

Научный руководитель – Харитоновна Анастасия Евгеньевна

Введение. Ежедневно совершается огромное количество полётов пассажирских самолётов, и существует необходимость анализировать их данные для оптимизации расходов авиакомпаний. При этом возникает проблема: по каждому рейсу авиакомпании получают большой объём информации. Поэтому необходимо обрабатывать эти данные, причём авиакомпании должны иметь возможность анализировать и сравнивать рейсы, затрачивая как можно меньше времени на их получение. Именно поэтому важно оптимизировать работу системы, чтобы авиакомпании могли быстрее взаимодействовать с ней для анализа результатов своих рейсов.

Основная часть.

Одной из главных составляющих данного приложения является представление, хранение и обработка основных данных о рейсах. К таким показателям относятся номер рейса, дата вылета, данные о загрузке рейса, топливные показатели, а также расчётные характеристики — оценки топливной и коммерческой эффективности.

Для представления этой информации используются следующие страницы:

Страницы с табличным представлением информации, фильтруемой по маршруту, табельному номеру пилота, типу воздушного судна (ВС) и временному промежутку. Поскольку данных очень много, для удобства на страницах с табличным представлением была введена группировка по маршрутам и типам ВС с указанием средней эффективности и количества рейсов в выборке. При нажатии на группу отображается информация обо всех рейсах, входящих в неё.

Страницы с группировкой данных по дням, неделям, месяцам, кварталам и годам в зависимости от выбранного периода. Это позволяет сравнивать разные периоды, выводить сводную статистику авиакомпании, а также формировать и сравнивать данные об эффективности пилотов за период.

В базе данных Postgres для хранения этих данных существует множество таблиц. Чтобы получать всю необходимую информацию о рейсах из одного источника, используется представление (view). Однако из-за большого объёма данных по каждому рейсу загрузка всех записей занимает много времени, а при выборке за большой период может возникать ошибка `java.lang.OutOfMemoryError`. Поэтому требуется оптимизация вывода данных.

Были реализованы следующие решения:

Для представления по рейсам создана промежуточная таблица, содержащая только основную информацию по каждому рейсу. На её основе формируется группировка по маршрутам и типам ВС. При выборе группы информация обо всех рейсах в ней отображается с использованием постраничной навигации (пагинации), где запрос данных для каждой страницы выполняется отдельно.

Для группировки по периодам создана промежуточная таблица с агрегированной информацией по дням: количество топлива, суммарная топливная и коммерческая эффективность, количество рейсов, а также данные по типам воздушных судов и пилотам. Это позволяет быстро выводить информацию за любой период, суммируя результаты за дни, входящие в него.

Выводы. Разработано веб-приложение для авиакомпаний, за счёт применённых методов оптимизации работы с данными, позволяет получать информацию о выполненных рейсах по заданным фильтрам в различных разрезах (по топливной и коммерческой эффективности, по рейтингу пилотов) и формах представления (таблицы и графики), благодаря чему аналитики и руководство быстрее и удобнее сможет получать статистику для последующего принятия решений на основе полученной информации о рейсах.

Список использованных источников:

1. Радченко И.А., Николаев И.Н. Технологии и инфраструктура Big Data: учебное пособие. – СПб.: Университет ИТМО, 2018. – 53 с.
2. Оптимизация работы с REST API: когда и как использовать пагинацию (+ простой пример на Python) [Электронный ресурс] — URL: <https://habr.com/en/companies/selectel/articles/890824/>