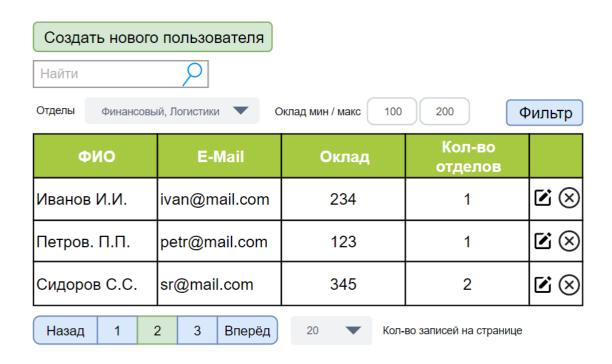


ТЕСТОВОЕ ЗАДАНИЕ

Создать приложение - список пользователей. Ниже приведён пример наброска UI, который есть у заказчика.



Принадлежность пользователя к отделу:

- Пользователь должен обязательно принадлежать хотя бы к одному отделу
- Пользователь может принадлежать нескольким отделам (см. приложение А)
- Отделы могут иметь неограниченную вложенность друг в друга (в UI представлены как дерево)
- Если пользователь принадлежит отделу с дочерними отделами, это не означает принадлежность его к дочерним отделам
- Отделы хранятся в БД и не могут быть изменены пользователем. Их список создается при первой миграции

Список пользователей

Необходим запрос для формирования списка, как на рисунке выше. Сортировка по ФИО.

- ФИО с сокращением ИО
- E-Mail
- Оклад
- Кол-во отделов у пользователя

Фильтрация списка пользователей

Список должен иметь функционал:

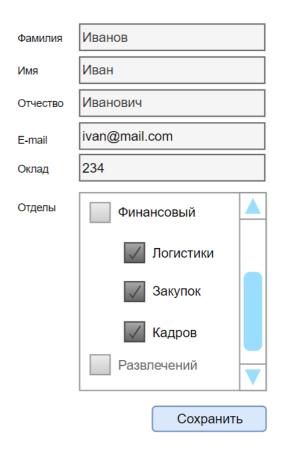
- Серверной паджинации
- Серверной фильтрации:
- о **Найти** общий поиск по полям фамилии, имени, отчества и e-mail
- о Отделы фильтр по списку отделов (множественный выбор)
- о Оклад мин / макс ограничить по мин и макс окладу

CRUD пользователей

Для пользователя реализовать CRUD:

- Создание
- Редактирование
- Удаление

Пример UI создания / редактирования пользователя:



Отчёт

Получить список отделов (плоский) со средней ЗП без паджинации. Сортировка по имени отдела.

Пример UI:



Вложенность отделов при подсчете оклада не учитывать.

Должна быть серверная фильтрация по мин. и макс. среднему окладу.

Особенности реализации

- БД Sqlite или Postgresql
- Работа с сервисом через Swagger

Приложение А. Список отделов

- Финансовый
- о **Логистики**
- о Закупок
- о Кадров
- Развлечений

Желаем успехов!

С уважением,

Rubius Team