



Анализ и проектирование

Введение

ФИТ Лекция №1. 25'

Карта основных навыков аналитика

Сбор данных и целеполагание

- Интервью
- CustDev
- Анализ информации
- Анализ данных

Концептуальное проектирование

- BPMN
- ФТ, НФТ
- Интеграции
- Базы данных
- Интерфейсы

Разработка и доставка

- Бэклог
- Декомпозиция
- Планирование и зависимости
- Управление изменениями
- Качество и внедрение

Знания и эксперименты

- Метрики
- А/В тесты
- Управление гипотезами
- Продуктовое видение

Маркетинг и продвижение

- Анализ NPS
- Стратегия удержания
- Продуктовое позиционирование

Экономика и продажи

- Pre-sales
- Оценка себестоимости
- Монетизация
- ROI

3 классические роли в аналитике

	Сбор данных и целеполагание	Концептуальное проектирование	Разработка и доставка	Управление знаниями и гипотезами	Маркетинг и удержание	Экономика и продажи
Продуктовый аналитик						
Бизнес аналитик						
Системный аналитик						

Критичный
уровень

Средний
уровень

Низкий
уровень



Инструменты продуктового анализа



Продуктовый анализ

Это **изучение поведения пользователей** с опорой на **данные**, а не на интуицию или субъективные впечатления.



Цели продуктового анализа

1. **Определять ценность** продукта в целом и отдельных фич в частности.
2. **Задавать приоритеты** команды исходя из соотношения пользы к затратам - **фокус на важном**.
3. **Повышать выгоду** бизнеса от работы продукта и развития в правильном направлении.
4. **Доносить особенности реального использования** продукта до бизнеса.

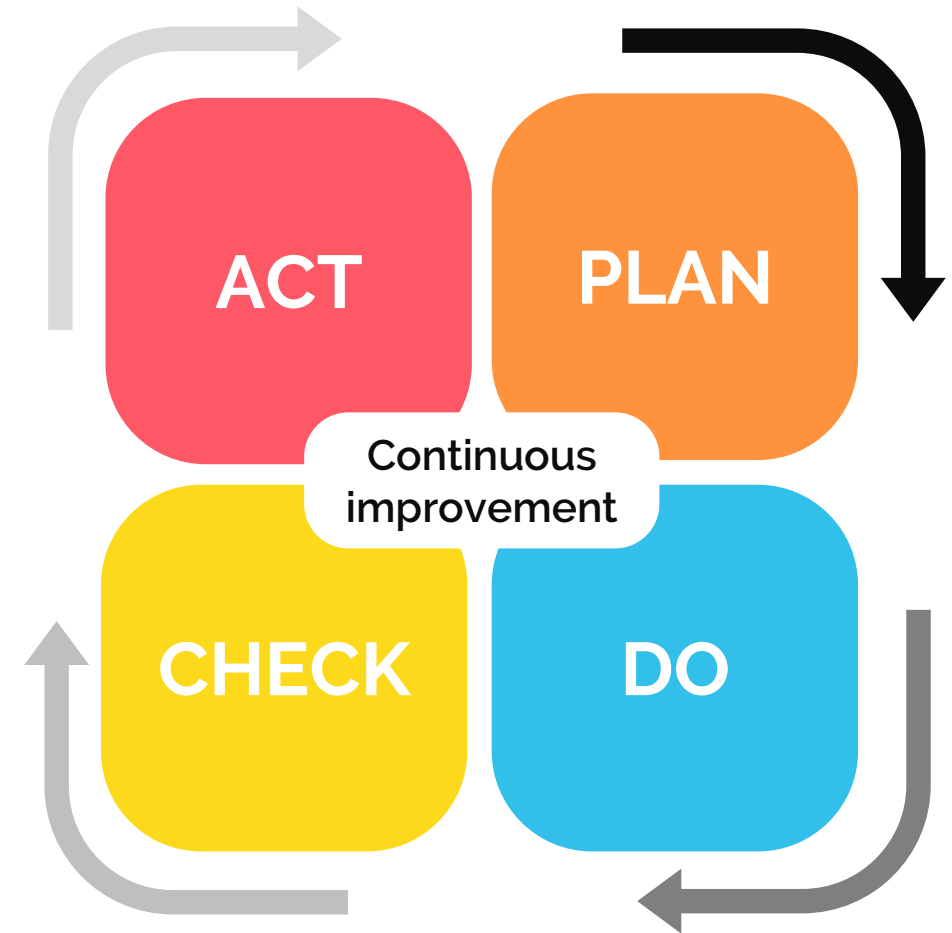
Состав продуктового анализа

1. **Стратегия** - цели продукта и варианты их достижения.
2. **Метрики** - показатели, ключевые в реализации стратегии
Примеры: число просмотров страницы (*должно расти*), время заполнения заявки (*должно сокращаться*), средний чек (*должен расти*).
3. **Данные** - сбор и анализ данных
Примеры: логи, дашборды, Яндекс-метрики, БД
4. **Гипотезы** - задачи для команды разработки (*= старт системного анализа*)
Пример: считаем, что сотрудник долго ищет задачу, потому что нет удобных фильтров и персонализации. Начинаем формулировать фичу.

Стратегия продукта

1. Ключевые цели
2. Текущий и желаемый показатель по каждой из них
3. Как достигаем - предполагаемые шаги, варианты реализации

Стратегию нельзя подготовить раз и навсегда - она должна развиваться вместе с продуктом.



Изменчивость стратегии

На старте разработки.

- **Цель** - перевод всех агентов(продажи) на новую систему расчётов для ускорения оборота средств и повышения точности данных.
Ключевые показатели: %агентов (=100%), Цикл оборота средств с продаж <3 дней
Гипотеза: автоматический сбор данных на стороне ген.компании + их достоверность с расшифровкой каждой суммы = процесс не требует ручной обработки и достигается нужная скорость

сейчас.

- Продукт разработан, изначальные цели достигнуты. Сейчас на поддержке, развитие не в приоритете. **Цель** - стабильная работа без сбоев.
Ключевые показатели: минимизируем число обращений в саппорт, ошибок в логах.
Гипотезы: выявление системных сбоев и их решение = негативный тренд багов


Изменчивость стратегии

На старте разработки.

- **Цель** – Оцифровать большой производственный процесс, устранить ручной труд по внесению и чтению информации. Сократить ошибки при работе с данными
Гипотеза: это повысит продуктивность и снизит затраты на коммуникации и ручной сбор/валидацию данных.
Метрики: увеличить мощность производства в 10 раз

через год.

- Постановка целей эволюционирует от глобальной бизнесовой, поставленной на годы, к целям каждого модуля. Например - повышение точности прогноза количества закупаемых деталей и материалов.
Гипотеза: расчёт по новым формулам даст более точное число деталей, необходимых к закупке до начала ремонта -> снизятся затраты.
Метрики: время ожидания деталей, объемы запасов



Этапы подготовки стратегии

Определение целей продукта

- Связка с бизнес-анализом
- Построение дерева целей

1

Достижение Product Market Fit

Мостик между бизнесовыми и
пользовательскими целями

2

Подбор метрик

Выбираем показатели,
наиболее подходящие для
нашего продукта на текущем
этапе развития

3

Анализ данных

Выявление закономерностей,
визуализация результатов для
принятия решений

4

Формулировка гипотез для развития

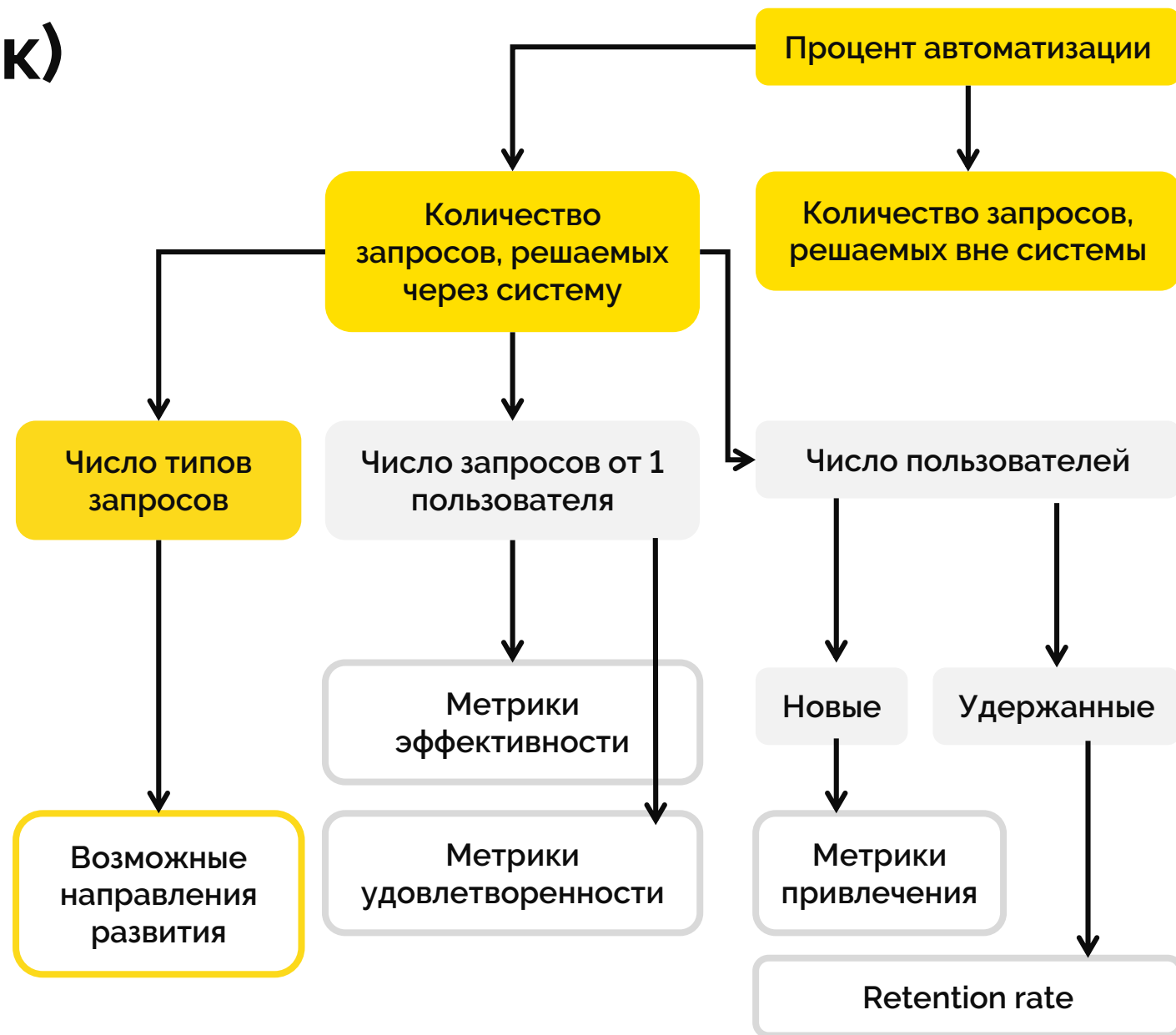
На основе данных строим
гипотезы, чтобы поставить
задачи, которые помогут нам в
достижении целей

5

Дерево целей (метрик) продукта

Даёт **иерархию** целей
и **корреляцию** между ними

1. Определить **ключевую бизнес-цель**
2. Разложить на составляющие (повторять до атомарного уровня)
3. Определить метрики, значимые для нижних уровней
4. **Запаздывающие показатели** - глобальные цели (Желтые). Не актуальны в моменте, складываются из нижних уровней.
5. **Опережающие показатели** - текущие цели (серые). Отражают "здесь и сейчас".
6. **Задача** - найти корреляцию между опережающими и запаздывающими



Product Market Fit

Product Market Fit (PMF) - состояние, при котором продукт соответствует ожиданиям аудитории.

Три составляющих PMF:

- **Клиент** (вы знаете, кому нужен ваш продукт),
- **Проблема** (вы решаете конкретную проблему Клиента),
- **Решение** (Клиент готов платить за это Решение).

PMF - инструмент для связи целей бизнеса и целей пользователя.

Бережливая приоритизация

- **Quick Win** - мало затрат, много пользы
- **Big Bet** – много затрат, много пользы
- **Maybe** – мало затрат, мало пользы
- **Time sinks** – много затрат, много пользы

Quick win

Лучший результат с наименьшими затратами

Maybe

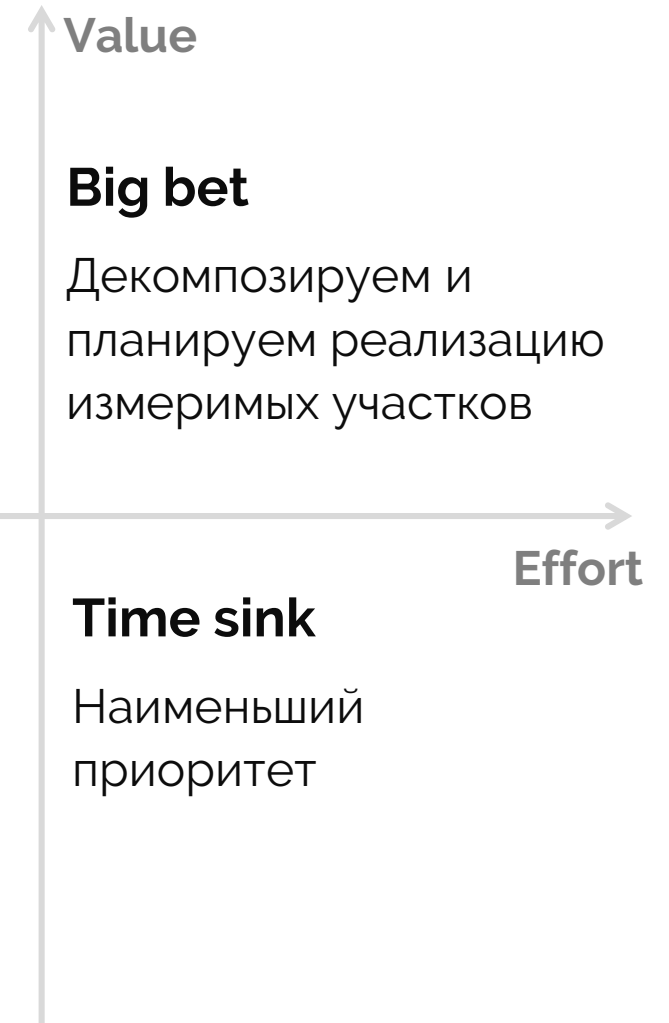
Заполняем мелкими задачами простои между крупными фичами

Big bet

Декомпозируем и планируем реализацию измеримых участков

Time sink

Наименьший приоритет





Виды метрик

1. **Финансовые** - прямо измеряют денежные показатели
2. **Ценность продукта** - косвенно влияют на денежные показатели, вносят вклад в финансовые метрики более высокого уровня
3. **Удержание пользователей** - дают процент постоянных пользователей за период
4. **Качественные** - говорят об эффективности решения пользовательской задачи
5. **Стабильность** - технические метрики, отражают доступность сервиса пользователям

Ценностные метрики

"Пиратские" метрики AARRR

1. **Acquisition** — привлечение
2. **Activation** — активация
3. **Retention** — удержание
4. **Revenue** — доход
5. **Referral** — рекомендации

Интересно, что 4 и 5 в разных источниках меняются местами.



Ценностные метрики - AARRR

Уровень	Привлечение	Активация	Удержание	Доход	Рекомендации
Ключевой вопрос	Откуда приходят клиенты?	Как привлеченные клиенты становятся активными?	Как удержать клиента?	Как монетизировать клиента?	Как мотивировать пользователя рекомендовать продукт?
Цель	Понимать путь клиента и оптимизировать его	Понять, сколько клиентов осознали ценность	Клиент 1 возвращается в сервис, клиент 2 - нет. Понять, почему.	Убедиться, что сервис начинает окупаться (прямо или косвенно)	Когда клиент привлекает других пользователей?

Ценностные метрики - AARRR

Уровень	Привлечение	Активация	Удержание	Доход	Рекомендации
Метрики	<ul style="list-style-type: none">• Трафик на сайт с каждого маркетингового канала.• SEO метрики• SMM метрики• Bounce rate• Рекламные метрики: показы, переходы, цена за клик, цена за установку, кликабельность, число лидов	<ul style="list-style-type: none">• Time to value• Conversion rate• Drop-off rate• Engagement rate• CPAU (цена активного пользователя)• Количество активных пользователей за период	<ul style="list-style-type: none">• Retention rate• Churn rate• Email open rate• RSS feed sign ups• Число просмотров от числа подписчиков	<ul style="list-style-type: none">• AOV (средний чек)• CAC - стоимость привлечения• LTV - общий доход по клиенту• ROI - возврат инвестиций• ARPU - средний доход с клиента	<ul style="list-style-type: none">• Процент рекомендующих пользователей• Процент пользователей по рекомендации• NPS• K-factor - сколько пользователей приводит существующий пользователь



Ценностные метрики - AARRR

Рассмотрим на примере

1. **Привлечение.** Число переходов в сервис с разбивкой по каналу привлечения.
2. **Активация.** Средняя глубина просмотра $> N$ или Время просмотра страниц $> T$.
3. **Удержание.** Retention Rate за месяц $> X$.
4. **Доход.** Число запросов в поддержку должно снижаться по разделам, покрытым справкой.
5. **Рекомендации.** Число переходов по прямой ссылке (в нашем случае побочный аспект)

Retention rate

Индекс удержания клиентов - сколько пользователей повторно воспользовались сервисом за период (зашли почитать новости, совершили покупку и т.п.)

- Действующий клиент для бизнеса **гораздо выгоднее** нового!
- **Привязка к периоду** обязательна
- Для каких-то случаев **может не подходить**:
 - Нерегулярные, конечные задачи (нашёл работу, снял квартиру, починил телефон)
 - Может сбивать с толку - постоянные пользователи, не приносящие денег
- **Значение зависит от индустрии**. Значение Netflix 66%/год - хорошее для ретейла, катастрофа для корпоративных SaaS типа Atlassian.

В сложном B2B тоже может быть важен - пользователь один раз обжёгся и навсегда вернулся в Excel или в почту.

Качественные метрики

Отвечают за удовлетворённость клиентов и эффективность решения их задач

Показатели клиентского обслуживания

Метрики
клиентского
опыта

Индекс
лояльности
(NPS)

Индекс
удовлетворённ
ости (CSAT)

Индекс
потребительск
их усилий (CES)

Referral rate

Показатели
эффективности
процессов

Объём
запросов

Показатель
открытых
тикето

Средняя
скорость
ответа

Среднее время
разрешения
запроса

Качественные метрики

Показатели эффективной работы пользователя

Скорость
прохождения
полного пути

Глубина
просмотра
(Page depth)

Скорость
выполнения
операций
(Load time,
latency)

Процент
отказов / Task
success rate

Время
просмотра
страницы /
Time-on-task

UX-метрики

Число кликов

User error rate

Search vs.
Navigation

Шкала
удобства
использования
(SUS)

Метрики стабильности

Отражают доступность выполнения ключевых задач и производительность

1. **SLI** – Service Level Indicator - **Факт**. Текущее измеренное значение показателя.
 - **Например**, 99% времени доступности сервиса за месяц **или** 1MB/сек средняя скорость загрузки или 1000 сессий пользователей одновременно.
2. **SLO** - Service Level Objective - Цель, Целевое значение индикаторов из SLI.
 - **Например**, стремимся, чтобы даунтайм при релизе был меньше 1 минуты
3. **SLA** - Service Level Agreement - Обязательство. Зафиксированное в договоре значение индикатора из SLI.
 - **Например**, "доставим пиццу за час или бесплатно".



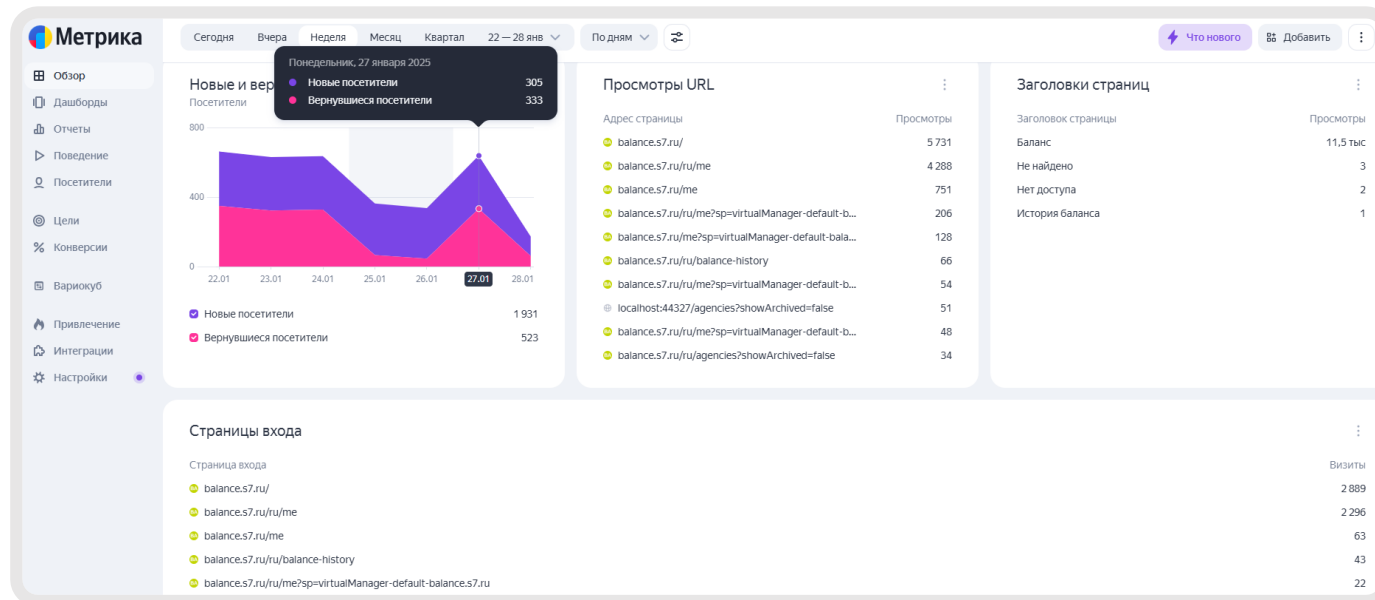
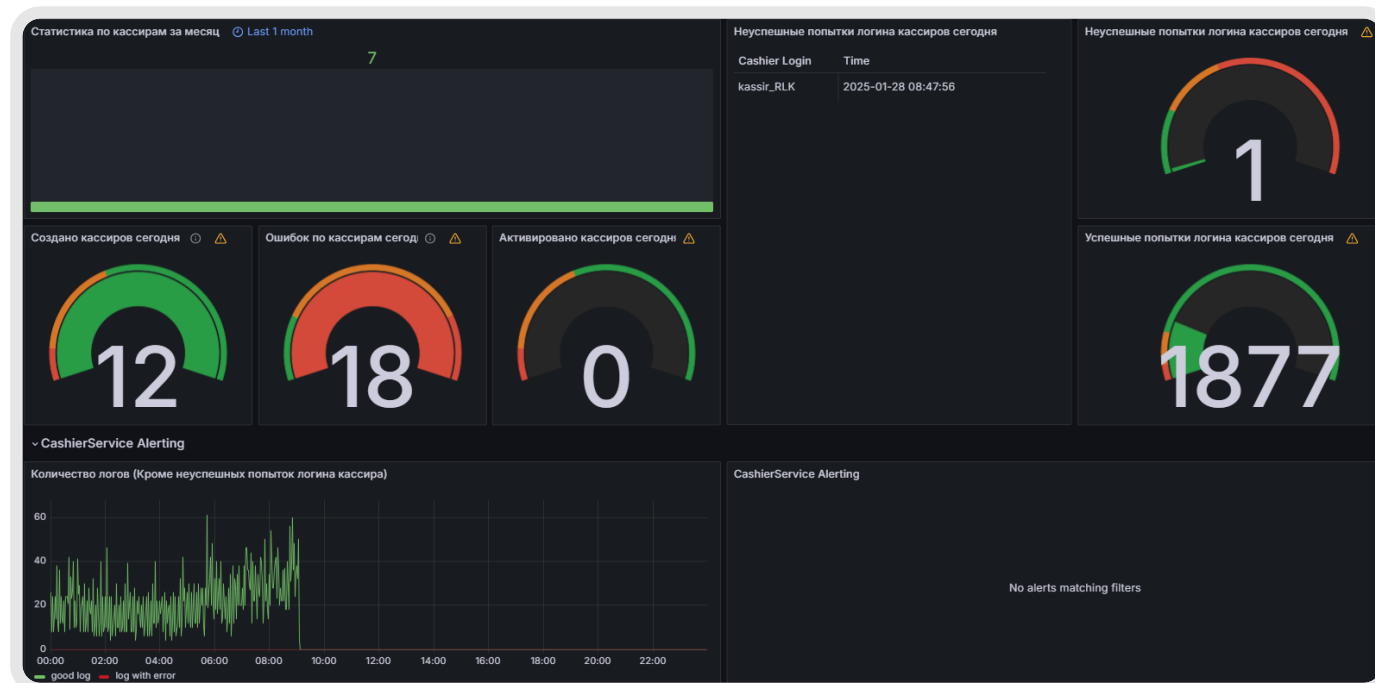
Подходы к измерению метрик

Продуктовый анализ основан на фактах, а не на эмоциях

1. **Измеримые цели**, которые можно перевести в деньги
2. **Измеримые результаты**, которые можно сравнивать с предыдущими значениями
3. **Декомпозиция на задачи** исходя из кратчайшего пути к максимальной пользе

Источники данных

1. Логи и дашборды (e.g. Kibana+Grafana)
2. База данных приложения
3. Результаты UX-исследований
4. Веб-аналитики (Яндекс-метрика, Google analytics)





Заключение

1. Продуктовый анализ (явно или неявно) есть всегда - это измерение ценности задачи для бизнеса и поиск новых решений
2. Ценность задач и приоритеты продукта меняются со временем.
3. Ценность должна обосновываться данными.
4. Метрик, подходов и инструментов много. Часть мы обсудили



Инструменты бизнес анализа



Подходы к бизнес-анализу

1. **Анализ требований** – Сбор и документирование информации, полученной разными способами: Интервью, воркшопы, мозговые штурмы, анализ документов
2. **Процессный** подход – Подход, направленный на достижение цели некоторого процесса: Моделирование процесса (BPMN, EPC), выявление узких мест, консалтинг
3. **Data-driven** подход – Подход, направленный на поиск решения через анализ данных и измерения на данных
4. **Agile** подход – Итеративная генерация решений, ориентирован на гибкость и изменения
5. **Анализ изменений** (Change Impact) – Разработка стратегии внедрения изменений
6. **SWOT** анализ – Поиск решения через оценку сильных и слабых сторон, возможностей и угроз

Когда использовать

1. **Анализ требований** – Старт нового проекта и сбор первичной информации и потребностей. Требуется четко понимать бизнес-потребность
2. **Процессный** подход – Оптимизация существующей технологии. Есть понятные проблемы с эффективностью. Автоматизация рутины
3. **Data-driven** подход – Нет известной потребности или проблемы от бизнеса, но есть задача поиска возможности: выявление закономерностей, трендов, провалов
4. **Agile** подход – Цель очень «высоко» и на пути достижения много рисков, большой объем работ
5. **Анализ изменений** (Change Impact) – Происходит изменение процесса или технологии. Требуется оценить риски и последствия. Подготовиться.
6. **SWOT** анализ – Оценка текущего состояния для принятия стратегических решений



Подходы разные, алгоритм похожий

1. **Целеполагание** – Напрямую вытекает из продуктового анализа
2. **Коммуникационная стратегия** – определяет участников и формат взаимодействия
3. **Сбор требований** – применение различных методов и структурирование
4. **Моделирование** – «Рисуй, чтобы победить»
5. **Анализ проблем и возможностей** – Ревью полноты, непротиворечивости, пропускной способности и т.д.
6. **Управление изменениями** – Оценка влияния изменений и оповещение участников
7. **Поддержка разработки** – синхронизация с действительностью
8. **Контроль результата** – Доводить до результата и оценивать его

Сущности и инструменты

1. Стратегия продукта (Устав) – Формирует цели и подходы. Документ или База знаний
2. Стратегия коммуникации – Формирует правила и подходы взаимодействия. Документ или База знаний
3. Дерево целей и метрик – Формирует зависимости глобальных целей от конкретных функций. Схема
4. Компонентная карта – Формирует самостоятельные блоки продукта и связи между ними. Схема
5. Roadmap – Формирует базовый план получения результатов. Gantt

Модель продукта

1. Статическая (структурная) – Описывает продукт вне последовательности действий: Субъекты, Роли, Портреты пользователей, их цели и задачи. Бизнес-сущности и атрибуты. Матрицы доступа и логические связи между объектами и субъектами: UML(Классы, Объекты, Компоненты), DFD
2. Динамическая (Поведенческая) – Описывает последовательность действий, их порядок и зависимость:
 1. UML (BUC, UC)
 2. UML (Activity, Sequence)
 3. BPMN
 4. EPC
 5. IDEF
 6. VSM



Заключение

1. Поиск и структурирование информации – важный этап и системно его выполняют не многие
2. Осознав цели и их измерения, мы выстраиваем зависимость целей от элементов продукта: Объектов и сценариев
3. Зависимости проверяются и считываются графически. Для того существует множество подходов
4. Описанная модель реализации превращается в задачи продуктового бэклога



Инструменты системного анализа

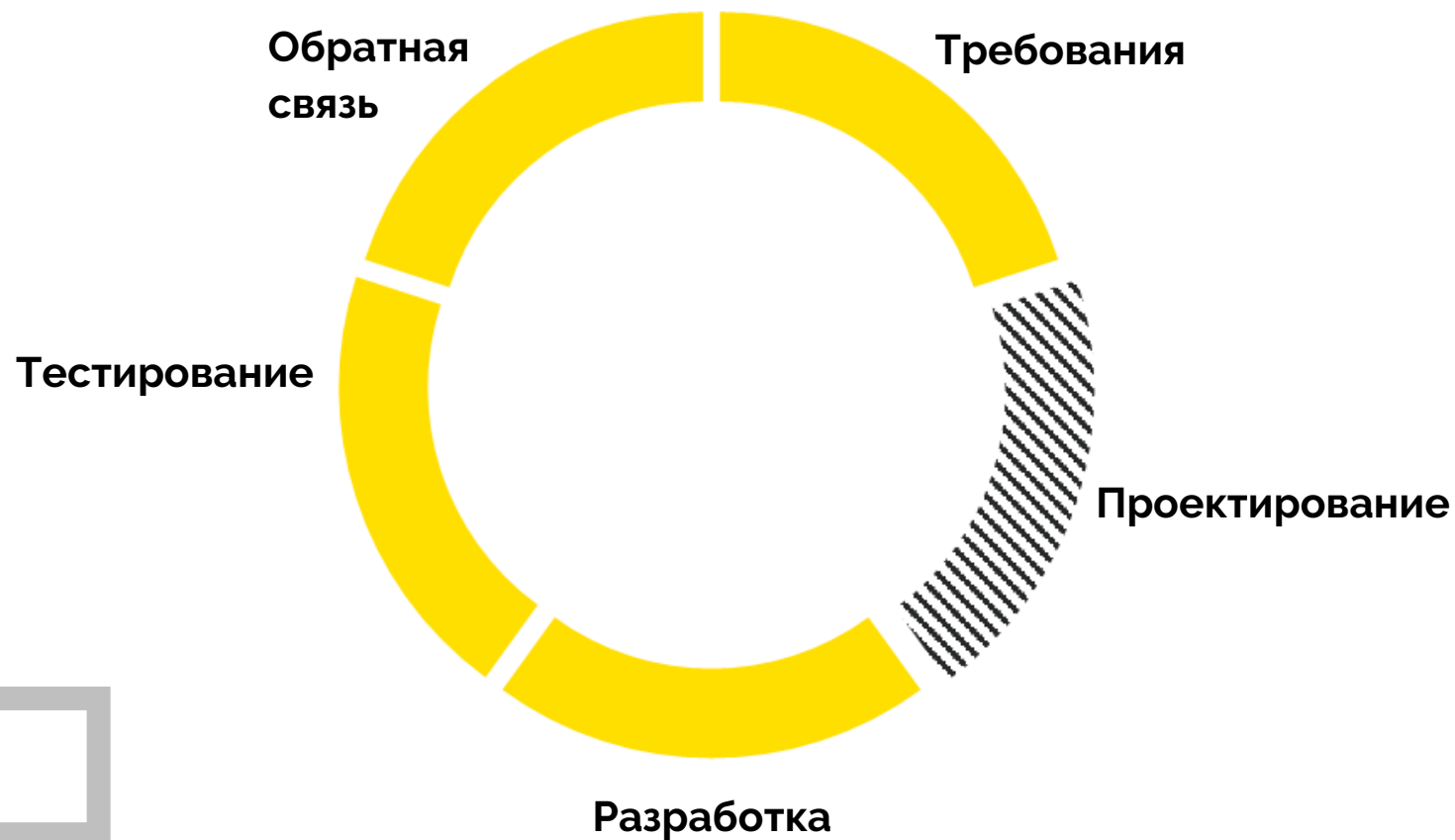
Системный анализ

Проектирование решения на базе
бизнес-требований

Продукт

Фича

Задача





Цели системного анализа

- **Спроектировать решение** оптимальным образом
- Выявить **влияние** на систему
- Создать **эффективную** техническую **документацию** для разработки
- **Декомпонировать** задачу для разработки
- Выявить влияние на внешнее окружения



Виды системного анализа

1. Функциональное проектирование
2. Техническое проектирование
3. Архитектурное проектирование



Функциональное проектирование

Функциональное проектирование — это определение того, **что** должна уметь делать система (или подсистема, сервис), прежде всего, для её пользователей и смежных систем (подсистем) и по каким **правилам**.

Из User Story в Use Case

User Story отвечает на вопрос ,
что хочет достичь пользователь

Use Case отвечает на вопрос, как
система **реагирует** на действия
пользователя

Я, как пользователь, хочу
иметь возможность оплатить
заказ

из БА в СА

Предисловие: Пользователь находится
на странице оплаты заказа

Пользователь: Нажимает "оплатить"
Система: проверяет сумму оплаты
Система: выполняет оплату
Система: изменяет статус Заказа

Use Case

Use Case - это детальное описание взаимодействия пользователя с системой.

Use Case описывают, как система должна реагировать на действия пользователя.

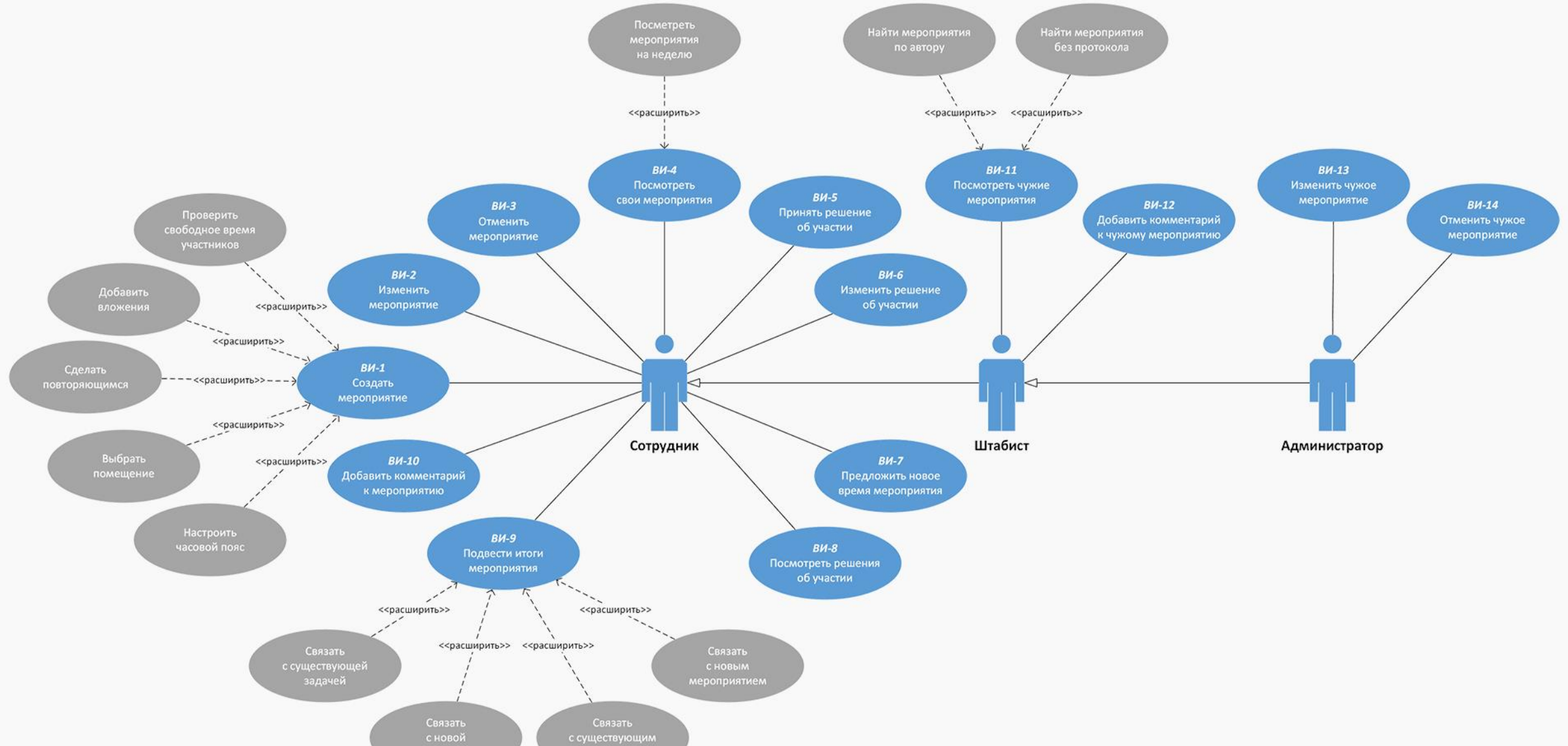
Артефакты:

текстовое описание UC,
диаграмма

Составные части:

1. Система
2. Действующее лицо
3. Цель Предусловие
4. Триггер
5. Основной поток действий
6. Альтернативные потоки действий
7. Результат

UC Diagram





Декомпозиция

Вертикальная

Задачи разбиваются таким образом, что в рамках одной задачи делаются все типы работ.

Горизонтальная

Задачи разбиваются по типу работ (отдельно BE, отдельно FE, отдельно тестирование)



INVEST

- **I**ndependent — независимая от других историй, то есть истории могут быть реализованы в любом порядке
- **N**egotiable — обсуждаемая, отражает суть, а не детали; не содержит конкретных шагов реализации
- **V**aluable — ценная для клиентов, бизнеса и стейкхолдеров
- **E**stimable — оцениваемая по сложности и трудозатратам
- **S**mall — компактная, может быть сделана командой за одну итерацию
- **T**estable — тестируемая (например, имеет критерии приемки)



Шаблоны декомпозиции

- Use Case
- Выделение MVP
- Операции (CRUD)
- По шагам workflow
- Роли
- Типы объектов
- Нефункциональные требования
- Варианты данных (моки/реальные данные, сохраняемые/несохраняемые)
- Обработка ошибок



Техническое проектирование

Техническое проектирование подразумевает проработку детальных проектных решений по разным аспектам системы, таких как:

- Хранение данных
- Пользовательские интерфейсы
- Интеграции (внутренние и внешние)

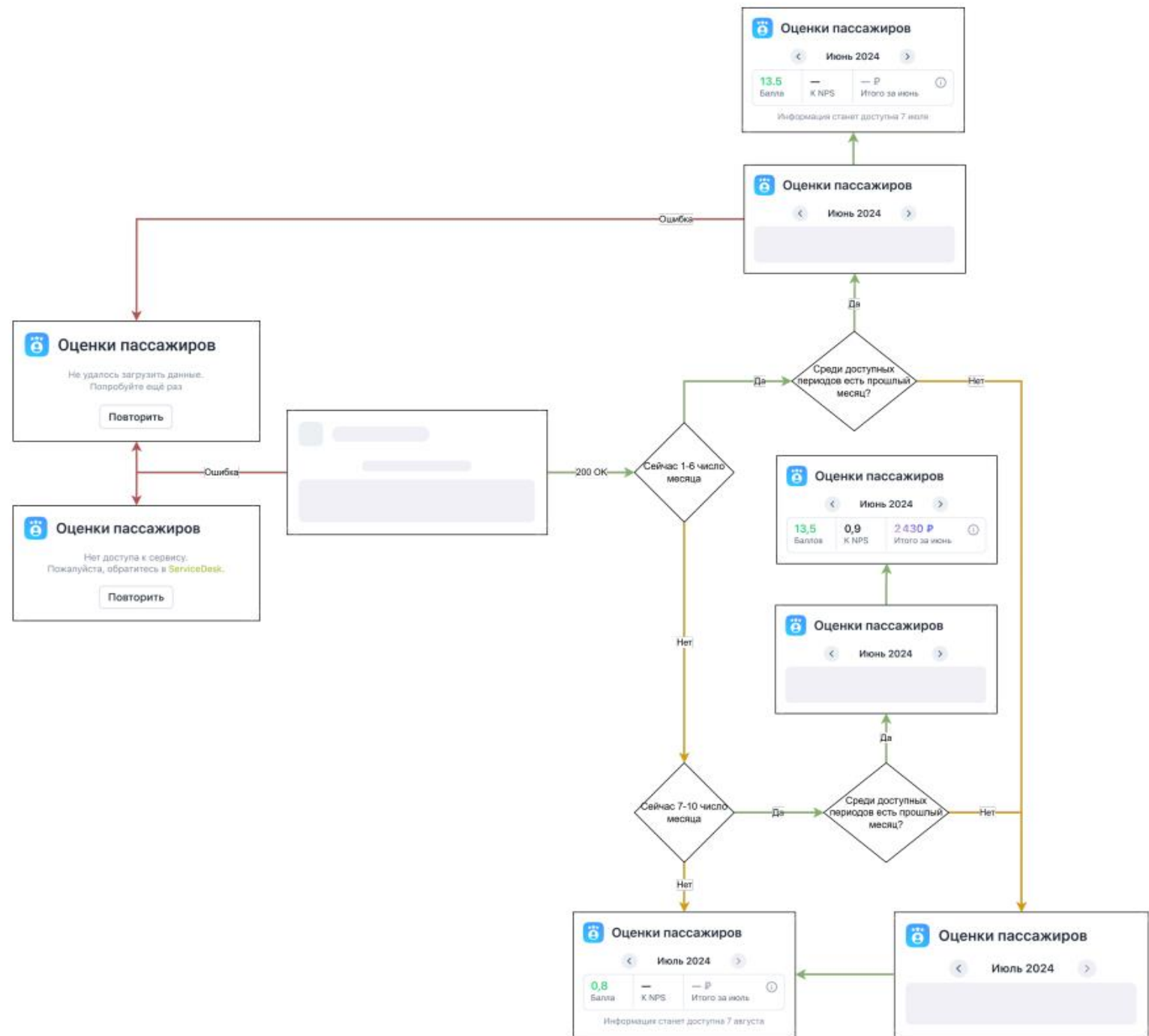


Пользовательские интерфейсы

Задачи СА

1. Описание логики взаимодействия пользователя с UI (вариативность поведения контролов)
2. Описание контрактов между FE и BE

Карта навигации





Хранение данных

Задачи СА

1. Выявить ключевые сущности и связи
2. Зафиксировать атрибуты
3. Выбрать способ хранения

ER-модель

Когда нужна

- Перед началом проекта (когда ещё не понятно, с какими данными предстоит работать)
- Когда нужно создать новую базу данных или добавить таблицу в уже существующую

1

Уровни

1. Концептуальный (сущности и связи)
2. Логический (атрибуты)
3. Физический (типы данных, ключи)

2

Представление

- ER-диаграмма
- Текстовое представление

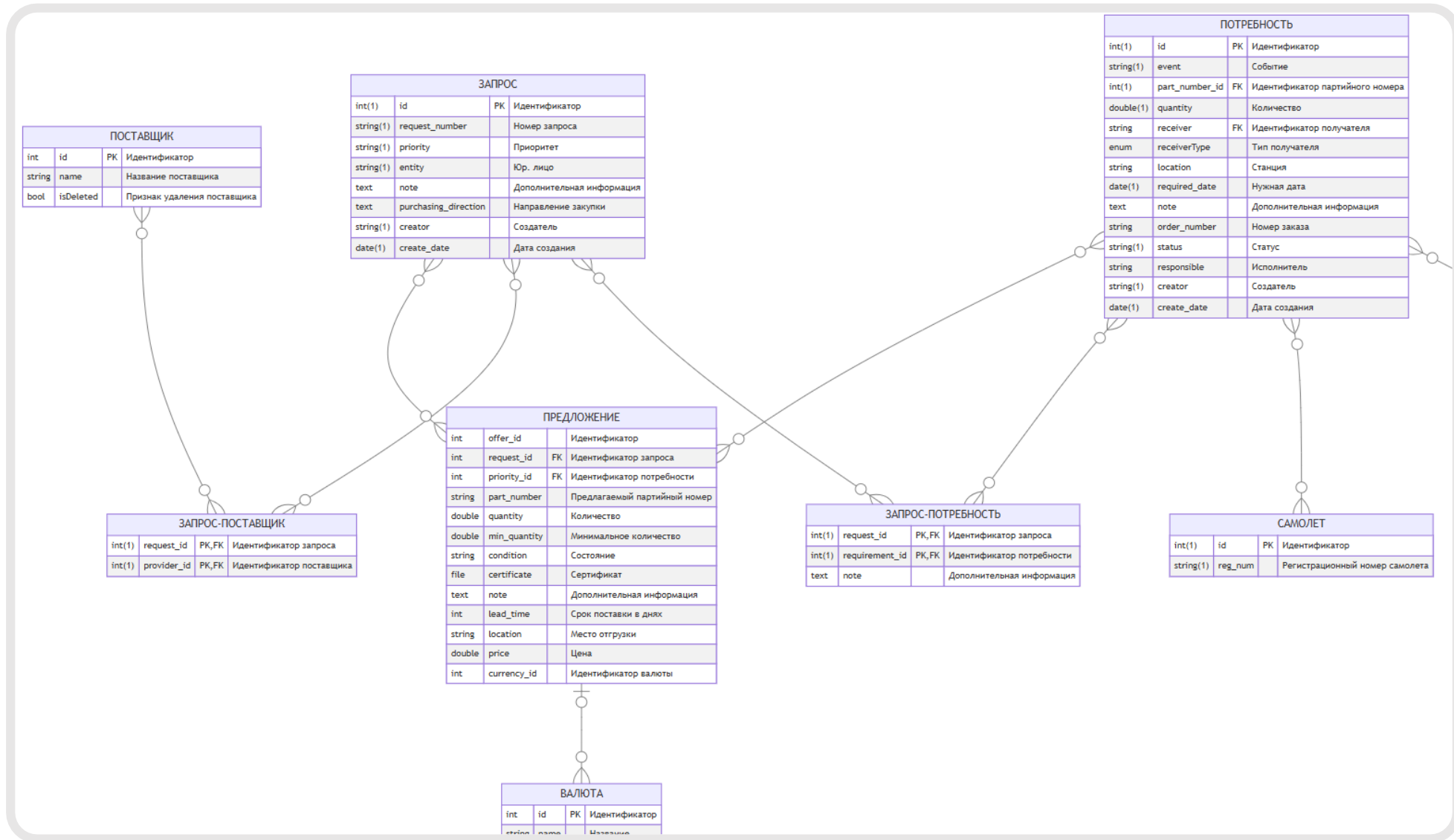
3

Инструменты

- Mermaid
- Draw io

4

ER Diagram





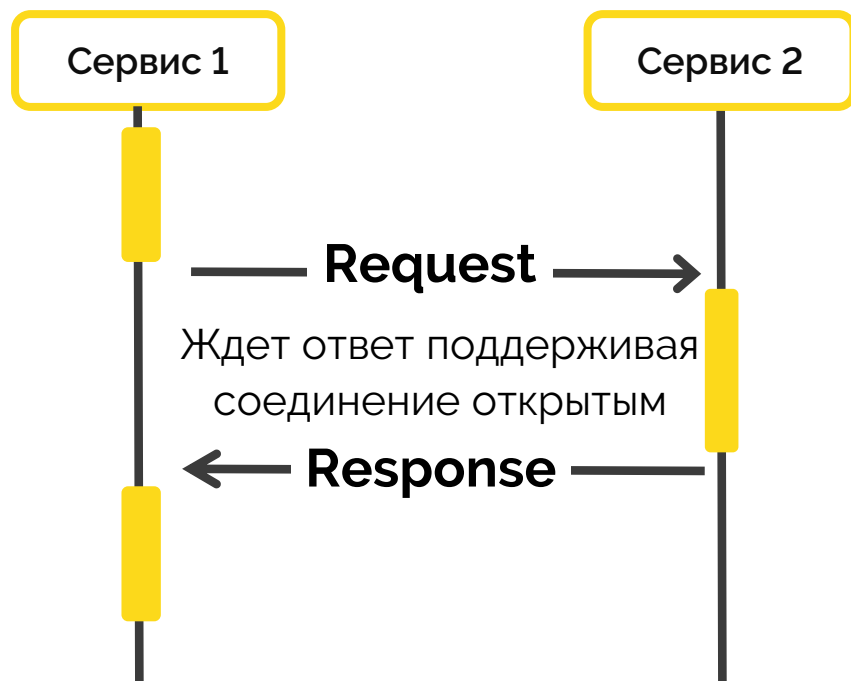
Интеграции (внутренние и внешние)

Задачи СА

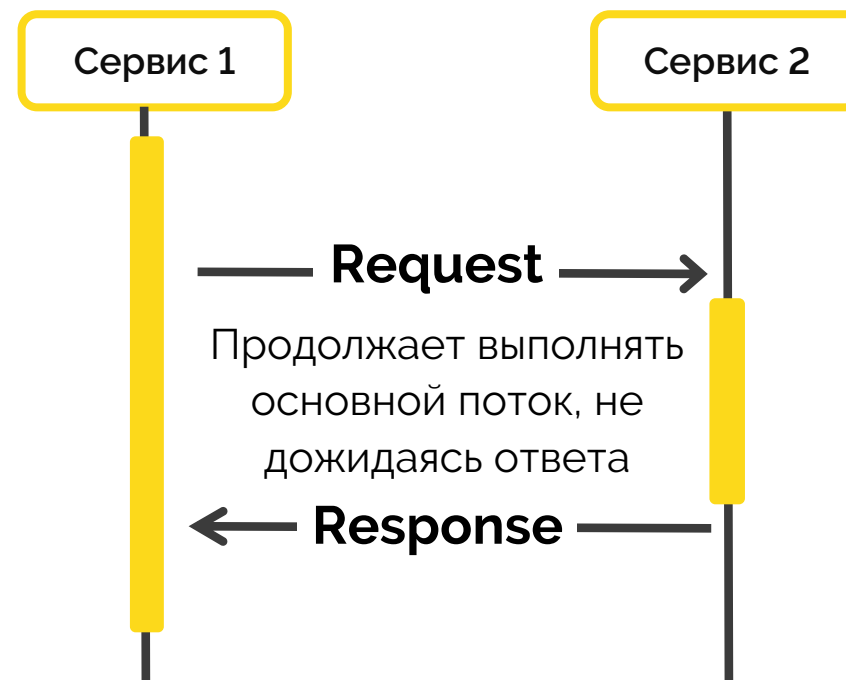
- Выявление точек интеграции
- Определение способов интеграции между системами и внутри компонентов системы
- Проектирование схемы взаимодействия
- Определение контрактов взаимодействия

Типы интеграций

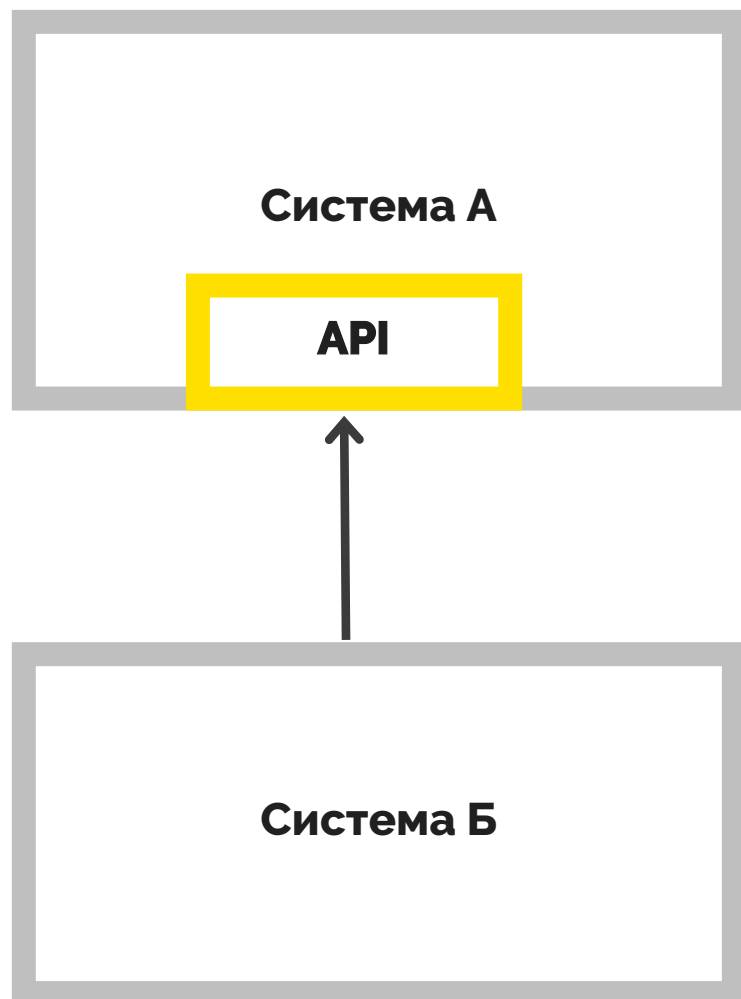
Синхронная - интеграция, при которой одна система **отправляет сообщение** другой и **ждет ответа**.



Асинхронная - интеграция, при которой одна система отправляет сообщение другой и не ждет ответа, а продолжает работу.



Интеграция через API



Система А - "Ко мне можно обратиться вот так, я обязуюсь сделать так"

Требования к API:

- Точка подключения, метод
- Способ авторизации
- Параметры запроса и ответа
- Модель запроса и модель ответа
- Примеры запросов

Описание интеграции в задаче

Description

Описание API

Запрос регистрации заказа - registerOrder

Параметры запроса

Тип запроса GET для отправки запроса следует использовать при условии, что параметры будут передаваться только в URL запроса (см. таблицу "Па

Параметры URL запроса:

Параметр	Тип	Обязательно	Описание
systemId	AN...50	да	Идентификатор Compass в ПМ UralAir, выданный при регистрации
pnr	AN6	да	Локатор PNR
lastname	A...30	нет	Фамилия пассажира
clientId	AN...20	нет	Номер участника программы лояльности «Крылья».
language	N2	нет	Должен использоваться FFN или Номер карты программы лояльности язык, выбранный клиентом. По умолчанию язык интерфейса Compass
hash	AN32	да	Значение, вычисляется как: HMAC [SHA256 (pnr.systemId.clientId.secretWord)]. Значение secretWord выдаётся при регистрации Внешней системы в ПМ UralAir. Примечание: Все точки в строке, от которой рассчитывается hash, обязательны. Если в запросе не передаётся clientId, то расчёт hash должен производиться от строки (pnr.systemId..secretWord).
returnUrl	AN...512	нет	Адрес перенаправления Клиента после оплаты в случае оплаты с применением технологии 3-D Secure (технология, описывающая механизм и последовательность действий, которые должен выполнить интернет-покупатель, чтобы подтвердить, что он действительно является законным владельцем карты, которой пытается расплатиться в интернет-магазине).
rowId	AN...20	нет	
callbackUrl	AN...512	нет	URL для получения callback о смене статуса заказа

Пример

Параметры ответа

Параметр	Тип	Обязательность	Описание
orderId	AN..50	да	Идентификатор заказа
date	UTC	да	Дата и время создания заказа (UTC).
errorCode	N..5	да	Код ошибки. Если вызов не привел к ошибке, используется значение 0. Список возможных значений представлен в таблице ниже
paymentFormUrl	AN...1024	да	Ссылка на платёжную страницу с параметрами: orderId, hash, language. Значение hash вычисляется как: HMAC SHA256 (orderId).
pnr	AN6	нет	Локатор PNR
remainMilisecs	N..20	нет	Оставшееся время для оплаты, в миллисекундах.
language	A2	нет	язык
isServiceOnlyOrder	A..5	нет	Возможные значения: true, false. Индикатор того, что заказ представляет собой допродажу Upsale.
saveBindingEnabled	A..5	нет	Возможные значения: true, false. Показывает, разрешено ли сохранять связь при оплате.
bindingItemEms		нет	Список связей Клиента.

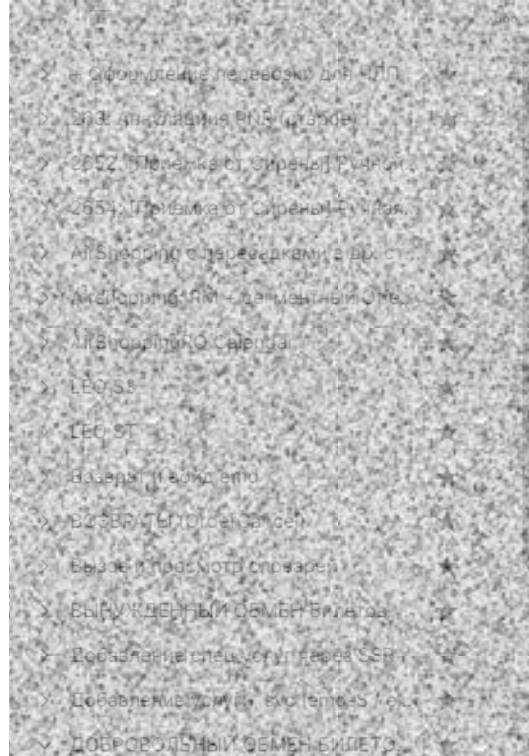
Возможные коды ошибок

Значение поля errorCode	Описание
OK	Корректно обработанный запрос
WRONG_REGISTRATION_CHECKSUM	Неверно рассчитан hash.
INVALID_SYSTEM_ID	Неверный systemId.
INTERNAL_ERROR	Ошибка при получении данных бронирования.
WRONG_REQUEST_DUE_ERRORS_IN_GETORDERINFO	Ошибка в данных бронирования.
ORDER_ALREADY_PAID	Заказ уже оплачен.

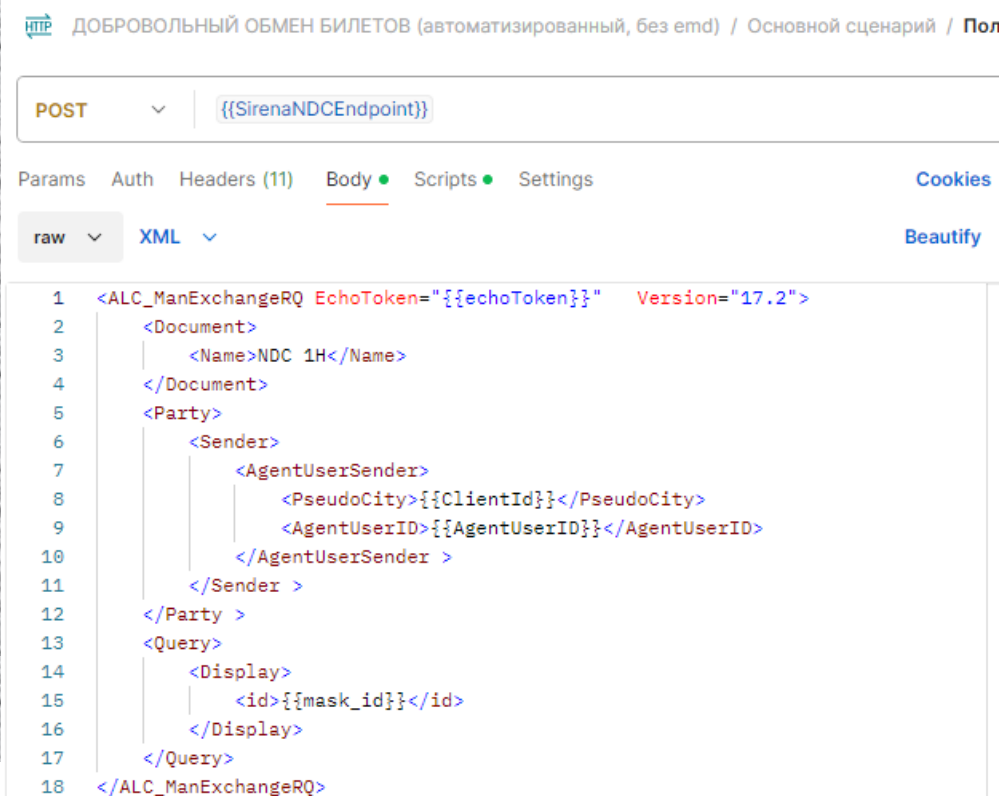
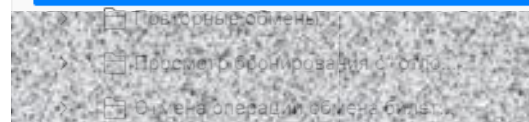
Пример ответа



Postman-коллекция



- Основной сценарий
 - POST Авторизация
 - POST Просмотр брони
 - > POST Получение расчета
 - POST Разделение брони
 - > POST Добавление сегментов
 - > POST Создание маски (ManExchang...
 - POST Получение маски (ManExchan...
 - > POST Оплата
 - POST Снять места со старых сегмен...
 - POST Удаление сегмента
 - POST Выход из системы



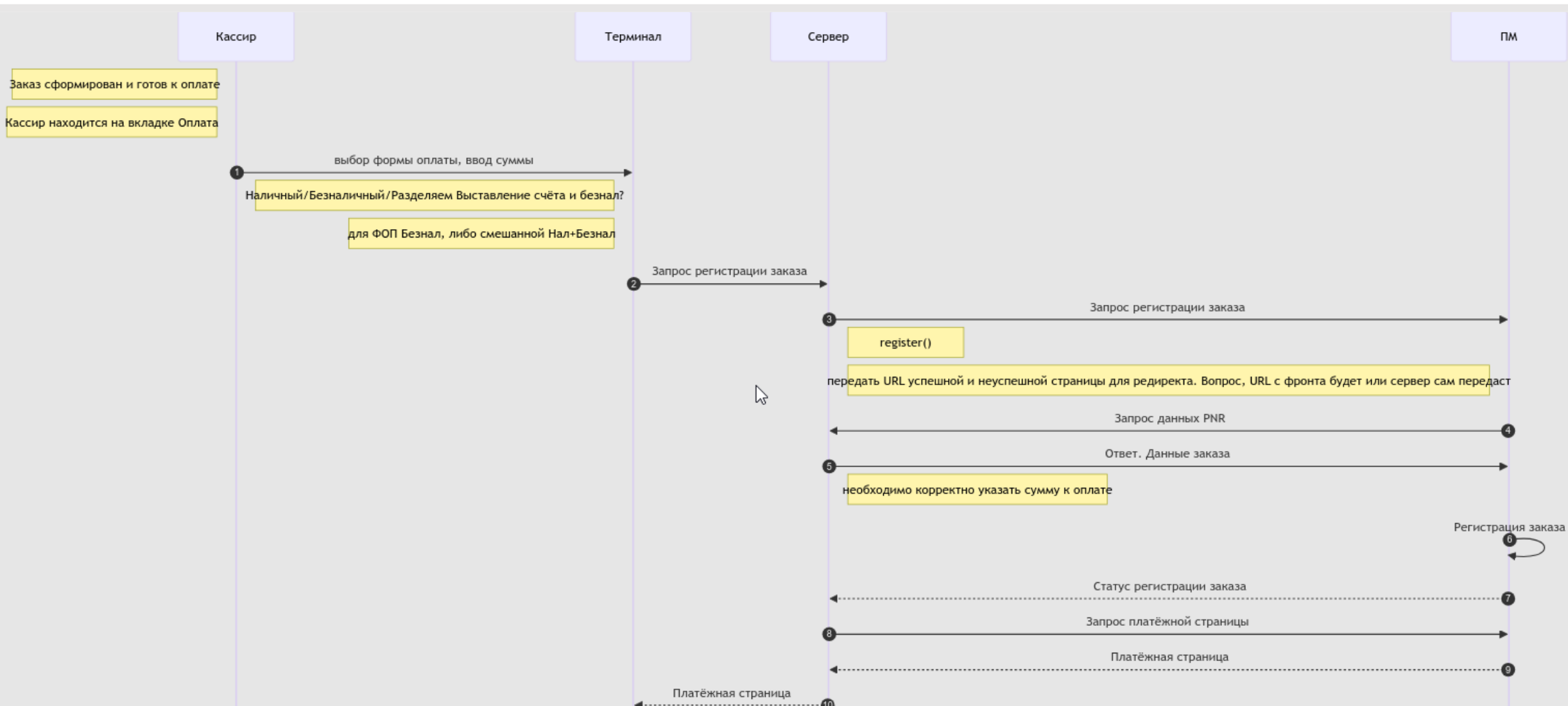
Интеграция через RabbitMQ и Apache Kafka

Требования к интеграции:

- Точка подключения (Адрес, креды)
- Название топиков, очередей
- Модель сообщения
- Примеры



Sequence-диаграмма



Архитектурное проектирование

Архитектурное проектирование - определяет **принципиальное внутреннее** устройство информационной системы — языки программирования, СУБД, архитектурные шаблоны, структурное и модульное разбиение, готовые компоненты, стили и технологии интеграции

Модель C4

Подход к описанию
архитектуры системы

Диаграмма контекста

Описывает систему в масштабе ее взаимодействия с пользователями и другими системами.

1

Диаграмма контейнеров

Описывает разбиение системы на взаимосвязанные контейнеры. Контейнер - это исполняемая и развертываемая подсистема.

2

Диаграмма компонентов

Описывает разбиение контейнеров на взаимосвязанные компоненты и отражают связи компонент с другими контейнерами или другими системами.

3

Диаграмма кода

Дополнительные сведения о дизайне архитектурных элементов, которые могут быть сопоставлены с программным кодом. Модель C4 на этом уровне опирается на существующие нотации (UML, диаграммы отношений сущностей - ERD).

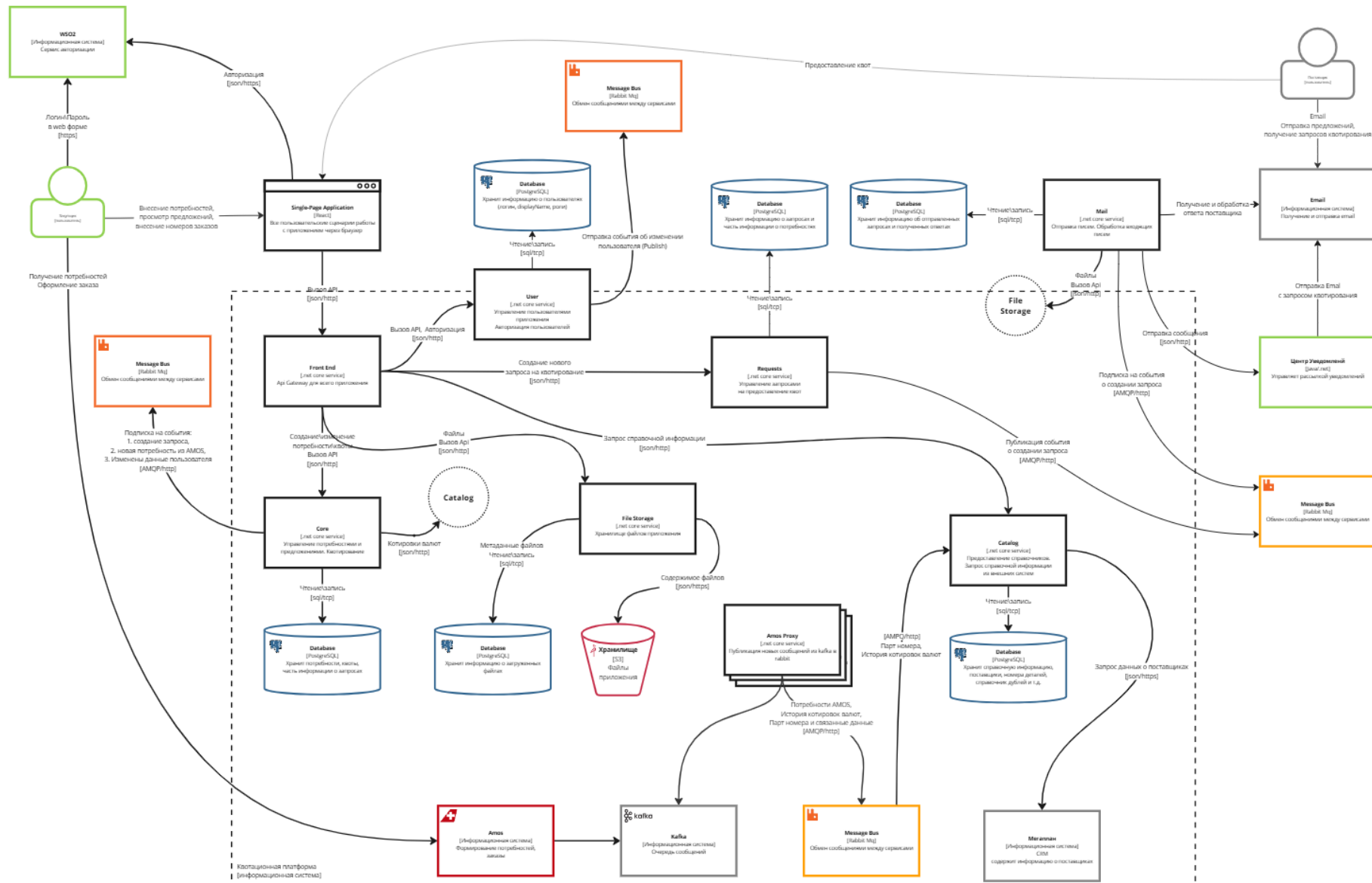
4

Инструменты

1. Draw.io
2. Miro
3. PlantUML

5

QUESTION



Нефункциональные требования

Требования, определяющие свойства, которые система должна **демонстрировать**, или **ограничения**, которые она должна соблюдать, не относящиеся к поведению системы.

Атрибуты качества

- Доступность, надежность
- Производительность
- Целостность *
- Совместимость
- Безопасность
- Удобство использования
- Масштабируемость *

Ограничения

- Технологии, языки программирования
- Обратная совместимость *
- Нормативные документы
- Интеграции с другими системами, форматы данных *
- Характеристики устройств, на которых будут работать пользователи



Выводы

Уровень абстракции/ Вид СА	Продукт	Фича	Задача
Архитектурное проектирование	C1, C2	C2, C3	-
Функциональное проектирование	Use Case - диаграмма	Use Case - диаграмма	Use Case
		Шаблоны декомпозиции	Экраны
Техническое проектирование	ER (концептуальный)	Базовый CJM	ER (логический)
		ER (логический) Sequence-диаграмма	Sequence-диаграмма Postman-коллекции Описание контрактов



Заключение

1. Аналитика нужна для уменьшения риска в производстве. Где нет риска – нет потребности в аналитике
2. Какие риски несет задача, такая и аналитика по ней несет максимальную ценность
3. В зависимости от коммерческой модели компании/ продукта/ процессов, позиция аналитика может объединять разные области анализа

**Спасибо за
ВНИМАНИЕ**