What is Interaction?

People interact with interfaces in a conversational way; their willingness to participate in the exchange is driven by the product's ability to respond. The design language of the product is a built-in personality that's perceived in all interactions. Every aspect of the design contributes to the design language, think of it like the speed and the rhythm of a dance. Ballroom dancing is formal with precise footwork and elegant movements, a samba is energetic and rhythmic, both have formality and structure, but it's up to the dancers to react to the style of their partner. How do we let customers feel that they are part of the conversation, engaged in the dance?

Design is communication.

Communicating with customers is an attempt to relate to them – it's not about winning them over, it's about building rapport. The conversation is initiated when someone encounters the product for the first time, and the product extends its hand and invites them onto the floor. The personality of the system is recognized in these first few moments. The product reflects its values, and the customer becomes aware of how the product sees itself. "Yes, I'd love to dance," they think.

Customers continue the interaction when the business proposition meets their expected needs. Designers deploy a hierarchy of information that makes the product offering clear and drives the customer towards product acceptance. Show off your moves with a catchy headline, product photos, or demo videos, accompanied by engaging copywriting, pull-quotes, and testimonials. These elements are metric driven design decisions that capture customer attention.

Product acceptance is realized by "decision moments" which assume the pattern of a call to action, often a conspicuous "BUY NOW" button – someone's going home with you tonight, *wink*.

Interaction is the point at which the conversation can be considered successful or unsuccessful.

People by nature are interactive, they can give and receive instant feedback. People will expect your system to provide the following: all the information required to interact, assistance in recovering from error, and the ability to take the lead. If the system doesn't respond as expected, or an error cannot be

recovered from, the system is considered broken. Besides, there's really only one place where you don't receive any feedback, and that's purgatory.

static→reactive→interactive

- ➔ The more control a person has, the more interactive the system will feel.
- ➔ The more feedback the system gives, the more interactive the system will appear.

Take the simple act of sending an email, for example. Users engage in a sequence of interactions, and the system responds in accordance with user actions.

*Reading this next section takes moderate concentration

Customer → System(s)

1. Find and click the "email" application icon or link → email application opens
2. Select the "New" button → an instance of the "compose new mail" component appears
3. Select the "To:" input field → the input field gains focus and its style changes
4. Enter "Jack@lumber.com" → list of contacts appears matching the users input
5. Select message area → a validation rule checks that the email address is well formatted upon leaving the "To:" field (the blur state)
6. Start Typing → Suggested intro text is populated, "Hello Jack,..." + *instructions to hit tab*
7. Enter the message → Text is auto saved
8. All sorts of editing, formatting, thinking → rich text editor does its many things
9. Click on the Send button → a confirmation or error message appears as servers react
10. Navigates back to inbox → sent email appears in "sent" folder

The interface guides the customer towards their specific goals using its discreet capabilities. Trust is built if the required steps for completing a task are clear, and the customer recognizes that each action was acknowledged by the system. System feedback for digital products is often a shift in the visual style of the interface; the border color of a text input when it's selected, or button states like *hover* or *active* are all part of the interaction. The customer is motivated to continue because they are confident that the system is "listening" based on these visual cues.

People treat computers like people.

—Clifford Nass, and Byron Reeves (*The Media Equation 1996*)

When visual cues fall short, words become the bridge. Users rely on the system's writing style, its *voice* and *tone* to determine meaning and to form an understanding of the system itself. The system's voice is a consistent, overarching trait, offering users a familiar guide. The tone is an adaptation of the voice to various situations.

Maintaining a consistent voice ensures a recognizable personality across interactions. Tone, drawing cues from voice, flexibly navigates diverse situations. For example, when designing a system that helps users book an appointment, you might craft a voice that is friendly and helpful. The tone of messages will be more serious when the user is about to make a payment, and more lighthearted when the user is confirming their appointment.

A professional setting will have a different voice and tone than a casual one. Consider the target audience when crafting a writing style. For example, a system designed for a bar restaurant will have a fun and inviting voice compared to a more serious one for a law firm.

Crafting a comprehensive writing style-guide often falls to the designer. Ensuring the system's personality remains reliable and cohesive enhances the user experience because it becomes predictable. When users know what to expect from the system, it helps them feel confident while using it.

Creating a consistent voice and tone for your system

- ➔ Use short sentences with clear and concise language
- ➔ Avoid jargon and technical terms
- ➔ Be consistent in your use of capitalization, punctuation, and grammar
- ➔ Use an active voice "Our product enhances experiences..."
- ➔ Avoid using the second person (you)

✅ **Success. Congratulations brave warrior.**

❌ **Error. Uh-oh, look out for those booby-traps.**

Lan Pham

Novel interactions are not required for a product to capture an audience – instead use common interaction patterns in intelligent ways. People prefer interactions that are similar to systems they have encountered before and do not prefer to learn new ways of doing things; this is called familiarity bias. For example, email clients like Gmail and Outlook adopt a similar layout and organization structure. Users expect to find their inbox and sent-items in a specific location, and want access to standard features like [accidental] reply-all and [rage] forwarding emails. This "copy-cat" approach reduces the learning curve for new users and allows them to focus on their tasks.

An interaction is considered authentic if it matches the naturalness and spirit of human interaction. In the process of communication, be clear and succinct to be understood. Concise messaging with intention makes a good impression. Be direct, and deliver your message in as few words as possible.

listen intently→think quickly→speak well

Enhanced interaction

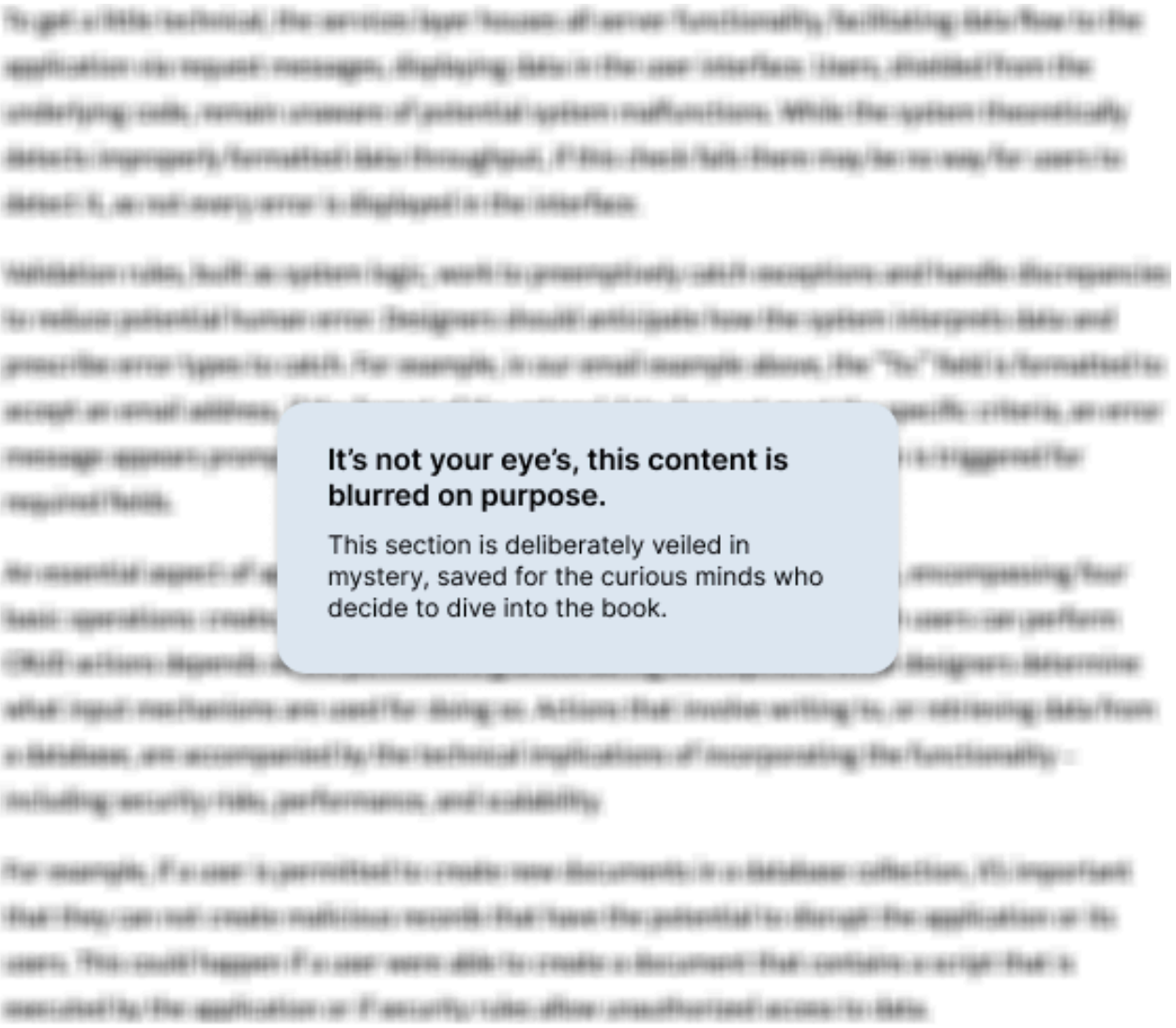
Instruments/capabilities at your disposal (in many mobile devices) that facilitate interaction

- ➔ A full color screen with multi-touch capabilities
- ➔ microphone and speakers
- ➔ global positioning system (GPS)
- ➔ accelerometer and magnetometer
- ➔ step-counter
- ➔ haptics (vibration)
- ➔ gyroscope (orientation of phone in space)
- ➔ camera(s) with ambient light sensor
- ➔ internal storage
- ➔ Bluetooth

Learn more about web and hardware APIs at
https://www.w3.org/TR/?tags[0]=webapi

Advanced Digital UI Interactions

The apparent simplicity of an interface conceals the underlying complexity. While users need not concern themselves with the intricate code running behind the scenes to operate the application, designers play a crucial role in comprehending the system's inner workings.



Additionally, it's important to consider the performance of the database. If the system is unable to handle increased synchronous access to the database, it risks crashing the application. It's important to design CRUD operations in a way that scales as the application grows. For instance, consider how often the application is required to read from one or many databases based on number of users, database structures, and interface requirements.

Interaction rehashed

- ➔ Free flowing creativity, productivity, and person-to-person communication lend interactivity
- ➔ Community, adaptability, and personalization lead to greater interactivity
- ➔ Visual states of the interface are key to the conversation between the system and user
- ➔ Voice and tone act as the personality of the interactive system
- ➔ Interfaces abstract away the complexity behind them

This is a preview chapter from the book

Your Cup of Tea : Design for Experience

To read the book in its entirety please visit

https://dlightning.org/tea-landing.html

Thank you.

Dlightning⚡

Dlightning.org

© 2024