LAPORAN TUGAS PROYEK AKHIR

PERANCANGAN ”WEORGANIZE” PORTAL EVENT APP

Logo, company name

Description automatically generated

Disusun oleh :

|  |  |
| --- | --- |
| **Nama** | **NIM** |
| Candra | 20190801153 |
| Dhea Devita Oktaryana | 20190801135 |
| Much Etrik Wijanarko | 20190801178 |
| Teguh Agung Prabowo | 512121230007 |
| Yayan Afriyanto | 30818059 |

**FAKULTAS ILMU KOMPUTER**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**UNIVERSITAS ESA UNGGUL KEBON JERUK**

**TAHUN 2022**

KATA PENGANTAR

Puji suyukur kehadirat Tuhan Yang Maha Esa atas segala rahmat- Nya sehingga laporan proyek akhir yang berjudul “Perancangan ’WeOrganize’ Portal Event App” dapat tersusun hingga selesai. Tidak lupa pula penulis mengucapkan terimakasih atas bantuan pihak-pihak yang telah berkontribusi dengan memberikan dukungan moral maupun materi yang diberikan kepada penulisan laporan ini.

Dan harapan penulis semoga laporan ini dapat menambah pengetahuan dan pengalaman bagi para pembaca, untuk kedepanya dapat memperbaiki bentuk maupun menambah isi makalah agar menjadi lebih baik lagi.

Penulis menyadari bahwa laporan ini belum sempurna. Oleh karena itu, saran dan kritik dari pembaca sangat dibutuhkan untuk penyempurnaan laporan ini.

Jakarta, Juli 2022

Penulis

# DAFTAR ISI

[1 PENDAHULUAN 3](#_Toc77203408)

[2 LANDASAN TEORI 3](#_Toc77203409)

[3 RUANG LINGKUP PROYEK SISTEM INFORMASI 3](#_Toc77203410)

[3.1 Definisi Masalah Dan Ruang Lingkup 3](#_Toc77203411)

[3.2 Kebutuhan Sistem Secara Fungsional 3](#_Toc77203412)

[3.3 Kebutuhan Sistem Secara Non Fungsional 3](#_Toc77203413)

[3.4 Business Proses Saat Ini Dan Business Process Yang Diusulkan 3](#_Toc77203414)

[3.5 Analisis Masalah dengan Metode FISHBONE/PIECES/SWOT 3](#_Toc77203415)

[3.6 Tahapan Rencana Pembangunan Sistem Informasi 3](#_Toc77203416)

[3.7 Pembagian Tugas Kelompok dan Bukti Kerja Kelompok 3](#_Toc77203417)

[4 PROSES / LANGKAH METODOLOGI PENGEMBANGAN SISTEM INFORMASI 3](#_Toc77203418)

[5 PERANCANGAN APLIKASI 4](#_Toc77203419)

[6 MODEL ARSITEKTUR APLIKASI 4](#_Toc77203420)

[7 UI / UX APLIKASI / SISTEM INFORMASI 4](#_Toc77203421)

[8 DAFTAR PUSTAKA 4](#_Toc77203422)

# PENDAHULUAN

Dalam menyelenggarakan kegiatan-kegiatan yang bersifat kajian ilmiah, manajerial dan operasional pada umumnya tidak terlepas dari penyelenggaraan kegiatan seminar dan workshop. Penyelenggaran kegiatan tersebut tentunya untuk membantu kemampuan sumber daya manusia dalam meningkatkan pengetahuan dan keterampilan, kegiatan ini biasanya diadakan oleh kampus-kampus dan perkantoran untuk mengadakan kegiatan kajian ilmiah, pelatihan manajerial dan operasional, kegiatannya dapat dilaksanakan satu bulan sekali atau sesuai kesepakan yang ditentukan.

Biasanya seminar dipromosikan melalui poster dan pamflet yang ditempelkan di tempat pengumuman kampus ataupun kantor. Namun di masa pandemi ini, kebanyakan mahasiswa maupun karyawan kesulitan untuk menemukan wadah yang membagikan kegiatan bermanfaat seperti *event* dan seminar yang dapat menambah wawasan. Maka dari itu, dirancanglah suatu aplikasi yang dapat mengatasi masalah tersebut, yakni WeOrganize.

WeOrganize merupakan aplikasi yang digunakan sebagai sarana untuk mengikuti *event* dan menyelenggarakan *event* seperti seminar *online* maupun *offline*. WeOrganize bersifat *general* dimana semua *user* dapat menyelenggarakan maupun mengikuti *event* yang ada. *Event-*event ini dapat diselenggarakan tidak hanya dari satu universitas ataupun suatu lembaga saja tapi dapat diselenggarakan oleh perusahaan juga. Semua *user* dapat mengikuti *event-event* yang diselenggarakan dalam WeOrganize.

WeOrganize juga membantu penyelenggaraan pada *event* seperti seminar karena dapat menyelesaikan permasalahan seputar pelaksanaan *event* yang masih banyak menggunakan sistem manual sehingga dalam penelitian ini dapat memberikan solusi terhadap masalah yang terjadi pada penyelenggara dan peserta agar dapat memberikan informasi seputar pengumuman *event* dan pendaftaran peserta secara *online*, dapat menghindari absensi fiktif, karena absen kehadiran peserta menggunakan QRCode yang dikirim ke *email* masing-masing, serta peserta dapat bertanya langsung melalui aplikasi tanpa menyela isi materi yang sedang disampaikan oleh pembicara.

1. LANDASAN TEORI
2. RUANG LINGKUP PROYEK SISTEM INFORMASI
   1. Definisi Masalah dan Ruang Lingkup
   2. Kebutuhan Sistem Fungsional

Kebutuhan fungsional adalah suatu kebutuhan berupa fungsi atau fitur – fitur yang harus ada dalam sistem yang akan dibuat dan berkaitan langsung dengan studi kasus yang dipilih. Daftar kebutuhan fungsional sistem akan ditampilkan pada tabel berikut ini.

|  |  |  |
| --- | --- | --- |
| No | Kebutuhan Fungsional | User |
| 1 | Sistem dapat menampilkan verifikasi *login* untuk EO | EO |
| 2 | Sistem dapat menampilkan *form* konfirmasi data bagi EO | EO |
| 3 | Sistem dapat menampilkan data *Participant* yang mengikuti *event* | EO |
| 4 | Sistem dapat menampilkan data *profile EO* | EO |
| 5 | Sistem dapat menampilkan halaman beranda | EO |
| 6 | Sistem dapat menampilkan menu membuat *event* | EO |
| 7 | Sistem dapat menampilkan menu *edit event* | EO |
| 8 | Sistem dapat menampilkan verifikasi *login* untuk *Participant* | Participant |
| 9 | Sistem dapat menampilkan *form* konfirmasi data bagi *Participant* | Participant |
| 10 | Sistem dapat menampilkan data *history event* | Participant |
| 11 | Sistem dapat menampilkan data *profile Participant* | Participant |
| 12 | Sistem dapat menampilkan halaman beranda | Participant |
| 13 | Sistem dapat menampilkan menu *scan qrcode* untuk *check-in event* | Participant |
| 14 | Sistem dapat menampilkan menu daftar *event* | Participant |
| 15 | Sistem dapat menampilkan notifikasi setelah berhasil *check-in* | Participant |

* 1. Kebutuhan Sistem Non Fungsional

Kebutuhan non fungsional merupakan kebutuhan yang menitikberatkan pada properti yang dimiliki oleh sistem atau yang diperlukan di luar kebutuhan fungsional. Adapun kebutuhan non fungsional yang dibutuhkan adalah :

1. Karakteristik Sistem

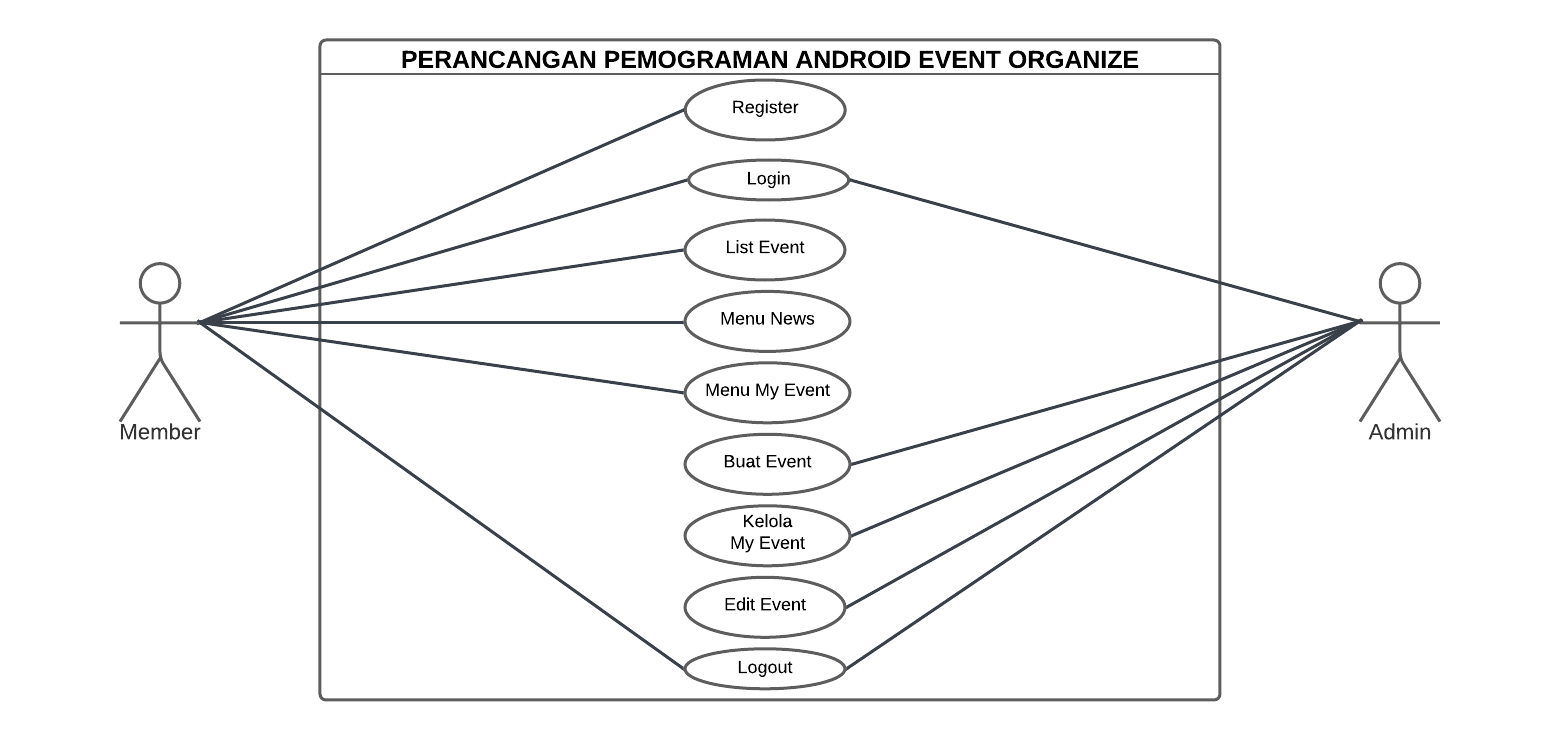
|  |  |  |
| --- | --- | --- |
| No | Karakteristik Sistem | Keterangan |
| 1 | *Portability* | Aplikasi dapat diakses dimana saja dan kapan saja asalkan terhubung dengan internet. |
| 2 | *User Friendly* | Penggunaan fitur serta navigasi aplikasi yang mudah. |
| 3 | *Ergonomy* | Menjamin kenyamanan pemakaian antarmuka sistem dengan pengguna. |
| 4 | *Responsive* | *Responsive* merupakan suatu teknik agar tampilan aplikasi dapat menyesuaikan dengan layar *device* yang mengaksesnya. |

1. Kebutuhan Perangkat Lunak (*Software)* dan Perangkat Keras (*Hardware)*

|  |  |
| --- | --- |
| Kebutuhan Perangkat Lunak | Kebutuhan Perangkat Keras |
| 1. *Java Development Kit* versi 8   Digunakan untuk mengkompilasi kode program, dan *java* *development kit* ini kompatibel dengan spesifikasi laptop yang digunakan. | 1. Processor Intel® Core™ i3-3217U CPU @ 1.80GHz (4 CPUs), ~1.8GHz |
| 1. *Java Runtime*   Merupakan *platform* untuk menjalankan sistem, ini digunakan karena kompatibel dengan *Eclipse* *Juno*. | 1. RAM 2GB DDR3 |
| 1. Android Studio   Digunakan untuk membangun aplikasi *client* pada android. Aplikasi ini digunkan karena mudah diinstal dan digunakan dalam pembuatan *java script* dan juga *free*. | 1. HDD 500GB |
| 1. Android SDK,   Untuk pengembangan aplikasi android. | 1. Smartphone Android |
| 1. *Android Development Tools, plugin android* pada *android studio.* |  |
| 1. *Android Virtual Device, emulator* untuk menjalankan sistem android.   *Android SDK, Android Development Tools, dan Android Virtual Device* menggunakan tools bawaan dari *Eclipse Juno* yang dapat digunakan dengan mudah. |  |
| 1. Minimal Sistem Operasi Windows 7   Sistem Operasi Windows digunakan untuk proses *development* aplikasi ini. |  |
| 1. Android OS   Sistem Operasi Android ini dipilih karena sistem operasi ini bersifat *open source*, sehingga dapat dikembangkan siapa saja yang nantinya mau mengembangkannya. |  |

1. METODOLOGI PENGEMBANGAN SISTEM INFORMASI
2. PERANCANGAN APLIKASI
   1. *Use Case Diagram*

*Use Case* *Diagram* merupakan diagram yang menggambarkan hubungan antara aktor dengan sistem. *Use Case Diagram* bisa mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem yang akan dibuat*. Use Case Diagram* juga bisa digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem dan bisa juga mempresentasikan sebuah interaksi aktor dengan sistem.

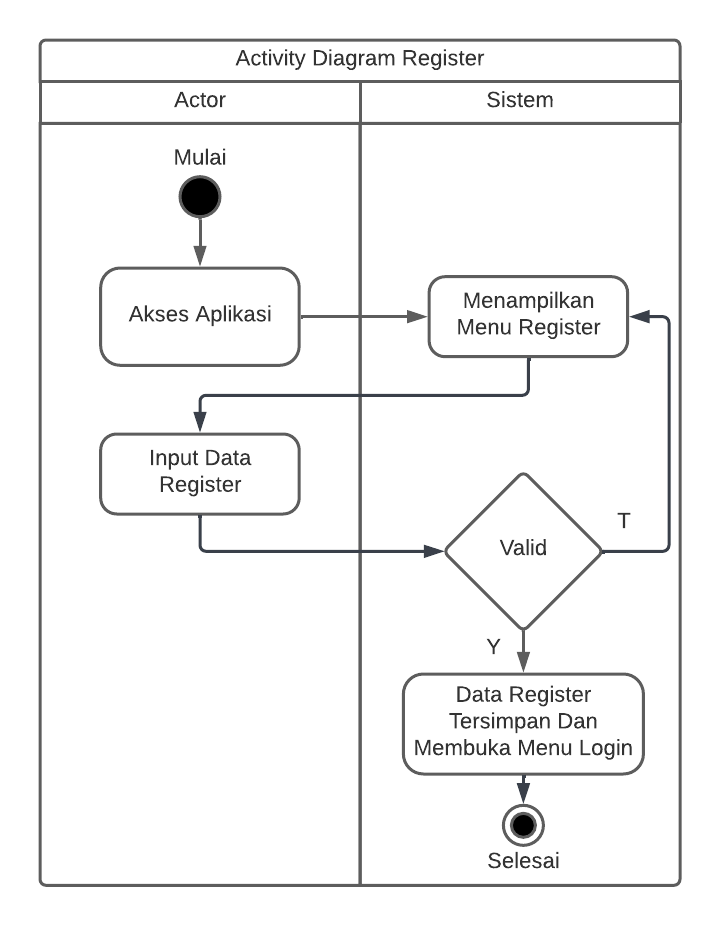
Berikut ini *Use Case Diagram* dari Perancangan Aplikasi WeOrganize :  


* 1. *Activity Diagram* (Sistem / Proses Bisnis)

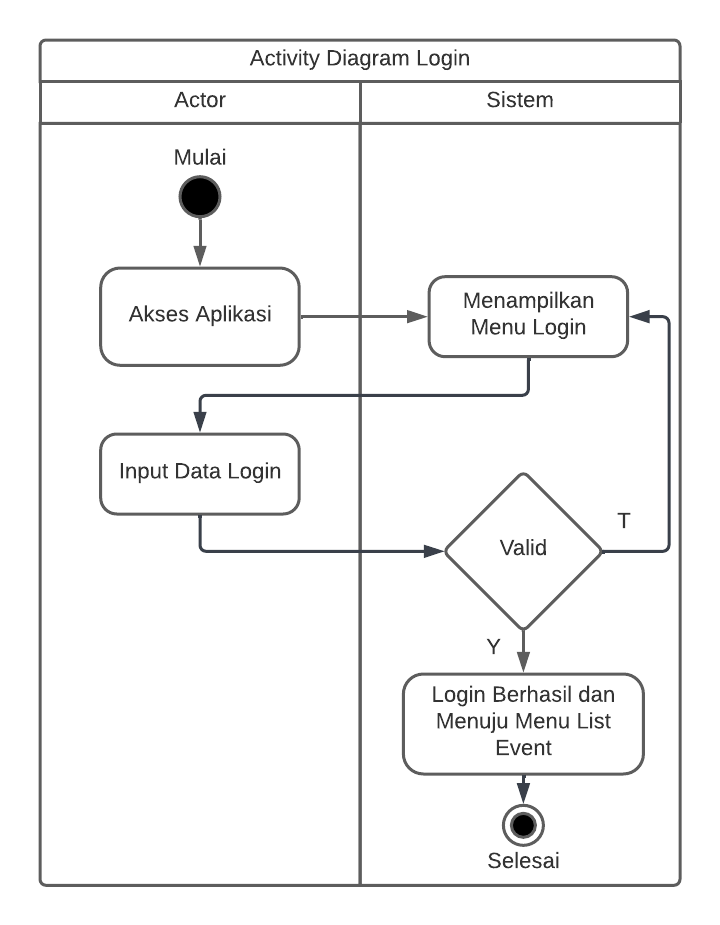
*Activity Diagram* adalah salah satu cara untuk memodelkan *event-event* yang terjadi dalam suatu *use case.* Berikut ini *activity diagram* dari Perancangan

Aplikasi WeOrganize :

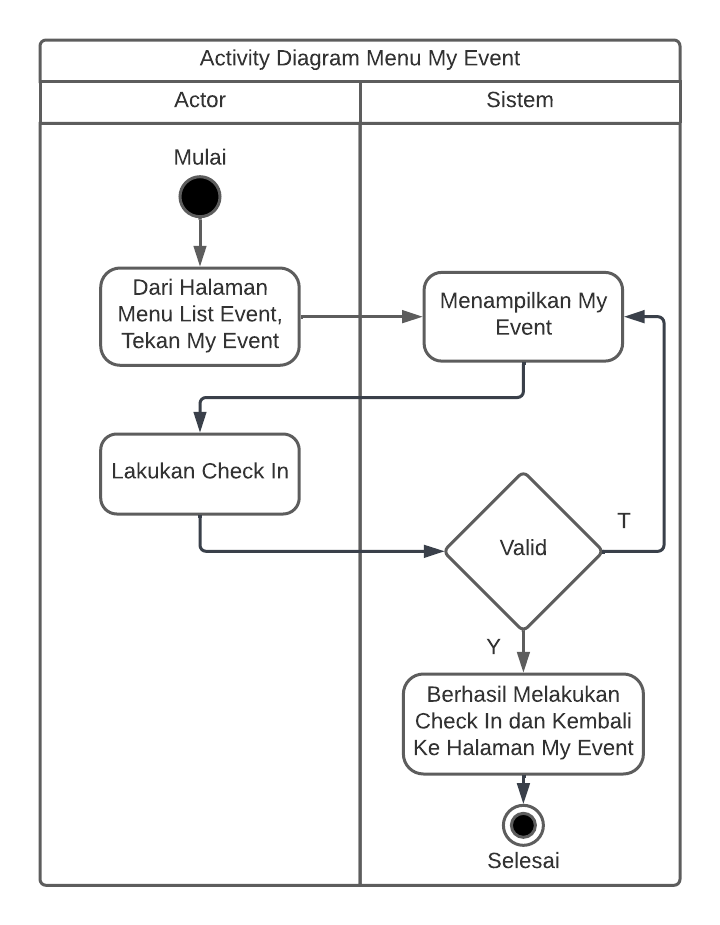
* 1. Register



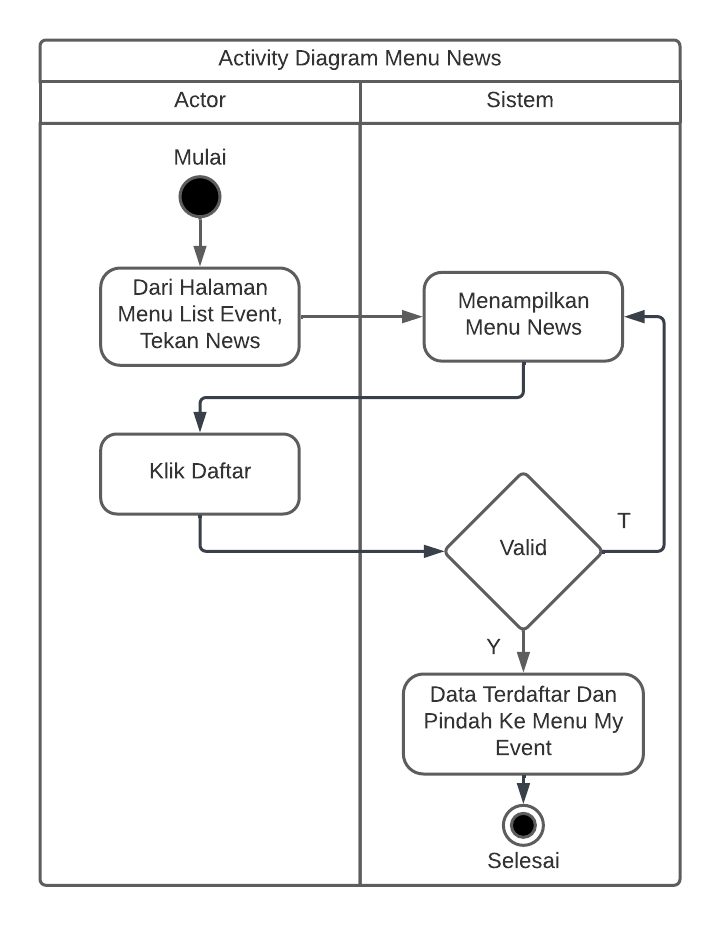
* 1. Login



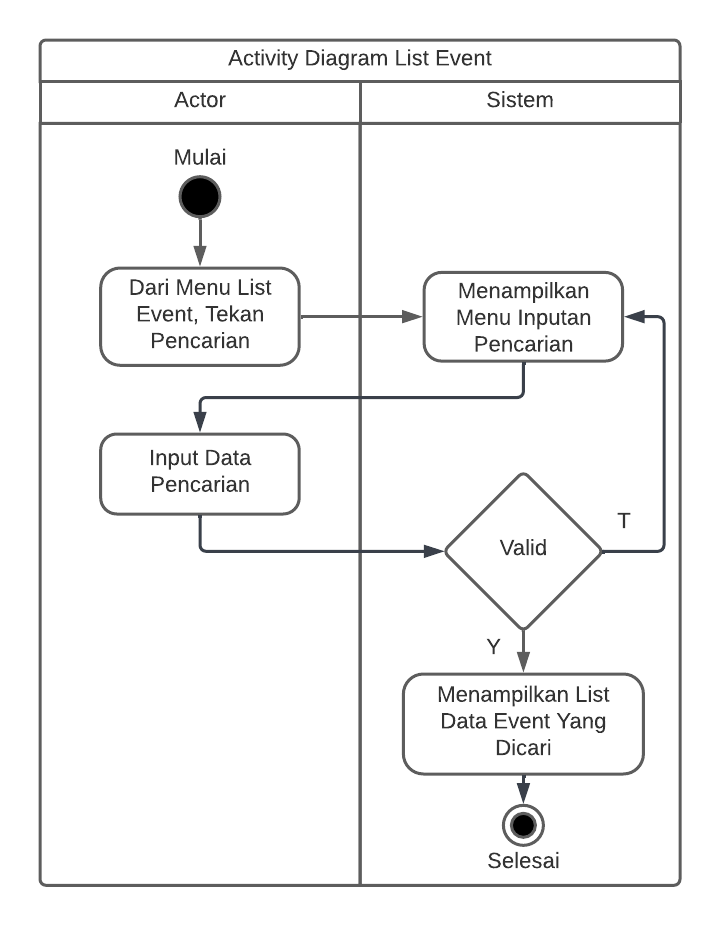
* 1. My Event



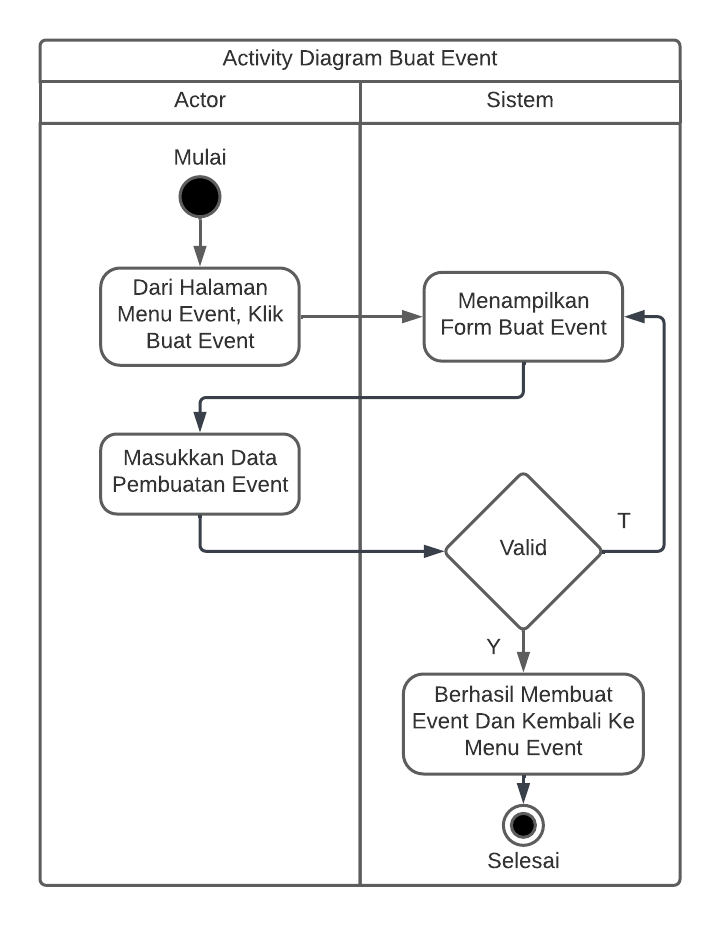
* 1. News



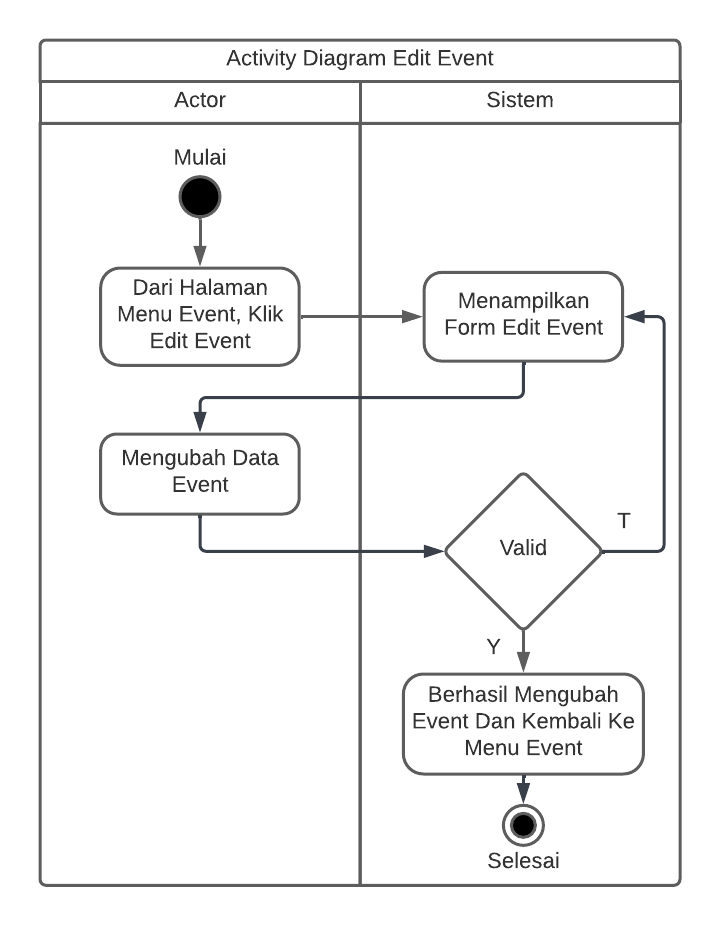
* 1. Event List



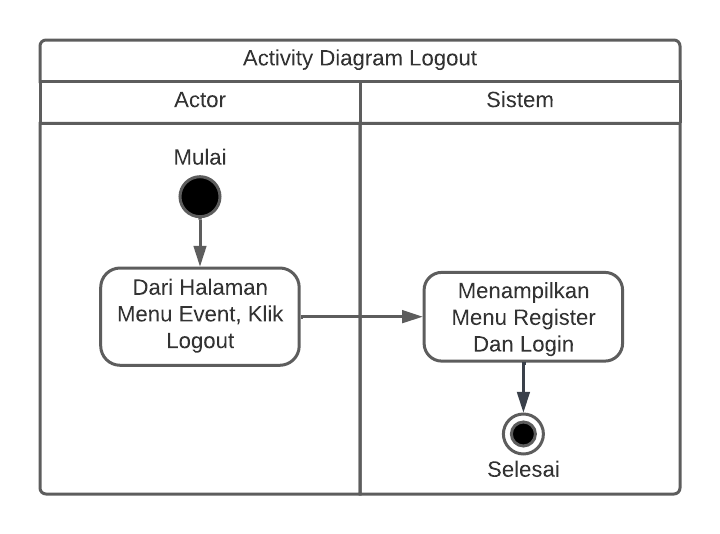
* 1. Create Event



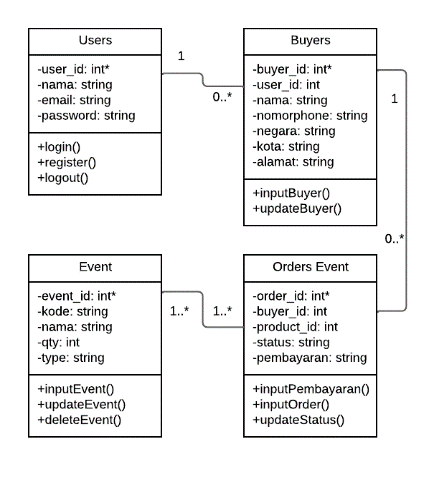
* 1. Edit Event



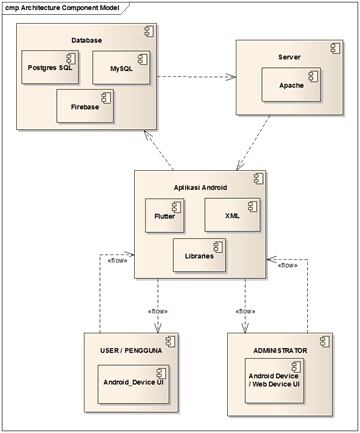
* 1. Logout



* 1. *Class Diagram* (Untuk Perancangan Database)  
     *Class Diagram* adalah model statis yang menggambarkan struktur dan deskripsi *class* serta hubungannya antara *class*. *Class Diagram* mirip *ER-Diagram* pada perancangan database, bedanya pada *ER-Diagram* tidak terdapat operasi/metode tapi hanya atribut. *Class* terdiri dari nama kelas, atribut dan operasi/metode. Berikut ini *Class Diagram* dari Perancangan Aplikasi WeOrganize :



1. MODEL ARSITEKTUR APLIKASI
   1. *Deployment Diagram* Untuk Skema Arsitektur Aplikasi

*Deployment Diagram* digunakan untuk memvisualisasikan hubungan antara *software* dan *hardware*. Secara spesifik *Deployment Diagram* dapat membuat *physical model* tentang bagaimana komponen perangkat lunak digunakan pada komponen perangkat keras, yang dikenal sebagai node. Berikut ini *Deployment Diagram* dari Perancangan Aplikasi WeOrganize :  
  


1. UI / UX APLIKASI

Implementasi UI dari aplikasi kami, dimana terdapat tampilan-tampilan berikut

1. Login

Tampilan login kami menggunakan firebase authentication sebagai backendnya, dan dengan fitur google sign-in jadi tak perlu lagi mengingat password dan email.

A picture containing graphical user interface

Description automatically generated Graphical user interface, text, application

Description automatically generated

1. Memilih role

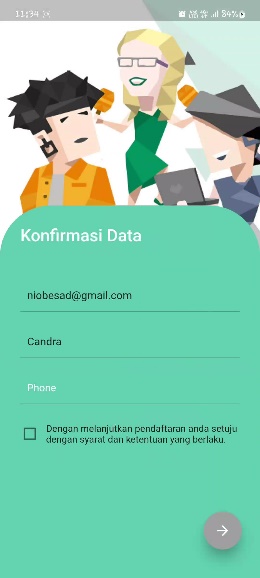
Aplikasi kita bisa digunakan tak hanya oleh partisipan, tapi juga oleh para even organizer hanya perlu memilih role setelah sign in pada aplikasi kami

Text

Description automatically generated

1. Tampilan Konifrmasi Data

Setelah memilih role maka akan ada tampilan untuk konfirmasi data, data Sebagian sudah terisi karena kita menarik data dari backend dalam hal ini firebase, perbedaan antara 2 role ini adalah untuk organizer wajib menyatakan instansi terkait

 A close up of a calculator

Description automatically generated with medium confidence

1. Tampilan Awal Role

Pada tampilan awal role untuk role organizer sama dengan role participant yang membedakan hanya pada tombol floatingbotton untuk membuat event baru yang hanya ada pada role organizer.

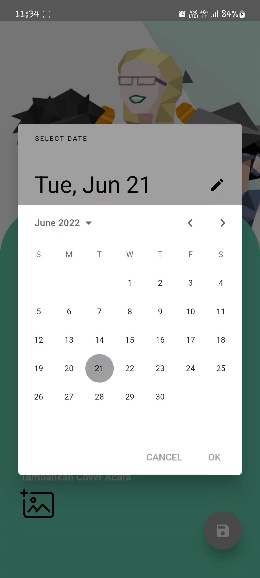
Timeline

Description automatically generated Graphical user interface, text, application

Description automatically generated

1. Tampilan membuat Event

Pada tampilan ini user dengan role organizer bisa membuat event baru dengan bebas memilih tanggal, judul, deskripsi bahkan gambar, berikut tampilanya

 Diagram

Description automatically generated 