

**CETAKIN : APLIKASI PERCETAKAN *ONLINE*
BERBASIS WEB**

PROPOSAL TUGAS AKHIR



Oleh :
HENDRIYANTO
NIM. 361555401019

**PROGRAM STUDI DIPLOMA III
TEKNIK INFORMATIKA
POLITEKNIK NEGERI BANYUWANGI
2018**

**CETAKIN : APLIKASI PERCETAKAN *ONLINE*
BERBASIS WEB**

PROPOSAL TUGAS AKHIR



**Tugas Akhir ini Dibuat dan Diajukan untuk Memenuhi Salah Satu Syarat
Kelulusan Program Studi Diploma III Teknik Informatika
dan Mencapai Gelar Ahli Madya (A.Md)**

Oleh :

HENDRIYANTO

NIM. 361555401019

**PROGRAM STUDI DIPLOMA III
TEKNIK INFORMATIKA
POLITEKNIK NEGERI BANYUWANGI
2018**

...Halaman ini sengaja dikosongkan...

LEMBAR PENGESAHAN
PROPOSAL TUGAS AKHIR

Judul : Cetakin : Aplikasi Percetakan *Online* Berbasis Web
Oleh : Hendriyanto
NIM. : 361555401019

Telah diseminarkan pada :

Hari : Kamis
Tanggal : 01 Februari 2018
Tempat : Lab. TUK

Mengetahui/Menyetujui

Dosen Penguji :

Dosen Pembimbing :

1. Farizqi Panduardi, S.ST.,M.T
NIK. 2013.36.106

1. Devit Suwardiyanto, S.Si.,M.T
NIP. 198311052015041001

2. I Wayan Suardinata, S.Kom.,M.T
NIP. 198010222015041001

2. Dianni Yusuf, S.Kom.,M.Kom
NIK. 2008.36.005

...Halaman ini sengaja dikosongkan...

DAFTAR ISI

LEMBAR PENGESAHAN	iii
DAFTAR ISI.....	v
DAFTAR GAMBAR.....	vii
DAFTAR TABEL	ix
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Perumusan Masalah	2
1.3 Tujuan.....	2
1.4 Manfaat	2
1.5 Batasan Masalah	3
BAB 2 TINJAUAN PUSTAKA.....	5
2.1 Percetakan	5
2.2 <i>Hypertext Markup Language (HTML)</i>	5
2.3 <i>Cascading Style Sheets (CSS)</i>	6
2.4 Pemrograman Web	7
2.4.1 <i>Hypertext Preprocessor (PHP)</i>	7
2.4.2 <i>JavaScript</i>	8
2.5 <i>Framework</i>	11
2.5.1 <i>CodeIgniter</i>	11
2.5.2 <i>Laravel</i>	13
2.5.3 <i>Yes It Is (YII)</i>	14
2.6 Basis data	14
2.6.1 <i>My Structured Query Language (MySQL)</i>	15
2.6.2 <i>Postgre SQL</i>	16
2.6.3 <i>Oracle</i>	17
2.7 Metode Pengembangan Sistem	18
2.7.1 <i>Matode Waterfall</i>	18
2.7.2 <i>Metode Rapid Aplication Development (RAD)</i>	19
2.7.3 <i>Metode Prorotype</i>	20
2.8 Konsep Pemodelan Sistem.....	21
2.8.1 <i>Diagram Use case</i>	22
2.8.2 <i>Diagram Activity</i>	24

2.8.3	<i>Entity Relationship Diagram (ERD)</i>	24
2.9	Pengujian	26
2.9.1	<i>Black Box Testing</i>	26
2.9.2	<i>White Box Testing</i>	26
2.10	Penelitian Terkait	27
BAB 3	29
METODE PENELITIAN	29
3.1	Waktu, Tempat, dan Jadwal Penelitian	29
3.1.1	Waktu Penelitian	29
3.1.2	Tempat Penelitian	29
3.1.3	Jadwal Penelitian.....	29
3.2	Metode yang Digunakan.....	30
3.2.1	Perencanaan Syarat – syarat.....	30
3.2.2	Desain Sistem	31
3.2.3	Pengujian.....	49
3.2.4	Implementasi Sistem	49
3.3	Gambaran Umum Sistem	50
3.3.1	Gambaran Sistem yang Berjalan.....	50
3.3.2	Gambaran Sistem yang Diusulkan	51
3.4	Struktur Menu.....	52
DAFTAR PUSTAKA	

DAFTAR GAMBAR

Gambar 3.1 Metode RAD Pengerjaan Tugas Akhir	30
Gambar 3.2 Diagram <i>Use Case</i>	32
Gambar 3.3 Diagram <i>Activity</i> Daftar	38
Gambar 3.4 Diagram <i>Activity</i> Pengisian Saldo	39
Gambar 3.5 Diagram <i>Activity</i> Pemesanan	41
Gambar 3.6 Diagram <i>Activity</i> Pemrosesan Pesanan	42
Gambar 3.7 <i>Entity Relationship Diagram</i>	44
Gambar 3.8 Gambaran Umum Sistem yang Berjalan	50
Gambar 3.9 Gambaran Umum Sistem yang Diusulkan	51
Gambar 3.10 Struktur Menu Aplikasi	52
Gambar 3.11 Halaman utama pelanggan.....	55
Gambar 3.12 Halaman Pilih Percetakan.....	55
Gambar 3.13 Halaman isi formulir pesanan	56
Gambar 3.14 Halaman menu daftar pesanan.....	56
Gambar 3.15 Halaman utama petugas percetakan.....	57
Gambar 3.16 Halaman menu daftar pesanan petugas percetakan	57

...Halaman ini sengaja dikosongkan...

DAFTAR TABEL

Tabel 2.1 Diagram <i>Use case</i>	22
Tabel 2.2 Diagram <i>Activity</i>	24
Tabel 2.3 <i>Entity Relationship Diagram</i>	25
Tabel 2.4 Penelitian Terkait	27
Tabel 3.1 Jadwal Kegiatan Proyek Akhir.....	29
Tabel 3.2 Data admin aplikasi	31
Tabel 3.3 Keterangan Aktor	32
Tabel 3.4 Deskripsi <i>Use Case</i> Mendaftar.....	33
Tabel 3.5 Deskripsi <i>Use Case</i> Verifikasi Data Percetakan	34
Tabel 3.6 Deskripsi <i>Use Case</i> Mengatur Jenis Layanan	34
Tabel 3.7 Deskripsi <i>Use Case</i> Mengatur Informasi Harga Cetak	35
Tabel 3.8 Deskripsi <i>Use Case</i> Melihat Daftar Percetakan	35
Tabel 3.9 Deskripsi <i>Use Case</i> Mengisi Saldo	36
Tabel 3.10 Deskripsi <i>Use Case</i> Melakukan Pemesanan	37
Tabel 3.11 Deskripsi <i>Use Case</i> Memeriksa Status Pesanan	37
Tabel 3.12 Tabel <i>User</i>	45
Tabel 3.13 Tabel Percetakan	45
Tabel 3.14 Tabel Pesanan Dokumen	46
Tabel 3.15 Tabel Pesanan Foto	46
Tabel 3.16 Tabel Pesan.....	47
Tabel 3.17 Tabel Pembelian Saldo	47
Tabel 3.18 Tabel Saldo.....	48
Tabel 3.19 Tabel Syarat dan Ketentuan	48
Tabel 3.20 Tabel Deskripsi Harga.....	48
Table 3.21 Daftar Menu Aplikasi Percetakan <i>Online</i> Berbasis Web	53

...Halaman ini sengaja dikosongkan...

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi dan informasi di era globalisasi saat ini semakin maju dan berkembang. Peranan teknologi informasi memiliki pengaruh yang sangat besar pada kehidupan manusia. Hal ini terjadi karena dengan semakin berkembangnya teknologi dapat membuat apa saja menjadi lebih mudah, cepat dan efisien. Sejalan dengan pesatnya perkembangan teknologi, Penggunaan teknologi berbasis *internet* terus meningkat dari tahun ke tahun hingga dewasa ini *internet* menjadi suatu kebutuhan sehari - hari. Sebagian besar masyarakat menggunakan internet untuk memudahkan penyelesaian urusan mereka. Beragam informasi, layanan, dan ilmu pengetahuan semuanya tersedia dan dapat diakses dengan mudah oleh semua pengguna *internet*. Dengan semakin pesatnya perkembangan teknologi, maka perlu adanya penyesuaian antara pola kerja dengan perkembangan teknologi. Berbagai kemudahan, kenyamanan dan efisiensi waktu menjadi tuntutan yang harus diperhatikan bagi para pendiri usaha, salah satunya adalah usaha percetakan.

Percetakan merupakan salah satu usaha yang tidak pernah surut bahkan semakin berkembang pesat pada setiap tahunnya, baik dalam skala besar, menengah, maupun skala kecil. Percetakan merupakan sebuah usaha yang didirikan untuk memenuhi kebutuhan pelanggan yang ingin mencetak sebuah dokumen, foto, dan sebagainya. Seiring dengan perkembangan usaha percetakan yang sangat pesat, persaingan bisnis percetakan juga semakin ketat sehingga pelayanan pada pelanggan sangat penting diperhatikan dalam meningkatkan daya saing usaha.

Pada umumnya pelanggan yang ingin mencetak *file* diharuskan datang langsung ke tempat percetakan untuk mengirimkan *file* yang akan dicetak menggunakan *flashdisk* sekaligus memastikan apakah percetakan yang dituju sedang beroperasi ataukah sedang tutup. Tidak jarang pula mereka harus mengantri ketika sampai di tempat percetakan bahkan harus mencari percetakan lain karena terlalu banyak antrian, hal ini tentunya banyak membuang waktu dan kurang efisien.

Berdasarkan uraian yang telah dijelaskan, diusulkan sebuah aplikasi yang berjudul CETAKIN : Aplikasi Percetakan *Online* Berbasis Web. Dengan adanya aplikasi cetakin diharapkan dapat memudahkan pelanggan untuk memilih tempat percetakan. Pelanggan dapat mengirimkan *file* yang akan dicetak tanpa harus datang ke tempat percetakan sehingga lebih menghemat waktu karena pelanggan tidak harus mengantri dan menunggu di tempat percetakan sampai pesanan selesai dicetak sehingga dapat digunakan untuk keperluan lain.

1.2 Perumusan Masalah

Adapun permasalahan yang dapat dirumuskan dari latar belakang adalah sebagai berikut:

1. Bagaimana mengelola pesanan pelanggan dengan menggunakan aplikasi cetakin.
2. Bagaimana mengelola saldo dengan menggunakan aplikasi cetakin.

1.3 Tujuan

Dalam Tugas Akhir terdapat beberapa tujuan diantaranya adalah sebagai berikut:

1. Petugas percetakan dapat memproses atau menolak pesanan menggunakan aplikasi cetakin.
2. Admin aplikasi dapat menambahkan saldo pelanggan dan petugas percetakan dapat memantau saldo dari pelanggan yang akan mencetak file.

1.4 Manfaat

1. Bagi Pelanggan

Pelanggan dapat memilih tempat percetakan yang diinginkan dan tidak harus datang ke tempat percetakan untuk mengirimkan *file* melalui *flasdisk*. Pelanggan dapat langsung mengirimkan *file* yang akan dicetak melalui aplikasi dan mengambil *file* yang telah dicetak ketika sudah menerima konfirmasi bahwa pesanan telah selesai.

2. Bagi Percetakan

Petugas percetakan dapat langsung mengunduh *file* pesanan pelanggan untuk diproses sesuai deskripsi dari pesanan.

1.5 Batasan Masalah

Dalam Tugas Akhir terdapat beberapa tujuan diantaranya adalah sebagai berikut:

1. Aplikasi yang dirancang merupakan aplikasi berbasis *web* yang dibuat menggunakan bahasa pemrograman PHP dengan *framework CodeIgniter* dan menggunakan basis data MySQL untuk penyimpanan data.
2. Studi kasus yang digunakan adalah percetakan di kota Banyuwangi.
3. Layanan yang diberikan adalah cetak dokumen dan cetak foto.
4. Proses pengisian, pengecekan, dan pengurangan saldo bersifat manual.
5. *File* yang dikirim dalam cetak dokumen diutamakan menggunakan ekstensi pdf.

...Halaman ini sengaja dikosongkan...

BAB 2

TINJAUAN PUSTAKA

2.1 Percetakan

Percetakan adalah sebuah proses industri untuk memproduksi salinan tulisan dan gambar dengan media kertas secara massal. Mesin cetak yang digunakan adalah *printer* dan mesin *offset*. Percetakan merupakan salah satu penemuan yang memiliki peran penting di dalam sejarah peradaban manusia. Bahkan setelah ditemukannya penemuan-penemuan seperti radio, televisi, dan gambar bergerak, hasil dari percetakan tetap menjadi sumber informasi utama bagi dunia. Pada masa sekarang ini, percetakan merupakan industri penting di setiap negara maju di dunia. Karena hasil percetakan dapat dengan cepat mengomunikasikan berbagai macam pemikiran dan informasi ke setiap orang. (smart2printpadang.com).

Percetakan memiliki tanggung jawab langsung terhadap konsumen, yaitu kepuasan terhadap hasil kerja yang sesuai dengan keinginan konsumen. Karena kemajuan sebuah perusahaan tergantung pada kepuasan para konsumen.

2.2 Hypertext Markup Language (HTML)

HTML adalah sebuah bahasa pemrograman untuk menampilkan konten pada halaman web. HTML merupakan bahasa pemrograman yang bebas, artinya tidak dimiliki oleh siapapun, pengembangannya dilakukan oleh banyak orang di banyak negara dan bisa dikatakan sebagai sebuah bahasa yang dikembangkan bersama-sama secara global. (Edy Winarno, Ali Zaki, 2014).

Kelebihan dari HTML adalah:

1. HTML bersifat fleksibel yang dapat digunakan pada berbagai jenis mesin komputer dan berbagai macam sistem operasi yang berbeda.
2. Dapat disisipkan berbagai macam gambar, baik gambar statis atau dinamis termasuk menggunakan gambar untuk dijadikan *hyperlink*. Gambar di sini digunakan untuk merujuk pada suatu halaman web, dimana setiap titik-titik yang sudah didefinisikan.

3. Dapat ditambahkan animasi berupa *Java Applet* atau *file* animasi dari *Macromedia Flash* maupun *Macromedia Shockwave*. Namun untuk menjalankan *file-file* animasi browser harus memiliki *plug-in* khusus.
4. Dapat disisipi bahasa pemrograman seperti *JavaScript*, *VBScript*, *Active Server Pages*, *Perl*, *Tcl*, *Hypertext preprocessor* dan sebagainya untuk mempercantik halaman web.

Kekurangan dari HTML adalah:

1. Menghasilkan halaman yang statis, untuk menghasilkan halaman yang dinamis harus menggunakan bahasa pemrograman tertentu seperti *JavaScript* atau *VBScript* dan animasi seperti *Flash* atau *Shockwave*.
2. Mempunyai banyak *tag* sehingga sulit dipelajari untuk orang yang baru mengenal bahasa pemrograman.
3. Tidak dapat menghasilkan halaman yang interaktif antara klien dengan *server* apabila tidak disisipkan bahasa pemrograman yang dapat menangani hal tersebut, contohnya *Perl* atau *Tool Command Language*.

2.3 Cascading Style Sheets (CSS)

Bekti (2015), CSS merupakan salah satu bahasa pemrograman web yang digunakan untuk mempercantik halaman web. CSS berfungsi mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam, dan teratur.

CSS memungkinkan untuk menampilkan halaman yang sama dengan format yang berbeda. CSS dapat mengendalikan ukuran gambar, warna tabel, jenis *border*, warna *hyperlink*, warna *mouse hover*, *margin* kiri, kanan, atas, bawah, dan parameter lainnya.

Kelebihan penggunaan CSS pada web yaitu:

1. Mempercantik tampilan dari sebuah web yang lebih terstruktur dan teratur.
2. Memisahkan antara desain dengan konten pada halaman web.
3. Memungkinkan untuk mengatur desain seefisien mungkin.
4. Satu CSS dapat digunakan pada banyak halaman web.
5. Mengurangi beban *bandwidth* sehingga proses saat mengakses halaman web lebih cepat.

Kekurangan penggunaan CSS pada web yaitu:

1. Tidak semua *browser* mendukung kode CSS.
2. Tampilan pada setiap *browser* dapat berbeda-beda.
3. Dibutuhkan waktu yang lama untuk membuatnya

2.4 Pemrograman Web

Pemrograman web terdiri dari dua kata yaitu pemrograman yang artinya proses atau pembuatan, dan web yang artinya jaringan komputer yang terdiri dari kumpulan situs *internet* yang menampilkan informasi seperti teks, gambar, suara, dan animasi. Halaman web merupakan *file* teks murni yang berisi sintaks-sintaks *Hypertext Markup Language* yang dapat dibuka, dilihat, dan diterjemahkan dengan *internet browser*. (programakita.com).

Dari uraian yang telah dipaparkan dapat ditarik kesimpulan bahwa pemrograman web merupakan proses menulis kode atau membangun sebuah situs web yang berasal dari bahasa pemrograman. Hasil dari pemrograman web mampu memberikan informasi halaman *browser* dengan akses koneksi melalui *internet*.

2.4.1 Hypertext Preprocessor (PHP)

PHP merupakan singkatan dari *Hypertext Preprocessor* yaitu bahasa pemrograman yang berbasis web. Bahasa pemrograman PHP bersifat terbuka yang artinya dapat digunakan oleh siapapun secara gratis. PHP dapat membuat web dinamis serta dapat dijalankan oleh berbagai sistem operasi karena PHP berjalan secara *web base* yang artinya semua sistem operasi.

Winarno dan Ali Zaki (2014), PHP adalah sebuah bahasa pemrograman web berbasis *server* yang mampu memarsing kode PHP dari kode web dengan ekstensi *.php*. Sehingga menghasilkan tampilan *website* yang dinamis pada sisi pengguna. PHP adalah bahasa *script* yang sangat cocok untuk pengembangan web dan dapat dimasukkan ke dalam *Hypertext Markup Language* (HTML).

Kelebihan PHP adalah:

1. Dapat membuat web menjadi dinamis.
2. Bersifat terbuka sehingga dapat digunakan oleh siapapun dengan gratis.
3. Dapat dijalankan pada semua sistem operasi.
4. Aplikasi PHP lebih cepat dibandingkan dengan *Active Server Pages* (ASP) maupun *Java*.
5. Terdapat banyak basis data yang mendukung PHP seperti *MySQL*, *Oracle*, *PostgreSQL*, dan lain-lain.
6. Tidak memerlukan kompilasi dalam penggunaannya.
7. Terdapat banyak *Web Server* yang mendukung PHP seperti *Apache*, *Lighttpd*, *Internet Information Service* (IIS) dan lain-lain.
8. Terdapat banyak dokumentasi, *refrensi* dan *developer* yang membantu dalam pengembangannya sehingga aplikasi PHP mudah dikembangkan.
9. Terdapat banyak aplikasi dan program PHP yang gratis serta siap digunakan seperti *WordPress*, *PrestaShop*, dan lain-lain.

Kekurangan PHP adalah:

1. PHP Tidak mengenal *package*.
2. Kode PHP dapat dibaca semua orang apabila tidak dilakukan *encoding*. Untuk melakukan *encoding* dibutuhkan *tool* dari *Zend* dengan biaya yang sangat mahal.
3. PHP memiliki kelemahan dari segi keamanan. Sehingga *programmer* harus lebih teliti dan berhati-hati dalam melakukan pemrograman dan konfigurasi pada PHP.

2.4.2 JavaScript

JavaScript adalah sekumpulan perintah khusus yang digunakan untuk membuat sebuah halaman web yang lebih responsif dan interaktif. *JavaScript* merupakan bahasa *script* yang dicantumkan pada sebuah halaman web dan dijalankan pada penjelajah web.

JavaScript adalah bahasa *scripting client side* yang sangat populer di dunia karena *JavaScript* dapat digunakan pada HTML, web, untuk *server*, PC, laptop, tablet, dan lainnya. Hampir semua *programmer* web menggunakan

JavaScript untuk memberi efek pemrograman di halaman. *JavaScript* tidak hanya berdiri sendiri, tapi juga menjadi dasar yang bisa digunakan untuk teknologi lainnya, seperti *Asynchronous JavaScript and XML* (Ajax), *jQuery* dan *jQuery Mobile*. (Edy dkk, 2014).

Sebagai sebuah bahasa pemrograman yang banyak digunakan dalam mengembangkan sebuah aplikasi dan juga *website*, tentunya *JavaScript* memiliki beberapa kelebihan dan juga keunggulan. Berikut ini adalah beberapa kelebihan dan juga keunggulan dari bahasa pemrograman *JavaScript* diantaranya adalah:

1. Sumber penyimpanan yang relative kecil dan ringan

Kelebihan dan keunggulan pertama dari bahasa pemrograman *JavaScript* adalah dari sumber penyimpanannya. *JavaScript* membutuhkan sumber penyimpanan yang relative kecil dan ringan. Sehingga sangat membantu para developer dalam mengembangkan aplikasi – aplikasi ringan, namun tetap memiliki fitur dan kegunaan yang sangat banyak bagi user.

2. Mudah untuk dipelajari

JavaScript merupakan sebuah bahasa pemrograman banyak digunakan dan dikembangkan karena *JavaScript* merupakan bahasa pemrograman yang simpel dan mudah untuk dipahami.

3. Dinamis

JavaScript adalah salah satu bahasa pemrograman yang dinamis diantara banyak bahasa pemrograman lainnya. selain itu, *JavaScript* juga mudah untuk diaplikasikan, dan mudah untuk dilakukan penambahan *fitur* yang berhubungan dengan fungsi – fungsi penting dari aplikasi.

4. Dapat dijalankan pada banyak sistem operasi

Dapat dijalankan pada banyak sistem operasi atau sering disebut dengan *Multi platform* berarti sebuah aplikasi yang menggunakan bahasa pemrograman *JavaScript* bisa dibuat dengan basis sistem operasi apapun.

5. Mudah untuk dikembangkan

Ketika suatu aplikasi sudah selesai dibuat dengan menggunakan basis bahasa pemrograman *JavaScript*, maka aplikasi tersebut bisa dengan mudah dikembangkan. Hal ini juga mengacu pada konsep dinamis yang diusung oleh bahasa pemrograman *JavaScript*. Aplikasi akan lebih mudah untuk diperbaharui dan dikembangkan sesuai dengan kebutuhan pengguna juga perkembangan teknologi yang ada.

Selain memiliki kelebihan dan keunggulan. Bahasa pemrograman *JavaScript* tentunya juga memiliki kekurangan dan kelemahan. Berikut ini adalah beberapa kekurangan dari bahasa pemrograman *JavaScript* diantaranya adalah:

1. Script yang digunakan pada *JavaScript* tidak terenkripsi

Sejalan dengan bahasa pemrograman yang simple dan mudah dikembangkan, Aplikasi yang dikembangkan dengan menggunakan bahasa pemrograman berbasis *JavaScript* memiliki enkripsi yang lemah, dan cenderung tidak terenkripsi.

2. Bukan untuk pengembangan aplikasi yang mandiri atau *stand alone*

Dengan tingkat keamanan yang sangat rendah, membuat bahasa pemrograman *JavaScript* hanya bisa dioptimalkan pada sebuah aplikasi yang tergantung pada aplikasi lain agar dapat berjalan dengan optimal.

3. Memiliki keterbatasan objek

Bahasa pemrograman *JavaScript* yang sangat simple dan juga sederhana, sehingga mempengaruhi fitur – fitur yang dimiliki oleh bahasa pemrograman *JavaScript* sendiri.

2.5 Framework

Framework adalah sebuah kerangka kerja yang memudahkan para pengembang perangkat lunak dalam membuat dan mengembangkan aplikasi. *Framework* berisikan perintah dan fungsi umum yang digunakan untuk membangun sebuah perangkat lunak aplikasi. Sejalan dengan penggunaan *framework*. Diharapkan aplikasi dapat dibangun dengan lebih cepat serta tersusun dan terstruktur dengan cukup rapi. Framework juga dapat diartikan sebagai komponen - komponen pemrograman yang sudah jadi dan siap untuk digunakan. Sehingga pengembang aplikasi diharuskan membuat script yang sama untuk fungsi yang sama. (utopiccomputers.com).

Dalam penerapannya fungsi Framework dapat diuraikan sebagai berikut:

1. Mempercepat proses dalam pembuatan aplikasi baik aplikasi berbasis *desktop*, *mobile* ataupun web.
2. Membantu para pengembang dalam perencanaan, pembuatan dan pemeliharaan sebuah aplikasi.
3. Menghasilkan aplikasi yang lebih stabil dan handal, karena *framework* sudah melalui proses uji baik itu stabilitas dan juga kehandalannya.
4. Memudahkan para pengembang dalam membaca kode program.
5. Memiliki tingkat keamanan yang lebih, karena *framework* telah mengantisipasi cela - cela keamanan yang mungkin timbul.

2.5.1 CodeIgniter

Menurut Basuki (2014), *CodeIgniter* merupakan suatu *framework* PHP yang membantu pengembangan aplikasi web berbasis PHP lebih cepat dan rapi dibandingkan dengan menulis semua kode program dari awal. *CodeIgniter* menyediakan banyak *library* untuk mengerjakan tugas – tugas yang umumnya ada pada sebuah aplikasi berbasis web. Struktur dan susunan logis dari *CodeIgniter* membuat aplikasi menjadi semakin teratur dan rapi. Hal ini dikarenakan *CodeIgniter* dibangun menggunakan metode *Model, View, Controller* (MVC).

MVC adalah konsep untuk memisahkan komponen utama menjadi 3 komponen yaitu *model*, *view*, dan *controller*.

1. *Model*

Model merupakan komponen yang berhubungan dengan pengolahan data atau manipulasi pada basis data. Perintah-perintah yang berhubungan dengan basis data disimpan dalam *model*.

2. *View*

View merupakan komponen yang berhubungan dengan tampilan pengguna yang diatur oleh *controller*. *View* berfungsi untuk menerima dan mempresentasikan data kepada pengguna.

3. *Controller*

Controller merupakan komponen yang mengatur hubungan antara *model* dengan *view*. *Controller* berfungsi untuk menerima permintaan data dari pengguna kemudian menentukan proses yang dijalankan oleh aplikasi.

Adapun kelebihan dan kekurangan dari *CodeIgniter* yaitu sebagai berikut:

1. Kelebihan *CodeIgniter*:

- a. *CodeIgniter* menggunakan model MVC sehingga lebih tepat digunakan dalam pembuatan dan pengembangan *website*.
- b. Dilengkapi dengan dokumentasi yang mendukung.
- c. Memiliki kinerja yang baik dan cepat.
- d. Kompatibilitas dengan *hosting*. *CodeIgniter* mampu berjalan dengan baik pada hampir semua *platform hosting*.
- e. Tersedia banyak *helper* dan *library*.
- f. Mendukung PHP4, PHP5, dan AJAX.

2. Kekurangan *CodeIgniter*:

- a. Tidak mendukung *Object Relational Mapper* (ORM)
- b. Banyak kelonggaran dalam *coding*, penamaan *file* dan membebaskan *programmer* untuk melanggar aturan MVC.

2.5.2 *Laravel*

Laravel adalah sebuah *framework* PHP yang dirilis dibawah lisensi MIT. *Laravel* merupakan pengembangan website berbasis MVC yang ditulis dalam PHP yang dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi biaya pengembangan awal dan biaya pemeliharaan serta untuk meningkatkan pengalaman bekerja dengan aplikasi dengan menyediakan sintaks yang ekspresif, jelas dan menghemat waktu. (Aminudin, 2015).

MVC adalah sebuah pendekatan perangkat lunak yang memisahkan aplikasi logika dari presentasi. MVC memisahkan aplikasi berdasarkan komponen-komponen aplikasi, seperti: manipulasi data, *controller*, dan *user interface*.

Fitur – fitur yang terdapat pada *laravel* adalah:

1. *Bundles*, yaitu sebuah fitur dengan sistem pengemasan *modular* dan tersedia beragam di aplikasi.
2. *Eloquent ORM*, *laravel* menyediakan metode *internal* dari pola “*active record*” yang mengatasi masalah hubungan objek pada basis data.
3. *Application logic*, menggunakan *controller* atau bagian *route* dan merupakan bagian dari aplikasi.
4. *Reverse routing*, mendefinisikan hubungan antara *link* dan *route*.
5. *Restful controllers*, memisahkan logika *HTTP GET* and *POST*.
6. *Class auto loading*, menyediakan *loading* otomatis untuk class PHP.
7. *View composer*, merupakan kode unit *logikal* yang dapat dieksekusi pada saat *view* sedang *loading*.
8. *IoC container*, memungkinkan menghasilkan obyek baru dengan pembalikan *controller*.
9. *Migration*, menyediakan sistem kontrol untuk basis data.
10. *Unit testing*, terdapat banyak tes untuk mendeteksi dan mencegah *regresi*.
11. *Automatic pagination*, menyederhanakan tugas dari penerapan halaman yang dibuat secara otomatis.

Kelemahan *laravel* adalah:

1. Tingkat kesulitan lebih tinggi.
2. *Instalasi* di server lebih sulit.
3. Banyak *file* dan *asset* yang dijalankan sehingga membuat *laravel* relative lebih berat.

2.5.3 Yes It Is (YII)

Menurut Badiyanto (2013), YII adalah *framework* PHP berbasis komponen untuk pengembangan aplikasi web berskala besar. *Framework* YII menyediakan *reusability maksimum* dalam pemrograman web dan mampu meningkatkan kecepatan dalam membuat aplikasi web.

YII mengimplementasikan pola desain MVC. MVC bertujuan untuk memisahkan logika dari pertimbangan antarmuka pengguna agar para pengembang bisa lebih mudah mengubah setiap bagian tanpa mempengaruhi yang lain.

Kelebihan YII *framework*:

1. Ringan untuk diaplikasikan.
2. Dokumentasi yang jelas.
3. Dilengkapi dengan *code generator*.

Kekurangan YII *framework*:

1. Tidak mendukung PHP versi 4

2.6 Basis data

Menurut Indrajani (2015), basis data adalah kumpulan data yang saling berhubungan secara logis dan didesain untuk mendapatkan data yang dibutuhkan oleh suatu organisasi.

Menurut Rosa dan M. Shalahudin (2014), Basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat.

2.6.1 *My Structured Query Language (MySQL)*

MySQL adalah salah satu aplikasi *Database Management System (DBMS)* yang banyak digunakan oleh para pemrogram aplikasi *web*. MySQL merupakan tipe data relasional yang artinya MySQL menyimpan datanya dalam bentuk tabel-tabel yang saling berhubungan (Edy Winarno, Ali Zaki, 2014).

MySQL merupakan sebuah *database server* yang dapat diperoleh atau digunakan secara gratis. Sehingga bebas menggunakan basis data ini untuk keperluan pribadi atau usaha tanpa harus membeli atau membayar lisensinya. Basis data MySQL merupakan suatu perangkat lunak yang menggunakan suatu bahasa yaitu *Structure Query Language (SQL)*.

Berikut kelebihan MySQL yang menyebabkannya sangat populer diantaranya:

1. Bersifat *open source* dibawah lisensi *General Public License (GPL)* sehingga mudah untuk didapatkan.
2. Memiliki tipe data yang kompleks.
3. Dapat digunakan oleh beberapa *user* dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
4. Tidak membutuhkan spesifikasi *hardware* yang tinggi.
5. Dapat diintegrasikan dengan berbagai bahasa pemrograman.
6. Memiliki kecepatan yang tinggi dalam menangani *query* sederhana.

Sedangkan kelemahan MySQL adalah:

1. Sulit untuk diaplikasikan pada instansi atau perusahaan dengan database yang besar.
2. *Technical support* dari MySQL kurang baik.
3. Kurang mendukung untuk aplikasi *mobile* dan *game*.

2.6.2 *Postgre SQL*

PostgreSQL merupakan sistem pengelolaan basis data secara relasional. fungsi utama dari *PostgreSQL* adalah menyimpan data dengan aman dan mengembalikan data tersebut sebagai respon atas permintaan dari aplikasi perangkat lunak lainnya. Pada macOS Server, PostgreSQL adalah basis data default, PostgreSQL juga tersedia untuk Microsoft Windows dan Linux. (DosenIT.com).

Kelebihan *PostgreSQL* adalah:

1. Terdapat fitur *OO*: Berfungsi untuk mewarisi tabel, dan tipe data (tipe data array). Sehingga menjadikan *PostgreSQL* lebih praktis dalam penyimpanan *item* data dalam jumlah banyak dalam satu *recordnya*.
2. Memiliki arsitektur *multiproses*: Arsitektur *multiproses* (*forking*) yang dimiliki oleh *PostgreSQL* menjadikan PostgreSQL mempunyai stabilitas yang tinggi.
3. Memiliki kecepatan meski dalam *load* tinggi: *PostgreSQL* mendukung *locking level* di bagian yang lebih rendah, yakni pada *locking level row* sehingga *PostgreSQL* memiliki kecepatan yang tinggi.
4. Menyediakan seluruh fitur basis data: Fitur database yang disediakan oleh *PostgreSQL* hampir seluruhnya mencakup fitur-fitur database yang sering digunakan pada produk basis data komersial lain pada umumnya.
5. Memiliki semua fasilitas standar: Tersedia berbagai macam fasilitas standar yang sering digunakan seperti *view*, *sub select*, *foreign key constraint*, *stored procedure* dan *trigger* yang dapat meningkatkan kemampuan *PostgreSQL* dalam menjalankan tugasnya.
6. Dapat mendefinisikan *field* sebagai *array*: *PostgreSQL* dapat menerjemahkan atau mendefinisikan *field* yang biasanya belum dapat terdefinisi otomatis sebagai *array*.
7. Memiliki lisensi *General Public License (GPL)*: *PostgreSQL* dapat digunakan secara bebas oleh setiap penggunaanya tanpa harus melalui proses pemesanan, dan pembayaran.

Selain memiliki keunggulan yang beragam, *PostgreSQL* juga memiliki sisi kelemahan yang perlu diperhatikan jika ingin menggunakan database sistem tersebut. Kekurangan *PostgreSQL* adalah:

1. Fungsi *PostgreSQL* terbatas: *PostgreSQL* hanya dapat melakukan beberapa fungsi yaitu penambahan kolom, penggantian nama kolom, dan juga penggantian pada nama tabel.
2. Kurang populer: Penggunaan *PostgreSQL* cenderung kurang populer dibandingkan dengan basis data lainnya, selain itu *PostgreSQL* juga kurang cocok untuk pekerjaan di lingkungan web jika dibandingkan dengan database MySQL.
3. *Multiproses* dan arsitekturnya sulit diterapkan pada windows: meski sudah dapat diterapkan pada windows dengan menggunakan lapisan *emulasi Cygwin*, namun pada kenyataannya *windows* sangat *thread-oriented*. Sehingga arsitektur dengan *multiproses* sulit untuk diterapkan tanpa menggunakan *emulator*.
4. Belum tersedianya replikasi: *PostgreSQL* merupakan jenis basis data yang *open source*. Namun di dalam distribusi standarnya *PostgreSQL* masih belum menyertakan replikasi sehingga perlu kerja keras dari pengguna untuk mendistribusikan *PostgreSQL*.

2.6.3 Oracle

Oracle merupakan sistem manajemen basis data *relasional* yang diproduksi dan dipasarkan oleh *Oracle Corporation*. Penggunaan basis data *Oracle* mengacu pada struktur memori *server-side* sebagai *System Global Area (SGA)*. SGA menyimpan informasi *cache* seperti *buffer* data, perintah SQL, dan informasi pengguna. (DosenIT.com).

Kelebihan basis data *Oracle* adalah:

1. Kemampuan untuk melakukan manajemen sistem basis data yang baik.
2. Menangani jumlah data dan juga angka yang sangat besar.
3. Data dapat dikelola dengan cepat dan akurat.
4. Memiliki kemampuan untuk melakukan *cluster server*.
5. Dapat melakukan management pengguna.
6. Pemrosesan data yang cepat.

Kekurangan basis data *Oracle* adalah:

1. Harga *software* DBMS tinggi dan mahal.
2. Spesifikasi *minimum hardware* yang tinggi.
3. Tidak cocok untuk diaplikasikan pada perusahaan atau instansi kecil.
4. Rumit dalam mengaplikasikannya.

2.7 Metode Pengembangan Sistem

2.7.1 Metode Waterfall

Menurut Pressman (2015), Metode *waterfall* sering dikenal sebagai sekuensial linier (*sequential linier*) atau siklus hidup klasik (*classic life cycle*). Metode *waterfall* merupakan pendekatan alur hidup perangkat lunak secara sekuensial atau berurut dimulai dari tahap analisis kebutuhan, desain sistem, pengerjaan, pengujian, dan penerapan. Pengerjaan suatu sistem yang menggunakan metode *waterfall* dilakukan secara terstruktur dan berkelanjutan. Dimana Setiap tahapan harus diselesaikan untuk melanjutkan ke tahapan selanjutnya, jadi apabila langkah ke-1 belum dikerjakan, maka tidak bisa dilanjutkan ke langkah ke-2, begitu juga untuk langkah selanjutnya. Selain itu, metode *waterfall* juga memungkinkan untuk kembali ke tahapan sebelumnya jika terjadi kesalahan.

Kelebihan penggunaan metode *waterfall* adalah:

1. Memiliki proses yang urut.
2. Sebuah sistem dapat dikembangkan sesuai dengan apa yang dikehendaki karena setiap proses memiliki spesifikasinya sendiri.
3. Setiap proses tidak dapat saling tumpang tindih.

Kekurangan penggunaan metode *waterfall* adalah:

1. Proses yang dilakukan cenderung lama dan panjang.
2. Biaya penggunaan metode cenderung mahal.
3. Membutuhkan banyak *riset* dan juga penelitian pendukung untuk mengembangkan sistem menggunakan metode *waterfall*.

2.7.2 Metode *Rapid Application Development* (RAD)

Menurut Wahyuningrum (2014), *Rapid Application Development* (RAD) adalah sebuah proses perkembangan perangkat lunak sekuensial linier yang menekankan siklus perkembangan dalam waktu yang singkat. Metode RAD dapat menghasilkan suatu sistem dengan cepat karena sistem yang dikembangkan dapat memenuhi keinginan dari para pemakai. Sehingga setelah tahap implementasi, dapat mengurangi waktu untuk pengembangan ulang. Metode RAD memiliki 3 tahapan sebagai berikut.

1. Rencana Kebutuhan (*Requirement Planning*)

User dan *analyst* melakukan pertemuan untuk mengidentifikasi tujuan dari sistem dan kebutuhan informasi untuk mencapai tujuan. Pada tahap ini merupakan hal terpenting yaitu adanya keterlibatan dari kedua belah pihak.

2. Proses Desain Sistem (*Design System*)

Pada tahap ini keaktifan *user* yang terlibat menentukan untuk mencapai tujuan karena pada proses ini melakukan proses desain dan melakukan perbaikan-perbaikan apabila masih terdapat ketidaksesuaian desain antara *user* dan *analyst*. Seorang *user* dapat langsung memberikan komentar apabila terdapat ketidaksesuaian pada desain, merancang sistem dengan mengacu pada dokumentasi kebutuhan *user* yang dibuat pada tahap sebelumnya. Keluaran dari tahapan ini adalah spesifikasi software yang meliputi organisasi sistem secara umum, struktur data dan yang lain.

3. Implementasi (*Implementation*)

Tahapan ini adalah tahapan *programmer* yang mengembangkan desain suatu program yang telah disetujui oleh *user* dan *analyst*. Sebelum diaplikasikan pada suatu organisasi terlebih dahulu dilakukan proses pengujian terhadap program tersebut apakah ada kesalahan atau tidak. Pada tahap ini *user* biasa memberikan tanggapan akan sistem yang sudah dibuat serta mendapat persetujuan mengenai sistem tersebut.

Menurut Wahyuningrum (2014), metode RAD memiliki kelebihan sebagai berikut:

1. Setiap *fungsi* mayor dapat dimodulkan dalam waktu tertentu kurang dari 3 bulan dan dapat dibicarakan oleh tim RAD yang terpisah dan kemudian diintegrasikan sehingga waktunya lebih efisien.
2. RAD mengikuti tahapan pengembangan sistem seperti umumnya, tetapi mempunyai kemampuan untuk menggunakan kembali komponen yang ada (*reusable object*) sehingga pengembang tidak perlu membuat dari awal lagi dan waktu lebih singkat .

Sedangkan kelemahan metode RAD adalah:

1. Proyek yang besar dan berskala, RAD memerlukan sumber daya manusia yang memadai untuk menciptakan jumlah tim yang baik.

2.7.3 Metode *Prototype*

Menurut Djahir (2014), *Prototype* merupakan hal yang memberikan sebuah ide bagi pembuat dan pemakai potensi tentang cara sistem akan berfungsi dalam bentuk lengkap. Dengan proses menghasilkan suatu *prototype* disebut dengan *prototype*.

Pressman (2015), Berikut adalah tahapan dalam metode *prototype*:

1. Komunikasi, merupakan tahap awal yaitu analisis terhadap kebutuhan pengguna atau pemakai.
2. Desain cepat, merupakan tahap membuat desain secara umum untuk selanjutnya dikembangkan kembali.
3. Pembentukan *prototype*, merupakan tahap pembuatan perangkat *prototype* yang meliputi pengujian dan penyempurnaan.
4. Evaluasi terhadap *prototype*, merupakan tahap mengevaluasi dan memperhalus analisis terhadap kebutuhan pengguna atau pemakai.
5. Perbaikan *prototype*, merupakan tahap pembuatan tipe yang sebenarnya yang berdasarkan dari evaluasi *prototype*.
6. Produksi akhir, merupakan tahap memproduksi perangkat secara benar dan sehingga dapat digunakan oleh pengguna atau pemakai.

Kelebihan Metode *Prototype* adalah:

1. Adanya komunikasi yang baik antara pengembang dan pelanggan tau pemakai.
2. Pelanggan berperan aktif dalam pengembangan sistem.
3. Pengembang dapat bekerja lebih baik dalam menentukan kebutuhan pelanggan atau pemakai.
4. Lebih menghemat waktu dalam pengembangan sistem.
5. Penerapan menjadi lebih mudah karena pemakai mengetahui apa yang diharapkannya.

Kekurangan Metode *Prototype* adalah:

1. Interaksi pemakai penting, dengan sistem harus menyediakan percakapan *online* antara pelanggan dan komputer.
2. Hubungan pelanggan dan komputer yang disediakan mungkin tidak mencerminkan teknik perancangan yang baik.
3. Resiko tinggi adalah untuk masalah – masalah yang tidak terstruktur dengan baik, ada perubahan yang besar dari waktu ke waktu, dan adanya persyaratan data yang tidak menentu.

2.8 Konsep Pemodelan Sistem

Pemodelan sistem merupakan proses menggambarkan konsep yang mewakili obyek-obyek dalam pengembangan sistem informasi. Pemodelan sistem berkaitan dengan bagaimana sistem yang diwujudkan dengan menggunakan teknologi. Pemodelan sistem sebagian besar merupakan kegiatan teknologi yang mencoba untuk menerjemahkan model aplikasi ke dalam bangunan sistem operasional. Jacobson dkk, membuat standar pemodelan untuk pengembangan sistem pemodelan yang relevan, dengan pendekatan obyek yang dikenal dengan *Unified Modeling Language* (UML). UML memiliki keuntungan yang menghasilkan representasi yang dapat diverifikasi melalui penalaran logis, pengujian, atau bahkan simulasi. (sis.binus.ac.id).


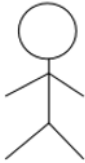

2.8.1 Diagram *Use case*

Use Case Diagram Sukamto dan Shalahuddin (2014), *Use case* atau diagram *use case* merupakan pemodelan untuk kelakuan sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Syarat penamaan pada *use case* adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*.

1. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
2. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

Berikut adalah simbol-simbol yang ada pada diagram *use case* ditunjukkan pada tabel 2.1.

Tabel 2.1 Diagram *Use case*



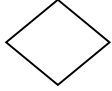


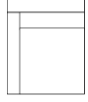
No	Simbol	Deskripsi
1		Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja diawal <i>frase</i> nama <i>use case</i> .
2		Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal <i>frase</i> nama aktor.
3		Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.

4	<p><i>Exstensi</i></p> <p>----- <<extend>> ----></p>	<p>Relasi <i>use case</i> tambahan kesebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu, mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek, biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan, Arah panah mengarah pada <i>use case</i> yang ditambahkan, biasanya <i>use case</i> yang menjadi <i>extend</i>-nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya.</p>
5	<p><i>Generalisasi</i></p> <p>—————→</p>	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p> <p>arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya (umum)</p>
6	<p>Menggunakan / <i>Include</i> / <i>Uses</i></p> <p><<include>> —————→</p> <p><<uses>> —————→▷</p>	<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini Ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i> :</p> <ul style="list-style-type: none"> - <i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu di panggil saat <i>use case</i> tambahan dijalankan. - <i>Include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang di tambahkan telah dijalankan sebelum <i>use case</i> tambahan di jalankan. <p>Kedua interpretasi di atas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan.</p>

2.8.2 Diagram Activity

Sukamto dan Shalahuddin (2014), Diagram aktivitas atau *activity diagram* menggambarkan aliran kerja atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Berikut adalah simbol-simbol yang ada pada diagram aktivitas yang perlu di perhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Berikut adalah simbol-simbol yang ada pada diagram aktivitas ditunjukkan pada table 2.2.

Tabel 2.2 *Diagram Activity*

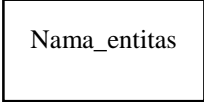
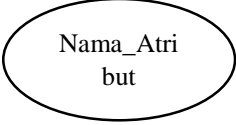
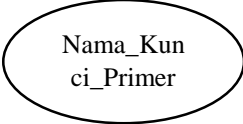



No.	Gambar	Nama	Keterangan
1.		Status Awal	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2.		Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3.		Percabangan / <i>decision</i>	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4.		Penggabungan / <i>join</i>	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5.		Status Akhir	Status akhir yang dilakukan oleh sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6.		Swimlane	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

2.8.3 Entity Relationship Diagram (ERD)

Rosa A.S dan M. Shalahuddin (2014), *Entity Relationship Diagram* adalah bentuk paling awal dalam melakukan perancangan basis data relasional. *Entity Relationship Diagram* memiliki beberapa aliran notasi seperti notasi *Chen* (dikembangkan oleh Peter Chen), *Baker* (dikembangkan oleh Ricard Barker), *Ian* (dikembangkan oleh Palmer, Harry Ellis), notasi *Crow's Foot*, dan beberapa notasi

lainnya. Berikut adalah simbol-simbol yang digunakan pada *Entity Relationship Diagram* dengan notasi *Chen*.

Tabel 2.3 *Entity Relationship Diagram*

No	Simbol	Deskripsi
1	Entitas/ <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat digunakan aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama table.
2	Atribut 	Field atau kolom data yang butuh disimpan dalam suatu entitas.
3	Atribut Kunci Primer 	Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record, biasanya berupa id; kunci primer dapat lebih dari satu kolom tersebut dapat bersifat unik
4	Atribut <i>multivalue</i> 	Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
5	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kerja
6	Asosiasi 	Penghubung antara relasi dan entitas di mana kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian

2.9 Pengujian

Pengujian merupakan proses menjalankan dan mengevaluasi sebuah perangkat lunak secara manual maupun otomatis untuk menguji apakah perangkat lunak sudah memenuhi persyaratan atau belum. Pengujian dapat dilakukan menggunakan dua metode yaitu *black box testing* dan *white box testing*. (Abdul Rouf, 2015).

2.9.1 *Black Box Testing*

Black box testing adalah pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. (Abdul Rouf, 2015).

Beberapa kelebihan metode *black box testing* adalah:

1. Mendeteksi kesalahan desain atau *user interface*.
2. Penguji tidak harus seorang *programmer*.
3. Tidak perlu melihat source code secara detail.

Beberapa kekurangan metode *black box testing* adalah:

1. Ketergantungan dengan dokumen dan desain.
2. *Tester* tidak mengetahui tingkat keamanan.

2.9.2 *White Box Testing*

White box testing adalah pengujian yang didasarkan pada pengecekan terhadap detail perancangan, menggunakan struktur kontrol dari desain program secara procedural untuk membagi pengujian ke dalam beberapa kasus pengujian. (Abdul Rouf, 2015).

Beberapa kelebihan metode *black box testing* adalah:

1. Memudahkan *programmer* dalam mengoptimasi *source code*.
2. Mengecek efisiensi *code*.
3. Memilih fungsi-fungsi yang cocok digunakan dalam *software development*.

Beberapa kelebihan metode *black box testing* adalah:

1. *Tester* harus memiliki ilmu pemrograman.

2.10 Penelitian Terkait

Penelitian yang dilakukan tidak terlepas dari hasil penelitian terdahulu yang sudah dilakukan dan menjadi bahan perbandingan dan kajian bagi penelitian yang akan dilakukan. Penelitian terdahulu yang dipilih dan dijadikan bahan acuan tentunya tidak lepas dari topik penelitian yang akan dibuat yaitu mengenai percetakan online.

Tabel 2.4 Penelitian Terkait

Unsur Perbandingan	Dwi Puspita, dkk	Fauzi Rahman, Santoso	Leonardo MP, Fany Junianti	Adi supriyatna	Hendriyanto
Judul	Sistem informasi pemesanan dan pembayaran pada percetakan mahardika	Aplikasi pemesanan undangan online	Aplikasi Pelayanan Pemesanan Online Pada Digital Printing Ecoprint Palembang	Sistem informasi pemesanan jasa percetakan berbasis web	Aplikasi percetakan online berbasis web
Fitur	<ul style="list-style-type: none"> - Transaksi pemesanan - Transaksi pembayaran - Laporan bulanan 	<ul style="list-style-type: none"> - Transaksi pemesanan - Menu detail barang - Menu keranjang 	<ul style="list-style-type: none"> - Forum diskusi - Kelola pesanan - Komentar - lihat katalog - Status pesnan 	<ul style="list-style-type: none"> - Kelola pesanan - Pilih kategori - Pembuatn laporan - Pilih layanan 	<ul style="list-style-type: none"> - Notifikasi - Saldo - Chat - Kelola pesanan - Pilih layanan - Pilih percetakan - Status pesanan
Teknologi	web	web	web	web	web
Kelebihan	<ul style="list-style-type: none"> - Sistem yang dibuat sebagai pengganti sistem lama yang masih konvensional. - membantu proses transaksi pemesanan 	<ul style="list-style-type: none"> - memudahkan pelanggan untuk memilih undangan yang akan dipesan 	<ul style="list-style-type: none"> - mempermudah pelanggan dalam melakukan pemesanan yang cukup diakses melalui internet. 	<ul style="list-style-type: none"> - Memperlus jangkauan pemasaran produk jasa percetakan. - memberikan kemudahan bagi konsumen untuk dapat 	<ul style="list-style-type: none"> - Memperlus jangkauan pemasaran produk jasa percetakan. - memberikan kemudahan bagi pelanggan untuk dapat

	<ul style="list-style-type: none"> - n dan pembayar an laporan pendapat an dapat diakses dengan mudah karena per transaksi pembayar an secara otomatis akan tersimpan kedalam laporan pendapat an 			<ul style="list-style-type: none"> - mencari informasi terkait produk dan jasa yang ditawarkan memudahkan konsumen dalam melakukan pesanan 	<ul style="list-style-type: none"> - mencari informasi harga maupun informasi percetakan memudahkan konsumen dalam melakukan pesanan dapat diterapkan untuk banyak percetakan pelanggan dapat menentukan halaman yang akan dicetak
Kekurangan	<ul style="list-style-type: none"> - 	<ul style="list-style-type: none"> - Hanya melayani pesanan kelipatan 100 - Tidak terdapat proses pengisian data undangan - Pembayaran masih bersifat manual - Memungkin adanya pesanan palsu yang merugikan perusahaan 	<ul style="list-style-type: none"> - Hanya diterapkan pada satu perusahaan - Pelanggan tidak dapat memilih halaman yang akan dicetak. - Memungkin adanya pesanan palsu yang merugikan perusahaan 	<ul style="list-style-type: none"> - Hanya diterapkan pada satu perusahaan - Pelanggan tidak dapat memilih halaman yang akan dicetak. - Memungkin adanya pesanan palsu yang merugikan perusahaan - Pembayaran dilakukan dengan cara transfer 	<ul style="list-style-type: none"> - Pengisian dan pengurangan saldo bersifat manual - Perhitungan biaya mencetak bersifat manual dilakukan oleh petugas percetakan

BAB 3

METODE PENELITIAN

3.1 Waktu, Tempat, dan Jadwal Penelitian

3.1.1 Waktu Penelitian

Waktu penelitian Proyek Akhir akan dilaksanakan selama kurang lebih enam bulan, dimulai dari bulan Januari 2018 sampai dengan bulan Juni 2018.

3.1.2 Tempat Penelitian

Tempat penelitian Proyek Akhir ini akan dilakukan di wilayah Banyuwangi sebagai tempat studi literatur dan Laboratorium Komputer Politeknik Negeri Banyuwangi sebagai tempat pembuatan aplikasi.

3.1.3 Jadwal Penelitian

Jadwal kegiatan dibuat dengan tujuan untuk mengkonsep setiap kegiatan yang akan dikerjakan mulai dari tahap proses pengumpulan data sampai tahap proses pembuatan laporan agar apa yang dikerjakan dapat diselesaikan tepat waktu dan dapat berjalan sesuai jadwal seperti pada tabel 3.1

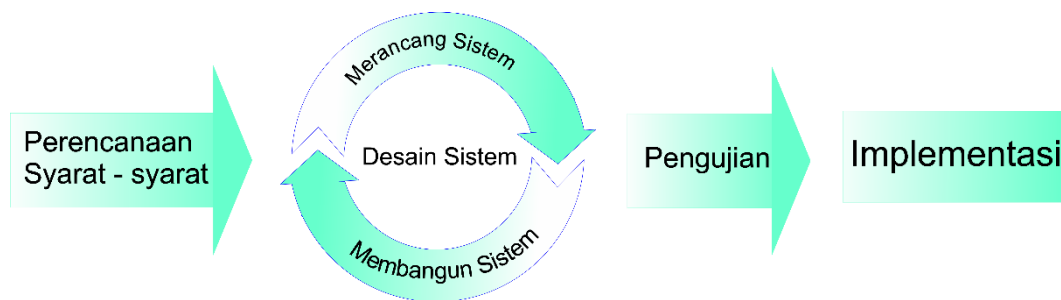
Tabel 3.1 Jadwal Kegiatan Proyek Akhir

Uraian kegiatan	Januari				Februari				Maret				April				Mei				Juni			
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
Analisa Kebutuhan																								
Perancangan dan Desain Sistem																								
Implementasi Desain																								
Pengujian Sistem																								
Perbaikan Sistem																								
Penyusunan Laporan																								

3.2 Metode yang Digunakan

Metode penelitian yang digunakan dalam penelitian Tugas Akhir dengan judul cetakin : aplikasi percetakan *online* berbasis web menggunakan metode *Rapid Application Development* (RAD).

Berikut tahapan penyelesaian proyek yang dikerjakan menggunakan metode RAD ditunjukkan pada Gambar 3.1



Gambar 3.1 Metode RAD Pengerjaan Tugas Akhir

Gambar 3.1 menunjukkan metode yang digunakan pada pengerjaan Tugas Akhir yang berjudul cetakin : percetakan *online* berbasis web yang terdiri dari:

3.2.1 Perencanaan Syarat – syarat

Perencanaan syarat - syarat merupakan tahap awal yang dilakukan untuk mengumpulkan informasi guna mendapatkan beberapa kebutuhan yang dianggap menunjang penelitian yang dilakukan.

Perancangan syarat - syarat dilakukan dengan melakukan studi literatur mengenai percetakan online serta melakukan wawancara kepada beberapa perusahaan percetakan terkait mengenai sistem yang akan dibuat.

Informasi atau data yang dihasilkan akan menjadi acuan sistem analis untuk diterjemahkan ke dalam pemrograman sesuai dengan hasil wawancara atau kebutuhan perusahaan dalam pembuatan sistem.

Dari hasil wawancara, dilakukan analisa untuk mengidentifikasi kebutuhan sistem, kegiatan analisa yang dilakukan adalah sebagai berikut:

a. Analisa Teknologi

Teknologi yang digunakan dalam pembuatan aplikasi percetakan online yaitu: menggunakan bahasa pemrograman PHP dengan *framework CodeIgniter* dan menggunakan basis data MySQL untuk penyimpanan data.

b. Analisa Informasi

Informasi yang dibutuhkan berupa data admin dari aplikasi cetakin yang dapat dilihat pada tabel 3.2

Tabel 3.2 Data admin aplikasi

Nama	Vensil Creative
Alamat	Dusun Krajan, RT/RW 02/02, Desa Sukojadi Kecamatan Blimbingsari, Kabupaten Banyuwangi.
Nomor telepon	08113581344
Email	vensilcreative@gmail.com

Tabel 3.2 merupakan data admin aplikasi cetakin : aplikasi percetakan *online* berbasis web. Data admin aplikasi dapat dilihat oleh semua pengunjung aplikasi yang terdapat pada menu hubungi kami.

3.2.2 Desain Sistem

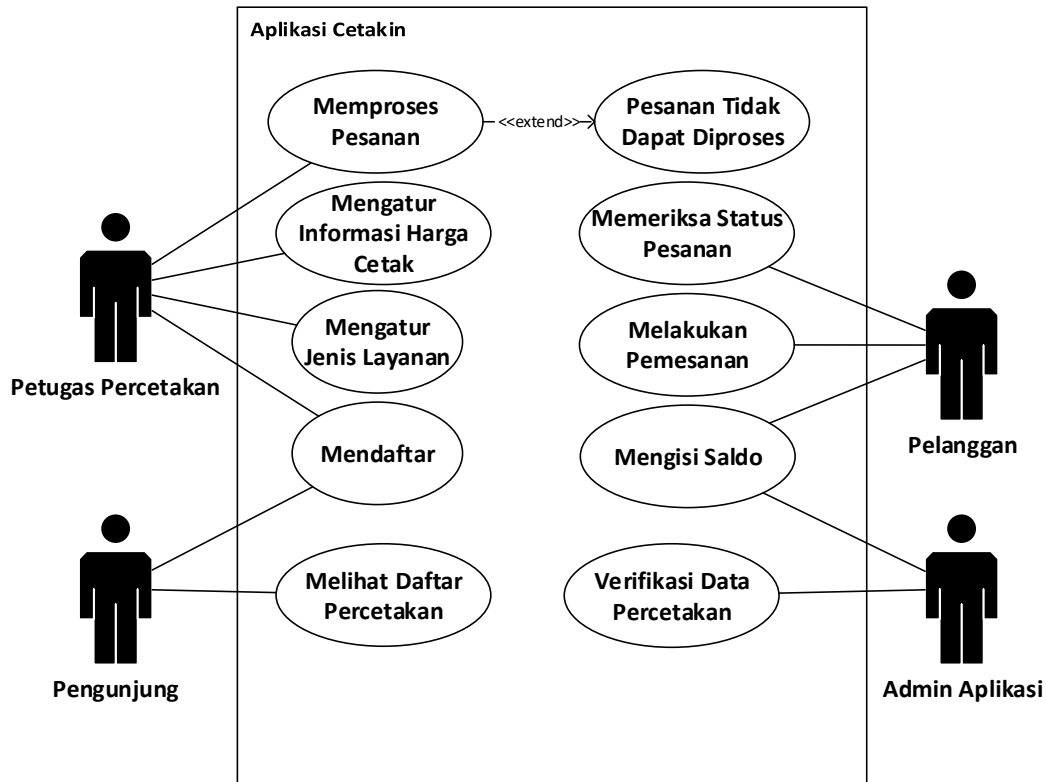
Pada tahap desain sistem terdapat dua tahap yaitu tahap bekerja dengan pengguna untuk merancang sistem dan tahap kedua adalah membangun sistem.

1. Merancang sistem

Tahapan dimana dilakukan penuangan pikiran dan perancangan sistem terhadap solusi dari permasalahan yang ada dengan menggunakan perangkat pemodelan diagram *use case* untuk mengetahui fungsi dari aplikasi yang dirancang, *entity relationship diagram* untuk merancang kebutuhan dari basis data yang dibuat, serta perancangan struktur aplikasi sehingga akan lebih memudahkan dalam pembuatan sistem.

A. Diagram *Use Case*

Diagram *use case* menggambarkan fungsionalitas sistem aplikasi percetakan *online* berbasis web ditunjukkan pada Gambar 3.2



Gambar 3.2 Diagram *Use Case*

Gambar 3.2 merupakan *use case* dari aplikasi cetakin yang menggambarkan fungsionalitas sistem dan interaksi antara pengguna aplikasi dengan sistem yang akan dibuat.

Tabel 3.3 Keterangan Aktor

No.	Aktor	Keterangan
1	Admin aplikasi	Merupakan seseorang yang bertugas dan memiliki hak akses secara keseluruhan untuk mengelola data percetakan, mengelola data petugas percetakan, mengelola saldo percetakan, mengelola data pelanggan, dan menambah saldo pelanggan.
2	Petugas percetakan	Merupakan seorang yang memproses pesanan dari pelanggan, mengubah status dari pesanan, mengatur layanan yang tersedia, mengelola info harga, dan memperbarui data percetakan.

3	Pelanggan	Merupakan seorang yang dapat melakukan pemesanan dengan memilih layanan percetakan, memilih percetakan, mengisi formulir pesanan, dan mengirim pesanan. Pelanggan juga memiliki hak akses untuk melihat daftar pesanan yang telah dilakukan.
4	Pengunjung	Merupakan seorang yang mengakses aplikasi tanpa masuk ke dalam sistem aplikasi. Pengunjung memiliki hak akses untuk melihat data percetakan, melihat syarat dan ketentuan aplikasi, dan melakukan pendaftaran.

Tabel 3.3 merupakan keterangan atau deskripsi dari setiap aktor yang memiliki interaksi dengan *use case* pada aplikasi cetakin.

Tabel 3.4 Deskripsi *Use Case* Mendaftar

<i>Use Case Name</i>	Mendaftar
<i>Actor</i>	Petugas percetakan, pengunjung
<i>Description</i>	Petugas percetakan mendaftar sebagai penyedia layanan dan pelanggan mendaftar sebagai pelanggan.
<i>Normal Course</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu daftar 2. Sistem menampilkan formulir pendaftaran 3. Aktor melengkapi formulir pendaftaran 4. Aktor mengirim formulir pendaftaran
<i>Alternate Course</i>	-
<i>Pre-Condition</i>	Belum mendaftar
<i>Post-Condition</i>	Telah terdaftar

Tabel 3.4 berisi deskripsi dari *use case* mendaftar. Terdapat dua aktor yang memiliki interaksi dengan *use case* mendaftar yaitu petugas percetakan dan pengunjung aplikasi. Petugas percetakan mendaftar sebagai penyedia layanan dan pelanggan mendaftar sebagai pelanggan agar dapat melakukan pemesanan.

Tabel 3.5 Deskripsi *Use Case* Verifikasi Data Percetakan

<i>Use Case Name</i>	Verifikasi data percetakan
<i>Actor</i>	Admin aplikasi
<i>Description</i>	Admin aplikasi melakukan verifikasi data percetakan yang telah melakukan pendaftaran
<i>Normal Course</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu kelola data percetakan 2. Sistem menampilkan data percetakan 3. Sistem menampilkan menu untuk melakukan verifikasi data 4. Aktor melakukan verifikasi data
<i>Alternate Course</i>	-
<i>Pre-Condition</i>	Data belum diverifikasi
<i>Post-Condition</i>	Data telah diverifikasi

Tabel 3.5 berisi deskripsi dari *use case* verifikasi data percetakan. Terdapat satu aktor yang memiliki interaksi dengan *use case* verifikasi data percetakan yaitu admin aplikasi. Admin aplikasi melakukan verifikasi data percetakan yang telah melakukan pendaftaran.

Tabel 3.6 Deskripsi *Use Case* Mengatur Jenis Layanan

<i>Use Case Name</i>	Mengatur jenis layanan
<i>Actor</i>	Petugas percetakan
<i>Description</i>	Petugas percetakan mengatur jenis layanan yang tersedia pada percetakan yang dikelola.
<i>Normal Course</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu data percetakan 2. Sistem menampilkan data percetakan 3. Sistem menampilkan menu untuk menentukan jenis layanan yang tersedia pada percetakan 4. Aktor mengatur jenis layanan
<i>Alternate Course</i>	-
<i>Pre-Condition</i>	Layanan belum ditentukan
<i>Post-Condition</i>	Layanan telah ditentukan

Tabel 3.6 berisi deskripsi dari *use case* mengatur jenis layanan. Terdapat satu aktor yang memiliki interaksi dengan *use case* mengatur jenis layanan yaitu petugas percetakan. Petugas percetakan dapat mengatur layanan yang disediakan oleh percetakan yang dikelola.

Tabel 3.7 Deskripsi *Use Case* Mengatur Informasi Harga Cetak

<i>Use Case Name</i>	Mengatur informasi harga cetak
<i>Actor</i>	Petugas percetakan
<i>Description</i>	Petugas percetakan mengatur informasi harga cetak dari percetakan yang dikelola.
<i>Normal Course</i>	<ol style="list-style-type: none">1. Aktor memilih menu kelola info harga2. Sistem menampilkan data info harga3. Sistem menampilkan menu untuk menentukan info harga4. Aktor mengatur info harga
<i>Alternate Course</i>	-
<i>Pre-Condition</i>	Informasi harga cetak belum diatur
<i>Post-Condition</i>	Informasi harga cetak tersimpan

Tabel 3.7 berisi deskripsi dari *use case* mengatur informasi harga cetak. Terdapat satu aktor yang memiliki interaksi dengan *use case* mengatur informasi harga cetak yaitu petugas percetakan. Petugas percetakan dapat mengatur informasi harga cetak dari percetakan yang dikelola.

Tabel 3.8 Deskripsi *Use Case* Melihat Daftar Percetakan

<i>Use Case Name</i>	Melihat daftar percetakan
<i>Actor</i>	Pengunjung
<i>Description</i>	Pengunjung melihat daftar percetakan yang terdapat pada aplikasi.
<i>Normal Course</i>	<ol style="list-style-type: none">1. Aktor memilih menu daftar percetakan2. Sistem menampilkan daftar percetakan
<i>Alternate Course</i>	-
<i>Pre-Condition</i>	Daftar percetakan belum ditampilkan
<i>Post-Condition</i>	Daftar percetakan ditampilkan

Tabel 3.8 berisi deskripsi dari *use case* melihat daftar percetakan. Terdapat satu aktor yang memiliki interaksi dengan *use case* melihat daftar percetakan yaitu pengunjung aplikasi. Pengunjung aplikasi dapat melihat daftar percetakan yang terdapat pada aplikasi cetakin.

Tabel 3.9 Deskripsi *Use Case* Mengisi Saldo

<i>Use Case Name</i>	Mengisi saldo
<i>Actor</i>	Pelanggan, admin aplikasi
<i>Description</i>	Aktor melakukan pengisian saldo.
<i>Normal Course</i>	<ol style="list-style-type: none">1. Pelanggan melakukan pembayaran kepada admin aplikasi dengan cara transfer2. Pelanggan membuka menu isi saldo3. Sistem menampilkan formulir isi saldo4. Pelanggan mengisi formulir isi saldo dan mengunggah bukti pembayaran5. Pelanggan mengirim formulir isi saldo6. Admin aplikasi melakukan verifikasi data saldo7. Admin aplikasi mengisi saldo pelanggan
<i>Alternate Course</i>	-
<i>Pre-Condition</i>	Saldo belum diisi
<i>Post-Condition</i>	Saldo telah diisi

Tabel 3.9 berisi deskripsi dari *use case* mengisi saldo. Terdapat dua aktor yang memiliki interaksi dengan *use case* mengisi saldo yaitu pelanggan dan admin aplikasi. Pelanggan melakukan pengisian saldo dengan cara transfer kepada admin aplikasi. Setelah melakukan transfer, pelanggan membuka aplikasi dan mengisi formulir pengisian saldo serta mengunggah bukti pembayaran. Admin aplikasi melakukan verifikasi data pengisian saldo dari pelanggan untuk diproses.

Tabel 3.10 Deskripsi *Use Case* Melakukan Pemesanan

<i>Use Case Name</i>	Melakukan pemesanan
<i>Actor</i>	Pelanggan
<i>Description</i>	Pelanggan melakukan pemesanan.
<i>Normal Course</i>	<ol style="list-style-type: none"> 1. Aktor masuk ke sistem aplikasi 2. Sistem menampilkan layanan 3. Aktor memilih layanan 4. Sistem menampilkan daftar percetakan 5. Aktor memilih percetakan 6. Sistem menampilkan formulir pesanan 7. Aktor mengisi formulir pesanan dan mengunggah <i>file</i> 8. Aktor mengirim pesanan
<i>Alternate Course</i>	<p>7a. Gagal mengunggah file</p> <p>8a. Gagal mengirim pesanan karena formulir pesanan tidak diisi dengan lengkap</p>
<i>Pre-Condition</i>	Belum melakukan pemesanan
<i>Post-Condition</i>	Pesanan telah tersimpan

Tabel 3.10 berisi deskripsi dari *use case* melakukan pemesanan. Terdapat satu aktor yang memiliki interaksi dengan *use case* melakukan pemesanan yaitu pelanggan. Pelanggan dapat melakukan pemesanan setelah masuk ke dalam sistem aplikasi.

Tabel 3.11 Deskripsi *Use Case* Memeriksa Status Pesanan

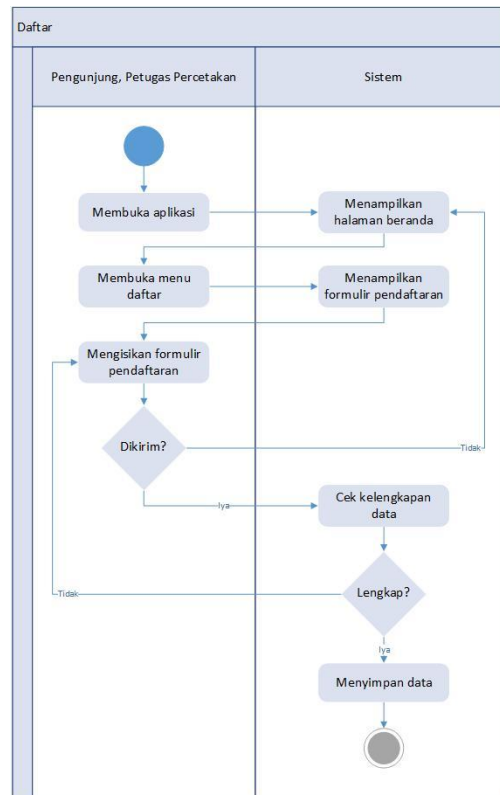
<i>Use Case Name</i>	Memeriksa status pesanan
<i>Actor</i>	Pelanggan
<i>Description</i>	Aktor memeriksa status dari pesanan
<i>Normal Course</i>	<ol style="list-style-type: none"> 1. Aktor memilih menu daftar pesanan 2. Sistem menampilkan daftar dan status pesanan
<i>Alternate Course</i>	-
<i>Pre-Condition</i>	Status pesanan belum diperiksa
<i>Post-Condition</i>	Status pesanan ditampilkan

Tabel 3.11 berisi deskripsi dari *use case* memeriksa status pesanan. Terdapat satu aktor yang memiliki interaksi dengan *use case* memeriksa status pesanan yaitu pelanggan. Pelanggan dapat memeriksa status dari pesanan yang telah dikirim apakah pesanan ditolak, belum diproses, sedang diproses, selesai ataukah sudah diambil.

B. Diagram Activity

Berikut merupakan diagram aktivitas yang menggambarkan alur kerja atau aktivitas dari sistem aplikasi percetakan *online* berbasis web yang akan dibuat.

1. Diagram Activity Daftar

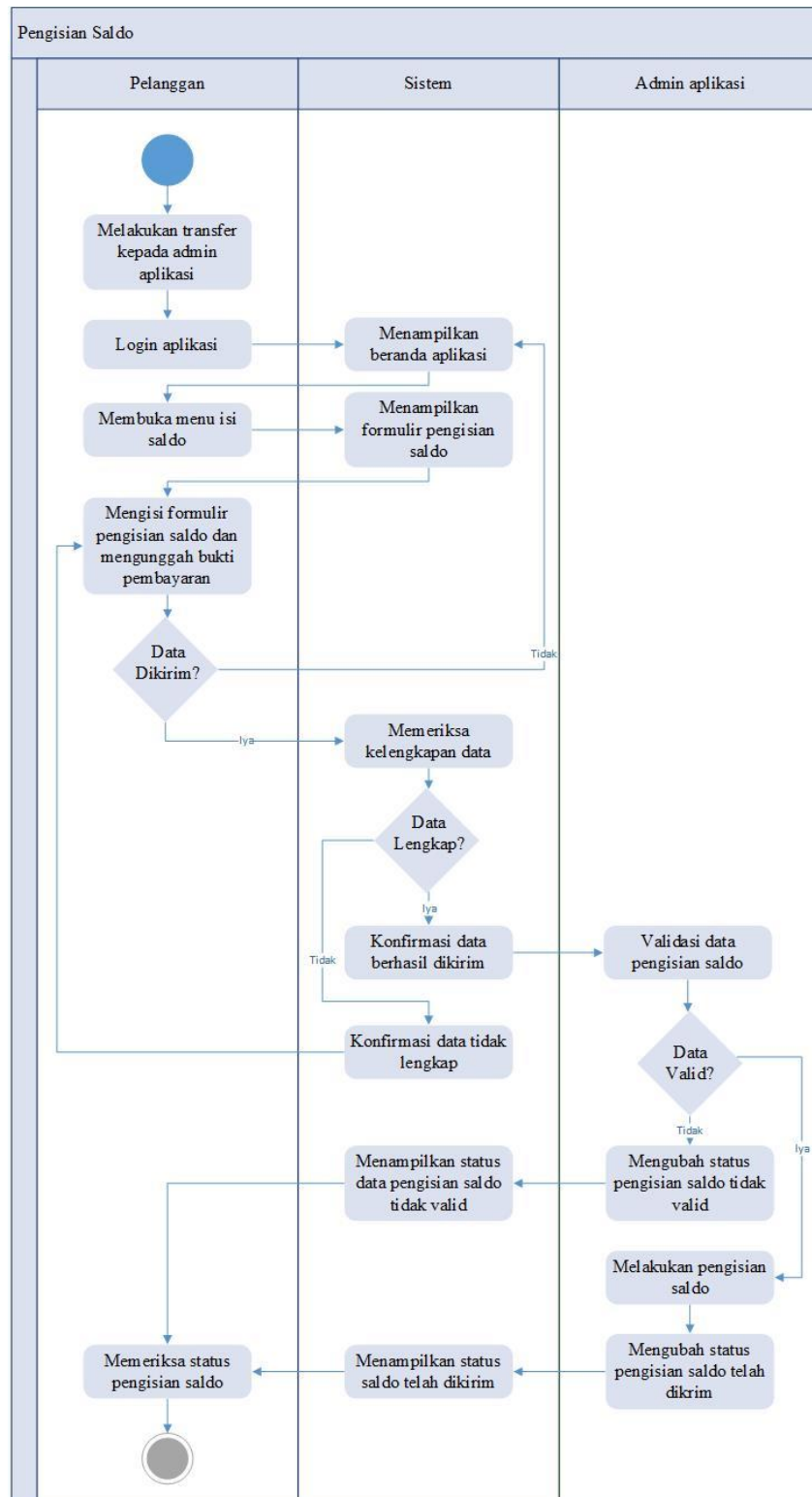


Gambar 3.3 Diagram Activity Daftar

Pada gambar 3.3 merupakan diagram *activity* daftar yang memperlihatkan proses pendaftaran akun pada aplikasi dengan penjelasan sebagai berikut:

- 1) Pengguna membuka aplikasi kemudian memilih menu daftar.
- 2) Sistem menampilkan formulir pendaftaran.
- 3) Pengguna mengisi formulir pendaftaran.
- 4) Apabila formulir pesanan tidak dikirim, sistem menampilkan halaman beranda.
- 5) Apabila formulir pendaftaran dikirim, sistem melakukan pengecekan kelengkapan data. Jika data lengkap, data disimpan kedalam basis data *server*. Jika tidak, pengguna kembali melengkapi formulir pendaftaran.

2. Diagram Activity Pengisian Saldo

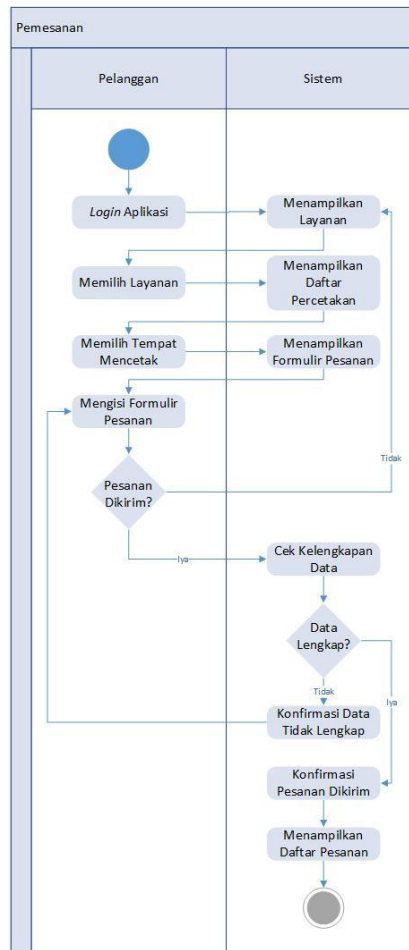


Gambar 3.4 Diagram Activity Pengisian Saldo

Pada gambar 3.6 merupakan diagram *activity* pengisian saldo yang memperlihatkan alur dalam membeli dan mengisi saldo pelanggan pada cetakin dengan penjelasan sebagai berikut:

- 1) Pelanggan melakukan pembayaran saldo kepada admin aplikasi dengan cara transfer.
- 2) Pelanggan masuk ke sistem aplikasi dengan menggunakan *email* dan *password* yang dimiliki.
- 3) Setelah berhasil masuk, sistem menampilkan halaman beranda pelanggan.
- 4) Pelanggan membuka menu isi saldo.
- 5) Sistem menampilkan formulir pengisian saldo.
- 6) Pelanggan mengisi formulir pengisian saldo serta mengunggah bukti pembayaran.
- 7) Apabila data tidak dikirim, sistem menampilkan halaman beranda. Apabila data dikirim, sistem memeriksa kelengkapan data.
- 8) Jika data tidak lengkap, sistem menampilkan konfirmasi bahwa data tidak lengkap dan pelanggan kembali melengkapi formulir pengisian saldo.
- 9) Jika data lengkap, sistem menampilkan konfirmasi bahwa data berhasil dikirim dan admin aplikasi melakukan validasi data.
- 10) Apabila data pengisian saldo tidak valid, admin aplikasi mengubah status data pengisian saldo tidak valid kemudian ditampilkan oleh sistem.
- 11) Apabila data pengisian saldo valid, admin aplikasi melakukan pengisian saldo dan mengubah status data pengisian saldo telah dikirim kemudian ditampilkan oleh sistem.
- 12) Pelanggan memeriksa status pengisian saldo.

3. Diagram Activity Pemesanan



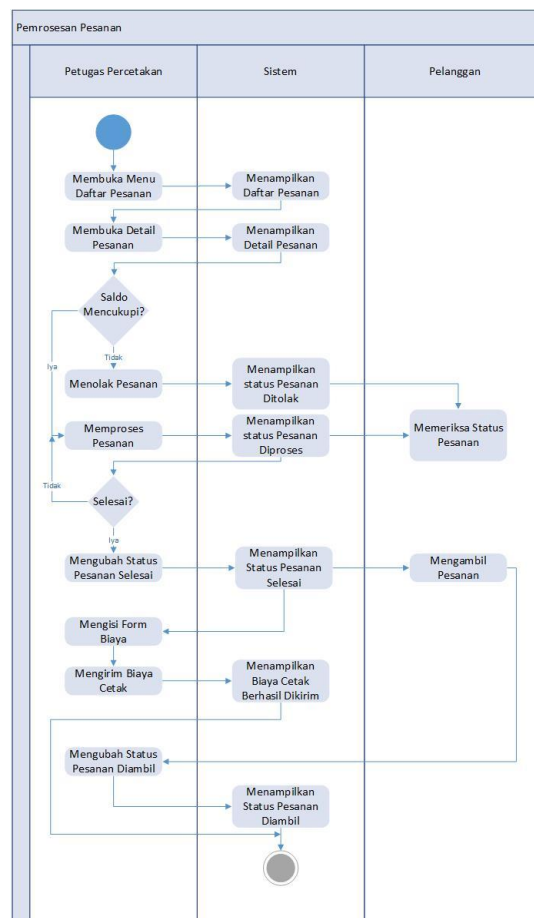
Gambar 3.5 Diagram Activity Pemesanan

Pada gambar 3.4 merupakan diagram *activity* pemesanan yang memperlihatkan proses pemesanan menggunakan aplikasi cetakin dengan penjelasan sebagai berikut:

- 1) Pelanggan masuk ke dalam sistem aplikasi dengan menggunakan *email* dan *password* yang telah tersimpan pada basis data *server*.
- 2) Sistem menampilkan layanan yang tersedia yaitu cetak dokumen dan cetak foto.
- 3) Pelanggan memilih layanan.
- 4) Setelah pelanggan memilih layanan, sistem menampilkan daftar percetakan sesuai layanan yang dipilih.
- 5) Pelanggan memilih tempat mencetak.

- 6) Setelah pelanggan memilih tempat mencetak, sistem menampilkan formulir pesanan.
- 7) Pelanggan mengisi formulir pesanan dan mengunggah *file* yang akan dicetak.
- 8) Jika pesanan dikirim, sistem melakukan pengecekan terhadap kelengkapan data pesanan. Apabila data lengkap, sistem menampilkan konfirmasi pesanan dikirim. Jika tidak lengkap, sistem menampilkan konfirmasi data pesanan tidak lengkap dan pelanggan melengkapi formulir pesanan kembali.
- 9) Jika pesanan dibatalkan, sistem menampilkan layanan percetakan kembali.
- 10) Setelah sistem menampilkan konfirmasi pesanan dikirim. Sistem menampilkan daftar pesanan.

4. Diagram *Activity* Pemrosesan Pesanan



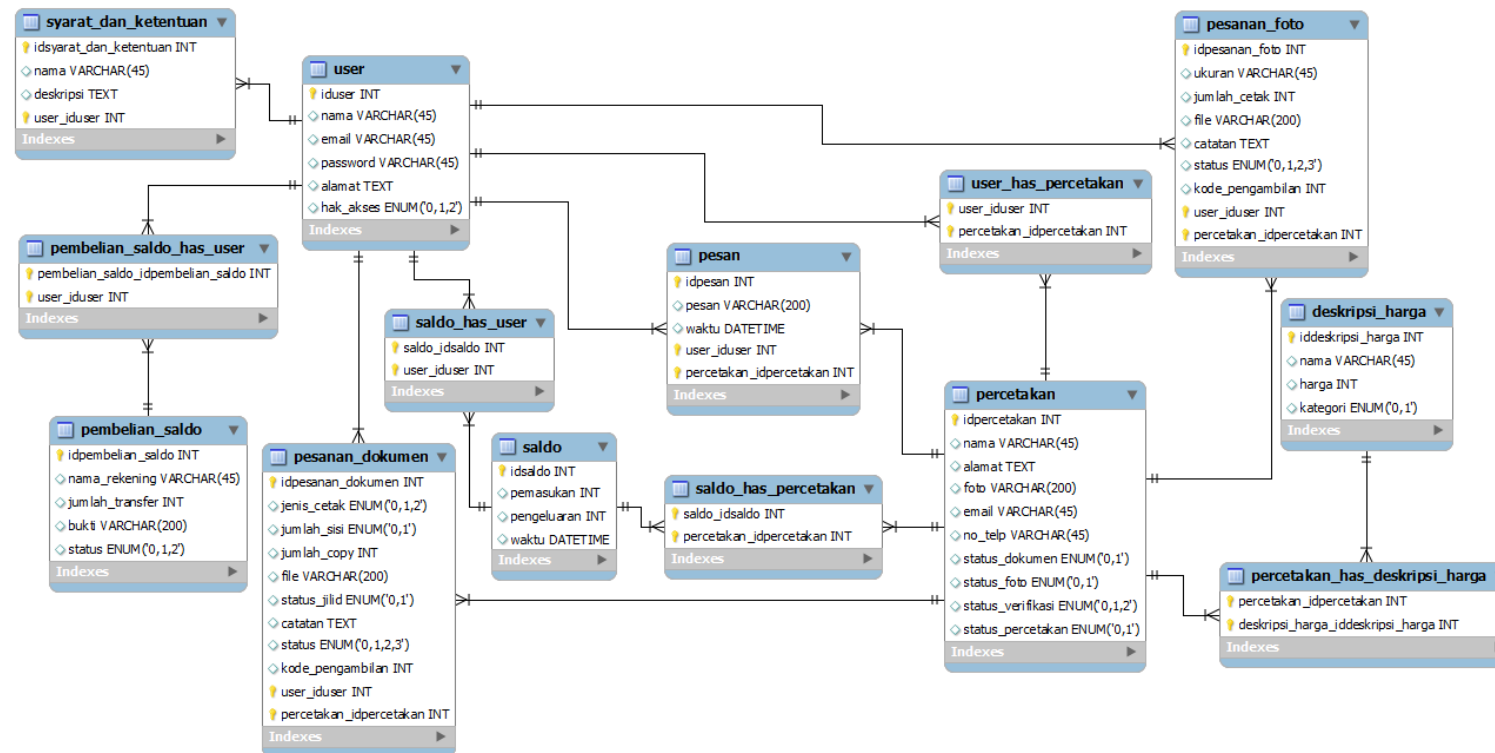
Gambar 3.6 Diagram *Activity* Pemrosesan Pesanan

Pada gambar 3.5 merupakan diagram *activity* pemrosesan pesanan yang memperlihatkan alur pemrosesan pesanan pada cetakin dengan penjelasan sebagai berikut:

- 1) Petugas percetakan membuka menu daftar pesanan.
- 2) Sistem menampilkan daftar pesanan.
- 3) Petugas percetakan membuka detail pesanan.
- 4) Sistem menampilkan detail pesanan.
- 5) Petugas percetakan melakukan pengecekan saldo pelanggan apakah kurang atau melebihi dari biaya pesanan yang akan dicetak.
- 6) Jika saldo tidak mencukupi, petugas percetakan menolak pesanan dengan mengubah status pesanan menjadi ditolak.
- 7) Sistem menampilkan status pesanan ditolak.
- 8) Jika saldo mencukupi, petugas percetakan memproses pesanan dan mengubah status pesanan menjadi diproses.
- 9) Sistem menampilkan status pesanan sedang diproses.
- 10) Apabila pesanan telah selesai diproses. Petugas percetakan mengubah status pesanan menjadi selesai dan mengisi form biaya cetak.
- 11) Sistem menampilkan status pesanan telah selesai.
- 12) Setelah mendapat konfirmasi pesanan selesai, pelanggan mengambil pesanan dengan menyebutkan kode pengambilan dan nama akun.
- 13) Petugas percetakan mengubah status pesanan diambil.
- 14) Sistem menampilkan konfirmasi pesanan telah diambil.

C. Entity Relationship Diagram (ERD)

Berikut merupakan rancangan relasi basis data yang akan dibuat dalam pengerjaan aplikasi percetakan *online* berbasis web ditunjukkan pada gambar 3.7



Gambar 3.7 *Entity Relationship Diagram*

Pada gambar 3.7 merupakan pemodelan struktur data *Entity Relationship Diagram* dari aplikasi cetakin : aplikasi percetakan *online* berbasis *web* untuk menjelaskan hubungan antar tabel yang saling terkait.

Tabel 3.12 Tabel *User*

Nama Atribut	Tipe Data	Kata Kunci	Keterangan
iduser	Int	Primary Key	Berisi id unik user
email	Varchar(45)		Berisi email user untuk login
password	Varchar(45)		Berisi password user untuk login
nama	Varchar(45)		Berisi nama user
alamat	Text		Berisi alamat user
hak_akses	Enum		Berisi hak akses dari user untuk menentukan akses user.

Tabel 3.12 merupakan tabel *user* yang digunakan untuk menyimpan data pengguna aplikasi cetakin. *User* terdiri dari pelanggan, petugas percetakan dan admin aplikasi.

Tabel 3.13 Tabel Percetakan

Nama Atribut	Tipe Data	Kata Kunci	Keterangan
idpercetakan	Int	Primary Key	Berisi id unik percetakan
nama	Varchar(45)		Berisi nama percetakan
alamat	Text		Berisi alamat percetakan
foto	Varchar(200)		Berisi foto percetakan
email	Varchar(45)		Beisi email percetakan
no_telp	Varchar(45)		Berisi nomor telepon percetakan
status_dokumen	Enum		Berisi status layanan dokumen apakah tersedia atau tidak tersedia
status_foto	Enum		Berisi status layanan foto apakah tersedia atau tidak tersedia
status_verifikasi	Enum		Berisi status verifikasi data percetakan apakah belum diverifikasi, ditolak atau diterima
status_percetakan	Enum		Berisi status dari percetakan apakah sedang buka atau tutup

Tabel 3.13 merupakan tabel percetakan yang digunakan untuk menyimpan data dari setiap percetakan.

Tabel 3.14 Tabel Pesanan Dokumen

Nama Atribut	Tipe Data	Kata Kunci	Keterangan
idpesanan_dokumen	Int	Primary Key	Berisi id unik dari pesanan dokumen
jenis_cetak	Enum		Berisi jenis cetak dokumen apakah print copy, normal, atau high quality
jumlah_sisi	Enum		Berisi jumlah sisi dokumen apakah satu sisi atau dua sisi
jumlah_copy	Int		Berisi jumlah copy dokumen
file	Varchar(200)		Berisi file yang akan dicetak
status_jilid	Enum		Berisi status jilid dokumen apakah iya atau tidak
catatan	Text		Berisi catatan dari pelanggan
kode_pengambilan	int		Berisi kode untuk mengambil pesanan.
status	Enum		Berisi status pesanan apakah belum diproses, ditolak, diproses, selesai, atau telah diambil
iduser	Int	Foreign Key	Berisi iduser dari tabel user
idpercetakan	Int	Foreign Key	Berisi idpercetakan dari tabel percetakan

Tabel 3.14 merupakan tabel pesanan dokumen yang digunakan untuk menyimpan data pesanan dokumen.

Tabel 3.15 Tabel Pesanan Foto

Nama Atribut	Tipe Data	Kata Kunci	Keterangan
idpesanan_foto	Int	Primary Key	Berisi id unik dari pesanan foto
ukuran	Varchar(45)		Berisi ukuran foto yang akan dicetak
jumlah_cetak	Int		Berisi jumlah cetak foto
file	Varchar(200)		Berisi file yang akan dicetak
catatan	Text		Berisi catatan dari pelanggan
kode_pengambilan	int		Berisi kode untuk mengambil pesanan.
status	Enum		Berisi status pesanan apakah belum diproses, ditolak, diproses, selesai, atau telah diambil.
iduser	Int	Foreign Key	Berisi iduser dari tabel user
idpercetakan	Int	Foreign Key	Berisi idpercetakan dari tabel percetakan

Tabel 3.15 merupakan tabel pesanan foto yang digunakan untuk menyimpan data pesanan foto. Tabel pesanan foto berelasi dengan tabel *user* dan tabel percetakan. Sehingga dari iduser dapat diketahui data pengirim pesanan dan dari idpercetakan dapat diketahui percetakan yang dituju.

Tabel 3.16 Tabel Pesan

Nama Atribut	Tipe Data	Kata Kunci	Keterangan
idpesan	Int	Primary Key	Berisi id unik dari pesan
pesan	Varchar(200)		Berisi pesan yang dikirim
waktu	Datetime		Berisi waktu pengiriman pesan
iduser	Int	Foreign Key	Berisi iduser dari tabel user
idpercetakan	Int	Foreign Key	Berisi idpercetakan dari tabel percetakan

Tabel 3.16 merupakan tabel pesan yang digunakan untuk menyimpan pesan antara pelanggan dan petugas percetakan. Tabel pesan memiliki relasi dengan tabel *user* dan tabel percetakan. Sehingga dari iduser dapat diketahui data pengirim pesan dan dari idpercetakan dapat diketahui percetakan yang dituju begitu juga dengan sebaliknya.

Tabel 3.17 Tabel Pembelian Saldo

Nama Atribut	Tipe Data	Kata Kunci	Keterangan
idpembelian_saldo	Int	Primary Key	Berisi id unik dari pembelian saldo
nama_rekening	Varchar(45)		Berisi nama rekening yang digunakan untuk transfer
jumlah_transfer	Int		Berisi jumlah saldo yang dibeli
bukti	Varchar		Berisi bukti pembayaran yang telah dilakukan
status	Enum		Berisi status dari saldo apakah belum divalidasi, ditolak atau diterima

Tabel 3.17 merupakan tabel pembelian saldo yang digunakan untuk menyimpan data pembelian saldo. Tabel pembelian saldo berelasi dengan tabel *user*. Sehingga dari iduser dapat diketahui data user yang membeli saldo.

Tabel 3.18 Tabel Saldo

Nama Atribut	Tipe Data	Kata Kunci	Keterangan
idsaldo	Int	Primary Key	Berisi id unik dari saldo
deskripsi	Varchar(200)		Berisi deskripsi keluar atau masuknya saldo
pemasukan	Int		Berisi nilai saldo yang masuk
pengeluaran	Int		Berisi nilai saldo yang keluar
waktu	Datetime		Berisi waktu keluar atau masuknya saldo

Tabel 3.18 merupakan tabel saldo yang digunakan untuk menyimpan data saldo.

Tabel 3.19 Tabel Syarat dan Ketentuan

Nama Atribut	Tipe Data	Kata Kunci	Keterangan
idsyarat_dan_ketentuan	Int	Primary Key	Berisi id unik dari syarat dan ketentuan
nama	Varchar(45)		Berisi nama dari syarat dan ketentuan
Deskripsi	Text		Berisi deskripsi dari syarat dan ketentuan
iduser	Int		Berisi iduser dari tabel user

Tabel 3.19 merupakan tabel syarat dan ketentuan yang digunakan untuk menyimpan data syarat dan ketentuan dari aplikasi. Tabel syarat dan ketentuan berelasi dengan tabel *user* sehingga dari iduser dapat diketahui data user yang dapat mengelola syarat dan ketentuan.

Tabel 3.20 Tabel Deskripsi Harga

Nama Atribut	Tipe Data	Kata Kunci	Keterangan
iddeskripsi_harga	Int	Primary Key	Berisi id unik dari deskripsi harga
nama	Varchar(200)		Berisi nama dari deskripsi harga
harga	Int		Berisi harga dari nama deskripsi
kategori	emun		Berisi kategori deskripsi harga apakah dokumen atau foto

Tabel 3.20 merupakan tabel deskripsi harga yang digunakan untuk menyimpan data deskripsi harga dari layanan percetakan.

2. Membangun sistem

Membangun sistem merupakan tahapan implementasi hasil dari desain sistem yang telah dibuat. Implementasi dari diagram *use case* dan diagram *activity* diubah menjadi bentuk tampilan di dalam bahasa komputer yaitu kedalam bahasa pemrograman melalui proses *coding* dengan menggunakan bahasa pemrograman PHP, *framework CodeIgniter*, dan web *service* Apache, karena sistem ini dibuat berbasis web. Sedangkan untuk implementasi ERD menggunakan basis data MySQL.

Kedua tahap diatas dilakukan terus menerus hingga sistem dapat berjalan dengan baik dan sesuai dengan harapan pengguna.

3.2.3 Pengujian

Pengujian program merupakan tahapan dimana aplikasi yang telah dibuat kemudian diuji untuk menemukan kekurangan atau kelebihan dari sistem yang kemudian dilakukan pengkajian ulang dan perbaikan terhadap aplikasi agar aplikasi menjadi lebih baik.

Pada aplikasi cetakin ini pengujian dilakukan dengan menggunakan metode *black box testing*. Pada metode ini, pengujian yang dilakukan lebih mengarah ke tampilan luar dan fungsional dari aplikasi.

3.2.4 Implementasi Sistem

Tahap terakhir adalah tahapan implementasi sistem. Tahapan implepentasi sistem merupakan tahapan penggunaan sistem atau penerapan sistem. Pada tahapan ini dilakukan apabila telah mendapat persetujuan dari calon pengguna dan segala aspek telah terpenuhi.

Penerapan aplikasi cetakin untuk saat ini hanya dilakukan diperusahaan percetakan daerah Banyuwangi. Apliksai cetakin akan dipublikasikan menggunakan hosting dengan nama domain www.cetakin.com. Selama proses penerapan program harus dilakukan pemeliharaan agar aplikasi yang telah diterapkan dapat terjaga dari kesalahan kecil yang mungkin terjadi atau tidak ditemukan sebelumnya.

3.3 Gambaran Umum Sistem

Gambaran umum sangatlah diperlukan dalam melakukan pembuatan ataupun pengembangan suatu sistem untuk mengetahui pembaharuan atau pengembangan yang akan lakukan terhadap sistem yang telah berjalan sebelumnya. Pada gambaran umum sistem terdapat dua pokok bahasan yang akan dijelaskan yaitu, gambaran umum sistem yang berjalan dan gambaran umum sistem yang diusulkan.

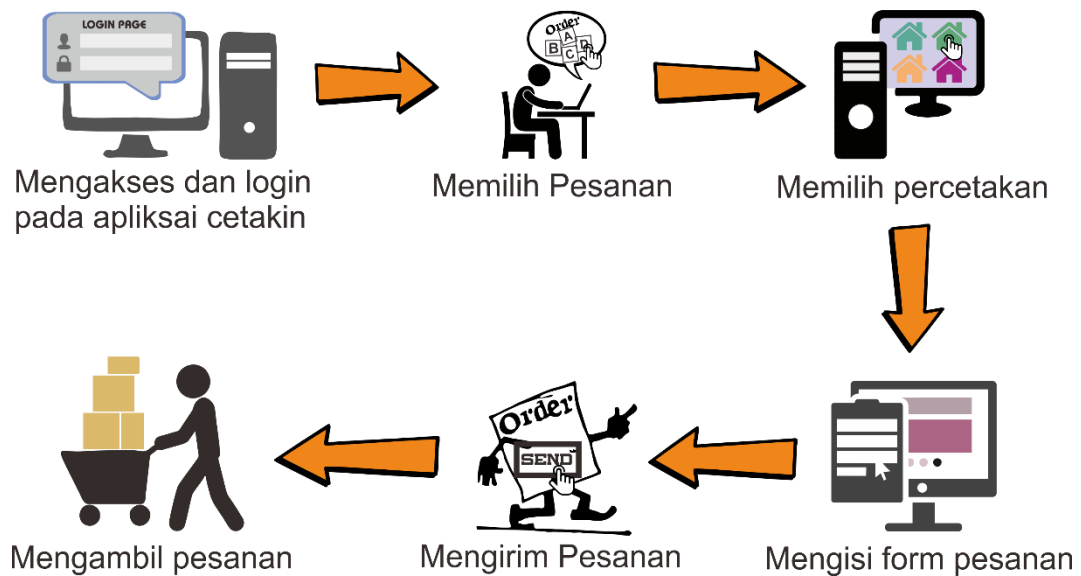
3.3.1 Gambaran Sistem yang Berjalan



Gambar 3.8 Gambaran Umum Sistem yang Berjalan

Gambar 3.8 merupakan alur gambaran umum sistem yang telah berjalan saat ini, ketika pelanggan ingin mencetak *file*, mereka diharuskan datang langsung ke tempat percetakan untuk mengirimkan *file* yang akan dicetak menggunakan *flashdisk* sekaligus memastikan apakah percetakan tersebut sedang beroperasi ataukah sedang tutup. Sesampainya di tempat percetakan, pelanggan diharuskan mengikuti antrian apabila banyak pelanggan yang juga ingin mencetak file di tempat percetakan tersebut. Kemudian pelanggan mengirimkan *file* yang akan dicetak kepada petugas percetakan untuk diproses dengan menggunakan *flasdisk* atau alat penyimpanan lainnya, tidak jarang pula banyak data pelanggan yang tidak dapat dibuka bahkan hilang karena terkena serangan virus dari komputer percetakan. Sebelum pesanan selesai dicetak, pelanggan diharuskan menunggu di tempat percetakan atau dengan membayar uang muka sebesar 50% dari total pembayaran apabila mereka ingin meninggalkan pesanan tersebut. Pelanggan dapat mengambil pesanan ketika pesanan telah selesai dicetak dan dibayar sesuai dengan harga yang harus dibayarkan.

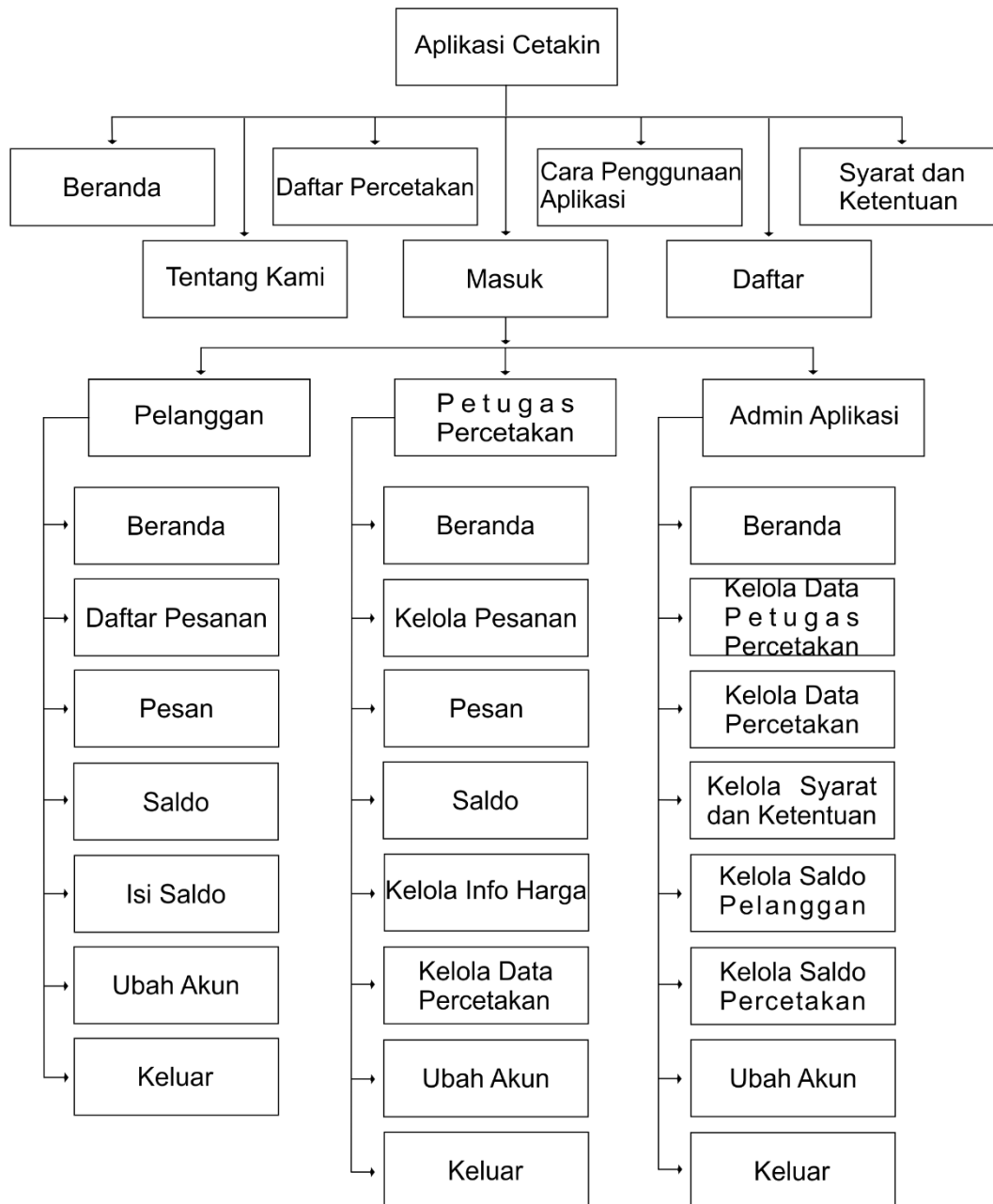
3.3.2 Gambaran Sistem yang Diusulkan



Gambar 3.9 Gambaran Umum Sistem yang Diusulkan

Gambar 3.9 merupakan alur gambaran umum sistem yang diusulkan. Gambaran sistem yang diusulkan adalah pelanggan mengakses aplikasi cetakin menggunakan komputer atau telepon pintar mereka yang terhubung dengan jaringan *internet* untuk masuk ke dalam sistem aplikasi agar dapat melakukan pemesanan. Kemudian pelanggan memilih layanan yang tersedia pada aplikasi cetakin. Setelah memilih layanan, pelanggan dapat memilih tempat percetakan yang tergabung dalam aplikasi cetakin untuk menentukan dimana pesanan tersebut akan dicetak. Proses selanjutnya yaitu pelanggan mengisi formulir pesanan dan memasukkan *file* yang akan dicetak kemudian mengirim pesanan sampai pada proses akhir yaitu memantau status pesanan apakah pesanan diterima, ditolak, diproses ataukah selesai. Pelanggan dapat mengambil pesanan apabila status pesanan telah selesai.

3.4 Struktur Menu



Gambar 3.10 Struktur Menu Aplikasi

Gambar 3.10 menunjukkan struktur menu yang dirancang pada aplikasi percetakan *online* berbasis web. Daftar menu yang tersedia pada struktur menu aplikasi percetakan *online* berbasis web ditunjukkan pada Tabel 3.21.

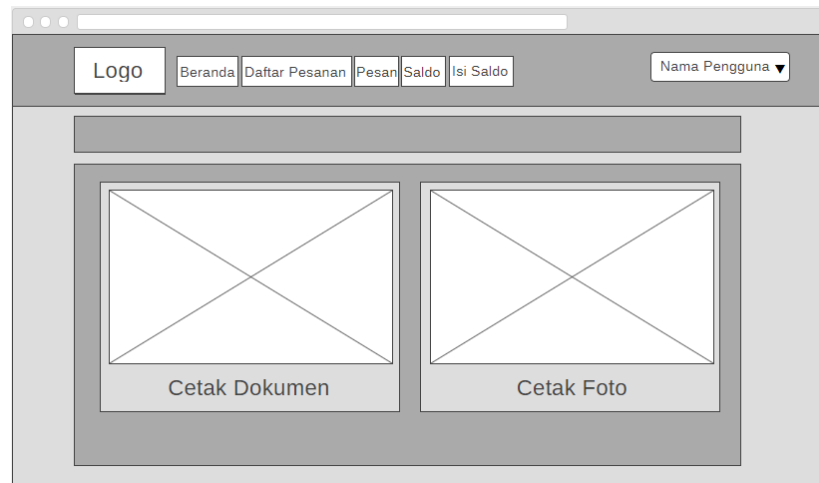
Table 3.21 Daftar Menu Aplikasi Percetakan *Online* Berbasis Web

No	Daftar Menu	Keterangan
1	Beranda	Beranda merupakan halaman utama dari aplikasi cetakin. Tampilan atau konten halaman beranda dari setiap <i>user</i> akan berbeda sesuai dengan hak akses yang dimiliki.
2	Daftar percetakan	Daftar percetakan merupakan menu yang digunakan oleh pengunjung aplikasi untuk melihat daftar percetakan yang terdapat pada aplikasi cetakin.
3	Cara penggunaan aplikasi	Cara penggunaan aplikasi merupakan menu yang digunakan oleh pengunjung aplikasi untuk mengetahui bagaimana cara menggunakan aplikasi cetakin.
4	Syarat dan ketentuan	Syarat dan ketentuan berisi syarat yang harus dipenuhi oleh pengguna aplikasi beserta ketentuan–ketentuan yang ditetapkan pada aplikasi cetakin. Syarat dan ketentuan dapat dilihat oleh semua pengguna aplikasi.
5	Hubungi kami	Hubungi kami berisi <i>contact person</i> dari admin aplikasi yang dapat dihubungi. Menu hubungi kami meliputi nomor telepon, alamat <i>email</i> , nama pengelola aplikasi, dan alamat pengelola aplikasi.
6	Masuk	Fitur ini digunakan untuk masuk kedalam aplikasi sesuai dengan hak akses. <i>User</i> dapat masuk ke dalam sistem aplikasi dengan memasukkan <i>email</i> dan <i>password</i> yang sudah terdaftar pada basis data <i>server</i> .
7	Daftar	Fitur ini digunakan untuk mendaftar pada aplikasi cetakin. Pengunjung aplikasi yang telah melakukan pendaftaran dapat masuk ke dalam sistem aplikasi menggunakan <i>email</i> dan <i>password</i> yang telah tersimpan pada basis data <i>server</i> . Sedangkan untuk petugas percetakan harus menunggu verifikasi dari admin aplikasi untuk masuk kedalam sistem.
8	Daftar pesanan	Daftar pesanan merupakan menu yang digunakan oleh pelanggan untuk melihat daftar pesanan yang telah dikirim. Pada menu ini pelanggan juga dapat memantau status dari pesannya
9	Pesan	Pesan digunakan oleh pelanggan dan petugas percetakan untuk mengirim dan menerima pesan.
10	Saldo	Saldo merupakan menu yang digunakan oleh pelanggan dan petugas percetakan untuk melihat saldo yang dimiliki.

11	Ubah akun	Fitur ini untuk mengubah <i>akun</i> atau data pribadi. Data yang telah diubah dapat dibatalkan dengan memilih tombol batalkan atau disimpan dengan memilih tombol simpan.
12	Keluar	Fitur ini digunakan untuk keluar dari sistem aplikasi. Fitur ini digunakan dengan tujuan untuk menjaga keamanan <i>akun</i> yang telah digunakan agar tidak disalah gunakan oleh orang lain.
13	Kelola pesanan	Petugas percetakan melakukan pengelolaan pesanan yang diterima dengan menggunakan fitur kelola pesanan. Pada menu ini petugas percetakan juga dapat mengubah status dari pesanan apakah pesanan ditolak, belum diproses, sedang diproses atau sudah selesai.
14	Isi saldo	Menu isi saldo digunakan oleh pelanggan untuk mengisi formulir pengisian saldo. Pelanggan dapat mengisi formulir pengisian saldo dan mengunggah bukti pembayaran.
15	Kelola info harga	Kelola info harga digunakan oleh petugas percetakan untuk mengelola info harga dari percetakan yang dikelola.
16	Kelola data percetakan	Kelola data percetakan digunakan oleh petugas percetakan dan admin aplikasi untuk mengelola data percetakan. Pada petugas percetakan menu ini berisi data percetakan yang dikelola. Sedangkan pada admin aplikasi menu ini berisi semua data percetakan yang mendaftar pada aplikasi cetakin serta digunakan untuk melakukan verifikasi data percetakan.
17	Kelola data petugas percetakan	Kelola data petugas percetakan digunakan oleh admin aplikasi untuk mengelola data petugas percetakan. Menu ini berisi data dari petugas percetakan yang bekerja sama dengan aplikasi cetakin.
18	Kelola syarat dan ketentuan	Menu ini digunakan oleh admin aplikasi untuk mengelola syarat dan ketentuan. Syarat dan ketentuan berisi syarat yang harus dipenuhi oleh pengguna dan ketentuan–ketentuan yang ditetapkan pada aplikasi cetakin.
19	Kelola saldo pelanggan	Kelola saldo pelanggan merupakan menu yang digunakan oleh admin aplikasi untuk mengelola saldo pelanggan.
20	Kelola saldo percetakan	Menu ini digunakan oleh admin aplikasi untuk mengelola saldo dari setiap percetakan. Saldo akan ditransfer kepada petugas percetakan setelah memenuhi batas maksimal saldo yaitu lima ratus ribu rupiah.

3.5 Desain Aplikasi

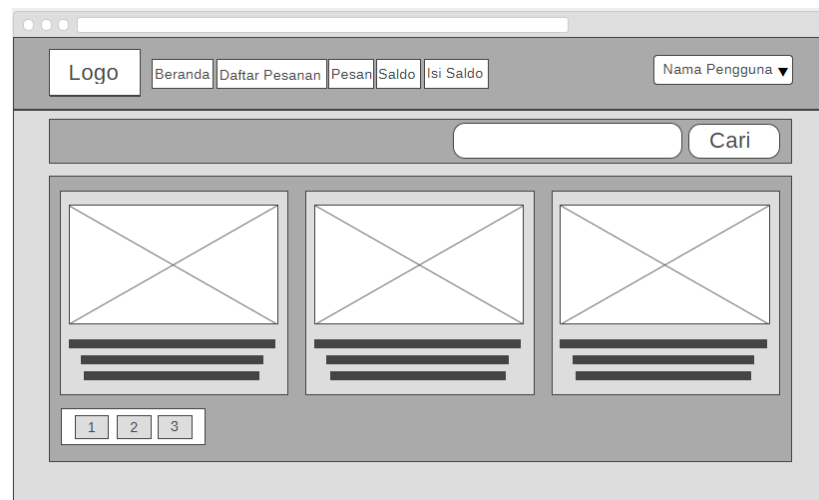
A. Halaman beranda pelanggan



Gambar 3.11 Halaman utama pelanggan

Gambar 3.11 merupakan halaman utama pelanggan setelah *login*. Pada halaman ini pelanggan dapat memilih layanan yang tersedia yaitu cetak dokumen dan cetak foto.

B. Halaman pilih percetakan



Gambar 3.12 Halaman Pilih Percetakan

Gambar 3.12 merupakan halaman untuk memilih percetakan. Halaman ini digunakan oleh pelanggan untuk memilih percetakan yang terdapat aplikasi cetakin.

C. Halaman isi formulir pesanan

Gambar 3.13 Halaman isi formulir pesanan

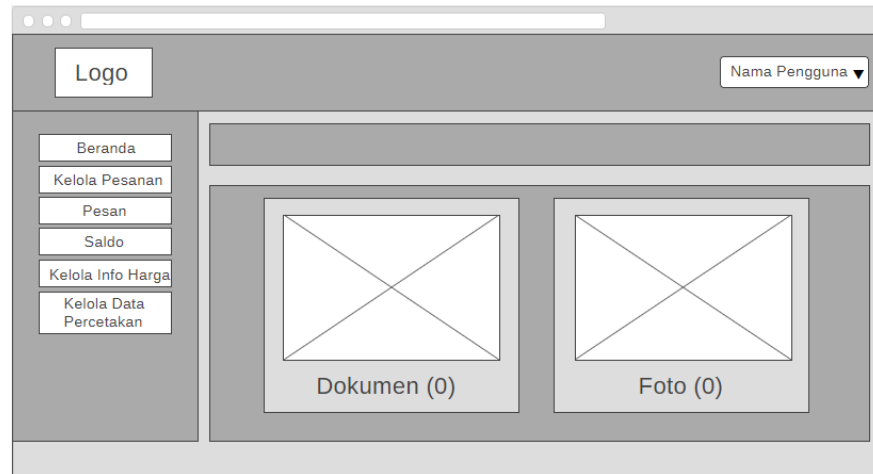
Gambar 3.13 merupakan halaman isi formulir pesanan. Halaman ini digunakan oleh pelanggan untuk mengisi detail keterangan pesanan, mengunggah *file*, hingga mengirimkan atau membatalkan pesanan.

D. Halaman daftar pesanan

Gambar 3.14 Halaman menu daftar pesanan

Gambar 3.14 merupakan halaman menu daftar pesanan. Halaman ini digunakan oleh pelanggan untuk melihat daftar pesanan yang telah dikirim.

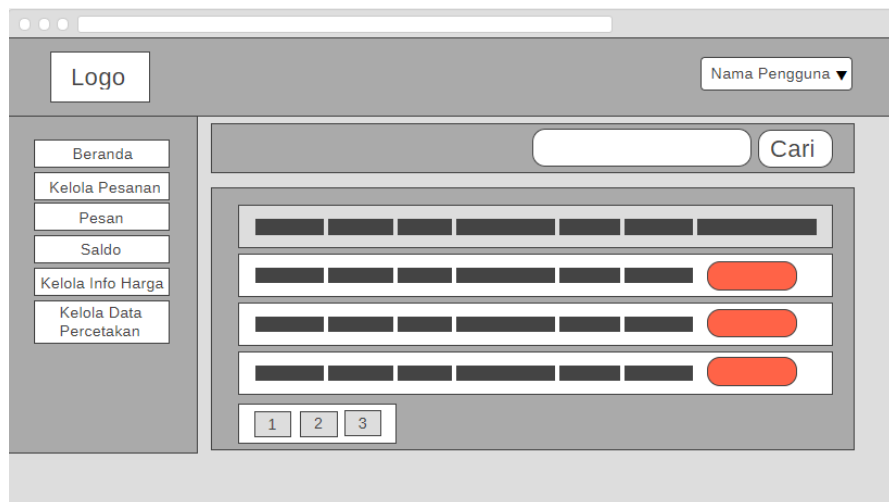
E. Halaman beranda petugas percetakan



Gambar 3.15 Halaman utama petugas percetakan

Gambar 3.15 merupakan halaman utama petugas percetakan setelah *login*. Pada halaman ini petugas percetakan dapat melihat jumlah pesanan yang diterima.

F. Halaman daftar pesanan petugas percetakan



Gambar 3.16 Halaman menu daftar pesanan petugas percetakan

Gambar 3.16 merupakan halaman menu daftar pesanan petugas percetakan setelah *login*. Pada halaman ini petugas percetakan dapat melihat daftar pesanan yang masuk untuk diproses.

...Halaman ini sengaja dikosongkan...

DAFTAR PUSTAKA

- Edy Winarno, Ali Zaki, SmithDev. 2014. *Pemrograman Web Berbasis HTML5, PHP, dan JavaScript*. Jakarta: PT Elex Media Komputindo.
- Bekti, Bintu Humairah. 2015. *Mahir Membuat Website dengan Adobe Dreamweaver CS6, CSS dan JQuery*. Yogyakarta: ANDI
- Aminudin. 2015. *Cara Efektif Belajar Framework Laravel*, Lokomedia, Yogyakarta
- Badiyanto. 2013. *Buku Pintar Framework Yii Cara Mudah Membangun Aplikasi Web PHP*, Mediakom, Yogyakarta.
- Abdullah, Rohi. 2015. *Web Programing is Easy*. Jakarta: Elek Media Komputindo
- Indrajani. 2015. *Database Design (Case Study All in One)*. Jakarta: PT Elex Media Komputindo.
- Pressman. 2015. *Rekayasa Perangkat Lunak: Pendekatan Praktisi Buku I*. Yogyakarta
- Rosa. A.S., dan Shalahuddin. M. 2014. *Rekayasa Perangkat Lunak*. Bandung: Informatika.
- Sukamto, R. A., dan Shalahudin, M. 2014. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika Bandung.
- Abdul Rouf, 2015. *Pengujian Perangkat Lunak dengan Menggunakan Metode White Box dan Black Box*, STMIK HIMSYA, Semarang

