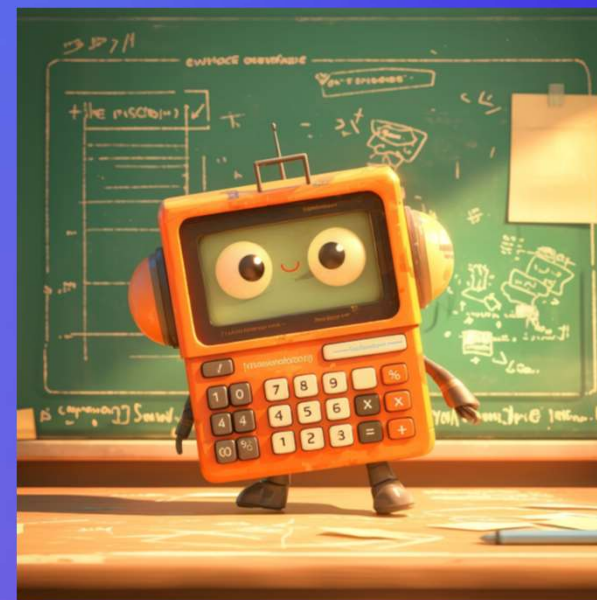


## C# Developer. Basic

TelegramBot учета ежемесячного контроля затрат подотчетных средств для передвижения сотрудников на личном транспорте и транспорте в компании монтажно-сервисной организации специализирующейся на системах ОВиК (отопление вентиляции и кондиционирования).



Меня хорошо видно  
& слышно?



## Защита проекта



Тема: Telegram Bot учета ежемесячного контроля затрат подотчетных средств для передвижения сотрудников на личном транспорте и транспорте в компании монтажно-сервисной организации специализирующейся на системах ОВиК/HVAC (отопление вентиляции и кондиционирования).



**Алексей Благов**

Инженер систем ОВиК / HVAC

# План защиты

Цель и задачи проекта

Какие технологии использовались

Что получилось

Выводы

Вопросы и рекомендации



# Цель и задачи проекта













Цель проекта: Автоматический сбор данных, возможность добавления расходов в режиме реального времени, а также отчеты по расходам за месяц. Это позволит сотрудникам легко и быстро фиксировать свои затраты, а руководству — контролировать бюджет и компенсировать траты.

1. Задача, разработать посредством полученных знаний на курсе TelegramBot
2. Изучить дополнительный материал и skills для реализации задачи
3. Внедрить TelegramBot в работу компании упростив работу сотрудникам
4. Отрефлексировать работу, добавить необходимый инструментарий в рабочий TelegramBot

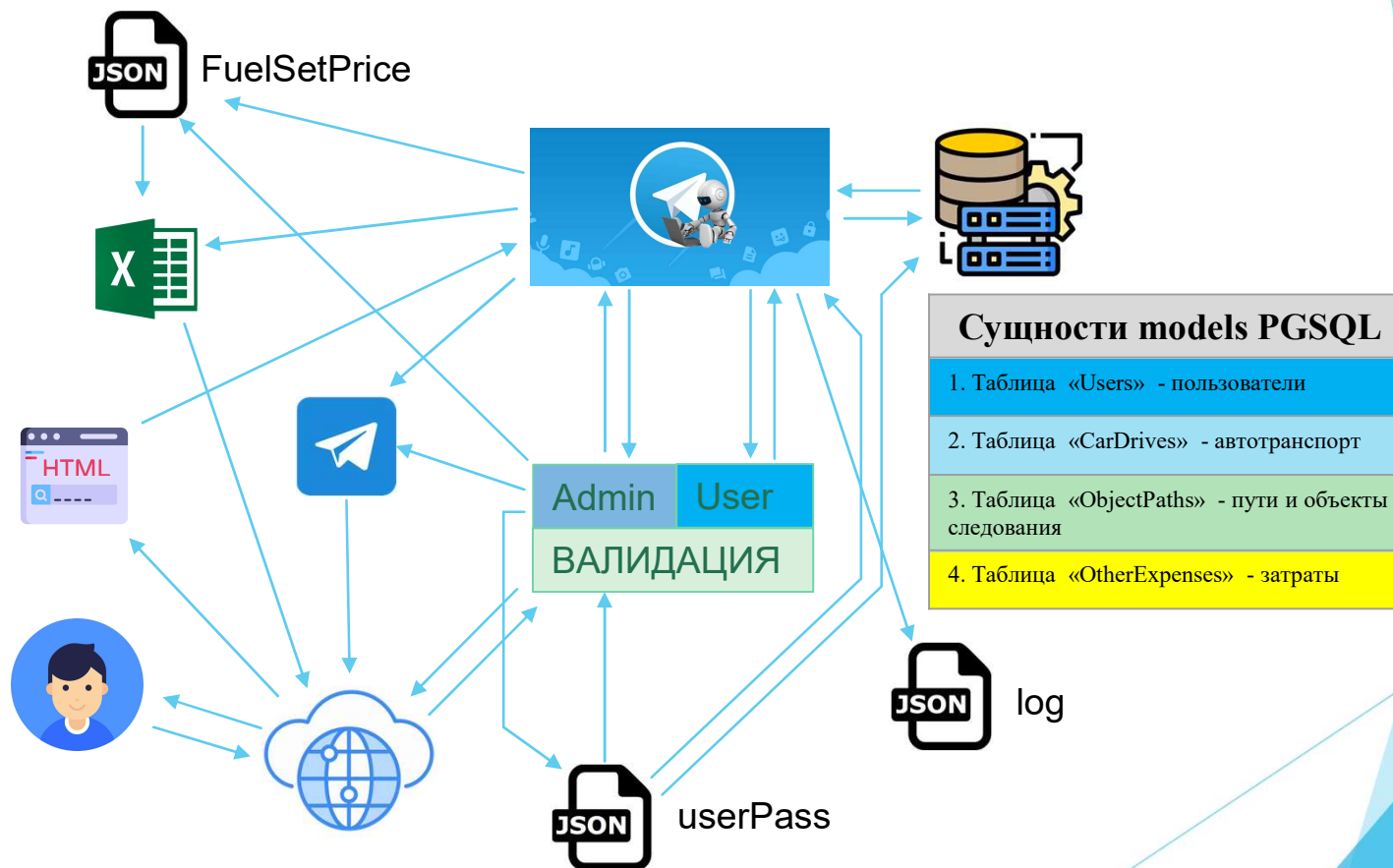
Проект: [https://github.com/Alexey-Blagov/Project\\_Work\\_My\\_Telegram\\_bot\\_ver.1](https://github.com/Alexey-Blagov/Project_Work_My_Telegram_bot_ver.1)

# Какие технологии использовались

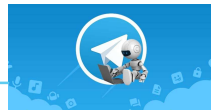
1. Dot NET 8.0, Visual Studio   
2. Telegram Bot API ver. 22.3.0 
3. EF Core, PGSQL 17  
4. EPPlus пакет формирования Excel 
5. Html Agility Pack 
6. Microsoft Extensions Configuration Json 
7. Контроль версий Git CI/CD 
8. Observer pattern



## Схема работы Telegram Bot



# Структура базы данных



Сущность «User»	
TgId	Внешний ключ ChatId
TgUsername	Имя при регистрации
UserRol	Идентификатор права доступа
UserName	ФИО пользователя
JobTitle	Должность в компании
Personalcar	Персональный автомобиль
ObjectPaths	Список объектов все пути
OtherExpenses	Список все траты пользователя

Сущность «CarDrive»	
CarId	Ключ
CarName	Марка машины
isPersonalCar	Тип машины собственная/компании
CarNumber	Номер машины
GasConsum	Средний расход топл. на 100 км. пути
TypeFuel	Марка топлива
PersonalId	Внешний кл. пользователя по БД на которого зарегистрирован авто
ObjectPaths	Список поездок данного авто

Сущность «OtherExpenses»	
ExpId	Ключ
NameExpense	Наименование затрат
Coast	Стоимость затрат
DateTimeExp	Дата затрат
UserId	Ссылочный ключ пользователя на затраты
UserExp	Экземпляр класса User на трарты

Сущность «ObjectPath»	
IdPath	Ключ
ObjectName	Наименование пункта назначения объекта
PathLeng	Длина пути следования
DatePath	Дата поездки
CarId	Ссылочный ключ на класс CarDrive
CarDrive	Экземпляр класса CarDrive на данный путь
UserId	Ссылочный ключ пользователя на данный путь
UserPath	Экземпляр класса User на данный путь



# Структура кода

## Классы работы с EF Core DB

**DataBaseHandler** (Классы обработчик сообщений сохранения и чтения DB)

**RepositoryReportMaker**  
(Классы получения данных из DB)

**ApplicationContext** (Классы контекст DB)

### ClassDB (Классы модели сущностей)

Класс	User
Класс	CarDrive
Класс	ObjectPath
Класс	OtherExpenses

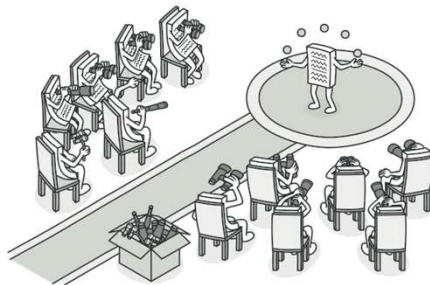
### Configurations (Настройка миграций DB)

Класс	UserConfigurations
Класс	CarDriveConfigurationsBase
Класс	ObjectPathDriveConfigurations
Класс	OtherExpensesConfigurations

### Migrations (Сервисная папка миграций)

Класс	20250216080154_InitCreate
Класс	ApplicationContextModelSnapshot

## Pattern Observer



## Классы работы с файлами данных

**FileExcelHandler** (Класс формирования Excel файл)

**PassUser** (Класс получения данных паролей из файла userPass.json и сохранение пароля User в файл)

**FuelPrice** (Класс получения информации по цене топлива с сайта выгрузка в файл FuelSetPrice.json )

**LoggerTgBot** (Класс логирования log.json )

## Основные классы обработки и взаимодействия с Telegram API

**Program** (Класс точки входа в программу, формирования прослушивания TelegramBot)

**MessageProcessing** (Основной класс обработки сообщений TelegramBot)

**KeyBoardSetting** (Вспомогательный класс формирования и обработки Инлайн, Реплай клавиатур для TelegramBot)



# Структура меню

Регистрация вход пользователя

Меню функции User (Пользователь)

/start - запуск 16:08

Сообщение

Пользователь Администратор

Меню функции Admin  
(Администратор)

Сообщения

Установка пароля User Стоимость бензина

Регистрация автопарка компании Вывести отчет по User

Смена статуса Admin/User

Меню выбора типа топлива: 16:19

Стоимость AI-92

Стоимость AI-95

Стоимость ДТ

/Main - запуск основного меню 16:08

Сообщения

Профиль Вывести отчет

Регистрация поездки Регистрация трат

Смена статуса Admin/User

Регистрация/изменение профиля: 16:32

Ф.И.О Должность

Марка машины Госномер

Используемое топливо Средний расход л. на 100 км

Закончить и сохранить

Регистрация автопарка компании 16:19

Меню регистрации: 16:19

Марка машины Госномер

Расход на 100 км. Используемое топливо

Закончить и сохранить

Регистрация доп. трат: 16:10

Регистрация доп. трат: 16:10

Наименование затрат

Сумма 00,00 руб Дата затраты

Закончить и сохранить

Регистрация поездки: 16:10

Регистрация поездки: 16:10

Точка назначения

Полный путь в км

Собственный транс Дата поездки

Закончить и сохранить

Меню вывода отчета: 16:17

Сообщения

Отчет за текущий месяц Отчет за выбранный месяц

Возврат в основное меню

# Выводы

1. Создал готовый к использованию продукт применяя полученные знания на курсах OTUS. Повысил свои skills подключил дополнительные пакеты изучил их функционал.
2. Для обработки ввода с клавиатуры Chat bot в проекте реализован паттерн “наблюдатель” .
3. Есть идеи дополнительных инструментами для улучшения проекта. Добавить реализацию - автомашина в ремонте / в пути. Переработать интерфейс ввода/вывода на использование инлайн клавиатуры.

Времени потратил порядка 2х месяцев

# Вопросы и рекомендации



если есть вопросы



если вопросов нет

Спасибо за внимание!

