

FIXING THE DELETE QUESTION BUTTON

If you open `Question.vue` you'll see that the *Delete question* button will be visible if the current user signed in is authorize to delete the question.

```
<button v-if="authorize('deleteQuestion', question)"
...>Delete</button>
```

Now If you see the `deleteQuestion` in the `authorization/policies.js` file. It will return `true` if two conditions are met.

1. The current user is the author of the question
2. The question that is going to be removed has no answer

```
deleteQuestion (user, question) {
  return user.id === question.user.id && question.answers_count < 1;
}
```

It's look like the `answers_count` was not change when we added or deleted any answer. So to fix this issue we need to emit an event when adding or removing any answer on `Answers` component. In the `Question` component we can listen to that event and change the `answers_count`'s value.

1. Emitting an event

Let's open the `Answers.vue` file then go to script section. Since we're going to use global event we need to import the `EventBus`.

```
import EventBus from '../event-bus'
```

We can emit an event let's say `answers-count-changed` on both `add` and `remove` methods. We also need to pass in the local `count` variable as a payload.

```
add (answer) {
  // ...
  EventBus.$emit('answers-count-changed', this.count);
},

remove (index) {
  // ...
  EventBus.$emit('answers-count-changed', this.count);
},
```

We actually need to emit the event only for the first answer added or last answer being removed. So here we can put it in if statement.

```
add (answer) {
  // ...
  if (this.count === 1) {
    EventBus.$emit('answers-count-changed', this.count);
  }
},

remove (index) {
  // ...
  if (this.count === 0) {
    EventBus.$emit('answers-count-changed', this.count);
  }
},
```

2. Listening to event

Now let's open `question.vue` file then go to script section. Import the `EventBus` at the top and listen the `answers-count-changed` event in `mounted` hook. We can then assign the `count` to `question.answers_count`.

```
// ...
import EventBus from '../event-bus';

export default {
  mounted () {
    EventBus.$on('answers-count-changed', (count) => {
      this.question.answers_count = count;
    })
  },
  // ...
}
```

3. When deletion succeed

Last, let's go to `delete` method. In that method you'll see that once deletion successful it will be redirected to `questions` page using native javascript `window.location.href` which will make a page to get reloaded.

Since we have Vue router installed in our application, now we can replace that command with this.

```
this.$router.push({ name: 'questions' });
```

So here the `delete` button look like.

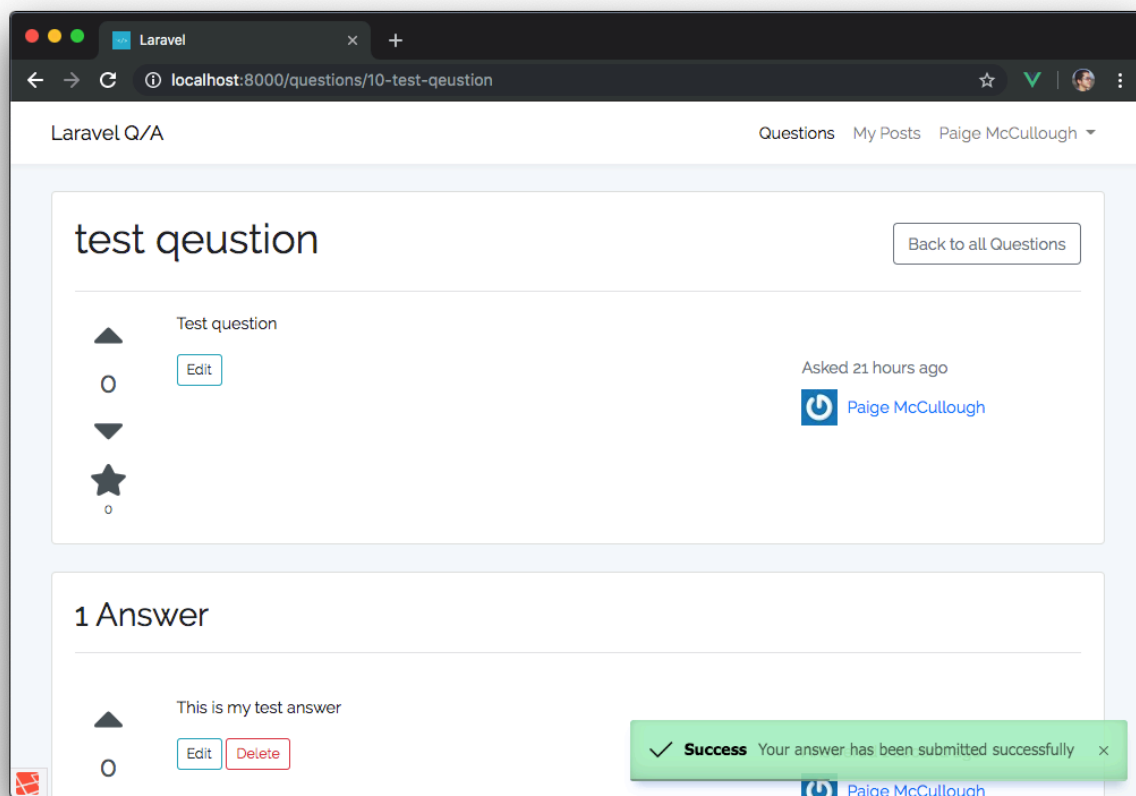
```

delete () {
  axios.delete(this.endpoint)
    .then(({data}) => {
      this.$toast.success(data.message, "Success", { timeout: 2000 });
      this.$router.push({ name: 'questions' });
    });
}

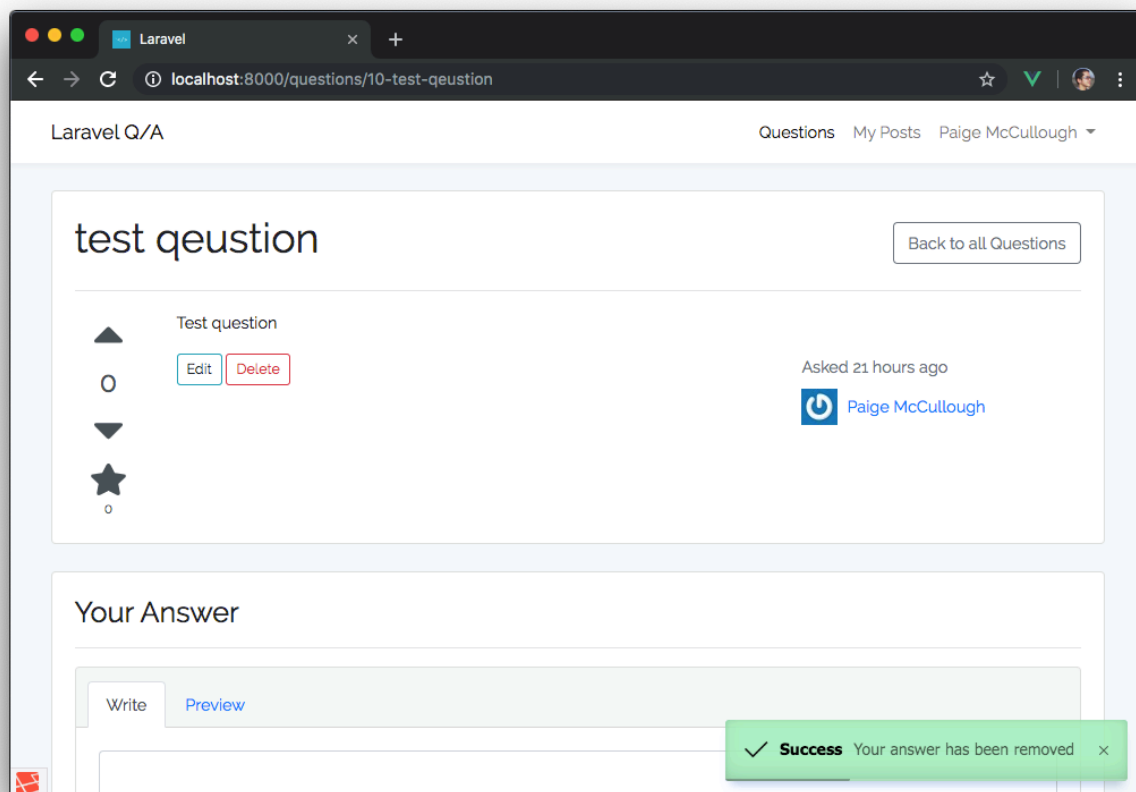
```

4. Testing the button

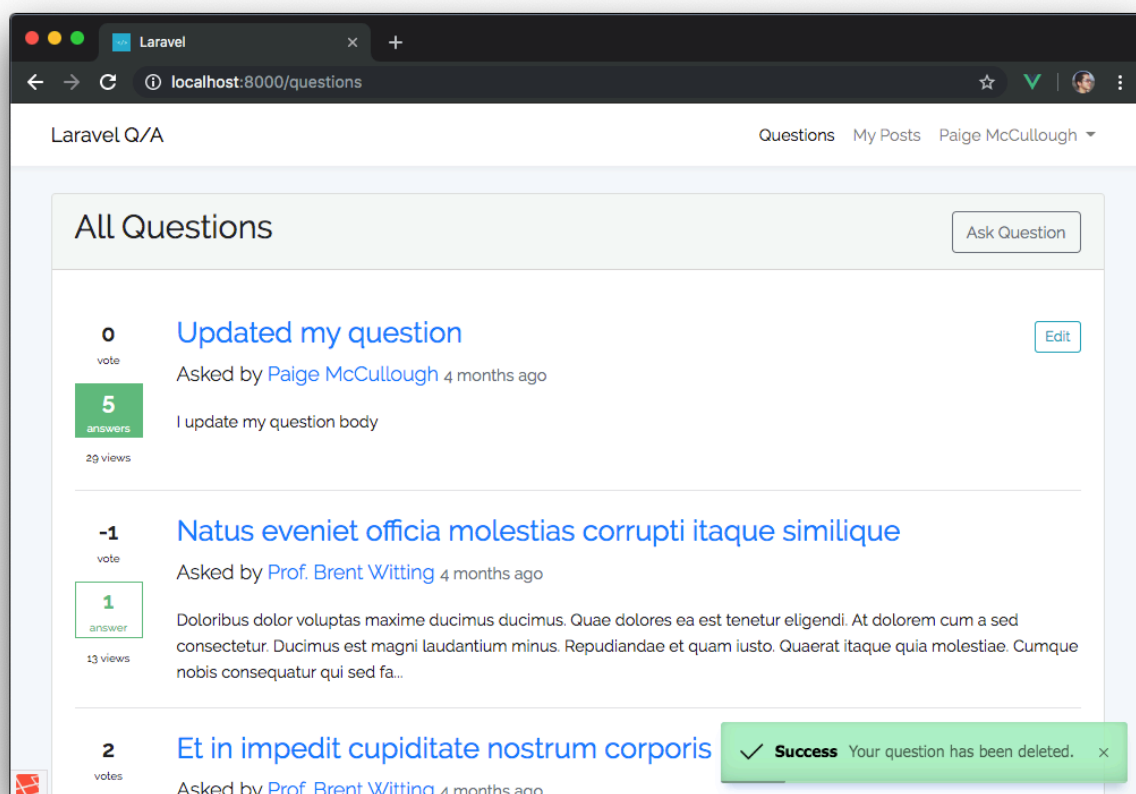
Now let's save all changes then go to the browser. Find a question which has no any answer. You'll see the Delete question still appear initially. If you add new answer you'll see the button disappear immediately.



But if you delete the answer you'll see the *Delete question* button appears immediately.



Then if you hit the *Delete* button and hit Yes to confirm. The answer successfully removed and you'll be taken to all question page.



So now we've successfully fix the issue on *Delete question* button. Let's fix another issue on *Load more answers* button.