

## Exercise (Instructions): Fetch Post Comment

### Objectives and Outcomes

In this exercise you will learn how to configure Fetch to be able to post data to the server. At the end of this exercise you will be able to:

- Configure Fetch to post data to the server
- Receive and process the response to the POST operation on the server

### Posting a Comment

- Open *ActionCreators.js* and update it as follows:

```
1  ...
2
3  export const addComment = (comment) => ({
4    type: ActionTypes.ADD_COMMENT,
5    payload: comment
6  });
7
8  export const postComment = (dishId, rating, author, comment) => (dispatch) => {
9
10     const newComment = {
11       dishId: dishId,
12       rating: rating,
13       author: author,
14       comment: comment
15     };
16     newComment.date = new Date().toISOString();
17
18     return fetch(baseUrl + 'comments', {
19       method: "POST",
20       body: JSON.stringify(newComment),
21       headers: {
22         'Content-Type': "application/json"
23       },
24       credentials: "same-origin"
25     })
26     .then(response => {
27       if (response.ok) {
28         return response;
29       } else {
30         var error = new Error('Error ' + response.status + ': ' + response
31           .statusText);
32         error.response = response;
33         throw error;
34       }
35     },
36     error => {
37       throw error;
38     })
39     .then(response => response.json())
40     .then(response => dispatch(addComment(response)))
41     .catch(error => { console.log('post comments', error.message); alert('Your
42       comment could not be posted\nError: ' + error.message); });
43   ...
44 }
```

- Open *comment.js* and **remove** the following two lines from it:

```
1  ...
2
3  comment.id = state.comments.length;
4  comment.date = new Date().toISOString();
5
6  ...
```

- Open *MainComponent.js* and update it as follows:

```
1  ...
2
3  import { postComment, fetchDishes, fetchComments, fetchPromos } from './redux
4    /ActionCreators';
5
6  ...
7  postComment: (dishId, rating, author, comment) => dispatch(postComment(dishId,
8    rating, author, comment))
9
10 ...
11
12 postComment={this.props.postComment}
13 ...
```

- Finally, open *DishdetailComponent.js* and update it as follows:

```
1  ...
2
3  function RenderComments({comments, postComment, dishId}) {
4
5    ...
6
7    <CommentForm dishId={dishId} postComment={postComment} />
8
9    ...
10
11    this.props.postComment(this.props.dishId, values.rating, values
12      .author, values.comment);
13
14    ...
15
16    postComment={props.postComment}
17  ...
```

- Save all the changes and do a Git commit with the message "Fetch Post Comment".

## Conclusions

In this exercise you learnt to use Fetch to post data to the server.

Mark as completed

