

Тестовое задание для фронтенд-разработчика на React

Цель:

Разработать одностраничное приложение (SPA) с тремя последовательными формами для сбора данных пользователя с валидацией данных и финальным модальным окном подтверждения.

Технические требования:

1. Использовать TypeScript.
2. Использовать React.
3. Использовать React Router для переключения между формами.
4. Можно использовать Bootstrap для стилизации по желанию.
5. Использование дополнительных библиотек приветствуется, но необходимо обязательно обосновать выбор (комментарием в коде).

Описание форм:

Форма 1: Личные данные

- **Поля ввода:**
 - Телефон (input type="tel", обязательное поле, маска для ввода 0XXX XXX XXX)
 - Имя (input type="text", обязательное поле)
 - Фамилия (input type="text", обязательное поле)
 - Пол (select с опциями "Мужской", "Женский", обязательное поле)
- **Кнопка:**
 - "Далее" (переключение на следующую форму)

Форма 2: Адрес и место работы

- **Поля ввода:**
 - Место работы (select с предварительно заданными опциями, обязательное поле). В качестве источника данных для данного поля использовать https://dummyjson.com/docs/products#products-category_list (https://dummyjson.com/docs/products#products-category_list) (метод Get products category list). Предусмотреть возможность переиспользования результата полученного по API.
 - Адрес проживания (input type="text", обязательное поле)
- **Кнопки:**
 - "Назад" (возвращение на предыдущую форму)
 - "Далее" (переключение на следующую форму)

Форма 3: Параметры займа

- **Поля ввода:**
 - Ползунок для выбора суммы займа (от \$200 до \$1000, шаг 100, обязательное поле)
 - Ползунок для выбора срока займа (от 10 до 30 дней, шаг 1 день, обязательное поле)
- **Кнопки:**

- о "Назад" (возвращение на предыдущую форму)
- о "Подать заявку". При нажатии на кнопку использовать тестовый API метод <https://dummyjson.com/docs/products#products-add> (<https://dummyjson.com/docs/products#products-add>) (метод Add a new product). Отправить для примера: `{"title": firstName + ' ' + lastName}`. Затем показ модального окна.

Финальная модалка:

- После получения результата от тестового сервера, должно появиться модальное окно с текстом:

Поздравляем, <Фамилия> <Имя>. Вам одобрена <сумма> на <срок> дней.

где <Фамилия>, <Имя>, <сумма> и <срок> заменяются на данные, введенные пользователем.

Дополнительные требования:

1. Все поля обязательны для заполнения. При попытке перейти на следующую форму без заполнения всех обязательных полей должна отображаться ошибка.
2. Данные из всех форм должны сохраняться в состоянии приложения и быть доступны на любом шаге.
3. Кнопка "Назад" на второй форме должна возвращать на первую форму, сохраняя при этом введенные данные.
4. Для стилизации можно использовать Bootstrap или любую другую библиотеку на ваше усмотрение.
5. Важно придерживаться технического задания.
6. Необходимо указать время затраченное на выполнение задания.

Предоставление результата:

- Исходный код должен быть размещен на GitHub или другой платформе для совместной работы с кодом.
- В README файле должно быть указано, как запустить проект и проверить его работоспособность.