

Тестовое задание для стажера

Frontend-разработчик в команду CorpDev

Ниже описана тестовая задача, а также условия её реализации. Условия не являются строго обязательными, если вы (разработчик) не можете соблюсти какие-либо из них, **в этом нет ничего страшного и это не означает провал**, ваше задание в любом случае проверят и дадут вам обратную связь.

Рекомендуемое время выполнения от 2 до 8 часов, в зависимости от вашего навыка.

Задача описана минимально, можно выполнить её как есть, а можно проявить инициативу (подойти к выполнению творчески) и дополнить по своему усмотрению. Четкое следование заданию не является обязательным критерием.

Рекомендуется использовать сервис тестовых REST ответов API reqres.in см [Swagger](https://swagger.io) (или подобный). В нем есть все необходимые эндпоинты для выполнения текущей задачи.

Результатом задачи должна быть ссылка на проект на GitLab, GitHub или BitBucket. Можете использовать приватный репозиторий, по вашему усмотрению.

Задача

1. Создать форму для регистрации и аутентификации пользователя;
2. Закрыть возможность любого действия, если пользователь не авторизован;
3. Сверстать таблицу или использовать готовую компоненту из UI библиотек [Ng-Zorro](#), [Material](#), [Bootstrap 4+](#) (или любой другой);
4. Наполнить таблицу данными, которые поступают через API;

5. Реализовать серверную пагинацию для таблицы;
6. Разработать страницу с кнопкой создания объекта, которая открывает форму в модальном окне. После создания объекта, необходимо отобразить его на этой же странице и показать кнопки редактирования и удаления. Редактирование должно открывать модальное окно с формой редактирования текущего объекта. Кнопка удаления удаляет объект. Все действия должны происходить без перезагрузки;
7. Создать систему оповещений об ответах сервера (на основе статуса HTTP ответа выводить сообщение для пользователя; например: “Данные получены за N секунд”, “Данные сохранены. ID: 356”, “Данные обновлены”, “Данные удалены”);

Условия

Вёрстка

1. Первая строка таблицы (заголовок) должна быть привязана и оставаться на месте при прокрутке таблицы;
2. При загрузке данных с сервера должен быть loader;
3. Оповещения должны быть в виде небольших окошек в углу окна браузера. Оповещения должны автоматически скрываться по истечению 5 секунд или при клике по ним. Оповещений может быть несколько сразу.

Код

1. Приветствуется использование фреймворков в следующем порядке: Angular (команда работает с ним), Vue, React;
2. Может быть использован любой из следующих языков: TypeScript, JavaScript;
3. Приложение должно использовать npm (либо аналогичный пакетный менеджер) для установки требуемых пакетов;
4. Приложение должно быть объектно-ориентированным;

5. Код должен быть полностью покрыт комментариями;
6. При разработке необходимо использовать Git;
7. Должен быть создан и наполнен файл README.md, с описанием проекта, инструкцией запуска;
8. Необходимо использовать любой из стандартов code style (используемый указать в README);

Будет плюсом

1. Таблица должна быть готова к использованию на мобильных устройствах (реализация не важна, на усмотрение разработчика);
2. Покрытие приложения тестами;
3. Автоматическая генерация документации;
4. Обертка приложения в docker контейнер;
5. Настройка CI (на усмотрение разработчика: GitLab, Travis и т.д.) для автоматического тестирования при коммите;
6. Деплой приложения в любое окружение любым доступным способом.