

**RANCANG BANGUNG APLIKASI E-MARKET HIBURAN UNTUK  
KEBUTUHAN HAJATAN BERBASIS WEB**

**Tugas Akhir**

diajukan untuk memenuhi sebagian persyaratan mencapai  
gelar Ahli Madya pada jenjang Diploma III  
Jurusan Teknik Informatika



Oleh  
**AHMAD RIFA'I**  
**NIM. 1403004**

**JURUSAN TEKNIK INFORMATIKA  
POLITEKNIK NEGERI INDRAMAYU  
2017**

## **HALAMAN PENGESAHAN**

Tugas Akhir ini diajukan oleh :

Nama : Ahmad Rifa'i  
NIM : 1403004  
Program Studi : Teknik Informatika  
Judul : Rancang Bangun Aplikasi E-Market Hiburan untuk Kebutuhan Hajatan Berbasis Web  
  
Pembimbing : 1. A. Lubis G., S.Kom., M.Kom .....  
  
2. Iryanto, S.Si., M.Si .....

Telah berhasil dipertahankan dihadapan dewan penguji pada tanggal 14 Agustus 2017 dan diterima sebagai bagian yang diperlukan untuk memperoleh gelar Ahli Madya Program Studi Teknik Informatika, Jurusan Teknik Informatika, Politeknik Negeri Indramayu.

### **DEWAN PENGUJI**

Nama	Jabatan	Tandatangan	Tanggal
1. Moh. Yani, ST., MT., M.Sc	Ketua Penguji	.....	14 Agustus 2017
2. A. Lubis G., S.Kom., M.Kom	Sekertaris Penguji	.....	14 Agustus 2017
3. Munengsih S. B., S.Kom., M.Eng	Anggota	.....	14 Agustus 2017

Indramayu, 14 Agustus 2017

Ketua Jurusan Teknik Informatika

A. Sumarudin, S.Pd., MT., M.Sc

NIK. 09098630

## **PERNYATAAN KEASLIAN**

Saya menyatakan dengan sesungguhnya bahwa Tugas Akhir ini adalah asli hasil karya saya sendiri, dan sepanjang pengetahuan saya tidak terdapat karya atau pendapat yang pernah ditulis atau dipublikasikan oleh orang lain, kecuali yang secara tertulis disebutkan sumbernya dalam naskah dan dalam daftar pustaka.

Indramayu, 14 Agustus 2017  
Yang menyatakan,

AHMAD RIFA'I  
NIM. 1403004

## **ABSTRAK**

Hajatan merupakan acara yang sangat istimewa bagi setiap orang. Kemerahan hajatan merupakan suatu hal yang sering diadakan guna memberikan rasa sukacita dalam hajatan. Hiburan merupakan salah satu komponen dalam hajatan untuk memeriahkan suatu acara hajatan, tetapi penyewaan hiburan terkadang menjadi kendala seperti tanggal dan bulan yang tidak cocok serta harga sewa dan fasilitas yang sering berubah-ubah. E-*Marketing* merupakan salah satu bentuk media yang memasarkan layanan kepada pelanggan menggunakan media *web*. E-*Marketing* menjadi sangat dibutuhkan karena kemudahan akses dan murahnya penyebaran informasi. Mulai dari *smartphone* hingga perangkat *desktop* sekarang sudah mendukung kemudahan akses karena aplikasi dibangun dengan teknologi *responsive* desain. Aplikasi e-market hiburan untuk kebutuhan hajatan ini merupakan solusi bagi permasalahan sewa hiburan di suatu hajatan agar setiap orang yang akan hajatan tidak terlalu kesulitan dalam melakukan penyewaan suatu hiburan.

Kata Kunci: E-*Marketing*, Hajatan, Hiburan, Penyewaan, *Responsive*, *Web*.

## **ABSTRACT**

Celebrations are a very special event for everyone. Festival celebrations are often held to give a sense of joy in this festival. Entertainment is a component of the festival to enliven celebrations, but entertainment rentals sometimes become obstructions such as inappropriate dates and months and often change rates and ease of rent. E-Marketing is one of the forms of media marketing services to customers using web media. E-Marketing is very important because of the ease of access and cheapness of information dissemination. Starting from smartphones to desktop devices now supports easy access since applications are built with responsive design technology. The e-market entertainment application for this festive occasion is the solution to the entertainment problem in the rental of a celebration so that everyone who will celebrate is not too difficult in making entertainment rentals.

Keyword : Celebration, E-Marketing, Entertaiment, Rental, Responsive, Web

## **MOTO**

*“Jadilah manusia yang bermanfaat untuk manusia yang lain”*

**Man Jadda Wa Jada**

**KEEP MOVING FORWARD**

**~ Ahmad Rifa'i ~**

## **KATA PENGANTAR**

Assalamu'alaikum Warahmatullahi Wabarakatuh.

Puji dan syukur penulis panjatkan kehadirat Allah SWT. karena atas limpahan rahmat dan karunianya penulis dapat menyelesaikan tugas akhir ini dengan tepat waktu. Tidak lupa juga solawat serta salam semoga terus tercurahkan kepada baginda Nabi Muhammad SAW beserta keluarganya, para sahabatnya, hingga umatnya sampai akhir zaman.

Tugas akhir ini merupakan salah satu tugas yang wajib ditempuh oleh mahasiswa tingkat akhir. Penulisan laporan tugas akhir ini dibuat sebagai persyaratan utama untuk dapat dinyatakan lulus sebagai Ahli Madya Diploma 3.

Penulis menyadari bahwa dalam pengerjaan tugas akhir ini dapat terlaksana dan lancar berkat kerjasama, bantuan, arahan, dan dukungan dari berbagai pihak baik secara langsung maupun tidak langsung. Oleh karena itu penulis dengan senang hati menyampaikan terima kasih kepada :

1. Kedua Orang Tua penulis (Bapak Darita dan Ibu Titi Rohaeti), dan keluarga yang selalu menjadi motivasi dan semangat untuk dapat menyelesaikan laporan ini dengan tepat waktu.
2. Bapak Casiman Sukardi, ST., MT selaku Direktur Politeknik Negeri Indramayu.
3. Bapak A. Sumarudin, S.Pd., MT., M.Sc selaku Kepala Jurusan Teknik Informatika.
4. Bapak Ahmad Lubis Ghozali, S.Kom., M.Kom selaku Dosen Pembimbing I.
5. Bapak Iryanto, S.Si., M.Si selaku Dosen Pembimbing II.
6. Pegawai Dinas Kebudayaan dan Pariwisata yang telah membantu untuk mendapatkan data tentang kebudayaan yang ada di Indramayu.
7. Seseorang yang selalu menjadi semangat dan motivasi untuk terus berjuang, yang insyaAllah akan menjadi jodoh penulis.
8. Seluruh teman-teman kelas Teknik Informatika tahun angkatan 2014 yang telah banyak membantu dalam penyelesaian tugas akhir ini.
9. Sahabat kampung yang selalu menjadi teman disaat malas datang.

10. Semua pihak yang tidak dapat penulis sebutkan satu persatu yang telah terlibat dalam penyelesaian pembuatan tugas akhir ini.

Penulis telah berusaha semaksimal mungkin untuk dapat menyelesaikan laporan tugas akhir ini dengan baik dan benar, tetapi manusia merupakan tempatnya salah dan lupa. Di dalam dunia ini kesempurnaan hanyalah milik Allah SWT, manusia hanya bisa berikhtiar. Apabila terdapat kesalahan dalam penulisan tugas akhir ini penulis sangat menerima dengan senang hati apabila pembaca memberikan kritik dan saran yang membangun tentangnya. Kritik dan saran yang membangun akan menjadi bekal penulis guna kesempurnaan penulisan dikemudian hari. Penulis berharap semoga Laporan ini dapat bermanfaat bagi Almamater dan rekan mahasiswa lainnya.

Wassalamu'alaikum Warahmatullahi Wabarakatuh.

Indramayu, 14 Agustus 2017

Penulis

## DAFTAR ISI

<b>HALAMAN PENGESAHAN .....</b>	ii
<b>PERNYATAAN KEASLIAN .....</b>	iii
<b>ABSTRAK .....</b>	iv
<b>MOTTO .....</b>	vi
<b>KATA PENGANTAR .....</b>	vii
<b>DAFTAR ISI .....</b>	ix
<b>DAFTAR TABEL .....</b>	xiv
<b>DAFTAR GAMBAR .....</b>	xvi
<b>DAFTAR LAMPIRAN .....</b>	xx
<b>BAB I PENDAHULUAN .....</b>	1
1.1    Latar Belakang.....	1
1.2    Rumusan Masalah .....	2
1.3    Batasan Masalah.....	2
1.4    Tujuan Penelitian.....	2
1.5    Manfaat Penelitian.....	3
1.6    Sistematika Penulisan Laporan.....	3
<b>BAB II LANDASAN TEORI .....</b>	5
2.1    E- <i>Marketing</i> .....	5
2.1.1    Definisi E- <i>Marketing</i> .....	5
2.2    Manfaat E- <i>Marketing</i> .....	5
2.3    HTML ( <i>Hyper Text Markup Language</i> ).....	6
2.3.1    Definisi HTML ( <i>Hyper Text Markup Language</i> ) .....	6
2.3.2    Sejarah HTML ( <i>Hyper Text Markup Language</i> ) .....	6
2.3.3    Pengenalan Dasar HTML .....	8
2.4    CSS ( <i>Cascading Style Sheet</i> ).....	9
2.4.1    Jenis CSS .....	10
2.1.1    Pengenalan Dasar CSS.....	10
2.5    PHP .....	11
2.5.1    Pengenalan Dasar PHP .....	12
2.6    MySQL.....	13

2.6.1	Elemen SQL.....	14
2.6.2	Kelompok Pernyataan SQL .....	16
2.7	Java Script .....	19
2.7.1	Karakteristik Java Script .....	19
2.1.2	Pengenalan Dasar Java Script .....	20
2.8	JQuery.....	21
2.8.1	Fitur JQuery .....	22
2.9	Desain Web Responsif .....	22
2.9.1	Ciri Web Desain Responsif.....	22
2.10	Bootstrap.....	23
2.11	Balsamiq Mockups .....	23
2.12	Dreamweaver.....	24
2.12.1	Pengenalan Ruang Kerja Dreamweaver .....	25
2.13	Microsoft Visio.....	26
2.14	Corel Draw .....	27
2.15	Filezilla .....	27
2.15.1	Fitur Filezilla.....	27
2.16	Regexp ( <i>Regular Expression</i> ).....	28
2.16.1	Pengenalan Dasar Regexp.....	28
2.17	Normalisasi <i>Database</i> .....	32
2.17.1	Proses Normalisasi.....	33
2.17.2	Tahapan Normalisasi .....	33
2.18	UML ( <i>Unified Modeling Language</i> ) .....	35
2.18.1	Beberapa Diagram Dalam UML .....	35
2.18.2	<i>Use Case Diagram</i> .....	37
2.18.3	<i>Activity Diagram</i> .....	40
2.18.4	<i>Class Diagram</i> .....	41
2.19	<i>Black-Box Testing</i> .....	42
	<b>BAB III METODOLOGI PENELITIAN .....</b>	44
3.1	Metode Penelitian.....	44
3.1.1	Desain Penelitian .....	45
3.2	Metode Pengumpulan Data .....	46

3.2.1	Observasi.....	46
3.2.2	Studi Pustaka.....	46
3.3	Analisis Kebutuhan Sistem.....	46
3.3.1	Kebutuhan <i>Hardware</i> .....	47
3.3.2	Kebutuhan <i>Software</i> .....	47
3.4	Perancangan Sistem dengan UML .....	47
3.4.1	<i>Use Case Diagram</i> .....	48
3.4.2	<i>Activity Diagram</i> .....	50
3.4.3	<i>Class Diagram</i> .....	54
3.5	Perancangan Antar Muka .....	56
3.5.1	Rancangan Halaman Awal.....	56
3.5.2	Rancangan Halaman <i>List Penyedia Hiburan</i> .....	56
3.5.3	Rancangan Halaman Deskripsi Penyedia Hiburan .....	57
3.5.4	Rancangan Halaman Ulasan Penyedia Hiburan .....	58
3.5.5	Rancangan Halaman Riwayat Penyewaan Penyedia Hiburan .....	58
3.5.6	Rancangan Halaman Detail Informasi Hajatan.....	59
3.5.7	Rancangan Halaman <i>Register</i> .....	60
3.5.8	Rancangan Halaman <i>Login</i> .....	60
3.5.9	Rancangan Halaman Informasi <i>User</i> .....	60
3.5.10	Rancangan Halaman <i>List Hajatan User</i> .....	61
3.5.11	Rancangan Halaman <i>List Favorite</i> .....	61
3.5.12	Rancangan Halaman Pesan .....	62
3.5.13	Rancangan Halaman Ulasan .....	62
3.5.14	Rancangan Halaman Perbarui Informasi .....	63
3.5.15	Rancangan Halaman Ubah <i>Password</i> .....	63
3.5.16	Rancangan Halaman Ganti <i>Cover</i> .....	64
3.5.17	Rancangan Halaman Buat Hajatan .....	64
3.5.18	Rancangan Halaman Buka Usaha .....	65
3.5.19	Rancangan Halaman Sewa Hiburan.....	66
3.5.20	Rancangan <i>Popover Notification</i> .....	66
3.5.21	Rancangan <i>Popover Pesan</i> .....	66
3.5.22	Rancangan <i>Popover Jadwal Hajatan</i> .....	67

3.5.23	Rancangan <i>Popover</i> Profil .....	67
3.5.24	Rancangan <i>Popover</i> Menu.....	67
3.5.25	Rancangan Halaman Penyedia Hiburan.....	67
3.5.26	Rancangan Halaman Edit Informasi Penyedia Hiburan .....	68
3.5.27	Rancangan Halaman Tambah Foto Promosi.....	69
3.5.28	Rancangan Halaman Tambah Vidio Promosi.....	69
3.5.29	Rancangan Halaman Fans .....	70
3.5.30	Rancangan Halaman Jadwal .....	71
3.5.31	Rancangan Halaman Transaksi.....	71
3.5.32	Rancangan <i>Popover Notification</i> Penyedia Hiburan .....	72
3.5.33	Rancangan Halaman <i>Login</i> Admin dan Pegawai.....	72
3.5.34	Rancangan Halaman <i>Dashboard</i> Admin dan Pegawai.....	73
3.5.35	Rancangan Halaman Data Pegawai .....	73
3.5.36	Rancangan Halaman Data <i>User</i> .....	74
3.5.37	Rancangan Halaman Data <i>User</i> Penyedia Hiburan .....	74
3.6	Normalisasi <i>Database</i> .....	75
3.6.1	Tabel <i>Unnormal</i> .....	75
3.6.2	Tabel Normalisasi Kesatu (1NF) .....	75
3.6.3	Tabel Normalisasi Kedua (2NF) .....	76
3.6.4	Tabel Normalisasi Ketiga (3NF).....	76
3.7	Perancangan <i>Database</i> .....	77
3.7.1	Tabel detail_chatting.....	77
3.7.2	Tabel detail_transaksi .....	78
3.7.3	Tabel districts.....	78
3.7.4	Tabel extra_event.....	78
3.7.5	Tabel foto_promosi.....	79
3.7.6	Tabel log_admin .....	79
3.7.7	Tabel log_user.....	79
3.7.8	Tabel provinces.....	79
3.7.9	Tabel regencies .....	80
3.7.10	Tabel status_admin .....	80
3.7.11	Tabel status_log .....	80

3.7.12	Tabel status_user.....	80
3.7.13	Tabel tb_admin .....	81
3.7.14	Tabel tb_chatting .....	81
3.7.15	Tabel tb_favorite.....	81
3.7.16	Tabel tb_hajatan.....	82
3.7.17	Tabel tb_harga .....	82
3.7.18	Tabel tb_kategori .....	82
3.7.19	Tabel tb_penyedia.....	83
3.7.20	Tabel tb_transaksi .....	83
3.7.21	Tabel tb_ulasan .....	83
3.7.22	Tabel tb_user.....	84
3.7.23	Tabel tb_visitor .....	84
3.7.24	Tabel vidio_promosi .....	85
3.7.25	Tabel villages .....	85
3.7.26	Relasi <i>Database</i> .....	85
<b>BAB IV HASIL DAN PEMBAHASAN</b>	.....	87
4.1	Hasil.....	87
4.2	Pembahasan .....	87
4.2.1	Implementasi Desain <i>Interface</i> dan Program.....	96
4.2.2	Pengujian Aplikasi .....	114
4.2.3	Pengujian Melalui Kuisioner .....	122
4.2.4	Kesimpulan Pengujian .....	131
4.2.5	Kelebihan dan Kekurangan Aplikasi E-Market Hiburan untuk Kebutuhan Hajatan .....	132
<b>BAB V PENUTUP</b>	.....	134
5.1	Kesimpulan.....	134
5.2	Saran.....	134
<b>DAFTAR PUSTAKA</b>	.....	136
<b>DAFTAR LAMPIRAN</b>	.....	138

## DAFTAR TABEL

Tabel 2.1 Contoh Bentuk Data Tidak Normal ( <i>Unnormal</i> ) .....	33
Tabel 2.2 Contoh Bentuk Normal Pertama (1NF) .....	33
Tabel 2.3 Contoh Bentuk Normal Kedua (2NF) Tabel Mahasiswa.....	34
Tabel 2.4 Contoh Bentuk Normal Kedua (2NF) Tabel Mata Kuliah.....	34
Tabel 2.5 Contoh Bentuk Normal Kedua (2NF) Tabel Nilai.....	34
Tabel 2.6 Contoh Bentuk Normal Ketiga (3NF) Tabel Dosen .....	34
Tabel 2.7 Contoh Bentuk Normal Ketiga (3NF) Tabel Mata Kuliah .....	35
Tabel 3.1 Daftar Kebutuhan <i>Hardware</i> Laptop .....	47
Tabel 3.2 Daftar Kebutuhan <i>Software</i> Laptop .....	47
Tabel 3.3 Tabel Penyewaan Hiburan untuk Hajatan <i>Unnormal</i> .....	75
Tabel 3.4 Tabel Penyewaan Hiburan untuk Hajatan Normalisasi kesatu (1NF) ..	76
Tabel 3.5 Tabel <i>User</i> .....	76
Tabel 3.6 Tabel Penyewaan .....	76
Tabel 3.7 Tabel <i>User</i> .....	76
Tabel 3.8 Tabel Hajatan .....	77
Tabel 3.9 Tabel Penyedia.....	77
Tabel 3.10 Tabel Transaksi .....	77
Tabel 3.11 Tabel detail_chatting.....	78
Tabel 3.12 Tabel detail_transaksi .....	78
Tabel 3.13 Tabel districts.....	78
Tabel 3.14 Tabel extra_event.....	78
Tabel 3.15 Tabel foto_promosi .....	79
Tabel 3.16 Tabel log_admin .....	79
Tabel 3.17 Tabel log_user.....	79
Tabel 3.18 Tabel provinces.....	80
Tabel 3.19 Tabel regencies .....	80
Tabel 3.20 Tabel staus_admin.....	80
Tabel 3.21 Tabel status_log .....	80
Tabel 3.22 Tabel status_user.....	81
Tabel 3.23 Tabel tb_admin .....	81

Tabel 3.24 Tabel tb_chatting.....	81
Tabel 3.25 Tabel tb_favorite .....	82
Tabel 3.26 Tabel tb_hajatan.....	82
Tabel 3.27 Tabel tb_harga.....	82
Tabel 3.28 Tabel tb_kategori .....	82
Tabel 3.29 Tabel tb_penyedia .....	83
Tabel 3.30 Tabel tb_transaksi .....	83
Tabel 3.31 Tabel tb_ulasan .....	83
Tabel 3.32 Tabel tb_user.....	84
Tabel 3.33 Tabel tb_visitor .....	84
Tabel 3.34 Tabel vudio_promosi .....	85
Tabel 3.35 Tabel villages .....	85
Tabel 4.1 Tabel Hasil Pengujian dengan <i>Black-Box</i> .....	115
Tabel 4.2 Tabel Hasil Pengujian dengan <i>Black-Box</i> (Lanjutan).....	116
Tabel 4.3 Tabel Hasil Pengujian dengan <i>Black-Box</i> (Lanjutan).....	117
Tabel 4.4 Tabel Hasil Pengujian dengan <i>Black-Box</i> (Lanjutan).....	118
Tabel 4.5 Tabel Hasil Pengujian dengan <i>Black-Box</i> (Lanjutan).....	119
Tabel 4.6 Tabel Hasil Pengujian dengan <i>Black-Box</i> (Lanjutan).....	120
Tabel 4.7 Tabel Hasil Pengujian dengan <i>Black-Box</i> (Lanjutan).....	121
Tabel 4.8 Tabel Hasil Pengujian dengan <i>Black-Box</i> (Lanjutan).....	122
Tabel 4.9 Tabel Pertanyaan Kuisisioner .....	123
Tabel 4.10 Hasil Jawaban Kuisisioner.....	124
Tabel 4.11 Keterangan Nilai .....	125
Tabel 4.12 Hasil Perhitungan Tampilan Aplikasi .....	125
Tabel 4.13 Hasil Perhitungan Tampilan Aplikasi (Lanjutan) .....	126
Tabel 4.14 Hasil Perhitungan Penggunaan Aplikasi.....	126
Tabel 4.15 Hasil Perhitungan Penggunaan Aplikasi (Lanjutan) .....	127
Tabel 4.16 Hasil Perhitungan Kegunaan Aplikasi .....	127
Tabel 4.17 Hasil Perhitungan Kegunaan Aplikasi (Lanjutan) .....	128
Tabel 4.18 Hasil Perhitungan Kualitas Aplikasi .....	128
Tabel 4.19 Rekapitulasi Hasil Kuisisioner.....	129

## DAFTAR GAMBAR

Gambar 2.1 Penulisan Elemen HTML yang Tumpang Tindih .....	8
Gambar 2.2 Penulisan Elemen HTML yang Normal.....	9
Gambar 2.3 Struktur CSS.....	10
Gambar 2.4 Cara Kerja PHP .....	12
Gambar 2.5 Komponen <i>User Interface</i> Balsamiq Mockups .....	24
Gambar 2.6 Ruang Kerja Dreamweaver .....	25
Gambar 2.7 <i>Use Case</i> Sederhana .....	37
Gambar 2.8 <i>Use Case</i> Tanpa Generalisasi .....	38
Gambar 2.9 <i>Use Case</i> dengan Generalisasi .....	39
Gambar 2.10 <i>Use Case</i> dengan Ekstensi.....	39
Gambar 2.11 <i>Use Case</i> dengan Inklusi .....	40
Gambar 2.12 Contoh <i>Activity Diagram</i> Airport Check-In .....	41
Gambar 2.13 Bentuk Umum <i>Class Diagram</i> .....	42
Gambar 3.1 Desain Penelitian.....	45
Gambar 3.2 <i>Use Case Diagram</i> .....	49
Gambar 3.3 <i>Activity Diagram</i> Login User .....	50
Gambar 3.4 <i>Activity Diagram</i> Register User Baru .....	51
Gambar 3.5 <i>Activity Diagram</i> Buat Hajatan .....	51
Gambar 3.6 <i>Activity Diagram</i> Buka Usaha .....	52
Gambar 3.7 <i>Activity Diagram</i> Sewa Hiburan .....	53
Gambar 3.8 <i>Activity Diagram</i> Tambah Jadwal .....	54
Gambar 3.9 <i>Class Diagram</i> .....	55
Gambar 3.10 Rancangan Halaman Awal .....	56
Gambar 3.11 Rancangan Halaman <i>List</i> Penyedia Hiburan .....	57
Gambar 3.12 Rancangan Halaman Deskripsi Penyedia Hiburan.....	57
Gambar 3.13 Rancangan Halaman Ulasan Penyedia Hiburan .....	58
Gambar 3.14 Rancangan Halaman Riwayat Penyewaan Penyedia Hiburan .....	59
Gambar 3.15 Rancangan Halaman Detail Informasi Hajatan .....	59
Gambar 3.16 Rancangan Halaman <i>Register</i> .....	60
Gambar 3.17 Rancangan Halaman <i>Login</i> .....	60

Gambar 3.18 Rancangan Halaman Informasi <i>User</i> .....	61
Gambar 3.19 Rancangan Halaman <i>List Hajatan User</i> .....	61
Gambar 3.20 Rancangan Halaman <i>List Favorite</i> .....	62
Gambar 3.21 Rancangan Halaman Pesan .....	62
Gambar 3.22 Rancangan Halaman Ulasan.....	63
Gambar 3.23 Rancangan Halaman Perbarui Informasi.....	63
Gambar 3.24 Rancangan Halaman Ubah <i>Password</i> .....	64
Gambar 3.25 Rancangan Halaman Ganti <i>Cover</i> .....	64
Gambar 3.26 Rancangan Halaman Buat Hajatan.....	65
Gambar 3.27 Rancangan Halaman Buka Usaha .....	65
Gambar 3.28 Rancangan Halaman Sewa Hiburan .....	66
Gambar 3.29 Rancangan <i>Popover Notification</i> .....	66
Gambar 3.30 Rancangan <i>Popover Pesan</i> .....	66
Gambar 3.31 Rancangan <i>Popover Jadwal Hajatan</i> .....	67
Gambar 3.32 Rancangan <i>Popover Profil</i> .....	67
Gambar 3.33 Rancangan <i>Popover Menu</i> .....	67
Gambar 3.34 Rancangan Halaman Penyedia Hiburan.....	68
Gambar 3.35 Rancangan Halaman Edit Informasi Penyedia Hiburan.....	68
Gambar 3.36 Rancangan Halaman Tambah Foto Promosi .....	69
Gambar 3.37 Rancangan Halaman Vidio Promosi .....	70
Gambar 3.38 Rancangan Halaman Fans .....	70
Gambar 3.39 Rancangan Halaman Jadwal.....	71
Gambar 3.40 Rancangan Halaman Transaksi .....	72
Gambar 3.41 Rancangan <i>Popover Notification</i> Penyedia Hiburan.....	72
Gambar 3.42 Rancangan Halaman Login Admin dan Pegawai.....	73
Gambar 3.43 Rancangan Halaman <i>Dashboard</i> Admin dan Pegawai .....	73
Gambar 3.44 Rancangan Halaman Data Pegawai.....	74
Gambar 3.45 Rancangan Halaman Data <i>User</i> .....	74
Gambar 3.46 Rancangan Halaman Data <i>User</i> Penyedia Hiburan.....	75
Gambar 3.47 Relasi <i>Database</i> .....	86
Gambar 4.1 Tampilan Halaman Awal .....	96
Gambar 4.2 Tampilan Halaman <i>List Penyedia Hiburan</i> .....	97

Gambar 4.3 Tampilan Halaman Deskripsi Penyedia Hiburan .....	98
Gambar 4.4 Tampilan Halaman Ulasan Penyedia Hiburan .....	99
Gambar 4.5 Tampilan Halaman Riwayat Penyewaan Penyedia Hiburan .....	99
Gambar 4.6 Tampilan Halaman Detail Informasi Hajatan .....	100
Gambar 4.7 Tampilan Halaman <i>Register</i> .....	100
Gambar 4.8 Tampilan Halaman <i>Login</i> .....	101
Gambar 4.9 Tampilan Halaman Informasi <i>User</i> .....	101
Gambar 4.10 Tampilan Halaman <i>List</i> Hajatan <i>User</i> .....	102
Gambar 4.11 Tampilan Halaman <i>List Favorite</i> .....	102
Gambar 4.12 Tampilan Halaman Pesan.....	103
Gambar 4.13 Tampilan Halaman Ulasan .....	103
Gambar 4.14 Tampilan Halaman Perbarui Informasi .....	104
Gambar 4.15 Tampilan Halaman Ubah <i>Password</i> .....	104
Gambar 4.16 Tampilan Halaman Ganti <i>Cover</i> .....	105
Gambar 4.17 Tampilan Halaman Buat Hajatan .....	105
Gambar 4.18 Tampilan Halaman Buka Usaha.....	106
Gambar 4.19 Tampilan Halaman Sewa Hiburan .....	106
Gambar 4.20 Tampilan <i>Popover Notification</i> .....	107
Gambar 4.21 Tampilan <i>Popover</i> Pesan.....	107
Gambar 4.22 Tampilan <i>Popover</i> Jadwal Hajatan .....	107
Gambar 4.23 Tampilan <i>Popover</i> Profil.....	107
Gambar 4.24 Tampilan <i>Popover</i> Menu.....	108
Gambar 4.25 Tampilan Halaman Penyedia Hiburan .....	108
Gambar 4.26 Tampilan Halaman Edit Informasi Penyedia Hiburan .....	109
Gambar 4.27 Tampilan Halaman Tambah Foto Promosi .....	109
Gambar 4.28 Tampilan Halaman Tambah Vidio Promosi.....	110
Gambar 4.29 Tampilan Halaman Fans.....	110
Gambar 4.30 Tampilan Halaman Jadwal .....	111
Gambar 4.31 Tampilan Halaman Transaksi.....	111
Gambar 4.32 Tampilan <i>Popover Notification</i> Penyedia Hiburan .....	112
Gambar 4.33 Tampilan Halaman <i>Login</i> Admin dan Pegawai .....	112
Gambar 4.34 Tampilan Halaman <i>Dashboard</i> Admin dan Pegawai.....	113

Gambar 4.35 Tampilan Halaman Data Pegawai .....	113
Gambar 4.36 Tampilan Halaman Data <i>User</i> .....	114
Gambar 4.37 Tampilan Halaman Data <i>User</i> Penyedia Hiburan .....	114
Gambar 4.38 <i>Pie Chart</i> Tampilan Aplikasi .....	126
Gambar 4.39 <i>Pie Chart</i> Penggunaan Aplikasi .....	127
Gambar 4.40 <i>Pie Chart</i> Kegunaan Aplikasi .....	128
Gambar 4.41 <i>Pie Chart</i> Kualitas Aplikasi .....	129
Gambar 4.42 Hasil Pengujian Melalui Kuisioner .....	131

## **DAFTAR LAMPIRAN**

LAMPIRAN 1 KUISIONER DATA *REQUIREMENT*

LAMPIRAN 2 KUISIONER PENGUJIAN APLIKASI

LAMPIRAN 3 DATA KESENIAN KABUPATEN INDRAMAYU

LAMPIRAN 4 SINTAKS PROGRAM

LAMPIRAN 5 BIODATA PENULIS

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Kemajuan teknologi sekarang ini semakin terus berkembang. Informasi yang *realtime* dalam lintas dunia pun sudah dapat diakses dengan mudah menggunakan komputer, *smartphone* dan lain sebagainya yang terhubung dengan *internet*. Kemudahan akses informasi tersebut terus dikembangkan sehingga memunculkan ide-ide baru untuk membuat sistem pengolahan informasi.

Hajatan merupakan acara yang sangat istimewa bagi siapapun. Kemerahan hajatan merupakan unsur penting dalam hajatan. Menyewa tempat, hiburan, dan lain sebagainya adalah suatu kebiasaan yang biasa dilakukan oleh pihak yang mempunyai hajatan agar tamu yang hadir dapat lebih bersukacita merayakan hajatan tersebut.

Sebaliknya, melakukan penyewaan tempat, hiburan, dan lain sebagainya merupakan hal yang cukup merepotkan yang harus dilakukan oleh pihak yang mempunyai hajatan. Tanggal dan bulan yang tidak cocok, harga dan fasilitas yang sering berubah-ubah merupakan hal-hal yang sering kali menjadi hambatan antara penyedia hiburan dengan pihak yang mempunyai hajatan, sehingga itu membuat ketidaksesuaian dengan keinginan dan harapan dari pihak yang mempunyai hajatan.

Pengelolaan informasi yang membantu antara pihak yang mempunyai hajatan dengan pihak penyedia hiburan sangat diperlukan. Dengan dibuatnya pengelolaan informasi tersebut diharapkan dapat membantu untuk mempermudah pihak yang mempunyai hajatan untuk menyelenggarakan hajatannya. Oleh karena itu penulis berharap dengan mengambil judul “**RANCANG BANGUN APLIKASI E-MARKET HIBURAN UNTUK KEBUTUHAN HAJATAN BERBASIS WEB**” merupakan upaya untuk memudahkan penyewaan hiburan untuk suatu acara hajatan.

## **1.2 Rumusan Masalah**

Dari latar belakang yang telah dijelaskan, penulis mendapatkan beberapa rumusan masalah sebagai berikut :

1. Bagaimana merancang aplikasi e-market hiburan.
2. Bagaimana membuat aplikasi e-market hiburan yang bisa memudahkan pengguna.
3. Bagaimana mengimplementasikan proses transaksi ke dalam sistem aplikasi.
4. Bagaimana pengujian aplikasi yang dapat mengukur keberhasilan dari hasil implementasi.

## **1.3 Batasan Masalah**

Dari topik dan rumusan masalah yang telah dituliskan didapatkan beberapa batasan masalah sebagai berikut :

1. Aplikasi ini dibangun dengan bahasa pemrograman PHP *Native*.
2. Aplikasi ini dibuat dalam bentuk RWD (*Responsive Web Desain*).
3. Aplikasi ini diterapkan berbasis *online*.
4. Jadwal penyewaan hiburan ditampilkan dalam bentuk kalender sebagai informasi untuk *user* yang akan menyewa.
5. Media komunikasi dibuat dalam bentuk *chatting*.
6. Transaksi dalam sistem aplikasi e-market hiburan ini hanya sampai transaksi manual (pertemuan antara pihak hajatan dengan pihak penyewa).

## **1.4 Tujuan Penelitian**

Beberapa tujuan dari pembuatan sistem aplikasi ini adalah sebagai berikut :

1. Menyediakan tempat promosi untuk para penyedia hiburan.
2. Menyediakan tempat komunikasi antar *user* yang terkait dengan hajatan.
3. Memudahkan pemesanan hiburan untuk hajatan.
4. Terbentuknya sistem informasi hajatan dan hiburan.
5. Terbentuknya aplikasi yang baik untuk digunakan oleh *user*.

## **1.5 Manfaat Penelitian**

Beberapa manfaat dari sistem aplikasi ini adalah sebagai berikut :

1. Penulis mendapatkan pengetahuan tentang bagaimana cara membuat aplikasi e-market
2. Memberikan pengetahuan kepada mahasiswa yang ingin mengetahui contoh sistem informasi dan aplikasi e-market.
3. Penyedia hiburan dapat mempromosikan hiburan-hiburannya.
4. Pihak hajatan dengan pihak penyedia hiburan dapat melakukan komunikasi.
5. Pihak hajatan dapat melakukan penyewaan hiburan-hiburan untuk hajatannya.
6. Pengunjung dapat mengetahui informasi-informasi terkait hiburan dan hajatan.
7. Dapat dijadikan data referensi untuk dinas kebudayaan dan pariwisata pada masing-masing kabupaten/kota.

## **1.6 Sistematika Penulisan Laporan**

Dalam sistematika penulisan laporan tugas akhir ini dibagi menjadi beberapa bagian bab, di mana pada setiap bagian membuat pembahasan yang berbeda namun masih memiliki keterkaitan satu sama lain. Untuk memudahkan penulisan laporan tugas akhir ini penulis menguraikan dan mengurutkan bab-babnya secara sistematis. Adapun urutan penulisannya sebagai berikut.

### **1. BAB I PENDAHULUAN**

Bab pendahuluan berisi latar belakang masalah, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, sistematika penulisan laporan.

### **2. BAB II LANDASAN TEORI**

Bab ini menguraikan tentang teori dasar yang berkaitan secara langsung dengan masalah yang diteliti. Usahakan mengambil referensi dari buku-buku atau sumber lainnya yang masih baru. Pada bagian ini juga bisa diuraikan penelitian-penelitian lain yang terkait dengan penelitian anda.

### **3. BAB III METODE PENELITIAN**

Bagian ini berisi penjelasan tentang tahapan dan metode penelitian yang ditempuh untuk mencapai tujuan yang ditetapkan. Diagram alir (*flow chart*) harus dibuat untuk menjelaskan tahapan tersebut.

#### **4. BAB IV HASIL DAN PEMBAHASAN**

Bab ini membahas hasil-hasil dari tahapan penelitian, dari tahap analisis, desain, implementasi desain, hasil pengujian dan implementasinya, berupa penjelasan teoritik, baik secara kualitatif, kuantitatif, atau secara statistik. Di samping itu, sebaiknya hasil penelitian juga dibandingkan dengan hasil penelitian terdahulu yang sejenis atau keadaan sebelumnya.

#### **5. BAB V PENUTUP**

Bab ini berisikan kesimpulan dan saran yang didapatkan atau diperoleh dari hasil penelitian atau pembuatan aplikasi

## **BAB II** **LANDASAN TEORI**

### **2.1 E- *Marketing***

Produk aplikasi dari tugas akhir ini merupakan salah satu jenis dari aplikasi *E-Marketing*. *E-Marketing* atau *Electronic-Marketing* adalah bagian dari e-Business yang memanfaatkan medium elektronik untuk melakukan aktivitas marketing dalam upaya mencapai tujuan *marketing* (Dwiputra, 2014). Penjelasan tentang *E-Marketing* akan dibahas pada subbab berikut.

#### **2.1.1 Definisi *E-Marketing***

Menurut Chen-Ling dan Lie (2006) seperti yang dikutip dari Tandusan (2015), *e-marketing* adalah proses memasarkan produk dan layanan kepada pelanggan dengan menggunakan media *web*. Promosi, iklan, transaksi, dan pembayaran dapat dilakukan melalui halaman *web*. Pengguna *internet marketing* dapat dengan mudah mengakses informasi dimana saja dengan perangkat yang terhubung ke internet.

Menurut Mohammed dkk. (2003) seperti yang dikutip dari Tandusan (2015), *internet marketing* adalah sebuah proses untuk membangun dan memelihara hubungan dengan pelanggan melalui kegiatan secara online sebagai sarana untuk pertukaran pendapat, produk, dan jasa sehingga dapat mencapai tujuan bersama kedua kelompok.

Menurut Widodo (2002) seperti yang dikutip dari Tandusan (2015), mengemukakan bahwa *internet marketing* memiliki lima keuntungan besar terhadap perusahaan yang menggunakannya, yaitu : Perusahaan kecil dan perusahaan besar dapat menggunakannya, Tidak memiliki batasan ruangan untuk beriklan jika dibandingkan dengan media cetak dan media penyiaran, Akses dan pencarian keterangan sangat cepat jika dibandingkan dengan surat kilat atau bahkan *fax*, Situsnya dapat dikunjungi oleh siapapun, dimanapun, dan kapanpun dan Belanja dapat dilakukan lebih cepat dan dapat dilakukan sendiri.

### **2.2 Manfaat *E-Marketing***

Periklanan dan pemasaran secara online memberikan manfaat yang sangat besar bagi konsumen dan pemasar. Kotler dan Armstrong (2008) seperti dikutip

dari Tandusan (2015) menjabarkan beberapa manfaat pemasaran online yaitu Manfaat untuk konsumen (*Convenient Easy and private, Information, Interactive and immediate*) dan Manfaat untuk pemasar (Pemasar online baik untuk membangun hubungan dengan pelanggan, Pemasaran online mengurangi biaya dan meningkatkan efisiensi, Pemasaran online menawarkan keluwesan yang lebih baik dan Pemasaran online merupakan medium global.

### **2.3 HTML (*Hyper Text Markup Language*)**

HTML merupakan script yang dominan digunakan dalam pembuatan aplikasi e-market hiburan untuk kebutuhan hajatan ini, karena memang aplikasi ini dibangun berbasis web. Penjelasan-penjelas lengkap tentang HTML akan dibahas pada subbab berikut.

#### **2.3.1 Definisi HTML (*Hyper Text Markup Language*)**

HTML (*Hyper Text Markup Language*) dikenal sebagai bahasa kode berbasis teks untuk membuat sebuah halaman web, keberadaannya dikenal dengan adanya ekstensi \*.htm atau \*.html. HTML merupakan suatu bahasa dari *website* (www) yang dipergunakan untuk menyusun dan membentuk dokumen agar dapat ditampilkan pada program *browser*. Ketika *user* mengakses *web*, maka ia mengakses dokumen seseorang yang ditulis dengan menggunakan format HTML. Dapat disimpulkan bahwa HTML merupakan protokol yang digunakan untuk transfer data atau dokumen dari *web server* ke *browser* (Pranata dkk, 2015).

#### **2.3.2 Sejarah HTML (*Hyper Text Markup Language*)**

Dari waktu ke waktu HTML terus berkembang dimulai dari HTML 1.0 hingga sekarang HTML 5. Beberapa penjelasan tentang perkembangan versi HTML menurut Shannon (2012) adalah sebagai berikut:

##### **1. HTML 1.0**

HTML 1.0 merupakan versi pertama yang dikenalkan ke dunia. Tidak banyak orang terlibat dalam pembuatan *website* pada waktu itu dan bahasa tersebut masih sangat terbatas.

##### **2. HTML 2.0**

HTML 2.0 merupakan versi yang di dalamnya adalah versi HTML 1.0 tetapi terdapat penambahan beberapa fitur baru yang digabungkan. HTML 2.0 telah

menjadi standar desain *website* sampai bulan Januari 1997 dan telah mendefinisikan banyak inti fungsi utama HTML pada waktu itu.

### **3. HTML 3.0**

HTML 3.0 merupakan versi yang dikembangkan karena ditemukannya beberapa masalah-masalah yang dihadapi oleh *browser market*. Sehingga HTML *working group* yang dipimpin oleh Dave Raggett mengenalkan sebuah bahasa HTML baru yaitu HTML 3.0. Di versi yang baru terdapat banyak peningkatan kemampuan *script* HTML itu sendiri dan yakin akan lebih baik lagi untuk para *webmaster* mendesain suatu halaman-halaman *web*.

### **4. HTML 3.2**

*Tag-tag* khusus dari tiap-tiap *browser* mulai bermunculan dan ini mulai meningkatkan untuk dibuatkan standar yang jelas. Pada waktu itu W3C di tahun 1994 telah menemukan standarisasi bahasa dan mengembangkannya. Pada waktu penggeraan pertama dinamakan WILBUR dan kemudian menjadi HTML 3.2.

### **5. HTML 4.01**

HTML 4.0 telah menjadi perubahan yang besar dari standarisasi HTML. Dan ini merupakan peningkatan terakhir dari HTML yang lama. Pengembangan sebelumnya dinamakan COUGAR. Banyak fungsi-fungsi baru yang dibawa dari versi HTML 3.0. Dan pada versi baru HTML ini telah mendukung bahasa CSS (*Cascading Style Sheet*).

HTML 4.0 telah menjadi rekomendasi dari W3C di bulan Desember 1997 dan menjadi standar resmi di bulan April 1998.

### **6. XHTML 1.0**

Di abad ke-21 W3C mempersoalkan spesifikasi dari XHTML 1.0 sebagai sebuah rekomendasi. Sejak 26 Januari 2000 XHTML telah bersatu menjadi standar dengan HTML 4.01. XHTML beranjak dari spesifikasi yang baru bekerja dan sepenuhnya merupakan cabang dari HTML.

### **7. HTML 5**

Setelah HTML 4.01 dan XHTML 1.0 orang-orang yang selalu mengontrol arah dari berjalannya HTML mulai bekerja untuk membuat proposal baru untuk XHTML 2. Dan pada waktu yang sama, seseorang pengembang *web* yang

cerdas telah berinovasi membagi fungsi-fungsi yang baru ke dalam *website* dan *browser*. Bagian dari XHTML 2 mulai terlihat membosankan dan tidak realistik, dan ini menjadi langkah yang bagus untuk menunjukkan apa yang diperlukan.

HTML 5 didesain untuk *web*, untuk sekarang dan selanjutnya. Ini menjadi spesifikasi bahwa mereka akan bekerja untuk di akhir dekade berikutnya, sehingga proses pengembangannya relatif lambat dan perlu pertimbangan. Banyak bagian yang akan dikenali dan juga banyak elemen, atribut, serta fitur yang baru.

### 2.3.3 Pengenalan Dasar HTML

Di dalam *script* HTML terdapat *tag*, elemen, dan atribut. Dokumen HTML disimpan dalam format teks reguler dan mengandung *tag-tag* yang memerintahkan *web browser* untuk mengeksekusi perintah-perintah yang dispesifikasikan.

#### 1. Tag HTML

*Tag* adalah teks khusus (*markup*) berupa dua karakter “<” dan “>” (tanpa tanda kutip) yang merupakan cara untuk memberitahu *web browser* bagaimana suatu teks ditampilkan (Rahardi, 2014).

#### 2. Elemen HTML

Menurut Rahardi (2014) Elemen adalah isi dari *tag* yang berada di antara *tag* pembuka dan *tag* penutup. Elemen terdiri atas tiga bagian, yaitu *tag* pembuka, isi, dan *tag* penutup. Dalam penulisan elemen HTML harus diperhatikan jangan sampai saling tumpah tindih seperti pada Gambar 2.1 dan penulisan elemen HTML yang benar (tidak tumpang tindih) terdapat pada Gambar 2.2.



Gambar 2.1 Penulisan Elemen HTML yang Tumpang Tindih

```
<p>
<b>
.....
</b>
</p>
```

Gambar 2.2 Penulisan Elemen HTML yang Normal

Pada Gambar 2.1 disebut penulisan elemen HTML yang tumpang tindih karena barisan elemen `<p>` yang pertama, pasangan akhirnya adalah `</b>`. penulisan *script* yang seperti itu akan mengakibatkan *script* tidak berfungsi.

### 3. Atribut HTML

Atribut adalah informasi tambahan yang diberikan kepada *tag*. Informasi ini bisa berupa instruksi untuk warna dari *text*, besar huruf dari *text*, dan lain sebagainya. Setiap atribut juga memiliki pasangan nama dan nilai (*value*), dan ditulis dengan `name="value"`. *Value* diapit tanda kutip, boleh tanda kutip satu (‘) atau dua (“) (Rahardi, 2017).

Penulisan atribut harus berada di antara *tag* pembuka dan penutup, contoh untuk membuat halaman *web* menjadi warna hitam dan tulisan menjadi warna kuning adalah sebagai berikut.

```
<html>
<head>
<title>Judul Dokumen HTML</title>
</head>
<body>
<body bgcolor="black" text="yellow">
    Isi Dokumen (Belajar HTML)
</body>
</html>
```

### 2.4 CSS (*Cascading Style Sheet*)

Menurut Wiswakarma dalam Nina Rahayu (2013) seperti yang dikutip oleh Susanto (2015), “CSS (*Cascading Style Sheet*) adalah salah satu bahasa pemrograman desain *web* (*style sheet language*) yang mengontrol format tampilan sebuah halaman *web* yang ditulis dengan menggunakan bahasa penanda (*make up language*)”.

Menurut Agus Saputra dkk (2013) seperti yang dikutip oleh Susanto (2015), “CSS atau yang memiliki kepanjangan *Cascading Style Sheet*, merupakan suatu bahasa pemrograman web yang digunakan untuk mengendalikan dan membangun berbagai komponen dalam web sehingga tampilan web akan lebih rapi, terstruktur, dan seragam”.

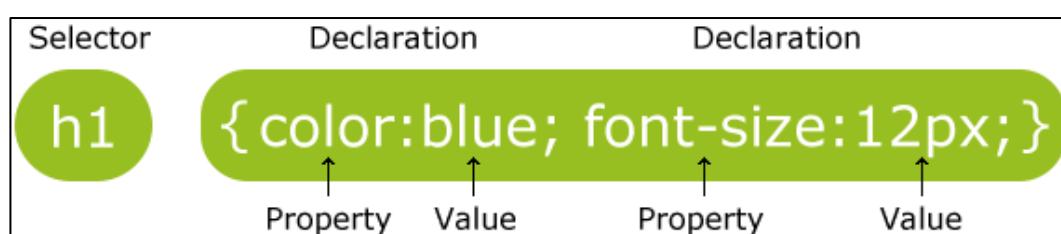
#### 2.4.1 Jenis CSS

CSS memiliki dua jenis sifat yaitu *internal* dan eksternal. Jenis tersebut adalah sebagai berikut.

1. **Internal**, CSS yang dibuat disisipkan ke dalam *file* kode *HTML*. Cara ini akan merepotkan jika membuat *file* lain dengan metode yang sama, artinya kode yang sama akan terus diulang-ulang.
2. **Eksternal**, pembuatan kode CSS dan HTML dilakukan secara terpisah. Artinya membuat satu *file* CSS dan *file* CSS tersebut dapat dipanggil berulang-ulang kali untuk dihubungkan dengan *file* HTML (melalui *linked*). Cara ini sangat efektif untuk pembuatan desain *web* karena dapat mempersingkat waktu pembuatan.

##### 2.1.1 Pengenalan Dasar CSS

Menurut Syakirurahman (2013) CSS atau *Cascading Style Sheet* terdiri dari selektor, deklarasi, properti dan nilai. Seperti pada HTML, PHP dan bahasa pemrograman lainnya, CSS juga memiliki aturan tersendiri dalam penulisan kodenya. Adapun *syntax* dari CSS dapat dilihat pada Gambar 2.3.



Gambar 2.3 Struktur CSS

Contoh penulisan dasar kode CSS sebagai berikut.

```
body{  
    background-color : white;  
}
```

“body” merupakan selektor, “{}” merupakan deklarasi, “background-color” adalah properti, dan “white” merupakan nilai. Maksud dari kode tersebut

adalah mengatur warna latar belakang (*background color*) dari *tag body* sebuah halaman *web*.

Penulisan CSS langsung pada *file* HTML merupakan jenis CSS internal. Contoh penulisannya adalah sebagai berikut.

```
<html>
  <head>
    <style type="text/css">
      body {
        background: #ffa500;
      }
      p {
        background: black;
        color: white;
      }
    </style>
  </head>
  <body>
    <p>
      Hallo dunia!
    </p>
  </body>
</html>
```

Contoh penulisan pemanggilan *file* CSS dalam *script* HTML yang merupakan jenis CSS eksternal adalah sebagai berikut.

```
<link rel="stylesheet" src="folder/namafile.css" />
```

“*link*” merupakan *tag html*, “*rel*” (*relationship*) merupakan *attribut*, “*src*” adalah *attribut* yang menandai alamat sumber *file* yang akan dipanggil.

## 2.5 PHP

Menurut Steven Suehring dan Janet Valade (2013) PHP adalah bahasa *scripting* yang dirancang khusus untuk digunakan di *web*. PHP memiliki fitur untuk membantu dalam pembuatan aplikasi *web* dinamis.

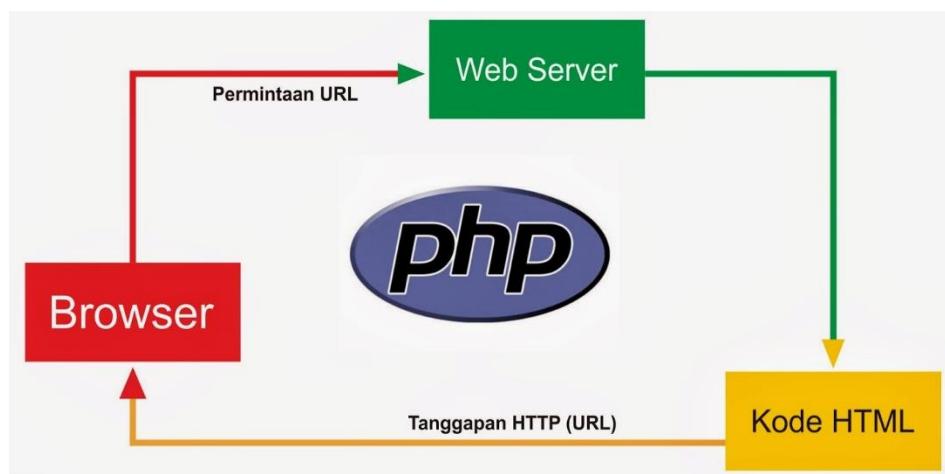
Seperti yang dikutip oleh Ritonga (2015) PHP adalah Bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis. Karena PHP merupakan *server-side scripting* maka sintaks dan perintah-perintah PHP akan dieseksekusi di *server* kemudian hasilnya akan dikirimkan ke *browser* dengan format HTML (Arief, 2011).

Menurut Nugroho (2006) seperti yang dikutip oleh Ritonga (2015) “PHP atau singkatan dari *Personal Home Page* merupakan bahasa skrip yang tertanam dalam HTML untuk dieksekusi bersifat *server side*”.

Karena PHP bersifat *server side* maka *user* tidak akan melihat kode-kode khusus terkait *password* atau akun yang merupakan data-data penting *server* sehingga *server web* lebih terjamin dalam hal keamanan.

PHP bersifat *open source* sehingga dapat dipakai cuma-cuma dan mampu lintas *platform*, yaitu dapat berjalan pada sistem operasi windows maupun linux. PHP dibangun juga sebagai modul pada *web server* apache dan sebagai *binary* yang dapat berjalan sebagai CGI (*Common Gateway Interface*).

Secara prinsip, *server* akan bekerja jika terdapat permintaan dari *client*, yaitu permintaan *client* akan dikirimkan ke *server* dan kemudian *server* akan mengembalikan pada halaman *client* sesuai dengan instruksi yang diminta.



Gambar 2.4 Cara Kerja PHP

PHP hanya mengeksekusi kode yang ditulis dalam pembatas sebagaimana ditentukan oleh dasar sintaks PHP. Adapun di luar pembatas tidak diproses oleh PHP (meskipun teks PHP ini masih mengendalikan struktur yang dijelaskan dalam kode PHP). Pembatas yang paling umum adalah “<?php” untuk membuka dan “?>” untuk menutup kode PHP. Tujuan dari pembatas ini adalah memisahkan kode PHP dari kode di luar PHP, seperti HTML dan javascript.

### 2.5.1 Pengenalan Dasar PHP

Sebuah halaman *web* yang ditulis menggunakan bahasa pemrograman PHP adalah sebagai berikut.

```
<?php  
    echo "Hallo dunia";  
?>
```

Penulisan dan pemanggilan fungsi dalam bahasa pemrograman PHP adalah sebagai berikut.

```
<?php  
function penjumlahan($i1,$i2){  
    return $i1+$i2;  
}  
echo penjumlahan(1,2);  
#output : 3  
?>
```

Di dalam bahasa pemrograman PHP terdapat 3 jenis sintaks sebagai komentar pada kode yaitu tanda blok /\* \*/ komentar 2 baris, // serta tanda pagar # digunakan untuk komentar satu baris. Komentar bertujuan untuk meninggalkan catatan pada kode PHP dan tidak akan diterjemahkan ke program. Contoh penulisan komentar dalam program PHP adalah sebagai berikut.

```
<?php  
/*  
    Di sini dapat ditulis komentar,  
    Dapat digunakan untuk komentar banyak baris  
 */  
//digunakan untuk komentar satu baris  
# digunakan untuk komentar satu baris  
?>
```

## 2.6 MySQL

Menurut Arief (2011) seperti yang dikutip oleh Ritonga (2015) “MySQL adalah salah satu jenis *database server* yang sangat terkenal dan banyak digunakan untuk membangun aplikasi *web* yang menggunakan *database* sebagai sumber dan pengolahan datanya”.

Sedangkan menurut Sulhan (2007) seperti yang dikutip oleh Rani dkk (2014) dalam jurnalnya, “MySQL merupakan perangkat lunak yang digunakan untuk membangun *database* yang sering digunakan di lingkungan linux. MySQL merupakan *software open source* yang berarti *free* untuk digunakan. Selain di lingkungan linux, MySQL juga tersedia di lingkungan windows”.

MySQL dikembangkan oleh perusahaan swedia bernama MySQL AB yang pada saat ini bernama Tcx Data Konsult AB sekitar tahun 1994-1995, namun cikal

bakal kodennya sudah ada sejak tahun 1979. Awalnya Tcx merupakan perusahaan pengembang *software* dan konsultan *database*, dan saat ini MySQL sudah diambil alih oleh Oracle Corp.

Kepopuleran MySQL antara lain karena MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya sehingga mudah untuk digunakan, kinerja *query* cepat, dan mencukupi untuk kebutuhan *database* perusahaan-perusahaan yang berskala kecil sampai menengah, MySQL juga bersifat open source (tidak berbayar).

MySQL merupakan *database* yang pertama kali didukung oleh bahasa pemrograman *script* untuk internet (PHP dan Perl). MySQL dan PHP dianggap sebagai pasangan *software* pembangun aplikasi *web* yang ideal. MySQL lebih sering digunakan untuk membangun aplikasi berbasis *web*, umumnya pengembangan aplikasinya menggunakan bahasa pemrograman *script* PHP.

MySQL didistribusikan dengan *licensi open source* GPL (*General Public License*) mulai versi 3.23 pada bulan Juni 2000.

### 2.6.1 Elemen SQL

Menurut Rifa'i (2013) elemen dari SQL yang paling dasar antara lain adalah pernyataan, nama, tipe data, ekspresi, konstanta, dan fungsi bawaan.

#### 1. Pernyataan

Perintah dari SQL yang digunakan untuk meminta sebuah tindakan kepada DBMS. Pernyataan dasar SQL antara lain sebagai berikut.

- a. ALTER : Merubah struktur tabel.
- b. COMMIT : Mengakhiri eksekusi transaksi.
- c. CREATE : Membuat tabel, *index*s.
- d. DELETE : Menghapus baris pada sebuah tabel.
- e. DROP : Menghapus tabel, *index*s.
- f. GRANT : Menugaskan hak terhadap basis data kepada *user*.
- g. INSERT : Menambah baris pada tabel.
- h. REVOKE : Membatalkan hak kepada basis data.
- i. ROLLBACK : Mengembalikan pada keadaan semula apabila transaksi gagal dilaksanakan.
- j. SELECT : Memilih baris dan kolom pada sebuah tabel.

k. UPDATE : Mengubah *value* pada baris sebuah tabel.

## 2. Nama

Nama digunakan sebagai identitas, yaitu identitas untuk objek pada DBMS.

Misal tabel, kolom, dan pengguna.

## 3. Tipe Data

Tipe data yang ada dalam MySQL adalah sebagai berikut.

a. Tipe data numerik antara lain sebagai berikut.

- 1) TINYINT : nilai *integer* yang sangat kecil.
- 2) SMALLINT : nilai *integer* yang kecil.
- 3) MEDIUMINT : nilai *integer* yang sedang.
- 4) INT : nilai *integer* dengan nilai standar.
- 5) BIGINT : nilai *integer* dengan nilai besar.
- 6) FLOAT : bilangan *decimal* dengan *single-precision*.
- 7) DOUBLE : bilangan *decimal* dengan *double-precision*.
- 8) DECIMAL (M, D) : bilangan *float* yang dinyatakan sebagai *string*. M : jumlah digit yang disimpan, D : jumlah angka di belakang koma.

b. Tipe data *string* antara lain sebagai berikut.

- 1) CHAR : karakter yang memiliki panjang tetap yaitu sebanyak n.
- 2) VARCHAR : karakter yang memiliki panjang tidak tetap yaitu maksimum n.
- 3) TINYBLOB : blob dengan ukuran sangat kecil.
- 4) BLOB : blob yang memiliki ukuran kecil.
- 5) MEDIUMBLOB : blob yang memiliki ukuran sedang.
- 6) LONGBLOB : blob yang memiliki ukuran besar.
- 7) TINYTEXT : teks dengan ukuran sangat kecil.
- 8) TEXT : teks yang memiliki ukuran sedang.
- 9) MEDIUMTEXT : teks yang memiliki ukuran sedang.
- 10) LONGTEXT : teks yang memiliki ukuran besar.
- 11) ENUM : kolom diisi dengan satu *member enumerasi*.

12) SET : kolom dapat diisi dengan beberapa nilai anggota himpunan.

- c. Tipe data tunggal dan jam antara lain sebagai berikut.
  - 1) DATE : *date* memiliki format tahun-bulan-tanggal.
  - 2) TIME : *time* memiliki format jam-menit-detik.
  - 3) DATETIME : gabungan dari format *date* dan *time*.

#### 4. Ekspresi

Ekspresi digunakan untuk menghasilkan/menghitung nilai. Misalnya : jumlah = harga-diskon. Ekspresi aritmatika antara lain sebagai berikut.

- a. + : tambah
- b. - : kurang
- c. / : bagi
- d. \* : kali

#### 5. Konstanta

Nilai yang tetap.

#### 6. Fungsi Bawaan

Fungsi adalah *subprogram* yang dapat menghasilkan suatu nilai apabila fungsi tersebut dipanggil. Fungsi agregat adalah fungsi yang digunakan untuk melakukan *summary*, statistik yang dilakukan pada suatu tabel/*query*.

- a. AVG (ekspresi) : digunakan untuk mencari nilai rata-rata dalam kolom dari tabel.
- b. COUNT (ekspresi) : digunakan untuk menghitung jumlah baris dari sebuah kolom dari tabel.
- c. MAX (ekspresi) : digunakan untuk mencari nilai yang paling besar dari suatu kolom dari tabel.
- d. MIN (ekspresi) : digunakan untuk mencari nilai yang paling kecil dari suatu kolom dari tabel.
- e. SUM (ekspresi) : digunakan untuk menghitung jumlah keseluruhan dari suatu kolom dari tabel.

#### 2.6.2 Kelompok Pernyataan SQL

Menurut Rifa'i (2013) pernyataan SQL dapat dikelompokan menjadi 3 kelompok, yaitu : DDL, DML, dan DCL.

## 1. DDL (*Data Definition Language*)

DDL berfungsi untuk mendefinisikan atribut basis data, tabel, atribut (kolom), batasan-batasan terhadap suatu atribut, serta hubungan antar tabel. Perintah yang digunakan biasanya : CREATE, ALTER, dan DROP.

- a. Membuat *Database*

```
CREATE DATABASE namadatabase;
```

- b. Menghapus *Database*

```
DROP DATABASE namadatabase;
```

- c. Membuat Tabel

```
CREATE TABLE namatabel;
```

- d. Menghapus Tabel

```
DROP TABLE namatabel;
```

- e. Mendefinisikan Null/Not null

```
CREATE TABLE namatabel ( Field1 TipeData1 NOT NULL, Field2 TipeData2 );
```

- f. Mendefinisikan Nilai *Default*

```
CREATE TABLE namatabel ( Field1 TipeData1, Field2 TipeData2 DEFAULT nilai );
```

- g. Mendefinisikan *Primary Key* pada Tabel

```
CREATE TABLE namatabel ( Field1 TipeData1 NOT NULL PRIMARY KEY, Field2 TipeData2 );
```

- h. Menghapus *Primary Key* pada Tabel

```
ALTER TABLE NAMATABEL DROP PRIMARY KEY;
```

- i. Menambah Kolom Baru pada Tabel

```
ALTER TABLE NAMATABEL ADD newfield tipe;
```

- j. Mengubah Tipe Data atau Lebar Kolom pada Tabel

```
ALTER TABLE namatabel MODIFY COLUMN field tipe;
```

k. Mengubah Nama Kolom

```
ALTER TABLE namatabel CHANGE COLUMN namakolomlama  
namakolombaru tipe;
```

l. Menghapus Kolom pada Tabel

```
ALTER TABLE namatabel DROP COLUMN namakolom;
```

m. Mendefinikan Foreign Key pada Tabel

```
CREATE TABLE namatabel ( Field1 TipeData1, Field2  
TipeData2, FOREIGN KEY (Field2) REFERENCES  
namatabelinduk (namakolominduk) ON UPDATE CASCADE  
ON DELETE NO ACTION)
```

n. Menghapus *Foreign Key*

```
ALTER TABLE namatabel DROP FOREIGN KEY  
namaconstraint;
```

## 2. DML (*Data Manipulation Language*)

DML berfungsi untuk memanipulasi data yang ada di dalam basis data, contohnya untuk pengambilan data, penyisipan data, pengubahan data dan penghapusan data.

a. INSERT, merupakan perintah untuk menambah baris pada tabel.

```
INSERT INTO namatabel VALUES (nilai1, nilai2,  
nilai-n);
```

b. DELETE, merupakan perintah untuk menghapus baris pada tabel.

```
DELETE FROM namatabel [where kondisi];
```

c. UPDATE, merupakan perintah untuk mengubah isi beberapa kolom pada tabel.

```
UPDATE namatabel SET kolom1=nilai1, kolom2=nilai2  
[where kondisi];
```

d. SELECT, merupakan perintah untuk menampilkan isi dari suatu tabel yang bisa dihubungkan dengan tabel yang lainnya.

```
SELECT * FROM namatabel
```

### **3. DCL (*Data Control Language*)**

*Data Control Language* (DCL) merupakan *sub* bahasa SQL yang digunakan untuk melakukan pengontrolan data dan *server database*-nya. Perintah DCL diantaranya sebagai berikut.

- a. GRANT, perintah ini digunakan untuk memberikan hak/ijin akses oleh administrator (pemilik utama) *server* kepada *user* (pengguna biasa). Hak akses tersebut berupa hak membuat (CREATE), mengambil (SELECT), menghapus (DELETE), mengubah (UPDATE), dan hak khusus berkenaan dengan sistem *database*-nya.
- b. REVOKE, Perintah ini memiliki kegunaan terbalik dengan GRAND, yaitu untuk menghilangkan atau mencabut hak akses yang telah diberikan kepada *user* oleh administrator.

## **2.7 Java Script**

Dikutip dari Piyaneo (2014) dalam artikelnya mengatakan Java Script merupakan bahasa *scripting* yang pertama kali dikembangkan oleh Netscape pada tahun 1955. Penulisan Java Script berada di dalam dokumen HTML dan pemanggilan program tersebut tergantung pada *browser (navigator)* yang digunakan dalam memanggil halaman yang terdapat pada *script* tersebut. Java Script juga tidak memerlukan kompilator atau penerjemah khusus untuk menjalankannya (Utomo, 2007).

Sedangkan menurut Bride (2007) seperti yang dikutip oleh Piyaneo (2014) Java Script adalah bahasa pemrograman berbasis *browser*. Kode-kodenya ditulis langsung ke dalam HTML dari halaman-halaman *web* dan diterjemahkan serta dieksekusi sebagai respon terhadap aktivitas-aktivitas pada halaman *web*.

### **2.7.1 Karakteristik Java Script**

Menurut Sidik (2007) serperti yang dikutip oleh Piyaneo (2014), terdapat beberapa hal khusus yang terdapat dalam Java Script, hal-hal khusus tersebut adalah sebagai berikut.

1. Menggunakan blok awal “{“ dan blok akhir “}”.
2. *Automatic conversion* dalam pengoperasian tipe data yang berbeda.

3. *Case sensitive*, sehingga *programmer* harus berhati-hati dalam menggunakan variabel, fungsi, dan lain-lain.
4. *File extension* yang umumnya digunakan adalah “\*.js”.
5. Setiap *statement* dapat diakhiri dengan “;” tetapi dapat juga tidak.
6. Jika tidak didukung oleh *browser* tipe lama, *script*-nya dapat disembunyikan di antara *tag* “<!–“ dan “–>”.
7. Jika program dalam satu baris terlalu panjang, dapat disambung ke baris berikutnya dengan karakter “\” di awal baris tersebut.

### 2.1.2 Pengenalan Dasar Java Script

Untuk membuat *file javascript* dengan penulisan *syntax* yang benar terdapat pengetahuan dasar yang harus diketahui yaitu sebagai berikut.

1. *File* disimpan dengan ekstensi “.js”.
2. *Syntax javascript* yang disisipkan pada halaman HTML ditulis di dalam *tag* <script>.
3. *Syntax javascript* bisa diletakan di bagian *element tag* <head> atau *tag* <body>.
4. Javascript bersifat *case sensitive*.
5. Di setiap akhir baris harus ditutup dengan titik koma/semicolon (;).

Penulisan javascript dalam dilakukan dengan dua cara yaitu secara internal atau secara eksternal.

1. **Internal**, merupakan cara penulisan javascript dengan cara menyisipkan langsung *syntax javascript* di dalam HTML. Contoh penulisan javascript secara internal adalah sebagai berikut.

```
<!DOCTYPE html>
<html>
<head>
    <title>Belajar Javascript</title>

</head>
<body>
    <h1>Belajar Javascript </h1>
    <h2>Tugas Akhir</h2>
    <div id="nama"></div>

    <script>
        document.getElementById("nama").innerHTML =
        "Nama Saya Ahmad Rifai";
    </script>
```

```
</body>
</html>
```

2. **Eksternal**, merupakan cara penulisan javascript dilakukan secara terpisah dengan *file* HTML dan untuk mengeksekusi *file* javascript harus dilakukan pemanggilan *file* javascript dalam *file* HTML. Contoh penulisan javascript secara eksternal adalah sebagai berikut.

- a. *File* HTML

```
<!DOCTYPE html>
<html>
<head>
    <title>Belajar Javascript </title>

</head>
<body>
    <h1>Belajar</h1>
    <h2>TA</h2>

    <div id="nama"></div>
</body>
<script type="text/javascript"
src="belajar.js"></script>
</html>
```

- b. *File* Javascript (belajar.js)

```
document.getElementById("nama").innerHTML = "Nama
Saya Malas Ngoding";
```

## 2.8 JQuery

Menurut Aloysius Sigit W. (2011) seperti yang dikutip oleh Eywa (2015) jQuery adalah librari atau kumpulan kode JavaScript siap pakai. Keunggulan menggunakan jQuery dibandingkan dengan JavaScript standar, yaitu menyederhanakan kode JavaScript dengan cara memanggil fungsi-fungsi yang disediakan oleh jQuery. JavaScript sendiri merupakan bahasa *Scripting* yang bekerja disisi *Client/Browser* sehingga *website* bisa lebih interaktif.

Sedangkan menurut Alexander F.K Sibero (2011) seperti yang dikutip oleh Eywa (2015) jQuery adalah salah satu javascript *framework* terbaik saat ini. JQuery dikembangkan oleh John Resig pada tahun 2006 di BarCamp NYC. Pada awal perkembangannya, jQuery pertama dibuat untuk meringkas penggunaan CSS

*Selector* dalam suatu pustaka fungsi. jQuery memiliki ciri khas pada penggunaan perintahnya, *prefix* untuk jQuery dengan tanda \$ kemudian dilanjutkan dengan fungsi atau perintah.

### 2.8.1 Fitur JQuery

1. Dalam Pemakaian menggunakan seleksi *element* DOM, sehingga *website* lebih dinamis dan interaktif.
2. JQuery bisa memanipulasi *Class* pada CSS dan *Support CSS 3*
3. *Event*
4. AJAX
5. Efek-efek dan animasi
6. Ekstensi dan *Plug-ins*
7. Kompatibilitas dengan hampir semua *Browser modern*
8. Keperluan lain seperti : *User Agent*, *Feature detection* dan lainnya

## 2.9 Desain Web Responsif

Menurut Jeffrey Zeldman seperti dikutip dari Hidayat dkk. (2016), responsive design adalah sebuah teknik yang digunakan desainer website untuk memberikan pengalaman visual yang elegan tanpa mempedulikan ukuran browser yang digunakan dan batasan apapun tentang cara mengakses perangkat tersebut. Sebuah desain dianggap responsif jika menggunakan tiga poin yaitu grid yang fleksibel, gambar dan media yang fleksibel, dan permintaan media.

Responsive web design digunakan dengan tujuan untuk memastikan informasi website yang akan disampaikan berjalan dengan baik tanpa kehilangan informasi dan terlepas diakses dari perangkat mobile apapun. Menurut Cerejo, kualitas informasi dari sebuah website dapat diamati melalui 3 dari 12 aspek pengalaman pengguna mobile, yaitu: fungsionalitas, arsitektur informasi, dan konten (Hidayat dkk, 2016).

### 2.9.1 Ciri Web Desain Responsif

Beberapa ciri web dengan desain web responsif

1. *User interface*-nya beradaptasi di berbagai perangkat berbeda.
2. Ukuran huruf, tata letak dan gambarnya beradaptasi di berbagai ukuran layar yang berbeda.

3. Umumnya menggunakan *scrolling* dan *swipe* untuk bernaligasi di perangkat *mobile*.
4. Lintas *platform*.

## 2.10 Bootstrap

Menurut Widyantoro dalam buku panduan bootstrap, “Bootstrap adalah sebuah *framework* yang dapat menyelesaikan permasalahan dalam mendesain *web*. Slogan dari *framework* ini adalah “*Sleek, intuitive, and powerful front-end framework for faster and easier web development*”, yang berarti kita dapat mendesain sebuah *website* dengan lebih rapi, cepat dan mudah. Selain itu Bootstrap juga *responsive* terhadap banyak platform, artinya tampilan halaman *website* yang menggunakan Bootstrap ini akan tampak tetap rapi, baik versi *mobile* maupun *desktop*”.

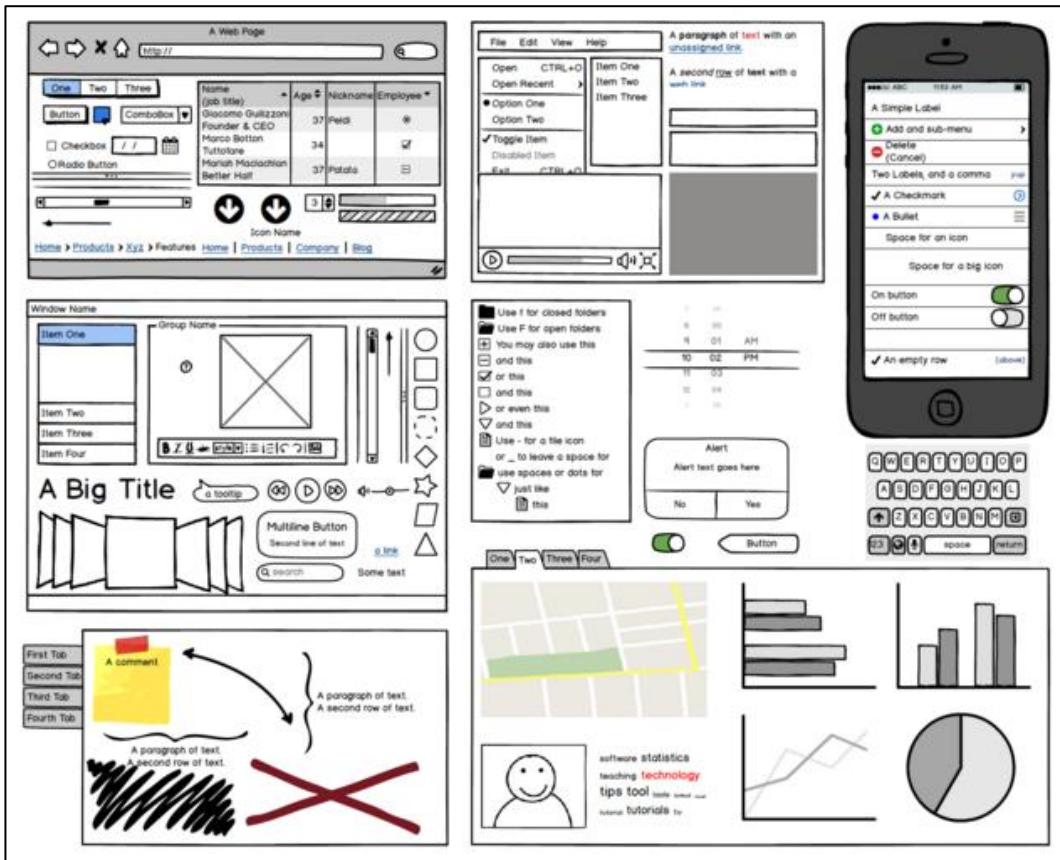
Bootstrap adalah *front-end framework* atau *framework* css yang sengaja dibangun untuk memudahkan pendesain web dalam membuat *website* atau membuat *template website*. *Framework* yang dulu pernah dikenal dengan twitter bootstrap ini dilengkapi dengan komponen-komponen *website* yang sering digunakan seperti *breadcrumb*, *pagination*, *modal dialog*, *navigation bar* dan lain-lain yang sangat memudahkan pendesain web saat membuat *website*, bahkan bootstrap ini selalu diunggul-unggulkan dengan fitur *website responsive*-nya yang memungkinkan kita cukup membuat satu *template website* saja dan kita bisa memperoleh template yang bisa dibuka atau kompatibel diberbagai *device* (laptop, handphone, atau tablet) (Onphpid, 2015).

## 2.11 Balsamiq Mockups

Menurut Dwi (2016), “Balsamiq Mockups adalah program aplikasi yang digunakan dalam pembuatan tampilan *user interface* sebuah aplikasi. *Software* ini sudah menyediakan *tools* yang dapat memudahkan dalam membuat desain *prototyping* aplikasi yang akan kita buat. *Software* ini berfokus pada konten yang ingin digambar dan fungsionalitas yang dibutuhkan oleh pengguna”.

Menggambar sketsa (*wireframe*) atau *prototype* rancangan desain *website* di atas kertas Balsamiq Mockups membantu untuk memudahkan dalam melakukan desainer *web*. Tujuannya agar dapat menyesuaikan dengan kebutuhan *customer*

(pelanggan). Dengan alat pembuat *mockup* maka akan dapat menganalisa tata letak, desain, dan fungsi. Berikut komponen *user interface* pada Balsamiq Mockups.

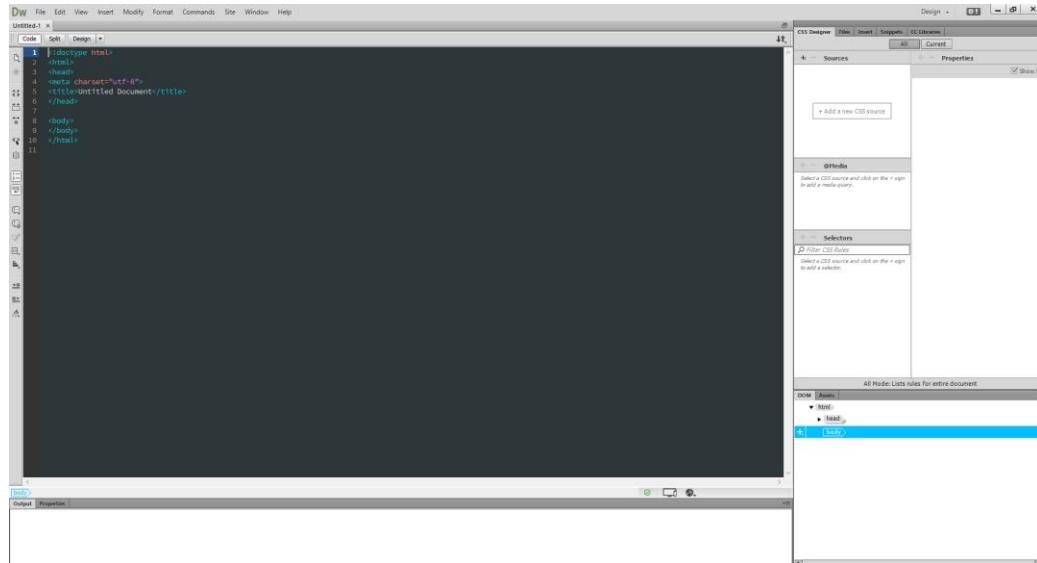


Gambar 2.5 Komponen *User Interface* Balsamiq Mockups

## 2.12 Dreamweaver

Menurut Madcoms (2011) seperti yang dikutip oleh Nanda (2015) memberikan batasan bahwa “Dreamweaver adalah sebuah HTML editor profesional untuk mendesain *web* secara visual dan mengelola situs atau halaman *web*”. Saat ini terdapat *software* dari kelompok adobe yang belakangan banyak digunakan untuk mendesain suatu situs *web*. Versi terbaru dari Adobe Dreamweaver saat ini adalah Dreamweaver CC, terdapat beberapa kemampuan bukan hanya sebagai *software* untuk desain *web* saja tetapi juga untuk menyunting kode serta pembuatan aplikasi *web* dengan menggunakan berbagai bahasa pemograman *web*, antara lain: JPS, PHP, ASP dan ColdFusion.

## 2.12.1 Pengenalan Ruang Kerja Dreamweaver



## Gambar 2.6 Ruang Kerja Dreamweaver

## 1. Aplikasi *Bar*

Berada di bagian paling atas jendela aplikasi Dreamweaver. Baris ini berisi tombol *workspace* (*workspace switcher*), menu, dan lain sebagainya.

## 2. *Toolbar Dokumen*

Berisi tombol-tombol yang digunakan untuk mengubah tampilan jendela dokumen, sebagai contoh tampilan *design* atau tampilan kode. Juga dapat digunakan untuk operasi-operasi umum, misalnya untuk melihat hasil sementara halaman *web* pada jendela *browser*.

### **3. Jendela Dokumen**

Lembar kerja tempat anda membuat dan mengedit desain halaman *web*.

## 4. *Workspace Switcher*

Digunakan untuk mengubah tampilan ruang kerja (*workspace*) Dreamweaver CS5 *up to abover*.

## **5. Panel Groups**

Adalah kumpulan panel yang saling berkaitan, *panel-panel* ini dikelompokkan pada judul-judul tertentu berdasarkan fungsinya. Panel ini digunakan untuk memonitor dan memodifikasi pekerjaan.

## 6. Tag Selector

Diletakkan di bagian bawah jendela dokumen, satu baris dengan status bar.  
Bagian ini menampilkan hirarki pekerjaan yang sedang terpilih pada jendela

dokumen, dapat juga digunakan untuk memilih objek pada jendela *design* berdasarkan jenis atau kategori objek tersebut.

#### **7. *Panel Property***

Digunakan untuk melihat dan mengubah berbagai properti objek atau teks pada jendela *design*.

#### **8. *Toolbar Coding***

Berisi tombol-tombol yang digunakan untuk melakukan operasi kode-kode standar

#### **9. *Panel Insert***

Berisi tombol-tombol untuk menyisipkan berbagai jenis objek, seperti *image*, tabel, atau objek media ke dalam jendela dokumen.

#### **10. *Panel Groups***

Adalah kumpulan panel yang saling berkaitan satu sama lainnya yang dikelompokkan di bawah satu judul, seperti: *Insert*, *Files*, *CSS Style*, dan sebagainya.

#### **11. *Panel Files***

Digunakan untuk mengatur *file-file* dan folder-folder yang membentuk situs *web*.

### **2.13 Microsoft Visio**

Menurut Nur Desiana (2013), Microsoft Visio adalah aplikasi yang digunakan untuk merancang suatu model perencanaan, model ini dimanfaatkan untuk kebutuhan *developer* maupun *engineering* yang didesain untuk berbagai macam kebutuhan.

Microsoft Visio adalah aplikasi untuk menggambarkan diagram yang dapat membantu pekerja IT dan staf profesional untuk memvisualisasikan informasi yang kompleks. Berawal dari teks yang rumit dan tabel yang kompleks yang sulit dipahami, Microsoft Visio berusaha untuk membantu kondisi tersebut (Willyanto Santoso, 2013).

Microsoft Visio adalah *tools* untuk membuat berbagai jenis diagram, mulai dari diagram jaringan sampai ke kelender dan mulai dari tampilan teks biasa, sampai ke bentuk flowchart. Visio membuat dokument visual secara profesional

untuk membantu analisa dan komunikasi informasi, sistem dan proses yang kompleks (Willyanto Santoso, 2013).

## 2.14 Corel Draw

Menurut Pranowo (2010) CorelDraw adalah sebuah software desain grafis yang sangat terkenal dan termasuk sebagai salah satu aplikasi pengolah gambar berbasis vector. Berbagai fasilitas untuk keperluan desain tersedia dalam program ini, sehingga memudahkan para pengguna untuk memanfaatkannya. Sebagai pengolah *vector*, CorelDraw sering digunakan untuk membuat gambar kartun, logo, ilustrasi, dan sebagainya. Selain itu CorelDraw juga dapat digunakan untuk mengatur tata letak secara sederhana, walaupun mungkin tidak terlalu lengkap seperti aplikasi yang memang diperuntukkan untuk mengatur tata letak. Contoh penggunanya antar lain untuk membuat brosur, pamphlet, surat undangan, *cover* buku, dan sebagainya.

Sedangkan menurut Awwali (2014), Corel Draw merupakan sebuah aplikasi grafis berbasis *vector*. Corel Draw berfungsi untuk membuat desain logo dan desain tampilan *interface* halaman *web* dalam tugas akhir ini.

## 2.15 Filezilla

Filezilla adalah FTP *client* dan *server* yang cepat dan reliabel dengan Pada percobaan banyak fitur yang berguna serta *interface* yang intuitif. Dengan Filezilla sebagai jembatan untuk dapat men-*download* dari server maupun meng-*upload* dari *client* ke *server* (Pratiwi Masero dkk, 2013).

### 2.15.1 Fitur Filezilla

1. **Site Manager** (Manajer situs) Mengizinkan pengguna untuk membuat daftar situs FTP beserta data konesinya, seperti nomor *port* yang akan digunakan, protokol yang digunakan, dan apakah akan menggunakan log anonim atau normal. Untuk log normal, nama pengguna dan kata sandinya akan disimpan. Penyimpanan kata sandi adalah opsional.
2. **Message Log** (Log pesan) Ditampilkan di bagian atas jendela. Fitur ini menampilkan *output* berjenis konsol (*console-type*) yang menunjukkan perintah yang dikirim oleh FileZilla dan respon yang diterima dari *server*.

3. ***File and Folder View*** Ditampilkan di bawah pesan *log* (*Message log*), menyediakan sebuah tampilan grafis antarmuka untuk FTP. Pengguna dapat menavigasi folder serta melihat dan mengubah isinya pada komputer lokal dan *server* dengan menggunakan tampilan antarmuka gaya Explorer. Pengguna dapat men-drag dan drop *file* antara komputer lokal dan *server*.
4. ***Transfer Queue* (*Transfer* antrian)** Ditampilkan di sepanjang bagian bawah jendela, menunjukkan status *real-time* setiap antrian atau *transfer file* yang aktif.

## 2.16 Regexp (*Regular Expression*)

Menurut Fitzgerald (2012), *Regular Expression* secara khusus pengkodean teks sebagai pola pencocokan teks. *Regular expression* mulai muncul pada tahun 1940-an sebagai cara untuk menggambarkan bahasa reguler. Ekspresi reguler adalah pola yang menentukan satu *set string* karakter untuk dicocokan dengan string tertentu.

Dikutip dari Muliantara (2014) dalam jurnalnya mengatakan Regular expression atau yang sering disebut sebagai Regex adalah sebuah formula untuk pencarian pola suatu kalimat/string (Regular,2009).

### 2.16.1 Pengenalan Dasar Regexp

Regex memiliki sintak yang terdiri dari *delimiter*, *pattern*, dan *modifier*. *Delimiter* digunakan untuk menentukan dimana *pattern* berawal dan berakhir. Pada javascript hanya bisa menggunakan garis miring sebagai *delimiter*. *Pattern* adalah pola string yang ingin ditemukan atau dicocokkan. *Modifier* adalah mode regex meng-handle hasil pencarian *string* atau teks (Morphine, 2012).

#### 1. *Modifier*

Modifier berfungsi untuk mengatur hasil pencarian regex. Terdapat 3 *modifier* umum yang di-support berbagai bahasa pemrograman, yaitu *Global*, *Case-insensitive*, dan *Multiline*.

#### 2. *Metacharacter*

*Metacharacter* adalah karakter-karakter yang memiliki arti spesial dalam *pattern* regex tergantung dimana karakter tersebut digunakan. *Metacharacter* dalam regex adalah (. \ + \* ? [ ^ ] \$ ( ) { } = ! < > | : -).

### 3. Escaping metacharacter

*Escaping metacharacter* adalah *backslash* (\). *Escaping* maksudnya mengabaikan arti spesial dari karakter sehingga bersifat *literal*. *Escaping* hanya berlaku pada *metacharacter* dan *delimiter*, sedangkan penggunaan *backslash* pada karakter lain dapat memiliki arti berbeda. *Backslash* digunakan tepat sebelum karakter yang akan di-*escape*. Kasus *escaping* misalnya pada *pattern* yang memiliki karakter garis miring yang merupakan *delimiter*. *Pattern* “http://” dalam regex tidak bisa dituliskan seperti /http://i, kode tersebut akan menganggap *pattern* adalah “http:” karena *string* tersebut berada di antara *delimiter* (/http://i) dan garis miring setelahnya (/http://i) akan dianggap sebagai *modifier* yang tidak diketahui sehingga bersifat *error*. Jadi, untuk mengabaikan arti karakter garis miring sebagai *delimiter* perlu meng-*escape*-nya seperti contoh berikut ini.

```
/http:\//.test("http://www.TA.com");  
/http:\//.test("ftp://www.TA.com");
```

### 4. Backslash dan Special Character

*Backslash* dan *Special Character* digunakan sebagai pembentuk *special character*, contohnya *control character* dan ASCII karakter yang tidak ada pada keyboard standard. Penggunaan karakter *backslash* (“\”) sebelum karakter “t” akan memiliki arti bahwa kedua karakter tersebut adalah satu karakter, yaitu *horizontal tab (tab)*. Jadi regex \t\t akan mencocokan dua karakter *tab*, bukan *string* dengan karakter berurutan “*backslash, t, backslash, t*”. *Control characters* yang dapat digunakan pada regex adalah sebagai berikut.

- a. \a *alert (bell)*
- b. \b *backspace*
- c. \e *ASCII escape character*
- d. \f *form feed*
- e. \n *newline*
- f. \r *carriage return*
- g. \t *horizontal tab*
- h. \v *vertical tab*

Selain karakter-karakter tersebut dapat juga mengakses karakter dari *code hexadecimal* atau *unicode* dengan format /x<code><code> atau \u<code><code><code><code>. Misalnya karakter copyright (©) dapat ditulis dengan \xA9 atau \u00A9, format tersebut berlaku dalam javascript, beberapa bahasa pemrograman memiliki formatnya sendiri, seperti PHP menggunakan \x dibandingkan \u.

## 5. Wildcard

Karakter *wildcard* dalam regex adalah titik (.). *Wildcard* dapat dicocokan dengan hampir seluruh karakter, bisa huruf, angka atau tanda baca termasuk titik. Karakter yang tidak bisa dicocokan dengan *wildcard* adalah baris baru (“\n”). Satu *wildcard* hanya cocok dengan satu karakter. Contoh kode regex *wildcard* seperti berikut.

```
/w.b/g.test("web wwb w3b w@b w.b w?b w b w\nb w\tb w\xA5b woob");
```

## 6. Alternation

*Alternation* digunakan untuk mencari beberapa *substring* seperti “jpg”, “png”, dan “gif” dalam suatu *string*, maka *pattern* masing-masing *substring* tersebut adalah /jpg/, /gif/, dan /png/. Hal tersebut akan lebih efektif jika menggunakan 3 *pattern* untuk satu *input string*. Tiga *pattern* tersebut dapat digabung dan memiliki makna “alternatif”. Contoh *pattern*-nya adalah jpg|png|gif. Dapat dilihat dari contoh tersebut bahwa karakter *pipe* () adalah *metacharacter* untuk *alternation*. Regex / jpg|png|gif / memiliki arti cari *substring* “jpg” atau “png” atau “gif”. contoh kodennya adalah sebagai berikut.

```
/jpg|gif|png/g.test("avatar.gif");
/jpg|gif|png/g.test("avatar.png");
/jpg|gif|png/g.test("banner.png");
/jpg|gif|png/g.test("banner.jpg");
/jpg|gif|png/g.test("pngjpggif");
```

## 7. Grouping

*Grouping* adalah pengelompokan *subpattern* (ekspresi) tertentu sehingga dapat dibedakan dengan ekspresi lainnya. *Grouping* menggunakan buka kurung (“(” dan tutup kurung “)”) dan ekspresi diletakan di antara buka kurung dan

tutup kurung. Kasus penggunaan *grouping* misalnya mencocokan *file* dengan nama “banner” yang memiliki ekstensi “jpg”, “png”, atau “gif”. *Pattern* yang dapat digunakan adalah /banner\.(jpg|banner\.(png|banner\.(gif)/. dari *pattern* tersebut bisa dilihat bahwa hanya terdapat perbedaan pada ekstensi jpg, png, dan gif. ekstensi tersebut dapat digabung dalam satu *alternation* menjadi jpg|png|gif, dan untuk membedakan bahwa jpg|png|gif adalah kelompok tersendiri., maka dilakukan *grouping*, sehingga regex akhirnya adalah /banner\.(jpg|png|gif)/. Regex /banner\.(jpg|banner\.(png|banner\.(gif/ memiliki arti cari *substring* “banner.jpg” atau “banner.png” atau “banner.gif”, sedangkan regex /gambar\.(jpg|png|gif) memiliki arti cari *substring* “banner.” Diikuti dengan substring “jpg” atau “png” atau “gif”. Contoh kodennya adalah sebagai berikut.

```
/banner\.(jpg|banner\.(gif|banner\.(png/.test("path/avatar.gif");
/banner\.(jpg|banner\.(gif|banner\.(png/.test("path/banner.gif\n");
/banner\.(jpg|gif|png)/.test("path/avatar.gif");
/banner\.(jpg|gif|png)/.test("path/banner.gif\n");
/banner\.(jpg|gif|png)/.test("path/banner.png");
/banner\.(jpg|gif|png)/.test("path/banner.jpg");
```

## 8. Nesting Grouping

*Grouping* juga dapat dilakukan dalam *group* lain, yang disebut dengan *nesting grouping* atau pengelompokan bersarang. Contoh terdapat folder dengan nama “gambar” atau “image” dan di dalam folder terdapat file dengan nama “banner” atau “top” dengan ekstensi “jpg” atau “png” atau “gif”. maka format regex untuk kasus tersebut adalah sebagai berikut.

```
/((gambar|image)\/( (banner|top)\.(jpg|gif|png))) /
```

## 9. Backreferences Index Group

Urutan pengelompokan pada penjelasan di atas disebut *index group*. *Index group* ini dapat dipakai lagi dalam *pattern* regex dengan menggunakan *backreferences index* atau *indexing*. *Indexing* dilakukan dengan menggunakan

*backslash ()* diikuti nomor *index*, contohnya \1, \2, \3 dan seterusnya. Misalnya ingin membuat pola *attribut* HTML yang memiliki tanda petik (*quote*) yang harus sama antara *open* dan *close*-nya, contohnya adalah src="banner.png" atau src='banner' tetapi akan cocok juga untuk src='banner.png' atau src="banner.png" (perhatikan kedua *quote*-nya) dimana *string* tersebut seharusnya tidak valid. *Indexing* berfungsi untuk menangani persoalan tersebut. Contoh kode regex-nya adalah sebagai berikut.

```
/src=(" | ')banner.png\1/.test("src=\"banner.png\"");  
/src=(" | ')banner.png\1/.test("src='\''banner.png'\''");  
/src=(" | ')banner.png\1/.test("src=\"banner.png'\\"");  
/src=(" | ')banner.png\1/.test("src='banner.png\"");
```

## 10. Skip Indexing

*Grouping* regex secara otomatis akan meng-*index* hasil *grouping* sehingga dapat digunakan untuk *backreferences*. *Indexing* dapat di-skip dalam *grouping* dengan menggunakan 2 *metacharacter* yaitu tanda tanya dan titik dua (?), yang diletakkan setelah kurung buka ("("). Contoh regex-nya adalah sebagai berikut.

```
/(?:(?:gambar|image)\|(?:banner|top)\|.jpg|gif|png))/
```

## 2.17 Normalisasi Database

Menurut Abdul kadir (2009) seperti yang dikutip oleh Saputra (2013) normalisasi adalah suatu proses yang digunakan untuk menentukan kelompok atribut-atribut dalam sebuah relasi sehingga di peroleh relasi yang berstruktur baik. Dalam hal ini yang dimaksud dengan relasi yang berstruktur dengan baik adalah relasi yang memenuhi 2 kondisi yaitu mengandung redundansi sesedikit mungkin dan memungkinkan barisbaris dalam relasi disisipkan, dimodifikasi, dan dihapus tanpa menimbulkan kesalahan dan ketidak konsistennan.

Tujuan normalisasi adalah mengorganisasikan data ke dalam tabel-tabel untuk memenuhi kebutuhan pemakai, menghilangkan kerangkapan data, mengurangi kompleksitas, mempermudah modifikasi data (Latief, 2012).

Menurut Latief (2012) proses normalisasi dan tahapan normalisasi adalah sebagai berikut.

### **2.17.1 Proses Normalisasi**

1. Data diuraikan dalam bentuk tabel, selanjutnya dianalisis berdasarkan persyaratan tertentu ke beberapa tingkat
2. Apabila tabel yang diuji belum memenuhi persyaratan tertentu maka tabel tersebut perlu dipecah menjadi beberapa tabel yang lebih sederhana sampai memenuhi bentuk yang optimal

### **2.17.2 Tahapan Normalisasi**

1. Bentuk tidak normal adalah menghilangkan perulangan *group*.

Tabel 2.1 Contoh Bentuk Data Tidak Normal (*Unnormal*)

No-mhs	Nama Mhs	Jurusan	Kode-MK	Nama-MK	Kode Dosen	Nama Dosen	Nilai
2663	Jono	MI	M1350	Manajemen DB	B104	Ati	A
			M1465	Analisis Perc. Sistem	B317	Dita	B
5432	Bakti	Ak.	M1350	Manajemen DB	B104	Ati	C
			MKT300	Dasar Pemrograman	B212	Lola	A
			AKN201	Akutansi Keuangan	D310	Lia	B

2. Bentuk normal pertama (1NF) adalah menghilangkan ketergantungan sebagian. Yaitu suatu relasi dikatakan sudah memenuhi bentuk normal kesatu bila setiap data bersifat atomik yaitu setiap irisan dan kolom hanya mempunyai satu nilai data.

Tabel 2.2 Contoh Bentuk Normal Pertama (1NF)

No-mhs	Nama Mhs	Jurusan	Kode-MK	Nama-MK	Kode Dosen	Nama Dosen	Nilai
2663	Jono	MI	M1350	Manajemen DB	B104	Ati	A
2663	Jono	MI	M1465	Analisis Perc. Sistem	B317	Dita	B
5432	Bakti	Ak.	M1350	Manajemen DB	B104	Ati	C
5432	Bakti	Ak.	MKT300	Dasar Pemrograman	B212	Lola	A
5432	Bakti	Ak.	AKN201	Akutansi Keuangan	D310	Lia	B

3. Bentuk normal kedua (2NF) adalah menghilangkan ketergantungan transatif. Yaitu suatu relasi dikatakan sudah memenuhi bentuk normal kedua bila relasi tersebut sudah memenuhi bentuk normal kesatu dan atribut yang bukan *key* sudah tergantung penuh terhadap *key*-nya.

Tabel 2.3 Contoh Bentuk Normal Kedua (2NF) Tabel Mahasiswa

No-mhs	Nama Mhs	Jurusan
2663	Jono	MI
5432	Bakti	Ak.

Tabel 2.4 Contoh Bentuk Normal Kedua (2NF) Tabel Mata Kuliah

Kode-MK	Nama-MK	Kode Dosen	Nama Dosen
M1350	Manajemen DB	B104	Ati
M1465	Analisis Perc. Sistem	B317	Dita
MKT300	Dasar Pemrograman	B212	Lola
AKN201	Akutansi Keuangan	D310	Lia

Tabel 2.5 Contoh Bentuk Normal Kedua (2NF) Tabel Nilai

Kode Dosen	No-mhs	Nilai
B104	2663	A
B317	2663	B
B104	5432	C
B212	5432	A
D310	5432	B

4. Bentuk normal ketiga (3NF) adalah menghilangkan anomali-anomali hasil dari ketergantungan fungsional. Yaitu suatu relasi dikatakan sudah memenuhi bentuk normal ketiga bila relasi tersebut sudah memenuhi bentuk normal kedua dan atribut yang bukan *key* tidak tergantung transitif terhadap *key*-nya.

Tabel 2.6 Contoh Bentuk Normal Ketiga (3NF) Tabel Dosen

Kode Dosen	Nama Dosen
B104	Ati
B317	Dita
B212	Lola
D310	Lia

Tabel 2.7 Contoh Bentuk Normal Ketiga (3NF) Tabel Mata Kuliah

Kode-MK	Nama-MK	Kode Dosen
M1350	Manajemen DB	B104
M1465	Analisis Perc. Sistem	B317
MKT300	Dasar Pemrograman	B212
AKN201	Akutansi Keuangan	D310

5. Bentuk normal Boyce-Codd (BCNF) adalah menghilangkan ketergantungan *mutivalue*.
6. Bentuk normal keempat (4NF) adalah menghilangkan anomali-anomali yang tersisa.
7. Bentuk Normal kelima (5NF).

## 2.18 UML (*Unified Modeling Language*)

Menurut Muslihudin (2016) UML singkatan dari *Unified Modeling Language* yang berarti bahasa pemodelan standar. Chonoles (2013) mengatakan sebagai bahasa, berarti UML memiliki sintaks dan semantik.

Dalam pembuatan UML harus mengikuti aturan-aturan yang ada karena terdapat hubungan antara satu elemen dengan elemen lainnya pada suatu model.

UML bukan hanya sekadar diagram, tetapi juga menceritakan konteksnya. Ketika pelanggan memesan sesuatu dari sistem, bagaimana transaksinya ? bagaimana sistem mengatasi *error* yang terjadi ? bagaimana keamanan terhadap sistem yang kita buat ? dan sebagainya dapat dijawab dengan UML.

### 2.18.1 Beberapa Diagram Dalam UML

1. Diagram Kelas (*Class Diagram*), bersifat statis. Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka - antarmuka, kolaborasi-kolaborasi, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula diagram kelas memuat kelas-kelas aktif.
2. Diagram Paket (*Package Diagram*), bersifat statis. Diagram ini memperlihatkan kumpulan kelas-kelas, merupakan bagian dari diagram komponen.
3. Diagram *Use Case*, bersifat statis. Diagram ini memperlihatkan himpunan *use case* dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama

sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna.

4. Diagram Interaksi dan *Squence* (Urutan), bersifat dinamis. Dinamis urutan adalah diagram interaksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu.
5. Diagram Komunikasi (*Communication Diagram*), bersifat dinamis. Diagram sebagai pengganti diagram kolaborasi UML 1.4 yang menekankan organisasi struktural dari objek-objek yang menerima serta mengirim pesan.
6. Diagram *Statechart* (*Statechart Diagram*), bersifat dinamis. Diagram status memperlihatkan keadaan-keadaan pada sistem, memuat status (*state*), transisi, kejadian serta aktivitas. Diagram ini terutama penting untuk memperlihatkan sifat dinamis dari antarmuka (*interface*), kelas, kolaborasi, dan terutama pada pemodelan sistem-sistem yang reaktif.
7. Diagram Aktivitas (*Activity Diagram*), bersifat dinamis. Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem. Diagram ini terutama penting dalam suatu sistem serta pemodelan fungsi-fungsi suatu sistem dan memberi tekanan pada aliran kendali antarobjek.
8. Diagram Komponen (*Component Diagram*), bersifat statis. Diagram komponen ini memperlihatkan organisasi serta ketergantungan sistem/perangkat lunak pada komponen-komponen yang telah ada sebelumnya. Diagram ini berhubungan dengan diagram kelas dimana komponen secara tipikal dipetakan ke dalam satu atau lebih kelas-kelas, antarmuka - antarmuka serta kolaborasi-kolaborasi
9. Diagram *Deployment* (*Deployment Diagram*), bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan (*run-time*). Memuat simpul-simpul beserta komponen - komponen yang ada di dalamnya. Diagram *deployment* berhubungan erat dengan diagram komponen dimana diagram ini memuat satu atau lebih komponen-komponen. Diagram ini sangat berguna saat aplikasi kita berlaku sebagai aplikasi yang dijalankan pada banyak mesin (*distributed computing*).

## 2.18.2 Use Case Diagram

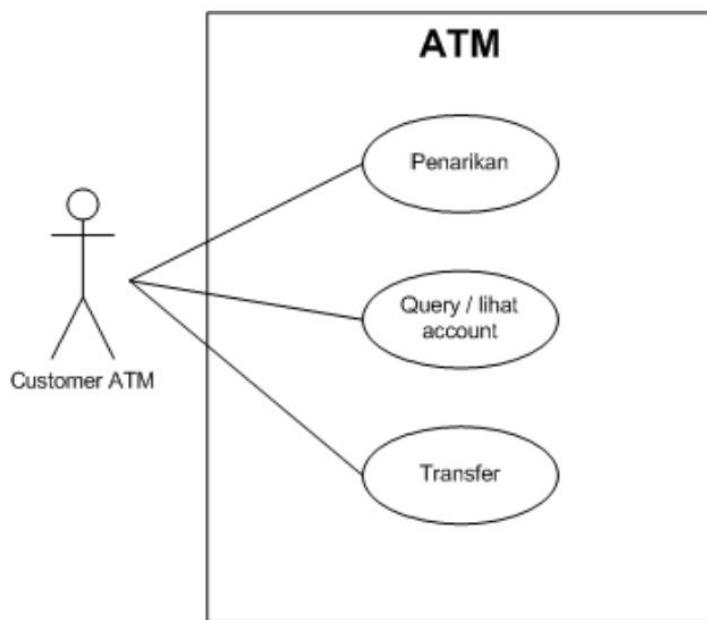
Menurut Mujilan (2013) *Use case diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya *login* ke sistem, *meng-create* sebuah daftar belanja, dan sebagainya. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan - pekerjaan tertentu.

*Use case diagram* dapat sangat membantu bila kita sedang menyusun *requirement* sebuah sistem, mengkomunikasikan rancangan dengan klien, dan merancang *test case* untuk semua *feature* yang ada pada sistem.

Sebuah *use case* dapat meng-*include* fungsionalitas *use case* lain sebagai bagian dari proses dalam dirinya. Secara umum diasumsikan bahwa *use case* yang di-*include* akan dipanggil setiap kali *use case* yang meng-*include* dieksekusi secara normal.

Sebuah *use case* dapat di-*include* oleh lebih dari satu *use case* lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang *common*.

Contoh *use case* sederhana dapat dilihat pada Gambar 2.7.



Gambar 2.7 *Use Case* Sederhana

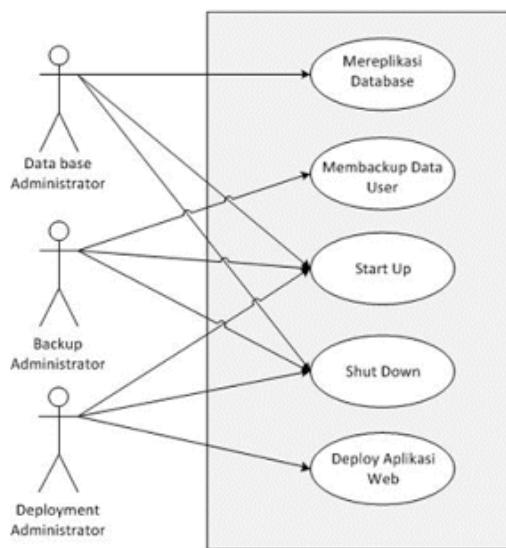
Beberapa relasi yang dapat digunakan dalam *use case* adalah sebagai berikut.

### 1. Generalisasi (*Generalisation*)

Generalisasi pada aktor dan *use case* dimaksudkan untuk menyederhanakan model dengan cara menarik keluar sifat-sifat pada aktor - aktor maupun *use case* – *use case* yang sejenis. Memberikan cara untuk mengetahui kapan dibutuhkan generalisasi, dalam agen yang berbeda dengan satu tujuan yang sama dimana jika lebih dari satu aktor mencoba membangun satu tujuan yang sama kita dapat membuat generalisasi antar aktor tersebut. Yang perlu diperhatikan adalah kemungkinan aktor-aktor yang terlibat memiliki hak akses, kemampuan dan *user interface* yang berbeda tidak dimungkinkan untuk dibentuknya generalisasi.

Sebagai contoh, pada diagram di bawah ini kita melihat ada *database administrator*, *backup administrator* dan *deployment administrator* yang semuanya memiliki tugas yang berbeda dengan jelas, namun demikian, sebagian besar fungsinya saling tumpah tindih oleh karena itu untuk menyederhanakannya diperlukan generalisasi.

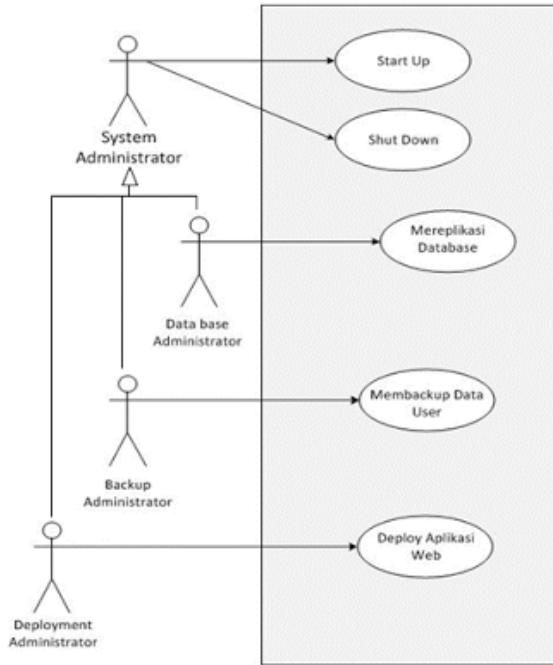
Contoh *use case* tanpa generalisasi dapat dilihat pada Gambar 2.8.



Gambar 2.8 *Use Case* Tanpa Generalisasi

Untuk mengurangi kerumitan diagram *use case* pada gambar di atas, kita dapat menambahkan satu aktor lagi misalnya System Administrator yang menjalankan fungsi *start up* dan *shut down* data *base* yang merupakan

generalisasi dari *database*, *backup* dan *deployment* administrator. Notasi generalisasi dalam UML 2.0 adalah anak panah tertutup.

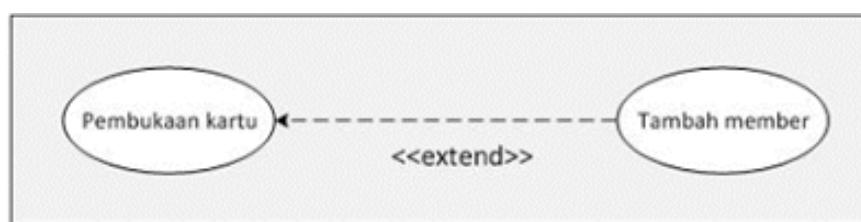


Gambar 2.9 *Use Case* dengan Generalisasi

## 2. Ekstensi (*Extension*)

Ekstensi merupakan *use case* yang lebih kompleks untuk menyederhanakan masalah orisinal dan karena untuk menyederhanakan masalah orisinal dan karena itu memperluas fungsinya. Hubungan antara ekstensi *use case* dan *use case* yang diperluas disebut *extend relationship*, diberi simbol “<<extend>>” dan hubungannya berupa garis putus-putus berpanah terbuka.

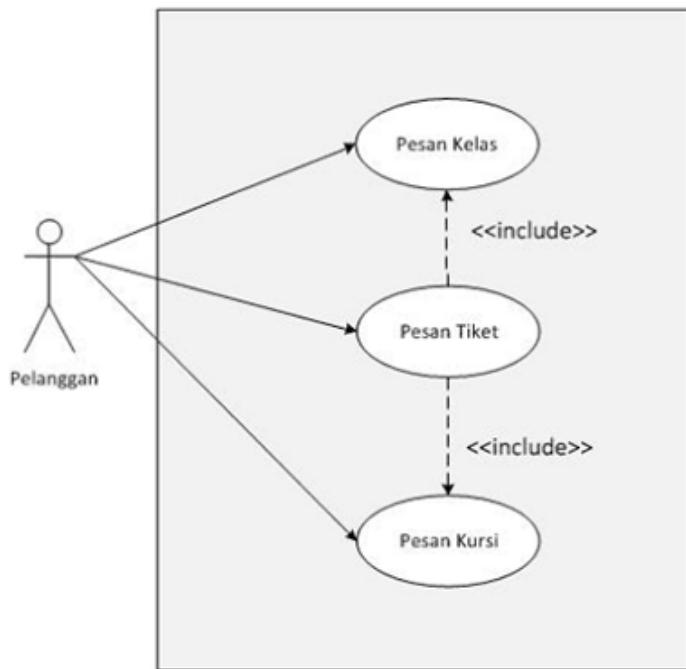
Dalam aplikasi perbankan, misalnya kartu kredit Bank Niaga, selain kita buat *use case* “Pembukaan Kartu Kredit” yang menyatakan bagaimana kredit kredit baru dibuka di bank tersebut, kita juga membuat *use case* “Tambah Member” karena bank tersebut juga menyediakan kartu kredit khusus untuk anak dan istri pengguna kartu baru itu. Gambar berikut ini menggambarkan diagram *use case*-nya.



Gambar 2.10 *Use Case* dengan Ekstensi

### 3. Inklusi (*Inclusion*)

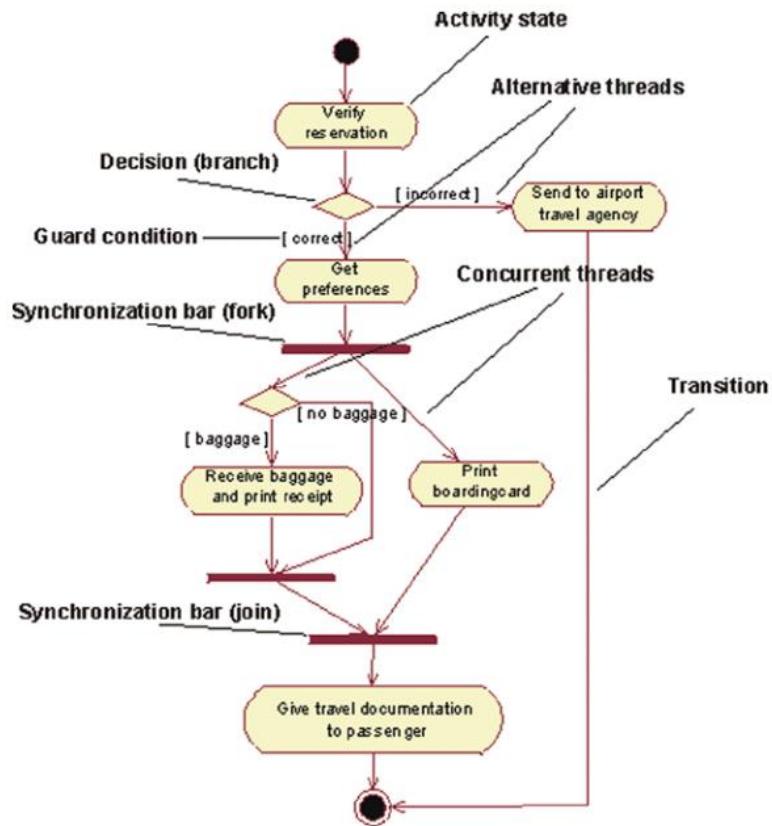
*Use case* dasar yang diinklusi tidak lengkap, berbeda dengan *use case* dasar yang akan diekstensi. Sehingga *use case* inklusi bukan merupakan *use case* optimal dan tidak boleh tidak dijalankan. Simbol hubungan inklusi adalah garis putus-putus dengan anak panah terbuka dan diberi keterangan “<<include>>”.



Gambar 2.11 *Use Case* dengan Inklusi

### 2.18.3 *Activity Diagram*

Seperti yang dikutip oleh Mujilan (2013) Pada dasarnya *activity diagram* adalah suatu ilustrasi sederhana akan apa yang terjadi dalam aliran kerja, aktivitas apa yang dapat dilakukan secara paralel, dan apakah terdapat jalur alternatif dalam aliran kerja (Ericsson, 2004). Berikut ini adalah contoh bentuk diagram dasar dalam UML: *Activity Diagram*. Dicantohkan pula suatu aktivitas diagram untuk kasus bisnis *airport check-in*.



Gambar 2.12 Contoh *Activity Diagram* Airport Check-In

Penjelasan bentuk simbol-simbol pada *activity diagram* adalah sebagai berikut.

1. *Activity states* (pernyataan aktivitas), merepresentasikan bentuk atas langkah yang akan dilaksanakan dalam aliran kerja.
2. *Transitions* (perpindahan), menunjukkan pernyataan aktivitas yang terjadi setelah suatu kegiatan tertentu.
3. *Decisions* (keputusan), merupakan suatu set pengendali (*a set of guard*) atas suatu kondisi yang ditentukan.
4. *Synchronization bars* (garis sinkronisasi). Dapat digunakan untuk menunjukkan sub aliran secara paralel.

#### 2.18.4 Class Diagram

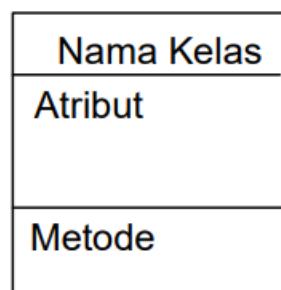
Fowler (2005) seperti yang dikutip oleh Mujilan (2013) menyebutkan bahwa *class diagram* mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang terdapat di antara mereka. *Class diagram* juga menunjukkan properti dan operasi sebuah *class* dan batasan-batasan yang terdapat

dalam hubungan-hubungan objek tersebut. UML menggunakan istilah fitur sebagai istilah umum yang meliputi properti dan operasi sebuah *class*.

Kuske et al. (2009) seperti yang dikutip oleh Mujilan (2013) menyatakan bahwa *class diagrams* digunakan untuk merepresentasikan struktur statis (*static structure*) dari sistem berorientasi objek (*object-oriented systems*). Diagram kelas ini merupakan bagian dalam menjelaskan struktur-struktur objek dasar (*the fundamental object structures*).

Menurut Kuske et al. (2009) seperti yang dikutip oleh Mujilan (2013) sebuah *class* terdiri dari :

1. Nama (*a name*)
2. *Set atribut* (*a set of attributes*)
3. Metode atau *set* operasi (*a set of operations*)



Gambar 2.13 Bentuk Umum *Class Diagram*

## 2.19 *Black-Box Testing*

Seperti yang dikutip oleh Sinaga (2015) dalam jurnalnya menurut Efendi (2011) menjelaskan bahwa “Pengujian *Black Box* adalah pengujian yang dilakukan terhadap *interface* tertentu untuk menguji bahwa fungsi-fungsi *interface* tersebut bekerja dengan baik”.

*Black-Box Testing* bukanlah solusi alternatif dari *White-Box Testing* tapi lebih merupakan pelengkap untuk menguji hal-hal yang tidak dicakup oleh *White-Box Testing* (Sukamto Ariani, 2009).

*Black-Box Testing* cenderung untuk menemukan hal-hal berikut (Sukamto Ariani, 2009).

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (*interface errors*).
3. Kesalahan pada struktur data dan akses basis data.

4. Kesalahan performansi (*performance errors*).
5. Kesalahan inisialisasi dan terminasi.

Pengujian didesain untuk menjawab pertanyaan-pertanyaan berikut:

1. Bagaimana fungsi-fungsi diuji agar dapat dinyatakan valid?
2. *Input* seperti apa yang dapat menjadi bahan kasus uji yang baik?
3. Apakah sistem sensitif pada *input-input* tertentu?
4. Bagaimana sekumpulan data dapat diisolasi?
5. Berapa banyak rata-rata data dan jumlah data yang dapat ditangani sistem?
6. Efek apa yang dapat membuat kombinasi data ditangani spesifik pada operasi sistem?

## **BAB III**

### **METODOLOGI PENELITIAN**

Metodologi penelitian merupakan bab yang berisikan langkah-langkah yang digunakan dalam pembuatan aplikasi dalam tugas akhir ini. Dengan sistematika ini proses pembuatan aplikasi dapat dipahami dan diikuti oleh pihak lain. Penelitian ini dibuat berdasarkan pengamatan data-data yang didapatkan.

#### **3.1 Metode Penelitian**

Dalam pengembangan aplikasinya penulis menggunakan metode pengembangan perangkat lunak SDLC (*System Development Life Cycle*). Seperti yang dikutip oleh Saleh (2016), SDLC (*System Development Life Cycle*) adalah aplikasi penerapan dari penemuan permasalahan (*problem solving*) yang didapat dari pendekatan sistem (*system approach*) menjadi pengembangan dari solusi sistem informasi terhadap masalah bisnis (O'brien, 2000).

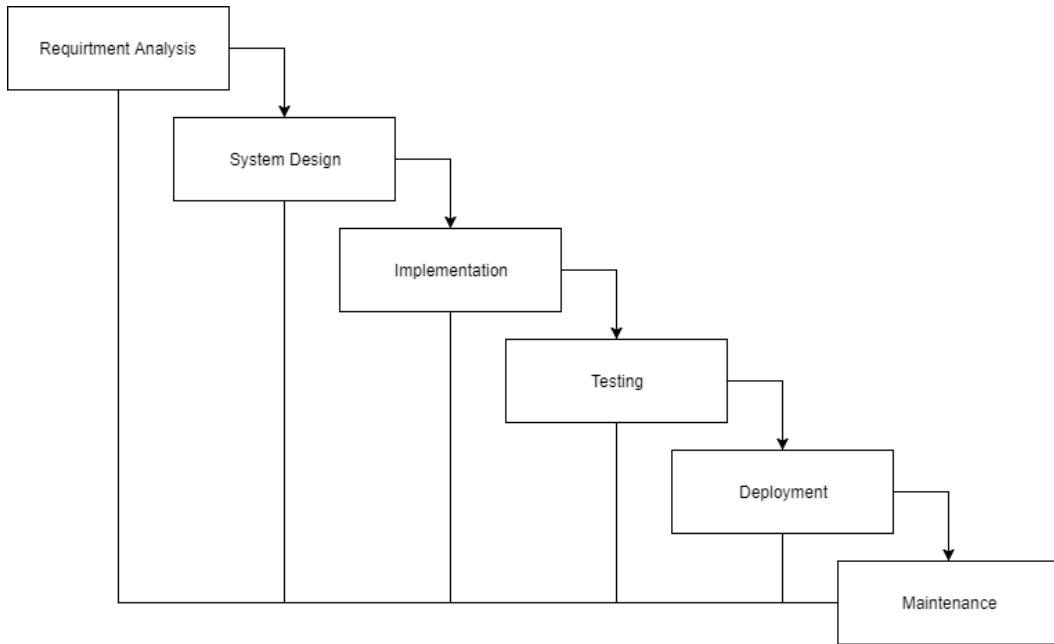
Menurut Turban (2003) seperti yang dikutip oleh Saleh (2016), SDLC adalah metode pengembangan sistem tradisional yang digunakan sebagian besar organisasi saat ini. SDLC adalah kerangka kerja (*framework*) yang terstruktur yang berisi proses-proses sekuensial di mana sistem informasi dikembangkan.

Sedangkan menurut Azhar Susanto (2004) seperti yang dikutip oleh Saleh (2016) menyatakan bahwa SDLC adalah salah satu metode pengembangan sistem informasi yang populer pada saat sistem informasi pertama kali dikembangkan.

Model *waterfall* adalah salah satu metode dalam SDLC. Seperti dikutip oleh Ayu Langening Tyas (2016), Model *waterfall* merupakan model klasik yang sederhana dengan aliran sistem yang linier. *Output* dari setiap tahap merupakan *input* bagi tahap berikutnya (Kristanto, 2004). Artinya pada tiap-tiap tahap dalam model *waterfall* harus dilakukan secara maksimal karena tidak adanya penggerjaan yang sifatnya paralel.

### 3.1.1 Desain Penelitian

Langkah-langkah pembuatan aplikasi e-market hiburan untuk kebutuhan hajatan berbasis web dapat ditunjukan pada Gambar 3.1.



Gambar 3.1 Desain Penelitian

Tahapan-tahapan metode waterfall SDLC adalah sebagai berikut.

#### 1. *Requirement Analysis*

*Requirement analysis* atau analisis kebutuhan merupakan tahapan menganalisis kebutuhan data, mengumpulkan data dan menyiapkan semua kebutuhan yang akan digunakan dalam penelitian.

#### 2. *System Design*

*System design* atau desain sistem adalah tahapan untuk persiapan rancangan bangun implementasi yang menggambarkan bagaimana suatu sistem itu dibentuk yang berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh.

#### 3. *Implementation*

Tahap implementasi adalah tahap dimana semua elemen dan aktifitas sistem disatukan, dengan kata lain tahapan ini disebut juga tahapan pengkodingan yang dimana penulis menulis kode-kode program untuk merealisasikan desain sistem pada tahap sebelumnya.

#### **4. Testing**

Setelah semua proses pada tahap sebelumnya selesai maka tahap selanjutnya adalah tahapan testing. Tahapan testing adalah tahapan pengujian yang dilakukan penulis untuk menguji kelayakan aplikasi apakah aplikasi sudah sesuai dengan desain dan masih ada kesalahan atau tidak.

#### **5. Deployment**

Tahap ini adalah tahap yang dilakukan penulis untuk menyebarluaskan aplikasinya kepada masyarakat, yaitu pemasangan *web* pada *web server*.

#### **6. Maintenance**

Tahap ini adalah tahap yang diperlukan untuk terus memenuhi kebutuhan pengguna aplikasi dan juga perbaikan serta pengembangan aplikasi yang sudah dibuat karena aplikasi yang dibuat tidak akan selamanya seperti itu.

### **3.2 Metode Pengumpulan Data**

#### **3.2.1 Observasi**

Penulis melakukan pengamatan dari hasil data-data yang didapatkan melalui kuisioner-kuisioner dan data-data dari Dinas Kebudayaan dan Pariwisata Kabupaten Indramayu yang dikemudian penulis menyimpulkannya ke dalam proses yang terstruktur untuk di implementasikan ke dalam sistem aplikasi.

#### **3.2.2 Studi Pustaka**

Penulis mencari bahan-bahan atau referensi yang dapat digunakan sebagai referensi penulis untuk merancang sistem aplikasi yang dapat digunakan oleh masyarakat dalam bertransaksi. Referensi tersebut baik dari *website-website* sejenis dengan penelitian penulis seperti *website* tokopedia, bukalapak, dan facebook.

### **3.3 Analisis Kebutuhan Sistem**

Dalam pembuatan aplikasi e-market hiburan untuk kebutuhan hajatan ini terdapat kebutuhan baik dalam bentuk *hardware* maupun *software*. Adapun kebutuhan-kebutuhan tersebut dijelaskan pada sub bab selanjutnya.

### 3.3.1 Kebutuhan *Hardware*

Adapun kebutuhan *hardware* untuk membuat aplikasi e-market hiburan untuk kebutuhan hajatan ini terdapat pada Tabel 3.1.

Tabel 3.1 Daftar Kebutuhan *Hardware* Laptop

No.	Jenis <i>Hardware</i>	Kebutuhan <i>Hardware</i>	Keterangan
1	<i>Processor</i>	2.0 GHz <i>up to above</i>	Disarankan agar tidak lambat karena menggunakan software <i>high performance</i>
2	RAM	2 GB <i>up to above</i>	Disarankan agar tidak lambat untuk menjalankan beberapa aplikasi
3	Harddisk	180 GB <i>up to above</i>	Disarankan karena aplikasi menyimpan <i>cache file</i> yang cukup besar

### 3.3.2 Kebutuhan *Software*

Kebutuhan *software* merupakan kebutuhan yang harus terpenuhi dalam membuat aplikasi ini. Adapun kebutuhan *software* dalam pembuatan aplikasi ini terdapat pada Tabel 3.2.

Tabel 3.2 Daftar Kebutuhan *Software* Laptop

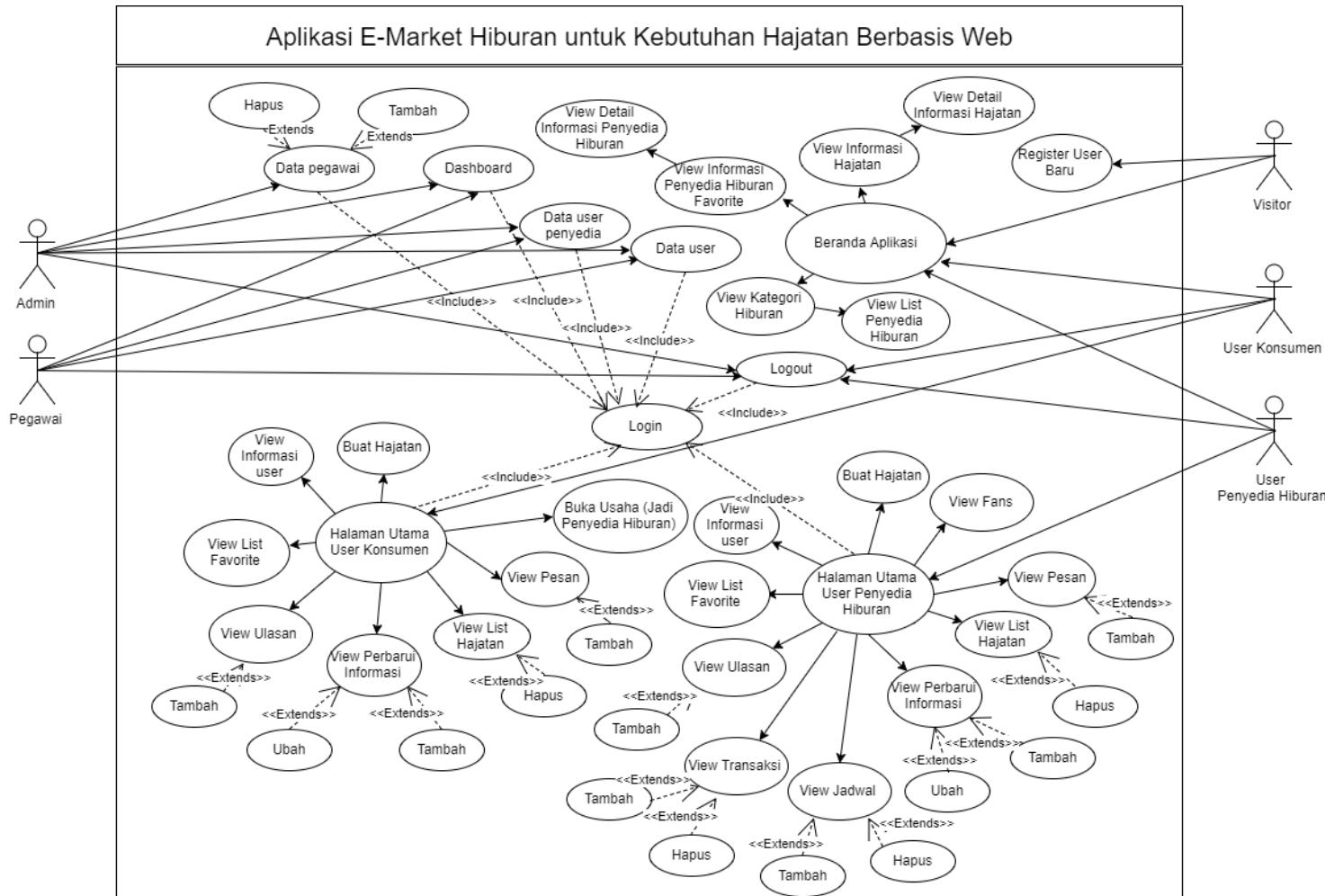
No.	Jenis <i>Software</i>	Kebutuhan <i>Software</i>
1	Sistem Operasi	<i>Windows 7 Profesional up to above</i>
2	Desain Aplikasi	Balsamiq Mockups, CorelDraw X3 <i>up to above</i>
3	Bahasa <i>Scripting</i>	PHP 5.6 ( <i>Hypertext Preprocessor</i> ), HTML 5, CSS 3, Javascript, JQuery
4	<i>Software Pengolah</i>	Dreamweaver CS5.5 <i>up to above</i>
5	Penyimpan Data	Database (MySQL)
6	<i>Web browser</i>	Chrome, Firefox, Microsoft Edge, dan lain-lain

## 3.4 Perancangan Sistem dengan UML

Perancangan sistem ini menggunakan UML (*Unified Modelling Language*). Pada perancangan ini akan digambarkan secara garis besar mengenai aplikasi e-market hiburan untuk kebutuhan hajatan yang akan dijelaskan pada sub bab selanjutnya.

### **3.4.1 Use Case Diagram**

*Use case diagram* merupakan *graphical* yang mendeskripsikan fungsionalitas sistem yang seharusnya dilakukan sesuai yang diinginkan *external actors*. Berikut adalah *use case diagram* dari aplikasi e-market hiburan untuk kebutuhan hajatan terdapat pada Gambar 3.2.



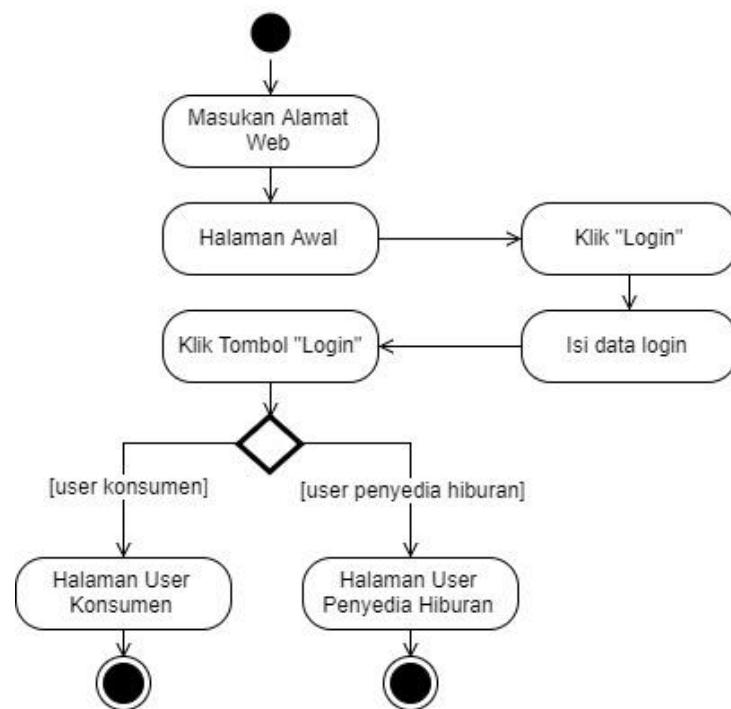
Gambar 3.2 *Use Case Diagram*

### 3.4.2 Activity Diagram

*Activity diagram* merupakan gambaran aliran aktifitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, *decision* (keputusan) yang mungkin terjadi dan bagaimana itu berakhiri. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi dalam sistem.

#### 1. Activity Diagram Login User

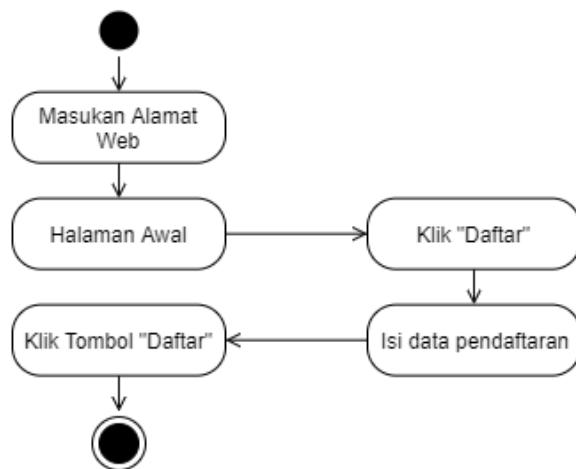
User konsumen/penyedia hiburan harus melakukan login untuk dapat menggunakan fitur-fitur yang ada pada sistem. *Login user* dikelompokan menjadi dua jenis *user* yaitu *user* konsumen dan *user* penyedia hiburan. Rancangan *activity diagram* *login user* terdapat pada Gambar 3.3.



Gambar 3.3 Activity Diagram Login User

#### 2. Activity Diagram Register

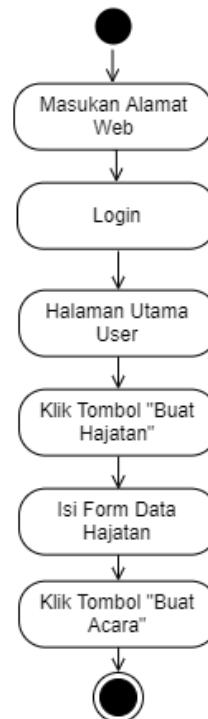
Pengunjung yang ingin menggunakan aplikasi ini untuk dapat mempromosikan hiburan-hiburannya/keseniannya atau pengunjung yang ingin melakukan penyewaan hiburan/kesenian harus terdaftar sebagai anggota dalam sistem. *Activity diagram register* yang menggambarkan alur pendaftaran pengunjung terdapat pada Gambar 3.4.



Gambar 3.4 Activity Diagram Register User Baru

### 3. Activity Diagram Buat Hajatan

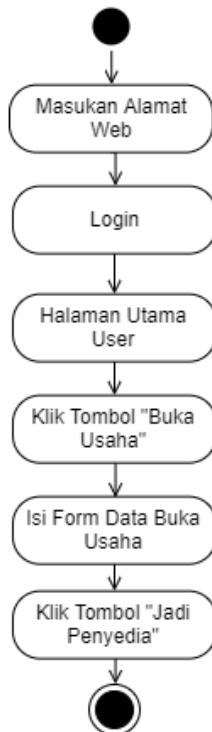
User konsumen yang ingin menampilkan acara hajatan pada sistem aplikasi dan ingin melakukan penyewaan hiburan melalui sistem, harus melakukan buat hajatan. Adapun *activity* buat hajatan terdapat pada Gambar 3.5.



Gambar 3.5 Activity Diagram Buat Hajatan

### 4. Activity Diagram Buka Usaha

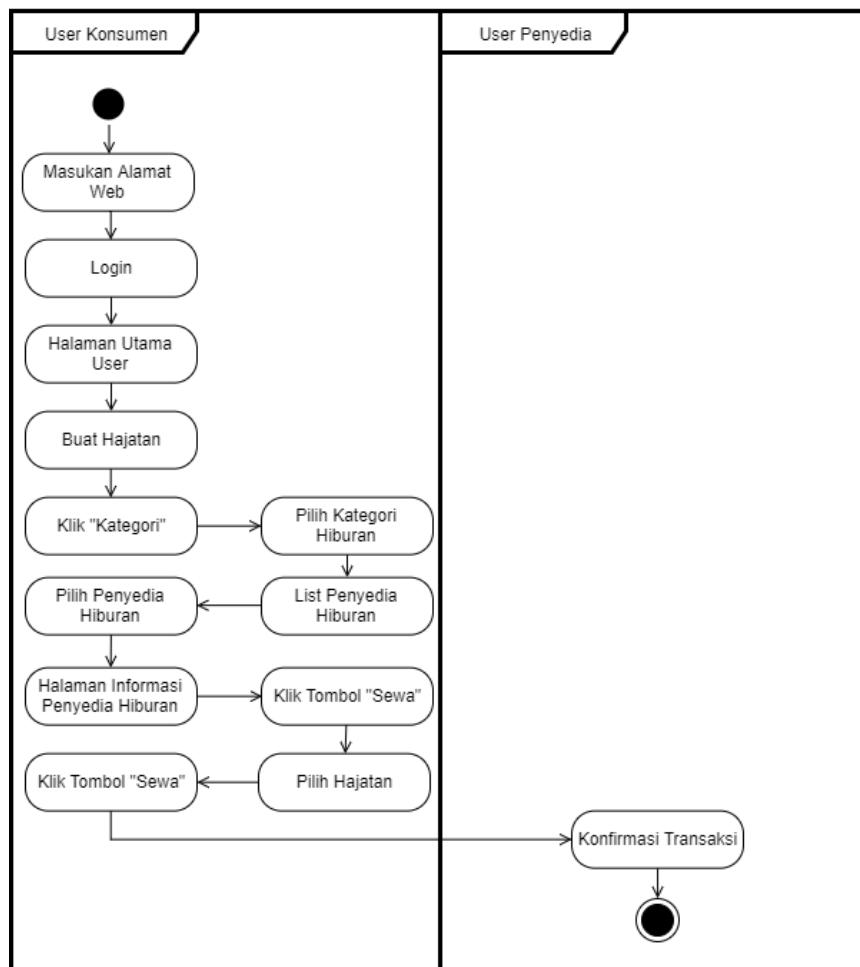
User yang berkeinginan untuk membuka usaha penyewaan hiburan dalam sistem ini dapat dilakukan dengan cara buka usaha yang ditunjukkan pada Gambar 3.6.



Gambar 3.6 *Activity Diagram Buka Usaha*

##### **5. *Activity Diagram Sewa Hiburan***

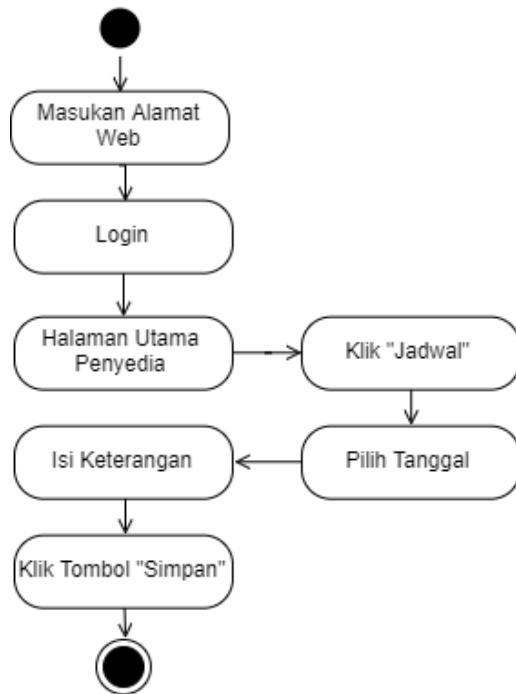
Sewa hiburan merupakan aktifitas yang melibatkan dua pelaku aktor dimana *user* yang telah membuat hajatan maka dapat melakukan penyewaan hiburan dan apabila sudah melakukan sewa hiburan dalam sistem ini maka tahap selanjutnya menunggu konfirmasi dari penyedia hiburannya. Adapun *activity* sewa hiburan terdapat pada Gambar 3.7.



Gambar 3.7 *Activity Diagram* Sewa Hiburan

## 6. *Activity Diagram* Tambah Jadwal

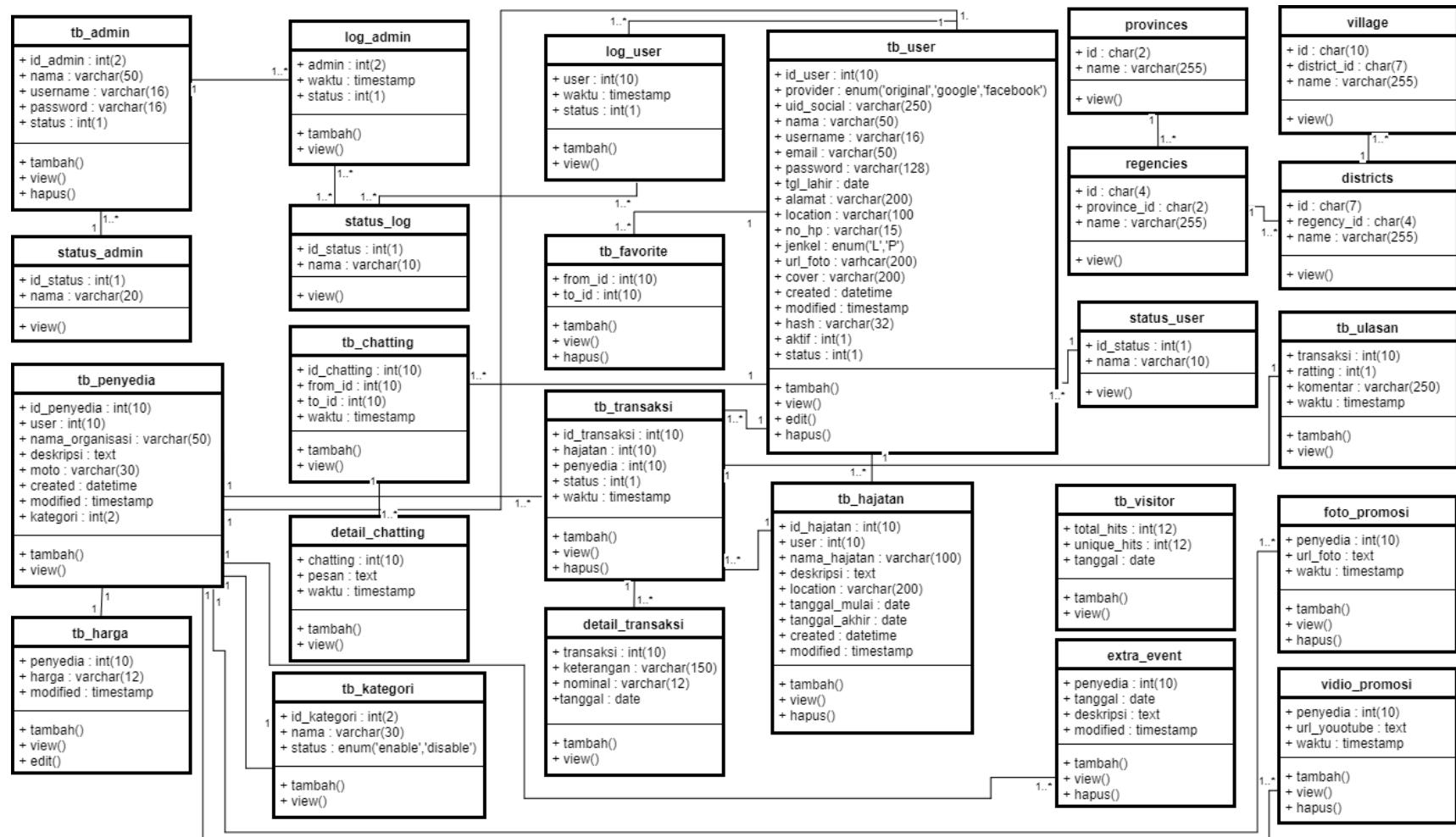
Penyedia hiburan yang ingin menginformasikan jadwal penyewaan yang dilakukan diluar sistem ini dapat dilakukan dengan melakukan tambah jadwal. Adapun *activity* tambah jadwal terdapat pada Gambar 3.8.



Gambar 3.8 *Activity Diagram* Tambah Jadwal

### 3.4.3 *Class Diagram*

*Class diagram* menunjukkan interaksi antar kelas dalam sistem. *Class diagram* juga memperlihatkan struktur dari sistem berupa sistem kelas, attribut, dan hubungan antar kelas. Adapun *class diagram* dari aplikasi ini terdapat pada Gambar 3.9.



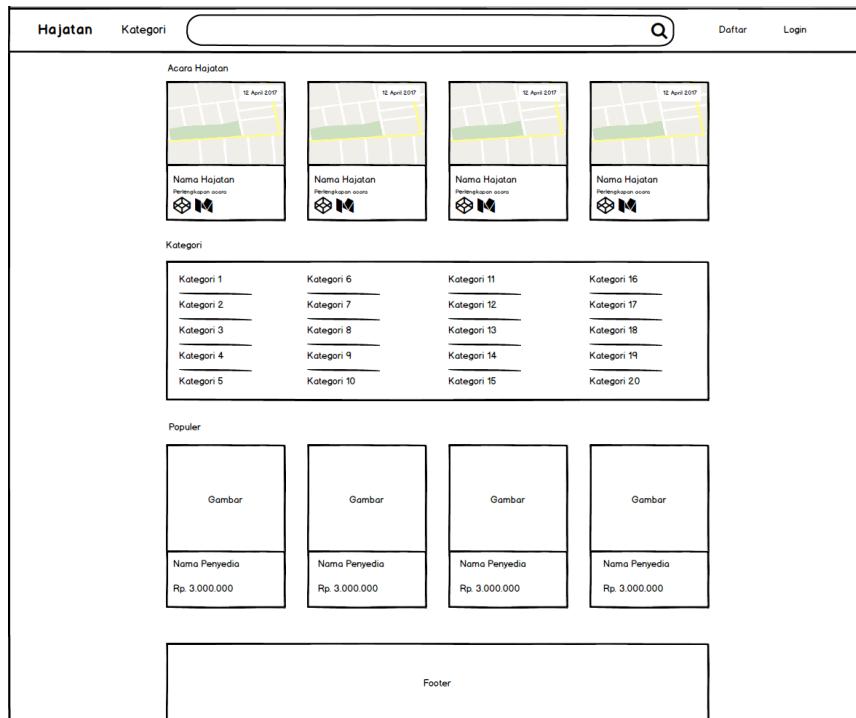
Gambar 3.9 Class Diagram

### 3.5 Perancangan Antar Muka

Beberapa rancangan desain antar muka dari aplikasi e-market hiburan untuk kebutuhan hajatan yang akan dijelaskan pada sub bab berikut.

#### 3.5.1 Rancangan Halaman Awal

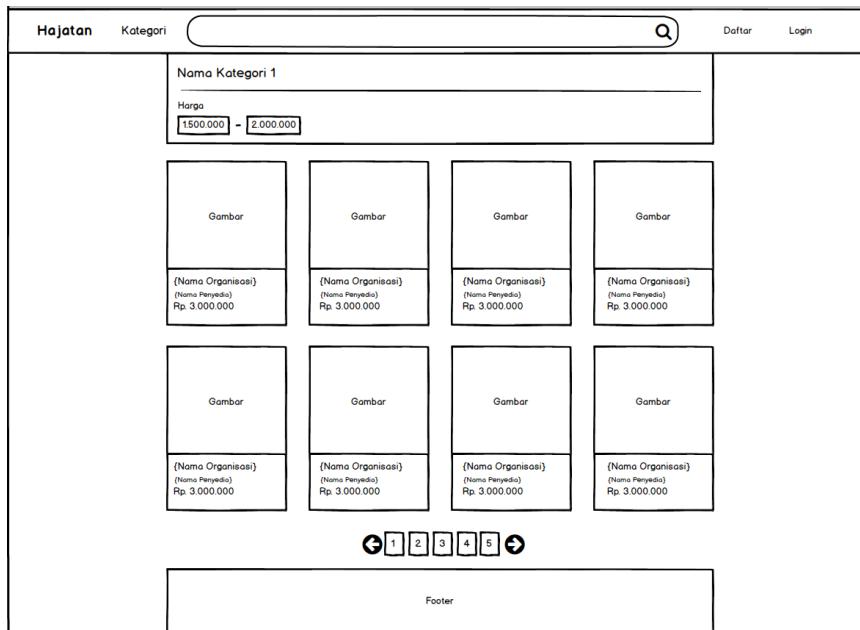
Halaman awal adalah halaman yang pertama akan dilihat oleh pengunjung website. Adapun rancangan halaman awal terdapat pada Gambar 3.10.



Gambar 3.10 Rancangan Halaman Awal

#### 3.5.2 Rancangan Halaman *List Penyedia Hiburan*

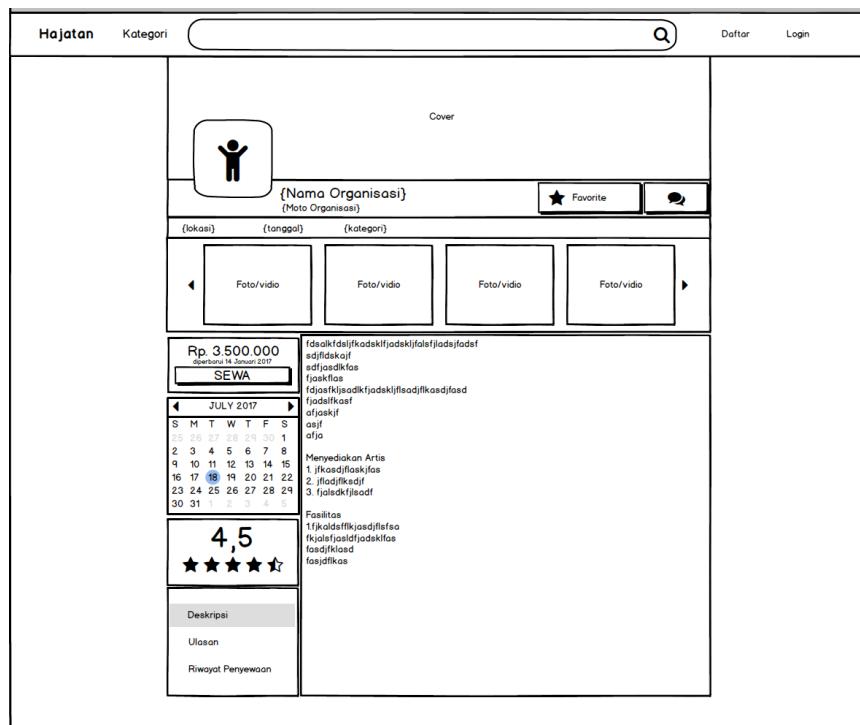
Halaman *list* penyedia hiburan adalah halaman untuk melihat penyedia-hiburan yang ada pada kategori yang telah dipilih. Adapun rancangan halaman *list* penyedia hiburan terdapat pada Gambar 3.11.



Gambar 3.11 Rancangan Halaman *List* Penyedia Hiburan

### 3.5.3 Rancangan Halaman Deskripsi Penyedia Hiburan

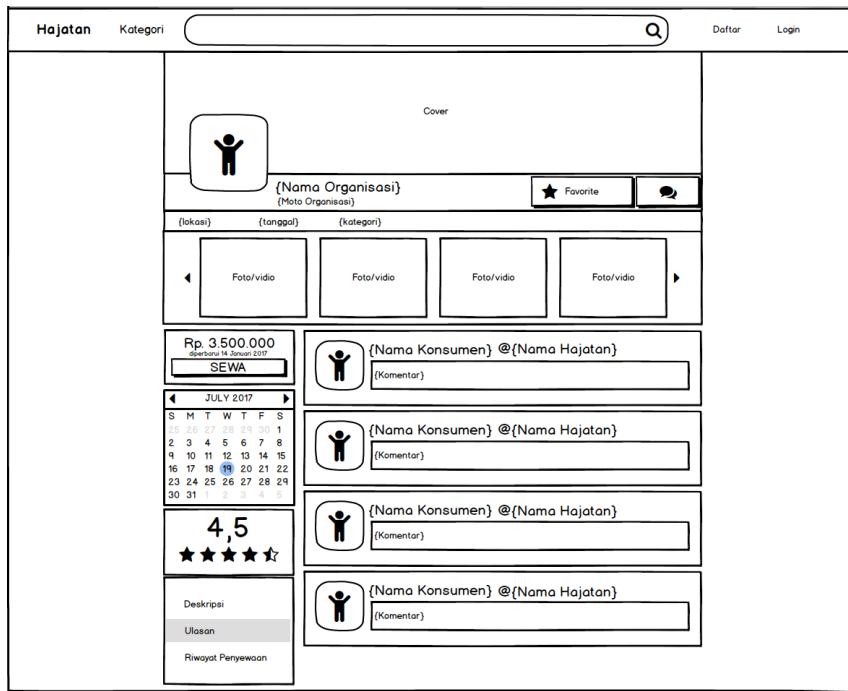
Halaman deskripsi penyedia hiburan merupakan halaman yang berfungsi untuk memberikan informasi kepada pengunjung atau *user* yang ingin melihat deskripsi atau informasi dari penyedia hiburan. Adapun rancangan halaman deskripsi penyedia hiburan terdapat pada Gambar 3.12.



Gambar 3.12 Rancangan Halaman Deskripsi Penyedia Hiburan

### 3.5.4 Rancangan Halaman Ulasan Penyedia Hiburan

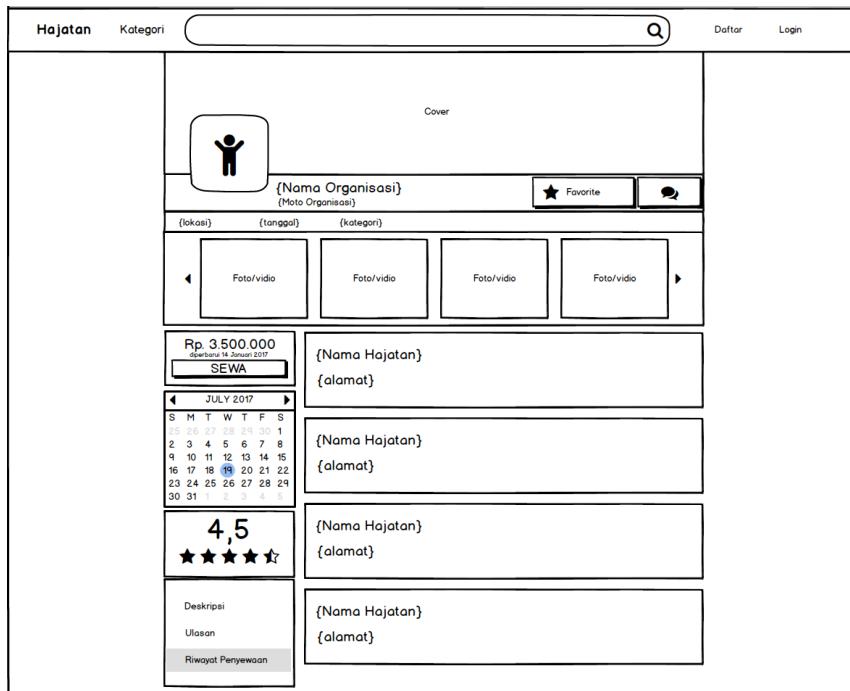
Halaman ulasan penyedia hiburan difungsikan untuk memberikan informasi tentang kualitas dari penyedia hiburan yang dimana *input* tersebut berasal dari penyewa yang telah menyewa penyedia hiburan tersebut. Adapun rancangan halaman ulasan penyedia hiburan terdapat pada Gambar 3.13.



Gambar 3.13 Rancangan Halaman Ulasan Penyedia Hiburan

### 3.5.5 Rancangan Halaman Riwayat Penyewaan Penyedia Hiburan

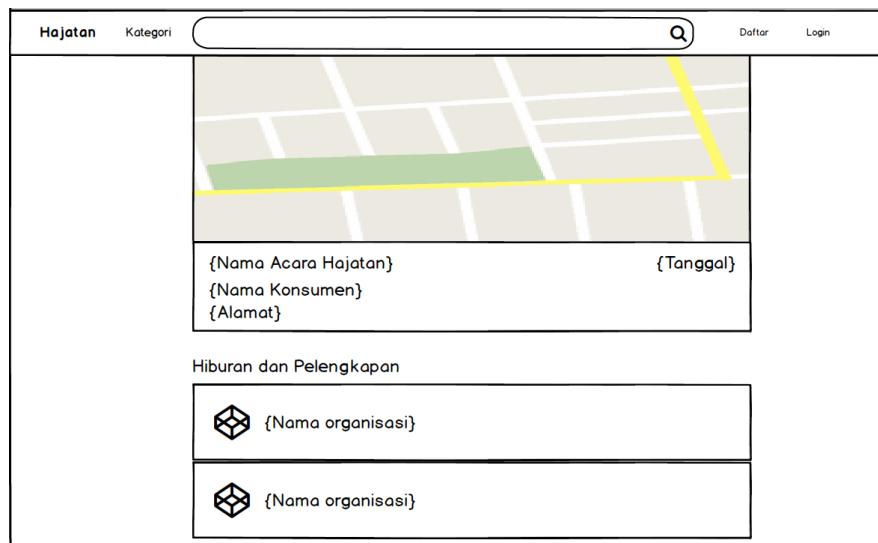
Halaman riwayat penyewaan penyedia hiburan merupakan halaman yang berfungsi untuk menampilkan siapa-siapa saja yang telah melakukan penyewaan hiburan tersebut. Adapun rancangan halaman ulasan penyedia hiburan terdapat pada Gambar 3.14.



Gambar 3.14 Rancangan Halaman Riwayat Penyewaan Penyedia Hiburan

### 3.5.6 Rancangan Halaman Detail Informasi Hajatan

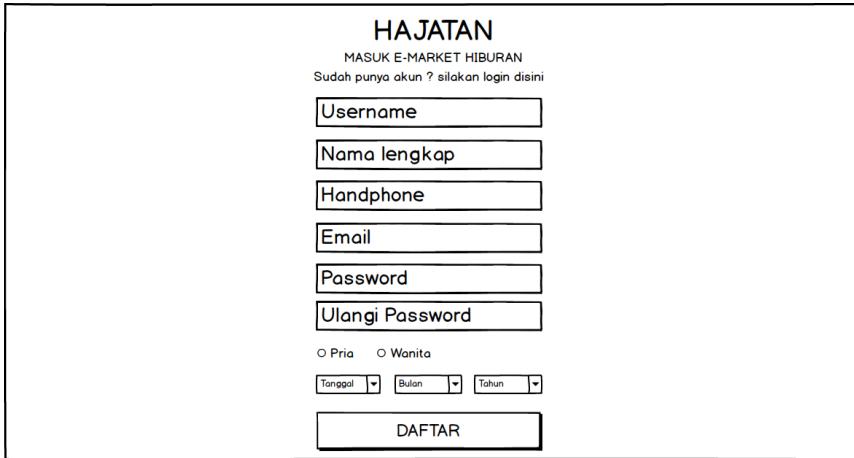
Rancangan halaman detail informasi hajatan adalah halaman yang berfungsi untuk memberikan informasi kepada para pengguna aplikasi tentang detail dari suatu acara hajatan, seperti lokasi hajatan, penyewaan hiburan, dan kapan hajatan itu akan dilaksanakan. Adapun rancangan halaman detail informasi hajatan terdapat pada Gambar 3.15.



Gambar 3.15 Rancangan Halaman Detail Informasi Hajatan

### 3.5.7 Rancangan Halaman *Register*

Halaman register adalah halaman yang digunakan untuk mendaftar sebagai anggota/*user* dari aplikasi e-market hiburan untuk kebutuhan hajatan ini. Adapun rancangan halaman register terdapat pada Gambar 3.16.



Rancangan Halaman Register (Gambar 3.16) menunjukkan formulir pendaftaran dengan tampilan sebagai berikut:

- Kop: HAJATAN  
MASUK E-MARKET HIBURAN  
Sudah punya akun ? silakan login disini
- Formulir pendaftaran:
  - Username
  - Nama lengkap
  - Handphone
  - Email
  - Password
  - Ulangi Password
- Pilihan gender:
  - Pria
  - Wanita
- Pilih tanggal lahir:
  - Tanggal
  - Bulan
  - Tahun
- Tombol DAFTAR

Gambar 3.16 Rancangan Halaman *Register*

### 3.5.8 Rancangan Halaman *Login*

Halaman *login* adalah halaman yang digunakan pengunjung untuk *login* ke dalam aplikasi untuk mendapatkan hak akses *user* konsumen/penyedia hiburan. Adapun rancangan halaman *login* terdapat pada Gambar 3.17.



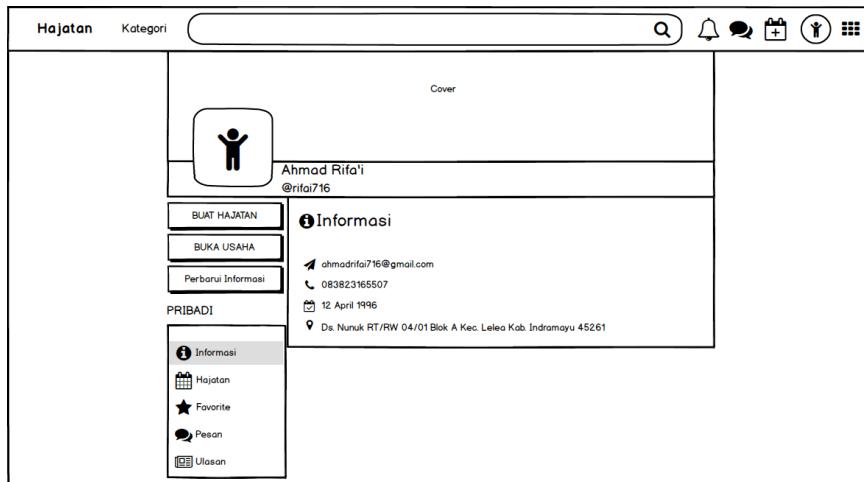
Rancangan Halaman Login (Gambar 3.17) menunjukkan formulir login dengan tampilan sebagai berikut:

- Kop: HAJATAN  
MASUK E-MARKET HIBURAN  
Belum punya akun ? silakan daftar disini
- Formulir login:
  - username / email
  - password
- Tombol MASUK

Gambar 3.17 Rancangan Halaman *Login*

### 3.5.9 Rancangan Halaman Informasi *User*

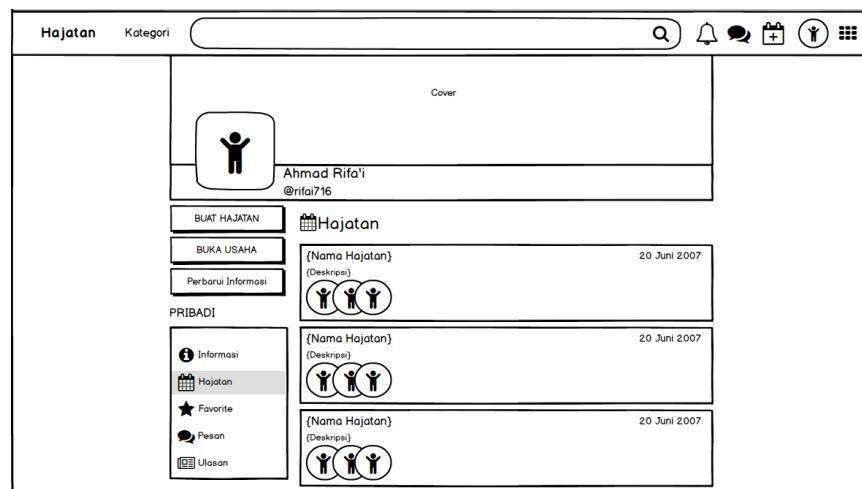
Halaman informasi *user* difungsikan untuk menampilkan informasi biodata diri dari *user* yang sudah melakukan *login* ke dalam aplikasi. Adapun rancangan halaman informasi *user* terdapat pada Gambar 3.18.



Gambar 3.18 Rancangan Halaman Informasi *User*

### 3.5.10 Rancangan Halaman *List Hajatan User*

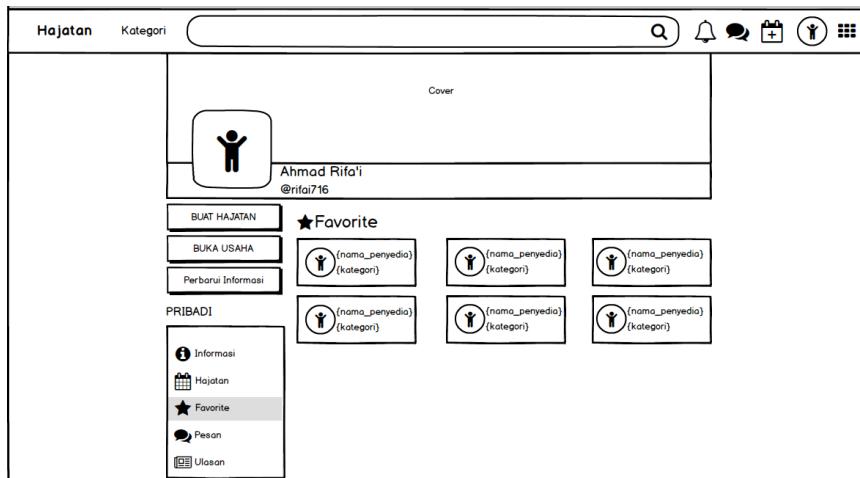
Halaman *list hajatan user* adalah halaman yang digunakan untuk menampilkan *list* hajatan yang telah dilakukan atau belum dilakukan. Adapun rancangan halaman *list hajatan user* terdapat pada Gambar 3.19.



Gambar 3.19 Rancangan Halaman *List Hajatan User*

### 3.5.11 Rancangan Halaman *List Favorite*

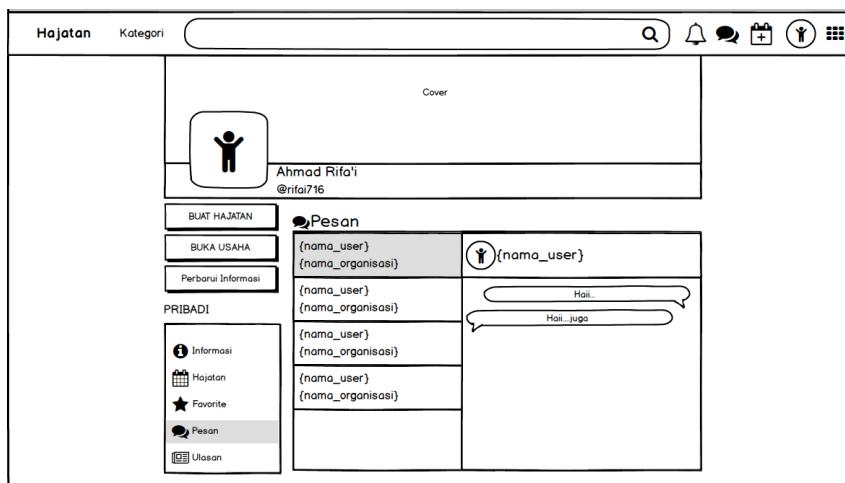
Halaman *list favorite* adalah halaman yang berfungsi untuk menampilkan penyedia-penyedia yang telah difavoritkan. Adapun rancangan halaman *List Favorite* terdapat pada Gambar 3.20.



Gambar 3.20 Rancangan Halaman *List Favorite*

### 3.5.12 Rancangan Halaman Pesan

Halaman pesan adalah halaman yang digunakan untuk berkirim pesan dengan penyedia hiburan untuk mendapatkan kesepakatan tentang negosiasi harga penyewaan, tempat janjian untuk melakukan pertemuan transaksi dan sebagainya. Adapun rancangan halaman pesan terdapat pada Gambar 3.21.



Gambar 3.21 Rancangan Halaman Pesan

### 3.5.13 Rancangan Halaman Ulasan

Halaman ulasan adalah halaman yang digunakan untuk melihat ulasan kepada penyedia hiburan dan juga untuk menambahkan ulasan baru. Adapun rancangan halaman ulasan terdapat pada Gambar 3.22.

Rancangan Halaman Ulasan (Review Page Design) menunjukkan tampilan halaman ulasan. Di bagian atas terdapat menu 'Hajatan' dan 'Kategori' dengan ikon pencarian, notifikasi, pesan, dan profil. Pada bagian kiri terdapat sidebar dengan opsi: BUAT HAJATAN, BUKA USAHA, Perbarui Informasi, PRIJADI, dan PRIBADI. Di bawahnya ada ikon untuk Informasi, Hajatan, Favorite, Pesan, dan Ulasan. Di bagian tengah ada foto profil Ahmad Rifa'i (@rifai716) dan tombol 'Cover'. Berikutnya terdapat dua panel ulasan: 'Ulasan' dan 'Terulus'. Panel 'Ulasan' menampilkan data ulasan dengan nama pengguna (nama\_hajatan), nama organisasi (nama\_organisasi), rating (Kualah bintang), dan komentar (Komentar tentang penyelesaian...). Tombol 'Simpan' terdapat di bawahnya. Panel 'Terulus' menampilkan data ulasan yang sama dengan rating yang lebih tinggi (★★★★★) dan komentar ('Oke ...oke bagus...').

Gambar 3.22 Rancangan Halaman Ulasan

### 3.5.14 Rancangan Halaman Perbarui Informasi

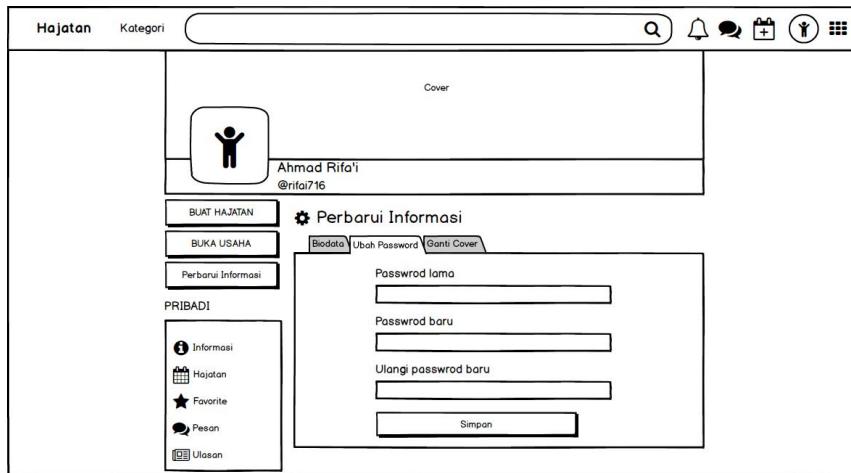
Halaman perbarui informasi merupakan halaman yang digunakan untuk *update* infomasi biodata diri dan juga untuk mengganti foto profil. Adapun rancangan halaman perbarui informasi terdapat pada Gambar 3.23.

Rancangan Halaman Perbarui Informasi (Update Information Page Design) menunjukkan tampilan halaman perbarui informasi. Di bagian atas terdapat menu 'Hajatan' dan 'Kategori' dengan ikon pencarian, notifikasi, pesan, dan profil. Pada bagian kiri terdapat sidebar dengan opsi: BUAT HAJATAN, BUKA USAHA, Perbarui Informasi, PRIJADI, dan PRIBADI. Di bawahnya ada ikon untuk Informasi, Hajatan, Favorite, Pesan, dan Ulasan. Di bagian tengah terdapat form 'Perbarui Informasi' dengan sub-panel 'Biodata'. Form ini meminta inputan: Nama Lengkap (Ahmad Rifa'i), Email (ahmadrifai716@gmail.com), No. Handphone (083823165507), Tanggal Lahir (12 April 1996), Alamat (Provinsi, Kabupaten, Kecamatan, Desa, RT, RW, Kode pos), Jenis kelamin (Laki-Laki), dan tombol 'Simpan'. Terdapat pula ikon untuk 'Ubah Foto' dan tombol 'Ubah Password' dan 'Ganti Cover'.

Gambar 3.23 Rancangan Halaman Perbarui Informasi

### 3.5.15 Rancangan Halaman Ubah Password

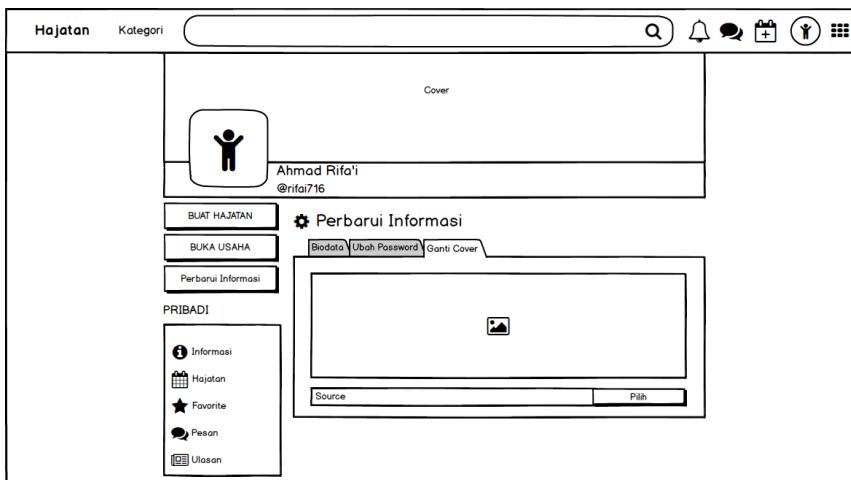
Halaman ubah password adalah halaman yang digunakan untuk mengganti password yang lama dengan password yang baru. Adapun rancangan halaman ubah password terdapat pada Gambar 3.24.



Gambar 3.24 Rancangan Halaman Ubah *Password*

### 3.5.16 Rancangan Halaman Ganti *Cover*

Halaman ganti *cover* adalah halaman yang digunakan untuk mengganti *cover*. Adapun rancangan halaman *cover* terdapat pada Gambar 3.25.



Gambar 3.25 Rancangan Halaman Ganti *Cover*

### 3.5.17 Rancangan Halaman Buat Hajatan

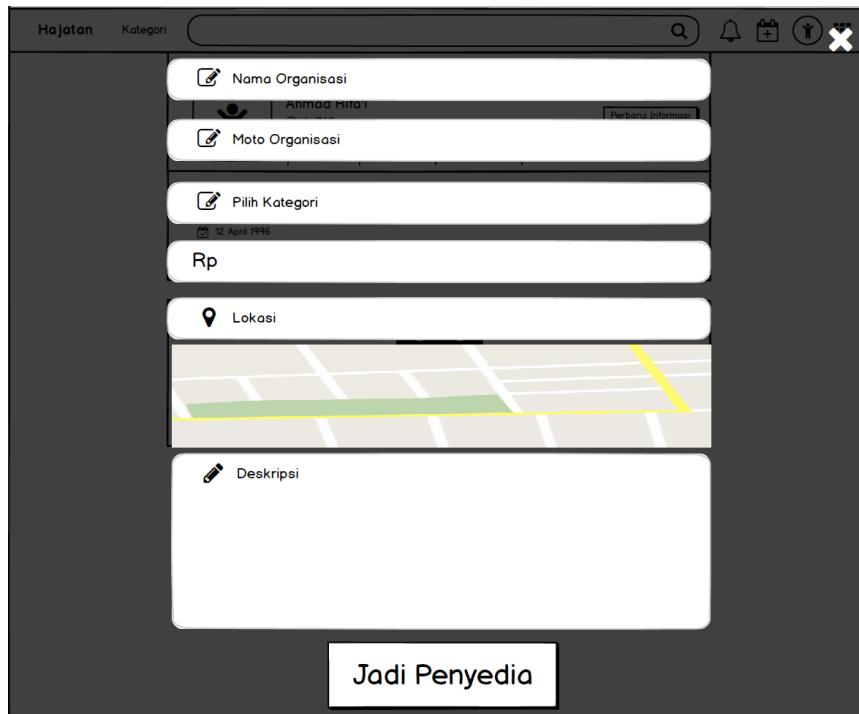
Halaman buat hajatan adalah halaman yang digunakan untuk membuat acara hajatan. Adapun rancangan halaman buat hajatan terdapat pada Gambar 3.26.



Gambar 3.26 Rancangan Halaman Buat Hajatan

### 3.5.18 Rancangan Halaman Buka Usaha

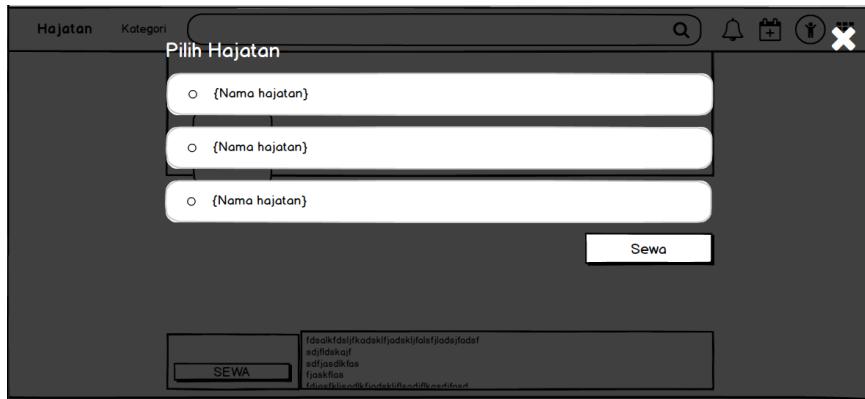
Halaman buka usaha adalah halaman yang digunakan untuk membuka usaha menjadi penyedia hiburan. Adapun rancangan halaman buka usaha terdapat pada Gambar 3.27.



Gambar 3.27 Rancangan Halaman Buka Usaha

### 3.5.19 Rancangan Halaman Sewa Hiburan

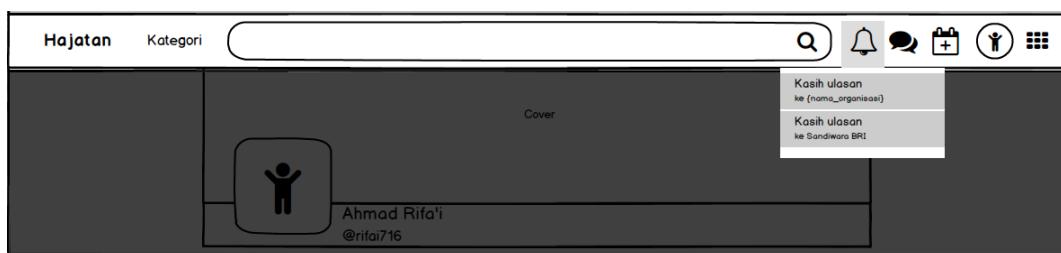
Halaman sewa hiburan adalah halaman yang digunakan untuk melakukan penyewaan hiburan untuk acara hajatan yang telah dibuat. Adapun rancangan halaman sewa hiburan terdapat pada Gambar 3.28.



Gambar 3.28 Rancangan Halaman Sewa Hiburan

### 3.5.20 Rancangan Popover Notification

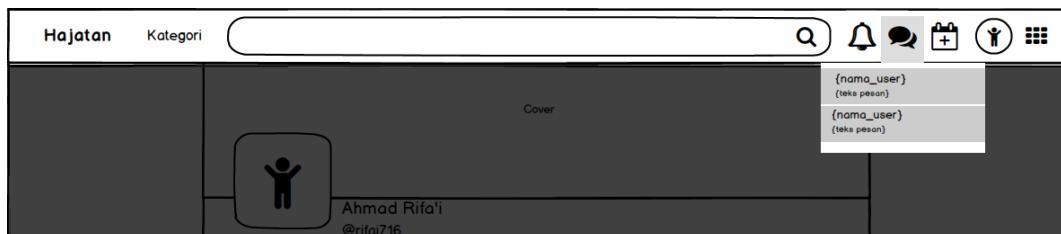
*Popover notification* merupakan navigasi yang digunakan untuk menampilkan pemberitahuan untuk mengulas penyedia hiburan yang disewa setelah acara hajatan selesai. Adapun rancangan *popover notification* terdapat pada Gambar 3.29.



Gambar 3.29 Rancangan Popover Notification

### 3.5.21 Rancangan Popover Pesan

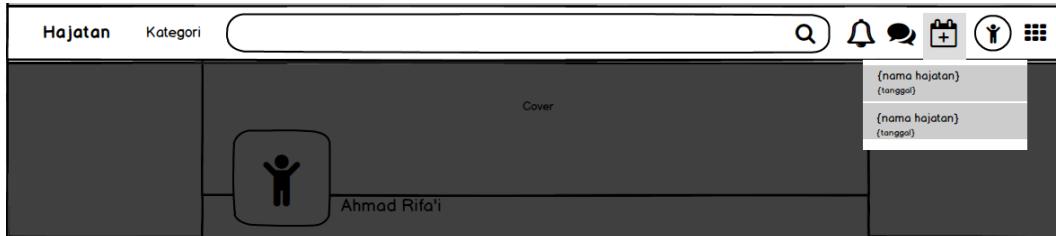
*Popover pesan* merupakan navigasi yang digunakan untuk menampilkan cuplikan pesan. Adapun rancangan *popover pesan* terdapat pada Gambar 3.30.



Gambar 3.30 Rancangan Popover Pesan

### 3.5.22 Rancangan *Popover* Jadwal Hajatan

*Popover* jadwal hajatan adalah navigasi yang digunakan untuk menampilkan *list* hajatan yang belum dimulai. Adapun rancangan *popover* jadwal hajatan terdapat pada Gambar 3.31.



Gambar 3.31 Rancangan *Popover* Jadwal Hajatan

### 3.5.23 Rancangan *Popover* Profil

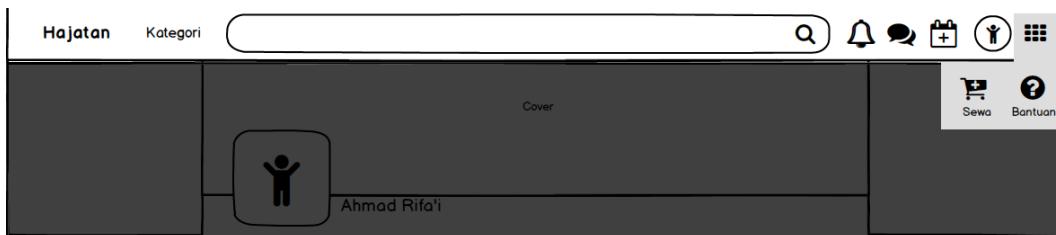
*Popover* profil adalah navigasi yang digunakan untuk memudahkan menuju ke halaman user atau untuk melakukan *logout*. Adapun rancangan *popover* profil terdapat pada Gambar 3.32.



Gambar 3.32 Rancangan *Popover* Profil

### 3.5.24 Rancangan *Popover* Menu

*Popover* menu adalah navigasi yang digunakan untuk memudahkan akses menu-menu tambahan khusus yang terdapat pada aplikasi. Adapun rancangan *popover* menu terdapat pada Gambar 3.33.

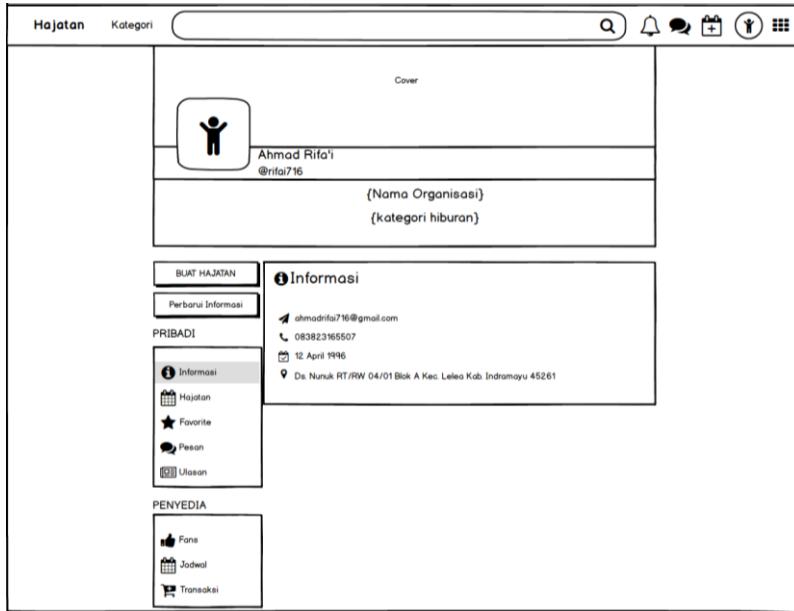


Gambar 3.33 Rancangan *Popover* Menu

### 3.5.25 Rancangan Halaman Penyedia Hiburan

Halaman penyedia hiburan adalah halaman yang digunakan *user* penyedia hiburan untuk mengontrol aktifitas-aktifitas dalam aplikasi. Tampilan ini dapat

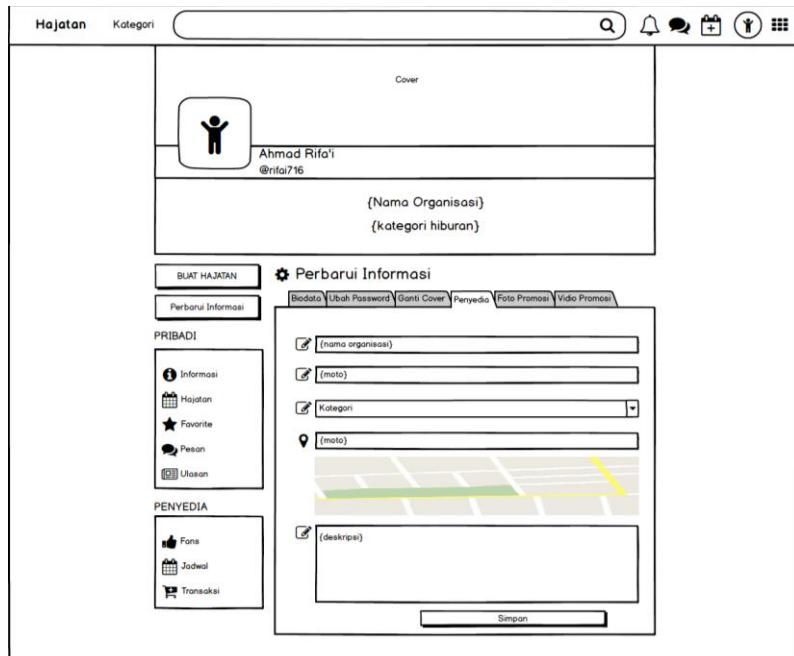
diakses ketika *user* sudah melakukan buka usaha di dalam aplikasi ini. Adapun rancangan halaman penyedia hiburan terdapat pada Gambar 3.34.



Gambar 3.34 Rancangan Halaman Penyedia Hiburan

### 3.5.26 Rancangan Halaman Edit Informasi Penyedia Hiburan

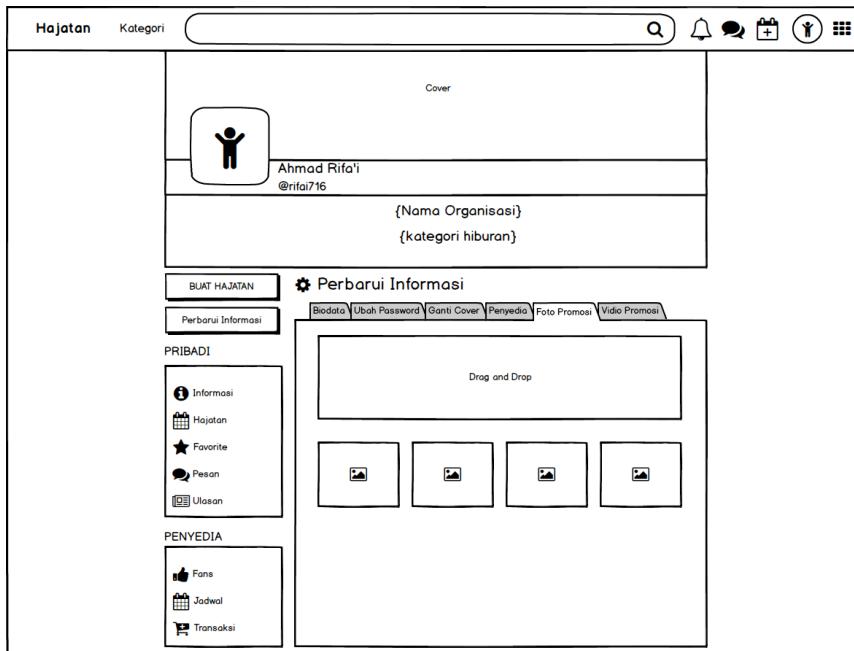
Halaman edit informasi penyedia hiburan adalah halaman yang digunakan untuk *update* informasi tentang hiburan yang dipromosikan, seperti merubah harga, merubah nama organisasi, atau merubah deskripsi. Adapun rancangan halaman edit informasi penyedia hiburan terdapat pada Gambar 3.35.



Gambar 3.35 Rancangan Halaman Edit Informasi Penyedia Hiburan

### 3.5.27 Rancangan Halaman Tambah Foto Promosi

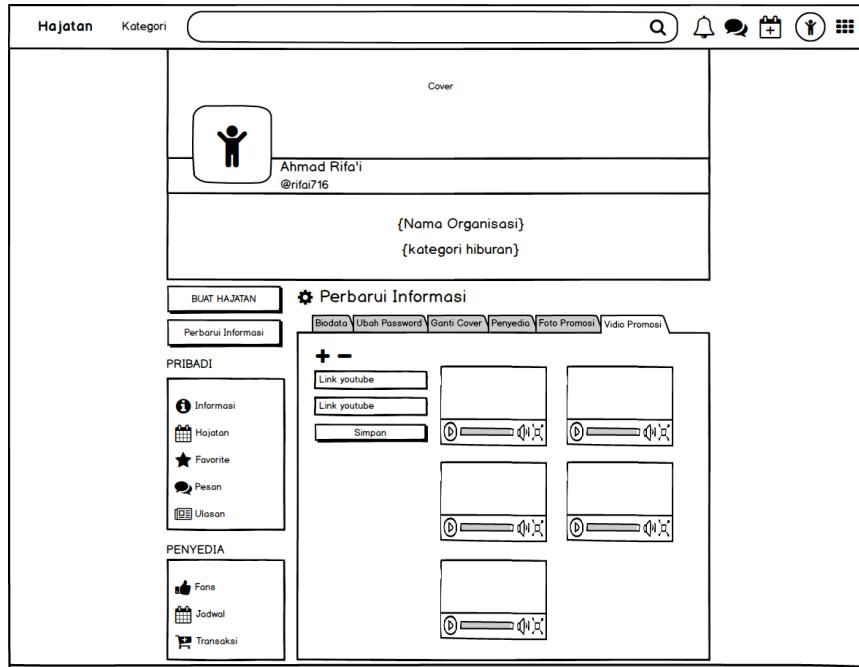
Halaman tambah foto promosi adalah halaman yang digunakan untuk *upload* foto-foto kegiatan ataupun foto-foto pendukung guna menunjukkan fasilitas dan fitur-fitur kelengkapan dari penyedia hiburan. Adapun rancangan halaman tambah foto promosi terdapat pada Gambar 3.36.



Gambar 3.36 Rancangan Halaman Tambah Foto Promosi

### 3.5.28 Rancangan Halaman Tambah Vidio Promosi

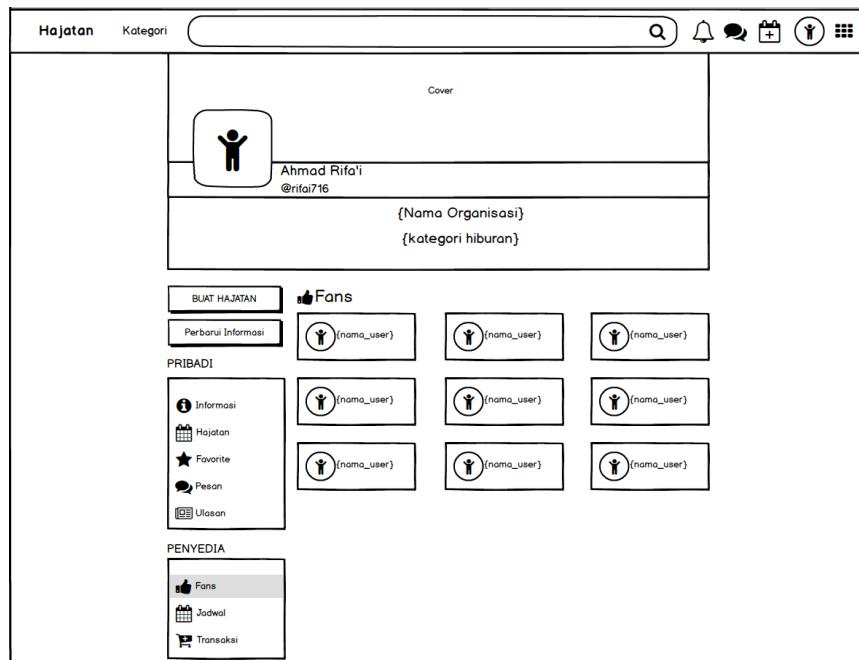
Halaman tambah vidio promosi adalah halaman yang digunakan untuk *upload* vidio-vidio kegiatan ataupun vidio pendukung yang bersumber dari youtube guna menunjukkan fasilitas dan fitur-fitur kelengkapan dari penyedia hiburan. Adapun rancangan halaman tambah vidio promosi terdapat pada Gambar 3.37.



Gambar 3.37 Rancangan Halaman Vidio Promosi

### 3.5.29 Rancangan Halaman Fans

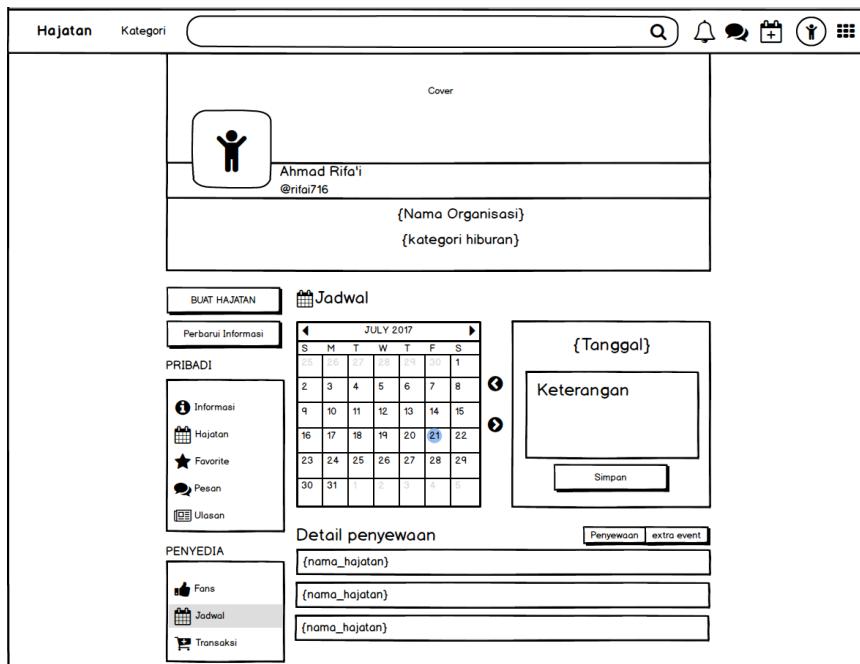
Halaman fans adalah halaman yang digunakan oleh penyedia hiburan untuk melihat siapa saja *user* yang menyukai hiburannya. Adapun rancangan halaman fans terdapat pada Gambar 3.38.



Gambar 3.38 Rancangan Halaman Fans

### 3.5.30 Rancangan Halaman Jadwal

Halaman jadwal adalah halaman yang digunakan oleh penyedia hiburan untuk mengontrol serta menambahkan jadwal kegiatan penyewaan. Adapun rancangan halaman jadwal terdapat pada Gambar 3.39.



Gambar 3.39 Rancangan Halaman Jadwal

### 3.5.31 Rancangan Halaman Transaksi

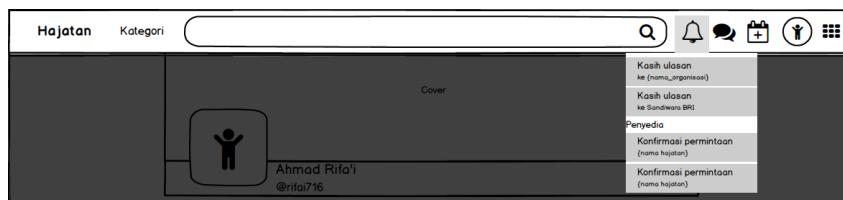
Halaman transaksi adalah halaman yang digunakan oleh penyedia hiburan untuk mengkonfirmasi penyewaan-penyewaan yang ada dan juga dapat memasukan keterangan detail transaksi. Adapun rancangan halaman transaksi terdapat pada Gambar 3.40.



Gambar 3.40 Rancangan Halaman Transaksi

### 3.5.32 Rancangan Popover Notification Penyedia Hiburan

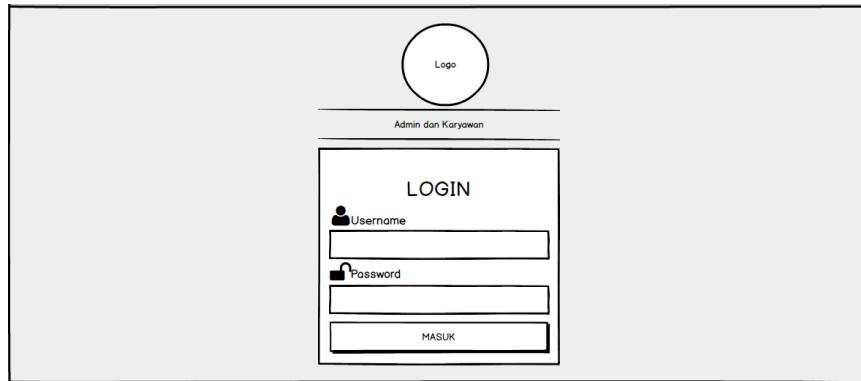
*Popover notification* penyedia hiburan adalah navigasi yang digunakan oleh user penyedia hiburan untuk menampilkan *list* penyewaan. Adapun rancangan *popover notification* terdapat pada Gambar 3.41.



Gambar 3.41 Rancangan Popover Notification Penyedia Hiburan

### 3.5.33 Rancangan Halaman Login Admin dan Pegawai

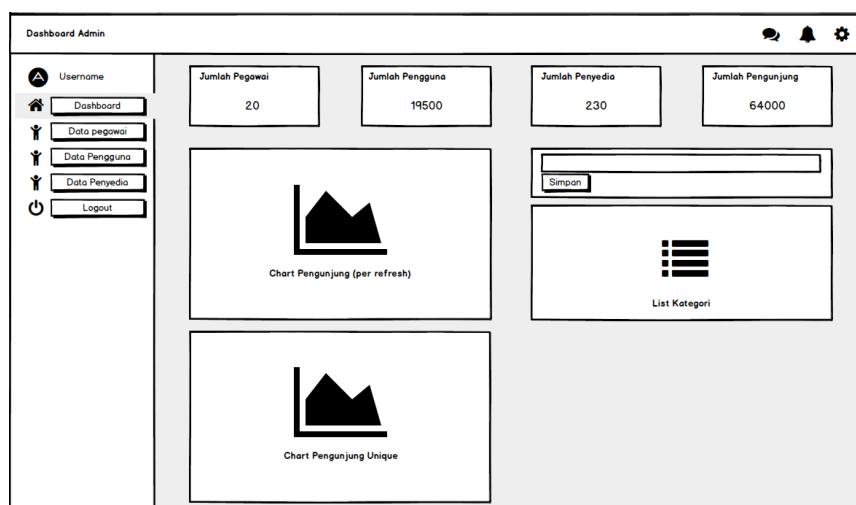
Halaman *login* admin dan pegawai adalah halaman yang berfungsi untuk *login user* dengan level admin dan pegawai dimana admin dan juga pegawai merupakan *user* yang memantau segala kegiatan/aktifitas dalam aplikasi ini. Adapun rancangan halaman *login* admin dan pegawai terdapat pada Gambar 3.42.



Gambar 3.42 Rancangan Halaman Login Admin dan Pegawai

### 3.5.34 Rancangan Halaman *Dashboard* Admin dan Pegawai

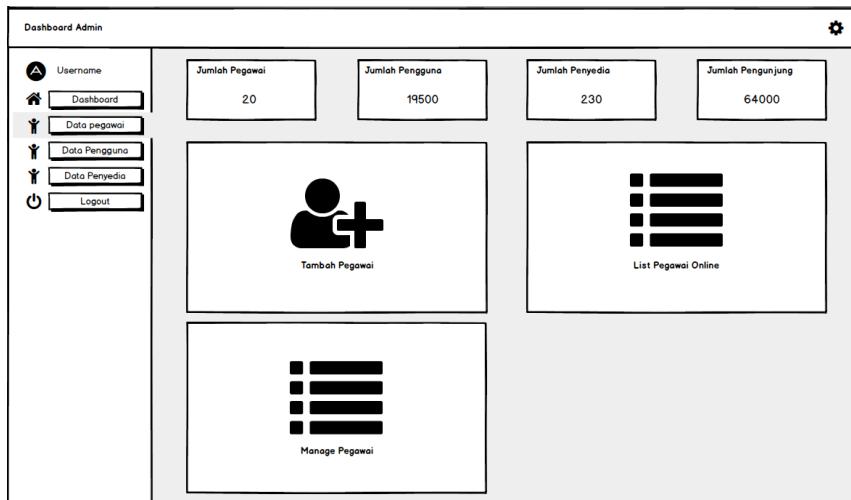
Halaman *dashboard* admin dan pegawai merupakan halaman yang digunakan untuk memantau aktifitas-aktifitas pengunjung *website*. Adapun rancangan halaman *dashboard* admin dan pegawai terdapat pada Gambar 3.43.



Gambar 3.43 Rancangan Halaman *Dashboard* Admin dan Pegawai

### 3.5.35 Rancangan Halaman Data Pegawai

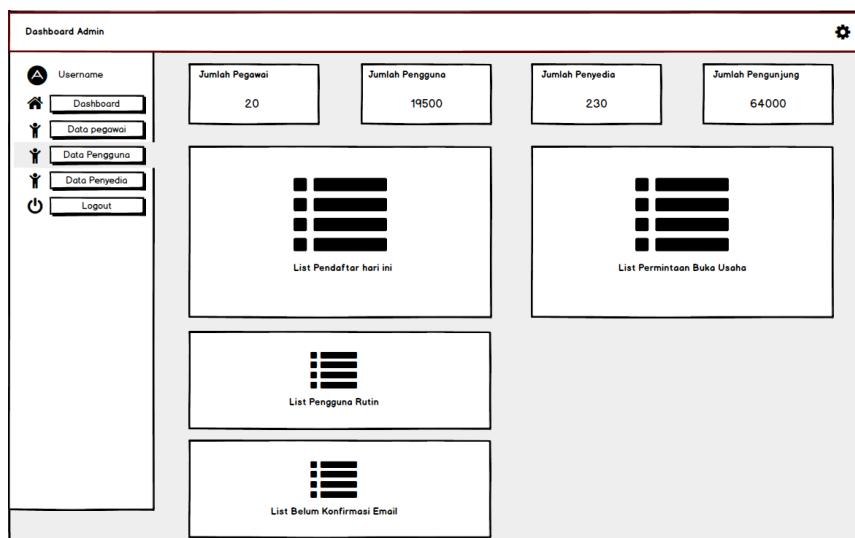
Halaman data pegawai adalah halaman yang digunakan oleh admin untuk menambahkan pegawai dan juga untuk memantau kegiatan-kegiatan pegawai dalam aplikasi ini. Adapun rancangan halaman data pegawai terdapat pada Gambar 3.44.



Gambar 3.44 Rancangan Halaman Data Pegawai

### 3.5.36 Rancangan Halaman Data User

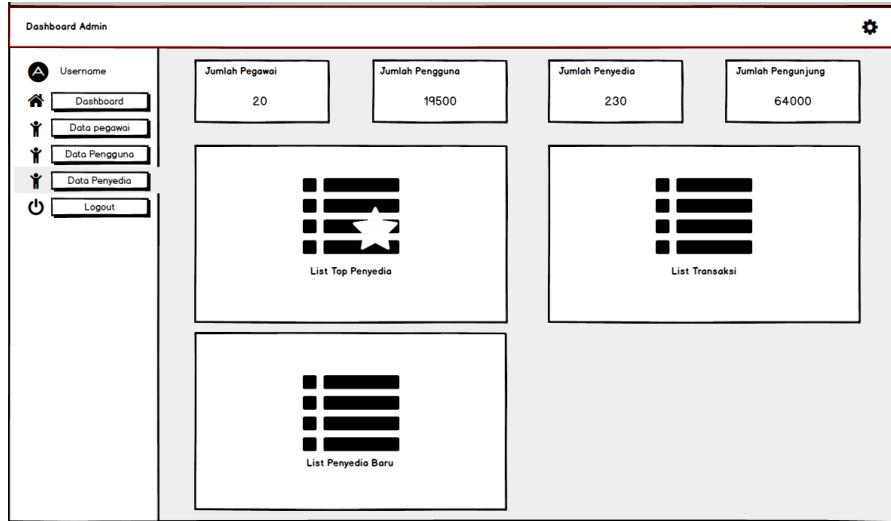
Halaman data *user* adalah halaman yang berfungsi untuk memantau kegiatan dari setiap *user* dalam aplikasi ini. Adapun rancangan halaman data *user* terdapat pada Gambar 3.45.



Gambar 3.45 Rancangan Halaman Data User

### 3.5.37 Rancangan Halaman Data User Penyedia Hiburan

Halaman data *user* penyedia hiburan adalah halaman yang berfungsi untuk memantau kegiatan-kegiatan dari *user* penyedia hiburan. Adapun rancangan halaman data *user* penyedia hiburan terdapat pada Gambar 3.46.



Gambar 3.46 Rancangan Halaman Data *User Penyedia Hiburan*

### 3.6 Normalisasi Database

Berikut ini merupakan hasil normalisasi database pada aplikasi e-market hiburan untuk kebutuhan hajatan.

#### 3.6.1 Tabel *Unnormal*

Tabel penyewaan hiburan untuk suatu hajatan dalam bentuk *unnormal* dapat dilihat pada Tabel 3.3.

Tabel 3.3 Tabel Penyewaan Hiburan untuk Hajatan *Unnormal*

id_user	nama	id_hajatan	nama_hajatan	id_penyedia	nama_organisasi
1	rifai	1	Sunatan	7	Genjringan
		2	Nikahan	8	Sandiwara BRI
		3	Sunatan	7	Genjringan
2	Safitri	4	Rasulan	8	Sandiwara BRI
		5	Nikahan	7	Genjringan
		6	Rasulan	9	S. Entertainment

#### 3.6.2 Tabel Normalisasi Kesatu (1NF)

Proses selanjutnya ada bentuk normalisasi kesatu (1NF) dari tabel unnormal. Berikut adalah tabel penyewaan hiburan untuk suatu hajatan dalam bentuk normalisasi kesatu dapat dilihat pada Tabel 3.4.

Tabel 3.4 Tabel Penyewaan Hiburan untuk Hajatan Normalisasi kesatu (1NF)

id_user	nama	id_hajatan	nama_hajatan	id_penyedia	nama_organisasi
1	rifai	1	Sunatan	7	Genjringan
1	rifai	2	Nikahan	8	Sandiwara BRI
1	rifai	3	Sunatan	7	Genjringan
2	Safitri	4	Rasulan	8	Sandiwara BRI
2	Safitri	5	Nikahan	7	Genjringan
2	Safitri	6	Rasulan	9	S. Entertainment

### 3.6.3 Tabel Normalisasi Kedua (2NF)

Selanjutnya pemisahan tabel atau pembuatan tabel baru. Berikut adalah tabel penyewaan hiburan untuk suatu hajatan dalam bentuk normalisasi kedua (2NF) dapat dilihat pada Tabel 3.5 dan Tabel 3.6.

Tabel 3.5 Tabel *User*

id_user	nama
1	rifai
2	Safitri

Tabel 3.6 Tabel Penyewaan

id_user	id_hajatan	nama_hajatan	id_penyedia	nama_organisasi
1	1	Sunatan	7	Genjringan
1	2	Nikahan	8	Sandiwara BRI
1	3	Sunatan	7	Genjringan
2	4	Rasulan	8	Sandiwara BRI
2	5	Nikahan	7	Genjringan
2	6	Rasulan	9	S. Entertainment

### 3.6.4 Tabel Normalisasi Ketiga (3NF)

Proses selanjutnya menghilangkan semua ketergantungan yang transitif terhadap key-nya. Berikut adalah tabel penyewaan hiburan untuk suatu hajatan dalam bentuk normalisasi ketiga (3NF) dapat dilihat pada Tabel 3.7, Tabel 3.8, Tabel 3.9, dan Tabel 3.10.

Tabel 3.7 Tabel *User*

id_user	nama
1	rifai
2	Safitri

Tabel 3.8 Tabel Hajatan

id_hajatan	nama_hajatan	id_user
1	Sunatan	1
2	Nikahan	1
3	Sunatan	1
4	Rasulan	2
5	Nikahan	2
6	Rasulan	2

Tabel 3.9 Tabel Penyedia

id_penyedia	nama_organisasi
7	Genjringan
8	Sandiwara BRI
7	Genjringan
8	Sandiwara BRI
7	Genjringan
9	S. Entertainment

Tabel 3.10 Tabel Transaksi

id_transaksi	id_hajatan	id_penyedia
1	1	7
2	2	8
3	3	7
4	4	8
5	5	7
6	6	9

### 3.7 Perancangan Database

Berikut ini merupakan perancangan dari tabel-tabel dalam *database* pembuatan aplikasi e-market hiburan untuk kebutuhan hajatan.

#### 3.7.1 Tabel detail\_chatting

Tabel ini digunakan untuk menyimpan teks percakapan dan juga tanggal dari masing-masing teks percakapan tersebut. Daftar kolom-kolom tabel detail\_chatting dapat dilihat pada Tabel 3.11.

Tabel 3.11 Tabel detail\_chatting

No.	Nama Kolom	Tipe Data	Keterangan
1	chatting	int(10)	Menyimpan id_chatting (FK)
2	pesan	text	Menyimpan teks pesan
3	waktu	timestamp	Menyimpan waktu

### 3.7.2 Tabel detail\_transaksi

Tabel ini digunakan untuk menyimpan catatan-catatan transaksi dari tiap-tiap transaksi penyewaan yang dilakukan dalam sistem. Daftar kolom-kolom tabel detail\_transaksi dapat dilihat pada Tabel 3.12.

Tabel 3.12 Tabel detail\_transaksi

No.	Nama Kolom	Tipe Data	Keterangan
1	transaksi	int(10)	Menyimpan id_transaksi (FK)
2	keterangan	varchar(150)	Menyimpan keterangan
3	nominal	varchar(12)	Menyimpan nominal uang
4	tanggal	date	Menyimpan tanggal

### 3.7.3 Tabel districts

Tabel ini digunakan untuk menyimpan data-data kecamatan yang ada di negara Indonesia. Daftar kolom-kolom tabel districts dapat dilihat pada Tabel 3.13.

Tabel 3.13 Tabel districts

No.	Nama Kolom	Tipe Data	Keterangan
1	id	char(7)	Menyimpan id (PK)
2	regency_id	char(4)	Menyimpan regency_id (FK)
3	name	varchar(255)	Menyimpan keterangan districts

### 3.7.4 Tabel extra\_event

Tabel ini digunakan untuk menyimpan data-data penyewaan atau suatu kegiatan diluar sistem yang dimasukan oleh *user* penyedia hiburan untuk memberitahukan kepada *user* yang akan menyewa hiburannya. Daftar kolom-kolom tabel extra\_event dapat dilihat pada Tabel 3.14.

Tabel 3.14 Tabel extra\_event

No.	Nama Kolom	Tipe Data	Keterangan
1	penyedia	int(10)	Menyimpan id_penyedia (FK)
2	tanggal	date	Menyimpan tanggal acara
3	deskripsi	text	Keterangan acara
4	modified	timestamp	Menyimpan waktu

### **3.7.5 Tabel foto\_promosi**

Tabel ini digunakan untuk menyimpan foto-foto promosi yang dimasukan oleh *user* penyedia hiburan untuk menunjukan kegiatan-kegiatannya ataupun hal-hal yang dapat meyakinkan dan tertarik dengan hiburannya. Daftar kolom-kolom tabel foto\_promosi dapat dilihat pada Tabel 3.15.

Tabel 3.15 Tabel foto\_promosi

No.	Nama Kolom	Tipe Data	Keterangan
1	penyedia	int(10)	Menyimpan id_penyedia (FK)
2	url_foto	text	Menyimpan url foto
3	waktu	timestamp	Menyimpan waktu

### **3.7.6 Tabel log\_admin**

Tabel ini digunakan untuk menyimpan catatan-catatan *login* dan *logout* dari admin atau pegawai. Daftar kolom-kolom tabel log\_admin dapat dilihat pada Tabel 3.16.

Tabel 3.16 Tabel log\_admin

No.	Nama Kolom	Tipe Data	Keterangan
1	admin	int(2)	Menyimpan id_admin (FK)
2	waktu	timestamp	Menyimpan waktu
3	status	int(1)	Menyimpan id_status (FK)

### **3.7.7 Tabel log\_user**

Tabel ini digunakan untuk menyimpan catatan-catatan *login* dan *logout* dari user. Daftar kolom-kolom tabel log\_user dapat dilihat pada Tabel 3.17.

Tabel 3.17 Tabel log\_user

No.	Nama Kolom	Tipe Data	Keterangan
1	user	int(10)	Menyimpan id_user (FK)
2	waktu	timestamp	Menyimpan waktu
3	status	int(1)	Menyimpan id_status (FK)

### **3.7.8 Tabel provinces**

Tabel ini digunakan untuk menyimpan data-data provinsi yang ada di negara Indonesia. Daftar kolom-kolom tabel provinces dapat dilihat pada Tabel 3.18.

Tabel 3.18 Tabel provinces

No.	Nama Kolom	Tipe Data	Keterangan
1	id	char(2)	Menyimpan id (PK)
2	name	varchar(255)	Menyimpan nama provinsi

### 3.7.9 Tabel regencies

Tabel ini digunakan untuk menyimpan data-data kabupaten yang di negara Indonesia. Daftar kolom-kolom tabel regencies dapat dilihat pada Tabel 3.19.

Tabel 3.19 Tabel regencies

No.	Nama Kolom	Tipe Data	Keterangan
1	id	char(4)	Menyimpan id (PK)
2	province_id	char(2)	Menyimpan id provinces (FK)
3	name	varchar(255)	Menyimpan nama kabupaten

### 3.7.10 Tabel status\_admin

Tabel ini digunakan untuk menyimpan data-data status admin yaitu admin atau pegawai. Daftar kolom-kolom tabel status\_admin dapat dilihat pada Tabel 3.20.

Tabel 3.20 Tabel status\_admin

No.	Nama Kolom	Tipe Data	Keterangan
1	Id_status	int(1)	Menyimpan id_staus (PK)
2	nama	varchar(20)	Menyimpan nama status admin

### 3.7.11 Tabel status\_log

Tabel ini digunakan untuk menyimpan data-data status log yaitu *login* atau *logout*. Daftar kolom-kolom tabel status\_log dapat dilihat pada Tabel 3.21.

Tabel 3.21 Tabel status\_log

No.	Nama Kolom	Tipe Data	Keterangan
1	id_status	int(1)	Menyimpan id_staus (PK)
2	nama	varchar(10)	Menyimpan nama status log

### 3.7.12 Tabel status\_user

Tabel ini digunakan untuk menyimpan data-data status *user* yaitu konsumen, penyedia, dan *pending*, dimana *pending* merupakan status yang diberikan kepada *user* yang sedang menunggu perubahan status dari konsumen

menjadi penyedia. Daftar kolom-kolom tabel status\_user dapat dilihat pada Tabel 3.22.

Tabel 3.22 Tabel status\_user

No.	Nama Kolom	Tipe Data	Keterangan
1	id_status	int(1)	Menyimpan id_status (PK)
2	nama	varchar(10)	Menyimpan nama status user

### 3.7.13 Tabel tb\_admin

Tabel ini digunakan untuk menyimpan data-data admin seperti nama, *username*, dan *password*. Daftar kolom-kolom tabel tb\_admin dapat dilihat pada Tabel 3.23.

Tabel 3.23 Tabel tb\_admin

No.	Nama Kolom	Tipe Data	Keterangan
1	id_admin	int(2)	Menyimpan id_admin (PK)
2	nama	varchar(50)	Menyimpan nama admin
3	username	varchar(16)	Menyimpan username admin
4	password	varchar(16)	Menyimpan password admin
5	status	int(1)	Menyimpan id_status admin (FK)

### 3.7.14 Tabel tb\_chatting

Tabel ini digunakan untuk menyimpan identitas suatu percakapan dari siapa dan ditujukan kepada siapa. Daftar kolom-kolom tabel tb\_chatting dapat dilihat pada Tabel 3.24.

Tabel 3.24 Tabel tb\_chatting

No.	Nama Kolom	Tipe Data	Keterangan
1	id_chatting	int(10)	Menyimpan id_chatting (PK)
2	from_id	int(10)	Menyimpan id_user (FK)
3	to_id	int(10)	Menyimpan id_user (FK)
4	waktu	timestamp	Menyimpan waktu

### 3.7.15 Tabel tb\_favorite

Tabel ini digunakan untuk menyimpan data-data favorit dari identitas *user* ke identitas *user*. Daftar kolom-kolom tabel tb\_favorite dapat dilihat pada Tabel 3.25.

Tabel 3.25 Tabel tb\_favorite

No.	Nama Kolom	Tipe Data	Keterangan
1	from_id	int(10)	Menyimpan id_user (FK)
2	to_id	int(10)	Menyimpan id_penyedia (FK)

### 3.7.16 Tabel tb\_hajatan

Tabel ini digunakan untuk menyimpan data-data hajatan seperti nama hajatan, tanggal hajatan, lokasi hajatan, dan indetitas *user* yang membuat hajatan tersebut. Daftar kolom-kolom tabel tb\_hajatan dapat dilihat pada Tabel 3.26.

Tabel 3.26 Tabel tb\_hajatan

No.	Nama Kolom	Tipe Data	Keterangan
1	id_hajatan	int(10)	Menyimpan id_hajatan (PK)
2	user	int(10)	Menyimpan id_user (FK)
3	nama_hajatan	varchar(100)	Menyimpan nama hajatan
4	deskripsi	text	Menyimpan ketarangan hajatan
5	location	varchar(100)	Menyimpan lokasi hajatan
6	tanggal_mulai	date	Menyimpan tanggal
7	tanggal_akhir	date	Menyimpan tanggal
8	created	datetime	Menyimpan waktu pembuatan
9	modified	timestamp	Menyimpan waktu perubahan

### 3.7.17 Tabel tb\_harga

Tabel ini digunakan untuk menyimpan nominal uang penyewaan dari suatu hiburan. Daftar kolom-kolom tabel tb\_harga dapat dilihat pada Tabel 3.27.

Tabel 3.27 Tabel tb\_harga

No.	Nama Kolom	Tipe Data	Keterangan
1	penyedia	int(10)	Menyimpan id_penyedia (FK)
2	harga	varchar(12)	Menyimpan nominal harga
3	modified	timestamp	Menyimpan waktu perubahan

### 3.7.18 Tabel tb\_kategori

Tabel ini digunakan untuk menyimpan data-data kategori hiburan. Daftar kolom-kolom tabel tb\_kategori dapat dilihat pada Tabel 3.28.

Tabel 3.28 Tabel tb\_kategori

No.	Nama Kolom	Tipe Data	Keterangan
1	id_kategori	int(2)	Menyimpan id (PK)
2	nama	varchar(30)	Menyimpan nama kategori
3	status	enum('enable','disable')	Menyimpan status kategori

### 3.7.19 Tabel tb\_penyedia

Tabel ini digunakan untuk menyimpan data-data *user* penyedia hiburan seperti nama hiburan/nama organisasi, moto, identitas kategori, dan identitas *user* dari penyedia hiburan tersebut. Daftar kolom-kolom tabel tb\_penyedia dapat dilihat pada Tabel 3.29.

Tabel 3.29 Tabel tb\_penyedia

No.	Nama Kolom	Tipe Data	Keterangan
1	id_penyedia	int(10)	Menyimpan id_penyedia (PK)
2	user	int(10)	Menyimpan id_user (FK)
3	nama_organisasi	varchar(50)	Menyimpan nama organisasi
4	deskripsi	text	Menyimpan deskripsi hiburan
5	moto	varchar(30)	Menyimpan moto hiburan
6	created	datetime	Menyimpan tanggal pembuatan
7	modified	modified	Menyimpan tanggal perubahan
8	kategori	int(2)	Menyimpan id_kategori (FK)

### 3.7.20 Tabel tb\_transaksi

Tabel ini digunakan untuk menyimpan data-data transaksi yang dilakukan antara *user* yang telah membuat hajatan dan menyewa hiburan. Daftar kolom-kolom tabel tb\_transaksi dapat dilihat pada Tabel 3.30.

Tabel 3.30 Tabel tb\_transaksi

No.	Nama Kolom	Tipe Data	Keterangan
1	id_transaksi	int(10)	Menyimpan id_transaksi (PK)
2	hajatan	int(10)	Menyimpan id_hajatan (FK)
3	penyedia	int(10)	Menyimpan id_penyedia (FK)
4	status	int(1)	Menyimpan status
5	waktu	timestamp	Menyimpan waktu

### 3.7.21 Tabel tb\_ulasan

Tabel ini digunakan untuk menyimpan ulasan dari suatu transaksi seperti *ratting* dan komentar untuk pelayanan yang telah diberikan oleh penyedia hiburan. Daftar kolom-kolom tabel tb\_ulasan dapat dilihat pada Tabel 3.31.

Tabel 3.31 Tabel tb\_ulasan

No.	Nama Kolom	Tipe Data	Keterangan
1	transaksi	int(10)	Menyimpan id_transaksi (FK)
2	ratting	int(1)	Menyimpan value ratting
3	komentar	varchar(250)	Menyimpan komentar
4	waktu	timestamp	Menyimpan waktu

### 3.7.22 Tabel tb\_user

Tabel ini digunakan untuk menyimpan data-data dari *user* seperti nama, *username*, *password*, tanggal lahir, alamat, jenis kelamin, dan lain sebagainya. Daftar kolom-kolom tabel tb\_user dapat dilihat pada Tabel 3.32.

Tabel 3.32 Tabel tb\_user

No.	Nama Kolom	Tipe Data	Keterangan
1	id_user	int(10)	Menyimpan id_user (PK)
2	provider	enum('original', 'google', 'facebook')	Menyimpan keterangan cara daftar dan login user
3	uid_social	varchar(255)	Menyimpan id unik dari provider
4	nama	varchar(50)	Menyimpan nama user
5	username	varchar(16)	Menyimpan username
6	email	varchar(50)	Menyimpan email
7	password	varchar(128)	Menyimpan password
8	tgl_lahir	date	Menyimpan tanggal lahir
9	alamat	varchar(200)	Menyimpan alamat
10	location	varchar(100)	Menyimpan kordinat lokasi
11	no_hp	varchar(15)	Menyimpan nomer handphone
12	jenkel	enum('L', 'P')	Menyimpan jenis kelamin
13	url_foto	varchar(200)	Menyimpan url foto
14	cover	varchar(200)	Menyimpan url cover
15	created	datetime	Menyimpan waktu pembuatan
16	modified	timestamp	Menyimpan waktu perubahan
17	hash	varchar(32)	Menyimpan code unik
18	aktif	int(1)	Menyimpan status akun
19	status	int(1)	Menyimpan id_status (FK)

### 3.7.23 Tabel tb\_visitor

Tabel ini digunakan untuk menyimpan nilai hasil *counting* banyaknya pengunjung *website* pada tanggal itu. Daftar kolom-kolom tabel tb\_visitor dapat dilihat pada Tabel 3.33.

Tabel 3.33 Tabel tb\_visitor

No.	Nama Kolom	Tipe Data	Keterangan
1	total_hits	int(12)	Menyimpan value hasil <i>counting</i>
2	unique_hits	int(12)	Menyimpan hasil <i>counting</i> pengunjung per session
3	tanggal	date	Menyimpan tanggal

### **3.7.24 Tabel `vidio_promosi`**

Tabel ini digunakan oleh *user* penyedia hiburan untuk menyimpan *link* dari youtube guna menarik minat dari *user* konsumen yang akan melakukan penyewaan hiburan untuk kebutuhan hanjatannya. Daftar kolom-kolom tabel `vidio_promosi` dapat dilihat pada Tabel 3.34.

Tabel 3.34 Tabel `vidio_promosi`

No	Nama Kolom	Tipe Data	Keterangan
1	penyedia	int(10)	Menyimpan id_penyedia (FK)
2	url_youtube	text	Menyimpan url youtube
3	waktu	timestamp	Menyimpan waktu

### **3.7.25 Tabel `villages`**

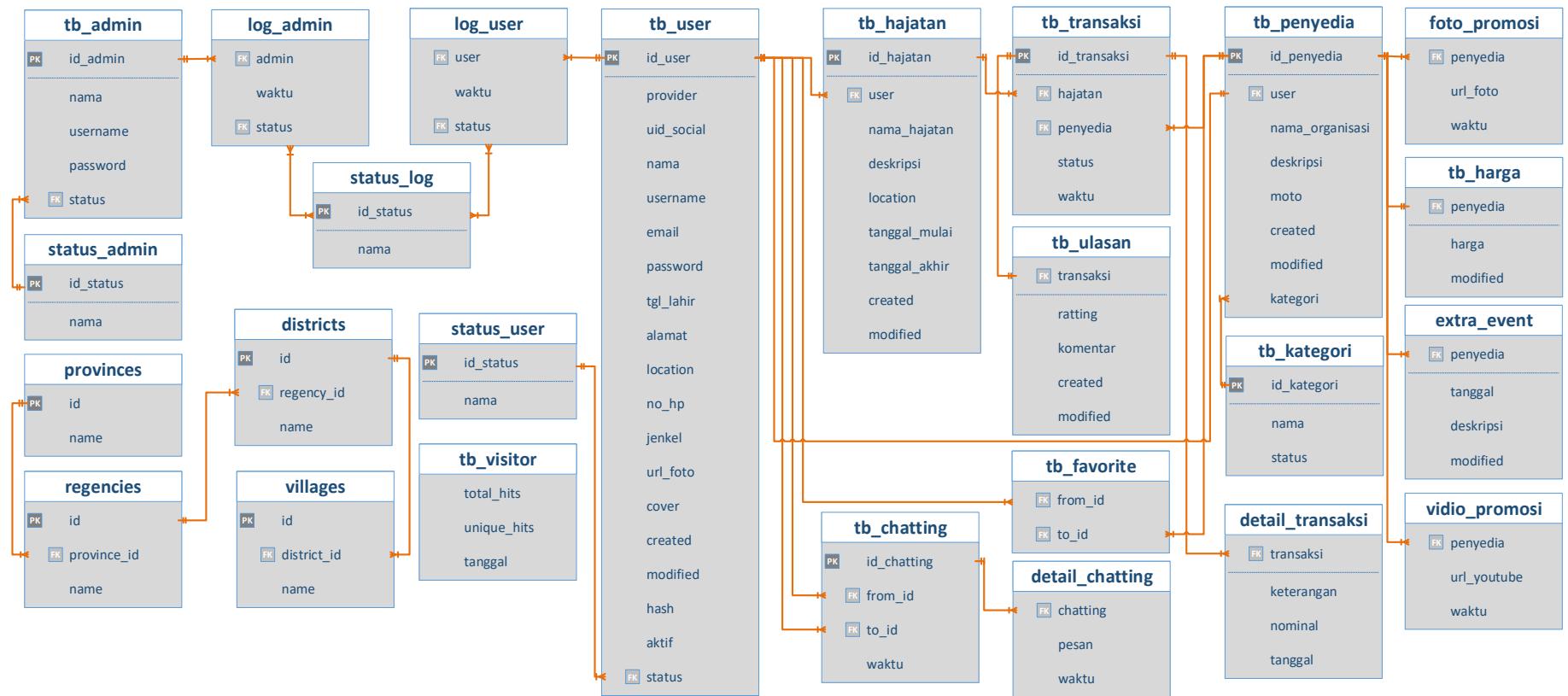
Tabel ini digunakan untuk menyimpan data-data desa yang ada di negara Indonesia. Daftar kolom-kolom tabel `villages` dapat dilihat pada Tabel 3.35.

Tabel 3.35 Tabel `villages`

No	Nama Kolom	Tipe Data	Keterangan
1	id	char(10)	Menyimpan id (PK)
2	district_id	char(7)	Menyimpan id districts (FK)
3	name	varchar(255)	Menyimpan nama desa

### **3.7.26 Relasi Database**

Adapun relasi dari tabel-tabel aplikasi e-market hiburan untuk kebutuhan hajatan dapat dilihat pada Gambar 3.47.



Gambar 3.47 Relasi Database

## **BAB IV**

### **HASIL DAN PEMBAHASAN**

#### **4.1 Hasil**

Penelitian ini menghasilkan sebuah aplikasi e-market hiburan untuk kebutuhan hajatan. Aplikasi ini dapat digunakan untuk mempromosikan hiburan bagi pihak-pihak yang mempunyai hiburan. Aplikasi ini juga dapat digunakan untuk menginformasikan tentang suatu acara hajatan sehingga setiap orang yang masih kerabat dekat pembuat hajatan dapat mengetahui ada hajatan di mana lokasi hajatannya serta dapat menginformasikan hiburan apa saja yang ada pada hajatan tersebut dengan catatan penyewaan hiburan dilakukan dalam aplikasi ini.

Penulis dapat menyimpulkan aplikasi e-market hiburan untuk kebutuhan hajatan yang telah dibuat adalah aplikasi yang digunakan untuk media informasi, komunikasi, dan transaksi antara pihak yang akan hajatan dengan pihak penyedia hiburan untuk kebutuhan hajatan.

#### **4.2 Pembahasan**

Setelah mengumpulkan, menganalisis, dan merancang maka penulis mengimplementasikan semuanya untuk terciptanya aplikasi e-market hiburan untuk kebutuhan hajatan.

##### **4.1.1 Implementasi *Database***

*Database* aplikasi e-market hiburan untuk kebutuhan hajatan terdiri dari 25 tabel yaitu tabel detail\_chatting, tabel detail\_transaksi, tabel districts, tabel extra\_event, tabel foto\_promosi, tabel log\_admin, tabel log\_user, tabel provinces, tabel regencies, tabel status\_admin, tabel status\_log, tabel status\_user, tabel tb\_admin, tabel tb\_chatting, tabel tb\_favorite, tabel tb\_hajatan, tabel tb\_harga, tabel tb\_kategori, tabel tb\_penyedia, tabel tb\_transaksi, tabel tb\_ulasan, tabel tb\_user, tabel tb\_visitor, tabel vido\_promosi, tabel villages. Berikut implementasi pembuatan *database* untuk aplikasi e-market hiburan untuk kebutuhan hajatan.

## **1. Database u191296018\_ta**

*Database u191296018\_ta* adalah nama *database* yang dibuat di *server hosting* yang penulis gunakan. *Database u191296018\_ta* merupakan sebuah tempat untuk menyimpan tabel-tabel *database* yang telah dibuat.

## **2. Tabel detail\_chatting**

Tabel ini digunakan untuk menyimpan teks percakapan dan juga tanggal dari masing-masing teks percakapan tersebut. Berikut *query* untuk membuatnya.

```
CREATE TABLE IF NOT EXISTS `detail_chatting` (
  `chatting` int(10) NOT NULL,
  `pesan` text NOT NULL,
  `waktu` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

## **3. Tabel detail\_transaksi**

Tabel ini digunakan untuk menyimpan catatan-catatan transaksi dari tiap-tiap transaksi penyewaan yang dilakukan dalam sistem. Berikut *query* untuk membuatnya.

```
CREATE TABLE IF NOT EXISTS `detail_transaksi` (
  `transaksi` int(10) NOT NULL,
  `keterangan` varchar(150) NOT NULL,
  `nominal` varchar(12) NOT NULL,
  `tanggal` date NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

## **4. Tabel districts**

Tabel ini digunakan untuk menyimpan data-data kecamatan yang ada di negara Indonesia. Berikut *query* untuk membuatnya.

```
CREATE TABLE IF NOT EXISTS `districts` (
  `id` char(7) COLLATE utf8_unicode_ci NOT NULL,
  `regency_id` char(4) COLLATE utf8_unicode_ci NOT NULL,
  `name` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8
COLLATE=utf8_unicode_ci;
```

## 5. Tabel extra\_event

Tabel ini digunakan untuk menyimpan data-data penyewaan atau suatu kegiatan diluar sistem yang dimasukan oleh *user* penyedia hiburan untuk memberitahukan kepada *user* yang akan menyewa hiburannya. Berikut *query* untuk membuatnya.

```
CREATE TABLE IF NOT EXISTS `extra_event` (
    `penyedia` int(10) NOT NULL,
    `tanggal` date NOT NULL,
    `deskripsi` text NOT NULL COMMENT
'mendeskripsikan keterangan event diluar sistem',
    `modified` timestamp NOT NULL DEFAULT
CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

## 6. Tabel foto\_promosi

Tabel ini digunakan untuk menyimpan foto-foto promosi yang dimasukan oleh *user* penyedia hiburan untuk menunjukan kegiatan-kegiatannya ataupun hal-hal yang dapat meyakinkan dan tertarik dengan hiburannya. Berikut *query* untuk membuatnya.

```
CREATE TABLE IF NOT EXISTS `foto_promosi` (
    `penyedia` int(10) NOT NULL,
    `url_foto` text NOT NULL,
    `waktu` timestamp NOT NULL DEFAULT
CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

## 7. Tabel log\_admin

Tabel ini digunakan untuk menyimpan catatan-catatan *login* dan *logout* dari admin atau pegawai. Berikut *query* untuk membuatnya.

```
CREATE TABLE IF NOT EXISTS `log_admin` (
    `admin` int(2) NOT NULL,
    `waktu` timestamp NOT NULL DEFAULT
CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
    `status` int(1) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

## 8. Tabel log\_user

Tabel ini digunakan untuk menyimpan catatan-catatan *login* dan *logout* dari user. Berikut *query* untuk membuatnya.

```

CREATE TABLE IF NOT EXISTS `log_user` (
    `user` int(10) NOT NULL,
    `waktu` timestamp     NOT NULL      DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
    `status` int(1) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

```

## 9. Tabel provinces

Tabel ini digunakan untuk menyimpan data-data provinsi yang ada di negara Indonesia. Berikut *query* untuk membuatnya.

```

CREATE TABLE IF NOT EXISTS `provinces` (
    `id` char(2) COLLATE utf8_unicode_ci NOT NULL,
    `name` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
    PRIMARY KEY (`id`)
) ENGINE=MyISAM      DEFAULT      CHARSET=utf8
COLLATE=utf8_unicode_ci;

```

## 10. Tabel regencies

Tabel ini digunakan untuk menyimpan data-data kabupaten yang di negara Indonesia. Berikut *query* untuk membuatnya.

```

CREATE TABLE IF NOT EXISTS `regencies` (
    `id` char(4) COLLATE utf8_unicode_ci NOT NULL,
    `province_id` char(2) COLLATE utf8_unicode_ci NOT NULL,
    `name` varchar(255) COLLATE utf8_unicode_ci NOT NULL,
    PRIMARY KEY (`id`)
) ENGINE=MyISAM      DEFAULT      CHARSET=utf8
COLLATE=utf8_unicode_ci;

```

## 11. Tabel status\_admin

Tabel ini digunakan untuk menyimpan data-data status admin yaitu admin atau pegawai. Berikut *query* untuk membuatnya.

```

CREATE TABLE IF NOT EXISTS `status_admin` (
    `id_status` int(1) NOT NULL AUTO_INCREMENT,
    `nama` varchar(20) NOT NULL,
    PRIMARY KEY (`id_status`)
) ENGINE=MyISAM      DEFAULT      CHARSET=latin1
AUTO_INCREMENT=1 ;

```

## 12. Tabel status\_log

Tabel ini digunakan untuk menyimpan data-data status log yaitu *login* atau *logout*. Berikut *query* untuk membuatnya.

```
CREATE TABLE IF NOT EXISTS `status_log` (
  `id_status` int(1) NOT NULL AUTO_INCREMENT,
  `nama` varchar(10) NOT NULL,
  PRIMARY KEY (`id_status`)
) ENGINE=MyISAM           DEFAULT      CHARSET=latin1
AUTO_INCREMENT=1 ;
```

## 13. Tabel status\_user

Tabel ini digunakan untuk menyimpan data-data status *user* yaitu konsumen, penyedia, dan *pending*, dimana *pending* merupakan status yang diberikan kepada *user* yang sedang menunggu perubahan status dari konsumen menjadi penyedia. Berikut *query* untuk membuatnya.

```
CREATE TABLE IF NOT EXISTS `status_user` (
  `id_status` int(1) NOT NULL AUTO_INCREMENT,
  `nama` varchar(10) NOT NULL,
  PRIMARY KEY (`id_status`)
) ENGINE=MyISAM           DEFAULT      CHARSET=latin1
AUTO_INCREMENT=1 ;
```

## 14. Tabel tb\_admin

Tabel ini digunakan untuk menyimpan data-data admin seperti nama, *username*, dan *password*. Berikut *query* untuk membuatnya.

```
CREATE TABLE IF NOT EXISTS `tb_admin` (
  `id_admin` int(2) NOT NULL AUTO_INCREMENT,
  `nama` varchar(50) NOT NULL,
  `username` varchar(16) NOT NULL,
  `password` varchar(16) NOT NULL,
  `status` int(1) NOT NULL,
  PRIMARY KEY (`id_admin`)
) ENGINE=MyISAM           DEFAULT      CHARSET=latin1
AUTO_INCREMENT=1 ;
```

## 15. Tabel tb\_chatting

Tabel ini digunakan untuk menyimpan identitas suatu percakapan dari siapa dan ditujukan kepada siapa. Berikut *query* untuk membuatnya.

```

CREATE TABLE IF NOT EXISTS `tb_chatting` (
  `id_chatting` int(10) NOT NULL AUTO_INCREMENT,
  `from_id` int(10) NOT NULL,
  `to_id` int(10) NOT NULL,
  `waktu` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_chatting`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
AUTO_INCREMENT=1 ;

```

## 16. Tabel tb\_favorite

Tabel ini digunakan untuk menyimpan data-data favorit dari identitas *user* ke identitas *user*. Berikut *query* untuk membuatnya.

```

CREATE TABLE IF NOT EXISTS `tb_favorite` (
  `from_id` int(10) NOT NULL,
  `to_id` int(10) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

```

## 17. Tabel tb\_hajatan

Tabel ini digunakan untuk menyimpan data-data hajatan seperti nama hajatan, tanggal hajatan, lokasi hajatan, dan indentitas *user* yang membuat hajatan tersebut. Berikut *query* untuk membuatnya.

```

CREATE TABLE IF NOT EXISTS `tb_hajatan` (
  `id_hajatan` int(10) NOT NULL AUTO_INCREMENT,
  `user` int(10) NOT NULL,
  `nama_hajatan` varchar(100) NOT NULL,
  `deskripsi` text NOT NULL,
  `location` varchar(100) NOT NULL COMMENT 'lat,lng',
  `tanggal_mulai` date NOT NULL,
  `tanggal_akhir` date NOT NULL,
  `created` datetime NOT NULL,
  `modified` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_hajatan`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
AUTO_INCREMENT=1 ;

```

## 18. Tabel tb\_harga

Tabel ini digunakan untuk menyimpan nominal uang penyewaan dari suatu hiburan. Berikut *query* untuk membuatnya.

```

CREATE TABLE IF NOT EXISTS `tb_harga` (
  `penyedia` int(10) NOT NULL,
  `harga` varchar(12) NOT NULL,
  `modified` timestamp     NOT NULL      DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

```

## 19. Tabel tb\_kategori

Tabel ini digunakan untuk menyimpan data-data kategori hiburan. Berikut *query* untuk membuatnya.

```

CREATE TABLE IF NOT EXISTS `tb_kategori` (
  `id_kategori` int(2) NOT NULL AUTO_INCREMENT,
  `nama` varchar(30) NOT NULL,
  `status` enum('enable','disable') NOT NULL,
  PRIMARY KEY (`id_kategori`)
) ENGINE=MyISAM           DEFAULT      CHARSET=latin1
AUTO_INCREMENT=1 ;

```

## 20. Tabel tb\_penyedia

Tabel ini digunakan untuk menyimpan data-data *user* penyedia hiburan seperti nama hiburan/nama organisasi, moto, identitas kategori, dan identitas *user* dari penyedia hiburan tersebut. Berikut *query* untuk membuatnya.

```

CREATE TABLE IF NOT EXISTS `tb_penyedia` (
  `id_penyedia` int(10) NOT NULL AUTO_INCREMENT,
  `user` int(10) NOT NULL,
  `nama_organisasi` varchar(50) NOT NULL,
  `deskripsi` text NOT NULL,
  `moto` varchar(30) NOT NULL,
  `created` datetime NOT NULL,
  `modified` timestamp     NOT NULL      DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `kategori` int(2) NOT NULL,
  PRIMARY KEY (`id_penyedia`)
) ENGINE=MyISAM           DEFAULT      CHARSET=latin1
AUTO_INCREMENT=1 ;

```

## 21. Tabel tb\_transaksi

Tabel ini digunakan untuk menyimpan data-data transaksi yang dilakukan antara *user* yang telah membuat hajatan dan menyewa hiburan. Berikut *query* untuk membuatnya.

```
CREATE TABLE IF NOT EXISTS `tb_transaksi` (
  `id_transaksi` int(10) NOT NULL AUTO_INCREMENT,
  `hajatan` int(10) NOT NULL,
  `penyedia` int(10) NOT NULL,
  `status` int(1) NOT NULL DEFAULT '0',
  `waktu` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id_transaksi`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1
AUTO_INCREMENT=1 ;
```

## 22. Tabel tb\_ulasan

Tabel ini digunakan untuk menyimpan ulasan dari suatu transaksi seperti *rattting* dan komentar untuk pelayanan yang telah diberikan oleh penyedia hiburan. Berikut *query* untuk membuatnya.

```
CREATE TABLE IF NOT EXISTS `tb_ulasan` (
  `transaksi` int(10) NOT NULL,
  `ratting` int(1) NOT NULL DEFAULT '0',
  `komentar` varchar(250) NOT NULL,
  `waktu` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

## 23. Tabel tb\_user

Tabel ini digunakan untuk menyimpan data-data dari *user* seperti nama, *username*, *password*, tanggal lahir, alamat, jenis kelamin, dan lain sebagainya. Berikut *query* untuk membuatnya.

```

`url_foto`  varchar(200)      NOT      NULL      DEFAULT
'default.jpg',
`cover`     varchar(200)      NOT      NULL      DEFAULT
'default.jpg',
`created`   datetime        NOT      NULL      DEFAULT
CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
`hash`      varchar(32)      NOT      NULL      DEFAULT
`aktif`     int(1)          NOT      NULL      DEFAULT
`status`    int(1)          NOT      NULL      DEFAULT
PRIMARY KEY (`id_user`)
)      ENGINE=MyISAM           DEFAULT      CHARSET=latin1
AUTO_INCREMENT=1 ;

```

#### **24. Tabel tb\_visitor**

Tabel ini digunakan untuk menyimpan nilai hasil *counting* banyaknya pengunjung *website* pada tanggal itu. Berikut *query* untuk membuatnya.

```

CREATE TABLE IF NOT EXISTS `tb_visitor` (
`total_hits` int(12) NOT NULL,
`unique_hits` int(12) NOT NULL,
`tanggal` date NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

```

#### **25. Tabel vidio\_promosi**

Tabel ini digunakan oleh user penyedia hiburan untuk menyimpan *link* dari youtube guna menarik minat dari *user* konsumen yang akan melakukan penyewaan hiburan untuk kebutuhan hanjatannya. Berikut *query* untuk membuatnya.

```

CREATE TABLE IF NOT EXISTS `vidio_promosi` (
`penyedia` int(10) NOT NULL,
`url_youtube` text NOT NULL,
`waktu`     timestamp     NOT      NULL      DEFAULT
CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

```

#### **26. Tabel villages**

Tabel ini digunakan untuk menyimpan data-data desa yang ada di negara Indonesia. Berikut *query* untuk membuatnya.

```

CREATE TABLE IF NOT EXISTS `villages` (
`id` char(10) COLLATE utf8_unicode_ci NOT NULL,

```

```

    `district_id` char(7) COLLATE utf8_unicode_ci NOT
    NULL,
    `name` varchar(255) COLLATE utf8_unicode_ci NOT
    NULL,
    PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8
COLLATE=utf8_unicode_ci;

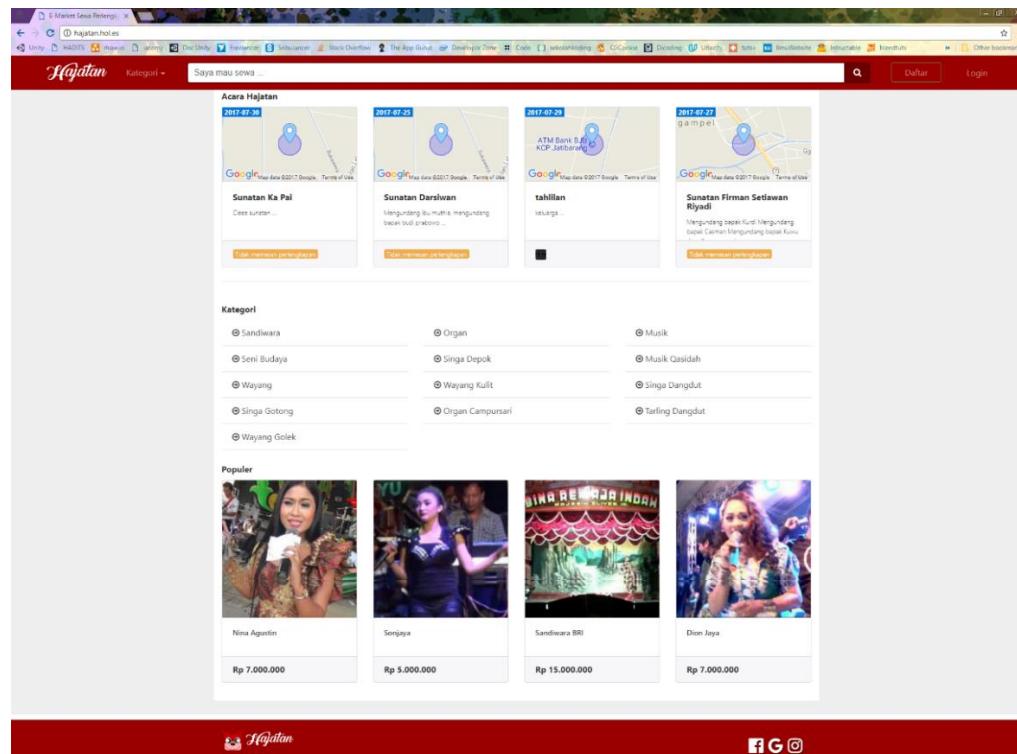
```

#### 4.2.1 Implementasi Desain *Interface* dan Program

Implementasi desain *interface* dan program merupakan tahap untuk mengimplementasikan desain yang telah dirancang pada bab sebelumnya. Setiap desain halaman/*form/popover* dibuat dengan menggunakan dreamweaver CC 2015. Antarmuka atau *interface* merupakan cara untuk memudahkan pengguna/*user* dalam melakukan aktivitas dalam aplikasi. Untuk *sintaks* program sendiri dapat dilihat pada lampiran. Berikut beberapa tampilan *interface* dalam aplikasi e-market hiburan untuk kebutuhan hajatan.

##### 1. Tampilan Halaman Awal

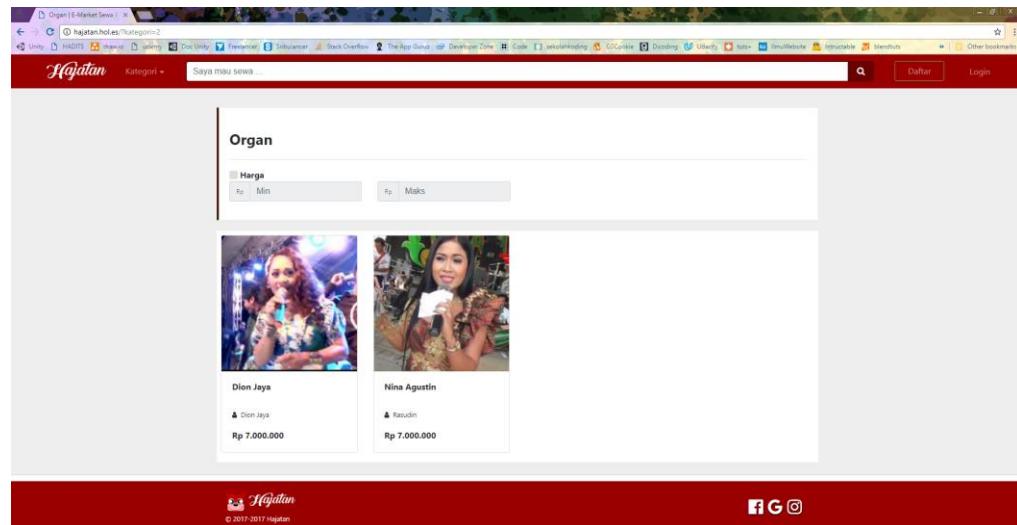
Pertama kali pengunjung mengakses *website* maka akan menampilkan halaman ini. Pada halaman tersebut terdapat tombol untuk melakukan *login* atau daftar. Berikut tampilan halaman awal dapat dilihat pada Gambar 4.1.



Gambar 4.1 Tampilan Halaman Awal

## 2. Tampilan Halaman *List* Penyedia Hiburan

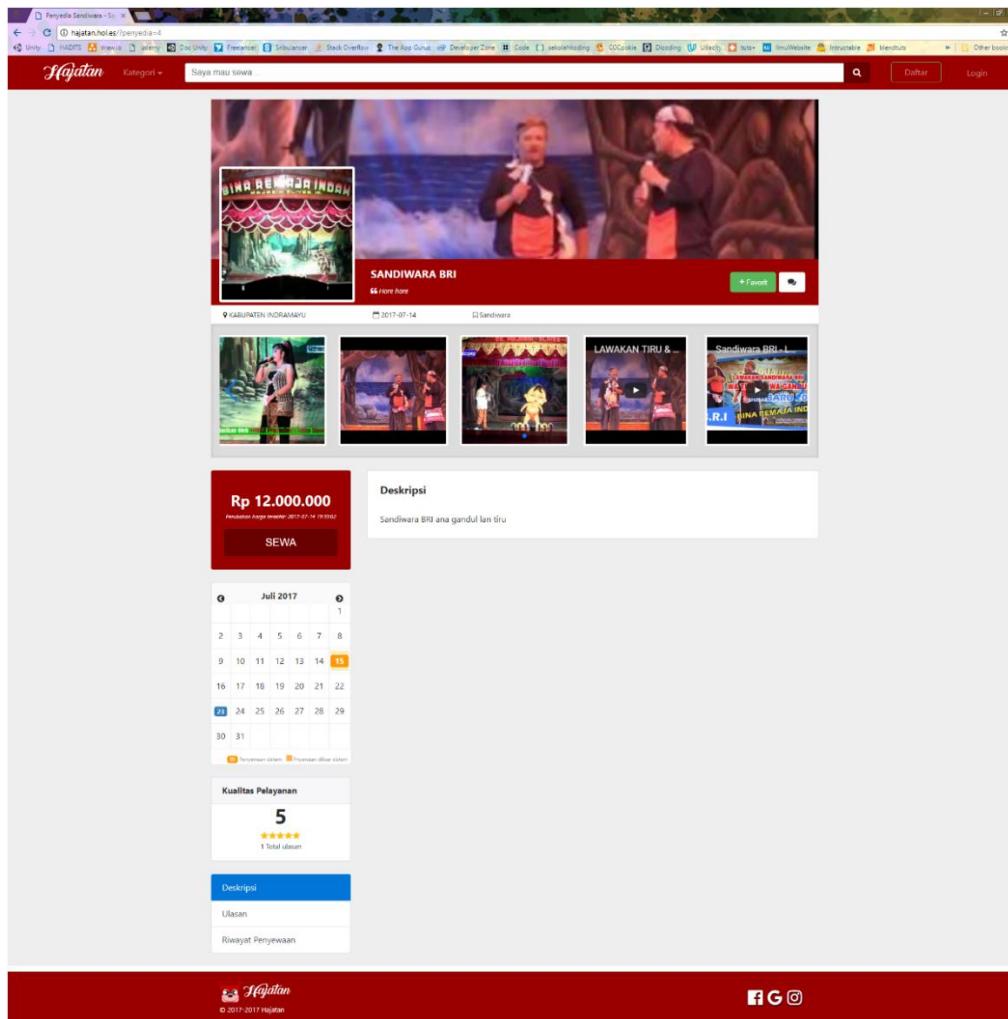
Pada saat pengunjung memilih kategori maka akan menuju ke tampilan halaman *list* penyedia hiburan dari kategori yang terpilih. Berikut tampilan halaman *list* penyedia hiburan dapat dilihat pada Gambar 4.2.



Gambar 4.2 Tampilan Halaman *List* Penyedia Hiburan

## 3. Tampilan Halaman Deskripsi Penyedia Hiburan

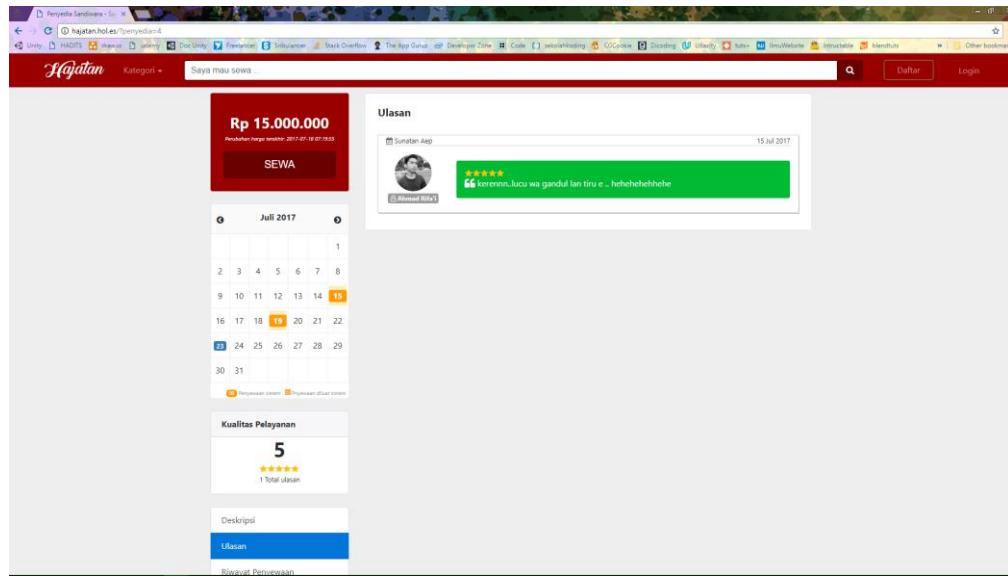
Pada saat pengunjung aplikasi memilih penyedia hiburan yang tertera pada tampilan halaman *list* penyedia hiburan maka akan menuju ke tampilan halaman penyedia hiburan yang secara *default* menampilkan deskripsi dari penyedia hiburan. Berikut tampilan halaman deskripsi penyedia hiburan dapat dilihat pada Gambar 4.3.



Gambar 4.3 Tampilan Halaman Deskripsi Penyedia Hiburan

#### 4. Tampilan Halaman Ulasan Penyedia Hiburan

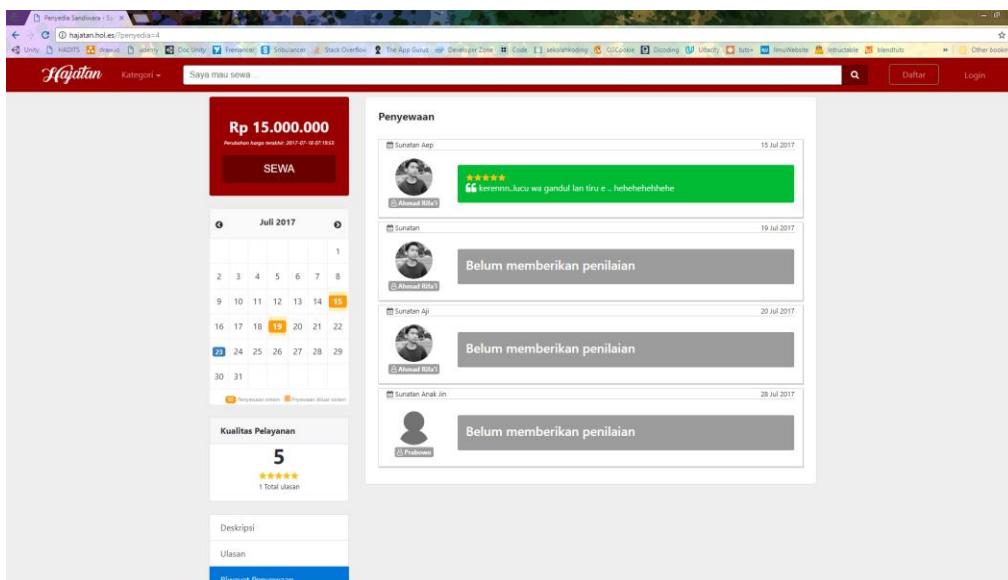
Ketika pengunjung memilih ulasan pada halaman penyedia hiburan maka akan menampilkan hasil ulasan dari *user* yang pernah menyewa hiburan tersebut. Berikut tampilan halaman ulasan penyedia hiburan dapat dilihat pada Gambar 4.4.



Gambar 4.4 Tampilan Halaman Ulasan Penyedia Hiburan

## 5. Tampilan Halaman Riwayat Penyewaan Penyedia Hiburan

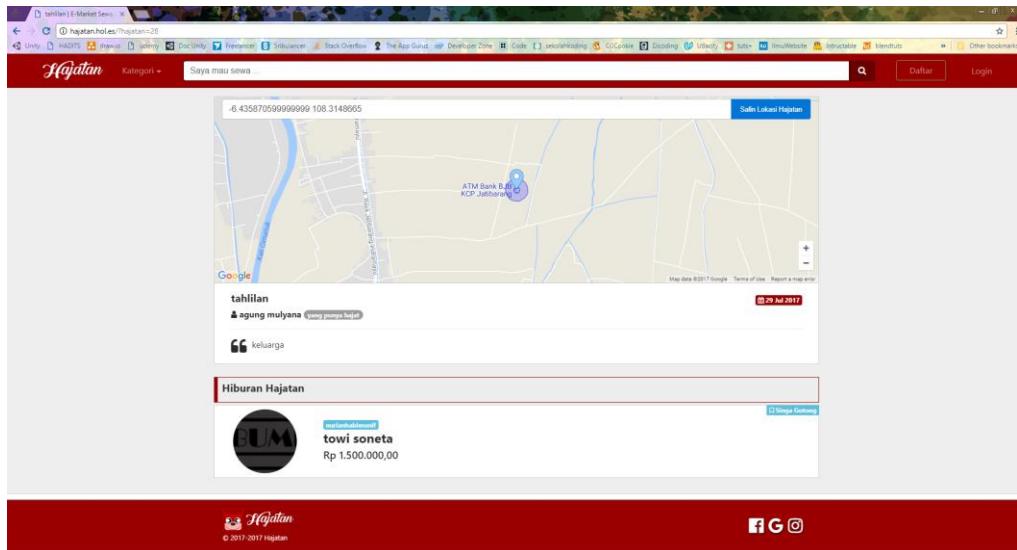
Ketika pengunjung memilih riwayat penyewaan pada halaman penyedia hiburan maka akan menampilkan informasi *user* yang pernah menyewa hiburan tersebut. Berikut tampilan halaman ulasan penyedia hiburan dapat dilihat pada Gambar 4.5.



Gambar 4.5 Tampilan Halaman Riwayat Penyewaan Penyedia Hiburan

## 6. Tampilan Halaman Detail Informasi Hajatan

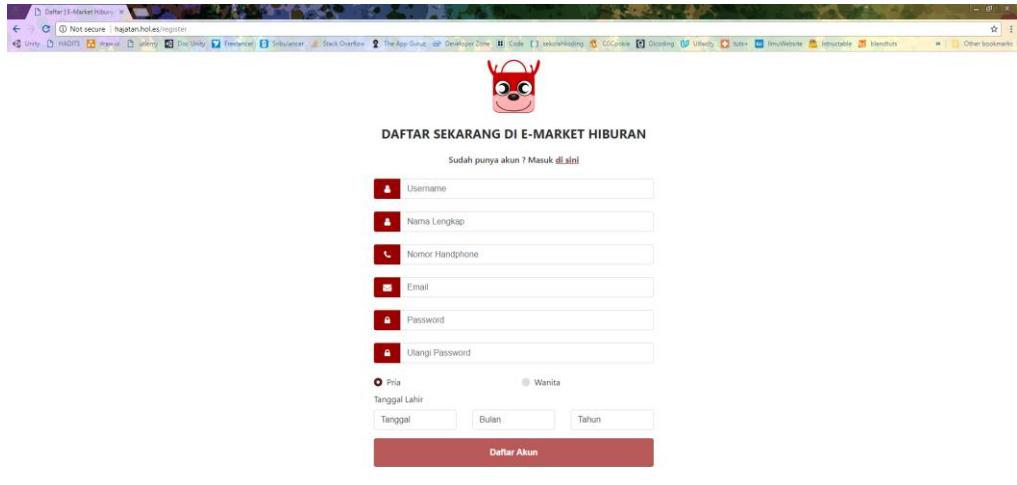
Pada halaman awal ketika pengunjung memilih salah satu hajatan maka akan menuju ke tampilan halaman detail informasi hajatan. Berikut tampilan halaman detail informasi hajatan dapat dilihat pada Gambar 4.6.



Gambar 4.6 Tampilan Halaman Detail Informasi Hajatan

## 7. Tampilan Halaman Register

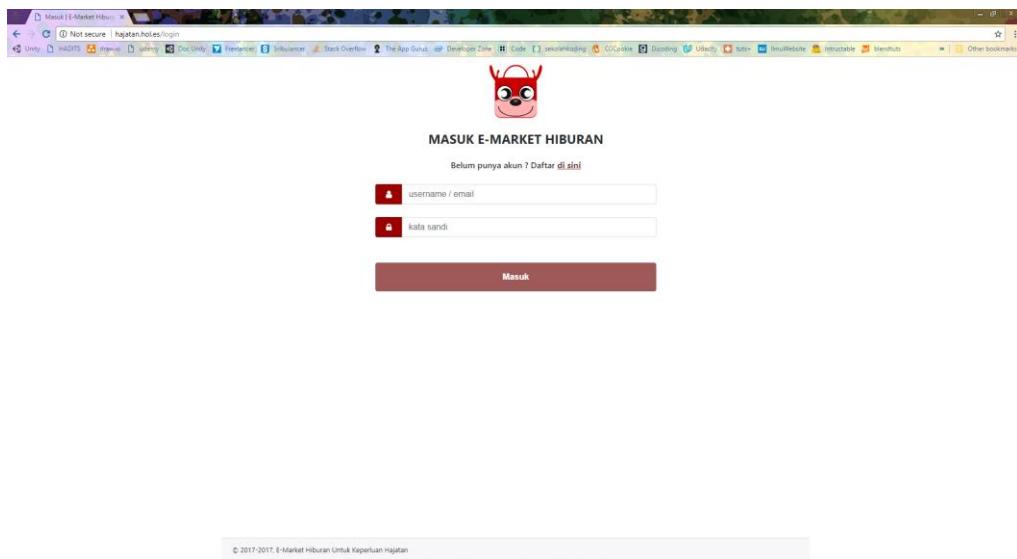
Pengunjung yang ingin mendaftar sebagai pengguna aplikasi mengisi *form* yang ada pada halaman *register*. Berikut tampilan halaman *register* dapat dilihat pada Gambar 4.7.



Gambar 4.7 Tampilan Halaman Register

## 8. Tampilan Halaman Login

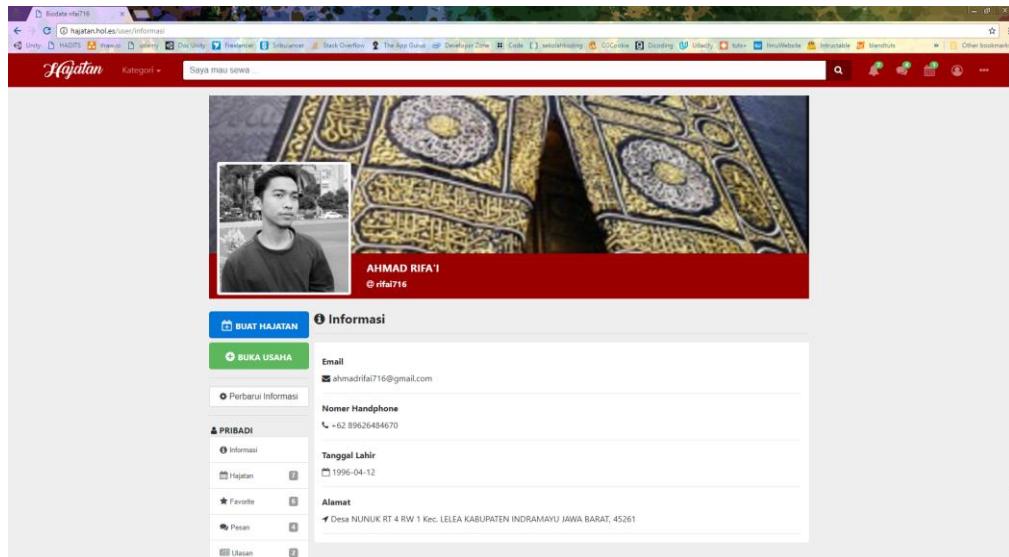
Pengunjung yang telah melakukan *register* dan ketika sudah mendapatkan *link* konfirmasi email yang dikirimkan ke emailnya maka ketika *link* konfirmasi tersebut diklik akan menuju ke halaman *login*. Berikut tampilan halaman *login* dapat dilihat pada Gambar 4.8.



Gambar 4.8 Tampilan Halaman *Login*

## 9. Tampilan Halaman Informasi *User*

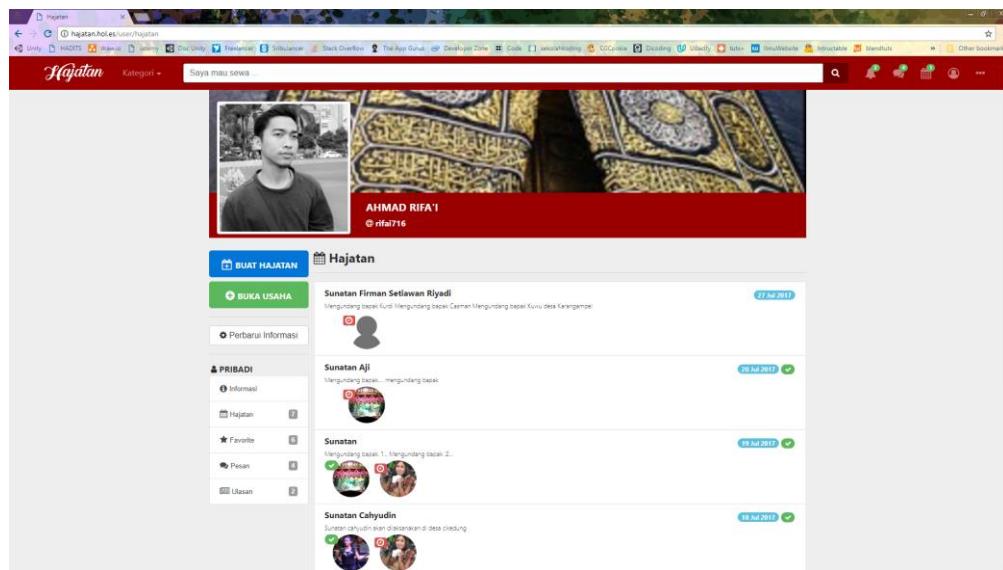
Halaman ini merupakan halaman yang dapat diakses oleh *user* yang telah melakukan login. Halaman ini akan menampilkan informasi biodata *user*. Berikut tampilan halaman informasi dapat dilihat pada Gambar 4.9.



Gambar 4.9 Tampilan Halaman Informasi *User*

## 10. Tampilan Halaman *List Hajatan User*

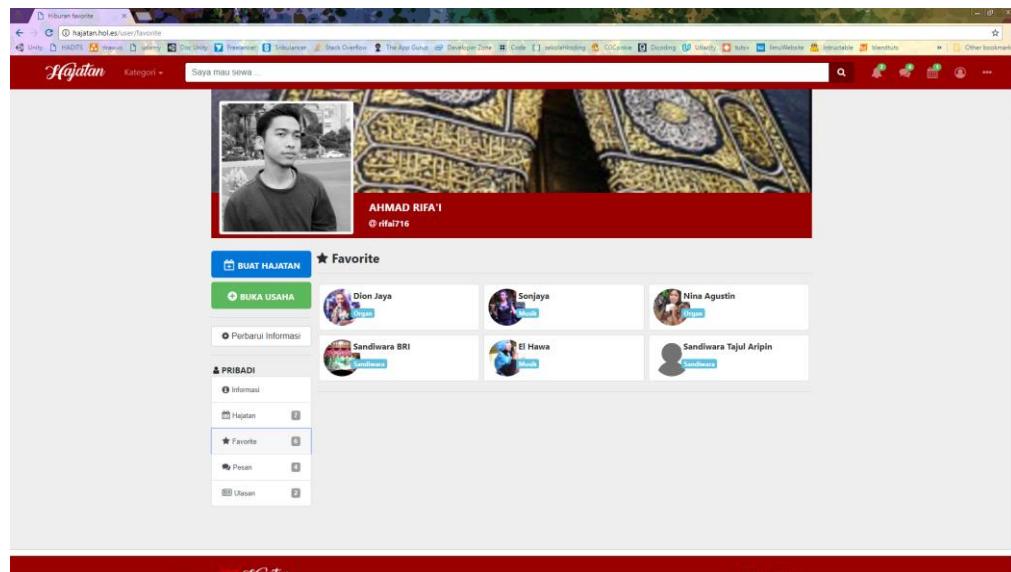
Halaman ini menampilkan informasi acara hajatan-hajatan yang telah dibuat dan dapat dilihat juga hiburan yang disewa pada hajatan-hajatan tersebut. Berikut tampilan halaman *list* hajatan *user* dapat dilihat pada Gambar 4.10.



Gambar 4.10 Tampilan Halaman *List Hajatan User*

## 11. Tampilan Halaman *List Favorite*

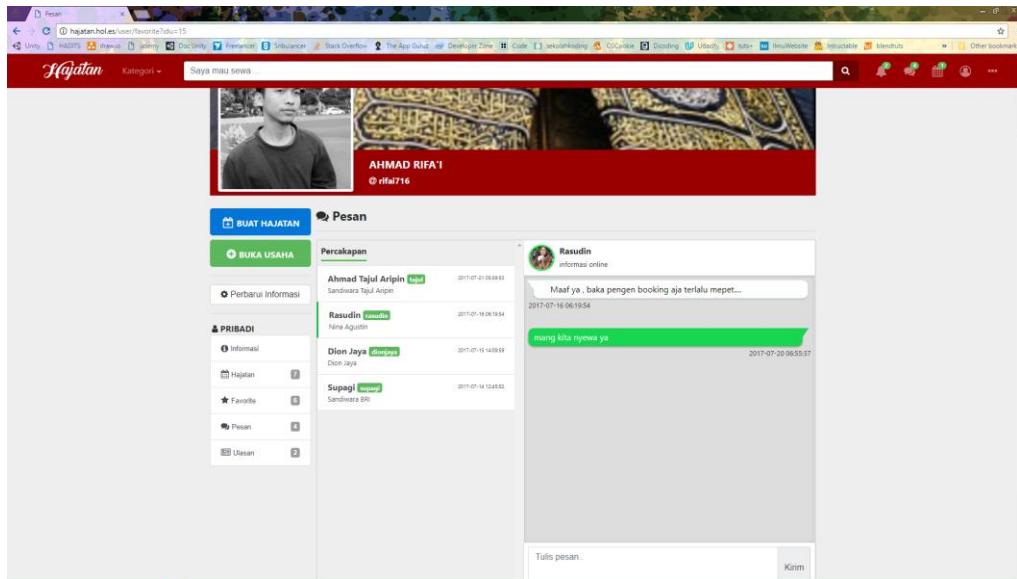
Halaman ini menampilkan daftar penyedia hiburan yang telah difavoritkan. Berikut tampilan halaman *list favorite* dapat dilihat pada Gambar 4.11.



Gambar 4.11 Tampilan Halaman *List Favorite*

## 12. Tampilan Halaman Pesan

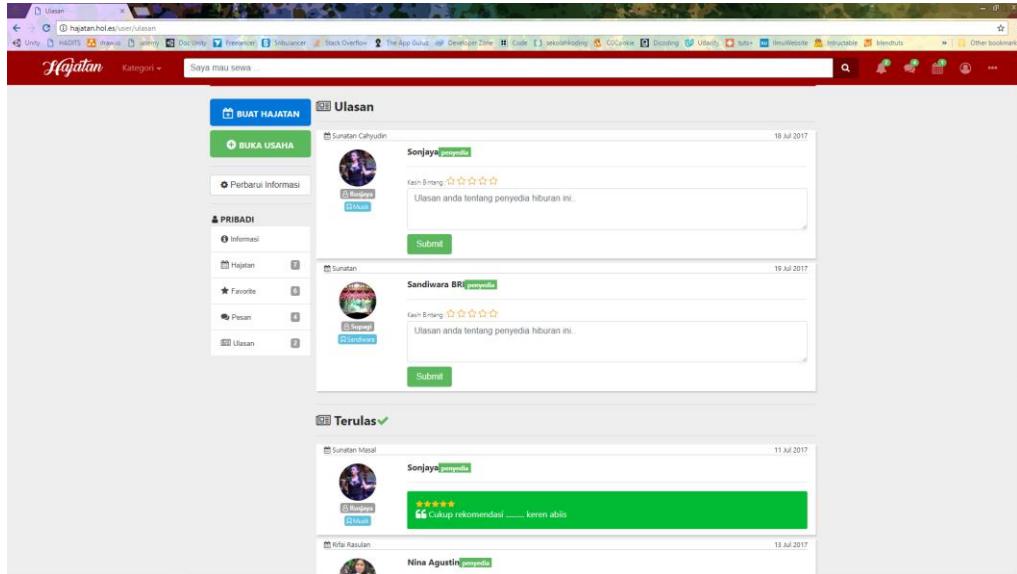
Halaman ini berfungsi untuk melakukan *chatting/kirim pesan* dengan penyedia hiburan. Berikut tampilan halaman pesan dapat dilihat pada Gambar 4.12.



Gambar 4.12 Tampilan Halaman Pesan

### 13. Tampilan Halaman Ulasan

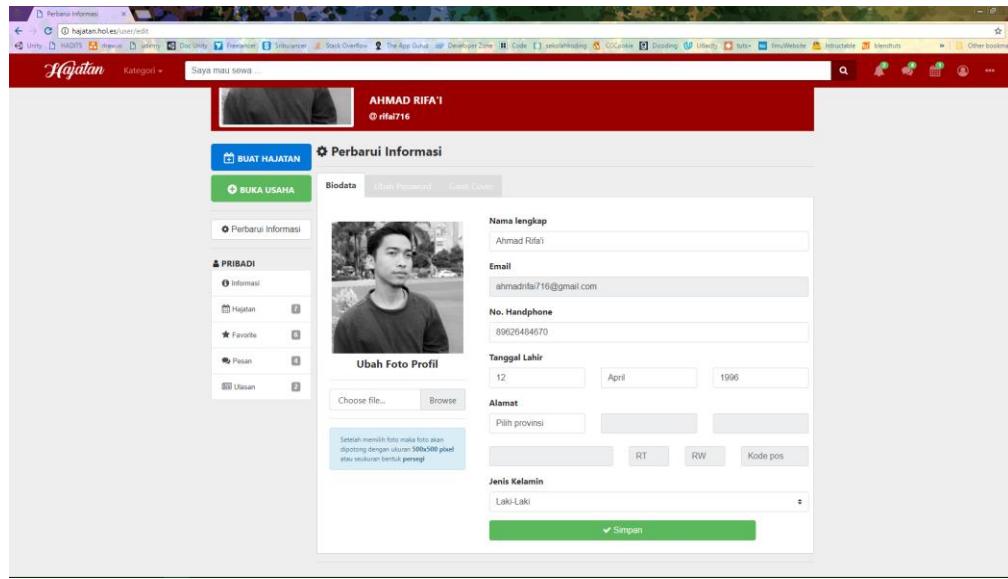
Halaman ini berfungsi untuk melakukan ulasan kepada penyedia hiburan yang telah disewa. Berikut tampilan halaman ulasan dapat dilihat pada Gambar 4.13.



Gambar 4.13 Tampilan Halaman Ulasan

### 14. Tampilan Halaman Perbarui Informasi

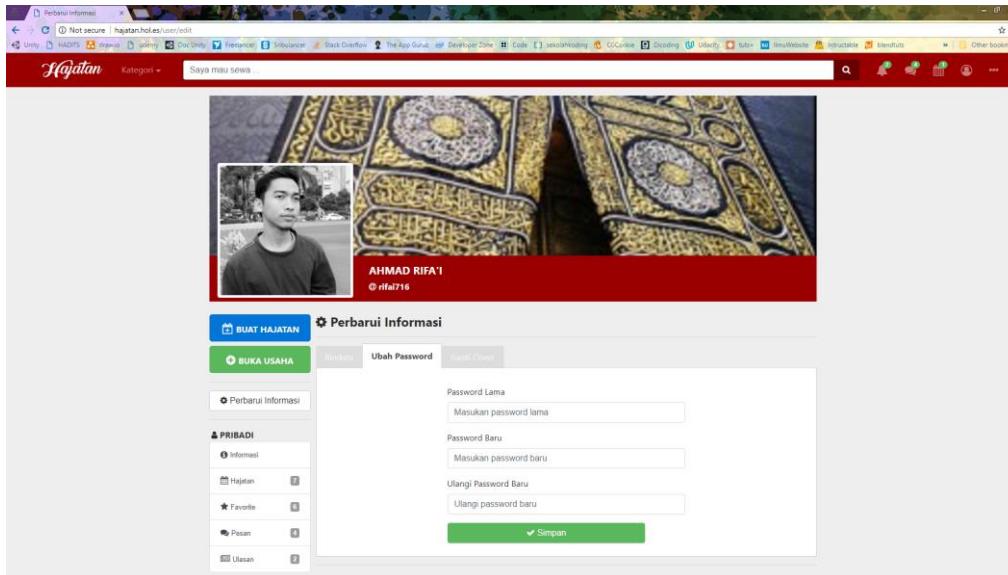
Halaman ini berfungsi untuk melakukan edit atau perubahan informasi biodata diri dan juga dapat digunakan untuk melakukan perubahan foto profil. Berikut tampilan halaman perbarui informasi dapat dilihat pada Gambar 4.14.



Gambar 4.14 Tampilan Halaman Perbarui Informasi

### 15. Tampilan Halaman Ubah Password

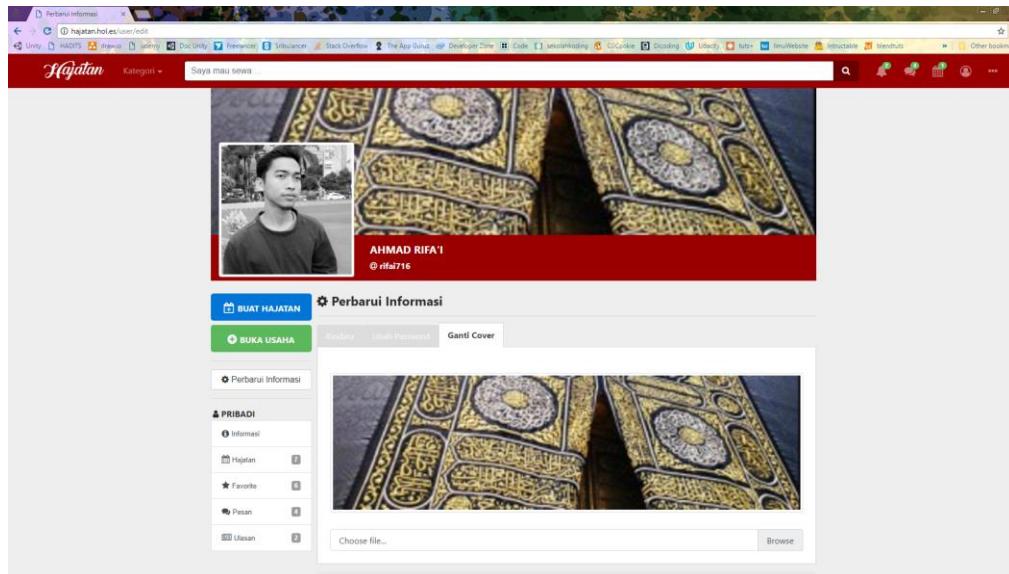
Halaman ini berfungsi untuk mengganti *password* akun. Berikut tampilan halaman ubah *password* dapat dilihat pada Gambar 4.15.



Gambar 4.15 Tampilan Halaman Ubah Password

### 16. Tampilan Halaman Ganti Cover

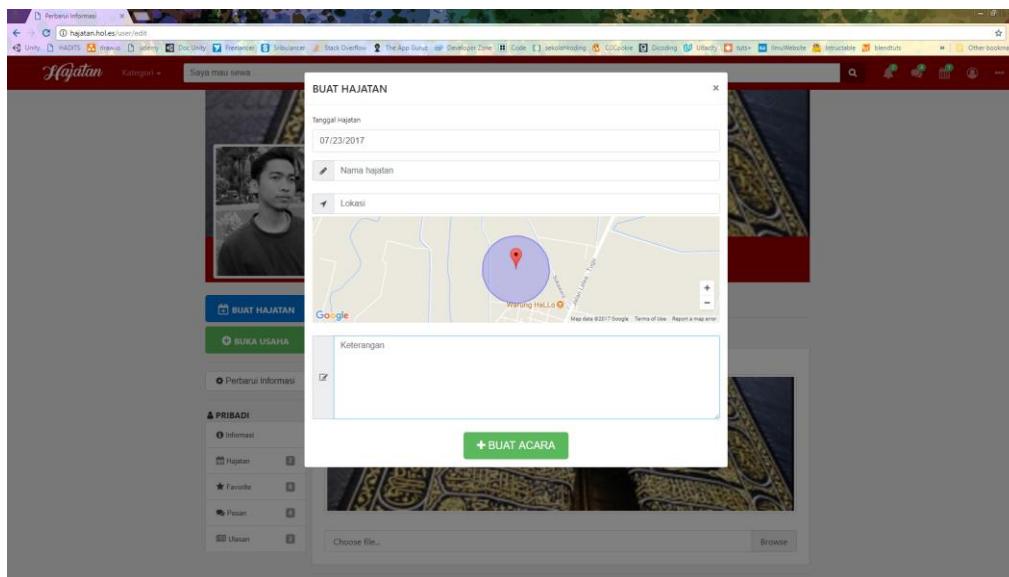
Halaman ini berfungsi untuk melakukan ganti *cover*. Berikut tampilan halaman ganti *cover* dapat dilihat pada Gambar 4.16.



Gambar 4.16 Tampilan Halaman Ganti Cover

### 17. Tampilan Halaman Buat Hajatan

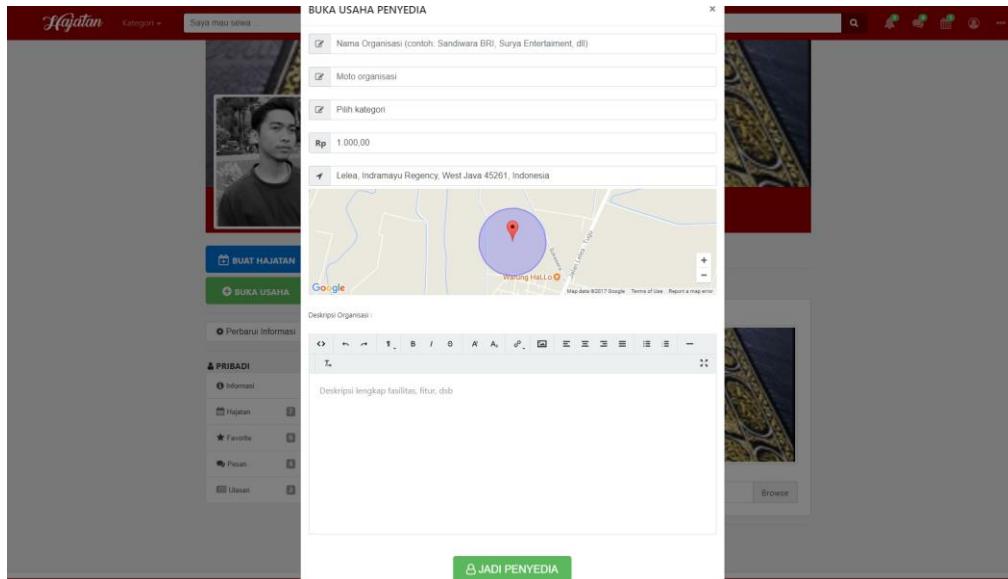
Halaman ini berfungsi untuk melakukan pembuatan acara hajatan. Berikut tampilan halaman buat hajatan dapat dilihat pada Gambar 4.17.



Gambar 4.17 Tampilan Halaman Buat Hajatan

### 18. Tampilan Halaman Buka Usaha

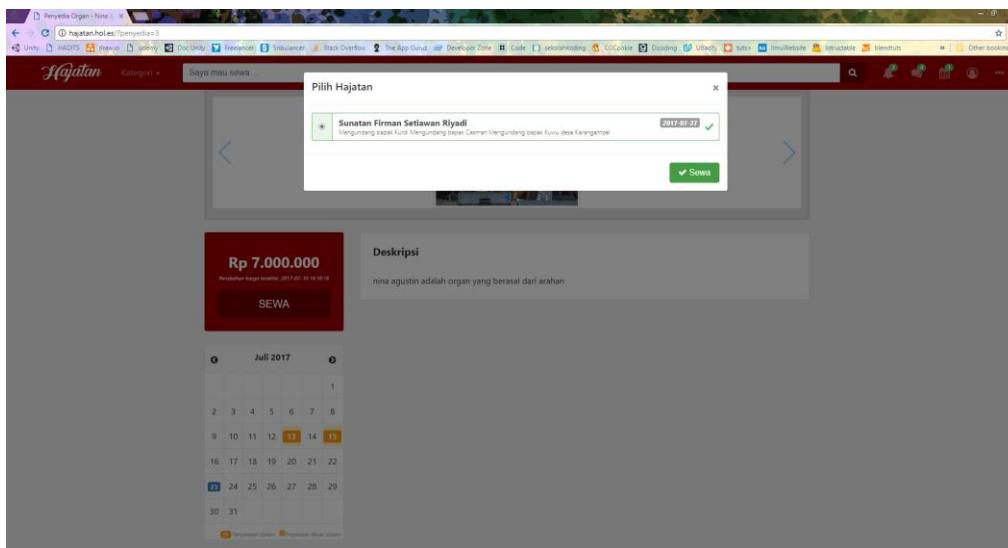
Halaman ini berfungsi untuk membuka usaha. Apabila *user* yang mempunyai usaha dan ingin membuka usaha dapat dilakukan dengan mengisi *form* pada halaman ini. Berikut tampilan halaman buka usaha dapat dilihat pada Gambar 4.18.



Gambar 4.18 Tampilan Halaman Buka Usaha

## 19. Tampilan Halaman Sewa Hiburan

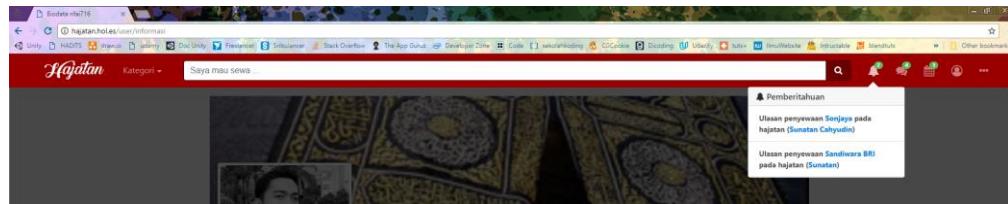
Halaman ini berfungsi untuk melakukan penyewaan pada hajatan yang telah dibuat. Berikut tampilan halaman sewa hiburan dapat dilihat pada Gambar 4.19.



Gambar 4.19 Tampilan Halaman Sewa Hiburan

## 20. Tampilan Popover Notification

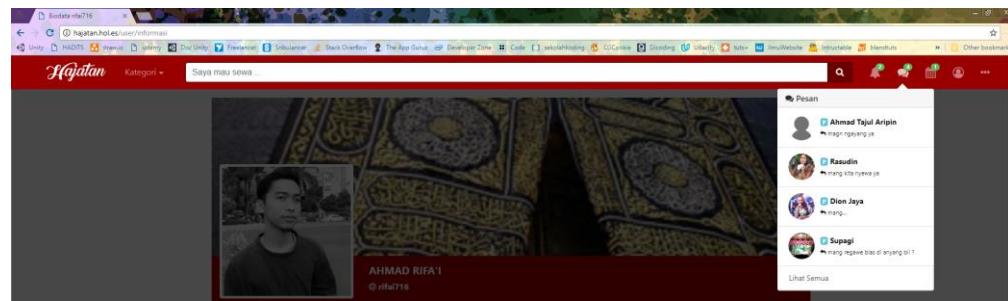
Navigasi ini berfungsi untuk memberitahukan ulasan-ulasan yang belum dilakukan. Berikut tampilan *popover notification* dapat dilihat pada Gambar 4.20.



Gambar 4.20 Tampilan *Popover Notification*

## 21. Tampilan *Popover Pesan*

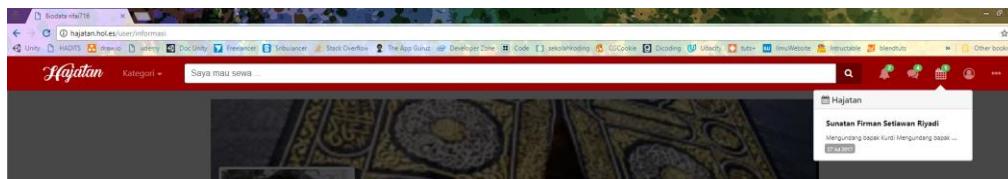
Navigasi ini berfungsi untuk memberitahukan cuplikan pesan/*chatting*. Berikut adalah tampilan *popover* pesan dapat dilihat pada Gambar 4.21.



Gambar 4.21 Tampilan *Popover Pesan*

## 22. Tampilan *Popover Jadwal Hajatan*

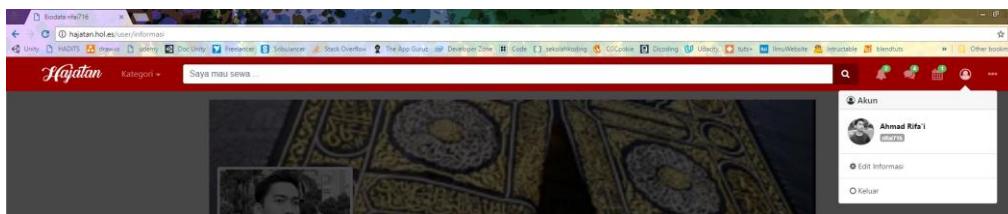
Navigasi ini berfungsi untuk memberitahukan hajatan yang akan atau sedang dilaksanakan. Berikut tampilan *popover* jadwal hajatan dapat dilihat pada Gambar 4.22.



Gambar 4.22 Tampilan *Popover Jadwal Hajatan*

## 23. Tampilan *Popover Profil*

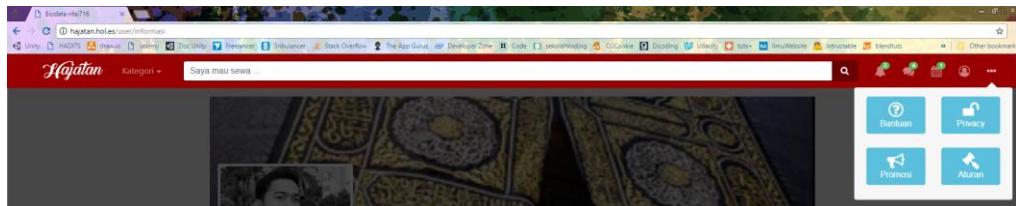
Navigasi ini berfungsi untuk menampilkan informasi *user* yang sedang *login*. Berikut tampilan *popover* profil dapat dilihat pada Gambar 4.23.



Gambar 4.23 Tampilan *Popover Profil*

## 24. Tampilan Popover Menu

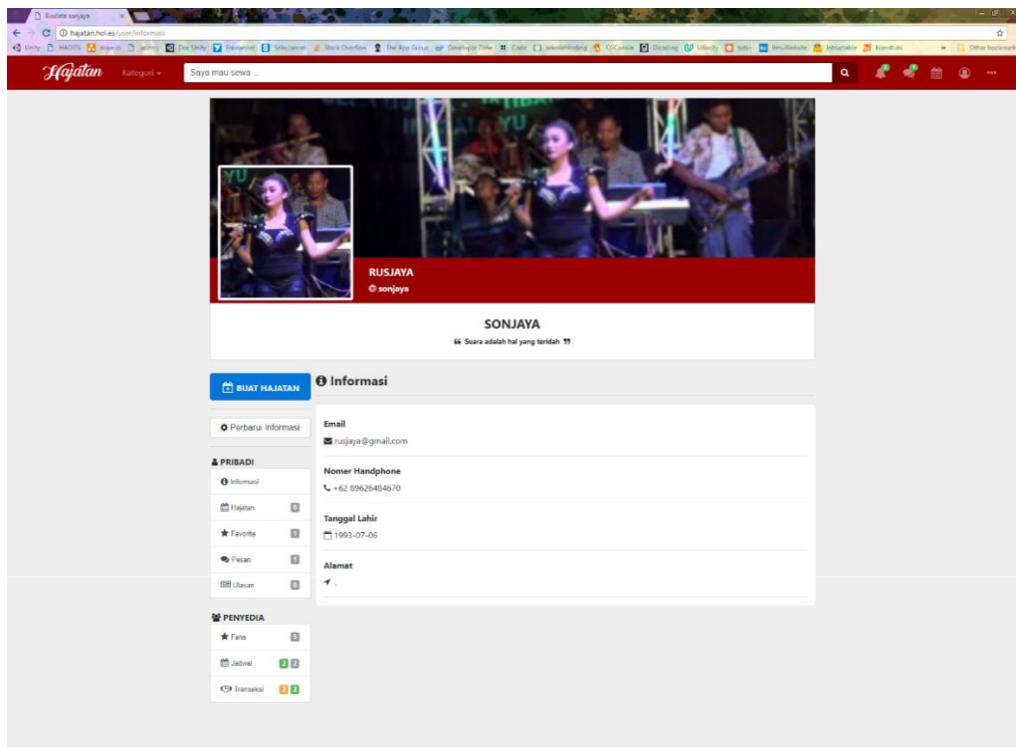
Navigasi ini berfungsi untuk pintasan mengakses menu-menu khusus dalam aplikasi ini. Berikut tampilan popover menu dapat dilihat pada Gambar 4.24.



Gambar 4.24 Tampilan Popover Menu

## 25. Tampilan Halaman Penyedia Hiburan

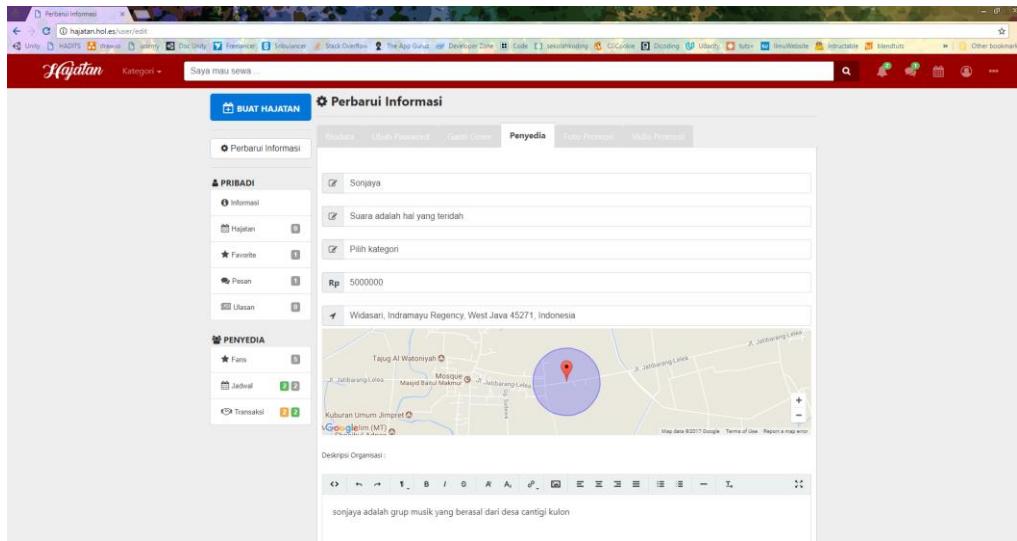
Halaman ini merupakan halaman yang dapat diakses oleh *user* yang telah berganti status menjadi penyedia hiburan. Berikut tampilan halaman penyedia hiburan dapat dilihat pada Gambar 4.25.



Gambar 4.25 Tampilan Halaman Penyedia Hiburan

## 26. Tampilan Halaman Edit Informasi Penyedia Hiburan

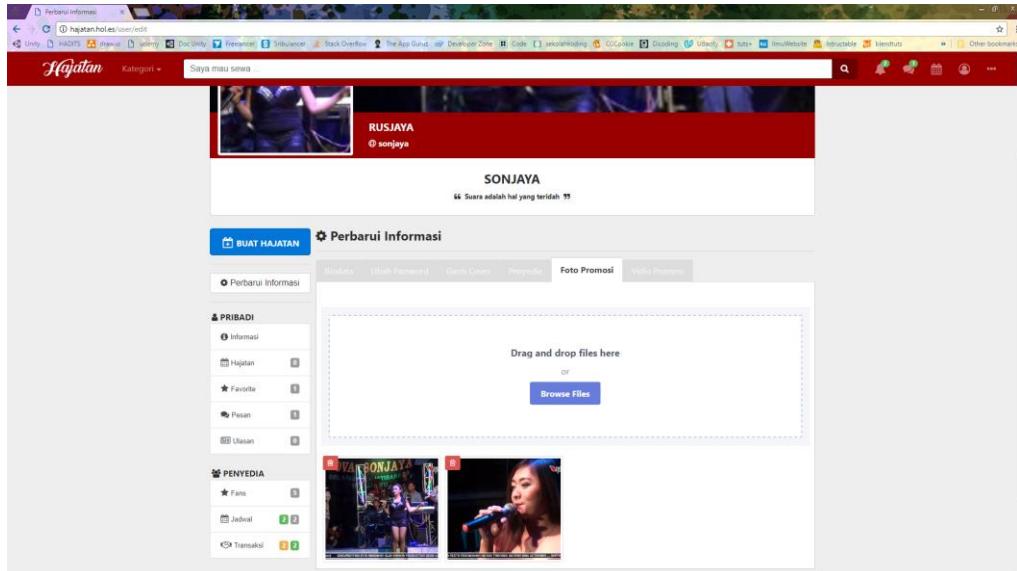
Halaman ini digunakan oleh penyedia hiburan untuk mengubah informasi tentang hiburan yang ditawarkannya, seperti merubah harga, lokasi, atau deskripsi dari hiburan. Berikut tampilan halaman edit informasi penyedia hiburan dapat dilihat pada Gambar 4.26.



Gambar 4.26 Tampilan Halaman Edit Informasi Penyedia Hiburan

## 27. Tampilan Halaman Tambah Foto Promosi

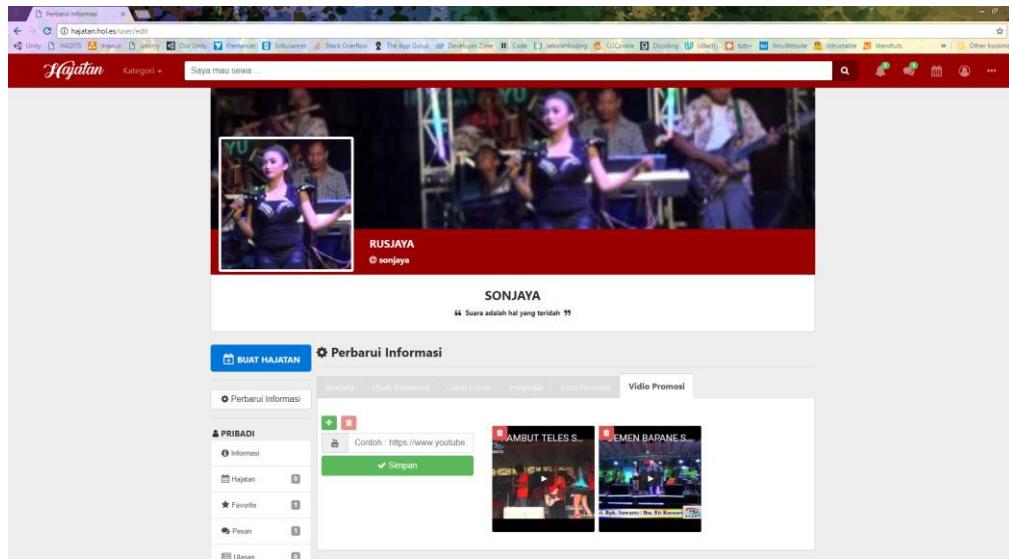
Halaman ini berfungsi untuk menambahkan foto-foto guna menarik minat pengunjung halaman penyedia untuk menyewa hiburannya. Berikut adalah tampilan halaman tambah foto promosi dapat dilihat pada Gambar 4.27.



Gambar 4.27 Tampilan Halaman Tambah Foto Promosi

## 28. Tampilan Halaman Tambah Vidio Promosi

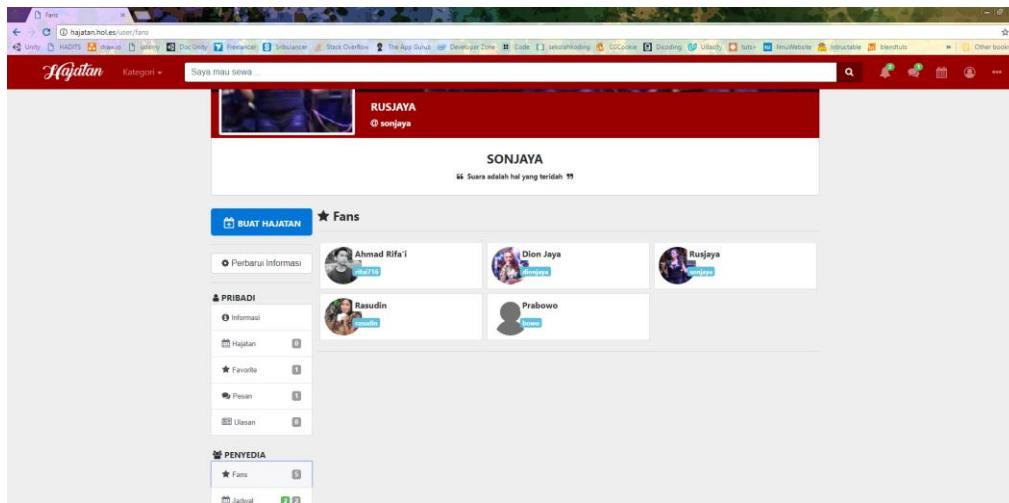
Halaman ini berfungsi untuk menambahkan vidio-vidio yang bersumber dari youtube dimana vidio bisa berupa kegiatan yang sudah dilakukan ataupun vidio lainnya guna menarik minat pengunjung halaman penyedia untuk menyewa hiburannya. Berikut tampilan halaman tambah vidio promosi dapat dilihat pada Gambar 4.28.



Gambar 4.28 Tampilan Halaman Tambah Vidio Promosi

## 29. Tampilan Halaman Fans

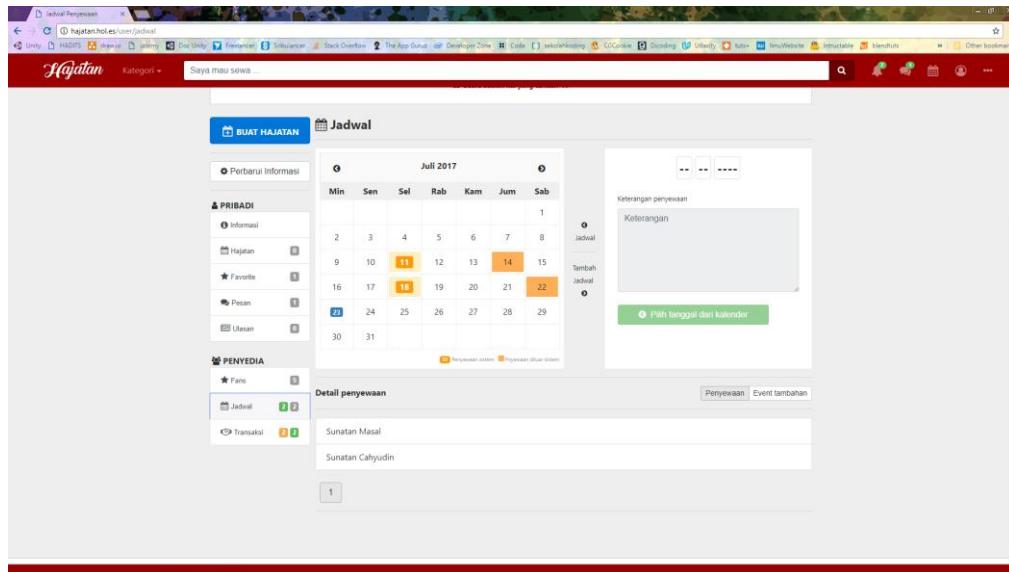
Halaman ini digunakan oleh penyedia hiburan untuk memantau atau melihat penyuksesan dari hiburannya. Berikut adalah tampilan halaman fans dapat dilihat pada Gambar 4.29.



Gambar 4.29 Tampilan Halaman Fans

## 30. Tampilan Halaman Jadwal

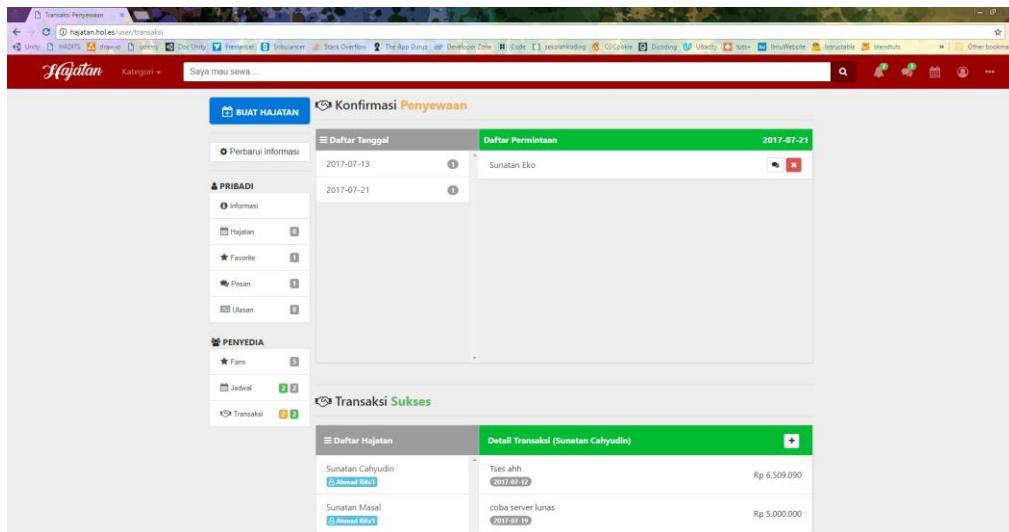
Halaman ini berfungsi untuk memantau jadwal penyewaan dan juga dapat menambahkan jadwal penyewaan yang dilakukan diluar sistem ini. Berikut tampilan halaman jadwal dapat dilihat pada Gambar 4.30.



Gambar 4.30 Tampilan Halaman Jadwal

### 31. Tampilan Halaman Transaksi

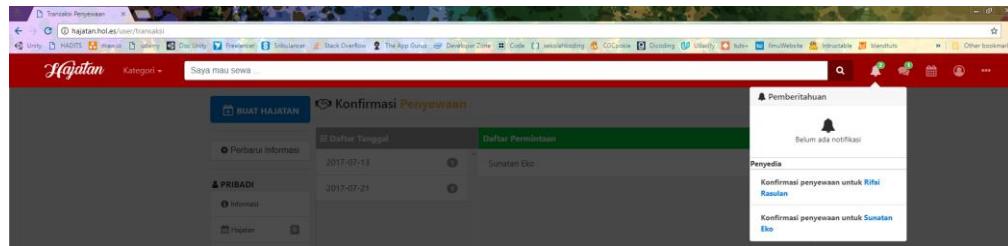
Halaman ini adalah halaman yang digunakan oleh *user* penyedia hiburan untuk memantau transaksi dan juga menambahkan detail dari transaksi penyewaan. Berikut tampilan halaman transaksi dapat dilihat pada Gambar 4.31.



Gambar 4.31 Tampilan Halaman Transaksi

### 32. Tampilan Popover Notification Penyedia Hiburan

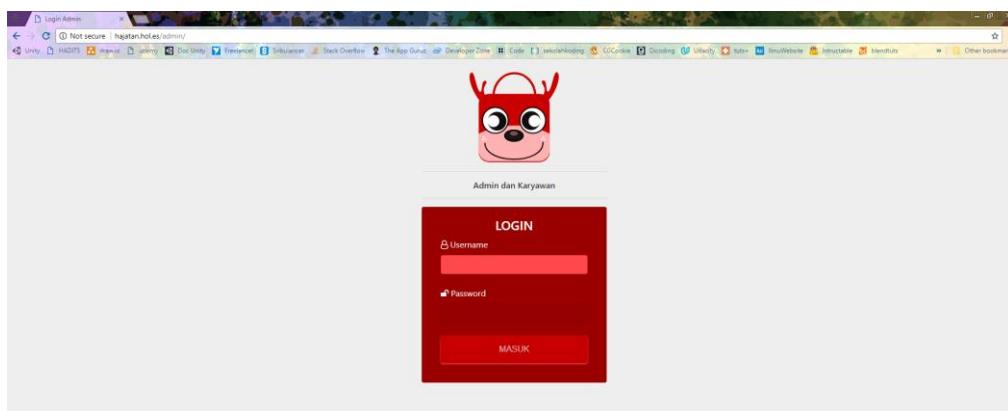
Navigasi ini berfungsi untuk menampilkan informasi permintaan penyewaan kepada *user* penyedia hiburan. Berikut adalah tampilan *popover notification* penyedia hiburan dapat dilihat pada Gambar 4.32.



Gambar 4.32 Tampilan *Popover Notification* Penyedia Hiburan

### 33. Tampilan Halaman *Login* Admin dan Pegawai

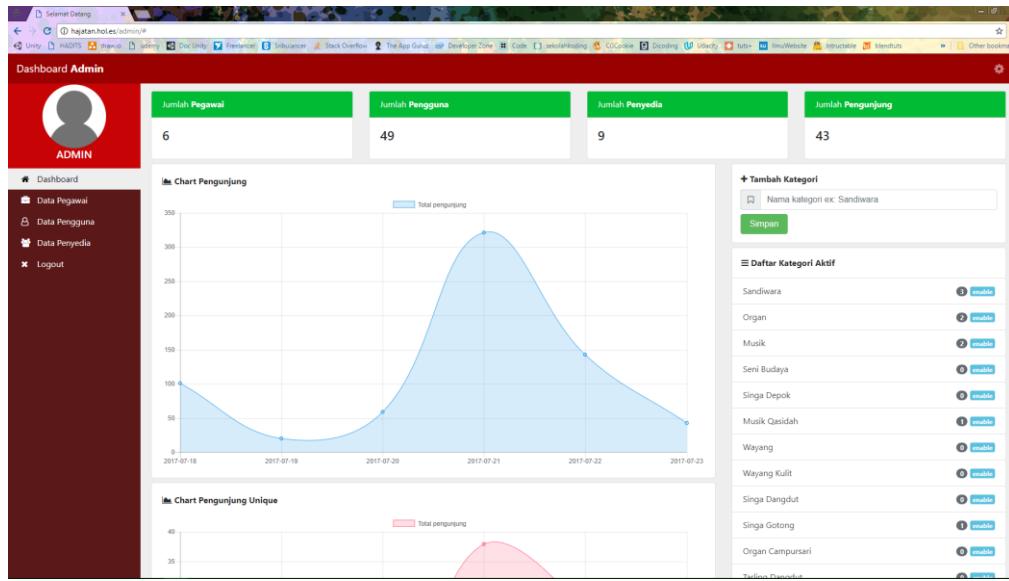
Halaman ini berfungsi untuk masuk ke halaman admin atau pegawai untuk mengontrol aktivitas dalam aplikasi e-market hiburan untuk kebutuhan hajatan. Berikut tampilan halaman *login* admin dan pegawai dapat dilihat pada Gambar 4.33.



Gambar 4.33 Tampilan Halaman *Login* Admin dan Pegawai

### 34. Tampilan Halaman *Dashboard* Admin dan Pegawai

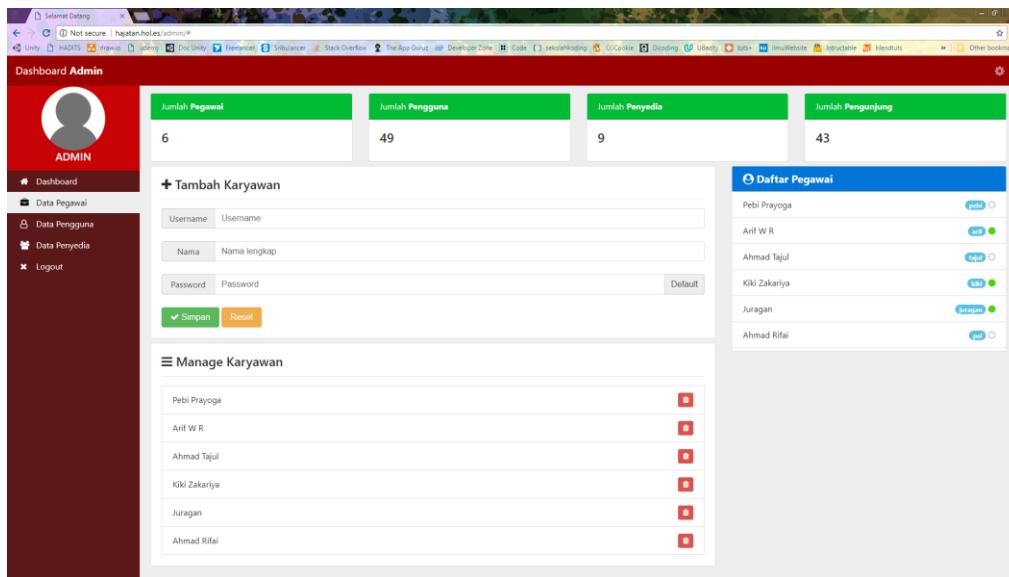
Halaman ini berfungsi untuk melihat statistik dari aplikasi e-market hiburan untuk kebutuhan hajatan. Berikut tampilan halaman *dashboard* admin dan pegawai dapat dilihat pada Gambar 4.34.



Gambar 4.34 Tampilan Halaman *Dashboard* Admin dan Pegawai

### 35. Tampilan Halaman Data Pegawai

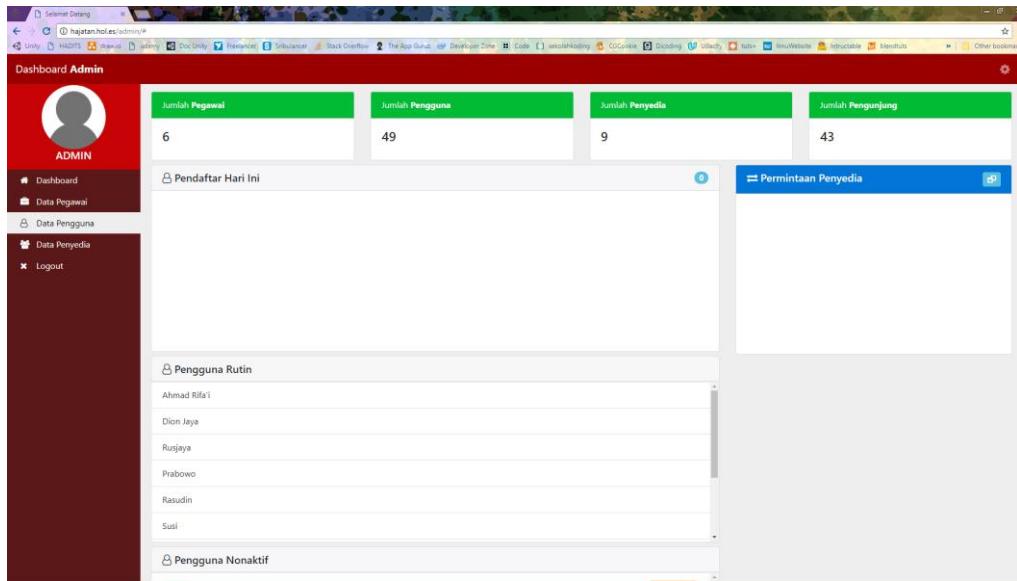
Halaman ini digunakan admin untuk menambahkan dan mengontrol aktivitas pegawai. Berikut tampilan halaman data pegawai dapat dilihat pada Gambar 4.35.



Gambar 4.35 Tampilan Halaman Data Pegawai

### 36. Tampilan Halaman Data *User*

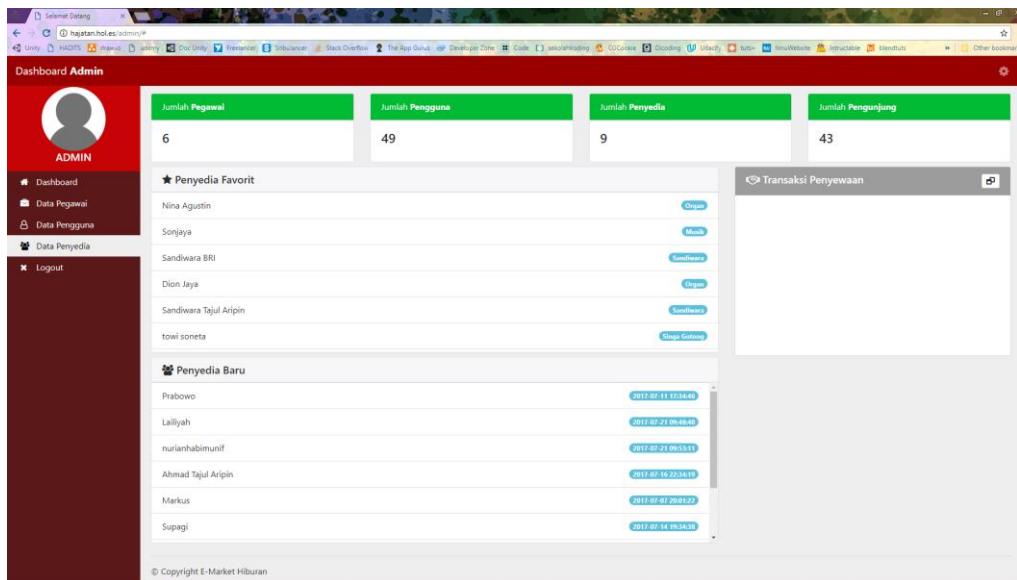
Halaman ini berfungsi untuk mengontrol aktivitas *user*. Berikut tampilan halaman data *user* dapat dilihat pada Gambar 4.36.



Gambar 4.36 Tampilan Halaman Data *User*

### 37. Tampilan Halaman Data *User* Penyedia Hiburan

Halaman ini berfungsi untuk mengontrol aktivitas - aktivitas *user* penyedia hiburan. Berikut tampilan halaman data *user* penyedia hiburan dapat dilihat pada Gambar 4.37.



Gambar 4.37 Tampilan Halaman Data *User* Penyedia Hiburan

#### 4.2.2 Pengujian Aplikasi

Ketika semuanya sudah diimplementasikan tahap selanjutnya yaitu dilakukan pengujian. Pengujian aplikasi menggunakan *Black-Box Testing*. Hasil pengujian aplikasi dapat dilihat pada Tabel 4.1.

Tabel 4.1 Tabel Hasil Pengujian dengan *Black-Box*

No	Deskripsi	Prosedur Pengujian	Keluaran yang diharapkan	Hasil yang didapat	Kesimpulan
<b>Pengunjung</b>					
1	Pengunjung ingin melihat halaman awal	• Masukan URL	Pengunjung dapat melihat halaman awal	Halaman awal dapat dilihat oleh Pengunjung	berhasil
2	Pengunjung ingin melihat halaman <i>list</i> penyedia hiburan	• Pilih salah satu kategori	Pengunjung dapat melihat halaman <i>list</i> penyedia hiburan	Halaman <i>list</i> penyedia hiburan dapat dilihat oleh Pengunjung	berhasil
3	Pengunjung ingin melihat halaman deskripsi penyedia hiburan	• Pilih salah satu penyedia hiburan	Pengunjung dapat melihat halaman deskripsi penyedia hiburan	Halaman deskripsi penyedia hiburan dapat dilihat oleh Pengunjung	berhasil
4	Pengunjung ingin melihat halaman ulasan penyedia hiburan	• Pilih menu ulasan	Pengunjung dapat melihat halaman ulasan penyedia hiburan	Halaman ulasan penyedia hiburan dapat dilihat oleh Pengunjung	berhasil
5	Pengunjung ingin melihat halaman riwayat penyewaan penyedia hiburan	• Pilih menu riwayat penyewaan	Pengunjung dapat melihat halaman riwayat penyewaan penyedia hiburan	Halaman riwayat penyewaan penyedia hiburan dapat dilihat oleh Pengunjung	berhasil
6	Pengunjung ingin melihat halaman detail informasi hajatan	• Halaman awal • Pilih salah satu hajatan	Pengunjung dapat melihat halaman detail informasi hajatan	Halaman detail informasi hajatan dapat dilihat oleh Pengunjung	berhasil

Tabel 4.2 Tabel Hasil Pengujian dengan *Black-Box* (Lanjutan)

No	Deskripsi	Prosedur Pengujian	Keluaran yang diharapkan	Hasil yang didapat	Kesimpulan
7	Pengunjung ingin mendaftar	<ul style="list-style-type: none"> <li>• Halaman awal</li> <li>• Pilih register</li> <li>• Isi <i>username</i></li> <li>• Isi nama lengkap</li> <li>• Isi <i>handphone</i></li> <li>• Isi email</li> <li>• Isi <i>password</i></li> <li>• Isi ulangi <i>password</i></li> <li>• Pilih pria atau wanita</li> <li>• Pilih tanggal, bulan, tahun</li> <li>• Klik daftar</li> </ul>	Pengunjung dapat melihat halaman register dan melakukan pengisian pada <i>form-form</i> secara benar	Data-data yang di <i>input-kan</i> valid dan pengunjung dapat melakukan pendaftaran <i>user</i> baru	berhasil
8	Pengunjung ingin <i>login</i>	<ul style="list-style-type: none"> <li>• Halaman awal</li> <li>• Pilih <i>login</i></li> <li>• Isi <i>username</i></li> <li>• Isi <i>password</i></li> <li>• Klik masuk</li> </ul>	Pengunjung dapat melakukan <i>login</i>	Pengunjung berhasil <i>login</i>	berhasil
<i>User Penyewa Hiburan</i>					
9	<i>User</i> ingin melihat halaman <i>user</i>	<ul style="list-style-type: none"> <li>• <i>Login</i></li> <li>• Pilih simbol <i>user</i></li> <li>• Pilih <i>user</i></li> </ul>	<i>user</i> dapat melihat halaman <i>user</i>	Halaman <i>user</i> dapat diakses oleh <i>user</i>	berhasil
10	<i>User</i> ingin melihat halaman <i>list hajatan</i>	<ul style="list-style-type: none"> <li>• Halaman <i>user</i></li> <li>• Pilih hajatan</li> </ul>	<i>user</i> dapat melihat <i>list hajatan</i>	Halaman <i>list hajatan</i> dapat diakses oleh <i>user</i>	berhasil
11	<i>User</i> ingin melihat halaman <i>list favorite</i>	<ul style="list-style-type: none"> <li>• Halaman <i>user</i></li> <li>• Pilih <i>favorite</i></li> </ul>	<i>User</i> dapat melihat <i>list favorite</i>	Halaman <i>list favorite</i> dapat diakses oleh <i>user</i>	berhasil
12	<i>User</i> ingin melihat halaman pesan	<ul style="list-style-type: none"> <li>• Halaman <i>user</i></li> <li>• Pilih pesan</li> </ul>	<i>User</i> dapat melihat pesan dan melakukan <i>chatting</i>	Halaman pesan dapat diakses oleh <i>user</i> dan <i>user</i> dapat melakukan <i>chatting</i>	berhasil

Tabel 4.3 Tabel Hasil Pengujian dengan *Black-Box* (Lanjutan)

No	Deskripsi	Prosedur Pengujian	Keluaran yang diharapkan	Hasil yang didapat	Kesimpulan
13	<i>User</i> ingin melihat halaman ulasan	<ul style="list-style-type: none"> <li>• Halaman <i>user</i></li> <li>• Pilih ulasan</li> </ul>	<i>User</i> dapat melihat halaman ulasan	<i>User</i> dapat ulasan dapat diakses oleh <i>user</i> dan <i>user</i> dapat melakukan ulasan kepada penyedia hiburan yang telah disewa	berhasil
14	<i>User</i> ingin memperbarui informasi biodata diri	<ul style="list-style-type: none"> <li>• Halaman <i>user</i></li> <li>• Pilih perbarui informasi</li> <li>• Edit <i>form</i></li> <li>• Klik tombol simpan</li> </ul>	<i>User</i> dapat mengedit dan memperbarui informasi	<i>User</i> dapat mengedit informasi dan meng-update informasi	berhasil
15	<i>User</i> ingin mengganti foto profil	<ul style="list-style-type: none"> <li>• Halaman <i>user</i></li> <li>• Pilih perbarui informasi</li> <li>• Pilih foto</li> <li>• <i>Resize</i> ukuran foto</li> <li>• Klik tombol simpan</li> </ul>	<i>User</i> dapat mengganti foto profil	Foto profil dapat diganti	berhasil
16	<i>User</i> ingin mengganti <i>password</i>	<ul style="list-style-type: none"> <li>• Halaman <i>user</i></li> <li>• Pilih perbarui informasi</li> <li>• Pilih tab ubah <i>password</i></li> <li>• Isi <i>password</i> lama</li> <li>• Isi <i>password</i> baru</li> <li>• Ulangi <i>password</i> baru</li> <li>• Klik tombol simpan</li> </ul>	<i>User</i> dapat mengganti <i>password</i> yang lama dengan <i>password</i> yang baru	<i>Password</i> dapat diganti	berhasil

Tabel 4.4 Tabel Hasil Pengujian dengan *Black-Box* (Lanjutan)

No	Deskripsi	Prosedur Pengujian	Keluaran yang diharapkan	Hasil yang didapat	Kesimpulan
17	User ingin mengganti foto <i>cover</i>	<ul style="list-style-type: none"> <li>• Halaman <i>user</i></li> <li>• Pilih perbarui informasi</li> <li>• Pilih tab ganti <i>cover</i></li> <li>• Pilih foto <i>cover</i></li> <li>• Resize ukuran <i>cover</i></li> <li>• Klik tombol simpan</li> </ul>	<i>User</i> dapat mengganti foto <i>cover</i>	Foto <i>cover</i> dapat diganti	berhasil
18	User ingin membuat hajatan	<ul style="list-style-type: none"> <li>• Halaman <i>user</i></li> <li>• Pilih buat hajatan</li> <li>• pilih tanggal</li> <li>• Masukan nama hajatan</li> <li>• Masukan lokasi hajatan</li> <li>• Masukan keterangan hajatan</li> <li>• Klik tombol buat acara</li> </ul>	<i>User</i> dapat membuat hajatan	Hajatan dapat dibuat	berhasil
19	User ingin membuka usaha	<ul style="list-style-type: none"> <li>• Halaman <i>user</i></li> <li>• Pilih buka usaha</li> <li>• Masukan nama organisasi</li> <li>• Masukan moto organisasi</li> <li>• Masukan kategori</li> <li>• Masukan harga sewa</li> <li>• Masukan lokasi</li> <li>• Masukan deskripsi</li> <li>• Klik tombol jadi penyedia</li> </ul>	<i>User</i> dapat membuka usaha dan akan berganti status menjadi <i>user</i> penyedia hiburan setelah dikonfirmasi oleh admin	status <i>user</i> penyedia hiburan akan diperoleh setelah dikonfirmasi oleh admin	berhasil

Tabel 4.5 Tabel Hasil Pengujian dengan *Black-Box* (Lanjutan)

No	Deskripsi	Prosedur Pengujian	Keluaran yang diharapkan	Hasil yang didapat	Kesimpulan
20	User ingin menyewa hiburan	<ul style="list-style-type: none"> <li>Buat hajatan</li> <li>Pilih kategori</li> <li>Pilih salah satu kategori</li> <li>Pilih salah satu penyedia hiburan</li> <li>Klik tombol sewa</li> <li>Pilih salah satu hajatan</li> <li>Klik tombol sewa</li> </ul>	<i>User/pihak penyewa dapat melakukan penyewaan hiburan untuk hajatan yang telah dibuat</i>	Penyewaan hiburan pada hajatan yang terpilih dapat dilakukan	berhasil
21	User ingin melihat pemberitahuan ulasan	<ul style="list-style-type: none"> <li><i>Login</i></li> <li>Pilih simbol lonceng</li> </ul>	<i>User dapat melihat pemberitahuan untuk mengulas penyedia hiburan</i>	Pemberitahuan ulasan dapat ditampilkan	berhasil
22	User ingin melihat pemberitahuan pesan	<ul style="list-style-type: none"> <li><i>Login</i></li> <li>Pilih simbol <i>chatting</i></li> </ul>	<i>User dapat melihat pemberitahuan pesan</i>	Pemberitahuan pesan dapat ditampilkan	berhasil
23	User ingin melihat pemberitahuan hajatan	<ul style="list-style-type: none"> <li><i>Login</i></li> <li>Pilih simbol kalender</li> </ul>	<i>User dapat melihat list hajatan yang akan dilaksanakan</i>	Pemberitahuan <i>list</i> hajatan dapat ditampilkan	berhasil
<i>User Penyedia Hiburan</i>					
24	Penyedia hiburan ingin melihat halaman penyedia hiburan	<ul style="list-style-type: none"> <li><i>Login</i></li> <li>Pilih simbol <i>user</i></li> <li>Pilih <i>user</i></li> </ul>	<i>User penyedia hiburan dapat melihat halaman penyedia hiburan</i>	Halaman penyedia hiburan dapat dilihat oleh <i>user</i> penyedia hiburan	berhasil

Tabel 4.6 Tabel Hasil Pengujian dengan *Black-Box* (Lanjutan)

No	Deskripsi	Prosedur Pengujian	Keluaran yang diharapkan	Hasil yang didapat	Kesimpulan
25	Penyedia hiburan ingin memperbarui informasi informasi hiburan	<ul style="list-style-type: none"> <li>• Halaman penyedia hiburan</li> <li>• Pilih perbarui informasi</li> <li>• Pilih <i>tab</i> penyedia</li> <li>• Edit <i>form</i></li> <li>• Klik tombol simpan</li> </ul>	<i>User</i> penyedia hiburan dapat memperbarui informasi tentang hiburannya	Informasi tentang hiburan dapat diperbarui	berhasil
26	Penyedia hiburan ingin menambahkan foto promosi	<ul style="list-style-type: none"> <li>• Halaman penyedia hiburan</li> <li>• Pilih perbarui informasi</li> <li>• Pilih <i>tab</i> foto promosi</li> <li>• <i>Drag and drop/pilih foto</i></li> </ul>	<i>User</i> penyedia hiburan dapat menambahkan foto promosi	Foto promosi dapat ditambahkan oleh penyedia hiburan	berhasil
27	Penyedia hiburan ingin menambahkan vidio promosi	<ul style="list-style-type: none"> <li>• Halaman penyedia hiburan</li> <li>• Pilih perbarui informasi</li> <li>• Pilih <i>tab</i> vidio promosi</li> <li>• <i>Copy-paste link</i> vidio youtube</li> <li>• Klik tombol simpan</li> </ul>	User penyedia hiburan dapat menambahkan vidio promosi	vidio promosi dapat ditambahkan oleh penyedia hiburan	berhasil
28	Penyedia hiburan ingin melihat halaman fans	<ul style="list-style-type: none"> <li>• Halaman penyedia hiburan</li> <li>• Pilih fans</li> </ul>	<i>User</i> penyedia hiburan dapat melihat halaman fans	Halaman fans dapat dilihat oleh penyedia hiburan	berhasil

Tabel 4.7 Tabel Hasil Pengujian dengan *Black-Box* (Lanjutan)

No	Deskripsi	Prosedur Pengujian	Keluaran yang diharapkan	Hasil yang didapat	Kesimpulan
29	Penyedia hiburan ingin melihat halaman jadwal	<ul style="list-style-type: none"> <li>• Halaman penyedia hiburan</li> <li>• Pilih jadwal</li> </ul>	<i>User</i> penyedia hiburan dapat melihat halaman jadwal	Halaman jadwal dapat dilihat oleh penyedia hiburan	berhasil
30	Penyedia hiburan ingin melihat halaman transaksi	<ul style="list-style-type: none"> <li>• Halaman penyedia hiburan</li> <li>• Pilih transaksi</li> </ul>	<i>User</i> penyedia hiburan dapat melihat halaman transaksi	Halaman transaksi dapat dilihat oleh penyedia hiburan	berhasil
31	<i>User</i> ingin melakukan <i>logout</i>	<ul style="list-style-type: none"> <li>• <i>Login</i></li> <li>• Pilih simbol user</li> <li>• Pilih <i>logout</i></li> </ul>	<i>User</i> dapat melakukan <i>logout</i>	<i>Logout</i> dapat dilakukan	berhasil
<b>Admin/Pegawai</b>					
32	Admin/pegawai ingin melakukan <i>login</i>	<ul style="list-style-type: none"> <li>• Masukan url</li> <li>• Isi <i>username</i></li> <li>• Isi <i>password</i></li> <li>• Klik <i>login</i></li> </ul>	Admin atau pegawai dapat melakukan <i>login</i>	<i>Login</i> untuk admin/pegawai dapat dilakukan	berhasil
33	Admin/pegawai ingin melihat halaman <i>dashboard</i>	<ul style="list-style-type: none"> <li>• <i>Login</i></li> <li>• Pilih <i>dashboard</i></li> </ul>	Admin atau pegawai dapat melihat halaman <i>dashboard</i>	Halaman <i>dashboard</i> dapat dilihat oleh admin/pegawai	berhasil
34	Admin ingin melihat, menambah, menghapus data pegawai	<ul style="list-style-type: none"> <li>• <i>Login</i></li> <li>• Pilih data pegawai</li> <li>• Isi <i>username</i></li> <li>• Isi nama</li> <li>• Isi <i>password</i></li> <li>• Klik simpan</li> <li>• Pilih hapus</li> <li>• Klik tombol oke</li> </ul>	Admin dapat melihat, menambah, dan menghapus data pegawai	Melihat, menambah, dan menghapus data pegawai dapat dilakukan oleh admin	berhasil

Tabel 4.8 Tabel Hasil Pengujian dengan *Black-Box* (Lanjutan)

No	Deskripsi	Prosedur Pengujian	Keluaran yang diharapkan	Hasil yang didapat	Kesimpulan
35	Admin/pegawai ingin melihat data pengguna aplikasi	<ul style="list-style-type: none"> <li>• <i>Login</i></li> <li>• Pilih data pengguna</li> </ul>	Admin/pegawai dapat melihat data pengguna aplikasi	Halaman data pengguna aplikasi dapat dilihat oleh admin/pegawai	berhasil
36	Admin/pegawai ingin mengkonfirmasi permintaan penyedia baru	<ul style="list-style-type: none"> <li>• <i>Login</i></li> <li>• Pilih data pengguna</li> <li>• Pilih salah satu permintaan</li> <li>• Klik tombol konfirmasi</li> </ul>	Admin/pegawai dapat mengkonfirmasi permintaan penyedia baru	Konfirmasi penyedia baru dapat dilakukan oleh admin/pegawai	berhasil
37	Admin/pegawai ingin melihat data penyedia hiburan	<ul style="list-style-type: none"> <li>• <i>Login</i></li> <li>• Pilih data penyedia</li> </ul>	Admin/pegawai dapat melihat data penyedia hiburan	Halaman data penyedia hiburan dapat dilihat oleh admin/pegawai	berhasil
38	Admin/pegawai ingin <i>logout</i>	<ul style="list-style-type: none"> <li>• <i>Login</i></li> <li>• Pilih <i>logout</i></li> </ul>	Admin/pegawai dapat melakukan <i>logout</i>	<i>Logout</i> dapat dilakukan	berhasil

#### 4.2.3 Pengujian Melalui Kuisioner

Penulis melakukan pengujian aplikasi ini dengan kuisioner yang dilakukan kepada 15 orang penguji/responden. Rekapitulasi hasil pengujian melalui kuisioner adalah sebagai berikut.

## 1. Pertanyaan Kuisioner

Tabel 4.9 Tabel Pertanyaan Kuisioner

Tampilan Aplikasi	
No	Pertanyaan
1	Bagaimana pendapat anda mengenai tampilan desain aplikasi ?
2	Bagaimana pendapat anda mengenai tampilan halaman awal ?
3	Bagaimana pendapat anda mengenai tampilan detail hajatan ?
4	Bagaimana pendapat anda mengenai tampilan detail penyedia hiburan ?
5	Bagaimana pendapat anda mengenai tampilan detail kategori ?
6	Bagaimana pendapat anda mengenai tampilan halaman <i>register</i> ?
7	Bagaimana pendapat anda mengenai tampilan halaman <i>login</i> ?
8	Bagaimana pendapat anda mengenai tampilan halaman user ?
Penggunaan Aplikasi	
No	Pertanyaan
1	Bagaimana pendapat anda, apakah aplikasi ini cukup mudah digunakan ?
2	Apakah membuat acara hajatan di dalam sistem ini mudah anda lakukan ?
3	Apakah membuka usaha di dalam sistem ini mudah anda lakukan ?
4	Apakah melakukan penyewaan hiburan mudah anda lakukan ?
Kegunaan Aplikasi	
No	Pertanyaan
1	Apakah fitur <i>chatting</i> sudah sangat membantu komunikasi antar <i>user</i> dalam aplikasi ?
2	Apakah aplikasi ini sangat membantu penyewa untuk melakukan pemesanan acara hiburan ?
3	Apakah aplikasi ini sangat membantu penyewa untuk mengetahui informasi acara hiburan ?
4	Apakah aplikasi ini sangat bermanfaat bagi penyedia hiburan untuk mempromosikan hiburannya ?
Kualitas Aplikasi	
No	Pertanyaan
1	Apakah semua fungsi di halaman awal berjalan dengan baik ?
2	Apakah pada halaman <i>register</i> sudah dapat memvalidasi <i>input</i> data ?
3	Apakah semua fungsi di halaman user berjalan dengan baik ?
4	Bagaimana pendapat anda tentang keseluruhan aplikasi ini ?

## 2. Hasil Jawaban Kuisioner

Tabel 4.10 Hasil Jawaban Kuisioner

No	Nama	Jawaban																			
		Tampilan Aplikasi								Penggunaan Aplikasi				Kegunaan Aplikasi				Kualitas Aplikasi			
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	Aep S H	B	B	B	B	B	A	A	B	B	B	A	B	A	A	A	A	A	A	A	B
2	Ahmad Tajul A	A	A	B	B	B	A	B	A	B	A	B	B	A	B	B	A	A	A	A	B
3	Ardhi Irfan S	B	A	B	B	B	A	B	A	B	B	B	B	A	A	A	A	B	A	B	B
4	Darsiwani	B	B	A	B	A	A	A	A	A	A	B	A	A	A	A	A	B	A	A	A
5	Dodi Septo W	B	A	B	B	A	A	A	A	B	B	A	C	A	B	A	B	A	A	B	A
6	Firman S R	A	A	A	A	B	A	A	A	B	B	A	A	B	A	A	B	A	B	B	A
7	Irvan Kharisma F	B	B	B	B	C	A	B	B	B	B	C	B	B	A	B	B	A	C	B	B
8	Lailiyah	A	A	A	B	B	A	A	A	B	A	A	A	A	A	B	A	B	A	B	B
9	Nurian Habimunif	A	B	B	A	B	A	A	A	B	B	B	A	A	B	B	A	A	A	B	B
10	Riza M F	B	B	A	A	B	B	A	A	A	A	A	B	B	A	A	B	B	A	A	A
11	Robbi Zamzamil A	A	B	B	B	A	B	A	A	B	C	B	C	B	A	B	B	B	B	B	A
12	Rudi Tawangga	B	A	B	B	B	B	B	B	B	B	B	B	B	B	A	A	B	A	B	B
13	Tantowi Y Y T	A	A	A	A	A	A	A	B	A	B	B	A	A	A	B	B	A	B	B	A
14	Wiwit A S	A	A	B	B	B	B	B	A	B	B	B	A	A	A	A	A	B	A	B	B
15	Yusup A A	B	B	C	C	B	B	B	B	C	A	A	B	B	B	C	B	B	A	B	B

### 3. Keterangan Nilai

Tabel 4.11 Keterangan Nilai

Tampilan Aplikasi	
A	Amat Baik (Warna, Layout, Teks, Navigasi, Gambar)
B	Baik (Layout, Teks, Navigasi)
C	Cukup (Layout, Teks)
D	Kurang (Teks)
Penggunaan Aplikasi	
A	Amat Baik ( <i>User guide</i> )
B	Baik (Tahapan jelas)
C	Cukup (Perintah jelas)
D	Kurang (Perintah ambigu)
Kegunaan Aplikasi	
A	Amat Baik (Dapat bertransaksi)
B	Baik (Dapat berkomunikasi)
C	Cukup (Mendapatkan informasi)
D	Kurang (Tidak membantu)
Kualitas Aplikasi	
A	Amat Baik (Data valid, output valid, akses cepat, modul-modul berfungsi)
B	Baik (Data valid, output valid)
C	Cukup (Pesan peringatan, akses lama)
D	Kurang (Error)

### 4. Hasil Perhitungan

#### a. Tampilan Aplikasi

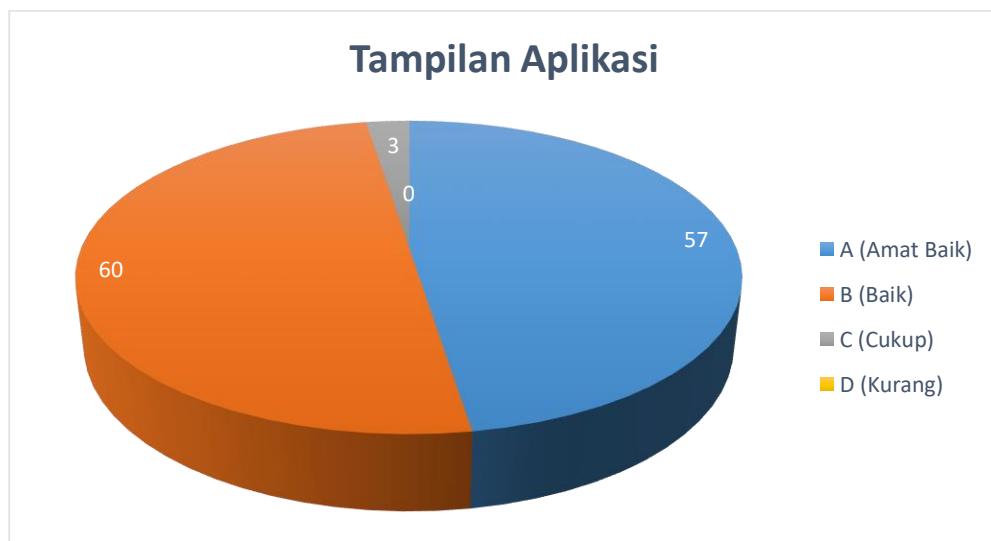
Tabel 4.12 Hasil Perhitungan Tampilan Aplikasi

No	Pertanyaan	Amat Baik	Baik	Cukup	Kurang
1	Bagaimana pendapat anda mengenai tampilan desain aplikasi ?	7	8	0	0
2	Bagaimana pendapat anda mengenai tampilan halaman awal ?	8	7	0	0
3	Bagaimana pendapat anda mengenai tampilan detail hajatan ?	5	9	1	0
4	Bagaimana pendapat anda mengenai tampilan detail penyedia hiburan ?	4	10	1	0

Tabel 4.13 Hasil Perhitungan Tampilan Aplikasi (Lanjutan)

No	Pertanyaan	Amat Baik	Baik	Cukup	Kurang
5	Bagaimana pendapat anda mengenai tampilan detail kategori ?	4	10	1	0
6	Bagaimana pendapat anda mengenai tampilan halaman <i>register</i> ?	10	5	0	0
7	Bagaimana pendapat anda mengenai tampilan halaman <i>login</i> ?	9	6	0	0
8	Bagaimana pendapat anda mengenai tampilan halaman user ?	10	5	0	0
Total		57	60	3	0

Keterangan menggunakan *Pie Chart*.



Gambar 4.38 *Pie Chart* Tampilan Aplikasi

### b. Penggunaan Aplikasi

Tabel 4.14 Hasil Perhitungan Penggunaan Aplikasi

No	Pertanyaan	Amat Baik	Baik	Cukup	Kurang
1	Bagaimana pendapat anda, apakah aplikasi ini cukup mudah digunakan ?	3	11	1	0
2	Apakah membuat acara hajatan didalam sistem ini mudah anda lakukan ?	5	9	1	0

Tabel 4.15 Hasil Perhitungan Penggunaan Aplikasi (Lanjutan)

3	Apakah membuka usaha didalam sistem ini mudah anda lakukan ?	6	8	1	0
4	Apakah melakukan penyewaan hiburan mudah anda lakukan ?	6	7	2	0
	Total	20	35	5	0

Keterangan menggunakan *Pie Chart*.



Gambar 4.39 *Pie Chart* Penggunaan Aplikasi

### c. Kegunaan Aplikasi

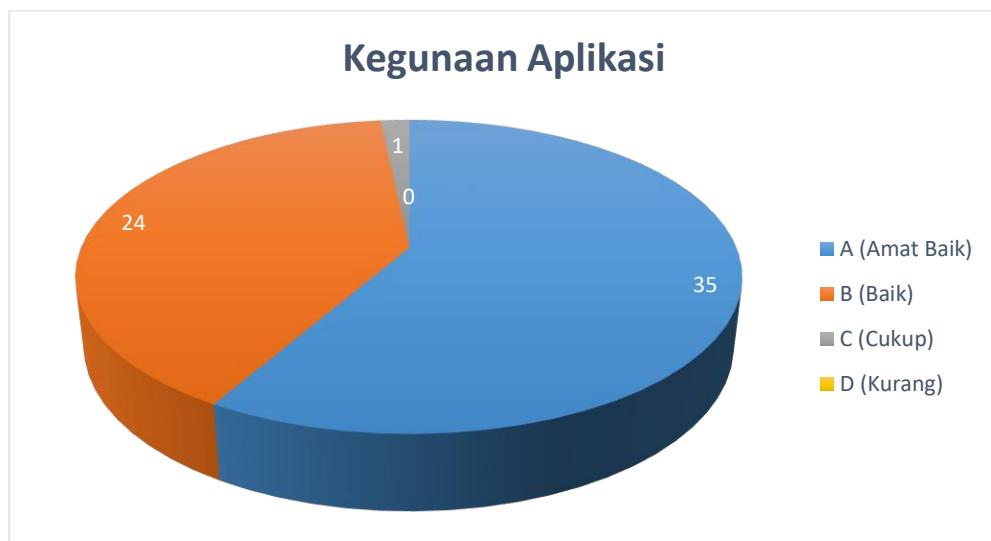
Tabel 4.16 Hasil Perhitungan Kegunaan Aplikasi

No	Pertanyaan	Amat Baik	Baik	Cukup	Kurang
1	Apakah fitur <i>chatting</i> sudah sangat membantu komunikasi antar user dalam aplikasi ?	9	6	0	0
2	Apakah aplikasi ini sangat membantu penyewa untuk melakukan pemesanan acara hiburan ?	10	5	0	0
3	Apakah aplikasi ini sangat membantu penyewa untuk mengetahui informasi acara hiburan ?	8	6	1	0

Tabel 4.17 Hasil Perhitungan Kegunaan Aplikasi (Lanjutan)

4	Apakah aplikasi ini sangat bermanfaat bagi penyedia hiburan untuk mempromosikan hiburannya ?	8	7	0	0
	Total	35	24	1	0

Keterangan menggunakan *Pie Chart*.



Gambar 4.40 *Pie Chart* Kegunaan Aplikasi

#### d. Kualitas Aplikasi

Tabel 4.18 Hasil Perhitungan Kualitas Aplikasi

No	Pertanyaan	Amat Baik	Baik	Cukup	Kurang
1	Apakah semua fungsi di halaman awal berjalan dengan baik ?	7	8	0	0
2	Apakah pada halaman <i>register</i> sudah dapat memvalidasi <i>input</i> data ?	10	4	1	0
3	Apakah semua fungsi di halaman user berjalan dengan baik ?	4	11	0	0
4	Bagaimana pendapat anda tentang keseluruhan aplikasi ini ?	6	9	0	0
	Total	27	32	1	0

Keterangan menggunakan *Pie Chart*.



Gambar 4.41 *Pie Chart* Kualitas Aplikasi

## 5. Rekapitulasi Hasil Kuisioner

Tabel 4.19 Rekapitulasi Hasil Kuisioner

No	Parameter	Amat Baik	Baik	Cukup	Kurang
1	Tampilan Aplikasi	57	60	3	0
2	Penggunaan Aplikasi	20	35	5	0
3	Kegunaan Aplikasi	35	24	1	0
4	Kualitas Aplikasi	27	32	1	0
Total		139	151	10	0

Dari hasil rekapitulasi kuisioner tersebut dapat disimpulkan dengan persamaan 1 menggunakan ketentuan skala sebagai berikut.

- a. Amat Baik = 4
- b. Baik = 3
- c. Cukup = 2
- d. Kurang = 1

Dimana skala nilai maksimal adalah 4 yaitu amat baik.

$$Hasil Pengujian = \sum_{i=1}^{i=4} Hi * 100\% \quad (1)$$

Dimana  $Hi$  didapatkan dari persamaan 2 berikut.

$$Hi = \frac{\sum_{j=1}^n x_j}{Np} \quad (2)$$

Keterangan :

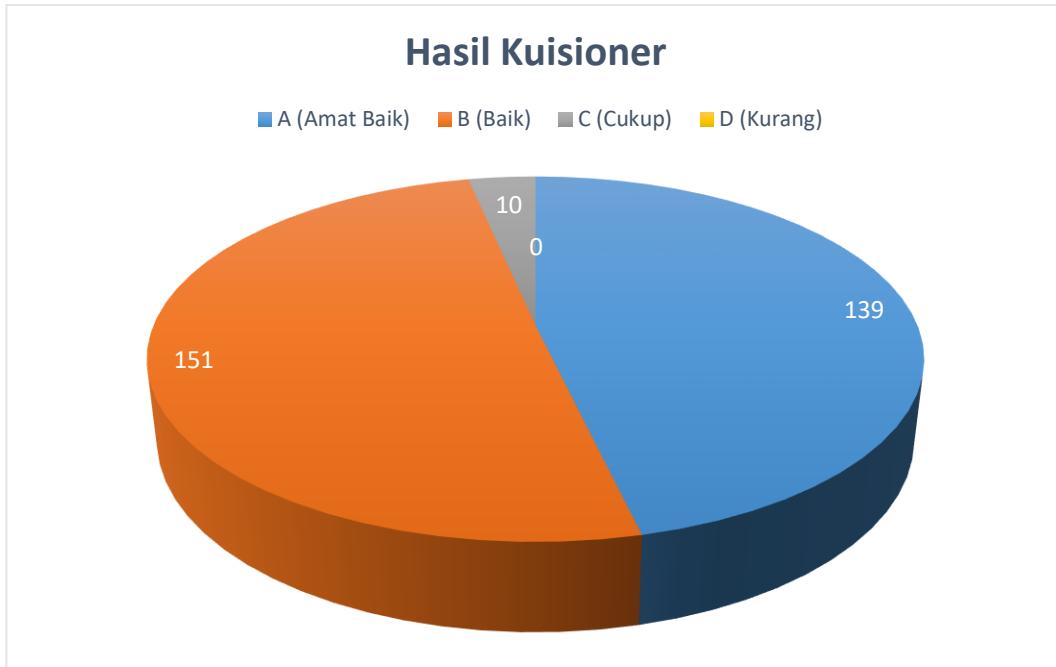
- $j$  = Index soal ke  $j$
- $x_j$  = Nilai/skor pertanyaan ke  $j$  untuk index  $i$
- $n$  = Maksimal soal = 20
- $Np$  = Total poin = 1200, didapatkan dari persamaan 3.

$$Np = Jumlah soal \times 4 \text{ (skala maksimal)} \times Jumlah responden \quad (3)$$

Dengan menggunakan persamaan 1 maka akan dapat dihitung hasil pengujian sebagai berikut.

Kurang	= $(0*1/1200)*100\%$	= 0 %
Cukup	= $(10*2/1200)*100\%$	= 1,67%
Baik	= $(151*3/1200)*100\%$	= 37,75%
Amat Baik	= $(139*4/1200)*100\%$	= 46,33% +
<hr/>		
<b>Hasil</b>		<b>= 85,75 %</b>

Dari hasil pengujian kuisioner diperoleh persentase pengujian sebesar 85,75% dan dapat disimpulkan bahwa aplikasi ini amat baik. perolehan dalam bentuk grafik dapat dilihat pada Gambar 4.42 dan untuk data-data kuisioner dapat dilihat pada lampiran 2.



Gambar 4.42 Hasil Keseluruhan Pengujian Melalui Kuisioner

#### 4.2.4 Kesimpulan Pengujian

Berdasarkan hasil *testing* yang dilakukan oleh *user* bahwa aplikasi yang dibuat penulis sudah berjalan dengan baik. Pada dasarnya hasil pengujian aplikasi sesuai harapan, program berjalan dengan baik, dan membantu penyedia hiburan untuk mempromosikan/memberitahukan ke publik tentang hiburannya, dan juga membantu orang-orang yang akan hajatan untuk melakukan penyewaan hiburan. Perolehan hasil presentase pengujian dengan menggunakan kuisioner yang dilakukan oleh 15 orang penguji/responden dengan menguji dari sisi tampilan aplikasi, penggunaan aplikasi, kegunaan aplikasi, dan kualitas aplikasi hasilnya adalah 85,75%, dengan presentase tersebut dapat dikatakan aplikasi ini sudah amat baik digunakan oleh *user*.

#### **4.2.5 Kelebihan dan Kekurangan Aplikasi E-Market Hiburan untuk Kebutuhan Hajatan**

Dalam pembuatan suatu aplikasi pasti terdapat kelebihan dan kekurangannya. Berikut penulis menjelaskan kelebihan dan kekurangan aplikasi ini sebagai bahan pertimbangan jika suatu saat ada yang berminat untuk mengembangkannya lebih lanjut.

##### **1. Kelebihan Aplikasi E-Market Hiburan untuk Kebutuhan Hajatan**

- a. Register pengguna baru sudah dapat tervalidasi.
- b. Pengguna aplikasi sudah dapat melakukan pembuatan hajatan.
- c. Pengguna aplikasi sudah dapat membuka usaha (berganti status menjadi *user* penyedia hiburan).
- d. Penyewa hiburan dengan penyedia hiburan sudah dapat saling berkomunikasi menggunakan fitur *chatting real-time*, sehingga dengan adanya fitur ini antara pihak penyewa dengan pihak penyedia dapat langsung berkomunikasi menentukan kesepakatan.
- e. Jadwal penyewaan sudah dapat ditampilkan dalam bentuk kalender yang tertandai. Dengan fitur ini pihak yang akan menyewa hiburan dapat mengetahui secara langsung jadwal kegiatan penyedia hiburan, sehingga ketika jadwal penyedia hiburan pada tanggal hajatan penyewa sudah tertandai (tersewa) maka pihak penyewa tidak dapat melakukan penyewaan pada penyedia hiburan tersebut.
- f. Penyedia hiburan dapat menambahkan jadwal penyewaan di luar sistem.
- g. Penyewa hiburan sudah dapat melakukan ulasan ke hiburan yang telah disewa. Dengan fitur ulasan ini diharapkan dapat menjadi bahan pertimbangan untuk pihak penyewa yang akan menyewa hiburan pada penyedia hiburan yang dipilih.
- h. Hasil ulasan dapat ditampilkan dalam bentuk *ratting* popularitas. Dengan fitur *ratting* ini diharapkan dapat menjadi bahan pertimbangan untuk pihak penyewa yang akan menyewa hiburan pada penyedia hiburan yang dipilih.
- i. Lokasi hajatan ditampilkan dalam bentuk map. Dengan fitur ini diharapkan untuk tamu-tamu atau pengunjung yang belum mengetahui tempat/lokasi hajatan dapat mengetahui dan melihat lokasi hajatan melalui fitur ini.

- j. Tanggal terakhir perubahan harga sudah dapat ditampilkan. Dengan fitur ini pihak yang akan menyewa hiburan dapat melihat kapan perubahan harga terakhir dari penyedia hiburan yang dipilih.
- k. Mengganti foto profil atau foto cover sudah dilengkapi dengan *cropping*.
- l. Aplikasi sudah *responsive* di *smartphone* dan *desktop*.

## 2. Kekurangan Aplikasi E-Market Hiburan untuk Kebutuhan Hajatan

- a. Notifikasi *chatting* belum dapat menampilkan notifikasi *chatting* yang terbaru.
- b. Transaksi masih dilakukan secara manual (pertemuan antara pihak penyewa dengan pihak penyedia).
- c. Informasi *user*/penyedia hiburan yang sedang *online* masih belum terimplementasi.
- d. *Cropping* foto cover masih *low resolution*.

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Kesimpulan yang didapat setelah mengumpulkan, mengolah, menganalisa, mengimplementasi serta melakukan pengujian aplikasi adalah sebagai berikut.

1. Aplikasi ini dirancang dengan cara mengumpulkan data-data melalui metode observasi dan studi pustaka, kemudian setelah data-data terkumpul dilakukan proses desain sistem.
2. Aplikasi ini dibuat dalam bentuk *responsive* desain sehingga dapat diakses dengan tampilan yang baik dengan menggunakan *smartphone* ataupun komputer/PC (*Desktop*).
3. Proses transaksi diterapkan dengan melibatkan dua *user* yaitu pihak penyewa dan pihak penyedia hiburan yang dimana *user* penyewa sebelum melakukan penyewaan hiburan harus membuat suatu hajatan.
4. Pengujian aplikasi dilakukan dengan menggunakan *Black-Box Testing* yang telah diuji oleh *user* melalui kuisioner dan hasil dari pengujian yang didapatkan melalui kuisioner tersebut menyatakan aplikasi e-market hiburan untuk kebutuhan ini amat baik dengan presentase nilai 85,75%.

#### **5.2 Saran**

Aplikasi e-market hiburan untuk kebutuhan hajatan ini tidak luput dari kekurangan. Oleh karena itu penulis menyarankan kepada pengembang selanjutnya agar dapat mempertimbangkan beberapa saran yang diberikan supaya aplikasi e-market hiburan untuk kebutuhan hajatan ini semakin lengkap dan baik. Saran-saran tersebut antara lain sebagai berikut.

1. Pengumpulan data-data untuk tahap perancangan aplikasi dapat ditingkatkan dengan cara melakukan wawancara langsung dan terstruktur kepada pihak penyedia hiburan supaya data yang didapatkan lebih *valid*.
2. Dapat dibuat dalam bentuk *mobile application* sehingga akan lebih memudahkan akses bagi pengguna aplikasi.

3. Proses transaksi dapat ditingkatkan menjadi transaksi yang terkomputerisasi dari mulai melakukan pemesan hiburan sampai transaksi transfer uang. Sehingga semua transaksi dapat terkontrol oleh sistem.
4. Dapat ditambahkan pengujian menggunakan *White-Box Testing* agar sistem dapat lebih baik lagi dan optimal untuk diakses.

## DAFTAR PUSTAKA

- Alfarabi Hadi, Diki. 2016. *Cara Penulisan Dan Penggunaan Javascript*. <http://www.malasngoding.com>. Diakses pada tanggal 7 Agustus 2017.
- Ariani Sukamto, Rosa. 2009. *Black-Box Testing*. <http://julian.unsri.ac.id>. Diakses pada tanggal 7 Agustus 2017.
- Awwali, Muhammad Ryza dan Sulartopo. 2014. *Media Pembelajaran Desain Grafis Di Sma 1 Kudus Berbasis Multimedia Interaktif*. Semarang : STEKOM.
- Ayu Langening Tyas, Dyah dan Didik Dwi Prasetya. 2016. *Sistem Informasi Geografis Lokasi Masjid Berbasis Web di Kota Malang*. Vol. 8.
- Dwiputra Chondro, Rinaldi. 2014. *E-Marketing dan perbedaannya*. <http://e-business-petra.blogspot.co.id>. Diakses pada tanggal 9 Agustus 2017.
- Desiana, Nur. 2013. *Microsoft Visio*. <http://nurdesiana22.blogspot.co.id>. Diakses pada tanggal 7 Agustus 2017.
- Dwi, Hendraswati Enggar. 2016. *Balsamiq Mockup*. <http://enggar.net>. Diakses pada tanggal 7 Agustus 2017.
- Dwiputra, Rinaldi. 2014. *E-Marketing dan Perbedaannya*. <http://e-business-petra.blogspot.co.id>. Diakses pada tanggal 7 Agustus 2017.
- Eywa. 2015. *Pengertian JQuery Dan Menurut Para Ahli*. <http://bedahrumahnew.blogspot.co.id>. Dikases pada tanggal 7 Agustus 2017.
- Firmansyah, Wahyu. 2014. *Pengertian dan Contoh DDL, DML dan DCL*. <http://gudang-ilmu2an.blogspot.co.id>. Diakses pada tanggal 6 Agustus 2017.
- Fitzgerald, Michael. 2012. *Introducing Regular Expression*. 1005 Gravenstein Highway North : O'Reilly Media, Inc.
- Herwanto. 2016. *Sistem pakar diagnosis gangguan kehamilan Dengan metode forward chaining*. Indramayu : Politeknik Negeri Indramayu.
- Hidayat, Arief, dkk. 2016. *Penerapan Responsive Web Design dalam Perancangan Sistem Modul Online Adaptif*. Vol. 12. No. 1.
- Iansyah. 2016. *Apa itu UML Use Case Diagram part 2*. <http://www.iansyahbelajar.com>. Diakses pada tanggal 9 Agustus 2017.
- Kamil, Ratnasari. 2014. *Pengertian UML Unified Modeling Language*. <http://tips-triks-ilmu-komputer.blogspot.co.id>. Diakses pada tanggal 5 Mei 2017.
- Latief, Mukhlisulfatih. 2012. *Pendekatan Database untuk Manajemen Data dalam Meningkatkan Kemampuan Mahasiswa Mengaplikasikan Konsep Basisdata*. Peran LPTK Dalam Pengembangan Pendidikan Vokasi di Indonesia. Vol. 1, No. 234.
- Mujilan, Agustinus. 2013. *Analisis dan Perancangan Sistem*. Madiun : Universitas Widya Mandala Madiun.
- Morphine. 2012. *Mengenal Regular Expression*. <http://belajarwebdesign.com>. Diakses pada tanggal 7 Agustus 2017.
- Muliantara, Agus. 2014. *Penerapan Regular Expression dalam Melindungi Alamat Email dari Spam Robot Pada Konten Wordpress*. Universitas Udayana : Kuta Selatan.
- .

- Muslihudin, Muhamad dan Oktafianto. 2016. *Analisis dan Perancangan Sistem Informasi Menggunakan Model Terstruktur dan UML*. Yogyakarta : ANDI
- Nanda, Arbia. 2015. *Dreamweaver CS5*. <http://ehnanda.blogspot.co.id>. Diakses pada tanggal 7 Agustus 2017.
- Onphpid. 2015. *Belajar Membuat Template Web Menggunakan Bootstrap*. <https://onphpid.com>. Diakses pada tanggal 7 Agustus 2017.
- Piyaneo. 2014. *Javascript*. <https://piyaneo.wordpress.com>. Diakses pada tanggal 7 Agustus 2017.
- Pranowo, Galih. 2010. *Mastering Corel Draw X4*. Yogyakarta : ANDI.
- Pranata, Dana, dkk. 2015. *Rancang Bangun Website Jurnal Ilmiah Bidang Komputer (Studi Kasus : Program Studi Ilmu Komputer Universitas Mulawarman)*. Jurnal Informatika Mulawarman. Vol. 10, No. 2.
- Pratiwi Masero, Agnesie, Joko Triyono, & Dina Andayati. 2013. *Perancangan Pengelolaan Jaringan IT Pada Institut Sains & Teknologi Akprind Menggunakan Teknologi VPN (Virtual Private Network)*. Jurnal Jarkom. Vol. 1, No. 1.
- Rahardi, Wisa. 2014. *Belajar HTML Mengenal Tag, Elemen, Atribut*. <http://blogooblok.com>. Diakses pada tanggal 6 Agustus 2017.
- Rifa'i, Saifulloh. 2013. *Pengertian dan Sejarah MySQL*. <https://upyes.wordpress.com>. Diakses pada tanggal 6 Agustus 2017.
- Ritonga, Pahmi. 2015. *Pengertian Bahasa Pemrograman PHP Menurut Para Pakar*. <http://bangpahmi.com>. Diakses pada tanggal 6 Agustus 2017
- Ritonga, Pahmi. 2015. *Pengertian MySQL Menurut Para Pakar*. <http://bangpahmi.com>. Diakses pada tanggal 6 Agustus 2017.
- Saleh, Maulinda. 2016. *System Development Life Cycle (SDLC)*. <http://maulindaugsalemba.blogspot.co.id>. Diakses pada tanggal 28 Agustus 2017.
- Saputra, Mario. 2013. *Perancangan Sistem Basis Data Akademik Pada SMP Negeri 26 Palembang*. Palembang : Universitas Binadarma.
- Shannon, Ross. 2012. *The History of HTML*. <http://yourhtmlsource.com>. Diakses pada tanggal 5 Mei 2017.
- Sinaga, Juniar dan Ahmad Nouvel. 2015. *Perancangan Aplikasi Berbasis Android untuk Menentukan Resistansi Resistor Berdasarkan Gelang Warna*. Evolusi. Vol.3, No.1.
- Suehring, Steven dan Janet Valade. 2013. *PHP, MySQL, Javascript & HTML5 all-in-one for Dummies*. John Wiley & Sons Inc. Hoboken. New Jersey.
- Susanto, Deni. 2015. *Perancangan Sistem Informasi Laporan Penjualan Pada Pt Kinnara Mitra Selaras*. Tanggerang : STMIK Raharja.
- Syakirurahman. 2013. *Mengenal Apa Itu CSS Dan Bagaimana Menggunakannya*. <http://devaradise.com>. Diakses pada tanggal 6 Agustus 2017.
- Tandusan, Chevy F. 2015. *Jurnal Riset Bisnis dan Manajemen*. Vol.3, No.2.
- Widyantoro, Wahyu. *Buku Panduan Bootstrap*. Margotekno.
- Willyanto Santoso, Leo. 2013. *Pelatihan Microsoft Visio 2010 Profesional*. <http://puskom.petra.ac.id>. Diakses pada tanggal 7 Agustus 2017.

# **LAMPIRAN**