# Условия домашки

|  |
| --- |
| Чтобы выполнить основную часть домашнего задания, используйте материалы урока – в них вы найдете все необходимое для решения. |

|  |
| --- |
| 💡 В этой домашке мы продолжим разрабатывать приложение «Лента комментариев». |

## В каком виде сдавать работу

Сделайте pull request в свой репозиторий с прошлой домашкой с результатами выполнения этой домашки. На момент сдачи работы PR должен быть открытым.

После того как наставник примет работу, смержите свой пул-реквест в main-ветку.

### Документация API

[https://github.com/GlebkaF/webdev-hw-api/blob/main/pages/api/v1/[key]/comments/README.md](https://github.com/GlebkaF/webdev-hw-api/blob/main/pages/api/v1/%5Bkey%5D/comments/README.md)

## Задание

* Я, как пользователь, понимаю, что ввел слишком короткую строчку в имя или текст комментария при добавлении комментария.
* Я, как пользователь, понимаю, что у меня пропал интернет и мне нужно повторить попытку добавить комментарий позже.

### Что нужно сделать

* Обработать 400-ю ошибку в POST-запросе на добавление комментария и показать пользователю alert.
* Обработать 500-е ошибки в запросе на получение комментов и на добавление коммента.
* Обработать ошибки типа «у пользователя пропал интернет» во всех запросах.
* Форма добавления не должна сбрасываться в случае ошибки.

«Как не терять введенные данные при перерендере формы добавления?»

|  |
| --- |
| **Подсказка**  **Простой вариант:** сделайте форму статичной — просто скрывайте ее на время добавления комментария через display: block.  **Сложный вариант:** используйте событие input на элементах формы, чтобы отслеживать, что пользователь ввел в форму, и сохранить их в переменную. Во время рендера подставляйте эти данные в инпуты. |

POST-запрос будет в половине случаев отвечать 500-й ошибкой, если в теле запроса передать параметр forceError: true.

Используйте это для проверки своего кода: 

## Дополнительное задание

POST API на добавление комментария время от времени падает и отвечает 500-м кодом. В таком случае мы рекомендуем пользователю попробовать позже снова и ему приходится снова руками кликать на кнопку «Написать».

Это не очень удобно с точки зрения пользовательского опыта, ведь мы можем попробовать снова автоматически.

Реализуйте автоматический повторный запрос в API в случае 500-го ответа.

Как это должно работать, посмотрите на демо.

|  |
| --- |
| **Подсказка: псевдокод решения**  const handlePostClick = () => {  //...  postComment(text.value, name.value)  .catch((error) => {  if (error.message === "Ошибка сервера") {  handlePostClick();  }  });  //...  };  addButton.addEventListener("click", handlePostClick); |

## Условия проверки и выполнения задания

* Если в любом запросе от API придет 500-я ошибка, приложение должно показать alert «Сервер сломался, попробуй позже».
* 400 в запросе на добавление:
  + Я ввожу в поле «Имя» или «Текст» строчку короче трех символов.
  + В консоли я вижу, что выполняется запрос в API, в ответе вижу 400-й код.
  + Вижу alert с текстом: «Имя и комментарий должны быть не короче 3 символов».
  + В форме добавления остался текст, который я ввел в первом шаге.
* Нет интернета:
  + открываю приложение;
  + выключаю интернет;
  + пытаюсь добавить комментарий;
  + вижу alert: «Кажется, у вас сломался интернет, попробуйте позже».