

ERP System:

Project Overview

This project implements a **ERP (Enterprise Resource Planning) System** designed to manage and automate core business functions such as inventory tracking, employee management, supplier coordination, sales processing, purchase monitoring, leave handling, and customer relations. Built using **Laravel for the backend** and **React.js for the frontend**, the system provides a modular, API-driven architecture with real-time updates and clean UI for seamless user interaction.

The ERP includes the following key modules:

- **Inventory Management:** Categories, Products, Suppliers, Purchases
- **HR Management:** Departments, Employees, Attendances, Leaves
- **Sales Management:** Customers, Sales Records
- **Finance Overview:** Expense tracking and a consolidated dashboard for metrics

It is tailored for small to medium-sized organizations to efficiently handle their day-to-day administrative and operational tasks.

Project Members:

Fiza Nihar

Afia Shah

Hafsa Javed

Conceptual schema :

Finalized Conceptual Schema (Entity-Level Description):

1. Product

Represents items available for sale or purchase in the system.

- **product_id** (*Primary Key*): Unique identifier for each product.
 - **name**: Name of the product.
 - **category_id** (*Foreign Key* → *Categories.category_id*): Links each product to its category.
 - **price**: Cost per unit of the product.
 - **quantity**: Current inventory stock level.
-

2. Category

Represents classifications for products.

- **category_id** (*Primary Key*): Unique identifier for a category.
 - **name**: Unique category name (e.g., Electronics, Stationery).
-

3. Supplier

Vendors providing products.

- **supplier_id** (*Primary Key*): Unique identifier for each supplier.
 - **name**: Supplier name.
 - **contact**: Phone or email contact.
 - **address**: Physical address of the supplier.
-

4. Purchase

Records procurement of products.

- **purchase_id** (*Primary Key*): Unique identifier for a purchase transaction.
 - **product_id** (*Foreign Key* → *Products.product_id*): Product being purchased.
 - **supplier_id** (*Foreign Key* → *Suppliers.supplier_id*): Supplier from whom the product was purchased.
 - **quantity**: Number of units purchased.
 - **date**: Date of the purchase transaction.
-

5. Employee

Staff members of the organization.

- **employee_id** (*Primary Key*): Unique identifier for each employee.
 - **name**: Full name.
 - **department_id** (*Foreign Key* → *Departments.department_id*): Which department they belong to.
 - **salary**: Monthly salary.
 - **contact**: Phone or email contact.
-

6. Department

Units within the company.

- **department_id** (*Primary Key*): Unique identifier for a department.
 - **name**: Department name (e.g., Sales, IT).
-

7. Attendance

Tracks employee presence.

- **attendance_id** (*Primary Key*): Unique identifier for an attendance record.
 - **employee_id** (*Foreign Key* → *Employees.employee_id*): Employee present or absent.
 - **date**: Date of the record.
 - **status** (*ENUM: Present, Absent*): Attendance status.
-

8. Customer

Clients who purchase products.

- **customer_id** (*Primary Key*): Unique ID for each customer.
 - **name**: Full name of the customer.
 - **email**: Email address.
 - **phone**: Contact number.
 - **address**: Physical location.
-

9. Sale

Tracks customer purchases.

- **sale_id** (*Primary Key*): Unique sale transaction ID.

- **customer_id** (*Foreign Key* → *Customers.customer_id*): The buyer.
 - **product_id** (*Foreign Key* → *Products.product_id*): Product sold.
 - **quantity**: Number of units sold.
 - **total_price**: Computed total amount (price × quantity).
 - **date**: Date of sale.
-

10. Leave

Requests for time off by employees.

- **leave_id** (*Primary Key*): Unique identifier.
 - **employee_id** (*Foreign Key* → *Employees.employee_id*): Applicant employee.
 - **type**: Type of leave (e.g., Sick, Casual).
 - **start_date**: Starting date of leave.
 - **end_date**: Ending date of leave.
 - **status** (*ENUM*: *pending, approved, rejected*): Approval state.
-

11. Expense

Any organizational spending.

- **expense_id** (*Primary Key*): Unique ID.
 - **description**: Nature of the expense.
 - **amount**: Monetary value.
 - **date**: Date of expense.
-

◆ Relationships Overview (Diagrammatically Inferred)

- One **Category** → Many **Products**
- One **Supplier** → Many **Purchases**
- One **Product** → Many **Purchases**
- One **Department** → Many **Employees**
- One **Employee** → Many **Attendances**
- One **Customer** → Many **Sales**
- One **Product** → Many **Sales**
- One **Employee** → Many **Leaves**

Finalized Normalized Relations :

Each relation (table) is in **1NF, 2NF, and 3NF**, with **no repeating groups, partial dependencies, or transitive dependencies**.

1. departments

| Attribute | Type | Constraints |
|-----------|---------|-----------------------------|
| id | INT | Primary Key, Auto-Increment |
| name | VARCHAR | Unique, Not Null |

2. employees

| Attribute | Type | Constraints |
|---------------|---------|-------------------------------|
| id | INT | Primary Key, Auto-Increment |
| name | VARCHAR | Not Null |
| department_id | INT | Foreign Key → departments(id) |
| salary | DECIMAL | Not Null |
| contact | VARCHAR | Nullable |

Fully normalized: No repeating or dependent fields; department_id relates to department table.

3. categories

| Attribute | Type | Constraints |
|-----------|---------|------------------|
| id | INT | Primary Key |
| name | VARCHAR | Unique, Not Null |

4. products

| Attribute | Type | Constraints |
|-------------|---------|------------------------------|
| id | INT | Primary Key |
| name | VARCHAR | Not Null |
| category_id | INT | Foreign Key → categories(id) |

| Attribute | Type | Constraints |
|-----------|---------|-------------|
| price | DECIMAL | Not Null |
| quantity | INT | Default 0 |

Normalized: No category names here, just a foreign key to categories.

5. suppliers

| Attribute | Type | Constraints |
|-----------|---------|-------------|
| id | INT | Primary Key |
| name | VARCHAR | Not Null |
| contact | VARCHAR | Nullable |
| address | VARCHAR | Nullable |

6. purchases

| Attribute | Type | Constraints |
|-------------|---------|-----------------------------|
| id | INT | Primary Key |
| product_id | INT | Foreign Key → products(id) |
| supplier_id | INT | Foreign Key → suppliers(id) |
| quantity | INT | Not Null |
| total_cost | DECIMAL | Not Null (auto-calculated) |
| date | DATE | Not Null |

7. customers

| Attribute | Type | Constraints |
|-----------|---------|------------------|
| id | INT | Primary Key |
| name | VARCHAR | Not Null |
| email | VARCHAR | Unique, Nullable |
| phone | VARCHAR | Nullable |
| address | VARCHAR | Nullable |

8. sales

| Attribute | Type | Constraints |
|-------------|---------|-----------------------------|
| id | INT | Primary Key |
| customer_id | INT | Foreign Key → customers(id) |
| product_id | INT | Foreign Key → products(id) |
| quantity | INT | Not Null |
| total_price | DECIMAL | Not Null (price × quantity) |
| date | DATE | Not Null |

No redundancy; price is not stored from products directly.

9. attendances

| Attribute | Type | Constraints |
|-------------|------|-----------------------------|
| id | INT | Primary Key |
| employee_id | INT | Foreign Key → employees(id) |
| date | DATE | Not Null |
| status | ENUM | Values: 'Present', 'Absent' |

10. leaves

| Attribute | Type | Constraints |
|-------------|---------|---|
| id | INT | Primary Key |
| employee_id | INT | Foreign Key → employees(id) |
| type | VARCHAR | e.g., Sick, Casual |
| start_date | DATE | Not Null |
| end_date | DATE | Not Null, >= start_date |
| status | ENUM | Values: 'pending', 'approved', 'rejected' |

11. expenses

| Attribute | Type | Constraints |
|-------------|---------|-------------|
| id | INT | Primary Key |
| description | TEXT | Not Null |
| amount | DECIMAL | Not Null |

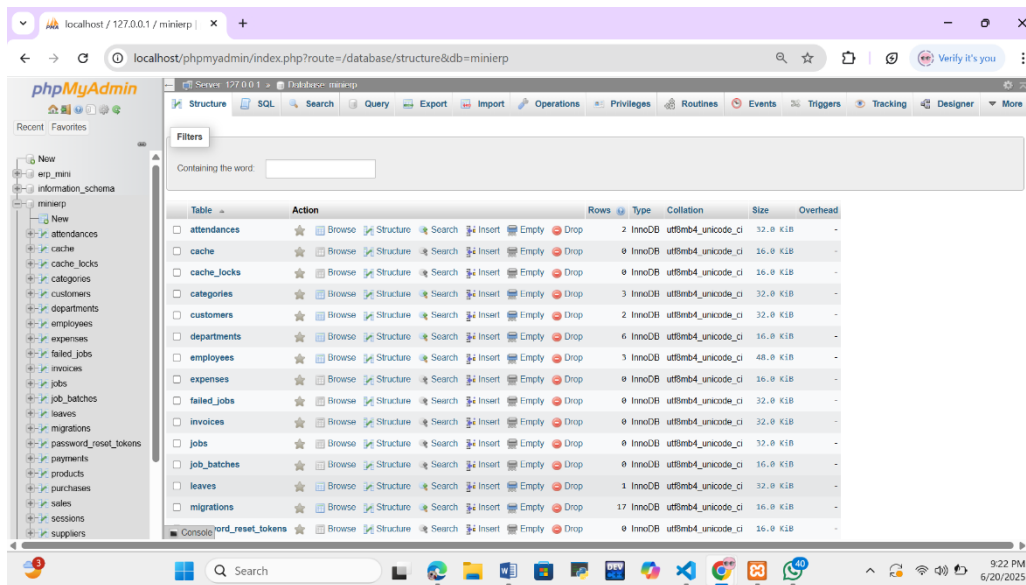
| Attribute | Type | Constraints |
|-----------|------|-------------|
| date | DATE | Not Null |

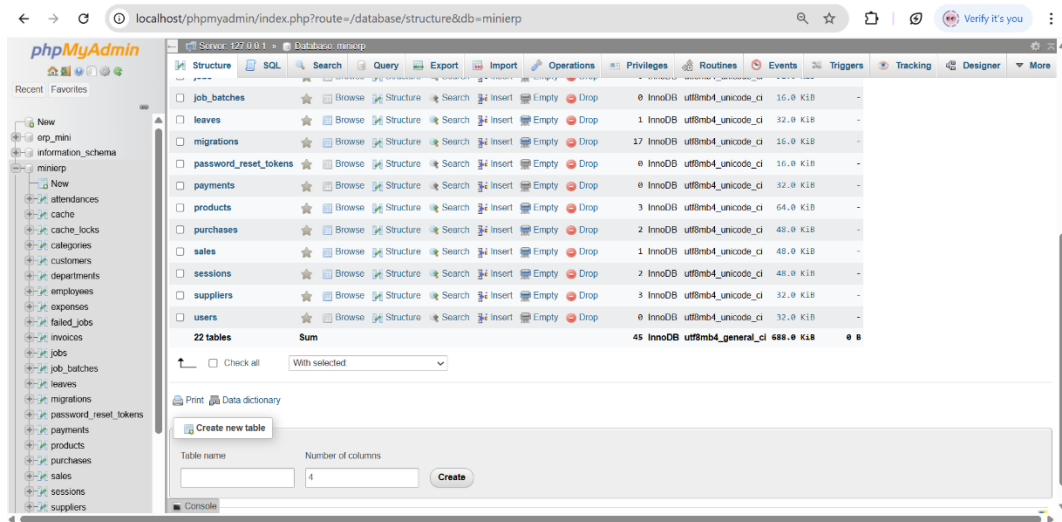
Normalization Summary

- All tables are in **1NF** (no repeating or multi-valued attributes).
- **2NF** is satisfied (no partial dependencies).
- **3NF** is ensured (no transitive dependencies).

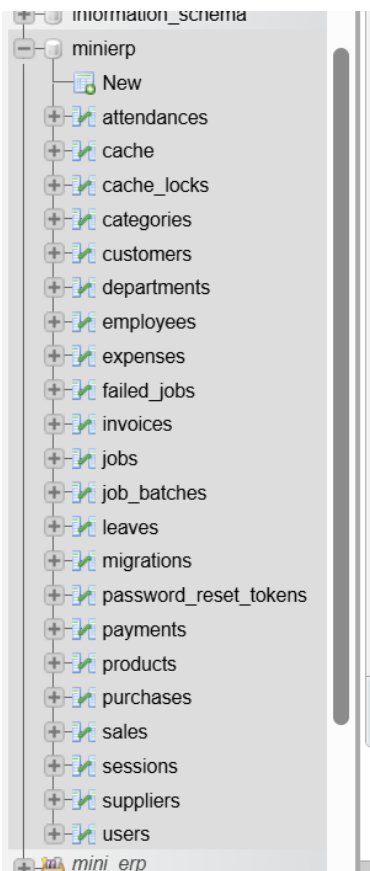
SQL Database Tables and Queries:

Database Creation:





Tables:



Viewing Metadata:

- Structure of attendance table:

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|--------------------------|---------------|------------------------------------|--------------------|------------|------|---------|----------|----------------|--------------------|
| <input type="checkbox"/> | 1 id | bigint(20) | | UNSIGNED | No | None | | AUTO_INCREMENT | Change Drop More |
| <input type="checkbox"/> | 2 employee_id | bigint(20) | | UNSIGNED | No | None | | | Change Drop More |
| <input type="checkbox"/> | 3 date | date | | | No | None | | | Change Drop More |
| <input type="checkbox"/> | 4 status | enum('Present', 'Absent', 'Leave') | utf8mb4_unicode_ci | | No | Present | | | Change Drop More |
| <input type="checkbox"/> | 5 created_at | timestamp | | | Yes | NULL | | | Change Drop More |
| <input type="checkbox"/> | 6 updated_at | timestamp | | | Yes | NULL | | | Change Drop More |

☐ Check all With selected: Browse Change Drop Primary Unique Index Spatial Fulltext Add to central columns
 Remove from central columns

Print Propose table structure Track table Move columns Normalize

Add 1 column(s) after updated_at Go

- Structure of categories table:

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|--------------------------|--------------|--------------|--------------------|------------|------|---------|----------|----------------|--------------------|
| <input type="checkbox"/> | 1 id | bigint(20) | | UNSIGNED | No | None | | AUTO_INCREMENT | Change Drop More |
| <input type="checkbox"/> | 2 name | varchar(255) | utf8mb4_unicode_ci | | No | None | | | Change Drop More |
| <input type="checkbox"/> | 3 created_at | timestamp | | | Yes | NULL | | | Change Drop More |
| <input type="checkbox"/> | 4 updated_at | timestamp | | | Yes | NULL | | | Change Drop More |

☐ Check all With selected: Browse Change Drop Primary Unique Index Spatial Fulltext Add to central columns
 Remove from central columns

Print Propose table structure Track table Move columns Normalize

- Structure of categories table:

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|--------------------------|--------------|--------------|--------------------|------------|------|---------|----------|----------------|--------------------|
| <input type="checkbox"/> | 1 id | bigint(20) | | UNSIGNED | No | None | | AUTO_INCREMENT | Change Drop More |
| <input type="checkbox"/> | 2 name | varchar(255) | utf8mb4_unicode_ci | | No | None | | | Change Drop More |
| <input type="checkbox"/> | 3 created_at | timestamp | | | Yes | NULL | | | Change Drop More |
| <input type="checkbox"/> | 4 updated_at | timestamp | | | Yes | NULL | | | Change Drop More |

☐ Check all With selected: Browse Change Drop Primary Unique Index Spatial Fulltext Add to central columns
 Remove from central columns

Server: 127.0.0.1 » Database: minerp » Table: customers

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[Operations](#)
[Tracking](#)
[Triggers](#)

[Table structure](#)
[Relation view](#)

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|----------------------------|-------------------|--------------|--------------------|------------|------|---------|----------|----------------|--|
| <input type="checkbox"/> 1 | id | bigint(20) | | UNSIGNED | No | None | | AUTO_INCREMENT | Change Drop More |
| <input type="checkbox"/> 2 | name | varchar(255) | utf8mb4_unicode_ci | | No | None | | | Change Drop More |
| <input type="checkbox"/> 3 | email | varchar(255) | utf8mb4_unicode_ci | | No | None | | | Change Drop More |
| <input type="checkbox"/> 4 | phone | varchar(255) | utf8mb4_unicode_ci | | Yes | NULL | | | Change Drop More |
| <input type="checkbox"/> 5 | address | text | utf8mb4_unicode_ci | | Yes | NULL | | | Change Drop More |
| <input type="checkbox"/> 6 | created_at | timestamp | | | Yes | NULL | | | Change Drop More |
| <input type="checkbox"/> 7 | updated_at | timestamp | | | Yes | NULL | | | Change Drop More |

☐ Check all
 With selected:
 [Browse](#)
[Change](#)
[Drop](#)
[Primary](#)
[Unique](#)
[Index](#)
[Spatial](#)
[Fulltext](#)
[Add to](#)

[Remove from central columns](#)

- Structure of employees table:

Server: 127.0.0.1 » Database: minerp » Table: employees

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[Operations](#)
[Tracking](#)
[Triggers](#)

Showing rows 0 - 2 (3 total, Query took 0.0015 seconds)

[SELECT * FROM `employees`](#)

[Profiling](#)
[Edit inline](#)
[Edit](#)
[Explain SQL](#)
[Create PHP code](#)
[Refresh](#)

☐ Show all
 Number of rows: 25
 Filter rows: Search this table
 Sort by key: None

Extra options

| | id | name | email | department_id | created_at | updated_at |
|---|----|------------|------------------------|---------------|---------------------|---------------------|
| <input type="checkbox"/> Edit Copy Delete | 2 | fiza.txt | fizanihar125@gmail.com | 4 | 2025-06-19 17:48:23 | 2025-06-19 17:48:23 |
| <input type="checkbox"/> Edit Copy Delete | 3 | FizaGull | fizanihar01@gmail.com | 2 | 2025-06-20 08:56:43 | 2025-06-20 08:56:43 |
| <input type="checkbox"/> Edit Copy Delete | 4 | Fiza Nihar | fiza@example.com | 1 | 2025-06-20 12:03:16 | 2025-06-20 12:03:16 |

☐ Check all
 With selected:
 [Edit](#)
[Copy](#)
[Delete](#)
[Export](#)

☐ Show all
 Number of rows: 25
 Filter rows: Search this table
 Sort by key: None

Query results operations

- Structure of expenses table:

Server: 127.0.0.1 » Database: minerp » Table: expenses

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[Operations](#)
[Tracking](#)
[Triggers](#)

[Table structure](#)
[Relation view](#)

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|----------------------------|--------------------|---------------|--------------------|------------|------|---------|----------|----------------|--|
| <input type="checkbox"/> 1 | id | bigint(20) | | UNSIGNED | No | None | | AUTO_INCREMENT | Change Drop More |
| <input type="checkbox"/> 2 | description | varchar(255) | utf8mb4_unicode_ci | | No | None | | | Change Drop More |
| <input type="checkbox"/> 3 | amount | decimal(10,2) | | | No | None | | | Change Drop More |
| <input type="checkbox"/> 4 | date | date | | | No | None | | | Change Drop More |
| <input type="checkbox"/> 5 | created_at | timestamp | | | Yes | NULL | | | Change Drop More |
| <input type="checkbox"/> 6 | updated_at | timestamp | | | Yes | NULL | | | Change Drop More |

☐ Check all
 With selected:
 [Browse](#)
[Change](#)
[Drop](#)
[Primary](#)
[Unique](#)
[Index](#)
[Spatial](#)
[Fulltext](#)
[Add to central](#)

[Remove from central columns](#)

[Print](#)
[Process table structure](#)
[Test table](#)
[Mass columns](#)
[Normalize](#)

Server: 127.0.0.1 > Database: minerp > Table: leaves

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Table structure Relation view

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|----------------------------|-------------|--------------|--------------------|------------|------|---------|----------|----------------|--------------------|
| <input type="checkbox"/> 1 | id | bigint(20) | | UNSIGNED | No | None | | AUTO_INCREMENT | Change Drop More |
| <input type="checkbox"/> 2 | employee_id | bigint(20) | | UNSIGNED | No | None | | | Change Drop More |
| <input type="checkbox"/> 3 | type | varchar(255) | utf8mb4_unicode_ci | | No | None | | | Change Drop More |
| <input type="checkbox"/> 4 | start_date | date | | | No | None | | | Change Drop More |
| <input type="checkbox"/> 5 | end_date | date | | | No | None | | | Change Drop More |
| <input type="checkbox"/> 6 | status | varchar(255) | utf8mb4_unicode_ci | | No | Pending | | | Change Drop More |
| <input type="checkbox"/> 7 | created_at | timestamp | | | Yes | NULL | | | Change Drop More |
| <input type="checkbox"/> 8 | updated_at | timestamp | | | Yes | NULL | | | Change Drop More |

☐ Check all
 With selected: Browse Change Drop Primary Unique Index Spatial Fulltext Add to central columns
 Remove from central columns

- Structure of payments table:

Server: 127.0.0.1 > Database: minerp > Table: payments

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Table structure Relation view

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|----------------------------|------------|---------------|-----------|------------|------|---------|----------|----------------|--------------------|
| <input type="checkbox"/> 1 | id | bigint(20) | | UNSIGNED | No | None | | AUTO_INCREMENT | Change Drop More |
| <input type="checkbox"/> 2 | invoice_id | bigint(20) | | UNSIGNED | No | None | | | Change Drop More |
| <input type="checkbox"/> 3 | amount | decimal(10,2) | | | No | None | | | Change Drop More |
| <input type="checkbox"/> 4 | paid_on | date | | | No | None | | | Change Drop More |
| <input type="checkbox"/> 5 | created_at | timestamp | | | Yes | NULL | | | Change Drop More |
| <input type="checkbox"/> 6 | updated_at | timestamp | | | Yes | NULL | | | Change Drop More |

☐ Check all
 With selected: Browse Change Drop Primary Unique Index Spatial Fulltext Add to central columns
 Remove from central columns

- Structure of products table:

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Table structure Relation view

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|----------------------------|-------------|---------------|--------------------|------------|------|---------|----------|----------------|--------------------|
| <input type="checkbox"/> 1 | id | bigint(20) | | UNSIGNED | No | None | | AUTO_INCREMENT | Change Drop More |
| <input type="checkbox"/> 2 | name | varchar(255) | utf8mb4_unicode_ci | | No | None | | | Change Drop More |
| <input type="checkbox"/> 3 | sku | varchar(255) | utf8mb4_unicode_ci | | No | None | | | Change Drop More |
| <input type="checkbox"/> 4 | quantity | int(11) | | | No | 0 | | | Change Drop More |
| <input type="checkbox"/> 5 | price | decimal(10,2) | | | No | None | | | Change Drop More |
| <input type="checkbox"/> 6 | category_id | bigint(20) | | UNSIGNED | No | None | | | Change Drop More |
| <input type="checkbox"/> 7 | created_at | timestamp | | | Yes | NULL | | | Change Drop More |
| <input type="checkbox"/> 8 | updated_at | timestamp | | | Yes | NULL | | | Change Drop More |
| <input type="checkbox"/> 9 | supplier_id | bigint(20) | | UNSIGNED | Yes | NULL | | | Change Drop More |

☐ Check all
 With selected: Browse Change Drop Primary Unique Index Spatial Fulltext Add to central columns
 Remove from central columns

- Structure of purchases table:

Server: 127.0.0.1 » Database: minierp » Table: products

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[Operations](#)
[Tracking](#)
[Triggers](#)

[Table structure](#)
[Relation view](#)

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|----------------------------|-------------|---------------|-----------|------------|------|---------|----------|----------------|--|
| <input type="checkbox"/> 1 | id | bigint(20) | | UNSIGNED | No | None | | AUTO_INCREMENT | Change Drop More |
| <input type="checkbox"/> 2 | product_id | bigint(20) | | UNSIGNED | No | None | | | Change Drop More |
| <input type="checkbox"/> 3 | supplier_id | bigint(20) | | UNSIGNED | No | None | | | Change Drop More |
| <input type="checkbox"/> 4 | quantity | int(11) | | | No | None | | | Change Drop More |
| <input type="checkbox"/> 5 | total_cost | decimal(10,2) | | | No | None | | | Change Drop More |
| <input type="checkbox"/> 6 | created_at | timestamp | | | Yes | NULL | | | Change Drop More |
| <input type="checkbox"/> 7 | updated_at | timestamp | | | Yes | NULL | | | Change Drop More |

☐ Check all
 With selected:
 [Browse](#)
[Change](#)
[Drop](#)
[Primary](#)
[Unique](#)
[Index](#)
[Spatial](#)
[Fulltext](#)
[Add to central col](#)

[Remove from central columns](#)

[Print](#)
[Propose table structure](#)
[Track table](#)
[Move columns](#)
[Normalize](#)

- Structure of sales table:

Server: 127.0.0.1 » Database: minierp » Table: sales

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[Operations](#)
[Tracking](#)
[Triggers](#)

[Table structure](#)
[Relation view](#)

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|----------------------------|-------------|---------------|-----------|------------|------|---------|----------|----------------|--|
| <input type="checkbox"/> 1 | id | bigint(20) | | UNSIGNED | No | None | | AUTO_INCREMENT | Change Drop More |
| <input type="checkbox"/> 2 | product_id | bigint(20) | | UNSIGNED | No | None | | | Change Drop More |
| <input type="checkbox"/> 3 | customer_id | bigint(20) | | UNSIGNED | No | None | | | Change Drop More |
| <input type="checkbox"/> 4 | quantity | int(11) | | | No | None | | | Change Drop More |
| <input type="checkbox"/> 5 | total_price | decimal(10,2) | | | No | None | | | Change Drop More |
| <input type="checkbox"/> 6 | date | date | | | No | None | | | Change Drop More |
| <input type="checkbox"/> 7 | created_at | timestamp | | | Yes | NULL | | | Change Drop More |
| <input type="checkbox"/> 8 | updated_at | timestamp | | | Yes | NULL | | | Change Drop More |

☐ Check all
 With selected:
 [Browse](#)
[Change](#)
[Drop](#)
[Primary](#)
[Unique](#)
[Index](#)
[Spatial](#)
[Fulltext](#)
[Add to cen](#)

[Remove from central columns](#)

- Structure of suppliers table:

Server: 127.0.0.1 - 2 Database: minicp - 2 Table: Suppliers

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Insert](#)
[Export](#)
[Import](#)
[Privileges](#)
[Operations](#)
[Tracking](#)
[Triggers](#)

[Table structure](#)
[Relation view](#)

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|----------------------------|-------------------|--------------|--------------------|------------|------|---------|----------|----------------|--------------------|
| <input type="checkbox"/> 1 | id | bigint(20) | | UNSIGNED | No | None | | AUTO_INCREMENT | Change Drop More |
| <input type="checkbox"/> 2 | name | varchar(255) | utf8mb4_unicode_ci | | No | None | | | Change Drop More |
| <input type="checkbox"/> 3 | email | varchar(255) | utf8mb4_unicode_ci | | Yes | NULL | | | Change Drop More |
| <input type="checkbox"/> 4 | phone | varchar(255) | utf8mb4_unicode_ci | | Yes | NULL | | | Change Drop More |
| <input type="checkbox"/> 5 | created_at | timestamp | | | Yes | NULL | | | Change Drop More |
| <input type="checkbox"/> 6 | updated_at | timestamp | | | Yes | NULL | | | Change Drop More |

☐ Check all *With selected:* Browse Change Drop Primary Unique Index Spatial Fulltext Add to

Remove from central columns

Print Propose table structure Track table Move columns Normalize

QUERIES:

Query 1: Show All Sales with Product and Customer Name:

```
1 SELECT sales.id, products.name AS product, customers.name AS customer, sales.quantity, sales.total_price
2 FROM sales
3 JOIN products ON sales.product_id = products.id
4 JOIN customers ON sales.customer_id = customers.id;
5
6
```

OUTPUT:

Extra Options

| id | product | customer | quantity | total_price |
|----|--------------|---------------|----------|-------------|
| 1 | Demo Product | Test Customer | 2 | 200.00 |

Query 2: Products with Category Info:


Run SQL query/queries on table minierp.categories: 

```
1 SELECT products.name, categories.name AS category, products.quantity
2 FROM products
3 JOIN categories ON products.category_id = categories.id;
4
5
6
```

OUTPUT:

| name | category | quantity |
|--------------|----------|----------|
| good | bye | 1 |
| Fiza | fizagull | 3 |
| Demo Product | General | 12 |

Query 3: Purchases with Supplier Info:

Run SQL query/queries on table minierp.categories: 

```

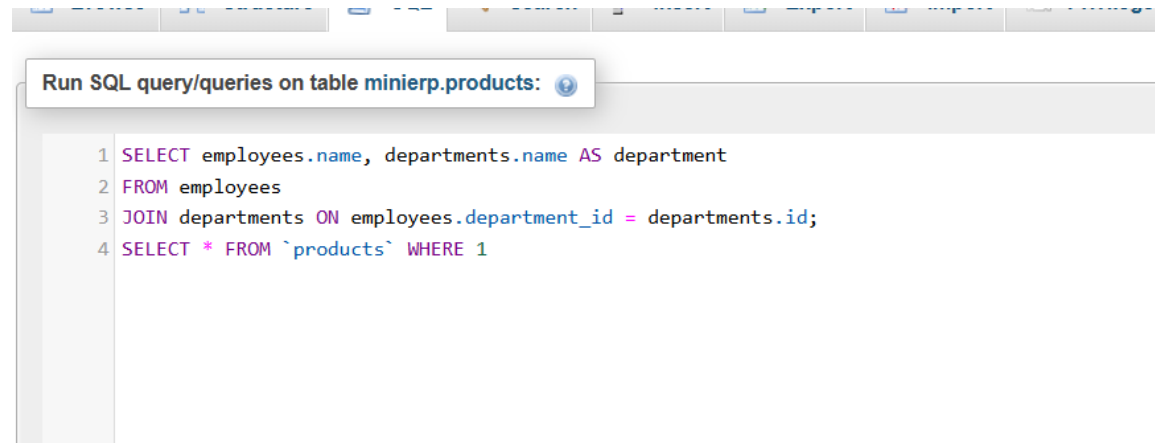
1 SELECT purchases.id, products.name, suppliers.name AS supplier, purchases.quantity, purchases.total_cost
2 FROM purchases
3 JOIN products ON purchases.product_id = products.id
4 JOIN suppliers ON purchases.supplier_id = suppliers.id;
5
6
7
8
9

```

OUTPUT:

| id | name | supplier | quantity | total_cost |
|----|--------------|---------------|----------|------------|
| 9 | Demo Product | Demo Supplier | 2 | 200.00 |
| 10 | Demo Product | Demo Supplier | 2 | 200.00 |

Query 4: Employees with Department Names:



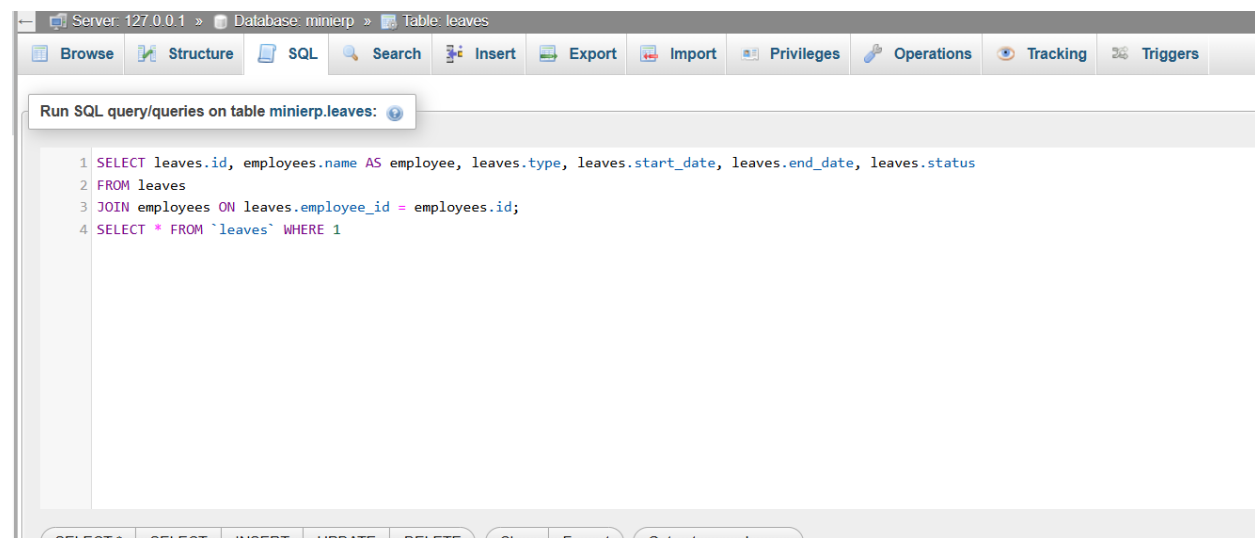
OUTPUT:

Extra options

| name | department |
|------------|------------|
| fiza.txt | Marketing |
| FizaGull | HR |
| Fiza Nihar | IT |

Show all | Number of rows: 25

Query 5: Show All Leave Requests:



OUTPUT:

| Extra options | | | | | |
|---------------|------------|------|------------|------------|---------|
| id | employee | type | start_date | end_date | status |
| 2 | Fiza Nihar | Sick | 2021-05-20 | 2025-06-20 | pending |

Comprehensive Laravel Implementation Details

This section outlines how the **ERP System** was developed using the **Laravel framework**, providing details on models, controllers, routes, and frontend interaction for each module.

Laravel Project Structure

The project follows a RESTful API architecture and is divided into modular components, each representing a core entity: Departments, Employees, Categories, Products, Suppliers, Purchases, Attendances, Leaves, Customers, and Sales.

Laravel Models Used

Each entity in the ERP has a dedicated Eloquent Model:

- Department.php
- Employee.php
- Category.php
- Product.php
- Supplier.php
- Purchase.php
- Attendance.php
- Leave.php
- Customer.php
- Sale.php

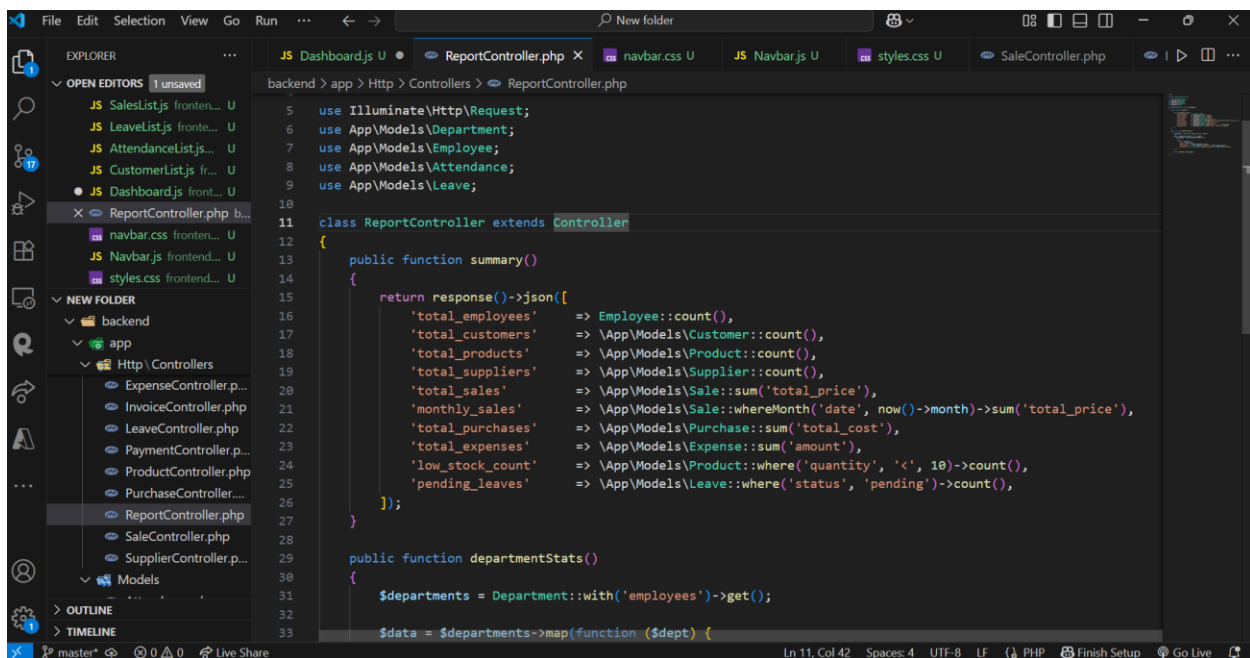
These models define table relationships and encapsulate business logic.

Controllers

Each resource has its own controller:

- DepartmentController: Handles department CRUD.
- EmployeeController: Manages employees and their department relationship.
- CategoryController, ProductController, SupplierController, etc.: Control respective inventories.
- PurchaseController: On storing a purchase, auto-increments product quantity.
- SaleController: Auto-decrements product quantity upon sale.
- AttendanceController and LeaveController: Manage HR activities.
- CustomerController: Manages CRM functions.
- ReportController: Generates summary metrics used in the dashboard.

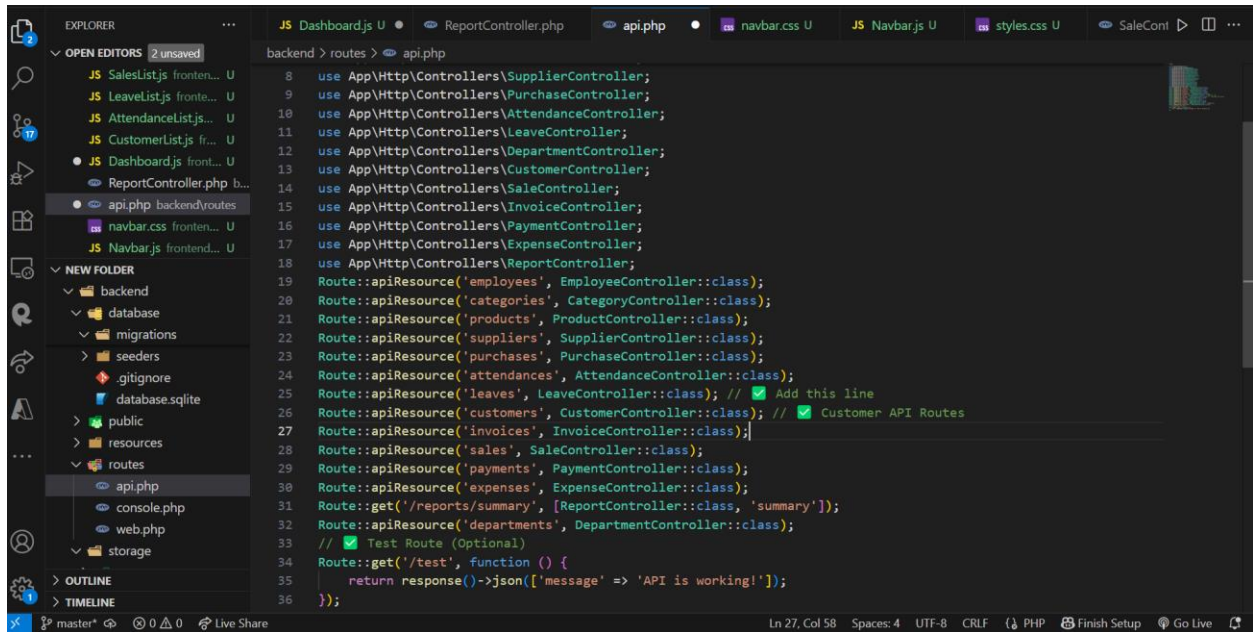
Example from ReportController:



```
5 use Illuminate\Http\Request;
6 use App\Models\Department;
7 use App\Models\Employee;
8 use App\Models\Attendance;
9 use App\Models\Leave;
10
11 class ReportController extends Controller
12 {
13     public function summary()
14     {
15         return response()->json([
16             'total_employees' => Employee::count(),
17             'total_customers' => \App\Models\Customer::count(),
18             'total_products' => \App\Models\Product::count(),
19             'total_suppliers' => \App\Models\Supplier::count(),
20             'total_sales' => \App\Models\Sale::sum('total_price'),
21             'monthly_sales' => \App\Models\Sale::whereMonth('date', now()->month)->sum('total_price'),
22             'total_purchases' => \App\Models\Purchase::sum('total_cost'),
23             'total_expenses' => \App\Models\Expense::sum('amount'),
24             'low_stock_count' => \App\Models\Product::where('quantity', '<', 10)->count(),
25             'pending_leaves' => \App\Models\Leave::where('status', 'pending')->count(),
26         ]);
27     }
28
29     public function departmentStats()
30     {
31         $departments = Department::with('employees')->get();
32
33         $data = $departments->map(function ($dept) {
```

API Routes

Defined in routes/api.php:

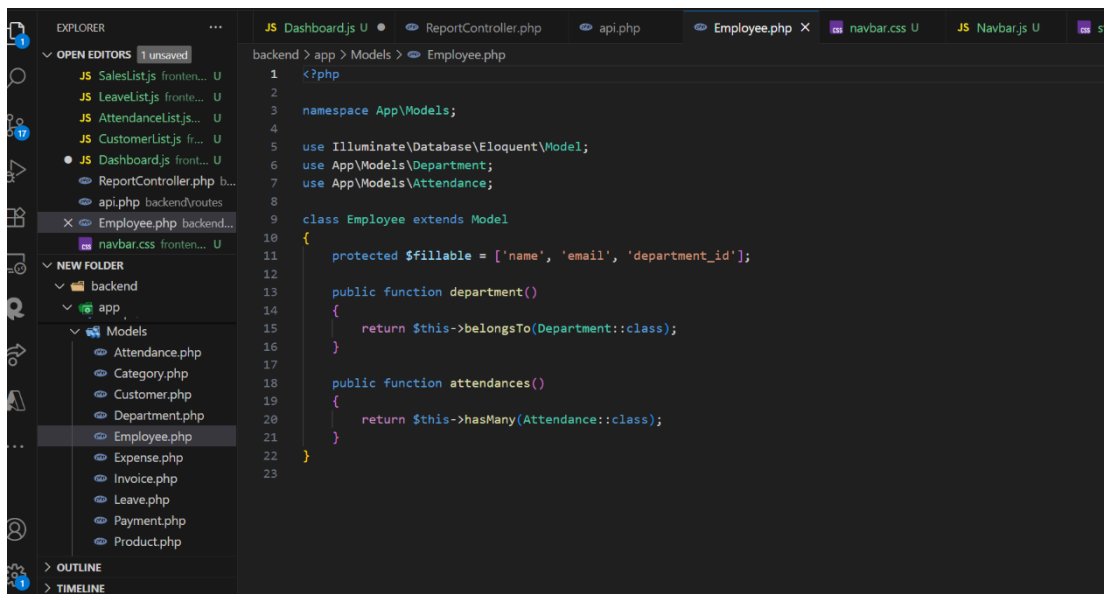


```
backend > routes > api.php
8 use App\Http\Controllers\SupplierController;
9 use App\Http\Controllers\PurchaseController;
10 use App\Http\Controllers\AttendanceController;
11 use App\Http\Controllers\LeaveController;
12 use App\Http\Controllers\DepartmentController;
13 use App\Http\Controllers\CustomerController;
14 use App\Http\Controllers\SaleController;
15 use App\Http\Controllers\InvoiceController;
16 use App\Http\Controllers\PaymentController;
17 use App\Http\Controllers\ExpenseController;
18 use App\Http\Controllers\ReportController;
19 Route::apiResource('employees', EmployeeController::class);
20 Route::apiResource('categories', CategoryController::class);
21 Route::apiResource('products', ProductController::class);
22 Route::apiResource('suppliers', SupplierController::class);
23 Route::apiResource('purchases', PurchaseController::class);
24 Route::apiResource('attendances', AttendanceController::class);
25 Route::apiResource('leaves', LeaveController::class); // Add this line
26 Route::apiResource('customers', CustomerController::class); // Customer API Routes
27 Route::apiResource('invoices', InvoiceController::class);
28 Route::apiResource('sales', SaleController::class);
29 Route::apiResource('payments', PaymentController::class);
30 Route::apiResource('expenses', ExpenseController::class);
31 Route::get('/reports/summary', [ReportController::class, 'summary']);
32 Route::apiResource('departments', DepartmentController::class);
33 // Test Route (Optional)
34 Route::get('/test', function () {
35     return response()->json(['message' => 'API is working!']);
36 });
```

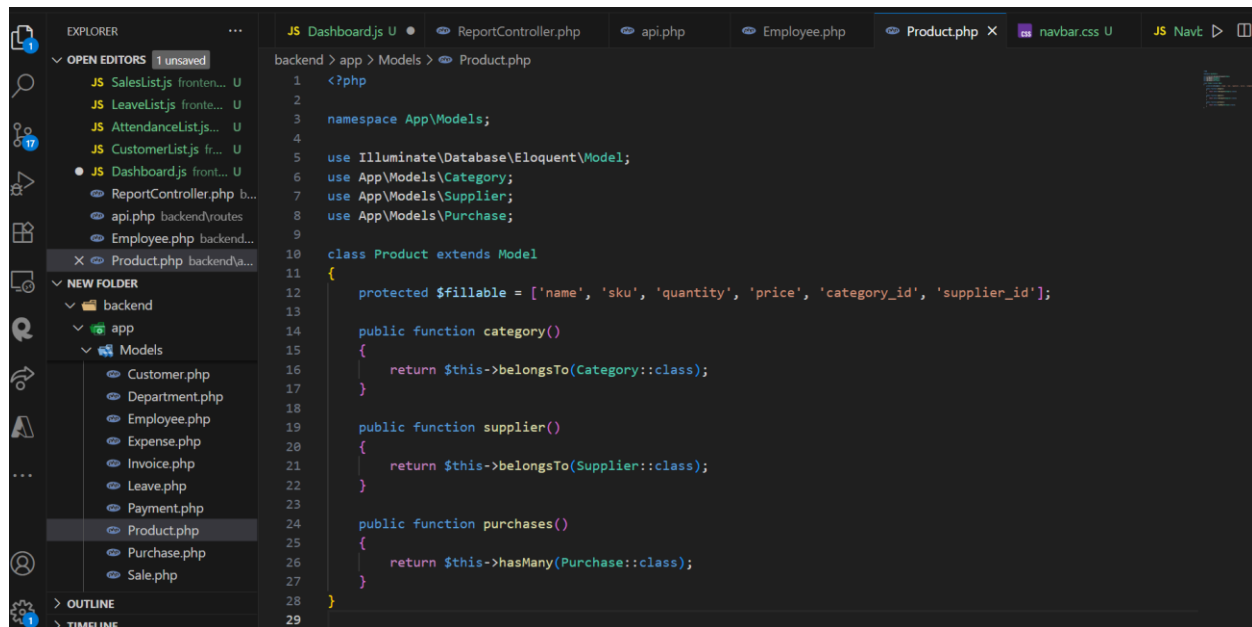
These routes are consumed by the React frontend using fetch() for GET and POST operations.

Relationships

Defined inside models:

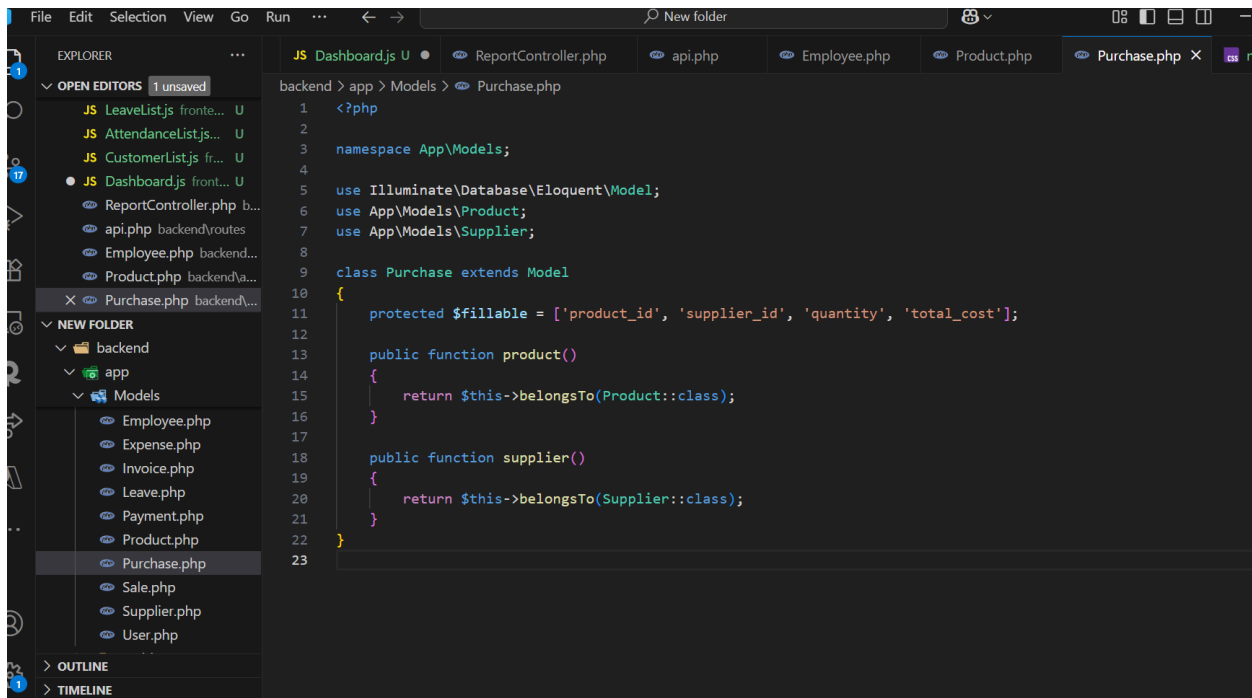


```
backend > app > Models > Employee.php
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Model;
6 use App\Models\Department;
7 use App\Models\Attendance;
8
9 class Employee extends Model
10 {
11     protected $fillable = ['name', 'email', 'department_id'];
12
13     public function department()
14     {
15         return $this->belongsTo(Department::class);
16     }
17
18     public function attendances()
19     {
20         return $this->hasMany(Attendance::class);
21     }
22 }
23
```



The screenshot shows the Visual Studio Code editor with the 'Product.php' file open. The Explorer sidebar on the left shows the project structure with folders 'backend' and 'app', and a 'Models' folder containing several PHP files. The main editor area displays the code for 'Product.php'.

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Model;
6 use App\Models\Category;
7 use App\Models\Supplier;
8 use App\Models\Purchase;
9
10 class Product extends Model
11 {
12     protected $fillable = ['name', 'sku', 'quantity', 'price', 'category_id', 'supplier_id'];
13
14     public function category()
15     {
16         return $this->belongsTo(Category::class);
17     }
18
19     public function supplier()
20     {
21         return $this->belongsTo(Supplier::class);
22     }
23
24     public function purchases()
25     {
26         return $this->hasMany(Purchase::class);
27     }
28 }
29
```



The screenshot shows the Visual Studio Code editor with the 'Purchase.php' file open. The Explorer sidebar on the left shows the project structure with folders 'backend' and 'app', and a 'Models' folder containing several PHP files. The main editor area displays the code for 'Purchase.php'.

```
1 <?php
2
3 namespace App\Models;
4
5 use Illuminate\Database\Eloquent\Model;
6 use App\Models\Product;
7 use App\Models\Supplier;
8
9 class Purchase extends Model
10 {
11     protected $fillable = ['product_id', 'supplier_id', 'quantity', 'total_cost'];
12
13     public function product()
14     {
15         return $this->belongsTo(Product::class);
16     }
17
18     public function supplier()
19     {
20         return $this->belongsTo(Supplier::class);
21     }
22 }
23
```

Frontend Interaction (React)

Each module has a React component (e.g., EmployeeList.js, DepartmentList.js) that:

- Fetches data using Laravel API.
- Submits forms to create new entries.
- Shows lists with edit/delete functionality.

```

1  import React, { useEffect, useState } from 'react';
2  import './styles.css';
3
4  const LeaveList = () => {
5    const [items, setItems] = useState([]);
6    const [form, setForm] = useState({
7      employee_id: "",
8      start_date: "",
9      end_date: "",
10     type: "",
11     status: "pending"
12   });
13   const [editId, setEditId] = useState(null);
14
15   useEffect(() => {
16     fetchLeaves();
17   }, []);
18
19   const fetchLeaves = () => {
20     fetch('http://127.0.0.1:8000/api/leaves')
21       .then(res => res.json())
22       .then(data => setItems(data));
23   };
24
25   const handleChange = (e) => {
26     setForm({ ...form, [e.target.name]: e.target.value });
27   };
28
29   const handleSubmit = async (e) => {

```

Special Logic Implemented

- **Inventory Auto-Update:**
 - In PurchaseController, Product::increment('quantity', \$request->quantity);
 - In SaleController, Product::decrement('quantity', \$request->quantity);
- **Leave Approval Logic:**
 - Status defaults to pending and can be updated later.
- **Dashboard Summary:**
 - Data fetched from ReportController@summary and visualized on a dashboard React component using cards.

Laravel Features Used

- Eloquent ORM for database interaction
- API Resource Routing
- Form validation via Request class or inline
- Controller-based logic separation
- Middleware (optional for future role-based access)

ERP Dashboard

Dashboard

Employees

Departments

Categories

Products

Suppliers

Purchases

Customers

Sales

Attendances

Leaves

ERP Summary Dashboard

Total Employees

3

Total Customers

2

Products in Stock

3

Suppliers

3

Low Stock Products

2

Total Sales (Rs)

Rs. 200.00

Total Purchases (Rs)

Rs. 400.00

Total Expenses (Rs)

Rs. 0

ERP Dashboard

Dashboard

Employees

Departments

Categories

Products

Suppliers

Purchases

Customers

Sales

Attendances

Leaves

Employee List

Name

Email

Select Department

Add Employee

fiza.txt — fizanihar125@gmail.com (Dept ID: 4)

EditDelete

FizaGull — fizanihar01@gmail.com (Dept ID: 2)

EditDelete

Fiza Nihar — fiza@example.com (Dept ID: 1)

EditDelete

ERP Dashboard

Dashboard

Employees

Departments

Categories

Products

Suppliers

Purchases

Customers

Sales

Attendances

Leaves

localhost:3000/departments

Verify it's you

Department

Department Name

Add Department

IT (ID: 1)

HR (ID: 2)

Finance (ID: 3)

Marketing (ID: 4)

Operations (ID: 5)

gulli (ID: 6)

ERP Dashboard

Dashboard

Employees

Departments

Categories

Products

Suppliers

Purchases

Customers

Sales

Attendances

Leaves

localhost:3000/categories

Verify it's you

Category List

Category name

Add Category

bye

Edit

Delete

fizagull

Edit

Delete

General

Edit

Delete

ERP Dashboard

Dashboard

Employees

Departments

Categories

Products

Suppliers

Purchases

Customers

Sales

Attendances

Leaves

Product List

Name

SKU

Quantity

Price

Category ID

Supplier ID

Add Product

good — SKU: hoo, Qty: 1, Price: Rs.2.00

Category ID: 1, Supplier ID:

EditDelete

Fiza — SKU: gkb, Qty: 3, Price: Rs.3.00

Category ID: 2, Supplier ID: 2

EditDelete

Demo Product — SKU: SKU001, Qty: 12, Price: Rs.100.00

Category ID: 3, Supplier ID: 4

EditDelete

ERP Dashboard

Dashboard

Employees

Departments

Categories

Products

Suppliers

Purchases

Customers

Sales

Attendances

Leaves

Supplier List

Name

Email

Add

fizagull — fizanihar125@gmail.com

EditDelete

amin — ami@gmail.com

EditDelete

Demo Supplier — demo@supplier.com

EditDelete

← → ↺

localhost:3000/purchases

📄 🔍 ☆ 🗂️ ⬇️ 🔒 Verify it's you ⋮

ERP Dashboard

Dashboard

Employees

Departments

Categories

Products

Suppliers

Purchases

Customers

Sales

Attendances

Leaves

Purchase

Product ID

Quantity

Supplier ID

Add Purchase

Product: Demo Product, Supplier: Demo Supplier, Quantity: 2, Total Cost: Rs.200.00

Edit Delete

Product: Demo Product, Supplier: Demo Supplier, Quantity: 2, Total Cost: Rs.200.00

Edit Delete

ERP Dashboard

Dashboard

Employees

Departments

Categories

Products

Suppliers

Purchases

Customers

Sales

Attendances

Leaves

Customers

Name

Email

Phone

Address

Add Customer

Amina — amina01@gmail.com | 03420970410

📍 girls hostel

Edit Delete

Test Customer — customer@test.com | N/A

📍 N/A

Edit Delete

ERP Dashboard

Dashboard

Employees

Departments

Categories

Products

Suppliers

Purchases

Customers

Sales

Attendances

Leaves

localhost:3000/sales

Verify it's you

Sales

Product ID

Customer ID

Quantity

Add Sale

Product: Demo Product, Customer: Test Customer, Quantity: 2

Delete

ERP Dashboard

Dashboard

Employees

Departments

Categories

Products

Suppliers

Purchases

Customers

Sales

Attendances

Leaves

localhost:3000/attendances

Verify it's you

Attendance

Employee ID

mm/dd/yyyy

Status (e.g., Present/Absen)

Add Attendance

Employee ID: 4, Date: 2025-06-17, Status: Present

Edit

Delete

Employee ID: 4, Date: 2025-06-17, Status: Present

Edit

Delete

ERP Dashboard

Dashboard

Employees

Departments

Categories

Products

Suppliers

Purchases

Customers

Sales

Attendances

Leaves

localhost:3000/leaves

Verify it's you

Leave

Employee ID

mm/dd/yyyy

mm/dd/yyyy

Leave Type (e.g., Sick, Cas

Pending

Add Leave

Employee ID: 4, Type: Sick, From: 2021-05-20, To: 2025-06-20, Status: pending

Edit

Delete