

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     . . .
12     .statusText);
31     error.response = response;
32     throw error;
33   }
34   },
35   error => {
36     throw error;
37   })
38   .then(response => response.json())
39   .then(response => dispatch(addComment(response)))
40   .catch(error => { console.log('post comments', error.message); alert('Your
    comment could not be posted\nError: '+error.message); });
41 };
42
43 . . .
```

- 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  . . .
```

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

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

