

Тестовое задание для Vue Frontend Developer

Необходимо реализовать форму управления учетными записями.
Время на выполнение задания, максимум 2 дня.

Результатом выполнения работы, является ссылка на github/gitlab репозиторий с историей коммитов.

Макет представлен для примера, главное соблюдать тз, структуру и расположение полей, визуальное отображение можно сделать по своему предпочтению.

Учетные записи



? Для указания нескольких меток для одной пары логин/пароль используйте разделитель ;

Метки	Тип записи	Логин	Пароль	
XXX	Локальная ▾	Значение	● ● ● ● ● ● ● ●	
XXX; YYYYYYYYYYYY; IIIIIIII; MMMMMMMMMM	Локальная ▾	Значение	● ● ● ● ● ● ● ●	
XXX	Локальная ▾	Значение	● ● ● ● ● ● ● ●	
Значение	LDAP ▾	Значение		
Значение	LDAP ▾	Значение		

Стек:

1. Vue.js 3 + Composition API (фреймворк)
2. TypeScript (язык)
3. Pinia (стейт менеджер)
4. Любой UI фреймворк на выбор.

Форма состоит из:

1. Заголовка и кнопки добавления учетной записи.
2. Подсказки для поля метка.
3. Списка учетных записей с лейблами полей.

Учетная запись состоит из:

1. Метка - Необязательное поле к заполнению. Максимум 50 символов. В поле вводятся текстовые метки, через знак ;
2. Тип записи - Выпадающий список. Допускаются выборы из заранее определенных вариантов:

- **LDAP** - При выборе этого типа - поле "**Пароль**" скрывается и сохраняется как null
 - **Локальная** - При выборе этого типа - поле "**Пароль**" показывается и сохраняется как введенное значение
3. **Логин** - Обязательное к заполнению. Максимум 100 символов
 4. **Пароль** - Обязательное к заполнению. Отображается в случае выбора значения "**Локальная**" в поле "**Тип записи**". Максимум 100 символов
 5. Кнопки удаления учетной записи.

Логика работы формы:

1. По нажатию кнопки "+" в конец списка учетных записей, добавляется новая пустая запись, **состоящая из: Метки, Типа записи, Логина, Пароля.**
2. При нажатии кнопки удаления, учетная запись должна полностью удалиться.
3. По окончании ввода значений в поля учетной записи (для текстовых - потеря фокуса, для селекта - изменение значения), происходит валидация обязательных полей. В случае валидности - учетная запись сохраняется/обновляется, в обратном, поля обозначаются красной обводкой.
4. Все поля сохраняются в текстовом виде, кроме поле **Метка**. Оно должно преобразовываться в массив, где один элемент это строковое значение до ;. В массиве должен лежать объект, где поле text равно одному элементу метки - ({ text: элемент метки })
5. Сохранение учетной записи должно происходить в стейт менеджер. При обновлении страницы, сохраненные учетные записи должны отображаться.