

Nama : Amanda Febriana Putri  
NIM : 12030123130155  
Kelas : C  
Mata Kuliah : Analisis dan Desain Sistem

### Kode Mermaid Untuk Membuat Diagram

#### Use case diagram

flowchart TD

%% Nodes

A("Admin")

B("Akuntan")

C("Staf")

D("Pencatatan Transaksi")

E("Melihat Buku Besar")

F("Menghasilkan Laporan Keuangan")

G("Mengelola Pembayaran dan Penerimaan")

H("Analisis Keuangan")

I("Pengelolaan Pengguna")

%% Edge connections between nodes

A -->|Input| D

A -->|Manage| I

B -->|View| E

B -->|Generate| F

B -->|Analyze| H

C -->|Record| G

%% Relationships

D -->|Updates| E

E -->|Feeds| F

F -->|Supports| H

%% Individual node styling

style A fill:#FFCC00, stroke:#333, stroke-width:2px

style B fill:#00CCFF, stroke:#333, stroke-width:2px

style C fill:#FF6699, stroke:#333, stroke-width:2px

style D fill:#FFFF99, stroke:#333, stroke-width:1px

style E fill:#FFFF99, stroke:#333, stroke-width:1px

style F fill:#FFFF99, stroke:#333, stroke-width:1px

style G fill:#FFFF99, stroke:#333, stroke-width:1px

style H fill:#FFFF99, stroke:#333, stroke-width:1px

style I fill:#FFFF99, stroke:#333, stroke-width:1px

### Diagram Activity

%%{ init: { "theme": "default" } }%%

flowchart TD

A("Mulai") --> B{"Pilih Aksi"}

B -->|Input Transaksi| C("Masukkan Detail Transaksi")

C --> D("Simpan Transaksi")

D --> E("Update Buku Besar")

B -->|Generate Laporan| F{"Pilih Jenis Laporan"}

F -->|Laba Rugi| G("Hasilkan Laporan Laba Rugi")

F -->|Neraca| H("Hasilkan Laporan Neraca")

F -->|Arus Kas| I("Hasilkan Laporan Arus Kas")

B -->|Kelola Pengguna| J("Update Hak Akses Pengguna")

E --> B

G --> B

H --> B

I --> B

J --> B

B --> |Selesai| K("Akhiri Proses")

### Diagram class

classDiagram

%% Classes

class User {

+String username

+String password

+String role

+login()

+logout()

}

class Transaction {

+int id

+String type

+double amount

+String date

+String description

+save()

}

class Ledger {

+double totalIncome

+double totalExpenses

+List<Transaction> transactions

+addTransaction(Transaction transaction)

+generateReport()

```
}
```

```
class Report {  
    +String reportType  
    +Date startDate  
    +Date endDate  
    +generate()  
}
```

%% Relationships

User "1" -- "0..\*" Transaction : creates  
Transaction "1" -- "1" Ledger : belongsTo  
Ledger "1" -- "0..\*" Report : generates

### **sequenceDiagram**

participant Pengguna  
participant Transaksi  
participant BukuBesar  
participant Laporan

Pengguna->>Transaksi: buatTransaksi()  
Transaksi->>BukuBesar: simpanTransaksi()  
BukuBesar-->>Transaksi: konfirmasiSimpan()

Pengguna->>BukuBesar: mintaLaporan()  
BukuBesar->>Laporan: buatLaporan()  
Laporan-->>BukuBesar: kirimLaporan()  
BukuBesar-->>Pengguna: berikanLaporan()

### **stateDiagram**

%% States

[\*] --> Menunggu

Menunggu --> Masuk : Pengguna memulai sistem

Masuk --> Terautentikasi : Kredensial valid

Masuk --> Kesalahan : Kredensial tidak valid

Terautentikasi --> MelihatDasbor : Akses dasbor

MelihatDasbor --> MelihatTransaksi : Pilih transaksi

MelihatDasbor --> MelihatLaporan : Pilih laporan

MelihatDasbor --> Keluar : Pengguna keluar

MelihatTransaksi --> DetailTransaksi : Lihat detail transaksi

MelihatTransaksi --> MelihatDasbor : Kembali ke dasbor

MelihatLaporan --> DetailLaporan : Lihat detail laporan

MelihatLaporan --> MelihatDasbor : Kembali ke dasbor

Keluar --> Menunggu : Pengguna keluar

Kesalahan --> Masuk : Coba masuk lagi