



Web Programming

Spring 2021



#26

Chi-Jen Wu



先把google meet 麥克風關掉！
如果要發言再打開!!
不然全班都會聽到奇怪的聲音

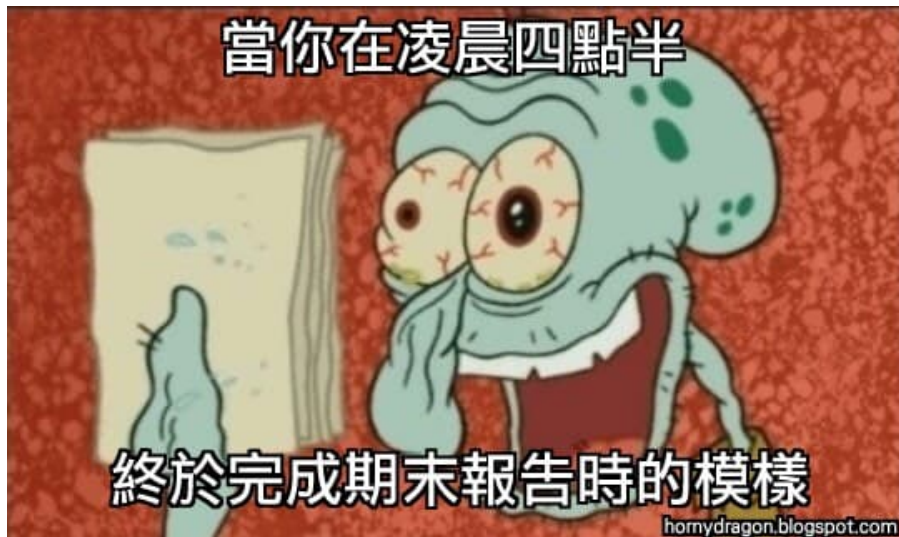
Topics

- The concepts of Web Services
- Web data protocols
 - HTTP, WebSocket, WebRTC
 - HTML, CSS
- Web JavaScript programming
- Cookies and sessions
- **Web Frontend frameworks**
- Web Backend frameworks
- RESTful API design



期末報告分組 (40%)

- 2個人一組 或 乙個人一組 皆可
- 請用React 實作至少包含**五**個頁面的網站
 - 像是登入頁 個人頁 首頁 等等
 - 要有RWD
 - 要有Google Analytics
- 後端必須要連動DB
 - API
- 部署到 heroku
 - 現在可以申請
 - 三個月免費使用 有VM, 有DB
- 下星期三前確定組員 (5/19)
- 最後兩堂課 請同學上來和報告 demo 成果
 - 每組10頁投影片介紹 每組大概十分鐘

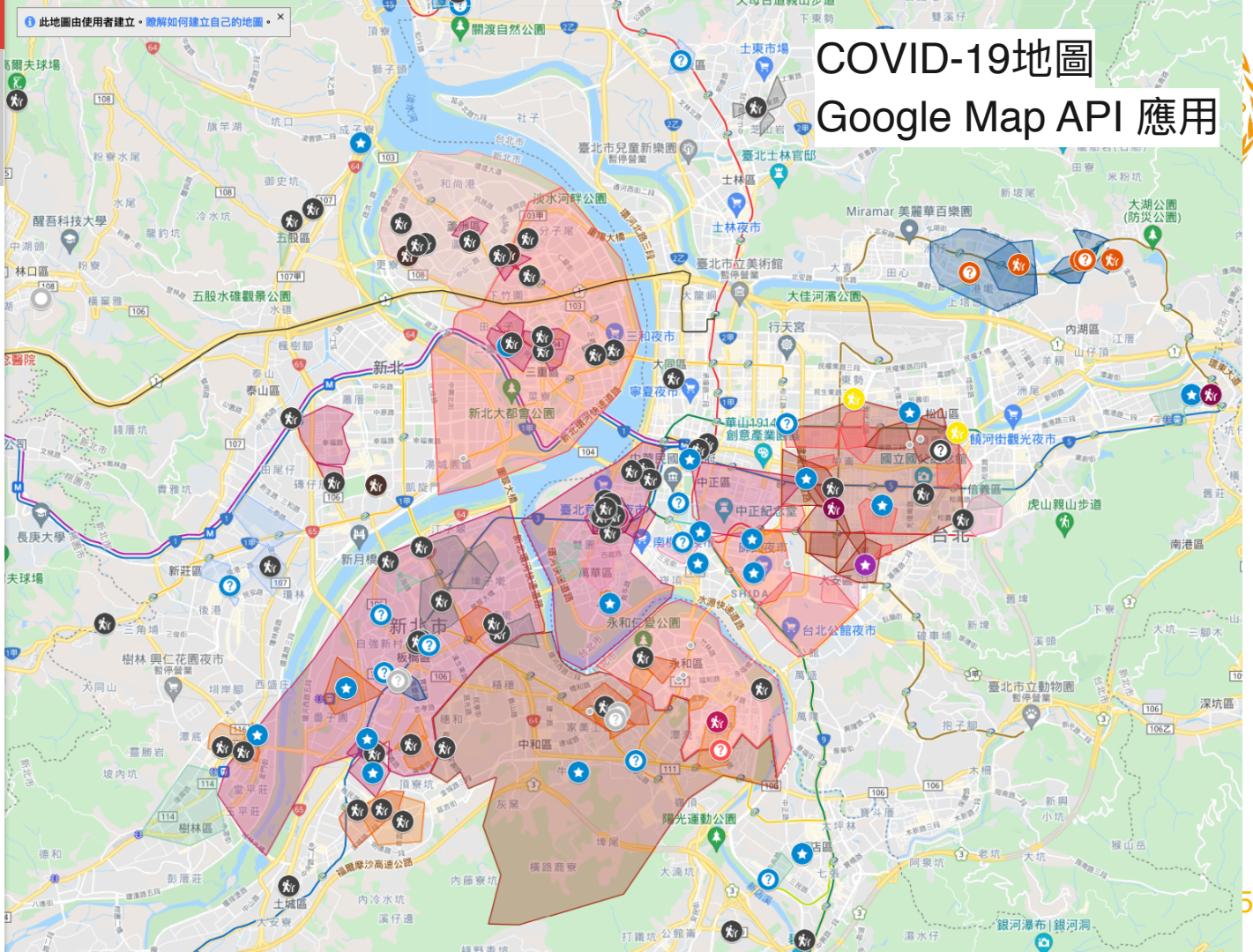


口罩戴好，現在不把每個人當確診，你就會確診。
請愛用====臺灣社交距離App====
瀏覽次數：16,091,102 次
分享

生活圈-皆為預估、皆為預估、皆為預估

說明用

- 危險(高)-5/17板橋區
- 危險(高)-萬華
- 危險(高)-大安區
- 危險(中低)-蘆洲
- 危險(中高)-中和
- 危險(中)-三重
- 危險(中)-永和
- 生活圈-中山女高+中正國小
- 生活圈-敦化國中
- 生活圈-松山國小
- 生活圈-育達商職
- 生活圈-中崙高中
- 生活圈-懷生國小
- 生活圈-北科大
- 生活圈-懷生國中
- 生活圈-幸安國小
- 生活圈-師大高中
- 生活圈-東華大學+大安高工+大安國中+建...
- 生活圈-仁愛國小
- 生活圈-松山高中
- 生活圈-松山高工
- 生活圈-興雅國中+博愛國小
- 生活圈-成功國宅
- 生活圈-北教大
- 生活圈-台大





先收個心
先來看個電影先!!



電影欣賞 quiz#6

[https://www.youtube.com/
watch?v=pOHefN8SIGw](https://www.youtube.com/watch?v=pOHefN8SIGw)



Steve Jobs 賈伯斯 - 史丹佛大學畢業典禮演說

0. 看完 (15分鐘)
1. 寫心得 300字以上&500字以內心得
2. 填寫到 <https://forms.gle/hm1CXNQ1vHJdUbtNA>
3. Deadline 5/19 3:00PM



React 重點回顧

- 開發環境
- 核心概念 — 元件(Components)
- 單向資料流 (Unidirectional data flow)
- **React Router**
- **React 打API的方法**



React Conditional Router

```
function UserGreeting(props) {  
  return <h1>Welcome back, CGUER!</h1>;  
}
```

```
function GuestGreeting(props) {  
  return <h1>Hi CGUer, please sign up.</h1>;  
}
```

```
function Greeting(props) {  
  const isLoggedIn = props.isLoggedIn;  
  return props.isLoggedIn ? <UserGreeting /> : <GuestGreeting />;  
}
```

```
ReactDOM.render(  
  <Greeting isLoggedIn={false} />,  
  document.getElementById('root')  
)
```



課堂練習 react_quiz#7
把 homework#4 登入頁
和 homework#6 todolist
用 React Conditional Router 連起來
假設登入都是正確的



線上練習 react_quiz#7

codesandbox.io

0. **npm run build**

1. Commit **all file** to **react_quiz7**

2. push to **react_quiz7** repository

3. Deadline 5/19 4:00PM

React 打API的方法

- 之前講過透過jQuery和Ajax來和API拿資料
- 還有很多種方式可以讓React和API溝通
- 我們要介紹一下目前最多人用的
 - Axios
 - 簡單好用
 - 又有Promise

The Axios logo, consisting of the word 'Axios' in a sans-serif font. The 'A' is blue, and the rest of the letters are white, set against a dark gray rectangular background.

Axios

React + Axios

- 基於 Promise 的 HTTP 客戶端
- 他不是專門給React 用的
- 和 jQuery一樣
- `npm install axios`

The Axios logo, consisting of the word "Axios" in a sans-serif font. The "A" is blue, and the "xios" is white, all set against a dark gray rectangular background.

React api.js



```
<script>
```

```
import axios from 'axios';
```

```
const userRequest = axios.create({  
  baseURL: 'https://api/user/'  
});
```

```
const todoitemsRequest = axios.create({  
  baseURL: 'https://api/todoitems/'  
});
```

```
export const apiUserLogin = data => userRequest.post('/signIn', data);  
export const apiUserProfile = () => userRequest.get('/profile');
```

```
export const apiToDoItems = () => articleRequest.get('/todoitems');  
export const apiToDoItemsUpdate = data => articleRequest.post('/todoitems', data);
```

```
</script>
```

React api.js



```
<script>
```

```
import { apiUserLogin, apiUserProfile } from "api.js";
```

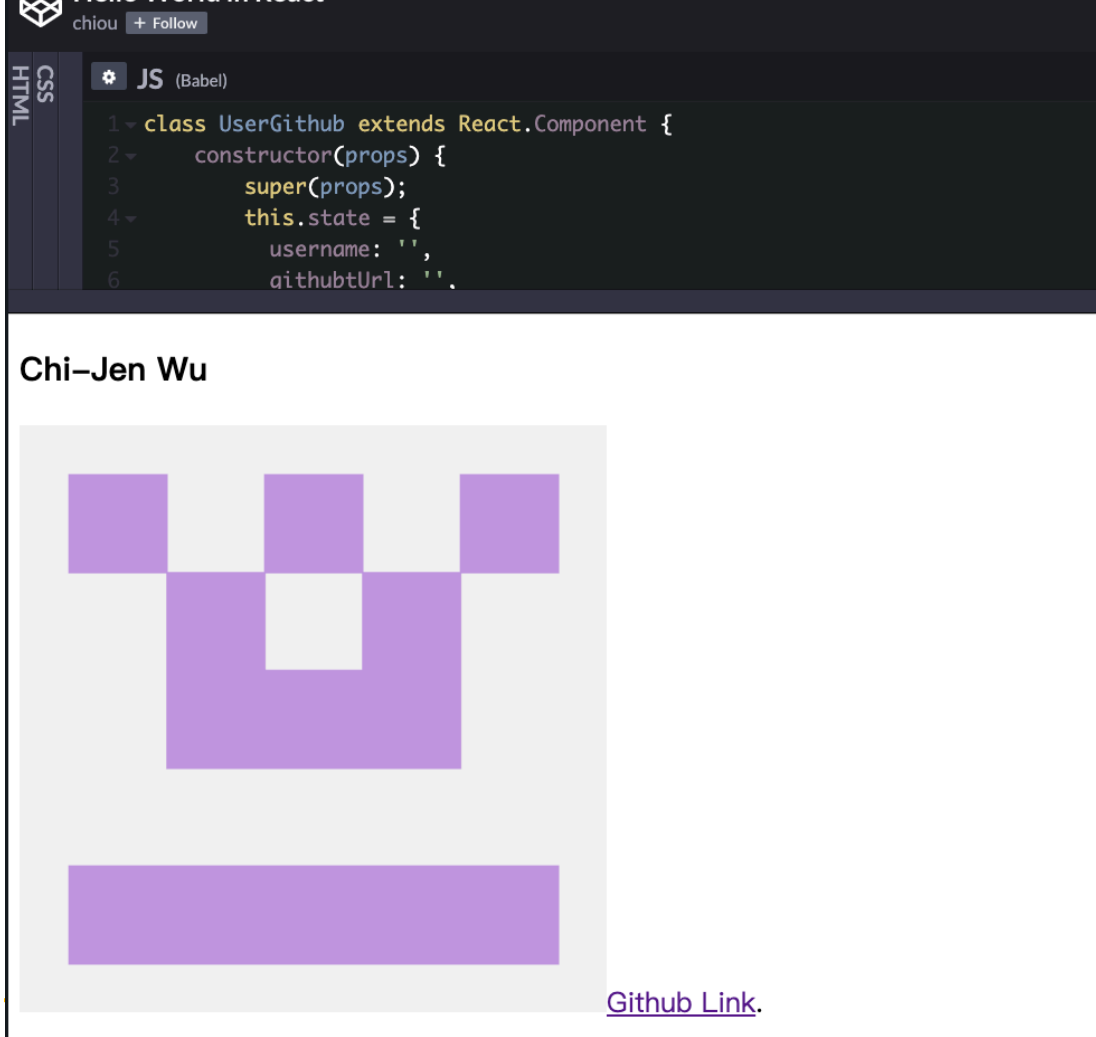
```
apiUserLogin({  
  email: "cjwu@mail.cgu.edu.tw",  
  password: "cgu123"  
})
```

```
.then(res=> {  
  console.log(res);  
})  
.catch(err=> {  
  console.log(err);  
})
```

Promise方式

```
</script>
```


React Components + jQuery + Ajax

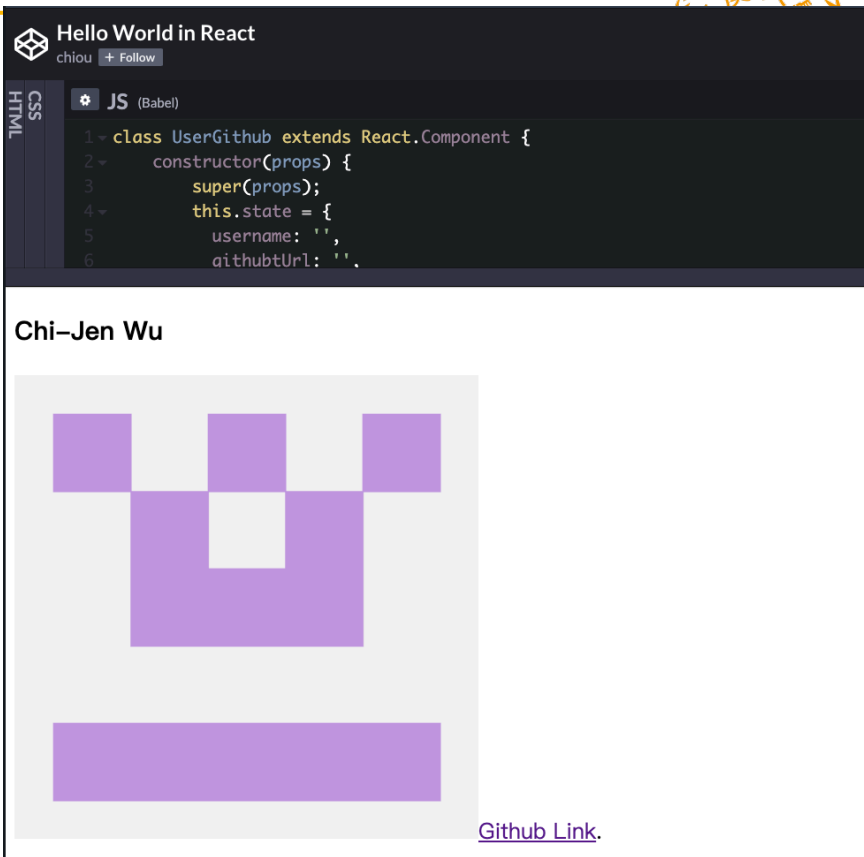


```

<script>
class UserGithub extends React.Component {
  constructor(props) {
    super(props);
    this.state = {
      username: '',
      githubUrl: '',
      avatarUrl: '',
    }
  }
  componentDidMount() {
    $.get(this.props.source, (result) => {
      console.log(result);
      const data = result;
      if (data) {
        this.setState({
          username: data.name,
          githubUrl: data.html_url,
          avatarUrl: data.avatar_url
        });
      }
    });
  }
  render() {
    return (
      <div>
        <h3>{this.state.username}</h3>
        <img src={this.state.avatarUrl} />
        <a href={this.state.githubUrl}>Github Link</a>.
      </div>
    );
  }
}

ReactDOM.render(
  <UserGithub source="https://api.github.com/users/cjwu" />,
  document.getElementById('root')
);
</script>

```



用axios 修改 原本的方式



api.js

```
<script>
```

```
import axios from 'axios';
```

```
const userRequest = axios.create({  
  baseURL: 'https://api.github.com/users/'  
});
```

```
export const apiUserProfile = (username) =>  
  userRequest.get(username);
```

```
</script>
```

```
<script>
```

```
import {apiUserProfile} from "api.js";
```

```
class UserGithub extends React.Component {  
  constructor(props) {  
    super(props);  
    this.state = {  
      username: '',  
      githubtUrl: '',  
      avatarUrl: '',  
    }  
  }  
  componentDidMount() {  
    apiUserProfile(this.state.username)  
      .then(res=> {  
        If (res) {  
          this.setState({  
            username: res.name,  
            githubtUrl: res.html_url,  
            avatarUrl: res.avatar_url  
          });  
        }  
      })  
      .catch(err=> {  
        console.log(err);  
      })  
  }  
}
```

app.js



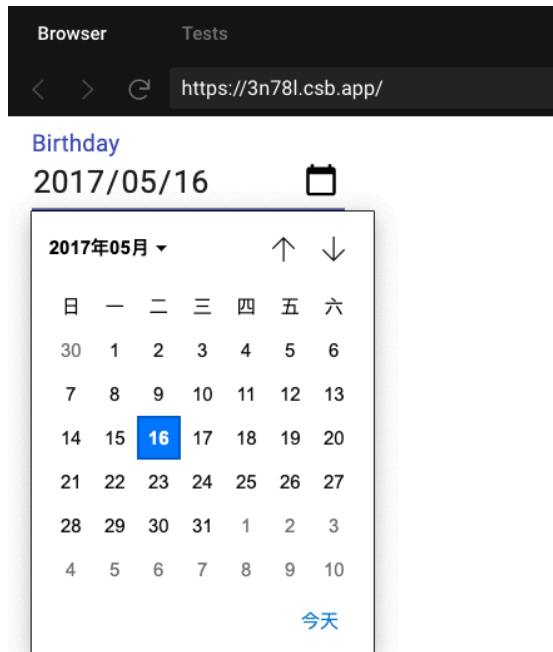
api.js



```
<script>
  render() {
    return (
      <div>
        <h3>{this.state.username}</h3>
        <img src={this.state.avatarUrl} />
        <a href={this.state.githubUrl}>Github
Link</a>
      </div>
    );
  }
}
ReactDOM.render(
  <UserGithub source="cjwu" />,
  document.getElementById('root')
);
</script>
```

Homework #7 阿公店開店日選擇

- 修改material-ui Datepickers
 - 原本是 2021-05-16
 - 改成呈現民國110年5月16號
- npm run build
 - git add ./build
 - Without node_modules
- 然後 push github
- 2021/05/26 12:00PM



```
import React from 'react';
import { makeStyles } from '@material-ui/core/styles';
import TextField from '@material-ui/core/TextField';
```

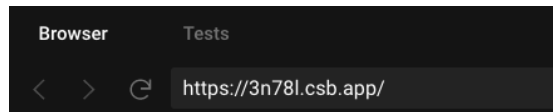
記得install



```
const useStyles = makeStyles((theme) => ({
  container: {
    display: 'flex',
    flexWrap: 'wrap',
  },
  textField: {
    marginLeft: theme.spacing(1),
    marginRight: theme.spacing(1),
    width: 200,
  },
}));

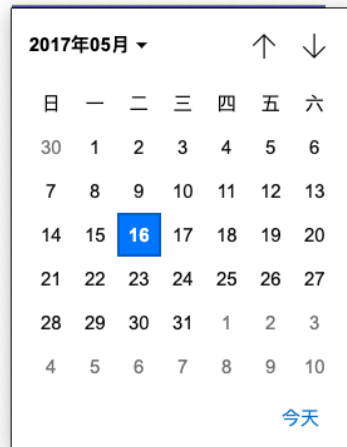
export default function DatePickers() {
  const classes = useStyles();

  return (
    <form className={classes.container} noValidate>
      <TextField
        id="date"
        label="Birthday"
        type="date"
        defaultValue="2017-05-24"
        className={classes.textField}
        InputLabelProps={{
          shrink: true,
        }}
      />
    </form>
  );
}
```

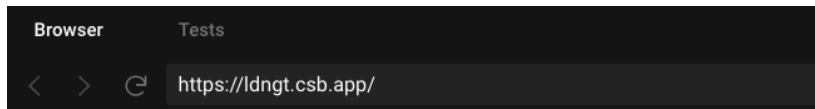


Birthday

2017/05/16



Homework #7 阿公店開店日選擇



民國110年-5月-16號



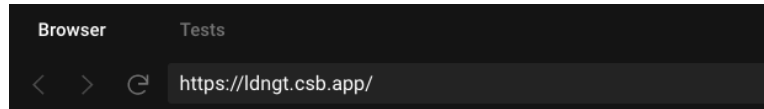
2021年05月 ▾

↑

↓

日	一	二	三	四	五	六
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

今天



民國110年-5月-29號



2021年05月 ▾

↑

↓

日	一	二	三	四	五	六
25	26	27	28	29	30	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31	1	2	3	4	5

今天



Homework #7 阿公店開店日選擇

有時候現成元件

無法滿足需求時

就必須動手修改他

改不改得動 就是工程師的差別





Thanks!

Open for any questions

CJ Wu

cjwu@mail.cgu.edu.tw