НП "Обучение за ИТ кариера"

Име на проекта -„Beyond“

Екип: Исмет Исмет, Кристиян Калеканов

Хасково 2023г.

1.Цел на проекта:

Целта на този проект е да имитира процесите, които могат да бъдат извършени в една туристическа агенция и чрез код да се автоматизират. Използвайки програмата човек може да добавя, изтрива, взема и заменя автобуси, шофьори, градове, клиенти и пътувания от базата с данни.

2.Разпределение на ролите:

Умишлено си разделихме задачите 50% на 50% за да няма конфликти, а в случай че работата на единия е по-трудна от на другия, все намирахме нещо, което да научим или да оправим по програмата. Пример за допълнителна е оправянето на четливостта на кода или хващане на нежелани Exception-и.

3. Основни етапи в реализирането в проекта:

Основните етапи през, които преминахме са 3 на брой:

-Първи етап – първите ни стъпки в проекта бяха тук, измислихме тема, създадохме проекта и Repository в Github.

-Втори етап – във втория етап разпределихме кой над кои класове ще работи и почнахме да пишем кода. Сблъскахме се с git conflict и търсехме начини да го оправим. Едно нещо, което ни помогна да разрешим проблема беше да добавим .gitignore файл, който да игнорира нежеланите файлове причиняващи конфликт. Във края на този етап довършихме програмата.

-Трети етап – в последния етап тествахме програмата, оправяхме неизправностите, добавихме XML коментари за всеки метод. Също така опитахме да направим компонентно тестване, но ни бъркаше в базата от данни и решихме да минем без.

4. Реализация

Използвани технологии в проекта:

-EntityFramework Core 5.0.17

Описание на методите:

-Бизнес слой:

+void Add(Bus/Travel/Client/City/Driver x)

Всички бизнес класовете съдържат метода void Add(x).

Зададеният параметър(от тип Bus, Travel, Client, City или Driver) бива добавен в базата от данни.

+void Delete(int id)

Всички бизнес класовете съдържат метода void Delete(id).

Методът намира в таблицата кой запис (от тип Bus, Travel, Client, City или Driver в зависимост от кой клас се извиква) има основен ключ равен на параметъра и го изтрива.

+Bus/Travel/Client/City/Driver Get(int id)

Всички бизнес класовете съдържат метода Get(id).

Методът намира в таблицата кой запис (от тип Bus, Travel, Client, City или Driver в зависимост от кой клас се извиква) има основен ключ равен на параметъра и го връща.

+List< Bus/Travel/Client/City/Driver > GetAll()

Всички бизнес класовете съдържат метода GetAll().

Методът връща списък (от тип Bus, Travel, Client, City или Driver в зависимост от кой клас се извиква) с всички настоящи елементи от таблицата

+void Update(Bus/Travel/Client/City/Driver x)

Всички бизнес класовете съдържат метода Update(x).

Методът взима номера на първичния ключ на параметъра и актуализира този запис в таблицата, който има съвпадащ ключ.

Класът ClientBusiness има един допълнителен метод:

+Travel GetTravel(int clientId)

Този метод взема за параметър първичния ключ на клиент и връща в кое пътуване ще участва съответния клиент.

Класът TravelBusiness има 4 допълнителни метода:

+Bus GetBusByTravelId(int travelId)

В таблицата с пътувания намира този с първичен ключ равен на параметъра и връща автобуса, с който ще се провежда пътуването.

+City GetFromCity(int travelId)

В таблицата с пътувания намира този с първичен ключ равен на параметъра и връща града, от който пътуването започва.

+City GetToCity(int travelId)

В таблицата с пътувания намира този с първичен ключ равен на параметъра и връща града, в което пътуването приключва.

+List<Client> ShowClients(int travelId)

В таблицата с пътувания намира този с първичен ключ равен на параметъра и връща списък с всички клиенти, които са регистрирани за това пътуване.

-Презентационен слой

Class Display

+void PrintLogo()

Методът принтира логото на фирмата

+void ShowMainMenu()

Методът принтира главното меню.

+void Input()

Методът извиква ShowMainMenu() и след това чака вход от потребителя

Class BusDisplay

-void ShowBusMenu()

Методът принтира менюто с действия за автобусите

-void BusMenuInput()

Методът извиква ShowBusMenu() и след това чака вход от потребителя, за да се предприеме следващото действие.

-void BusAdd()

Създава се автобус с въведената от потребителя информация и се добавя в базата от данни чрез метода Add(Bus bus) от бизнес слоя

-void BusDelete()

Потребителя въвежда id-то на автобуса, който желае да изтрие и ако това е възможно, записът се изтрива от таблицата с автобуси.

-void BusGet()

Потребителя въвежда id-то на автобуса, чиято информация иска да види. Ако се намери автобус се изпечатват данните му. Ако не се намери ,се показва подходящо съобщение.

-void BusGetAll()

Изпечатват се данните на всеки един автобус в таблицата

-void BusUpdate()

Методът актуализира данните за записа, което потребителя е въвел.

Class CityDisplay

-void ShowCityMenu()

Методът принтира менюто с действия за градовете

-void CityMenuInput()

Методът извиква ShowCityMenu() и след това чака вход от потребителя, за да се предприеме следващото действие.

-void CityAdd()

Създава се град с въведената от потребителя информация и се добавя в базата от данни чрез метода Add(City city) от бизнес слоя

-void CityDelete()

Потребителя въвежда id-то на града, който желае да изтрие и ако това е възможно, записът се изтрива от таблицата с градове.

-void CityGet()

Потребителя въвежда id-то на града, чиято информация иска да види. Ако се намери град се изпечатват данните му. Ако не се намери ,се показва подходящо съобщение.

-void CityGetAll()

Изпечатват се данните на всеки един град в таблицата

-void CityUpdate()

Методът актуализира данните за записа, което потребителя е въвел.

Class DriverDisplay

-void ShowDriverMenu()

Методът принтира менюто с действия за шофьорите

-void DriverMenuInput()

Методът извиква ShowDriverMenu() и след това чака вход от потребителя, за да се предприеме следващото действие.

-void DriverAdd()

Създава се шофьора с въведената от потребителя информация и се добавя в базата от данни чрез метода Add(Driver driver) от бизнес слоя

-void DriverDelete()

Потребителя въвежда id-то на шофьора, който желае да изтрие и ако това е възможно, записът се изтрива от таблицата с шофьори.

-void DriverGet()

Потребителя въвежда id-то на шофьора, чиято информация иска да види. Ако се намери шофьор се изпечатват данните му. Ако не се намери ,се показва подходящо съобщение.

-void DriverGetAll()

Изпечатват се данните на всеки един шофьор в таблицата

-void DriverUpdate()

Методът актуализира данните за записа, което потребителя е въвел.

Class ClientDisplay

-void ShowClientMenu()

Методът принтира менюто с действия за клиентите.

-void ClientMenuInput()

Методът извиква ShowClientMenu() и след това чака вход от потребителя, за да се предприеме следващото действие.

-void ClientAdd()

Създава се клиент с въведената от потребителя информация и се добавя в базата от данни чрез метода Add(Client client) от бизнес слоя

-void ClientDelete()

Потребителя въвежда id-то на клиента, който желае да изтрие и ако това е възможно, записът се изтрива от таблицата с клиенти.

-void ClientGet()

Потребителя въвежда id-то на клиента, чиято информация иска да види. Ако се намери клиент се изпечатват данните му. Ако не се намери ,се показва подходящо съобщение.

-void ClientGetAll()

Изпечатват се данните на всеки един клиент в таблицата

-void ClientUpdate()

Методът актуализира данните за записа, което потребителя е въвел.

-void GetTravelByClientId()

Методът печата информацията за пътуването в което участва клиента с въведеното от потребителя id.

Class TravelDisplay

-void ShowTravelMenu()

Методът принтира менюто с действия за пътуванията.

-void TravelMenuInput()

Методът извиква ShowTravelMenu() и след това чака вход от потребителя, за да се предприеме следващото действие.

-void TravelAdd()

Създава се пътуване с въведената от потребителя информация и се добавя в базата от данни чрез метода Add(Travel travel) от бизнес слоя.

-void TravelDelete()

Потребителя въвежда id-то на пътуването, което желае да изтрие и ако това е възможно, записът се изтрива от таблицата с пътувания.

-void TravelGet()

Потребителя въвежда id-то на пътуването, чиято информация иска да види. Ако се намери пътуване се изпечатват данните му. Ако не се намери ,се показва подходящо съобщение.

-void TravelGetAll()

Изпечатват се данните на всяко едно пътуване в таблицата

-void TravelUpdate()

Методът актуализира данните за записа, което потребителя е въвел.

-void TravelGetBus()

Намира се пътуването с id-то въведено от потребителя и се печатат данните на автобуса участващ в него.

-void TravelGetFromCity()

Намира се пътуването с id-то въведено от потребителя и се печатат данните на града, от който тръгва пътуването.

-void TravelGetToCity()

Намира се пътуването с id-то въведено от потребителя и се печатат данните на града, в който свършва пътуването.

-void ShowClients()

Намира се пътуването с id-то въведено от потребителя и се печатат данните на всички клиенти участващи в него.

5.Използвана литература

-Данните предоставени от ИТ Кариера

- <https://stackoverflow.com/questions/39434878/how-to-include-related-tables-in-dbset-find>

- Видеа в интернет