

Критерии приемки (АС)

АС-001: Анализ валидной User Story

Дано: Пользователь отправляет валидную User Story

Когда: Выполняется запрос анализа

Тогда: Система возвращает анализ с оценкой INVEST и рекомендациями

АС-002: Обработка невалидного формата

Дано: Пользователь отправляет текст в неправильном формате

Когда: Выполняется проверка валидности

Тогда: Система возвращает сообщение об ошибке с инструкцией

АС-003: Использование кэша

Дано: Пользователь отправляет уже анализировавшуюся User Story

Когда: Выполняется поиск в кэше

Тогда: Система возвращает результат из кэша

АС-004: Поиск с порогом схожести

Дано: В базе есть истории с разной степенью схожести

Когда: Пользователь отправляет запрос

Тогда: Система возвращает истории с схожестью \geq установленного порога

АС-005: Улучшение User Story

Дано: Пользователь нажимает "Улучшить историю"

Когда: Выполняется запрос улучшения

Тогда: Система возвращает улучшенную версию с сохранением цели

АС-006: Предотвращение дубликатов

Дано: Пользователь пытается добавить существующую User Story

Когда: Выполняется проверка уникальности

Тогда: Система пропускает дубликат (в seed) или показывает сообщение

АС-007: Автоматическое добавление улучшенных историй

Дано: Пользователь улучшает историю через ИИ

Когда: Оценка улучшенной истории $\geq 4/6$

Тогда: Система автоматически сохраняет ее как golden историю

Нефункциональные требования (NFR)

NFR-PERF-001: Время отклика

Требование: Анализ User Story выполняется за разумное время (< 10 секунд)

Реализация: Асинхронная обработка, кэширование

NFR-REL-001: Отказоустойчивость GigaChat

Требование: Система обрабатывает ошибки GigaChat API

Реализация: Try-catch блоки, повторные попытки

NFR-REL-002: Целостность данных

Требование: Данные сохраняются в PostgreSQL

Реализация: ACID транзакции через asynprg

NFR-SEC-001: Базовая аутентификация

Требование: Доступ только через авторизованного Telegram бота

Реализация: Telegram Bot Token