﻿using MailKit.Net.Smtp;

using MailKit.Security;

using MimeKit;

using OnlineHotelManagementAPI.Models;

namespace OnlineHotelManagementAPI.Repositories

{

public class PaymentRepo : IPayment

{

private HotelContext \_context;

public PaymentRepo(HotelContext context)

{

\_context = context;

}

#region GetAllPayment

/// <summary>

/// This method to read all payment values.

/// </summary>

/// <returns></returns>

public List<Payment> GetAllPayment()

{

try

{

List<Payment> payment = \_context.Payments.ToList();

return payment;

}

catch (Exception ex)

{

throw ex;

}

}

#endregion

#region InsertPayment

/// <summary>

/// This method is to insert payment details.

/// </summary>

/// <param name="payment"></param>

/// <returns></returns>

public string InsertPayment(Payment payment)

{

string stcode = string.Empty;

try

{

string s1 = payment.CardNumber;

string s2 = "XXXX XXXX XXXX " + s1.Substring(12);

payment.CardNumber = s2;

\_context.Payments.Add(payment);

\_context.SaveChanges();

stcode = "200";

SendEmail(payment);

}

catch (Exception e)

{

stcode = "400";

}

return stcode;

}

#endregion

#region SavePayment

/// <summary>

/// This method is to save payment details.

/// </summary>

/// <param name="payment"></param>

public void SavePayment(Payment payment)

{

\_context.SaveChanges();

}

#endregion

#region UpdatePayment

/// <summary>

/// This method is to update payment details.

/// </summary>

/// <param name="payment"></param>

/// <returns></returns>

public string UpdatePayment(Payment payment)

{

string stcode = string.Empty;

try

{

\_context.Payments.Update(payment);

\_context.SaveChanges();

stcode = "200";

}

catch (Exception e)

{

stcode = "400";

}

return stcode;

}

#endregion

#region GetPaymentById

/// <summary>

/// This method is to get payment values by specific id.

/// </summary>

/// <param name="id"></param>

/// <returns></returns>

public string GetPaymentById(int Id)

{

Payment payment;

string stcode = string.Empty;

try

{

payment = \_context.Payments.Find(Id);

if (payment != null)

{

\_context.SaveChanges();

stcode = "200";

}

else

{

stcode = "400";

}

}

catch (Exception e)

{

stcode = e.Message;

}

return stcode;

}

#endregion

#region DeletePayment

/// <summary>

/// This method is to delete payment details.

/// </summary>

/// <param name="id"></param>

/// <returns></returns>

public string DeletePayment(int id)

{

string stcode = string.Empty;

try

{

var payment = \_context.Payments.Find(id);

if (payment != null)

{

\_context.Payments.Remove(payment);

\_context.SaveChanges();

stcode = "200";

}

else

{

stcode = "400";

}

}

catch

{

stcode = "400";

}

return stcode;

}

#endregion

#region SendEmail

/// <summary>

/// This method is to send email to the registered email id.

/// </summary>

/// <param name="payment"></param>

/// <returns></returns>

public async Task<Payment> SendEmail(Payment payment)

{

try

{

var email = new MimeMessage();

email.From.Add(MailboxAddress.Parse("add\_email\_address\_here"));

email.To.Add(MailboxAddress.Parse(payment.Email));

email.Subject = "Payment Done";

email.Body = new TextPart(MimeKit.Text.TextFormat.Html)

{

Text = "Dear " + payment.CardholderName + ",<br>" +

"Welcome to our Hotel. Your payment was successful. Visit us again."+"<br>"+

"<pre> Regards, Hotel.</pre>"

};

using var smtp = new SmtpClient();

smtp.Connect("smtp.gmail.com",587,false);

smtp.Authenticate("abhilash\_m.ee2018@msit.edu.in", "msit1234");

smtp.Send(email);

smtp.Disconnect(true);

return payment;

}

catch(Exception ex)

{

string msg = "Error";

}

return payment;

}

#endregion

}

}