Nama: Muhammad Rafif

NIM: 2440057741

**JAWABAN:**

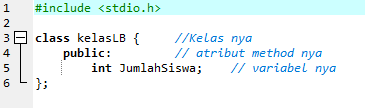
1. **A**

|  |  |
| --- | --- |
| Class | Object |
| Kelas merupakan blueprint yang dimana bisa dibentuk secara instant. | Obyek merupakan turunan kelas, yang membantu programmer menggunakan variable dan metode dari dalam kelas |
| Kelas digunakan untuk menggabungkan data dan metode Bersama sebagai satu unit | Bertindak seperti variable kelas |
| Keberadaan nya logis | Keberadaannya berbentuk fisik |
| Tidak memakan memori saat dibentuk | Memakan memori |
| Bisa digunakan hanya sekali | Bisa digunakan berkali kali bergantung berapa banyak yang dibutuhkan |

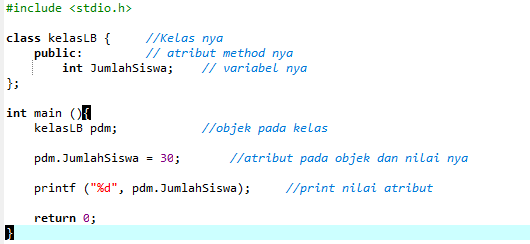
|  |
| --- |
| Persamaan |
| Sebagai Benda dalam deskripsi masalah |
| Mempunyai atribut dan metodoe |
| Sama sama membutuhkan |
| Dibentuk secara instan yang representasi nya nyata |
| Sama sama akan menjadi unit atau entitas dalam sebuah program |

**B.**

Class :



Object:



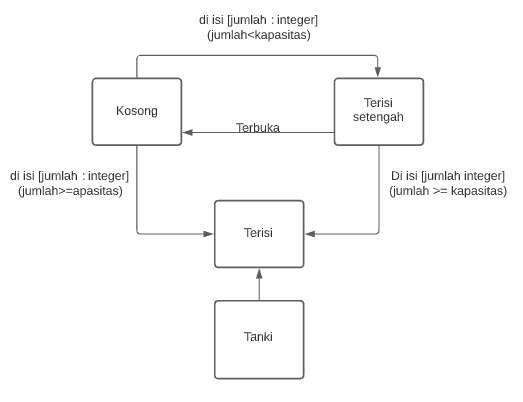
Contoh lain Dari Kelas dan Objek

|  |  |
| --- | --- |
| Kelas | Objek |
| Motor | Motor Honda |
| Mobil | Mobil Sedan |
| Buah Buahan | Buah Mangga |
| Sayur Sayuran | Sayur Nangka |

Referensi : <https://www.ncl.ucar.edu/Document/HLUs/User_Guide/classes/classoview.shtml>

1. **A.** Menurut saya tergantung, apabila state diagram tersebut untuk membuat system seperti atm, memasukkan password, memasukkan email dan sebagai nya self transition sangatlah penting karena dengan adanya self transition apabila user salah memasukkan pin, password dll, self transition bisa membalikkan keadaan sehingga user bisa mengulang lagi dengan batas yang sudah ditentukan oleh program. Dengan kata lain berarti self transition tidak dibutuhkan di setiap state diagram.

**B.** Contoh nya adalah tanki air, dalam state tanki air ini ada 2 yaitu dikosongkan atau di isi setengah yang menghasilkan event tersebut merespon menjadi 2 state yang berbeda.

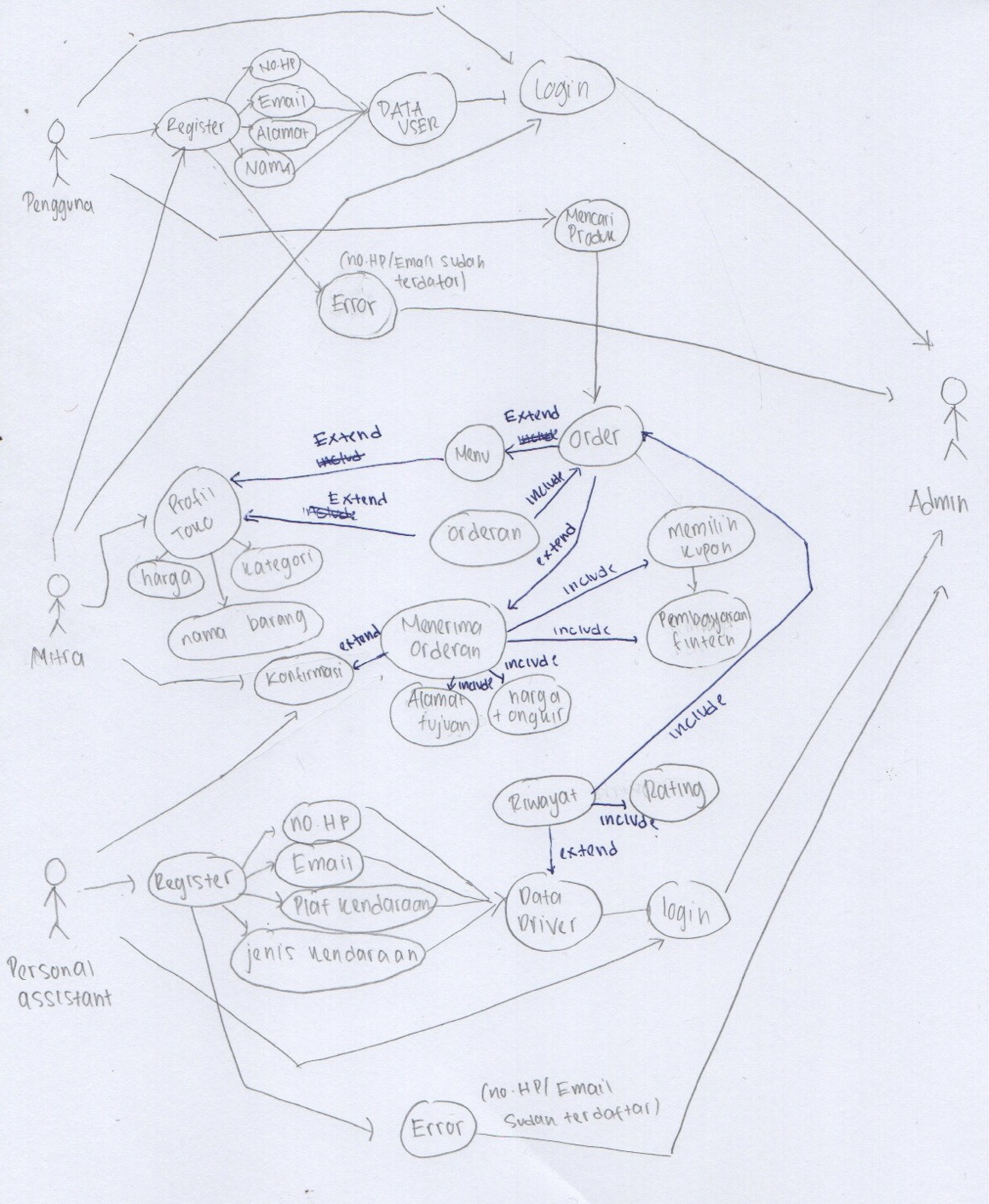


1. **Component diagram:** Digunakan untuk menggambarkan distribusi fisik dari modul perangkat lunak melalui jaringan. Misalnya, Ketika merancang system client-server, hal ini berguna untuk menunjukkan mana kelas atau paket kelas akan berada pada node klien dan mana yang akan berada dalam server.

**Deployment diagram:** menggambarkan detail bagaimana komponen di deploy dalam infrastruktur system, dimana komponen akan terletak pada mesin atau server, yang sebagaimana kemampuan jaringan pada lokasi tsb, spesifikasi serever, dll yang bersifat fisik. Hubungan antar node (misalnya TCP/IP) dan requirement(kebutuhan) dapat juga didefinisikan dalam diagram ini.

**K**

**II. Kasus**



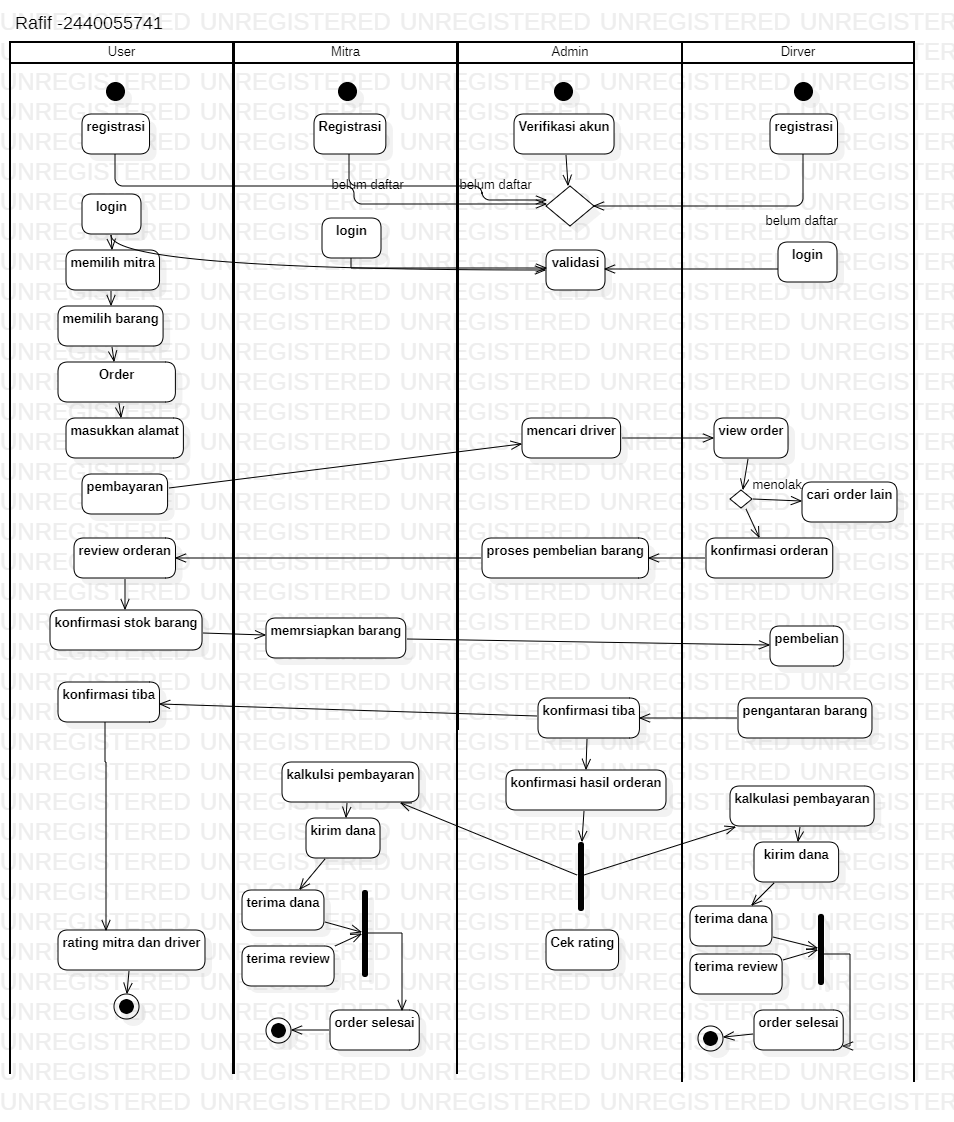
|  |
| --- |
| User |
| -Nama  -No HP  -Alamat rumah  - Email  -Riwayat Orderan |

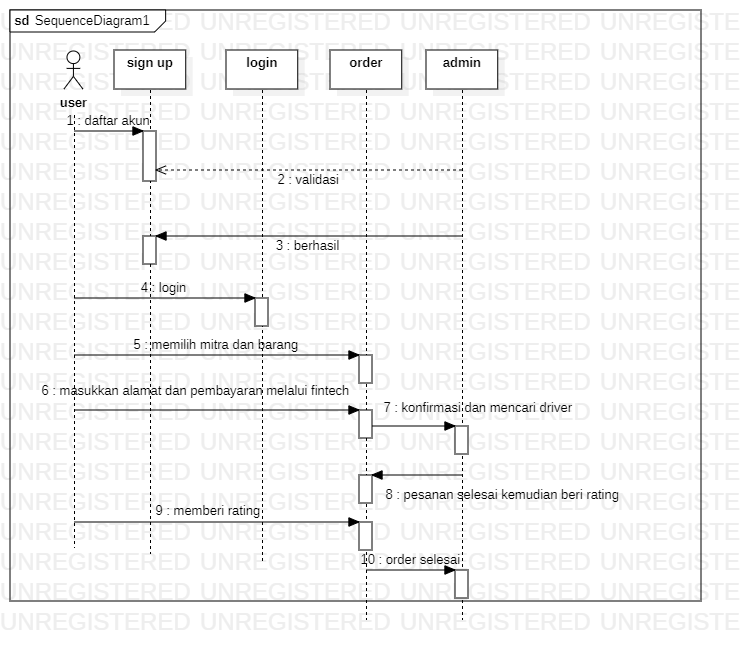
|  |
| --- |
| Mitra |
| -Nama Mitra  - No HP  -Alamat  -Email  -Produk  -Harga produk  -kuantitas produk  -Rating Mitra |

|  |
| --- |
| Sistem (admin) |
| -mempunyai data user (bisa di block dan unblock)  -mempunyai data Mitra (bisa di block dan unblock)  -mempunyai data Driver (bisa di block dan unblock)  -Menerima verifikasi orderan |

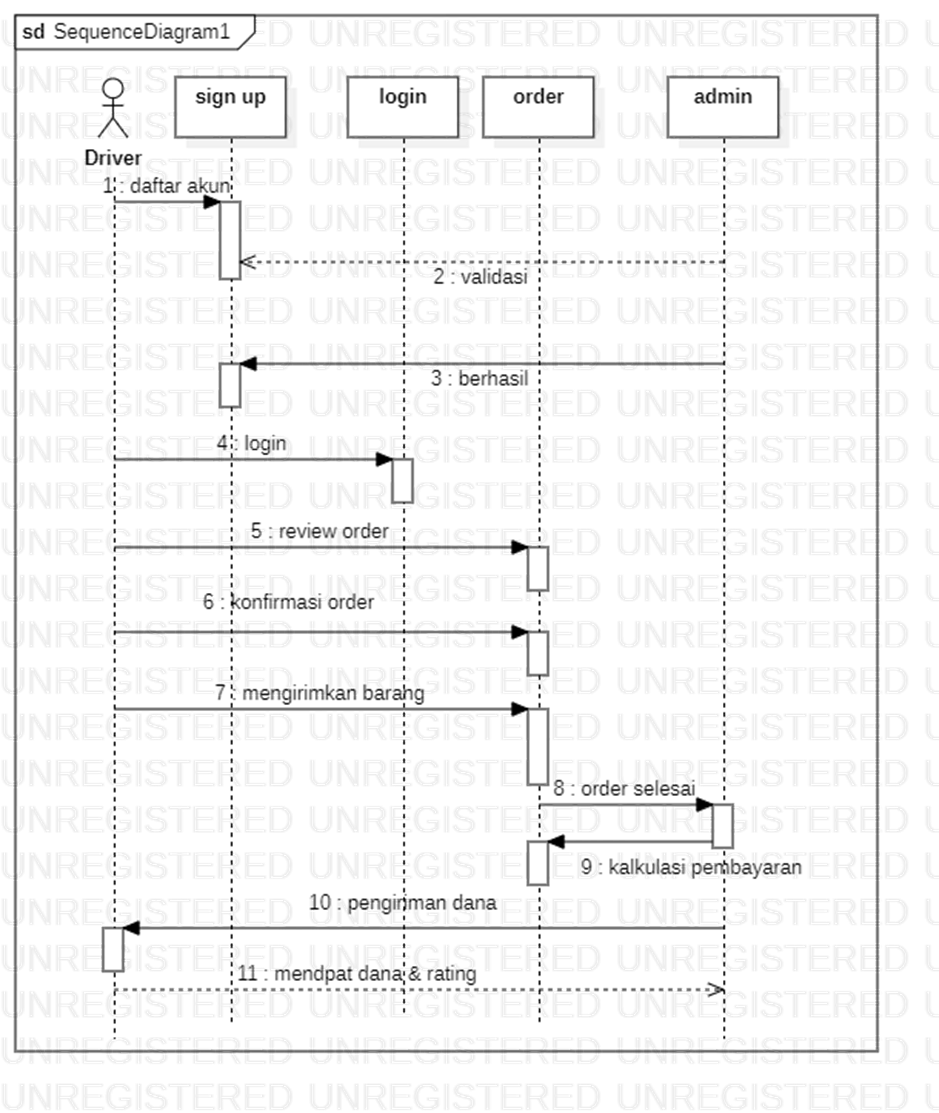
|  |
| --- |
| Driver |
| -Nama driver  -No hp  - Plat kendaraan  -Jenis kendaraan  -Rating Driver  -Riwayat Hasil orderan |

|  |
| --- |
| Orderan |
| -Alamat Pembeli  -Barang yang ingin dibeli  -Harga total barang dengan ongkos kirim  -Jarak, Durasi  -Pembayaran  -Rating mitra dan driver |



1. 



****