

DocOps: Документирование баз данных

Содержание

1. Что? Как? Зачем	1
2. Кодим	2
3. Delivery Pipeline	4

ParHandler@gmail.com

 @ParHandler  @ParHandler

1. Что? Как? Зачем

- Что?
 - Документ в пространстве Confluence, отражающий актуальное состояние базы данных из master(релизной, фича)-ветки репозитория.
- Как? Инструменты в работе:
 - AsciiDoctor.
 - Модель данных в проекте .net.
 - Генератор статической документации.
- Зачем?
 - Код и документация единый артефакт... ну почти единый.
 - Сокращение затрат на поддержку документации в актуальном состоянии.
 - Осваивать DocOps не так сложно как казалось бы на первый взгляд.
 - Дружить с DocOps значит быть в тренде.)

2. Кодим

RegularPaymentDetailEntity.cs

```
using System.Collections.Generic;

namespace Infrastructure.DB.Entities
{
    /// <summary>
    /// Детали регулярных переводов
    /// </summary>
    public class RegularPaymentDetailEntity
    {
        /// <summary>
        /// Идентификатор ППП
        /// </summary>
        public long UnId { get; set; }

        /// <summary>
        /// Идентификатор клиента
        /// </summary>
        public long Cuid { get; set; }

        /// <summary>
        /// Номер карты списания
        /// </summary>
        public string CardNumber { get; set; }

        /// <summary>
        /// Счет списания
        /// </summary>
        public string AccountNumber { get; set; }

        /// <summary>
        /// Название ППП
        /// </summary>
        public string Name { get; set; }

        /// <summary>
        /// Сумма операции
        /// </summary>
        public decimal Amount { get; set; }

        /// <summary>
        /// Валюта операции
        /// </summary>
        public string Currency { get; set; }

        /// <summary>
        /// Счет списания
    }
}
```

```

    /// </summary>
    public string AccountTo { get; set; }

    /// <summary>
    /// Назначение платежа
    /// </summary>
    public string PaymentPurpose { get; set; }

    /// <summary>
    /// Имя/название получателя
    /// </summary>
    public string Recipient { get; set; }

    /// <summary>
    /// БИК банка получателя
    /// </summary>
    public string Bic { get; set; }

    /// <summary>
    /// ИНН получателя
    /// </summary>
    public string Inn { get; set; }

    /// <summary>
    /// КПП получателя
    /// </summary>
    public string Kpp { get; set; }

    /// <summary>
    /// Кор счет банка получателя
    /// </summary>
    public string CorrAccount { get; set; }

    /// <summary>
    /// Сумма налогов
    /// </summary>
    public decimal? TaxAmount { get; set; }

    /// <summary>
    /// УИН/УИП
    /// </summary>
    public string Uip { get; set; }

    /// <summary>
    /// Связанные транзакции
    /// </summary>
    public List<TransactionEntity> Transactions { get; set; }
}
}

```

3. Delivery Pipeline

