

**APLIKASI PENDAFTARAN SEMINAR MENGGUNAKAN METODE
MVC BERBASIS *WEBSITE* MENGGUNAKAN *FRAMEWORK*
*CODEIGNITER 3.1.10***

SKRIPSI



Disusun Oleh :

NAMA : DERMAWAN

NIM : 20151000093

TEKNIK INFORMATIKA

**FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS BUDDHI DHARMA
TANGERANG**

2020

**APLIKASI PENDAFTARAN SEMINAR MENGGUNAKAN METODE
MVC BERBASIS *WEBSITE* MENGGUNAKAN *FRAMEWORK*
*CODEIGNITER 3.1.10***

Skripsi

**Diajukan sebagai salah satu syarat untuk kelengkapan gelar kesarjanaan pada
Program Studi Teknik Informatika
Jenjang Pendidikan Strata 1**



Disusun Oleh :

NAMA : DERMAWAN

NIM : 20151000093

TEKNIK INFORMATIKA

**FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS BUDDHI DHARMA
TANGERANG**

2020

LEMBAR PERSEMBAHAN

If You Don't Give Up, You Still Have A Chance. Giving Up Is The Greatest Failure
*“Jika Anda Tidak Menyerah, Anda Masih Memiliki Kesempatan. Menyerah Adalah
Kegagalan Terbesar ”*

-Jack Ma-

Dengan mengucapkan puji syukur kepada Tuhan Yang Maha Esa, Skripsi ini kupersembahkan untuk :

1. Bapak Sin An dan Ibu Ong Yun Wah tercinta yang telah membesarkan aku dan selalu membimbing, mendukung, memotivasi, memberi apa yang terbaik bagiku serta selalu mendoakan aku untuk meraih kesuksesanku.
2. Kakakku yang telah memberikan dukungan semangat serta dorongan yang senantiasa diberikan.
3. Teman-teman kelompok belajar yang bernama Virdi Gunawan dan Andrey Anggata yang selalu berjuang bersama.

UNIVERSITAS BUDDHI DHARMA

LEMBAR PENYATAAN KEASLIAN SKRIPSI

Yang bertanda tangan di bawah ini,

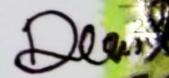
NIM : 20151000093
Nama : Dermawan
Jenjang Studi : Strata 1
Program Studi : Teknik Informatika
Peminatan : Basis Data

Dengan ini saya menyatakan bahwa :

1. Skripsi ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar akademik (Diploma/Sarjana) atau kelengkapan studi, baik di Universitas Buddhi Dharma maupun di Perguruan Tinggi lainnya.
2. Skripsi ini saya buat sendiri tanpa bantuan pihak lain, kecuali arahan dosen pembimbing.
3. Dalam Skripsi ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dan dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dicantumkan daftar pustaka.
4. Dalam Skripsi ini tidak terdapat pemalsuan (kebohongan), seperti : buku, artikel, jurnal, pengolahan data, dan pemalsuan tanda tangan dosen atau Ketua Program Studi di Universitas Buddhi Dharma yang dibuktikan dengan keasliannya.
5. Lembar Pernyataan ini saya buat dengan sesungguhnya, tanpa paksaan dan apabila di kemudian hari atau pada waktu lainnya terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah saya peroleh karena Skripsi ini, serta sanksi lainnya sesuai dengan peraturan dan norma yang berlaku

Tangerang, 20 Januari 2020

Penulis,



(Dermawan)
20151000093

UNIVERSITAS BUDDHI DHARMA

LEMBAR PERSETUJUAN PUBLIKASI KARYA ILMIAH

NIM : 20151000093
Nama : Dermawan
Jenjang Studi : Strata 1
Program Studi : Teknik Informatika
Peminatan : Basis Data

Dengan ini menyetujui untuk memberikan izin kepada pihak Universitas Buddhi Dharma Fakultas Sains & Teknologi. Hak bebas royalti Non-eksklusif (*Non-exclusif Royalty-Free Right*) atas karya ilmiah saya yang berjudul "**Aplikasi Pendaftaran Seminar Menggunakan Metode MVC Berbasis Website Menggunakan Framework Codeigniter 3.1.10**"

Dengan Hak Bebas Royalti Non-Eksklusif ini pihak Universitas Buddhi Dharma berhak menyimpan, mengalih media atau format-kan, mengelolanya dalam pangkalan data (*database*), mendistribusikannya dan menampilkan atau mempublikasikannya di *internet* atau media lain untuk kepentingan akademis tanpa perlu meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis atau pencipta karya ilmiah tersebut.

Saya bersedia untuk menanggung secara pribadi, tanpa melibatkan pihak Universitas Buddhi Dharma, segala bentuk tuntutan hukum yang timbul atas Hak Cipta atas karya ilmiah saya ini.

Demikian pernyataan ini saya buat dengan sebenarnya.

Tangerang, 20 Januari 2020

Penulis,



UNIVERSITAS BUDDHI DHARMA

LEMBAR PENGESAHAN PEMBIMBING
APLIKASI PENDAFTARAN SEMINAR MENGGUNAKAN METODE
MVC* BERBASIS *WEBSITE* MENGGUNAKAN *FRAMEWORK

CODEIGNITER 3.1.10

Dibuat Oleh :

NIM : 20151000093

Nama : Dermawan

Telah disetujui untuk dipertahankan di hadapan Tim Penguji Ujian

Komprehensif

Program Studi Teknik Informatika

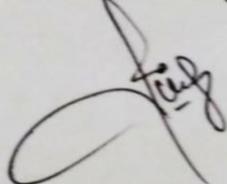
Peminatan Basis Data

20191

Tangerang, 20 Desember 2019

Disahkan oleh,

Pembimbing,



Dicky Surya Dwi Putra, M.Kom.

NIDN. 0427068701

UNIVERSITAS BUDDHI DHARMA

LEMBAR PENGESAHAN SKRIPSI

**APLIKASI PENDAFTARAN SEMINAR MENGGUNAKAN METODE
MVC BERBASIS WEBSITE MENGGUNAKAN FRAMEWORK
CODEIGNITER 3.1.10**

Dibuat Oleh :

NIM : 20151000093

Nama : Dermawan

Telah disetujui untuk dipertahankan di hadapan Tim Penguji Ujian
Komprehensif

Program Studi Teknik Informatika

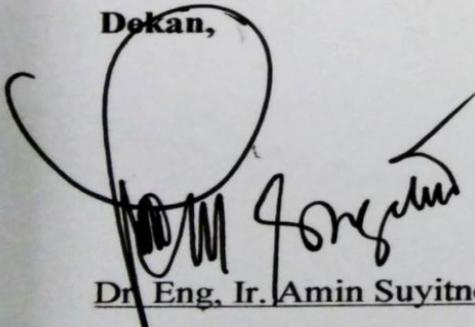
Peminatan Basis Data

20191

Tangerang, 20 Januari 2020

Disahkan oleh,

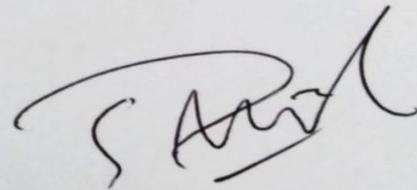
Dekan,



Dr. Eng. Ir. Amin Suyitno, M.Eng.

NIDN. 9906000711

Ketua Program Studi



Rino, M.Kom.

NIDN. 0420058502

LEMBAR PENGESAHAN TIM PENGUJI

Nama : Dermawan
NIM : 20151000093
Fakultas : Sains dan Teknologi
Judul Skripsi : APLIKASI PENDAFTARAN SEMINAR MENGGUNAKAN METODE
MVC BERBASIS WEBSITE MENGGUNAKAN FRAMEWORK
CODEIGNITER 3.1.10

Dinyatakan LULUS setelah mempertahankan di depan Tim Penguji pada hari 20 Januari 2020.

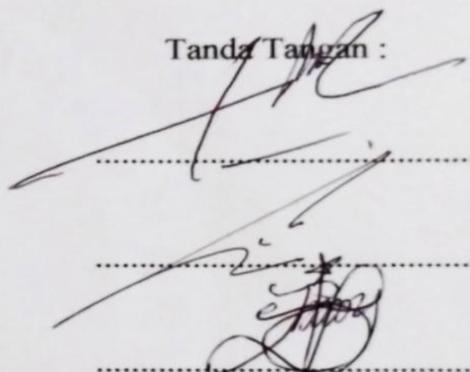
Nama Penguji :

Ketua Sidang : **Rudy Arijanto, M.Kom.**
NIDN. 0415077105

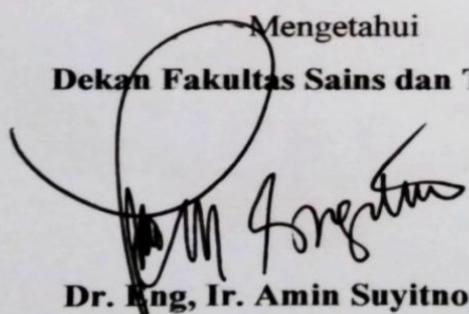
Penguji I : **Dram Renaldi, M.Kom.**
NIDN. 0411019001

Penguji II : **Benny Daniawan, M.Kom.**
NIDN. 0424049006

Tanda Tangan :



Mengetahui
Dekan Fakultas Sains dan Teknologi



Dr. Eng, Ir. Amin Suyitno, M.Eng.
NIDN. 9906000711

KATA PENGANTAR

Dengan mengucapkan Puji Syukur kepada Tuhan Yang Maha Esa, yang telah memberikan Rahmat dan karunia-Nya kepada penulis sehingga dapat menyusun dan menyelesaikan Skripsi ini dengan judul “**Aplikasi Pendaftaran Seminar Menggunakan Metode MVC Berbasis Website Menggunakan Codeigniter 3.1.10**”. Tujuan utama dari pembuatan Skripsi ini adalah sebagai salah satu syarat kelengkapan dalam menyelesaikan program pendidikan Strata 1 Program Studi Teknik Informatika di Universitas Buddhi Dharma. Dalam penyusunan Skripsi ini penulis banyak menerima bantuan dan dorongan baik moril maupun materiil dari berbagai pihak, maka pada kesempatan ini penulis menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. Bapak Dr. Sofian Sugioko, M.M., CPMA., sebagai Rektor Universitas Buddhi Dharma.
2. Bapak Dr. Eng, Ir. Amin Suyitno, M.Eng., Dekan Fakultas Sains dan Teknologi
3. Bapak Rudy Arijanto, M.Kom., sebagai Wakil Dekan Fakultas Sains dan Teknologi
4. Bapak Rino, M. Kom., sebagai Ketua Program Studi Teknik Informatika
5. Bapak Dicky Surya Dwi Putra, M. Kom. Sebagai pembimbing yang telah membantu dan memberikan dukungan serta harapan untuk menyelesaikan penulisan Skripsi ini.
6. Orang tua dan keluarga yang selalu memberikan dukungan baik moril dan materiil.
7. Teman-teman yang selalu membantu dan memberikan semangat

Serta semua pihak yang terlalu banyak untuk disebutkan satu-persatu sehingga terwujudnya penulisan ini. Penulis menyadari bahwa penulisan Skripsi ini masih belum sempurna, untuk itu penulis mohon kritik dan saran yang bersifat membangun demi kesempurnaan penulisan di masa yang akan datang. Akhir kata semoga Skripsi ini dapat berguna bagi penulis khususnya dan bagi para pembaca yang berminat pada umumnya.

Tangerang, 20 Desember 2019

Penulis

Aplikasi Pendaftaran Seminar Menggunakan Metode *MVC* Berbasis *Website* Menggunakan *Framework Codeigniter 3.1.10*
93 + xi halaman/ 18 tabel / 53 gambar/ 2 Lampiran

ABSTRAK

Seminar adalah sebuah pertemuan yang bertujuan untuk membahas suatu masalah secara ilmiah. Perkembangan teknologi informasi dan komunikasi semakin berkembang pesat, hampir semua seluruh komponen dan institusi negeri maupun swasta bahkan kalangan masyarakat umum telah menggunakan teknologi informasi sebagai sarana komunikasi. Teknologi internet yang saat ini kian marak dan berkembang pesat adalah teknologi internet. *Website* merupakan kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, video, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait di mana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*). *Framework* adalah kerangka kerja. *Framework* juga dapat diartikan sebagai kumpulan *scrip* (terutama *class* dan *function*) yang dapat membantu *developer/programmer* dalam menangani berbagai masalah-masalah dalam pemrograman, seperti koneksi ke *database*, pemanggilan *variable*, *file*, dan lain-lain sehingga pekerjaan *developer* lebih fokus dan lebih cepat dalam membangun aplikasi. *Codeigniter* merupakan sebuah *web framework* yang dikembangkan oleh Rick Ellis dari Ellis Lab. *Codeigniter* dirancang untuk menjadi sebuah *web framework* yang ringan dan mudah untuk digunakan. Pola *MVC* memecahkan sebuah aplikasi menjadi tiga modul asosiasi: *model*, *view*, dan *controller*. Dengan *framework codeigniter* aplikasi *web* pendaftaran seminar menjadi *web* dinamis. Dengan metode *mvc* proses pembuatan aplikasi pendaftaran seminar dapat diimplementasikan. Dengan adanya aplikasi pendaftaran seminar ini, Dapat membantu dan mempermudah pengguna dalam mendapatkan informasi tentang seminar dan juga dalam melakukan pendaftaran seminar lalu dapat mencetak sertifikat secara langsung.

Kata Kunci : Pendaftaran Seminar, *MVC*, *Framework Codeigniter*, *Website*.

ABSTRACT

The seminar is a meeting that aims to discuss a problem scientifically. The development of information and communication technology is growing rapidly, almost all all components and public and private institutions and even the general public has used information technology as a means of communication. Internet technology that is currently increasingly widespread and growing rapidly is internet technology. Website is a collection of pages that display information on text data, still or motion picture data, animated data, sound, video, and or a combination of all, both static and dynamic that form a series of interrelated buildings where each is connected with page networks (hyperlinks). Framework is a framework. Framework can also be interpreted as a collection of scrip (especially classes and functions) that can help developers / programmers in handling various problems in programming, such as connection to the database, calling variables, files, etc. so that the work of the developer is more focused and faster in building applications. Codeigniter is a web framework developed by Rick Ellis of Ellis Lab. Codeigniter is designed to be a lightweight and easy-to-use web framework. The MVC pattern breaks an application into three association modules: model, view, and controller. With the codeigniter framework the seminar registration web application becomes a dynamic web. With the mvc method the process of making a seminar registration application can be implemented. With this seminar registration application, it can help and facilitate users in getting information about the seminar and also in conducting seminar registration and then can print certificates directly.

Key Word : *Registration for Seminars, MVC, Codeigniter Framework, Website.*

DAFTAR ISI

Halaman:

LEMBAR JUDUL DALAM

LEMBAR PERSEMBAHAN

LEMBAR PENYATAAN KEASLIAN SKRIPSI

LEMBAR PERSETUJUAN PUBLIKASI KARYA ILMIAH

LEMBAR PENGESAHAN PEMBIMBING

LEMBAR PENGESAHAN SKRIPSI

LEMBAR PENGESAHAN TIM PENGUJI

KATA PENGANTAR	i
ABSTRAK.....	ii
<i>ABSTRACT</i>	iii
DAFTAR ISI	iv
DAFTAR TABEL	viii
DAFTAR GAMBAR.....	ix
DAFTAR LAMPIRAN	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Identifikasi Masalah	3
1.3 Rumusan Masalah	3
1.4 Tujuan dan Manfaat Penulisan	4
1.4.1 Tujuan	4
1.4.2 Manfaat	4
1.5 Ruang Lingkup Masalah.....	4
1.6 Metode Penelitian dan Pengumpulan Data.....	5
1.6.1 Metode Penelitian	5

1.6.2	Teknik Pengumpulan Data.....	5
1.7	Sistematika Penulisan.....	6
BAB II	LANDASAN TEORI.....	7
2.1	Teori Umum.....	7
2.1.1	Data.....	7
2.1.2	Informasi.....	7
2.1.3	Internet.....	9
2.1.4	Basis Data.....	10
2.2	Teori Khusus.....	11
2.2.1	Aplikasi.....	11
2.2.2	Seminar.....	13
2.2.3	<i>Website</i>	13
2.2.4	<i>Framework</i>	14
2.2.5	<i>MVC</i>	15
2.3	Teori Analisa dan Perancangan.....	16
2.3.1	<i>Entity Relationship Diagram (ERD)</i>	16
2.3.2	<i>Flowchart</i>	18
2.3.3	<i>Browser</i>	21
2.3.4	<i>MySQL</i>	22
2.3.5	<i>Codeigniter</i>	26
2.3.6	<i>Visual Studio Code</i>	28
2.3.7	<i>XAMPP</i>	29
2.3.8	<i>PhpMyAdmin</i>	29
2.3.9	<i>HTML (Hyper Text Markup Language)</i>	30
2.3.10	<i>CSS (Cascading Style Sheet)</i>	32
2.3.11	<i>Bootstrap</i>	34
2.3.12	<i>PHP (Hypertext Preprocessor)</i>	35

2.3.13	<i>Java Script</i>	37
2.3.14	<i>Blackbox Testing</i>	38
2.3.15	<i>Whitebox Testing</i>	39
2.4	Tinjauan Pustaka	41
2.4.1	Penelitian Pastima Simanjuntak dan Arwin Kasnady.....	41
2.4.2	Penelitian Melda Agarina, Arman Suryadi Karim.....	42
2.4.3	Penelitian Dwi Arianto Adi Nugroho, Heru Supriyono	43
2.4.4	Rangkuman Model Penelitian.....	43
2.5	Kerangka Pemikiran	45
BAB III ANALISA MASALAH DAN PERANCANGAN APLIKASI.....		46
3.1	Analisa Kebutuhan	46
3.1.1	Analisa kebutuhan membangun sistem.....	46
3.1.2	Analisa Kebutuhan Pemakai	47
3.1.3	Analisa Kebutuhan Aplikasi	47
3.2	Alur Kerja <i>Framework Codeigniter</i>	48
3.3	Perancangan Layar, Menu, <i>Database</i>	49
3.3.1	Perancangan Layar dan Menu.....	49
3.3.2	Rancangan <i>Database</i>	58
3.3.3	Rancangan <i>Flowchart</i>	60
BAB IV PENGUJIAN DAN IMPLEMENTASI.....		72
4.1	Pembahasan Cara Kerja <i>Codeigniter</i>	72
4.1.1	Struktur Direktori <i>Codeigniter</i>	72
4.1.2	<i>MVC</i> Pada <i>Codeigniter</i>	73
4.2	Spesifikasi <i>Hardware</i> dan <i>Software</i>	78
4.2.1	Spesifikasi <i>Hardware</i>	78
4.2.2	Spesifikasi <i>Software</i>	79
4.3	Tampilan Program	79

4.3.1	Tampilan Login.....	79
4.3.2	Tampilan Register.....	80
4.3.3	Tampilan Dashboard Admin.....	81
4.3.4	Tampilan List Seminar Admin	81
4.3.5	Tampilan Tambah Seminar.....	82
4.3.6	Tampilan Ubah Seminar	82
4.3.7	Tampilan Detail Pendaftar Seminar.....	83
4.3.8	Tampilan Beranda <i>User</i>	83
4.3.9	Tampilan Daftar Transfer	84
4.3.10	Tampilan Daftar Cash.....	84
4.3.11	Tampilan Cetak Absen Seminar	85
4.3.12	Tampilan Cetak Kehadiran	85
4.3.13	Tampilan List Seminar User	86
4.3.14	Tampilan ReUpload Pembayaran	86
4.3.15	Tampilan Cetak Nomor Pendaftaran	87
4.3.16	Tampilan Cetak Sertifikat.....	87
4.4	Pengujian Aplikasi.....	88
4.4.1	<i>Blackbox Testing</i>	88
4.4.2	<i>Whitebox Testing</i>	93
BAB V SIMPULAN DAN SARAN		95
5.1	Simpulan.....	95
5.2	Saran	95
DAFTAR PUSTAKA		
LAMPIRAN-LAMPIRAN		
DAFTAR RIWAYAT HIDUP		

DAFTAR TABEL

Tabel 2-1: Tipe Data	22
Tabel 2-2: Tipe data <i>Numeric</i>	23
Tabel 2-3: Tipe data <i>String</i>	24
Tabel 2-4: Tipe data Tanggal.....	25
Tabel 2-5: Penelitian Pastima Simanjuntak, Arwin Kasnady.....	41
Tabel 2-6: Penelitian Melda Agarina, Arman Suryadi Karim	42
Tabel 2-7 Penelitian Dwi Arianto Adi Nugroho, Heru Supriyono	43
Tabel 2-8: Rangkuman Model Penelitian	43
Tabel 3-1: Analisa Kebutuhan Pemakai	47
Tabel 3-2: Analisa Kebutuhan Aplikasi	47
Tabel 3-3: tbl_user.....	59
Tabel 3-4: tbl_seminar.....	59
Tabel 3-5: tbl_pendaftaran.....	59
Tabel 4-1: Kebutuhan Perangkat Keras Untuk Pengembangan	78
Tabel 4-2: Kebutuhan Perangkat Keras Untuk Pengguna	78
Tabel 4-3: Kebutuhan Perangkat Lunak Untuk Pengembang	79
Tabel 4-4 : Kebutuhan Perangkat Lunak Untuk Pengguna	79
Tabel 4-5: Hasil Pengujian <i>Blackbox</i>	88

DAFTAR GAMBAR

Gambar 2-1: <i>MVC Pattern</i>	15
Gambar 2-2: Bentuk <i>Entity</i>	17
Gambar 2-3: Bentuk <i>Atribut</i>	17
Gambar 2-4: Bentuk <i>Line</i>	17
Gambar 2-5: Bentuk <i>Relasi</i>	17
Gambar 2-6: Simbol <i>Flowchart</i>	21
Gambar 2-7: Logo <i>MySQL</i>	25
Gambar 2-8: Logo <i>Codeigniter</i>	26
Gambar 2-9: Logo <i>Visual Studio Code</i>	28
Gambar 2-10: Logo <i>XAMPP</i>	29
Gambar 2-11: Logo <i>HTML</i>	32
Gambar 2-12: Logo <i>CSS</i>	34
Gambar 2-13: <i>Official Website Bootstrap</i>	35
Gambar 2-14: Contoh penulisan <i>PHP</i>	37
Gambar 2-15: Logo <i>PHP</i>	37
Gambar 2-16: Logo <i>Java Script</i>	38
Gambar 2-17: Kerangka Pemikiran.....	45
Gambar 3-1: Alur Kerja <i>Codeigniter</i>	48
Gambar 3-2: Rancangan Layar <i>Login</i>	49
Gambar 3-3: Rancangan Layar Daftar Pengguna.....	50
Gambar 3-4: Rancangan Layar <i>Dashboard Admin</i>	51
Gambar 3-5: Rancangan Layar Seminar.....	51
Gambar 3-6: Rancangan Tambah Seminar.....	52
Gambar 3-7: Rancangan Layar Ubah Seminar.....	53
Gambar 3-8: Rancangan Detail Pendaftar Seminar.....	54
Gambar 3-9: Rancangan Tampilan Halaman Utama <i>User</i>	55
Gambar 3-10: Rancangan Daftar Seminar.....	56
Gambar 3-11: Rancangan List Seminar <i>User</i>	57
Gambar 3-12: Rancangan <i>ReUpload</i> Foto Pembayaran.....	58
Gambar 3-13: Logical Record Structure (LRS).....	58
Gambar 3-14: <i>Flowchart</i> Proses Login.....	61
Gambar 3-15: <i>Flowchart</i> Proses register.....	63

Gambar 3-16: <i>Flowchart</i> Proses Tambah, Ubah, Detail	65
Gambar 3-17: <i>Flowchart</i> Proses <i>Report Absen, Report Kehadiran, Pembayaran, Kehadiran, Nomor Sertifikat</i>	67
Gambar 3-18: <i>Flowchart</i> Proses <i>Daftar User</i>	69
Gambar 3-19: <i>Flowchart</i> Proses <i>Cetak Tiket, Cetak Sertifikat, ReUpload Pembayaran</i>	71
Gambar 4-1: Struktur Direktori <i>Codeigniter</i>	72
Gambar 4-2: Tampilan Login	80
Gambar 4-3: Tampilan Register	80
Gambar 4-4: Tampilan Dashboard Admin	81
Gambar 4-5: Tampilan List Seminar Admin	81
Gambar 4-6: Tampilan Tambah Seminar	82
Gambar 4-7: Tampilan Ubah Seminar	82
Gambar 4-8: Tampilan Detail Pendaftar Seminar	83
Gambar 4-9: Tampilan Beranda <i>User</i>	83
Gambar 4-10: Tampilan Daftar Transfer	84
Gambar 4-11: Tampilan Daftar Transfer	84
Gambar 4-12: Tampilan Cetak Absen Seminar	85
Gambar 4-13: Tampilan Cetak Kehadiran	85
Gambar 4-14: Tampilan List Seminar <i>User</i>	86
Gambar 4-15: Tampilan <i>ReUpload Pembayaran</i>	86
Gambar 4-16: Tampilan Cetak Nomor Pendaftaran	87
Gambar 4-17: Tampilan Cetak Sertifikat	87

DAFTAR LAMPIRAN

	Halaman
Lampiran A-1 : <i>Requiremet Elicitation</i>	97
Lampiran A-2 : Jurnal Pendukung	97

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Nugroho dan Supriyono mengemukakan bahwa: “Seminar adalah sebuah pertemuan yang bertujuan untuk membahas suatu masalah secara ilmiah. Untuk menghadiri acara seminar, beberapa seminar menggunakan tiket sebagai akses masuk. Tiket adalah sebuah tanda masuk yang digunakan untuk mendatangi atau menghadiri sebuah tempat/acara tertentu. Dalam sebuah acara seminar, peserta harus mendaftar terlebih dahulu untuk mendapatkan tiket yang digunakan untuk masuk kedalam acara seminar dan tiket tersebut masih menggunakan tiket dalam bentuk kertas atau yang biasa disebut dengan tiket konvensional” (Nugroho dan Supriyono 2019, 34).

Agarina dan Karim mengemukakan bahwa: “Perkembangan teknologi informasi dan komunikasi semakin berkembang pesat, hampir semua semua seluruh komponen dan institusi negeri maupun swasta bahkan kalangan masyarakat umum telah menggunakan teknologi informasi sebagai sarana komunikasi. Teknologi internet yang saat ini kian marak dan berkembang pesat adalah teknologi internet. Teknologi internet yang sering diaplikasikan dalam bentuk basis *Web* semakin lama semakin dikembangkan untuk berbagai keperluan dan kebutuhan” (Agarina dan Karim 2019, 26).

Murhada dan Yo Ceng Giap mengungkapkan bahwa: “Perangkat lunak aplikasi sejatinya adalah program yang didesain khusus untuk menyelesaikan masalah spesifik pengguna, seperti melakukan tugas-tugas mengetik dokumen, manipulasi foto, merancang bangunan” (Murhada dan Yo 2011, 104)

Fathurrahman mengungkapkan bahwa: “*Website* merupakan kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, video, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait di mana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*)” (Fathurrahman 2014, 2).

Yudhanto dan Prasetyo mengungkapkan bahwa: “*Framework* adalah kerangka kerja. *Framework* juga dapat diartikan sebagai kumpulan *scrip* (terutama *class* dan *function*) yang dapat membantu *developer/programmer* dalam menangani berbagai masalah-masalah dalam pemrograman, seperti koneksi ke *database*, pemanggilan *variable*, *file*, dan lain-lain sehingga pekerjaan *developer* lebih focus dan lebih cepat dalam membangun aplikasi. *Framework* adalah komponen pemrograman yang siap digunakan kapan saja sehingga programmer tidak harus membuat *script* yang sama untuk tugas yang sama” (Yudhanto dan Prasetyo 2018, 8).

Simanjuntak dan Kasnady mengungkapkan bahwa: “Pola *MVC* memecahkan sebuah aplikasi menjadi tiga modul asosiasi: *model*, *view*, dan *controller*. *Model* adalah logika bisnis dari aplikasi dan inti dari sebuah aplikasi. *View* adalah *user interface* dari *controller* yang merupakan muka umum untuk respon *event* pengguna. Komponen *controller* mengimplementasi *flow* yang mengontrol antara *view* dan *model*” (Simanjuntak dan Kasnady 2016, 58).

Subagia mengungkapkan bahwa: “*Codeigniter* merupakan sebuah *web framework* yang dikembangkan oleh Rick Ellis dari Ellis Lab. *Codeigniter* dirancang untuk menjadi sebuah *web framework* yang ringan dan mudah untuk digunakan. Bahkan pengakuan dari Rasmus Lerdorf, sang pencipta Bahasa pemrograman PHP

mengatakan bahwa Codeigniter merupakan *web framework* yang mudah, cepat dan handal” (Subagia 2018, 2).

Berdasarkan pada hal-hal yang dijabarkan di atas, maka dibangun sebuah aplikasi pendaftaran seminar berbasis web dengan metode (MVC), sehingga seseorang akan mudah dalam pendaftaran seminar tanpa harus datang ke tempat penyelenggara seminar tersebut lalu dipilih judul “**APLIKASI PENDAFTARAN SEMINAR MENGGUNAKAN METODE MVC BERBASIS WEBSITE MENGGUNAKAN FRAMEWORK CODEIGNITER 3.1.10**”.

1.2 Identifikasi Masalah

Berdasarkan dari latar belakang yang telah penulis jelaskan di atas, maka yang menjadi identifikasi masalah dalam penelitian ini adalah dapatkah metode MVC mengubah pendaftaran seminar konvensional menjadi pendaftaran seminar dengan sistem yang membantu memudahkan pengguna.

1.3 Rumusan Masalah

Dalam penulisan ini ada rumusan masalah yang diangkat adalah sebagai berikut:

1. Dapatkah metode MVC diimplementasi dalam proses pengerjaan aplikasi ini?
2. Seberapa efektif penggunaan metode *MVC* pada *Framework Codeigniter* yang membantu pengembang dalam perancangan aplikasi?

1.4 Tujuan dan Manfaat Penulisan

1.4.1 Tujuan

Tujuan dari penelitian ini adalah sebagai berikut:

- a. Menerapkan metode *MVC* dalam aplikasi berbasis *website*.
- b. Memperoleh informasi berupa tingkat efektif dan tingkat efisien dalam penggunaan metode *MVC* pada *Framework Codeigniter* dalam perancangan aplikasi.

1.4.2 Manfaat

Manfaat dari penelitian ini adalah sebagai berikut:

- a. Menjadi referensi dalam menggunakan metode *MVC* dalam aplikasi berbasis *website*.
- b. Diharapkan dapat menjadi aplikasi berbasis *website* yang dapat membantu pengguna dalam pendaftaran seminar serta pengambilan sertifikat seminar.

1.5 Ruang Lingkup Masalah

Ruang lingkup yang terdapat dalam penelitian ini adalah sebagai berikut :

- a. Aplikasi ini dirancang menggunakan *Framework Codeigniter 3.1.10*.
- b. Aplikasi ini dirancang untuk membantu pengguna dalam pendaftaran seminar serta pengambilan sertifikat seminar.
- c. Pembuatan dalam aplikasi ini menggunakan metode *MVC* dan *Framework*.
- d. Aplikasi ini dapat memberikan laporan terhadap pengguna dengan *level* admin berupa: absensi peserta seminar.
- e. Aplikasi ini dengan *level user* dapat mencetak sertifikat.

1.6 Metode Penelitian dan Pengumpulan Data

1.6.1 Metode Penelitian

a. Perencanaan

Tahap ini diawali dengan memilih topik, judul serta aplikasi yang akan dibangun, pada tahap ini melakukan pencarian data dan informasi yang dibutuhkan dalam penelitian sehingga tidak terjadi hambatan dalam penelitian.

b. Analisis

Untuk dapat membantu pengembangan aplikasi, di lakukan analisa melalui beberapa media baik dari media cetak ataupun media elektronik, sehingga dapat membantu pengembangan aplikasi.

c. Desain

Di tahap ini untuk dapat menyelesaikan masalah yang dihadapi, dilakukan perancangan aplikasi mulai dari identifikasi masalah, analisa kebutuhan sistem, perancangan alur sistem, dan perancangan layar atau tampilan aplikasi tersebut.

d. Implementasi

Pada tahap ini melakukan percobaan terhadap aplikasi yang sudah dibuat apakah aplikasi sudah sesuai dengan kriteria yang dikehendaki atau belum, jika sudah sesuai maka aplikasi siap di implementasikan.

1.6.2 Teknik Pengumpulan Data

a. Studi Pustaka

Pengumpulan data dan informasi yang diperlukan untuk digunakan dalam pembuatan aplikasi ini adalah metode *MVC* dan juga *Framework Codeigniter*.

1.7 Sistematika Penulisan

BAB I PENDAHULUAN

Pendahuluan berisi mengenai latar belakang, identifikasi masalah, rumusan masalah, tujuan dan manfaat penulisan, ruang lingkup masalah, metode penelitian dan pengumpulan data, sistematika penulisan.

BAB II LANDASAN TEORI

Landasan teori mengenai penelitian terdahulu dan rencana penelitian yang dilakukan oleh penulis.

BAB III ANALISA MASALAH DAN PERANCANGAN APLIKASI

Metodelogi berisi mengenai metodelogi atau langkah-langkah dalam penyelesaian masalah, meliputi penyusunan formula dan algoritma yang akan digunakan dalam penelitian.

BAB IV PENGUJIAN DAN IMPLEMENTASI

Bab ini menjelaskan analisa sistem yang diusulkan dengan menggunakan flowchart dan mind map dari sistem yang diimplementasikan, serta pembahasan secara detail final elisitasi yang ada di bab sebelumnya, di jabarkan secara satu persatu dengan menerapkan konsep sesudah adanya sistem yang diusulkan.

BAB V SIMPULAN DAN SARAN

Bab ini berisi kesimpulan dan saran yang berkaitan dengan analisa dan optimalisasi sistem berdasarkan yang telah diuraikan pada bab-bab sebelumnya.

BAB II

LANDASAN TEORI

2.1 Teori Umum

2.1.1 Data

Menurut Fathansyah mengemukakan bahwa “Data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya yang diwujudkan dalam bentuk, huruf, simbol, teks, gambar, bunyi atau kombinasinya” (Fathansyah 2015, 2).

Menurut Sugiyono (Sugiyono 2013, 137) terdapat 2 jenis data yaitu:

a. Data Primer

Data yang langsung diperoleh secara langsung meliputi dokumen-dokumen perusahaan, struktur organisasi dan lain-lain yang berhubungan dengan penelitian.

b. Data Sekunder

Data yang diperlukan untuk mendukung hasil penelitian berasal dari *litelature*, artikel dan berbagai sumber lainnya yang berhubungan dengan penelitian.

2.1.2 Informasi

Menurut Anggraeni dan Irviani mengemukakan bahwa “Informasi adalah data yang diolah menjadi lebih berguna dan berarti bagi penerimanya, serta

untuk mengurangi ketidakpastian dalam proses pengambilan keputusan mengenai suatu keadaan” (Anggraeni dan Irviani 2017, 1).

Sutabri mengemukakan di dalam buku yang berjudul Analisis Sistem Informasi bahwa:

“Informasi adalah data yang telah diolah atau diklasifikasikan atau diinterpretasikan untuk digunakan di dalam proses pengambilan keputusan. Sistem pengolahan informasi akan mengolah data menjadi informasi atau mengolah data dari bentuk yang tidak berguna menjadi berguna bagi yang menerimanya. Nilai informasi berhubungan dengan keputusan, bila tidak ada pilihan atau keputusan maka informasi tidak diperlukan” (Sutabri 2012, 22).

Menurut Muslihudin dan Oktafianto menambahkan bahwa “kualitas dari suatu informasi tergantung dari tiga hal, yaitu” (Muslihudin dan Oktafianto 2016, 10):

a. Informasi harus akurat(*accurate*)

Informasi harus bebas dari kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti bahwa informasi harus jelas mencerminkan maksudnya.

b. Tepat waktu

Informasi yang sampai pada si penerima tidak boleh terlambat, informasi yang sudah usang tidak akan mempunyai nilai lagi. Karena informasi merupakan landasan di dalam pengambilan keputusan.

c. Relevan

Informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi untuk setiap orang, satu dengan lainnya adalah berbeda.

Berdasarkan hal-hal dijabarkan di atas dapat ditarik kesimpulan bahwa informasi adalah sebuah fakta atau data yang telah diolah untuk menghasilkan sebuah pengetahuan yang berguna untuk mengambil suatu keputusan.

2.1.3 Internet

Menurut Murhada dan Yo Ceng Giap “Internet adalah jaringan komputer yang bisa dikategorikan sebagai WAN, menghubungkan berjuta komputer di seluruh dunia, di mana setiap orang memiliki komputer dapat bergabung ke dalam jaringan ini” (Murhada dan Yo 2011, 195).

Penjelasan internet menurut (Kustiyahningsih dan Anamisa 2011, 2) adalah “Sebuah jaringan komputer dunia, semua berbicara dengan bahasa yang sama. Dengan internet tersebut, satu komputer dapat berkomunikasi secara langsung dengan komputer yang lain diberbagai belahan dunia.”

Berikut ini adalah karakteristik internet:

- a. Beroperasi secara virtual / maya.

Berada didunia maya (*Cyberspace*), dihuni oleh orang-orang yang saling berinteraksi, berdiskusi dan bertukar pikiran, tetapi tanpa harus melakukan pertemuan secara fisik. Dan sebenarnya, penghuni dunia maya tidak hanya manusia, tetapi termasuk didalamnya adalah data, informasi, surat elektronik ide-ide dan bahkan sampai ilmu pengetahuan.

- b. Dunia *Cyber* selalu berubah dengan cepat.

Dengan didukung kemudahan *Update* data, maka perubahan yang terjadi dalam dunia *Cyber* pun sangat cepat. Informasi-informasi yang terjadi didalamnya sangat cepat berubah.

- c. Orang-orang yang hidup dalam dunia maya tersebut dapat melaksanakan aktivitas tanpa harus menunjukkan identitasnya. Interaksi yang dilakukan dalam *Cyberspace* tanpa melibatkan interaksi secara fisik maka interaksi yang dilakukan pun tidak harus menunjukkan identitas yang sesungguhnya. Namun, walaupun identitas yang diberikan palsu, registrasi tetap bisa dilaksanakan dengan baik.

2.1.4 Basis Data

Annis menemukan bahwa “Basis Data atau lebih dikenal database, dalam teknologi informasi adalah aplikasi komputer yang digunakan untuk menyimpan, mengelola, dan mengakses informasi yang dikenal sebagai data. Terdapat banyak aplikasi basis data, biasa disebut DBMS (DataBase Management System)” (Annis 2014, 5).

Sebuah basis data (database) memiliki satu atau banyak entitas (umumnya menggunakan istilah table), dan masing-masing table tersebut memiliki satu atau banyak atribut (umumnya menggunakan istilah field atau column).

Penulis lain menambahkan bahwa “ada banyak jenis basis data, Anda dapat mendapatkannya secara gratis ataupun berbayar. Contoh aplikasi basis data yang bisa Anda kunjungi adalah” (Politwika 2017, 345):

- a. *MS Access*
- b. *DBV*
- c. *FoxPro*
- d. *Paradox*
- e. *MySQL*

2.2 Teori Khusus

2.2.1 Aplikasi

Murhada dan Yo Ceng Giap mengungkapkan bahwa “Perangkat lunak aplikasi sejatinya adalah program yang didesain khusus untuk menyelesaikan masalah spesifik pengguna, seperti melakukan tugas-tugas mengetik dokumen, manipulasi foto, merancang bangunan” (Murhada dan Yo 2011, 104).

a. Perangkat lunak hiburan

Mencakup berbagai macam program permainan (game), program untuk memainkan musik, dan program menonton film.

b. Perangkat lunak Pendidikan

Berguna untuk mempelajari atau mereferensi suatu pengetahuan, yang termasuk kategori ini misalkan aplikasi kamus, ensiklopedia, pelajaran ilmu pengetahuan, pelajaran bahasa asing, dan simulasi.

c. Perangkat lunak bisnis

Digunakan untuk menangani aplikasi bisnis. Misalnya berupa program untuk menangani persediaan barang (inventory control), pencatatan pembayaran piutang (account receivable) dan registrasi mahasiswa (sistem informasi akademik).

d. Perangkat lunak produktivitas kerja

Berguna untuk meningkatkan produktivitas kerja pemakai. Beberapa golongan perangkat lunak untuk meningkatkan kerja diantaranya word processing, spreadsheet, desktop publishing, presentasi grafik, komunikasi, personal information manager, dan manajemen data.

Penjelasan Aplikasi menurut (Arifin 2010, 69) adalah “ Program komputer yang memiliki fungsi tertentu untuk membantu atau memberikan kemudahan pengguna menyelesaikan tugas tertentu.”

Berdasarkan basis pengembangan aplikasi (*software*) dapat dibedakan menjadi dua macam, yaitu:

a. Aplikasi berbasis *desktop*

Aplikasi berbasis *desktop* dikembangkan untuk dijalankan dimasing-masing *client* (komputer pengakses aplikasi pengolahan *database*). *Database* diletakan diserver sedangkan aplikasi diinstal dimasing-masing *client*. Pada aplikasi berbasis *desktop*, aplikasi dibangun dengan menggunakan *tools* tertentu, kemudian dikompilasi. Hasilnya hanya bisa digunakan pada komputer.

b. Aplikasi berbasis *web*

Aplikasi berbasis *web* tidak perlu diinstal dimasing-masing *client*, pengakses aplikasi karena aplikasi cukup dikonfigurasi diserver kemudian *client* mengakses dari *browser* seperti *Internet Explorer*, *Opera*, *Firefox* dan *Google Chrome*.

Perbedaan lain aplikasi berbasis *desktop* dan *web* adalah bahwa untuk aplikasi berbasis *desktop* peningkatan kecepatan dan kinerja aplikasi dengan mengoptimasi penggunaan memori, manajemen proses dan pengaturan *input-output*. Pada aplikasi berbasis *web*, faktor yang menentukan kinerja aplikasi adalah kecepatan akses *database* dan kecepatan akses jaringan dan internet.

2.2.2 Seminar

Menurut (Laba dan Rinayanthi 2018, 222) Seminar adalah “satu pertemuan di mana semua para pesertanya terlibat aktif. Di dalam seminar yang dimaksud ini, tidak ada pembicara dan peserta, seperti yang dikenal dalam seminar pada umumnya. Tidak ada perbedaan antara pembicara dan peserta. Dengan demikian seminar dibedakan dari kuliah, di mana ada seorang lektor membawakan suatu tema atau ide, dan peserta kuliah mendengarkan dan bertanya. Lektor adalah seseorang yang menguasai tema tersebut, sedangkan peserta adalah orang yang mempelajari tema tersebut.”

2.2.3 Website

Penjelasan *Website* menurut (Fathurrahman 2014, 2) adalah “Merupakan kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, video, dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait di mana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*).”

Penjelasan *Website* menurut (Arief 2011, 7) adalah “Salah satu aplikasi yang berisikan dokumen-dokumen multimedia (teks, gambar, suara, animasi, video). Didalamnya yang menggunakan *protocol HTTP (hypertext transfer protocol)* dan untuk mengaksesnya menggunakan perangkat lunak yang disebut *browser*.”

Menurut *web* digolongkan menjadi 2 yaitu:

a. *Web statis*

Web yang isinya / *content* tidak berubah-ubah. Maksudnya adalah isi dari dokumen tersebut tidak dapat diubah secara cepat dan mudah.

b. *Web dinamis*

Jenis *web* yang *content* atau isinya dapat berubah-ubah setiap saat. *Web* yang banyak menampilkan animasi flash belum tentu termasuk *web* dinamis karena *web* dinamis/berubah-ubah isinya tidak sama dengan animasi. Contoh situs *web* dinamis diantaranya: situs *web* berita, situs *web e-commerce*, situs *web e-banking*.

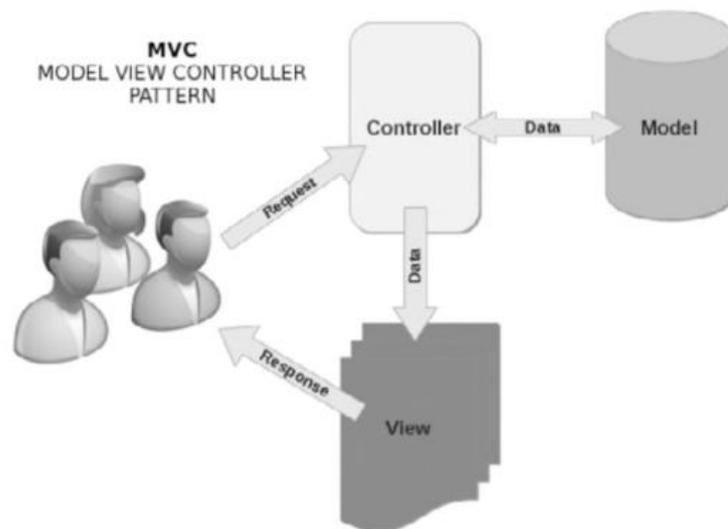
2.2.4 Framework

Menurut (Yudhanto dan Prasetyo 2018, 8) *Framework* adalah “kerangka kerja. *Framework* juga dapat diartikan sebagai kumpulan *scrip* (terutama *class* dan *function*) yang dapat membantu *developer/programmer* dalam menangani berbagai masalah-masalah dalam pemrograman, seperti koneksi ke *database*, pemanggilan variable, file, dan lain-lain sehingga pekerjaan *developer* lebih focus dan lebih cepat dalam membangun aplikasi. *Framework* adalah komponen pemrograman yang siap digunakan kapan saja sehingga programmer tidak harus membuat *script* yang sama untuk tugas yang sama.”

2.2.5 MVC

Menurut (Yudhanto dan Prasetyo 2018, 6) *Model-View-Controller* atau *MVC* adalah “sebuah metode untuk membuat sebuah aplikasi dengan memisahkan data (*Model*) dari tampilan (*View*) dan cara bagaimana memprosesnya (*Controller*).”

- a. *Model* mewakili struktur data. Biasanya *model* berisi fungsi-fungsi yang membantu seseorang dalam pengelolaan basis data, seperti memasukkan data ke basis data, pembaruan data, dan lain-lain.
- b. *View* adalah bagian yang mengatur tampilan ke pengguna. Bisa dikatakan berupa halaman web.
- c. *Controller* merupakan bagian yang menjembatani *model* dan *view*. *Controller* berisi perintah-perintah yang berfungsi untuk memproses suatu data dan mengirimkannya ke halaman web.



Gambar 2-1: MVC Pattern

(sumber: (Yudhanto dan Prasetyo 2018, 6) Panduan Mudah Belajar *Framework Laravel*)

2.3 Teori Analisa dan Perancangan

2.3.1 *Entity Relationship Diagram (ERD)*

Menurut (Aji 2014, 28), mendefinisikan ERD merupakan model jaringan data yang menekankan pada struktur-struktur dan relationship data. Sedangkan menurut (Sutanta 2011, 91), Entity Relationship Diagram(ERD) merupakan suatu model data yang dikembangkan berdasarkan objek. Entity Relationship Diagram (ERD) digunakan untuk menjelaskan hubungan antar data dalam basis data kepada pengguna secara logis. Entity Relationship Diagram (ERD) didasarkan pada suatu persepsi bahwa real world terdiri atas objek-objek dasar tersebut. Penggunaan Entity Relationship Diagram (ERD) relatif mudah dipahami, bahkan oleh para pengguna yang awam. Bagi perancang atau analis sistem, Entity Relationship Diagram (ERD) berguna untuk memodelkan sistem yang nantinya, basis data akan dikembangkan. Model ini juga membantu perancang atau analis sistem pada saat melakukan analis dan perancangan basis data karena model ini dapat menunjukkan macam data yang dibutuhkan dan kerelasian antar data didalamnya.

Entitas adalah suatu objek dalam dunia nyata yang dapat dibedakan dari objek lain. Sebagai contoh masing-masing motor adalah entitas dan konsumen dapat pula dianggap sebagai entitas. Entitas digambarkan dalam basis data dengan kumpulan atribut misalnya atribut kode, nama motor, harga cash, bisa menggambarkan data. Relasi adalah hubungan antara beberapa entitas. Sebagai contoh, relasi menghubungkan mahasiswa dengan mata kuliah yang diambilnya.

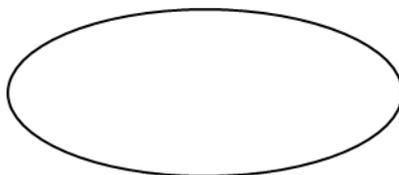
Struktur logis skema database dapat ditunjukkan secara grafis dengan diagram ERD yang dibentuk dari komponen-komponen berikut :

Entity adalah objek riil yang dapat dibedakan satu dengan yang lain. Entity digambarkan symbol seperti BOX.



Gambar 2-2: Bentuk *Entity*

Atribut adalah elemen dari entitas yang berfungsi untuk menerangkan entitas tersebut.



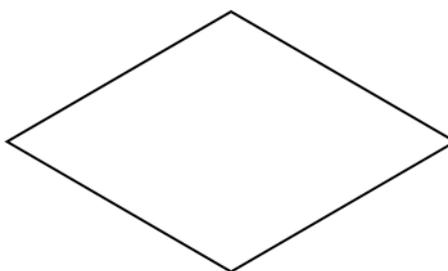
Gambar 2-3: Bentuk *Atribut*

Line berfungsi untuk menghubungkan atribut dengan entity dan entity dengan relationship(relasi).



Gambar 2-4: Bentuk *Line*

Hubungan sama saja dengan relasi, yaitu hubungan yang terjadi antara satu entitas atau lebih.



Gambar 2-5: Bentuk *Relasi*

Dari pengertian di atas dapat disimpulkan pengertian Entity Relationship (ERD) adalah memodelkan suatu kebutuhan data dari suatu organisasi yang saling terhubung satu dengan yang lainnya.

2.3.2 Flowchart

Menurut Indrajani “Flowchart (bagan alir) merupakan penggambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program. Flowchart bisa mempermudah penyelesaian masalah, khususnya yang perlu dipelajari dan dievaluasi lagi” (Indrajani 2015, 36).

Bagan alir merupakan langkah awal pembuatan program. Dengan adanya flowchart urutan proses kegiatan menjadi lebih jelas. Jika ada penambahan proses maka dapat dilakukan lebih mudah. Setelah flowchart selesai disusun, selanjutnya pemrogram (programmer) akan menerjemahkan ke dalam bentuk program dengan bahasa pemrograman.

Bagan alir (flowchart) memiliki berbagai aturan – aturan dalam perancangan diagram alirnya, yaitu:

1. System flowchart; disebut juga bagan alir sistem merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem. Bagan ini menjelaskan urutan – urutan dari prosedur – prosedur yang ada di dalam sistem dan apa yang dikerjakan di sistem.
2. Document flowchart; disebut juga bagan alir formulir (form flowchart) atau paperwork flowchart merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan – tembusannya.
3. Schematic flowchart; disebut juga bagan alir skematik merupakan bagan alir yang mirip dengan bagan alir sistem, yaitu untuk menggambarkan

prosedur di dalam sistem. Perbedaannya adalah bagan alir skematik selain menggunakan simbol – simbol bagan alir sistem, juga menggunakan gambar – gambar komputer dan peralatan lainnya yang digunakan. Penggunaan gambar – gambar dimaksudkan untuk memudahkan komunikasi kepada orang yang kurang paham dengan simbol – simbol dari bagan alir. Penggunaan gambar memudahkan untuk dipahami namun sulit dan lama dalam proses penggambaran dan pengerjaan.

4. Program flowchart; disebut juga bagan alir program merupakan bagan yang menjelaskan secara rinci langkah – langkah dari proses program. Bagan alir program dibuat dari derivikasi bagan alir sistem. Bagan alir program terbagi menjadi dua macam, yaitu bagan alir logika program (program logic flowchart) dan bagan alir komputer terinci (detail computer program flowchart). Bagan alir logika program digunakan untuk menggambarkan tiap – tiap langkah di dalam program komputer secara logika. Bagan alir logika program dipersiapkan oleh analis sitem. Sedangkan bagan alir program komputer terinci digunakan untuk menggambarkan instruksi – instruksi program komputer secara terinci. Bagan alir dipersiapkan oleh pemrogram.
5. Process flowchart; disebut juga bagan alir proses merupakan bagan alir yang digunakan di teknik industri. Bagan alir proses juga berguna pada analis sitem untuk menggambarkan proses dalam suatu prosedur.

Dalam pembuatan flowchat tidak ada rumus atau patokan yang bersifat mutlak digunakan. Karena flowchart merupakan gambaran hasil pemikiran dalam menganalisa suatu masalah dengan komputer. Sehingga flowchart yang dihasilkan dapat bervariasi antara satu pemrogram dengan pemrogram lainnya.

Namun secara garis besar setiap pengolahan selalu terdiri dari tiga bagian utama yaitu:

- a. Input; berupa masukan bahan mentah.
- b. Proses; pengolahan bahan mentah.
- c. Output; berupa hasil jadi.

Pengolahan data dengan komputer, dapat dilakukan dengan urutan dasar untuk pemecahan suatu masalah, sebagai berikut:

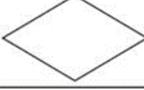
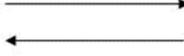
- a. START; berisikan instruksi untuk persiapan peralatan yang diperlukan sebelum menangani pemecahan masalah.
- b. READ; berisikan intruksi untuk membaca dara dari suatu peralatan input.
- c. PROCESS; berisikan kegiatan yang berkaitan dengan pemecahan masalah sesuai dengan data yang dibaca.
- d. WRITE; berisikan instruksi untuk merekam hasil kegiatan ke peralatan output.
- e. END; mengakhiri kegiatan pengolahan.

Walaupun tidak ada aturan – aturan yang baku dalam penyusunan flowchart, namun harus ada beberapa syarat yang harus ditepati, yaitu:

1. Hindari pengulangan proses yang tidak perlu dan logika yang berbeda sehingga jalannya proses menjadi singkat.
2. Penggambaran flowchart harus simetris dengan arah yang jelas.
3. Sebuah flowchart harus diawali dari satu titik START dan diakhiri dengan END.

Tujuan dari flowchart adalah untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, teratur, rapi, dan jelas dengan menggunakan simbol – simbol standar yang digunakan.

Dalam penggunaan flowchart terdapat simbol – simbol dasar yang sering digunakan, yaitu :

		
Awal / Akhir	Proses	Input Output
		
Pengujian	Keterangan	Arah
		
Pemberian Nilai Awal	Konektor pada Satu halaman	Konektor pada halaman lain

Gambar 2-6: Simbol *Flowchart*

2.3.3 *Browser*

Penjelasan *browser* menurut (Oktavian 2010, 13) adalah “program komputer yang digunakan untuk membaca *HTML*, kemudian menerjemahkan dan menampilkan hasilnya secara *visual* ke layer komputer. Anda dapat menggunakan salah satu program *web browser*, seperti: *Mozilla FireFox*, *Internet Explorer(IE)*, *Opera*, *Safari*, *Google Chrome*, dan sebagainya.”

Penjelasan *browser* menurut (Kustiyahningsih dan Anamisa 2011, 8) adalah “*Software* yang digunakan untuk menampilkan informasi dari *server web*. *Software* ini kini telah dikembangkan dengan menggunakan *user interface* grafis, sehingga pemakai dapat melakukan ‘*point and click*’ untuk pindah antar dokumen.”

2.3.4 MySQL

Penjelasan *MySQL* menurut (Komputer, Wahana 2010, 5) adalah “*MySQL* pertama kali dirintis oleh seorang *programmer database* bernama Michael Widenius. *MySQL database server* adalah *RDBMS(Relational Database Management System)* yang dapat menangani data yang bervolume besar. Meskipun begitu tidak menuntut *resource* yang besar. *MySQL* adalah *database* yang paling populer diantara *database-database* yang lain.

MySQL adalah program *database* yang mampu mengirim dan menerima data dengan sangat cepat dan multi *user*. *MySQL* memiliki dua bentuk lisensi, yaitu *free software* dan *shareware*.”

Menurut (Solichin 2005, 101) “Tipe data untuk field dalam *MySQL* diantaranya ditampilkan pada tabel berikut ini” :

Tabel 2-1: Tipe Data

(Sumber : (Solichin 2005, 101))

Jenis Tipe	Tipe	Keterangan
NUMERIK	TINYINT	-128 s/d 127 SIGNED 0 s/d 255 UNSIGNED
	SMALLINT	-32768 s/d 32767 SIGNED 0 s/d 65535 UNSIGNED
	MEDIUMINT	-8388608 s/d 8388607 SIGNED 0 s/d 16777215 UNSIGNED
	INT	-2147483648 s/d 2147483647 SIGNED 0 s/d 4294967295 UNSIGNED
	BIGINT	-9223372036854775808 s/d 9223372036854775807 SIGNED 0 s/d 18446744073709551615 UNSIGNED
	FLOAT	Bilangan pecahan presisi tunggal
	DOUBLE	Bilangan pecahan presisi ganda
	DECIMAL	Bilangan dengan desimal
DATE/TIME	DATE	Tanggal dengan format YYYY-MM-DD

Jenis Tipe	Tipe	Keterangan
	DATETIME	Tanggal dan waktu dengan format : YYYY-MM-DD HH:MM:SS
	TIMESTAMP	Tanggal dan waktu dengan format : YYYYMMDDHHMMSS
	TIME	Waktu dengan format HH:MM:SS
	YEAR	Tahun dengan format YYYY
STRING	CHAR	0 – 255 karakter
	VARCHAR	0 – 255 karakter
	TINYTEXT	String dengan panjang maksimum 255 karakter
	TEXT	String dengan panjang maksimum 65535 karakter
	MEDIUMTEXT	String dengan panjang maksimum 16777215 karakter
	LONGTEXT	String dengan panjang maksimum 4294967295 karakter
KHUSUS	ENUM	Tipe data dengan isi tertentu
	SET	Tipe data dengan isi tertentu

Penjelasan *MySQL* menurut (Kustiyahningsih dan Anamisa 2011, 145) adalah “Sebuah *Basis Data* yang mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau beberapa kolom.”

MySQL memiliki beberapa tipe data *field* yaitu:

a. Tipe data *Numeric*

Tipe data *Numeric* dibedakan dalam 2 macam kelompok yaitu *integer* dan *floating point*. *Integer* digunakan untuk data bilangan bulat sedangkan *floating point* digunakan untuk bilangan *decimal*.

Tabel 2-2: Tipe data *Numeric*

(Sumber : (Kustiyahningsih dan Anamisa 2011, 147))

Tipe Data	Kisaran Nilai
TYINT	(-128)-127 atau (0-255)
SMALLINT	(-32768)-32767 atau (0-65535)

Type Data	Kisaran Nilai
MEDIUMINT	0 - 16777215
INT(INTEGER)	0 - 4294967295
FLOAT	(-3.4 E+38)-(-1.17E-38) 0 dan (1.175E-38)- (3.4E+38)
DOUBLE	(-1.79E+308)-(-2.225E-308) 0 dan (2.225E- 308)- (1.79E+308)

b. Tipe data *String*

String adalah rangkaian karakter. Tipe-tipe data yang termasuk dalam tipe data *string*

Tabel 2-3: Tipe data *String*

(Sumber : (Kustiyahningsih dan Anamisa 2011, 148))

Type Data	Kisaran Nilai
CHAR	1 – 255 karakter
BARCHAR	1 – 255 karakter
TINYTEXT	1 – 255 karakter
TEXT	1 – 65535 karakter
MEDIUMTEXT	1 – 16777215 karakter
LONGTEXT	1 – 424967295 karakter

c. Tipe data *char()* dan *varchar()*

Tipe data *char()* dan *varchar()* pada prinsipnya sama, perbedaannya hanya terletak pada jumlah memori yang dibutuhkan untuk penyimpanannya. Memori yang dibutuhkan untuk tipe data *char()* bersifat statis, besarnya bergantung pada berapa jumlah karakter yang ditetapkan pada saat *field* tersebut dideklarasikan. Pada tipe data *varchar()* besarnya memori penyimpanan tergantung pada jumlah karakter ditambah 1 *byte*.

d. Tipe data Tanggal

Untuk tanggal dan jam, tersedia tipe-tipe data *field* berupa *DATETIME*, *DATE*, *TIMESTAMP*, *TIME*, *YEAR*. Masing-masing tipe mempunyai kisaran nilai tertentu.

Tabel 2-4: Tipe data Tanggal

(Sumber : (Kustiyahningsih dan Anamisa 2011, 149))

Tipe Data	Kisaran Nilai	Memori
DATETIME	1000-01-01 sampai 9999-12-31, 23:59:59	3 byte
DATE	1000-01-01 sampai 9999-12-31	8 byte
TIMESTAMP	1970-01-01, 00:00:00 sampai 2037	4 byte
TIME	-839:59:59 sampai 838:59:59	3 byte
YEAR	1901 sampai 2155	1 byte

Penjelasan *MySQL* menurut (Arief 2011, 151) adalah “Salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengelolaan datanya.”



Gambar 2-7: Logo MySQL

(sumber : <https://www.codepolitan.com/mysql-cli-query-5b288ea26d26c>)

2.3.5 Codeigniter

Menurut (Hakim 2010, 8), Codeigniter adalah sebuah framework PHP yang dapat membantu mempercepat developer dalam pengembangan aplikasi web berbasis PHP dibandingkan jika menulis semua kode program dari awal.



Gambar 2-8: Logo Codeigniter

(Sumber : (Hakim 2010, 9) Membangun *Web* Berbasis *PHP* dengan *Framework Codeigniter*)

Codeigniter pertama kali dibuat oleh Rick Ellis, CEO Ellislab, Inc. (www.ellislab.com), sebuah perusahaan yang memproduksi *CMS* (*Content Management System*) yang cukup handal, yaitu *Expression Engine* (www.expressionengine.com). Saat ini, *Codeigniter* dikembangkan dan dimaintain oleh *Expression Engine Development Team*.

Adapun beberapa keuntungan menggunakan *Codeigniter*, diantaranya:

1. Gratis

Codeigniter berlisensi di bawah *Apache/BSD opensource*.

2. Ditulis menggunakan *PHP 4*

Meskipun *Codeigniter* dapat berjalan di *PHP 5*, namun sampai saat ini kode program *Codeigniter* masih dibuat dengan menggunakan *PHP 4*.

3. Berukuran kecil

Ukuran *Codeigniter* yang kecil merupakan keunggulan tersendiri.

Dibandingkan dengan *framework* lain yang berukuran besar.

4. Menggunakan konsep *MVC*

Codeigniter menggunakan konsep *MVC* yang memungkinkan pemisahan *layer application-logic* dan *presentation*.

5. *URL* yang sederhana

Secara default, *URL* yang dihasilkan *Codeigniter* sangat bersih dan *Search Engine Friendly(SEF)*.

6. Memiliki paket *library* yang lengkap

Codeigniter mempunyai *library* yang lengkap untuk mengerjakan operasi-operasi yang umum dibutuhkan oleh sebuah aplikasi berbasis *web*, misalnya mengakses *database*, mengirim *email*, memvalidasi *form*, menangani *session* dan sebagainya.

7. *Extensible*

Sistem dapat dikembangkan dengan mudah menggunakan *plugin* dan *helper*, atau dengan menggunakan *hooks*.

8. Tidak memerlukan *template engine*

Meskipun *Codeigniter* dilengkapi dengan *template parser* sederhana yang dapat digunakan, tetapi hal ini tidak mengharuskan kita untuk menggunakannya.

9. Dokumentasi lengkap dan jelas

Dari sekian banyak *framework*, *Codeigniter* adalah satu-satunya *framework* dengan dokumentasi yang lengkap dan jelas.

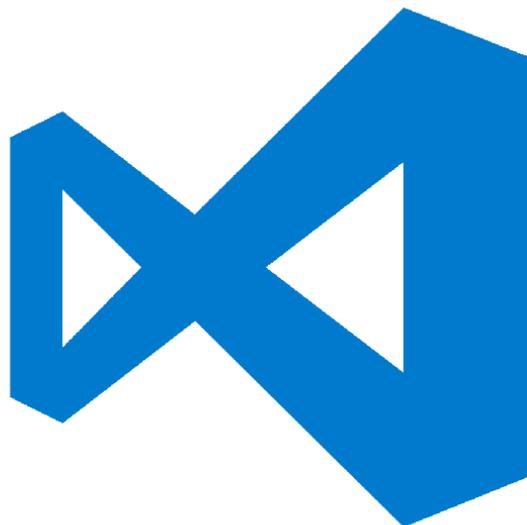
10. Komunitas-komunitas *codeigniter* saat ini berkembang pesat. Salah satu komunitasnya bisa dilihat di (<https://forum.codeigniter.com/>).

2.3.6 *Visual Studio Code*

Penjelasan *Visual Studio Code* menurut (Faisal 2017, 13) adalah “*source code editor multiplatform* yang dapat digunakan pada sistem operasi *Windows*, *Linux* dan *Mac OSX*. *Visual Studio Code* juga mendukung banyak bahasa pemrograman seperti halnya *Visual Studio 2015* ditambah bahasa pemrograman *PHP*, *Node.js* dan lain-lain.”

Fitur-fitur utama dari *Visual Studio Code* adalah:

1. *Intelligent code completion*, fitur ini akan membantu *software developer* untuk melengkapi *variable*, *method* dan modul yang ditulis.
2. *Streamlined debugging*, fitur ini berfungsi untuk melakukan *debug* terhadap kode yang ditulis.
3. *Linters*, *multi-cursor editing*, parameter hints.
4. *Code navigation*.
5. *Refactoring*.
6. Dukungan akses *Git*.



Gambar 2-9: Logo *Visual Studio Code*

(sumber: <https://github.com/Microsoft/vscode/issues/36877>)

2.3.7 XAMPP

Penjelasan *XAMPP* menurut (Yudhanto dan Prasetyo 2018, 14) adalah “kompilasi program aplikasi gratis terfavorit di kalangan *developer/programmer* yang berguna untuk pengembangan *website* berbasis *php* dan *MySQL*. Dengan satu aplikasi ini, kita akan mendapatkan paket *software* komplet yang bisa dijalankan pada *Windows* ataupun *Linux* sehingga *programmer* dapat dengan mudah melakukan simulasi pada komputer *local* sebelum diunggah ke internet.”



Gambar 2-10: Logo XAMPP

(sumber : <https://blog.pucc.or.id/port-apache-pada-xampp-error-ini-solusinya/xampp-logo/>)

2.3.8 PhpMyAdmin

Penjelasan *PhpMyAdmin* menurut (Yudhanto dan Prasetyo 2018, 15) adalah “aplikasi *web* untuk mengelola database *MySQL* dan database *MariaDB* dengan mudah melalui antarmuka(*interface*) grafis.”

Menurut (B. Nugroho 2013, 71), *PhpMyAdmin* adalah *tools* yang dapat digunakan dengan mudah untuk manajemen database *MySQL* secara visual

dan *server MySQL*, sehingga kita tidak perlu lagi harus menulis *query SQL* setiap akan melakukan perintah operasi *database*. *Tools* ini cukup populer, anda dapat mendapatkan fasilitas ini ketika menginstal paket *trial PhpMyAdmin*, karena termasuk dalam *XAMPP* yang sudah diinstal.

Menurut (B. Nugroho 2013, 71), *PhpMyAdmin* adalah *Tools MySQL Client* berlisensi *Freeware*. *PhpMyAdmin* harus dijalankan di sisi *server web* dan pada komputer harus tersedia *PHP*, karena berbasis *web*.

Menurut (B. Nugroho 2013, 72), adapun cara untuk menjalankan *PhpMyAdmin* adalah sebagai berikut :

1. Buka *XAMPP Control Panel*, lalu jalankan modul servis *Apache web server*.
2. Jalankan juga *MySQL Server*-nya, sehingga sekarang *Apache* dan *MySQL* statusnya *Running*.
3. Sekarang buka *web browser* anda, misalnya *Mozilla Firefox* atau *Google Chrome*.
4. Pada kotak alamat, ketikkan : (<http://localhost/phpmyadmin/>) lalu tekan *Enter*.

2.3.9 HTML (*Hyper Text Markup Language*)

“*HTML* adalah suatu bahasa yang dikenali oleh *web browser* untuk menampilkan informasi dengan lebih menarik dibandingkan dengan tulisan teks biasa(*plain text*)” (Oktavian 2010, 13).

Jubilee Enterprise mengungkapkan *HTML* di dalam buku yang berjudul “*HTML 5 Komplet : Bacaan Komplet Untuk Membantu Memahami HTML Secara Lengkap*” bahwa (*Jubilee Enterprise* 2017, 1): “*HTML* merupakan

singkatan dari *Hypertext Markup Language*, dari definisi tersebut, kita dapat mengetahui ciri-ciri *HTML* sebagai berikut:”

a. *Hypertext*

Di dalam kode *HTML*, kita dapat membuat teks menjadi *link* yang fungsinya untuk berpindah satu halaman ke halaman lainnya atau dari satu halaman ke alamat situs lain hanya dengan mengklik teks tersebut.

b. *Markup Language*

HTML menggunakan tanda (dalam bahasa Inggris disebut *mark*) untuk menandai bagian-bagian dari teks agar teks itu memiliki tampilan/fungsi tertentu. Misalnya, teks yang berada di antara *mark* `` atau `<i>` akan menjadi tebal serta miring. Dalam prakteknya, tanda atau *mark* tersebut disebut dengan istilah *tag*.



Gambar 2-11: Logo HTML

(sumber: https://commons.wikimedia.org/wiki/File:HTML5_logo_and_wordmark.svg)

2.3.10 CSS (*Cascading Style Sheet*)

CSS adalah bahasa yang digunakan untuk format *HTML* agar menjadi lebih bagus dan efektif dalam tampilan (Yudhanto and Prasetyo 2018, 4).

Menurut (Sya'ban 2010, 37), Cascading Style Sheet (CSS) merupakan salah satu bahasa pemrograman web untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam. CSS memperkenalkan template yang berupa style untuk membuat dan mempermudah penulisan dari halaman-halaman yang dirancang. Hal ini sangat penting karena halaman yang menggunakan CSS dapat dibaca secara bolak balik dan isinya dapat dilihat oleh pengunjung dari manapun. CSS mampu menciptakan halaman yang tampak sama pada resolusi layar dari 28 pengunjung yang berbeda tanpa memerlukan suatu table. Dengan CSS, anda akan lebih mudah melakukan setting

tampilan keseluruhan web. Hanya dengan menggantikan atribut-atribut atau perintah dalam style CSS dengan atribut yang diinginkan tanpa harus mengubah satu per satu atribut tiap elemen yang ada dalam situs yang dibuat. Saat ini CSS merupakan style yang banyak digunakan karena berbagai kemudahan dan kelengkapan atribut yang dimilikinya. Penggunaan CSS dalam web akan lebih efisien karena CSS dapat digunakan untuk penggunaan secara berulang pada tag-tag tertentu sehingga tidak usah menyetikkan ulang seluruh perintah pemformatan seperti halnya HTML klasik. CSS akan lebih mudah dipelajari jika telah mengetahui struktur pembuatan dokumen web dengan Bahasa HTML

CSS dapat mengendalikan ukuran gambar, warna bagian tubuh pada text, warna table, ukuran border, warna border, warna hyperlink, warna mouse over, spasi antar paragraph, spasi antar teks, margin kiri,kanan,atas,bawah, dan parameter lainnya. CSS adalah Bahasa style sheet yang digunakan untuk mengatur tampilan dokumen. Dengan adanya CSS memungkinkan kita untuk menampilkan halaman yang sama dengan format yang berbeda. Untuk saat ini terdapat tiga versi CSS yaitu CSS1,CSS2, dan CSS3. CSS1 dikembangkan berpusat pada pemformatan dokumen HTML, CSS2 dikembangkan untuk memenuhi kebutuhan terhadap format dokumen agar bisa ditampilkan di printer, sedangkan CSS3 adalah versi terbaru dari CSS yang mampu melakukan banyak hal dalam desain website. CSS2 mendukung penentuan posisi konten, downloadable, huruf (font), tampilan pada table (table layout) dan media tipe untuk printer. Kehadiran versi CSS yang kedua diharapkan lebih baik dari versi pertama. CSS3 juga dapat melakukan animasi pada halaman website 3D. Dengan CSS3 desainer lebih dimudahkan dalam hal kompatibilitas websitenya pada smartphone dengan dukungan fitur baru yakni media query, Selain itu,

banyak fitur baru pada CSS3 seperti: multiple, background, borderradius, drop-shadow, border-image, CSS math, dan CSS Object Model.



Gambar 2-12: Logo CSS

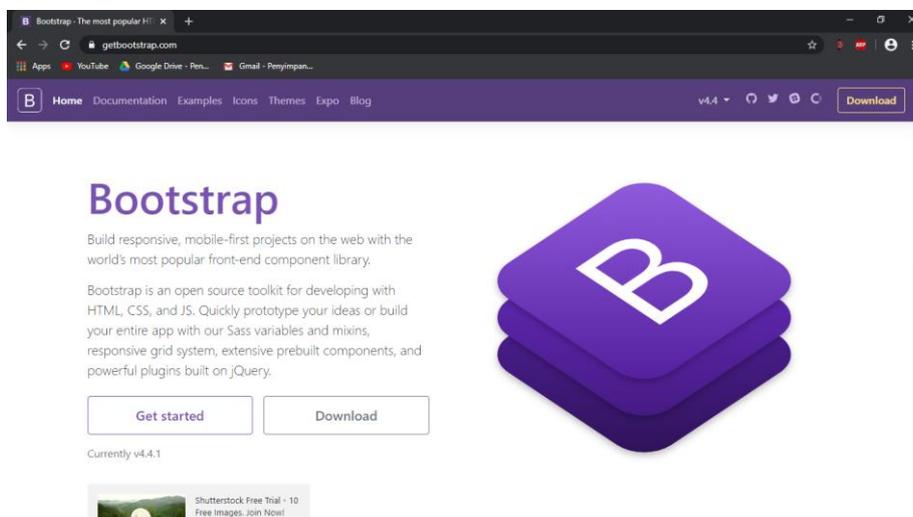
(sumber: https://commons.wikimedia.org/wiki/File:CSS3_logo_and_wordmark.svg)

2.3.11 Bootstrap

Rozi dan Community mengungkapkan bahwa: “*Bootstrap* adalah paket aplikasi siap pakai untuk membuat *front-end* sebuah *website*. Bisa dikatakan, *Bootstrap* adalah *template* desain *web* dengan fitur *plus*. *Bootstrap* diciptakan untuk mempermudah proses desain *web* bagi berbagai tingkat pengguna, mulai dari level pemula hingga yang sudah berpengalaman. Cukup bermodalkan pengetahuan dasar mengenai *HTML* dan *CSS*, Anda pun siap menggunakan *Bootstrap*.

Paket *Bootstrap* berisi sekumpulan file *CSS*, *font*, dan *JavaScript* yang siap diintegrasikan ke sebuah dokumen *HTML* menggunakan kaidah-kaidah tertentu. Dokumen *HTML* yang dihasilkan pun secara dinamis akan tampil dalam *layout* yang disesuaikan dengan ukuran layar piranti pengunjung.

Daya tarik *Bootstrap* terletak pada kemudahan dan kepraktisan penggunaannya. Tata *layout* dan pewarnaannya juga bersih, *simple*, indah, dan berkesan modern. Selain itu, mengingat akses *web* via piranti *mobile* semakin meningkat tajam, maka pendekatan desain *web responsive* dan *mobile-friendly* yang ditawarkan *Bootstrap* pun juga menjadi solusi praktis dan murah bagi para pengembang *web*. Cukup dengan *Bootstrap*, *website* anda bisa diakses secara nyaman via ponsel tanpa menggunakan aplikasi khusus untuk piranti *mobile*” (Rozi dan Community 2105, 1).



Gambar 2-13: Official Website Bootstrap

(sumber: <https://getbootstrap.com/>)

2.3.12 PHP (*Hypertext Preprocessor*)

Sianipar mengungkapkan bahwa: “*PHP* merupakan bahasa skrip yang di tanamkan di dalam *HTML*, ini berarti Anda dapat mencampur kode *PHP* dan *HTML* pada file yang sama” (R. Sianipar 2015, 2).

Sianipar menambahkan bahwa di dalam buku yang berjudul “*PHP & MySQL*”: “Untuk memulai pemrograman *PHP*, anda memulainya dengan menulis kode *HTML* sederhana. Untuk menambahkan kode *PHP* pada sebuah

halaman, tempatkan di dalam tag-tag PHP. Segala sesuatu yang ditempatkan di dalam tag-tag ini akan dianggap oleh web server sebagai kode PHP. Sembarang teks yang ada diluar tag-tag PHP ini dianggap sebagai kode HTML biasa” (R. Sianipar 2015, 2).

PHP adalah akronim dari *Hypertext Preprocessor*, yaitu suatu bahasa pemrograman berbasis kode-kode (*script*) yang digunakan untuk mengolah suatu data dan mengirimkannya kembali ke *web browsing* menjadi kode *HTML*. Kode PHP mempunyai ciri-ciri khusus, yaitu (Oktavian 2010, 31):

- a. Hanya dapat dijalankan menggunakan *web server*, misalnya: *Apache*.
- b. Kode *PHP* diletakkan dan dijalankan di *web server*.
- c. Kode *PHP* dapat digunakan untuk mengakses *database*, seperti: *MySQL*, *PostgreSQL*, *Oracle*, dan lain-lain.
- d. Merupakan *software* yang bersifat *open source*.
- e. Gratis untuk di-*download* dan digunakan.
- f. Memiliki sifat *multiplatform*, artinya dapat dijalankan menggunakan *sistem operasi* apapun, seperti: *Linux*, *Unix*, *Windows*, dan lain-lain.

Penjelasan *PHP* menurut (Kustiyahningsih dan Anamisa 2011, 114) adalah “*PHP* (atau resminya *PHP : Hypertext Preprocessor*) adalah skrip bersifat *server-side* yang ditambahkan kedalam *HTML*. *PHP* sendiri merupakan singkatan dari *Personal Home Page Tools*. Skrip ini akan membuat suatu aplikasi dapat diintegrasikan kedalam *HTML* sehingga suatu halaman *web* tidak lagi bersifat statis, namun menjadi bersifat dinamis. Sifat *Server side* berarti pengerjaan skrip dilakukan di *server*, baru kemudian hasilnya dikirimkan ke *browser*.”

```

1 <html>
2 <body>
3 <?php echo "Belajar"; ?>
4 </body>
5 </html>

```

Gambar 2-14: Contoh penulisan *PHP*

(**Sumber:** (Kustiyahningsih dan Anamisa 2011, 114) Pemrograman Basis Data Berbasis Web Menggunakan PHP & MySQL : Edisi Pertama)

Penjelasan *PHP* menurut (Arief 2011, 143) adalah “Bahasa *server-side scripting* yang menyatu dengan *HTML* untuk membuat halaman *web* yang dinamis. Karena *PHP* merupakan *server-side scripting* maka *sintaks* dan perintah-perintah *PHP* akan dieksekusi diserver kemudian hasilnya dikirimkan ke *browser*.”



Gambar 2-15: Logo *PHP*

(**sumber:** <https://www.updateilmu.com/apa-itu-php/logo-php/>)

2.3.13 *Java Script*

Menurut (E. R. Sianipar 2017, 1) *JavaScript* adalah “Bahasa script dinamis yang dapat dipakai untuk membangun interaktifitas pada halaman-halaman *HTML* statis. Ini dilakukan dengan menanamkan blok-blok kode *JavaScript* di hampir semua tempat pada halaman *web* anda.”



Gambar 2-16: Logo Java Script

(sumber: <https://www.brandsoftheworld.com/logo/javascript>)

2.3.14 Blackbox Testing

Simarmata mengungkapkan di dalam buku berjudul “Rekayasa Perangkat Lunak” bahwa:

“*Black box testing* merupakan pendekatan pengujian di mana program dianggap sebagai suatu kotak hitam program *test case* berbasiskan spesifikasi *test planning* dapat dimulai sejak awal proses pengembangan sistem. *Black box testing* berfokus pada persyaratan fungsional perangkat lunak. Metode ini memungkinkan *software developer* untuk mendapatkan serangkaian kondisi input yang mempergunakan semua persyaratan fungsional program” (Simarmata 2010, 316).

Menurut (Rizky 2011, 264) *Black box testing* adalah “Tipe testing yang memperlakukan perangkat lunak yang tidak diketahui kinerja internalnya, sehingga para *tester* memandang perangkat lunak seperti layaknya sebuah kotak hitam yang tidak penting dilihat isinya, tapi cukup dikenali proses *testing* dibagian luar.”

Beberapa teknik testing yang tergolong dalam tipe *Black Box Testing* menurut (Rizky 2011, 265) antara lain:

a. *Equivalence Partitioning*

Pada teknik ini, tiap inputan data dikelompokkan ke dalam grup tertentu, yang kemudian dibandingkan *output*-nya.

b. *Boundary Value Analysis*

Pada teknik *Boundary Value Analysis*, dilakukan *input*-an yang melebihi batasan sebuah data, contohnya untuk sebuah *input*-an harga barang, maka dapat dilakukan *testing* dengan menggunakan angka *negatif* yang tidak diperbolehkan dalam sebuah harga. Jika perangkat lunak berhasil mengatasi *input*-an yang salah tersebut, maka dapat dikatakan teknik ini telah selesai dilakukan.

c. *Cause Effect Graph*

Dalam teknik ini, dilakukan proses *testing* yang menghubungkan sebab dari sebuah *input*-an dan akibatnya pada *output* yang dihasilkan.

2.3.15 *Whitebox Testing*

Menurut kompasiana *white box testing*: “White Box Testing merupakan cara pengujian dengan melihat ke dalam modul untuk meneliti kode-kode program yang ada, dan menganalisis apakah ada kesalahan atau tidak. Jika ada modul yang menghasilkan output yang tidak sesuai dengan proses bisnis yang dilakukan, maka baris-baris program, variabel, dan parameter yang terlibat pada unit tersebut akan dicek satu persatu dan diperbaiki, kemudian di-compile ulang” (Zahara, Larastia dan Nurmala 2015).

Dengan menggunakan white box akan didapatkan kasus uji yang :

1. Menguji semua keputusan logikal.
2. Menguji seluruh Loop yang sesuai dengan batasannya.

3. Menguji seluruh struktur data internal yang menjamin validitas.

Kelebihan White Box Testing

- a. Kesalahan Logika

Digunakan pada sintaks “ if ” dan pengulangan. Dimana White Box Testing akan mendeteksi kondisi-kondisi yang tidak sesuai dan mendeteksi kapan proses pengulangan akan berhenti.

- b. Ketidaksesuaian asumsi

Menampilkan asumsi yang tidak sesuai dengan kenyataan, untuk di analisa dan diperbaiki.

- c. Kesalahan ketik

Mendeteksi bahasa pemrograman yang bersifat case sensitive.

Kelemahan White Box Testing

Untuk perangkat lunak yang tergolong besar, White Box Testing dianggap sebagai strategi yang tergolong boros, karena akan melibatkan sumber daya yang besar untuk melakukannya.

Beberapa teknik testing yang tergolong dalam tipe *White Box Testing* diantaranya:

- a. *Basis Path*

Basis Path Testing memungkinkan perancang test case mendapatkan ukuran kompleks logical dari perancangan prosedural dan menggunakan ukuran ini sebagai petunjuk untuk mendefinisikan himpunan jalur yang akan diuji. Basis Path menggunakan notasi graph atau flow graph untuk menggambarkan aliran kontrolnya.

b. *Cyclomatic Complexity*

Cyclomatic Complexity merupakan suatu sistem pengukuran yang menyediakan ukuran kuantitatif dari kompleksitas logika suatu program. Pada Basis Path Testing, hasil dari cyclomatic complexity digunakan untuk menentukan banyaknya independent paths. Independent path adalah sebuah kondisi pada program yang menghubungkan node awal dengan node akhir.

2.4 Tinjauan Pustaka

2.4.1 Penelitian Pastima Simanjuntak dan Arwin Kasnady

Tabel 2-5: Penelitian Pastima Simanjuntak, Arwin Kasnady

No.	Data Jurnal / Makalah	Keterangan
1.	Judul	Analisis <i>Model View Controller(MVC)</i> pada bahasa <i>PHP</i>
2.	Jurnal	Jurnal <i>ISD</i>
3.	Volume dan Halaman	Volume 02 Nomor 02, 56-66
4.	Tanggal dan Tahun	2016
5.	Penulis	1. Pastima Simanjuntak 2. Arwin Kasnady
6.	Penerbit	Fakultas Teknik Program Studi Teknik Informatika
7.	Tujuan Penelitian	Bahwa menerapkan model <i>MVC</i> dalam pemograman <i>php</i>
8.	Lokasi dan Subjek	-
9.	Perancangan Sistem	<i>Apache Benchmark(ab)</i> , <i>Google Chrome Developer Tools(Timeline)</i> .
10.	Hasil Penelitian	Hasil penelitian ini adalah: Untuk mengetahui performa web.
11.	Kekuatan Penelitian	Kekuatan penelitian ini adalah : a. Memiliki perhitungan kecepatan koneksi dalam <i>request</i> pengguna. b. Dalam analisa ini menggunakan aplikasi <i>Apache Benchmark(ab)</i> .
12.	Kelemahan Penelitan	Kelemahan penelitian ini adalah : a. Tidak memiliki <i>Flowchart</i> aplikasi.
13.	Kesimpulan	Kesimpulan : Untuk melihat seberapa kuat performa web saat menghadapi <i>request</i> dan koneksi dari pengguna.

2.4.2 Penelitian Melda Agarina, Arman Suryadi Karim

Tabel 2-6: Penelitian Melda Agarina, Arman Suryadi Karim

No.	Data Jurnal / Makalah	Keterangan
1.	Judul	Rancangan Bangun Sistem Informasi Kegiatan Seminar Nasional Berbasis Web Pada Institut Informatics dan Bisnis Darmajaya
2.	Jurnal	Jurnal Sistem Informasi dan Telematika
3.	Volume dan Halaman	Volume 10 Nomor 01, 26-32
4.	Tanggal dan Tahun	2019
5.	Penulis	1. Melda Agarina 2. Arman Suryadi Karim
6.	Penerbit	Explore
7.	Tujuan Penelitian	Menyediakan fasilitas berupa <i>website</i> seminar nasional yang dapat digunakan sebagai media informasi dimana dalam <i>website</i> tersebut dapat memenuhi kebutuhan peserta yaitu dapat membantu peserta mendapatkan informasi seminar, membantu mendapatkan rute perjalanan ke lokasi seminar, membantu pemakai melakukan pengecekan status makalahnya dan setelah kegiatan seminar tersebut dilaksanakan peserta dapat mengunduh materi kegiatan dan dokumentasi kegiatan di dalam <i>website</i> tersebut.
8.	Lokasi dan Subjek	Kota Bandar Lampung
9.	Perancangan Sistem	<i>Unified Modelling Language (UML)</i>
10.	Hasil Penelitian	Hasil penelitian ini adalah : a. Memudahkan pihak manajemen dalam inventarisasi data kegiatan seminar. b. Memberikan <i>feedback</i> yang cepat dan akurat bagi peserta seminar
11.	Kekuatan Penelitian	Kekuatan penelitian ini adalah : a. Melakukan penelitian di Instansi. b. Dapat melihat rute lokasi seminar. c. Dapat mengunduh materi dan dokumentasi dalam <i>website</i> tersebut.
12.	Kelemahan Penelitian	Kelemahan penelitian ini adalah : a. Tidak menjelaskan bahasa pemrograman yang dipakai.
13.	Kesimpulan	Kesimpulan : a. Telah tersusun sebuah <i>system</i> informasi <i>website</i> yang dapat menangani pendaftaran peserta seminar nasional berbasis <i>web</i> pada IIB Darmajaya. b. <i>Output</i> dapat menyimpan data serta menghasilkan laporan akhir kegiatan seminar nasional.

2.4.3 Penelitian Dwi Arianto Adi Nugroho, Heru Supriyono

Tabel 2-7 Penelitian Dwi Arianto Adi Nugroho, Heru Supriyono

No.	Data Jurnal / Makalah	Keterangan
1.	Judul	Sistem Informasi Pendaftaran Seminar Dengan Tiket Berbasis <i>Qr Code</i>
2.	Jurnal	Jurnal <i>Emitor</i>
3.	Volume dan Halaman	Volume 19 Nomor 01, 34-40
4.	Tanggal dan Tahun	2019
5.	Penulis	1. Dwi Arianto Adi Nugroho 2. Heru Supriyono
6.	Penerbit	Universitas Muhammadiyah Surakarta
7.	Tujuan Penelitian	Dapat digunakan sebagai salah satu alat yang dapat mempermudah peserta seminar untuk menghadiri acara seminar tersebut.
8.	Lokasi dan Subjek	-
9.	Perancangan Sistem	<i>XAMPP, Notepad ++</i>
10.	Hasil Penelitian	Hasil penelitian ini adalah : Sebuah sistem pendaftaran seminar online dengan tiket berbasis <i>Qr Code</i> .
11.	Kekuatan Penelitian	Kekuatan penelitian ini adalah : a. Dapat mencetak tiket dalam <i>Qr Code</i> . b. Pengecekan <i>Qr Code</i> menggunakan webcam. c. Menggunakan Metode <i>SDLC waterfall</i> dalam pembuatan aplikasi.
12.	Kelemahan Penelitian	Kelemahan penelitian ini adalah : a. Tidak dapat <i>scan</i> dengan perangkat lain.
13.	Kesimpulan	Kesimpulan : a. Sistem bekerja dengan baik sesuai dengan rancangan awal dan fitur-fitur yang ada pada sistem berjalan dengan baik dan efisien. b. Perangkat <i>webcam</i> yang digunakan untuk membaca data <i>Qr Code</i> juga berjalan dengan baik di percahayaan yang cukup terang dalam jarak 20-30cm. c. Pengujian yang melibatkan peserta acara yang menggunakan sistem ini memberikan tanggapan melalui kuesioner mencapai 91.47% yang membuktikan bahwa sistem ini efisien dan layak diterapkan.

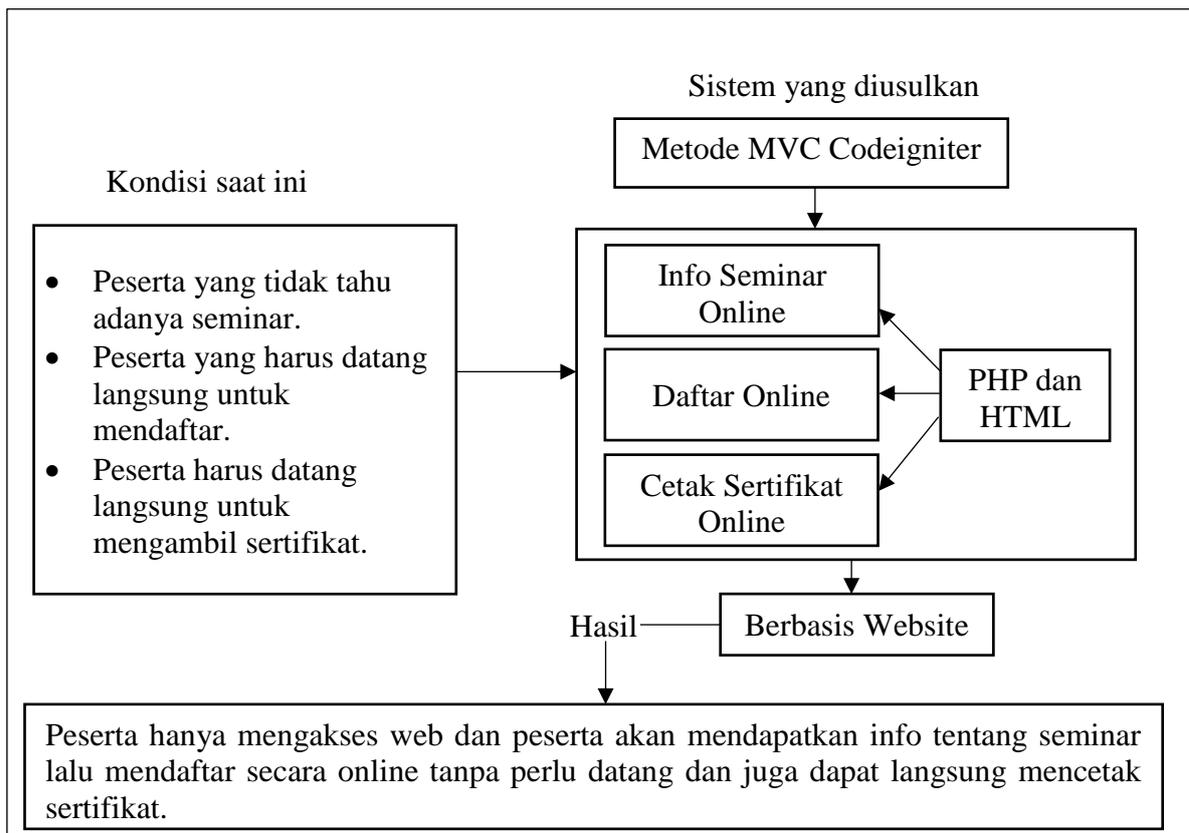
2.4.4 Rangkuman Model Penelitian

Tabel 2-8: Rangkuman Model Penelitian

Penelitian	1. Pastima Simanjuntak	1. Melda Agarina	1. Dwi Arianto Adi Nugroho
-------------------	------------------------	------------------	----------------------------

	2. Arwin Kasnady	2. Arman Suryadi Karim	2. Heru Supriyono
Nama Jurnal	Jurnal ISD ISSN: 2528-5114	Jurnal Sistem Informasi dan Telematika ISSN: 2087-2062	Jurnal <i>Emitor</i> ISSN: 2541-4518
Tahun	2016	2019	2019
Institusi	-	Informatika dan Bisnis Darmajaya	-
Judul dan Metode yang digunakan	Analisis Model View Conroller(MVC) pada bahasa PHP	Rancangan Bangun Sistem Informasi Kegiatan Seminar Nasional Berbasis Web Pada Institut Informatics dan Bisnis Darmajaya	Sistem Informasi Pendaftaran Seminar Dengan Tiket Berbasis Qr Code
Kesimpulan	Untuk melihat seberapa kuat performa web saat menghadapi request dan koneksi dari pengguna.	<p>a. Telah tersusun sebuah <i>system informasi website</i> yang dapat menangani pendaftaran peserta seminar nasional berbasis <i>web</i> pada IIB Darmajaya.</p> <p>b. <i>Output</i> dapat menyimpan data serta menghasilkan laporan akhir kegiatan seminar nasional.</p>	<p>a. Sistem bekerja dengan baik sesuai dengan rancangan awal dan fitur-fitur yang ada pada sistem berjalan dengan baik dan efisien.</p> <p>b. Perangkat <i>webcam</i> yang digunakan untuk membaca data <i>Qr Code</i> juga berjalan dengan baik di percahayaan yang cukup terang dalam jarak 20-30cm.</p> <p>c. Pengujian yang melibatkan peserta acara yang menggunakan sistem ini memberikan tanggapan melalui kuesioner mencapai 91.47% yang membuktikan bahwa sistem ini efisien dan layak diterapkan.</p>

2.5 Kerangka Pemikiran



Gambar 2-17: Kerangka Pemikiran

BAB III

ANALISA MASALAH DAN PERANCANGAN APLIKASI

3.1 Analisa Kebutuhan

Pada tahap ini akan dilakukannya analisa kebutuhan dalam penelitian ini di mana analisa tersebut terdiri dari analisa kebutuhan untuk membangun sistem, analisa kebutuhan pengguna dan analisa kebutuhan aplikasi.

3.1.1 Analisa kebutuhan membangun sistem

Berikut ini adalah komponen-komponen dalam membangun aplikasi ini:

- a. Laptop *Aspire E5-475G*
- b. *OS Windows 10 Pro 1909 64-Bit*
- c. Prosesor *i5-7200U CPU@2,50 – 2,70 GHz*
- d. *RAM 12GB*
- e. *Visual Studio Code (Text Editor)*
- f. *XAMPP*
- g. *MySQL*
- h. *Apache*
- i. *Google Chrome*
- j. *Codeigniter*
- k. *Bootstrap 4*

3.1.2 Analisa Kebutuhan Pemakai

Berdasarkan analisa kebutuhan calon peserta anggota pendaftaran seminar adalah sebagai berikut:

Tabel 3-1: Analisa Kebutuhan Pemakai

No	Kebutuhan Pemakai
1	Tampilan yang rapih dan dapat diakses oleh media manapun.
2	Memberikan data pendaftaran yang jelas.
3	Memiliki metode pembayaran yang mudah seperti cash, transfer bank, dll.
4	Memiliki fitur untuk memberikan sertifikat secara online.
5	Dapat mentracking tempat seminar melalui google maps.
6	Memberikan pesan validasi terhadap data pendaftar.
7	Memiliki fitur untuk mengupload bukti transfer.
8	User dapat dengan mudah mengakses info seminar.
9	Memberikan kemudahan dalam mendaftar seminar.
10	Sertifikat seminar tersedia diwebsite dan bisa didownload.
11	Dapat dengan mudah merubah data user diprofile.
12	Tampilan pada web menggunakan bahasa indonesia.
13	User dapat mengetahui status pembayaran.

3.1.3 Analisa Kebutuhan Aplikasi

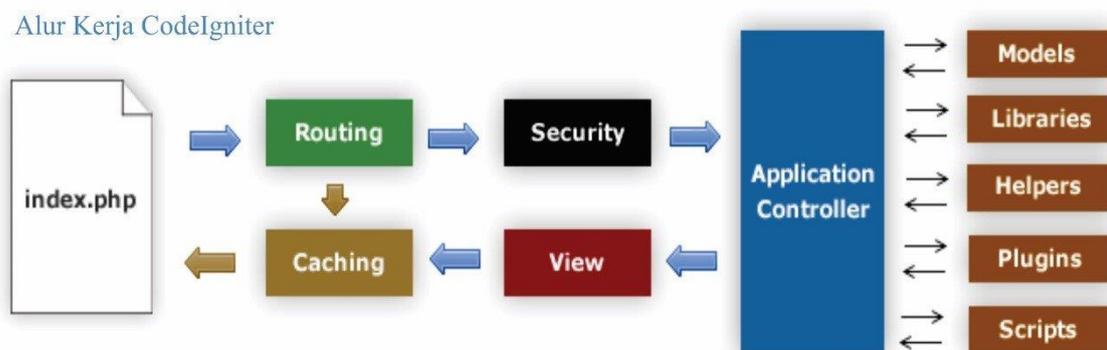
Berdasarkan analisa kebutuhan pengguna di atas dibuatlah program aplikasi yang dapat memenuhi kebutuhan pengguna yang dijelaskan pada table di bawah ini:

Tabel 3-2: Analisa Kebutuhan Aplikasi

No	Kebutuhan Pemakai	Keterangan
1	Tampilan yang rapih dan dapat diakses oleh media manapun.	√
2	Memberikan data pendaftaran yang jelas.	√
3	Memiliki metode pembayaran yang mudah seperti cash, transfer bank, dll.	√
4	Memiliki fitur untuk memberikan sertifikat secara online.	√
5	Dapat mentracking tempat seminar melalui google maps.	x
6	Memberikan pesan validasi terhadap data pendaftar.	√
7	Memiliki fitur untuk mengupload bukti transfer.	√

No	Kebutuhan Pemakai	Keterangan
8	User dapat dengan mudah mengakses info seminar.	√
9	Memberikan kemudahan dalam mendaftar seminar.	√
10	Sertifikat seminar tersedia diwebsite dan bisa didownload.	√
11	Dapat dengan mudah merubah data user diprofile.	x
12	Tampilan pada web menggunakan bahasa indonesia.	√
13	User dapat mengetahui status pembayaran.	√

3.2 Alur Kerja *Framework Codeigniter*



Gambar 3-1: Alur Kerja *Codeigniter*

1. *File index.php* (kadang disebut *entry scripts*) berfungsi sebagai *front controller*, menginisialisasi *resource* utama yang dibutuhkan untuk menjalankan *Codeigniter*.
2. *Router* memeriksa *HTTP request* untuk menentukan apa yang harus dilakukan dengan itu.
3. Jika *file cache* ada, dikirim langsung ke *browser*, melewati eksekusi sistem normal.
4. *Security*, Sebelum *controller* aplikasi dimuat, *HTTP request* dan setiap data pengguna yang di-submit disaring terlebih dahulu untuk keamanan.
5. *Controller* memuat *model*, *library* utama, *helper*, dan setiap *resource* lainnya yang diperlukan untuk memproses permintaan khusus.

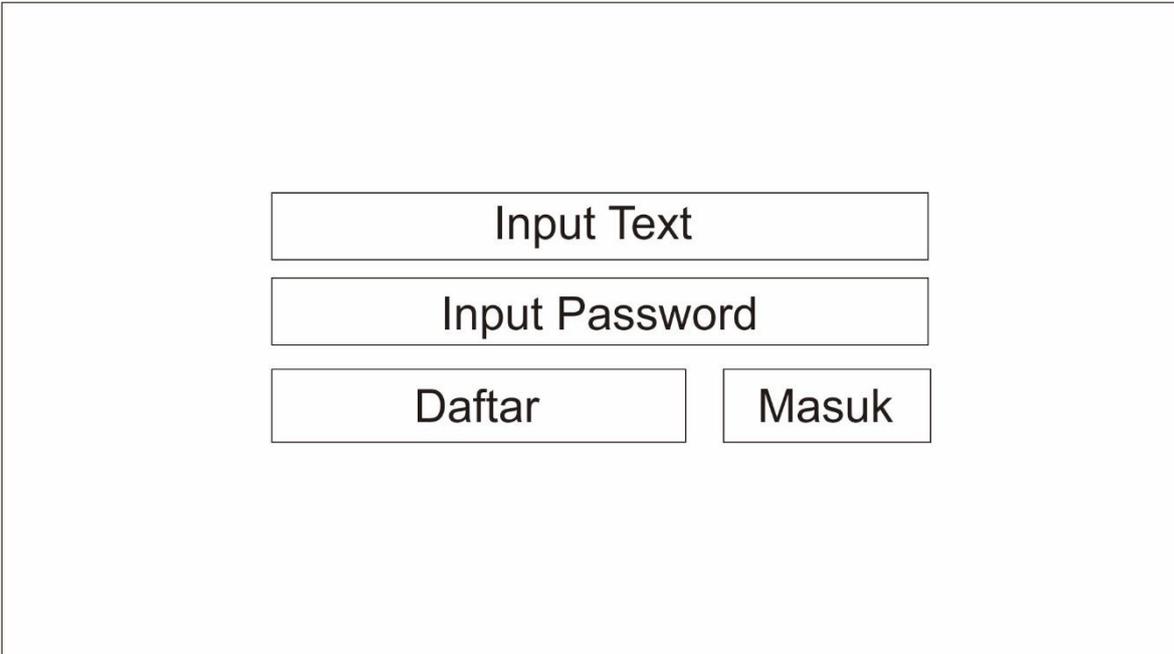
6. *View* di-render kemudian dikirim ke *web browser* agar dapat dilihat. Jika *caching* diaktifkan, *view* di-cache terlebih dahulu sehingga pada permintaan berikutnya dapat dilayani.

3.3 Perancangan Layar, Menu, *Database*

3.3.1 Perancangan Layar dan Menu

- a. Rancangan Layar *Login*

Pada rancangan layar ini terdapat dua *input text* untuk memasukan *username* dan *password* dan juga terdapat dua tombol, yaitu tombol daftar untuk menampilkan halaman daftar dan juga tombol masuk untuk menampilkan halaman *homepage/dashboard*. Untuk lebih jelasnya dapat dilihat pada gambar berikut ini.



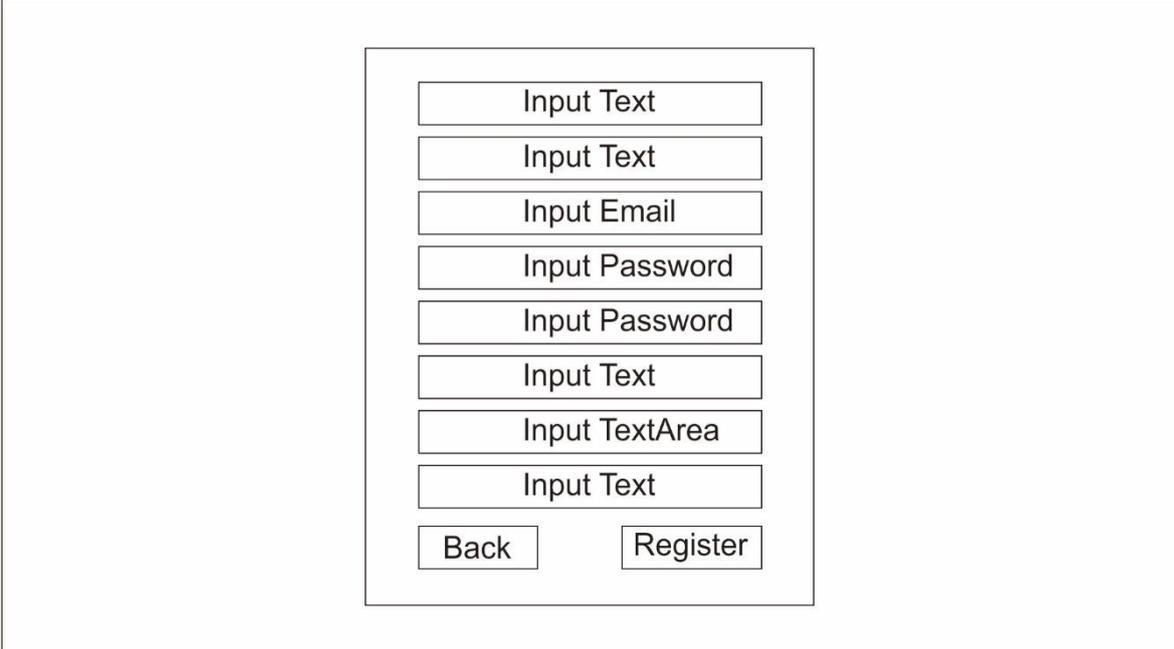
The diagram illustrates the layout of a login screen. It consists of four rectangular elements arranged vertically and horizontally. At the top is a wide box labeled 'Input Text'. Below it is another wide box labeled 'Input Password'. At the bottom, there are two smaller boxes side-by-side: the left one is labeled 'Daftar' and the right one is labeled 'Masuk'.

Gambar 3-2: Rancangan Layar *Login*

- b. Rancangan Layar Daftar Pengguna

Pada rancangan layar ini terdapat empat *field* yaitu *Input Text*, *Input Email*, *Input Password*, dan *TextArea* dan juga terdapat dua tombol, yaitu tombol register untuk

menyimpan data dan juga tombol back untuk kembali ke halaman *login*, Untuk lebih jelasnya dapat dilihat pada gambar berikut ini.

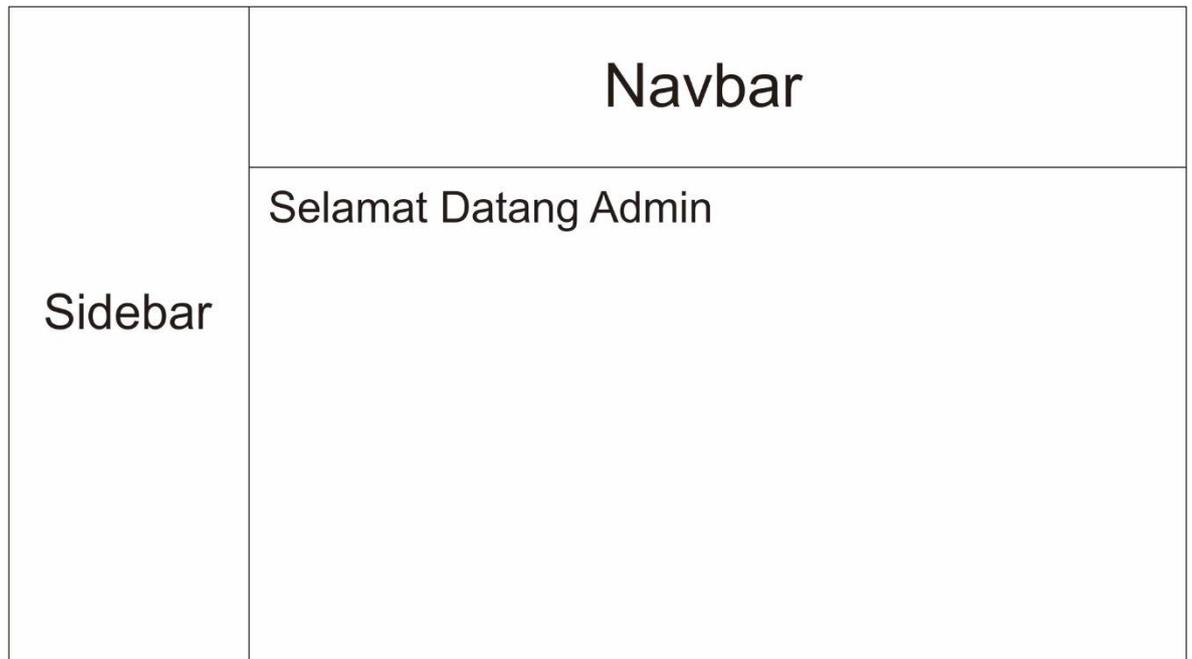


The image shows a wireframe for a user registration page. It consists of a central rectangular box containing a vertical stack of input fields and buttons. From top to bottom, the elements are: an 'Input Text' field, another 'Input Text' field, an 'Input Email' field, an 'Input Password' field, a second 'Input Password' field, an 'Input Text' field, an 'Input TextArea' field, and a third 'Input Text' field. At the bottom of the stack are two buttons: 'Back' on the left and 'Register' on the right.

Gambar 3-3: Rancangan Layar Daftar Pengguna

c. Rancangan *Dashboard Admin*

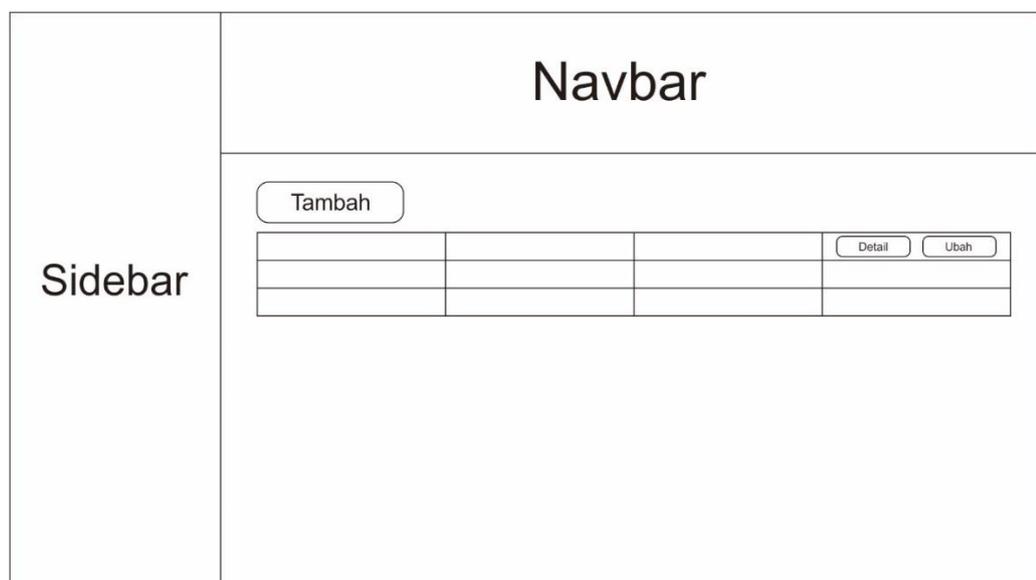
Pada rancangan layar ini terdapat *navbar* dan juga *sidebar* yang digunakan untuk pengoperasian aplikasi ini, Untuk lebih jelasnya dapat dilihat pada gambar berikut ini.



Gambar 3-4: Rancangan Layar *Dashboard Admin*

d. Rancangan Layar Seminar

Pada rancangan layar ini menampilkan data dari tabel seminar kedalam tabel dan juga terdapat tiga tombol, yaitu tombol tambah untuk menambahkan seminar dan juga tombol detail untuk menampilkan detail pendaftar seminar dan juga tombol ubah untuk mengubah data seminar, Untuk lebih jelasnya dapat dilihat pada gambar berikut ini.



Gambar 3-5: Rancangan Layar Seminar

e. Rancangan Tambah Seminar

Pada rancangan layar ini terdapat 3 *field* yaitu *Input Text*, *Input Date*, dan *Input Time* dan juga terdapat 1 tombol simpan yang berfungsi untuk menyimpan data seminar pada tabel seminar, Untuk lebih jelasnya dapat dilihat pada gambar berikut ini.

Sidebar	Navbar	
	Tambah Seminar	
	Label	Input Text
	Label	Input Text
	Label	Input Date
	Label	Input Time
	Simpan	

Gambar 3-6: Rancangan Tambah Seminar

f. Rancangan Ubah Seminar

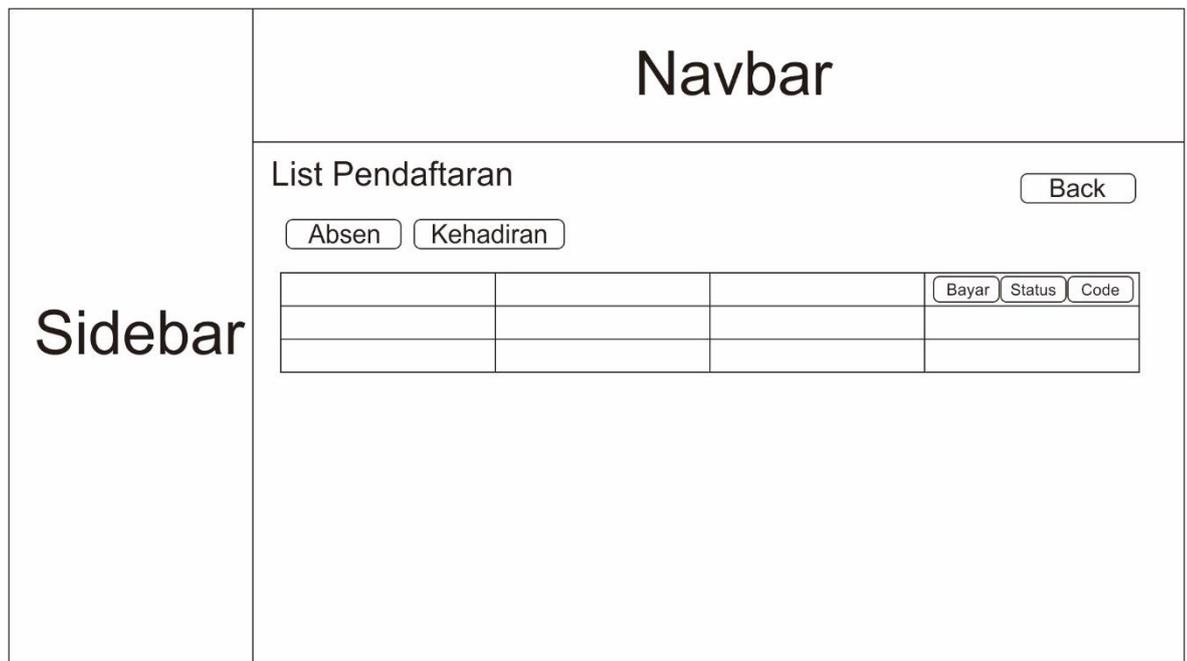
Pada rancangan layar ini terdapat beberapa *Input Text*, satu *Input Date*, dua *Input Time* dan juga tombol simpan yang dimana berfungsi untuk memperbaharui data pada tabel seminar, Untuk lebih jelasnya dapat dilihat pada gambar berikut ini.

Sidebar	Navbar																	
	<div style="border: 1px solid black; padding: 10px; margin: 10px auto; width: 80%;"> <p style="text-align: center; margin: 0;">Ubah seminar</p> <table style="width: 100%; border-collapse: collapse;"> <tr><td style="width: 60%;">Label</td><td><input type="text"/></td></tr> <tr><td>Label</td><td><input type="text"/></td></tr> </table> <p style="text-align: center; margin-top: 10px;"><input type="button" value="Simpan"/></p> </div>	Label	<input type="text"/>	Label														
Label	<input type="text"/>																	
Label	<input type="text"/>																	
Label	<input type="text"/>																	
Label	<input type="text"/>																	
Label	<input type="text"/>																	
Label	<input type="text"/>																	
Label	<input type="text"/>																	
Label	<input type="text"/>																	
Label	<input type="text"/>																	

Gambar 3-7: Rancangan Layar Ubah Seminar

g. Rancangan Detail Pendaftar Seminar

Pada rancangan layar ini terdapat tabel untuk menampilkan detail calon pendaftar pada setiap seminar dan juga terdapat enam tombol yaitu tombol absen, kehadiran, bayar, status, code, dan back. Dan jika menekan tombol absen akan muncul *report* absen yang digunakan untuk mengabsen peserta pada saat seminar dan tombol kehadiran untuk melihat siapa peserta yang hadir dan tidak hadir pada saat seminar lalu ada tombol bayar yang berfungsi untuk mengupdate status pembayaran dan tombol status yang digunakan untuk mengupdate kehadiran peserta lalu tombol code yang berfungsi untuk memberikan kode sertifikat dan tombol back yang berfungsi untuk kembali ke halaman list seminar admin. Untuk lebih jelasnya dapat dilihat pada gambar berikut ini.



Gambar 3-8: Rancangan Detail Pendaftar Seminar

h. Rancangan Tampilan Halaman Utama *User*

Pada rancangan layar ini terdapat daftar seminar serta detail seminar yang dimana berfungsi untuk menginformasikan kepada pengguna bahwa ada seminar dan juga bisa melakukan pendaftaran dengan menekan tombol daftar dimana terdapat dua opsi yaitu transfer dan cash, setelah menekan tombol tersebut maka pengguna akan berpindah ke halaman daftar. Untuk lebih jelasnya dapat dilihat pada gambar berikut ini.



Gambar 3-9: Rancangan Tampilan Halaman Utama User

i. Rancangan Daftar Seminar

Pada rancangan layar ini terdapat *input text*, tombol *Choose file*, dan tombol simpan, Yang dimana pada *input text* berfungsi untuk memasukan deskripsi pembayaran dan tombol *choose file* berfungsi untuk memasukan gambar pembayaran lalu tombol simpan yang berfungsi untuk melakukan pendaftaran seminar tersebut. Untuk lebih jelasnya dapat dilihat pada gambar berikut ini.

Sidebar	Navbar
	Daftar
	Deskripsi:
	<input type="button" value="Choose File"/> <input type="button" value="Simpan"/>

Gambar 3-10: Rancangan Daftar Seminar

j. Rancangan List Seminar *User*

Pada rancangan layar ini terdapat tabel list seminar yang telah di daftar oleh pengguna lalu terdapat tiga tombol yaitu tombol cetak untuk mencetak nomor pendaftar/tiket dan tombol sertifikat yang berfungsi untuk mencetak sertifikat lalu tombol *ReUpload* yang berfungsi untuk merubah gambar pembayaran bila terjadi kesalahan. Untuk lebih jelasnya dapat dilihat pada gambar berikut ini.

Sidebar	Navbar		
	Seminar		
			<input type="button" value="Cetak"/> <input type="button" value="Sertifikat"/> <input type="button" value="ReUpload"/>

Gambar 3-11: Rancangan List Seminar User

k. Rancangan *ReUpload* Foto Pembayaran

Pada rancangan layar ini terdapat *input text*, tombol *choose file*, tombol simpan, *text*, dan tombol lihat. Yang dimana *input text* untuk merubah deskripsi pembayaran bila salah dalam memberikan deskripsi dan tombol *choose file* berfungsi untuk memilih gambar pembayaran terbaru lalu tombol simpan yang berfungsi untuk merubah data pembayaran lalu setelah itu data yang dirubah akan muncul dalam *text* dan setelah itu menekan tombol lihat untuk memastikan kembali gambar telah berubah. Untuk lebih jelasnya dapat dilihat pada gambar berikut ini.

Sidebar	Navbar
	<div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Ubah Gambar Seminar</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;">Deskripsi:</div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> </div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> <input type="button" value="Choose File"/> </div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> <input type="button" value="Simpan"/> </div> <div style="border: 1px solid black; padding: 5px; margin-bottom: 5px;"> <small>File Name</small> </div> <div style="border: 1px solid black; padding: 5px;"> <div style="display: flex; justify-content: space-between;"> Text <input type="button" value="Lihat"/> </div> </div>

Gambar 3-12: Rancangan *ReUpload* Foto Pembayaran

3.3.2 Rancangan Database

Berikut ini merupakan rancangan basis data tergambar pada gambar di bawah ini di mana terdapat total tiga tabel pada data, tabel tersebut terdiri dari dua tabel master yaitu `tbl_seminar` dan `tbl_user` dan 1 tabel transaksi yaitu `tbl_pendaftaran`

skripsi tbl_user	skripsi tbl_pendaftaran	skripsi tbl_seminar
id_user : int(11)	id_pendaftaran : int(11)	id_seminar : int(11)
nama : varchar(50)	no_pendaftaran : varchar(25)	judul_seminar : varchar(50)
username : varchar(50)	# id_seminar : int(11)	pembicara : varchar(50)
password : varchar(50)	# id_user : int(11)	tanggal : date
level : enum('1','2')	img : text	jam_mulai : time
email : varchar(50)	setatus : enum('UNPAID','PAID','')	jam_berakhir : time
no_handphone : varchar(15)	email : varchar(50)	ruangan : varchar(50)
alamat : text	no_handphone : varchar(15)	lokasi : text
kota : varchar(50)	alamat : text	# kuota : int(4)
	kota : varchar(50)	# harga : int(11)
	# kuota : int(4)	
	no_sertifikat : varchar(50)	
	kehadiran : enum('hadir','tidak')	
	deskripsi : text	
	pembayaran_cash : text	

Gambar 3-13: Logical Record Structure (LRS)

Tabel 3-3: tbl_user

NAMA	TIPE DATA	PANJANG	KETERANGAN
id_user	int	11	Primary Key
nama	varchar	50	
username	varchar	50	
password	varchar	50	
level	enum	-	
email	varchar	50	
no_handphone	varchar	15	
alamat	text	-	
kota	varchar	50	

Tabel 3-4: tbl_seminar

NAMA	TIPE DATA	PANJANG	KETERANGAN
id_seminar	int	11	Primary Key
judul_seminar	varchar	50	
pembicara	varchar	50	
tanggal	date	-	
jam_mulai	time	-	
jam_berakhir	time	-	
ruangan	varchar	50	
lokasi	text	-	
kuota	int	4	
harga	int	11	

Tabel 3-5: tbl_pendaftaran

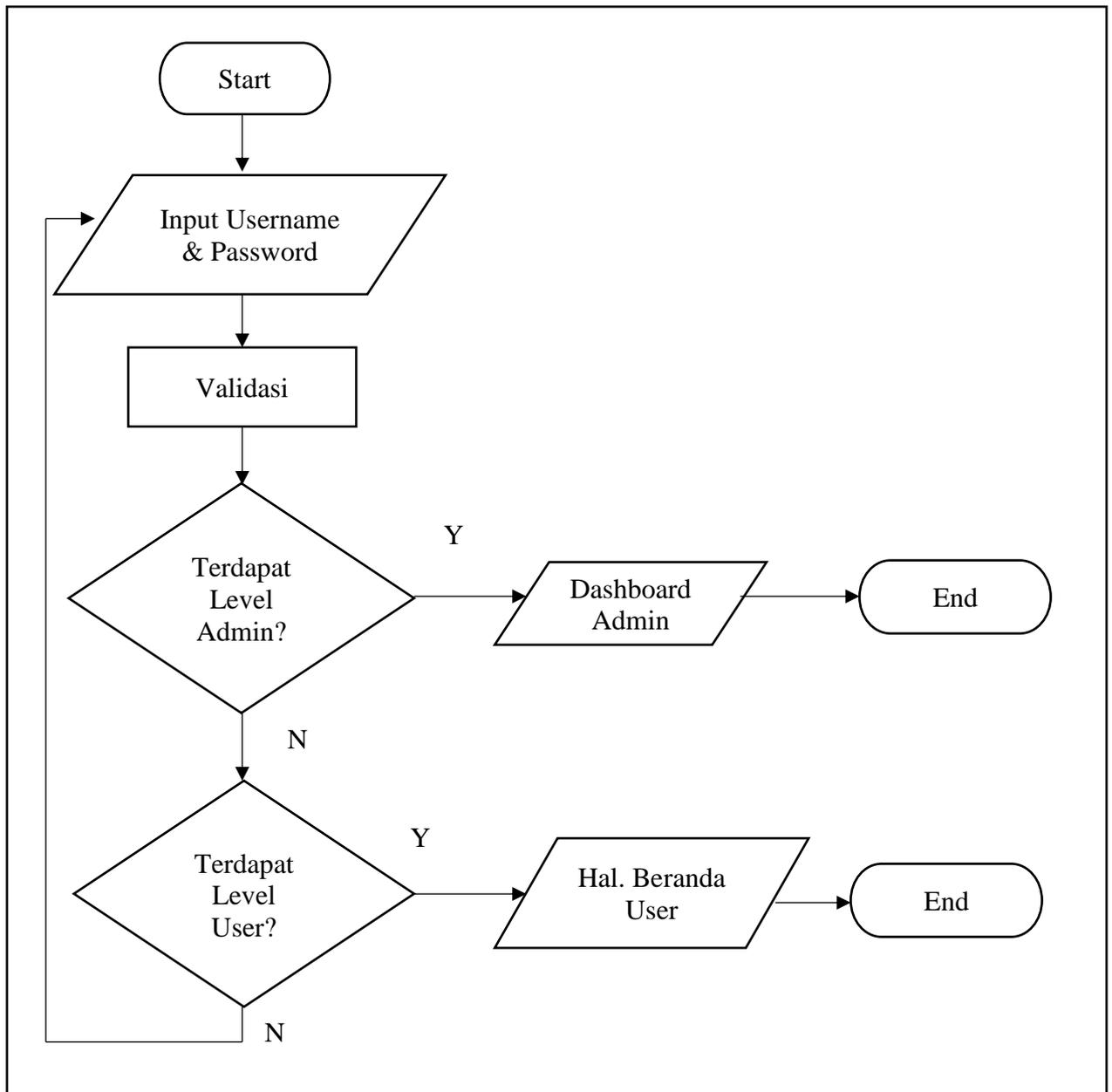
NAMA	TIPE DATA	PANJANG	KETERANGAN
id_pendaftaran	int	11	Primary Key
no_pendaftaran	varchar	25	
id_seminar	int	11	Foreign Key
id_user	int	11	Foreign Key
img	text	-	
setatus	enum	-	
email	varchar	50	
no_handphone	varchar	15	
alamat	text	-	
kota	varchar	50	
kuota	int	4	

NAMA	TIPE DATA	PANJANG	KETERANGAN
no_sertifikat	varchar	50	
kehadiran	enum	-	
deskripsi	text	-	
pembayaran_cash	text	-	

3.3.3 Rancangan *Flowchart*

a. Proses Login

Berikut ini adalah diagram alur proses login aplikasi ini, diawali dengan memasukan *username* dan *password* setelah itu sistem akan melakukan validasi kesesuaian username dan password dengan database, jika username dan password terdapat pada kolom level tipe admin maka sistem akan meneruskannya kedalam halaman dashboard admin namun jika level pengguna adalah user maka sistem akan meneruskannya ke halaman beranda user, dan juga jika username dan password yang dimasukan pengguna tidak terdapat didatabase maka sistem akan mengembalikannya ke halaman login untuk memasukan username dan password kembali. Untuk lebih jelasnya dapat dilihat pada gambar berikut ini:

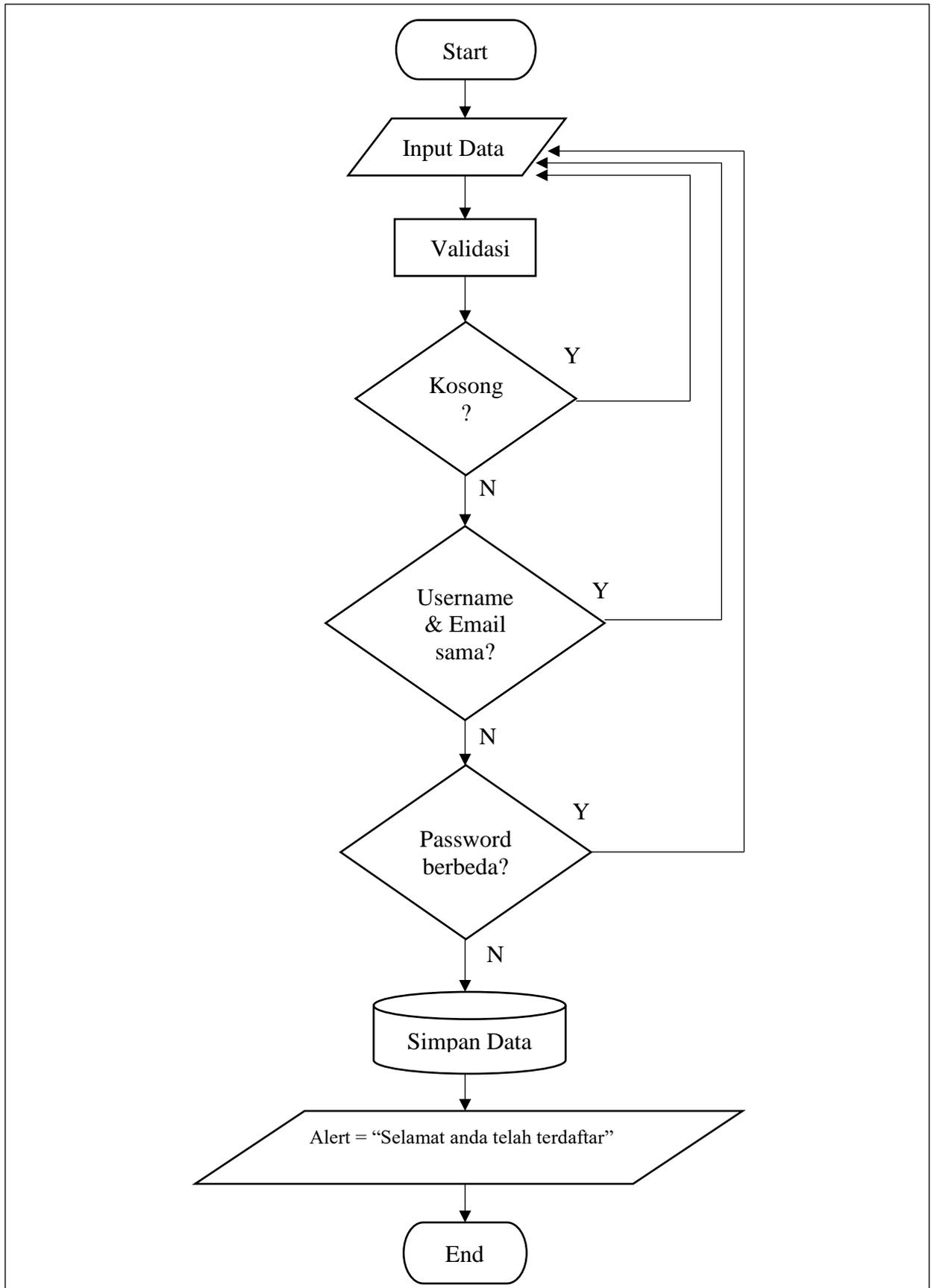


Gambar 3-14: Flowchart Proses Login

b. Proses register

Pada proses register diawali dengan user memasukkan data ke dalam aplikasi setelah itu sistem akan melakukan validasi berupa kekosongan data yang dimasukkan user, username dan email sudah terdapat dalam database, password dan retype password berbeda, jika validasi yang telah disebutkan sebelumnya kondisinya benar maka sistem akan mengembalikannya ke proses

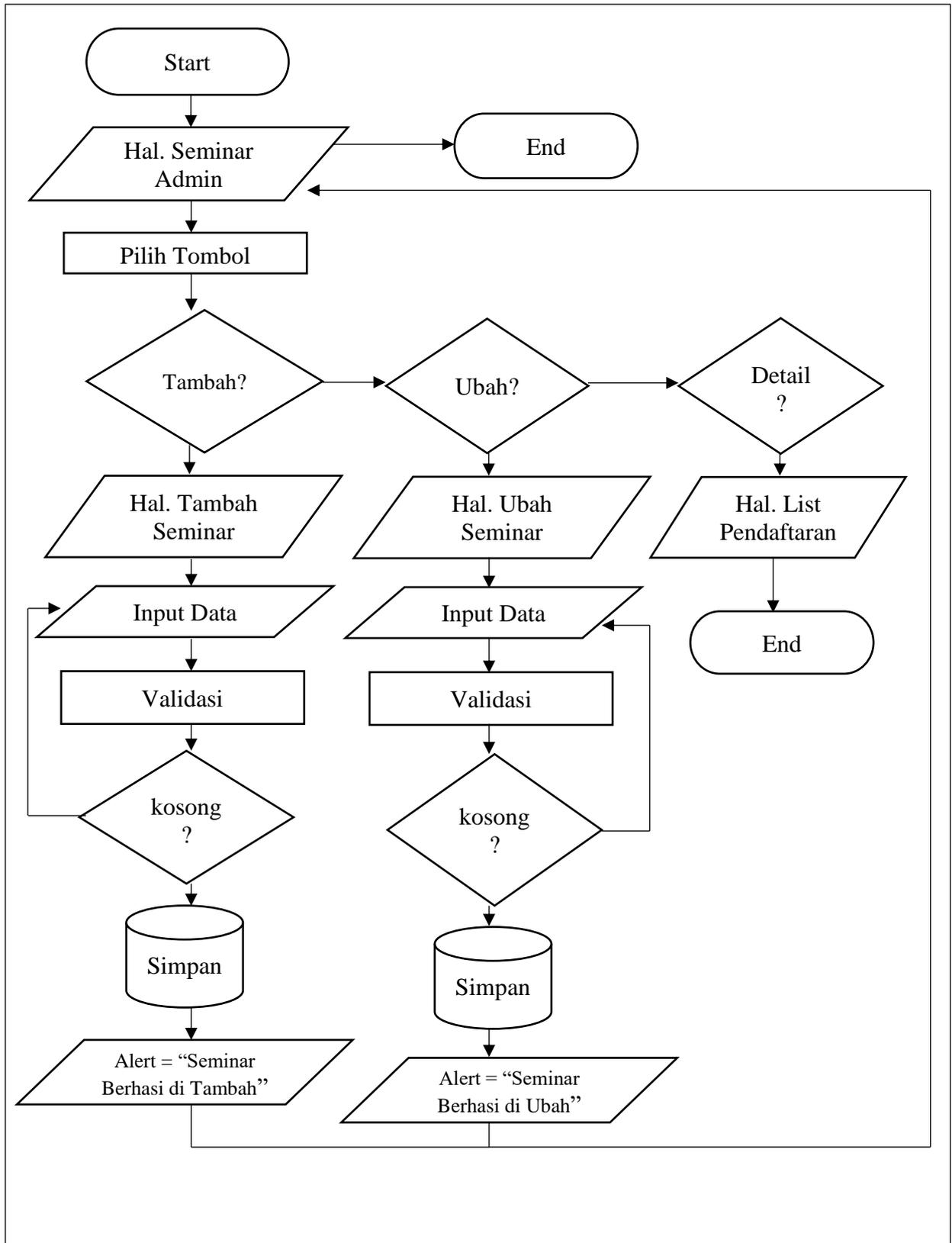
input data namun jika kondisi validasi bernilai tidak/salah maka sistem akan menyimpan data dan menampilkan informasi berupa alert. Untuk lebih jelasnya dapat dilihat pada gambar berikut ini:



Gambar 3-15: Flowchart Proses register

c. Proses Tambah, Ubah, Detail

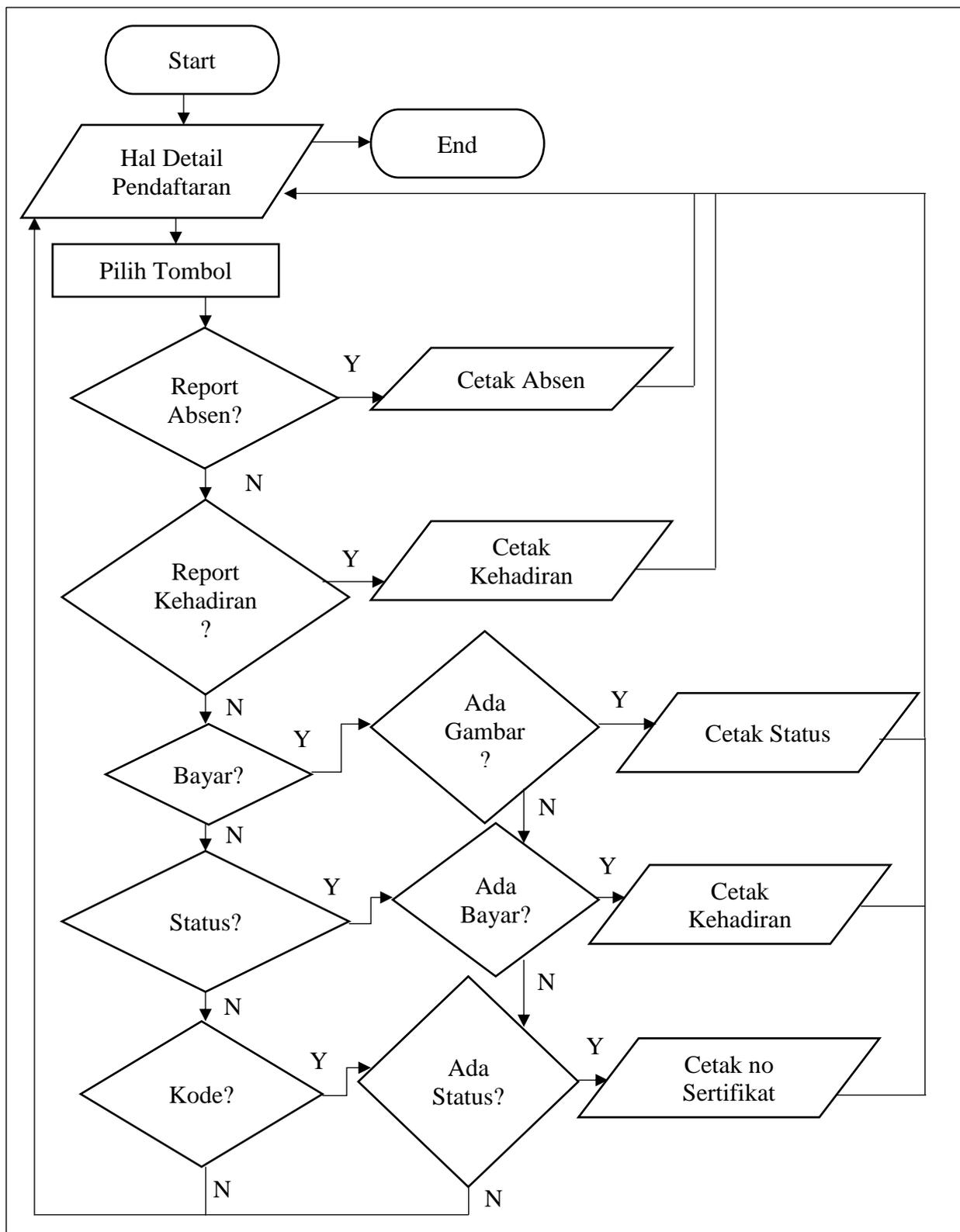
Pada proses ini diawali dengan menampilkan halaman seminar lalu user akan memilih fungsi apa yang akan dijalankan (tambah, ubah, detail), jika user memilih tombol tambah maka sistem akan menampilkan halaman tambah seminar lalu user melakukan penambahan data, lalu sistem akan melakukan validasi apakah data yang kosong dan jika kosong sistem akan mengembalikannya ke proses input data dan jika data yang diinput ada maka data akan tersimpan ke dalam database. Lalu jika user menekan tombol ubah sistem akan menampilkan halaman ubah seminar untuk melakukan perubahan data, setelah itu sistem melakukan validasi apakah data yang kosong, jika kosong maka sistem akan mengembalikannya ke proses penginputan data, jika tidak kosong data akan disimpan oleh sistem. Setelah itu jika user menekan tombol detail maka sistem akan menampilkan halaman list pendaftar seminar. Untuk lebih jelasnya dapat dilihat pada gambar berikut ini:



Gambar 3-16: Flowchart Proses Tambah, Ubah, Detail

d. Proses *Report Absen, Report Kehadiran, Pembayaran, Kehadiran, Nomor Sertifikat*

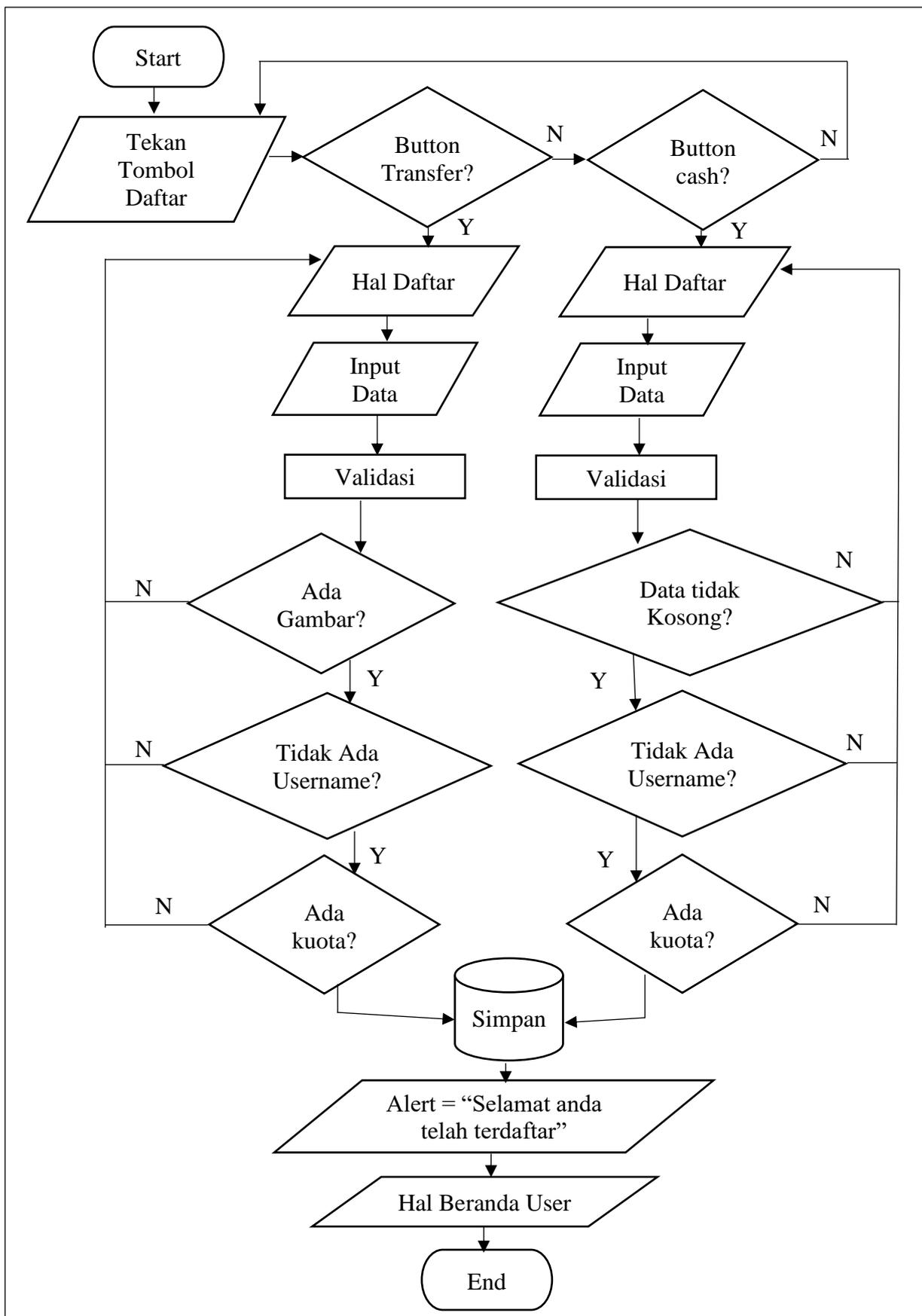
Diawali dengan halaman detail pendaftar seminar lalu user memilih tombol yang tersedia (absen, kehadiran, bayar, status, kode), Jika menekan tombol absen maka sistem akan menampilkan report absensi seminar. Lalu jika user menekan tombol kehadiran maka sistem akan menampilkan report kehadiran peserta seminar yang telah hadir dan tidak hadir pada saat seminar. Lalu jika user menekan tombol bayar sistem akan melakukan validasi adakah gambar pembayaran seminar yang telah di unggah jika ada sistem akan memperbaharui status pembayaran, lalu jika user tidak ada gambar maka sistem akan mengembalikannya ke halaman detail pendaftar seminar. Lalu jika user menekan tombol status sistem akan melakukan validasi status bayar bernilai “paid”, jika bernilai “paid” maka sistem akan memperbaharui status kehadiran, Jika status bernilai “unpaid” maka sistem akan mengembalikannya ke halaman detail pendaftar seminar. Lalu jika user menekan tombol kode sistem akan melakukan validasi adakah status kehadiran seminar, jika status kehadiran bernilai “hadir” maka sistem akan memperbaharui kode sertifikat, jika status kehadiran bernilai “tidak” maka sistem akan mengembalikan ke halaman detail pendaftar seminar. Untuk lebih jelasnya dapat dilihat pada gambar berikut ini:



Gambar 3-17: Flowchart Proses Report Absen, Report Kehadiran, Pembayaran, Kehadiran, Nomor Sertifikat

e. Proses Daftar *User*

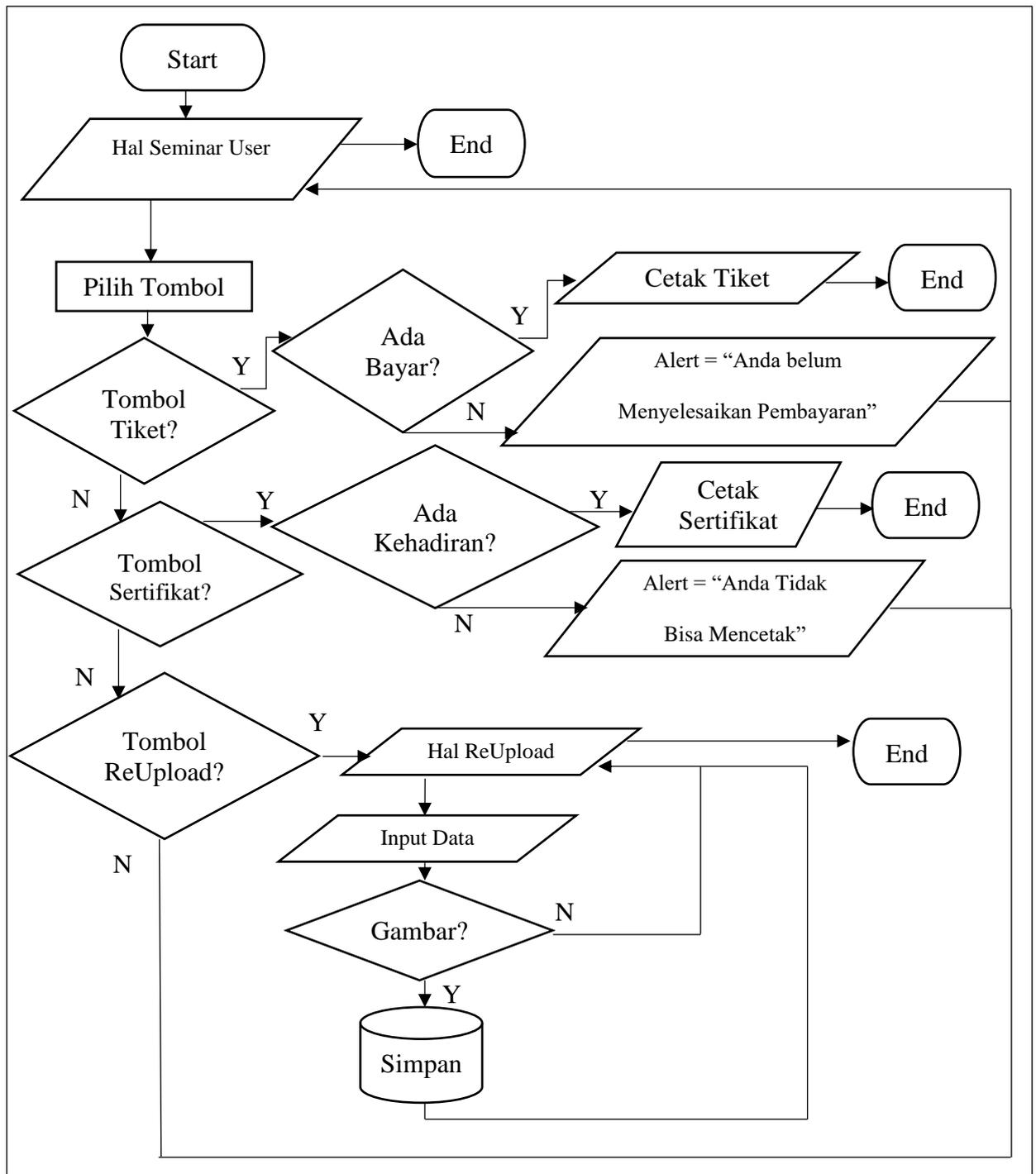
Diawali dengan menekan tombol daftar lalu sistem akan menampilkan halaman pendaftaran seminar, lalu user melakukan input data setelah itu sistem melakukan validasi yaitu (adakah gambar, tidak adakah username, adakah kuota) jika semua kondisi validasi tersebut adalah tidak terpenuhi maka sistem akan mengembalikannya ke halaman pendaftaran seminar, jika semua kondisi validasi tersebut terpenuhi maka sistem akan melakukan penyimpanan data dan memberikan alert, Setelah itu sistem akan menampilkan halaman beranda user. Untuk lebih jelasnya dapat dilihat pada gambar berikut ini:



Gambar 3-18: Flowchart Proses Daftar User

f. Proses Cetak Tiket, Cetak Sertifikat, *ReUpload* Pembayaran

Diawali dengan menampilkan halaman list seminar user setelah itu user memilih tombol (tiket, sertifikat, reupload), jika user menekan tombol cetak tiket maka sistem akan melakukan validasi adakah pembayarannya, jika ada maka aplikasi akan menampilkan ke halaman cetak tiket, jika tidak maka sistem akan memunculkan alert. Jika user menekan tombol sertifikat maka sistem akan melakukan validasi adakah status kehadiran, Jika ada maka aplikasi akan menampilkan ke halaman cetak sertifikat, Jika tidak maka sistem akan memunculkan alert. Jika pengguna menekan tombol reupload maka sistem akan menampilkan halaman reupload pembayaran setelah itu pengguna melakukan penginputan data dan sistem akan melakukan validasi adakah gambar pembayaran yang telah diunggah, Jika ada sistem akan menyimpan data dan meneruskannya ke halaman reupload pembayaran, jika tidak sistem akan mengembalikan ke halaman reupload pembayaran. Untuk lebih jelasnya dapat dilihat pada gambar berikut ini:



Gambar 3-19: Flowchart Proses Cetak Tiket, Cetak Sertifikat, ReUpload Pembayaran

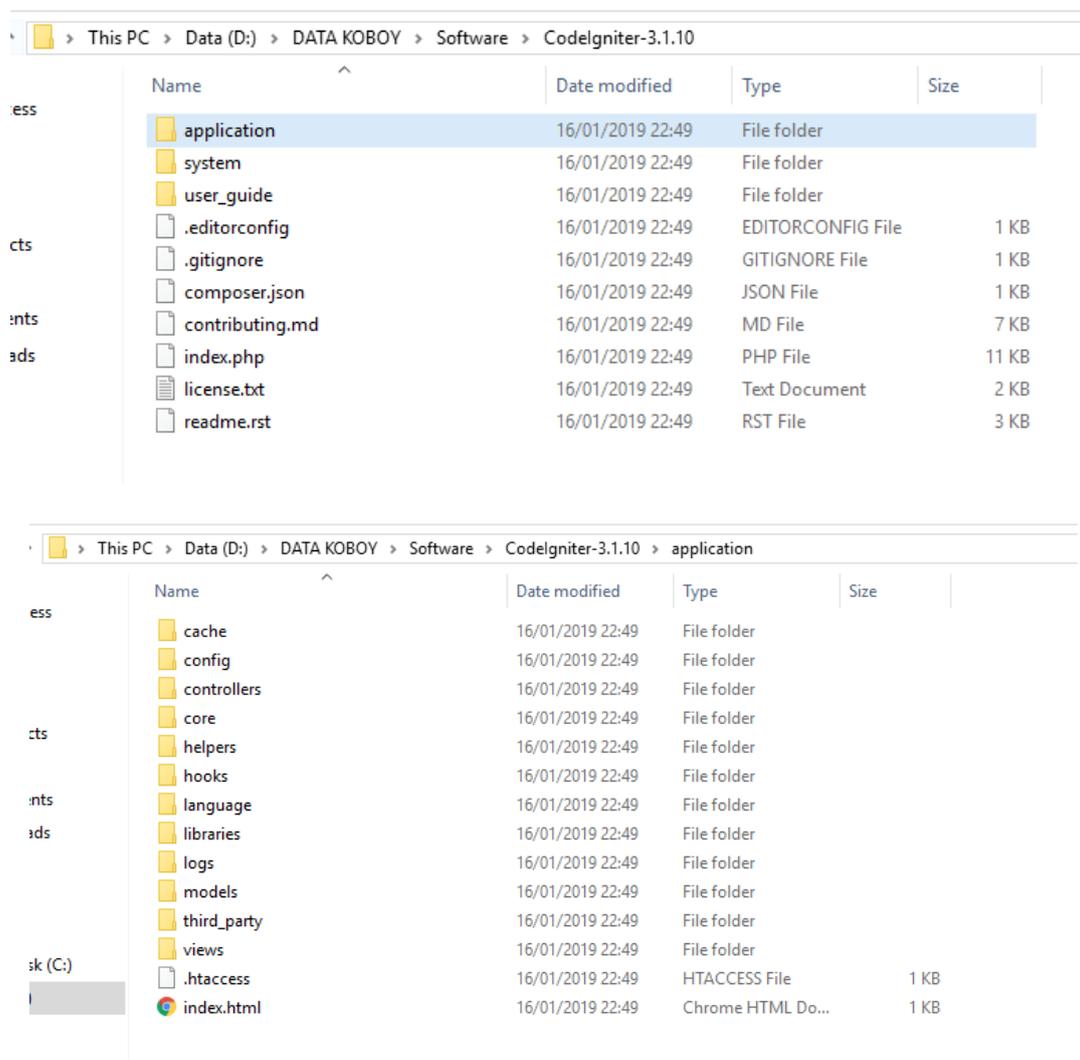
BAB IV

PENGUJIAN DAN IMPLEMENTASI

4.1 Pembahasan Cara Kerja *Codeigniter*

4.1.1 Struktur Direktori *Codeigniter*

Sebelum menggunakan CI, terlebih dahulu perlu anda lihat dan pahami struktur direktori CI. Tujuannya adalah agar bisa tepat dalam meletakkan file sesuai dengan fungsinya masing – masing.



Gambar 4-1: Struktur Direktori *Codeigniter*

Terdapat tiga folder utama CI, yaitu *application*, *system*, dan *user_guide*. Dari ketiga folder tersebut, yang perlu diperhatikan lebih adalah folder *application* karena di sinilah direktori kerja CI dan disini pula tempat anda meletakkan semua file yang berkaitan dengan aplikasi yang akan dikembangkan. Dalam folder *application* ini terdapat beberapa subfolder dengan peran masing-masing. Beberapa subfolder inilah yang nantinya digunakan untuk mengelompokkan file aplikasi yang akan dibuat sesuai dengan fungsinya.

Dari sekian banyak subfolder yang ada, terlebih dahulu fokuskan perhatian anda pada empat subfolder utama, yaitu :

- a. *Config*, digunakan untuk menyimpan berbagai macam file konfigurasi sistem aplikasi yang akan dibuat.
- b. *Controllers*, digunakan untuk meletakkan *controller* sistem aplikasi yang akan dibuat.
- c. *Models*, digunakan untuk meletakkan file *model* sistem aplikasi yang akan dibuat.
- d. *Views*, digunakan untuk meletakkan file *view* sistem aplikasi yang akan dibuat.

Dengan bermodalkan empat folder tersebut, anda sudah dapat membuat aplikasi PHP menggunakan framework CI.

4.1.2 MVC Pada Codeigniter

MVC (Model View Controller) adalah sebuah pola rancangan yang memungkinkan para pengembang memisahkan kode program mereka ke dalam tiga kelompok, yaitu : *Model, View, Controller*.

a. *Model*

Model merupakan bagian yang didesain untuk bekerja dengan database. Model bisa juga diartikan sebagai bagian yang digunakan untuk membuat semacam prototipe data. Bagian model ini biasanya berisi berbagai query yang dialamatkan pada sebuah database. Berbagai query tersebut disesuaikan dengan kebutuhan data yang diolah melalui controller, sebelum akhirnya ditampilkan menggunakan view. File-file model diletakkan didalam folder application/models.

Menuliskan Model

Berikut adalah langkah-langkah dan cara penulisan sebuah model dalam framework CI.

- Buat sebuah file model dan letakkan pada direktori application/models, misalkan nama file tersebut adalah user_m.php
- Isi file user_m.php tersebut sesuai dengan format yang ditentukan CI.
- Nama class harus disesuaikan dengan nama file yang bersangkutan, diawali dengan huruf kapital, dan tanpa diikuti ekstensi file. Misalkan, nama file adalah user_m.php maka nama kelasnya adalah User_m
- Fungsi-fungsi spesifik dapat dituliskan sebagai method kelas dari model yang anda buat.

Memanggil Model

Untuk mempertahankan konsep, selayaknya model di panggil (loaded) melalui sebuah controller. Adapun sintaks yang digunakan adalah :

```
$this->load->model ('nama_model');
```

Jika anda meletakkan model pada sebuah subdirektori dalam direktori kerja model maka sintaks yang digunakan adalah:

```
$this->load->model ('nama_direktori/nama_model');
```

Menggunakan Method Pada Model

Seperti yang telah dijelaskan sebelumnya, didalam sebuah model, Anda dapat menambahkan beberapa method yang bertujuan untuk mengelola data dari database. Hasil pengelolaan data dari method tersebut dapat dipanggil ke dalam sebuah controller yang telah berinisiatif memanggil model (yan didalamnya terhdapat method tersebut). Misalnya akan menggunakan data yang dihasilkan oleh method `get_all()` dari model `user_m` maka baris program yang perlu dituliskan adalah :

```
$this->load->model ('user_m'); // inisialisasi
atau load model

$this->user_m->get_all; // memanggil method
get_all yang ada di user_m
```

b. Controller

Controller merupakan bagian yang digunakan untuk menempatkan fungsi-fungsi logika pengolah data yang telah diperoleh dari model (atau sumber lainnya), kemudian ditampilkan melalui (atau tanpa) view. Inti

dari program yang dibuat dengan CI terletak pada controller. Melalui controller inilah request HTTP akan diolah.

Nantinya, nama kelas dan method beserta parameternya yang terdapat dalam controller akan digunakan dalam pengasosiasian URI sehingga pemahaman tentang controller ini mutlak dibutuhkan. File Controller diletakkan dibawah folder **application/controllers/**.

Menuliskan Controller

Berikut adalah format, cara dan contoh penulisan sebuah controller:

- Buat sebuah file controller dengan nama user.php, letakkan file tersebut didalam application/controllers.
- Isi file tersebut dengan format standar penulisan controller.
- Sebagaimana model, nama kelas pada controller juga harus sesuai dengan nama file, diawali huruf kapital dan tanpa disertai ekstensi file.

Cara Kerja Controller

Pada CI, controller sangat berpengaruh terhadap pengasosiasian URI suatu aplikasi. Dengan kata lain, URI berkaitan erat dengan controller. Pada umumnya, format penulisan untuk mengakses sebuah halaman PHP adalah

*http://nama_domain.com/nama_folder/nama_file.php?parameter=n
ilai*

Dengan menggunakan CI, format pengaksesannya sedikit berbeda. Berikut adalah format pengaksesan halaman PHP yang menggunakan framework CI.

```
http://nama_domain.com/index.php/nama_controller/nama_method/  
parameter_metod
```

Default Controller

Dalam CI terdapat istilah default controller, yaitu controller yang akan diakses pertama kali pada saat aplikasi dijalankan. Default controller ini diatur melalui file `routes.php` yang berada didalam folder `application/config/`. Baris program yang menjadi titik pengaturan default controller adalah:

```
$route['default_controller'] = "user";
```

Dengan menggunakan pengaturan (setting) diatas, pada saat aplikasi dibuka maka halaman diarahkan menuju controller user.

c. *View*

View adalah bagian presentasi yang berfungsi sebagai antarmuka program aplikasi dengan pengguna. View akan menampilkan data yang telah diolah sebelumnya melalui controller. Penggunaan view tergantung pada controller, view mana yang akan digunakan dan data apa saja yang akan ditampilkan. Pada bagian view biasanya tidak terdapat terdapat fungsi logika sama sekali. Yang ada hanyalah kode-kode HTML dan sebagian kecil kode PHP yang digunakan untuk melakukan parsing data. File view dalam CI diletakkan di bawah folder **application/view**.

Menuliskan View

Sebenarnya tidak ada yang istimewa dalam penulisan view. Membuat view sama halnya dengan membuat halaman HTML standar, tetapi ekstensi yang digunakan adalah *.php.

Cara Memanggil View

View dipanggil melalui sebuah method dari suatu controller. View dipanggil dengan menggunakan sintaks sebagai berikut:

```
$this->load->view('nama_view');
```

4.2 Spesifikasi *Hardware* dan *Software*

4.2.1 Spesifikasi *Hardware*

Berikut ini adalah spesifikasi yang digunakan untuk menjalankan aplikasi ini:

Tabel 4-1: Kebutuhan Perangkat Keras Untuk Pengembangan

Kebutuhan Perangkat Keras Untuk Pengembangan	
Nama Perangkat	Spesifikasi
<i>Processor</i>	2,50 – 2,70 GHz
<i>Memory</i>	4 GB
<i>VGA</i>	1 GB
<i>Hardisk</i>	20 GB
<i>Mouse</i>	1 <i>Unit</i>
<i>Keyboard</i>	1 <i>Unit</i>

Berikut ini adalah kebutuhan perangkat keras untuk pengguna:

Tabel 4-2: Kebutuhan Perangkat Keras Untuk Pengguna

Kebutuhan Perangkat Keras Untuk Pengguna	
Nama Perangkat	Spesifikasi
<i>Processor</i>	2,20 GHz
<i>Memory</i>	2 GB
<i>VGA</i>	512 MB
<i>Hardisk</i>	10 GB

4.2.2 Spesifikasi Software

Berikut ini adalah kebutuhan perangkat lunak dalam penembangan aplikasi:

Tabel 4-3: Kebutuhan Perangkat Lunak Untuk Pengembang

Kebutuhan Perangkat Lunak Untuk Pengembang	
Nama Software	Spesifikasi
<i>Operating System</i>	<i>Windows 7 Windows 8 Windows 10</i>
<i>XAMPP</i>	<i>Apache MySQL</i>
<i>Text Editor</i>	<i>Visual Studio Code</i>
<i>Browser</i>	<i>Google Chrome</i>

Berikut ini adalah kebutuhan perangkat lunak yang digunakan pengguna di antaranya sebagai berikut:

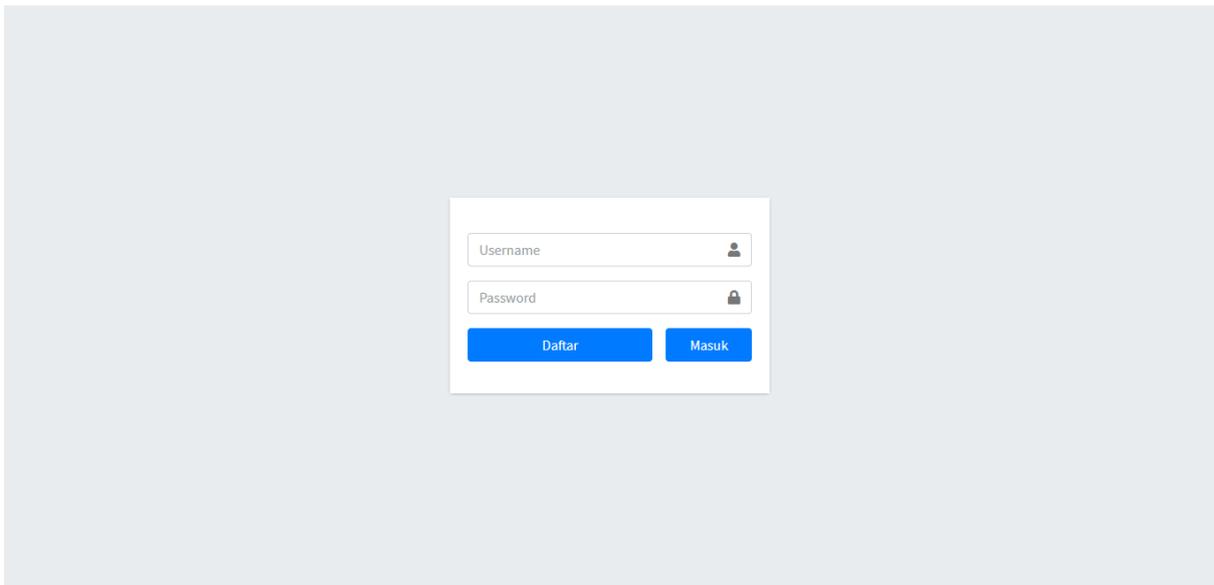
Tabel 4-4 : Kebutuhan Perangkat Lunak Untuk Pengguna

Kebutuhan Perangkat Lunak Untuk Pengguna	
Nama Software	Spesifikasi
<i>Operating System</i>	<i>MacOS Linux Android Windows 7 Windows 8 Windows 10</i>
<i>Browser</i>	<i>Microsoft Edge Google Chrome Mozilla Firefox Opera Safari Internet Explorer</i>

4.3 Tampilan Program

4.3.1 Tampilan Login

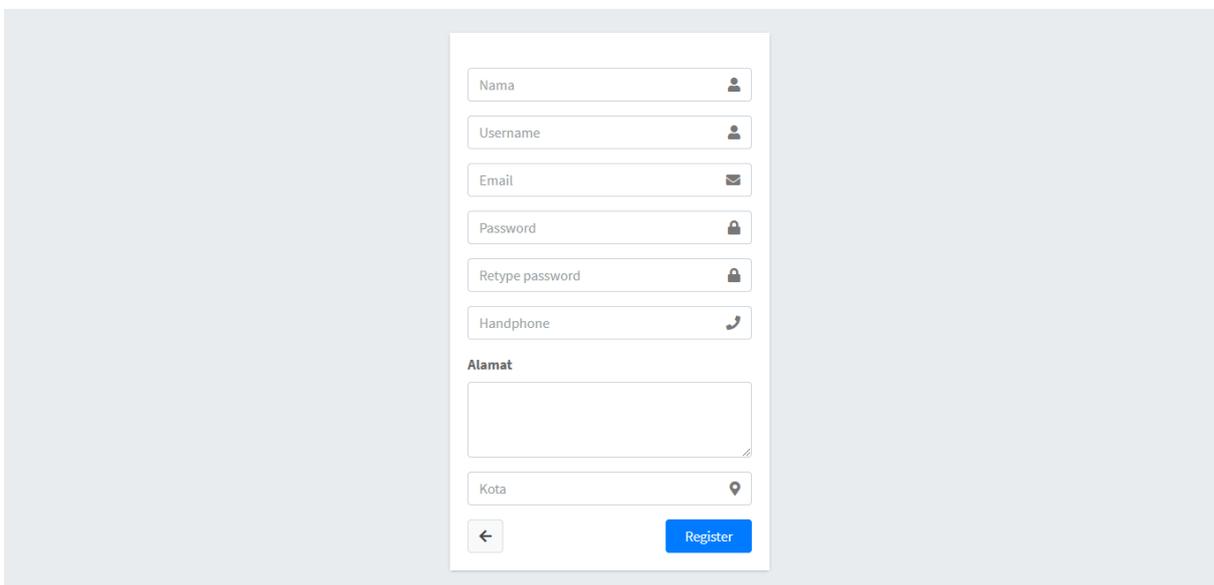
Berikut ini adalah tampilan halaman *login* dalam aplikasi ini, untuk lebih jelasnya dapat dilihat pada gambar berikut ini:



Gambar 4-2: Tampilan Login

4.3.2 Tampilan Register

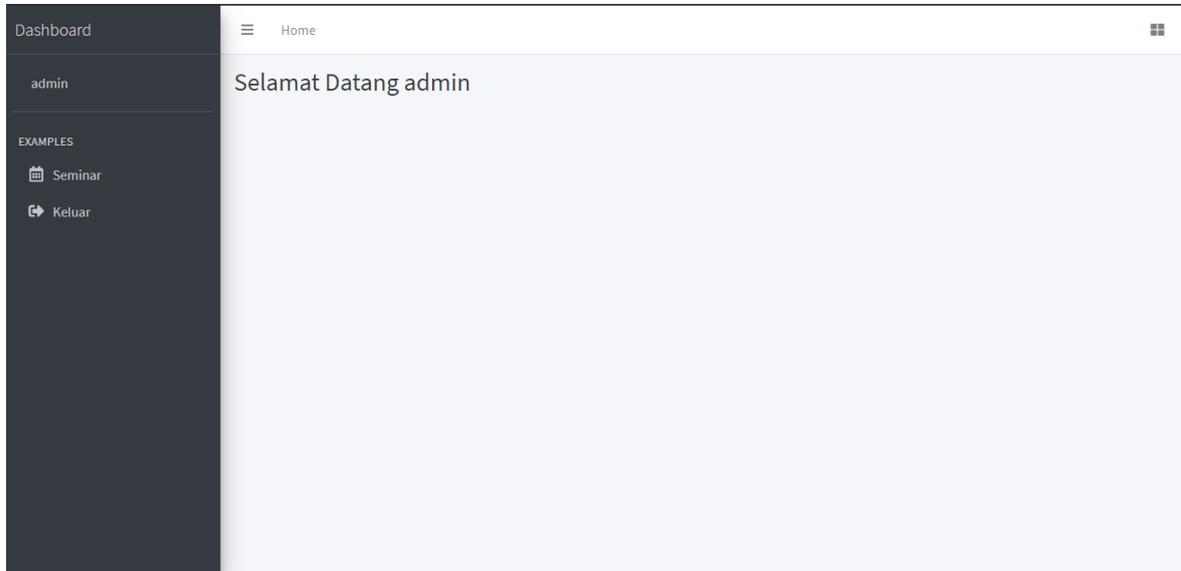
Berikut ini adalah tampilan halaman register dalam aplikasi ini, untuk lebih jelasnya dapat dilihat pada gambar berikut ini:



Gambar 4-3: Tampilan Register

4.3.3 Tampilan Dashboard Admin

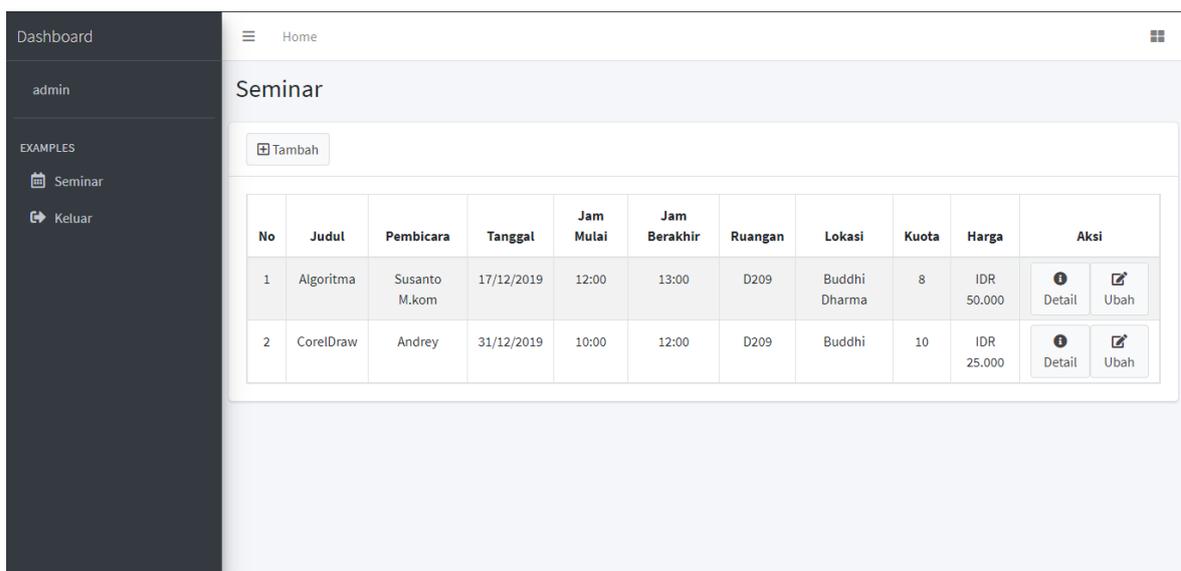
Berikut ini adalah tampilan halaman *dashboard admin* dalam aplikasi ini, untuk lebih jelasnya dapat dilihat pada gambar berikut ini:



Gambar 4-4: Tampilan Dashboard Admin

4.3.4 Tampilan List Seminar Admin

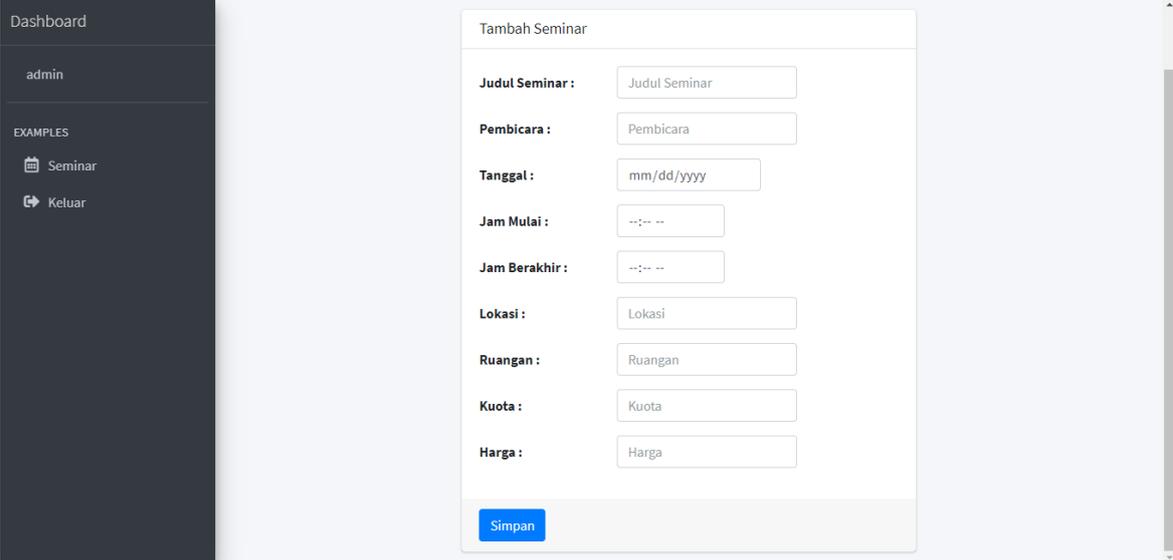
Berikut ini adalah tampilan halaman list seminar admin dalam aplikasi ini, untuk lebih jelasnya dapat dilihat pada gambar berikut ini:



Gambar 4-5: Tampilan List Seminar Admin

4.3.5 Tampilan Tambah Seminar

Berikut ini adalah tampilan halaman tambah seminar dalam aplikasi ini, untuk lebih jelasnya dapat dilihat pada gambar berikut ini:

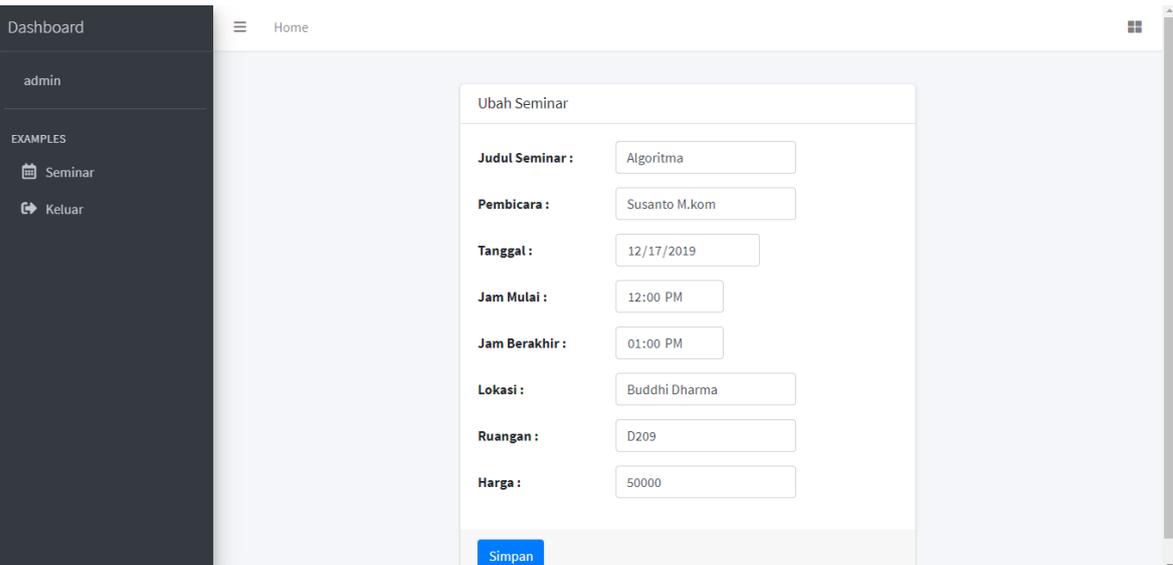


The screenshot shows a web application interface with a dark sidebar on the left and a main content area. The sidebar contains the text 'Dashboard', 'admin', and 'EXAMPLES' with sub-items 'Seminar' and 'Keluar'. The main content area displays a form titled 'Tambah Seminar'. The form fields are: 'Judul Seminar' (placeholder: Judul Seminar), 'Pembicara' (placeholder: Pembicara), 'Tanggal' (placeholder: mm/dd/yyyy), 'Jam Mulai' (placeholder: --:-- --), 'Jam Berakhir' (placeholder: --:-- --), 'Lokasi' (placeholder: Lokasi), 'Ruangan' (placeholder: Ruangan), 'Kuota' (placeholder: Kuota), and 'Harga' (placeholder: Harga). A blue 'Simpan' button is located at the bottom of the form.

Gambar 4-6: Tampilan Tambah Seminar

4.3.6 Tampilan Ubah Seminar

Berikut ini adalah tampilan halaman ubah seminar dalam aplikasi ini, untuk lebih jelasnya dapat dilihat pada gambar berikut ini:



The screenshot shows the same web application interface as in Gambar 4-6. The main content area displays a form titled 'Ubah Seminar'. The form fields are: 'Judul Seminar' (value: Algoritma), 'Pembicara' (value: Susanto M.kom), 'Tanggal' (value: 12/17/2019), 'Jam Mulai' (value: 12:00 PM), 'Jam Berakhir' (value: 01:00 PM), 'Lokasi' (value: Buddhi Dharma), 'Ruangan' (value: D209), and 'Harga' (value: 50000). A blue 'Simpan' button is located at the bottom of the form.

Gambar 4-7: Tampilan Ubah Seminar

4.3.7 Tampilan Detail Pendaftar Seminar

Berikut ini adalah tampilan halaman detail pendaftar seminar dalam aplikasi ini, untuk lebih jelasnya dapat dilihat pada gambar berikut ini:

Judul Seminar	Nama	Pembayaran	Deskripsi	Email	Handphone	Status	Kehadiran	Nomor Serifikat	Aksi
Algoritma	dermawan		Pembayaran seminar	dermawanoboy@gmail.com	081111997666	PAID	hadir	SEM0001	Bayar, i status, kode
Algoritma	andrey		Pembayaran seminar	andrey@gmail.com	082111997666	UNPAID	tidak		Bayar, i status, kode

Gambar 4-8: Tampilan Detail Pendaftar Seminar

4.3.8 Tampilan Beranda *User*

Berikut ini adalah tampilan halaman beranda *user* dalam aplikasi ini, untuk lebih jelasnya dapat dilihat pada gambar berikut ini:

Seminar

Java
Hendry
 📅 Tanggal: 31-01-2020
 ⌚ Mulai: 10:00 - 12:00
 📍 Lokasi: Buddhi
 📍 Ruangan: D209
 🗳️ Kuota: 14
 💰 IDR 50.000

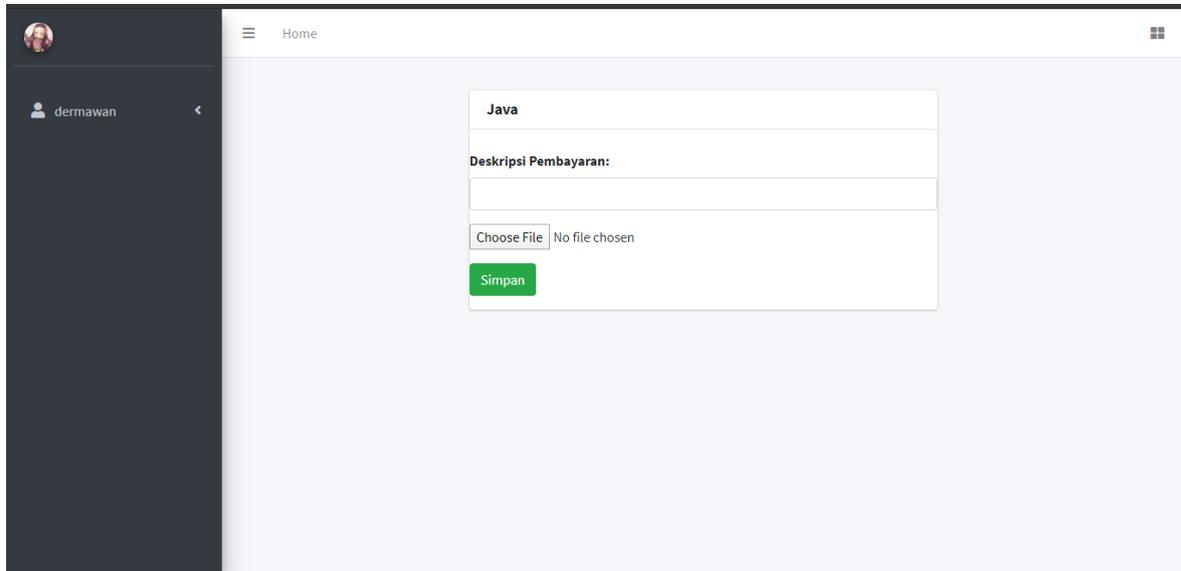
[Daftar Transfer](#) [Daftar Cash](#)

Hacker
Mulyono
 📅 Tanggal: 30-01-2020
 ⌚ Mulai: 10:00 - 12:00
 📍 Lokasi: Buddhi
 📍 Ruangan: D209
 🗳️ Kuota: 15
 💰 IDR 35.000

Gambar 4-9: Tampilan Beranda *User*

4.3.9 Tampilan Daftar Transfer

Berikut ini adalah tampilan halaman daftar via transfer dalam aplikasi ini, untuk lebih jelasnya dapat dilihat pada gambar berikut ini:

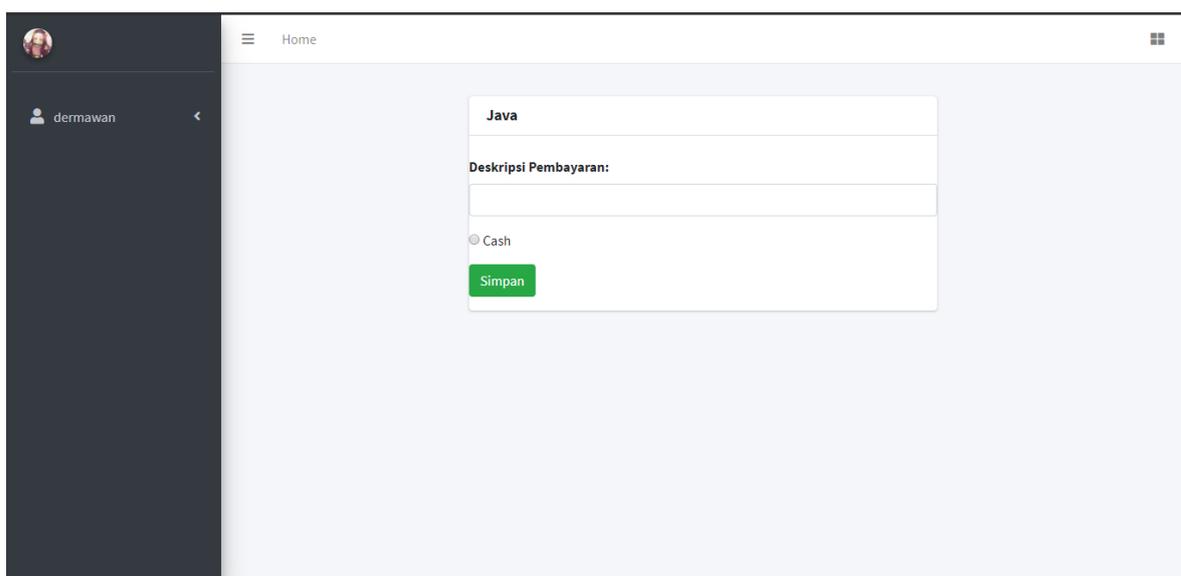


The screenshot shows a mobile application interface with a dark sidebar on the left containing a profile picture and the name 'dermawan'. The main content area is titled 'Home' and features a form for recording a transfer. The form includes a text input field with the value 'Java', a label 'Deskripsi Pembayaran:' followed by an empty text input field, a file selection area with a 'Choose File' button and the text 'No file chosen', and a green 'Simpan' (Save) button at the bottom.

Gambar 4-10: Tampilan Daftar Transfer

4.3.10 Tampilan Daftar Cash

Berikut ini adalah tampilan halaman daftar via cash dalam aplikasi ini, untuk lebih jelasnya dapat dilihat pada gambar berikut ini:

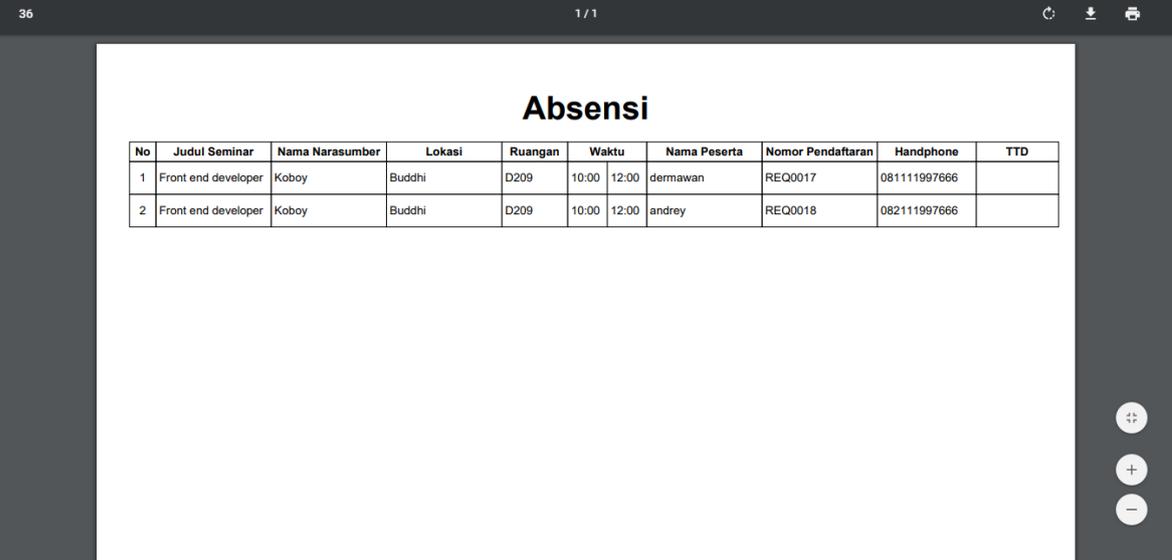


The screenshot shows the same mobile application interface as in Gambar 4-10. The form for recording a cash payment is displayed. It includes a text input field with the value 'Java', a label 'Deskripsi Pembayaran:' followed by an empty text input field, a radio button labeled 'Cash' which is currently selected, and a green 'Simpan' (Save) button at the bottom.

Gambar 4-11: Tampilan Daftar Transfer

4.3.11 Tampilan Cetak Absen Seminar

Berikut ini adalah tampilan halaman cetak absen seminar dalam aplikasi ini, untuk lebih jelasnya dapat dilihat pada gambar berikut ini:

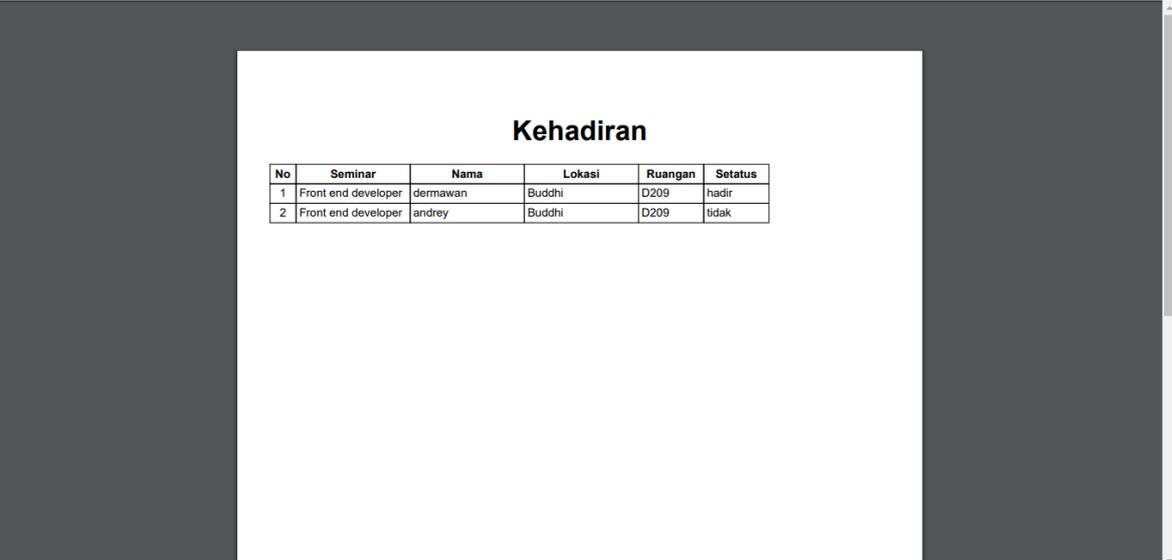


No	Judul Seminar	Nama Narasumber	Lokasi	Ruangan	Waktu	Nama Peserta	Nomor Pendaftaran	Handphone	TTD
1	Front end developer	Koboy	Buddhi	D209	10:00 - 12:00	dermawan	REQ0017	081111997666	
2	Front end developer	Koboy	Buddhi	D209	10:00 - 12:00	andrey	REQ0018	082111997666	

Gambar 4-12: Tampilan Cetak Absen Seminar

4.3.12 Tampilan Cetak Kehadiran

Berikut ini adalah tampilan halaman cetak kehadiran dalam aplikasi ini, untuk lebih jelasnya dapat dilihat pada gambar berikut ini:

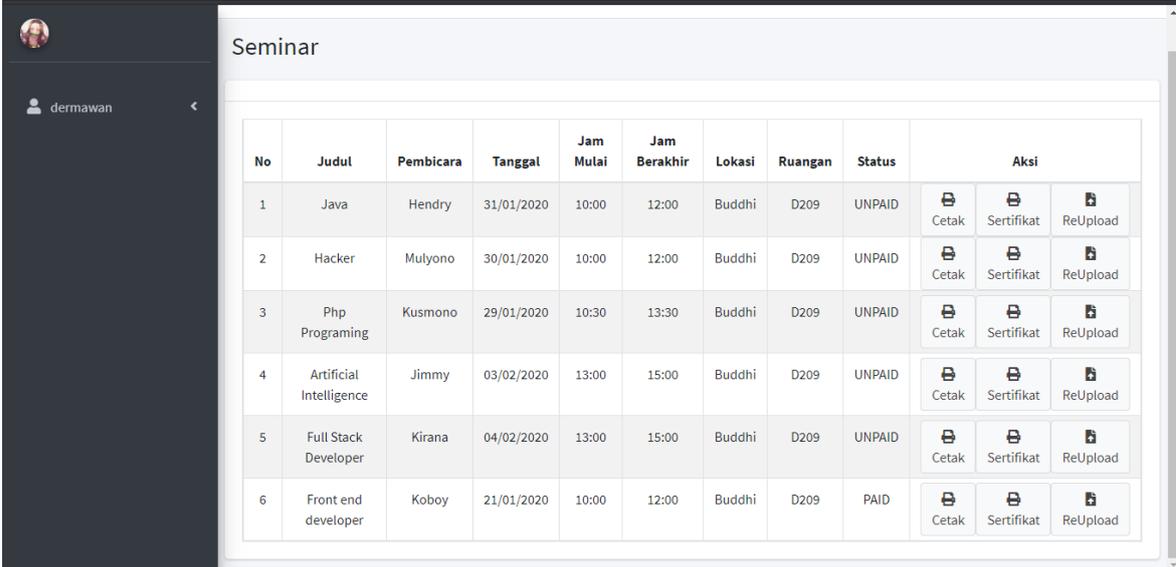


No	Seminar	Nama	Lokasi	Ruangan	Setatus
1	Front end developer	dermawan	Buddhi	D209	hadir
2	Front end developer	andrey	Buddhi	D209	tidak

Gambar 4-13: Tampilan Cetak Kehadiran

4.3.13 Tampilan List Seminar User

Berikut ini adalah tampilan halaman list seminar *user* dalam aplikasi ini, untuk lebih jelasnya dapat dilihat pada gambar berikut ini:

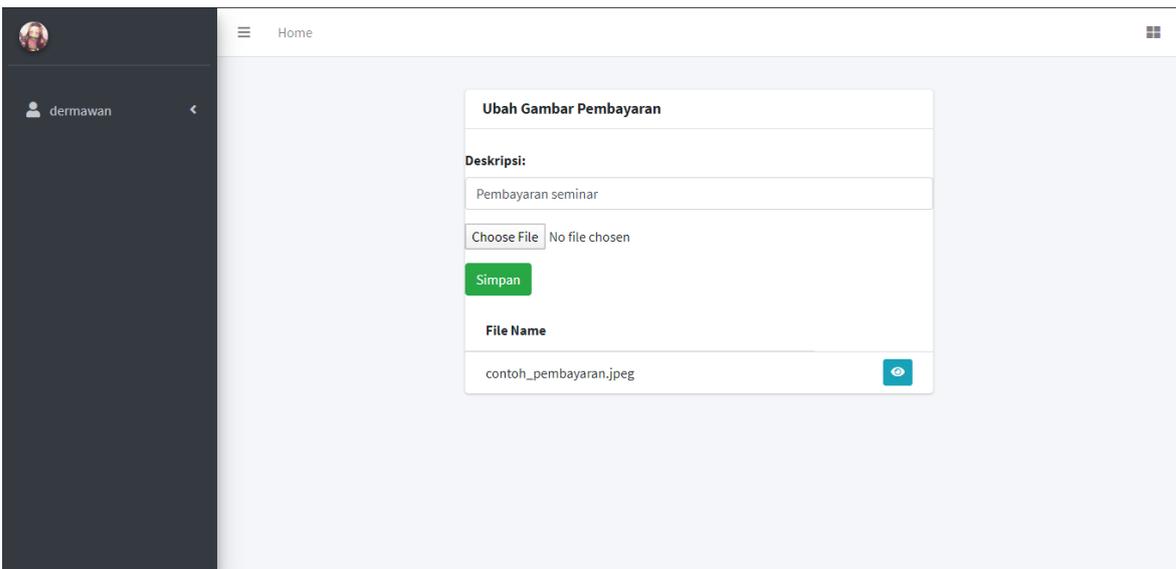


No	Judul	Pembicara	Tanggal	Jam Mulai	Jam Berakhir	Lokasi	Ruangan	Status	Aksi		
1	Java	Hendry	31/01/2020	10:00	12:00	Buddhi	D209	UNPAID	Cetak	Sertifikat	ReUpload
2	Hacker	Mulyono	30/01/2020	10:00	12:00	Buddhi	D209	UNPAID	Cetak	Sertifikat	ReUpload
3	Php Programing	Kusmono	29/01/2020	10:30	13:30	Buddhi	D209	UNPAID	Cetak	Sertifikat	ReUpload
4	Artificial Intelligence	Jimmy	03/02/2020	13:00	15:00	Buddhi	D209	UNPAID	Cetak	Sertifikat	ReUpload
5	Full Stack Developer	Kirana	04/02/2020	13:00	15:00	Buddhi	D209	UNPAID	Cetak	Sertifikat	ReUpload
6	Front end developer	Koboy	21/01/2020	10:00	12:00	Buddhi	D209	PAID	Cetak	Sertifikat	ReUpload

Gambar 4-14: Tampilan List Seminar User

4.3.14 Tampilan ReUpload Pembayaran

Berikut ini adalah tampilan halaman *ReUpload* pembayar dalam aplikasi ini, untuk lebih jelasnya dapat dilihat pada gambar berikut ini:



Home

Ubah Gambar Pembayaran

Deskripsi:
Pembayaran seminar

Choose File No file chosen

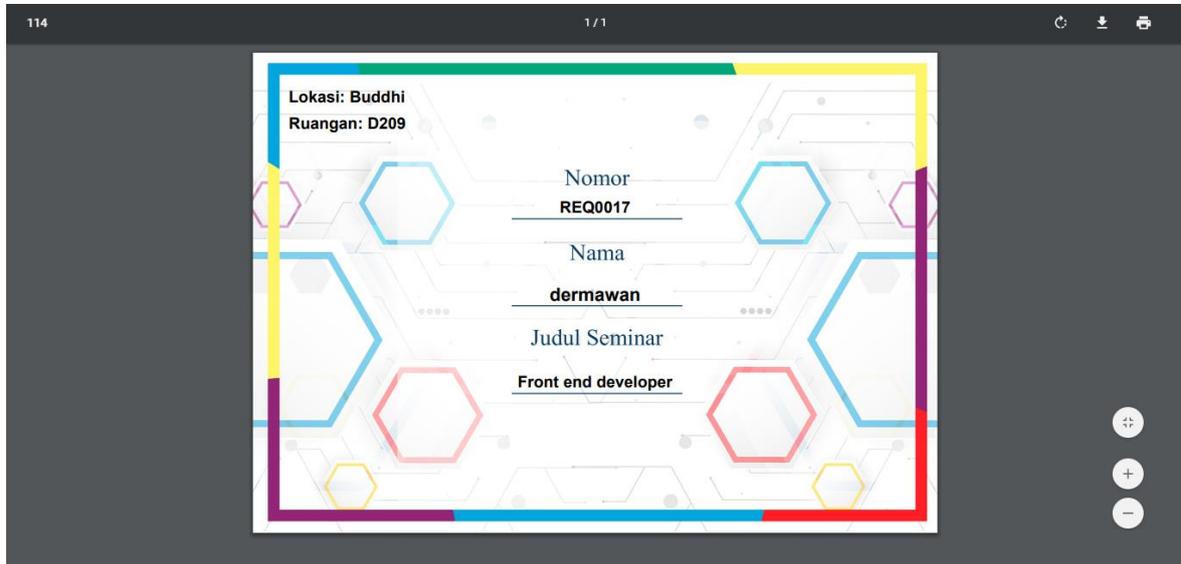
Simpan

File Name
contoh_pembayaran.jpeg

Gambar 4-15: Tampilan ReUpload Pembayaran

4.3.15 Tampilan Cetak Nomor Pendaftaran

Berikut ini adalah tampilan halaman cetak nomor pendaftaran dalam aplikasi ini, untuk lebih jelasnya dapat dilihat pada gambar berikut ini:



Gambar 4-16: Tampilan Cetak Nomor Pendaftaran

4.3.16 Tampilan Cetak Sertifikat

Berikut ini adalah tampilan halaman cetak sertifikat dalam aplikasi ini, untuk lebih jelasnya dapat dilihat pada gambar berikut ini:



Gambar 4-17: Tampilan Cetak Sertifikat

4.4 Pengujian Aplikasi

4.4.1 *Blackbox Testing*

Pengujian *Blackbox* dilakukan dengan tes fungsionalitas dari aplikasi atau perangkat lunak. Tujuan pengujian ini adalah untuk menemukan kesalahan dalam aplikasi. Berikut adalah tabel hasil pengujian *blackbox*:

Tabel 4-5: Hasil Pengujian *Blackbox*

No	Scene Pengujian	Skenario	Hasil yang diharapkan	Hasil Pengujian
1	Halaman Login	Pengguna menginput Username dan Password dan menekan tombol login untuk melakukan login atau menekan tombol daftar untuk pindah ke halaman daftar	Jika Username dan Password terdapat dalam database level 1 maka akan masuk ke halaman dashboard admin dan bila level 2 maka akan masuk ke halaman beranda user dan jika user menekan tombol daftar maka akan membuka halaman register	Valid
2	Halaman Register	Pengguna menginput data diri lalu setelah itu menekan tombol register atau menekan tombol back	Jika data yang di input kosong maka akan tampil pesan bahwa data harus dilengkapi terlebih dahulu dan jika input username dan email sama maka akan muncul pesan username / email telah terdaftar dan	Valid

No	Scene Pengujian	Skenario	Hasil yang diharapkan	Hasil Pengujian
			jika pada saat input password dan retype password tidak cocok maka akan muncul pesan password yang ada masukan tidak sama dan jika semua kondisi benar maka akan muncul pesan anda telah terdaftar dan kembali ke halaman login	
3	Halaman Dashboard Admin	Pengguna menekan tombol home atau seminar atau logout	Bila pengguna menekan menu home maka pengguna akan masuk ke halaman beranda dashboard dan jika pengguna menekan tombol seminar maka pengguna masuk ke halaman list seminar admin dan jika pengguna menekan tombol logout maka pengguna akan keluar dari dashboard dan pindah ke halaman login	Valid
4	Halaman List Seminar Admin	Pengguna menekan tombol tambah atau tombol detail atau tombol ubah	Bila pengguna menekan tombol tambah maka pengguna akan masuk ke halaman tambah	Valid

No	Scene Pengujian	Skenario	Hasil yang diharapkan	Hasil Pengujian
			seminar dan jika pengguna menekan tombol detail maka pengguna akan masuk ke halaman detail pendaftar seminar dan jika pengguna menekan tombol ubah maka pengguna akan masuk ke halaman ubah seminar	
5	Halaman Tambah Seminar	Pengguna melakukan penginputan data seminar dan menekan tombol simpan	Jika data yang di input kosong maka akan muncul pesan bahwa data harus dilengkapi terlebih dahulu dan jika data yang di input benar maka muncul pesan seminar berhasil ditambah dan kembali ke halaman list seminar admin	Valid
6	Halaman Ubah Seminar	Pengguna melakukan perubahan data seminar dan menekan tombol simpan	Jika perubahan data dilakukan maka data akan berubah sesuai dengan data yang dirubah oleh pengguna	Valid
7	Halaman Detail Pendaftar Seminar	Pengguna menekan tombol report absensi atau tombol report kehadiran atau tombol bayar atau tombol status atau tombol kode	jika menekan tombol absen maka akan menampilkan halaman cetak absen, jika menekan	Valid

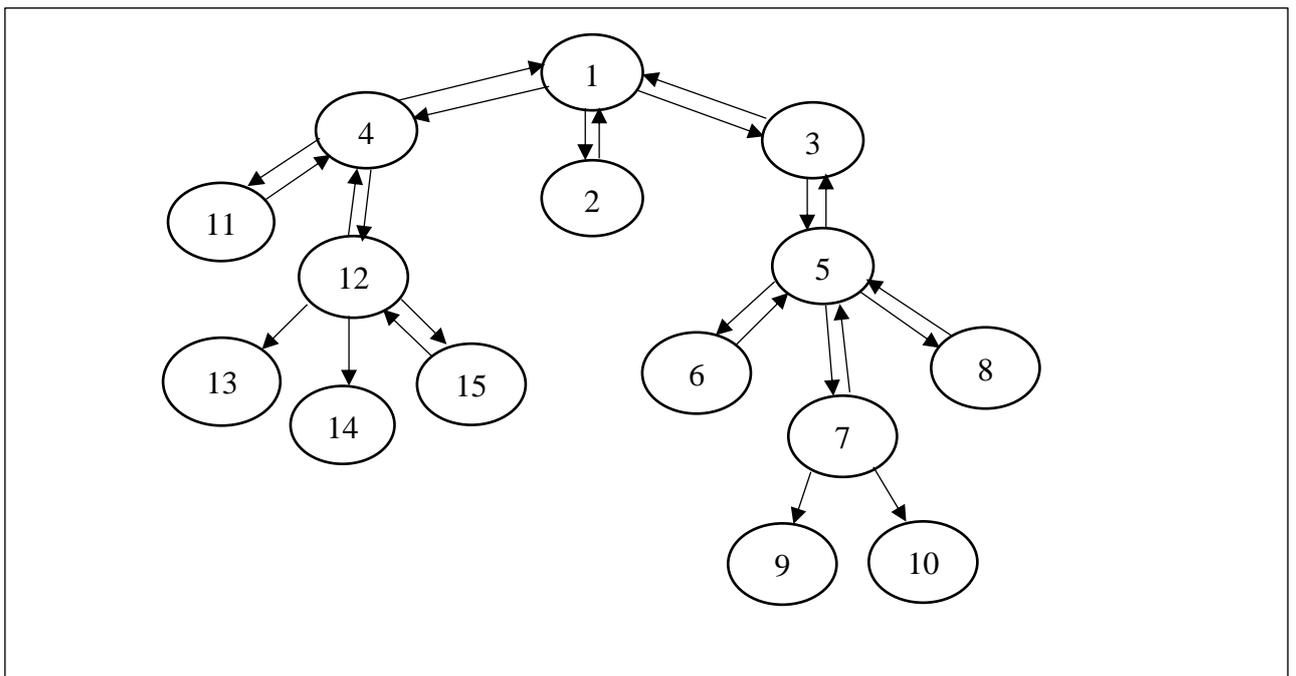
No	Scene Pengujian	Skenario	Hasil yang diharapkan	Hasil Pengujian
		atau tombol back atau mengklik gambar pada tabel	tombol kehadiran maka akan tampil cetak daftar penonton seminar yang hadir dan juga yang tidak hadir, jika menekan tombol bayar akan mengubah dari unpaid menjadi paid dan menyimpan status pembayaran, jika menekan tombol status maka akan mengubah status peserta dari tidak menjadi hadir, jika menekan tombol kode akan meminta kode seminar untuk cetak sertifikat.	
8	Halaman Beranda User	Pengguna menekan tombol seminar untuk melihat seminar yang telah pengguna daftar tombol logout untuk keluar aplikasi dan tombol daftar untuk daftar seminar dan home untuk ke beranda	jika menekan tombol seminar akan tampil list seminar user, jika menekan tombol logout akan kembali ke halaman login, jika menekan tombol daftar akan menampilkan form pendaftaran dan jika	Valid

No	Scene Pengujian	Skenario	Hasil yang diharapkan	Hasil Pengujian
			pengguna menekan tombol home maka akan menampilkan homepage user	
9	Halaman List Seminar User	pengguna menekan tombol home untuk menampilkan beranda, pengguna menekan tombol cetak untuk mencetak tiket, pengguna menekan tombol sertifikat untuk mencetak sertifikat, dan tombol reupload untuk menampilkan halaman reupload	jika pengguna menekan tombol home maka akan menampilkan beranda, jika pengguna menekan tombol cetak maka aplikasi akan membuka tab baru untuk mencetak tiket, jika pengguna menekan tombol sertifikat maka aplikasi akan membuka tab baru untuk mencetak sertifikat jika pengguna menekan tombol reupload maka aplikasi akan menampilkan halaman reupload	Valid
10	Halaman ReUpload Pembayaran	pengguna menekan tombol home untuk kembali ke beranda aplikasi, jika pengguna menekan tombol choose file untuk memilih gambar untuk diunggah, jika pengguna menekan tombol ikon mata melihat gambar	jika pengguna menekan home maka aplikasi akan menampilkan halaman beranda, jika pengguna menekan tombol choose file maka akan membuka	Valid

No	Scene Pengujian	Skenario	Hasil yang diharapkan	Hasil Pengujian
		yang telah diubah, jika pengguna menekan tombol simpan maka akan menyimpan perubahan.	jendela untuk memilih file yang akan diunggah, jika pengguna menekan ikon mata maka akan membuka tab baru untuk melihat gambar bukti pembayaran, jika pengguna menekan tombol simpan akan menyimpan perubahan dan memperbaharui halaman reupload.	

4.4.2 Whitebox Testing

Dalam pengujian *whitebox testing* ini terdapat 15 simpul (*Node*) dan terdapat 31 jalur (*Edge*) dimana terdapat pada gambar dibawah ini:



Berikut ini adalah keterangan halaman yang terdapat dalam simpul gambar di atas diantaranya:

Node	Keterangan
1	Halaman login
2	Halaman Register
3	Halaman Dashboard Admin
4	Halaman Homepage User
5	Halaman Seminar Admin
6	Halaman Tambah Seminar
7	Halaman Detail Pendaftar Seminar
8	Halaman Ubah Seminar
9	Halaman Cetak Absensi Seminar
10	Halaman Cetak Kehadiran Peserta Seminar
11	Halaman Daftar Seminar
12	Halaman List Seminar User
13	Halaman Cetak Nomor Pendaftaran
14	Halaman Cetak Sertifikat
15	Halaman ReUpload Pembayaran

Langkah berikutnya adalah menentukan kompleksitas *cyclomatic* dari grafik aliran yang digunakan untuk mencari banyaknya jalur (*Path*) didalam aplikasi ini, berikut ini adalah rumus kompleksitas *cyclomatic* :

$$V(G) = E - N + 2$$

$$= 24 - 15 + 2$$

$$= 11$$

Setelah ini menentukan langkah-langkah 18 jalur tersebut ini diantaranya:

Path	Langkah-langkah Sistematis Jalur
1	1-2-1
2	1-4-1
3	1-4-11-4-1
4	1-4-12-13
5	1-4-12-14
6	1-4-12-15-12-4-1
7	1-3-1
8	1-3-5-6-5-3-1
9	1-3-5-7-9
10	1-3-5-7-10
11	1-3-5-8-5-3-1

BAB V

SIMPULAN DAN SARAN

5.1 Simpulan

Setelah dilakukan perancangan, pembuatan dan pengujian maka dapat diambil kesimpulan sebagai berikut:

1. Dengan adanya aplikasi pendaftaran seminar ini, Dapat membantu dan mempermudah pengguna dalam mendapatkan informasi tentang seminar dan juga dalam melakukan pendaftaran seminar lalu dapat mencetak sertifikat secara langsung.
2. Dengan metode *mvc* proses pembuatan aplikasi pendaftaran seminar dapat diimplementasikan.
3. Dengan metode *mvc* proses pengembangan menjadi lebih mudah.
4. Dengan *framework codeigniter* aplikasi *web* pendaftaran seminar menjadi *web* dinamis.

5.2 Saran

Dalam penerapannya, Aplikasi ini masih jauh dari kata sempurna dan masih banyak kekurangan. Dan oleh karena itu agar pengguna aplikasi ini dapat lebih maksimal maka penulis memberikan beberapa saran antara lain:

1. Perlu dilakukan uji coba dengan menggunakan framework lain seperti Laravel, Yii, Dan lain-lain, Untuk membandingkan tingkat ke efektifan metode MVC dari masing-masing framework.
2. Membuat penyimpanan data secara *real time*.

3. Dalam pembayaran saat ini hanya berupa gambar, mungkin bisa dikembangkan dengan membuat fitur bank *virtual account*.

DAFTAR PUSTAKA

- Agarina, Melda, and Arman Suryadi Karim. 2019. "Rancangan Bangun Sistem Informasi Kegiatan Seminar Nasional Berbasis Web Pada Institut Informatics dan Bisnis Darmajaya." *Jurnal Sistem Informasi dan Telematika* 26-32.
- Aji, S.M. 2014. "Journal on Networking and Security." *Rancangan Bangun Sistem Informasi Disposisi Surat Berbasis Web* Volume 3 no 3.
- Anggraeni, Elisabet Yunaeti, and Rita Irviani. 2017. *Pengantar Sistem Informasi*. Yogyakarta: ANDI.
- Annis, Matt. 2014. *What Is a Database and How Do I Use It?* New York: Reese.
- Arief, M.Rudyanto. 2011. *Pemrograman Web Dinamis Menggunakan Php dan MySQL*. Yogyakarta: ANDI.
- Arifin, Hasnul. 2010. *Merakit Sendiri Komputer Tahan Banting dengan Modal 1 Jutaan*. Yogyakarta: MediaKom.
- Faisal, M Reza. 2017. *Seri Belajar ASP.NET: ASP.NET Core MVC & MySQL dengan Visual Studio Code*. <https://www.researchgate.net/publication/316597650>.
- Fathansyah. 2015. *Basis Data Revisi Kedua*. Bandung: Informatika.
- Fathurrahman. 2014. *Membuat Website Mudah dan Praktis dengan Weebly*. Jakarta: PT Elek Media Komputindo.
- Hakim, Lukmanul. 2010. *Membangun Web Berbasis PHP dengan Framework Codeigniter*. Yogyakarta: Lokomedia.
- Indrajani. 2015. *Database Design*. Jakarta: PT. Elex Media Komputindo.
- Jubilee Enterprise. 2017. *HTML 5 Komplet : Bacaan Komplet Untuk Membantu Untuk Memahami HTML Secara Lengkap*. Jakarta: PT. Elek Media Komputindo.
- Komputer, Wahana. 2010. *Panduan Belajar MySQL Database Server*. Jakarta: mediakita.
- Kustiyahningsih, Yeni, and Devie Rosa Anamisa. 2011. *Pemrograman Basis Data Berbasis Web Menggunakan PHP & MySQL : Edisi Pertama*. Yogyakarta: Graha Ilmu.
- Laba, I Nengah, and Ni Made Rinyanthi. 2018. *BUKU AJAR BAHASA INDONESIA BERBASIS KARYA TULIS ILMIAH*. Yogyakarta: CV BUDI UTAMA.
- Murhada, and Ceng Giap Yo. 2011. *Pengantar Teknologi Informasi*. Jakarta: Mitra Wacana Media.

- Muslihudin, Muhamad, and Oktafianto. 2016. *Analisis dan Perancangan Sistem Informasi Menggunakan Model Terstruktur dan UML*. Yogyakarta: ANDI.
- Nugroho, Bunafit. 2013. *Dasar Pemrograman Web PHP - MySQL dengan Dreamweaver*. Yogyakarta: Gaya Media.
- Nugroho, Dwi Arianto Adi, dan Heru Supriyono. 2019. "Sistem Informasi Pendaftaran Seminar Dengan Tiket Berbasis Qr Code." *Jurnal Emitor* 34-40.
- Oktavian, Diar Puji. 2010. *Menjadi Programmer Jempolan Menggunakan PHP*. Yogyakarta: MediaKom.
- Politwika. 2017. *60+ Cara Online Menghasilkan Uang Saat Weekend*. Jakarta: Gramedia Widiasarana Indonesia.
- Rizky, Soetam. 2011. *Konsep Dasar Rekayasa Perangkat Lunak*. Jakarta: Prestasi Pustaka.
- Rozi, Zaenal A., and SmitDev Community. 2105. *Bootstrap Design Framework*. Jakarta: PT Elex Media Komputindo.
- Sianipar, Eng RH. 2017. *Dasar Pemrograman Java Script: Langkah demi langkah*. Yogyakarta: ANDI Publisher.
- Sianipar, R. H. 2015. *PHP & MySQL*. Yogyakarta: CV. Andi Offset.
- Simanjuntak, Pastima, and Arwin Kasnady. 2016. "Analisis Model View Controller (MVC) Pada Bahasa PHP." *Jurnal ISD Vol.2 No.2* 56-66.
- Simarmata, Janner. 2010. *Rekayasa Perangkat Lunak*. Yogyakarta: Andi.
- Solichin, Achmad. 2005. *Pemrograman Web dengan PHP dan MySQL*. Jakarta: <https://www.researchgate.net/publication/236885805>.
- Subagia, Anton. 2018. *Menguasai Framework Codeigniter 3*. Cirebon: CV. ASFA Solution.
- Sugiyono. 2013. *Metode Penelitian Kuantitatif dan Kualitatif dan R&D*. Bandung: Alfabeta.
- Sutabri, Tata. 2012. *Analisis Sistem Informasi*. Yogyakarta: CV. Andi Offset.
- Sutanta, Edhy. 2011. *Basis Data dalam Tinjauan Konseptual*. Yogyakarta: Andi.
- Sya'ban, Wahyu. 2010. *Build Your Blogger XML Template*. Yogyakarta: Andi.
- Yudhanto, Yudho, and Helmi Adi Prasetyo. 2018. *Panduan Mudah Belajar Framework Laravel*. Jakarta: PT Elex Media Komputindo.
- Zahara, Rina , Bella Citra Larastia, and Lisa Nurmala. 2015. *White Box Testing - kompasiana.com*. June 26. Accessed January 05, 2020.

https://www.kompasiana.com/diskbrain_bibeh/55004dc4813311791bfa74d0/white-box-testing.

LAMPIRAN-LAMPIRAN

DAFTAR RIWAYAT HIDUP



Biodata Mahasiswa

Nama Lengkap : Dermawan
Tempat Tanggal Lahir : Tangerang, 02 Februari 1997
Jenis Kelamin : Laki-Laki
Alamat Lengkap : Jl.Mutiara Kp.Dongkal RT 005/005 RW Kel.Pondok Jagung Timur Kec.Serpong Utara Tangerang Selatan.
Agama : Budha
Nomor Telepon : 087889944600
E-Mail : dermawansevenfold@gmail.com

Pendidikan Formal

2003-2009 : SDN Jelupang 1
2009-2012 : SMPN 16 Tangerang Selatan
2012-2015 : SMK Setia Bhakti Tangerang
2015 – Sekarang : Program Studi Teknik Informatika, Peminatan Basis Data, Universitas Buddhi Dharma

Tangerang, 20 Desember 2019

A handwritten signature in black ink, appearing to read 'Dermawan'.

Dermawan
20151000093