

# Отчёт по реализации модуля аналитики

## Automaspec - Analytics Dashboard

Дата: Декабрь 2025

Версия: 1.0

---

### 1. Введение

В рамках дипломной работы был реализован модуль аналитики для системы управления тестовыми спецификациями Automaspec. Данный модуль обеспечивает сбор, обработку и визуализацию пользовательских метрик, что полностью соответствует критерию «Сбор, обработка и визуализация пользовательских метрик, отчёты и графики».

---

### 2. Реализованный функционал

#### 2.1 Расчёт ключевых метрик

На основе данных в базе формируются следующие метрики:

Метрика	Описание
Количество тестов	Общее число тестов в организации
Количество требований	Общее число тестовых требований
Количество спецификаций	Общее число тестовых спецификаций
Активные пользователи	Количество членов команды в организации
Распределение по статусам	Тесты сгруппированные по статусам (passed, failed, pending и др.)
Устаревшие тесты	Спецификации, не обновлявшиеся за выбранный период

---

#### 2.2 Визуализация в виде Dashboard

Dashboard включает следующие компоненты:

- Карточки ключевых показателей (Metrics Cards)**
    - Total Tests - общее количество тестов
    - Requirements - количество требований
    - Specifications - количество спецификаций
    - Team Members - активные члены организации
  - График роста числа тестов (Tests Growth Chart)**
    - Линейный график с точками данных
    - Отображает динамику создания тестов во времени
    - Интерактивный tooltip при наведении
  - Диаграмма распределения по статусам (Status Distribution Chart)**
    - Визуализация тестов по статусам
    - Показывает passed, failed, pending, skipped, todo и другие статусы
  - Таблица устаревших тестов (Stale Tests Table)**
    - Список спецификаций, не обновлявшихся за выбранный период
    - Отображает название спецификации и время последнего обновления
-

### 3. Архитектура решения

#### 3.1 Backend (oRPC)

Серверная часть реализована с использованием oRPC framework:

orpc/routes/analytics.ts

Эндпоинт: analytics.getMetrics

**Входные параметры:** - period - период для анализа (7d, 30d, 90d)

**Возвращаемые данные:** - totalTests - общее количество тестов - totalRequirements - количество требований - totalSpecs - количество спецификаций - activeMembers - активные пользователи - testsByStatus - распределение по статусам - testsGrowth - данные для графика роста - staleTests - список устаревших спецификаций

#### 3.2 Frontend (React + Next.js)

Клиентская часть построена на компонентной архитектуре:

```
app/analytics/
  page.tsx          #
  components/
    analytics-header.tsx  #
    metrics-cards.tsx    #
    tests-growth-chart.tsx #
    status-distribution-chart.tsx #
    stale-tests-table.tsx  #
```

#### 3.3 Визуализация (Recharts)

Для построения графиков используется библиотека Recharts: - LineChart - для графика роста тестов - PieChart / BarChart - для распределения по статусам

---

### 4. Скриншоты интерфейса

#### 4.1 Dashboard с метриками (30 дней)

*Рисунок 1: Главная страница аналитики с карточками ключевых метрик*

#### 4.2 Dashboard с выбором периода 7 дней

*Рисунок 2: Dashboard с выбранным периодом 7 дней*

#### 4.3 Таблица устаревших тестов

*Рисунок 3: Таблица устаревших спецификаций с динамическим периодом*

---

### 5. Функциональные возможности

#### 5.1 Выбор периода анализа

Пользователь может выбрать один из трёх периодов: - **7 дней** - краткосрочный анализ - **30 дней** - стандартный период (по умолчанию) - **90 дней** - долгосрочный анализ

При смене периода автоматически пересчитываются: - Данные графика роста тестов - Список устаревших спецификаций - Текст описания в таблице устаревших тестов

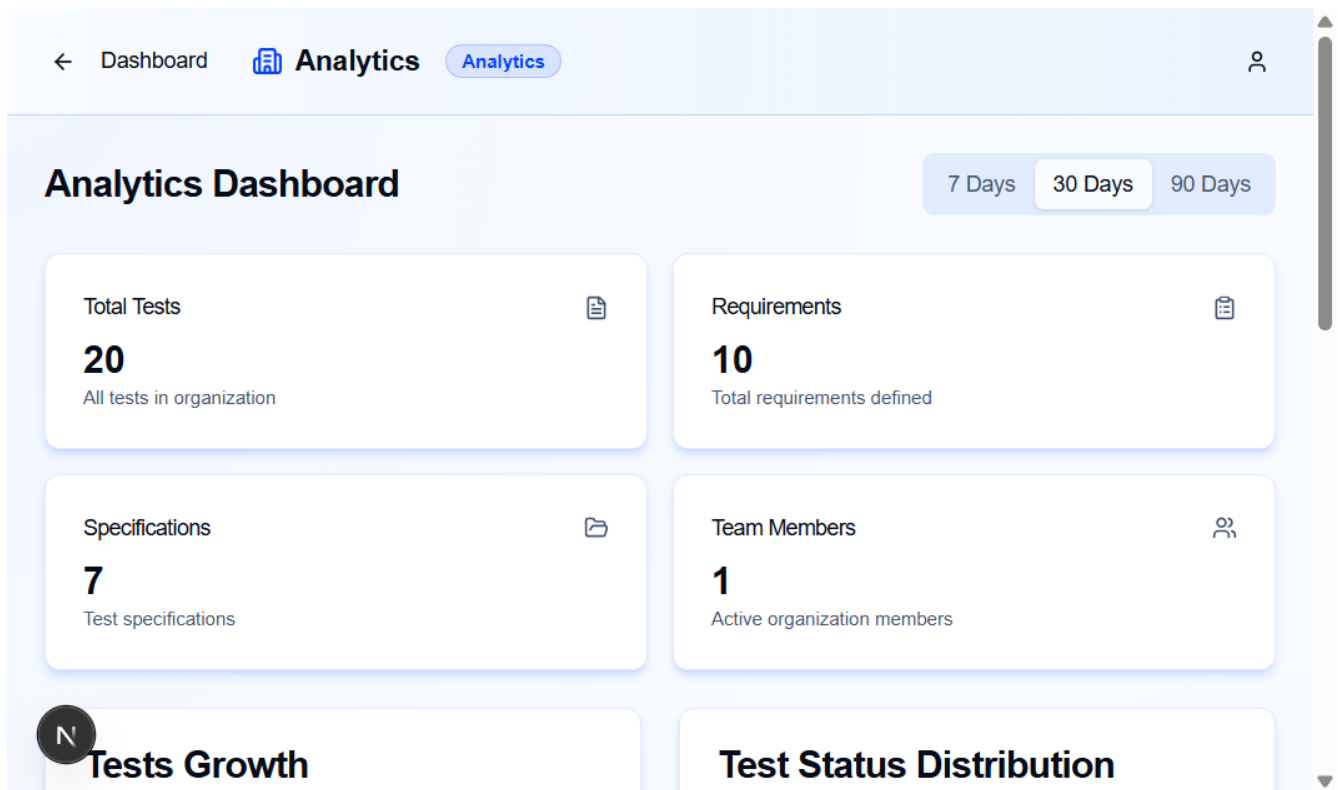


Figure 1: Analytics Dashboard - 30 Days

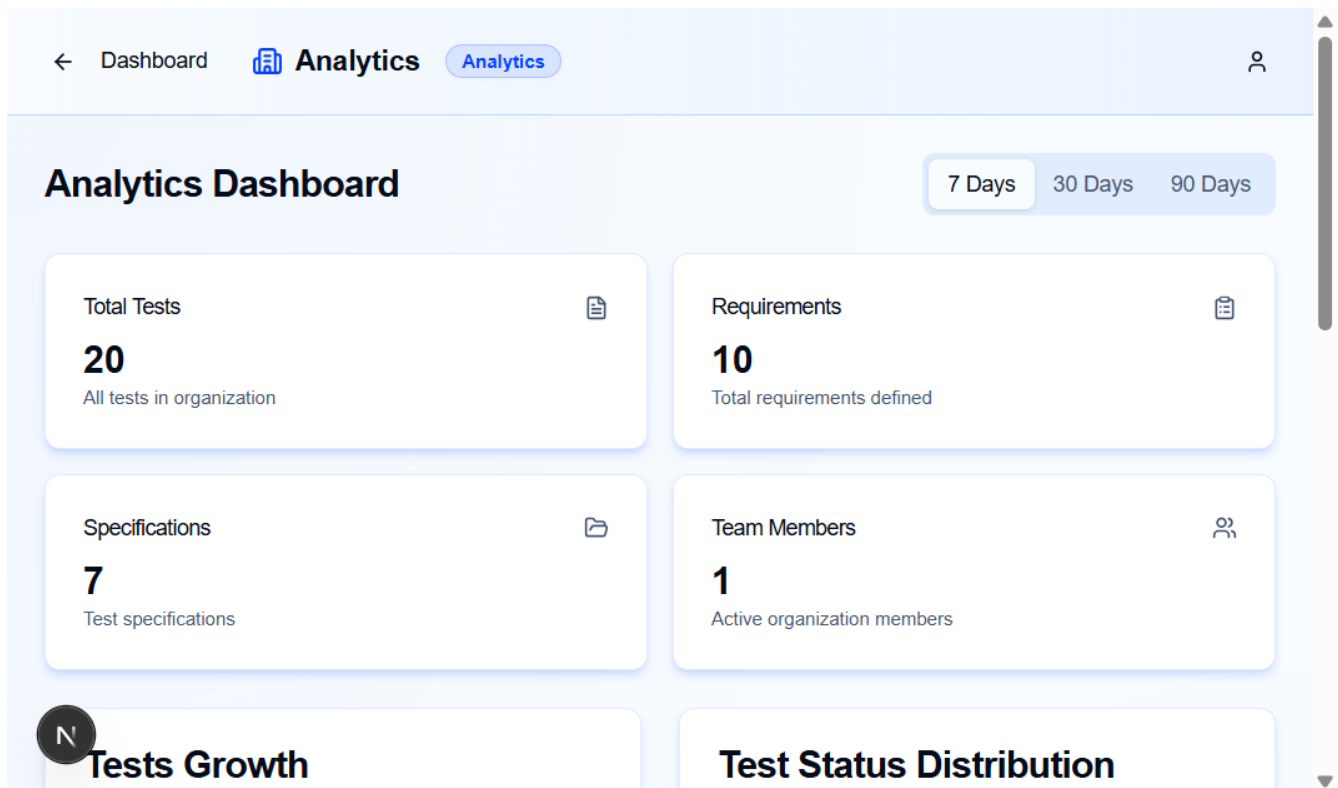


Figure 2: Analytics Dashboard - 7 Days

Specification Name	Last Updated
Legacy Auth Tests	about 2 months ago
Deprecated API	2 months ago
Old Utils	about 1 month ago

Figure 3: Stale Tests Table

## 5.2 Интерактивные элементы

- **Графики** - при наведении отображается tooltip с точными значениями
- **Точки данных** - на графике роста отображаются точки для каждой даты
- **Tabs** - переключение между периодами без перезагрузки страницы

## 6. Технические детали

### 6.1 Используемые технологии

Технология	Назначение
Next.js 16	Framework для React
React 19	UI библиотека
TypeScript	Типизация
Recharts	Визуализация данных
TanStack Query	Управление серверным состоянием
oRPC	API layer
Drizzle ORM	Работа с базой данных
Tailwind CSS	Стилизация

### 6.2 Структура базы данных

Метрики собираются из следующих таблиц: - test\_spec - спецификации (statuses, updatedAt) - test\_requirement - требования - test - тесты (status, createdAt) - member - члены организации

### 6.3 Константы

```
// lib/constants.ts
export const ANALYTICS_PERIODS = {
  '7d': 7,
  '30d': 30,
```

```
'90d': 90  
} as const
```

---

## 7. Соответствие требованиям

Требование	Реализация	Статус
Количество активных пользователей за период	Team Members карточка	<input type="checkbox"/>
Количество созданных тестов и требований	Total Tests, Requirements карточки	<input type="checkbox"/>
Распределение тестов по статусам	Status Distribution Chart	<input type="checkbox"/>
«Возраст» тестов	Stale Tests Table	<input type="checkbox"/>
График роста числа тестов	Tests Growth Chart	<input type="checkbox"/>
Диаграмма распределения по статусам	Status Distribution Chart	<input type="checkbox"/>
Карточки ключевых показателей	Metrics Cards	<input type="checkbox"/>
Список устаревших тестов	Stale Tests Table с динамическим периодом	<input type="checkbox"/>

---

## 8. Заключение

Реализованный модуль аналитики полностью закрывает процесс сбора, обработки и визуализации пользовательских метрик. Dashboard обеспечивает прозрачное понимание активности пользователей и «здоровья» тестовых спецификаций.

Функционал соответствует заявленному критерию analytics и был реализован в рамках дипломной работы с использованием современных технологий и лучших практик разработки.

---

**Автор:** Студент

**Проект:** Automaspec - Система управления тестовыми спецификациями