

A COMPARISON

SVELTE VS **REACT**

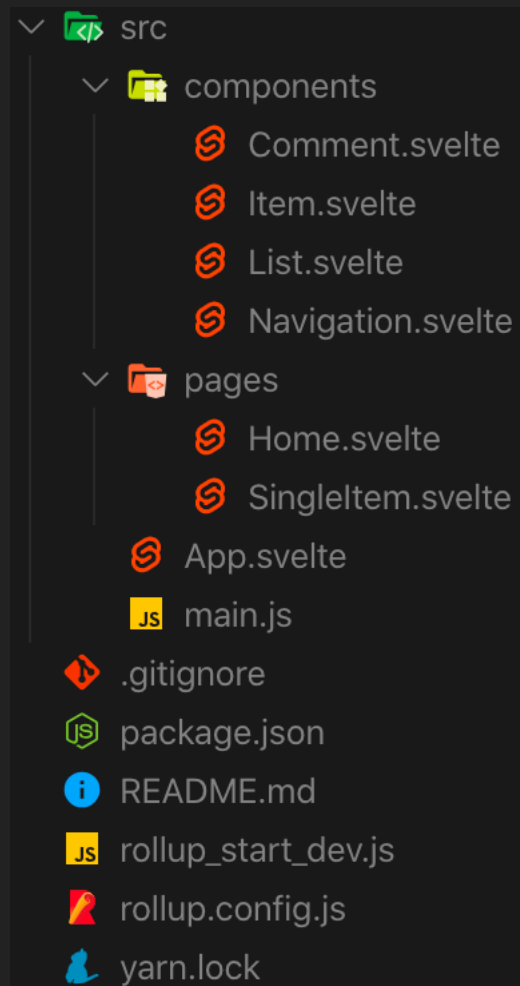
LET'S BEGIN

WHO AM I?

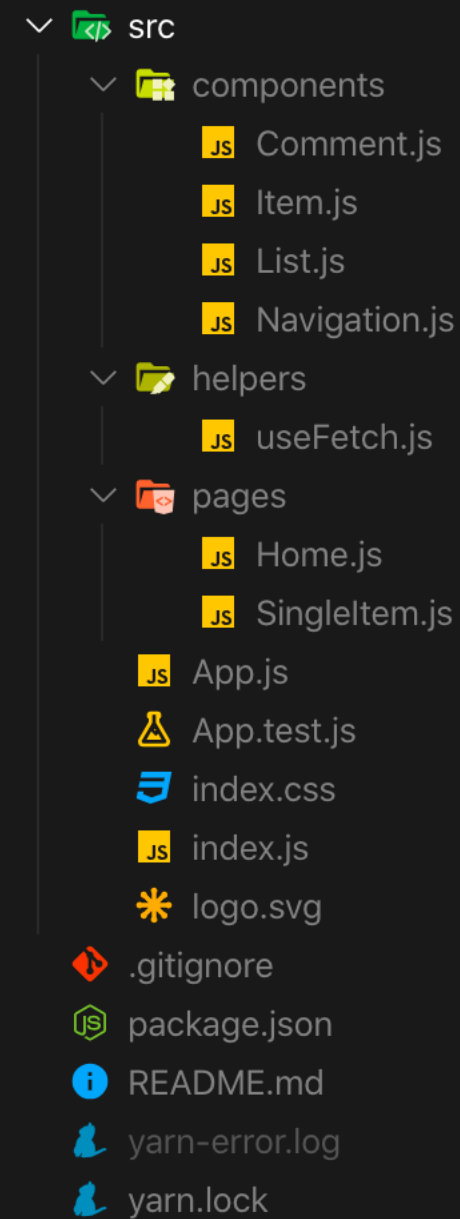
- ▶ Web Developer
 - ▶ Github (<https://github.com/kevmodrome>)
 - ▶ Twitter ([@kevmodrome](https://twitter.com/kevmodrome))
- ▶ Svelte Community contributor (<https://github.com/sveltejs/community>)
- ▶ Svelte News Podcast

THE STRUCTURE

SVELTE




REACT



THE STARTING POINT

SVELTE

src >  App.svelte > ...

```
1  <script>
2    import { Router, Route } from "svelte-routing";
3
4    import Navigation from "../components/Navigation.svelte";
5    import Home from "../pages/Home.svelte";
6    import SingleItem from "../pages/SingleItem.svelte";
7  </script>
8
9  <style>
10    div {
11      max-width: 700px;
12      margin: 0 auto;
13    }
14  </style>
15
16  <Router>
17    <Navigation />
18    <div>
19      <Route path="/item/:id" component={SingleItem} />
20      <Route path="/" component={Home} />
21    </div>
22  </Router>
23
```

REACT

src >  App.js > ...

```
1  import React from 'react';
2  import {
3    BrowserRouter as Router,
4    Switch,
5    Route
6  } from "react-router-dom";
7
8  import Navigation from '../components/Navigation'
9  import Home from '../pages/Home'
10 import SingleItem from '../pages/SingleItem'
11
12
13 function App() {
14   return (
15     <Router>
16       <Navigation />
17       <div style={{ maxWidth: '700px', margin: '0 auto' }}>
18         <Switch>
19           <Route path="/item/:id">
20             <SingleItem />
21           </Route>
22           <Route path="/">
23             <Home />
24           </Route>
25         </Switch>
26       </div>
27     </Router>
28   );
29 }
30
31 export default App;
```

ADDING A HOMESCREEN

SVELTE

src > pages >  Home.svelte > ...

```
1  <script>
2  |   import List from "../components/List.svelte";
3  </script>
4
5  <div>
6  |   <List route="https://node-hnapi.herokuapp.com/news?page=" page="1" />
7  </div>
```

REACT

src > pages >  Home.js > ...

```
1  import React from 'react'
2
3  import List from '../components/List'
4
5  const Home = () => {
6  |
7  |   return (
8  |     <List route="https://node-hnapi.herokuapp.com/news?page=" page="1" />
9  |   )
10 | }
11
12 export default Home
```

FETCHING DATA & LIST COMPONENT – REACT

REACT

src > components > **JS** List.js > ...

```
1  import React from 'react'
2  import { useFetch } from '../helpers/useFetch'
3
4  import Item from './Item'
5
6  const List = ({ route, page }) => {
7    const [loading, data] = useFetch(`${route}${page}`)
8
9    return (
10     <>
11       <div>
12         {loading ? <h2>Loading...</h2> : data.map((item) => {
13           return <Item {...item} />
14         })}
15       </div>
16     </>
17   )
18 }
19
20 export default List
```

src > helpers > **JS** useFetch.js > ...

```
1  import { useState, useEffect } from 'react'
2
3  export const useFetch = (url) => {
4    const [loading, setLoading] = useState(false)
5    const [data, setData] = useState([]);
6    useEffect(() => {
7      (async () => {
8        setLoading(true)
9        const res = await fetch(url);
10       const data = await res.json();
11       setData(data);
12       setLoading(false)
13     })();
14   }, [url]);
15   return [loading, data];
16 };
```

FETCHING DATA & LIST COMPONENT – SVELTE

SVELTE

src > components >  List.svelte > ...


```
1  <script>
2    import Item from "../Item.svelte";
3
4    export let route, page;
5
6    let promise = fetch(`${route}${page}`).then(res => res.json());
7  </script>
8
9  {#await promise}
10   <h2>Loading...</h2>
11  {:then data}
12    {#each data as headline, i}
13      <Item {...headline} index={i} />
14    {/each}
15  {/await}
```


DISPLAYING THE ITEMS

SVELTE

```
src > components >  Item.svelte > ...
1  <script>
2    import { fade } from "svelte/transition";
3    export let id, title, points, user, time_ago, comments_count, url, index;
4
5    import { Link } from "svelte-routing";
6  </script>
7
8  <style>
9    article {
10     padding: 5px 10px;
11     margin: 15px;
12     background: rgb(248, 248, 248);
13     border-radius: 10px;
14   }
15 </style>
16
17 <article>
18   <h2>
19     <a href={url}>{title}</a>
20   </h2>
21   <p>
22     {points} points by {user} {time_ago}
23     <Link to={`/item/${id}`}>{comments_count} comments</Link>
24   </p>
25 </article>
```

REACT

```
src > components >  Item.js > ...
1  import React from 'react'
2  import { Link } from "react-router-dom";
3
4  const Item = ({ id, title, points, user, time_ago, comments_count, url }) => {
5    return (
6      <article style={{ padding: '5px 10px', margin: '15px', background: 'rgb(248, 248 ,248)', borderRadius: '10px' }}>
7        <h2><a href={url}>{title}</a></h2>
8        <p>{points} points by {user} {time_ago} <Link to={`/item/${id}`}>{comments_count} comments</Link></p>
9      </article>
10    )
11  }
12
13  export default Item
```

SHOWING COMMENTS - 1

SVELTE

src > pages > SingleItem.svelte > ...

```
1  <script>
2    import Comment from "../components/Comment.svelte";
3
4    export let id;
5
6    let promise = fetch(`http://node-hnapi.herokuapp.com/item/${id}`).then(res =>
7      res.json()
8    );
9  </script>
10
11  {#await promise}
12    <h2>Loading...</h2>
13  {:then data}
14    <h2>{data.title}</h2>
15    <p>submitted by {data.user} {data.time_ago}</p>
16    <Comment comment={data} />
17  {/await}
```

REACT

src > pages > SingleItem.js > ...

```
1  import React from 'react'
2  import { useParams } from "react-router-dom";
3  import { useFetch } from '../helpers/useFetch'
4
5  import Comment from '../components/Comment'
6
7  const SingleItem = () => {
8    let { id } = useParams()
9    const [loading, data] = useFetch(`http://node-hnapi.herokuapp.com/item/${id}`)
10
11
12
13    return (
14      <div>
15        {loading ? <h1>Loading comments...</h1> :
16          <>
17            <h1>{data.title}</h1>
18            <p>submitted by {data.user} {data.time_ago}</p>
19            <Comment comment={data} />
20          </>
21        </div>
22      )
23    }
24
25    export default SingleItem
```

SHOWING COMMENTS – REACT

REACT

src > components >  Comment.js > ...

```
1  import React from 'react'
2
3  const Comment = ({ comment }) => {
4    const nestedComments = (comment.comments || []).map(comment => {
5      return <Comment key={comment.id} comment={comment} type="child" />
6    })
7
8    return (
9      comment.title ? nestedComments : <div style={{ "marginLeft": "25px", "marginTop": "20px" }}>
10        <p style={{ color: 'grey', fontSize: '0.9em' }}>{comment.user} {comment.time_ago}</p>
11        <div style={{ borderBottom: '1px solid lightgrey' }} dangerouslySetInnerHTML={{ __html: comment.content }} />
12        {nestedComments}
13      </div>
14    )
15  }
16
17  export default Comment
```

SHOWING COMMENTS – SVELTE

SVELTE

```
src > components >  Comment.svelte > ...
1  <script>
2  |   export let comment;
3  </script>
4
5  <style>
6  |   .container {
7  |     margin-left: 25px;
8  |     margin-top: 20px;
9  |   }
10 |   p {
11 |     color: grey;
12 |     font-size: 0.9em;
13 |   }
14 |   .html {
15 |     border-bottom: 1px solid lightgrey;
16 |   }
17 </style>
18
19 {#each comment.comments as comment}
20 |   <div class="container">
21 |     <p>{comment.user} {comment.time_ago}</p>
22 |     <div class="html">
23 |       {@html comment.content}
24 |     </div>
25 |     <svelte:self {comment} />
26 |   </div>
27 {/each}
```

WHAT'S THE END RESULT?

- ▶ Smaller bundle size
- ▶ Less code

SVELTE



Performance

Metrics			
● First Contentful Paint	0.9 s	● First Meaningful Paint	0.9 s
● Speed Index	1.1 s	● First CPU Idle	0.9 s
● Time to Interactive	1.1 s	● Max Potential First Input Delay	60 ms

View Trace

REACT



Performance

Metrics			
● First Contentful Paint	1.9 s	● First Meaningful Paint	1.9 s
● Speed Index	1.9 s	● First CPU Idle	2.8 s
● Time to Interactive	2.8 s	● Max Potential First Input Delay	90 ms

THE END