

APLIKASI PENYEWAAN JASA PERIKLANAN
(Studi Kasus: PT. RAJASA MITRA ABADI)

LAPORAN KERJA PRAKTIK

Disusun Oleh:
Kevin Kindangen
(13013058)



PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KATOLIK DE LA SALLE
MANADO
2017

APLIKASI PENYEWAAN JASA PERIKLANAN
(Studi Kasus: PT. RAJASA MITRA ABADI)

LAPORAN KERJA PRAKTIK

Diajukan untuk memenuhi syarat mata kuliah Kerja Praktik
(IF5161)

Disusun Oleh:
Kevin Kindangen
(13013058)



PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KATOLIK DE LA SALLE
MANADO

2017

**LEMBAR PENGESAHAN
LAPORAN KERJA PRAKTIK**

Judul :
APLIKASI PENYEWAAN JASA PERIKLANAN

Studi Kasus: PT. Rajasa Mitra Abadi

Telah disetujui dan disahkan pada tanggal: 09 Agustus 2017

Oleh :

PT. RAJASA MITRA ABADI

Tanda Tangan & Cap Instansi

Anri Mailoor
Pemilik

LEMBAR PERNYATAAN

Yang bertandatangan di bawah ini :

Nama : Kevin Kindangen
NIM : 13013058
Tempat/Tanggal Lahir : Manado, 31 Maret 1996
Fakultas/Jurusan : Teknik/Teknik Informatika

Menyatakan bahwa Laporan Kerja Praktik dan atau Program berjudul **Aplikasi Penyewaan Jasa Periklanan** (Studi Kasus PT. Rajasa Mitra Abadi) yang saya buat adalah benar hasil karya saya dan bukan karya tulis orang lain, baik sebagian atau seluruhnya kecuali dalam bentuk kutipan yang telah disebutkan sumbernya.

Demikian surat pernyataan ini saya buat dengan sebenar-benarnya dan apabila pernyataan ini tidak benar maka saya bersedia menerima sanksi akademis sesuai dengan yang ditetapkan oleh Fakultas, berupa pembatalan Laporan Kerja Praktik dan hasilnya.

Manado, 7 Desember 2017

Yang Menyatakan,

Kevin Kindangen

Menyetujui,

Dosen Pembimbing I

Dosen Pembimbing II

Thomas Suwanto, S.Kom., M.Mm

Indah Yessi Kairupan, S.T., M.Sc

Mengetahui,

Ketua Program Studi

Dekan Fakultas Teknik

Thomas Suwanto, S.Kom., M.Mm

Debby Paseru, S.T., MMSI., M.Ed



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KATOLIK DE LA SALLE
MANADO**

FORM KP - 003

FORMULIR DATA UMUM PERUSAHAAN

NAMA MAHASISWA : Kevin Kindangen
NIM : 13013058

NAMA INSTANSI : PT. RAJASA MITRA ABADI
ALAMAT INSTANSI : Jl. A.A Maramis, No. 170 Manado

DIDIRIKAN TAHUN : 2007
IZIN USAHA : 204/1332/79/IWK/DPMPTSP/111/2017
BIDANG BISNIS : Jasa Kontraktor, Advertising, dan Rental Alat Berat

JUMLAH PEGAWAI :
PEMILIK : Anri Mailoor

WAKIL PERUSAHAAN
Tanggal : 15 Desember 2017
Nama : Hasanuddin
Jabatan : HRD

(Tanda tangan dan :
cap perusahaan)



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KATOLIK DE LA SALLE
MANADO**

FORM KP - 004

FORMULIR PENILAIAN KEMAJUAN KERJA PRAKTIK

A. UMUM

Nama Mahasiswa : Kevin Kindangen
NIM Mahasiswa : 13013058
Program Studi : Teknik Informatika
Dosen Pembimbing Akademik : Thomas Suwanto, S.Kom., M.Mm.
Topik/Rencana Bidang : Aplikasi Penyewaan Jasa Periklanan
Pembimbing 1 : Thomas Suwanto, S.Kom., M.Mm.
Pembimbing 2 : Indah Yessi Kairupan, S.T., M.Sc
Terhitung Mulai : 27 Agustus 2017
Target Selesai : 24 November 2017

B. KEGIATAN PELAKSANAAN KERJA PRAKTIK

No.	Tanggal	Jenis Kegiatan	Paraf
1.	28 Agustus	Sidang Kerja Praktik Proposal (DP 1)	
2.	28 Agustus	Sidang Kerja Praktik Proposal (DP 2)	
3.	4 September	Konsultasi Bab 1 dan 2 Revisi Bab 1 (DP 1)	
4.	4 September	Konsultasi Bab 1 dan 2 Revisi Bab 1 (DP 2)	
5.	11 September	Konsultasi Bab 3 (DP 1)	
6.	11 September	Konsultasi Bab 3 (DP 2)	
7.	18 September	Konsultasi Basis Data (DP 1)	
8.	18 September	Konsultasi Basis Data (DP 2)	
9.	25 September	Konsultasi Aplikasi, Revisi Bab 3 (DP 1)	
10.	25 September	Konsultasi Aplikasi, Revisi Bab 3 (DP 2)	
11.	2 Oktober	Konsultasi Aplikasi, Bab 4 (DP 1)	
12.	2 Oktober	Konsultasi Aplikasi, Bab 4 (DP 2)	
13.	25 Oktober	Konsultasi Aplikasi, Revisi Bab 4 (DP 1)	
14.	25 Oktober	Konsultasi Aplikasi, Revisi Bab 4 (DP 2)	
15.	7 November	Konsultasi Bab 5, Revisi Aplikasi (DP 1)	
16.	7 November	Konsultasi Bab 5, Revisi Aplikasi (DP 2)	

Manado, 7 Desember 2017

Dosen Pembimbing KP

(Thomas Suwanto, S.Kom., M.Mm.)



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KATOLIK DE LA SALLE
MANADO**

FORM KP - 005

FORMULIR PENILAIAN PELAKSANAAN KERJA PRAKTIK

Mohon diisi dan dicek seperlunya,

NAMA MAHASISWA : Kevin Kindangen
NIM : 13013058
NAMA INSTANSI : PT. Rajasa Mitra Abadi
ALAMAT INSTANSI : Jl. A. A. Maramis, No. 170 Manado
TGL KERJA PRAKTIK : 7 Agustus 2017-16 September 2017
TOPIK YANG DIBAHAS : Aplikasi Penyewaan Jasa Periklanan

Nilai	=	50	60	70	80	90	100
Sikap	=	50	60	70	80	90	100
Kerajinan	=	50	60	70	80	90	100
Prestasi	=	50	60	70	80	90	100

KOMENTAR/SARAN

NILAI RATA-RATA :
TANGGAL :
NAMA PENILAI : Hasanuddin
JABATAN : HRD

(Tanda tangan dan
cap perusahaan) :

KATA PENGANTAR

Pertama-tama penulis menyampaikan puji dan syukur kepada Tuhan Yang Maha Esa, karena atas berkat dan tuntunan-Nya sehingga bisa menyelesaikan laporan kerja praktik ini dengan baik. Pembuatan laporan Kerja Praktik ini untuk memenuhi persyaratan mata kuliah Kerja Praktik yang merupakan persyaratan akademik untuk Program Studi Teknik Informatika Universitas Katolik De La Salle Manado.

Laporan kerja praktik ini membahas mengenai Aplikasi Penyewaan Jasa Periklanan yang diharapkan dapat membantu PT. Rajasa Mitra Abadi dalam pengolahan pemesanan periklanan.

Dalam menyusun laporan kerja praktik ini telah melibatkan banyak pihak yang turut membantu, mendukung dan memfasilitasi penulis dalam menyelesaikan aplikasi dan laporan kerja praktik. Penulis mengucapkan terima kasih atas doa dan dukungan kepada pihak-pihak tersebut yang diantaranya sebagai berikut:

1. Prof. Dr. Johanis Ohoitumur selaku Rektor Universitas Katolik De La Salle Manado.
2. Ibu Debby Paseru, ST., MMSI., M.Ed selaku Dekan Fakultas Teknik Universitas Katolik De La Salle Manado.
3. Bapak Thomas Suwanto, S.Kom., M.Mm selaku Ketua Program Studi Teknik Informatika Universitas Katolik De La Salle Manado dan Dosen Pembimbing kerja praktik yang selalu memberikan motivasi dan arahan untuk penulis.
4. Ibu Indah Yessi Kairupan, S.T., M.Sc selaku Co-Pembimbing kerja praktik yang selalu memberikan bimbingan selama melakukan kerja praktik.
5. PT. Rajasa Mitra Abadi, pimpinan beserta staf yang telah menerima dalam melakukan Kerja Praktik dan membantu dalam memberikan dukungan untuk penulis.
6. Orang tua yang terus memberikan dukungan, doa, semangat dan motivasi dalam menyelesaikan aplikasi dan laporan Kerja Praktik ini.
7. Teman-teman angkatan 2013 yang membantu dan memberikan dukungan dalam penyelesaian laporan tugas akhir ini.
8. Seluruh pihak yang tidak dapat disebutkan satu per satu.

Laporan kerja praktik ini diharapkan dapat memberikan manfaat bagi para pembaca. Penulis meminta maaf apabila dalam penulisan laporan kerja praktik ini masih terdapat beberapa kesalahan dalam penulisan kata.

Manado, Desember 2017

Penulis

DAFTAR ISI

Judul Laporan.....	i
Lembar Pengesahan	ii
Lembar Pernyataan.....	iii
FORM KP – 003: Formulir Data Umum Perusahaan	iv
FORM KP – 004: Formulir Penilaian Kemajuan Kerja Praktik	v
FORM KP – 005: Formulir Penilaian Pelaksanaan Kerja Praktik	vi
Kata Pengantar	vii
Daftar Isi.....	viii
Daftar Tabel	ix
Daftar Gambar.....	x
Daftar Lampiran	xi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	2
1.2 Rumusan Masalah	2
1.3 Tujuan Kerja Praktik	2
1.4 Manfaat Kerja Praktik	2
1.5 Ruang Lingkup dan Batasan Masalah	2
1.6 Sistematika Penulisan	4
BAB II DATA UMUM PERUSAHAAN	6
2.1 Sejarah Umum Perusahaan	6
2.1.1 Tempat dan Kedudukan.....	6
2.1.2 Visi dan Misi	6
2.2 Lingkup Pekerjaan Perusahaan	7
2.2.1 Tugas dan Fungsi.....	7
2.2.2 Struktur Organisasi Perusahaan.....	8
2.3 Lingkup Pekerjaan yang Dilakukan.....	8
BAB III LANDASAN TEORI.....	9
3.1 Teori Pendukung.....	9
3.1.1 Aplikasi	9
3.1.2 Basis Data.....	9
3.1.3 <i>User Interface</i>	10
3.1.4 CSS.....	10
3.1.5 Bootstrap	11
3.1.6 XAMPP	11
3.1.7 <i>Code Igniter</i>	11
3.2 Langkah Pemecahan Masalah.....	12
3.2.1 Metodologi Pengembangan Sistem.....	12
3.2.2 Tahapan Metodologi RAD	12
3.2.3 <i>Unified Modeling Language 2.5</i>	13
3.3 Prosedur Pengumpulan Data.....	19

3.3.1 Wawancara	19
3.3.2 Cara mengolah data	19
3.3.3 Studi Pustaka	20
BAB IV PEMBAHASAN.....	21
4.1 <i>Requirements Planning</i>	21
4.1.1 Persyaratan Awal.....	21
4.1.2 Mengidentifikasi dan Menganalisis Masalah	23
4.1.3 Ruang Lingkup Proyek	25
4.2 <i>User Design</i>	28
4.2.1 Memodelkan Aplikasi Yang Akan Dibuat	28
4.3 Konstruksi.....	40
4.3.1 Pengembangan Aplikasi	40
4.3.2 Melakukan Pemrograman.....	45
4.4 <i>Testing and Turn Over</i>	50
4.4.1 Pengujian	50
BAB V KESIMPULAN DAN SARAN.....	74
5.1 Kesimpulan	74
5.2 Saran	74
DAFTAR PUSTAKA	75
LAMPIRAN	

DAFTAR TABEL

Tabel 3.1	<i>Use Case Diagram</i>	14
Tabel 3.2	<i>Class Diagram</i>	16
Tabel 3.3	<i>Activity Diagram</i>	18
Tabel 4.1	Mengidentifikasi dan menganalisis masalah.....	24
Tabel 4.2	Identifikasi Prioritas Persyaratan Pengguna.....	24
Tabel 4.3	Spesifikasi Persyaratan <i>Admin</i>	25
Tabel 4.4	Spesifikasi Persyaratan Pengguna.....	26
Tabel 4.5	<i>Use Case</i> Pelanggan datang dan mengantri	29
Tabel 4.6	<i>Use Case</i> Melakukan Pemesanan.....	29
Tabel 4.7	<i>Use Case</i> Pembahasan Harga.....	30
Tabel 4.8	<i>Use Case</i> Melakukan Pembayaran Pemesanan.....	30
Tabel 4.9	<i>Use Case</i> Melakukan <i>Login</i>	31
Tabel 4.10	<i>Use Case</i> Melakukan Pemesanan.....	31
Tabel 4.11	<i>Use Case</i> Melakukan Pendaftaran.....	32
Tabel 4.12	<i>Use Case List Order</i>	32
Tabel 4.13	<i>Use Case</i> Melihat <i>History</i>	33
Tabel 4.14	<i>Use Case</i> Melakukan Pembayaran.....	33
Tabel 4.15	<i>Use Case</i> Konfirmasi Pembayaran.....	33
Tabel 4.16	Kasus Pengujian	51
Tabel 4.17	<i>Test Plan</i>	52
Tabel 4.18	<i>White Box Testing</i>	54
Tabel 4.19	<i>Black Box Testing</i>	63

DAFTAR GAMBAR

Gambar 2.1	Struktur Organisasi Divisi Advertising PT. Rajasa Mitra Abadi ..	8
Gambar 4.1	<i>Use Case Diagram</i> Sistem Lama	29
Gambar 4.2	<i>Use Case Diagram</i> Sistem Baru	31
Gambar 4.3	<i>Activity Diagram</i> Pelanggan	34
Gambar 4.4	<i>Activity Diagram</i> Admin	35
Gambar 4.5	<i>Class Diagram</i>	36
Gambar 4.6	Tampilan <i>Login</i>	37
Gambar 4.7	Tampilan <i>Register</i> Pelanggan	37
Gambar 4.8	Tampilan <i>Home</i> Pelanggan	38
Gambar 4.9	Tampilan <i>Order</i>	38
Gambar 4.10	Tampilan Pembayaran Pelanggan	39
Gambar 4.11	Tampilan Konfirmasi Pembayaran	39
Gambar 4.12	Tampilan <i>Login</i>	40
Gambar 4.13	Tampilan <i>Register</i>	41
Gambar 4.14	Tampilan <i>Home</i>	41
Gambar 4.15	Tabel <i>Feedback</i>	42
Gambar 4.16	Tabel <i>History</i>	42
Gambar 4.17	Tabel Pelanggan	43
Gambar 4.18	Tabel Pembayaran	43
Gambar 4.19	Tabel Pengguna	43
Gambar 4.20	Tabel Pesanan	44
Gambar 4.21	Tabel Tagihan	44
Gambar 4.22	Data dari tabel Pengguna	70
Gambar 4.23	Data dari tabel Pelanggan	70
Gambar 4.24	Data dari tabel Pesanan	70
Gambar 4.25	Data dari tabel Tagihan	71
Gambar 4.26	Data dari tabel Pembayaran	71
Gambar 4.27	Data dari tabel <i>Feedback</i>	71
Gambar 4.28	Data dari tabel <i>History</i>	71
Gambar 4.29	Aplikasi Penyewaan Jasa Periklanan pada Chrome 80.0.3987.132	72
Gambar 4.30	Aplikasi Penyewaan Jasa Periklanan pada Firefox 18.5.0.0	72
Gambar 4.31	Aplikasi Penyewaan Jasa Periklanan pada Opera 67.0.3575.53 .	73

DAFTAR LAMPIRAN

Lampiran A Hasil Wawancara	A-1
Lampiran B <i>User Acceptance Test</i>	B-1

BAB I

PENDAHULUAN

1.1. Latar Belakang

Pada zaman sekarang, teknologi informasi sangat berdampak dalam kehidupan sehari-hari. Teknologi informasi yang dikembangkan dengan kebutuhan manusia, agar dapat memudahkan, memperoleh dan memproses informasi-informasi yang bernilai, karena pada zaman sekarang ini teknologi informasi memiliki peran penting dalam beberapa bidang seperti bisnis, pendidikan, dan lain sebagainya.

PT. Rajasa Mitra Abadi bergerak di bidang jasa konstruksi dan jasa periklanan. Perusahaan ini mempunyai dua divisi utama yaitu divisi konstruksi yang bertugas untuk melaksanakan pembangunan gedung dan bangunan sipil, sedangkan untuk tugas utama dari divisi periklanan yaitu menyediakan jasa untuk pembuatan papan iklan, *neon box*, dan huruf timbul dan sebagainya.

Perusahaan memiliki kendala dalam alur pemesanan yang dibuat, dikarenakan sebagian pelanggan lebih memilih melakukan pemesanan dengan cara langsung menghubungi pemilik perusahaan, dan alur pemesanannya menjadi tidak sesuai dengan *standard operational procedure* (SOP) yang dibuat oleh perusahaan. *Standard operational procedure* (SOP) untuk pemesanan jasa *advertising* dari perusahaan ini, yaitu melakukan pemesanan pada karyawan yang bertugas, memberikan *design* yang diinginkan (opsional), memberikan permintaan material yang ingin digunakan dalam pemesanan (opsional), dan memberikan alamat dan kontak yang dapat dihubungi. Setelah barang telah selesai dibuat maka karyawan dari perusahaan akan menghubungi pelanggan dan akan mengonfirmasi alamat yang nantinya akan diantarkan. Sebaliknya para pelanggan lebih memilih melakukan pemesanan dengan cara langsung menghubungi pemilik perusahaan.

Masalah ini mengganggu proses pekerjaan perusahaan karena pemesanannya langsung kepada pemilik perusahaan dan bukan pada karyawan yang bertugas. Selain masalah pada pemesanan perusahaan ini juga memiliki kendala dalam hal mengumpulkan informasi tentang kualitas dari produk dan pelayanan yang mereka sediakan sehingga pelanggan tidak mempunyai tempat untuk memberikan kritik dan saran atas jasa, produk, atau pelayanan yang

disediakan di PT. Rajasa Mitra Abadi. Perusahaan juga tidak mempunyai laporan bulanan dari hasil pemesanan yang didapatkan.

Dari penjelasan di atas maka penulis akan membuat aplikasi penyewaan jasa periklanan yang diharapkan dapat mempermudah perusahaan dalam mengatur kembali alur pemesanan perusahaan untuk mengantisipasi kesalahan dalam proses pemesanan, dan memberikan tempat untuk pelanggan memberikan kritik atau masukan. Zaman sekarang ini hampir semua orang sudah mengenal internet dan bisa mengakses internet, itulah alasan utama mengapa perusahaan memilih aplikasi ini dibuat pada *platform web*.

1.2. Rumusan Masalah

Bagaimana merancang aplikasi penyewaan jasa periklanan?

1.3. Tujuan Kerja Praktik

Tujuan dari kerja praktik ini untuk membangun aplikasi penyewaan jasa periklanan sehingga dapat mengembalikan alur pemesanan yang ada pada jasa *advertising* dan membangun hubungan baik antara perusahaan dan pelanggan.

1.4. Manfaat Kerja Praktik

Manfaat Kerja praktik bagi pihak perusahaan PT. Rajasa Mitra Abadi sebagai berikut :

1. Membantu *operator* yang bertugas menerima pemesanan.
2. Membantu menjalin hubungan yang baik antara *customer* dan perusahaan.
3. Membantu pelanggan dalam pemesanan.

Bagi Penulis :

1. Penulis mendapatkan pengalaman dalam membangun sebuah aplikasi berbasis web.
2. Penulis mendapatkan gambaran dalam lingkungan kerja.

1.5. Ruang Lingkup dan Batasan Masalah

Ruang lingkup dalam kerja praktik ini adalah :

1. Mencari dan mengumpulkan data yang berhubungan dengan pembuatan aplikasi penyewaan jasa periklanan.
2. Menganalisis masalah pada perusahaan PT. Rajasa Mitra Abadi.
3. Merancang dan membuat *database*, *interface*, dan proses untuk aplikasi yang akan dibangun
4. Melakukan pemrograman dan pengujian terhadap aplikasi yang akan dibangun. Beberapa fitur yang disediakan antara lain :
 - a. *Admin* (karyawan)
 - a) Halaman *login* dan *logout*, berfungsi untuk mengakses tampilan *admin* pada aplikasi penyewaan jasa periklanan.
 - b) Halaman *order list* berfungsi untuk melihat daftar pemesanan yang telah dipesan oleh pelanggan.
 - c) Halaman *edit order* berfungsi untuk untuk menambah dan mengurangi pemesanan.
 - d) Halaman notifikasi berfungsi untuk memberikan pemberitahuan ketika pesanan yang dilakukan oleh pelanggan sudah tidak dapat dibatalkan atau sudah diselesaikan.
 - e) Halaman *feedback* berfungsi untuk melihat kritik dan saran yang diberikan oleh pelanggan.
 - f) Halaman laporan bulanan berfungsi untuk melihat keuntungan tiap bulan.
 - g) Halaman tambah *admin* berfungsi untuk menambahkan *admin*.
 - h) Halaman pembayaran berfungsi untuk mengonfirmasi pembayaran yang dilakukan pelanggan.
 - i) Halaman *upload* berfungsi untuk melihat desain yang telah diunggah pelanggan.
 - b. *User* (Pelanggan)
 - a) Halaman *login* dan *logout*, berfungsi untuk mengakses tampilan *user* pada aplikasi penyewaan jasa periklanan.
 - b) Halaman *register* berfungsi untuk melakukan pendaftaran.

- c) Halaman *history* berfungsi untuk melihat riwayat pemesanan yang telah dilakukan.
- d) Halaman *order* berfungsi untuk melakukan pemesanan
- e) Halaman *order list* berfungsi untuk melihat daftar pemesanan.
- f) Halaman *cancel order* berfungsi untuk melakukan pembatalan pemesanan tetapi jika pesanan sudah memasuki tahap pembuatan maka pelanggan sudah tidak bisa membatalkan pesannya.
- g) Halaman *edit order* berfungsi untuk menambah dan mengurangi pemesanan.
- h) Halaman notifikasi berfungsi untuk menerima pemberitahuan bahwa pesanan sudah memasuki tahap pembuatan atau sudah diselesaikan dan sudah bisa di ambil.
- i) Halaman *feedback* berfungsi untuk memberikan kritik dan saran kepada perusahaan.
- j) Halaman pembayaran berfungsi untuk melakukan pembayaran.
- k) Halaman *tracking queue* berfungsi untuk melihat apabila barang sudah diproses atau belum. Jika barang sudah diproses maka pelanggan tidak dapat membatalkan pemesanan barang.
- l) Halaman *upload* berfungsi untuk mengunggah desain yang akan digunakan untuk pembuatan barang, juga bukti dari pembayaran yang sudah dilakukan.

Adapun batasan masalah dalam kerja praktik ini adalah :

- 1) Sistem ini hanya dikembangkan sampai pada tahap pengujian di tempat kerja praktik dan tidak membahas sampai pada tahap *maintenance*.
- 2) Tipe *file* yang diunggah hanya format JPEG, dan PNG dengan ukuran maksimal 10 *megabyte* (MB).

1.6. Sistematika Penulisan

Secara penulisan, laporan kerja praktik ini dibagi dalam beberapa bab yang diatur secara terstruktur dengan pembahasan tiap bab, yaitu :

- 1) BAB I PENDAHULUAN, dalam bab ini diuraikan tentang latar belakang, rumusan masalah, tujuan kerja praktik, manfaat kerja praktik, batasan masalah dan sistematika penulisan.
- 2) BAB II DATA UMUM PERUSAHAAN, dalam bab ini diuraikan tentang sejarah PT. Rajasa Mitra Abadi, lingkup kerja perusahaan, dan lingkup kerja yang dilakukan.
- 3) BAB III LANDASAN TEORI, dalam bab ini diuraikan tentang teori pendukung, metodologi pengembangan sistem, dan prosedur pengumpulan dan pengolahan data.
- 4) BAB IV PEMBAHASAN, berisi tentang langkah-langkah metodologi pengembangan sistem.
- 5) BAB V KESIMPULAN DAN SARAN, dalam bab ini berisi kesimpulan yang merupakan hasil dari kerja praktik dan saran yang dapat membantu dalam pengembangan aplikasi ke depannya.

BAB II

DATA UMUM PERUSAHAAN

2.1 Sejarah Umum Perusahaan

PT. Rajasa Mitra Abadi didirikan tahun 2000 dan bergerak di bidang bengkel mobil *body repair*, *advertising*, dan rental alat berat, pada tahun 2007 PT. Rajasa Mitra Abadi didirikan dihadapan notaris Winar, SH, lalu mengalami perubahan pertama dihadapan Notaris Kiki Hertanto, SH, dengan nomor 9 tanggal 10 Agustus 2010.

Pada tahun 2000 PT. Rajasa Mitra Abadi hanya bergerak di bidang *body repair* untuk kendaraan roda empat dan belum memiliki divisi apapun sehingga pekerjaan utama dari perusahaan ini hanya menyediakan jasa *body repair* untuk kendaraan roda empat saja.

Pada tahun 2000-2007 PT. Rajasa Mitra Abadi menambahkan divisi yaitu divisi *advertising*, divisi konstruksi. Divisi *advertising* bergerak dalam penyediaan jasa pembuatan papan iklan, neon box, dan huruf timbul. Divisi konstruksi bergerak dalam pembangunan gedung dan bangunan sipil juga menyediakan jasa dalam penyewaan alat berat [1].

2.1.1 Tempat dan Kedudukan

PT. Rajasa Mitra Abadi bertempat di Jln. A.A Maramis No. 170 Manado, Sulawesi Utara.

2.1.2 Visi dan Misi

Berikut ini adalah visi dan misi dari perusahaan PT. Rajasa Mitra Abadi.

1. Visi

“Menjadi perusahaan swasta nasional terdepan di industri jasa konstruksi, berkembang secara berkesinambungan, memberikan kesejahteraan kepada karyawan, pengurus, pemegang saham dan stake holder lainnya melalui komitmen terhadap CSR (*corporate social responsibility*) dan tata kelola perusahaan yang baik (*good corporate governance*).”

2. Misi

- a) Pemberdayaan maksimal dari lima pilar usaha konstruksi: Pemasaran, Operasional, Keuangan, Sumber Daya Manusia dan Informasi.
- b) Menyamakan persepsi di antara manajemen untuk mempertahankan nilai-nilai perusahaan dan mencapai tujuan bersama.
- c) Pelatihan dan rekrutmen sumber daya manusia yang tepat, untuk menghasilkan tenaga kerja yang kompeten, loyal, berdedikasi dan bersemangat tinggi.
- d) Peningkatan kompetisi di era globalisasi lewat kerja sama dengan perusahaan kontraktor nasional maupun internasional.

2.2 Lingkup Pekerjaan Perusahaan

Bagian ini menjelaskan tentang salah satu divisi pada PT. Rajasa Mitra Abadi, yaitu. Divisi *advertising*, divisi ini menyediakan pemesanan jasa *advertising* berupa pembuatan *billboard*, *neon box*, dan huruf timbul.

2.2.1 Tugas dan Fungsi

Berikut ini adalah tugas dan fungsi pada PT. Rajasa Mitra Abadi:

a. Tugas

Divisi *advertising* mempunyai tugas utama yaitu memberikan pelayanan dalam pemesanan jasa *advertising* juga melakukan pembuatan hasil pemesanan.

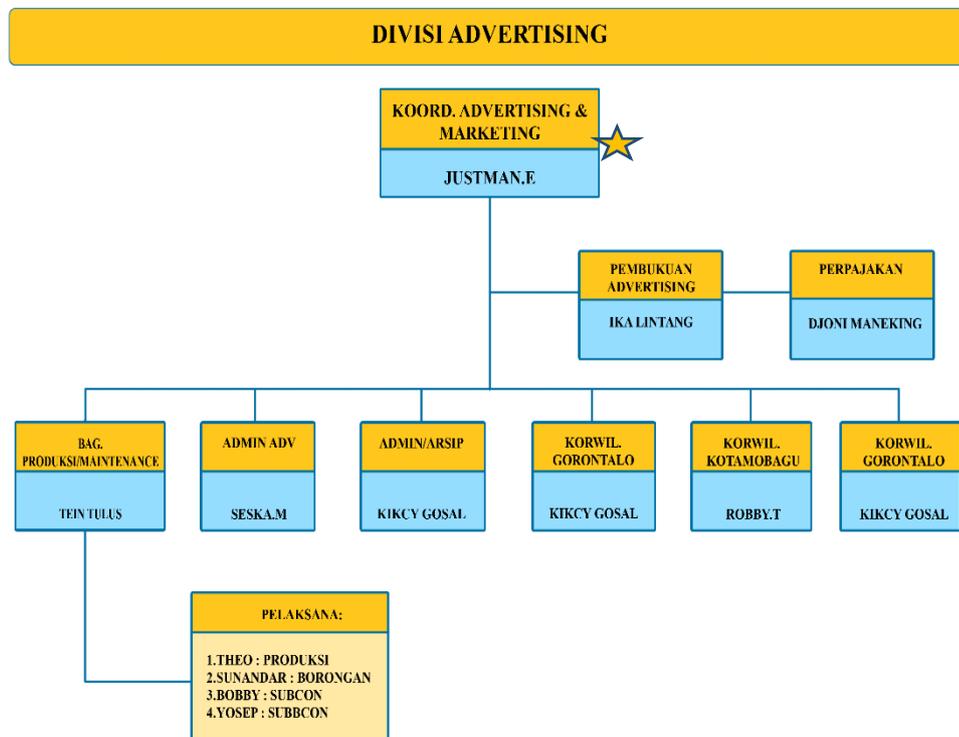
b. Fungsi

Untuk menyelenggarakan tugas sebagaimana yang dimaksud di atas, divisi *advertising* mempunyai fungsi:

1. Memberikan pelayanan jasa *advertising*.
2. Melakukan pembuatan hasil pemesanan.

2.2.2 Struktur Organisasi Perusahaan

Berikut ini adalah struktur Organisasi Perusahaan dapat dilihat pada Gambar 2.1.



Gambar 2.1 Struktur Organisasi Divisi *Advertising* PT. Rajasa Mitra Abadi [1]

2.3 Lingkup Pekerjaan yang Dilakukan

Selama melakukan kerja praktik di PT. Rajasa Mitra Abadi, penulis ditempatkan di divisi *advertising*. Penulis diberi tugas membangun sebuah aplikasi aplikasi penyewaan jasa periklanan, agar dapat membuat aplikasi yang dibutuhkan, maka penulis berusaha menganalisis kebutuhan dan kesulitan yang dihadapi perusahaan ini. Selain itu, penulis juga mengumpulkan data yang akan digunakan dalam pembuatan aplikasi ini. Penulis juga ikut membantu kegiatan yang ada di perusahaan sehingga penulis dapat mengumpulkan data yang diperlukan dalam pembuatan aplikasi dan laporan kerja praktik.

BAB III

LANDASAN TEORI

3.1 Teori Pendukung

Dalam proses pembangunan aplikasi penyewaan jasa periklanan, dilakukan pengumpulan data yang dibutuhkan dalam mendukung dalam pembangunan aplikasi yang didapatkan dari berbagai sumber, yaitu artikel *online*, buku, *e-book*, dan jurnal.

3.1.1 Aplikasi

Aplikasi atau perangkat lunak adalah program yang dibuat untuk menyelesaikan masalah-masalah khusus yang dihadapi pengguna dengan menggunakan kemampuan dari komputer [2].

Aplikasi web adalah sebuah program yang disimpan di *server* dan dikirim melalui internet dan diakses melalui *browser*, dari pengertian di atas dapat disimpulkan aplikasi web merupakan aplikasi yang diakses menggunakan *web browser* melalui jaringan internet atau intranet.

Aplikasi web juga biasa dibangun menggunakan bahasa pemrograman yang mendukung perangkat lunak berbasis web seperti HTML, PHP, CSS, JavaScript, Java dan bahasa pemrograman lainnya [3].

3.1.2 Basis Data

Basis Data terdiri atas 2 kata, yaitu basis dan data. Basis kurang lebih dapat diartikan sebagai markas atau gedung, tempat berkumpul [4]. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia(pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang mewujudkan dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya Sebagai satu kesatuan istilah, basis data (*Database*) sendiri dapat didefinisikan dalam sejumlah sudut pandang [4] seperti:

- a) Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.

- b) Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (*redundansi*) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
- c) Kumpulan *file*/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

3.1.3 User Interface

User Interface adalah *input* dan *output* yang langsung melibatkan sistem pengguna akhir. Antarmuka pengguna dapat digunakan langsung oleh pengguna internal atau eksternal sistem. Desain dari *user interface* sendiri mempunyai macam-macam model tergantung pada faktor-faktor seperti tujuan antarmuka, karakteristik pengguna, dan karakteristik perangkat *interface* tertentu. Sebagai contoh, meskipun semua antarmuka pengguna harus dirancang untuk kemudahan maksimal penggunaan, ada beberapa pertimbangan lainnya, seperti efisiensi operasional, yang mungkin penting bagi pengguna internal yang dapat dilatih untuk menggunakan *interface* tertentu, dikombinasikan dan dioptimalkan untuk perangkat keras tertentu (misalnya, *keyboard*, *mouse*, dan layar resolusi tinggi besar) [5].

3.1.4 CSS

CSS atau *cascading style sheets*, memuat aturan-aturan gaya yang memberitahu *browser* anda bagaimana menyajikan sebuah dokumen. Aturan tersebut memuat dua bagian selektor elemen HTML yang dicoba, dan blok deklarasi properti-properti dan nilai-nilai yang menjelaskan gaya untuk selektor [6].

CSS adalah kumpulan kode program yang digunakan untuk mendesain atau mempercantik tampilan halaman HTML. Dengan CSS kita bisa mengubah desain dari text, warna, gambar dan latar belakang dari (hampir) semua kode tag HTML [7].

3.1.5 *Bootstrap*

Bootstrap sebuah alat bantu untuk membuat sebuah tampilan halaman *website* yang dapat mempercepat pekerjaan seorang pengembang *website* ataupun pendesain halaman *website*. *Website* yang dibuat dengan alat bantu dapat mengubah tampilan halaman *website* sesuai dengan kebutuhan, tampilan *web* yang dibuat *bootstrap* akan menyesuaikan ukuran layar dari browser yang kita gunakan, baik di *desktop*, tablet ataupun *mobile device* [8].

3.1.6 XAMPP

XAMPP merupakan sebuah perangkat lunak gratis sehingga bebas digunakan yang berfungsi sebagai *server* yang berdiri sendiri atau biasa dikenal sebagai *localhost*, XAMPP terdiri dari apache HTTP *server*, MySQL *database* dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. XAMPP dikembangkan oleh perusahaan apache *friends* yang biasanya digunakan untuk simulasi pengembangan *website* [9].

3.1.7 *Code Igniter*

Code igniter adalah salah satu *framework* PHP bahkan *framework* PHP yang paling banyak digunakan saat ini karena di dalamnya terdapat lengkap aplikasi *web* di mana fitur-fitur tersebut ada pada *code igniter*. [10].

Code igniter adalah salah satu *framework* PHP yang bersifat *open source* dan menggunakan metode MVC (*model, view, controller*), *code igniter* bersifat *free* atau tidak berbayar jika anda ingin menggunakannya. *Framework code igniter* dibuat dengan tujuan sama seperti *framework* lainnya yaitu untuk memudahkan *developer* atau *programmer* dalam membangun sebuah aplikasi berbasis *web* tanpa harus membuatnya dari awal [11].

3.2 Langkah Pemecahan Masalah

Membangun aplikasi penyewaan jasa periklanan dibutuhkan beberapa tahap dan metode pendukung, bagian ini akan menjelaskan metode apa yang akan digunakan dalam membangun aplikasi ini.

3.2.1 Metodologi Pengembangan Sistem

Pada penelitian ini, penulis akan membangun aplikasi sesuai tahapan dari metodologi *Rapid Application Development* (RAD), *Rapid Application Development* (RAD) adalah metodologi yang bertujuan untuk menyediakan pengembangan yang jauh lebih cepat dan mendapatkan hasil yang baik. *Rapid Application Development* (RAD) merupakan gabungan dari bermacam-macam teknik terstruktur dengan teknik *prototyping* dan teknik pengembangan *join application* untuk mempercepat pengembangan sistem/aplikasi [12].

3.2.2 Tahapan Metodologi RAD

Perancangan aplikasi dengan menggunakan metodologi RAD memiliki empat tahapan [12], yaitu:

RAD Fase 1: *Requirements Planning*

Membuat persyaratan awal, kendala dan mengatur ruang lingkup proyek.

RAD Fase 2: *User Design*

Tahap ini bertujuan untuk melakukan perancangan model yang mewakili semua proses.

RAD Fase 3: *Construction*

Tahap ini bertujuan untuk mengembangkan aplikasi dan berfokus pada pembangunan aplikasi (pemrograman).

RAD Fase 4: *Testing and Turn Over*

Tahap ini merupakan tahapan akhir, yang bertujuan untuk melakukan pengujian dan memberikan pelatihan kepada pengguna. RAD memiliki kelebihan dan kelemahan, dalam penggunaannya disesuaikan dengan pengembangan proyek.

A. Kelebihan RAD

1. Untuk digunakan secara *stand alone*.
2. Digunakan untuk memperluas SDLC.
3. *Prototype* melibatkan pengguna dalam analisis dan desain.
4. Punya kemampuan menangkap kebutuhan secara konkrit daripada secara abstrak.

B. Kelemahan RAD

1. Untuk pembuatan sistem perangkat lunak dengan skala besar akan memerlukan sumber daya manusia yang cukup besar untuk membentuk tim-tim yang mengembangkan komponen-komponen.
2. Jika tidak ada persetujuan untuk mengembangkan perangkat lunak secara dengan cepat (*rapid*) maka proyek dengan model ini akan gagal karena hanya akan bingung mendefinisikan kebutuhan pelanggan (*customer*) atau *user*.
3. Jika sistem perangkat lunak yang akan dibuat tidak bisa dimodulkan (dibagi-bagi menjadi beberapa komponen) maka model RAD tidak dapat digunakan untuk membuat sistem perangkat lunak ini karena terlalu banyak campur tangan antar tim.

Model RAD tidak cocok digunakan untuk sistem perangkat lunak yang memiliki resiko teknis sangat tinggi, misalnya menggunakan teknologi baru yang belum banyak dikenal dan dikuasai pengembang [13].

3.2.3 Unified Modeling Language 2.5

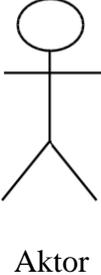
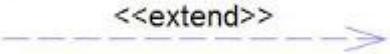
Salah satu pemodelan yang saat ini paling banyak digunakan adalah *Unified Modeling Language* (UML). UML adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan dari sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek, Dalam perancangan aplikasi ini dibutuhkan *use case diagram*, *class diagram*, *activity diagram* [14].

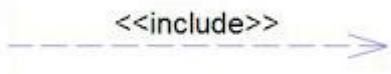
1. Use Case Diagram

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya *login* ke sistem, membuat sebuah daftar belanja, dan sebagainya. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-

pekerjaan tertentu. Berikut penjelasan simbol-simbol *use case diagram* yang dapat dilihat pada Tabel 3.1.

Tabel 3.1 Use Case Diagram [14]

Nama	Simbol	Deskripsi
<i>Use case</i>		<p><i>Use case</i> adalah fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau <i>actor</i>.</p>
Aktor/ <i>Actor</i>		<p>Aktor adalah orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat, jadi meskipun simbol dari aktor ialah gambar orang, tapi aktor belum tentu merupakan orang.</p>
Asosiasi / <i>association</i>		<p>Asosiasi adalah komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case diagram</i> atau <i>use case</i> yang memiliki interaksi dengan aktor.</p>
Ekstend / <i>extend</i>		<p>Relasi <i>use case</i> tambahan ke sebuah</p>

Nama	Simbol	Deskripsi
		<p><i>use case</i> di mana <i>use case</i> yang ditambahkan dapat berdiri sendiri meski tanpa <i>use case</i> tambahan itu arah panah mengarah pada <i>use case</i> yang ditambahkan.</p>
<p><i>Include</i></p>		<p>Relasi <i>use case</i> tambahan ke sebuah <i>use case</i> di mana <i>use case</i> yang ditambahkan membutuhkan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini.</p>
<p>Generalisasi / <i>generalization</i></p>		<p>Hubungan generalisasi dan spesialisasi (umum - khusus) antara dua buah <i>use case</i> di mana fungsi yang satu merupakan fungsi yang lebih umum dari lainnya.</p>

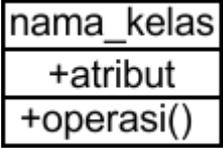
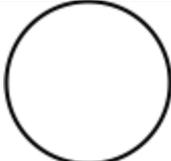
Nama	Simbol	Deskripsi
		arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya (umum).

2. Class Diagram

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).

Atribut dan metoda dapat memiliki salah satu sifat berikut: *private* tidak dapat dipanggil dari luar *class* yang bersangkutan, *protected* hanya dapat dipanggil oleh *class* yang bersangkutan dan anak-anak yang mewarisinya dan, *public* dapat dipanggil oleh siapa saja. Berikut penjelasan simbol-simbol *class diagram* yang dapat dilihat pada Tabel 3.2.

Tabel 3.2 *Class Diagram* [14]

No	Simbol	Keterangan
1		Simbol ini adalah simbol untuk sebuah kelas pada struktur sistem. Penulisan disana tidak diperbolehkan menggunakan spasi. Simbol ini memiliki 3 susunan, yaitu kotak pertama adalah nama kelas, kedua atribut dan terakhir operasi.
2		Lingkaran ini adalah simbol untuk <i>interface</i> . Konsep yang digunakan pun sama dengan pemrograman berorientasi <i>object</i> (OOP).
3		Simbol ini sering disebut dengan simbol <i>Association</i> . Garis ini adalah garis yang

No	Simbol	Keterangan
		digunakan untuk menghubungkan atau merelasikan kelas satu dengan kelas yang lainnya dengan makna umum.
4		Simbol ini bernama Generalisasi. Generalisasi digunakan untuk menghubungkan antar kelas dengan arti umum-khusus. Jadi jika ada kelas bermakna umum dan kelas bermakna khusus dapat menggunakan simbol ini.
5		Nama dari simbol ini adalah <i>Aggregation</i> . Simbol ini adalah simbol yang menghubungkan antar kelas dengan makna untuk semua bagian. Jadi relasi ini digunakan jika kelas yang satu adalah semua bagian dari kelas yang lainnya.
6		Nama dari simbol ini adalah <i>Dependency</i> . Simbol ini adalah simbol yang menghubungkan antar kelas dengan makna untuk kebergantungan antar kelas.

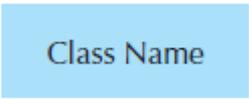
3. Activity Diagram

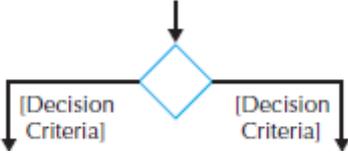
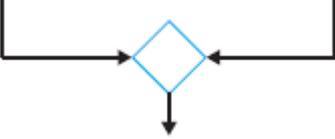
Activity diagram menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alur berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

Activity diagram merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi dipicu oleh selesainya *state* sebelumnya (*internal processing*). Oleh karena itu *activity diagram* tidak menggambarkan *behaviour internal* sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses

dan jalur-jalur aktivitas dari level atas secara umum. Berikut penjelasan *activity diagram* yang dapat dilihat pada Tabel 3.3.

Tabel 3.3 Activity Diagram [14]

Simbol	Keterangan
 <p><i>Action</i></p>	<p><i>In Action</i></p> <p><i>In Action</i> merupakan sebuah aksi yang digunakan untuk mewakili serangkaian tindakan.</p>
 <p><i>Activity</i></p>	<p><i>In Activity</i></p> <p><i>In Activity</i> ini akan digunakan untuk mewakili serangkaian tindakan.</p>
 <p><i>Object Node</i></p>	<p><i>Object Node</i></p> <p><i>Object Node</i> ini digunakan untuk mewakili suatu objek yang terhubung ke satu set arus objek. Diberi label sesuai dengan nama kelasnya.</p>
 <p><i>Control Flow</i></p>	<p><i>Control Flow</i></p> <p><i>Control Flow</i> ini akan menunjukkan urutan eksekusi.</p>
 <p><i>Object Flow</i></p>	<p><i>Object Flow</i></p> <p><i>Object Flow</i> akan menunjukkan aliran objek dari satu kegiatan (atau tindakan) untuk aktivitas lain (atau tindakan).</p>
 <p><i>Initial Node</i></p>	<p><i>Initial Node</i></p> <p><i>Initial Node</i> ini akan menggambarkan awal dari serangkaian tindakan atau kegiatan.</p>

Simbol	Keterangan
 <p data-bbox="424 499 679 533"><i>Final Activity Node</i></p>	<p data-bbox="826 322 1090 356"><i>Final Activity Node</i></p> <p data-bbox="826 376 1342 577"><i>Final Activity Node</i> digunakan untuk menghentikan semua arus kontrol dan arus objek dalam suatu kegiatan (atau tindakan).</p>
 <p data-bbox="456 790 647 824"><i>Decision Node</i></p>	<p data-bbox="826 602 1023 636"><i>Decision Node</i></p> <p data-bbox="826 656 1342 857">Digunakan untuk mewakili kondisi tes untuk memastikan bahwa aliran kontrol atau aliran objek hanya turun satu jalur.</p>
 <p data-bbox="464 1070 639 1104"><i>Merger Node</i></p>	<p data-bbox="826 878 1007 911"><i>Merger Node</i></p> <p data-bbox="826 931 1342 1133"><i>Merger Node</i> digunakan untuk membawa kembali jalur keputusan bersama yang berbeda yang diciptakan menggunakan keputusan simpul.</p>

3.3 Prosedur Pengumpulan Data

Pengumpulan data bertujuan untuk membantu tahap analisis dan tahap perancangan agar permasalahan dan kebutuhan dapat diketahui dengan jelas.

3.3.1. Wawancara

Wawancara bertujuan untuk menganalisis masalah yang sedang terjadi di PT. Rajasa Mitra Abadi dengan menyiapkan pertanyaan-pertanyaan yang berhubungan dengan masalah yang sedang terjadi. Dalam hal ini wawancara dilakukan kepada Bapak Anri Mailoor selaku pemilik dari PT. Rajasa Mitra Abadi yang dapat dilihat pada lampiran.

3.3.2. Cara Mengolah Data

Data yang terkumpul dari hasil wawancara berupa catatan secara garis besar yang kemudian disalin dalam bentuk dokumen atau format doc.

3.3.3. Studi Pustaka

Mengumpulkan data dari PT. Rajasa Mitra Abadi, data dari jurnal *online*, dan buku, *e-book*, serta artikel *online*.

BAB IV

PEMBAHASAN

Pada bab, penulis akan melakukan pembahasan tentang bagaimana cara memecahkan masalah yang terjadi di PT. Rajasa Mitra Abadi dalam penyusunan laporan kerja Praktik. Penulis menggunakan 4 tahapan pada metodologi RAD yaitu *Requirements Planning*, *User Design*, *Construction*, dan *Testing and Turn Over* dan adapula *tools* UML yang digunakan untuk membantu membuat diagram – diagram pada aplikasi ini.

4.1. Requirements Planning

Fase pertama dari RAD adalah *Requirements Planning*, yaitu mempelajari persyaratan awal, kendala dan ruang lingkup proyek pada bagian ini. Tujuan utama dari *Requirements Planning* ini yaitu mengumpulkan informasi – informasi yang berkaitan dengan persyaratan awal, kendala yang dihadapi, ruang lingkup proyek, dan alur dari aplikasi yang akan dibuat.

4.1.1 Persyaratan Awal

Persyaratan awal dari aplikasi yang akan dibangun. Hasil dari persyaratan awal ini harus dapat menjawab beberapa pertanyaan berikut :

- a. Apakah perusahaan membutuhkan aplikasi dalam hal pemesanan *billboard*, *neonbox*, dan huruf timbul ?
- b. Apakah aplikasi yang dibangun dapat membantu perusahaan dalam pemesanan *billboard*, *neonbox*, dan huruf timbul ?
- c. Apakah aplikasi yang dibangun dapat membantu perusahaan dalam melihat laporan bulanan ?
- d. Apakah aplikasi yang dibangun dapat membantu alur pemesanan perusahaan ?

Hasil dari pertanyaan di atas adalah untuk memutuskan bahwa aplikasi diperlukan atau tidak. Jika aplikasi diperlukan, maka tahap selanjutnya adalah melakukan studi kelayakan. Studi kelayakan yang dilakukan adalah sebagai berikut:

- a. Daftar spesifikasi sumber daya yang dibutuhkan ?
- b. Operasional pembuatan aplikasi.

Hasil dari *Requirements Planning* dalam pengembangan aplikasi penyewaan jasa periklanan adalah penilaian awal dari studi kelayakan.

4.1.1.1 Penilaian Awal

Untuk dapat menjawab pertanyaan dari penilaian awal aplikasi, maka perlu untuk dilakukan pengumpulan data dari PT. Rajasa Mitra Abadi agar dapat diketahui masalah yang muncul. Pengumpulan data yang dilakukan oleh penulis adalah wawancara dengan pemilik perusahaan yang berhubungan dengan pembuatan aplikasi.

4.1.1.2 Pengumpulan dan Pemrosesan Data

Berikut adalah beberapa cara yang dilakukan untuk mengumpulkan data dari PT. Rajasa Mitra Abadi

a. Penentuan Kriteria Sampel

Kriteria yang digunakan untuk pemilihan sampel yang digunakan adalah sebagai berikut:

1. Merupakan pemilik dari PT. Rajasa Mitra Abadi.
2. Memahami bagaimana cara kerja dalam bidangnya.

b. Membuat Daftar Pertanyaan

Berikut adalah daftar dari pertanyaan wawancara yang telah dilakukan pada pemilik PT. Rajasa Mitra Abadi :

1. Apa perusahaan memiliki aplikasi pemesanan *online*?
2. Apa perusahaan memiliki kesulitan dalam proses penerimaan pemesanan?
3. Bagaimana jika proses pemesanan dilakukan secara *online* menggunakan aplikasi berbasis *web*?
4. Media apa yang sering digunakan dalam proses penerimaan pemesanan?
5. Apakah pemesanan yang dilakukan pelanggan bisa dibatalkan?
6. Apa saja produk yang disediakan oleh perusahaan?

7. Bagaimana cara pelanggan melakukan pemesanan?
8. Apakah perusahaan memiliki masalah selain proses penerimaan pemesanan?
9. Tahun berapakah PT. Mira Rajasa Abadi didirikan?

c. Pembahasan Hasil Analisis Wawancara

Berdasarkan hasil wawancara, masalah yang terjadi di PT. Rajasa Mitra Abadi yaitu dalam alur pemesanan yang dibuat, dikarenakan sebagian pelanggan lebih memilih melakukan pemesanan dengan cara langsung menghubungi pemilik perusahaan, dan alur pemesanannya menjadi tidak sesuai dengan *standard operational procedure* (SOP) yang dibuat oleh perusahaan. SOP perusahaan yaitu pelanggan melakukan pemesanan pada karyawan yang bertugas, memberikan *design* yang diinginkan (opsional), memberikan permintaan material yang ingin digunakan dalam pemesanan (opsional), dan memberikan alamat dan kontak yang dapat dihubungi. Setelah barang telah selesai dibuat maka karyawan dari perusahaan akan menghubungi pelanggan dan akan mengonfirmasi alamat yang nantinya akan diantarkan. Perusahaan juga memiliki kendala dalam hal mengumpulkan informasi tentang kualitas dari produk dan pelayanan yang mereka sediakan sehingga pelanggan tidak mempunyai tempat untuk memberikan kritik atau masukan atas jasa, produk, ataupun pelayanan yang disediakan, dan perusahaan juga tidak mempunyai laporan bulanan dari hasil pemesanan yang didapatkan.

4.1.2 Mengidentifikasi dan Menganalisis Masalah

Proses pemesanan yang tidak sesuai alur yang dibuat oleh PT. Rajasa Mitra Abadi yang mengakibatkan proses pekerjaan perusahaan terganggu karena orang yang menerima pemesanan tidak sesuai dengan *job description*. Hal ini dikarenakan perusahaan ini belum mempunyai aplikasi pemesanan secara *online* yang dapat mempermudah proses pemesanan, juga mengurangi waktu dan tenaga yang digunakan. Proses mengidentifikasi dan menganalisis masalah yang ada dapat dilihat pada Tabel 4.1 yang bisa dilihat pada halaman selanjutnya.

Tabel 4.1 Mengidentifikasi dan menganalisis masalah

Proyek : Aplikasi Penyewaan Jasa Periklanan	Manajer Proyek : Kevin Kindangen
Dikerjakan oleh : Kevin Kindangen	Terakhir diperbarui oleh : Kevin Kindangen
Tanggal pengerjaan : 24 Oktober 2017	Terakhir diperbarui : 24 Oktober 2017
Masalah	Solusi yang diusulkan
Sebagian pelanggan lebih memilih melakukan proses pemesanan kepada orang yang kurang tepat <i>job description</i> -nya, sehingga mengakibatkan alur pemesanan menjadi tidak sesuai yang dibuat PT. Rajasa Mitra Abadi.	Membuat aplikasi pemesanan <i>online</i> yang dapat membantu pelanggan dan perusahaan dalam pemesanan dan penerimaan pemesanan, sehingga dapat mengatur kembali alur pemesanan.
Adanya kesulitan dalam menampung keluhan dan masukan dari pelanggan.	Melengkapi aplikasi dengan fitur <i>feedback</i> .
Adanya kesulitan dalam melakukan rekapitulasi keuntungan per bulan	Melengkapi aplikasi dengan fitur laporan bulanan.

4.1.2.1 Mengidentifikasi Prioritas Persyaratan Pengguna

Adapun beberapa prioritas persyaratan dari pengguna yang bisa dilihat pada Tabel 4.2.

Tabel 4.2 Identifikasi Prioritas Persyaratan Pengguna

No.	Persyaratan Pengguna	Prioritas
1.	Membuat sebuah aplikasi yang mudah diakses.	Penting
2.	Membuat sebuah aplikasi yang dapat mempermudah proses Pemesanan.	Penting
3.	Membuat sebuah aplikasi yang dapat membantu dalam menampung keluhan dan masukan pelanggan.	Penting
5.	Pembuatan laporan bulanan.	Cukup

No.	Persyaratan Pengguna	Prioritas
		Penting
6.	Membuat <i>interface</i> aplikasi yang baik sesuai kriteria <i>user interface</i> .	Penting

4.1.3 Ruang Lingkup Proyek

Adapun ruang lingkup dari proyek ini adalah:

1. Mencari dan mengumpulkan data yang berhubungan dengan pembuatan aplikasi ini.
2. Menganalisis masalah dan kebutuhan.
3. Merancang dan membuat basis data, *interface*, dan proses untuk aplikasi yang akan dibangun.
4. Melakukan pemrograman dan pengujian terhadap aplikasi yang akan dibangun.

4.1.3.1 Spesifikasi Persyaratan *Admin*

Tahap ini akan mendaftar kebutuhan aplikasi, dalam hal ini fitur-fitur yang ada, fungsi dan batasannya yang dapat dilihat pada Tabel 4.3.

Tabel 4.3 Spesifikasi Persyaratan *Admin*

No.	Fitur	Fungsi	Batasan
1.	<i>Login dan Logout</i>	Berfungsi untuk mengakses tampilan <i>admin</i> .	<i>Admin</i> bisa mengakses fitur ini dan memberikan batasan kepada pengguna
2.	<i>Order List</i>	Untuk melihat daftar pemesanan yang telah dipesan oleh pelanggan.	<i>Admin</i> dapat mengakses fitur ini untuk melihat daftar pesanan.
3.	<i>Edit Order</i>	Untuk untuk menambah dan mengurangi pemesanan.	Proses ini dilakukan oleh <i>admin</i> .
4.	Notifikasi	Pemberitahuan mengenai pesanan yang dilakukan oleh pelanggan.	<i>Admin</i> dapat melihat pemberitahuan yang dilakukan pelanggan dalam

No.	Fitur	Fungsi	Batasan
			memesan barang.
5.	<i>Feedback</i>	Untuk melihat kritik dan saran yang diberikan oleh pelanggan.	Admin dapat mengakses fitur ini untuk melihat kritik dan saran yang diberikan oleh pelanggan.
6.	Laporan Bulanan	Untuk melihat keuangan setiap bulan.	<i>Admin</i> yang bisa melakukan proses ini untuk melihat laporan keuangan.
7.	Tambah pengguna	Untuk menambahkan <i>admin</i> dan pelanggan	Hanya <i>admin</i> yang dapat melakukan proses ini untuk menambahkan admin dan pelanggan.
8.	Pembayaran	Untuk mengonfirmasi pembayaran yang dilakukan oleh pelanggan.	<i>Admin</i> yang dapat melakukan proses ini untuk mengonfirmasi pemesanan jika pelanggan sudah pada tahapan pembayaran dan tidak dapat dibatalkan pesanan tersebut.
9.	<i>Upload</i>	Untuk melihat desain yang telah dikirimkan oleh pelanggan.	<i>Admin</i> dapat mengakses proses ini untuk melihat desain yang akan digunakan pelanggan.

4.1.3.2 Spesifikasi Pesyaratan Pengguna

Berikut ini akan mendaftar kebutuhan aplikasi pada tampilan pelanggan untuk melihat fungsi dan batasannya yang dapat dilakukan oleh pengguna yang dapat dilihat pada Tabel 4.4.

Tabel 4.4 Spesifikasi Persyaratan Pengguna

No.	Fitur	Fungsi	Batasan
1.	<i>Login</i> dan	Untuk mengakses	Pelanggan bisa mengakses

No.	Fitur	Fungsi	Batasan
	<i>Logout</i>	tampilan <i>user</i> pada aplikasi penyewaan jasa periklanan.	proses ini. Jika belum memiliki akses maka harus dibuat pada tampilan <i>register</i> .
2.	<i>Register</i>	Untuk melakukan pendaftaran pelanggan jika belum memiliki akses.	Proses ini ini dilakukan oleh calon pelanggan untuk melakukan pendaftaran sebagai pelanggan.
3.	<i>History</i>	Melihat riwayat pemesanan.	Pelanggan bisa mengakses proses ini.
4.	<i>Order</i>	Melakukan pemesanan.	Proses ini dilakukan oleh pelanggan.
5.	<i>Order List</i>	Untuk melihat daftar pesanan.	Proses ini dilakukan oleh pelanggan.
6.	<i>Cancel Order</i>	Untuk membatalkan pesanan.	Pelanggan bisa mengakses fitur ini jika pesanan belum diproses. Jika sudah diproses maka pelanggan tidak bisa melakukan pembatalan pesanan.
7.	<i>Edit Order</i>	Menambah pesanan atau mengurangi pesanan.	Pelanggan bisa mengakses fitur ini.
8.	Notifikasi	Pemberitahuan mengenai pesanan kepada pelanggan	Pelanggan bisa mengakses pemberitahuan untuk melihat proses pesanan.
9.	<i>Feedback</i>	Untuk memberikan kritik dan saran.	Pelanggan bisa mengakses fitur ini untuk memberikan kritik dan saran.
10.	Pembayaran	Untuk melakukan pembayaran.	Pelanggan hanya dapat melakukan transaksi.
11.	<i>Tracking</i>	Untuk melihat pesanan	Pelanggan hanya dapat

No.	Fitur	Fungsi	Batasan
	<i>queue</i>	sudah diproses atau belum.	melihat proses pembuatan pesanan.
12.	<i>Upload</i>	Untuk mengunggah desain dari pesanan yang akan digunakan dan bukti pembayaran yang sudah dilakukan.	Pelanggan hanya dapat mengunggah <i>file</i> yang mempunyai format JPEG, dan PNG.

4.2 User Design

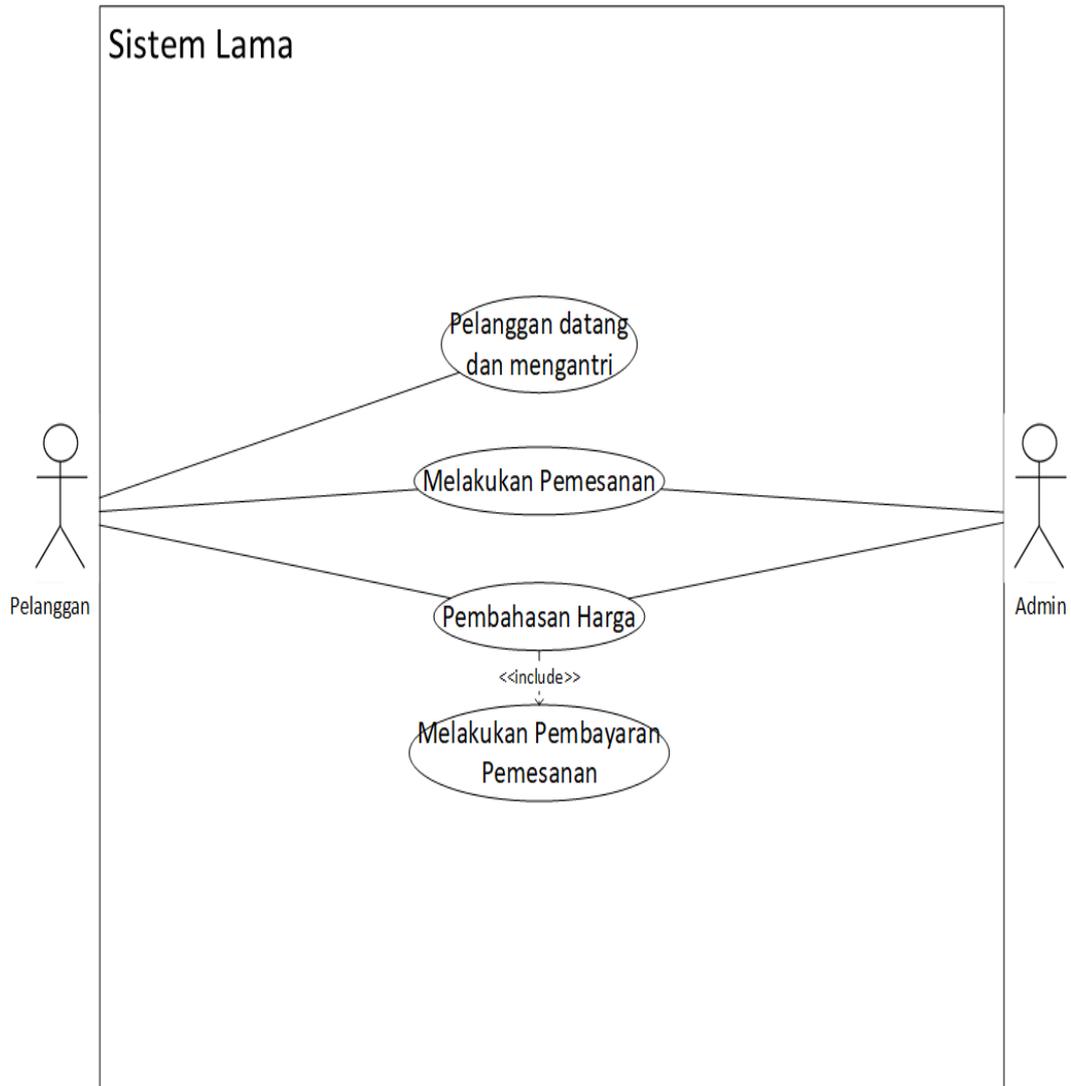
Fase kedua dari RAD adalah *User Design*. Penulis akan memodelkan semua model yang mewakili semua proses, dari aplikasi yang akan dibuat dan rancangan *storyboard*, sehingga didapatkan hasil berupa model aplikasi yang akan dibuat dan *storyboard*. Adapun langkah-langkah dalam tahapan ini adalah sebagai berikut.

4.2.1 Memodelkan Aplikasi Yang Akan Dibuat

Melakukan pemodelan aplikasi merupakan proses pengembangan abstrak model dari suatu aplikasi, di mana setiap model menggambarkan pandangan yang berbeda dari aplikasi tersebut. Pemodelan aplikasi membantu untuk memahami fungsionalitas aplikasi dan model. Pemodelan ini menggunakan dua diagram UML, yaitu *use case diagram*, dan *activity diagram*.

4.2.1.1 Use Case Diagram

Penulis akan menggambarkan hubungan aktor dan aplikasi. Berikut adalah gambar dari *use case diagram* sistem lama yang sementara ini digunakan. *Use case* tersebut dapat dilihat pada Gambar 4.1 yang ada di halaman selanjutnya.



Gambar 4.1 Use Case Diagram Sistem Lama

Tabel 4.5 Use Case Pelanggan datang dan mengantri

Nama Use Case	Pelanggan datang dan mengantri
Aktor	Pelanggan
Deskripsi	Pelanggan mengantri sebelum melakukan pemesanan
Normal Course	Melakukan Pemesanan
Alternate Course	Tidak jadi melakukan pemesanan
Pre-Condition	-
Post-Condition	-

Tabel 4.6 Use Case Melakukan Pemesanan

Nama Use Case	Melakukan Pemesanan
----------------------	---------------------

Aktor	Pelanggan
Deskripsi	Melakukan Pemesanan dan memberikan detail tentang produk yang akan dibuat
Normal Course	Melakukan Pemesanan
Alternate Course	Tidak jadi melakukan pemesanan
Pre-Condition	-
Post-Condition	-

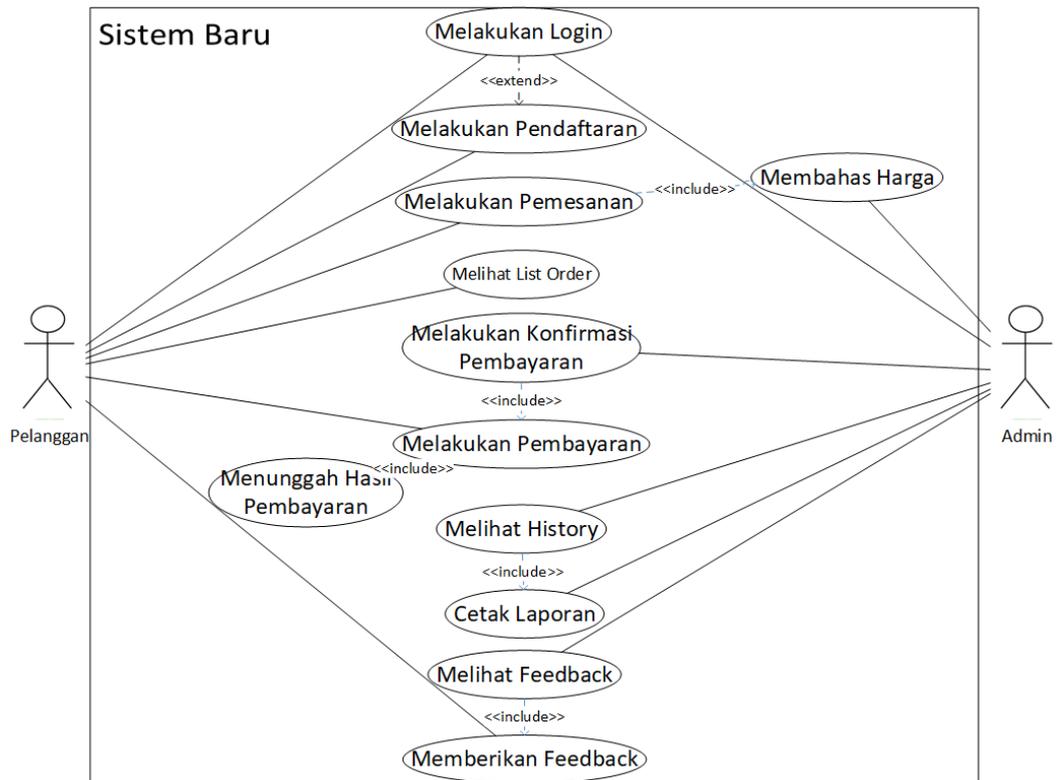
Tabel 4.7 Use Case Pembahasan Harga

Nama Use Case	Pembahasan Harga
Aktor	Pelanggan, <i>Admin</i>
Deskripsi	Melakukan percakapan tentang harga
Normal Course	Negosiasi harga
Alternate Course	Tidak jadi melakukan pemesanan
Pre-Condition	Melakukan pemesanan
Post-Condition	-

Tabel 4.8 Use Case Melakukan Pembayaran Pemesanan

Nama Use Case	Melakukan Pembayaran Pemesanan
Aktor	Pelanggan, <i>Admin</i>
Deskripsi	Melakukan Pembayaran
Normal Course	Pembayaran
Alternate Course	Tidak jadi membayar
Pre-Condition	-
Post-Condition	-

Berikut adalah gambar dari *use case diagram* sistem baru. *Use case* tersebut dapat dilihat pada Gambar 4.2 yang ada di halaman selanjutnya.



Gambar 4.2 Use Case Diagram Sistem Baru

Kegiatan pertama yang sesuai dengan yang ada pada *use case diagram* dapat dilihat pada Tabel 4.9.

Tabel 4.9 Use Case Melakukan Login

Nama Use Case	Melakukan <i>Login</i>
Aktor	Pelanggan dan <i>Admin</i>
Deskripsi	Menjelaskan hak akses untuk masuk ke dalam aplikasi
Normal Course	<ol style="list-style-type: none"> 1. Memasukkan <i>username</i> dan kata sandi. 2. <i>Username</i> dan kata sandi benar, masuk ke tampilan utama.
Alternate Course	Salah memasukkan <i>username</i> atau kata sandi.
Pre-Condition	Register
Post-Condition	Order

Kegiatan berikutnya adalah melakukan pemesanan kamar yang di deskripsikan pada Tabel 4.10.

Tabel 4.10 Use Case Melakukan Pemesanan

Nama Use Case	Melakukan <i>Pemesanan</i>
Aktor	Pelanggan

Deskripsi	Menjelaskan hak akses untuk <i>Order</i>
Normal Course	1. Memilih Produk 2. Melakukan Pemesanan Produk
Alternate Course	Tidak jadi melakukan pemesanan Mengganti Produk yang dipesan Membatalkan pemesanan
Pre-Condition	<i>Login</i>
Post-Condition	<i>Order</i>

Kegiatan berikutnya adalah melakukan *Register* yang dideskripsikan pada Tabel 4.11.

Tabel 4.11 Use Case Melakukan Pendaftaran

Nama Use Case	Melakukan Pendaftaran
Aktor	Pelanggan
Deskripsi	Menjelaskan hak akses untuk <i>Register</i>
Normal Course	1. Mengisi <i>form</i> pendaftaran
Alternate Course	Tidak jadi melakukan <i>Register</i>
Pre-Condition	-
Post-Condition	-

Kegiatan berikutnya adalah melihat laporan yang dideskripsikan pada Tabel 4.12.

Tabel 4.12 Use Case List Order

Nama Use Case	<i>List Order</i>
Aktor	Pelanggan
Deskripsi	Menjelaskan hak akses untuk melihat daftar pemesanan
Normal Course	Melihat daftar pemesanan
Alternate Course	Tidak jadi melihat daftar pemesanan
Pre-Condition	<i>Login</i>
Post-Condition	-

Kegiatan berikutnya adalah melihat *logout* yang dideskripsikan pada Tabel 4.13. di halaman selanjutnya.

Tabel 4.13 Use Case Melihat History

Nama Use Case	Melihat <i>History</i>
Aktor	<i>Admin</i>
Deskripsi	Menjelaskan hak akses untuk melihat kronologi pesanan
Normal Course	Melihat <i>history</i>
Alternate Course	Tidak jadi melihat <i>history</i>
Pre-Condition	Login
Post-Condition	-

Kegiatan berikutnya adalah melihat pembayaran yang dideskripsikan pada Tabel 4.14.

Tabel 4.14 Use Case Melakukan Pembayaran

Nama Use Case	Melakukan Pembayaran
Aktor	Pelanggan
Deskripsi	Menjelaskan hak akses untuk Pembayaran
Normal Course	<ol style="list-style-type: none"> 1. Melihat harga dari hasil pemesanan produk. 2. Mengunggah hasil dari pembayaran dalam bentuk <i>file images</i>. 3. Melihat notifikasi dari hasil unggahan yang akan disetujui <i>admin</i>.
Alternate Course	Tidak jadi mengakses pembayaran
Pre-Condition	<i>Login</i>
Post-Condition	-

Kegiatan berikutnya adalah melihat Konfirmasi Pembayaran yang dideskripsikan pada Tabel 4.15.

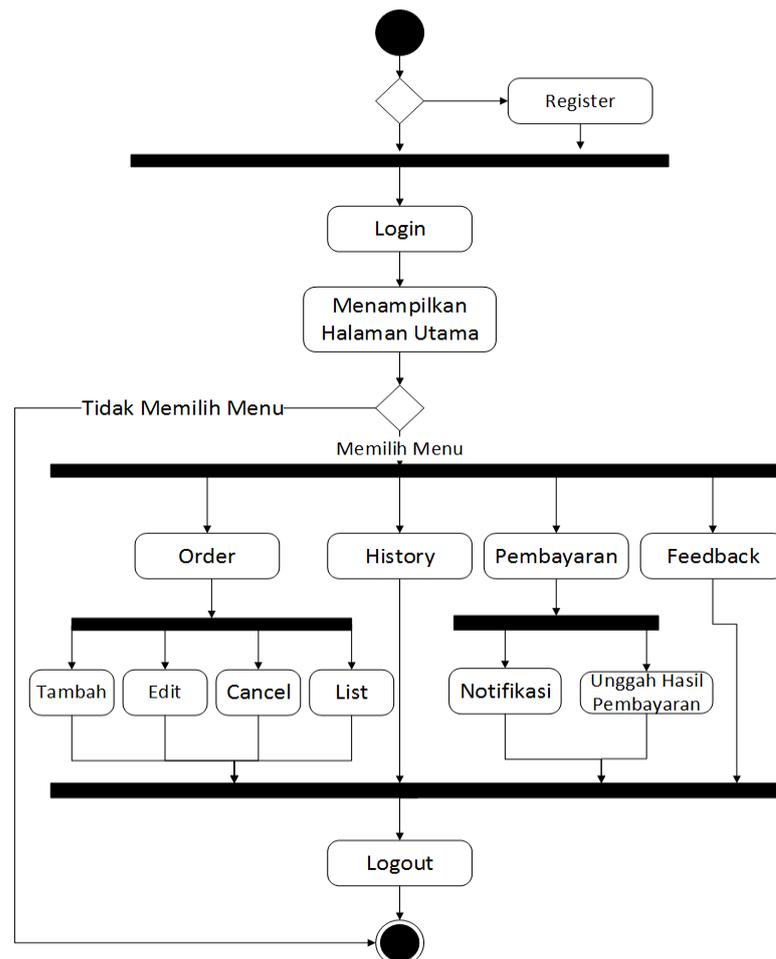
Tabel 4.15 Use Case Konfirmasi Pembayaran

Nama Use Case	Konfirmasi Pembayaran
Aktor	<i>Admin</i>
Deskripsi	Menjelaskan hak akses untuk Konfirmasi Pembayaran
Normal Course	<ol style="list-style-type: none"> 1. Melihat hasil dari unggahan hasil dari pembayaran pelanggan. 2. Melakukan penyetujuan bahwa hasil dari unggahan

	sudah di setuju <i>admin</i> . 3. Memberikan Pemberitahuan kepada pelanggan jika hasil dari pembayaran sudah diterima dan juga sudah dilanjutkan ke tahap <i>queue</i> .
Alternate Course	Tidak jadi mengakses pembayaran
Pre-Condition	<i>Login</i>
Post-Condition	-

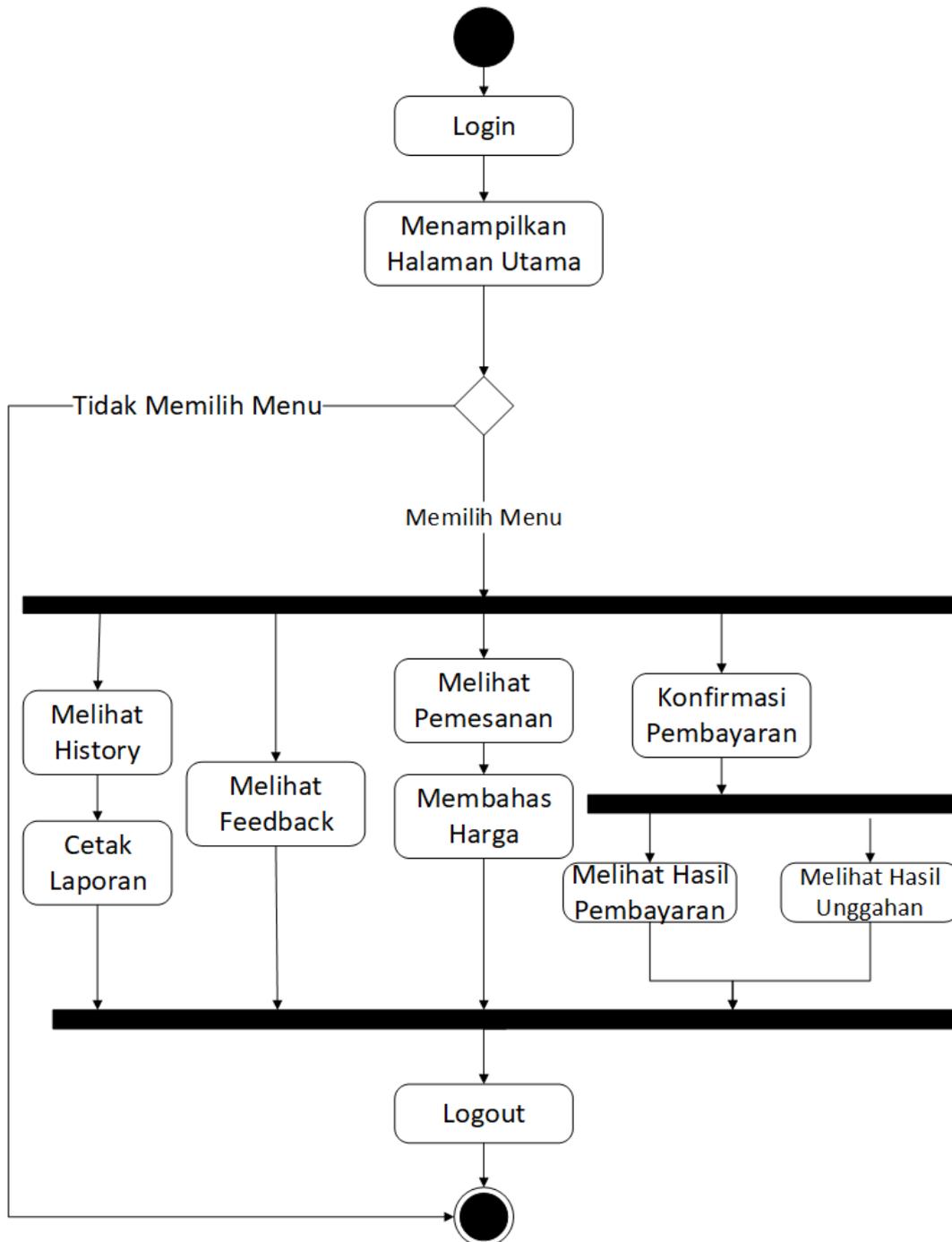
4.2.1.2 Activity Diagram

Pada tahapan ini akan digambarkan *activity diagram* dan penjelasan tentang hubungan antar objek serta fungsionalitas dari *activity diagram*, yang terbagi 2 yaitu *activity diagram* pelanggan ditunjukkan pada Gambar 4.3, dan *activity diagram admin* yang ditunjukkan pada Gambar 4.4.



Gambar 4.3 Activity Diagram Pelanggan

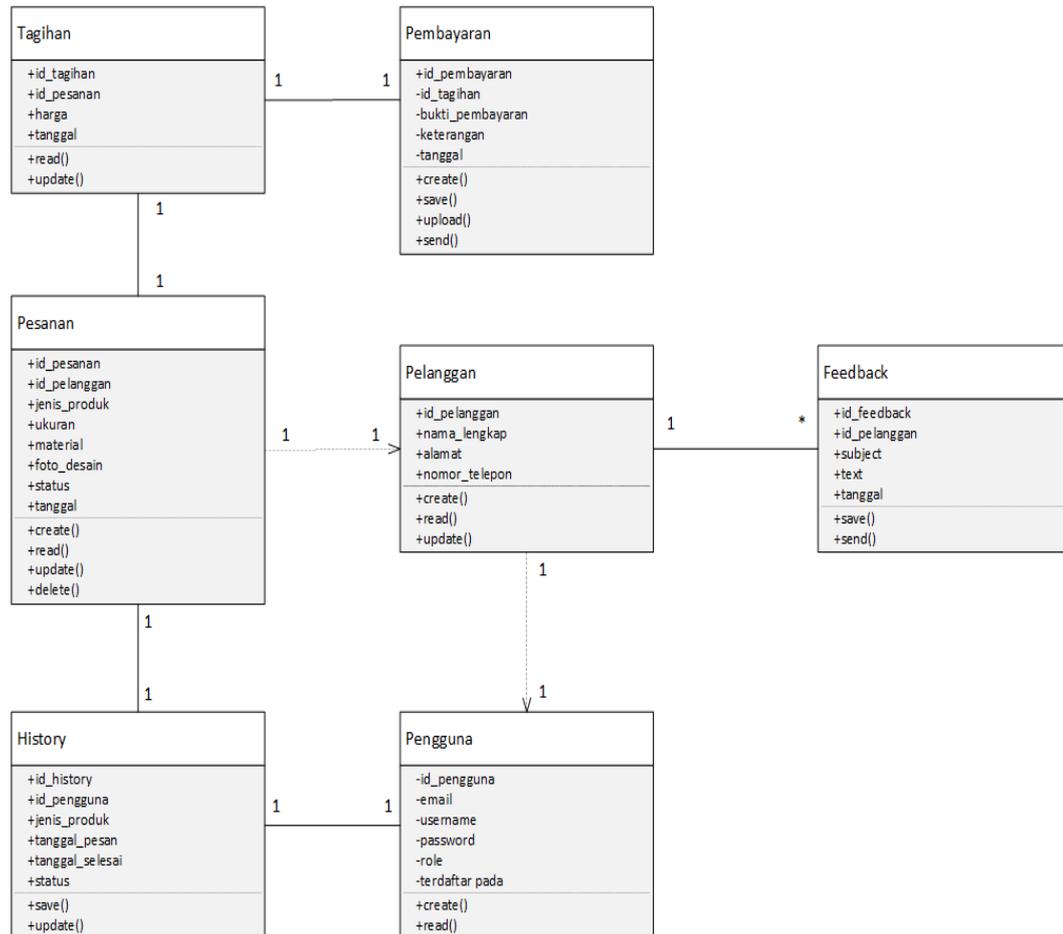
Berikut adalah *Activity Diagram Admin* yang bisa dilihat pada Gambar 4.4.



Gambar 4.4 Activity Diagram Admin

4.2.1.3 Class Diagram

Class Diagram untuk menggambarkan secara detail perbedaan mendasar antar *class*, hubungan antar *class*, dan di mana subsistem *class* tersebut. *Class Diagram* dapat dilihat pada Gambar 4.5



Gambar 4.5 Class Diagram

4.2.1.4 Storyboard

Dalam tahapan ini, *storyboard* yang telah dibuat dibagi menjadi 2 bagian yaitu *Admin* dan pelanggan.

a. Tampilan Pelanggan

Berikut akan ditunjukkan tampilan yang ada pada halaman pelanggan.

1. Login

Login sebagai pemberi hak akses kepada pengguna aplikasi, yaitu admin dan pelanggan. Berikut adalah tampilan *Login* Pelanggan yang bisa dilihat pada Gambar 4.6 di halaman selanjutnya.

The image shows a login form titled "LOGIN". It contains three input fields: "username", "password", and a "login" button. Below the button is a link that says "Belum punya akun? Daftar Sekarang!".

Gambar 4.6 Tampilan *Login*

2. *Register*

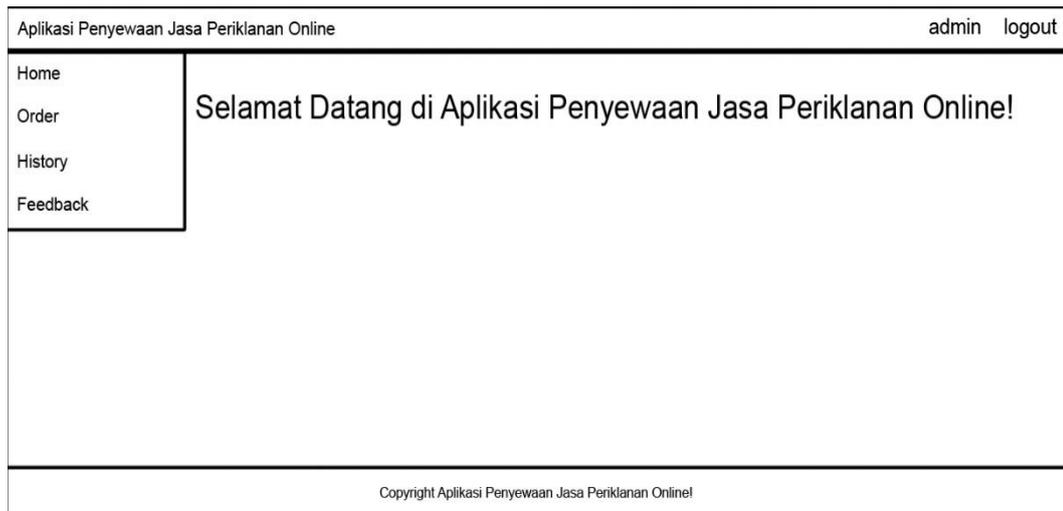
Register adalah halaman pengguna untuk mendaftarkan diri agar mendapatkan hak akses halaman *login*.

The image shows a register form titled "REGISTER". It contains seven input fields: "email", "username", "password", "nama lengkap", "alamat", "nomor telepon", and a "register" button.

Gambar 4.7 Tampilan *Register* Pelanggan

3. *Home*

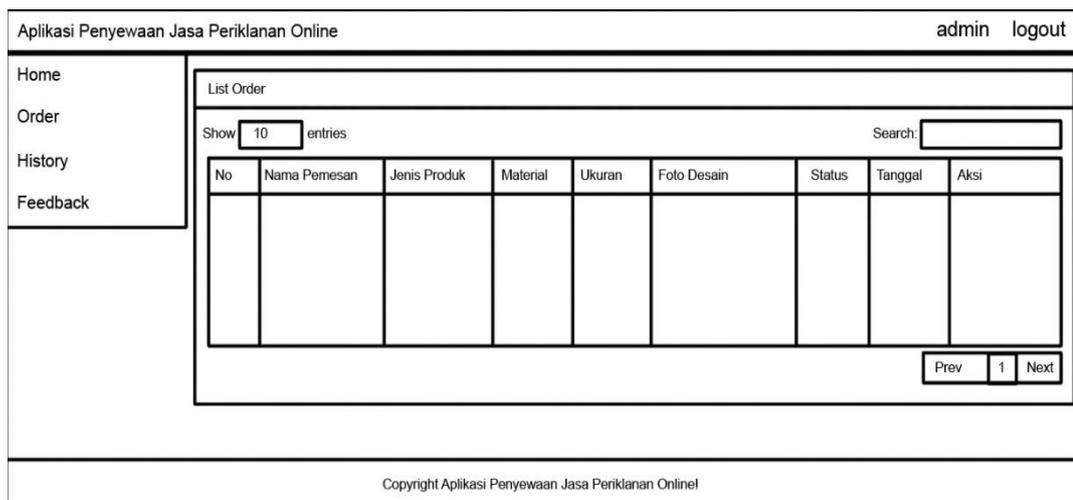
Home adalah halaman awal setelah pelanggan melakukan *login* yang menampilkan beberapa contoh hasil produk *advertisement*. Berikut adalah tampilan Home Pelanggan yang bisa dilihat pada Gambar 4.8. di halaman selanjutnya.



Gambar 4.8 Tampilan *Home* Pelanggan

4. *Order*

Order adalah halaman untuk pelanggan melakukan proses pemesanan, dan pembatalan. Berikut adalah tampilan *order* yang bisa dilihat pada Gambar 4.9.



Gambar 4.9 Tampilan *Order*

5. *Pembayaran*

Pembayaran adalah halaman untuk pelanggan melihat notifikasi dari *Admin* ketika pesanan mereka sudah diterima lalu dapat melanjutkan ke tahap pembayaran di mana pelanggan dapat mengunggah hasil pembayaran dari halaman ini. Berikut adalah tampilan dari pembayaran pelanggan pada gambar 4.10 di halaman selanjutnya.

Pembayaran	
Pesanan:	
Tidak ada pesanan	
Bukti Pembayaran :	
choose file	No file chosen
Keterangan:	
<input type="button" value="Konfirmasi"/>	

Gambar 4.10 Tampilan Pembayaran Pelanggan

b. Tampilan Admin

Berikut adalah tampilan yang ada pada *admin*.

1. Konfirmasi Pembayaran

Konfirmasi Pembayaran adalah halaman untuk *admin* melihat dari pesanan pelanggan lalu memberitahukan pelanggan kalau pesannya akan segera diproses, juga untuk melihat hasil dari unggahan pembayaran pelanggan. Berikut adalah tampilan dari konfirmasi pembayaran pada gambar 4.11.

Bahas Harga		x
Detail Pesanan		
Jenis Produk	:	Billboard
Material	:	ultraflex
Ukuran	:	small
Harga:		
<input type="text" value="200000"/>		
		<input type="button" value="close"/> <input type="button" value="pesan sekarang"/>

Gambar 4.11 Tampilan Konfirmasi Pembayaran

4.3 Konstruksi

Pada tahap ini, adalah tahap yang membahas pembuatan aplikasi yang akan dibangun sesuai dengan rancangan dan analisis yang telah dilakukan. Hasil dari tahapan ini adalah aplikasi penyewaan jasa periklanan. Berikut ini adalah langkah-langkah yang dilakukan pada tahapan ini.

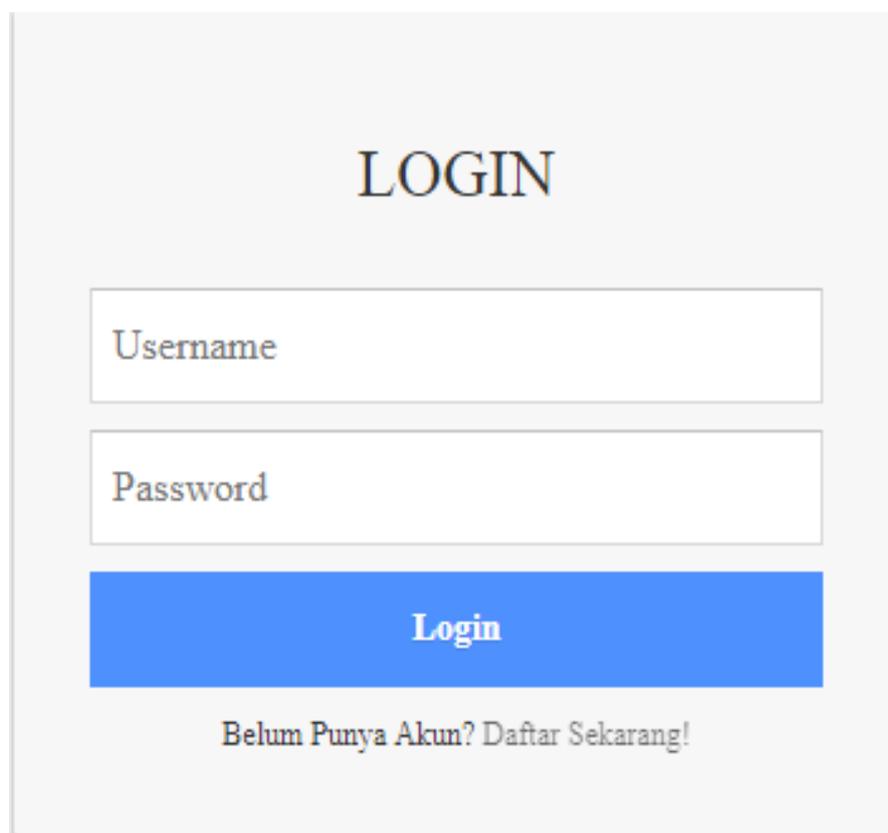
4.3.1 Pengembangan Aplikasi

Pada tahapan ini adalah penjelasan dari pembuatan aplikasi yang terdiri dari implimentasi *storyboard* dan implementasi basis data.

a. Implementasi *Storyboard*

1. Halaman *Login*

Halaman *login* ini berfungsi untuk mengakses halaman utama, tetapi jika belum mempunyai akun maka harus melakukan registrasi terlebih dahulu. Berikut adalah tampilan dari *login* pada Gambar 4.12.



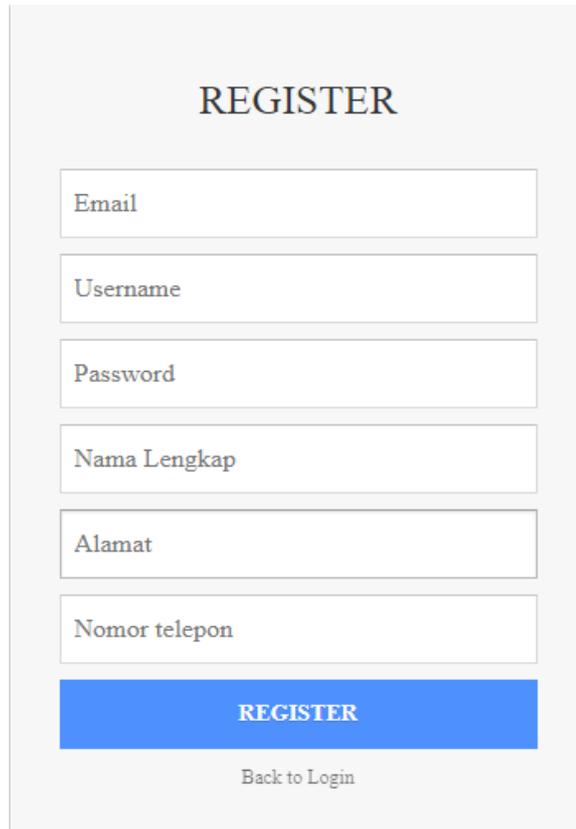
The image shows a login form with the following elements:

- The word "LOGIN" is centered at the top in a large, blue, serif font.
- Below it is a white rectangular input field with the placeholder text "Username" in a light blue font.
- Below that is another white rectangular input field with the placeholder text "Password" in a light blue font.
- Below the password field is a solid blue rectangular button with the word "Login" in white, bold, sans-serif font.
- At the bottom of the form is the text "Belum Punya Akun? Daftar Sekarang!" in a light blue, sans-serif font.

Gambar 4.12 Tampilan *Login*

2. Halaman *Register*

Halaman ini berfungsi untuk membuat akun agar dapat melakukan *login* dan mengakses halaman utama. Berikut adalah tampilan dari halaman *register* pada gambar 4.13.

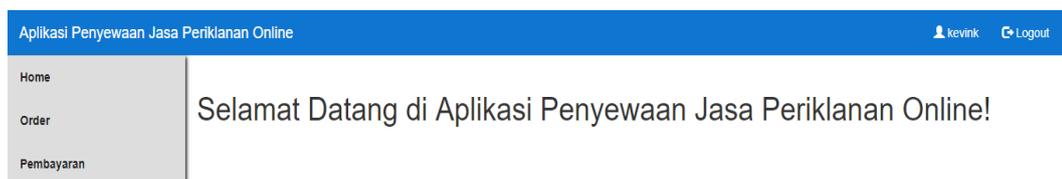


The image shows a registration form titled "REGISTER". It contains six input fields stacked vertically: "Email", "Username", "Password", "Nama Lengkap", "Alamat", and "Nomor telepon". Below the input fields is a blue button labeled "REGISTER" and a link labeled "Back to Login".

Gambar 4.13 Tampilan *Register*

3. Halaman *Home*

Berikut adalah Halaman *Home* yang dapat dilihat pada Gambar 4.14.



Gambar 4.14 Tampilan *Home*

b. Implementasi Basis Data

Berikut adalah hasil *screenshot* basis data pada aplikasi penyewaan jasa periklanan.

1. Tabel *Feedback*

Berikut adalah hasil *screenshot* tabel *feedback* yang dapat dilihat pada Gambar 4.15.

#	Name	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/>	1 id_feedback 	int(11)			No	None	AUTO_INCREMENT
<input type="checkbox"/>	2 id_pelanggan 	int(11)			No	None	
<input type="checkbox"/>	3 subject	varchar(100)			No	None	
<input type="checkbox"/>	4 text	text			No	None	
<input type="checkbox"/>	5 tanggal	date			No	None	

Gambar 4.15 Tabel *Feedback*

2. Tabel *History*

Berikut adalah hasil *screenshot* tabel *history* yang dapat dilihat pada Gambar 4.16.

#	Name	Type	Collation	Attributes	Null	Default	Extra
<input type="checkbox"/>	1 id_history 	int(11)			No	None	AUTO_INCREMENT
<input type="checkbox"/>	2 id_pengguna 	int(11)			No	None	
<input type="checkbox"/>	3 jenis_produk	varchar(50)			No	None	
<input type="checkbox"/>	4 tanggal_pesan	date			No	None	
<input type="checkbox"/>	5 tanggal_selesai	date			No	None	
<input type="checkbox"/>	6 status	varchar(10)			No	None	

Gambar 4.16 Tabel *History*

3. Tabel Pelanggan

Berikut adalah hasil *screenshot* tabel pelanggan yang dapat dilihat pada Gambar 4.17 di halaman selanjutnya.

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	id_pelanggan	int(11)			No	None	
2	nama_lengkap	varchar(50)			No	None	
3	alamat	varchar(50)			No	None	
4	nomor_telepon	varchar(14)			No	None	

Gambar 4.17 Tabel Pelanggan

4. Tabel Pembayaran

Berikut adalah hasil *screenshot* tabel pembayaran yang dapat dilihat pada Gambar 4.18.

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	id_pembayaran	int(11)			No	None	AUTO_INCREMENT
2	id_tagihan	int(11)			No	None	
3	bukti_pembayaran	varchar(100)			No	None	
4	keterangan	text			No	None	
5	tanggal	date			No	None	

Gambar 4.18 Tabel Pembayaran

5. Tabel Pengguna

Berikut adalah hasil *screenshot* tabel pengguna yang dapat dilihat pada Gambar 4.19.

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	id_pengguna	int(11)			No	None	AUTO_INCREMENT
2	email	varchar(50)			No	None	
3	username	varchar(50)			No	None	
4	password	varchar(50)			No	None	
5	role	set('pelanggan', 'admin')			No	pelanggan	
6	terdaftar_pada	date			No	None	

Gambar 4.19 Tabel Pengguna

6. Tabel Pesanan

Berikut adalah hasil *screenshot* tabel pesanan yang dapat dilihat pada Gambar 4.20 di halaman selanjutnya.

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	id_pesanan	int(11)			No	None	AUTO_INCREMENT
2	id_pelanggan	int(11)			No	None	
3	jenis_produk	varchar(50)			No	None	
4	ukuran	varchar(50)			No	None	
5	material	varchar(50)			No	None	
6	foto_desain	varchar(100)			No	None	
7	status	set('menunggu', 'dibayar', 'diproses', 'dikonfirmasi')			No	menunggu	
8	tanggal	date			No	None	

Gambar 4.20 Tabel Pesanan

7. Tabel Tagihan

Berikut adalah hasil *screenshot* tabel tagihan yang dapat dilihat pada Gambar 4.21.

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	id_tagihan	int(11)			No	None	AUTO_INCREMENT
2	id_pesanan	int(11)			No	None	
3	harga	int(11)			No	0	
4	tanggal	date			No	None	

Gambar 4.21 Tabel Tagihan

4.3.2 Melakukan Pemrograman

```

Script
<?php
defined('BASEPATH') OR exit('No direct script access allowed');
class Order extends CI_Controller {
    public function __construct()
    {
        parent::__construct();
        $this->load->model('auth_model');
        $this->load->model('user_model');
        $this->load->model('order_model');
        $this->auth_model->check();
    }

    public function index()

```

Script

```

{
    // Halaman Order
    $head['title'] = "Order";
    $head['user'] = $this->user_model->getUserByID($this->session->userdata('id_pengguna'));
    $this->load->view('_templates/header', $head);
    $this->load->view('_templates/sidemenu');
    $data['page'] = '/order/index_order';
    $data['name'] = $this->config->item("site_name");
    $data['user'] = $this->user_model->getUserByID($this->session->userdata('id_pengguna'));
    if($data['user']->role == 'pelanggan'){
        // Jika user yg login adalah pelanggan
        $data['order'] = $this->order_model->getOrderByPelanggan($data['user']->id_pengguna);
    }else $data['order'] = $this->order_model->getOrder();
    $this->load->view('_templates/content', $data);
    if(isset($_GET['new']) && $data['user']->role == 'pelanggan')
    { // menampilkan modal order
        $this->load->view('order/modal_order_baru', $data);
    }
    if(isset($_GET['offer_price']) && $data['user']->role == 'admin')
    { // menampilkan modal pembahasan harga
        $data['detail'] = $this->order_model->getOrderByID($this->input->get('offer_price'));
        $this->load->view('order/modal_offer_price', $data);
    }
    if(isset($_GET['check_price']) && $data['user']->role == 'pelanggan')
    { // menampilkan modal lihat harga
        $this->load->model('pembayaran_model');
        $data['detail'] = $this->pembayaran_model->getTagihanByPesanan($this->input->get('check_price'));
        $data['a'] = 'aa';
        $this->load->view('order/modal_check_price', $data);
    }
}

```

Script

```

        if(isset($_GET['pembayaran']) && $data['user']->role ==
'admin')
        { // menampilkan modal pembahasan harga
            $this->load->model('pembayaran_model');
            $data['detail'] = $this->pembayaran_model-
>getPembayaranByID($this->input->get('pembayaran'));
            $this->load->view('order/modal_cek_pembayaran',
$data);
        }

        $foot['name'] = $data['name'];
        $this->load->view('_templates/footer', $foot);
    }

    public function tambah_baru()
    {
        $user = $this->user_model->getUserByID($this->session-
>userdata('id_pengguna'));
        if($user->role == 'pelanggan'){ // Hanya Untuk Pelanggan
            $config['upload_path']          = './app-uploads/';
            $config['allowed_types']        = 'gif|jpg|png';
            $config['encrypt_name']         = true;
            $this->load->library('upload', $config);
            if ( ! $this->upload->do_upload('foto_desain'))
            {
                $error = $this->upload->display_errors();
            }
            else
            {
                $data = array('upload_data' => $this-
>upload->data());
                $foto_desain =
$data['upload_data']['file_name'];
            }
            $data = $this->input->post();
            unset($data['foto_desain']);
            if(isset($data['customSize'])) {

```


Script

```

{
    $err = "Pemesanan hanya bisa dilakukan
Pelanggan";
    $this->session->set_flashdata('error', $err);
    redirect('order/?&error='.$err);
}

}

public function offer_price()

{

    $data = $this->input->post();
    $id = $this->input->post('id_pesanan');
    if($this->order_model->cekTagihanByID($id)){
        $err = "Tagihan sudah ada";
        $this->session->set_flashdata('error', $err);
        redirect('order/?offer_price='.$id.'&error='.$err);
    }else

{

        $insert = $this->order_model->offerPrice($data,
$id);
        if($insert['status']){
            $this->session->set_flashdata('msg',
'sukses');
            redirect('order');
        }else
        {
            $this->session->set_flashdata('error',
$insert['error']);
            redirect('order/?offer_price='.$id.'&error='.$insert['error']);
        }
}
}

```

Script

```
    }

}

public function konfirmasiPesanan()

{

    $this->order_model->konfirmasiPesanan($this->uri-
>segment(3));
    if($insert['status'])
{
        $this->session->set_flashdata('msg', 'sukses');
        redirect('order');

    }else
{

        $this->session->set_flashdata('error',
$insert['error']);
        redirect('order/?error='.$insert['error']);

    }
}

public function cancelPesanan()

{

    $this->order_model->cancelPesanan($this->uri-
>segment(3));
    if($insert['status'])
{
        $this->session->set_flashdata('msg', 'sukses');
        redirect('order');

    }else
```

Script

```

{
    $this->session->set_flashdata('error',
$insert['error']);
    redirect('order/?error='.$insert['error']);
}
}
}
}

```

4.4 **Testing and Turn Over**

Fase keempat dari RAD adalah *testing and turn over*. Pada fase ini akan diuji apakah aplikasi yang telah dibuat telah berjalan dengan baik serta telah sesuai dengan spesifikasi analisis dan perancangan, sehingga didapatkan hasil berupa pengujian yang terdiri dari tujuan pengujian, kriteria pengujian, kasus pengujian, *test plan*, *user acceptance test*, dan hasil pengujian,. Adapun langkah-langkah dalam tahapan ini adalah sebagai berikut.

4.4.1 Pengujian

Dalam pengujian terdapat tujuan pengujian, kriteria pengujian, kasus pengujian, *test plan*, *user acceptance test*, dan hasil pengujian.

1. Tujuan Pengujian

Pada tahap ini akan dijelaskan tentang tujuan pengujian dari aplikasi. Berikut tujuan dari pengujian aplikasi:

1. Memastikan apakah aplikasi yang telah dibangun sesuai dengan analisis dan perancangan yang dilakukan.
2. Mengetahui apakah manfaat dan kebutuhan pengguna telah terpenuhi atau belum terpenuhi.
3. Memastikan semua persyaratan pengguna dan aplikasi sudah dapat diimplementasikan.
4. Memastikan fungsi-fungsi dalam aplikasi telah berjalan sesuai dengan keinginan pengguna.
5. Mengetahui respons dari aplikasi apabila dimasukkan data.
6. Menerima respons dari pengguna mengenai aplikasi yang telah dibangun.

2. Kriteria Pengujian

Berikut kriteria-kriteria dari aplikasi penyewaan jasa periklanan yang telah dibuat:

1. Aplikasi telah memenuhi persyaratan pengguna.
2. Aplikasi mudah digunakan oleh pengguna.
3. Aplikasi sudah dapat membantu penyelesaian masalah yang ada.
4. Semua fungsi dalam aplikasi berjalan sesuai yang diharapkan.

3. Kasus Pengujian

Berikut penulis akan menjelaskan kasus pengujian dari aplikasi yang telah dibangun yang dapat dilihat pada Tabel 4.16.

Tabel 4.16 Kasus Pengujian

No.	Titik Pemeriksaan
1.	Fitur <i>Login</i>
	Apakah <i>username</i> dan <i>password</i> yang dimasukkan dapat dibaca oleh aplikasi?
	Apakah fitur login dapat membedakan <i>admin</i> , pelanggan?
2.	Fitur <i>Order</i>
	Apakah dapat melakukan pemesanan?
	Apakah dapat mengedit pemesanan?
3.	Fitur <i>History</i>
	Apakah riwayat pemesanan dapat ditampilkan?
4.	Fitur Pembayaran
	Apakah notifikasi yang diberikan <i>admin</i> berfungsi?
	Apakah bukti pembayaran yang diunggah masuk ke dalam basis data?
5.	Fitur <i>Feedback</i>
	Apakah hasil <i>feedback</i> dapat dibaca oleh <i>admin</i> ?
6.	Fitur Konfirmasi Pembayaran
	Apakah dapat melakukan konfirmasi pembayaran?
	Apakah dapat mengirimkan notifikasi jika hasil desain sudah diterima?
7.	Fitur Laporan Bulanan

No.	Titik Pemeriksaan
	Apakah dapat melihat keuntungan bulanan?

4. Test Plan

Berikut ini merupakan tabel *test plan* sebagai bahan pengujian yang diisi oleh pembuat aplikasi yang dapat dilihat pada Tabel 4.17.

Tabel 4.17 Test Plan

Kegiatan Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Tampilan Register		
Mengisi semua <i>form register</i> yang ada.	Aplikasi akan menyimpan hasil isian <i>form</i> ke dalam basis data.	Menampilkan bahwa Registrasi berhasil.
Tidak mengisi salah satu dari <i>form register</i> yang ada.	Aplikasi akan menampilkan pesan peringatan.	Menampilkan pesan peringatan.
Tampilan Login		
Memasukkan <i>username</i> dan <i>password</i> yang benar	Aplikasi akan menampilkan tampilan beranda.	Menampilkan tampilan beranda aplikasi.
Memasukkan <i>username</i> dan <i>password</i> yang salah, atau hanya memasukkan salah satu.	Aplikasi akan menampilkan pesan peringatan.	Menampilkan pesan peringatan.
Tampilan Order		
Memilih jenis produk, mengunggah desain dan mengisi keterangan pada <i>form</i> yang disediakan.	Aplikasi akan menyimpan hasil unggahan pada basis data.	Menampilkan pesan berhasil disimpan.

Kegiatan Pengujian	Hasil yang Diharapkan	Hasil Pengujian
Tampilan <i>History</i>		
Melihat riwayat dari pemesanan yang telah dilakukan.	Aplikasi akan menampilkan riwayat dari pemesanan yang telah dilakukan.	Menampilkan riwayat dari pemesanan yang telah dilakukan.
Tampilan Pembayaran		
Memilih menu pada pembayaran (unggah hasil pembayaran, Melihat notifikasi konfirmasi dari <i>admin</i>).	Aplikasi akan menampilkan pesan berhasil menunggu, atau notifikasi konfirmasi dari <i>admin</i> .	Menampilkan pesan berhasil menunggu, atau notifikasi konfirmasi dari <i>admin</i> .
Tampilan <i>Feedback</i>		
Mengisi <i>form Feedback</i> .	Aplikasi akan menampilkan pesan berhasil dikirimkan	Menampilkan pesan berhasil dikirimkan.
Fitur <i>Logout</i>		
Pengguna dapat keluar dari aplikasi.	Aplikasi akan merespons aksi pengguna.	Aplikasi merespon aksi pengguna.
Tampilan Konfirmasi Pembayaran (<i>Admin</i>)		
Melihat hasil dari unggahan desain dari pelanggan, dan hasil unggahan pembayaran serta mengirimkan konfirmasi dari hasil unggahan.	Aplikasi akan menampilkan hasil unggahan dari pelanggan, dan mengirimkan konfirmasi dari unggahan.	Aplikasi akan menampilkan hasil unggahan dari pelanggan, dan mengirimkan konfirmasi dari unggahan.
Tampilan Laporan Bulanan (<i>Admin</i>)		
Melihat hasil	Aplikasi akan	Aplikasi dapat menampilkan

Kegiatan Pengujian	Hasil yang Diharapkan	Hasil Pengujian
keuntungan bulanan.	menampilkan hasil keuntungan bulanan.	hasil keuntungan bulanan.
Tampilan <i>Feedback</i> (Admin)		
Melihat <i>feedback</i> yang diberikan oleh pelanggan.	Aplikasi akan menampilkan <i>feedback</i> dari pelanggan.	Menampilkan <i>feedback</i> dari pelanggan.
Fitur <i>Logout</i>(Admin)		
Pengguna dapat keluar dari aplikasi.	Aplikasi akan merespons aksi pengguna.	Aplikasi merespons aksi pengguna.

a. *White Box Testing*

Pada pengujian ini akan dilakukan pada kode program untuk melihat alur logika pada kode program dan melihat kesalahan-kesalahan yang kemungkinan terdapat pada kode program. Berikut ini adalah tabel untuk pengujian *white box* dapat dilihat pada Tabel 4.18.

Tabel 4.18 *White Box Testing*

Kegiatan Pengujian	Kode Program	Hasil
Koneksi ke basis data	<pre> \$active_group = 'default'; \$query_builder = TRUE; \$db['default'] = array('dsn' => '', 'hostname' => 'localhost', 'username' => 'root', 'password' => '', 'database' => 'penyewaan_periklanan_db', 'dbdriver' => 'mysqli', 'dbprefix' => '', 'pconnect' => FALSE, </pre>	Berhasil

Kegiatan Pengujian	Kode Program	Hasil
	<pre> 'db_debug' => (ENVIRONMENT !=='production'), 'cache_on' => FALSE, 'cachedir' => '', 'char_set' => 'utf8', 'dbcollat' => 'utf8_general_ci', 'swap_pre' => '', 'encrypt' => FALSE, 'compress' => FALSE, 'stricton' => FALSE, 'failover' => array(), 'save_queries' => TRUE); </pre>	
<i>Login</i>	<pre> public function checkLogin() { // cek login \$this->auth_model->logged_in(); \$username = \$this->input- >post('username'); \$password = \$this->input- >post('password'); // memanggil fungsi cek login \$this->auth_model- >login(\$username, \$password); } </pre>	Berhasil
<i>Register</i>	<pre> //jika pendaftaran berhasil if(!isset(\$err)){ \$insert = \$this->auth_model- >new_user(\$data, \$data_pelanggan); if(\$insert['status']){ \$this->session- >set_flashdata('msg', array('message'=>'Pendaftaran Berhasil! Silahkan Login untuk </pre>	Berhasil

Kegiatan Pengujian	Kode Program	Hasil
	<pre> melanjutkan.', 'type'=>'success')); redirect('auth/register?login=Pendaftaran Berhasil!'); }else { \$this->session->set_flashdata('error', \$insert['error']); redirect('auth/register?error='.\$insert['error']); } //jika pendaftaran tidak berhasil }else { \$error = implode(" ", \$error); \$this->session->set_flashdata('error', \$error); redirect('auth/register?error='.\$error); } </pre>	
Tambah pesanan baru	<pre> //tambah pesanan if(!isset(\$error)){ \$insert = \$this->order_model->tambahPesanan(\$data); //print_r(\$insert); if(\$insert['status']){ \$this->session->set_flashdata('msg', 'Sukses'); redirect('order'); }else { \$this->session->set_flashdata('error', \$insert['error']); redirect('order/?new&error='.\$insert['error']); } } </pre>	Berhasil

Kegiatan Pengujian	Kode Program	Hasil
	<pre> }else { \$err = implode(" ", \$err); \$this->session->set_flashdata('error', \$err); redirect('order/?new&error='.\$err); }; </pre>	
<p>Lihat tagihan (Pelanggan)</p>	<pre> //tagihan pesanan \$data = \$this->input->post(); \$id = \$this->input->post('id_pesanan'); if(\$this->order_model->cekTagihanByID(\$id){ \$err = "Tagihan sudah ada"; \$this->session->set_flashdata('error', \$err); redirect('order/?offer_price='.\$id.'&error='.\$err); }else { \$insert = \$this->order_model->offerPrice(\$data, \$id); if(\$insert['status']){ \$this->session->set_flashdata('msg', 'sukses'); redirect('order'); }else { \$this->session->set_flashdata('error', \$insert['error']); redirect('order/?offer_price='.\$id.'&error='.\$insert['error']); } } </pre>	Berhasil
<p>Konfirmasi pesanan (Admin)</p>	<pre> //cek pesanan di database berdasarkan id_pesanan \$insert = \$this->order_model->konfirmasiPesanan(\$this->input->get('id_pesanan'), \$this->input->get('tanggal_selesai')); </pre>	Berhasil

Kegiatan Pengujian	Kode Program	Hasil
	<pre> //session untuk cek status pesanan kemudian masuk ke dalam tagihan pelanggan if(\$insert['status']){ \$this->session- >set_flashdata('msg', 'sukses'); redirect('order'); }else { \$this->session- >set_flashdata('error', \$insert['error']); redirect('order/?error='.\$insert[' error']); </pre>	
Melakukan pembayaran (Pelanggan)	<pre> //cek jika tagihan sesuai dengan pelanggan if(\$data['user']->role == 'pelanggan'){ // Jika user yg login adalah pelanggan \$data['pesanan'] = \$this- >pembayaran_model- >getOrderByPelanggan(\$data['user'] ->id_pengguna); } </pre>	Berhasil
Bayar (Pelanggan)	<pre> //cek pesanan berdasarkan id_pesanan if(isset(\$bukti_pembayaran)){ \$pesanan = \$this- >pembayaran_model- >getTagihanByPesanan(\$id_pesanan); \$data2 = array('id_tagihan' => \$pesanan->id_tagihan, 'tanggal' => date("Y/m/d"), 'bukti_pembayaran' => </pre>	Berhasil

Kegiatan Pengujian	Kode Program	Hasil
	<pre> \$bukti_pembayaran); \$data = array_merge(\$data,\$data2); if(isset(\$error)){ \$error = array_merge(\$error, \$error); } if(!isset(\$error)){ \$insert = \$this->pembayaran_model->bayar(\$data, \$id_pesanan); //print_r(\$insert); if(\$insert['status']){ \$this->session->set_flashdata('msg', 'sukses'); redirect('pembayaran'); }else { \$this->session->set_flashdata('error', \$insert['error']); redirect('pembayaran/?error='.\$insert['error']); } }else { \$error = implode(" ", \$error); \$this->session->set_flashdata('error', \$error); redirect('pembayaran/?error='.\$error); } } // cek jika pelanggan tidak upload bukti pembayaran }else { \$error = "Harap upload bukti pembayaran"; \$this->session->set_flashdata('error', \$error); redirect('pembayaran/?error='.\$error); } </pre>	

Kegiatan Pengujian	Kode Program	Hasil
	<pre> }else { \$err = \$error; \$this->session->set_flashdata('error', \$err); redirect('pembayaran/?error='.\$err); } </pre>	
Upload bukti pembayaran (Pelanggan)	<pre> \$user = \$this->user_model->getUserByID(\$this->session->userdata('id_pengguna')); if(\$user->role == 'pelanggan'){ // Hanya Untuk Pelanggan //syarat upload bukti pembayaran \$config['upload_path'] = './app-uploads/'; \$config['allowed_types'] = 'gif jpg png'; \$config['encrypt_name'] = true; \$config['max_size'] = '1000'; //kilobyte //untuk upload bukti pembayaran if (! \$this->upload->do_upload('bukti_pembayaran')) { \$error = \$this->upload->display_errors(); } else{ \$data = array('upload_data' => \$this->upload->data()); \$bukti_pembayaran = \$data['upload_data']['file_name']; } </pre>	Berhasil
Membatalkan pesanan	//pelanggan melakukan pembatalan pesanan	Berhasil

Kegiatan Pengujian	Kode Program	Hasil
(Pelanggan)	<pre> \$this->order_model- >cancelPesanan(\$this->uri- >segment(3)); //cek status pesanan, jika sudah dikonfirmasi tidak bisa dibatalkan if(\$insert['status']){ \$this->session- >set_flashdata('msg', 'sukses'); redirect('order'); }else { \$this->session- >set_flashdata('error', \$insert['error']); redirect('order/?error='.\$insert[' error']); </pre>	
Menampilkan halaman <i>history</i> (<i>admin</i>)	<pre> \$head['title'] = "History Page"; \$head['user'] = \$this- >user_model->getUserByID(\$this- >session- >userdata('id_pengguna')); \$this->load- >view('_templates/header', \$head); \$this->load- >view('_templates/sidemenu'); \$data['page'] = 'order/order_history'; \$data['name'] = \$this->config- >item("site_name"); //menampilkan history berdasarkan pengguna \$data['history'] = \$this->db- >query("select `history`.*,`pengguna`.*,`pelangga n`.* from history join `pengguna` </pre>	Berhasil

Kegiatan Pengujian	Kode Program	Hasil
	<pre> on `pengguna`.`id_pengguna`=`history` .id_pengguna join `pelanggan` on `pelanggan`.id_pelanggan=`pengguna` .id_pengguna"); \$this->load- >view('_templates/content', \$data); </pre>	
<p>Cetak laporan (Admin)</p>	<pre> // menampilkan laporan \$data['history'] = \$this->db- >query("select `history`.*,`pengguna`.*,`pelangga n`.* from history join `pengguna` on `pengguna`.`id_pengguna`=`history` .id_pengguna join `pelanggan` on `pelanggan`.id_pelanggan=`pengguna` .id_pengguna"); //konvert laporan ke PDF \$this->load- >library('PdfGenerator'); \$html = \$this->load- >view('order/print', \$data, true); </pre>	Berhasil
<p>Feedback (Pelanggan)</p>	<pre> //pelanggan memberikan feedback if(!isset(\$err)){ \$insert = \$this->user_model- >sendFeedback(\$data); //print_r(\$insert); if(\$insert['status']){ \$this->session- >set_flashdata('msg', 'sukses'); redirect('feedback?msg=Suskes'); }else { \$this->session- </pre>	Berhasil

Kegiatan Pengujian	Kode Program	Hasil
	<pre>>set_flashdata('error', \$insert['error']); redirect('feedback/?error='.\$insert['error']); }</pre>	

b. *Black Box*

Berikut ini adalah pengujian untuk melihat detail dari tampilan aplikasi, fungsi-fungsi aplikasi, dan kebutuhan dari pengguna. Pengujian *black box* dapat dilihat pada Tabel 4.19.

Tabel 4.19 Black Box Testing

Kegiatan Pengujian	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
Menampilkan tampilan <i>login</i> .	Aplikasi menampilkan halaman <i>login</i> .	Tampilan <i>login</i> terdapat dua <i>text field</i> , yaitu <i>username</i> dan <i>password</i> dan tombol <i>login</i> .	[√] Diterima [] Ditolak
Melakukan <i>login</i>	Aplikasi dapat membedakan pengguna antara pelanggan dan <i>admin</i> .	Pelanggan dan admin dapat login dengan memasukkan <i>username</i> dan <i>password</i> dengan benar.	[√] Diterima [] Ditolak
Mengisi <i>username</i> dan <i>password</i> dengan benar.	Menampilkan halaman utama aplikasi.	Tombol “ <i>Login</i> ” berfungsi dengan baik.	[√] Diterima [] Ditolak
Mengisi <i>username</i> dan <i>password</i> dengan	Kembali ke tampilan <i>login</i> .	<i>Username</i> dan <i>password</i> harus dimasukkan dengan	[√] Diterima [] Ditolak

Kegiatan Pengujian	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
salah.		benar.	
Jika hanya mengisi <i>username</i> atau <i>password</i> .	Aplikasi akan menampilkan pesan peringatan.	Menampilkan pesan peringatan jika <i>username</i> atau <i>password</i> salah.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Menampilkan tampilan registrasi.	Pelanggan baru yang belum mempunyai data, harus melakukan registrasi.	Mengisi <i>form register</i> .	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Mengisi semua <i>form register</i> .	Aplikasi akan menampilkan pesan peringatan.	Menampilkan bahwa Registrasi berhasil.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Tidak mengisi salah satu dari <i>text field</i> di <i>form register</i> .	Menampilkan pesan peringatan.	Menampilkan pesan peringatan.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Menampilkan halaman utama aplikasi.	Pengguna masuk ke dalam halaman utama aplikasi.	<i>Username</i> dan <i>password</i> diverifikasi.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Posisi menu berada pada sebelah kiri aplikasi.	Memilih menu.	Pengguna bisa memilih menu. Pada tampilan pelanggan terdapat menu <i>Home</i> , <i>Order</i> , <i>Pembayaran</i> , dan <i>Feedback</i> .	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Aplikasi Penyewaan Jasa Periklanan pada tampilan Pelanggan			
Tampilan Menu <i>Order</i>			
Memilih menu <i>order</i> .	Menampilkan tampilan <i>order</i> .	Aplikasi menampilkan	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak

Kegiatan Pengujian	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
		tampilan <i>order</i> .	
Menampilkan tampilan <i>order</i> .	Tampilan <i>order</i> akan menampilkan pesanan yang telah dibuat.	Tombol “Order” berfungsi dengan baik.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Mencari pesanan	Pesanan dapat dicari dengan mengisi kata kunci terkait.	Aplikasi akan mencari pesanan.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Tabel pesanan.	Tabel pesanan akan menampilkan informasi pesanan.	Tabel pesanan dapat menampilkan informasi pesanan.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Tombol pada kolom aksi.	Melihat harga dan membatalkan pesanan.	Tombol “Lihat Harga” dan “Batalkan” berfungsi dengan baik.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Tombol “Previous” dan “Next”	Menampilkan halaman sebelumnya atau selanjutnya.	Tombol “Previous” dan “Next” berfungsi dengan baik.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Melakukan Pesanan			
Klik Tombol “Order Baru”.	Menampilkan halaman <i>pop-up</i> detail pesanan yang harus diisi.	Tombol “Order Baru” berfungsi dengan baik.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Mengisi pesanan pada jenis produk.	Tombol <i>dropdown</i> akan menampilkan pilihan jenis produk, yaitu <i>billboard</i> , huruf	Klik tombol <i>dropdown</i> berfungsi dengan baik dan menampilkan pilihan jenis produk.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak

Kegiatan Pengujian	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
	timbul dan neon box.		
Mengisi pesanan pada ukuran.	Tombol <i>dropdown</i> akan menampilkan pilihan jenis produk, yaitu <i>small, medium, dan large</i> .	Klik tombol <i>dropdown</i> berfungsi dengan baik pada jenis produk untuk memilih ukuran.	[√] Diterima [] Ditolak
Mengunggah desain pesanan	Gambar dipilih melalui tombol yang telah disediakan.	Tombol “ <i>choose file</i> ” berfungsi dengan baik untuk memilih gambar.	[√] Diterima [] Ditolak
Selesai memesan.	Selesai melakukan pesanan akan ditampilkan pada tampilan <i>order</i> .	Tombol “pesan sekarang” berfungsi dengan baik untuk melakukan pesanan.	[√] Diterima [] Ditolak
Tampilan Pembayaran			
Menampilkan halaman pembayaran.	Melihat tagihan pembayaran	<ol style="list-style-type: none"> 1. Tombol <i>dropdown</i> “Pesanan” berfungsi dengan baik dan akan menampilkan pesanan yang harus dibayar. 2. Bukti pembayaran berfungsi dengan baik dengan mengunggah foto bukti pembayaran. 3. <i>Text field</i> 	[√] Diterima [] Ditolak

Kegiatan Pengujian	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
		“Keterangan” berfungsi dengan baik.	
<i>Admin</i> telah mengonfirmasi pesanan.	Melakukan pembayaran	Pelanggan bisa melakukan pembayaran	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Selesai melakukan pembayaran.	Pelanggan menekan konfirmasi untuk menyelesaikan proses pembayaran.	Tombol “Konfirmasi” berfungsi dengan baik.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Tampilan <i>Feedback</i>			
Menampilkan tampilan <i>Feedback</i> .	Terdapat dua <i>text field</i> , yaitu <i>Subject</i> dan Keterangan dan tombol kirim.	Tombol pada menu “ <i>Feedback</i> ” berfungsi dengan baik	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Mengisi <i>feedback</i> dan mengirimkan kepada <i>admin</i> .	<i>Text field</i> “ <i>subject</i> ” dan “keterangan” telah diisi oleh pelanggan.	Pelanggan berhasil mengirimkan <i>feedback</i> kepada <i>admin</i> dengan mengisi <i>Subject</i> dan Keterangan. Tombol “Kirim” berfungsi dengan baik.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Aplikasi Penyewaan Jasa Periklanan pada tampilan <i>Admin</i>			
Tampilan Menu <i>Order</i>			
Memilih menu <i>order</i> .	Menampilkan daftar <i>order</i> .	Aplikasi menampilkan daftar <i>order</i> .	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Mencari pesanan	Pesanan dapat	Aplikasi akan	<input checked="" type="checkbox"/> Diterima

Kegiatan Pengujian	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
	dicari dengan mengisi kata kunci terkait.	mencari pesanan.	<input type="checkbox"/> Ditolak
Tabel pesanan.	Tabel pesanan akan menampilkan informasi pesanan.	Tabel pesanan dapat menampilkan informasi pesanan.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Tombol pada kolom aksi.	Melihat harga dan membatalkan pesanan.	Tombol “Bahas Harga” berfungsi dengan baik untuk konfirmasi harga kepada pelanggan.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Tombol “Previous” dan “Next”	Menampilkan halaman sebelumnya atau selanjutnya.	Tombol “Previous” dan “Next” berfungsi dengan baik.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Tampilan History			
Menampilkan tampilan <i>History</i> pesanan.	Tampilan history akan menampilkan daftar pesanan yang telah masuk.	Tombol pada menu “History” berfungsi dengan baik.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Mencetak daftar pesanan.	Akan menampilkan halaman yang telah menjadi dokumen PDF dan bisa dicetak.	Tombol “Print” berfungsi dengan baik.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Mencari pesanan	Pesanan dapat dicari dengan mengisi kata kunci terkait.	Aplikasi akan mencari pesanan.	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Tombol	Menampilkan	Tombol “Previous”	<input checked="" type="checkbox"/> Diterima

Kegiatan Pengujian	Hasil yang diharapkan	Hasil pengujian	Kesimpulan
“Previous” dan “Next”	halaman sebelumnya atau selanjutnya.	dan “Next” berfungsi dengan baik.	[] Ditolak
Tampilan Feedback			
Menampilkan tampilan “Feedback”	Terdapat tabel yang berisi kritik dan saran yang telah dikirimkan oleh pelanggan.	Tombol pada menu “Feedback” berfungsi dengan baik.	[√] Diterima [] Ditolak
Mencari pesanan	Pesanan dapat dicari dengan mengisi kata kunci terkait.	Aplikasi akan mencari pesanan.	[√] Diterima [] Ditolak
Tombol “Previous” dan “Next”	Menampilkan halaman sebelumnya atau selanjutnya.	Tombol “Previous” dan “Next” berfungsi dengan baik.	[√] Diterima [] Ditolak

Berikut ini adalah pengujian yang dilakukan pada *database* untuk melihat data yang masuk dan tersimpan di *database*.

1. Data Pengguna

Data pengguna berhasil masuk ke dalam *database* di mana terdapat 1 *admin* sebagai *admin* dan masih terdapat 1 pengguna sebagai pelanggan. Data pengguna dapat dilihat pada Gambar 4.22 di halaman selanjutnya.

id_pengguna	email	username	password	role	terdaftar_pada
1	admin@	admin	admin	admin	2018-01-01
4	kevin.kindangen1996@gmail.com	kevin	kevin123	pelanggan	2020-03-10

Gambar 4.22 Data dari Tabel Pengguna

2. Data Pelanggan

Data Pelanggan berhasil masuk ke dalam *database* setelah melakukan proses *registrasi*. Data pelanggan dapat dilihat pada Gambar 4.23.

	id_pelanggan	nama_lengkap	alamat	nomor_telepon
<input type="checkbox"/>	4	Kevin Kindangen	Mahakeret	083131148292

Gambar 4.23 Data dari Tabel Pelanggan

3. Data Pesanan

Data pesanan berhasil masuk ke dalam *database* di mana pesanan akan berganti status setiap kali *admin* menerima pesanan tersebut. Data pesanan dapat dilihat pada Gambar 4.24.

	id_pesanan	id_pelanggan	jenis_produk	ukuran	material	foto_desain	status	tanggal	tanggal_selesai
<input type="checkbox"/>	1	4	Billboard	small		ad49d7011e96f96eeeb2643f6332d73a.jpg	menunggu	2020-03-10	0000-00-00
<input type="checkbox"/>	1	4	Billboard	small		ad49d7011e96f96eeeb2643f6332d73a.jpg	diproses	2020-03-10	0000-00-00
<input type="checkbox"/>	1	4	Billboard	small		ad49d7011e96f96eeeb2643f6332d73a.jpg	dibayar	2020-03-10	0000-00-00
<input type="checkbox"/>	1	4	Billboard	small		ad49d7011e96f96eeeb2643f6332d73a.jpg	pembuatan	2020-03-10	2020-03-10

Gambar 4.24 Data dari Tabel Pesanan

4. Data Tagihan

Data tagihan berhasil masuk ke dalam *database* agar pelanggan bisa langsung membayar tagihan tersebut. Data tagihan dapat dilihat pada Gambar 4.25 di halaman selanjutnya.

	id_tagihan	id_pesanan	harga	tanggal	batas_pembayaran
<input type="checkbox"/>	1	1	200000	0000-00-00	2020-03-10

Gambar 4.25 Data dari Tabel Tagihan

5. Data Pembayaran

Data pembayaran berhasil masuk ke dalam *database* agar *admin* dapat mengonfirmasi pesanan yang telah dibuat dan tinggal menunggu proses pembuatan sampai selesai. Data pembayaran dapat dilihat pada Gambar 4.26.



+ Options					
← T →					
	id_pembayaran	id_tagihan	bukti_pembayaran	keterangan	tanggal
<input type="checkbox"/>		1	1	fc9ebda8a52ceb7e40c8301d01640129.jpg	2020-03-10

↑ Check all With selected: Edit Copy Delete Export

Gambar 4.26 Data dari Tabel Pembayaran

6. Data *Feedback*

Data *feedback* dari pelanggan berhasil masuk ke dalam *database* sehingga *admin* bisa melihat kritik dan saran yang diberikan oleh pelanggan. Data *feedback* dapat dilihat pada Gambar 4.27.



+ Options						
← T →						
	id_feedback	id_pelanggan	subject	text	tanggal	
<input type="checkbox"/>		1	4	Pelayanan	pelayanan lebih di tingkatkan lagi	2020-03-10

↑ Check all With selected: Edit Copy Delete Export

Gambar 4.27 Data dari Tabel *Feedback*

7. Data *History*

Data *history* berhasil masuk ke dalam *database* sehingga *admin* bisa melihat riwayat pemesanan yang ada. Data *history* dapat dilihat pada Gambar 4.28.



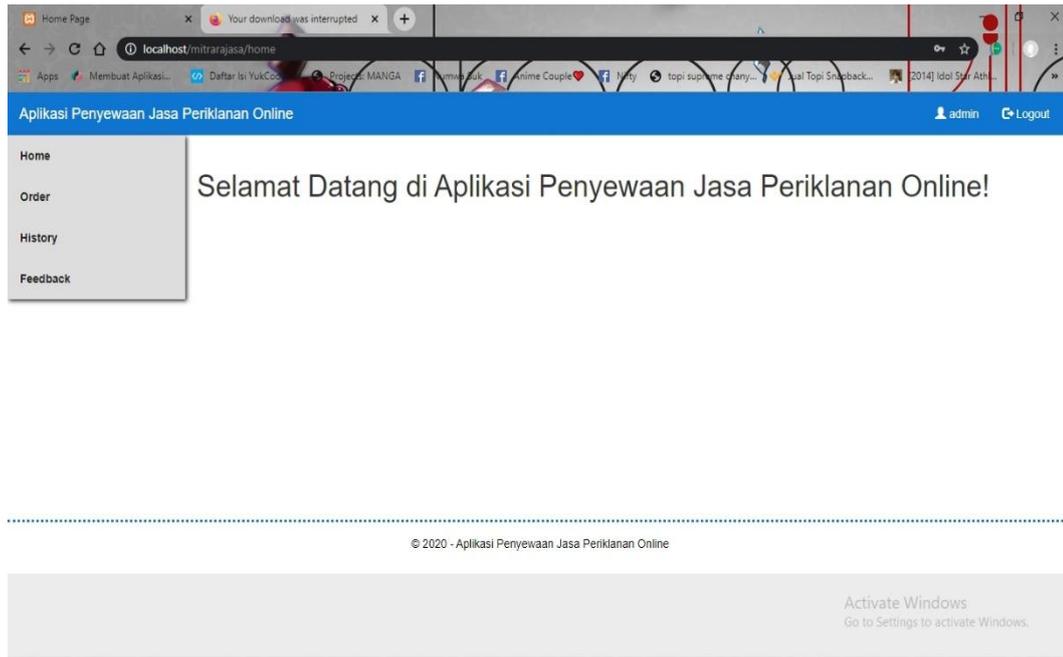
+ Options							
← T →							
	id_history	id_pengguna	jenis_produk	tanggal_pesanan	tanggal_selesai	status	
<input type="checkbox"/>		1	4	Billboard	2020-03-10	2020-03-10	pembuatan

↑ Check all With selected: Edit Copy Delete Export

Gambar 4.28 Data dari Tabel *History*

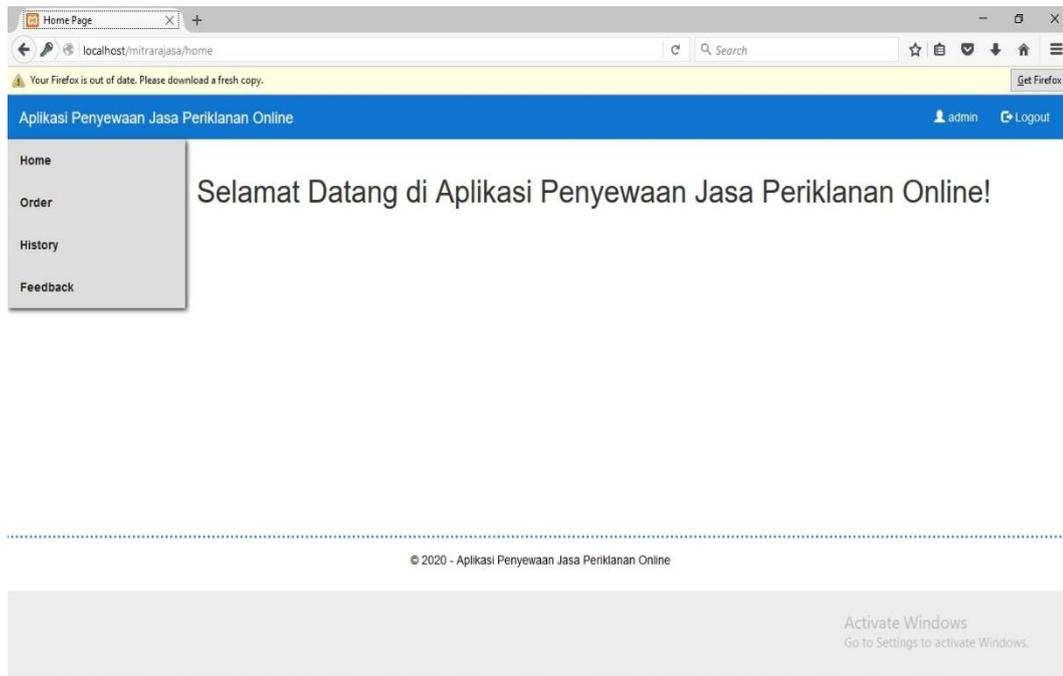
Berikut ini adalah pengujian yang telah dilakukan pada 3 *web browser*.

1. Google Chrome



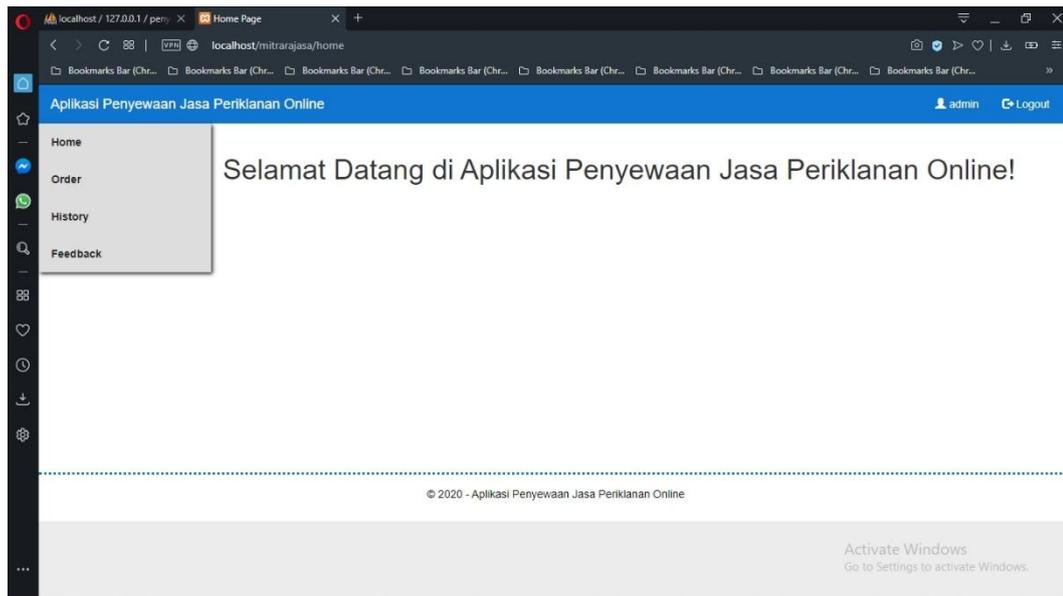
Gambar 4.29 Aplikasi Penyewaan Jasa Periklanan pada Chrome 80.0.3987.132

2. Firefox



Gambar 4.30 Aplikasi Penyewaan Jasa Periklanan pada Firefox 18.5.0.0

3. Opera



Gambar 4.31 Aplikasi Penyewaan Jasa Periklanan pada Opera 67.0.3575.53

5. ***User Acceptance Test***

Untuk *user acceptance test* yang dapat dilihat pada lampiran B.

6. ***Hasil Pengujian***

Berdasarkan pengujian aplikasi yang telah dilakukan, maka dapat disimpulkan bahwa semua fungsi telah berjalan dengan baik dan aplikasi penyewaan jasa periklanan berhasil dibuat. Pengujian pada *database* berjalan dengan baik semua data berhasil masuk ke dalam *database* sedangkan pengujian untuk ketiga *web browser* di atas tampilan dari aplikasi tidak terdapat perubahan dan bisa dijalankan dengan baik pada ketiga *web browser* tersebut.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Dari hasil pembuatan aplikasi penyewaan jasa periklanan pada PT. Rajasa Mitra Abadi, maka dapat diambil kesimpulan:

1. Aplikasi berhasil dibuat sesuai dengan syarat dan kebutuhan oleh PT. Rajasa Mitra Abadi.
2. Dengan adanya aplikasi ini, maka alur proses pemesanan perusahaan dapat kembali normal lagi.
3. Memberikan tempat bagi pelanggan dari PT. Rajasa Mitra Abadi untuk memberikan kritik dan saran bagi perusahaan.
4. Perusahaan dapat melihat keuntungan bulanan melalui laporan bulanan yang sudah dibuat.

5.2. Saran

Adapun saran-saran yang ada untuk pengembangan aplikasi ini adalah sebagai berikut:

1. Aplikasi dapat melakukan *automatic reject* pada konfirmasi pembayaran *admin* jika tidak ada *admin* yang sedang memantau penerimaan pemesanan dan konfirmasi pembayaran
2. Aplikasi dapat memberikan pemberitahuan ketika durasi dari penyewaan iklan *billboard* akan segera berakhir.

DAFTAR PUSTAKA

- [1] A. Mailoor, Interviewee, *Data Perusahaan*. [Interview]. 7 December 2017.
- [2] A. Kurniawan, "Rekayasa Perangkat Lunak Aplikasi Penjualan Pada Toko Story," vol. 1, no. 2, pp. 1-10, 2012.
- [3] "Smart Of Studio," 16 March 2014. [Online]. Available: <http://smartsoftstudio.com/pengertian-tentang-aplikasi-berbasis-web/>. [Accessed 10 January 2018].
- [4] Farhansyah, "Basis Data," in *Basis Data*, Bandung, Informatika Bandung, 2015, p. 2.
- [5] J. B. Satzinger, *Systems Analysis and Design In A Changing World*, Boston: Joe Sabatino, 2012.
- [6] R. Sianipar, "Apa itu CSS?," in *Pemrograman Javascript teori dan implementasi*, Bandung, Informatika Bandung, 2015, p. 483.
- [7] Andre, "Duniaikom," 1 October 2013. [Online]. Available: <https://www.duniaikom.com/tutorial-belajar-css-part-1-pengertian-css-apa-yang-dimaksud-dengan-css/>. [Accessed 5 February 2018].
- [8] W. Widyantoro, *Bootstrap*, Margotekno, 2014, p. 6.
- [9] Y. Murya, "XAMPP," in *Framework PHP yii2*, Jasakom, 2016.
- [10] P. Hidayatullah, "Code Igniter," in *Pemrograman Web*, Bandung, INFORMATIKA Bandung, 2017.
- [11] D. A. Hadi, "malasngoding," 12 January 2016. [Online]. Available: <https://www.malasngoding.com/pengertian-dan-cara-menggunakan-codeigniter/>. [Accessed 31 January 2018].
- [12] K. E. Kendall and J. E. Kendall, *System Analysis and Design*, New york: Pearson, 2013.
- [13] R. A. Sukamto and M. Shalahuddin, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*, Bandung: Modula, 2013.
- [14] Y. Sugiarti, *Analisis Dan Perancangan UML Generated VB.6*, Yogyakarta: Graha Ilmu, 2013.

LAMPIRAN A

HASIL WAWANCARA

Berikut ini adalah hasil wawancara yang dilakukan kepada Bapak Anri Mailoor.

PT. Rajasa Mitra Abadi didirikan tahun 2000 dan bergerak di bidang bengkel mobil *body repair*, *advertising*, dan rental alat berat, pada tahun 2007 PT. Berdasarkan hasil wawancara, masalah yang terjadi di PT. Rajasa Mitra Abadi adalah alur pemesanan yang dibuat, dikarenakan sebagian pelanggan lebih memilih melakukan pemesanan dengan cara langsung menghubungi pemilik perusahaan, dan alur pemesanannya menjadi tidak sesuai dengan *standard operational procedure* yang dibuat oleh perusahaan, yaitu melakukan pemesanan di kantor dan melakukan antrian lalu dilayani oleh karyawan yang menerima pesanan. Sehingga mengganggu proses pekerjaan perusahaan karena orang yang menerima pemesanan tidak sesuai dengan *job description*. Perusahaan juga memiliki kendala dalam hal mengumpulkan informasi tentang kualitas dari produk dan pelayanan yang mereka sediakan sehingga pelanggan tidak mempunyai tempat untuk memberikan kritik atau masukan atas jasa, produk, ataupun pelayanan yang disediakan, dan perusahaan juga tidak mempunyai laporan bulanan dari hasil pemesanan yang didapatkan.

LAMPIRAN B
USER ACCEPTANCE TEST

Nama Penguji Aplikasi:

Kasus Pengujian	Hasil	
	✓	X
Apakah aplikasi dibangun sesuai dengan permintaan pengguna?		
Apakah aplikasi dapat digunakan dengan baik?		
Apakah semua fungsi berjalan dengan baik?		
Apakah aplikasi dapat membantu <i>admin</i> dalam mengolah data pemesanan?		
Apakah aplikasi dapat membantu <i>admin</i> dalam melakukan transaksi pembayaran?		

Kritik dan saran:

.....

.....

.....

.....

.....

Manado, Manado 2017

PT. Rajasa Mitra Abadi

(.....)