

ДИПЛОМНАЯ РАБОТА

КУРСА «АРХИТЕКТОР ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ»

Этап 2

разработчик Драчёв О.Е.

11. Список ADR. (Architecture decision records - Записи архитектурных решений)

011 Разработка проекта – «Компания для привлечения людей к спортивному образу жизни»

Статус – accepted (принят).

Контекст:

Компания имеет большой штат разработчиков, говорящих на различных языках, и охотно адаптирует новые технологии для экспериментальных приложений. 90% всех систем, используемых в компании, расположены у облачных провайдеров, при этом нет одного выбранного провайдера — используется то, что больше подходит под конкретную задачу.

Решения:

Провести разработку проекта «Компания для привлечения людей к спортивному образу жизни».

Причины принятия решения:

Достижение бизнес целей.

Последствия:

При своевременных корректирующих действиях на риски, компания получит прибыль и выполнит бизнес цели.

Комплаенс (проверка соответствия):

Еженедельное подведение итогов разработка и демонстрация результатов.

Заметки:

ИТ архитектор Драчёв О.Е. 11.03.2022.

012 Архитектурный стиль – Microservices.

Статус – accepted (принят).

Контекст:

Рассмотрены предложения архитектурных стилей монолит или микросервис.

(010 Анализ и описание архитектурных опций и обоснование выбора).

Решения:

Будем использовать Архитектурный стиль – Microservices.

Причины принятия решения:

Архитектурный стиль – Монолит не соответствует пункту 8 НФТ.

Последствия:

Увеличивается общая сложность разработки. Разделение проекта на сервисы упрощает работу команд разработчиков над отдельными модулями проекта. Вместе с тем повышаются требования к взаимодействию.

Комплаенс (проверка соответствия):

Необходима разработка сценариев тестирования взаимодействия модулей системы.

Заметки:

ИТ архитектор Драчёв О.Е. 14.03.2022.

013 Применение распределенной архитектуры.

Статус – proposed (предложен).

Контекст:

Компания предполагает большой охват населения разных частей мира.

Решения:

Для решения поставленной задачи, будем использовать облака региональных операторов.

Причины принятия решения:

Снижение трафика локальных клиентов.

Последствия:

Постепенное расширение охвата локальных территорий.

Комплаенс (проверка соответствия):

Необходимо контролировать трафик в разрезе локализации. Поднимать новые центры при увеличении трафика выше контрольных показателей.

Заметки:

ИТ архитектор Драчёв О.Е. 14.03.2022.

014 Хранение данных.

Статус – accepted (принят).

Контекст:

Наша компания работает в различных регионах мира.

В результате работы активных клиентов формируется много клиентской информации. При хранении данных в едином хранилище возникает проблема передачи данных из различных регионов мира.

Решения:

Организовать региональные хранилища клиентских данных. Данные домена заказ будем хранить в региональном хранилище, а также передавать в центральное хранилище.

Причины принятия решения:

Экономия трафика на пересылке клиентских данных. Бизнес данные о заказах хранятся в 2х хранилищах.

Последствия:

Синхронизация коммерческих данных в 2х хранилищах повысит надежность системы.

Комплаенс (проверка соответствия):

Провести тестовую выборку данных из различных хранилищ и сравнить их.

Заметки:

ИТ архитектор Драчёв О.Е. 14.03.2022.

12. Описание сценариев использования приложения.

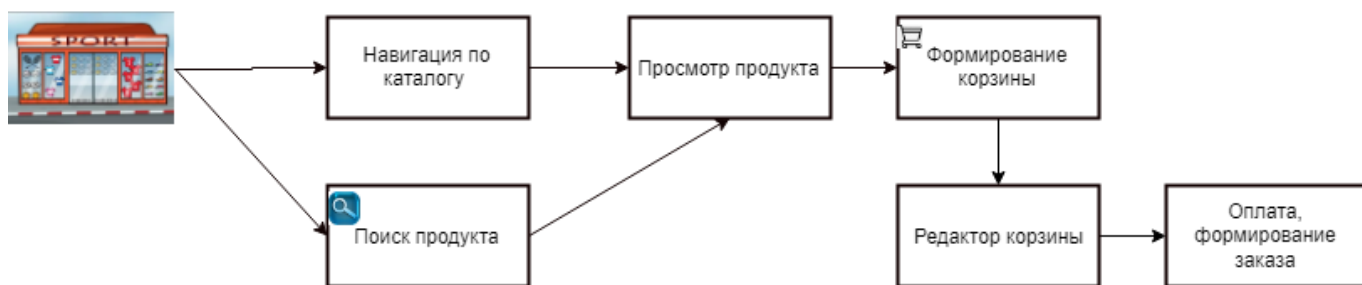
Стартовая страница приложения

Цель - увидеть состояние окружения и личной переписки.



Страница "Витрина товаров"

Цель - просмотр товаров, поиск товаров и формирование корзины



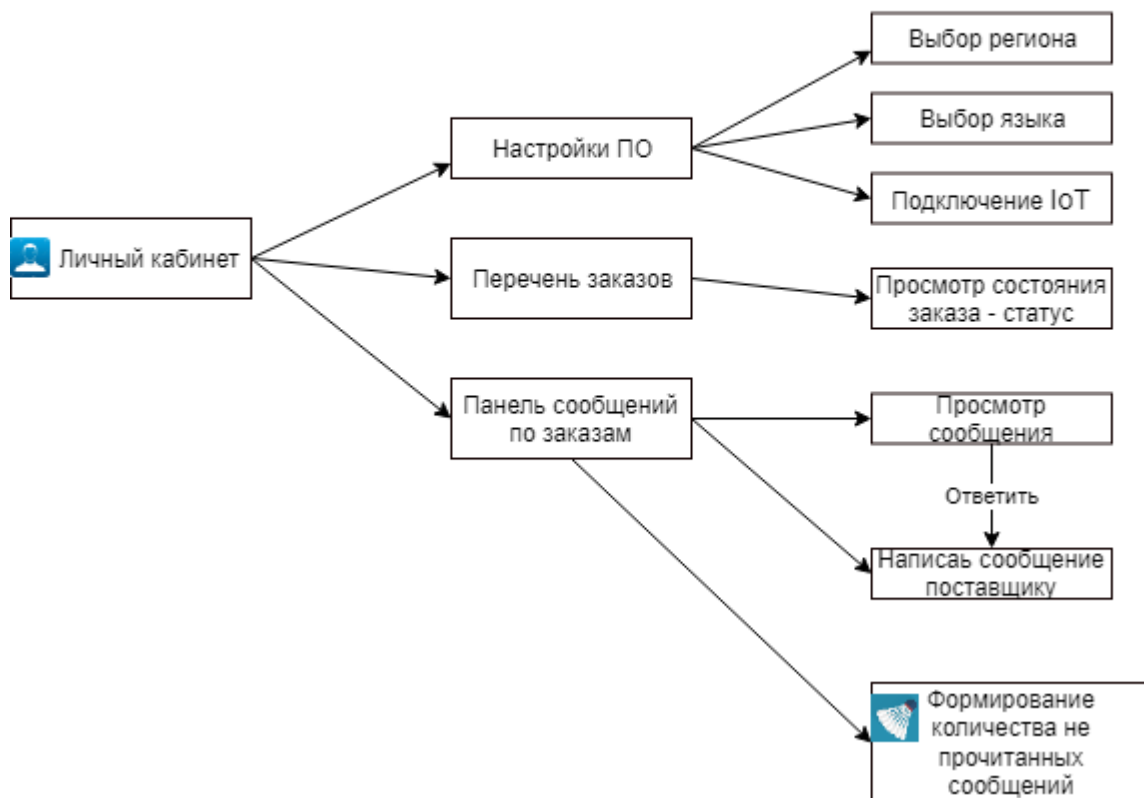
Страница "Спортивные показатели"

Цель - формирование планов тренировок, выбор группы по интересам, сбор материалов показателей организма, сохранение роликов и фотографий клиентов

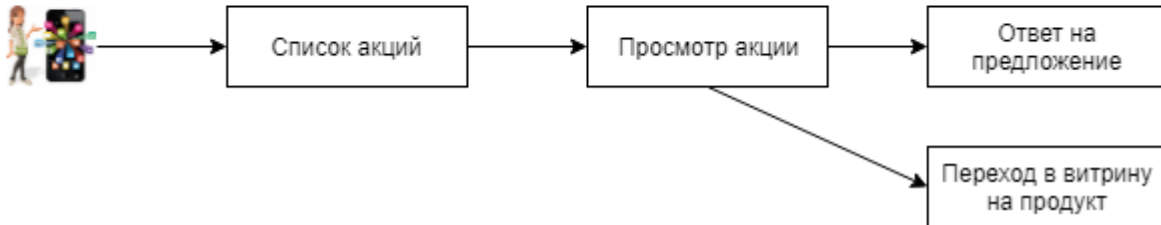


Страница "Личный кабинет"

Цель - обеспечить работу клиента: с заказами, личной перепиской, настройками ПО (локация, языковая панель, подключение IoT)



Страница "Региональных акций"
Цель - Демонстрация акций клиентам



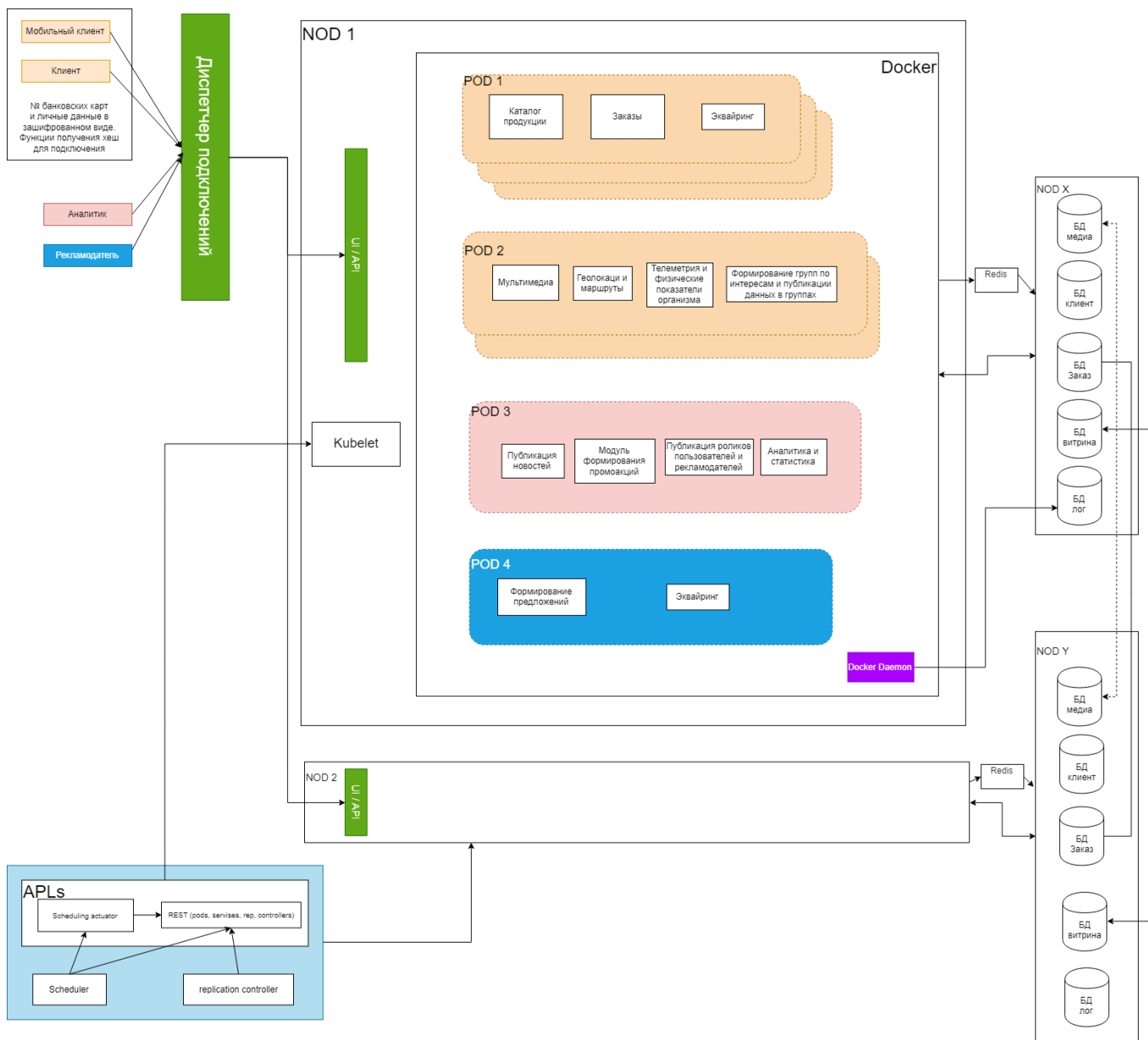
Страница "Индикатор сообщений"
Цель - Перейти на контекст сообщения



Страница "Корзина продукта"
Цель - Перейти в редактор корзины



13. **Базовая архитектура с учётом ограничений бизнес - требований, НФТ, выбранной архитектуры, адресация атрибутов качества.**

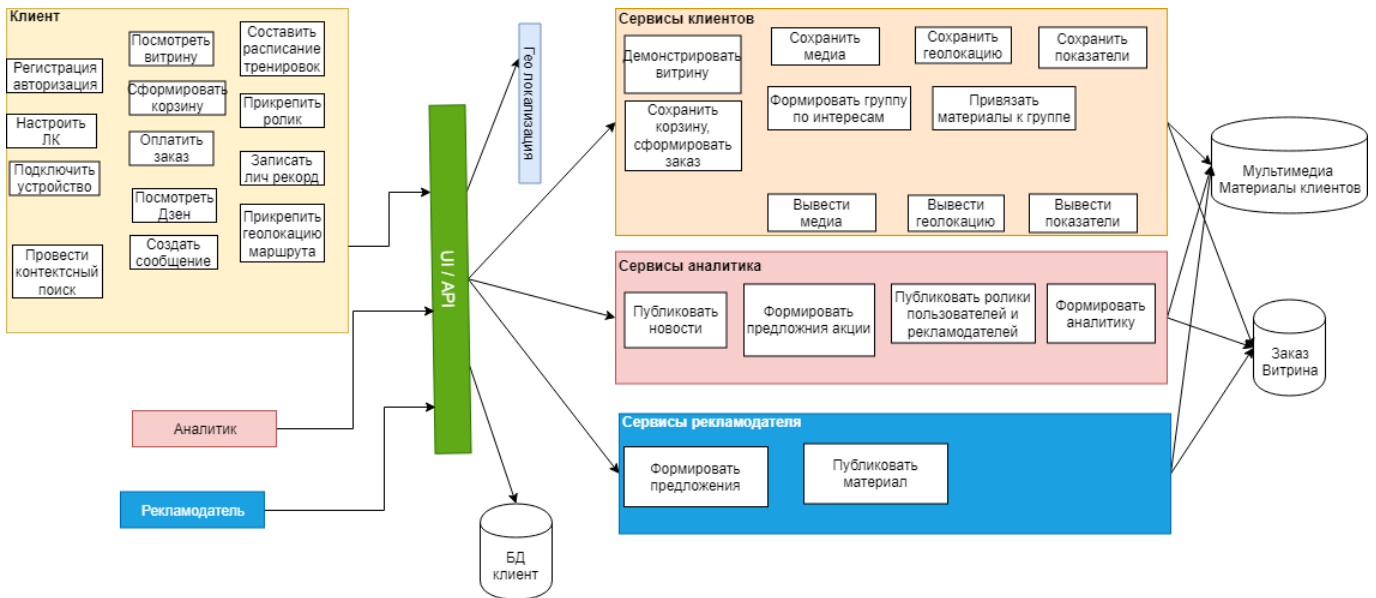


На рисунке представлена базовая схема проекта:

- Схема представляет распределенную систему под управлением Kubernetes.
- На схеме представлены основные блоки контейнеров.
- Система хранения данных и логов.
 - Данные разделены по типу.
 - БД Redis применяется для хранения ключей данных.
- Kubernetes обеспечивает оркестрацию подов при изменении нагрузки.
- Схема обеспечивает:
 - • Хранение личных данных – фт п.1.
 - Оптимальное использование ресурсов ВТ (вычислительной техники) – нфт п. 8.
-

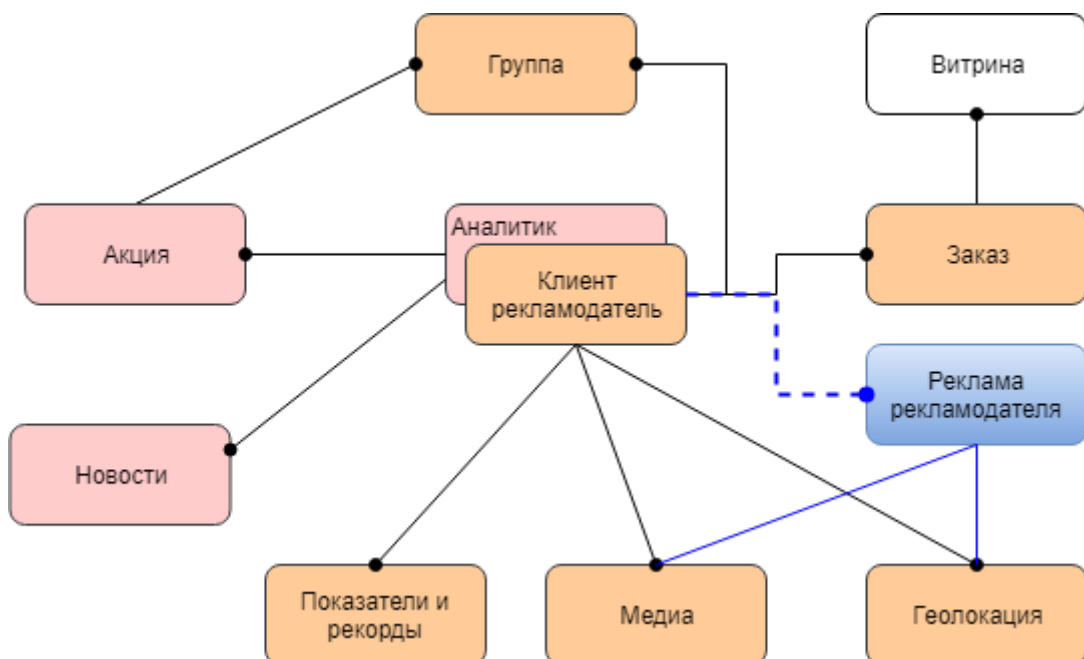
14. Основные представления:

а. Функциональное.



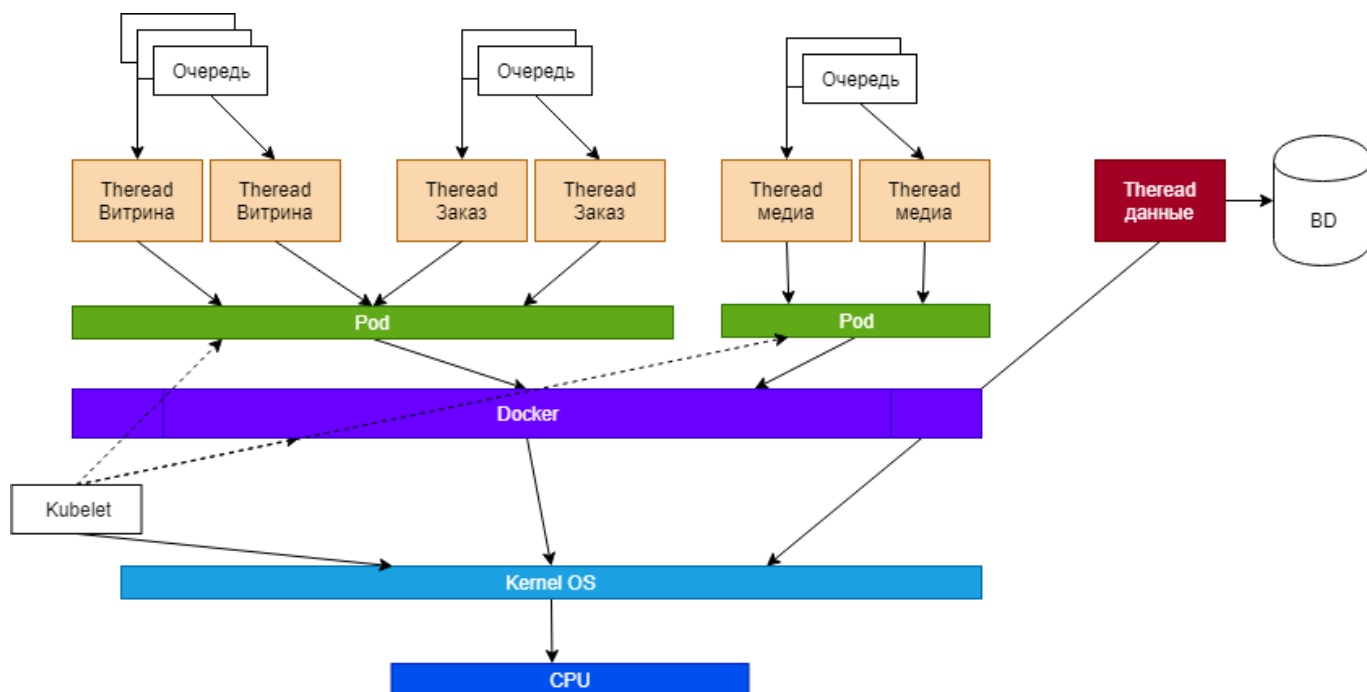
В схеме описаны основные функции пользователей системы. Не показаны функции бак-энда, в части разделения доступа к информации и т.д.

б. Информационное.



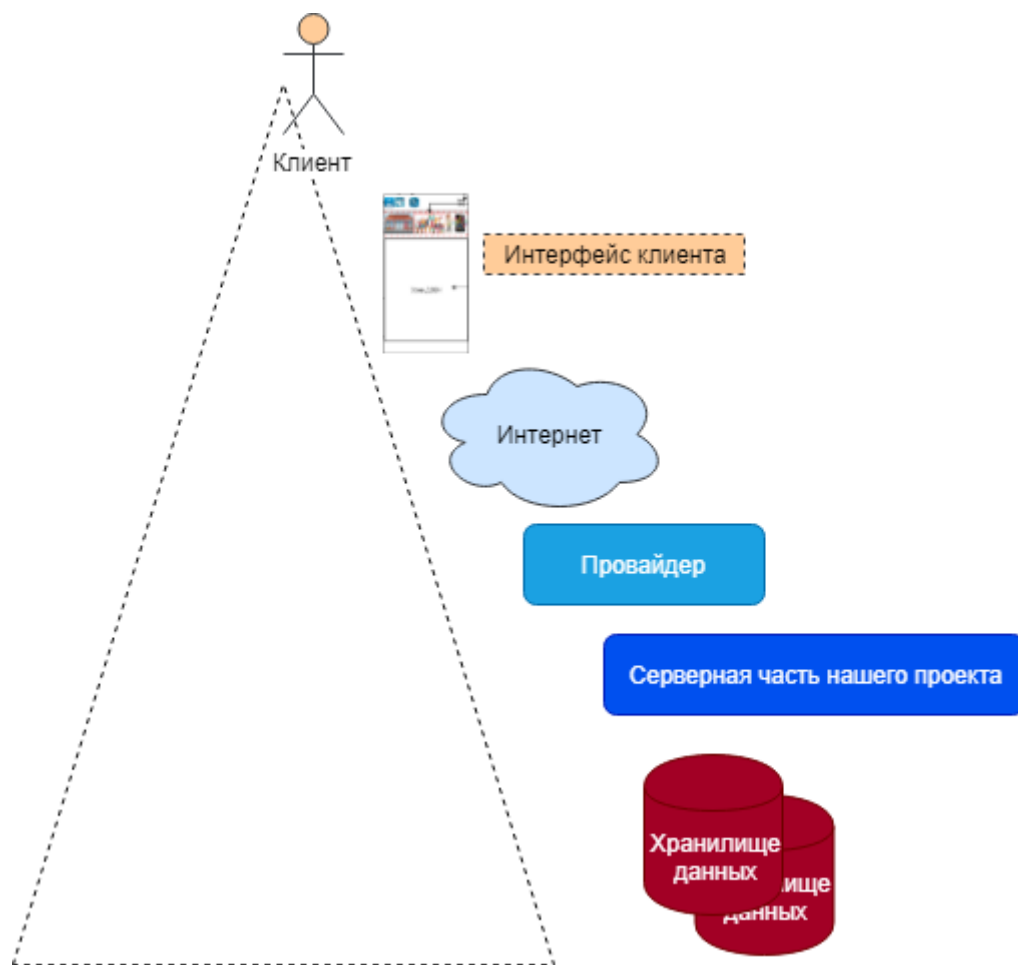
Информационная модель основана на потоках данных пользователь. В центре схемы находится клиент-рекламодатель. Рекламодатель использует данные медиа и геолокации совместно с клиентом. Аналитик использует статистические данные по заказам и активность клиентов в группах.

с. Многозадачность (concurrency).



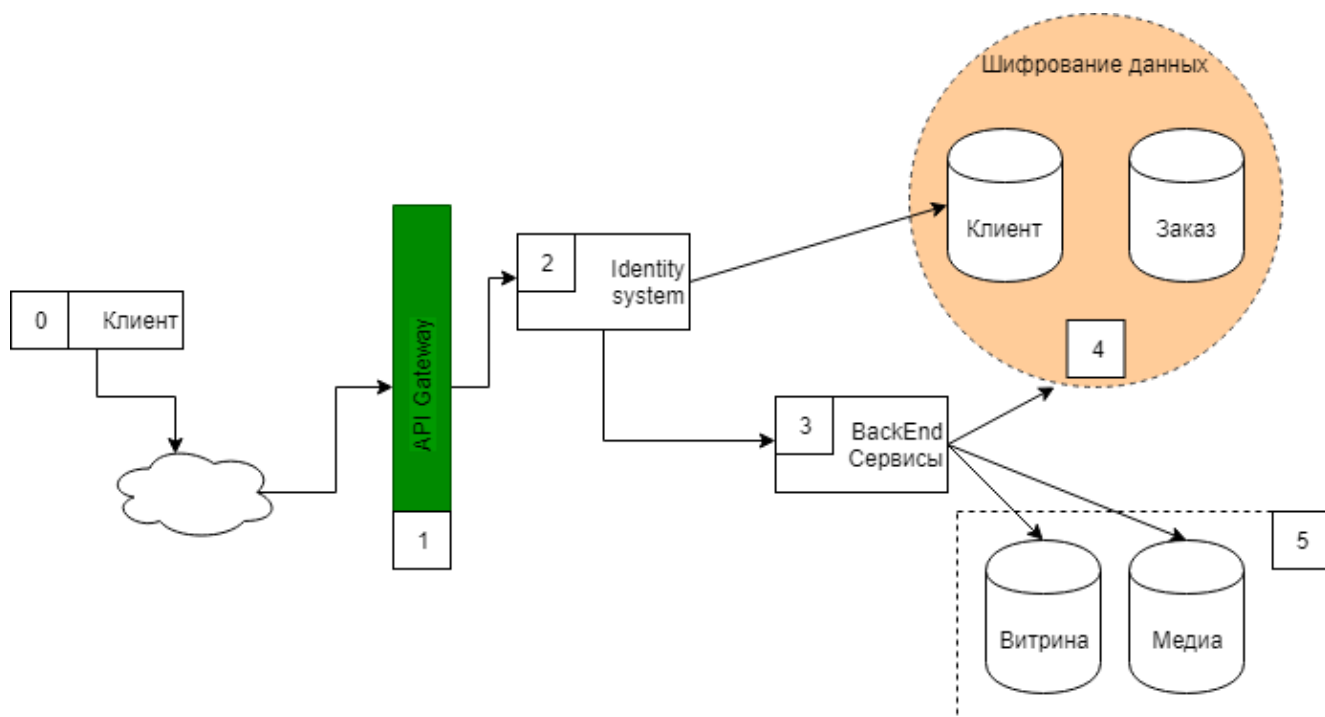
На схеме многозадачности представлена схема серверной части системы на основе Kubernetes. Клиентская часть ПО передает свои запросы в очереди серверной части. API Gateway на схеме не рассмотрен. Схеме представлена с уровня микросервисов пользовательского функционала.

d. Инфраструктурное.



На вершине схемы находится клиент, который работает с интерфейсом клиентского ПО. ПО клиента через интернет и потоки провайдера обращается к серверной части ПО. Серверное ПО работает с хранилищем данных. Вся схема опирается на электросети (на схеме не показаны).

е. Безопасность.



Элементы системы	№ вектора	Возможные векторы атак	Способы защиты от векторов атак
0,1	1	1. Инъекции — Injections	<ul style="list-style-type: none"> Использование более безопасного API, исключающего использование интерпретатора. Использование параметризованных запросов при кодировании. Отделение команд от данных во избежание атак.
0,1	2	Sensitive Data Exposure (незащищённость конфиденциальных данных)	<ul style="list-style-type: none"> Используя защищённый URL. Использование надёжных и уникальных паролей. Шифрование всей конфиденциальной информации, которую необходимо сохранить.
0,3	3	XML External Entities (XXE) Insecure Deserialization (внешние сущности XML, небезопасная десериализация)	<ul style="list-style-type: none"> Использование менее сложных форматов данных, таких как JSON. Обновление процессоров и библиотек XML. Использование инструментов SAST.
2	4	Broken Access Control (нарушение контроля доступа)	<ul style="list-style-type: none"> Удаление аккаунтов, которые больше не нужны или неактивны. Отключение ненужных служб для снижения нагрузки на серверы. Использование тестирования на проникновение.
Вся система	5	Небезопасная конфигурация (Security Misconfiguration)	<ul style="list-style-type: none"> Использование динамического тестирования безопасности приложений (DAST). Отключение использования паролей по умолчанию. Следите за облачными ресурсами, приложениями и серверами.
0,1	6	Межсайтовый скриптинг – XSS (Cross Site Scripting)	<ul style="list-style-type: none"> Использование соответствующих заголовков ответа. Фильтрация ввода и кодирование вывода. Использование политики безопасности контента.

			<ul style="list-style-type: none"> • Применение подхода с нулевым доверием к пользовательскому вводу.
2	7	Broken Authentication (нарушенная аутентификация)	<ul style="list-style-type: none"> • Реализация многофакторной аутентификации. • Защита учётных данных пользователя. • Отправка паролей через зашифрованные соединения.
0,2,3	8	Использование компонентов с известными уязвимостями	<ul style="list-style-type: none"> • Удаление всех ненужных зависимостей. • Использование виртуального исправления. • Использование компонентов только из официальных и проверенных источников.
3	9	Небезопасная десериализация	<ul style="list-style-type: none"> • Внедрение цифровых подписей. • Использование тестирования на проникновение. • Изоляция кода, который десериализует, и запуск его в средах с низким уровнем привилегий для предотвращения несанкционированных действий.
Вся система	10	Недостаточное ведение журнала и мониторинг	<ul style="list-style-type: none"> • Внедрение программного обеспечения для ведения журналов и аудита. • Создание эффективной системы мониторинга. • Думайте, как злоумышленник и используйте метод проверки на проникновение.

15. Анализ рисков созданной архитектуры, компромиссов.

№	Риск	Способы снижения рисков
1.	<p>Принятие неправильных архитектурных решений или неправильным выбором платформы:</p> <ul style="list-style-type: none"> компоненты и подсистемы ИС и их физическое расположение, синхронность/асинхронность общения между компонентами системы, выбор и определение характеристик каналов связи, выбор протоколов и программных интерфейсов, выбор типа ПО промежуточного слоя (middleware), выбор форматов документов, передаваемых в системе. 	<ul style="list-style-type: none"> Тщательная проработка проекта и привлечение опытных архитекторов. Тщательное фиксирование принятых решений.
2.	Не четкие требования заказчика, когда детали проявляются в процессе разработки.	<ul style="list-style-type: none"> Организация тесного взаимодействия с заказчиком. Включение представителя заказчика в команду разработки.
3.	Потеря коллектива разработчиков	<ul style="list-style-type: none"> Создание положительного климата в коллективе. Документирование всех принятых решений по проекту.
4.	Остановка отдельных функций системы (не своевременное увеличение размеров хранилища)	<ul style="list-style-type: none"> НФТ п.4. Оперативная отработка сообщений систем (Остановка записи данных). Анализ логов, разработка мероприятий по достижению устойчивости системы.
5.	Взлом сайта (DDoS атака)	<ul style="list-style-type: none"> Применение последних практик безопасности
6.	Потеря данных на клиентских устройствах	<ul style="list-style-type: none"> Частичное хранение личных данных, для восстановления связи.
7.	Потеря ключей в БД Redis	<ul style="list-style-type: none"> Восстановление ключей.
8.	Нет провайдеров с требуемым набором услуг для развития нашего проекта на территориях удаленных от центра.	<ul style="list-style-type: none"> Пользоваться ближайшим провайдером (мирится с более дорогим трафиком) Стимулировать развитие провайдеров (спонсорское участие).
9.	Потеря данных у локального провайдера	<ul style="list-style-type: none"> Заключение договоров с более высокими требованиями хранения данных. Создание подсистемы дублирования данных у ближайшего провайдера. Увеличение стоимости владения т.к. основная стоимость владения – это система хранения данных. Создание дополнительных дата центров.
10.	Не своевременная оплата услуг провайдера	<ul style="list-style-type: none"> Организация рассылки сообщений о завершении оплаченного периода.

16. Стоимость владения системой в первый, второй и пятый годы с учётом роста данных и базы пользователей

Первым приведем расчет стоимости на рекламу и будем считать его постоянным на каждый год.

Расходы на каждый вид рекламы													Год сумма
Каналы рекламы\месяцы	1	2	3	4	5	6	7	8	9	10	11	12	
SEO	25 000,00 Р	25 000,00 Р	25 000,00 Р	25 000,00 Р	25 000,00 Р	25 000,00 Р	25 000,00 Р	25 000,00 Р	25 000,00 Р	25 000,00 Р	25 000,00 Р	25 000,00 Р	300 000,00 Р
Яндекс Директ	10 000,00 Р	5 000,00 Р		10 000,00 Р	5 000,00 Р		10 000,00 Р	5 000,00 Р		10 000,00 Р	5 000,00 Р		60 000,00 Р
Google Adwards			20 000,00 Р			20 000,00 Р			20 000,00 Р			20 000,00 Р	80 000,00 Р
Рекламные банеры		30 000,00 Р			30 000,00 Р			30 000,00 Р			30 000,00 Р		120 000,00 Р
Рассылка	4 000,00 Р		4 000,00 Р		4 000,00 Р		4 000,00 Р		4 000,00 Р		4 000,00 Р		24 000,00 Р
Прямые расходы	12 000,00 Р	12 000,00 Р	12 000,00 Р	12 000,00 Р	12 000,00 Р	12 000,00 Р	12 000,00 Р	12 000,00 Р	12 000,00 Р	12 000,00 Р	12 000,00 Р	12 000,00 Р	144 000,00 Р
Итого:													728 000,00 Р

Для быстрой доставки товаров, товаров для проведения акций и учета повышенного спроса на товары в период проведения спортивных мероприятий – необходим склад. Полного расчета по заданным условиям провести не возможно (как минимум нет групп товаров). Расчет стоимости взят с аналога, интернет магазина и составляет:

1й год - 1 768 686,00 Р.

2й год - 2 299 291,80 Р.

5й год - 3 360 503,40 Р

Для расчета затрат на поддержание сайта я решил использовать «Cloud» калькулятор яндекса (<https://cloud.yandex.ru/prices>).

За аналог расчета были использованы статистические показатели сети ВКонтакте:

- Использована статистика открытых источников.
- Зарегистрированные пользователи – 380 миллионов.
- Нас интересуют активные пользователи.
 - 1-й год 8 миллионов.
 - 2-й год 21 миллион.
 - 5-й год 50 миллионов.
- За основу расчета взяты следующие метрики (в месяц):
 - Объем исходящий трафик более 10Gb (дальнейшее увеличение не учитывается).
 - Выделенный Ip адрес.
 - SSD диски для загрузки ПО
 - Ram = 6Gb
 - Get операции = 800000000
 - Post операции = 300000000
 - Размер хранилища = 500 Tb
 - Размер хранилища логов = 2 Tb
 - Redis Размер хранилища = 710 Гб

Итоги расчета первого года эксплуатации:

- Compute Cloud – 4072.95 руб.
- Object Storage – 1180183.29 руб.
- Kubernetes – 25891.67 руб.
- PostgreSQL – 34752.56 руб.
- Redis - 14339.37 руб.
- Elasticsearch - 40348.72 руб.

- Тех. Поддержка - 53,55 руб.
- Итого: **1 299 657,41 руб./мес.**

Рассчитать стоимость

Compute Cloud

Object Storage

Kubernetes*

PostgreSQL

ClickHouse

MongoDB

MySQL*

Redis™

Elasticsearch

Greenplum*

SpeechKit

Translate

Техническая поддержка

Object Storage

Тип хранилища ?

Стандартное Холодное

Размер хранилища

500000 ГБ

GET-операции

- 800 000 0С +

POST-операции

- 300 000 0С +

Исходящий трафик ?

20 ГБ

+ Добавить хранилище

Итого:

Compute Cloud × 1	4 072,95 Р ×
Intel Ice Lake, 50% vCPU	1 843,20 Р
Intel Ice Lake, RAM	806,40 Р
Публичный IP-адрес	172,80 Р
Быстрое сетевое хранилище (SSD)	1 250,55 Р
Object Storage	1 180 183,29 Р ×
Занятое место в стандартном хранилище	1 004 991,99 Р
Стандартное хранилище — операции GET	31 196,10 Р
Стандартное хранилище — операции POST	143 995,20 Р
Kubernetes*	25 891,67 Р ×
Managed Kubernetes, Zonal Master - small	6 336,00 Р
Intel Ice Lake, 50% vCPU	3 686,40 Р
Intel Ice Lake, RAM	1 612,80 Р
Публичный IP-адрес	345,60 Р
Быстрое сетевое хранилище (SSD)	13 910,87 Р
PostgreSQL	34 752,56 Р ×
PostgreSQL, Intel Cascade Lake, 100% vCPU	2 606,40 Р
PostgreSQL, Intel Cascade Lake, RAM	5 644,80 Р
Быстрое сетевое хранилище — PostgreSQL	26 501,36 Р
Redis™	14 339,37 Р ×
Redis, Intel Cascade Lake, 100% vCPU	2 419,20 Р
Redis, Intel Cascade Lake, RAM	2 592,00 Р
Быстрое сетевое хранилище — Redis	9 328,17 Р
Elasticsearch	40 348,72 Р ×
Elasticsearch, Intel Cascade Lake, 100% vCPU	2 419,20 Р
Elasticsearch, Gold, Intel Cascade Lake, RAM	14 169,60 Р
Быстрое сетевое хранилище — Elasticsearch	23 587,12 Р
Публичный IP-адрес - Elasticsearch	172,80 Р
Техническая поддержка	0,00 Р ×
Техническая поддержка – тарифный план Базовый	0,00 Р
Исходящий трафик из Object Storage в интернет	15,30 Р
Исходящий трафик в интернет	53,55 Р
1 299 657,41 Р	в месяц

Второй год владения. Произойдет увеличение функций и возможностей пользователя, потребуются новые ресурсы.

- За основу расчета взяты следующие метрики (в месяц):
 - Объем исходящий трафик более 10гб (дальнейшее увеличение не учитывается).
 - Выделенный Ip адрес.
 - SSD диски для загрузки ПО
 - Ram = 64Гб
 - Get операции = 24000000000
 - Post операции = 10000000000
 - Размер хранилища = 4000 Tb (*92Мб\клиент - 65 миллиона клиентов 2й год владения*
200 Мб\клиент - 72 миллиона клиентов 5й год владения)
 - Размер хранилища логов = 2 Tb
 - Redis Размер хранилища = 710 Гб

Кроме увеличения вычислительных мощностей, нужно увеличить количество нодов. Устанавливаем 2 дополнительных нодов. Снижение стоимости можно ожидать за счет подключения холодного хранилища.

Итого: $4\,867\,369,05 * 3 = \underline{\underline{14\,602\,107,15 \text{ руб.}}}$

Рассчитать стоимость

⚙️ Compute Cloud

📁 Object Storage

🔗 Kubernetes*

🗄️ PostgreSQL

🗄️ ClickHouse

🗄️ MongoDB

🗄️ MySQL*

🗄️ Redis™

🔍 Elasticsearch

🌿 Greenplum*

🗣️ SpeechKit

🗣️ Translate

🛠️ Техническая поддержка

Compute Cloud

Операционная система ?

Ubuntu 20.04 LTS

Платформа ?

Intel Ice Lake

Гарантированная доля vCPU ?

20%50%100%

Количество vCPU ?

16

296

Объём RAM ?

64 ГБ

16 ГБ256 ГБ

Прерываемая ВМ ?

☐

Публичный IP-адрес

☒

Исходящий трафик ?

20

ГБ

Диск – 1 (Загрузочный)

☐ HDD☒ SSD

Итого:

Compute Cloud × 1	26 421,75 Р	×
Intel Ice Lake. 100% vCPU	12 096,00 Р	
Intel Ice Lake. RAM	12 902,40 Р	
Публичный IP-адрес	172,80 Р	
Быстрое сетевое хранилище (SSD)	1 250,55 Р	
Object Storage	4 690 044,71 Р	×
Занятое место в стандартном хранилище	4 116 453,41 Р	
Стандартное хранилище — операции GET	93 596,10 Р	
Стандартное хранилище — операции POST	479 995,20 Р	
Kubernetes*	36 922,07 Р	×
Managed Kubernetes. Zonal Master - small	6 336,00 Р	
Intel Ice Lake. 100% vCPU	9 072,00 Р	
Intel Ice Lake. RAM	7 257,60 Р	
Публичный IP-адрес	345,60 Р	
Быстрое сетевое хранилище (SSD)	13 910,87 Р	
PostgreSQL	41 764,95 Р	×
PostgreSQL. Intel Cascade Lake. 100% vCPU	2 606,40 Р	
PostgreSQL. Intel Cascade Lake. RAM	5 644,80 Р	
Быстрое сетевое хранилище — PostgreSQL	33 513,75 Р	
Redis™	25 801,17 Р	×
Redis. Intel Cascade Lake. 100% vCPU	2 419,20 Р	
Redis. Intel Cascade Lake. RAM	2 592,00 Р	
Быстрое сетевое хранилище — Redis	20 789,97 Р	
Elasticsearch	46 307,30 Р	×
Elasticsearch. Intel Cascade Lake. 100% vCPU	2 419,20 Р	
Elasticsearch. Gold. Intel Cascade Lake. RAM	14 169,60 Р	
Быстрое сетевое хранилище — Elasticsearch	29 545,70 Р	
Публичный IP-адрес - Elasticsearch	172,80 Р	
Техническая поддержка	0,00 Р	×
Техническая поддержка – тарифный план Базовый	0,00 Р	
Исходящий трафик из Object Storage в интернет	15,30 Р	
Исходящий трафик в интернет	91,80 Р	
4 867 369,05 Р	в месяц	▼

Пятый год владения – предполагаем расширение за счет периферийных узлов, в количестве 6 штук. Основные затраты - это системы хранения данных.

Стоимость 1 го месяца системы на пятый год эксплуатации.

Итого: $4\,867\,369,05 * 7 = \underline{\underline{34\,071\,583,35 \text{ руб/месяц}}}$.

Итоговая таблица затрат по годам владения

Год владения	Реклама	Склад	Затраты на сайт\месяц	Затраты на сайт\год	Сумма затрат в год
1й год	728 000,00	1 768 686,00	1 299 657,41	15 595 888,92	18 092 574,92
2й год	728 000,00	2 299 291,80	14 602 107,15	175 225 285,80	178 252 577,60
5й год	728 000,00	3 360 503,40	34 071 583,35	408 859 000,20	412 947 503,60

