существующие и потенциальные проблемные места

Открыт прямой доступ к API сервисов.

Онлайн-магазин и CRM используют одну БД (ShopDB).

MES рассчитывает стоимость производства изделия и поддерживает производство.

Онлайн-магазин и MES - нет интеграции, значит стоимость попадает в магазин через шаренную БД.

Онлайн-магазин (API) загружает файлы и работает с заказами.

Долго рассчитывается стоимость заказа.

Создание заказа внешними продавцами через API напрямую в MES.

Вероятно так же MES загружает файлы в FS для расчета и создания заказа.

Потеря заказов клиентами онлайн-магазина.

Потеря заказов клиентами API.

MES, система долго прогружается.

Одна команда для всех сервисов.

Сильная связь между сервисами, отсутствие версионирования.

Задержки релизов.

Неполная/непродуманная статусная модель.

Отсутствие E2E автотестов.

инициативы

Закрыть прямой доступ к API сервисов и использовать API-шлюз

Выделить из API онлайн магазина загрузку файлов в отдельный сервис.

Выделить из MES сервис расчета стоимости производства.

Сделать интеграцию всех сервисов с брокером.

Разделить БД для онлайн-магазина и CRM.

Масштабировать сервис расчета стоимости заказа.

Унифицировать процесс создания заказа.

Выделить отдельные команды для сервисов онлайн-магазина, MES и CRM.

Проработать статусную модель.

Внедрить версионирование API.

Внедрить публикацию по фичам.

Написать автотесты.

Добавить: Мониторинг, Трейсинг, Логирование, Кеширование, алёртинг

Оптимизировать запросы и модели данных.

целевая архитектура через полгода

Целевая:

У сервисов онлайн-магазина, MES и CRM отдельные команды.

Проработана статусная модель, процесс создания заказа и модели данных, у каждого сервиса своя БД.

Сервисы общаются между собой через брокер.

Внедрена публикация по фичам.

Все внешние запросы через API-шлюз (версионирование).

Настроены: Мониторинг, Трейсинг, Логирование, Кеширование, алёртинг.

Расчет стоимости отдельный масштабируемый сервис.

Загрузка файлов отдельный масштабируемый сервис.

Написаны автотесты.

Оптимизированы запросы.

Описание:

В первую очередь создаем команды для основных сервисов и даем им возможность работать и публиковаться независимо.

Прорабатываем статусную модель и модели данных и выделяем каждому сервису свою БД.

Внедряем версионирование в API сервисов, внешние запросы переключаем на API-шлюз, а взаимодействие между сервисам делаем асинхронным.

Настраиваем кеширование, логирование, мониторинг, трейсинг, алертинг.

Выделяем нагруженные сервисы и масштабируем отдельно.

Оптимизируем.

Если бы можно было выполнить только три пункта, то:

1. Выделил отдельные команды сервисам.
2. Проработал модели данных и статусную модель разделив по сервисам.
3. Настроил асинхронное взаимодействие между сервисами.