

Prompt2Pic

Ĺ HĠÉ3HǺ! ĠGFĠĒFĒǺHĒĒĠǺ ĦHĠĠǺ Ēt ĠĠĒĒĠĒĒ Ēt ĠĠĒĒĠĒĒ
Ĺ ĒĒĒĠĒĒ FĒĒ
Ĺ ĒH3HĠĠĒĒĒĒĒ
ŸĒĠĒĒĒFĒĒĒĒĒ

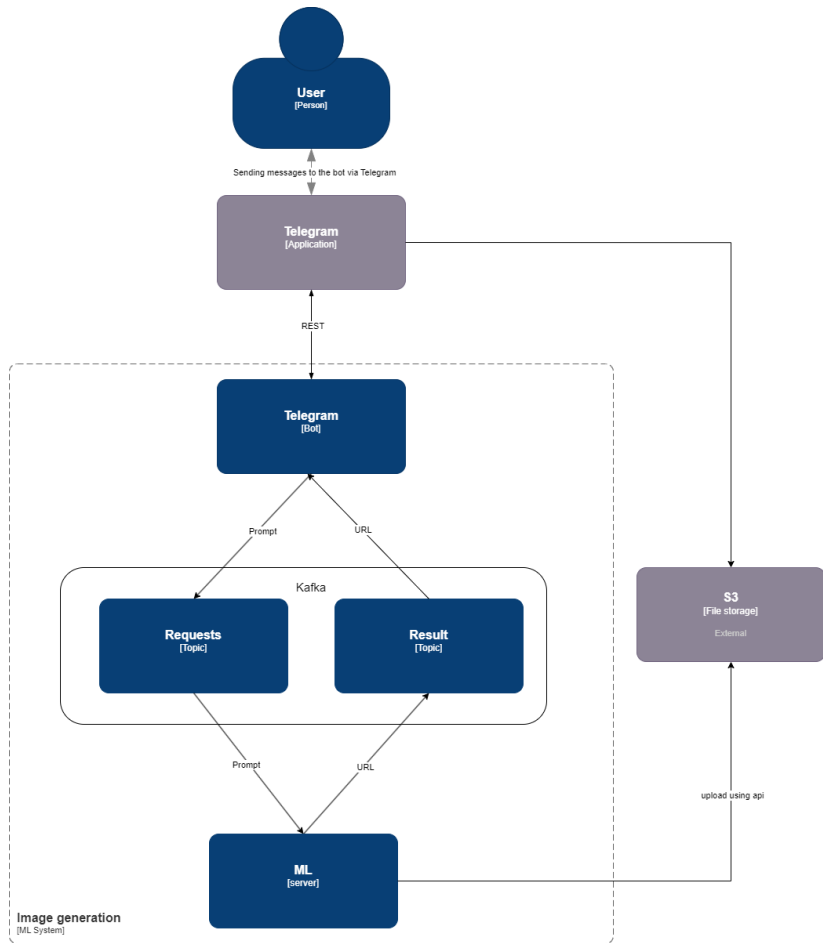
Цель проекта

- **Бизнес-цель:** создание удобного сервиса, позволяющего пользователям генерировать изображения по текстовым описаниям через Telegram, что упрощает доступ к возможностям генеративных моделей.
- **ML-цель:** использование предобученной оптимизированной модели машинного обучения (Würstchen) для генерации изображений высокого качества с минимальными затратами ресурсов CPU.

Задачи проекта

- Разработать надёжную архитектуру, обеспечивающую:
 - масштабируемость: поддержка большого числа пользователей.
 - производительность: минимальные задержки при обработке запросов.
- Настроить взаимодействие между компонентами через Apache Kafka как надёжную систему очередей.
- Упаковать приложение в Docker-контейнеры для простоты развертывания.
- Реализовать защищённый процесс деплоя на облачной платформе cloud.ru.
- Развернуть рабочую демо-версию, доступную через Telegram.

Архитектура проекта



Пользователь взаимодействует с ботом через Telegram. Бот отправляет запросы в серверную часть, где текстовые описания преобразуются в изображения. Результаты возвращаются пользователю.

Почему такая архитектура?

- Использование Apache Kafka позволяет распределять нагрузку и обеспечивать надёжную передачу данных.
- Контейнеризация с Docker упрощает развертывание и масштабирование.
- Telegram обеспечивает простой доступ для пользователей через привычный интерфейс.

Почему мы выбрали Telegram?

- Широкая доступность: Telegram используется на всех типах устройств и имеет простую интеграцию через Telegram API.
- Скорость разработки: создание бота не требует больших затрат времени, благодаря удобной библиотеке Python (например, aiogram).
- Удобство для пользователей: интерфейс Telegram интуитивно понятен, что снижает порог входа.

Почему Stable Diffusion?

- Качество: модель генерирует реалистичные изображения высокого разрешения.
- Скорость: обработка запроса занимает считанные секунды.
- Гибкость: модель позволяет кастомизировать результаты с помощью fine-tuning или дополнительных параметров.
- Поддержка сообщества: стабильное развитие благодаря open-source экосистеме.

Почему Apache Kafka?

- Производительность: Kafka может обрабатывать миллионы сообщений в секунду, что важно для масштабируемых приложений.
- Распределённость: позволяет распределять нагрузку между несколькими серверами, обеспечивая отказоустойчивость.
- Гибкость: поддерживает как асинхронное, так и синхронное взаимодействие между компонентами.
- Надёжность: все сообщения сохраняются в логах, что предотвращает потерю данных при сбоях.

Пример использования

- Пользователь вводит текстовое описание через команду /txt2img.
- Telegram-Bot отправляет запрос в Kafka.
- ML-сервер обрабатывает запрос, генерирует изображение через Stable Diffusion и сохраняет его во внешнем хранилище.
- Ссылка на изображение возвращается пользователю через Telegram.

Доступ к боту

- Вы можете найти бота в Telegram по имени: @MAINeuroML_bot