

## Требования к заданию:

### UI Task Tracker

Обязательные технологии: CSS, HTML, JS-фреймворк (React, Angular или Vue).  
Допускается использовать: Библиотеки компонентов или CSS фреймворки (Ant Design, Material UI, Twitter Bootstrap и др.), а также любые JS-библиотеки, но в адекватном количестве. Описание: Приложение должно представлять собой Single Page Application и быть UI частью клиент-серверного приложения. Данные должны храниться в формате JSON и должны быть легко заменимы на соответствующие AJAX-запросы (REST API). Данные можно хранить непосредственно в JSON-файлах и считывать fetch-запросом оттуда, либо написать какой-то простой REST API-сервер (если есть умения и время). Вся конфигурация для приложения должна быть вынесена отдельно, легко редактируема. Для тестирования работы приложения должно быть создано несколько учетных записей. Внешний вид приложения определяется разработчиком по его желанию, верстка произвольна, необходимо ее корректное отображение в основных современных браузерах. Plusом будет использование инструментов сборки.

1. Вход в приложение, авторизация Приложение должно содержать два уровня доступа.

Неавторизованному пользователю должна быть доступна только страница ввода данных (логин, пароль). После аутентификации пользователь получает доступ к своему списку задач.

2. Список задач Список задач пользователя, представленный в табличном виде, с возможностью сортировки списка, фильтрации (к примеру - по статусам) и смены представления (подробный/краткий вид/scrum доска). Автообновление списка каждые n минут.

3. Страница задачи Страница с подробной информацией о задаче - название, описание, дата, приоритет, планируемое и затраченное время, статус выполнения. Желательные требования (реализация будет плюсом):

4. Изменение задачи Возможность сменить планируемое время на выполнение задачи и ее статус, а также время, затраченное на выполнение.

5. Добавление задачи Возможность добавить новую задачу.

6. Scrum доска Таблица, представленная в виде трех колонок - План, В процессе, Готово. Задачи должны быть распределены в таблице в соответствии с их статусами и приоритетами. Перемещение задач в таблице с помощью механизма drag&drop, переход между столбцами должен сопровождаться сменой статуса у соответствующей задачи.