SPESIFIKASI KEBUTUHAN PERANGKAT LUNAK

APLIKASI PENCARI LOKASI DONOR DARAH TERDEKAT (DONORIN)

REKAYASA PERANGKAT LUNAK | KELAS INFORMATIKA-I

DAFTAR PERUBAHAN

|  |  |
| --- | --- |
| **Revisi** | **Deskripsi** |
| **A** |  |
| **B** |  |
| **C** |  |
| **D** |  |
| **F** |  |
| **G** |  |

DAFTAR HALAMAN PERUBAHAN

|  |  |  |  |
| --- | --- | --- | --- |
| **Halaman** | **Revisi** | **Halaman** | **Revisi** |
|  |  |  |  |

DAFTAR ISI

[DAFTAR PERUBAHAN i](#_Toc463773300)

[DAFTAR HALAMAN PERUBAHAN iii](#_Toc463773301)

[DAFTAR ISI v](#_Toc463773302)

[DAFTAR TABEL vii](#_Toc463773303)

[DAFTAR DIAGRAM ix](#_Toc463773304)

[DAFTAR GAMBAR xi](#_Toc463773305)

DAFTAR TABEL

Tabel 1.1 Definisi Istilah dan Singkatan 2

Tabel 1.2 Aturan Penomoran Perangkat Lunak 3

Tabel 2.1 Karakteristik User 6

Tabel 3.2 Kebutuhan Fungsional 9

Tabel 3.2.1 Spesifikasi Kebutuhan 11

Tabel 3.2 Kebutuhan Non-Fungsional 13

DAFTAR DIAGRAM

3.5.1 Use Case 14

3.5.3.1 Login (DNRN-1-001) 23

3.5.3.2 Register (DNRN-1-002) 24

3.5.3.3 Logout (DNRN-1-003) 24

3.5.3.4 Isi Identitas (DNRN-1-004) 25

3.5.3.5 Melihat Daftar Event (DNRN-1-005) 25

3.5.3.6 Meminta Donor Darah (DNRN-1-006) 26

3.5.3.7 Menampilkan notifikasi (DNRN-1-007) 26

3.5.3.8 Melihat lokasi event dalam bentuk peta (DNRN-1-008) 27

3.5.3.9 Verifikasi Pendonoran Darah (DNRN-1-009) 27

3.5.3.10 Melihat Riwayat Donor Darah (DNRN-1-010) 28

3.5.3.11 Login (DNRN-1-011) 28

3.5.3.12 Logout (DNRN-1-012) 29

3.5.3.13 Buat Event Baru (DNRN-1-013) 29

3.5.3.14 Menghapus Event (DNRN-1-014) 30

3.5.3.15 Melihat Daftar User (DNRN-1-015) 30

3.5.4 Class Diagram 31

DAFTAR GAMBAR

Gambar 1. Gambaran Umum Sistem Donorin 5

# PENDAHULUAN

Dokumen ini berisi Spesifikasi Kebutuhan Perangkat Lunak (SKPL) untuk Aplikasi Pencari Lokasi Donor Darah Terdekat.

## Deskripsi Umum Dokumen

Dokumen ini secara garis besar terdiri dari tiga bab dengan perincian sebagai berikut :

* Bab 1 Pendahuluan, merupakan pengantar dokumen SPKL yang berisi tujuan penulisan dokumen, lingkup masalah pengembangan perangkat lunak, juga memuat definisi,istilah, dan singkatan serta aturan penomoran.
* Bab 2 Deskripsi Sistem, berisi deskripsi dari sistem yang akan dibuat, fungsi dari sistem tersebut, karakteristik dari User sistem, batasan-batasan dari sistem, dan lingkungan operasi dari sistem.
* Bab 3 Deskripsi Kebutuhan Perangkat Lunak, mendeskripsikan kebutuhan khusus bagi sistem yang akan dibuat yang meliputi kebutuhan antarmuka eksternal,kebutuhan fungsional,kebutuhan non-fungsional,dan pemodelan kebutuhan untuk mendukung agar sistem yang akan dibuat dapat bekerja sesuai dengan yang diharapkan.

## Tujuan Penulisan Dokumen

Dokumen Spesifikasi Kebutuhan Perangkat Lunak ( SKPL ) merupakan dokumen spesifikasi kebutuhan untuk pengembangan Aplikasi Pencari Lokasi Donor Darah Terdekat. Dokumen ini digunakan oleh pengembang perangkat lunak sebagai acuan teknis pengembangan perangkat lunak tahap selanjutnya.

## Lingkup Masalah

Penyebab kematian terbesar di Indonesia adalah akibat kecelakaan, penyakit jantung, kanker, stroke, dan bencana alam (National Center of Health Statistic, 2015) . Dari beberapa kasus tersebut, penyumbang kematian terbesar adalah dari kecelakaan dan bencana alam dikarenakan banyak sekali korban atau pasien yang kehilangan darah akan tetapi jumlah persediaan darah yang ada terbatas atau bahkan tidak ada. Selain kurangnya persediaan darah, susahnya mencari orang yang memiliki golongan darah yang sama adalah salah satu masalah dalam hal ini. Di Indonesia sendiri masih ada beberapa masyarakat yang enggan untuk mendonorkan darah karena kurangnya edukasi kepada masyarakat umum tentang pentingnya donor darah dan beberapa faktor lainnya.

Penggunaan aplikasi mobile untuk memberi informasi serta mengajak masyarakat melakukan donor darah merupakan langkah yang tepat karena pengguna smartphone di Indonesia yang meningkat tiap tahunnya yakni mencapai 86,6 juta pada tahun 2017 (Tech in Asia, 2017). Namun, aplikasi donor darah yang berada di Indonesia hanya sebatas memberi informasi tempat melakukan donor darah terdekat serta penjadwalan kapan User dapat melakukan donor darah.

Memahami hal ini, Donorin adalah aplikasi donor darah yang berjalan di platform Android dan iOS. Aplikasi ini berfungsi untuk mempermudah masyarakat dalam menemukan event donor darah terdekat dan mempermudah mencari pendonor darah di waktu tertentu, Aplikasi ini dapat mengetahui event-event terdekat dalam bentuk daftar, dan juga dapat mengirim notifikasi terhadap User dengan darah yang sesuai dengan pencari darah.

## Definisi, Istilah, dan Singkatan

Definisi dari istilah yang akan digunakan pada dokumen ini yaitu :

Tabel 1.1 Definisi Istilah dan Singkatan

|  |  |  |
| --- | --- | --- |
| No. | Istilah dan Singkatan | Uraian Penjelasan |
| 1 | SKPL | Merupakan Spesifikasi Kebutuhan Perangkat Lunak yaitu spesifikasi dari suatu sistem yang melakukan suatu fungsi tertentu pada lingkungan tertentu |
| 2 | DNRN | Merupakan singkatan dari nama sistem |
| 3 | Login | Proses masuk atau mengakses kedalam sebuah layanan online yang biasanya berisi username dan password |
| 4 | Logout | Proses keluar atau selesai mengakses sebuah layanan online, yang biasanya harus login dengan nama atau password |
| 5 | Register | Proses melakukan pengisian data untuk disimpan kedalam database sebelum melakukan login |
| 6 | Scan | Memindai suatu objek untuk diambil informasinya |
| 7 | Request | Permintaan atau permohonan |
| 8 | Verifikasi | Proses menentukan kebenaran dari suatu pernyataan |
| 9 | Notifikasi | Pemberitahuan |
| 10 | Use case | Sesuatu yang menyediakan hasil yang dapat diukur ke pemakai atau sistem eksternal |
| 11 | Class Diagram | Diagram yang digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada dalam sistem/perangkat lunak yang sedang kita gunakan. |
| 12 | Sequence Diagram | Suatu diagram yang menggambarkan interaksi antar obyek dan mengindikasikan komunikasi antara obyek-obyek tersebut |
| 13 | Activity Diagram | Representasi grafis dari seluruh tahapan alur kerja. Diagram ini mengandung aktivitas, pilihan tindakan, perulangan dan hasil dari aktivitas tersebut |
| 14 | Smartphone | Telepon genggam yang mempunyai layar sentuh dan dapat menjalankan fungsi seperti komputer |
| 15 | Android | Sistem operasi smartphone yang dikembangkan oleh Google. |
| 16 | Credentials | Token yang berfungsi untuk menentukan user yang sudah login |
| 17 | Token | Data berupa huruf, angka, dan simbol acak |
| 18 | Event | Kegiatan Donor Darah |

## Aturan Penomoran

Aturan penomoran yang digunakan sebagai berikut :

**DNRN-X-YYY**

Tabel 1.2 Aturan Penomoran Perangkat Lunak

|  |  |
| --- | --- |
| Hal/Bagian | Deskripsi |
| DNRN | Merupakan singkatan dari nama Aplikasi Pencari Lokasi Donor Darah Terdekat |
| X | Merupakan kode representasi kebutuhan sebagai berikut :  Kebutuhan Fungsional  Kebutuhan Non Fungsional |
| YYY | Merupakan urutan no urut “001” dan seterusnya |

## Referensi

Modul Praktikum Rekayasa Perangkat Lunak, 2017

<https://en.wikipedia.org/wiki/Smartphone>

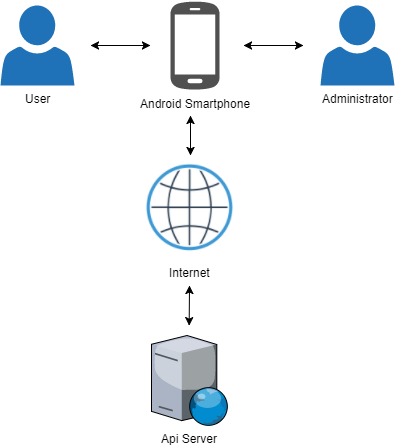
<https://developer.android.com/index.html>

# DESKRIPSI SISTEM

## Deskripsi Umum Sistem

Donorin adalah perangkat lunak yang dikembangkan untuk membantu User yang membutuhkan dan ingin mendonorkan darah. Dalam sistem ini User aplikasi mendapatkan notifikasi sesuai dengan tipe darah pencarinya. Untuk mendapatkan notifikasi, user yang membutuhkan darah perlu menginputkan data pribadi dan tipe darahnya yang telah disediakan dalam sistem ini.

Selain mendapatkan notifikasi, Donorin dapat mengetahui jadwal kapan User dapat berdonor darah lagi. Donorin juga menyediakan event-event terdekat dalam bentuk list yang dapat di filter berdasarkan tanggal dan lokasi.



## Deskripsi Umum Perangkat Lunak

Aplikasi “Donorin” adalah aplikasi di bidang kesehatan dan keselamatan masyarakat yang berjalan di platform Android dan iOS. Dimana aplikasi ini memiliki 4 fungsi utama dengan interface yang dapat dipahami dengan mudah oleh orang awam. 4 fungsi utamanya adalah :

* Daftar Event

Daftar event merupakan fungsi untuk menampilkan daftar event – event donor darah yang ada. Didalam fungsi ini juga terdapat fungsi lain yaitu memfilter event berdasarkan tanggal dan lokasi

* Scan QR Code

Scan QR Code merupakan fungsi untuk melakukan *scan code* jika User telah melakukan donor untuk mendapatkan poin yang dapat ditukar dengan reward.

* Permintaan Darah

Permintaan merupakan fungsi untuk menginputkan data permintaan darah berupa nama, nomor telepon, lokasi rumah sakit yang akan menangani proses pengambilan darah, tipe darah yang dibutuhkan, dan lokasi rumah sakit.

* Riwayat

Riwayat merupakan fungsi untuk menampilkan tampilan riwayat User yang telah melakukan donor darah dan meminta darah kepada pendonor lain melalui aplikasi ini, tampilan pertukaran poin yakni riwayat pertukaran poin yang dimiliki User dengan reward yang tersedia.

## Fungsi Produk/Perangkat Lunak

Pada bagian ini menjelaskan gambaran secara umum fungsi-fungsi yang dapat diakses oleh User dalam aplikasi Donorin. Adapun fungsi-fungsi yang dimiliki oleh aplikasi ini adalah

Sistem dapat menyediakan fungsi untuk masuk ke dalam sistem

Mendaftar ke dalam sistem bagi User

Fungsi untuk keluar dari sistem bagi User

Memasukkan identitas diri bagi User

Melihat daftar event – event donor darah di lokasi terdekat

Meminta donor darah kepada User lain

Menampilkan notifikasi pasien yang membutuhkan darah bagi User

Menampilkan informasi tempat donor darah

Verifikasi pendonoran darah menggunakan *QR code* bagi User

Menampilkan riwayat pendonor

## Karakteristik User

Tabel 2.1 Karakteristik User

|  |  |  |
| --- | --- | --- |
| **No** | **Identifikasi User** | **Karakteristik** |
| 1 | User | * Mendaftar dan masuk ke dalam sistem * Menginputkan identitas diri * Melihat daftar event donor darah * Meminta donor darah * Mendapatkan notifikasi pasien yang membutuhkan darah * Melihat riwayat donor darah |
| 2 | Administrator | * Menambah event donor darah * Menghapus event donor darah * Melihat daftar User yang telah terdaftar |

## Batasan

Batasan – batasan yang ada pada aplikasi Donorin antara lain:

Perangkat yang digunakan User harus terhubung ke internet untuk mengakses aplikasi Donorin

Aplikasi ini hanya dapat digunakan pada platform android

Aplikasi hanya bisa dibuka atau dipakai oleh akun yang telah terdaftar

Informasi event yang digunakan dalam aplikasi Donorin berasal dari PMI dan organisasi lainnya yang telah terverifikasi

## Lingkungan Operasi

Pengembangan dan pengoperasian Donorin dilakukan dengan lingkungan operasi sebagai berikut:

Server

* 1. OS : Mac OS High Sierra
  2. RAM : 8Gb
  3. VGA : Intel graphic 600
  4. Bahasa : Java
  5. DBMS : Firebase

Client

a. OS : Android versi Jelly Bean keatas

b. RAM : Minimal 1 Gb

# DESKRIPSI KEBUTUHAN PERANGKAT LUNAK

## Identifikasi Kebutuhan PL

Salah satu penyebab kematian terbesar di Indonesia adalah kecelakaan yang menyebabkan korban yang kehilangan darah tidak mendapatkan suplai darah tepat waktu. Di Indonesia sendiri masih banyak masyarakat yang tidak paham akan pentingnya donor darah sehingga masih enggan untuk menjadi donor. Terkait dengan masalah tersebut, kami melakukan observasi dengan masyarakat, bagaimana mendapatkan info tentang donor darah dan apakah pernah memerlukan donor darah melalui media sosial atau konvensional.

Memahami hal ini, kami membuat suatu aplikasi bernama Donorin. Aplikasi ini berfungsi untuk mempermudah masyarakat dalam menemukan event donor darah terdekat dan mempermudah mencari pendonor darah di waktu tertentu.

## Kebutuhan Fungsional

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Kode Fungsi** | **Nama Fungsi** | **Deskripsi** |
| 1 | DNRN-1-001 | Login | Sistem harus dapat menyediakan fungsi untuk masuk ke dalam sistem |
| 2 | DNRN-1-002 | Register | Sistem menyediakan fungsi bagi user untuk mendaftar ke dalam sistem dimana user harus menyediakan username minimal 6 karakter, email yang valid, serta password minimal 8 karakter. |
| 3 | DNRN-1-003 | Logout | Sistem menyediakan fungsi bagi user untuk keluar dari sistem dan sistem akan memblokir akses user tersebut sampai login kembali. |
| 4 | DNRN-1-004 | Isi identitas | Sistem menyediakan fungsi bagi user untuk memasukkan beberapa data identitas diri yang terdiri atas nama, No. hp, alamat, tanggal lahir, golongan darah, rhesus, dan tanggal terakhir user melakukan donor darah |
| 5 | DNRN-1-005 | Melihat Daftar event | Sistem menyediakan fungsi untuk melihat event-event donor darah dalam radius 3km dari lokasi terkini user dalam bentuk list. |
| 6 | DNRN-1-006 | Meminta donor darah | Sistem menyediakan fungsi bagi user untuk meminta donor darah kepada user lain yang berada dalam radius 5km dari lokasi user dimana user harus menyediakan informasi, yaitu nama pasien, No. hp pasien, nama rumah sakit tempat pasien dirawat, golongan darah dan rhesus yang dibutuhkan oleh pasien. |
| 7 | DNRN-1-007 | Menampilkan notifikasi | Sistem menyediakan fungsi untuk menampilkan notifikasi bagi para user ketika ada pasien yang membutuhkan darah dalam radius 5km. |
| 8 | DNRN-1-008 | Melihat lokasi event dalam bentuk peta | Sistem menyediakan fungsi untuk menampilkan lokasi event dalam bentuk peta |
| 9 | DNRN-1-009 | Verifikasi pendonoran darah | Sistem menyediakan fungsi untuk menampilkan *QR code* bagi user lainnya yang meminta darah, dan membuka kamera bagi user donor untuk *scan* *QR code* untuk memverifikasi bahwa user donor telah melakukan donor |
| 10 | DNRN-1-010 | Melihat Riwayat Donor Darah | Sistem menyediakan fungsi untuk menampilkan riwayat donor user selama menggunakan aplikasi |
| 11 | DNRN-1-011 | Login | Sistem harus dapat menyediakan fungsi untuk masuk ke dalam sistem |
| 12 | DNRN-1-012 | Logout | Sistem menyediakan fungsi bagi admin untuk keluar dari sistem dan sistem akan memblokir akses admin tersebut sampai login kembali. |
| 13 | DNRN-1-013 | Buat event baru | Sistem menyediakan fungsi untuk membuat event dengan memasukkan nama event dan lokasi event berupa alamat dan koordinat serta tanggal pelaksanaan event |
| 14 | DNRN-1-014 | Menghapus event | Sistem menyediakan fungsi untuk mengapus event yang sudah ada |
| 15 | DNRN-1-015 | Melihat Daftar user | Sistem menyediakan fungsi untuk menampilkan daftar User yang telah terdaftar dalam sistem |

### Spesifikasi Kebutuhan

1. User

|  |  |  |
| --- | --- | --- |
| **No** | **Kode Fungsi** | **Deskripsi** |
| 1 | DNRN-1-001 | * 1. Sistem memberikan form input username   2. Sistem memberikan form input password   3. Sistem memberikan tombol login untuk autentikasi user |
| 2 | DNRN-1-002 | 2.1 Sistem memberikan form input username dengan ketentuan username minimal 6 karakter  2.2 Sistem memberikan form input email  2.3 Sistem memberikan form input password dengan ketentuan password minimal 8 karakter |
| 3 | DNRN-1-003 | 3.1 Sistem memberikan tombol logout untuk mengeluarkan user dari sistem |
| 4 | DNRN-1-004 | 4.1 Sistem memberikan form input nama  4.2 Sistem memberikan form input nomor hp  4.3 Sistem memberikan form input alamat  4.4 Sistem memberikan form input tanggal lahir  4.5 Sistem memberikan form input golongan darah  4.6 Sistem memberikan form input rhesus  4.7 Sistem memberikan form input tanggal terakhir user melakukan donor darah |
| 5 | DNRN-1-005 | 5.1 Sistem memberikan daftar event yang berada dalam radius 3km dari lokasi user  5.2 Sistem memberikan gambar lokasi event  5.3 Sistem memberikan alamat lokasi event  5.4 Sistem memberikan tombol untuk memulai navigasi dari lokasi user saat ini menuju lokasi event |
| 6 | DNRN-1-006 | 6.1 Sistem memberikan form input nama pasien  6.2 Sistem memberikan form input nomor hp pasien  6.3 Sistem memberikan form input nama rumah sakit tempat pasien dirawat  6.4 Sistem memberikan form input golongan darah yang dibutuhkan  6.5 Sistem memberikan form input rhesus yang dibutuhkan  6.6 Sistem memberikan tombol untuk mem-*broadcast* permintaan darah kepada user lain |
| 7 | DNRN-1-007 | 7.1 Sistem memberikan notifikasi ketika ada pasien yang membutuhkan darah |
| 8 | DNRN-1-008 | 8.1 Sistem memberikan tampilan peta  8.2 Sistem memberikan penanda pada peta yang menandakan lokasi event |
| 9 | DNRN-1-009 | 9.1 Sistem memberikan gambar *QR code*  untuk di-*scan* oleh donor  9.2 Sistem membuka kamera untuk me-*scan* *QR code* yang ditampilkan oleh pasien |
| 10 | DNRN-1-010 | 10.1 Sistem memberikan daftar riwayat donor darah user  10.2 Sistem memberikan tanggal riwayat donor darah  10.3 Sistem memberikan tempat riwayat donor darah user |

1. Administrator

|  |  |  |
| --- | --- | --- |
| **No** | **Kode Fungsi** | **Deskripsi** |
| 1 | DNRN-1-011 | 11.1 Sistem memberikan form input username  11.2 Sistem memberikan form input password  11.3 Sistem memberikan tombol login untuk memberikan akses ke dalam sistem |
| 2 | DNRN-1-012 | 3.1 Sistem memberikan tombol logout untuk mengeluarkan admin dari sistem |
| 3 | DNRN-1-013 | 12.1 Sistem memberikan form input nama event  12.2 Sistem memberikan form input alamat event  12.3 Sistem memberikan tombol untuk memilih koordinat event melalui peta  12.4 Sistem memberikan form input tanggal pelaksanaan event |
| 4 | DNRN-1-014 | 13.1 Sistem memberikan daftar event yang telah dibuat sebelumnya  13.2 Sistem memberikan tombol untuk menghapus event yang bersangkutan |
| 5 | DNRN-1-015 | 14.1 Sistem memberikan daftar User yang telah terdaftar dalam sistem  14.2 Sistem menampilkan nama User  14.3 Sistem menampilkan nomor hp User  14.4 Sistem menampilkan golongan darah User |

## Kebutuhan Non Fungsional

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Kode** | **Parameter** | **Deskripsi Kebutuhan** |
| 1 | DNRN-2-001 | Performa | Sistem harus berjalan selama 7 hari 24 jam secara nonstop. |
| 2 | DNRN-2-002 | Atribut | Sistem menggunakan bahasa Indonesia sesuai dengan EYD |
| 3 | DNRN-2-003 | Performa | Sistem mampu melayani minimal 50 User per detik. |

## Kebutuhan Antarmuka Eksternal

### Antarmuka User

User menggunakan user infterface berbasis aplikasi mobile. User berinteraksi dengan DNRN melalui user interface aplikasi android. Selanjutnya perangkat lunak akan menerima inputan dari User melalui tulisan atau pilihan yang di ketik oleh keyboard serta pilihan dan perintah melalui layar touchscreen. Hasil keluaran dari sistem dapat dilihat oleh User melalui layar smartphone android User.

### Antarmuka Perangkat Keras

Perangkat lunak ini membutuhkan perangkat keras *smartphone Android* dengan layar sentuh.

### Antarmuka Perangkat Lunak

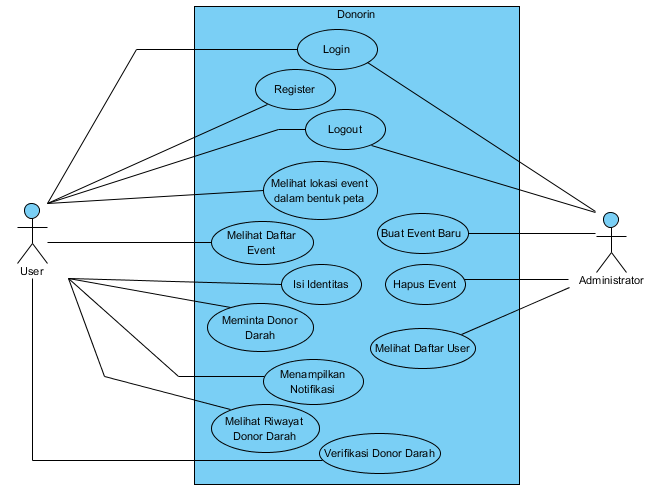
Perangkat lunak ini membutuhkan API dari sistem lain untuk menyimpan dan mengambil data.

### Antarmuka Komunikasi

Perangkat lunak ini membutuhkan internet untuk dapat berkomunikasi dengan API Server.

## Pemodelan Kebutuhan

### Use Case



### Use Case Scenario

|  |  |
| --- | --- |
| **Login [DNRN-1-001]** | |
| Objective | User masuk ke dalam sistem. |
| Actor | User |
| Pre-Condition | User telah membuka aplikasi Donorin dan terkoneksi dengan internet |
| Main Flow | 1. Sistem menampilkan halaman login 2. User memasukkan username dan password yang telah terdaftar sebelumnya 3. User memilih tombol Login 4. Sistem menerima input user dan melakukan proses login |
| Alternative Flows | 1. Jika user tidak mengisi username dan password pada saat mengklik tombol login, maka sistem akan menampilkan pemberitahuan kesalahan memasukkan data. |
| Post-Condition | Sistem telah menampilkan halaman profile yang terdapat identitas diri |

|  |  |
| --- | --- |
| **Register [DNRN-1-002]** | |
| Objective | User mendaftar ke dalam sistem |
| Actor | User |
| Pre-Condition | User telah membuka halaman register dan terkoneksi dengan internet |
| Main Flow | 1. Aktor memilih menu register 2. Aktor memasukkan alamat email yang valid, username minimal 6 karakter, dan password minimal 8 karakter 3. Aktor memilih tombol register 4. Sistem menerima input aktor dan melakukan proses register |
| Alternative Flows | 1. Jika aktor tidak mengisi username,email, dan password sesuai aturan pada saat mengklik tombol register, maka sistem akan menampilkan pemberitahuan kesalahan memasukkan data. |
| Post-Condition | Sistem telah menampilkan halaman isi identitas |

|  |  |
| --- | --- |
| **Logout [DNRN-1-003]** | |
| Objective | User keluar dari sistem dan akses aktor tersebut diblokir sampai login kembali |
| Actor | User |
| Pre-Condition | User telah login ke dalam sistem dan terkoneksi dengan internet |
| Main Flow | 1. Aktor memilih menu profile 2. Aktor memilih tombol logout 3. Sistem menerima input aktor dan sistem menghapus token |
| Alternative Flows | - |
| Post-Condition | Sistem telah menampilkan halaman login |

|  |  |
| --- | --- |
| **Isi Identitas [DNRN-1-004]** | |
| Objective | User memasukkan identitas diri |
| Actor | User |
| Pre-Condition | User telah login ke dalam sistem dan terkoneksi dengan internet |
| Main Flow | 1. Aktor memilih menu profile 2. Aktor memilih tombol edit profile 3. Sistem menampilkan form edit profile 4. Aktor mengisi form identitas 5. Aktor memilih tombol edit 6. Sistem menyimpan data identitas aktor |
| Alternative Flows | Jika aktor tidak mengisi salah satu identitas, maka sistem akan menampilkan pemberitahuan kesalahan memasukkan data  Jika koneksi internet mati, maka sistem akan menampilkan pemberitahuan kesalahan |
| Post-Condition | Sistem telah menampilkan halaman profile |

|  |  |
| --- | --- |
| **Melihat Daftar Event [DNRN-1-005]** | |
| Objective | User dapat melihat event – event donor darah dalam radius 3 km dari lokasi user |
| Actor | User |
| Pre-Condition | User telah login ke dalam sistem dan terkoneksi dengan internet |
| Main Flow | 1. Aktor memilih menu event 2. Aktor memilih tombol filter berdasarkan tanggal maupun lokasi 3. Sistem melakukan proses filter |
| Alternative Flows | - |
| Post-Condition | Sistem telah menampilkan daftar event |

|  |  |
| --- | --- |
| **Meminta Donor Darah [DNRN-1-006]** | |
| Objective | User dapat meminta donor darah kepada user lainnya |
| Actor | User |
| Pre-Condition | Aktor telah masuk dalam menu event dan terkoneksi dengan internet |
| Main Flow | 1. Aktor memilih ikon bergambar darah 2. Sistem menampilkan form permintaan darah 3. Aktor memasukkan data permintaan darah berupa nama, nomor telepon, lokasi rumah sakit, dan golongan darah yang dibutuhkan 4. Sistem menerima input aktor dan melakukan proses permintaan darah |
| Alternative Flows | 1. Jika aktor tidak mengisi salah satu data permintaan, maka sistem akan menampilkan pemberitahuan kesalahan memasukkan data 2. Jika koneksi internet mati, maka sistem akan menampilkan pemberitahuan kesalahan |
| Post-Condition | Sistem telah menampilkan status permintaan dan mengirimkan notifikasi kepada User lain |

|  |  |
| --- | --- |
| **Menampilkan notifikasi [DNRN-1-007]** | |
| Objective | Menampilkan notifikasi bagi user ketika ada pasien yang membutuhkan darah |
| Actor | User |
| Pre-Condition | Aktor telah masuk kedalam sistem dan terkoneksi dengan internet |
| Main Flow | 1. User menerima pop up notifikasi pada *smartphone* 2. Sistem melakukan proses untuk mengakses detail permintaan darah |
| Alternative Flows | - |
| Post-Condition | Sistem telah menampilkan detail permintaan darah |

|  |  |
| --- | --- |
| **Melihat lokasi event dalam bentuk peta [DNRN-1-008]** | |
| Objective | Sistem menampilkan lokasi event dalam bentuk peta |
| Actor | User |
| Pre-Condition | User telah masuk kedalam menu event dan terkoneksi dengan internet |
| Main Flow | 1. User memilih menu event yang telah ditampilkan sistem 2. Sistem melakukan proses mencari lokasi |
| Alternative Flows | - |
| Post-Condition | Sistem telah menampilkan lokasi event dalam bentuk peta |

|  |  |
| --- | --- |
| **Verifikasi Pendonoran Darah [DNRN-1-009]** | |
| Objective | Menampilkan *QR code* bagi user yang meminta darah dan membuka kamera bagi user donor untuk *scan* *QR code* |
| Actor | User |
| Pre-Condition | User telah masuk ke dalam aplikasi dan terkoneksi dengan internet |
| Main Flow | 1. User memilih menu *Scan QR* 2. Sistem menampilkan *QR code* yang dapat di *scan* 3. User memilih ikon kamera untuk melakukan *scan code* 4. Sistem melakukan proses *scanning* untuk verifikasi bahwa user telah melakukan donor |
| Alternative Flows | - |
| Post-Condition | Sistem telah menampilkan status verifikasi |

|  |  |
| --- | --- |
| **Melihat Riwayat Donor Darah [DNRN-1-010]** | |
| Objective | Menampilkan riwayat donor user selama menggunakan aplikasi |
| Actor | User |
| Pre-Condition | User telah masuk ke dalam menu profile dan terkoneksi dengan internet |
| Main Flow | 1. User memilih tombol history yang terdapat pada halaman profile 2. Sistem mengakses database user aplikasi |
| Alternative Flows | - |
| Post-Condition | Sistem telah menampilkan tampilan riwayat User yang telah melakukan donor dan meminta darah |

|  |  |
| --- | --- |
| **Login [DNRN-1-011]** | |
| Objective | Administrator dapat masuk ke dalam sistem. |
| Actor | Administrator |
| Pre-Condition | Administrator telah membuka aplikasi Donorin dan terkoneksi dengan internet |
| Main Flow | 1. Sistem menampilkan halaman login 2. Administrator memasukkan username dan password yang telah terdaftar sebelumnya 3. Administrator memilih tombol Login 4. Sistem menerima input administrator dan melakukan proses login |
| Alternative Flows | 1. Jika administrator tidak mengisi username dan password pada saat mengklik tombol login, maka sistem akan menampilkan pemberitahuan kesalahan memasukkan data. |
| Post-Condition | Sistem telah menampilkan halaman utama admin |

|  |  |
| --- | --- |
| **Logout [DNRN-1-012]** | |
| Objective | Administrator dapat keluar dari sistem dan akses administrator tersebut diblokir sampai login kembali |
| Actor | Administrator |
| Pre-Condition | Administrator telah login ke dalam sistem dan terkoneksi dengan internet |
| Main Flow | 1. Administrator memilih menu profile 2. Administrator memilih tombol logout 3. Sistem menerima input administrator dan sistem menghapus token |
| Alternative Flows | - |
| Post-Condition | Sistem telah menampilkan halaman login |

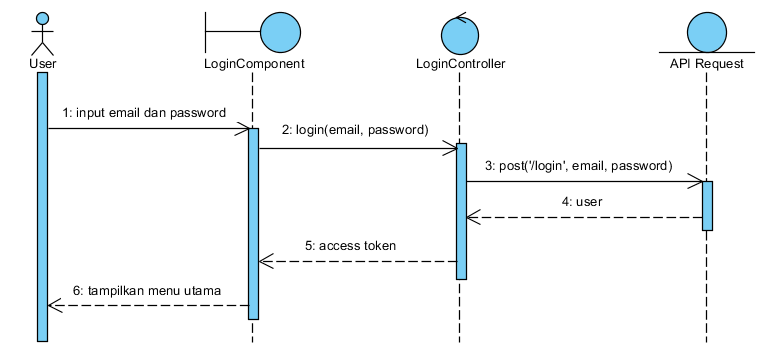
|  |  |
| --- | --- |
| **Membuat Event Baru [DNRN-1-013]** | |
| Objective | Administrator dapat membuat event baru dengan memasukkan nama dan lokasi |
| Actor | Administrator |
| Pre-Condition | Administrator telah masuk ke menu event dan terkoneksi dengan internet |
| Main Flow | 1. Sistem memberikan form input nama dan alamat event 2. Administrator memasukkan nama dan alamat 3. Sistem memberikan tombol untuk memilih koordinat event melalui peta 4. Administrator memilih tombol OK |
| Alternative Flows | - |
| Post-Condition | Sistem telah membuat event baru dan halaman event ditunjukkan |

|  |  |
| --- | --- |
| **Menghapus Event [DNRN-1-014]** | |
| Objective | Administrator dapat menghapus event yang sudah ada |
| Actor | Administrator |
| Pre-Condition | Administrator telah masuk ke dalam menu event dan terkoneksi dengan internet |
| Main Flow | 1. Sistem memberikan daftar event yang telah dibuat 2. Sistem memberikan tombol untuk menghapus event 3. Administrator memilih tombol hapus 4. Sistem menerima perintah dari administrator dan melakukan penghapusan |
| Alternative Flows | - |
| Post-Condition | Sistem telah menghapus event yang sudah ada |

|  |  |
| --- | --- |
| **Menampilkan Daftar User [DNRN-1-015]** | |
| Objective | Menampilkan daftar user yang telah terdaftar dalam sistem |
| Actor | Administrator |
| Pre-Condition | Administrator telah masuk kedalam sistem admin dan terkoneksi dengan internet |
| Main Flow | 1. Administrator memilih menu menampilkan Daftar User 2. Sistem memberikan daftar User yang telah terdaftar dalam asisten |
| Alternative Flows | - |
| Post-Condition | Sistem telah menampilkan daftar User yang telah terdaftar |

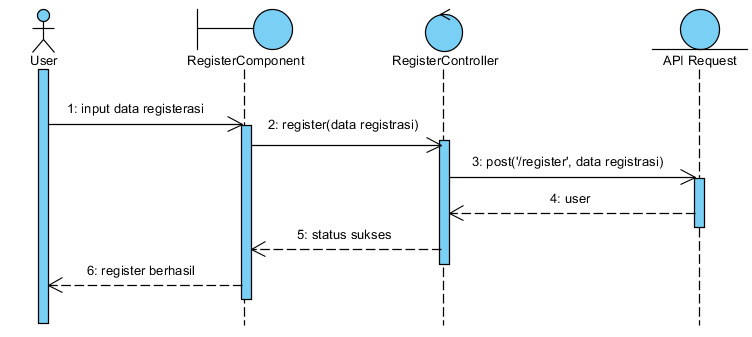
### Sequence Diagram

#### Login (DNRN-1-001)



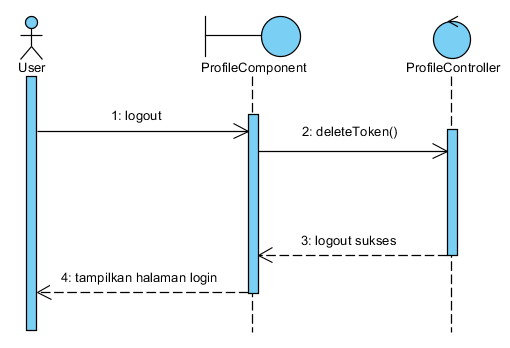
Ketika proses login, setelah memasukkan email dan password untuk login, akan dilakukan pengiriman untuk mengambil data user yang sesuai dengan credentials yang dimasukkan. Jika ada maka API akan mengembalikan token dan login sukses. Jika tidak ada, maka API akan mengembalikan null dan maka login gagal.

#### Register (DNRN-1-002)



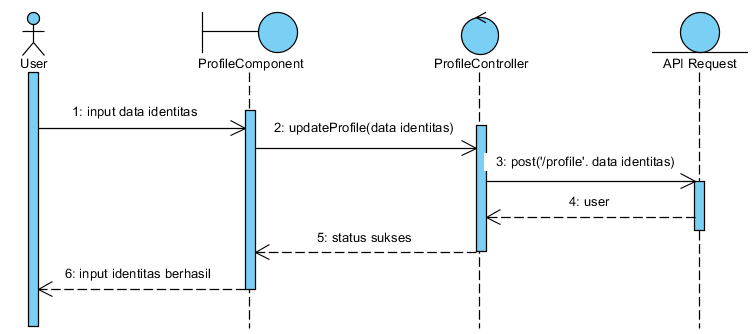
User mengakses RegisterComponent kemudian mengisi data yang diperlukan, data akan dikirimkan ke RegisterController yang kemudian diproses dan dikirimkan ke API. API akan mengembalikan kembali data yang sudah dimasukkan tersebut dan register berhasil.

#### Logout (DNRN-1-003)



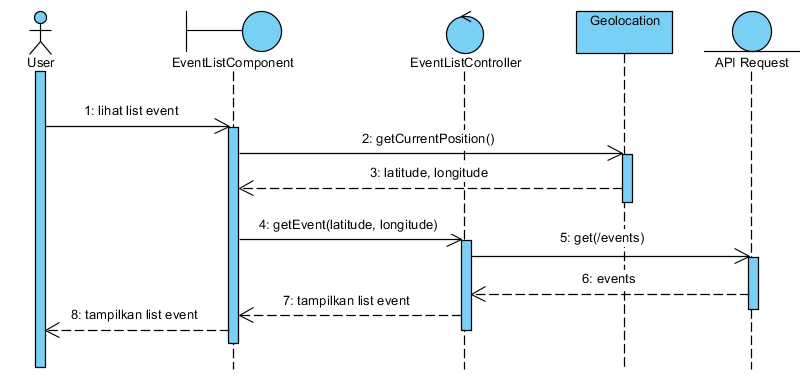
Ketika user ingin keluar dari aplikasi, user akan melakukan logout dengan cara menekan tombol logout. Kemudian akan memamnggil method deleteToken yang ada pada ProfileController yang menyebabkan tidak ada nya data credentials pada aplikasi. Kemudian diarahkan ke halaman login kembali.

#### Isi Identitas (DNRN-1-004)



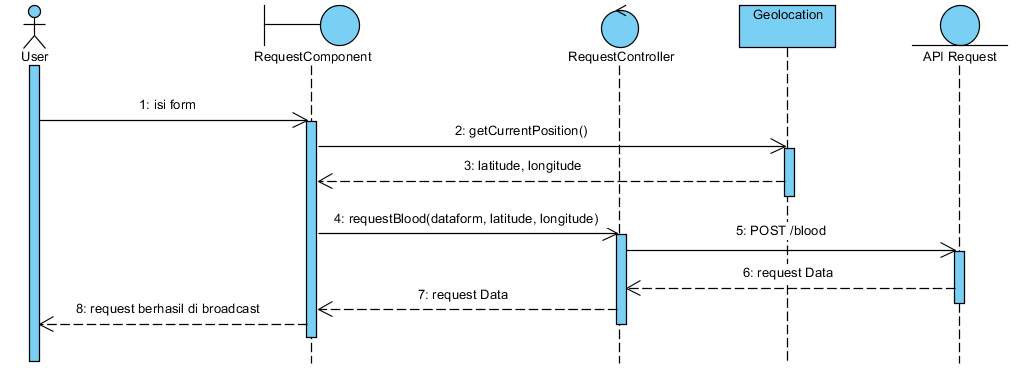
User harus mengisi identitas terlebih dahulu pada form di ProfileComponent, kemudian identitas tersebut akan dikirimkan pada method updateProfile pada ProfileController. ProfileController akan mem-POST data tersebut ke API, dan akan didapatkan balasan berupa data user yang sudah diperbaharui.

#### Melihat Daftar Event (DNRN-1-005)



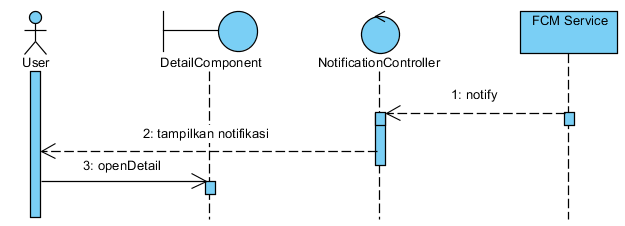
Pada EventListComponent user dapat melihat daftar event terdekat dari user tersebut. EventListComponent akan meminta lokasi saat ini berupa latitude dan longitude dari objek Geolocation yang akan menentukan lokasi dari GPS *smartphone.* Setelah didapatkannya lokasi, akan dipanggil method getEvent dengan lokasi sebelumnya yang akan dikirimkan ke API, sehingga API akan mengembalikan daftar event-event terdekat sesuai dengan lokasi yang sudah dikirimkan sebelumnya.

#### Meminta Donor Darah (DNRN-1-006)



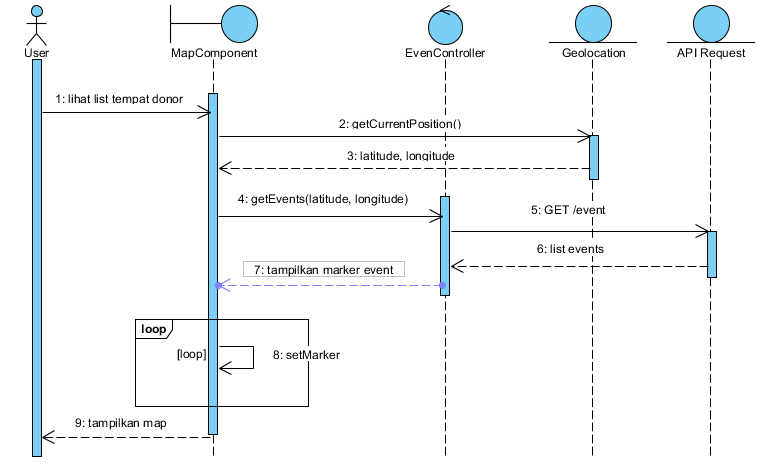
Setelah user menekan tombol meminta donor darah, user akan ditampilkan form pada RequestComponent yang harus diisi, sekaligus RequestComponent akan mengambil lokasi dari class Geolocation yang akan menentukan lokasi dari GPS *smartphone.* Data yang sudah dimasukkan user dan juga lokasi nya akan dipanggil ke method requestBlood yang akan mem-POST data tersebut ke API sehingga data akan di proses pada server. Setelah itu server akan mengambalikan lagi request yang sudah dibuat, dan tampilkan bahwa request sudah di broadcast.

#### Menampilkan notifikasi (DNRN-1-007)



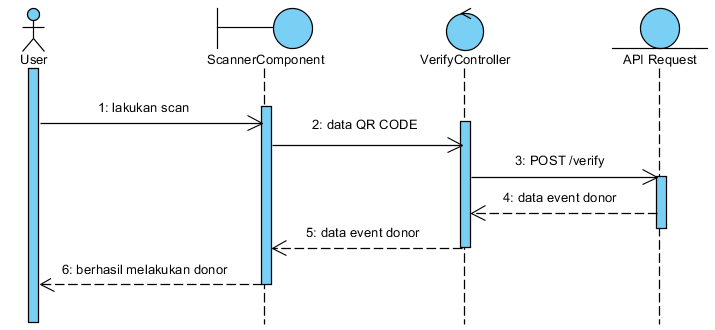
Firebase Cloud Messaging (FCM) akan menunggu notifikasi dari server. Jika ada maka FCM Service akan mengirimkan data notifikasi tersebut ke NotificationController yang akan menampilkan notifikasi pada Android user. Jika user menekan notifikasi tersebut, akan membuka DetailComponent yang menampilkan detail dari notifikasi yang didapat.

#### Melihat lokasi event dalam bentuk peta (DNRN-1-008)



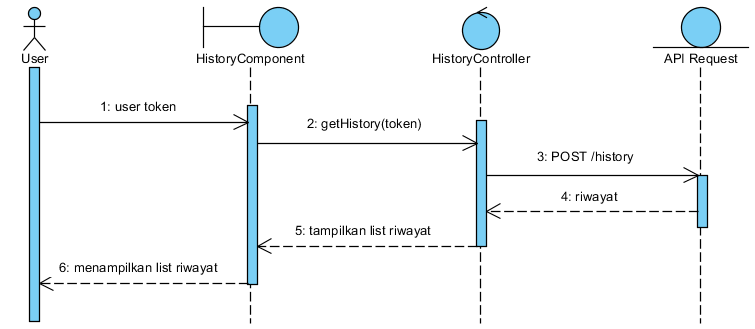
User dapat melihat daftar lokasi event terdekat dalam bentuk peta pada MapComponent. MapComponent akan mengambil lokasi dari class Geolocation yang akan menentukan lokasi dari GPS *smartphone.* Setelah itu akan memanggil method getPlaces pada EventListController yang akan meng-GET data event dari API. API akan mengembalikan daftar event dalam bentuk array latitude dan longitude, yang akan diiterasi untuk ditampilkan marker pada map.

#### Verifikasi Pendonoran Darah (DNRN-1-009)



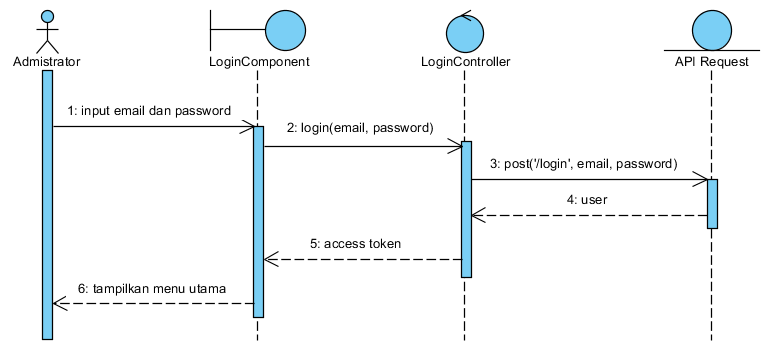
Setelah user melakukan donor darah pada event yang dilaksanakan, user dapat melakukan scan QR code yang disediakan. User bisa mengarahkan kamera yang ditunjukkan pada ScannerComponent. Jika terdeteksi QR Code maka data QR tersebut akan dikirimkan pada VerifyController yang akan mem-POST data tersebut ke API. API akan mengembalikan data event donor dan hasil update donor user terbaru.

#### Melihat Riwayat Donor Darah (DNRN-1-010)



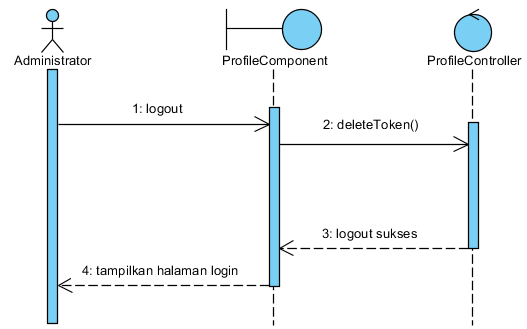
User dapat melihat donor darah yang sudah dilakukan. Pada saat user membuka HistoryComponent, akan memanggil method getHistory yang akan meng-GET riwayat dari user, yang kemudian akan ditampilkan dalam bentuk daftar.

#### Login (DNRN-1-011)



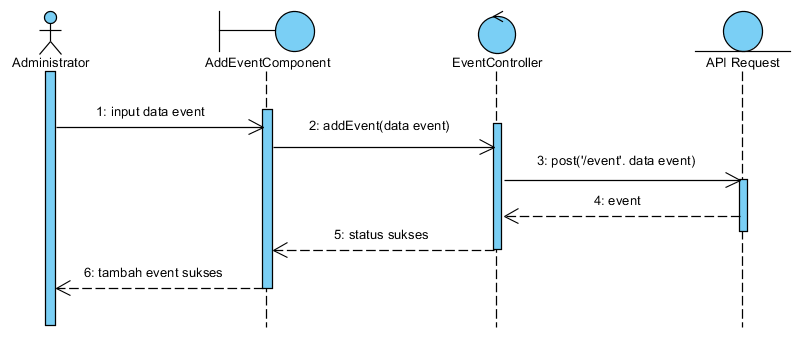
Ketika proses login, setelah memasukkan email dan password untuk login, akan dilakukan pengiriman untuk mengambil data user yang sesuai dengan credentials yang dimasukkan. Jika user bertipe administrator, maka akan menampilkan menu utama administrator.

#### Logout (DNRN-1-012)



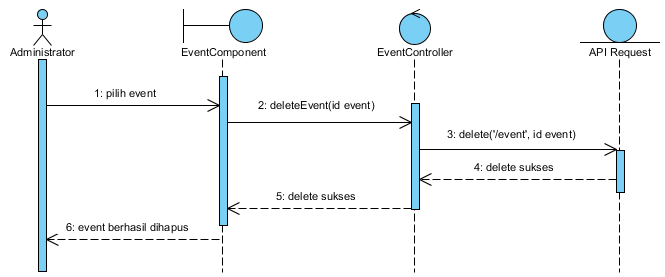
Ketika administrator ingin keluar dari aplikasi, administrator akan melakukan logout dengan cara menekan tombol logout. Kemudian akan memamnggil method deleteToken yang ada pada ProfileController yang menyebabkan tidak ada nya data credentials pada aplikasi. Kemudian diarahkan ke halaman login kembali.

#### Buat Event Baru (DNRN-1-013)



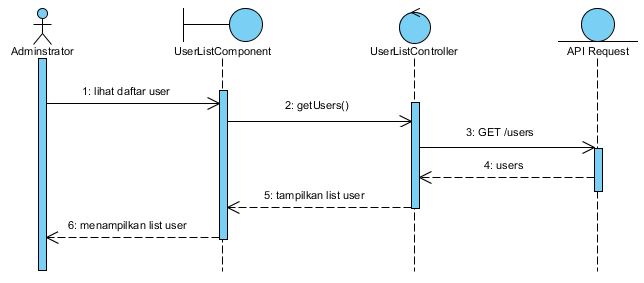
Administrator menginput data event berupa nama event, alamat dan koordinat serta tanggal pelaksanaan event pada addEventComponent. Kemudian data tersebut akan dikirimkan ke EventController yang akan mem-POST data tersebut ke API dan akan didapatkan balasan berupa data event yang sudah dibuat.

#### Menghapus Event (DNRN-1-014)



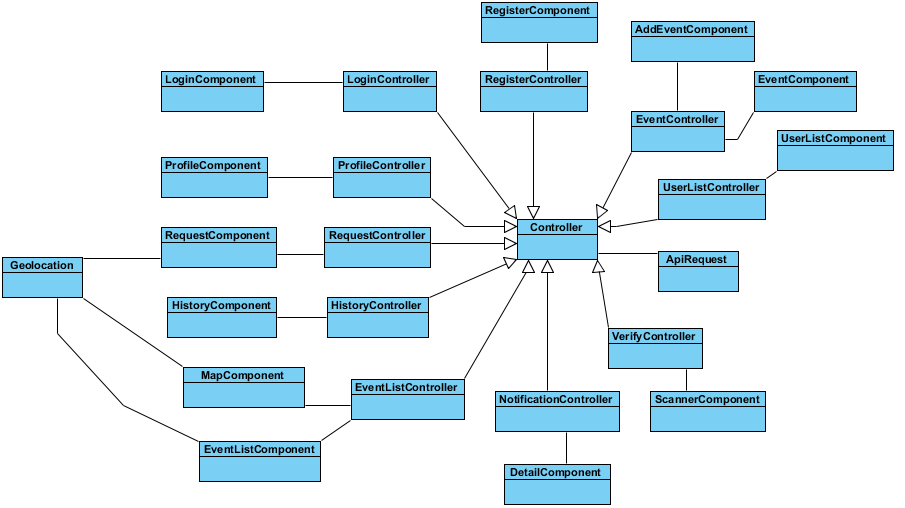
Administrator memilih event yang ingin dihapus pada EventComponent. Kemudian ide dari event tersebut akan dikirimkan ke method deleteEvent pada EventController yang akan mem-DELETE ide event tersebut ke API dan akan didapatkan balasan bahwa status delete sudah berhasil.

#### Melihat Daftar User (DNRN-1-015)



Administrator bisa melihat daftar user pada UserListComponent. Daftar tersebut didapat dari method getUsers dari UserListController yang meng-GET data users. Data users dari API akan dikembalikan lagi ke UserListController yang akan menampilkan datanya pada UserListComponent.

### Class Diagram



Di dalam class diagram diatas, terdapat beberapa Class Controller diantaranya LoginController, RegisterController, ProfileController, RequestController, HistoryController, EventListController, NotificationController, ScannerComponent, EventController dan UserListController. Dimana class-class tersebu dapat memanggil class ApiRequest untuk melakukan panggilan API ke server. Component View RequestComponent, MapComponent, dan EventListComponent dapat memanggil class Geolocation untuk mendapatkan lokasi GPS dari perangkat.