

- ▶ Para visualizar el componente que acabamos de crear modificamos App.js
- ▶ Visualizamos en Browser y vemos los cambio y el cambio de componente con su estado.

```
class App extends Component {  
  render() {  
    return (  
      <div className="App">  
        <TodoForm addTodo={(todo) => console.log(todo)} />  
      </div>  
    );  
  }  
}
```

MANEJO DE ESTADOS Y EVENTOS COMPLEJOS

- ▶ Creamos un componente con el nombre "TodoItem.js"
- ▶ Definimos los diferentes estados del componente.
- ▶ Definimos los eventos para los cambios de estos estados.

```
import React, {Component} from 'react'; 7.6K (gzipped:
```

Complexity is 9 It's time to do something...

```
export default class TodoItem extends Component { ■
```

```
  constructor(props) {  
    super(props);  
    this.state = {  
      editMode: false,  
      todo: props.children  
    }  
  }
```

```
  onEditMode () {  
    if(this.state.editMode) {  
      const {onUpdateTodo} = this.props;  
      onUpdateTodo(this.state.todo);  
    }  
    this.setState({  
      editMode: !this.state.editMode  
    });  
  }
```

```
  handleChange (evt) {  
    this.setState({todo: evt.target.value});  
  }
```