya
$this->load->library('email', $config4);

// Email dan nama pengirim
$this->email-
>from('internship1wildan@gmail.com', 'Rutan Klas 1 Bandung');

// Email penerima
$this->email->to($this->input-
>post('email')); // Ganti dengan email tujuan

// Subject email
$this->email-
>subject('Verifikasi Email Kunjungan Rutan Klas 1 Bandung');

// Isi email
$this->email->message($pesan);

// Tampilkan pesan sukses atau error
if ($this->email->send()) {
    echo 'Sukses! email berhasil dikirim.';
} else {

```

```

        echo 'Error! email tidak dapat dikirim.';
    }

}

$data['id_peserta']=$this->input->post('id_peserta');
$data['nik']=$this->input->post('nik');
$data['nama_lengkap']=$this->input->post('nama_lengkap');
$data['nama_panggilan']=$this->input-
>post('nama_panggilan');

$data['jenis_kelamin']=$this->input->post('jenis_kelamin');
$data['alamat']=$this->input->post('alamat');
$data['email']=$this->input->post('email');
$data['status_hubungan']=$this->input-
>post('status_hubungan');

$datap['id_peserta']=$this->input->post('id_peserta');
$datap['id_jadwal']=$this->input->post('id_jadwal');
$datap['tanggal_kunjungan']=$this->input->post('tanggal');
$laki_laki = $this->input->post('laki_laki');
if ($laki_laki == "") {
    $datap['laki_laki']= "-";
} else{
    $datap['laki_laki']= $laki_laki;
}
$datap['perempuan']=$this->input->post('perempuan');

```

```

$datap['anak_anak']=$this->input->post('anak_anak');

$jumlah = $this->input->post('jumlah');
if ($jumlah == "") {
    $datap['jumlah'] = "-";
} else {
    $datap['jumlah'] = $jumlah;
}

// $datap['jumlah']=$this->input->post('jumlah');
$datap['jeniskelamin2']=$this->input->post('jeniskelamin2');
$datap['noktp2']=$this->input->post('noktp2');
$datap['wbpsertabin']=$this->input->post('wbpsertabin');
$datap['pasal']=$this->input->post('pasal');
$datap['blok']=$this->input->post('blok');
$datap['keterangan'] = $keterangan;
$foto = $this->_do_upload('fotowbppengunjung');
$datap['fotowbppengunjung'] = $foto;

$tanggal_today=date('d');
$bulan_today=date('m');
$tahun_today=date('Y');

if ($datap['status'] == 'diverifikasi') {
    $jadwal = $this->Kunjungan_model-
>get_jadwal_by_id($datap['id_jadwal']);
    $kuota = $jadwal->kuota;
    $jumlah_pengunjung = $this->Kunjungan_model-
>get_jumlah_jadwal($datap['tanggal_kunjungan'], $datap['id_jadwal']);
}

```

```

if ($jumlah_pengunjung < $kuota) {
    $nomor = $jumlah_pengunjung + 1;
    $datap['kode_antrian'] = date('Ymd', strtotime($datap['tanggal_kunjungan'])).$datap['id_jadwal'].$nomor;
    $asdf = $this->Kunjungan_model->select_by_id($this->session->userdata('id_peserta'));
    $pesan = '<font color="green"><strong>-----';
    - PENDAFTARAN DITERIMA ----- </strong></font>
    <font color="black"><br><br> No Registrasi : '.$datap['kode_antrian'].'
    '<br>No Identitas : '.$this->input->post('nik').
    '<br>Nama Lengkap : '.$this->input->post('nama_lengkap').
    '<br>Nama Wbp Yang Dikunjungi : '.$this->input->post('wbpsertabin').
    '<br><br></font><font color="green"><strong>-----';
    --- PENDAFTARAN DITERIMA -----
    - </strong></font><br><strong> TUNJUKAN BUKTI EMAIL INI KEPADA PETUGAS KUNJUNGAN</strong>';
    // Konfigurasi email
    $config1 = [
        'mailtype' => 'html',
        'charset' => 'utf-8',
        'protocol' => 'smtp',
        'smtp_host' => 'smtp.gmail.com',
        'smtp_user' => 'internship1wildan@gmail.com', // Email g
    mail

```

```

'smtp_pass' => 'npm1164058', // Password gmail
'smtp_crypto' => 'ssl',
'smtp_port' => 465,
'crlf' => "\r\n",
'newline' => "\r\n"
];

// Load library email dan konfigurasinya
$this->load->library('email', $config1);

// Email dan nama pengirim
$this->email-
>from('internship1wildan@gmail.com', 'Rutan Klas 1 Bandung');

// Email penerima
$this->email->to($this->input-
>post('email')); // Ganti dengan email tujuan

// Subject email
$this->email-
>subject('Verifikasi Email Kunjungan Rutan Klas 1 Bandung');

// Isi email
$this->email->message($pesan);

// Tampilkan pesan sukses atau error
if ($this->email->send()) {
    echo 'Sukses! email berhasil dikirim.';
}

```

```

} else {
    echo 'Error! email tidak dapat dikirim.';
}
} else {
    $data2['peserta'] = $this->Kunjungan_model->select_by_id($data['id_peserta']);
    $data2['jadwal'] = $this->admin_model->get_jadwal();
    $datap['kode_antrian'] = 'Nomor Antrian tidak didapatkan karena jadwal penuh';
    $asdf = $this->Kunjungan_model->select_by_id($this->session->userdata('id_peserta'));
    $pesan = '<font color="red"><strong>-----  

- PENDAFTARAN DITOLAK ----- </strong></font>  

<font color="black"><br><br> No Registrasi : '. $datap['kode_antrian'].  

'<br>No Identitas : '.$this->input->post('nik').  

'<br>Nama Lengkap : '.$this->input->post('nama_lengkap').  

'<br>Nama Wbp Yang Dikunjungi : '.$this->input->post('wbpsertabin').  

'  

'<br><br></font><font color="red"><strong>-----  

- PENDAFTARAN DITOLAK -----  

- </strong></font><br><strong> TUNJUKAN BUKTI EMAIL INI KEPADA PETUGAS KUNJUNGAN</strong>';
// Konfigurasi email
$config1 = [
    'mailtype' => 'html',
    'charset' => 'utf-8',

```

```

'protocol' => 'smtp',
'smtp_host' => 'smtp.gmail.com',
'smtp_user' => 'internship1wildan@gmail.com', // Email g
mail

'smtp_pass' => 'npm1164058', // Password gmail
'smtp_crypto' => 'ssl',
'smtp_port' => 465,
'crlf' => "\r\n",
'newline' => "\r\n"

];

// Load library email dan konfigurasinya
$this->load->library('email', $config1);

// Email dan nama pengirim
$this->email-
>from('internship1wildan@gmail.com', 'Rutan Klas 1 Bandung');

// Email penerima
$this->email->to($this->input-
>post('email')); // Ganti dengan email tujuan

// Subject email
$this->email-
>subject('Verifikasi Email Kunjungan Rutan Klas 1 Bandung');

// Isi email
$this->email->message($pesan);

```

```

// Tampilkan pesan sukses atau error
if ($this->email->send()) {
    echo 'Sukses! email berhasil dikirim.';
} else {
    echo 'Error! email tidak dapat dikirim.';
}
$this->session-
>set_flashdata('info','kuota untuk jadwal yang dipilih sudah penuh, silahkan
pilih jadwal lain');
$this->template-
>dashboard_user('Kunjungan/dashboard/form_biodata',$data2);
}

} else {
# code...
}

$this->Kunjungan_model-
>update_biodata($data, $data['id_peserta']);
$this->Kunjungan_model->insert_pengunjung($datap);

$data2['peserta'] = $this->Kunjungan_model-
>select_by_id($data['id_peserta']);
$data2['jadwal'] = $this->admin_model->get_jadwal();
$this->session-
>set_flashdata('info','Dokumen telah berhasil diupdate');

```

```

    $this->template-
>dashboard_user('Kunjungan/dashboard/form_biodata',$data2);
}

else
{
    $this->template->Kunjungan('Kunjungan/content');
}
}

private function _do_upload($value)
{
    $config['upload_path']      = './uploads/kk/';
    $config['allowed_types']    = 'gif|jpg|png';
    $config['max_size']         = 1000;
    $config['max_widht']        = 1000;
    $config['max_height']       = 1000;
    $config['file_name']        = round(microtime(true)*1000);

    $this->load->library('upload', $config);
    if (!$this->upload->do_upload($value)) {
        $id_peserta = $this->session->userdata('id_peserta');
        $this->session-
>set_flashdata('info','FOTO PENGUNJUNG WAJIB DI ISI !');
        redirect('Kunjungan/update_biodata/'.$id_peserta);
    }
    return $this->upload->data('file_name');
}

```

5.8 SEND NOTIFIKASI KUNJUNGAN VIA EMAIL



Gambar

5.8.1 Source Code MODEL Notifikasi Email

```
function select_by_id($id_peserta)
{
    $this->db->select('*');
    $this->db->from('user');
    $this->db->where('id_peserta',$id_peserta);

    return $this->db->get()->row();
```

5.8.2 Source Code VIEW Notifikasi Email

```
<table class="table table-bordered" id="table1">
<thead>
<tr>
    <th colspan="8" id="header2"><center><b>History Pendaftaran</b></center></th>
</tr>
<tr>
```

```

<th>Tanggal Pendaftaran</th>
<th>Nomor Antrian</th>
<th>Jam Kunjungan</th>
<th>Status</th>
<th>Keterangan</th>
<th>Nama Napi</th>
<th>Pasal</th>
<th>Blok</th>
<th>Foto</th>
</tr>
</thead>
<tbody>
<?php foreach ($history as $data) { ?>
<tr>
<td><?=$data->created?></td>
<td><?=$data->kode_antrian?></td>
<td><?=$data->dari?> - <?=$data->sampai?></td>
<td><?=$data->status?></td>
<td><?=$data->keterangan?></td>
<td><?=$data->wbpertabin?></td>
<td><?=$data->pasal?></td>
<td><?=$data->blok?></td>
<td> <a href="<?php echo base_url(); ?>uploads/kk/<?php echo
$data-
>photowbppengunjung; ?>" target="_blank"></td>
</tr>

```

```

<?php } ?>
</tbody>
</table>

```

5.8.3 Source Code CONTROLLER Notifikasi Email

```

$keterangan = 'Ditolak, Blok NAMA Napi yang Anda inputkan tidak cocok, silahkan hubungi Admin';

$datap['status'] = "ditolak";

$asdf = $this->Kunjungan_model->select_by_id($this->session->userdata('id_peserta'));

$pesan = '<font color="RED"><strong>-----  

- PENDAFTARAN DITOLAK ----- </strong></font>  

<font color="black"><br><br> No Registrasi : Nomor Antrian tidak didapatkan'.  

'<br>No Identitas : '.$this->input->post('nik').  

'<br>Nama Lengkap : '.$this->input->post('nama_lengkap').  

'<br>Nama Wbp Yang Dikunjungi : '.$this->input->post('wbpertabin').  

'<br><br></font><font color="RED"><strong>-----  

- PENDAFTARAN DITOLAK ----- </strong></font>';  

// Konfigurasi email  

$config2 = [  

    'mailtype' => 'html',  

    'charset' => 'utf-8',  

    'protocol' => 'smtp',  

    'smtp_host' => 'smtp.gmail.com',

```

```

'smtp_user' => 'internship1wildan@gmail.com', // Email
gmail

'smtp_pass' => 'npm1164058', // Password gmail
'smtp_crypto' => 'ssl',
'smtp_port' => 465,
'crlf' => "\r\n",
'newline' => "\r\n"

];

// Load library email dan konfigurasinya
$this->load->library('email', $config2);

// Email dan nama pengirim
$this->email-
>from('internship1wildan@gmail.com', 'Rutan Klas 1 Bandung');

// Email penerima
$this->email->to($this->input-
>post('email')); // Ganti dengan email tujuan

// Subject email
$this->email-
>subject('Verifikasi Email Kunjungan Rutan Klas 1 Bandung');

// Isi email
$this->email->message($pesan);

// Tampilkan pesan sukses atau error

```

```

if ($this->email->send()) {
    echo 'Sukses! email berhasil dikirim.';
} else {
    echo 'Error! email tidak dapat dikirim.';
}

```

5.9 HISTORY KUNJUNGAN

History Pendaftaran								
Tanggal Pendaftaran	Nomor Antrian	Jam Kunjungan	Status	Keterangan	Nama Napi	Pasal	Blok	Foto
2020-01-17 20:58:36	2020011811	09:00:00 - 10:00:00	diverifikasi	Info napi yang diinputkan terdapat di database	JAMAL	12	24	

5.9.1 Source Code MODEL History Kunjungan

```

function get_history_kunjungan($id_peserta)
{
    $this->db->select('*');
    $this->db->from('pengunjung');
    $this->db->join('jadwal','jadwal.id_jadwal = pengunjung.id_jadwal');
    $this->db->where('pengunjung.id_peserta',$id_peserta);
    $this->db->order_by('pengunjung.created','DESC');

    return $this->db->get()->result();
}

```

5.9.2 Source Code VIEW History Kunjungan

```
<table class="table table-bordered" id="table1">

    <thead>
        <tr>
            <th colspan="8" id="header2"><center><b>History Pendaftaran</b></center></th>
        </tr>
        <tr>
            <th>Tanggal Pendaftaran</th>
            <th>Nomor Antrian</th>
            <th>Jam Kunjungan</th>
            <th>Status</th>
            <th>Keterangan</th>
            <th>Nama Napi</th>
            <th>Pasal</th>
            <th>Blok</th>
            <th>Foto</th>
        </tr>
    </thead>
    <tbody>
        <?php foreach ($history as $data) { ?>
            <tr>
                <td><?=$data->created?></td>
                <td><?=$data->kode_antrian?></td>
                <td><?=$data->dari?> - <?=$data->sampai?></td>
                <td><?=$data->status?></td>
                <td><?=$data->keterangan?></td>
```

```

<td><?=$data->wbpsertabin?></td>
<td><?=$data->pasal?></td>
<td><?=$data->blok?></td>
<td> <a href="php echo base_url(); ?&gt;uploads/kk/&lt;?php echo
$data-
&gt;fotowbappengunjung; ?&gt;" target="_blank"&gt;&lt;img src="&lt;?php echo base_url(
); ?&gt;uploads/kk/&lt;?php echo $data-
&gt;fotowbappengunjung; ?&gt;" style="width: 120px"&gt;&lt;/td&gt;
&lt;/tr&gt;
&lt;?php } ?&gt;
&lt;/tbody&gt;
&lt;/table&gt;
</pre

```

5.9.3 Source Code CONTROLLER History Kunjungan

```

$this->session->set_userdata($data_session);
$data['history'] = $this->Kunjungan_model-
>get_history_kunjungan($akun->id_peserta);
$this->template-
>dashboard_user('Kunjungan/dashboard/home', $data);

```

5.10 PESAN

CONTACT

Kontak kami dan kami akan menjawab dalam
24 jam

📍 JL.

📞 022

✉️ rutanbdgjuara@gmail.com

WILDAN KHAUSTARA
 NPM 1164058
D4 TEKNIK INFORMATIKA 4B
POLITEKNIK POS INDONESIA

5.10.1 Source Code MODEL Pesan

```
function select_by_id($id_peserta)
{
    $this->db->select('*');
    $this->db->from('user');
    $this->db->where('id_peserta',$id_peserta);

    return $this->db->get()->row();
```

5.10.2 Source Code VIEW Pesan

```
<hr>

<div id="kontak" class="container-fluid">
    <h2 class="text-center">CONTACT</h2>
    <div class="row">
        <div class="col-sm-5">
            <p>Kontak kami dan kami akan menjawab dalam 24 jam</p>
            <p><span class="glyphicon glyphicon-map-marker"></span> JL.</p>
            <p><span class="glyphicon glyphicon-phone"></span> 022</p>
            <p><span class="glyphicon glyphicon-envelope"></span> rutanbdgjuara@gmail.com</p>
        </div>
        <div class="col-sm-7">
            <?php
                $info = $this->session->flashdata('info');
                if (!empty($info))
                {
```

```

echo $info;
}
?>
<form role="form" action="=php echo site_url('pengunjung_pesan/kirim_pesanan');?&gt;" method="post"&gt;
&lt;div class="row"&gt;
&lt;div class="col-sm-12 form-group"&gt;
&lt;input type="hidden" required name="id_peserta" value="<?=session-&gt;userdata('id_peserta')?&gt;"&gt;
&lt;input type="text" required name="subject" class="form-control" placeholder="Subject"&gt;&lt;br&gt;
&lt;input type="text" required name="no_hp" class="form-control" placeholder="Nomor HP"&gt;&lt;br&gt;
&lt;textarea class="form-control" required id="isi" name="isi" placeholder="Isi Pesan" rows="5"&gt;&lt;/t extarea&gt;&lt;br&gt;
&lt;button class="btn btn-default pull-right" type="submit"&gt;Send&lt;/button&gt;
&lt;/div&gt;
&lt;/div&gt;
&lt;/form&gt;
&lt;/div&gt;
</pre

```

5.10.3 Source Code CONTROLLER Pesan

```

function kirim_pesanan() {
    $id_peserta = $this->input->post('id_peserta');
    $data = $this->Kunjungan_model->get_by_id($id_peserta);
    if ($data->nama_lengkap == "") {

```

```

    $this->session-
>set_flashdata('info','Nama Lengkap tidak ada, Anda Harus Mengisi Form P
endaftaran Terlebih dahulu !);
    redirect('Kunjungan');
} else if ($data->email == "") {
    $this->session-
>set_flashdata('info','Email tidak ada, Anda Harus Mengisi Form Pendaftara
n Terlebih dahulu !);
    redirect('Kunjungan');
} else {
    $data = array(
        'id_peserta' => $id_peserta,
        'isi' => $this->input->post('isi'),
        'subject' => $this->input->post('subject'),
        'no_hp' => $this->input->post('no_hp'),
        'nama_lengkap' => $data->nama_lengkap,
        'tujuan' => 1,
        'email' => $data->email
    );
}

$insert = $this->Kunjungan_model->insert_pesan($data);
$this->session-
>set_flashdata('info','Pesan Terkirim, tunggu balasan dari Admin');
redirect('Kunjungan');

}
}

```

5.11 DAFTAR PENGUNJUNG

NIK	USERNAME	Nama Lengkap	Jenis Kelamin	Status Hubungan	Email	Foto	download foto	Tambah Peserta
213213	DWIKY	DWIKY DD	LAKI-LAKI	keluarga - Suami	wkhaustara@gmail.com		Download Foto	Selengkapnya Edit Hapus Data Kunjungan
336555	wildan	WILDAN	laki-laki	keluarga - Suami	wkhaustara@gmail.com		Download Foto	Selengkapnya Edit Hapus Data Kunjungan

5.11.1 Source Code MODEL Daftar Pengunjung

```
function daftarpeserta()
{
    return $this->db->get_where('user', array('level' => 'peserta'))->result();
}
```

5.11.2 Source Code VIEW Daftar Pengunjung

```
<div class="container">
    <h3 class="header smaller lighter blue">Daftar Keseluruhan Pengunjung :</h3>
    <?php echo $jumlah;?>
    <table class="table table-striped table-bordered data">
        <thead>
            <tr>
                <th>NIK</th>
                <th>USERNAME</th>
                <th>Nama Lengkap</th>
                <th>Jenis Kelamin</th>
                <th>Status Hubungan</th>
                <th>Email</th>
            </tr>
        </thead>
        <tbody>
            <tr>
                <td>213213</td>
                <td>DWIKY</td>
                <td>DWIKY DD</td>
                <td>LAKI-LAKI</td>
                <td>keluarga - Suami</td>
                <td>wkhaustara@gmail.com</td>
            </tr>
            <tr>
                <td>336555</td>
                <td>wildan</td>
                <td>WILDAN</td>
                <td>laki-laki</td>
                <td>keluarga - Suami</td>
                <td>wkhaustara@gmail.com</td>
            </tr>
        </tbody>
    </table>
</div>
```

```

<th>Foto</th>
<th>download foto</th>
<th align="center"><a href="php echo site_url('admin/tambahpeserta');?&gt;"&gt;Tambah Peserta&lt;/a&gt;&lt;/th&gt;

&lt;/tr&gt;
&lt;/thead&gt;
&lt;tbody&gt;
&lt;?php foreach ($peserta as $peserta)
{
?
&lt;tr&gt;
&lt;td&gt;&lt;?php echo $peserta-&gt;nik;?&gt;&lt;/td&gt;
&lt;td&gt;&lt;?php echo $peserta-&gt;username;?&gt;&lt;/td&gt;
&lt;td&gt;&lt;?php echo $peserta-&gt;nama_lengkap;?&gt;&lt;/td&gt;
&lt;td&gt;&lt;?php echo $peserta-&gt;jenis_kelamin;?&gt;&lt;/td&gt;
&lt;td&gt;&lt;?php echo $peserta-&gt;status_hubungan;?&gt;&lt;/td&gt;
&lt;td&gt;&lt;?php echo $peserta-&gt;email;?&gt;&lt;/td&gt;
&lt;td class="text-center"&gt;
&lt;a href="<?php echo base_url(); ?&gt;uploads/kk/&lt;?php echo $peserta-&gt;dokumen; ?&gt;" target="_blank"&gt;&lt;img src="<?php echo base_url(); ?&gt;uploads/kk/&lt;?php echo $peserta-&gt;dokumen; ?&gt;" style="width: 120px"&gt;
&lt;/a&gt;
&lt;/td&gt;
&lt;td&gt;&lt;a href="<?php echo base_url(); ?&gt;uploads/kk/&lt;?php echo $peserta-&gt;dokumen; ?&gt;" download&gt;Download Foto&lt;/a&gt;&lt;/td&gt;
&lt;td&gt;&lt;a href="<?php echo site_url('admin/biodatapeserta/'.$peserta-&gt;id_peserta);?&gt;"&gt;Selengkapnya | &lt;/a&gt;
</pre

```

```

        <a href="<?php echo site_url('admin/editpeserta/'.$peserta->id_peserta);?>">Edit | </a>
        <a href="<?php echo site_url('admin/hapus_peserta/'.$peserta->id_peserta);?>">Hapus |</a>
        <a href="<?php echo site_url('admin/data_kunjungan/'.$peserta->id_peserta);?>">Data Kunjungan</a>
    </td>

</tr>
<?php
}
?>
</tbody>
</table>
</div>

```

5.11.3 Source Code CONTROLLER Daftar Pengunjung

```

function daftarpeserta()
{
    $logged_in = $this->session->userdata('logged_in');
    $level = $this->session->userdata('level');
    if (!empty($logged_in) && $level=='admin')
    {
        $peserta = $this->admin_model->daftarpeserta();
        $data['jumlah'] = count($peserta);

        $data['peserta'] = $this->admin_model->daftarpeserta();
        $this->template->admin('admin/daftarpeserta',$data);
    }
}

```

```

        }
    else
    {
        $this->template->home('home/content');
    }
}

```

5.12 EDIT DATA PENGUNJUNG

Admin Layanan Kunjungan

Home > Other Pages > Blank Page

Edit Data Pengunjung

Username	DWIKY
Password	*****
NIK	213213
Nama Lengkap	DWIKY DD
Nama Panggilan	DWIKY
Jenis Kelamin	LAKI-LAKI
Alamat	ASDASDASDASDASDAS
Status Hubungan	keluarga - Suami
Email	wkhaustara@gmail.com
Upload Foto Pengunjung	<input type="button" value="Browse..."/>

5.12.1 Source Code MODEL Edit Pengunjung

```

function editpeserta($data, $id_peserta)
{
    $this->db->where('id_peserta',$id_peserta);
    $this->db->update('user',$data);
}

```

5.12.2 Source Code VIEW Edit Pengunjung

```
<h1>Edit Data Pengunjung</h1>
<div class="progress" data-percent="">
    <div class="bar" style="width:100%;"></div>
</div>

<?php
$info = $this->session->flashdata('info');
if (!empty($info))
{
    echo $info;
}
?>

<form class="form-
horizontal" action=<?php echo site_url('admin/proseseditpeserta');?>" meth
od="post" enctype="multipart/form-data">
    <div class="control-group">
        <label class="control-label" for="form-field-2">Username</label>
        <div class="controls">
            <input type="text" placeholder="Masukkan Username" name="usern
ame" value=<?=$peserta->username?>" required />
        </div>
    </div>
    <div class="control-group">
        <label class="control-label" for="nama_lengkap">Password</label>
        <div class="controls">
```

```
<input type="Password" placeholder="Masukkan Password" name="password" value="<?=$peserta->password?>" required />
</div>
</div>
<div class="control-group">
    <label class="control-label" for="nama_lengkap">NIK</label>
    <div class="controls">
        <input type="text" placeholder="Masukkan Nik" name="nik" value="<?=$peserta->nik?>" required />
    </div>
</div>
<div class="control-group">
    <label class="control-label" for="nama_lengkap">Nama Lengkap</label>
    <div class="controls">
        <input type="text" placeholder="Masukkan Nama Lengkap" name="nama_lengkap" value="<?=$peserta->nama_lengkap?>" required />
    </div>
</div>
<div class="control-group">
    <label class="control-label" for="nama_lengkap">Nama Panggilan</label>
    <div class="controls">
        <input type="text" placeholder="Masukkan Nama Panggilan" name="nama_panggilan" value="<?=$peserta->nama_panggilan?>" required />
    </div>

```

```
</div>
<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">Jenis Kelamin</label>
    <div class="controls">
        <select class="form-control" name="jenis_kelamin">
            <option><?=$peserta->jenis_kelamin?></option>
            <option>laki-laki</option>
            <option>Perempuan</option>
        </select>
    </div>
</div>
<br>
<div class="control-group">
    <label class="control-label" for="nama_lengkap">Alamat</label>
    <div class="controls">
        <textarea name="alamat" rows="3"><?=$peserta-
>alamat?></textarea>
    </div>
</div>
<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-
label">Status Hubungan</label>
    <div class="controls">
        <select class="form-control" name="status_hubungan">
            <option><?=$peserta->status_hubungan?></option>
            <option>keluarga - Suami</option>
            <option>Keluarga - Istri</option>
        </select>
    </div>
</div>
```

```

<option>Keluarga - Ayah</option>
<option>Keluarga - Ibu</option>
<option>Keluarga - Adik</option>
<option>Keluarga - Saudara jauh</option>
<option>Teman</option>
</select>
</div>
</div>
<br>
<div class="control-group">
    <label class="control-label" for="nama_lengkap">Email</label>
    <div class="controls">
        <input type="text" placeholder="Masukkan Email" name="email" value="<?=$peserta->email?>" required />
    </div>
</div>
<div class="form-group">
    <label for="inputPassword3" class="col-sm-2 control-label">Upload Foto Pengunjung</label>
    <div class="controls">
        <input type="file" name="userfile">
    </div>
</div>
<br>
<input type="hidden" name="old_foto" value="<?=$peserta->dokumen?>">
<input type="hidden" name="id_peserta" value="<?=$peserta->id_peserta?>">

```

```
<button type="submit" class="btn btn-primary btn-md btn-block">Kirim</button>
</form>
```

5.12.3 Source Code CONTROLLER Edit Pengunjung

```
function proseseditpeserta() {
    $config['upload_path'] = './uploads/kk/';
    $config['allowed_types'] = 'jpg|jpeg|png';
    $config['max_size'] = 12000;
    $config['max_width'] = 12000;
    $config['max_height'] = 12000;
    $this->load->library('upload',$config);

    $id_peserta = $this->input->post('id_peserta');
    $data = array(
        'username' => $this->input->post('username'),
        'password' => $this->input->post('password'),
        'nik' => $this->input->post('nik'),
        'nama_lengkap' => $this->input->post('nama_lengkap'),
        'nama_panggilan' => $this->input->post('nama_panggilan'),
        'jenis_kelamin' => $this->input->post('jenis_kelamin'),
        'alamat' => $this->input->post('alamat'),
        'status_hubungan' => $this->input->post('status_hubungan'),
        'email' => $this->input->post('email')
    );

    if (!$this->upload->do_upload())
    {
```

```

        $data['dokumen'] = $this->input->post('old_foto');

    }

else

{

    $gambar = $this->upload->data();

    $data['dokumen'] = $gambar['file_name'];

}

$insert = $this->admin_model->editpeserta($data, $id_peserta);

$this->session->set_flashdata('info','Data Pengunjung Tersunting');

redirect('admin/editpeserta/'.$id_peserta);

}

```

5.13 DAFTAR WBP

No WBP	Nama lengkap serta bin	pasal	blok kamar	foto	Tambah WBP
5	ASEP BIN TATANG	12	24		Selengkapnya Edit Hapus Data History
6	JAMAL	12	24		Selengkapnya Edit Hapus Data History

5.13.1 Source Code MODEL Daftar Data WBP

```

function daftarwbp()
{
    return $this->db->get_where('datawbp')->result();
}

```

5.13.2 Source Code VIEW Daftar Data WBP

```
<div class="container">

    <h3 class="header smaller lighter blue">Daftar Keseluruhan WBP : <?php
echo $jumlah;?></h3>

    <table class="table table-striped table-bordered data">

        <thead>
            <tr>
                <th>No WBP </th>
                <th>Nama lengkap serta bin </th>
                <th>pasal</th>
                <th>blok kamar</th>
                <th> foto</th>
                <th align="center"><a href="<?php echo site_url('Admin_kelola_
wbp/tambahdatawbp');?>">Tambah WBP</a></th>
            </tr>
        </thead>
        <tbody>
            <?php foreach ($datawbp as $datawbp)
{
    ?>
            <tr>
                <td><?php echo $datawbp->id_wbp;?></td>
                <td><?php echo $datawbp->nama_dan_bin;?></td>
                <td><?php echo $datawbp->pasal;?></td>
                <td><?php echo $datawbp->blok_kamar;?></td>
                <td class="text-center">
                    <a href="<?php echo base_url(); ?>uploads/fotowbp/<?php ec
ho $datawbp-
```

```

>foto; ?>" target="_blank">
    </a>
</td>
<td><a href="<?php echo site_url('admin/biodatadatawbp/'.$dat
awbp->nama_dan_bin.'/'.$datawbp->pasal.'/'.$datawbp-
>blok_kamar);?>">Selengkapnya | </a>
    <a href="<?php echo site_url('Admin_kelola_wbp/editdatawb
p/'.$datawbp->id_wbp);?>">Edit | </a>
    <a href="<?php echo site_url('Admin_kelola_wbp/hapus_data
wbp.'/'.$datawbp->id_wbp);?>">Hapus |</a>
    <a href="<?php echo site_url('Admin_kelola_wbp/data_peng
unjung.'/'.$datawbp->nama_dan_bin.'/'.$datawbp->pasal.'/'.$datawbp-
>blok_kamar);?>">Data History</a>
</td>

</tr>
<?php
}
?>
</tbody>
</table>
</div>

```

5.13.3 Source Code CONTROLLER Daftar Data WBP

```

function daftarwbp()
{
    $logged_in = $this->session->userdata('logged_in');

```

```
$level = $this->session->userdata('level');
if (!empty($logged_in) && $level=='admin')
{
    $peserta = $this->admin_model->daftarwbp();
    $data['jumlah'] = count($peserta);

    $data['datawbp'] = $this->admin_model->daftarwbp();
    $this->template->admin('admin/datawbp',$data);
}
else
{
    $this->template->home('home/content');
}
}
```

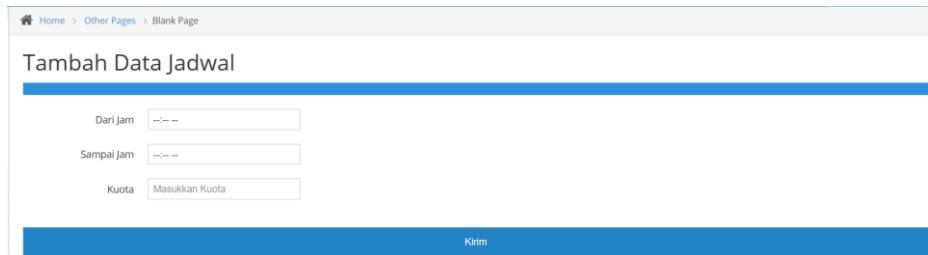
5.14 EDIT DATA WBP

5.14.1 Source Code MODEL Edit Data WBP

5.14.2 Source Code VIEW Edit Data WBP

5.14.3 Source Code CONTROLLER Edit Data WBP

5.15 TAMBAH DATA JADWAL



The screenshot shows a web page with a header navigation bar. The main content area is titled "Tambah Data Jadwal". It contains three input fields: "Dari Jam" (From Time), "Sampai Jam" (Until Time), and "Kuota" (Quota). Below these fields is a button labeled "Masukkan Kuota". At the bottom of the form is a large blue "Kirim" (Send) button.

5.15.1 Source Code MODEL Tambah Jadwal

```
function tambahjadwal($data)
{
    $this->db->insert('jadwal',$data);
}
```

5.15.2 Source Code VIEW Tambah Jadwal

```
<h1>Tambah Data Jadwal</h1>
<div class="progress" data-percent="">
    <div class="bar" style="width:100%;"></div>
</div>

<?php
```

```
$info = $this->session->flashdata('info');
if (!empty($info))
{
    echo $info;
}
?>

<form class="form-
horizontal" action="php echo site_url('Admin_kelola_jadwal/prosesstambahjadwal');?&gt;" method="post" enctype="multipart/form-data"&gt;
    &lt;div class="control-group"&gt;
        &lt;label class="control-label" for="form-field-2"&gt;Dari Jam&lt;/label&gt;
        &lt;div class="controls"&gt;
            &lt;input type="time" placeholder="Masukkan Jam" name="dari" required /&gt;
        &lt;/div&gt;
    &lt;/div&gt;
    &lt;div class="control-group"&gt;
        &lt;label class="control-label" for="nama_lengkap"&gt;Sampai Jam&lt;/label&gt;
        &lt;div class="controls"&gt;
            &lt;input type="time" placeholder="Masukkan Jam" name="sampai" required /&gt;
        &lt;/div&gt;
    &lt;/div&gt;
    &lt;div class="control-group"&gt;
        &lt;label class="control-label" for="nama_lengkap"&gt;Kuota&lt;/label&gt;
        &lt;div class="controls"&gt;</pre
```

```

<input type="number" placeholder="Masukkan Kuota" name="kuota"
" required />
</div>
</div>
<br>
<button type="submit" class="btn btn-primary btn-md btn-
block">Kirim</button>
</form>

```

5.15.3 Source Code CONTROLLER Tambah Data Jadwal

```

function prosesambahjadwal()
{
    $data = $this->input->post();
    $insert = $this->admin_model->tambahjadwal($data);
    $this->session->set_flashdata('info','Data Jadwal Tersimpan');
    redirect('Admin_kelola_jadwal/tambahdatajadwal');

}

function editdatajadwal() {
    $logged_in = $this->session->userdata('logged_in');
    $level = $this->session->userdata('level');
    if (!empty($logged_in) && $level=='admin')
    {
        $id_jadwal = $this->uri->segment(3);
        $data['datajadwal'] = $this->admin_model-
>get_jadwal_by_id($id_jadwal);
        $this->template->admin('admin/editdatajadwal', $data);
    }
}

```

```

else
{
    $this->template->home('home/content');
}
}

```

5.16 PENCARIAN WBP (*BOYER MOORE*)

The screenshot displays a web-based administration system for managing visitors. On the left, there's a sidebar with various menu items: Daftar Pengunjung, Daftar WBP, Data Jadwal, Verifikasi, Cari Napi, Cari Kode Antrian, Cari Tanggal, and Pesan. The main content area has a header 'Admin Layanan Kunjungan'. Below the header, the URL 'Home > Other Pages > Blank Page' is shown. The main part of the page features a search form for 'Cari Narapidana'. It includes a text input field labeled 'Nama' with the value 'JAMAL', a 'Submit' button, and a link 'List Pencarian Nama' which leads to a table showing search results for 'JAMAL'. Another table below is titled 'Data Narapidana' and lists details for a person named 'JAMAL'.

Gambar?

5.16.1 Source Code MODEL Pencarian WBP

```

function get_tahanan() {
    $this->db->select('*');
    $this->db->from('datawbp');

    return $this->db->get()->result();
}

```

5.16.2 Source Code VIEW Pencarian WBP

```

<div class="col-xs-6 col-sm-6 widget-container-col" id="widget-container-
col-4">

```

```
<div class="widget-box" id="widget-box-4">
    <div class="widget-header widget-header-large">
        <h4 class="widget-title">Cari Narapidana</h4>
    </div>

    <div class="widget-body">
        <div class="widget-main">
            <center>
                <div class="form-group">
                    <label for="psw"><span class="glyphicon glyphicon-user"></span> Nama</label>
                    <input type="text" class="form-control" id="nama" name="nama" placeholder="Masukkan Nama Pendaftar Kunjungan">
                </div>
                <button id="search" onclick="getInputValue();" class="btn btn-info" type="button">
                    <i class="ace-icon fa fa-check bigger-110"></i>
                    Submit
                </button>
            </center>
        </div>
    </div>
</div>
</div>
</div>
<hr>
```

```

<h4 id="header4" style="display: none;"><center>Nama Narapidana Tidak
Ditemukan</center></h4>
<table class="table table-bordered" id="table" style="display: none;">
<thead>
    <tr>
        <th colspan="2" id="header2"><center><b>List Pencarian Nama</b>
</center></th>
    </tr>
</thead>
<tbody class="nama_pendaftar">
    <tr>
        <td id="nama_lengkapi">kakaka</td>
    </tr>
</tbody>
</table>
<table class="table table-bordered" id="table1" style="display: none;">
<thead>
    <tr>
        <th colspan="2" id="header2"><center><b>Data Narapidana</b></c
enter></th>
    </tr>
    <tr>
        <th>Nama</th>
        <td id="nama_napi"></td>
    </tr>
    <tr>
        <th>Pasal</th>
        <td id="pasal"></td>
    </tr>

```

```

        </tr>
        <th>Blok Kamar</th>
        <td id="blok"></td>

    </tr>
</thead>
</table>
</div>
<script type="text/javascript">
function getInputValue(){
    var nama = document.getElementById("nama").value;
    $.ajax({
        url : "<?php echo base_url();?>index.php/admin_pencarian/cek_napi"
        ,
        method : "GET",
        data : {nama: nama},
        dataType : 'json',
        success: function(data){
            if (data) {
                $("#table").show();
                $("#table1").hide();
                $("#header4").hide();
                var html = "";
                var i;
                for(i=0; i<data.length; i++){
                    html += '<tr><td id="nama_lengkap"><center><a href="#" o
nclick="getValue(\"'+data[i].id_wbp+'\");">' +data[i].nama_dan_bin+ '</a></td
></tr>'
```

```

        }
        $('.nama_pendaftar').html(html);
    } else {
        $("#table").hide();
        $("#table1").hide();
        $("#header4").show();
    }
}

});

};

function getValue(cel){
    var id = cel;
    $.ajax({
        url : "<?php echo base_url();?>index.php/admin_pencarian/cek_napi
_detail",
        method : "GET",
        data : {id: id},
        dataType : 'json',
        success: function(data){
            if (data) {
                $("#table1").show();
                $("#header4").hide();
                $("#nama_napi").text(data.nama_dan_bin);
                $("#pasal").text(data.pasal);
                $("#blok").text(data.blok_kamar);
            } else {
                $("#table").hide();
                $("#table1").hide();
            }
        }
    });
}

```

```

        $("#"header4").show();
    }
}
});
};

</script>

```

5.16.3 Source Code CONTROLLER Pencarian WBP (*BOYER MOORE*)

```

public function cek_napi(){
    $nama = $_GET["nama"];
    $pattern = $nama;
    $data = $this->admin_model->get_tahanan();
    $array = array();
    $x = 0;
    foreach ($data as $data) {
        $data = $data->nama_dan_bin;
        $data = $this->BoyerMoore($data, $pattern);
        if ($data != -1) {
            $array[$x] = $data;
        }
        $x++;
    }

    if (!empty($array)) {
        $data = $this->admin_model->get_napi_by_nama($nama);
        echo json_encode($data);
    } else {

```

```

        echo json_encode(null);
    }
}

```

5.17 PENCARIAN KODE PENDAFTARAN (*BOYER MOORE*)

Data Antrian	
Nama Pengunjung	WILDAN
Nomor Antrian	2020011711
Jam Kunjungan	09:00:00 - 10:00:00
Alamat	adasdas
Jenis Kelamin	laki-laki
Status Hubungan	keluarga - Suami
Foto Pengunjung	

5.17.1 Source Code MODEL Pencarian Kode Pendaftaran

```

function get_pengunjung() {
    $this->db->select('*');
    $this->db->from('pengunjung');

    return $this->db->get()->result();
}

```

5.17.2 Source Code VIEW Pencarian Kode Pendaftaran

```

<div class="col-xs-6 col-sm-6 widget-container-col" id="widget-container-
col-4">

```

```
<div class="widget-box" id="widget-box-4">
    <div class="widget-header widget-header-large">
        <h4 class="widget-title">Cari Kode Antrian</h4>
    </div>

    <div class="widget-body">
        <div class="widget-main">
            <center>
                <div class="form-group">
                    <label for="psw"><span class="glyphicon glyphicon-user"></span> Kode Antrian</label>
                    <input type="text" class="form-control" id="kode" name="kode" placeholder="Masukkan Kode Antrian">
                </div>
                <button id="search" onclick="getInputValue();" class="btn btn-info" type="button">
                    <i class="ace-icon fa fa-check bigger-110"></i>
                    Submit
                </button>
            </center>
        </div>
    </div>
</div>
</div>
</div>
<hr>
```

```

<h4 id="header4" style="display: none;"><center>Kode Antrian Tidak Dite  
mukan</center></h4>

<table class="table table-bordered" id="table" style="display: none;">  
    <thead>  
        <tr>  
            <th colspan="2" id="header2"><center><b>List Pencarian Kode</b>  
</center></th>  
    </thead>  
    <tbody class="kode_antrian">  
        <tr>  
            <td id="nama_lengkapi">kakaka</td>  
        </tr>  
    </tbody>  
</table>  
  
<table class="table table-bordered" id="table1" style="display: none;">  
    <thead>  
        <tr>  
            <th colspan="2" id="header2"><center><b>Data Antrian</b></cente  
r></th>  
        </tr>  
        <tr>  
            <th>Nama Pengunjung</th>  
            <td id="nama_pendaftar"></td>  
        </tr>  
        <tr>  
            <th>Nomor Antrian</th>  
            <td id="nomor_antrian"></td>

```

```

</tr>
<tr>
    <th>Jam Kunjungan</th>
    <td id="jam_kunjungan"></td>
</tr>
<tr>
    <th>Alamat</th>
    <td id="alamat"></td>
</tr>
<tr>
    <th>Jenis Kelamin</th>
    <td id="jeniskelamin"></td>
</tr>
<tr>
    <th>Status Hubungan</th>
    <td id="status_hubungan"></td>
</tr>
<tr>
    <th>Foto Pengunjung</th>
    <td id="foto_pengunjung"><img id="fotoPengunjung" src="" width="200" height="200"></td>
</tr>
<tr>
    <th>Foto KTP</th>
    <td id="foto_ktp"><img id="fotoKtp" src="" width="200" height="200"></td>
</tr>
<tr>

```

```
<th>Status</th>
<td id="status"></td>
</tr>
<tr>
    <th>Keterangan</th>
    <td id="keterangan"></td>
</tr>
<tr>
    <th>Nama Narapidana</th>
    <td id="nama_napi"></td>
</tr>
<tr>
    <th>Pasal</th>
    <td id="pasal"></td>
</tr>
<tr>
    <th>Blok</th>
    <td id="blok"></td>
</tr>
<tr>
    <th>Nama pengikut 1</th>
    <td id="namapengikut1"></td>
</tr>
<tr>
    <th>Jenis kelamin pengikut 1</th>
    <td id="jkpengikut1"></td>
</tr>
<tr>
```

```
<th>Nomor KTP pengikut 1</th>
<td id="ktpengikut1"></td>
</tr>
<tr>
    <th>Foto pengikut 1</th>
    <td id="fotopengikut1"><img id="imgpengikut1" src="" width="20
0" height="200"></td>
</tr>
<tr>
    <th>Foto KTP pengikut 1</th>
    <td id=""><img id="imgktpengikut1" src="" width="200" height=""
200"></td>
</tr>
<tr>
    <th>Nama pengikut 2</th>
    <td id="namapengikut2"></td>
</tr>
<tr>
    <th>Jenis kelamin pengikut 2</th>
    <td id="jkpengikut2"></td>
</tr>
<tr>
    <th>Nomor KTP pengikut 2</th>
    <td id="ktpengikut2"></td>
</tr>
<tr>
    <th>Foto pengikut 2</th>
```

```

<td id="fotopengikut2"><img id="imgpengikut2" src="" width="200" height="200"></td>
</tr>
<tr>
    <th>Foto KTP pengikut 2</th>
    <td id=""><img id="imgktp pengikut2" src="" width="200" height="200"></td>
</tr>
</thead>
<tr>
    <th colspan="2" onclick="cetak()"><center><b><a href="#">Cetak</a></b></center></th>
</tr>
</table>
<p id="id_pengunjung" style="display: none;"></p>
</div>
<script type="text/javascript">
function getInputValue(){
    var kode = document.getElementById("kode").value;
    $.ajax({
        url : "<?php echo base_url();?>index.php/admin_pencarian/cek_kode_antrian",
        method : "GET",
        data : {kode: kode},
        dataType : 'json',
        success: function(data){
            if (data) {
                $("#table").show();
            }
        }
    });
}

```

```

        $("#" + table).hide();
        $("#" + header4).hide();

        var html = "";
        var i;
        for(i=0; i<data.length; i++){
            html += '<tr><td id="kode_antrian"><center><a href="#" onclick="getValue(\''+data[i].kode_antrian+'\')">' + data[i].kode_antrian + '</a></td></tr>';
        }
        $('.kode_antrian').html(html);
    } else {
        $("#" + table).hide();
        $("#" + table1).hide();
        $("#" + header4).show();
    }
}
});

};

function getValue(cel){
    var kode = cel;
    $.ajax({
        url : "<?php echo base_url();?>index.php/admin_pencarian/cek_kode_detail",
        method : "GET",
        data : {kode: kode},
        dataType : 'json',
        success: function(data){
            if (data) {

```

```

        $("#" + table).show();
        $("#" + header4).hide();
        $("#" + nama_pendaftar).text(data.nama_lengkap);
        $("#" + nomor_antrian).text(data.kode_antrian);
        $("#" + jam_kunjungan).text(data.dari + ' - ' + data.sampai);
        $("#" + status).text(data.status);
        $("#" + keterangan).text(data.keterangan);
        $("#" + nama_napi).text(data.wbpsertabin);
        $("#" + pasal).text(data.pasal);
        $("#" + blok).text(data.blok);
        $("#" + id_pengunjung).text(data.id_pengunjung);
        $("#" + alamat).text(data.alamat);
        $("#" + status_hubungan).text(data.status_hubungan);
        $("#" + namapengikut1).text(data.laki_laki);
        $("#" + jkpengikut1).text(data.perempuan);
        $("#" + ktppengikut1).text(data.anak_anak);
        $("#" + namapengikut2).text(data.jumlah);
        $("#" + jkpengikut2).text(data.jeniskelamin2);
        $("#" + ktppengikut2).text(data.noktp2);
        $("#" + jenis_kelamin).text(data.jenis_kelamin);
        document.getElementById("fotoPengunjung").src = "<?=base_url('uploads/kk')?>" + '/' + data.dokumen;
        document.getElementById("fotoKtp").src = "<?=base_url('uploads/kk')?>" + '/' + data.kk;
        document.getElementById("imgpengikut1").src = "<?=base_url('uploads/kk')?>" + '/' + data.fotopengikut1;
        document.getElementById("imgpengikut2").src = "<?=base_url('uploads/kk')?>" + '/' + data.fotopengikut2;
    
```

```
document.getElementById("imgktpengikut1").src = "<?=base_
url('uploads/kk')?>" +'/' +data.ktppengikut1;
document.getElementById("imgktpengikut2").src = "<?=base_
url('uploads/kk')?>" +'/' +data.ktppengikut2;
} else {
    $("#table").hide();
    $("#table1").hide();
    $("#header4").show();
}
});
};

function cetak()
{
    var cel = document.getElementById("id_pengunjung").textContent;
    window.open("<?=base_url('index.php/admin/cetak')?>" +'/' + cel);
}
</script>
```

5.17.3 Source Code CONTROLLER Pencarian Kode Pendaftaran

```
public function cek_kode_antrian(){

    $kode = $_GET["kode"];
    $pattern = $kode;
    $data = $this->admin_model->get_pengunjung();
    $array = array();
    $x = 0;
    foreach ($data as $data) {
        $data = $data->kode_antrian;
        $data = $this->BoyerMoore($data, $pattern);
        if ($data != -1) {
            $array[$x] = $data;
        }
        $x++;
    }

    if (!empty($array)) {
        $data = $this->admin_model->get_kode_by_id($kode);
        echo json_encode($data);
    } else {
        echo json_encode(null);
    }
}
```

5.18 PENCARIAN BERDASARKAN TANGGAL PENDAFTARAN

The screenshot shows a web application interface. At the top, there is a navigation bar with links to Home, Other Pages, and Blank Page. Below the navigation, a search form titled "Cari Pengunjung Berdasarkan Tanggal" is displayed. It has a text input field labeled "Masukkan Tanggal" containing the value "1/17/2020" and a blue "Submit" button. Below the search form, a table titled "List Pencarian Kode" shows one result: "2020011711". To the right of the search results, a table titled "Data Antrian" displays visitor information:

Nama Pengunjung	WILDAN
Nomor Antrian	2020011711
Jam Kunjungan	09:00:00 - 10:00:00
Alamat	adasdas
Jenis Kelamin	laki-laki
Status Hubungan	keluarga - Suami
Foto Pengunjung	[Placeholder for photo]

gambar

5.18.1 Source Code MODEL Pencarian tanggal pendaftaran

```
function get_pengunjung() {  
    $this->db->select('*');  
    $this->db->from('pengunjung');  
  
    return $this->db->get()->result();  
}
```

5.18.2 Source Code VIEW Pencarian tanggal pendaftaran

```
<div class="col-xs-6 col-sm-6 widget-container-col" id="widget-container-col-4">  
    <div class="widget-box" id="widget-box-4">  
        <div class="widget-header widget-header-large">  
            <h4 class="widget-  
title">Cari Pengunjung Berdasarkan Tanggal</h4>  
        </div>
```

```
<div class="widget-body">
  <div class="widget-main">
    <center>
      <div class="form-group">
        <label for="psw"><span class="glyphicon glyphicon-user"></span> Masukkan Tanggal</label>
        <input type="date" class="form-control" id="kode" name="kode" placeholder="Masukkan Tanggal">
      </div>
      <button id="search" onclick="getInputValue();" class="btn btn-info" type="button">
        <i class="ace-icon fa fa-check bigger-110"></i>
        Submit
      </button>
    </center>
  </div>
</div>
</div>
</div>
<hr>
<h4 id="header4" style="display: none;"><center>Pengunjung pada tanggal ini tidak ditemukan.</center></h4>
<table class="table table-bordered" id="table" style="display: none;">
  <thead>
    <tr>
```

```

<th colspan="2" id="header2"><center><b>List Pencarian Kode</b>
</center></th>

</tr>
</thead>

<tbody class="kode_antrian">
<tr>
    <td id="nama_lengkapi">kakaka</td>
</tr>
</tbody>
</table>

<table class="table table-bordered" id="table1" style="display: none;">
<thead>
<tr>
    <th colspan="2" id="header2"><center><b>Data Antrian</b></center></th>
</tr>
<tr>
    <th>Nama Pengunjung</th>
    <td id="nama_pendaftar"></td>
</tr>
<tr>
    <th>Nomor Antrian</th>
    <td id="nomor_antrian"></td>
</tr>
<tr>
    <th>Jam Kunjungan</th>
    <td id="jam_kunjungan"></td>
</tr>

```

```

<tr>
    <th>Alamat</th>
    <td id="alamat"></td>
</tr>
<tr>
    <th>Jenis Kelamin</th>
    <td id="jeniskelamin"></td>
</tr>
<tr>
    <th>Status Hubungan</th>
    <td id="status_hubungan"></td>
</tr>
<tr>
    <th>Foto Pengunjung</th>
    <td id="foto_pengunjung"><img id="fotoPengunjung" src="" width="200" height="200"></td>
</tr>
<tr>
    <th>Foto KTP</th>
    <td id="foto_ktp"><img id="fotoKtp" src="" width="200" height="200"></td>
</tr>
<tr>
    <th>Status</th>
    <td id="status"></td>
</tr>
<tr>
    <th>Keterangan</th>

```

```
<td id="keterangan"></td>
</tr>
<tr>
    <th>Nama Narapidana</th>
    <td id="nama_napi"></td>
</tr>
<tr>
    <th>Pasal</th>
    <td id="pasal"></td>
</tr>
<tr>
    <th>Blok</th>
    <td id="blok"></td>
</tr>
<tr>
    <th>Nama pengikut 1</th>
    <td id="namapengikut1"></td>
</tr>
<tr>
    <th>Jenis kelamin pengikut 1</th>
    <td id="jkpengikut1"></td>
</tr>
<tr>
    <th>Nomor KTP pengikut 1</th>
    <td id="ktppengikut1"></td>
</tr>
<tr>
    <th>Foto pengikut 1</th>
```

```
<td id="fotopengikut1"><img id="imgpengikut1" src="" width="200" height="200"></td>
</tr>
<tr>
    <th>Foto KTP pengikut 1</th>
    <td id=""><img id="imgktp pengikut1" src="" width="200" height="200"></td>
</tr>
<tr>
    <th>Nama pengikut 2</th>
    <td id="namapengikut2"></td>
</tr>
<tr>
    <th>Jenis kelamin pengikut 2</th>
    <td id="jkpengikut2"></td>
</tr>
<tr>
    <th>Nomor KTP pengikut 2</th>
    <td id="ktppengikut2"></td>
</tr>
<tr>
    <th>Foto pengikut 2</th>
    <td id="fotopengikut2"><img id="imgpengikut2" src="" width="200" height="200"></td>
</tr>
<tr>
    <th>Foto KTP pengikut 2</th>
```

```

<td id=""><img id="imgktppengikut2" src="" width="200" height="200"></td>
</tr>
</thead>
<tr>
<th colspan="2" onclick="cetak()"><center><b><a href="#">Cetak</a></b></center></th>
</tr>
</table>
<p id="id_pengunjung" style="display: none;"></p>
</div>
<script type="text/javascript">
function getInputValue(){
    var tanggal = document.getElementById("kode").value;
    $.ajax({
        url : "<?php echo base_url();?>index.php/admin_pencarian/cek_kode_tanggal",
        method : "GET",
        data : {tanggal: tanggal},
        dataType : 'json',
        success: function(data){
            if (data) {
                $("#table").show();
                $("#table1").hide();
                $("#header4").hide();
                var html = "";
                var i;
                for(i=0; i<data.length; i++){

```

```

        html += '<tr><td id="kode_antrian"><center><a href="#" onclick="getValue(\"'+data[i].kode_antrian+'\");">' + data[i].kode_antrian + '</a></td></tr>'
    }
    $('.kode_antrian').html(html);
} else {
    $("#table").hide();
    $("#table1").hide();
    $("#header4").show();
}
}
});

};

function getValue(cel){
var kode = cel;
$.ajax({
    url : "<?php echo base_url();?>index.php/admin_pencarian/cek_kode_detail",
    method : "GET",
    data : {kode: kode},
    dataType : 'json',
    success: function(data){
        if (data) {
            $("#table1").show();
            $("#header4").hide();
            $("#nama_pendaftar").text(data.nama_lengkap);
            $("#nomor_antrian").text(data.kode_antrian);
            $("#jam_kunjungan").text(data.dari + ' - ' + data.sampai);
        }
    }
});
}

```

```

        $("#" + status).text(data.status);
        $("#" + keterangan).text(data.keterangan);
        $("#" + nama_napi).text(data.wbpsertabin);
        $("#" + pasal).text(data.pasal);
        $("#" + blok).text(data.blok);
        $("#" + id_pengunjung).text(data.id_pengunjung);
        $("#" + alamat).text(data.alamat);
        $("#" + status_hubungan).text(data.status_hubungan);
        $("#" + namapengikut1).text(data.laki_laki);
        $("#" + jkpengikut1).text(data.perempuan);
        $("#" + ktppengikut1).text(data.anak_anak);
        $("#" + namapengikut2).text(data.jumlah);
        $("#" + jkpengikut2).text(data.jeniskelamin2);
        $("#" + ktppengikut2).text(data.noktp2);
        $("#" + jeniskelamin).text(data.jenis_kelamin);

        document.getElementById("fotoPengunjung").src = "<?=base_url('uploads/kk')?>" + '/' + data.dokumen;

        document.getElementById("fotoKtp").src = "<?=base_url('uploads/kk')?>" + '/' + data.kk;

        document.getElementById("imgpengikut1").src = "<?=base_url('uploads/kk')?>" + '/' + data.fotopengikut1;

        document.getElementById("imgpengikut2").src = "<?=base_url('uploads/kk')?>" + '/' + data.fotopengikut2;

        document.getElementById("imgktppengikut1").src = "<?=base_url('uploads/kk')?>" + '/' + data.ktppengikut1;

        document.getElementById("imgktppengikut2").src = "<?=base_url('uploads/kk')?>" + '/' + data.ktppengikut2;

    } else {

```

```

        $("#" + id).show();
    }
}
);
};

function cetak()
{
    var cel = document.getElementById("id_pengunjung").textContent;
    window.open("<?=base_url('index.php/admin/cetak')?>" + '/' + cel);
}
</script>

```

5.18.3 Source Code CONTROLLER Pencarian tanggal pendaftaran

```

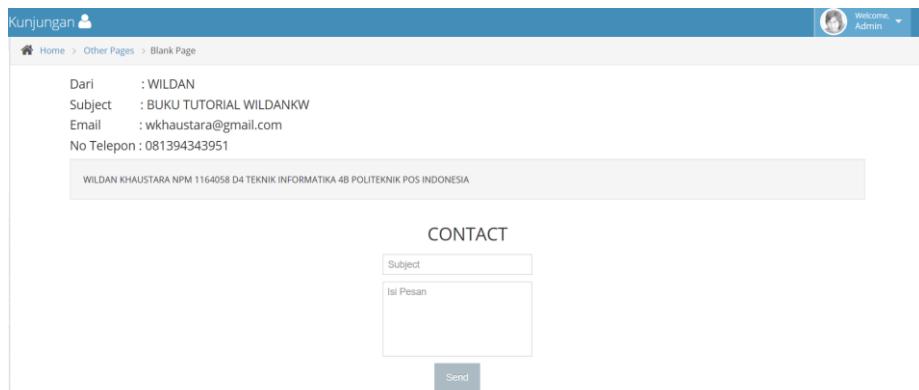
public function cek_kode_tanggal(){
    $tanggal = $_GET["tanggal"];
    $pattern = $tanggal;
    $data = $this->admin_model->get_pengunjung();
    $array = array();
    $x = 0;
    foreach ($data as $data) {
        $data = $data->tanggal_kunjungan;
        $data = $this->BoyerMoore($data, $pattern);
        if ($data != -1) {
            $array[$x] = $data;
        }
        $x++;
    }
}

```

```
}
```

```
if (!empty($array)) {  
    $data = $this->admin_model->get_kode_by_tanggal($tanggal);  
    echo json_encode($data);  
} else {  
    echo json_encode(null);  
}  
}
```

5.19 PESAN



5.19.1 Source Code MODEL Pesan

```
function bukapesan($id)  
{  
    $this->db->select('*');  
    $this->db->from('pesan');  
    $this->db->where('id_pesan',$id);
```

```
        return $this->db->get()->row();  
    }  
  
    function insert_pesan($data)  
    {  
        $this->db->insert('pesan',$data);  
        return mysql_affected_rows();  
    }  
}
```

5.19.2 Source Code VIEW Pesan

```

echo $info;
}
?>
<form role="form" action=<?php echo site_url('admin_pesanan');?>" method="post">
<div class="row">
<div class="col-sm-12 form-group">
<input type="hidden" name="id_peserta" value=<?=$pesan->id_peserta?>>
<input type="hidden" name="id_pesanan" value=<?=$pesan->id_pesanan?>>
<input type="text" name="subject" class="form-control" placeholder="Subject"><br>
<textarea class="form-control" id="isi" name="isi" placeholder="Isi Pesan" rows="5"></textarea><br>
<button class="btn btn-default" type="submit">Send</button>
</div>
</div>
</form>
</div>
</center>
```

5.19.3 Source Code CONTROLLER Pesan

```

function bukapesan() {
$logged_in = $this->session->userdata('logged_in');
$level = $this->session->userdata('level');
if (!empty($logged_in) && $level=='admin')
```

```

{
    $id_pesan = $this->uri->segment(3);
    $data['pesan'] = $this->admin_model->bukapesan($id_pesan);
    $this->template->admin('admin/bukapesan',$data);
}

else
{
    $this->template->home('home/content');
}

}

function balas_pesan() {
    $id_peserta = $this->input->post('id_peserta');
    $id_pesan = $this->input->post('id_pesan');
    $data = array(
        'id_peserta' => 1,
        'isi' => $this->input->post('isi'),
        'subject' => $this->input->post('subject'),
        'no_hp' => $this->input->post('no_hp'),
        'nama_lengkap' => 'Admin',
        'tujuan' => $this->input->post('id_peserta'),
    );
}

$insert = $this->Kunjungan_model->insert_pesan($data);
$this->session->set_flashdata('info','Pesan Terkirim');
redirect('admin_pesan/bukapesan/'.$id_pesan);

}
}

```

BAB VI PENERAPAN BOYER MOORE

6.1 PENERAPAN ALGORITMA PADA APLIKASI

6.2.2 Alasan Menggunakan Algoritma Boyer moore

Alasan menggunakan algoritma *Boyer moore* karena algoritma Boyer Moree adalah algoritma paling cepat dalam melakukan pencarian data atau paling cepat dalam melakukan *string matching* karena algoritma *Boyer moore* dapat sekaligus bergeser sesuai dengan Panjang text yang di cari, sedangkan algoritma lain seperti algoritma *Brute Force*, *Knuth Morris Pratt*, *Karp Rabin* mereka hanya melakukan pergeseran per1 karakter atau huruf, sedangkan algoritma *Boyer moore* mencari kecocokan yang sesuai dari karakter akhir dengan karakter yang akan dicocokan.

Algortima Pencocokan Data WBP.

```
function BoyerMoore($text, $pattern) {  
    $patlen = strlen($pattern);  
    $textlen = strlen($text);  
    $table = $this->makeCharTable($pattern);  
  
    for ($i=$patlen-1; $i < $textlen;) {  
        $t = $i;  
        for ($j=$patlen-1; $pattern[$j]==$text[$i]; $j--, $i--) {  
            if($j == 0) [  
                return $i;  
            ]  
        }  
        $i = $t;  
        if(array_key_exists($text[$i], $table))  
            $i = $i + max($table[$text[$i]], 1);  
        else  
            $i += $patlen;  
    }  
    return -1;  
}  
  
function makeCharTable($string) {  
    $len = strlen($string);  
    $table = array();  
    for ($i=0; $i < $len; $i++) {  
        $table[$string[$i]] = $len - $i - 1;  
    }  
    return $table;  
}
```

Gambar 7.1 Pencocokan Data (String Matching)

- function BoyerMoore(\$text, \$pattern) {
fungsi ini butuh 2 parameter yaitu text dan pattern}
- \$patlen = strlen(\$pattern);
ini membuat parameter yang isinya dari fungsi fungsi sebelumnya tapi dibuat lowercase semua
- \$textlen = strlen(\$text);
fungsi ini untuk mengembalikan panjang string, string yang di maksud adalah variabel \$text
- \$tabel = \$this->makeCharTabel(\$pattern);
Membuat fungsi tabel makechartabel
- for (\$i=\$patlen-1; \$i < \$textlen;) {
fungsi ini untuk menghitung dimana \$i hasilnya \$patlen dikurang 1, \$i selalu lebih kecil dari variabel \$textlen}
- \$t = \$i;
Variable \$t = \$i
- for (\$j=\$patlen-1; \$pattern[\$j]==\$text[\$i]; \$j--, \$i--) {
fungsi ini untuk menghitung dimana \$j hasilnya \$patlen sama dengan 1\$
- if(\$j == 0) {
jika \$j sama dengan 0
- return \$i;
maka kembali ke \$i nilai pergeseran
- \$i = \$t;
\$i sama dengan data tabel
- if(array_key_exists(\$text[\$i], \$tabel))
jika array tersedia maka akan melakukan pergeseran tabel
- \$i = \$i + max(\$tabel[\$text[\$i]], 1);
Melakukan pergeseran tabel array

- `$i += $patlen;`
- Melakukan pergeseran tabel array
- `return -1;`
kembali ke kondisi awal
- `function makeCharTabel($string) {`
fungsi ini membutuhkan parameter `$string`
- `$len = strlen($string);`
ini untuk membuat variabel `$len` dimana akan mengembalikan panjang kata dari `$string`
- `$tabel = array();`
membuat variabel berisi array
- `for ($i=0; $i < $len; $i++) {`
perhitungan, jadi `$i` adalah 0, dan `$i` selalu lebih kecil dari `$len`
- `$tabel[$string[$i]] = $len - $i - 1;`
setelah perhitungan dijalankan, akan membuat lagi variabel `$tabel` yang array tadi di tampung oleh `$string` isinya `$len-$i-1`
- `return $tabel;`
mengembalikan hasil array tadi berapa

Hasil dari kodingan di atas bertujuan untuk:

1. Melakukan pencarian *pattern* text dari paling kanan apakah text tersebut tersedia didalam *pattern*
2. Jika huruf paling kanan tersedia pada *pattern* maka akan dilanjutkan ke proses pergeseran
3. Melakukan pergeseran nilai *bad suffix* dan *good suffix* pada inputan text
4. Apakah text yang di inputkan memiliki nilai kecocokan dengan *pattern* jika ada maka pencarian akan selesai dan dinyatakan *string matching*, dan jika tidak ada kecocokan akan terjadi miss match maka pencarian akan selesai

6.1.2 Algoritma Pencarian Data (WBP, REGISTRASI, TANGGAL)

```
function cari_napi($pattern) { //Fungsi cari napi dengan menangkap parameter pattern
    $pattern = $pattern; //Membuat variabel dengan nama pattern isinya pattern
    $text = $this->admin_model->get_tahanan(); //Variabel text berisi model pengambilan data tahanan
    $array = array(); //Membuat variabel array yang berisi array
    $x = 0; //Membuat variabel x bernilai 0
    foreach ($text as $text) { //Pemanggilan setiap data tahanan
        $text = $text->nama_dan_bin; //Pengambilan data nama_dan_bin
        $data = $this->BoyerMoore($text, $pattern); //Mengambil fungsi boyermoore dengan
        //melempar parameter text dan pattern
        if ($data != -1) { //jika hasil dari boyer moore tidak sama dengan -1
            $array[$x] = $text; //Array 0 adalah nama_dan_bin yang sudah di foreach
        } //Tutup if
        $x++; //Nilai 0 bertambah 1 setiap kali melakukan foreach
    } //Tutup if
    foreach ($array as $tahanan) { //Untuk setiap array yg dipanggil
        return 1; //Kembalikan nilai 1
    } //Tutup foreach
    if (empty($array)) { //Jika array tidak berisi apapun atau null
        return 0; //Kembalikan nilai 0
    } //Tutup if

    if ($this->cari_napi($this->input->post('wbpsertabin')) == 1) {
        if ($this->cari_pasal($this->input->post('pasal'), $this->input->post('wbpsertabin')) == 1) {
            if ($this->cari_blok($this->input->post('blok'), $this->input->post('wbpsertabin')) == 1) {
                $keterangan = "Info napi yang diinputkan terdapat di database";
                $datap['status'] = "diverifikasi";
            }
        }
    }
}
```

Gambar 7.2 Pencocokan Nama Napi

- function cari_napi(\$pattern) {
 Fungsi cari napi dengan menangkap parameter *pattern*
- \$pattern = \$pattern;
 Membuat variabel dengan nama *pattern* isinya *pattern*
- \$text = \$this->admin_model->get_tahanan();
 Variabel text berisi model pengambilan data tahanan
- \$array = array();
 Membuat variabel array yang berisi array
- \$x = 0;
 Membuat variabel x bernilai 0
- foreach (\$text as \$text) {
 Pemanggilan setiap data tahanan
- \$text = \$text->nama_dan_bin;
 Pengambilan data nama_dan_bin

- `$data = $this->BoyerMoore($text, $pattern);`
 Mengambil fungsi booyer moore dengan melempar parameter text dan *pattern*
- `if ($data != -1) {`
 Jika hasil dari booyer moore tidak sama dengan -1
- `$array[$x] = $text;`
 Array 0 adalah nama_dan_bin yang sudah di foreach
- `}`
 Tutup if
- `$x++;`
 Nilai 0 bertambah 1 setiap kali melakukan foreach
- `}`
 Tutup if
- `foreach ($array as $tahanan) {`
 Untuk setiap array yg dipanggil
- `return 1;`
 Kembalikan nilai 1
- `}`
 Tutup foreach
- `if (empty($array)) {`
 Jika array tidak berisi apapun atau null
- `return 0;`
 Kembalikan nilai 0

- }
Tutup if

- }
Tutup fungsi

Hasil dari kodingan di atas bertujuan untuk:

1. Mencari (nama napi, blok, dan pasal dengan menggunakan parameter *pattern*)
2. Lalu memanggil fungsi metode *boyer moore*
3. Lalu metode *Boyer Moore* Melakukan pergeseran nilai *bad suffix* dan *good suffix* pada inputan text
4. Apakah text yang di inputkan memiliki nilai kecocokan dengan *pattern* dan melakukan pencocokan yang akan dilemparkan kepada *database*, jika ada kecocokan maka akan melakukan set status diverifikasi dan jika tidak ada kecocokan akan melakukan set status ditolak

WBP YANG DIKUNJUNGI

Pilih Tanggal

Pilih Jadwal

Nama WBP

pasal

Blok Sel tahanan

Upload foto WBP

Gambar 7.3 Tempat *Boyer Moore* Di Implementasikan

6.1.3 Hasil Pengujian

Berikut merupakan daftar data WBP yang tersedia (data *pattern*)
Terdapat 2 data wbp yang bernama JAMAL dan ASEP BIN TATANG

No WBP	Nama lengkap serta bin	pasal	blok kamar	foto	Tambah WBP
5	ASEP BIN TATANG	12	24		Selengkapnya Edit Hapus Data History
6	JAMAL	12	24		Selengkapnya Edit Hapus Data History

Gambar 7.4 data wbp

Jika pengunjung melakukan *input* data dengan data wbp yang sesuai yaitu JAMAL / ASEP BIN TATANG maka pendaftaran akan di verifikasi

Pilih Tanggal

Pilih Jadwal

Nama WBP

pasal

Blok Sel tahanan

Upload foto WBP

Gambar 7.5 *input* data yang sesuai

Hasil dari verifikasi pengunjung akan mendapatkan nomor antrian dan bukti pendaftaran verifikasi via email

History Pendaftaran							
Tanggal Pendaftaran	Nomor Antrian	Jam Kunjungan	Status	Keterangan	Nama Napi	Pasal	Blok
2020-01-05 19:53:23	2020010611	09:00:00 - 10:00:00	diverifikasi	Info napi yang diinputkan terdapat di database	JAMAL	12	24

Gambar 7.6 hasil verifikasi *input* data sesuai

Jika text inputan tidak sesuai dengan *database* maka pendaftaran akan ditolak

WBP YANG DIKUNJUNGI

Pilih Tanggal	1/6/2020
Pilih Jadwal	09:00:00 - 10:00:00
Nama WBP	AJIS TRUGUNGUN
pasal	24
Blok Sel tahanan	21
Upload foto WBP	[C:\xampp\htdocs\KunjunganRutan\Bandung\assets\5.ASEP HENDRA HARMAEN.jpg] <input type="button" value="Browse..."/>

Gambar 7.7 *input* data yang tidak sesuai

Hasil dari pendaftaran ditolak (text tidak sesuai dengan *pattern* yang ada pada *database*)

History Pendaftaran								
Tanggal Pendaftaran	Nomor Autrian	Jam Kunjungan	Status	Keterangan	Nama Napi	Pasal	Blok	Foto
2020-01-05 19:59:50		09:00:00 - 10:00:00	ditolak	Ditolak, NAMA Napi yang Anda inputkan tidak cocok, silahkan hubungi Admin	AJIS TRUGUNGUN	24	21	

Gambar 7.8 hasil *input* data yang tidak sesuai

6.2 KELEBIHAN DAN KEKURANGAN ALGORITMA BOYER MOORE

6.2.1 Kelebihan

Tidak seperti pencarian *string* lainnya Brute Force, Knuth-Morris-Pratt yang mempunyai cara kerja membandingkan satu – persatu karakter dari kiri ke kanan. Boyer-Moore membandingkan karakter dari kanan ke kiri dan memiliki loncatan karakter yang besarsehingga mempercepat pencarian *string* karena dengan hanya memeriksa sedikit karakter, dapat langsung diketahui bahwa *string* yang dicari tidak ditemukan dan dapat digeser ke posisi berikutnya.[30]

6.2.2 Kekurangan

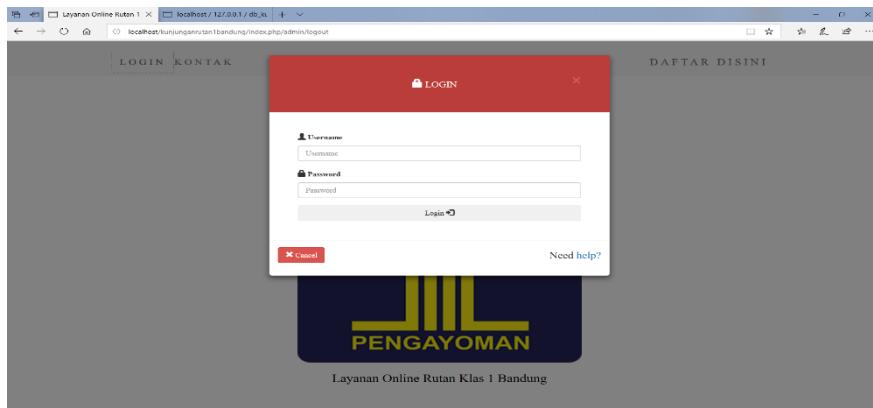
Algoritma Boyer-Moore mencocokan *Pattern* dari kanan ke kiri oleh sebab itu kelemahan dari algoritma ini adalah ketika semua karakter memiliki kesamaan atau cocok dan hanya karakter terakhir atau karakter paling kiri yang berbeda maka pencarian ini akan memerlukan waktu yang sedikit lama [Utomo, 2008]

BAB VII HASIL ANTAR MUKA APLIKASI

7.1 HALAMAN ANTARMUKA ADMIN

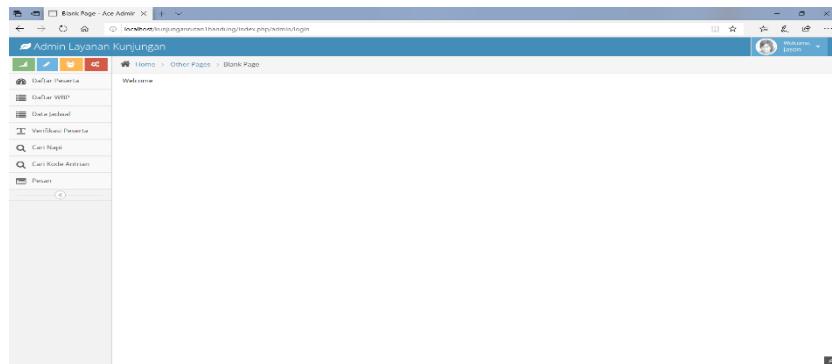
7.4.1 Halaman Login

Di halaman ini admin melakukan *login* untuk dapat mengakses program dan melakukan pendataan pinjaman serta pengiriman slip gaji melalui email



Gambar 7.6 Halaman Login

7.4.2 Utama Admin



Gambar 7.7 Halaman Utama Admin Dashboard

Pada halaman ini Merupakan Halaman Utama Antarmuka Admin dan terdapat menu

1.Daftar Peserta Kunjungan

2.Daftar WBP

3.Data Jadwal

4.Verifikasi Peserta

5.Cari Data WBP

6.Cari Kode Antrian

7.Pesan

7.4.3 Kelola Pengunjung

NIK	USERNAME	Nama Lengkap	Jenis Kelamin	Status Hubungan	Email	Foto	download foto	Tambah Peserta
	gigi	Gigi					Download Foto	Selengkapnya Edit Hapus
	elex	elex					Download Foto	Selengkapnya Edit Hapus
12345	anek	Hikmah Muhammadi	laki-laki	keluarga - Suami	wkhaurita@gmail.com		Download Foto	Selengkapnya Edit Hapus
2213123	a	a	laki-laki	keluarga - Suami	weldanhkhausiar@rediffmail.com		Download Foto	Selengkapnya Edit Hapus
78645387	urang	Urang Aring	laki-laki	keluarga - Suami	urang@gmail.com		Download Foto	Selengkapnya Edit Hapus
78645267	gaaja	Angga	laki-laki	keluarga - ibu	hulidumst52@gmail.com		Download Foto	Selengkapnya Edit Hapus

Gambar 7.8 Halaman Utama Admin

Pada halaman ini admin dapat melakukan pengelolaan terhadap data Pengunjung mulai dari *view*, *update*, *delete* data Pengunjung.

7.4.4 Halaman Kelola WBP

The screenshot shows a web-based administration interface for managing visitors (WBP). The main title is "Daftar Keseluruhan WBP : 4". A sidebar on the left contains links for "Daftar Peserta", "Daftar WBP", "Data Jadwal", "Verifikasi Peserta", "Cari Napi", "Cari Kode Antrian", and "Pesan". The main content area displays a table with four entries:

No WBP	Nama lengkap serta bin	pasal	blok kamar	foto	Tambah WBP
1	Ariel Tatim	2	8A		Selenglaprya Edit Hapus Data Pengunjung
2	Ariel Peterpan	363	22		Selenglaprya Edit Hapus Data Pengunjung
5	Doblish	378	7A		Selenglaprya Edit Hapus Data Pengunjung
6	Jamal	12	24		Selenglaprya Edit Hapus Data Pengunjung

Showing 1 to 4 of 4 entries

Gambar 7.9 Halaman Utama Admin WBP

Pada halaman ini admin dapat melakukan pengelolaan terhadap data WBP mulai dari *view*, *update*, *delete* data WBP.

7.4.5 Kelola Jadwal

The screenshot shows a web-based administration interface for managing visit schedules (jadwal kunjungan). The main title is "Jumlah Data Jadwal : 6". A sidebar on the left contains links for "Daftar Peserta", "Daftar WBP", "Data Jadwal", "Verifikasi Peserta", "Cari Napi", "Cari Kode Antrian", and "Pesan". The main content area displays a table with six entries:

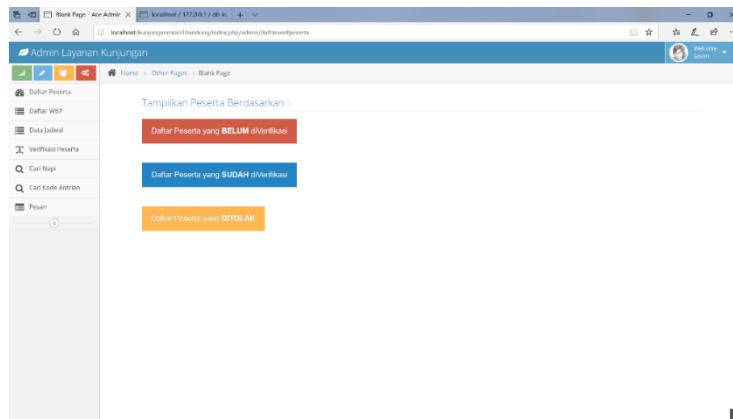
Dari Jam	Sampai Jam	Kuota	Tambah Jadwal
09:00:00	10:00:00	25	Edit Hapus
10:00:00	11:00:00	25	Edit Hapus
11:00:00	12:00:00	25	Edit Hapus
13:00:00	14:00:00	25	Edit Hapus
14:00:00	15:00:00	25	Edit Hapus
15:00:00	16:00:00	25	Edit Hapus

Showing 1 to 6 of 6 entries

Gambar 7.10 Halaman kelola jadwal kunjungan

Pada halaman ini admin dapat melakukan pengelolaan terhadap data Jadwal mulai dari *view*, *update*, *delete* data Jadwal.

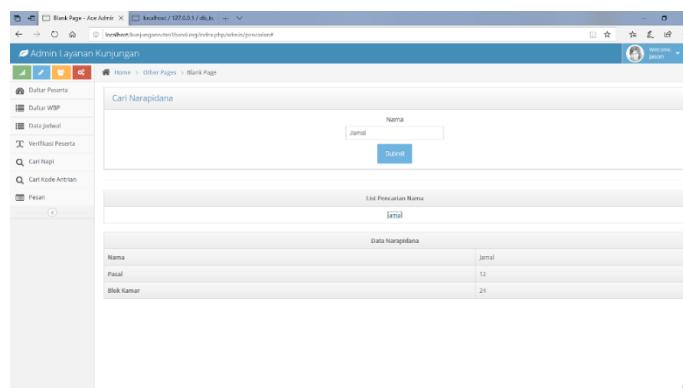
7.4.6 Kelola Daftar Verifikasi



Gambar 7.11 Halaman kelola Verifikasi Pengunjung

Pada Halaman Ini Admin Dapat Melihat Daftar Verifikasi Pengunjung, mulai dari DiVerifikasi,Ditolak Dan Belum diverifikasi

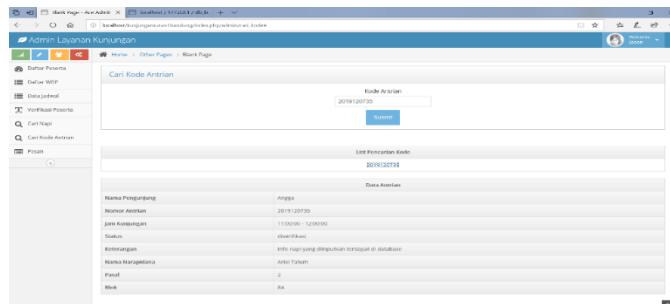
7.4.7 Cari NamaWBP



Gambar 7.12 Halaman Cari Nama WBP

Pada Halaman Ini admin dapat melakukan Pencarian Nama WBP Menggunakan Metode Algoritma *Boyer moore*

7.4.8 Cari No Registrasi



Gambar 7.13 Halaman Cari No Pengunjung

Pada Halaman Ini admin dapat melakukan Pencarian Nomor registrasi pengunjung kemudian admin dapat melakukan print terhadap no pengunjung yang di tampilkan

7.4.2 Cetak



Gambar 7.14 Hasil Print data Pengunjung

Hasil dari Print Out Data Pengunjung, untuk berkas masuk kedalam rutan dan melakukan kunjungan

7.4.2 Pemberitahuan Email



Rutan Klas 1 Bandung <internship1wildan@gmail.com>
kepada saya ▾

----- PENDAFTARAN DITOLAK -----

No Registrasi : Nomor Antrian tidak didapatkan
No Identitas : 336555
Nama Lengkap : WILDAN
Nama Wbp Yang Dikunjungi : ASEP JAMAL BIN TATANG

----- PENDAFTARAN DITOLAK -----



Rutan Klas 1 Bandung <internship1wildan@gmail.com>
kepada saya ▾

----- PENDAFTARAN DITERIMA -----

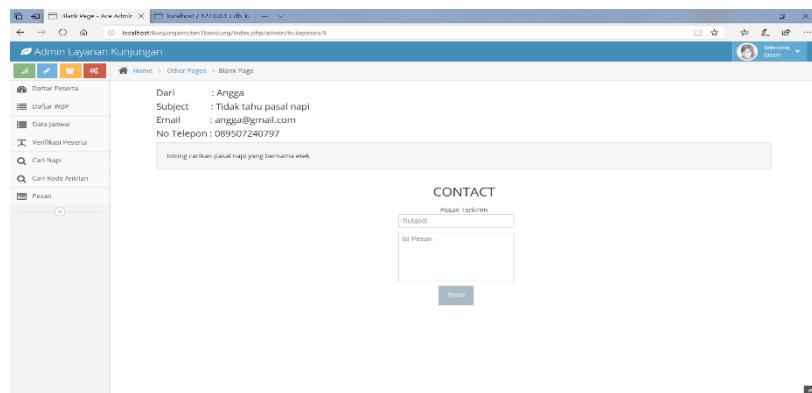
No Registrasi : 2019122612
No Identitas : 336555
Nama Lengkap : WILDAN
Nama Wbp Yang Dikunjungi : JAMAL

----- PENDAFTARAN DITERIMA -----
TUNJUKAN BUKTI EMAIL INI KEPADA PETUGAS KUNJUNGAN

Gambar 7.15 Hasil Pemberitahuan Via Email Pengunjung

Hasil dari pemberitahuan Verifikasi Via email

6.4.2 Pesan

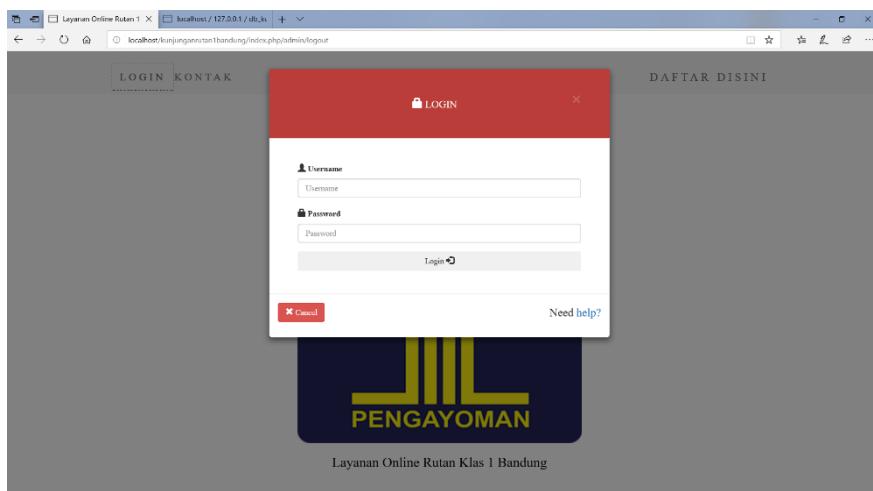


Gambar 7.16 Halaman Cari No Pengunjung

Pada Halaman Ini admin dapat melihat isi pesan masuk dan membalas pesan masuk tersebut

7.5 HALAMAN ANTARMUKA PENGUNJUNG

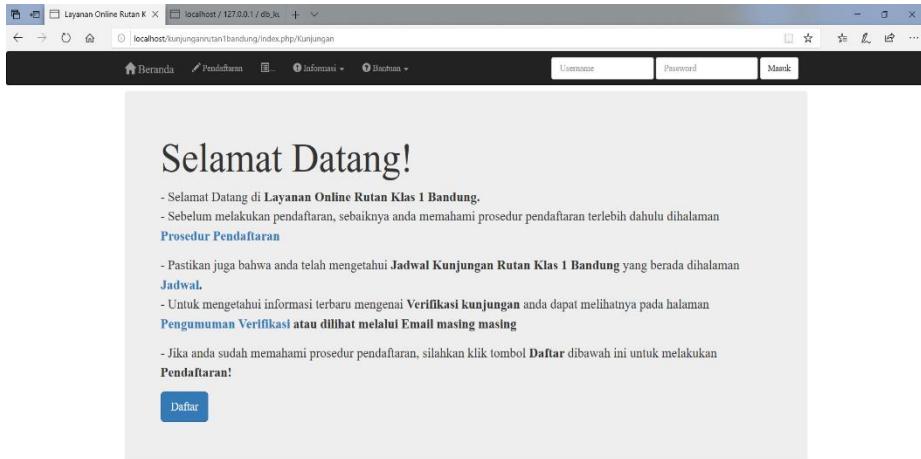
7.5.1 Login



Gambar 7.17 Halaman Login

Di halaman ini Pengunjung melakukan *login* untuk dapat mengakses program dan melakukan Pengisian Formulir Pendaftaran

7.5.2 Pendaftaran



Gambar 7.18 Halaman pendaftaran

Di halaman ini Pengunjung melakukan *login* untuk dapat mengakses program dan melakukan Pengisian Formulir Pendaftaran

7.5.3 Prosedur Pendaftaran



1. Pendaftaran.

Calon pengunjung melakukan pendaftaran pada website layanan online Rutan I Bandung, melalui halaman Pendaftaran. Kemudian calon pengunjung akan diarahkan untuk Login menggunakan username dan password yang telah didaftarkan.

2. Mengisi biodata.

Melengkapi form pendaftaran (**nama WBP yang akan dikunjungi harus sesuai dengan nama registrasi WBP dan menggunakan huruf kapital**) atau pendaftaran kunjungan akan di **TOLAK**.

3. Menunggu verifikasi.

Menunggu verifikasi dari pihak rutan untuk menindak lanjuti, yang akan dikirim via **Email**.

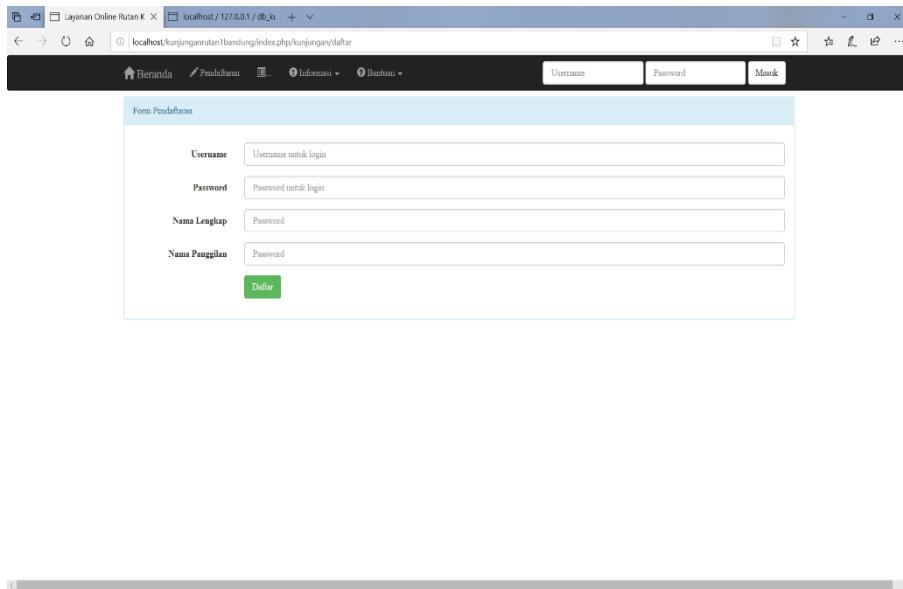
4. Mencetak Bukti Pendaftaran.

Biodata dan data kunjungan yang sudah dilengkapi kemudian dicetak di kertas A4. Untuk mencetak, masuk ke menu cetak. Menu ini akan tampil jika Anda sudah login. Biodata dan data kunjungan yang sudah dicetak tersebut selanjutnya dibawa ke **Pelugas Anggota Jaga Kunjungan Rutan Klas I Bandung** untuk tiken masuk.

Gambar 7.19 Halaman Prosedur Pendaftaran

Di halaman ini Pengunjung diarahkan bagaimana melakukan pendaftaran online

7.5.4 Daftar Akun

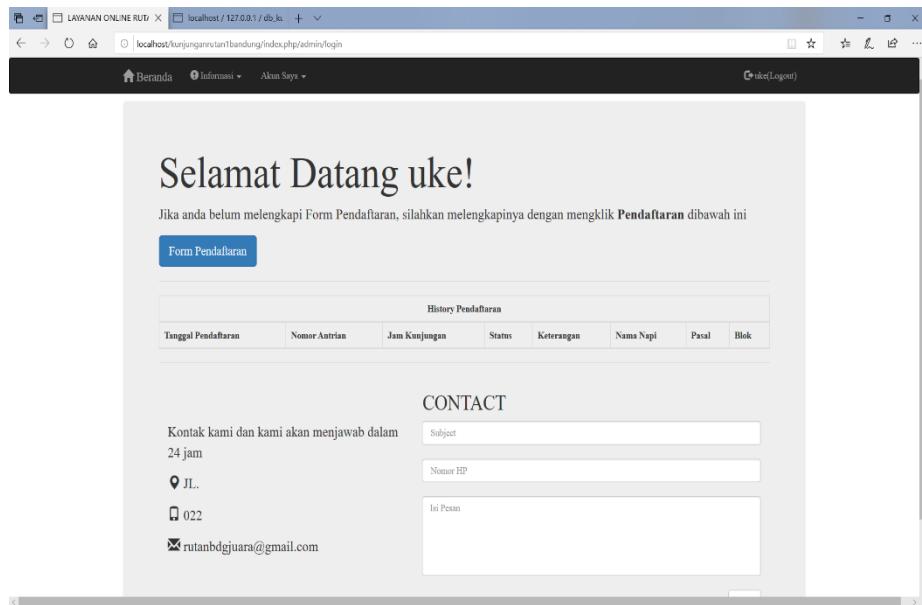


The screenshot shows a web browser window with the URL `localhost:kunjunganonline/banting/index.php/kunjungan/daftar`. The page title is "Form Pendaftaran". The form contains four input fields: "Username" (placeholder: "Username untuk login"), "Password" (placeholder: "Password untuk login"), "Nama Lengkap" (placeholder: "Password"), and "Nama Panggilan" (placeholder: "Password"). Below the form is a green "Daftar" button.

Gambar 7.20 Halaman daftar Login

Di halaman ini Pengunjung melakukan pendaftaran online, dan mengisi username,password,nama lengkap dan nama panggilan

7.5.5 Dashboard



Gambar 7.21 Halaman tampilan Dahsboard

Di halaman ini Pengunjung akan melakukan pengisian Formulir pendaftaran.

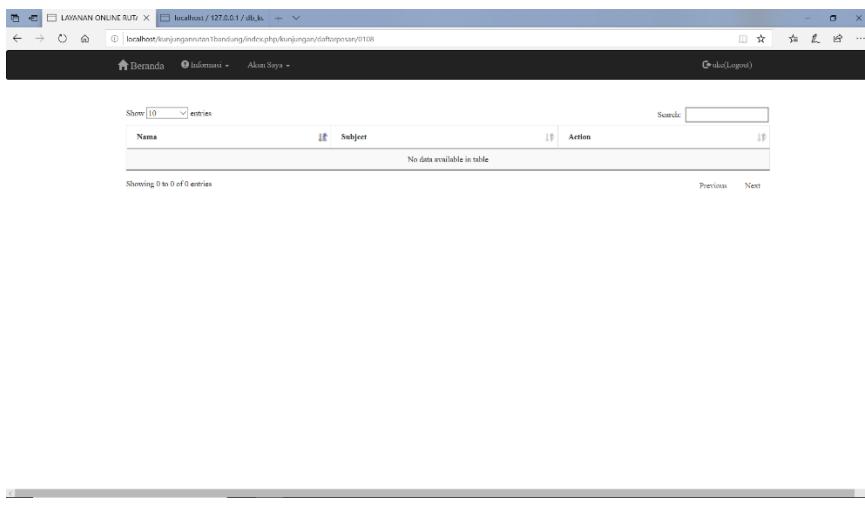
7.5.5 Formulir Kunjungan

A screenshot of a web browser displaying a visitor registration form. The title bar shows 'LAYANAN ONLINE RUT/ X localhost / 127.0.0.1 / db_k...'. The main content area has a header 'WBP YANG DIKUNJUNGI'. Below it is a form with fields: 'Pilih Tanggal' (date), 'Pilih Jadwal' (time), 'Nama WBP' (visitor name) with a photo placeholder, 'pasal' (article), and 'Blok Set tahanan' (detention block). At the bottom is a large blue 'Kirim' button.

Gambar 7.22 Halaman Formulir Pendaftaran

Di halaman ini Pengunjung Melakukan Pengisian formulir pendaftaran

7.5.6 Cek Pesan



Nama	Subject	Action
No data available in table		

Gambar 7.23 Halaman Cek Pesan Dari Admin

Di halaman ini Pengunjung jika memiliki keluhan akan dibalas pesan oleh admin

BAB VIII KESIMPULAN

8.1 KESIMPULAN MASALAH

Berdasarkan penelitian yang telah dilakukan penelitian dengan latar belakang masalah mengenai pendaftaran online kunjungan dengan meningkatkan kemananran rutan klas 1 bandung, dimana sering terjadinya penyeludupan dan lamanya proses pendaftaran kunjungan sehingga memerlukan waktu 15 menit untuk setiap orangnya yang cukup memakan waktu yang lama, dari masalah tersebut dapat ditemukan bahwa penelitian ini dapat menjawab atau memberikan solusi terhadap masalah yang ada. Dengan menggunakan metode *Boyer Moore* sebagai fitur untuk menentukan diterima atau ditolaknya pendaftaran tersebut memberikan hasil yang lebih baik.

8.2 KESIMPULAN METODE

Setelah melakukan analisis sistem, dapat disimpulkan bahwa:

1. Membangun aplikasi Kunjungan Online Rutan Klas 1 Bandung berbasis website menggunakan Bahasa pemrograman *PHP*, *Database MySQL* dan untuk meningkatkan keamanan rutan klas 1 bandung yaitu dengan cara menempelkan Metode Algoritma *Boyer Moore* kedalam form inputan pengunjung dengan mencocokan data dengan *database* rutan klas 1 bandung dengan cara melakukan pergeseran setiap karakter yang akan dicocokan sampai dinyatakan *String Matching*, dan jika tidak ada kecocokan maka akan dinyatakan data yang di inputkan tidak memiliki kesamaan *string*.
2. Menentukan Pendaftaran di terima atau tidak yaitu dengan cara Memasukan metode algoritma *Boyer Moore* Kedalam Form Inputan WBP yang akan di kunjungi pada halaman *input* data kunjungan, dimana Inputan tersebut akan dicocokan dengan *database* Rutan Klas 1 Bandung sesuai dengan identitas WBP yang ada, jika inputan tersebut sesuai dengan *Database* Rutan akan dinyatakan bahwa pendaftaran

kunjungan tersebut Diterima dan jika data inputan tersebut tidak sesuai maka pendaftaran kunjungan akan otomatis ditolak oleh sistem karena tidak ada rentan kesamaan kata dalam nama wbp tersebut.

3. Cara algoritma *Boyer Moore* dalam melakukan *searching, retrieve* yaitu langkah pertama algoritma *boyer moore* akan mencari huruf paling kanan yang sama dan akan melakukan terus menerus pergeseran jika text tersebut belum di temukan, jika sudah ditemukan algoritma *boyer moore* akan berhenti melakukan pencocokan *string*. Dan jika tidak ada yang sama dari keseluruhan data algoritma *Boyer Moore* Pun akan berhenti dan menyatakan bahwa kalimat yang di cari tidak ada yang sesuai.

8.3 KESIMPULAN PENGUJIAN SISTEM

Setelah melakukan beberapa tahapan dan melakukan implementasi metode pada sistem, menyimpulkan bahwa hasil implementasi *Boyer Moore* pada sistem sesuai dengan pergeseran manual pada *string* yang digunakan, Jika *pattern* pada *database* dan text yang di inputkan sesuai maka pendaftaran kunjungan diverifikasi, dan jika pendaftaran kunjungan tidak sesuai maka hasilnya akan ditolak

id_wbp	nama_dan_bin	pasal	blok_kamar
5	ASEP BIN TATANG	12	24
6	JAMAL	12	24

Gambar7.1 Database WBP

Gambar tersebut merupakan data tahanan yang tersedia pada database yang akan dicocokan dengan text yang akan diinputkan oleh pengunjung

History Pendaftaran							
Tanggal Pendaftaran	Nomor Antrian	Jam Kunjungan	Status	Keterangan	Nama Napi	Pasal	Blok
2019-12-25 22:38:51	2019122614	09:00:00 - 10:00:00	diverifikasi	Info napi yang diinputkan terdapat di database	JAMAL	12	24

Gambar7.2 Kunjunga Diverifikasi

Gambar tersebut merupakan data registrasi yang di verifikasi karena memiliki kecocokan data dengan *database sistem*

status	keterangan	wbpsertabin	pasal	blok
ditolak	Ditolak, blok dari Napi yang Anda inputkan tidak C...	A	ASD	ASD
diverifikasi	Info napi yang diinputkan terdapat di database	JAMAL	12	24
diverifikasi	Info napi yang diinputkan terdapat di database	JAMAL	12	24
diverifikasi	Info napi yang diinputkan terdapat di database	JAMAL	12	24
diverifikasi	Info napi yang diinputkan terdapat di database	ASEP JAMAL BIN TATANG	12	24

Gambar7.3 Daftar Kunjungan Kunjungan

Gambar tersebut merupakan data kunjungan, dapat dilihat pada status ditolak dikarenakan pada database tidak terdapat WBP atas nama A pasal ASD dan Blok ASD sehingga pendaftaran kunjungan tersebut ditolak.