Name : Muhammad Amin Iqbaal Alam
Position : System Analyst

Sebagai seorang System Analyst, Anda jugs merupakan seorang bendahara RT ditempat Anda tinggal. Anda perlu membuat sebuah aplikasi yang bisa mempermudah Anda dalam melakukan tugas sebagai bendahara dan juga. Sebelum membuat aplikasi, tentu Anda perlu membuat design databasenya terlebih dahulu.

Kondisi terkait lingkungan RT tempat Anda antara lain:

1. Kavling Tanah:
    - Terdapat 90 kavling tanah, terdiri dari:
        i. Blok A1 – A20
        ii. Blok B1 – B20
        iii. Blok C1 – C20
        iv. Blok D1 – D20
        v. Blok E1 – E10
    - Ukuran kavling Blok E adalah 200m² dan sisanya 100m². Setiap kavling bisa jadi sudah berdiri bangunan dan berpenghuni, tidak berpenghuni ataupun tanah kosong / sedang dibangun.
2. Iuran RT:
    - Iuran dihitung berdasarkan luas kavling per 100m² dengan ketentuan:
        i. Dihuni:
            1. Iuran Kebersihan: Rp. 100.000,-
            2. Iuran Keamanan: Rp. 100.000,-
        ii. Tidak dihuni:
            1. Iuran Kebersihan: Rp. 100.000,-
            2. Iuran Keamanan: Rp. 50.000,-
        iii. Kavling kosong / sedang dibangun:
            1. Iuran Kebersihan: Rp. 50.000,-
            2. Iuran Keamanan: Rp. 50.000,-
    - Semua pemilik kavling di Blok E pasti memiliki 2 kavling di blok lainnya. 1 kavling tersebut berpenghuni (disewakan) dan 1 kavling kosong / sedang dibangun. 3.
3. Pemasukkan dan Pengeluaran:
    - Data tetap yang dikeluarkan bendahara setiap bulannya adalah Rp. 2.000.000 untuk kebersihan dan Rp. 8.000.000 untuk keamanan.
    - Dana yang bisa dikeluarkan dari kas Bendahara adalah:
        i. Sumbangan pernikahan: Rp. 500.000,-
        ii. Sumbangan duka: Rp. 1.000.000,-
        iii. Sumbangan sakit: Rp. 250.000,-
    - Bendahara juga menerima sumbangan dana dari donatur warga yang tidak ditentukan nilai untuk keperluan acara lomba 17, acara keagamaan atau acara lainnya.

Task 1 :

Buatkan ERD dan mockup aplikasi Bendahara RT tersebut dengan relasi antar tabelnya

Task 2 :

Buatkan query dan tampilan datanya untuk report berikut:

1. Laporan Pemasukkan dan Pengeluaran Per Bulan secara Summary
2. Laporan Pemasukan dan Pengeluaran Per Bulan secara Detail – dapat terlihat siapa saja yang sudah membayar, belum membayar, atau menunggak beberapa bulan.
3. Dalam kondisi normal (semua warga membayar / tidak ada yang menunggak), berapa total pemasukan dan pengeluaran dalam 1 bulan?

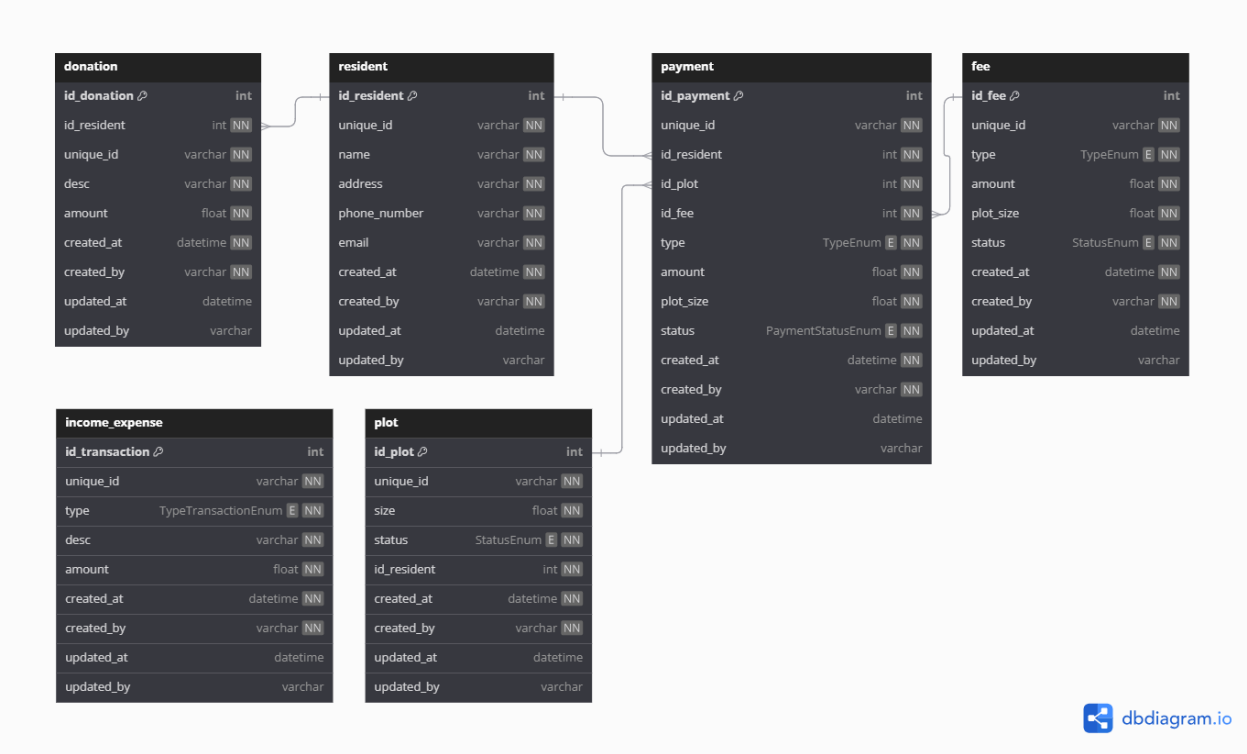
Task 3 :

Jika Anda dibantu oleh tetangga Anda yang juga seorang developer fullstack (frontend – backend). Buatkan sebuah file excel yang bisa me-mapping-kan kebutuhan Anda dengan halaman-halaman web yang perlu dibuatkan. Harapannya file tersebut ini bisa menjadikan acuan tetangga Anda dalam memahami kebutuhan Anda dan mengerjakan aplikasi Bendahara RT tersebut

Jawaban :

Task 1: ERD dan Mockup Aplikasi Bendahara RT

ERD (Entity-Relationship Diagram)



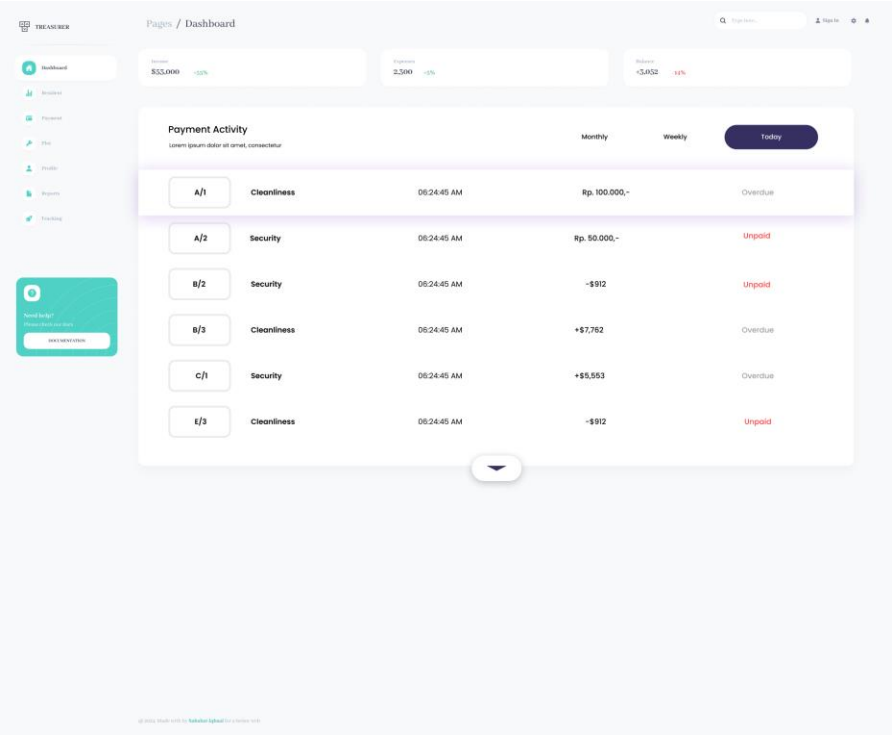
Query create database use PostgreSQL

```
CREATE TYPE "StatusEnum" AS ENUM (
  'Occupied',
  'Unoccupied',
  'Under Construction'
);

CREATE TYPE "TypeEnum" AS ENUM (
  'Cleanliness',
  'Security'
);

CREATE TYPE "PaymentStatusEnum" AS ENUM (
  'Paid',
  'Unpaid',
  'Overdue'
);

CREATE TYPE "TypeTransactionEnum" AS ENUM (
  'Income',
  'Expense'
);

CREATE TABLE "plot" (
  "id_plot" int PRIMARY KEY,
  "unique_id" varchar NOT NULL,
  "size" float NOT NULL,
  "status" StatusEnum NOT NULL,
  "id_resident" int NOT NULL,
  "created_at" datetime NOT NULL,
  "created_by" varchar NOT NULL,
  "updated_at" datetime,
  "updated_by" varchar
);

CREATE TABLE "resident" (
  "id_resident" int PRIMARY KEY,
  "unique_id" varchar NOT NULL,
  "name" varchar NOT NULL,
  "address" varchar NOT NULL,
  "phone_number" varchar NOT NULL,
  "email" varchar NOT NULL,
  "created_at" datetime NOT NULL,
  "created_by" varchar NOT NULL,
  "updated_at" datetime,
  "updated_by" varchar
);

CREATE TABLE "fee" (
  "id_fee" int PRIMARY KEY,
  "unique_id" varchar NOT NULL,
  "type" TypeEnum NOT NULL,
  "amount" float NOT NULL,
  "plot_size" float NOT NULL,
  "status" StatusEnum NOT NULL,
```

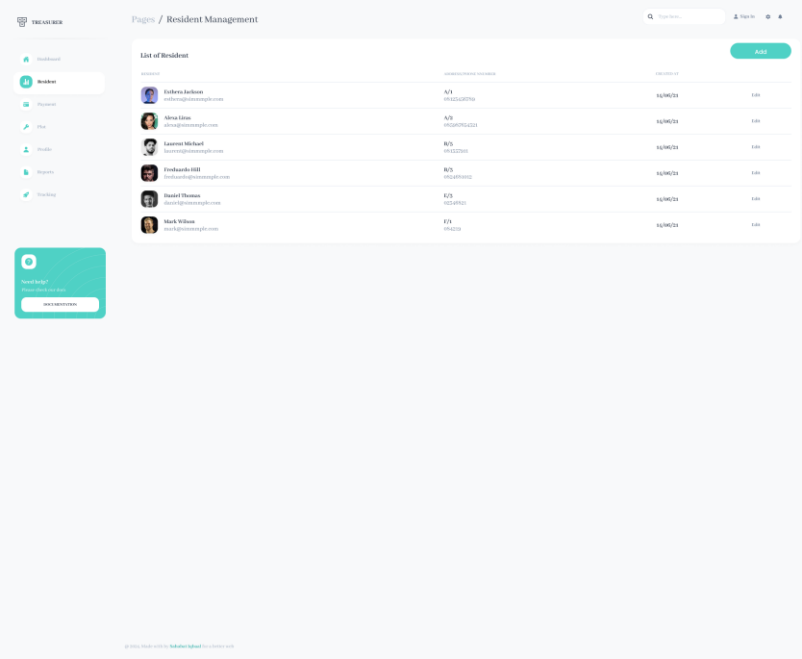
```
  "created_at" datetime NOT NULL,
  "created_by" varchar NOT NULL,
  "updated_at" datetime,
  "updated_by" varchar
);

CREATE TABLE "payment" (
  "id_payment" int PRIMARY KEY,
  "unique_id" varchar NOT NULL,
  "id_resident" int NOT NULL,
  "id_plot" int NOT NULL,
  "id_fee" int NOT NULL,
  "type" TypeEnum NOT NULL,
  "amount" float NOT NULL,
  "plot_size" float NOT NULL,
  "status" PaymentStatusEnum NOT NULL,
  "created_at" datetime NOT NULL,
  "created_by" varchar NOT NULL,
  "updated_at" datetime,
  "updated_by" varchar
);

CREATE TABLE "income_expense" (
  "id_transaction" int PRIMARY KEY,
  "unique_id" varchar NOT NULL,
  "type" TypeTransactionEnum NOT NULL,
  "desc" varchar NOT NULL,
  "amount" float NOT NULL,
  "created_at" datetime NOT NULL,
  "created_by" varchar NOT NULL,
  "updated_at" datetime,
  "updated_by" varchar
);

CREATE TABLE "donation" (
  "id_donation" int PRIMARY KEY,
  "id_resident" int NOT NULL,
  "unique_id" varchar NOT NULL,
  "desc" varchar NOT NULL,
  "amount" float NOT NULL,
  "created_at" datetime NOT NULL,
  "created_by" varchar NOT NULL,
  "updated_at" datetime,
  "updated_by" varchar
);

ALTER TABLE "payment" ADD FOREIGN KEY ("id_resident") REFERENCES "resident"
("id_resident") ON DELETE NO ACTION;

ALTER TABLE "payment" ADD FOREIGN KEY ("id_plot") REFERENCES "plot"
("id_plot") ON DELETE NO ACTION;

ALTER TABLE "payment" ADD FOREIGN KEY ("id_fee") REFERENCES "fee" ("id_fee")
ON DELETE NO ACTION;

ALTER TABLE "donation" ADD FOREIGN KEY ("id_resident") REFERENCES "resident"
("id_resident") ON DELETE NO ACTION;
```

Mockup Overview

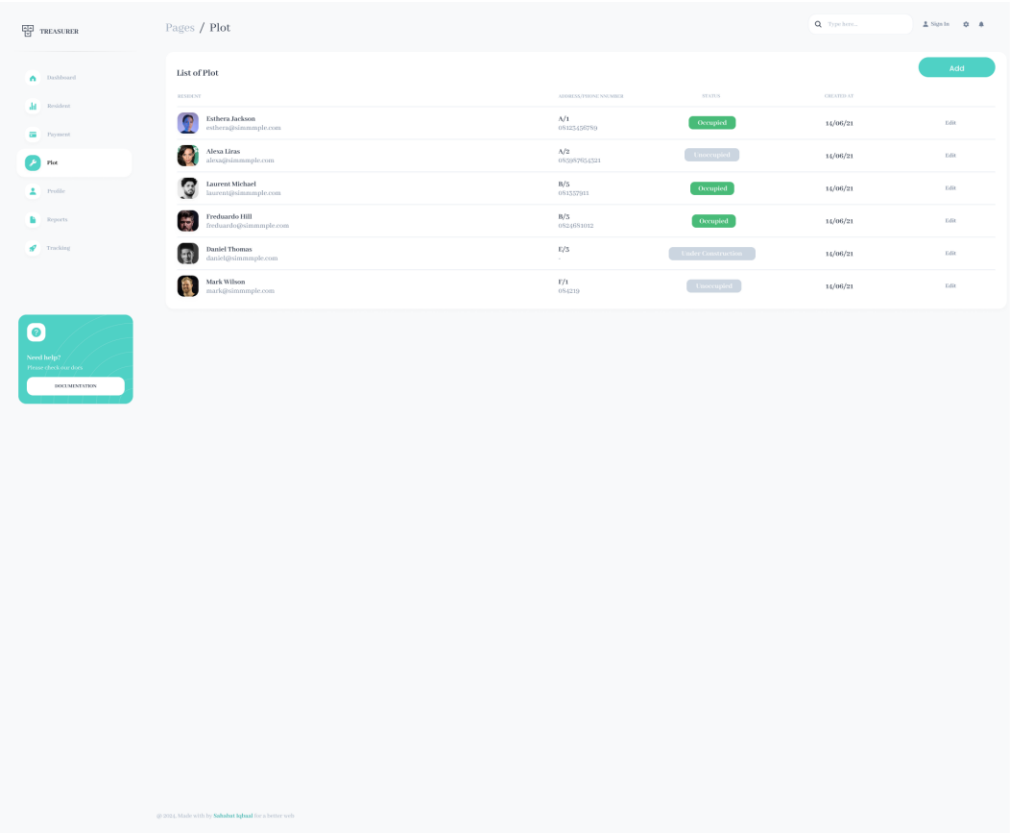
1. **Dashboard**
    - Summary of total income, total expenses, and balance.
    - Notification of unpaid or overdue payments.



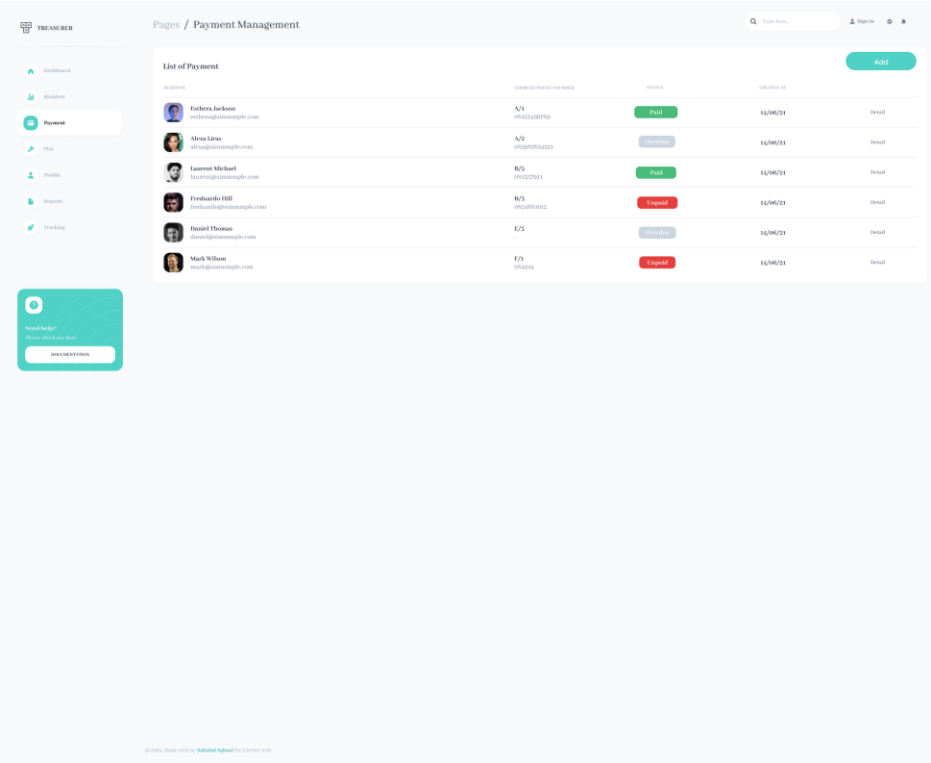
2. **Resident Management**
    - Add, edit, and view resident details (name, contact, etc.).



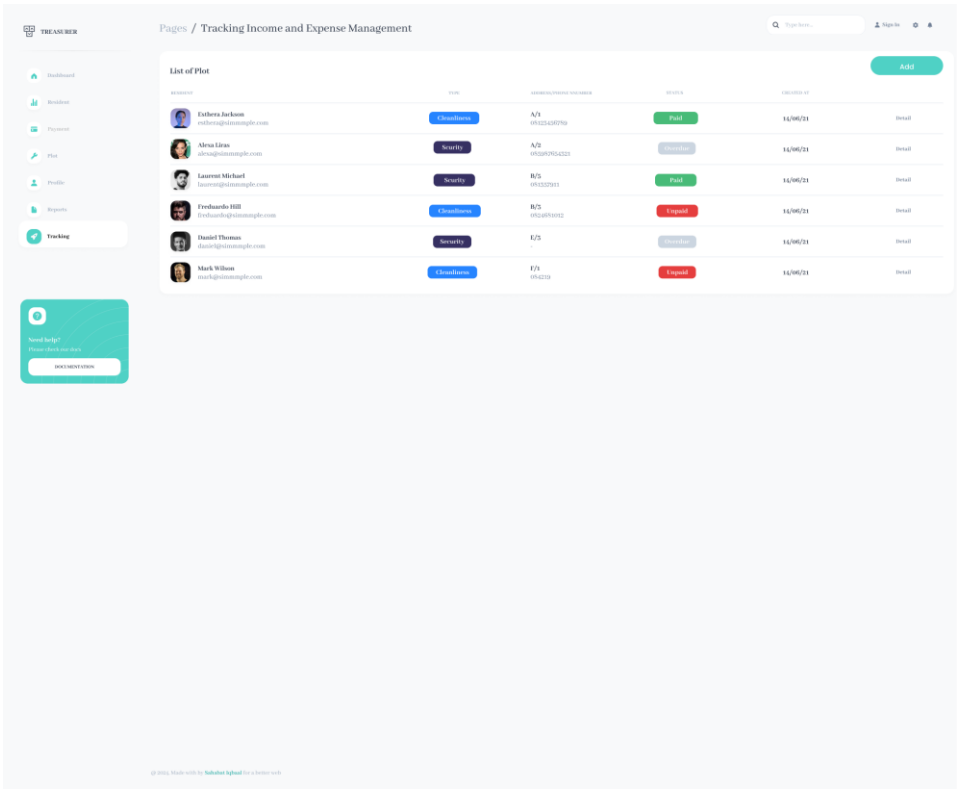
3. **Plot Management**
    - View, add, and edit plot details (block, number, size, status).
    - Assign residents to plots.



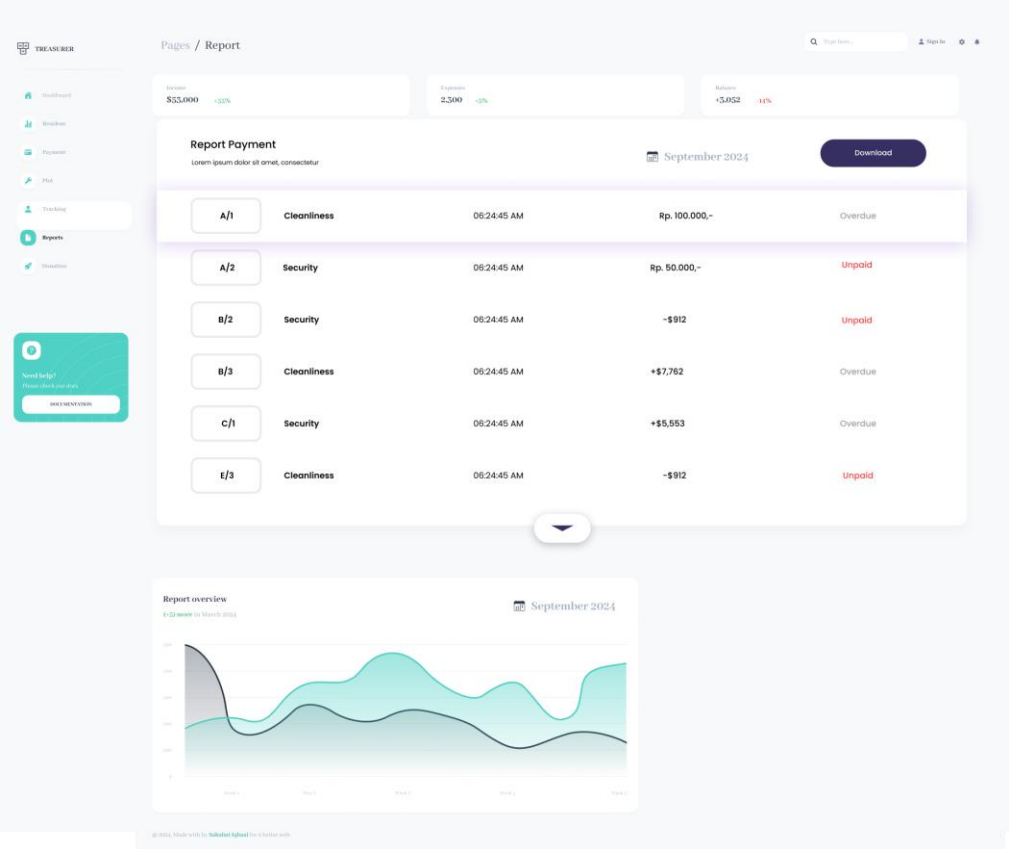
4. **Payment Management**
    - View payment status by plot and resident.
    - Record payments (paid, unpaid, overdue).



5. **Income and Expense Management**
    - Record monthly expenses (cleanliness, security).
    - View donation records.
    - Track donation for weddings, funerals, etc.



6. **Reports**
    - Monthly income and expense summary.
    - Detailed income and expense report (who has paid, who is overdue).



Task 2: Query dan Report Tampilan

1. Summary Report: Monthly Income and Expense

```
SELECT
    SUM(CASE WHEN Type = 'Income' THEN Amount ELSE 0 END) AS
Total_Income,
    SUM(CASE WHEN Type = 'Expense' THEN Amount ELSE 0 END) AS
Total_Expense,
    (SUM(CASE WHEN Type = 'Income' THEN Amount ELSE 0 END) - SUM(CASE
WHEN Type = 'Expense' THEN Amount ELSE 0 END)) AS Balance
FROM Income_Expense
WHERE MONTH(Date_Created) = '2024-09';
```

Output :

Bulan	Total Pemasukan	Total Pengeluaran	Sisa Saldo
Januari	Rp 9.000.000	Rp 10.000.000	-Rp 1.000.000
Februari	Rp 8.000.000	Rp 10.000.000	-Rp 2.000.000

2. Detailed Report: Income and Expense per Resident/Plot

```
SELECT
    R.Name AS Resident_Name,
    P.Block AS Block,
```

```
    P.Plot_Number,
    PM.Payment_Status,
    PM.Payment_Date,
    F.Type AS Fee_Type,
    F.Amount
FROM Payment PM
JOIN Resident R ON PM.Resident_ID = R.ID_Resident
JOIN Plot P ON PM.Plot_ID = P.ID_Plot
JOIN Fee F ON PM.Fee_ID = F.ID_Fee
WHERE MONTH(PM.Payment_Date) = '2024-09';
```

Output:

Blok	No	Bulan	Jumlah Iuran	Status Pembayaran
A	1	Januari	Rp 9.000.000	Paid
A	2	Februari	Rp 8.000.000	Unpaid

3. Normal Condition: Total Income and Expense per Month

```
SELECT
    SUM(F.Amount) AS Total_Income,
    (2000000 + 8000000) AS Total_Expense
FROM Fee F
WHERE F.Status = 'Occupied'
OR F.Status = 'Unoccupied'
OR F.Status = 'Under Construction'
AND MONTH(F.Date_Created) = '2024-09';
```

Output:

- Total Income: Rp 9.000.000
- Total Expense: Rp 10.000.000

Task 3: Excel Mapping for Web Pages

Feature	Page	Description
Dashboard	Dashboard Page	Summary of income, expenses, balance, notifications.
Resident Management	Resident List Page	View, add, edit resident details.
	Add/Edit Resident Page	Form for adding or editing resident details.
Plot Management	Plot List Page	View, add, edit plot details, assign residents.
	Add/Edit Plot Page	Form for adding or editing plot details.
Payment Management	Payment List Page	View payment status per plot/resident.
	Add Payment Page	Record a new payment.
Income/Expense Tracking	Income/Expense List Page	View, add, edit income and expenses.
	Add/Edit Income/Expense Page	Form for adding/editing income and expense details.

Donation Management	Donation List Page	View and record donations.
Reports	Monthly Summary Report Page	View monthly summary of income and expenses.
	Monthly Detailed Report Page	View detailed report of payments per resident/plot.