

SMART-план проекта

Цель проекта (SMART)

S (Specific): Создать автоматизированную систему сбора и мониторинга данных о загрузенности отелей с сайтов-агрегаторов (101hotels, Ostrovok, Яндекс.Путешествия)

M (Measurable):

- Интеграция минимум 3 источников данных
- Сбор данных по минимум 100 отелям
- Время обновления данных 24 часа

A (Achievable): Проект основан на существующей наработке, требуется доработка и расширение функционала

R (Relevant): Актуально для туристической отрасли Иркутска, позволит отслеживать рыночную ситуацию

T (Time-bound): Срок реализации - 12-16 недель

Декомпозированный план задач

Этап 1: Изучение и восстановление базового функционала (3-4 недели)

Задача 1.1: Анализ существующего кода

- **Срок:** Неделя 1 (5 дней)
- **Результат:** Документация по архитектуре проекта, выявленные проблемы
- **Критерии:** Составлена схема работы системы, определены зависимости

Задача 1.2: Миграция на новый API 101hotels

- **Срок:** Неделя 2-3 (10 дней)
- **Результат:** Работающий парсер с новым API endpoint
- **Критерии:**
 - Успешное получение данных по test hotel_id
 - Обработка минимум 50 отелей
 - Обработка ошибок и rate limiting

Задача 1.3: Тестирование и валидация данных

- **Срок:** Неделя 4 (5 дней)
 - **Результат:** Проверенный функционал сбора данных
 - **Критерии:** Корректность данных подтверждена сравнением с сайтом
-

Этап 2: Расширение базы источников (4-5 недель)

Задача 2.1: Интеграция Ostrovok.ru

- **Срок:** Неделя 5-6 (10 дней)
- **Результат:** Модуль парсинга Ostrovok
- **Критерии:**
 - Реверс-инжиниринг API/парсинг сайта
 - Сбор данных: название, цена, доступность
 - Унификация формата данных

Задача 2.2: Интеграция Яндекс.Путешествия

- **Срок:** Неделя 7-8 (10 дней)
- **Результат:** Модуль парсинга Яндекс.Путешествия
- **Критерии:** Аналогично задаче 2.1

Задача 2.3: Создание единой базы данных

- **Срок:** Неделя 9 (5 дней)
 - **Результат:** Консолидированная БД с данными из всех источников
 - **Критерии:**
 - Схема БД поддерживает все источники
 - Реализована дедупликация отелей
 - Настроены индексы для быстрого поиска
-

Этап 3: Система устойчивости и мониторинга (3-4 недели)

Задача 3.1: Разработка механизма обнаружения изменений API

- **Срок:** Неделя 10-11 (8 дней)
- **Результат:** Модуль валидации структуры ответов API
- **Критерии:**
 - Проверка по ключевым полям (цена, доступность, название)
 - Сравнение со схемой данных
 - Логирование несоответствий

Задача 3.2: Система оповещений

- **Срок:** Неделя 11-12 (7 дней)
- **Результат:** Автоматические уведомления о сбоях
- **Критерии:**
 - Email/Telegram уведомления
 - Оповещение при: отсутствии данных >24ч, изменении структуры API, критических ошибках
 - Настраиваемые пороги алертов

Задача 3.3: Fallback-механизмы

- **Срок:** Неделя 12-13 (5 дней)
 - **Результат:** Альтернативные методы сбора данных
 - **Критерии:**
 - Web scraping как backup для API
 - Кэширование последних успешных данных
 - Graceful degradation функционала
-

Этап 4: Тестирование и развёртывание (2-3 недели)

Задача 4.1: Комплексное тестирование

- **Срок:** Неделя 14 (5 дней)
- **Результат:** Протестированная система
- **Критерии:**
 - Unit-тесты (покрытие $\geq 70\%$)
 - Интеграционные тесты
 - Нагрузочное тестирование

Задача 4.2: Документация

- **Срок:** Неделя 15 (5 дней)
- **Результат:** Полная техническая документация
- **Критерии:**
 - README с инструкциями по развёртыванию
 - API документация
 - Руководство по мониторингу

Задача 4.3: Развёртывание и запуск

- **Срок:** Неделя 16 (5 дней)
- **Результат:** Работающая production-система
- **Критерии:**
 - Развёртывание на сервере
 - Настройка автоматических задач (cron/scheduler)
 - Мониторинг первых циклов сбора данных