

Тестовое задание

В данном задании от вас требуется, по возможности, удовлетворить все принципы работы и условия ограничений описанные ниже. Срок выполнения - 2 дня.

Цель задания

Реализовать форму регистрации пользователя.

Описание формы

Форма регистрации включает в себя следующие поля:

- Имя (text)
- Фамилия (text)
- Номер телефона (number, невозможен ввод букв или других символов)
- Email (text, при невалидном email красная подсветка поля с сообщением под ним)
- Пол мужской/женский (select)
- Адрес (text, с подсказками при поиске используя сервис here-maps)
- Подтверждение использования данных (checkbox)
- Кнопка register

Принцип работы:

1. Пользователь заполняет форму. Все поля являются обязательными, если поле не заполнено или не валидно - оно выделяется красным цветом, и под ним появляется текст ошибки. Валидация происходит только после нажатия кнопки Register.
2. При вводе адреса появляются подсказки на основе введенного текста (см. Ограничения пункт 4).
3. После нажатия на кнопку Register - данные клиента добавляются в таблицу под формой (все хранится в state). Вид таблицы не принципиален, главное чтобы было видно все введенные данные. После чего форма очищается.

Стек технологий: React, Redux, Redux-thunk, styled-components, react-hook-forms.

Ограничения

1. Использовать готовую сборку: <https://github.com/Vlad991/frontend-test>. Не качать дополнительные библиотеки.
2. Форма должна работать с использованием библиотеки react-hook-forms, и ее возможностей. Запрещено хранить состояние формы в локальном state или redux-state.
3. Таблицу клиентов хранить в redux-state. Никаких api для хранения клиентов не использовать.
4. Для поиска адреса использовать сервис here-maps autosuggest или autocomplete: <https://developer.here.com/documentation/geocoding-search-api/api-reference-swagger.html>

Сервис предоставляет возможность ограниченного бесплатного использования.

Для запросов использовать redux-thunk.

5. У формы должен присутствовать минимальный css. Поля ввода должны быть чуть больше и шире чем стандартные стили браузера. Все стили прописывать с использованием styled-components.
6. Все компоненты должны быть функциональными (классовые запрещены). Также нельзя использовать НОС. Вместо всего этого используем hooks.
7. Форма должна быть разбита на маленькие компоненты. Каждое поле - отдельная компонента.
8. Результат работы присылать ссылкой на свой git-репозиторий.

Тестирование формы

- Написать unit-тесты для каждой компоненты используя библиотеку jest + enzyme.
- Написать unit-тесты для всех js-файлов (reducer, action, axios, thunk).
- Написать unit-тесты для всей формы используя react-testing-library.