# Proposal: ArtVault Management System

---

## 1. Introduction

The **ArtVault Management System** is an ASP.NET (Framework) web application designed to manage an art gallery's operations, including admin dashboard, artwork displaying, artist profiles, customer profiles, employee profiles, customer orders, event management, and employee assignments. This proposal outlines the system's architecture, database design, security measures, and implementation approach using **VB.NET** with **role-based authentication, data validation, and hashing** for security.

---

## 2. System Overview

### 2.1 Objectives

- Provide a centralized platform for managing artworks, artists, customers, orders, and events.
- Ensure data integrity with relational database constraints.
- Implement secure authentication and role-based access control.
- Enable efficient tracking of artwork statuses (Available, Sold, On Loan).
- Support event ticketing and sales management.

### 2.2 Technology Stack

- **Backend:** ASP.NET Framework (VB.NET)
- **Frontend:** ASP.NET Web Forms
- **Database:** SQL Server (Relational DB with proper constraints)

- **Security:**
  - Role-based authentication (Admin, Employee, Customer, Artist)
  - Password hashing (SHA-256)
  - Input validation (Server-side & Client-side)
  - SQL Injection prevention

---

# 3. Database Design

## 3.1 Key Tables & Relationships

| Table | Primary Key | Foreign Keys | Constraints |
| --- | --- | --- | --- |
| **Person** | PERSON_ID | - | Unique CNIC |
| **Customer** | CUSTOMER_ID | PERSON_ID (FK) | Customer cannot register as employee |
| **Employee** | EMPLOYEE_ID | PERSON_ID (FK) | Role assignment |
| **Artist** | ARTIST_ID | PERSON_ID (FK) | Cannot delete if artworks exist |
| **Artwork** | ARTWORK_ID | ARTIST_ID (FK) | Status: Available/Sold/On Loan |
| **Event** | EVENT_ID | - | - |
| **Ticket** | TICKET_ID | EVENT_ID (FK) | No deletion (Audit trail) |
| **Order** | ORDER_ID | CUSTOMER_ID (FK) | - |
| **OrderLine** | (ORDER_ID, ARTWORK_ID) | Composite FK | No duplicate artwork per order |

---

# 4. Security Implementation

## 4.1 Authentication & Authorization

- **Login System:**
  - Users (Employees, Customers, Artists) authenticate via email & password.
  - Passwords stored as **hashes (SHA-256)**.
- **Role-Based Access Control (RBAC):**
  - **Admin:** Full system access, Manage events, orders, artworks.
  - **Employee:** View assigned events and profile.
  - **Artist:** Update own artworks, view sales.
  - **Customer:** Place orders, view events.

## 4.2 Data Validation

- **Server-Side:**
  - VB.NET validation for forms.
  - Date validation.
- **Client-Side:**
  - ASP.NET Validator Controls.

---

# 5. Functional Modules

## 5.1 Artwork Management

- Add/Edit/View artworks (Admin/Artist).
- Status tracking (Available/Sold/On Loan).
- Price history.

## 5.2 Order Processing

- Customers place orders.

### 5.3 Event & Ticketing

- Create events with start/end dates.
- Sell tickets (prevent duplicate bookings).

### 5.4 Reporting & Analytics

- Sales reports (Monthly/Yearly).

---

# 8. Conclusion

The **ArtVault Management System** will streamline gallery operations while ensuring **security, scalability, and compliance** with business rules.