

APLIKASI BERBASIS WEB *MARKETPLACE EVENT*
ORGANIZER EVORIA



Disusun Oleh:

N a m a : Aditya Mahavira
NIM : 16523110

PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA

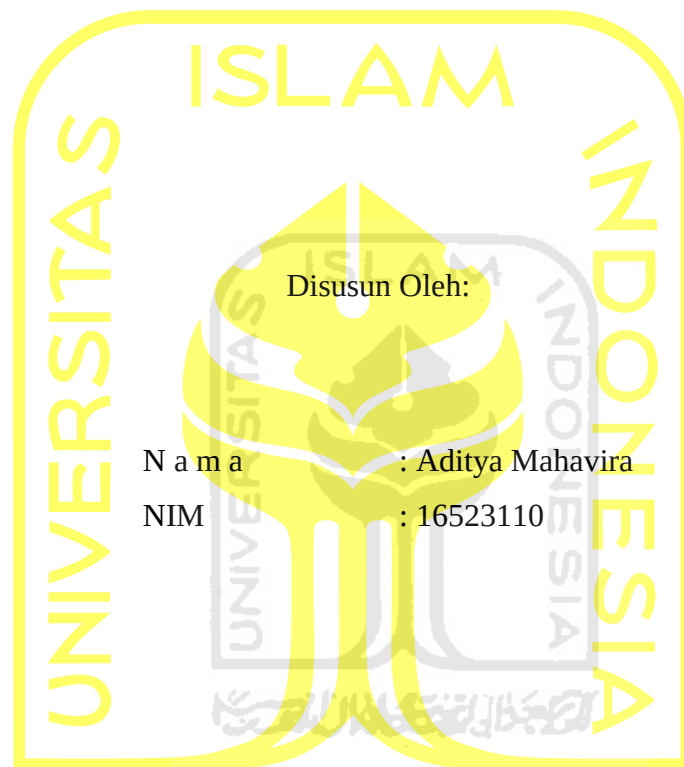
2020

HALAMAN PENGESAHAN DOSEN PEMBIMBING

APLIKASI BERBASIS WEB *MARKETPLACE EVENT*

ORGANIZER EVORIA

TUGAS AKHIR



N a m a : Aditya Mahavira

NIM : 16523110

الجامعة الإسلامية
Yogyakarta, 12 November 2020

Pembimbing,

(Irving Vitra Paputungan, S.T., M.Sc., Ph.D.)

HALAMAN PENGESAHAN DOSEN PENGUJI

APLIKASI BERBASIS WEB *MARKETPLACE EVENT* *ORGANIZER EVORIA*

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 12 November 2020

Tim Penguji

Irving Vitra Paputungan, S.T., M.Sc.,
Ph.D.

Anggota 1

Dr. Raden Teduh Dirgahayu, S.T., M.Sc.

Anggota 2

Hari Setiaji, S.Kom., M.Eng.

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia



(Dr. Raden Teduh Dirgahayu, S.T., M.Sc.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Aditya Mahavira

NIM : 16523110

Tugas akhir dengan judul:

APLIKASI BERBASIS WEB *MARKETPLACE EVENT* *ORGANIZER EVORIA*

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila dikemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung resiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 12 November 2020



(Aditya Mahavira)

HALAMAN PERSEMBAHAN

Allhamdulillahirobbil'alamin, puja dan puji syukur kehadiran Allah *Subhanahu wata'ala* atas segala nikmat yang diberikan kepada kita semua. Shalawat serta salam kepada junjungan kita Nabi Muhammad *Sallallahu 'alaihi wasallam* yang telah membimbing kita menuju Islam yang *rahmatan lil 'alamin*, dan yang kita nantikan syafa'atnya kelak di hari akhir.

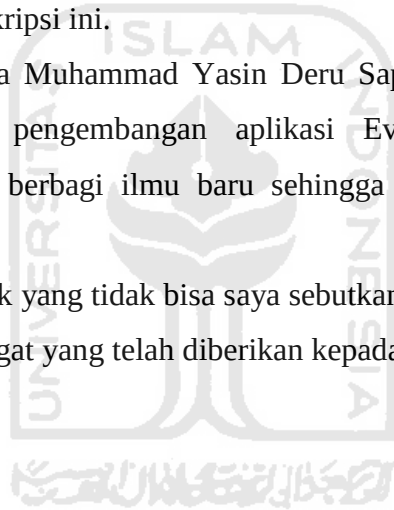
Terimakasih yang tidak terhingga untuk kedua orangtua saya, yang telah mengasuh, mendidik, membiayai hidup, dan mendoakan saya dengan tulus hingga saat ini.

Terimakasih untuk bapak Irving Vitra Paputungan, ST., M.Sc. yang telah membimbing, mengajar, dan memberikan saya motivasi dalam pembuatan laporan skripsi ini.

Terimakasih untuk teman-teman satu tim saya yaitu Muhammad Rasyid Shadiq dan Mutia Mernisiola Zipa yang mau bekerja sama, saling memotivasi, saling menyemangati sehingga saya bisa menyelesaikan laporan skripsi ini.

Terimakasih untuk teman saya Muhammad Yasin Deru Saputra S. Kom. Yang sudah membantu saya dalam proses pengembangan aplikasi Evoria, mengatasi berbagai permasalahan pemrograman, dan berbagi ilmu baru sehingga saya dapat menyelesaikan aplikasi serta laporan skripsi saya.

Terimakasih untuk semua pihak yang tidak bisa saya sebutkan satu-satu, atas segala do'a, ilmu, bantuan, motivasi dan semangat yang telah diberikan kepada saya.



HALAMAN MOTO

“Manfaatkan apa yang kita punya dengan maksimal”

“Pikirkan resiko yang akan didapat sebelum bertindak”



KATA PENGANTAR

Allhamdulillahirobbil'alamin, puja dan puji syukur kehadiran Allah *Subhanahu wata'ala* atas segala nikmat yang diberikan kepada kita semua. Shalawat serta salam kepada junjungan kita Nabi Muhammad *Sallallahu 'alaihi wasallam* yang telah membimbing kita menuju Islam yang *rahmatan lil 'alamin*, dan yang kita nantikan syafa'atnya kelak di hari akhir.

Laporan tugas akhir ini disusun guna memenuhi salah satu syarat menyelesaikan studi serta dalam rangka memperoleh gelar Sarjana Komputer Strata Satu pada Program Studi Informatika Fakultas Teknologi Industri Universitas Islam Indonesia.

Oleh karena itu, penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada semua pihak-pihak yang disebutkan di bawah ini :

1. Kedua orangtua saya, yang telah mengasuh, mendidik, membiayai hidup, dan mendoakan saya dengan tulus hingga saat ini.
2. Dr. Raden Teduh Dirgahayu, S.T., M.Sc. selaku Kaprodi Informatika UII.
3. Irving Vitra Paputungan, ST., M.Sc. selaku Dosen pembimbing.
4. Teman-teman satu tim saya yaitu Muhammad Rasyid Shadiq dan Mutia Mernisiola Zipa yang mau bekerja sama, saling memotivasi, saling menyemangati.
5. Teman-teman Jurusan Informatika UII angkatan 2016.
6. Semua pihak yang telah memberikan semangat dan do'a kepada penulis.

Semoga penulisan laporan tugas akhir ini bisa bermanfaat bagi semua orang serta berguna bagi nusa dan bangsa.

Yogyakarta, 12 November 2020



(Aditya Mahavira)

SARI

Jumlah Event Organizer (EO) yang relatif banyak sekarang ini membuat adanya keperluan akan sebuah platform yang dapat mewadahi semua EO tersebut. Terutama untuk EO yang sudah memiliki portofolio namun belum diketahui banyak orang atau baru berdiri. Evoria adalah aplikasi *marketplace event organizer* berbasis web yang mampu menyediakan tempat untuk memasarkan jasa dan portofolio para EO agar dapat dilihat dan diketahui banyak orang. Aplikasi Evoria dapat menghubungkan EO dengan *customer* yang ingin mengadakan suatu acara sehingga memudahkan proses pemesanan jasa EO.

Laporan ini memilih fokus pada bagian pengembangan aplikasi. Proses pengembangan dilakukan menggunakan metode *Rapid Application Development* (RAD). *Rapid Application Development* (RAD) merupakan metode pengembangan sistem informasi dengan waktu singkat, sehingga dinilai tepat digunakan dalam pengembangan Evoria. Pemodelan dan perancangan basis data Evoria menggunakan *Entity Relationship Diagram* (ERD).

Pengembangan Evoria berbasis *website* terbukti dapat digunakan dengan mudah serta menjangkau pengguna yang lebih luas, sekaligus turut serta dalam perkembangan *startup* di Indonesia.

Kata kunci: *event organizer*, *evoria*, *marketplace*, pengembangan aplikasi, *rapid application development*.

GLOSARIUM

<i>Feedback</i>	memberi masukan kembali terhadap suatu hal yang ditunjukkan dan mendorong kepada perbaikan.
<i>Back-End</i>	posisi pengembang aplikasi yang bertugas dan bertanggung jawab pada operasi pengembangan di belakang layar, tidak diketahui langsung oleh pengguna.
<i>Front-End</i>	posisi pengembang aplikasi yang bertugas dan bertanggung jawab pada operasi pengembangan di tampilan antarmuka aplikasi, dapat diketahui langsung oleh pengguna.
Iteratif	proses perulangan terhadap beberapa instruksi dan akan berhenti ketika persyaratan telah terpenuhi.
<i>User Interface</i>	tampilan aplikasi yang berhubungan langsung dengan pengguna.
RAD	sebuah nama metode pengembangan perangkat lunak.
Waterfall	sebuah nama metode pengembangan perangkat lunak.
<i>Framework</i>	kerangka kerja yang digunakan untuk pengembangan perangkat lunak.
CodeIgniter	sebuah nama kerangka kerja yang digunakan untuk pengembangan perangkat lunak menggunakan bahasa PHP.
PHP	sebuah nama bahasa pemrograman.
<i>Workshop</i>	kegiatan dimana orang-orang yang memiliki keahlian tertentu berkumpul lalu membahas permasalahan tertentu dan memberi pengajaran/pelatihan kepada para partisipan.
<i>Prototype</i>	model kerja dasar sebagai pengembangan perangkat lunak.
MySQL	suatu sistem manajemen basis data relasional yang mampu bekerja dengan cepat, kokoh, dan mudah digunakan serta gratis.
<i>Black-Box</i>	sebuah pengujian yang berfokus pada tampilan, fungsi-fungsi pada aplikasi serta kesesuaian alur fungsi dengan proses bisnis yang diinginkan calon pengguna.

DAFTAR ISI

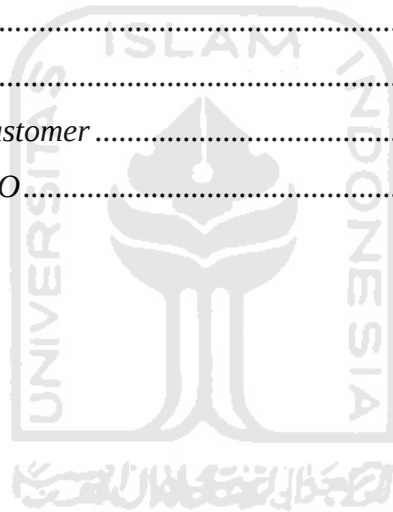
HALAMAN JUDUL	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING	ii
HALAMAN PENGESAHAN DOSEN PENGUJI	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR.....	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTO	vi
KATA PENGANTAR.....	vii
SARI	viii
GLOSARIUM	ix
DAFTAR ISI	x
DAFTAR TABEL	xii
DAFTAR GAMBAR.....	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Lingkup Aplikasi.....	2
1.4 Tujuan	2
1.5 Manfaat	3
1.6 Metodologi	3
1.7 Sistematika Penulisan	4
BAB II LANDASAN TEORI	5
2.1 Event Organizer	5
2.2 Rapid Application Development (RAD).....	5
2.3 Framework CodeIgniter	7
BAB III METODOLOGI	9
3.1 Metodologi RAD.....	9
3.2 Proses Perencanaan Kebutuhan	9
3.2.1 Validasi Ide.....	10
3.2.2 Analisis Kebutuhan Fungsional.....	10
3.2.3 Analisis Kebutuhan Non-Fungsional	12
3.3 Workshop Desain Sistem	13
3.3.1 Iterasi-1.....	13

3.3.2 Iterasi-2.....	21
3.4 Proses Implementasi	21
BAB IV HASIL DAN PEMBAHASAN.....	23
4.1 Implementasi Sistem Evoria	23
4.1.1 Implementasi Iterasi-1	23
4.1.2 Implementasi Iterasi-2	31
4.2 Pengujian Black-Box	50
4.3 Analisis Komersialisasi.....	56
BAB V KESIMPULAN DAN SARAN	57
5.1 Kesimpulan	57
5.2 Saran.....	57
DAFTAR PUSTAKA.....	58
LAMPIRAN	60



DAFTAR TABEL

Tabel 3.1 Kebutuhan Fungsional Evoria	12
Tabel 3.2 Tabel users	18
Tabel 3.3 Tabel jasa	18
Tabel 3.4 Tabel kategori_jasa	19
Tabel 3.5 Tabel user_access_menu.....	19
Tabel 3.6 Tabel user_ menu.....	19
Tabel 3.7 Tabel user_role	19
Tabel 3.8 Tabel user_sub_menu	20
Tabel 3.9 Tabel user_token.....	20
Tabel 3.10 Tabel pemesanan.....	20
Tabel 3.11 Tabel inspirasi.....	21
Tabel 4.1 Tabel pengujian <i>Admin</i>	50
Tabel 4.2 Tabel pengujian <i>User/Customer</i>	52
Tabel 4.3 Tabel pengujian <i>Seller/EO</i>	54



DAFTAR GAMBAR

Gambar 2.1 Tahapan RAD	6
Gambar 3.1 Tahapan RAD Evoria.....	9
Gambar 3.2 Flowchart Evoria.....	14
Gambar 3.3 <i>Use Case Diagram</i> Evoria	15
Gambar 3.4 Rancangan ERD Evoria	16
Gambar 3.5 Rancangan Activity Diagram Evoria	17
Gambar 4.1 <i>Layout Database</i> Evoria.....	23
Gambar 4.2 <i>Layout</i> Tabel users	24
Gambar 4.3 <i>Layout</i> Tabel jasa	24
Gambar 4.4 <i>Layout</i> Tabel kategori_jasa	25
Gambar 4.5 <i>Layout</i> Tabel user_access_menu.....	25
Gambar 4.6 <i>Layout</i> Tabel user_menu.....	25
Gambar 4.7 <i>Layout</i> Tabel user_role.....	25
Gambar 4.8 <i>Layout</i> Tabel user_sub_menu	26
Gambar 4.9 <i>Layout</i> Tabel user_token.....	26
Gambar 4.10 <i>Layout</i> Tabel pemesanan.....	26
Gambar 4.11 <i>Layout</i> Tabel inspirasi.....	27
Gambar 4.12 Tampilan Beranda Evoria	27
Gambar 4.13 Tampilan <i>Seller/EO</i>	28
Gambar 4.14 Tampilan <i>Inspiration</i>	28
Gambar 4.15 Tampilan <i>Order 1</i>	29
Gambar 4.16 Tampilan <i>Order 2</i>	29
Gambar 4.17 Tampilan <i>Order 3</i>	30
Gambar 4.18 Tampilan <i>Checkout</i>	30
Gambar 4.19 Tampilan Status Pesanan	31
Gambar 4.20 Tampilan <i>Register</i>	32
Gambar 4.21 Tampilan <i>Login</i>	32
Gambar 4.22 Tampilan <i>Carousel</i> Pada Beranda.....	33
Gambar 4.23 Tampilan Pilihan Tawaran Jasa Dari EO Pada Beranda.....	33
Gambar 4.24 Tampilan Pilihan Jenis EO Pada Beranda.....	34
Gambar 4.25 Tampilan Profil <i>Seller/EO 1</i>	35
Gambar 4.26 Tampilan Profil <i>Seller/EO 2</i>	35

Gambar 4.27 Tampilan Inspirasi.....	36
Gambar 4.28 Tampilan <i>Order</i> 1.....	36
Gambar 4.29 Tampilan <i>Order</i> 2.....	37
Gambar 4.30 Tampilan <i>Order</i> 3.....	37
Gambar 4.31 Tampilan Pembayaran 1.....	38
Gambar 4.32 Tampilan Pembayaran 2.....	38
Gambar 4.33 Tampilan Pembayaran 3.....	39
Gambar 4.34 Tampilan Pesanan Saya 1	39
Gambar 4.35 Tampilan Pesanan Saya 2	40
Gambar 4.36 Tampilan <i>Register Seller/EO</i> 1	40
Gambar 4.37 Tampilan <i>Register Seller/EO</i> 2	41
Gambar 4.38 Tampilan <i>Login Seller/EO</i>	41
Gambar 4.39 Tampilan Halaman Bisnis.....	42
Gambar 4.40 Tampilan Unggah Jasa	43
Gambar 4.41 Tampilan Unggah Inspirasi	43
Gambar 4.42 Tampilan Profil Bisnis <i>Seller/EO</i>	44
Gambar 4.43 Tampilan Edit Profil Bisnis <i>Seller/EO</i>	44
Gambar 4.44 Tampilan Pesanan 1	45
Gambar 4.45 Tampilan Pesanan 2	45
Gambar 4.46 Tampilan <i>Role</i>	46
Gambar 4.47 Tampilan Tambah <i>Role</i>	46
Gambar 4.48 Tampilan Kategori	47
Gambar 4.49 Tampilan Tambah Kategori	47
Gambar 4.50 Tampilan Menu <i>Management</i>	48
Gambar 4.51 Tampilan Tambah Menu	48
Gambar 4.52 Tampilan Submenu <i>Management</i>	49
Gambar 4.53 Tampilan Tambah Submenu	49

BAB I

PENDAHULUAN

1.1 Latar Belakang

Beberapa tahun terakhir, sektor MICE (*Meeting, Incentive, Convention, and Exhibition*) Yogyakarta bergantung pada industri pameran. Syamsun Hasani, Ketua Asosiasi Perusahaan Pameran Indonesia (Asperapi) Yogyakarta mengatakan, misalnya, dalam setahun Yogyakarta bisa menggelar 11 pameran komputer dengan 90.000-100.000 pengunjung. Yogyakarta memiliki sejumlah *Event Organizer* (EO) kreatif. Contohnya XO Production sukses menggelar pameran bertema kuliner yang tak biasa dengan judul Festival Jajanan Kekinian di Jogja Expo Center (JEC) dan Festival Jajanan Kekunoan yang mendatangkan lebih dari 45.000 pengunjung. Contoh lain adalah Kustomfest 2015, acara pameran modifikasi motor yang melibatkan 650 motor kustom, 300 mobil hotrod, dan 50 pintriper. Sekitar 300 *booth* yang ada dalam Kustomfest 2015 sukses menarik perhatian 20.000 pengunjung (Venuemagz, 2016).

EO juga diperlukan pada acara pernikahan terutama yang menghabiskan biaya relatif tinggi. Menurut Badan Pusat Statistik (BPS), di Yogyakarta setidaknya terdapat 10.000 lebih pasangan beragama Islam yang menikah setiap tahunnya (BPS, 2018). Sedangkan untuk biaya pernikahan, menurut survei dari Bridestory pada tahun 2017, 5000 pengantin pria dan wanita menghabiskan biaya resepsi rata-rata adalah 20 juta sampai 400 juta. Biaya tersebut normalnya dialokasikan ke beberapa vendor maupun EO. Dalam hal memilih vendor atau EO, survei membuktikan bahwa faktor harga (40.3%), portofolio (25.9%) dan *review* (12.9%) merupakan tiga hal utama yang membantu pasangan menentukan satu vendor untuk hari pernikahan (Bridestory, 2017).

Selain acara resepsi pernikahan, berdasarkan *Calendar of Event* (COE) Yogyakarta tahun 2019, terdapat 200 lebih acara yang diselenggarakan oleh pemerintah. Belum lagi acara lainnya seperti konser yang dilakukan minimal sekali dalam sebulan (Visitingjogja, 2019). Acara tersebut juga akan memerlukan beberapa vendor maupun EO sehingga potensinya sangat besar bagi pelaku usaha EO.

Pengguna jasa EO ini biasanya adalah masyarakat kelas menengah ke atas (Sugiarto, 2019). Banyak acara-acara di daerah yang memiliki potensi untuk digarap. Hanya saja kompetensi pelaku usaha masih belum merata di setiap daerah sehingga membuat acara lebih terpusat di kota-kota besar. Padahal, menurut *founder* Backstagers Indonesia Krisnanto

Sutrisman, industri EO memiliki pertumbuhan sekitar 15% hingga 20% dengan nilai industri lebih dari Rp. 500 triliun. Bahkan saat ini sudah ada sekitar 4.000 pelaku usaha dengan serapan tenaga kerja formal sekitar 40.000 orang (Richard, 2019).

Selama ini masyarakat masih mengandalkan kenalan dari orang lain untuk mencari EO yang sesuai dan yang dihubungi biasanya adalah EO yang sudah terkenal atau sudah mempunyai nama. Banyak para EO yang baru merintis dan sudah memiliki portofolio namun belum diketahui banyak orang atau belum memiliki nama. Oleh karena itu dibutuhkan Sistem Informasi yang dapat memasarkan dan meningkatkan perkembangan industri ini. Berdasarkan permasalahan tersebut dirintislah ide pembuatan *startup* yang bernama Evoria. Evoria adalah aplikasi *marketplace event organizer* berbasis web yang bertujuan menghubungkan EO dengan masyarakat yang ingin menggunakan jasa EO untuk kebutuhan acaranya. Evoria menyediakan tempat untuk memasarkan jasa dan portofolio para EO agar dapat dilihat dan diketahui banyak orang.

Pekerjaan dalam perintisan bisnis Evoria ini dibagi menjadi 3 sesuai dengan *role* tiap-tiap anggotanya, yaitu: 1) yang berfokus pada sudut pandang bisnisnya (*Hustler*), 2) yang berfokus pada desain aplikasinya (*Hipster*), dan 3) yang berfokus pada pengembangan aplikasinya (*Hacker*). Laporan ini memilih fokus pada bagian pengembangan aplikasi.

1.2 Rumusan Masalah

Berdasarkan penjelasan pada bagian latar belakang, rumusan masalah pada perintisan bisnis ini adalah bagaimana cara mengembangkan aplikasi Evoria.

1.3 Lingkup Aplikasi

Lingkup aplikasi Evoria ini adalah sebagai berikut:

- a. Aplikasi digunakan untuk memasarkan jasa EO yang berdomisili di Yogyakarta dikarenakan tujuan awal pengembangan aplikasi fokus kepada EO yang hanya melayani acara di Yogyakarta.
- b. Aplikasi berbasis *website* dan hanya dapat diakses menggunakan browser.
- c. Aplikasi akan memasarkan jasa *event/acara* yang disediakan oleh para EO.

1.4 Tujuan

Tujuan dari perintisan bisnis bagian *hacker* ini adalah mengembangkan aplikasi Evoria yang berbasis web.

1.5 Manfaat

Manfaat dari pengembangan aplikasi Evoria adalah sebagai berikut:

- a. Mengimplementasikan ilmu yang diperoleh peneliti selama belajar di Universitas Islam Indonesia.
- b. Membantu meningkatkan perekonomian EO.
- c. Mempermudah masyarakat yang ingin mencari dan menggunakan jasa EO.
- d. Turut serta dalam perkembangan *startup* di Indonesia.

1.6 Metodologi

Dalam melakukan pengembangan aplikasi Evoria, peneliti menggunakan pendekatan *Rapid Application Development* (RAD). Pendekatan RAD memiliki tahap-tahap sebagai berikut:

- a. Proses Perencanaan Kebutuhan

Dalam rencana mengembangkan Evoria, validasi ide perlu dilakukan dan melalui beberapa proses. Proses tersebut juga dipilih untuk menentukan apa saja kebutuhan calon pengguna terhadap Evoria. Jika data kebutuhan telah didapatkan, maka proses pengembangan bisa dilakukan. Proses validasi terdiri atas: *interview* (tanya jawab), *questionnaires* (pembagian kuesioner), dan *observation* (observasi). Selain validasi ide terdapat analisis kebutuhan fungsional dan non fungsional.

- b. Proses Desain Sistem

Proses desain sistem dilakukan dengan membuat tampilan *prototype* sebagai gambaran desain sistem yang akan diimplementasikan menjadi sebuah tampilan *website*. Desain *prototype* kemudian ditunjukkan kepada calon pengguna untuk dikonfirmasi sehingga didapatkan spesifikasi desain yang sesuai dengan kebutuhan pengguna.

- c. Proses Implementasi

Proses implementasi adalah proses pengembangan aplikasi dengan mengimplementasikan apa saja yang sudah disepakati berdasarkan kebutuhan dan desain sistem. Kemudian pengujian dilakukan untuk memeriksa dan menguji apakah kebutuhan pada aplikasi sudah sesuai atau belum. Apabila sudah sesuai maka aplikasi dapat digunakan sebagaimana mestinya. Apabila masih terdapat kekurangan, maka dilakukan evaluasi dan perbaikan sampai aplikasi benar-benar berjalan sesuai kebutuhan.

1.7 Sistematika Penulisan

Untuk mempermudah penyusunan laporan, maka digunakan sistematika penulisan sebagai berikut :

BAB I PENDAHULUAN

Bagian pendahuluan membahas latar belakang, rumusan masalah, lingkup aplikasi, tujuan, manfaat, metodologi serta sistematika penulisan.

BAB II LANDASAN TEORI

Bagian landasan teori membahas tentang perintisan bisnis, teori-teori yang berkaitan dengan proses pembuatan aplikasi Evoria dan hal-hal pendukung lainnya.

BAB III METODOLOGI

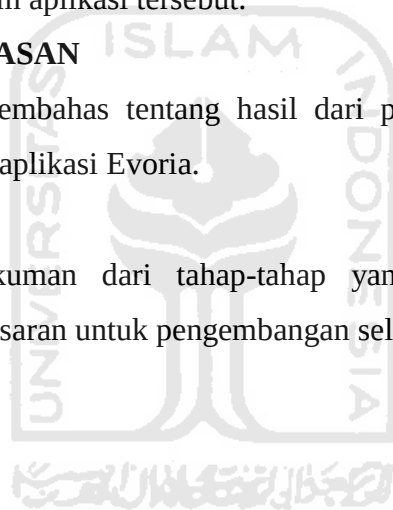
Bagian metodologi membahas tentang pendekatan yang digunakan untuk mengembangkan aplikasi Evoria dan gambaran umum aplikasi tersebut.

BAB IV HASIL DAN PEMBAHASAN

Bagian hasil dan pembahasan membahas tentang hasil dari penerapan pendekatan yang digunakan untuk mengembangkan aplikasi Evoria.

BAB V KESIMPULAN

Bagian kesimpulan berisi rangkuman dari tahap-tahap yang telah dilakukan dalam pengembangan Evoria, serta saran-saran untuk pengembangan selanjutnya.



BAB II

LANDASAN TEORI

2.1 Event Organizer

Event Organizer (EO) adalah istilah penyedia jasa profesional yang bertugas sebagai penyelenggara acara. EO bertugas membantu kliennya dalam menyelenggarakan acara baik yang berukuran kecil, menengah, maupun besar. Jasa EO dibutuhkan karena keterbatasan waktu dan sumber daya yang dimiliki klien (Dr. Bachruddin Saleh Luturlean, 2019).

Beberapa jenis EO berdasarkan dari jenis acara yang diadakan:

1. *One Stop Service Agency*: EO besar yang mampu menyelenggarakan berbagai jenis acara hingga skala internasional sekalipun.
2. *MICE (Meeting, Incentive, Convention, Exhibition)*: EO yang khusus bergerak di bidang penyelenggaraan acara berbentuk pertemuan.
3. *Brand Activation*: adalah EO yang secara spesifik membantu client-nya untuk mempromosikan *brand*/merk di kalangan konsumen, dengan berinteraksi langsung ke target pasarnya.
4. Musik dan Hiburan: EO yang memiliki spesialisasi di bidang hiburan terutama musik.
5. Penyelenggara Pernikahan: EO yang khusus membantu klien mengadakan pesta pernikahan.
6. Penyelenggara Ulang Tahun: EO yang ahli membuat pesta ulang tahun termasuk untuk anak-anak.
7. Penyelenggara Pribadi: EO khusus yang bergerak untuk penyelenggaraan pesta pribadi.

2.2 Rapid Application Development (RAD)

Rapid Application Development (RAD) merupakan metode pengembangan sistem informasi dengan waktu singkat, sehingga dinilai tepat digunakan dalam pengembangan Evoria. RAD menggunakan metode iteratif (berulang) dalam mengembangkan sistem dimana *working model* (model bekerja) sistem dikonstruksikan di awal tahap pengembangan dengan tujuan menetapkan kebutuhan (*requirement*) pengguna dan selanjutnya disingkirkan. Dibandingkan dengan metode lain seperti Waterfall, keunggulan dari metode RAD adalah kecepatan, ketepatan dan biaya yang relatif lebih rendah. Namun bukan berarti metode pengembangan lain tidak lebih baik dari metode RAD (Widiyanto, 2018). Tim Axel

memutuskan untuk menggunakan pendekatan RAD dikarenakan masa pengembangannya memerlukan waktu yang cepat. Dalam pengembangan sistem informasi normal, memerlukan waktu minimal 180 hari, namun dengan menggunakan metode RAD, sistem dapat diselesaikan dalam waktu 30-90 hari (Wahyuningrum & Januarita, 2014). Selain itu metode RAD melibatkan pengguna dalam proses pengembangan sehingga kebutuhan sistem dapat terpenuhi dengan baik dan sesuai dengan keinginan pengguna itu sendiri. Terdapat 3 tahapan dari metode RAD seperti pada Gambar 2.1 :



Gambar 2.1 Tahapan RAD
Sumber: (Piyaneo, 2014)

a. Proses Perencanaan Syarat-Syarat/Kebutuhan

Tahap ini merupakan hal yang paling penting karena diperlukan adanya keterlibatan antara kedua belah pihak yaitu calon pengguna dan analis. Calon pengguna dan Analis melakukan pertemuan untuk mendiskusikan kebutuhan sistem dan kebutuhan informasi untuk mencapai tujuan yang diinginkan.

b. Proses Desain Sistem/*Workshop Desain*

Proses desain sistem dilakukan dengan melibatkan keaktifan calon pengguna dalam melakukan konfirmasi dan perbaikan terhadap kesesuaian desain antara calon pengguna dan analis. Seorang calon pengguna dapat memberikan masukan dan komentar apabila terdapat ketidaksesuaian pada desain serta melakukan perancangan sistem yang mengacu pada kebutuhan sistem yang sudah dibuat pada tahap sebelumnya. Keluaran dari tahap ini adalah spesifikasi sistem yang meliputi organisasi sistem secara umum, struktur data, dan lain-lain.

c. Proses Implementasi

Proses implementasi adalah proses pengembangan aplikasi dengan mengimplementasikan apa saja yang sudah disepakati berdasarkan kebutuhan dan desain sistem. Kemudian pengujian dilakukan untuk memeriksa dan menguji apakah kebutuhan pada aplikasi sudah sesuai atau belum. Apabila sudah sesuai maka aplikasi dapat digunakan sebagaimana mestinya. Apabila masih terdapat kekurangan, maka dilakukan evaluasi dan perbaikan sampai aplikasi benar-benar berjalan sesuai kebutuhan.

2.3 Framework CodeIgniter

CodeIgniter adalah Kerangka Pengembangan Aplikasi (*toolkit*) dengan model MVC (*Model, View, Controller*) untuk orang yang ingin membangun situs web menggunakan PHP. CodeIgniter pada awalnya dikembangkan oleh EllisLab dan dirilis pertama kali pada 28 Februari 2006. Pada tahun 2014 CodeIgniter diakuisisi oleh Institut Teknologi British Columbia dan kemudian secara resmi diumumkan sebagai *community-maintained project*. Tujuannya adalah memungkinkan *developer* untuk mengembangkan proyek lebih cepat daripada jika menulis kode dari awal, dengan menyediakan serangkaian *library* untuk tugas-tugas yang dibutuhkan, antarmuka yang sederhana, dan struktur logis untuk mengakses *library* ini.

CodeIgniter memungkinkan *developer* secara kreatif fokus pada proyek dengan meminimalkan jumlah kode yang diperlukan untuk tugas yang diberikan. Jika memungkinkan, CodeIgniter telah dibuat sefleksibel mungkin, memungkinkan *developer* untuk bekerja dengan cara yang mereka inginkan, tidak dipaksa untuk bekerja dengan cara tertentu. Kerangka kerja ini dapat memiliki bagian inti yang mudah di-*extend* atau sepenuhnya diganti untuk membuat sistem kerja seperti yang diinginkan (CodeIgniter, Welcome to CodeIgniter4, 2020).

Database diperlukan untuk sebagian besar pemrograman aplikasi web. *Database* yang didukung saat ini adalah:

1. MySQL (5.1+) melalui *driver* MySQLi.
2. PostgreSQL melalui *driver* Postgre.
3. SQLite3 melalui *driver* SQLite3.

Seperti kebanyakan kerangka kerja web, CodeIgniter menggunakan pola *Model, View, Controller* (MVC) untuk mengatur file. Ini membuat data, presentasi, dan mengalir melalui aplikasi sebagai bagian yang terpisah (CodeIgniter, CodeIgniter4 Overview, 2020).

1. *Model* mengelola data aplikasi dan membantu menegakkan aturan bisnis khusus yang mungkin diperlukan aplikasi dan berhubungan langsung dengan pangkalan data untuk memanipulasi data (*insert, update, delete, search*).
2. *View* adalah file sederhana, dengan sedikit atau tanpa logika, yang menampilkan informasi kepada pengguna.
3. *Controller* merupakan bagian yang menghubungkan antara bagian *model* dan bagian *view*, controller berfungsi untuk menerima permintaan dan data dari pengguna kemudian menentukan apa yang akan diproses oleh aplikasi.

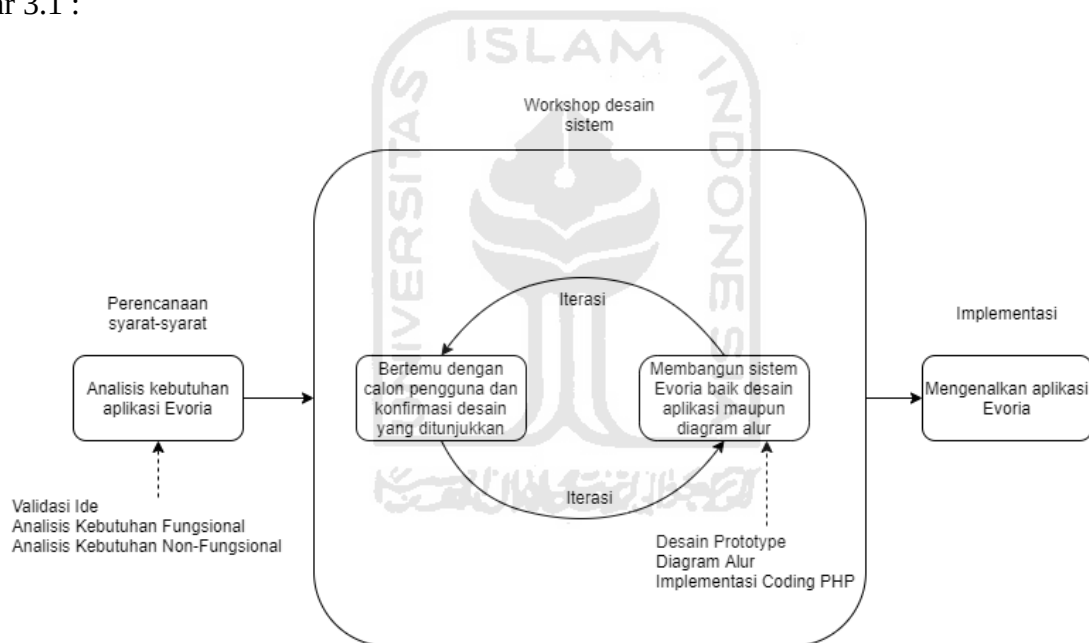
Dalam pengembangan Evoria, tim Axel menggunakan framework CodeIgniter karena aplikasi dibuat berbasis web dan menggunakan bahasa pemrograman PHP. CodeIgniter dinilai lebih rapi dalam menghasilkan struktur pemograman, baik dari segi kode maupun struktur *file* PHPnya. Selain itu CodeIgniter memudahkan orang lain yang ingin mempelajari sistem yang dibuat sehingga sistem dapat dikembangkan dengan cepat dan lebih baik.



BAB III METODOLOGI

3.1 Metodologi RAD

Rapid Application Development (RAD) merupakan metodologi atau pendekatan untuk pengembangan sistem informasi dengan waktu singkat, sehingga dinilai tepat digunakan dalam pengembangan Evoria. Keunggulan dari metode ini adalah kecepatan, ketepatan dan biaya yang relatif lebih rendah. Selain itu RAD melibatkan pengguna dalam proses pengembangan sehingga kebutuhan sistem dapat terpenuhi dengan baik dan sesuai dengan keinginan pengguna itu sendiri. Terdapat 3 tahapan dari pendekatan RAD dalam pengembangan Evoria seperti pada Gambar 3.1 :



Gambar 3.1 Tahapan RAD Evoria

3.2 Proses Perencanaan Kebutuhan

Dalam proses perencanaan kebutuhan, tim Axel melakukan langkah-langkah seperti validasi ide, analisis kebutuhan fungsional serta analisis kebutuhan non fungsional. Proses perencanaan kebutuhan merupakan proses paling penting karena tanpa adanya data tentang kebutuhan, maka aplikasi tidak dapat dikembangkan dan solusi permasalahan tidak akan didapatkan.

3.2.1 Validasi Ide

Sebuah *startup* berpeluang sukses lebih besar ketika mampu menguji dan melakukan validasi terhadap ide yang dimilikinya. Dengan berinteraksi terhadap orang lain serta menanyakan pertanyaan yang tepat, kita dapat menguji apakah ide yang kita miliki benar-benar dapat menyelesaikan permasalahan yang ada atau tidak. Selain itu, kita juga akan mengetahui calon konsumen dan ketersediaan pasar untuk produk yang dikembangkan. Salah satu risiko terbesar dari mengembangkan sebuah produk adalah mendesain sesuatu yang salah (Manuwu, 2018). Dalam pengembangan sebuah ide ataupun produk *startup*, proses validasi ide adalah tahap yang paling penting. Validasi adalah proses pengumpulan bukti dan pembelajaran terkait ide bisnis melalui eksperimen dan *user testing*. Karena dengan ide yang valid, maka desain yang dihasilkan akan lebih baik dan bermanfaat ketika dikembangkan menjadi sebuah produk. Apalagi jika produk tersebut adalah produk bisnis dengan modal tertentu (Utami, 2020). Dengan membagikan ide kepada orang lain dan mendapatkan *feedback*, maka gagasan dapat disempurnakan menjadi lebih matang (Annisa, 2020).

Dalam rencana pengembangan aplikasi Evoria, validasi ide perlu dilakukan dan melalui beberapa proses. Proses tersebut juga dipilih untuk menentukan apa saja kebutuhan calon pengguna terhadap aplikasi yang dibuat. Jika data kebutuhan telah didapatkan, maka proses pengembangan bisa dilakukan. Proses validasi terdiri atas: *interview* (tanya jawab), *questionnaires* (pembagian kuesioner), dan *observation* (observasi) (Jamaluddin, 2017).

Dalam proses validasi ide dan analisis kebutuhan, tim Axel bertemu langsung dengan calon pengguna dan melakukan wawancara. Data yang diperoleh berupa fitur-fitur, proses bisnis, proses persetujuan antara vendor dan calon pengguna, proses transaksi, dan proses komplain/kompensasi. Semua data tersebut akan digunakan dalam pengembangan Evoria.

Selain itu tim Axel juga membagikan kuesioner secara *online* kepada para calon pengguna Evoria. Data yang diperoleh berupa jawaban dari pertanyaan seputar permasalahan yang terjadi di dunia EO baik dari vendor maupun pengguna jasanya, perlu tidaknya Evoria ini untuk dikembangkan, serta saran responden terhadap Evoria ini. Respon/jawaban yang diperoleh kemudian dianalisis dan diringkas.

3.2.2 Analisis Kebutuhan Fungsional

Dari hasil wawancara langsung yang telah dilakukan, tim Axel mendapatkan beberapa data tentang proses bisnis Konvensional EO :

1. Menghubungi *Event Organizer*, bisa *chat* via Whatsapp, SMS, dll.
2. Pilih paket dan bujet/anggaran (proses persetujuan 1 bulan sebelum hari H).
3. Menghubungi lagi *Event Organizer*, bisa *chat* via Whatsapp, SMS, dll.
4. MOU (*Memorandum of Understanding*), yang dalam bahasa Indonesia disebut Nota Kesepahaman atau Nota Kesepakatan.
5. Acara berlangsung, hingga acara selesai.
6. Komplain dan kompensasi.

Selain data tentang proses bisnis Konvensional EO, tim Axel juga mendapatkan beberapa data untuk mengembangkan aplikasi Evoria ini berdasarkan kebutuhan responden. Berikut beberapa kebutuhan pengembangan Evoria:

1. Aplikasi ditujukan untuk calon pengguna yang berusia remaja hingga dewasa.
2. Aplikasi dapat digunakan untuk EO, *customer*, dan administrator.
3. Aplikasi dapat digunakan sesuai dengan ketersediaan wilayah layanan EO.
4. Aplikasi berbasis web dan dapat diakses melalui peramban pada komputer maupun *smartphone*.
5. Setiap EO dapat memasang portofolio di dalam aplikasi dan memasarkan jasanya.
6. Setiap EO yang mendaftar akan dikenai persyaratan minimum untuk mengindikasikan bahwa EO yang didaftarkan adalah asli.
7. Terdapat fitur tag eksklusif yang diperuntukkan bagi EO yang sudah terverifikasi supaya menambah kepercayaan *customer*.
8. *Customer* dapat melihat informasi dan memilih layanan yang disediakan EO sesuai kebutuhan.
9. *Customer* dan EO harus mentaati MOU dan dapat menyetujuinya lewat aplikasi.
10. *Customer* dapat melakukan pembayaran transaksi melalui metode pembayaran yang disediakan aplikasi.
11. *Customer* dapat memberikan penilaian terhadap kinerja EO.

Kebutuhan fungsional berisi layanan-layanan dan fitur-fitur yang disediakan oleh aplikasi Evoria untuk diakses penggunanya. Kebutuhan fungsional Evoria dapat dilihat pada Tabel 3.1 :

Tabel 3.1 Kebutuhan Fungsional Evoria

No	Pengguna	Tujuan
1	<i>Customer</i>	<ul style="list-style-type: none"> • Masuk menu <i>Customer</i> • Mengelola data profil • Melihat portofolio <i>Seller/EO</i> • Melihat produk jasa • Memilih produk jasa • Melihat informasi detail produk jasa • Memesan produk jasa • Membatalkan pesanan produk jasa • Memantau proses pemesanan produk jasa • Menyelesaikan proses pemesanan
2	<i>Seller/EO</i>	<ul style="list-style-type: none"> • Masuk menu <i>Seller/EO</i> • Mengelola data profil • Menambah produk jasa • Mengubah data produk jasa • Melihat informasi detail produk jasa • Melihat data pesanan produk jasa • Memproses pesanan produk jasa • Membatalkan/menolak pesanan produk jasa • Memantau proses pesanan produk jasa
3	<i>Admin</i>	<ul style="list-style-type: none"> • Masuk menu <i>Admin</i> • Mengakses semua menu • Mengolah akses menu • Menambahkan menu • Menghapus menu

3.2.3 Analisis Kebutuhan Non-Fungsional

Kebutuhan non-fungsional merupakan spesifikasi yang dibutuhkan untuk mengakses sistem agar penggunaan sistem dapat efektif serta *user-friendly* sehingga tidak menyusahkan penggunanya. Berikut adalah analisis kebutuhan non-fungsional aplikasi Evoria :

Kebutuhan Perangkat Keras

Perangkat keras merupakan komponen yang paling penting dalam menjalankan aplikasi Evoria ini. Perangkat keras mempunyai peranan penting dalam pembuatan sistem maupun pengolahan data. Berikut spesifikasi yang dibutuhkan untuk menjalankan aplikasi Evoria :

1. Laptop/Komputer Prosesor Intel® Core™ i3 - ke atas.
2. Modem.

Kebutuhan Perangkat Lunak

Perangkat keras tidak akan berfungsi tanpa adanya perangkat lunak karena perangkat lunak merupakan penggerak fungsi dari perangkat keras tersebut. Berikut perangkat lunak yang diperlukan untuk menjalankan aplikasi Evoria :

1. Sistem Operasi *Windows 10*.
2. Browser *Google Chrome*.
3. XAMPP.

3.3 Workshop Desain Sistem

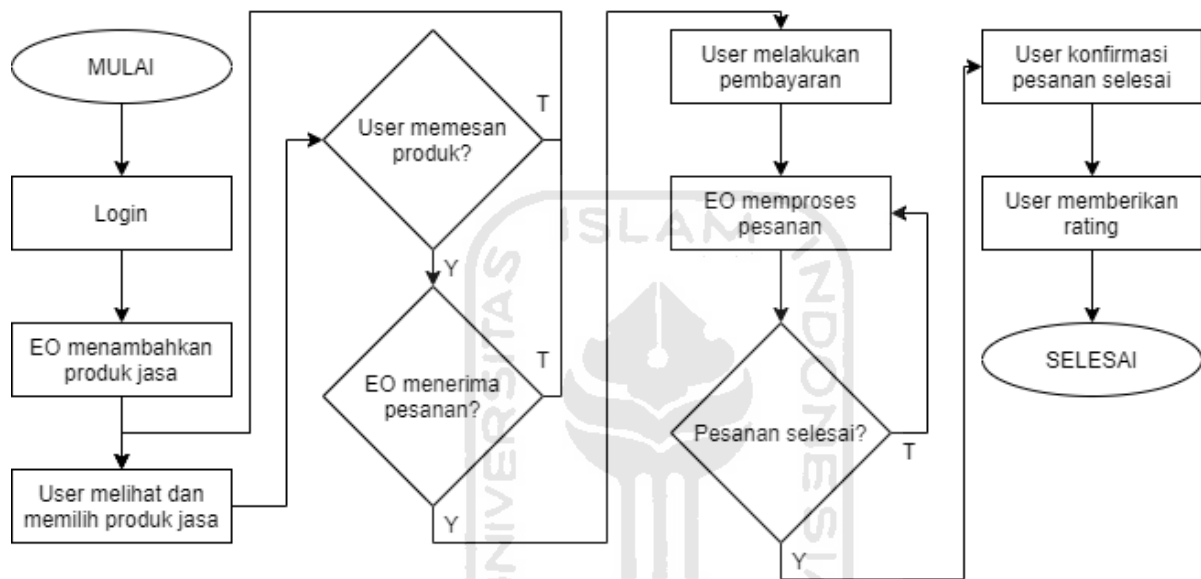
Workshop desain sistem dilakukan dengan membuat tampilan *prototype* sebagai gambaran desain sistem yang akan diimplementasikan menjadi sebuah tampilan *website*. Selain desain *prototype*, terdapat rancangan diagram alur yang kemudian ditunjukkan kepada calon pengguna untuk dikonfirmasi sehingga didapatkan spesifikasi desain yang sesuai dengan kebutuhan pengguna.

3.3.1 Iterasi-1

Setelah membuat *prototype*, tim Axel menunjukkan hasilnya kepada calon pengguna dengan melakukan pertemuan langsung. Dalam iterasi pertama, tim Axel membuat desain *prototype* yang menggambarkan tampilan beranda, *Seller/EO*, dan lain-lain pada aplikasi Evoria menggunakan *tools* Adobe XD dan MarvelApp. Selain *prototype*, tim Axel juga menunjukkan rancangan proses berjalannya aplikasi dalam beberapa bagan yaitu *Flowchart*, *Use Case Diagram*, rancangan ERD, dan *Activity Diagram*. Pembuatan bagan-bagan tersebut menggunakan *tools* ERDPlus.

Flowchart Evoria

Dalam rencana pengembangan aplikasi tentu saja pengembang harus menentukan gambaran alur proses dari aplikasi tersebut. Gambaran alur proses dapat diimplementasikan ke dalam *Flowchart*. *Flowchart* adalah diagram yang menggambarkan langkah-langkah atau alur proses penyelesaian suatu masalah sehingga lebih mudah dipahami. *Flowchart* Evoria menggambarkan bagaimana proses berjalannya aplikasi Evoria ini. Proses berjalannya Evoria dapat dilihat pada Gambar 3.2 :

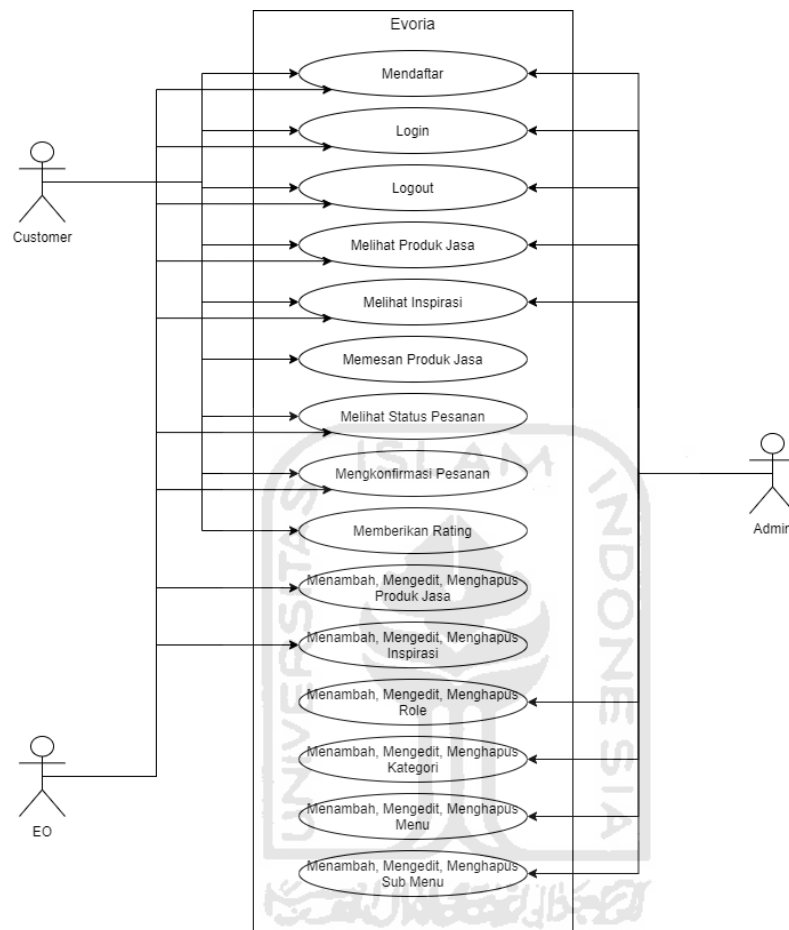


Gambar 3.2 Flowchart Evoria

Pada Gambar 3.2, proses dimulai dengan *user* dan EO melakukan login. Setelah *user* dan EO melakukan login, EO mulai menambahkan produk jasa yang akan ditawarkan kepada calon *customer* (*user*). *User* kemudian memilih produk jasa yang diinginkan dilanjutkan dengan proses pemesanan. Setelah *user* memesan produk jasa, EO menerima pesanan dan *user* melakukan pembayaran. Kemudian EO memproses pesanan hingga status pesanan tersebut selesai. Setelah status pesanan selesai, *user* melakukan konfirmasi selesai dan memberikan rating kepada EO.

Use Case Diagram Evoria

Use Case Diagram merupakan diagram yang menggambarkan hubungan antara aktor dengan sistem. Rancangan *Use Case Diagram* Evoria dapat dilihat pada Gambar 3.3 :

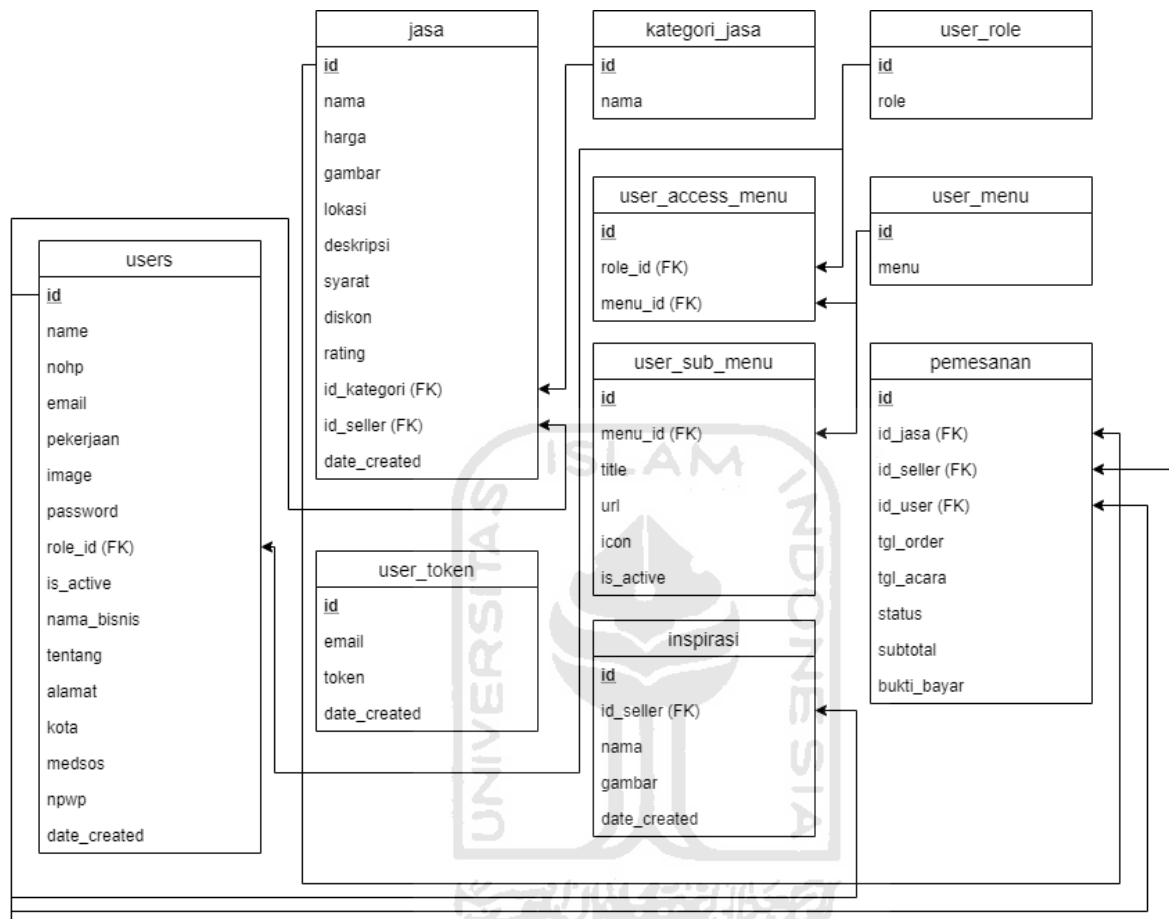


Gambar 3.3 *Use Case Diagram* Evoria

Pada Gambar 3.3, rancangan *Use Case Diagram* menggambarkan apa saja yang bisa dilakukan masing-masing aktor terhadap fitur-fitur pada aplikasi Evoria.

ERD Evoria

ERD (*Entity Relationship Diagram*) menggambarkan hubungan/relasi antar tabel dalam sebuah basis data. Rancangan ERD Evoria dapat dilihat pada Gambar 3.4 :

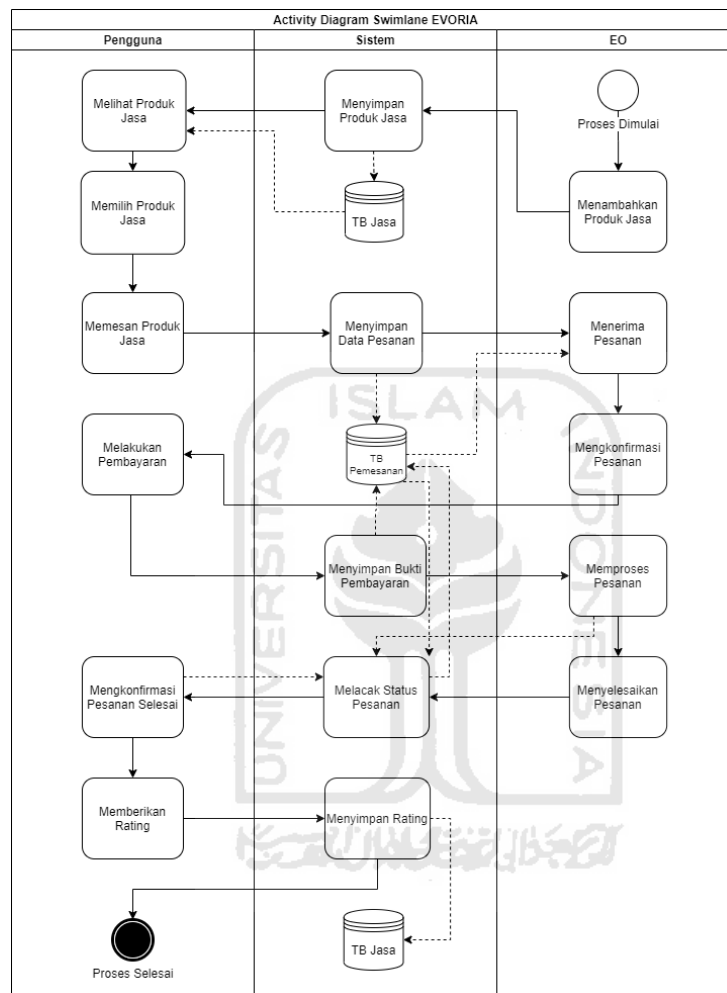


Gambar 3.4 Rancangan ERD Evoria

Pada Gambar 3.4, rancangan ERD menggambarkan hubungan antar tabel dalam sebuah basis data yang digunakan pada aplikasi Evoria.

Activity Diagram Evoria

Activity diagram Evoria menggambarkan proses mulai dari EO menambahkan produk jasa sampai *user* selesai melakukan pemesanan produk jasa. Activity diagram Evoria dapat dilihat pada Gambar 3.5 :



Gambar 3.5 Rancangan Activity Diagram Evoria

Rancangan Basis Data

Rancangan basis data terdiri dari beberapa tabel. Rancangan basis data digunakan dalam proses pengembangan aplikasi Evoria ini. Berikut tabel-tabel yang digunakan pada aplikasi Evoria :

1. Tabel users

Tabel users digunakan untuk menampung data pengguna yaitu *user*, *Seller/EO*, dan *Admin*. Tabel users dapat dilihat pada Tabel 3.2 :

Tabel 3.2 Tabel users

Field	Type	Size	Keterangan
id	int	11	PK
name	varchar	128	
nohp	varchar	15	
email	varchar	128	
pekerjaan	varchar	100	
image	varchar	128	
password	varchar	256	
role_id	int	11	FK
is_active	int	1	0 atau 1
nama_bisnis	varchar	100	
tentang	varchar	10000	
alamat	varchar	1000	
kota	varchar	100	
medsos	varchar	100	
npwp	varchar	100	
date_created	int	11	

2. Tabel jasa

Tabel jasa digunakan untuk menampung data produk jasa. Tabel jasa dapat dilihat pada Tabel 3.3 :

Tabel 3.3 Tabel jasa

Field	Type	Size	Keterangan
id	int	10	PK
nama	varchar	100	
harga	int	100	
gambar	varchar	100	
lokasi	varchar	100	
deskripsi	varchar	10000	
syarat	varchar	10000	
diskon	float		
rating	float		
id_kategori	int	11	FK
id_seller	int	11	FK
date_created	int	11	

3. Tabel kategori_jasa

Tabel kategori_jasa digunakan untuk menampung kategori produk jasa. Tabel kategori_jasa dapat dilihat pada Tabel 3.4 :

Tabel 3.4 Tabel kategori_jasa

Field	Type	Size	Keterangan
id	int	11	PK
nama	varchar	100	

4. Tabel user_access_menu

Tabel user_access_menu menentukan menu apa saja yang boleh diakses oleh masing-masing *user* berdasarkan *role_id*. Tabel user_access_menu dapat dilihat pada Tabel 3.5 :

Tabel 3.5 Tabel user_access_menu

Field	Type	Size	Keterangan
id	int	11	PK
role_id	int	11	FK
menu_id	int	11	FK

5. Tabel user_menu

Tabel user_menu berisi nama pembagian menu untuk setiap pengguna yaitu *user*, *Seller/EO*, dan *Admin*. Tabel user_menu dapat dilihat pada Tabel 3.6 :

Tabel 3.6 Tabel user_menu

Field	Type	Size	Keterangan
id	int	11	PK
menu	varchar	128	

6. Tabel user_role

Tabel user_role berisi nama role dari setiap pengguna yaitu *user*, *Seller/EO*, dan *Admin*. Tabel user_role dapat dilihat pada Tabel 3.7 :

Tabel 3.7 Tabel user_role

Field	Type	Size	Keterangan
id	int	11	PK
role	varchar	128	

7. Tabel user_sub_menu

Tabel user_sub_menu berisi nama/judul sub menu yang akan muncul pada setiap menu pengguna berdasarkan *menu_id*, jadi setiap role pengguna akan memiliki sub

menu yang berbeda-beda karena hak aksesnya juga berbeda. Tabel `user_sub_menu` dapat dilihat pada Tabel 3.8 :

Tabel 3.8 Tabel `user_sub_menu`

Field	Type	Size	Keterangan
id	int	11	PK
menu_id	int	11	
title	varchar	128	
url	varchar	128	
icon	varchar	128	
is_active	int	1	

8. Tabel `user_token`

Tabel `user_token` digunakan untuk menampung token registrasi guna verifikasi email. Tabel `user_token` dapat dilihat pada Tabel 3.9 :

Tabel 3.9 Tabel `user_token`

Field	Type	Size	Keterangan
id	int	11	PK
email	varchar	128	
token	varchar	128	
date_created	int	11	

9. Tabel pemesanan

Tabel pemesanan digunakan untuk menampung data pesanan dari *user* serta sebagai *tracking* status dari pesanan tersebut. Tabel pemesanan dapat dilihat pada Tabel 3.10 :

Tabel 3.10 Tabel pemesanan

Field	Type	Size	Keterangan
id	int	11	PK
id_jasa	int	11	FK
id_seller	int	11	FK
id_user	int	11	FK
tgl_order	date		
tgl_acara	date		
status	int	11	
subtotal	float		
bukti_bayar	varchar	100	
rating	float		

10. Tabel inspirasi

Tabel inspirasi digunakan untuk menampung data inspirasi dari *Seller/EO* yang akan ditampilkan pada halaman Inspirasi sehingga bisa dilihat oleh *user*. Tabel inspirasi dapat dilihat pada Tabel 3.11 :

Tabel 3.11 Tabel inspirasi

Field	Type	Size	Keterangan
id	int	11	PK
id_seller	int	11	FK
nama	varchar	100	
gambar	varchar	100	
date_created	int	11	

3.3.2 Iterasi-2

Dalam iterasi kedua, tim Axel menunjukkan hasil penerapan *prototype* ke *coding* kepada calon pengguna dengan melakukan pertemuan langsung. Tim Axel membuat desain tampilan website yang menggambarkan tampilan beranda, *Seller/EO*, dan lain-lain dengan cara *coding* menggunakan bahasa PHP dan *framework* CodeIgniter.

3.4 Proses Implementasi

Dalam pengembangan aplikasi Evoria, tim Axel sepakat untuk membuat aplikasi berbasis web. Kesuksesan menggunakan RAD tergantung dari keahlian dan kemampuan tim untuk memahami keinginan *customer* dari proyek pengembangan yang dibuat. Kunci utama kesuksesan RAD adalah proses pembelajaran dan adaptasi. Pada bagian *back-end* tim Axel melakukan pengembangan menggunakan *tools* sebagai berikut :

1. Bahasa pemrograman PHP, penggunaan bahasa PHP untuk mengembangkan *website* Evoria membuat proses pengembangannya menjadi lebih mudah dan lebih cepat dikarenakan pengembang sudah memahaminya.
2. *Framework* Code Igniter, penggunaan *framework* Code Igniter membuat pengelolaan *file* proyek menjadi lebih rapi.
3. Bootstrap, penggunaan Bootstrap membuat tampilan website menjadi lebih bagus dan cukup mudah dimodifikasi
4. XAMPP, Program aplikasi XAMPP berfungsi sebagai server lokal untuk mengampu berbagai jenis data website yang sedang dalam proses pengembangan.

5. VSCode, Program aplikasi VSCode berisi banyak fitur dan ekstensi mulai dari penataan kode otomatis, local terminal, *auto rename tag*, dan lain-lain. Hal ini membuat proses *coding* menjadi minim kesalahan sehingga pengembang tidak perlu membuang waktu untuk mencari baris kode yang bermasalah.
6. *Database MySQL*, *syntax* pada database MySQL sangat mudah dipahami dan cocok digunakan untuk Evoria sehingga pengolahan *database* menjadi lebih mudah.

Sedangkan pada bagian *front-end* tim Axel menggunakan bantuan *tools* pengolahan grafis seperti Adobe Photoshop, Canva, Adobe XD, dan MarvelApp. Setelah aplikasi dikembangkan, maka dilakukan proses pengujian *Black-Box* pada bagian *Admin*, *User/Customer*, dan *Seller/EO*.



BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi Sistem Evoria

Implementasi sistem Evoria merupakan proses merealisasikan desain *prototype* ke dalam *website* melalui proses *coding*. Dalam iterasi pertama dan kedua, calon pengguna mengkonfirmasi dan menyetujui *prototype* dan penerapan *coding* desain yang telah dibuat sehingga tim Axel setuju untuk melanjutkan proses implementasi aplikasi Evoria.

4.1.1 Implementasi Iterasi-1

Pada implementasi iterasi-1 tim Axel telah membuat basis data yang akan digunakan untuk menampung data-data yang dimasukkan dari aplikasi Evoria. Selain basis data tim Axel juga membuat desain *prototype*. Desain *prototype* ini digunakan untuk acuan dalam pembuatan tampilan *website* Evoria.

Basis Data

Basis data Evoria menggunakan MySQL dan diberi nama 'evoriaci'. Basis data digunakan untuk menyimpan data-data yang dimasukkan dari aplikasi Evoria. Implementasi basis data Evoria dapat dilihat pada Gambar 4.1 :

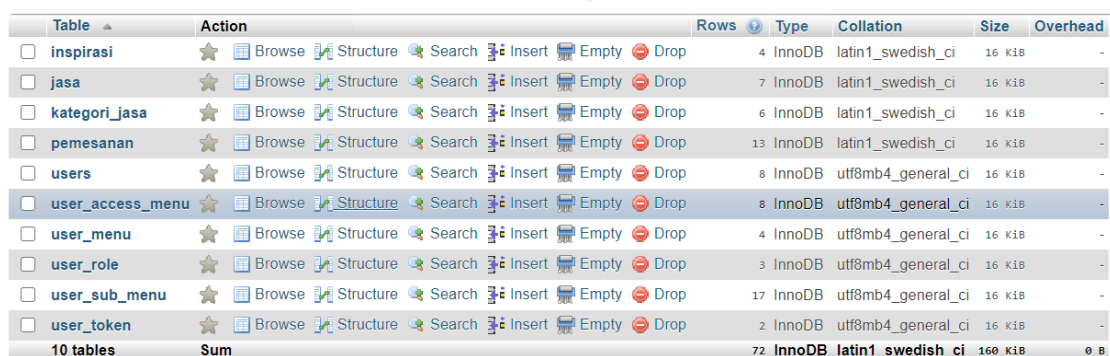


Table	Action	Rows	Type	Collation	Size	Overhead
☐ inspirasi	★ Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16 KiB	-
☐ jasa	★ Browse Structure Search Insert Empty Drop	7	InnoDB	latin1_swedish_ci	16 KiB	-
☐ kategori_jasa	★ Browse Structure Search Insert Empty Drop	6	InnoDB	latin1_swedish_ci	16 KiB	-
☐ pemesanan	★ Browse Structure Search Insert Empty Drop	13	InnoDB	latin1_swedish_ci	16 KiB	-
☐ users	★ Browse Structure Search Insert Empty Drop	8	InnoDB	utf8mb4_general_ci	16 KiB	-
☐ user_access_menu	★ Browse Structure Search Insert Empty Drop	8	InnoDB	utf8mb4_general_ci	16 KiB	-
☐ user_menu	★ Browse Structure Search Insert Empty Drop	4	InnoDB	utf8mb4_general_ci	16 KiB	-
☐ user_role	★ Browse Structure Search Insert Empty Drop	3	InnoDB	utf8mb4_general_ci	16 KiB	-
☐ user_sub_menu	★ Browse Structure Search Insert Empty Drop	17	InnoDB	utf8mb4_general_ci	16 KiB	-
☐ user_token	★ Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	16 KiB	-
10 tables	Sum	72	InnoDB	latin1_swedish_ci	160 KiB	0 B

Gambar 4.1 Layout Database Evoria

1. Tabel users

Tabel users dapat dilihat pada Gambar 4.2 :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1 id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
☐	2 name	varchar(128)	utf8mb4_general_ci		No	None			Change Drop More
☐	3 nohp	varchar(15)	utf8mb4_general_ci		No	None			Change Drop More
☐	4 email	varchar(128)	utf8mb4_general_ci		No	None			Change Drop More
☐	5 pekerjaan	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
☐	6 image	varchar(128)	utf8mb4_general_ci		No	None			Change Drop More
☐	7 password	varchar(256)	utf8mb4_general_ci		No	None			Change Drop More
☐	8 role_id	int(11)			No	None			Change Drop More
☐	9 is_active	int(1)			No	None			Change Drop More
☐	10 nama_bisnis	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
☐	11 tentang	varchar(10000)	utf8mb4_general_ci		No	None			Change Drop More
☐	12 alamat	varchar(1000)	utf8mb4_general_ci		No	None			Change Drop More
☐	13 kota	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
☐	14 medsos	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
☐	15 npwp	varchar(100)	utf8mb4_general_ci		No	None			Change Drop More
☐	16 date_created	int(11)			No	None			Change Drop More

Gambar 4.2 Layout Tabel users

2. Tabel jasa

Tabel jasa dapat dilihat pada Gambar 4.3 :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐	1 id	int(10)			No	None		AUTO_INCREMENT	Change Drop More
☐	2 nama	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
☐	3 harga	int(100)			No	None			Change Drop More
☐	4 gambar	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
☐	5 lokasi	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
☐	6 deskripsi	varchar(10000)	latin1_swedish_ci		No	None			Change Drop More
☐	7 syarat	varchar(10000)	latin1_swedish_ci		No	None			Change Drop More
☐	8 diskon	float			No	None			Change Drop More
☐	9 rating	float			No	None			Change Drop More
☐	10 id_kategori	int(11)			No	None			Change Drop More
☐	11 id_seller	int(11)			No	None			Change Drop More
☐	12 date_created	int(11)			No	None			Change Drop More

Gambar 4.3 Layout Tabel jasa

3. Tabel kategori_jasa

Tabel kategori_jasa dapat dilihat pada Gambar 4.4 :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐ 1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
☐ 2	nama	varchar(100)	latin1_swedish_ci		No	None			Change Drop More

Gambar 4.4 *Layout* Tabel kategori_jasa

4. Tabel user_access_menu

Tabel user_access_menu dapat dilihat pada Gambar 4.5 :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐ 1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
☐ 2	role_id	int(11)			No	None			Change Drop More
☐ 3	menu_id	int(11)			No	None			Change Drop More

Gambar 4.5 *Layout* Tabel user_access_menu

5. Tabel user_menu

Tabel user_menu dapat dilihat pada Gambar 4.6 :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐ 1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
☐ 2	menu	varchar(128)	utf8mb4_general_ci		No	None			Change Drop More

Gambar 4.6 *Layout* Tabel user_menu

6. Tabel user_role

Tabel user_role dapat dilihat pada Gambar 4.7 :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐ 1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
☐ 2	role	varchar(128)	utf8mb4_general_ci		No	None			Change Drop More

Gambar 4.7 *Layout* Tabel user_role

7. Tabel user_sub_menu

Tabel user_sub_menu dapat dilihat pada Gambar 4.8 :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐ 1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
☐ 2	menu_id	int(11)			No	None			Change Drop More
☐ 3	title	varchar(128)	utf8mb4_general_ci		No	None			Change Drop More
☐ 4	url	varchar(128)	utf8mb4_general_ci		No	None			Change Drop More
☐ 5	icon	varchar(128)	utf8mb4_general_ci		No	None			Change Drop More
☐ 6	is_active	int(1)			No	None			Change Drop More

Gambar 4.8 Layout Tabel user_sub_menu

8. Tabel user_token

Tabel user_token dapat dilihat pada Gambar 4.9 :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐ 1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
☐ 2	email	varchar(128)	utf8mb4_general_ci		No	None			Change Drop More
☐ 3	token	varchar(128)	utf8mb4_general_ci		No	None			Change Drop More
☐ 4	date_created	int(11)			No	None			Change Drop More

Gambar 4.9 Layout Tabel user_token

9. Tabel pemesanan

Tabel pemesanan dapat dilihat pada Gambar 4.10 :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐ 1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
☐ 2	id_jasa	int(11)			No	None			Change Drop More
☐ 3	id_seller	int(11)			No	None			Change Drop More
☐ 4	id_user	int(11)			No	None			Change Drop More
☐ 5	tgl_order	date			No	None			Change Drop More
☐ 6	tgl_acara	date			No	None			Change Drop More
☐ 7	status	int(11)			No	None			Change Drop More
☐ 8	subtotal	float			No	None			Change Drop More
☐ 9	bukti_bayar	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
☐ 10	rating	float			No	None			Change Drop More

Gambar 4.10 Layout Tabel pemesanan

10. Tabel inspirasi

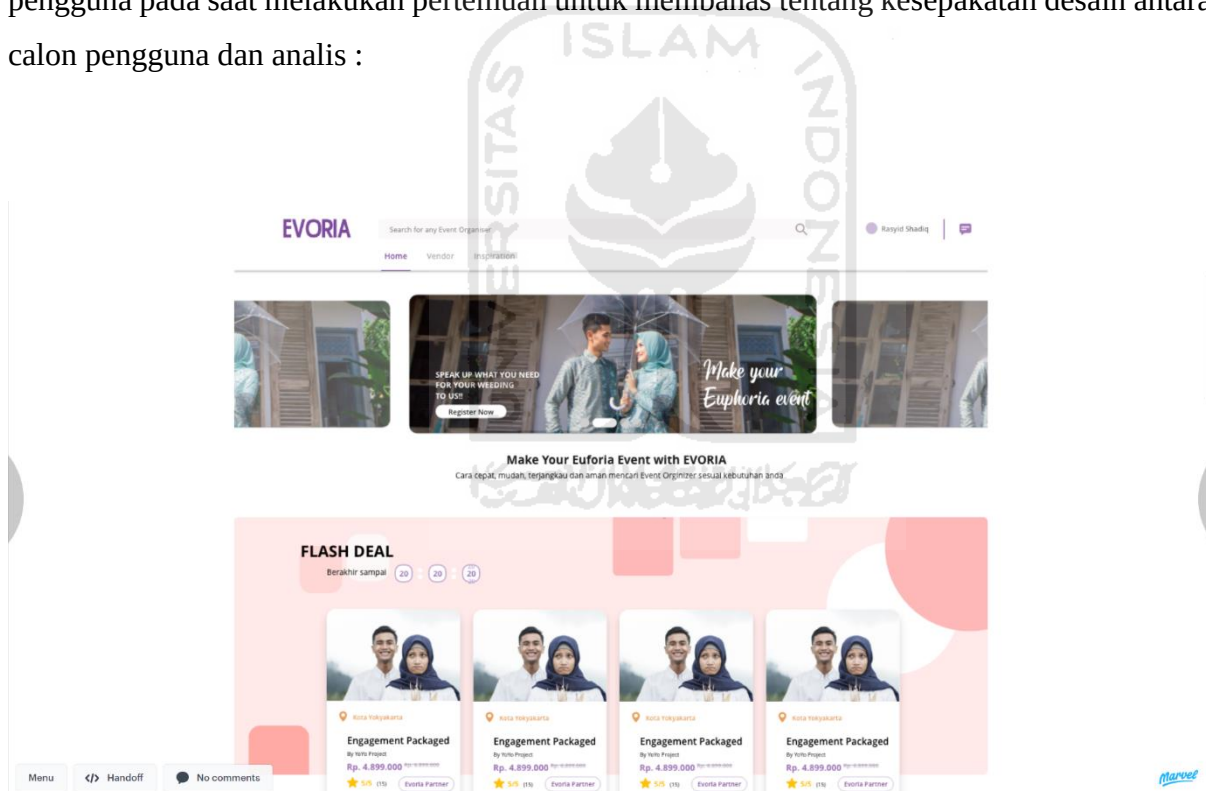
Tabel inspirasi dapat dilihat pada Gambar 4.11 :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
☐ 1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
☐ 2	id_seller	int(11)			No	None			Change Drop More
☐ 3	nama	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
☐ 4	gambar	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
☐ 5	date_created	int(11)			No	None			Change Drop More

Gambar 4.11 Layout Tabel inspirasi

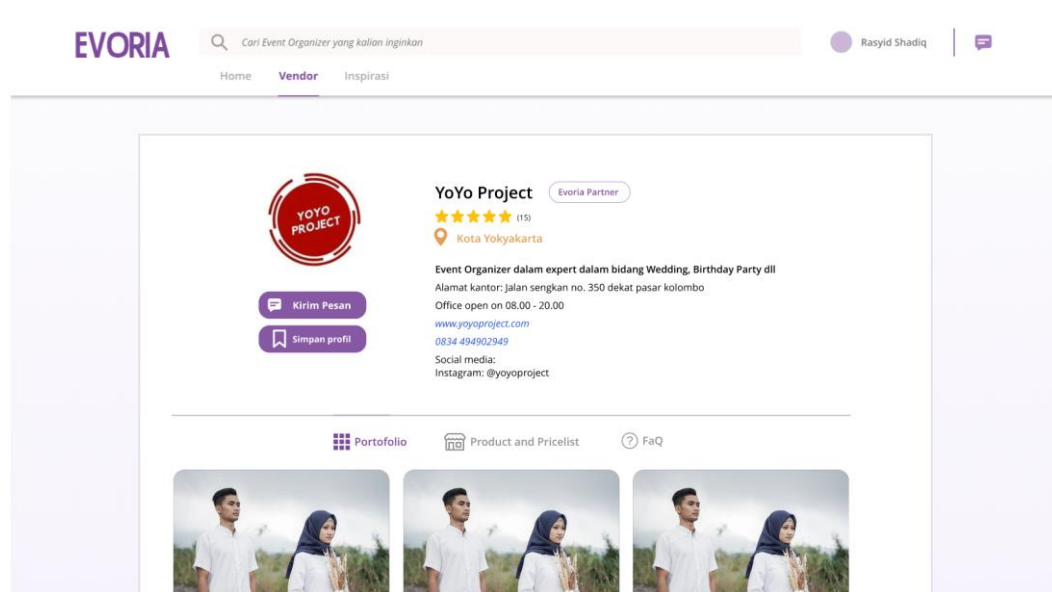
Prototype

Berikut beberapa hasil *prototype* yang telah dibuat dan ditunjukkan kepada calon pengguna pada saat melakukan pertemuan untuk membahas tentang kesepakatan desain antara calon pengguna dan analis :

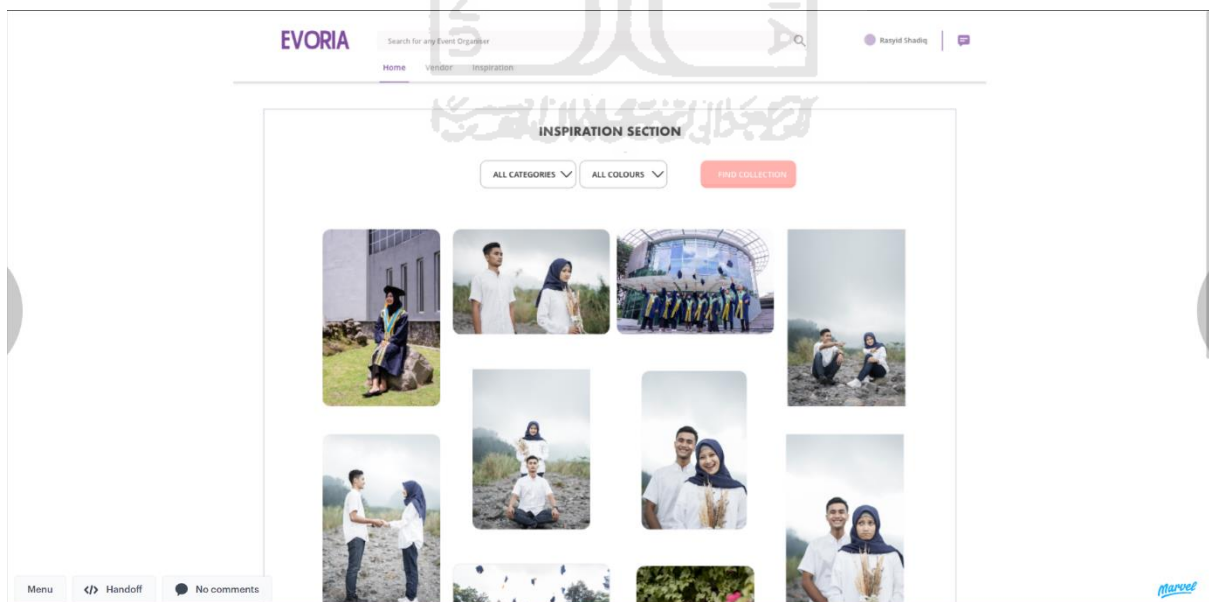


Gambar 4.12 Tampilan Beranda Evoria

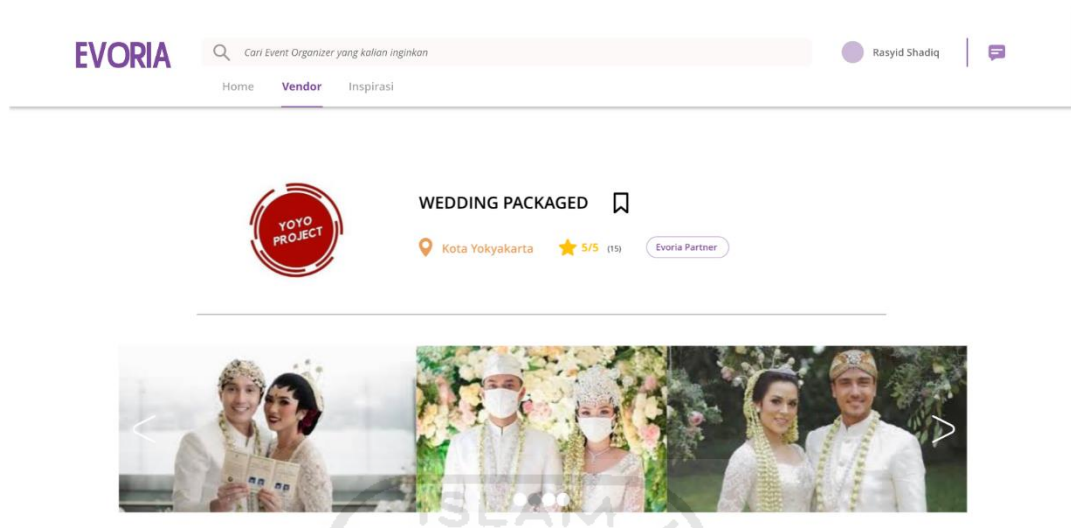
Tampilan Beranda pada Gambar 4.12 menunjukkan jasa apa saja yang ditawarkan oleh berbagai EO dan gambaran pelayanan mereka (portofolio). Selain itu *user* juga bisa mencari jenis jasa yang dibutuhkan melalui fitur pencarian yang terdapat pada *header website*.

Gambar 4.13 Tampilan *Seller/EO*

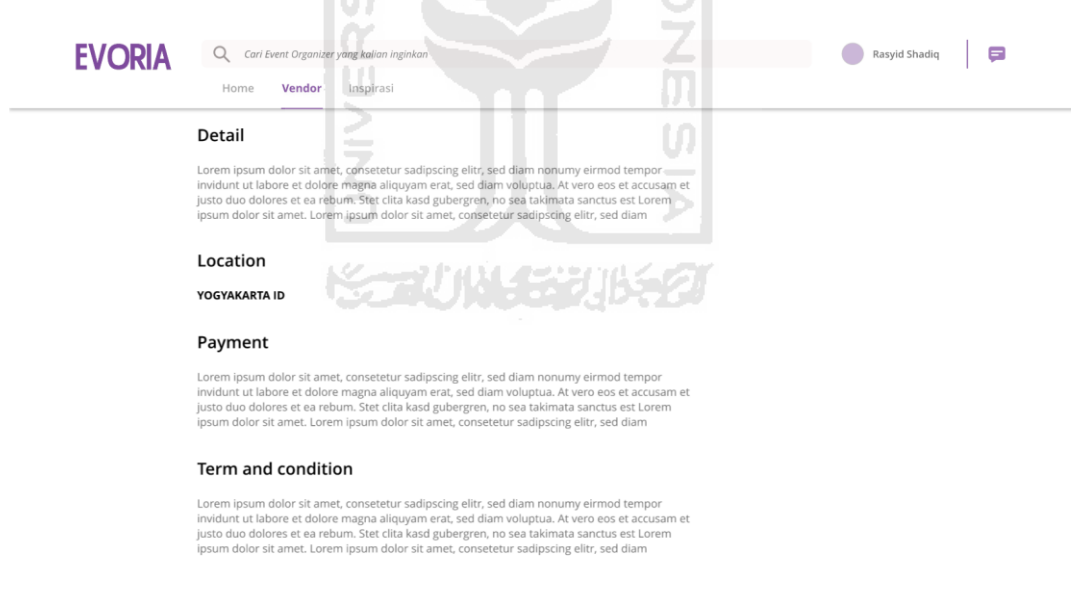
Tampilan Profil *Seller/EO* pada Gambar 4.13 menunjukkan apa saja jasa/layanan yang ditawarkan oleh sebuah EO dan portofolio yang mereka tampilkan sehingga *user* bisa menyimpulkan apakah dia akan menggunakan jasa EO tersebut atau tidak berdasarkan rating/reputasi EO tersebut.

Gambar 4.14 Tampilan *Inspiration*

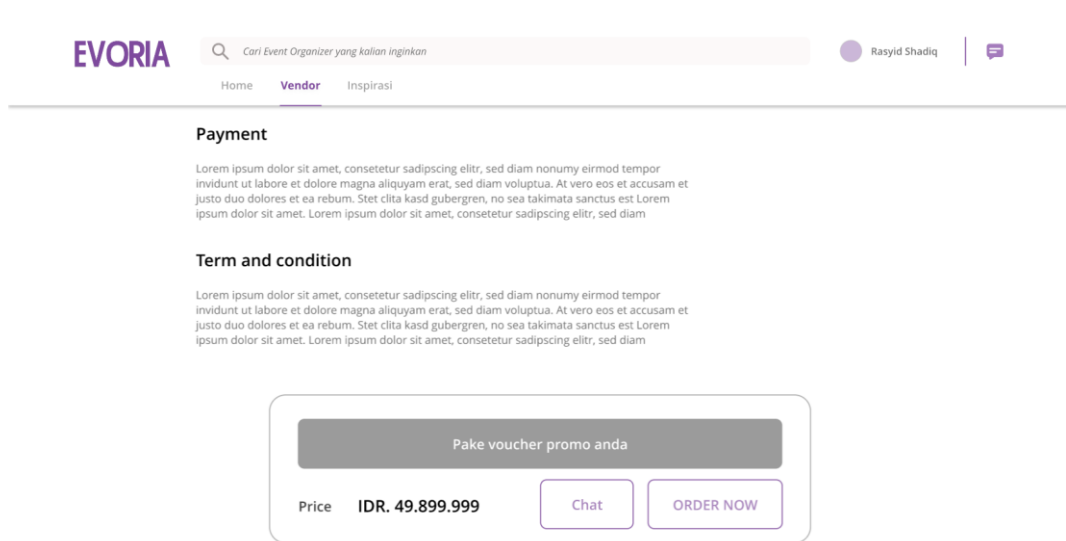
Tampilan *Inspiration* pada Gambar 4.14 menunjukkan beberapa inspirasi dan gambaran acara yang telah dikerjakan oleh EO. Berdasarkan inspirasi tersebut maka *user* dapat membayangkan konsep acara yang ditawarkan EO.



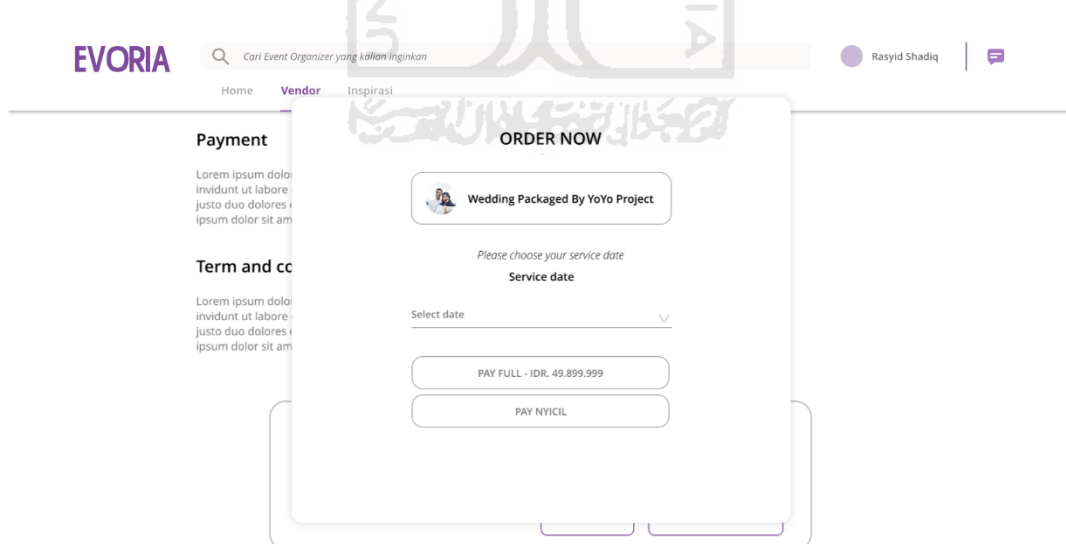
Gambar 4.15 Tampilan *Order 1*



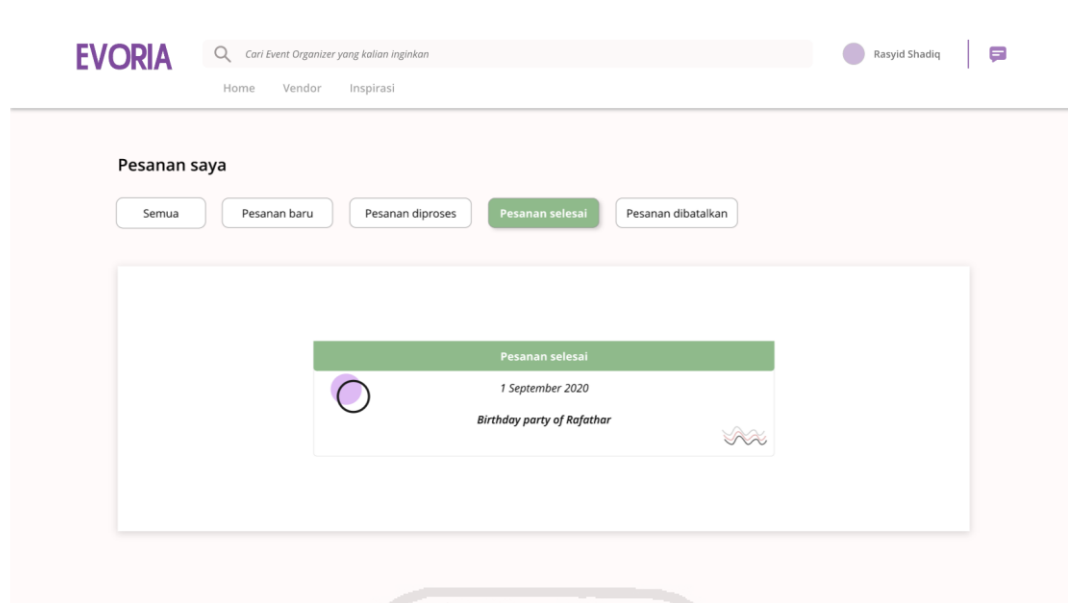
Gambar 4.16 Tampilan *Order 2*

Gambar 4.17 Tampilan *Order 3*

Tampilan *Order* pada Gambar 4.15, Gambar 4.16, dan Gambar 4.17 menunjukkan deskripsi dan detail dari pesanan yang akan di-checkout oleh user.

Gambar 4.18 Tampilan *Checkout*

Tampilan *Checkout* pada Gambar 4.18 berisi nama jasa yang dipesan, pemilihan tanggal acara, harga, serta pemilihan metode pembayaran.



Gambar 4.19 Tampilan Status Pesanan

Tampilan Status Pesanan pada Gambar 4.19 menunjukkan status dari pesanan yang sudah/sedang diproses oleh EO.

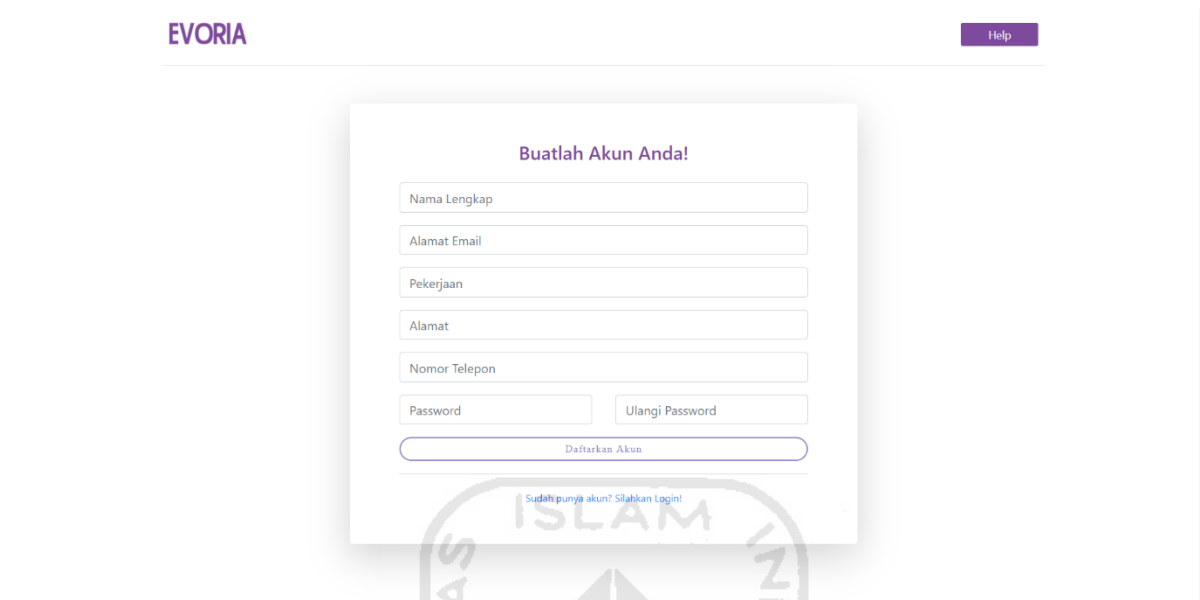
4.1.2 Implementasi Iterasi-2

Berdasarkan pengembangan aplikasi Evoria menggunakan *framework* Code Igniter dan metode *Rapid Application Development* (RAD), maka didapatkan desain antarmuka dari *website* Evoria.

Desain Antarmuka Menu User/Customer

Berikut implementasi tampilan *website* Evoria berdasarkan *prototype* yang telah dibuat dan disepakati :

a. Tampilan Register



EVORIA [Help](#)

Buatlah Akun Anda!

Nama Lengkap

Alamat Email

Pekerjaan

Alamat

Nomor Telepon

Password Ulangi Password

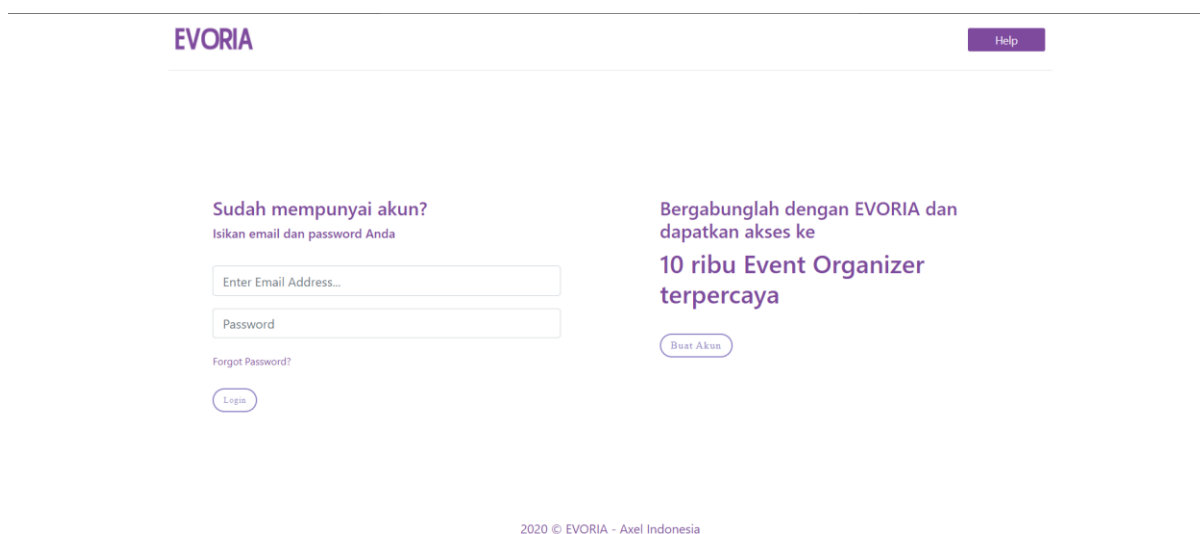
[Daftarkan Akun](#)

[Sudah punya akun? Silakan Login!](#)

Gambar 4.20 Tampilan *Register*

Tampilan *Register* pada Gambar 4.20 menampilkan *form* yang digunakan *user/customer* untuk dapat mendaftarkan akunnya ke dalam *website* Evoria dengan mengisi beberapa data diri dari *user/customer* tersebut.

b. Tampilan Login



EVORIA [Help](#)

Sudah mempunyai akun?
Isikan email dan password Anda

Enter Email Address...

Password

[Forgot Password?](#)

[Login](#)

Bergabunglah dengan EVORIA dan dapatkan akses ke 10 ribu Event Organizer terpercaya

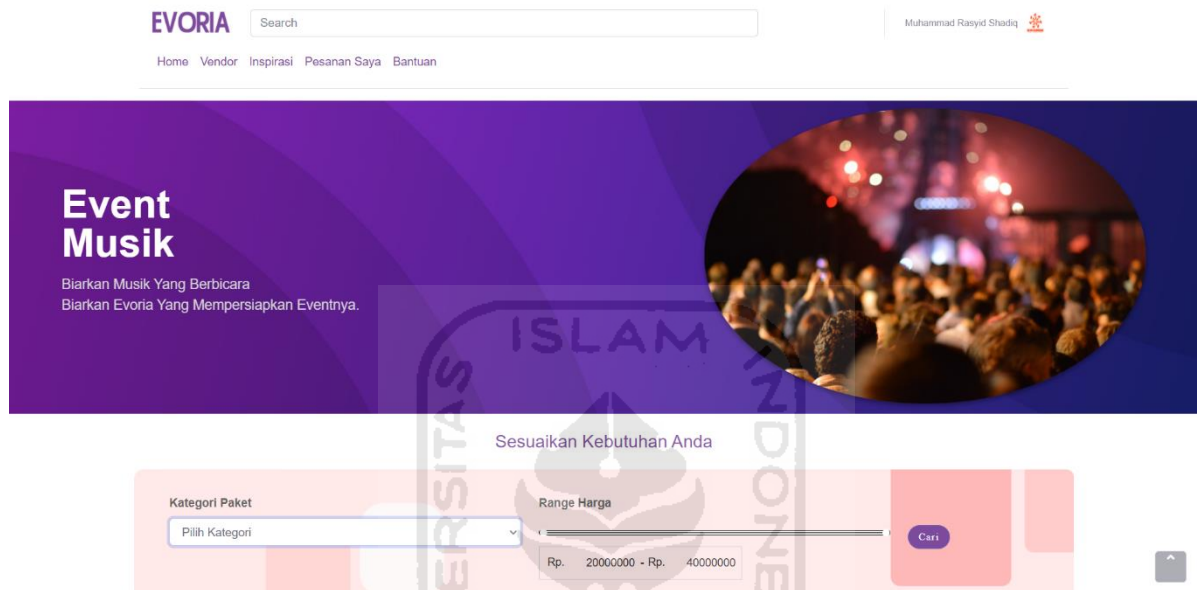
[Buat Akun](#)

2020 © EVORIA - Axel Indonesia

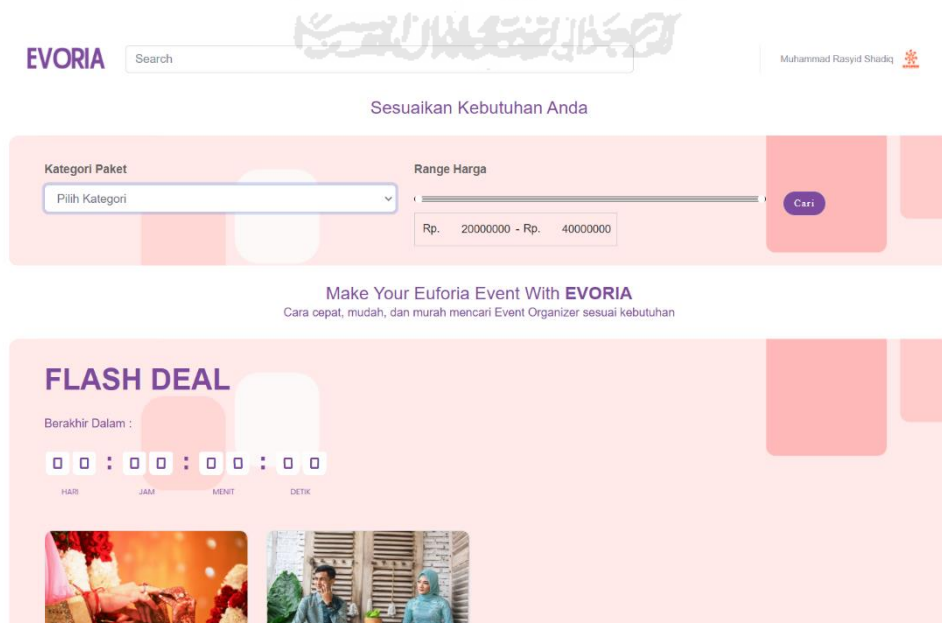
Gambar 4.21 Tampilan *Login*

Tampilan *Login* pada Gambar 4.21 menampilkan *form* yang digunakan *user/customer* untuk dapat *login* ke dalam *website* Evoria dengan mengisi *email* dan *password user/customer* tersebut.

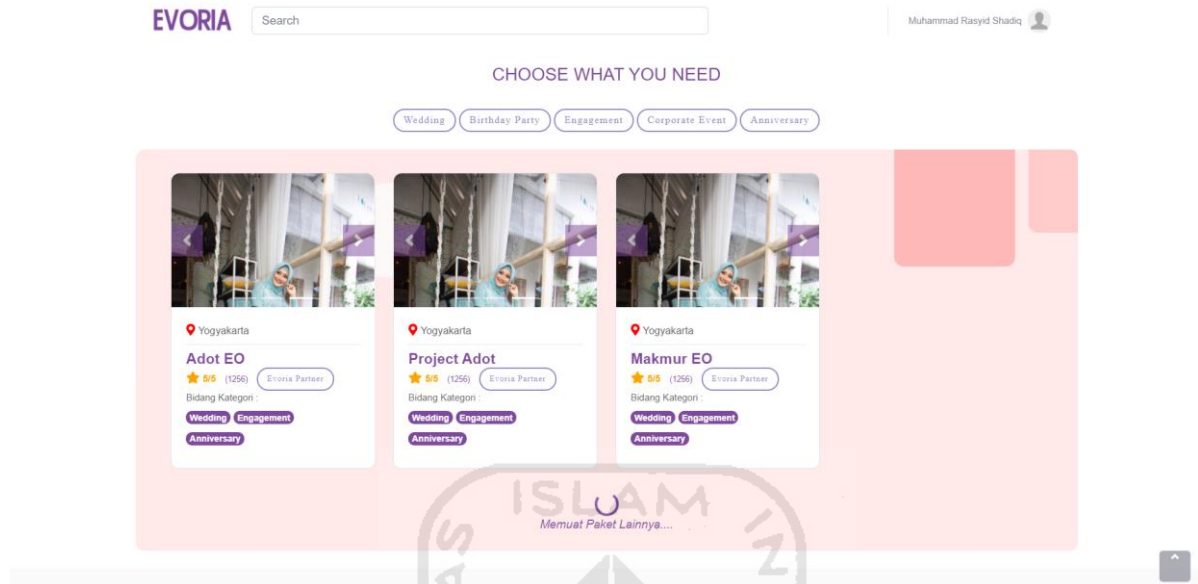
c. Tampilan Beranda



Gambar 4.22 Tampilan *Carousel* Pada Beranda



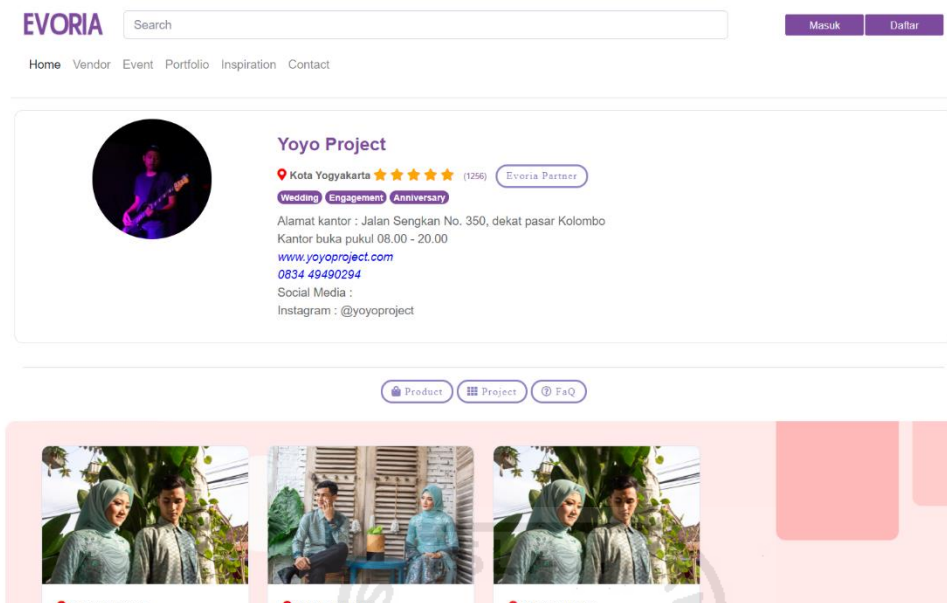
Gambar 4.23 Tampilan Pilihan Tawaran Jasa Dari EO Pada Beranda



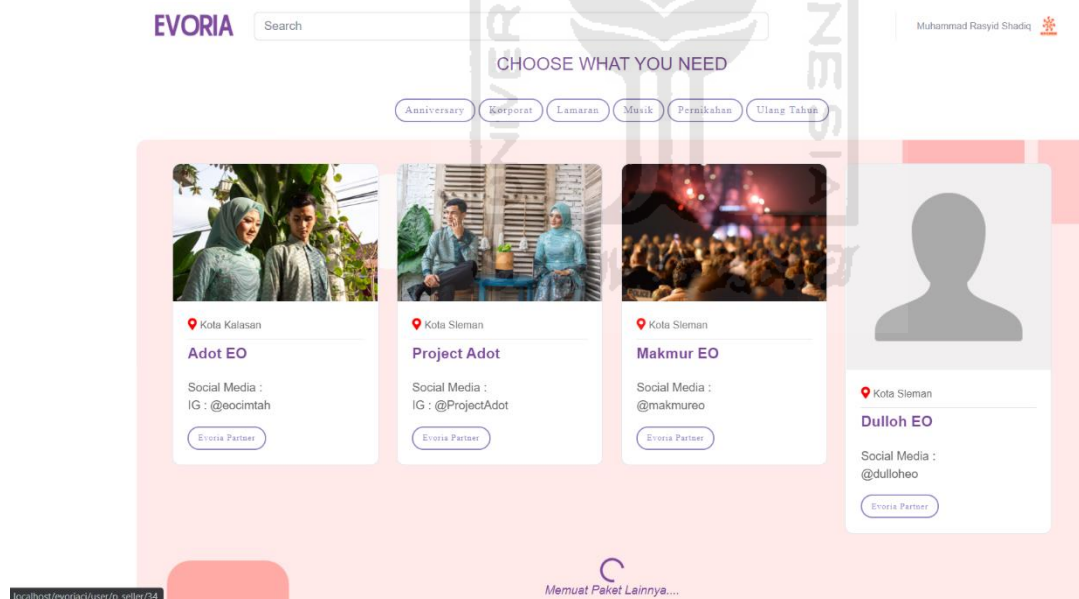
Gambar 4.24 Tampilan Pilihan Jenis EO Pada Beranda

Tampilan Beranda pada Gambar 4.22, Gambar 4.23, dan Gambar 4.24 menampilkan jasa-jasa dan portofolio yang ditawarkan oleh beberapa EO sehingga *user* bisa memilih jasa-jasa yang ditawarkan EO tersebut. Selain itu *user* juga bisa mencari jasa yang diinginkan dengan menggunakan fitur pencarian yang telah disediakan. Selain memilih jasa yang ditawarkan EO, *user* juga bisa memilih salah satu EO yang ditampilkan. Satu EO bisa menawarkan beberapa jenis jasa seperti acara pernikahan, ulang taun, acara musik, dan lain-lain.

d. Tampilan Profil Seller/EO



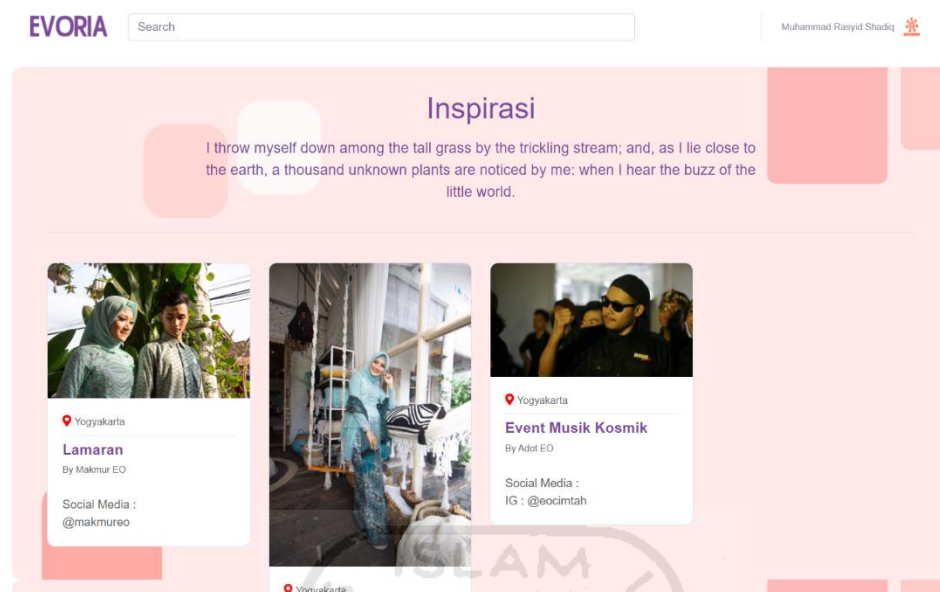
Gambar 4.25 Tampilan Profil Seller/EO 1



Gambar 4.26 Tampilan Profil Seller/EO 2

Tampilan Profil *Seller/EO* pada Gambar 4.25 dan Gambar 4.26 menampilkan apa saja jasa/layanan yang ditawarkan oleh sebuah EO dan portofolio yang mereka tampilkan sehingga *user* bisa memilih jasa/layanan yang mereka inginkan.

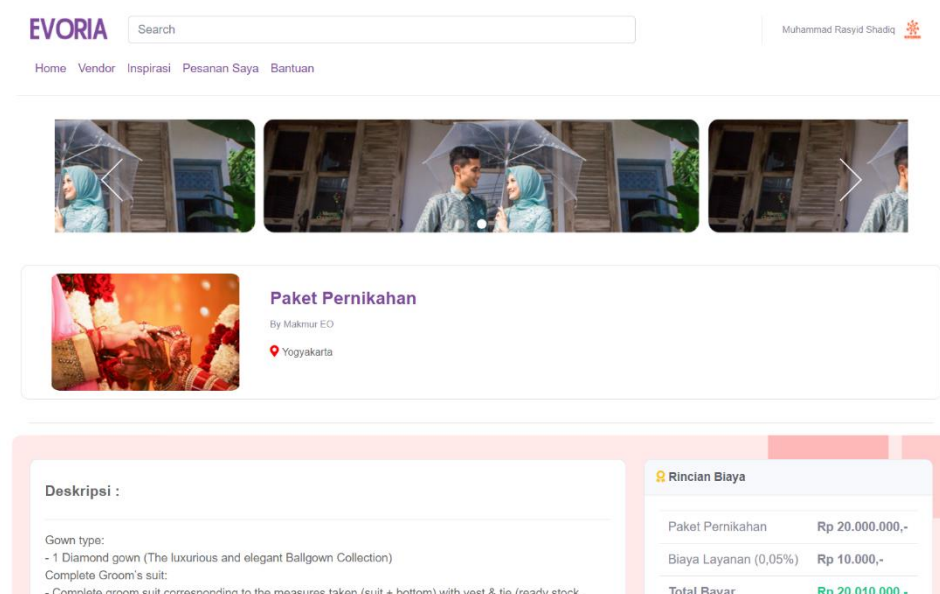
e. Tampilan Inspirasi



Gambar 4.27 Tampilan Inspirasi

Tampilan Inspirasi pada Gambar 4.27 menampilkan *form* yang berisi galeri foto yang diunggah oleh para *Seller/EO* sehingga dapat dilihat oleh *user/customer*. Berdasarkan inspirasi tersebut maka *user/customer* dapat membayangkan konsep acara yang ditawarkan EO.

f. Tampilan Order



Gambar 4.28 Tampilan Order 1

EVORIA Search Muhammad Rasyid Shadiq

Deskripsi :

Gown type:
 - 1 Diamond gown (The luxurious and elegant Ballgown Collection)
 Complete Groom's suit:
 - Complete groom suit corresponding to the measures taken (suit + bottom) with vest & tie (ready stock, rented)
 Make-up & Hair do for wedding:
 - 1x Make-up & hair-do for bride & groom at Alissha
 - 1x Test make-up only for bride at pre-wedding
 - 1x Retouch only for bride & groom for Jakarta service area (on the spot)
 - 1x Make-up & hair-do for 2 mothers at Alissha (not include touch-up/retouch)
 Pre-Wedding Photoshoot:
 - Indoor & outdoor at Alissha (package w/o additional charge for outside location)
 - Include pre-wedding make-up (as test make-up for wedding day)
 - All file photos (raw/JPEG/not yet edited)
 - Outfit for photo + 1 casual pose with 2 gowns and 1 suit
 - 25 Edited Photos
 - 20 x 30 album size (1 unit)
 - 50 x 60 canvas + photo frame (1 unit)
 - Wedding Car (driver, decoration, fuel)
 - Not include parking & toll fee, only Jakarta service area & standard service time above 6 am. (outside service area & service time, additional charge):
 - Alphard (rent 10 hours)
 Wedding Cake (dummy cake, cake table, cake knife):
 - The default package includes a standard 3 layer dummy cake of voucher/limit value IDR 1.000.000,- for only Jakarta service area (outside service area, additional charge)

Rincian Biaya

Paket Pernikahan	Rp 20.000.000,-
Biaya Layanan (0,05%)	Rp 10.000,-
Total Bayar	Rp 20.010.000,-

Pesan Sekarang

Gambar 4.29 Tampilan *Order 2*

EVORIA Search Muhammad Rasyid Shadiq

Pilih Tanggal Acara

Checkout

Tanggal Acara
 mm/dd/yyyy

Rincian Biaya

Paket Pernikahan	Rp 20.000.000,-
Biaya Layanan (0,05%)	Rp 10.000,-
Total Bayar	Rp 20.010.000,-

Biaya Layanan Evoria
 Setiap pemesanan akan dikenai biaya layanan sebesar 0,05% dari harga

Pesan Sekarang

Gambar 4.30 Tampilan *Order 3*

Tampilan *Order* pada Gambar 4.28, Gambar 4.29 dan Gambar 4.30 menampilkan *form* yang berisi tentang detail lengkap dari sebuah pesanan dan *form checkout* yang berisi tentang data pesanan, pemilihan tanggal acara, serta rincian biaya yang harus dibayar. Terdapat biaya layanan Evoria sebesar 0,05% dari harga paket.

g. Tampilan Pembayaran

EVORIA

Muhammad Rasyid Shadiq

MEMORANDUM OF UNDERSTANDING (MOU)

Yang bertanda tangan di bawah ini :

Nama	:	Fikri Almakmuri Gym
Jabatan	:	Marketing Makmur EO
Alamat	:	Yogyakarta

Yang mana selanjutnya akan disebut sebagai Pihak Pertama

Nama	:	Muhammad Rasyid Shadiq
Jabatan	:	Karyawan
Alamat	:	Yogyakarta

Yang mana selanjutnya akan disebut sebagai Pihak Kedua

Berdasarkan hal tersebut di atas maka kedua belah pihak bersepakat untuk mengadakan perjanjian kerjasama untuk waktu tertentu dengan syarat dan ketentuan sebagai berikut :

1. Pihak **Pertama** memberikan fasilitas acara yaitu :

Gown type:

Checkout

Data Pemesan

Nama: Muhammad Rasyid Shadiq
No HP: 082293641087
E-mail: rasyidshadiq97@gmail.com

Rincian Biaya

Paket Pernikahan Rp 20.010.000,-

Metode Pembayaran

Pembayaran Instan :

☐ gopay
 ☐ jenius pay
 ☐ DANA

Transfer Bank :

☐ BCA
 ☐ BNI
 ☐ mandiri

Gambar 4.31 Tampilan Pembayaran 1

EVORIA

Muhammad Rasyid Shadiq

Paket Pernikahan

Rp 20.010.000,-

BCA

No. Rekening:

126 081262406367

Dicek dalam 10 menit

Petunjuk Transfer melalui Mesin ATM

- Masukkan kartu ATM ke tempat yang tersedia.
- Selanjutnya, pilih bahasa yang ingin Anda gunakan. Untuk mempermudah prosesnya, Anda bisa memilih bahasa Indonesia. Tunggu sesaat hingga muncul tampilan menu baru. Setelah muncul, klik pilihan LANJUTKAN yang ada pada bagian pojok bawah layar.
- Masukkan PIN ATM
- Pilih Transaksi lainnya
- Pilih transfer
- Masukkan kode bank dan nomor rekening tujuan
- Masukkan jumlah nominal uang
- Masukkan nomor referensi. Jika tidak ada nomor referensi tertentu, Anda bisa lewat langkah ini.
- Terakhir jangan lupa ambil Kartu ATM Anda

Petunjuk Transfer melalui Mobile Banking

Saya Sudah Bayar

Yang mana selanjutnya akan disebut sebagai Pihak Kedua

Berdasarkan hal tersebut di atas maka kedua belah pihak bersepakat untuk mengadakan perjanjian kerjasama untuk waktu tertentu dengan syarat dan ketentuan sebagai berikut :

1. Pihak **Pertama** memberikan fasilitas acara yaitu :

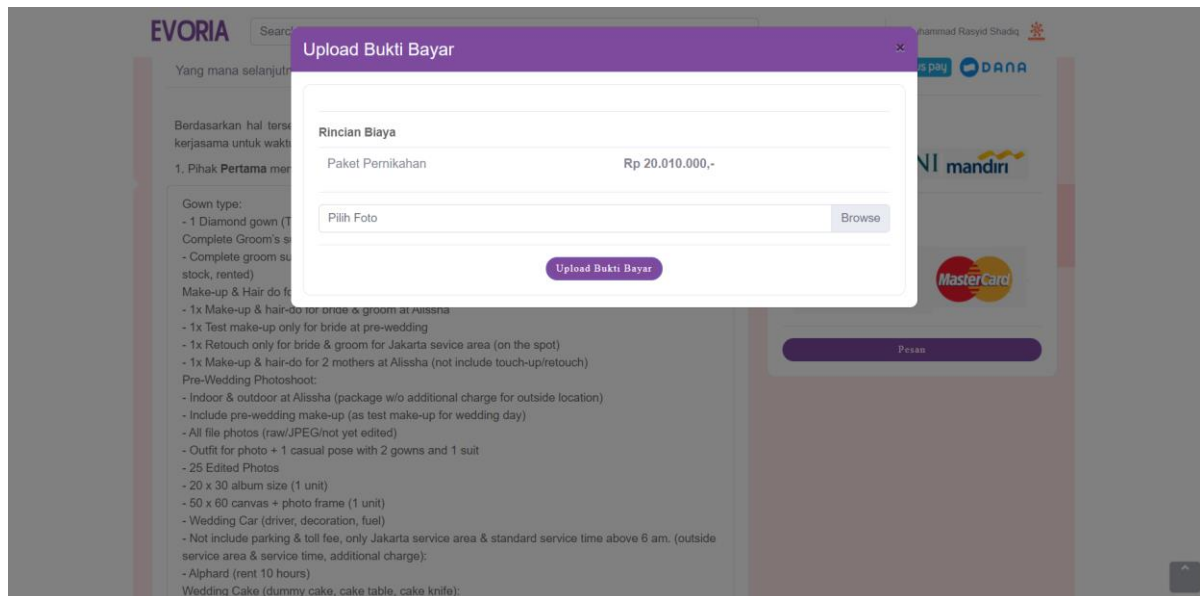
Gown type:

- 1 Diamond gown (T)
- Complete Groom's suit
- Complete groom's stock, rented)
- Make-up & Hair do for
- 1x Make-up & hair-
- 1x Test make-up on
- 1x Retouch only for
- 1x Make-up & hair-
- Pre-Wedding Photos
- Indoor & outdoor at
- Include pre-wedding
- All file photos (raw)
- Outfit for photo + 1
- 25 Edited Photos
- 20 x 30 album size
- 50 x 60 canvas + pl
- Wedding Car (drive
- Not include parking

service area & service time, additional charge):

- Alphard (rent 10 hours)
- Wedding Cake (dummy cake, cake table, cake knife):

Gambar 4.32 Tampilan Pembayaran 2



Gambar 4.33 Tampilan Pembayaran 3

Tampilan Pembayaran pada Gambar 4.31, Gambar 4.32, dan Gambar 4.33 menampilkan *form* yang berisi MOU, metode pembayaran, serta *upload* bukti bayar/transfer pemesanan.

h. Tampilan Pesanan Saya

Menunggu Konfirmasi						
No	ID Pesanan	Nama Pesanan	Tanggal Pesan	Tanggal Acara	Status	Action
1	31	Paket Musik	2020-10-19	2020-10-31	Menunggu Konfirmasi	Batalkan
2	29	Paket Musik	2020-10-16	2020-10-31	Menunggu Konfirmasi	Batalkan
3	28	Paket Musik	2020-10-16	2020-10-24	Menunggu Konfirmasi	Batalkan
4	30	Paket Musik A	2020-10-19	2020-11-19	Menunggu Konfirmasi	Batalkan

Menunggu Pembayaran						
No	ID Pesanan	Nama Pesanan	Tanggal Pesan	Tanggal Acara	Status	Action
1	32	Paket Pernikahan	2020-10-19	2020-11-19	Menunggu Pembayaran	Lakukan Pembayaran

Pesanan Diproses						
No	ID Pesanan	Nama Pesanan	Tanggal Pesan	Tanggal Acara	Status	Action

Gambar 4.34 Tampilan Pesanan Saya 1

EVORIA

Search

Muhammad Yasin Deru Saputra

Pesanan Diproses

No	ID Pesanan	Nama Pesanan	Tanggal Pesan	Tanggal Acara	Status	Action

Pesanan Selesai

No	ID Pesanan	Nama Pesanan	Tanggal Pesan	Tanggal Acara	Status	Action
1	26	Paket Pernikahan	2020-10-16	2020-10-23	Pesanan Selesai	

Pesanan Dibatalkan

No	ID Pesanan	Nama Pesanan	Tanggal Pesan	Tanggal Acara	Status	Action
1	27	Paket Musik A	2020-10-16	2020-10-30	Pesanan Dibatalkan	

2020 © EVORIA - Axel Indonesia

Gambar 4.35 Tampilan Pesanan Saya 2

Tampilan Pesanan Saya pada Gambar 4.34, dan Gambar 4.35 menampilkan daftar tabel yang berisi status pesanan (*status tracking*).

Desain Antarmuka Menu Seller/EO

Berikut implementasi tampilan *website* Evoria berdasarkan *prototype* yang telah dibuat dan disepakati :

a. Tampilan Register

EVORIA
Help

Daftar Akun Bisnis

Data Pribadi Pemilik Bisnis

Data Tentang Bisnis Anda

Isikan Nama Lengkap Anda

Isikan Nama Bisnis Anda

Nomor Telepon Bisnis

Tentang Bisnis Anda

Isikan Jabatan Pekerjaan

Alamat Bisnis Anda

Isikan Email Bisnis Anda

Kota

Isikan Password Anda

Media Sosial

Unggah Dokumen NPWP

Gambar 4.36 Tampilan Register Seller/EO 1

EVORIA Help

Informasi Kontak

Nomor Telepon

Isikan Jabatan Pekerjaan

Masukkan Jabatan Pekerjaan

Isikan Email Bisnis Anda

Masukkan Email Anda

Isikan Password Anda

Password Ulangi Password

Informasi Bisnis

Ceritakan tentang bisnis Anda...

Alamat Bisnis Anda

Alamat bisnis Anda...

Kota

Kota

Media Sosial

Instagram : @instagrambisnis

Unggah Dokumen NPWP

Pilih Dokumen PDF Browse

[Daftar](#)

2020 © EVORIA - Axel Indonesia

Gambar 4.37 Tampilan *Register Seller/EO* 2

Tampilan *Register Seller/EO* pada Gambar 4.36 dan Gambar 4.37 menampilkan *form* yang digunakan *Seller/EO* untuk dapat mendaftarkan akun untuk bisnisnya ke dalam *website* Evoria dengan mengisi beberapa data diri serta data tentang bisnis dari *Seller/EO* tersebut.

b. Tampilan Login

EVORIA Help

Sudah mempunyai akun bisnis?

Isikan email dan password Anda

Enter Email Address...

Password

[Forgot Password?](#)

[Login](#)

Belum mempunyai akun bisnis?

Bergabunglah dengan EVORIA
Gratis!

[Buat Akun](#)

Melihat Analisa

Akan memiliki akses melihat analisa dan statistik kunjungan bisnis profil dari konsumen

Gratis Pendaftaran

Untuk semua Event organizer yang ingin mendaftarkan vendor mereka tidak akan dipungut biaya

Banyak bidang

Kita tidak hanya bergerak dalam satu bidang seperti wedding saja akan tetapi bisa untuk banyak bidang dalam perjalanan event organizer

Transparansi Harga

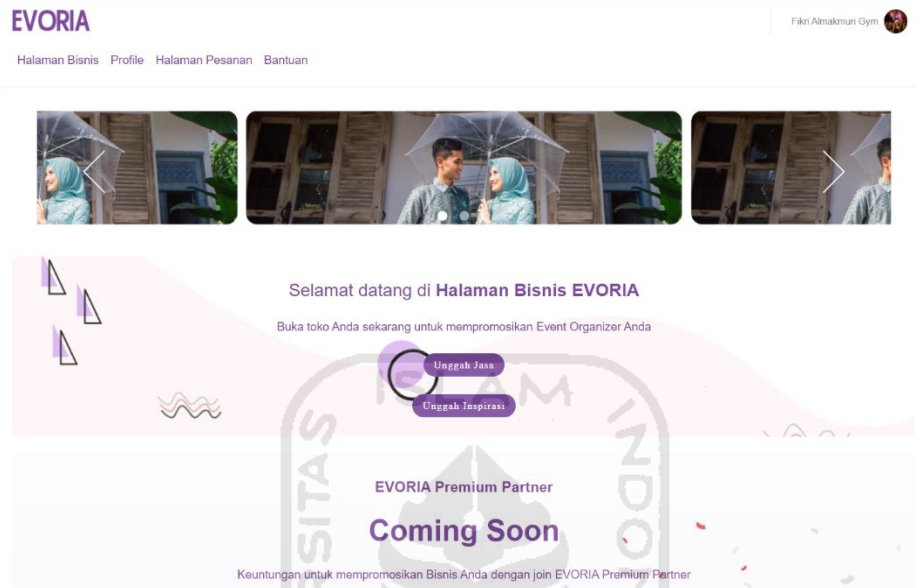
Dapat melihat harga-harga yang ditawarkan tiap-tiap vendor dan bisa nego sampai deal antara dua pihak.

2020 © EVORIA - Axel Indonesia

Gambar 4.38 Tampilan *Login Seller/EO*

Tampilan *Login Seller/EO* pada Gambar 4.38 menampilkan *form* yang digunakan *Seller/EO* untuk dapat login ke dalam *website* Evoria dengan mengisi *email* dan *password Seller/EO* tersebut.

c. Tampilan Halaman Bisnis



Gambar 4.39 Tampilan Halaman Bisnis

Tampilan Halaman Bisnis pada Gambar 4.39 berisi sambutan selamat datang kepada *Seller/EO*, tombol Unggah Jasa yang mengarahkan pada halaman Unggah Jasa, dan tombol Unggah Inspirasi yang mengarahkan pada halaman Unggah Inspirasi.

d. Tampilan Unggah Jasa

EVORIA Search

Fikri Almakmuri

Tambah Jasa

Pilih Foto Browse

Projects

Project	Progress
Server migration	20%
Sales tracking	40%
Customer Database	60%
Payout Details	80%
Account setup	Complete!

Data Paket

Nama Paket: Contoh : Paket Pernikahan Kategori Paket: Pilih Kategori

Lokasi Layanan: Lokasi Layanan Harga Paket: Harga Paket

Deskripsi Isi Paket: Contoh : 1. Dekorasi, 2. Vendor Makanan, 3. MUA, dll

Syarat & Ketentuan: Jelaskan syarat & ketentuan untuk paket jasa ini

Tambah Jasa

Gambar 4.40 Tampilan Unggah Jasa

Tampilan Unggah Jasa pada Gambar 4.40 berisi *form* untuk menambahkan jasa yang dapat dilihat dan dipilih *user/customer*.

e. Tampilan Unggah Inspirasi

EVORIA Search

Fikri Almakmuri

Tambah Inspirasi

Pilih Foto Browse

Projects

Project	Progress
Server migration	20%
Sales tracking	40%
Customer Database	60%
Payout Details	80%
Account setup	Complete!

Data Paket

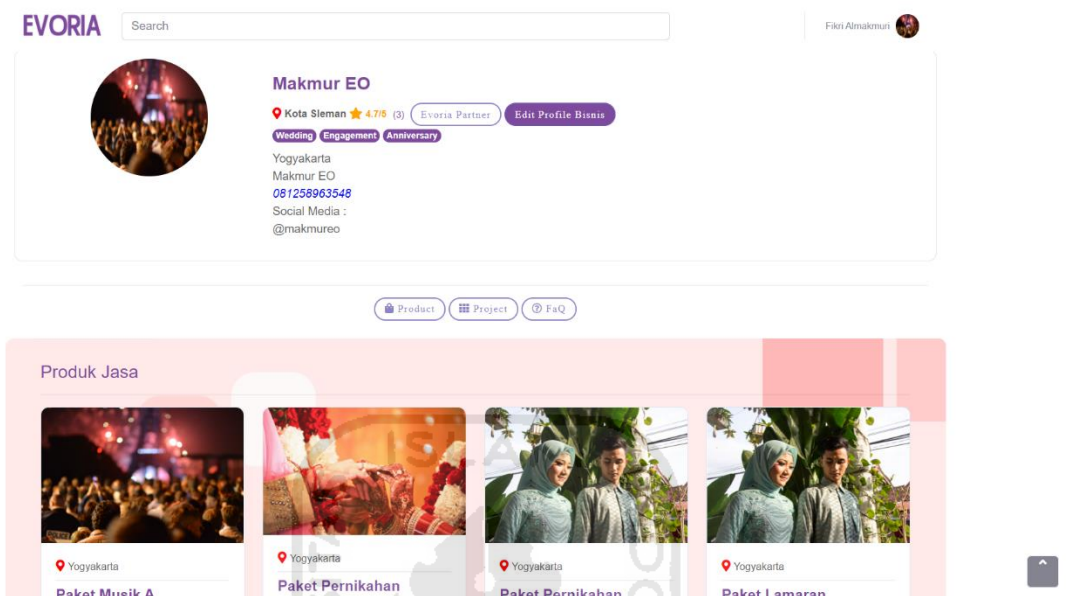
Nama Inspirasi: Nama Inspirasi

Tambah Inspirasi

Gambar 4.41 Tampilan Unggah Inspirasi

Tampilan Unggah Jasa pada Gambar 4.41 berisi *form* untuk menambahkan inspirasi yang dapat dilihat pada halaman Inspirasi.

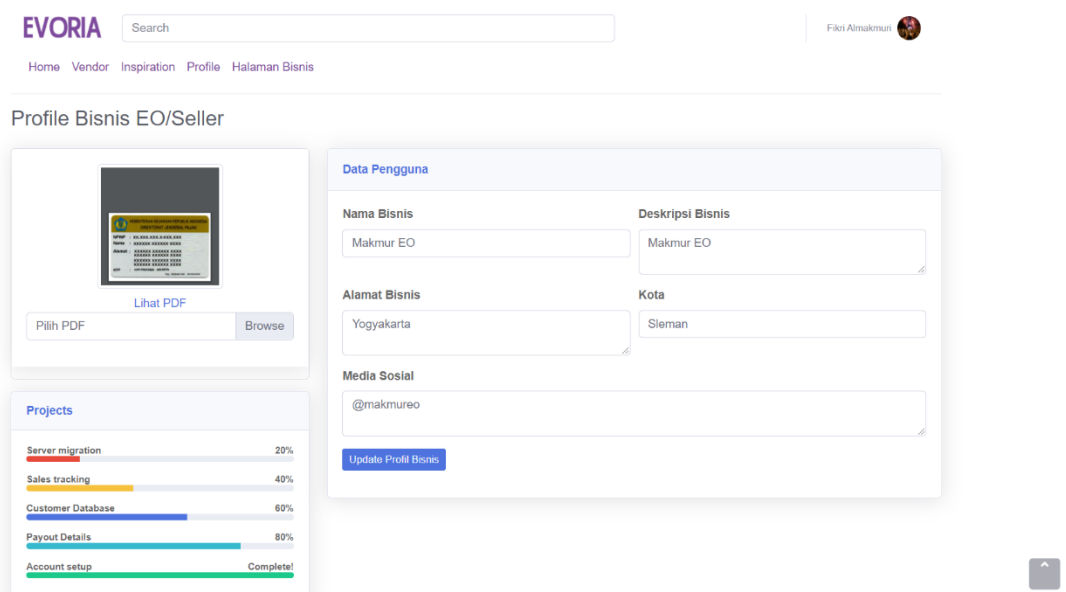
f. Tampilan Profil Bisnis Seller/EO



Gambar 4.42 Tampilan Profil Bisnis Seller/EO

Tampilan Profil Bisnis Seller/EO pada Gambar 4.42 berisi portofolio, foto-foto inspirasi dan jasa/layanan yang sudah ditambahkan oleh sebuah EO sehingga *user/customer* bisa memilih jasa/layanan yang mereka inginkan.

g. Tampilan Edit Profil Bisnis Seller/EO



Gambar 4.43 Tampilan Edit Profil Bisnis Seller/EO

Tampilan Edit Profil Bisnis *Seller/EO* pada Gambar 4.43 berisi *form* edit profil bisnis dari *Seller/EO* yang sedang *login* ke dalam aplikasi Evoria.

h. Tampilan Pesanan

EVORIA

Home Vendor Inspiration Pesanan Saya Profile Halaman Bisnis Halaman Pesanan

Fikri Almakmuri

Menunggu Konfirmasi

No	ID Pesanan	Nama Pesanan	Nama Pemesan	Tanggal Pesan	Tanggal Acara	Status	Bukti Pembayaran	Action
1	33	Paket Musik A	Muhammad Rasyid Shadiq	2020-10-20	2020-11-28	Menunggu Konfirmasi		Konfirmasi Batalan

Menunggu Pembayaran

No	ID Pesanan	Nama Pesanan	Nama Pemesan	Tanggal Pesan	Tanggal Acara	Status	Bukti Pembayaran	Action
1	30	Paket Musik A	Muhammad Yasin Deru Saputra	2020-10-19	2020-11-19	Menunggu Pembayaran		

Pesanan Diproses

No	ID Pesanan	Nama Pesanan	Nama Pemesan	Tanggal Pesan	Tanggal Acara	Status	Bukti Pembayaran	Action
1	32	Paket	Muhammad Yasin Deru Saputra	2020-10-19	2020-11-19	Menunggu Konfirmasi		

Gambar 4.44 Tampilan Pesanan 1

EVORIA

Fikri Almakmuri

Pesanan Diproses

No	ID Pesanan	Nama Pesanan	Nama Pemesan	Tanggal Pesan	Tanggal Acara	Status	Bukti Pembayaran	Action
1	32	Paket Pernikahan	Muhammad Yasin Deru Saputra	2020-10-19	2020-11-19	Menunggu Konfirmasi Selesai		

Pesanan Selesai

No	ID Pesanan	Nama Pesanan	Nama Pemesan	Tanggal Pesan	Tanggal Acara	Status	Bukti Pembayaran	Action
1	25	Paket Musik A	Muhammad Rasyid Shadiq	2020-10-16	2020-10-24	Pesanan Selesai		

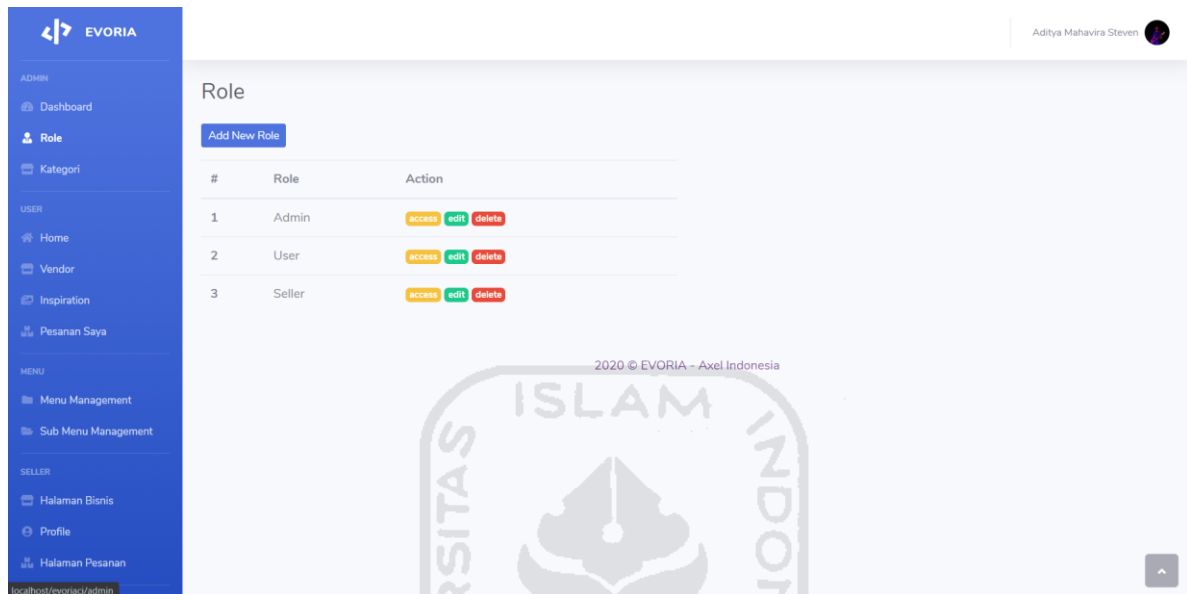
Gambar 4.45 Tampilan Pesanan 2

Tampilan Pesanan pada Gambar 4.44 dan Gambar 4.45 menampilkan daftar tabel yang berisi status pesanan (*status tracking*).

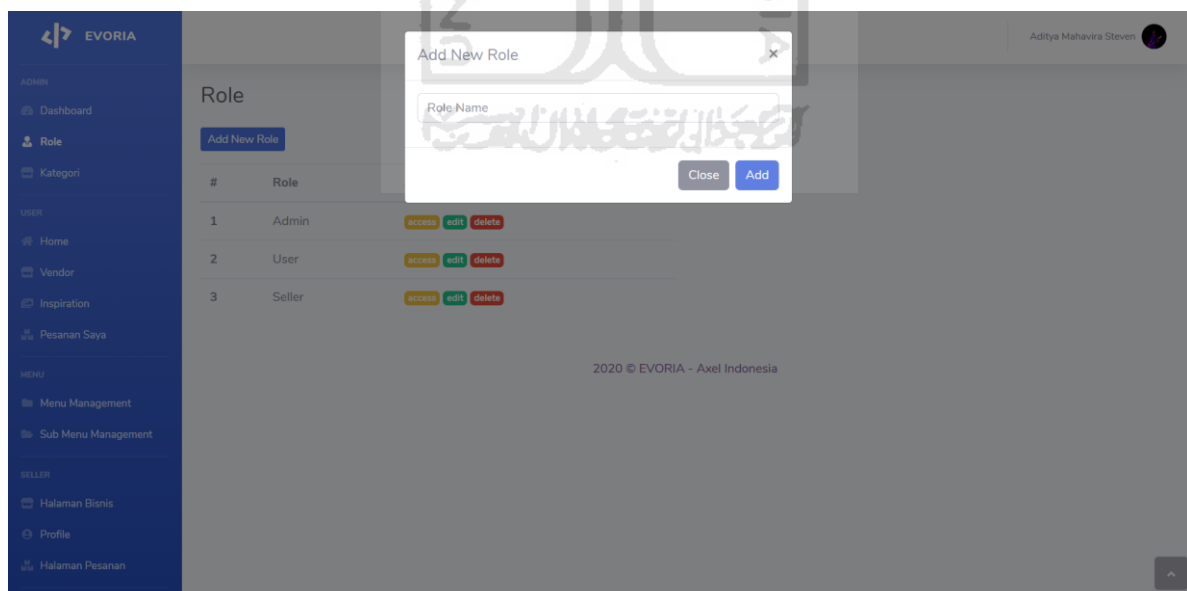
Desain Antarmuka Menu Admin

Berikut implementasi tampilan *website* Evoria yang telah dibuat :

a. Tampilan Role Management



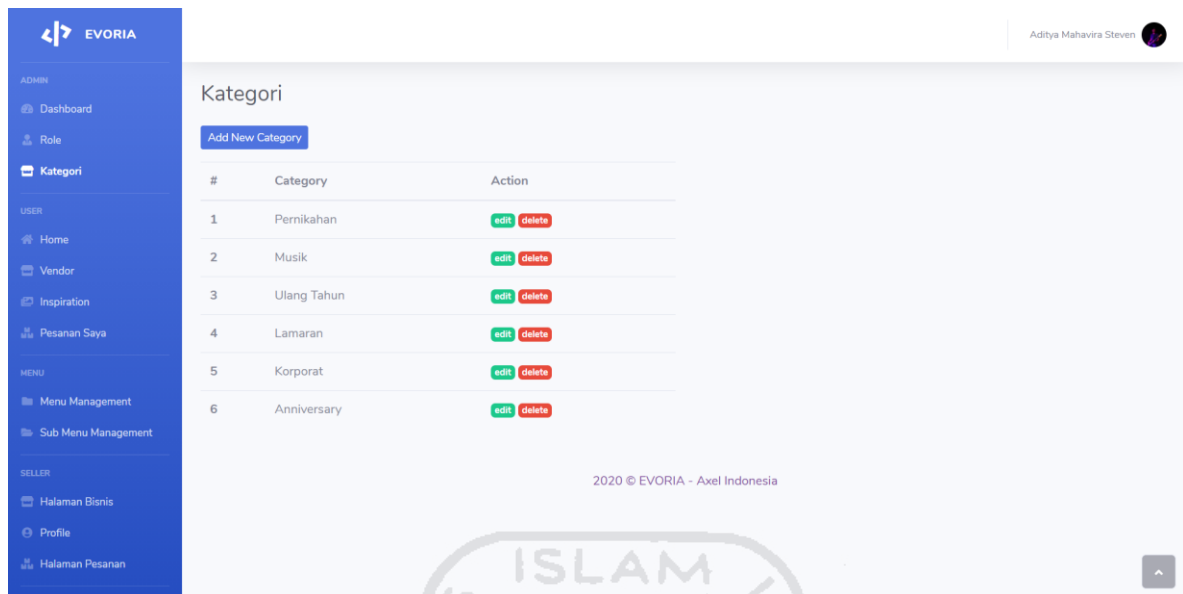
Gambar 4.46 Tampilan *Role*



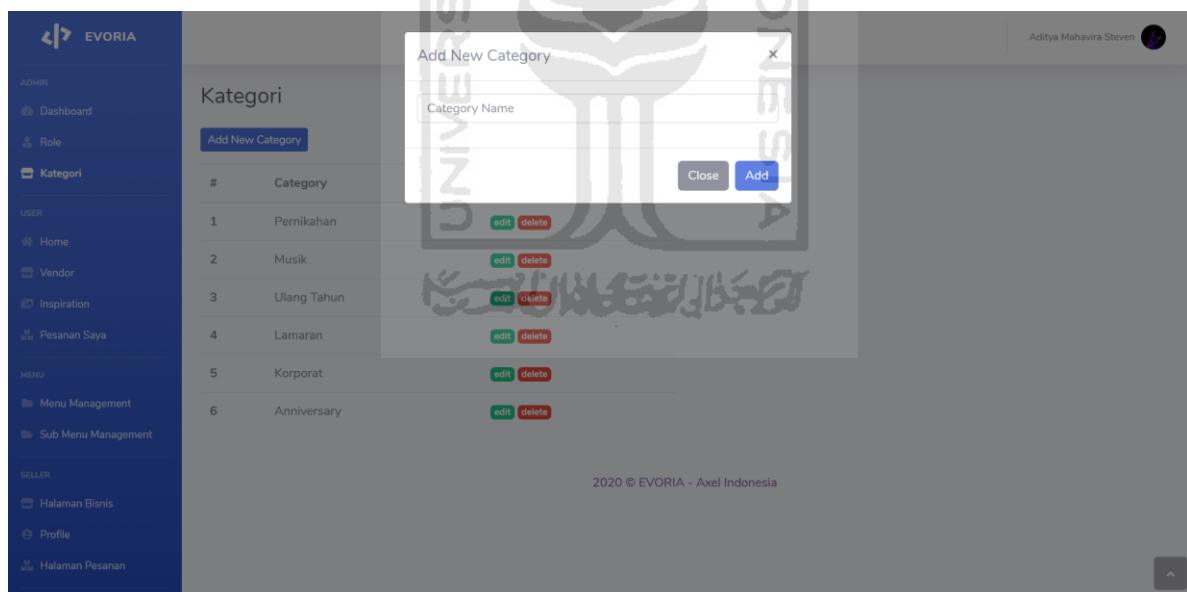
Gambar 4.47 Tampilan Tambah *Role*

Tampilan *Role* pada Gambar 4.46 dan Gambar 4.47 menampilkan daftar *Role* pengguna yang dapat diatur hak aksesnya.

b. Tampilan Kategori Management



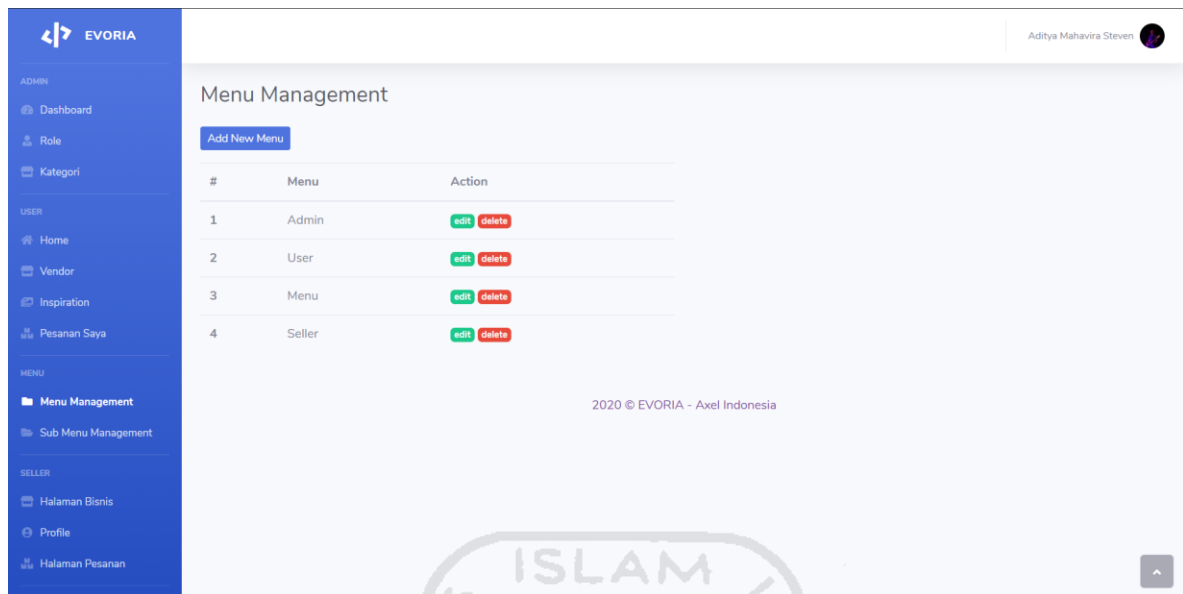
Gambar 4.48 Tampilan Kategori



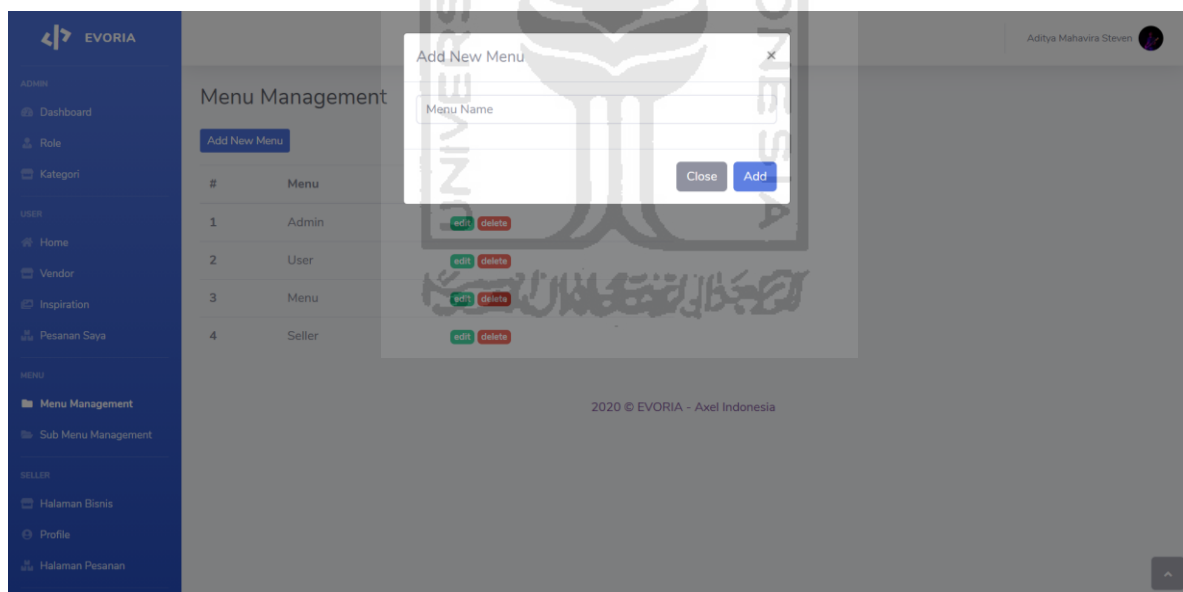
Gambar 4.49 Tampilan Tambah Kategori

Tampilan Kategori pada Gambar 4.48 dan Gambar 4.49 menampilkan daftar Kategori yang akan ditampilkan pada tampilan Tambah Jasa.

c. Tampilan Menu Management



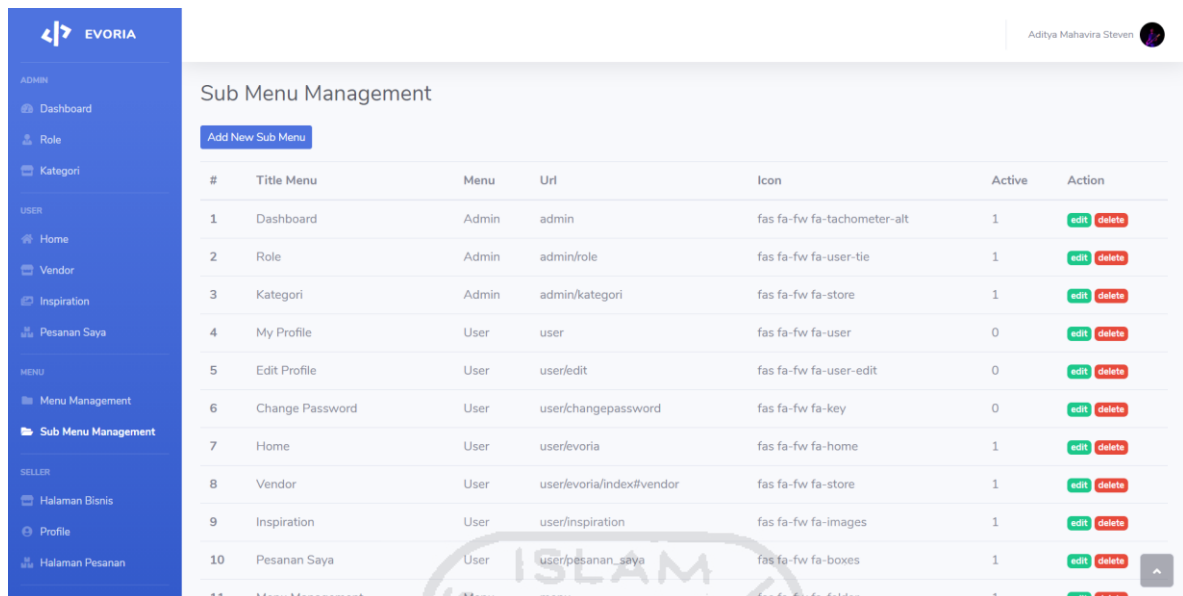
Gambar 4.50 Tampilan Menu Management



Gambar 4.51 Tampilan Tambah Menu

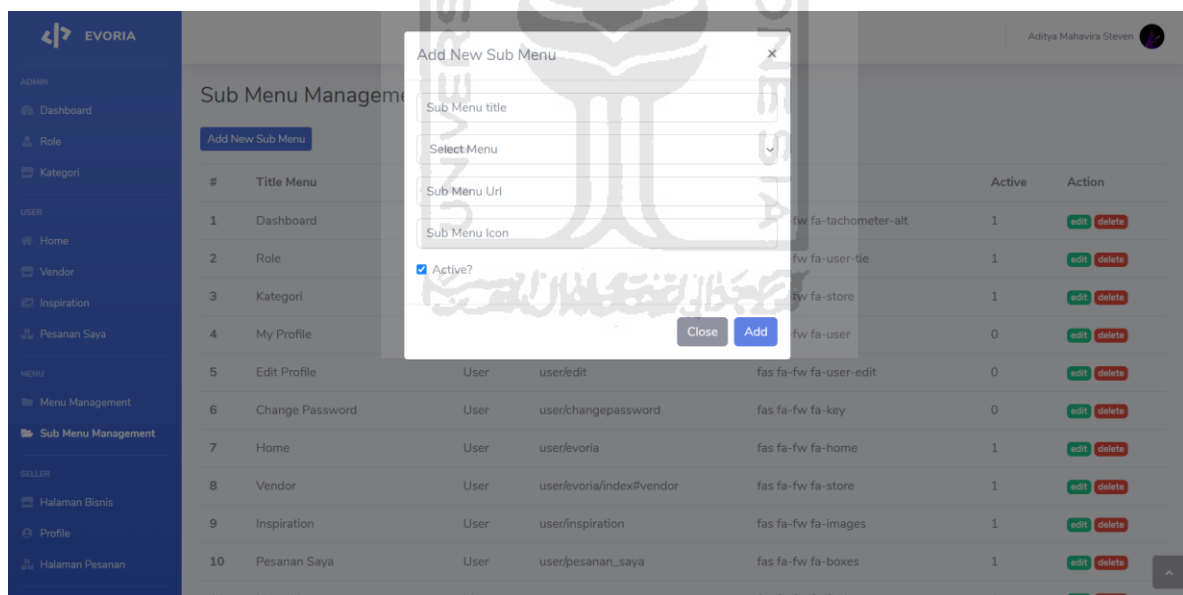
Tampilan Menu Management pada Gambar 4.50 dan Gambar 4.51 menampilkan daftar Menu yang telah ditambahkan.

d. Tampilan Submenu Management



#	Title Menu	Menu	Url	Icon	Active	Action
1	Dashboard	Admin	admin	fas fa-fw fa-tachometer-alt	1	edit delete
2	Role	Admin	admin/role	fas fa-fw fa-user-tie	1	edit delete
3	Kategori	Admin	admin/kategori	fas fa-fw fa-store	1	edit delete
4	My Profile	User	user	fas fa-fw fa-user	0	edit delete
5	Edit Profile	User	user/edit	fas fa-fw fa-user-edit	0	edit delete
6	Change Password	User	user/changepassword	fas fa-fw fa-key	0	edit delete
7	Home	User	user/evoria	fas fa-fw fa-home	1	edit delete
8	Vendor	User	user/evoria/index#vendor	fas fa-fw fa-store	1	edit delete
9	Inspiration	User	user/inspiration	fas fa-fw fa-images	1	edit delete
10	Pesanan Saya	User	user/pesanan_saya	fas fa-fw fa-boxes	1	edit delete
11	Menu Management	Menu	menu	fas fa-fw fa-folder	1	edit delete

Gambar 4.52 Tampilan Submenu Management



#	Title Menu	Menu	Url	Icon	Active	Action
1	Dashboard	Admin	admin	fas fa-fw fa-tachometer-alt	1	edit delete
2	Role	Admin	admin/role	fas fa-fw fa-user-tie	1	edit delete
3	Kategori	Admin	admin/kategori	fas fa-fw fa-store	1	edit delete
4	My Profile	User	user	fas fa-fw fa-user	0	edit delete
5	Edit Profile	User	user/edit	fas fa-fw fa-user-edit	0	edit delete
6	Change Password	User	user/changepassword	fas fa-fw fa-key	0	edit delete
7	Home	User	user/evoria	fas fa-fw fa-home	1	edit delete
8	Vendor	User	user/evoria/index#vendor	fas fa-fw fa-store	1	edit delete
9	Inspiration	User	user/inspiration	fas fa-fw fa-images	1	edit delete
10	Pesanan Saya	User	user/pesanan_saya	fas fa-fw fa-boxes	1	edit delete
11	Menu Management	Menu	menu	fas fa-fw fa-folder	1	edit delete

Gambar 4.53 Tampilan Tambah Submenu

Tampilan Submenu Management pada Gambar 4.52 dan Gambar 4.53 menampilkan daftar Submenu yang ditujukan untuk setiap Role pengguna sehingga setiap Role pengguna memiliki Submenu yang berbeda-beda.

4.2 Pengujian Black-Box

Pengujian *Black-Box* merupakan pengujian yang berfokus pada tampilan, fungsi-fungsi pada aplikasi serta kesesuaian alur fungsi dengan proses bisnis yang diinginkan calon pengguna.

1. Pengujian Admin

Pengujian Admin dapat dilihat pada Tabel 4.1 :

Tabel 4.1 Tabel pengujian Admin

No	Fungsi Yang Diuji	Cara Yang Dilakukan	Hasil Yang Diharapkan	Hasil Pengujian
1.	Fungsi Login	Masuk ke halaman login serta memasukkan <i>email</i> dan <i>password</i> yang benar	Masuk ke dalam sistem dan menampilkan Halaman Admin	Berhasil
2.	Fungsi Login	Masuk ke halaman login serta memasukkan <i>email</i> dan <i>password</i> yang salah	Tidak bisa masuk ke dalam sistem dan kembali ke Halaman Login	Berhasil
3.	Fungsi Add New Role	Masuk ke halaman Role dan klik tombol Add New Role	Menambahkan Role baru ke dalam <i>database</i> dan menampilkannya di halaman Role	Berhasil
4.	Fungsi Edit Role	Masuk ke halaman Role dan pilih Role yang ingin di-Edit	Menampilkan Form Edit dan melakukan proses Edit	Berhasil
5.	Fungsi Hapus Role	Masuk ke halaman Role dan pilih Role yang ingin dihapus	Melakukan proses Hapus data	Berhasil
6.	Fungsi Role Access	Masuk ke halaman Role dan mengganti akses menu yang diperbolehkan masing-masing Role	Masing-masing Role hanya bisa mengakses menu yang diperbolehkan sesuai yang ditentukan Admin	Berhasil
7.	Fungsi Add New Category	Masuk ke halaman Kategori dan klik tombol Add New Category	Menambahkan Category baru ke dalam <i>database</i> dan menampilkannya di halaman Kategori	Berhasil

8.	Fungsi Edit <i>Category</i>	Masuk ke halaman Kategori dan pilih <i>Category</i> yang ingin di-Edit	Menampilkan <i>Form</i> Edit dan melakukan proses Edit	Berhasil
9.	Fungsi Hapus <i>Category</i>	Masuk ke halaman Kategori dan pilih <i>Category</i> yang ingin dihapus	Melakukan proses Hapus data	Berhasil
10.	Fungsi Add Menu	Masuk ke halaman Menu <i>Management</i> dan menambahkan Menu baru	Menu yang ditambahkan dapat ditampilkan pada halaman Menu <i>Management</i>	Berhasil
11.	Fungsi Edit Menu	Masuk ke halaman Menu <i>Management</i> dan pilih Menu yang ingin di-Edit	Menampilkan <i>Form</i> Edit dan melakukan proses Edit	Berhasil
12.	Fungsi Hapus Menu	Masuk ke halaman Menu <i>Management</i> dan pilih Menu yang ingin dihapus	Melakukan proses Hapus data	Berhasil
13.	Fungsi Add Sub Menu	Masuk ke halaman Sub Menu <i>Management</i> dan Menambahkan Sub Menu baru	Sub Menu yang ditambahkan dapat ditampilkan pada halaman Sub Menu <i>Management</i>	Berhasil
14.	Fungsi Edit Sub Menu	Masuk ke halaman Sub Menu <i>Management</i> dan pilih Sub Menu yang ingin di-Edit	Menampilkan <i>Form</i> Edit dan melakukan proses Edit	Berhasil
15.	Fungsi Hapus Sub Menu	Masuk ke halaman Sub Menu <i>Management</i> dan pilih Sub Menu yang ingin dihapus	Melakukan proses Hapus data	Berhasil
16.	Fungsi My <i>Profile</i>	Masuk ke halaman My <i>Profile</i> dan melihat data diri	Menampilkan data diri	Berhasil
17.	Fungsi Edit <i>Profile</i>	Masuk ke halaman Edit <i>Profile</i> dan melakukan proses Edit data diri	Menampilkan <i>Form</i> Edit dan melakukan proses Edit	Berhasil
18.	Fungsi <i>Change Password</i>	Masuk ke halaman <i>Change Password</i> dan melakukan ganti <i>password</i>	Menampilkan <i>Form Change Password</i> dan melakukan ganti <i>password</i>	Berhasil
19.	Fungsi <i>Logout</i>	Memilih menu <i>Logout</i> dan melakukan <i>Logout</i>	Kembali ke halaman <i>Login</i>	Berhasil

2. Pengujian *User/Customer*

Pengujian *User/Customer* dapat dilihat pada Tabel 4.2 :

Tabel 4.2 Tabel pengujian *User/Customer*

No	Fungsi Yang Diuji	Cara Yang Dilakukan	Hasil Yang Diharapkan	Hasil Pengujian
1.	Fungsi <i>Register</i>	Masuk ke halaman <i>Register</i>	Menampilkan <i>Form Register</i> , <i>User/Customer</i> mengisi data diri dan mendaftar	Berhasil
2.	Fungsi <i>Login</i>	Masuk ke halaman login serta memasukkan <i>email</i> dan <i>password</i> yang benar	Masuk ke dalam sistem dan menampilkan Halaman <i>User/Customer</i>	Berhasil
3.	Fungsi <i>Login</i>	Masuk ke halaman login serta memasukkan <i>email</i> dan <i>password</i> yang salah	Tidak bisa masuk ke dalam sistem dan kembali ke Halaman <i>Login</i>	Berhasil
4.	Fungsi <i>Home</i>	Masuk ke halaman utama Evoria	Menampilkan halaman utama yang berisi <i>Carousel</i> , pilihan Produk Jasa, serta pilihan <i>Seller/EO</i>	Berhasil
5.	Fungsi <i>Inspirasi</i>	Masuk ke halaman <i>Inspirasi</i> dan melihat galeri dari semua <i>Inspirasi</i>	Menampilkan halaman <i>Inspirasi</i> yang berisi gambar-gambar inspirasi beserta judul dan informasi <i>Seller/EO</i> yang mengunggah	Berhasil
6.	Fungsi <i>Detail Jasa</i>	Memilih Produk Jasa dan masuk ke halaman <i>Detail Jasa</i>	Menampilkan <i>Detail</i> dari Produk Jasa yang dipilih	Berhasil
7.	Fungsi <i>Pesan Produk Jasa</i>	Memilih Produk Jasa dan masuk ke halaman <i>Detail Jasa</i> kemudian klik tombol <i>Pesan Sekarang</i>	Menampilkan <i>Form</i> yang berisi pemilihan tanggal acara, setelah memilih tanggal masuk ke halaman <i>Pesanan Saya</i>	Berhasil
8.	Fungsi <i>Pesanan Saya</i>	Masuk ke halaman <i>Pesanan Saya</i>	Menampilkan <i>Form</i> yang berisi daftar pesanan dari <i>User/Customer</i>	Berhasil

9.	Fungsi Batal Pesanan	Masuk ke halaman Pesanan Saya dan batalkan pesanan yang dipilih	Status pesanan menjadi “Pesanan Dibatalkan” dan pesanan pindah ke tabel Pesanan Dibatalkan	Berhasil
10.	Fungsi Pembayaran	Masuk ke halaman Pesanan Saya dan klik tombol Lanjutkan Pembayaran	Masuk ke halaman <i>Payment</i> dan Menampilkan <i>Form</i> metode pembayaran serta <i>Form Upload</i> Bukti Pembayaran, status pesanan menjadi “Menunggu Konfirmasi Pembayaran” dan pesanan pindah ke tabel Menunggu Pembayaran	Berhasil
11.	Fungsi Konfirmasi Pesanan Selesai	Masuk ke halaman Pesanan Saya dan klik tombol Selesai	Status pesanan menjadi “Pesanan Selesai” dan pesanan pindah ke tabel Pesanan Selesai	Berhasil
12.	Fungsi <i>Rating</i>	Masuk ke halaman Pesanan Saya dan klik tombol Beri <i>Rating</i> pada pesanan yang telah diselesaikan	Masuk ke modal beri rating dan memberikan <i>Rating</i>	Berhasil
13.	Fungsi <i>My Profile</i>	Masuk ke halaman <i>My Profile</i> dan melihat data diri	Menampilkan data diri	Berhasil
14.	Fungsi Edit <i>Profile</i>	Masuk ke halaman Edit <i>Profile</i> dan melakukan proses Edit data diri	Menampilkan <i>Form</i> Edit dan melakukan proses Edit	Berhasil
15.	Fungsi <i>Change Password</i>	Masuk ke halaman <i>Change Password</i> dan melakukan ganti <i>password</i>	Menampilkan <i>Form Change Password</i> dan melakukan ganti <i>password</i>	Berhasil
16.	Fungsi <i>Logout</i>	Memilih menu <i>Logout</i> dan melakukan <i>Logout</i>	Kembali ke halaman <i>Login</i>	Berhasil

3. Pengujian *Seller/EO*

Pengujian *Seller/EO* dapat dilihat pada Tabel 4.3 :

Tabel 4.3 Tabel pengujian *Seller/EO*

No	Fungsi Yang Diuji	Cara Yang Dilakukan	Hasil Yang Diharapkan	Hasil Pengujian
1.	Fungsi <i>Register</i>	Masuk ke halaman <i>Register</i>	Menampilkan <i>Form Register</i> , <i>Seller/EO</i> mengisi data diri serta data bisnis kemudian mendaftar	Berhasil
2.	Fungsi <i>Login</i>	Masuk ke halaman login serta memasukkan <i>email</i> dan <i>password</i> yang benar	Masuk ke dalam sistem dan menampilkan Halaman <i>Seller/EO</i>	Berhasil
3.	Fungsi <i>Login</i>	Masuk ke halaman login serta memasukkan <i>email</i> dan <i>password</i> yang salah	Tidak bisa masuk ke dalam sistem dan kembali ke Halaman <i>Login</i>	Berhasil
4.	Fungsi <i>Home</i>	Masuk ke halaman utama Evoria	Menampilkan halaman utama yang berisi <i>Carousel</i> , pilihan Produk Jasa, serta pilihan <i>Seller/EO</i>	Berhasil
5.	Fungsi Detail Jasa	Memilih Produk Jasa dan masuk ke halaman Detail Jasa	Menampilkan Detail dari Produk Jasa yang dipilih	Berhasil
6.	Fungsi Unggah Jasa	Masuk ke halaman Bisnis dan klik tombol Unggah Jasa	Menampilkan Form Unggah Jasa dan melakukan proses tambah data	Berhasil
7.	Fungsi Edit <i>Profile</i> Bisnis	Masuk ke halaman <i>Profile</i> dan klik tombol Edit Profile Bisnis	Menampilkan <i>Form</i> Edit Profile Bisnis dan melakukan proses Edit	Berhasil
8.	Fungsi Edit Produk Jasa	Masuk ke halaman <i>Profile</i> dan pilih Edit Produk Jasa yang diinginkan	Menampilkan <i>Form</i> Edit Produk Jasa dan melakukan proses Edit	Berhasil

9.	Fungsi Hapus Produk Jasa	Masuk ke halaman <i>Profile</i> dan pilih Hapus Produk Jasa yang diinginkan	Melakukan proses Hapus data Produk Jasa	Berhasil
10.	Fungsi Unggah Inspirasi	Masuk ke halaman Bisnis dan klik tombol Unggah Inspirasi	Menampilkan Form Unggah Inspirasi dan melakukan proses tambah data	Berhasil
11.	Fungsi Edit Inspirasi	Masuk ke halaman <i>Profile</i> dan pilih Edit Inspirasi yang diinginkan	Menampilkan <i>Form</i> Edit Inspirasi dan melakukan proses Edit	Berhasil
12.	Fungsi Hapus Inspirasi	Masuk ke halaman <i>Profile</i> dan pilih Hapus Inspirasi yang diinginkan	Melakukan proses Hapus data Inspirasi	Berhasil
13.	Fungsi Pesanan	Masuk ke halaman Pesanan	Menampilkan semua pesanan dari <i>User/Customer</i> yang ditujukan kepada <i>Seller/EO</i> yang sedang <i>Login</i>	Berhasil
14.	Fungsi Konfirmasi Pesanan	Masuk ke halaman Pesanan dan pilih pesanan yang ingin dikonfirmasi kemudian klik tombol Konfirmasi	Status pesanan menjadi “Menunggu Pembayaran” dan pesanan pindah ke tabel Menunggu Pembayaran	Berhasil
15.	Fungsi Batal Pesanan	Masuk ke halaman Pesanan dan batalkan pesanan yang dipilih	Status pesanan menjadi “Pesanan Dibatalkan” dan pesanan pindah ke tabel Pesanan Dibatalkan	Berhasil
16.	Fungsi Konfirmasi Pembayaran	Masuk ke halaman pesanan dan klik tombol Konfirmasi Pembayaran	Status pesanan menjadi “Pesanan Diproses” dan pesanan pindah ke tabel Pesanan Diproses	Berhasil
17.	Fungsi Pesanan Selesai	Masuk ke halaman Pesanan dan klik tombol Selesai	Status pesanan menjadi “Menunggu Konfirmasi Selesai” dan pesanan masih di tabel Pesanan Diproses	Berhasil
18.	Fungsi <i>My Profile</i>	Masuk ke halaman <i>My Profile</i> dan melihat data diri	Menampilkan data diri	Berhasil
19.	Fungsi Edit <i>Profile</i>	Masuk ke halaman Edit <i>Profile</i> dan melakukan proses Edit data diri	Menampilkan <i>Form</i> Edit dan melakukan proses Edit	Berhasil
20.	Fungsi <i>Change Password</i>	Masuk ke halaman <i>Change Password</i> dan melakukan ganti <i>password</i>	Menampilkan <i>Form Change Password</i> dan melakukan ganti <i>password</i>	Berhasil
21.	Fungsi <i>Logout</i>	Memilih menu <i>Logout</i> dan melakukan <i>Logout</i>	Kembali ke halaman <i>Login</i>	Berhasil

4.3 Analisis Komersialisasi

Komersialisasi Evoria adalah mengenalkan Evoria kepada masyarakat sehingga dapat dikonsumsi secara tepat guna dan menghasilkan keuntungan ekonomi bagi inventor/Evoria. Tujuan tahap ini adalah untuk menentukan ekspektasi nilai teknologi, termasuk penyusunan aplikasi potensial, pertimbangan berbagai keuntungan terhadap berbagai peluang, dan perlindungan teknologi melalui paten.

Langkah pertama yang dilakukan oleh tim Axel adalah mencari EO dan melakukan pertemuan dilanjutkan dengan penjelasan mengenai aplikasi Evoria. Ketika melakukan pertemuan dan wawancara, tim Axel juga membahas tentang apakah aplikasi Evoria ini akan dibutuhkan dan dapat mengatasi permasalahan yang ada atau tidak. Dari hasil wawancara yang telah dilakukan, ternyata aplikasi Evoria dibutuhkan oleh para EO karena dapat memudahkan mereka dalam memasarkan portofolionya sehingga calon pengguna jasa lebih mudah menemukan EO yang mereka butuhkan.

Langkah selanjutnya yang dilakukan oleh tim Axel adalah membagikan kuesioner secara *online* kepada para calon pengguna Evoria. Data yang diperoleh berupa jawaban dari pertanyaan seputar permasalahan yang terjadi di dunia EO baik dari vendor maupun pengguna jasanya, perlu tidaknya Evoria ini untuk dikembangkan, serta saran responden terhadap Evoria ini. Respon/jawaban yang diperoleh kemudian dianalisis dan diringkas. Berdasarkan hasil kuesoiner yang telah dianalisis dan diringkas, maka dapat disimpulkan bahwa aplikasi Evoria perlu ada karena dinilai dapat memudahkan para calon pengguna jasa EO dalam menemukan EO sesuai kebutuhan.

Setelah melakukan proses pengembangan kurang lebih selama 3 bulan akhirnya aplikasi Evoria telah berhasil dibuat. Berdasarkan hasil pengembangan, tim Axel mengambil kesimpulan bahwa aplikasi Evoria cukup layak untuk dikomersialkan. Berikut beberapa alasan mengapa aplikasi Evoria cukup layak untuk dikomersialkan :

1. Aplikasi dapat digunakan sebagai media promosi yang efektif karena berbasis web.
2. Pendapatan Evoria didapatkan dari tambahan biaya layanan sebesar 0,05% dari harga produk jasa.
3. Aplikasi mudah digunakan.
4. Berdasarkan hasil pertemuan dengan calon pengguna, mereka membutuhkan aplikasi Evoria untuk mencari jasa EO sesuai kebutuhan.
5. Susunan kode pemrograman cukup mudah dipahami sehingga akan lebih mudah jika ingin memodifikasi aplikasi ini.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil pengembangan aplikasi berbasis web *marketplace* Event Organizer Evoria menggunakan *framework* CodeIgniter dan metode *Rapid Application Development* (RAD) maka dapat disimpulkan sebagai berikut:

- a. Penggunaan metode RAD pada aplikasi Evoria dinilai membuat proses pengembangannya menjadi cukup singkat karena keunggulan dari metode ini adalah kecepatan, ketepatan dan biaya yang relatif lebih rendah. Selain itu metode RAD melibatkan pengguna dalam proses pengembangan sehingga kebutuhan sistem dapat terpenuhi dengan baik dan sesuai dengan keinginan pengguna itu sendiri.
- b. Aplikasi Evoria dikembangkan menggunakan bahasa pemrograman PHP, *framework* CodeIgniter, *web server* Apache, dan *database* MySQL.
- c. *Tools* yang digunakan dalam pengembangan aplikasi Evoria adalah XAMPP, VSCode, *Browser* Google Chrome, serta perangkat keras yaitu Laptop dan Modem.
- d. Aplikasi Evoria berbasis web sehingga dapat digunakan dengan mudah serta menjangkau pengguna yang lebih luas.
- e. Aplikasi Evoria dapat mempermudah para calon pengguna jasa EO dalam menemukan jasa EO yang sesuai kebutuhannya serta dapat meningkatkan perekonomian EO karena lebih banyak dikenal dan digunakan jasanya.

5.2 Saran

Berdasarkan pengembangan yang telah dilakukan maka sistem perlu perawatan, perbaikan apabila terdapat *bug*, penambahan fitur baru jika diperlukan, peningkatan keamanan serta pengelolaan basis data karena seiring banyaknya pengguna di masa mendatang.

DAFTAR PUSTAKA

- Annisa, T. (2020, Mei 11). *6 Langkah Validasi Ide Startup Yang Penting Dilakukan*. Retrieved from <https://www.ekrut.com/media/validasi-ide-startup>
- BPS. (2018). *Statistik Indonesia 2018*. Retrieved from <https://www.bps.go.id/publication/2018/07/03/5a963c1ea9b0fed6497d0845/statistik-indonesia-2018.html>
- Bridestory. (2017). *2017 INDONESIA WEDDING INDUSTRY REPORT BY BRIDESTORY*. Retrieved from <https://www.bridestory.com/id/blog/2017-indonesia-wedding-industry-report-by-bridestory>
- CodeIgniter. (2020, Mei 1). *CodeIgniter4 Overview*. Retrieved from https://codeigniter.com/user_guide/concepts/mvc.html
- CodeIgniter. (2020, Mei 1). *Welcome to CodeIgniter4*. Retrieved from https://codeigniter.com/user_guide/intro/index.html
- Dr. Bachruddin Saleh Luturlean, S. M. (2019). STRATEGI BISNIS PARIWISATA. In *STRATEGI BISNIS PARIWISATA* (p. 186). Bandung: Humaniora. Retrieved from https://books.google.co.id/books?id=BVi9DwAAQBAJ&pg=PA186&dq=penyelenggara+acara+event+organizer&hl=id&sa=X&ved=2ahUKEwjJj5_ly6bqAhVDWysKHSa7CtgQ6AEwAHoECAQQAg#v=onepage&q=penyelenggara%20acara%20event%20organizer&f=false
- HAFIDH. (2020, Februari 7). *Mengenal Bisnis Startup: Peluang dan Tips untuk Memulainya*. Retrieved from <https://www.jurnal.id/id/blog/mengenal-bisnis-startup-peluang-dan-tips-untuk-memulainya/>
- Jamaluddin, A. A. (2017). BEBERAPA METODE MELAKUKAN ANALISIS KEBUTUHAN. 10-17.
- Jobhun. (2018, Juli 1). *Mengenal Tiga Karakter Penting yang Ada di Startup*. Retrieved from <https://jobhun.id/mengenal-tiga-karakter-penting-yang-ada-di-startup/>
- Manuwu, J. P. (2018, April 25). *4 Tahapan Lean Market Validation untuk Melakukan Validasi terhadap Ide Startup*. Retrieved from <https://id.techinasia.com/validasi-ide-startup>
- Piyaneo. (2014, Mei 10). *RAPID APPLICATION DEVELOPMENT (RAD)*. Retrieved from <https://piyaneo.wordpress.com/2014/05/10/rapid-application-development-rad/>

- Richard, M. (2019, Februari 13). *Industri Event Organizer Diproyeksi Tumbuh 20% Tahun Ini*. Retrieved from <https://ekonomi.bisnis.com/read/20190213/12/888451/industri-event-organizer-diproyeksi-tumbuh-20-tahun-ini>
- Sugiarto, C. (2019). *18 Cara Bisnis Event Organizer Sukses di 2019 Mulai Dari Nol*. Retrieved from <https://erakini.com/bisnis-event-organizer/>
- Utami, N. W. (2020, Januari 4). *Cara Cerdas Validasi Ide dalam Rencana Bisnis*. Retrieved from <https://www.jurnal.id/id/blog/cara-cerdas-validasi-ide-dalam-rencana-bisnis/>
- Venuemagz. (2016, Februari 11). *Mengejar Pertumbuhan MICE Yogyakarta*. Retrieved from <https://venuemagz.com/feature/mengejar-pertumbuhan-mice-yogyakarta/>
- Visitingjogja. (2019). *Yogyakarta Calender of Event 2019*. Retrieved from https://drive.google.com/file/d/1JOLA71KyrvGkjJF4DH4AMvCr4_31kBvV/view
- Wahyuningrum, T., & Januarita, D. (2014). Perancangan WEB e-Commerce dengan Metode Rapid Application Development (RAD) untuk Produk Unggulan Desa. 81.
- Widiyanto, W. W. (2018). *ANALISA METODOLOGI PENGEMBANGAN SISTEM DENGAN PERBANDINGAN MODEL PERANGKAT LUNAK SISTEM INFORMASI KEPEGAWAIAN MENGGUNAKAN WATERFALL DEVELOPMENT MODEL, MODEL PROTOTYPE, DAN MODEL RAPID APPLICATION DEVELOPMENT (RAD)*. Retrieved from <https://informa.poltekindonusa.ac.id/index.php/informa/article/view/34/30>

LAMPIRAN

Lampiran A

Perintisan Bisnis

Perintisan bisnis atau bisnis *startup* adalah bisnis rintisan yang dapat tumbuh dan menguasai pasar sehingga diharapkan dapat berkembang menjadi perusahaan besar. Bisnis *startup* lebih mengutamakan ide-ide baru dan memberi solusi untuk permasalahan konsumen yang terjadi pada era sekarang (HAFIDH, 2020).

Dalam pengembangan aplikasi Evoria tentu dibutuhkan tim yang masing-masing memiliki bidang pekerjaan atau peran yang dikuasai (Jobhun, 2018). Tim Axel merupakan kelompok yang terdiri dari tiga orang dalam pembuatan *startup* yang berfokus dalam bidang *event organizer*. Adapun identitas dan posisi setiap anggota sebagai berikut :

1. Hustler

Hustler dalam tim Axel adalah Mutia Mernisiola Zipa yang bertugas dan bertanggung jawab dalam bidang pemasaran untuk memasarkan suatu aplikasi yang akan dibuat yaitu Evoria.

2. Hipster

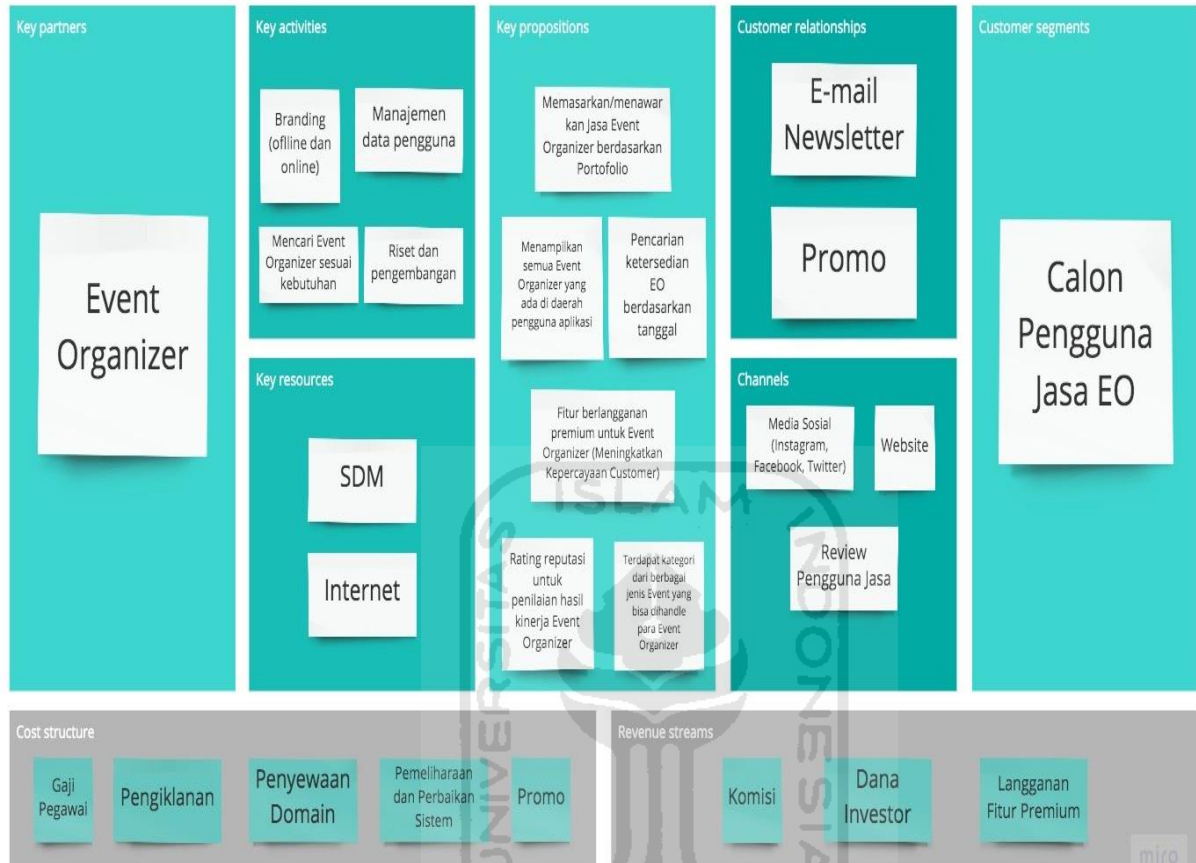
Hipster dalam tim Axel adalah Muhammad Rasyid Shadiq yang bertanggung jawab dalam bidang desain *user interface* dan *user experience* pada aplikasi Evoria.

3. Hacker

Hacker dalam tim Axel adalah Aditya Mahavira yang bertanggung jawab dalam bidang pemrograman atau *coding* dalam mengimplementasikan desain dan mengembangkan aplikasi Evoria.

Lampiran B

Business Model Canvas (BMC) Evoria.



Lampiran C

Dokumentasi pengembangan Evoria.

```
File Edit Selection View Go Run Terminal Help
User.php - evoria - Visual Studio Code

EXPLORER
evoria
  application
  cache
  controllers
  Admin.php
  Auth.php
  index.html
  Kategori.php
  Menu.php
  Seller.php
  User.php
  Welcome.php
  core
  helpers
  evoria_helper.php
  index.html
  tanggal_helper.php
  test.php
  hooks
  language
  libraries
  logs
  models
  BismillahModel.php
  HapusModel.php
  index.html
  KategoriModel.php
  MenuModel.php
  PesanModel.php
  RatingModel.php

OUTLINE
  NPM SCRIPTS

TERMINAL
  powerhell
  Copyright (c) Microsoft Corporation. All rights reserved.
  Try the new cross-platform PowerShell https://aka.ms/powershell
  PS C:\xampp\htdocs\evoria>
```

```
{
  public function edit()
  {
    $data['judul'] = 'Edit Profile';
    $data['user'] = $this->db->get_where('users', ['email' =>
      $this->session->userdata('email')]);->row_array();

    $this->form_validation->set_rules('name', 'Nama Lengkap', 'required|trim');
    $this->form_validation->set_rules('nohp', 'No HP', 'required|trim');
    $this->form_validation->set_rules('pekerjaan', 'Pekerjaan', 'required|trim');

    if ($this->form_validation->run() == false) {
      $this->load->view('templates/header_evoria', $data);
      $this->load->view('user/edit', $data);
      $this->load->view('templates/footer');
    }
  }
}
```