**📄 AccChiTietThanhToanTheoDon.cs (CS, 43 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class AccChiTietThanhToanTheoDon

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaThanhToan { get; set; }

11 │

12 │ public int? MaDonHang { get; set; }

13 │

14 │ public int? MaHoaDon { get; set; }

15 │

16 │ public decimal? PhaiThu { get; set; }

17 │

18 │ public decimal? DaThanhToan { get; set; }

19 │

20 │ public decimal? ConLai { get; set; }

21 │

22 │ public int? MaTienTeTt { get; set; }

23 │

24 │ public decimal? TyGiaTt { get; set; }

25 │

26 │ public int? UserCreated { get; set; }

27 │

28 │ public DateTime? CreatedDate { get; set; }

29 │

30 │ public int? UserModified { get; set; }

31 │

32 │ public DateTime? ModifiedDate { get; set; }

33 │

34 │ public bool? Deleted { get; set; }

35 │

36 │ public int? UserDeleted { get; set; }

37 │

38 │ public DateTime? DeletedDate { get; set; }

39 │

40 │ public bool? TuDong { get; set; }

41 │

42 │ public decimal? ChenhLechTyGia { get; set; }

43 │ }

────────────────────────────────────────────────────────────

**📄 AccChuyenHinhThucTamUng.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class AccChuyenHinhThucTamUng

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaDatCoc { get; set; }

13 │

14 │ public DateTime? NgayChuyen { get; set; }

15 │

16 │ public string? LyDo { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 AccCongNoDauKy.cs (CS, 59 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class AccCongNoDauKy

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaKh { get; set; }

11 │

12 │ public string? TenKh { get; set; }

13 │

14 │ public int? MaDonHang { get; set; }

15 │

16 │ public string? SoDonHang { get; set; }

17 │

18 │ public int? MaHoaDon { get; set; }

19 │

20 │ public string? SoHoaDon { get; set; }

21 │

22 │ public DateTime? NgayHoaDon { get; set; }

23 │

24 │ public DateTime? NgayHetHanTt { get; set; }

25 │

26 │ public int? MaTienTe { get; set; }

27 │

28 │ public int? TyGiaHoaDon { get; set; }

29 │

30 │ public decimal? SoDuNo { get; set; }

31 │

32 │ public int? TyGiaTt { get; set; }

33 │

34 │ public decimal? DaThanhToan { get; set; }

35 │

36 │ public DateTime? NgayThanhToan { get; set; }

37 │

38 │ public string? NguoiThanhToan { get; set; }

39 │

40 │ public int? ThoiHanTt { get; set; }

41 │

42 │ public int? SoNgayNo { get; set; }

43 │

44 │ public string? GhiChu { get; set; }

45 │

46 │ public int? UserCreated { get; set; }

47 │

48 │ public DateTime? CreatedDate { get; set; }

49 │

50 │ public int? UserModified { get; set; }

51 │

52 │ public DateTime? ModifiedDate { get; set; }

53 │

54 │ public bool? Deleted { get; set; }

55 │

56 │ public int? UserDeleted { get; set; }

57 │

58 │ public DateTime? DeletedDate { get; set; }

59 │ }

────────────────────────────────────────────────────────────

**📄 AccHinhThucThu.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class AccHinhThucThu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public bool? TrangThai { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 AccLoaiPhieuThu.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class AccLoaiPhieuThu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public bool? PhieuThanhToan { get; set; }

15 │

16 │ public bool? TrangThai { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 AccPhatSinhCongNo.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class AccPhatSinhCongNo

7 │ {

8 │ public long Ma { get; set; }

9 │

10 │ public DateTime? NgayPhatSinh { get; set; }

11 │

12 │ public string? Code { get; set; }

13 │

14 │ public int? MaKh { get; set; }

15 │

16 │ public string? TenKh { get; set; }

17 │

18 │ public decimal? NoDauKy { get; set; }

19 │

20 │ public decimal? CoDauKy { get; set; }

21 │

22 │ public decimal? NoTrongKy { get; set; }

23 │

24 │ public decimal? CoTrongKy { get; set; }

25 │

26 │ public decimal? NoCuoiKy { get; set; }

27 │

28 │ public decimal? CoCuoiKy { get; set; }

29 │

30 │ public int? UserDeleted { get; set; }

31 │

32 │ public DateTime? DeletedDate { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 AccSoDu131.cs (CS, 21 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class AccSoDu131

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaKh { get; set; }

11 │

12 │ public string? NgayPhatSinh { get; set; }

13 │

14 │ public decimal? NoCuoiKyNt { get; set; }

15 │

16 │ public decimal? CoCuoiKyNt { get; set; }

17 │

18 │ public decimal? NoCuoiKy { get; set; }

19 │

20 │ public decimal? CoCuoiKy { get; set; }

21 │ }

────────────────────────────────────────────────────────────

**📄 AccSoDu131ChietKhau.cs (CS, 21 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class AccSoDu131ChietKhau

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public DateOnly? Ngay { get; set; }

11 │

12 │ public int? MaKh { get; set; }

13 │

14 │ public decimal? NoDauKyNt { get; set; }

15 │

16 │ public decimal? NoDauKy { get; set; }

17 │

18 │ public decimal? CoDauKyNt { get; set; }

19 │

20 │ public decimal? CoDauKy { get; set; }

21 │ }

────────────────────────────────────────────────────────────

**📄 AccTaiKhoanNh.cs (CS, 37 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class AccTaiKhoanNh

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? TenNganHang { get; set; }

11 │

12 │ public string? DiaChi { get; set; }

13 │

14 │ public string? SoTaiKhoan { get; set; }

15 │

16 │ public int? TienTe { get; set; }

17 │

18 │ public string? TenChuTaiKhoan { get; set; }

19 │

20 │ public string? GhiChu { get; set; }

21 │

22 │ public int? UserCreated { get; set; }

23 │

24 │ public DateTime? CreatedDate { get; set; }

25 │

26 │ public int? UserModified { get; set; }

27 │

28 │ public DateTime? ModifiedDate { get; set; }

29 │

30 │ public bool? Deleted { get; set; }

31 │

32 │ public int? UserDeleted { get; set; }

33 │

34 │ public DateTime? DeletedDate { get; set; }

35 │

36 │ public int? MaChiNhanh { get; set; }

37 │ }

────────────────────────────────────────────────────────────

**📄 AccTamUng.cs (CS, 59 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class AccTamUng

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaLoaiPhieu { get; set; }

13 │

14 │ public string? SoPhieu { get; set; }

15 │

16 │ public int? MaKh { get; set; }

17 │

18 │ public string? NguoiNop { get; set; }

19 │

20 │ public DateTime? NgayChungTu { get; set; }

21 │

22 │ public int? HinhThucThu { get; set; }

23 │

24 │ public string? LyDo { get; set; }

25 │

26 │ public decimal? SoTien { get; set; }

27 │

28 │ public decimal? DaThanhToan { get; set; }

29 │

30 │ public decimal? ConLai { get; set; }

31 │

32 │ public int? TienTe { get; set; }

33 │

34 │ public int? TyGia { get; set; }

35 │

36 │ public int? MaChuyenTamUng { get; set; }

37 │

38 │ public int? MaDatCoc { get; set; }

39 │

40 │ public int? MaLenhXuatKho { get; set; }

41 │

42 │ public int? MaTaiKhoanNh { get; set; }

43 │

44 │ public int? TrangThai { get; set; }

45 │

46 │ public int? UserCreated { get; set; }

47 │

48 │ public DateTime? CreatedDate { get; set; }

49 │

50 │ public int? UserModified { get; set; }

51 │

52 │ public DateTime? ModifiedDate { get; set; }

53 │

54 │ public bool? Deleted { get; set; }

55 │

56 │ public int? UserDeleted { get; set; }

57 │

58 │ public DateTime? DeletedDate { get; set; }

59 │ }

────────────────────────────────────────────────────────────

**📄 AccThanhToanTheoDon.cs (CS, 59 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class AccThanhToanTheoDon

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaLoaiPhieu { get; set; }

13 │

14 │ public int? HinhThucTt { get; set; }

15 │

16 │ public int? MaTamUng { get; set; }

17 │

18 │ public string? SoPhieu { get; set; }

19 │

20 │ public int? MaKh { get; set; }

21 │

22 │ public DateTime? NgayChungTu { get; set; }

23 │

24 │ public string? NguoiNopTien { get; set; }

25 │

26 │ public string? LyDo { get; set; }

27 │

28 │ public int? TienTe { get; set; }

29 │

30 │ public int? TyGia { get; set; }

31 │

32 │ public decimal? SoTienThanhToan { get; set; }

33 │

34 │ public int? TrangThai { get; set; }

35 │

36 │ public DateTime? NgayChuyenSoKt { get; set; }

37 │

38 │ public int? MaTaiKhoanNh { get; set; }

39 │

40 │ public string? TruongSo1 { get; set; }

41 │

42 │ public string? TruongSo2 { get; set; }

43 │

44 │ public int? UserCreated { get; set; }

45 │

46 │ public DateTime? CreatedDate { get; set; }

47 │

48 │ public int? UserModified { get; set; }

49 │

50 │ public DateTime? ModifiedDate { get; set; }

51 │

52 │ public bool? Deleted { get; set; }

53 │

54 │ public int? UserDeleted { get; set; }

55 │

56 │ public DateTime? DeletedDate { get; set; }

57 │

58 │ public int? KieuPhanBo { get; set; }

59 │ }

────────────────────────────────────────────────────────────

**📄 AccTrangThaiTamUng.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class AccTrangThaiTamUng

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 AccXuLyChenhLechTyGia.cs (CS, 41 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class AccXuLyChenhLechTyGia

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaHoaDon { get; set; }

11 │

12 │ public int? TienTe { get; set; }

13 │

14 │ public int? TyGiaDonHang { get; set; }

15 │

16 │ public int? TyGiaThanhToan { get; set; }

17 │

18 │ public decimal? PhaiTraUsddonHang { get; set; }

19 │

20 │ public decimal? PhaiTraVnddonHang { get; set; }

21 │

22 │ public decimal? ThanhToanUsddonHang { get; set; }

23 │

24 │ public decimal? ThanhToanVnddonHang { get; set; }

25 │

26 │ public decimal? SoTienXuLyChenhLechVnd { get; set; }

27 │

28 │ public int? UserCreated { get; set; }

29 │

30 │ public DateTime? CreatedDate { get; set; }

31 │

32 │ public int? UserModified { get; set; }

33 │

34 │ public DateTime? ModifiedDate { get; set; }

35 │

36 │ public bool? Deleted { get; set; }

37 │

38 │ public int? UserDeleted { get; set; }

39 │

40 │ public DateTime? DeletedDate { get; set; }

41 │ }

────────────────────────────────────────────────────────────

**📄 AccountController.cs (CS, 156 lines)**

1 │ ﻿using Microsoft.AspNetCore.Authentication;

2 │ using Microsoft.AspNetCore.Authentication.Cookies;

3 │ using Microsoft.AspNetCore.Mvc;

4 │ using Microsoft.Extensions.Logging;

5 │ using Microsoft.IdentityModel.Tokens;

6 │ using System.IdentityModel.Tokens.Jwt;

7 │ using System.Security.Claims;

8 │ using System.Text;

9 │ using System.Threading.Tasks;

10 │ using Microsoft.EntityFrameworkCore;

11 │ using QuanLyChatLuong.Models;

12 │ using System.Security.Cryptography;

13 │ using QuanLyChatLuong.Common.Attributes;

14 │

15 │ namespace QuanLyChatLuong.Controllers

16 │ {

17 │ [Route("account")]

18 │ public class AccountController : Controller

19 │ {

20 │ private readonly ILogger<AccountController> \_logger;

21 │ private readonly QuanLyChatLuongDbContext \_context;

22 │

23 │ public AccountController(ILogger<AccountController> logger, QuanLyChatLuongDbContext context)

24 │ {

25 │ \_logger = logger;

26 │ \_context = context;

27 │ }

28 │

29 │ [HttpGet]

30 │ public IActionResult Login()

31 │ {

32 │ var token = Request.Cookies["Token"];

33 │

34 │ if (string.IsNullOrEmpty(token))

35 │ {

36 │ var branches = \_context.tbl\_ChiNhanhs.ToList();

37 │ return View(branches);

38 │ }

39 │

40 │ return RedirectToAction("Index", "Home");

41 │ }

42 │

43 │ [HttpPost]

44 │ public async Task<IActionResult> LoginAPI(string userName, string password, int branch)

45 │ {

46 │ if (string.IsNullOrEmpty(userName))

47 │ {

48 │ return Json(new { success = false, message = "Tên đăng nhập không được để trống" });

49 │ }

50 │

51 │ var existingUser = await \_context.sys\_Users.FirstOrDefaultAsync(x => x.TenDangNhap == userName);

52 │ if (existingUser == null)

53 │ {

54 │ return Json(new { success = false, message = "Tên đăng nhập không tồn tại" });

55 │ }

56 │

57 │ if (string.IsNullOrEmpty(password))

58 │ {

59 │ return Json(new { success = false, message = "Mật khẩu không được để trống" });

60 │ }

61 │

62 │ bool isPasswordValid = VerifyPasswordWithSHA512(password, existingUser.Password);

63 │ if (!isPasswordValid)

64 │ {

65 │ return Json(new { success = false, message = "Mật khẩu không đúng" });

66 │ }

67 │

68 │ // Kiểm tra mã chi nhánh

69 │ //var userBranch = await \_context.sys\_UserRoles

70 │ // .FirstOrDefaultAsync(x => x.MaUser == existingUser.Ma && x.MaChiNhanh == branch);

71 │

72 │ //if (userBranch == null)

73 │ //{

74 │ // return Json(new { success = false, message = "Người dùng không có quyền truy cập vào chi nh...

75 │ //}

76 │

77 │ var claims = new[] {

78 │ new Claim(ClaimTypes.Name, existingUser.HoTen),

79 │ new Claim(ClaimTypes.NameIdentifier, existingUser.Ma.ToString()),

80 │ new Claim("Branch", branch.ToString()) // Thêm chi nhánh vào claim

81 │ };

82 │

83 │ var key = new SymmetricSecurityKey(Encoding.UTF8.GetBytes("your-very-secure-secret-key1234567"));

84 │ var creds = new SigningCredentials(key, SecurityAlgorithms.HmacSha256);

85 │

86 │ var claimsIdentity = new ClaimsIdentity(claims, CookieAuthenticationDefaults.AuthenticationScheme);

87 │

88 │ await HttpContext.SignInAsync(

89 │ CookieAuthenticationDefaults.AuthenticationScheme,

90 │ new ClaimsPrincipal(claimsIdentity),

91 │ new AuthenticationProperties

92 │ {

93 │ IsPersistent = true,

94 │ ExpiresUtc = DateTimeOffset.UtcNow.AddMinutes(30)

95 │ });

96 │

97 │ var token = new JwtSecurityToken(

98 │ issuer: "your-issuer",

99 │ audience: "your-audience",

100 │ claims: claims,

101 │ expires: DateTime.Now.AddHours(1),

102 │ signingCredentials: creds

103 │ );

104 │

105 │ var tokenHandler = new JwtSecurityTokenHandler();

106 │ var jwtToken = tokenHandler.WriteToken(token);

107 │

108 │ Response.Cookies.Append("AuthToken", jwtToken, new Microsoft.AspNetCore.Http.CookieOptions

109 │ {

110 │ SameSite = Microsoft.AspNetCore.Http.SameSiteMode.Lax,

111 │ Expires = DateTime.Now.AddHours(1),

112 │ Path = "/",

113 │ });

114 │

115 │ return Ok(new

116 │ {

117 │ success = true,

118 │ message = "Đăng nhập thành công",

119 │ token = jwtToken

120 │ });

121 │ }

122 │

123 │

124 │

125 │ [HttpGet]

126 │ public IActionResult AccessDenied()

127 │ {

128 │ return View();

129 │ }

130 │ [TokenUserCheckHTTP]

131 │ [HttpPost("logout")]

132 │ public IActionResult Logout()

133 │ {

134 │ Response.Cookies.Delete("AuthToken");

135 │

136 │ return Ok(new { success = true });

137 │ }

138 │

139 │ private bool VerifyPasswordWithSHA512(string enteredPassword, string storedHash)

140 │ {

141 │ string enteredPasswordHash = HashPasswordWithSHA512(enteredPassword);

142 │

143 │ return enteredPasswordHash == storedHash;

144 │ }

145 │ private string HashPasswordWithSHA512(string password)

146 │ {

147 │ using (SHA512 shaM = SHA512.Create())

148 │ {

149 │ byte[] bytes = Encoding.UTF8.GetBytes(password);

150 │ byte[] hash = shaM.ComputeHash(bytes);

151 │ return BitConverter.ToString(hash).Replace("-", "").ToUpper();

152 │ }

153 │ }

154 │

155 │ }

156 │ }

────────────────────────────────────────────────────────────

**📄 AuthController.cs (CS, 73 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using Microsoft.IdentityModel.Tokens;

3 │ using QuanLyChatLuong.Models;

4 │ using System;

5 │ using System.IdentityModel.Tokens.Jwt;

6 │ using System.Linq;

7 │ using System.Security.Claims;

8 │ using System.Text;

9 │

10 │ namespace QuanLyChatLuong.Controllers

11 │ {

12 │ public class AuthController : Controller

13 │ {

14 │ private readonly string \_jwtSecretKey = "your-very-secure-secret-key1234567";

15 │

16 │ private readonly QuanLyChatLuongDbContext \_context;

17 │

18 │ public AuthController(QuanLyChatLuongDbContext context)

19 │ {

20 │ \_context = context;

21 │ }

22 │

23 │ private ClaimsPrincipal GetUserFromToken(string token)

24 │ {

25 │ var tokenHandler = new JwtSecurityTokenHandler();

26 │ var key = Encoding.UTF8.GetBytes(\_jwtSecretKey);

27 │

28 │ try

29 │ {

30 │ var principal = tokenHandler.ValidateToken(token, new TokenValidationParameters

31 │ {

32 │ ValidateIssuer = true,

33 │ ValidateAudience = true,

34 │ ValidateLifetime = true,

35 │ ValidIssuer = "your-issuer",

36 │ ValidAudience = "your-audience",

37 │ IssuerSigningKey = new SymmetricSecurityKey(key)

38 │ }, out SecurityToken validatedToken);

39 │

40 │ return principal;

41 │ }

42 │ catch (Exception)

43 │ {

44 │ return null;

45 │ }

46 │ }

47 │

48 │ public IActionResult GetUserInfo(string token)

49 │ {

50 │ var userClaims = GetUserFromToken(token);

51 │ if (userClaims == null)

52 │ {

53 │ return Json(new { success = false, message = "Token không hợp lệ" });

54 │ }

55 │

56 │ var userName = userClaims.FindFirst(ClaimTypes.Name)?.Value;

57 │ var userId = userClaims.FindFirst(ClaimTypes.NameIdentifier)?.Value;

58 │

59 │ var user = \_context.sys\_Users.FirstOrDefault(u => u.HoTen == userName);

60 │ if (user == null)

61 │ {

62 │ return Json(new { success = false, message = "Người dùng không tồn tại" });

63 │ }

64 │

65 │ return Json(new

66 │ {

67 │ success = true,

68 │ userName = userName,

69 │

70 │ });

71 │ }

72 │ }

73 │ }

────────────────────────────────────────────────────────────

**📄 BaoBiController.cs (CS, 169 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using Microsoft.EntityFrameworkCore;

3 │ using QuanLyChatLuong.Common;

4 │ using QuanLyChatLuong.Models;

5 │ using QuanLyChatLuong.Models.APIModels;

6 │

7 │ namespace QuanLyChatLuong.Controllers

8 │ {

9 │ [Route("baobi")]

10 │ public class BaoBiController : BaseController

11 │ {

12 │ public BaoBiController(QuanLyChatLuongDbContext dbcontext, MemoryCacheManager memoryCache) : base(dbcontext, mem...

13 │ {

14 │ m\_MenuID = MenuID.baobi;

15 │ }

16 │

17 │ private ResponseModel CheckRequire(string VatLieuBaoBi)

18 │ {

19 │ if (string.IsNullOrEmpty(VatLieuBaoBi))

20 │ {

21 │ return new ResponseModel(QLCLConst.Error, "Chưa nhập vật liệu bao bì");

22 │ }

23 │ return null;

24 │ }

25 │

26 │ [TokenUserCheckHTTP]

27 │ [HttpGet]

28 │ public IActionResult Index()

29 │ {

30 │

31 │ return View();

32 │ }

33 │

34 │ [TokenUserCheckHTTP(Permission.Add)]

35 │ [HttpGet("add")]

36 │ public IActionResult Add()

37 │ {

38 │

39 │ return View("Form");

40 │ }

41 │

42 │ [TokenUserCheckHTTP()]

43 │ [HttpPost("addAPI")]

44 │ public IActionResult AddAPI(int Ma, string Ten, string VatLieuBaoBi, string TrangThai)

45 │ {

46 │ var result = CheckRequire(VatLieuBaoBi);

47 │

48 │ if (result != null)

49 │ {

50 │ return Json(result);

51 │ }

52 │

53 │ if (dbcontext.bt\_BaoBis.Any(x => x.VatLieuBaoBi == VatLieuBaoBi && x.Ma != Ma))

54 │ {

55 │ return Json(new ResponseModel(QLCLConst.Error, "Vật liệu bao bì đã tồn tại"));

56 │ }

57 │

58 │ var item = new bt\_BaoBi

59 │ {

60 │ Ten = Cast.ToInt(Ten),

61 │ VatLieuBaoBi = VatLieuBaoBi,

62 │ TrangThai = TrangThai == "1",

63 │ NguoiTao = objToken!.userid,

64 │ NgayTao = DateTime.Now,

65 │ MaChiNhanh = objToken.branch

66 │ };

67 │

68 │

69 │ dbcontext.bt\_BaoBis.Add(item);

70 │ dbcontext.SaveChanges();

71 │

72 │ return Json(new ResponseModel(QLCLConst.Successfully, "Thêm thành công"));

73 │ }

74 │

75 │ [TokenUserCheckHTTP(Permission.Edit)]

76 │ [HttpGet("edit")]

77 │ public IActionResult Edit(string id)

78 │ {

79 │ var item = dbcontext.bt\_BaoBis.FirstOrDefault(c => c.Ma == Cast.ToInt(id.Decrypt()));

80 │

81 │ if (item == null)

82 │ {

83 │ return NotFound("Bao bi not found");

84 │ }

85 │

86 │ ViewData["item"] = item;

87 │ return View("Form");

88 │ }

89 │

90 │ [TokenUserCheckHTTP()]

91 │ [HttpPost("updateAPI")]

92 │ public IActionResult UpdateAPI(int Ma, string Ten, string VatLieuBaoBi, string TrangThai)

93 │ {

94 │ var item = dbcontext.bt\_BaoBis.FirstOrDefault(x => x.Ma == Ma && !(x.DaXoa ?? false));

95 │ if (item == null)

96 │ {

97 │ return Json(new ResponseModel(QLCLConst.Error, "Không tồn tại"));

98 │ }

99 │ if (item.VatLieuBaoBi != VatLieuBaoBi && dbcontext.bt\_BaoBis.Any(x => x.VatLieuBaoBi == VatLieuBaoBi && x.Ma...

100 │ {

101 │ return Json(new ResponseModel(QLCLConst.Error, "Vật liệu bao bì đã tồn tại"));

102 │ }

103 │

104 │ item.Ten = Cast.ToInt(Ten);

105 │ item.VatLieuBaoBi = VatLieuBaoBi;

106 │ item.TrangThai = TrangThai == "1";

107 │ item.NguoiSua = objToken!.userid;

108 │ item.NgaySua = DateTime.Now;

109 │ item.MaChiNhanh = objToken.branch;

110 │ try

111 │ {

112 │ dbcontext.bt\_BaoBis.Update(item);

113 │ dbcontext.SaveChanges();

114 │ }

115 │ catch (Exception ex)

116 │ {

117 │ return Json(new ResponseModel(QLCLConst.Error, "Cập nhật thất bại: " + ex.Message));

118 │ }

119 │

120 │ return Json(new ResponseModel(QLCLConst.Successfully, "Cập nhật thành công"));

121 │ }

122 │

123 │ [HttpPost("DeleteAPI")]

124 │ [TokenUserCheckHTTP(Permission.Delete)]

125 │ public IActionResult DeleteAPI(string Ma)

126 │ {

127 │ var item = dbcontext.bt\_BaoBis.FirstOrDefault(x => x.Ma == Cast.ToInt(Ma.Decrypt()));

128 │ if (item == null)

129 │ {

130 │ return Json(QLCLConst.ResponseNotExist);

131 │ }

132 │ item.DaXoa = true;

133 │ dbcontext.SaveChanges();

134 │ return Json(QLCLConst.ResponseDeleteSuccessfully);

135 │ }

136 │

137 │ [TokenUserCheckHTTP(Permission.View)]

138 │ [HttpGet("getListBaoBi")]

139 │ public IActionResult getListBaoBi()

140 │ {

141 │ var listData = dbcontext.bt\_BaoBis.Where(x => !(x.DaXoa ?? false) && x.MaChiNhanh == objToken.branch).OrderB...

142 │ .Select(t => new

143 │ {

144 │ Ma = t.Ma,

145 │ eMa = t.Ma.Encrypt(),

146 │ Ten = t.Ten,

147 │ VatLieuBaoBi = t.VatLieuBaoBi,

148 │ TrangThai = t.TrangThai,

149 │

150 │ }).ToList();

151 │ return Json(new ResponseModel(QLCLConst.Successfully, "", listData));

152 │ }

153 │ [TokenUserCheckHTTP(Permission.View)]

154 │ [HttpPost("getListData")]

155 │ public IActionResult getListData()

156 │ {

157 │ var listData = (from bb in dbcontext.bt\_BaoBis.Where(x => !(x.DaXoa ?? false) && x.MaChiNhanh == objToken.b...

158 │ select new

159 │ {

160 │ ma = bb.Ma,

161 │ eMa = bb.Ma.Encrypt(),

162 │ ten = bb.VatLieuBaoBi,

163 │ kichThuoc = bb.Ten

164 │ }

165 │ ).ToList();

166 │ return Json(new ResponseModelSuccess(QLCLConst.Successfully, listData));

167 │ }

168 │ }

169 │ }

────────────────────────────────────────────────────────────

**📄 BaoCaoBotBienTinhController.cs (CS, 543 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using Microsoft.EntityFrameworkCore;

3 │ using Microsoft.IdentityModel.Tokens;

4 │ using QuanLyChatLuong.Common;

5 │ using QuanLyChatLuong.Models;

6 │ using QuanLyChatLuong.Models.APIModels;

7 │ using QuanLyChatLuong.Models.CRMEntity;

8 │ using QuanLyChatLuong.Models.StoreProcedures;

9 │ using Spire.Xls;

10 │ using System;

11 │ using System.Data;

12 │ using System.Globalization;

13 │ using static System.Net.WebRequestMethods;

14 │ using Utility = QuanLyChatLuong.Common.Utility;

15 │

16 │ namespace QuanLyChatLuong.Controllers

17 │ {

18 │ [Route("catSoLieuBienTinh")]

19 │ public class BaoCaoBotBienTinhController : BaseController

20 │ {

21 │ private readonly IWebHostEnvironment \_hostingEnvironment;

22 │ private CrmContext crmContext { get; }

23 │

24 │ public BaoCaoBotBienTinhController(QuanLyChatLuongDbContext dbcontext, CrmContext crmdbcontext, IWebHostEnvironm...

25 │ {

26 │ m\_MenuID = MenuID.catSoLieuBienTinh;

27 │ \_hostingEnvironment = hostingEnvironment;

28 │ crmContext = crmdbcontext;

29 │ }

30 │ public class BaoCaoForm

31 │ {

32 │ public string? MaChiNhanh { get; set; }

33 │ public string? MaChiNhanh2 { get; set; }

34 │ public string? Ngay { get; set; } = DateTime.Now.ToString();

35 │ public string? TuNgay { get; set; } = DateTime.Now.ToString();

36 │ public string? DenNgay { get; set; } = DateTime.Now.ToString();

37 │ public string? FileType { get; set; }

38 │ public string? BaoCao { get; set; }

39 │ }

40 │ public class ColorGioiHanDto

41 │ {

42 │ public string MinColor { get; set; }

43 │ public string MaxColor { get; set; }

44 │ public string Code { get; set; }

45 │ public int MaCN { get; set; }

46 │ public string Loai { get; set; }

47 │ public decimal GiaTri { get; set; }

48 │ public string Ten { get; set; }

49 │ }

50 │ [HttpGet]

51 │ [TokenUserCheckHTTP]

52 │ public IActionResult Index()

53 │ {

54 │ ViewData["CurrentUser"] = dbcontext.sys\_Users.FirstOrDefault(x => x.Ma == objToken.userid);

55 │ ViewData["CurrentBranch"] = dbcontext.tbl\_ChiNhanhs.FirstOrDefault(x => x.Ma == objToken.branch);

56 │ return View();

57 │ }

58 │ [HttpPost("BaocaoBotBienTinhTheoNgay")]

59 │ [TokenUserCheckHTTP]

60 │ public IActionResult BaocaoBotBienTinhTheoNgay([FromBody] BaoCaoForm model)

61 │ {

62 │

63 │ var data = dbcontext.Set<sp\_BaoCaoSoLieuBienTinhTheoNgayResult>().FromSqlRaw($"spBaoCaoSoLieuBienTinhTuNgayD...

64 │ int current = (int)objToken.branch;

65 │ string codeChiNhanh = dbcontext.tbl\_ChiNhanhs.FirstOrDefault(x => x.Ma == current).Code;

66 │ var listLoThuMua = (from l in crmContext.CloseChotLo.Where(t => !(t.Deleted ?? false) && (t.TrangThai != (in...

67 │ join cn in crmContext.SysBranch.Where(t => t.Code == codeChiNhanh && !(t.Deleted ?? fals...

68 │ orderby l.CreatedDate

69 │ select new

70 │ {

71 │ malo = l.Ma,

72 │ codelo = l.MaLo + "",

73 │ }).ToList();

74 │

75 │ return Json(new ResponseModel(QLCLConst.OK, "", new

76 │ {

77 │ ListData = data,

78 │ ListLo = listLoThuMua

79 │ }));

80 │ }

81 │ [HttpGet("ExportBaocao")]

82 │ [TokenUserCheckHTTP(Permission.NotCheck)]

83 │ public IActionResult ExportBaocaoBotBienTinhTheoNgay(string BaoCao, string Ngay, string TuNgay, string DenNgay, ...

84 │ {

85 │ int DongBatDau = 0;

86 │ DataTable dt = new DataTable();

87 │ List<ExcelValue> ListGiaTriRiengLe = new List<ExcelValue>();

88 │ string fileTemplate = "", contentType = "", fileName = "", cotKT = "T";

89 │ DateTime dtpTuNgay = DateTime.TryParse(TuNgay, out DateTime tempTuNgay) ? tempTuNgay : DateTime.Now;

90 │ DateTime dtpDenNgay = DateTime.TryParse(DenNgay, out DateTime tempDenNgay) ? tempDenNgay : DateTime.Now;

91 │

92 │ int branch = Cast.ToInt(objToken.branch);

93 │ int maBotBT = Cast.ToInt(MaBotBienTinh);

94 │ int maBotBienTinhBTP = Cast.ToInt(MaBotBienTinhBTP);

95 │

96 │ if (MaChiNhanh != null || !MaChiNhanh.IsEmpty())

97 │ {

98 │ branch = Cast.ToInt(MaChiNhanh);

99 │ }

100 │

101 │ switch (BaoCao)

102 │ {

103 │ case "baocaoBotBienTinhTheoNgay":

104 │ var data = (from sl in dbcontext.bt\_SoLieuBienTinhs

105 │ join ctsl in dbcontext.bt\_ChiTietSoLieus on sl.Ma equals ctsl.MaSoLieu

106 │ join cn in dbcontext.tbl\_ChiNhanhs on sl.MaChiNhanh equals cn.Ma

107 │ where (sl.Ngay != null && (sl.Ngay.Value.Date >= dtpTuNgay.Date && sl.Ngay.Value.Date <=...

108 │ && sl.MaChiNhanh == branch

109 │ && !(ctsl.DaXoa ?? false)

110 │ && !(sl.DaXoa ?? false)

111 │ && sl.Ma == maBotBT

112 │ select new

113 │ {

114 │ Ngay = ctsl.NgayTao,

115 │ Gio = ctsl.Gio,

116 │ Phut = ctsl.Phut,

117 │ Whiteness = ctsl.Whiteness,

118 │ Moisture = ctsl.Moisture,

119 │ Fineness = ctsl.Fineness,

120 │ pHValue = ctsl.pHValue,

121 │ Residue = ctsl.Residue,

122 │ ViscosityPeakBU = ctsl.ViscosityPeakBU,

123 │ ViscosityMax950C0MinBU = ctsl.ViscosityMax950C0MinBU,

124 │ ViscosityMax950C10MinBU = ctsl.ViscosityMax950C10MinBU,

125 │ GelatinnizationTemperature = ctsl.GelatinnizationTemperature,

126 │ AcetylGroup = ctsl.AcetylGroup,

127 │ DS = ctsl.DS,

128 │ Conductivity = ctsl.Conductivity,

129 │ Note = ctsl.Note,

130 │ ViscosityEndPointBU = ctsl.ViscosityEndPointBU,

131 │ ForeignMatter = ctsl.ForeignMatter,

132 │ SO2 = ctsl.SO2,

133 │ StarchContent = ctsl.StarchContent,

134 │ Sensory = ctsl.Sensory,

135 │ DonHang = sl.DonHang,

136 │ TenSanPham = sl.TenSanPham,

137 │ MaLo = sl.MaLo

138 │ }

139 │ ).ToList();

140 │ if (!FileType.IsNullOrEmpty())

141 │ {

142 │ dt.Columns.Add("DATE");

143 │ dt.Columns.Add("HOUR", typeof(int));

144 │ dt.Columns.Add("MINUTE", typeof(int));

145 │ dt.Columns.Add("WHITENESS (%) ", typeof(decimal));

146 │ dt.Columns.Add("MOISTURE (%) ", typeof(decimal));

147 │ dt.Columns.Add("FINENESS (%) ", typeof(decimal));

148 │ dt.Columns.Add("PH", typeof(decimal));

149 │ dt.Columns.Add("RESIDUDE (%) ", typeof(decimal));

150 │ dt.Columns.Add("VISCOSITY PEAK", typeof(decimal));

151 │ dt.Columns.Add("VISCOSITY (START OF HOLDING)", typeof(decimal));

152 │ dt.Columns.Add("VISCOSITY ( START OF COOLING)", typeof(decimal));

153 │ dt.Columns.Add("VISCOSITY ( END OF FINAL)", typeof(decimal));

154 │ dt.Columns.Add("Gelatinization Temperature", typeof(decimal));

155 │ dt.Columns.Add("SO2 (mg/kg) ", typeof(decimal));

156 │ dt.Columns.Add("Acetyl Group %", typeof(decimal));

157 │ dt.Columns.Add("DS", typeof(decimal));

158 │ dt.Columns.Add("Conductivity (mS/Cm)", typeof(decimal));

159 │ dt.Columns.Add("STARCH CONTENT (%)", typeof(decimal));

160 │ dt.Columns.Add("FOREIGN MATTER");

161 │ dt.Columns.Add("SENSORY");

162 │ dt.Columns.Add("Note");

163 │

164 │ foreach (var item in data)

165 │ {

166 │ DateTime? datee = item.Ngay;

167 │ datee ??= DateTime.Now;

168 │ string? formattedDate = datee.Value.ToString("dd/MM/yyyy");

169 │

170 │ dt.Rows.Add(new object[] {

171 │ formattedDate,

172 │ Cast.ToInt(item.Gio),

173 │ Cast.ToInt(item.Phut),

174 │ Cast.ToDouble(item.Whiteness) != 0 ? Cast.ToDouble(item.Whiteness) : DBNull.Value,

175 │ Cast.ToDouble(item.Moisture) != 0 ? Cast.ToDouble(item.Moisture) : DBNull.Value,

176 │ Cast.ToDouble(item.Fineness) != 0 ? Cast.ToDouble(item.Fineness) : DBNull.Value,

177 │ Cast.ToDouble(item.pHValue) != 0 ? Cast.ToDouble(item.pHValue) : DBNull.Value,

178 │ Cast.ToDouble(item.Residue) != 0 ? Cast.ToDouble(item.Residue) : DBNull.Value,

179 │ Cast.ToDouble(item.ViscosityPeakBU) != 0 ? Cast.ToDouble(item.ViscosityPeakBU) : DBNull....

180 │ Cast.ToDouble(item.ViscosityMax950C0MinBU) != 0 ? Cast.ToDouble(item.ViscosityMax950C0Mi...

181 │ Cast.ToDouble(item.ViscosityMax950C10MinBU) != 0 ? Cast.ToDouble(item.ViscosityMax950C10...

182 │ Cast.ToDouble(item.ViscosityEndPointBU) != 0 ? Cast.ToDouble(item.ViscosityEndPointBU) :...

183 │ Cast.ToDouble(item.GelatinnizationTemperature) != 0 ? Cast.ToDouble(item.Gelatinnization...

184 │ Cast.ToDouble(item.SO2) != 0 ? Cast.ToDouble(item.SO2) : DBNull.Value,

185 │ Cast.ToDouble(item.AcetylGroup) != 0 ? Cast.ToDouble(item.AcetylGroup) : DBNull.Value,

186 │ Cast.ToDouble(item.DS) != 0 ? Cast.ToDouble(item.DS) : DBNull.Value,

187 │ Cast.ToDouble(item.Conductivity) != 0 ? Cast.ToDouble(item.Conductivity) : DBNull.Value,

188 │ Cast.ToDouble(item.StarchContent) != 0 ? Cast.ToDouble(item.StarchContent) : DBNull.Valu...

189 │ item.ForeignMatter ,

190 │ item.Sensory ,

191 │ item.Note

192 │ });

193 │ }

194 │

195 │ // Prepare file export

196 │ fileName = FileType == "Excel" ? "BotBienTinhThanhPham.xlsx" : "BotBienTinhThanhPham.pdf";

197 │ contentType = FileType == "Excel" ? "application/ms-excel" : "application/pdf";

198 │

199 │ fileTemplate = Path.Combine(\_hostingEnvironment.ContentRootPath, "wwwroot", "templates", "BotBie...

200 │ if (!System.IO.File.Exists(fileTemplate))

201 │ {

202 │ return NotFound("File template not found.");

203 │ }

204 │ ExcelFormat format = new ExcelFormat();

205 │ format.NumberFormat = "#,###0.####";

206 │

207 │ ExcelFormat format2 = new ExcelFormat();

208 │ format2.Border = true;

209 │ ListGiaTriRiengLe = new List<ExcelValue>

210 │ {

211 │ new ExcelValue { Range = "A4:D4", Value = string.Format("Từ ngày {0:dd/MM/yyyy} đế...

212 │ new ExcelValue { Range = "A5:D5", Value = data?.Count>0? "Đơn hàng: " + data?.FirstOr...

213 │ new ExcelValue { Range = "A6:D6", Value = data?.Count>0? "Tên sản phẩm: " + data?.F...

214 │ new ExcelValue { Range = "A7:D7", Value = data?.Count>0? "Lot no: " + data?.FirstOrDefau...

215 │ //new ExcelValue { Range = "A11:U11", Format = format2,KhongGopO = true, KhongDienGiaTri...

216 │ //new ExcelValue { Range = "A11:U11", Format = format,KhongGopO = true, KhongDienGiaTri ...

217 │ };

218 │ DongBatDau = 11;

219 │ cotKT = "U";

220 │ Workbook book = new Workbook();

221 │ book.LoadFromFile(fileTemplate);

222 │ }

223 │ break;

224 │ case "baocaoBotBienTinhBTPTheoNgay":

225 │

226 │ var data2 = (from btp in dbcontext.bt\_BanThanhPhams

227 │ join dv in dbcontext.tbl\_BanThanhPhams on btp.Ma equals dv.MaBanThanhPham

228 │ join cn in dbcontext.tbl\_ChiNhanhs on btp.MaChiNhanh equals cn.Ma

229 │ where (btp.Ngay != null && (btp.Ngay.Value.Date >= dtpTuNgay.Date && btp.Ngay.Value.Dat...

230 │ && btp.MaChiNhanh == branch

231 │ && dv.MaBanThanhPham == maBotBienTinhBTP

232 │ && !(btp.Deleted ?? false) && !(dv.DaXoa ?? false)

233 │ select new

234 │ {

235 │ Gio = dv.Gio,

236 │ Phut = dv.Phut,

237 │ Ngay = dv.CreatedDate,

238 │ ChiNhanh = cn.Ten,

239 │ MaLo = btp.MaLoThuMua,

240 │ MaSP = btp.MaSanPham,

241 │ BonDem = btp.BonDem,

242 │ BonPU = btp.BonPhanUng,

243 │ PhSB = btp.PHSB,

244 │ NDHH = btp.GelatinnizationTemperature,

245 │ Dn = btp.BrabenderViscosity,

246 │ Acetyl = btp.PercentAcetyl,

247 │ DS = dv.DS,

248 │ FreeClo = btp.FreeClo,

249 │ SO2 = dv.SO2,

250 │ Note = dv.Note,

251 │ Ph = dv.pH\_RO,

252 │ HCl = dv.HCL,

253 │ NaOH = dv.NaOH,

254 │ RasperChao = dv.RasperChao,

255 │ CamQuanNaHCO3 = dv.CamQuanNaHCO3,

256 │ pHSuaBot = dv.pHSuaBot,

257 │ BaumeSB = dv.BauMeSuaBot,

258 │ ViscosityEndPointBU = dv.ViscosityEndPointBU,

259 │ ViscosityMax950C0MinBU = dv.ViscosityMax950C0MinBU,

260 │ ViscosityMax950C10MinBU = dv.ViscosityMax950C10MinBU,

261 │ ViscosityPeakBU = dv.ViscosityPeakBU,

262 │ GelatinnizationTemperature = dv.GelatinnizationTemperature,

263 │ AcetylGroup = dv.AcetylGroup,

264 │ Clo = dv.Clo,

265 │ MaSanPham = btp.MaSanPham,

266 │ TinhBotBaMi = dv.TinhBotTrongBaMi

267 │ }

268 │ ).ToList();

269 │ if (!FileType.IsNullOrEmpty())

270 │ {

271 │ dt.Columns.Add("NGÀY");

272 │ dt.Columns.Add("GIỜ", typeof(int));

273 │ dt.Columns.Add("PHÚT", typeof(int));

274 │ dt.Columns.Add("pH RO", typeof(decimal));

275 │ dt.Columns.Add("HCl", typeof(decimal));

276 │ dt.Columns.Add("NaOH", typeof(decimal));

277 │ dt.Columns.Add("Cảm Quan NaHCO3");

278 │ dt.Columns.Add("Rasper cháo", typeof(decimal));

279 │ dt.Columns.Add("Tinh bột trong bã mì (%)");

280 │ dt.Columns.Add("Baume Sữa bột");

281 │ dt.Columns.Add("pH Sữa bột");

282 │ dt.Columns.Add("VISCOSITY PEAK", typeof(decimal));

283 │ dt.Columns.Add("VISCOSITY (START OF HOLDING)", typeof(decimal));

284 │ dt.Columns.Add("VISCOSITY (START OF COOLING)", typeof(decimal));

285 │ dt.Columns.Add("VISCOSITY (END OF FINAL)", typeof(decimal));

286 │ dt.Columns.Add("GELATINIZATION TEMPRATURE", typeof(decimal));

287 │ dt.Columns.Add("SO2", typeof(decimal));

288 │ dt.Columns.Add("ACETYL GROUP", typeof(decimal));

289 │ dt.Columns.Add("DS", typeof(decimal));

290 │ dt.Columns.Add("CLO (%)", typeof(decimal));

291 │ dt.Columns.Add("Note");

292 │ foreach (var item in data2)

293 │ {

294 │ DateTime? datee = item.Ngay;

295 │ datee ??= DateTime.Now;

296 │ string? formattedDate = datee.Value.ToString("dd/MM/yyyy");

297 │

298 │ dt.Rows.Add(new object[] {

299 │ formattedDate,

300 │ Cast.ToInt(item.Gio),

301 │ Cast.ToInt(item.Phut),

302 │ Cast.ToDouble(item.Ph) != 0 ? Cast.ToDouble(item.Ph) : DBNull.Value,

303 │ Cast.ToDouble(item.HCl) != 0 ? Cast.ToDouble(item.HCl) : DBNull.Value,

304 │ Cast.ToDouble(item.NaOH) != 0 ? Cast.ToDouble(item.NaOH) : DBNull.Value,

305 │ item.CamQuanNaHCO3 ,

306 │ Cast.ToDouble(item.RasperChao) != 0 ? Cast.ToDouble(item.RasperChao) : DBNull.Value,

307 │ item.TinhBotBaMi,

308 │ item.BaumeSB,

309 │ item.pHSuaBot,

310 │ Cast.ToDouble(item.ViscosityPeakBU) != 0 ? Cast.ToDouble(item.ViscosityPeakBU) : DBNull....

311 │ Cast.ToDouble(item.ViscosityMax950C0MinBU) != 0 ? Cast.ToDouble(item.ViscosityMax950C0Mi...

312 │ Cast.ToDouble(item.ViscosityMax950C10MinBU) != 0 ? Cast.ToDouble(item.ViscosityMax950C10...

313 │ Cast.ToDouble(item.ViscosityEndPointBU) != 0 ? Cast.ToDouble(item.ViscosityEndPointBU) :...

314 │ Cast.ToDouble(item.GelatinnizationTemperature) != 0 ? Cast.ToDouble(item.Gelatinnization...

315 │ Cast.ToDouble(item.SO2) != 0 ? Cast.ToDouble(item.SO2) : DBNull.Value,

316 │ Cast.ToDouble(item.AcetylGroup) != 0 ? Cast.ToDouble(item.AcetylGroup) : DBNull.Value,

317 │ Cast.ToDouble(item.DS) != 0 ? Cast.ToDouble(item.DS) : DBNull.Value,

318 │ Cast.ToDouble(item.Clo) != 0 ? Cast.ToDouble(item.Clo) : DBNull.Value,

319 │ item.Note

320 │ });

321 │

322 │ // Prepare file export

323 │ fileName = FileType == "Excel" ? "BotBienTinhBanThanhPham.xlsx" : "BotBienTinhBanThanhPham.p...

324 │ contentType = FileType == "Excel" ? "application/ms-excel" : "application/pdf";

325 │

326 │ fileTemplate = Path.Combine(\_hostingEnvironment.ContentRootPath, "wwwroot", "templates", "Bo...

327 │ if (!System.IO.File.Exists(fileTemplate))

328 │ {

329 │ return NotFound("File template not found.");

330 │ }

331 │ ExcelFormat format = new ExcelFormat();

332 │ format.NumberFormat = "#,###0.####";

333 │

334 │ ExcelFormat format2 = new ExcelFormat();

335 │ format2.Border = true;

336 │ ListGiaTriRiengLe = new List<ExcelValue>

337 │ {

338 │ new ExcelValue { Range = "A3:D3", Value = string.Format("Từ ngày {0:dd/MM/yyy...

339 │ new ExcelValue { Range = "A4:D4", Value = data2.Count>0? "Nhà máy: " + data2?....

340 │ new ExcelValue { Range = "A5:D5", Value = data2.Count>0? "Tên sản phẩm: " +...

341 │ new ExcelValue { Range = "A6:D6", Value = data2.Count>0? "Bồn phản ứng: " ...

342 │ new ExcelValue { Range = "A7:D7", Value = data2.Count>0? "Bồn đệm: " + data...

343 │ //new ExcelValue { Range = "A13:U13", Format = format2,KhongGopO = true, KhongDi...

344 │ //new ExcelValue { Range = "A13:U13", Format = format,KhongGopO = true, KhongDie...

345 │ };

346 │ DongBatDau = 13;

347 │ cotKT = "U";

348 │ Workbook book = new Workbook();

349 │ book.LoadFromFile(fileTemplate);

350 │ }

351 │ }

352 │ break;

353 │ case "baocaoBotTuNhienTheoNgay":

354 │ var data3 = (from sl in dbcontext.tbl\_SoLieus

355 │ join ctsl in dbcontext.tbl\_ChiTietSoLieus on sl.Ma equals ctsl.MaSoLieu

356 │ join cn in dbcontext.tbl\_ChiNhanhs on sl.MaChiNhanh equals cn.Ma

357 │ where (sl.Ngay != null && (sl.Ngay.Date >= dtpTuNgay.Date && sl.Ngay.Date <= dtpDenNgay...

358 │ && sl.MaChiNhanh == branch

359 │ && !(ctsl.DaXoa ?? false)

360 │ && !(sl.DaXoa ?? false)

361 │ select new

362 │ {

363 │ Ngay = ctsl.NgayTao,

364 │ Gio = ctsl.Gio,

365 │ Phut = ctsl.Phut,

366 │ Whiteness = ctsl.B\_DoTrang,

367 │ Moisture = ctsl.B\_DoAm,

368 │ Fineness = ctsl.B\_DoMin,

369 │ pHValueBot = ctsl.B\_DoPH,

370 │ pHValueWater = ctsl.NS\_DoPH,

371 │ Residue = ctsl.B\_TapChat,

372 │ ViscosityPeakBU = ctsl.B\_DoKeoBU,

373 │ ViscosityPeakCP = ctsl.B\_DoKeoCP,

374 │ GelatinnizationTemperature = ctsl.ND\_HoHoa,

375 │ DryingTemperature = ctsl.ND\_Say,

376 │ SO2 = ctsl.SO2,

377 │ StarchContent = ctsl.HamLuongTB,

378 │ ForeignMatter = ctsl.ForeignMatter,

379 │ Sensory = ctsl.Sensory,

380 │ Note = ctsl.GhiChu,

381 │ }

382 │ )

383 │ .OrderByDescending(x => x.Ngay)

384 │ .ToList();

385 │ if (!FileType.IsNullOrEmpty())

386 │ {

387 │ dt.Columns.Add("DATE");

388 │ dt.Columns.Add("HOUR", typeof(int));

389 │ dt.Columns.Add("MINUTE", typeof(int));

390 │ dt.Columns.Add("WHITENESS (%) ", typeof(decimal));

391 │ dt.Columns.Add("MOISTURE (%) ", typeof(decimal));

392 │ dt.Columns.Add("FINENESS (%) ", typeof(decimal));

393 │ dt.Columns.Add("PH STARCH", typeof(decimal));

394 │ dt.Columns.Add("PH WATER", typeof(decimal));

395 │ dt.Columns.Add("RESIDUDE (%) ", typeof(decimal));

396 │ dt.Columns.Add("VISCOSITY PEAK (BU)", typeof(decimal));

397 │ dt.Columns.Add("VISCOSITY PEAK (CP)", typeof(decimal));

398 │ dt.Columns.Add("Gelatinization Temperature", typeof(decimal));

399 │ dt.Columns.Add("DRYING Temperature", typeof(decimal));

400 │ dt.Columns.Add("SO2 (mg/kg) ", typeof(decimal));

401 │ dt.Columns.Add("STARCH CONTENT (%)", typeof(decimal));

402 │ dt.Columns.Add("FOREIGN MATTER");

403 │ dt.Columns.Add("SENSORY");

404 │ dt.Columns.Add("Note");

405 │

406 │ foreach (var item in data3)

407 │ {

408 │ DateTime? datee = item.Ngay;

409 │ datee ??= DateTime.Now;

410 │ string? formattedDate = datee.Value.ToString("dd/MM/yyyy");

411 │

412 │ dt.Rows.Add(new object[] {

413 │ formattedDate,

414 │ Cast.ToInt(item.Gio),

415 │ Cast.ToInt(item.Phut),

416 │ Cast.ToDouble(item.Whiteness) != 0 ? Cast.ToDouble(item.Whiteness) : DBNull.Value,

417 │ Cast.ToDouble(item.Moisture) != 0 ? Cast.ToDouble(item.Moisture) : DBNull.Value,

418 │ Cast.ToDouble(item.Fineness) != 0 ? Cast.ToDouble(item.Fineness) : DBNull.Value,

419 │ Cast.ToDouble(item.pHValueBot) != 0 ? Cast.ToDouble(item.pHValueBot) : DBNull.Value,

420 │ Cast.ToDouble(item.pHValueWater) != 0 ? Cast.ToDouble(item.pHValueWater) : DBNull.Value,

421 │ Cast.ToDouble(item.Residue) != 0 ? Cast.ToDouble(item.Residue) : DBNull.Value,

422 │ Cast.ToDouble(item.ViscosityPeakBU) != 0 ? Cast.ToDouble(item.ViscosityPeakBU) : DBNull....

423 │ Cast.ToDouble(item.ViscosityPeakCP) != 0 ? Cast.ToDouble(item.ViscosityPeakCP) : DBNull....

424 │ Cast.ToDouble(item.GelatinnizationTemperature) != 0 ? Cast.ToDouble(item.Gelatinnization...

425 │ Cast.ToDouble(item.DryingTemperature) != 0 ? Cast.ToDouble(item.DryingTemperature) : DBN...

426 │ Cast.ToDouble(item.SO2) != 0 ? Cast.ToDouble(item.SO2) : DBNull.Value,

427 │ Cast.ToDouble(item.StarchContent) != 0 ? Cast.ToDouble(item.StarchContent) : DBNull.Valu...

428 │ item.ForeignMatter ,

429 │ item.Sensory ,

430 │ item.Note

431 │ });

432 │ }

433 │

434 │ // Prepare file export

435 │ fileName = FileType == "Excel" ? "BotTuNhienThanhPham.xlsx" : "BotTuNhienThanhPham.pdf";

436 │ contentType = FileType == "Excel" ? "application/ms-excel" : "application/pdf";

437 │

438 │ fileTemplate = Path.Combine(\_hostingEnvironment.ContentRootPath, "wwwroot", "templates", "BotTuN...

439 │ if (!System.IO.File.Exists(fileTemplate))

440 │ {

441 │ return NotFound("File template not found.");

442 │ }

443 │ ExcelFormat format = new ExcelFormat();

444 │ format.NumberFormat = "#,###0.####";

445 │

446 │ ExcelFormat format2 = new ExcelFormat();

447 │ format2.Border = true;

448 │ ListGiaTriRiengLe = new List<ExcelValue>

449 │ {

450 │ new ExcelValue { Range = "A4:D4", Value = string.Format("Từ ngày {0:dd/MM/yyyy} đế...

451 │ //new ExcelValue { Range = "A5:D5", Value = data?.Count>0? "Đơn hàng: " + data?.First...

452 │ //new ExcelValue { Range = "A6:D6", Value = data?.Count>0? "Tên sản phẩm: " + data?...

453 │ //new ExcelValue { Range = "A7:D7", Value = data?.Count>0? "Lot no: " + data?.FirstOrDef...

454 │ //new ExcelValue { Range = "A11:U11", Format = format2,KhongGopO = true, KhongDienGiaTri...

455 │ //new ExcelValue { Range = "A11:U11", Format = format,KhongGopO = true, KhongDienGiaTri ...

456 │ };

457 │ DongBatDau = 8;

458 │ cotKT = "R";

459 │ Workbook book = new Workbook();

460 │ book.LoadFromFile(fileTemplate);

461 │ }

462 │ break;

463 │ }

464 │ Stream x = ExportToExcel.FromTemplate(fileTemplate, FileType, ListGiaTriRiengLe, dt, true, DongBatDau, "A", ...

465 │ return new FileStreamResult(x, contentType)

466 │ {

467 │ FileDownloadName = fileName

468 │ };

469 │ }

470 │

471 │ [HttpPost("BaocaoBotBienTinhSSTheoNgay")]

472 │ [TokenUserCheckHTTP]

473 │ public IActionResult BaocaoBotBienTinhSSTheoNgay([FromBody] BaoCaoForm model)

474 │ {

475 │ var data = dbcontext.Set<sp\_BaoCaoSoLieuBienTinhSoSanhTheoNgayResult>().FromSqlRaw($"spBaoCaoSoLieuBienTinhS...

476 │

477 │ return Json(new ResponseModel(QLCLConst.OK, "", data));

478 │ }

479 │ [HttpPost("getColorGioiHanSoLieu")]

480 │ [TokenUserCheckHTTP]

481 │ public IActionResult getColorGioiHanSoLieu([FromBody] BaoCaoForm model)

482 │ {

483 │ int branch = model.MaChiNhanh != null ? Cast.ToInt(model.MaChiNhanh) : objToken.branch;

484 │

485 │ var data = (from gh in dbcontext.tbl\_GioiHanSoLieus.AsNoTracking()

486 │ join c in dbcontext.tbl\_ColorGioiHanSoLieus.AsNoTracking()

487 │ on new { gh.Code, gh.MaChiNhanh }

488 │ equals new { Code = c.Code ?? "", MaChiNhanh = c.MaChiNhanh ?? 0 }

489 │

490 │ where !(gh.DaXoa ?? false) && !(c.DaXoa ?? false) && (gh.TrangThai ?? false) && (branch == 0 || ...

491 │ select new ColorGioiHanDto

492 │ {

493 │ MinColor = c.MinColor,

494 │ MaxColor = c.MaxColor,

495 │ Code = gh.Code,

496 │ MaCN = (int)gh.MaChiNhanh,

497 │ Loai = gh.Loai,

498 │ GiaTri = (int)gh.GiaTri,

499 │ Ten = gh.TenNgan,

500 │ }).ToList(); ;

501 │ return Json(new ResponseModel(QLCLConst.OK, "", data));

502 │ }

503 │ [HttpPost("getBranch")]

504 │ [TokenUserCheckHTTP]

505 │ public IActionResult getBranch()

506 │ {

507 │ bool bShowAll = Utility.checkPermissionIDModel(objToken!.userid, (int)MenuID.catChiTietSoLieuBienTinh, (int)...

508 │ List<tbl\_ChiNhanh> sysBranches = new List<tbl\_ChiNhanh>();

509 │ if (bShowAll)

510 │ {

511 │ sysBranches = dbcontext.tbl\_ChiNhanhs.Where(t => !(t.DaXoa ?? false) && (t.TrangThai ?? false)).ToList()...

512 │ }

513 │ else

514 │ {

515 │ sysBranches = dbcontext.tbl\_ChiNhanhs.Where(t => !(t.DaXoa ?? false) && (t.TrangThai ?? false) && t.Ma =...

516 │ }

517 │ return Json(new ResponseModel(QLCLConst.OK, "", sysBranches.Select(i => new

518 │ {

519 │ ma = i.Ma,

520 │ code = i.Code,

521 │ ten = i.Ten,

522 │ diachi = i.DiaChi,

523 │ trangthai = i.TrangThai,

524 │ })));

525 │ }

526 │ [TokenUserCheckHTTP]

527 │ [HttpPost("getLoThuMua")]

528 │ public IActionResult getLoThuMua()

529 │ {

530 │ string codeChiNhanh = dbcontext.tbl\_ChiNhanhs.FirstOrDefault(x => x.Ma == objToken.branch).Code;

531 │ var listLoThuMua = (from l in crmContext.CloseChotLo.Where(t => !(t.Deleted ?? false) && (t.TrangThai != (in...

532 │ join cn in crmContext.SysBranch.Where(t => t.Code == codeChiNhanh && !(t.Deleted ?? fals...

533 │ orderby l.CreatedDate

534 │ select new

535 │ {

536 │ malo = l.Ma,

537 │ codelo = l.MaLo + "",

538 │ }).ToList();

539 │

540 │ return Json(new ResponseModel("OK", "", listLoThuMua));

541 │ }

542 │ }

543 │ }

────────────────────────────────────────────────────────────

**📄 Base.cs (CS, 70 lines)**

1 │ ﻿using Microsoft.EntityFrameworkCore;

2 │ using System.Linq.Expressions;

3 │

4 │ namespace QuanLyChatLuong.Models

5 │ {

6 │ public class Base<T> : IDisposable where T : class

7 │ {

8 │ protected readonly DbContext \_dbContext;

9 │ public Base(DbContext dbContext)

10 │ {

11 │ \_dbContext = dbContext;

12 │ }

13 │

14 │ public T First(Expression<Func<T, bool>> predicate)

15 │ {

16 │ try

17 │ {

18 │ return \_dbContext.Set<T>().First(predicate);

19 │ }

20 │ catch

21 │ {

22 │ return null;

23 │ }

24 │ }

25 │

26 │ public int Add(T item, bool isE = false)

27 │ {

28 │ try

29 │ {

30 │ \_dbContext.Set<T>().Add(item);

31 │ return \_dbContext.SaveChanges() > 0 ? 1 : 0;

32 │ }

33 │ catch (Exception e)

34 │ {

35 │ return -1;

36 │ }

37 │ }

38 │

39 │ public int Update(T item, bool isE = false, bool v = false)

40 │ {

41 │ try

42 │ {

43 │ \_dbContext.Set<T>().Update(item);

44 │ return \_dbContext.SaveChanges() > 0 ? 1 : 0;

45 │ }

46 │ catch

47 │ {

48 │ return -1;

49 │ }

50 │ }

51 │ public IQueryable<T> Get(Expression<Func<T, bool>> predicate)

52 │ {

53 │ return \_dbContext.Set<T>().Where(predicate);

54 │ }

55 │ public IQueryable<T> Get()

56 │ {

57 │ return \_dbContext.Set<T>().Where(t => true);

58 │ }

59 │

60 │ public bool Any(Expression<Func<T, bool>> predicate)

61 │ {

62 │ return \_dbContext.Set<T>().Any(predicate);

63 │ }

64 │

65 │ public void Dispose()

66 │ {

67 │ \_dbContext.Dispose();

68 │ }

69 │ }

70 │ }

────────────────────────────────────────────────────────────

**📄 BaseController.cs (CS, 57 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using QuanLyChatLuong.Common;

3 │ using QuanLyChatLuong.Models;

4 │ using QuanLyChatLuong.Models.APIModels;

5 │

6 │

7 │ namespace QuanLyChatLuong.Controllers

8 │ {

9 │ public abstract class BaseController : Controller

10 │ {

11 │ private UserToken \_objToken;

12 │ public UserToken objToken

13 │ {

14 │ get => \_objToken;

15 │ set

16 │ {

17 │ \_objToken = value;

18 │ dbcontext.userId = \_objToken.userid;

19 │ }

20 │ }

21 │ public MenuID m\_MenuID;

22 │ public bool isAPI;

23 │

24 │ public bool isAdmin

25 │ {

26 │ get

27 │ {

28 │ try

29 │ {

30 │ return dbcontext.vQuyens.Any(t => t.MaQuyen == (int)RoleID.Admin && t.MaMenu == (int)m\_MenuID && t.M...

31 │ }

32 │ catch

33 │ {

34 │ }

35 │ return false;

36 │ }

37 │ }

38 │ public QuanLyChatLuongDbContext dbcontext { get; }

39 │ public MemoryCacheManager memoryCache { get; }

40 │

41 │ public BaseController(QuanLyChatLuongDbContext dbcontext, MemoryCacheManager memoryCache)

42 │ {

43 │ this.dbcontext = dbcontext;

44 │ this.memoryCache = memoryCache;

45 │ }

46 │

47 │ protected override void Dispose(bool disposing)

48 │ {

49 │ if (disposing)

50 │ {

51 │ //dbcontext.Dispose();

52 │ GC.Collect();

53 │ }

54 │ base.Dispose(disposing);

55 │ }

56 │ }

57 │ }

────────────────────────────────────────────────────────────

**📄 BaseRedis.cs (CS, 340 lines)**

1 │ ﻿using Microsoft.EntityFrameworkCore.Storage;

2 │ using Newtonsoft.Json;

3 │ using NReJSON;

4 │ using StackExchange.Redis;

5 │ using System;

6 │ using System.Collections.Generic;

7 │ using System.Drawing.Drawing2D;

8 │ using System.Linq;

9 │ using IDatabase = StackExchange.Redis.IDatabase;

10 │

11 │ namespace QuanLyChatLuong.Models

12 │ {

13 │ public class Redis : IDisposable

14 │ {

15 │ private static string redisConnection = "172.16.10.49:6379,password=tri260197,ssl=False,abortConnect=False,conne...

16 │

17 │ private static readonly ConnectionMultiplexer redisConnections = ConnectionMultiplexer.Connect(redisConnection);

18 │

19 │ private static IDatabase db = redisConnections.GetDatabase();

20 │

21 │ public void SetString<T>(string key, T objectToCache, TimeSpan? expire) where T : class

22 │ {

23 │ // IDatabase db = redisConnections.GetDatabase();

24 │ db.StringSet(key, JsonConvert.SerializeObject(objectToCache

25 │ , Formatting.Indented

26 │ , new JsonSerializerSettings

27 │ {

28 │ ReferenceLoopHandling = ReferenceLoopHandling.Serialize,

29 │ PreserveReferencesHandling = PreserveReferencesHandling.Objects

30 │ }));

31 │ if (expire != null)

32 │ {

33 │ db.KeyExpire(key, expire);

34 │ }

35 │ }

36 │

37 │ public T GetString<T>(string key) where T : class

38 │ {

39 │ var redisObject = db.StringGet(key);

40 │ if (redisObject.HasValue)

41 │ {

42 │ return JsonConvert.DeserializeObject<T>(redisObject

43 │ , new JsonSerializerSettings

44 │ {

45 │ ReferenceLoopHandling = ReferenceLoopHandling.Serialize,

46 │ PreserveReferencesHandling = PreserveReferencesHandling.Objects

47 │ });

48 │ }

49 │ else

50 │ {

51 │ return (T)null;

52 │ }

53 │ }

54 │

55 │ public List<T> GetArray<T>(string key) where T : class

56 │ {

57 │ // IDatabase db = redisConnections.GetDatabase();

58 │ RedisResult redisResult = db.JsonGet(key);

59 │ if (!redisResult.IsNull)

60 │ {

61 │ return JsonConvert.DeserializeObject<List<T>>(redisResult.ToString());

62 │ }

63 │ return new List<T>();

64 │ }

65 │ public void SetListArray<T>(string key, List<T> obj, TimeSpan? expire = null) where T : class

66 │ {

67 │ // IDatabase db = redisConnections.GetDatabase();

68 │ //RedisResult redisResult = db.JsonGet(key);

69 │ db.JsonSet(key, JsonConvert.SerializeObject(obj));

70 │ if (expire != null)

71 │ {

72 │ db.KeyExpire(key, expire);

73 │ }

74 │ }

75 │

76 │ public void SetArray<T>(string key, T obj, TimeSpan? expire = null, string properties = "") where T : class

77 │ {

78 │ // IDatabase db = redisConnections.GetDatabase();

79 │ RedisResult redisResult = db.JsonGet(key);

80 │ if (redisResult.IsNull)

81 │ {

82 │ List<T> x = new List<T>

83 │ {

84 │ obj

85 │ };

86 │ db.JsonSet(key, JsonConvert.SerializeObject(x));

87 │ if (expire != null)

88 │ {

89 │ db.KeyExpire(key, expire);

90 │ }

91 │ }

92 │ try

93 │ {

94 │ List<T> data = JsonConvert.DeserializeObject<List<T>>(redisResult.ToString());

95 │ if (data != null && data.Count > 0)

96 │ {

97 │ if (properties != "")

98 │ {

99 │ int i = 0;

100 │ var propertiesInfo = typeof(T).GetProperty(properties);

101 │ object propertiesValue = propertiesInfo.GetValue(obj, null);

102 │ if (propertiesValue == null)

103 │ {

104 │ // throw new Exception("Properties not Found");

105 │ return;

106 │ }

107 │ foreach (T t in data)

108 │ {

109 │ if (propertiesInfo.GetValue(t, null).Equals(propertiesValue))

110 │ {

111 │ db.JsonSet(key, JsonConvert.SerializeObject(obj), "$.[" + i + "]");

112 │ return;

113 │ }

114 │ else

115 │ {

116 │ i++;

117 │ }

118 │ }

119 │ }

120 │ db.JsonArrayAppend(key, ".", JsonConvert.SerializeObject(obj));

121 │ }

122 │ else

123 │ {

124 │ List<T> x = new List<T>

125 │ {

126 │ obj

127 │ };

128 │ db.JsonSet(key, JsonConvert.SerializeObject(x));

129 │ if (expire != null)

130 │ {

131 │ db.KeyExpire(key, expire);

132 │ }

133 │ }

134 │ }

135 │ catch

136 │ {

137 │ }

138 │ }

139 │

140 │ public void DelItemsArray<T>(string key, Func<T, bool> query)

141 │ {

142 │ // IDatabase db = redisConnections.GetDatabase();

143 │ var indexs = get(key, query).Select((s, i) => i).ToList();

144 │ foreach (int i in indexs)

145 │ {

146 │ db.JsonArrayPop(key, index: i);

147 │ }

148 │ }

149 │

150 │ public List<T> get<T>(string key, Func<T, bool> query)

151 │ {

152 │ // IDatabase db = redisConnections.GetDatabase();

153 │ RedisResult redisResult = db.JsonGet(key);

154 │ if (!redisResult.IsNull)

155 │ {

156 │ List<T> item = JsonConvert.DeserializeObject<List<T>>(redisResult.ToString());

157 │ if (item.Count > 0)

158 │ {

159 │ return item.Where(query).ToList();

160 │ }

161 │ }

162 │ return new List<T>();

163 │ }

164 │ public List<T> get<T>(string key)

165 │ {

166 │ // IDatabase db = redisConnections.GetDatabase();

167 │ //RedisResult redisResult = db.JsonGet(key).;

168 │ try

169 │ {

170 │ return JsonConvert.DeserializeObject<List<T>>(db.JsonGet(key).ToString());

171 │ }

172 │ catch

173 │ {

174 │ }

175 │ return new List<T>();

176 │ }

177 │

178 │ public bool isKeyExist(string key)

179 │ {

180 │ // IDatabase db = redisConnections.GetDatabase();

181 │ return db.KeyExists(key);

182 │ }

183 │

184 │ public bool DelItemArray<T>(string key, Func<T, bool> query)

185 │ {

186 │ // IDatabase db = redisConnections.GetDatabase();

187 │ try

188 │ {

189 │ int index = get(key, query).Select((s, i) => i).FirstOrDefault();

190 │ db.JsonArrayPop(key, index: index);

191 │ return true;

192 │ }

193 │ catch

194 │ {

195 │ return false;

196 │ }

197 │ }

198 │

199 │

200 │ public void CleanItemArray(string key)

201 │ {

202 │ // IDatabase db = redisConnections.GetDatabase();

203 │ db.JsonClear(key, ".");

204 │ }

205 │

206 │ public void Dispose()

207 │ {

208 │ redisConnections.Dispose();

209 │ }

210 │

211 │ // THÊM VÀO Redis CLASS các methods sau:

212 │

213 │ /// <summary>

214 │ /// Lưu string value với key và thời gian expire

215 │ /// </summary>

216 │ public void SetValue(string key, string value, TimeSpan? expire = null)

217 │ {

218 │ db.StringSet(key, value);

219 │ if (expire != null)

220 │ {

221 │ db.KeyExpire(key, expire);

222 │ }

223 │ }

224 │

225 │ /// <summary>

226 │ /// Lấy string value theo key

227 │ /// </summary>

228 │ public string GetValue(string key)

229 │ {

230 │ var redisValue = db.StringGet(key);

231 │ return redisValue.HasValue ? redisValue.ToString() : null;

232 │ }

233 │

234 │ /// <summary>

235 │ /// Xóa key khỏi Redis

236 │ /// </summary>

237 │ public void RemoveKey(string key)

238 │ {

239 │ db.KeyDelete(key);

240 │ }

241 │

242 │ /// <summary>

243 │ /// Xóa nhiều keys khỏi Redis

244 │ /// </summary>

245 │ public void RemoveKeys(params string[] keys)

246 │ {

247 │ if (keys != null && keys.Length > 0)

248 │ {

249 │ var redisKeys = keys.Select(k => (RedisKey)k).ToArray();

250 │ db.KeyDelete(redisKeys);

251 │ }

252 │ }

253 │

254 │ /// <summary>

255 │ /// Lấy thời gian sống còn lại của key (TTL)

256 │ /// </summary>

257 │ public TimeSpan? GetKeyTimeToLive(string key)

258 │ {

259 │ return db.KeyTimeToLive(key);

260 │ }

261 │

262 │ /// <summary>

263 │ /// Set expire cho key đã tồn tại

264 │ /// </summary>

265 │ public bool SetKeyExpire(string key, TimeSpan expire)

266 │ {

267 │ return db.KeyExpire(key, expire);

268 │ }

269 │

270 │ /// <summary>

271 │ /// Kiểm tra key có tồn tại không (alias cho isKeyExist)

272 │ /// </summary>

273 │ public bool KeyExists(string key)

274 │ {

275 │ return db.KeyExists(key);

276 │ }

277 │

278 │ /// <summary>

279 │ /// Lưu object với expire (wrapper cho SetString)

280 │ /// </summary>

281 │ public void SetObject<T>(string key, T obj, TimeSpan? expire = null) where T : class

282 │ {

283 │ SetString(key, obj, expire);

284 │ }

285 │

286 │ /// <summary>

287 │ /// Lấy object (wrapper cho GetString)

288 │ /// </summary>

289 │ public T GetObject<T>(string key) where T : class

290 │ {

291 │ return GetString<T>(key);

292 │ }

293 │

294 │ /// <summary>

295 │ /// Atomic increment counter

296 │ /// </summary>

297 │ public long Increment(string key, long value = 1)

298 │ {

299 │ return db.StringIncrement(key, value);

300 │ }

301 │

302 │ /// <summary>

303 │ /// Atomic decrement counter

304 │ /// </summary>

305 │ public long Decrement(string key, long value = 1)

306 │ {

307 │ return db.StringDecrement(key, value);

308 │ }

309 │

310 │ /// <summary>

311 │ /// Set value chỉ khi key chưa tồn tại

312 │ /// </summary>

313 │ public bool SetValueIfNotExists(string key, string value, TimeSpan? expire = null)

314 │ {

315 │ bool success = db.StringSet(key, value, expire, When.NotExists);

316 │ return success;

317 │ }

318 │

319 │ /// <summary>

320 │ /// Tìm tất cả keys theo pattern

321 │ /// </summary>

322 │ public List<string> GetKeysByPattern(string pattern)

323 │ {

324 │ var server = redisConnections.GetServer(redisConnection.Split(',')[0]);

325 │ return server.Keys(pattern: pattern).Select(k => k.ToString()).ToList();

326 │ }

327 │

328 │ /// <summary>

329 │ /// Xóa tất cả keys theo pattern

330 │ /// </summary>

331 │ public void RemoveKeysByPattern(string pattern)

332 │ {

333 │ var keys = GetKeysByPattern(pattern);

334 │ if (keys.Any())

335 │ {

336 │ RemoveKeys(keys.ToArray());

337 │ }

338 │ }

339 │ }

340 │ }

────────────────────────────────────────────────────────────

**📄 BotBienTinhBTPController.cs (CS, 565 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using Microsoft.EntityFrameworkCore;

3 │ using Newtonsoft.Json;

4 │ using QuanLyChatLuong.Common;

5 │ using QuanLyChatLuong.Models;

6 │ using QuanLyChatLuong.Models.APIModels;

7 │ using QuanLyChatLuong.Models.CRMEntity;

8 │ using System.Globalization;

9 │ using System.Reflection;

10 │ using static QuanLyChatLuong.Controllers.BotBienTinhController;

11 │

12 │ namespace QuanLyChatLuong.Controllers

13 │ {

14 │ [Route("BotBienTinhBTP")]

15 │ public class BotBienTinhBTPController : BaseController

16 │ {

17 │ private readonly QuanLyChatLuongDbContext \_dbContext;

18 │ private CrmContext crmContext { get; }

19 │

20 │ public BotBienTinhBTPController(QuanLyChatLuongDbContext dbcontext, CrmContext crmdbcontext, MemoryCacheManager ...

21 │ {

22 │ \_dbContext = dbcontext;

23 │ m\_MenuID = MenuID.botBienTinhBTP;

24 │ crmContext = crmdbcontext;

25 │ }

26 │ public class BotBienTinhBTP

27 │ {

28 │ public int? BanThanhPhamId { get; set; }

29 │ public string? eMa { get; set; }

30 │ public string? ListCT { get; set; }

31 │ public string? MaLoThuMua { get; set; }

32 │ public string? MaSanPham { get; set; }

33 │ public string? BonPhanUng { get; set; }

34 │ public string? BonDem { get; set; }

35 │ public string? TruongCa { get; set; }

36 │ public string? PhanUng { get; set; }

37 │ public string? QC { get; set; }

38 │ public string? GhiChu { get; set; }

39 │ public string? Status { get; set; }

40 │ public string Ngay { get; set; } = DateTime.Now.ToString();

41 │ }

42 │

43 │ public class ChiTietBotBienTinhBTP()

44 │ {

45 │ public int? CT { get; set; }

46 │ public int? Ma { get; set; }

47 │ public string? Gio { get; set; }

48 │ public string? Phut { get; set; }

49 │ public string? PHRO { get; set; }

50 │ public string? HCL { get; set; }

51 │ public string? NaOH { get; set; }

52 │ public string? CamQuanNaHCO3 { get; set; }

53 │ public string? RasperChao { get; set; }

54 │ public string? TinhBotTrongBaMi { get; set; }

55 │ public string? BauMeSuaBot { get; set; }

56 │ public string? PHSuaBot { get; set; }

57 │ public string? ViscosityPeakBU { get; set; }

58 │ public string? ViscosityMax950C0MinBU { get; set; }

59 │ public string? ViscosityMax950C10MinBU { get; set; }

60 │ public string? ViscosityEndPointBU { get; set; }

61 │ public string? GelatinnizationTemperature { get; set; }

62 │ public string? SO2 { get; set; }

63 │ public string? AcetylGroup { get; set; }

64 │ public string? DS { get; set; }

65 │ public string? Clo { get; set; }

66 │ public string? Note { get; set; }

67 │ }

68 │

69 │ public class RequestBody

70 │ {

71 │ public string TuNgay { get; set; } = DateTime.Now.ToString();

72 │ public string DenNgay { get; set; } = DateTime.Now.ToString();

73 │ public string? MaChiNhanh { get; set; }

74 │ }

75 │ [HttpGet]

76 │ [TokenUserCheckHTTP]

77 │ public IActionResult Index()

78 │ {

79 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinhBTP, (int)Permiss...

80 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinhBTP, (int)Permis...

81 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinhBTP, (int)Perm...

82 │ return View();

83 │ }

84 │ [HttpPost("getsolieutheongay")]

85 │ [TokenUserCheckHTTP]

86 │ public IActionResult GetSoLieuTheoNgay([FromBody] RequestBody model)

87 │ {

88 │

89 │ DateTime TuNgay = DateTime.Parse(model.TuNgay, CultureInfo.InvariantCulture);

90 │ DateTime DenNgay = DateTime.Parse(model.DenNgay, CultureInfo.InvariantCulture);

91 │

92 │ if (TuNgay > DenNgay)

93 │ {

94 │ return Json(new ResponseModel("Error", "Từ ngày không lớn hơn Đến Ngày."));

95 │ }

96 │

97 │ int branch = objToken.branch;

98 │

99 │ if (model.MaChiNhanh != null && !model.MaChiNhanh.IsEmpty())

100 │ {

101 │ branch = Cast.ToInt(model.MaChiNhanh);

102 │ }

103 │

104 │ var chiNhanh = dbcontext.tbl\_ChiNhanhs.FirstOrDefault(x => x.Ma == branch);

105 │

106 │ string codeChiNhanh = chiNhanh?.Code ?? "";

107 │

108 │ var data = dbcontext.bt\_BanThanhPhams

109 │ .AsNoTracking()

110 │ .Where(x => (x.Deleted == null || x.Deleted == false)

111 │ && x.Ngay.HasValue

112 │ && x.Ngay.Value.Date >= TuNgay && x.Ngay.Value.Date <= DenNgay

113 │ && x.MaChiNhanh == branch)

114 │ .OrderByDescending(x => x.Ngay)

115 │ .ToList()

116 │ .Select(t => new

117 │ {

118 │ Ma = t.Ma,

119 │ eMa = t.Ma.Encrypt(),

120 │ Gio = t.Gio,

121 │ Phut = t.Phut,

122 │ MaLoThuMua = t.MaLoThuMua,

123 │ MaSanPham = t.MaSanPham,

124 │ BonPhanUng = t.BonPhanUng,

125 │ BonDem = t.BonDem,

126 │ PHSB = t.PHSB,

127 │ GelatinnizationTemperature = t.GelatinnizationTemperature,

128 │ BrabenderViscosity = t.BrabenderViscosity,

129 │ PercentAcetyl = t.PercentAcetyl,

130 │ DS = t.DS,

131 │ FreeClo = t.FreeClo,

132 │ SulfurDioxide = t.SulfurDioxide,

133 │ ConductivityOfHydrocarbons = t.ConductivityOfHydrocarbons,

134 │ Note = t.Note,

135 │ UserAnalyst = t.UserAnalyst,

136 │

137 │ }).ToList();

138 │

139 │ if (data.Count > 0)

140 │ {

141 │ List<tbl\_BanThanhPham> dataBanThanhPham = new List<tbl\_BanThanhPham>();

142 │

143 │ var listLoThuMua = (from l in crmContext.CloseChotLo.Where(t => !(t.Deleted ?? false) && (t.TrangThai !=...

144 │ join cn in crmContext.SysBranch.Where(t => t.Code == codeChiNhanh && !(t.Deleted ?? ...

145 │ orderby l.CreatedDate

146 │ select new

147 │ {

148 │ malo = l.Ma,

149 │ codelo = l.MaLo + "",

150 │ }).ToList();

151 │

152 │ foreach (var item in data)

153 │ {

154 │ dataBanThanhPham.AddRange(dbcontext.tbl\_BanThanhPhams

155 │ .Where(x => x.MaBanThanhPham == item.Ma && !(x.DaXoa ?? false))

156 │ .OrderByDescending(x => x.UserCreated)

157 │ .ToList());

158 │ }

159 │ return Json(new ResponseModel(QLCLConst.OK, "", new

160 │ {

161 │ ListBanThanhPham = data,

162 │ ListDauVao = dataBanThanhPham,

163 │ ListLo = listLoThuMua

164 │ }));

165 │ }

166 │ return Json(new ResponseModel(QLCLConst.OK, "", new

167 │ {

168 │ ListBanThanhPham = data,

169 │ }));

170 │ }

171 │

172 │ [HttpGet("add")]

173 │ [TokenUserCheckHTTP]

174 │ public IActionResult Add()

175 │ {

176 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinhBTP, (int)Permiss...

177 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinhBTP, (int)Permis...

178 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinhBTP, (int)Perm...

179 │ return View("Form");

180 │ }

181 │ [HttpGet("update")]

182 │ [TokenUserCheckHTTP(Permission.Edit)]

183 │ public IActionResult Edit(string Ma)

184 │ {

185 │ var data = dbcontext.bt\_BanThanhPhams.Find(Cast.ToInt(Ma.Decrypt()));

186 │ if(data == null)

187 │ {

188 │ return NotFound("Bot bien tinh BTP not found");

189 │ }

190 │ ViewData["BanTP"] = data;

191 │ ViewData["DauVao"] = dbcontext.tbl\_BanThanhPhams.Where(x => x.MaBanThanhPham == Cast.ToInt(Ma.Decrypt()) && ...

192 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinhBTP, (int)Permiss...

193 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinhBTP, (int)Permis...

194 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinhBTP, (int)Perm...

195 │ return View("Form");

196 │ }

197 │

198 │ [HttpPost("updateapi")]

199 │ [TokenUserCheckHTTP(Permission.Edit)]

200 │ public async Task<IActionResult> UpdateAPIAsync([FromBody] BotBienTinhBTP model)

201 │ {

202 │ var idUser = objToken.userid;

203 │ var idBranch = objToken.branch;

204 │

205 │ if (model.BanThanhPhamId == null)

206 │ {

207 │ return Json(new ResponseModel("Error", "Lỗi không xác định được Số liệu."));

208 │ }

209 │ if (idUser == 0 || idBranch == 0)

210 │ {

211 │ return Json(new ResponseModel("Error", "Hết phiên đăng nhập. Vui lòng đăng nhập lại."));

212 │ }

213 │ if (model.TruongCa == null

214 │ || model.Ngay == null

215 │ || model.QC == null)

216 │ {

217 │ return Json(Utility.MakeResponseFromEditResult("0"));

218 │ }

219 │

220 │ int result = 0;

221 │

222 │ using (var transaction = await \_dbContext.Database.BeginTransactionAsync())

223 │ {

224 │ try

225 │ {

226 │ bt\_BanThanhPham? item = await \_dbContext.bt\_BanThanhPhams.FindAsync(Cast.ToInt(model.BanThanhPhamId)...

227 │

228 │ item.MaLoThuMua = Cast.ToInt(model.MaLoThuMua);

229 │ item.MaSanPham = model.MaSanPham;

230 │ item.BonPhanUng = model.BonPhanUng;

231 │ item.BonDem = model.BonDem;

232 │

233 │ item.TruongCa = model.TruongCa;

234 │ item.PhanUng = model.PhanUng;

235 │ item.QC = model.QC;

236 │ item.Status = model.Status == "1";

237 │ item.Note = model.GhiChu;

238 │ item.UserModified = Cast.ToInt(idUser);

239 │ item.ModifiedDate = DateTime.Now;

240 │ item.MaChiNhanh = Cast.ToInt(idBranch);

241 │

242 │ \_dbContext.bt\_BanThanhPhams.Update(item);

243 │ await \_dbContext.SaveChangesAsync();

244 │

245 │ if (!string.IsNullOrEmpty(model.ListCT) && model.ListCT.Length > 3)

246 │ {

247 │ try

248 │ {

249 │ List<ChiTietBotBienTinhBTP> dataList = JsonConvert.DeserializeObject<List<ChiTietBotBienTinh...

250 │

251 │ List<tbl\_BanThanhPham> chitietSLCu = await \_dbContext.tbl\_BanThanhPhams

252 │ .Where(x => x.MaBanThanhPham == item.Ma && !(x.DaXoa ?? false))

253 │ .ToListAsync();

254 │

255 │ foreach (var ct in dataList)

256 │ {

257 │ tbl\_BanThanhPham? chitiet = await \_dbContext.tbl\_BanThanhPhams.FindAsync(ct.Ma);

258 │

259 │ if (chitiet == null)

260 │ {

261 │ chitiet = new tbl\_BanThanhPham

262 │ {

263 │ MaBanThanhPham = item.Ma,

264 │ CreatedDate = DateTime.Now

265 │ };

266 │ }

267 │

268 │ chitiet.Gio = ct.Gio;

269 │ chitiet.Phut = ct.Phut;

270 │ chitiet.MaBanThanhPham = item.Ma;

271 │

272 │ chitiet.pH\_RO = ct.PHRO.ToDecimal() == 0 ? null : ct.PHRO.ToDecimal();

273 │ chitiet.HCL = ct.HCL.ToDecimal() == 0 ? null : ct.HCL.ToDecimal();

274 │ chitiet.NaOH = ct.NaOH.ToDecimal() == 0 ? null : ct.NaOH.ToDecimal();

275 │

276 │ chitiet.CamQuanNaHCO3 = ct.CamQuanNaHCO3;

277 │ chitiet.RasperChao = ct.RasperChao.ToDecimal() == 0 ? null : ct.RasperChao.ToDecimal();

278 │

279 │ chitiet.TinhBotTrongBaMi = ct.TinhBotTrongBaMi;

280 │ chitiet.BauMeSuaBot = ct.BauMeSuaBot;

281 │ chitiet.pHSuaBot = ct.PHSuaBot;

282 │

283 │ chitiet.ViscosityEndPointBU = ct.ViscosityEndPointBU.ToDecimal() == 0 ? null : ct.Viscos...

284 │ chitiet.ViscosityMax950C0MinBU = ct.ViscosityMax950C0MinBU.ToDecimal() == 0 ? null : ct....

285 │ chitiet.ViscosityMax950C10MinBU = ct.ViscosityMax950C10MinBU.ToDecimal() == 0 ? null : c...

286 │ chitiet.ViscosityPeakBU = ct.ViscosityPeakBU.ToDecimal() == 0 ? null : ct.ViscosityPeakB...

287 │

288 │ chitiet.GelatinnizationTemperature = ct.GelatinnizationTemperature.ToDecimal() == 0 ? nu...

289 │ chitiet.SO2 = ct.SO2.ToDecimal() == 0 ? null : ct.SO2.ToDecimal();

290 │ chitiet.AcetylGroup = ct.AcetylGroup.ToDecimal() == 0 ? null : ct.AcetylGroup.ToDecimal(...

291 │ chitiet.DS = ct.DS.ToDecimal() == 0 ? null : ct.DS.ToDecimal();

292 │ chitiet.Clo = ct.Clo.ToDecimal() == 0 ? null : ct.Clo.ToDecimal();

293 │

294 │ chitiet.Note = ct.Note;

295 │

296 │ chitiet.UserModified = Cast.ToInt(idUser);

297 │ chitiet.ModifiedDate = DateTime.Now;

298 │

299 │ \_dbContext.tbl\_BanThanhPhams.Update(chitiet);

300 │

301 │ int MaChitiet = chitiet.Ma;

302 │ if (chitietSLCu.Any(x => x.Ma == MaChitiet))

303 │ {

304 │ chitietSLCu.RemoveAll(x => x.Ma == MaChitiet);

305 │ }

306 │ }

307 │

308 │ await \_dbContext.SaveChangesAsync();

309 │

310 │ if (chitietSLCu.Count > 0)

311 │ {

312 │ foreach (var item1 in chitietSLCu)

313 │ {

314 │ item1.DaXoa = true;

315 │ item1.UserDeleted = Cast.ToInt(idUser);

316 │ item1.CreatedDate = DateTime.Now;

317 │ \_dbContext.tbl\_BanThanhPhams.Update(item1);

318 │ }

319 │

320 │ await \_dbContext.SaveChangesAsync();

321 │ }

322 │ }

323 │ catch (Exception e)

324 │ {

325 │ return Json(new ResponseModel("Error", e.Message));

326 │ }

327 │ }

328 │ await transaction.CommitAsync();

329 │ result = 1;

330 │ }

331 │ catch (Exception ex)

332 │ {

333 │ // Nếu có lỗi, rollback transaction

334 │ await transaction.RollbackAsync();

335 │ return Json(new ResponseModel("Error", ex.Message));

336 │ }

337 │ }

338 │ memoryCache.Clear(t => t.Contains(QLCLConst.cacheKeyMenu));

339 │ return Json(Utility.MakeResponseFromEditResult(Cast.ToString(result)));

340 │

341 │ }

342 │ [HttpPost("addapi")]

343 │ [TokenUserCheckHTTP(Permission.Add)]

344 │ public async Task<IActionResult> AddAPIAsync([FromBody] BotBienTinhBTP model)

345 │ {

346 │ var idUser = objToken.userid;

347 │ var idBranch = objToken.branch;

348 │

349 │ if (idUser == 0 || idBranch == 0)

350 │ {

351 │ return Json(new ResponseModel("Error", "Hết phiên đăng nhập. Vui lòng đăng nhập lại."));

352 │ }

353 │

354 │ if (model.MaSanPham == null ||

355 │ model.BonDem == null ||

356 │ model.TruongCa == null ||

357 │ model.PhanUng == null ||

358 │ model.QC == null)

359 │ {

360 │ return Json(Utility.MakeResponseFromAddResult("0"));

361 │ }

362 │

363 │ DateTime ngayNhap = DateTime.Parse(model.Ngay, CultureInfo.InvariantCulture);

364 │

365 │ int result = 0;

366 │

367 │ if (!string.IsNullOrEmpty(model.ListCT) && model.ListCT.Length > 2)

368 │ {

369 │ try

370 │ {

371 │ List<ChiTietBotBienTinhBTP> dataList = JsonConvert.DeserializeObject<List<ChiTietBotBienTinhBTP>>(mo...

372 │

373 │ if (CheckThoiGian(dataList, ngayNhap, model.TruongCa, model.QC))

374 │ {

375 │ return Json(new ResponseModel("Error", "Chi Tiết Bột biến tính có dữ liệu bị trùn...

376 │ }

377 │ }

378 │ catch (Exception e)

379 │ {

380 │ result = 0;

381 │ return Json(Utility.MakeResponseFromAddResult(Cast.ToString(result)));

382 │ }

383 │ }

384 │ using (var transaction = await dbcontext.Database.BeginTransactionAsync())

385 │ {

386 │ try

387 │ {

388 │ bt\_BanThanhPham item = new bt\_BanThanhPham();

389 │ item.Ngay = ngayNhap;

390 │

391 │ item.MaLoThuMua = Cast.ToInt(model.MaLoThuMua);

392 │ item.MaSanPham = model.MaSanPham;

393 │ item.BonPhanUng = model.BonPhanUng;

394 │ item.BonDem = model.BonDem;

395 │

396 │ item.TruongCa = model.TruongCa;

397 │ item.PhanUng = model.PhanUng;

398 │ item.QC = model.QC;

399 │ item.Status = model.Status == "1";

400 │ item.Note = model.GhiChu;

401 │ item.UserCreated = Cast.ToInt(idUser);

402 │ item.CreatedDate = DateTime.Now;

403 │ item.MaChiNhanh = Cast.ToInt(idBranch);

404 │

405 │ // Thêm dữ liệu vào bảng

406 │ \_dbContext.bt\_BanThanhPhams.Add(item);

407 │ await \_dbContext.SaveChangesAsync();

408 │

409 │ // Nếu có danh sách chi tiết BTP, thêm dữ liệu vào bảng tbl\_BanThanhPham

410 │ if (!string.IsNullOrEmpty(model.ListCT) && model.ListCT.Length > 2)

411 │ {

412 │ try

413 │ {

414 │ List<ChiTietBotBienTinhBTP> dataList = JsonConvert.DeserializeObject<List<ChiTietBotBienTinh...

415 │

416 │ foreach (var ct in dataList)

417 │ {

418 │ tbl\_BanThanhPham chitiet = new tbl\_BanThanhPham();

419 │

420 │ chitiet.Gio = ct.Gio;

421 │ chitiet.Phut = ct.Phut;

422 │ chitiet.MaBanThanhPham = item.Ma;

423 │

424 │ chitiet.pH\_RO = ct.PHRO.ToDecimal() == 0 ? null : ct.PHRO.ToDecimal();

425 │ chitiet.HCL = ct.HCL.ToDecimal() == 0 ? null : ct.HCL.ToDecimal();

426 │ chitiet.NaOH = ct.NaOH.ToDecimal() == 0 ? null : ct.NaOH.ToDecimal();

427 │

428 │ chitiet.CamQuanNaHCO3 = ct.CamQuanNaHCO3;

429 │ chitiet.RasperChao = ct.RasperChao.ToDecimal() == 0 ? null : ct.RasperChao.ToDecimal();

430 │

431 │ chitiet.TinhBotTrongBaMi = ct.TinhBotTrongBaMi;

432 │ chitiet.BauMeSuaBot = ct.BauMeSuaBot;

433 │ chitiet.pHSuaBot = ct.PHSuaBot;

434 │

435 │ chitiet.ViscosityEndPointBU = ct.ViscosityEndPointBU.ToDecimal() == 0 ? null : ct.Viscos...

436 │ chitiet.ViscosityMax950C0MinBU = ct.ViscosityMax950C0MinBU.ToDecimal() == 0 ? null : ct....

437 │ chitiet.ViscosityMax950C10MinBU = ct.ViscosityMax950C10MinBU.ToDecimal() == 0 ? null : c...

438 │ chitiet.ViscosityPeakBU = ct.ViscosityPeakBU.ToDecimal() == 0 ? null : ct.ViscosityPeakB...

439 │

440 │ chitiet.GelatinnizationTemperature = ct.GelatinnizationTemperature.ToDecimal() == 0 ? nu...

441 │ chitiet.SO2 = ct.SO2.ToDecimal() == 0 ? null : ct.SO2.ToDecimal();

442 │ chitiet.AcetylGroup = ct.AcetylGroup.ToDecimal() == 0 ? null : ct.AcetylGroup.ToDecimal(...

443 │ chitiet.DS = ct.DS.ToDecimal() == 0 ? null : ct.DS.ToDecimal();

444 │ chitiet.Clo = ct.Clo.ToDecimal() == 0 ? null : ct.Clo.ToDecimal();

445 │

446 │ chitiet.Note = ct.Note;

447 │

448 │ chitiet.UserCreated = Cast.ToInt(idUser);

449 │ chitiet.CreatedDate = DateTime.Now;

450 │

451 │ \_dbContext.tbl\_BanThanhPhams.Add(chitiet);

452 │ await \_dbContext.SaveChangesAsync();

453 │ }

454 │ }

455 │ catch (Exception e)

456 │ {

457 │ Console.WriteLine("Lỗi parse JSON: " + e.Message);

458 │ return Json(new ResponseModel("Error", e.Message));

459 │ }

460 │ }

461 │ await transaction.CommitAsync();

462 │ result = 1;

463 │ }

464 │ catch (Exception ex)

465 │ {

466 │ // Nếu có lỗi, rollback transaction

467 │ await transaction.RollbackAsync();

468 │ return Json(new ResponseModel("Error", ex.Message));

469 │ }

470 │ }

471 │

472 │ return Json(Utility.MakeResponseFromAddResult(Cast.ToString(result)));

473 │ }

474 │

475 │ [HttpPost("deleteapi")]

476 │ [TokenUserCheckHTTP(Permission.Delete)]

477 │ public IActionResult DeleteAPI([FromBody] BotBienTinhBTP model)

478 │ {

479 │ int result = 1;

480 │ var userID = objToken.userid;

481 │ if (userID == 0)

482 │ {

483 │ return Json(new ResponseModel("Error", "Hết phiên đăng nhập. Vui lòng đăng nhập lại."));

484 │ }

485 │ if (model.BanThanhPhamId != null)

486 │ {

487 │ bt\_BanThanhPham? item = dbcontext.bt\_BanThanhPhams.Find(Cast.ToInt(model.eMa.Decrypt()));

488 │ if (item == null)

489 │ {

490 │ return Json(new ResponseModel("Error", "Lỗi không xác định được dữ liệu."));

491 │ }

492 │ item.Deleted = true;

493 │ item.DeletedDate = DateTime.Now;

494 │ item.UserDeleted = Cast.ToInt(userID);

495 │ dbcontext.bt\_BanThanhPhams.Update(item);

496 │ dbcontext.SaveChanges();

497 │ tbl\_BanThanhPham? item2 = dbcontext.tbl\_BanThanhPhams.Find(Cast.ToInt(item.Ma));

498 │ if (item2 == null)

499 │ {

500 │ return Json(new ResponseModel("Error", "Lỗi không xác định được dữ liệu."));

501 │ }

502 │ item2.DaXoa = true;

503 │ item2.DeletedDate = DateTime.Now;

504 │ item2.UserDeleted = Cast.ToInt(userID);

505 │ dbcontext.tbl\_BanThanhPhams.Update(item2);

506 │ dbcontext.SaveChanges();

507 │ }

508 │ memoryCache.Clear(t => t.Contains(QLCLConst.cacheKeyMenu));

509 │ return Json(Utility.MakeResponseFromDeleteResult(Cast.ToString(result)));

510 │ }

511 │ [TokenUserCheckHTTP]

512 │ [HttpPost("getLoThuMua")]

513 │ public IActionResult getLoThuMua()

514 │ {

515 │ var chiNhanh = dbcontext.tbl\_ChiNhanhs.FirstOrDefault(x => x.Ma == objToken.branch);

516 │

517 │ string codeChiNhanh = chiNhanh?.Code ?? "";

518 │

519 │ var listLoThuMua = (from l in crmContext.CloseChotLo.Where(t => !(t.Deleted ?? false) && (t.TrangThai != (in...

520 │ join cn in crmContext.SysBranch.Where(t => t.Code == codeChiNhanh && !(t.Deleted ?? fals...

521 │ orderby l.CreatedDate

522 │ select new

523 │ {

524 │ malo = l.Ma,

525 │ codelo = l.MaLo + "",

526 │ }).ToList();

527 │

528 │ return Json(new ResponseModel("OK", "", listLoThuMua));

529 │ }

530 │ private bool CheckThoiGian(List<ChiTietBotBienTinhBTP> dataList, DateTime ngay, string TruongCa, string QC)

531 │ {

532 │ bt\_BanThanhPham? banThanhPham = dbcontext.bt\_BanThanhPhams

533 │ .FirstOrDefault(x => x.Ngay != null && x.Ngay.Value.Date == ngay.Date

534 │ && x.QC != null && x.QC == QC

535 │ && x.TruongCa != null && x.TruongCa == TruongCa

536 │ && !(x.Deleted ?? false)

537 │ && (x.MaChiNhanh == objToken.branch));

538 │

539 │ if (dataList.Count > 0 && banThanhPham != null)

540 │ {

541 │ List<tbl\_BanThanhPham> listChitiet = dbcontext.tbl\_BanThanhPhams

542 │ .Where(x => x.MaBanThanhPham == banThanhPham.Ma

543 │ && !(x.DaXoa ?? false)).ToList();

544 │ foreach (var item in listChitiet)

545 │ {

546 │ ChiTietBotBienTinhBTP ct = new ChiTietBotBienTinhBTP();

547 │ ct.Gio = item.Gio;

548 │ ct.Phut = item.Phut;

549 │ dataList.Add(ct);

550 │ }

551 │ List<ChiTietBotBienTinhBTP> res = new List<ChiTietBotBienTinhBTP>();

552 │ foreach (var item in dataList)

553 │ {

554 │ if (res.Count > 0 && res.Any(x => x.Gio == item.Gio

555 │ && x.Phut == item.Phut)) continue;

556 │ res.Add(item);

557 │ }

558 │

559 │ return res.Count != dataList.Count;

560 │

561 │ }

562 │ return false;

563 │ }

564 │ }

565 │ }

────────────────────────────────────────────────────────────

**📄 BotBienTinhController.cs (CS, 538 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using Microsoft.EntityFrameworkCore;

3 │ using Newtonsoft.Json;

4 │ using QuanLyChatLuong.Common;

5 │ using QuanLyChatLuong.Models;

6 │ using QuanLyChatLuong.Models.APIModels;

7 │ using QuanLyChatLuong.Models.CRMEntity;

8 │ using QuanLyChatLuong.Models.StoreProcedures;

9 │ using System.Data;

10 │ using System.Globalization;

11 │

12 │ namespace QuanLyChatLuong.Controllers

13 │ {

14 │ [Route("botbientinh")]

15 │ public class BotBienTinhController : BaseController

16 │ {

17 │ private readonly QuanLyChatLuongDbContext \_dbContext;

18 │ private CrmContext crmContext { get; }

19 │

20 │ public BotBienTinhController(QuanLyChatLuongDbContext dbcontext, CrmContext crmdbcontext, MemoryCacheManager mem...

21 │ {

22 │ \_dbContext = dbcontext;

23 │ m\_MenuID = MenuID.botBienTinh;

24 │ crmContext = crmdbcontext;

25 │ }

26 │ public class BotBienTinh

27 │ {

28 │ public int? SoLieuId { get; set; }

29 │ public string? eMa { get; set; }

30 │ public string? ListCT { get; set; }

31 │ public string? TruongCa { get; set; }

32 │ public string? TenSanPham { get; set; }

33 │ public string? DonHang { get; set; }

34 │ public string? MaLo { get; set; }

35 │ public string? Lo { get; set; }

36 │

37 │ public string? QC { get; set; }

38 │ public string? ThuKhoBot { get; set; }

39 │ public string? TruongBPTP { get; set; }

40 │ public string? DotLo { get; set; }

41 │ public string? Status { get; set; }

42 │ public string? Ngay { get; set; }

43 │ }

44 │ public class ChiTiet

45 │ {

46 │ public int? CT { get; set; }

47 │ public int? Ma { get; set; }

48 │ public string? Gio { get; set; }

49 │ public string? Phut { get; set; }

50 │ public string? SanPham { get; set; }

51 │ public string? Note { get; set; }

52 │ public string? Whiteness { get; set; }

53 │ public string? Moisture { get; set; }

54 │ public string? Fineness { get; set; }

55 │ public string? Ph { get; set; }

56 │ public string? Residue { get; set; }

57 │ public string? DS { get; set; }

58 │ public string? ViscosityPeakBU { get; set; }

59 │ public string? ViscosityEndPointBU { get; set; }

60 │ public string? ViscosityMax950C0MinBU { get; set; }

61 │ public string? ViscosityMax950C10MinBU { get; set; }

62 │ public string? AcetylGroup { get; set; }

63 │ public string? Conductivity { get; set; }

64 │ public string? GelatinnizationTemperature { get; set; }

65 │

66 │ public string? ForeignMatter { get; set; }

67 │ public string? Sensory { get; set; }

68 │ public string? SO2 { get; set; }

69 │ public string? StarchContent { get; set; }

70 │ }

71 │

72 │ public class RequestBody

73 │ {

74 │ public string TuNgay { get; set; }

75 │ public string DenNgay { get; set; }

76 │ }

77 │

78 │ [HttpGet]

79 │ [TokenUserCheckHTTP]

80 │ public IActionResult Index()

81 │ {

82 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permission...

83 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permissio...

84 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permiss...

85 │ return View();

86 │ }

87 │

88 │ [HttpPost("getsolieutheongay")]

89 │ [TokenUserCheckHTTP]

90 │ public IActionResult GetSoLieuTheoNgay([FromBody] RequestBody model)

91 │ {

92 │

93 │ DateTime TuNgay = DateTime.Parse(model.TuNgay, CultureInfo.InvariantCulture);

94 │ DateTime DenNgay = DateTime.Parse(model.DenNgay, CultureInfo.InvariantCulture);

95 │

96 │ if (TuNgay > DenNgay)

97 │ {

98 │ return Json(new ResponseModel("Error", "Từ ngày không lớn hơn Đến Ngày."));

99 │ }

100 │

101 │ int current = objToken.branch;

102 │

103 │ var chiNhanh = dbcontext.tbl\_ChiNhanhs.FirstOrDefault(x => x.Ma == current);

104 │

105 │ string codeChiNhanh = chiNhanh?.Code ?? "";

106 │

107 │ var dataSLBT = dbcontext.bt\_SoLieuBienTinhs

108 │ .AsNoTracking()

109 │ .Where(x => !(x.DaXoa ?? false)

110 │ && x.Ngay != null

111 │ && (x.Ngay.Value.Date >= TuNgay.Date && x.Ngay.Value.Date <= DenNgay.Date)

112 │ && x.MaChiNhanh == current)

113 │ .OrderByDescending(x => x.Ngay)

114 │ .ToList()

115 │ .Select(t => new

116 │ {

117 │ Ma = t.Ma,

118 │ eMa = t.Ma.Encrypt(),

119 │ Ngay = t.Ngay,

120 │ TruongCa = t.TruongCa,

121 │ QC = t.QC,

122 │ ThuKhoBot = t.ThuKhoBot,

123 │ TruongBPTP = t.TruongBPTP,

124 │ DotLo = t.DotLo,

125 │ TrangThai = t.TrangThai,

126 │

127 │ }).ToList();

128 │

129 │ if (dataSLBT.Count > 0)

130 │ {

131 │ List<bt\_ChiTietSoLieu> dataCT = new List<bt\_ChiTietSoLieu>();

132 │

133 │ foreach (var item in dataSLBT)

134 │ {

135 │ dataCT.AddRange(dbcontext.bt\_ChiTietSoLieus

136 │ .Where(x => x.MaSoLieu == item.Ma && !(x.DaXoa ?? false))

137 │ .OrderByDescending(x => x.NgayTao)

138 │ .ToList());

139 │ }

140 │

141 │ var listLoThuMua = (from l in crmContext.CloseChotLo.Where(t => !(t.Deleted ?? false) && (t.TrangThai !=...

142 │ join cn in crmContext.SysBranch.Where(t => t.Code == codeChiNhanh && !(t.Deleted ?? ...

143 │ orderby l.CreatedDate

144 │ select new

145 │ {

146 │ malo = l.Ma,

147 │ codelo = l.MaLo + "",

148 │ }).ToList();

149 │

150 │ return Json(new ResponseModel(QLCLConst.OK, "", new

151 │ {

152 │ ListSolieu = dataSLBT,

153 │ ListCT = dataCT,

154 │ ListLo = listLoThuMua

155 │ }));

156 │ }

157 │

158 │ return Json(new ResponseModel(QLCLConst.OK, "", new

159 │ {

160 │ ListSolieu = dataSLBT,

161 │ }));

162 │ }

163 │

164 │ [HttpGet("add")]

165 │ [TokenUserCheckHTTP]

166 │ public IActionResult Add()

167 │ {

168 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permission...

169 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permissio...

170 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permiss...

171 │ return View("Form");

172 │ }

173 │

174 │ [HttpGet("update")]

175 │ [TokenUserCheckHTTP(Permission.Edit)]

176 │ public IActionResult Edit(string Ma)

177 │ {

178 │ var dataSL = dbcontext.bt\_SoLieuBienTinhs.Find(Cast.ToInt(Ma.Decrypt));

179 │ if (dataSL == null) return NotFound("Bot bien tinh not found");

180 │ var ss = dbcontext.bt\_ChiTietSoLieus

181 │ .Where(x => x.MaSoLieu == Cast.ToInt(Ma.Decrypt())

182 │ && !(x.DaXoa ?? false))

183 │ .ToList();

184 │

185 │ ViewData["SoLieu"] = dataSL;

186 │ ViewData["ChiTiet"] = ss;

187 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permission...

188 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permissio...

189 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permiss...

190 │

191 │ return View("Form");

192 │ }

193 │

194 │ [HttpPost("updateapi")]

195 │ [TokenUserCheckHTTP(Permission.Edit)]

196 │ public async Task<IActionResult> UpdateAPIAsync([FromBody] BotBienTinh model)

197 │ {

198 │ var idUser = objToken.userid;

199 │ var idBranch = objToken.branch;

200 │

201 │ if (model.SoLieuId == null)

202 │ {

203 │ return Json(new ResponseModel("Error", "Lỗi không xác định được Số liệu."));

204 │ }

205 │ if (idUser == 0 || idBranch == 0)

206 │ {

207 │ return Json(new ResponseModel("Error", "Hết phiên đăng nhập. Vui lòng đăng nhập lại."));

208 │ }

209 │ if (model.TruongCa == null || model.Ngay == null || model.QC == null || model.TruongBPTP == null || model.Do...

210 │ {

211 │ return Json(Utility.MakeResponseFromEditResult("0"));

212 │ }

213 │

214 │ int result = 0;

215 │

216 │ using (var transaction = await \_dbContext.Database.BeginTransactionAsync())

217 │ {

218 │ try

219 │ {

220 │ bt\_SoLieuBienTinh? item = await \_dbContext.bt\_SoLieuBienTinhs.FindAsync(Cast.ToInt(model.SoLieuId));

221 │

222 │ item.TenSanPham = model.TenSanPham;

223 │ item.DonHang = model.DonHang;

224 │ item.MaLo = model.MaLo;

225 │ if (model.Lo != null && !model.Lo.IsEmpty())

226 │ {

227 │ item.MaLoThuMua = Cast.ToInt(model.Lo);

228 │ }

229 │ item.TruongCa = model.TruongCa;

230 │ item.ThuKhoBot = model.ThuKhoBot;

231 │ item.TruongBPTP = model.TruongBPTP;

232 │ item.DotLo = model.DotLo;

233 │ item.QC = model.QC;

234 │ item.TrangThai = model.Status == "1";

235 │ item.NguoiSua = Cast.ToInt(idUser);

236 │ item.NgaySua = DateTime.Now;

237 │

238 │ \_dbContext.bt\_SoLieuBienTinhs.Update(item);

239 │ await \_dbContext.SaveChangesAsync();

240 │

241 │ if (!string.IsNullOrEmpty(model.ListCT) && model.ListCT.Length > 3)

242 │ {

243 │ try

244 │ {

245 │ List<ChiTiet> dataList = JsonConvert.DeserializeObject<List<ChiTiet>>(model.ListCT) ?? [];

246 │

247 │ List<bt\_ChiTietSoLieu> chitietSLCu = await \_dbContext.bt\_ChiTietSoLieus

248 │ .Where(x => x.MaSoLieu == item.Ma && !(x.DaXoa ?? false))

249 │ .ToListAsync();

250 │

251 │ foreach (var ct in dataList)

252 │ {

253 │ bt\_ChiTietSoLieu? chitiet = await \_dbContext.bt\_ChiTietSoLieus.FindAsync(ct.Ma);

254 │

255 │ if (chitiet == null)

256 │ {

257 │ chitiet = new bt\_ChiTietSoLieu();

258 │ chitiet.MaSoLieu = item.Ma;

259 │ }

260 │

261 │ chitiet.Gio = ct.Gio;

262 │ chitiet.Phut = ct.Phut;

263 │

264 │ chitiet.AcetylGroup = ct.AcetylGroup.ToDecimal() == 0 ? null : ct.AcetylGroup.ToDecimal(...

265 │ chitiet.pHValue = ct.Ph.ToDecimal() == 0 ? null : ct.Ph.ToDecimal();

266 │ chitiet.Whiteness = ct.Whiteness.ToDecimal() == 0 ? null : ct.Whiteness.ToDecimal();

267 │ chitiet.Moisture = ct.Moisture.ToDecimal() == 0 ? null : ct.Moisture.ToDecimal();

268 │ chitiet.Fineness = ct.Fineness.ToDecimal() == 0 ? null : ct.Fineness.ToDecimal();

269 │

270 │ chitiet.ViscosityEndPointBU = ct.ViscosityEndPointBU.ToDecimal() == 0 ? null : ct.Viscos...

271 │ chitiet.ViscosityMax950C0MinBU = ct.ViscosityMax950C0MinBU.ToDecimal() == 0 ? null : ct....

272 │ chitiet.ViscosityMax950C10MinBU = ct.ViscosityMax950C10MinBU.ToDecimal() == 0 ? null : c...

273 │ chitiet.ViscosityPeakBU = ct.ViscosityPeakBU.ToDecimal() == 0 ? null : ct.ViscosityPeakB...

274 │

275 │ chitiet.GelatinnizationTemperature = ct.GelatinnizationTemperature.ToDecimal() == 0 ? nu...

276 │ chitiet.DS = ct.DS.ToDecimal() == 0 ? null : ct.DS.ToDecimal();

277 │ chitiet.Conductivity = ct.Conductivity.ToDecimal() == 0 ? null : ct.Conductivity.ToDecim...

278 │ chitiet.Residue = ct.Residue.ToDecimal() == 0 ? null : ct.Residue.ToDecimal();

279 │ chitiet.SO2 = ct.SO2.ToDecimal() == 0 ? null : ct.SO2.ToDecimal();

280 │ chitiet.StarchContent = ct.StarchContent.ToDecimal() == 0 ? null : ct.StarchContent.ToDe...

281 │

282 │ chitiet.Note = ct.Note;

283 │ chitiet.ForeignMatter = ct.ForeignMatter;

284 │ chitiet.Sensory = ct.Sensory;

285 │

286 │ chitiet.NguoiSua = Cast.ToInt(idUser);

287 │ chitiet.NgaySua = DateTime.Now;

288 │

289 │ \_dbContext.bt\_ChiTietSoLieus.Update(chitiet);

290 │ await \_dbContext.SaveChangesAsync();

291 │

292 │ int MaChitiet = chitiet.Ma;

293 │ if (chitietSLCu.Any(x => x.Ma == MaChitiet))

294 │ {

295 │ chitietSLCu.RemoveAll(x => x.Ma == MaChitiet);

296 │ }

297 │ }

298 │

299 │ if (chitietSLCu.Count > 0)

300 │ {

301 │ foreach (var item1 in chitietSLCu)

302 │ {

303 │ item1.DaXoa = true;

304 │ item1.NguoiXoa = Cast.ToInt(idUser);

305 │ item1.NgayXoa = DateTime.Now;

306 │ \_dbContext.bt\_ChiTietSoLieus.Update(item1);

307 │ }

308 │ await \_dbContext.SaveChangesAsync();

309 │ }

310 │ }

311 │ catch (Exception e)

312 │ {

313 │ return Json(new ResponseModel("Error", e.Message));

314 │ }

315 │ }

316 │ await transaction.CommitAsync();

317 │ result = 1;

318 │ }

319 │ catch (Exception ex)

320 │ {

321 │ // Nếu có lỗi, rollback transaction

322 │ await transaction.RollbackAsync();

323 │ return Json(new ResponseModel("Error", ex.Message));

324 │ }

325 │ }

326 │ memoryCache.Clear(t => t.Contains(QLCLConst.cacheKeyMenu));

327 │ return Json(Utility.MakeResponseFromEditResult(Cast.ToString(result)));

328 │

329 │ }

330 │

331 │ [HttpPost("addapi")]

332 │ [TokenUserCheckHTTP(Permission.Add)]

333 │ public async Task<IActionResult> AddAPIAsync([FromBody] BotBienTinh model)

334 │ {

335 │ var idUser = objToken.userid;

336 │ var idBranch = objToken.branch;

337 │ DateTime ngayNhap = DateTime.Parse(model.Ngay, CultureInfo.InvariantCulture);

338 │

339 │ if (idUser == 0 || idBranch == 0)

340 │ {

341 │ return Json(new ResponseModel("Error", "Hết phiên đăng nhập. Vui lòng đăng nhập lại."));

342 │ }

343 │

344 │ if (model.TenSanPham == null ||

345 │ model.DonHang == null ||

346 │ model.MaLo == null ||

347 │ model.Lo == null ||

348 │ model.TruongCa == null ||

349 │ model.Ngay == null ||

350 │ model.QC == null ||

351 │ model.TruongBPTP == null ||

352 │ model.DotLo == null ||

353 │ model.ThuKhoBot == null)

354 │ {

355 │ return Json(Utility.MakeResponseFromAddResult("0"));

356 │ }

357 │

358 │ int result = 0;

359 │

360 │ using (var transaction = await \_dbContext.Database.BeginTransactionAsync())

361 │ {

362 │ try

363 │ {

364 │ bt\_SoLieuBienTinh item = new bt\_SoLieuBienTinh();

365 │ item.Ngay = ngayNhap;

366 │

367 │ item.TenSanPham = model.TenSanPham;

368 │ item.DonHang = model.DonHang;

369 │ item.MaLo = model.MaLo;

370 │ item.MaLoThuMua = Cast.ToInt(model.Lo);

371 │

372 │ item.TruongCa = model.TruongCa;

373 │ item.ThuKhoBot = model.ThuKhoBot;

374 │ item.TruongBPTP = model.TruongBPTP;

375 │ item.DotLo = model.DotLo;

376 │ item.QC = model.QC;

377 │ item.TrangThai = model.Status == "1";

378 │ item.NguoiTao = Cast.ToInt(idUser);

379 │ item.NgayTao = DateTime.Now;

380 │ item.MaChiNhanh = Cast.ToInt(idBranch);

381 │

382 │ // Thêm dữ liệu vào bảng

383 │ \_dbContext.bt\_SoLieuBienTinhs.Add(item);

384 │ await \_dbContext.SaveChangesAsync();

385 │

386 │ // Nếu có danh sách chi tiết, thêm dữ liệu vào bảng ChiTiet

387 │ if (!string.IsNullOrEmpty(model.ListCT) && model.ListCT.Length > 2)

388 │ {

389 │ try

390 │ {

391 │ List<ChiTiet> dataList = JsonConvert.DeserializeObject<List<ChiTiet>>(model.ListCT) ?? [];

392 │

393 │ foreach (var ct in dataList)

394 │ {

395 │ bt\_ChiTietSoLieu chitiet = new bt\_ChiTietSoLieu();

396 │

397 │ chitiet.Gio = ct.Gio;

398 │ chitiet.Phut = ct.Phut;

399 │ chitiet.MaSoLieu = item.Ma;

400 │

401 │ chitiet.AcetylGroup = ct.AcetylGroup.ToDecimal() == 0 ? null : ct.AcetylGroup.ToDecimal(...

402 │ chitiet.pHValue = ct.Ph.ToDecimal() == 0 ? null : ct.Ph.ToDecimal();

403 │ chitiet.Whiteness = ct.Whiteness.ToDecimal() == 0 ? null : ct.Whiteness.ToDecimal();

404 │ chitiet.Moisture = ct.Moisture.ToDecimal() == 0 ? null : ct.Moisture.ToDecimal();

405 │ chitiet.Fineness = ct.Fineness.ToDecimal() == 0 ? null : ct.Fineness.ToDecimal();

406 │ chitiet.Residue = ct.Residue.ToDecimal() == 0 ? null : ct.Residue.ToDecimal();

407 │

408 │ chitiet.ViscosityEndPointBU = ct.ViscosityEndPointBU.ToDecimal() == 0 ? null : ct.Viscos...

409 │ chitiet.ViscosityMax950C0MinBU = ct.ViscosityMax950C0MinBU.ToDecimal() == 0 ? null : ct....

410 │ chitiet.ViscosityMax950C10MinBU = ct.ViscosityMax950C10MinBU.ToDecimal() == 0 ? null : c...

411 │ chitiet.ViscosityPeakBU = ct.ViscosityPeakBU.ToDecimal() == 0 ? null : ct.ViscosityPeakB...

412 │

413 │ chitiet.GelatinnizationTemperature = ct.GelatinnizationTemperature.ToDecimal() == 0 ? nu...

414 │ chitiet.DS = ct.DS.ToDecimal() == 0 ? null : ct.DS.ToDecimal();

415 │ chitiet.Conductivity = ct.Conductivity.ToDecimal() == 0 ? null : ct.Conductivity.ToDecim...

416 │

417 │ chitiet.SO2 = ct.SO2.ToInt() == 0 ? null : ct.SO2.ToInt();

418 │ chitiet.StarchContent = ct.StarchContent.ToInt() == 0 ? null : ct.StarchContent.ToInt();

419 │

420 │ chitiet.Sensory = ct.Sensory;

421 │ chitiet.ForeignMatter = ct.ForeignMatter;

422 │ chitiet.Note = ct.Note;

423 │

424 │ chitiet.NguoiTao = Cast.ToInt(idUser);

425 │ chitiet.NgayTao = DateTime.Now;

426 │

427 │ \_dbContext.bt\_ChiTietSoLieus.Add(chitiet);

428 │ await \_dbContext.SaveChangesAsync();

429 │ }

430 │ }

431 │ catch (Exception e)

432 │ {

433 │ return Json(new ResponseModel("Error", e.Message));

434 │ }

435 │ }

436 │ await transaction.CommitAsync();

437 │ result = 1;

438 │ }

439 │ catch (Exception ex)

440 │ {

441 │ // Nếu có lỗi, rollback transaction

442 │ await transaction.RollbackAsync();

443 │ return Json(new ResponseModel("Error", ex.Message));

444 │ }

445 │ }

446 │

447 │ return Json(Utility.MakeResponseFromAddResult(Cast.ToString(result)));

448 │ }

449 │

450 │ [HttpPost("deleteapi")]

451 │ [TokenUserCheckHTTP(Permission.Delete)]

452 │ public IActionResult DeleteAPI([FromBody] BotBienTinh model)

453 │ {

454 │ int result = 0;

455 │ bt\_SoLieuBienTinh? item = dbcontext.bt\_SoLieuBienTinhs.Find(Cast.ToInt(model.eMa.Decrypt()));

456 │ if (item != null)

457 │ {

458 │ try

459 │ {

460 │ item.DaXoa = true;

461 │ item.NgayXoa = DateTime.Now;

462 │ item.NguoiXoa = Cast.ToInt(objToken.userid);

463 │ dbcontext.bt\_SoLieuBienTinhs.Update(item);

464 │ dbcontext.SaveChanges();

465 │

466 │ List<bt\_ChiTietSoLieu> chitiets = dbcontext.bt\_ChiTietSoLieus

467 │ .Where(x => x.MaSoLieu == item.Ma && !(x.DaXoa ?? false))

468 │ .ToList();

469 │

470 │ foreach (var item1 in chitiets)

471 │ {

472 │ item1.DaXoa = true;

473 │ item1.NgayXoa = DateTime.Now;

474 │ item1.NguoiXoa = Cast.ToInt(objToken.userid);

475 │ dbcontext.bt\_ChiTietSoLieus.Update(item1);

476 │ dbcontext.SaveChanges();

477 │ }

478 │ result = 1;

479 │ }

480 │ catch (Exception ex)

481 │ {

482 │ return Json(new ResponseModel("Error", ex.Message));

483 │ }

484 │

485 │ }

486 │ memoryCache.Clear(t => t.Contains(QLCLConst.cacheKeyMenu));

487 │ return Json(Utility.MakeResponseFromDeleteResult(Cast.ToString(result)));

488 │ }

489 │

490 │ [TokenUserCheckHTTP]

491 │ [HttpPost("getLoThuMua")]

492 │ public IActionResult getLoThuMua()

493 │ {

494 │ var chiNhanh = dbcontext.tbl\_ChiNhanhs.FirstOrDefault(x => x.Ma == objToken.branch);

495 │

496 │ string codeChiNhanh = chiNhanh?.Code ?? "";

497 │

498 │ var listLoThuMua = (from l in crmContext.CloseChotLo.Where(t => !(t.Deleted ?? false) && (t.TrangThai != (in...

499 │ join cn in crmContext.SysBranch.Where(t => t.Code == codeChiNhanh && !(t.Deleted ?? fals...

500 │ orderby l.CreatedDate

501 │ select new

502 │ {

503 │ malo = l.Ma,

504 │ codelo = l.MaLo + "",

505 │ }).ToList();

506 │

507 │ return Json(new ResponseModel("OK", "", listLoThuMua));

508 │ }

509 │

510 │ private bool CheckThoiGian(List<ChiTiet> dataList, DateTime ngay, string TruongCa, string QC)

511 │ {

512 │ bt\_SoLieuBienTinh? solieu = dbcontext.bt\_SoLieuBienTinhs.FirstOrDefault(x => x.Ngay != null && x.Ngay.Value....

513 │ if (dataList.Count > 0 && solieu != null)

514 │ {

515 │ List<bt\_ChiTietSoLieu> listChitiet = dbcontext.bt\_ChiTietSoLieus.Where(x => x.MaSoLieu == solieu.Ma && !...

516 │ foreach (var item in listChitiet)

517 │ {

518 │

519 │ ChiTiet ct = new ChiTiet();

520 │ ct.Gio = item.Gio;

521 │ ct.Phut = item.Phut;

522 │ ct.SanPham = item.MaSanPham.ToString();

523 │ dataList.Add(ct);

524 │ }

525 │ List<ChiTiet> res = new List<ChiTiet>();

526 │ foreach (var item in dataList)

527 │ {

528 │ if (res.Count > 0 && res.Any(x => x.Gio == item.Gio && x.Phut == item.Phut && x.SanPham == item.SanP...

529 │ res.Add(item);

530 │ }

531 │

532 │ return res.Count != dataList.Count;

533 │

534 │ }

535 │ return false;

536 │ }

537 │ }

538 │ }

────────────────────────────────────────────────────────────

**📄 BotTuNhienController.cs (CS, 431 lines)**

1 │ ﻿using QuanLyChatLuong.Common;

2 │ using QuanLyChatLuong.Models;

3 │ using Microsoft.AspNetCore.Mvc;

4 │ using QuanLyChatLuong.Models.APIModels;

5 │ using System.Globalization;

6 │ using Newtonsoft.Json;

7 │ using static QuanLyChatLuong.Controllers.BotBienTinhController;

8 │ using Microsoft.EntityFrameworkCore;

9 │ using System;

10 │ using System.Collections.Generic;

11 │

12 │ namespace QuanLyChatLuong.Controllers

13 │ {

14 │

15 │ [Route("bottunhien")]

16 │ public class BotTuNhienController : BaseController

17 │ {

18 │ private readonly QuanLyChatLuongDbContext \_dbContext;

19 │

20 │ public BotTuNhienController(QuanLyChatLuongDbContext dbcontext, MemoryCacheManager memoryCache) : base(dbcontext...

21 │ {

22 │ \_dbContext = dbcontext;

23 │ m\_MenuID = MenuID.botTuNhien;

24 │ }

25 │

26 │ public class BotTuNhien

27 │ {

28 │ public int? BotTuNhienID { get; set; }

29 │ public string? eMa { get; set; }

30 │ public string? ListCT { get; set; }

31 │ public string? TruongCa { get; set; }

32 │ public string? QC { get; set; }

33 │ public string? ThuKhoBot { get; set; }

34 │ public string? TruongBPTP { get; set; }

35 │ public string? DotLo { get; set; }

36 │ public string? Status { get; set; }

37 │ public string Ngay { get; set; } = DateTime.Now.ToString();

38 │ }

39 │

40 │ public class ChiTietBotTuNhien

41 │ {

42 │ public int? CT { get; set; }

43 │ public int? Ma { get; set; }

44 │ public string? Gio { get; set; }

45 │ public string? Phut { get; set; }

46 │ public string? Whiteness { get; set; }

47 │ public string? Moisture { get; set; }

48 │ public string? Fineness { get; set; }

49 │ public string? Ph { get; set; }

50 │ public string? PhWater { get; set; }

51 │ public string? Residue { get; set; }

52 │ public string? ViscosityPeakBU { get; set; }

53 │ public string? ViscosityPeakCP { get; set; }

54 │ public string? GelatinnizationTemperature { get; set; }

55 │ public string? DryingTemperature { get; set; }

56 │ public string? SO2 { get; set; }

57 │ public string? ForeignMatter { get; set; }

58 │ public string? StarchContent { get; set; }

59 │ public string? Sensory { get; set; }

60 │ public string? Note { get; set; }

61 │ }

62 │

63 │ public class RequestBody

64 │ {

65 │ public string TuNgay { get; set; } = DateTime.Now.ToString();

66 │ public string DenNgay { get; set; } = DateTime.Now.ToString();

67 │ }

68 │

69 │ [HttpGet]

70 │ [TokenUserCheckHTTP]

71 │ public IActionResult Index()

72 │ {

73 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botTuNhien, (int)Permission....

74 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botTuNhien, (int)Permission...

75 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botTuNhien, (int)Permissi...

76 │ return View();

77 │ }

78 │

79 │ [HttpPost("getbottunhientheongay")]

80 │ [TokenUserCheckHTTP]

81 │ public IActionResult GetBotTuNhienTheoNgay([FromBody] RequestBody model)

82 │ {

83 │

84 │ DateTime TuNgay = DateTime.Parse(model.TuNgay, CultureInfo.InvariantCulture);

85 │ DateTime DenNgay = DateTime.Parse(model.DenNgay, CultureInfo.InvariantCulture);

86 │

87 │ if (TuNgay > DenNgay)

88 │ {

89 │ return Json(new ResponseModel("Error", "Từ ngày không lớn hơn Đến Ngày."));

90 │ }

91 │

92 │ int current = objToken.branch;

93 │

94 │ var listBotTuNhiens = dbcontext.tbl\_SoLieus

95 │ .AsNoTracking()

96 │ .Where(x => !(x.DaXoa ?? false)

97 │ && x.Ngay.Date >= TuNgay.Date

98 │ && x.Ngay.Date <= DenNgay.Date

99 │ && x.MaChiNhanh == current)

100 │ .OrderByDescending(x => x.Ngay)

101 │ .ToList()

102 │ .Select(t => new

103 │ {

104 │ Ma = t.Ma,

105 │ eMa = t.Ma.Encrypt(),

106 │ MaChiNhanh = t.MaChiNhanh,

107 │ Ngay = t.Ngay,

108 │ TruongCa = t.TruongCa,

109 │ QC = t.QC,

110 │ ThuKhoBot = t.ThuKhoBot,

111 │ TruongBPTP = t.TruongBPTP,

112 │ DotLo = t.DotLo,

113 │ TrangThai = t.TrangThai,

114 │

115 │ });

116 │

117 │ List<tbl\_ChiTietSoLieu> listChiTietBotTuNhiens = new List<tbl\_ChiTietSoLieu>();

118 │

119 │ foreach (var item in listBotTuNhiens)

120 │ {

121 │ listChiTietBotTuNhiens.AddRange(dbcontext.tbl\_ChiTietSoLieus

122 │ .Where(x => x.MaSoLieu == item.Ma && !(x.DaXoa ?? false))

123 │ .OrderByDescending(x => x.NgayTao)

124 │ .ToList());

125 │ }

126 │

127 │ return Json(new ResponseModel(QLCLConst.OK, "", new

128 │ {

129 │ ListSolieu = listBotTuNhiens,

130 │ ListCT = listChiTietBotTuNhiens,

131 │ }));

132 │ }

133 │

134 │ [HttpGet("add")]

135 │ [TokenUserCheckHTTP]

136 │ public IActionResult Add()

137 │ {

138 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botTuNhien, (int)Permission....

139 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botTuNhien, (int)Permission...

140 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botTuNhien, (int)Permissi...

141 │ return View("Form");

142 │ }

143 │

144 │ [HttpPost("addapi")]

145 │ [TokenUserCheckHTTP(Permission.Add)]

146 │ public async Task<IActionResult> AddAPIAsync([FromBody] BotTuNhien model)

147 │ {

148 │ var idUser = objToken.userid;

149 │ var idBranch = objToken.branch;

150 │ DateTime ngayNhap = DateTime.Parse(model.Ngay, CultureInfo.InvariantCulture);

151 │

152 │ if (idUser == 0 || idBranch == 0)

153 │ {

154 │ return Json(new ResponseModel("Error", "Hết phiên đăng nhập. Vui lòng đăng nhập lại."));

155 │ }

156 │

157 │ if (

158 │ model.TruongCa == null ||

159 │ model.Ngay == null ||

160 │ model.QC == null ||

161 │ model.TruongBPTP == null ||

162 │ model.DotLo == null ||

163 │ model.ThuKhoBot == null)

164 │ {

165 │ return Json(new ResponseModel("Error", "Các trường ở Bột tự nhiên không được để trố...

166 │ }

167 │

168 │ int result = 0;

169 │

170 │ using (var transaction = await \_dbContext.Database.BeginTransactionAsync())

171 │ {

172 │ try

173 │ {

174 │ tbl\_SoLieu item = new tbl\_SoLieu();

175 │

176 │ item.Ngay = ngayNhap;

177 │ item.TruongCa = model.TruongCa;

178 │ item.ThuKhoBot = model.ThuKhoBot;

179 │ item.TruongBPTP = model.TruongBPTP;

180 │ item.DotLo = model.DotLo;

181 │ item.QC = model.QC;

182 │ item.TrangThai = model.Status == "1";

183 │ item.NguoiTao = Cast.ToInt(idUser);

184 │ item.NgayTao = DateTime.Now;

185 │ item.MaChiNhanh = Cast.ToInt(idBranch);

186 │

187 │ // Thêm dữ liệu vào bảng

188 │ \_dbContext.tbl\_SoLieus.Add(item);

189 │ await \_dbContext.SaveChangesAsync();

190 │

191 │ // Nếu có danh sách chi tiết, thêm dữ liệu vào bảng ChiTiet

192 │ if (!string.IsNullOrEmpty(model.ListCT) && model.ListCT.Length > 2)

193 │ {

194 │ try

195 │ {

196 │ List<ChiTietBotTuNhien> dataList = JsonConvert.DeserializeObject<List<ChiTietBotTuNhien>>(mo...

197 │

198 │ foreach (var ct in dataList)

199 │ {

200 │ var chitiet = new tbl\_ChiTietSoLieu

201 │ {

202 │ Gio = ct.Gio,

203 │ Phut = ct.Phut,

204 │ MaSoLieu = item.Ma,

205 │ B\_DoTrang = string.IsNullOrWhiteSpace(ct.Whiteness) ? null : ct.Whiteness.ToDecimal(...

206 │ B\_DoAm = string.IsNullOrWhiteSpace(ct.Moisture) ? null : ct.Moisture.ToDecimal(),

207 │ B\_DoMin = string.IsNullOrWhiteSpace(ct.Fineness) ? null : ct.Fineness.ToDecimal(),

208 │ B\_DoPH = string.IsNullOrWhiteSpace(ct.Ph) ? null : ct.Ph.ToDecimal(),

209 │ NS\_DoPH = string.IsNullOrWhiteSpace(ct.PhWater) ? null : ct.PhWater.ToDecimal(),

210 │ B\_TapChat = string.IsNullOrWhiteSpace(ct.Residue) ? null : ct.Residue.ToDecimal(),

211 │ B\_DoKeoBU = string.IsNullOrWhiteSpace(ct.ViscosityPeakBU) ? null : ct.ViscosityPeakB...

212 │ B\_DoKeoCP = string.IsNullOrWhiteSpace(ct.ViscosityPeakCP) ? null : ct.ViscosityPeakC...

213 │ ND\_HoHoa = string.IsNullOrWhiteSpace(ct.GelatinnizationTemperature) ? null : ct.Gela...

214 │ ND\_Say = string.IsNullOrWhiteSpace(ct.DryingTemperature) ? null : ct.DryingTemperatu...

215 │ SO2 = string.IsNullOrWhiteSpace(ct.SO2) ? null : ct.SO2.ToDecimal(),

216 │ HamLuongTB = string.IsNullOrWhiteSpace(ct.StarchContent) ? null : ct.StarchContent.T...

217 │ ForeignMatter = ct.ForeignMatter,

218 │ Sensory = ct.Sensory,

219 │ GhiChu = ct.Note,

220 │ NguoiTao = Cast.ToInt(idUser),

221 │ NgayTao = DateTime.Now

222 │ };

223 │

224 │ \_dbContext.tbl\_ChiTietSoLieus.Add(chitiet);

225 │ }

226 │ await \_dbContext.SaveChangesAsync();

227 │ }

228 │ catch (Exception e)

229 │ {

230 │ return Json(new ResponseModel("Error", e.Message));

231 │ }

232 │ }

233 │ await transaction.CommitAsync();

234 │ result = 1;

235 │ }

236 │ catch (Exception ex)

237 │ {

238 │ // Nếu có lỗi, rollback transaction

239 │ await transaction.RollbackAsync();

240 │ return Json(new ResponseModel("Error", ex.Message));

241 │ }

242 │ }

243 │

244 │ return Json(Utility.MakeResponseFromAddResult(Cast.ToString(result)));

245 │ }

246 │

247 │ [HttpGet("update")]

248 │ [TokenUserCheckHTTP(Permission.Edit)]

249 │ public IActionResult Edit(string Ma)

250 │ {

251 │ var dataSL = dbcontext.tbl\_SoLieus.Find(Cast.ToInt(Ma.Decrypt()));

252 │ if (dataSL == null) return NotFound("Bot tu nhien not found");

253 │ var ss = dbcontext.tbl\_ChiTietSoLieus

254 │ .Where(x => x.MaSoLieu == Cast.ToInt(Ma.Decrypt())

255 │ && !(x.DaXoa ?? false))

256 │ .ToList();

257 │

258 │ ViewData["SoLieu"] = dataSL;

259 │ ViewData["ChiTiet"] = ss;

260 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botTuNhien, (int)Permission....

261 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botTuNhien, (int)Permission...

262 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botTuNhien, (int)Permissi...

263 │

264 │ return View("Form");

265 │ }

266 │

267 │ [HttpPost("updateapi")]

268 │ [TokenUserCheckHTTP(Permission.Edit)]

269 │ public async Task<IActionResult> UpdateAPIAsync([FromBody] BotTuNhien model)

270 │ {

271 │ var idUser = objToken.userid;

272 │ var idBranch = objToken.branch;

273 │

274 │ if (model.BotTuNhienID == null)

275 │ {

276 │ return Json(new ResponseModel("Error", "Lỗi không xác định được Số liệu."));

277 │ }

278 │ if (idUser == 0 || idBranch == 0)

279 │ {

280 │ return Json(new ResponseModel("Error", "Hết phiên đăng nhập. Vui lòng đăng nhập lại."));

281 │ }

282 │ if (model.TruongCa == null || model.Ngay == null || model.QC == null || model.TruongBPTP == null || model.Do...

283 │ {

284 │ return Json(new ResponseModel("Error", "Các trường ở Bột tự nhiên không được để trố...

285 │ }

286 │

287 │ int result = 0;

288 │

289 │ using (var transaction = await \_dbContext.Database.BeginTransactionAsync())

290 │ {

291 │ try

292 │ {

293 │ tbl\_SoLieu? item = await \_dbContext.tbl\_SoLieus.FindAsync(Cast.ToInt(model.BotTuNhienID));

294 │

295 │ item.TruongCa = model.TruongCa;

296 │ item.ThuKhoBot = model.ThuKhoBot;

297 │ item.TruongBPTP = model.TruongBPTP;

298 │ item.DotLo = model.DotLo;

299 │ item.QC = model.QC;

300 │ item.TrangThai = model.Status == "1";

301 │ item.NguoiSua = Cast.ToInt(idUser);

302 │ item.NgaySua = DateTime.Now;

303 │

304 │ \_dbContext.tbl\_SoLieus.Update(item);

305 │ await \_dbContext.SaveChangesAsync();

306 │

307 │ if (!string.IsNullOrEmpty(model.ListCT) && model.ListCT.Length > 3)

308 │ {

309 │ try

310 │ {

311 │ List<ChiTietBotTuNhien> dataList = JsonConvert.DeserializeObject<List<ChiTietBotTuNhien>>(mo...

312 │

313 │ List<tbl\_ChiTietSoLieu> chitietSLCu = await \_dbContext.tbl\_ChiTietSoLieus

314 │ .Where(x => x.MaSoLieu == item.Ma && !(x.DaXoa ?? false))

315 │ .ToListAsync();

316 │

317 │ foreach (var ct in dataList)

318 │ {

319 │ tbl\_ChiTietSoLieu? chitiet = await \_dbContext.tbl\_ChiTietSoLieus.FindAsync(ct.Ma);

320 │

321 │ if (chitiet == null)

322 │ {

323 │ chitiet = new tbl\_ChiTietSoLieu();

324 │ chitiet.MaSoLieu = item.Ma;

325 │ }

326 │

327 │ chitiet.Gio = ct.Gio;

328 │ chitiet.Phut = ct.Phut;

329 │ chitiet.B\_DoTrang = string.IsNullOrWhiteSpace(ct.Whiteness) ? null : ct.Whiteness.ToDeci...

330 │ chitiet.B\_DoAm = string.IsNullOrWhiteSpace(ct.Moisture) ? null : ct.Moisture.ToDecimal()...

331 │ chitiet.B\_DoMin = string.IsNullOrWhiteSpace(ct.Fineness) ? null : ct.Fineness.ToDecimal(...

332 │ chitiet.NS\_DoPH = string.IsNullOrWhiteSpace(ct.PhWater) ? null : ct.PhWater.ToDecimal();

333 │ chitiet.B\_DoPH = string.IsNullOrWhiteSpace(ct.Ph) ? null : ct.Ph.ToDecimal();

334 │ chitiet.B\_TapChat = string.IsNullOrWhiteSpace(ct.Residue) ? null : ct.Residue.ToDecimal(...

335 │ chitiet.B\_DoKeoBU = string.IsNullOrWhiteSpace(ct.ViscosityPeakBU) ? null : ct.ViscosityP...

336 │ chitiet.B\_DoKeoCP = string.IsNullOrWhiteSpace(ct.ViscosityPeakCP) ? null : ct.ViscosityP...

337 │ chitiet.ND\_HoHoa = string.IsNullOrWhiteSpace(ct.GelatinnizationTemperature) ? null : ct....

338 │ chitiet.ND\_Say = string.IsNullOrWhiteSpace(ct.DryingTemperature) ? null : ct.DryingTempe...

339 │ chitiet.SO2 = string.IsNullOrWhiteSpace(ct.SO2) ? null : ct.SO2.ToDecimal();

340 │ chitiet.HamLuongTB = string.IsNullOrWhiteSpace(ct.StarchContent) ? null : ct.StarchConte...

341 │ chitiet.ForeignMatter = ct.ForeignMatter;

342 │ chitiet.Sensory = ct.Sensory;

343 │ chitiet.GhiChu = ct.Note;

344 │ chitiet.NguoiSua = Cast.ToInt(idUser);

345 │ chitiet.NgaySua = DateTime.Now;

346 │

347 │ \_dbContext.tbl\_ChiTietSoLieus.Update(chitiet);

348 │ await \_dbContext.SaveChangesAsync();

349 │

350 │ int MaChitiet = chitiet.Ma;

351 │ if (chitietSLCu.Any(x => x.Ma == MaChitiet))

352 │ {

353 │ chitietSLCu.Remove(chitietSLCu.First(x => x.Ma == MaChitiet));

354 │ }

355 │ }

356 │

357 │ if (chitietSLCu.Count > 0)

358 │ {

359 │ foreach (var item1 in chitietSLCu)

360 │ {

361 │ item1.DaXoa = true;

362 │ item1.NguoiXoa = Cast.ToInt(idUser);

363 │ item1.NgayXoa = DateTime.Now;

364 │ \_dbContext.tbl\_ChiTietSoLieus.Update(item1);

365 │ await \_dbContext.SaveChangesAsync();

366 │ }

367 │ }

368 │ }

369 │ catch (Exception e)

370 │ {

371 │ return Json(new ResponseModel("Error", e.Message));

372 │ }

373 │ }

374 │ await transaction.CommitAsync();

375 │ result = 1;

376 │ }

377 │ catch (Exception ex)

378 │ {

379 │ // Nếu có lỗi, rollback transaction

380 │ await transaction.RollbackAsync();

381 │ return Json(new ResponseModel("Error", ex.Message));

382 │ }

383 │ }

384 │ memoryCache.Clear(t => t.Contains(QLCLConst.cacheKeyMenu));

385 │ return Json(Utility.MakeResponseFromEditResult(Cast.ToString(result)));

386 │

387 │ }

388 │

389 │

390 │ [HttpPost("deleteapi")]

391 │ [TokenUserCheckHTTP(Permission.Delete)]

392 │ public IActionResult DeleteAPI([FromBody] BotTuNhien model)

393 │ {

394 │ int result = 0;

395 │ tbl\_SoLieu? item = dbcontext.tbl\_SoLieus.Find(Cast.ToInt(model.eMa.Decrypt()));

396 │

397 │ if (item != null)

398 │ {

399 │ try

400 │ {

401 │ item.DaXoa = true;

402 │ item.NgayXoa = DateTime.Now;

403 │ item.NguoiXoa = Cast.ToInt(objToken.userid);

404 │ dbcontext.tbl\_SoLieus.Update(item);

405 │ dbcontext.SaveChanges();

406 │

407 │ List<tbl\_ChiTietSoLieu> chitiets = dbcontext.tbl\_ChiTietSoLieus

408 │ .Where(x => x.MaSoLieu == item.Ma && !(x.DaXoa ?? false))

409 │ .ToList();

410 │

411 │ foreach (var item1 in chitiets)

412 │ {

413 │ item1.DaXoa = true;

414 │ item1.NgayXoa = DateTime.Now;

415 │ item1.NguoiXoa = Cast.ToInt(objToken.userid);

416 │ dbcontext.tbl\_ChiTietSoLieus.Update(item1);

417 │ dbcontext.SaveChanges();

418 │ }

419 │ result = 1;

420 │ }

421 │ catch (Exception ex)

422 │ {

423 │ return Json(new ResponseModel("Error", ex.Message));

424 │ }

425 │

426 │ }

427 │ memoryCache.Clear(t => t.Contains(QLCLConst.cacheKeyMenu));

428 │ return Json(Utility.MakeResponseFromDeleteResult(Cast.ToString(result)));

429 │ }

430 │ }

431 │ }

────────────────────────────────────────────────────────────

**📄 Cast.cs (CS, 388 lines)**

1 │ ﻿using System.Globalization;

2 │ using System.Reflection;

3 │

4 │ namespace QuanLyChatLuong.Common

5 │ {

6 │ public static class Cast

7 │ {

8 │ public static string ToString(object? o)

9 │ {

10 │ return (o ?? "").ToString() ?? "";

11 │ }

12 │

13 │ public static byte ToByte(object? o)

14 │ {

15 │ byte result = 0;

16 │ if (!Utility.IsNull(o))

17 │ {

18 │ byte.TryParse(o.ToString(), out result);

19 │ }

20 │ return result;

21 │ }

22 │

23 │ public static int ToInt(object? o)

24 │ {

25 │ return (int)ToDouble(o);

26 │ }

27 │

28 │ public static float ToFloat(object? o)

29 │ {

30 │ float result = 0f;

31 │ if (!Utility.IsNull(o))

32 │ {

33 │ float.TryParse(o.ToString(), out result);

34 │ }

35 │ return result;

36 │ }

37 │

38 │ public static double ToDouble(object? o)

39 │ {

40 │ double result = 0.0;

41 │ if (!Utility.IsNull(o))

42 │ {

43 │ double.TryParse(o.ToString(), out result);

44 │ }

45 │ return result;

46 │ }

47 │

48 │ public static decimal ToDecimal(object? o)

49 │ {

50 │ decimal result = 0m;

51 │ if (!Utility.IsNull(o))

52 │ {

53 │ decimal.TryParse(o.ToString(), out result);

54 │ }

55 │ return result;

56 │ }

57 │

58 │ public static decimal ToDecimalVietNam(string input)

59 │ {

60 │ if (string.IsNullOrWhiteSpace(input))

61 │ return 0;

62 │

63 │ try

64 │ {

65 │ return decimal.Parse(input, new CultureInfo("vi-VN"));

66 │ }

67 │ catch

68 │ {

69 │ return 0;

70 │ }

71 │ }

72 │ /// <summary>

73 │ /// Định dạng số decimal thành chuỗi kiểu Việt Nam với số chữ số thập phân cố định...

74 │ /// </summary>

75 │ /// <param name="value">Giá trị decimal? cần định dạng.</param>

76 │ /// <param name="decimalPlaces">Số chữ số thập phân mong muốn (sau dấu phẩy).</param>

77 │ /// <returns>Chuỗi đã định dạng hoặc chuỗi rỗng nếu giá trị là null.</returns>

78 │ public static string FormatDecimalVietnamese(decimal? value, int decimalPlaces)

79 │ {

80 │ if (value == null)

81 │ {

82 │ return ""; // Hoặc trả về null, hoặc "0,00..." tùy yêu cầu

83 │ }

84 │

85 │ // Đảm bảo số thập phân không âm

86 │ if (decimalPlaces < 0)

87 │ {

88 │ decimalPlaces = 0;

89 │ }

90 │

91 │ // Tạo đối tượng CultureInfo cho tiếng Việt

92 │ CultureInfo vietnameseCulture = new CultureInfo("vi-VN");

93 │

94 │ // Tạo chuỗi định dạng "F" + số chữ số thập phân

95 │ // Ví dụ: decimalPlaces = 3 -> formatString = "F3"

96 │ string formatString = "F" + decimalPlaces.ToString();

97 │

98 │ // Gọi ToString với định dạng và văn hóa cụ thể

99 │ // value.Value vì value là nullable (decimal?)

100 │ return value.Value.ToString(formatString, vietnameseCulture);

101 │ }

102 │

103 │ public static bool ToBoolean(object? o)

104 │ {

105 │ bool result = false;

106 │ if (!Utility.IsNull(o))

107 │ {

108 │ bool.TryParse(o.ToString(), out result);

109 │ }

110 │ return result;

111 │ }

112 │

113 │ public static DateTime ToDateTime(object? o)

114 │ {

115 │ DateTime result = new DateTime(2000, 1, 1);

116 │ if (!Utility.IsNull(o))

117 │ {

118 │ DateTime.TryParse(o.ToString(), out result);

119 │ }

120 │ return result;

121 │ }

122 │

123 │ public static DateTime ToDateTime(string dateTime, string Format)

124 │ {

125 │ DateTime result = new DateTime(2000, 1, 1);

126 │ try

127 │ {

128 │ if (!Utility.IsNull(dateTime))

129 │ {

130 │ return DateTime.ParseExact(dateTime, Format, new CultureInfo("en-US"), DateTimeStyles.None);

131 │ }

132 │ }

133 │ catch

134 │ {

135 │ }

136 │ return result;

137 │ }

138 │ public static DateTime? ToDateTime2(this object o, DateTime? df = null)

139 │ {

140 │ string tmp = (o + "");

141 │ if (tmp.IsEmpty())

142 │ {

143 │ return df;

144 │ }

145 │ try

146 │ {

147 │ DateTime x;

148 │ if (tmp.Contains("T") || tmp.Contains(" "))//Datetime

149 │ {

150 │ if (DateTime.TryParseExact(tmp, tmp.Contains("-") ? "yyyy-MM-ddTHH:mm:ss" : "dd/MM/yyyy HH:mm:ss", n...

151 │ {

152 │ return x;

153 │ }

154 │ if (DateTime.TryParseExact(tmp, tmp.Contains("-") ? "yyyy-MM-ddTHH:mm" : "dd/MM/yyyyTHH:mm", null, D...

155 │ {

156 │ return x;

157 │ }

158 │ if (DateTime.TryParseExact(tmp, tmp.Contains("-") ? "yyyy-MM-dd HH:mm" : "dd/MM/yyyy HH:mm", null, D...

159 │ {

160 │ return x;

161 │ }

162 │ return df;

163 │ }

164 │ else//Date

165 │ {

166 │ if (tmp.Contains("-") || tmp.Contains("/"))

167 │ {

168 │ if (DateTime.TryParseExact(tmp, tmp.Contains("-") ? "yyyy-MM-dd" : "dd/MM/yyyy", null, DateTimeS...

169 │ {

170 │ return x;

171 │ }

172 │ return df;

173 │ }

174 │ else

175 │ {

176 │ if (Cast.ToInt(tmp) > 100000)//yyyyMd hoặc dMyyyy hoặc yyyyMMdd hoặc ddMMyyyy

177 │ {

178 │ if (tmp.Length == 8) //yyyyMMdd hoặc ddMMyyyy

179 │ {

180 │ //coming soon

181 │ }

182 │ }

183 │ else if (Cast.ToInt(tmp) > 1900)//yyyy

184 │ {

185 │ return new DateTime(Cast.ToInt(o + ""), 1, 1);

186 │ }

187 │ }

188 │ }

189 │ }

190 │ catch { }

191 │ return df;

192 │ }

193 │ public static TimeSpan ToTimeSpan(object o)

194 │ {

195 │ TimeSpan result = new TimeSpan(0L);

196 │ if (!Utility.IsNull(o))

197 │ {

198 │ TimeSpan.TryParse(o.ToString(), out result);

199 │ }

200 │ return result;

201 │ }

202 │

203 │ public static object Nz(object o, object oReplace)

204 │ {

205 │ object result;

206 │ if (!Utility.IsNull(o))

207 │ {

208 │ result = o;

209 │ }

210 │ else

211 │ {

212 │ result = oReplace;

213 │ }

214 │ return result;

215 │ }

216 │

217 │ public static object TryConvert(PropertyInfo pro, object o)

218 │ {

219 │ string text = pro.PropertyType.ToString();

220 │ object result;

221 │ if (text.Contains("Decimal"))

222 │ {

223 │ result = ToDecimal(o);

224 │ }

225 │ else if (text.Contains("Double"))

226 │ {

227 │ result = ToDouble(o);

228 │ }

229 │ else if (text.Contains("Int32"))

230 │ {

231 │ result = ToInt(o);

232 │ }

233 │ else if (text.Contains("Boolean"))

234 │ {

235 │ result = ToBoolean(o);

236 │ }

237 │ else if (text.Contains("DateTime"))

238 │ {

239 │ result = ToDateTime(o);

240 │ }

241 │ else if (text.Contains("String"))

242 │ {

243 │ result = Nz(o, "").ToString() ?? "";

244 │ }

245 │ else

246 │ {

247 │ result = o;

248 │ }

249 │ return result;

250 │ }

251 │

252 │ public static short ToShort(object o)

253 │ {

254 │ return (short)ToDouble(o);

255 │ }

256 │ }

257 │

258 │ public static class CastExtend

259 │ {

260 │ public static string ToString(this object o)

261 │ {

262 │ return Cast.ToString(o);

263 │ }

264 │

265 │ public static byte ToByte(this object o)

266 │ {

267 │ return Cast.ToByte(o);

268 │ }

269 │

270 │ public static int ToInt(this object o)

271 │ {

272 │ return Cast.ToInt(o);

273 │ }

274 │

275 │ public static float ToFloat(this object o)

276 │ {

277 │ return Cast.ToFloat(o);

278 │ }

279 │

280 │ public static double ToDouble(this object o)

281 │ {

282 │ return Cast.ToDouble(o);

283 │ }

284 │

285 │ public static decimal ToDecimal(this object o)

286 │ {

287 │ return Cast.ToDecimal(o);

288 │ }

289 │

290 │ public static bool ToBoolean(this object o)

291 │ {

292 │ return Cast.ToBoolean(o);

293 │ }

294 │

295 │ public static DateTime ToDateTime(this object o)

296 │ {

297 │ return Cast.ToDateTime(o);

298 │ }

299 │ public static DateTime? ToDateTime2(this object o, DateTime? df = null)

300 │ {

301 │ string tmp = (o + "");

302 │ if (tmp.IsEmpty())

303 │ {

304 │ return df;

305 │ }

306 │ try

307 │ {

308 │ DateTime x;

309 │ if (tmp.Contains("T") || tmp.Contains(" "))//Datetime

310 │ {

311 │ if (DateTime.TryParseExact(tmp, tmp.Contains("-") ? "yyyy-MM-ddTHH:mm:ss" : "dd/MM/yyyy HH:mm:ss", n...

312 │ {

313 │ return x;

314 │ }

315 │ if (DateTime.TryParseExact(tmp, tmp.Contains("-") ? "yyyy-MM-ddTHH:mm" : "dd/MM/yyyyTHH:mm", null, D...

316 │ {

317 │ return x;

318 │ }

319 │ if (DateTime.TryParseExact(tmp, tmp.Contains("-") ? "yyyy-MM-dd HH:mm" : "dd/MM/yyyy HH:mm", null, D...

320 │ {

321 │ return x;

322 │ }

323 │ return df;

324 │ }

325 │ else//Date

326 │ {

327 │ if (tmp.Contains("-") || tmp.Contains("/"))

328 │ {

329 │ if (DateTime.TryParseExact(tmp, tmp.Contains("-") ? "yyyy-MM-dd" : "dd/MM/yyyy", null, DateTimeS...

330 │ {

331 │ return x;

332 │ }

333 │ return df;

334 │ }

335 │ else

336 │ {

337 │ if (Cast.ToInt(tmp) > 100000)//yyyyMd hoặc dMyyyy hoặc yyyyMMdd hoặc ddMMyyyy

338 │ {

339 │ if (tmp.Length == 8) //yyyyMMdd hoặc ddMMyyyy

340 │ {

341 │ //coming soon

342 │ }

343 │ }

344 │ else if (Cast.ToInt(tmp) > 1900)//yyyy

345 │ {

346 │ return new DateTime(Cast.ToInt(o + ""), 1, 1);

347 │ }

348 │ }

349 │ }

350 │ }

351 │ catch { }

352 │ return df;

353 │ }

354 │

355 │ public static bool IsToday(this DateTime? dateTime)

356 │ {

357 │ var now = DateTime.Now;

358 │ return dateTime?.Day == now.Day && dateTime?.Month == now.Month && dateTime?.Year == now.Year;

359 │ }

360 │

361 │ public static DateTime ToDateTime(this string dateTime, string Format)

362 │ {

363 │ return Cast.ToDateTime(dateTime, Format);

364 │ }

365 │

366 │ public static TimeSpan ToTimeSpan(this object o)

367 │ {

368 │ return Cast.ToTimeSpan(o);

369 │ }

370 │

371 │ public static object Nz(this object o, object oReplace)

372 │ {

373 │ return Cast.Nz(o, oReplace);

374 │ }

375 │

376 │ public static object TryConvert(this PropertyInfo pro, object o)

377 │ {

378 │ return Cast.TryConvert(pro, o);

379 │ }

380 │

381 │ public static short ToShort(object o)

382 │ {

383 │ return Cast.ToShort(o);

384 │ }

385 │

386 │

387 │ }

388 │ }

────────────────────────────────────────────────────────────

**📄 CloseChiTietChotLoBotPhe.cs (CS, 35 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CloseChiTietChotLoBotPhe

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaLo { get; set; }

11 │

12 │ public int? MaNhapXuat { get; set; }

13 │

14 │ public int? MaLoaiBot { get; set; }

15 │

16 │ public decimal? SlbotPhe { get; set; }

17 │

18 │ public decimal? HeSoQuyDoi { get; set; }

19 │

20 │ public decimal? SoLuongBotPheQuiDoi { get; set; }

21 │

22 │ public int? UserCreated { get; set; }

23 │

24 │ public DateTime? CreatedDate { get; set; }

25 │

26 │ public int? UserModified { get; set; }

27 │

28 │ public DateTime? ModifiedDate { get; set; }

29 │

30 │ public bool? Deleted { get; set; }

31 │

32 │ public int? UserDeleted { get; set; }

33 │

34 │ public DateTime? DeletedDate { get; set; }

35 │ }

────────────────────────────────────────────────────────────

**📄 CloseChiTietChotThuMua.cs (CS, 39 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CloseChiTietChotThuMua

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaLo { get; set; }

11 │

12 │ public int? MaPhieuCan { get; set; }

13 │

14 │ public int? LoaiCan { get; set; }

15 │

16 │ public int? Tlhang { get; set; }

17 │

18 │ public int? Tlsach { get; set; }

19 │

20 │ public decimal? ThanhTien { get; set; }

21 │

22 │ public decimal? ThanhTienTroGia { get; set; }

23 │

24 │ public decimal? PhiVanChuyen { get; set; }

25 │

26 │ public int? UserCreated { get; set; }

27 │

28 │ public DateTime? CreatedDate { get; set; }

29 │

30 │ public int? UserModified { get; set; }

31 │

32 │ public DateTime? ModifiedDate { get; set; }

33 │

34 │ public bool? Deleted { get; set; }

35 │

36 │ public int? UserDeleted { get; set; }

37 │

38 │ public DateTime? DeletedDate { get; set; }

39 │ }

────────────────────────────────────────────────────────────

**📄 CloseChotLo.cs (CS, 61 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CloseChotLo

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? MaLo { get; set; }

11 │

12 │ public string? MaLoThanhPham { get; set; }

13 │

14 │ public int? MaChiNhanhTapioCa { get; set; }

15 │

16 │ public string? MaChiNhanhCanMi { get; set; }

17 │

18 │ public int? UserId { get; set; }

19 │

20 │ public DateTime? NgayThuMua { get; set; }

21 │

22 │ public DateTime? NgayChotThuMua { get; set; }

23 │

24 │ public decimal? HeSoQuyDoiBotPhe { get; set; }

25 │

26 │ public DateTime? NgayChot { get; set; }

27 │

28 │ public DateTime? NgayCapNhatChotLo { get; set; }

29 │

30 │ public int? MaPhieuXuat { get; set; }

31 │

32 │ public decimal? SanLuongBotMi { get; set; }

33 │

34 │ public decimal? TiLeThuHoi { get; set; }

35 │

36 │ public decimal? TiLeThuHoiQuiDoi { get; set; }

37 │

38 │ public decimal? TiLeBotBienTinh { get; set; }

39 │

40 │ public int? GiaVon { get; set; }

41 │

42 │ public int? GiaVonTbthang { get; set; }

43 │

44 │ public string? KetQua { get; set; }

45 │

46 │ public int? TrangThai { get; set; }

47 │

48 │ public int? UserCreated { get; set; }

49 │

50 │ public DateTime? CreatedDate { get; set; }

51 │

52 │ public int? UserModified { get; set; }

53 │

54 │ public DateTime? ModifiedDate { get; set; }

55 │

56 │ public bool? Deleted { get; set; }

57 │

58 │ public int? UserDeleted { get; set; }

59 │

60 │ public DateTime? DeletedDate { get; set; }

61 │ }

────────────────────────────────────────────────────────────

**📄 CloseHeSoQuyDoiBotPhe.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CloseHeSoQuyDoiBotPhe

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaLoaiBot { get; set; }

11 │

12 │ public decimal? GiaBotThuong { get; set; }

13 │

14 │ public decimal? GiaBotPhe { get; set; }

15 │

16 │ public decimal? HeSoQuyDoi { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 Cndauky1.cs (CS, 61 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class Cndauky1

7 │ {

8 │ public string? SốHóaĐơn { get; set; }

9 │

10 │ public string? SốĐơnHàng { get; set; }

11 │

12 │ public string? MaDonHang { get; set; }

13 │

14 │ public string? MãKh { get; set; }

15 │

16 │ public string? MaKh { get; set; }

17 │

18 │ public string? MãChiNhánh { get; set; }

19 │

20 │ public string? MaChiNhanh { get; set; }

21 │

22 │ public string? ChiNhánh { get; set; }

23 │

24 │ public string? MãPhiếuXuất { get; set; }

25 │

26 │ public string? NgàyHóaĐơn { get; set; }

27 │

28 │ public string? NgàyNhậnHàng { get; set; }

29 │

30 │ public string? NgàyĐếnHạnTt { get; set; }

31 │

32 │ public string? MãTiềnTệ { get; set; }

33 │

34 │ public double? TỷGiá { get; set; }

35 │

36 │ public double? TiềnHàng { get; set; }

37 │

38 │ public double? MãThuế { get; set; }

39 │

40 │ public double? TiềnThuế { get; set; }

41 │

42 │ public double? ThànhTiền { get; set; }

43 │

44 │ public double? SốLượng { get; set; }

45 │

46 │ public double? ĐơnGiá { get; set; }

47 │

48 │ public double? PhảiThu { get; set; }

49 │

50 │ public double? ĐãThu { get; set; }

51 │

52 │ public double? CònLại { get; set; }

53 │

54 │ public double? NgàyĐiĐường { get; set; }

55 │

56 │ public double? NgàyCôngNợ { get; set; }

57 │

58 │ public string? GhiChú { get; set; }

59 │

60 │ public string? TrạngThái { get; set; }

61 │ }

────────────────────────────────────────────────────────────

**📄 Cnkhxk.cs (CS, 61 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class Cnkhxk

7 │ {

8 │ public string? SốHóaĐơn { get; set; }

9 │

10 │ public string? SốĐơnHàng { get; set; }

11 │

12 │ public string? MaDonHang { get; set; }

13 │

14 │ public string? MãKh { get; set; }

15 │

16 │ public string? MaKh { get; set; }

17 │

18 │ public string? MãChiNhánh { get; set; }

19 │

20 │ public string? MaChiNhanh { get; set; }

21 │

22 │ public string? ChiNhánh { get; set; }

23 │

24 │ public string? MãPhiếuXuất { get; set; }

25 │

26 │ public DateTime? NgàyHóaĐơn { get; set; }

27 │

28 │ public DateTime? NgàyNhậnHàng { get; set; }

29 │

30 │ public DateTime? NgàyĐếnHạnTt { get; set; }

31 │

32 │ public string? MãTiềnTệ { get; set; }

33 │

34 │ public double? TỷGiá { get; set; }

35 │

36 │ public double? TiềnHàng { get; set; }

37 │

38 │ public double? MãThuế { get; set; }

39 │

40 │ public double? TiềnThuế { get; set; }

41 │

42 │ public double? ThànhTiền { get; set; }

43 │

44 │ public double? SốLượng { get; set; }

45 │

46 │ public double? ĐơnGiá { get; set; }

47 │

48 │ public double? PhảiThu { get; set; }

49 │

50 │ public double? ĐãThu { get; set; }

51 │

52 │ public double? CònLại { get; set; }

53 │

54 │ public double? NgàyĐiĐường { get; set; }

55 │

56 │ public double? NgàyCôngNợ { get; set; }

57 │

58 │ public string? GhiChú { get; set; }

59 │

60 │ public string? TrạngThái { get; set; }

61 │ }

────────────────────────────────────────────────────────────

**📄 ContChiTietHopDong.cs (CS, 69 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class ContChiTietHopDong

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaHopDong { get; set; }

11 │

12 │ public int? MaHh { get; set; }

13 │

14 │ public int? MaDvt { get; set; }

15 │

16 │ public int? LoaiBao { get; set; }

17 │

18 │ public int? SoLuong { get; set; }

19 │

20 │ public int? SoLuongDaXuat { get; set; }

21 │

22 │ public decimal? DonGiaTaiKho { get; set; }

23 │

24 │ public decimal? BaoBiHd { get; set; }

25 │

26 │ public decimal? Bao10Kg { get; set; }

27 │

28 │ public decimal? GiaVanChuyen { get; set; }

29 │

30 │ public decimal? BocXep { get; set; }

31 │

32 │ public int? NgayCongNo { get; set; }

33 │

34 │ public decimal? ChiPhiNgayCongNo { get; set; }

35 │

36 │ public decimal? HoaHong { get; set; }

37 │

38 │ public decimal? ChiPhiHoTro { get; set; }

39 │

40 │ public decimal? DonGiaSauTinh { get; set; }

41 │

42 │ public decimal? DonGiaBan { get; set; }

43 │

44 │ public decimal? TienThue { get; set; }

45 │

46 │ public decimal? ThanhTien { get; set; }

47 │

48 │ public decimal? TongThanhTien { get; set; }

49 │

50 │ public int? MaLyDo { get; set; }

51 │

52 │ public string? GhiChu { get; set; }

53 │

54 │ public bool? Status { get; set; }

55 │

56 │ public int? UserCreated { get; set; }

57 │

58 │ public DateTime? CreatedDate { get; set; }

59 │

60 │ public int? UserModified { get; set; }

61 │

62 │ public DateTime? ModifiedDate { get; set; }

63 │

64 │ public bool? Deleted { get; set; }

65 │

66 │ public int? UserDeleted { get; set; }

67 │

68 │ public DateTime? DeletedDate { get; set; }

69 │ }

────────────────────────────────────────────────────────────

**📄 ContHopDong.cs (CS, 81 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class ContHopDong

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? LoaiHopDong { get; set; }

13 │

14 │ public int? MaKh { get; set; }

15 │

16 │ public int? MaBaoGia { get; set; }

17 │

18 │ public string? SoHopDong { get; set; }

19 │

20 │ public string? SoHopDongDoiTac { get; set; }

21 │

22 │ public DateTime? NgayHopDong { get; set; }

23 │

24 │ public int? PhapNhan { get; set; }

25 │

26 │ public DateTime? NgayBatDau { get; set; }

27 │

28 │ public DateTime? NgayKetThuc { get; set; }

29 │

30 │ public DateTime? NgayGiaHan { get; set; }

31 │

32 │ public int? MaTienTe { get; set; }

33 │

34 │ public int? MaTyGia { get; set; }

35 │

36 │ public int? TyGia { get; set; }

37 │

38 │ public bool? XuatKhau { get; set; }

39 │

40 │ public string? GhiChu { get; set; }

41 │

42 │ public string? FileDinhKem { get; set; }

43 │

44 │ public bool? DaDuyet { get; set; }

45 │

46 │ public bool? XuatHang { get; set; }

47 │

48 │ public bool? HoanThanh { get; set; }

49 │

50 │ public int? TrangThaiHopDong { get; set; }

51 │

52 │ public bool? RemindExpireDate { get; set; }

53 │

54 │ public bool? RemindSendReport { get; set; }

55 │

56 │ public int? NguoiKetThuc { get; set; }

57 │

58 │ public DateTime? NgayKetThucHd { get; set; }

59 │

60 │ public string? CangXuatDen { get; set; }

61 │

62 │ public bool? Status { get; set; }

63 │

64 │ public int? UserCreated { get; set; }

65 │

66 │ public DateTime? CreatedDate { get; set; }

67 │

68 │ public int? UserModified { get; set; }

69 │

70 │ public DateTime? ModifiedDate { get; set; }

71 │

72 │ public bool? Deleted { get; set; }

73 │

74 │ public int? UserDeleted { get; set; }

75 │

76 │ public DateTime? DeletedDate { get; set; }

77 │

78 │ public int? NguoiLienHe { get; set; }

79 │

80 │ public int? MaHopDongChinh { get; set; }

81 │ }

────────────────────────────────────────────────────────────

**📄 ContLoaiHopDong.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class ContLoaiHopDong

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public bool? Status { get; set; }

15 │

16 │ public bool? BoSungHang { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 ContPhucLucGanNhatDto.cs (CS, 15 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class ContPhucLucGanNhatDto

7 │ {

8 │ public int? MaHopDongChinh { get; set; }

9 │

10 │ public int? MaPhuLuc { get; set; }

11 │

12 │ public DateTime? CreatedDate { get; set; }

13 │

14 │ public int? RowNum { get; set; }

15 │ }

────────────────────────────────────────────────────────────

**📄 CostChiPhiBocXep.cs (CS, 37 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CostChiPhiBocXep

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaPhieuXuat { get; set; }

13 │

14 │ public DateTime? NgayChiPhi { get; set; }

15 │

16 │ public decimal? ChiPhiBoSung { get; set; }

17 │

18 │ public decimal? ThanhTien { get; set; }

19 │

20 │ public decimal? PhaiTra { get; set; }

21 │

22 │ public string? GhiChu { get; set; }

23 │

24 │ public int? UserCreated { get; set; }

25 │

26 │ public DateTime? CreatedDate { get; set; }

27 │

28 │ public int? UserModified { get; set; }

29 │

30 │ public DateTime? ModifiedDate { get; set; }

31 │

32 │ public bool? Deleted { get; set; }

33 │

34 │ public int? UserDeleted { get; set; }

35 │

36 │ public DateTime? DeletedDate { get; set; }

37 │ }

────────────────────────────────────────────────────────────

**📄 CostChiPhiDichVuNoiDia.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CostChiPhiDichVuNoiDia

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaDichVuNoiDia { get; set; }

11 │

12 │ public int? MaCont { get; set; }

13 │

14 │ public decimal? Gia { get; set; }

15 │

16 │ public int? Tu { get; set; }

17 │

18 │ public int? Den { get; set; }

19 │

20 │ public int? UserCreated { get; set; }

21 │

22 │ public DateTime? CreatedDate { get; set; }

23 │

24 │ public int? UserModified { get; set; }

25 │

26 │ public DateTime? ModifiedDate { get; set; }

27 │

28 │ public bool? Deleted { get; set; }

29 │

30 │ public int? UserDeleted { get; set; }

31 │

32 │ public DateTime? DeletedDate { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 CostChiPhiDongGoi.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CostChiPhiDongGoi

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public decimal? GiaVnd { get; set; }

13 │

14 │ public decimal? ChenhLechBao50kg { get; set; }

15 │

16 │ public decimal? ChenhLechUsd { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 CostChiPhiKhac.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CostChiPhiKhac

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public decimal? GiaVnd { get; set; }

13 │

14 │ public decimal? GiaUsd { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │

30 │ public string? CongThuc { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 CostChiPhiSx.cs (CS, 39 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CostChiPhiSx

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public DateTime? NgayNhap { get; set; }

13 │

14 │ public int? Thang { get; set; }

15 │

16 │ public int? Nam { get; set; }

17 │

18 │ public DateTime? TuNgay { get; set; }

19 │

20 │ public DateTime? DenNgay { get; set; }

21 │

22 │ public decimal? ThanhPham { get; set; }

23 │

24 │ public bool? DaKetThuc { get; set; }

25 │

26 │ public int? UserCreated { get; set; }

27 │

28 │ public DateTime? CreatedDate { get; set; }

29 │

30 │ public int? UserModified { get; set; }

31 │

32 │ public DateTime? ModifiedDate { get; set; }

33 │

34 │ public bool? Deleted { get; set; }

35 │

36 │ public int? UserDeleted { get; set; }

37 │

38 │ public DateTime? DeletedDate { get; set; }

39 │ }

────────────────────────────────────────────────────────────

**📄 CostChiTietChiPhi.cs (CS, 39 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CostChiTietChiPhi

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiPhi { get; set; }

11 │

12 │ public int? MaDanhMucChiPhi { get; set; }

13 │

14 │ public decimal? SoluongSd { get; set; }

15 │

16 │ public decimal? DonGiaTb { get; set; }

17 │

18 │ public decimal? ChiPhiTong { get; set; }

19 │

20 │ public decimal? ChiPhiTheoDv { get; set; }

21 │

22 │ public decimal? GiaThanhSx { get; set; }

23 │

24 │ public string? GhiChu { get; set; }

25 │

26 │ public int? UserCreated { get; set; }

27 │

28 │ public DateTime? CreatedDate { get; set; }

29 │

30 │ public int? UserModified { get; set; }

31 │

32 │ public DateTime? ModifiedDate { get; set; }

33 │

34 │ public bool? Deleted { get; set; }

35 │

36 │ public int? UserDeleted { get; set; }

37 │

38 │ public DateTime? DeletedDate { get; set; }

39 │ }

────────────────────────────────────────────────────────────

**📄 CostChiTietChiPhiBocXep.cs (CS, 43 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CostChiTietChiPhiBocXep

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiPhi { get; set; }

11 │

12 │ public int? MaHangHoa { get; set; }

13 │

14 │ public int? MaLoaiBao { get; set; }

15 │

16 │ public int? LoaiBao { get; set; }

17 │

18 │ public decimal? SoLuong { get; set; }

19 │

20 │ public decimal? SoLuongBao { get; set; }

21 │

22 │ public decimal? DonGia { get; set; }

23 │

24 │ public decimal? ThanhTien { get; set; }

25 │

26 │ public int? MaCtphieuXuat { get; set; }

27 │

28 │ public string? GhiChu { get; set; }

29 │

30 │ public int? UserCreated { get; set; }

31 │

32 │ public DateTime? CreatedDate { get; set; }

33 │

34 │ public int? UserModified { get; set; }

35 │

36 │ public DateTime? ModifiedDate { get; set; }

37 │

38 │ public bool? Deleted { get; set; }

39 │

40 │ public int? UserDeleted { get; set; }

41 │

42 │ public DateTime? DeletedDate { get; set; }

43 │ }

────────────────────────────────────────────────────────────

**📄 CostDanhMucChiPhiSx.cs (CS, 37 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CostDanhMucChiPhiSx

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public string? MoTa { get; set; }

13 │

14 │ public int? MaLoaiChiPhi { get; set; }

15 │

16 │ public int? MaNhomChiPhi { get; set; }

17 │

18 │ public bool? KetChuyenTuDong { get; set; }

19 │

20 │ public decimal? DonGia { get; set; }

21 │

22 │ public bool? Status { get; set; }

23 │

24 │ public int? UserCreated { get; set; }

25 │

26 │ public DateTime? CreatedDate { get; set; }

27 │

28 │ public int? UserModified { get; set; }

29 │

30 │ public DateTime? ModifiedDate { get; set; }

31 │

32 │ public bool? Deleted { get; set; }

33 │

34 │ public int? UserDeleted { get; set; }

35 │

36 │ public DateTime? DeletedDate { get; set; }

37 │ }

────────────────────────────────────────────────────────────

**📄 CostDinhMucChiPhiSx.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CostDinhMucChiPhiSx

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? Nam { get; set; }

13 │

14 │ public int? MaDanhMucChiPhiSx { get; set; }

15 │

16 │ public double? DinhMuc { get; set; }

17 │

18 │ public string? GhiChu { get; set; }

19 │

20 │ public int? UserCreated { get; set; }

21 │

22 │ public DateTime? CreatedDate { get; set; }

23 │

24 │ public int? UserModified { get; set; }

25 │

26 │ public DateTime? ModifiedDate { get; set; }

27 │

28 │ public bool? Deleted { get; set; }

29 │

30 │ public int? UserDeleted { get; set; }

31 │

32 │ public DateTime? DeletedDate { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 CostDongHoDien.cs (CS, 43 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CostDongHoDien

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaDongHoTong { get; set; }

13 │

14 │ public string? Ten { get; set; }

15 │

16 │ public string? MoTa { get; set; }

17 │

18 │ public int? MaLoaiChiPhi { get; set; }

19 │

20 │ public bool? DongHoTong { get; set; }

21 │

22 │ public string? MaKhdien { get; set; }

23 │

24 │ public string? UserLogin { get; set; }

25 │

26 │ public string? MatKhau { get; set; }

27 │

28 │ public bool? Status { get; set; }

29 │

30 │ public int? UserCreated { get; set; }

31 │

32 │ public DateTime? CreatedDate { get; set; }

33 │

34 │ public int? UserModified { get; set; }

35 │

36 │ public DateTime? ModifiedDate { get; set; }

37 │

38 │ public bool? Deleted { get; set; }

39 │

40 │ public int? UserDeleted { get; set; }

41 │

42 │ public DateTime? DeletedDate { get; set; }

43 │ }

────────────────────────────────────────────────────────────

**📄 CostGiaDien.cs (CS, 35 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CostGiaDien

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public decimal? GiaCaoDiem { get; set; }

13 │

14 │ public decimal? GiaThapDiem { get; set; }

15 │

16 │ public decimal? GiaBinhThuong { get; set; }

17 │

18 │ public DateTime? NgayApDung { get; set; }

19 │

20 │ public bool? Status { get; set; }

21 │

22 │ public int? UserCreated { get; set; }

23 │

24 │ public DateTime? CreatedDate { get; set; }

25 │

26 │ public int? UserModified { get; set; }

27 │

28 │ public DateTime? ModifiedDate { get; set; }

29 │

30 │ public bool? Deleted { get; set; }

31 │

32 │ public int? UserDeleted { get; set; }

33 │

34 │ public DateTime? DeletedDate { get; set; }

35 │ }

────────────────────────────────────────────────────────────

**📄 CostLichSuDungDien.cs (CS, 45 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CostLichSuDungDien

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public DateTime? NgaySuDung { get; set; }

13 │

14 │ public int? MaDongHo { get; set; }

15 │

16 │ public decimal? GioCaoDiem { get; set; }

17 │

18 │ public decimal? GioThapDiem { get; set; }

19 │

20 │ public decimal? GioBinhThuong { get; set; }

21 │

22 │ public decimal? GiaCaoDiem { get; set; }

23 │

24 │ public decimal? GiaThapDiem { get; set; }

25 │

26 │ public decimal? GiaBinhThuong { get; set; }

27 │

28 │ public bool? DaKetChuyen { get; set; }

29 │

30 │ public bool? SanLuongTieuThu { get; set; }

31 │

32 │ public int? UserCreated { get; set; }

33 │

34 │ public DateTime? CreatedDate { get; set; }

35 │

36 │ public int? UserModified { get; set; }

37 │

38 │ public DateTime? ModifiedDate { get; set; }

39 │

40 │ public bool? Deleted { get; set; }

41 │

42 │ public int? UserDeleted { get; set; }

43 │

44 │ public DateTime? DeletedDate { get; set; }

45 │ }

────────────────────────────────────────────────────────────

**📄 CostLoaiChiPhi.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CostLoaiChiPhi

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public string? MoTa { get; set; }

13 │

14 │ public bool? Status { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 CostNhomChiPhi.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CostNhomChiPhi

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public string? MoTa { get; set; }

13 │

14 │ public bool? Status { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 CrmContext.cs (CS, 4645 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │ using Microsoft.EntityFrameworkCore;

4 │

5 │ namespace QuanLyChatLuong.Models.CRMEntity;

6 │

7 │ public partial class CrmContext : DbContext

8 │ {

9 │ public CrmContext()

10 │ {

11 │ }

12 │

13 │ public CrmContext(DbContextOptions<CrmContext> options)

14 │ : base(options)

15 │ {

16 │ }

17 │

18 │ public virtual DbSet<AccChiTietThanhToanTheoDon> AccChiTietThanhToanTheoDon { get; set; }

19 │

20 │ public virtual DbSet<AccChuyenHinhThucTamUng> AccChuyenHinhThucTamUng { get; set; }

21 │

22 │ public virtual DbSet<AccCongNoDauKy> AccCongNoDauKy { get; set; }

23 │

24 │ public virtual DbSet<AccHinhThucThu> AccHinhThucThu { get; set; }

25 │

26 │ public virtual DbSet<AccLoaiPhieuThu> AccLoaiPhieuThu { get; set; }

27 │

28 │ public virtual DbSet<AccPhatSinhCongNo> AccPhatSinhCongNo { get; set; }

29 │

30 │ public virtual DbSet<AccSoDu131> AccSoDu131 { get; set; }

31 │

32 │ public virtual DbSet<AccSoDu131ChietKhau> AccSoDu131ChietKhau { get; set; }

33 │

34 │ public virtual DbSet<AccTaiKhoanNh> AccTaiKhoanNh { get; set; }

35 │

36 │ public virtual DbSet<AccTamUng> AccTamUng { get; set; }

37 │

38 │ public virtual DbSet<AccThanhToanTheoDon> AccThanhToanTheoDon { get; set; }

39 │

40 │ public virtual DbSet<AccTrangThaiTamUng> AccTrangThaiTamUng { get; set; }

41 │

42 │ public virtual DbSet<AccXuLyChenhLechTyGia> AccXuLyChenhLechTyGia { get; set; }

43 │

44 │ public virtual DbSet<CloseChiTietChotLoBotPhe> CloseChiTietChotLoBotPhe { get; set; }

45 │

46 │ public virtual DbSet<CloseChiTietChotThuMua> CloseChiTietChotThuMua { get; set; }

47 │

48 │ public virtual DbSet<CloseChotLo> CloseChotLo { get; set; }

49 │

50 │ public virtual DbSet<CloseHeSoQuyDoiBotPhe> CloseHeSoQuyDoiBotPhe { get; set; }

51 │

52 │ public virtual DbSet<Cndauky1> Cndauky1 { get; set; }

53 │

54 │ public virtual DbSet<Cnkhxk> Cnkhxk { get; set; }

55 │

56 │ public virtual DbSet<ContChiTietHopDong> ContChiTietHopDong { get; set; }

57 │

58 │ public virtual DbSet<ContHopDong> ContHopDong { get; set; }

59 │

60 │ public virtual DbSet<ContLoaiHopDong> ContLoaiHopDong { get; set; }

61 │

62 │ public virtual DbSet<ContPhucLucGanNhatDto> ContPhucLucGanNhatDto { get; set; }

63 │

64 │ public virtual DbSet<CostChiPhiBocXep> CostChiPhiBocXep { get; set; }

65 │

66 │ public virtual DbSet<CostChiPhiDichVuNoiDia> CostChiPhiDichVuNoiDia { get; set; }

67 │

68 │ public virtual DbSet<CostChiPhiDongGoi> CostChiPhiDongGoi { get; set; }

69 │

70 │ public virtual DbSet<CostChiPhiKhac> CostChiPhiKhac { get; set; }

71 │

72 │ public virtual DbSet<CostChiPhiSx> CostChiPhiSx { get; set; }

73 │

74 │ public virtual DbSet<CostChiTietChiPhi> CostChiTietChiPhi { get; set; }

75 │

76 │ public virtual DbSet<CostChiTietChiPhiBocXep> CostChiTietChiPhiBocXep { get; set; }

77 │

78 │ public virtual DbSet<CostDanhMucChiPhiSx> CostDanhMucChiPhiSx { get; set; }

79 │

80 │ public virtual DbSet<CostDinhMucChiPhiSx> CostDinhMucChiPhiSx { get; set; }

81 │

82 │ public virtual DbSet<CostDongHoDien> CostDongHoDien { get; set; }

83 │

84 │ public virtual DbSet<CostGiaDien> CostGiaDien { get; set; }

85 │

86 │ public virtual DbSet<CostLichSuDungDien> CostLichSuDungDien { get; set; }

87 │

88 │ public virtual DbSet<CostLoaiChiPhi> CostLoaiChiPhi { get; set; }

89 │

90 │ public virtual DbSet<CostNhomChiPhi> CostNhomChiPhi { get; set; }

91 │

92 │ public virtual DbSet<CusDoiThu> CusDoiThu { get; set; }

93 │

94 │ public virtual DbSet<DebtCongNoKh> DebtCongNoKh { get; set; }

95 │

96 │ public virtual DbSet<DebtSoDuDauKy> DebtSoDuDauKy { get; set; }

97 │

98 │ public virtual DbSet<Dmkh1> Dmkh1 { get; set; }

99 │

100 │ public virtual DbSet<GiaVonVatTu> GiaVonVatTu { get; set; }

101 │

102 │ public virtual DbSet<HisChamSocKhachHang> HisChamSocKhachHang { get; set; }

103 │

104 │ public virtual DbSet<HisLoaiThongTin> HisLoaiThongTin { get; set; }

105 │

106 │ public virtual DbSet<HisNguonThongTin> HisNguonThongTin { get; set; }

107 │

108 │ public virtual DbSet<HisNhatKyKhachHang> HisNhatKyKhachHang { get; set; }

109 │

110 │ public virtual DbSet<ImportSdSpp> ImportSdSpp { get; set; }

111 │

112 │ public virtual DbSet<ImportSoDu> ImportSoDu { get; set; }

113 │

114 │ public virtual DbSet<InvChiTietHoaDon> InvChiTietHoaDon { get; set; }

115 │

116 │ public virtual DbSet<InvHoaDon> InvHoaDon { get; set; }

117 │

118 │ public virtual DbSet<InvLoaiHoaDon> InvLoaiHoaDon { get; set; }

119 │

120 │ public virtual DbSet<InvTrangThaiHoaDon> InvTrangThaiHoaDon { get; set; }

121 │

122 │ public virtual DbSet<LocCheckIn> LocCheckIn { get; set; }

123 │

124 │ public virtual DbSet<LocHuyenThi> LocHuyenThi { get; set; }

125 │

126 │ public virtual DbSet<LocPhuongXa> LocPhuongXa { get; set; }

127 │

128 │ public virtual DbSet<LocQuocGia> LocQuocGia { get; set; }

129 │

130 │ public virtual DbSet<LocTinhThanh> LocTinhThanh { get; set; }

131 │

132 │ public virtual DbSet<MesThongBaoApp> MesThongBaoApp { get; set; }

133 │

134 │ public virtual DbSet<Nhập> Nhập { get; set; }

135 │

136 │ public virtual DbSet<NtfGroupThongBao> NtfGroupThongBao { get; set; }

137 │

138 │ public virtual DbSet<NtfMemberGroupThongBao> NtfMemberGroupThongBao { get; set; }

139 │

140 │ public virtual DbSet<OpsBaoGia> OpsBaoGia { get; set; }

141 │

142 │ public virtual DbSet<OpsChiTietBaoGia> OpsChiTietBaoGia { get; set; }

143 │

144 │ public virtual DbSet<OpsChiTietDoanhThu> OpsChiTietDoanhThu { get; set; }

145 │

146 │ public virtual DbSet<OpsDoanhThu> OpsDoanhThu { get; set; }

147 │

148 │ public virtual DbSet<OpsLichSuChiTietBaoGia> OpsLichSuChiTietBaoGia { get; set; }

149 │

150 │ public virtual DbSet<OpsThanhToan> OpsThanhToan { get; set; }

151 │

152 │ public virtual DbSet<OpsTieuChuanKyThuat> OpsTieuChuanKyThuat { get; set; }

153 │

154 │ public virtual DbSet<PlBaoCaoSanLuong> PlBaoCaoSanLuong { get; set; }

155 │

156 │ public virtual DbSet<PlChiTietChiTieuBan> PlChiTietChiTieuBan { get; set; }

157 │

158 │ public virtual DbSet<PlChiTieuBanHang> PlChiTieuBanHang { get; set; }

159 │

160 │ public virtual DbSet<PlChiTieuBanHangTheoThang> PlChiTieuBanHangTheoThang { get; set; }

161 │

162 │ public virtual DbSet<PlChiTieuSx> PlChiTieuSx { get; set; }

163 │

164 │ public virtual DbSet<PlChiTieuThuMuaCuMi> PlChiTieuThuMuaCuMi { get; set; }

165 │

166 │ public virtual DbSet<PromChietKhauThuongMai> PromChietKhauThuongMai { get; set; }

167 │

168 │ public virtual DbSet<PurChiTietDeNghi> PurChiTietDeNghi { get; set; }

169 │

170 │ public virtual DbSet<PurChiTietDonHang> PurChiTietDonHang { get; set; }

171 │

172 │ public virtual DbSet<PurChiTietHoaDon> PurChiTietHoaDon { get; set; }

173 │

174 │ public virtual DbSet<PurChiTietThanhToan> PurChiTietThanhToan { get; set; }

175 │

176 │ public virtual DbSet<PurDeNghiThanhToan> PurDeNghiThanhToan { get; set; }

177 │

178 │ public virtual DbSet<PurDonDatHang> PurDonDatHang { get; set; }

179 │

180 │ public virtual DbSet<PurHoaDon> PurHoaDon { get; set; }

181 │

182 │ public virtual DbSet<PurPhieuDeNghi> PurPhieuDeNghi { get; set; }

183 │

184 │ public virtual DbSet<PurThanhToan> PurThanhToan { get; set; }

185 │

186 │ public virtual DbSet<PurTrangThaiDonDatHang> PurTrangThaiDonDatHang { get; set; }

187 │

188 │ public virtual DbSet<PxhPhieuXuat> PxhPhieuXuat { get; set; }

189 │

190 │ public virtual DbSet<RptBaoCaoDoanhThu> RptBaoCaoDoanhThu { get; set; }

191 │

192 │ public virtual DbSet<StocChiTietDieuChuyen> StocChiTietDieuChuyen { get; set; }

193 │

194 │ public virtual DbSet<StocChiTietLenhXuatKho> StocChiTietLenhXuatKho { get; set; }

195 │

196 │ public virtual DbSet<StocChiTietNhapXuat> StocChiTietNhapXuat { get; set; }

197 │

198 │ public virtual DbSet<StocChiTietVanChuyenLenhXuatKho> StocChiTietVanChuyenLenhXuatKho { get; set; }

199 │

200 │ public virtual DbSet<StocChiTietXuatKhoTuLenh> StocChiTietXuatKhoTuLenh { get; set; }

201 │

202 │ public virtual DbSet<StocDanhMucKho> StocDanhMucKho { get; set; }

203 │

204 │ public virtual DbSet<StocGhepLo> StocGhepLo { get; set; }

205 │

206 │ public virtual DbSet<StocKhoKeToan> StocKhoKeToan { get; set; }

207 │

208 │ public virtual DbSet<StocKhoaSo> StocKhoaSo { get; set; }

209 │

210 │ public virtual DbSet<StocLenhXuatKho> StocLenhXuatKho { get; set; }

211 │

212 │ public virtual DbSet<StocLineKho> StocLineKho { get; set; }

213 │

214 │ public virtual DbSet<StocLoaiKho> StocLoaiKho { get; set; }

215 │

216 │ public virtual DbSet<StocLoaiNhapXuat> StocLoaiNhapXuat { get; set; }

217 │

218 │ public virtual DbSet<StocPhieuDieuChuyen> StocPhieuDieuChuyen { get; set; }

219 │

220 │ public virtual DbSet<StocPhieuNhapXuat> StocPhieuNhapXuat { get; set; }

221 │

222 │ public virtual DbSet<StocTonKho> StocTonKho { get; set; }

223 │

224 │ public virtual DbSet<SuaDonHang> SuaDonHang { get; set; }

225 │

226 │ public virtual DbSet<SysBranch> SysBranch { get; set; }

227 │

228 │ public virtual DbSet<SysLog> SysLog { get; set; }

229 │

230 │ public virtual DbSet<SysLogo> SysLogo { get; set; }

231 │

232 │ public virtual DbSet<SysMenu> SysMenu { get; set; }

233 │

234 │ public virtual DbSet<SysOtp> SysOtp { get; set; }

235 │

236 │ public virtual DbSet<SysParameter> SysParameter { get; set; }

237 │

238 │ public virtual DbSet<SysPermission> SysPermission { get; set; }

239 │

240 │ public virtual DbSet<SysRole> SysRole { get; set; }

241 │

242 │ public virtual DbSet<SysRoleBaoCaoBld> SysRoleBaoCaoBld { get; set; }

243 │

244 │ public virtual DbSet<SysRoleMenu> SysRoleMenu { get; set; }

245 │

246 │ public virtual DbSet<SysRolePermission> SysRolePermission { get; set; }

247 │

248 │ public virtual DbSet<SysSentEmailHistory> SysSentEmailHistory { get; set; }

249 │

250 │ public virtual DbSet<SysStatus> SysStatus { get; set; }

251 │

252 │ public virtual DbSet<SysToken> SysToken { get; set; }

253 │

254 │ public virtual DbSet<SysUser> SysUser { get; set; }

255 │

256 │ public virtual DbSet<SysUserHistory> SysUserHistory { get; set; }

257 │

258 │ public virtual DbSet<SysUserRole> SysUserRole { get; set; }

259 │

260 │ public virtual DbSet<SysUserSetting> SysUserSetting { get; set; }

261 │

262 │ public virtual DbSet<TblBaoCaoBld> TblBaoCaoBld { get; set; }

263 │

264 │ public virtual DbSet<TblCaSanXuat> TblCaSanXuat { get; set; }

265 │

266 │ public virtual DbSet<TblCangDen> TblCangDen { get; set; }

267 │

268 │ public virtual DbSet<TblCheDoKhachHang> TblCheDoKhachHang { get; set; }

269 │

270 │ public virtual DbSet<TblChiTietDuyetChungTu> TblChiTietDuyetChungTu { get; set; }

271 │

272 │ public virtual DbSet<TblChiTietQuyTrinhDuyet> TblChiTietQuyTrinhDuyet { get; set; }

273 │

274 │ public virtual DbSet<TblChiTieu> TblChiTieu { get; set; }

275 │

276 │ public virtual DbSet<TblChucVu> TblChucVu { get; set; }

277 │

278 │ public virtual DbSet<TblCont> TblCont { get; set; }

279 │

280 │ public virtual DbSet<TblDichVuNoiDia> TblDichVuNoiDia { get; set; }

281 │

282 │ public virtual DbSet<TblDonViTinh> TblDonViTinh { get; set; }

283 │

284 │ public virtual DbSet<TblEmail> TblEmail { get; set; }

285 │

286 │ public virtual DbSet<TblGiaBocXep> TblGiaBocXep { get; set; }

287 │

288 │ public virtual DbSet<TblGiaNiemYet> TblGiaNiemYet { get; set; }

289 │

290 │ public virtual DbSet<TblHangHoa> TblHangHoa { get; set; }

291 │

292 │ public virtual DbSet<TblLaiTaiChinh> TblLaiTaiChinh { get; set; }

293 │

294 │ public virtual DbSet<TblLichSuNhaVanChuyen> TblLichSuNhaVanChuyen { get; set; }

295 │

296 │ public virtual DbSet<TblLichSuTyGia> TblLichSuTyGia { get; set; }

297 │

298 │ public virtual DbSet<TblLoaiBao> TblLoaiBao { get; set; }

299 │

300 │ public virtual DbSet<TblLoaiBot> TblLoaiBot { get; set; }

301 │

302 │ public virtual DbSet<TblLoaiChungTu> TblLoaiChungTu { get; set; }

303 │

304 │ public virtual DbSet<TblLoaiHangHoa> TblLoaiHangHoa { get; set; }

305 │

306 │ public virtual DbSet<TblLyDo> TblLyDo { get; set; }

307 │

308 │ public virtual DbSet<TblLyDoDuyetDonHang> TblLyDoDuyetDonHang { get; set; }

309 │

310 │ public virtual DbSet<TblNhaCungCap> TblNhaCungCap { get; set; }

311 │

312 │ public virtual DbSet<TblNhaVanChuyen> TblNhaVanChuyen { get; set; }

313 │

314 │ public virtual DbSet<TblNhomChucVu> TblNhomChucVu { get; set; }

315 │

316 │ public virtual DbSet<TblPallet> TblPallet { get; set; }

317 │

318 │ public virtual DbSet<TblPhongBan> TblPhongBan { get; set; }

319 │

320 │ public virtual DbSet<TblQuanLySale> TblQuanLySale { get; set; }

321 │

322 │ public virtual DbSet<TblQuyTrinhDuyetChungTu> TblQuyTrinhDuyetChungTu { get; set; }

323 │

324 │ public virtual DbSet<TblSoChungTu> TblSoChungTu { get; set; }

325 │

326 │ public virtual DbSet<TblTaiKhoanNhaCc> TblTaiKhoanNhaCc { get; set; }

327 │

328 │ public virtual DbSet<TblThueVat> TblThueVat { get; set; }

329 │

330 │ public virtual DbSet<TblThuogHieu> TblThuogHieu { get; set; }

331 │

332 │ public virtual DbSet<TblTienTe> TblTienTe { get; set; }

333 │

334 │ public virtual DbSet<TblTrangThai> TblTrangThai { get; set; }

335 │

336 │ public virtual DbSet<TblTrangThaiDonHang> TblTrangThaiDonHang { get; set; }

337 │

338 │ public virtual DbSet<TblTrangThaiHopDong> TblTrangThaiHopDong { get; set; }

339 │

340 │ public virtual DbSet<TblTrangThaiLenhXuatKho> TblTrangThaiLenhXuatKho { get; set; }

341 │

342 │ public virtual DbSet<TblTyGia> TblTyGia { get; set; }

343 │

344 │ public virtual DbSet<TmpAccSoDu131> TmpAccSoDu131 { get; set; }

345 │

346 │ public virtual DbSet<TmpData> TmpData { get; set; }

347 │

348 │ public virtual DbSet<TmpData1> TmpData1 { get; set; }

349 │

350 │ public virtual DbSet<TmpDataTongHopBaoGiaVaNhatKy> TmpDataTongHopBaoGiaVaNhatKy { get; set; }

351 │

352 │ public virtual DbSet<TmpTtkh> TmpTtkh { get; set; }

353 │

354 │ public virtual DbSet<TranChiTietThanhToanVanChuyen> TranChiTietThanhToanVanChuyen { get; set; }

355 │

356 │ public virtual DbSet<TranChiTietVanChuyen> TranChiTietVanChuyen { get; set; }

357 │

358 │ public virtual DbSet<TranDanhSachXe> TranDanhSachXe { get; set; }

359 │

360 │ public virtual DbSet<TranGiaVanChuyen> TranGiaVanChuyen { get; set; }

361 │

362 │ public virtual DbSet<TranLoaiXe> TranLoaiXe { get; set; }

363 │

364 │ public virtual DbSet<TranThanhToan> TranThanhToan { get; set; }

365 │

366 │ public virtual DbSet<TranThanhToanVanChuyen> TranThanhToanVanChuyen { get; set; }

367 │

368 │ public virtual DbSet<TranVanChuyen> TranVanChuyen { get; set; }

369 │

370 │ public virtual DbSet<TtkhBangDanhGia> TtkhBangDanhGia { get; set; }

371 │

372 │ public virtual DbSet<TtkhCauHoiKh> TtkhCauHoiKh { get; set; }

373 │

374 │ public virtual DbSet<TtkhChiTietBangDanhGia> TtkhChiTietBangDanhGia { get; set; }

375 │

376 │ public virtual DbSet<TtkhChiTietTtkh> TtkhChiTietTtkh { get; set; }

377 │

378 │ public virtual DbSet<TtkhDiaChiGiaoHang> TtkhDiaChiGiaoHang { get; set; }

379 │

380 │ public virtual DbSet<TtkhDiemDanhGia> TtkhDiemDanhGia { get; set; }

381 │

382 │ public virtual DbSet<TtkhLoaiKh> TtkhLoaiKh { get; set; }

383 │

384 │ public virtual DbSet<TtkhNguoiLienHe> TtkhNguoiLienHe { get; set; }

385 │

386 │ public virtual DbSet<TtkhNguonKh> TtkhNguonKh { get; set; }

387 │

388 │ public virtual DbSet<TtkhNhomGiaoDich> TtkhNhomGiaoDich { get; set; }

389 │

390 │ public virtual DbSet<TtkhNoiDungDanhGia> TtkhNoiDungDanhGia { get; set; }

391 │

392 │ public virtual DbSet<TtkhThongTinKh> TtkhThongTinKh { get; set; }

393 │

394 │ public virtual DbSet<TtkhThongTinKhachHangNguoiLienHe> TtkhThongTinKhachHangNguoiLienHe { get; set; }

395 │

396 │ public virtual DbSet<V1> V1 { get; set; }

397 │

398 │ public virtual DbSet<VBaoCaoDanhThu> VBaoCaoDanhThu { get; set; }

399 │

400 │ public virtual DbSet<VBaoGia> VBaoGia { get; set; }

401 │

402 │ public virtual DbSet<VChotLo> VChotLo { get; set; }

403 │

404 │ public virtual DbSet<VDoanhThu> VDoanhThu { get; set; }

405 │

406 │ public virtual DbSet<VHangHoa> VHangHoa { get; set; }

407 │

408 │ public virtual DbSet<VHoaDon> VHoaDon { get; set; }

409 │

410 │ public virtual DbSet<VHoaDonMuaHang> VHoaDonMuaHang { get; set; }

411 │

412 │ public virtual DbSet<VHopDong> VHopDong { get; set; }

413 │

414 │ public virtual DbSet<VKhoHang> VKhoHang { get; set; }

415 │

416 │ public virtual DbSet<VLenhXuatKho> VLenhXuatKho { get; set; }

417 │

418 │ public virtual DbSet<VLoaiXe> VLoaiXe { get; set; }

419 │

420 │ public virtual DbSet<VNhapKho> VNhapKho { get; set; }

421 │

422 │ public virtual DbSet<VPermission> VPermission { get; set; }

423 │

424 │ public virtual DbSet<VPhieuNhapKho> VPhieuNhapKho { get; set; }

425 │

426 │ public virtual DbSet<VPhieuXuatKho> VPhieuXuatKho { get; set; }

427 │

428 │ public virtual DbSet<VPhuLucHopDongGanNhat> VPhuLucHopDongGanNhat { get; set; }

429 │

430 │ public virtual DbSet<VQlclChiNhanh> VQlclChiNhanh { get; set; }

431 │

432 │ public virtual DbSet<VQlclChiTietSoLieu> VQlclChiTietSoLieu { get; set; }

433 │

434 │ public virtual DbSet<VQlclGioiHanSoLieu> VQlclGioiHanSoLieu { get; set; }

435 │

436 │ public virtual DbSet<VQlclSoLieu> VQlclSoLieu { get; set; }

437 │

438 │ public virtual DbSet<VQuanLySale> VQuanLySale { get; set; }

439 │

440 │ public virtual DbSet<VThanhToan> VThanhToan { get; set; }

441 │

442 │ public virtual DbSet<VThongTinKhachHangCongNo> VThongTinKhachHangCongNo { get; set; }

443 │

444 │ public virtual DbSet<VTieuChuanKyThuatHd> VTieuChuanKyThuatHd { get; set; }

445 │

446 │ public virtual DbSet<VVanChuyen> VVanChuyen { get; set; }

447 │

448 │ public virtual DbSet<Xuất> Xuất { get; set; }

449 │

450 │ protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)

451 │ => optionsBuilder.UseSqlServer("Name=ConnectionStrings:CRMHungDuy");

452 │

453 │ protected override void OnModelCreating(ModelBuilder modelBuilder)

454 │ {

455 │ modelBuilder.Entity<AccChiTietThanhToanTheoDon>(entity =>

456 │ {

457 │ entity.HasKey(e => e.Ma);

458 │

459 │ entity.ToTable("acc\_ChiTietThanhToanTheoDon");

460 │

461 │ entity.Property(e => e.ChenhLechTyGia).HasColumnType("decimal(18, 4)");

462 │ entity.Property(e => e.ConLai).HasColumnType("decimal(18, 4)");

463 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

464 │ entity.Property(e => e.DaThanhToan).HasColumnType("decimal(18, 4)");

465 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

466 │ entity.Property(e => e.MaTienTeTt).HasColumnName("MaTienTeTT");

467 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

468 │ entity.Property(e => e.PhaiThu).HasColumnType("decimal(18, 4)");

469 │ entity.Property(e => e.TyGiaTt)

470 │ .HasColumnType("decimal(18, 4)")

471 │ .HasColumnName("TyGiaTT");

472 │ });

473 │

474 │ modelBuilder.Entity<AccChuyenHinhThucTamUng>(entity =>

475 │ {

476 │ entity.HasKey(e => e.Ma);

477 │

478 │ entity.ToTable("acc\_ChuyenHinhThucTamUng");

479 │

480 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

481 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

482 │ entity.Property(e => e.LyDo).HasMaxLength(250);

483 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

484 │ entity.Property(e => e.NgayChuyen).HasColumnType("datetime");

485 │ });

486 │

487 │ modelBuilder.Entity<AccCongNoDauKy>(entity =>

488 │ {

489 │ entity.HasKey(e => e.Ma);

490 │

491 │ entity.ToTable("acc\_CongNoDauKy");

492 │

493 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

494 │ entity.Property(e => e.DaThanhToan).HasColumnType("decimal(18, 4)");

495 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

496 │ entity.Property(e => e.GhiChu).HasMaxLength(300);

497 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

498 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

499 │ entity.Property(e => e.NgayHetHanTt)

500 │ .HasColumnType("datetime")

501 │ .HasColumnName("NgayHetHanTT");

502 │ entity.Property(e => e.NgayHoaDon).HasColumnType("datetime");

503 │ entity.Property(e => e.NgayThanhToan).HasColumnType("datetime");

504 │ entity.Property(e => e.NguoiThanhToan).HasMaxLength(100);

505 │ entity.Property(e => e.SoDonHang)

506 │ .HasMaxLength(30)

507 │ .IsUnicode(false);

508 │ entity.Property(e => e.SoDuNo).HasColumnType("decimal(18, 4)");

509 │ entity.Property(e => e.SoHoaDon)

510 │ .HasMaxLength(30)

511 │ .IsUnicode(false);

512 │ entity.Property(e => e.TenKh)

513 │ .HasMaxLength(300)

514 │ .HasColumnName("TenKH");

515 │ entity.Property(e => e.ThoiHanTt).HasColumnName("ThoiHanTT");

516 │ entity.Property(e => e.TyGiaTt).HasColumnName("TyGiaTT");

517 │ });

518 │

519 │ modelBuilder.Entity<AccHinhThucThu>(entity =>

520 │ {

521 │ entity.HasKey(e => e.Ma);

522 │

523 │ entity.ToTable("acc\_HinhThucThu");

524 │

525 │ entity.Property(e => e.Code)

526 │ .HasMaxLength(20)

527 │ .IsUnicode(false);

528 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

529 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

530 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

531 │ entity.Property(e => e.Ten).HasMaxLength(200);

532 │ });

533 │

534 │ modelBuilder.Entity<AccLoaiPhieuThu>(entity =>

535 │ {

536 │ entity.HasKey(e => e.Ma);

537 │

538 │ entity.ToTable("acc\_LoaiPhieuThu");

539 │

540 │ entity.Property(e => e.Code)

541 │ .HasMaxLength(20)

542 │ .IsUnicode(false);

543 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

544 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

545 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

546 │ entity.Property(e => e.Ten).HasMaxLength(200);

547 │ });

548 │

549 │ modelBuilder.Entity<AccPhatSinhCongNo>(entity =>

550 │ {

551 │ entity.HasKey(e => e.Ma);

552 │

553 │ entity.ToTable("acc\_PhatSinhCongNo");

554 │

555 │ entity.Property(e => e.CoCuoiKy).HasColumnType("decimal(18, 2)");

556 │ entity.Property(e => e.CoDauKy).HasColumnType("decimal(18, 2)");

557 │ entity.Property(e => e.CoTrongKy).HasColumnType("decimal(18, 2)");

558 │ entity.Property(e => e.Code)

559 │ .HasMaxLength(30)

560 │ .IsUnicode(false);

561 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

562 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

563 │ entity.Property(e => e.NgayPhatSinh).HasColumnType("datetime");

564 │ entity.Property(e => e.NoCuoiKy).HasColumnType("decimal(18, 2)");

565 │ entity.Property(e => e.NoDauKy).HasColumnType("decimal(18, 2)");

566 │ entity.Property(e => e.NoTrongKy).HasColumnType("decimal(18, 2)");

567 │ entity.Property(e => e.TenKh)

568 │ .HasMaxLength(500)

569 │ .HasColumnName("TenKH");

570 │ });

571 │

572 │ modelBuilder.Entity<AccSoDu131>(entity =>

573 │ {

574 │ entity.HasKey(e => e.Ma);

575 │

576 │ entity.ToTable("acc\_SoDu131");

577 │

578 │ entity.Property(e => e.CoCuoiKy).HasColumnType("decimal(18, 2)");

579 │ entity.Property(e => e.CoCuoiKyNt)

580 │ .HasColumnType("decimal(18, 2)")

581 │ .HasColumnName("CoCuoiKyNT");

582 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

583 │ entity.Property(e => e.NgayPhatSinh)

584 │ .HasMaxLength(10)

585 │ .IsFixedLength();

586 │ entity.Property(e => e.NoCuoiKy).HasColumnType("decimal(18, 2)");

587 │ entity.Property(e => e.NoCuoiKyNt)

588 │ .HasColumnType("decimal(18, 2)")

589 │ .HasColumnName("NoCuoiKyNT");

590 │ });

591 │

592 │ modelBuilder.Entity<AccSoDu131ChietKhau>(entity =>

593 │ {

594 │ entity.HasKey(e => e.Ma).HasName("PK\_\_acc\_SoDu\_\_3214CC9FFB4C3D91");

595 │

596 │ entity.ToTable("acc\_SoDu131ChietKhau");

597 │

598 │ entity.Property(e => e.CoDauKy).HasColumnType("decimal(18, 3)");

599 │ entity.Property(e => e.CoDauKyNt)

600 │ .HasColumnType("decimal(18, 3)")

601 │ .HasColumnName("CoDauKyNT");

602 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

603 │ entity.Property(e => e.NoDauKy).HasColumnType("decimal(18, 3)");

604 │ entity.Property(e => e.NoDauKyNt)

605 │ .HasColumnType("decimal(18, 3)")

606 │ .HasColumnName("NoDauKyNT");

607 │ });

608 │

609 │ modelBuilder.Entity<AccTaiKhoanNh>(entity =>

610 │ {

611 │ entity.HasKey(e => e.Ma);

612 │

613 │ entity.ToTable("acc\_TaiKhoanNH");

614 │

615 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

616 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

617 │ entity.Property(e => e.DiaChi).HasMaxLength(300);

618 │ entity.Property(e => e.GhiChu)

619 │ .HasMaxLength(10)

620 │ .IsFixedLength();

621 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

622 │ entity.Property(e => e.SoTaiKhoan)

623 │ .HasMaxLength(50)

624 │ .IsUnicode(false);

625 │ entity.Property(e => e.TenChuTaiKhoan).HasMaxLength(300);

626 │ entity.Property(e => e.TenNganHang).HasMaxLength(300);

627 │ });

628 │

629 │ modelBuilder.Entity<AccTamUng>(entity =>

630 │ {

631 │ entity.HasKey(e => e.Ma);

632 │

633 │ entity.ToTable("acc\_TamUng", tb => tb.HasTrigger("tg\_acc\_TamUng\_ins"));

634 │

635 │ entity.Property(e => e.ConLai).HasColumnType("decimal(18, 4)");

636 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

637 │ entity.Property(e => e.DaThanhToan).HasColumnType("decimal(18, 4)");

638 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

639 │ entity.Property(e => e.LyDo).HasMaxLength(500);

640 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

641 │ entity.Property(e => e.MaTaiKhoanNh).HasColumnName("MaTaiKhoanNH");

642 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

643 │ entity.Property(e => e.NgayChungTu).HasColumnType("datetime");

644 │ entity.Property(e => e.NguoiNop).HasMaxLength(200);

645 │ entity.Property(e => e.SoPhieu)

646 │ .HasMaxLength(30)

647 │ .IsUnicode(false);

648 │ entity.Property(e => e.SoTien).HasColumnType("decimal(18, 4)");

649 │ });

650 │

651 │ modelBuilder.Entity<AccThanhToanTheoDon>(entity =>

652 │ {

653 │ entity.HasKey(e => e.Ma);

654 │

655 │ entity.ToTable("acc\_ThanhToanTheoDon", tb => tb.HasTrigger("tg\_acc\_ThanhToanTheoDon\_ins"));

656 │

657 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

658 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

659 │ entity.Property(e => e.HinhThucTt).HasColumnName("HinhThucTT");

660 │ entity.Property(e => e.LyDo).HasMaxLength(300);

661 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

662 │ entity.Property(e => e.MaTaiKhoanNh).HasColumnName("MaTaiKhoanNH");

663 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

664 │ entity.Property(e => e.NgayChungTu).HasColumnType("datetime");

665 │ entity.Property(e => e.NgayChuyenSoKt)

666 │ .HasColumnType("datetime")

667 │ .HasColumnName("NgayChuyenSoKT");

668 │ entity.Property(e => e.NguoiNopTien).HasMaxLength(200);

669 │ entity.Property(e => e.SoPhieu)

670 │ .HasMaxLength(30)

671 │ .IsUnicode(false);

672 │ entity.Property(e => e.SoTienThanhToan).HasColumnType("decimal(18, 4)");

673 │ entity.Property(e => e.TruongSo1)

674 │ .HasMaxLength(50)

675 │ .IsUnicode(false);

676 │ entity.Property(e => e.TruongSo2)

677 │ .HasMaxLength(50)

678 │ .IsUnicode(false);

679 │ });

680 │

681 │ modelBuilder.Entity<AccTrangThaiTamUng>(entity =>

682 │ {

683 │ entity.HasKey(e => e.Ma);

684 │

685 │ entity.ToTable("acc\_TrangThaiTamUng");

686 │

687 │ entity.Property(e => e.Code)

688 │ .HasMaxLength(20)

689 │ .IsUnicode(false);

690 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

691 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

692 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

693 │ entity.Property(e => e.Ten).HasMaxLength(200);

694 │ });

695 │

696 │ modelBuilder.Entity<AccXuLyChenhLechTyGia>(entity =>

697 │ {

698 │ entity.HasKey(e => e.Ma);

699 │

700 │ entity.ToTable("acc\_XuLyChenhLechTyGia");

701 │

702 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

703 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

704 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

705 │ entity.Property(e => e.PhaiTraUsddonHang)

706 │ .HasColumnType("decimal(18, 4)")

707 │ .HasColumnName("PhaiTraUSDDonHang");

708 │ entity.Property(e => e.PhaiTraVnddonHang)

709 │ .HasColumnType("decimal(18, 4)")

710 │ .HasColumnName("PhaiTraVNDDonHang");

711 │ entity.Property(e => e.SoTienXuLyChenhLechVnd)

712 │ .HasColumnType("decimal(18, 4)")

713 │ .HasColumnName("SoTienXuLyChenhLechVND");

714 │ entity.Property(e => e.ThanhToanUsddonHang)

715 │ .HasColumnType("decimal(18, 4)")

716 │ .HasColumnName("ThanhToanUSDDonHang");

717 │ entity.Property(e => e.ThanhToanVnddonHang)

718 │ .HasColumnType("decimal(18, 4)")

719 │ .HasColumnName("ThanhToanVNDDonHang");

720 │ });

721 │

722 │ modelBuilder.Entity<CloseChiTietChotLoBotPhe>(entity =>

723 │ {

724 │ entity.HasKey(e => e.Ma).HasName("PK\_close\_ChiTietChoLoBotPhe");

725 │

726 │ entity.ToTable("close\_ChiTietChotLoBotPhe");

727 │

728 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

729 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

730 │ entity.Property(e => e.HeSoQuyDoi).HasColumnType("decimal(18, 4)");

731 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

732 │ entity.Property(e => e.SlbotPhe)

733 │ .HasColumnType("decimal(18, 4)")

734 │ .HasColumnName("SLBotPhe");

735 │ entity.Property(e => e.SoLuongBotPheQuiDoi).HasColumnType("decimal(18, 4)");

736 │ });

737 │

738 │ modelBuilder.Entity<CloseChiTietChotThuMua>(entity =>

739 │ {

740 │ entity.HasKey(e => e.Ma);

741 │

742 │ entity.ToTable("close\_ChiTietChotThuMua");

743 │

744 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

745 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

746 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

747 │ entity.Property(e => e.PhiVanChuyen).HasColumnType("decimal(18, 0)");

748 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(18, 0)");

749 │ entity.Property(e => e.ThanhTienTroGia).HasColumnType("decimal(18, 0)");

750 │ entity.Property(e => e.Tlhang).HasColumnName("TLHang");

751 │ entity.Property(e => e.Tlsach).HasColumnName("TLSach");

752 │ });

753 │

754 │ modelBuilder.Entity<CloseChotLo>(entity =>

755 │ {

756 │ entity.HasKey(e => e.Ma);

757 │

758 │ entity.ToTable("close\_ChotLo");

759 │

760 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

761 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

762 │ entity.Property(e => e.GiaVonTbthang).HasColumnName("GiaVonTBThang");

763 │ entity.Property(e => e.HeSoQuyDoiBotPhe).HasColumnType("decimal(18, 4)");

764 │ entity.Property(e => e.KetQua).HasMaxLength(50);

765 │ entity.Property(e => e.MaChiNhanhCanMi)

766 │ .HasMaxLength(10)

767 │ .IsFixedLength();

768 │ entity.Property(e => e.MaLo)

769 │ .HasMaxLength(30)

770 │ .IsUnicode(false);

771 │ entity.Property(e => e.MaLoThanhPham)

772 │ .HasMaxLength(30)

773 │ .IsUnicode(false);

774 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

775 │ entity.Property(e => e.NgayCapNhatChotLo).HasColumnType("datetime");

776 │ entity.Property(e => e.NgayChot).HasColumnType("datetime");

777 │ entity.Property(e => e.NgayChotThuMua).HasColumnType("datetime");

778 │ entity.Property(e => e.NgayThuMua).HasColumnType("datetime");

779 │ entity.Property(e => e.SanLuongBotMi).HasColumnType("decimal(18, 2)");

780 │ entity.Property(e => e.TiLeBotBienTinh).HasColumnType("decimal(18, 2)");

781 │ entity.Property(e => e.TiLeThuHoi).HasColumnType("decimal(18, 2)");

782 │ entity.Property(e => e.TiLeThuHoiQuiDoi).HasColumnType("decimal(18, 2)");

783 │ entity.Property(e => e.UserId).HasColumnName("UserID");

784 │ });

785 │

786 │ modelBuilder.Entity<CloseHeSoQuyDoiBotPhe>(entity =>

787 │ {

788 │ entity.HasKey(e => e.Ma);

789 │

790 │ entity.ToTable("close\_HeSoQuyDoiBotPhe");

791 │

792 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

793 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

794 │ entity.Property(e => e.GiaBotPhe).HasColumnType("decimal(18, 4)");

795 │ entity.Property(e => e.GiaBotThuong).HasColumnType("decimal(18, 4)");

796 │ entity.Property(e => e.HeSoQuyDoi).HasColumnType("decimal(18, 4)");

797 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

798 │ });

799 │

800 │ modelBuilder.Entity<Cndauky1>(entity =>

801 │ {

802 │ entity

803 │ .HasNoKey()

804 │ .ToTable("'CNDAUKY-1$'");

805 │

806 │ entity.Property(e => e.ChiNhánh)

807 │ .HasMaxLength(255)

808 │ .HasColumnName("Chi nhánh");

809 │ entity.Property(e => e.CònLại).HasColumnName("Còn lại");

810 │ entity.Property(e => e.GhiChú)

811 │ .HasMaxLength(255)

812 │ .HasColumnName("Ghi chú");

813 │ entity.Property(e => e.MaChiNhanh).HasMaxLength(255);

814 │ entity.Property(e => e.MaDonHang).HasMaxLength(255);

815 │ entity.Property(e => e.MaKh)

816 │ .HasMaxLength(255)

817 │ .HasColumnName("MaKH");

818 │ entity.Property(e => e.MãChiNhánh)

819 │ .HasMaxLength(255)

820 │ .HasColumnName("Mã chi nhánh");

821 │ entity.Property(e => e.MãKh)

822 │ .HasMaxLength(255)

823 │ .HasColumnName("Mã KH");

824 │ entity.Property(e => e.MãPhiếuXuất)

825 │ .HasMaxLength(255)

826 │ .HasColumnName("Mã phiếu xuất");

827 │ entity.Property(e => e.MãThuế).HasColumnName("Mã thuế");

828 │ entity.Property(e => e.MãTiềnTệ)

829 │ .HasMaxLength(255)

830 │ .HasColumnName("Mã tiền tệ");

831 │ entity.Property(e => e.NgàyCôngNợ).HasColumnName("Ngày Công nợ");

832 │ entity.Property(e => e.NgàyHóaĐơn)

833 │ .HasMaxLength(255)

834 │ .HasColumnName("Ngày hóa đơn");

835 │ entity.Property(e => e.NgàyNhậnHàng)

836 │ .HasMaxLength(255)

837 │ .HasColumnName("Ngày nhận hàng");

838 │ entity.Property(e => e.NgàyĐiĐường).HasColumnName("Ngày đi đường");

839 │ entity.Property(e => e.NgàyĐếnHạnTt)

840 │ .HasMaxLength(255)

841 │ .HasColumnName("Ngày đến hạn TT");

842 │ entity.Property(e => e.PhảiThu).HasColumnName("Phải thu");

843 │ entity.Property(e => e.SốHóaĐơn)

844 │ .HasMaxLength(255)

845 │ .HasColumnName("Số hóa đơn");

846 │ entity.Property(e => e.SốLượng).HasColumnName("Số lượng");

847 │ entity.Property(e => e.SốĐơnHàng)

848 │ .HasMaxLength(255)

849 │ .HasColumnName("Số đơn hàng");

850 │ entity.Property(e => e.ThànhTiền).HasColumnName("Thành tiền");

851 │ entity.Property(e => e.TiềnHàng).HasColumnName("Tiền hàng");

852 │ entity.Property(e => e.TiềnThuế).HasColumnName("Tiền thuế");

853 │ entity.Property(e => e.TrạngThái)

854 │ .HasMaxLength(255)

855 │ .HasColumnName("Trạng thái");

856 │ entity.Property(e => e.TỷGiá).HasColumnName("Tỷ giá");

857 │ entity.Property(e => e.ĐãThu).HasColumnName("Đã thu");

858 │ entity.Property(e => e.ĐơnGiá).HasColumnName("Đơn giá");

859 │ });

860 │

861 │ modelBuilder.Entity<Cnkhxk>(entity =>

862 │ {

863 │ entity

864 │ .HasNoKey()

865 │ .ToTable("CNKHXK$");

866 │

867 │ entity.Property(e => e.ChiNhánh)

868 │ .HasMaxLength(255)

869 │ .HasColumnName("Chi nhánh");

870 │ entity.Property(e => e.CònLại).HasColumnName("Còn lại");

871 │ entity.Property(e => e.GhiChú)

872 │ .HasMaxLength(255)

873 │ .HasColumnName("Ghi chú");

874 │ entity.Property(e => e.MaChiNhanh).HasMaxLength(255);

875 │ entity.Property(e => e.MaDonHang).HasMaxLength(255);

876 │ entity.Property(e => e.MaKh)

877 │ .HasMaxLength(255)

878 │ .HasColumnName("MaKH");

879 │ entity.Property(e => e.MãChiNhánh)

880 │ .HasMaxLength(255)

881 │ .HasColumnName("Mã chi nhánh");

882 │ entity.Property(e => e.MãKh)

883 │ .HasMaxLength(255)

884 │ .HasColumnName("Mã KH");

885 │ entity.Property(e => e.MãPhiếuXuất)

886 │ .HasMaxLength(255)

887 │ .HasColumnName("Mã phiếu xuất");

888 │ entity.Property(e => e.MãThuế).HasColumnName("Mã thuế");

889 │ entity.Property(e => e.MãTiềnTệ)

890 │ .HasMaxLength(255)

891 │ .HasColumnName("Mã tiền tệ");

892 │ entity.Property(e => e.NgàyCôngNợ).HasColumnName("Ngày Công nợ");

893 │ entity.Property(e => e.NgàyHóaĐơn)

894 │ .HasColumnType("datetime")

895 │ .HasColumnName("Ngày hóa đơn");

896 │ entity.Property(e => e.NgàyNhậnHàng)

897 │ .HasColumnType("datetime")

898 │ .HasColumnName("Ngày nhận hàng");

899 │ entity.Property(e => e.NgàyĐiĐường).HasColumnName("Ngày đi đường");

900 │ entity.Property(e => e.NgàyĐếnHạnTt)

901 │ .HasColumnType("datetime")

902 │ .HasColumnName("Ngày đến hạn TT");

903 │ entity.Property(e => e.PhảiThu).HasColumnName("Phải thu");

904 │ entity.Property(e => e.SốHóaĐơn)

905 │ .HasMaxLength(255)

906 │ .HasColumnName("Số hóa đơn");

907 │ entity.Property(e => e.SốLượng).HasColumnName("Số lượng");

908 │ entity.Property(e => e.SốĐơnHàng)

909 │ .HasMaxLength(255)

910 │ .HasColumnName("Số đơn hàng");

911 │ entity.Property(e => e.ThànhTiền).HasColumnName("Thành tiền");

912 │ entity.Property(e => e.TiềnHàng).HasColumnName("Tiền hàng");

913 │ entity.Property(e => e.TiềnThuế).HasColumnName("Tiền thuế");

914 │ entity.Property(e => e.TrạngThái)

915 │ .HasMaxLength(255)

916 │ .HasColumnName("Trạng thái");

917 │ entity.Property(e => e.TỷGiá).HasColumnName("Tỷ giá");

918 │ entity.Property(e => e.ĐãThu).HasColumnName("Đã thu");

919 │ entity.Property(e => e.ĐơnGiá).HasColumnName("Đơn giá");

920 │ });

921 │

922 │ modelBuilder.Entity<ContChiTietHopDong>(entity =>

923 │ {

924 │ entity.HasKey(e => e.Ma);

925 │

926 │ entity.ToTable("cont\_ChiTietHopDong");

927 │

928 │ entity.Property(e => e.Bao10Kg)

929 │ .HasColumnType("decimal(18, 5)")

930 │ .HasColumnName("Bao10KG");

931 │ entity.Property(e => e.BaoBiHd)

932 │ .HasColumnType("decimal(18, 5)")

933 │ .HasColumnName("BaoBiHD");

934 │ entity.Property(e => e.BocXep).HasColumnType("decimal(18, 5)");

935 │ entity.Property(e => e.ChiPhiHoTro).HasColumnType("decimal(18, 5)");

936 │ entity.Property(e => e.ChiPhiNgayCongNo).HasColumnType("decimal(18, 5)");

937 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

938 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

939 │ entity.Property(e => e.DonGiaBan).HasColumnType("decimal(18, 5)");

940 │ entity.Property(e => e.DonGiaSauTinh).HasColumnType("decimal(18, 5)");

941 │ entity.Property(e => e.DonGiaTaiKho).HasColumnType("decimal(18, 5)");

942 │ entity.Property(e => e.GhiChu).HasMaxLength(500);

943 │ entity.Property(e => e.GiaVanChuyen).HasColumnType("decimal(18, 5)");

944 │ entity.Property(e => e.HoaHong).HasColumnType("decimal(18, 5)");

945 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

946 │ entity.Property(e => e.MaHh).HasColumnName("MaHH");

947 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

948 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(25, 5)");

949 │ entity.Property(e => e.TienThue).HasColumnType("decimal(25, 5)");

950 │ entity.Property(e => e.TongThanhTien).HasColumnType("decimal(25, 5)");

951 │ });

952 │

953 │ modelBuilder.Entity<ContHopDong>(entity =>

954 │ {

955 │ entity.HasKey(e => e.Ma);

956 │

957 │ entity.ToTable("cont\_HopDong", tb => tb.HasTrigger("tg\_cont\_HopDong\_ins"));

958 │

959 │ entity.Property(e => e.CangXuatDen).HasMaxLength(100);

960 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

961 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

962 │ entity.Property(e => e.FileDinhKem).HasMaxLength(500);

963 │ entity.Property(e => e.GhiChu).HasMaxLength(1000);

964 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

965 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

966 │ entity.Property(e => e.NgayBatDau).HasColumnType("datetime");

967 │ entity.Property(e => e.NgayGiaHan).HasColumnType("datetime");

968 │ entity.Property(e => e.NgayHopDong).HasColumnType("datetime");

969 │ entity.Property(e => e.NgayKetThuc).HasColumnType("datetime");

970 │ entity.Property(e => e.NgayKetThucHd)

971 │ .HasColumnType("datetime")

972 │ .HasColumnName("NgayKetThucHD");

973 │ entity.Property(e => e.SoHopDong).HasMaxLength(100);

974 │ entity.Property(e => e.SoHopDongDoiTac).HasMaxLength(100);

975 │ });

976 │

977 │ modelBuilder.Entity<ContLoaiHopDong>(entity =>

978 │ {

979 │ entity.HasKey(e => e.Ma);

980 │

981 │ entity.ToTable("cont\_LoaiHopDong");

982 │

983 │ entity.Property(e => e.BoSungHang).HasDefaultValue(false);

984 │ entity.Property(e => e.Code).HasMaxLength(10);

985 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

986 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

987 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

988 │ entity.Property(e => e.Ten).HasMaxLength(500);

989 │ });

990 │

991 │ modelBuilder.Entity<ContPhucLucGanNhatDto>(entity =>

992 │ {

993 │ entity

994 │ .HasNoKey()

995 │ .ToTable("cont\_PhucLucGanNhatDTO");

996 │

997 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

998 │ });

999 │

1000 │ modelBuilder.Entity<CostChiPhiBocXep>(entity =>

1001 │ {

1002 │ entity.HasKey(e => e.Ma);

1003 │

1004 │ entity.ToTable("cost\_ChiPhiBocXep");

1005 │

1006 │ entity.Property(e => e.ChiPhiBoSung).HasColumnType("decimal(18, 0)");

1007 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1008 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1009 │ entity.Property(e => e.GhiChu).HasMaxLength(500);

1010 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1011 │ entity.Property(e => e.NgayChiPhi).HasColumnType("datetime");

1012 │ entity.Property(e => e.PhaiTra).HasColumnType("decimal(18, 0)");

1013 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(18, 0)");

1014 │ });

1015 │

1016 │ modelBuilder.Entity<CostChiPhiDichVuNoiDia>(entity =>

1017 │ {

1018 │ entity.HasKey(e => e.Ma);

1019 │

1020 │ entity.ToTable("cost\_ChiPhiDichVuNoiDia");

1021 │

1022 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1023 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1024 │ entity.Property(e => e.Gia).HasColumnType("decimal(18, 5)");

1025 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1026 │ });

1027 │

1028 │ modelBuilder.Entity<CostChiPhiDongGoi>(entity =>

1029 │ {

1030 │ entity.HasKey(e => e.Ma);

1031 │

1032 │ entity.ToTable("cost\_ChiPhiDongGoi");

1033 │

1034 │ entity.Property(e => e.ChenhLechBao50kg).HasColumnType("decimal(18, 5)");

1035 │ entity.Property(e => e.ChenhLechUsd)

1036 │ .HasColumnType("decimal(18, 5)")

1037 │ .HasColumnName("ChenhLechUSD");

1038 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1039 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1040 │ entity.Property(e => e.GiaVnd)

1041 │ .HasColumnType("decimal(18, 5)")

1042 │ .HasColumnName("GiaVND");

1043 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1044 │ entity.Property(e => e.Ten).HasMaxLength(200);

1045 │ });

1046 │

1047 │ modelBuilder.Entity<CostChiPhiKhac>(entity =>

1048 │ {

1049 │ entity.HasKey(e => e.Ma);

1050 │

1051 │ entity.ToTable("cost\_ChiPhiKhac");

1052 │

1053 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1054 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1055 │ entity.Property(e => e.GiaUsd)

1056 │ .HasColumnType("decimal(18, 5)")

1057 │ .HasColumnName("GiaUSD");

1058 │ entity.Property(e => e.GiaVnd)

1059 │ .HasColumnType("decimal(18, 5)")

1060 │ .HasColumnName("GiaVND");

1061 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1062 │ entity.Property(e => e.Ten).HasMaxLength(200);

1063 │ });

1064 │

1065 │ modelBuilder.Entity<CostChiPhiSx>(entity =>

1066 │ {

1067 │ entity.HasKey(e => e.Ma).HasName("PK\_cos\_ChiPhiSX\_1");

1068 │

1069 │ entity.ToTable("cost\_ChiPhiSX");

1070 │

1071 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1072 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1073 │ entity.Property(e => e.DenNgay).HasColumnType("datetime");

1074 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1075 │ entity.Property(e => e.NgayNhap).HasColumnType("datetime");

1076 │ entity.Property(e => e.ThanhPham).HasColumnType("decimal(18, 0)");

1077 │ entity.Property(e => e.TuNgay).HasColumnType("datetime");

1078 │ });

1079 │

1080 │ modelBuilder.Entity<CostChiTietChiPhi>(entity =>

1081 │ {

1082 │ entity.HasKey(e => e.Ma).HasName("PK\_cos\_ChiTietChiPhi");

1083 │

1084 │ entity.ToTable("cost\_ChiTietChiPhi");

1085 │

1086 │ entity.Property(e => e.ChiPhiTheoDv)

1087 │ .HasColumnType("decimal(18, 5)")

1088 │ .HasColumnName("ChiPhiTheoDV");

1089 │ entity.Property(e => e.ChiPhiTong).HasColumnType("decimal(18, 5)");

1090 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1091 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1092 │ entity.Property(e => e.DonGiaTb)

1093 │ .HasColumnType("decimal(18, 5)")

1094 │ .HasColumnName("DonGiaTB");

1095 │ entity.Property(e => e.GhiChu).HasMaxLength(500);

1096 │ entity.Property(e => e.GiaThanhSx)

1097 │ .HasColumnType("decimal(18, 5)")

1098 │ .HasColumnName("GiaThanhSX");

1099 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1100 │ entity.Property(e => e.SoluongSd)

1101 │ .HasColumnType("decimal(18, 5)")

1102 │ .HasColumnName("SoluongSD");

1103 │ });

1104 │

1105 │ modelBuilder.Entity<CostChiTietChiPhiBocXep>(entity =>

1106 │ {

1107 │ entity.HasKey(e => e.Ma);

1108 │

1109 │ entity.ToTable("cost\_ChiTietChiPhiBocXep");

1110 │

1111 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1112 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1113 │ entity.Property(e => e.DonGia).HasColumnType("decimal(18, 0)");

1114 │ entity.Property(e => e.GhiChu).HasMaxLength(500);

1115 │ entity.Property(e => e.MaCtphieuXuat).HasColumnName("MaCTPhieuXuat");

1116 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1117 │ entity.Property(e => e.SoLuong).HasColumnType("decimal(18, 0)");

1118 │ entity.Property(e => e.SoLuongBao).HasColumnType("decimal(18, 0)");

1119 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(18, 0)");

1120 │ });

1121 │

1122 │ modelBuilder.Entity<CostDanhMucChiPhiSx>(entity =>

1123 │ {

1124 │ entity.HasKey(e => e.Ma).HasName("PK\_cos\_ChiPhiSX");

1125 │

1126 │ entity.ToTable("cost\_DanhMucChiPhiSX");

1127 │

1128 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1129 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1130 │ entity.Property(e => e.DonGia).HasColumnType("decimal(18, 2)");

1131 │ entity.Property(e => e.MoTa).HasMaxLength(250);

1132 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1133 │ entity.Property(e => e.Ten).HasMaxLength(250);

1134 │ });

1135 │

1136 │ modelBuilder.Entity<CostDinhMucChiPhiSx>(entity =>

1137 │ {

1138 │ entity.HasKey(e => e.Ma);

1139 │

1140 │ entity.ToTable("cost\_DinhMucChiPhiSX");

1141 │

1142 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1143 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1144 │ entity.Property(e => e.GhiChu).HasMaxLength(100);

1145 │ entity.Property(e => e.MaDanhMucChiPhiSx).HasColumnName("MaDanhMucChiPhiSX");

1146 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1147 │ });

1148 │

1149 │ modelBuilder.Entity<CostDongHoDien>(entity =>

1150 │ {

1151 │ entity.HasKey(e => e.Ma).HasName("PK\_cos\_DongHoDien");

1152 │

1153 │ entity.ToTable("cost\_DongHoDien");

1154 │

1155 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1156 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1157 │ entity.Property(e => e.MaKhdien)

1158 │ .HasMaxLength(30)

1159 │ .IsUnicode(false)

1160 │ .HasColumnName("MaKHDien");

1161 │ entity.Property(e => e.MatKhau).HasMaxLength(80);

1162 │ entity.Property(e => e.MoTa).HasMaxLength(250);

1163 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1164 │ entity.Property(e => e.Ten).HasMaxLength(250);

1165 │ entity.Property(e => e.UserLogin)

1166 │ .HasMaxLength(50)

1167 │ .IsUnicode(false);

1168 │ });

1169 │

1170 │ modelBuilder.Entity<CostGiaDien>(entity =>

1171 │ {

1172 │ entity.HasKey(e => e.Ma);

1173 │

1174 │ entity.ToTable("cost\_GiaDien");

1175 │

1176 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1177 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1178 │ entity.Property(e => e.GiaBinhThuong).HasColumnType("decimal(18, 2)");

1179 │ entity.Property(e => e.GiaCaoDiem).HasColumnType("decimal(18, 2)");

1180 │ entity.Property(e => e.GiaThapDiem).HasColumnType("decimal(18, 2)");

1181 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1182 │ entity.Property(e => e.NgayApDung).HasColumnType("datetime");

1183 │ });

1184 │

1185 │ modelBuilder.Entity<CostLichSuDungDien>(entity =>

1186 │ {

1187 │ entity.HasKey(e => e.Ma).HasName("PK\_cos\_LichSuDungDien");

1188 │

1189 │ entity.ToTable("cost\_LichSuDungDien");

1190 │

1191 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1192 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1193 │ entity.Property(e => e.GiaBinhThuong).HasColumnType("decimal(18, 2)");

1194 │ entity.Property(e => e.GiaCaoDiem).HasColumnType("decimal(18, 2)");

1195 │ entity.Property(e => e.GiaThapDiem).HasColumnType("decimal(18, 2)");

1196 │ entity.Property(e => e.GioBinhThuong).HasColumnType("decimal(18, 5)");

1197 │ entity.Property(e => e.GioCaoDiem).HasColumnType("decimal(18, 5)");

1198 │ entity.Property(e => e.GioThapDiem).HasColumnType("decimal(18, 5)");

1199 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1200 │ entity.Property(e => e.NgaySuDung).HasColumnType("datetime");

1201 │ });

1202 │

1203 │ modelBuilder.Entity<CostLoaiChiPhi>(entity =>

1204 │ {

1205 │ entity.HasKey(e => e.Ma).HasName("PK\_cos\_LoaiChiPhi");

1206 │

1207 │ entity.ToTable("cost\_LoaiChiPhi");

1208 │

1209 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1210 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1211 │ entity.Property(e => e.MoTa).HasMaxLength(250);

1212 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1213 │ entity.Property(e => e.Ten).HasMaxLength(250);

1214 │ });

1215 │

1216 │ modelBuilder.Entity<CostNhomChiPhi>(entity =>

1217 │ {

1218 │ entity.HasKey(e => e.Ma).HasName("PK\_cos\_NhomChiPhi");

1219 │

1220 │ entity.ToTable("cost\_NhomChiPhi");

1221 │

1222 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1223 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1224 │ entity.Property(e => e.MoTa).HasMaxLength(250);

1225 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1226 │ entity.Property(e => e.Ten).HasMaxLength(250);

1227 │ });

1228 │

1229 │ modelBuilder.Entity<CusDoiThu>(entity =>

1230 │ {

1231 │ entity.HasKey(e => e.Ma);

1232 │

1233 │ entity.ToTable("cus\_DoiThu");

1234 │

1235 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1236 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1237 │ entity.Property(e => e.Duong).HasMaxLength(300);

1238 │ entity.Property(e => e.GhiChu).HasMaxLength(500);

1239 │ entity.Property(e => e.Latitude).HasColumnType("decimal(18, 6)");

1240 │ entity.Property(e => e.Logo)

1241 │ .HasMaxLength(500)

1242 │ .IsUnicode(false);

1243 │ entity.Property(e => e.Longitude).HasColumnType("decimal(18, 6)");

1244 │ entity.Property(e => e.MaSoThue)

1245 │ .HasMaxLength(20)

1246 │ .IsUnicode(false);

1247 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1248 │ entity.Property(e => e.Ten).HasMaxLength(500);

1249 │ });

1250 │

1251 │ modelBuilder.Entity<DebtCongNoKh>(entity =>

1252 │ {

1253 │ entity.HasKey(e => e.Ma);

1254 │

1255 │ entity.ToTable("debt\_CongNoKH");

1256 │

1257 │ entity.Property(e => e.CoCuoiKy).HasColumnType("decimal(18, 0)");

1258 │ entity.Property(e => e.CoDauKy).HasColumnType("decimal(18, 0)");

1259 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

1260 │ entity.Property(e => e.NoConLai).HasColumnType("decimal(18, 0)");

1261 │ entity.Property(e => e.NoDauKy).HasColumnType("decimal(18, 0)");

1262 │ entity.Property(e => e.NoTrongKy).HasColumnType("decimal(18, 0)");

1263 │ entity.Property(e => e.ThanhToan).HasColumnType("decimal(18, 0)");

1264 │ });

1265 │

1266 │ modelBuilder.Entity<DebtSoDuDauKy>(entity =>

1267 │ {

1268 │ entity.HasKey(e => e.Ma);

1269 │

1270 │ entity.ToTable("debt\_SoDuDauKy");

1271 │

1272 │ entity.Property(e => e.CoCuoiKy).HasColumnType("decimal(18, 0)");

1273 │ entity.Property(e => e.CoDauKy).HasColumnType("decimal(18, 0)");

1274 │ entity.Property(e => e.CoTrongKy).HasColumnType("decimal(18, 0)");

1275 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1276 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1277 │ entity.Property(e => e.MaKh1)

1278 │ .HasMaxLength(30)

1279 │ .IsUnicode(false);

1280 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1281 │ entity.Property(e => e.Ngay).HasColumnType("datetime");

1282 │ entity.Property(e => e.NoCuoiKy).HasColumnType("decimal(18, 0)");

1283 │ entity.Property(e => e.NoDauKy).HasColumnType("decimal(18, 0)");

1284 │ entity.Property(e => e.NoTrongKy).HasColumnType("decimal(18, 0)");

1285 │ });

1286 │

1287 │ modelBuilder.Entity<Dmkh1>(entity =>

1288 │ {

1289 │ entity

1290 │ .HasNoKey()

1291 │ .ToTable("'DMKH-1$'");

1292 │

1293 │ entity.Property(e => e.MaKh)

1294 │ .HasMaxLength(255)

1295 │ .HasColumnName("MaKH");

1296 │ entity.Property(e => e.NgàyCôngNợ).HasColumnName("Ngày công nợ");

1297 │ entity.Property(e => e.NgàyĐiĐường).HasColumnName("Ngày đi đường");

1298 │ entity.Property(e => e.TenKh)

1299 │ .HasMaxLength(255)

1300 │ .HasColumnName("TenKH");

1301 │ });

1302 │

1303 │ modelBuilder.Entity<GiaVonVatTu>(entity =>

1304 │ {

1305 │ entity

1306 │ .HasNoKey()

1307 │ .ToTable("GiaVonVatTu$");

1308 │

1309 │ entity.Property(e => e.Code).HasMaxLength(255);

1310 │ entity.Property(e => e.DonGiaNm1).HasColumnName("DonGia NM1");

1311 │ entity.Property(e => e.DonGiaNm2).HasColumnName("DonGiaNM2");

1312 │ entity.Property(e => e.F7).HasMaxLength(255);

1313 │ entity.Property(e => e.F8).HasMaxLength(255);

1314 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

1315 │ entity.Property(e => e.Ten).HasMaxLength(255);

1316 │ });

1317 │

1318 │ modelBuilder.Entity<HisChamSocKhachHang>(entity =>

1319 │ {

1320 │ entity.HasKey(e => e.Ma).HasName("PK\_his\_NhatKyKhachHang");

1321 │

1322 │ entity.ToTable("his\_ChamSocKhachHang", tb => tb.HasTrigger("tg\_his\_ChamSocKhachHang\_ins"));

1323 │

1324 │ entity.Property(e => e.ChiDaoBld)

1325 │ .HasMaxLength(2000)

1326 │ .HasColumnName("ChiDaoBLD");

1327 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1328 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1329 │ entity.Property(e => e.KetQua).HasMaxLength(2000);

1330 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

1331 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1332 │ entity.Property(e => e.NgayKetThuc).HasColumnType("datetime");

1333 │ entity.Property(e => e.NgayNhatKy).HasColumnType("datetime");

1334 │ entity.Property(e => e.NoiDung).HasMaxLength(2000);

1335 │ entity.Property(e => e.SoNhatKy)

1336 │ .HasMaxLength(20)

1337 │ .IsUnicode(false);

1338 │ entity.Property(e => e.TenKh)

1339 │ .HasMaxLength(255)

1340 │ .HasColumnName("TenKH");

1341 │ entity.Property(e => e.XuLy).HasMaxLength(2000);

1342 │ });

1343 │

1344 │ modelBuilder.Entity<HisLoaiThongTin>(entity =>

1345 │ {

1346 │ entity.HasKey(e => e.Ma);

1347 │

1348 │ entity.ToTable("his\_LoaiThongTin");

1349 │

1350 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1351 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1352 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1353 │ entity.Property(e => e.Ten).HasMaxLength(500);

1354 │ });

1355 │

1356 │ modelBuilder.Entity<HisNguonThongTin>(entity =>

1357 │ {

1358 │ entity.HasKey(e => e.Ma);

1359 │

1360 │ entity.ToTable("his\_NguonThongTin");

1361 │

1362 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1363 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1364 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1365 │ entity.Property(e => e.Ten).HasMaxLength(500);

1366 │ });

1367 │

1368 │ modelBuilder.Entity<HisNhatKyKhachHang>(entity =>

1369 │ {

1370 │ entity.HasKey(e => e.Ma).HasName("PK\_his\_NhatKyKhachHang\_1");

1371 │

1372 │ entity.ToTable("his\_NhatKyKhachHang");

1373 │

1374 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1375 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1376 │ entity.Property(e => e.DiaChi).HasMaxLength(2000);

1377 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1378 │ entity.Property(e => e.NgayNhatKy).HasColumnType("datetime");

1379 │ entity.Property(e => e.NoiDung).HasMaxLength(2000);

1380 │ entity.Property(e => e.Sdt)

1381 │ .HasMaxLength(255)

1382 │ .HasColumnName("SDT");

1383 │ entity.Property(e => e.TenKh)

1384 │ .HasMaxLength(255)

1385 │ .HasColumnName("TenKH");

1386 │ });

1387 │

1388 │ modelBuilder.Entity<ImportSdSpp>(entity =>

1389 │ {

1390 │ entity

1391 │ .HasNoKey()

1392 │ .ToTable("'importSD-SPP$'");

1393 │

1394 │ entity.Property(e => e.ChiNhánh)

1395 │ .HasMaxLength(255)

1396 │ .HasColumnName("Chi nhánh");

1397 │ entity.Property(e => e.CònLại).HasColumnName("Còn lại");

1398 │ entity.Property(e => e.GhiChú)

1399 │ .HasMaxLength(255)

1400 │ .HasColumnName("Ghi chú");

1401 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

1402 │ entity.Property(e => e.MãChiNhánh)

1403 │ .HasMaxLength(255)

1404 │ .HasColumnName("Mã chi nhánh");

1405 │ entity.Property(e => e.MãKh)

1406 │ .HasMaxLength(255)

1407 │ .HasColumnName("Mã KH");

1408 │ entity.Property(e => e.MãPhiếuXuất)

1409 │ .HasMaxLength(255)

1410 │ .HasColumnName("Mã phiếu xuất");

1411 │ entity.Property(e => e.MãThuế).HasColumnName("Mã thuế");

1412 │ entity.Property(e => e.MãTiềnTệ)

1413 │ .HasMaxLength(255)

1414 │ .HasColumnName("Mã tiền tệ");

1415 │ entity.Property(e => e.NgàyCôngNợ).HasColumnName("Ngày Công nợ");

1416 │ entity.Property(e => e.NgàyHóaĐơn)

1417 │ .HasColumnType("datetime")

1418 │ .HasColumnName("Ngày hóa đơn");

1419 │ entity.Property(e => e.NgàyNhậnHàng)

1420 │ .HasColumnType("datetime")

1421 │ .HasColumnName("Ngày nhận hàng");

1422 │ entity.Property(e => e.NgàyĐiĐường).HasColumnName("Ngày đi đường");

1423 │ entity.Property(e => e.NgàyĐếnHạnTt)

1424 │ .HasColumnType("datetime")

1425 │ .HasColumnName("Ngày đến hạn TT");

1426 │ entity.Property(e => e.PhảiThu).HasColumnName("Phải thu");

1427 │ entity.Property(e => e.SốHóaĐơn)

1428 │ .HasMaxLength(255)

1429 │ .HasColumnName("Số hóa đơn");

1430 │ entity.Property(e => e.SốLượng).HasColumnName("Số lượng");

1431 │ entity.Property(e => e.SốĐơnHàng)

1432 │ .HasMaxLength(255)

1433 │ .HasColumnName("Số đơn hàng");

1434 │ entity.Property(e => e.ThànhTiền).HasColumnName("Thành tiền");

1435 │ entity.Property(e => e.TiềnHàng).HasColumnName("Tiền hàng");

1436 │ entity.Property(e => e.TiềnThuế).HasColumnName("Tiền thuế");

1437 │ entity.Property(e => e.TrạngThái)

1438 │ .HasMaxLength(255)

1439 │ .HasColumnName("Trạng thái");

1440 │ entity.Property(e => e.TỷGiá).HasColumnName("Tỷ giá");

1441 │ entity.Property(e => e.ĐãThu).HasColumnName("Đã thu");

1442 │ entity.Property(e => e.ĐơnGiá).HasColumnName("Đơn giá");

1443 │ });

1444 │

1445 │ modelBuilder.Entity<ImportSoDu>(entity =>

1446 │ {

1447 │ entity

1448 │ .HasNoKey()

1449 │ .ToTable("ImportSoDu$");

1450 │

1451 │ entity.Property(e => e.ChiNhánh)

1452 │ .HasMaxLength(255)

1453 │ .HasColumnName("Chi nhánh");

1454 │ entity.Property(e => e.CònLại).HasColumnName("Còn lại");

1455 │ entity.Property(e => e.GhiChú)

1456 │ .HasMaxLength(255)

1457 │ .HasColumnName("Ghi chú");

1458 │ entity.Property(e => e.Idkh).HasColumnName("IDKH");

1459 │ entity.Property(e => e.MãChiNhánh)

1460 │ .HasMaxLength(255)

1461 │ .HasColumnName("Mã chi nhánh");

1462 │ entity.Property(e => e.MãKh)

1463 │ .HasMaxLength(255)

1464 │ .HasColumnName("Mã KH");

1465 │ entity.Property(e => e.MãPhiếuXuất)

1466 │ .HasMaxLength(255)

1467 │ .HasColumnName("Mã phiếu xuất");

1468 │ entity.Property(e => e.MãThuế).HasColumnName("Mã thuế");

1469 │ entity.Property(e => e.MãTiềnTệ)

1470 │ .HasMaxLength(255)

1471 │ .HasColumnName("Mã tiền tệ");

1472 │ entity.Property(e => e.NgàyCôngNợ).HasColumnName("Ngày Công nợ");

1473 │ entity.Property(e => e.NgàyHóaĐơn)

1474 │ .HasColumnType("datetime")

1475 │ .HasColumnName("Ngày hóa đơn");

1476 │ entity.Property(e => e.NgàyNhậnHàng)

1477 │ .HasColumnType("datetime")

1478 │ .HasColumnName("Ngày nhận hàng");

1479 │ entity.Property(e => e.NgàyĐiĐường).HasColumnName("Ngày đi đường");

1480 │ entity.Property(e => e.NgàyĐếnHạnTt)

1481 │ .HasColumnType("datetime")

1482 │ .HasColumnName("Ngày đến hạn TT");

1483 │ entity.Property(e => e.PhảiThu).HasColumnName("Phải thu");

1484 │ entity.Property(e => e.SốHóaĐơn)

1485 │ .HasMaxLength(255)

1486 │ .HasColumnName("Số hóa đơn");

1487 │ entity.Property(e => e.SốLượng).HasColumnName("Số lượng");

1488 │ entity.Property(e => e.SốĐơnHàng)

1489 │ .HasMaxLength(255)

1490 │ .HasColumnName("Số đơn hàng");

1491 │ entity.Property(e => e.ThànhTiền).HasColumnName("Thành tiền");

1492 │ entity.Property(e => e.TiềnHàng).HasColumnName("Tiền hàng");

1493 │ entity.Property(e => e.TiềnThuế).HasColumnName("Tiền thuế");

1494 │ entity.Property(e => e.TrạngThái)

1495 │ .HasMaxLength(255)

1496 │ .HasColumnName("Trạng thái");

1497 │ entity.Property(e => e.TỷGiá).HasColumnName("Tỷ giá");

1498 │ entity.Property(e => e.ĐãThu).HasColumnName("Đã thu");

1499 │ entity.Property(e => e.ĐơnGiá).HasColumnName("Đơn giá");

1500 │ });

1501 │

1502 │ modelBuilder.Entity<InvChiTietHoaDon>(entity =>

1503 │ {

1504 │ entity.HasKey(e => e.Ma);

1505 │

1506 │ entity.ToTable("inv\_ChiTietHoaDon");

1507 │

1508 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1509 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1510 │ entity.Property(e => e.DonGiaBan).HasColumnType("decimal(18, 5)");

1511 │ entity.Property(e => e.GhiChu).HasMaxLength(500);

1512 │ entity.Property(e => e.GiaVon).HasColumnType("decimal(18, 5)");

1513 │ entity.Property(e => e.GiamGia).HasColumnType("decimal(18, 5)");

1514 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

1515 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1516 │ entity.Property(e => e.PhaiThu).HasColumnType("decimal(18, 5)");

1517 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(18, 5)");

1518 │ entity.Property(e => e.ThueVat)

1519 │ .HasColumnType("decimal(18, 5)")

1520 │ .HasColumnName("ThueVAT");

1521 │ entity.Property(e => e.TienThue).HasColumnType("decimal(18, 5)");

1522 │ });

1523 │

1524 │ modelBuilder.Entity<InvHoaDon>(entity =>

1525 │ {

1526 │ entity.HasKey(e => e.Ma).HasName("PK\_inv\_HoaDonBH");

1527 │

1528 │ entity.ToTable("inv\_HoaDon");

1529 │

1530 │ entity.Property(e => e.ChenhLechTyGia).HasColumnType("decimal(18, 0)");

1531 │ entity.Property(e => e.ConLai).HasColumnType("decimal(18, 4)");

1532 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1533 │ entity.Property(e => e.DaThanhToan).HasColumnType("decimal(18, 4)");

1534 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1535 │ entity.Property(e => e.GhiChu).HasMaxLength(200);

1536 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

1537 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1538 │ entity.Property(e => e.NgayHetHanTt)

1539 │ .HasColumnType("datetime")

1540 │ .HasColumnName("NgayHetHanTT");

1541 │ entity.Property(e => e.NgayHoaDon).HasColumnType("datetime");

1542 │ entity.Property(e => e.NgayNhanHang).HasColumnType("datetime");

1543 │ entity.Property(e => e.PhaiThu).HasColumnType("decimal(18, 4)");

1544 │ entity.Property(e => e.SoDonHangDauKy)

1545 │ .HasMaxLength(30)

1546 │ .IsUnicode(false);

1547 │ entity.Property(e => e.SoHoaDon)

1548 │ .HasMaxLength(30)

1549 │ .IsUnicode(false);

1550 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(18, 4)");

1551 │ entity.Property(e => e.ThueVat)

1552 │ .HasColumnType("decimal(18, 4)")

1553 │ .HasColumnName("ThueVAT");

1554 │ entity.Property(e => e.TienThue).HasColumnType("decimal(18, 4)");

1555 │ });

1556 │

1557 │ modelBuilder.Entity<InvLoaiHoaDon>(entity =>

1558 │ {

1559 │ entity.HasKey(e => e.Ma).HasName("PK\_tbl\_LoaiHoaDon");

1560 │

1561 │ entity.ToTable("inv\_LoaiHoaDon");

1562 │

1563 │ entity.Property(e => e.Code).HasMaxLength(50);

1564 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1565 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1566 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1567 │ entity.Property(e => e.Ten).HasMaxLength(500);

1568 │ });

1569 │

1570 │ modelBuilder.Entity<InvTrangThaiHoaDon>(entity =>

1571 │ {

1572 │ entity.HasKey(e => e.Ma);

1573 │

1574 │ entity.ToTable("inv\_TrangThaiHoaDon");

1575 │

1576 │ entity.Property(e => e.Code)

1577 │ .HasMaxLength(50)

1578 │ .IsUnicode(false);

1579 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1580 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1581 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1582 │ entity.Property(e => e.Ten).HasMaxLength(200);

1583 │ });

1584 │

1585 │ modelBuilder.Entity<LocCheckIn>(entity =>

1586 │ {

1587 │ entity.HasKey(e => e.Ma);

1588 │

1589 │ entity.ToTable("loc\_CheckIn");

1590 │

1591 │ entity.Property(e => e.Address).HasMaxLength(200);

1592 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1593 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1594 │ entity.Property(e => e.GhiChu).HasMaxLength(200);

1595 │ entity.Property(e => e.Latitude).HasColumnType("decimal(18, 15)");

1596 │ entity.Property(e => e.Longitude).HasColumnType("decimal(18, 15)");

1597 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

1598 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1599 │ });

1600 │

1601 │ modelBuilder.Entity<LocHuyenThi>(entity =>

1602 │ {

1603 │ entity.HasKey(e => e.Ma);

1604 │

1605 │ entity.ToTable("loc\_HuyenThi");

1606 │

1607 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1608 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1609 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1610 │ entity.Property(e => e.Ten).HasMaxLength(100);

1611 │ });

1612 │

1613 │ modelBuilder.Entity<LocPhuongXa>(entity =>

1614 │ {

1615 │ entity.HasKey(e => e.Ma);

1616 │

1617 │ entity.ToTable("loc\_PhuongXa");

1618 │

1619 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1620 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1621 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1622 │ entity.Property(e => e.Ten).HasMaxLength(100);

1623 │ });

1624 │

1625 │ modelBuilder.Entity<LocQuocGia>(entity =>

1626 │ {

1627 │ entity.HasKey(e => e.Ma);

1628 │

1629 │ entity.ToTable("loc\_QuocGia");

1630 │

1631 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1632 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1633 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1634 │ entity.Property(e => e.Ten)

1635 │ .HasMaxLength(10)

1636 │ .IsFixedLength();

1637 │ });

1638 │

1639 │ modelBuilder.Entity<LocTinhThanh>(entity =>

1640 │ {

1641 │ entity.HasKey(e => e.Ma);

1642 │

1643 │ entity.ToTable("loc\_TinhThanh");

1644 │

1645 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1646 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1647 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1648 │ entity.Property(e => e.Ten).HasMaxLength(100);

1649 │ });

1650 │

1651 │ modelBuilder.Entity<MesThongBaoApp>(entity =>

1652 │ {

1653 │ entity.HasKey(e => e.Ma);

1654 │

1655 │ entity.ToTable("mes\_ThongBaoApp");

1656 │

1657 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1658 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1659 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1660 │ entity.Property(e => e.NoiDung).HasMaxLength(1000);

1661 │ entity.Property(e => e.TieuDe).HasMaxLength(200);

1662 │ });

1663 │

1664 │ modelBuilder.Entity<Nhập>(entity =>

1665 │ {

1666 │ entity

1667 │ .HasNoKey()

1668 │ .ToTable("NHẬP$");

1669 │

1670 │ entity.Property(e => e.Mancc)

1671 │ .HasMaxLength(255)

1672 │ .HasColumnName("MANCC");

1673 │ entity.Property(e => e.NhàCungCấpMới)

1674 │ .HasMaxLength(255)

1675 │ .HasColumnName("NHÀ CUNG CẤP MỚI");

1676 │ entity.Property(e => e.SốPhiếu)

1677 │ .HasMaxLength(255)

1678 │ .HasColumnName("Số phiếu");

1679 │ });

1680 │

1681 │ modelBuilder.Entity<NtfGroupThongBao>(entity =>

1682 │ {

1683 │ entity.HasKey(e => e.Ma);

1684 │

1685 │ entity.ToTable("ntf\_GroupThongBao");

1686 │

1687 │ entity.Property(e => e.Code).HasMaxLength(100);

1688 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1689 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1690 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1691 │ entity.Property(e => e.Ten).HasMaxLength(150);

1692 │ });

1693 │

1694 │ modelBuilder.Entity<NtfMemberGroupThongBao>(entity =>

1695 │ {

1696 │ entity.HasKey(e => e.Ma).HasName("PK\_ntf\_ChiTietGroupThongBao");

1697 │

1698 │ entity.ToTable("ntf\_MemberGroupThongBao");

1699 │

1700 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1701 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1702 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1703 │ });

1704 │

1705 │ modelBuilder.Entity<OpsBaoGia>(entity =>

1706 │ {

1707 │ entity.HasKey(e => e.Ma);

1708 │

1709 │ entity.ToTable("ops\_BaoGia", tb => tb.HasTrigger("tg\_ops\_BaoGia\_ins"));

1710 │

1711 │ entity.Property(e => e.ChiPhiBocXep).HasMaxLength(500);

1712 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1713 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1714 │ entity.Property(e => e.DenNgay).HasColumnType("datetime");

1715 │ entity.Property(e => e.DiaDiemGiaoHang).HasMaxLength(200);

1716 │ entity.Property(e => e.GhiChu).HasMaxLength(1000);

1717 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

1718 │ entity.Property(e => e.MaNvhoTro).HasColumnName("MaNVHoTro");

1719 │ entity.Property(e => e.MaNvphuTrach).HasColumnName("MaNVPhuTrach");

1720 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1721 │ entity.Property(e => e.NgayBaoGia).HasColumnType("datetime");

1722 │ entity.Property(e => e.SoBaoGia)

1723 │ .HasMaxLength(100)

1724 │ .IsUnicode(false);

1725 │ entity.Property(e => e.ThoiGianGiaoHang).HasMaxLength(200);

1726 │ entity.Property(e => e.ThoiGianThanhToan).HasMaxLength(200);

1727 │ entity.Property(e => e.TuNgay).HasColumnType("datetime");

1728 │ });

1729 │

1730 │ modelBuilder.Entity<OpsChiTietBaoGia>(entity =>

1731 │ {

1732 │ entity.HasKey(e => e.Ma);

1733 │

1734 │ entity.ToTable("ops\_ChiTietBaoGia", tb => tb.HasTrigger("tg\_ops\_ChiTietBaoGia\_ins"));

1735 │

1736 │ entity.Property(e => e.Bao10Kg)

1737 │ .HasColumnType("decimal(18, 5)")

1738 │ .HasColumnName("Bao10KG");

1739 │ entity.Property(e => e.BaoBiHd)

1740 │ .HasColumnType("decimal(18, 5)")

1741 │ .HasColumnName("BaoBiHD");

1742 │ entity.Property(e => e.BocXep).HasColumnType("decimal(18, 5)");

1743 │ entity.Property(e => e.ChiPhiHoTro).HasColumnType("decimal(18, 5)");

1744 │ entity.Property(e => e.ChiPhiNgayCongNo).HasColumnType("decimal(18, 5)");

1745 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1746 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1747 │ entity.Property(e => e.DonGiaBan).HasColumnType("decimal(18, 5)");

1748 │ entity.Property(e => e.DonGiaSauTinh).HasColumnType("decimal(18, 5)");

1749 │ entity.Property(e => e.DonGiaTaiKho).HasColumnType("decimal(18, 5)");

1750 │ entity.Property(e => e.GhiChu).HasMaxLength(300);

1751 │ entity.Property(e => e.GiaVanChuyen).HasColumnType("decimal(18, 5)");

1752 │ entity.Property(e => e.HoaHong).HasColumnType("decimal(18, 5)");

1753 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

1754 │ entity.Property(e => e.MaHh).HasColumnName("MaHH");

1755 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1756 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(25, 5)");

1757 │ });

1758 │

1759 │ modelBuilder.Entity<OpsChiTietDoanhThu>(entity =>

1760 │ {

1761 │ entity.HasKey(e => e.Ma);

1762 │

1763 │ entity.ToTable("ops\_ChiTietDoanhThu");

1764 │

1765 │ entity.Property(e => e.Bao10Kg)

1766 │ .HasColumnType("decimal(18, 5)")

1767 │ .HasColumnName("Bao10KG");

1768 │ entity.Property(e => e.BaoBiHd)

1769 │ .HasColumnType("decimal(18, 5)")

1770 │ .HasColumnName("BaoBiHD");

1771 │ entity.Property(e => e.BocXep).HasColumnType("decimal(18, 5)");

1772 │ entity.Property(e => e.ChiPhiHoTro).HasColumnType("decimal(18, 5)");

1773 │ entity.Property(e => e.ChiPhiNgayCongNo).HasColumnType("decimal(18, 5)");

1774 │ entity.Property(e => e.ChietKhau).HasColumnType("decimal(25, 5)");

1775 │ entity.Property(e => e.DonGiaSauTinh).HasColumnType("decimal(18, 5)");

1776 │ entity.Property(e => e.DonGiaTaiKho).HasColumnType("decimal(18, 5)");

1777 │ entity.Property(e => e.GiaBan).HasColumnType("decimal(18, 5)");

1778 │ entity.Property(e => e.GiaGiam).HasColumnType("decimal(18, 5)");

1779 │ entity.Property(e => e.GiaNiemYet).HasColumnType("decimal(18, 5)");

1780 │ entity.Property(e => e.GiaTaiKhoCoBaoBi).HasColumnType("decimal(18, 5)");

1781 │ entity.Property(e => e.GiaVanChuyen).HasColumnType("decimal(18, 5)");

1782 │ entity.Property(e => e.GiaVonVanChuyen).HasColumnType("decimal(18, 5)");

1783 │ entity.Property(e => e.HoaHong).HasColumnType("decimal(18, 5)");

1784 │ entity.Property(e => e.KhoHang).HasMaxLength(20);

1785 │ entity.Property(e => e.Lo).HasMaxLength(20);

1786 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

1787 │ entity.Property(e => e.MaHopDongCt).HasColumnName("MaHopDongCT");

1788 │ entity.Property(e => e.PhaiThu).HasColumnType("decimal(25, 5)");

1789 │ entity.Property(e => e.SoLuongHd).HasColumnName("SoLuongHD");

1790 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(25, 5)");

1791 │ entity.Property(e => e.ThueVat)

1792 │ .HasColumnType("decimal(25, 5)")

1793 │ .HasColumnName("ThueVAT");

1794 │ entity.Property(e => e.TienThue).HasColumnType("decimal(25, 5)");

1795 │ });

1796 │

1797 │ modelBuilder.Entity<OpsDoanhThu>(entity =>

1798 │ {

1799 │ entity.HasKey(e => e.Ma);

1800 │

1801 │ entity.ToTable("ops\_DoanhThu", tb =>

1802 │ {

1803 │ tb.HasTrigger("tg\_ops\_DoanhThu\_ins");

1804 │ tb.HasTrigger("tg\_ops\_DoanhThu\_upd");

1805 │ });

1806 │

1807 │ entity.Property(e => e.ConLai).HasColumnType("decimal(18, 5)");

1808 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1809 │ entity.Property(e => e.DaThanhToan).HasColumnType("decimal(18, 5)");

1810 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1811 │ entity.Property(e => e.DienGiai).HasMaxLength(500);

1812 │ entity.Property(e => e.ExportedDate).HasColumnType("datetime");

1813 │ entity.Property(e => e.LyDoDuyet).HasMaxLength(500);

1814 │ entity.Property(e => e.MaCnxuatHoaDon).HasColumnName("MaCNXuatHoaDon");

1815 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

1816 │ entity.Property(e => e.MaNvbh).HasColumnName("MaNVBH");

1817 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1818 │ entity.Property(e => e.NgayChungTu).HasColumnType("datetime");

1819 │ entity.Property(e => e.NgayDuKienGiaoHang).HasColumnType("datetime");

1820 │ entity.Property(e => e.NgayHetHanThanhToan).HasColumnType("datetime");

1821 │ entity.Property(e => e.NgayKetThuc).HasColumnType("datetime");

1822 │ entity.Property(e => e.NhaVanChuyen)

1823 │ .HasMaxLength(20)

1824 │ .IsUnicode(false);

1825 │ entity.Property(e => e.SoChungTu).HasMaxLength(100);

1826 │ entity.Property(e => e.SoHopDong).HasMaxLength(100);

1827 │ entity.Property(e => e.SoPo)

1828 │ .HasMaxLength(50)

1829 │ .HasColumnName("SoPO");

1830 │ entity.Property(e => e.SoXe).HasMaxLength(100);

1831 │ });

1832 │

1833 │ modelBuilder.Entity<OpsLichSuChiTietBaoGia>(entity =>

1834 │ {

1835 │ entity.HasKey(e => e.Ma);

1836 │

1837 │ entity.ToTable("ops\_LichSuChiTietBaoGia");

1838 │

1839 │ entity.Property(e => e.Bao10Kg).HasColumnName("Bao10KG");

1840 │ entity.Property(e => e.BaoBiHd).HasColumnName("BaoBiHD");

1841 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1842 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1843 │ entity.Property(e => e.GhiChu).HasMaxLength(300);

1844 │ entity.Property(e => e.MaHh).HasColumnName("MaHH");

1845 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1846 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(18, 0)");

1847 │ });

1848 │

1849 │ modelBuilder.Entity<OpsThanhToan>(entity =>

1850 │ {

1851 │ entity.HasKey(e => e.Ma);

1852 │

1853 │ entity.ToTable("ops\_ThanhToan");

1854 │

1855 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1856 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1857 │ entity.Property(e => e.GhiChu).HasMaxLength(2000);

1858 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

1859 │ entity.Property(e => e.MaKh1)

1860 │ .HasMaxLength(50)

1861 │ .IsUnicode(false);

1862 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1863 │ entity.Property(e => e.NgayThanhToan).HasColumnType("datetime");

1864 │ entity.Property(e => e.SoChungTu)

1865 │ .HasMaxLength(50)

1866 │ .IsUnicode(false);

1867 │ entity.Property(e => e.SoTien).HasColumnType("decimal(18, 0)");

1868 │ });

1869 │

1870 │ modelBuilder.Entity<OpsTieuChuanKyThuat>(entity =>

1871 │ {

1872 │ entity.HasKey(e => e.Ma);

1873 │

1874 │ entity.ToTable("ops\_TieuChuanKyThuat");

1875 │

1876 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1877 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1878 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1879 │ entity.Property(e => e.TieuChuan).HasMaxLength(100);

1880 │ });

1881 │

1882 │ modelBuilder.Entity<PlBaoCaoSanLuong>(entity =>

1883 │ {

1884 │ entity

1885 │ .HasNoKey()

1886 │ .ToTable("pl\_BaoCaoSanLuong");

1887 │

1888 │ entity.Property(e => e.DonGia).HasColumnType("decimal(18, 5)");

1889 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

1890 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

1891 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(18, 5)");

1892 │ });

1893 │

1894 │ modelBuilder.Entity<PlChiTietChiTieuBan>(entity =>

1895 │ {

1896 │ entity.HasKey(e => e.Ma).HasName("PK\_pl\_ChiTietChiTieu");

1897 │

1898 │ entity.ToTable("pl\_ChiTietChiTieuBan");

1899 │

1900 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1901 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1902 │ entity.Property(e => e.GhiChu).HasMaxLength(50);

1903 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

1904 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1905 │ });

1906 │

1907 │ modelBuilder.Entity<PlChiTieuBanHang>(entity =>

1908 │ {

1909 │ entity.HasKey(e => e.Ma);

1910 │

1911 │ entity.ToTable("pl\_ChiTieuBanHang");

1912 │

1913 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1914 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1915 │ entity.Property(e => e.GhiChu).HasMaxLength(200);

1916 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1917 │ });

1918 │

1919 │ modelBuilder.Entity<PlChiTieuBanHangTheoThang>(entity =>

1920 │ {

1921 │ entity.HasKey(e => e.Ma);

1922 │

1923 │ entity.ToTable("pl\_ChiTieuBanHangTheoThang");

1924 │

1925 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1926 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1927 │ entity.Property(e => e.GhiChu).HasMaxLength(50);

1928 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

1929 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1930 │ });

1931 │

1932 │ modelBuilder.Entity<PlChiTieuSx>(entity =>

1933 │ {

1934 │ entity.HasKey(e => e.Ma);

1935 │

1936 │ entity.ToTable("pl\_ChiTieuSX");

1937 │

1938 │ entity.Property(e => e.ChiTieuSx).HasColumnName("ChiTieuSX");

1939 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1940 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1941 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1942 │ });

1943 │

1944 │ modelBuilder.Entity<PlChiTieuThuMuaCuMi>(entity =>

1945 │ {

1946 │ entity.HasKey(e => e.Ma);

1947 │

1948 │ entity.ToTable("pl\_ChiTieuThuMuaCuMi");

1949 │

1950 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1951 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1952 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1953 │ entity.Property(e => e.SoLuongMua).HasColumnType("decimal(18, 2)");

1954 │ });

1955 │

1956 │ modelBuilder.Entity<PromChietKhauThuongMai>(entity =>

1957 │ {

1958 │ entity.HasKey(e => e.Ma).HasName("PK\_prom\_ChietKhauThuongMai\_1");

1959 │

1960 │ entity.ToTable("prom\_ChietKhauThuongMai");

1961 │

1962 │ entity.Property(e => e.Ma).HasDefaultValueSql("(newid())");

1963 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1964 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1965 │ entity.Property(e => e.DonGiaCk)

1966 │ .HasColumnType("decimal(18, 5)")

1967 │ .HasColumnName("DonGiaCK");

1968 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

1969 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

1970 │ entity.Property(e => e.NganHangNhan).HasMaxLength(200);

1971 │ entity.Property(e => e.NgayChiCk)

1972 │ .HasColumnType("datetime")

1973 │ .HasColumnName("NgayChiCK");

1974 │ entity.Property(e => e.NgayLap).HasColumnType("datetime");

1975 │ entity.Property(e => e.NguoiNhan).HasMaxLength(200);

1976 │ entity.Property(e => e.SoLuongCk)

1977 │ .HasColumnType("decimal(18, 2)")

1978 │ .HasColumnName("SoLuongCK");

1979 │ entity.Property(e => e.SoPhieu)

1980 │ .HasMaxLength(50)

1981 │ .IsUnicode(false);

1982 │ entity.Property(e => e.SoTaiKhoanNhan)

1983 │ .HasMaxLength(50)

1984 │ .IsUnicode(false);

1985 │ entity.Property(e => e.ThanhTienCk)

1986 │ .HasColumnType("decimal(18, 5)")

1987 │ .HasColumnName("ThanhTienCK");

1988 │ entity.Property(e => e.TyGia).HasColumnType("decimal(18, 2)");

1989 │ });

1990 │

1991 │ modelBuilder.Entity<PurChiTietDeNghi>(entity =>

1992 │ {

1993 │ entity.HasKey(e => e.Ma).HasName("PK\_pur\_ChiTietDonDatHang");

1994 │

1995 │ entity.ToTable("pur\_ChiTietDeNghi");

1996 │

1997 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

1998 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

1999 │ entity.Property(e => e.GhiChu).HasMaxLength(150);

2000 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

2001 │ entity.Property(e => e.MauMa).HasMaxLength(200);

2002 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2003 │ entity.Property(e => e.NgaySxtrenBb)

2004 │ .HasColumnType("datetime")

2005 │ .HasColumnName("NgaySXTrenBB");

2006 │ });

2007 │

2008 │ modelBuilder.Entity<PurChiTietDonHang>(entity =>

2009 │ {

2010 │ entity.HasKey(e => e.Ma);

2011 │

2012 │ entity.ToTable("pur\_ChiTietDonHang");

2013 │

2014 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2015 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2016 │ entity.Property(e => e.DonGia).HasColumnType("decimal(18, 4)");

2017 │ entity.Property(e => e.GhiChu).HasMaxLength(200);

2018 │ entity.Property(e => e.MaCtdeNghi).HasColumnName("MaCTDeNghi");

2019 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

2020 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2021 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(18, 4)");

2022 │ });

2023 │

2024 │ modelBuilder.Entity<PurChiTietHoaDon>(entity =>

2025 │ {

2026 │ entity.HasKey(e => e.Ma);

2027 │

2028 │ entity.ToTable("pur\_ChiTietHoaDon");

2029 │

2030 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2031 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2032 │ entity.Property(e => e.DonGia).HasColumnType("decimal(18, 4)");

2033 │ entity.Property(e => e.MaCtnhap).HasColumnName("MaCTNhap");

2034 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

2035 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2036 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(18, 4)");

2037 │ entity.Property(e => e.TienThue).HasColumnType("decimal(18, 4)");

2038 │ entity.Property(e => e.TongThanhTien).HasColumnType("decimal(18, 4)");

2039 │ });

2040 │

2041 │ modelBuilder.Entity<PurChiTietThanhToan>(entity =>

2042 │ {

2043 │ entity.HasKey(e => e.Ma);

2044 │

2045 │ entity.ToTable("pur\_ChiTietThanhToan");

2046 │

2047 │ entity.Property(e => e.ConLai).HasColumnType("decimal(18, 4)");

2048 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2049 │ entity.Property(e => e.DaThanhToan).HasColumnType("decimal(18, 4)");

2050 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2051 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2052 │ entity.Property(e => e.PhaiTra).HasColumnType("decimal(18, 4)");

2053 │ });

2054 │

2055 │ modelBuilder.Entity<PurDeNghiThanhToan>(entity =>

2056 │ {

2057 │ entity.HasKey(e => e.Ma);

2058 │

2059 │ entity.ToTable("pur\_DeNghiThanhToan", tb => tb.HasTrigger("tg\_pur\_DeNghiThanhToan\_ins"));

2060 │

2061 │ entity.Property(e => e.ConLai).HasColumnType("decimal(18, 4)");

2062 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2063 │ entity.Property(e => e.DaThanhToan).HasColumnType("decimal(18, 4)");

2064 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2065 │ entity.Property(e => e.GhiChu).HasMaxLength(500);

2066 │ entity.Property(e => e.MaNhaCc).HasColumnName("MaNhaCC");

2067 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2068 │ entity.Property(e => e.NganHang).HasMaxLength(250);

2069 │ entity.Property(e => e.NgayDeNghi).HasColumnType("datetime");

2070 │ entity.Property(e => e.NoiDung).HasMaxLength(250);

2071 │ entity.Property(e => e.SoPhieu)

2072 │ .HasMaxLength(30)

2073 │ .IsUnicode(false);

2074 │ entity.Property(e => e.SoTaiKhoan).HasMaxLength(25);

2075 │ entity.Property(e => e.SoTien).HasColumnType("decimal(18, 4)");

2076 │ });

2077 │

2078 │ modelBuilder.Entity<PurDonDatHang>(entity =>

2079 │ {

2080 │ entity.HasKey(e => e.Ma).HasName("PK\_pur\_DonDatHang\_1");

2081 │

2082 │ entity.ToTable("pur\_DonDatHang", tb => tb.HasTrigger("tg\_pur\_DonDatHang\_ins"));

2083 │

2084 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2085 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2086 │ entity.Property(e => e.GhiChu).HasMaxLength(200);

2087 │ entity.Property(e => e.MaNhaCc).HasColumnName("MaNhaCC");

2088 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2089 │ entity.Property(e => e.NgayDonHang).HasColumnType("datetime");

2090 │ entity.Property(e => e.NgayGiaoHang).HasColumnType("datetime");

2091 │ entity.Property(e => e.SoDonHang)

2092 │ .HasMaxLength(30)

2093 │ .IsUnicode(false);

2094 │ });

2095 │

2096 │ modelBuilder.Entity<PurHoaDon>(entity =>

2097 │ {

2098 │ entity.HasKey(e => e.Ma);

2099 │

2100 │ entity.ToTable("pur\_HoaDon", tb => tb.HasTrigger("tg\_pur\_HoaDon\_ins"));

2101 │

2102 │ entity.Property(e => e.ConLai).HasColumnType("decimal(18, 4)");

2103 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2104 │ entity.Property(e => e.DaThanhToan).HasColumnType("decimal(18, 4)");

2105 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2106 │ entity.Property(e => e.GhiChu).HasMaxLength(200);

2107 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2108 │ entity.Property(e => e.NgayHoaDon).HasColumnType("datetime");

2109 │ entity.Property(e => e.PhaiTra).HasColumnType("decimal(18, 4)");

2110 │ entity.Property(e => e.SoHoaDon)

2111 │ .HasMaxLength(30)

2112 │ .IsUnicode(false);

2113 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(18, 4)");

2114 │ entity.Property(e => e.ThueVat).HasColumnName("ThueVAT");

2115 │ entity.Property(e => e.TienThue).HasColumnType("decimal(18, 4)");

2116 │ });

2117 │

2118 │ modelBuilder.Entity<PurPhieuDeNghi>(entity =>

2119 │ {

2120 │ entity.HasKey(e => e.Ma).HasName("PK\_pur\_DonDatHang");

2121 │

2122 │ entity.ToTable("pur\_PhieuDeNghi", tb => tb.HasTrigger("tg\_pur\_PhieuDeNghi\_ins"));

2123 │

2124 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2125 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2126 │ entity.Property(e => e.GhiChu).HasMaxLength(250);

2127 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2128 │ entity.Property(e => e.NgayDeNghi).HasColumnType("datetime");

2129 │ entity.Property(e => e.NgayGiaoDuKien).HasColumnType("datetime");

2130 │ entity.Property(e => e.NoiGiao).HasMaxLength(50);

2131 │ entity.Property(e => e.SoPhieu)

2132 │ .HasMaxLength(30)

2133 │ .IsUnicode(false);

2134 │ });

2135 │

2136 │ modelBuilder.Entity<PurThanhToan>(entity =>

2137 │ {

2138 │ entity.HasKey(e => e.Ma);

2139 │

2140 │ entity.ToTable("pur\_ThanhToan", tb => tb.HasTrigger("tg\_pur\_ThanhToan\_ins"));

2141 │

2142 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2143 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2144 │ entity.Property(e => e.LyDo).HasMaxLength(500);

2145 │ entity.Property(e => e.MaDeNghiTt).HasColumnName("MaDeNghiTT");

2146 │ entity.Property(e => e.MaNhaCc).HasColumnName("MaNhaCC");

2147 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2148 │ entity.Property(e => e.NganHang).HasMaxLength(250);

2149 │ entity.Property(e => e.NgayChuyenSoKt)

2150 │ .HasColumnType("datetime")

2151 │ .HasColumnName("NgayChuyenSoKT");

2152 │ entity.Property(e => e.NgayThanhToan).HasColumnType("datetime");

2153 │ entity.Property(e => e.SoPhieu).HasMaxLength(30);

2154 │ entity.Property(e => e.SoTaiKhoan).HasMaxLength(25);

2155 │ entity.Property(e => e.SoTienThanhToan).HasColumnType("decimal(18, 4)");

2156 │ });

2157 │

2158 │ modelBuilder.Entity<PurTrangThaiDonDatHang>(entity =>

2159 │ {

2160 │ entity.HasKey(e => e.Ma);

2161 │

2162 │ entity.ToTable("pur\_TrangThaiDonDatHang");

2163 │

2164 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2165 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2166 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2167 │ entity.Property(e => e.Ten).HasMaxLength(200);

2168 │ });

2169 │

2170 │ modelBuilder.Entity<PxhPhieuXuat>(entity =>

2171 │ {

2172 │ entity.HasKey(e => e.Ma);

2173 │

2174 │ entity.ToTable("pxh\_PhieuXuat");

2175 │

2176 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2177 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2178 │ entity.Property(e => e.GhiChu).HasMaxLength(1000);

2179 │ entity.Property(e => e.MaHh).HasColumnName("MaHH");

2180 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2181 │ entity.Property(e => e.NgayXuat).HasColumnType("datetime");

2182 │ entity.Property(e => e.SoHopDong)

2183 │ .HasMaxLength(100)

2184 │ .IsUnicode(false);

2185 │ });

2186 │

2187 │ modelBuilder.Entity<RptBaoCaoDoanhThu>(entity =>

2188 │ {

2189 │ entity.HasKey(e => e.Ma);

2190 │

2191 │ entity.ToTable("rpt\_BaoCaoDoanhThu");

2192 │

2193 │ entity.Property(e => e.DonGiaNytb)

2194 │ .HasColumnType("decimal(18, 3)")

2195 │ .HasColumnName("DonGiaNYTB");

2196 │ entity.Property(e => e.DonGiaTb)

2197 │ .HasColumnType("decimal(18, 3)")

2198 │ .HasColumnName("DonGiaTB");

2199 │ entity.Property(e => e.GiaTri).HasColumnType("decimal(18, 0)");

2200 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

2201 │ });

2202 │

2203 │ modelBuilder.Entity<StocChiTietDieuChuyen>(entity =>

2204 │ {

2205 │ entity.HasKey(e => e.Ma);

2206 │

2207 │ entity.ToTable("stoc\_ChiTietDieuChuyen");

2208 │

2209 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2210 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2211 │ entity.Property(e => e.DonGia).HasColumnType("decimal(18, 4)");

2212 │ entity.Property(e => e.GhiChu).HasMaxLength(250);

2213 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

2214 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2215 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(18, 4)");

2216 │ });

2217 │

2218 │ modelBuilder.Entity<StocChiTietLenhXuatKho>(entity =>

2219 │ {

2220 │ entity.HasKey(e => e.Ma);

2221 │

2222 │ entity.ToTable("stoc\_ChiTietLenhXuatKho");

2223 │

2224 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2225 │ entity.Property(e => e.GhiChu).HasMaxLength(100);

2226 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

2227 │ entity.Property(e => e.MaLoaiBot).HasMaxLength(300);

2228 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2229 │ entity.Property(e => e.SoXe).HasMaxLength(300);

2230 │ });

2231 │

2232 │ modelBuilder.Entity<StocChiTietNhapXuat>(entity =>

2233 │ {

2234 │ entity.HasKey(e => e.Ma);

2235 │

2236 │ entity.ToTable("stoc\_ChiTietNhapXuat", tb =>

2237 │ {

2238 │ tb.HasTrigger("tg\_stoc\_ChiTietNhapXuat\_ins");

2239 │ tb.HasTrigger("tg\_stoc\_ChiTietNhapXuat\_upd");

2240 │ });

2241 │

2242 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2243 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2244 │ entity.Property(e => e.DonGia).HasColumnType("decimal(18, 4)");

2245 │ entity.Property(e => e.GhiChu).HasMaxLength(200);

2246 │ entity.Property(e => e.LoSanXuat).HasMaxLength(50);

2247 │ entity.Property(e => e.MaCtdeNghi).HasColumnName("MaCTDeNghi");

2248 │ entity.Property(e => e.MaCtlenhXuatKho).HasColumnName("MaCTLenhXuatKho");

2249 │ entity.Property(e => e.MaCtnhap).HasColumnName("MaCTNhap");

2250 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

2251 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2252 │ entity.Property(e => e.NgayBatDauSx)

2253 │ .HasColumnType("datetime")

2254 │ .HasColumnName("NgayBatDauSX");

2255 │ entity.Property(e => e.NgayDongBao).HasColumnType("datetime");

2256 │ entity.Property(e => e.NgayHetHan).HasColumnType("datetime");

2257 │ entity.Property(e => e.NgayNgungSx)

2258 │ .HasColumnType("datetime")

2259 │ .HasColumnName("NgayNgungSX");

2260 │ entity.Property(e => e.NgaySanXuat).HasColumnType("datetime");

2261 │ entity.Property(e => e.NgaySanXuatTrenBao).HasColumnType("datetime");

2262 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(18, 5)");

2263 │ });

2264 │

2265 │ modelBuilder.Entity<StocChiTietVanChuyenLenhXuatKho>(entity =>

2266 │ {

2267 │ entity.HasKey(e => e.Ma);

2268 │

2269 │ entity.ToTable("stoc\_ChiTietVanChuyenLenhXuatKho");

2270 │

2271 │ entity.Property(e => e.GhiChu).HasMaxLength(150);

2272 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

2273 │ });

2274 │

2275 │ modelBuilder.Entity<StocChiTietXuatKhoTuLenh>(entity =>

2276 │ {

2277 │ entity.HasKey(e => e.Ma);

2278 │

2279 │ entity.ToTable("stoc\_ChiTietXuatKhoTuLenh");

2280 │

2281 │ entity.Property(e => e.MaCtnhap).HasColumnName("MaCTNhap");

2282 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

2283 │ });

2284 │

2285 │ modelBuilder.Entity<StocDanhMucKho>(entity =>

2286 │ {

2287 │ entity.HasKey(e => e.Ma);

2288 │

2289 │ entity.ToTable("stoc\_DanhMucKho");

2290 │

2291 │ entity.Property(e => e.Code)

2292 │ .HasMaxLength(20)

2293 │ .IsUnicode(false);

2294 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2295 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2296 │ entity.Property(e => e.GroupZalo).HasMaxLength(100);

2297 │ entity.Property(e => e.KyHieu)

2298 │ .HasMaxLength(10)

2299 │ .IsUnicode(false);

2300 │ entity.Property(e => e.MaKhoKeToan)

2301 │ .HasMaxLength(20)

2302 │ .IsUnicode(false);

2303 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2304 │ entity.Property(e => e.Ten).HasMaxLength(100);

2305 │ entity.Property(e => e.TenNgan).HasMaxLength(50);

2306 │ });

2307 │

2308 │ modelBuilder.Entity<StocGhepLo>(entity =>

2309 │ {

2310 │ entity.HasKey(e => e.Ma);

2311 │

2312 │ entity.ToTable("stoc\_GhepLo", tb => tb.HasTrigger("tg\_stoc\_GhepLo\_ins"));

2313 │

2314 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2315 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2316 │ entity.Property(e => e.GhiChu).HasMaxLength(250);

2317 │ entity.Property(e => e.LoSanXuat)

2318 │ .HasMaxLength(50)

2319 │ .IsUnicode(false);

2320 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2321 │ entity.Property(e => e.NgayChungTu).HasColumnType("datetime");

2322 │ entity.Property(e => e.SoPhieu)

2323 │ .HasMaxLength(50)

2324 │ .IsUnicode(false);

2325 │ });

2326 │

2327 │ modelBuilder.Entity<StocKhoKeToan>(entity =>

2328 │ {

2329 │ entity.HasKey(e => e.Ma);

2330 │

2331 │ entity.ToTable("stoc\_KhoKeToan");

2332 │

2333 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2334 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2335 │ entity.Property(e => e.MaKhoKeToan)

2336 │ .HasMaxLength(20)

2337 │ .IsUnicode(false);

2338 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2339 │ entity.Property(e => e.Ten).HasMaxLength(50);

2340 │ });

2341 │

2342 │ modelBuilder.Entity<StocKhoaSo>(entity =>

2343 │ {

2344 │ entity.HasKey(e => e.Ma);

2345 │

2346 │ entity.ToTable("stoc\_KhoaSo");

2347 │

2348 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2349 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2350 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2351 │ });

2352 │

2353 │ modelBuilder.Entity<StocLenhXuatKho>(entity =>

2354 │ {

2355 │ entity.HasKey(e => e.Ma);

2356 │

2357 │ entity.ToTable("stoc\_LenhXuatKho", tb => tb.HasTrigger("tg\_stoc\_LenhXuatKho\_ins"));

2358 │

2359 │ entity.Property(e => e.Cang).HasMaxLength(500);

2360 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2361 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2362 │ entity.Property(e => e.GhiChu).HasMaxLength(500);

2363 │ entity.Property(e => e.GhiChuBaoMat).HasMaxLength(600);

2364 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

2365 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2366 │ entity.Property(e => e.NgayChungTu).HasColumnType("datetime");

2367 │ entity.Property(e => e.NgayGiaoHang).HasColumnType("datetime");

2368 │ entity.Property(e => e.NhaVanChuyen)

2369 │ .HasMaxLength(50)

2370 │ .IsUnicode(false);

2371 │ entity.Property(e => e.SoChungTu).HasMaxLength(50);

2372 │ entity.Property(e => e.SoTau).HasMaxLength(200);

2373 │ entity.Property(e => e.ThuTienMat).HasColumnType("decimal(18, 4)");

2374 │ entity.Property(e => e.TrangThai).HasComment("0: chưa kết thúc, 1: đã kết thúc");

2375 │ entity.Property(e => e.XuongTau).HasMaxLength(500);

2376 │ });

2377 │

2378 │ modelBuilder.Entity<StocLineKho>(entity =>

2379 │ {

2380 │ entity.HasKey(e => e.Ma);

2381 │

2382 │ entity.ToTable("stoc\_LineKho");

2383 │

2384 │ entity.Property(e => e.Code).HasMaxLength(20);

2385 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2386 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2387 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2388 │ entity.Property(e => e.Ten).HasMaxLength(200);

2389 │ });

2390 │

2391 │ modelBuilder.Entity<StocLoaiKho>(entity =>

2392 │ {

2393 │ entity.HasKey(e => e.Ma);

2394 │

2395 │ entity.ToTable("stoc\_LoaiKho");

2396 │

2397 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2398 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2399 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2400 │ entity.Property(e => e.Ten).HasMaxLength(50);

2401 │ });

2402 │

2403 │ modelBuilder.Entity<StocLoaiNhapXuat>(entity =>

2404 │ {

2405 │ entity.HasKey(e => e.Ma);

2406 │

2407 │ entity.ToTable("stoc\_LoaiNhapXuat");

2408 │

2409 │ entity.Property(e => e.Code).HasMaxLength(20);

2410 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2411 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2412 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2413 │ entity.Property(e => e.NhapKho).HasComment("1: Nhập kho, 0:Xuất kho");

2414 │ entity.Property(e => e.Ten).HasMaxLength(200);

2415 │ });

2416 │

2417 │ modelBuilder.Entity<StocPhieuDieuChuyen>(entity =>

2418 │ {

2419 │ entity.HasKey(e => e.Ma);

2420 │

2421 │ entity.ToTable("stoc\_PhieuDieuChuyen", tb => tb.HasTrigger("tg\_stoc\_PhieuDieuChuyen\_ins"));

2422 │

2423 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2424 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2425 │ entity.Property(e => e.GhiChu).HasMaxLength(250);

2426 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2427 │ entity.Property(e => e.NgayNhapXuat).HasColumnType("datetime");

2428 │ entity.Property(e => e.SoPhieuNhapXuat).HasMaxLength(30);

2429 │ });

2430 │

2431 │ modelBuilder.Entity<StocPhieuNhapXuat>(entity =>

2432 │ {

2433 │ entity.HasKey(e => e.Ma);

2434 │

2435 │ entity.ToTable("stoc\_PhieuNhapXuat", tb => tb.HasTrigger("tg\_stoc\_PhieuNhapXuat\_ins"));

2436 │

2437 │ entity.Property(e => e.Cang).HasMaxLength(500);

2438 │ entity.Property(e => e.ChiPhiBocVac).HasColumnType("decimal(18, 2)");

2439 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2440 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2441 │ entity.Property(e => e.GhiChu).HasMaxLength(200);

2442 │ entity.Property(e => e.MaLoaiNx).HasColumnName("MaLoaiNX");

2443 │ entity.Property(e => e.MaNcc).HasColumnName("MaNCC");

2444 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2445 │ entity.Property(e => e.NgayGiaoHang).HasColumnType("datetime");

2446 │ entity.Property(e => e.NgayNhapXuat).HasColumnType("datetime");

2447 │ entity.Property(e => e.NguoiPhuTrach).HasMaxLength(100);

2448 │ entity.Property(e => e.SoPhieu).HasMaxLength(30);

2449 │ entity.Property(e => e.SoTau).HasMaxLength(200);

2450 │ entity.Property(e => e.XuongTau).HasMaxLength(500);

2451 │ });

2452 │

2453 │ modelBuilder.Entity<StocTonKho>(entity =>

2454 │ {

2455 │ entity.HasKey(e => e.Ma);

2456 │

2457 │ entity.ToTable("stoc\_TonKho");

2458 │

2459 │ entity.Property(e => e.GiaVon).HasColumnType("decimal(18, 4)");

2460 │ entity.Property(e => e.MaCtnhap).HasColumnName("MaCTNhap");

2461 │ entity.Property(e => e.MaNhaCc).HasColumnName("MaNhaCC");

2462 │ });

2463 │

2464 │ modelBuilder.Entity<SuaDonHang>(entity =>

2465 │ {

2466 │ entity

2467 │ .HasNoKey()

2468 │ .ToTable("SuaDonHang$");

2469 │

2470 │ entity.Property(e => e.GiaUsd).HasColumnName("GiaUSD");

2471 │ entity.Property(e => e.MaKh)

2472 │ .HasMaxLength(255)

2473 │ .HasColumnName("MaKH");

2474 │ entity.Property(e => e.NgayDonHang).HasMaxLength(255);

2475 │ entity.Property(e => e.SoDonHang).HasMaxLength(255);

2476 │ });

2477 │

2478 │ modelBuilder.Entity<SysBranch>(entity =>

2479 │ {

2480 │ entity.ToTable("sys\_Branch");

2481 │

2482 │ entity.Property(e => e.Id).HasColumnName("ID");

2483 │ entity.Property(e => e.Address).HasMaxLength(250);

2484 │ entity.Property(e => e.Code).HasMaxLength(50);

2485 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2486 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2487 │ entity.Property(e => e.Email).HasMaxLength(100);

2488 │ entity.Property(e => e.Fax).HasMaxLength(50);

2489 │ entity.Property(e => e.FullName).HasMaxLength(300);

2490 │ entity.Property(e => e.GroupBanTaiChinh).HasMaxLength(100);

2491 │ entity.Property(e => e.GroupBld)

2492 │ .HasMaxLength(50)

2493 │ .HasColumnName("GroupBLD");

2494 │ entity.Property(e => e.GroupBldKsx)

2495 │ .HasMaxLength(50)

2496 │ .HasColumnName("GroupBLD\_KSX");

2497 │ entity.Property(e => e.GroupBldQlcl)

2498 │ .HasMaxLength(50)

2499 │ .IsUnicode(false)

2500 │ .HasColumnName("GroupBLD\_QLCL");

2501 │ entity.Property(e => e.GroupBldTonKho)

2502 │ .HasMaxLength(50)

2503 │ .IsUnicode(false)

2504 │ .HasColumnName("GroupBLD\_TonKho");

2505 │ entity.Property(e => e.GroupBldnm)

2506 │ .HasMaxLength(50)

2507 │ .HasColumnName("GroupBLDNM");

2508 │ entity.Property(e => e.GroupCanMi).HasMaxLength(100);

2509 │ entity.Property(e => e.GroupKeToan).HasMaxLength(50);

2510 │ entity.Property(e => e.GroupQlcl)

2511 │ .HasMaxLength(100)

2512 │ .HasColumnName("GroupQLCL");

2513 │ entity.Property(e => e.GroupTmcanhBao)

2514 │ .HasMaxLength(100)

2515 │ .HasColumnName("GroupTMCanhBao");

2516 │ entity.Property(e => e.GroupTonKhoNm)

2517 │ .HasMaxLength(100)

2518 │ .HasColumnName("GroupTonKhoNM");

2519 │ entity.Property(e => e.GroupXuatHang).HasMaxLength(100);

2520 │ entity.Property(e => e.GroupZalo).HasMaxLength(100);

2521 │ entity.Property(e => e.IndexDv)

2522 │ .HasMaxLength(10)

2523 │ .IsUnicode(false)

2524 │ .HasColumnName("IndexDV");

2525 │ entity.Property(e => e.Logo).HasMaxLength(500);

2526 │ entity.Property(e => e.MaDvcs)

2527 │ .HasMaxLength(50)

2528 │ .HasColumnName("MaDVCS");

2529 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2530 │ entity.Property(e => e.NId).HasColumnName("nID");

2531 │ entity.Property(e => e.Name).HasMaxLength(150);

2532 │ entity.Property(e => e.NgayApDungChotLo).HasColumnType("datetime");

2533 │ entity.Property(e => e.Tel).HasMaxLength(50);

2534 │ entity.Property(e => e.TenPhapNhan).HasMaxLength(500);

2535 │ entity.Property(e => e.Website).HasMaxLength(150);

2536 │ });

2537 │

2538 │ modelBuilder.Entity<SysLog>(entity =>

2539 │ {

2540 │ entity.HasKey(e => e.Ma);

2541 │

2542 │ entity.ToTable("sys\_log");

2543 │ });

2544 │

2545 │ modelBuilder.Entity<SysLogo>(entity =>

2546 │ {

2547 │ entity

2548 │ .HasNoKey()

2549 │ .ToTable("sys\_Logo");

2550 │

2551 │ entity.Property(e => e.BranchId).HasColumnName("branchID");

2552 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2553 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2554 │ entity.Property(e => e.Id)

2555 │ .ValueGeneratedOnAdd()

2556 │ .HasColumnName("id");

2557 │ entity.Property(e => e.Index).HasColumnName("index");

2558 │ entity.Property(e => e.LogoImage)

2559 │ .HasMaxLength(50)

2560 │ .HasColumnName("logoImage");

2561 │ entity.Property(e => e.LogoName)

2562 │ .HasMaxLength(50)

2563 │ .HasColumnName("logoName");

2564 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2565 │ });

2566 │

2567 │ modelBuilder.Entity<SysMenu>(entity =>

2568 │ {

2569 │ entity.ToTable("sys\_Menu");

2570 │

2571 │ entity.Property(e => e.Id).HasColumnName("ID");

2572 │ entity.Property(e => e.ClassIcon)

2573 │ .HasMaxLength(50)

2574 │ .IsUnicode(false);

2575 │ entity.Property(e => e.Code).HasMaxLength(50);

2576 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2577 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2578 │ entity.Property(e => e.Icon)

2579 │ .HasMaxLength(50)

2580 │ .IsUnicode(false);

2581 │ entity.Property(e => e.Link).HasMaxLength(200);

2582 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2583 │ entity.Property(e => e.Name).HasMaxLength(100);

2584 │ entity.Property(e => e.ParentId).HasColumnName("ParentID");

2585 │ entity.Property(e => e.UsingFor).HasMaxLength(50);

2586 │ });

2587 │

2588 │ modelBuilder.Entity<SysOtp>(entity =>

2589 │ {

2590 │ entity.HasKey(e => e.Ma);

2591 │

2592 │ entity.ToTable("sys\_OTP");

2593 │

2594 │ entity.Property(e => e.CreateDate).HasColumnType("datetime");

2595 │ entity.Property(e => e.Otp)

2596 │ .HasMaxLength(500)

2597 │ .IsUnicode(false)

2598 │ .HasColumnName("OTP");

2599 │ });

2600 │

2601 │ modelBuilder.Entity<SysParameter>(entity =>

2602 │ {

2603 │ entity.ToTable("sys\_Parameter");

2604 │

2605 │ entity.Property(e => e.Id).HasColumnName("ID");

2606 │ entity.Property(e => e.Code).HasMaxLength(50);

2607 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2608 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2609 │ entity.Property(e => e.Description).HasMaxLength(300);

2610 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2611 │ entity.Property(e => e.Name).HasMaxLength(250);

2612 │ entity.Property(e => e.Value).HasMaxLength(50);

2613 │ });

2614 │

2615 │ modelBuilder.Entity<SysPermission>(entity =>

2616 │ {

2617 │ entity.ToTable("sys\_Permission");

2618 │

2619 │ entity.Property(e => e.Id).HasColumnName("ID");

2620 │ entity.Property(e => e.Code).HasMaxLength(50);

2621 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2622 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2623 │ entity.Property(e => e.Description).HasMaxLength(300);

2624 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2625 │ entity.Property(e => e.Name).HasMaxLength(250);

2626 │ });

2627 │

2628 │ modelBuilder.Entity<SysRole>(entity =>

2629 │ {

2630 │ entity.ToTable("sys\_Role");

2631 │

2632 │ entity.Property(e => e.Id).HasColumnName("ID");

2633 │ entity.Property(e => e.Code).HasMaxLength(50);

2634 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2635 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2636 │ entity.Property(e => e.Description).HasMaxLength(300);

2637 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2638 │ entity.Property(e => e.Name).HasMaxLength(250);

2639 │ });

2640 │

2641 │ modelBuilder.Entity<SysRoleBaoCaoBld>(entity =>

2642 │ {

2643 │ entity.HasKey(e => e.Ma);

2644 │

2645 │ entity.ToTable("sys\_RoleBaoCaoBLD");

2646 │

2647 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2648 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2649 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2650 │ entity.Property(e => e.UserId).HasColumnName("UserID");

2651 │ });

2652 │

2653 │ modelBuilder.Entity<SysRoleMenu>(entity =>

2654 │ {

2655 │ entity.ToTable("sys\_RoleMenu");

2656 │

2657 │ entity.Property(e => e.Id).HasColumnName("ID");

2658 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2659 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2660 │ entity.Property(e => e.MenuId).HasColumnName("MenuID");

2661 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2662 │ entity.Property(e => e.RoleId).HasColumnName("RoleID");

2663 │ });

2664 │

2665 │ modelBuilder.Entity<SysRolePermission>(entity =>

2666 │ {

2667 │ entity.ToTable("sys\_RolePermission");

2668 │

2669 │ entity.Property(e => e.Id).HasColumnName("ID");

2670 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2671 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2672 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2673 │ entity.Property(e => e.PermissionId).HasColumnName("PermissionID");

2674 │ entity.Property(e => e.RoleMenuId).HasColumnName("RoleMenuID");

2675 │ });

2676 │

2677 │ modelBuilder.Entity<SysSentEmailHistory>(entity =>

2678 │ {

2679 │ entity.ToTable("sys\_SentEmailHistory", tb => tb.HasTrigger("tg\_sys\_SentEmailHistory\_ins"));

2680 │

2681 │ entity.Property(e => e.Id).HasColumnName("ID");

2682 │ entity.Property(e => e.Bcc)

2683 │ .HasMaxLength(1000)

2684 │ .HasColumnName("BCC");

2685 │ entity.Property(e => e.Cc)

2686 │ .HasMaxLength(1000)

2687 │ .HasColumnName("CC");

2688 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2689 │ entity.Property(e => e.Description).HasMaxLength(1000);

2690 │ entity.Property(e => e.Sender).HasMaxLength(100);

2691 │ entity.Property(e => e.Subject).HasMaxLength(500);

2692 │ entity.Property(e => e.Type).HasMaxLength(50);

2693 │ });

2694 │

2695 │ modelBuilder.Entity<SysStatus>(entity =>

2696 │ {

2697 │ entity.HasKey(e => e.Id).HasName("PK\_sysStatus");

2698 │

2699 │ entity.ToTable("sys\_Status");

2700 │

2701 │ entity.Property(e => e.Id).HasColumnName("ID");

2702 │ entity.Property(e => e.Page).HasMaxLength(50);

2703 │ entity.Property(e => e.StatusMes).HasMaxLength(50);

2704 │ });

2705 │

2706 │ modelBuilder.Entity<SysToken>(entity =>

2707 │ {

2708 │ entity.HasKey(e => e.Ma).HasName("PK\_\_sys\_Toke\_\_3214CC9FE011D5E8");

2709 │

2710 │ entity.ToTable("sys\_Token");

2711 │

2712 │ entity.HasIndex(e => new { e.QlclToken, e.CreateDate }, "IX\_sys\_Token\_qlclToken").HasFilter("([qlclToken] IS...

2713 │

2714 │ entity.Property(e => e.DeviceId).HasMaxLength(255);

2715 │ entity.Property(e => e.DeviceName).HasMaxLength(255);

2716 │ entity.Property(e => e.Ip).HasMaxLength(50);

2717 │ entity.Property(e => e.LastChangeIp).HasColumnType("datetime");

2718 │ entity.Property(e => e.QlclToken)

2719 │ .HasMaxLength(255)

2720 │ .HasColumnName("qlclToken");

2721 │ });

2722 │

2723 │ modelBuilder.Entity<SysUser>(entity =>

2724 │ {

2725 │ entity.ToTable("sys\_User", tb => tb.HasTrigger("tg\_sys\_User\_upd"));

2726 │

2727 │ entity.Property(e => e.Id).HasColumnName("ID");

2728 │ entity.Property(e => e.BranchId).HasColumnName("BranchID");

2729 │ entity.Property(e => e.CodeCv)

2730 │ .HasMaxLength(50)

2731 │ .HasColumnName("CodeCV");

2732 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2733 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2734 │ entity.Property(e => e.DepartmentId)

2735 │ .HasMaxLength(100)

2736 │ .HasColumnName("DepartmentID");

2737 │ entity.Property(e => e.DepartmentName).HasMaxLength(100);

2738 │ entity.Property(e => e.Email).HasMaxLength(50);

2739 │ entity.Property(e => e.Fcm)

2740 │ .HasMaxLength(200)

2741 │ .HasColumnName("FCM");

2742 │ entity.Property(e => e.FullName).HasMaxLength(100);

2743 │ entity.Property(e => e.KhoHang)

2744 │ .HasMaxLength(150)

2745 │ .IsUnicode(false);

2746 │ entity.Property(e => e.LoaiHangHoa)

2747 │ .HasMaxLength(150)

2748 │ .IsUnicode(false);

2749 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2750 │ entity.Property(e => e.Msnv)

2751 │ .HasMaxLength(50)

2752 │ .HasColumnName("MSNV");

2753 │ entity.Property(e => e.NhomVungMien).HasMaxLength(100);

2754 │ entity.Property(e => e.Password).HasMaxLength(300);

2755 │ entity.Property(e => e.Photo).HasMaxLength(200);

2756 │ entity.Property(e => e.Position).HasMaxLength(100);

2757 │ entity.Property(e => e.Sdt)

2758 │ .HasMaxLength(20)

2759 │ .HasColumnName("SDT");

2760 │ entity.Property(e => e.TyTrongHqcv).HasColumnName("TyTrongHQCV");

2761 │ entity.Property(e => e.TyTrongNqqd).HasColumnName("TyTrongNQQD");

2762 │ entity.Property(e => e.Username).HasMaxLength(100);

2763 │ });

2764 │

2765 │ modelBuilder.Entity<SysUserHistory>(entity =>

2766 │ {

2767 │ entity.ToTable("sys\_UserHistory");

2768 │

2769 │ entity.Property(e => e.Id).HasColumnName("ID");

2770 │ entity.Property(e => e.BranchId).HasColumnName("BranchID");

2771 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2772 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2773 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2774 │ entity.Property(e => e.Msnv)

2775 │ .HasMaxLength(50)

2776 │ .HasColumnName("MSNV");

2777 │ entity.Property(e => e.TyTrongHqcv).HasColumnName("TyTrongHQCV");

2778 │ entity.Property(e => e.TyTrongNqqd).HasColumnName("TyTrongNQQD");

2779 │ entity.Property(e => e.UserId).HasColumnName("UserID");

2780 │ });

2781 │

2782 │ modelBuilder.Entity<SysUserRole>(entity =>

2783 │ {

2784 │ entity.ToTable("sys\_UserRole");

2785 │

2786 │ entity.Property(e => e.Id).HasColumnName("ID");

2787 │ entity.Property(e => e.BranchId).HasColumnName("BranchID");

2788 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2789 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2790 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2791 │ entity.Property(e => e.RoleId).HasColumnName("RoleID");

2792 │ entity.Property(e => e.UserId).HasColumnName("UserID");

2793 │ });

2794 │

2795 │ modelBuilder.Entity<SysUserSetting>(entity =>

2796 │ {

2797 │ entity.ToTable("sys\_UserSetting");

2798 │

2799 │ entity.Property(e => e.Id).HasColumnName("ID");

2800 │ entity.Property(e => e.BrandColor).HasMaxLength(10);

2801 │ entity.Property(e => e.FooterPosition).HasMaxLength(10);

2802 │ entity.Property(e => e.HeaderColor).HasMaxLength(10);

2803 │ entity.Property(e => e.HeaderPosition).HasMaxLength(10);

2804 │ entity.Property(e => e.MenuColor).HasMaxLength(10);

2805 │ entity.Property(e => e.MenuPosition).HasMaxLength(10);

2806 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2807 │ entity.Property(e => e.UserId).HasColumnName("UserID");

2808 │ });

2809 │

2810 │ modelBuilder.Entity<TblBaoCaoBld>(entity =>

2811 │ {

2812 │ entity.HasKey(e => e.Ma);

2813 │

2814 │ entity.ToTable("tbl\_BaoCaoBLD");

2815 │

2816 │ entity.Property(e => e.Code).HasMaxLength(50);

2817 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2818 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2819 │ entity.Property(e => e.MoTa).HasMaxLength(500);

2820 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2821 │ entity.Property(e => e.Ten).HasMaxLength(200);

2822 │ });

2823 │

2824 │ modelBuilder.Entity<TblCaSanXuat>(entity =>

2825 │ {

2826 │ entity.HasKey(e => e.Ma);

2827 │

2828 │ entity.ToTable("tbl\_CaSanXuat");

2829 │

2830 │ entity.Property(e => e.Code).HasMaxLength(10);

2831 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2832 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2833 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2834 │ entity.Property(e => e.Ten).HasMaxLength(100);

2835 │ });

2836 │

2837 │ modelBuilder.Entity<TblCangDen>(entity =>

2838 │ {

2839 │ entity.HasKey(e => e.Ma);

2840 │

2841 │ entity.ToTable("tbl\_CangDen");

2842 │

2843 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2844 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2845 │ entity.Property(e => e.GiaVnd)

2846 │ .HasColumnType("decimal(18, 5)")

2847 │ .HasColumnName("GiaVND");

2848 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2849 │ entity.Property(e => e.Ten).HasMaxLength(200);

2850 │ });

2851 │

2852 │ modelBuilder.Entity<TblCheDoKhachHang>(entity =>

2853 │ {

2854 │ entity.HasKey(e => e.Ma);

2855 │

2856 │ entity.ToTable("tbl\_CheDoKhachHang");

2857 │

2858 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2859 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2860 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

2861 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2862 │ entity.Property(e => e.NgayApDung).HasColumnType("datetime");

2863 │ });

2864 │

2865 │ modelBuilder.Entity<TblChiTietDuyetChungTu>(entity =>

2866 │ {

2867 │ entity.HasKey(e => e.Ma);

2868 │

2869 │ entity.ToTable("tbl\_ChiTietDuyetChungTu");

2870 │

2871 │ entity.Property(e => e.Ma).ValueGeneratedNever();

2872 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2873 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2874 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2875 │ entity.Property(e => e.NgayDuyet).HasColumnType("datetime");

2876 │ entity.Property(e => e.TenViTri).HasMaxLength(200);

2877 │ });

2878 │

2879 │ modelBuilder.Entity<TblChiTietQuyTrinhDuyet>(entity =>

2880 │ {

2881 │ entity.HasKey(e => e.Ma);

2882 │

2883 │ entity.ToTable("tbl\_ChiTietQuyTrinhDuyet");

2884 │

2885 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2886 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2887 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2888 │ entity.Property(e => e.TenViTri).HasMaxLength(200);

2889 │ });

2890 │

2891 │ modelBuilder.Entity<TblChiTieu>(entity =>

2892 │ {

2893 │ entity.HasKey(e => e.Ma).HasName("PK\_tbl\_TieuChuan");

2894 │

2895 │ entity.ToTable("tbl\_ChiTieu");

2896 │

2897 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2898 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2899 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

2900 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2901 │ entity.Property(e => e.Ten).HasMaxLength(200);

2902 │ entity.Property(e => e.TenTiengAnh).HasMaxLength(200);

2903 │ entity.Property(e => e.TieuChuan).HasMaxLength(100);

2904 │ entity.Property(e => e.TieuChuanTiengAnh).HasMaxLength(100);

2905 │ });

2906 │

2907 │ modelBuilder.Entity<TblChucVu>(entity =>

2908 │ {

2909 │ entity.HasKey(e => e.Ma);

2910 │

2911 │ entity.ToTable("tbl\_ChucVu");

2912 │

2913 │ entity.Property(e => e.Code)

2914 │ .HasMaxLength(50)

2915 │ .IsUnicode(false);

2916 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2917 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2918 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2919 │ entity.Property(e => e.Ten).HasMaxLength(200);

2920 │ });

2921 │

2922 │ modelBuilder.Entity<TblCont>(entity =>

2923 │ {

2924 │ entity.HasKey(e => e.Ma);

2925 │

2926 │ entity.ToTable("tbl\_Cont");

2927 │

2928 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2929 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2930 │ entity.Property(e => e.GiaUsd)

2931 │ .HasColumnType("decimal(18, 5)")

2932 │ .HasColumnName("GiaUSD");

2933 │ entity.Property(e => e.GiaVnd)

2934 │ .HasColumnType("decimal(18, 5)")

2935 │ .HasColumnName("GiaVND");

2936 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2937 │ entity.Property(e => e.Ten).HasMaxLength(200);

2938 │ });

2939 │

2940 │ modelBuilder.Entity<TblDichVuNoiDia>(entity =>

2941 │ {

2942 │ entity.HasKey(e => e.Ma);

2943 │

2944 │ entity.ToTable("tbl\_DichVuNoiDia");

2945 │

2946 │ entity.Property(e => e.Code)

2947 │ .HasMaxLength(50)

2948 │ .IsUnicode(false);

2949 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2950 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2951 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2952 │ entity.Property(e => e.Ten).HasMaxLength(200);

2953 │ });

2954 │

2955 │ modelBuilder.Entity<TblDonViTinh>(entity =>

2956 │ {

2957 │ entity.HasKey(e => e.Ma);

2958 │

2959 │ entity.ToTable("tbl\_DonViTinh");

2960 │

2961 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2962 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2963 │ entity.Property(e => e.MaDvtquiDoi).HasColumnName("MaDVTQuiDoi");

2964 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2965 │ entity.Property(e => e.Ten).HasMaxLength(30);

2966 │ });

2967 │

2968 │ modelBuilder.Entity<TblEmail>(entity =>

2969 │ {

2970 │ entity.HasKey(e => e.Ma);

2971 │

2972 │ entity.ToTable("tbl\_Email");

2973 │

2974 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2975 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2976 │ entity.Property(e => e.Domain).HasMaxLength(100);

2977 │ entity.Property(e => e.Email).HasMaxLength(100);

2978 │ entity.Property(e => e.HoTen).HasMaxLength(100);

2979 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2980 │ entity.Property(e => e.ParentId)

2981 │ .HasMaxLength(2000)

2982 │ .HasColumnName("ParentID");

2983 │ entity.Property(e => e.Type).HasMaxLength(100);

2984 │ });

2985 │

2986 │ modelBuilder.Entity<TblGiaBocXep>(entity =>

2987 │ {

2988 │ entity.HasKey(e => e.Ma);

2989 │

2990 │ entity.ToTable("tbl\_GiaBocXep");

2991 │

2992 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

2993 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

2994 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

2995 │ });

2996 │

2997 │ modelBuilder.Entity<TblGiaNiemYet>(entity =>

2998 │ {

2999 │ entity.HasKey(e => e.Ma);

3000 │

3001 │ entity.ToTable("tbl\_GiaNiemYet");

3002 │

3003 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3004 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3005 │ entity.Property(e => e.GiaNiemYetUsd)

3006 │ .HasColumnType("decimal(18, 5)")

3007 │ .HasColumnName("GiaNiemYetUSD");

3008 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3009 │ entity.Property(e => e.NgayApDung).HasColumnType("datetime");

3010 │ });

3011 │

3012 │ modelBuilder.Entity<TblHangHoa>(entity =>

3013 │ {

3014 │ entity.HasKey(e => e.Ma);

3015 │

3016 │ entity.ToTable("tbl\_HangHoa");

3017 │

3018 │ entity.Property(e => e.Code).HasMaxLength(20);

3019 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3020 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3021 │ entity.Property(e => e.DonGiaUsd)

3022 │ .HasColumnType("decimal(18, 5)")

3023 │ .HasColumnName("DonGiaUSD");

3024 │ entity.Property(e => e.KhoXuat)

3025 │ .HasMaxLength(50)

3026 │ .IsUnicode(false);

3027 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

3028 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3029 │ entity.Property(e => e.Ten).HasMaxLength(500);

3030 │ entity.Property(e => e.TenTiengAnh).HasMaxLength(500);

3031 │ });

3032 │

3033 │ modelBuilder.Entity<TblLaiTaiChinh>(entity =>

3034 │ {

3035 │ entity.HasKey(e => e.Ma);

3036 │

3037 │ entity.ToTable("tbl\_LaiTaiChinh");

3038 │

3039 │ entity.Property(e => e.LaiTc)

3040 │ .HasColumnType("decimal(18, 3)")

3041 │ .HasColumnName("LaiTC");

3042 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3043 │ });

3044 │

3045 │ modelBuilder.Entity<TblLichSuNhaVanChuyen>(entity =>

3046 │ {

3047 │ entity.HasKey(e => e.Ma).HasName("PK\_tbl\_LichSuGiaVanChuyen");

3048 │

3049 │ entity.ToTable("tbl\_LichSuNhaVanChuyen");

3050 │

3051 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3052 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3053 │ entity.Property(e => e.GiaUsd)

3054 │ .HasColumnType("decimal(18, 5)")

3055 │ .HasColumnName("GiaUSD");

3056 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3057 │ entity.Property(e => e.NgayApDung).HasColumnType("datetime");

3058 │ entity.Property(e => e.Ten).HasMaxLength(200);

3059 │ });

3060 │

3061 │ modelBuilder.Entity<TblLichSuTyGia>(entity =>

3062 │ {

3063 │ entity.HasKey(e => e.Ma);

3064 │

3065 │ entity.ToTable("tbl\_LichSuTyGia");

3066 │

3067 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3068 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3069 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3070 │ entity.Property(e => e.NgayApDung).HasColumnType("datetime");

3071 │ });

3072 │

3073 │ modelBuilder.Entity<TblLoaiBao>(entity =>

3074 │ {

3075 │ entity.HasKey(e => e.Ma);

3076 │

3077 │ entity.ToTable("tbl\_LoaiBao");

3078 │

3079 │ entity.Property(e => e.ChiPhiUsd)

3080 │ .HasColumnType("decimal(18, 5)")

3081 │ .HasColumnName("ChiPhiUSD");

3082 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3083 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3084 │ entity.Property(e => e.DienGiai).HasMaxLength(50);

3085 │ entity.Property(e => e.LoaiBao).HasMaxLength(50);

3086 │ entity.Property(e => e.MaLo).HasMaxLength(50);

3087 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3088 │ entity.Property(e => e.Ten).HasMaxLength(150);

3089 │ });

3090 │

3091 │ modelBuilder.Entity<TblLoaiBot>(entity =>

3092 │ {

3093 │ entity.HasKey(e => e.Ma);

3094 │

3095 │ entity.ToTable("tbl\_LoaiBot");

3096 │

3097 │ entity.Property(e => e.Code)

3098 │ .HasMaxLength(10)

3099 │ .IsUnicode(false);

3100 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3101 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3102 │ entity.Property(e => e.DienGiai).HasMaxLength(1000);

3103 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3104 │ entity.Property(e => e.Ten).HasMaxLength(100);

3105 │ });

3106 │

3107 │ modelBuilder.Entity<TblLoaiChungTu>(entity =>

3108 │ {

3109 │ entity.HasKey(e => e.Ma);

3110 │

3111 │ entity.ToTable("tbl\_LoaiChungTu");

3112 │

3113 │ entity.Property(e => e.Code)

3114 │ .HasMaxLength(50)

3115 │ .IsUnicode(false);

3116 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3117 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3118 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3119 │ entity.Property(e => e.Ten).HasMaxLength(200);

3120 │ });

3121 │

3122 │ modelBuilder.Entity<TblLoaiHangHoa>(entity =>

3123 │ {

3124 │ entity.HasKey(e => e.Ma);

3125 │

3126 │ entity.ToTable("tbl\_LoaiHangHoa");

3127 │

3128 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3129 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3130 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3131 │ entity.Property(e => e.Ten).HasMaxLength(200);

3132 │ });

3133 │

3134 │ modelBuilder.Entity<TblLyDo>(entity =>

3135 │ {

3136 │ entity.HasKey(e => e.Ma);

3137 │

3138 │ entity.ToTable("tbl\_LyDo");

3139 │

3140 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3141 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3142 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3143 │ entity.Property(e => e.Ten).HasMaxLength(200);

3144 │ });

3145 │

3146 │ modelBuilder.Entity<TblLyDoDuyetDonHang>(entity =>

3147 │ {

3148 │ entity.HasKey(e => e.Ma);

3149 │

3150 │ entity.ToTable("tbl\_LyDoDuyetDonHang");

3151 │

3152 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3153 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3154 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3155 │ entity.Property(e => e.Ten).HasMaxLength(1000);

3156 │ });

3157 │

3158 │ modelBuilder.Entity<TblNhaCungCap>(entity =>

3159 │ {

3160 │ entity.HasKey(e => e.Ma);

3161 │

3162 │ entity.ToTable("tbl\_NhaCungCap");

3163 │

3164 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3165 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3166 │ entity.Property(e => e.DiaChi).HasMaxLength(300);

3167 │ entity.Property(e => e.DienThoai).HasMaxLength(20);

3168 │ entity.Property(e => e.Email).HasMaxLength(50);

3169 │ entity.Property(e => e.MaKt)

3170 │ .HasMaxLength(20)

3171 │ .HasColumnName("MaKT");

3172 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3173 │ entity.Property(e => e.Mst)

3174 │ .HasMaxLength(30)

3175 │ .HasColumnName("MST");

3176 │ entity.Property(e => e.NguoiLienHe).HasMaxLength(100);

3177 │ entity.Property(e => e.TenNhaCc)

3178 │ .HasMaxLength(500)

3179 │ .HasColumnName("TenNhaCC");

3180 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

3181 │ });

3182 │

3183 │ modelBuilder.Entity<TblNhaVanChuyen>(entity =>

3184 │ {

3185 │ entity.HasKey(e => e.Ma).HasName("PK\_tbl\_GiaVanChuyen");

3186 │

3187 │ entity.ToTable("tbl\_NhaVanChuyen", tb => tb.HasTrigger("tg\_tbl\_NhaVanChuyen\_Insert\_Update"));

3188 │

3189 │ entity.Property(e => e.Code)

3190 │ .HasMaxLength(20)

3191 │ .IsUnicode(false);

3192 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3193 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3194 │ entity.Property(e => e.DiaChi).HasMaxLength(200);

3195 │ entity.Property(e => e.GiaUsd)

3196 │ .HasColumnType("decimal(18, 5)")

3197 │ .HasColumnName("GiaUSD");

3198 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

3199 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3200 │ entity.Property(e => e.SoDienThoai)

3201 │ .HasMaxLength(20)

3202 │ .IsUnicode(false);

3203 │ entity.Property(e => e.Ten).HasMaxLength(200);

3204 │ });

3205 │

3206 │ modelBuilder.Entity<TblNhomChucVu>(entity =>

3207 │ {

3208 │ entity.HasKey(e => e.Ma);

3209 │

3210 │ entity.ToTable("tbl\_NhomChucVu");

3211 │

3212 │ entity.Property(e => e.Code)

3213 │ .HasMaxLength(50)

3214 │ .IsUnicode(false);

3215 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3216 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3217 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3218 │ entity.Property(e => e.Ten).HasMaxLength(200);

3219 │ });

3220 │

3221 │ modelBuilder.Entity<TblPallet>(entity =>

3222 │ {

3223 │ entity.HasKey(e => e.Ma);

3224 │

3225 │ entity.ToTable("tbl\_Pallet");

3226 │

3227 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3228 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3229 │ entity.Property(e => e.GiaVnd)

3230 │ .HasColumnType("decimal(18, 5)")

3231 │ .HasColumnName("GiaVND");

3232 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3233 │ entity.Property(e => e.Ten).HasMaxLength(200);

3234 │ });

3235 │

3236 │ modelBuilder.Entity<TblPhongBan>(entity =>

3237 │ {

3238 │ entity.HasKey(e => e.Ma);

3239 │

3240 │ entity.ToTable("tbl\_PhongBan");

3241 │

3242 │ entity.Property(e => e.Code).HasMaxLength(50);

3243 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3244 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3245 │ entity.Property(e => e.GroupSpp)

3246 │ .HasMaxLength(100)

3247 │ .HasColumnName("GroupSPP");

3248 │ entity.Property(e => e.GroupTmcanhBao)

3249 │ .HasMaxLength(100)

3250 │ .HasColumnName("GroupTMCanhBao");

3251 │ entity.Property(e => e.GroupZaloBaoGia).HasMaxLength(100);

3252 │ entity.Property(e => e.GroupZaloDonHang).HasMaxLength(100);

3253 │ entity.Property(e => e.GroupZaloHopDong).HasMaxLength(100);

3254 │ entity.Property(e => e.MaCn).HasColumnName("MaCN");

3255 │ entity.Property(e => e.MoTa).HasMaxLength(300);

3256 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3257 │ entity.Property(e => e.Ten).HasMaxLength(200);

3258 │ entity.Property(e => e.TenVietTat).HasMaxLength(100);

3259 │ });

3260 │

3261 │ modelBuilder.Entity<TblQuanLySale>(entity =>

3262 │ {

3263 │ entity.HasKey(e => e.Ma);

3264 │

3265 │ entity.ToTable("tbl\_QuanLySale");

3266 │

3267 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3268 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3269 │ entity.Property(e => e.MaNhanVienBh).HasColumnName("MaNhanVienBH");

3270 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3271 │ });

3272 │

3273 │ modelBuilder.Entity<TblQuyTrinhDuyetChungTu>(entity =>

3274 │ {

3275 │ entity.HasKey(e => e.Ma);

3276 │

3277 │ entity.ToTable("tbl\_QuyTrinhDuyetChungTu");

3278 │

3279 │ entity.Property(e => e.Code)

3280 │ .HasMaxLength(50)

3281 │ .IsUnicode(false);

3282 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3283 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3284 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3285 │ entity.Property(e => e.Ten).HasMaxLength(200);

3286 │ });

3287 │

3288 │ modelBuilder.Entity<TblSoChungTu>(entity =>

3289 │ {

3290 │ entity.HasKey(e => e.Ma);

3291 │

3292 │ entity.ToTable("tbl\_SoChungTu");

3293 │ });

3294 │

3295 │ modelBuilder.Entity<TblTaiKhoanNhaCc>(entity =>

3296 │ {

3297 │ entity.HasKey(e => e.Ma);

3298 │

3299 │ entity.ToTable("tbl\_TaiKhoanNhaCC");

3300 │

3301 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3302 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3303 │ entity.Property(e => e.MaNhaCc).HasColumnName("MaNhaCC");

3304 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3305 │ entity.Property(e => e.SoTaiKhoan)

3306 │ .HasMaxLength(20)

3307 │ .IsUnicode(false);

3308 │ entity.Property(e => e.TenNganHang).HasMaxLength(250);

3309 │ });

3310 │

3311 │ modelBuilder.Entity<TblThueVat>(entity =>

3312 │ {

3313 │ entity.HasKey(e => e.Ma);

3314 │

3315 │ entity.ToTable("tbl\_ThueVAT");

3316 │

3317 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3318 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3319 │ entity.Property(e => e.GiaTri).HasColumnType("decimal(18, 2)");

3320 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3321 │ entity.Property(e => e.Ten).HasMaxLength(20);

3322 │ });

3323 │

3324 │ modelBuilder.Entity<TblThuogHieu>(entity =>

3325 │ {

3326 │ entity.HasKey(e => e.Ma);

3327 │

3328 │ entity.ToTable("tbl\_ThuogHieu");

3329 │

3330 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3331 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3332 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3333 │ entity.Property(e => e.Ten).HasMaxLength(100);

3334 │ });

3335 │

3336 │ modelBuilder.Entity<TblTienTe>(entity =>

3337 │ {

3338 │ entity.HasKey(e => e.Ma);

3339 │

3340 │ entity.ToTable("tbl\_TienTe");

3341 │

3342 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3343 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3344 │ entity.Property(e => e.MoTa).HasMaxLength(200);

3345 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3346 │ entity.Property(e => e.Ten).HasMaxLength(100);

3347 │ });

3348 │

3349 │ modelBuilder.Entity<TblTrangThai>(entity =>

3350 │ {

3351 │ entity.HasKey(e => e.Ma);

3352 │

3353 │ entity.ToTable("tbl\_TrangThai");

3354 │

3355 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3356 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3357 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3358 │ entity.Property(e => e.Ten).HasMaxLength(200);

3359 │ });

3360 │

3361 │ modelBuilder.Entity<TblTrangThaiDonHang>(entity =>

3362 │ {

3363 │ entity.HasKey(e => e.Ma);

3364 │

3365 │ entity.ToTable("tbl\_TrangThaiDonHang");

3366 │

3367 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3368 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3369 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3370 │ entity.Property(e => e.Ten).HasMaxLength(200);

3371 │ });

3372 │

3373 │ modelBuilder.Entity<TblTrangThaiHopDong>(entity =>

3374 │ {

3375 │ entity.HasKey(e => e.Ma);

3376 │

3377 │ entity.ToTable("tbl\_TrangThaiHopDong");

3378 │

3379 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3380 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3381 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3382 │ entity.Property(e => e.Ten).HasMaxLength(200);

3383 │ });

3384 │

3385 │ modelBuilder.Entity<TblTrangThaiLenhXuatKho>(entity =>

3386 │ {

3387 │ entity.HasKey(e => e.Ma).HasName("PK\_stoc\_TrangThaiLenhXuatKho");

3388 │

3389 │ entity.ToTable("tbl\_TrangThaiLenhXuatKho");

3390 │

3391 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3392 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3393 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3394 │ entity.Property(e => e.Ten).HasMaxLength(50);

3395 │ });

3396 │

3397 │ modelBuilder.Entity<TblTyGia>(entity =>

3398 │ {

3399 │ entity.HasKey(e => e.Ma);

3400 │

3401 │ entity.ToTable("tbl\_TyGia");

3402 │

3403 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3404 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3405 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3406 │ entity.Property(e => e.NgayApDung).HasColumnType("datetime");

3407 │ });

3408 │

3409 │ modelBuilder.Entity<TmpAccSoDu131>(entity =>

3410 │ {

3411 │ entity.HasKey(e => e.Ma);

3412 │

3413 │ entity.ToTable("tmp\_acc\_SoDu131");

3414 │

3415 │ entity.Property(e => e.CoCuoiKy).HasColumnType("decimal(18, 2)");

3416 │ entity.Property(e => e.CoCuoiKyNt)

3417 │ .HasColumnType("decimal(18, 2)")

3418 │ .HasColumnName("CoCuoiKyNT");

3419 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

3420 │ entity.Property(e => e.NgayPhatSinh)

3421 │ .HasMaxLength(10)

3422 │ .IsFixedLength();

3423 │ entity.Property(e => e.NoCuoiKy).HasColumnType("decimal(18, 2)");

3424 │ entity.Property(e => e.NoCuoiKyNt)

3425 │ .HasColumnType("decimal(18, 2)")

3426 │ .HasColumnName("NoCuoiKyNT");

3427 │ });

3428 │

3429 │ modelBuilder.Entity<TmpData>(entity =>

3430 │ {

3431 │ entity

3432 │ .HasNoKey()

3433 │ .ToTable("tmpData");

3434 │

3435 │ entity.Property(e => e.CauHoi).HasMaxLength(200);

3436 │ entity.Property(e => e.MaTtkh).HasColumnName("MaTTKH");

3437 │ entity.Property(e => e.Test3)

3438 │ .HasMaxLength(10)

3439 │ .IsFixedLength();

3440 │ });

3441 │

3442 │ modelBuilder.Entity<TmpData1>(entity =>

3443 │ {

3444 │ entity

3445 │ .HasNoKey()

3446 │ .ToTable("tmpData1");

3447 │

3448 │ entity.Property(e => e.ChiDaoBld)

3449 │ .HasMaxLength(2000)

3450 │ .HasColumnName("ChiDaoBLD");

3451 │ entity.Property(e => e.CongNoHienTai).HasColumnType("decimal(18, 0)");

3452 │ entity.Property(e => e.DiaChi).HasMaxLength(458);

3453 │ entity.Property(e => e.DienThoaiNlh)

3454 │ .HasMaxLength(50)

3455 │ .HasColumnName("DienThoaiNLH");

3456 │ entity.Property(e => e.DonGiaBan).HasColumnType("decimal(37, 19)");

3457 │ entity.Property(e => e.DonGiaBanHd)

3458 │ .HasColumnType("decimal(37, 19)")

3459 │ .HasColumnName("DonGiaBanHD");

3460 │ entity.Property(e => e.DonGiaTaiKho).HasColumnType("decimal(37, 19)");

3461 │ entity.Property(e => e.DonGiaTaiKhoHd)

3462 │ .HasColumnType("decimal(37, 19)")

3463 │ .HasColumnName("DonGiaTaiKhoHD");

3464 │ entity.Property(e => e.EmailNlh)

3465 │ .HasMaxLength(50)

3466 │ .HasColumnName("EmailNLH");

3467 │ entity.Property(e => e.FullName).HasMaxLength(100);

3468 │ entity.Property(e => e.GhiChuBg)

3469 │ .HasMaxLength(1000)

3470 │ .HasColumnName("GhiChuBG");

3471 │ entity.Property(e => e.GhiChuHd)

3472 │ .HasMaxLength(1000)

3473 │ .HasColumnName("GhiChuHD");

3474 │ entity.Property(e => e.HanMucNo).HasColumnType("decimal(18, 0)");

3475 │ entity.Property(e => e.HoTenNlh)

3476 │ .HasMaxLength(50)

3477 │ .HasColumnName("HoTenNLH");

3478 │ entity.Property(e => e.KetQua).HasMaxLength(2000);

3479 │ entity.Property(e => e.LanMuaGanNhat)

3480 │ .HasMaxLength(1)

3481 │ .IsUnicode(false);

3482 │ entity.Property(e => e.LoaiBaoGia).HasMaxLength(9);

3483 │ entity.Property(e => e.LoaiKh)

3484 │ .HasMaxLength(500)

3485 │ .HasColumnName("LoaiKH");

3486 │ entity.Property(e => e.LoaiThongTinNhan).HasMaxLength(500);

3487 │ entity.Property(e => e.MaBg).HasColumnName("MaBG");

3488 │ entity.Property(e => e.MaHd).HasColumnName("MaHD");

3489 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

3490 │ entity.Property(e => e.MaNguonKh).HasColumnName("MaNguonKH");

3491 │ entity.Property(e => e.NgayBaoGia).HasColumnType("datetime");

3492 │ entity.Property(e => e.NgayHopDong).HasColumnType("datetime");

3493 │ entity.Property(e => e.NgayNhatKy).HasColumnType("datetime");

3494 │ entity.Property(e => e.NguonThongTinNhan).HasMaxLength(500);

3495 │ entity.Property(e => e.NhomGd1)

3496 │ .HasMaxLength(50)

3497 │ .HasColumnName("NhomGD1");

3498 │ entity.Property(e => e.NhomGd3)

3499 │ .HasMaxLength(50)

3500 │ .HasColumnName("NhomGD3");

3501 │ entity.Property(e => e.NoiDung).HasMaxLength(2000);

3502 │ entity.Property(e => e.SoBaoGia)

3503 │ .HasMaxLength(100)

3504 │ .IsUnicode(false);

3505 │ entity.Property(e => e.SoHopDong).HasMaxLength(100);

3506 │ entity.Property(e => e.SoLuongHd).HasColumnName("SoLuongHD");

3507 │ entity.Property(e => e.SoNhatKy)

3508 │ .HasMaxLength(20)

3509 │ .IsUnicode(false);

3510 │ entity.Property(e => e.TenKh)

3511 │ .HasMaxLength(100)

3512 │ .HasColumnName("TenKH");

3513 │ entity.Property(e => e.XuLy).HasMaxLength(2000);

3514 │ });

3515 │

3516 │ modelBuilder.Entity<TmpDataTongHopBaoGiaVaNhatKy>(entity =>

3517 │ {

3518 │ entity

3519 │ .HasNoKey()

3520 │ .ToTable("tmpDataTongHopBaoGiaVaNhatKy");

3521 │

3522 │ entity.Property(e => e.CongNoHienTai).HasColumnType("decimal(18, 0)");

3523 │ entity.Property(e => e.DiaChi).HasMaxLength(458);

3524 │ entity.Property(e => e.DienThoaiNlh)

3525 │ .HasMaxLength(50)

3526 │ .HasColumnName("DienThoaiNLH");

3527 │ entity.Property(e => e.DonGiaBan).HasColumnType("decimal(37, 19)");

3528 │ entity.Property(e => e.DonGiaBanHd)

3529 │ .HasColumnType("decimal(37, 19)")

3530 │ .HasColumnName("DonGiaBanHD");

3531 │ entity.Property(e => e.DonGiaTaiKho).HasColumnType("decimal(37, 19)");

3532 │ entity.Property(e => e.DonGiaTaiKhoHd)

3533 │ .HasColumnType("decimal(37, 19)")

3534 │ .HasColumnName("DonGiaTaiKhoHD");

3535 │ entity.Property(e => e.EmailNlh)

3536 │ .HasMaxLength(50)

3537 │ .HasColumnName("EmailNLH");

3538 │ entity.Property(e => e.FullName).HasMaxLength(100);

3539 │ entity.Property(e => e.GhiChuBg)

3540 │ .HasMaxLength(1000)

3541 │ .HasColumnName("GhiChuBG");

3542 │ entity.Property(e => e.GhiChuHd)

3543 │ .HasMaxLength(1000)

3544 │ .HasColumnName("GhiChuHD");

3545 │ entity.Property(e => e.HanMucNo).HasColumnType("decimal(18, 0)");

3546 │ entity.Property(e => e.HoTenNlh)

3547 │ .HasMaxLength(50)

3548 │ .HasColumnName("HoTenNLH");

3549 │ entity.Property(e => e.LanMuaGanNhat)

3550 │ .HasMaxLength(20)

3551 │ .IsUnicode(false);

3552 │ entity.Property(e => e.LoaiBaoGia).HasMaxLength(9);

3553 │ entity.Property(e => e.LoaiHopDong).HasMaxLength(500);

3554 │ entity.Property(e => e.LoaiKh)

3555 │ .HasMaxLength(500)

3556 │ .HasColumnName("LoaiKH");

3557 │ entity.Property(e => e.MaBg).HasColumnName("MaBG");

3558 │ entity.Property(e => e.MaHd).HasColumnName("MaHD");

3559 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

3560 │ entity.Property(e => e.NgayBaoGia).HasColumnType("datetime");

3561 │ entity.Property(e => e.NgayHopDong).HasColumnType("datetime");

3562 │ entity.Property(e => e.NguonKh)

3563 │ .HasMaxLength(500)

3564 │ .HasColumnName("NguonKH");

3565 │ entity.Property(e => e.NhomGd1)

3566 │ .HasMaxLength(50)

3567 │ .HasColumnName("NhomGD1");

3568 │ entity.Property(e => e.NhomGd3)

3569 │ .HasMaxLength(50)

3570 │ .HasColumnName("NhomGD3");

3571 │ entity.Property(e => e.SoBaoGia)

3572 │ .HasMaxLength(100)

3573 │ .IsUnicode(false);

3574 │ entity.Property(e => e.SoHopDong).HasMaxLength(100);

3575 │ entity.Property(e => e.SoLuongHd).HasColumnName("SoLuongHD");

3576 │ entity.Property(e => e.TenKh)

3577 │ .HasMaxLength(100)

3578 │ .HasColumnName("TenKH");

3579 │ entity.Property(e => e.TrangThai).HasMaxLength(200);

3580 │ entity.Property(e => e.VuotHanMuc).HasColumnType("decimal(18, 0)");

3581 │ });

3582 │

3583 │ modelBuilder.Entity<TmpTtkh>(entity =>

3584 │ {

3585 │ entity

3586 │ .HasNoKey()

3587 │ .ToTable("tmpTTKH");

3588 │

3589 │ entity.Property(e => e.CauHoi).HasMaxLength(200);

3590 │ entity.Property(e => e.MaTtkh).HasColumnName("MaTTKH");

3591 │ });

3592 │

3593 │ modelBuilder.Entity<TranChiTietThanhToanVanChuyen>(entity =>

3594 │ {

3595 │ entity.HasKey(e => e.Ma);

3596 │

3597 │ entity.ToTable("tran\_ChiTietThanhToanVanChuyen");

3598 │

3599 │ entity.Property(e => e.ConLai).HasColumnType("decimal(18, 4)");

3600 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3601 │ entity.Property(e => e.DaThanhToan).HasColumnType("decimal(18, 4)");

3602 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3603 │ entity.Property(e => e.MaTienTeTt).HasColumnName("MaTienTeTT");

3604 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3605 │ entity.Property(e => e.PhaiTra).HasColumnType("decimal(18, 4)");

3606 │ entity.Property(e => e.TyGiaTt)

3607 │ .HasColumnType("decimal(18, 4)")

3608 │ .HasColumnName("TyGiaTT");

3609 │ });

3610 │

3611 │ modelBuilder.Entity<TranChiTietVanChuyen>(entity =>

3612 │ {

3613 │ entity.HasKey(e => e.Ma);

3614 │

3615 │ entity.ToTable("tran\_ChiTietVanChuyen");

3616 │

3617 │ entity.Property(e => e.ConLai).HasColumnType("decimal(18, 5)");

3618 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3619 │ entity.Property(e => e.DaThanhToan).HasColumnType("decimal(18, 5)");

3620 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3621 │ entity.Property(e => e.DonGia).HasColumnType("decimal(18, 5)");

3622 │ entity.Property(e => e.GhiChu).HasMaxLength(150);

3623 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3624 │ entity.Property(e => e.PhaiTra).HasColumnType("decimal(18, 5)");

3625 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(18, 5)");

3626 │ entity.Property(e => e.TienVat)

3627 │ .HasColumnType("decimal(18, 5)")

3628 │ .HasColumnName("TienVAT");

3629 │ entity.Property(e => e.Vat)

3630 │ .HasColumnType("decimal(18, 2)")

3631 │ .HasColumnName("VAT");

3632 │ });

3633 │

3634 │ modelBuilder.Entity<TranDanhSachXe>(entity =>

3635 │ {

3636 │ entity.HasKey(e => e.Ma).HasName("PK\_tbl\_DanhSachXe");

3637 │

3638 │ entity.ToTable("tran\_DanhSachXe");

3639 │

3640 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3641 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3642 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

3643 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3644 │ entity.Property(e => e.SoDienThoai).HasMaxLength(20);

3645 │ entity.Property(e => e.SoXe).HasMaxLength(20);

3646 │ entity.Property(e => e.Ten).HasMaxLength(100);

3647 │ entity.Property(e => e.TenTaiXe).HasMaxLength(80);

3648 │ entity.Property(e => e.TrongTai).HasColumnType("decimal(18, 2)");

3649 │ });

3650 │

3651 │ modelBuilder.Entity<TranGiaVanChuyen>(entity =>

3652 │ {

3653 │ entity.HasKey(e => e.Ma);

3654 │

3655 │ entity.ToTable("tran\_GiaVanChuyen");

3656 │

3657 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3658 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3659 │ entity.Property(e => e.DenTrongTaiXe).HasColumnType("decimal(18, 2)");

3660 │ entity.Property(e => e.GiaVanChuyenUsd)

3661 │ .HasColumnType("decimal(18, 5)")

3662 │ .HasColumnName("GiaVanChuyenUSD");

3663 │ entity.Property(e => e.GiaVanChuyenVnd)

3664 │ .HasColumnType("decimal(18, 0)")

3665 │ .HasColumnName("GiaVanChuyenVND");

3666 │ entity.Property(e => e.MaDvtgia).HasColumnName("MaDVTGia");

3667 │ entity.Property(e => e.MaDvttrongTai).HasColumnName("MaDVTTrongTai");

3668 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

3669 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3670 │ entity.Property(e => e.TuTrongTaiXe).HasColumnType("decimal(18, 2)");

3671 │ });

3672 │

3673 │ modelBuilder.Entity<TranLoaiXe>(entity =>

3674 │ {

3675 │ entity.HasKey(e => e.Ma);

3676 │

3677 │ entity.ToTable("tran\_LoaiXe");

3678 │

3679 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3680 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3681 │ entity.Property(e => e.MaDvttheoTrongTai).HasColumnName("MaDVTTheoTrongTai");

3682 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3683 │ entity.Property(e => e.Ten).HasMaxLength(150);

3684 │ });

3685 │

3686 │ modelBuilder.Entity<TranThanhToan>(entity =>

3687 │ {

3688 │ entity.HasKey(e => e.Ma);

3689 │

3690 │ entity.ToTable("tran\_ThanhToan");

3691 │

3692 │ entity.Property(e => e.ChungTuThanhToan)

3693 │ .HasMaxLength(20)

3694 │ .IsUnicode(false);

3695 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3696 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3697 │ entity.Property(e => e.DienGiai).HasMaxLength(50);

3698 │ entity.Property(e => e.KyThanhToan)

3699 │ .HasMaxLength(20)

3700 │ .IsUnicode(false);

3701 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3702 │ entity.Property(e => e.NgayThanhToan).HasColumnType("datetime");

3703 │ });

3704 │

3705 │ modelBuilder.Entity<TranThanhToanVanChuyen>(entity =>

3706 │ {

3707 │ entity.HasKey(e => e.Ma);

3708 │

3709 │ entity.ToTable("tran\_ThanhToanVanChuyen");

3710 │

3711 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3712 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3713 │ entity.Property(e => e.HinhThucTt).HasColumnName("HinhThucTT");

3714 │ entity.Property(e => e.LyDo).HasMaxLength(300);

3715 │ entity.Property(e => e.MaTaiKhoanNh).HasColumnName("MaTaiKhoanNH");

3716 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3717 │ entity.Property(e => e.NgayChungTu).HasColumnType("datetime");

3718 │ entity.Property(e => e.NgayChuyenSoKt)

3719 │ .HasColumnType("datetime")

3720 │ .HasColumnName("NgayChuyenSoKT");

3721 │ entity.Property(e => e.NguoiNopTien).HasMaxLength(200);

3722 │ entity.Property(e => e.SoPhieu)

3723 │ .HasMaxLength(30)

3724 │ .IsUnicode(false);

3725 │ entity.Property(e => e.SoTienThanhToan).HasColumnType("decimal(18, 4)");

3726 │ entity.Property(e => e.TruongSo1)

3727 │ .HasMaxLength(50)

3728 │ .IsUnicode(false);

3729 │ entity.Property(e => e.TruongSo2)

3730 │ .HasMaxLength(50)

3731 │ .IsUnicode(false);

3732 │ });

3733 │

3734 │ modelBuilder.Entity<TranVanChuyen>(entity =>

3735 │ {

3736 │ entity.HasKey(e => e.Ma).HasName("PK\_tran\_ChiPhiVanChuyen");

3737 │

3738 │ entity.ToTable("tran\_VanChuyen");

3739 │

3740 │ entity.Property(e => e.ConLai).HasColumnType("decimal(18, 5)");

3741 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3742 │ entity.Property(e => e.DaThanhToan).HasColumnType("decimal(18, 5)");

3743 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3744 │ entity.Property(e => e.MaHoaDonBh).HasColumnName("MaHoaDonBH");

3745 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3746 │ entity.Property(e => e.NgayThanhToan).HasColumnType("datetime");

3747 │ entity.Property(e => e.NgayVanChuyen).HasColumnType("datetime");

3748 │ entity.Property(e => e.PhaiTra).HasColumnType("decimal(18, 5)");

3749 │ });

3750 │

3751 │ modelBuilder.Entity<TtkhBangDanhGia>(entity =>

3752 │ {

3753 │ entity.HasKey(e => e.Ma);

3754 │

3755 │ entity.ToTable("ttkh\_BangDanhGia");

3756 │

3757 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3758 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3759 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3760 │ });

3761 │

3762 │ modelBuilder.Entity<TtkhCauHoiKh>(entity =>

3763 │ {

3764 │ entity.HasKey(e => e.Ma).HasName("PK\_ttkh\_YeuCauKH");

3765 │

3766 │ entity.ToTable("ttkh\_CauHoiKH");

3767 │

3768 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3769 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3770 │ entity.Property(e => e.LoaiCauTraLoi).HasMaxLength(50);

3771 │ entity.Property(e => e.MoTaNgan).HasMaxLength(100);

3772 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3773 │ entity.Property(e => e.PhuLuc).HasMaxLength(10);

3774 │ entity.Property(e => e.TenCauHoi).HasMaxLength(200);

3775 │ });

3776 │

3777 │ modelBuilder.Entity<TtkhChiTietBangDanhGia>(entity =>

3778 │ {

3779 │ entity.HasKey(e => e.Ma);

3780 │

3781 │ entity.ToTable("ttkh\_ChiTietBangDanhGia");

3782 │

3783 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3784 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3785 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3786 │ });

3787 │

3788 │ modelBuilder.Entity<TtkhChiTietTtkh>(entity =>

3789 │ {

3790 │ entity.HasKey(e => e.Ma);

3791 │

3792 │ entity.ToTable("ttkh\_ChiTietTTKH");

3793 │

3794 │ entity.Property(e => e.CauHoi).HasMaxLength(200);

3795 │ entity.Property(e => e.CauTraLoi).HasMaxLength(200);

3796 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3797 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3798 │ entity.Property(e => e.MaTtkh).HasColumnName("MaTTKH");

3799 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3800 │ });

3801 │

3802 │ modelBuilder.Entity<TtkhDiaChiGiaoHang>(entity =>

3803 │ {

3804 │ entity.HasKey(e => e.Ma);

3805 │

3806 │ entity.ToTable("ttkh\_DiaChiGiaoHang");

3807 │

3808 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3809 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3810 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

3811 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3812 │ entity.Property(e => e.TenDuong).HasMaxLength(500);

3813 │ });

3814 │

3815 │ modelBuilder.Entity<TtkhDiemDanhGia>(entity =>

3816 │ {

3817 │ entity.HasKey(e => e.Ma);

3818 │

3819 │ entity.ToTable("ttkh\_DiemDanhGia");

3820 │

3821 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3822 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3823 │ entity.Property(e => e.MoTa).HasMaxLength(200);

3824 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3825 │ });

3826 │

3827 │ modelBuilder.Entity<TtkhLoaiKh>(entity =>

3828 │ {

3829 │ entity.HasKey(e => e.Ma);

3830 │

3831 │ entity.ToTable("ttkh\_LoaiKH");

3832 │

3833 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3834 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3835 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3836 │ entity.Property(e => e.Ten).HasMaxLength(500);

3837 │ });

3838 │

3839 │ modelBuilder.Entity<TtkhNguoiLienHe>(entity =>

3840 │ {

3841 │ entity.HasKey(e => e.Ma);

3842 │

3843 │ entity.ToTable("ttkh\_NguoiLienHe");

3844 │

3845 │ entity.Property(e => e.Code)

3846 │ .HasMaxLength(50)

3847 │ .IsUnicode(false);

3848 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3849 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3850 │ entity.Property(e => e.DiaChi).HasMaxLength(200);

3851 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3852 │ entity.Property(e => e.NganHang).HasMaxLength(250);

3853 │ entity.Property(e => e.SoDienThoai)

3854 │ .HasMaxLength(50)

3855 │ .IsUnicode(false);

3856 │ entity.Property(e => e.SoTaiKhoan)

3857 │ .HasMaxLength(50)

3858 │ .IsUnicode(false);

3859 │ entity.Property(e => e.Ten).HasMaxLength(200);

3860 │ entity.Property(e => e.TenTaiKhoan).HasMaxLength(200);

3861 │ });

3862 │

3863 │ modelBuilder.Entity<TtkhNguonKh>(entity =>

3864 │ {

3865 │ entity.HasKey(e => e.Ma);

3866 │

3867 │ entity.ToTable("ttkh\_NguonKH");

3868 │

3869 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3870 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3871 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3872 │ entity.Property(e => e.Ten).HasMaxLength(500);

3873 │ });

3874 │

3875 │ modelBuilder.Entity<TtkhNhomGiaoDich>(entity =>

3876 │ {

3877 │ entity.HasKey(e => e.Ma);

3878 │

3879 │ entity.ToTable("ttkh\_NhomGiaoDich");

3880 │

3881 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3882 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3883 │ entity.Property(e => e.MoTa).HasMaxLength(100);

3884 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3885 │ entity.Property(e => e.Ten).HasMaxLength(50);

3886 │ });

3887 │

3888 │ modelBuilder.Entity<TtkhNoiDungDanhGia>(entity =>

3889 │ {

3890 │ entity.HasKey(e => e.Ma);

3891 │

3892 │ entity.ToTable("ttkh\_NoiDungDanhGia");

3893 │

3894 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3895 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3896 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3897 │ entity.Property(e => e.Ten).HasMaxLength(200);

3898 │ });

3899 │

3900 │ modelBuilder.Entity<TtkhThongTinKh>(entity =>

3901 │ {

3902 │ entity.HasKey(e => e.Ma);

3903 │

3904 │ entity.ToTable("ttkh\_ThongTinKH", tb => tb.HasTrigger("tg\_ttkh\_KhachHang\_ins"));

3905 │

3906 │ entity.Property(e => e.ChiTietKh).HasColumnName("ChiTietKH");

3907 │ entity.Property(e => e.ChietKhauThuongMaiMax).HasColumnType("decimal(18, 4)");

3908 │ entity.Property(e => e.ChietKhauThuongMaiMin).HasColumnType("decimal(18, 4)");

3909 │ entity.Property(e => e.ChucVu).HasMaxLength(20);

3910 │ entity.Property(e => e.ChucVuKt)

3911 │ .HasMaxLength(50)

3912 │ .HasColumnName("ChucVuKT");

3913 │ entity.Property(e => e.ChucVuNlh)

3914 │ .HasMaxLength(50)

3915 │ .HasColumnName("ChucVuNLH");

3916 │ entity.Property(e => e.Cmnd)

3917 │ .HasMaxLength(20)

3918 │ .HasColumnName("CMND");

3919 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3920 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3921 │ entity.Property(e => e.DienThoai).HasMaxLength(30);

3922 │ entity.Property(e => e.DienThoaiKt)

3923 │ .HasMaxLength(50)

3924 │ .HasColumnName("DienThoaiKT");

3925 │ entity.Property(e => e.DienThoaiNlh)

3926 │ .HasMaxLength(50)

3927 │ .HasColumnName("DienThoaiNLH");

3928 │ entity.Property(e => e.Email).HasMaxLength(50);

3929 │ entity.Property(e => e.EmailKt)

3930 │ .HasMaxLength(50)

3931 │ .HasColumnName("EmailKT");

3932 │ entity.Property(e => e.EmailNlh)

3933 │ .HasMaxLength(50)

3934 │ .HasColumnName("EmailNLH");

3935 │ entity.Property(e => e.Fax).HasMaxLength(20);

3936 │ entity.Property(e => e.FaxKt)

3937 │ .HasMaxLength(50)

3938 │ .HasColumnName("FaxKT");

3939 │ entity.Property(e => e.FaxNlh)

3940 │ .HasMaxLength(50)

3941 │ .HasColumnName("FaxNLH");

3942 │ entity.Property(e => e.HanMucNo).HasColumnType("decimal(18, 0)");

3943 │ entity.Property(e => e.HoTenKt)

3944 │ .HasMaxLength(50)

3945 │ .HasColumnName("HoTenKT");

3946 │ entity.Property(e => e.HoTenNlh)

3947 │ .HasMaxLength(50)

3948 │ .HasColumnName("HoTenNLH");

3949 │ entity.Property(e => e.Latitude).HasColumnType("decimal(18, 6)");

3950 │ entity.Property(e => e.Longitude).HasColumnType("decimal(18, 6)");

3951 │ entity.Property(e => e.MaKh)

3952 │ .HasMaxLength(50)

3953 │ .HasColumnName("MaKH");

3954 │ entity.Property(e => e.MaLoaiKh).HasColumnName("MaLoaiKH");

3955 │ entity.Property(e => e.MaNguonKh).HasColumnName("MaNguonKH");

3956 │ entity.Property(e => e.MaSoThue).HasMaxLength(50);

3957 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3958 │ entity.Property(e => e.NganHang).HasMaxLength(50);

3959 │ entity.Property(e => e.NguoiDaiDienPl)

3960 │ .HasMaxLength(100)

3961 │ .HasColumnName("NguoiDaiDienPL");

3962 │ entity.Property(e => e.NhomGd1).HasColumnName("NhomGD1");

3963 │ entity.Property(e => e.NhomGd2).HasColumnName("NhomGD2");

3964 │ entity.Property(e => e.NhomGd3).HasColumnName("NhomGD3");

3965 │ entity.Property(e => e.Photo).HasMaxLength(500);

3966 │ entity.Property(e => e.Stk)

3967 │ .HasMaxLength(50)

3968 │ .HasColumnName("STK");

3969 │ entity.Property(e => e.TenDuong).HasMaxLength(1000);

3970 │ entity.Property(e => e.TenDuongGiaoHang).HasMaxLength(50);

3971 │ entity.Property(e => e.TenKh)

3972 │ .HasMaxLength(500)

3973 │ .HasColumnName("TenKH");

3974 │ });

3975 │

3976 │ modelBuilder.Entity<TtkhThongTinKhachHangNguoiLienHe>(entity =>

3977 │ {

3978 │ entity.HasKey(e => e.Ma);

3979 │

3980 │ entity.ToTable("ttkh\_ThongTinKhachHangNguoiLienHe");

3981 │

3982 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3983 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

3984 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

3985 │ });

3986 │

3987 │ modelBuilder.Entity<V1>(entity =>

3988 │ {

3989 │ entity

3990 │ .HasNoKey()

3991 │ .ToView("v\_1");

3992 │

3993 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

3994 │ });

3995 │

3996 │ modelBuilder.Entity<VBaoCaoDanhThu>(entity =>

3997 │ {

3998 │ entity

3999 │ .HasNoKey()

4000 │ .ToView("v\_BaoCaoDanhThu");

4001 │

4002 │ entity.Property(e => e.CodeHangHoa).HasMaxLength(20);

4003 │ entity.Property(e => e.DonGiaNytb)

4004 │ .HasColumnType("decimal(18, 3)")

4005 │ .HasColumnName("DonGiaNYTB");

4006 │ entity.Property(e => e.DonGiaTb)

4007 │ .HasColumnType("decimal(18, 3)")

4008 │ .HasColumnName("DonGiaTB");

4009 │ entity.Property(e => e.GiaTri).HasColumnType("decimal(18, 0)");

4010 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

4011 │ entity.Property(e => e.NhomGd1).HasColumnName("NhomGD1");

4012 │ entity.Property(e => e.NhomGd2).HasColumnName("NhomGD2");

4013 │ entity.Property(e => e.NhomGd3).HasColumnName("NhomGD3");

4014 │ entity.Property(e => e.Sale).HasMaxLength(100);

4015 │ entity.Property(e => e.TenHangHoa).HasMaxLength(500);

4016 │ entity.Property(e => e.TenKh)

4017 │ .HasMaxLength(500)

4018 │ .HasColumnName("TenKH");

4019 │ entity.Property(e => e.TenLoaiBao).HasMaxLength(150);

4020 │ entity.Property(e => e.TenNhomGd1)

4021 │ .HasMaxLength(50)

4022 │ .HasColumnName("TenNhomGD1");

4023 │ entity.Property(e => e.TenNhomGd2)

4024 │ .HasMaxLength(50)

4025 │ .HasColumnName("TenNhomGD2");

4026 │ entity.Property(e => e.TenNhomGd3)

4027 │ .HasMaxLength(50)

4028 │ .HasColumnName("TenNhomGD3");

4029 │ });

4030 │

4031 │ modelBuilder.Entity<VBaoGia>(entity =>

4032 │ {

4033 │ entity

4034 │ .HasNoKey()

4035 │ .ToView("v\_BaoGia");

4036 │

4037 │ entity.Property(e => e.ChiPhiBocXep).HasMaxLength(500);

4038 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

4039 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

4040 │ entity.Property(e => e.DenNgay).HasColumnType("datetime");

4041 │ entity.Property(e => e.DiaDiemGiaoHang).HasMaxLength(200);

4042 │ entity.Property(e => e.GhiChu).HasMaxLength(1000);

4043 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

4044 │ entity.Property(e => e.MaNvhoTro).HasColumnName("MaNVHoTro");

4045 │ entity.Property(e => e.MaNvphuTrach).HasColumnName("MaNVPhuTrach");

4046 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

4047 │ entity.Property(e => e.NgayBaoGia).HasColumnType("datetime");

4048 │ entity.Property(e => e.SoBaoGia)

4049 │ .HasMaxLength(100)

4050 │ .IsUnicode(false);

4051 │ entity.Property(e => e.TenNguoiLienHe).HasMaxLength(200);

4052 │ entity.Property(e => e.ThoiGianGiaoHang).HasMaxLength(200);

4053 │ entity.Property(e => e.ThoiGianThanhToan).HasMaxLength(200);

4054 │ entity.Property(e => e.TuNgay).HasColumnType("datetime");

4055 │ });

4056 │

4057 │ modelBuilder.Entity<VChotLo>(entity =>

4058 │ {

4059 │ entity

4060 │ .HasNoKey()

4061 │ .ToView("v\_ChotLo");

4062 │

4063 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

4064 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

4065 │ entity.Property(e => e.GiaVonTbthang).HasColumnName("GiaVonTBThang");

4066 │ entity.Property(e => e.MaLo)

4067 │ .HasMaxLength(30)

4068 │ .IsUnicode(false);

4069 │ entity.Property(e => e.MaLoThanhPham)

4070 │ .HasMaxLength(30)

4071 │ .IsUnicode(false);

4072 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

4073 │ entity.Property(e => e.NgayCapNhatChotLo).HasColumnType("datetime");

4074 │ entity.Property(e => e.NgayChot).HasColumnType("datetime");

4075 │ entity.Property(e => e.NgayChotThuMua).HasColumnType("datetime");

4076 │ entity.Property(e => e.NgayThuMua).HasColumnType("datetime");

4077 │ entity.Property(e => e.SlbotPheQuiDoi)

4078 │ .HasColumnType("decimal(38, 4)")

4079 │ .HasColumnName("SLBotPheQuiDoi");

4080 │ entity.Property(e => e.Slnhap).HasColumnName("SLNHap");

4081 │ entity.Property(e => e.TiLeThuHoi).HasColumnType("decimal(18, 2)");

4082 │ entity.Property(e => e.TiLeThuHoiQuiDoi).HasColumnType("decimal(18, 2)");

4083 │ entity.Property(e => e.Tlsach).HasColumnName("TLSach");

4084 │ entity.Property(e => e.TongGiaMua).HasColumnType("numeric(38, 2)");

4085 │ });

4086 │

4087 │ modelBuilder.Entity<VDoanhThu>(entity =>

4088 │ {

4089 │ entity

4090 │ .HasNoKey()

4091 │ .ToView("v\_DoanhThu");

4092 │

4093 │ entity.Property(e => e.ConLai).HasColumnType("decimal(38, 0)");

4094 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

4095 │ entity.Property(e => e.LyDoDuyet).HasMaxLength(500);

4096 │ entity.Property(e => e.MaCnxuatHoaDon).HasColumnName("MaCNXuatHoaDon");

4097 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

4098 │ entity.Property(e => e.MaNvbh).HasColumnName("MaNVBH");

4099 │ entity.Property(e => e.NgayChungTu).HasColumnType("datetime");

4100 │ entity.Property(e => e.NgayHetHanThanhToan).HasColumnType("datetime");

4101 │ entity.Property(e => e.NgayKetThuc).HasColumnType("datetime");

4102 │ entity.Property(e => e.NhaVanChuyen)

4103 │ .HasMaxLength(20)

4104 │ .IsUnicode(false);

4105 │ entity.Property(e => e.PhaiThu).HasColumnType("decimal(38, 5)");

4106 │ entity.Property(e => e.SlbaoThucXuat).HasColumnName("SLBaoThucXuat");

4107 │ entity.Property(e => e.SoChungTu).HasMaxLength(100);

4108 │ entity.Property(e => e.SoHopDong).HasMaxLength(100);

4109 │ entity.Property(e => e.SoPo)

4110 │ .HasMaxLength(50)

4111 │ .HasColumnName("SoPO");

4112 │ entity.Property(e => e.SoTienThanhToan).HasColumnType("decimal(38, 0)");

4113 │ entity.Property(e => e.Ten).HasMaxLength(150);

4114 │ entity.Property(e => e.TenChiNhanhXuat).HasMaxLength(150);

4115 │ entity.Property(e => e.TenCnxuatHoaDon)

4116 │ .HasMaxLength(150)

4117 │ .HasColumnName("TenCNXuatHoaDon");

4118 │ entity.Property(e => e.TenDv)

4119 │ .HasMaxLength(30)

4120 │ .HasColumnName("TenDV");

4121 │ entity.Property(e => e.TenLyDoDuyet).HasMaxLength(1000);

4122 │ entity.Property(e => e.TenNguoiLienHe).HasMaxLength(200);

4123 │ entity.Property(e => e.TenTienTe).HasMaxLength(100);

4124 │ entity.Property(e => e.TenTrangThaiDonHang).HasMaxLength(200);

4125 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(38, 5)");

4126 │ entity.Property(e => e.TienThue).HasColumnType("decimal(38, 5)");

4127 │ });

4128 │

4129 │ modelBuilder.Entity<VHangHoa>(entity =>

4130 │ {

4131 │ entity

4132 │ .HasNoKey()

4133 │ .ToView("v\_HangHoa");

4134 │

4135 │ entity.Property(e => e.Code).HasMaxLength(20);

4136 │ entity.Property(e => e.DonGiaUsd)

4137 │ .HasColumnType("decimal(18, 5)")

4138 │ .HasColumnName("DonGiaUSD");

4139 │ entity.Property(e => e.KhoXuat)

4140 │ .HasMaxLength(50)

4141 │ .IsUnicode(false);

4142 │ entity.Property(e => e.MaDvt).HasColumnName("MaDVT");

4143 │ entity.Property(e => e.Ten).HasMaxLength(500);

4144 │ entity.Property(e => e.TenDvt)

4145 │ .HasMaxLength(30)

4146 │ .HasColumnName("TenDVT");

4147 │ entity.Property(e => e.TenLoaiHangHoa).HasMaxLength(200);

4148 │ entity.Property(e => e.TenTiengAnh).HasMaxLength(500);

4149 │ });

4150 │

4151 │ modelBuilder.Entity<VHoaDon>(entity =>

4152 │ {

4153 │ entity

4154 │ .HasNoKey()

4155 │ .ToView("v\_HoaDon");

4156 │

4157 │ entity.Property(e => e.ConLai).HasColumnType("decimal(18, 4)");

4158 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

4159 │ entity.Property(e => e.DaThanhToan).HasColumnType("decimal(18, 4)");

4160 │ entity.Property(e => e.DienThoai).HasMaxLength(30);

4161 │ entity.Property(e => e.GhiChu).HasMaxLength(200);

4162 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

4163 │ entity.Property(e => e.NgayHetHanTt)

4164 │ .HasColumnType("datetime")

4165 │ .HasColumnName("NgayHetHanTT");

4166 │ entity.Property(e => e.NgayHoaDon).HasColumnType("datetime");

4167 │ entity.Property(e => e.NgayNhanHang).HasColumnType("datetime");

4168 │ entity.Property(e => e.NhomGd1).HasColumnName("NhomGD1");

4169 │ entity.Property(e => e.PhaiThu).HasColumnType("decimal(18, 4)");

4170 │ entity.Property(e => e.SoDonHang).HasMaxLength(100);

4171 │ entity.Property(e => e.SoHoaDon)

4172 │ .HasMaxLength(30)

4173 │ .IsUnicode(false);

4174 │ entity.Property(e => e.SoPhieuXuat).HasMaxLength(30);

4175 │ entity.Property(e => e.TenKh)

4176 │ .HasMaxLength(500)

4177 │ .HasColumnName("TenKH");

4178 │ entity.Property(e => e.TenNvbh)

4179 │ .HasMaxLength(100)

4180 │ .HasColumnName("TenNVBH");

4181 │ entity.Property(e => e.TenTienTe).HasMaxLength(100);

4182 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(18, 4)");

4183 │ entity.Property(e => e.TienThue).HasColumnType("decimal(18, 4)");

4184 │ });

4185 │

4186 │ modelBuilder.Entity<VHoaDonMuaHang>(entity =>

4187 │ {

4188 │ entity

4189 │ .HasNoKey()

4190 │ .ToView("v\_HoaDonMuaHang");

4191 │

4192 │ entity.Property(e => e.ConLai).HasColumnType("decimal(18, 4)");

4193 │ entity.Property(e => e.DaThanhToan).HasColumnType("decimal(18, 4)");

4194 │ entity.Property(e => e.GhiChu).HasMaxLength(200);

4195 │ entity.Property(e => e.NgayHoaDon).HasColumnType("datetime");

4196 │ entity.Property(e => e.SoDeNghi)

4197 │ .HasMaxLength(30)

4198 │ .IsUnicode(false);

4199 │ entity.Property(e => e.SoDonHang)

4200 │ .HasMaxLength(30)

4201 │ .IsUnicode(false);

4202 │ entity.Property(e => e.SoHoaDon)

4203 │ .HasMaxLength(30)

4204 │ .IsUnicode(false);

4205 │ entity.Property(e => e.TenCn)

4206 │ .HasMaxLength(150)

4207 │ .HasColumnName("TenCN");

4208 │ entity.Property(e => e.TenNhaCc)

4209 │ .HasMaxLength(500)

4210 │ .HasColumnName("TenNhaCC");

4211 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(38, 4)");

4212 │ });

4213 │

4214 │ modelBuilder.Entity<VHopDong>(entity =>

4215 │ {

4216 │ entity

4217 │ .HasNoKey()

4218 │ .ToView("v\_HopDong");

4219 │

4220 │ entity.Property(e => e.CangXuatDen).HasMaxLength(100);

4221 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

4222 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

4223 │ entity.Property(e => e.FileDinhKem).HasMaxLength(500);

4224 │ entity.Property(e => e.GhiChu).HasMaxLength(1000);

4225 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

4226 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

4227 │ entity.Property(e => e.NgayBatDau).HasColumnType("datetime");

4228 │ entity.Property(e => e.NgayGiaHan).HasColumnType("datetime");

4229 │ entity.Property(e => e.NgayHopDong).HasColumnType("datetime");

4230 │ entity.Property(e => e.NgayKetThuc).HasColumnType("datetime");

4231 │ entity.Property(e => e.SoHopDong).HasMaxLength(100);

4232 │ entity.Property(e => e.SoHopDongDoiTac).HasMaxLength(100);

4233 │ entity.Property(e => e.TenNguoiLienHe).HasMaxLength(200);

4234 │ });

4235 │

4236 │ modelBuilder.Entity<VKhoHang>(entity =>

4237 │ {

4238 │ entity

4239 │ .HasNoKey()

4240 │ .ToView("v\_KhoHang");

4241 │

4242 │ entity.Property(e => e.TenKho).HasMaxLength(153);

4243 │ });

4244 │

4245 │ modelBuilder.Entity<VLenhXuatKho>(entity =>

4246 │ {

4247 │ entity

4248 │ .HasNoKey()

4249 │ .ToView("v\_LenhXuatKho");

4250 │

4251 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

4252 │ entity.Property(e => e.GhiChu).HasMaxLength(500);

4253 │ entity.Property(e => e.GhiChuBaoMat).HasMaxLength(600);

4254 │ entity.Property(e => e.LenhXuatKho).HasMaxLength(50);

4255 │ entity.Property(e => e.MaKh).HasColumnName("MaKH");

4256 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

4257 │ entity.Property(e => e.Name).HasMaxLength(150);

4258 │ entity.Property(e => e.NgayChungTu).HasColumnType("datetime");

4259 │ entity.Property(e => e.NhaVc)

4260 │ .HasMaxLength(200)

4261 │ .HasColumnName("NhaVC");

4262 │ entity.Property(e => e.NhomGd1).HasColumnName("NhomGD1");

4263 │ entity.Property(e => e.NhomGd2).HasColumnName("NhomGD2");

4264 │ entity.Property(e => e.SoChungTu).HasMaxLength(50);

4265 │ entity.Property(e => e.SoDonHang).HasMaxLength(100);

4266 │ entity.Property(e => e.SoXe).HasMaxLength(300);

4267 │ entity.Property(e => e.Ten).HasMaxLength(150);

4268 │ entity.Property(e => e.TenDv)

4269 │ .HasMaxLength(30)

4270 │ .HasColumnName("TenDV");

4271 │ entity.Property(e => e.TenKh)

4272 │ .HasMaxLength(500)

4273 │ .HasColumnName("TenKH");

4274 │ entity.Property(e => e.TenTrangThai).HasMaxLength(50);

4275 │ entity.Property(e => e.ThuTienMat).HasColumnType("decimal(18, 4)");

4276 │ });

4277 │

4278 │ modelBuilder.Entity<VLoaiXe>(entity =>

4279 │ {

4280 │ entity

4281 │ .HasNoKey()

4282 │ .ToView("v\_LoaiXe");

4283 │

4284 │ entity.Property(e => e.MaDvttheoTrongTai).HasColumnName("MaDVTTheoTrongTai");

4285 │ entity.Property(e => e.Ten).HasMaxLength(150);

4286 │ entity.Property(e => e.TenDvt)

4287 │ .HasMaxLength(30)

4288 │ .HasColumnName("TenDVT");

4289 │ });

4290 │

4291 │ modelBuilder.Entity<VNhapKho>(entity =>

4292 │ {

4293 │ entity

4294 │ .HasNoKey()

4295 │ .ToView("v\_NhapKho");

4296 │

4297 │ entity.Property(e => e.CaSx)

4298 │ .HasMaxLength(100)

4299 │ .HasColumnName("CaSX");

4300 │ entity.Property(e => e.Code).HasMaxLength(50);

4301 │ entity.Property(e => e.LoSanXuat).HasMaxLength(30);

4302 │ entity.Property(e => e.LoaiBao).HasMaxLength(150);

4303 │ entity.Property(e => e.LoaiBot).HasMaxLength(100);

4304 │ entity.Property(e => e.NgayBatDauSx)

4305 │ .HasColumnType("datetime")

4306 │ .HasColumnName("NgayBatDauSX");

4307 │ entity.Property(e => e.NgayDongBao).HasColumnType("datetime");

4308 │ entity.Property(e => e.NgayNgungSx)

4309 │ .HasColumnType("datetime")

4310 │ .HasColumnName("NgayNgungSX");

4311 │ entity.Property(e => e.NgayNhapXuat).HasColumnType("datetime");

4312 │ entity.Property(e => e.NgaySanXuat).HasColumnType("datetime");

4313 │ entity.Property(e => e.NgaySanXuatTrenBao).HasColumnType("datetime");

4314 │ entity.Property(e => e.NguoiPhuTrach).HasMaxLength(100);

4315 │ entity.Property(e => e.SoPhieu).HasMaxLength(30);

4316 │ entity.Property(e => e.TenCn)

4317 │ .HasMaxLength(150)

4318 │ .HasColumnName("TenCN");

4319 │ entity.Property(e => e.TenHh)

4320 │ .HasMaxLength(500)

4321 │ .HasColumnName("TenHH");

4322 │ entity.Property(e => e.TenKho).HasMaxLength(100);

4323 │ entity.Property(e => e.ViTri).HasMaxLength(200);

4324 │ });

4325 │

4326 │ modelBuilder.Entity<VPermission>(entity =>

4327 │ {

4328 │ entity

4329 │ .HasNoKey()

4330 │ .ToView("vPermission");

4331 │

4332 │ entity.Property(e => e.BranchId).HasColumnName("BranchID");

4333 │ entity.Property(e => e.CodeCv)

4334 │ .HasMaxLength(50)

4335 │ .HasColumnName("CodeCV");

4336 │ entity.Property(e => e.DepartmentId)

4337 │ .HasMaxLength(100)

4338 │ .HasColumnName("DepartmentID");

4339 │ entity.Property(e => e.Email).HasMaxLength(50);

4340 │ entity.Property(e => e.FullName).HasMaxLength(100);

4341 │ entity.Property(e => e.MenuId).HasColumnName("MenuID");

4342 │ entity.Property(e => e.PermissionId).HasColumnName("PermissionID");

4343 │ entity.Property(e => e.RoleId).HasColumnName("RoleID");

4344 │ entity.Property(e => e.UserId).HasColumnName("UserID");

4345 │ entity.Property(e => e.Username).HasMaxLength(100);

4346 │ });

4347 │

4348 │ modelBuilder.Entity<VPhieuNhapKho>(entity =>

4349 │ {

4350 │ entity

4351 │ .HasNoKey()

4352 │ .ToView("v\_PhieuNhapKho");

4353 │

4354 │ entity.Property(e => e.CaSanXuat).HasMaxLength(100);

4355 │ entity.Property(e => e.GhiChu).HasMaxLength(200);

4356 │ entity.Property(e => e.MaLoaiNx).HasColumnName("MaLoaiNX");

4357 │ entity.Property(e => e.MaNcc).HasColumnName("MaNCC");

4358 │ entity.Property(e => e.NgayNhapXuat).HasColumnType("datetime");

4359 │ entity.Property(e => e.NguoiPhuTrach).HasMaxLength(100);

4360 │ entity.Property(e => e.SoDonHang).HasMaxLength(100);

4361 │ entity.Property(e => e.SoPhieu).HasMaxLength(30);

4362 │ entity.Property(e => e.SoPhieuDeNghi)

4363 │ .HasMaxLength(30)

4364 │ .IsUnicode(false);

4365 │ entity.Property(e => e.TenLo)

4366 │ .HasMaxLength(30)

4367 │ .IsUnicode(false);

4368 │ entity.Property(e => e.TenNhaCc)

4369 │ .HasMaxLength(500)

4370 │ .HasColumnName("TenNhaCC");

4371 │ });

4372 │

4373 │ modelBuilder.Entity<VPhieuXuatKho>(entity =>

4374 │ {

4375 │ entity

4376 │ .HasNoKey()

4377 │ .ToView("v\_PhieuXuatKho");

4378 │

4379 │ entity.Property(e => e.ChiPhiBocVac).HasColumnType("decimal(18, 2)");

4380 │ entity.Property(e => e.GhiChu).HasMaxLength(200);

4381 │ entity.Property(e => e.LenhXuatKho).HasMaxLength(50);

4382 │ entity.Property(e => e.MaLoaiNx).HasColumnName("MaLoaiNX");

4383 │ entity.Property(e => e.MaNcc).HasColumnName("MaNCC");

4384 │ entity.Property(e => e.NgayGiaoHang).HasColumnType("datetime");

4385 │ entity.Property(e => e.NgayNhapXuat).HasColumnType("datetime");

4386 │ entity.Property(e => e.NguoiPhuTrach).HasMaxLength(100);

4387 │ entity.Property(e => e.SoDonHang).HasMaxLength(100);

4388 │ entity.Property(e => e.SoPhieu).HasMaxLength(30);

4389 │ entity.Property(e => e.SoXe).HasMaxLength(300);

4390 │ entity.Property(e => e.TenDv)

4391 │ .HasMaxLength(30)

4392 │ .HasColumnName("TenDV");

4393 │ entity.Property(e => e.TenKh)

4394 │ .HasMaxLength(500)

4395 │ .HasColumnName("TenKH");

4396 │ entity.Property(e => e.TenLoaiBao).HasMaxLength(150);

4397 │ entity.Property(e => e.TenNhaCc)

4398 │ .HasMaxLength(500)

4399 │ .HasColumnName("TenNhaCC");

4400 │ });

4401 │

4402 │ modelBuilder.Entity<VPhuLucHopDongGanNhat>(entity =>

4403 │ {

4404 │ entity

4405 │ .HasNoKey()

4406 │ .ToView("v\_PhuLucHopDongGanNhat");

4407 │

4408 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

4409 │ });

4410 │

4411 │ modelBuilder.Entity<VQlclChiNhanh>(entity =>

4412 │ {

4413 │ entity

4414 │ .HasNoKey()

4415 │ .ToView("v\_QLCL\_ChiNhanh");

4416 │

4417 │ entity.Property(e => e.Code).HasMaxLength(50);

4418 │ entity.Property(e => e.Id).HasColumnName("ID");

4419 │ entity.Property(e => e.Name).HasMaxLength(150);

4420 │ });

4421 │

4422 │ modelBuilder.Entity<VQlclChiTietSoLieu>(entity =>

4423 │ {

4424 │ entity

4425 │ .HasNoKey()

4426 │ .ToView("v\_QLCL\_ChiTietSoLieu");

4427 │

4428 │ entity.Property(e => e.BDoAm)

4429 │ .HasColumnType("decimal(18, 4)")

4430 │ .HasColumnName("B\_DoAm");

4431 │ entity.Property(e => e.BDoKeoBu)

4432 │ .HasColumnType("decimal(18, 4)")

4433 │ .HasColumnName("B\_DoKeoBU");

4434 │ entity.Property(e => e.BDoKeoCp)

4435 │ .HasColumnType("decimal(18, 4)")

4436 │ .HasColumnName("B\_DoKeoCP");

4437 │ entity.Property(e => e.BDoMin)

4438 │ .HasColumnType("decimal(18, 4)")

4439 │ .HasColumnName("B\_DoMin");

4440 │ entity.Property(e => e.BDoPh)

4441 │ .HasColumnType("decimal(18, 4)")

4442 │ .HasColumnName("B\_DoPH");

4443 │ entity.Property(e => e.BDoTrang)

4444 │ .HasColumnType("decimal(18, 4)")

4445 │ .HasColumnName("B\_DoTrang");

4446 │ entity.Property(e => e.BTapChat)

4447 │ .HasColumnType("decimal(18, 4)")

4448 │ .HasColumnName("B\_TapChat");

4449 │ entity.Property(e => e.BuDoAm)

4450 │ .HasColumnType("decimal(18, 4)")

4451 │ .HasColumnName("BU\_DoAm");

4452 │ entity.Property(e => e.Cepa).HasMaxLength(200);

4453 │ entity.Property(e => e.GhiChu).HasMaxLength(2000);

4454 │ entity.Property(e => e.Gio).HasMaxLength(2);

4455 │ entity.Property(e => e.Ma).ValueGeneratedOnAdd();

4456 │ entity.Property(e => e.NdHoHoa)

4457 │ .HasColumnType("decimal(18, 4)")

4458 │ .HasColumnName("ND\_HoHoa");

4459 │ entity.Property(e => e.NdLo)

4460 │ .HasColumnType("decimal(18, 4)")

4461 │ .HasColumnName("ND\_Lo");

4462 │ entity.Property(e => e.NdSay)

4463 │ .HasColumnType("decimal(18, 4)")

4464 │ .HasColumnName("ND\_Say");

4465 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

4466 │ entity.Property(e => e.NgayTao).HasColumnType("datetime");

4467 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

4468 │ entity.Property(e => e.NsDoPh)

4469 │ .HasColumnType("decimal(18, 4)")

4470 │ .HasColumnName("NS\_DoPH");

4471 │ entity.Property(e => e.Phut).HasMaxLength(2);

4472 │ entity.Property(e => e.XmsDoAm)

4473 │ .HasColumnType("decimal(18, 4)")

4474 │ .HasColumnName("XMS\_DoAm");

4475 │ entity.Property(e => e.XmtCoBot)

4476 │ .HasMaxLength(200)

4477 │ .HasColumnName("XMT\_CoBot");

4478 │ });

4479 │

4480 │ modelBuilder.Entity<VQlclGioiHanSoLieu>(entity =>

4481 │ {

4482 │ entity

4483 │ .HasNoKey()

4484 │ .ToView("v\_QLCL\_GioiHanSoLieu");

4485 │

4486 │ entity.Property(e => e.Code).HasMaxLength(50);

4487 │ entity.Property(e => e.Dvt)

4488 │ .HasMaxLength(50)

4489 │ .HasColumnName("DVT");

4490 │ entity.Property(e => e.GhiChu).HasMaxLength(2000);

4491 │ entity.Property(e => e.GiaTri).HasColumnType("decimal(18, 3)");

4492 │ entity.Property(e => e.Loai).HasMaxLength(50);

4493 │ entity.Property(e => e.Ma).ValueGeneratedOnAdd();

4494 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

4495 │ entity.Property(e => e.NgayTao).HasColumnType("datetime");

4496 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

4497 │ entity.Property(e => e.Ten).HasMaxLength(250);

4498 │ entity.Property(e => e.TenNgan).HasMaxLength(50);

4499 │ entity.Property(e => e.TenZh)

4500 │ .HasMaxLength(250)

4501 │ .HasColumnName("Ten\_ZH");

4502 │ entity.Property(e => e.TieuChuan).HasMaxLength(100);

4503 │ });

4504 │

4505 │ modelBuilder.Entity<VQlclSoLieu>(entity =>

4506 │ {

4507 │ entity

4508 │ .HasNoKey()

4509 │ .ToView("v\_QLCL\_SoLieu");

4510 │

4511 │ entity.Property(e => e.DotLo).HasMaxLength(100);

4512 │ entity.Property(e => e.Ma).ValueGeneratedOnAdd();

4513 │ entity.Property(e => e.Ngay).HasColumnType("datetime");

4514 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

4515 │ entity.Property(e => e.NgayTao).HasColumnType("datetime");

4516 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

4517 │ entity.Property(e => e.Qc)

4518 │ .HasMaxLength(100)

4519 │ .HasColumnName("QC");

4520 │ entity.Property(e => e.ThuKhoBot).HasMaxLength(100);

4521 │ entity.Property(e => e.TruongBptp)

4522 │ .HasMaxLength(100)

4523 │ .HasColumnName("TruongBPTP");

4524 │ entity.Property(e => e.TruongCa).HasMaxLength(100);

4525 │ });

4526 │

4527 │ modelBuilder.Entity<VQuanLySale>(entity =>

4528 │ {

4529 │ entity

4530 │ .HasNoKey()

4531 │ .ToView("v\_QuanLySale");

4532 │

4533 │ entity.Property(e => e.FullName).HasMaxLength(100);

4534 │ entity.Property(e => e.MaNhanVienBh).HasColumnName("MaNhanVienBH");

4535 │ entity.Property(e => e.Msnv)

4536 │ .HasMaxLength(50)

4537 │ .HasColumnName("MSNV");

4538 │ });

4539 │

4540 │ modelBuilder.Entity<VThanhToan>(entity =>

4541 │ {

4542 │ entity

4543 │ .HasNoKey()

4544 │ .ToView("v\_ThanhToan");

4545 │

4546 │ entity.Property(e => e.NgayThanhToan).HasColumnType("datetime");

4547 │ entity.Property(e => e.SoChungTu).HasMaxLength(100);

4548 │ entity.Property(e => e.SoHopDong).HasMaxLength(100);

4549 │ entity.Property(e => e.SoTien).HasColumnType("decimal(18, 0)");

4550 │ entity.Property(e => e.TenKh)

4551 │ .HasMaxLength(500)

4552 │ .HasColumnName("TenKH");

4553 │ });

4554 │

4555 │ modelBuilder.Entity<VThongTinKhachHangCongNo>(entity =>

4556 │ {

4557 │ entity

4558 │ .HasNoKey()

4559 │ .ToView("v\_ThongTinKhachHang\_CongNo");

4560 │

4561 │ entity.Property(e => e.ConLai).HasColumnType("decimal(38, 4)");

4562 │ entity.Property(e => e.CongNoTheoDonHang).HasColumnType("decimal(38, 4)");

4563 │ entity.Property(e => e.DaThanhToan).HasColumnType("decimal(38, 4)");

4564 │ entity.Property(e => e.FullName).HasMaxLength(100);

4565 │ entity.Property(e => e.HanMucNo).HasColumnType("decimal(18, 0)");

4566 │ entity.Property(e => e.MaKh1)

4567 │ .HasMaxLength(50)

4568 │ .HasColumnName("MaKH1");

4569 │ entity.Property(e => e.Makh).HasColumnName("MAKH");

4570 │ entity.Property(e => e.NgayHetHanThanhToan).HasColumnType("datetime");

4571 │ entity.Property(e => e.TenKh)

4572 │ .HasMaxLength(500)

4573 │ .HasColumnName("TenKH");

4574 │ });

4575 │

4576 │ modelBuilder.Entity<VTieuChuanKyThuatHd>(entity =>

4577 │ {

4578 │ entity

4579 │ .HasNoKey()

4580 │ .ToView("v\_TieuChuanKyThuatHD");

4581 │

4582 │ entity.Property(e => e.TenDvt)

4583 │ .HasMaxLength(30)

4584 │ .HasColumnName("TenDVT");

4585 │ entity.Property(e => e.TenTieuChuan).HasMaxLength(200);

4586 │ entity.Property(e => e.TenTieuChuanTiengAnh).HasMaxLength(200);

4587 │ entity.Property(e => e.TieuChuan).HasMaxLength(100);

4588 │ entity.Property(e => e.TieuChuanTiengAnh).HasMaxLength(100);

4589 │ });

4590 │

4591 │ modelBuilder.Entity<VVanChuyen>(entity =>

4592 │ {

4593 │ entity

4594 │ .HasNoKey()

4595 │ .ToView("v\_VanChuyen");

4596 │

4597 │ entity.Property(e => e.NgayVanChuyen).HasColumnType("datetime");

4598 │ entity.Property(e => e.SoDonHang).HasMaxLength(100);

4599 │ entity.Property(e => e.TenKh)

4600 │ .HasMaxLength(500)

4601 │ .HasColumnName("TenKH");

4602 │ entity.Property(e => e.TenNvc)

4603 │ .HasMaxLength(200)

4604 │ .HasColumnName("TenNVC");

4605 │ entity.Property(e => e.TenTienTe).HasMaxLength(100);

4606 │ entity.Property(e => e.ThanhTien).HasColumnType("decimal(38, 5)");

4607 │ });

4608 │

4609 │ modelBuilder.Entity<Xuất>(entity =>

4610 │ {

4611 │ entity

4612 │ .HasNoKey()

4613 │ .ToTable("XUẤT$");

4614 │

4615 │ entity.Property(e => e.KháchHàng)

4616 │ .HasMaxLength(255)

4617 │ .HasColumnName("Khách hàng");

4618 │ entity.Property(e => e.LoạiPhiếu)

4619 │ .HasMaxLength(255)

4620 │ .HasColumnName("Loại phiếu");

4621 │ entity.Property(e => e.Mancc)

4622 │ .HasMaxLength(255)

4623 │ .HasColumnName("MANCC");

4624 │ entity.Property(e => e.NgàyXuất)

4625 │ .HasColumnType("datetime")

4626 │ .HasColumnName("Ngày xuất");

4627 │ entity.Property(e => e.NhàCungCấpMới)

4628 │ .HasMaxLength(255)

4629 │ .HasColumnName("NHÀ CUNG CẤP MỚI");

4630 │ entity.Property(e => e.SốLệnhXuấtKho)

4631 │ .HasMaxLength(255)

4632 │ .HasColumnName("Số lệnh xuất kho");

4633 │ entity.Property(e => e.SốPhiếu)

4634 │ .HasMaxLength(255)

4635 │ .HasColumnName("Số phiếu");

4636 │ entity.Property(e => e.SốĐơnHàng)

4637 │ .HasMaxLength(255)

4638 │ .HasColumnName("Số đơn hàng");

4639 │ });

4640 │

4641 │ OnModelCreatingPartial(modelBuilder);

4642 │ }

4643 │

4644 │ partial void OnModelCreatingPartial(ModelBuilder modelBuilder);

4645 │ }

────────────────────────────────────────────────────────────

**📄 CusDoiThu.cs (CS, 47 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class CusDoiThu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public string? Logo { get; set; }

13 │

14 │ public string? Duong { get; set; }

15 │

16 │ public int? PhuongXa { get; set; }

17 │

18 │ public int? HuyenThi { get; set; }

19 │

20 │ public int? TinhThanh { get; set; }

21 │

22 │ public int? QuocGia { get; set; }

23 │

24 │ public string? MaSoThue { get; set; }

25 │

26 │ public decimal? Latitude { get; set; }

27 │

28 │ public decimal? Longitude { get; set; }

29 │

30 │ public string? GhiChu { get; set; }

31 │

32 │ public bool? Status { get; set; }

33 │

34 │ public int? UserCreated { get; set; }

35 │

36 │ public DateTime? CreatedDate { get; set; }

37 │

38 │ public int? UserModified { get; set; }

39 │

40 │ public DateTime? ModifiedDate { get; set; }

41 │

42 │ public bool? Deleted { get; set; }

43 │

44 │ public int? UserDeleted { get; set; }

45 │

46 │ public DateTime? DeletedDate { get; set; }

47 │ }

────────────────────────────────────────────────────────────

**📄 DebtCongNoKh.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class DebtCongNoKh

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? Nam { get; set; }

13 │

14 │ public int? Thang { get; set; }

15 │

16 │ public int? MaKh { get; set; }

17 │

18 │ public decimal? NoDauKy { get; set; }

19 │

20 │ public decimal? CoDauKy { get; set; }

21 │

22 │ public decimal? NoTrongKy { get; set; }

23 │

24 │ public decimal? ThanhToan { get; set; }

25 │

26 │ public decimal? NoConLai { get; set; }

27 │

28 │ public decimal? CoCuoiKy { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 DebtSoDuDauKy.cs (CS, 43 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class DebtSoDuDauKy

7 │ {

8 │ public long Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaKh { get; set; }

13 │

14 │ public string? MaKh1 { get; set; }

15 │

16 │ public DateTime? Ngay { get; set; }

17 │

18 │ public decimal? NoDauKy { get; set; }

19 │

20 │ public decimal? CoDauKy { get; set; }

21 │

22 │ public decimal? NoTrongKy { get; set; }

23 │

24 │ public decimal? CoTrongKy { get; set; }

25 │

26 │ public decimal? NoCuoiKy { get; set; }

27 │

28 │ public decimal? CoCuoiKy { get; set; }

29 │

30 │ public int? UserCreated { get; set; }

31 │

32 │ public DateTime? CreatedDate { get; set; }

33 │

34 │ public int? UserModified { get; set; }

35 │

36 │ public DateTime? ModifiedDate { get; set; }

37 │

38 │ public bool? Deleted { get; set; }

39 │

40 │ public int? UserDeleted { get; set; }

41 │

42 │ public DateTime? DeletedDate { get; set; }

43 │ }

────────────────────────────────────────────────────────────

**📄 Dmkh1.cs (CS, 17 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class Dmkh1

7 │ {

8 │ public double? Ma { get; set; }

9 │

10 │ public string? MaKh { get; set; }

11 │

12 │ public string? TenKh { get; set; }

13 │

14 │ public double? NgàyCôngNợ { get; set; }

15 │

16 │ public double? NgàyĐiĐường { get; set; }

17 │ }

────────────────────────────────────────────────────────────

**📄 Enum.cs (CS, 103 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │

3 │ namespace QuanLyChatLuong.Common

4 │ {

5 │ public enum TrangThaiChotLo

6 │ {

7 │ ChotThuMua = 1,

8 │ ChotThanhPham = 3,

9 │ ChotLo = 2

10 │ }

11 │ public enum RoleID

12 │ {

13 │ Admin = 1,

14 │ Report = 2,

15 │ Manager = 3,

16 │ Staff = 4,

17 │ GioiHanSoLieu = 1004,

18 │ XemChiTietSoLieuNM3 = 1005,

19 │ XemChiTietSoLieu = 1006,

20 │ XemSoLieuBienTinh = 1007,

21 │ RoleXemQuyen = 1008,

22 │ }

23 │ public enum Status

24 │ {

25 │ Active = 1,

26 │ Inactive = 0

27 │ }

28 │ public enum MenuID

29 │ {

30 │

31 │ file = 1,

32 │ cat = 2,

33 │ sys = 3,

34 │ sysUser = 4,

35 │ sysUserRole = 5,

36 │ sysQuyen = 6,

37 │ sysRole = 7,

38 │ sysRoleMenu = 8,

39 │ sysRoleQuyen = 9,

40 │ sysMenu = 10,

41 │ catChiNhanh = 11,

42 │ catSoLieu = 12,

43 │ catChiTietSoLieu = 13,

44 │ catGioiHanSoLieu = 14,

45 │ sysDashboard = 15,

46 │ ops = 16,

47 │ opsKiemTra = 17,

48 │ opsChiTietKiemTra = 18,

49 │ opsKiemTraKimLoai = 19,

50 │ catSoLieuBienTinh = 20,

51 │ catChiTietSoLieuBienTinh = 21,

52 │ botBienTinh = 25,

53 │ botBienTinhBTP = 27,

54 │ botTuNhien = 28,

55 │ baobi = 29,

56 │ tieuchuan = 30,

57 │ manHinhLed = 31,

58 │ }

59 │ public enum Permission

60 │ {

61 │ View = 99999,

62 │ Add = 1,

63 │ Edit = 2,

64 │ Delete = 3,

65 │ OpenStartup = 4,

66 │ ResetPassword = 5,

67 │ ChenHinhAnh = 6,

68 │ Preview = 7,

69 │ CompareReport = 8,

70 │ DoTrangBot = 9,

71 │ DoAmBot = 10,

72 │ DoMinBot = 11,

73 │ DoPHBot = 12,

74 │ DoKeoCPBot = 13,

75 │ DoKeoPUBot = 14,

76 │ TapChatBot = 15,

77 │ NhietDoLo = 16,

78 │ NhietDoSay = 17,

79 │ NhietDoHoHoa = 18,

80 │ DoPHNuocSach = 19,

81 │ DoAmBotUot = 20,

82 │ Cepa = 21,

83 │ DoAmXacMiSay = 22,

84 │ XacMiTuoiCoBot = 23,

85 │ HamLuongTB = 24,

86 │ LoginAs = 25, //db:39-34 db:8-25

87 │ ShowAll = 26, //db:39-35 db:8-26

88 │ ViewChild = 27, //db:39-36 db:8-27

89 │ UserEdit = 28, //db:39-37 db:8-28

90 │ PhanQuyen = 29, //db:39-38 db:8-29

91 │ NotCheck = 0

92 │ }

93 │

94 │ public enum MaChiNhanh

95 │ {

96 │ NM2 = 1,

97 │ NM1 = 2,

98 │ NM3 = 3,

99 │ NM4 = 4,

100 │ NM5 = 5,

101 │ NM6 = 6,

102 │ }

103 │ }

────────────────────────────────────────────────────────────

**📄 ErrorViewModel.cs (CS, 9 lines)**

1 │ namespace QuanLyChatLuong.Models

2 │ {

3 │ public class ErrorViewModel

4 │ {

5 │ public string? RequestId { get; set; }

6 │

7 │ public bool ShowRequestId => !string.IsNullOrEmpty(RequestId);

8 │ }

9 │ }

────────────────────────────────────────────────────────────

**📄 ExportToExcel.cs (CS, 898 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using Microsoft.AspNetCore.Razor.Language;

3 │ using Microsoft.EntityFrameworkCore;

4 │ using Microsoft.Extensions.Caching.Memory;

5 │ using Microsoft.SqlServer.Server;

6 │ //using OfficeOpenXml;

7 │ //using OfficeOpenXml.Style;

8 │ using Spire.Xls;

9 │ using Spire.Xls.Core.Spreadsheet.Sorting;

10 │ using System;

11 │ using System.Data;

12 │ using System.Drawing;

13 │

14 │ namespace QuanLyChatLuong.Common

15 │ {

16 │ public class ExportToExcel

17 │ {

18 │ private readonly MemoryCacheManager \_memoryCache;

19 │ public ExportToExcel(MemoryCacheManager memoryCache)

20 │ {

21 │ \_memoryCache = memoryCache;

22 │ }

23 │

24 │ public Stream Test(DataTable dt)

25 │ {

26 │ //var fileName = "Test.pdf";

27 │ //var contentType = "application/pdf";

28 │ Stream outStream = \_memoryCache.GetValueObject("WorkBook-" + 1, () =>

29 │ {

30 │ Stream outStream = new MemoryStream();

31 │ Workbook book = new Workbook();

32 │ Worksheet sheet = book.Worksheets[0];

33 │ sheet.InsertDataTable(dt, true, 1, 1);

34 │ book.ConverterSetting.SheetFitToPage = true;

35 │ //book.SaveToFile("C:\\Users\\PC\\Desktop\\insertTableToExcel.pdf", FileFormat.PDF);

36 │ book.SaveToStream(outStream, Spire.Xls.FileFormat.PDF);

37 │ outStream.Position = 0;

38 │ outStream.Seek(0, SeekOrigin.Begin);

39 │ return outStream;

40 │ }, new MemoryCacheEntryOptions().SetSlidingExpiration(TimeSpan.FromMinutes(5)));

41 │

42 │

43 │ return outStream;

44 │ // new FileStreamResult(outStream, contentType)

45 │ //{

46 │ // FileDownloadName = fileName

47 │ //};

48 │ }

49 │

50 │ public Stream TestPDF(DataTable dt, string type)

51 │ {

52 │ Stream outStream = new MemoryStream();

53 │ Workbook book = new Workbook();

54 │ Worksheet sheet = book.Worksheets[0];

55 │ sheet.InsertDataTable(dt, true, 1, 1);

56 │ sheet.PageSetup.PaperSize = PaperSizeType.PaperA4;

57 │ sheet.Range["E1:E10"].Merge();

58 │ sheet.Range["E1"].Value2 = "AAAAAAAA";

59 │ if (type == "Excel")

60 │ {

61 │ book.SaveToStream(outStream, Spire.Xls.FileFormat.Version2007);

62 │ }

63 │ else

64 │ {

65 │ book.SaveToStream(outStream, Spire.Xls.FileFormat.PDF);

66 │ }

67 │

68 │ outStream.Position = 0;

69 │ outStream.Seek(0, SeekOrigin.Begin);

70 │ return outStream;

71 │

72 │ }

73 │ public static Stream FromDataTable(DataTable dtDetail, string FileType, int startRow = 1, int startColumn = 1,

74 │ List<ColumnFormat>? ListHeaderFormat = null, List<ExcelFooter>? ListFooter = null, List<ColumnFormat>? ListC...

75 │ List<ExcelValue>? ListGiaTriRiengLe = null, bool FreezeTable = false, ePaperSize paperSize = ePaperSize.A4,

76 │ ePageOrientation pageOrientation = ePageOrientation.Portrait, PageSetup pageSetup = null)

77 │ {

78 │ Stream outStream = new MemoryStream();

79 │ Workbook book = new Workbook();

80 │ Worksheet sheet = book.Worksheets[0];

81 │ sheet.InsertDataTable(dtDetail, true, startRow, startColumn);

82 │ //sheet.Range[$"A{startRow + 1}:M{startRow + dtDetail.Rows.Count + 1}"].VerticalAlignment = VerticalAlignTyp...

83 │

84 │ for (int i = 0; i < dtDetail.Columns.Count; i++)

85 │ {

86 │ string cell = GetExcelColumnName(startColumn + i) + startRow.ToString();

87 │ ExcelFormat format = new ExcelFormat

88 │ {

89 │ HorizontalAlignment = eHorizontalAlignment.Center,

90 │ VerticalAlignment = eVerticalAlignment.Center,

91 │ WrapText = true,

92 │ IsBold = true

93 │ };

94 │ FormatRange(sheet, cell, format, book);

95 │ }

96 │

97 │ List<int> ColHeaderFormated = new List<int>();

98 │ if (ListHeaderFormat != null && ListHeaderFormat.Count > 0)

99 │ {

100 │ foreach (var item in ListHeaderFormat)

101 │ {

102 │ foreach (var c in item.ColumnIndex)

103 │ {

104 │ string cell = GetExcelColumnName(startColumn + c) + startRow.ToString();

105 │ FormatRange(sheet, cell, item.Format, book);

106 │ ColHeaderFormated.Add(c);

107 │ }

108 │ }

109 │ }

110 │

111 │ if (ListColumnFormat != null && ListColumnFormat.Count > 0)

112 │ {

113 │ foreach (var item in ListColumnFormat)

114 │ {

115 │ foreach (var c in item.ColumnIndex)

116 │ {

117 │ string col = GetExcelColumnName(c + startColumn);

118 │ string startcell = col + (startRow + 1).ToString();

119 │ string endcell = col + (startRow + 1 + dtDetail.Rows.Count).ToString();

120 │ string cell = startcell + ":" + endcell;

121 │ FormatRange(sheet, cell, item.Format, book);

122 │ }

123 │ }

124 │ }

125 │

126 │ int haveFooter = 0;

127 │ if (ListFooter != null && ListFooter.Count > 0)

128 │ {

129 │ haveFooter = 1;

130 │ foreach (var item in ListFooter)

131 │ {

132 │ foreach (var c in item.ColumnIndex)

133 │ {

134 │ string col = GetExcelColumnName(c + startColumn);

135 │ string cell = col + (startRow + 1 + dtDetail.Rows.Count).ToString();

136 │ if (item.IsFormula)

137 │ {

138 │ sheet.Range[cell].Formula = item.Text;

139 │ }

140 │ else

141 │ {

142 │ sheet.Range[cell].Value2 = item.Text;

143 │ }

144 │ if (item.Format == null)

145 │ {

146 │ item.Format = new ExcelFormat

147 │ {

148 │ IsBold = true

149 │ };

150 │ }

151 │ FormatRange(sheet, cell, item.Format, book);

152 │ }

153 │ }

154 │ }

155 │

156 │ var sc = GetExcelColumnName(startColumn);

157 │ var ec = GetExcelColumnName(startColumn + dtDetail.Columns.Count - 1);

158 │

159 │ var rang = sc + startRow.ToString() + ":" + ec + (startRow + dtDetail.Rows.Count + haveFooter).ToString();

160 │ if (rang.Split(":").Length == 2)

161 │ {

162 │ sheet.Range[rang].BorderInside(LineStyleType.Thin, System.Drawing.Color.Black);

163 │ }

164 │ sheet.Range[rang].BorderAround(LineStyleType.Thin, System.Drawing.Color.Black);

165 │ sheet.Range[rang].IsWrapText = true;

166 │ sheet.Range[rang].VerticalAlignment = VerticalAlignType.Center;

167 │ if (FreezeTable)

168 │ {

169 │ sheet.FreezePanes(startRow + 1, startColumn);

170 │ }

171 │ string startCol = GetExcelColumnName(startColumn);

172 │ var maxCol = startColumn + dtDetail.Columns.Count - 1;

173 │ var maxRow = startRow + dtDetail.Rows.Count + haveFooter;

174 │

175 │ if (ListGiaTriRiengLe != null && ListGiaTriRiengLe.Count > 0)

176 │ {

177 │ foreach (var item in ListGiaTriRiengLe)

178 │ {

179 │ string cell = item.Range.Split(':')[0];

180 │ if (!item.KhongGopO)

181 │ {

182 │ sheet.Range[item.Range].Merge();

183 │ }

184 │ if (item.IsCheckBox)

185 │ {

186 │ var address = GetAddress(item.Range);

187 │ Spire.Xls.Core.ICheckBox cb = sheet.CheckBoxes.AddCheckBox(address.Row, address.ColumnIdx, item....

188 │ cb.CheckState = item.CheckedState ? CheckState.Checked : CheckState.Unchecked;

189 │ cb.Left = item.CheckBoxLeft;

190 │ cb.Top = item.CheckBoxTop;

191 │ cb.Text = item.Value + "";

192 │ }

193 │ else if (!item.KhongDienGiaTri)

194 │ {

195 │ sheet.Range[cell].Value2 = item.Value;

196 │ }

197 │

198 │ FormatRange(sheet, item.Range, item.Format, book);

199 │ }

200 │ //Méo hiểu sao gom chung 1 for thì ko định dạng được nên phải tách ra như này chứ k...

201 │ foreach (var item in ListGiaTriRiengLe.Where(t => t.Format != null && (t.Format.Width > 0 || t.Format.He...

202 │ {

203 │ var address = GetAddress(item.Range, 1);

204 │ if (address.ColumnIdx > maxCol)

205 │ {

206 │ maxCol = GetColumnNumber(address.ColumnName);

207 │ }

208 │ if (address.Row > maxRow)

209 │ {

210 │ maxRow = address.Row;

211 │ }

212 │ if (item.Format != null)

213 │ {

214 │ if (item.Format.Width > 0)

215 │ {

216 │ sheet.SetColumnWidth(address.ColumnIdx, item.Format.Width);

217 │ }

218 │ if (item.Format.Height > 0)

219 │ {

220 │ sheet.SetRowHeight(address.Row, item.Format.Height);

221 │ }

222 │ }

223 │ }

224 │ }

225 │

226 │ string endCol = GetExcelColumnName(maxCol);

227 │ if (pageSetup != null && (pageSetup.PrintTitleColumns + "") != "AUTO")

228 │ {

229 │ sheet.PageSetup.PrintTitleColumns = pageSetup.PrintTitleColumns + "";

230 │ }

231 │ else

232 │ {

233 │ sheet.PageSetup.PrintTitleColumns = "$" + startCol + ":$" + endCol;

234 │ }

235 │

236 │ if (pageSetup != null && (pageSetup.PrintArea + "") != "AUTO")

237 │ {

238 │ sheet.PageSetup.PrintArea = pageSetup.PrintArea + "";

239 │ }

240 │ else

241 │ {

242 │ sheet.PageSetup.PrintArea = "A1:" + endCol + maxRow.ToString();

243 │ }

244 │

245 │ sheet.PageSetup.PaperSize = (PaperSizeType)paperSize;

246 │ sheet.PageSetup.Orientation = (PageOrientationType)pageOrientation;

247 │ sheet.PageSetup.FitToPagesWide = (pageSetup == null || pageSetup.FitToPagesWide == null) ? 1 : pageSetup.Fit...

248 │ sheet.PageSetup.FitToPagesTall = (pageSetup == null || pageSetup.FitToPagesTall == null) ? 0 : pageSetup.Fit...

249 │ sheet.PageSetup.LeftMargin = (pageSetup == null || pageSetup.LeftMargin == null) ? 0.25 : pageSetup.LeftMarg...

250 │ sheet.PageSetup.RightMargin = (pageSetup == null || pageSetup.RightMargin == null) ? 0.25 : pageSetup.RightM...

251 │ sheet.PageSetup.TopMargin = (pageSetup == null || pageSetup.TopMargin == null) ? 0.25 : pageSetup.TopMargin....

252 │ sheet.PageSetup.BottomMargin = (pageSetup == null || pageSetup.BottomMargin == null) ? 0.25 : pageSetup.Bott...

253 │ if (FileType == "Excel")

254 │ {

255 │ book.SaveToStream(outStream, Spire.Xls.FileFormat.Version2007);

256 │ }

257 │ else

258 │ {

259 │ book.SaveToStream(outStream, Spire.Xls.FileFormat.PDF);

260 │ }

261 │ outStream.Position = 0;

262 │ outStream.Seek(0, SeekOrigin.Begin);

263 │ return outStream;

264 │ }

265 │

266 │ public static (int ColumnIdx, string ColumnName, int Row) GetAddress(string range, int pos = 0)

267 │ {

268 │ if (range.IsEmpty())

269 │ {

270 │ return (-1, "", -1);

271 │ }

272 │ string numbers = "0123456789";

273 │ var tmp = range.Split(":");

274 │ var cell = tmp.Length == 2 ? tmp[pos] : tmp[0];

275 │ int idx = 0;

276 │ for (int i = 0; i < cell.Length; i++)

277 │ {

278 │ if (numbers.IndexOf(cell[i]) >= 0)

279 │ {

280 │ idx = i;

281 │ break;

282 │ }

283 │ }

284 │ return (GetColumnNumber(cell.Substring(0, idx)), cell.Substring(0, idx), cell.Substring(idx).ToInt());

285 │ }

286 │

287 │ public static void FormatRange(Worksheet sheet, string range, ExcelFormat format, Workbook wb)

288 │ {

289 │ if (format == null) return;

290 │ if (!format.BackgroundColor.IsEmpty())

291 │ {

292 │ sheet.Range[range].Style.Color = ColorTranslator.FromHtml(format.BackgroundColor);

293 │ }

294 │ if (!format.ForeColor.IsEmpty())

295 │ {

296 │ sheet.Range[range].Style.Font.Color = ColorTranslator.FromHtml(format.ForeColor);

297 │ }

298 │

299 │ if (format.Font != null)

300 │ {

301 │ sheet.Range[range].Style.Font.FontName = format.Font.Name;

302 │ sheet.Range[range].Style.Font.Size = format.Font.Size;

303 │ }

304 │ sheet.Range[range].Style.Font.IsBold = format.IsBold;

305 │ sheet.Range[range].Style.Font.IsItalic = format.IsItalic;

306 │ if (!format.NumberFormat.IsEmpty())

307 │ {

308 │ sheet.Range[range].NumberFormat = format.NumberFormat;

309 │ }

310 │ sheet.Range[range].IsWrapText = format.WrapText;

311 │ sheet.Range[range].Style.HorizontalAlignment = (HorizontalAlignType)format.HorizontalAlignment;

312 │ sheet.Range[range].Style.VerticalAlignment = (VerticalAlignType)format.VerticalAlignment;

313 │ var address = GetAddress(range);

314 │ if (format.Width > 0)

315 │ {

316 │ //sheet.Range[range].ColumnWidth = format.Width;

317 │ sheet.SetColumnWidth(address.ColumnIdx, format.Width);

318 │ }

319 │ if (format.Height > 0)

320 │ {

321 │ //sheet.Range[range].RowHeight = format.Height;

322 │ sheet.SetRowHeight(address.Row, format.Height);

323 │ }

324 │ if (format.Border)

325 │ {

326 │ if (range.Split(":").Length == 2)

327 │ {

328 │ sheet.Range[range].BorderInside(LineStyleType.Thin, System.Drawing.Color.Black);

329 │ }

330 │ sheet.Range[range].BorderAround(LineStyleType.Thin, System.Drawing.Color.Black);

331 │ }

332 │ if (format.ListMultiFormatCell != null && format.ListMultiFormatCell.Count > 0)

333 │ {

334 │ RichText rtx = sheet.Range[range].RichText;

335 │ foreach (var item in format.ListMultiFormatCell)

336 │ {

337 │ try

338 │ {

339 │ ExcelFont f = wb.CreateFont();

340 │ f.IsBold = item.IsBold;

341 │ f.IsItalic = item.IsItalic;

342 │ f.FontName = item.Font.FontFamily.Name;

343 │ f.Size = item.Font.Size.ToDouble();

344 │ if (!item.ForeColor.IsEmpty())

345 │ {

346 │ f.Color = ColorTranslator.FromHtml(item.ForeColor);

347 │ }

348 │ if (item.To > item.From)

349 │ {

350 │ rtx.SetFont(item.From, item.To, f);

351 │ }

352 │ }

353 │ catch { }

354 │ }

355 │ }

356 │ }

357 │ public static string GetExcelColumnName(int columnNumber)

358 │ {

359 │ string columnName = "";

360 │

361 │ while (columnNumber > 0)

362 │ {

363 │ int modulo = (columnNumber - 1) % 26;

364 │ columnName = Convert.ToChar('A' + modulo) + columnName;

365 │ columnNumber = (columnNumber - modulo) / 26;

366 │ }

367 │

368 │ return columnName;

369 │ }

370 │ public static int GetColumnNumber(string name)

371 │ {

372 │ int number = 0;

373 │ int pow = 1;

374 │ for (int i = name.Length - 1; i >= 0; i--)

375 │ {

376 │ number += (name[i] - 'A' + 1) \* pow;

377 │ pow \*= 26;

378 │ }

379 │

380 │ return number;

381 │ }

382 │

383 │ ////////////////////////////////////////////////////////////////////////

384 │ public static Stream FromTemplate(string fileName, string FileType, List<ExcelValue> ListGiaTriRiengLe, List<Lis...

385 │ {

386 │

387 │ Stream outStream = new MemoryStream();

388 │ Workbook book = new Workbook();

389 │ book.LoadFromFile(fileName);

390 │ Worksheet sheet = book.Worksheets[0];

391 │

392 │ //Điền giá trị riêng lẻ

393 │ foreach (ExcelValue item in ListGiaTriRiengLe)

394 │ {

395 │ if (item.IsImage)

396 │ {

397 │ CellRange cell = sheet.Range[item.Range];

398 │ int rowIndex = cell.Row;

399 │ int columnIndex = cell.Column;

400 │

401 │ // Tính toán kích thước và vị trí của ô

402 │ double cellWidth = cell.ColumnWidth; // Chiều rộng của ô

403 │ double cellHeight = cell.RowHeight; // Chiều cao của ô

404 │

405 │ ExcelPicture picture = sheet.Pictures.Add(item.ImageY, item.ImageX, item.ImagePath);

406 │ picture.TopRowOffset = 0; // Căn chỉnh ảnh từ hàng

407 │ picture.Width = item.WidthImage; // Đặt chiều rộng ảnh

408 │ picture.Height = item.HeightImage; // Đặt chiều cao ảnh

409 │ picture.Left += item.LeftImage;

410 │ picture.Top += item.TopImage;

411 │ // Căn chỉnh nội dung ô nếu cần

412 │ sheet.Range[item.Range].Style.VerticalAlignment = VerticalAlignType.Top;

413 │ sheet.Range[item.Range].Style.HorizontalAlignment = HorizontalAlignType.Center;

414 │ }

415 │ else if (item.IsFormula)

416 │ {

417 │ sheet.Range[item.Range].Formula = item.Value.ToString();

418 │ }

419 │ else

420 │ {

421 │ sheet.Range[item.Range].Value2 = item.Value;

422 │ }

423 │ }

424 │

425 │

426 │ int donghientai = DongBatDau;

427 │ for (int i = 0; i < ListGiaTriBang1.Count; i++)

428 │ {

429 │ donghientai++;

430 │ sheet.InsertRow(donghientai);

431 │ CellRange sourceRow = sheet.Rows[DongBatDau - 1];

432 │ CellRange destinationRange = sheet.Rows[donghientai - 1];

433 │ sourceRow.Copy(destinationRange, CopyRangeOptions.All);

434 │

435 │ for (int j = 0; j < ListGiaTriBang1[i].Count; j++)

436 │ {

437 │ string cell = ListGiaTriBang1[i][j].Range.Split(':')[0];

438 │ if (!ListGiaTriBang1[i][j].KhongGopO)

439 │ {

440 │ sheet.Range[ListGiaTriBang1[i][j].Range].Merge();

441 │ FormatRange(sheet, ListGiaTriBang1[i][j].Range, ListGiaTriBang1[i][j].Format, book);

442 │ }

443 │ if (!ListGiaTriBang1[i][j].KhongDienGiaTri)

444 │ {

445 │ sheet.Range[cell].Value2 = ListGiaTriBang1[i][j].Value;

446 │ }

447 │ }

448 │ AdjustRowHeight(sheet, donghientai - 1);

449 │ }

450 │

451 │ //Copy Header cho bảng 2 nếu có

452 │ if (!VungHeader.IsEmpty() && ListGiaTriBang2 != null && ListGiaTriBang2.Count > 0)

453 │ {

454 │ donghientai++;

455 │ //thêm dòng trống cho cách ra

456 │ sheet.InsertRow(donghientai);

457 │ sheet.Rows[donghientai - 1].RowHeight = 4.5;

458 │ donghientai++;

459 │

460 │ // Chèn các dòng mới ở vị trí insertPosition

461 │ sheet.InsertRow(donghientai, SoDongHeader);

462 │ CellRange sourceRow = sheet.Range[VungHeader];

463 │ var address1 = GetAddress(VungHeader.Split(':')[0]);

464 │ var address2 = GetAddress(VungHeader.Split(':')[1]);

465 │

466 │ CellRange destinationRange = sheet.Range[address1.ColumnName + donghientai + ":" + address2.ColumnName +...

467 │ sourceRow.Copy(destinationRange, CopyRangeOptions.All);

468 │

469 │ // Điều chỉnh chiều cao của các hàng

470 │ for (int i = address1.Row; i < address2.Row; i++)

471 │ {

472 │ sheet.Rows[donghientai].RowHeight = sheet.Rows[i].RowHeight;

473 │ donghientai++;

474 │ }

475 │ }

476 │

477 │ //Điền giá trị riêng lẻ 2

478 │ if (ListGiaTriRiengLe2 != null)

479 │ {

480 │ foreach (ExcelValue item in ListGiaTriRiengLe2)

481 │ {

482 │ if (item.IsFormula)

483 │ {

484 │ sheet.Range[item.Range].Formula = item.Value.ToString();

485 │ }

486 │ else

487 │ {

488 │ sheet.Range[item.Range].Value2 = item.Value;

489 │ }

490 │ }

491 │ }

492 │

493 │ //Điền giá trị bảng 2

494 │ if (ListGiaTriBang2 != null)

495 │ {

496 │ for (int i = 0; i < ListGiaTriBang2.Count; i++)

497 │ {

498 │ donghientai++;

499 │ sheet.InsertRow(donghientai);

500 │ CellRange sourceRow = sheet.Rows[DongBatDau - 1];

501 │ CellRange destinationRange = sheet.Rows[donghientai - 1];

502 │ sourceRow.Copy(destinationRange, CopyRangeOptions.All);

503 │

504 │ for (int j = 0; j < ListGiaTriBang2[i].Count; j++)

505 │ {

506 │ string cell = ListGiaTriBang2[i][j].Range.Split(':')[0];

507 │ if (!ListGiaTriBang2[i][j].KhongGopO)

508 │ {

509 │ sheet.Range[ListGiaTriBang2[i][j].Range].Merge();

510 │ FormatRange(sheet, ListGiaTriBang2[i][j].Range, ListGiaTriBang2[i][j].Format, book);

511 │ }

512 │ if (!ListGiaTriBang2[i][j].KhongDienGiaTri)

513 │ {

514 │ sheet.Range[cell].Value2 = ListGiaTriBang2[i][j].Value;

515 │ }

516 │ }

517 │ AdjustRowHeight(sheet, donghientai - 1);

518 │ }

519 │ }

520 │

521 │ //Xóa dòng định dạng

522 │ if (XoaDongTemplate)

523 │ {

524 │ sheet.DeleteRow(DongBatDau);

525 │ }

526 │ //AdjustRowHeight(sheet, 21);

527 │ //AutoFit column width and row height

528 │ //sheet.AllocatedRange.AutoFitColumns();

529 │ //sheet.AllocatedRange.AutoFitRows();

530 │ if (FileType == "Excel")

531 │ {

532 │ book.SaveToStream(outStream, Spire.Xls.FileFormat.Version2007);

533 │ }

534 │ else

535 │ {

536 │ book.SaveToStream(outStream, Spire.Xls.FileFormat.PDF);

537 │ }

538 │ outStream.Position = 0;

539 │ outStream.Seek(0, SeekOrigin.Begin);

540 │ return outStream;

541 │ }

542 │

543 │ public static Stream FromTemplate(string fileName, string FileType, List<ExcelValue> ListGiaTriRiengLe, DataTabl...

544 │ {

545 │

546 │ Stream outStream = new MemoryStream();

547 │ Workbook book = new Workbook();

548 │ book.LoadFromFile(fileName);

549 │ Worksheet sheet = book.Worksheets[0];

550 │

551 │

552 │

553 │ //Điền giá trị riêng lẻ

554 │ FormatGiaTriRiengLe(book, sheet, ListGiaTriRiengLe);

555 │

556 │ if (TableGiaTriBang.Rows.Count > 0)

557 │ {

558 │ sheet.InsertRow(DongBatDau + 1, TableGiaTriBang.Rows.Count);

559 │ }

560 │ sheet.InsertDataTable(TableGiaTriBang, false, DongBatDau + 1, GetColumnNumber(CotBatDau));

561 │ int cotbatdau = GetColumnNumber(CotBatDau);

562 │ int cotketthuc = GetColumnNumber(CotKetThuc);

563 │ for (int c = cotbatdau; c <= cotketthuc; c++)

564 │ {

565 │ // Lấy 2 vùng (range) cần sao chép định dạng

566 │ CellRange sourceRange = sheet.Range[string.Format("{0}{1}", GetExcelColumnName(c), DongBatDau)];

567 │ int idxCotDau = DongBatDau + (TableGiaTriBang.Rows.Count == 0 ? 0 : 1);

568 │ CellRange targetRange = sheet.Range[string.Format("{0}{1}:{2}{3}", GetExcelColumnName(c), idxCotDau, Get...

569 │

570 │ // Sao chép font

571 │ targetRange.Style.Font.FontName = sourceRange.Style.Font.FontName;

572 │ targetRange.Style.Font.Size = sourceRange.Style.Font.Size;

573 │ targetRange.Style.Font.IsBold = sourceRange.Style.Font.IsBold;

574 │ targetRange.Style.Font.IsItalic = sourceRange.Style.Font.IsItalic;

575 │ targetRange.Style.Font.Underline = sourceRange.Style.Font.Underline; // Sao chép gạch chân

576 │ targetRange.Style.Font.Color = sourceRange.Style.Font.Color;

577 │

578 │ // Sao chép căn lề

579 │ targetRange.Style.HorizontalAlignment = sourceRange.Style.HorizontalAlignment; // Căn lề ngang

580 │ targetRange.Style.VerticalAlignment = sourceRange.Style.VerticalAlignment; // Căn lề dọc

581 │

582 │ // Sao chép định dạng số

583 │ targetRange.Style.NumberFormat = sourceRange.Style.NumberFormat;

584 │

585 │ // Sao chép nền

586 │ targetRange.Style.Color = sourceRange.Style.Color;

587 │

588 │ // Sao chép viền

589 │ targetRange.Borders[BordersLineType.EdgeTop].LineStyle = sourceRange.Borders[BordersLineType.EdgeTop].Li...

590 │ targetRange.Borders[BordersLineType.EdgeBottom].LineStyle = sourceRange.Borders[BordersLineType.EdgeBott...

591 │ targetRange.Borders[BordersLineType.EdgeLeft].LineStyle = sourceRange.Borders[BordersLineType.EdgeLeft]....

592 │ targetRange.Borders[BordersLineType.EdgeRight].LineStyle = sourceRange.Borders[BordersLineType.EdgeRight...

593 │

594 │ // Sao chép độ co giãn (wrap text)

595 │ targetRange.Style.WrapText = sourceRange.Style.WrapText;

596 │

597 │ }

598 │ //Điền giá trị riêng lẻ 2

599 │ FormatGiaTriRiengLe(book, sheet, ListGiaTriRiengLe2);

600 │

601 │ //Xóa dòng định dạng

602 │ if (XoaDongTemplate)

603 │ {

604 │ sheet.DeleteRow(DongBatDau);

605 │ }

606 │ //AdjustRowHeight(sheet, 21);

607 │ //AutoFit column width and row height

608 │ //sheet.AllocatedRange.AutoFitColumns();

609 │ //sheet.AllocatedRange.AutoFitRows();

610 │ if (FileType == "Excel")

611 │ {

612 │ book.SaveToStream(outStream, Spire.Xls.FileFormat.Version2007);

613 │ }

614 │ else

615 │ {

616 │ book.SaveToStream(outStream, Spire.Xls.FileFormat.PDF);

617 │ }

618 │ outStream.Position = 0;

619 │ outStream.Seek(0, SeekOrigin.Begin);

620 │ return outStream;

621 │ }

622 │

623 │ static void FormatGiaTriRiengLe(Workbook book, Worksheet sheet, List<ExcelValue> ListGiaTriRiengLe)

624 │ {

625 │ if (ListGiaTriRiengLe != null)

626 │ {

627 │ foreach (ExcelValue item in ListGiaTriRiengLe)

628 │ {

629 │ if (item.IsImage)

630 │ {

631 │ if (!item.ImagePath.IsEmpty())

632 │ {

633 │ ExcelPicture picture = sheet.Pictures.Add(item.ImageY, item.ImageX, item.ImagePath);

634 │ picture.Width = item.WidthImage; // Chiều rộng tính bằng điểm ảnh

635 │ picture.Height = item.HeightImage; // Chiều cao tính bằng điểm ảnh

636 │ picture.Left = picture.Left + item.LeftImage;

637 │ picture.Top = picture.Top + item.TopImage;

638 │ }

639 │ }

640 │ else if (item.IsCheckBox)

641 │ {

642 │ var idxchk = 0;

643 │ if (item.RightToLeft)

644 │ {

645 │ idxchk = (item.Value + "").Length + 1;

646 │ }

647 │ var spacing = "";

648 │ if (item.CheckBoxSpacing > 0)

649 │ {

650 │ for (int i = 0; i < item.CheckBoxSpacing; i++)

651 │ {

652 │ spacing += " ";

653 │ }

654 │ }

655 │ char chkbox = item.CheckedState ? (char)254 : (char)168;

656 │ sheet.Range[item.Range].Text = item.RightToLeft ? (item.Value + spacing + chkbox) : (chkbox + sp...

657 │ RichText rtx = sheet.Range[item.Range].RichText;

658 │ try

659 │ {

660 │ ExcelFont f = book.CreateFont();

661 │ f.FontName = "Wingdings";

662 │ f.Size = item.Format != null && item.Format.Font != null ? item.Format.Font.Size : 12;

663 │ rtx.SetFont(idxchk, idxchk + 1, f);

664 │ }

665 │ catch { }

666 │

667 │ }

668 │ else

669 │ {

670 │ FormatRange(sheet, item.Range, item.Format, book);

671 │ if (!item.KhongGopO)

672 │ {

673 │ sheet.Range[item.Range].Merge();

674 │ }

675 │ if (!item.KhongDienGiaTri)

676 │ {

677 │ if (item.IsFormula)

678 │ {

679 │ sheet.Range[item.Range].Formula = item.Value.ToString();

680 │ }

681 │ else

682 │ {

683 │ sheet.Range[item.Range].Value2 = item.Value;

684 │ }

685 │ }

686 │ }

687 │ }

688 │ }

689 │ }

690 │

691 │ static void AdjustRowHeight(Worksheet sheet, int rowIndex)

692 │ {

693 │ // Điều chỉnh chiều cao hàng thủ công bằng cách đo nội dung của các ô

694 │ foreach (CellRange range in sheet.Rows[rowIndex].CellList)

695 │ {

696 │ if (range.Style.WrapText && range.Text != null && range.Text.Length > 0)

697 │ {

698 │ float fontSize = (float)range.Style.Font.Size;

699 │ FontFamily ff = new FontFamily(range.Style.Font.FontName);

700 │ Font font = new Font(ff, fontSize);

701 │ SizeF size = range.MeasureString(range.Text);

702 │ double cellWidth = 0;

703 │

704 │ CellRange mergeArea = range.MergeArea;

705 │ if (mergeArea != null && (mergeArea.Row != range.Row || mergeArea.Column != range.Column || mergeAre...

706 │ {

707 │ for (int col = mergeArea.Column; col < mergeArea.Column + mergeArea.Columns.Length; col++)

708 │ {

709 │ cellWidth += sheet.Columns[col - 1].ColumnWidth;

710 │ }

711 │ }

712 │ else

713 │ {

714 │ cellWidth += sheet.Columns[range.Column - 1].ColumnWidth;

715 │ }

716 │ cellWidth = cellWidth == 0 ? 8.43 : cellWidth;//8.43 là chiều rộng mặc định của 1 cột ...

717 │ double textWidth = 0;

718 │ using (Graphics g = Graphics.FromImage(new Bitmap(1, 1)))

719 │ {

720 │ SizeF textSize = g.MeasureString(range.Text, font);

721 │ // Điều chỉnh chiều cao hàng dựa trên chiều cao của văn bản

722 │ textWidth = textSize.Width / 0.75; // Chuyển đổi từ điểm sang pixel

723 │ }

724 │ int lines = (int)Math.Ceiling((double)(textWidth / cellWidth) / 9.4);

725 │ lines = lines == 0 ? 1 : lines;

726 │ double newHeight = lines \* size.Height;

727 │ if (newHeight > sheet.Rows[rowIndex].RowHeight)

728 │ {

729 │ sheet.Rows[rowIndex].RowHeight = lines \* size.Height;

730 │ }

731 │ }

732 │ }

733 │ }

734 │

735 │ }

736 │

737 │ public class ExcelValue

738 │ {

739 │ /// <summary>

740 │ /// Giá trị của ô

741 │ /// </summary>

742 │ public object Value { get; set; }

743 │ /// <summary>

744 │ /// Vùng dữ liệu

745 │ /// </summary>

746 │ public string Range { get; set; }

747 │ /// <summary>

748 │ /// Chỉ format ô không điền giá trị vào ô

749 │ /// </summary>

750 │ public bool KhongDienGiaTri { get; set; }

751 │ /// <summary>

752 │ /// Xác định có muốn gộp ô trong Range lại không

753 │ /// </summary>

754 │ public bool KhongGopO { get; set; } = false;

755 │ public ExcelFormat Format { get; set; }

756 │ /// <summary>

757 │ /// Giá trị là checkbox

758 │ /// </summary>

759 │ public bool IsCheckBox { get; set; } = false;

760 │ /// <summary>

761 │ /// Trạng thái ô Checkbox True, False

762 │ /// </summary>

763 │ public bool CheckedState { get; set; } = false;

764 │ /// <summary>

765 │ /// False: ô check nằm bên trái chữ, True: ô check nằm bên phải chữ

766 │ /// </summary>

767 │ public bool RightToLeft { get; set; } = false;

768 │

769 │ /// <summary>

770 │ /// Khoảng trống giữa chữ và checkbox

771 │ /// </summary>

772 │ public uint CheckBoxSpacing { get; set; } = 1;

773 │ /// <summary>

774 │ /// Giá trị là công thức excel bắt đầu bằng dấu =

775 │ /// </summary>

776 │ public bool IsFormula { get; set; } = false;

777 │ /// <summary>

778 │ /// Giá trị là hình ảnh

779 │ /// </summary>

780 │ public bool IsImage { get; set; } = false;

781 │ /// <summary>

782 │ /// ĐƯờng dẫn hình ảnh

783 │ /// </summary>

784 │ public string ImagePath { get; set; } = "";

785 │

786 │ /// <summary>

787 │ /// cột chèn hình

788 │ /// </summary>

789 │ public int ImageX { get; set; }

790 │

791 │ /// <summary>

792 │ /// Dòng chèn hình

793 │ /// </summary>

794 │ public int ImageY { get; set; }

795 │ /// <summary>

796 │ /// Chiều ngang hình

797 │ /// </summary>

798 │ public int WidthImage { get; set; }

799 │ /// <summary>

800 │ /// Chiều cao hình

801 │ /// </summary>

802 │ public int HeightImage { get; set; }

803 │

804 │ /// <summary>

805 │ /// Di chuyển hình qua trái, phải xíu

806 │ /// </summary>

807 │ public int LeftImage { get; set; } = 0;

808 │ /// <summary>

809 │ /// Di chuyển hình lên, xuống xíu

810 │ /// </summary>

811 │ public int TopImage { get; set; } = 0;

812 │

813 │ public int CheckBoxHeight { get; set; } = 15;

814 │ public int CheckBoxWidth { get; set; } = 100;

815 │ public int CheckBoxLeft { get; set; } = 0;

816 │ public int CheckBoxTop { get; set; } = 0;

817 │ }

818 │ public class ExcelFormat

819 │ {

820 │ public string NumberFormat { get; set; }

821 │ public Font Font { get; set; }

822 │ public string BackgroundColor { get; set; }

823 │ public string ForeColor { get; set; }

824 │ public bool WrapText { get; set; }

825 │ public bool IsBold { get; set; }

826 │ public bool IsItalic { get; set; }

827 │ public double Width { get; set; }

828 │ public double Height { get; set; }

829 │ public bool Border { get; set; }

830 │ public eHorizontalAlignment HorizontalAlignment { get; set; }

831 │ public eVerticalAlignment VerticalAlignment { get; set; }

832 │ public List<MultiFormatCell> ListMultiFormatCell { get; set; }

833 │ }

834 │ public class ColumnFormat

835 │ {

836 │ public List<int> ColumnIndex { get; set; }

837 │ public ExcelFormat Format { get; set; }

838 │ }

839 │ public class ExcelFooter

840 │ {

841 │ public List<int> ColumnIndex { get; set; }

842 │ public string Text { get; set; }

843 │ public bool IsFormula { get; set; }

844 │ public ExcelFormat Format { get; set; }

845 │ }

846 │

847 │ public class MultiFormatCell

848 │ {

849 │ public int From { get; set; }

850 │ public int To { get; set; }

851 │ public Font Font { get; set; }

852 │ public string ForeColor { get; set; }

853 │ public bool IsBold { get; set; }

854 │ public bool IsItalic { get; set; }

855 │ }

856 │

857 │ public class PageSetup

858 │ {

859 │ public string? PrintTitleColumns { get; set; }

860 │ public string? PrintArea { get; set; }

861 │ public int? FitToPagesWide { get; set; }

862 │ public int? FitToPagesTall { get; set; }

863 │ public double? LeftMargin { get; set; }

864 │ public double? RightMargin { get; set; }

865 │ public double? TopMargin { get; set; }

866 │ public double? BottomMargin { get; set; }

867 │ }

868 │

869 │ public enum ePaperSize

870 │ {

871 │ A4 = PaperSizeType.PaperA4,

872 │ A5 = PaperSizeType.PaperA5

873 │ }

874 │ public enum ePageOrientation

875 │ {

876 │ Portrait = PageOrientationType.Portrait,

877 │ Landscape = PageOrientationType.Landscape,

878 │ }

879 │ public enum eHorizontalAlignment

880 │ {

881 │ General = HorizontalAlignType.General,

882 │ Left = HorizontalAlignType.Left,

883 │ Center = HorizontalAlignType.Center,

884 │ Right = HorizontalAlignType.Right,

885 │ Fill = HorizontalAlignType.Fill,

886 │ Justify = HorizontalAlignType.Justify,

887 │ CenterAcrossSelection = HorizontalAlignType.CenterAcrossSelection,

888 │ Distributed = HorizontalAlignType.Distributed

889 │ }

890 │ public enum eVerticalAlignment

891 │ {

892 │ Top = VerticalAlignType.Top,

893 │ Center = VerticalAlignType.Center,

894 │ Bottom = VerticalAlignType.Bottom,

895 │ Justify = VerticalAlignType.Justify,

896 │ Distributed = VerticalAlignType.Distributed

897 │ }

898 │ }

────────────────────────────────────────────────────────────

**📄 Extension.cs (CS, 190 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc.ViewFeatures;

2 │

3 │ using Microsoft.Extensions.Caching.Memory;

4 │ using Newtonsoft.Json;

5 │ using System.Reflection;

6 │ using System.Text;

7 │ using System.Xml.Serialization;

8 │ using System.Xml;

9 │

10 │ namespace QuanLyChatLuong.Common

11 │ {

12 │ public static class Exten

13 │ {

14 │ public static string Decrypt(this string toDecrypt)

15 │ {

16 │ return Utility.Decrypt(toDecrypt);

17 │ }

18 │ public static string Encrypt(this string toEncrypt)

19 │ {

20 │ return Utility.Encrypt(toEncrypt);

21 │ }

22 │ public static string Decrypt(this int toDecrypt)

23 │ {

24 │ return Utility.Decrypt(toDecrypt + "");

25 │ }

26 │ public static string Encrypt(this int toEncrypt)

27 │ {

28 │ return Utility.Encrypt(toEncrypt + "");

29 │ }

30 │ public static bool IsEmpty(this string? str)

31 │ {

32 │ return str == null || str == "";

33 │ }

34 │

35 │ public static int? GetGuidNull(this string? str)

36 │ {

37 │ try

38 │ {

39 │ return str.IsEmpty() ? null : int.Parse(str);

40 │ }

41 │ catch

42 │ {

43 │ return Utility.defaultUID;

44 │ }

45 │ }

46 │ public static int GetGuid(this string? str)

47 │ {

48 │ try

49 │ {

50 │ return int.Parse(str);

51 │ }

52 │ catch

53 │ {

54 │ return Utility.defaultUID;

55 │ }

56 │ }

57 │ public static bool IsNullOrEmpty(this int? str)

58 │ {

59 │ try

60 │ {

61 │ return str == null || str == Utility.defaultUID;

62 │ }

63 │ catch

64 │ {

65 │ return true;

66 │ }

67 │ }

68 │ public static bool IsNotEmpty(this string str)

69 │ {

70 │ return !str.IsEmpty();

71 │ }

72 │ public static int DefaultIfNull(this int? str)

73 │ {

74 │ return str ?? Utility.defaultUID;

75 │ }

76 │ public static DateTime ParseDate(this string? str)

77 │ {

78 │ try{

79 │ return DateTime.ParseExact(str, str.Contains("-") ? "yyyy-MM-dd" : "dd/MM/yyyy", null);

80 │ }catch{

81 │ return DateTime.Now;

82 │ }

83 │ }

84 │ public delegate T CallBack<T>();

85 │ public delegate void CallBackData<T>(List<T> b);

86 │ public static T? DeepCopy<T>(this T self)

87 │ {

88 │ var serialized = JsonConvert.SerializeObject(self);

89 │ return JsonConvert.DeserializeObject<T>(serialized);

90 │ }

91 │ public class Utf8StringWriter : StringWriter

92 │ {

93 │ public override Encoding Encoding => Encoding.UTF8;

94 │ }

95 │ public static string SerializeChild<T>(this T value)

96 │ {

97 │ return Serialize(value, new XmlSerializerNamespaces(new[] { XmlQualifiedName.Empty }));

98 │ }

99 │ public static string Serialize<T>(this T value, XmlSerializerNamespaces? namespaces = null)

100 │ {

101 │ if (value == null) return string.Empty;

102 │ var xmlSerializer = new XmlSerializer(typeof(T));

103 │ using (var stringWriter = new Utf8StringWriter())

104 │ {

105 │ using (var xmlWriter = XmlWriter.Create(stringWriter, new XmlWriterSettings { Indent = true, OmitXmlDecl...

106 │ {

107 │ xmlSerializer.Serialize(xmlWriter, value, namespaces);

108 │ return stringWriter.ToString();

109 │ }

110 │ }

111 │ }

112 │ public static bool checkRole(this ViewDataDictionary<dynamic> viewData, int role)

113 │ {

114 │ return Utility.CheckRole(viewData, role);

115 │ }

116 │ public static bool checkPermis(this ViewDataDictionary<dynamic> viewData, int permission)

117 │ {

118 │ return Utility.checkTokenPermission(viewData, permission);

119 │ }

120 │ public static T getViewData<T>(this ViewDataDictionary<dynamic> viewData,string key)

121 │ {

122 │ return Utility.getViewData<T>(viewData,key);

123 │ }

124 │

125 │ private static MemoryCacheEntryOptions cacheEntryOptions = new MemoryCacheEntryOptions().SetSlidingExpiration(Ti...

126 │ private static List<string> \_key = new List<string>();

127 │ public static T GetValueObject<T>(this IMemoryCache \_memoryCache, string key, CallBack<T> callBack, MemoryCacheE...

128 │ {

129 │ if (!\_memoryCache.TryGetValue(key, out T? obj))

130 │ {

131 │ obj = callBack();

132 │ \_memoryCache.Set(key, obj, option ?? cacheEntryOptions);

133 │ if (!\_key.Any(t => t == key))

134 │ {

135 │ \_key.Add(key);

136 │ }

137 │ }

138 │ return obj ?? Activator.CreateInstance<T>();

139 │ }

140 │

141 │

142 │ public static T GetValue<T>(this IMemoryCache \_memoryCache, string key, CallBack<T> callBack, MemoryCacheEntryOp...

143 │ {

144 │ return callBack();

145 │ if (!\_memoryCache.TryGetValue(key, out T? obj))

146 │ {

147 │ obj = callBack();

148 │ \_memoryCache.Set(key, obj, option ?? cacheEntryOptions);

149 │ if (!\_key.Any(t => t == key))

150 │ {

151 │ \_key.Add(key);

152 │ }

153 │ }

154 │ return obj.DeepCopy();

155 │ }

156 │ public delegate Task<T> CallBackAsync<T>();

157 │

158 │ public static async Task<T> GetValueAsync<T>(this IMemoryCache \_memoryCache, string key, CallBackAsync<T> callBa...

159 │ {

160 │ return await callBack();

161 │ if (!\_memoryCache.TryGetValue(key, out T? obj))

162 │ {

163 │ obj = await callBack();

164 │ \_memoryCache.Set(key, obj, option ?? cacheEntryOptions);

165 │ if (!\_key.Any(t => t == key))

166 │ {

167 │ \_key.Add(key);

168 │ }

169 │ }

170 │ return obj.DeepCopy();

171 │ }

172 │ public static void Clear(this IMemoryCache \_memoryCache, Func<string, bool>? predicate)

173 │ {

174 │ List<string> keyRemove = new List<string>();

175 │ foreach (var k in predicate != null ? \_key.Where(predicate) : \_key)

176 │ {

177 │ try

178 │ {

179 │ keyRemove.Add(k);

180 │ \_memoryCache.Remove(k);

181 │ }

182 │ catch { }

183 │ }

184 │ foreach(var key in keyRemove)

185 │ {

186 │ \_key.Remove(key);

187 │ }

188 │ }

189 │ }

190 │ }

────────────────────────────────────────────────────────────

**📄 FileController.cs (CS, 35 lines)**

1 │ ﻿using Microsoft.AspNetCore.Authentication;

2 │ using Microsoft.AspNetCore.Authentication.Cookies;

3 │ using Microsoft.AspNetCore.Mvc;

4 │ using Microsoft.Extensions.Logging;

5 │ using Microsoft.IdentityModel.Tokens;

6 │ using System.IdentityModel.Tokens.Jwt;

7 │ using System.Security.Claims;

8 │ using System.Text;

9 │ using System.Threading.Tasks;

10 │ using Microsoft.EntityFrameworkCore;

11 │ using QuanLyChatLuong.Models;

12 │ using System.Security.Cryptography;

13 │ using QuanLyChatLuong.Common;

14 │ using QuanLyChatLuong.Models.APIModels;

15 │

16 │ namespace QuanLyChatLuong.Controllers

17 │ {

18 │ [Route("file")]

19 │ public class FileController : BaseController

20 │ {

21 │ public FileController(QuanLyChatLuongDbContext dbcontext, MemoryCacheManager memoryCache) : base(dbcontext, memo...

22 │ {

23 │ m\_MenuID = MenuID.file;

24 │ }

25 │ [TokenUserCheckHTTP(Permission.View)]

26 │ [HttpGet]

27 │ public IActionResult Index()

28 │ {

29 │

30 │ return View();

31 │ }

32 │

33 │

34 │ }

35 │ }

────────────────────────────────────────────────────────────

**📄 FirebaseUtility.cs (CS, 262 lines)**

1 │ ﻿using FirebaseAdmin.Messaging;

2 │ using FirebaseAdmin;

3 │ using Google.Apis.Auth.OAuth2;

4 │ using System;

5 │ using System.Collections.Generic;

6 │ using System.Linq;

7 │ using System.Web;

8 │ using System.Threading.Tasks;

9 │ using QuanLyChatLuong.Models;

10 │ using QuanLyChatLuong.Models.CRMEntity;

11 │

12 │ namespace HungDuyCRM.Common.Firebase

13 │ {

14 │ public class FirebaseUtility

15 │ {

16 │ private class \_MesThongBaoApp

17 │ {

18 │ public readonly string id;

19 │ public readonly string title;

20 │ public readonly string body;

21 │ public readonly string token;

22 │ public readonly bool enableReminder;

23 │ public readonly int badge;

24 │

25 │ public \_MesThongBaoApp(string id, string title, string body, string token, bool enableReminder, int badge)

26 │ {

27 │ this.id = id;

28 │ this.title = title;

29 │ this.body = body;

30 │ this.token = token;

31 │ this.enableReminder = enableReminder;

32 │ this.badge = badge;

33 │ }

34 │ }

35 │

36 │ public static readonly string mobileReminderKey = "\_hung\_duy\_qlcl";

37 │ private static FirebaseMessaging mess;

38 │ static FirebaseUtility()

39 │ {

40 │ var app = FirebaseApp.Create(new AppOptions()

41 │ {

42 │ Credential = GoogleCredential.FromJson("{\r\n \"type\": \"service\_account\",\r\n \"project\_id\": \"qlc...

43 │ });

44 │ mess = FirebaseMessaging.GetMessaging(app);

45 │ }

46 │

47 │ public static IEnumerable<List<T>> Chunk<T>(List<T> source, int chunkSize)

48 │ {

49 │ for (int i = 0; i < source.Count; i += chunkSize)

50 │ {

51 │ yield return source.Skip(i).Take(chunkSize).ToList();

52 │ }

53 │ }

54 │

55 │

56 │ /// <summary>

57 │ /// Dùng để tạo tin nhắn cho tất cả thiết bị di động được kết nối

58 │ /// </summary>

59 │ /// <param name="group">

60 │ /// Các thiết bị đã kết nối với group được chọn sẽ nhận được tin nhắn

61 │ /// </param>

62 │ /// <param name="action">

63 │ /// Dùng các action có trong <see cref="ReminderActionKey"/> để trigger chức năng sau khi tắt tin nh�...

64 │ /// </param>

65 │ public static async Task ReminderAction(List<string> tokens, string title, string body, string action)

66 │ {

67 │ //tokens = new List<string> { "cqYM05ybSumjfsNpfJ5WEt:APA91bFecG7aIBezA5rE97teybpAKL9HeZTPqa45hFUDmE7HbyUwop...

68 │ var chunk = Chunk(tokens, 500).ToList();

69 │ for (int k = 0; k < chunk.Count; k++)

70 │ {

71 │ List<Message> messages = new List<Message>();

72 │ for (int i = 0; i < chunk[k].Count; i++)

73 │ {

74 │ messages.Add(new Message

75 │ {

76 │ Token = chunk[k][i],

77 │ Notification = new Notification

78 │ {

79 │ Title = title,

80 │ Body = body,

81 │ },

82 │ Data = new Dictionary<string, string>

83 │ {

84 │ { "action", FirebaseActionKey.reminderAction },

85 │ { "title", title },

86 │ { "body", body },

87 │ { "reminder\_action", action },

88 │ },

89 │ Android = new AndroidConfig

90 │ {

91 │ Priority = Priority.High,

92 │ Notification = new AndroidNotification

93 │ {

94 │ Sound = "marimba", //android <26

95 │ ChannelId = "hhh\_his\_channel", //android >=26

96 │ },

97 │ },

98 │ Apns = new ApnsConfig

99 │ {

100 │ Headers = new Dictionary<string, string>()

101 │ {

102 │ { "apns-priority", "10" },

103 │ },

104 │ Aps = new Aps

105 │ {

106 │ //Sound = "marimba.wav",

107 │ ContentAvailable = true,

108 │ CriticalSound = new CriticalSound()

109 │ {

110 │ Name = "marimba.wav",

111 │ Volume = 1,

112 │ Critical = true,

113 │ },

114 │ },

115 │

116 │ },

117 │

118 │ });

119 │ }

120 │ await mess.SendEachAsync(messages);

121 │ }

122 │ }

123 │

124 │ private static List<\_MesThongBaoApp> createMessage(List<sys\_User> users, string title, string body)

125 │ {

126 │ users = users.Where(e => e.FCM != null && e.FCM != "").ToList();

127 │ List<\_MesThongBaoApp> mess = users.Select(e => new \_MesThongBaoApp("0", title, body, e.FCM, true, 0)).ToList...

128 │ return mess;

129 │ }

130 │

131 │ private static void sendMessage(List<\_MesThongBaoApp> res, string action, SoundModel sound)

132 │ {

133 │ res = res.Where(e => e.enableReminder).ToList();

134 │ var chunk = Chunk(res, 500).ToList();

135 │ for (int k = 0; k < chunk.Count; k++)

136 │ {

137 │ List<Message> messages = new List<Message>();

138 │ for (int i = 0; i < chunk[k].Count; i++)

139 │ {

140 │ var msg = chunk[k][i];

141 │ messages.Add(new Message

142 │ {

143 │ Token = msg.token,

144 │ Notification = new Notification

145 │ {

146 │ Title = msg.title,

147 │ Body = msg.body,

148 │

149 │ },

150 │ Data = new Dictionary<string, string>

151 │ {

152 │ { "action", action },

153 │ { "title", msg.title },

154 │ { "body", msg.body },

155 │ { "id" , msg.id },

156 │ { "badge" , msg.badge+"" }

157 │ },

158 │ Android = new AndroidConfig

159 │ {

160 │ Priority = Priority.High,

161 │ Notification = new AndroidNotification

162 │ {

163 │ Sound = sound.androidName, //android <26

164 │ ChannelId = sound.androidChannel, //android >=26

165 │ },

166 │ },

167 │ Apns = new ApnsConfig

168 │ {

169 │ Headers = new Dictionary<string, string>()

170 │ {

171 │ { "apns-priority", "10" },

172 │ },

173 │ Aps = new Aps

174 │ {

175 │ //Sound = "marimba.wav",

176 │ ContentAvailable = true,

177 │ CriticalSound = new CriticalSound()

178 │ {

179 │ Name = sound.iosName,

180 │ Volume = 1,

181 │ Critical = true,

182 │ },

183 │ Badge = msg.badge,

184 │ },

185 │

186 │ },

187 │

188 │ });

189 │ }

190 │ mess.SendEachAsync(messages);

191 │ }

192 │ }

193 │

194 │ public static void NotifyAction(List<sys\_User> users, string title, string body, SoundModel sound)

195 │ {

196 │ var mess = createMessage(users, title, body);

197 │ sendMessage(mess, FirebaseActionKey.notifyAction, sound);

198 │ }

199 │

200 │

201 │

202 │ public static void sendRequestLogin(SysUser User, string connectionID, string sign)

203 │ {

204 │ \_ = mess.SendAsync(new Message

205 │ {

206 │ Token = User.Fcm,//User.FCM,

207 │ Notification = new Notification

208 │ {

209 │ Title = "Yêu cầu đăng nhập",

210 │ Body = "Tài khoản của bạn đang yêu cầu đăng nhập quản lí chất lượng. Vui lòn...

211 │ ImageUrl = "https://tapioca.hungduy.vn/assets/images/logo/logoTapDoan.jpg"

212 │ },

213 │ Data = new Dictionary<string, string>

214 │ {

215 │ { "action", FirebaseActionKey.requestLogin },

216 │ { "connectionId",connectionID },

217 │ { "sign",sign }

218 │ }

219 │ });

220 │ }

221 │

222 │

223 │

224 │ /// <summary>

225 │ /// Chức năng của tin nhắn firebase

226 │ /// </summary>

227 │ public class FirebaseActionKey

228 │ {

229 │ public static string requestLogin = "request\_login";

230 │ public static string reminderAction = "reminder\_action";

231 │ public static string notifyAction = "notify\_action";

232 │ }

233 │

234 │

235 │ public class ReminderActionKey

236 │ {

237 │ // Tài xế thực hiện tiếp nhận ca cấp cứu

238 │ public static string notify = "notify";

239 │ public static string reminderNext = "reminderNext";

240 │ }

241 │

242 │ public class FirebaseSound

243 │ {

244 │ public static readonly SoundModel none = new SoundModel("", "", "default");

245 │ public static readonly SoundModel ringtone\_1 = new SoundModel("ringtone\_1", "sample\_channel", "ringtone\_1.wa...

246 │ }

247 │

248 │ public class SoundModel

249 │ {

250 │ public SoundModel(string androidName, string androidChannel, string iosName)

251 │ {

252 │ this.androidName = androidName;

253 │ this.androidChannel = androidChannel;

254 │ this.iosName = iosName;

255 │ }

256 │

257 │ public string androidName;

258 │ public string androidChannel;

259 │ public string iosName;

260 │ }

261 │ }

262 │ }

────────────────────────────────────────────────────────────

**📄 GiaVonVatTu.cs (CS, 25 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class GiaVonVatTu

7 │ {

8 │ public double? Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public double? MaDvt { get; set; }

15 │

16 │ public double? DonGiaNm1 { get; set; }

17 │

18 │ public double? DonGiaNm2 { get; set; }

19 │

20 │ public string? F7 { get; set; }

21 │

22 │ public string? F8 { get; set; }

23 │

24 │ public double? F9 { get; set; }

25 │ }

────────────────────────────────────────────────────────────

**📄 GioiHanSoLieuController.cs (CS, 188 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using Microsoft.EntityFrameworkCore;

3 │ using QuanLyChatLuong.Common;

4 │ using QuanLyChatLuong.Models;

5 │ using QuanLyChatLuong.Models.APIModels;

6 │

7 │ namespace QuanLyChatLuong.Controllers

8 │ {

9 │ [Route("catGioiHanSoLieu")]

10 │ public class GioiHanSoLieuController : BaseController

11 │ {

12 │ public GioiHanSoLieuController(QuanLyChatLuongDbContext dbcontext, MemoryCacheManager memoryCache) : base(dbcont...

13 │ {

14 │ m\_MenuID = MenuID.catGioiHanSoLieu;

15 │ }

16 │

17 │ [TokenUserCheckHTTP]

18 │ [HttpGet]

19 │ public IActionResult Index()

20 │ {

21 │

22 │ return View();

23 │ }

24 │ private bool CheckExists(int Ma, string Ten)

25 │ {

26 │ return dbcontext.tbl\_GioiHanSoLieus.Any(x => x.Ten == Ten && x.Ma != Ma && !(x.DaXoa ?? false) && x.MaChiNha...

27 │ }

28 │

29 │ private ResponseModel CheckRequire(string Ten, string Code, string GiaTri)

30 │ {

31 │ if (string.IsNullOrEmpty(Code))

32 │ {

33 │ return new ResponseModel(QLCLConst.Error, "Chưa nhập mã code");

34 │ }

35 │ if (string.IsNullOrEmpty(Ten))

36 │ {

37 │ return new ResponseModel(QLCLConst.Error, "Chưa nhập tên");

38 │ }

39 │ if (string.IsNullOrEmpty(GiaTri))

40 │ {

41 │ return new ResponseModel(QLCLConst.Error, "Chưa nhập giá trị");

42 │ }

43 │ return null;

44 │ }

45 │

46 │ [TokenUserCheckHTTP(Permission.Add)]

47 │ [HttpGet("add")]

48 │ public IActionResult Add()

49 │ {

50 │

51 │ return View("Form");

52 │ }

53 │

54 │ [TokenUserCheckHTTP()]

55 │ [HttpPost("addAPI")]

56 │ public IActionResult AddAPI(int Ma, string Ten, string Code, string TenNgan, string Ten\_ZH, string Loai, string ...

57 │ {

58 │ var result = CheckRequire(Code, Ten, GiaTri);

59 │ if (result != null)

60 │ {

61 │ return Json(result);

62 │ }

63 │

64 │ if (CheckExists(Ma, Ten))

65 │ {

66 │ return Json(new ResponseModel(QLCLConst.Error, "Tên menu đã tồn tại"));

67 │ }

68 │

69 │ var item = new tbl\_GioiHanSoLieu

70 │ {

71 │ Ten = Ten,

72 │ Code = Code,

73 │ TenNgan = TenNgan,

74 │ Ten\_ZH = Ten\_ZH,

75 │ Loai = Loai,

76 │ GiaTri = Cast.ToDecimal(GiaTri),

77 │ DVT = DVT,

78 │ TieuChuan = TieuChuan,

79 │ GhiChu = GhiChu,

80 │ TrangThai = TrangThai == "1",

81 │ NguoiTao = objToken!.userid,

82 │ NgayTao = DateTime.Now,

83 │ MaChiNhanh = (int)MaChiNhanh.NM1

84 │ };

85 │

86 │

87 │ dbcontext.tbl\_GioiHanSoLieus.Add(item);

88 │ dbcontext.SaveChanges();

89 │

90 │ return Json(new ResponseModel(QLCLConst.Successfully, "Thêm thành công"));

91 │ }

92 │

93 │ [TokenUserCheckHTTP(Permission.Edit)]

94 │ [HttpGet("edit")]

95 │ public IActionResult Edit(string id)

96 │ {

97 │ int ma = Cast.ToInt(id.Decrypt());

98 │ var item = dbcontext.tbl\_GioiHanSoLieus.FirstOrDefault(c => c.Ma == ma);

99 │

100 │ if (item == null)

101 │ {

102 │ return NotFound("Không tìm thấy");

103 │ }

104 │

105 │ ViewData["item"] = item;

106 │ return View("Form");

107 │ }

108 │

109 │

110 │ [TokenUserCheckHTTP()]

111 │ [HttpPost("updateAPI")]

112 │ public IActionResult UpdateAPI(int Ma, string Ten, string Code, string TenNgan, string Ten\_ZH, string Loai, stri...

113 │ {

114 │ var item = dbcontext.tbl\_GioiHanSoLieus.FirstOrDefault(x => x.Ma == Ma && !(x.DaXoa ?? false));

115 │ if (item == null)

116 │ {

117 │ return Json(new ResponseModel(QLCLConst.Error, " Không tồn tại"));

118 │ }

119 │ if (item.Ten != Ten && dbcontext.tbl\_GioiHanSoLieus.Any(x => x.Ten == Ten && x.Ma != Ma))

120 │ {

121 │ return Json(new ResponseModel(QLCLConst.Error, "Tên đã tồn tại"));

122 │ }

123 │

124 │ item.Ten = Ten;

125 │ item.Code = Code;

126 │ item.Ten\_ZH = Ten\_ZH;

127 │ item.TenNgan = TenNgan;

128 │ item.Loai = Loai;

129 │ item.GiaTri = Cast.ToDecimal(GiaTri);

130 │ item.DVT = DVT;

131 │ item.GhiChu = GhiChu;

132 │ item.TieuChuan = TieuChuan;

133 │ item.TrangThai = TrangThai == "1";

134 │ item.NguoiSua = objToken!.userid;

135 │ item.NgaySua = DateTime.Now;

136 │ item.MaChiNhanh = (int)MaChiNhanh.NM1;

137 │

138 │ try

139 │ {

140 │ dbcontext.tbl\_GioiHanSoLieus.Update(item);

141 │ dbcontext.SaveChanges();

142 │ }

143 │ catch (Exception ex)

144 │ {

145 │ return Json(new ResponseModel(QLCLConst.Error, "Cập nhật thất bại: " + ex.Message));

146 │ }

147 │

148 │ return Json(new ResponseModel(QLCLConst.Successfully, "Cập nhật thành công"));

149 │ }

150 │

151 │ [HttpPost("DeleteAPI")]

152 │ [TokenUserCheckHTTP(Permission.Delete)]

153 │ public IActionResult DeleteAPI(string eMa)

154 │ {

155 │ var item = dbcontext.tbl\_GioiHanSoLieus.FirstOrDefault(x => x.Ma == Cast.ToInt(eMa.Decrypt()));

156 │ if (item == null)

157 │ {

158 │ return Json(QLCLConst.ResponseNotExist);

159 │ }

160 │ item.DaXoa = true;

161 │ dbcontext.SaveChanges();

162 │ return Json(QLCLConst.ResponseDeleteSuccessfully);

163 │ }

164 │

165 │ [TokenUserCheckHTTP(Permission.View)]

166 │ [HttpGet("getListGioiHanSolieu")]

167 │ public IActionResult getListGioiHanSolieu()

168 │ {

169 │ var listData = dbcontext.tbl\_GioiHanSoLieus.Where(x => !(x.DaXoa ?? false) && x.MaChiNhanh == (int)MaChiNhan...

170 │ .Select(t => new

171 │ {

172 │ Ma = t.Ma,

173 │ eMa = t.Ma.Encrypt(),

174 │ Code = t.Code,

175 │ Ten = t.Ten,

176 │ TenNgan = t.TenNgan,

177 │ Ten\_ZH = t.Ten\_ZH,

178 │ Loai = t.Loai,

179 │ GiaTri = t.GiaTri,

180 │ DVT = t.DVT,

181 │ TieuChuan = t.TieuChuan,

182 │ GhiChu = t.GhiChu,

183 │ TrangThai = t.TrangThai,

184 │ }).ToList();

185 │ return Json(new ResponseModel(QLCLConst.Successfully, "", listData));

186 │ }

187 │ }

188 │ }

────────────────────────────────────────────────────────────

**📄 HisChamSocKhachHang.cs (CS, 53 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class HisChamSocKhachHang

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public string? SoNhatKy { get; set; }

13 │

14 │ public int? MaKh { get; set; }

15 │

16 │ public string? TenKh { get; set; }

17 │

18 │ public int? MaBaoGia { get; set; }

19 │

20 │ public int? MaHopDong { get; set; }

21 │

22 │ public DateTime? NgayNhatKy { get; set; }

23 │

24 │ public int? MaNguonThongTinNhan { get; set; }

25 │

26 │ public int? MaLoaiThongTinNhan { get; set; }

27 │

28 │ public string? NoiDung { get; set; }

29 │

30 │ public string? XuLy { get; set; }

31 │

32 │ public string? KetQua { get; set; }

33 │

34 │ public string? ChiDaoBld { get; set; }

35 │

36 │ public bool? Status { get; set; }

37 │

38 │ public int? UserCreated { get; set; }

39 │

40 │ public DateTime? CreatedDate { get; set; }

41 │

42 │ public int? UserModified { get; set; }

43 │

44 │ public DateTime? ModifiedDate { get; set; }

45 │

46 │ public bool? Deleted { get; set; }

47 │

48 │ public int? UserDeleted { get; set; }

49 │

50 │ public DateTime? DeletedDate { get; set; }

51 │

52 │ public DateTime? NgayKetThuc { get; set; }

53 │ }

────────────────────────────────────────────────────────────

**📄 HisLoaiThongTin.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class HisLoaiThongTin

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public bool? Status { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 HisNguonThongTin.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class HisNguonThongTin

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public bool? Status { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 HisNhatKyKhachHang.cs (CS, 35 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class HisNhatKyKhachHang

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public DateTime? NgayNhatKy { get; set; }

13 │

14 │ public string? TenKh { get; set; }

15 │

16 │ public string? Sdt { get; set; }

17 │

18 │ public string? DiaChi { get; set; }

19 │

20 │ public string? NoiDung { get; set; }

21 │

22 │ public int? UserCreated { get; set; }

23 │

24 │ public DateTime? CreatedDate { get; set; }

25 │

26 │ public int? UserModified { get; set; }

27 │

28 │ public DateTime? ModifiedDate { get; set; }

29 │

30 │ public bool? Deleted { get; set; }

31 │

32 │ public int? UserDeleted { get; set; }

33 │

34 │ public DateTime? DeletedDate { get; set; }

35 │ }

────────────────────────────────────────────────────────────

**📄 HomeController.cs (CS, 32 lines)**

1 │ using System.Diagnostics;

2 │ using Microsoft.AspNetCore.Authorization;

3 │ using Microsoft.AspNetCore.Mvc;

4 │ using QuanLyChatLuong.Common;

5 │ using QuanLyChatLuong.Models;

6 │

7 │ namespace QuanLyChatLuong.Controllers

8 │ {

9 │ public class HomeController : BaseController

10 │ {

11 │ public HomeController(QuanLyChatLuongDbContext dbcontext, MemoryCacheManager memoryCache) : base(dbcontext, memo...

12 │ {

13 │ }

14 │ [TokenUserCheckHTTP]

15 │ public IActionResult Index()

16 │ {

17 │

18 │ return View();

19 │ }

20 │

21 │ public IActionResult Privacy()

22 │ {

23 │ return View();

24 │ }

25 │

26 │ [ResponseCache(Duration = 0, Location = ResponseCacheLocation.None, NoStore = true)]

27 │ public IActionResult Error()

28 │ {

29 │ return View(new ErrorViewModel { RequestId = Activity.Current?.Id ?? HttpContext.TraceIdentifier });

30 │ }

31 │ }

32 │ }

────────────────────────────────────────────────────────────

**📄 ImportSdSpp.cs (CS, 61 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class ImportSdSpp

7 │ {

8 │ public string? SốHóaĐơn { get; set; }

9 │

10 │ public string? SốĐơnHàng { get; set; }

11 │

12 │ public double? MaDonHang { get; set; }

13 │

14 │ public string? MãKh { get; set; }

15 │

16 │ public double? MaKh { get; set; }

17 │

18 │ public string? MãChiNhánh { get; set; }

19 │

20 │ public string? ChiNhánh { get; set; }

21 │

22 │ public double? MaChiNhanh { get; set; }

23 │

24 │ public string? MãPhiếuXuất { get; set; }

25 │

26 │ public DateTime? NgàyHóaĐơn { get; set; }

27 │

28 │ public DateTime? NgàyNhậnHàng { get; set; }

29 │

30 │ public DateTime? NgàyĐếnHạnTt { get; set; }

31 │

32 │ public string? MãTiềnTệ { get; set; }

33 │

34 │ public double? TỷGiá { get; set; }

35 │

36 │ public double? TiềnHàng { get; set; }

37 │

38 │ public double? MãThuế { get; set; }

39 │

40 │ public double? TiềnThuế { get; set; }

41 │

42 │ public double? ThànhTiền { get; set; }

43 │

44 │ public double? SốLượng { get; set; }

45 │

46 │ public double? ĐơnGiá { get; set; }

47 │

48 │ public double? PhảiThu { get; set; }

49 │

50 │ public double? ĐãThu { get; set; }

51 │

52 │ public double? CònLại { get; set; }

53 │

54 │ public double? NgàyĐiĐường { get; set; }

55 │

56 │ public double? NgàyCôngNợ { get; set; }

57 │

58 │ public string? GhiChú { get; set; }

59 │

60 │ public string? TrạngThái { get; set; }

61 │ }

────────────────────────────────────────────────────────────

**📄 ImportSoDu.cs (CS, 61 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class ImportSoDu

7 │ {

8 │ public string? SốHóaĐơn { get; set; }

9 │

10 │ public string? SốĐơnHàng { get; set; }

11 │

12 │ public double? MaDonHang { get; set; }

13 │

14 │ public string? MãKh { get; set; }

15 │

16 │ public double? Idkh { get; set; }

17 │

18 │ public string? MãChiNhánh { get; set; }

19 │

20 │ public string? ChiNhánh { get; set; }

21 │

22 │ public double? MaChiNhanh { get; set; }

23 │

24 │ public string? MãPhiếuXuất { get; set; }

25 │

26 │ public DateTime? NgàyHóaĐơn { get; set; }

27 │

28 │ public DateTime? NgàyNhậnHàng { get; set; }

29 │

30 │ public DateTime? NgàyĐếnHạnTt { get; set; }

31 │

32 │ public string? MãTiềnTệ { get; set; }

33 │

34 │ public double? TỷGiá { get; set; }

35 │

36 │ public double? TiềnHàng { get; set; }

37 │

38 │ public double? MãThuế { get; set; }

39 │

40 │ public double? TiềnThuế { get; set; }

41 │

42 │ public double? ThànhTiền { get; set; }

43 │

44 │ public double? SốLượng { get; set; }

45 │

46 │ public double? ĐơnGiá { get; set; }

47 │

48 │ public double? PhảiThu { get; set; }

49 │

50 │ public double? ĐãThu { get; set; }

51 │

52 │ public double? CònLại { get; set; }

53 │

54 │ public double? NgàyĐiĐường { get; set; }

55 │

56 │ public double? NgàyCôngNợ { get; set; }

57 │

58 │ public string? GhiChú { get; set; }

59 │

60 │ public string? TrạngThái { get; set; }

61 │ }

────────────────────────────────────────────────────────────

**📄 InvChiTietHoaDon.cs (CS, 53 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class InvChiTietHoaDon

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaHoaDon { get; set; }

11 │

12 │ public int? MaHangHoa { get; set; }

13 │

14 │ public int? MaDvt { get; set; }

15 │

16 │ public int? SoLuong { get; set; }

17 │

18 │ public int? SoLuongBao { get; set; }

19 │

20 │ public int? MaLoaiBao { get; set; }

21 │

22 │ public int? MaLoaiBot { get; set; }

23 │

24 │ public decimal? DonGiaBan { get; set; }

25 │

26 │ public decimal? GiamGia { get; set; }

27 │

28 │ public decimal? GiaVon { get; set; }

29 │

30 │ public decimal? ThueVat { get; set; }

31 │

32 │ public decimal? TienThue { get; set; }

33 │

34 │ public decimal? ThanhTien { get; set; }

35 │

36 │ public decimal? PhaiThu { get; set; }

37 │

38 │ public string? GhiChu { get; set; }

39 │

40 │ public int? UserCreated { get; set; }

41 │

42 │ public DateTime? CreatedDate { get; set; }

43 │

44 │ public int? UserModified { get; set; }

45 │

46 │ public DateTime? ModifiedDate { get; set; }

47 │

48 │ public bool? Deleted { get; set; }

49 │

50 │ public int? UserDeleted { get; set; }

51 │

52 │ public DateTime? DeletedDate { get; set; }

53 │ }

────────────────────────────────────────────────────────────

**📄 InvHoaDon.cs (CS, 71 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class InvHoaDon

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanhBan { get; set; }

11 │

12 │ public string? SoHoaDon { get; set; }

13 │

14 │ public int? MaDonHang { get; set; }

15 │

16 │ public string? SoDonHangDauKy { get; set; }

17 │

18 │ public int? MaKh { get; set; }

19 │

20 │ public int? MaChiNhanhXuat { get; set; }

21 │

22 │ public int? MaPhieuXuat { get; set; }

23 │

24 │ public DateTime? NgayHoaDon { get; set; }

25 │

26 │ public DateTime? NgayNhanHang { get; set; }

27 │

28 │ public DateTime? NgayHetHanTt { get; set; }

29 │

30 │ public int? MaTienTe { get; set; }

31 │

32 │ public int? TyGia { get; set; }

33 │

34 │ public decimal? ThueVat { get; set; }

35 │

36 │ public decimal? TienThue { get; set; }

37 │

38 │ public decimal? ThanhTien { get; set; }

39 │

40 │ public decimal? PhaiThu { get; set; }

41 │

42 │ public decimal? DaThanhToan { get; set; }

43 │

44 │ public decimal? ConLai { get; set; }

45 │

46 │ public decimal? ChenhLechTyGia { get; set; }

47 │

48 │ public int? NgayDiDuong { get; set; }

49 │

50 │ public int? NgayCongNo { get; set; }

51 │

52 │ public string? GhiChu { get; set; }

53 │

54 │ public int? TrangThai { get; set; }

55 │

56 │ public int? LoaiHoaDon { get; set; }

57 │

58 │ public int? UserCreated { get; set; }

59 │

60 │ public DateTime? CreatedDate { get; set; }

61 │

62 │ public int? UserModified { get; set; }

63 │

64 │ public DateTime? ModifiedDate { get; set; }

65 │

66 │ public bool? Deleted { get; set; }

67 │

68 │ public int? UserDeleted { get; set; }

69 │

70 │ public DateTime? DeletedDate { get; set; }

71 │ }

────────────────────────────────────────────────────────────

**📄 InvLoaiHoaDon.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class InvLoaiHoaDon

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public bool? Status { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 InvTrangThaiHoaDon.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class InvTrangThaiHoaDon

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public bool? TrangThai { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 KiemTraChatLuongController.cs (CS, 721 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using Microsoft.EntityFrameworkCore;

3 │ using Newtonsoft.Json;

4 │ using QuanLyChatLuong.Common;

5 │ using QuanLyChatLuong.Models;

6 │ using QuanLyChatLuong.Models.APIModels;

7 │ using QuanLyChatLuong.Models.CRMEntity;

8 │ using System.Data;

9 │ using System.Globalization;

10 │ using static QuanLyChatLuong.Controllers.BotBienTinhController;

11 │ using Utility = QuanLyChatLuong.Common.Utility;

12 │

13 │ namespace QuanLyChatLuong.Controllers

14 │ {

15 │ [Route("opsKiemTra")]

16 │ public class KiemTraChatLuongController : BaseController

17 │ {

18 │ private readonly IWebHostEnvironment \_hostingEnvironment;

19 │ private CrmContext crmContext { get; }

20 │

21 │ public KiemTraChatLuongController(QuanLyChatLuongDbContext dbcontext, IWebHostEnvironment hostingEnvironment, Me...

22 │ {

23 │ m\_MenuID = MenuID.opsKiemTra;

24 │ \_hostingEnvironment = hostingEnvironment;

25 │ crmContext = crmdbcontext;

26 │ }

27 │

28 │

29 │ public class KiemTraChatLuong

30 │ {

31 │ public int? KiemTraID { get; set; }

32 │ public string? ListCT { get; set; }

33 │ public string? TenSanPham { get; set; }

34 │ public string? SoDonHang { get; set; }

35 │ public string? TenKhachHang { get; set; }

36 │ public string? NgayGiaoHang { get; set; }

37 │ public string? NgaySanXuat { get; set; }

38 │ public string? SoLuong { get; set; }

39 │ public string? NguoiKiemTra { get; set; }

40 │ public string? HanSuDung { get; set; }

41 │ public string? ChatLuongBot { get; set; }

42 │ public string? KetLuan { get; set; }

43 │ public string? GhiChu { get; set; }

44 │ public string? MaLo { get; set; }

45 │ public string? KichThuoc { get; set; }

46 │ public string? TieuChuan { get; set; }

47 │ public string? VatLieuBaoBi { get; set; }

48 │ public string? Status { get; set; }

49 │ }

50 │ public class ChiTiet

51 │ {

52 │ public int? CT { get; set; }

53 │ public int? Ma { get; set; }

54 │ public string? No { get; set; }

55 │ public string? TestItem { get; set; }

56 │ public string? Unit { get; set; }

57 │ public string? Specification { get; set; }

58 │ public string? Method { get; set; }

59 │ public string? Results { get; set; }

60 │ }

61 │ public class ChiTietTieuChuanDTO

62 │ {

63 │ public string TenSanPham { get; set; }

64 │ public string TieuChuan { get; set; }

65 │ public string KichThuoc { get; set; }

66 │ public string VatLieuBaoBi { get; set; }

67 │ }

68 │ public class DeleteDTO

69 │ {

70 │ public string eMa { get; set; }

71 │ }

72 │ [TokenUserCheckHTTP]

73 │ [HttpGet]

74 │ public IActionResult Index()

75 │ {

76 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permission...

77 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permissio...

78 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permiss...

79 │ return View();

80 │ }

81 │

82 │ [TokenUserCheckHTTP(Permission.Add)]

83 │ [HttpGet("add")]

84 │ public IActionResult Add()

85 │ {

86 │ List<tbl\_ChiNhanh> listCN = dbcontext.tbl\_ChiNhanhs

87 │ .Where(x => !(x.DaXoa ?? false) && x.Ma == objToken!.branch)

88 │ .ToList();

89 │

90 │ var chiNhanh = dbcontext.tbl\_ChiNhanhs.FirstOrDefault(x => x.Ma == objToken!.branch);

91 │

92 │ string codeChiNhanh = chiNhanh?.Code ?? "";

93 │

94 │ var listHangHoa = crmContext.VHangHoa.Where(x => x.HangHoa ?? false).ToList();

95 │

96 │ var listTieuChuan = (from tc in dbcontext.bt\_TieuChuans

97 │ join bb in dbcontext.bt\_BaoBis on tc.MaVatLieuBaoBi equals bb.Ma

98 │ join cn in dbcontext.tbl\_ChiNhanhs.Where(t => t.Code == codeChiNhanh && !(t.DaXoa ?? fa...

99 │ on tc.MaChiNhanh equals cn.Ma

100 │ where !(tc.DaXoa ?? false)

101 │ select new TieuChuanViewModel

102 │ {

103 │ MaSanPham = tc.MaSanPham,

104 │ KichThuoc = bb.Ten,

105 │ VatLieuBaoBi = bb.VatLieuBaoBi,

106 │ TieuChuan = tc.TieuChuan,

107 │ STT = tc.No

108 │ }).ToList();

109 │

110 │ var combineHangHoaTieuChuan = (from hh in listHangHoa

111 │ join tc in listTieuChuan on hh.Ma equals tc.MaSanPham

112 │ select new TieuChuanViewModel

113 │ {

114 │ STT = tc.STT,

115 │ KichThuoc = tc.KichThuoc,

116 │ VatLieuBaoBi = tc.VatLieuBaoBi,

117 │ TieuChuan = tc.TieuChuan,

118 │ TenSanPham = hh.Ten,

119 │ TenSanPhamTiengAnh = hh.TenTiengAnh,

120 │ }).OrderBy(x => x.STT)

121 │ .ToList();

122 │ var listBaoBi = dbcontext.bt\_BaoBis.Where(x => !(x.DaXoa ?? false) && x.MaChiNhanh == objToken.branch).Order...

123 │

124 │ ViewData["listTieuChuan"] = combineHangHoaTieuChuan;

125 │

126 │ ViewData["listCN"] = listCN;

127 │ ViewData["listBaoBi"] = listBaoBi;

128 │

129 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permission...

130 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permissio...

131 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permiss...

132 │ return View("Form");

133 │ }

134 │

135 │ [TokenUserCheckHTTP(Permission.Edit)]

136 │ [HttpGet("update")]

137 │ public IActionResult Edit(string Ma)

138 │ {

139 │ var dataKiemTra = dbcontext.tbl\_KiemTraChatLuongs.FirstOrDefault(x => !(x.DaXoa ?? false) && x.Ma == Cast.To...

140 │

141 │ if (dataKiemTra == null) return NotFound("KiemTraChatLuong not found");

142 │

143 │ var chiTiet = dbcontext.tbl\_ChiTietKiemTras

144 │ .Where(x => x.MaKTCL == Cast.ToInt(Ma.Decrypt())

145 │ && !(x.DaXoa ?? false))

146 │ .OrderBy(x => x.STT).ToList();

147 │

148 │ List<tbl\_ChiNhanh> listCN = dbcontext.tbl\_ChiNhanhs

149 │ .Where(x => !(x.DaXoa ?? false) && x.Ma == objToken!.branch)

150 │ .ToList();

151 │

152 │ var chiNhanh = dbcontext.tbl\_ChiNhanhs.FirstOrDefault(x => x.Ma == objToken!.branch);

153 │

154 │ string codeChiNhanh = chiNhanh?.Code ?? "";

155 │

156 │ var listHangHoa = crmContext.VHangHoa.Where(x => x.HangHoa ?? false).ToList();

157 │

158 │ var listTieuChuan = (from tc in dbcontext.bt\_TieuChuans

159 │ join bb in dbcontext.bt\_BaoBis on tc.MaVatLieuBaoBi equals bb.Ma

160 │ join cn in dbcontext.tbl\_ChiNhanhs.Where(t => t.Code == codeChiNhanh && !(t.DaXoa ?? fa...

161 │ on tc.MaChiNhanh equals cn.Ma

162 │ where !(tc.DaXoa ?? false)

163 │ select new TieuChuanViewModel

164 │ {

165 │ MaSanPham = tc.MaSanPham,

166 │ KichThuoc = bb.Ten,

167 │ VatLieuBaoBi = bb.VatLieuBaoBi,

168 │ TieuChuan = tc.TieuChuan,

169 │ }).ToList();

170 │

171 │ var combineHangHoaTieuChuan = (from hh in listHangHoa

172 │ join tc in listTieuChuan on hh.Ma equals tc.MaSanPham

173 │ select new TieuChuanViewModel

174 │ {

175 │ KichThuoc = tc.KichThuoc,

176 │ VatLieuBaoBi = tc.VatLieuBaoBi,

177 │ TieuChuan = tc.TieuChuan,

178 │ TenSanPham = hh.Ten,

179 │ TenSanPhamTiengAnh = hh.TenTiengAnh,

180 │ }).ToList();

181 │ var listBaoBi = dbcontext.bt\_BaoBis.Where(x => !(x.DaXoa ?? false) && x.MaChiNhanh == objToken.branch).Order...

182 │

183 │ ViewData["listTieuChuan"] = combineHangHoaTieuChuan;

184 │ ViewData["listCN"] = listCN;

185 │

186 │ ViewData["item"] = dataKiemTra;

187 │ ViewData["listChiTietKiemTra"] = chiTiet;

188 │ ViewData["listBaoBi"] = listBaoBi;

189 │

190 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permission...

191 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permissio...

192 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2(objToken.userid, (int)MenuID.botBienTinh, (int)Permiss...

193 │

194 │ return View("Form");

195 │ }

196 │

197 │ [TokenUserCheckHTTP()]

198 │ [HttpPost("addAPI")]

199 │ public async Task<IActionResult> AddAPI([FromBody] KiemTraChatLuong model)

200 │ {

201 │ int idUser = objToken.userid;

202 │ int idBranch = objToken.branch;

203 │

204 │ if (idUser == 0 || idBranch == 0)

205 │ {

206 │ return Json(new ResponseModel("Error", "Hết phiên đăng nhập. Vui lòng đăng nhập lại."));

207 │ }

208 │

209 │ if (string.IsNullOrWhiteSpace(model.SoDonHang) ||

210 │ string.IsNullOrWhiteSpace(model.MaLo) ||

211 │ string.IsNullOrWhiteSpace(model.TenKhachHang) ||

212 │ string.IsNullOrWhiteSpace(model.HanSuDung) ||

213 │ string.IsNullOrWhiteSpace(model.ChatLuongBot) ||

214 │ string.IsNullOrWhiteSpace(model.KetLuan) ||

215 │ string.IsNullOrWhiteSpace(model.SoLuong?.ToString()) ||

216 │ string.IsNullOrWhiteSpace(model.TenSanPham) ||

217 │ string.IsNullOrWhiteSpace(model.TieuChuan) ||

218 │ string.IsNullOrWhiteSpace(model.KichThuoc?.ToString()) ||

219 │ string.IsNullOrWhiteSpace(model.VatLieuBaoBi))

220 │ {

221 │ return Json(new ResponseModel("Error", "Các trường (\*) không được để trống."));

222 │ }

223 │

224 │ int result = 0;

225 │ using (var transaction = await dbcontext.Database.BeginTransactionAsync())

226 │ {

227 │ try

228 │ {

229 │ var item = new tbl\_KiemTraChatLuong

230 │ {

231 │ NgayTao = DateTime.Now,

232 │ TenSanPham = model.TenSanPham,

233 │ SoDonHang = model.SoDonHang,

234 │ NgayGiaoHang = Cast.ToDateTime(model.NgayGiaoHang),

235 │ SoLuong = model.SoLuong,

236 │ TenKhachHang = model.TenKhachHang,

237 │ NgaySanXuat = Cast.ToDateTime(model.NgaySanXuat),

238 │ HanSuDung = model.HanSuDung,

239 │ ChatLuongBot = model.ChatLuongBot,

240 │ KetLuan = model.KetLuan,

241 │ GhiChu = model.GhiChu,

242 │ NguoiKiemTra = model.NguoiKiemTra,

243 │ MaLo = model.MaLo,

244 │ TieuChuan = model.TieuChuan,

245 │ KichThuoc = model.KichThuoc.ToInt(),

246 │ VatLieuBaoBi = model.VatLieuBaoBi,

247 │ TrangThai = model.Status == "1",

248 │ NguoiTao = Cast.ToInt(idUser),

249 │ MaCN = Cast.ToInt(idBranch)

250 │ };

251 │

252 │ dbcontext.tbl\_KiemTraChatLuongs.Add(item);

253 │ await dbcontext.SaveChangesAsync();

254 │

255 │ if (!string.IsNullOrEmpty(model.ListCT) && model.ListCT.Length > 2)

256 │ {

257 │ List<ChiTiet> dataList;

258 │ try

259 │ {

260 │ dataList = JsonConvert.DeserializeObject<List<ChiTiet>>(model.ListCT) ?? new List<ChiTiet>()...

261 │ }

262 │ catch (Exception ex)

263 │ {

264 │ return Json(new ResponseModel("Error", $"Lỗi khi xử lý danh sách chi tiết: {ex.Messa...

265 │ }

266 │

267 │ if (dataList.Any())

268 │ {

269 │ var detailEntities = new List<tbl\_ChiTietKiemTra>();

270 │

271 │ foreach (var ct in dataList)

272 │ {

273 │ var chitiet = new tbl\_ChiTietKiemTra

274 │ {

275 │ Code = ct.TestItem,

276 │ DVT = ct.Unit,

277 │ TieuChuan = ct.Specification,

278 │ KetQua = ct.Results,

279 │ STT = Cast.ToInt(ct.No),

280 │ Method = ct.Method,

281 │ NguoiTao = idUser,

282 │ NgayTao = DateTime.Now,

283 │ MaKTCL = item.Ma

284 │ };

285 │

286 │ detailEntities.Add(chitiet);

287 │ }

288 │

289 │ dbcontext.tbl\_ChiTietKiemTras.AddRange(detailEntities);

290 │ await dbcontext.SaveChangesAsync();

291 │ }

292 │ }

293 │

294 │ await transaction.CommitAsync();

295 │ result = 1;

296 │ }

297 │ catch (Exception ex)

298 │ {

299 │ await transaction.RollbackAsync();

300 │ return Json(new ResponseModel("Error", ex.Message));

301 │ }

302 │ }

303 │

304 │ return Json(Utility.MakeResponseFromAddResult(Cast.ToString(result)));

305 │ }

306 │

307 │ [TokenUserCheckHTTP()]

308 │ [HttpPost("updateAPI")]

309 │ public async Task<IActionResult> UpdateAPI([FromBody] KiemTraChatLuong model)

310 │ {

311 │ var idUser = objToken.userid;

312 │ var idBranch = objToken.branch;

313 │

314 │ if (idUser == 0 || idBranch == 0)

315 │ {

316 │ return Json(new ResponseModel("Error", "Hết phiên đăng nhập. Vui lòng đăng nhập lại."));

317 │ }

318 │

319 │ if (model.SoDonHang == null ||

320 │ model.MaLo == null ||

321 │ model.TenKhachHang == null ||

322 │ model.HanSuDung == null ||

323 │ model.ChatLuongBot == null ||

324 │ model.KetLuan == null ||

325 │ model.SoLuong == null)

326 │ {

327 │ return Json(new ResponseModel("Error", "Các trường (\*) không được để trống."));

328 │

329 │ }

330 │

331 │ int result = 0;

332 │

333 │ using (var transaction = await dbcontext.Database.BeginTransactionAsync())

334 │ {

335 │ try

336 │ {

337 │ tbl\_KiemTraChatLuong? item = await dbcontext.tbl\_KiemTraChatLuongs.FindAsync(Cast.ToInt(model.KiemTr...

338 │

339 │ item.TenSanPham = model.TenSanPham;

340 │ item.SoDonHang = model.SoDonHang;

341 │ item.NgayGiaoHang = Cast.ToDateTime(model.NgayGiaoHang);

342 │ item.SoLuong = model.SoLuong;

343 │ item.TenKhachHang = model.TenKhachHang;

344 │ item.NgaySanXuat = Cast.ToDateTime(model.NgaySanXuat);

345 │ item.HanSuDung = model.HanSuDung;

346 │ item.ChatLuongBot = model.ChatLuongBot;

347 │ item.KetLuan = model.KetLuan;

348 │ item.GhiChu = model.GhiChu;

349 │ item.NguoiKiemTra = model.NguoiKiemTra;

350 │ item.MaLo = model.MaLo;

351 │ item.TieuChuan = model.TieuChuan;

352 │ item.KichThuoc = model.KichThuoc.ToInt();

353 │ item.VatLieuBaoBi = model.VatLieuBaoBi;

354 │

355 │

356 │ item.TrangThai = model.Status == "1";

357 │ item.NguoiSua = Cast.ToInt(idUser);

358 │ item.NgaySua = DateTime.Now;

359 │ item.MaCN = Cast.ToInt(idBranch);

360 │

361 │ // Thêm dữ liệu vào bảng

362 │ dbcontext.tbl\_KiemTraChatLuongs.Update(item);

363 │ await dbcontext.SaveChangesAsync();

364 │

365 │ // Nếu có danh sách chi tiết, thêm dữ liệu vào bảng ChiTiet

366 │ if (!string.IsNullOrEmpty(model.ListCT) && model.ListCT.Length > 2)

367 │ {

368 │ try

369 │ {

370 │ List<ChiTiet> dataList = JsonConvert.DeserializeObject<List<ChiTiet>>(model.ListCT) ?? [];

371 │

372 │ List<tbl\_ChiTietKiemTra> chiTietKiemTras = await dbcontext.tbl\_ChiTietKiemTras

373 │ .Where(x => x.MaKTCL == item.Ma && !(x.DaXoa ?? false))

374 │ .ToListAsync();

375 │

376 │ foreach (var ct in dataList)

377 │ {

378 │ tbl\_ChiTietKiemTra? chitiet = await dbcontext.tbl\_ChiTietKiemTras.FindAsync(ct.Ma);

379 │

380 │ if (chitiet == null)

381 │ {

382 │ chitiet = new tbl\_ChiTietKiemTra();

383 │ chitiet.MaKTCL = item.Ma;

384 │ }

385 │

386 │ chitiet.Code = ct.TestItem;

387 │ chitiet.DVT = ct.Unit;

388 │ chitiet.TieuChuan = ct.Specification;

389 │ chitiet.KetQua = ct.Results;

390 │ chitiet.STT = Cast.ToInt(ct.No) != null ? Cast.ToInt(ct.No) : null;

391 │ chitiet.Method = ct.Method;

392 │

393 │ chitiet.NguoiTao = idUser;

394 │ chitiet.NgayTao = DateTime.Now;

395 │ chitiet.MaKTCL = item.Ma;

396 │

397 │ chitiet.NguoiSua = idUser;

398 │ chitiet.NgaySua = DateTime.Now;

399 │

400 │ dbcontext.tbl\_ChiTietKiemTras.Update(chitiet);

401 │ await dbcontext.SaveChangesAsync();

402 │

403 │ // Xác nhận transaction

404 │ int MaChitiet = chitiet.Ma;

405 │ if (chiTietKiemTras.Any(x => x.Ma == chitiet.Ma))

406 │ {

407 │ chiTietKiemTras.Remove(chiTietKiemTras.First(x => x.Ma == MaChitiet));

408 │ }

409 │ }

410 │ if (chiTietKiemTras.Count > 0)

411 │ {

412 │ foreach (var item1 in chiTietKiemTras)

413 │ {

414 │ item1.DaXoa = true;

415 │ item1.NguoiXoa = Cast.ToInt(idUser);

416 │ item1.NgayXoa = DateTime.Now;

417 │

418 │ dbcontext.tbl\_ChiTietKiemTras.Update(item1);

419 │ await dbcontext.SaveChangesAsync();

420 │ }

421 │ }

422 │ }

423 │ catch (Exception e)

424 │ {

425 │ return Json(new ResponseModel("Error", e.Message));

426 │ }

427 │ }

428 │ await transaction.CommitAsync();

429 │ result = 1;

430 │ }

431 │ catch (Exception ex)

432 │ {

433 │ // Nếu có lỗi, rollback transaction

434 │ await transaction.RollbackAsync();

435 │ return Json(new ResponseModel("Error", ex.Message));

436 │ }

437 │ }

438 │

439 │ return Json(Utility.MakeResponseFromAddResult(Cast.ToString(result)));

440 │ }

441 │

442 │ [HttpPost("DeleteAPI")]

443 │ [TokenUserCheckHTTP(Permission.Delete)]

444 │ public IActionResult DeleteAPI([FromBody] DeleteDTO request)

445 │ {

446 │ var item = dbcontext.tbl\_KiemTraChatLuongs.FirstOrDefault(x => x.Ma == Cast.ToInt(request.eMa.Decrypt()));

447 │ if (item == null)

448 │ {

449 │ return Json(QLCLConst.ResponseNotExist);

450 │ }

451 │ item.DaXoa = true;

452 │ dbcontext.SaveChanges();

453 │ return Json(QLCLConst.ResponseDeleteSuccessfully);

454 │ }

455 │

456 │ [TokenUserCheckHTTP(Permission.View)]

457 │ [HttpGet("getListDetail")]

458 │ public IActionResult getListDetail(string maDanhSach)

459 │ {

460 │ if (string.IsNullOrEmpty(maDanhSach))

461 │ {

462 │ return Json(new ResponseModel(QLCLConst.Successfully, "", new List<tbl\_ChiTietKiemTra>()));

463 │ }

464 │ int maDanhSachInt = Cast.ToInt(maDanhSach);

465 │ var listData = dbcontext.tbl\_ChiTietKiemTras

466 │ .Where(x => x.MaKTCL == maDanhSachInt)

467 │ .OrderBy(x => x.STT).ToList();

468 │

469 │ return Json(new ResponseModel(QLCLConst.Successfully, "", listData));

470 │ }

471 │

472 │ [TokenUserCheckHTTP(Permission.View)]

473 │ [HttpGet("GetListKiemTraChatLuongTheoNgay")]

474 │ public IActionResult GetListKiemTraChatLuongTheoNgay(string TuNgay, string DenNgay, string FileType, string MaKT...

475 │ {

476 │ DateTime dtpTuNgay = DateTime.TryParse(TuNgay, out DateTime tempTuNgay) ? tempTuNgay : DateTime.Now;

477 │ DateTime dtpDenNgay = DateTime.TryParse(DenNgay, out DateTime tempDenNgay) ? tempDenNgay : DateTime.Now;

478 │

479 │ int maKTCLInt = Cast.ToInt(MaKTCL);

480 │ int branch = Cast.ToInt(objToken.branch);

481 │

482 │ var hanghoa = crmContext.VHangHoa

483 │ .Where(x => x.HangHoa ?? false)

484 │ .Select(h => new { h.Ten, h.TenTiengAnh })

485 │ .AsNoTracking()

486 │ .ToList();

487 │

488 │ var listData = (from sl in dbcontext.tbl\_KiemTraChatLuongs.AsNoTracking()

489 │ join ctsl in dbcontext.tbl\_ChiTietKiemTras.AsNoTracking() on sl.Ma equals ctsl.MaKTCL

490 │ join cn in dbcontext.tbl\_ChiNhanhs.AsNoTracking() on sl.MaCN equals cn.Ma

491 │ where sl.NgayTao.HasValue &&

492 │ (sl.NgayTao.Value.Date >= dtpTuNgay.Date && sl.NgayTao.Value.Date <= dtpDenNgay.Date) ...

493 │ sl.MaCN == branch &&

494 │ !(ctsl.DaXoa ?? false) &&

495 │ !(sl.DaXoa ?? false) &&

496 │ sl.Ma == maKTCLInt

497 │ select new

498 │ {

499 │ sl.Ma,

500 │ sl.MaCN,

501 │ sl.SoDonHang,

502 │ sl.TenSanPham,

503 │ sl.NgayGiaoHang,

504 │ sl.NgaySanXuat,

505 │ sl.NgayTao,

506 │ sl.SoLuong,

507 │ sl.TenKhachHang,

508 │ sl.HanSuDung,

509 │ sl.ChatLuongBot,

510 │ sl.KetLuan,

511 │ sl.NguoiKiemTra,

512 │ sl.TrangThai,

513 │ sl.MaLo,

514 │ ctsl.Code,

515 │ ctsl.DVT,

516 │ ctsl.TieuChuan,

517 │ ctsl.KetQua,

518 │ ctsl.STT,

519 │ ctsl.Method

520 │ }).ToList();

521 │

522 │ var combineData = (from hh in hanghoa

523 │ join ktcl in listData on hh.Ten equals ktcl.TenSanPham

524 │ select new

525 │ {

526 │ ktcl.Ma,

527 │ ktcl.MaCN,

528 │ ktcl.SoDonHang,

529 │ ktcl.TenSanPham,

530 │ NgayGiaoHang = ktcl.NgayGiaoHang.HasValue ? $"{ktcl.NgayGiaoHang:dd/MM/yyyy}" : "",

531 │ NgaySanXuat = ktcl.NgaySanXuat.HasValue ? $"{ktcl.NgaySanXuat:dd/MM/yyyy}" : "",

532 │ HanSuDung = ktcl.HanSuDung,

533 │ ktcl.SoLuong,

534 │ ktcl.TenKhachHang,

535 │ ktcl.ChatLuongBot,

536 │ ktcl.KetLuan,

537 │ ktcl.NguoiKiemTra,

538 │ ktcl.TrangThai,

539 │ ktcl.Code,

540 │ ktcl.DVT,

541 │ ktcl.TieuChuan,

542 │ ktcl.MaLo,

543 │ // Format KetQua with thousand separators if available

544 │ KetQua = ktcl.KetQua,

545 │ ktcl.STT,

546 │ ktcl.Method,

547 │ hh.TenTiengAnh

548 │ }).OrderBy(x => x.STT).ToList();

549 │

550 │ if (FileType.IsEmpty())

551 │ {

552 │ return Json(new ResponseModel(QLCLConst.OK, "", listData));

553 │ }

554 │

555 │ DataTable dt = new DataTable();

556 │ dt.Columns.Add("NO");

557 │ dt.Columns.Add("TESTING ITEM");

558 │ dt.Columns.Add("UNIT");

559 │ dt.Columns.Add("SPECIFICATION");

560 │ dt.Columns.Add("METHOD");

561 │ dt.Columns.Add("RESULTS");

562 │

563 │ foreach (var item in combineData)

564 │ {

565 │ dt.Rows.Add(new object[]

566 │ {

567 │ item.STT,

568 │ item.Code,

569 │ item.DVT,

570 │ item.TieuChuan,

571 │ item.Method,

572 │ item.KetQua

573 │ });

574 │ }

575 │

576 │ string fileName = FileType == "Excel" ? "KiemTraChatLuongCOA.xlsx" : "KiemTraChatLuongCOA.pdf";

577 │ string contentType = FileType == "Excel" ? "application/ms-excel" : "application/pdf";

578 │

579 │ string fileTemplate = Path.Combine(\_hostingEnvironment.ContentRootPath, "wwwroot", "templates", "KiemTraChat...

580 │ if (!System.IO.File.Exists(fileTemplate))

581 │ {

582 │ return NotFound("File template not found.");

583 │ }

584 │

585 │ ExcelFormat format = new ExcelFormat { NumberFormat = "#,###0.####" };

586 │ ExcelFormat format2 = new ExcelFormat { Border = true };

587 │

588 │ var firstItem = combineData.FirstOrDefault();

589 │ var getTenTiengAnh = !string.IsNullOrWhiteSpace(firstItem?.TenTiengAnh)

590 │ ? firstItem.TenTiengAnh

591 │ : firstItem?.TenSanPham;

592 │

593 │ List<ExcelValue> ListGiaTriRiengLe = new List<ExcelValue>

594 │ {

595 │ new ExcelValue { Range = "E5", Value = getTenTiengAnh },

596 │ new ExcelValue { Range = "E6", Value = firstItem?.MaLo },

597 │ new ExcelValue { Range = "E7", Value = firstItem?.SoLuong },

598 │ new ExcelValue { Range = "E8", Value = firstItem?.NgaySanXuat },

599 │ new ExcelValue { Range = "E9", Value = firstItem?.HanSuDung },

600 │ };

601 │

602 │ int startRow = 12;

603 │ Stream fileStream = ExportToExcel.FromTemplate(fileTemplate, FileType, ListGiaTriRiengLe, dt, true, startRow...

604 │

605 │ return new FileStreamResult(fileStream, contentType)

606 │ {

607 │ FileDownloadName = fileName

608 │ };

609 │ }

610 │

611 │ [HttpPost("getkiemtrachatluongtheongay")]

612 │ [TokenUserCheckHTTP]

613 │ public IActionResult GetKiemTraSoLuongTheoNgay([FromBody] RequestBody model)

614 │ {

615 │ int current = objToken.branch;

616 │ if (!DateTime.TryParse(model.TuNgay, out DateTime dtTuNgay) || !DateTime.TryParse(model.DenNgay, out DateTim...

617 │ {

618 │ return Json(new ResponseModel("Error", "Ngày không hợp lệ."));

619 │ }

620 │

621 │ if (dtTuNgay > dtDenNgay)

622 │ {

623 │ return Json(new ResponseModel("Error", "Từ ngày không lớn hơn Đến Ngày."));

624 │ }

625 │

626 │ var dataKTCL = dbcontext.tbl\_KiemTraChatLuongs

627 │ .Where(x => !(x.DaXoa ?? false)

628 │ && x.NgayTao != null

629 │ && x.NgayTao.Value.Date >= dtTuNgay.Date

630 │ && x.NgayTao.Value.Date <= dtDenNgay.Date

631 │ && x.MaCN == current)

632 │ .OrderByDescending(x => x.NgayTao)

633 │ .ToList()

634 │ .Select(t => new

635 │ {

636 │ Ma = t.Ma,

637 │ eMa = t.Ma.Encrypt(),

638 │ TenSanPham = t.TenSanPham,

639 │ SoDonHang = t.SoDonHang,

640 │ NgayGiaoHang = t.NgayGiaoHang,

641 │ SoLuong = t.SoLuong,

642 │ TenKhachHang = t.TenKhachHang,

643 │ NgaySanXuat = t.NgaySanXuat,

644 │ HanSuDung = t.HanSuDung,

645 │ ChatLuongBot = t.ChatLuongBot,

646 │ KetLuan = t.KetLuan,

647 │ GhiChu = t.GhiChu,

648 │ NguoiKiemTra = t.NguoiKiemTra,

649 │ TrangThai = t.TrangThai,

650 │

651 │ })

652 │ .ToList();

653 │

654 │ var dataCT = new List<tbl\_ChiTietKiemTra>();

655 │

656 │ foreach (var item in dataKTCL)

657 │ {

658 │ var chiTiet = dbcontext.tbl\_ChiTietKiemTras

659 │ .Where(x => x.MaKTCL == item.Ma && !(x.DaXoa ?? false))

660 │ .OrderBy(x => x.NgayTao)

661 │ .ThenBy(x => x.STT)

662 │ .ToList();

663 │

664 │ dataCT.AddRange(chiTiet);

665 │ }

666 │

667 │ return Json(new ResponseModel(QLCLConst.OK, "", new

668 │ {

669 │ ListKTCL = dataKTCL,

670 │ ListCT = dataCT,

671 │ }));

672 │ }

673 │

674 │ [HttpPost("getListChiTietTieuChuan")]

675 │ [TokenUserCheckHTTP]

676 │ public IActionResult GetListChiTietTieuChuan([FromBody] ChiTietTieuChuanDTO request)

677 │ {

678 │ int currentBranch = objToken.branch;

679 │

680 │ var hanghoa = crmContext.VHangHoa

681 │ .Where(x => x.HangHoa ?? false)

682 │ .Select(h => new { h.Ma, h.Ten })

683 │ .AsNoTracking()

684 │ .ToList();

685 │

686 │ var listData = (from tc in dbcontext.bt\_TieuChuans.AsNoTracking()

687 │ join cn in dbcontext.tbl\_ChiNhanhs.AsNoTracking() on tc.MaChiNhanh equals cn.Ma

688 │ join cttc in dbcontext.bt\_ChiTietTieuChuans.AsNoTracking() on tc.Ma equals cttc.MaTieuChuan

689 │ join bb in dbcontext.bt\_BaoBis.AsNoTracking() on tc.MaVatLieuBaoBi equals bb.Ma

690 │ where tc.MaChiNhanh == currentBranch

691 │ && bb.Ten == request.KichThuoc.ToInt()

692 │ && tc.TieuChuan == request.TieuChuan

693 │ && bb.VatLieuBaoBi == request.VatLieuBaoBi

694 │ && !(tc.DaXoa ?? false)

695 │ select new

696 │ {

697 │ cttc.No,

698 │ cttc.TestItem,

699 │ cttc.Unit,

700 │ cttc.Specification,

701 │ cttc.Method,

702 │ tc.MaSanPham

703 │ }).ToList();

704 │

705 │ var combineHangHoaTieuChuan = (from hh in hanghoa

706 │ join tc in listData on hh.Ma equals tc.MaSanPham

707 │ where hh.Ten == request.TenSanPham

708 │ select new

709 │ {

710 │ tc.No,

711 │ tc.TestItem,

712 │ tc.Unit,

713 │ tc.Specification,

714 │ tc.Method,

715 │ }).OrderBy(x => x.No)

716 │ .ToList();

717 │

718 │ return Json(new ResponseModel(QLCLConst.OK, "", combineHangHoaTieuChuan));

719 │ }

720 │ }

721 │ }

────────────────────────────────────────────────────────────

**📄 KiemTraKimLoaiController.cs (CS, 44 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using Microsoft.EntityFrameworkCore;

3 │ using QuanLyChatLuong.Common;

4 │ using QuanLyChatLuong.Models;

5 │ using QuanLyChatLuong.Models.APIModels;

6 │

7 │ namespace QuanLyChatLuong.Controllers

8 │ {

9 │ [Route("opsKiemTraKimLoai")]

10 │ public class KiemTraKimLoaiController : BaseController

11 │ {

12 │ public KiemTraKimLoaiController(QuanLyChatLuongDbContext dbcontext, MemoryCacheManager memoryCache) : base(dbcon...

13 │ {

14 │ m\_MenuID = MenuID.opsKiemTraKimLoai;

15 │ }

16 │

17 │ [TokenUserCheckHTTP]

18 │ [HttpGet]

19 │ public IActionResult Index()

20 │ {

21 │

22 │ return View();

23 │ }

24 │

25 │ [TokenUserCheckHTTP(Permission.View)]

26 │ [HttpGet("getDeviceList")]

27 │ public IActionResult getDeviceList(string TuNgay, string DenNgay)

28 │ {

29 │ var branchCode = objToken!.branch;

30 │ DateTime fromDate = string.IsNullOrWhiteSpace(TuNgay) ? DateTime.Now : DateTime.Parse(TuNgay);

31 │ DateTime toDate = string.IsNullOrWhiteSpace(DenNgay) ? DateTime.Now : DateTime.Parse(DenNgay);

32 │

33 │ string strFromDate = fromDate.ToString("MM/dd/yyyy");

34 │ string strToDate = toDate.ToString("MM/dd/yyyy");

35 │ var listData = dbcontext.Set<sp\_DeviceListResult>()

36 │ .FromSqlRaw("EXEC sp\_SoLieuKiemTraKimLoai\_TuNgay\_DenNgay @MaCN = {0}, @TuNgay = {1}, @DenNgay = {2}",

37 │ branchCode, strFromDate, strToDate)

38 │ .ToList();

39 │

40 │ return Json(new ResponseModel(QLCLConst.OK, "", listData));

41 │ }

42 │

43 │ }

44 │ }

────────────────────────────────────────────────────────────

**📄 LocCheckIn.cs (CS, 35 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class LocCheckIn

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaNhanVien { get; set; }

11 │

12 │ public int? MaKh { get; set; }

13 │

14 │ public decimal? Longitude { get; set; }

15 │

16 │ public decimal? Latitude { get; set; }

17 │

18 │ public string? Address { get; set; }

19 │

20 │ public string? GhiChu { get; set; }

21 │

22 │ public int? UserCreated { get; set; }

23 │

24 │ public DateTime? CreatedDate { get; set; }

25 │

26 │ public int? UserModified { get; set; }

27 │

28 │ public DateTime? ModifiedDate { get; set; }

29 │

30 │ public bool? Deleted { get; set; }

31 │

32 │ public int? UserDeleted { get; set; }

33 │

34 │ public DateTime? DeletedDate { get; set; }

35 │ }

────────────────────────────────────────────────────────────

**📄 LocHuyenThi.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class LocHuyenThi

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public int? MaTinh { get; set; }

13 │

14 │ public bool? Status { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 LocPhuongXa.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class LocPhuongXa

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public int? MaHuyen { get; set; }

13 │

14 │ public bool? Status { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 LocQuocGia.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class LocQuocGia

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public bool? Status { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 LocTinhThanh.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class LocTinhThanh

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public int? MaQuocGia { get; set; }

13 │

14 │ public bool? Status { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 LoginController.cs (CS, 92 lines)**

1 │ ﻿using System.Net;

2 │ using Microsoft.AspNetCore.Mvc;

3 │ using Microsoft.EntityFrameworkCore;

4 │ using NetTools;

5 │ using Newtonsoft.Json;

6 │ using QuanLyChatLuong.Common;

7 │ using QuanLyChatLuong.Models;

8 │ using QuanLyChatLuong.Models.APIModels;

9 │

10 │ namespace QuanLyChatLuong.Controllers

11 │ {

12 │ public class LoginController : BaseController

13 │ {

14 │ public LoginController(QuanLyChatLuongDbContext dbcontext, MemoryCacheManager memoryCache) : base(dbcontext, mem...

15 │ {

16 │ }

17 │ public IActionResult Index()

18 │ {

19 │ ViewData["ListBranch"] = dbcontext.tbl\_ChiNhanhs.Where(x => !(x.DaXoa ?? false) && (x.TrangThai ?? false)).T...

20 │ return View();

21 │ }

22 │

23 │ [HttpPost]

24 │ public IActionResult Authenticate(string username, string password, string branch)

25 │ {

26 │ int branchId = Convert.ToInt32(branch);

27 │

28 │ var user = dbcontext.sys\_Users.FirstOrDefault(x => x.TenDangNhap == username);

29 │

30 │ if (user == null) return Json(new ResponseModel("Error", "Tài khoản không tồn tại"));

31 │

32 │ if (user.DaXoa ?? false || !(user.TrangThai ?? false)) return Json(new ResponseModel("Error", "Tài khoản ...

33 │

34 │ if (user != null && user.TenDangNhap == username && user.Password == Utility.GetSHA512(password))

35 │ {

36 │ List<sys\_UserRole> ur = dbcontext.sys\_UserRoles.Where(x => x.MaUser == user.Ma

37 │ && (x.TrangThai ?? false) && !(x.DaXoa ?? false) && x.MaChiNhanh == branchId).ToList();

38 │ if (ur.Count > 0)

39 │ {

40 │ DateTime now = DateTime.Now;

41 │ DateTime expiresDate;

42 │ if (now.Month == 12)

43 │ {

44 │ expiresDate = new DateTime(now.Year + 1, 1, 1);

45 │ }

46 │ else

47 │ {

48 │ expiresDate = new DateTime(now.Year, now.Month + 1, 1);

49 │ }

50 │ string token = signToken(new UserToken(user.Ma, branchId, username));

51 │ Response.Cookies.Append("AuthToken", token, new() { Expires = expiresDate });

52 │ return Json(new ResponseModel("OK", "Đăng nhập thành công"));

53 │

54 │ }

55 │ else return Json(new ResponseModel("Error", "Bạn không thể đăng nhập vào chi nhánh này"));

56 │ }

57 │ else

58 │ {

59 │ return Json(new ResponseModel("Error", "Tài khoản hoặc mật khẩu không đúng"));

60 │ }

61 │ }

62 │ private string signToken(UserToken objToken)

63 │ {

64 │ string obj = Utility.Base64Encode(JsonConvert.SerializeObject(objToken));

65 │ string sign = Utility.GetSHA256(obj + QLCLConst.passcode + DateTime.Now.ToString("yyyyMM"));

66 │ string token = obj + "." + sign;

67 │ return token;

68 │ }

69 │

70 │ [HttpGet]

71 │ [TokenUserCheckHTTP(Permission.LoginAs)]

72 │ public IActionResult LoginAs(string userid)

73 │ {

74 │

75 │ DateTime now = DateTime.Now;

76 │ DateTime expiresDate;

77 │ if (now.Month == 12)

78 │ {

79 │ expiresDate = new DateTime(now.Year + 1, 1, 1);

80 │ }

81 │ else

82 │ {

83 │ expiresDate = new DateTime(now.Year, now.Month + 1, 1);

84 │ }

85 │ objToken.loginAs = true;

86 │ objToken.idAs = userid.ToInt();

87 │ string token = signToken(objToken);

88 │ Response.Cookies.Append("AuthToken", token, new() { Expires = expiresDate });

89 │ return Redirect("/home");

90 │ }

91 │ }

92 │ }

────────────────────────────────────────────────────────────

**📄 ManHinhLedController.cs (CS, 247 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using Microsoft.EntityFrameworkCore;

3 │ using Newtonsoft.Json;

4 │ using QuanLyChatLuong.Common;

5 │ using QuanLyChatLuong.Models;

6 │ using QuanLyChatLuong.Models.APIModels;

7 │ using QuanLyChatLuong.Models.CRMEntity;

8 │ using QuanLyChatLuong.Models.StoreProcedures;

9 │ using System.Net.Http.Headers;

10 │ using System.Text;

11 │ using System.Web.Http.ModelBinding;

12 │

13 │ namespace QuanLyChatLuong.Controllers

14 │ {

15 │ [Route("ManHinhLed")]

16 │ public class ManHinhLedController : BaseController

17 │ {

18 │ public class RequestBody

19 │ {

20 │ public string? MaChiNhanh { get; set; }

21 │ }

22 │ public class ManHinhLed

23 │ {

24 │ public string? ID { get; set; }

25 │ public string IpLed { get; set; }

26 │ public string ChiNhanh { get; set; }

27 │ public string Status { get; set; }

28 │ }

29 │ private readonly QuanLyChatLuongDbContext \_dbContext;

30 │ public ManHinhLedController(QuanLyChatLuongDbContext dbcontext,MemoryCacheManager memoryCache) : base(dbcontext,...

31 │ {

32 │ \_dbContext = dbcontext;

33 │ m\_MenuID = MenuID.manHinhLed;

34 │ }

35 │ [HttpGet]

36 │ [TokenUserCheckHTTP(Permission.View)]

37 │ public IActionResult Index()

38 │ {

39 │ var ss = dbcontext.tbl\_ManLeds.ToList();

40 │ ViewData["lstChiNhanh"] = dbcontext.tbl\_ChiNhanhs.Where(x => !(x.DaXoa ?? false) && (x.TrangThai ?? false))....

41 │ ViewData["lstManHinhLed"] = dbcontext.tbl\_ManLeds.ToList();

42 │ ViewData["CurrentBranch"] = objToken.branch;

43 │ return View();

44 │ }

45 │ [TokenUserCheckHTTP(Permission.Add)]

46 │ [HttpGet("add")]

47 │ public IActionResult Add()

48 │ {

49 │ ViewData["lstChiNhanh"] = dbcontext.tbl\_ChiNhanhs.Where(x => !(x.DaXoa ?? false) && (x.TrangThai ?? false))....

50 │ ViewData["CurrentBranch"] = objToken.branch;

51 │ return View("Form");

52 │ }

53 │ [TokenUserCheckHTTP(Permission.Edit)]

54 │ [HttpGet("edit")]

55 │ public IActionResult Edit(string id)

56 │ {

57 │ ViewData["lstChiNhanh"] = dbcontext.tbl\_ChiNhanhs.Where(x => !(x.DaXoa ?? false) && (x.TrangThai ?? false))....

58 │ ViewData["item"] = dbcontext.tbl\_ManLeds.FirstOrDefault(x => x.Ma==Cast.ToInt(id.Decrypt()));

59 │ ViewData["CurrentBranch"] = objToken.branch;

60 │ return View("Form");

61 │ }

62 │ [TokenUserCheckHTTP()]

63 │ [HttpPost("updateAPI")]

64 │ public IActionResult UpdateAPI([FromBody] ManHinhLed model)

65 │ {

66 │ var item = dbcontext.tbl\_ManLeds.FirstOrDefault(x => x.Ma == Cast.ToInt(model.ID));

67 │ if (item == null)

68 │ {

69 │ tbl\_ManLed manLed = new tbl\_ManLed();

70 │ manLed.ipLed = model.IpLed;

71 │ manLed.MaCN = Cast.ToInt(model.ChiNhanh);

72 │ manLed.TrangThai = model.Status == "1";

73 │ //dbcontext.tbl\_ManLeds.Add(item);

74 │ //dbcontext.SaveChanges();

75 │ return Json(new ResponseModel(QLCLConst.Successfully, "Cập nhật thành công"));

76 │ }

77 │

78 │ item.ipLed= model.IpLed;

79 │ item.MaCN = Cast.ToInt(model.ChiNhanh);

80 │ item.TrangThai = model.Status == "1";

81 │ try

82 │ {

83 │ //dbcontext.tbl\_ManLeds.Update(item);

84 │ //dbcontext.SaveChanges();

85 │ }

86 │ catch (Exception ex)

87 │ {

88 │ return Json(new ResponseModel(QLCLConst.Error, "Cập nhật thất bại: " + ex.Message));

89 │ }

90 │

91 │ return Json(new ResponseModel(QLCLConst.Successfully, "Cập nhật thành công"));

92 │ }

93 │ [HttpPost]

94 │ [TokenUserCheckHTTP]

95 │ public async Task<IActionResult> getData([FromBody] RequestBody obj)

96 │ {

97 │ int MaChiNhanh = Cast.ToInt(obj.MaChiNhanh) == 0 ? objToken.branch : Cast.ToInt(obj.MaChiNhanh);

98 │ var content = dbcontext.Set<sp\_HienThiLedResult>().FromSqlRaw($"sp\_HienThiLed1 @MaChiNhanh={MaChiNhanh}").To...

99 │ return Json(new ResponseModel(QLCLConst.OK, "", content));

100 │

101 │ }

102 │ Dictionary<string, int> HexToRgb(string hex)

103 │ {

104 │ return new Dictionary<string, int>

105 │ {

106 │ ["R"] = Convert.ToInt32(hex.Substring(1, 2), 16),

107 │ ["G"] = Convert.ToInt32(hex.Substring(3, 2), 16),

108 │ ["B"] = Convert.ToInt32(hex.Substring(5, 2), 16),

109 │ };

110 │ }

111 │

112 │ [HttpPost("getDataManHinhLed")]

113 │ [TokenUserCheckHTTP(Permission.NotCheck)]

114 │ public async Task<IActionResult> getDataManHinhLed([FromBody] RequestBody obj)

115 │ {

116 │ int MaChiNhanhX = Cast.ToInt(obj.MaChiNhanh) == 0?objToken.branch: Cast.ToInt(obj.MaChiNhanh);

117 │ var content = dbcontext.Set<sp\_HienThiLedResult>().FromSqlRaw($"sp\_HienThiLed1 @MaChiNhanh={MaChiNhanhX}").T...

118 │

119 │ //string hostAPILed\_NM5 = "http://172.16.10.54:8000/docs";

120 │ tbl\_ManLed ManHinhLedChiNhanh = new tbl\_ManLedBusiness(dbcontext).GetDataManLed(MaChiNhanhX).FirstOrDefault(...

121 │ string sss = JsonConvert.SerializeObject(content);

122 │ if (ManHinhLedChiNhanh != null)

123 │ {

124 │ HttpRequestMessage httpRequestMessage = null;

125 │ string responseString = string.Empty;

126 │ if (string.IsNullOrWhiteSpace(ManHinhLedChiNhanh.ipLed))

127 │ {

128 │ throw new ArgumentException("URL không được để trống.");

129 │ }

130 │

131 │ string url = ManHinhLedChiNhanh.ipLed;

132 │

133 │ if (!url.StartsWith("http://") && !url.StartsWith("https://"))

134 │ {

135 │ url = "http://" + url;

136 │ }

137 │ string stringContent = "";

138 │ if (MaChiNhanhX == (int) MaChiNhanh.NM5)

139 │ {

140 │ List<tbl\_ColorGioiHanSoLieu> color = dbcontext.tbl\_ColorGioiHanSoLieus.Where(x => x.MaChiNhanh == Ma...

141 │ List<tbl\_GioiHanSoLieu> gioihan = dbcontext.tbl\_GioiHanSoLieus.Where(x => x.MaChiNhanh == (int)MaChi...

142 │ var solieu2 = content.FirstOrDefault();

143 │ var redColor = HexToRgb("#FF0000");

144 │ var blueColor = HexToRgb("#0000FF");

145 │ var whiteColor = HexToRgb("#FFFFFF");

146 │ var DoTrangColor = whiteColor;

147 │ var DoTrangGHMaxColor = gioihan.FirstOrDefault(x => x.Code == "B\_DoTrang" && x.GiaTri <= Cast.ToDeci...

148 │ if (DoTrangGHMaxColor != null)

149 │ {

150 │ var color1 = color.FirstOrDefault(x => x.Code == "B\_DoTrang");

151 │ if (color1 != null && color1.MaxColor != null) DoTrangColor = HexToRgb(color1.MaxColor);

152 │ }

153 │ var DoTrangGHMinColor = gioihan.FirstOrDefault(x => x.Code == "B\_DoTrang" && x.GiaTri >= Cast.ToDeci...

154 │ if (DoTrangGHMinColor != null)

155 │ {

156 │ var color1 = color.FirstOrDefault(x => x.Code == "B\_DoTrang");

157 │ if (color1 != null && color1.MinColor != null) DoTrangColor = HexToRgb(color1.MinColor);

158 │ }

159 │ var DoAmColor = whiteColor;

160 │ var DoAmGHMaxColor = gioihan.FirstOrDefault(x => x.Code == "B\_DoAm" && x.GiaTri <= Cast.ToDecimal(so...

161 │ if (DoAmGHMaxColor != null)

162 │ {

163 │ var color1 = color.FirstOrDefault(x => x.Code == "B\_DoAm");

164 │ if (color1 != null && color1.MaxColor != null) DoAmColor = HexToRgb(color1.MaxColor);

165 │ }

166 │ var DoAmGHMinColor = gioihan.FirstOrDefault(x => x.Code == "B\_DoAm" && x.GiaTri >= Cast.ToDecimal(so...

167 │ if (DoAmGHMinColor != null)

168 │ {

169 │ var color1 = color.FirstOrDefault(x => x.Code == "B\_DoAm");

170 │ if (color1 != null && color1.MinColor != null) DoAmColor = HexToRgb(color1.MinColor);

171 │ }

172 │ var TapChatColor = whiteColor;

173 │ var TapChatGHMaxColor = gioihan.FirstOrDefault(x => x.Code == "B\_TapChat" && x.GiaTri <= Cast.ToDeci...

174 │ if (TapChatGHMaxColor != null)

175 │ {

176 │ var color1 = color.FirstOrDefault(x => x.Code == "B\_TapChat");

177 │ if (color1 != null && color1.MaxColor != null) TapChatColor = HexToRgb(color1.MaxColor);

178 │ }

179 │ var TapChatGHMinColor = gioihan.FirstOrDefault(x => x.Code == "B\_TapChat" && x.GiaTri >= Cast.ToDeci...

180 │ if (TapChatGHMinColor != null)

181 │ {

182 │ var color1 = color.FirstOrDefault(x => x.Code == "B\_TapChat");

183 │ if (color1 != null && color1.MinColor != null) TapChatColor = HexToRgb(color1.MinColor);

184 │ }

185 │ var objSend = new

186 │ {

187 │

188 │ do\_trang = solieu2.B\_DoTrang,

189 │ do\_trang\_color = DoTrangColor,

190 │ do\_am = solieu2.B\_DoAm,

191 │ do\_am\_color = DoAmColor,

192 │ tap\_chat = solieu2.B\_TapChat,

193 │ tap\_chat\_color = TapChatColor

194 │ };

195 │ stringContent = JsonConvert.SerializeObject(objSend);

196 │ }

197 │ else

198 │ {

199 │ stringContent = JsonConvert.SerializeObject(content);

200 │ }

201 │ if (MaChiNhanhX == (int)MaChiNhanh.NM2)

202 │ {

203 │ httpRequestMessage = new HttpRequestMessage

204 │ {

205 │ Method = HttpMethod.Get,

206 │ };

207 │ httpRequestMessage.Headers.Accept.Add(new MediaTypeWithQualityHeaderValue("application/json"));

208 │ var solieu = content.FirstOrDefault();

209 │ url += "?dt=" + solieu.B\_DoTrang + "&da=" + solieu.B\_DoAm + "&tc=" + solieu.B\_TapChat;

210 │ }

211 │ else

212 │ {

213 │ httpRequestMessage = new HttpRequestMessage

214 │ {

215 │ Method = HttpMethod.Post,

216 │ Content = new StringContent(stringContent, Encoding.UTF8, "application/json"),

217 │ };

218 │

219 │ httpRequestMessage.Headers.Accept.Add(new MediaTypeWithQualityHeaderValue("application/json"));

220 │ }

221 │

222 │ try

223 │ {

224 │ Uri uri = new Uri(url);

225 │ using (HttpClient client = new HttpClient() { BaseAddress = uri })

226 │ {

227 │ HttpResponseMessage response = await client.SendAsync(httpRequestMessage);

228 │ responseString = await response.Content.ReadAsStringAsync();

229 │ }

230 │ if (responseString.Contains("Not found"))

231 │ {

232 │ return Json(new ResponseModel(QLCLConst.Error, "Không tìm thấy màn hình led", responseStri...

233 │ }

234 │ return Json(new ResponseModel(QLCLConst.OK, "Gửi dữ liệu thành công", responseString));

235 │ }

236 │ catch (Exception e)

237 │ {

238 │ return Json(new ResponseModel(QLCLConst.Error, "Gửi dữ liệu không thành công"));

239 │ }

240 │ }

241 │ else

242 │ {

243 │ return Json(new ResponseModel(QLCLConst.Error, "Xem lai man hinh led"));

244 │ }

245 │ }

246 │ }

247 │ }

────────────────────────────────────────────────────────────

**📄 MemoryCacheManager.cs (CS, 33 lines)**

1 │ ﻿using Microsoft.Extensions.Caching.Memory;

2 │ using Microsoft.Extensions.Options;

3 │

4 │ namespace QuanLyChatLuong.Common

5 │ {

6 │ public sealed class MemoryCacheManager : IMemoryCache

7 │ {

8 │ readonly IOptions<MemoryCacheOptions> \_optionsAccessor;

9 │

10 │ IMemoryCache \_inner;

11 │

12 │ public MemoryCacheManager(IOptions<MemoryCacheOptions> optionsAccessor)

13 │ {

14 │ \_optionsAccessor = optionsAccessor;

15 │ \_inner = new MemoryCache(\_optionsAccessor);

16 │ }

17 │

18 │ public void Reset()

19 │ {

20 │ var old = \_inner;

21 │ \_inner = new MemoryCache(\_optionsAccessor);

22 │ old.Dispose();

23 │ }

24 │

25 │ public void Dispose() => \_inner.Dispose();

26 │

27 │ public ICacheEntry CreateEntry(object key) => \_inner.CreateEntry(key);

28 │

29 │ public void Remove(object key) => \_inner.Remove(key);

30 │

31 │ public bool TryGetValue(object key, out object value) => \_inner.TryGetValue(key, out value);

32 │ }

33 │ }

────────────────────────────────────────────────────────────

**📄 MenuController.cs (CS, 267 lines)**

1 │ ﻿using Microsoft.AspNetCore.Authentication;

2 │ using Microsoft.AspNetCore.Authentication.Cookies;

3 │ using Microsoft.AspNetCore.Mvc;

4 │ using Microsoft.Extensions.Logging;

5 │ using Microsoft.IdentityModel.Tokens;

6 │ using System.IdentityModel.Tokens.Jwt;

7 │ using System.Security.Claims;

8 │ using System.Text;

9 │ using System.Threading.Tasks;

10 │ using Microsoft.EntityFrameworkCore;

11 │ using QuanLyChatLuong.Models;

12 │ using System.Security.Cryptography;

13 │ using QuanLyChatLuong.Common;

14 │ using QuanLyChatLuong.Models.APIModels;

15 │

16 │ namespace QuanLyChatLuong.Controllers

17 │ {

18 │ [Route("sysMenu")]

19 │ public class MenuController : BaseController

20 │ {

21 │ public MenuController(QuanLyChatLuongDbContext dbcontext, MemoryCacheManager memoryCache) : base(dbcontext, memo...

22 │ {

23 │ m\_MenuID = MenuID.sysMenu;

24 │ }

25 │ [TokenUserCheckHTTP(Permission.View)]

26 │ [HttpGet]

27 │ public IActionResult Index()

28 │ {

29 │

30 │ return View();

31 │ }

32 │ private bool CheckExists(int Ma, string TenSanPham)

33 │ {

34 │ return dbcontext.sys\_Menus.Any(x => x.Ten == TenSanPham && x.Ma != Ma && !(x.DaXoa ?? false));

35 │ }

36 │

37 │

38 │ private ResponseModel CheckRequire(string Ten, string Code, string duongDan, string soThuTu)

39 │ {

40 │ if (string.IsNullOrEmpty(Ten))

41 │ {

42 │ return new ResponseModel(QLCLConst.Error, "Chưa nhập tên");

43 │ }

44 │ if (string.IsNullOrEmpty(Code))

45 │ {

46 │ return new ResponseModel(QLCLConst.Error, "Chưa nhập mã code");

47 │ }

48 │ if (string.IsNullOrEmpty(soThuTu))

49 │ {

50 │ return new ResponseModel(QLCLConst.Error, "Chưa nhập số thứ tự");

51 │ }

52 │ return null;

53 │ }

54 │

55 │ [TokenUserCheckHTTP]

56 │ [HttpGet("getListMenu")]

57 │ public IActionResult getListMenu()

58 │ {

59 │ var listData = dbcontext.sys\_Menus

60 │ .Where(x => !(x.DaXoa ?? false))

61 │ .Select(x => new

62 │ {

63 │ x.Ma,

64 │ eMa = x.Ma.Encrypt(),

65 │ x.Code,

66 │ x.MenuCha,

67 │ x.Ten,

68 │ x.DuongDan,

69 │ x.Icon,

70 │ x.IconWeb,

71 │ x.SoThuTu,

72 │ x.ThamSo,

73 │ x.HienMenu,

74 │ x.MenuCon,

75 │ LuonLuonHien = x.LuonLuonHien ?? false,

76 │ x.TrangThai,

77 │ x.NguoiTao,

78 │ x.NgayTao,

79 │ x.NguoiSua,

80 │ x.NgaySua,

81 │ x.DaXoa,

82 │ x.NguoiXoa,

83 │ x.NgayXoa,

84 │ MenuChaTen = dbcontext.sys\_Menus

85 │ .Where(m => m.Ma == x.MenuCha)

86 │ .Select(m => m.Ten)

87 │ .FirstOrDefault(),

88 │ MenuConTen = dbcontext.sys\_Menus

89 │ .Where(m => m.Ma == x.MenuCon)

90 │ .Select(m => m.Ten)

91 │ .FirstOrDefault()

92 │ })

93 │ .ToList();

94 │

95 │ return Json(new { success = true, listData });

96 │ }

97 │

98 │ [TokenUserCheckHTTP(Permission.Add)]

99 │ [HttpGet("add")]

100 │ public IActionResult Add()

101 │ {

102 │ var listMenuCha = dbcontext.sys\_Menus

103 │ .Where(x => !(x.DaXoa ?? false) && x.MenuCha == null)

104 │ .Select(x => new sys\_Menu

105 │ {

106 │ Ma = x.Ma,

107 │ Ten = x.Ten

108 │ })

109 │ .ToList();

110 │ var listMenuCon = dbcontext.sys\_Menus

111 │ .Where(x => !(x.DaXoa ?? false) && x.MenuCha != null)

112 │ .Select(x => new sys\_Menu

113 │ {

114 │ Ma = x.Ma,

115 │ Ten = x.Ten

116 │ })

117 │ .ToList();

118 │

119 │ ViewData["listMenuCha"] = listMenuCha;

120 │ ViewData["listMenuCon"] = listMenuCon;

121 │

122 │ return View("Form");

123 │ }

124 │

125 │ [TokenUserCheckHTTP()]

126 │ [HttpPost("addAPI")]

127 │ public IActionResult AddAPI(int Ma, string Ten, string Code, string duongDan, string Icon, string soThuTu, strin...

128 │ {

129 │ var result = CheckRequire(Ten, Code, duongDan, soThuTu);

130 │ if (result != null)

131 │ {

132 │ return Json(result);

133 │ }

134 │

135 │ if (CheckExists(Ma, Ten))

136 │ {

137 │ return Json(new ResponseModel(QLCLConst.Error, "Tên menu đã tồn tại"));

138 │ }

139 │

140 │ var item = new sys\_Menu

141 │ {

142 │ Ten = Ten,

143 │ Code = Code,

144 │ DuongDan = duongDan,

145 │ IconWeb = Icon,

146 │ SoThuTu = Cast.ToInt(soThuTu),

147 │ MenuCha = Cast.ToInt(menuCha),

148 │ MenuCon = Cast.ToInt(menuCon),

149 │ TrangThai = TrangThai == "1",

150 │ NguoiTao = objToken!.userid,

151 │ NgayTao = DateTime.Now

152 │ };

153 │

154 │

155 │ dbcontext.sys\_Menus.Add(item);

156 │ dbcontext.SaveChanges();

157 │

158 │ return Json(new ResponseModel(QLCLConst.Successfully, "Thêm thành công"));

159 │ }

160 │

161 │ [TokenUserCheckHTTP(Permission.Edit)]

162 │ [HttpGet("edit")]

163 │ public IActionResult Edit(string id)

164 │ {

165 │ var item = dbcontext.sys\_Menus.FirstOrDefault(c => c.Ma == Cast.ToInt(id.Decrypt()));

166 │

167 │ if (item == null)

168 │ {

169 │ return NotFound("Menu not found");

170 │ }

171 │

172 │ var listMenuCha = dbcontext.sys\_Menus

173 │ .Where(x => !(x.DaXoa ?? false) && (x.MenuCha == 0 || x.MenuCha == null) && (x.HienMenu ??false))

174 │ .Select(x => new sys\_Menu

175 │ {

176 │ Ma = x.Ma,

177 │ Ten = x.Ten

178 │ })

179 │ .ToList();

180 │

181 │ var listMenuCon = dbcontext.sys\_Menus

182 │ .Where(x => !(x.DaXoa ?? false) && x.MenuCha != null)

183 │ .Select(x => new sys\_Menu

184 │ {

185 │ Ma = x.Ma,

186 │ Ten = x.Ten

187 │ })

188 │ .ToList();

189 │

190 │ ViewData["item"] = item;

191 │ ViewData["listMenuCha"] = listMenuCha;

192 │ ViewData["listMenuCon"] = listMenuCon;

193 │

194 │ return View("Form");

195 │ }

196 │ [TokenUserCheckHTTP()]

197 │ [HttpPost("updateAPI")]

198 │ public IActionResult UpdateAPI(int Ma, string Ten, string Code, string duongDan, string Icon, string soThuTu, st...

199 │ {

200 │ var item = dbcontext.sys\_Menus.FirstOrDefault(x => x.Ma == Ma && !(x.DaXoa ?? false));

201 │ if (item == null)

202 │ {

203 │ return Json(new ResponseModel(QLCLConst.Error, "Menu không tồn tại"));

204 │ }

205 │ if (dbcontext.sys\_Menus.Any(x => x.Ten == Ten && x.Ma != Ma))

206 │ {

207 │ return Json(new ResponseModel(QLCLConst.Error, "Tên sản phẩm đã tồn tại"));

208 │ }

209 │

210 │ item.Ten = Ten;

211 │ item.Code = Code;

212 │ item.DuongDan = duongDan;

213 │ item.IconWeb = Icon;

214 │ item.SoThuTu = Cast.ToInt(soThuTu);

215 │ item.MenuCha = Cast.ToInt(menuCha);

216 │ item.MenuCon = Cast.ToInt(menuCon);

217 │ item.TrangThai = TrangThai == "1";

218 │ item.NguoiSua = objToken!.userid;

219 │ item.NgaySua = DateTime.Now;

220 │ try

221 │ {

222 │ dbcontext.sys\_Menus.Update(item);

223 │ dbcontext.SaveChanges();

224 │ }

225 │ catch (Exception ex)

226 │ {

227 │ return Json(new ResponseModel(QLCLConst.Error, "Cập nhật thất bại: " + ex.Message));

228 │ }

229 │

230 │ return Json(new ResponseModel(QLCLConst.Successfully, "Cập nhật thành công"));

231 │ }

232 │

233 │ [HttpPost("DeleteAPI")]

234 │ [TokenUserCheckHTTP(Permission.Delete)]

235 │ public IActionResult DeleteAPI(string eMa)

236 │ {

237 │ var item = dbcontext.sys\_Menus.FirstOrDefault(x => x.Ma == Cast.ToInt(eMa.Decrypt()));

238 │ if (item == null)

239 │ {

240 │ return Json(QLCLConst.ResponseNotExist);

241 │ }

242 │ item.DaXoa = true;

243 │ dbcontext.SaveChanges();

244 │ return Json(QLCLConst.ResponseDeleteSuccessfully);

245 │ }

246 │ [TokenUserCheckHTTP()]

247 │ [HttpPost("getListCheckBox")]

248 │ public IActionResult getListCheckBox(string isChecked, string ma)

249 │ {

250 │ var item = dbcontext.sys\_Menus.FirstOrDefault(x => x.Ma == Cast.ToInt(ma));

251 │

252 │ if (item == null)

253 │ {

254 │ return Json(new ResponseModel("Error", "Menu không tồn tại"));

255 │ }

256 │

257 │ item.HienMenu = Cast.ToBoolean(isChecked);

258 │ item.NguoiSua = objToken!.userid;

259 │ item.NgaySua = DateTime.Now;

260 │

261 │ dbcontext.SaveChanges();

262 │ return Json(new ResponseModel(QLCLConst.Successfully, "Chỉnh sửa thành công"));

263 │ }

264 │

265 │

266 │ }

267 │ }

────────────────────────────────────────────────────────────

**📄 MesThongBaoApp.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class MesThongBaoApp

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaUser { get; set; }

11 │

12 │ public string? TieuDe { get; set; }

13 │

14 │ public string? NoiDung { get; set; }

15 │

16 │ public bool? DaXem { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 MultiSQL.cs (CS, 12 lines)**

1 │ ﻿

2 │

3 │ namespace QuanLyChatLuong.Models

4 │ {

5 │ public partial class QuanLyChatLuongDbContext

6 │ {

7 │ public int? userId { get; set; }

8 │

9 │ }

10 │ }

11 │ //Multi SQL ID Generator

12 │ //ConcurrentStack<long> ids = new ConcurrentStack<long>();

────────────────────────────────────────────────────────────

**📄 MultiServer.cs (CS, 24 lines)**

1 │ ﻿using Microsoft.EntityFrameworkCore;

2 │ using System.Net.Sockets;

3 │ using System.Net;

4 │ using System.Text;

5 │ using Newtonsoft.Json;

6 │ using Microsoft.EntityFrameworkCore.Metadata.Internal;

7 │ using Microsoft.AspNetCore.Mvc;

8 │

9 │ using System.Data;

10 │ using System.Reflection;

11 │ using Microsoft.EntityFrameworkCore.Infrastructure;

12 │ using System.ComponentModel;

13 │ using System.Diagnostics.CodeAnalysis;

14 │ using System;

15 │ using System.Diagnostics;

16 │ using static Microsoft.EntityFrameworkCore.DbLoggerCategory;

17 │

18 │ namespace QuanLyChatLuong.Common.MultiServer

19 │ {

20 │ public class MultiServer : Controller

21 │ {

22 │

23 │ }

24 │ }

────────────────────────────────────────────────────────────

**📄 Nhập.cs (CS, 13 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class Nhập

7 │ {

8 │ public string? SốPhiếu { get; set; }

9 │

10 │ public string? NhàCungCấpMới { get; set; }

11 │

12 │ public string? Mancc { get; set; }

13 │ }

────────────────────────────────────────────────────────────

**📄 NotEqual.cs (CS, 24 lines)**

1 │ ﻿namespace QuanLyChatLuong.Common

2 │ {

3 │ public class NotEqual : IRouteConstraint

4 │ {

5 │ private string \_match = String.Empty;

6 │

7 │ public NotEqual(string match)

8 │ {

9 │ \_match = match;

10 │ }

11 │

12 │ public bool Match(HttpContext? httpContext, IRouter? route, string parameterName, RouteValueDictionary values, R...

13 │ {

14 │ try

15 │ {

16 │ return String.Compare(values[parameterName]!.ToString(), \_match, true) != 0;

17 │ }

18 │ catch

19 │ {

20 │ return false;

21 │ }

22 │ }

23 │ }

24 │ }

────────────────────────────────────────────────────────────

**📄 NtfGroupThongBao.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class NtfGroupThongBao

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public bool? Status { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 NtfMemberGroupThongBao.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class NtfMemberGroupThongBao

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaGroupThongBao { get; set; }

11 │

12 │ public int? MaUser { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 OpsBaoGia.cs (CS, 75 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class OpsBaoGia

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public string? SoBaoGia { get; set; }

13 │

14 │ public int? MaKh { get; set; }

15 │

16 │ public int? MaNhaVanChuyen { get; set; }

17 │

18 │ public int? NgayCongNo { get; set; }

19 │

20 │ public DateTime? NgayBaoGia { get; set; }

21 │

22 │ public DateTime? TuNgay { get; set; }

23 │

24 │ public DateTime? DenNgay { get; set; }

25 │

26 │ public string? ThoiGianGiaoHang { get; set; }

27 │

28 │ public string? DiaDiemGiaoHang { get; set; }

29 │

30 │ public string? ThoiGianThanhToan { get; set; }

31 │

32 │ public string? GhiChu { get; set; }

33 │

34 │ public int? NguoiLienHe { get; set; }

35 │

36 │ public bool? ChotHopDong { get; set; }

37 │

38 │ public int? MaTrangThai { get; set; }

39 │

40 │ public int? MaTienTe { get; set; }

41 │

42 │ public int? MaTyGia { get; set; }

43 │

44 │ public int? TyGia { get; set; }

45 │

46 │ public int? MaNvphuTrach { get; set; }

47 │

48 │ public int? MaNvhoTro { get; set; }

49 │

50 │ public bool? RemindExpireDate { get; set; }

51 │

52 │ public bool? RemindSendReport { get; set; }

53 │

54 │ public int? TieuChuanChatLuong { get; set; }

55 │

56 │ public string? ChiPhiBocXep { get; set; }

57 │

58 │ public int? SoTanMoiContainer { get; set; }

59 │

60 │ public int? LoaiContainer { get; set; }

61 │

62 │ public int? UserCreated { get; set; }

63 │

64 │ public DateTime? CreatedDate { get; set; }

65 │

66 │ public int? UserModified { get; set; }

67 │

68 │ public DateTime? ModifiedDate { get; set; }

69 │

70 │ public bool? Deleted { get; set; }

71 │

72 │ public int? UserDeleted { get; set; }

73 │

74 │ public DateTime? DeletedDate { get; set; }

75 │ }

────────────────────────────────────────────────────────────

**📄 OpsChiTietBaoGia.cs (CS, 67 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class OpsChiTietBaoGia

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaBaoGia { get; set; }

11 │

12 │ public int? MaHh { get; set; }

13 │

14 │ public int? MaDvt { get; set; }

15 │

16 │ public int? LoaiBao { get; set; }

17 │

18 │ public int? SoLuong { get; set; }

19 │

20 │ public int? SoLuongDaXuat { get; set; }

21 │

22 │ public int? MaGiaNiemYet { get; set; }

23 │

24 │ public decimal? DonGiaTaiKho { get; set; }

25 │

26 │ public decimal? BaoBiHd { get; set; }

27 │

28 │ public decimal? Bao10Kg { get; set; }

29 │

30 │ public decimal? GiaVanChuyen { get; set; }

31 │

32 │ public decimal? BocXep { get; set; }

33 │

34 │ public int? NgayCongNo { get; set; }

35 │

36 │ public decimal? ChiPhiNgayCongNo { get; set; }

37 │

38 │ public decimal? HoaHong { get; set; }

39 │

40 │ public decimal? ChiPhiHoTro { get; set; }

41 │

42 │ public decimal? DonGiaSauTinh { get; set; }

43 │

44 │ public decimal? DonGiaBan { get; set; }

45 │

46 │ public decimal? ThanhTien { get; set; }

47 │

48 │ public int? MaLyDo { get; set; }

49 │

50 │ public string? GhiChu { get; set; }

51 │

52 │ public bool? Status { get; set; }

53 │

54 │ public int? UserCreated { get; set; }

55 │

56 │ public DateTime? CreatedDate { get; set; }

57 │

58 │ public int? UserModified { get; set; }

59 │

60 │ public DateTime? ModifiedDate { get; set; }

61 │

62 │ public bool? Deleted { get; set; }

63 │

64 │ public int? UserDeleted { get; set; }

65 │

66 │ public DateTime? DeletedDate { get; set; }

67 │ }

────────────────────────────────────────────────────────────

**📄 OpsChiTietDoanhThu.cs (CS, 75 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class OpsChiTietDoanhThu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaDoanhThu { get; set; }

11 │

12 │ public int? MaHang { get; set; }

13 │

14 │ public int? MaDvt { get; set; }

15 │

16 │ public string? KhoHang { get; set; }

17 │

18 │ public string? Lo { get; set; }

19 │

20 │ public int? SoLuong { get; set; }

21 │

22 │ public int? SoLuongThucXuat { get; set; }

23 │

24 │ public decimal? GiaBan { get; set; }

25 │

26 │ public decimal? ThanhTien { get; set; }

27 │

28 │ public decimal? ChietKhau { get; set; }

29 │

30 │ public int? MaThue { get; set; }

31 │

32 │ public decimal? ThueVat { get; set; }

33 │

34 │ public decimal? TienThue { get; set; }

35 │

36 │ public decimal? PhaiThu { get; set; }

37 │

38 │ public int? LoaiBao { get; set; }

39 │

40 │ public decimal? DonGiaTaiKho { get; set; }

41 │

42 │ public decimal? BaoBiHd { get; set; }

43 │

44 │ public decimal? Bao10Kg { get; set; }

45 │

46 │ public decimal? GiaVanChuyen { get; set; }

47 │

48 │ public decimal? GiaVonVanChuyen { get; set; }

49 │

50 │ public decimal? BocXep { get; set; }

51 │

52 │ public int? NgayCongNo { get; set; }

53 │

54 │ public decimal? ChiPhiNgayCongNo { get; set; }

55 │

56 │ public decimal? HoaHong { get; set; }

57 │

58 │ public decimal? ChiPhiHoTro { get; set; }

59 │

60 │ public decimal? DonGiaSauTinh { get; set; }

61 │

62 │ public decimal? GiaNiemYet { get; set; }

63 │

64 │ public int? MaLyDo { get; set; }

65 │

66 │ public decimal? GiaTaiKhoCoBaoBi { get; set; }

67 │

68 │ public int? MaHopDongCt { get; set; }

69 │

70 │ public int? SoLuongHd { get; set; }

71 │

72 │ public int? SoLuongConLai { get; set; }

73 │

74 │ public decimal? GiaGiam { get; set; }

75 │ }

────────────────────────────────────────────────────────────

**📄 OpsDoanhThu.cs (CS, 93 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class OpsDoanhThu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaPhongBan { get; set; }

13 │

14 │ public DateTime? NgayChungTu { get; set; }

15 │

16 │ public int? MaHopDong { get; set; }

17 │

18 │ public string? SoHopDong { get; set; }

19 │

20 │ public string? SoChungTu { get; set; }

21 │

22 │ public int? MaKh { get; set; }

23 │

24 │ public int? MaNvbh { get; set; }

25 │

26 │ public int? MaNhaVanChuyen { get; set; }

27 │

28 │ public string? NhaVanChuyen { get; set; }

29 │

30 │ public int? MaChiNhanhXuat { get; set; }

31 │

32 │ public int? MaKho { get; set; }

33 │

34 │ public string? DienGiai { get; set; }

35 │

36 │ public DateTime? NgayHetHanThanhToan { get; set; }

37 │

38 │ public int? MaTienTe { get; set; }

39 │

40 │ public int? MaTyGia { get; set; }

41 │

42 │ public int? TyGia { get; set; }

43 │

44 │ public string? SoXe { get; set; }

45 │

46 │ public string? SoPo { get; set; }

47 │

48 │ public bool? DaXuat { get; set; }

49 │

50 │ public int? TrangThaiDonHang { get; set; }

51 │

52 │ public bool? XuatKhau { get; set; }

53 │

54 │ public bool? DonHangChoThanhToan { get; set; }

55 │

56 │ public decimal? DaThanhToan { get; set; }

57 │

58 │ public decimal? ConLai { get; set; }

59 │

60 │ public int? UserCreated { get; set; }

61 │

62 │ public DateTime? CreatedDate { get; set; }

63 │

64 │ public int? UserModified { get; set; }

65 │

66 │ public DateTime? ModifiedDate { get; set; }

67 │

68 │ public bool? Deleted { get; set; }

69 │

70 │ public int? UserDeleted { get; set; }

71 │

72 │ public DateTime? DeletedDate { get; set; }

73 │

74 │ public bool? Exported { get; set; }

75 │

76 │ public int? UserExportd { get; set; }

77 │

78 │ public DateTime? ExportedDate { get; set; }

79 │

80 │ public int? MaDonHangTraLai { get; set; }

81 │

82 │ public DateTime? NgayKetThuc { get; set; }

83 │

84 │ public int? MaCnxuatHoaDon { get; set; }

85 │

86 │ public int? MaLyDoDuyet { get; set; }

87 │

88 │ public string? LyDoDuyet { get; set; }

89 │

90 │ public int? MaNguoiLienHe { get; set; }

91 │

92 │ public DateTime? NgayDuKienGiaoHang { get; set; }

93 │ }

────────────────────────────────────────────────────────────

**📄 OpsLichSuChiTietBaoGia.cs (CS, 67 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class OpsLichSuChiTietBaoGia

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaChiTiet { get; set; }

11 │

12 │ public int? MaBaoGia { get; set; }

13 │

14 │ public int? MaHh { get; set; }

15 │

16 │ public int? LoaiBao { get; set; }

17 │

18 │ public int? SoLuong { get; set; }

19 │

20 │ public int? SoLuongDaXuat { get; set; }

21 │

22 │ public int? MaGiaNiemYet { get; set; }

23 │

24 │ public int? DonGiaTaiKho { get; set; }

25 │

26 │ public int? BaoBiHd { get; set; }

27 │

28 │ public int? Bao10Kg { get; set; }

29 │

30 │ public int? GiaVanChuyen { get; set; }

31 │

32 │ public int? BocXep { get; set; }

33 │

34 │ public int? NgayCongNo { get; set; }

35 │

36 │ public int? ChiPhiNgayCongNo { get; set; }

37 │

38 │ public int? HoaHong { get; set; }

39 │

40 │ public int? ChiPhiHoTro { get; set; }

41 │

42 │ public int? DonGiaSauTinh { get; set; }

43 │

44 │ public int? DonGiaBan { get; set; }

45 │

46 │ public decimal? ThanhTien { get; set; }

47 │

48 │ public int? MaLyDo { get; set; }

49 │

50 │ public string? GhiChu { get; set; }

51 │

52 │ public bool? Status { get; set; }

53 │

54 │ public int? UserCreated { get; set; }

55 │

56 │ public DateTime? CreatedDate { get; set; }

57 │

58 │ public int? UserModified { get; set; }

59 │

60 │ public DateTime? ModifiedDate { get; set; }

61 │

62 │ public bool? Deleted { get; set; }

63 │

64 │ public int? UserDeleted { get; set; }

65 │

66 │ public DateTime? DeletedDate { get; set; }

67 │ }

────────────────────────────────────────────────────────────

**📄 OpsThanhToan.cs (CS, 41 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class OpsThanhToan

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaDoanhThu { get; set; }

13 │

14 │ public string? SoChungTu { get; set; }

15 │

16 │ public int? MaKh { get; set; }

17 │

18 │ public string? MaKh1 { get; set; }

19 │

20 │ public DateTime? NgayThanhToan { get; set; }

21 │

22 │ public decimal? SoTien { get; set; }

23 │

24 │ public string? GhiChu { get; set; }

25 │

26 │ public bool? Status { get; set; }

27 │

28 │ public int? UserCreated { get; set; }

29 │

30 │ public DateTime? CreatedDate { get; set; }

31 │

32 │ public int? UserModified { get; set; }

33 │

34 │ public DateTime? ModifiedDate { get; set; }

35 │

36 │ public bool? Deleted { get; set; }

37 │

38 │ public int? UserDeleted { get; set; }

39 │

40 │ public DateTime? DeletedDate { get; set; }

41 │ }

────────────────────────────────────────────────────────────

**📄 OpsTieuChuanKyThuat.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class OpsTieuChuanKyThuat

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaBaoGia { get; set; }

11 │

12 │ public int? MaTieuChuan { get; set; }

13 │

14 │ public string? TieuChuan { get; set; }

15 │

16 │ public bool? Status { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 PageNotFoundController.cs (CS, 27 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using QuanLyChatLuong.Common;

3 │

4 │ namespace QuanLyChatLuong.Controllers

5 │ {

6 │ public class PageNotFoundController : Controller

7 │ {

8 │ public ActionResult Index()

9 │ {

10 │ return View();

11 │ }

12 │

13 │ public ActionResult BaoTri()

14 │ {

15 │ return View();

16 │ }

17 │ public ActionResult LoginAsDenied()

18 │ {

19 │ return View();

20 │ }

21 │

22 │ public ActionResult PermissionDenied()

23 │ {

24 │ return View();

25 │ }

26 │ }

27 │ }

────────────────────────────────────────────────────────────

**📄 PermissionController.cs (CS, 105 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using Microsoft.AspNetCore.Mvc.ModelBinding;

3 │ using QuanLyChatLuong.Common;

4 │ using QuanLyChatLuong.Models;

5 │ using QuanLyChatLuong.Models.APIModels;

6 │ using System.Net.WebSockets;

7 │ using System.Runtime.CompilerServices;

8 │

9 │ namespace QuanLyChatLuong.Controllers

10 │ {

11 │ [Route("sysQuyen")]

12 │ public class PermissionController : BaseController

13 │ {

14 │ public PermissionController(QuanLyChatLuongDbContext dbcontext, MemoryCacheManager memoryCache) : base(dbcontext...

15 │ {

16 │ m\_MenuID = MenuID.sysQuyen;

17 │ }

18 │ public class PermissionForm

19 │ {

20 │ public string? ID { get; set; }

21 │ public string? Code { get; set; }

22 │ public string? Name { get; set; }

23 │ public string? Description { get; set; }

24 │ public string? FormAccess { get; set; }

25 │ public string? Status { get; set; }

26 │ }

27 │ [TokenUserCheckHTTP]

28 │ public IActionResult Index()

29 │ {

30 │ ViewData["ListData"] = new sysQuyenModel(dbcontext).GetData();

31 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysQuyen, (int)Permissi...

32 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysQuyen, (int)Permiss...

33 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysQuyen, (int)Permi...

34 │ return View();

35 │ }

36 │ [HttpGet("add")]

37 │ [TokenUserCheckHTTP]

38 │ public IActionResult Add()

39 │ {

40 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysQuyen, (int)Permissi...

41 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysQuyen, (int)Permiss...

42 │ return View("Edit");

43 │ }

44 │ [HttpGet("edit")]

45 │ [TokenUserCheckHTTP]

46 │ public IActionResult Edit(string id)

47 │ {

48 │ var item = new sysQuyenModel(dbcontext).GetItem(Cast.ToInt(id.Decrypt()));

49 │ if (item == null) return NotFound("Permission not found");

50 │ ViewData["item"] = item;

51 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysQuyen, (int)Permissi...

52 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysQuyen, (int)Permiss...

53 │ return View("Edit");

54 │ }

55 │ [HttpPost("updateapi")]

56 │ [TokenUserCheckHTTP]

57 │ public IActionResult UpdateAPI([FromBody] PermissionForm model)

58 │ {

59 │ int result = new sysQuyenModel(dbcontext).Edit(Cast.ToInt(model.ID), model.Code, model.Name, model.Descripti...

60 │ return Json(Utility.MakeResponseFromEditResult(Cast.ToString(result)));

61 │ }

62 │

63 │ [HttpPost("addapi")]

64 │ [TokenUserCheckHTTP]

65 │ public IActionResult AddAPI([FromBody] PermissionForm model)

66 │ {

67 │ int result = new sysQuyenModel(dbcontext).Add(model.Code, model.Name, model.Description, model.FormAccess ==...

68 │ return Json(Utility.MakeResponseFromAddResult(Cast.ToString(result)));

69 │ }

70 │

71 │ [TokenUserCheckHTTP]

72 │ [HttpPost("deleteapi")]

73 │ public IActionResult DeleteAPI([FromBody] PermissionForm model)

74 │ {

75 │ string result = Cast.ToString(new sysQuyenModel(dbcontext).Delete(Cast.ToInt(model.ID.Decrypt), objToken!.us...

76 │ return Json(Utility.MakeResponseFromDeleteResult(Cast.ToString(result)));

77 │ }

78 │

79 │ [HttpPost("getquyen")]

80 │ [TokenUserCheckHTTP]

81 │ public ResponseModel GetQuyen()

82 │ {

83 │ List<dynamic> data = new List<dynamic>();

84 │ List<sys\_Menu> menu = dbcontext.sys\_Menus.Where(x => !(x.DaXoa ?? false)).ToList();

85 │ List<vQuyen> permissions = dbcontext.vQuyens.Where(t => t.MaUser == objToken!.userid).ToList();

86 │

87 │ foreach (vQuyen p in permissions.GroupBy(t => t.MaMenu).Select(t => t.FirstOrDefault()).ToList())

88 │ {

89 │ List<int> permissionid = new List<int>();

90 │ List<vQuyen> pp = permissions.Where(t => t.MaUser == p?.MaMenu).ToList();

91 │ foreach (vQuyen p2 in permissions.Where(t => t.MaMenu == p?.MaMenu))

92 │ {

93 │ permissionid.Add(p2?.MaQuyen ?? Utility.defaultUID);

94 │ }

95 │ data.Add(new

96 │ {

97 │ MenuID = p?.MaMenu,

98 │ MenuName = (menu.FirstOrDefault(t => t.Ma == p?.MaMenu) ?? new sys\_Menu() { Ten = "" }).Ten,

99 │ PermissionList = permissionid

100 │ });

101 │ }

102 │ return new ResponseModel(QLCLConst.OK, "", data);

103 │ }

104 │ }

105 │ }

────────────────────────────────────────────────────────────

**📄 PlBaoCaoSanLuong.cs (CS, 35 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PlBaoCaoSanLuong

7 │ {

8 │ public long? Ma { get; set; }

9 │

10 │ public int? Thang { get; set; }

11 │

12 │ public int? Nam { get; set; }

13 │

14 │ public int? MaKh { get; set; }

15 │

16 │ public int? MaNhanVienBanHang { get; set; }

17 │

18 │ public int? MaTienTe { get; set; }

19 │

20 │ public int? TyGia { get; set; }

21 │

22 │ public int? MaChiNhanhXuat { get; set; }

23 │

24 │ public int? MaHangHoa { get; set; }

25 │

26 │ public int? MaLoaiBao { get; set; }

27 │

28 │ public int? MaDvt { get; set; }

29 │

30 │ public int? SoLuong { get; set; }

31 │

32 │ public decimal? DonGia { get; set; }

33 │

34 │ public decimal? ThanhTien { get; set; }

35 │ }

────────────────────────────────────────────────────────────

**📄 PlChiTietChiTieuBan.cs (CS, 37 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PlChiTietChiTieuBan

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiTieu { get; set; }

11 │

12 │ public int? MaPhongBan { get; set; }

13 │

14 │ public bool? XuatKhau { get; set; }

15 │

16 │ public int? MaTienTe { get; set; }

17 │

18 │ public int? ChiTieuBan { get; set; }

19 │

20 │ public int? MaDvt { get; set; }

21 │

22 │ public string? GhiChu { get; set; }

23 │

24 │ public int? UserCreated { get; set; }

25 │

26 │ public DateTime? CreatedDate { get; set; }

27 │

28 │ public int? UserModified { get; set; }

29 │

30 │ public DateTime? ModifiedDate { get; set; }

31 │

32 │ public bool? Deleted { get; set; }

33 │

34 │ public int? UserDeleted { get; set; }

35 │

36 │ public DateTime? DeletedDate { get; set; }

37 │ }

────────────────────────────────────────────────────────────

**📄 PlChiTieuBanHang.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PlChiTieuBanHang

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? Nam { get; set; }

11 │

12 │ public int? XuatKhau { get; set; }

13 │

14 │ public int? NoiDia { get; set; }

15 │

16 │ public string? GhiChu { get; set; }

17 │

18 │ public int? TrangThai { get; set; }

19 │

20 │ public int? UserCreated { get; set; }

21 │

22 │ public DateTime? CreatedDate { get; set; }

23 │

24 │ public int? UserModified { get; set; }

25 │

26 │ public DateTime? ModifiedDate { get; set; }

27 │

28 │ public bool? Deleted { get; set; }

29 │

30 │ public int? UserDeleted { get; set; }

31 │

32 │ public DateTime? DeletedDate { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 PlChiTieuBanHangTheoThang.cs (CS, 35 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PlChiTieuBanHangTheoThang

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? Nam { get; set; }

11 │

12 │ public int? Thang { get; set; }

13 │

14 │ public int? MaPhongBan { get; set; }

15 │

16 │ public int? SoLuongBan { get; set; }

17 │

18 │ public string? GhiChu { get; set; }

19 │

20 │ public int? UserCreated { get; set; }

21 │

22 │ public DateTime? CreatedDate { get; set; }

23 │

24 │ public int? UserModified { get; set; }

25 │

26 │ public DateTime? ModifiedDate { get; set; }

27 │

28 │ public bool? Deleted { get; set; }

29 │

30 │ public int? UserDeleted { get; set; }

31 │

32 │ public DateTime? DeletedDate { get; set; }

33 │

34 │ public int? MaDvt { get; set; }

35 │ }

────────────────────────────────────────────────────────────

**📄 PlChiTieuSx.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PlChiTieuSx

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? Nam { get; set; }

13 │

14 │ public int? ChiTieuSx { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 PlChiTieuThuMuaCuMi.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PlChiTieuThuMuaCuMi

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public bool? TheoThang { get; set; }

13 │

14 │ public int? Nam { get; set; }

15 │

16 │ public decimal? SoLuongMua { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 PromChietKhauThuongMai.cs (CS, 61 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PromChietKhauThuongMai

7 │ {

8 │ public Guid Ma { get; set; }

9 │

10 │ public int? MaLoaiChungTu { get; set; }

11 │

12 │ public int? MaChiNhanh { get; set; }

13 │

14 │ public int? MaPhongBan { get; set; }

15 │

16 │ public string? SoPhieu { get; set; }

17 │

18 │ public DateTime? NgayLap { get; set; }

19 │

20 │ public int? MaKh { get; set; }

21 │

22 │ public int? MaHopDong { get; set; }

23 │

24 │ public int? MaDonHang { get; set; }

25 │

26 │ public decimal? DonGiaCk { get; set; }

27 │

28 │ public decimal? SoLuongCk { get; set; }

29 │

30 │ public decimal? ThanhTienCk { get; set; }

31 │

32 │ public int? MaTienTe { get; set; }

33 │

34 │ public Guid? MaChiTietDuyetHienTai { get; set; }

35 │

36 │ public int? TrangThai { get; set; }

37 │

38 │ public DateTime? NgayChiCk { get; set; }

39 │

40 │ public string? SoTaiKhoanNhan { get; set; }

41 │

42 │ public string? NganHangNhan { get; set; }

43 │

44 │ public string? NguoiNhan { get; set; }

45 │

46 │ public int? UserCreated { get; set; }

47 │

48 │ public DateTime? CreatedDate { get; set; }

49 │

50 │ public int? UserModified { get; set; }

51 │

52 │ public DateTime? ModifiedDate { get; set; }

53 │

54 │ public bool? Deleted { get; set; }

55 │

56 │ public int? UserDeleted { get; set; }

57 │

58 │ public DateTime? DeletedDate { get; set; }

59 │

60 │ public decimal? TyGia { get; set; }

61 │ }

────────────────────────────────────────────────────────────

**📄 PurChiTietDeNghi.cs (CS, 37 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PurChiTietDeNghi

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaDeNghi { get; set; }

11 │

12 │ public int? MaHangHoa { get; set; }

13 │

14 │ public int? MaDvt { get; set; }

15 │

16 │ public int? SoLuong { get; set; }

17 │

18 │ public DateTime? NgaySxtrenBb { get; set; }

19 │

20 │ public string? MauMa { get; set; }

21 │

22 │ public string? GhiChu { get; set; }

23 │

24 │ public int? UserCreated { get; set; }

25 │

26 │ public DateTime? CreatedDate { get; set; }

27 │

28 │ public int? UserModified { get; set; }

29 │

30 │ public DateTime? ModifiedDate { get; set; }

31 │

32 │ public bool? Deleted { get; set; }

33 │

34 │ public int? UserDeleted { get; set; }

35 │

36 │ public DateTime? DeletedDate { get; set; }

37 │ }

────────────────────────────────────────────────────────────

**📄 PurChiTietDonHang.cs (CS, 39 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PurChiTietDonHang

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaCtdeNghi { get; set; }

11 │

12 │ public int? MaDonDatHang { get; set; }

13 │

14 │ public int? MaHangHoa { get; set; }

15 │

16 │ public int? MaDvt { get; set; }

17 │

18 │ public int? SoLuong { get; set; }

19 │

20 │ public decimal? DonGia { get; set; }

21 │

22 │ public decimal? ThanhTien { get; set; }

23 │

24 │ public string? GhiChu { get; set; }

25 │

26 │ public int? UserCreated { get; set; }

27 │

28 │ public DateTime? CreatedDate { get; set; }

29 │

30 │ public int? UserModified { get; set; }

31 │

32 │ public DateTime? ModifiedDate { get; set; }

33 │

34 │ public bool? Deleted { get; set; }

35 │

36 │ public int? UserDeleted { get; set; }

37 │

38 │ public DateTime? DeletedDate { get; set; }

39 │ }

────────────────────────────────────────────────────────────

**📄 PurChiTietHoaDon.cs (CS, 43 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PurChiTietHoaDon

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaCtnhap { get; set; }

11 │

12 │ public int? MaHoaDon { get; set; }

13 │

14 │ public int? MaNhapKho { get; set; }

15 │

16 │ public int? MaHangHoa { get; set; }

17 │

18 │ public int? MaDvt { get; set; }

19 │

20 │ public int? SoLuong { get; set; }

21 │

22 │ public decimal? DonGia { get; set; }

23 │

24 │ public decimal? ThanhTien { get; set; }

25 │

26 │ public decimal? TienThue { get; set; }

27 │

28 │ public decimal? TongThanhTien { get; set; }

29 │

30 │ public int? UserCreated { get; set; }

31 │

32 │ public DateTime? CreatedDate { get; set; }

33 │

34 │ public int? UserModified { get; set; }

35 │

36 │ public DateTime? ModifiedDate { get; set; }

37 │

38 │ public bool? Deleted { get; set; }

39 │

40 │ public int? UserDeleted { get; set; }

41 │

42 │ public DateTime? DeletedDate { get; set; }

43 │ }

────────────────────────────────────────────────────────────

**📄 PurChiTietThanhToan.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PurChiTietThanhToan

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaThanhToan { get; set; }

11 │

12 │ public int? MaHoaDon { get; set; }

13 │

14 │ public decimal? PhaiTra { get; set; }

15 │

16 │ public decimal? DaThanhToan { get; set; }

17 │

18 │ public decimal? ConLai { get; set; }

19 │

20 │ public int? UserCreated { get; set; }

21 │

22 │ public DateTime? CreatedDate { get; set; }

23 │

24 │ public int? UserModified { get; set; }

25 │

26 │ public DateTime? ModifiedDate { get; set; }

27 │

28 │ public bool? Deleted { get; set; }

29 │

30 │ public int? UserDeleted { get; set; }

31 │

32 │ public DateTime? DeletedDate { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 PurDeNghiThanhToan.cs (CS, 49 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PurDeNghiThanhToan

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaChiNhanhPhanBo { get; set; }

13 │

14 │ public int? MaNhaCc { get; set; }

15 │

16 │ public string? SoPhieu { get; set; }

17 │

18 │ public DateTime? NgayDeNghi { get; set; }

19 │

20 │ public string? NoiDung { get; set; }

21 │

22 │ public string? SoTaiKhoan { get; set; }

23 │

24 │ public string? NganHang { get; set; }

25 │

26 │ public decimal? SoTien { get; set; }

27 │

28 │ public decimal? DaThanhToan { get; set; }

29 │

30 │ public decimal? ConLai { get; set; }

31 │

32 │ public string? GhiChu { get; set; }

33 │

34 │ public int? TrangThai { get; set; }

35 │

36 │ public int? UserCreated { get; set; }

37 │

38 │ public DateTime? CreatedDate { get; set; }

39 │

40 │ public int? UserModified { get; set; }

41 │

42 │ public DateTime? ModifiedDate { get; set; }

43 │

44 │ public bool? Deleted { get; set; }

45 │

46 │ public int? UserDeleted { get; set; }

47 │

48 │ public DateTime? DeletedDate { get; set; }

49 │ }

────────────────────────────────────────────────────────────

**📄 PurDonDatHang.cs (CS, 41 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PurDonDatHang

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaChiNhanhDat { get; set; }

13 │

14 │ public int? MaDeNghi { get; set; }

15 │

16 │ public int? MaNhaCc { get; set; }

17 │

18 │ public DateTime? NgayDonHang { get; set; }

19 │

20 │ public string? SoDonHang { get; set; }

21 │

22 │ public DateTime? NgayGiaoHang { get; set; }

23 │

24 │ public string? GhiChu { get; set; }

25 │

26 │ public int? TrangThai { get; set; }

27 │

28 │ public int? UserCreated { get; set; }

29 │

30 │ public DateTime? CreatedDate { get; set; }

31 │

32 │ public int? UserModified { get; set; }

33 │

34 │ public DateTime? ModifiedDate { get; set; }

35 │

36 │ public bool? Deleted { get; set; }

37 │

38 │ public int? UserDeleted { get; set; }

39 │

40 │ public DateTime? DeletedDate { get; set; }

41 │ }

────────────────────────────────────────────────────────────

**📄 PurHoaDon.cs (CS, 55 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PurHoaDon

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaLoaiHoaDon { get; set; }

11 │

12 │ public int? MaChiNhanh { get; set; }

13 │

14 │ public int? MaPhieuNhap { get; set; }

15 │

16 │ public int? MaDeNghi { get; set; }

17 │

18 │ public int? MaNhaCungCap { get; set; }

19 │

20 │ public int? MaDonDatHang { get; set; }

21 │

22 │ public string? SoHoaDon { get; set; }

23 │

24 │ public DateTime? NgayHoaDon { get; set; }

25 │

26 │ public int? ThueVat { get; set; }

27 │

28 │ public decimal? TienThue { get; set; }

29 │

30 │ public decimal? ThanhTien { get; set; }

31 │

32 │ public decimal? PhaiTra { get; set; }

33 │

34 │ public int? TrangThai { get; set; }

35 │

36 │ public decimal? DaThanhToan { get; set; }

37 │

38 │ public decimal? ConLai { get; set; }

39 │

40 │ public string? GhiChu { get; set; }

41 │

42 │ public int? UserCreated { get; set; }

43 │

44 │ public DateTime? CreatedDate { get; set; }

45 │

46 │ public int? UserModified { get; set; }

47 │

48 │ public DateTime? ModifiedDate { get; set; }

49 │

50 │ public bool? Deleted { get; set; }

51 │

52 │ public int? UserDeleted { get; set; }

53 │

54 │ public DateTime? DeletedDate { get; set; }

55 │ }

────────────────────────────────────────────────────────────

**📄 PurPhieuDeNghi.cs (CS, 37 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PurPhieuDeNghi

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public string? SoPhieu { get; set; }

13 │

14 │ public DateTime? NgayDeNghi { get; set; }

15 │

16 │ public DateTime? NgayGiaoDuKien { get; set; }

17 │

18 │ public string? NoiGiao { get; set; }

19 │

20 │ public int? TrangThai { get; set; }

21 │

22 │ public string? GhiChu { get; set; }

23 │

24 │ public int? UserCreated { get; set; }

25 │

26 │ public DateTime? CreatedDate { get; set; }

27 │

28 │ public int? UserModified { get; set; }

29 │

30 │ public DateTime? ModifiedDate { get; set; }

31 │

32 │ public bool? Deleted { get; set; }

33 │

34 │ public int? UserDeleted { get; set; }

35 │

36 │ public DateTime? DeletedDate { get; set; }

37 │ }

────────────────────────────────────────────────────────────

**📄 PurThanhToan.cs (CS, 47 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PurThanhToan

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaDeNghiTt { get; set; }

13 │

14 │ public string? SoPhieu { get; set; }

15 │

16 │ public int? MaNhaCc { get; set; }

17 │

18 │ public int? MaChiNhanhPhanBo { get; set; }

19 │

20 │ public DateTime? NgayThanhToan { get; set; }

21 │

22 │ public string? LyDo { get; set; }

23 │

24 │ public decimal? SoTienThanhToan { get; set; }

25 │

26 │ public int? TrangThai { get; set; }

27 │

28 │ public DateTime? NgayChuyenSoKt { get; set; }

29 │

30 │ public string? SoTaiKhoan { get; set; }

31 │

32 │ public string? NganHang { get; set; }

33 │

34 │ public int? UserCreated { get; set; }

35 │

36 │ public DateTime? CreatedDate { get; set; }

37 │

38 │ public int? UserModified { get; set; }

39 │

40 │ public DateTime? ModifiedDate { get; set; }

41 │

42 │ public bool? Deleted { get; set; }

43 │

44 │ public int? UserDeleted { get; set; }

45 │

46 │ public DateTime? DeletedDate { get; set; }

47 │ }

────────────────────────────────────────────────────────────

**📄 PurTrangThaiDonDatHang.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PurTrangThaiDonDatHang

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public bool? TrangThai { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 PxhPhieuXuat.cs (CS, 37 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class PxhPhieuXuat

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? SoPhieuXuat { get; set; }

11 │

12 │ public string? SoHopDong { get; set; }

13 │

14 │ public DateTime? NgayXuat { get; set; }

15 │

16 │ public int? MaHh { get; set; }

17 │

18 │ public int? SoLuongXuat { get; set; }

19 │

20 │ public string? GhiChu { get; set; }

21 │

22 │ public bool? Status { get; set; }

23 │

24 │ public int? UserCreated { get; set; }

25 │

26 │ public DateTime? CreatedDate { get; set; }

27 │

28 │ public int? UserModified { get; set; }

29 │

30 │ public DateTime? ModifiedDate { get; set; }

31 │

32 │ public bool? Deleted { get; set; }

33 │

34 │ public int? UserDeleted { get; set; }

35 │

36 │ public DateTime? DeletedDate { get; set; }

37 │ }

────────────────────────────────────────────────────────────

**📄 QLCLConst.cs (CS, 87 lines)**

1 │ ﻿using QuanLyChatLuong.Models.APIModels;

2 │

3 │ namespace QuanLyChatLuong.Common

4 │ {

5 │ public class QLCLConst

6 │ {

7 │ public static readonly string JSVersion = "?v=20250706.01";

8 │ public static readonly List<string> AllowDocumentExt = new List<string> { ".DOC", ".DOCX", ".XLS", ".XLSX", ".XL...

9 │ public static readonly List<string> AllowImageExt = new List<string> { ".JPG", ".JPEG", ".BMP", ".GIF", ".PNG" }...

10 │ public static readonly List<string> AllowAudioExt = new List<string> { ".MP3", ".AAC", ".WMA", ".WAV" };

11 │ public static readonly List<string> AllowVideoExt = new List<string> { ".AVI", ".MP4", ".WMV", ".FLV", ".3GP" };

12 │ public static readonly int MaxFileSize = int.MaxValue;//MB

13 │ //public static readonly string UploadPath = @"C:\Uploads\";

14 │ public static readonly string UploadPath = @"D:\Uploads\";

15 │ public static readonly string DownloadPath = @"/File/DownloadFile?filename=";

16 │ public static readonly string DocumentPath = UploadPath + "documents";

17 │ public static readonly string ImagePath = UploadPath + "images";

18 │ public static readonly string AudioPath = UploadPath + "audios";

19 │ public static readonly string VideoPath = UploadPath + "videos";

20 │ public static readonly string SecurePath = UploadPath + "secure";

21 │

22 │ public static int smsLimit = 3;

23 │ public static int OTPLenght = 6;

24 │ public static string wwwRoot = "";

25 │

26 │ public static string UserNotFound = "UsernotFound";

27 │ public static string OK = "OK";

28 │ public static string Error = "Error";

29 │ public const string Pending = "Pending";

30 │

31 │

32 │ public static string YES = "YES";

33 │ public static string NO = "NO";

34 │ public static string PerrmissionDenied = "PerrmissionDenied";

35 │ public static string TokenIsNotCorrect = "TokenIsNotCorrect";

36 │

37 │ public static string OTPRequired = "OTPRequired";

38 │ public static string SDTRequired = "SDTRequired";

39 │ public static string DEVLogin = "DEVLogin";

40 │

41 │ public static string Category = "Danh mục";

42 │ public static string System = "Hệ thống";

43 │ public static string Dashboard = "Dashboard";

44 │ public static string Inactive = "Hết hoạt động";

45 │ public static string Active = "Hoạt động";

46 │

47 │ public static string AddSuccessfully = "Thêm thành công";

48 │ public static string AddFailure = "Thêm thất bại";

49 │ public static string ObjectExists = "Đối tượng đã tồn tại";

50 │ public static string UpdateSuccessfully = "Cập nhật thành công";

51 │ public static string UpdateFailure = "Cập nhật thất bại";

52 │ public static string NotExist = "Đối tượng không tồn tại";

53 │ public static string DeleteSuccessfully = "Xóa thành công";

54 │ public static string DeleteFailure = "Xóa không thành công";

55 │ public static string CannotDelete = "Không thể xóa";

56 │ public static string NoPermission = "Bạn không có quyền thực hiện thao tác này, vui lòng liên hệ...

57 │ public static string HaveError = "Đã xảy ra lỗi";

58 │ public static string FileError = "Lỗi tập tin";

59 │ public static string LockedObject = "Đối tượng đã bị khoá";

60 │ public static string Successfully = "Thành công";

61 │ public static string NotSuccessfully = "Không thành công";

62 │ public static string CodeCannotEdit = "Mã đối tượng đã phát sinh giao dịch";

63 │ public static string DateFormatIsIncorrect = "Định dạng ngày không đúng";

64 │ public static string NoPermissionHTML = "Bạn không có quyền thực hiện thao tác này,<br/> vui lòng l...

65 │

66 │ public static ResponseModel ResponseAddSuccessfully = new ResponseModelSuccess(AddSuccessfully);

67 │ public static ResponseModel ResponseAddFailure = new ResponseModelError(AddFailure);

68 │ public static ResponseModel ResponseObjectExists = new ResponseModelError(ObjectExists);

69 │ public static ResponseModel ResponseUpdateSuccessfully = new ResponseModelSuccess(UpdateSuccessfully);

70 │ public static ResponseModel ResponseUpdateFailure = new ResponseModelError(UpdateFailure);

71 │ public static ResponseModel ResponseNotExist = new ResponseModelError(NotExist);

72 │ public static ResponseModel ResponseDeleteSuccessfully = new ResponseModelSuccess(DeleteSuccessfully);

73 │ public static ResponseModel ResponseDeleteFailure = new ResponseModelError(DeleteFailure);

74 │ public static ResponseModel ResponseCannotDelete = new ResponseModelError(CannotDelete);

75 │ public static ResponseModel ResponseNoPermission = new ResponseModelError(NoPermission);

76 │ public static ResponseModel ResponseHaveError = new ResponseModelError(HaveError);

77 │ public static ResponseModel ResponseFileError = new ResponseModelError(FileError);

78 │ public static ResponseModel ResponseLockedObject = new ResponseModelError(LockedObject);

79 │ public static ResponseModel ResponseSuccessfully = new ResponseModelSuccess(Successfully);

80 │ public static ResponseModel ResponseCodetCannotEdit = new ResponseModelError(CodeCannotEdit);

81 │ public static ResponseModel ResponseDateFormatIsIncorrect = new ResponseModelError(DateFormatIsIncorrect);

82 │ public static ResponseModel notFound = new ResponseModelError("Không tìm thấy !!");

83 │

84 │ public static string cacheKeyMenu = "menu-";

85 │ public static string passcode = "Qu@nLyCh@tLu0nG";

86 │ }

87 │ }

────────────────────────────────────────────────────────────

**📄 QuanLyChatLuongContext.cs (CS, 47 lines)**

1 │ ﻿using Microsoft.EntityFrameworkCore;

2 │ using QuanLyChatLuong.Models.StoreProcedures;

3 │

4 │ namespace QuanLyChatLuong.Models

5 │ {

6 │ public partial class QuanLyChatLuongDbContext

7 │ {

8 │ public virtual DbSet<sp\_RoleMenuResult> sp\_RoleMenu { get; set; }

9 │ public virtual DbSet<sp\_RolePermissionResult> sp\_RolePermission { get; set; }

10 │ public virtual DbSet<sp\_BaoCaoSoLieuBienTinhSoSanhTheoNgayResult> sp\_BaoCaoSoLieuBienTinhSSTheoNgay { get; set; ...

11 │ public virtual DbSet<sp\_BaoCaoSoLieuBienTinhTheoNgayResult> sp\_BaoCaoSoLieuBienTinhTheoNgay { get; set; }

12 │ public virtual DbSet<sp\_BaoCaoSoLieuBienTinhTuNgayDenNgayResult> sp\_BaoCaoSoLieuBienTinhTuNgayDenNgay { get; set...

13 │ public virtual DbSet<sp\_DeviceListResult> sp\_DeviceListResult { get; set; }

14 │ public virtual DbSet<sp\_HienThiLedResult> Sp\_HienThiLedResults { get; set; }

15 │ partial void OnModelCreatingPartial(ModelBuilder modelBuilder)

16 │ {

17 │ modelBuilder.Entity<sp\_RoleMenuResult>(entity =>

18 │ {

19 │ entity.HasNoKey();

20 │ });

21 │ modelBuilder.Entity<sp\_RolePermissionResult>(entity =>

22 │ {

23 │ entity.HasNoKey();

24 │ });

25 │ modelBuilder.Entity<sp\_BaoCaoSoLieuBienTinhSoSanhTheoNgayResult>(entity =>

26 │ {

27 │ entity.HasNoKey();

28 │ });

29 │ modelBuilder.Entity<sp\_BaoCaoSoLieuBienTinhTheoNgayResult>(entity =>

30 │ {

31 │ entity.HasNoKey();

32 │ });

33 │ modelBuilder.Entity<sp\_BaoCaoSoLieuBienTinhTuNgayDenNgayResult>(entity =>

34 │ {

35 │ entity.HasNoKey();

36 │ });

37 │ modelBuilder.Entity<sp\_DeviceListResult>(entity =>

38 │ {

39 │ entity.HasNoKey();

40 │ });

41 │ modelBuilder.Entity<sp\_HienThiLedResult>(entity =>

42 │ {

43 │ entity.HasNoKey();

44 │ });

45 │ }

46 │ }

47 │ }

────────────────────────────────────────────────────────────

**📄 QuanLyChatLuongDbContext.cs (CS, 808 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │ using Microsoft.EntityFrameworkCore;

4 │

5 │ namespace QuanLyChatLuong.Models;

6 │

7 │ public partial class QuanLyChatLuongDbContext : DbContext

8 │ {

9 │ public QuanLyChatLuongDbContext()

10 │ {

11 │ }

12 │

13 │ public QuanLyChatLuongDbContext(DbContextOptions<QuanLyChatLuongDbContext> options)

14 │ : base(options)

15 │ {

16 │ }

17 │

18 │ public virtual DbSet<ThoiGian> ThoiGians { get; set; }

19 │

20 │ public virtual DbSet<arduino\_CheckInOut> arduino\_CheckInOuts { get; set; }

21 │

22 │ public virtual DbSet<arduino\_Log> arduino\_Logs { get; set; }

23 │

24 │ public virtual DbSet<arduino\_ThietBi> arduino\_ThietBis { get; set; }

25 │

26 │ public virtual DbSet<bt\_BanThanhPham> bt\_BanThanhPhams { get; set; }

27 │

28 │ public virtual DbSet<bt\_BaoBi> bt\_BaoBis { get; set; }

29 │

30 │ public virtual DbSet<bt\_ChiTietSoLieu> bt\_ChiTietSoLieus { get; set; }

31 │

32 │ public virtual DbSet<bt\_ChiTietTieuChuan> bt\_ChiTietTieuChuans { get; set; }

33 │

34 │ public virtual DbSet<bt\_DauVao> bt\_DauVaos { get; set; }

35 │

36 │ public virtual DbSet<bt\_SoLieuBienTinh> bt\_SoLieuBienTinhs { get; set; }

37 │

38 │ public virtual DbSet<bt\_TenTieuDeBaoCao> bt\_TenTieuDeBaoCaos { get; set; }

39 │

40 │ public virtual DbSet<bt\_TieuChuan> bt\_TieuChuans { get; set; }

41 │

42 │ public virtual DbSet<sx\_SoLieuSX> sx\_SoLieuSXes { get; set; }

43 │

44 │ public virtual DbSet<sys\_Menu> sys\_Menus { get; set; }

45 │

46 │ public virtual DbSet<sys\_OTP> sys\_OTPs { get; set; }

47 │

48 │ public virtual DbSet<sys\_Quyen> sys\_Quyens { get; set; }

49 │

50 │ public virtual DbSet<sys\_Role> sys\_Roles { get; set; }

51 │

52 │ public virtual DbSet<sys\_RoleMenu> sys\_RoleMenus { get; set; }

53 │

54 │ public virtual DbSet<sys\_RoleQuyen> sys\_RoleQuyens { get; set; }

55 │

56 │ public virtual DbSet<sys\_Token> sys\_Tokens { get; set; }

57 │

58 │ public virtual DbSet<sys\_User> sys\_Users { get; set; }

59 │

60 │ public virtual DbSet<sys\_UserRole> sys\_UserRoles { get; set; }

61 │

62 │ public virtual DbSet<tbl\_BanThanhPham> tbl\_BanThanhPhams { get; set; }

63 │

64 │ public virtual DbSet<tbl\_ChiNhanh> tbl\_ChiNhanhs { get; set; }

65 │

66 │ public virtual DbSet<tbl\_ChiTietKiemTra> tbl\_ChiTietKiemTras { get; set; }

67 │

68 │ public virtual DbSet<tbl\_ChiTietSoLieu> tbl\_ChiTietSoLieus { get; set; }

69 │

70 │ public virtual DbSet<tbl\_ChiTieuCL> tbl\_ChiTieuCLs { get; set; }

71 │

72 │ public virtual DbSet<tbl\_ColorGioiHanSoLieu> tbl\_ColorGioiHanSoLieus { get; set; }

73 │

74 │ public virtual DbSet<tbl\_DanhMucSoLieu> tbl\_DanhMucSoLieus { get; set; }

75 │

76 │ public virtual DbSet<tbl\_GioiHanSoLieu> tbl\_GioiHanSoLieus { get; set; }

77 │

78 │ public virtual DbSet<tbl\_KiemTraChatLuong> tbl\_KiemTraChatLuongs { get; set; }

79 │

80 │ public virtual DbSet<tbl\_LoaiSoLieu> tbl\_LoaiSoLieus { get; set; }

81 │

82 │ public virtual DbSet<tbl\_ManLed> tbl\_ManLeds { get; set; }

83 │

84 │ public virtual DbSet<tbl\_NhomSoLieu> tbl\_NhomSoLieus { get; set; }

85 │

86 │ public virtual DbSet<tbl\_SoLieu> tbl\_SoLieus { get; set; }

87 │

88 │ public virtual DbSet<tmp> tmps { get; set; }

89 │

90 │ public virtual DbSet<tmpDatum> tmpData { get; set; }

91 │

92 │ public virtual DbSet<vQuyen> vQuyens { get; set; }

93 │

94 │ protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)

95 │ => optionsBuilder.UseSqlServer("Name=ConnectionStrings:QuanLyChatLuong");

96 │

97 │ protected override void OnModelCreating(ModelBuilder modelBuilder)

98 │ {

99 │ modelBuilder.Entity<ThoiGian>(entity =>

100 │ {

101 │ entity

102 │ .HasNoKey()

103 │ .ToTable("ThoiGian");

104 │

105 │ entity.Property(e => e.Gio).HasMaxLength(2);

106 │ entity.Property(e => e.Ngay).HasColumnType("datetime");

107 │ entity.Property(e => e.Phut).HasMaxLength(2);

108 │ });

109 │

110 │ modelBuilder.Entity<arduino\_CheckInOut>(entity =>

111 │ {

112 │ entity

113 │ .HasNoKey()

114 │ .ToTable("arduino\_CheckInOut");

115 │

116 │ entity.Property(e => e.Ma).ValueGeneratedOnAdd();

117 │ entity.Property(e => e.NgayGioHeThong).HasMaxLength(50);

118 │ entity.Property(e => e.NgayGioThietBi).HasMaxLength(50);

119 │ entity.Property(e => e.Status).HasMaxLength(50);

120 │ });

121 │

122 │ modelBuilder.Entity<arduino\_Log>(entity =>

123 │ {

124 │ entity.HasKey(e => e.MA).HasName("PK\_Log");

125 │

126 │ entity.ToTable("arduino\_Log");

127 │

128 │ entity.Property(e => e.NgayGioHeThong).HasColumnType("datetime");

129 │ entity.Property(e => e.NgayGioThietBi).HasMaxLength(50);

130 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

131 │ entity.Property(e => e.NgayTao)

132 │ .HasDefaultValueSql("(getdate())")

133 │ .HasColumnType("datetime");

134 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

135 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

136 │ });

137 │

138 │ modelBuilder.Entity<arduino\_ThietBi>(entity =>

139 │ {

140 │ entity.HasKey(e => e.MaThietBi).HasName("PK\_thietbi");

141 │

142 │ entity.ToTable("arduino\_ThietBi");

143 │

144 │ entity.Property(e => e.NgayDangKi).HasMaxLength(50);

145 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

146 │ entity.Property(e => e.NgayTao)

147 │ .HasDefaultValueSql("(getdate())")

148 │ .HasColumnType("datetime");

149 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

150 │ entity.Property(e => e.TenThietBi).HasMaxLength(50);

151 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

152 │ entity.Property(e => e.ViTri).HasMaxLength(50);

153 │ });

154 │

155 │ modelBuilder.Entity<bt\_BanThanhPham>(entity =>

156 │ {

157 │ entity.HasKey(e => e.Ma).HasName("PK\_\_bt\_BanTh\_\_3214CC9F7E074B50");

158 │

159 │ entity.ToTable("bt\_BanThanhPham");

160 │

161 │ entity.Property(e => e.BonDem).HasMaxLength(200);

162 │ entity.Property(e => e.BonPhanUng).HasMaxLength(200);

163 │ entity.Property(e => e.BrabenderViscosity).HasColumnType("decimal(18, 4)");

164 │ entity.Property(e => e.ConductivityOfHydrocarbons).HasColumnType("decimal(18, 4)");

165 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

166 │ entity.Property(e => e.DS).HasColumnType("decimal(18, 4)");

167 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

168 │ entity.Property(e => e.FreeClo).HasColumnType("decimal(18, 4)");

169 │ entity.Property(e => e.GelatinnizationTemperature).HasColumnType("decimal(18, 4)");

170 │ entity.Property(e => e.GhiChu).HasMaxLength(200);

171 │ entity.Property(e => e.Gio).HasMaxLength(2);

172 │ entity.Property(e => e.MaSanPham).HasMaxLength(200);

173 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

174 │ entity.Property(e => e.Ngay).HasColumnType("datetime");

175 │ entity.Property(e => e.Note).HasMaxLength(1000);

176 │ entity.Property(e => e.PHSB).HasColumnType("decimal(18, 4)");

177 │ entity.Property(e => e.PercentAcetyl).HasColumnType("decimal(18, 4)");

178 │ entity.Property(e => e.PhanUng).HasMaxLength(200);

179 │ entity.Property(e => e.Phut).HasMaxLength(2);

180 │ entity.Property(e => e.QC).HasMaxLength(200);

181 │ entity.Property(e => e.SulfurDioxide).HasColumnType("decimal(18, 4)");

182 │ entity.Property(e => e.TruongCa).HasMaxLength(200);

183 │ entity.Property(e => e.UserAnalyst).HasMaxLength(200);

184 │ });

185 │

186 │ modelBuilder.Entity<bt\_BaoBi>(entity =>

187 │ {

188 │ entity.HasKey(e => e.Ma);

189 │

190 │ entity.ToTable("bt\_BaoBi");

191 │

192 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

193 │ entity.Property(e => e.NgayTao).HasColumnType("datetime");

194 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

195 │ entity.Property(e => e.VatLieuBaoBi).HasMaxLength(100);

196 │ });

197 │

198 │ modelBuilder.Entity<bt\_ChiTietSoLieu>(entity =>

199 │ {

200 │ entity.HasKey(e => e.Ma);

201 │

202 │ entity.ToTable("bt\_ChiTietSoLieu");

203 │

204 │ entity.Property(e => e.AcetylGroup).HasColumnType("decimal(18, 4)");

205 │ entity.Property(e => e.Col1).HasMaxLength(20);

206 │ entity.Property(e => e.Col2).HasMaxLength(20);

207 │ entity.Property(e => e.Col3).HasMaxLength(20);

208 │ entity.Property(e => e.Col4).HasMaxLength(20);

209 │ entity.Property(e => e.Conductivity).HasColumnType("decimal(18, 4)");

210 │ entity.Property(e => e.DS).HasColumnType("decimal(18, 4)");

211 │ entity.Property(e => e.Fineness).HasColumnType("decimal(18, 4)");

212 │ entity.Property(e => e.ForeignMatter).HasMaxLength(50);

213 │ entity.Property(e => e.GelatinnizationTemperature).HasColumnType("decimal(18, 4)");

214 │ entity.Property(e => e.Gio).HasMaxLength(2);

215 │ entity.Property(e => e.Moisture).HasColumnType("decimal(18, 4)");

216 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

217 │ entity.Property(e => e.NgayTao)

218 │ .HasDefaultValueSql("(getdate())")

219 │ .HasColumnType("datetime");

220 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

221 │ entity.Property(e => e.Note).HasMaxLength(200);

222 │ entity.Property(e => e.Phut).HasMaxLength(2);

223 │ entity.Property(e => e.Residue).HasColumnType("decimal(18, 4)");

224 │ entity.Property(e => e.SO2).HasColumnType("decimal(18, 4)");

225 │ entity.Property(e => e.Sensory).HasMaxLength(100);

226 │ entity.Property(e => e.StarchContent).HasColumnType("decimal(18, 4)");

227 │ entity.Property(e => e.ViscosityEndPointBU).HasColumnType("decimal(18, 4)");

228 │ entity.Property(e => e.ViscosityMax950C0MinBU).HasColumnType("decimal(18, 4)");

229 │ entity.Property(e => e.ViscosityMax950C10MinBU).HasColumnType("decimal(18, 4)");

230 │ entity.Property(e => e.ViscosityPeakBU).HasColumnType("decimal(18, 4)");

231 │ entity.Property(e => e.ViscosityPeakCP).HasColumnType("decimal(18, 4)");

232 │ entity.Property(e => e.Whiteness).HasColumnType("decimal(18, 4)");

233 │ entity.Property(e => e.pHValue).HasColumnType("decimal(18, 4)");

234 │ });

235 │

236 │ modelBuilder.Entity<bt\_ChiTietTieuChuan>(entity =>

237 │ {

238 │ entity.HasKey(e => e.Ma);

239 │

240 │ entity.ToTable("bt\_ChiTietTieuChuan");

241 │

242 │ entity.Property(e => e.Method).HasMaxLength(100);

243 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

244 │ entity.Property(e => e.NgayTao).HasColumnType("datetime");

245 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

246 │ entity.Property(e => e.Specification).HasMaxLength(100);

247 │ entity.Property(e => e.TestItem).HasMaxLength(100);

248 │ entity.Property(e => e.Unit).HasMaxLength(100);

249 │ });

250 │

251 │ modelBuilder.Entity<bt\_DauVao>(entity =>

252 │ {

253 │ entity.HasKey(e => e.Ma).HasName("PK\_\_bt\_DauVa\_\_3214CC9FCBE7E3E5");

254 │

255 │ entity.ToTable("bt\_DauVao");

256 │

257 │ entity.Property(e => e.BaumeSB).HasColumnType("decimal(18, 4)");

258 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

259 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

260 │ entity.Property(e => e.HCl).HasColumnType("decimal(18, 4)");

261 │ entity.Property(e => e.HClLoang).HasColumnType("decimal(18, 4)");

262 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

263 │ entity.Property(e => e.NaOH).HasColumnType("decimal(18, 4)");

264 │ entity.Property(e => e.NaOHLoang).HasColumnType("decimal(18, 4)");

265 │ entity.Property(e => e.RasperChao).HasColumnType("decimal(18, 4)");

266 │ entity.Property(e => e.pHValue).HasColumnType("decimal(18, 4)");

267 │ });

268 │

269 │ modelBuilder.Entity<bt\_SoLieuBienTinh>(entity =>

270 │ {

271 │ entity.HasKey(e => e.Ma);

272 │

273 │ entity.ToTable("bt\_SoLieuBienTinh");

274 │

275 │ entity.Property(e => e.DonHang).HasMaxLength(200);

276 │ entity.Property(e => e.DotLo).HasMaxLength(100);

277 │ entity.Property(e => e.MaLo).HasMaxLength(200);

278 │ entity.Property(e => e.Ngay).HasColumnType("datetime");

279 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

280 │ entity.Property(e => e.NgayTao)

281 │ .HasDefaultValueSql("(getdate())")

282 │ .HasColumnType("datetime");

283 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

284 │ entity.Property(e => e.QC).HasMaxLength(100);

285 │ entity.Property(e => e.TenSanPham).HasMaxLength(200);

286 │ entity.Property(e => e.ThuKhoBot).HasMaxLength(100);

287 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

288 │ entity.Property(e => e.TruongBPTP).HasMaxLength(100);

289 │ entity.Property(e => e.TruongCa).HasMaxLength(100);

290 │ });

291 │

292 │ modelBuilder.Entity<bt\_TenTieuDeBaoCao>(entity =>

293 │ {

294 │ entity.ToTable("bt\_TenTieuDeBaoCao");

295 │

296 │ entity.Property(e => e.ColumnName).HasMaxLength(50);

297 │ entity.Property(e => e.ColumnText).HasMaxLength(100);

298 │ });

299 │

300 │ modelBuilder.Entity<bt\_TieuChuan>(entity =>

301 │ {

302 │ entity.HasKey(e => e.Ma);

303 │

304 │ entity.ToTable("bt\_TieuChuan");

305 │

306 │ entity.Property(e => e.Method).HasMaxLength(100);

307 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

308 │ entity.Property(e => e.NgayTao).HasColumnType("datetime");

309 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

310 │ entity.Property(e => e.Specification).HasMaxLength(100);

311 │ entity.Property(e => e.TestingItem).HasMaxLength(100);

312 │ entity.Property(e => e.TieuChuan).HasMaxLength(50);

313 │ entity.Property(e => e.Unit).HasMaxLength(50);

314 │ });

315 │

316 │ modelBuilder.Entity<sx\_SoLieuSX>(entity =>

317 │ {

318 │ entity.HasKey(e => e.Ma);

319 │

320 │ entity.ToTable("sx\_SoLieuSX");

321 │

322 │ entity.Property(e => e.DotLo).HasMaxLength(100);

323 │ entity.Property(e => e.Ngay).HasColumnType("datetime");

324 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

325 │ entity.Property(e => e.NgayTao)

326 │ .HasDefaultValueSql("(getdate())")

327 │ .HasColumnType("datetime");

328 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

329 │ entity.Property(e => e.QC).HasMaxLength(100);

330 │ entity.Property(e => e.ThuKhoBot).HasMaxLength(100);

331 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

332 │ entity.Property(e => e.TruongBP).HasMaxLength(100);

333 │ entity.Property(e => e.TruongBPTP).HasMaxLength(100);

334 │ });

335 │

336 │ modelBuilder.Entity<sys\_Menu>(entity =>

337 │ {

338 │ entity.HasKey(e => e.Ma);

339 │

340 │ entity.ToTable("sys\_Menu");

341 │

342 │ entity.Property(e => e.Code).HasMaxLength(100);

343 │ entity.Property(e => e.DuongDan).HasMaxLength(300);

344 │ entity.Property(e => e.Icon).HasMaxLength(200);

345 │ entity.Property(e => e.IconWeb).HasMaxLength(50);

346 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

347 │ entity.Property(e => e.NgayTao)

348 │ .HasDefaultValueSql("(getdate())")

349 │ .HasColumnType("datetime");

350 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

351 │ entity.Property(e => e.Ten).HasMaxLength(300);

352 │ entity.Property(e => e.ThamSo).HasMaxLength(300);

353 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

354 │ });

355 │

356 │ modelBuilder.Entity<sys\_OTP>(entity =>

357 │ {

358 │ entity.HasKey(e => e.Ma).HasName("PK\_\_sys\_OTP\_\_3214CC9FE69B141D");

359 │

360 │ entity.ToTable("sys\_OTP");

361 │

362 │ entity.Property(e => e.CreateDate).HasColumnType("datetime");

363 │ entity.Property(e => e.OTP).HasMaxLength(500);

364 │ });

365 │

366 │ modelBuilder.Entity<sys\_Quyen>(entity =>

367 │ {

368 │ entity.HasKey(e => e.Ma);

369 │

370 │ entity.ToTable("sys\_Quyen");

371 │

372 │ entity.Property(e => e.Code).HasMaxLength(50);

373 │ entity.Property(e => e.MoTa).HasMaxLength(300);

374 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

375 │ entity.Property(e => e.NgayTao)

376 │ .HasDefaultValueSql("(getdate())")

377 │ .HasColumnType("datetime");

378 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

379 │ entity.Property(e => e.Ten).HasMaxLength(250);

380 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

381 │ });

382 │

383 │ modelBuilder.Entity<sys\_Role>(entity =>

384 │ {

385 │ entity.HasKey(e => e.Ma);

386 │

387 │ entity.ToTable("sys\_Role");

388 │

389 │ entity.Property(e => e.Code).HasMaxLength(50);

390 │ entity.Property(e => e.MoTa).HasMaxLength(300);

391 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

392 │ entity.Property(e => e.NgayTao)

393 │ .HasDefaultValueSql("(getdate())")

394 │ .HasColumnType("datetime");

395 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

396 │ entity.Property(e => e.Ten).HasMaxLength(250);

397 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

398 │ });

399 │

400 │ modelBuilder.Entity<sys\_RoleMenu>(entity =>

401 │ {

402 │ entity.HasKey(e => e.Ma);

403 │

404 │ entity.ToTable("sys\_RoleMenu");

405 │

406 │ entity.Property(e => e.ChoPhepTruyCap).HasDefaultValue(true);

407 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

408 │ entity.Property(e => e.NgayTao)

409 │ .HasDefaultValueSql("(getdate())")

410 │ .HasColumnType("datetime");

411 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

412 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

413 │ });

414 │

415 │ modelBuilder.Entity<sys\_RoleQuyen>(entity =>

416 │ {

417 │ entity.HasKey(e => e.Ma);

418 │

419 │ entity.ToTable("sys\_RoleQuyen");

420 │

421 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

422 │ entity.Property(e => e.NgayTao)

423 │ .HasDefaultValueSql("(getdate())")

424 │ .HasColumnType("datetime");

425 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

426 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

427 │ });

428 │

429 │ modelBuilder.Entity<sys\_Token>(entity =>

430 │ {

431 │ entity.HasKey(e => e.Ma).HasName("PK\_\_sys\_Toke\_\_3214CC9F2C683A63");

432 │

433 │ entity.ToTable("sys\_Token");

434 │

435 │ entity.Property(e => e.DeviceId).HasMaxLength(255);

436 │ entity.Property(e => e.DeviceName).HasMaxLength(255);

437 │ entity.Property(e => e.Ip).HasMaxLength(50);

438 │ entity.Property(e => e.LastChangeIp).HasColumnType("datetime");

439 │ });

440 │

441 │ modelBuilder.Entity<sys\_User>(entity =>

442 │ {

443 │ entity.HasKey(e => e.Ma);

444 │

445 │ entity.ToTable("sys\_User");

446 │

447 │ entity.Property(e => e.Email).HasMaxLength(50);

448 │ entity.Property(e => e.FCM).HasMaxLength(200);

449 │ entity.Property(e => e.HoTen).HasMaxLength(100);

450 │ entity.Property(e => e.MSNV)

451 │ .HasMaxLength(30)

452 │ .IsUnicode(false);

453 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

454 │ entity.Property(e => e.NgayTao)

455 │ .HasDefaultValueSql("(getdate())")

456 │ .HasColumnType("datetime");

457 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

458 │ entity.Property(e => e.Password).HasMaxLength(300);

459 │ entity.Property(e => e.Photo).HasColumnType("image");

460 │ entity.Property(e => e.SDT).HasMaxLength(20);

461 │ entity.Property(e => e.TenDangNhap).HasMaxLength(100);

462 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

463 │ });

464 │

465 │ modelBuilder.Entity<sys\_UserRole>(entity =>

466 │ {

467 │ entity.HasKey(e => e.Ma);

468 │

469 │ entity.ToTable("sys\_UserRole");

470 │

471 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

472 │ entity.Property(e => e.NgayTao)

473 │ .HasDefaultValueSql("(getdate())")

474 │ .HasColumnType("datetime");

475 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

476 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

477 │ });

478 │

479 │ modelBuilder.Entity<tbl\_BanThanhPham>(entity =>

480 │ {

481 │ entity.HasKey(e => e.Ma);

482 │

483 │ entity.ToTable("tbl\_BanThanhPham");

484 │

485 │ entity.Property(e => e.AcetylGroup).HasColumnType("decimal(18, 4)");

486 │ entity.Property(e => e.BauMeSuaBot).HasMaxLength(100);

487 │ entity.Property(e => e.CQ\_SodiumCarbonat).HasColumnType("decimal(18, 4)");

488 │ entity.Property(e => e.CamQuanNaHCO3).HasMaxLength(100);

489 │ entity.Property(e => e.Clo).HasColumnType("decimal(18, 4)");

490 │ entity.Property(e => e.CreatedDate).HasColumnType("datetime");

491 │ entity.Property(e => e.DS).HasColumnType("decimal(18, 4)");

492 │ entity.Property(e => e.DeletedDate).HasColumnType("datetime");

493 │ entity.Property(e => e.DoNhot).HasColumnType("decimal(18, 4)");

494 │ entity.Property(e => e.GelatinnizationTemperature).HasColumnType("decimal(18, 4)");

495 │ entity.Property(e => e.GhiChu).HasMaxLength(50);

496 │ entity.Property(e => e.Gio)

497 │ .HasMaxLength(2)

498 │ .IsFixedLength();

499 │ entity.Property(e => e.HCL).HasColumnType("decimal(18, 4)");

500 │ entity.Property(e => e.HCL\_Loang).HasColumnType("decimal(18, 4)");

501 │ entity.Property(e => e.HamLuongTBTrongBaMi).HasMaxLength(50);

502 │ entity.Property(e => e.ModifiedDate).HasColumnType("datetime");

503 │ entity.Property(e => e.ND\_HoHoa).HasColumnType("decimal(18, 4)");

504 │ entity.Property(e => e.NaHCO3).HasMaxLength(100);

505 │ entity.Property(e => e.NaOH).HasColumnType("decimal(18, 4)");

506 │ entity.Property(e => e.NaOH\_Loang).HasColumnType("decimal(18, 4)");

507 │ entity.Property(e => e.NguoiPhanTich).HasMaxLength(50);

508 │ entity.Property(e => e.Note).HasMaxLength(200);

509 │ entity.Property(e => e.Phut)

510 │ .HasMaxLength(2)

511 │ .IsFixedLength();

512 │ entity.Property(e => e.RasperChao).HasColumnType("decimal(18, 4)");

513 │ entity.Property(e => e.SO2).HasColumnType("decimal(18, 4)");

514 │ entity.Property(e => e.TinhBotTrongBaMi).HasMaxLength(100);

515 │ entity.Property(e => e.ViscosityEndPointBU).HasColumnType("decimal(18, 4)");

516 │ entity.Property(e => e.ViscosityMax950C0MinBU).HasColumnType("decimal(18, 4)");

517 │ entity.Property(e => e.ViscosityMax950C10MinBU).HasColumnType("decimal(18, 4)");

518 │ entity.Property(e => e.ViscosityPeakBU).HasColumnType("decimal(18, 4)");

519 │ entity.Property(e => e.pHSuaBot).HasMaxLength(50);

520 │ entity.Property(e => e.pH\_RO).HasColumnType("decimal(18, 4)");

521 │ });

522 │

523 │ modelBuilder.Entity<tbl\_ChiNhanh>(entity =>

524 │ {

525 │ entity.HasKey(e => e.Ma);

526 │

527 │ entity.ToTable("tbl\_ChiNhanh");

528 │

529 │ entity.Property(e => e.Code).HasMaxLength(20);

530 │ entity.Property(e => e.DiaChi).HasMaxLength(250);

531 │ entity.Property(e => e.DienThoai).HasMaxLength(50);

532 │ entity.Property(e => e.Email).HasMaxLength(100);

533 │ entity.Property(e => e.Fax).HasMaxLength(50);

534 │ entity.Property(e => e.Logo).HasColumnType("image");

535 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

536 │ entity.Property(e => e.NgayTao)

537 │ .HasDefaultValueSql("(getdate())")

538 │ .HasColumnType("datetime");

539 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

540 │ entity.Property(e => e.Ten).HasMaxLength(150);

541 │ entity.Property(e => e.TenDayDu).HasMaxLength(150);

542 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

543 │ entity.Property(e => e.Website).HasMaxLength(150);

544 │ });

545 │

546 │ modelBuilder.Entity<tbl\_ChiTietKiemTra>(entity =>

547 │ {

548 │ entity.HasKey(e => e.Ma);

549 │

550 │ entity.ToTable("tbl\_ChiTietKiemTra");

551 │

552 │ entity.Property(e => e.Code).HasMaxLength(50);

553 │ entity.Property(e => e.DVT).HasMaxLength(50);

554 │ entity.Property(e => e.KetQua).HasMaxLength(50);

555 │ entity.Property(e => e.Method).HasMaxLength(100);

556 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

557 │ entity.Property(e => e.NgayTao)

558 │ .HasDefaultValueSql("(getdate())")

559 │ .HasColumnType("datetime");

560 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

561 │ entity.Property(e => e.TieuChuan).HasMaxLength(100);

562 │ });

563 │

564 │ modelBuilder.Entity<tbl\_ChiTietSoLieu>(entity =>

565 │ {

566 │ entity.HasKey(e => e.Ma);

567 │

568 │ entity.ToTable("tbl\_ChiTietSoLieu");

569 │

570 │ entity.Property(e => e.BU\_DoAm).HasColumnType("decimal(18, 4)");

571 │ entity.Property(e => e.B\_DoAm).HasColumnType("decimal(18, 4)");

572 │ entity.Property(e => e.B\_DoKeoBU).HasColumnType("decimal(18, 4)");

573 │ entity.Property(e => e.B\_DoKeoCP).HasColumnType("decimal(18, 4)");

574 │ entity.Property(e => e.B\_DoMin).HasColumnType("decimal(18, 4)");

575 │ entity.Property(e => e.B\_DoPH).HasColumnType("decimal(18, 4)");

576 │ entity.Property(e => e.B\_DoTrang).HasColumnType("decimal(18, 4)");

577 │ entity.Property(e => e.B\_TapChat).HasColumnType("decimal(18, 4)");

578 │ entity.Property(e => e.Cepa).HasMaxLength(200);

579 │ entity.Property(e => e.ForeignMatter).HasMaxLength(200);

580 │ entity.Property(e => e.GhiChu).HasMaxLength(2000);

581 │ entity.Property(e => e.Gio).HasMaxLength(2);

582 │ entity.Property(e => e.HamLuongTB).HasColumnType("decimal(18, 4)");

583 │ entity.Property(e => e.ND\_HoHoa).HasColumnType("decimal(18, 4)");

584 │ entity.Property(e => e.ND\_Lo).HasColumnType("decimal(18, 4)");

585 │ entity.Property(e => e.ND\_Say).HasColumnType("decimal(18, 4)");

586 │ entity.Property(e => e.NS\_DoPH).HasColumnType("decimal(18, 4)");

587 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

588 │ entity.Property(e => e.NgayTao)

589 │ .HasDefaultValueSql("(getdate())")

590 │ .HasColumnType("datetime");

591 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

592 │ entity.Property(e => e.Phut).HasMaxLength(2);

593 │ entity.Property(e => e.SO2).HasColumnType("decimal(18, 4)");

594 │ entity.Property(e => e.Sensory).HasMaxLength(200);

595 │ entity.Property(e => e.XMS\_DoAm).HasColumnType("decimal(18, 4)");

596 │ entity.Property(e => e.XMT\_CoBot).HasMaxLength(200);

597 │ });

598 │

599 │ modelBuilder.Entity<tbl\_ChiTieuCL>(entity =>

600 │ {

601 │ entity.HasKey(e => e.Ma);

602 │

603 │ entity.ToTable("tbl\_ChiTieuCL");

604 │

605 │ entity.Property(e => e.Code)

606 │ .HasMaxLength(50)

607 │ .IsUnicode(false);

608 │ entity.Property(e => e.Ten).HasMaxLength(150);

609 │ });

610 │

611 │ modelBuilder.Entity<tbl\_ColorGioiHanSoLieu>(entity =>

612 │ {

613 │ entity

614 │ .HasNoKey()

615 │ .ToTable("tbl\_ColorGioiHanSoLieu");

616 │

617 │ entity.Property(e => e.Code).HasMaxLength(1000);

618 │ entity.Property(e => e.Ma).ValueGeneratedOnAdd();

619 │ entity.Property(e => e.MaxColor).HasMaxLength(50);

620 │ entity.Property(e => e.MinColor).HasMaxLength(50);

621 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

622 │ entity.Property(e => e.NgayTao).HasColumnType("datetime");

623 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

624 │ });

625 │

626 │ modelBuilder.Entity<tbl\_DanhMucSoLieu>(entity =>

627 │ {

628 │ entity.HasKey(e => e.Ma);

629 │

630 │ entity.ToTable("tbl\_DanhMucSoLieu");

631 │

632 │ entity.Property(e => e.Code)

633 │ .HasMaxLength(50)

634 │ .IsUnicode(false);

635 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

636 │ entity.Property(e => e.NgayTao)

637 │ .HasDefaultValueSql("(getdate())")

638 │ .HasColumnType("datetime");

639 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

640 │ entity.Property(e => e.Ten).HasMaxLength(200);

641 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

642 │ });

643 │

644 │ modelBuilder.Entity<tbl\_GioiHanSoLieu>(entity =>

645 │ {

646 │ entity.HasKey(e => e.Ma);

647 │

648 │ entity.ToTable("tbl\_GioiHanSoLieu");

649 │

650 │ entity.Property(e => e.Code).HasMaxLength(50);

651 │ entity.Property(e => e.DVT).HasMaxLength(50);

652 │ entity.Property(e => e.GhiChu).HasMaxLength(2000);

653 │ entity.Property(e => e.GiaTri).HasColumnType("decimal(18, 3)");

654 │ entity.Property(e => e.Loai).HasMaxLength(50);

655 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

656 │ entity.Property(e => e.NgayTao)

657 │ .HasDefaultValueSql("(getdate())")

658 │ .HasColumnType("datetime");

659 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

660 │ entity.Property(e => e.Ten).HasMaxLength(250);

661 │ entity.Property(e => e.TenNgan).HasMaxLength(50);

662 │ entity.Property(e => e.Ten\_ZH).HasMaxLength(250);

663 │ entity.Property(e => e.TieuChuan).HasMaxLength(100);

664 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

665 │ });

666 │

667 │ modelBuilder.Entity<tbl\_KiemTraChatLuong>(entity =>

668 │ {

669 │ entity.HasKey(e => e.Ma);

670 │

671 │ entity.ToTable("tbl\_KiemTraChatLuong");

672 │

673 │ entity.Property(e => e.ChatLuongBot).HasMaxLength(250);

674 │ entity.Property(e => e.GhiChu).HasMaxLength(250);

675 │ entity.Property(e => e.HanSuDung).HasMaxLength(200);

676 │ entity.Property(e => e.KetLuan).HasMaxLength(250);

677 │ entity.Property(e => e.MaLo).HasMaxLength(50);

678 │ entity.Property(e => e.NgayGiaoHang).HasColumnType("datetime");

679 │ entity.Property(e => e.NgaySanXuat).HasColumnType("datetime");

680 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

681 │ entity.Property(e => e.NgayTao)

682 │ .HasDefaultValueSql("(getdate())")

683 │ .HasColumnType("datetime");

684 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

685 │ entity.Property(e => e.NguoiKiemTra).HasMaxLength(100);

686 │ entity.Property(e => e.SoDonHang).HasMaxLength(50);

687 │ entity.Property(e => e.SoLuong).HasMaxLength(50);

688 │ entity.Property(e => e.TenKhachHang).HasMaxLength(250);

689 │ entity.Property(e => e.TenSanPham).HasMaxLength(250);

690 │ entity.Property(e => e.TieuChuan).HasMaxLength(50);

691 │ entity.Property(e => e.VatLieuBaoBi).HasMaxLength(50);

692 │ });

693 │

694 │ modelBuilder.Entity<tbl\_LoaiSoLieu>(entity =>

695 │ {

696 │ entity.HasKey(e => e.Ma);

697 │

698 │ entity.ToTable("tbl\_LoaiSoLieu");

699 │

700 │ entity.Property(e => e.Code)

701 │ .HasMaxLength(50)

702 │ .IsUnicode(false);

703 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

704 │ entity.Property(e => e.NgayTao)

705 │ .HasDefaultValueSql("(getdate())")

706 │ .HasColumnType("datetime");

707 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

708 │ entity.Property(e => e.Ten).HasMaxLength(50);

709 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

710 │ });

711 │

712 │ modelBuilder.Entity<tbl\_ManLed>(entity =>

713 │ {

714 │ entity.HasKey(e => e.Ma).HasName("PK\_tbl\_ManLed\_1");

715 │

716 │ entity.ToTable("tbl\_ManLed");

717 │

718 │ entity.Property(e => e.http).HasMaxLength(100);

719 │ entity.Property(e => e.ipLed).HasMaxLength(100);

720 │ });

721 │

722 │ modelBuilder.Entity<tbl\_NhomSoLieu>(entity =>

723 │ {

724 │ entity.HasKey(e => e.Ma);

725 │

726 │ entity.ToTable("tbl\_NhomSoLieu");

727 │

728 │ entity.Property(e => e.Code)

729 │ .HasMaxLength(50)

730 │ .IsUnicode(false);

731 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

732 │ entity.Property(e => e.NgayTao)

733 │ .HasDefaultValueSql("(getdate())")

734 │ .HasColumnType("datetime");

735 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

736 │ entity.Property(e => e.Ten).HasMaxLength(50);

737 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

738 │ });

739 │

740 │ modelBuilder.Entity<tbl\_SoLieu>(entity =>

741 │ {

742 │ entity.HasKey(e => e.Ma);

743 │

744 │ entity.ToTable("tbl\_SoLieu");

745 │

746 │ entity.Property(e => e.DotLo).HasMaxLength(100);

747 │ entity.Property(e => e.Ngay).HasColumnType("datetime");

748 │ entity.Property(e => e.NgaySua).HasColumnType("datetime");

749 │ entity.Property(e => e.NgayTao)

750 │ .HasDefaultValueSql("(getdate())")

751 │ .HasColumnType("datetime");

752 │ entity.Property(e => e.NgayXoa).HasColumnType("datetime");

753 │ entity.Property(e => e.QC).HasMaxLength(100);

754 │ entity.Property(e => e.ThuKhoBot).HasMaxLength(100);

755 │ entity.Property(e => e.TrangThai).HasDefaultValue(true);

756 │ entity.Property(e => e.TruongBPTP).HasMaxLength(100);

757 │ entity.Property(e => e.TruongCa).HasMaxLength(100);

758 │ });

759 │

760 │ modelBuilder.Entity<tmp>(entity =>

761 │ {

762 │ entity

763 │ .HasNoKey()

764 │ .ToTable("tmp");

765 │

766 │ entity.Property(e => e.\_12\_30\_PM\_\_5\_11)

767 │ .HasColumnType("decimal(38, 4)")

768 │ .HasColumnName("12:30 PM\r\n5/11");

769 │ entity.Property(e => e.\_13\_30\_PM\_\_5\_11)

770 │ .HasColumnType("decimal(38, 4)")

771 │ .HasColumnName("13:30 PM\r\n5/11");

772 │ entity.Property(e => e.\_14\_00\_PM\_\_5\_11)

773 │ .HasColumnType("decimal(38, 4)")

774 │ .HasColumnName("14:00 PM\r\n5/11");

775 │ entity.Property(e => e.\_14\_30\_PM\_\_5\_11)

776 │ .HasColumnType("decimal(38, 4)")

777 │ .HasColumnName("14:30 PM\r\n5/11");

778 │ entity.Property(e => e.name).HasMaxLength(128);

779 │ });

780 │

781 │ modelBuilder.Entity<tmpDatum>(entity =>

782 │ {

783 │ entity.HasNoKey();

784 │

785 │ entity.Property(e => e.B\_DoAm).HasColumnType("decimal(18, 4)");

786 │ entity.Property(e => e.B\_DoTrang).HasColumnType("decimal(18, 4)");

787 │ entity.Property(e => e.B\_TapChat).HasColumnType("decimal(18, 4)");

788 │ entity.Property(e => e.ThoiGian)

789 │ .HasMaxLength(39)

790 │ .IsUnicode(false);

791 │ });

792 │

793 │ modelBuilder.Entity<vQuyen>(entity =>

794 │ {

795 │ entity

796 │ .HasNoKey()

797 │ .ToView("vQuyen");

798 │

799 │ entity.Property(e => e.Email).HasMaxLength(50);

800 │ entity.Property(e => e.HoTen).HasMaxLength(100);

801 │ entity.Property(e => e.TenDangNhap).HasMaxLength(100);

802 │ });

803 │

804 │ OnModelCreatingPartial(modelBuilder);

805 │ }

806 │

807 │ partial void OnModelCreatingPartial(ModelBuilder modelBuilder);

808 │ }

────────────────────────────────────────────────────────────

**📄 ResponseModel.cs (CS, 51 lines)**

1 │ ﻿using Newtonsoft.Json;

2 │ using QuanLyChatLuong.Common;

3 │

4 │ namespace QuanLyChatLuong.Models.APIModels

5 │ {

6 │ public class ResponseModel

7 │ {

8 │ public string? Status { get; set; }

9 │ public string? Messenge { get; set; }

10 │ public string? Data { get; set; }

11 │ public ResponseModel() { }

12 │ public ResponseModel(string status, string messenge, Object data)

13 │ {

14 │ Status = status;

15 │ Messenge = messenge;

16 │ Data = JsonConvert.SerializeObject(data);

17 │ }

18 │ public ResponseModel(string status, string messenge)

19 │ {

20 │ Status = status;

21 │ Messenge = messenge;

22 │ Data = null;

23 │ }

24 │ }

25 │ public class ResponseModelSuccess : ResponseModel

26 │ {

27 │ public ResponseModelSuccess()

28 │ {

29 │ }

30 │

31 │ public ResponseModelSuccess(string messenge, Object? data = null)

32 │ {

33 │ Status = QLCLConst.OK;

34 │ Messenge = messenge;

35 │ Data = JsonConvert.SerializeObject(data);

36 │ }

37 │ }

38 │ public class ResponseModelError : ResponseModel

39 │ {

40 │ public ResponseModelError()

41 │ {

42 │ }

43 │

44 │ public ResponseModelError(string messenge, Object? data = null)

45 │ {

46 │ Status = QLCLConst.Error;

47 │ Messenge = messenge;

48 │ Data = JsonConvert.SerializeObject(data);

49 │ }

50 │ }

51 │ }

────────────────────────────────────────────────────────────

**📄 RoleController.cs (CS, 80 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using Microsoft.AspNetCore.Mvc.ModelBinding;

3 │ using Microsoft.EntityFrameworkCore;

4 │ using QuanLyChatLuong.Common;

5 │ using QuanLyChatLuong.Models;

6 │

7 │ namespace QuanLyChatLuong.Controllers

8 │ {

9 │ [Route("sysRole")]

10 │ public class RoleController : BaseController

11 │ {

12 │ public class RoleForm

13 │ {

14 │ public string? ID { get; set; }

15 │ public string? Code { get; set; }

16 │ public string? Name { get; set; }

17 │ public string? Description { get; set; }

18 │ public string? Status { get; set; }

19 │ }

20 │ public RoleController(QuanLyChatLuongDbContext dbcontext, MemoryCacheManager memoryCache) : base(dbcontext, memo...

21 │ {

22 │ m\_MenuID = MenuID.sysRole;

23 │ }

24 │ [HttpGet]

25 │ [TokenUserCheckHTTP()]

26 │ public IActionResult Index()

27 │ {

28 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRole, (int)Permissio...

29 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRole, (int)Permissi...

30 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRole, (int)Permis...

31 │ ViewData["bViewChild"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRole, (int)Per...

32 │ ViewData["ListData"] = new sysRoleModel(dbcontext).GetData();

33 │ return View();

34 │ }

35 │ [HttpGet("add")]

36 │ [TokenUserCheckHTTP(Permission.Add)]

37 │ public ActionResult Add()

38 │ {

39 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRole, (int)Permissio...

40 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRole, (int)Permissi...

41 │ return View("Edit");

42 │ }

43 │ [HttpGet("edit")]

44 │ [TokenUserCheckHTTP(Permission.Edit)]

45 │ public ActionResult Edit(string ID)

46 │ {

47 │ var item = new sysRoleModel(dbcontext).GetItem(Cast.ToInt(ID.Decrypt()));

48 │ if (item == null) return NotFound("Role not found");

49 │ ViewData["item"] = item;

50 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRole, (int)Permissio...

51 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRole, (int)Permissi...

52 │ return View("Edit");

53 │ }

54 │ [HttpPost("updateapi")]

55 │ [TokenUserCheckHTTP(Permission.Edit)]

56 │ public IActionResult UpdateAPI([FromBody] RoleForm model)

57 │ {

58 │ int result = new sysRoleModel(dbcontext).Edit(Cast.ToInt(model.ID), model.Code, model.Name, model.Descriptio...

59 │ return Json(Utility.MakeResponseFromEditResult(Cast.ToString(result)));

60 │ }

61 │

62 │ [HttpPost("addapi")]

63 │ [TokenUserCheckHTTP(Permission.Add)]

64 │ public IActionResult AddAPI([FromBody] RoleForm model)

65 │ {

66 │ int result = new sysRoleModel(dbcontext).Add(model.Code, model.Name, model.Description, model.Status == "1",...

67 │ return Json(Utility.MakeResponseFromAddResult(Cast.ToString(result)));

68 │ }

69 │

70 │ [HttpPost("deleteapi")]

71 │ [TokenUserCheckHTTP(Permission.Delete)]

72 │ public ActionResult DeleteAPI([FromBody] RoleForm model)

73 │ {

74 │ //

75 │ string result = Cast.ToString(new sysRoleModel(dbcontext).Delete(Cast.ToInt(model.ID.Decrypt()), objToken!.u...

76 │ return Json(Utility.MakeResponseFromDeleteResult(Cast.ToString(result)));

77 │ }

78 │

79 │ }

80 │ }

────────────────────────────────────────────────────────────

**📄 RoleMenuController.cs (CS, 88 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using QuanLyChatLuong.Common;

3 │ using QuanLyChatLuong.Models;

4 │ using System.Runtime.CompilerServices;

5 │

6 │ namespace QuanLyChatLuong.Controllers

7 │ {

8 │ [Route("sysRoleMenu")]

9 │ public class RoleMenuController : BaseController

10 │ {

11 │ public RoleMenuController(QuanLyChatLuongDbContext dbcontext, MemoryCacheManager memoryCache) : base(dbcontext, ...

12 │ {

13 │ m\_MenuID = MenuID.sysRoleMenu;

14 │ }

15 │ public class RoleMenuForm

16 │ {

17 │ public string? ID { get; set; }

18 │ public string? RoleID { get; set; }

19 │ public string? MenuID { get; set; }

20 │ public string? AllowAccess { get; set; }

21 │ public string? Status { get; set; }

22 │ }

23 │ [HttpGet("detail")]

24 │ [TokenUserCheckHTTP]

25 │ public IActionResult Detail(string RoleID)

26 │ {

27 │ ViewData["ListData"] = new sysRoleMenuModel(dbcontext).GetDataFollowRoleID(Cast.ToInt(RoleID.Decrypt()));

28 │ ViewData["ListMenu"] = dbcontext.sys\_Menus.Where(x => !(x.DaXoa ?? false)).ToList();

29 │ ViewData["RoleItem"] = new sysRoleModel(dbcontext).GetItem(RoleID.GetGuid());

30 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleMenu, (int)Permi...

31 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleMenu, (int)Perm...

32 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleMenu, (int)Pe...

33 │ ViewData["bViewChild"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleMenu, (int...

34 │ return View("Index");

35 │ }

36 │ [HttpGet("add")]

37 │ [TokenUserCheckHTTP(Permission.Add)]

38 │ public IActionResult Add(string RoleID)

39 │ {

40 │ ViewData["RoleItem"] = new sysRoleModel(dbcontext).GetItem(Cast.ToInt(RoleID));

41 │ ViewData["ListMenu"] = dbcontext.sys\_Menus.Where(x => !(x.DaXoa ?? false) && (x.TrangThai ?? false)).ToList(...

42 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleMenu, (int)Permi...

43 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleMenu, (int)Perm...

44 │

45 │ return View("Edit");

46 │ }

47 │ [HttpGet("edit")]

48 │ [TokenUserCheckHTTP(Permission.Edit)]

49 │ public IActionResult Edit(string ID)

50 │ {

51 │ sys\_RoleMenu drRM = new sysRoleMenuModel(dbcontext).GetItem(Cast.ToInt(ID.Decrypt()));

52 │ ViewData["item"] = drRM;

53 │ ViewData["ListMenu"] = dbcontext.sys\_Menus.Where(x => !(x.DaXoa ?? false) && (x.TrangThai ?? false)).ToList(...

54 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleMenu, (int)Permi...

55 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleMenu, (int)Perm...

56 │

57 │ if (drRM != null)

58 │ ViewData["RoleItem"] = new sysRoleModel(dbcontext).GetItem(drRM.MaRole);

59 │

60 │ return View("Edit");

61 │ }

62 │ [HttpPost("updateapi")]

63 │ [TokenUserCheckHTTP(Permission.Edit)]

64 │ public IActionResult UpdateAPI([FromBody] RoleMenuForm model)

65 │ {

66 │ int result = new sysRoleMenuModel(dbcontext).Edit(Cast.ToInt(model.ID), Cast.ToInt(model.MenuID), model.Allo...

67 │ memoryCache.Clear(t => t.Contains(QLCLConst.cacheKeyMenu));

68 │ return Json(Utility.MakeResponseFromEditResult(Cast.ToString(result)));

69 │

70 │ }

71 │ [HttpPost("addapi")]

72 │ [TokenUserCheckHTTP(Permission.Add)]

73 │ public IActionResult AddAPI([FromBody] RoleMenuForm model)

74 │ {

75 │ int result = new sysRoleMenuModel(dbcontext).Add(Cast.ToInt(model.RoleID), Cast.ToInt(model.MenuID), model.A...

76 │ return Json(Utility.MakeResponseFromAddResult(Cast.ToString(result)));

77 │ }

78 │

79 │ [HttpPost("deleteapi")]

80 │ [TokenUserCheckHTTP(Permission.Delete)]

81 │ public IActionResult DeleteAPI([FromBody] RoleMenuForm model)

82 │ {

83 │ int result = new sysRoleMenuModel(dbcontext).Delete(Cast.ToInt(model.ID.Decrypt()), objToken!.userid);

84 │ memoryCache.Clear(t => t.Contains(QLCLConst.cacheKeyMenu));

85 │ return Json(Utility.MakeResponseFromDeleteResult(Cast.ToString(result)));

86 │ }

87 │ }

88 │ }

────────────────────────────────────────────────────────────

**📄 RolePermissionController.cs (CS, 141 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using QuanLyChatLuong.Common;

3 │ using QuanLyChatLuong.Models;

4 │ using QuanLyChatLuong.Models.APIModels;

5 │ using System.Runtime.CompilerServices;

6 │

7 │ namespace QuanLyChatLuong.Controllers

8 │ {

9 │ [Route("sysRolePermission")]

10 │ public class RolePermissionController : BaseController

11 │ {

12 │ public RolePermissionController(QuanLyChatLuongDbContext dbcontext, MemoryCacheManager memoryCache) : base(dbcon...

13 │ {

14 │ m\_MenuID = MenuID.sysRoleQuyen;

15 │ }

16 │ public class RolePermissionForm

17 │ {

18 │ public string? ID { get; set; }

19 │ public string? RoleID { get; set; }

20 │ public string? MenuID { get; set; }

21 │ public string? PermissionID { get; set; }

22 │ public string? Status { get; set; }

23 │ }

24 │ public class QuicklyPermissionForm

25 │ {

26 │ public string? RoleID { get; set; }

27 │ public string? RMAllowAccess { get; set; }

28 │ public string? RMStatus { get; set; }

29 │ public string? RPStatus { get; set; }

30 │ }

31 │ [HttpGet("detail")]

32 │ [TokenUserCheckHTTP]

33 │ public IActionResult Detail(string RoleMenuID)

34 │ {

35 │ sys\_RoleMenu drRM = new sysRoleMenuModel(dbcontext).GetItem(Cast.ToInt(RoleMenuID.Decrypt()));

36 │ ViewData["ListData"] = new sysRolePermissionModel(dbcontext).GetDataFollowRoleMenuID(drRM.MaMenu, drRM.MaRol...

37 │ ViewData["ListPermission"] = new sysQuyenModel(dbcontext).GetData();

38 │ ViewData["RoleMenuItem"] = drRM;

39 │

40 │ if (drRM != null)

41 │ {

42 │ ViewData["RoleItem"] = new sysRoleModel(dbcontext).GetItem(drRM.MaRole);

43 │ ViewData["MenuItem"] = new sysMenuModel(dbcontext).GetItem(drRM.MaMenu);

44 │ }

45 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleQuyen, (int)Perm...

46 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleQuyen, (int)Per...

47 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleQuyen, (int)P...

48 │ return View("index");

49 │ }

50 │

51 │ [HttpGet("add")]

52 │ [TokenUserCheckHTTP(Permission.Add)]

53 │ public ActionResult Add(string RoleMenuID)

54 │ {

55 │ sys\_RoleMenu drRM = new sysRoleMenuModel(dbcontext).GetItem(Cast.ToInt(RoleMenuID));

56 │ ViewData["RoleMenuItem"] = drRM;

57 │ ViewData["ListPermission"] = new sysQuyenModel(dbcontext).GetData().Where(t => (t.TrangThai ?? false)).ToLis...

58 │ if (drRM != null)

59 │ {

60 │ ViewData["RoleItem"] = new sysRoleModel(dbcontext).GetItem(drRM.MaRole);

61 │ ViewData["MenuItem"] = new sysMenuModel(dbcontext).GetItem(drRM.MaMenu);

62 │ }

63 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleQuyen, (int)Perm...

64 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleQuyen, (int)Per...

65 │ return View("Edit");

66 │ }

67 │

68 │ [HttpGet("edit")]

69 │ [TokenUserCheckHTTP(Permission.Edit)]

70 │ public ActionResult Edit(string ID)

71 │ {

72 │ sys\_RoleQuyen drRP = new sysRolePermissionModel(dbcontext).GetItem(Cast.ToInt(ID.Decrypt()));

73 │ ViewData["item"] = drRP;

74 │ ViewData["ListPermission"] = new sysQuyenModel(dbcontext).GetData().Where(t => (t.TrangThai ?? false)).ToLis...

75 │

76 │ if (drRP != null)

77 │ {

78 │ sys\_RoleMenu drRM = new sysRoleMenuModel(dbcontext).GetItem(drRP.MaRole, drRP.MaMenu);

79 │ ViewData["RoleMenuItem"] = drRM;

80 │ if (drRM != null)

81 │ {

82 │ ViewData["RoleItem"] = new sysRoleModel(dbcontext).GetItem(drRM.MaRole);

83 │ ViewData["MenuItem"] = new sysMenuModel(dbcontext).GetItem(drRM.MaMenu);

84 │ }

85 │ }

86 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleQuyen, (int)Perm...

87 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleQuyen, (int)Per...

88 │ return View("Edit");

89 │ }

90 │

91 │ [HttpPost("delete")]

92 │ [TokenUserCheckHTTP(Permission.Delete)]

93 │ public ActionResult Delete([FromBody] RolePermissionForm model)

94 │ {

95 │ string result = Cast.ToString(new sysRolePermissionModel(dbcontext).Delete(Cast.ToInt(model.ID.Decrypt()), o...

96 │ memoryCache.Clear(t => t.Contains(QLCLConst.cacheKeyMenu));

97 │ return Json(new ResponseModel(QLCLConst.OK, result == "1" ? "Thành công" : "Đã xảy ra lỗi " + result...

98 │ }

99 │ [HttpPost("addapi")]

100 │ [TokenUserCheckHTTP(Permission.Add)]

101 │ public ActionResult AddAPI([FromBody] RolePermissionForm model)

102 │ {

103 │ int result = new sysRolePermissionModel(dbcontext).Add(Cast.ToInt(model.RoleID), Cast.ToInt(model.MenuID), C...

104 │ memoryCache.Clear(t => t.Contains(QLCLConst.cacheKeyMenu));

105 │ return Json(new ResponseModel(QLCLConst.OK, result == 1 ? "Thành công" : "Đã xảy ra lỗi " + result))...

106 │ }

107 │

108 │ [HttpPost("updateapi")]

109 │ [TokenUserCheckHTTP(Permission.Edit)]

110 │ public ActionResult UpdateAPI([FromBody] RolePermissionForm model)

111 │ {

112 │ int result = new sysRolePermissionModel(dbcontext).Edit(Cast.ToInt(model.ID), Cast.ToInt(model.PermissionID)...

113 │ memoryCache.Clear(t => t.Contains(QLCLConst.cacheKeyMenu));

114 │ return Json(new ResponseModel(QLCLConst.OK, result == 1 ? "Thành công" : "Đã xảy ra lỗi " + result))...

115 │ }

116 │ [HttpGet("QuicklyPermission")]

117 │ [TokenUserCheckHTTP]

118 │ public ActionResult QuicklyPermission(string RoleID)

119 │ {

120 │ var item = new sysRoleModel(dbcontext).GetItem(Cast.ToInt(RoleID.Decrypt()));

121 │ if (item == null) return NotFound("RolePermission not found");

122 │ ViewData["RoleItem"] = item;

123 │ ViewData["ListRoleMenu"] = new sysRoleMenuModel(dbcontext).GetRoleMenuFollowRoleID(Cast.ToInt(RoleID.Decrypt...

124 │ ViewData["ListRolePermisson"] = new sysRolePermissionModel(dbcontext).GetRolePermissionFollowRoleID(Cast.ToI...

125 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleQuyen, (int)Perm...

126 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleQuyen, (int)Per...

127 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysRoleQuyen, (int)P...

128 │ return View("QuicklyPermission");

129 │ }

130 │

131 │ [HttpPost("UpdateQuicklyPermission")]

132 │ [TokenUserCheckHTTP(Permission.Edit)]

133 │ public ActionResult UpdateQuicklyPermission([FromBody] QuicklyPermissionForm model)

134 │ {

135 │ bool kq = new sysRolePermissionModel(dbcontext).UpdateQuicklyPermission(Cast.ToInt(model.RoleID), model.RMAl...

136 │ memoryCache.Clear(t => t.Contains(QLCLConst.cacheKeyMenu));

137 │ return Json(new ResponseModel(QLCLConst.OK, kq ? "Thành công" : "Đã xảy ra lỗi "));

138 │ }

139 │ }

140 │

141 │ }

────────────────────────────────────────────────────────────

**📄 RptBaoCaoDoanhThu.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class RptBaoCaoDoanhThu

7 │ {

8 │ public long Ma { get; set; }

9 │

10 │ public int? Nam { get; set; }

11 │

12 │ public int? Thang { get; set; }

13 │

14 │ public int? MaKh { get; set; }

15 │

16 │ public int? MaHangHoa { get; set; }

17 │

18 │ public int? MaLoaiBao { get; set; }

19 │

20 │ public decimal? GiaTri { get; set; }

21 │

22 │ public int? SoLuong { get; set; }

23 │

24 │ public decimal? DonGiaTb { get; set; }

25 │

26 │ public decimal? DonGiaNytb { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 SanPhamController.cs (CS, 6 lines)**

1 │ ﻿namespace QuanLyChatLuong.Controllers

2 │ {

3 │ public class SanPhamController

4 │ {

5 │ }

6 │ }

────────────────────────────────────────────────────────────

**📄 SoLieuController.cs (CS, 983 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using Microsoft.EntityFrameworkCore;

3 │ using QuanLyChatLuong.Common;

4 │ using QuanLyChatLuong.Models;

5 │ using QuanLyChatLuong.Models.APIModels;

6 │ using QuanLyChatLuong.Models.CRMEntity; // Assuming CloseChotLo, SysBranch are here

7 │ using System.Data; // For DataTable

8 │ using System.Data.SqlTypes;

9 │ using System.Diagnostics;

10 │

11 │ namespace QuanLyChatLuong.Controllers

12 │ {

13 │ [Route("catSoLieu")]

14 │ public class SoLieuController : BaseController

15 │ {

16 │ // Các DTOs (ReportFilterModel, SoLieuInputDto, SoLieuChiTietInputDto, SoLieuBanThanhPhamInputDto)

17 │ // giữ nguyên như phiên bản trước của bạn (SoLieuControllerRefactored\_CS\_v3\_Full)

18 │ #region DTOs and Models

19 │ public class ReportFilterModel

20 │ {

21 │ public DateTime FromDate { get; set; }

22 │ public DateTime ToDate { get; set; }

23 │ public int MaSoLieu { get; set; }

24 │ }

25 │

26 │ public class SoLieuInputDto

27 │ {

28 │ public int Ma { get; set; }

29 │ public string? TruongCa { get; set; }

30 │ public string? QC { get; set; }

31 │ public string? ThuKhoBot { get; set; }

32 │ public string? TruongBan { get; set; }

33 │ public string? DotLo { get; set; }

34 │ }

35 │

36 │ public class SoLieuChiTietInputDto

37 │ {

38 │ public int Ma { get; set; }

39 │ public int MaSoLieu { get; set; }

40 │ public string? Gio { get; set; }

41 │ public string? Phut { get; set; }

42 │ public int MaLo { get; set; } // Đây là Ma (PK) của CloseChotLo

43 │ public decimal? DoTrang { get; set; }

44 │ public decimal? DoAm { get; set; }

45 │ public decimal? DoMin { get; set; }

46 │ public decimal? Ph { get; set; }

47 │ public decimal? DoKeoCP { get; set; }

48 │ public decimal? DoKeoBU { get; set; }

49 │ public decimal? TapChat { get; set; }

50 │ public decimal? NhietDoLo { get; set; }

51 │ public decimal? NhietDoSay { get; set; }

52 │ public decimal? NhietDoHoHoa { get; set; }

53 │ public decimal? PHNuocSach { get; set; }

54 │ public decimal? DoAmBU { get; set; }

55 │ public string? Cepa { get; set; }

56 │ public decimal? DoAmXMS { get; set; }

57 │ public string? XacMiTuoiCoBot { get; set; }

58 │ public decimal? So2 { get; set; }

59 │ public decimal? HamLuongTB { get; set; }

60 │ public string? GhiChu { get; set; }

61 │ }

62 │

63 │ public class SoLieuBanThanhPhamInputDto

64 │ {

65 │ public int Ma { get; set; }

66 │ public int MaSoLieu { get; set; }

67 │ public string? Gio { get; set; }

68 │ public string? Phut { get; set; }

69 │ public int MaLo { get; set; } // Đây là Ma (PK) của CloseChotLo

70 │ public decimal? PhRO { get; set; }

71 │ public decimal? HCL { get; set; }

72 │ public decimal? NaOH { get; set; }

73 │ public decimal? HCL\_Loang { get; set; }

74 │ public decimal? NaOH\_Loang { get; set; }

75 │ public decimal? SodiumCarbonat { get; set; }

76 │ public decimal? RasperChao { get; set; }

77 │ public string? HamLuongTinhBot { get; set; }

78 │ public string? PHSuaBot { get; set; }

79 │ public decimal? NhietDoHoHoa { get; set; }

80 │ public decimal? DoNhot { get; set; }

81 │ public string? GhiChu { get; set; }

82 │ public string? NguoiPhanTich { get; set; }

83 │ }

84 │ #endregion

85 │

86 │ private readonly CrmContext \_crmContext;

87 │ private readonly IWebHostEnvironment \_hostingEnvironment;

88 │

89 │ public SoLieuController(QuanLyChatLuongDbContext dbContext, CrmContext crmContext, IWebHostEnvironment hostingEn...

90 │ : base(dbContext, memoryCache)

91 │ {

92 │ \_crmContext = crmContext;

93 │ \_hostingEnvironment = hostingEnvironment;

94 │ m\_MenuID = MenuID.catSoLieu;

95 │ }

96 │

97 │ [TokenUserCheckHTTP]

98 │ public IActionResult Index()

99 │ {

100 │ return View();

101 │ }

102 │

103 │ #region Data Retrieval Actions

104 │

105 │ [TokenUserCheckHTTP]

106 │ [HttpPost("getDataSoLieuChinh")]

107 │ public IActionResult GetDataSoLieuChinh([FromBody] ReportFilterModel filter)

108 │ {

109 │ if (!IsValidDateRange(filter.FromDate, filter.ToDate))

110 │ return Json(new ResponseModel(QLCLConst.Error, "Ngày không hợp lệ.", null));

111 │

112 │ var data = dbcontext.tbl\_SoLieus

113 │ .Where(sl => sl.Ngay.Date >= filter.FromDate.Date && sl.Ngay.Date <= filter.ToDate.Date &&

114 │ sl.MaChiNhanh == objToken.branch && !(sl.DaXoa ?? false))

115 │ .Select(sl => new

116 │ {

117 │ ma = sl.Ma,

118 │ ngaynhap = sl.Ngay,

119 │ truongca = sl.TruongCa,

120 │ qc = sl.QC,

121 │ thukhobot = sl.ThuKhoBot,

122 │ truongbptp = sl.TruongBPTP,

123 │ dotlo = sl.DotLo,

124 │ trangthai = sl.TrangThai

125 │ })

126 │ .OrderByDescending(x => x.ngaynhap)

127 │ .ToList();

128 │ // For reports 1 & 2, this endpoint returns only main data. Detail data is fetched by a separate call.

129 │ return Json(new ResponseModel(QLCLConst.OK, "Lấy dữ liệu thành công", data));

130 │ }

131 │

132 │ [TokenUserCheckHTTP]

133 │ [HttpPost("getDataSoLieuChiTiet")]

134 │ public IActionResult GetDataSoLieuChiTiet([FromBody] ReportFilterModel filter)

135 │ {

136 │ if (filter.MaSoLieu <= 0)

137 │ return Json(new ResponseModel(QLCLConst.Error, "Mã số liệu không hợp lệ.", null));

138 │

139 │ var loThuMuaList = GetActiveLoThuMuaList();

140 │

141 │ var chiTietSoLieuData = dbcontext.tbl\_ChiTietSoLieus

142 │ .Where(ct => ct.MaSoLieu == filter.MaSoLieu && !(ct.DaXoa ?? false))

143 │ .Select(ct => new

144 │ {

145 │ ct.Ma,

146 │ ct.MaSoLieu,

147 │ ct.Gio,

148 │ ct.Phut,

149 │ ct.MaLoThuMua,

150 │ ct.B\_DoTrang,

151 │ ct.B\_DoAm,

152 │ ct.B\_DoMin,

153 │ ct.B\_DoPH,

154 │ ct.B\_DoKeoCP,

155 │ ct.B\_DoKeoBU,

156 │ ct.B\_TapChat,

157 │ ct.ND\_Lo,

158 │ ct.ND\_Say,

159 │ ct.ND\_HoHoa,

160 │ ct.NS\_DoPH,

161 │ ct.BU\_DoAm,

162 │ ct.Cepa,

163 │ ct.XMS\_DoAm,

164 │ ct.XMT\_CoBot,

165 │ ct.SO2,

166 │ ct.HamLuongTB,

167 │ ct.GhiChu

168 │ })

169 │ .ToList();

170 │

171 │ var data = chiTietSoLieuData

172 │ .GroupJoin(loThuMuaList, chiTiet => chiTiet.MaLoThuMua, lo => lo.Ma, (chiTiet, loGroup) => new { chiTiet, loGroup })

173 │ .SelectMany(x => x.loGroup.DefaultIfEmpty(), (x, lo) => new

174 │ {

175 │ ma = x.chiTiet.Ma,

176 │ maSoLieu = x.chiTiet.MaSoLieu,

177 │ gio = x.chiTiet.Gio,

178 │ phut = x.chiTiet.Phut,

179 │ maLo = lo?.MaLoDisplay,

180 │

181 │ doTrang = ToNullableDecimal(x.chiTiet.B\_DoTrang, "0.0"), // 1 số lẻ

182 │ doAm = ToNullableDecimal(x.chiTiet.B\_DoAm, "0.0"), // 1 số lẻ

183 │ doMin = ToNullableDecimal(x.chiTiet.B\_DoMin, "0.00"), // 2 số lẻ

184 │ ph = ToNullableDecimal(x.chiTiet.B\_DoPH, "0.00"), // 2 số lẻ

185 │ doKeoCP = ToNullableDecimal(x.chiTiet.B\_DoKeoCP, "0"), // không thập phân

186 │ doKeoBU = ToNullableDecimal(x.chiTiet.B\_DoKeoBU, "0"), // không thập phân

187 │ tapChat = ToNullableDecimal(x.chiTiet.B\_TapChat, "0.000"), // 3 số lẻ

188 │ nhietDoLo = ToNullableDecimal(x.chiTiet.ND\_Lo, "0"), // không thập phân

189 │ nhietDoSay = ToNullableDecimal(x.chiTiet.ND\_Say, "0"), // không thập phân

190 │ nhietDoHoHoa = ToNullableDecimal(x.chiTiet.ND\_HoHoa, "0"), // không thập phân

191 │ pHNuocSach = ToNullableDecimal(x.chiTiet.NS\_DoPH, "0.00"), // 2 số lẻ

192 │ doAmBU = ToNullableDecimal(x.chiTiet.BU\_DoAm, "0.0"), // 1 số lẻ

193 │ cepa = x.chiTiet.Cepa,

194 │ doAmXMS = ToNullableDecimal(x.chiTiet.XMS\_DoAm, "0.0"), // 1 số lẻ

195 │ xacMiTuoiCoBot = x.chiTiet.XMT\_CoBot,

196 │ so2 = ToNullableDecimal(x.chiTiet.SO2, "0.00"), // 2 số lẻ

197 │ hamLuongTB = ToNullableDecimal(x.chiTiet.HamLuongTB, "0.000"), // 3 số lẻ

198 │ ghiChu = x.chiTiet.GhiChu

199 │ })

200 │ .OrderBy(x => Convert.ToInt32(x.gio)).ThenBy(x => Convert.ToInt32(x.phut))

201 │ .ToList();

202 │

203 │ // This endpoint returns detail data for a specific MaSoLieu (for report type 1)

204 │ return Json(new ResponseModel(QLCLConst.OK, "Lấy dữ liệu chi tiết thành công", data));

205 │ }

206 │

207 │ [TokenUserCheckHTTP]

208 │ [HttpPost("getDataSoLieuBanThanhPham")]

209 │ public IActionResult GetDataSoLieuBanThanhPham([FromBody] ReportFilterModel filter)

210 │ {

211 │ if (filter.MaSoLieu <= 0)

212 │ return Json(new ResponseModel(QLCLConst.Error, "Mã số liệu không hợp lệ.", null));

213 │

214 │ var loThuMuaList = GetActiveLoThuMuaList();

215 │

216 │ var banThanhPhamData = dbcontext.tbl\_BanThanhPhams

217 │ .Where(btp => btp.MaSoLieu == filter.MaSoLieu && !(btp.DaXoa ?? false))

218 │ .Select(btp => new

219 │ {

220 │ btp.Ma,

221 │ btp.MaSoLieu,

222 │ btp.Gio,

223 │ btp.Phut,

224 │ btp.MaLo,

225 │ btp.pH\_RO,

226 │ btp.HCL,

227 │ btp.NaOH,

228 │ btp.HCL\_Loang,

229 │ btp.NaOH\_Loang,

230 │ btp.CQ\_SodiumCarbonat,

231 │ btp.RasperChao,

232 │ btp.HamLuongTBTrongBaMi,

233 │ btp.pHSuaBot,

234 │ btp.ND\_HoHoa,

235 │ btp.DoNhot,

236 │ btp.GhiChu,

237 │ btp.NguoiPhanTich

238 │ })

239 │ .ToList();

240 │

241 │ var data = banThanhPhamData

242 │ .GroupJoin(loThuMuaList, btp => btp.MaLo, lo => lo.Ma, (btp, loGroup) => new { btp, loGroup })

243 │ .SelectMany(x => x.loGroup.DefaultIfEmpty(), (x, lo) => new

244 │ {

245 │ ma = x.btp.Ma,

246 │ maSoLieu = x.btp.MaSoLieu,

247 │ gio = x.btp.Gio,

248 │ phut = x.btp.Phut,

249 │ maLo = lo?.MaLoDisplay,

250 │ phRO = x.btp.pH\_RO,

251 │ HCL = x.btp.HCL,

252 │ NaOH = x.btp.NaOH,

253 │ HCL\_Loang = x.btp.HCL\_Loang,

254 │ NaOH\_Loang = x.btp.NaOH\_Loang,

255 │ sodiumCarbonat = x.btp.CQ\_SodiumCarbonat,

256 │ rasperChao = x.btp.RasperChao,

257 │ hamLuongTinhBot = x.btp.HamLuongTBTrongBaMi,

258 │ pHSuaBot = x.btp.pHSuaBot,

259 │ nhietDoHoHoa = x.btp.ND\_HoHoa,

260 │ doNhot = x.btp.DoNhot,

261 │ ghiChu = x.btp.GhiChu,

262 │ nguoiPhanTich = x.btp.NguoiPhanTich

263 │ })

264 │ .OrderBy(x => Convert.ToInt32(x.gio)).ThenBy(x => Convert.ToInt32(x.phut))

265 │ .ToList();

266 │ // This endpoint returns detail data for a specific MaSoLieu (for report type 2)

267 │ return Json(new ResponseModel(QLCLConst.OK, "Lấy dữ liệu bán thành phẩm thành công", data));

268 │ }

269 │

270 │ [TokenUserCheckHTTP]

271 │ [HttpPost("getDataBotThuongThanhPham")]

272 │ public IActionResult GetDataBotThuongThanhPham([FromBody] ReportFilterModel filter)

273 │ {

274 │ var listDataMain = GetBotThuongThanhPhamReportData(filter);

275 │ var listDataDetail = new List<object>(); // For report type 3, detail is empty

276 │ return Json(new ResponseModel(QLCLConst.OK, "Lấy dữ liệu thành công", new { dataMain = listDataMain,...

277 │ }

278 │

279 │ [TokenUserCheckHTTP]

280 │ [HttpPost("getDataBotThuongBanThanhPham")]

281 │ public IActionResult GetDataBotThuongBanThanhPham([FromBody] ReportFilterModel filter)

282 │ {

283 │ var listDataMain = GetBotThuongBanThanhPhamReportData(filter);

284 │ var listDataDetail = new List<object>(); // For report type 4, detail is empty

285 │ return Json(new ResponseModel(QLCLConst.OK, "Lấy dữ liệu thành công", new { dataMain = listDataMain,...

286 │ }

287 │

288 │ #endregion

289 │

290 │ #region Excel Export Actions

291 │

292 │ [TokenUserCheckHTTP]

293 │ [HttpPost("exportBotThuongThanhPhamToExcel")]

294 │ public IActionResult ExportBotThuongThanhPhamToExcel([FromBody] ReportFilterModel filter)

295 │ {

296 │ DateTime dtpTuNgay = ValidateSqlDateTime(filter.FromDate);

297 │ DateTime dtpDenNgay = ValidateSqlDateTime(filter.ToDate);

298 │ var reportData = GetBotThuongThanhPhamReportData(new ReportFilterModel { FromDate = dtpTuNgay, ToDate = dtpD...

299 │

300 │ if (!reportData.Any())

301 │ return Json(new ResponseModel("INFO", "Không có dữ liệu để xuất.", null));

302 │

303 │ System.Data.DataTable dt = new System.Data.DataTable("BotThuongThanhPham");

304 │ dt.Columns.Add("NGÀY");

305 │ dt.Columns.Add("GIỜ", typeof(int));

306 │ dt.Columns.Add("PHÚT", typeof(int));

307 │ dt.Columns.Add("Trưởng ca");

308 │ dt.Columns.Add("QC");

309 │ dt.Columns.Add("Thủ kho");

310 │ dt.Columns.Add("Đốt lò");

311 │ //dt.Columns.Add("MÃ LÔ");

312 │

313 │ dt.Columns.Add("ĐỘ TRẮNG", typeof(decimal));

314 │ dt.Columns.Add("ĐỘ ẨM", typeof(decimal));

315 │ dt.Columns.Add("ĐỘ MỊN", typeof(decimal));

316 │ dt.Columns.Add("PH", typeof(decimal));

317 │

318 │ dt.Columns.Add("ĐỘ KEO CP", typeof(decimal));

319 │ dt.Columns.Add("ĐỘ KEO BU", typeof(decimal));

320 │ dt.Columns.Add("NHIỆT ĐỘ HỒ HOÁ", typeof(decimal));

321 │ dt.Columns.Add("TẠP CHẤT", typeof(decimal));

322 │

323 │ dt.Columns.Add("HÀM LƯỢNG TINH BỘT", typeof(decimal));

324 │ dt.Columns.Add("PH NƯỚC SẠCH", typeof(decimal));

325 │ dt.Columns.Add("SO2", typeof(decimal));

326 │ dt.Columns.Add("GHI CHÚ");

327 │

328 │

329 │ foreach (var item in reportData)

330 │ {

331 │ dt.Rows.Add(

332 │ string.Format("{0:dd/MM/yyyy}", item.ngay),

333 │ item.gio,

334 │ item.phut,

335 │ item.truongca,

336 │ item.qc,

337 │ item.thukhobot,

338 │ item.dotlo,

339 │ //item.maLo,

340 │ item.doTrang,

341 │ item.doAm,

342 │ item.doMin,

343 │ item.ph,

344 │ item.doKeoCP,

345 │ item.doKeoBU,

346 │ item.nhietDoHoHoa,

347 │ item.tapChat,

348 │ item.hamLuongTinhBot,

349 │ item.pHNuocSach,

350 │ item.sO2,

351 │ item.ghiChu

352 │ );

353 │ }

354 │ return CreateExcelFile(dt, "BaoCaoThanhPhamBotThuong.xlsx", "BaoCaoThanhPhamBotThuong.xlsx", $"Từ ngày {d...

355 │ }

356 │

357 │ [TokenUserCheckHTTP]

358 │ [HttpPost("exportBotThuongBanThanhPhamToExcel")]

359 │ public IActionResult ExportBotThuongBanThanhPhamToExcel([FromBody] ReportFilterModel filter)

360 │ {

361 │ DateTime dtpTuNgay = ValidateSqlDateTime(filter.FromDate);

362 │ DateTime dtpDenNgay = ValidateSqlDateTime(filter.ToDate);

363 │ var reportData = GetBotThuongBanThanhPhamReportData(new ReportFilterModel { FromDate = dtpTuNgay, ToDate = d...

364 │

365 │ if (!reportData.Any())

366 │ return Json(new ResponseModel("INFO", "Không có dữ liệu để xuất.", null));

367 │

368 │ System.Data.DataTable dt = new System.Data.DataTable("BotThuongBanThanhPham");

369 │ dt.Columns.Add("NGÀY");

370 │ dt.Columns.Add("GIỜ");

371 │ dt.Columns.Add("PHÚT");

372 │ dt.Columns.Add("MÃ LÔ");

373 │

374 │ dt.Columns.Add("PH/RO", typeof(decimal));

375 │ dt.Columns.Add("HCL", typeof(decimal));

376 │ dt.Columns.Add("NAOH", typeof(decimal));

377 │ dt.Columns.Add("HCL LOÃNG", typeof(decimal));

378 │

379 │ dt.Columns.Add("NAOH LOÃNG", typeof(decimal));

380 │ dt.Columns.Add("CẢM QUAN SODIUM CARBONAT", typeof(decimal));

381 │ dt.Columns.Add("RASPER", typeof(decimal));

382 │

383 │ dt.Columns.Add("HÀM LƯỢNG TINH BỘT", typeof(decimal));

384 │ dt.Columns.Add("PH SỮA BỘT", typeof(decimal));

385 │ dt.Columns.Add("NHIỆT ĐỘ HỒ HOÁ", typeof(decimal));

386 │

387 │ dt.Columns.Add("ĐỘ NHỚT", typeof(decimal));

388 │ dt.Columns.Add("GHI CHÚ");

389 │ dt.Columns.Add("NGƯỜI PHÂN TÍCH");

390 │

391 │ foreach (var item in reportData)

392 │ {

393 │ dt.Rows.Add(

394 │ item.ngay.ToString("dd/MM/yyyy"),

395 │ item.gio,

396 │ item.phut,

397 │ item.maLo,

398 │

399 │ item.phRO,

400 │ item.HCL,

401 │ item.NaOH,

402 │

403 │ item.HCL\_Loang,

404 │ item.NaOH\_Loang,

405 │ item.sodiumCarbonat,

406 │

407 │ item.rasperChao,

408 │ item.hamLuongTinhBot,

409 │ item.pHSuaBot,

410 │

411 │ item.nhietDoHoHoa,

412 │ item.doNhot,

413 │

414 │ item.ghiChu,

415 │ item.nguoiPhanTich

416 │ );

417 │ }

418 │ return CreateExcelFile(dt, "BaoCaoBotThuongBanThanhPham.xlsx", "BaoCaoBotThuongBanThanhPham.xlsx", $"Từ ng...

419 │ }

420 │

421 │ #endregion

422 │

423 │ #region CRUD Actions

424 │ [TokenUserCheckHTTP]

425 │ [HttpPost("addSoLieu")]

426 │ public IActionResult AddSoLieu([FromBody] SoLieuInputDto dto)

427 │ {

428 │ if (!ModelState.IsValid) return Json(new ResponseModelError("Dữ liệu không hợp lệ."));

429 │ try

430 │ {

431 │ var newSoLieu = new tbl\_SoLieu

432 │ {

433 │ MaChiNhanh = objToken.branch,

434 │ Ngay = DateTime.Now,

435 │ TruongCa = dto.TruongCa,

436 │ QC = dto.QC,

437 │ ThuKhoBot = dto.ThuKhoBot,

438 │ TruongBPTP = dto.TruongBan,

439 │ DotLo = dto.DotLo,

440 │ TrangThai = true,

441 │ NguoiTao = objToken.userid,

442 │ NgayTao = DateTime.Now,

443 │ DaXoa = false

444 │ };

445 │ dbcontext.tbl\_SoLieus.Add(newSoLieu);

446 │ dbcontext.SaveChanges();

447 │ return Json(new ResponseModelSuccess("Đã thêm số liệu thành công.", new { newId = newSoLieu.Ma ...

448 │ }

449 │ catch (Exception ex)

450 │ {

451 │ Debug.WriteLine($"Lỗi khi thêm số liệu: {ex.Message}");

452 │ return Json(new ResponseModelError($"Lỗi server: {ex.InnerException?.Message ?? ex.Message}"));

453 │ }

454 │ }

455 │

456 │ [TokenUserCheckHTTP]

457 │ [HttpPost("updateSoLieu")]

458 │ public IActionResult UpdateSoLieu([FromBody] SoLieuInputDto dto)

459 │ {

460 │ if (!ModelState.IsValid) return Json(new ResponseModelError("Dữ liệu không hợp lệ."));

461 │ try

462 │ {

463 │ var sl = dbcontext.tbl\_SoLieus.FirstOrDefault(x => x.Ma == dto.Ma && !(x.DaXoa ?? false) && x.MaChiNhanh...

464 │ if (sl != null)

465 │ {

466 │ sl.TruongCa = dto.TruongCa; sl.QC = dto.QC; sl.ThuKhoBot = dto.ThuKhoBot;

467 │ sl.TruongBPTP = dto.TruongBan; sl.DotLo = dto.DotLo;

468 │ sl.NguoiSua = objToken.userid; sl.NgaySua = DateTime.Now;

469 │ dbcontext.SaveChanges();

470 │ return Json(new ResponseModelSuccess("Đã cập nhật số liệu."));

471 │ }

472 │ return Json(new ResponseModelError("Không tìm thấy số liệu hoặc bạn không có quyền sửa...

473 │ }

474 │ catch (Exception ex)

475 │ {

476 │ Debug.WriteLine($"Lỗi khi cập nhật số liệu: {ex.Message}");

477 │ return Json(new ResponseModelError($"Lỗi server: {ex.InnerException?.Message ?? ex.Message}"));

478 │ }

479 │ }

480 │

481 │ [TokenUserCheckHTTP]

482 │ [HttpPost("deleteSoLieu")]

483 │ public IActionResult DeleteSoLieu([FromBody] SoLieuInputDto dto)

484 │ {

485 │ try

486 │ {

487 │ var sl = dbcontext.tbl\_SoLieus.FirstOrDefault(x => x.Ma == dto.Ma && !(x.DaXoa ?? false) && x.MaChiNhanh...

488 │ if (sl != null)

489 │ {

490 │ sl.DaXoa = true; sl.NgayXoa = DateTime.Now; sl.NguoiXoa = objToken.userid;

491 │ var chiTietSoLieu = dbcontext.tbl\_ChiTietSoLieus.Where(ct => ct.MaSoLieu == sl.Ma && !(ct.DaXoa ?? f...

492 │ foreach (var ct in chiTietSoLieu) { ct.DaXoa = true; ct.NgayXoa = DateTime.Now; ct.NguoiXoa = objTok...

493 │ var banThanhPham = dbcontext.tbl\_BanThanhPhams.Where(btp => btp.MaSoLieu == sl.Ma && !(btp.DaXoa ?? ...

494 │ foreach (var btp in banThanhPham) { btp.DaXoa = true; btp.DeletedDate = DateTime.Now; btp.UserDelete...

495 │ dbcontext.SaveChanges();

496 │ return Json(new ResponseModelSuccess("Đã xóa số liệu và các chi tiết liên quan."));

497 │ }

498 │ return Json(new ResponseModelError("Không tìm thấy số liệu hoặc đã được xóa."));

499 │ }

500 │ catch (Exception ex)

501 │ {

502 │ Debug.WriteLine($"Lỗi khi xóa số liệu: {ex.Message}");

503 │ return Json(new ResponseModelError($"Lỗi server: {ex.InnerException?.Message ?? ex.Message}"));

504 │ }

505 │ }

506 │

507 │ [TokenUserCheckHTTP]

508 │ [HttpPost("addSoLieuChiTiet")]

509 │ public IActionResult AddSoLieuChiTiet([FromBody] SoLieuChiTietInputDto dto)

510 │ {

511 │ if (!ModelState.IsValid) return Json(new ResponseModelError("Dữ liệu không hợp lệ."));

512 │ try

513 │ {

514 │ bool exists = dbcontext.tbl\_ChiTietSoLieus.Any(x => x.MaSoLieu == dto.MaSoLieu && x.Gio == dto.Gio && x....

515 │ if (exists) return Json(new ResponseModelError("Đã tồn tại số liệu chi tiết cho mốc thời...

516 │

517 │ var newChiTiet = new tbl\_ChiTietSoLieu

518 │ {

519 │ MaSoLieu = dto.MaSoLieu,

520 │ Gio = dto.Gio,

521 │ Phut = dto.Phut,

522 │ MaLoThuMua = dto.MaLo,

523 │ B\_DoTrang = dto.DoTrang,

524 │ B\_DoAm = dto.DoAm,

525 │ B\_DoMin = dto.DoMin,

526 │ B\_DoPH = dto.Ph,

527 │ B\_DoKeoCP = dto.DoKeoCP,

528 │ B\_DoKeoBU = dto.DoKeoBU,

529 │ B\_TapChat = dto.TapChat,

530 │ ND\_Lo = dto.NhietDoLo,

531 │ ND\_Say = dto.NhietDoSay,

532 │ ND\_HoHoa = dto.NhietDoHoHoa,

533 │ NS\_DoPH = dto.PHNuocSach,

534 │ BU\_DoAm = dto.DoAmBU,

535 │ Cepa = dto.Cepa,

536 │ XMS\_DoAm = dto.DoAmXMS,

537 │ XMT\_CoBot = dto.XacMiTuoiCoBot,

538 │ SO2 = dto.So2,

539 │ HamLuongTB = dto.HamLuongTB,

540 │ GhiChu = dto.GhiChu,

541 │ NguoiTao = objToken.userid,

542 │ NgayTao = DateTime.Now,

543 │ DaXoa = false

544 │ };

545 │ dbcontext.tbl\_ChiTietSoLieus.Add(newChiTiet);

546 │ dbcontext.SaveChanges();

547 │ return Json(new ResponseModelSuccess("Đã thêm chi tiết số liệu.", new { newId = newChiTiet.Ma }...

548 │ }

549 │ catch (Exception ex)

550 │ {

551 │ Debug.WriteLine($"Lỗi khi thêm chi tiết số liệu: {ex.Message}");

552 │ return Json(new ResponseModelError($"Lỗi server: {ex.InnerException?.Message ?? ex.Message}"));

553 │ }

554 │ }

555 │

556 │ [TokenUserCheckHTTP]

557 │ [HttpPost("updateSoLieuChiTiet")]

558 │ public IActionResult UpdateSoLieuChiTiet([FromBody] SoLieuChiTietInputDto dto)

559 │ {

560 │ if (!ModelState.IsValid) return Json(new ResponseModelError("Dữ liệu không hợp lệ."));

561 │ try

562 │ {

563 │ var slct = dbcontext.tbl\_ChiTietSoLieus.FirstOrDefault(x => x.Ma == dto.Ma && !(x.DaXoa ?? false));

564 │ if (slct != null)

565 │ {

566 │ bool timeChanged = slct.Gio != dto.Gio || slct.Phut != dto.Phut;

567 │ if (timeChanged)

568 │ {

569 │ bool exists = dbcontext.tbl\_ChiTietSoLieus.Any(x => x.MaSoLieu == slct.MaSoLieu && x.Ma != slct....

570 │ if (exists) return Json(new ResponseModelError("Đã tồn tại số liệu chi tiết khác ch...

571 │ }

572 │ slct.Gio = dto.Gio;

573 │ slct.Phut = dto.Phut;

574 │ slct.MaLoThuMua = dto.MaLo;

575 │ slct.B\_DoTrang = dto.DoTrang;

576 │ slct.B\_DoAm = dto.DoAm;

577 │

578 │ slct.B\_DoMin = dto.DoMin;

579 │ slct.B\_DoPH = dto.Ph;

580 │ slct.B\_DoKeoCP = dto.DoKeoCP;

581 │ slct.B\_DoKeoBU = dto.DoKeoBU;

582 │ slct.B\_TapChat = dto.TapChat;

583 │

584 │ slct.ND\_Lo = dto.NhietDoLo;

585 │ slct.ND\_Say = dto.NhietDoSay;

586 │ slct.ND\_HoHoa = dto.NhietDoHoHoa;

587 │ slct.NS\_DoPH = dto.PHNuocSach;

588 │

589 │ slct.BU\_DoAm = dto.DoAmBU;

590 │ slct.Cepa = dto.Cepa;

591 │ slct.XMS\_DoAm = dto.DoAmXMS;

592 │ slct.XMT\_CoBot = dto.XacMiTuoiCoBot;

593 │

594 │ slct.SO2 = dto.So2;

595 │ slct.HamLuongTB = dto.HamLuongTB;

596 │ slct.GhiChu = dto.GhiChu;

597 │ slct.NguoiSua = objToken.userid; slct.NgaySua = DateTime.Now;

598 │ dbcontext.SaveChanges();

599 │ return Json(new ResponseModelSuccess("Đã cập nhật chi tiết số liệu."));

600 │ }

601 │ return Json(new ResponseModelError("Không tìm thấy chi tiết số liệu."));

602 │ }

603 │ catch (Exception ex)

604 │ {

605 │ Debug.WriteLine($"Lỗi khi cập nhật chi tiết số liệu: {ex.Message}");

606 │ return Json(new ResponseModelError($"Lỗi server: {ex.InnerException?.Message ?? ex.Message}"));

607 │ }

608 │ }

609 │

610 │ [TokenUserCheckHTTP]

611 │ [HttpPost("deleteSoLieuChiTiet")]

612 │ public IActionResult DeleteSoLieuChiTiet([FromBody] SoLieuChiTietInputDto dto)

613 │ {

614 │ try

615 │ {

616 │ var slct = dbcontext.tbl\_ChiTietSoLieus.FirstOrDefault(x => x.Ma == dto.Ma && !(x.DaXoa ?? false));

617 │ if (slct != null)

618 │ {

619 │ slct.DaXoa = true; slct.NgayXoa = DateTime.Now; slct.NguoiXoa = objToken.userid;

620 │ dbcontext.SaveChanges();

621 │ return Json(new ResponseModelSuccess("Đã xóa chi tiết số liệu."));

622 │ }

623 │ return Json(new ResponseModelError("Không tìm thấy chi tiết số liệu hoặc đã được xóa....

624 │ }

625 │ catch (Exception ex)

626 │ {

627 │ Debug.WriteLine($"Lỗi khi xóa chi tiết số liệu: {ex.Message}");

628 │ return Json(new ResponseModelError($"Lỗi server: {ex.InnerException?.Message ?? ex.Message}"));

629 │ }

630 │ }

631 │

632 │ [TokenUserCheckHTTP]

633 │ [HttpPost("addSoLieuBanThanhPham")]

634 │ public IActionResult AddSoLieuBanThanhPham([FromBody] SoLieuBanThanhPhamInputDto dto)

635 │ {

636 │ if (!ModelState.IsValid) return Json(new ResponseModelError("Dữ liệu không hợp lệ."));

637 │ try

638 │ {

639 │ bool exists = dbcontext.tbl\_BanThanhPhams.Any(x => x.MaSoLieu == dto.MaSoLieu && x.Gio == dto.Gio && x.P...

640 │ if (exists) return Json(new ResponseModelError("Đã tồn tại số liệu bán thành phẩm cho mố...

641 │

642 │ var newBtp = new tbl\_BanThanhPham

643 │ {

644 │ MaSoLieu = dto.MaSoLieu,

645 │ Gio = dto.Gio,

646 │ Phut = dto.Phut,

647 │ MaLo = dto.MaLo,

648 │ pH\_RO = dto.PhRO,

649 │ HCL = dto.HCL,

650 │ NaOH = dto.NaOH,

651 │ HCL\_Loang = dto.HCL\_Loang,

652 │ NaOH\_Loang = dto.NaOH\_Loang,

653 │ CQ\_SodiumCarbonat = dto.SodiumCarbonat,

654 │ RasperChao = dto.RasperChao,

655 │ HamLuongTBTrongBaMi = dto.HamLuongTinhBot,

656 │ pHSuaBot = dto.PHSuaBot,

657 │ ND\_HoHoa = dto.NhietDoHoHoa,

658 │ DoNhot = dto.DoNhot,

659 │ GhiChu = dto.GhiChu,

660 │ NguoiPhanTich = dto.NguoiPhanTich,

661 │ UserCreated = objToken.userid,

662 │ CreatedDate = DateTime.Now,

663 │ DaXoa = false

664 │ };

665 │ dbcontext.tbl\_BanThanhPhams.Add(newBtp);

666 │ dbcontext.SaveChanges();

667 │ return Json(new ResponseModelSuccess("Đã thêm số liệu bán thành phẩm.", new { newId = newBtp....

668 │ }

669 │ catch (Exception ex)

670 │ {

671 │ Debug.WriteLine($"Lỗi khi thêm số liệu BTP: {ex.Message}");

672 │ return Json(new ResponseModelError($"Lỗi server: {ex.InnerException?.Message ?? ex.Message}"));

673 │ }

674 │ }

675 │

676 │ [TokenUserCheckHTTP]

677 │ [HttpPost("updateSoLieuBanThanhPham")]

678 │ public IActionResult UpdateSoLieuBanThanhPham([FromBody] SoLieuBanThanhPhamInputDto dto)

679 │ {

680 │ if (!ModelState.IsValid) return Json(new ResponseModelError("Dữ liệu không hợp lệ."));

681 │ try

682 │ {

683 │ var slbtp = dbcontext.tbl\_BanThanhPhams.FirstOrDefault(x => x.Ma == dto.Ma && !(x.DaXoa ?? false));

684 │ if (slbtp != null)

685 │ {

686 │ bool timeChanged = slbtp.Gio != dto.Gio || slbtp.Phut != dto.Phut;

687 │ if (timeChanged)

688 │ {

689 │ bool exists = dbcontext.tbl\_BanThanhPhams.Any(x => x.MaSoLieu == slbtp.MaSoLieu && x.Ma != slbtp...

690 │ if (exists) return Json(new ResponseModelError("Đã tồn tại số liệu BTP khác cho mốc...

691 │ }

692 │ slbtp.Gio = dto.Gio;

693 │ slbtp.Phut = dto.Phut;

694 │ slbtp.MaLo = dto.MaLo;

695 │ slbtp.pH\_RO = dto.PhRO;

696 │ slbtp.HCL = dto.HCL;

697 │ slbtp.NaOH = dto.NaOH;

698 │

699 │ slbtp.HCL\_Loang = dto.HCL\_Loang;

700 │ slbtp.NaOH\_Loang = dto.NaOH\_Loang;

701 │ slbtp.CQ\_SodiumCarbonat = dto.SodiumCarbonat;

702 │

703 │ slbtp.RasperChao = dto.RasperChao;

704 │ slbtp.HamLuongTBTrongBaMi = dto.HamLuongTinhBot;

705 │ slbtp.pHSuaBot = dto.PHSuaBot;

706 │

707 │ slbtp.ND\_HoHoa = dto.NhietDoHoHoa;

708 │ slbtp.DoNhot = dto.DoNhot;

709 │ slbtp.GhiChu = dto.GhiChu;

710 │ slbtp.NguoiPhanTich = dto.NguoiPhanTich;

711 │

712 │ slbtp.UserModified = objToken.userid;

713 │ slbtp.ModifiedDate = DateTime.Now;

714 │ dbcontext.SaveChanges();

715 │ return Json(new ResponseModelSuccess("Đã cập nhật số liệu bán thành phẩm."));

716 │ }

717 │ return Json(new ResponseModelError("Không tìm thấy số liệu bán thành phẩm."));

718 │ }

719 │ catch (Exception ex)

720 │ {

721 │ Debug.WriteLine($"Lỗi khi cập nhật số liệu BTP: {ex.Message}");

722 │ return Json(new ResponseModelError($"Lỗi server: {ex.InnerException?.Message ?? ex.Message}"));

723 │ }

724 │ }

725 │

726 │ [TokenUserCheckHTTP]

727 │ [HttpPost("deleteSoLieuBanThanhPham")]

728 │ public IActionResult DeleteSoLieuBanThanhPham([FromBody] SoLieuBanThanhPhamInputDto dto)

729 │ {

730 │ try

731 │ {

732 │ var slbtp = dbcontext.tbl\_BanThanhPhams.FirstOrDefault(x => x.Ma == dto.Ma && !(x.DaXoa ?? false));

733 │ if (slbtp != null)

734 │ {

735 │ slbtp.DaXoa = true; slbtp.DeletedDate = DateTime.Now; slbtp.UserDeleted = objToken.userid;

736 │ dbcontext.SaveChanges();

737 │ return Json(new ResponseModelSuccess("Đã xóa số liệu bán thành phẩm."));

738 │ }

739 │ return Json(new ResponseModelError("Không tìm thấy số liệu bán thành phẩm hoặc đã đư��...

740 │ }

741 │ catch (Exception ex)

742 │ {

743 │ Debug.WriteLine($"Lỗi khi xóa số liệu BTP: {ex.Message}");

744 │ return Json(new ResponseModelError($"Lỗi server: {ex.InnerException?.Message ?? ex.Message}"));

745 │ }

746 │ }

747 │ #endregion

748 │

749 │ #region Helper Actions & Private Methods

750 │

751 │ [TokenUserCheckHTTP]

752 │ [HttpPost("getGioiHanSoLieu")]

753 │ public IActionResult GetGioiHanSoLieu()

754 │ {

755 │ var data = dbcontext.tbl\_GioiHanSoLieus

756 │ .Where(x => x.MaChiNhanh == (int)MaChiNhanh.NM1 && (x.TrangThai ?? false) && !(x.DaXoa ?? false))

757 │ .Select(x => new { nameValue = x.Code, valueType = x.Loai, value = x.GiaTri })

758 │ .ToList();

759 │ return Json(new ResponseModel(QLCLConst.OK, "Lấy giới hạn số liệu thành công.", data));

760 │ }

761 │

762 │ [TokenUserCheckHTTP]

763 │ [HttpPost("getLoThuMua")]

764 │ public IActionResult GetLoThuMua()

765 │ {

766 │ var chiNhanh = dbcontext.tbl\_ChiNhanhs.FirstOrDefault(x => x.Ma == objToken.branch);

767 │ if (chiNhanh == null)

768 │ return Json(new ResponseModel(QLCLConst.Error, "Không tìm thấy thông tin chi nhánh.", null));

769 │

770 │ var maChiNhanhTapioCa = \_crmContext.SysBranch

771 │ .Where(sb => sb.Code == chiNhanh.Code && !(sb.Deleted ?? false))

772 │ .Select(sb => sb.Id)

773 │ .FirstOrDefault();

774 │

775 │

776 │ var data = \_crmContext.CloseChotLo

777 │ .Where(l => l.MaChiNhanhTapioCa == maChiNhanhTapioCa && !(l.Deleted ?? false) &&

778 │ (l.TrangThai != (int)TrangThaiChotLo.ChotThuMua && l.TrangThai != (int)TrangThaiChotLo.Cho...

779 │ .OrderByDescending(l => l.CreatedDate)

780 │ .Select(l => new { malo = l.Ma, codelo = l.MaLo })

781 │ .ToList();

782 │ return Json(new ResponseModel(QLCLConst.OK, "Lấy danh sách lô thu mua thành công.", data));

783 │ }

784 │

785 │ private List<LoThuMuaViewModel> GetActiveLoThuMuaList()

786 │ {

787 │ return \_crmContext.CloseChotLo

788 │ .Where(l => !(l.Deleted ?? false))

789 │ .Select(l => new LoThuMuaViewModel { Ma = l.Ma, MaLoDisplay = l.MaLo })

790 │ .ToList();

791 │ }

792 │ private class LoThuMuaViewModel

793 │ {

794 │ public int Ma { get; set; }

795 │ public string? MaLoDisplay { get; set; }

796 │ }

797 │

798 │

799 │ private List<dynamic> GetBotThuongThanhPhamReportData(ReportFilterModel filter)

800 │ {

801 │ var loThuMuaList = GetActiveLoThuMuaList();

802 │ var intermediateData = (from sl in dbcontext.tbl\_SoLieus

803 │ join ctsl in dbcontext.tbl\_ChiTietSoLieus on sl.Ma equals ctsl.MaSoLieu

804 │ where sl.Ngay.Date >= filter.FromDate.Date && sl.Ngay.Date <= filter.ToDate.Date &&

805 │ sl.MaChiNhanh == objToken.branch &&

806 │ !(sl.DaXoa ?? false) && !(ctsl.DaXoa ?? false)

807 │ select new

808 │ {

809 │ sl.Ngay,

810 │ sl.TruongCa,

811 │ sl.QC,

812 │ sl.ThuKhoBot,

813 │ sl.DotLo,

814 │ ctsl.Gio,

815 │ ctsl.Phut,

816 │ ctsl.MaLoThuMua,

817 │ ctsl.B\_DoTrang,

818 │ ctsl.B\_DoAm,

819 │ ctsl.B\_DoMin,

820 │ ctsl.B\_DoPH,

821 │ ctsl.B\_DoKeoCP,

822 │ ctsl.B\_DoKeoBU,

823 │ ctsl.ND\_HoHoa,

824 │ ctsl.B\_TapChat,

825 │ ctsl.HamLuongTB,

826 │ ctsl.NS\_DoPH,

827 │ ctsl.SO2,

828 │ ctsl.GhiChu,

829 │ }).ToList();

830 │

831 │ return intermediateData

832 │ .GroupJoin(loThuMuaList, item => item.MaLoThuMua, ltm => ltm.Ma, (item, loGroup) => new { item, loGroup ...

833 │ .SelectMany(x => x.loGroup.DefaultIfEmpty(), (x, ltm) => new

834 │ {

835 │ ngay = x.item.Ngay,

836 │ truongca = x.item.TruongCa,

837 │ qc = x.item.QC,

838 │ thukhobot = x.item.ThuKhoBot,

839 │ dotlo = x.item.DotLo,

840 │ gio = x.item.Gio,

841 │ phut = x.item.Phut,

842 │ maLo = ltm?.MaLoDisplay,

843 │ doTrang = ToNullableDecimal(x.item.B\_DoTrang, "0.0"),

844 │ doAm = ToNullableDecimal(x.item.B\_DoAm, "0.0"),

845 │ doMin = ToNullableDecimal(x.item.B\_DoMin, "0.00"),

846 │ ph = ToNullableDecimal(x.item.B\_DoPH, "0.00"),

847 │ doKeoCP = ToNullableDecimal(x.item.B\_DoKeoCP, "0"),

848 │ doKeoBU = ToNullableDecimal(x.item.B\_DoKeoBU, "0"),

849 │ tapChat = ToNullableDecimal(x.item.B\_TapChat, "0.000"),

850 │ nhietDoHoHoa = ToNullableDecimal(x.item.ND\_HoHoa, "0"),

851 │ hamLuongTinhBot = ToNullableDecimal(x.item.HamLuongTB, "0.000"),

852 │ pHNuocSach = ToNullableDecimal(x.item.NS\_DoPH, "0.00"),

853 │ sO2 = ToNullableDecimal(x.item.SO2, "0.00"),

854 │ ghiChu = x.item.GhiChu,

855 │ })

856 │ .OrderByDescending(x => x.ngay)

857 │ .ThenByDescending(x => Convert.ToInt32(x.gio))

858 │ .ThenByDescending(x => Convert.ToInt32(x.phut))

859 │ .ToList<dynamic>();

860 │ }

861 │

862 │ private List<dynamic> GetBotThuongBanThanhPhamReportData(ReportFilterModel filter)

863 │ {

864 │ var loThuMuaList = GetActiveLoThuMuaList();

865 │ var intermediateData = (from sl in dbcontext.tbl\_SoLieus

866 │ join btp in dbcontext.tbl\_BanThanhPhams on sl.Ma equals btp.MaSoLieu

867 │ where sl.Ngay.Date >= filter.FromDate.Date && sl.Ngay.Date <= filter.ToDate.Date &&

868 │ sl.MaChiNhanh == objToken.branch &&

869 │ !(sl.DaXoa ?? false) && !(btp.DaXoa ?? false)

870 │ select new

871 │ {

872 │ sl.Ngay,

873 │ btp.Gio,

874 │ btp.Phut,

875 │ btp.MaLo,

876 │ btp.pH\_RO,

877 │ btp.HCL,

878 │ btp.NaOH,

879 │ btp.HCL\_Loang,

880 │ btp.NaOH\_Loang,

881 │ btp.CQ\_SodiumCarbonat,

882 │ btp.RasperChao,

883 │ btp.HamLuongTBTrongBaMi,

884 │ btp.pHSuaBot,

885 │ btp.ND\_HoHoa,

886 │ btp.DoNhot,

887 │ btp.GhiChu,

888 │ btp.NguoiPhanTich,

889 │ }).ToList();

890 │

891 │ return intermediateData

892 │ .GroupJoin(loThuMuaList, item => item.MaLo, ltm => ltm.Ma, (item, loGroup) => new { item, loGroup })

893 │ .SelectMany(x => x.loGroup.DefaultIfEmpty(), (x, ltm) => new

894 │ {

895 │ ngay = x.item.Ngay,

896 │ gio = x.item.Gio,

897 │ phut = x.item.Phut,

898 │ maLo = ltm?.MaLoDisplay,

899 │ phRO = x.item.pH\_RO,

900 │ HCL = x.item.HCL,

901 │ NaOH = x.item.NaOH,

902 │ HCL\_Loang = x.item.HCL\_Loang,

903 │ NaOH\_Loang = x.item.NaOH\_Loang,

904 │ sodiumCarbonat = x.item.CQ\_SodiumCarbonat,

905 │ rasperChao = x.item.RasperChao,

906 │ hamLuongTinhBot = x.item.HamLuongTBTrongBaMi,

907 │ pHSuaBot = x.item.pHSuaBot,

908 │ nhietDoHoHoa = x.item.ND\_HoHoa,

909 │ doNhot = x.item.DoNhot,

910 │ ghiChu = x.item.GhiChu,

911 │ nguoiPhanTich = x.item.NguoiPhanTich,

912 │ })

913 │ .OrderBy(x => x.ngay)

914 │ .ThenBy(x => Convert.ToInt32(x.gio))

915 │ .ThenBy(x => Convert.ToInt32(x.phut))

916 │ .ToList<dynamic>();

917 │ }

918 │

919 │ private IActionResult CreateExcelFile(System.Data.DataTable dataTable, string templateFileName, string outputFil...

920 │ {

921 │ string fileTemplate = Path.Combine(\_hostingEnvironment.ContentRootPath, "wwwroot", "templates", templateFile...

922 │ if (!System.IO.File.Exists(fileTemplate))

923 │ {

924 │ return NotFound($"File template không tìm thấy: {templateFileName}");

925 │ }

926 │

927 │ var listGiaTriRiengLe = new List<ExcelValue>

928 │ {

929 │ new ExcelValue { Range = dateRangeCell, Value = dateRangeString }

930 │ };

931 │

932 │ try

933 │ {

934 │ Stream excelStream = ExportToExcel.FromTemplate(fileTemplate, "Excel", listGiaTriRiengLe, dataTable, tru...

935 │ string contentType = "application/vnd.openxmlformats-officedocument.spreadsheetml.sheet";

936 │

937 │ var contentDisposition = new System.Net.Mime.ContentDisposition

938 │ {

939 │ FileName = outputFileName,

940 │ Inline = false // buộc tải về

941 │ };

942 │ Response.Headers["Content-Disposition"] = contentDisposition.ToString();

943 │

944 │ return File(excelStream, contentType);

945 │ }

946 │ catch (Exception ex)

947 │ {

948 │ Debug.WriteLine($"Lỗi khi tạo file Excel: {ex.Message}");

949 │ return Json(new ResponseModelError($"Lỗi server khi tạo file Excel: {ex.Message}"));

950 │ }

951 │ }

952 │

953 │

954 │ private DateTime ValidateSqlDateTime(DateTime dt)

955 │ {

956 │ if (dt < (DateTime)SqlDateTime.MinValue.Value) return (DateTime)SqlDateTime.MinValue.Value;

957 │ if (dt > (DateTime)SqlDateTime.MaxValue.Value) return (DateTime)SqlDateTime.MaxValue.Value;

958 │ return dt;

959 │ }

960 │ private bool IsValidDateRange(DateTime fromDate, DateTime toDate)

961 │ {

962 │ return fromDate != DateTime.MinValue && toDate != DateTime.MinValue && fromDate <= toDate;

963 │ }

964 │

965 │ #endregion

966 │ private static object ToNullableDecimal(decimal? value, string format)

967 │ {

968 │ if (!value.HasValue || value.Value == 0)

969 │ return DBNull.Value;

970 │

971 │ int decimalPlaces = 0;

972 │

973 │ if (format.Contains('.'))

974 │ {

975 │ var parts = format.Split('.');

976 │ decimalPlaces = parts[1].Length;

977 │ }

978 │

979 │ return Math.Round(value.Value, decimalPlaces);

980 │ }

981 │ }

982 │

983 │ }

────────────────────────────────────────────────────────────

**📄 StocChiTietDieuChuyen.cs (CS, 43 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class StocChiTietDieuChuyen

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaPhieuDieuChuyen { get; set; }

11 │

12 │ public int? MaHangHoa { get; set; }

13 │

14 │ public int? MaDvt { get; set; }

15 │

16 │ public int? MaLoaiBao { get; set; }

17 │

18 │ public int? MaLoaiBot { get; set; }

19 │

20 │ public int? SoLuong { get; set; }

21 │

22 │ public int? SoLuongBao { get; set; }

23 │

24 │ public decimal? DonGia { get; set; }

25 │

26 │ public decimal? ThanhTien { get; set; }

27 │

28 │ public string? GhiChu { get; set; }

29 │

30 │ public int? UserCreated { get; set; }

31 │

32 │ public DateTime? CreatedDate { get; set; }

33 │

34 │ public int? UserModified { get; set; }

35 │

36 │ public DateTime? ModifiedDate { get; set; }

37 │

38 │ public bool? Deleted { get; set; }

39 │

40 │ public int? UserDeleted { get; set; }

41 │

42 │ public DateTime? DeletedDate { get; set; }

43 │ }

────────────────────────────────────────────────────────────

**📄 StocChiTietLenhXuatKho.cs (CS, 37 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class StocChiTietLenhXuatKho

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaLenhXuatKho { get; set; }

11 │

12 │ public int? MaHangHoa { get; set; }

13 │

14 │ public int? MaLoaiBao { get; set; }

15 │

16 │ public string? MaLoaiBot { get; set; }

17 │

18 │ public int? MaNhaVanChuyen { get; set; }

19 │

20 │ public string? SoXe { get; set; }

21 │

22 │ public int? SoLuong { get; set; }

23 │

24 │ public int? MaDvt { get; set; }

25 │

26 │ public int? SoLuongBao { get; set; }

27 │

28 │ public string? GhiChu { get; set; }

29 │

30 │ public int? UserCreated { get; set; }

31 │

32 │ public DateTime? CreatedDate { get; set; }

33 │

34 │ public int? UserModified { get; set; }

35 │

36 │ public DateTime? ModifiedDate { get; set; }

37 │ }

────────────────────────────────────────────────────────────

**📄 StocChiTietNhapXuat.cs (CS, 67 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class StocChiTietNhapXuat

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaNhapXuat { get; set; }

11 │

12 │ public int? MaKho { get; set; }

13 │

14 │ public int? MaLineKho { get; set; }

15 │

16 │ public string? LoSanXuat { get; set; }

17 │

18 │ public int? MaHangHoa { get; set; }

19 │

20 │ public DateTime? NgaySanXuat { get; set; }

21 │

22 │ public DateTime? NgaySanXuatTrenBao { get; set; }

23 │

24 │ public DateTime? NgayDongBao { get; set; }

25 │

26 │ public int? MaLoaiBot { get; set; }

27 │

28 │ public DateTime? NgayBatDauSx { get; set; }

29 │

30 │ public DateTime? NgayNgungSx { get; set; }

31 │

32 │ public DateTime? NgayHetHan { get; set; }

33 │

34 │ public int? MaLoaiBao { get; set; }

35 │

36 │ public int? SoLuongBao { get; set; }

37 │

38 │ public int? SoLuong { get; set; }

39 │

40 │ public int? MaDvt { get; set; }

41 │

42 │ public int? MaCtnhap { get; set; }

43 │

44 │ public int? MaCtlenhXuatKho { get; set; }

45 │

46 │ public decimal? DonGia { get; set; }

47 │

48 │ public decimal? ThanhTien { get; set; }

49 │

50 │ public int? MaCtdeNghi { get; set; }

51 │

52 │ public string? GhiChu { get; set; }

53 │

54 │ public int? UserCreated { get; set; }

55 │

56 │ public DateTime? CreatedDate { get; set; }

57 │

58 │ public int? UserModified { get; set; }

59 │

60 │ public DateTime? ModifiedDate { get; set; }

61 │

62 │ public bool? Deleted { get; set; }

63 │

64 │ public int? UserDeleted { get; set; }

65 │

66 │ public DateTime? DeletedDate { get; set; }

67 │ }

────────────────────────────────────────────────────────────

**📄 StocChiTietVanChuyenLenhXuatKho.cs (CS, 21 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class StocChiTietVanChuyenLenhXuatKho

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaLenhXuatKho { get; set; }

11 │

12 │ public int? MaNhaVanChuyen { get; set; }

13 │

14 │ public int? MaXe { get; set; }

15 │

16 │ public int? SoLuongChuyenCho { get; set; }

17 │

18 │ public int? MaDvt { get; set; }

19 │

20 │ public string? GhiChu { get; set; }

21 │ }

────────────────────────────────────────────────────────────

**📄 StocChiTietXuatKhoTuLenh.cs (CS, 23 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class StocChiTietXuatKhoTuLenh

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaLenhXuatKho { get; set; }

13 │

14 │ public int? MaKho { get; set; }

15 │

16 │ public int? MaLineKho { get; set; }

17 │

18 │ public int? MaCtnhap { get; set; }

19 │

20 │ public int? SoLuongXuat { get; set; }

21 │

22 │ public int? MaDvt { get; set; }

23 │ }

────────────────────────────────────────────────────────────

**📄 StocDanhMucKho.cs (CS, 41 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class StocDanhMucKho

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? KyHieu { get; set; }

13 │

14 │ public int? MaChiNhanh { get; set; }

15 │

16 │ public string? Ten { get; set; }

17 │

18 │ public string? TenNgan { get; set; }

19 │

20 │ public string? GroupZalo { get; set; }

21 │

22 │ public string? MaKhoKeToan { get; set; }

23 │

24 │ public int? MaLoaiKho { get; set; }

25 │

26 │ public bool? Status { get; set; }

27 │

28 │ public int? UserCreated { get; set; }

29 │

30 │ public DateTime? CreatedDate { get; set; }

31 │

32 │ public int? UserModified { get; set; }

33 │

34 │ public DateTime? ModifiedDate { get; set; }

35 │

36 │ public bool? Deleted { get; set; }

37 │

38 │ public int? UserDeleted { get; set; }

39 │

40 │ public DateTime? DeletedDate { get; set; }

41 │ }

────────────────────────────────────────────────────────────

**📄 StocGhepLo.cs (CS, 43 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class StocGhepLo

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public string? SoPhieu { get; set; }

13 │

14 │ public DateTime? NgayChungTu { get; set; }

15 │

16 │ public int? MaKho { get; set; }

17 │

18 │ public int? MaViTri { get; set; }

19 │

20 │ public int? MaHangHoa { get; set; }

21 │

22 │ public int? MaLoaiBao { get; set; }

23 │

24 │ public int? MaLoaiBot { get; set; }

25 │

26 │ public string? LoSanXuat { get; set; }

27 │

28 │ public string? GhiChu { get; set; }

29 │

30 │ public int? UserCreated { get; set; }

31 │

32 │ public DateTime? CreatedDate { get; set; }

33 │

34 │ public int? UserModified { get; set; }

35 │

36 │ public DateTime? ModifiedDate { get; set; }

37 │

38 │ public bool? Deleted { get; set; }

39 │

40 │ public int? UserDeleted { get; set; }

41 │

42 │ public DateTime? DeletedDate { get; set; }

43 │ }

────────────────────────────────────────────────────────────

**📄 StocKhoKeToan.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class StocKhoKeToan

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaHangHoa { get; set; }

13 │

14 │ public string? MaKhoKeToan { get; set; }

15 │

16 │ public string? Ten { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 StocKhoaSo.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class StocKhoaSo

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? Thang { get; set; }

13 │

14 │ public int? Nam { get; set; }

15 │

16 │ public bool? DaKhoaSo { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 StocLenhXuatKho.cs (CS, 62 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class StocLenhXuatKho

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaKh { get; set; }

13 │

14 │ public int? MaDonHang { get; set; }

15 │

16 │ public int? MaNhaVanChuyen { get; set; }

17 │

18 │ public string? NhaVanChuyen { get; set; }

19 │

20 │ public int? MaChiNhanhXuat { get; set; }

21 │

22 │ public DateTime? NgayChungTu { get; set; }

23 │

24 │ public string? SoChungTu { get; set; }

25 │

26 │ /// <summary>

27 │ /// 0: chưa kết thúc, 1: đã kết thúc

28 │ /// </summary>

29 │ public int? TrangThai { get; set; }

30 │

31 │ public string? GhiChu { get; set; }

32 │

33 │ public string? DiaChiGiaoHang { get; set; }

34 │

35 │ public string? Cang { get; set; }

36 │

37 │ public string? XuongTau { get; set; }

38 │

39 │ public string? SoTau { get; set; }

40 │

41 │ public string? GhiChuBaoMat { get; set; }

42 │

43 │ public decimal? ThuTienMat { get; set; }

44 │

45 │ public int? NguoiThu { get; set; }

46 │

47 │ public DateTime? NgayGiaoHang { get; set; }

48 │

49 │ public int? UserCreated { get; set; }

50 │

51 │ public DateTime? CreatedDate { get; set; }

52 │

53 │ public int? UserModified { get; set; }

54 │

55 │ public DateTime? ModifiedDate { get; set; }

56 │

57 │ public bool? Deleted { get; set; }

58 │

59 │ public int? UserDeleted { get; set; }

60 │

61 │ public DateTime? DeletedDate { get; set; }

62 │ }

────────────────────────────────────────────────────────────

**📄 StocLineKho.cs (CS, 39 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class StocLineKho

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaKho { get; set; }

13 │

14 │ public int? LoaiBao { get; set; }

15 │

16 │ public int? SoLuongToiDa { get; set; }

17 │

18 │ public string? Code { get; set; }

19 │

20 │ public string? Ten { get; set; }

21 │

22 │ public bool? CoHang { get; set; }

23 │

24 │ public bool? Status { get; set; }

25 │

26 │ public int? UserCreated { get; set; }

27 │

28 │ public DateTime? CreatedDate { get; set; }

29 │

30 │ public int? UserModified { get; set; }

31 │

32 │ public DateTime? ModifiedDate { get; set; }

33 │

34 │ public bool? Deleted { get; set; }

35 │

36 │ public int? UserDeleted { get; set; }

37 │

38 │ public DateTime? DeletedDate { get; set; }

39 │ }

────────────────────────────────────────────────────────────

**📄 StocLoaiKho.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class StocLoaiKho

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public bool? ChoPhepNhieuKho { get; set; }

13 │

14 │ public bool? Status { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 StocLoaiNhapXuat.cs (CS, 36 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class StocLoaiNhapXuat

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public bool? Status { get; set; }

15 │

16 │ /// <summary>

17 │ /// 1: Nhập kho, 0:Xuất kho

18 │ /// </summary>

19 │ public bool? NhapKho { get; set; }

20 │

21 │ public bool? BotBienTinh { get; set; }

22 │

23 │ public int? UserCreated { get; set; }

24 │

25 │ public DateTime? CreatedDate { get; set; }

26 │

27 │ public int? UserModified { get; set; }

28 │

29 │ public DateTime? ModifiedDate { get; set; }

30 │

31 │ public bool? Deleted { get; set; }

32 │

33 │ public int? UserDeleted { get; set; }

34 │

35 │ public DateTime? DeletedDate { get; set; }

36 │ }

────────────────────────────────────────────────────────────

**📄 StocPhieuDieuChuyen.cs (CS, 39 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class StocPhieuDieuChuyen

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public bool? NhapKho { get; set; }

11 │

12 │ public string? SoPhieuNhapXuat { get; set; }

13 │

14 │ public int? MaChiNhanhXuat { get; set; }

15 │

16 │ public int? MaChiNhanhNhan { get; set; }

17 │

18 │ public DateTime? NgayNhapXuat { get; set; }

19 │

20 │ public int? TrangThai { get; set; }

21 │

22 │ public int? MaPhieuXuatDieuChuyen { get; set; }

23 │

24 │ public string? GhiChu { get; set; }

25 │

26 │ public int? UserCreated { get; set; }

27 │

28 │ public DateTime? CreatedDate { get; set; }

29 │

30 │ public int? UserModified { get; set; }

31 │

32 │ public DateTime? ModifiedDate { get; set; }

33 │

34 │ public bool? Deleted { get; set; }

35 │

36 │ public int? UserDeleted { get; set; }

37 │

38 │ public DateTime? DeletedDate { get; set; }

39 │ }

────────────────────────────────────────────────────────────

**📄 StocPhieuNhapXuat.cs (CS, 69 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class StocPhieuNhapXuat

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public string? SoPhieu { get; set; }

13 │

14 │ public int? MaLoaiNx { get; set; }

15 │

16 │ public int? MaDonHang { get; set; }

17 │

18 │ public int? MaLenhXuatKho { get; set; }

19 │

20 │ public DateTime? NgayNhapXuat { get; set; }

21 │

22 │ public DateTime? NgayGiaoHang { get; set; }

23 │

24 │ public int? MaCa { get; set; }

25 │

26 │ public int? MaLo { get; set; }

27 │

28 │ public int? MaGhepLo { get; set; }

29 │

30 │ public string? NguoiPhuTrach { get; set; }

31 │

32 │ public string? GhiChu { get; set; }

33 │

34 │ public int? MaPhieuXuat { get; set; }

35 │

36 │ public bool? DaNhapKho { get; set; }

37 │

38 │ public string? Cang { get; set; }

39 │

40 │ public string? XuongTau { get; set; }

41 │

42 │ public string? SoTau { get; set; }

43 │

44 │ public bool? DaKetThuc { get; set; }

45 │

46 │ public int? MaNcc { get; set; }

47 │

48 │ public int? MaChiNhanhNhan { get; set; }

49 │

50 │ public int? MaPhieuDeXuat { get; set; }

51 │

52 │ public decimal? ChiPhiBocVac { get; set; }

53 │

54 │ public bool? DaTinhChiPhiBocXep { get; set; }

55 │

56 │ public int? UserCreated { get; set; }

57 │

58 │ public DateTime? CreatedDate { get; set; }

59 │

60 │ public int? UserModified { get; set; }

61 │

62 │ public DateTime? ModifiedDate { get; set; }

63 │

64 │ public bool? Deleted { get; set; }

65 │

66 │ public int? UserDeleted { get; set; }

67 │

68 │ public DateTime? DeletedDate { get; set; }

69 │ }

────────────────────────────────────────────────────────────

**📄 StocTonKho.cs (CS, 39 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class StocTonKho

7 │ {

8 │ public long Ma { get; set; }

9 │

10 │ public int? Nam { get; set; }

11 │

12 │ public int? Thang { get; set; }

13 │

14 │ public int? MaChiNhanh { get; set; }

15 │

16 │ public int? MaKho { get; set; }

17 │

18 │ public int? MaLineKho { get; set; }

19 │

20 │ public int? MaCtnhap { get; set; }

21 │

22 │ public int? MaNhaCc { get; set; }

23 │

24 │ public int? MaHangHoa { get; set; }

25 │

26 │ public decimal? GiaVon { get; set; }

27 │

28 │ public int? MaLoaiBao { get; set; }

29 │

30 │ public int? MaLoaiBot { get; set; }

31 │

32 │ public int? SoLuongDauKy { get; set; }

33 │

34 │ public int? SoLuongNhap { get; set; }

35 │

36 │ public int? SoLuongXuat { get; set; }

37 │

38 │ public int? SoLuongTon { get; set; }

39 │ }

────────────────────────────────────────────────────────────

**📄 SuaDonHang.cs (CS, 15 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SuaDonHang

7 │ {

8 │ public string? NgayDonHang { get; set; }

9 │

10 │ public string? SoDonHang { get; set; }

11 │

12 │ public string? MaKh { get; set; }

13 │

14 │ public double? GiaUsd { get; set; }

15 │ }

────────────────────────────────────────────────────────────

**📄 SysBranch.cs (CS, 85 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SysBranch

7 │ {

8 │ public int Id { get; set; }

9 │

10 │ public string Code { get; set; } = null!;

11 │

12 │ public string Name { get; set; } = null!;

13 │

14 │ public string? FullName { get; set; }

15 │

16 │ public string? TenPhapNhan { get; set; }

17 │

18 │ public string? Address { get; set; }

19 │

20 │ public string? Tel { get; set; }

21 │

22 │ public string? Fax { get; set; }

23 │

24 │ public string? Website { get; set; }

25 │

26 │ public string? Email { get; set; }

27 │

28 │ public string? Logo { get; set; }

29 │

30 │ public bool? PhapNhan { get; set; }

31 │

32 │ public string? GroupZalo { get; set; }

33 │

34 │ public string? GroupQlcl { get; set; }

35 │

36 │ public string? GroupCanMi { get; set; }

37 │

38 │ public string? GroupXuatHang { get; set; }

39 │

40 │ public string? GroupBanTaiChinh { get; set; }

41 │

42 │ public string? GroupTmcanhBao { get; set; }

43 │

44 │ public string? GroupTonKhoNm { get; set; }

45 │

46 │ public string? GroupBld { get; set; }

47 │

48 │ public string? GroupBldnm { get; set; }

49 │

50 │ public string? GroupBldKsx { get; set; }

51 │

52 │ public string? GroupKeToan { get; set; }

53 │

54 │ public DateTime? NgayApDungChotLo { get; set; }

55 │

56 │ public bool? KiemTraMaLoKhiNhap { get; set; }

57 │

58 │ public string? GroupBldQlcl { get; set; }

59 │

60 │ public string? GroupBldTonKho { get; set; }

61 │

62 │ public bool? Status { get; set; }

63 │

64 │ public string? MaDvcs { get; set; }

65 │

66 │ public string? IndexDv { get; set; }

67 │

68 │ public bool? BaoCaoNhanh { get; set; }

69 │

70 │ public int? UserCreated { get; set; }

71 │

72 │ public DateTime? CreatedDate { get; set; }

73 │

74 │ public int? UserModified { get; set; }

75 │

76 │ public DateTime? ModifiedDate { get; set; }

77 │

78 │ public bool? Deleted { get; set; }

79 │

80 │ public int? UserDeleted { get; set; }

81 │

82 │ public DateTime? DeletedDate { get; set; }

83 │

84 │ public int? NId { get; set; }

85 │ }

────────────────────────────────────────────────────────────

**📄 SysLog.cs (CS, 11 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SysLog

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Log { get; set; }

11 │ }

────────────────────────────────────────────────────────────

**📄 SysLogo.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SysLogo

7 │ {

8 │ public int Id { get; set; }

9 │

10 │ public string? LogoName { get; set; }

11 │

12 │ public string? LogoImage { get; set; }

13 │

14 │ public int? BranchId { get; set; }

15 │

16 │ public int? Index { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 SysMenu.cs (CS, 45 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SysMenu

7 │ {

8 │ public int Id { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public int? ParentId { get; set; }

13 │

14 │ public string? Name { get; set; }

15 │

16 │ public int? Sequence { get; set; }

17 │

18 │ public string? Link { get; set; }

19 │

20 │ public string? Icon { get; set; }

21 │

22 │ public string? ClassIcon { get; set; }

23 │

24 │ public int? ChildOfMenu { get; set; }

25 │

26 │ public bool? Target { get; set; }

27 │

28 │ public bool? Status { get; set; }

29 │

30 │ public string? UsingFor { get; set; }

31 │

32 │ public int? UserCreated { get; set; }

33 │

34 │ public DateTime? CreatedDate { get; set; }

35 │

36 │ public int? UserModified { get; set; }

37 │

38 │ public DateTime? ModifiedDate { get; set; }

39 │

40 │ public bool? Deleted { get; set; }

41 │

42 │ public int? UserDeleted { get; set; }

43 │

44 │ public DateTime? DeletedDate { get; set; }

45 │ }

────────────────────────────────────────────────────────────

**📄 SysOtp.cs (CS, 17 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SysOtp

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaUser { get; set; }

11 │

12 │ public int? MaToken { get; set; }

13 │

14 │ public string? Otp { get; set; }

15 │

16 │ public DateTime? CreateDate { get; set; }

17 │ }

────────────────────────────────────────────────────────────

**📄 SysParameter.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SysParameter

7 │ {

8 │ public int Id { get; set; }

9 │

10 │ public string Code { get; set; } = null!;

11 │

12 │ public string Name { get; set; } = null!;

13 │

14 │ public string Value { get; set; } = null!;

15 │

16 │ public string? Description { get; set; }

17 │

18 │ public bool? Status { get; set; }

19 │

20 │ public int? UserCreated { get; set; }

21 │

22 │ public DateTime? CreatedDate { get; set; }

23 │

24 │ public int? UserModified { get; set; }

25 │

26 │ public DateTime? ModifiedDate { get; set; }

27 │

28 │ public bool? Deleted { get; set; }

29 │

30 │ public int? UserDeleted { get; set; }

31 │

32 │ public DateTime? DeletedDate { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 SysPermission.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SysPermission

7 │ {

8 │ public int Id { get; set; }

9 │

10 │ public string Code { get; set; } = null!;

11 │

12 │ public string Name { get; set; } = null!;

13 │

14 │ public string? Description { get; set; }

15 │

16 │ public bool? FormAccess { get; set; }

17 │

18 │ public bool? Status { get; set; }

19 │

20 │ public int? UserCreated { get; set; }

21 │

22 │ public DateTime? CreatedDate { get; set; }

23 │

24 │ public int? UserModified { get; set; }

25 │

26 │ public DateTime? ModifiedDate { get; set; }

27 │

28 │ public bool? Deleted { get; set; }

29 │

30 │ public int? UserDeleted { get; set; }

31 │

32 │ public DateTime? DeletedDate { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 SysRole.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SysRole

7 │ {

8 │ public int Id { get; set; }

9 │

10 │ public string Code { get; set; } = null!;

11 │

12 │ public string Name { get; set; } = null!;

13 │

14 │ public string? Description { get; set; }

15 │

16 │ public bool? Status { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │

32 │ public int? MaChucVu { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 SysRoleBaoCaoBld.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SysRoleBaoCaoBld

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? UserId { get; set; }

11 │

12 │ public int? MaBaoCao { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 SysRoleMenu.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SysRoleMenu

7 │ {

8 │ public int Id { get; set; }

9 │

10 │ public int RoleId { get; set; }

11 │

12 │ public int MenuId { get; set; }

13 │

14 │ public bool? AllowAccess { get; set; }

15 │

16 │ public bool? Status { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 SysRolePermission.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SysRolePermission

7 │ {

8 │ public int Id { get; set; }

9 │

10 │ public int RoleMenuId { get; set; }

11 │

12 │ public int PermissionId { get; set; }

13 │

14 │ public bool? Status { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 SysSentEmailHistory.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SysSentEmailHistory

7 │ {

8 │ public int Id { get; set; }

9 │

10 │ public string Sender { get; set; } = null!;

11 │

12 │ public string Receiver { get; set; } = null!;

13 │

14 │ public string? Cc { get; set; }

15 │

16 │ public string? Bcc { get; set; }

17 │

18 │ public string? Subject { get; set; }

19 │

20 │ public string? Description { get; set; }

21 │

22 │ public bool? SendSuccess { get; set; }

23 │

24 │ public DateTime? CreatedDate { get; set; }

25 │

26 │ public string? Type { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 SysStatus.cs (CS, 15 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SysStatus

7 │ {

8 │ public int Id { get; set; }

9 │

10 │ public string? StatusMes { get; set; }

11 │

12 │ public int? Status { get; set; }

13 │

14 │ public string? Page { get; set; }

15 │ }

────────────────────────────────────────────────────────────

**📄 SysUser.cs (CS, 79 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SysUser

7 │ {

8 │ public int Id { get; set; }

9 │

10 │ public int BranchId { get; set; }

11 │

12 │ public string Username { get; set; } = null!;

13 │

14 │ public string Password { get; set; } = null!;

15 │

16 │ public string? Sdt { get; set; }

17 │

18 │ public string FullName { get; set; } = null!;

19 │

20 │ public string? Fcm { get; set; }

21 │

22 │ public string? CodeCv { get; set; }

23 │

24 │ public string? DepartmentId { get; set; }

25 │

26 │ public string? Msnv { get; set; }

27 │

28 │ public int? MaPhongBan { get; set; }

29 │

30 │ public string? DepartmentName { get; set; }

31 │

32 │ public string? NhomVungMien { get; set; }

33 │

34 │ public string? Position { get; set; }

35 │

36 │ public int? MaViTri { get; set; }

37 │

38 │ public int? MaViTri1 { get; set; }

39 │

40 │ public int? MaChucNang { get; set; }

41 │

42 │ public string? KhoHang { get; set; }

43 │

44 │ public string? LoaiHangHoa { get; set; }

45 │

46 │ public bool? EmailReceive { get; set; }

47 │

48 │ public bool? Gender { get; set; }

49 │

50 │ public string? Email { get; set; }

51 │

52 │ public string? Photo { get; set; }

53 │

54 │ public bool? Status { get; set; }

55 │

56 │ public bool? LdapLogin { get; set; }

57 │

58 │ public int? TyTrongHqcv { get; set; }

59 │

60 │ public int? TyTrongNqqd { get; set; }

61 │

62 │ public int? UserCreated { get; set; }

63 │

64 │ public DateTime? CreatedDate { get; set; }

65 │

66 │ public int? UserModified { get; set; }

67 │

68 │ public DateTime? ModifiedDate { get; set; }

69 │

70 │ public bool? Deleted { get; set; }

71 │

72 │ public int? UserDeleted { get; set; }

73 │

74 │ public DateTime? DeletedDate { get; set; }

75 │

76 │ public int? MaChucVu { get; set; }

77 │

78 │ public bool? Reminder { get; set; }

79 │ }

────────────────────────────────────────────────────────────

**📄 SysUserHistory.cs (CS, 39 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SysUserHistory

7 │ {

8 │ public int Id { get; set; }

9 │

10 │ public int UserId { get; set; }

11 │

12 │ public int BranchId { get; set; }

13 │

14 │ public string? Msnv { get; set; }

15 │

16 │ public int? MaPhongBan { get; set; }

17 │

18 │ public int? MaViTri { get; set; }

19 │

20 │ public int? MaChucNang { get; set; }

21 │

22 │ public int? TyTrongHqcv { get; set; }

23 │

24 │ public int? TyTrongNqqd { get; set; }

25 │

26 │ public int? UserCreated { get; set; }

27 │

28 │ public DateTime? CreatedDate { get; set; }

29 │

30 │ public int? UserModified { get; set; }

31 │

32 │ public DateTime? ModifiedDate { get; set; }

33 │

34 │ public bool? Deleted { get; set; }

35 │

36 │ public int? UserDeleted { get; set; }

37 │

38 │ public DateTime? DeletedDate { get; set; }

39 │ }

────────────────────────────────────────────────────────────

**📄 SysUserRole.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class SysUserRole

7 │ {

8 │ public int Id { get; set; }

9 │

10 │ public int UserId { get; set; }

11 │

12 │ public int BranchId { get; set; }

13 │

14 │ public int RoleId { get; set; }

15 │

16 │ public bool? Status { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 TblBaoCaoBld.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblBaoCaoBld

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public int? MaLoaiBaoCao { get; set; }

13 │

14 │ public string? Ten { get; set; }

15 │

16 │ public string? MoTa { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 TblCaSanXuat.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblCaSanXuat

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public bool? Status { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 TblCangDen.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblCangDen

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public decimal? GiaVnd { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 TblCheDoKhachHang.cs (CS, 37 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblCheDoKhachHang

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaKh { get; set; }

11 │

12 │ public int? HoaHong { get; set; }

13 │

14 │ public int? ChiPhiKhac { get; set; }

15 │

16 │ public int? ChiPhiHoTro { get; set; }

17 │

18 │ public int? HanMucNo { get; set; }

19 │

20 │ public int? ThoiHanNo { get; set; }

21 │

22 │ public DateTime? NgayApDung { get; set; }

23 │

24 │ public int? UserCreated { get; set; }

25 │

26 │ public DateTime? CreatedDate { get; set; }

27 │

28 │ public int? UserModified { get; set; }

29 │

30 │ public DateTime? ModifiedDate { get; set; }

31 │

32 │ public bool? Deleted { get; set; }

33 │

34 │ public int? UserDeleted { get; set; }

35 │

36 │ public DateTime? DeletedDate { get; set; }

37 │ }

────────────────────────────────────────────────────────────

**📄 TblChiTietDuyetChungTu.cs (CS, 47 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblChiTietDuyetChungTu

7 │ {

8 │ public Guid Ma { get; set; }

9 │

10 │ public Guid? MaChungTu { get; set; }

11 │

12 │ public int? ThuTu { get; set; }

13 │

14 │ public int? MaNhomChucVu { get; set; }

15 │

16 │ public int? MaChucVu { get; set; }

17 │

18 │ public string? TenViTri { get; set; }

19 │

20 │ public int? NguoiDuyet { get; set; }

21 │

22 │ public bool? DuyetTheoPhongBan { get; set; }

23 │

24 │ public int? DaDuyet { get; set; }

25 │

26 │ public int? UserCreated { get; set; }

27 │

28 │ public DateTime? CreatedDate { get; set; }

29 │

30 │ public int? UserModified { get; set; }

31 │

32 │ public DateTime? ModifiedDate { get; set; }

33 │

34 │ public bool? Deleted { get; set; }

35 │

36 │ public int? UserDeleted { get; set; }

37 │

38 │ public DateTime? DeletedDate { get; set; }

39 │

40 │ public DateTime? NgayDuyet { get; set; }

41 │

42 │ public bool? ChoPhepDuyet { get; set; }

43 │

44 │ public bool? DuyetTheoDonViChi { get; set; }

45 │

46 │ public int? NguoiKiemTra { get; set; }

47 │ }

────────────────────────────────────────────────────────────

**📄 TblChiTietQuyTrinhDuyet.cs (CS, 39 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblChiTietQuyTrinhDuyet

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaQuyTrinh { get; set; }

11 │

12 │ public int? ThuTu { get; set; }

13 │

14 │ public string? TenViTri { get; set; }

15 │

16 │ public int? MaNhomChucVu { get; set; }

17 │

18 │ public bool? DuyetTheoPhongBan { get; set; }

19 │

20 │ public bool? Status { get; set; }

21 │

22 │ public int? UserCreated { get; set; }

23 │

24 │ public DateTime? CreatedDate { get; set; }

25 │

26 │ public int? UserModified { get; set; }

27 │

28 │ public DateTime? ModifiedDate { get; set; }

29 │

30 │ public bool? Deleted { get; set; }

31 │

32 │ public int? UserDeleted { get; set; }

33 │

34 │ public DateTime? DeletedDate { get; set; }

35 │

36 │ public int? MaTrangThaiQuyTrinhDuyet { get; set; }

37 │

38 │ public bool? DuyetTheoDonViChi { get; set; }

39 │ }

────────────────────────────────────────────────────────────

**📄 TblChiTieu.cs (CS, 35 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblChiTieu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public string? TenTiengAnh { get; set; }

13 │

14 │ public int? MaDvt { get; set; }

15 │

16 │ public string? TieuChuan { get; set; }

17 │

18 │ public string? TieuChuanTiengAnh { get; set; }

19 │

20 │ public bool? Status { get; set; }

21 │

22 │ public int? UserCreated { get; set; }

23 │

24 │ public DateTime? CreatedDate { get; set; }

25 │

26 │ public int? UserModified { get; set; }

27 │

28 │ public DateTime? ModifiedDate { get; set; }

29 │

30 │ public bool? Deleted { get; set; }

31 │

32 │ public int? UserDeleted { get; set; }

33 │

34 │ public DateTime? DeletedDate { get; set; }

35 │ }

────────────────────────────────────────────────────────────

**📄 TblChucVu.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblChucVu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public int? MaNhomChucVu { get; set; }

15 │

16 │ public bool? TrangThai { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 TblCont.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblCont

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public decimal? GiaVnd { get; set; }

13 │

14 │ public decimal? GiaUsd { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 TblDichVuNoiDia.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblDichVuNoiDia

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public bool? TrangThai { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 TblDonViTinh.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblDonViTinh

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public int? MaDvtquiDoi { get; set; }

13 │

14 │ public int? GiaTriQuiDoi { get; set; }

15 │

16 │ public bool? Status { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 TblEmail.cs (CS, 37 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblEmail

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string Domain { get; set; } = null!;

11 │

12 │ public string Type { get; set; } = null!;

13 │

14 │ public string HoTen { get; set; } = null!;

15 │

16 │ public string Email { get; set; } = null!;

17 │

18 │ public string? ParentId { get; set; }

19 │

20 │ public bool? CongKhai { get; set; }

21 │

22 │ public bool? TrangThai { get; set; }

23 │

24 │ public int? UserCreated { get; set; }

25 │

26 │ public DateTime? CreatedDate { get; set; }

27 │

28 │ public int? UserModified { get; set; }

29 │

30 │ public DateTime? ModifiedDate { get; set; }

31 │

32 │ public bool? Deleted { get; set; }

33 │

34 │ public int? UserDeleted { get; set; }

35 │

36 │ public DateTime? DeletedDate { get; set; }

37 │ }

────────────────────────────────────────────────────────────

**📄 TblGiaBocXep.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblGiaBocXep

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? LoaiBao { get; set; }

13 │

14 │ public int? DonGia { get; set; }

15 │

16 │ public bool? TrangThai { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 TblGiaNiemYet.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblGiaNiemYet

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaHang { get; set; }

11 │

12 │ public int GiaNiemYet { get; set; }

13 │

14 │ public decimal? GiaNiemYetUsd { get; set; }

15 │

16 │ public DateTime? NgayApDung { get; set; }

17 │

18 │ public bool? Status { get; set; }

19 │

20 │ public int? UserCreated { get; set; }

21 │

22 │ public DateTime? CreatedDate { get; set; }

23 │

24 │ public int? UserModified { get; set; }

25 │

26 │ public DateTime? ModifiedDate { get; set; }

27 │

28 │ public bool? Deleted { get; set; }

29 │

30 │ public int? UserDeleted { get; set; }

31 │

32 │ public DateTime? DeletedDate { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 TblHangHoa.cs (CS, 51 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblHangHoa

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public string? TenTiengAnh { get; set; }

15 │

16 │ public int? MaDvt { get; set; }

17 │

18 │ public int? DonGia { get; set; }

19 │

20 │ public decimal? DonGiaUsd { get; set; }

21 │

22 │ public bool? HangHoa { get; set; }

23 │

24 │ public int? MaLoaiHangHoa { get; set; }

25 │

26 │ public string? KhoXuat { get; set; }

27 │

28 │ public int? MaNhaVanChuyen { get; set; }

29 │

30 │ public bool? HangChoPhepBan { get; set; }

31 │

32 │ public int? MaDanhMucChiPhi { get; set; }

33 │

34 │ public bool? Status { get; set; }

35 │

36 │ public int? UserCreated { get; set; }

37 │

38 │ public DateTime? CreatedDate { get; set; }

39 │

40 │ public int? UserModified { get; set; }

41 │

42 │ public DateTime? ModifiedDate { get; set; }

43 │

44 │ public bool? Deleted { get; set; }

45 │

46 │ public int? UserDeleted { get; set; }

47 │

48 │ public DateTime? DeletedDate { get; set; }

49 │

50 │ public bool? HoaDon { get; set; }

51 │ }

────────────────────────────────────────────────────────────

**📄 TblLaiTaiChinh.cs (CS, 15 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblLaiTaiChinh

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public decimal? LaiTc { get; set; }

11 │

12 │ public int? UserModified { get; set; }

13 │

14 │ public DateTime? ModifiedDate { get; set; }

15 │ }

────────────────────────────────────────────────────────────

**📄 TblLichSuNhaVanChuyen.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblLichSuNhaVanChuyen

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaNhaVanChuyen { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public int? GiaVanChuyen { get; set; }

15 │

16 │ public decimal? GiaUsd { get; set; }

17 │

18 │ public DateTime? NgayApDung { get; set; }

19 │

20 │ public int? UserCreated { get; set; }

21 │

22 │ public DateTime? CreatedDate { get; set; }

23 │

24 │ public int? UserModified { get; set; }

25 │

26 │ public DateTime? ModifiedDate { get; set; }

27 │

28 │ public bool? Deleted { get; set; }

29 │

30 │ public int? UserDeleted { get; set; }

31 │

32 │ public DateTime? DeletedDate { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 TblLichSuTyGia.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblLichSuTyGia

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaTyGia { get; set; }

11 │

12 │ public int? MaTienTe { get; set; }

13 │

14 │ public int? TyGia { get; set; }

15 │

16 │ public DateTime? NgayApDung { get; set; }

17 │

18 │ public bool? Status { get; set; }

19 │

20 │ public int? UserCreated { get; set; }

21 │

22 │ public DateTime? CreatedDate { get; set; }

23 │

24 │ public int? UserModified { get; set; }

25 │

26 │ public DateTime? ModifiedDate { get; set; }

27 │

28 │ public bool? Deleted { get; set; }

29 │

30 │ public int? UserDeleted { get; set; }

31 │

32 │ public DateTime? DeletedDate { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 TblLoaiBao.cs (CS, 41 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblLoaiBao

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? MaLo { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public string? LoaiBao { get; set; }

15 │

16 │ public string? DienGiai { get; set; }

17 │

18 │ public int? ChiPhi { get; set; }

19 │

20 │ public decimal? ChiPhiUsd { get; set; }

21 │

22 │ public bool? BaoHungDuy { get; set; }

23 │

24 │ public int? MaThuongHieu { get; set; }

25 │

26 │ public bool? Status { get; set; }

27 │

28 │ public int? UserCreated { get; set; }

29 │

30 │ public DateTime? CreatedDate { get; set; }

31 │

32 │ public int? UserModified { get; set; }

33 │

34 │ public DateTime? ModifiedDate { get; set; }

35 │

36 │ public bool? Deleted { get; set; }

37 │

38 │ public int? UserDeleted { get; set; }

39 │

40 │ public DateTime? DeletedDate { get; set; }

41 │ }

────────────────────────────────────────────────────────────

**📄 TblLoaiBot.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblLoaiBot

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public string? Code { get; set; }

13 │

14 │ public string? Ten { get; set; }

15 │

16 │ public string? DienGiai { get; set; }

17 │

18 │ public bool? Status { get; set; }

19 │

20 │ public int? UserCreated { get; set; }

21 │

22 │ public DateTime? CreatedDate { get; set; }

23 │

24 │ public int? UserModified { get; set; }

25 │

26 │ public DateTime? ModifiedDate { get; set; }

27 │

28 │ public bool? Deleted { get; set; }

29 │

30 │ public int? UserDeleted { get; set; }

31 │

32 │ public DateTime? DeletedDate { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 TblLoaiChungTu.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblLoaiChungTu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public bool? TrangThai { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 TblLoaiHangHoa.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblLoaiHangHoa

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public bool? NhieuSanPham { get; set; }

13 │

14 │ public bool? BaoCao { get; set; }

15 │

16 │ public bool? Status { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 TblLyDo.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblLyDo

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public bool? Status { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 TblLyDoDuyetDonHang.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblLyDoDuyetDonHang

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public bool? Status { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 TblNhaCungCap.cs (CS, 39 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblNhaCungCap

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? MaKt { get; set; }

11 │

12 │ public string TenNhaCc { get; set; } = null!;

13 │

14 │ public string? DiaChi { get; set; }

15 │

16 │ public string? DienThoai { get; set; }

17 │

18 │ public string? Mst { get; set; }

19 │

20 │ public string? Email { get; set; }

21 │

22 │ public string? NguoiLienHe { get; set; }

23 │

24 │ public bool? TrangThai { get; set; }

25 │

26 │ public int? UserCreated { get; set; }

27 │

28 │ public DateTime? CreatedDate { get; set; }

29 │

30 │ public int? UserModified { get; set; }

31 │

32 │ public DateTime? ModifiedDate { get; set; }

33 │

34 │ public bool? Deleted { get; set; }

35 │

36 │ public int? UserDeleted { get; set; }

37 │

38 │ public DateTime? DeletedDate { get; set; }

39 │ }

────────────────────────────────────────────────────────────

**📄 TblNhaVanChuyen.cs (CS, 43 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblNhaVanChuyen

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public int? GiaVanChuyen { get; set; }

15 │

16 │ public decimal? GiaUsd { get; set; }

17 │

18 │ public int? MaDvt { get; set; }

19 │

20 │ public bool? Status { get; set; }

21 │

22 │ public bool? XuatKhau { get; set; }

23 │

24 │ public string? DiaChi { get; set; }

25 │

26 │ public string? SoDienThoai { get; set; }

27 │

28 │ public int? UserCreated { get; set; }

29 │

30 │ public DateTime? CreatedDate { get; set; }

31 │

32 │ public int? UserModified { get; set; }

33 │

34 │ public DateTime? ModifiedDate { get; set; }

35 │

36 │ public bool? Deleted { get; set; }

37 │

38 │ public int? UserDeleted { get; set; }

39 │

40 │ public DateTime? DeletedDate { get; set; }

41 │

42 │ public bool? TinhCongNo { get; set; }

43 │ }

────────────────────────────────────────────────────────────

**📄 TblNhomChucVu.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblNhomChucVu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public bool? TrangThai { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 TblPallet.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblPallet

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public decimal? GiaVnd { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 TblPhongBan.cs (CS, 47 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblPhongBan

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string Code { get; set; } = null!;

11 │

12 │ public int? MaCn { get; set; }

13 │

14 │ public bool? KhoTong { get; set; }

15 │

16 │ public string Ten { get; set; } = null!;

17 │

18 │ public string? TenVietTat { get; set; }

19 │

20 │ public string? MoTa { get; set; }

21 │

22 │ public string? GroupZaloDonHang { get; set; }

23 │

24 │ public string? GroupZaloBaoGia { get; set; }

25 │

26 │ public string? GroupZaloHopDong { get; set; }

27 │

28 │ public string? GroupTmcanhBao { get; set; }

29 │

30 │ public string? GroupSpp { get; set; }

31 │

32 │ public bool? TrangThai { get; set; }

33 │

34 │ public int? UserCreated { get; set; }

35 │

36 │ public DateTime? CreatedDate { get; set; }

37 │

38 │ public int? UserModified { get; set; }

39 │

40 │ public DateTime? ModifiedDate { get; set; }

41 │

42 │ public bool? Deleted { get; set; }

43 │

44 │ public int? UserDeleted { get; set; }

45 │

46 │ public DateTime? DeletedDate { get; set; }

47 │ }

────────────────────────────────────────────────────────────

**📄 TblQuanLySale.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblQuanLySale

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaQuanLy { get; set; }

11 │

12 │ public int? MaNhanVienBh { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 TblQuyTrinhDuyetChungTu.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblQuyTrinhDuyetChungTu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public int? MaLoaiChungTu { get; set; }

13 │

14 │ public string? Ten { get; set; }

15 │

16 │ public bool? TrangThai { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 TblSoChungTu.cs (CS, 17 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblSoChungTu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? Nam { get; set; }

13 │

14 │ public int? Thang { get; set; }

15 │

16 │ public int? SoChungTu { get; set; }

17 │ }

────────────────────────────────────────────────────────────

**📄 TblTaiKhoanNhaCc.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblTaiKhoanNhaCc

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaNhaCc { get; set; }

11 │

12 │ public string? SoTaiKhoan { get; set; }

13 │

14 │ public string? TenNganHang { get; set; }

15 │

16 │ public bool? Status { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 TblThueVat.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblThueVat

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public decimal? GiaTri { get; set; }

13 │

14 │ public bool? Status { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 TblThuogHieu.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblThuogHieu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public bool? TrangThai { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 TblTienTe.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblTienTe

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public string? MoTa { get; set; }

13 │

14 │ public bool? Status { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │

30 │ public int? TyGiaMin { get; set; }

31 │

32 │ public int? TyGiaMax { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 TblTrangThai.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblTrangThai

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public bool? Status { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 TblTrangThaiDonHang.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblTrangThaiDonHang

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public bool? Status { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 TblTrangThaiHopDong.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblTrangThaiHopDong

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public bool? Status { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 TblTrangThaiLenhXuatKho.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblTrangThaiLenhXuatKho

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public bool? Status { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 TblTyGia.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TblTyGia

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaTienTe { get; set; }

11 │

12 │ public int? TyGia { get; set; }

13 │

14 │ public DateTime? NgayApDung { get; set; }

15 │

16 │ public bool? Status { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 ThoiGian.cs (CS, 13 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class ThoiGian

7 │ {

8 │ public DateTime? Ngay { get; set; }

9 │

10 │ public string? Gio { get; set; }

11 │

12 │ public string? Phut { get; set; }

13 │ }

────────────────────────────────────────────────────────────

**📄 TieuChuanController.cs (CS, 228 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using QuanLyChatLuong.Common;

3 │ using QuanLyChatLuong.Models.APIModels;

4 │ using QuanLyChatLuong.Models;

5 │ using Microsoft.DotNet.Scaffolding.Shared.CodeModifier.CodeChange;

6 │ using NuGet.Configuration;

7 │ using System.Diagnostics;

8 │ using Spire.Pdf.Exporting.XPS.Schema;

9 │ using QuanLyChatLuong.Models.CRMEntity;

10 │ using System.Runtime.CompilerServices;

11 │ using Microsoft.EntityFrameworkCore;

12 │

13 │ namespace QuanLyChatLuong.Controllers

14 │ {

15 │ [Route("tieuchuan")]

16 │ public class TieuChuanController : BaseController

17 │ {

18 │ private readonly IWebHostEnvironment \_hostingEnvironment;

19 │ private CrmContext crmContext { get; }

20 │

21 │ public TieuChuanController(QuanLyChatLuongDbContext dbcontext, IWebHostEnvironment hostingEnvironment, MemoryCac...

22 │ {

23 │ m\_MenuID = MenuID.tieuchuan;

24 │ \_hostingEnvironment = hostingEnvironment;

25 │ crmContext = crmdbcontext;

26 │ }

27 │ public class UpdateTieuChuanModel

28 │ {

29 │ public List<bt\_TieuChuan> listTieuChuan { get; set; }

30 │ }

31 │ public class UpdateTieuChuanChiTietModel

32 │ {

33 │ public int maTieuChuan { get; set; }

34 │ public List<bt\_ChiTietTieuChuan> listChiTiet { get; set; }

35 │ }

36 │

37 │ [TokenUserCheckHTTP]

38 │ [HttpGet]

39 │ public IActionResult Index()

40 │ {

41 │

42 │ return View();

43 │ }

44 │

45 │ [TokenUserCheckHTTP(Permission.Add)]

46 │ [HttpGet("add")]

47 │ public IActionResult Add()

48 │ {

49 │

50 │ return View("Form");

51 │ }

52 │

53 │

54 │ [TokenUserCheckHTTP()]

55 │ [HttpPost("updateAPI")]

56 │ public IActionResult UpdateAPI([FromBody] UpdateTieuChuanModel model)

57 │ {

58 │ using (var transaction = dbcontext.Database.BeginTransaction())

59 │ {

60 │ try

61 │ {

62 │ var lstTieuChuan = model.listTieuChuan;

63 │ var list\_old = dbcontext.bt\_TieuChuans.Where(x => !(x.DaXoa ?? false)).ToList();

64 │ var list\_deleted = list\_old.Where(t => !lstTieuChuan.Select(x => x.Ma).Contains(t.Ma))

65 │ .Select(t => { t.DaXoa = true; t.NgayXoa = DateTime.Now; t.NguoiXoa = ob...

66 │

67 │ if (list\_deleted != null && list\_deleted.Count > 0)

68 │ {

69 │ dbcontext.UpdateRange(list\_deleted);

70 │ dbcontext.SaveChanges();

71 │ }

72 │

73 │ var list\_update = (from lst\_moi in lstTieuChuan

74 │ join lst\_cu in list\_old on lst\_moi.Ma equals lst\_cu.Ma

75 │ select new { cu = lst\_cu, moi = lst\_moi }).ToList().Select(t =>

76 │ {

77 │ t.cu.MaSanPham = t.moi.MaSanPham;

78 │ t.cu.TieuChuan = t.moi.TieuChuan;

79 │ t.cu.MaSanPham = t.moi.MaSanPham;

80 │ t.cu.MaVatLieuBaoBi = t.moi.MaVatLieuBaoBi;

81 │ t.cu.TrangThai = t.moi.TrangThai;

82 │ return t.cu;

83 │ }).ToList();

84 │ if (list\_update != null && list\_update.Count > 0)

85 │ {

86 │ dbcontext.UpdateRange(list\_update);

87 │ dbcontext.SaveChanges();

88 │ }

89 │ var list\_insert = lstTieuChuan.Where(t => !list\_old.Select(x => x.Ma).Contains(t.Ma)).Select(t =>

90 │ {

91 │ t.MaChiNhanh = objToken.branch; t.NgayTao = DateTime.Now; t.NguoiTao = objToken.userid; return t...

92 │ }).ToList();

93 │

94 │ if (list\_insert != null && list\_insert.Count > 0)

95 │ {

96 │ dbcontext.AddRange(list\_insert);

97 │ dbcontext.SaveChanges();

98 │ }

99 │

100 │ transaction.Commit();

101 │ }

102 │ catch (Exception ex)

103 │ {

104 │ transaction.Rollback();

105 │ return Json(new ResponseModelError(ex.Message));

106 │ }

107 │ }

108 │

109 │ return Json(new ResponseModelSuccess(QLCLConst.Successfully));

110 │ }

111 │ [TokenUserCheckHTTP()]

112 │ [HttpPost("updateAPIChiTiet")]

113 │ public IActionResult UpdateAPIChiTiet([FromBody] UpdateTieuChuanChiTietModel model)

114 │ {

115 │ using (var transaction = dbcontext.Database.BeginTransaction())

116 │ {

117 │ try

118 │ {

119 │ var lstChiTiet = model.listChiTiet;

120 │ var maTieuChuan = model.maTieuChuan;

121 │ if (maTieuChuan == 0) throw new Exception("Kiểm tra lại, chưa chọn dòng tiêu chuẩn");

122 │

123 │ var list\_old = dbcontext.bt\_ChiTietTieuChuans.Where(x => !(x.DaXoa ?? false) && x.MaTieuChuan == maT...

124 │ var list\_deleted = list\_old.Where(t => !lstChiTiet.Select(x => x.Ma).Contains(t.Ma))

125 │ .Select(t => { t.DaXoa = true; t.NgayXoa = DateTime.Now; t.NguoiXoa = ob...

126 │

127 │ if (list\_deleted != null && list\_deleted.Count > 0)

128 │ {

129 │ dbcontext.UpdateRange(list\_deleted);

130 │ dbcontext.SaveChanges();

131 │ }

132 │

133 │ var list\_update = (from lst\_moi in lstChiTiet

134 │ join lst\_cu in list\_old on lst\_moi.Ma equals lst\_cu.Ma

135 │ select new { cu = lst\_cu, moi = lst\_moi }).ToList().Select(t =>

136 │ {

137 │ t.cu.No = t.moi.No;

138 │ t.cu.TestItem = t.moi.TestItem;

139 │ t.cu.Unit = t.moi.Unit;

140 │ t.cu.Specification = t.moi.Specification;

141 │ t.cu.Method = t.moi.Method;

142 │ return t.cu;

143 │ }).ToList();

144 │ if (list\_update != null && list\_update.Count > 0)

145 │ {

146 │ dbcontext.UpdateRange(list\_update);

147 │ dbcontext.SaveChanges();

148 │ }

149 │ var list\_insert = lstChiTiet.Where(t => !list\_old.Select(x => x.Ma).Contains(t.Ma)).Select(t =>

150 │ {

151 │ t.MaTieuChuan = maTieuChuan; t.NgayTao = DateTime.Now; t.NguoiTao = objToken.userid; return t;

152 │ }).ToList();

153 │

154 │ if (list\_insert != null && list\_insert.Count > 0)

155 │ {

156 │ dbcontext.AddRange(list\_insert);

157 │ dbcontext.SaveChanges();

158 │ }

159 │

160 │ transaction.Commit();

161 │ }

162 │ catch (Exception ex)

163 │ {

164 │ transaction.Rollback();

165 │ return Json(new ResponseModelError(ex.Message));

166 │ }

167 │ }

168 │

169 │ return Json(new ResponseModelSuccess(QLCLConst.Successfully));

170 │ }

171 │

172 │ [TokenUserCheckHTTP(Permission.View)]

173 │ [HttpGet("getListTieuChuan")]

174 │ public IActionResult getListTieuChuan()

175 │ {

176 │ var listData = dbcontext.bt\_TieuChuans.Where(x => !(x.DaXoa ?? false) && x.MaChiNhanh == objToken.branch).Or...

177 │ return Json(new ResponseModel(QLCLConst.Successfully, "", listData));

178 │ }

179 │ [TokenUserCheckHTTP(Permission.View)]

180 │ [HttpPost("getListData")]

181 │ public IActionResult getListData()

182 │ {

183 │

184 │ var danhsach = (from tc in dbcontext.bt\_TieuChuans.Where(x => !(x.DaXoa ?? false) && x.MaChiNhanh == objToke...

185 │ select new

186 │ {

187 │ ma = tc.Ma,

188 │ tieuChuan = tc.TieuChuan ?? "",

189 │ maSanPham = tc.MaSanPham,

190 │ maVatLieuBaoBi = tc.MaVatLieuBaoBi,

191 │ trangThai = tc.TrangThai ?? false

192 │ }).ToList();

193 │ var danhsachchitiet = (from ct in dbcontext.bt\_ChiTietTieuChuans.Where(x => !(x.DaXoa ?? false))

194 │ select new

195 │ {

196 │ ma = ct.Ma,

197 │ maTieuChuan = ct.MaTieuChuan,

198 │ no = ct.No,

199 │ testItem = ct.TestItem,

200 │ unit = ct.Unit,

201 │ specification = ct.Specification,

202 │ method = ct.Method

203 │ }).OrderBy(x=>x.no).ToList();

204 │ return Json(new ResponseModelSuccess(QLCLConst.Successfully, new { danhsach, danhsachchitiet }));

205 │ }

206 │

207 │ [TokenUserCheckHTTP(Permission.View)]

208 │ [HttpPost("getListHangHoa")]

209 │ public IActionResult getListHangHoa()

210 │ {

211 │ var listHangHoa = (from hh in crmContext.VHangHoa.Where(x => x.HangHoa ?? false)

212 │ select new

213 │ {

214 │ ma = hh.Ma,

215 │ ten = hh.Ten,

216 │ tentienganh = hh.TenTiengAnh ?? "",

217 │ code = hh.Code ?? "",

218 │ madvt = hh.MaDvt,

219 │ tendvt = hh.TenDvt,

220 │ tenloaihanghoa = hh.TenLoaiHangHoa

221 │ }).ToList();

222 │

223 │ return Json(new ResponseModelSuccess(QLCLConst.Successfully, listHangHoa));

224 │ }

225 │

226 │

227 │ }

228 │ }

────────────────────────────────────────────────────────────

**📄 TieuChuanViewModel.cs (CS, 13 lines)**

1 │ ﻿namespace QuanLyChatLuong.Models

2 │ {

3 │ public class TieuChuanViewModel

4 │ {

5 │ public int? KichThuoc { get; set; }

6 │ public string? VatLieuBaoBi { get; set; }

7 │ public string? TieuChuan { get; set; }

8 │ public string? TenSanPham { get; set; }

9 │ public string? TenSanPhamTiengAnh { get; set; }

10 │ public int? MaSanPham { get; set; }

11 │ public int? STT { get; set; }

12 │ }

13 │ }

────────────────────────────────────────────────────────────

**📄 TmpAccSoDu131.cs (CS, 21 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TmpAccSoDu131

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaKh { get; set; }

11 │

12 │ public string? NgayPhatSinh { get; set; }

13 │

14 │ public decimal? NoCuoiKyNt { get; set; }

15 │

16 │ public decimal? CoCuoiKyNt { get; set; }

17 │

18 │ public decimal? NoCuoiKy { get; set; }

19 │

20 │ public decimal? CoCuoiKy { get; set; }

21 │ }

────────────────────────────────────────────────────────────

**📄 TmpData.cs (CS, 15 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TmpData

7 │ {

8 │ public int? MaTtkh { get; set; }

9 │

10 │ public string? CauHoi { get; set; }

11 │

12 │ public string? Test { get; set; }

13 │

14 │ public string? Test3 { get; set; }

15 │ }

────────────────────────────────────────────────────────────

**📄 TmpData1.cs (CS, 97 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TmpData1

7 │ {

8 │ public int MaKh { get; set; }

9 │

10 │ public string? TenKh { get; set; }

11 │

12 │ public string? DiaChi { get; set; }

13 │

14 │ public string? HoTenNlh { get; set; }

15 │

16 │ public string? EmailNlh { get; set; }

17 │

18 │ public string? DienThoaiNlh { get; set; }

19 │

20 │ public string? LoaiKh { get; set; }

21 │

22 │ public string? NhomGd3 { get; set; }

23 │

24 │ public string LanMuaGanNhat { get; set; } = null!;

25 │

26 │ public int SoLanMuaHang { get; set; }

27 │

28 │ public int TongSanLuongDaMua { get; set; }

29 │

30 │ public int? NgayCongNo { get; set; }

31 │

32 │ public decimal? HanMucNo { get; set; }

33 │

34 │ public decimal? CongNoHienTai { get; set; }

35 │

36 │ public int SoNgayQuaHan { get; set; }

37 │

38 │ public int VuotHanMuc { get; set; }

39 │

40 │ public int? MaNguonKh { get; set; }

41 │

42 │ public string FullName { get; set; } = null!;

43 │

44 │ public string? NhomGd1 { get; set; }

45 │

46 │ public int MaBg { get; set; }

47 │

48 │ public string? SoBaoGia { get; set; }

49 │

50 │ public string LoaiBaoGia { get; set; } = null!;

51 │

52 │ public DateTime? NgayBaoGia { get; set; }

53 │

54 │ public int? ChotHopDong { get; set; }

55 │

56 │ public int? MaTrangThai { get; set; }

57 │

58 │ public string? GhiChuBg { get; set; }

59 │

60 │ public int? SoLuong { get; set; }

61 │

62 │ public decimal? DonGiaTaiKho { get; set; }

63 │

64 │ public decimal? DonGiaBan { get; set; }

65 │

66 │ public int? MaHd { get; set; }

67 │

68 │ public DateTime? NgayHopDong { get; set; }

69 │

70 │ public string? SoHopDong { get; set; }

71 │

72 │ public int? LoaiHopDong { get; set; }

73 │

74 │ public string? GhiChuHd { get; set; }

75 │

76 │ public int? SoLuongHd { get; set; }

77 │

78 │ public decimal? DonGiaTaiKhoHd { get; set; }

79 │

80 │ public decimal? DonGiaBanHd { get; set; }

81 │

82 │ public DateTime? NgayNhatKy { get; set; }

83 │

84 │ public string? SoNhatKy { get; set; }

85 │

86 │ public string? NoiDung { get; set; }

87 │

88 │ public string? XuLy { get; set; }

89 │

90 │ public string? KetQua { get; set; }

91 │

92 │ public string? ChiDaoBld { get; set; }

93 │

94 │ public string? LoaiThongTinNhan { get; set; }

95 │

96 │ public string? NguonThongTinNhan { get; set; }

97 │ }

────────────────────────────────────────────────────────────

**📄 TmpDataTongHopBaoGiaVaNhatKy.cs (CS, 83 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TmpDataTongHopBaoGiaVaNhatKy

7 │ {

8 │ public int? MaKh { get; set; }

9 │

10 │ public string? TenKh { get; set; }

11 │

12 │ public string? DiaChi { get; set; }

13 │

14 │ public string? HoTenNlh { get; set; }

15 │

16 │ public string? EmailNlh { get; set; }

17 │

18 │ public string? DienThoaiNlh { get; set; }

19 │

20 │ public string? LoaiKh { get; set; }

21 │

22 │ public string? NhomGd3 { get; set; }

23 │

24 │ public string? LanMuaGanNhat { get; set; }

25 │

26 │ public int SoLanMuaHang { get; set; }

27 │

28 │ public int TongSanLuongDaMua { get; set; }

29 │

30 │ public int? NgayCongNo { get; set; }

31 │

32 │ public decimal? HanMucNo { get; set; }

33 │

34 │ public decimal? CongNoHienTai { get; set; }

35 │

36 │ public int SoNgayQuaHan { get; set; }

37 │

38 │ public decimal? VuotHanMuc { get; set; }

39 │

40 │ public string? NguonKh { get; set; }

41 │

42 │ public string? FullName { get; set; }

43 │

44 │ public string? NhomGd1 { get; set; }

45 │

46 │ public int MaBg { get; set; }

47 │

48 │ public string? SoBaoGia { get; set; }

49 │

50 │ public string LoaiBaoGia { get; set; } = null!;

51 │

52 │ public DateTime? NgayBaoGia { get; set; }

53 │

54 │ public bool? ChotHopDong { get; set; }

55 │

56 │ public int? MaTrangThai { get; set; }

57 │

58 │ public string? TrangThai { get; set; }

59 │

60 │ public string? GhiChuBg { get; set; }

61 │

62 │ public int? SoLuong { get; set; }

63 │

64 │ public decimal? DonGiaTaiKho { get; set; }

65 │

66 │ public decimal? DonGiaBan { get; set; }

67 │

68 │ public int? MaHd { get; set; }

69 │

70 │ public DateTime? NgayHopDong { get; set; }

71 │

72 │ public string? SoHopDong { get; set; }

73 │

74 │ public string? LoaiHopDong { get; set; }

75 │

76 │ public string? GhiChuHd { get; set; }

77 │

78 │ public int? SoLuongHd { get; set; }

79 │

80 │ public decimal? DonGiaTaiKhoHd { get; set; }

81 │

82 │ public decimal? DonGiaBanHd { get; set; }

83 │ }

────────────────────────────────────────────────────────────

**📄 TmpTtkh.cs (CS, 13 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TmpTtkh

7 │ {

8 │ public int? MaTtkh { get; set; }

9 │

10 │ public string? CauHoi { get; set; }

11 │

12 │ public string? CauTraLoi { get; set; }

13 │ }

────────────────────────────────────────────────────────────

**📄 TranChiTietThanhToanVanChuyen.cs (CS, 41 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TranChiTietThanhToanVanChuyen

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaThanhToan { get; set; }

11 │

12 │ public int? MaVanChuyen { get; set; }

13 │

14 │ public int? MaHoaDonBanHang { get; set; }

15 │

16 │ public decimal? PhaiTra { get; set; }

17 │

18 │ public decimal? DaThanhToan { get; set; }

19 │

20 │ public decimal? ConLai { get; set; }

21 │

22 │ public int? MaTienTeTt { get; set; }

23 │

24 │ public decimal? TyGiaTt { get; set; }

25 │

26 │ public int? UserCreated { get; set; }

27 │

28 │ public DateTime? CreatedDate { get; set; }

29 │

30 │ public int? UserModified { get; set; }

31 │

32 │ public DateTime? ModifiedDate { get; set; }

33 │

34 │ public bool? Deleted { get; set; }

35 │

36 │ public int? UserDeleted { get; set; }

37 │

38 │ public DateTime? DeletedDate { get; set; }

39 │

40 │ public bool? TuDong { get; set; }

41 │ }

────────────────────────────────────────────────────────────

**📄 TranChiTietVanChuyen.cs (CS, 49 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TranChiTietVanChuyen

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaVanChuyen { get; set; }

11 │

12 │ public int? MaHangHoa { get; set; }

13 │

14 │ public int? MaLoaiBao { get; set; }

15 │

16 │ public int? SoLuongBao { get; set; }

17 │

18 │ public int? SoLuong { get; set; }

19 │

20 │ public decimal? DonGia { get; set; }

21 │

22 │ public decimal? ThanhTien { get; set; }

23 │

24 │ public decimal? Vat { get; set; }

25 │

26 │ public decimal? TienVat { get; set; }

27 │

28 │ public decimal? PhaiTra { get; set; }

29 │

30 │ public decimal? DaThanhToan { get; set; }

31 │

32 │ public decimal? ConLai { get; set; }

33 │

34 │ public string? GhiChu { get; set; }

35 │

36 │ public int? UserCreated { get; set; }

37 │

38 │ public DateTime? CreatedDate { get; set; }

39 │

40 │ public int? UserModified { get; set; }

41 │

42 │ public DateTime? ModifiedDate { get; set; }

43 │

44 │ public bool? Deleted { get; set; }

45 │

46 │ public int? UserDeleted { get; set; }

47 │

48 │ public DateTime? DeletedDate { get; set; }

49 │ }

────────────────────────────────────────────────────────────

**📄 TranDanhSachXe.cs (CS, 43 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TranDanhSachXe

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaNhaVanChuyen { get; set; }

11 │

12 │ public int? MaLoaiXe { get; set; }

13 │

14 │ public string? Ten { get; set; }

15 │

16 │ public string? SoXe { get; set; }

17 │

18 │ public decimal? TrongTai { get; set; }

19 │

20 │ public int? MaDvt { get; set; }

21 │

22 │ public string? TenTaiXe { get; set; }

23 │

24 │ public string? SoDienThoai { get; set; }

25 │

26 │ public int? MaDonViTinh { get; set; }

27 │

28 │ public bool? Status { get; set; }

29 │

30 │ public int? UserCreated { get; set; }

31 │

32 │ public DateTime? CreatedDate { get; set; }

33 │

34 │ public int? UserModified { get; set; }

35 │

36 │ public DateTime? ModifiedDate { get; set; }

37 │

38 │ public bool? Deleted { get; set; }

39 │

40 │ public int? UserDeleted { get; set; }

41 │

42 │ public DateTime? DeletedDate { get; set; }

43 │ }

────────────────────────────────────────────────────────────

**📄 TranGiaVanChuyen.cs (CS, 45 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TranGiaVanChuyen

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaTinhThanh { get; set; }

11 │

12 │ public int? LoaiXe { get; set; }

13 │

14 │ public decimal? TuTrongTaiXe { get; set; }

15 │

16 │ public decimal? DenTrongTaiXe { get; set; }

17 │

18 │ public int? MaDvttrongTai { get; set; }

19 │

20 │ public int? MaKh { get; set; }

21 │

22 │ public int? MaLoaiBao { get; set; }

23 │

24 │ public decimal? GiaVanChuyenVnd { get; set; }

25 │

26 │ public decimal? GiaVanChuyenUsd { get; set; }

27 │

28 │ public int? MaDvtgia { get; set; }

29 │

30 │ public bool? Status { get; set; }

31 │

32 │ public int? UserCreated { get; set; }

33 │

34 │ public DateTime? CreatedDate { get; set; }

35 │

36 │ public int? UserModified { get; set; }

37 │

38 │ public DateTime? ModifiedDate { get; set; }

39 │

40 │ public bool? Deleted { get; set; }

41 │

42 │ public int? UserDeleted { get; set; }

43 │

44 │ public DateTime? DeletedDate { get; set; }

45 │ }

────────────────────────────────────────────────────────────

**📄 TranLoaiXe.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TranLoaiXe

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public int? MaDvttheoTrongTai { get; set; }

13 │

14 │ public bool? Status { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 TranThanhToan.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TranThanhToan

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? ChungTuThanhToan { get; set; }

11 │

12 │ public int? MaNhaVanChuyen { get; set; }

13 │

14 │ public string? KyThanhToan { get; set; }

15 │

16 │ public DateTime? NgayThanhToan { get; set; }

17 │

18 │ public string? DienGiai { get; set; }

19 │

20 │ public int? UserCreated { get; set; }

21 │

22 │ public DateTime? CreatedDate { get; set; }

23 │

24 │ public int? UserModified { get; set; }

25 │

26 │ public DateTime? ModifiedDate { get; set; }

27 │

28 │ public bool? Deleted { get; set; }

29 │

30 │ public int? UserDeleted { get; set; }

31 │

32 │ public DateTime? DeletedDate { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 TranThanhToanVanChuyen.cs (CS, 57 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TranThanhToanVanChuyen

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaLoaiPhieu { get; set; }

13 │

14 │ public int? HinhThucTt { get; set; }

15 │

16 │ public string? SoPhieu { get; set; }

17 │

18 │ public int? MaNhaVanChuyen { get; set; }

19 │

20 │ public DateTime? NgayChungTu { get; set; }

21 │

22 │ public string? NguoiNopTien { get; set; }

23 │

24 │ public string? LyDo { get; set; }

25 │

26 │ public int? TienTe { get; set; }

27 │

28 │ public int? TyGia { get; set; }

29 │

30 │ public decimal? SoTienThanhToan { get; set; }

31 │

32 │ public int? TrangThai { get; set; }

33 │

34 │ public DateTime? NgayChuyenSoKt { get; set; }

35 │

36 │ public int? MaTaiKhoanNh { get; set; }

37 │

38 │ public string? TruongSo1 { get; set; }

39 │

40 │ public string? TruongSo2 { get; set; }

41 │

42 │ public int? UserCreated { get; set; }

43 │

44 │ public DateTime? CreatedDate { get; set; }

45 │

46 │ public int? UserModified { get; set; }

47 │

48 │ public DateTime? ModifiedDate { get; set; }

49 │

50 │ public bool? Deleted { get; set; }

51 │

52 │ public int? UserDeleted { get; set; }

53 │

54 │ public DateTime? DeletedDate { get; set; }

55 │

56 │ public int? KieuPhanBo { get; set; }

57 │ }

────────────────────────────────────────────────────────────

**📄 TranVanChuyen.cs (CS, 51 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TranVanChuyen

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaXuatKho { get; set; }

13 │

14 │ public int? MaLenhXuatKho { get; set; }

15 │

16 │ public int? MaDonHang { get; set; }

17 │

18 │ public int? MaHoaDonBh { get; set; }

19 │

20 │ public DateTime? NgayVanChuyen { get; set; }

21 │

22 │ public int? MaNhaVanChuyen { get; set; }

23 │

24 │ public int? MaTienTe { get; set; }

25 │

26 │ public int? TyGia { get; set; }

27 │

28 │ public decimal? PhaiTra { get; set; }

29 │

30 │ public decimal? DaThanhToan { get; set; }

31 │

32 │ public decimal? ConLai { get; set; }

33 │

34 │ public DateTime? NgayThanhToan { get; set; }

35 │

36 │ public int? NguoiThanhToan { get; set; }

37 │

38 │ public int? UserCreated { get; set; }

39 │

40 │ public DateTime? CreatedDate { get; set; }

41 │

42 │ public int? UserModified { get; set; }

43 │

44 │ public DateTime? ModifiedDate { get; set; }

45 │

46 │ public bool? Deleted { get; set; }

47 │

48 │ public int? UserDeleted { get; set; }

49 │

50 │ public DateTime? DeletedDate { get; set; }

51 │ }

────────────────────────────────────────────────────────────

**📄 TtkhBangDanhGia.cs (CS, 25 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TtkhBangDanhGia

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaKhachHang { get; set; }

11 │

12 │ public int? UserCreated { get; set; }

13 │

14 │ public DateTime? CreatedDate { get; set; }

15 │

16 │ public int? UserModified { get; set; }

17 │

18 │ public DateTime? ModifiedDate { get; set; }

19 │

20 │ public bool? Deleted { get; set; }

21 │

22 │ public int? UserDeleted { get; set; }

23 │

24 │ public DateTime? DeletedDate { get; set; }

25 │ }

────────────────────────────────────────────────────────────

**📄 TtkhCauHoiKh.cs (CS, 37 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TtkhCauHoiKh

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? MoTaNgan { get; set; }

11 │

12 │ public string? TenCauHoi { get; set; }

13 │

14 │ public string? CauTraLoi { get; set; }

15 │

16 │ public string? LoaiCauTraLoi { get; set; }

17 │

18 │ public string? PhuLuc { get; set; }

19 │

20 │ public bool? ChonNhieu { get; set; }

21 │

22 │ public bool? Status { get; set; }

23 │

24 │ public int? UserCreated { get; set; }

25 │

26 │ public DateTime? CreatedDate { get; set; }

27 │

28 │ public int? UserModified { get; set; }

29 │

30 │ public DateTime? ModifiedDate { get; set; }

31 │

32 │ public bool? Deleted { get; set; }

33 │

34 │ public int? UserDeleted { get; set; }

35 │

36 │ public DateTime? DeletedDate { get; set; }

37 │ }

────────────────────────────────────────────────────────────

**📄 TtkhChiTietBangDanhGia.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TtkhChiTietBangDanhGia

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaDanhGia { get; set; }

11 │

12 │ public int? MaNoiDungDanhGia { get; set; }

13 │

14 │ public int? MaDiemDanhGia { get; set; }

15 │

16 │ public int? UserCreated { get; set; }

17 │

18 │ public DateTime? CreatedDate { get; set; }

19 │

20 │ public int? UserModified { get; set; }

21 │

22 │ public DateTime? ModifiedDate { get; set; }

23 │

24 │ public bool? Deleted { get; set; }

25 │

26 │ public int? UserDeleted { get; set; }

27 │

28 │ public DateTime? DeletedDate { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 TtkhChiTietTtkh.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TtkhChiTietTtkh

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaTtkh { get; set; }

11 │

12 │ public string? CauHoi { get; set; }

13 │

14 │ public string? CauTraLoi { get; set; }

15 │

16 │ public bool? Status { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 TtkhDiaChiGiaoHang.cs (CS, 37 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TtkhDiaChiGiaoHang

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaKh { get; set; }

11 │

12 │ public int? MaQuocGia { get; set; }

13 │

14 │ public int? MaTinhThanh { get; set; }

15 │

16 │ public int? MaHuyenThi { get; set; }

17 │

18 │ public int? MaPhuogXa { get; set; }

19 │

20 │ public string? TenDuong { get; set; }

21 │

22 │ public bool? Status { get; set; }

23 │

24 │ public int? UserCreated { get; set; }

25 │

26 │ public DateTime? CreatedDate { get; set; }

27 │

28 │ public int? UserModified { get; set; }

29 │

30 │ public DateTime? ModifiedDate { get; set; }

31 │

32 │ public bool? Deleted { get; set; }

33 │

34 │ public int? UserDeleted { get; set; }

35 │

36 │ public DateTime? DeletedDate { get; set; }

37 │ }

────────────────────────────────────────────────────────────

**📄 TtkhDiemDanhGia.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TtkhDiemDanhGia

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaNoiDungDanhGia { get; set; }

11 │

12 │ public string? MoTa { get; set; }

13 │

14 │ public int? Diem { get; set; }

15 │

16 │ public bool? TrangThai { get; set; }

17 │

18 │ public int? UserCreated { get; set; }

19 │

20 │ public DateTime? CreatedDate { get; set; }

21 │

22 │ public int? UserModified { get; set; }

23 │

24 │ public DateTime? ModifiedDate { get; set; }

25 │

26 │ public bool? Deleted { get; set; }

27 │

28 │ public int? UserDeleted { get; set; }

29 │

30 │ public DateTime? DeletedDate { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 TtkhLoaiKh.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TtkhLoaiKh

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public bool? Status { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 TtkhNguoiLienHe.cs (CS, 39 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TtkhNguoiLienHe

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public string? SoDienThoai { get; set; }

15 │

16 │ public string? DiaChi { get; set; }

17 │

18 │ public string? TenTaiKhoan { get; set; }

19 │

20 │ public string? SoTaiKhoan { get; set; }

21 │

22 │ public string? NganHang { get; set; }

23 │

24 │ public bool? TrangThai { get; set; }

25 │

26 │ public int? UserCreated { get; set; }

27 │

28 │ public DateTime? CreatedDate { get; set; }

29 │

30 │ public int? UserModified { get; set; }

31 │

32 │ public DateTime? ModifiedDate { get; set; }

33 │

34 │ public bool? Deleted { get; set; }

35 │

36 │ public int? UserDeleted { get; set; }

37 │

38 │ public DateTime? DeletedDate { get; set; }

39 │ }

────────────────────────────────────────────────────────────

**📄 TtkhNguonKh.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TtkhNguonKh

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public bool? Status { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 TtkhNhomGiaoDich.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TtkhNhomGiaoDich

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public string? MoTa { get; set; }

13 │

14 │ public int? Nhom { get; set; }

15 │

16 │ public bool? XuatKhau { get; set; }

17 │

18 │ public bool? Status { get; set; }

19 │

20 │ public int? UserCreated { get; set; }

21 │

22 │ public DateTime? CreatedDate { get; set; }

23 │

24 │ public int? UserModified { get; set; }

25 │

26 │ public DateTime? ModifiedDate { get; set; }

27 │

28 │ public bool? Deleted { get; set; }

29 │

30 │ public int? UserDeleted { get; set; }

31 │

32 │ public DateTime? DeletedDate { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 TtkhNoiDungDanhGia.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TtkhNoiDungDanhGia

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public bool? TrangThai { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 TtkhThongTinKh.cs (CS, 125 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TtkhThongTinKh

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? MaKh { get; set; }

11 │

12 │ public string? TenKh { get; set; }

13 │

14 │ public string? Photo { get; set; }

15 │

16 │ public int? NhomGd1 { get; set; }

17 │

18 │ public int? NhomGd2 { get; set; }

19 │

20 │ public int? NhomGd3 { get; set; }

21 │

22 │ public int? MaLoaiKh { get; set; }

23 │

24 │ public string? TenDuong { get; set; }

25 │

26 │ public int? MaPhuongXa { get; set; }

27 │

28 │ public int? MaHuyenThi { get; set; }

29 │

30 │ public int? MaTinh { get; set; }

31 │

32 │ public int? MaQuocGia { get; set; }

33 │

34 │ public decimal? Latitude { get; set; }

35 │

36 │ public decimal? Longitude { get; set; }

37 │

38 │ public string? TenDuongGiaoHang { get; set; }

39 │

40 │ public int? MaPhuongXaGiaoHang { get; set; }

41 │

42 │ public int? MaHuyenThiGiaoHang { get; set; }

43 │

44 │ public int? MaTinhGiaoHang { get; set; }

45 │

46 │ public int? MaQuocGiaGiaoHang { get; set; }

47 │

48 │ public string? MaSoThue { get; set; }

49 │

50 │ public string? NguoiDaiDienPl { get; set; }

51 │

52 │ public string? ChucVu { get; set; }

53 │

54 │ public string? DienThoai { get; set; }

55 │

56 │ public string? Fax { get; set; }

57 │

58 │ public string? Email { get; set; }

59 │

60 │ public string? NganHang { get; set; }

61 │

62 │ public string? Stk { get; set; }

63 │

64 │ public string? HoTenNlh { get; set; }

65 │

66 │ public string? ChucVuNlh { get; set; }

67 │

68 │ public string? DienThoaiNlh { get; set; }

69 │

70 │ public string? FaxNlh { get; set; }

71 │

72 │ public string? EmailNlh { get; set; }

73 │

74 │ public string? HoTenKt { get; set; }

75 │

76 │ public string? ChucVuKt { get; set; }

77 │

78 │ public string? DienThoaiKt { get; set; }

79 │

80 │ public string? FaxKt { get; set; }

81 │

82 │ public string? EmailKt { get; set; }

83 │

84 │ public bool? ChiTietKh { get; set; }

85 │

86 │ public decimal? HanMucNo { get; set; }

87 │

88 │ public int? NgayCongNo { get; set; }

89 │

90 │ public int? NgayDiDuong { get; set; }

91 │

92 │ public int? MaNguonKh { get; set; }

93 │

94 │ public bool? KhachLe { get; set; }

95 │

96 │ public string? Cmnd { get; set; }

97 │

98 │ public int? MaNhanVienPhuTrach { get; set; }

99 │

100 │ public bool? DaDongBo { get; set; }

101 │

102 │ public bool? Status { get; set; }

103 │

104 │ public int? UserCreated { get; set; }

105 │

106 │ public DateTime? CreatedDate { get; set; }

107 │

108 │ public int? UserModified { get; set; }

109 │

110 │ public DateTime? ModifiedDate { get; set; }

111 │

112 │ public bool? Deleted { get; set; }

113 │

114 │ public int? UserDeleted { get; set; }

115 │

116 │ public DateTime? DeletedDate { get; set; }

117 │

118 │ public int? MaNguoiLienHe { get; set; }

119 │

120 │ public decimal? ChietKhauThuongMaiMin { get; set; }

121 │

122 │ public decimal? ChietKhauThuongMaiMax { get; set; }

123 │

124 │ public int? MaTienTe { get; set; }

125 │ }

────────────────────────────────────────────────────────────

**📄 TtkhThongTinKhachHangNguoiLienHe.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class TtkhThongTinKhachHangNguoiLienHe

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaKhachHang { get; set; }

11 │

12 │ public int? MaNguoiLienHe { get; set; }

13 │

14 │ public int? UserCreated { get; set; }

15 │

16 │ public DateTime? CreatedDate { get; set; }

17 │

18 │ public int? UserModified { get; set; }

19 │

20 │ public DateTime? ModifiedDate { get; set; }

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public int? UserDeleted { get; set; }

25 │

26 │ public DateTime? DeletedDate { get; set; }

27 │ }

────────────────────────────────────────────────────────────

**📄 UserController.cs (CS, 246 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │ using System.Linq;

4 │ using System.Runtime.CompilerServices;

5 │ using System.Threading.Tasks;

6 │ using Microsoft.AspNetCore.Mvc;

7 │ using Microsoft.AspNetCore.Mvc.Rendering;

8 │ using Microsoft.EntityFrameworkCore;

9 │ using Newtonsoft.Json.Linq;

10 │ using QuanLyChatLuong.Common;

11 │ using QuanLyChatLuong.Models;

12 │ using QuanLyChatLuong.Models.APIModels;

13 │

14 │ namespace QuanLyChatLuong.Controllers

15 │ {

16 │ [Route("sysUser")]

17 │ public class UserController : BaseController

18 │ {

19 │

20 │ public UserController(QuanLyChatLuongDbContext dbcontext, MemoryCacheManager memoryCache) : base(dbcontext,memor...

21 │ {

22 │ m\_MenuID = MenuID.sysUser;

23 │ }

24 │ public class UserModel

25 │ {

26 │ public string? ID { get; set; }

27 │ public string? eMa { get; set; }

28 │ public string? Username { get; set; }

29 │ public string? FullName { get; set; }

30 │ public string? GioiTinh { get; set; }

31 │ public string? MSNV { get; set; }

32 │ public string? Status { get; set; }

33 │ public string? Password { get; set; }

34 │ public string? NewPassword { get; set; }

35 │ public string? VerifyNewPassword { get; set; }

36 │

37 │ }

38 │

39 │ [HttpGet]

40 │ [TokenUserCheckHTTP]

41 │ public IActionResult Index()

42 │ {

43 │

44 │ ViewData["ListData"] = new sysUserModel(dbcontext).GetData();

45 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUser, (int)Permissio...

46 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUser, (int)Permissi...

47 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUser, (int)Permis...

48 │ ViewData["bResetPassword"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUser, (int...

49 │ ViewData["bViewChild"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUser, (int)Per...

50 │ ViewData["bUserEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUser, (int)Perm...

51 │ ViewData["bPhanQuyenRole"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUser, (int...

52 │ return View();

53 │ }

54 │ [HttpGet("add")]

55 │ [TokenUserCheckHTTP(Permission.Add)]

56 │ public IActionResult Add()

57 │ {

58 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUser, (int)Permissio...

59 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUser, (int)Permissi...

60 │ return View("Edit");

61 │ }

62 │ [HttpGet("edit")]

63 │ [TokenUserCheckHTTP(Permission.Edit)]

64 │ public IActionResult Edit(string ID)

65 │ {

66 │ int userID = Cast.ToInt(ID.Decrypt());

67 │ var user = new sysUserModel(dbcontext).GetItem(userID);

68 │ if (user == null)

69 │ {

70 │ return NotFound("User not found");

71 │ }

72 │

73 │ ViewData["item"] = user;

74 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUser, (int)Permissio...

75 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUser, (int)Permissi...

76 │ return View();

77 │ }

78 │

79 │ [HttpPost("addapi")]

80 │ [TokenUserCheckHTTP(Permission.Add)]

81 │ public IActionResult AddAPI([FromBody] UserModel model)

82 │ {

83 │ var res = Utility.CheckParam(model.Username, model.FullName, model.GioiTinh, model.Status);

84 │ if (res is not null)

85 │ {

86 │ return Json(res);

87 │ }

88 │ if (new sysUserModel(dbcontext).CheckUsername(model.Username, 0)) {

89 │ return Json(new ResponseModelError("Tên đăng nhập đã tồn tại."));

90 │ }

91 │ if (model.MSNV == null || model.MSNV.IsEmpty())

92 │ {

93 │ return Json(new ResponseModelError("Vui lòng nhập Mã số nhân viên!"));

94 │ }

95 │ else if (model.MSNV.Trim().Length > 30)

96 │ {

97 │ return Json(new ResponseModelError("Mã số nhân viên không đúng. Yêu cầu MSNV không được ...

98 │ }

99 │ else if (new sysUserModel(dbcontext).CheckMSNV(model.MSNV, 0))

100 │ {

101 │ return Json(new ResponseModelError("Mã số nhân viên đã tồn tại."));

102 │ }

103 │

104 │ string Password = string.Format("{0}@{1}", model.Username, DateTime.Now.Year);

105 │

106 │ int result = new sysUserModel(dbcontext).Add(model.Username, model.FullName, Password, Cast.ToInt(model.Gioi...

107 │ if (result != 0)

108 │ {

109 │ return Json(new ResponseModelSuccess("Thao tác thành công. Mật khẩu của bạn là " + Password)...

110 │ }

111 │ return Json(new ResponseModelError("Đã xảy ra lỗi"));

112 │ }

113 │ [HttpPost("updateapi")]

114 │ [TokenUserCheckHTTP(Permission.Edit)]

115 │ public IActionResult UpdateApi([FromBody] UserModel model)

116 │ {

117 │ var res = Utility.CheckParam(model.Username, model.FullName, model.GioiTinh, model.Status);

118 │ if (model.ID == null || model.ID.IsEmpty()) {

119 │ return Json(new ResponseModelError("Lỗi không tìm thấy nhân viên này!"));

120 │ }

121 │ if (res is not null)

122 │ {

123 │ return Json(res);

124 │ }

125 │ if (new sysUserModel(dbcontext).CheckUsername(model.Username, Cast.ToInt(model.ID)))

126 │ {

127 │ return Json(new ResponseModelError("Tên đăng nhập đã tồn tại."));

128 │ }

129 │ if (model.MSNV == null || model.MSNV.IsEmpty())

130 │ {

131 │ return Json(new ResponseModelError("Vui lòng nhập Mã số nhân viên!"));

132 │ }

133 │ else if (model.MSNV.Trim().Length > 30)

134 │ {

135 │ return Json(new ResponseModelError("Mã số nhân viên không đúng. Yêu cầu MSNV không được ...

136 │ }

137 │ else if (new sysUserModel(dbcontext).CheckMSNV(model.MSNV, Cast.ToInt(model.ID)))

138 │ {

139 │ return Json(new ResponseModelError("Mã số nhân viên đã tồn tại."));

140 │ }

141 │

142 │ int result = new sysUserModel(dbcontext).Edit(Cast.ToInt(model.ID), model.Username, model.FullName, Cast.ToI...

143 │ if (result != 0)

144 │ {

145 │ return Json(new ResponseModelSuccess("Cập nhật người dùng thành công!"));

146 │ }

147 │

148 │ return Json(new ResponseModelError("Đã xảy ra lỗi"));

149 │ }

150 │ [HttpPost("resetPassword")]

151 │ [TokenUserCheckHTTP(Permission.ResetPassword)]

152 │ public IActionResult ResetPassword([FromBody] UserModel model)

153 │ {

154 │ int result = new sysUserModel(dbcontext).ResetPassword(Cast.ToInt(model.ID.Decrypt()), model.Password);

155 │

156 │ if (result == 1)

157 │ return Json(new ResponseModelSuccess("Đặt lại mật khẩu thành công"));

158 │ else if (result == 0)

159 │ return Json(new ResponseModelError("Đặt lại mật khẩu thất bại"));

160 │ else if (result == -1)

161 │ Json(QLCLConst.ResponseNotExist);

162 │ return Json(new ResponseModelError("Đã xảy ra lỗi"));

163 │ }

164 │ [HttpPost("changepassword")]

165 │ [TokenUserCheckHTTP]

166 │ public ActionResult ChangePassword([FromBody] UserModel model)

167 │ {

168 │ sys\_User drU = new sysUserModel(dbcontext).GetItem(Cast.ToInt(model.ID));

169 │

170 │ if (drU != null)

171 │ {

172 │ var res = Utility.CheckParam(model.Password, model.NewPassword, model.VerifyNewPassword);

173 │ if (res is not null)

174 │ {

175 │ return Json(res);

176 │ }

177 │

178 │ if (model.Password == "")

179 │ {

180 │ return Json(new ResponseModel(QLCLConst.Error, "Vui lòng nhập mật khẩu hiện tại"));

181 │ }

182 │ else

183 │ {

184 │ if (drU.Password != Utility.GetSHA512(model.Password))

185 │ {

186 │ return Json(new ResponseModel(QLCLConst.Error, "Mật khẩu hiện tại không đúng"));

187 │ }

188 │ }

189 │

190 │ if (model.NewPassword == "")

191 │ {

192 │ return Json(new ResponseModel(QLCLConst.Error, "Vui lòng nhập mật khẩu mới"));

193 │ }

194 │

195 │ if (model.VerifyNewPassword == "")

196 │ {

197 │ return Json(new ResponseModel(QLCLConst.Error, "Vui lòng nhập xác nhận mật khẩu"));

198 │ }

199 │ else

200 │ {

201 │ if (model.NewPassword != "" && model.VerifyNewPassword != "" && model.NewPassword != model.Verif...

202 │ {

203 │ return Json(new ResponseModel(QLCLConst.Error, "Mật khẩu mới và xác nhận mật khẩu ...

204 │ }

205 │ }

206 │

207 │

208 │ //..

209 │ string result = Cast.ToString(new sysUserModel(dbcontext).ChangePassword(Cast.ToInt(model.ID), model.Pas...

210 │

211 │ if (result == "1")//Success

212 │ return Json(new ResponseModel(QLCLConst.OK, QLCLConst.UpdateSuccessfully));

213 │ else if (result == "0")//Failure

214 │ return Json(new ResponseModel(QLCLConst.Error, QLCLConst.UpdateFailure));

215 │ else if (result == "-1")//Not exist

216 │ return Json(new ResponseModel(QLCLConst.Error, QLCLConst.NotExist));

217 │ }

218 │

219 │ return Json(new ResponseModel(QLCLConst.Error, "Đã xảy ra lỗi"));

220 │ }

221 │

222 │ [HttpPost("changepasswordapi")]

223 │ [TokenUserCheckHTTP]

224 │ public ActionResult ChangePasswordAPI([FromBody] UserModel model)

225 │ {

226 │ return ChangePassword(model);

227 │ }

228 │ [HttpPost("deleteapi")]

229 │ [TokenUserCheckHTTP(Permission.Delete)]

230 │ public IActionResult DeleteAPI([FromBody] UserModel model)

231 │ {

232 │ int result = new sysUserModel(dbcontext).Delete(Cast.ToInt(model.eMa), objToken!.userid);

233 │ if (result == 1)

234 │ return Json(QLCLConst.ResponseDeleteSuccessfully);

235 │ else if (result == 0)

236 │ return Json(QLCLConst.ResponseDeleteFailure);

237 │ else if (result == -1)

238 │ return Json(QLCLConst.ResponseCannotDelete);

239 │ else if (result == -2)

240 │ return Json(QLCLConst.ResponseNotExist);

241 │

242 │ return Json(new ResponseModelError("Đã xảy ra lỗi"));

243 │ }

244 │

245 │ }

246 │ }

────────────────────────────────────────────────────────────

**📄 UserRoleController.cs (CS, 101 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using QuanLyChatLuong.Common;

3 │ using QuanLyChatLuong.Models;

4 │

5 │ namespace QuanLyChatLuong.Controllers

6 │ {

7 │ [Route("sysUserRole")]

8 │ public class UserRoleController : BaseController

9 │ {

10 │ public UserRoleController(QuanLyChatLuongDbContext dbcontext, MemoryCacheManager memoryCache) : base(dbcontext, ...

11 │ {

12 │ m\_MenuID = MenuID.sysUserRole;

13 │ }

14 │ public class UserRoleForm

15 │ {

16 │ public string? ID { get; set; }

17 │ public string? RoleID { get; set; }

18 │ public string? Status { get; set; }

19 │ public string? Branch { get; set; }

20 │ public string? UserID { get; set; }

21 │

22 │ }

23 │ [HttpGet("detail")]

24 │ [TokenUserCheckHTTP]

25 │ public IActionResult Detail(string UserID)

26 │ {

27 │ ViewData["ListData"] = new sysUserRoleModel(dbcontext).GetDataFollowUserID(Cast.ToInt(UserID.Decrypt()));

28 │ ViewData["ListRole"] = new sysRoleModel(dbcontext).GetData();

29 │ ViewData["UserItem"] = new sysUserModel(dbcontext).GetItem(Cast.ToInt(UserID.Decrypt()));

30 │ ViewData["ListBranch"] = dbcontext.tbl\_ChiNhanhs.ToList();

31 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUserRole, (int)Permi...

32 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUserRole, (int)Perm...

33 │ ViewData["bDelete"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUserRole, (int)Pe...

34 │ return View("index");

35 │ }

36 │ [HttpGet("add")]

37 │ [TokenUserCheckHTTP(Permission.Add)]

38 │ public IActionResult Add(string UserID)

39 │ {

40 │ ViewData["UserItem"] = new sysUserModel(dbcontext).GetItem(Cast.ToInt(UserID.Decrypt()));

41 │ ViewData["ListRole"] = new sysRoleModel(dbcontext).GetData().Where(t => t.TrangThai ?? false).ToList();

42 │ bool bShowAll =Utility.checkPermissionIDModel(objToken.userid, (int)m\_MenuID, (int) Permission.ShowAll);

43 │ if (bShowAll)

44 │ ViewData["ListBranch"] = dbcontext.tbl\_ChiNhanhs.Where(t => !(t.DaXoa ?? false) && (t.TrangThai ?? false...

45 │ else

46 │ ViewData["ListBranch"] = dbcontext.tbl\_ChiNhanhs.Where(t => !(t.DaXoa ?? false) && (t.TrangThai ?? false...

47 │ && t.Ma == objToken.branch).ToList();

48 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUserRole, (int)Permi...

49 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUserRole, (int)Perm...

50 │ return View("Edit");

51 │ }

52 │ [HttpGet("edit")]

53 │ [TokenUserCheckHTTP(Permission.Edit)]

54 │ public IActionResult Edit(string ID)

55 │ {

56 │ sys\_UserRole? drUR = new sysUserRoleModel(dbcontext).GetItem(Cast.ToInt(ID.Decrypt()));

57 │ if(drUR == null)

58 │ {

59 │ return NotFound("UserRole not found");

60 │ }

61 │ ViewData["item"] = drUR;

62 │ ViewData["ListRole"] = new sysRoleModel(dbcontext).GetData().Where(t => t.TrangThai ?? false).ToList();

63 │ ViewData["bAdd"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUserRole, (int)Permi...

64 │ ViewData["bEdit"] = Utility.checkPermissionIDModel2((int)objToken.userid, (int)MenuID.sysUserRole, (int)Perm...

65 │ if (drUR != null)

66 │ ViewData["UserItem"] = new sysUserModel(dbcontext).GetItem(drUR.MaUser);

67 │ bool bShowAll = Utility.checkPermissionIDModel(objToken!.userid, (int) MenuID.sysUserRole,(int) Permission.S...

68 │ if (bShowAll)

69 │ ViewData["ListBranch"] = dbcontext.tbl\_ChiNhanhs.Where(t => !(t.DaXoa ?? false) && (t.TrangThai ?? false...

70 │ else

71 │ ViewData["ListBranch"] = dbcontext.tbl\_ChiNhanhs.Where(t => !(t.DaXoa ?? false) && (t.TrangThai ?? false...

72 │ && t.Ma == objToken!.branch).ToList();

73 │

74 │ return View("Edit");

75 │ }

76 │

77 │ [HttpPost("updateapi")]

78 │ [TokenUserCheckHTTP(Permission.Edit)]

79 │ public IActionResult UpdateAPI([FromBody] UserRoleForm model)

80 │ {

81 │ int result = new sysUserRoleModel(dbcontext).Edit(Cast.ToInt(model.ID), Cast.ToInt(model.RoleID), model.Stat...

82 │ return Json(Utility.MakeResponseFromEditResult(Cast.ToString(result)));

83 │ }

84 │

85 │ [HttpPost("addapi")]

86 │ [TokenUserCheckHTTP(Permission.Add)]

87 │ public IActionResult AddAPI([FromBody] UserRoleForm model)

88 │ {

89 │ int result = new sysUserRoleModel(dbcontext).Add(Cast.ToInt(model.UserID), Cast.ToInt(model.RoleID), model.S...

90 │ return Json(Utility.MakeResponseFromAddResult(Cast.ToString(result)));

91 │ }

92 │

93 │ [HttpPost("deleteapi")]

94 │ [TokenUserCheckHTTP(Permission.Delete)]

95 │ public IActionResult DeleteAPI([FromBody] UserRoleForm model)

96 │ {

97 │ string result = Cast.ToString(new sysUserRoleModel(dbcontext).Delete(Cast.ToInt(model.ID.Decrypt()), objToke...

98 │ return Json(Utility.MakeResponseFromDeleteResult(Cast.ToString(result)));

99 │ }

100 │ }

101 │ }

────────────────────────────────────────────────────────────

**📄 Utility.cs (CS, 831 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc.ViewFeatures;

2 │ using Newtonsoft.Json.Linq;

3 │ using Newtonsoft.Json;

4 │ using QuanLyChatLuong.Models.APIModels;

5 │ using QuanLyChatLuong.Models;

6 │ using System.ComponentModel;

7 │ using System.Data;

8 │ using System.Diagnostics;

9 │ using System.Drawing;

10 │ using System.Net.Mail;

11 │ using System.Reflection;

12 │ using System.Runtime.CompilerServices;

13 │ using System.Security.Cryptography.X509Certificates;

14 │ using System.Security.Cryptography;

15 │ using System.Text.RegularExpressions;

16 │ using System.Text;

17 │ using Microsoft.EntityFrameworkCore;

18 │ using QuanLyChatLuong.Models.CRMEntity;

19 │ using System.Buffers.Text;

20 │ using System.Numerics;

21 │

22 │

23 │ namespace QuanLyChatLuong.Common

24 │ {

25 │ class QLCLContextEntity : QuanLyChatLuongDbContext

26 │ {

27 │ public string connection { get; set; } = "";

28 │ protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder) => optionsBuilder.UseSqlServer(con...

29 │ }

30 │ public class Utility

31 │ {

32 │ private static string Key = "Qu@nlyC0ngv13c";

33 │ private static Random random = new Random();

34 │ private static QuanLyChatLuongDbContext \_QLCLContext;

35 │ public static int defaultUID = 0;

36 │ private static readonly List<string> ImageExt = new List<string> { ".JPG", ".JPE", ".BMP", ".GIF", ".PNG" };

37 │ private static readonly List<string> AllowDocumentExt = new List<string> { ".JPG", ".JPE", ".BMP", ".GIF", ".PNG...

38 │ , ".DOC", ".DOCX", ".XLS", ".XLSX", ".PDF", ".PPT", ".PPTX" };

39 │

40 │ private static readonly string base62 = "0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz";

41 │ public static void initConnection(string connection)

42 │ {

43 │ \_QLCLContext = new QLCLContextEntity()

44 │ {

45 │ connection = connection

46 │ };

47 │ }

48 │ public static string CheckPath(string path)

49 │ {

50 │ if (!Directory.Exists(path))

51 │ {

52 │ Directory.CreateDirectory(path);

53 │ }

54 │ return path;

55 │ }

56 │

57 │ public static bool IsEmail(string Email)

58 │ {

59 │ try

60 │ {

61 │ MailAddress m = new MailAddress(Email);

62 │ return true;

63 │ }

64 │ catch

65 │ {

66 │ return false;

67 │ }

68 │ }

69 │ public static T getViewData<T>(Microsoft.AspNetCore.Mvc.ViewFeatures.ViewDataDictionary<dynamic> d, string key)

70 │ {

71 │ try

72 │ {

73 │ return (T)(d[key] ?? Activator.CreateInstance<T>());

74 │ }

75 │ catch

76 │ {

77 │ return Activator.CreateInstance<T>();

78 │ }

79 │ }

80 │

81 │ public static bool IsNumber(string Text)

82 │ {

83 │ try

84 │ {

85 │ Convert.ToDecimal(Text);

86 │ return true;

87 │ }

88 │ catch

89 │ {

90 │ return false;

91 │ }

92 │ }

93 │

94 │ public static string Encrypt(string toEncrypt)

95 │ {

96 │ try

97 │ {

98 │ byte[] toEncryptArray = Encoding.UTF8.GetBytes(toEncrypt);

99 │ using var hash = MD5.Create();

100 │ using var tripleDes = TripleDES.Create();

101 │ tripleDes.Key = hash.ComputeHash(Encoding.UTF8.GetBytes(Key));

102 │ tripleDes.Mode = CipherMode.ECB;

103 │ tripleDes.Padding = PaddingMode.PKCS7;

104 │

105 │ ICryptoTransform cTransform = tripleDes.CreateEncryptor();

106 │ byte[] resultArray = cTransform.TransformFinalBlock(toEncryptArray, 0, toEncryptArray.Length);

107 │ return ToBase62(resultArray);

108 │ }

109 │ catch

110 │ {

111 │ return "Error";

112 │ }

113 │ }

114 │

115 │ public static string Decrypt(string toDecrypt)

116 │ {

117 │ try

118 │ {

119 │ byte[] toEncryptArray = FromBase62(toDecrypt ?? "");

120 │ using var hash = MD5.Create();

121 │ using var tripleDes = TripleDES.Create();

122 │ tripleDes.Key = hash.ComputeHash(Encoding.UTF8.GetBytes(Key));

123 │ tripleDes.Mode = CipherMode.ECB;

124 │ tripleDes.Padding = PaddingMode.PKCS7;

125 │

126 │ ICryptoTransform cTransform = tripleDes.CreateDecryptor();

127 │ byte[] resultArray = cTransform.TransformFinalBlock(toEncryptArray, 0, toEncryptArray.Length);

128 │ return Encoding.UTF8.GetString(resultArray);

129 │ }

130 │ catch

131 │ {

132 │ return "Error";

133 │ }

134 │ }

135 │ private static string ToBase62(byte[] bytes)

136 │ {

137 │ var num = new BigInteger(bytes.Append((byte)0).ToArray());

138 │ var sb = new StringBuilder();

139 │ while (num > 0)

140 │ {

141 │ sb.Insert(0, base62[(int)(num % 62)]);

142 │ num /= 62;

143 │ }

144 │ return sb.ToString();

145 │ }

146 │ private static byte[] FromBase62(string str)

147 │ {

148 │ BigInteger num = 0;

149 │ foreach (var c in str)

150 │ num = num \* 62 + base62.IndexOf(c);

151 │ var bytes = num.ToByteArray();

152 │ return bytes[^1] == 0 ? bytes[..^1] : bytes;

153 │ }

154 │ public static bool IsImage(string Extension)

155 │ {

156 │ try

157 │ {

158 │ if (ImageExt.Contains(Extension.ToUpper()))

159 │ return true;

160 │ return false;

161 │ }

162 │ catch

163 │ {

164 │ return false;

165 │ }

166 │ }

167 │

168 │ public static bool IsDocument(string Extension)

169 │ {

170 │ try

171 │ {

172 │ if (AllowDocumentExt.Contains(Extension.ToUpper()))

173 │ return true;

174 │ return false;

175 │ }

176 │ catch

177 │ {

178 │ return false;

179 │ }

180 │ }

181 │

182 │ public static bool AllowPreview(string Extension)

183 │ {

184 │ try

185 │ {

186 │ List<string> ListExt = new List<string> { "JPG", "JPE", "BMP", "GIF", "PNG", "PDF" };

187 │

188 │ if (ListExt.Contains(Extension.ToUpper()))

189 │ return true;

190 │ return false;

191 │ }

192 │ catch

193 │ {

194 │ return false;

195 │ }

196 │ }

197 │

198 │ public static string RandomNumber(int length)

199 │ {

200 │ const string chars = "0123456789";

201 │ return new string(Enumerable.Repeat(chars, length)

202 │ .Select(s => s[random.Next(s.Length)]).ToArray());

203 │ }

204 │

205 │ public static void sendRequestLogin(sys\_User User, string connectionID, string sign)

206 │ {

207 │ //\_ = mess.SendAsync(new Message

208 │ //{

209 │ // Token = User.Fcm,//User.FCM,

210 │ // Notification = new Notification

211 │ // {

212 │ // Title = "Cảnh báo bảo mật",

213 │ // Body = "Tài khoản của bạn đang được đăng nhập ở một thiết bị khác. Vui l�...

214 │ // ImageUrl = "https://fa.hungduy.vn/images/logo/logoTapDoan.jpg",

215 │ // },

216 │ // Data = new Dictionary<string, string>

217 │ // {

218 │ // { "action", "request\_login" },

219 │ // { "connectionId",connectionID },

220 │ // { "sign",sign }

221 │ // }

222 │ // //Android = new AndroidConfig

223 │ // //{

224 │ // // Notification = new AndroidNotification

225 │ // // {

226 │ // // Sound = "sound",

227 │ // // }

228 │ // //}

229 │ //});

230 │ }

231 │

232 │ public static (bool, string) checkPhone(string phone)

233 │ {

234 │ if (phone.StartsWith("0"))

235 │ {

236 │ phone = "+84" + phone.Substring(1);

237 │ }

238 │ if (phone.StartsWith("84"))

239 │ {

240 │ phone = "+" + phone;

241 │ }

242 │ //+84836406400

243 │ if (phone.Length != 12 || !phone.StartsWith("+84"))

244 │ {

245 │ return (false, "Số điện thoại không đúng định dạng");

246 │ }

247 │ return (true, phone);

248 │ }

249 │

250 │ public static string GetItemMultiSelect(object obj)

251 │ {

252 │ return string.Format("{0}", Cast.ToString(obj).Replace(";", ","));

253 │ }

254 │

255 │ public static string SetItemMultiSelect(object obj)

256 │ {

257 │ return string.Format(";{0};", Cast.ToString(obj).Replace(",", ";"));

258 │ }

259 │

260 │ public static string GetSHA256WithRSA(string dataToSign, string privateKeyData)

261 │ {

262 │ try

263 │ {

264 │ using var rsa = new RSACryptoServiceProvider();

265 │ rsa.ImportFromPem(privateKeyData);

266 │ var dataBytes = Encoding.UTF8.GetBytes(dataToSign);

267 │ var signatureBytes = rsa.SignData(dataBytes, CryptoConfig.MapNameToOID("SHA256"));

268 │ return Convert.ToBase64String(signatureBytes);

269 │ }

270 │ catch

271 │ {

272 │ return null;

273 │ }

274 │ }

275 │

276 │ public static bool VerifySHA256WithRSA(string originalData, string signature, string publicKeyData)

277 │ {

278 │ try

279 │ {

280 │

281 │ using var rsa = new RSACryptoServiceProvider();

282 │ if (publicKeyData.Contains("BEGIN CERTIFICATE"))

283 │ {

284 │ var cert = new X509Certificate(Encoding.UTF8.GetBytes(publicKeyData));

285 │ var publicKey = cert.GetPublicKey();

286 │ rsa.ImportRSAPublicKey(publicKey, out \_);

287 │ }

288 │ else

289 │ {

290 │ rsa.ImportFromPem(publicKeyData);

291 │ }

292 │ var dataBytes = Encoding.UTF8.GetBytes(originalData);

293 │ var signatureBytes = Convert.FromBase64String(signature);

294 │ return rsa.VerifyData(dataBytes, CryptoConfig.MapNameToOID("SHA256"), signatureBytes);

295 │ }

296 │ catch

297 │ {

298 │ return false;

299 │ }

300 │ }

301 │ public static string GetSHA1(string str)

302 │ {

303 │ return GetSHA(str, SHA1.Create());

304 │ }

305 │

306 │ public static string GetSHA256(string str)

307 │ {

308 │ return GetSHA(str, SHA256.Create());

309 │ }

310 │

311 │ public static string GetSHA512(string str)

312 │ {

313 │ return GetSHA(str, SHA512.Create());

314 │ }

315 │

316 │ public static string GetMD5(string str)

317 │ {

318 │ return GetSHA(str, MD5.Create());

319 │ }

320 │ private static string GetSHA(string str, HashAlgorithm hash)

321 │ {

322 │ try

323 │ {

324 │ string[] value = (from x in hash.ComputeHash(Encoding.UTF8.GetBytes(str))

325 │ select x.ToString("X2")).ToArray<string>();

326 │ return string.Join("", value).ToUpper();

327 │ }

328 │ catch

329 │ {

330 │ }

331 │ return "";

332 │ }

333 │ public static bool IsNull(object o)

334 │ {

335 │ return o == null || Convert.IsDBNull(o);

336 │ }

337 │ public static string Base64Encode(string plainText)

338 │ {

339 │ var plainTextBytes = Encoding.UTF8.GetBytes(plainText);

340 │ return Convert.ToBase64String(plainTextBytes);

341 │ }

342 │

343 │ public static string Base64Decode(string base64EncodedData)

344 │ {

345 │ var base64EncodedBytes = Convert.FromBase64String(base64EncodedData);

346 │ return Encoding.UTF8.GetString(base64EncodedBytes);

347 │ }

348 │ /\* public static bool checkTokenPermission(string token, int menu, int permission)

349 │ {

350 │ try

351 │ {

352 │ UserToken URtoken = new UserToken(JsonConvert.DeserializeObject<JObject>(token)!);

353 │ return checkTokenPermission(URtoken, menu, permission);

354 │ }

355 │ catch

356 │ {

357 │ return false;

358 │ }

359 │ }\*/

360 │ //public static bool checkTokenPermission(ViewDataDictionary<dynamic> ViewData, MenuID menu, Permission permissi...

361 │ //{

362 │ // return checkTokenPermission(ViewData, menu.GetEnumGuid(), permission.GetEnumGuid());

363 │ //}

364 │ public static bool checkTokenPermission(ViewDataDictionary<dynamic> ViewData, int menu, int permission)

365 │ {

366 │ try

367 │ {

368 │ sys\_User? currentUser = ViewData["CurrentUser"] as sys\_User;

369 │ if (currentUser == null)

370 │ {

371 │ return false;

372 │ }

373 │ return \_QLCLContext.vQuyens.Any(t => t.MaUser == currentUser.Ma && t.MaQuyen == permission && t.MaMenu =...

374 │ }

375 │ catch

376 │ {

377 │ return false;

378 │ }

379 │ }

380 │ public static bool checkTokenPermission(ViewDataDictionary<dynamic> ViewData, int permission)

381 │ {

382 │ try

383 │ {

384 │ sys\_User? currentUser = ViewData["CurrentUser"] as sys\_User;

385 │ int? currentMenu = (int)(ViewData["currentMenu"] ?? 0);

386 │ string JObjectString = ViewData["ur"]?.ToString() ?? "{}";

387 │ UserToken ur = new UserToken(JsonConvert.DeserializeObject<JObject>(JObjectString));

388 │

389 │

390 │ if (currentUser == null || currentMenu == 0)

391 │ {

392 │ return false;

393 │ }

394 │

395 │ var s = \_QLCLContext.vQuyens.Any(t => t.MaChiNhanh == ur.branch && t.MaUser == (ur.loginAs ? ur.idAs : u...

396 │ return s;

397 │ }

398 │ catch

399 │ {

400 │ return false;

401 │ }

402 │ return false;

403 │ }

404 │ public static bool checkTokenPermission(QuanLyChatLuongDbContext dbcontext, UserToken URtoken, int menu, int per...

405 │ {

406 │ return dbcontext.vQuyens.Any(t => t.MaChiNhanh == URtoken.branch && t.MaUser == (URtoken.loginAs ? URtoken.i...

407 │ return false;

408 │ }

409 │

410 │ public static bool CheckRole(ViewDataDictionary<dynamic> ViewData, int roleid)

411 │ {

412 │ sys\_User? currentUser = ViewData["CurrentUser"] as sys\_User;

413 │ int? currentMenu = (int)(ViewData["currentMenu"] ?? 0);

414 │ if (currentUser == null || currentMenu == 0)

415 │ {

416 │ return false;

417 │ }

418 │ return \_QLCLContext.vQuyens.Any(t => t.MaRole == roleid && t.MaMenu == currentMenu && t.MaUser == currentUse...

419 │ }

420 │ public static bool CheckRole(string token, int menu, int permission)

421 │ {

422 │ try

423 │ {

424 │ UserToken URtoken = new UserToken((JObject)JsonConvert.DeserializeObject(token)!);

425 │ return URtoken != null ? CheckRole(URtoken, menu, permission) : false;

426 │ }

427 │ catch

428 │ {

429 │ return false;

430 │ }

431 │

432 │ }

433 │ public static bool CheckRole(UserToken URtoken, int MenuID, int RoleID)

434 │ {

435 │ return \_QLCLContext.vQuyens.Any(t => t.MaRole == RoleID && t.MaMenu == MenuID && t.MaUser == URtoken.userid)...

436 │ }

437 │ public static ResponseModel? CheckParam(params object[] entities)

438 │ {

439 │ try

440 │ {

441 │ foreach (object x in entities)

442 │ {

443 │ if (x == null)

444 │ {

445 │ return new ResponseModelError("Vui lòng nhập đầy đủ dữ liệu");

446 │ }

447 │ if (x.GetType() == typeof(string) && ((string)x) == "")

448 │ {

449 │ return new ResponseModelError("Vui lòng nhập " + x.GetType().Name);

450 │ }

451 │ }

452 │ }

453 │ catch (Exception e)

454 │ {

455 │ return new ResponseModelError("Đã xảy ra lỗi khi kiểm tra biến " + e.Message);

456 │ }

457 │ return null;

458 │ }

459 │

460 │ private ResponseModel checkMethod(Delegate d)

461 │ {

462 │ try

463 │ {

464 │ ParameterInfo[] parameters = d.Method.GetParameters();

465 │ foreach (ParameterInfo parameter in parameters)

466 │ {

467 │ if (!parameter.HasDefaultValue)

468 │ {

469 │ Type t = parameter.GetType();

470 │ if (t == typeof(string))

471 │ {

472 │ //t.GetProperties().GetValue()

473 │ }

474 │ }

475 │ }

476 │ }

477 │ catch

478 │ {

479 │ }

480 │ return new ResponseModel();

481 │ }

482 │ public static ResponseModel MakeResponseFromAddResult(string result)

483 │ {

484 │ if (result == "1")//Success

485 │ {

486 │ return QLCLConst.ResponseAddSuccessfully;

487 │ }

488 │ else if (result == "0")//Failure

489 │ {

490 │ return QLCLConst.ResponseAddFailure;

491 │ }

492 │ else if (result == "-3")//Duplicate

493 │ {

494 │ return QLCLConst.ResponseObjectExists;

495 │ }

496 │ else if (result == "-1")//Duplicate

497 │ {

498 │ return QLCLConst.ResponseObjectExists;

499 │ }

500 │ else if (result == "-2")//Duplicate

501 │ {

502 │ return QLCLConst.ResponseObjectExists;

503 │ }

504 │ else if (result == "-4")//Duplicate

505 │ {

506 │ return QLCLConst.ResponseObjectExists;

507 │ }

508 │ else if (result == "-5")//Duplicate

509 │ {

510 │ return QLCLConst.ResponseObjectExists;

511 │ }

512 │ return QLCLConst.ResponseHaveError;

513 │ }

514 │ public static ResponseModel MakeResponseFromEditResult(string result)

515 │ {

516 │ if (result == "1")//Success

517 │ {

518 │ return QLCLConst.ResponseUpdateSuccessfully;

519 │ }

520 │

521 │ else if (result == "0")//Failure

522 │ {

523 │ return QLCLConst.ResponseUpdateFailure;

524 │ }

525 │

526 │ else if (result == "-2")//Not exist

527 │ {

528 │ return QLCLConst.ResponseNotExist;

529 │ }

530 │

531 │ else if (result == "-3")//Duplicate

532 │ {

533 │ return QLCLConst.ResponseObjectExists;

534 │ }

535 │

536 │ else if (result == "-4")//Duplicate

537 │ {

538 │ return QLCLConst.ResponseCodetCannotEdit;

539 │ }

540 │ return QLCLConst.ResponseHaveError;

541 │ }

542 │ public static ResponseModel MakeResponseFromDeleteResult(string result)

543 │ {

544 │ if (result == "1")//Success

545 │ {

546 │ return QLCLConst.ResponseDeleteSuccessfully;

547 │ }

548 │ else if (result == "0")//Failure

549 │ {

550 │ return QLCLConst.ResponseDeleteFailure;

551 │ }

552 │ else if (result == "-1")//Cannot delete

553 │ {

554 │ return QLCLConst.ResponseCannotDelete;

555 │ }

556 │ else if (result == "-2")//Not exist

557 │ {

558 │ return QLCLConst.ResponseNotExist;

559 │ }

560 │ else if (result == "-5")

561 │ {

562 │ return new ResponseModel(QLCLConst.Error, "Dữ liệu đang được sử dụng.");

563 │ }

564 │ return QLCLConst.ResponseHaveError;

565 │ }

566 │ public static string GetCodeDiaChi(string ten)

567 │ {

568 │ ten = BoDauTiengViet(ten);

569 │ var Reg = new Regex(@"^[0-9a-zA-Z/-]");

570 │ try

571 │ {

572 │ string[] tmps = ten.Split(' ');

573 │ string code = "";

574 │ for (int i = 0; i < tmps.Length; i++)

575 │ {

576 │ if (tmps[i].Trim() != "" && Reg.IsMatch(tmps[i].Trim()))

577 │ {

578 │ code += tmps[i].Trim()[0];

579 │ }

580 │ }

581 │ return code;

582 │ }

583 │ catch

584 │ {

585 │ return "";

586 │ }

587 │ }

588 │ public static DataTable ListToDataTable<T>(IList<T> data)

589 │ {

590 │ PropertyDescriptorCollection properties = TypeDescriptor.GetProperties(typeof(T));

591 │ DataTable table = new DataTable();

592 │ DataRow row;

593 │

594 │ foreach (PropertyDescriptor prop in properties)

595 │ table.Columns.Add(prop.Name, Nullable.GetUnderlyingType(prop.PropertyType) ?? prop.PropertyType);

596 │

597 │ foreach (T item in data)

598 │ {

599 │ row = table.NewRow();

600 │ foreach (PropertyDescriptor prop in properties)

601 │ row[prop.Name] = prop.GetValue(item) ?? DBNull.Value;

602 │ table.Rows.Add(row);

603 │ }

604 │

605 │ return table;

606 │ }

607 │ public static string NumberToText(double inputNumber, bool suffix = true)

608 │ {

609 │ string[] unitNumbers = new string[] { "không", "một", "hai", "ba", "bốn", "năm", "sáu", "bảy", "tá...

610 │ string[] placeValues = new string[] { "", "nghìn", "triệu", "tỷ" };

611 │ bool isNegative = false;

612 │

613 │ // -12345678.3445435 => "-12345678"

614 │ string sNumber = inputNumber.ToString("#");

615 │ if (sNumber == "")

616 │ {

617 │ sNumber = "0";

618 │

619 │ }

620 │ double number = Convert.ToDouble(sNumber);

621 │ if (number < 0)

622 │ {

623 │ number = -number;

624 │ sNumber = number.ToString();

625 │ isNegative = true;

626 │ }

627 │

628 │

629 │ int ones, tens, hundreds;

630 │

631 │ int positionDigit = sNumber.Length; // last -> first

632 │

633 │ string result = " ";

634 │

635 │

636 │ if (positionDigit == 0)

637 │ result = unitNumbers[0] + result;

638 │ else

639 │ {

640 │ // 0: ###

641 │ // 1: nghìn ###,###

642 │ // 2: triệu ###,###,###

643 │ // 3: tỷ ###,###,###,###

644 │ int placeValue = 0;

645 │

646 │ while (positionDigit > 0)

647 │ {

648 │ // Check last 3 digits remain ### (hundreds tens ones)

649 │ tens = hundreds = -1;

650 │ ones = Convert.ToInt32(sNumber.Substring(positionDigit - 1, 1));

651 │ positionDigit--;

652 │ if (positionDigit > 0)

653 │ {

654 │ tens = Convert.ToInt32(sNumber.Substring(positionDigit - 1, 1));

655 │ positionDigit--;

656 │ if (positionDigit > 0)

657 │ {

658 │ hundreds = Convert.ToInt32(sNumber.Substring(positionDigit - 1, 1));

659 │ positionDigit--;

660 │ }

661 │ }

662 │

663 │ if ((ones > 0) || (tens > 0) || (hundreds > 0) || (placeValue == 3))

664 │ result = placeValues[placeValue] + result;

665 │

666 │ placeValue++;

667 │ if (placeValue > 3) placeValue = 1;

668 │

669 │ if ((ones == 1) && (tens > 1))

670 │ result = "một " + result;

671 │ else

672 │ {

673 │ if ((ones == 5) && (tens > 0))

674 │ result = "lăm " + result;

675 │ else if (ones > 0)

676 │ result = unitNumbers[ones] + " " + result;

677 │ }

678 │ if (tens < 0)

679 │ break;

680 │ else

681 │ {

682 │ if ((tens == 0) && (ones > 0)) result = "lẻ " + result;

683 │ if (tens == 1) result = "mười " + result;

684 │ if (tens > 1) result = unitNumbers[tens] + " mươi " + result;

685 │ }

686 │ if (hundreds < 0) break;

687 │ else

688 │ {

689 │ if ((hundreds > 0) || (tens > 0) || (ones > 0))

690 │ result = unitNumbers[hundreds] + " trăm " + result;

691 │ }

692 │ result = " " + result;

693 │ }

694 │ }

695 │ result = result.Trim();

696 │ if (isNegative) result = "Âm " + result;

697 │ return result;

698 │ }

699 │ public static string HexToStringASCII(string hex)

700 │ {

701 │ hex = hex.Replace("-", "");

702 │ byte[] raw = new byte[hex.Length / 2];

703 │ for (int i = 0; i < raw.Length; i++)

704 │ {

705 │ raw[i] = Convert.ToByte(hex.Substring(i \* 2, 2), 16);

706 │ }

707 │ return Encoding.ASCII.GetString(raw);

708 │ }

709 │ public static string HexToStringUTF8(String hex)

710 │ {

711 │ int NumberChars = hex.Length / 2;

712 │ byte[] bytes = new byte[NumberChars];

713 │ using (var sr = new StringReader(hex))

714 │ {

715 │ for (int i = 0; i < NumberChars; i++)

716 │ bytes[i] =

717 │ Convert.ToByte(new string(new char[2] { (char)sr.Read(), (char)sr.Read() }), 16);

718 │ }

719 │ return Encoding.UTF8.GetString(bytes); ;

720 │ }

721 │ public static string BoDauTiengViet(string s)

722 │ {

723 │ Regex regex = new Regex("\\p{IsCombiningDiacriticalMarks}+");

724 │ string temp = (s + "").Normalize(NormalizationForm.FormD);

725 │ return regex.Replace(temp, String.Empty).Replace('\u0111', 'd').Replace('\u0110', 'D');

726 │ }

727 │ public static string ByteArrayToString(byte[] ba)

728 │ {

729 │ StringBuilder hex = new StringBuilder(ba.Length \* 2);

730 │ foreach (byte b in ba)

731 │ hex.AppendFormat("{0:x2}", b);

732 │ return hex.ToString();

733 │ }

734 │ public static string Window1252ToString(string s)

735 │ {

736 │ Encoding wind1252 = Encoding.GetEncoding(1252);

737 │ byte[] wind1252Bytes = wind1252.GetBytes(s);

738 │ string utf8String = ByteArrayToString(wind1252Bytes);

739 │ string res = HexToStringUTF8(utf8String);

740 │ return (res.Contains("?") || res.Contains("�")) ? s : res;

741 │ }

742 │

743 │

744 │ public static string ChuyenSoSangLaMa(int number)

745 │ {

746 │ if ((number < 0) || (number > 3999)) throw new ArgumentOutOfRangeException("insert value betwheen 1 and 3999...

747 │ if (number < 1) return string.Empty;

748 │ if (number >= 1000) return "M" + ChuyenSoSangLaMa(number - 1000);

749 │ if (number >= 900) return "CM" + ChuyenSoSangLaMa(number - 900);

750 │ if (number >= 500) return "D" + ChuyenSoSangLaMa(number - 500);

751 │ if (number >= 400) return "CD" + ChuyenSoSangLaMa(number - 400);

752 │ if (number >= 100) return "C" + ChuyenSoSangLaMa(number - 100);

753 │ if (number >= 90) return "XC" + ChuyenSoSangLaMa(number - 90);

754 │ if (number >= 50) return "L" + ChuyenSoSangLaMa(number - 50);

755 │ if (number >= 40) return "XL" + ChuyenSoSangLaMa(number - 40);

756 │ if (number >= 10) return "X" + ChuyenSoSangLaMa(number - 10);

757 │ if (number >= 9) return "IX" + ChuyenSoSangLaMa(number - 9);

758 │ if (number >= 5) return "V" + ChuyenSoSangLaMa(number - 5);

759 │ if (number >= 4) return "IV" + ChuyenSoSangLaMa(number - 4);

760 │ if (number >= 1) return "I" + ChuyenSoSangLaMa(number - 1);

761 │ throw new ArgumentOutOfRangeException("something bad happened");

762 │ }

763 │

764 │

765 │

766 │ public static async Task<bool> sendAsync(string log, string? controller = null, string? projectId = null)

767 │ {

768 │ StackTrace stackTrace = new StackTrace();

769 │ MethodBase method = stackTrace.GetFrame(1).GetMethod();

770 │ using HttpClient client = new HttpClient();

771 │ client.BaseAddress = new Uri("https://172.29.12.65/setLog");

772 │ client.DefaultRequestHeaders.Clear();

773 │ client.DefaultRequestHeaders.Host = "log.hungduy.vn";

774 │ return (await client.SendAsync(new HttpRequestMessage

775 │ {

776 │ Method = HttpMethod.Post,

777 │ Content = new StringContent(JsonConvert.SerializeObject(new

778 │ {

779 │ projectId = (projectId ?? method?.ReflectedType?.Namespace ?? string.Empty),

780 │ log = log,

781 │ controller = controller

782 │ }), Encoding.UTF8, "application/json")

783 │ })).IsSuccessStatusCode;

784 │ }

785 │

786 │ public static bool send(string log, string? controller = null, string? projectId = null)

787 │ {

788 │ string log2 = log;

789 │ string controller2 = controller;

790 │ string projectId2 = projectId;

791 │ bool kq = false;

792 │ Task.Run(async delegate

793 │ {

794 │ kq = await sendAsync(log2, controller2, projectId2);

795 │ }).Wait();

796 │ return kq;

797 │ }

798 │

799 │ public static string HtmlToPlainText(string html)

800 │ {

801 │ if (html == null) return "";

802 │ const string tagWhiteSpace = @"(>|$)(\W|\n|\r)+<";//matches one or more (white space or line breaks) between...

803 │ const string stripFormatting = @"<[^>]\*(>|$)";//match any character between '<' and '>', even when end tag i...

804 │ const string lineBreak = @"<(br|BR)\s{0,1}\/{0,1}>";//matches: <br>,<br/>,<br />,<BR>,<BR/>,<BR />

805 │ var lineBreakRegex = new Regex(lineBreak, RegexOptions.Multiline);

806 │ var stripFormattingRegex = new Regex(stripFormatting, RegexOptions.Multiline);

807 │ var tagWhiteSpaceRegex = new Regex(tagWhiteSpace, RegexOptions.Multiline);

808 │

809 │ var text = html;

810 │ //Decode html specific characters

811 │ text = System.Net.WebUtility.HtmlDecode(text);

812 │ //Remove tag whitespace/line breaks

813 │ text = tagWhiteSpaceRegex.Replace(text, "><");

814 │ //Replace <br /> with line breaks

815 │ text = lineBreakRegex.Replace(text, Environment.NewLine);

816 │ //Strip formatting

817 │ text = stripFormattingRegex.Replace(text, string.Empty);

818 │

819 │ return text;

820 │ }

821 │

822 │ internal static bool checkPermissionIDModel(int userid, int m\_MenuID, int permission)

823 │ {

824 │ return \_QLCLContext.vQuyens.Any(t => t.MaMenu == m\_MenuID && t.MaUser == userid && t.MaQuyen == permission);

825 │ }

826 │ internal static bool checkPermissionIDModel2(int userid, int m\_MenuID, int permission, int branchId)

827 │ {

828 │ return \_QLCLContext.vQuyens.Any(t => t.MaMenu == m\_MenuID && t.MaUser == userid && t.MaQuyen == permission &...

829 │ }

830 │ }

831 │ }

────────────────────────────────────────────────────────────

**📄 V1.cs (CS, 13 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class V1

7 │ {

8 │ public int? MaHopDongChinh { get; set; }

9 │

10 │ public int MaPhuLuc { get; set; }

11 │

12 │ public DateTime? CreatedDate { get; set; }

13 │ }

────────────────────────────────────────────────────────────

**📄 VBaoCaoDanhThu.cs (CS, 53 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VBaoCaoDanhThu

7 │ {

8 │ public long Ma { get; set; }

9 │

10 │ public int? Nam { get; set; }

11 │

12 │ public int? Thang { get; set; }

13 │

14 │ public int? MaKh { get; set; }

15 │

16 │ public int? MaHangHoa { get; set; }

17 │

18 │ public int? MaLoaiBao { get; set; }

19 │

20 │ public decimal? GiaTri { get; set; }

21 │

22 │ public int? SoLuong { get; set; }

23 │

24 │ public decimal? DonGiaTb { get; set; }

25 │

26 │ public decimal? DonGiaNytb { get; set; }

27 │

28 │ public string? CodeHangHoa { get; set; }

29 │

30 │ public string? TenHangHoa { get; set; }

31 │

32 │ public string? TenLoaiBao { get; set; }

33 │

34 │ public string? TenKh { get; set; }

35 │

36 │ public int? NhomGd1 { get; set; }

37 │

38 │ public string? TenNhomGd1 { get; set; }

39 │

40 │ public int? NhomGd2 { get; set; }

41 │

42 │ public string? TenNhomGd2 { get; set; }

43 │

44 │ public int? NhomGd3 { get; set; }

45 │

46 │ public string? TenNhomGd3 { get; set; }

47 │

48 │ public string Sale { get; set; } = null!;

49 │

50 │ public int? MaLoaiHangHoa { get; set; }

51 │

52 │ public bool? HangHoa { get; set; }

53 │ }

────────────────────────────────────────────────────────────

**📄 VBaoGia.cs (CS, 77 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VBaoGia

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public string? SoBaoGia { get; set; }

13 │

14 │ public int? MaKh { get; set; }

15 │

16 │ public int MaNhom { get; set; }

17 │

18 │ public int? MaNhaVanChuyen { get; set; }

19 │

20 │ public int? NgayCongNo { get; set; }

21 │

22 │ public DateTime? NgayBaoGia { get; set; }

23 │

24 │ public DateTime? TuNgay { get; set; }

25 │

26 │ public DateTime? DenNgay { get; set; }

27 │

28 │ public string? ThoiGianGiaoHang { get; set; }

29 │

30 │ public string? DiaDiemGiaoHang { get; set; }

31 │

32 │ public string? ThoiGianThanhToan { get; set; }

33 │

34 │ public string? GhiChu { get; set; }

35 │

36 │ public int? NguoiLienHe { get; set; }

37 │

38 │ public bool? ChotHopDong { get; set; }

39 │

40 │ public int? MaTrangThai { get; set; }

41 │

42 │ public int? MaTienTe { get; set; }

43 │

44 │ public int? MaTyGia { get; set; }

45 │

46 │ public int? TyGia { get; set; }

47 │

48 │ public int? MaNvphuTrach { get; set; }

49 │

50 │ public int? MaNvhoTro { get; set; }

51 │

52 │ public int? MaNhanVienPhuTrach { get; set; }

53 │

54 │ public int? TieuChuanChatLuong { get; set; }

55 │

56 │ public string? ChiPhiBocXep { get; set; }

57 │

58 │ public bool? RemindExpireDate { get; set; }

59 │

60 │ public bool? RemindSendReport { get; set; }

61 │

62 │ public int? UserCreated { get; set; }

63 │

64 │ public DateTime? CreatedDate { get; set; }

65 │

66 │ public int? UserModified { get; set; }

67 │

68 │ public DateTime? ModifiedDate { get; set; }

69 │

70 │ public bool? Deleted { get; set; }

71 │

72 │ public int? UserDeleted { get; set; }

73 │

74 │ public DateTime? DeletedDate { get; set; }

75 │

76 │ public string? TenNguoiLienHe { get; set; }

77 │ }

────────────────────────────────────────────────────────────

**📄 VChotLo.cs (CS, 53 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VChotLo

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? MaLo { get; set; }

11 │

12 │ public string? MaLoThanhPham { get; set; }

13 │

14 │ public int? MaChiNhanhTapioCa { get; set; }

15 │

16 │ public DateTime? NgayThuMua { get; set; }

17 │

18 │ public DateTime? NgayChotThuMua { get; set; }

19 │

20 │ public DateTime? NgayChot { get; set; }

21 │

22 │ public DateTime? NgayCapNhatChotLo { get; set; }

23 │

24 │ public decimal? TiLeThuHoi { get; set; }

25 │

26 │ public decimal? TiLeThuHoiQuiDoi { get; set; }

27 │

28 │ public int? GiaVon { get; set; }

29 │

30 │ public int? GiaVonTbthang { get; set; }

31 │

32 │ public int? TrangThai { get; set; }

33 │

34 │ public int? Tlsach { get; set; }

35 │

36 │ public int? Slnhap { get; set; }

37 │

38 │ public decimal SlbotPheQuiDoi { get; set; }

39 │

40 │ public decimal? TongGiaMua { get; set; }

41 │

42 │ public int? UserCreated { get; set; }

43 │

44 │ public DateTime? CreatedDate { get; set; }

45 │

46 │ public int? UserModified { get; set; }

47 │

48 │ public DateTime? ModifiedDate { get; set; }

49 │

50 │ public bool? Deleted { get; set; }

51 │

52 │ public DateTime? DeletedDate { get; set; }

53 │ }

────────────────────────────────────────────────────────────

**📄 VDoanhThu.cs (CS, 99 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VDoanhThu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaHopDong { get; set; }

11 │

12 │ public int? MaChiNhanh { get; set; }

13 │

14 │ public int? MaPhongBan { get; set; }

15 │

16 │ public DateTime? NgayChungTu { get; set; }

17 │

18 │ public string? SoHopDong { get; set; }

19 │

20 │ public string? SoChungTu { get; set; }

21 │

22 │ public int? MaKh { get; set; }

23 │

24 │ public int MaNhom { get; set; }

25 │

26 │ public int? MaNvbh { get; set; }

27 │

28 │ public DateTime? NgayHetHanThanhToan { get; set; }

29 │

30 │ public int? SoLuong { get; set; }

31 │

32 │ public string? TenDv { get; set; }

33 │

34 │ public int? SoLuongBao { get; set; }

35 │

36 │ public int? SoLuongThucXuat { get; set; }

37 │

38 │ public int? SlbaoThucXuat { get; set; }

39 │

40 │ public string? Ten { get; set; }

41 │

42 │ public decimal? PhaiThu { get; set; }

43 │

44 │ public decimal SoTienThanhToan { get; set; }

45 │

46 │ public decimal? ThanhTien { get; set; }

47 │

48 │ public decimal? TienThue { get; set; }

49 │

50 │ public decimal? ConLai { get; set; }

51 │

52 │ public bool? Deleted { get; set; }

53 │

54 │ public bool? DaXuat { get; set; }

55 │

56 │ public int? TrangThaiDonHang { get; set; }

57 │

58 │ public int? MaTienTe { get; set; }

59 │

60 │ public string? TenTienTe { get; set; }

61 │

62 │ public string? TenTrangThaiDonHang { get; set; }

63 │

64 │ public int? TyGia { get; set; }

65 │

66 │ public bool? DonHangChoThanhToan { get; set; }

67 │

68 │ public DateTime? NgayKetThuc { get; set; }

69 │

70 │ public string? SoPo { get; set; }

71 │

72 │ public DateTime? CreatedDate { get; set; }

73 │

74 │ public string? NhaVanChuyen { get; set; }

75 │

76 │ public int? MaNhaVanChuyen { get; set; }

77 │

78 │ public int? MaChiNhanhXuat { get; set; }

79 │

80 │ public string? TenChiNhanhXuat { get; set; }

81 │

82 │ public int? MaCnxuatHoaDon { get; set; }

83 │

84 │ public string? TenCnxuatHoaDon { get; set; }

85 │

86 │ public int? MaLoaiHangHoa { get; set; }

87 │

88 │ public int? MaLyDoDuyet { get; set; }

89 │

90 │ public string? TenLyDoDuyet { get; set; }

91 │

92 │ public string? LyDoDuyet { get; set; }

93 │

94 │ public int? MaDonHangTraLai { get; set; }

95 │

96 │ public int? MaNguoiLienHe { get; set; }

97 │

98 │ public string? TenNguoiLienHe { get; set; }

99 │ }

────────────────────────────────────────────────────────────

**📄 VHangHoa.cs (CS, 35 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VHangHoa

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public int? MaDvt { get; set; }

15 │

16 │ public int? DonGia { get; set; }

17 │

18 │ public decimal? DonGiaUsd { get; set; }

19 │

20 │ public bool? HangHoa { get; set; }

21 │

22 │ public int? MaLoaiHangHoa { get; set; }

23 │

24 │ public string? KhoXuat { get; set; }

25 │

26 │ public int? MaNhaVanChuyen { get; set; }

27 │

28 │ public string? TenLoaiHangHoa { get; set; }

29 │

30 │ public string? TenDvt { get; set; }

31 │

32 │ public bool? Status { get; set; }

33 │

34 │ public string? TenTiengAnh { get; set; }

35 │ }

────────────────────────────────────────────────────────────

**📄 VHoaDon.cs (CS, 65 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VHoaDon

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanhBan { get; set; }

11 │

12 │ public int? MaChiNhanhXuat { get; set; }

13 │

14 │ public string? SoHoaDon { get; set; }

15 │

16 │ public DateTime? NgayHoaDon { get; set; }

17 │

18 │ public string? SoDonHang { get; set; }

19 │

20 │ public int? MaKh { get; set; }

21 │

22 │ public string? TenKh { get; set; }

23 │

24 │ public string? TenNvbh { get; set; }

25 │

26 │ public DateTime? NgayHetHanTt { get; set; }

27 │

28 │ public int? MaTienTe { get; set; }

29 │

30 │ public string? TenTienTe { get; set; }

31 │

32 │ public DateTime? NgayNhanHang { get; set; }

33 │

34 │ public int? TyGia { get; set; }

35 │

36 │ public decimal? TienThue { get; set; }

37 │

38 │ public decimal? ThanhTien { get; set; }

39 │

40 │ public decimal? PhaiThu { get; set; }

41 │

42 │ public decimal? DaThanhToan { get; set; }

43 │

44 │ public decimal? ConLai { get; set; }

45 │

46 │ public int? MaLoaiHangHoa { get; set; }

47 │

48 │ public string? SoPhieuXuat { get; set; }

49 │

50 │ public string? DienThoai { get; set; }

51 │

52 │ public int? TrangThai { get; set; }

53 │

54 │ public int? LoaiHoaDon { get; set; }

55 │

56 │ public int? MaDonHang { get; set; }

57 │

58 │ public string? GhiChu { get; set; }

59 │

60 │ public int? NgayCongNo { get; set; }

61 │

62 │ public int? NhomGd1 { get; set; }

63 │

64 │ public DateTime? CreatedDate { get; set; }

65 │ }

────────────────────────────────────────────────────────────

**📄 VHoaDonMuaHang.cs (CS, 39 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VHoaDonMuaHang

7 │ {

8 │ public int MaChiNhanh { get; set; }

9 │

10 │ public string TenCn { get; set; } = null!;

11 │

12 │ public int Ma { get; set; }

13 │

14 │ public int? MaLoaiHoaDon { get; set; }

15 │

16 │ public string? SoHoaDon { get; set; }

17 │

18 │ public DateTime? NgayHoaDon { get; set; }

19 │

20 │ public int? MaNhaCungCap { get; set; }

21 │

22 │ public string TenNhaCc { get; set; } = null!;

23 │

24 │ public int? SoLuong { get; set; }

25 │

26 │ public decimal? ThanhTien { get; set; }

27 │

28 │ public string? GhiChu { get; set; }

29 │

30 │ public int? TrangThai { get; set; }

31 │

32 │ public decimal DaThanhToan { get; set; }

33 │

34 │ public decimal? ConLai { get; set; }

35 │

36 │ public string? SoDonHang { get; set; }

37 │

38 │ public string? SoDeNghi { get; set; }

39 │ }

────────────────────────────────────────────────────────────

**📄 VHopDong.cs (CS, 77 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VHopDong

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? LoaiHopDong { get; set; }

13 │

14 │ public string? SoHopDong { get; set; }

15 │

16 │ public int? MaKh { get; set; }

17 │

18 │ public int MaNhom { get; set; }

19 │

20 │ public int? MaBaoGia { get; set; }

21 │

22 │ public string? SoHopDongDoiTac { get; set; }

23 │

24 │ public DateTime? NgayHopDong { get; set; }

25 │

26 │ public int? PhapNhan { get; set; }

27 │

28 │ public DateTime? NgayBatDau { get; set; }

29 │

30 │ public DateTime? NgayKetThuc { get; set; }

31 │

32 │ public DateTime? NgayGiaHan { get; set; }

33 │

34 │ public int? MaTienTe { get; set; }

35 │

36 │ public string? GhiChu { get; set; }

37 │

38 │ public int? MaTyGia { get; set; }

39 │

40 │ public int? TyGia { get; set; }

41 │

42 │ public bool? XuatKhau { get; set; }

43 │

44 │ public string? FileDinhKem { get; set; }

45 │

46 │ public bool? XuatHang { get; set; }

47 │

48 │ public bool? HoanThanh { get; set; }

49 │

50 │ public int? TrangThaiHopDong { get; set; }

51 │

52 │ public bool? RemindExpireDate { get; set; }

53 │

54 │ public bool? RemindSendReport { get; set; }

55 │

56 │ public string? CangXuatDen { get; set; }

57 │

58 │ public int? UserCreated { get; set; }

59 │

60 │ public DateTime? CreatedDate { get; set; }

61 │

62 │ public int? UserModified { get; set; }

63 │

64 │ public DateTime? ModifiedDate { get; set; }

65 │

66 │ public bool? Deleted { get; set; }

67 │

68 │ public int? UserDeleted { get; set; }

69 │

70 │ public DateTime? DeletedDate { get; set; }

71 │

72 │ public int? NguoiLienHe { get; set; }

73 │

74 │ public string? TenNguoiLienHe { get; set; }

75 │

76 │ public int? MaHopDongChinh { get; set; }

77 │ }

────────────────────────────────────────────────────────────

**📄 VKhoHang.cs (CS, 15 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VKhoHang

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? TenKho { get; set; }

11 │

12 │ public int? MaChiNhanh { get; set; }

13 │

14 │ public bool? Status { get; set; }

15 │ }

────────────────────────────────────────────────────────────

**📄 VLenhXuatKho.cs (CS, 69 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VLenhXuatKho

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? SoChungTu { get; set; }

11 │

12 │ public int? MaChiNhanh { get; set; }

13 │

14 │ public string? Name { get; set; }

15 │

16 │ public int? MaKh { get; set; }

17 │

18 │ public string? TenKh { get; set; }

19 │

20 │ public DateTime? NgayChungTu { get; set; }

21 │

22 │ public string? GhiChu { get; set; }

23 │

24 │ public string? NhaVc { get; set; }

25 │

26 │ public string? SoXe { get; set; }

27 │

28 │ public int? SoLuong { get; set; }

29 │

30 │ public string? TenDv { get; set; }

31 │

32 │ public int? SoLuongBao { get; set; }

33 │

34 │ public string? Ten { get; set; }

35 │

36 │ public int? MaDonHang { get; set; }

37 │

38 │ public string? SoDonHang { get; set; }

39 │

40 │ public bool? Deleted { get; set; }

41 │

42 │ public string? LenhXuatKho { get; set; }

43 │

44 │ public int MaLenhXuatKho { get; set; }

45 │

46 │ public int? TrangThai { get; set; }

47 │

48 │ public string? TenTrangThai { get; set; }

49 │

50 │ public int? MaNhaVanChuyen { get; set; }

51 │

52 │ public int? MaChiNhanhXuat { get; set; }

53 │

54 │ public DateTime? CreatedDate { get; set; }

55 │

56 │ public string? GhiChuBaoMat { get; set; }

57 │

58 │ public decimal? ThuTienMat { get; set; }

59 │

60 │ public int? NhomGd1 { get; set; }

61 │

62 │ public int? NhomGd2 { get; set; }

63 │

64 │ public int? TrangThaiDonHang { get; set; }

65 │

66 │ public DateTime? ModifiedDate { get; set; }

67 │

68 │ public int? MaLoaiHangHoa { get; set; }

69 │ }

────────────────────────────────────────────────────────────

**📄 VLoaiXe.cs (CS, 15 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VLoaiXe

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Ten { get; set; }

11 │

12 │ public int? MaDvttheoTrongTai { get; set; }

13 │

14 │ public string? TenDvt { get; set; }

15 │ }

────────────────────────────────────────────────────────────

**📄 VNhapKho.cs (CS, 59 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VNhapKho

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public string Code { get; set; } = null!;

13 │

14 │ public string TenCn { get; set; } = null!;

15 │

16 │ public string? SoPhieu { get; set; }

17 │

18 │ public DateTime? NgayNhapXuat { get; set; }

19 │

20 │ public int? MaCa { get; set; }

21 │

22 │ public string? CaSx { get; set; }

23 │

24 │ public string? NguoiPhuTrach { get; set; }

25 │

26 │ public int? MaHangHoa { get; set; }

27 │

28 │ public string? TenHh { get; set; }

29 │

30 │ public int? MaKho { get; set; }

31 │

32 │ public string? TenKho { get; set; }

33 │

34 │ public int? MaLineKho { get; set; }

35 │

36 │ public string? ViTri { get; set; }

37 │

38 │ public string? LoSanXuat { get; set; }

39 │

40 │ public int? MaLoaiBao { get; set; }

41 │

42 │ public string? LoaiBao { get; set; }

43 │

44 │ public int? MaLoaiBot { get; set; }

45 │

46 │ public string? LoaiBot { get; set; }

47 │

48 │ public int? SoLuongBao { get; set; }

49 │

50 │ public DateTime? NgaySanXuat { get; set; }

51 │

52 │ public DateTime? NgayBatDauSx { get; set; }

53 │

54 │ public DateTime? NgayNgungSx { get; set; }

55 │

56 │ public DateTime? NgayDongBao { get; set; }

57 │

58 │ public DateTime? NgaySanXuatTrenBao { get; set; }

59 │ }

────────────────────────────────────────────────────────────

**📄 VPermission.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VPermission

7 │ {

8 │ public int RoleId { get; set; }

9 │

10 │ public int MenuId { get; set; }

11 │

12 │ public int? PermissionId { get; set; }

13 │

14 │ public int UserId { get; set; }

15 │

16 │ public string Username { get; set; } = null!;

17 │

18 │ public string? Email { get; set; }

19 │

20 │ public string? DepartmentId { get; set; }

21 │

22 │ public string? CodeCv { get; set; }

23 │

24 │ public bool EmailReceive { get; set; }

25 │

26 │ public string FullName { get; set; } = null!;

27 │

28 │ public bool? AllowAccess { get; set; }

29 │

30 │ public int BranchId { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 VPhieuNhapKho.cs (CS, 55 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VPhieuNhapKho

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? SoPhieu { get; set; }

11 │

12 │ public int? MaChiNhanh { get; set; }

13 │

14 │ public DateTime? NgayNhapXuat { get; set; }

15 │

16 │ public int? MaCa { get; set; }

17 │

18 │ public string? NguoiPhuTrach { get; set; }

19 │

20 │ public string? GhiChu { get; set; }

21 │

22 │ public int? MaLoaiNx { get; set; }

23 │

24 │ public bool? NhapKho { get; set; }

25 │

26 │ public int? DaXuatKho { get; set; }

27 │

28 │ public bool? Deleted { get; set; }

29 │

30 │ public string? CaSanXuat { get; set; }

31 │

32 │ public int? MaDonHang { get; set; }

33 │

34 │ public string? SoDonHang { get; set; }

35 │

36 │ public int? MaPhieuXuat { get; set; }

37 │

38 │ public int? MaNcc { get; set; }

39 │

40 │ public string? TenNhaCc { get; set; }

41 │

42 │ public int? MaLo { get; set; }

43 │

44 │ public string? TenLo { get; set; }

45 │

46 │ public int? TrangThaiLo { get; set; }

47 │

48 │ public int? MaGhepLo { get; set; }

49 │

50 │ public bool? DaKetThuc { get; set; }

51 │

52 │ public int? MaPhieuDeXuat { get; set; }

53 │

54 │ public string? SoPhieuDeNghi { get; set; }

55 │ }

────────────────────────────────────────────────────────────

**📄 VPhieuXuatKho.cs (CS, 57 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VPhieuXuatKho

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? SoPhieu { get; set; }

11 │

12 │ public int? MaChiNhanh { get; set; }

13 │

14 │ public string? TenKh { get; set; }

15 │

16 │ public DateTime? NgayNhapXuat { get; set; }

17 │

18 │ public int? MaCa { get; set; }

19 │

20 │ public string? NguoiPhuTrach { get; set; }

21 │

22 │ public string? GhiChu { get; set; }

23 │

24 │ public int? MaLoaiNx { get; set; }

25 │

26 │ public string? SoXe { get; set; }

27 │

28 │ public int? SoLuong { get; set; }

29 │

30 │ public string? TenDv { get; set; }

31 │

32 │ public int? SoLuongBao { get; set; }

33 │

34 │ public string? TenLoaiBao { get; set; }

35 │

36 │ public int? MaDonHang { get; set; }

37 │

38 │ public string? SoDonHang { get; set; }

39 │

40 │ public bool? Deleted { get; set; }

41 │

42 │ public bool? DaNhapKho { get; set; }

43 │

44 │ public bool? DaKetThuc { get; set; }

45 │

46 │ public string? LenhXuatKho { get; set; }

47 │

48 │ public int? MaLenhXuatKho { get; set; }

49 │

50 │ public DateTime? NgayGiaoHang { get; set; }

51 │

52 │ public int? MaNcc { get; set; }

53 │

54 │ public string? TenNhaCc { get; set; }

55 │

56 │ public decimal? ChiPhiBocVac { get; set; }

57 │ }

────────────────────────────────────────────────────────────

**📄 VPhuLucHopDongGanNhat.cs (CS, 13 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VPhuLucHopDongGanNhat

7 │ {

8 │ public int? MaHopDongChinh { get; set; }

9 │

10 │ public int MaPhuLuc { get; set; }

11 │

12 │ public DateTime? CreatedDate { get; set; }

13 │ }

────────────────────────────────────────────────────────────

**📄 VQlclChiNhanh.cs (CS, 15 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VQlclChiNhanh

7 │ {

8 │ public string Code { get; set; } = null!;

9 │

10 │ public string Name { get; set; } = null!;

11 │

12 │ public int MaChiNhanh { get; set; }

13 │

14 │ public int Id { get; set; }

15 │ }

────────────────────────────────────────────────────────────

**📄 VQlclChiTietSoLieu.cs (CS, 61 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VQlclChiTietSoLieu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaSoLieu { get; set; }

11 │

12 │ public string? Gio { get; set; }

13 │

14 │ public string? Phut { get; set; }

15 │

16 │ public decimal? BDoTrang { get; set; }

17 │

18 │ public decimal? BDoAm { get; set; }

19 │

20 │ public decimal? BDoMin { get; set; }

21 │

22 │ public decimal? BDoPh { get; set; }

23 │

24 │ public decimal? BDoKeoCp { get; set; }

25 │

26 │ public decimal? BDoKeoBu { get; set; }

27 │

28 │ public decimal? BTapChat { get; set; }

29 │

30 │ public decimal? NdLo { get; set; }

31 │

32 │ public decimal? NdSay { get; set; }

33 │

34 │ public decimal? NdHoHoa { get; set; }

35 │

36 │ public decimal? NsDoPh { get; set; }

37 │

38 │ public decimal? BuDoAm { get; set; }

39 │

40 │ public string? Cepa { get; set; }

41 │

42 │ public decimal? XmsDoAm { get; set; }

43 │

44 │ public string? XmtCoBot { get; set; }

45 │

46 │ public string? GhiChu { get; set; }

47 │

48 │ public int? NguoiTao { get; set; }

49 │

50 │ public DateTime? NgayTao { get; set; }

51 │

52 │ public int? NguoiSua { get; set; }

53 │

54 │ public DateTime? NgaySua { get; set; }

55 │

56 │ public bool? DaXoa { get; set; }

57 │

58 │ public int? NguoiXoa { get; set; }

59 │

60 │ public DateTime? NgayXoa { get; set; }

61 │ }

────────────────────────────────────────────────────────────

**📄 VQlclGioiHanSoLieu.cs (CS, 45 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VQlclGioiHanSoLieu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaChiNhanh { get; set; }

11 │

12 │ public string Code { get; set; } = null!;

13 │

14 │ public string? Ten { get; set; }

15 │

16 │ public string? TenNgan { get; set; }

17 │

18 │ public string? TenZh { get; set; }

19 │

20 │ public string? Loai { get; set; }

21 │

22 │ public decimal? GiaTri { get; set; }

23 │

24 │ public string? Dvt { get; set; }

25 │

26 │ public string? TieuChuan { get; set; }

27 │

28 │ public string? GhiChu { get; set; }

29 │

30 │ public bool? TrangThai { get; set; }

31 │

32 │ public int? NguoiTao { get; set; }

33 │

34 │ public DateTime? NgayTao { get; set; }

35 │

36 │ public int? NguoiSua { get; set; }

37 │

38 │ public DateTime? NgaySua { get; set; }

39 │

40 │ public bool? DaXoa { get; set; }

41 │

42 │ public int? NguoiXoa { get; set; }

43 │

44 │ public DateTime? NgayXoa { get; set; }

45 │ }

────────────────────────────────────────────────────────────

**📄 VQlclSoLieu.cs (CS, 39 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VQlclSoLieu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaChiNhanh { get; set; }

11 │

12 │ public DateTime Ngay { get; set; }

13 │

14 │ public string? TruongCa { get; set; }

15 │

16 │ public string? Qc { get; set; }

17 │

18 │ public string? ThuKhoBot { get; set; }

19 │

20 │ public string? TruongBptp { get; set; }

21 │

22 │ public string? DotLo { get; set; }

23 │

24 │ public bool? TrangThai { get; set; }

25 │

26 │ public int? NguoiTao { get; set; }

27 │

28 │ public DateTime? NgayTao { get; set; }

29 │

30 │ public int? NguoiSua { get; set; }

31 │

32 │ public DateTime? NgaySua { get; set; }

33 │

34 │ public bool? DaXoa { get; set; }

35 │

36 │ public int? NguoiXoa { get; set; }

37 │

38 │ public DateTime? NgayXoa { get; set; }

39 │ }

────────────────────────────────────────────────────────────

**📄 VQuanLySale.cs (CS, 15 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VQuanLySale

7 │ {

8 │ public int? MaQuanLy { get; set; }

9 │

10 │ public int? MaNhanVienBh { get; set; }

11 │

12 │ public string? Msnv { get; set; }

13 │

14 │ public string FullName { get; set; } = null!;

15 │ }

────────────────────────────────────────────────────────────

**📄 VThanhToan.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VThanhToan

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public int? MaDoanhThu { get; set; }

13 │

14 │ public string? SoHopDong { get; set; }

15 │

16 │ public string? SoChungTu { get; set; }

17 │

18 │ public int? MaKh { get; set; }

19 │

20 │ public string? TenKh { get; set; }

21 │

22 │ public DateTime? NgayThanhToan { get; set; }

23 │

24 │ public decimal? SoTien { get; set; }

25 │

26 │ public bool? Status { get; set; }

27 │

28 │ public bool? Deleted { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 VThongTinKhachHangCongNo.cs (CS, 35 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VThongTinKhachHangCongNo

7 │ {

8 │ public int Makh { get; set; }

9 │

10 │ public string? MaKh1 { get; set; }

11 │

12 │ public int MaNhom { get; set; }

13 │

14 │ public string? TenKh { get; set; }

15 │

16 │ public int NgayCongNo { get; set; }

17 │

18 │ public int? MaNhanVienPhuTrach { get; set; }

19 │

20 │ public string FullName { get; set; } = null!;

21 │

22 │ public bool? Deleted { get; set; }

23 │

24 │ public bool? Status { get; set; }

25 │

26 │ public DateTime? NgayHetHanThanhToan { get; set; }

27 │

28 │ public decimal? CongNoTheoDonHang { get; set; }

29 │

30 │ public decimal? DaThanhToan { get; set; }

31 │

32 │ public decimal? ConLai { get; set; }

33 │

34 │ public decimal HanMucNo { get; set; }

35 │ }

────────────────────────────────────────────────────────────

**📄 VTieuChuanKyThuatHd.cs (CS, 19 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VTieuChuanKyThuatHd

7 │ {

8 │ public int MaTieuChuan { get; set; }

9 │

10 │ public string? TenTieuChuan { get; set; }

11 │

12 │ public string? TenDvt { get; set; }

13 │

14 │ public string? TieuChuan { get; set; }

15 │

16 │ public string? TenTieuChuanTiengAnh { get; set; }

17 │

18 │ public string? TieuChuanTiengAnh { get; set; }

19 │ }

────────────────────────────────────────────────────────────

**📄 VVanChuyen.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class VVanChuyen

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaLenhXuatKho { get; set; }

11 │

12 │ public int? MaDonHang { get; set; }

13 │

14 │ public string? SoDonHang { get; set; }

15 │

16 │ public string? TenKh { get; set; }

17 │

18 │ public DateTime? NgayVanChuyen { get; set; }

19 │

20 │ public int? MaNhaVanChuyen { get; set; }

21 │

22 │ public string? TenNvc { get; set; }

23 │

24 │ public int? MaTienTe { get; set; }

25 │

26 │ public string? TenTienTe { get; set; }

27 │

28 │ public int? TyGia { get; set; }

29 │

30 │ public decimal? ThanhTien { get; set; }

31 │

32 │ public int? DaThanhToan { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 Xuất.cs (CS, 23 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models.CRMEntity;

5 │

6 │ public partial class Xuất

7 │ {

8 │ public string? SốPhiếu { get; set; }

9 │

10 │ public string? LoạiPhiếu { get; set; }

11 │

12 │ public DateTime? NgàyXuất { get; set; }

13 │

14 │ public string? NhàCungCấpMới { get; set; }

15 │

16 │ public string? SốLệnhXuấtKho { get; set; }

17 │

18 │ public string? SốĐơnHàng { get; set; }

19 │

20 │ public string? KháchHàng { get; set; }

21 │

22 │ public string? Mancc { get; set; }

23 │ }

────────────────────────────────────────────────────────────

**📄 arduino\_CheckInOut.cs (CS, 17 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class arduino\_CheckInOut

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaThietBi { get; set; }

11 │

12 │ public string NgayGioHeThong { get; set; } = null!;

13 │

14 │ public string NgayGioThietBi { get; set; } = null!;

15 │

16 │ public string Status { get; set; } = null!;

17 │ }

────────────────────────────────────────────────────────────

**📄 arduino\_Log.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class arduino\_Log

7 │ {

8 │ public int MA { get; set; }

9 │

10 │ public int MaThietBi { get; set; }

11 │

12 │ public string NgayGioThietBi { get; set; } = null!;

13 │

14 │ public DateTime NgayGioHeThong { get; set; }

15 │

16 │ public bool CamBienKimLoai { get; set; }

17 │

18 │ public bool? TrangThai { get; set; }

19 │

20 │ public int? NguoiTao { get; set; }

21 │

22 │ public DateTime? NgayTao { get; set; }

23 │

24 │ public int? NguoiSua { get; set; }

25 │

26 │ public DateTime? NgaySua { get; set; }

27 │

28 │ public bool? DaXoa { get; set; }

29 │

30 │ public int? NguoiXoa { get; set; }

31 │

32 │ public DateTime? NgayXoa { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 arduino\_ThietBi.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class arduino\_ThietBi

7 │ {

8 │ public int MaThietBi { get; set; }

9 │

10 │ public int? MaCN { get; set; }

11 │

12 │ public string TenThietBi { get; set; } = null!;

13 │

14 │ public string ViTri { get; set; } = null!;

15 │

16 │ public string NgayDangKi { get; set; } = null!;

17 │

18 │ public bool? TrangThai { get; set; }

19 │

20 │ public int? NguoiTao { get; set; }

21 │

22 │ public DateTime? NgayTao { get; set; }

23 │

24 │ public int? NguoiSua { get; set; }

25 │

26 │ public DateTime? NgaySua { get; set; }

27 │

28 │ public bool? DaXoa { get; set; }

29 │

30 │ public int? NguoiXoa { get; set; }

31 │

32 │ public DateTime? NgayXoa { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 bt\_BanThanhPham.cs (CS, 71 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class bt\_BanThanhPham

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Gio { get; set; }

11 │

12 │ public string? Phut { get; set; }

13 │

14 │ public int? MaLoThuMua { get; set; }

15 │

16 │ public string? MaSanPham { get; set; }

17 │

18 │ public string? BonPhanUng { get; set; }

19 │

20 │ public string? BonDem { get; set; }

21 │

22 │ public decimal? PHSB { get; set; }

23 │

24 │ public decimal? GelatinnizationTemperature { get; set; }

25 │

26 │ public decimal? BrabenderViscosity { get; set; }

27 │

28 │ public decimal? PercentAcetyl { get; set; }

29 │

30 │ public decimal? DS { get; set; }

31 │

32 │ public decimal? FreeClo { get; set; }

33 │

34 │ public decimal? SulfurDioxide { get; set; }

35 │

36 │ public decimal? ConductivityOfHydrocarbons { get; set; }

37 │

38 │ public string? Note { get; set; }

39 │

40 │ public string? UserAnalyst { get; set; }

41 │

42 │ public int? UserCreated { get; set; }

43 │

44 │ public DateTime? CreatedDate { get; set; }

45 │

46 │ public int? UserModified { get; set; }

47 │

48 │ public DateTime? ModifiedDate { get; set; }

49 │

50 │ public int? UserDeleted { get; set; }

51 │

52 │ public DateTime? DeletedDate { get; set; }

53 │

54 │ public DateTime? Ngay { get; set; }

55 │

56 │ public int? MaDauVao { get; set; }

57 │

58 │ public int? MaChiNhanh { get; set; }

59 │

60 │ public bool? Status { get; set; }

61 │

62 │ public bool? Deleted { get; set; }

63 │

64 │ public string? TruongCa { get; set; }

65 │

66 │ public string? PhanUng { get; set; }

67 │

68 │ public string? QC { get; set; }

69 │

70 │ public string? GhiChu { get; set; }

71 │ }

────────────────────────────────────────────────────────────

**📄 bt\_BaoBi.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class bt\_BaoBi

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? Ten { get; set; }

11 │

12 │ public string? VatLieuBaoBi { get; set; }

13 │

14 │ public bool? TrangThai { get; set; }

15 │

16 │ public int? NguoiTao { get; set; }

17 │

18 │ public DateTime? NgayTao { get; set; }

19 │

20 │ public int? NguoiSua { get; set; }

21 │

22 │ public DateTime? NgaySua { get; set; }

23 │

24 │ public bool? DaXoa { get; set; }

25 │

26 │ public int? NguoiXoa { get; set; }

27 │

28 │ public DateTime? NgayXoa { get; set; }

29 │

30 │ public int MaChiNhanh { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 bt\_ChiTietSoLieu.cs (CS, 79 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class bt\_ChiTietSoLieu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaSoLieu { get; set; }

11 │

12 │ public string? Gio { get; set; }

13 │

14 │ public string? Phut { get; set; }

15 │

16 │ public decimal? Whiteness { get; set; }

17 │

18 │ public decimal? Moisture { get; set; }

19 │

20 │ public decimal? Fineness { get; set; }

21 │

22 │ public decimal? pHValue { get; set; }

23 │

24 │ public decimal? Residue { get; set; }

25 │

26 │ public decimal? ViscosityPeakBU { get; set; }

27 │

28 │ public decimal? ViscosityMax950C0MinBU { get; set; }

29 │

30 │ public decimal? ViscosityMax950C10MinBU { get; set; }

31 │

32 │ public decimal? GelatinnizationTemperature { get; set; }

33 │

34 │ public decimal? AcetylGroup { get; set; }

35 │

36 │ public decimal? DS { get; set; }

37 │

38 │ public decimal? Conductivity { get; set; }

39 │

40 │ public decimal? ViscosityEndPointBU { get; set; }

41 │

42 │ public string? Note { get; set; }

43 │

44 │ public string? Col1 { get; set; }

45 │

46 │ public string? Col2 { get; set; }

47 │

48 │ public string? Col3 { get; set; }

49 │

50 │ public string? Col4 { get; set; }

51 │

52 │ public int? NguoiTao { get; set; }

53 │

54 │ public DateTime? NgayTao { get; set; }

55 │

56 │ public int? NguoiSua { get; set; }

57 │

58 │ public DateTime? NgaySua { get; set; }

59 │

60 │ public bool? DaXoa { get; set; }

61 │

62 │ public int? NguoiXoa { get; set; }

63 │

64 │ public DateTime? NgayXoa { get; set; }

65 │

66 │ public int? MaSanPham { get; set; }

67 │

68 │ public int? MaLoThuMua { get; set; }

69 │

70 │ public string? ForeignMatter { get; set; }

71 │

72 │ public decimal? SO2 { get; set; }

73 │

74 │ public string? Sensory { get; set; }

75 │

76 │ public decimal? StarchContent { get; set; }

77 │

78 │ public decimal? ViscosityPeakCP { get; set; }

79 │ }

────────────────────────────────────────────────────────────

**📄 bt\_ChiTietTieuChuan.cs (CS, 35 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class bt\_ChiTietTieuChuan

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaTieuChuan { get; set; }

11 │

12 │ public int? No { get; set; }

13 │

14 │ public string? TestItem { get; set; }

15 │

16 │ public string? Unit { get; set; }

17 │

18 │ public string? Specification { get; set; }

19 │

20 │ public string? Method { get; set; }

21 │

22 │ public int? NguoiTao { get; set; }

23 │

24 │ public DateTime? NgayTao { get; set; }

25 │

26 │ public int? NguoiSua { get; set; }

27 │

28 │ public DateTime? NgaySua { get; set; }

29 │

30 │ public bool? DaXoa { get; set; }

31 │

32 │ public int? NguoiXoa { get; set; }

33 │

34 │ public DateTime? NgayXoa { get; set; }

35 │ }

────────────────────────────────────────────────────────────

**📄 bt\_DauVao.cs (CS, 37 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class bt\_DauVao

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public decimal? pHValue { get; set; }

11 │

12 │ public decimal? HCl { get; set; }

13 │

14 │ public decimal? NaOH { get; set; }

15 │

16 │ public decimal? HClLoang { get; set; }

17 │

18 │ public decimal? NaOHLoang { get; set; }

19 │

20 │ public decimal? RasperChao { get; set; }

21 │

22 │ public decimal? BaumeSB { get; set; }

23 │

24 │ public int? UserCreated { get; set; }

25 │

26 │ public DateTime? CreatedDate { get; set; }

27 │

28 │ public int? UserModified { get; set; }

29 │

30 │ public DateTime? ModifiedDate { get; set; }

31 │

32 │ public int? UserDeleted { get; set; }

33 │

34 │ public DateTime? DeletedDate { get; set; }

35 │

36 │ public bool? Deleted { get; set; }

37 │ }

────────────────────────────────────────────────────────────

**📄 bt\_SoLieuBienTinh.cs (CS, 47 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class bt\_SoLieuBienTinh

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public DateTime? Ngay { get; set; }

13 │

14 │ public string? TruongCa { get; set; }

15 │

16 │ public string? QC { get; set; }

17 │

18 │ public string? ThuKhoBot { get; set; }

19 │

20 │ public string? TruongBPTP { get; set; }

21 │

22 │ public string? DotLo { get; set; }

23 │

24 │ public bool? TrangThai { get; set; }

25 │

26 │ public int? NguoiTao { get; set; }

27 │

28 │ public DateTime? NgayTao { get; set; }

29 │

30 │ public int? NguoiSua { get; set; }

31 │

32 │ public DateTime? NgaySua { get; set; }

33 │

34 │ public bool? DaXoa { get; set; }

35 │

36 │ public int? NguoiXoa { get; set; }

37 │

38 │ public DateTime? NgayXoa { get; set; }

39 │

40 │ public string? TenSanPham { get; set; }

41 │

42 │ public string? DonHang { get; set; }

43 │

44 │ public string? MaLo { get; set; }

45 │

46 │ public int? MaLoThuMua { get; set; }

47 │ }

────────────────────────────────────────────────────────────

**📄 bt\_TenTieuDeBaoCao.cs (CS, 15 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class bt\_TenTieuDeBaoCao

7 │ {

8 │ public int id { get; set; }

9 │

10 │ public string? ColumnName { get; set; }

11 │

12 │ public string? ColumnText { get; set; }

13 │

14 │ public int? DecimalNumber { get; set; }

15 │ }

────────────────────────────────────────────────────────────

**📄 bt\_TieuChuan.cs (CS, 43 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class bt\_TieuChuan

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaChiNhanh { get; set; }

11 │

12 │ public string? TieuChuan { get; set; }

13 │

14 │ public int? MaSanPham { get; set; }

15 │

16 │ public int? MaVatLieuBaoBi { get; set; }

17 │

18 │ public bool? TrangThai { get; set; }

19 │

20 │ public int? NguoiTao { get; set; }

21 │

22 │ public DateTime? NgayTao { get; set; }

23 │

24 │ public int? NguoiSua { get; set; }

25 │

26 │ public DateTime? NgaySua { get; set; }

27 │

28 │ public bool? DaXoa { get; set; }

29 │

30 │ public int? NguoiXoa { get; set; }

31 │

32 │ public DateTime? NgayXoa { get; set; }

33 │

34 │ public int? No { get; set; }

35 │

36 │ public string? TestingItem { get; set; }

37 │

38 │ public string? Unit { get; set; }

39 │

40 │ public string? Specification { get; set; }

41 │

42 │ public string? Method { get; set; }

43 │ }

────────────────────────────────────────────────────────────

**📄 catChiNhanhController.cs (CS, 235 lines)**

1 │ ﻿using Microsoft.AspNetCore.Mvc;

2 │ using Microsoft.EntityFrameworkCore;

3 │ using Microsoft.IdentityModel.Tokens;

4 │ using Newtonsoft.Json.Linq;

5 │ using QuanLyChatLuong.Common;

6 │ using QuanLyChatLuong.Controllers;

7 │ using QuanLyChatLuong.Models;

8 │ using QuanLyChatLuong.Models.APIModels;

9 │ using System.Data;

10 │ using System.Drawing.Printing;

11 │ using System.Net;

12 │ using System.Runtime.CompilerServices;

13 │

14 │ namespace QuanLyChatLuong.Controllers

15 │ {

16 │ [Route("catChiNhanh")]

17 │ public class catChiNhanhController : BaseController

18 │ {

19 │ public catChiNhanhController(QuanLyChatLuongDbContext dbcontext, MemoryCacheManager memoryCache) : base(dbcontex...

20 │ {

21 │ m\_MenuID = MenuID.catChiNhanh;

22 │ }

23 │

24 │ [TokenUserCheckHTTP]

25 │ [HttpGet]

26 │ public IActionResult Index()

27 │ {

28 │ var listData = dbcontext.tbl\_ChiNhanhs

29 │ .Where(x => !(x.DaXoa ?? false))

30 │ .ToList();

31 │ ViewData["ListData"] = listData;

32 │ return View();

33 │ }

34 │

35 │ [TokenUserCheckHTTP(Permission.Add)]

36 │ [HttpGet("add")]

37 │ public IActionResult Add()

38 │ {

39 │ return View("Form");

40 │ }

41 │

42 │ [TokenUserCheckHTTP(Permission.Edit)]

43 │ [HttpGet("edit")]

44 │ public IActionResult Edit(string id)

45 │ {

46 │ var item = dbcontext.tbl\_ChiNhanhs.FirstOrDefault(c => c.Ma == Cast.ToInt(id.Decrypt()));

47 │ if (item == null) return NotFound("Chi nhanh not found");

48 │ if (item != null && item.Logo != null)

49 │ {

50 │ string base64Logo = Convert.ToBase64String(item.Logo);

51 │ ViewBag.LogoBase64 = base64Logo;

52 │ }

53 │

54 │

55 │ ViewData["item"] = item;

56 │ return View("Form");

57 │ }

58 │

59 │

60 │

61 │ private string CheckExists(int Id, string Code, string Name)

62 │ {

63 │ if (dbcontext.tbl\_ChiNhanhs.Any(x => x.Code == Code && x.Ma != Id && !(x.DaXoa ?? false)))

64 │ {

65 │ return "Mã chi nhánh đã tồn tại";

66 │ }

67 │ if (dbcontext.tbl\_ChiNhanhs.Any(x => x.Ten == Name && x.Ma != Id && !(x.DaXoa ?? false)))

68 │ {

69 │ return "Tên chi nhánh đã tồn tại";

70 │ }

71 │

72 │ return string.Empty;

73 │ }

74 │

75 │ private ResponseModel CheckRequire(string Code, string Name, string FullName, string Address)

76 │ {

77 │ if (Code.IsEmpty())

78 │ {

79 │ return new ResponseModel(QLCLConst.Error, "Chưa nhập mã chi nhánh");

80 │ }

81 │

82 │ if (Name.IsEmpty())

83 │ {

84 │ return new ResponseModel(QLCLConst.Error, "Chưa nhập tên chi nhánh");

85 │ }

86 │ if (FullName.IsEmpty())

87 │ {

88 │ return new ResponseModel(QLCLConst.Error, "Chưa chọn tên đầy đủ");

89 │ }

90 │ if (Address.IsEmpty())

91 │ {

92 │ return new ResponseModel(QLCLConst.Error, "Chưa chọn địa chỉ");

93 │ }

94 │

95 │

96 │ return null;

97 │ }

98 │ [TokenUserCheckHTTP()]

99 │

100 │ [HttpPost("addAPI")]

101 │ public IActionResult AddAPI(IFormFile logoFile, int Id, string Code, string Name, string FullName, string Addres...

102 │ {

103 │ var result = CheckRequire(Code, Name, FullName, Address);

104 │ if (result != null)

105 │ {

106 │ return Json(result);

107 │ }

108 │ var errorMessage = CheckExists(Id, Code, Name);

109 │ if (!string.IsNullOrEmpty(errorMessage))

110 │ {

111 │ return Json(new ResponseModel(QLCLConst.Error, errorMessage));

112 │ }

113 │

114 │

115 │ byte[] logoData = null;

116 │ if (logoFile != null && logoFile.Length > 0)

117 │ {

118 │ using (var memoryStream = new MemoryStream())

119 │ {

120 │ logoFile.CopyTo(memoryStream);

121 │ logoData = memoryStream.ToArray();

122 │ }

123 │ }

124 │

125 │ var item = new tbl\_ChiNhanh

126 │ {

127 │ Code = Code,

128 │ Ten = Name,

129 │ TenDayDu = FullName,

130 │ DiaChi = Address,

131 │ DienThoai = Phone,

132 │ Fax = Fax,

133 │ Website = Website,

134 │ Email = Email,

135 │ Logo = logoData,

136 │ TrangThai = Status == "1",

137 │ NguoiTao = objToken!.userid,

138 │ NgayTao = DateTime.Now,

139 │ };

140 │

141 │ dbcontext.tbl\_ChiNhanhs.Add(item);

142 │ dbcontext.SaveChanges();

143 │

144 │ return Json(new ResponseModel(QLCLConst.Successfully, "Thêm chi nhánh thành công"));

145 │ }

146 │ [TokenUserCheckHTTP()]

147 │ [HttpPost("updateAPI")]

148 │ public IActionResult UpdateAPI(IFormFile logoFile, int Id, string Code, string Name, string FullName, string Add...

149 │ {

150 │ var result = CheckRequire(Code, Name, FullName, Address);

151 │ if (result != null)

152 │ {

153 │ return Json(result);

154 │ }

155 │ var errorMessage = CheckExists(Id, Code, Name);

156 │ if (!string.IsNullOrEmpty(errorMessage))

157 │ {

158 │ return Json(new ResponseModel(QLCLConst.Error, errorMessage));

159 │ }

160 │

161 │

162 │

163 │ tbl\_ChiNhanh item = dbcontext.tbl\_ChiNhanhs.FirstOrDefault(x => x.Ma == Id && !(x.DaXoa ?? false));

164 │ if (item == null)

165 │ {

166 │ return Json(new ResponseModel(QLCLConst.Error, "Chi nhánh không tồn tại"));

167 │ }

168 │

169 │ item.Code = Code;

170 │ item.Ten = Name;

171 │ item.TenDayDu = FullName;

172 │ item.DiaChi = Address;

173 │ item.DienThoai = Phone;

174 │ item.Fax = Fax;

175 │ item.Website = Website;

176 │ item.Email = Email;

177 │ item.TrangThai = Status == "1";

178 │ item.NguoiSua = objToken!.userid;

179 │ item.NgaySua = DateTime.Now;

180 │

181 │ if (logoFile != null && logoFile.Length > 0)

182 │ {

183 │ using (var memoryStream = new MemoryStream())

184 │ {

185 │ logoFile.CopyTo(memoryStream);

186 │ item.Logo = memoryStream.ToArray();

187 │ }

188 │ }

189 │

190 │ try

191 │ {

192 │ dbcontext.tbl\_ChiNhanhs.Update(item);

193 │ dbcontext.SaveChanges();

194 │ }

195 │ catch (Exception ex)

196 │ {

197 │ return Json(new ResponseModel(QLCLConst.Error, "Cập nhật chi nhánh thất bại: " + ex.Message));

198 │ }

199 │

200 │ return Json(new ResponseModel(QLCLConst.Successfully, "Cập nhật chi nhánh thành công"));

201 │ }

202 │

203 │ [HttpPost("DeleteAPI")]

204 │ [TokenUserCheckHTTP(Permission.Delete)]

205 │ public IActionResult DeleteAPI(string eMa)

206 │ {

207 │ var item = dbcontext.tbl\_ChiNhanhs.FirstOrDefault(x => x.Ma == Cast.ToInt(eMa.Decrypt()));

208 │ if (item == null)

209 │ {

210 │ return Json(QLCLConst.ResponseNotExist);

211 │ }

212 │ item.DaXoa = true;

213 │ item.NguoiXoa = objToken!.userid;

214 │ item.NgayXoa = DateTime.Now;

215 │ dbcontext.SaveChanges();

216 │ return Json(QLCLConst.ResponseDeleteSuccessfully);

217 │ }

218 │ [HttpPost("getBranch")]

219 │ [TokenUserCheckHTTP]

220 │ public ActionResult getBranch()

221 │ {

222 │ List<tbl\_ChiNhanh> sysBranches = dbcontext.tbl\_ChiNhanhs.Where(x => !(x.DaXoa ?? false)).ToList();

223 │ return Json(new ResponseModel(QLCLConst.OK, "", sysBranches.Select(i => new

224 │ {

225 │ ma = i.Ma,

226 │ code = i.Code,

227 │ ten = i.Ten,

228 │ diachi = i.DiaChi,

229 │ trangthai = i.TrangThai,

230 │ })));

231 │ }

232 │ }

233 │

234 │

235 │ }

────────────────────────────────────────────────────────────

**📄 sp\_BaoCaoSoLieuBienTinhSoSanhTheoNgayResult.cs (CS, 39 lines)**

1 │ ﻿namespace QuanLyChatLuong.Models.StoreProcedures

2 │ {

3 │ public class sp\_BaoCaoSoLieuBienTinhSoSanhTheoNgayResult

4 │ {

5 │ public string? Ngay { get; set; }

6 │ public string? Gio { get; set; }

7 │ public string? Phut { get; set; }

8 │ public string? TruongCa\_1 { get; set; }

9 │ public string? TruongCa\_2 { get; set; }

10 │ public decimal? Whiteness\_1 { get; set; }

11 │ public decimal? Whiteness\_2 { get; set; }

12 │ public decimal? Moisture\_1 { get; set; }

13 │ public decimal? Moisture\_2 { get; set; }

14 │ public decimal? Fineness\_1 { get; set; }

15 │ public decimal? Fineness\_2 { get; set; }

16 │ public decimal? pHValue\_1 { get; set; }

17 │ public decimal? pHValue\_2 { get; set; }

18 │ public decimal? Residue\_1 { get; set; }

19 │ public decimal? Residue\_2 { get; set; }

20 │ public decimal? ViscosityPeakBU\_1 { get; set; }

21 │ public decimal? ViscosityPeakBU\_2 { get; set; }

22 │ public decimal? ViscosityMax950C0MinBU\_1 { get; set; }

23 │ public decimal? ViscosityMax950C0MinBU\_2 { get; set; }

24 │ public decimal? ViscosityMax950C10MinBU\_1 { get; set; }

25 │ public decimal? ViscosityMax950C10MinBU\_2 { get; set; }

26 │ public decimal? GelatinnizationTemperature\_1 { get; set; }

27 │ public decimal? GelatinnizationTemperature\_2 { get; set; }

28 │ public decimal? AcetylGroup\_1 { get; set; }

29 │ public decimal? AcetylGroup\_2 { get; set; }

30 │ public decimal? DS\_1 { get; set; }

31 │ public decimal? DS\_2 { get; set; }

32 │ public decimal? Conductivity\_1 { get; set; }

33 │ public decimal? Conductivity\_2 { get; set; }

34 │ public string? Note\_1 { get; set; }

35 │ public string? Note\_2 { get; set; }

36 │ public decimal? ViscosityEndPointBU\_1 { get; set; }

37 │ public decimal? ViscosityEndPointBU\_2 { get; set; }

38 │ }

39 │ }

────────────────────────────────────────────────────────────

**📄 sp\_BaoCaoSoLieuBienTinhTheoNgayResult.cs (CS, 41 lines)**

1 │ ﻿namespace QuanLyChatLuong.Models.StoreProcedures

2 │ {

3 │ public class sp\_BaoCaoSoLieuBienTinhTheoNgayResult

4 │ {

5 │ public DateTime? Ngay { get; set; }

6 │ public string? TruongCa { get; set; }

7 │ public string? QC { get; set; }

8 │ public string? ThuKhoBot { get; set; }

9 │ public string? TruongBPTP { get; set; }

10 │ public string? DotLo { get; set; }

11 │ public string? Gio { get; set; }

12 │ public string? Phut { get; set; }

13 │ public decimal? Whiteness { get; set; }

14 │ public decimal? Moisture { get; set; }

15 │ public decimal? Fineness { get; set; }

16 │ public decimal? pHValue { get; set; }

17 │ public decimal? Residue { get; set; }

18 │ public decimal? ViscosityPeakBU { get; set; }

19 │ public decimal? ViscosityMax950C0MinBU { get; set; }

20 │ public decimal? ViscosityMax950C10MinBU { get; set; }

21 │ public decimal? ViscosityEndPointBU { get; set; }

22 │ public decimal? GelatinnizationTemperature { get; set; }

23 │ public decimal? AcetylGroup { get; set; }

24 │ public decimal? DS { get; set; }

25 │ public decimal? Conductivity { get; set; }

26 │ public string? Note { get; set; }

27 │ public bool? TrangThai { get; set; }

28 │ public int? MaLo { get; set; }

29 │ public int? MaSoLieu { get; set; }

30 │ public int? MaSanPham { get; set; }

31 │ public int? MaCT { get; set; }

32 │

33 │ public string? ForeignMatter { get; set; }

34 │ public string? Sensory { get; set; }

35 │ public decimal? SO2 { get; set; }

36 │ public decimal? StarchContent { get; set; }

37 │ public string? DonHang { get; set; }

38 │ public string? TenSanPham { get; set; }

39 │ public string? CodeLo { get; set; }

40 │ }

41 │ }

────────────────────────────────────────────────────────────

**📄 sp\_BaoCaoSoLieuBienTinhTuNgayDenNgayResult.cs (CS, 31 lines)**

1 │ ﻿namespace QuanLyChatLuong.Models.StoreProcedures

2 │ {

3 │ public class sp\_BaoCaoSoLieuBienTinhTuNgayDenNgayResult

4 │ {

5 │ public DateTime? Ngay { get; set; }

6 │ public string? TuNgay { get; set; }

7 │ public string? DenNgay { get; set; }

8 │ public string? TruongCa { get; set; }

9 │ public string? QC { get; set; }

10 │ public string? ThuKhoBot { get; set; }

11 │ public string? TruongBPTP { get; set; }

12 │ public string? DotLo { get; set; }

13 │ public string? Gio { get; set; }

14 │ public string? Phut { get; set; }

15 │ public decimal? Whiteness { get; set; }

16 │ public decimal? Moisture { get; set; }

17 │ public decimal? Fineness { get; set; }

18 │ public decimal? pHValue { get; set; }

19 │ public decimal? Residue { get; set; }

20 │ public decimal? ViscosityPeakBU { get; set; }

21 │ public decimal? ViscosityMax950C0MinBU { get; set; }

22 │ public decimal? ViscosityMax950C10MinBU { get; set; }

23 │ public decimal? ViscosityEndPointBU { get; set; }

24 │ public decimal? GelatinnizationTemperature { get; set; }

25 │ public decimal? AcetylGroup { get; set; }

26 │ public decimal? DS { get; set; }

27 │ public decimal? Conductivity { get; set; }

28 │ public string? Note { get; set; }

29 │ public bool? TrangThai { get; set; }

30 │ }

31 │ }

────────────────────────────────────────────────────────────

**📄 sp\_DeviceListResult.cs (CS, 15 lines)**

1 │ ﻿using System;

2 │ using System.ComponentModel.DataAnnotations.Schema;

3 │

4 │ namespace QuanLyChatLuong.Models

5 │ {

6 │ public class sp\_DeviceListResult

7 │ {

8 │ public int Ma { get; set; }

9 │ public int MaThietBi { get; set; }

10 │ public string? TenThietBi { get; set; }

11 │ public string? NgayGioThietBi { get; set; }

12 │ public DateTime NgayGioHeThong { get; set; }

13 │ public bool CamBienKimLoai { get; set; }

14 │ }

15 │ }

────────────────────────────────────────────────────────────

**📄 sp\_HienThiLedResult.cs (CS, 11 lines)**

1 │ ﻿namespace QuanLyChatLuong.Models.StoreProcedures

2 │ {

3 │ public class sp\_HienThiLedResult

4 │ {

5 │ public int Ma { get; set; }

6 │ public string? Ngay { get; set; }

7 │ public string? B\_DoTrang { get; set; }

8 │ public string? B\_DoAm { get; set; }

9 │ public string? B\_TapChat { get; set; }

10 │ }

11 │ }

────────────────────────────────────────────────────────────

**📄 sp\_RoleMenuResult.cs (CS, 13 lines)**

1 │ ﻿namespace QuanLyChatLuong.Models

2 │ {

3 │ public class sp\_RoleMenuResult

4 │ {

5 │ public int? ID { get; set; }

6 │ public int? MenuID { get; set; }

7 │ public bool? AllowAccess { get; set; }

8 │ public bool? Status { get; set; }

9 │ public int? Sequence { get; set; }

10 │ public string? TypeMenu { get; set; } = null!;

11 │ public string? MenuName { get; set; } = null!;

12 │ }

13 │ }

────────────────────────────────────────────────────────────

**📄 sp\_RolePermissionResult.cs (CS, 11 lines)**

1 │ ﻿namespace QuanLyChatLuong.Models

2 │ {

3 │ public class sp\_RolePermissionResult

4 │ {

5 │ public int? Ma { get; set; }

6 │ public int? MaMenu { get; set; }

7 │ public int? MaQuyen { get; set; }

8 │ public bool? TrangThai { get; set; }

9 │ public string? TenQuyen { get; set; }

10 │ }

11 │ }

────────────────────────────────────────────────────────────

**📄 sx\_SoLieuSX.cs (CS, 39 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class sx\_SoLieuSX

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaChiNhanh { get; set; }

11 │

12 │ public DateTime Ngay { get; set; }

13 │

14 │ public string? TruongBP { get; set; }

15 │

16 │ public string? QC { get; set; }

17 │

18 │ public string? ThuKhoBot { get; set; }

19 │

20 │ public string? TruongBPTP { get; set; }

21 │

22 │ public string? DotLo { get; set; }

23 │

24 │ public bool? TrangThai { get; set; }

25 │

26 │ public int? NguoiTao { get; set; }

27 │

28 │ public DateTime? NgayTao { get; set; }

29 │

30 │ public int? NguoiSua { get; set; }

31 │

32 │ public DateTime? NgaySua { get; set; }

33 │

34 │ public bool? DaXoa { get; set; }

35 │

36 │ public int? NguoiXoa { get; set; }

37 │

38 │ public DateTime? NgayXoa { get; set; }

39 │ }

────────────────────────────────────────────────────────────

**📄 sys\_Menu.cs (CS, 47 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class sys\_Menu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string Code { get; set; } = null!;

11 │

12 │ public int? MenuCha { get; set; }

13 │

14 │ public string Ten { get; set; } = null!;

15 │

16 │ public string? DuongDan { get; set; }

17 │

18 │ public string? Icon { get; set; }

19 │

20 │ public int SoThuTu { get; set; }

21 │

22 │ public string? ThamSo { get; set; }

23 │

24 │ public bool? HienMenu { get; set; }

25 │

26 │ public int? MenuCon { get; set; }

27 │

28 │ public bool? LuonLuonHien { get; set; }

29 │

30 │ public bool? TrangThai { get; set; }

31 │

32 │ public int? NguoiTao { get; set; }

33 │

34 │ public DateTime? NgayTao { get; set; }

35 │

36 │ public int? NguoiSua { get; set; }

37 │

38 │ public DateTime? NgaySua { get; set; }

39 │

40 │ public bool? DaXoa { get; set; }

41 │

42 │ public int? NguoiXoa { get; set; }

43 │

44 │ public DateTime? NgayXoa { get; set; }

45 │

46 │ public string? IconWeb { get; set; }

47 │ }

────────────────────────────────────────────────────────────

**📄 sys\_MenuModel.cs (CS, 125 lines)**

1 │ ﻿using Microsoft.EntityFrameworkCore;

2 │ using QuanLyChatLuong.Common;

3 │

4 │ namespace QuanLyChatLuong.Models

5 │ {

6 │ public class sysMenuModel : Base<sys\_Menu>

7 │ {

8 │ public sysMenuModel(DbContext dbContext) : base(dbContext)

9 │ { }

10 │ public List<sys\_Menu> GetParentMenu()

11 │ {

12 │ return Get(t => t.MenuCha == null && (t.TrangThai ?? false) && !(t.DaXoa ?? false))

13 │ .OrderBy(t => t.SoThuTu).ToList();

14 │ }

15 │ public sys\_Menu GetItem(int MenuID)

16 │ {

17 │ return First(t => t.Ma == MenuID && !(t.DaXoa ?? false));

18 │ }

19 │ public List<sys\_Menu> GetData()

20 │ {

21 │ return Get(t => !(t.DaXoa ?? false)).OrderBy(t => t.SoThuTu).ToList();

22 │ }

23 │

24 │ public List<sys\_Menu> GetMenu(int? UserID)

25 │ {

26 │ var menu = Get(m => \_dbContext.Set<vQuyen>().Any(p => p.MaMenu == m.Ma

27 │ && p.MaUser == UserID && p.MaQuyen == (int)Permission.View)

28 │ && (m.TrangThai ?? false) && !(m.DaXoa ?? false)).OrderBy(t => t.SoThuTu).ToList();

29 │

30 │ return menu;

31 │ }

32 │ private bool CheckExists(int Ma, string Code)

33 │ {

34 │ return Any(t => t.Ma != Ma && t.Code == Code && !(t.DaXoa ?? false));

35 │ }

36 │ public (int row, int? id) Add(string Code, int? ParentID, string Name, int Sequence, string Link, string Icon, b...

37 │ , int? ChildOfMenu, string UsingFor)

38 │ {

39 │ if (CheckExists(Utility.defaultUID, Code))

40 │ return (-3, null);

41 │

42 │ sys\_Menu menuAdd = new sys\_Menu

43 │ {

44 │ Code = Code,

45 │ MenuCha = ParentID,

46 │ Ten = Name,

47 │ SoThuTu= Sequence,

48 │ DuongDan = Link,

49 │ Icon = Icon,

50 │ MenuCon= ChildOfMenu,

51 │ TrangThai = Status,

52 │ ThamSo = UsingFor,

53 │ NguoiTao = UserID,

54 │ NgayTao = DateTime.Now

55 │ };

56 │

57 │ \_dbContext.Set<sys\_Menu>().Add(menuAdd);

58 │ return (\_dbContext.SaveChanges(), menuAdd.Ma);

59 │ }

60 │ // Summary:

61 │ // Edit menu.

62 │ // Returns:

63 │ // Case when Result = 1 then successful

64 │ // when Result = 0 then failure

65 │ // when Result = -2 then not exist

66 │ // when Result = -3 then object is duplicate

67 │ public int Edit(int ID, string Code, int? ParentID, string Name, int Sequence, string Link, string Icon, bool St...

68 │ , int? UserID, int? ChildOfMenu, string UsingFor)

69 │ {

70 │ if (CheckExists(ID, Code))

71 │ return -3;

72 │

73 │ sys\_Menu menuUpdate = GetItem(ID);

74 │

75 │ if (menuUpdate != null)

76 │ {

77 │ menuUpdate.Code = Code;

78 │ menuUpdate.MenuCha = ParentID;

79 │ menuUpdate.Ten = Name;

80 │ menuUpdate.SoThuTu = Sequence;

81 │ menuUpdate.DuongDan = Link;

82 │ menuUpdate.Icon = Icon;

83 │ menuUpdate.MenuCon = ChildOfMenu;

84 │ menuUpdate.TrangThai = Status;

85 │ menuUpdate.ThamSo = UsingFor;

86 │ menuUpdate.NguoiSua = UserID;

87 │ menuUpdate.NgaySua = DateTime.Now;

88 │ return Update(menuUpdate);

89 │ }

90 │

91 │ return -2;

92 │ }

93 │

94 │ // Summary:

95 │ // Delete menu.

96 │ // Returns:

97 │ // Case when Result = 1 then successful

98 │ // when Result = 0 then failure

99 │ // when Result = -1 then cannot delete

100 │ // when Result = -2 then not exist

101 │ public int Delete(int ID, int UserID)

102 │ {

103 │ sys\_Menu drM = GetItem(ID);

104 │

105 │ if (drM != null)

106 │ {

107 │ //Check children

108 │ sys\_Menu? drPM = First(t => t.MenuCha == ID && !(t.DaXoa ?? false));

109 │ sys\_RoleMenu? drRM = \_dbContext.Set<sys\_RoleMenu>().FirstOrDefault(t => t.MaMenu == ID && !(t.DaXoa ?? f...

110 │

111 │ if (drPM != null || drRM != null)

112 │ return -1;

113 │ else

114 │ {

115 │ drM.DaXoa= true;

116 │ drM.NgayXoa = DateTime.Now;

117 │ drM.NguoiXoa = UserID;

118 │ return Update(drM);

119 │ }

120 │ }

121 │

122 │ return -2;

123 │ }

124 │ }

125 │ }

────────────────────────────────────────────────────────────

**📄 sys\_OTP.cs (CS, 17 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class sys\_OTP

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaUser { get; set; }

11 │

12 │ public string? OTP { get; set; }

13 │

14 │ public int? MaToken { get; set; }

15 │

16 │ public DateTime? CreateDate { get; set; }

17 │ }

────────────────────────────────────────────────────────────

**📄 sys\_Quyen.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class sys\_Quyen

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string Code { get; set; } = null!;

11 │

12 │ public string Ten { get; set; } = null!;

13 │

14 │ public string? MoTa { get; set; }

15 │

16 │ public bool? TruyCapForm { get; set; }

17 │

18 │ public bool? TrangThai { get; set; }

19 │

20 │ public int? NguoiTao { get; set; }

21 │

22 │ public DateTime? NgayTao { get; set; }

23 │

24 │ public int? NguoiSua { get; set; }

25 │

26 │ public DateTime? NgaySua { get; set; }

27 │

28 │ public bool? DaXoa { get; set; }

29 │

30 │ public int? NguoiXoa { get; set; }

31 │

32 │ public DateTime? NgayXoa { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 sys\_QuyenModel.cs (CS, 86 lines)**

1 │ ﻿using Microsoft.EntityFrameworkCore;

2 │ using QuanLyChatLuong.Common;

3 │ using System.Net;

4 │

5 │ namespace QuanLyChatLuong.Models

6 │ {

7 │ public class sysQuyenModel : Base<sys\_Quyen>

8 │ {

9 │ public sysQuyenModel(DbContext dbContext) : base(dbContext)

10 │ { }

11 │ public sys\_Quyen GetItem(int? permissionID)

12 │ {

13 │ return First(t => t.Ma == permissionID && !(t.DaXoa ?? false));

14 │ }

15 │

16 │ public List<sys\_Quyen> GetData()

17 │ {

18 │ return Get(t => !(t.DaXoa ?? false)).OrderByDescending(t => t.Ma).ToList();

19 │ }

20 │ public int Add(string Code, string Name, string Description, bool FormAccess, bool Status, int UserID)

21 │ {

22 │ if (CheckExists(Utility.defaultUID, Code))

23 │ return -3;

24 │

25 │ sys\_Quyen roleAdd = new sys\_Quyen

26 │ {

27 │ Code = Code,

28 │ Ten = Name,

29 │ MoTa= Description,

30 │ TruyCapForm = FormAccess,

31 │ TrangThai = Status,

32 │ NguoiTao = UserID,

33 │ NgayTao = DateTime.Now

34 │ };

35 │ return Add(roleAdd);

36 │ }

37 │ public int Edit(int ID, string Code, string Name, string Description, bool FormAccess, bool Status, int UserID)

38 │ {

39 │ if (CheckExists(ID, Code))

40 │ return -3;

41 │

42 │ sys\_Quyen roleUpdate = GetItem(ID);

43 │

44 │ if (roleUpdate != null)

45 │ {

46 │ roleUpdate.Code = Code;

47 │ roleUpdate.Ten = Name;

48 │ roleUpdate.MoTa = Description;

49 │ roleUpdate.TruyCapForm = FormAccess;

50 │ roleUpdate.TrangThai = Status;

51 │ roleUpdate.NguoiSua = UserID;

52 │ roleUpdate.NgaySua = DateTime.Now;

53 │ return Update(roleUpdate);

54 │ }

55 │

56 │ return -2;

57 │ }

58 │ public int Delete(int ID, int UserID)

59 │ {

60 │ sys\_Quyen drR = GetItem(ID);

61 │

62 │ if (drR != null)

63 │ {

64 │ //Check children

65 │ sys\_RoleQuyen? drRolePermission = \_dbContext.Set<sys\_RoleQuyen>().FirstOrDefault(t => t.MaQuyen == ID &&...

66 │

67 │ if (drRolePermission != null)

68 │ return -1;

69 │ else

70 │ {

71 │ drR.DaXoa = true;

72 │ drR.NgayXoa = DateTime.Now;

73 │ drR.NguoiXoa = UserID;

74 │ return Update(drR);

75 │ }

76 │ }

77 │

78 │ return -2;

79 │ }

80 │ private bool CheckExists(int ID, string Code)

81 │ {

82 │ return Any(t => t.Ma != ID && t.Code == Code && !(t.DaXoa ?? false));

83 │ }

84 │ }

85 │ }

86 │

────────────────────────────────────────────────────────────

**📄 sys\_Role.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class sys\_Role

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string Code { get; set; } = null!;

11 │

12 │ public string Ten { get; set; } = null!;

13 │

14 │ public string? MoTa { get; set; }

15 │

16 │ public bool? TrangThai { get; set; }

17 │

18 │ public int? NguoiTao { get; set; }

19 │

20 │ public DateTime? NgayTao { get; set; }

21 │

22 │ public int? NguoiSua { get; set; }

23 │

24 │ public DateTime? NgaySua { get; set; }

25 │

26 │ public bool? DaXoa { get; set; }

27 │

28 │ public int? NguoiXoa { get; set; }

29 │

30 │ public DateTime? NgayXoa { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 sys\_RoleMenu.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class sys\_RoleMenu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaRole { get; set; }

11 │

12 │ public int MaMenu { get; set; }

13 │

14 │ public bool? ChoPhepTruyCap { get; set; }

15 │

16 │ public bool? TrangThai { get; set; }

17 │

18 │ public int? NguoiTao { get; set; }

19 │

20 │ public DateTime? NgayTao { get; set; }

21 │

22 │ public int? NguoiSua { get; set; }

23 │

24 │ public DateTime? NgaySua { get; set; }

25 │

26 │ public bool? DaXoa { get; set; }

27 │

28 │ public int? NguoiXoa { get; set; }

29 │

30 │ public DateTime? NgayXoa { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 sys\_RoleMenuModel.cs (CS, 93 lines)**

1 │ ﻿using Microsoft.EntityFrameworkCore;

2 │ using QuanLyChatLuong.Common;

3 │

4 │ namespace QuanLyChatLuong.Models

5 │ {

6 │ public class sysRoleMenuModel : Base<sys\_RoleMenu>

7 │ {

8 │ public sysRoleMenuModel(DbContext dbContext) : base(dbContext)

9 │ {

10 │ }

11 │

12 │ public List<sp\_RoleMenuResult> GetRoleMenuFollowRoleID(int RoleID)

13 │ {

14 │ return \_dbContext.Set<sp\_RoleMenuResult>().FromSqlRaw($"sp\_RoleMenu @iRole='{RoleID}'").ToList();

15 │ }

16 │ public sys\_RoleMenu GetItem(int RoleMenuID)

17 │ {

18 │ return First(t => t.Ma == RoleMenuID && !(t.DaXoa ?? false));

19 │ }

20 │ public List<sys\_RoleMenu> GetDataFollowRoleID(int RoleID)

21 │ {

22 │ return Get(t => t.MaRole == RoleID && !(t.DaXoa ?? false)).OrderByDescending(t => t.NgayTao).ToList();

23 │ }

24 │ public sys\_RoleMenu GetItem(int RoleID, int MenuID)

25 │ {

26 │ return First(t => t.MaRole == RoleID && t.MaMenu == MenuID && !(t.DaXoa ?? false));

27 │ }

28 │ public int Add(int RoleID, int MenuID, bool AllowAccess, bool Status, int UserID)

29 │ {

30 │ if (CheckExists(Utility.defaultUID, RoleID, MenuID))

31 │ return -3;

32 │

33 │ sys\_RoleMenu roleMenuAdd = new sys\_RoleMenu

34 │ {

35 │ MaRole = RoleID,

36 │ MaMenu = MenuID,

37 │ ChoPhepTruyCap = AllowAccess,

38 │ TrangThai = Status,

39 │ NguoiTao = UserID,

40 │ NgayTao = DateTime.Now,

41 │ };

42 │ return Add(roleMenuAdd);

43 │ }

44 │ public int Edit(int ID, int MenuID, bool AllowAccess, bool Status, int UserID)

45 │ {

46 │ sys\_RoleMenu roleMenuUpdate = GetItem(ID);

47 │

48 │ if (roleMenuUpdate != null)

49 │ {

50 │ if (CheckExists(ID, roleMenuUpdate.MaRole, MenuID))

51 │ return -3;

52 │

53 │ roleMenuUpdate.MaMenu = MenuID;

54 │ roleMenuUpdate.ChoPhepTruyCap = AllowAccess;

55 │ roleMenuUpdate.TrangThai = Status;

56 │ roleMenuUpdate.NguoiSua = UserID;

57 │ roleMenuUpdate.NgaySua = DateTime.Now;

58 │

59 │ return Update(roleMenuUpdate);

60 │ }

61 │

62 │ return -2;

63 │ }

64 │ public int Delete(int ID, int UserID)

65 │ {

66 │ sys\_RoleMenu drRM = GetItem(ID);

67 │

68 │ if (drRM != null)

69 │ {

70 │ //Check children

71 │ sys\_RoleQuyen? drRP = \_dbContext.Set<sys\_RoleQuyen>().FirstOrDefault(t => (t.MaRole == drRM.MaRole && t....

72 │

73 │ if (drRP != null)

74 │ return -1;

75 │ else

76 │ {

77 │ drRM.DaXoa = true;

78 │ drRM.NgayXoa = DateTime.Now;

79 │ drRM.NguoiXoa = UserID;

80 │

81 │ return Update(drRM);

82 │ }

83 │ }

84 │

85 │ return -2;

86 │ }

87 │ private bool CheckExists(int ID, int RoleID, int MenuID)

88 │ {

89 │ return Any(t => t.Ma != ID && t.MaRole == RoleID && t.MaMenu == MenuID && !(t.DaXoa ?? false));

90 │ }

91 │

92 │ }

93 │ }

────────────────────────────────────────────────────────────

**📄 sys\_RoleModel.cs (CS, 84 lines)**

1 │ ﻿using Microsoft.EntityFrameworkCore;

2 │ using QuanLyChatLuong.Common;

3 │

4 │ namespace QuanLyChatLuong.Models

5 │ {

6 │ public class sysRoleModel : Base<sys\_Role>

7 │ {

8 │ public sysRoleModel(DbContext dbContext) : base(dbContext)

9 │ {

10 │ }

11 │ public List<sys\_Role> GetData()

12 │ {

13 │ return Get(t => !(t.DaXoa ?? false)).OrderByDescending(t => t.Ma).ToList();

14 │ }

15 │

16 │ public sys\_Role GetItem(int RoleID)

17 │ {

18 │ return First(t => t.Ma == RoleID && !(t.DaXoa ?? false));

19 │ }

20 │ public int Add(string Code, string Name, string Description, bool Status, int UserID)

21 │ {

22 │ if (CheckExists(Utility.defaultUID, Code))

23 │ return -3;

24 │

25 │ sys\_Role roleAdd = new sys\_Role

26 │ {

27 │ Code = Code,

28 │ Ten = Name,

29 │ MoTa = Description,

30 │ TrangThai = Status,

31 │ NguoiTao = UserID,

32 │ NgayTao = DateTime.Now

33 │ };

34 │ return Add(roleAdd);

35 │ }

36 │ public int Edit(int ID, string Code, string Name, string Description, bool Status, int UserID)

37 │ {

38 │ if (CheckExists(ID, Code))

39 │ return -3;

40 │

41 │ sys\_Role roleUpdate = GetItem(ID);

42 │

43 │ if (roleUpdate != null)

44 │ {

45 │ roleUpdate.Code = Code;

46 │ roleUpdate.Ten = Name;

47 │ roleUpdate.MoTa = Description;

48 │ roleUpdate.TrangThai = Status;

49 │ roleUpdate.NguoiSua = UserID;

50 │ roleUpdate.NgaySua = DateTime.Now;

51 │ return Update(roleUpdate);

52 │ }

53 │

54 │ return -2;

55 │ }

56 │ public int Delete(int ID, int UserID)

57 │ {

58 │ sys\_Role drR = GetItem(ID);

59 │

60 │ //if (drR != null)

61 │ //{

62 │ // //Check children

63 │ // SysRoleMenu? drRM = \_dbContext.Set<SysRoleMenu>().FirstOrDefault(t => t.RoleId == ID && !(t.Deleted ??...

64 │ // SysUserRole? drUR = \_dbContext.Set<SysUserRole>().FirstOrDefault(t => t.RoleId == ID && !(t.Deleted ??...

65 │

66 │ // if (drUR != null || drRM != null)

67 │ // return -1;

68 │ // else

69 │ // {

70 │ // drR.Deleted = true;

71 │ // drR.DeletedDate = DateTime.Now;

72 │ // drR.UserDeleted = UserID;

73 │ // return Update(drR);

74 │ // }

75 │ //}

76 │

77 │ return -2;

78 │ }

79 │ private bool CheckExists(int ID, string Code)

80 │ {

81 │ return Any(t => t.Ma != ID && t.Code == Code && !(t.DaXoa ?? false));

82 │ }

83 │ }

84 │ }

────────────────────────────────────────────────────────────

**📄 sys\_RolePermissionModel.cs (CS, 238 lines)**

1 │ ﻿using Microsoft.EntityFrameworkCore;

2 │ using QuanLyChatLuong.Common;

3 │

4 │ namespace QuanLyChatLuong.Models

5 │ {

6 │ public class sysRolePermissionModel : Base<sys\_RoleQuyen>

7 │ {

8 │ private class RoleMenu

9 │ {

10 │ public int Ma { get; set; }

11 │ public int MaMenu { get; set; }

12 │ public bool AllowAccess { get; set; }

13 │ public bool Status { get; set; }

14 │ }

15 │

16 │ private class RolePermission

17 │ {

18 │ public int Ma { get; set; }

19 │ public int MaMenu { get; set; }

20 │ public int MaQuyen { get; set; }

21 │ public bool Status { get; set; }

22 │ }

23 │ public sysRolePermissionModel(DbContext dbContext) : base(dbContext)

24 │ {

25 │ }

26 │ public sys\_RoleQuyen GetItem(int RolePermissionID)

27 │ {

28 │ return First(t => t.Ma == RolePermissionID && !(t.DaXoa ?? false));

29 │ }

30 │ public List<sys\_RoleQuyen> GetDataFollowRoleMenuID(int MenuID, int RoleID)

31 │ {

32 │ return Get(t => (t.MaRole == RoleID && t.MaMenu == MenuID) && !(t.DaXoa ?? false)).OrderByDescending(t => t....

33 │ }

34 │ public List<sp\_RolePermissionResult> GetRolePermissionFollowRoleID(int RoleID)

35 │ {

36 │ return \_dbContext.Set<sp\_RolePermissionResult>().FromSqlRaw($"exec sp\_RolePermission @iRole='{RoleID}'").ToL...

37 │ }

38 │

39 │ public int Delete(int ID, int UserID)

40 │ {

41 │ sys\_RoleQuyen drRP = GetItem(ID);

42 │ if (drRP != null)

43 │ {

44 │ drRP.DaXoa = true;

45 │ drRP.NgayXoa = DateTime.Now;

46 │ drRP.NguoiXoa = UserID;

47 │ return Update(drRP);

48 │ }

49 │ return -2;

50 │ }

51 │ public int Add(int RoleID, int MenuID, int PermissionID, bool Status, int UserID)

52 │ {

53 │ if (CheckExists(Utility.defaultUID, RoleID, MenuID, PermissionID))

54 │ return -3;

55 │

56 │ sys\_RoleQuyen rolePermissionAdd = new sys\_RoleQuyen

57 │ {

58 │ MaRole = RoleID,

59 │ MaMenu = MenuID,

60 │ MaQuyen = PermissionID,

61 │ TrangThai = Status,

62 │ NguoiTao = UserID,

63 │ NgayTao = DateTime.Now

64 │ };

65 │

66 │ return Add(rolePermissionAdd);

67 │ }

68 │

69 │ public int Edit(int ID, int PermissionID, bool Status, int UserID)

70 │ {

71 │ sys\_RoleQuyen rolePermissionUpdate = GetItem(ID);

72 │ if (rolePermissionUpdate != null)

73 │ {

74 │ rolePermissionUpdate.MaQuyen = PermissionID;

75 │ rolePermissionUpdate.TrangThai = Status;

76 │ rolePermissionUpdate.NguoiSua = UserID;

77 │ rolePermissionUpdate.NgaySua = DateTime.Now;

78 │ return Update(rolePermissionUpdate);

79 │ }

80 │ return -2;

81 │ }

82 │ public bool UpdateQuicklyPermission(int RoleID, string RMAllowAccess, string RMStatus, string RPStatus, int User...

83 │ {

84 │ //..

85 │ List<RoleMenu>? ListRoleMenu = new List<RoleMenu>();

86 │ List<RolePermission> ListRolePermisson = new List<RolePermission>();

87 │ string[]? ArrayItem = null;

88 │

89 │ //Xu ly RMAllowAccess

90 │ string[] Array = RMAllowAccess.Split(',');

91 │

92 │ if (Array != null && Array.Count() > 0)

93 │ {

94 │ for (int i = 0; i < Array.Count(); i++)

95 │ {

96 │ ArrayItem = Array[i].Split('.');

97 │

98 │ if (ArrayItem != null && ArrayItem.Count() == 2)

99 │ {

100 │ RoleMenu roleMenuAdd = new RoleMenu

101 │ {

102 │ Ma = Cast.ToInt(ArrayItem[0]),

103 │ MaMenu = Cast.ToInt(ArrayItem[1]),

104 │ AllowAccess = true,

105 │ Status = false

106 │ };

107 │ ListRoleMenu.Add(roleMenuAdd);

108 │ }

109 │ }

110 │ ListRoleMenu = ListRoleMenu.GroupBy(t => new { t.Ma, t.MaMenu }).Select(t => t.FirstOrDefault()).OrderBy...

111 │ }

112 │

113 │ //Xu ly RMStatus

114 │ Array = RMStatus.Split(',');

115 │ int itemRoleMenuID; int itemMenuID;

116 │ RoleMenu? itemRoleMenu = null;

117 │

118 │ if (Array != null && Array.Count() > 0)

119 │ {

120 │ for (int i = 0; i < Array.Count(); i++)

121 │ {

122 │ ArrayItem = Array[i].Split('.');

123 │

124 │ if (ArrayItem != null && ArrayItem.Count() == 2)

125 │ {

126 │ itemRoleMenuID = ArrayItem[0].GetGuid();

127 │ itemMenuID = ArrayItem[1].GetGuid();

128 │ itemRoleMenu = ListRoleMenu.FirstOrDefault(t => t.Ma == itemRoleMenuID && t.MaMenu == itemMenuID...

129 │

130 │ if (itemRoleMenu == null)

131 │ {

132 │ RoleMenu roleMenuAdd = new RoleMenu

133 │ {

134 │ Ma = itemRoleMenuID,

135 │ MaMenu = itemMenuID,

136 │ AllowAccess = false,

137 │ Status = true

138 │ };

139 │ ListRoleMenu.Add(roleMenuAdd);

140 │ }

141 │ else

142 │ itemRoleMenu.Status = true;

143 │ }

144 │ }

145 │ }

146 │

147 │ //Xu ly RPStatus

148 │ Array = RPStatus.Split(',');

149 │

150 │ if (Array != null && Array.Count() > 0)

151 │ {

152 │ for (int i = 0; i < Array.Count(); i++)

153 │ {

154 │ ArrayItem = Array[i].Split('.');

155 │

156 │ if (ArrayItem != null && ArrayItem.Count() == 3)

157 │ {

158 │ RolePermission rolePermissionAdd = new RolePermission

159 │ {

160 │ Ma = Cast.ToInt(ArrayItem[0]),

161 │ MaMenu = Cast.ToInt(ArrayItem[1]),

162 │ MaQuyen = Cast.ToInt(ArrayItem[2]),

163 │ Status = true

164 │ };

165 │ ListRolePermisson.Add(rolePermissionAdd);

166 │ }

167 │ }

168 │ }

169 │

170 │ //..

171 │ if (ListRoleMenu != null && ListRolePermisson != null)

172 │ {

173 │ //Update false role menu, permission exist

174 │ bool bUpdateRMP = UpdateRoleMenuPermission(RoleID);

175 │

176 │ if (bUpdateRMP)

177 │ {

178 │ int iCountUpdateRoleMenu = 0, iCountUpdateRolePermission = 0;

179 │

180 │ //Update Role Menu

181 │ sysRoleMenuModel sys\_RoleMenuM = new sysRoleMenuModel(\_dbContext);

182 │ foreach (RoleMenu item in ListRoleMenu)

183 │ {

184 │ if (item.Ma == Utility.defaultUID)//Add

185 │ iCountUpdateRoleMenu += sys\_RoleMenuM.Add(RoleID, item.MaMenu, item.AllowAccess, item.Status...

186 │ else//Edit

187 │ {

188 │ sys\_RoleMenu drRM = sys\_RoleMenuM.GetItem(item.Ma);

189 │

190 │ if (drRM != null)

191 │ iCountUpdateRoleMenu += sys\_RoleMenuM.Edit(drRM.Ma, drRM.MaMenu, item.AllowAccess, item....

192 │ }

193 │ }

194 │

195 │ //Update Role Permission

196 │ sysRolePermissionModel sys\_RolePermissionM = new sysRolePermissionModel(\_dbContext);

197 │ foreach (RolePermission item in ListRolePermisson)

198 │ {

199 │ if (item.Ma == Utility.defaultUID)//Add

200 │ {

201 │ sys\_RoleMenu drRM = sys\_RoleMenuM.GetItem(RoleID, item.MaMenu);

202 │

203 │ if (drRM != null)

204 │ iCountUpdateRolePermission += sys\_RolePermissionM.Add(drRM.MaRole,drRM.MaMenu, item.MaQu...

205 │ }

206 │ else//Edit

207 │ {

208 │ sys\_RoleQuyen drRP = sys\_RolePermissionM.GetItem(item.Ma);

209 │

210 │ if (drRP != null)

211 │ iCountUpdateRolePermission += sys\_RolePermissionM.Edit(drRP.Ma, drRP.MaQuyen, item.Statu...

212 │ }

213 │ }

214 │

215 │ //Verify

216 │ if (ListRoleMenu.Count == iCountUpdateRoleMenu && ListRolePermisson.Count == iCountUpdateRolePermiss...

217 │ return true;

218 │ return false;

219 │ }

220 │

221 │ return false;

222 │ }

223 │

224 │ return false;

225 │ }

226 │

227 │ private bool UpdateRoleMenuPermission(int RoleID)

228 │ {

229 │ if (\_dbContext.Database.ExecuteSqlRaw($"sp\_UpdateRoleMenuPermission @iRole='{RoleID}'") >= 0)

230 │ return true;

231 │ return false;

232 │ }

233 │ private bool CheckExists(int ID, int RoleID, int MenuID, int PermissionID)

234 │ {

235 │ return Any(t => t.Ma != ID && (t.MaRole == RoleID && t.MaMenu == MenuID) && t.MaQuyen == PermissionID && !(t...

236 │ }

237 │ }

238 │ }

────────────────────────────────────────────────────────────

**📄 sys\_RoleQuyen.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class sys\_RoleQuyen

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaRole { get; set; }

11 │

12 │ public int MaMenu { get; set; }

13 │

14 │ public int MaQuyen { get; set; }

15 │

16 │ public bool? TrangThai { get; set; }

17 │

18 │ public int? NguoiTao { get; set; }

19 │

20 │ public DateTime? NgayTao { get; set; }

21 │

22 │ public int? NguoiSua { get; set; }

23 │

24 │ public DateTime? NgaySua { get; set; }

25 │

26 │ public bool? DaXoa { get; set; }

27 │

28 │ public int? NguoiXoa { get; set; }

29 │

30 │ public DateTime? NgayXoa { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 sys\_User.cs (CS, 45 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class sys\_User

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string TenDangNhap { get; set; } = null!;

11 │

12 │ public string Password { get; set; } = null!;

13 │

14 │ public string HoTen { get; set; } = null!;

15 │

16 │ public bool? TuChoiNhanEmail { get; set; }

17 │

18 │ public bool? GioiTinh { get; set; }

19 │

20 │ public string? Email { get; set; }

21 │

22 │ public byte[]? Photo { get; set; }

23 │

24 │ public bool? TrangThai { get; set; }

25 │

26 │ public int? NguoiTao { get; set; }

27 │

28 │ public DateTime? NgayTao { get; set; }

29 │

30 │ public int? NguoiSua { get; set; }

31 │

32 │ public DateTime? NgaySua { get; set; }

33 │

34 │ public bool? DaXoa { get; set; }

35 │

36 │ public int? NguoiXoa { get; set; }

37 │

38 │ public DateTime? NgayXoa { get; set; }

39 │

40 │ public string? MSNV { get; set; }

41 │

42 │ public string? FCM { get; set; }

43 │

44 │ public string? SDT { get; set; }

45 │ }

────────────────────────────────────────────────────────────

**📄 sys\_UserModel.cs (CS, 161 lines)**

1 │ ﻿using Microsoft.EntityFrameworkCore;

2 │ using QuanLyChatLuong.Common;

3 │

4 │ namespace QuanLyChatLuong.Models

5 │ {

6 │ public class sysUserModel : Base<sys\_User>

7 │ {

8 │ public sysUserModel(DbContext dbContext) : base(dbContext)

9 │ {

10 │ }

11 │

12 │ public sys\_User GetItem(int? UserID)

13 │ {

14 │ return First(t => t.Ma == UserID && !(t.DaXoa ?? false));

15 │ }

16 │

17 │ public List<sys\_User> GetData()

18 │ {

19 │ return Get(t => !(t.DaXoa ?? false)).OrderByDescending(t => t.Ma).ToList();

20 │ }

21 │ private bool CheckExists(int ID, string Username)

22 │ {

23 │ return Any(t => t.Ma != ID && t.TenDangNhap == Username && !(t.DaXoa ?? false));

24 │ }

25 │ public int Edit(int ID, string Username, string FullName, int GioiTinh,

26 │ string MSNV, bool Status, int UserID)

27 │ {

28 │ if (CheckExists(ID, Username))

29 │ {

30 │ return -3;

31 │ }

32 │

33 │ sys\_User userUpdate = GetItem(ID);

34 │

35 │ if (userUpdate != null)

36 │ {

37 │ userUpdate.TenDangNhap = Username;

38 │ userUpdate.HoTen = FullName;

39 │ userUpdate.GioiTinh = GioiTinh == 1;

40 │ userUpdate.MSNV = MSNV;

41 │ userUpdate.TrangThai = Status;

42 │ userUpdate.NguoiSua = UserID;

43 │ userUpdate.NgaySua = DateTime.Now;

44 │

45 │ try

46 │ {

47 │ return Update(userUpdate);

48 │ }

49 │ catch (Exception ex)

50 │ {

51 │ string ss = ex.Message;

52 │ return 0;

53 │ }

54 │

55 │ }

56 │ return -2;

57 │ }

58 │ public (int, byte[]?) UpdateAvatar(int ID, byte[] HinhAnh)

59 │ {

60 │ sys\_User userUpdate = GetItem(ID);

61 │

62 │ if (userUpdate != null )

63 │ {

64 │ userUpdate.Photo = HinhAnh != null ? HinhAnh : userUpdate.Photo;

65 │ return (Update(userUpdate), userUpdate.Photo ?? null);

66 │ }

67 │ return (-2, null);

68 │ }

69 │

70 │ public int Add(string Username, string FullName, string Password, int GioiTinh, string MSNV, bool Status, int Us...

71 │ {

72 │ if (CheckExists(Utility.defaultUID, Username))

73 │ {

74 │ return -3;

75 │ }

76 │ sys\_User userAdd = new sys\_User

77 │ {

78 │ TenDangNhap = Username,

79 │ Password = Utility.GetSHA512(Password),

80 │ HoTen = FullName,

81 │ GioiTinh = GioiTinh == 0,

82 │ MSNV = MSNV,

83 │ TrangThai = Status,

84 │ NguoiTao = UserID,

85 │ NgayTao = DateTime.Now

86 │ };

87 │ return Add(userAdd);

88 │ }

89 │

90 │ public int Delete(int ID, int UserID)

91 │ {

92 │ sys\_User drU = GetItem(ID);

93 │

94 │ if (drU != null)

95 │ {

96 │ sys\_UserRole? drUR = \_dbContext.Set<sys\_UserRole>().FirstOrDefault(t => t.MaUser == ID && !(t.DaXoa ?? f...

97 │

98 │ if (drUR != null)

99 │ {

100 │ return -1;

101 │ }

102 │ else

103 │ {

104 │ drU.DaXoa = true;

105 │ drU.NgayXoa= DateTime.Now;

106 │ drU.NguoiXoa= UserID;

107 │ return Update(drU);

108 │ }

109 │ }

110 │

111 │ return -2;

112 │ }

113 │

114 │ public int ResetPassword(int ID, string Password)

115 │ {

116 │ sys\_User drU = GetItem(ID);

117 │ if (drU != null)

118 │ {

119 │ drU.Password = Utility.GetSHA512(Password);

120 │ return Update(drU);

121 │ }

122 │

123 │ return -1;

124 │ }

125 │

126 │ public int ChangePassword( int ID, string OldPassword, string NewPassword)

127 │ {

128 │ sys\_User drU = GetItem(ID);

129 │

130 │ if (drU != null)

131 │ {

132 │ if (drU.Password == Utility.GetSHA512(OldPassword))

133 │ return ResetPassword(ID, NewPassword);

134 │ return -2;

135 │ }

136 │

137 │ return -1;

138 │ }

139 │ public bool CheckUsername(string Username, int id)

140 │ {

141 │ return Any(x => x.TenDangNhap == Username && !(x.DaXoa ?? false) && (x.TrangThai ?? false) && (id == 0 || x....

142 │ }

143 │ public bool CheckMSNV(string Msnv, int id)

144 │ {

145 │ return Any(x => x.MSNV == Msnv && !(x.DaXoa ?? false) && (x.TrangThai ?? false) && (id == 0 || x.Ma!=id ));

146 │ }

147 │

148 │ public int UpdateFCM(int ID, string FCM)

149 │ {

150 │ sys\_User userUpdate = GetItem(ID);

151 │

152 │ if (userUpdate != null)

153 │ {

154 │ userUpdate.FCM = FCM;

155 │ int result = Update(userUpdate);

156 │ return result > 0 ? 1 : 0;

157 │ }

158 │ return -2;

159 │ }

160 │ }

161 │ }

────────────────────────────────────────────────────────────

**📄 sys\_UserRole.cs (CS, 31 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class sys\_UserRole

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaUser { get; set; }

11 │

12 │ public int MaRole { get; set; }

13 │

14 │ public int MaChiNhanh { get; set; }

15 │

16 │ public bool? TrangThai { get; set; }

17 │

18 │ public int? NguoiTao { get; set; }

19 │

20 │ public DateTime? NgayTao { get; set; }

21 │

22 │ public int? NguoiSua { get; set; }

23 │

24 │ public DateTime? NgaySua { get; set; }

25 │

26 │ public bool? DaXoa { get; set; }

27 │

28 │ public int? NguoiXoa { get; set; }

29 │

30 │ public DateTime? NgayXoa { get; set; }

31 │ }

────────────────────────────────────────────────────────────

**📄 sys\_UserRoleModel.cs (CS, 79 lines)**

1 │ ﻿using Microsoft.EntityFrameworkCore;

2 │ using QuanLyChatLuong.Common;

3 │ using QuanLyChatLuong.Models;

4 │

5 │ namespace QuanLyChatLuong.Models

6 │ {

7 │ public class sysUserRoleModel : Base<sys\_UserRole>

8 │ {

9 │ public sysUserRoleModel(DbContext dbContext) : base(dbContext)

10 │ {

11 │ }

12 │

13 │ public sys\_UserRole GetItem(int UserRoleID)

14 │ {

15 │ return First(t => t.Ma == UserRoleID && !(t.DaXoa ?? false));

16 │ }

17 │

18 │ public List<sys\_UserRole> GetData()

19 │ {

20 │ return Get(t => !(t.DaXoa ?? false)).OrderByDescending(t => t.NgayTao).ToList();

21 │ }

22 │

23 │ public List<sys\_UserRole> GetDataFollowUserID(int UserID)

24 │ {

25 │ return Get(t => t.MaUser == UserID && !(t.DaXoa ?? false)).OrderByDescending(t => t.NgayTao).ToList();

26 │ }

27 │ public int Edit(int ID, int RoleID, bool Status, int UserID, int Branch)

28 │ {

29 │ sys\_UserRole userRoleUpdate = GetItem(ID);

30 │ if (userRoleUpdate != null)

31 │ {

32 │ if (CheckExists(ID, userRoleUpdate.MaUser, RoleID, Branch))

33 │ return -3;

34 │

35 │ userRoleUpdate.MaRole = RoleID;

36 │ userRoleUpdate.TrangThai = Status;

37 │ userRoleUpdate.NguoiSua = UserID;

38 │ userRoleUpdate.MaChiNhanh = Branch;

39 │ userRoleUpdate.NgaySua = DateTime.Now;

40 │ return Update(userRoleUpdate);

41 │ }

42 │

43 │ return -2;

44 │ }

45 │ public int Add(int UserID, int RoleID, bool Status, int UserCreatedID, int Branch)

46 │ {

47 │ if (CheckExists(Utility.defaultUID, UserID, RoleID, Branch))

48 │ return -3;

49 │

50 │ sys\_UserRole userRoleAdd = new sys\_UserRole

51 │ {

52 │ MaUser = UserID,

53 │ MaRole = RoleID,

54 │ TrangThai = Status,

55 │ MaChiNhanh = Branch,

56 │ NguoiTao = UserCreatedID,

57 │ NgayTao = DateTime.Now

58 │ };

59 │ return Add(userRoleAdd);

60 │ }

61 │ public int Delete(int ID, int UserID)

62 │ {

63 │ sys\_UserRole drUR = GetItem(ID);

64 │ if (drUR != null)

65 │ {

66 │ drUR.DaXoa = true;

67 │ drUR.NgayXoa = DateTime.Now;

68 │ drUR.NguoiXoa = UserID;

69 │ return Update(drUR);

70 │ }

71 │ return -2;

72 │ }

73 │

74 │ private bool CheckExists(int ID, int UserID, int RoleID, int MaChiNhanh)

75 │ {

76 │ return Any(t => t.Ma != ID && t.MaUser == UserID && t.MaRole == RoleID && t.MaChiNhanh == MaChiNhanh && !(t....

77 │ }

78 │ }

79 │ }

────────────────────────────────────────────────────────────

**📄 tbl\_BanThanhPham.cs (CS, 87 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class tbl\_BanThanhPham

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaSoLieu { get; set; }

11 │

12 │ public string? Gio { get; set; }

13 │

14 │ public string? Phut { get; set; }

15 │

16 │ public int? MaLo { get; set; }

17 │

18 │ public decimal? pH\_RO { get; set; }

19 │

20 │ public decimal? HCL { get; set; }

21 │

22 │ public decimal? NaOH { get; set; }

23 │

24 │ public decimal? HCL\_Loang { get; set; }

25 │

26 │ public decimal? NaOH\_Loang { get; set; }

27 │

28 │ public decimal? CQ\_SodiumCarbonat { get; set; }

29 │

30 │ public decimal? RasperChao { get; set; }

31 │

32 │ public string? HamLuongTBTrongBaMi { get; set; }

33 │

34 │ public string? pHSuaBot { get; set; }

35 │

36 │ public decimal? ND\_HoHoa { get; set; }

37 │

38 │ public decimal? DoNhot { get; set; }

39 │

40 │ public string? GhiChu { get; set; }

41 │

42 │ public string? NguoiPhanTich { get; set; }

43 │

44 │ public bool? DaXoa { get; set; }

45 │

46 │ public int? UserCreated { get; set; }

47 │

48 │ public DateTime? CreatedDate { get; set; }

49 │

50 │ public int? UserModified { get; set; }

51 │

52 │ public DateTime? ModifiedDate { get; set; }

53 │

54 │ public int? UserDeleted { get; set; }

55 │

56 │ public DateTime? DeletedDate { get; set; }

57 │

58 │ public int? MaBanThanhPham { get; set; }

59 │

60 │ public string? BauMeSuaBot { get; set; }

61 │

62 │ public decimal? ViscosityPeakBU { get; set; }

63 │

64 │ public decimal? ViscosityMax950C0MinBU { get; set; }

65 │

66 │ public decimal? ViscosityMax950C10MinBU { get; set; }

67 │

68 │ public decimal? ViscosityEndPointBU { get; set; }

69 │

70 │ public decimal? GelatinnizationTemperature { get; set; }

71 │

72 │ public decimal? SO2 { get; set; }

73 │

74 │ public decimal? AcetylGroup { get; set; }

75 │

76 │ public decimal? DS { get; set; }

77 │

78 │ public decimal? Clo { get; set; }

79 │

80 │ public string? Note { get; set; }

81 │

82 │ public string? NaHCO3 { get; set; }

83 │

84 │ public string? TinhBotTrongBaMi { get; set; }

85 │

86 │ public string? CamQuanNaHCO3 { get; set; }

87 │ }

────────────────────────────────────────────────────────────

**📄 tbl\_ChiNhanh.cs (CS, 45 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class tbl\_ChiNhanh

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string Ten { get; set; } = null!;

13 │

14 │ public string? TenDayDu { get; set; }

15 │

16 │ public string? DiaChi { get; set; }

17 │

18 │ public string? DienThoai { get; set; }

19 │

20 │ public string? Fax { get; set; }

21 │

22 │ public string? Website { get; set; }

23 │

24 │ public string? Email { get; set; }

25 │

26 │ public byte[]? Logo { get; set; }

27 │

28 │ public bool? TrangThai { get; set; }

29 │

30 │ public int? NguoiTao { get; set; }

31 │

32 │ public DateTime? NgayTao { get; set; }

33 │

34 │ public int? NguoiSua { get; set; }

35 │

36 │ public DateTime? NgaySua { get; set; }

37 │

38 │ public bool? DaXoa { get; set; }

39 │

40 │ public int? NguoiXoa { get; set; }

41 │

42 │ public DateTime? NgayXoa { get; set; }

43 │

44 │ public int? nID { get; set; }

45 │ }

────────────────────────────────────────────────────────────

**📄 tbl\_ChiTietKiemTra.cs (CS, 41 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class tbl\_ChiTietKiemTra

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaKTCL { get; set; }

11 │

12 │ public int? STT { get; set; }

13 │

14 │ public string? Code { get; set; }

15 │

16 │ public string? DVT { get; set; }

17 │

18 │ public string? TieuChuan { get; set; }

19 │

20 │ public string? KetQua { get; set; }

21 │

22 │ public bool? TrangThai { get; set; }

23 │

24 │ public int? NguoiTao { get; set; }

25 │

26 │ public DateTime? NgayTao { get; set; }

27 │

28 │ public int? NguoiSua { get; set; }

29 │

30 │ public DateTime? NgaySua { get; set; }

31 │

32 │ public bool? DaXoa { get; set; }

33 │

34 │ public int? NguoiXoa { get; set; }

35 │

36 │ public DateTime? NgayXoa { get; set; }

37 │

38 │ public string? Method { get; set; }

39 │

40 │ public int? MaTieuChuan { get; set; }

41 │ }

────────────────────────────────────────────────────────────

**📄 tbl\_ChiTietSoLieu.cs (CS, 73 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class tbl\_ChiTietSoLieu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaSoLieu { get; set; }

11 │

12 │ public string? Gio { get; set; }

13 │

14 │ public string? Phut { get; set; }

15 │

16 │ public decimal? B\_DoTrang { get; set; }

17 │

18 │ public decimal? B\_DoAm { get; set; }

19 │

20 │ public decimal? B\_DoMin { get; set; }

21 │

22 │ public decimal? B\_DoPH { get; set; }

23 │

24 │ public decimal? B\_DoKeoCP { get; set; }

25 │

26 │ public decimal? B\_DoKeoBU { get; set; }

27 │

28 │ public decimal? B\_TapChat { get; set; }

29 │

30 │ public decimal? ND\_Lo { get; set; }

31 │

32 │ public decimal? ND\_Say { get; set; }

33 │

34 │ public decimal? ND\_HoHoa { get; set; }

35 │

36 │ public decimal? NS\_DoPH { get; set; }

37 │

38 │ public decimal? BU\_DoAm { get; set; }

39 │

40 │ public string? Cepa { get; set; }

41 │

42 │ public decimal? XMS\_DoAm { get; set; }

43 │

44 │ public string? XMT\_CoBot { get; set; }

45 │

46 │ public decimal? HamLuongTB { get; set; }

47 │

48 │ public string? GhiChu { get; set; }

49 │

50 │ public int? NguoiTao { get; set; }

51 │

52 │ public DateTime? NgayTao { get; set; }

53 │

54 │ public int? NguoiSua { get; set; }

55 │

56 │ public DateTime? NgaySua { get; set; }

57 │

58 │ public bool? DaXoa { get; set; }

59 │

60 │ public int? NguoiXoa { get; set; }

61 │

62 │ public DateTime? NgayXoa { get; set; }

63 │

64 │ public int? MaLoaiBot { get; set; }

65 │

66 │ public int? MaLoThuMua { get; set; }

67 │

68 │ public decimal? SO2 { get; set; }

69 │

70 │ public string? ForeignMatter { get; set; }

71 │

72 │ public string? Sensory { get; set; }

73 │ }

────────────────────────────────────────────────────────────

**📄 tbl\_ChiTieuCL.cs (CS, 15 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class tbl\_ChiTieuCL

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public int? SoTT { get; set; }

15 │ }

────────────────────────────────────────────────────────────

**📄 tbl\_ColorGioiHanSoLieu.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class tbl\_ColorGioiHanSoLieu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int? MaChiNhanh { get; set; }

11 │

12 │ public string? Code { get; set; }

13 │

14 │ public string? MinColor { get; set; }

15 │

16 │ public string? MaxColor { get; set; }

17 │

18 │ public int? TrangThai { get; set; }

19 │

20 │ public int? NguoiTao { get; set; }

21 │

22 │ public DateTime? NgayTao { get; set; }

23 │

24 │ public int? NguoiSua { get; set; }

25 │

26 │ public DateTime? NgaySua { get; set; }

27 │

28 │ public bool? DaXoa { get; set; }

29 │

30 │ public int? NguoiXoa { get; set; }

31 │

32 │ public DateTime? NgayXoa { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 tbl\_DanhMucSoLieu.cs (CS, 33 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class tbl\_DanhMucSoLieu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public int? MaNhom { get; set; }

15 │

16 │ public int? MaLoai { get; set; }

17 │

18 │ public bool? TrangThai { get; set; }

19 │

20 │ public int? NguoiTao { get; set; }

21 │

22 │ public DateTime? NgayTao { get; set; }

23 │

24 │ public int? NguoiSua { get; set; }

25 │

26 │ public DateTime? NgaySua { get; set; }

27 │

28 │ public bool? DaXoa { get; set; }

29 │

30 │ public int? NguoiXoa { get; set; }

31 │

32 │ public DateTime? NgayXoa { get; set; }

33 │ }

────────────────────────────────────────────────────────────

**📄 tbl\_GioiHanSoLieu.cs (CS, 45 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class tbl\_GioiHanSoLieu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaChiNhanh { get; set; }

11 │

12 │ public string Code { get; set; } = null!;

13 │

14 │ public string? Ten { get; set; }

15 │

16 │ public string? TenNgan { get; set; }

17 │

18 │ public string? Ten\_ZH { get; set; }

19 │

20 │ public string? Loai { get; set; }

21 │

22 │ public decimal? GiaTri { get; set; }

23 │

24 │ public string? DVT { get; set; }

25 │

26 │ public string? TieuChuan { get; set; }

27 │

28 │ public string? GhiChu { get; set; }

29 │

30 │ public bool? TrangThai { get; set; }

31 │

32 │ public int? NguoiTao { get; set; }

33 │

34 │ public DateTime? NgayTao { get; set; }

35 │

36 │ public int? NguoiSua { get; set; }

37 │

38 │ public DateTime? NgaySua { get; set; }

39 │

40 │ public bool? DaXoa { get; set; }

41 │

42 │ public int? NguoiXoa { get; set; }

43 │

44 │ public DateTime? NgayXoa { get; set; }

45 │ }

────────────────────────────────────────────────────────────

**📄 tbl\_KiemTraChatLuong.cs (CS, 57 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class tbl\_KiemTraChatLuong

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaCN { get; set; }

11 │

12 │ public string? TenSanPham { get; set; }

13 │

14 │ public string? SoDonHang { get; set; }

15 │

16 │ public DateTime? NgayGiaoHang { get; set; }

17 │

18 │ public string? SoLuong { get; set; }

19 │

20 │ public string? TenKhachHang { get; set; }

21 │

22 │ public DateTime? NgaySanXuat { get; set; }

23 │

24 │ public string? HanSuDung { get; set; }

25 │

26 │ public string? ChatLuongBot { get; set; }

27 │

28 │ public string? KetLuan { get; set; }

29 │

30 │ public string? GhiChu { get; set; }

31 │

32 │ public string? NguoiKiemTra { get; set; }

33 │

34 │ public bool? TrangThai { get; set; }

35 │

36 │ public int? NguoiTao { get; set; }

37 │

38 │ public DateTime? NgayTao { get; set; }

39 │

40 │ public int? NguoiSua { get; set; }

41 │

42 │ public DateTime? NgaySua { get; set; }

43 │

44 │ public bool? DaXoa { get; set; }

45 │

46 │ public int? NguoiXoa { get; set; }

47 │

48 │ public DateTime? NgayXoa { get; set; }

49 │

50 │ public string? MaLo { get; set; }

51 │

52 │ public string? VatLieuBaoBi { get; set; }

53 │

54 │ public string? TieuChuan { get; set; }

55 │

56 │ public int? KichThuoc { get; set; }

57 │ }

────────────────────────────────────────────────────────────

**📄 tbl\_LoaiSoLieu.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class tbl\_LoaiSoLieu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public bool? TrangThai { get; set; }

15 │

16 │ public int? NguoiTao { get; set; }

17 │

18 │ public DateTime? NgayTao { get; set; }

19 │

20 │ public int? NguoiSua { get; set; }

21 │

22 │ public DateTime? NgaySua { get; set; }

23 │

24 │ public bool? DaXoa { get; set; }

25 │

26 │ public int? NguoiXoa { get; set; }

27 │

28 │ public DateTime? NgayXoa { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 tbl\_ManLed.cs (CS, 17 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class tbl\_ManLed

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? ipLed { get; set; }

11 │

12 │ public string? http { get; set; }

13 │

14 │ public int? MaCN { get; set; }

15 │

16 │ public bool? TrangThai { get; set; }

17 │ }

────────────────────────────────────────────────────────────

**📄 tbl\_ManLedBusiness.cs (CS, 16 lines)**

1 │ ﻿using Microsoft.EntityFrameworkCore;

2 │

3 │ namespace QuanLyChatLuong.Models

4 │ {

5 │ public class tbl\_ManLedBusiness : Base<tbl\_ManLed>

6 │ {

7 │ public tbl\_ManLedBusiness(DbContext dbContext) : base(dbContext)

8 │ {

9 │ }

10 │

11 │ public List<tbl\_ManLed> GetDataManLed(int MaCN)

12 │ {

13 │ return Get(t => t.MaCN == MaCN && (t.TrangThai ?? false)).OrderByDescending(t => t.Ma).ToList();

14 │ }

15 │ }

16 │ }

────────────────────────────────────────────────────────────

**📄 tbl\_NhomSoLieu.cs (CS, 29 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class tbl\_NhomSoLieu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public string? Code { get; set; }

11 │

12 │ public string? Ten { get; set; }

13 │

14 │ public bool? TrangThai { get; set; }

15 │

16 │ public int? NguoiTao { get; set; }

17 │

18 │ public DateTime? NgayTao { get; set; }

19 │

20 │ public int? NguoiSua { get; set; }

21 │

22 │ public DateTime? NgaySua { get; set; }

23 │

24 │ public bool? DaXoa { get; set; }

25 │

26 │ public int? NguoiXoa { get; set; }

27 │

28 │ public DateTime? NgayXoa { get; set; }

29 │ }

────────────────────────────────────────────────────────────

**📄 tbl\_SoLieu.cs (CS, 39 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class tbl\_SoLieu

7 │ {

8 │ public int Ma { get; set; }

9 │

10 │ public int MaChiNhanh { get; set; }

11 │

12 │ public DateTime Ngay { get; set; }

13 │

14 │ public string? TruongCa { get; set; }

15 │

16 │ public string? QC { get; set; }

17 │

18 │ public string? ThuKhoBot { get; set; }

19 │

20 │ public string? TruongBPTP { get; set; }

21 │

22 │ public string? DotLo { get; set; }

23 │

24 │ public bool? TrangThai { get; set; }

25 │

26 │ public int? NguoiTao { get; set; }

27 │

28 │ public DateTime? NgayTao { get; set; }

29 │

30 │ public int? NguoiSua { get; set; }

31 │

32 │ public DateTime? NgaySua { get; set; }

33 │

34 │ public bool? DaXoa { get; set; }

35 │

36 │ public int? NguoiXoa { get; set; }

37 │

38 │ public DateTime? NgayXoa { get; set; }

39 │ }

────────────────────────────────────────────────────────────

**📄 tmp.cs (CS, 17 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class tmp

7 │ {

8 │ public string? name { get; set; }

9 │

10 │ public decimal? \_12\_30\_PM\_\_5\_11 { get; set; }

11 │

12 │ public decimal? \_13\_30\_PM\_\_5\_11 { get; set; }

13 │

14 │ public decimal? \_14\_00\_PM\_\_5\_11 { get; set; }

15 │

16 │ public decimal? \_14\_30\_PM\_\_5\_11 { get; set; }

17 │ }

────────────────────────────────────────────────────────────

**📄 tmpDatum.cs (CS, 15 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class tmpDatum

7 │ {

8 │ public string? ThoiGian { get; set; }

9 │

10 │ public decimal B\_DoTrang { get; set; }

11 │

12 │ public decimal B\_DoAm { get; set; }

13 │

14 │ public decimal B\_TapChat { get; set; }

15 │ }

────────────────────────────────────────────────────────────

**📄 vPermissionModel.cs (CS, 17 lines)**

1 │ ﻿using Microsoft.EntityFrameworkCore;

2 │ using QuanLyChatLuong.Models.CRMEntity;

3 │ namespace QuanLyChatLuong.Models

4 │ {

5 │ public class vPermissionModel : Base<vQuyen>

6 │ {

7 │ public vPermissionModel(DbContext dbContext) : base(dbContext)

8 │ {

9 │ // disableCache = true;

10 │ }

11 │

12 │ public List<vQuyen> GetPermissionOfUser(int UserID, int BranchID)

13 │ {

14 │ return Get(t => t.MaUser == UserID && t.MaChiNhanh == BranchID).ToList();

15 │ }

16 │ }

17 │ }

────────────────────────────────────────────────────────────

**📄 vQuyen.cs (CS, 27 lines)**

1 │ ﻿using System;

2 │ using System.Collections.Generic;

3 │

4 │ namespace QuanLyChatLuong.Models;

5 │

6 │ public partial class vQuyen

7 │ {

8 │ public int MaRole { get; set; }

9 │

10 │ public int MaMenu { get; set; }

11 │

12 │ public int? MaQuyen { get; set; }

13 │

14 │ public int MaUser { get; set; }

15 │

16 │ public string TenDangNhap { get; set; } = null!;

17 │

18 │ public int MaChiNhanh { get; set; }

19 │

20 │ public string? Email { get; set; }

21 │

22 │ public bool TuChoiNhanEmail { get; set; }

23 │

24 │ public string HoTen { get; set; } = null!;

25 │ }

26 │

27 │