

**Nama : Sabrina Putri Malika**

**Nim : 12030123120034**

## **USE CASE DIAGRAM**

### **Pengertian Use Case Diagram**

Use case diagram adalah satu dari berbagai jenis diagram *Unified Modelling Language* (UML) yang menggambarkan hubungan interaksi antara aktor dan sistem. Yang dimaksud aktor disini dapat berupa manusia atau perangkat. Jadi, secara umum bisa didefinisikan bahwa pengertian use case diagram adalah teknik guna menunjukkan hubungan antara pengguna sistem dengan sistem itu sendiri. Hasilnya berupa skema sederhana untuk memudahkan user membaca dan memahami informasi yang diberikan.

### **Fungsi Use Case Diagram**

#### **1. Memperlihatkan proses aktivitas dalam sistem**

Fungsi pertama yaitu berperan memperkenalkan tahap awal setiap kegiatan proses dalam sebuah sistem yang sedang dikembangkan. Hal tersebut memudahkan pengembang dalam menentukan kebutuhan yang sesuai dengan software dan pengguna.

#### **2. Menggambarkan proses bisnis dalam sistem**

Kedua, fungsi use case diagram adalah berperan menggambarkan urutan proses bisnis dengan lebih jelas, mudah dimengerti dan transparan untuk mencegah terjadinya kesalahan pada sistem yang akan dikembangkan atau dibangun.

#### **3. Sebagai jembatan antara developer dan konsumen**







Fungsi use case diagram adalah menjadi jembatan penghubung antara developer dengan konsumen untuk mendeskripsikan kejelasan sebuah sistem yang akan dikembangkan.

### **Manfaat Use Case Diagram**

1. Sebagai kebutuhan verifikasi.
2. Menjadi gambaran antarmuka sebuah sistem karena setiap sistem yang dibangun harus memiliki interface.
3. Mengidentifikasi orang yang dapat berinteraksi dengan sistem dan apa yang bisa dilakukan oleh sistem.

4. Memberikan kepastian mengenai kebutuhan sistem.
5. Memudahkan proses komunikasi antara domain expert dan pengguna.

### Simbol Use Case Diagram

Simbol	Keterangan
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i>
	<i>Use case</i> : Abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan <i>use case</i>
	<i>Generalisasi</i> : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i>
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya
	Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi

### Komponen Use Case Diagram

Terdapat tiga komponen utama untuk membuat desain sketsa atau diagram. Komponen-komponen itu di antaranya:

#### 1. Sistem

Komponen ini menyatakan batasan dari sistem dengan aktor yang menggunakannya. Sistem di sini diberikan label-label yang sesuai. Namun, umumnya sistem tidaklah diberi gambar karena kita tidak terlalu memberikan makna pada sebuah desain diagram.

#### 2. Actor

Meskipun banyak yang menyangka bahwa aktor adalah sebuah diagram, tetapi nyatanya tidak. Aktor bukanlah bagian dari diagram. Aktor di sini mempunyai peran begitu penting, yaitu menciptakan *use case* jadi lebih mudah. Aktor menjelaskan siapa

yang berinteraksi dengan sistem. Selain itu aktor juga akan memberikan informasi sekaligus menerima informasi dari sistem. Hebatnya, kedua aktivitas tersebut bisa terjadi secara bersamaan, Sob. Namun, aktor sama sekali tidak mengontrol sistem, melainkan hanya memberi gambaran tentang hubungannya dengan sistem tersebut.

### **3. Use case**

Komponen selanjutnya adalah use case. Use case merupakan komponen gambaran fungsional dalam sebuah sistem. Dengan begitu, pengguna atau konsumen dapat mengetahui setiap fungsi yang dibangun dalam sistem tersebut.

## **Relasi dalam Use Case Diagram**

### **1. Association**

Association adalah teknik yang digunakan untuk mengidentifikasi interaksi yang dilakukan oleh aktor tertentu dengan use case tertentu. Ini digambarkan dengan garis penghubung antara aktor dengan use case.

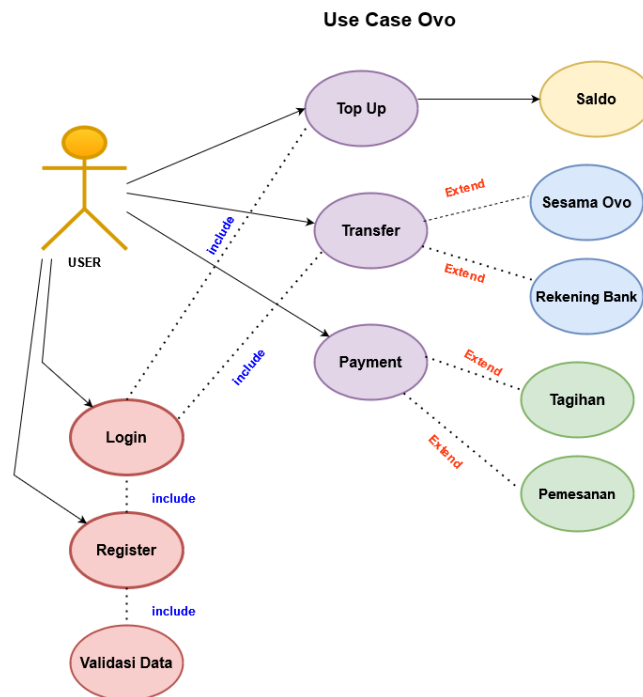
### **2. Generalisation**

Generalisasi adalah hubungan antara dua use case atau dua aktor. Dimana salah satunya meng-*inherit* dan menambahkan atau *override* sifat dari yang lainnya.

### **3. Dependency**

Dependency adalah relasi yang terbagi menjadi dua jenis, yaitu include dan exclude. Include berfungsi untuk mengidentifikasi hubungan atau relasi antara dua use case, dimana use case yang satu akan memanggil use case lainnya. Sementara exclude merupakan jenis yang jika dilakukan pemanggilan memerlukan suatu kondisi tertentu dan akan terjadi dependensi.

## Contoh dari Use Case Diagram

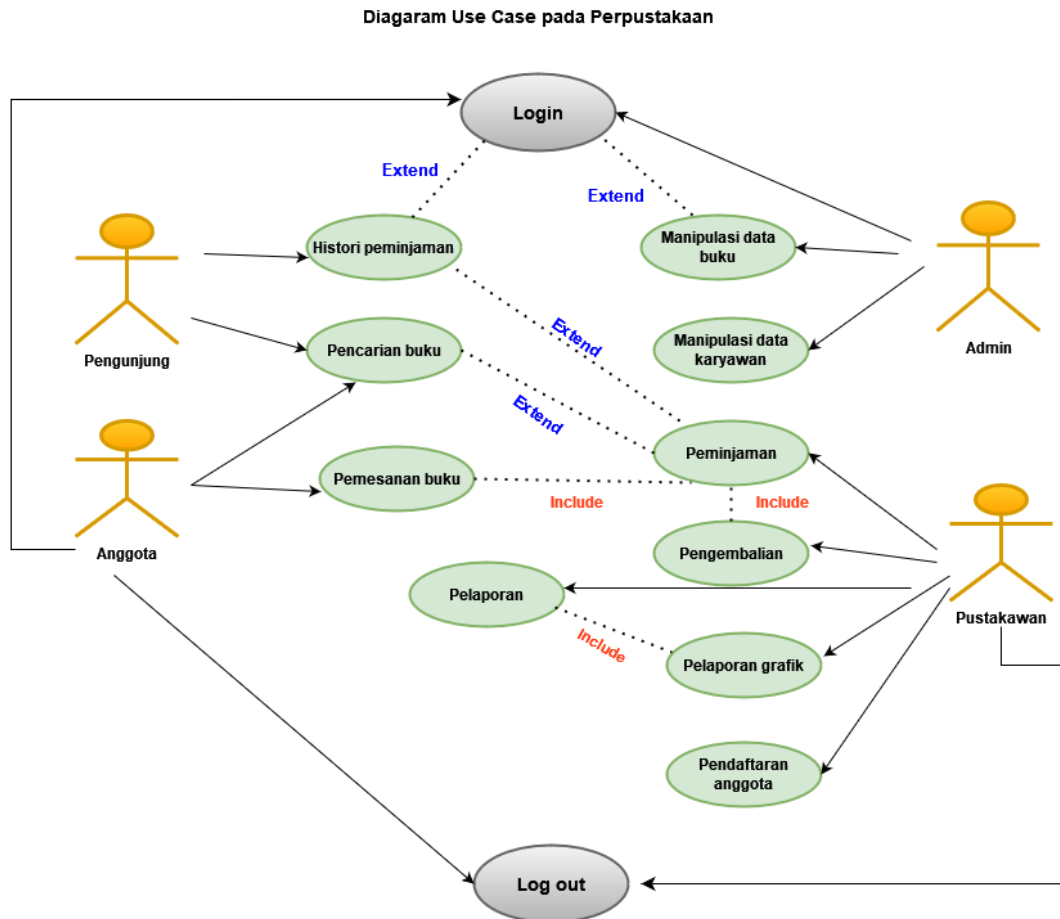


OVO adalah aplikasi pintar yang memberikan kemudahan dalam bertransaksi (OVO Cash), serta berfungsi untuk mengumpulkan poin di banyak tempat (OVO Points). Inilah contoh use case diagram studi kasus pada OVO.

Gambar di atas merupakan contoh use case diagram pada OVO, adapun penjelasannya sebagai berikut ini:

- *User*: Orang yang dapat mengakses atau menggunakan aplikasi OVO, mulai dari login ke aplikasi hingga melakukan aksi terhadap aplikasi seperti top up saldo, transfer, dan payment.
- *Register*: Register merupakan langkah pertama yang dilakukan user ketika ia tidak mempunyai akses pada aplikasi OVO. Mendaftarkan data diri ke dalam aplikasi agar dikenali.
- *Login*: Setelah mendapatkan akun, user harus melakukan login agar dapat mengakses berbagai fitur aplikasi OVO.
- *Top up*: Suatu kegiatan yang dilakukan user untuk mengisi ulang saldo OVO. Terdapat 2 pilihan alternatif untuk melakukan top up saldo, yaitu melalui ATM dan internet banking.

- *Transfer*: Transfer berfungsi untuk mengirim atau membagikan saldo dalam aplikasi OVO ke pengguna lain, baik sesama OVO atau ke rekening tertentu.
- *Payment*: Ketika user memilih menu payment, maka user dapat melakukan pembayaran lewat aplikasi.



Pada use case sistem perpustakaan di atas terdapat 4 aktor, di antaranya :

1. **Anggota**
2. **Pustakawan**
3. **Pengunjung**
4. **Dan Admin**

### Anggota

Apa saja yang bisa dilakukan anggota? Anggota dapat melakukan :

- Login

- Mencari buku
- Memesan Buku
- Dan Logout

Pemesanan buku di sini meliputi peminjaman yang sudah disetujui oleh pustakawan. Lalu ketika anggota melakukan login, maka dia dapat melihat histori peminjaman buku.

### **Pengunjung**

Aktor ini hanya bisa melakukan pencarian buku dan melihat histori peminjaman. Pengunjung baru bisa mengakses peminjaman buku jika telah disetujui oleh pustakawan.

### **Admin**

Aktor selanjutnya yaitu admin.

Beberapa aktifitas yang bisa dilakukan admin yaitu :

- Login
- Melihat histori peminjaman pengunjung
- Memanipulasi data buku
- Dan memanipulasi data karyawan

### **Pustakawan**

Terakhir aktor pustakawan.

Beberapa aktifitas yang bisa dilakukan oleh pustakawan yaitu :

- Mengakses peminjaman di mana peminjaman tersebut salah satunya yaitu pengembalian buku
- Membuat laporan
- Mengakses pendaftaran anggota