

System Design: Catamac Invoice System

By: Shafayet UI Islam (Audrie)

Title: Catamac Invoice System

Description

This is an invoice system, specifically designed for a coding challenge for Catamac. Any Admin/Accountant (user) who spends time issuing invoices to the other company/business can use this. Admins can manage product, client and invoices through this application

Youtube video of the system walk through: <https://youtu.be/LA80Dc2GNoQ>

Top Level Feature.....	3
Tech Stack.....	3
High-Level Architecture: Request Pipeline for Backend.....	4
Project Structure and Responsibility.....	5
CatamacDbDesign.....	6
API Endpoints.....	7
Core Use-Case.....	7

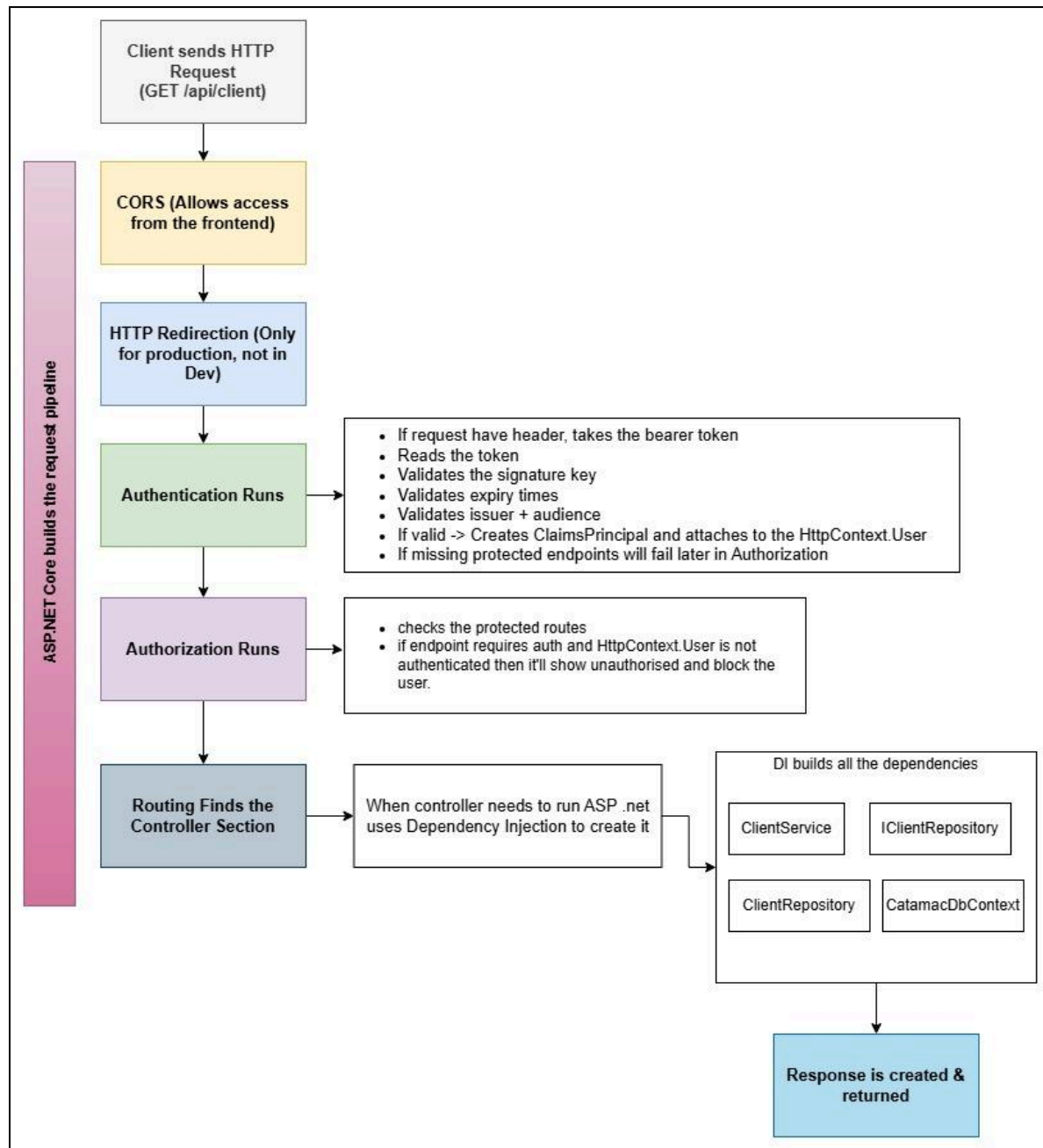
Top Level Feature

1. User can register into the system
2. User can login into the system
3. User can create client details (client management)
4. User can create product details (product management)
5. User can create invoice
 - a. User can add product, quantity, price
 - b. User can add client
 - c. User can select date
6. User can view details of an invoice
7. User can view all the generated invoice

Tech Stack

- ASP .NET Core Web API, EF Core, SQL Server
- Auth: JWT, Password Hashing
- Frontend: React + JS + Bootstrap + Vite
- Tools: Postman

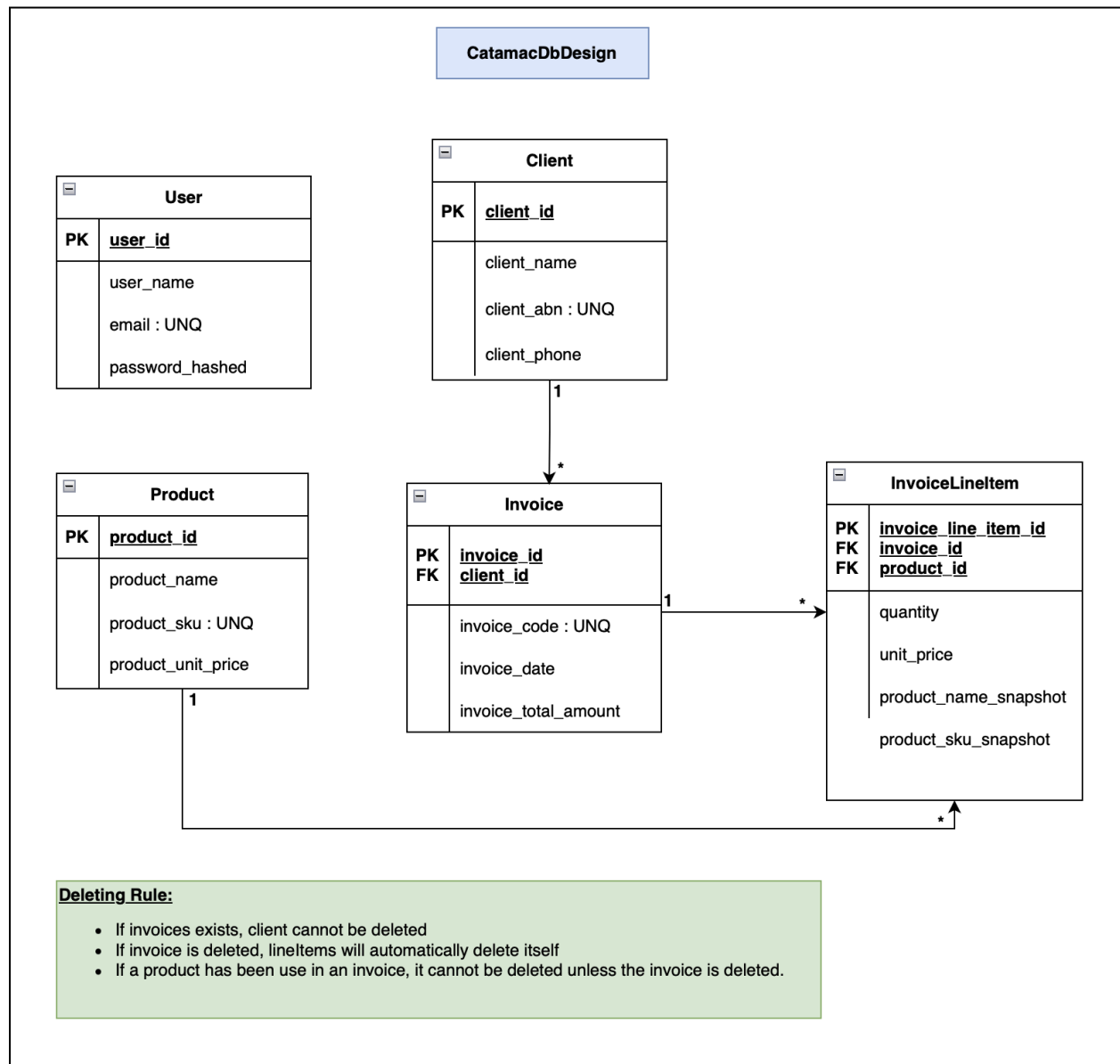
High-Level Architecture: Request Pipeline for Backend



Project Structure and Responsibility

- **Catamac.Domain**
 - Entities (user, client, product, invoice, lineItems) : Basically Data Model
- **Catamac.Application**
 - Data Transfer Objects : DTOs (Request/Response)
 - Interfaces
 - IUserRepository
 - IClientRepository
 - IProductRepository
 - IInvoiceRepository
 - IPasswordHasher
 - ITokenService
 - Service (Business Logics)
- **Catamac.Infrastructure**
 - EF Core DbContext + migration
 - Repository Implementation
 - Security Helper (JWT token generation & Password hashing)
- **CatamacInvoice.API**
 - Controller
 - Middleware - auth
 - Dependency Injection (DI)
- **Client**
 - Frontend - React App
- **Docs**
 - Important images, doc file, etc.

CatamacDbDesign



Details on Lineltem: Line Item basically refers to each product that is included in the invoice. One invoice can have multiple products, each with their quantity and price. Lineltem simply refers to one single line of product in an invoice.

API Endpoints

Auth

1. **POST** `/api/auth/register` : Create an admin user.
2. **POST** `/api/auth/login`: Authenticate and return JWT access token.
3. **GET** `/api/auth/me`: Return current logged-in user info.

Clients

1. **GET** `/api/clients`
2. **POST** `/api/clients`

Products

1. **GET** `/api/products`
2. **POST** `/api/products`

Invoice

1. **GET** `/api/invoices`
2. **GET** `/api/invoices/{invoiceId}`
3. **POST** `/api/invoices`

Core Use-Case

Walking through the main flow:

Create Invoice Flow:

- User sends invoice + lineItems
- Services validates
 - Client Exists
 - Product Exists
 - $Qt > 0$, price validation
- Compute Totals
- Save Invoice, save the line items
- Takes to the details page of the newly created Invoice

Fetch Invoice Details

- Return invoice header + lineItems
- Includes product snapshot details to preserve history