**Technical Design Specification- Customer Profile Management Application**

1. **Introduction**

The purpose of this document is to provide a comprehensive technical design specification for the Customer Profile Management Application. The application's primary function is to enable the management of customer profiles, allowing users to perform insert, update, and delete operations on customer information. A customer profile includes essential fields such as name, photo, email, phone, fax, internet, remarks, and supports the association of multiple addresses.

**2. System Architecture**

The application is a web-based system built on the ASP.NET Core framework. It follows a typical three-tier architecture, including a presentation layer, business logic layer, and data access layer. It is designed to be hosted on a web server.

**3. Data Design**

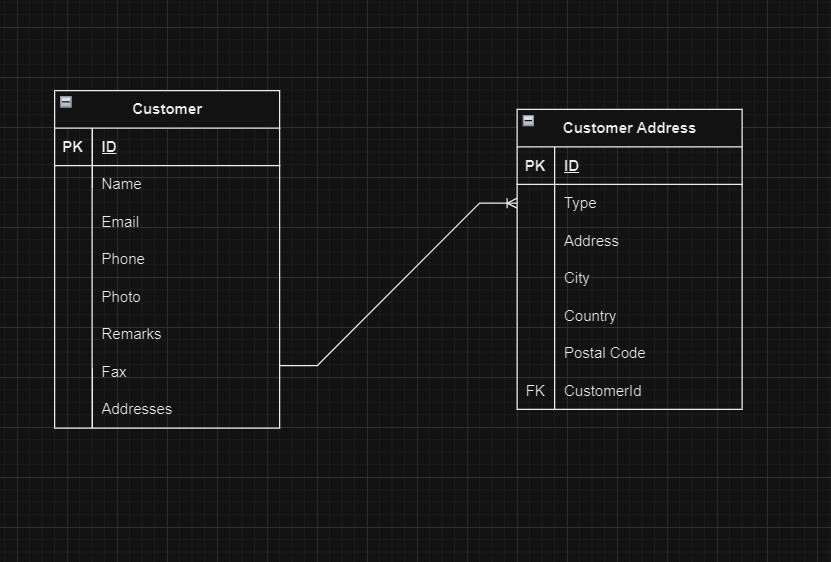
**3.1. Database Schema**

The application uses a relational database to store customer profiles and their associated addresses. The primary tables are:

Customers: Stores customer profile information, including fields for name, photo, email, phone, fax, internet, and remarks.

Addresses: Stores address information, including fields for address type, address, city, country, postal code, and a foreign key linking each address to a customer.

**3.2. Entity-Relationship Diagram (ERD)**



**4. User Interface (UI) Design**

The user interface is designed to be intuitive and user-friendly, allowing users to perform CRUD (Create, Read, Update, Delete) operations on customer profiles and their associated addresses. Key design elements include:

A customer profile form with fields for name, photo upload, email, phone, fax, internet, and remarks.

A section for managing customer addresses with fields for address type, address, city, country, and postal code.

Appropriate input validation and error handling.

1. **Component Design**

Customer Profile Component

Customer Repository: Handles data access for customer profiles.

1. **Security Design**

B*ecause of time limit, didn’t implement about security.If I have time, I will implement about this features.*

*User authentication and authorization will be implemented using [Authentication and Authorization System]. Access to customer profiles and address data will be restricted based on user roles and permissions*.

1. **Integration Points**

The application may integrate with external services or APIs for various purposes, such as authentication, sending emails. Integration details will be provided in the respective components.

**8. Testing Strategy**

Testing will include unit testing for repository. Test cases will cover different scenarios for inserting, updating, and deleting customer profiles.

**9. Deployment Plan**

The application will be deployed in multiple environments, including development, staging, and production. Deployment will be automated using deployment tools, and configurations will be managed through environment-specific configuration files.

**10. Dependencies**

The application relies on the following dependencies:

ASP.NET Core (version 6.0.23)

Entity Framework Core (version 6.0.23)

Serilog (version 3.0.1)

1. **References**

* OpenAI