

Проект *Llamagramm* — это автономный агент, реализованный на языке Python 3.12 с использованием фреймворка *aiogram 3.x* (для сетевого взаимодействия) и открытого API языковой модели (OpenAI-совместимым). Его задача — предоставить пользователям и исследователям быстрый доступ к интеллектуальному диалогу: агент принимает текст, файлы и другие данные, формирует запросы к языковой модели, обрабатывает полученные ответы, сохраняет историю взаимодействий и выводит результат в машиночитаемом формате.

§ Функциональные возможности

Функция	Описание
Обработка входящих данных	Принимает текстовые сообщения и файлы, автоматически преобразует их в строковый вид и передаёт модели.
Интеграция с языковой моделью	Использует открытый сервис (например, <i>Llama.cpp</i> , OpenAI, Anthropic) с настройкой температуры, длины ответа и потоковой выдачи.
История общения	Сохраняет контекст беседы в оперативной памяти и в файловом журнале, что позволяет вести диалог с сохранением текущего состояния.
Форматирование ответов	Убирает нежелательные теги, выделяет рассуждения в блоки «спойлер», обрезает сообщения до 4096 символов в соответствии с ограничениями, установленными мессенджером.
Логирование	Записывает все сообщения и ошибки в отдельные каталоги, поддерживает ротацию файлов и уровни детализации.
Управление администраторами	Предоставляет команды <code>/start</code> , <code>/reset</code> , <code>/stop</code> (только для уполномоченных пользователей).
Асинхронность	Полностью асинхронный код на <i>asyncio</i> , позволяющий обрабатывать сотни запросов одновременно без блокировок.

§ Технологический стек

- **Python 3.12** – современный язык высокого уровня с богатой экосистемой.
- **aiogram 3.x** – высокопроизводительный фреймворк для сетевого взаимодействия.
- **httpx** – асинхронный HTTP-клиент для обращения к сервису языковой модели.

- **pydantic-settings** — безопасная загрузка конфигурации из файла `.env`.
 - **logging + RotatingFileHandler** — централизованная система ведения журналов.
 - **pytest + pytest-asyncio** — модульные тесты, обеспечивающие надёжность.
 - **systemd** — развёртывание.
-

§ Педагогическая ценность

1. **Практика работы с искусственным интеллектом** — студенты наблюдают за тем, как формируются запросы, обрабатываются ответы и сохраняется контекст.
2. **Развитие навыков асинхронного программирования** — в проекте используются `asyncio`, `Lock` и асинхронные файловые операции.
3. **Понимание архитектурного подхода** — разделение на уровни (агент, промежуточное ПО, маршрутизатор, сервисы, утилиты) демонстрирует чистую структуру.
4. **Тестирование и CI/CD** – набор автоматических тестов и Docker-конфигурация обучают созданию полностью проверяемого продукта.
5. **Подготовка к реальным проектам** – опыт интеграции с внешними сервисами, работы с ограничениями сетевых протоколов и безопасного хранения токенов.

Исходный код проекта «llamagramm» : закрытый репозиторий [Электронный ресурс]. Режим доступа: <https://gitflic.ru/project/ivanovga/llamagramm> (дата обращения: 23.03.2026).