



รายงาน LAB 3SA03

เรื่อง 3SA03 - React

จัดทำโดย

นายอุสมาน สุหลง

รหัสนักศึกษา 5935512036

Sec02

เสนอ

Teacher: THITINAN KLIANGSUWAN

ภาคเรียนที่1 ปีการศึกษา2562

สาขาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

Check point

แสดง Hello World

```
src > JS App.js > ...
1 import React, { Component } from 'react';
2 import './App.css';
3 class App extends Component {
4   render() {
5     return (
6       <div className="App">
7         Hello World
8       </div>
9     );
10   }
11 }
12 export default App;
13
```



ผลลัพธ์

```
src > JS App.js > ...
1 import React, { Component } from 'react';
2 import './App.css';
3 import CharacterCard from './CharacterCard';
4 import WordCard from './WordCard';
5
6 const word = "Hello";
7 class App extends Component {
8   render() {
9     return (
10      <div className="App">
11        <WordCard value="Happy" />
12      </div>
13    );
14   }
15 }
16
17 export default App;
```

1. App.js ในส่วนของไฟล์ App.js เป็นไฟล์ที่หน้าที่แสดงหน้าต่างผลของการรันโปรแกรม ซึ่งในไฟล์ App.js นี้มีการเรียกใช้ไฟล์ อื่นภายในตัวด้วย เช่น การเรียกใช้ไฟล์ App.css , การเรียกใช้ไฟล์ WordCard.js

2. CharacterCard.js CharacterCard.js เป็นการทำให้ค่า values ที่เรากำหนดมานั้น สามารถกดได้ ซึ่งไฟล์ CharacterCard.js มีการ import จากไฟล์ App.css ในเรื่องของรูปแบบการจัดและสีของตัวอักษร และในไฟล์ CharacterCard.js มีการประกาศฟังก์ชันต่าง ๆ เช่น ฟังก์ชัน activate , componentDidMount และมีการประกาศ constructor เป็น state มี active : false การประกาศเป็น state เหมือนกับว่า state เป็นตัวแม่ ฟังก์ชันอื่นๆเป็นลูก ๆ ดังนั้นเวลาฟังก์ชันอื่นต้องการใช้ค่าก็ สามารถดึงค่านั้นไปใช้ได้ผ่าน ผ่านตัว this.state.active

```
src > JS CharacterCard.js > JS CharacterCard
1 import React, { Component, seq } from 'react';
2 import './App.css';
3 import App from './App';
4
5 export default class CharacterCard extends Component {
6   constructor(props) {
7     super(props);
8     this.state = {
9       active: false,
10     };
11   }
12
13   activate = () => {
14     if (!this.state.active) {
15       this.props.activationHandler(this.props.value);
16       this.setState({ active: true });
17     }
18   }
19   componentDidMount(preProps) {
20     if(preProps.attempt !== this.props.attempt) {
21       this.setState({ active: false });
22     }
23   }
24   render() {
25     let className = `card ${this.state.active ? 'activeCard' : ''}`;
26     return (
27       <div className={className} onClick={this.activate}>
28         {this.props.value}
29       </div>
30     );
31   }
32 }
33
```

3. WordCard.js ในส่วนของไฟล์ Wordcard.js เป็นไฟล์ที่ใช้สำหรับการสับคตำแหน่งของค่า values ที่เรากำหนดจากไฟล์ App.js และ ภายในไฟล์มีการประกาศใช้ฟังก์ชัน componentWillMount , activationHandler , prepareStateFromWord

```
90 // WordCard.js > WordCard > render
1 import React, { Component } from 'react';
2 import './App.css';
3 import CharacterCard from './CharacterCard';
4 import _ from 'lodash';
5 import App from './App';
6
7 const prepareStateFromWord = (given_word) => {
8   let word = given_word.toUpperCase();
9   let chars = _.shuffle(Array.from(word));
10  return {
11    word,
12    chars,
13    attempt: 1,
14    guess: [],
15    completed: false
16  }
17 }
18
19 export default class WordCard extends Component {
20   constructor(props) {
21     super(props);
22     this.state = {
23       attempt: "",
24     }
25   }
26   componentWillMount() {
27     let data = prepareStateFromWord(this.props.value);
28     this.setState({
29       word: data.word,
30       chars: data.chars,
31       attempt: data.attempt,
32       guess: data.guess,
33       completed: data.completed,
34     })
35   }
36 }
```

```
36
37   activationHandler = (c) => {
38     let guess = [...this.state.guess, c]
39     this.setState({ guess })
40     if (guess.length == this.state.chars.length) {
41       if (guess.join('') == this.state.word) {
42         this.setState({
43           guess: [],
44           completed: true
45         })
46       }
47       else {
48         this.setState({
49           guess: [],
50           attempt: this.state.attempt + 1
51         })
52       }
53     }
54   }
55
56   render() {
57     console.log(this.state);
58     return (
59       <div>
60         { this.state.chars.map((c, i) =>
61           <CharacterCard value={c}
62             key={i}
63             attempt={this.state.attempt}
64             activationHandler={this.activationHandler} />
65         ) }
66       </div>
67     )
68   }
```

4. App.css เป็นไฟล์สำหรับกำหนด layout ต่าง ๆ ที่ใช้ที่ไฟล์อื่น หรือ components อื่นสามารถนำไปใช้ได้

Code การทำงาน

```
src > # App.css > .card
1  .App {
2    text-align: center;
3  }
4  .App-logo {
5    animation: App-logo-spin infinite 20s linear;
6    height: 80px;
7  }
8  .App-header {
9    background-color: #222;
10   height: 150px;
11   padding: 20px;
12   color: white;
13 }
14 .App-title {
15   font-size: 1.5em;
16 }
17 .App-intro {
18   font-size: large;
19 }
20 .card {
21   display: inline-block;
22   text-align: center;
23   width: 3em;
24   font-size: 2em;
25   box-shadow: 0 4px 8px 0 rgba(0,0,0,0.2);
26   margin: 1em;
27   user-select: none;
28 }
29 @keyframes App-logo-spin {
30   from { transform: rotate(0deg); }
31   to { transform: rotate(360deg); }
32 }
```

ผลการทดลอง



Link : <https://github.com/5935512036/2SA03>

งานท้ายการทดลอง

หน้าเว็บเกม



โจทย์เพิ่มเติม

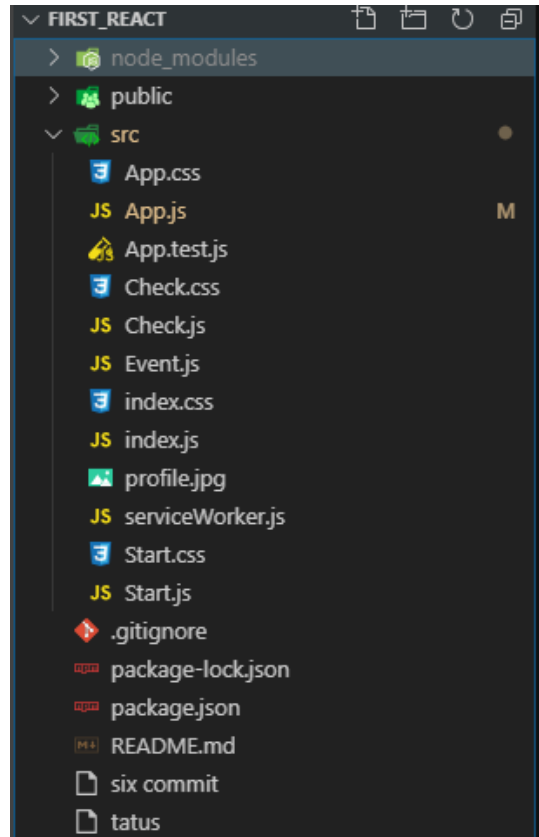
1. ให้ซ่อน Complete เมื่อตอบผิด
2. ทำให้ไม่สามารถเลือกปุ่มเดิมซ้ำกันได้
3. ให้รีเซ็ตหน้าเมื่อตอบเสร็จ (คืออยู่ในแลบซีท)

สิ่งที่คิดเพิ่มจากตัวอย่างในLAB



กำหนดให้เล่น 3 ครั้ง ถ้าหากตอบได้ก่อนให้มีปุมเล่นอีกครั้งพร้อมบอกว่าชนะ

ไฟล์ทั้งหมดในfolder



อัปเดต gihub 11 commit

Link : <https://github.com/5935512036/2SA03>

LABReact

Edit

Manage topics

11 commits

2 branches

0 releases

1 contributor

⚠ We found potential security vulnerabilities in your dependencies.
Only the owner of this repository can see this message.

View security alerts

Branch: master

New pull request

Create new file

Upload files

Find File

Clone or download

5935512036 last		Latest commit f8a6950 2 days ago
public	test game	4 days ago
src	last	2 days ago
.gitignore	first commit	11 days ago
README.md	test game	4 days ago
package-lock.json	gameV.thai	4 days ago
package.json	test game	4 days ago
six commit	test game	4 days ago
tatus	test game	4 days ago
yarn.lock	gameV.thai	4 days ago

Code App.js

```
1  import React from 'react';
2  import Event from './Event';
3  import Check from './Check';
4  import './App.css';
5  import Start from './Start'
6  import _ from 'lodash';
7  import profile from './profile.jpg'
8
9  const message = 'Apple'
10 const prepareStateFromWord = (given_word) => {
11   let word = given_word.toUpperCase()
12   let chars = _.shuffle(Array.from(word))
13   return {
14     word,
15     chars,
16     counter: 1,
17     guess: [],
18     completed: false,
19     check: 0
20   }
21 }
22
23 class App extends React.Component {
24   state = {
25     show: false,
26     give_up: false
27   }
28   state = prepareStateFromWord(message);
29   click = (value) => {
30     let guess = [...this.state.guess, value]
31     this.setState({ guess })
32     if (guess.length === this.state.chars.length) {
33       if (guess.join('').toString() === this.state.word) {
34         this.setState({ guess: [], completed: true })
35       } else {
36         this.setState({ guess: [], counter: this.state.counter + 1 })
37       }
38     }
39   }
40 }
```

1. Import ไฟล์ต่างๆเพื่อที่จะสามารถใช้codeหรือคำสั่งจากไฟล์ที่Import
2. เป็นฟังก์ชันที่returnค่าstateมา
Stateสำหรับจัดการข้อความโดยผมจะเพิ่มตัวแปรcheckเพื่อที่จะให้มันresetใหม่(เริ่มใหม่)
3. เป็นการต่อตัวอักษรให้เป็นข้อความ แล้วไปเปรียบเทียบกับค่าmessageที่แปลงเป็นตัวใหญ่แล้ว
4. ถ้าคำที่ทายมาไม่ถูกให้ทำการเดาเรื่อยๆแล้วเพิ่มค่าcounterไปหนึ่ง
5. ผมจะเพิ่มตัวstateที่โชว์คำไป กับ ชื่อของผม


```

39   }
40   show_name = () => {
41     this.setState({
42       show: !this.state.show
43     })
44     console.log(this.state.show) // check
45   }
46
47   reset = () => {
48     this.setState({ check: this.state.check + 1, completed: false })
49   }
50   give_ups = () => {
51     this.setState({
52       give_up: !this.state.give_up
53     })
54   }
55   render() {
56     let check = this.state.completed === false ? '' :
57     <button className="button" onClick={this.reset}><h1>Play Again</h1></button>;
58
59     let ans = this.state.completed === false ? '' : <h3 className="ans">This word is {message}</h3>;
60     let checks = this.state.completed === false ? '' : <h1 className="win">You Win</h1>;
61     //let count_end = this.state.counter > 5 ? "Game Over" : "Counter : " + this.state.counter;
62
63     return (
64       <div>
65         <div>
66           <Start />
67         </div>
68         <div className="first">
69           <div className="second">
70             <div className="profile-area"><img className="profile" src={profile}></img></div>
71             <div className="center-box">
72               <div className="center-text">
73                 <div className="text1"><h1 className="text">LAB:3SA03 ? 6035512059</h1></div>

```

- 1.เปลี่ยนค่าstateให้ตรงกันข้ามเรื่อยๆ หากมีการกด
- 2.กำหนดตัวแปรรับค่าที่ตรงเงื่อนไขเป็นปุ่มเริ่มเกมใหม่
- 3.ตัวแสดงรูปภาพ
- 4.เรียกใช้งานไฟล์Start.js

```

73 |         <div className="text1"><h1 className="text">LAB:3SA03 ? 6035512059</h1></div>
74 |         <div className="click">
75 |           <div className="button-a">
76 |             <button className="button-show" onClick={this.show_name}>{this.state.show === false ? 'Show my name' : 'Hide my name'}</button>
77 |           </div>
78 |           <div className="box-myname">
79 |             <div></div>
80 |             <h3 className="text">{this.state.show === true ? 'Arim Cheberahim' : ''}</h3>
81 |           </div>
82 |         </div>
83 |       </div>
84 |     </div>
85 |
86 |     <div>
87 |     {
88 |       Array.from(this.state.chars).map((x, y) => (
89 |         <Event
90 |           value={x}
91 |           key={y}
92 |           click={this.click}
93 |           number={this.state.counter}
94 |           check={this.state.check}
95 |         />
96 |       ))
97 |     }
98 |     <div className="box">
99 |       <h1 className="text">Choose</h1>
100 |       <h1 className="text">You can play 5 Round</h1>
101 |       {
102 |         Array.from(this.state.guess).map((x, y) => (
103 |           <Event
104 |             value={x}
105 |             key={y}
106 |             click={this.click}
107 |           />
108 |         ))
109 |       }
110 |     </div>

```

5. เป็นการแยกตัวอักษรแล้วส่งไปให้ Event.js จัดการต่อส่งตัวเมคทอด click ตัวแปร counter, check ที่เก็บค่า state แต่ละตัว

```

110 |         <div>
111 |           { /* <h1 className="text">{this.count_end}</h1> */ }
112 |           <Check check_count={this.state.counter} />
113 |         </div>
114 |         <div className="button-area">
115 |           {check}
116 |           {ans}
117 |           {checks}
118 |         </div>
119 |       </div>
120 |     </div>
121 |     <div className="center-text">
122 |       <button className="button-show" onClick={this.give_ups}>Hint</button>
123 |       <h3>{this.state.give_up === true ? 'Ans:Fruit' : ''}</h3>
124 |     </div>
125 |   </div>
126 | </div>
127 | </div>
128 | )
129 | }
130 | }
131 | export default App;
132 |

```

6. แสดงค่าที่ตัวแปร check, ans, checks เก็บไว้ดูได้จากโค้ดบรรทัด 56-60

Code App.css

```
1  body{
2    background-color: #273746;
3    font-family: 'Roboto', sans-serif;
4  }
5  .first{
6    background: #839192;
7    display: block;
8    margin: 0 auto;
9    width: 21.5cm;
10   height: 30.7cm;
11   margin-bottom: 0.5cm;
12   box-shadow: 0 0 0.5cm #000000;
13 }
14
15 .profile{
16   width: 150px;
17   height: 150px;
18   margin: 16px 32px;
19   border-radius: 100%;
20   box-shadow: 0 4px 8px #000000;
21   border-bottom: 3px solid #273746;
22 }
23
24 .profile-area{
25   display: flex;
26   justify-content: center;
27   align-items: center;
28 }
29 .card {
30   display: inline-block;
31   text-align: center;
32   width: 3em;
33   font-size: 2em;
34   box-shadow: 0 4px 8px #000000;
35   margin: 1em;
36   user-select: none;
37   cursor: pointer;
```

4. Class text กำหนดให้ข้อความของเรา
อยู่ตรงกลาง แล้วกำหนดสีของเส้นขอบ
ล่างกับเงาของข้อความด้วย

- 1.กำหนดสีbackground กับ font ในbody

Bodyเป็นclass ใหญ่สุดของbackground

- 2.เป็นส่วนของรูปprofileโดยกำหนด
ความกว้างความสูง ระยะห่าง องศาของ
ขอบรูป เงา เส้นขอบใต้ภาพ

- 3.กำหนดให้รูปอยู่ตรงกลาง

```
38   border-radius: 12px;
39   border-bottom: 3px solid #273746;
40 }
41 .activeCard {
42   color: #273746;
43   background: #566573;
44 }
45 .center-text{
46   display: flex;
47   justify-content: center;
48   align-items: center;
49   flex-direction: column;
50 }
51 .text{
52   text-align: center;
53   border-bottom-color: #273746;
54   text-shadow: 0 4px 8px #000000;
55 }
56
57 }
58 .text1{
59   box-sizing: border-box;
60   width: 50%;
61   box-shadow: 0 4px 8px #000000;
62   float: left;
63   border-radius: 12px;
64   border-bottom: 3px solid #273746;
65 }
66 .click{
67   display: flex;
68   justify-content: center;
69   align-items: center;
70   flex-direction: column;
71 }
```

```

73 .button-show{
74   border: none;
75   padding: 10px 15px;
76   text-align: center;
77   text-decoration: none;
78   display: inline-block;
79   margin: 4px 2px;
80   cursor: pointer;
81   color: black;
82   background: #808B96;
83   border-radius: 12px;
84   text-shadow: 0 4px 8px 0 rgba(0,0,0,0.2);
85   border-bottom: 3px solid #273746;
86 }
87 .button-show:hover{
88   box-shadow: 2px 2px 8px #000000;
89   background: whitesmoke;
90 }
91
92 .center-box{
93   display: flex;
94   justify-content: center;
95   align-content: center;
96   flex-direction: column;
97 }
98 .box-myname{
99   display: flex;
100  justify-content: center;
101  align-content: center;
102  flex-direction: column;
103  box-sizing: border-box;
104  width: 100%;
105  box-shadow: 0 4px 8px 0 rgba(0,0,0,0.2);
106  border-radius: 12px;
107  border-bottom: 3px solid #273746;
108 }

```

```

109 .second{
110   padding-top: 10px;
111 }
112 .box{
113   box-shadow: 0 4px 8px 0 rgba(0,0,0,0.2);
114   background: #566573;
115   border-radius: 12px;
116 }
117 .button{
118   border: none;
119   padding: 5px 10px;
120   text-align: center;
121   text-decoration: none;
122   display: inline-block;
123   margin: 4px 2px;
124   cursor: pointer;
125   background-color: white;
126   color: black;
127   border: 2px solid #273746;
128   border-radius: 12px;
129   text-shadow: 0 4px 8px 0 rgba(0,0,0,0.2);
130 }
131
132 .ans{
133   text-align: center;
134   color: white;
135   text-shadow: 0 4px 8px 0 rgba(0,0,0,0.2);
136 }
137 .button-area{
138   display: flex;
139   padding: 10px 300px;
140   justify-content: center;
141   flex-direction: column;
142   border-radius: 12px;
143 }

```

5. คำสั่ง border เป็นการ
เพิ่มกรอบของสิ่งนั้นๆ

solid เป็นเส้นทึบทำงาน

คล้ายๆ stroke ใน

```

144 .button:hover{
145   box-shadow: 2px 2px 8px #000000;
146   background: #808B96;
147   border: 2px solid #273746;
148 }
149 .win{
150   display: flex;
151   justify-content: center;
152   color: white;
153   text-shadow: 0 4px 8px 0 rgba(0,0,0,0.2);
154 }
155
156

```

Code Event.js (เป็นไฟล์ที่เอาตัวแปร เมคทอด state มาใช้งานแล้วส่งค่ากลับ)

```
1 import React from 'react';
2 import './App.css';
3 import App from './App';
4 import Check from './Check';
5 class Event extends React.Component {
6   state = {
7     active: false
8   }
9   activate = () => {
10     this.setState({
11       active: true
12     });
13     if (this.props.number > 5) {
14       this.setState({
15         active: true
16       });
17     }
18     else if (this.state.active === false)
19       this.props.click(this.props.value);
20   }
21   componentDidUpdate = (prevProps) => {
22     if (prevProps.number !== this.props.number || prevProps.check !== this.props.check) {
23       this.setState({ active: false })
24       console.log('...');
25     }
26   }
27   render() {
28     let activeClass = this.state.active ? 'activeCard' : '';
29     let className = `card ${activeClass}`
30     return (
31       <div className={className} onClick={this.activate}>
32         <h1>{this.props.value}</h1>
33       </div>
34     )
35   }
36 }
37 export default Event;
```

1.เป็นเงื่อนไขที่เช็คค่าถ้า

ตัวอักษรที่เราคลิกมันยังไม่ถูก

activeหรือยังไม่มีกรกดให้เข้า

เงื่อนไขนี้ได้ทำให้ไม่สามารถ

~

2.เป็นส่วนของการอัปเดตคลาสมาแสดงโดยเรา

จะเก็บคลาสที่จะอัปเดตไว้ในตัวแปรต้องมีตัว\$

Code Check.js

```
1 import React from 'react';
2 import './Check.css';
3 import Event from './Event';
4 import App from './App'
5 class Check extends React.Component {
6   render() {
7     //let count_end = this.props.check_count > 5 ? "Game Over" : "Counter : " + this.props.check_count;
8     let count_end = this.props.check_count > 5 ? 'area2' : '';
9     //let colors = count_end > 5 ? 'area2' : '';
10    let count_ends = this.props.check_count > 5 ? "Game Over" : "Counter : " + this.props.check_count;
11    let test = `area ${count_end}`
12    return (
13      <div>
14        <div className={test}>
15          <h1 className={test}>{this.count_end}{count_ends}</h1>
16        </div>
17      </div>
18    )
19  }
20 }
21
22
23 export default Check;
```

1. ชื่อ class ที่ทำตัวอัปเดตได้โดย
class area จะแสดงอยู่แล้วแต่ถ้า
หากเข้าเงื่อนไขที่ตัวแปร test เก็บไว้
มันก็จะอัปเดตตัว css ทันที

Code Check.css

```
1 1 .area{
2   display: flex;
3   justify-content: center;
4   align-items: center;
5   color: ☐ black;
6 }
7 2 .text{
8   text-align: center;
9 }
10 3 .area2{
11   display: flex;
12   justify-content: center;
13   align-items: center;
14   color: ☒ orange;
15 }
```

Code Start.js

```
1 import React, { useState } from 'react';
2 import ReactDOM from 'react-dom'
3 import './Start.css';
4 import { Route, Link, BrowserRouter as Router } from 'react-router-dom'
5 import App from './App'
6
7 function Start() {
8   return (
9     <div className="one">
10       <div className="row">
11         <h1 className="text-head">Let's play my first game on react</h1>
12         <div><h1 className="color">(js)</h1></div>
13       </div>
14     </div>
15   );
16 }
17 export default Start;
18
```

1. เป็นการ Import router แต่ผมไม่ได้ใช้ในการเขียนครั้งนี้ เพราะมันจะทำให้ตัวจัดไฟล์ CSS ต้องจัดใหม่หมด เพราะผมมาทำไอ้ที่หลังผมเลยตั้งทิ้งไว้เฉยๆ แต่รู้วิธีการใช้แล้ว

2. มันคือตัวที่ link ไปหน้า .js หรือ .html อื่น

Code Start.css

```
1 .body{
2   background-color: #273746;
3   font-family: 'Roboto', sans-serif;
4 }
5 .one{
6   display: flex;
7   align-items: center;
8   justify-content: center;
9   align-self: center;
10  flex-direction: column;
11 }
12 .two{
13   background: #839192;
14   display: block;
15   margin: 200px;
16   width: 50vh;
17   height: 25vh;
18   margin-bottom: 0.5cm;
19   box-shadow: 0 0 0.5cm rgba(0,0,0,0.5);
20   border-radius: 12px;
21 }
22
23 .button55{
24   background-color: white;
25   border: none;
26   color: black;
27   padding: 15px 32px;
28   text-align: center;
29   text-decoration: none;
30   display: inline-block;
31   font-size: 16px;
32   margin: 4px 2px;
33   cursor: pointer;
34   border: 2px solid #008CBA;
35 }
36
```

3. คลาส one ของผมเป็นคลาสที่จัดทุกอย่างที่อยู่โน้นให้อยู่ตรงกลางแล้วเรียนเป็นแถวๆ ตาม Div

```
37  .three{
38
39  }
40  .text-head{
41    color: ■aliceblue;
42    font-size: 350%;
43  }
44  }
45  .color{
46    color: ■yellow;
47    font-size: 350%;
48    margin-left: 10px;
49  }
50  .row{
51    display: flex;
52    flex-direction: row;
53    align-items: center;
54    justify-content: center;
55  }
```

4. Class color กำหนดสีของข้อความ แล้ว
ขนาดของข้อความกับระยะห่างทางด้านซ้าย