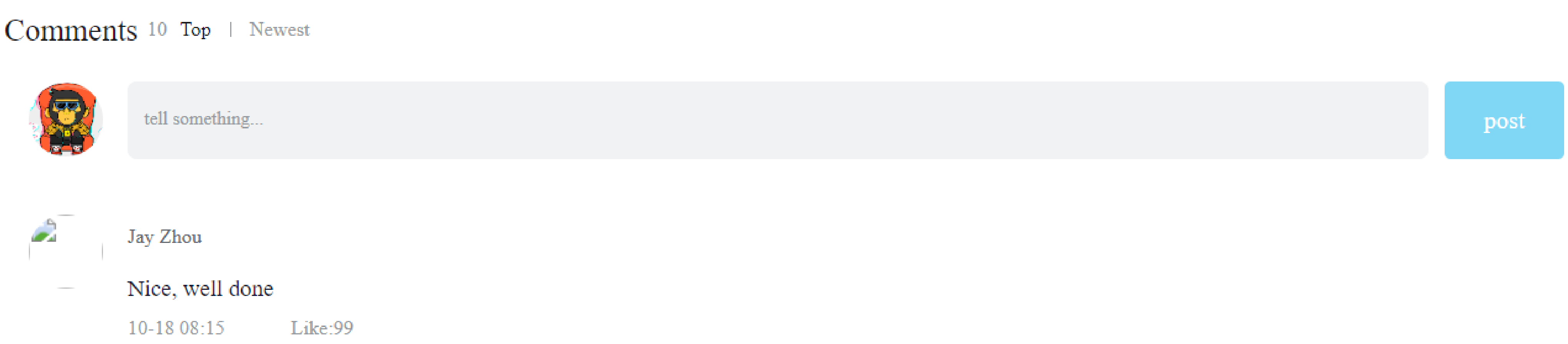
Workshop 2 - Forum

Refactor Features



Jay Zhou

Nice, well done

10-18 08:15 Like:99

1. Retrieve the list of comments using API and render them.
2. Encapsulate data retrieval logic using a custom Hook function.
3. Abstract each item in the comments into an independent component for rendering.

Feature 1: Get list from API

json-server

Get a full fake REST API with **zero coding** in **less than 30 seconds** (seriously)

- https://github.com/typicode/json-server

1. `npm install json-server -D`
2. Create a `db.json`, and put under the project root folder
3. Config script in `package.json`
   - `"serve": "json-server db.json --port 3004"`
4. Start json-server: `npm run serve`
5. Access the API: http://localhost:3004/list

Get list via useEffect

```
  useEffect(() => {
    async function getList(){
      const res = await fetch('http://localhost:3004/list');
      const data = await res.json();
      setComments(_.orderBy(data, 'like', 'desc'));
    }
    getList();
  }, []);
```

Feature 2: Custom hook

Encapsulate data retrieval logic using a custom Hook function

Ideas:

1. Define a function starts with use
2. Implement the feature in the function body
3. return state and function which will be used in the component
4. Use the function inside component

```
function useXxx(){
    //logic here
    return {
      // state
      // function
    }
  }
```

Code

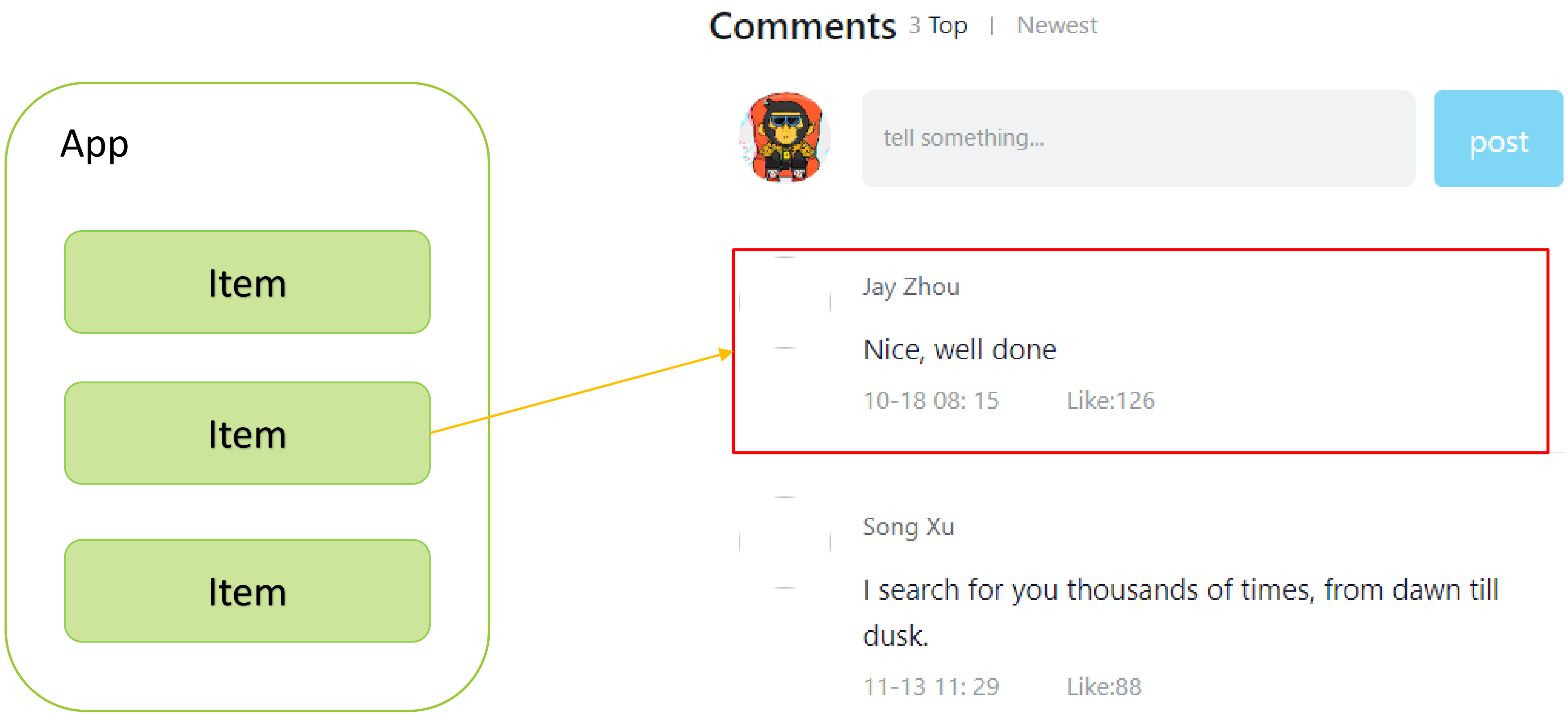
```
function useGetList(){
  const [comments, setComments] = useState<any[]>([]);

  useEffect(() => {
    async function getList() {
      const res = await fetch('http://localhost:3004/list');
      const data = await res.json();
      setComments(data);
    }
    getList();
  }, []);
  return {
    comments,
    setComments
  }
}
```

```
const {comments, setComments} = useGetList();
```

Feature 3: comment item component

Encapsulate the comment item as a component



Comments 3 Top | Newest

tell something...

post

Jay Zhou

Nice, well done

10-18 08: 15 Like:126

Song Xu

I search for you thousands of times, from dawn till dusk.

11-13 11: 29 Like:88

The App acts as the 'smart component' responsible for data retrieval, while the Item acts as the 'UI component' responsible for data rendering.

Code

```
export default function Item({user, item, onDel}: {user: any, item: any, onDel: any}){
    return (
        <div className="reply-item" key={item.rpid}>
              <div className="content-wrap">
                <div className="root-reply">
                  <div className="reply-info">
                    {
                      item.user.uid === user.uid && (
                        <span className="delete-btn" onClick={() => onDel(item.rpid)}>
                          Delete
                        </span>
                      )
                    }
                  </div>
                </div>
              </div>
            </div>
    )
}
```

```
{comments.map(item => (<Item user={user} item={item} onDel={deleteComment}/>))}
```