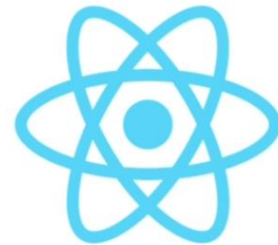


React JS

WHAT IS REACT?

A JavaScript library / framework created and maintained by Facebook that is used for building user interfaces

React gives us a way to build websites & UIs with organized and reusable components



UI COMPONENTS

OUTPUT / HTML (JSX)

LOGIC (JS)

STYLE (CSS)

Optional: Can also be separate

```
<App />
```

```
<MenuBar />
```

```
<MainPage />
```

```
<ListItem />
```

```
<ListItem />
```

```
<ListItem />
```

```
<SiteFooter />
```

React JS

WHAT YOU SHOULD KNOW

- ✓ JavaScript Basics - Data structures, loops, functions, etc
- ✓ The DOM - Document Object Model
- ✓ High Order Array Methods - forEach, map, filter, reduce
- ✓ Arrow Functions
- ✓ Async Programming & Fetch API
- ✓ NPM (Node Package Manager)

WHY LEARN REACT?

- ✓ Organization
- ✓ Reusable
- ✓ Flexibility
- ✓ Popularity & Support
- ✓ Performance



React JS

DECLARATIVE CODE

React is very declarative.

We can picture the result of this code by the declarative elements

```
<App>
  <Header title='Employee Manager' />
  <EmployeeSearch />
  <EmployeeList employees={employees} />
  <Footer />
</App>
```

Components

Props

State

Events

3. Environment Setup

4. Code Repos

React JS

2. React Basics & JSX

1. Feedback Project Intro

2. Create React App

`npx create-react-app Keep-Note --use-npm`

```
C:\Users\Dell\Desktop\Sudip>npx create-react-app keep-note --use-npm

Creating a new React app in C:\Users\Dell\Desktop\Sudip\keep-note.

Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts with cra-template...

added 1394 packages in 6m

210 packages are looking for funding
  run `npm fund` for details

Initialized a git repository.

Installing template dependencies using npm...

added 56 packages in 31s

210 packages are looking for funding
  run `npm fund` for details
Removing template package using npm...

removed 1 package, and audited 1450 packages in 10s

210 packages are looking for funding
  run `npm fund` for details

6 high severity vulnerabilities

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.

Created git commit.

Success! Created keep-note at C:\Users\Dell\Desktop\Sudip\keep-note
```

React JS

Success! Created keep-note at C:\Users\Dell\Desktop\Sudip\keep-note
Inside that directory, you can run several commands:

```
npm start
  Starts the development server.

npm run build
  Bundles the app into static files for production.

npm test
  Starts the test runner.

npm run eject
  Removes this tool and copies build dependencies, configuration files
  and scripts into the app directory. If you do this, you can't go back!
```

We suggest that you begin by typing:

```
cd keep-note
npm start
```

Happy hacking!

```
C:\Users\Dell\Desktop\Sudip\keep-note>code .
C:\Users\Dell\Desktop\Sudip\keep-note>
```

You can now view keep-note in the browser.

| | |
|------------------|--------------------------|
| Local: | http://localhost:3000 |
| On Your Network: | http://192.168.1.68:3000 |

Note that the development build is not optimized.
To create a production build, use `npm run build`.

webpack compiled **successfully**

React JS

3. Initializing React

```
JS index.js M X JS App.js U # index.css M
src > JS index.js
1
2 import React from 'react'
3 import ReactDOM from 'react-dom'
4 import './index.css'
5 import App from './App'
6
7 ReactDOM.render(
8   <React.StrictMode>
9   <App />
10 </React.StrictMode>
11 , document.getElementById('root'))

src > JS App.js > [e] default
1 function App(){
2   return <h1> Hello form the app component</h1>
3
4 }
5
6 export default App
```

Hello form the app component

4. Intro To JSX

React JS

src > JS App.js > ...

```
1  function App(){
2    return (
3      <div className="container">
4        <h1> My App</h1>
5
6      </div >
7    )
8  }
9
10 export default App
11
12
13
14 // import React from 'react'
15
16 // function App(){
17 //   return React.createElement(
18 //     'div',
19 //     {className: 'container'},
20 //     React.createElement('h1', {}, 'My App')
21 //   )
22 // }
```

JS index.js

```
import React from 'react'
import ReactDOM from 'react-dom'
import './index.css'
import App from './App'

ReactDOM.render(
  <React.StrictMode>
  <App />
</React.StrictMode>
, document.getElementById('root'))
```

React JS

5. Dynamic Values & Lists in JSX

```
JS App.js U X # index.css M
src > JS App.js > App > comments.map() callback
13 function App(){
14   const title = 'Blog Post'
15   const body = 'This is my blog post'
16   const comments = [
17     {id:1, text: 'Comment one'},
18     {id:2, text: 'Comment two'},
19     {id:3, text: 'Comment three'},
20   ]
21   return (
22     <div className="container">
23       <h1> {title.toUpperCase()}</h1>
24       <p> {body}</p>
25
26       <div className="comments">
27         <h3> Comments ({comments.length})</h3>
28         <ul>
29           {comments.map((comment, index) => (
30             <li key={index}>{comment.text}</li>
31           ))}
32         </ul>
33       </div>
34     </div >
35   )
36 }
37
38 export default App
20
```

BLOG POST

This is my blog post

Comments (3)

- Comment one
- Comment two
- Comment three

React JS

6. Conditionals in JSX

```
src > JS App.js > App > commentBlock
21
22   const loading = false
23   const showComments = true
24
25   if(loading) return <h1> Loading...</h1>
26
27   const commentBlock = (
28     <div className="comments">
29       <h3> Comments ({comments.length})</h3>
30       <ul>
31         {comments.map((comment, index) => (
32           <li key={index}>{comment.text}</li>
33         ))}
34       </ul>
35     </div>
36   )
37   return (
38     <div className="container">
39       <h1> {title.toUpperCase()}</h1>
40       <p> {body}</p>
41
42       {showComments && commentBlock}
43
44       </div >
45   )
46 }
47
48
49 export default App
```

BLOG POST

This is my blog post

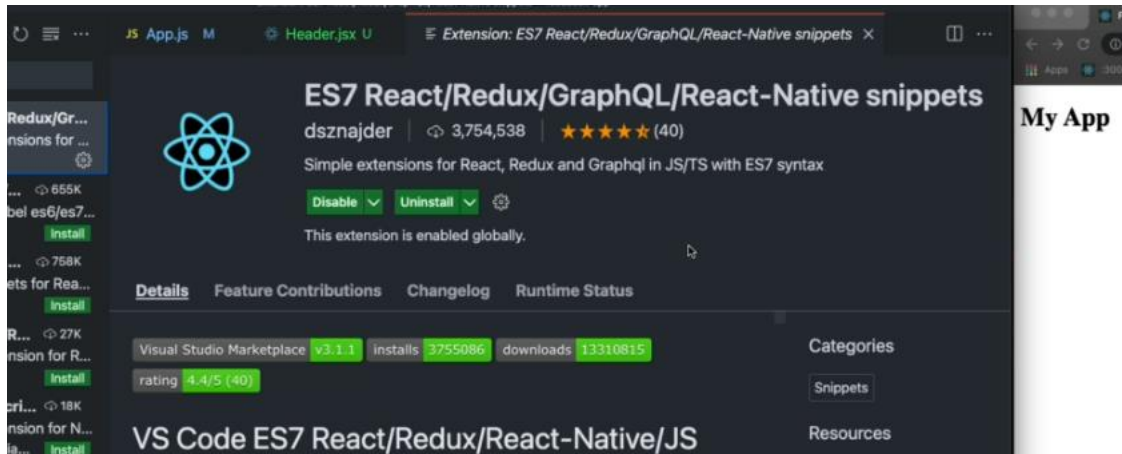
Comments (3)

- Comment one
- Comment two
- Comment three

React JS

3. Components, Props & State

1. Creating Your First Component & Props



rfce

```
JS App.js U # index.css M JS Header.jsx U ●
src > components > JS Header.jsx > header
1 import React from 'react'
2
3 function header() {
4   return (
5     <div>header</div>
6   )
7 }
8
9 export default header
```

Activate Windows

React JS

```
JS App.js U X JS Header.jsx U
src > JS App.js > [🔗] default
56 import Header from "../components/Header"
57
58 function App(){
59   return (
60     <>
61     <Header/>
62     </>
63   )
64 }
65 export default App
```

```
JS App.js U JS Header.jsx U X
src > components > JS Header.jsx > [🔗] default
1 import PropTypes from 'prop-types'
2
3 function header({ text }) {
4   return (
5     <header>
6       <div className="container">
7         <h2>{text}</h2>
8
9       </div>
10
11     </header>
12   )
13 }
14
15 header.defaultProps = {
16   text: 'Feedback UI',
17 }
18
19 header.propTypes = {
20   text: PropTypes.string
21 }
22
23 export default header
```

React JS

2. Adding Styles To A Component

```
src > JS App.js > App
57
58 function App(){
59   return (
60     <>
61     <Header bgColor = 'red' textColor='blue'/>
62     <div className="container">
63       <h1>Keep Notes</h1>
64     </div>
65     </>
66   )
67 }
68 export default App
69
```

```
JS App.js U JS Header.jsx U X
src > components > JS Header.jsx > header > constructor > headerStyles
1 import PropTypes from 'prop-types'
2
3 function header({ text, bgColor, textColor }) {
4
5   const headerStyles = {
6     backgroundColor: bgColor,
7     color: textColor,
8   }
9
10  return (
11    <header style={headerStyles}>
12    <div className="container"> ...
13  </div>
14  </header>
15  )
16 }
17
18
19
```

React JS

```
JS App.js U JS Header.jsx U X
src > components > JS Header.jsx > ...
4
5     const headerStyles = {
6         |     backgroundColor: bgColor,
7         |     color: textColor,
8     }
9
10    return (
11        <header style={headerStyles}>
12    > <div className="container"> ..
13
14    </div>
15
16    </header>
17
18    )
19 }
20
21 header.defaultProps = {
22     text: 'Feedback UI',
23     bgColor: 'rgba(0,0,0,0.4)',
24     textColor: '#ff6a95'
25 }
26
27 Headers.propTypes = {
28     text: PropTypes.string,
29     bgColor: PropTypes.string,
30     textColor: PropTypes.string,
31 }
32
33 export default header
```

```
JS App.js U X JS Header.jsx U
src > JS App.js > App
55
56 import Header from "../components/Header"
57
58 function App(){
59     return (
60         <>
61         <Header />
62         <div className="container">
63             <h1>Keep Notes</h1>
64         </div>
65         </>
66     )
67 }
68 export default App
69
70
```

React JS

3. State & useState Hook

```
JS App.js U    JS FeedbackItem.jsx U X
src > components > JS FeedbackItem.jsx > FeedbackItem
1  import {useState} from 'react'
2
3  function FeedbackItem() {
4      const [rating, setRating] = useState(7)
5      const [text, setText] = useState('This is an example of feedback item')
6      const handleClick = () =>{
7          setRating((prev) => {
8              console.log(prev)
9              return prev +1
10         })
11     }
12     return (
13         <div className="card">
14             <div className="num-display">{rating}</div>
15             <div className="text-display">{text} </div>
16             <button onClick={handleClick}>Click</button>
17         </div>
18     )
19 }
20
21 export default FeedbackItem
```

```
src > JS App.js > default
53
54 import Header from "../components/Header"
55 import FeedbackItem from "../components/FeedbackItem"
56
57 function App(){
58     return (
59         <>
60             <Header />
61             <div className="container">
62                 <FeedbackItem/>
63             </div>
64         </>
65     )
66 }
67 export default App
68
```

Feedback UI

18

This is an example of feedback item

Click

React JS

4. Managing Global State

src > components > JS FeedbackItem.jsx > FeedbackItem

```
12   function FeedbackItem({item}) {
13     return (
14       <div className="card">
15         <div className="num-display">{item.rating}</div>
16         <div className="text-display">{item.text} </div>
17         { /* <button onClick={handleClick}>Click</button> */ }
18       </div>
19     )
20   }
21
22   export default FeedbackItem
```

src > JS App.js > ...

```
54
55   import { useState } from 'react'
56   import Header from "../components/Header"
57   import FeedbackData from '../data/FeedbackData'
58   import FeedbackList from '../components/FeedbackList'
59   function App(){
60
61     const [feedback, setFeedback] = useState(FeedbackData)
62     return (
63       <>
64         <Header />
65         <div className="container">
66           <FeedbackList feedback={feedback}/>
67
68         </div>
69       </>
70     )
71   }
72   export default App
73
```

React JS

```
src > components > JS FeedbackList.jsx > ...
1  import FeedbackItem from "../FeedbackItem"
2
3  function FeedbackList({feedback}) {
4    if(!feedback || feedback.length === 0){
5      return <p> No Feedback Yet</p>
6    }
7    return (
8      <div className="feedback-list">
9        {feedback.map((item) => (
10         <FeedbackItem key= {item.id} item = {item}/>
11       ))}
12      </div>
13    )
14  }
15
16  export default FeedbackList
```

5. Card Component & Conditional Styles

```
FeedbackList.jsx > Card.jsx > App.js > FeedbackItem
src > components > shared > JS Card.jsx > Card
1
2  function Card({children, reverse}) {
3    return (
4      <div className={`card ${reverse && 'reverse'}}>
5        {children}
6      </div>
7    )
8  }
```

```
src > components > JS FeedbackItem.jsx > FeedbackItem
```

```
12  import PropTypes from 'prop-types'
```


React JS

src > components > JS FeedbackList.jsx > FeedbackList

```
18 FeedbackList.propTypes = {
19   feedback: PropTypes.arrayOf(
20     PropTypes.shape({
21       id: PropTypes.number.isRequired,
22       text: PropTypes.string.isRequired,
23       rating: PropTypes.number.isRequired,
24     })
25   ),
26 }
```

src > components > shared > JS Card.jsx > Card

```
11 import PropTypes from 'prop-types'
12 function Card({children, reverse}) {
13   return (
14     <div
15       className= 'card'
16       style={{
17         backgroundColor: reverse ? 'rgba(2, 2, 42, 0.9)' : '#fff',
18         color: reverse ? '#fff' : '#000',
19       }}
20     >
21       {children}
22     </div>
23   )
24 }
25
26 Card.defaultProps = {
27   reverse: true,
28 }
29
30 Card.propTypes = {
31   children: PropTypes.node.isRequired,
32   reverse: PropTypes.bool,
33 }
34
35 export default Card
```

React JS

6. Events & Prop Drilling

src > JS App.js > App > deleteFeedback

```
60 function App(){
61
62     const [feedback, setFeedback] = useState(FeedbackData)
63
64     const deleteFeedback = (id) => {
65         console.log(Object);
66     }
67
```

JS App.js U X JS FeedbackList.jsx U JS FeedbackItem.jsx U JS Card.jsx U

src > JS App.js > App > deleteFeedback

```
55 import { useState } from 'react'
56 import Header from "../components/Header"
57 import FeedbackData from '../data/FeedbackData'
58 import FeedbackList from '../components/FeedbackList'
59 import Card from "../components/shared/Card"
60 function App(){
61
62     const [feedback, setFeedback] = useState(FeedbackData)
63
64     const deleteFeedback = (id) => {
65         if(window.confirm('Are you sure you want to delete?')){
66             setFeedback(feedback.filter((item) => item.id !== id))
67         }
68     }
69
70
71     return (
72         <>
73         <Header />
74         <div className="container">
75             <FeedbackList feedback={feedback}
76                 handleDelete = {deleteFeedback}
77             />
78             <Card>
79                 Hello
80             </Card>
81
82         </div>

```

React JS

```
JS App.js U    JS FeedbackList.jsx U    JS FeedbackItem.jsx U    JS Card.jsx U X
src > components > shared > JS Card.jsx > Card
10
11 import PropTypes from 'prop-types'
12 function Card({children, reverse}) {
13   return (
14     <div
15       className= 'card'
16       style={{
17         backgroundColor:reverse ? 'rgba(2, 2, 42, 0.9)' : '#fff',
18         color: reverse ? '#fff' : '#000',
19       }}
20     >
21       {children}
22     </div>
23   )
24 }
25
26 Card.defaultProps = {
27   reverse:true,
28 }
29
30 Card.propTypes = {
31   children: PropTypes.node.isRequired,
32   reverse: PropTypes.bool,
33 }
34
35 export default Card
```

React JS

JS App.js U

JS FeedbackList.jsx U

JS FeedbackItem.jsx U X

JS Card.jsx U

src > components > JS FeedbackItem.jsx > FeedbackItem > constructor

```
12 import {FaTimes} from 'react-icons/fa'
13 import PropTypes from 'prop-types'
14 import Card from './shared/Card'
15     function FeedbackItem({item, handleDelete}) {
16
17
18     return (
19         <Card>
20         <div className="num-display">{item.rating}</div>
21         <button onClick={() => handleDelete(item.id)} className='close'>
22             <FaTimes color='purple' />
23         </button>
24         <div className="text-display">{item.text} </div>
25         { /* <button onClick={handleClick}>Click</button> */ }
26         </Card>
27     )
28 }
29
30 FeedbackItem.propTypes = {
31     item: PropTypes.object.isRequired
32 }
33 export default FeedbackItem
```

React JS

JS App.js U

JS FeedbackList.jsx U X

JS FeedbackItem.jsx U

src > components > JS FeedbackList.jsx > FeedbackList > constructor

```
2 import PropTypes from 'prop-types'
3 import FeedbackItem from "../FeedbackItem"
4
5 function FeedbackList({feedback, handleDelete}) {
6   if(!feedback || feedback.length ===0){
7     return <p> No Feedback Yet</p>
8   }
9   return (
10     <div className="feedback-list">
11       {feedback.map((item) => (
12         <FeedbackItem
13           key= {item.id}
14           item = {item}
15           handleDelete={handleDelete}
16         />
17       ))}
18     </div>
19   )
20 }
21
22 FeedbackList.propTypes = {
23   feedback: PropTypes.arrayOf(
24     PropTypes.shape({
25       id: PropTypes.number.isRequired,
26       text:PropTypes.string.isRequired,
27       rating: PropTypes.number.isRequired,
28     })
29   ),
30 }
```

React JS

7. FeedbackStats Component & Reactivity

```
JS App.js U    JS FeedbackStats.jsx U X
src > components > JS FeedbackStats.jsx > FeedbackStats
1  import PropTypes from 'prop-types'
2
3  function FeedbackStats({feedback}) {
4    // Calculate rating Average
5
6    let average = feedback.reduce((acc, cur) =>{
7      |   return acc +cur.rating
8    },0) / feedback.length
9
10   average = average.toFixed(1).replace(/[,]\d$/, ' ')
11
12   return (
13     <div className="feedback-stats">
14       <h4>{feedback.length} Reviews</h4>
15       <h4> Average Rating: {isNaN(average) ? 0 : average}</h4>
16     </div>
17   )
18 }
19
20 FeedbackStats.propTypes = {
21   feedback: PropTypes.array.isRequired,
22 }
23
24 export default FeedbackStats
```

React JS

4. Forms, Validation & Simple Animation

1. Form Input & State

```
JS App.js U    JS FeedbackStats.jsx U    JS FeedbackForm.jsx U ●
src > components > JS FeedbackForm.jsx > FeedbackForm
1  import { useState } from "react"
2  import Card from "../shared/Card"
3  function FeedbackForm() {
4
5      const [text, setText] = useState('')
6      const handleTextChange= (e) => {
7          |   setText(e.target.value)
8      }
9      return (
10         <Card>
11         <form>
12             <h2>How Would you rate your service with us?</h2>
13             { /* @todo - rating select component */ }
14             <div className="input-group">
15                 <input
16                     onChange={handleTextChange}
17                     type= 'text'
18                     placeholder="Write a review"
19                     value = {text}
20                 />
21                 <button type="submit">Send</button>
22             </div>
23         </form>
24         </Card>
25     )
26 }
27 export default FeedbackForm
```

React JS

2. Custom Button Component

```
# index.css M    JS Button.jsx U X
src > components > shared > JS Button.jsx > Button
1  import PropTypes from 'prop-types'
2  function Button({children, version, type, isDisabled}) {
3    |   return (
4    |     <button type={type} disabled={isDisabled} className={`btn
5    |     btn-${version}`}>
6    |     {children}
7    |     </button>
8    |   )
9    | }
10
11  Button.defaultProps = {
12    |    version: 'primary',
13    |    type: 'button',
14    |    isDisabled: false,
15    |  }
16
17  Button.propTypes = {
18    |    children: PropTypes.node.isRequired,
19    |    version: PropTypes.string,
20    |    type: PropTypes.string,
21    |    isDisabled: PropTypes.bool,
22    |  }
23  export default Button
```


React JS

3. Real-Time Validation

JS FeedbackForm.jsx U X

src > components > JS FeedbackForm.jsx > FeedbackForm > handleTextChange

```
1  import { useState } from "react"
2  import Card from "../shared/Card"
3  import Button from "../shared/Button"
4  function FeedbackForm() {
5
6      const [text, setText] = useState('')
7      const [btnDisabled, setBtnDisabled] = useState(true)
8      const [message, setMessage] =useState('')
9      const handleTextChange= (e) => {
10         if(text === ''){
11             setBtnDisabled(true)
12             setMessage(null)
13         } else if(text !== '' && text.trim().length <= 10){
14             setMessage('Text must be at least 10 characters')
15             setBtnDisabled(true)
16         } else {
17             setMessage(null)
18             setBtnDisabled(false)
19         }
20         setText(e.target.value)
21     }
22     return (
```

React JS

4. Rating Select Component

JS FeedbackForm.jsx U

JS RatingSelect.jsx U X

src > components > JS RatingSelect.jsx > RatingSelect

```
1  import {useState} from 'react'
2  function RatingSelect({select}) {
3
4      const [selected, setSelected] =useState(10);
5      const handleChange = (e) =>{
6          setSelected(+e.currentTarget.value)
7          select(+e.currentTarget.value)
8      }
9      return (
10         <ul className='rating'>
11             <li>
12                 <input
13                     type='radio'
14                     id='num1'
15                     name='rating'
16                     value='1'
17                     onChange={handleChange}
18                     checked={selected ===1}
19                 />
20                 <label htmlFor='num1'>1</label>
21             </li>
```

JS FeedbackForm.jsx U

JS RatingSelect.jsx U X

src > components > JS RatingSelect.jsx > RatingSelect

```
23
24     <li>
25         <input
26             type='radio'
27             id='num2'
28             name='rating'
29             value='2'
30             onChange={handleChange}
31             checked={selected ===2}
32         />
33         <label htmlFor='num2'>2</label>
34     </li>
35
36     <li>
37         <input
38             type='radio'
39             id='num3'
40             name='rating'
41             value='3'
42             onChange={handleChange}
43             checked={selected ===3}
44         />
45         <label htmlFor='num3'>3</label>
46     </li>
```

React JS

JS FeedbackForm.jsx U

JS RatingSelect.jsx U X

src > components > JS RatingSelect.jsx > RatingSelect

```
48     <li>
49         <input
50             type='radio'
51             id='num4'
52             name='rating'
53             value='4'
54             onChange={handleChange}
55             checked={selected ===4}
56         />
57         <label htmlFor='num4'>4</label>
58     </li>
59
60     <li>
61         <input
62             type='radio'
63             id='num5'
64             name='rating'
65             value='5'
66             onChange={handleChange}
67             checked={selected ===5}
68         />
69         <label htmlFor='num5'>5</label>
70     </li>
71     <li>
72         <input
73             type = 'radio'
74             id='num6'
75             name='rating'
76             value ='6'
```

React JS

JS FeedbackForm.jsx U

JS RatingSelect.jsx U X

src > components > JS RatingSelect.jsx > RatingSelect

```
76     value = '6'
77     onChange={handleChange}
78     checked={selected === 6}
79   />
80   <label htmlFor='num6'>6</label>
81 </li>
82
83   <li>
84     <input
85       type = 'radio'
86       id='num7'
87       name='rating'
88       value = '7'
89       onChange={handleChange}
90       checked={selected === 7}
91     />
92     <label htmlFor='num7'>7</label>
93   </li>
94
95   <li>
96     <input
97       type = 'radio'
98       id='num8'
99       name='rating'
100      value = '8'
101      onChange={handleChange}
102      checked={selected === 8}
103    />
104    <label htmlFor='num8'>8</label>
```

React JS

JS FeedbackForm.jsx U

JS RatingSelect.jsx U X

src > components > JS RatingSelect.jsx > RatingSelect

```
101     onChange={handleChange}
102     checked={selected === 8}
103   />
104   <label htmlFor='num8'>8</label>
105 </li>
106
107   <li>
108     <input
109       type = 'radio'
110       id='num10'
111       name='rating'
112       value = '10'
113       onChange={handleChange}
114       checked={selected === 10}
115     />
116     <label htmlFor='num10'>10</label>
117   </li>
118 </ul>
119
120
121 )
122 }
123
124 export default RatingSelect
```

React JS

JS FeedbackForm.jsx U X

JS RatingSelect.jsx U

src > components > JS FeedbackForm.jsx > FeedbackForm

```
24   return (  
25     <Card>  
26       <form>  
27         <h2>How Would you rate your service with us?</h2>  
28         { /* @todo - rating select component */ }  
29         <RatingSelect select= {(rating) => setRating(rating)} />  
30  
31  
32         <div className="input-group">  
33           <input  
34             onChange={handleTextChange}  
35             type= 'text'  
36             placeholder="Write a review"  
37             value = {text}  
38           />  
39           <Button type="submit" isDisabled={btnDisabled}>Send</Button>  
40         </div>  
41         {message && <div className="message">{message}</div>}  
42       </form>  
43     </Card>  
44   )  
45 }  
46 export default FeedbackForm
```

React JS

5. Add Feedback

```
JS FeedbackForm.jsx U JS App.js U X JS RatingSelect.jsx U
src > JS App.js > App > addFeedback
49 // )
50 // }
51
52 // export default App
53 import FeedbackItem from './components/FeedbackItem'
54 import {v4 as uuidv4} from 'uuid'
55 import {useState} from 'react'
56 import Header from './components/Header'
57 import FeedbackData from './data/FeedbackData'
58 import FeedbackList from './components/FeedbackList'
59 import Card from './components/shared/Card'
60 import FeedbackStats from './components/FeedbackStats'
61 import FeedbackForm from './components/FeedbackForm'
62 function App(){
63
64     const [feedback, setFeedback] = useState(FeedbackData)
65
66     const addFeedback = (newFeedback) =>{
67         newFeedback.id = uuidv4()
68         setFeedback([newFeedback, ...feedback])
69     }
70
71     const deleteFeedback = (id) => {
72         if(window.confirm('Are you sure you want to delete?')){
73             setFeedback(feedback.filter((item) => item.id !== id))
74         }
75     }
```

React JS

```
JS FeedbackForm.jsx U    JS App.js U X    JS RatingSelect.jsx U
src > JS App.js > ...
77
78     return (
79         <>
80             <Header />
81             <div className="container">
82                 <FeedbackForm handleAdd={addFeedback} />
83                 <FeedbackStats feedback={feedback}/>
84
85                 <FeedbackList feedback={feedback}
86                     |   handleDelete = {deleteFeedback}
87                 />
88                 <Card>
89                     |   Hello
90                 </Card>
91
92             </div>
93         </>
94     )
95 }
96 export default App
97
```


React JS

```
JS FeedbackForm.jsx U X JS App.js U JS RatingSelect.jsx U
src > components > JS FeedbackForm.jsx > FeedbackForm > handleSubmit
10 const [message, setMessage] =useState('')
11 const handleChange= (e) => {
12     if(text === ''){
13         setBtnDisabled(true)
14         setMessage(null)
15     } else if(text !== '' && text.trim().length <= 10){
16         setMessage('Text must be at least 10 characters')
17         setBtnDisabled(true)
18     } else {
19         setMessage(null)
20         setBtnDisabled(false)
21     }
22     setText(e.target.value)
23 }
24
25 const handleSubmit = (e) => {
26     e.preventDefault()
27     if(text.trim().length >10 ) {
28         const newFeedback = {
29             text,
30             rating,
31         }
32
33         handleAdd(newFeedback)
34
35         setText('')
36     }
37 }
38 return (
```

6. Fade Animation With Framer Motion

React JS

React JS

React JS

React JS