using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

namespace ConsoleApp3

{

// common functionality to authentication the user.

public class AuthCommon

{

public AuthData Login(AuthModel authModel)

{

// check the userid and password supplied

var validUser = true; //Assuming user is authenticated successfully

var techFullName = "";

if(validUser)

{

techFullName = GetTechName(authModel.UserID);

return new AuthData {

UserId = authModel.UserID,

FullName = "Pankaj Rayal", //get the full name from database

Token = "1232433242lljsdfdsfkj34" //a token generated by the api

};

}

return null;

}

public string GetTechName(string id) {

// query the databsae for user full name based on the id

return "Pankaj Rayal";

}

}

//common class. MaintenanceRepository is injected here. in .net core it will be injected by the StartUp.Configureservices method.

public class MaintenanceCommon

{

private readonly MaintenanceRepository \_maintenance;

public MaintenanceCommon(MaintenanceRepository maintenance)

{

\_maintenance = maintenance;

}

public void RegisteraMintenance(Maintenance maintenance)

{

this.\_maintenance.RegisterMaintenance(maintenance);

}

public void AssignMintenance(Maintenance maintenance)

{

this.\_maintenance.AssignMaintenance(maintenance);

}

public void CloseMintenance(Maintenance maintenance)

{

this.\_maintenance.CloseMaintenance(maintenance);

}

}

// interface with methods

public interface IMaintenanceRepository

{

void RegisterMaintenance(Maintenance maintenance);

void AssignMaintenance(Maintenance maintenance);

void CloseMaintenance(Maintenance maintenance);

void SendAlert(Maintenance maintenance, AlertType alertType);

}

// interface is implemented here

public class MaintenanceRepository : IMaintenanceRepository

{

public void AssignMaintenance(Maintenance maintenance)

{

// this function will assign the maintenance plan to the tech.

// send the alert that the maintenance plan is assigned

this.SendAlert(maintenance, AlertType.Assign);

}

public void CloseMaintenance(Maintenance maintenance)

{

// close the maintenance plan in database.

//send the alert to the group.

this.SendAlert(maintenance, AlertType.End);

}

public void RegisterMaintenance(Maintenance maintenance)

{

// This function will add the MaintenancePlan in database

this.SendAlert(maintenance, AlertType.Start);

}

public void SendAlert(Maintenance maintenance, AlertType alertType)

{

// this function can call other services to send messages

// a log must be maintained in the database for future reference.

}

}

// enum to get the alert type

public enum AlertType

{

Assign,

Start,

End

}

// this class for Maintenance plans

public class Maintenance

{

public int MaintenanceId { get; set; }

public string TowerId { get; set; }

public string TechName { get; set; }

public string SiteAddress { get; set; }

public string SiteCity { get; set; }

public string SiteState { get; set; }

public string SitePostalCode { get; set; }

// more properties can be added here

}

// class for technician details

public class Technician

{

public int TechId { get; set; }

public string TechName { get; set; }

public string Address { get; set; }

public string City{ get; set; }

public string State { get; set; }

public string PostalCode { get; set; }

public string EmailId { get; set; }

public string PhoneNumber { get; set; }

// more properties can be added here

}

// base class to hold the credentials

public class AuthModel

{

public string UserID { get; set; }

public string Password { get; set; }

}

// this class can be used to hold the data for web. to display fullname and store token

public class AuthData

{

public string UserId { get; set; }

public string FullName { get; set; }

public string Token { get; set; }

}

}