

Тестовое ТЗ “Регулярные платежи”

Регулярный платеж - это платеж который списывается по регламенту (каждый заданный период)

Реализовать систему хранения и списания платежей.

Должно быть Хранилище (PostgreSQL)

Хранение информации о платежах и результатах их списания

1. Инструкции регулярных платежей

Данные

- a. ФИО плательщика
- b. ИНН
- c. № Карты плательщика
- d. Расчетный счет получателя
- e. МФО получателя
- f. ОКПО получателя
- g. Наименование получателя
- h. Период списания - каждую N минут/часов/дней
- i. Сумма платежа

2. Проводки - Результаты их списания

- a. Дата/время проводки
ID - инструкции платежа
- b. Сумма платежа
- c. Статус -
A - Активна
S - Сторнированная

Сервис работы с данными (DAO Rest-API) - должен быть отдельный WEB сервис

1. Методы CRUD платежей

- a. Create Update Delete
- b. Метод получения платежа по ID
- c. Метод получения списка платежей по ИНН
- d. Метод получения списка платежей по Получателю (ОКПО)

2. Методы CRUD проводок

- a. Create Update Delete
- b. Метод получения проводки по ID
- c. Метод получения списка проводок по платежу

Сервис Бизнес-логики (Rest-API)

1. Создание платежа

- a. Валидация входящих данных

2. Списание платежа

- a. Создание проводки по платежу
- b. Проверка необходимости списания
- c. Сторнировка проводки (удаление)

3. Получение данных

- a. Получение всех платежей по Плательщику

- b. Получение всех платежей по получателю
- c. Получение истории списания платежа

Регламентный модуль (JAR)

Данный модуль должен использовать Сервис Бизнес-логики

1. Запуск процесса списания
 - a. Выбрать все платежи
 - b. По каждому платежу проверить дату/время последнего списания, если текущая дата/время более чем Дата/время списания + Период списания - списать платеж (создать проводку)