

Use Case: SafeValBot - Проверка покрытия FSR тест-кейсами

Заголовок	Engineer проверяет покрытие FSR тест-кейсами и генерирует недостающие тесты в SafeValBot (Telegram Mini App / Web).
Акторы	Engineer (основной актор), SafeValBot (TG/Web), LLM-агент (RAG + генерация тестов), Backend API (FastAPI), PostgreSQL, Zephyr/Jira.
Предусловия	В PostgreSQL есть SG и FSR, доступ к проекту настроен, интеграция с Zephyr или экспорт в файлы, доступен Web/TG интерфейс.
Постусловия	Матрица SG→FSR→Test Case актуальна, Missing FSR закрыты шаблонами TC, связи FSR↔TC сохранены, отчёт доступен.
Ограничения	Макс. 2000 FSR в одном расчёте, фильтрация по текущему проекту, сложные графики только в Web, LLM использует RAG.
Триггер	Engineer запускает SafeValBot и выбирает «Проверить покрытие».
Основной сценарий (Web)	<div>1. Авторизация и выбор проекта.</div> <div>2. Нажатие «Coverage» — загрузка SG и FSR.</div> <div>3. Выбор FSR для проверки.</div> <div>4. Подсчёт Covered/Missing/Failed и KPI.</div> <div>5. Генерация тестов для Missing с учётом ASIL.</div> <div>6. Сохранение TC в PostgreSQL и/или Zephyr.</div> <div>7. Экспорт отчёта (PDF/CSV/SQL).</div> <div>8. Обновление матрицы покрытия.</div>
Основной сценарий (TG Bot)	<div>1. /start → меню.</div> <div>2. Выбор «Проверить покрытие».</div> <div>3. Отображение FSR и выбор.</div> <div>4. Показ Covered/Missing/Failed.</div> <div>5. Генерация TC для Missing.</div> <div>6. Сохранение в PostgreSQL/Zephyr.</div> <div>7. Экспорт отчёта.</div>
Альтернативные сценарии	Нет FSR → импорт CSV/JSON, Частично заполнены связи → Partial, Выбор только Covered → экспорт без генерации.
Исключительные сценарии	Ошибка БД, LLM недоступен, ошибка Zephyr API, неверная команда в боте → /help.

**Use Case: SafeValBot - Проверка покрытия FSR тест-кейсами**

**Критерии успеха**

Use Case: SafeValBot - Проверка покрытия FSR тест-кейсами

Технические детали

- Время расчёта покрытия $\leq 2$ сек для $\leq 2\,000$ FSR.
- Доля корректной генерации ТС (валидная структура Zephyr) $\geq 95\%$ .
- Снижение Missing после автогенерации $\geq 80\%$ для выбранного набора.
- Синхронизация Web $\leftrightarrow$ Telegram $\leq 1$ сек по снимоту отчёта.
Web: React/Next.js, FastAPI, PostgreSQL, pgvector, S3. TG Bot: aiogram, WebApp initData. Экспорт: pdfmake/canvas. RAG: SG/FSR, политики, шаблоны метрик.