**ERP Project Documentation**

**Project Overview**

**What Are We Building?**

We are developing a **cloud-based ERP system** with a **microservices architecture**. This system is subscription-based and designed to cater to multiple organizations (clients) while being managed by an IT Company. The system will provide clients with dashboards to manage operations such as Human Resources (HRM), Customer Relationship Management (CRM), Accounting, Inventory, and Sales. The IT Company will manage subscriptions, monitor client usage, and generate revenue through the system.

**Key Features**

**1. IT Company Dashboard**

This dashboard will allow the service provider (IT Company) to:

* Manage clients and assign subscription plans.
* Monitor client activity, usage, and system health.
* Automate subscription billing and reminders.
* Track revenue generated by the system.
* Generate internal reports for client usage and revenue.

**2. Client Dashboards**

Clients will have role-based dashboards for their organization:

* **Admin Panel**:
  + Manage users (assign roles such as Manager or Employee).
  + Oversee all modules: HRM, CRM, Accounting, Inventory, and Sales.
  + Track financials, invoices, and subscriptions.
* **Manager Panel**:
  + Manage specific departments (e.g., HR, Sales, or Inventory).
  + Generate department-level reports.
* **Employee Panel**:
  + Task-level access for daily operations (e.g., barcode scanning, sales, receipts).

**3. Authentication and Security**

* Secure login for IT Company and clients with **JWT-based authentication**.
* Role-based access control (RBAC) to ensure proper permissions.
* Multi-tenancy for data isolation between clients.

**4. Core Modules**

1. **Human Resources (HRM)**:
   * Employee management, leave tracking, and payroll processing.
2. **Customer Relationship Management (CRM)**:
   * Manage contacts, leads, and sales pipelines.
3. **Accounting**:
   * Track expenses, invoices, and generate financial reports.
4. **Inventory & Sales**:
   * Barcode-based inventory tracking, product catalog, and sales management.
5. **Reporting**:
   * Generate PDF/Excel reports for performance metrics, financials, and trends.

**System Architecture**

**Microservices**

1. **Authentication Service**: Manages user login and role-based access.
2. **HRM Service**: Handles employee data, leave, and payroll.
3. **CRM Service**: Manages contacts, leads, and sales pipelines.
4. **Accounting Service**: Tracks financials, invoices, and expenses.
5. **Inventory & Sales Service**: Manages products, barcodes, stock levels, and sales.
6. **Reporting Service**: Aggregates data for dashboards and exports reports.
7. **Notification Service**: Sends emails and SMS alerts for reminders and approvals.

**Tech Stack**

* **Frontend**: React, Tailwind CSS, Redux.
* **Backend**: Django REST Framework (DRF) + GraphQL.
* **Database**: PostgreSQL + Redis.
* **DevOps**: Docker, Kubernetes, GitHub Actions.
* **Hosting**: Backend on DigitalOcean, Frontend on Vercel.

**Project Workflow**

**IT Company Dashboard Workflow**

1. **Manage Clients**:
   * Add/edit/delete client accounts.
   * Assign subscription plans and activate accounts.
2. **Subscription Management**:
   * Monitor usage limits and notify clients of renewals.
   * Generate subscription-based revenue reports.
3. **Insights**:
   * Track client activity (e.g., active users, sales, inventory).

**Client Dashboard Workflow**

1. **Admin Dashboard**:
   * Oversee all operations and manage users.
2. **Manager Dashboard**:
   * Manage specific modules (HR, Sales, Inventory).
3. **Employee Dashboard**:
   * Perform tasks such as barcode scanning and sales processing.

**Timeline**

**Phase 1: Planning and Setup (Feb 1 - Feb 15)**

* Finalize requirements and workflows.
* Set up GitHub repository and Docker containers.
* Design database schemas and wireframes.

**Phase 2: Core Development (Feb 16 - April 30)**

* Develop individual microservices:
  + Authentication (Feb 16 - Mar 1).
  + HRM (Mar 2 - Mar 20).
  + CRM (Mar 21 - Apr 10).
  + Accounting (Apr 11 - Apr 30).
  + Inventory & Sales (May 1 - May 20).
* Frontend integration with APIs.

**Phase 3: Testing & Integration (May 1 - May 31)**

* Perform unit and integration testing for all services.
* Validate role-based access control and client isolation.

**Phase 4: Final Features & Optimization (June 1 - June 30)**

* Add advanced features like report exports and notifications.
* Optimize APIs and database queries.

**Phase 5: Deployment & Launch (July 1 - Aug 30)**

* Deploy all services to production.
* Onboard IT Company users and initial clients.

**GitHub Project Structure**

erp-microservices/

├── auth-service/

├── hrm-service/

├── crm-service/

├── accounting-service/

├── inventory-sales-service/

├── reporting-service/

├── notification-service/

└── README.md

**Presentation Outline**

**Slide 1: Project Overview**

* Brief description of the ERP system.
* Purpose: A subscription-based ERP for organizations.

**Slide 2: Key Features**

* IT Company and Client Dashboards.
* Role-based access and core modules (HRM, CRM, Accounting, etc.).

**Slide 3: System Architecture**

* Microservices structure.
* Tech stack overview.

**Slide 4: Workflow**

* IT Company and Client workflows.

**Slide 5: Timeline**

* Project phases with milestones.

**Slide 6: Deployment and Scalability**

* Hosting plan and Kubernetes for scalability.

**Slide 7: Conclusion**

* Highlight project goals and expected outcomes.

Let me know if you'd like the detailed **presentation slides** to be prepared or require further customization for this document!