# Bí quyết code hạn chế bugs tối đa 😍

CẢM ƠN CÁC BẠN ĐÃ ỦNG HỘ MÌNH TUẦN ĐẦU TIÊN RA MẮT

Số videos: 7

#### Youtube

Likes: 334

• Comments: 313 💬

#### Facebook:

Reactions: 1635 Comments: 148

• Shares: 330 @

Và vẫn đang tiếp tục tăng đều! 🧡

RẤT NHIỀU PHẢN HỒI TÍCH CỰC TỪ CÁC BẠN



Tien Tran • 2 hours ago

anh luc nao cung dep zai, cam on a



DUC NGUYEN HUU • 11 hours ago

Cảm ơn anh, Video dạy thì rất nhiều người chia sẻ còn video kinh nghiệm như vậy thì rất ít <3



Phạm Vũ Hùng • 7 hours ago

chất lượng quá a, nghe mấy bài giảng kiểu này thấm kiến thức gê <3



12 Showbiz • 1 day ago

A làm video về cách sử dụng từng thẳng lifecycle trong reactjs đi ạ. A truyền đạt rất hay và dễ hiểu.



Nguyen Nhi • 4 days ago

rất cảm ơn anh vì chia sẻ những tips hay... Em cũng đang học html vs js. Mong anh ra nhiều video để em học hỏi. em ít khi cmt lắm... cơ mà xem vid của anh dễ hiểu và gần gũi nên rất thích ạ!



August • 4 days ago

Video của anh dễ hiểu và tận tâm. Keep it up nha anh



Minh Quang • 5 days ago

Kênh mới lập nhưng chất lượng đỉnh lắm a!!!mong a ra video nhiều hơn... <3

# Chia sẻ - Code thế nào cho đỡ bug

Chống chỉ định: hạn chế bugs thôi nha, còn code không bug thì chỉ có một cách duy nhất --> Nghỉ code



Giới thiệu mức độ tỉnh của bạn qua 3 giai đoạn

Giai đoạn	Mức độ tỉnh
Nhận task và phân tích	Rất tỉnh và đẹp trai 😎
Bay vào code	Ngu dần đều 🤣
Testing	Cuối cùng, code cũng chạy 🥹

## Những lỗi phổ biến:

- Code một nốt nhạc --> Fix bug cả bài nhạc 🐸
- Nghĩ tới đâu code tới đó --> Failed khi tích hợp toàn bộ flow.
- Test cho tính năng mới --> Bugs nơi khác trỗi dậy 🤣

--> Phần lớn lý do là phân tích chưa kỹ hoặc bỏ qua luôn bước này.

# Giải pháp là gì?

Giải quyết vấn đề theo từng giai đoạn một:

1. Nhận task và phân tích

Phân tích có 3 thứ:

- Yêu cầu của task, như thế nào là DONE. (định nghĩa test case)
- Kĩ thuật: làm sao code được như yêu cầu.
- Những phần bị ảnh hưởng (impact area).

HTML: Phân tích cây HTML

```
section.hero
div.container
h1.hero__title
```

```
section.about
  div.container
  div.row
    div.col-12
    h2.about__title
    p.about__description
    div.col-12
    img.about__thumbnail
```

#### **JAVASCRIPT**: function

- Validate inputs (important)
- Processing
- Output

#### JAVASCRIPT: flow

- Bao nhiêu bước?
- Mỗi bước cần hàm gì?
- Hàm có input, output gì?
- KQ cuối cùng là gì?

### ReactJS: component

- Container or Component
- Props
- State
- Render

### ReactJS: flow

- Routing / Nested routing
- Bao nhiêu bước
- Cần component gì?
- Dữ liệu được truyền giữa các bước.
- Nếu bị reload giữa chừng?
- Kết quả cuôi cùng?
- Phải xác định được bức tranh toàn cảnh trước khi bay vào code.

# 2. Bay vào code

- Bị ngu dần đều theo thời gian 🤣
- Nên phải dựa vào bản phân tích mà code.

# 3. Testing

- Test cái feature mới mình mới viết xong.
- Test thêm cả những impact area nữa.

# Nhớ nè

- Phần quan trọng nhất là phân tích.
- Không được bỏ qua với bất cứ lý do gì.
- Task nhỏ thì phân tích 1 phút.
- Task lớn thì phân tích 1 buổi / ngày / nhiều ngày.
- Vì mình sẽ ngu dần đều khi bay vào code 🤣

Phía trên là phần chia sẻ dựa theo kinh nghiệm cá nhân, có thể phù hợp hoặc chưa phù hợp, các bạn tham khảo thôi nhé! 💗

## HAPPY CODING!!!

Nhớ like, share và subscribe nếu thấy hay nha hehe.

Còn nếu muốn ủng hộ mình tiếp tục làm videos thì có thể donate 10k theo link trong phần mô tả video cho mình uống ly trà đá thượng hạng nhé 🥩