**THIẾT KẾ GIAO DIỆN**

* **Usability (Khả năng sử dụng):** Làm thế nào để người dùng có thể sử dụng tốt các chức năng của hệ thống

Usability được gọi theo nhiều cách khác nhau:

* *Computer Human Interaction (CHI):* Tương tác giữa người và máy
* *Human Computer Interaction (HCI):* Tương tác giữa máy và người
* *Man Machine Interface (MMI):* Giao diện người – máy
* *User System Interface (USI):* Giao diện hệ thống người dùng
* *User Interface (UI):* Giao diện người dùng
* *Graphical User Interface (GUI):* Giao diện đồ hoạ người dùng
* *Soldier Machine Interface (SMI)*
* *Human Computer Communication (HCC):* Giao tiếp người – máy
* *Operator Interface (OI):* Giao diện điều hành

Usability gồm:

* *Learnability (Khả năng học hỏi):* Làm thế nào để người dùng có thể học hỏi và sử dụng giao diện một cách nhanh chóng
* *Efficiency (Tính hiệu quả):* Làm thế nào để người dùng thực hiện các công việc của mình một cách nhanh chóng khi sử dụng giao diện (UI)
* *Memorability (Khả năng ghi nhớ):* Làm thế nào để người dùng có thể ghi nhớ việc thực hiện các thao tác trên giao diện một cách dễ dàng
* *Errors (Lỗi):* Làm thế nào để hệ thống có thể phục hồi lại khi gặp lỗi
* *Satisfaction (Khả năng hài lòng):* Người dùng có hài lòng với giao diện của hệ thống không

Tầm quan trọng của Usability:

* Usability là điều kiện sống còn
* Người dùng thường phán xét một hệ thống dựa trên giao diện của nó hơn là dựa trên chức năng
* Một thiết kế giao diện nghèo nàn có thể là nguyên nhân gây ra những lỗi người dùng nghiêm trọng
* Thiết kế giao diện người dùng nghèo nàn là lý do vì sao mà một số phần mềm không bao giờ được sử dụng
* **Learnability (Khả năng học hỏi):**

Cách để giao tiếp và trình bày mô hình hệ thống

* Affordances (Khả năng đáp ứng)
* Natural mapping (Ánh xạ tự nhiên)
* Visibility (Trực quan hoá)
* Metaphors (Ẩn dụ)
* Feedback (Phản hồi): âm thanh, rung động, hình ảnh, cảm giác,…

Tính đồng bộ/nhất quán trong giao diện:

* Đồng bộ các yếu tố bên trong hệ thống, bên ngoài hệ thống
* Sử dụng ngôn ngữ của người dùng: Sử dụng từ ngữ thông dụng, tránh những từ lóng, biệt ngữ, tránh diễn tả dài dòng
* Các yếu tố ẩn dụ phản ánh các đối tượng ngoài thế giới thực
* Các tiêu chuẩn nền tảng: Tuân theo các hướng dẫn, khuôn mẫu, học hỏi các ứng dụng có sẵn
* **Efficiency (Tính hiệu quả):**

***Định luật Fitts:*** Khoảng thời gian T để di chuyển tay đến một đối tượng có kích thước S ở khoảng cách D cách xa con trỏ chuột là:

**T = a + b \* log(D/S + 1)** với a, b là hằng số

Hiệu quả của việc trỏ đến đối tượng theo luật Fitts phụ thuộc vào:

* Kích thước của đối tượng
* Khoảng cách đến đối tượng

Áp dụng định luật Fitts:

* Gom các đối tượng tương đồng, các đối tượng được thực hiện thường xuyên thành 1 nhóm
* Các đối tượng ở cạnh của màn hình thì dễ dàng tương tác hơn
* Menu dạng bánh dễ sử dụng hơn menu tuyến tính
* Nên hạn chế sử dụng menu dài

Nguyên tắc cải thiện tính hiệu quả:

* Kích thước các đối tượng được sử dụng thường xuyên nên được thiết kế lớn hơn các đối tượng còn lại
* Gom nhóm các đối tượng có cùng mục đích sử dụng
* Đặt menu sử dụng thường xuyên ở phía trên
* Tận dụng các góc và cạnh màn hình
* Sử dụng các phím tắt bàn phím và các menu nhanh
* Định nghĩa các nhóm theo kiểu mẫu
* Tổng hợp và chọn lọc những lựa chọn chung thành mặc định
* Sử dụng các mặc định
* Lưu lịch sử người dùng
* Tự động điền
* Gợi ý
* Dự đoán hành động kế tiếp của người dùng
* **UI Design Processes (Quy trình thiết kế giao diện):**

Quy trình cho đồ án của lớp:

* Interative Model (Mô hình lặp)
* Mỗi vòng tròn là 1 vòng lặp
* Sản phẩm release được ra đời ở cuối mỗi vòng lặp
* Phản hồi và đánh giá của khách hàng có thể được hợp nhất trong sản phẩm release kế tiếp
* Mất nhiều thời gian của khách hàng trong việc kiểm tra
* Khách hàng có thể không có sẵn
* Khách hàng ko thích 🡪 Khách hàng ko mua
* Spiral Model (Mô hình xoắn ốc)
* Chu kỳ đầu sử dụng các prototype giá rẻ: paper prototype, sketches computer, quick prototype tools
* Cung cấp nhiều prototype thay thế: parallel prototyping
* Chu kỳ sau tốt hơn chu kỳ trước
* Chỉ những sản phẩm release của chu kỳ sau mới được phân phối tới người dùng
* User-Centered Design (Thiết kế tập trung vào người dùng)
* Thông tin chính xác và những gợi ý hữu ích
* Cơ hội tranh luận cho ra thiết kế quyết định
* Gia tăng sự tham gia của cái tôi trong việc thành công của hệ thống
* Người dùng không phải luôn có sẵn để tham dự vào
* Mất nhiều thời gian của người dùng
* Người dùng ko phải là người thiết kế UI
* Người dùng có quyền ưu tiên và cái tôi rất lớn
* Người thiết kế UI quá tuân theo quyền ưu tiên của người dùng

Làm thế nào để làm việc nhóm hiệu quả:

* Minh bạch mục tiêu cần đạt và kỳ vọng
* Minh bạch công việc và trách nhiệm của các thành viên
* Nói về trách nhiệm giải trình khi có sự cố
* Họp nhóm hàng tuần để xem lại các trạng thái công việc
* Bắt đầu làm việc sớm
* Thấu hiểu đồng đội

Một số nguyên tắc khi thiết kế UI:

* Xác định cấp độ kỹ năng của người dùng
* Xác định các công việc
* Chọn phong cách tương tác phù hợp
* Sử dụng 8 quy tắc vàng của Shneiderman về thiết kế giao diện
* Ngăn chặn và giảm lỗi
* Gia tăng tự động hoá trong trong sự kiểm soát của con người

***8 quy tắc vàng của Shneiderman:***

* Đảm bảo đồng nhất trong giao diện
* Phục vụ khả năng sử dụng toàn thể
* Hỗ trợ phản hồi có thông tin
* Sử dụng hộp thoại ràng buộc
* Tránh lỗi và phục hồi nhanh chóng
* Cho phép quay lại trạng thái trước đó
* Hỗ trợ người dùng điều khiển giao diện
* Giảm thiểu việc ghi nhớ cho người dùng
* **Task Analysis (Phân tích công việc):**

Phân tích người dùng (User Analysis): Xác định và mô tả các đặc trưng của người dùng tiềm năng

* *Age, gender, culture, language:* Độ tuổi, giới tính, văn hoá, ngôn ngữ
* *Computer experience:* Kinh nghiệm sử dụng máy tính
* *Domain experience, application experience:* Trải nghiệm về lĩnh vực, ứng dụng
* *Usage frequency:* Tần suất sử dụng
* *Physical limitations:* Giới hạn vật lý
* *Education:* Giáo dục
* *Motivation:* Động lực
* *Work environment:* Môi trường làm việc
* *User relationships:* Quan hệ người dùng
* *User social status:* Trạng thái, mức độ thành công của người dùng

Các kỹ thuật phân tích (Technique to do task analysis):

* *Ghi dữ liệu (Data recording):* documents, notes, audio, video, photographs,…
* *Phỏng vấn (Interviews):* Structured, Unstructured, Semi-Structured
* *Bảng câu hỏi (Questionnaires):* paper, tools, online questionnaires,…
* *Quan sát (Observation):* Direct observation, Indirect observation
* *Kết hợp các kỹ thuật trên (Combination of the above)*

Các kỹ thuật đánh giá và sàng lọc yêu cầu:

* Walk-through
* Thảo luận trực tiếp (Formal review/inspection)
* Gửi tài liệu và nhận phản hồi (Offline review)
* Họp online (Online review)

2 bước quan trọng khi tiến hành phân tích công việc:

* Mô hình công việc (Model task):
* Thu thập thông tin
* Mô tả công việc theo yêu cầu
* Đánh giá và tinh chế lại (Evaluation and refine): Xem lại và cập nhật yêu cầu
* **Prototype:**

Vì sao cần prototype:

* Thử nghiệm với các thiết kế thay thế
* Cung cấp một đại diện đầu tiên và cụ thể hoá cho ý tưởng thiết kế
* Cung cấp kinh nghiệm thực hành trên ứng dụng cho các stakeholders
* Dễ dàng thay đổi hoặc vứt đi
* Giữ lại thiết kế trọng tâm cho người dùng
* Tạo điều kiện để thiết kế và đánh giá nhiều lần
* Giảm thiểu rủi ro khiến khách hàng bất ngờ
* Phân loại prototype:
* Protype với độ trung thực thấp (Low-fidelity prototype)
* Protype với độ trung thực cao (High-fidelity prototype)

Các kiểu mẫu prototype:

* Horizontal: Prototype bao phủ phần lớn các chức năng và ít chi tiết
* Vertical: Prototype bao phủ phần nhỏ các chức năng và rất nhiều chi tiết
* Diagonal: Prototype bao phủ dọc xuống một mức nhất định
* Look:
* Xuất hiện thiết kế đồ hoạ cho UI
* Có thể sơ sài và được vẽ bằng tay
* Feel:
* Giới thiệu các phương pháp đầu vào để tương tác với giao diện
* Trỏ bằng tay hoặc viết

***Paper prototype:***

* Sử dụng giấy giả lập để đại diện cho UI
* Tương tác tự nhiên: Dùng ngón tay thay thế chuột, viết tay thay cho đánh máy
* Một người sẽ mô phỏng lại hoạt động của máy tính
* Độ trung thực thấp khi nhìn và cảm nhận
* Độ trung thực cao về chiều sâu khi có người mô phỏng hoạt động
* Dựng hình nhanh chóng, dễ thay đổi, rẻ, tập trung chú ý vào bức hình lớn
* Người không phải là lập trình viên cũng có thể làm được
* Thuận tiện

***Computer prototype:***

* Mô phỏng phần mềm tương tác
* Độ trung thực cao khi nhìn và cảm nhận
* Độ trung thực thấp về chiều sâu
* Nhanh hơn viết mã nguồn
* Không thể gỡ lỗi
* Dễ dàng thay đổi hoặc vứt đi
* Tách rời ý tưởng thiết kế với những gì mà các bộ cung cụ cung cấp (Visual Studio, C++ Builder,…)
* Người không phải là lập trình viên cũng có thể làm được
* Các kỹ thuật:
* Storyboard (Bảng chuyện): Vẽ tuần tự các màn hình, thỉnh thoảng kết hợp với links
* Form builder (Tạo mẫu): Tạo ra bảng mẫu với các của sổ, nhãn, nút bấm,…
* Wizard of Oz: Computer frontend, Con người backend
* **Web UI:**

Những điều quan trọng trong thiết kế web:

* 2 mục tiêu khó nắm bắt nhất của việc thiết kế web hiệu quả là sự hài lòng của người dùng và hoàn vốn đầu tư
* Khả năng sử dụng (Usability) vẫn là vấn đề quan trọng trong thiết kế web. Có nhiều lý do để người dùng rời bỏ trang web:
* Trang khó sử dụng, không rõ ràng, khó đọc, ko gây sự chú ý, ko thu hút, gây nhàm chán, trang không hoạt động theo cách mà nó từng hoạt động
* Người dùng “bị lạc”, người dùng cảm thấy bực mình

Text & Hypertext:

* Cấu trúc phi tuyến tính (Non-linear structure):
* Khối văn bản (trang)
* Links giữa các trang tạo thành 1 mạng lưới
* Rất mạnh mẽ
* Người dùng có thể theo dõi đường đi riêng của họ thông qua thông tin
* Thách thức của cấu trúc phi tuyến tính:
* Hoạt động kết nối được mở rộng mạnh mẽ
* Hệ quả là mất cấu trúc logic
* Nhận thức và nội dung: Fragmentary information – No integration 🡪 Confusion
* Điều hướng và cấu trúc: Hyperlink move across structure 🡪 Where am I?
* Tìm kiếm và phương thức chuyển giao là điều quan trọng

Khả năng sử dụng trong thiết kế web (Usability for Web design):

* Một số nguyên tắc khi thiết kế Web UI:
* Thiết kế lặp, hướng người dùng
* Áp dụng các mô hình
* Phát triển các nguyên tắc, luật, tiêu chuẩn
* Bố trí thông tin và các yếu tố về nhận thức
* Phương pháp đánh giá
* Một số chú ý đặc biệt cho Khả năng sử dụng web (Web Usability)
* Vấn đề mạng (Network issues):
* Băng thông (Bandwidth)
* Thời gian phản hồi (Timing)
* Độ trễ (Latency)
* Jitter
* Độ tin cậy (Reliability)
* Thiết lập kết nối (Connection set-up)
* Ý nghĩa thiết kế (Design implication): Bandwidth, Connection time, Latency
* Kiến trúc (Architectures):
* Kiến trúc thông tin: Phân loại thông tin, Tổ chức thông tin
* Kiến trúc Web UI: Điều phối công việc, Thiết kế chi tiết

Hướng dẫn và chú ý khi thiết kế web (Guidelines):

* ***Top 10 điều nên tránh khi thiết kế web (theo Jakob Nielsen, 2011)***

1. Khả năng tìm kiếm không tốt
2. Đọc thông tin trực tuyến bằng file PDF
3. Không đổi màu đường link những trang đã xem qua
4. Không thể scan tài liệu text
5. Thay đổi kích thước
6. Tiêu đề trang web không phản ánh được nội dung trang
7. Những điều nhìn tương tự như quảng cáo
8. Vi phạm quy ước thiết kế
9. Mở những cửa sổ trình duyệt mới
10. Không đáp ứng được nhu cầu của người dùng

* ***Top 10 điều khuyên làm khi thiết kế web (theo Jakob Nielsen)***

1. Đặt **tên và logo** ở mỗi trang và tạo một liên kết
2. Cung cấp chức năng **Tìm kiếm** đối với những trang web có nhiều hơn 100 trang
3. Viết ngắn gọn, đơn giản ở **dòng đầu và tiêu đề trang**
4. Cấu trúc trang **tạo điều kiện cho việc** **scan** tài liệu
5. Sử dụng **siêu văn bản để cấu trúc** không gian nội dung thành một số chủ đề nhỏ
6. Sử dụng **hình ảnh** nhưng tránh gây lộn xộn, cồng kềnh cho trang
7. Tăng cường liên quan hình ảnh, giảm bớt những hình ảnh nhỏ
8. Sử dung **link title** cho người dùng xem trước trang họ sẽ chuyển đến
9. Đảm bảo những người khuyết tật có thể truy cập được tất cả các trang quan trọng
10. Tất cả mọi người đều giống nhau

* Một số hướng dẫn khi thiết kế web
* *Site tourists:* Không thể điều khiển người dùng
* *Loyal visitors:* “**Please bookmark me**” – Hãy đánh dấu
* *Avoid frames:* Tránh sử dụng khung
* Trang web nên duy trì trang thái hiện tại của người dùng
* Người dùng không nhẫn nại
* Giảm thiểu việc sử dụng thanh cuộn, đặc biệt là ở trang Home
* Nên tránh sử dụng hiệu ứng nếu không có mục đích
* Tránh hiệu ứng liên tục
* Tránh việc “*Enter here*”
* Giới hạn số lượng màu
* Trang web cần rõ ràng những gì nó phải thực hiện và nó được tổ chức như thế nào
* Không phù hợp với phương tiện truyền thông khác
* Link đúng, tránh dùng “*Click here*” mà không có hightlight hoặc phản hồi
* Cho người biết họ đang ở đâu
* Các khảo sát, biểu mẫu cần có sự tương tác mượt mà, các nút bấm rõ ràng, chính xác, dễ quay lui
* Ẩn dụ mạnh mẽ
* Nền trắng được ưa thích vì dễ đọc
* **Mobile UI**

Mobile Computing (Điện toán di động):

* Phân loại:
* Wearable computer (Máy tính đeo được)
* Ultra-Mobile PC – UMPC (Máy tính siêu di động)
* Handheld computer – PDA (Máy tính cầm tay)
* Smart phone
* Tablet computer
* Đa dạng từ các yếu tố: Kích thước, Hình dạng, Yếu tố tương tác,…
* Đa dạng về chức năng: Truyền thông đa phương tiện, Game, Máy tính mini,…
* Người dùng gắn bó với chúng: mang theo, cầm tay, RFID, gắn chặt,…
* Nghiên cứu lĩnh vực HCI (Tương tác người – máy) “hot”

Thiết kế Mobile UI:

* Những hạn chế của Mobile UI
* Yêu cầu: Con người muốn các chức năng của PC nhưng không muốn chi phí hoạt động của PC
* Không thể chỉ cần làm cho màn hình lớn hơn và thêm bàn phím bởi vì ko sẽ ai mang theo nó
* Không thể sử dụng một bộ xử lý quá nhanh và thêm vào nhiều bộ nhớ bởi vì nó sử dụng quá nhiều năng lượng
* Khó tương tác
* Giới hạn input và output:
* Bàn phím, kích thước màn hình nhỏ, độ phân giải thấp
* Dữ liệu nhập vào dễ bị lỗi và chậm
* Môi trường người dùng tương phản
* Bộ xử lý chậm, bộ nhớ nhỏ, kết nối mạng chậm hơn so với desktop
* Tiêu dùng năng lượng giới hạn
* Chúng ta cần có cách tưởng tượng để khắc phục những vấn đề hạn chế
* ***Quy tắc vàng trong thiết kế Mobile UI:***
* Hiểu máy tính: Những hạn chế, Năng lực, Công cụ, Nền tảng
* Hiểu con người: Tâm lý, Các khía cạnh xã hội, Lỗi người dùng
* Hiểu tác vụ: Mục tiêu, Ràng buộc
* Hiểu cách tương tác của người dùng
* Các giải pháp đầu vào:
* Bàn phím vật lý nhỏ, bàn phím đính kèm
* Giảm điều hướng
* Màn hình cảm ứng
* Các cách thay thế đầu vào:
* Nhận dạng chữ viết tay
* Nhận dạng và tổng hợp lời nói
* Học sâu (Deep learning)
* Công nghệ (Technologies)
* Giao diện cử chỉ (Gestural interface)
* Cảm biến (Sensor)
* Các giải pháp đầu ra:
* Âm thanh đầu ra (giọng nói & không phải giọng nói)
* Hiển thị màu sắc, độ phân giải tốt hơn
* Kích thước màn hình
* Độ phân giải màn hình
* Màn hình cảm ứng ảo (Virtual Touch Screen – VTS)
* Sản phẩm: MS HoloLens, Oculus của Facebook, Gear VR của Samsung
* Nhận thức ngữ cảnh, dựa trên địa điểm, dựa trên hành động máy tính

Không gian sử dụng di động (Mobile usage space):

* Không gian thông tin – Information space: Mối liên hệ thông tin tĩnh *(báo giá cổ phiếu, tin tức,…)*
* Không gian tự nâng cao – Self exhancement space: Mở rộng khả năng của một người *(bộ nhớ, xây dựng vốn từ vựng,…)*
* Không gian mối quan hệ – Relationships space: Cho phép mở rông quan hệ xã hội
* Không gian giải trí – Entertainment space: Sự thưởng thức *(games, nhạc,…)*
* Không gia thương mại điện tử – M-commerce space: Dịch vụ được cung cấp từ doanh nghiệp *(phiếu giảm giá điện tử, tiền điện tử,…)*
* Không gian nhận định – Identify space: Thiết bị nhận thức về người dùng *(những thiết bị vô hình,…)*

Vai trò của hệ điều hành trên Mobile:

* Chế ngự cơ bản sự phức tạp công nghệ
* Cho phép vô số các thành phần phần mềm cùng tồn tại và hợp tác
* Làm cho nó nhìn đơn giản, tự nhiên
* Một HĐH tốt làm việc “siêng năng, chăm chỉ” thay cho người dùng
* **Graphic Design (Thiết kế đồ hoạ)**

UI tham khảo phần **nhìn và cảm nhận** của giao diện

* Người dùng gặp lần đầu: bắt mắt
* Truyền đạt ấn tượng, tâm trạng, vẻ đẹp,…
* Tạo điều kiện kết thúc công việc bằng tay
* Đề nghị tin tưởng

Khả năng đáp ứng (Affordance):

* Là chất lượng của một đối tượng cho phép 1 cá nhân thực hiện 1 hành động
* Phân loại:
* Khả năng đáp ứng nhận thức (Perceived affordance):
* Là các thuộc tính cảm nhận của đối tượng gợi ý cách chúng ta có thể sử dung nó
* Phụ thuộc vào: Kinh nghiệm, Kiến thức, Văn hoá người dùng, Bối cảnh, Bố cục, Nơi đặt đối tượng
* Khả năng đáp ứng nhận thức có thể khác với khả năng đáp ứng thực tế
* Vấn đề sẽ xảy ra khi nhận thức của người dùng khác với những gì designer nghĩ
* Khả năng đáp ứng thực tế (Actual affordance): Tính chất hành động thực tế của sản phẩm

Ràng buộc trực quan

* Giới hạn các hành động có thể nhận thức được bằng sự xuất hiện của đối tượng
* Ràng buộc càng nhiều 🡪 Lỗi xảy ra càng ít
* Lợi ích của ràng buộc:
* Hạn chế hành động không hợp lệ từ người dùng
* Giúp ngăn chặn những lựa chọn không chính xác
* Loại bỏ nhu cầu kiến thức hoàn hảo
* Giúp người dùng không cần nhớ nhiều
* Có 3 loại ràng buộc chính:
* Ràng buộc vật lý: Tham khảo cách mà các đối tượng vật lý hạn chế sự di chuyển của sự vật
* Ràng buộc văn hoá: *màu sắc, bật – tắt công tắc đèn, mở – đóng vòi nước*
* Ràng buộc logic: Khai thác những lý luận thông thường hằng ngày của con người về cách mà thế giới này hoạt động *(quan hệ logic giữa bố cục vật lý của 1 thiết bị và cách nó hoạt động)*

Triết lý thiết kế đồ hoạ:

* Hấp dẫn thẩm mỹ không tự động cung cấp khả năng sử dụng
* Thiết kế UI phải phải cân bằng ý nghĩa của các yếu tố thị giác cho phù hợp với mô hình hoạt động tinh thần
* Ưu tiên:
* “Ngôn ngữ” người dùng đơn giản và tự nhiên
* Nền kinh tế thị giác
* Tổ chức tốt, rõ ràng
* Càng đơn giản càng đẹp (Less is more)

Hướng dẫn cho một bản thiết kế đồ hoạ tốt:

* Nguyên tắc cơ bản:
* Ẩn dụ (Metaphor)
* Đơn giản và rõ ràng (Simplicity and Clarity)
* Tính nhất quán (Consistency)
* Tổ chức/ căn chỉnh/ khoảng cách gần/ sử dụng lưới (Organization/ Alignment/ Proximity/ Grid)
* Dễ đọc (Legibility and Readability)
* Màu sắc/Tương phản (Color/Contrast)
* Các kỹ thuật làm đơn giản hoá giao diện đồ hoạ:
* Xoá bớt các đối tượng (Reduction)
* Gom nhóm các đối tượng (Combine elements)
* Tập trung vào một chủ đề cho một màn hình (Consistency – Tính nhất quán)
* Sử dụng khoảng trắng thay thế các khung (White space)

Tổ chức các thành phần, yếu tố:

* Hệ thống lưới (Grid system)
* Cần thiết cho thiết kế đồ hoạ
* Các đường dọc và đường ngang để giới hạn các thành phần
* Nhất quán
* Gom nhóm các items (Group)
* Thứ tự các items (Order)
* Căn chỉnh các items (Alignment)
* Căn chỉnh các yếu tố dọc và ngang
* Sắp xếp các items (Arrangement)
* **Phân bố các yếu tố cân bằng và đối xứng xung quanh trục**
* **Interaction Styles – (Phong cách tương tác)**

Các phong cách tương tác:

* ***Menus:*** Menu
* Ưu điểm:
* Bản thân cách tương tác này có thể tự giải thích ý nghĩa, không cần có hướng dẫn kèm theo
* Yêu cầu ít bộ nhớ
* Nhấn tổ hợp phím, lỗi đầu vào của người dùng ít
* Xử lý lỗi dễ dàng
* Cải tiến có thể nhìn thấy
* Nhược điểm:
* Không hiệu quả cho người dùng chuyên gia và người dùng có tần suất sử dụng cao
* Không linh hoạt
* Chỉ giới hạn đầu vào hợp lệ ở một số điểm
* Real estate = pixel trên màn hình
* Hướng dẫn thiết kế:
* Ưu tiên cho các menu rộng và nông hơn là các menu hẹp và sâu
* Sử dụng các items làm tiêu đề cho menu con
* Gom nhóm các items có nghĩa
* Sử dụng các items ngắn gọn, bắt đầu bằng từ khoá
* Sử dụng ngữ pháp, bố cục, thuật ngữ nhất quán
* Cho phép gõ (type), bỏ qua bước (skip), phím tắt (shortcuts)
* Xem xét hỗ trợ trực tuyến, thời gian đáp ứng tối ưu, tốc độ hiển thị, kích thước màn hình
* ***Fill-in form:*** Điền vào mẫu
* Ưu điểm:
* Bản thân cách tương tác này có thể tự giải thích ý nghĩa, không cần có hướng dẫn kèm theo
* Yêu cầu ít bộ nhớ
* Sử dụng hiệu quả màn hình “Real estate”
* Chứa các tham số với nhiều giá trị có thể dùng
* Cung cấp ngữ cảnh
* Khuyết điểm:
* Giả định các kiến thức đầu vào đều hợp lệ
* Giả định về các kỹ năng đánh máy và kiến thức sử dụng các phím đặc biệt
* Nguy cơ lỗi người dùng cao
* Hướng dẫn thiết kế:
* Tiêu đề phải có nghĩa
* Có sự hướng dẫn toàn diện
* Gom nhóm theo logic và trình tự các trường thông tin
* Gán nhãn quen thuộc cho các trường thông tin
* Thuật ngữ nhất quán và được viết tắt
* Có không gian hiển thị và ranh giới cho các trường nhập dữ liệu
* Thuận tiện di chuyển con trỏ chuột
* Sửa lỗi cho từng ký tự và toàn bộ trường
* Phòng ngừa lỗi khi có thể
* Thông báo lỗi cho các giá trị không hợp lệ
* Đánh dấu các trường tuỳ chọn
* Thông báo giải thích cho các trường
* ***Direct manipulation:*** Thao tác trực tiếp, thay thế việc gõ bằng chỉ/chọn, hiển thị ngay lập tức kết quả của hành động
* Ưu điểm:
* Dễ học. dễ nhớ
* Trực tiếp, trực quan, WYSIWYG – What You See Is What You Get
* Hành động linh hoạt, dễ đảo ngược
* Cung cấp bối cảnh và phản hồi trực quan ngay lập tức
* Khai thác sử dụng các tín hiệu thị giác và không gian
* Gõ ít, phản hồi trực quan
* Nguy cơ lỗi người dùng thấp
* Khuyết điểm:
* Không hiệu quả cho người dùng chuyên gia và người dùng có tần suất sử dụng cao
* Có thể gây khó khăn trong việc thiết kế các biểu tượng dễ nhận biết cho nhiều đối tượng và nhiều hành động
* Các biểu tượng tiêu tốn nhiều pixels hơn ký tự
* Hướng dẫn thiết kế:
* Cung cấp giao diện thay thế cho người dùng chuyên gia và người dùng sử dụng thường xuyên
* Chọn sơ đồ biểu tưởng thiết kế nhất quán
* Biểu tưởng phải kèm theo tên
* Cung cấp phản hồi trực quan cho vị trí lựa chọn và chuyển động, phản hồi vật lý cho các chế độ (mode)
* ***Command language:*** Ngôn ngữ lệnh, tương tác với máy tính sử dụng dòng lệnh hoặc giọng nói dựa vào tên và cú pháp
* Ưu điểm:
* Linh hoạt
* Hỗ trợ sáng kiến người dùng
* Kêu gọi “người dùng năng lượng – power user”
* Xử lý nhanh chóng những nhiệm vụ phức tạp
* Hỗ trợ khả năng vĩ mô
* Khuyết điểm:
* Yêu cầu phải được đào tạo (training) và khả năng ghi nhớ
* Khó tiếp tục
* Xử lý lỗi kém
* Hướng dẫn thiết kế:
* Chọn tên có ý nghĩa, cụ thể, đặc biệt
* Hỗ trợ quy tắc viết tắt nhất quán
* Cung cấp khả năng tạo ra các vĩ mô (macros) cho người dùng thường xuyên
* Giới hạn số lượng lệnh và cách hoàn thành 1 nhiệm vụ
* Xem xét các menu lệnh trên màn hình tốc độ cao
* ***Function keys****:* Các phím chức năng
* Ưu điểm:
* Bản thân cách tương tác này có thể tự giải thích ý nghĩa, không cần có hướng dẫn kèm theo
* Dễ sử dụng
* Linh hoạt
* Yêu cầu con người nhớ ít
* Ít/Không cần “real estate” trên màn hình
* Hạn chế gõ
* Mối quan tâm:
* Số lượng phím chức năng có hạn
* Ứng dụng đặc biệt
* Không nhất quán giữa các ứng dụng: *Ctr + F trong Office và Outlook*
* Hướng dẫn:
* Các chức năng không thể áp dụng
* Tổ hợp phím: Dễ dàng tiếp cận, Ngữ pháp nhất quán
* ***Question and Answer:*** Câu hỏi và câu trả lời
* Kết hợp một số tính năng của **Menus** và **Fill-in form**
* Người dùng được đặt câu hỏi đơn
* Thích hợp cho những người ít vận động, ít kinh nghiệm
* Đào tạo (trainning) ít
* ***Natural language interaction – NLI :*** Tương tác bằng ngôn ngữ tự nhiên
* Tương tác với máy tính bằng ngôn ngữ tự nhiên (ngôn ngữ nói hoặc ngôn ngữ viết): Lệnh bằng giọng nói để GPS tìm quán ăn, Tìm kiếm bằng giọng nói với Google,…
* **UI Design Evaluation (Đánh giá thiết kế giao diện người dùng)**

Các phương pháp đánh giá:

* Đánh giá chuyên gia (Expert)
* Heuristic evaluation ***–*** Đánh giá dựa trên các heuristic
* Đánh giá dực trên thực tế người dùng
* Field studies – Nghiên cứu thực tế
* Controlled experiment – Thí nghiệm có kiểm soát trong nghiên cứu
* Formative evaluation – Đánh giá dựa trên các tiêu chuẩn nhất định

Mục đích của việc đánh giá:

* Xác định các vấn đề cụ thể trong UI:
* Không phù hợp giữa thiết kế và sản phẩm cuối cùng
* Không khớp giữa thiết kế và hành vi của người dùng thật sự
* Đánh giá hiệu quả tương tác: Khả năng sử dụng và hài lòng của người dùng
* Khám phá và tìm hiểu chức năng của hệ thống: Tìm hiểu tương tác thực sự của người dùng

Đánh giá Heuristic:

* Các heuristic thường dựa trên:
* 10 heuristic của Nielsen
* Các quy tắc thiết kế đồ dùng hàng ngày của Norman
* 8 quy tắc vàng của Shneiderman
* Các bước cơ bản:
* Xem xét/kiểm tra UI
* So sánh với các heuristic được chọn
* Tài liệu hoá các vấn đề về khả năng sử dụng
* Giải thích và phản biện các vấn đề chống lại các heuristic
* Hướng dẫn đánh giá theo heuristic:
* Cụ thể hoá từng lỗi với heuristic
* Liệt kê các lỗi
* Thực hiện lặp lại nhiều lần
* Nên chọn nhiều nguồn heuristic khác nhau
* Sử dụng nhiều đánh giá
* Sử dụng đánh giá heuristic ngoài việc kiểm tra người dùng
* Đánh giá nguyên mẫu (Prototype evaluation):
* Đánh giá heuristic được sử dụng để đánh giá: Sketchs, Paper prototype, Computer prototype, Code demo
* Một số các heuristic không được áp dụng để đánh giá: Phản hồi, Những thành phần không được hiển thị, Tài liệu và hướng dẫn sử dụng
* Quy trình đánh giá chính thức:
* *Đào tạo (Training):* Giới thiệu ứng dụng và giao diện của nó
* *Đánh giá (Evaluation):* Đánh giá độc lập, ghi lại các lỗi và báo cáo lỗi
* *Đánh giá mức độ nghiêm trọng (Severity rating):* Cung cấp mức độ nghiêm trọng cho các lỗi được tìm thấy
* Cosmetic or Trivial: Không quan trọng, độ ưu tiên thấp
* Minor: Không nghiêm trọng, độ ưu tiên thấp
* Major: Lỗi khá nghiêm trọng, độ ưu tiên cao
* Critical: Lỗi nghiêm trọng, độ ưu tiên cao

Các yếu tố cần xem xét để xếp hạng lỗi: Tần suất (Độ phổ biến), Tác động (Khó/Dễ giải quyết), Sự lặp đi lặp lại lỗi

* *Tóm tắt (Briefing):* Thảo luận kết quả và đề xuất giải pháp

Field studies (Nghiên cứu thực tế):

* Môi trường sử dụng thực tế:
* Quan sát hiệu ứng xảy ra trong thiết lập thực tế
* Không thể nó UI tốt như thế nào nào cho đến khi người dùng sử dụng sản phẩm và cảm thấy hài lòng
* Khó dự đoán hành động của người dùng thật sự
* Khả năng sử dụng có thể được đo lường trong quá trình thử nghiệm bằng cách tính toán hiệu suất người dùng
* Một số vấn đề:
* Chi phí/Thời gian tiêu tốn
* Có thể không khái quát

Controlled experiment (Thí nghiệm có kiểm soát trong nghiên cứu): Nhiệm vụ chính của thí nghiệm được kiểm soát

* Xác định giả thuyết
* Xác định các biến độc lập và các biến phụ thuộc
* Chọn môn
* Thực hiện phân tích thống kê
* Giải thích kết quả

Formative evaluation (Đánh giá dựa trên các tiêu chuẩn nhất định):

* Các bước cơ bản tương tự nghiên cứu thực tế
* Điểm khác:
* Người dùng được gán cho một số nhiệm vụ nhất định
* Đánh giá
* Chọn người tham gia
* Chỉ định nhiệm vụ cho những người tham gia
* Theo dõi những người tham gia thực hiện nhiệm vụ
* **“Think aloud”** – Phương pháp đánh giá được sử dụng rộng rãi trong công nghiệp
* Ưu điểm:

Đơn giản, ít chuyên môn

Có thể cung cấp cái nhìn sâu sắc và hữu ích về suy nghĩ của người dùng

Có thể chỉ ra cách hệ thống thực sự sử dụng

* Nhược điểm:

Chủ quan

Chọn lọc

Hành động mô tả có thể làm thay đổi hiệu suất công việc