***Карпачев Ярослав НБИбд-01-23 ТЗ Мессенджер***

Оглавление

[1. Общие положения 2](#_Toc209798703)

[1.1. Наименование и условное обозначение работы 2](#_Toc209798704)

[1.2. Наименование предприятий Заказчика и Исполнителя 2](#_Toc209798705)

[1.3. Сроки выполнения 2](#_Toc209798706)

[1.4. Особые условия 2](#_Toc209798707)

[2. Назначение разработки 3](#_Toc209798708)

[3. Требования к программе или программному изделию 4](#_Toc209798709)

[3.1. Требования к функциональным характеристикам 4](#_Toc209798710)

[3.1.1. Общие требования к функциям ПО (состав ПО) 4](#_Toc209798711)

[3.1.2. Требования к функциям ПО 10](#_Toc209798712)

[3.2. Требования к надежности 13](#_Toc209798713)

[3.2.1. Термины, охват и методика измерений 13](#_Toc209798714)

[3.2.2. Доступность (SLO по зонам) 14](#_Toc209798715)

[3.2.3. Учёт и атрибуция простоев 14](#_Toc209798716)

[3.2.4. Отказоустойчивость и отсутствие единой точки отказа (3 сервера) 15](#_Toc209798717)

[3.2.5. Восстановление (RPO/RTO, резервирование) 15](#_Toc209798718)

[3.2.6. Производительность (SLO по задержкам и пропускной способности) 15](#_Toc209798719)

[3.2.7. Деградация без отказа 16](#_Toc209798720)

[3.2.8. Политики очередей и повторов 16](#_Toc209798721)

[3.2.9. Предупреждения и операционные цели 16](#_Toc209798722)

[3.2.10. Идемпотентность и повторная доставка 16](#_Toc209798723)

[3.2.11. Коммуникации при инцидентах 16](#_Toc209798724)

[3.3. Требования к составу и параметрам технических средств 17](#_Toc209798725)

[3.3.1. Общие принципы и допущения 17](#_Toc209798726)

[3.3.2. Роли серверов (номинальный режим) 17](#_Toc209798727)

[3.3.3. Ресурсы, ретеншн и лимиты 19](#_Toc209798728)

[3.3.4. Сеть и безопасность 19](#_Toc209798729)

[3.3.5. Энергоснабжение и размещение 19](#_Toc209798730)

[3.4. Требования к информационной и программной совместимости 19](#_Toc209798731)

[Стек разработки 19](#_Toc209798732)

[Практическая детализация 20](#_Toc209798733)

[4. Требования к программной документации 22](#_Toc209798734)

[4.1. Актуализация 23](#_Toc209798735)

# 1. Общие положения

## 1.1. Наименование и условное обозначение работы

1.1.1 Наименование работы: «Разработка системы мгновенного обмена сообщениями “Стриж” (веб-клиент + мобильное приложение + серверная часть)».

## 1.2. Наименование предприятий Заказчика и Исполнителя

**Заказчик:** ООО «НордНет Тех», 197000, г. Санкт-Петербург, ул. Петровская, д. 10.

**Исполнитель:** ООО «Аврора Лабс», 125000, г. Москва, ул. Лесная, д. 5.

## 1.3. Сроки выполнения

* Начало работ: **01.10.2025**
* Окончание работ (релиз 1.0): **31.03.2026**

## 1.4. Особые условия

Настоящее Техническое задание (ТЗ) может уточняться и дополняться в процессе выполнения работ. Согласование и утверждение дополнений к настоящему Техническому заданию проводятся в порядке, установленном для ТЗ.

# 2. Назначение разработки

Цель — создание облачной системы мгновенного обмена сообщениями «Стриж»), для личной и групповой коммуникации, рассылок (каналов), обмена медиа и автоматизации через ботов.

Система предназначена для массового и корпоративного использования через веб-браузер (с возможностью дальнейшего расширения на мобильные и десктоп-клиенты).

Область применения ПО является: корпоративные и пользовательские коммуникации, поддержка сообществ/каналов, служебные рассылки, интеграции с ИТ-системами через Bot/API.

# 3. Требования к программе или программному изделию

## 3.1. Требования к функциональным характеристикам

### 3.1.1. Общие требования к функциям ПО (состав ПО)

#### Модули аутентификации и профиля:

**1. М. Аутентификации и профилей**

Назначение. Безопасный вход, учётная запись и данные профиля.

Функционал.

* Регистрация по email/телефону с OTP; подтверждение владения контактами.
* 2FA (TOTP/Email/SMS), passkeys (WebAuthn).
* Управление сессиями и устройствами, выход со всех устройств.
* Восстановление доступа (reset-токены/OTP), смена пароля.
* Профиль: аватар, био, username; проверка уникальности username.
* Анти-bruteforce, капча, уведомления о входе с нового устройства.

Технологии. TLS 1.3; Argon2id для хэширования паролей; JWT/opaque tokens; WebAuthn; CSP/SameSite/CSRF.

Зависимости. API-шлюз; Уведомления (алерты безопасности); Аналитика (события).

**2. М. Контактов и адресной книги**

Назначение. Социальный граф пользователя.

Функционал.

* Импорт/экспорт: vCard 4.0, CSV; нормализация телефонов, дедупликация.
* Матчинг по номеру/email, статусы контактов, блок/мьют.
* Пригласительные ссылки: генерация, TTL, лимиты использований, принятие инвайта.

Технологии. vCard 4.0; short-linking; anti-abuse.

Зависимости. Аутентификация; Уведомления (инвайты).

#### Модули коммуникаций:

**3. М. Чатов 1:1 и «секретных чатов» (E2EE)**

Назначение. Обмен сообщениями, включая end-to-end шифрование.

Функционал.

* Сообщения (текст/медиа/файлы/ссылки), статусы (sent/delivered/read).
* Реакции, ответы/цитирование, правка/удаление с историей версий.
* Секретные чаты: E2EE, таймеры самоуничтожения, запрет пересылки.
* Индикаторы «печатает/в сети», локальные черновики.
* Политики ретенции, жалобы на сообщение.

Технологии. Realtime через WebSocket; E2EE (double-ratchet/MLS — по выбору); HTTP/3 для REST.

Лимиты. Размер вложений по умолчанию до 2 ГБ (конфигурируемо).

Зависимости. Медиа; Хранилище медиа; Очереди; Поиск; Уведомления.

**4. М. Групп и супергрупп**

Назначение. Многосторонняя коммуникация с ролями и масштабом.

Функционал.

* Роли/права (владелец/админы/участники), запросы на вступление.
* До 200 000 участников (порог конфигурируемый), slow-mode.
* Закрепления, опросы/квизы, модерация и анти-рейд.

Технологии. Фан-аут через Очереди; WS для событий; HTTP/3 для REST.

Зависимости. Контакты; Уведомления; Аналитика; Админ-панель.

**5. М. Каналов/рассылок**

Назначение. Односторонние публикации (паблик/приват) с метриками охвата.

Функционал.

* Управление подписками, превью постов, черновики, планирование публикаций.
* Базовая статистика подписчиков и просмотров.

Технологии. HTTP/3; WebSocket (обновления); интеграция с CDN для медиа.

Зависимости. Медиа; Аналитика; Уведомления; Админ-панель.

#### Модули контента и хранения:

**6. М. Медиа и файлов**

Назначение. Приём, обработка и выдача вложений.

Функционал.

* Загрузка (chunked, возобновление), определение MIME, валидация типа/размера.
* Генерация миниатюр/превью (изображения/видео/документы), опциональный транскод.
* Вирус-скан, контент-хеши/дедупликация, presigned-URLs.

Технологии. HTTP/3; Image/Video pipelines; антивирус (ClamAV/коммерч.);

Зависимости. Хранилище медиа (S3-совместимое), CDN; Очереди (фоновая обработка).

**7. М. Хранилища медиа (S3-совместимое, CDN)**

Назначение. Надёжное хранение больших объектов.

Функционал.

* Бакеты/ключи, life-cycle, классы хранения, версионирование.
* Шифрование at-rest, presigned-URLs, multi-region (опционально).
* Интеграция с CDN (origin-pull), инвалидация кэша.

Технологии. S3 API; KMS; CDN (HTTP/3, TLS 1.3).

Зависимости. Медиа и файлы.

#### Модули поиска и индексации:

**8. М. Поиска и индексации**

Назначение. Полнотекстовый поиск с учётом ACL.

Функционал.

* Индексация диалогов/каналов/метаданных; подсветка сниппетов.
* Фильтры: тип контента, участник, дата, источник (чат/канал).
* Поиск по вложениям (OCR/метаданные — опционально).
* Инкрементальная переиндексация, фоновые задания.

Технологии. OpenSearch/Elasticsearch; analyzers; shard-replicas.

Зависимости. Очереди (feed индексации); Аутентификация/ACL.

#### Модули Доставки и уведомлений:

**9. М. Уведомлений**

Назначение. Доставка событий пользователю.

Функционал.

* Web Push + e-mail fallback; управление токенами устройств.
* Пер-чатные настройки (все/только упоминания/мьют), «тихие часы» (DND).
* Бэтчинг, дедупликация, ретраи; локализация шаблонов.

Технологии. Web Push (VAPID); SMTP; HTTP/3; retry/backoff; rate-limit.

Зависимости. Очереди; Аналитика; API-шлюз.

**10. М. Очереди/асинхронная доставки (Message Broker)**

Назначение. Транспорт событий и фоновых задач.

Функционал.

* Pub/Sub для сообщений, уведомлений, индексации, превью.
* Отложенные задачи, DLQ, ретраи, идемпотентность.

Технологии. Kafka / NATS JetStream / RabbitMQ (по окружению).

Зависимости. Все сервисы, особенно Медиа/Поиск/Уведомления/Аналитика.

#### Модули API и клиента:

**11. М. API шлюза**

Назначение. Единая точка входа и терминатор протоколов.

Функционал.

* Протоколы: HTTP/1.1 + HTTP/2 + HTTP/3 (QUIC); WebSocket; опционально WebTransport/HTTP/3 для low-latency.
* Аутентификация (JWT/сессионные токены, passkeys-proof), CORS, TLS offload.
* OpenAPI 3.x: контракт, валидация схем, генерация SDK.
* Rate-limit/quotas, защита от DDoS/ботов; маршрутизация к внутренним сервисам.

Технологии. TLS 1.3; mTLS (внутренний периметр); WAF; OpenAPI 3.x.

Зависимости. Все доменные сервисы; Аналитика (логирование запросов).

**12. М. Веб-клиента (SPA)**

Назначение. UI-слой для всех пользовательских сценариев.

Функционал.

* Онбординг, login/2FA/passkeys, управление сессиями.
* UI чатов/групп/каналов, загрузка/предпросмотр медиа, реакции/редакции.
* Управление контактами, уведомлениями, приватностью, настройками.
* Подписка на Web Push, оффлайн-кэш, realtime через WebSocket.

Технологии. ECMAScript (современный стандарт), TypeScript (актуальный релиз), React (актуальный релиз); сборка на Node LTS; HTTP/3/WS.

#### Модули экосистемы, модерации и статистики:

**13. М. Bot API**

Назначение. Расширения и автоматизация от внешних разработчиков.

Функционал.

* webhook’и по HTTPS, inline-режим, кастомные клавиатуры/кнопки.
* Скоупы/разрешения, изоляция: per-bot rate-limit, квоты, токены с ротацией.
* Безопасная выдача медиа (scoped presigned-URLs), инструменты тестирования.

Технологии. OpenAPI 3.x; подпись webhook; mTLS (опционально).

Зависимости. API-шлюз; Чаты/Каналы/Медиа (ограниченный доступ); Аналитика.

**14. М. Модерации**

Назначение. Контент-безопасность и управление ресурсами.

Функционал.

* Антиспам/антирейд (правила, эвристики, ML — опционально).
* Жалобы пользователей, очереди модерации, блокировки аккаунтов/чатов/каналов.
* Аудит действий админов, экспорт отчётов.

Технологии. RBAC/ABAC; журналирование; immutable-лог.

Зависимости. Все доменные сервисы; Аналитика и Логи.

**15. М. Аналитики и мониторинга**

Назначение. Наблюдаемость и продуктовые метрики.

Функционал.

* Метрики (RPS/latency/errors), логи, алёртинг; дешборды SRE.
* Distributed tracing; профилирование; аудит-лог.
* Продуктовые KPI: DAU/MAU, ретеншн, виральность, CTR уведомлений.
* Экспорт сырых событий в DWH/stream для BI.

Технологии. OpenTelemetry (traces/metrics/logs); Prometheus-совместимые метрики; ClickHouse/BQ/Snowflake (по окружению).

Зависимости. Очереди (stream событий); все сервисы.

### 3.1.2. Требования к функциям ПО

#### 3.1.2.1. Система должна обеспечивать:

* **Регистрацию, вход и управление сессиями**, 2FA (TOTP/SMS/e-mail) и **passkeys (WebAuthn)**; безопасное хранение паролей (Argon2id).
* **Синхронизацию** истории и настроек между устройствами (облако); для **E2EE** синхронизируются только метаданные/ключи по политике — содержимое серверу недоступно.
* **Отправку/получение сообщений** с подтверждениями статусов (sent/delivered/read), упоминания **@**, **#хэштеги**, черновики, закрепления, а также **редактирование/удаление с историей**.
* **Секретные чаты с E2EE**: Double Ratchet для 1:1 (forward secrecy/post-compromise security), **MLS** для групп (где применимо); таймеры самоуничтожения; запрет пересылки/скриншотов по возможности (на уровне клиента — не гарантируется устройством).
* **Группы/каналы**: роли/права на публикации и медиа, антиспам и **slow-mode**, приглашения по ссылке, публичные юзернеймы/ссылки; опционально — опросы и базовая статистика каналов.
* **Поиск** по пользователям, чатам и контенту с учётом **ACL**; сохранённые фильтры/запросы; **контент E2EE не индексируется сервером**.
* **Медиа**: chunked/resumable загрузки и скачивание, предпросмотры/миниатюры, серверное транскодирование видео по политике; опциональная антивирусная проверка (например, ClamAV); выдача **presigned URLs**.
* **Bot API**: события сообщений, inline-режим, кнопки/клавиатуры; **webhook**’и по HTTPS с подписью, **rate-limit** и квоты; выдача/ротация токенов; верификация ботов/приложений.
* **Локализация**: RU/EN на релизе; расширяемость i18n/l10n.
* **Доступность**: соответствие **WCAG 2.2 AA** для веб-клиента.
* **Экспорт данных пользователя** (портируемость, GDPR-совместимость): архивы JSON/HTML/ZIP; поддержка запросов на удаление/корректировку.
* **Логирование и аудит**: структурированные логи, журнал админ-действий и инцидентов; метрики и трассировка (OpenTelemetry).

#### 3.1.2.2. Входными данными разрабатываемого ПО должны быть:

**A) Идентификация и доступ**

* Учётные данные: email (RFC-формат), phone (E.164), OTP, password (не в открытом виде).

*Зачем:* подтверждение владения и аутентификация.

* Токены сессий: access\_token, refresh\_token, атрибуты устройства/сессии.

*Зачем:* поддержание и отзыв сеансов, SSO, ограничение по устройствам.

**B) Социальный граф (контакты и инвайты)**

* Контакты: contact\_id (UUID/идентификатор пользователя), name, phone/email, теги/заметки.

*Зачем:* адресная книга, быстрый старт диалогов.

* Пригласительные ссылки: invite\_code, expires\_at, max\_uses, scope (в чат/канал/группу).

*Зачем:* контролируемое добавление пользователей.

**C) Сообщения и контекст**

* Сообщение: message\_id (клиентский/временный), text, attachments[] (ссылки на медиа), reactions[], reply\_to, edit\_version.

*Зачем:* содержание и взаимосвязи сообщений.

* Метаданные диалога: chat\_id/peer\_id, participant\_ids, created\_at/sent\_at, флаги (e2ee, self\_destruct\_ttl), статусы доставки (клиент → сервер).

*Зачем:* маршрутизация, права, таймеры.

**D) Медиа и файлы**

* Потоки данных: bytes (chunked/возобновляемая загрузка).
* Метаданные: mime\_type, size, checksum/content\_hash, duration/exif (если есть), параметры предпросмотра.

**E) Настройки и политики**

* Пользовательские настройки: уведомления (мут/тихие часы), приватность, язык/часовой пояс.
* Настройки групп/каналов: роли/права, slow-mode, фильтры контента.

*Зачем:* персонализация и политика доступа.

**F) Сигналы модерации/жалобы**

* Репорты: report\_id, reporter\_id, объект жалобы (message\_id/chat\_id), reason, приложенные доказательства.
* Автосигналы: антиспам-скор, правила срабатывания.

*Зачем:* triage, принятие решений в админ-панели.

#### 3.1.2.3. Выходными данными разрабатываемого ПО должны быть:

**A) Месседжинг и ленты**

* **События сообщений:** message\_id (серверный), status (sent/delivered/read), read\_receipts (по пользователям), typing, presence.

*Для кого:* клиенты (реал-тайм синхронизация и корректные бейджи).

* **Ленты каналов:** пагинированные подборки post\_id, содержимое/превью, stats (просмотры/реакции — если разрешено).

*Для кого:* подписчики, редакторы каналов.

**B) Уведомления**

* **Web Push / e-mail:** тип события, заголовок/текст, deep-link, mute/dnd соблюдены, результаты доставки (успех/повтор/отказ).

*Для кого:* конечные пользователи; в отчёты — агрегированные статусы.

**C) Отчёты модерации и аудит**

* **Модерация:** решение (action: блок/снятие/игнор), объекты воздействия, причина, исполнитель, обратная связь репортёру.
* **Аудит-лог:** кто/что/когда (админ-действия, изменения прав/настроек).

*Для кого:* администраторы/комплаенс.

**D) Экспорт данных**

* **Архивы пользователей/чатов:** JSON (структурированные данные), HTML (читабельный вид), контейнер ZIP (+ ссылки/вложения).

*Для кого:* пользователь (портируемость/право на данные), расследования.

**E) Наблюдаемость**

* **Логи:** приложения/доступа/ошибок (с редактированием чувствительных данных).
* **Метрики:** RPS, latency, error-rate, queue-lag и др.

*Для кого:* SRE/DevOps/аналитика.

## 3.2. Требования к надежности

### 3.2.1. Термины, охват и методика измерений

* Охват SLO/SLI: клиентски наблюдаемая доступность и производительность для зон:
  1. Core API (REST/gRPC), 2) Auth (аутентификация и выдача токенов), 3) Realtime (WebSocket/SSE), 4) Административная панель.
* Окно измерения: календарный месяц (UTC).
* SLI (доступность): доля успешных запросов с точки зрения клиента.

К успешным относятся коды 2xx/3xx и ожидаемые 4xx (например, 400/401/403/404); неуспех — 5xx, 429, 408, сетевые ошибки.

* Методика измерений: синтетические проверки (не реже одного раза в 30 секунд) минимум из трёх независимых точек, логи L7-балансировщика, распределённая трассировка. При расхождениях берётся худший из источников.

### 3.2.2. Доступность (SLO по зонам)

* Core API: 99,95 %/мес.
* Auth: 99,99 %/мес.
* Realtime: 99,90 %/мес (установление и поддержание соединений).
* Административная панель: 99,50 %/мес.

Плановые работы: допускается до 2 часов в месяц с уведомлением не менее чем за 48 часов; при соблюдении регламента время исключается из расчёта доступности.

#### 3.2.2.1. Непрерывность при недоступности Auth

* Действующие access-токены валидируются локально по кэшированным JWK/ключам в пределах их срока действия.
* Операции обновления токенов (refresh) ставятся в очередь и повторяются с экспоненциальной задержкой до восстановления Auth.
* Для внутренних сервисов допускается льготный период до 5 минут при подтверждённой сессии и соблюдении политики безопасности.
* В отчётности фиксируются доля запросов, обработанных в льготном режиме, и длительность автономной валидации.

### 3.2.3. Учёт и атрибуция простоев

* Простой фиксируется при одновременном подтверждении отказа минимум двумя источниками (проверки + логи L7).
* Сбои внешних провайдеров учитываются как простой, если они влияют на пользователя и деградация без отказа не была предусмотрена.

### 3.2.4. Отказоустойчивость и отсутствие единой точки отказа (3 сервера)

* Критические компоненты работают минимум в двух экземплярах и распределяются по S1/S2/S3 с правилами разведения нагрузок (anti-affinity).
* Кворумные системы (БД, брокер сообщений, поиск, средства координации) — кластер из трёх узлов (по одному узлу на каждый сервер).
* Применяются rolling, canary и blue-green-развёртывания с автоматическим откатом.

### 3.2.5. Восстановление (RPO/RTO, резервирование)

* При потере одного сервера: RTO ≤ 30 мин, RPO ≤ 5 мин (синхронная/полусинхронная репликация, автоматический переключатель).
* При потере всей площадки (все три сервера): восстановление из удалённого хранилища (off-site) бэкапов/архива WAL: RTO ≤ 8 ч, RPO ≤ 15 мин.
* Ежедневные полные и часовые инкрементные бэкапы; ежеквартальные учения восстановления (включая кэши, брокеры, поиск).

### 3.2.6. Производительность (SLO по задержкам и пропускной способности)

* Core API: p95 ≤ 200 мс, p99 ≤ 800 мс для чтения; p95 ≤ 300 мс, p99 ≤ 1000 мс для записи.
* Auth: p95 ≤ 150 мс, p99 ≤ 500 мс для выдачи/валидации токена.
* Realtime: установление WebSocket p95 ≤ 1,5 с; доставка события p95 ≤ 500 мс.
* Очереди/брокер: средний лаг ≤ 1 с, p95 ≤ 3 с.

#### 3.2.6.1. Референсный профиль нагрузки (для приёмочных испытаний)

* Core API: 1 500 запросов/с (80 % чтение, 20 % запись), всплеск до 5 000 запросов/с в течение 5 минут.
* Realtime: 50 000 одновременных WebSocket-подключений кластером; рассылка до 10 000 сообщений/с.
* Хранилище: пик 200 операций записи/с в БД; индексирование поиска 500 документов/с.
* Брокер: доля DLQ (очередь безнадёжных сообщений) ≤ 0,1 %; возврат лага к базовому уровню — за ≤ 10 минут после всплеска.

### 3.2.7. Деградация без отказа

* При перегрузке — ограничение скорости, размыкатель цепей, приоритизация критичных путей.
* При сбое брокера/поиска — асинхронная буферизация с гарантированной доставкой после восстановления.
* При отказе кэша — работа по схеме cache-miss → источник; превышение p99 допускается не более 10 минут.

### 3.2.8. Политики очередей и повторов

* Повторы с экспоненциальной задержкой, дедупликация сообщений, DLQ с регламентом обработки.
* SLA очистки DLQ: ≤ 24 часов; доля DLQ — ≤ 0,1 %.

### 3.2.9. Предупреждения и операционные цели

* Предупреждения по симптомам (латентность/ошибки/потеря кворума) с эскалацией по уровням SEV.
* MTTD ≤ 5 мин, MTTR ≤ 30 мин для SEV-1 (падение сервиса/узла).

### 3.2.10. Идемпотентность и повторная доставка

* Все изменяющие операции API принимают *Idempotency-Key*; повтор с тем же ключом не меняет состояние.
* Консьюмеры очередей обеспечивают практически однократную обработку критичных операций; применяется дедупликация по бизнес-ключам.

### 3.2.11. Коммуникации при инцидентах

* Уровни SEV, публичная статус-страница/канал оповещений.
* Обновления статуса: для SEV-1 — не реже каждых 15 минут, для SEV-2 — не реже каждых 30 минут.

## 3.3. Требования к составу и параметрам технических средств

### 3.3.1. Общие принципы и допущения

* Используются три сервера: S1, S2, S3. Все кворумные компоненты (СУБД, брокер сообщений, поиск, координация) работают в конфигурации трёх узлов — по одному узлу на каждый сервер.
* Применяются правила разведения нагрузок (*anti-affinity*): экземпляры одного кворума не концентрируются на одном сервере сверх неизбежного минимума.
* Балансировка L4/L7 развёрнута на всех трёх серверах; для входного VIP используется VRRP/Keepalived или эквивалент.
* Время синхронизируется по NTP; все узлы имеют аппаратный мониторинг, out-of-band-доступ.
* ОС серверов — современная LTS-дистрибуция Linux; обновления пакетов выполняются без простоя по схеме rolling.

### 3.3.2. Роли серверов (номинальный режим)

#### S1 — фронтовой узел и первичный контур данных

* Балансировка и входной трафик
  + L7-балансировщик (HAProxy/Envoy), участник VRRP (приоритет 150, предпочитаемый актив).
  + L4-балансировщик для WebSocket/SSE.
* База данных
  + PostgreSQL — primary; синхронная репликация на S2; архив WAL в удалённое хранилище (off-site).
* Приложения
  + Пулы Core API, Auth, Realtime-шлюз (ориентир — 1/3 инстансов кластера).
  + Канареечные релизы по умолчанию стравливаются через S1 (до 10 % трафика).
* Кворумы и очереди
  + Узел координации (etcd/RAFT).
  + Брокер сообщений (Kafka/NATS/RabbitMQ) — broker-1 (leader-eligible).
* Поиск и кэш
  + Узел поиска (OpenSearch/Elasticsearch) — data/master-eligible.
  + Sentinel для Redis (наблюдатель кворума).
* Наблюдаемость и секреты
  + Grafana/UI, алертер.
  + Узел хранилища секретов (Vault/экв.), участник кворума.

#### S2 — вычислительный узел и контур кэша

* Балансировка
  + L7-балансировщик, VRRP (приоритет 120, резерв №1).
* База данных и кэш
  + PostgreSQL — синхронная реплика (кандидат на автоматическое переключение в primary).
  + Redis — master, AOF/RDB; фейловер на S3.
* Приложения
  + Пулы Core API/Auth/Realtime (≈ 1/3 инстансов).
  + Фоновые воркеры/очереди задач (job-runner, scheduler).
* Кворумы и очереди
  + Узел координации (etcd/RAFT).
  + Брокер сообщений — broker-2 (leader-eligible).
* Поиск и наблюдаемость
  + Узел поиска — data/master-eligible.
  + TSDB (Prometheus/экв.) — основной сбор метрик; удалённая запись (remote-write) на S3.
* Секреты
  + Узел хранилища секретов (участник кворума).

#### S3 — логирование, реплика и службы восстановления

* Балансировка
  + L7-балансировщик, VRRP (приоритет 100, резерв №2).
* База данных и кэш
  + PostgreSQL — witness/arbiter (или async-replica при наличии ресурсов).
  + Redis — replica; при фейловере становится master.
* Приложения
  + Пулы Core API/Auth/Realtime (≈ 1/3 инстансов).
* Кворумы и очереди
  + Узел координации (etcd/RAFT).
  + Брокер сообщений — broker-3 (leader-eligible).
* Поиск и наблюдаемость
  + Узел поиска — data/master-eligible; задачи snapshot/restore индексов.
  + Лог-стор (Loki/ELK), ретеншн и архивирование; вторичный TSDB/приёмник remote-write.
* Бэкапы и DR
  + Оркестратор бэкапов: полные ежедневные и часовые инкрементные бэкапы БД (старт с S2), выгрузка в off-site; регулярные проверки чтения и контрольные восстановления.

### 3.3.3. Ресурсы, ретеншн и лимиты

* Аппаратные ориентиры (на каждый сервер S1/S2/S3): не ниже 16 vCPU, 64 ГБ RAM, NVMe SSD, сеть ≥ 10 GbE.
* Резерв по ресурсам: не менее 30 % CPU/RAM/IO к прогнозному пику.
* Ретеншн логов: операционные — ≥ 14 дней, аудит-логи — ≥ 90 дней.
* Квоты очередей/индексов: лимиты на размер сегментов/партиций и количество шардов с авто-тюнингом при росте MAU/RPS.

### 3.3.4. Сеть и безопасность

* Раздельные security-группы/ACL; принцип **минимально достаточных привилегий** для межсервисных взаимодействий.
* TLS-терминация на L7-балансировщиках (TLS 1.3); секреты хранятся в секрет-хранилище; ротация ключей — **без простоя**.
* Сканирование уязвимостей образов и CIS-профили для ОС/СУБД; контроль целостности конфигураций.

### 3.3.5. Энергоснабжение и размещение

* Узлы S1/S2/S3 разнесены по разным стойкам/линиям питания; **UPS**; мониторинг среды (температура/влажность).
* Наличие out-of-band-доступа и, по возможности, резервных каналов связи от разных провайдеров.

## 3.4. Требования к информационной и программной совместимости

### Стек разработки

* **Языки и спецификации**
  + C (C23, ISO/IEC 9899:2024); C++ (C++23, ISO/IEC 14882:2024).
  + JavaScript (ECMAScript 2025, ECMA-262 16th ed.); TypeScript ≥ 5.x (актуальный релиз).
  + HTML — Living Standard (WHATWG); CSS — актуальные модули по snapshot браузеров.
* **Рантаймы/инструменты**
  + Node.js 22 LTS (рекомендовано); допускается Node.js 24 после перевода в LTS в октябре 2025.
  + Пакетный менеджер npm/pnpm; сборка — Vite/webpack (по проекту).
* **Серверная платформа**
  + Linux (x86\_64/ARM64); контейнеры OCI; оркестрация Kubernetes (стабильная ветка).
  + Криптобиблиотека OpenSSL 3.5 LTS (TLS 1.3); допустимы совместимые FIPS-модули.
* **Сетевые протоколы и транспорт**
  + HTTP/1.1 + HTTP/2 + HTTP/3 (QUIC); WebSocket; опционально WebTransport over HTTP/3 (стандарт в разработке).
* **API и контракты**
  + REST/JSON (контракты OpenAPI 3.2.0); gRPC/Protobuf для внутренних RPC.
* **Данные и интеграции**
  + PostgreSQL 17.x (переход на 18 после GA, сентябрь 2025); Redis для кэша/сессий.
  + Поиск: OpenSearch/Elasticsearch (актуальная минорная).
  + Очереди/стриминг: Kafka / NATS JetStream / RabbitMQ (по окружению).
  + Хранилище медиа: S3-совместимое + CDN.
* **Клиент (Web)**
  + SPA на React 19 (или эквивалент), TypeScript, Vite; realtime — WebSocket.
  + Совместимость браузеров: последние 2 мажорные версии Chrome/Firefox/Edge/Safari (HTTP/2/3, WebSocket, Web Push).
* **Кодировки/форматы**
  + UTF-8; JSON/JSON Lines; MIME; бинарные — Protobuf/CBOR (по месту).
  + Крупные загрузки: multipart/chunked, resumable (S3 Multipart Upload).
* **Безопасность**
  + TLS 1.3 на внешнем периметре; mTLS внутри; пароли — Argon2id; passkeys (WebAuthn).
  + Политики: CSP, HSTS, SRI, SameSite cookies, защита XSS/CSRF, rate-limit, CAPTCHA на регистрации.
* **Лицензирование**
  + Допускается open-source; ПО не должно требовать лицензионных отчислений от конечных пользователей (no per-seat royalties).
* **Платформы доступа**
  + Современные браузеры (Desktop/Mobile) и мобильные клиенты (iOS/Android) — по отдельному разделу совместимости.
* **Доступность**
  + Веб-клиент соответствует WCAG 2.2 AA.

### Практическая детализация

* **Сервер**
  + Языки: C++23 (по микросервису); API-шлюз — HTTP/1.1/2/3, TLS-терминация; REST (OpenAPI 3.2.0), gRPC для внутренних вызовов.
  + Сериализация: JSON/Protobuf; валидация схем (OpenAPI/gRPC).
  + Очереди: Kafka/NATS/RabbitMQ; idempotency-key + DLQ.
  + Хранилища: PostgreSQL 17.x (готовность к 18), S3 для объектов; антивирус/превью — фоновые worker’ы.
  + Крипто: OpenSSL 3.5 LTS (TLS 1.3), OCSP stapling; ключи в KMS.
* **Веб-клиент**
  + HTML (Living Standard), ECMAScript 2025, React 19 + TypeScript; сборка — Vite; realtime — WebSocket, пуши — Web Push.
  + Браузерные политики: CSP/HSTS/SRI, COOP/COEP/CORP; offline-кэш, Service Worker.
* **Совместимость браузеров**
  + Поддерживаются последние 2 мажорные версии Chrome/Firefox/Edge/Safari (HTTP/2/3, WebSocket, Push, ES2025).
* **Кодировки и форматы**
  + UTF-8 везде; MIME для медиа; большие файлы — chunked/resumable, сервер — range-запросы, S3 Multipart.
* **Безопасность**
  + TLS 1.3 (наружу), mTLS (внутри), Argon2id (пароли), passkeys/WebAuthn; WAF/бот-фильтрация, rate-limit; аудит (structured logs).
  + Секреты в KMS, ротация ключей; CSP и строгая политика cookies.
* **Прочее**
  + i18n/l10n: RU/EN на релизе; ICU/CLDR.
  + CI/CD: проверки SAST/DAST, подпись артефактов (Sigstore), SBOM.

# 4. Требования к программной документации

Информативность документации должна соответствовать потребностям жизненного цикла ПС. При создании ПО должны быть подготовлены следующие документы (в соответствии с разделом шаблона):

1. **Архитектурное описание «Стриж» (ADR/диаграммы)** — контекст и состав модулей: Аутентификация и профили, Контакты, Чаты 1:1 и «секретные» (E2EE), Группы/супергруппы, Каналы/рассылки, Медиа/хранилище медиа, Поиск, Уведомления, Bot API, Модерация, Аналитика и мониторинг; схемы данных и интеграций; варианты развертывания на S1/S2/S3 (VRRP и приоритеты).
2. **Спецификации API системы** — машинно-читаемые контракты для внешних REST (OpenAPI 3.x) и внутренних gRPC/Protobuf; коды ошибок, лимиты и квоты; примеры. Обязательно покрыть: Auth (включая passkeys/WebAuthn и Argon2id), обмен сообщениями (в т.ч. HTTP/3 и WebSocket), загрузки медиа (chunked/resumable), поиск, уведомления, Bot API (webhook’и, подпись, опционально mTLS).
3. **Руководство системного программиста (развёртывание/эксплуатация)** — пошаговое развертывание кластера на трёх серверах, настройка VRRP и health-checks, политики обновлений (rolling, canary, blue-green), процедуры бэкапов/восстановления (RPO/RTO), регламенты обслуживания кэша/брокера/поиска, возврат узла в строй.
4. **Руководство оператора** — наблюдаемость и алертинг по целевым SLO/SLI, дешборды и трассировки, процедуры реагирования (MTTD/MTTR, SEV-уровни), работа с DLQ и политиками повторов, правила деградации без отказа.
5. **Программа и методика испытаний (приёмка системы)** — профиль нагрузочного теста из п. 3.2.6.1 (Core API, Realtime, БД/поиск/брокер), сценарии функциональных и интеграционных испытаний для модулей, проверки безопасности (TLS 1.3/mTLS, CSP, passkeys), критерии успешности.
6. **Руководство пользователя (для веб-клиента)** — сценарии: регистрация/вход/2FA, чаты 1:1 и E2EE, группы, каналы, медиа, поиск, уведомления, базовые функции ботов. Термины и UI — в соответствии с реализацией.
7. **Руководство администратора/модератора** — права и роли, антиспам/антирейд, обработка жалоб, аудит действий, отчётность.
8. **План обеспечения безопасности для «Стриж» (Threat Model)** — границы доверия, активы и угрозы, меры: TLS 1.3 наружу, mTLS внутри, Argon2id для паролей, passkeys/WebAuthn, CSP/HSTS/SRI, политика cookies, хранение секретов в KMS и ротация.

## 4.1. Актуализация

Документы обновляются синхронно с изменениями модулей/интерфейсов; выпуск изменений API невозможен без обновлённых спецификаций и миграционных инструкций (см. требования по совместимости/миграциям).