Twitter Thread by Hardik Pandya



Hardik Pandya
@hvpandya

■■■

Sharing the learnings I've gathered over the years working in Design, as one-liners.

Starting here ↓

If a state is important to be communicated on a screen, use multiple avenues to reinforce it. Don't mention it once and assumer users will read it where you've put it.

Users glance around and catch loose cues. Multiple avenues increase the odds of successful conveyance.

Take the Pointer approach to designing. Point to where the info is by spreading links across the UI, instead of repeating the same information across multiple screens.

(break this rule sparingly)

Not all screens will need the primary CTA. Don't use your loudest button first just because it's the default style you have in your system.

Modals & bottom sheets work better than painting a fresh page via a reload when conveying additional information.

Fresh pages put the burden of reorientation and of relearning a new screen layout on the user. Keep users on the same page as much as possible.

Consistency is a tool, not the outcome. It's mostly misused and leads to lazy design where designers stop thinking hard.

If you find yourself leaning to consistency while solving for a UI, know that there is most likely a better solution possible that you're oblivious to.

Simple design is hard. What feels simple to you is never that simple for the user.

Your context is vastly different than the user's.

Users aren't lazy. They just don't have patience for having to learn your way of doing something.

Do the extra work and simplify things for them.

Don't invent new words, don't unnecessarily 'brand' features.

It confuses the users and is often disrespectful to them.

Copywriting is everything.

Also, good copy isn't always the shortest.

Brevity is important but not at the cost of clarity.

Squint test is very useful.

If you see more than one 'attention blobs' when you squint, time to improve the UI.

Very few affordances are obvious. Use known patterns as much as possible.

User behaviour is years of learned and acquired muscle and causal memory.

Don't try and force-change it. Harness and use it to your advantage.

Don't reinvent navigation patterns. They won't be 'better' than what's available and understood at scale.

Users will *not* automatically understand your completely arbitrary colour-feature association. They have good associative understanding of Green and Red.

For all other colours, train the user gradually.

If a screen has more than 2 colours present on it, question why.

Give users an immediate feedback for every action they take on the UI.

This is mostly non-negotiable and yet often ignored.

Remember: The designer (and often also the PM) are the only people who see all of the app's screens daily, side by side, at once.

The user only sees 1 at a time with a causal memory of at most 2-3 screens prior.

Use consistency to guide your designs. Consistency isn't the goal in itself.

(I think I said this already but worth repeating...)

It's a bad idea to stick with a UI pattern that no longer works, just because that's how it's been done.

Question everything, keep updating.

Every design has a shelf life. More often than not, a very short one.

Assume that the users will never see or interact with what's out of sight.

Always save user's progress. It helps them trust your product more.

More often than not, Delight stems from saving users time, effort, money... sometimes all 3 at the same time.

If the user didn't initiate or expect a change, and the change was performed, inform the user and explain the change.

Your one-time onboarding will *not* teach the users anything. Users will most likely skip past your onboarding steps.

Design accordingly.

Remember: Users use your app for a fleeting moment in their lives, while living their lives and using a hundred other apps.

Keep the humility, solve their problem and get out of the way.

It’s okay to have messy Figma files. You’re not shipping them.

Defaults are everything. Think for the 80% of the users and what their defaults should be.

Don’t discard interesting design approaches by Edge-case Thinking. Edge-case Thinking ruins many good ideas before they can even have a chance to grow.

Build a great Happy Path experience first and then add edge-case handlers.

‘How fast the users can do something on your product’ is everything. On an infinite time-scale, even the shittiest apps will ‘work’. You don’t have that time.

No two features are equal and hence no design should be flat and linear. Modulate the weightage you give to the features with the importance and prominence they deserve.

A design without a philosophy and a story will not fly. Your design approach has to stand for something meaningful beyond the pixels.

Visual design is a mission critical skill. The entire success of your design work hinges on the quality of the visual design.

Go deep while creating, zoom out and detach while curating. You'll become a better designer. Don't create and curate at the same time.

Keep another designer handy at all times. They can be useful in gut-checking your ideas and in critiquing your work quickly.

Keep tap-free zones in your design. It helps the user rest their thumb on the screen without worrying about accidental taps.

Remember: Your design system is only as good as the last product problem it helped solve.

Your first attempt will always be nowhere close to where the solution needs to be. Keep iterating.

If you're getting feedback from users, peers or leadership, take it and learn. You're one of the lucky few who are getting feedback. Most don't.

Don't break the laws of physics when you design.

Users don't know about your templates, symbols, styles or design system. They care about the screen in front of them and if that screen takes them closer to their goal or not.