**Tìm hiểu về User & usability**

# I. User

## 1. Introduction (Giới thiệu)

* Khi thiết kế giao diện, mục tiêu chính là giúp người dùng dễ sử dụng, dễ học, ít mắc lỗi và đạt được mục đích nhanh nhất.
* Một hệ thống có thiết kế tốt là hệ thống phù hợp với thói quen, khả năng và mong đợi của người dùng.

#### Ví dụ tốt: Trang chủ của Google

* Giao diện đơn giản: Chỉ có một ô tìm kiếm ở trung tâm màn hình.
* Màu sắc rõ ràng: Logo Google nhiều màu sắc giúp nhận diện thương hiệu.
* Chức năng dễ hiểu: Người dùng chỉ cần nhập từ khóa và bấm Enter để tìm kiếm.
* Không bị phân tâm: Không có nhiều nút bấm hoặc quảng cáo làm mất tập trung.

#### Ví dụ kém: Trang web của một ngân hàng cũ

* Quá nhiều thông tin: Trang chủ có hàng chục nút bấm với các kích thước khác nhau.
* Chức năng không rõ ràng: Người dùng không biết phải vào đâu để kiểm tra số dư tài khoản.
* Thiếu sự hướng dẫn: Không có thông báo hay gợi ý khi người dùng nhập sai thông tin đăng nhập.

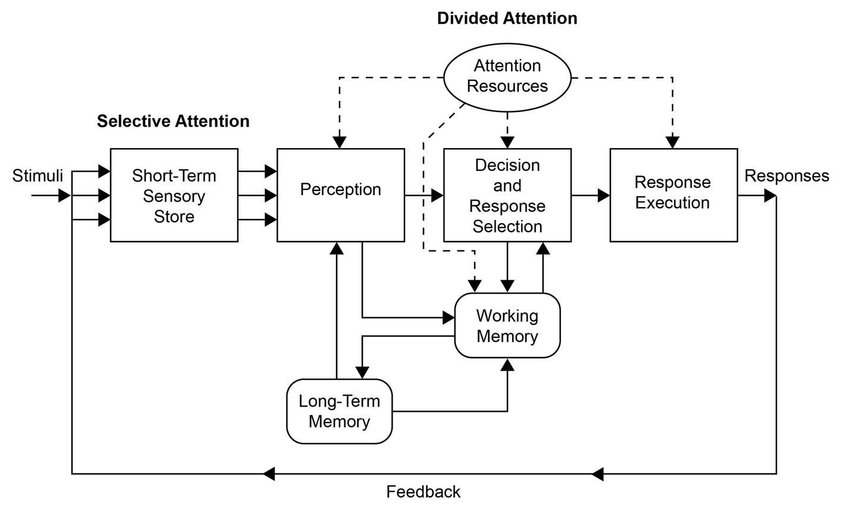
💡 Bài học: Giao diện tốt giúp người dùng hiểu ngay lập tức mà không cần học hay tìm kiếm hướng dẫn.

## 2. Human Information Process Model (Mô hình xử lý thông tin của con người)

**Mô hình này gồm 3 giai đoạn chính:**

1. Nhận đầu vào (Input)
   * Người dùng tiếp nhận thông tin qua mắt (nhìn), tai (nghe), tay (chạm).
2. Xử lý thông tin (Processing)
   * Não bộ phân tích dữ liệu, so sánh với những gì đã biết và quyết định hành động.
3. Phản hồi đầu ra (Output)
   * Người dùng thực hiện hành động như bấm chuột, nhập văn bản, vuốt màn hình…

### 



#### Ví dụ tốt: Nút "Đặt món ngay" trên một ứng dụng đặt đồ ăn

* Nhận đầu vào: Người dùng nhìn thấy một danh sách món ăn hấp dẫn.
* Xử lý thông tin: Họ đọc giá, hình ảnh và đánh giá để quyết định chọn món.
* Phản hồi đầu ra: Họ nhấn vào nút "Đặt món ngay" và chọn phương thức thanh toán.

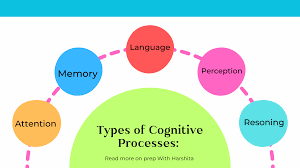
#### Ví dụ kém: Hệ thống đặt vé xe có giao diện rối rắm

* Nhận đầu vào: Người dùng thấy quá nhiều thông tin (giờ xe, giá vé, loại xe, chính sách hoàn vé…).
* Xử lý thông tin: Họ mất nhiều thời gian để tìm đúng chuyến xe mong muốn.
* Phản hồi đầu ra: Do giao diện khó hiểu, họ có thể bấm nhầm chuyến xe, dẫn đến đặt vé sai.

💡 Bài học: Hệ thống phải giúp người dùng xử lý thông tin một cách nhanh chóng và rõ ràng, tránh làm họ bối rối.

## 3. Cognitive Process (Quá trình nhận thức)

Quá trình này giúp người dùng hiểu, nhớ và học cách sử dụng hệ thống. Nó gồm 4 yếu tố chính:



### 1️⃣ Chú ý (Attention)

* Người dùng không thể tập trung vào tất cả mọi thứ trên màn hình, họ chỉ chú ý đến những gì nổi bật hoặc cần thiết.

#### Ví dụ tốt: Nút "Mua ngay" trên Shopee

* Màu sắc nổi bật (đỏ cam) giúp người dùng dễ nhận ra.
* Kích thước lớn và nằm ngay dưới hình ảnh sản phẩm.
* Không có quá nhiều yếu tố gây nhiễu xung quanh.

#### Ví dụ kém: Quảng cáo pop-up xuất hiện liên tục

* Khi người dùng đang đọc bài viết, một pop-up quảng cáo bật lên che mất nội dung.
* Điều này gây gián đoạn, làm giảm sự tập trung, thậm chí khiến họ thoát khỏi trang web.

### 2️⃣ Nhận thức (Perception)

* Người dùng nhận biết thông tin qua các giác quan như thị giác (màu sắc, hình ảnh), thính giác (âm thanh thông báo).

#### Ví dụ tốt: Biểu tượng thùng rác trên Windows

* Biểu tượng thùng rác giúp người dùng dễ dàng hiểu đây là nơi lưu trữ file đã xóa.
* Nếu thùng rác đầy, biểu tượng thay đổi để thể hiện rõ trạng thái.

#### Ví dụ kém: Biểu tượng khó hiểu trên phần mềm chỉnh sửa ảnh

* Một phần mềm có biểu tượng hình con mắt bị gạch chéo, nhưng người dùng không hiểu ý nghĩa của nó (ẩn/hiện layer).
* Do đó, họ mất thời gian để tìm hiểu hoặc bấm thử từng nút.

### 3️⃣ Ghi nhớ (Memory)

* Người dùng ghi nhớ giao diện và các thao tác để sử dụng dễ dàng hơn trong tương lai.

#### Ví dụ tốt: Facebook giữ nguyên vị trí nút "Thích" và "Bình luận"

* Người dùng quen thuộc với bố cục nên không phải học lại khi cập nhật phiên bản mới.

#### Ví dụ kém: Một ứng dụng thay đổi vị trí nút "Gửi tin nhắn" sau mỗi lần cập nhật

* Người dùng phải tìm lại từ đầu, gây khó chịu và mất thời gian.

### 4️⃣ Học hỏi (Learning)

* Người dùng học cách sử dụng hệ thống thông qua thực hành, hướng dẫn, hoặc khám phá.

#### Ví dụ tốt: Canva hướng dẫn người dùng bằng tooltip

* Khi lần đầu mở ứng dụng, Canva hiển thị tooltip (chú thích ngắn) chỉ cách sử dụng các công cụ.
* Người dùng học nhanh hơn mà không cần đọc tài liệu dài dòng.

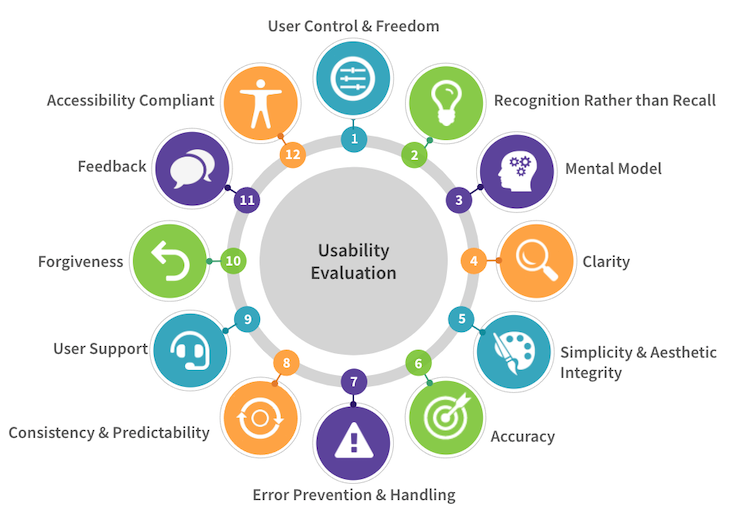
#### Ví dụ kém: Phần mềm kế toán có giao diện phức tạp, không có hướng dẫn

* Người dùng phải tự mò mẫm hoặc tìm tài liệu hướng dẫn, làm mất thời gian và dễ mắc lỗi.

# II. Usability

## **1. Concept (Khái niệm Usability)**

* **Usability (Khả năng sử dụng)** là một thuộc tính chất lượng đánh giá mức độ **dễ học, dễ sử dụng và mức độ hài lòng** của người dùng khi sử dụng sản phẩm.
* Một sản phẩm có tính Usability cao giúp người dùng:
  + **Hoàn thành nhiệm vụ hiệu quả** mà không gặp khó khăn.
  + **Giảm lỗi và sai sót** khi thao tác.
  + **Cảm thấy thoải mái và hài lòng** khi sử dụng.



### 

✅ **Ví dụ tốt: Giao diện của iPhone**

* **Thiết kế đơn giản**, dễ hiểu ngay cả với người lần đầu dùng.
* **Cử chỉ vuốt trực quan**, không cần nhớ nhiều nút bấm.
* **Phản hồi nhanh chóng** khi chạm vào màn hình.

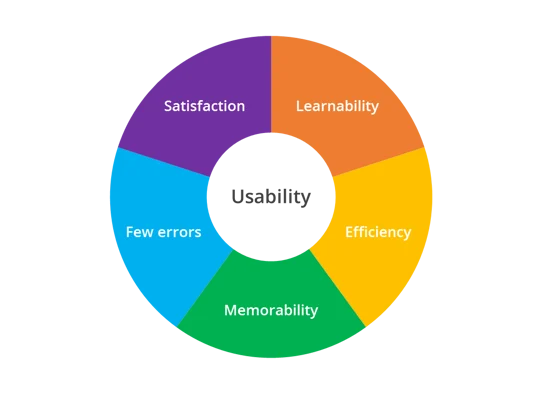
❌ **Ví dụ kém: Điều khiển từ xa có quá nhiều nút**

* **Quá nhiều nút bấm nhỏ**, khiến người dùng khó tìm nút cần thiết.
* **Không có bố cục rõ ràng**, làm người dùng dễ bấm nhầm.

💡 **Bài học:** Sản phẩm cần **trực quan, dễ hiểu, dễ sử dụng** ngay từ lần đầu tiên.

## **2. Usability Components (Các thành phần của Usability)**

Một hệ thống Usability tốt cần đảm bảo các thành phần sau:



### **1️⃣ Effectiveness (Hiệu quả)**

* Người dùng có thể **hoàn thành công việc đúng và đầy đủ** không?

✅ **Ví dụ tốt: Google Docs**

* **Tự động lưu nội dung**, giúp người dùng không lo mất dữ liệu.
* **Công cụ chỉnh sửa dễ dùng**, hỗ trợ cộng tác nhóm.

❌ **Ví dụ kém: Hệ thống quản lý nhân sự lỗi thời**

* Nhập dữ liệu **khó khăn, nhiều bước không cần thiết**.
* Người dùng dễ **mắc lỗi khi thao tác**, làm giảm hiệu quả công việc.

💡 **Bài học:** Giao diện cần giúp người dùng **hoàn thành công việc dễ dàng và chính xác**.

### **2️⃣ Efficiency (Hiệu suất)**

* Người dùng có thể **hoàn thành công việc nhanh chóng** không?

✅ **Ví dụ tốt: Mua sắm trên Shopee**

* **Giao diện tối ưu hóa**, chỉ mất vài giây để mua hàng.
* **Tích hợp thanh toán nhanh**, không cần nhập lại thông tin nhiều lần.

❌ **Ví dụ kém: Hệ thống đăng ký dịch vụ công chậm chạp**

* **Quá nhiều bước**, yêu cầu tải xuống và điền biểu mẫu thủ công.
* **Thời gian xử lý lâu**, người dùng dễ chán nản.

💡 **Bài học:** Giảm thiểu số bước không cần thiết để **tăng tốc độ sử dụng hệ thống**.

### **3️⃣ Safety (An toàn)**

* Hệ thống có giúp **ngăn chặn lỗi** và **bảo vệ dữ liệu** không?

✅ **Ví dụ tốt: Gmail có nút "Hoàn tác gửi"**

* Nếu gửi nhầm email, người dùng có thể **bấm "Undo" trong vài giây** để thu hồi.

❌ **Ví dụ kém: Không có tính năng "Lưu nháp" khi nhập thông tin dài**

* Người dùng nhập một biểu mẫu dài, nhưng **mất hết dữ liệu nếu trình duyệt bị lỗi**.

💡 **Bài học:** Hệ thống nên có tính năng **undo, auto-save, xác nhận trước khi thực hiện hành động quan trọng**.

### **4️⃣ Learnability (Dễ học)**

* Người dùng có thể **học cách sử dụng nhanh chóng** không?

✅ **Ví dụ tốt: Canva cung cấp hướng dẫn bằng hình ảnh**

* Khi mở Canva lần đầu, người dùng sẽ thấy **tooltip hướng dẫn ngắn gọn**.
* Có **mẫu thiết kế sẵn**, giúp họ làm quen nhanh hơn.

❌ **Ví dụ kém: Phần mềm kế toán phức tạp, không có hướng dẫn**

* Người mới sử dụng **không biết bắt đầu từ đâu**.
* Giao diện **rối rắm, không có hướng dẫn trực tiếp**.

💡 **Bài học:** Hãy giúp người dùng **học nhanh bằng hướng dẫn đơn giản và dễ hiểu**.

### **5️⃣ Memorability (Dễ nhớ)**

* Nếu người dùng quay lại sau một thời gian, họ có thể **nhớ cách sử dụng ngay không?**

✅ **Ví dụ tốt: Facebook giữ nguyên vị trí các nút chính**

* Nút **Like, Comment, Share** luôn ở một vị trí cố định.

❌ **Ví dụ kém: Giao diện thay đổi vị trí nút quan trọng mỗi lần cập nhật**

* Người dùng phải **tìm lại từ đầu**, gây khó chịu.

💡 **Bài học:** Giao diện cần **nhất quán và dễ nhớ** để người dùng không phải học lại từ đầu.

### **6️⃣ Satisfaction (Sự hài lòng của người dùng)**

* Người dùng có **cảm thấy thoải mái và thích thú** khi sử dụng hệ thống không?

✅ **Ví dụ tốt: Instagram có thiết kế đẹp, dễ sử dụng**

* Giao diện **đơn giản, hình ảnh đẹp, dễ thao tác**.

❌ **Ví dụ kém: Website chậm, nhiều quảng cáo bật lên**

* Người dùng cảm thấy **khó chịu và có thể rời bỏ trang web ngay lập tức**.

💡 **Bài học:** Thiết kế tốt giúp người dùng **cảm thấy hài lòng và quay lại sử dụng**.

## **3. Usability Design Guidelines (Hướng dẫn thiết kế Usability)**

Để cải thiện Usability, có 3 bộ nguyên tắc quan trọng:

### **1️⃣ Donald A. Norman – 6 Nguyên tắc thiết kế**

| **Nguyên tắc** | **Ý nghĩa** | **Ví dụ tốt** | **Ví dụ kém** |
| --- | --- | --- | --- |
| **Visibility (Hiển thị rõ ràng)** | Người dùng dễ dàng thấy những gì cần làm. | Nút "Đăng ký" lớn, màu nổi bật. | Nút quan trọng bị ẩn hoặc quá nhỏ. |
| **Affordance (Hành động dễ đoán)** | Thiết kế gợi ý rõ ràng về cách sử dụng. | Biểu tượng thùng rác -> Xóa. | Biểu tượng khó hiểu, người dùng không biết chức năng. |
| **Constraint (Ràng buộc hợp lý)** | Giới hạn những thao tác sai. | Chỉ nhập số vào ô "Số điện thoại". | Không giới hạn, người dùng nhập chữ vào ô số. |
| **Mapping (Ánh xạ hợp lý)** | Liên kết trực quan giữa điều khiển và kết quả. | Volume tăng -> Âm thanh to hơn. | Điều chỉnh volume nhưng không thấy thay đổi. |
| **Feedback (Phản hồi rõ ràng)** | Hệ thống phản hồi ngay sau thao tác. | Bấm "Gửi" -> Thông báo "Email đã gửi". | Không có thông báo, người dùng không biết email đã gửi hay chưa. |
| **Consistency (Tính nhất quán)** | Giữ bố cục, màu sắc đồng nhất. | Google giữ nguyên vị trí thanh tìm kiếm. | Giao diện thay đổi vị trí nút mỗi lần cập nhật. |

### **2️⃣ Shneiderman – 8 Quy tắc vàng**

1. Sự đồng nhất quan trọng trong *8 nguyên tắc vàng trong thiết kế giao diện*

Các thông báo nên được sử dụng thống nhất ở menu, màn hình trợ giúp, popup và thống nhất trong các mệnh lệnh xuyên suốt trang web.

  
2. Cho phép những người dùng quen thuộc có thể sử dụng phím tắt

-Khi việc sử dụng trở nên thường xuyên, người dùng muốn giảm số lần nhấp, click để rút ngắn các bước tương tác. Lúc này phím tắt, lệnh ẩn và các hỗ trợ nhỏ rất cần thiết.

3. Cung cấp phản hồi đầy đủ

-Cần xuất hiện mọi phản hồi trong hoạt động của giao diện với người dùng. Đối với những hành động liên tục và nhỏ các phẩn hồi có thể đơn giản, nhưng với các hành động không thường xuyên và các thao tác quan trọng, các phản hồi cần rõ ràng.

  
4. Các hộp thoại nên được đặt gần nhau

-Các chuỗi hành động nên được xếp chung nhóm với thứ tự: ban đầu, ở giữa và kết thúc. Các thông tin phản hồi ở mỗi nhóm khi hoàn thành cần được cung cấp để tạo cảm giác nhẹ nhõm cho người dùng, để họ sẵn sàng để đi tới nhóm hành động kế tiếp.

5. Đưa ra một lỗi dễ sửa chữa

-Thiết kế giao diện tốt là khi để người dùng không thể tạo một lỗi nghiêm trọng. Nếu có một lỗi được tạo ra, hệ thống cần xác định được lỗi và đưa ra một giải pháp đơn giản, dễ hiểu để xử lý lỗi. Đây là yếu tố rất quan trọng trong *8 nguyên tắc vàng trong thiết kế giao diện.*

6. Cho phép trở lại dễ dàng

-Tính năng này giúp người dùng không thấy lo sợ khi họ tạo ra một lỗi. Nó sẽ giúp người dùng tự tin khám phá những chức năng lạ lẫm trên website. Tính năng quay ngược có thể là một bước đơn giản, một bước trở lại hay một nhóm các hành động.

7. Hỗ trợ kiểm soát nội bộ

-Những người sử dụng có kinh nghiệm rất mong muốn cảm giác rằng họ đang phụ trách hệ thống và hệ thống phản ứng lại hành động của họ. Hãy thiết kế giao diện khiến cho người sử dụng khởi tạo các hành động chứ không phải chỉ dùng để phản hồi.

8. Giảm bộ nhớ ngắn hạn (short-term memory)

-Các thiết kế hiển thị cần đơn giản. Hiển thị kiểu multiple page nên có gắn kết vững chắc, các chuyển động của cửa sổ cần được giảm bớt, những thứ cần học phải được chia nhỏ, các hành động cần có trình tự để người dùng có thể dễ dàng ghi nhớ và thực hiện.

### **3️⃣ Jakob Nielsen – 10 Nguyên tắc Usability**

### **1. Khả năng hiển thị trạng thái hệ thống**

-Để tránh nhầm lẫn và thất vọng, việc thông báo cho người dùng về vị trí của họ trong một quy trình và những gì đang diễn ra đòi hỏi phản hồi phù hợp, thường là ngay lập tức (Nielsen gợi ý 'trong thời gian hợp lý'). Khi chúng tôi hành động, chúng tôi mong đợi sản phẩm phản ứng; nếu kết quả không rõ ràng, phản hồi xác nhận rằng mọi thứ đang đi đúng hướng.

-Với những cập nhật về trạng thái hệ thống, chúng tôi cảm thấy an toàn và kiểm soát được tình hình, xây dựng niềm tin vào hệ thống và thương hiệu.

**Ví dụ về khả năng hiển thị trạng thái hệ thống:**

* -Hộp thoại Thêm vào giỏ hàng hoặc biểu tượng giỏ hàng đã cập nhật
* -Hoạt ảnh xác nhận và một số [tương tác nhỏ](https://ux247.com/microinteraction-examples/)
* -Xác nhận đăng ký nhận bản tin
* -Thanh tiến trình và thanh tải và cập nhật thời gian
* -Đường dẫn để cho người dùng biết họ đang ở đâu trong phần điều hướng và nội dung của trang web

### **2. Sự phù hợp giữa hệ thống và thế giới thực**

-Để làm cho sản phẩm trực quan và yêu cầu người dùng ít phải học hơn, chúng tôi sử dụng các yếu tố mà người dùng đã nhận ra, thường bắt chước hoặc đại diện cho các mục từ thế giới thực. Điều đó có nghĩa là cung cấp các manh mối trực quan hoặc bỏ thuật ngữ kỹ thuật để nói ngôn ngữ của người dùng và sử dụng các cụm từ và khái niệm quen thuộc với họ theo thứ tự tự nhiên và hợp lý.

-Khi chúng ta tuân theo các quy ước trong thế giới thực, điều này sẽ giúp xây dựng một hệ thống mà người dùng đã quen thuộc; chúng tôi gọi đây là *lập bản đồ tự nhiên* .

**Ví dụ về các quy ước trong thế giới thực:**

* -Biểu tượng đại diện cho hành động: thùng rác, phong bì email, kính lúp tìm kiếm, v.v.
* -Các trang chuyển trong ứng dụng đọc sách điện tử
* -Sử dụng màu xanh lá cây để đi hoặc tốt và màu đỏ để dừng hoặc như một cảnh báo lỗi
* -Chụm để phóng to và vuốt để di chuyển trang hoặc màn hình

### **3. Kiểm soát và tự do của người dùng**

-Do các sản phẩm kỹ thuật số hiếm khi đi kèm hướng dẫn hoặc sổ tay hướng dẫn đầy đủ, chúng ta cần tùy chọn khôi phục khi nhấn nhầm nút hoặc mắc lỗi khác. Nielsen gọi đây là *'lối thoát khẩn cấp'* , cho phép người dùng quay lại đúng hướng mà không cần thực hiện quy trình dài để quay lại điểm xuất phát ban đầu.

**Ví dụ về quyền kiểm soát và tự do của người dùng:**

* -Chức năng hoàn tác và làm lại
* -Nút quay lại
* -Hủy bỏ tùy chọn
* -Đường dẫn điều hướng
* -Nhấp chuột phải hoặc menu phụ trong sản phẩm
* -Các thông báo lỗi được xây dựng cẩn thận để hướng dẫn người dùng quay lại -đúng hướng

**Kế tiếp**

### **4. Sự nhất quán và tiêu chuẩn**

-Một lĩnh vực khác mà chúng ta có thể giảm quá trình học của người dùng là tuân theo các mẫu và tiêu chuẩn của các sản phẩm và nền tảng hiện có và các mẫu và tiêu chuẩn từ thế giới thực. Có một lý do khiến hầu hết các giao diện người dùng trông giống nhau; việc phát minh lại bánh xe buộc người dùng phải thực hiện thêm các bước mà họ không cần phải học—duy trì tính nhất quán và các tiêu chuẩn mà người dùng mong đợi sẽ tiết kiệm thời gian và sự thất vọng.

-Có hai loại tính nhất quán: loại thứ nhất, trong sản phẩm (được gọi là **tính nhất quán nội bộ** ) và loại thứ hai, theo quy ước của ngành ( **tính nhất quán bên ngoài** ).

**Ví dụ về tính nhất quán và tiêu chuẩn:**

* -Sử dụng các biểu tượng được ngành công nghiệp chấp nhận nhưng theo phong cách thương hiệu sản phẩm
* -Tuân thủ các tiêu chuẩn về nhãn mác và điền mẫu
* -Thực hiện theo định dạng mua hàng/đăng ký/điều hướng mà họ đã hiểu

### **5. Phòng ngừa lỗi**

-Có nhiều cách để giải quyết lỗi hơn là cung cấp cho người dùng một lối thoát hoặc kế hoạch thoát hiểm. Lý tưởng nhất là một giao diện người dùng được cân nhắc kỹ lưỡng sẽ giúp người dùng tránh được những lỗi có thể xảy ra hoặc có khả năng xảy ra. Nhiều lỗi xảy ra do sự bất cẩn hoặc thiếu chú ý, nhưng những lỗi khác là do quy trình phức tạp hoặc thiếu hướng dẫn.

-Việc cung cấp thông tin và hướng dẫn cần thiết có thể giảm thiểu lỗi và loại bỏ các điều kiện dễ xảy ra lỗi; việc thông báo cho người dùng về nơi có thể xảy ra lỗi có thể giúp họ tiết kiệm thời gian và tránh gây thất vọng.

**Ví dụ về phòng ngừa lỗi:**

* -Cảnh báo và hành động giúp tránh lỗi và thông báo lỗi
* -Cung cấp hướng dẫn bằng ngôn ngữ đơn giản hoặc hướng dẫn trực quan hoặc đồ họa dễ hiểu
* -Thêm gợi ý vào các trường biểu mẫu
* -Gợi ý chính tả trong các trường biểu mẫu
* -Tùy chọn xác nhận trường biểu mẫu (sử dụng các trường nhập lặp lại), ví dụ, cho địa chỉ email và mật khẩu
* -Tùy chọn xác nhận khi xóa vĩnh viễn các tệp, tin nhắn email và thông tin quan trọng khác

### **6. Nhận biết thay vì nhớ lại**

-Để giúp giảm thiểu tải bộ nhớ của người dùng và tối đa hóa khả năng suy nghĩ của họ, một UX chất lượng sẽ đưa ra các gợi ý thay vì mong đợi họ nhớ thông tin chung hoặc đã học trước đó.

**Ví dụ về sự nhận biết thay vì nhớ lại:**

* -Lựa chọn trường biểu mẫu được xác định trước (quốc gia, ngày tháng, nghề nghiệp, v.v.)
* -Tự động điền truy vấn tìm kiếm
* -Hướng dẫn từng bước trong quá trình thực hiện
* -Các mục đã xem gần đây
* -Danh bạ gần đây trong điện thoại và ứng dụng nhắn tin
* -Tính năng 'tiếp tục' hoặc phần 'chương trình của tôi' trên các dịch vụ phát trực tuyến

### **7. Tính linh hoạt và hiệu quả sử dụng**

-Mặc dù chúng tôi không muốn làm người dùng mới bối rối với quá nhiều thông tin, nhưng việc giúp người dùng có kinh nghiệm sử dụng phần mềm hoặc sản phẩm hiệu quả hơn sẽ mang lại UX tốt hơn nhiều.

-Giống như trong cuộc sống thực, chúng ta thích một mẹo hoặc phím tắt giúp sản phẩm dễ sử dụng hơn hoặc giúp chúng ta đạt được mục tiêu nhanh hơn. Điều này phát triển khả năng kiểm soát của người dùng và cho phép người dùng chuyên gia điều chỉnh các hành động thường xuyên theo nhu cầu cụ thể của họ, mang lại UX thỏa mãn hơn thông qua ít nỗ lực nhận thức hơn.

**Ví dụ về tính linh hoạt và hiệu quả sử dụng:**

* -Giới thiệu các gợi ý *tăng tốc* , ví dụ như phím tắt và cử chỉ chạm
* -Cung cấp các tùy chọn cá nhân hóa để người dùng có thể truy cập trực tiếp vào nội dung hoặc chức năng cần thiết
* Cho phép tùy chỉnh để họ có thể quyết định cách sản phẩm hoạt động cho họ

### **8. Thiết kế thẩm mỹ và tối giản**

-Thiết kế tối giản không chỉ là trông bóng bẩy; mà còn là cung cấp giá trị thực sự. Mọi mục hoặc thành phần không cần thiết đều cạnh tranh sự chú ý của người dùng, khiến trang web hoặc sản phẩm kém hiệu quả hơn.

-Khi nói đến phương pháp tìm kiếm nội dung và khả năng sử dụng, ít hơn là nhiều hơn. Tuy nhiên, điều đó không có nghĩa là bạn xóa mọi thứ để tạo một trang sạch sẽ, rõ ràng—chỉ xóa những phần không mang lại giá trị và làm người dùng mất tập trung khỏi những gì quan trọng.

**Ví dụ về thiết kế thẩm mỹ và tối giản:**

* -Trang chủ của Google có lẽ là một trong những sản phẩm tối giản dễ nhận biết nhất trên Internet
* -Biểu tượng điều hướng đơn giản ở cuối màn hình trong ứng dụng di động
* Ứng dụng máy tính chuyển từ chức năng cơ bản sang đầy đủ tính năng khoa học khi xoay thiết bị

### **9. Giúp người dùng nhận ra, chẩn đoán và phục hồi sau lỗi**

-Sai sót và lỗi là điều không thể tránh khỏi. Có lối thoát khẩn cấp là một cách để giải quyết vấn đề, nhưng thông báo lỗi tốt cho biết vấn đề xảy ra như thế nào và hướng dẫn cách quay lại đúng hướng sẽ tốt hơn. Giúp người dùng hiểu vấn đề giúp họ tránh mắc lại lỗi tương tự trong khi xây dựng sự tự tin của họ.

**Ví dụ về chức năng phục hồi lỗi:**

* -Lỗi đăng nhập: giải thích điều kiện nào không được đáp ứng; ví dụ: 'không có tài khoản nào như vậy' hoặc 'mật khẩu không khớp'.
* -Tạo mật khẩu: bao gồm các điều kiện mà người dùng phải đáp ứng, ví dụ, số lượng ký tự tối thiểu, số, chữ hoa hoặc ký tự đặc biệt.

### **10. Trợ giúp và tài liệu**

-Hầu hết các ứng dụng và sản phẩm kỹ thuật số đều không cần hướng dẫn nhiều, nhưng người dùng thường cần hướng dẫn đầy đủ và rõ ràng cho các hệ thống và sản phẩm phức tạp hơn.

-Nielsen cho rằng tài liệu trợ giúp phải dễ tìm kiếm và tập trung vào nhiệm vụ của người dùng; tài liệu cần ngắn gọn, cung cấp các bước cụ thể dưới dạng danh sách và lý tưởng nhất là trình bày ở nơi người dùng cần.

**Ví dụ về trợ giúp và tài liệu:**

* -Hướng dẫn sử dụng
* -Trung tâm trợ giúp
* -Câu hỏi thường gặp
* -Trò chuyện hỗ trợ
* -Hướng dẫn sử dụng
* -Chú giải công cụ