

**LAPORAN PROYEK KELAS
MATA KULIAH MANAJEMEN DATA TI2106 K03
PERANCANGAN SISTEM DAN PROTOTIPE APLIKASI SEDERHANA
PENGELOLA RESTORAN “RESTORIFY”**



Disusun Oleh :

Kelompok 11

Ahmad Hasyir Bastari (13423119)

Chindy Herpati (13423121)

Radithya Guntoro Adhi (13423125)

**INSTITUT TEKNOLOGI BANDUNG
FAKULTAS TEKNOLOGI INDUSTRI
TEKNIK INDUSTRI
2024**

DAFTAR ISI

BAB 1	3
1.1 Latar Belakang.....	3
1.2 Rumusan Masalah	3
1.3 Tujuan	3
BAB 2	4
2.1 Deskripsi Sistem	4
2.2 Software Requirements Specification (SRS)	6
2.3 Product Perspective	8
Use Case	8
Block Diagram Sistem.....	8
2.4 Product Function.....	9
Decomposition Diagram.....	9
2.5 Entity Relationship Diagram (ERD).....	9
2.6 SQL Database.....	10
2.3.1 SQL Query	15
BAB 3	17
3.1 Kesimpulan	17
3.2 Saran.....	17
3.3 Pembagian Tugas	17

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Dalam dunia usaha restoran, dibutuhkan pengelolaan yang efektif antara karyawan dan manajer untuk menjaga kelancaran operasional, efisiensi, serta kepuasan pelanggan. Seiring dengan berkembangnya zaman, teknologi banyak digunakan pada berbagai bidang, termasuk restoran yang kini mulai beralih ke penggunaan sistem aplikasi untuk membantu manajemen operasional. Sistem aplikasi ini dirancang untuk membantu mempermudah koordinasi antara manajer dengan karyawan dalam mengelola operasional dan administratif restoran, yang sangat dibutuhkan. Sistem aplikasi pengelola restoran yang efektif dapat membantu mengoptimalkan alur kerja, memastikan pengelolaan yang lebih transparan dan terstruktur, serta mempermudah manajer dalam memonitor dan mengarahkan kinerja karyawan, yang pada akhirnya akan meningkatkan produktivitas dan kepuasan pelanggan.

1.2 Rumusan Masalah

1. Bagaimana cara merancang sistem aplikasi pengelola restoran yang dapat mempermudah komunikasi dan koordinasi antara manajer dan karyawan?
2. Apa saja fitur yang dibutuhkan dalam sistem aplikasi untuk mendukung pengelolaan jadwal kerja, pembagian tugas, serta pemantauan kinerja karyawan?
3. Bagaimana aplikasi pengelola restoran dapat meningkatkan efisiensi operasional restoran, seperti pengelolaan pesanan dan bahan baku?
4. Bagaimana cara mengintegrasikan sistem aplikasi dengan berbagai aspek operasional restoran, seperti sistem pemesanan, pembayaran, dan laporan keuangan?

1.3 Tujuan

1. Meningkatkan komunikasi dan koordinasi antara manajer dan karyawan
2. Mempermudah pengelolaan jadwal kerja dan pembagian tugas
3. Meningkatkan efisiensi operasional dan bahan baku restoran
4. Menyediakan sistem pelaporan yang transparan dan akurat
5. Meningkatkan kualitas pelayanan kepada pelanggan
6. Meningkatkan kinerja dan produktivitas karyawan

BAB 2

ISI DAN PEMBAHASAN

2.1 Deskripsi Sistem

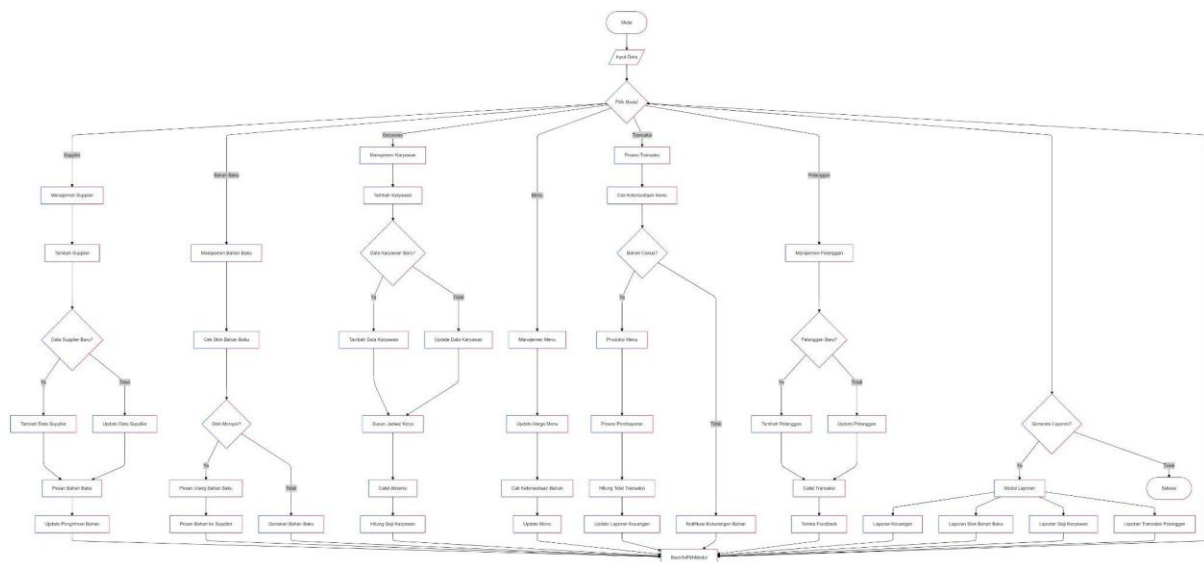
Restorify adalah aplikasi perangkat lunak yang dirancang untuk mengelola operasional restoran secara efisien. Sistem ini menyediakan alat bantu bagi manajer restoran untuk mengelola berbagai aspek operasional seperti pengelolaan bahan baku, manajemen karyawan, transaksi penjualan, hingga laporan keuangan dan hubungan dengan **supplier**. Dengan menggunakan sistem ini, restoran akan dapat meningkatkan efisiensi operasional, mengurangi kesalahan pencatatan, serta mendapatkan laporan yang lebih akurat untuk pengambilan keputusan.

Fitur Utama:

- **Manajemen Bahan Baku:** Memantau stok bahan baku, mencatat penggunaan, dan memberikan notifikasi saat bahan baku menipis, serta melakukan pemesanan ulang dari **supplier**.
- **Manajemen Supplier:** Mengelola data supplier, melakukan pemesanan bahan baku, serta melacak pengiriman dan pembayaran.
- **Manajemen Karyawan:** Menyusun jadwal kerja karyawan, mencatat absensi, serta menghitung gaji berdasarkan jam kerja.
- **Pengelolaan Menu:** Mengupdate daftar menu, harga menu, dan ketersediaan bahan baku untuk setiap item menu.
- **Pengelolaan Transaksi dan Keuangan:** Mencatat transaksi penjualan, menghitung pendapatan restoran, serta menghasilkan laporan keuangan yang berisi informasi pengeluaran dan pemasukan restoran.
- **Manajemen Pelanggan:** Menyimpan data pelanggan, riwayat transaksi, dan feedback pelanggan untuk meningkatkan kualitas pelayanan.

Dengan aplikasi **Restorify**, restoran dapat mengelola operasional secara lebih terstruktur dan efisien, serta memiliki kontrol yang lebih baik terhadap alur proses dari pengadaan bahan baku hingga pelayanan pelanggan.

2.1.1 Cara Kerja Sistem



Flowchart

Berikut adalah gambaran alur kerja sistem pengelolaan restoran **Restorify**:

1. Input Data:

- **Data Supplier:** Menyimpan informasi supplier, bahan baku yang disuplai, harga, dan waktu pengiriman.
- **Data Bahan Baku:** Memasukkan data bahan baku yang ada di restoran, termasuk stok dan penggunaan.
- **Data Karyawan:** Menyimpan data karyawan, jadwal kerja, dan absensi.
- **Data Menu:** Menyimpan dan mengupdate daftar menu, harga menu, serta bahan baku yang diperlukan.
- **Data Pelanggan:** Menyimpan data pelanggan dan riwayat transaksi mereka.
- **Data Transaksi:** Mencatat setiap transaksi penjualan yang terjadi di restoran.

2. Proses Pengelolaan:

- **Manajemen Supplier:** Mengelola informasi supplier, memesan bahan baku, serta melacak status pengiriman dan pembayaran.

- **Pengelolaan Bahan Baku:** Menghitung penggunaan bahan baku dan memberi notifikasi bila stok bahan baku mulai habis.
- **Pengelolaan Karyawan:** Mengelola absensi karyawan dan menghitung gaji berdasarkan jam kerja.
- **Pengelolaan Menu:** Mengupdate menu dan harga menu sesuai dengan ketersediaan bahan baku.
- **Pencatatan Transaksi:** Mencatat transaksi penjualan yang terjadi dan menghitung total pendapatan.

3. Output Data:

- **Laporan Bahan Baku:** Menampilkan status bahan baku yang tersedia, kebutuhan pengadaan bahan baku, serta laporan pengeluaran untuk pembelian bahan baku.
- **Laporan Menu:** Daftar menu yang tersedia beserta harga, serta ketersediaan bahan baku untuk setiap item menu.
- **Laporan Karyawan:** Status absensi karyawan dan perhitungan gaji yang harus dibayar.
- **Laporan Keuangan:** Laporan pendapatan dan pengeluaran restoran, termasuk transaksi penjualan dan pembayaran gaji karyawan.
- **Riwayat Transaksi Pelanggan:** Menyimpan riwayat transaksi pelanggan dan informasi terkait transaksi mereka.
- **Laporan Supplier:** Mengelola data pengadaan bahan baku dari supplier, serta status pengiriman dan pembayaran.

2.2 Software Requirements Specification (SRS)

Tujuan:

Tujuan dari aplikasi **Restorify** adalah untuk membantu restoran dalam mengelola operasional sehari-hari secara lebih efisien dan terorganisir, mulai dari pengelolaan bahan baku,

manajemen karyawan, transaksi penjualan, hingga pengelolaan laporan keuangan dan hubungan dengan **supplier**.

Scope:

- **Nama Aplikasi:** Restorify
- **Fungsi Aplikasi:**
 - **Manajemen Supplier:** Mengelola data supplier, memesan bahan baku, melacak pengiriman dan pembayaran.
 - **Manajemen Bahan Baku:** Memantau stok bahan baku dan memesan ulang saat stok menipis.
 - **Manajemen Karyawan:** Menyusun jadwal kerja karyawan, mencatat absensi, dan menghitung gaji.
 - **Pengelolaan Menu:** Mengupdate daftar menu dan harga sesuai dengan bahan baku yang tersedia.
 - **Pengelolaan Keuangan:** Mencatat transaksi penjualan dan menghitung pendapatan dan pengeluaran restoran.
 - **Manajemen Pelanggan:** Menyimpan riwayat transaksi pelanggan untuk meningkatkan pelayanan.

Manfaat:

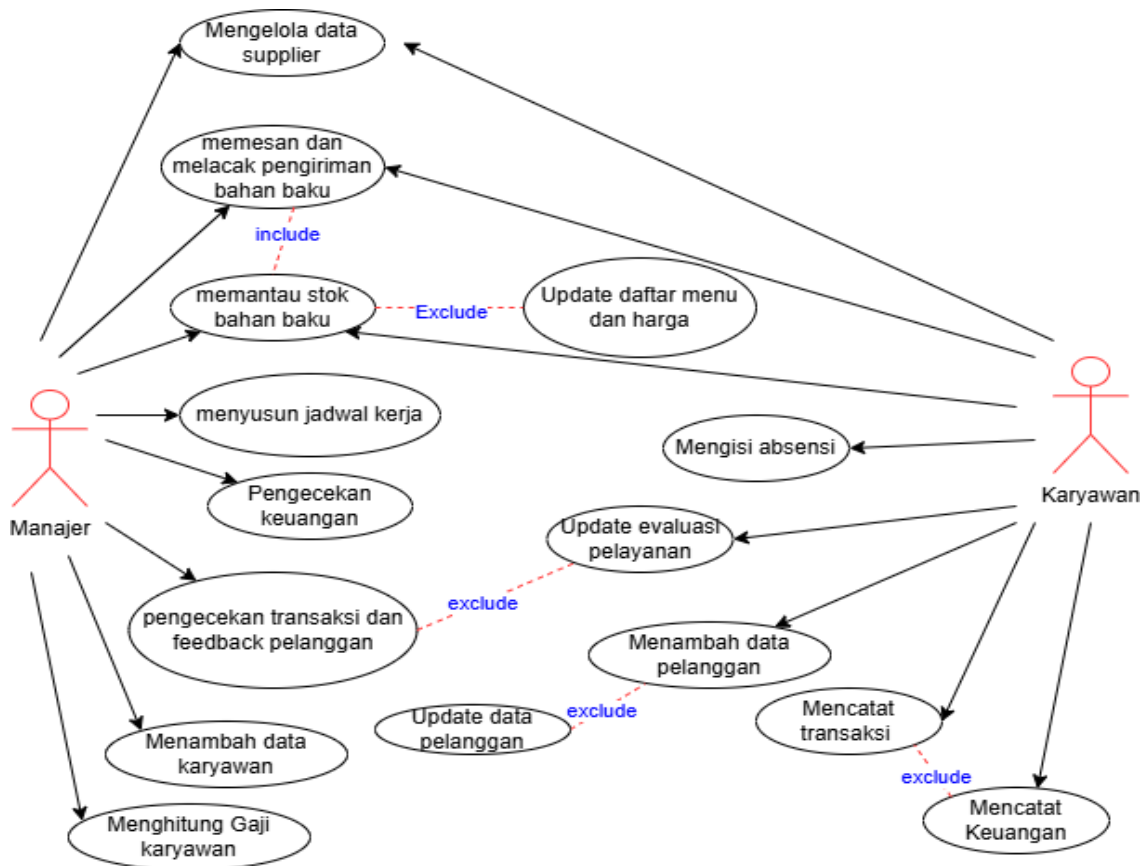
- **Efisiensi Operasional:** Pengelolaan bahan baku, supplier, karyawan, dan transaksi menjadi lebih efisien.
- **Pengelolaan Terpadu:** Semua data yang diperlukan restoran (bahan baku, menu, karyawan, transaksi) terintegrasi dalam satu sistem yang mudah diakses.
- **Laporan Real-Time:** Manajer restoran dapat dengan mudah memantau kinerja restoran dan membuat keputusan berdasarkan laporan yang dihasilkan sistem.

Sasaran:

Restoran kecil hingga menengah yang membutuhkan sistem untuk mengelola operasional dengan lebih efisien dan transparan.

2.3 Product Perspective

Use Case



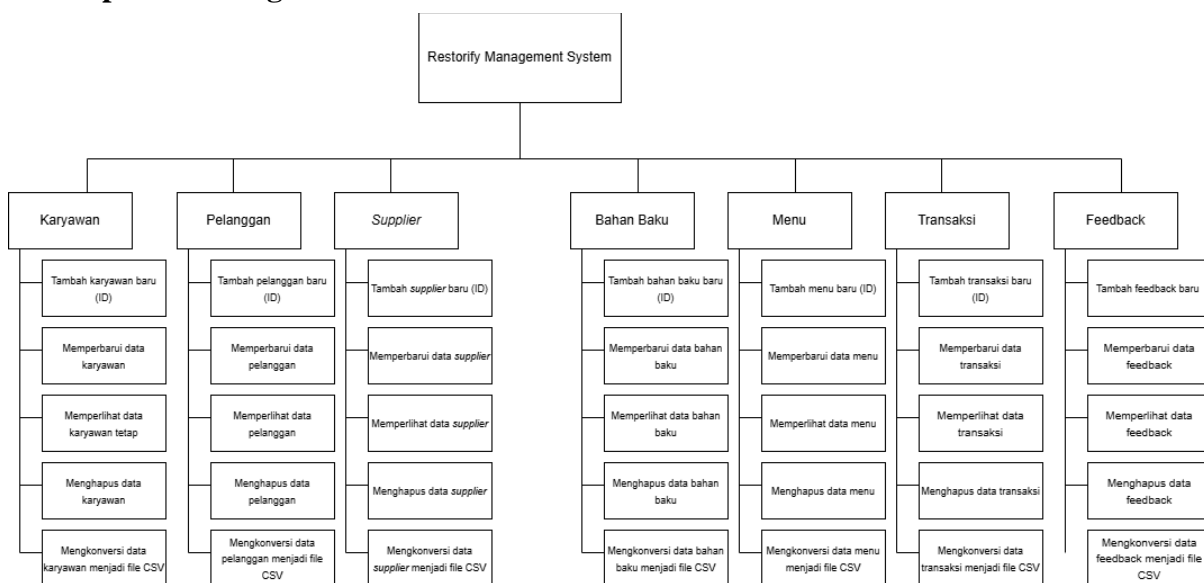
Block Diagram Sistem

Deskripsi Block Diagram: Diagram ini menggambarkan alur kerja sistem pengelolaan restoran, mulai dari input data yang terdiri dari **data supplier**, **data bahan baku**, **data karyawan**, **data menu**, **data pelanggan**, dan **data transaksi**. Proses aplikasi akan mengelola semua data tersebut dan menghasilkan berbagai output berupa laporan bahan baku, menu, transaksi, keuangan, serta laporan supplier dan pelanggan.



2.4 Product Function

Decomposition Diagram

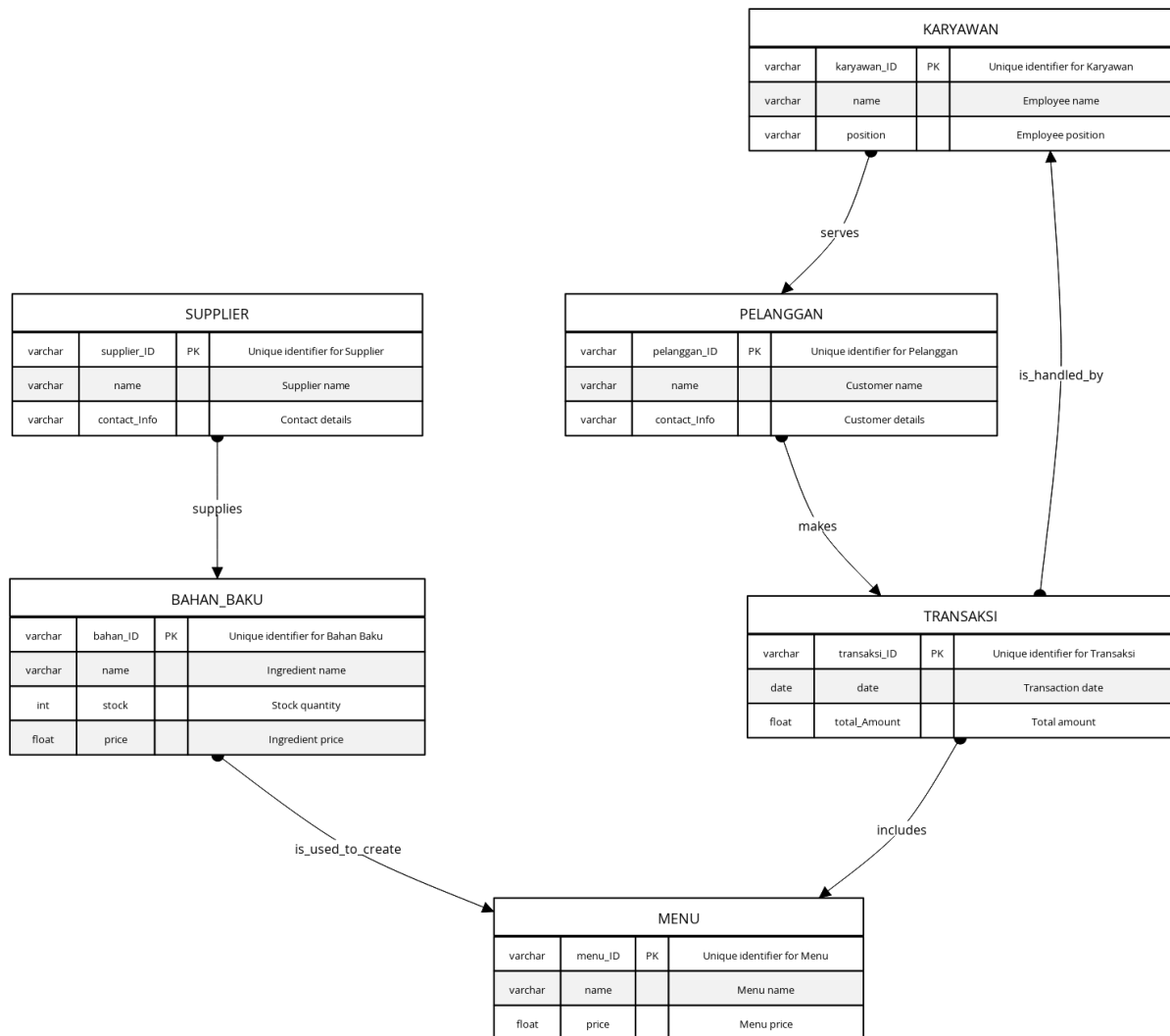


2.5 Entity Relationship Diagram (ERD)

Entitas dalam ERD:

- **Supplier:** Informasi tentang pemasok bahan baku (nama, kontak, bahan baku yang disuplai, harga, dll).
- **Bahan Baku:** Mengelola bahan baku yang ada di restoran, termasuk stok, penggunaan, dan pemasok.
- **Karyawan:** Data karyawan, jadwal kerja, absensi, dan gaji.

- **Menu:** Daftar menu, harga, dan bahan baku yang dibutuhkan.
- **Transaksi:** Mencatat setiap transaksi penjualan yang terjadi di restoran.
- **Pelanggan:** Menyimpan data pelanggan dan riwayat transaksi mereka.



2.6 SQL Database

```
CREATE DATABASE IF NOT EXISTS restorify;
USE restorify;

-- Tabel karyawan
CREATE TABLE IF NOT EXISTS karyawan (
    karyawan_id VARCHAR(5) PRIMARY KEY,
    employee_name VARCHAR(50) NOT NULL,
    position VARCHAR(25) NOT NULL
);
```

```

INSERT INTO karyawan (karyawan_id, employee_name, position) VALUES
    ('K001', 'Adi', 'Waiter'),
    ('K002', 'Agus', 'Waiter'),
    ('K003', 'Sinta', 'Cashier'),
    ('K004', 'Budi', 'Manajer'),
    ('K005', 'Bryan', 'Chef'),
    ('K006', 'Dian', 'Operational');

-- Tabel pelanggan
CREATE TABLE IF NOT EXISTS pelanggan (
    pelanggan_id VARCHAR(5) PRIMARY KEY,
    cus_name VARCHAR(50) NOT NULL,
    contact_info VARCHAR(15) NOT NULL
);

INSERT INTO pelanggan (pelanggan_id, cus_name, contact_info) VALUES
    ('P001', 'Dini', '0823416777'),
    ('P002', 'Anita', '0823452267'),
    ('P003', 'Daffa', '0890867736'),
    ('P004', 'Tiara', '083342156'),
    ('P005', 'Torik', '0876544325'),
    ('P006', 'Aziz', '087665489'),
    ('P007', 'Ayu', '087655394'),
    ('P008', 'Dedi', '0823416657');

-- Tabel supplier
CREATE TABLE IF NOT EXISTS supplier (
    supplier_id VARCHAR(5) PRIMARY KEY,
    supplier_name VARCHAR(50) NOT NULL,
    address VARCHAR(100) NOT NULL
);

INSERT INTO supplier (supplier_id, supplier_name, address) VALUES
    ('S001', 'Indofood', 'Jakarta'),
    ('S002', 'Ayla Farm', 'Bandung'),
    ('S003', 'FoodFresh', 'Tangerang');

-- Tabel bahan_baku
CREATE TABLE IF NOT EXISTS bahan_baku (
    bahan_id VARCHAR(5) PRIMARY KEY,
    nama_bahan VARCHAR(50) NOT NULL,
    stock INT NOT NULL CHECK (stock >= 0),
    satuan VARCHAR(20) NOT NULL,
    harga_bahan DECIMAL(15,2) NOT NULL,
    supplier_id VARCHAR(5),
    FOREIGN KEY (supplier_id) REFERENCES supplier(supplier_id)
);

INSERT INTO bahan_baku (bahan_id, nama_bahan, stock, satuan, harga_bahan, supplier_id) VALUES
    ('BB1', 'Ayam Potong', 50, 'Kg', 500000.00, 'S001'),
    ('BB2', 'Sayuran', 30, 'Kg', 200000.00, 'S002'),
    ('BB3', 'Ikan Lele', 20, 'Kg', 350000.00, 'S003'),
    ('BB4', 'Bawang', 50, 'Kg', 250000.00, 'S001'),
    ('BB5', 'Bumbu', 100, 'Bungkus', 600000.00, 'S001');

-- Tabel menu
CREATE TABLE IF NOT EXISTS menu (
    menu_id VARCHAR(5) PRIMARY KEY,
    nama_menu VARCHAR(50) NOT NULL,
    harga DECIMAL(15,2) NOT NULL CHECK (harga >= 0)
);

```

```

);

INSERT INTO menu (menu_id, nama_menu, harga) VALUES
    ('MN1', 'Ayam Bakar', 20000.00),
    ('MN2', 'Nasi Goreng', 15000.00),
    ('MN3', 'Oseng Kangkung', 10000.00),
    ('MN4', 'Tumis Labu', 12000.00),
    ('MN5', 'Es Krim', 10000.00),
    ('MN6', 'Es Jeruk', 6000.00),
    ('MN7', 'Es Teh Manis', 5000.00);

-- Tabel transaksi
CREATE TABLE IF NOT EXISTS transaksi (
    transaksi_id VARCHAR(5) PRIMARY KEY,
    tanggal_pembelian DATE NOT NULL,
    pelanggan_id VARCHAR(5) NOT NULL,
    karyawan_id VARCHAR(5) NOT NULL,
    total_transaksi DECIMAL(15,2) NOT NULL CHECK (total_transaksi >= 0),
    FOREIGN KEY (pelanggan_id) REFERENCES pelanggan(pelanggan_id),
    FOREIGN KEY (karyawan_id) REFERENCES karyawan(karyawan_id)
);

INSERT INTO transaksi (transaksi_id, tanggal_pembelian, pelanggan_id,
    karyawan_id, total_transaksi) VALUES
    ('T001', '2024-12-01', 'P001', 'K003', 170000.00),
    ('T002', '2024-12-03', 'P002', 'K003', 250000.00),
    ('T003', '2024-12-05', 'P003', 'K003', 150000.00),
    ('T004', '2024-12-04', 'P004', 'K003', 200000.00),
    ('T005', '2024-12-06', 'P005', 'K003', 280000.00),
    ('T006', '2024-12-07', 'P006', 'K003', 500000.00);

-- Tabel detail_transaksi
CREATE TABLE IF NOT EXISTS detail_transaksi (
    detail_id INT PRIMARY KEY AUTO_INCREMENT,
    transaksi_id VARCHAR(5) NOT NULL,
    menu_id VARCHAR(5) NOT NULL,
    jumlah INT NOT NULL CHECK (jumlah > 0),
    harga DECIMAL(15,2) NOT NULL CHECK (harga >= 0),
    FOREIGN KEY (transaksi_id) REFERENCES transaksi(transaksi_id),
    FOREIGN KEY (menu_id) REFERENCES menu(menu_id)
);

INSERT INTO detail_transaksi (transaksi_id, menu_id, jumlah, harga)
VALUES
    ('T001', 'MN1', 2, 20000.00),
    ('T001', 'MN6', 2, 6000.00),
    ('T002', 'MN2', 2, 15000.00),
    ('T002', 'MN6', 3, 6000.00),
    ('T003', 'MN3', 1, 10000.00),
    ('T003', 'MN7', 2, 5000.00),
    ('T004', 'MN4', 2, 12000.00),
    ('T004', 'MN7', 1, 5000.00),
    ('T005', 'MN1', 3, 20000.00),
    ('T005', 'MN5', 1, 10000.00),
    ('T006', 'MN2', 4, 15000.00),
    ('T006', 'MN6', 2, 6000.00);

-- Tabel komposisi_menu
CREATE TABLE IF NOT EXISTS komposisi_menu (
    menu_id VARCHAR(5) NOT NULL,
    bahan_id VARCHAR(5) NOT NULL,

```

```

        jumlah_bahan INT NOT NULL CHECK (jumlah_bahan > 0),
        PRIMARY KEY (menu_id, bahan_id),
        FOREIGN KEY (menu_id) REFERENCES menu(menu_id),
        FOREIGN KEY (bahan_id) REFERENCES bahan_baku(bahan_id)
    );

INSERT INTO komposisi_menu (menu_id, bahan_id, jumlah_bahan) VALUES
    ('MN1', 'BB1', 1),
    ('MN1', 'BB5', 1),
    ('MN2', 'BB2', 1),
    ('MN2', 'BB5', 1),
    ('MN3', 'BB2', 1),
    ('MN3', 'BB5', 1),
    ('MN4', 'BB2', 1),
    ('MN4', 'BB5', 1),
    ('MN5', 'BB5', 1),
    ('MN6', 'BB5', 1),
    ('MN7', 'BB4', 1);

-- Tabel jadwal_kerja
CREATE TABLE IF NOT EXISTS jadwal_kerja (
    jadwal_id INT PRIMARY KEY AUTO_INCREMENT,
    karyawan_id VARCHAR(5) NOT NULL,
    tanggal DATE NOT NULL,
    shift VARCHAR(10) NOT NULL,
    FOREIGN KEY (karyawan_id) REFERENCES karyawan(karyawan_id)
);

INSERT INTO jadwal_kerja (karyawan_id, tanggal, shift) VALUES
    ('K001', '2024-12-01', 'Pagi'),
    ('K002', '2024-12-01', 'Siang'),
    ('K003', '2024-12-01', 'Malam'),
    ('K001', '2024-12-02', 'Siang'),
    ('K002', '2024-12-02', 'Pagi'),
    ('K003', '2024-12-02', 'Malam');

-- Tabel absensi
CREATE TABLE IF NOT EXISTS absensi (
    absensi_id INT PRIMARY KEY AUTO_INCREMENT,
    karyawan_id VARCHAR(5) NOT NULL,
    tanggal DATE NOT NULL,
    status VARCHAR(10) NOT NULL,
    FOREIGN KEY (karyawan_id) REFERENCES karyawan(karyawan_id)
);

INSERT INTO absensi (karyawan_id, tanggal, status) VALUES
    ('K001', '2024-12-01', 'Hadir'),
    ('K002', '2024-12-01', 'Hadir'),
    ('K003', '2024-12-01', 'Hadir'),
    ('K001', '2024-12-02', 'Sakit'),
    ('K002', '2024-12-02', 'Hadir'),
    ('K003', '2024-12-02', 'Hadir');

-- Tabel penggajian
CREATE TABLE IF NOT EXISTS penggajian (
    penggajian_id INT PRIMARY KEY AUTO_INCREMENT,
    karyawan_id VARCHAR(5) NOT NULL,
    bulan INT NOT NULL CHECK (bulan BETWEEN 1 AND 12),
    tahun INT NOT NULL,
    jumlah_gaji DECIMAL(15,2) NOT NULL CHECK (jumlah_gaji >= 0),
    FOREIGN KEY (karyawan_id) REFERENCES karyawan(karyawan_id)
);

```

```

);

INSERT INTO penggajian (karyawan_id, bulan, tahun, jumlah_gaji) VALUES
    ('K001', 12, 2024, 3000000.00),
    ('K002', 12, 2024, 2800000.00),
    ('K003', 12, 2024, 3200000.00);

-- Tabel pemesanan_bahan
CREATE TABLE IF NOT EXISTS pemesanan_bahan (
    pemesanan_id VARCHAR(5) PRIMARY KEY,
    supplier_id VARCHAR(5) NOT NULL,
    tanggal_pemesanan DATE NOT NULL,
    status VARCHAR(20) NOT NULL,
    FOREIGN KEY (supplier_id) REFERENCES supplier(supplier_id)
);

INSERT INTO pemesanan_bahan (pemesanan_id, supplier_id,
    tanggal_pemesanan, status) VALUES
    ('PB001', 'S001', '2024-12-05', 'Dipesan'),
    ('PB002', 'S002', '2024-12-05', 'Dikirim');

-- Tabel detail_pemesanan_bahan
CREATE TABLE IF NOT EXISTS detail_pemesanan_bahan (
    detail_pemesanan_id INT PRIMARY KEY AUTO_INCREMENT,
    pemesanan_id VARCHAR(5) NOT NULL,
    bahan_id VARCHAR(5) NOT NULL,
    jumlah INT NOT NULL CHECK (jumlah > 0),
    harga_satuan DECIMAL(15,2) NOT NULL CHECK (harga_satuan >= 0),
    FOREIGN KEY (pemesanan_id) REFERENCES pemesanan_bahan(pemesanan_id),
    FOREIGN KEY (bahan_id) REFERENCES bahan_baku(bahan_id)
);

INSERT INTO detail_pemesanan_bahan (pemesanan_id, bahan_id, jumlah,
    harga_satuan) VALUES
    ('PB001', 'BB1', 20, 500000.00),
    ('PB001', 'BB4', 30, 250000.00),
    ('PB002', 'BB2', 50, 200000.00);

-- Tabel feedback pelanggan
CREATE TABLE IF NOT EXISTS feedback (
    feedback_id INT PRIMARY KEY AUTO_INCREMENT,
    pelanggan_id VARCHAR(5) NOT NULL,
    karyawan_id VARCHAR(5) NOT NULL,
    tanggal DATE NOT NULL,
    rating INT NOT NULL CHECK (rating BETWEEN 1 AND 5),
    komentar TEXT,
    FOREIGN KEY (pelanggan_id) REFERENCES pelanggan(pelanggan_id),
    FOREIGN KEY (karyawan_id) REFERENCES karyawan(karyawan_id)
);

INSERT INTO feedback (pelanggan_id, karyawan_id, tanggal, rating,
    komentar) VALUES
    ('P001', 'K003', '2024-12-02', 5, 'Pelayanan sangat memuaskan.'),
    ('P002', 'K001', '2024-12-03', 4, 'Makanan enak dan tempat nyaman.'),
    ('P003', 'K002', '2024-12-04', 3, NULL);

```

2.3.1 SQL Query

1. Menampilkan daftar karyawan berdasarkan posisi

```
SELECT * FROM karyawan WHERE position = 'waiter';
```

karyawan_Id	employee_name	position
K001	Adi	waiter
K002	Agus	waiter

2. Menampilkan daftar menu dengan harga di bawah Rp15.000

```
SELECT * FROM menu WHERE harga < 15000;
```

menu_Id	nama_menu	harga
MN1	ayam bakar	20
MN2	nasi goreng	15
MN3	oseng kangkung	10
MN4	tumis labu	12
MN5	eskrim	10
MN6	es jeruk	6
MN7	esteh manis	5

3. Menampilkan transaksi dengan total di atas Rp200.000

```
SELECT * FROM transaksi WHERE total_transaksi > 200000;
```

transaksi_Id	tanggal_pembelian	total_transaksi
T002	2024-12-3	250000
T005	2024-12-6	280000
T006	2024-12-7	500000

4. Menampilkan total pengeluaran berdasarkan pembelian bahan baku

```
SELECT
    nama_bahan AS Nama_Bahan,
    harga_bahan AS Harga_Per_Unit,
    stock AS Jumlah_Stok,
    (CAST(REPLACE(stock, ' bungkus', '') AS INT) * harga_bahan) AS
Total_Pengeluaran
FROM
    bahan_baku;
```

Nama_Bahan	Harga_Per_Unit	Jumlah_Stok	Total_Pengeluaran
Ayam potong	500000	50	25000000
Sayuran	200000	30	6000000
Ikan Lele	350000	20	7000000
Bawang	250000	50	12500000
Bumbu	600000	100 bungkus	60000000

5. Menampilkan jumlah pendapatan per hari berdasarkan tanggal pembelian

```
SELECT
    tanggal_pembelian AS Tanggal,
    SUM(total_transaksi) AS Total_Pendapatan
FROM
    transaksi
GROUP BY
    tanggal_pembelian
ORDER BY
    Tanggal ASC;
```

```
SELECT
    tanggal_pembelian AS Tanggal,
    SUM(total_transaksi) AS Total_Pendapatan
FROM
    transaksi
GROUP BY
    tanggal_pembelian
ORDER BY
    Tanggal ASC;
```

Tanggal	Total_Pendapatan
2024-12-1	170000
2024-12-3	250000
2024-12-4	200000
2024-12-5	150000
2024-12-6	280000
2024-12-7	500000

BAB 3

KESIMPULAN DAN SARAN

3.1 Kesimpulan

Aplikasi pengelola restoran ini dibuat menggunakan kode program python yang diintegrasikan dengan database restoran. Restorify merupakan aplikasi yang dapat mengelola seluruh aspek operasional restoran dengan lebih efisien, mulai dari pengelolaan bahan baku yang terintegrasi dengan *supplier*, manajemen karyawan, hingga pencatatan transaksi dan laporan keuangan. Dengan sistem yang terintegrasi ini, restoran akan memiliki kontrol yang lebih baik dalam pengelolaan bahan baku, karyawan, dan transaksi, serta mendapatkan laporan yang lebih akurat untuk mendukung pengambilan keputusan yang lebih baik.

3.2 Saran

Aplikasi ini masih dapat dikembangkan seperti menambahkan fitur untuk menyambungkan ke aplikasi pesanan *online* seperti, Grab, Gojek, Shopee agar restoran dapat mengekspansi pasar nya melalui penjualan *online*. Selain itu aplikasi ini masih memiliki kekurangan dalam sistem nya yaitu tidak adanya fitur untuk mengatur fasilitas restoran seperti meja, kursi, alat masak dan peralatan operasional lainnya. Sehingga aplikasi ini dapat dikembangkan lebih baik lagi dengan menambahkan fitur manajemen fasilitas operasional restoran.

3.3 Pembagian Tugas

1. Ahmad Hasyir Bastari : Pembuatan ERD, PPT konversi CSV, dekomposisi diagram
2. Chindy Herpati : Pembuatan laporan, PPT, use case, SQL database & Query, block diagram
3. Radithya Guntoro Adhi : Pembuatan kode program python, flowchart, SRS