**Quy định khi code**

**1. Quy tắc comment**

**Comment cho Class (Component):**

/\* Author: Tên tác giả – Desc: Nội dung của class (component) \*/

VD:

/\* Author: Đỗ Trường Giang – Desc: Đây là Component Header đùng để hiển thị menu chính của app

\*/

import React, {Component} from ‘react’;

class Header extends Component {

render() {

return(

<h1>Giang so handsome</h1>

);

}

}

**Comment cho Class (Component) khi có BUG:**

/\*

[bug01 - Nội dung của bug]

[bug02 - Nội dung của bug]

class Header extends Component {

render() {

return(

<h1>Giang so handsome</h1>

);

}

}

[/bug02]

[/bug01]

\*/

**Lưu ý:** Khi fix được bug nào thì chỉ cần bỏ tag bug đó

VD: Sau khi fix được bug02 nhưng chưa fix được bug01 thì sẽ comment như sau:

/\*

[bug01 - Nội dung của bug]

class Header extends Component {

render() {

return(

<h1>Giang so handsome</h1>

);

}

}

[/bug01]

\*/

**Comment cho Class (Component) khi đã test:**

/\*

Author: Tên tác giả – Desc: Nội dung của class (component) - Has been tested - lần thứ bao nhiêu

\*/

VD:

/\* Author: Đỗ Trường Giang – Desc: Đây là Component Header đùng để hiển thị menu chính của app - Has been tested - 02

\*/

import React, {Component} from ‘react’;

class Header extends Component {

render() {

return(

<h1>Giang so handsome</h1>

);

}

}

**Comment cho function:**

/\* Author: Tên tác giả – FunctionName: Tên function là gì - Desc: Nội dung của function là gì

param1: đối số thứ nhất là gì - kiểu dữ liệu của đối số 1 (nếu có)

param2: đối số thứ hai là gì - kiểu dữ liệu của đối số 2 (nếu có)

…

return: function này trả về cái gì

\*/

VD:

/\* Author: Đỗ Trường Giang – FunctionName: onShowList – Desc: Hiển thị danh sách các task của user

param1: tasks - array

return: danh sách các task của user

\*/

onShowList = tasks => {

let list = tasks.map( task => {

return <p key={task.id}>{task.content}</p>

} );

}

**Comment cho function khi có người sửa:**

/\* Author: Tên tác giả sửa – Desc: Nội dung sửa là gì \*/

**Lưu ý:** comment này phải ở dưới function của người làm ra function đó và phải comment cả function của người làm ra rồi sửa lại ở dưới để tiện cho việc xem lại code cũ

VD:

/\* Author: Đỗ Trường Giang – FunctionName: onShowList – Desc: Hiển thị danh sách các task của user

param1: tasks - array

return: danh sách các task của user

\*/

/\*

onShowList = tasks => {

let list = tasks.map( task => {

return <p key={task.id}>{task.content}</p>

} );

};

\*/

/\* Author: Lưu Thành Đạt – Desc: chuyển task.id thành index \*/

onShowList = tasks => {

let list = tasks.map( (task, index) => {

return <p key={index}>{task.content}</p>

} );

};

**Comment cho function khi có vấn đề về hiệu năng:**

/\*[metric]\*/

<Function có vấn đề về hiệu năng>

/\*[/metric]\*/

VD:

/\*[metric]\*/

onSort = (tasks) => {

for( let i = 0; i < tasks.length; i++ ) {

for( let j = i+1; j < tasks.length – 1; j++ ) {

<Nội dung xử lý>

}

}

};

/\*[/metric]\*/

**2. Quy tắc đặt tên cho file và class (component)**

Tên của class (component) phải là danh từ và in hoa chữ cái đầu và tên của file phải trùng với tên của class (component)

VD: file: Header.js

import React, {Component} from ‘react’;

class Header extends Component {};

export default Header;

**3. Quy tắc đặt tên cho function:**

Tên function phải là động từ viết theo camel case và phải có ‘on’ đằng trước tên function

VD: onShowList = () => {};

onAddTask = () => {};

**4. Quy tắc đặt tên biến:**

Tên biến không được đặt tùy tiện như là a hay b hay là ironman mà tên biến phải có nghĩa đọc vào có thể hiểu được biến đấy dùng để lưu trữ cái gì

VD: let tasks;

**Lưu ý 01:** khi tên biến quá dài thì hãy cho thêm dấu ‘\_’ vào để thay thế khoảng trắng (dùng bên phía back-end)

VD: let number\_of\_task;

khi tên biến quá dài thì hãy dùng camel case để thay thế khoảng trắng (dùng bên phía front-end)

VD: let numberOfTask;

**Lưu ý 02:** hạn chế sử dụng var để đặt cho biến hãy sử dụng let hoặc const để thay thế

VD: var task: HẠN CHẾ

let task: NÊN DÙNG

**Lưu ý 03:** khi đặt biến là const thì hãy ghi in hoa toàn bộ tên biến

VD: const PI;

**Lưu ý 04:** đối với array thì tên biến phải là danh từ số nhiều và object thì tên biến phải thể hiện được đối tượng

VD: let tasks; // array

let task; // object

**5. Quy tắc đặt tên class và id khi css:**

Dùng chuẩn BEM để đặt tên cho class và id khi cần css

.block {} /\* Block \*/

.block\_\_element {} /\* Element \*/

.block--modifier {} /\* Modifier \*/

VD Element:

<div class="td">

<div class="td\_\_title">

</div>

<div class="td\_\_description">

</div>

</div>

VD Modifier:

.btn--red {

background: red;

}

----- HẾT -----