Функциональные характеристики

1. Аутентификация и сессии

Параметр	Описание
Входные данные	Имя пользователя и пароль
Действия	Проверка ограничителя попыток входа, поиск учетной записи, проверка хеша пароля, создание сессии
Выходные данные	Защищенная cookie сессии и cookie для CSRF-защиты либо сообщение об ошибке
Роли	Пользователь, администратор

Пароли хешируются функцией scrypt с индивидуальной солью. В базе хранится хеш пароля и HMAC-токена сессии, а не исходный пароль или исходный токен. Срок сессии задается настройкой.

2. Работа с чатами

Функция	Входные данные	Результат	Роль
Создание чата	Необязательное название, выбранная модель	Новый чат, принадлежащий текущему пользователю	Пользователь, администратор
Просмотр списка	Текущая сессия	До 500 чатов пользователя по времени обновления	Пользователь, администратор
Просмотр чата	Идентификатор чата	Сообщения и параметры чата	Владелец чата
Переименование	Идентификатор и новое название	Измененное название	Владелец чата
Смена модели	Идентификатор и модель	Модель сохраняется в параметрах чата	Владелец чата

Функция	Входные данные	Результат	Роль
Удаление	Идентификатор чата	Удаление чата и связанных сообщений	Владелец чата

Доступ к чужим чатам не предоставляется: операции выполняются с одновременной проверкой идентификатора пользователя.

3. Получение ответа модели

Входные данные:

- выбранная модель;
- текст сообщения пользователя;
- ограниченное настройкой число последних сообщений текущего чата.

Последовательность действий:

1. ПО проверяет длину и непустое содержание сообщения.
2. Сообщение сохраняется в базе данных.
3. Формируется запрос POST /v1/chat/completions с параметром stream=true.
4. Сервер модели или шлюз возвращает поток Server-Sent Events.
5. ПО разделяет обычный ответ и рассуждения модели, если они представлены отдельным полем или тегами <think>.
6. Браузер получает события meta, reasoning, delta, done или error.
7. Завершенный ответ и доступные метрики сохраняются в истории.

Выходные данные:

- текст ответа;
- необязательный текст рассуждений;
- модель, число входных и выходных токенов, время до первого токена и общая длительность (если эти сведения доступны).

Остановка пользователем закрывает текущий HTTP-запрос. Возможность немедленно прекратить вычисление зависит от поведения подключенного сервера модели.

4. Отображение ответа

Интерфейс поддерживает абзацы, заголовки, списки, ссылки, выделение и блоки кода. Текст пользователя и модели помещается в DOM через textContent и createTextNode; произвольный HTML модели не исполняется.
Блок рассуждений можно свернуть. Наличие и достоверность рассуждений определяются используемой моделью и сервером вывода.

5. Получение списка моделей

ПО объединяет список из переменной MODELS и идентификаторы из GET /v1/models. Результат показывается пользователю в выпадающем списке. При недоступности сервера используется только явно заданный резервный список.

6. Администрирование пользователей

Администратор может:

- просматривать учетные записи;
- создавать пользователя или администратора;
- менять роль;
- задавать новый пароль;
- блокировать и разблокировать учетную запись.

Администратор не может заблокировать или понизить собственную текущую учетную запись. При блокировке или смене пароля существующие сессии соответствующего пользователя удаляются.

7. Статистика и журнал

Администратору доступны:

- общее количество пользователей, чатов и сообщений;
- число сообщений, ошибок модели и входов за 24 часа;
- по каждой модели: число ответов, среднее время до первого токена, число токенов и средняя скорость;
- 50 последних событий приложения.

Статистика строится по локальной базе SQLite. Она предназначена для эксплуатации, а не для финансовой или регламентированной отчетности.

8. Управление шлюзом моделей

Функция доступна только при GATEWAY_ADMIN_ENABLED=true.
Администратор получает состояние шлюза, активную модель, состояние переключения, длину очередей, телеметрию памяти ускорителя и сведения о последнем переключении. Команда переключения передается в /admin/switch и выполняется в фоне.

Шлюз реализует:

- очереди по идентификаторам моделей;
- обслуживание активной модели без лишних переключений;
- ограничение времени ожидания другой модели;
- завершение выполняющихся запросов перед остановкой сервера;
- запуск сервера модели по профилю;
- ожидание готовности /v1/models;
- откат к предыдущей модели при неудачном запуске;

- обнаружение сбоя активного сервера и автоматические повторные запуски;
- режим без ожидания с ответом 503 и Retry-After;
- потоковую передачу ответа без накопления полного ответа в шлюзе.

При использовании адаптера аппаратной телеметрии шлюз дополнительно проверяет освобождение памяти и наличие посторонних процессов. При TELEMETRY_PROVIDER=none эта проверка отключена, поэтому контроль ресурсов должен обеспечиваться внешней системой.

9. Резервное копирование

Инструмент резервного копирования использует SQLite Online Backup API, проверяет копию через PRAGMA integrity_check, сжимает ее в gzip и удаляет старые архивы сверх заданного количества. Восстановление выполняется отдельным сценарием с предварительной проверкой копии и сохранением заменяемой базы.

10. Ограничения

- Качество, точность и допустимость ответа определяются подключенной моделью; ПО не гарантирует истинность сгенерированного текста.
- ПО не содержит механизмов обучения или дообучения модели.
- ПО не является средством защиты информации и не имеет соответствующей сертификации.
- Встроенного биллинга, тарификации, интеграции с каталогами LDAP/AD и многофакторной аутентификации нет.
- Встроенного распределенного хранилища нет; основное хранилище представляет собой локальный файл SQLite.
- В одном экземпляре собственного шлюза одновременно активна одна группа сервера модели.
- Поддержка конкретной модели зависит от внешней среды вывода.