

Критерий	Вес
Работа в команде	
Каждый участник понимает структуру и может объяснить принцип работы любой части приложения	0,5
У команды есть репозиторий на GitHub (или аналог) и каждый участник сделал туда хотя бы один commit	2
Работоспособность	
Приложение работает в dev режиме у тимлида на компьютере	1
Приложение задеплоено и доступно по открытой ссылке	1
Идея	
Приложение соответствует идее, которую команда ставила в начале курса (написано в названии), и решает поставленные задачи.	0,5
Фронтенд	
<p>Качество верстки.</p> <p>1 балл Вёрстка соответствует figma-макету. Элементы расположены ровно, не наезжают друг на друга.</p> <p>2 балла Наличие адаптивной вёрстки, отсутствие горизонтальной полосы прокрутки при просмотре с телефона</p> <p>3 балла Наличие анимации, плавных переходов, уникальный дизайн</p>	1
<p>Команда использовала препроцессор.</p> <p>Стили прописаны в файлах препроцессора (scss или аналоги)</p>	0,5
<p>Команда использовала TypeScript, компилятор не выдает ошибок.</p> <p>Скрипты прописаны в файлах ts и tsx.</p>	1
Команда применила state менеджер (Redux, MobX или др.).	0,5
Наличие клиентского роутинга (переход с одной страницы на другую без обновления и возможность скопировать ссылку на любую страницу)	1
<p>Выполнена успешная синхронизация с сервером.</p> <p>Состояния (зависящие от сервера) на клиенте меняются только после ответа сервера.</p>	2
<p>В проекте отсутствуют необработанные исключения.</p> <p>Все ошибки как с сервера так и с клиента должны выводиться в удобном виде пользователю.</p>	2
Бэкенд	
Команда корректно использовала HTTP методы и URL (в соответствии с REST)	2
<p>Команда использовала TypeScript, компилятор не выдает ошибок.</p> <p>Скрипты прописаны в файлах ts и tsx.</p>	1
Приложение задеплоено на сервере с помощью Docker	0,5
Команда применила паттерн проектирования и ООП	0,5
В проекте есть хеширование паролей	1
Команда использовала ORM для взаимодействия с БД	1
В проекте использованы тесты	0,5