## 4.2 Tahap Analisis

Pada tahap ini akan dilakukan analisis terhadap *business requirements* dalam sistem usulan dan juga pemodelan dari sistem usulan secara fungsional maupun struktural. Hasil dari analisis tersebut akan digunakan sebagai dasar dalam pembuatan sistem yang diusulkan.

### 4.2.1 User Requirements

*Requirements* merupakan suatu penjelasan mengenai hal apa saja yang harus dilakukan oleh sistem dan karakteristik yang dimiliki oleh sistem yang dapat berguna bagi *user*. *Requirements* diperoleh melalui hasil analisis terhadap kebutuhan *user* dan studi literatur. Secara umum, *user requirements* terdiri dari *functional requirements* dan *non-functional requirements*.

#### *4.2.1.1 Functional Requirements*

*Functional requirements* didefinisikan sebagai kebutuhan yang harus dapat dilakukan oleh sistem. *Functional requirements* pada aplikasi layanan jasa *event organizer* berbasis *web* ini dibagi menjadi dua, yaitu:

1. Aplikasi berbasis *web* untuk *front-end*

Berikut adalah fungsi-fungsi yang harus dimiliki pada aplikasi layanan jasa *event organizer* berbasis *web* untuk *front-end*:

1. Melakukan pendaftaran akun
2. Fungsi ini digunakan oleh *user* untuk membuat akun;
3. User mengisi formulir yang terdiri dari nama pengguna (*username*), *email*, dan *password*.
4. *User* akan menerima bukti konfirmasi melalui *email* yang akan dikirimkan oleh sistem.
5. Melakukan pendaftaran sebagai EO
6. Fungsi ini digunakan oleh EO untuk membuat akun;
7. EO mengisi informasi mengenai data usaha yang dibutuhkan pada formulir pendaftaran sebagai EO;
   1. Semua informasi dalam formulir wajib diisi dengan lengkap agar dapat diproses lebih lanjut oleh sistem;
   2. EO perlu mengunggah dokumen seperti foto KTP, foto diri dengan KTP, foto tagihan listrik/telepon/air yang mencantumkan alamat, foto SIUP/TDP untuk memvalidasi bisnis.
8. Melihat jasa EO
9. Fungsi ini digunakan oleh *user* untuk melihat jasa EO yang ada;
10. Sistem melakukan kategorisasi terhadap EO yang telah terdaftar berdasarkan jenis pelayanan dan lokasi serta menampilkan informasinya yang lengkap.
11. Mengajukan *request* layanan

Klien dapat mengajukan *request* layanan dengan mengisi informasi-informasi yang dibutuhkan pada formulir yang telah disediakan dengan lengkap dan jelas.

1. Mengelola *request* layanan

EO dapat melihat *request* layanan yang telah diajukan dan dapat memilih untuk mencetak *request* layanan tersebut.

1. Memberi *review*

Klien memberi *review* terhadap EO berupa *rating* dan testimoni mengenai pelayanan yang telah disediakan EO tersebut.

1. Melihat rekomendasi EO
2. Klien menentukan bobot kepentingan dari kriteria EO yang diinginkan.
3. Sistem akan menampilkan rekomendasi EO yang sesuai dengan preferensi klien.
4. Mengelola *portfolio*

Fungsi ini digunakan oleh EO untuk menambah, mengubah, serta mengurangi data mengenai *portfolio*-nya.

1. Mengelola paket

Fungsi ini digunakan oleh EO untuk menambah, mengubah, serta mengurangi data mengenai paket yang disediakan.

1. Mengelola *appointment reminder*
2. Fungsi ini digunakan oleh EO untuk menambah dan mengurangi data *appointment reminder*;
3. EO dapat menentukan tanggal, jam dan keterangan mengenai hal yang harus dilakukan dan klien akan menerima notifikasi berupa *email*.
4. Mengelola pesan
5. Fungsi ini digunakan oleh klien untuk mengirim pesan kepada EO;
6. EO menerima pesan yang dikirimkan melalui fitur *inbox* dan dapat membalas pesan tersebut.
7. Aplikasi berbasis *web* untuk *back-end*

Berikut adalah fungsi-fungsi yang harus dimiliki pada aplikasi layanan jasa *event organizer* berbasis *web* untuk *back-end*:

1. Mengelola akun

Fungsi ini digunakan oleh *admin* untuk melakukan penambahan, perubahan serta pengurangan data mengenai akun klien dan EO.

1. Mengelola kriteria

Fungsi ini digunakan oleh *admin* untuk melakukan penambahan, perubahan serta pengurangan kriteria berupa nama, tipe dan bobotnya.

1. Mengelola data kriteria EO

Fungsi ini digunakan oleh *admin* untuk melakukan penambahan, perubahan serta pengurangan data kriteria berupa nama alternatif (EO), kriteria, dan nilai yang dimiliki.

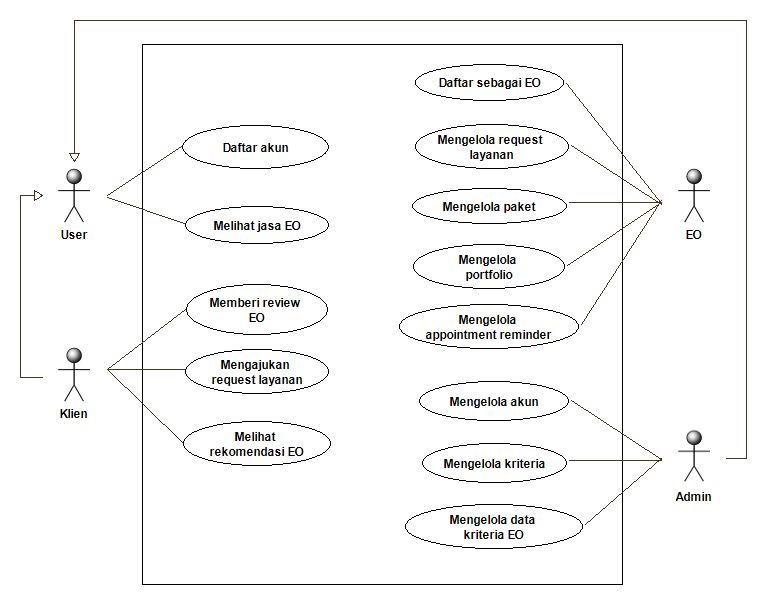
#### *4.2.1.2 Non-functional Requirements*

*Non-functional requirements* menjelaskan apa saja batasan-batasan yang terdapat pada sistem dan bagaimana sistem dapat berjalan. *Non-functional requirements* pada aplikasi layanan jasa *event organizer* berbasis *web*, antara lain:

1. Operasional
2. Aplikasi ini dapat dijalankan pada *web browser* Google Chrome (versi 53 hingga sekarang) dan Mozilla Firefox (versi 48 hingga sekarang).
3. Aplikasi ini dapat dijalankan dengan baik pada sistem operasi Windows 7 dan setelahnya.
4. Performa
5. Aplikasi ini dapat diakses selama 24 jam.
6. Informasi yang ada dalam *database* dapat diperbaharui secara *real-time*.
7. Keamanan
8. Aplikasi ini memiliki beberapa level akses pengguna yang terdiri dari klien, EO, dan *admin* yang masing-masing memiliki akses terhadap fungsi-fungsi tertentu.
9. Untuk menjalankan berbagai fungsi dalam aplikasi, pengguna harus masuk ke akun dengan memasukkan *username/email* dan *password* yang benar.
10. Lingkungan organisasi
11. Tampilan aplikasi dirancang dengan baik dan memiliki struktur yang teratur agar mudah dimengerti dan digunakan oleh pengguna.
12. Penggunaan bahasa dalam aplikasi ini adalah Bahasa Inggris.

### Use Case Diagram

Berikut *use case diagram* dari aplikasi layanan jasa *event organizer* berbasis *web* yang diusulkan dapat dilihat pada Gambar 4.1.

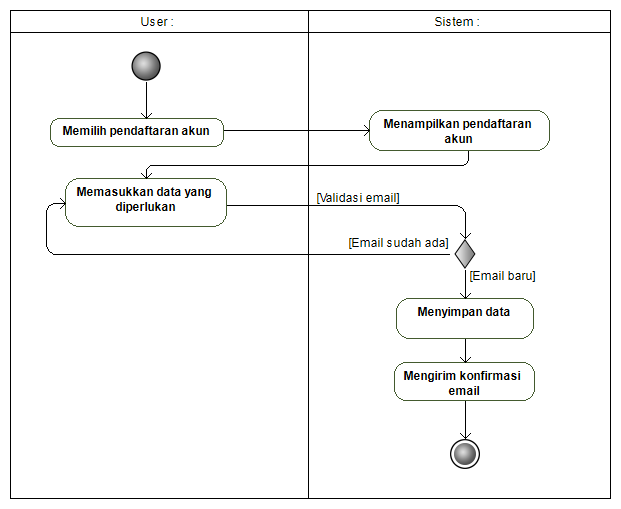


Gambar 4. *Use Case Diagram* Sistem Usulan

### Activity Diagram

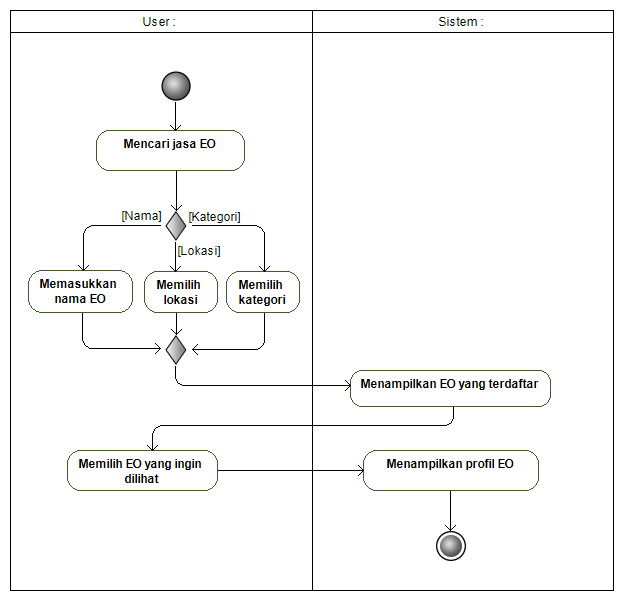
Berikut adalah proses-proses utama dari aplikasi layanan jasa *event organizer* berbasis *web* ini yang digambarkan melalui *activity diagram*:

1. Daftar akun



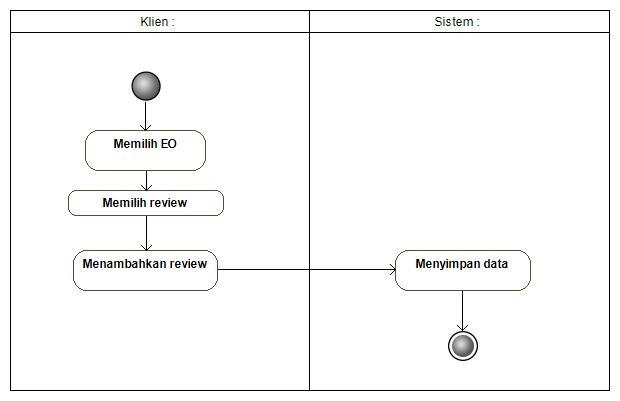
Gambar 4. *Activity Diagram:* Daftar akun

1. Melihat jasa EO



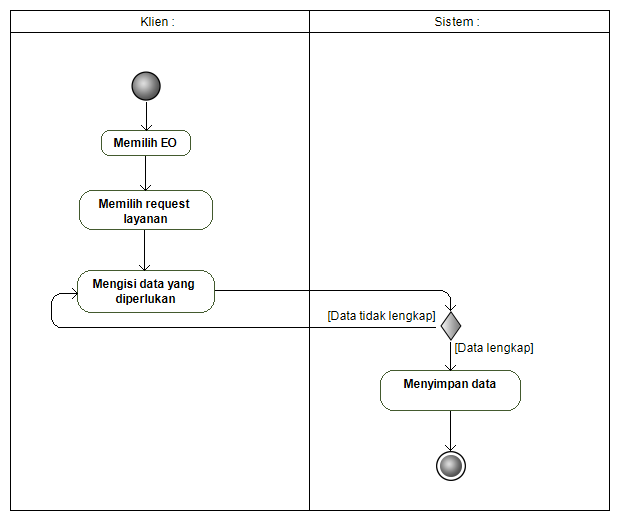
Gambar 4. *Activity Diagram*: Melihat jasa EO

1. Memberi *review*



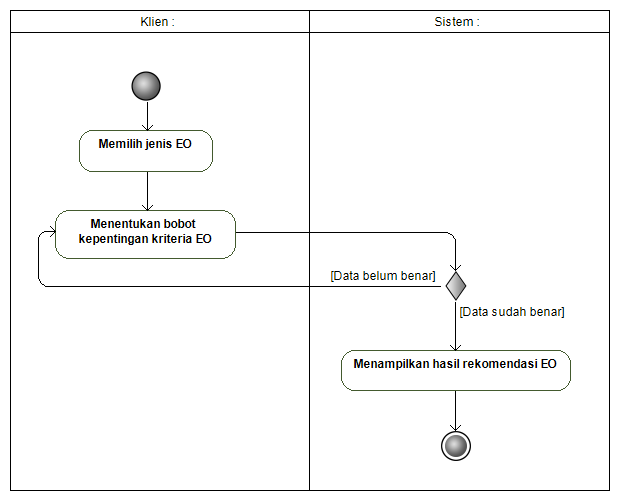
Gambar 4. *Activity Diagram*: Memberi *review*

1. Mengajukan *request* layanan



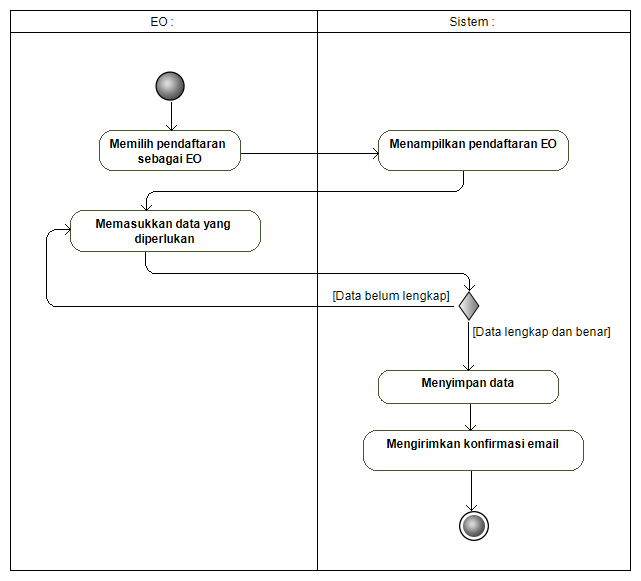
Gambar 4. *Activity Diagram*: Mengajukan *request* layanan

1. Melihat rekomendasi EO



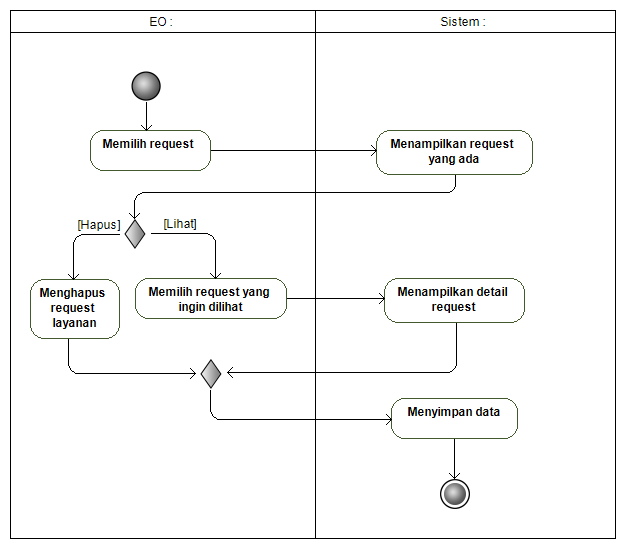
Gambar 4. *Activity Diagram*: Melihat rekomendasi EO

1. Daftar sebagai EO



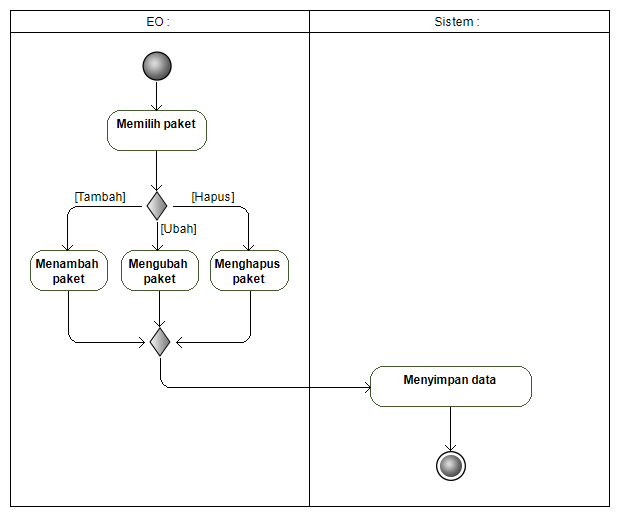
Gambar 4. *Activity Diagram*: Daftar sebagai EO

1. Mengelola *request* layanan



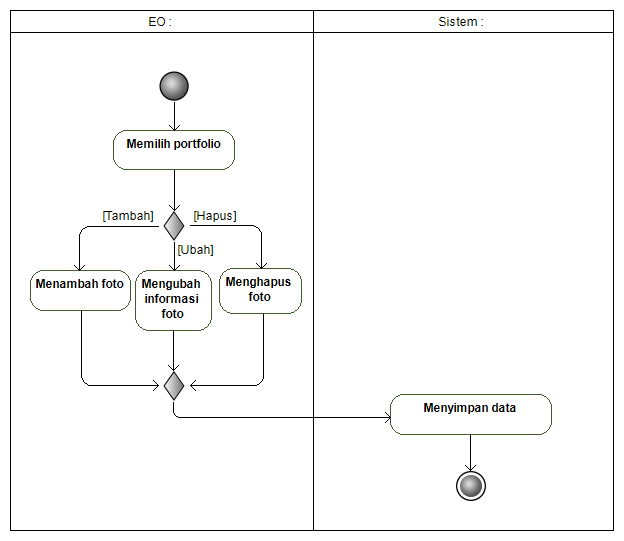
Gambar 4. *Activity Diagram*: Mengelola *request* layanan

1. Mengelola paket



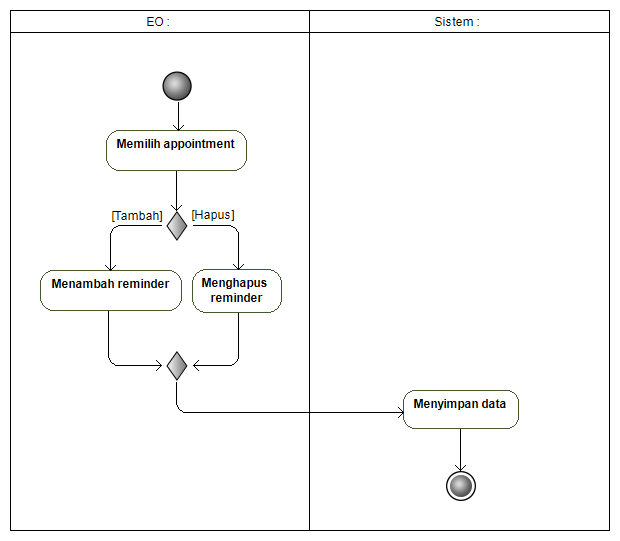
Gambar 4. *Activity Diagram*: Mengelola paket

1. Mengelola *portfolio*



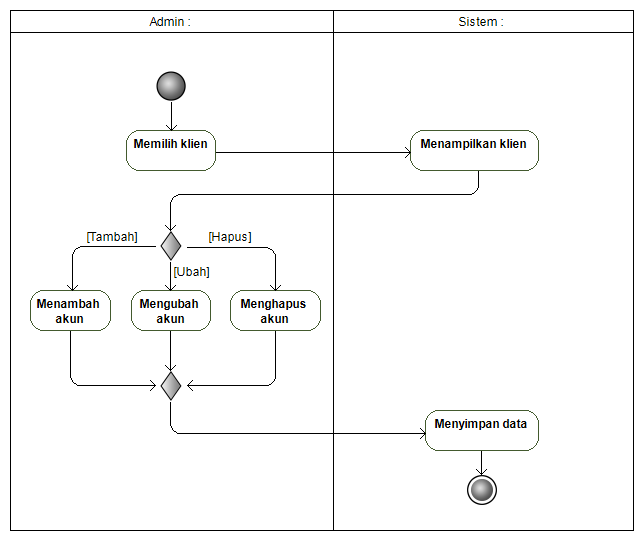
Gambar 4. *Activity Diagram*: Mengelola *portfolio*

1. Mengelola *appointment reminder*



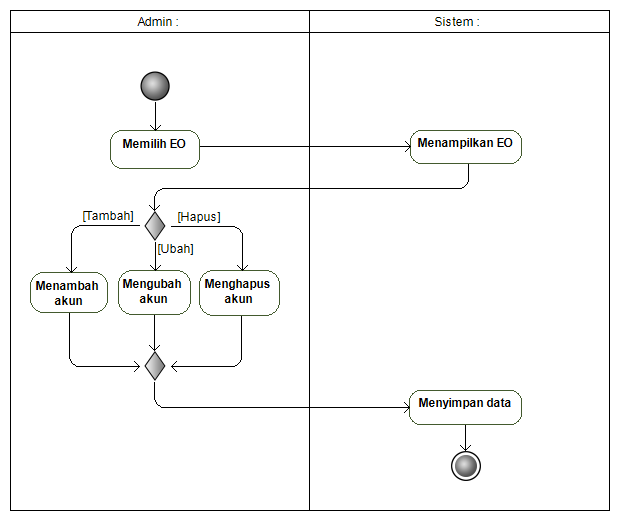
Gambar 4. *Activity Diagram*: Mengelola *appointment reminder*

1. Mengelola akun
2. Klien



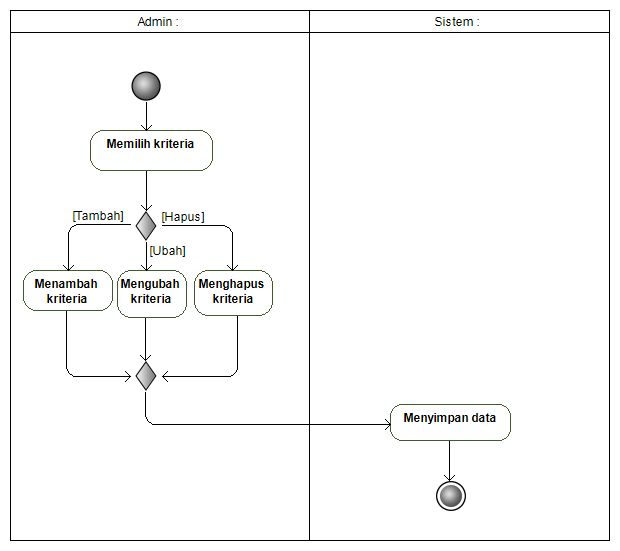
Gambar 4. *Activity Diagram*: Mengelola akun klien

1. EO



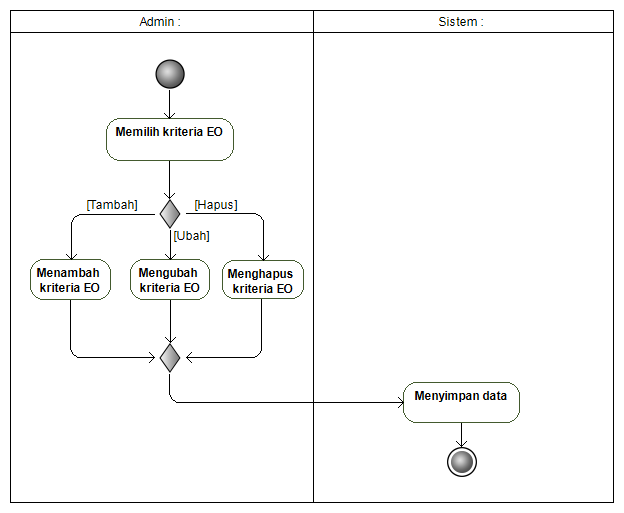
Gambar 4. *Activity Diagram*: Mengelola akun EO

1. Mengelola kriteria



Gambar 4. *Activity Diagram*: Mengelola kriteria

1. Mengelola data kriteria EO



Gambar 4. *Activity Diagram*: Mengelola data kriteria EO

### Class Diagram