

## Mục lục

|   |    |
|---|----|
| MỤC LỤC .....   | 1  |
| PHẦN I: GIỚI THIỆU CHUNG VỀ TƯƠNG TÁC NGƯỜI - MÁY ..... | 4  |
| CHƯƠNG I: GIỚI THIỆU VỀ TƯƠNG TÁC NGƯỜI - MÁY.....      | 4  |
| 1.1. Nhu cầu thiết kế khi thay đổi công nghệ.....       | 4  |
| 1.2 Thách thức của HCI .....                            | 4  |
| 1.3 Đích của HCI .....                                  | 4  |
| 1.4 Mục đích của HCI.....                               | 5  |
| 1.5 Các thành phần của HCI.....                         | 5  |
| CHƯƠNG II: CON NGƯỜI.....                               | 6  |
| 2.1. Giới thiệu .....                                   | 6  |
| 2.2. Các kênh vào - ra .....                            | 8  |
| 2.2.1. Thị giác .....                                   | 9  |
| 2.2.2. Thính giác .....                                 | 18 |
| 2.2.3. Xúc giác .....                                   | 21 |
| 2.2.4. Sự chuyển động.....                              | 22 |
| 2.3. Bộ nhớ .....                                       | 24 |
| 2.3.1. Bộ nhớ ngắn hạn .....                            | 24 |
| 2.3.2. Bộ nhớ dài hạn .....                             | 26 |
| CHƯƠNG III: MÁY TÍNH .....                              | 28 |
| 3.1. Các thiết bị nhập .....                            | 28 |
| 3.1.1. Các thiết bị nhập chuẩn.....                     | 28 |
| 3.1.2. Các phương pháp nhập tự chọn .....               | 28 |
| 3.2. Các thiết bị xuất .....                            | 30 |
| 3.3. Bộ nhớ .....                                       | 31 |
| 3.3.1. Bộ nhớ ngắn hạn .....                            | 31 |
| 3.3.2. Bộ nhớ dài hạn .....                             | 31 |
| CHƯƠNG IV: TƯƠNG TÁC.....                               | 36 |
| 4.1. Các mô hình tương tác .....                        | 36 |
| 4.1.1. Chu trình thực hiện-đánh giá .....               | 36 |
| 4.1.2. Khung tương tác.....                             | 38 |
| 4.2. Các khung và HCI .....                             | 39 |

|   |           |
|---|-----------|
| <b>4.3. Tương quan người – môi trường .....</b>                             | <b>40</b> |
| 4.3.1. Sắp xếp các điều khiển và hiển thị.....                              | 41        |
| 4.3.2. Môi trường vật lý của tương tác .....                                | 42        |
| 4.3.3. Các vấn đề về sức khoẻ.....  | 42        |
| 4.3.4. Sử dụng màu sắc .....  | 43        |
| <b>4.4. Một số phong cách giao diện .....</b>                               | <b>44</b> |
| 4.4.1. Giao diện lệnh .....   | 44        |
| 4.4.2. Menu .....   | 45        |
| 4.4.3. Ngôn ngữ tự nhiên .....  | 46        |
| 4.4.4. Đối thoại truy vấn và đối thoại khiếu hỏi /trả lời.....              | 46        |
| 4.4.5. Form –fill và bảng tính .....  | 47        |
| 4.4.6. WIMP.....  | 47        |
| 4.4.7. Point và click .....   | 48        |
| <b>PHẦN II: THIẾT KẾ GIAO DIỆN NGƯỜI DÙNG .....</b>                         | <b>49</b> |
| <b>CHƯƠNG V: GIỚI THIỆU CHUNG VỀ THIẾT KẾ GIAO DIỆN NGƯỜI DÙNG .....</b>    | <b>49</b> |
| <b>5.1. Khái niệm giao diện người dùng.....</b>                             | <b>49</b> |
| <b>5.2. Tại sao cần thiết kế giao diện .....</b>                            | <b>50</b> |
| <b>5.3. Các quy tắc thiết kế giao diện người dùng .....</b>                 | <b>51</b> |
| 5.3.1. Hướng dẫn tương tác chung.....                                       | 51        |
| 5.3.2. Hướng dẫn về việc hiển thị thông tin.....                            | 52        |
| 5.3.3. Hướng dẫn về việc vào dữ liệu .....                                  | 54        |
| <b>CHƯƠNG VI: CÁC HOẠT ĐỘNG PHÁT TRIỂN PHẦN MỀM.....</b>                    | <b>56</b> |
| <b>6.1. Kỹ nghệ phần mềm .....</b>  | <b>56</b> |
| <b>6.2. Vòng đời cổ điển.....</b>   | <b>57</b> |
| <b>6.3. Làm bản mẫu .....</b>   | <b>59</b> |
| <b>CHƯƠNG VII: CÁC MÔ HÌNH CỦA NGƯỜI DÙNG TRONG THIẾT KẾ GIAO DIỆN.....</b> | <b>61</b> |
| <b>7.1. Mô hình các yêu cầu của người dùng.....</b>                         | <b>61</b> |
| <b>7.2. Các mô hình kỹ thuật-xã hội.....</b>                                | <b>61</b> |
| 7.2.1. USTM/CUSTOM.....   | 61        |
| 7.2.2. OSTA.....  | 64        |
| 7.2.3. ETHICS .....   | 65        |
| <b>7.3. Phương pháp luận các hệ thống phần mềm .....</b>                    | <b>66</b> |

|  |           |
|--|-----------|
| <b>7.4. Thiết kế hợp tác.....</b>  | <b>68</b> |
| <b>7.5. Các mô hình nhận thức .....</b>  | <b>70</b> |
| <b>7.6. Mô hình phân cấp mục đích và nhiệm vụ.....</b>                             | <b>71</b> |
| 7.6.1. GOMS .....  | 73        |
| 7.6.2. CCT.....  | 76        |
| <b>CHƯƠNG VIII: QUY TRÌNH THIẾT KẾ GIAO DIỆN NGƯỜI DÙNG HƯỚNG NGƯỜI DÙNG .....</b> | <b>81</b> |
| <b>8.1. Quy trình thiết kế hướng người dùng .....</b>                              | <b>81</b> |
| <b>8.2. Pha phân tích.....</b>   | <b>83</b> |
| <b>8.3. Pha thiết kế .....</b>   | <b>84</b> |
| <b>8.4. Pha đánh giá .....</b>   | <b>85</b> |
| <b>8.5. Những ưu điểm của quy trình phát triển hướng người dùng.....</b>           | <b>85</b> |
| <b>CHƯƠNG IX: QUY TRÌNH THIẾT KẾ GIAO DIỆN NGƯỜI DÙNG HƯỚNG NHIỆM VỤ.....</b>      | <b>87</b> |
| <b>9.1. Quy trình thiết kế hướng nhiệm vụ.....</b>                                 | <b>87</b> |
| 9.1.1. Xác định mục đích sử dụng hệ thống của người sử dụng .....                  | 88        |
| 9.1.2. Lựa chọn các nhiệm vụ .....   | 89        |
| 9.1.3. Sao chép .....  | 90        |
| <b>9.2. Phân tích nhiệm vụ .....</b>   | <b>91</b> |
| 9.2.1. Định nghĩa.....   | 91        |
| 9.2.2. Các phương pháp và các kỹ thuật phân tích .....                             | 91        |
| <b>9.3. Mô hình hoá nhiệm vụ .....</b>   | <b>94</b> |
| 9.3.1. Mô hình nhiệm vụ truyền thống .....   | 94        |
| 9.3.2. UML.....  | 95        |
| 9.3.3. Mô hình ngữ cảnh .....  | 97        |

# **PHẦN I: GIỚI THIỆU CHUNG VỀ TƯƠNG TÁC NGƯỜI - MÁY**

## **CHƯƠNG I: GIỚI THIỆU VỀ TƯƠNG TÁC NGƯỜI - MÁY**

### **1.1. Nhu cầu thiết kế khi thay đổi công nghệ**

Để bán được nhiều, máy tính cần được thiết kế tốt hơn nữa, tiến đến máy tính tiện dùng có tri thức.

Các nhà tâm lý, xã hội học nhất trí rằng tính phức tạp của các hệ thống tính toán hầu hết là do giao diện người dung với máy tính quá nghèo nàn. D.Norman cho rằng hai nguyên tắc đảm bảo giao diện tốt là:

- Tính hiển hiện: người dung cần thể hiện được các đối tượng và thể hiện là rõ ràng.
- Tính tác động: là tính tự đề xuất các chức năng đã có.

### **1.2 Thách thức của HCI**

Trong chục năm qua, với kiến thức hạn chế người ta vẫn cảm thấy có nhiều yêu cầu đối với thiết kế giao diện hiệu quả. Thiết kế HCI là thách thức do nền công nghệ phát triển nhanh, tốc độ phát minh, môi trường mạng, mạng tích hợp số ISDN. Hai thách thức quan trọng là:

- Cách tồn tại chung với thay đổi của công nghệ.
- Cách đảm bảo thiết kế cho ra HCI tốt như là nền cho các công nghệ mới.

Thông qua quá trình tương tác với máy điện thoại, các thao tác nối đường, chuyển thông tin, ngắt đường truyền... người ta thấy sự cần thiết của giao diện tương tác trong đời sống.

### **1.3 Đích của HCI**

Đích của việc nghiên cứu HCI là cho ra hệ thống dung được và an toàn. Đích được tóm tắt như là phát triển hay cải thiện tính an toàn, tiện dụng và hiệu quả của hệ thống có máy tính, trong đó hệ thống có nghĩa gồm cả môi trường trong lí thuyết hệ thống.

Tính tiện ích nhằm vào các chức năng của hệ thống, là những điều có thể làm được. Việc cải thiện tính hiệu quả là yếu tố tự thân.

Để có giao diện sử dụng tốt, đặc tả HCI cần:

- Hiểu hết các nhân tố, như tâm lí, thực tiễn, tổ chức và nhân tố xã hội. Điều này cho phép xác định cách thao tác và tạo điều kiện cho công nghệ có hiệu quả
- Phát triển các công cụ và kĩ thuật để người thiết kế đảm bảo hệ thống máy tính phù hợp với các hoạt động của con người dùng máy.
- Đạt được tương tác an toàn, hiệu quả cả về thuật ngữ tương tác người – máy và tương tác nhóm.

#### **1.4 Mục đích của HCI**

- Mô tả HCI: thông qua cách mô tả ta biết được một phần nội dung bên trong
- HCI quan trọng trong việc phát triển của công nghệ
- HCI liên quan đến hiệu suất, sự an toàn của người sử dụng

#### **1.5 Các thành phần của HCI**

- HCI thuộc đa lĩnh vực
- Các lĩnh vực hỗ trợ: công nghệ, triết học, ngôn ngữ học....
- Mô hình khái niệm của HCI
- Thiết kế HCI

## CHƯƠNG II: CON NGƯỜI

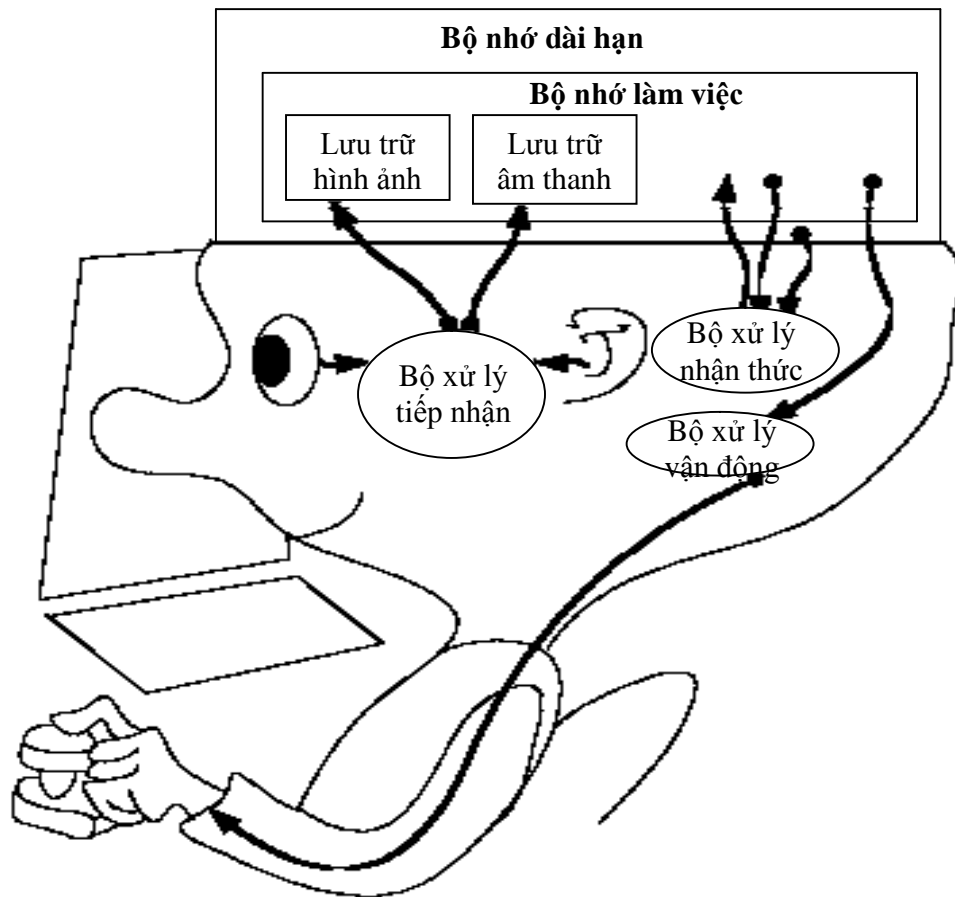
### 2.1. Giới thiệu

Trong chương này chúng ta sẽ giới thiệu qua một số vấn đề cơ bản của tương tác người máy. Chúng ta sẽ bắt đầu với con người, nhân vật trung tâm trong bất kỳ nghiên cứu nào về các hệ thống tương tác. Con người, hay còn gọi là người sử dụng, trên tất cả, là đối tượng mà các hệ thống máy tính được thiết kế ra để trợ giúp cho họ. Do đó, sẽ là rất hợp lý khi chúng ta đặt những yêu cầu của người sử dụng lên thứ tự ưu tiên hàng đầu.

Trong chương này, chúng ta sẽ xem xét qua các lĩnh vực của tâm lý học con người theo quan điểm của tâm lý học nhận thức. Điều này dường như có vẻ không liên quan gì đến thiết kế và xây dựng các hệ thống máy tính tương tác, nhưng sự thực không hẳn là như vậy. Để thiết kế một cái gì đó cho một ai đó, chúng ta cần biết được những khả năng và hạn chế của người đó, biết được những gì sẽ gây khó khăn cho người sử dụng và những gì sẽ làm cho người sử dụng không có khả năng sử dụng sản phẩm. Việc biết được khả năng và giới hạn của con người cũng sẽ giúp chúng ta biết được những gì là dễ dàng cho người sử dụng, để trong quá trình thiết kế chúng ta sẽ tập trung vào những đặc điểm đó. Chúng ta cũng sẽ xem xét qua các khía cạnh của tâm lý học nhận thức liên quan đến việc sử dụng các hệ thống máy tính: Cách con người tiếp nhận thông tin xung quanh họ, cách con người lưu trữ thông tin, xử lý thông tin và giải quyết vấn đề và cách họ xử lý các đối tượng theo một cách tự nhiên.

Trong giáo trình này, chúng ta chỉ tập trung nghiên cứu những khía cạnh liên quan đến tương tác người máy. Cụ thể chúng ta sẽ xem xét người sử dụng bằng cách tập trung vào các lĩnh vực mà chúng ta quan tâm nhất, bằng cách đưa ra một mô hình đơn giản về bộ xử lý của con người. Có rất nhiều mô hình đã được đưa ra, tuy nhiên ở đây chúng ta chỉ xem xét đến mô hình có ảnh hưởng lớn nhất. Năm 1983, Card, Moran và Newell đã đưa ra mô hình bộ xử lý của con người, một cách nhìn được đơn giản hoá về bộ xử lý của con người trong quá trình tương tác với máy tính (hình vẽ 1.1). Mô hình bao gồm 3 hệ thống con: hệ thống tiếp nhận dùng để xử lý các kích thích giác quan từ bên ngoài, hệ thống vận động dùng để điều khiển hành động, và hệ thống nhận thức đưa ra các xử lý cần thiết để kết nối với hai hệ thống vận động. Mỗi một hệ thống con đều có bộ nhớ và bộ xử lý riêng, mặc dù trong trường hợp này rõ

ràng là độ phức tạp của mỗi hệ thống là khác nhau và nó phụ thuộc vào độ phức tạp của các nhiệm vụ mà hệ thống thực hiện. Mô hình cũng bao gồm một số nguyên tắc hoạt động để điều khiển hành vi của hệ thống trong những điều kiện nhất định.



Hình 1.1: Mô hình bộ xử lý thông tin của con người (do Card, Moran và Newell đưa ra năm 1983)

Ở đây, chúng ta vẫn sẽ xem người sử dụng như một hệ thống xử lý thông tin để có sự tương đương nhiều hơn về mặt chức năng so với hệ thống máy tính thông thường. Thông tin đầu vào được lưu trữ và xử lý, và cho ra thông tin đầu ra. Do đó, chúng ta sẽ thảo luận đến 3 thành phần của hệ thống này: đầu vào-đầu ra, bộ nhớ, và quá trình xử lý. Đối với con người, một hệ thống xử lý thông tin thông minh, thì quá trình xử lý bao gồm giải quyết vấn đề, học, và, mắc lỗi. Mô hình này rõ ràng là một sự đơn giản hoá của mô hình thực tế, vì bộ nhớ và bộ xử lý là bắt buộc ở tất cả các mức, như trong mô hình bộ xử lý con người. Tuy nhiên, mô hình đó là một cách tiếp cận hợp lý để hiểu được cách thông tin được xử lý như thế nào bởi hệ thống con người.

Con người, không giống như máy tính, chịu ảnh hưởng bởi các tác nhân từ bên ngoài như là môi trường xã hội và tổ chức và chúng ta cần phải biết được những tác nhân ảnh hưởng đó. Tuy nhiên, hiện tại, chúng ta sẽ bỏ qua những nhân tố như vậy và chỉ tập trung vào các khả năng xử lý thông tin của con người. Chúng ta sẽ quay trở lại xem xét đến những ảnh hưởng xã hội và tổ chức trong Chương 3.

Trong chương này, đầu tiên chúng ta sẽ tìm hiểu về các kênh vào-ra của con người, các giác quan và các cơ quan phản ứng. Những kênh này liên quan đến một số quá trình xử lý mức thấp. Tiếp theo, chúng ta sẽ tìm hiểu về bộ nhớ con người và cách hoạt động của chúng. Sau đó chúng ta sẽ tìm hiểu cách con người thực hiện việc giải quyết vấn đề phức tạp, cách họ học và tích lũy các kỹ năng, và tại sao họ lại bị mắc lỗi. Cuối cùng, chúng ta sẽ tìm hiểu một vài điều có thể giúp ích chúng ta trong quá trình thiết kế các hệ thống máy tính.

## **2.2. Các kênh vào - ra**

Tương tác của con người với thế giới bên ngoài xảy ra khi thông tin được tiếp nhận và được gửi qua: đầu vào và đầu ra. Trong khi tương tác với máy tính, người sử dụng nhận thông tin từ đầu ra của máy tính, và phản hồi lại bằng cách gửi thông tin đến đầu vào của máy tính - đầu ra của người sử dụng trở thành đầu vào của máy tính và ngược lại. Kết quả là, việc sử dụng thuật ngữ đầu vào và đầu ra có thể gây nên sự nhầm lẫn vì ở một mức độ nào đó chúng ta đã bỏ qua sự khác biệt giữa chúng và chỉ tập trung vào bản thân các kênh. Sự bỏ qua này là thích hợp, vì trong một tương tác một kênh cụ thể có thể đóng vai trò chủ yếu là đầu vào hoặc đầu ra, nhưng nó cũng có thể được sử dụng trong các vai trò khác. Ví dụ, thị giác được sử dụng chủ yếu trong việc nhận thông tin từ máy tính, nhưng nó cũng có thể được sử dụng để cung cấp thông tin cho máy tính bằng cách định vị một điểm cụ thể trên màn hình.

Đầu vào trong con người chủ yếu xuất hiện thông qua các giác quan và đầu ra xuất hiện thông qua sự điều khiển vận động của các cơ quan phản ứng kích thích. Có 5 giác quan chính: thị giác, thính giác, xúc giác, vị giác và khứu giác. Trong số đó, 3 giác quan đầu tiên là những giác quan quan trọng nhất đối với tương tác người máy. Hiện tại, vị giác và khứu giác không đóng vai trò quan trọng trong tương tác người máy, và chúng cũng không có vai trò trong các hệ thống máy tính thông thường, mặc dù chúng có thể có vai trò trong các hệ thống chuyên dụng (ví dụ, mùi vị dùng để đưa ra các cảnh báo về những hỏng hóc, các hoạt động bất thường xảy ra). Tuy nhiên, thị giác, thính giác và xúc giác đóng vai trò trung tâm.



Tương tự như vậy, có rất nhiều các cơ quan phản ứng kích thích, bao gồm các chi, các ngón tay, mắt, đầu và hệ thống phát âm. Khi tương tác với máy tính, các ngón tay đóng vai trò chủ yếu, thông qua việc đánh máy hay điều khiển con chuột, còn đối thoại, mắt và vị trí của đầu thì ít được sử dụng hơn.

Tưởng tượng rằng chúng ta đang sử dụng máy tính cá nhân với một con chuột và một bàn phím. Ứng dụng mà ta đang sử dụng có giao diện đồ họa, với các thực đơn, biểu tượng và cửa sổ. Khi tương tác với hệ thống này, bạn nhận thông tin chủ yếu qua việc nhìn từ những gì đang xuất hiện trên màn hình. Tuy nhiên, bạn cũng có thể nhận thông tin bằng tai: ví dụ, máy tính có thể phát ra tiếng kêu bíp khi bạn mắc lỗi. Xúc giác cũng tham gia vào quá trình để bạn có được cảm nhận về những gì bạn đang làm: phím có được nhấn hay không hoặc chuột có di chuyển hay không. Trong ví dụ này, thị giác và thính giác không trực tiếp tham gia vào quá trình gửi thông tin, mặc dù chúng có thể được sử dụng để nhận thông tin từ một nguồn thứ ba (ví dụ, từ sách, hay lời nói của một người khác) và sau đó lại chuyển thông tin đó đến máy tính.

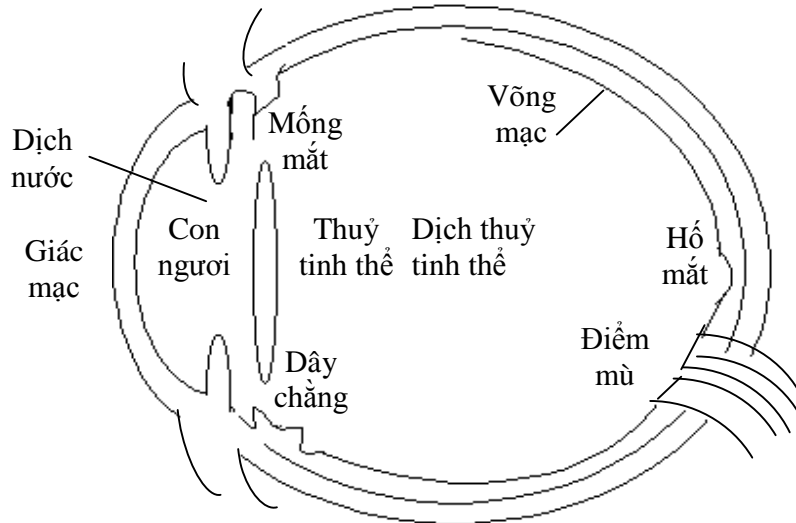
Trong phần này, chúng ta sẽ tìm hiểu qua về các thành phần chính của một tương tác như vậy. Trước tiên, chúng ta sẽ xem xét vai trò và hạn chế của 3 giác quan chính sau đó sẽ xem xét đến sự điều khiển vận động bằng tay.

### **2.2.1. Thị giác**

Thị lực của con người là một hoạt động rất phức tạp với tầm nhìn hạn chế, tuy nhiên, đối với một người bình thường nó là nguồn tiếp nhận thông tin chủ yếu. Chúng ta có thể phân chia một cách tương đối quá trình tiếp nhận bằng thị giác thành 2 giai đoạn: tiếp nhận tự nhiên kích thích từ môi trường bên ngoài, xử lý và phân tích kích thích đó. Mặt khác, đặc điểm vật lý của mắt và hệ thống thị giác có một số điểm không thể nhận biết bởi con người. Bên cạnh đó, khả năng diễn dịch của xử lý thị giác cho phép các hình ảnh có thể được xây dựng từ những thông tin không đầy đủ. Chúng ta cần hiểu cả 2 giai đoạn đó vì chúng ảnh hưởng đến khả năng tiếp nhận thị giác của một người bình thường, và điều này ảnh hưởng trực tiếp đến cách chúng ta thiết kế các hệ thống máy tính. Chúng ta sẽ bắt đầu bằng cách xem mắt như là một bộ tiếp nhận kích thích vật lý, và sau đó xem xét quá trình xử lý thị giác cơ bản.

### 2.2.1.1. Mắt người

Thị giác bắt đầu với ánh sáng. Mắt là một cơ cấu cho phép tiếp nhận ánh sáng và biến đổi nó thành năng lượng điện. Ánh sáng được phản chiếu từ các đối tượng trong thế giới và ảnh của các đối tượng đó được ghi lại là ảnh ảo nằm ở phía đuôi mắt. Các bộ tiếp nhận kích thích trong mắt biến đổi nó thành các tín hiệu điện được chuyển tới não.



Hình 1.2: Mắt người

Mắt có nhiều thành phần quan trọng (xem hình 1.2) và chúng ta sẽ xem xét chúng một cách chi tiết hơn. Giác mạc và thủy tinh thể ở phía trước mắt hội tụ ánh sáng thành một hình ảnh sắc nét nằm ở phía đuôi mắt, võng mạc. Võng mạc rất nhạy sáng và nó chứa hai loại tế bào tiếp nhận ánh sáng: tế bào hình que và tế bào hình nón.

Tế bào hình que là tế bào cực kỳ nhạy sáng và do đó cho phép chúng ta có thể nhìn được trong các điều kiện thiếu ánh sáng. Tuy nhiên, chúng không có khả năng phân giải những chi tiết nhỏ và bị lệ thuộc vào độ bão hòa ánh sáng. Điều này giải thích lí do tại sao chúng ta lại rơi vào tình trạng mù tạm thời khi đi từ một căn phòng tối ra ánh sáng: các tế bào hình que được kích hoạt và bị bão hòa bởi ánh sáng đột ngột (Tiếp nhận đủ ánh sáng, không thể tiếp nhận thêm). Các tế bào hình nón không hoạt động vì chúng bị chặn bởi các tế bào hình que. Do đó chúng ta tạm thời không thể nhìn được. Mỗi mắt có khoảng 120 triệu tế bào hình que chủ yếu nằm ở các viền của võng mạc. Do đó các tế bào hình que chi phối sự nhìn ngoại biên.

Các tế bào hình nón là bộ tiếp nhận kích thích thứ hai trong mắt. Chúng không nhạy sáng bằng các tế bào hình que và do đó có thể tiếp nhận được nhiều ánh sáng

hơn. Có 3 loại tế bào hình nón, và mỗi loại nhạy cảm với một bước sóng ánh sáng khác nhau. Điều này cho chúng ta nhìn thấy màu. Mắt có khoảng 6 triệu tế bào hình nón, chủ yếu tập trung ở hốc mắt, một vùng nhỏ trên võng mạc chứa hình ảnh.

Võng mạc chủ yếu là nơi chứa các tế bào tiếp nhận ánh sáng, ngoài ra trên võng mạc còn có một điểm mù, nơi nối các dây thần kinh thị giác với mắt. Điểm mù không có tế bào hình nón hoặc tế bào hình que, tuy nhiên hệ thống thị giác của chúng ta đã khắc phục điều này để trong các tình huống bình thường chúng ta không nhận thấy được sự thiếu hụt này.

Võng mạc cũng chứa các tế bào thần kinh được gọi là các tế bào hạch (ganglion cell). Có hai loại tế bào hạch: tế bào X, tập trung ở hốc mắt chịu trách nhiệm phát hiện các mẫu ban đầu; và tế bào Y phân tán rộng hơn trên võng mạc và chịu trách nhiệm phát hiện sự chuyển dịch ban đầu. Sự phân bố của các tế bào X, Y là để, trong khi chúng ta không có khả năng phát hiện những thay đổi trong mẫu khi nhìn ngoại biên, thì chúng ta có thể nhận biết được sự dịch chuyển.

#### **2.2.1.2. Thu nhận bằng thị giác**

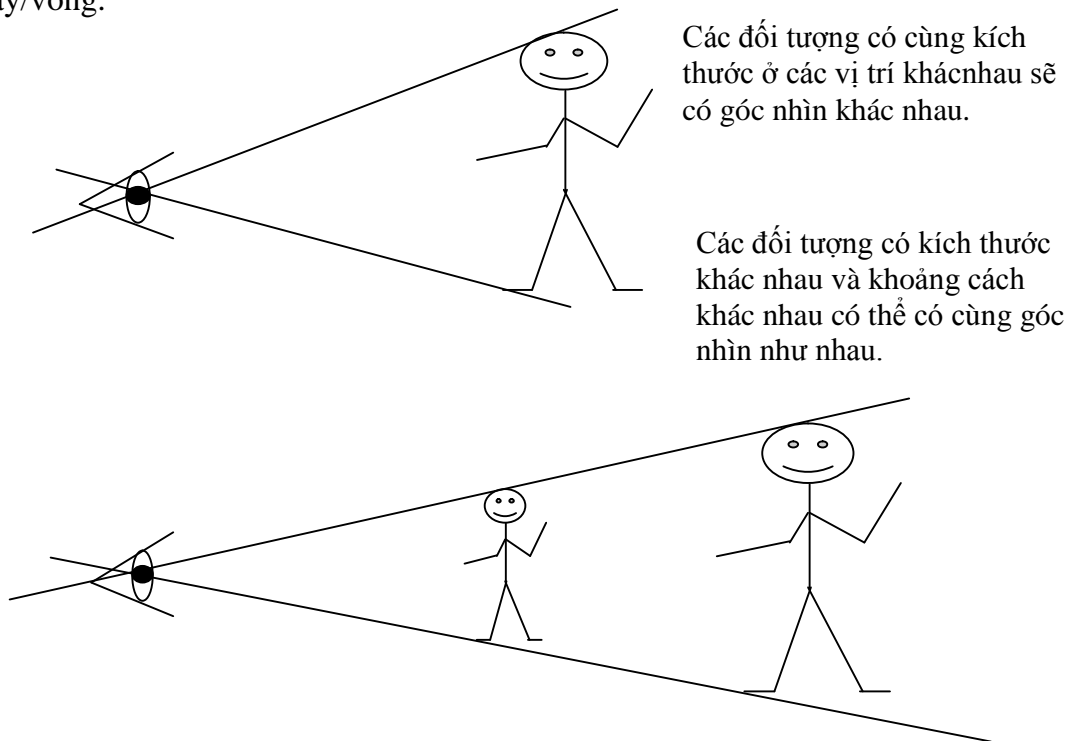
Việc hiểu được cấu trúc cơ bản của mắt sẽ giúp chúng ta giải thích được cơ cấu vật lý của thị giác, tuy nhiên quá trình thu nhận bằng thị giác thì phức tạp hơn rất nhiều. Thông tin nhận bởi bộ máy thị giác phải được lọc và chuyển đến các bộ phận xử lý để chúng ta có thể nhận dạng được các cảnh liên tục, đồng hoá khoảng cách tương đối và phân biệt được màu sắc. Trước khi xem xét những khả năng và hạn chế của xử lý thị giác, chúng ta sẽ xem xét một cách kĩ càng hơn về cách chúng ta nhận biết kích thước và chiều sâu, độ sáng và màu sắc, những điều rất quan trọng trong khi thiết kế giao diện.

Nhận biết kích thước và chiều sâu: Hãy tưởng tượng rằng bạn đang đứng trên một đỉnh đồi. Bên cạnh bạn, trên đỉnh đồi, bạn có thể nhìn thấy những tảng đá, một con cừu và một cây nhỏ. Bên sườn đồi là một trang trại với những ngôi nhà phụ và những chiếc máy cày. Trên con đường mòn, có một người đang đi về phía đỉnh đồi. Ở phía dưới, trong thung lũng, là một thị xã buôn bán nhỏ. Ta thấy rằng, khung cảnh trên đã được miêu tả một cách rất cụ thể, với kích thước của các đối tượng và khoảng cách giữa các đối tượng được cho trước. Tuy nhiên, với những cảnh như vậy, cơ quan thị giác của chúng ta rất dễ dàng hiểu được hình ảnh đó mà không cần biết đến những kích thước của các đối tượng, cũng như khoảng cách xa gần của chúng. Chúng ta có thể nhận dạng được các đối tượng tương tự nhau mà không cần quan tâm đến thực tế

rằng chúng xuất hiện với các kích thước rất khác nhau. Thực tế, chúng ta có thể sử dụng thông tin này để điều chỉnh các khoảng cách.

Vậy làm sao mà mắt người lại có thể nhận biết được kích thước, chiều sâu và các khoảng cách tương đối. Để hiểu rõ điều này chúng ta phải xem xét cách một hình ảnh xuất hiện trên võng mạc như thế nào. Như ta đã biết, ánh sáng được phản chiếu từ đối tượng tạo ra một ảnh ảo ngược chiều trên võng mạc. Kích thước của hình ảnh đó được đặc trưng bởi một góc nhìn. Hình 1.3 minh họa cách tính toán một góc nhìn.

Nếu ta vẽ một đường thẳng từ đỉnh của đối tượng đến điểm giữa ở phía trước của mắt và một đường thẳng thứ hai từ đáy của đối tượng đến cùng điểm giữa đó, thì góc nhìn của đối tượng là góc giữa hai đường thẳng đó. Góc nhìn bị ảnh hưởng bởi kích thước của đối tượng và khoảng cách của nó so với mắt. Do đó, nếu 2 đối tượng có cùng một khoảng cách, đối tượng lớn hơn sẽ có góc nhìn lớn hơn. Tương tự như vậy, nếu 2 đối tượng có cùng kích thước được đặt ở các khoảng cách khác nhau so với mắt, đối tượng nào đặt ở xa hơn sẽ có góc nhìn nhỏ hơn. Góc nhìn xác định phạm vi quan sát được của đối tượng là bao nhiêu. Đơn vị tính của góc nhìn là độ hoặc là vòng phút, trong đó 1 độ tương đương với 60 vòng phút, và 1 vòng phút tương đương với 60 giây/vòng.



Hình 1.3: Góc nhìn

Vậy, góc nhìn của đối tượng có ảnh hưởng như thế nào đến sự thu nhận của chúng ta về kích thước của đối tượng? Trước hết, nếu góc nhìn của một đối tượng là quá nhỏ, chúng ta sẽ không có khả năng thu nhận nó. *Tính sắc bén* (visual acuity) là khả năng thu nhận các chi tiết sắc nét. Có rất nhiều phương pháp có thể được dùng để kiểm tra tính sắc nét, trong đó đa phần là các phương pháp kiểm tra mắt tiêu chuẩn. Ví dụ, một người có thị lực bình thường có thể phát hiện một đường thẳng đơn nếu nó có góc nhìn là 0.5 giây. Khoảng trống giữa các đường thẳng có thể được phát hiện với góc nhìn từ 30 giây đến 1 vòng phút. Điều này cho thấy giới hạn của tính sắc bén của thị giác con người.

Tuy nhiên, giả sử rằng chúng ta đã nhận biết được đối tượng, thì góc nhìn của đối tượng có ảnh hưởng như thế nào đến sự thu nhận của chúng ta về kích thước của đối tượng? Ta biết rằng góc nhìn của một đối tượng sẽ giảm đi khi ta đưa nó ra xa mắt hơn, vậy liệu rằng chúng ta có thể thấy đối tượng nhỏ hơn không? Thực tế, sự thu nhận của chúng ta về kích thước của một đối tượng vẫn không thay đổi thậm chí ngay cả khi góc nhìn của nó thay đổi. Ví dụ chiều cao của một người là không thay đổi ngay cả khi họ di chuyển ra xa hơn so với mắt của ta. Đây chính là quy tắc kích thước không đổi, và nó cho ta biết rằng sự thu nhận của chúng ta về kích thước phụ thuộc vào các nhân tố khác hơn là phụ thuộc vào góc nhìn.

Một trong những nhân tố đó là sự thu nhận của chúng ta về chiều sâu. Quay trở lại ví dụ về cảnh ở trên đỉnh đồi, ta thấy có rất nhiều những điểm mốc để ta có thể xác định được vị trí tương đối và khoảng cách của các đối tượng trong cảnh. Nếu các đối tượng được xếp chồng lên nhau, đối tượng nào bị che lấp một phần thì nó sẽ nằm ở cảnh nền, và do đó sẽ ở xa hơn. Tương tự như vậy, kích thước và chiều cao của các đối tượng trong phạm vi quan sát có thể cho ta một cơ sở để xác định khoảng cách của đối tượng. Điểm mốc này rất quen thuộc. Ví dụ, khi muốn thay đổi kích thước của đối tượng, ta có thể dùng phương pháp điều chỉnh khoảng cách của chúng so với mắt để có được một kích thước mong muốn. Đây là điều đã được khai thác để làm tăng tính hài hước cho các quảng cáo: một quảng cáo đồ uống vẽ một người đàn ông đang đi ra từ cổ chai nằm ở phía nền trước. Khi ông ta đi, ông ta va vào cổ chai, nhưng thực tế đó là một chiếc cổ chai rất lớn nằm ở phía nền sau!

**Thu nhận độ sáng:** Một khía cạnh thứ hai của việc thu nhận bằng thị giác là sự thu nhận độ sáng. Thực tế, độ sáng là phản ứng mang tính chủ quan đối với các mức sáng. Nó chịu ảnh hưởng bởi độ chói, số đo lượng ánh sáng bức xạ từ nguồn. Độ chói

của đối tượng phụ thuộc vào lượng ánh sáng chiếu vào bề mặt của đối tượng và tính chất phản chiếu của nó. Độ sáng là một đặc tính vật lý và có thể đo được bằng cách sử dụng quang kế. *Độ tương phản* liên quan đến độ chói: nó là một hàm của độ chói đối tượng và độ chói của nền.

Mặc dù, độ sáng là một phản ứng chủ quan, nhưng nó có thể được miêu tả theo thuật ngữ của độ chói: là lượng độ chói tạo nên sự khác biệt rõ ràng trong độ sáng. Tuy nhiên, bản thân hệ thống thị giác cũng bù đắp cho những thay đổi trong độ sáng. Trong ánh sáng mờ, các tế bào hình que chiếm đa số. Do có nhiều tế bào hình que trên hốc mắt hơn, các đối tượng trong ánh sáng yếu sẽ khó nhìn hơn trên võng mạc và sẽ dễ nhìn hơn trong thị giác ngoại biên. Ngược lại, trong ánh sáng bình thường, các tế bào hình nón sẽ chiếm đa số.

Tính sắc bén của thị giác tăng khi độ chói tăng. Điều này đang là vấn đề gây tranh cãi khi ý tưởng sử dụng màn hình hiển thị có độ chói cao được đưa ra. Tuy nhiên, khi độ chói tăng, sự lập loè cũng tăng. Mắt sẽ chỉ thu nhận được một ánh sáng lập loè liên tục. Khi độ chói cao, sự lập loè tăng. Mặt khác, sự lập loè cũng dễ nhận biết hơn trong thị giác ngoại biên. Điều này có nghĩa là màn hình hiển thị càng lớn, độ lập loè sẽ xuất hiện càng nhiều.

**Thu nhận màu sắc:** Một nhân tố thứ ba mà chúng ta cần xem xét là sự thu nhận màu sắc. Màu thường được tạo thành từ 3 thành phần chính: sắc độ, cường độ và độ bão hoà (độ no màu). Sắc độ được xác định bởi bước sóng ánh sáng trong quang phổ. Các màu xanh da trời có bước sóng ngắn, các màu xanh lá cây có bước sóng trung bình và các màu đỏ có bước sóng dài. Trung bình một người có thể phân biệt được khoảng 150 sắc độ khác nhau. Cường độ là độ sáng của màu sắc, và độ bão hoà (độ no màu) là tổng số lượng màu trắng có trong màu. Bằng cách thay đổi cường độ và độ bão hoà, chúng ta có thể thu được khoảng 7 triệu màu khác nhau. Tuy nhiên, số lượng màu mà một người bình thường không được đào tạo có thể nhận biết được thì ít hơn rất nhiều, chỉ khoảng 10 màu.

Mắt cảm nhận được màu sắc là do các tế bào hình nón nhạy cảm với ánh sáng có các bước sóng khác nhau. Có 3 loại tế bào hình nón khác nhau, mỗi loại nhạy với một màu khác nhau (xanh da trời, xanh lá cây và đỏ). Màu được nhìn tốt nhất nằm ở hốc mắt và xấu nhất ở ngoại biên nơi mà các tế bào hình que chiếm đa số. Người ta thấy rằng các tế bào hình nón nhạy ánh sáng màu xanh da trời chỉ chiếm có 3-4 % hốc mắt, do đó tính sắc nét của màu xanh da trời thấp hơn các màu khác.

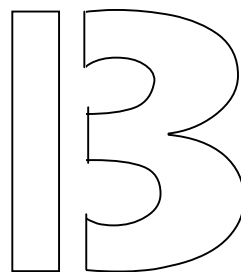
Cuối cùng, chúng ta hãy nhớ rằng chỉ có khoảng 8 % đàn ông và 1 % phụ nữ là không mắc bệnh mù màu, còn hầu hết trong số chúng ta đều không có khả năng phân biệt giữa màu đỏ và màu xanh lá cây. Nghĩa là, khả năng thu nhận màu sắc của con người là rất hạn chế.

### **2.2.1.3. Khả năng và hạn chế của xử lý thị giác**

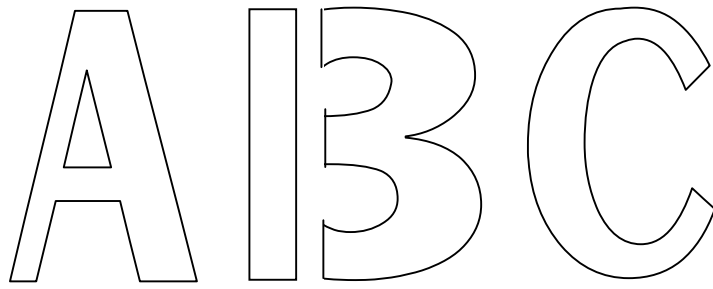
Trong các phần trước, chúng ta đã đề cập đến một số khả năng và hạn chế trong hệ thống xử lý thị giác của con người. Tuy nhiên, chúng ta chỉ mới tập trung phần lớn vào việc thu nhận ở mức thấp. Xử lý thị giác còn bao hàm sự biến đổi và diễn dịch một hình ảnh hoàn chỉnh. Ta thấy rằng, sự kì vọng của chúng ta có thể ảnh hưởng đến cách cảm nhận hình ảnh. Ví dụ, nếu ta đã biết trước kích thước của đối tượng, thì chúng ta sẽ chỉ cảm nhận nó với kích thước thật đó mà không quan tâm đến khoảng cách từ nó đến chúng ta là bao nhiêu.

Xử lý thị giác bổ sung cho sự chuyển động của hình ảnh trên võng mạc, xảy ra khi chúng ta dịch chuyển vòng quanh và khi đối tượng mà ta nhìn bị dịch chuyển. Mặc dù, hình ảnh trên võng mạc dịch chuyển, nhưng hình ảnh thực tế mà ta cảm nhận vẫn cố định. Tương tự như vậy, màu và độ sáng của đối tượng được cảm nhận cũng không thay đổi, mặc dù có sự thay đổi về độ chói.

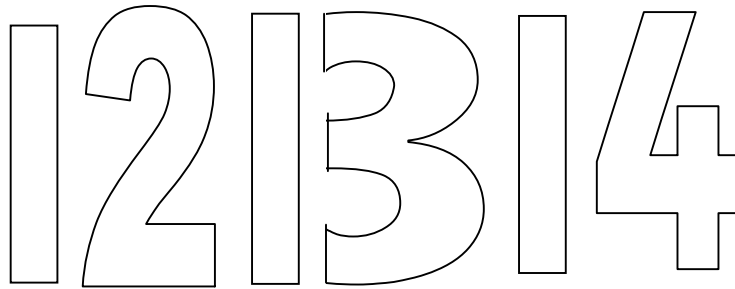
Khả năng diễn dịch và khám phá những ước đoán của chúng ta có thể sử dụng để giải quyết sự nhập nhằng. Ví dụ, ta hãy xem hình ảnh ở trong hình 1.4. Bạn có thể cảm nhận được gì? Thật khó hiểu!. Tuy nhiên, hãy xem tiếp hình ảnh trong hình 1.5 và hình 1.6. Những kí tự được thêm vào đã làm cho những ước đoán của chúng ta trở nên rõ ràng hơn. Do đó việc hiểu đối tượng cũng dễ hơn: Đối tượng chỉ có thể là A B hoặc A 1 3.



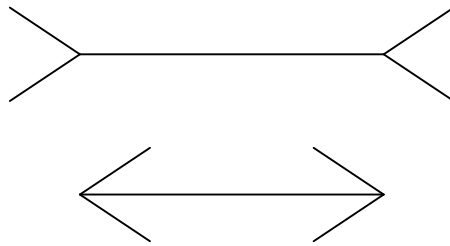
Hình 1.4: Một hình vẽ khó hiểu



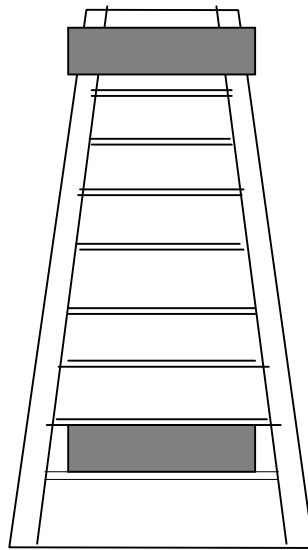
Hình 1.5: ABC



Hình 1.6: 12 13 14



Hình 1.7: Ảo giác quang học



Hình 1.8: Ảo giác Ponzo



Tuy nhiên, điều đó cũng tạo ra các ảo ảnh quang học. Ví dụ, xem hình 1.7. Bạn thấy đường nào dài hơn? Hầu hết trong số những người được hỏi đều trả lời một cách rất chắc chắn rằng đường nằm ở phía trên dài hơn đường nằm phía dưới. Thực tế là hai đường có chiều dài bằng nhau. Điều này có thể là do sự áp dụng sai quy tắc về kích thước không đổi: đường nằm trên xuất hiện với 2 ngạnh ở 2 đầu toả ra, còn đường nằm dưới với 2 ngạnh ở 2 đầu lại chụm vào. Do đó 2 ngạnh của đường trên dường như dài hơn hai ngạnh của đường dưới và vì thế tạo cảm giác đường trên dài hơn đường dưới. Một ảo ảnh tương tự nữa là ảo ảnh Ponzo (hình 1.8). Trên hình vẽ hình chữ nhật ở trên đỉnh dường như có kích thước lớn hơn hình chữ nhật nằm phía dưới, nhờ hiệu ứng khoảng cách, mặc dù cả 2 hình chữ nhật có cùng kích thước. Những ảo ảnh đó cho thấy rằng sự cảm nhận về khoảng cách của chúng ta thật sự không hoàn toàn đáng tin cậy.

Một ảo ảnh khác được tạo ra do những ước đoán của chúng ta bổ sung vào cho hình ảnh là ảo ảnh về việc đọc và sửa bản in thử. Hãy đọc lướt nhanh đoạn văn bản ở hình 1.9. Bạn có phát hiện ra điều gì sai không? Hầu hết mọi người khi đọc nhanh đều thấy rằng nó hoàn toàn chính xác, mặc dù là nếu để ý kỹ hơn ta sẽ thấy rằng từ “the” đã bị lặp lại ở dòng thứ hai và thứ ba.

The quick brown  
fox jumps over the  
the lazy dog.

Hình 1.9: Đoạn text này có chính xác?

Trên đây là một vài ví dụ về cách một hệ thống thị giác bổ sung cho hình ảnh, dù đôi khi là bổ sung thừa, để cho phép chúng ta cảm nhận được thế giới quanh ta.

**Đọc (Reading):** Chúng ta đã tập trung quá nhiều vào vấn đề thu nhận các hình ảnh mà quên mất rằng vấn đề cảm nhận và xử lý văn bản, một trường hợp cụ thể của vấn đề cảm nhận hình ảnh, cũng rất cần được xem xét. Trong thiết kế giao diện, điều này rất quan trọng vì trong quá trình thiết kế, đôi khi đòi hỏi việc hiển thị các đoạn văn bản. Do đó chúng ta sẽ kết thúc phần này bằng cách tìm hiểu về reading. Xử lý đọc bao gồm nhiều giai đoạn. Đầu tiên, mẫu của từ trên trang giấy sẽ được thu nhận.

Sau đó, nó được giải mã bằng cách đối chiếu với một các mẫu cơ bản của ngôn ngữ. Giai đoạn cuối cùng của việc xử lý ngôn ngữ là phân tích cú pháp và ngữ nghĩa, phân tích các cụm từ hoặc các câu.

Chúng ta chỉ quan tâm đến 2 giai đoạn đầu của quá trình này và những ảnh hưởng của chúng đến thiết kế giao diện. Trong khi đọc, mắt thực hiện các chuyển động lên xuống được gọi là sự di chuyển của mắt. Thu nhận xảy ra trong giai đoạn tập trung ánh mắt. Mắt di chuyển tiến, lùi qua các đoạn văn bản, và được gọi là các chuyển dịch tiến lui. Đoạn văn bản càng phức tạp, thì sẽ có càng nhiều các chuyển dịch tiến lui.

Người trưởng thành đọc được khoảng 250 từ trong 1 phút. Các từ có thể không được đọc lướt qua theo một cách tuần tự, kí tự nọ kế tiếp kí tự kia, vì những thí nghiệm cho thấy rằng nhận dạng từ có thể nhanh tương đương so với nhận dạng các kí tự đơn. Thay vì vậy, các từ quen thuộc sẽ được nhận dạng qua hình dạng của từ. Điều này có nghĩa là nếu bỏ đi các hình dạng quen thuộc của từ (ví dụ, bằng cách viết hoa các từ) sẽ ảnh hưởng có hại đến tốc độ đọc và độ chính xác.

Tốc độ đọc là một thước đo để đánh giá tính dễ đọc của đoạn văn bản. Các thí nghiệm cho thấy rằng, các kích thước font tiêu chuẩn từ 9 đến 12 points và các chiều dài của dòng từ 2.3 đến 5.2 inches (58 và 132 mm) thì đều có tính dễ đọc ngang nhau. Tuy nhiên, thực tế là việc đọc từ màn hình máy tính thường chậm hơn so với việc đọc từ sách. Điều này là do các nhân tố như: độ dài dòng dài hơn, lượng từ trên một trang ít hơn, và sự quen thuộc với phương tiện giấy ảnh hưởng đến. Những nhân tố đó có thể giảm bớt được bằng cách thiết kế cẩn thận các giao diện văn bản.

Vấn đề cuối cùng là sử dụng độ tương phản trong hiển thị: một độ tương phản âm (các kí tự đen trên nền màn hình trắng) sẽ tạo ra độ chói cao hơn và, do đó, làm tăng tính sắc nét hơn, so với một độ tương phản dương. Điều này cũng sẽ làm tăng tính dễ đọc. Tuy nhiên, độ tương phản âm cũng có thể sẽ xảy ra sự lập loè, không ổn định. Tuy nhiên, thực tế cho thấy rằng các màn hình hiển thị có độ tương phản âm được ưa chuộng hơn và cho phép hiển thị chính xác hơn.

### **2.2.2. Thính giác**

Thính giác được xét đến ngay sau thị giác, nhưng chúng ta thường đánh giá chưa đúng về lượng thông tin mà chúng ta nhận được thông qua tai của mình. Bạn hãy nhắm mắt lại một chút và lắng nghe. Bạn có thể nghe thấy những âm thanh gì? Chúng

**Error! No index entries found.**từ đâu đến? Và do cái gì phát ra? Khi tôi ngồi trong văn phòng làm việc, tôi nghe thấy tiếng của rất nhiều chiếc xe ô tô đang đi trên đường, máy móc đang hoạt động ở một vị trí gần đó, tiếng kêu vo vo của chiếc máy bay ở trên đầu và tiếng chim hót. Và tôi cũng có thể biết được những âm thanh đó từ đâu đến, và ước tính xem chúng cách xa bao nhiêu. Vì từ những âm thanh nghe được, tôi có thể biết được rằng có một chiếc xe đang đi trên một con đường ngay gần nhà tôi, và hướng mà nó đang đi. Tôi cũng biết được vị trí của công trường và loài chim nào đang hót trên cành cây sau vườn nhà tôi.

Hệ thống thính giác có một khả năng vô cùng to lớn trong việc truyền đạt thông tin về môi trường quanh ta. Vậy nó làm việc như thế nào?

### **2.2.2.1. Tai người**

Thính giác bắt đầu với những dao động trong không khí hoặc hoặc các sóng âm thanh. Tai tiếp nhận những dao động đó và truyền chúng, qua các giai đoạn khác nhau, đến các dây thần kinh thính giác. Tai bao gồm 3 phần, thường được gọi là, tai ngoài, tai giữa và tai trong.

Tai ngoài là phần nhìn thấy được của tai. Nó có 2 phần: loa tai, là cấu trúc được gắn vào 2 bên đầu, và ống thính giác, sóng âm thanh được truyền dọc theo những ống này đến tai giữa. Tai ngoài có 2 mục đích: Thứ nhất, nó bảo vệ phần tai giữa để bị tổn thương khỏi nguy hiểm. Ống thính giác chứa ráy tai ngăn bụi, bẩn và côn trùng vào phần tai giữa. Nó cũng duy trì cho tai giữa luôn ở một nhiệt độ không đổi. Thứ hai, loa tai và ống thính giác còn dùng để khuếch đại âm thanh.

Tai giữa là một khoang nhỏ nối với tai ngoài qua màng nhĩ và với tai trong qua ốc tai. Bên trong khoang là các xương nhỏ, các xương nhỏ nhất trong cơ thể con người. Sóng âm thanh đi dọc theo ống thính giác và làm rung màng nhĩ, làm cho các xương nhỏ dao động theo, truyền các dao động đến ốc tai và đi vào tai trong. Sự “trì hoãn” này là cần thiết vì không giống như tai giữa và tai ngoài chứa đầy không khí, tai trong chứa chất dịch ốc tai đặc hơn. Nếu truyền trực tiếp từ không khí vào chất dịch này, sóng âm thanh sẽ bị yếu đi. Bằng cách truyền chúng qua các xương nhỏ, sóng âm thanh sẽ được tập trung và được khuếch đại.

Sóng được truyền vào trong chất dịch ốc tai ở tai trong. Bên trong ốc tai là các tế bào rất nhỏ, hay còn gọi là các lông mao. Lông mao sẽ bị cong đi do sự dao động

trong chất dịch ốc tai và phát ra một tín hiệu hoá học để tạo ra các xung thần kinh thính giác.

#### **2.2.2.2. Xử lý âm thanh**

Như chúng ta đã biết, âm thanh là những thay đổi hay dao động của áp suất không khí. Nó có rất nhiều đặc trưng để chúng ta có thể phân biệt. Cường độ là tần số của âm thanh. Một tần số thấp sẽ tạo ra một cường độ thấp, và ngược lại, tần số cao thì cường độ cao. Độ to tỷ lệ với biên độ của âm thanh, khi tần số không thay đổi. Âm sắc liên quan đến các loại âm thanh: các âm thanh có thể có cùng cường độ và độ to nhưng nếu chúng được tạo ra từ các nhạc cụ khác nhau thì âm sắc của chúng sẽ khác nhau. Chúng ta cũng có thể nhận dạng được vị trí của một âm thanh, vì hai tai nhận được những âm thanh hầu như không khác nhau. Điều này có được là nhờ sự khác nhau về thời gian âm thanh đến hai tai và sự sụt giảm về cường độ do sóng âm thanh phản hồi từ đầu.

Tai người có thể nghe được các tần số từ khoảng 20 Hz đến 15 kHz. Nó có thể phân biệt được các thay đổi tần số trong phạm vi nhỏ hơn 1.5 Hz đối với các tần số thấp, và sẽ phân biệt ít chính xác hơn với các tần số cao. Các tần số khác nhau kích thích hoạt động của các nơ ron thần kinh ở các phần khác nhau trong hệ thống thính giác và tạo ra các tỷ lệ chấy khác nhau của các nơ ron thần kinh.

Hệ thống thính giác thực hiện chức năng lọc đối với các âm thanh nhận được, cho phép chúng ta bỏ qua phần tín hiệu nhiễu mà chỉ tập trung vào các thông tin quan trọng. Chúng ta có quyền chọn lựa những thông tin mà mình muốn nghe, ví dụ như trong một bữa tiệc cốc tai có rất nhiều âm thanh hỗn loạn nhưng nếu có một ai đó gọi tên của ta qua một căn phòng ồn ào, não nhiệt ta vẫn có thể nghe thấy tiếng gọi đó. Tuy nhiên, nếu âm thanh quá ồn, hoặc tần số của nó quá nhỏ, chúng ta sẽ không có khả năng phân biệt được các âm thanh khác nhau.

Như ta đã biết, âm thanh có thể mang theo một lượng thông tin rất lớn. Hiện tại, người ta ít khi sử dụng tiềm năng này của âm thanh trong khi thiết kế giao diện, mà thường chỉ giới hạn trong các âm thanh cảnh báo. Tuy nhiên, tai có thể phân biệt các thay đổi âm thanh khá tinh tế và có thể tổ chức lại thành các âm thanh quen thuộc mà không cần phải tập trung chú ý đến nguồn âm. Điều này cho thấy rằng âm thanh có thể được sử dụng một cách rộng rãi hơn trong quá trình thiết kế giao diện., ví dụ, để chuyển thông tin về trạng thái của hệ thống.

### 2.2.3. Xúc giác

Giác quan thứ ba và là giác quan cuối cùng mà chúng ta sẽ xem xét là xúc giác. Mặc dù giác quan này ít quan trọng hơn so với thị giác và thính giác, nhưng cuộc sống của con người lại không thể thiếu nó. Xúc giác cho chúng ta những thông tin cần thiết về môi trường quanh ta. Nó thông báo cảm giác mà chúng ta sẽ có được khi cầm nắm vào một vật và do đó nó đóng vai trò như một cảnh báo. Ví dụ, khi ta sờ vào một vật gì đó nóng ta có cảm giác nóng. Xúc giác cũng cung cấp thông tin phản hồi, khi chúng ta vận động cơ (ví dụ cố gắng nâng một đối tượng). Ta hãy xem hành động nâng một cốc nước. Nếu chúng ta chỉ nhìn vào chiếc cốc và không cảm giác được khi nào thì bàn tay của ta chạm vào cốc hoặc không cảm nhận được hình dạng của chiếc cốc thì tốc độ và độ chính xác của hành động sẽ giảm đi rất nhiều. Điều này là thường gặp đối với những người chơi game trong thế giới hiện thực ảo: họ có thể nhìn thấy các đối tượng được tạo ra bởi máy tính mà họ cần thao tác nhưng họ thực sự không có được cảm giác tự nhiên về việc cầm nắm hoặc tiếp xúc trực tiếp với các đối tượng đó một cách thực sự. Đây chính là một kinh nghiệm thú vị cho những người thiết kế đồ họa và thiết kế giao diện. Do đó, xúc giác là phương tiện quan trọng của phản hồi và thực tế là trong các hệ thống máy tính việc sử dụng các thông tin phản hồi là tương đối nhiều. Chúng ta biết rằng, mặc dù đối với một người bình thường, xúc giác chỉ là nguồn thông tin đứng vào hàng thứ hai, nhưng đối với những người mà các giác quan khác như thị giác hoặc thính giác bị hỏng, thì xúc giác sẽ trở nên vô cùng quan trọng. Với những người sử dụng như vậy, các giao diện như là hệ thống chữ nổi sẽ trở thành nguồn thông tin cơ bản cho tương tác. Do đó, chúng ta không nên đánh giá thấp tầm quan trọng của xúc giác.

Không giống như bộ máy thị giác và thính giác, bộ máy xúc giác không có một phân khu cố định. Chúng nhận kích thích thông qua da. Da chứa 3 loại cơ quan thụ cảm: cơ quan thụ cảm nhiệt phản ứng lại với nóng và lạnh, cơ quan thụ cảm thần kinh phản ứng lại với sức ép căng thẳng, sự nóng giận và đau đớn, và cơ quan thụ cảm cơ phản ứng lại với áp lực. Chúng ta chỉ quan tâm đến cơ quan thụ cảm cơ vì chúng liên quan đến tương tác người máy.

Có hai loại cơ quan thụ cảm cơ tương ứng với 2 loại áp lực khác nhau: Thụ cảm cơ thích ứng nhanh phản ứng lại với áp lực ngay lập tức sau khi da nhận kích thích. Khi áp lực càng gia tăng thì chúng phản ứng lại càng nhanh. Tuy nhiên, chúng sẽ

ngừng phản ứng nếu áp lực tác động lên là áp lực liên tục. Thụ cảm cơ thích ứng chậm phản ứng với các áp lực tác động một cách liên tục.

Mặc dù, toàn bộ cơ thể đều có các cơ quan thụ cảm như vậy, nhưng vẫn tồn tại những vùng cơ thể có độ nhạy cảm hoặc có tính nhạy bén cao hơn những vùng khác. Có thể đánh giá tính nhạy bén của các vùng khác nhau bằng cách sử dụng phương pháp kiểm tra ngưỡng 2 điểm. Lấy 2 chiếc bút chì, giữ để cho đầu của chúng cách nhau khoảng 12 mm. Đặt ngón tay cái lên 2 hai đầu, và xem xem bạn có thể cảm nhận được cả 2 đầu bút không? Nếu bạn không thể, hãy di chuyển để 2 đầu bút ra xa nhau hơn một chút nữa. Khi nào bạn cảm nhận được 2 đầu bút, hãy ghi lại khoảng cách giữa chúng. Khoảng cách càng lớn, độ nhạy cảm càng bé. Bạn có thể lặp lại bài kiểm tra này trên các phần khác nhau của cơ thể. Khi kiểm tra xong bạn sẽ thấy rằng độ nhạy cảm của các ngón tay là lớn nhất.

Một khía cạnh thứ hai của xúc giác là sự vận động: nhận biết vị trí của cơ thể và các chi. Điều này là do các cơ quan thụ cảm nằm ở các điểm đó. Có 3 loại: thích ứng nhanh, phản ứng khi một chi di chuyển theo một hướng cụ thể; thích ứng chậm, phản ứng với sự chuyển động và vị trí đứng yên; và các cơ quan thụ cảm vị trí, chỉ phản ứng khi một chi nằm ở vị trí cố định. Cảm nhận về sự cử động này tác động sự thoải mái và hiệu quả thực hiện công việc. Ví dụ, đối với một người đánh máy chữ, việc cảm nhận được cả vị trí tương đối của các ngón tay và thông tin phản hồi từ bàn phím là rất quan trọng, và ảnh hưởng đến tốc độ đánh.

#### **2.2.4. Sự chuyển động**

Trước khi tìm hiểu các kênh vào- ra của con người, chúng ta sẽ xem xét vấn đề điều khiển vận động và cách di chuyển của chúng ta có tác động như thế nào đến tương tác của ta với máy tính. Một hành động đơn giản, như là ấn phím để giành quyền trả lời một câu hỏi, bao gồm nhiều giai đoạn xử lý. Kích thích (câu hỏi) được nhận thông qua các cơ quan thụ cảm và được truyền đến não. Câu hỏi được xử lý và một câu trả lời được tạo ra. Sau đó, não bộ sẽ chuyển tín hiệu điều khiển đến các cơ thích hợp để trả lời. Mỗi một giai đoạn đều đòi hỏi phải có thời gian và chúng có thể được phân chia một cách tương đối thành thời gian phản ứng lại và thời gian lan truyền.

Thời gian lan truyền phụ thuộc phần lớn vào các đặc điểm vật lý của đối tượng, ví dụ như là tuổi tác và sức khỏe. Và thời gian phản ứng phụ thuộc vào giác quan nhận

kích thích. Một người bình thường có thể phản ứng lại với một tín hiệu thính giác trong khoảng 150ms, với một tín hiệu thị giác trong 200ms và với sự đau đớn trong 700ms. Tuy nhiên, một tín hiệu hỗn hợp sẽ tạo ra phản ứng có thời gian nhanh nhất. Các nhân tố như là kỹ năng hoặc sự luyện tập có thể làm giảm thời gian phản ứng lại, và ngược lại sự mệt mỏi có thể sẽ làm gia tăng thời gian phản ứng.

Một thước đo thứ hai của kỹ năng vận động là tính chính xác. Có một câu hỏi đặt ra là liệu rằng tốc độ của sự phản ứng lại có ảnh hưởng đến tính chính xác? Điều này hoàn toàn phụ thuộc vào người sử dụng và nhiệm vụ của họ. Trong một số trường hợp, yêu cầu thời gian phản ứng lại nhanh sẽ làm giảm tính chính xác. Ví dụ, trong các trò chơi game, những người chơi game thiếu kỹ năng thường không thể nào chơi được ở các mức yêu cầu người chơi phải có phản ứng nhanh, vì họ không thể xử lý chính xác được ở tốc độ cao. Ngược lại, đối với những người chơi có kỹ năng thì những mức đó là phù hợp. Trong một số trường hợp khác, thời gian phản ứng lại không có ảnh hưởng nhiều đến tính chính xác. Ví dụ, một nghiên cứu về những người đánh máy cho thấy rằng: những người đánh nhanh nhất cũng chỉ đánh chính xác gấp đôi so với những người khác và những người chậm nhất cũng chỉ sai khoảng 10 lỗi.

Tốc độ và độ chính xác của sự di chuyển là rất quan trọng trong thiết kế các hệ thống tương tác, khi tính đến thời gian cần phải di chuyển đến một đích đến cụ thể trên màn hình. Đích đến có thể là một nút, một biểu tượng, hay một khoản mục. Thời gian cần thiết để đi đến đích là một hàm của kích thước của đích với khoảng cách cần phải dịch chuyển. Đây là công thức Fitts. Có rất nhiều cách phát biểu công thức trên, tất cả đều tương đương nhau, do đó có thể phát biểu như sau:

$$\text{Thời gian dịch chuyển} = a + b \log_2 (\text{khoảng cách/kích thước} + 1)$$

trong đó a và b là các hằng số được xác định theo kinh nghiệm thực tế

Điều này tác động đến loại đích đến mà chúng ta thiết kế. Vì người dùng thường rất khó khăn khi thao tác với các đối tượng nhỏ, do đó các đích đến thường được thiết kế sao cho nó có kích thước đủ lớn và khoảng cách mà ta cần phải di chuyển là nhỏ nhất. Điều đó dẫn đến ý tưởng về loại menu hình biểu đồ tròn, dùng để liệt kê, trong đó tất cả các lựa chọn đều có khoảng cách tương đương nhau. Trong số các danh sách được liệt kê, lựa chọn nào thường dùng nhất phải được đặt ở vị trí gần với điểm bắt đầu của người sử dụng nhất (ví dụ, ở trên đỉnh của menu).

## 2.3. Bộ nhớ

### 2.3.1. Bộ nhớ ngắn hạn

Bộ nhớ ngắn hạn (STM) là một bộ nhớ của con người tương đương với bộ nhớ RAM của máy tính, nói một cách khác là bộ nhớ hoạt động của bộ xử lý trung tâm. Nó khác với máy tính ở chỗ, bộ nhớ ngắn hạn của con người nhỏ và hay đánh mất các nội dung của nó trừ khi nó tự nhớ lại sau mỗi 200ms. Tuy nhiên, thời gian nhận thông tin qua việc đọc, viết khoảng 70ms, là khá nhanh, vậy nên thông tin có thể được giữ lại trong bộ nhớ STM bằng cách viết đi viết lại liên tục. Theo như mô hình mẫu xử lý thông tin, bộ nhớ ngắn hạn phải lưu giữ thông tin từ nhiều nguồn mặc dù sức chứa của nó là rất giới hạn.

Năm 1956, Miller đã tiến hành các thử nghiệm để xác định giới hạn của bộ nhớ ngắn hạn của con người. Ông đã đi kết luận rằng, trung bình thì một người bình thường có thể nhớ được khoảng  $7 \pm 2$  khoản mục. Các khoản mục này không được lưu trữ trong bộ nhớ của máy tính dưới dạng các “bytes” mà là dưới dạng các “chunks” thông tin (đoạn thông tin) chúng có thể biến đổi từ những con số và các kí tự đơn giản thành các hình ảnh và các khái niệm trừu tượng phức tạp. Bí mật của việc mở rộng khả năng lưu trữ giới hạn trong bộ nhớ STM là chuyển thông tin cơ bản thành các đặc tính và thay vào đó là lưu trữ cách nhìn trừu tượng.

Để hiểu được ví dụ này một cách tốt nhất ta phải xem xét ví dụ sau đây: Đưa ra các mã điện thoại không theo thứ tự, ví dụ như 01612363311; những con số lớn như vậy rất là khó nhớ, nhưng khi tách chúng ra thành các đơn vị nhỏ hơn thì sẽ rất dễ nhớ, ví dụ tách chúng ra thành 0161-236-3311. Mục đích của nó là đưa ra cho người đọc gợi ý về cách phân các đoạn. Thay cho việc lưu trữ 10 con số tách biệt, các nhóm con số được lưu giữ dưới dạng các đoạn, như vậy ta có thể giảm được từ lưu giữ 10 đoạn xuống còn 3 đoạn. Việc sắp đặt thành trật tự có thể được áp dụng cho các dữ liệu chưa xử lý càng nhiều thì việc phân đoạn càng tốt. Để tự nhận thức được điều này hãy ghi nhớ nhanh những điều sau:

832751984221- Việc gọi lại chính xác sẽ rất hiếm khi chính xác

83-275-1984-221: Sẽ không có vấn đề gì xảy ra khi nhưng bạn hãy dùng những quy tắc gì để phân đoạn những con số này?



Các con số thứ hai và thứ 3 có trật tự riêng nên dễ phân đoạn. Những gì đã được lưu trữ chính là số lượng các dữ liệu được sử dụng để hội tụ lại, trong trường hợp cuối cùng thuật toán của bộ ba các số chẵn/lẻ xếp trong dãy các số tiền.

Một nghiên cứu tiếp theo chỉ ra rằng mẫu xử lý thông tin của con người khá đơn giản. Bộ nhớ STM có ít nhất 2 hệ thống phụ: một hệ chuyên giải quyết những dữ liệu ngôn ngữ cơ bản, hệ kia giải quyết thông tin nhìn được trong không gian. Các chức năng của hệ thống phụ về ngôn ngữ học dưới dạng một danh sách nhưng việc truy nhập thông tin giống như một hàng LIFO lai. Ta thường có xu hướng nhớ một vài mục sau cùng, và trước tiên trong một danh sách và thường dễ quên những mục ở giữa. Việc lưu giữ và nhớ lại nhìn chung xảy ra liên tục. Bộ nhớ đầu vào đối với tiếng nói giống như một vạch vòng mà tại đó các từ được đặt thành các câu. Khi ta quên mất mạch mà ta đang nói, thì ta nhớ lại các từ được ghi tại các phần khác của mạch vòng. Toàn bộ hệ thống ngắn hạn, mà được gọi là bộ nhớ làm việc, được điều khiển bởi một bộ xử lý làm việc, nó tương tự như khái niệm đối với bộ xử lý nhận thức. Mô hình tỉ mỉ hơn này giúp cho việc giải thích sự khác nhau trong bộ nhớ tạm thời đối với việc nhớ các thông tin nhìn thấy và thông tin văn bản, và giải thích về sự nhiều khi sự ghi nhớ làm giảm việc nhớ thông tin. Trong trường hợp sau bộ nhớ xử lý hoạt động bị lãng quên do trong quá trình lưu giữ và nhớ lại nội dung của bộ nhớ làm việc.

Một vài đặc điểm chính của bộ nhớ làm việc:

- Thời gian truy cập thông tin đọc, viết nhanh, khoảng 70ms
- Bộ nhớ bị loãng thông tin rất nhanh-sau mỗi 200ms nó lại tự nhớ lại
- Khả năng thu nhận bị giới hạn tới  $7 \pm 2$  đoạn, nhưng có thể lớn hơn đối với các chi tiết hình ảnh
- Có thể tăng khả năng lưu giữ bằng cách tóm tắt số lượng các thông tin chưa xử lý
- Việc xao nhãng làm cho quên đi các thông tin nhận biết được gần đây. Thậm chí một con số nhỏ trong các đoạn tin đơn giản cũng sẽ bị quên đi nếu như có một hiện tượng đứt quãng xảy ra trong quá trình tiếp nhận thông tin.
- Những lượng vào khác làm suy yếu đi khả năng gọi lại. Việc cung cấp lượng thông tin không thích hợp trong quá trình nhập lượng thông tin vào trong bộ nhớ làm việc sẽ làm cho khả năng gọi lại rất kém.

- Chính những đầu vào tương tự làm suy yếu khả năng gọi lại. Việc cung cấp các khoản mục liên quan trong thời gian ghi nhớ làm cho việc gọi lại gặp khó khăn.
- Bộ nhớ ngay lập tức các chi tiết trong hình ảnh phức tạp rất nghèo nàn.
- Gọi lại các khoản mục sẽ dễ hơn khi cả từ và hình ảnh của chúng được xuất hiện đồng thời, điều này được so sánh với hình ảnh hoặc từ khi bị tách ra.
- Con người nhớ trong một khoảng thời gian ngắn (<30s) bằng cách quét ngược lại đầu vào, vào sau ra trước.

Bộ nhớ làm việc là một trong số những giới hạn chủ chốt trong quá trình xử lý thông tin của con người. Nó tạo ra nhiều kết quả đối với thiết kế giao diện và những kết quả này có thể được thể hiện như sau:

- Tối thiểu hoá sự đứt quãng trong khi ghi nhớ
- Chú ý tới sự quá tải của bộ nhớ làm việc cả về số lượng thông tin và khoảng thời gian ghi nhớ.
- Cấu trúc (đoạn) thông tin giúp cho dễ ghi nhớ
- Các hình ảnh rất có tác dụng trong việc ghi nhớ nhưng cần phải đi kèm theo văn bản.

### **2.3.2. Bộ nhớ dài hạn**

Bộ nhớ dài hạn là tệp lưu trữ chính trong hệ thống con người. Nó có khả năng thu nhận vô hạn, vì không ai có khả năng chứng minh được một giới hạn cao hơn về những gì mà con người có thể ghi nhớ. Nếu thiếu bộ nhớ thì ta sẽ không có khả năng nhớ lại những gì đã xảy ra.

Việc nhớ lại những sự kiện từ bộ nhớ có thể được thực hiện rất nhanh, đặc biệt là đối với các khoản mục và các thủ tục được sử dụng thường xuyên. Thời gian nhớ lại đối với thông tin được sử dụng ít thường xuyên hơn luôn thay đổi; khoảng thời gian này có thể rất nhanh, nhưng cũng có thể rất lâu. Đặc biệt là đối với những người già. Việc nhớ lại theo mô hình xử lý thông tin là một chức năng của bộ xử lý nhận biết, nhưng trên thực tế quá trình xử lý thông tin này rất phức tạp. Thông thường, việc nhớ một sự kiện không xảy ra ngay lập tức, thay vào đó nó phải cần vài phút sau khi cố gắng nhớ lại. Trong khoảng thời gian ở giữa, sự chú ý sẽ được dành cho những vấn đề khác, vì lí do này mà bộ xử lý bộ nhớ gốc phải được dùng để tác động tới sự tìm kiếm

của bộ nhớ dài hạn. Việc dừng lại sẽ kích thích hoạt động của bộ nhớ, vậy nên những khoản mục được sử dụng thường xuyên hoặc mới được sử dụng sẽ dễ dàng nhớ lại hơn. Bộ nhớ cũng xuất hiện dưới dạng một quá trình xử lý gồm 2 giai đoạn, vậy nên bạn có thể thường xuyên nói một sự kiện nào đó nhưng bạn lại không thể nhớ lại nó được một cách dễ dàng. Đây là kết quả của:

- Sự nhận biết: hoạt động đầu tiên của một bộ nhớ được thể hiện bằng các tín hiệu
- Nhớ lại: tự nhớ lại thông tin

**Tổ chức của bộ nhớ:** Có 2 loại bộ nhớ dài hạn đó là: bộ nhớ đoạn và bộ nhớ ngữ nghĩa.

- Bộ nhớ đoạn: Dùng để biểu diễn sự ghi nhớ của chúng ta về các sự kiện và các kinh nghiệm theo một chuỗi liên tục. Bộ nhớ đoạn giúp chúng ta có thể nhớ lại các sự kiện thực tế đã xảy ra tại một thời điểm nào đó trong cuộc đời.
- Bộ nhớ ngữ nghĩa: Là một bản ghi có cấu trúc về các sự việc, các khái niệm và các kỹ năng mà chúng ta đã đạt được. Thông tin trong bộ nhớ ngữ nghĩa được kế thừa từ những kinh nghiệm trong bộ nhớ đoạn, những kinh nghiệm này giúp cho chúng ta có thể học được các khái niệm hoặc các sự việc mới.

## CHƯƠNG III: MÁY TÍNH

### 3.1. Các thiết bị nhập

Nếu ví CPU là bộ não của máy tính, thì các thiết bị nhập chính là tai mắt của nó. Khả năng sử dụng của thiết bị nhập có ảnh hưởng đến toàn bộ kết quả làm việc của bạn. Một thiết bị nhập có chức năng đúng như tên gọi của nó, đó là cho phép bạn nhập thông tin và lệnh vào máy tính. Các thiết bị nhập phổ biến nhất là con chuột và bàn phím. Do hầu hết mọi người sử dụng đã quen với 2 loại thiết bị này, cho nên trong phạm vi tài liệu này, chúng ta sẽ không đi sâu tìm hiểu về các loại thiết bị này.

#### 3.1.1. Các thiết bị nhập chuẩn

- Bàn phím: Bàn phím là một trong những thiết bị ngoại vi đầu tiên được sử dụng với máy tính, và nó vẫn là thiết bị chủ yếu để nhập văn bản và số liệu. Mỗi bàn phím chuẩn gồm có khoảng 100 phím và mỗi phím gửi một tín hiệu khác nhau đến CPU. Bàn phím của máy tính cá nhân có nhiều kiểu dáng. Các kiểu khác nhau về kích cỡ, hình dạng, và cảm tính của người sử dụng, nhưng ngoại trừ một vài bàn phím có chức năng đặc biệt, thì hầu hết các bàn phím đều được sắp xếp gần giống như nhau.
- Con chuột: Con chuột là một thiết bị nhập, có thể chạy vòng vòng trên bề mặt phẳng (thường là một tấm đệm bàn phím hoặc mặt bàn), và điều khiển con trỏ. Con trỏ là một đối tượng trên màn hình, thường là một mũi tên, được sử dụng để chọn văn bản; để truy cập các thực đơn; và tương tác với các chương trình, các file hoặc dữ liệu xuất hiện trên màn hình.

#### 3.1.2. Các phương pháp nhập tự chọn

Mặc dù bàn phím và con chuột là các thiết bị nhập mà người ta thường sử dụng nhất, nhưng một vài cách không mẫu mực khác để nhập dữ liệu vào máy tính vẫn được sử dụng. Có 3 loại thiết bị nhập tự chọn hiện nay đó là: Các thiết bị nhập bằng tay, các thiết bị nhập quang học và các thiết bị nghe nhìn

- Các thiết bị nhập bằng tay: Hầu hết các thiết bị nhập đều được thiết kế để dùng bằng tay. Khác với bàn phím và con chuột, các thiết bị này có vẻ trực quan hơn và dễ sử dụng mà không cần tập luyện hay phải có kỹ năng gì đặc biệt. Một số thiết bị loại này đang được sử dụng hiện nay bao gồm:
  - ✓ Các loại bút: Các hệ thống dùng bút – bao gồm nhiều loại máy hỗ trợ cá nhân kỹ thuật số và các kiểu máy tính cầm tay khác - sử dụng một “cây

bút” để nhập dữ liệu. Thiết bị giống cây bút này có khi được gọi là cây kim. Bạn cầm bút trên tay và viết trên một tấm bảng đặc biệt hoặc viết trực tiếp lên màn hình. Bạn cũng có thể dùng bút để định vị, hay chọn lệnh giống như dùng chuột.

- ✓ Màn hình xúc giác: Màn hình xúc giác nhập dữ liệu bằng cách cho phép người sử dụng đặt tay trực tiếp lên màn hình, thường là để chọn một thực đơn. Hầu hết các máy tính có màn hình xúc giác đều sử dụng thiết bị cảm biến ở trong hoặc ở gần màn hình để nhận biết sự tiếp xúc của ngón tay. Màn hình xúc giác thích hợp cho những nơi mà chất bẩn và bụi làm mất công dụng của bàn phím và con chuột, và những nơi cần một giao diện đơn giản và dễ nhìn. Chúng rất phù hợp cho các ứng dụng đơn giản như các loại máy nói tự động hoặc các ki-ốt thông tin công cộng.
- ✓ Bộ điều khiển trò chơi: Bạn có thể không nghĩ rằng bộ điều khiển trò chơi cũng là một thiết bị nhập. Bộ điều khiển trò chơi cũng có thể coi là một thiết bị nhập vì trò chơi trên máy tính cũng là một chương trình, giống như một chương trình xử lý văn bản. Nó nhập dữ liệu từ người sử dụng, xử lý và xuất ra kết quả dưới dạng âm thanh và hình ảnh. Khi các bộ trò chơi vi tính trở nên tinh vi và trau chuốt hơn, thì các bộ điều khiển trò chơi cũng phải được phát triển để đáp ứng các đặc tính của chúng.
- Các thiết bị nhập quang học: Máy tính không thể nhìn được như con người, nhưng các kỹ thuật mới cũng cho phép máy tính sử dụng nguồn nhập từ ánh sáng. Công cụ này là loại thiết bị nhập quang học.
  - ✓ Máy đọc mã vạch: Loại thiết bị nhập phổ biến nhất sau bàn phím và con chuột là máy đọc mã vạch. Kiểu máy đọc mã vạch thường gặp nhất là máy quét hình phẳng thường có trong các siêu thị và các cửa hàng tổng hợp. Thiết bị này biến đổi mã vạch, mảng các vạch in trên sản phẩm, thành một mã máy tính có thể hiểu được. Máy đọc mã vạch phát ra một chùm tia sáng- thường là tia laser- được phản chiếu bởi hình ảnh mã vạch. Một máy dò độ nhạy sáng sẽ xác định hình ảnh mã vạch bằng cách nhận dạng các vạch đặc biệt ở hai đầu. Các vạch đặc biệt này khác nhau, từ đó máy đọc có thể biết được rằng mã vạch đã đọc từ trên xuống hay

từ phải sang. Sau khi máy dò đã xác định được mã vạch, nó biến đổi các số mã vạch riêng lẻ sang dạng số. Sau đó máy đọc mã vạch đưa con số này sang máy tính, giống như số đó được nhập vào từ bàn phím.

- ✓ Máy quét ảnh và máy nhận biết các kí tự quang học: Máy quét ảnh biến đổi bất kỳ một hình ảnh nào đã được in sang dạng điện tử bằng cách chiếu sáng hình ảnh và đưa cường độ của sự phản chiếu ánh sáng lên từng điểm. Loại máy quét ảnh sử dụng bộ lọc để phân tích thành phần màu sắc thành các màu gốc (đỏ, xanh lá cây, và xanh da trời) tại mỗi điểm. Màu đỏ, màu xanh lá cây, màu xanh da trời được gọi là các màu gốc vì chúng có thể được pha trộn để tạo ra bất kì màu nào khác. Cách thức nhận biết màu sắc theo kiểu này được gọi là sử dụng màu RGB.
- Các thiết bị nghe nhìn: Ngày nay, máy tính mới đã có các đặc tính cho phép chúng thu nhận âm thanh và hình ảnh và phát lại. Một số kiểu thiết bị thuộc loại này bao gồm:
  - ✓ Micro và máy nhận biết giọng nói: Hiện nay khả năng xử lý âm thanh là một tiêu chuẩn của máy tính, micro trở nên thêm phần quan trọng như là một thiết bị nhập để ghi âm giọng nói. Người sử dụng có thể ghi âm giọng nói bằng một micro và card âm thanh, sau đó tạo các file trên đĩa.
  - ✓ Nhập hình ảnh: Với sự lớn mạnh của truyền thông đa phương tiện và Internet, người sử dụng máy tính đang bổ sung các khả năng nhập hình ảnh vào hệ thống với số lượng lớn, việc nhập này thường là do PC camera thực hiện.
  - ✓ Camera kỹ thuật số

### **3.2. Các thiết bị xuất**

Hiện tại, thiết bị xuất phổ biến và được sử dụng rộng rãi nhất là màn hình máy tính. Đây là một thiết bị rất quen thuộc với người sử dụng máy tính do đó trong phạm vi tài liệu này, chúng ta chỉ liệt kê một số thiết bị xuất mà không thảo luận một cách chi tiết về cách thức sử dụng, các đặc điểm hay nguyên lý hoạt động của từng loại thiết bị này.

Một số thiết bị xuất:

- Màn hình
- Máy in
- Máy chiếu

### 3.3. Bộ nhớ

#### 3.3.1. Bộ nhớ ngắn hạn

Đơn vị nhớ mức thấp nhất của máy tính là các thanh ghi trên chip, nhưng nó có ảnh hưởng rất ít đến người sử dụng ngoại trừ những ảnh hưởng đến tốc độ xử lý chung. Hầu hết mọi thông tin hoạt động hiện thời đều được lưu trữ trên RAM. Các loại Ram khác nhau thì có thời gian truy cập, mức tiêu thụ nguồn, và các đặc tính khác nhau. Thông tin trong Ram có thể được truy cập với tốc độ khoảng 10Mbytes/một giây. Tuy nhiên, khi bị ngắt điện thì mọi thứ trong Ram đều bị mất đi.

#### 3.3.2. Bộ nhớ dài hạn

Máy tính có các thiết bị ngoại vi có khả năng nhận và xuất dữ liệu - đó là các ổ đĩa máy tính, nơi ghi nhớ các thông tin dữ liệu. Những thiết bị này gọi là *các thiết bị lưu trữ thứ cấp - secondary storage* (thiết bị lưu trữ sơ cấp - primary storage là bộ nhớ máy tính.) Khác với thiết bị lưu trữ sơ cấp khi ngắt điện là mọi thứ trong RAM đều không còn, loại thứ cấp này có thể lưu dữ kiện ngay cả khi không có nguồn nuôi, xét về lý thuyết, dữ liệu trên loại này có thể tồn tại vĩnh viễn và có thể được đọc, ghi, sửa hay xóa lúc này hay lúc khác, Có hai phương pháp lưu dữ kiện tạo nên hai họ khác nhau, là dựa trên từ tính, và dựa trên khả năng ứng dụng quang học.

##### a. Đĩa từ tính

Có hai loại chủ yếu là đĩa mềm và đĩa cứng. Đĩa mềm, có thể hiểu đơn giản là loại đĩa dung lượng thấp, nhỏ gọn tháo lắp dễ dàng, nhiều đĩa dùng chung một ổ đĩa. Hiểu như vậy để có thể phân biệt với đĩa cứng là loại ổ đĩa thường lắp hẳn bên trong máy, ít được tháo rời, phức tạp, và bản thân nó là thiết bị hoàn chỉnh đọc ghi với dung lượng lớn.

- **Đĩa mềm - floppy disk**

Gồm vỏ bảo vệ và một đĩa plastic nhỏ có phủ vật liệu từ (oxit sắt, oxit niken hoặc oxit coban pha với vật liệu không từ tính hay đất hiếm). Dữ liệu thông tin dạng số sẽ được đại diện bởi các hạt từ tính, các hạt này do có tính từ tính, nên bằng một phương pháp nào đó nó được xác định một trong hai hướng rõ rệt - như vậy thể hiện được số 0 hay số 1. Đĩa mềm cũng trải qua nhiều thế hệ, những khác nhau giữa các hệ đĩa mềm chỉ là quanh vấn đề dung lượng nhớ của nó chứ về nguyên tắc hoạt động của đĩa cũng như ổ đĩa không có thay đổi lớn nào. Ngày nay người ta chỉ còn dùng loại đĩa mềm 3.5 inch thường gọi là 1 đĩa 1.44Mb, nhưng cũng có loại 2,88Mb. Loại 5,25 inch gần như không còn được dùng nữa. Đĩa mềm có tính cơ động cao, nhưng bị hạn

chế về dung lượng nhớ, hiện nay các chương trình hầu như không thể chạy trên đĩa mềm như cách đây khoảng 5 năm, cho nên đĩa mềm chủ yếu dùng sao lưu dữ liệu, và lại tốc độ đọc ghi của đĩa mềm rất thấp - bù lại giá đĩa mềm tương đối rẻ, bạn có thể mua một hộp đĩa mềm với giá khoảng 50.000 đồng (10 hay 11 đĩa).

- **Đĩa cứng - Hard disk**

Có thể so sánh tên gọi đĩa cứng và đĩa mềm cũng đủ rút ra kết luận. Đĩa cứng...cứng hơn đĩa mềm! Và giá cả cũng đắt hơn. Bạn có thể hình dung, ổ đĩa cứng gồm vỏ cứng bảo vệ, các bộ phận điều khiển xuất nhập, nguồn, và đĩa từ tính. Bộ khung vỏ bảo vệ thường là hợp kim nhôm đúc ở áp lực cao, cũng có hai cỡ 5.25 inch và 3.5 inch, và thông dụng nhất vẫn là loại 3.5 inch. Dung lượng ổ cứng không phải bị chi phối bởi vỏ ổ cứng mà ở đĩa từ. Đĩa từ của ổ cứng thường làm từ nhôm, thủy tinh hay gốm-phủ một lớp vật liệu từ và lớp bảo vệ ở cả hai mặt. Ổ cứng có thể có nhiều đĩa từ xếp chồng lên nhau trên trục mô-tơ quay. Để hoạt động đọc ghi, ổ cứng còn có các đầu từ, mô-tơ dịch chuyển đầu từ, mạch chính, mạch điện tử điều khiển, và thường có bộ nhớ cache. Đĩa cứng rất đa dạng về dung lượng, có thể từ vài chục Mb đến vài nghìn Mb hay hơn nữa, và phụ thuộc nhiều vào các chuẩn kỹ thuật giao tiếp. Loại ổ cứng thường dùng trong máy vi tính hiện nay khoảng 1 Gb đến 2 Gb, một con số khổng lồ nếu so với cách đây 5 năm khi mà ổ cứng chỉ có thể từ 100Mb đến 200Mb, thậm chí có máy không trang bị ổ cứng nữa.

Các ổ cứng giao tiếp với máy thông qua một dữ liệu cắm vào mạch điều khiển. Những máy tính thế hệ tiền PCI- tức là từ VesaLocalBus trở về trước giao tiếp giữa máy và thiết bị ngoại vi đều thông qua bảng mạch giao tiếp - thường gọi là các IO (input-output), tức là con chuột, ổ cứng, ổ mềm, máy in, joystick, đều nối vào đây. Để hoạt động, các ổ cứng giao tiếp với máy thông qua các chuẩn ESDI, IDE, SCSI. Chuẩn ESDI (Enhanced Small Device Interface) xuất hiện đầu năm 1983 dùng phương pháp mã hóa RLL, tốc độ có thể đạt đến khoảng 24 MB mỗi giây, là dạng giao tiếp cắm nên các điều khiển quan trọng đều do card quản lý. Chuẩn IDE (Intelligent Drive Electronic - Intergrated Drive Electronic) cũng còn gọi là ATA (AT Attachment) dùng loại mạch điện tử ổ đĩa thông minh, là giao tiếp ở mức hệ thống. Chuẩn này nối với máy bằng một cáp nguồn 4 chân và một cáp dữ liệu 40 chân. Loại ổ đĩa này có tốc độ khá cao nên được dùng trong hầu hết các máy vi tính hiện nay, giá thành cũng rẻ hơn so với các loại ổ cứng khác. Nếu tính cụ thể thì khoảng 1.5USD cho 10Mb, rẻ hơn cả đĩa mềm: 1.5USD cho gần 5 Mb. Chuẩn giao tiếp SCSI (Small



Computer System Interface) là một cấu trúc bus độc lập có thể truyền dữ liệu với tốc độ cao, từ 4Mb/giây đến khoảng 10Mb/giây. Được ứng dụng với ổ cứng tạo nên một khả năng lưu trữ cao với tốc độ đọc ghi cao. Để dùng được, cần có một bảng mạch điều hợp SCSI, tuy nhiên một card SCSI này có thể nối tiếp đến 7 thiết bị theo chuẩn này.

- **Đĩa floptical**

Là loại ổ đĩa từ mềm, có hình dáng giống như đĩa 3.5 inch, nhưng dùng phương pháp vị quang học để đọc ghi, nên mật độ dữ liệu trên đĩa cao hơn, dung lượng nhớ lớn hơn. Thiết bị này không ghi dữ liệu bằng quang học, chỉ làm thao tác định vị thôi. Nhưng do giá thành cao nên dù có khả năng lưu đến hơn 20MB, loại này vẫn không phổ dụng.

- **Ổ băng ghi lưu**

Cũng là thiết bị lưu trữ từ tính, nhưng loại này khác với các loại trên ở tính chất truy cập tuần tự của nó, do đó chỉ dùng sao lưu chứ không dùng để làm việc hằng ngày như thiết bị truy cập ngẫu nhiên - đĩa cứng, đĩa mềm. Ổ băng ghi lưu gồm một hộp băng và cuộn băng từ cỡ 0,25 inch. Loại này rất đa dạng về chủng loại và dung lượng, tùy yêu cầu công việc mà bạn lựa chọn.

- **Ổ đĩa tháo lắp ZIP**

Dùng loại đĩa có kích thước cũng khoảng 3.5 inch, dung lượng lên đến 100Mb trên một đĩa giá 20USD. Tốc độ đọc ghi trung bình, kỹ thuật dùng ở đây là định vị quang học để ghi dữ liệu. Nếu dùng với card SCSI, tốc độ không thua gì ổ cứng IDE.



Hình 3.1 : Một loại ổ ZIP

## **b. Đĩa từ quang**

*Đĩa từ quang - Magneto optical drive*, thường gọi tắt là MO, là thiết bị kết hợp giữa từ tính và quang học để lưu dữ liệu. Đĩa từ tính, dùng ánh sáng laser làm tác nhân đọc ghi. Dung lượng của loại 5.25 inch là 1.3Gb, loại 3.5inch là 230 Mb. Công nghệ này phù hợp để lưu trữ, theo các chuyên gia, có thể bảo đảm dữ liệu 50 năm so với 5 năm của ổ cứng, ổ mềm, băng từ.

## **c. Đĩa quang học**

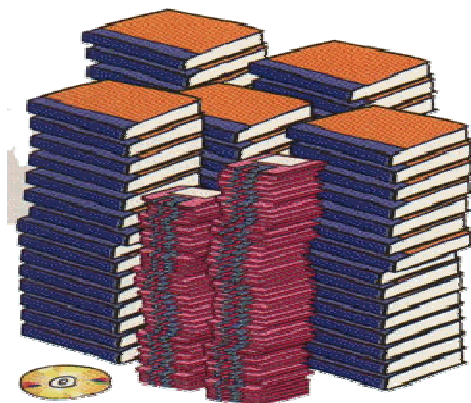
Gọi là đĩa quang học, tức là vấn đề cốt lõi về kỹ thuật - đọc ghi dữ liệu được thực hiện trên nguyên tắc quang học, dùng tia sáng laser. So với hệ thống từ tính, ổ quang có ba điểm khác biệt chính: vì độ chính xác cao của thao tác quang học, nên ổ đĩa quang có thể có dung lượng cao hơn ổ đĩa từ gấp nhiều lần so với ổ đĩa từ. Độ bền dữ liệu ghi bằng phương pháp quang học cao hơn so với phương pháp từ tính nhiều lần, tối thiểu cũng 50 năm. Đĩa quang có thể tháo lắp dễ dàng như một đĩa mềm mà hiệu quả hơn nhiều, do đó ngày càng phổ dụng hơn. Đĩa CD - compact disc là loại này.

Xuất phát từ nhu cầu âm thanh, CD âm thanh ra đời chứa dữ liệu dưới dạng các hốc lõm, khi CD quay tia laser sẽ phát đến đĩa và nhận sự phản xạ khác nhau giữa điểm lõm và điểm không lõm ứng với số 0 và 1 hệ nhị phân. Đĩa CD-ROM ta dùng hiện nay cũng hoạt động theo nguyên tắc đó, vì là loại đĩa CD chứa dữ kiện chỉ đọc được nên có tên là CD-ROM (*Compact Disc Read Only Memory*). Thông thường dữ liệu có thể đưa vào loại đĩa quang giá rẻ 680Mb (khoảng 10USD đĩa trắng - 1USD 68 MB) như chúng ta dùng rộng rãi hiện nay, đồng thời các loại đĩa âm thanh cũng có thể đọc hiểu và hoạt động được bằng ổ đĩa CD của máy tính, nhưng đầu đọc của máy CD âm thanh thì không thể đọc được đĩa CD dữ liệu. Nói là CD-ROM-chỉ đọc, nhưng dĩ nhiên là phải có một lần nào đó ghi dữ liệu lên đĩa rồi mới đọc, thao tác này theo nguyên tắc khắc trên đĩa các điểm lõm hay không lõm đại diện cho số 0,1 bằng một nguồn phát tia laser công suất lớn. Người ta tạo một đĩa gốc trước trên nguyên tắc này bằng đầu CD có thể ghi trên một đĩa CD mới, sau đó âm bản của đĩa gốc được tạo ra bằng quá trình mạ điện hoặc photopolymer. Tiến trình nhân bản thực hiện bằng cách phun polycarbonate-trong suốt, nhẹ, bền, ổn định, không nhiễm bẩn - nên đĩa CD giữ được thông tin gần như vĩnh viễn.

Như vậy, bạn có thể hiểu về bản chất các đĩa CD được chép lại bán ở một số dịch vụ tin học thực chất là một dạng đĩa gốc, do đó khi sử dụng phải tuyệt đối cẩn

thậm vì nó không hề có một lớp bảo vệ polycarbonate như các đĩa CD được phát hành chính quy hay các đĩa CD nhạc.

Khi mà các đĩa CD-ROM đã gần như trở nên một chuẩn không thể thiếu trong hầu hết các máy tính multimedia thì lại xuất hiện một thành viên mới trong họ đĩa quang học mà được dự đoán sẽ là thiết bị lưu trữ chủ đạo thế kỷ 21 - DVD.



Hình 3.2: Một đĩa CD lưu được một lượng thông tin tương đương 450 đĩa mềm và khoảng 500 cuốn sách

*DVD - Digital Vidéo Disc* tức là đĩa video kỹ thuật số hay Digital Versatile Disc - đĩa đa năng kỹ thuật số là một công nghệ chỉ mới ra đời gần đây. Cho đến thời điểm hiện nay, DVD vẫn chỉ còn trong dự án với giá thành khá cao, đã có DVD bán ra thị trường nhưng chỉ ở dạng hàng mẫu không phổ biến lắm. Vậy đâu là điểm mạnh mà DVD được đoán là sẽ nhanh chóng thay thế CD trong tương lai? Cũng như CD dần dần thay thế đĩa mềm bởi dung lượng hàng trăm Mb của nó, DVD thay thế CD-ROM bởi DVD có thể lưu ít nhất 3.8 Gb và có thể đạt đến 17 Gb. DVD có kích thước giống như CD (120mm đường kính và dày 1,2mm) cũng làm bằng nguyên liệu như CD. Như đã nói ở trên, đây là một bước tiến về công nghệ dữ liệu trên DVD sẽ được ghi vào đĩa với mật độ cao hơn, sít hơn nhiều so với CD, lượng thấu kính trong đầu đọc nhiều hơn để tăng độ chính xác - và đầu đọc sẽ dùng laser có sóng ngắn hơn, có thể là tia laser đỏ - laser hồng ngoại. Quan trọng nhất là kỹ thuật DVD cho phép loại đĩa có hai lớp trên một mặt, nên với mỗi lớp khoảng hơn 4Gb thì loại đĩa 2 lớp hai mặt hoàn toàn có thể chứa đến 17Gb dữ liệu - hãy hình dung bằng toàn bộ dữ liệu của cả một thư viện 10 ngàn cuốn sách

## CHƯƠNG IV: TƯƠNG TÁC

### 4.1. Các mô hình tương tác

Chúng ta thấy rằng, tương tác bao gồm ít nhất 2 thành phần tham dự đó là: người sử dụng và hệ thống. Cả hai thành phần này đều phức tạp, và cách để liên lạc và hiểu về phạm vi và nhiệm vụ giữa chúng là rất khó khăn. Do đó, giao diện phải làm nhiệm vụ truyền tải một cách có hiệu quả giữa 2 thành phần đó để đảm bảo cho tương tác thành công. Sự truyền tải này có thể thất bại ở một số điểm vì rất nhiều lí do. Chính vì vậy, việc sử dụng các mô hình tương tác có thể giúp chúng ta hiểu chính xác những gì đang diễn ra trong tương tác và nhận dạng được nguồn gốc của những khó khăn. Chúng cũng cung cấp cho chúng ta một nền tảng để có thể so sánh các kiểu tương tác khác nhau và xem xét các vấn đề liên quan đến tương tác.

Trong phần này, chúng ta sẽ bắt đầu bằng việc xem xét một mô hình tương tác có ảnh hưởng lớn nhất, mô hình chu trình thực hiện-đánh giá của Norman. Sau đó chúng ta sẽ xét đến một số mô hình mở rộng khác dựa trên ý tưởng của Norman. Tất cả các mô hình đó đều mô tả tương tác dưới dạng các mục đích và hành động của người sử dụng.

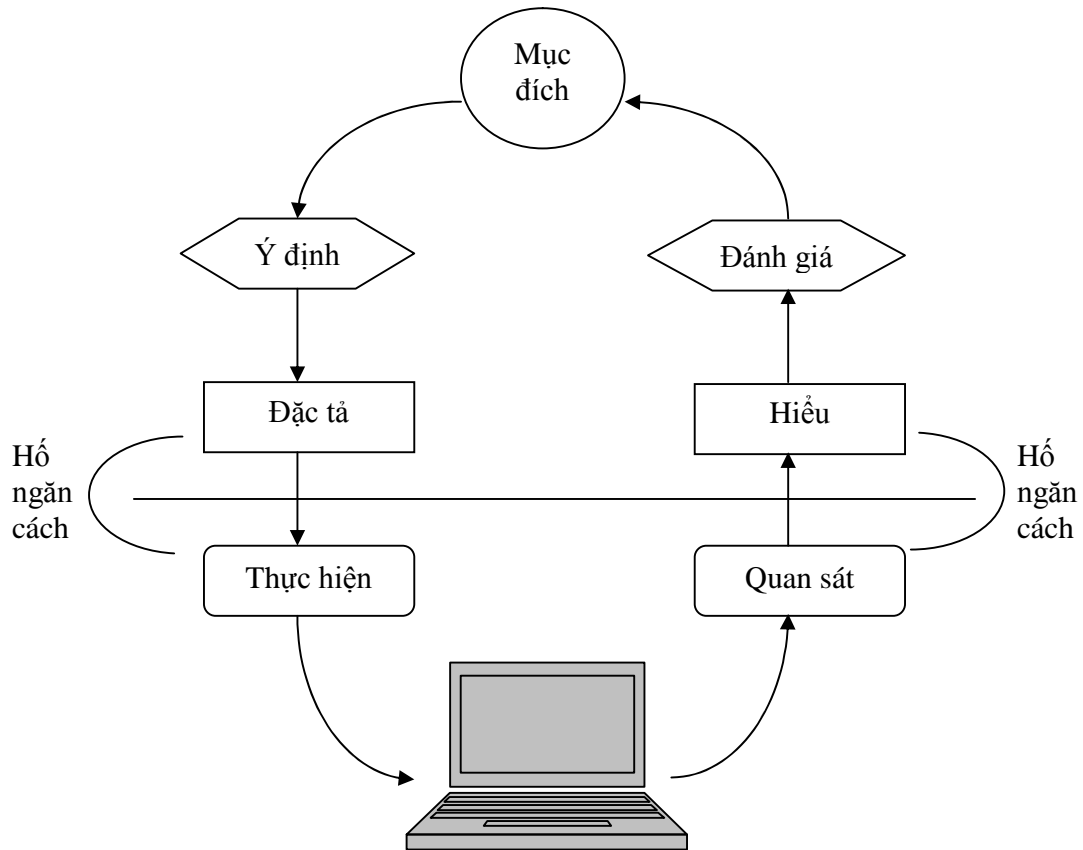
#### 4.1.1. Chu trình thực hiện-đánh giá

Mô hình tương tác của Norman là một mô hình có ảnh hưởng nhiều nhất trong lĩnh vực tương tác người-máy. Sự ảnh hưởng này có thể là do tính gần gũi của mô hình với những hiểu biết cảm tính của chúng ta về quá trình tương tác giữa người sử dụng và máy tính. Người sử dụng xây dựng một kế hoạch hành động, sau đó thực hiện kế hoạch đó thông qua giao diện máy tính. Khi kế hoạch hoặc một phần của kế hoạch được thực hiện, người sử dụng quan sát giao diện máy tính để đánh giá kết quả thực hiện được của kế hoạch và xác định các hành động tiếp theo.

Chu trình tương tác có thể được phân chia thành 2 giai đoạn chính: giai đoạn thực hiện và giai đoạn đánh giá. Các giai đoạn chính đó lại có thể được phân chia tiếp thành các giai đoạn con. Kết quả là, mô hình Norman bao gồm 7 giai đoạn sau:

- Thiết lập mục đích
- Sắp xếp ý định
- Đặc tả dãy hành động
- Thực hiện hành động

- Quan sát trạng thái hệ thống
- Hiểu trạng thái hệ thống
- Đánh giá trạng thái hệ thống về các mục đích và các ý định



Hình 3.1: Mô hình tương tác của Norman

Các giai đoạn thực hiện (nằm ở phía bên trái của hình vẽ) làm nhiệm vụ truyền tải mục đích ban đầu thành một ý định để thực hiện một việc gì đó. Ý định này sau đó được chuyển thành một dãy các hành động (tập các hành động có thể được hoàn thành để thỏa mãn ý định). Cuối cùng, hành động được thực hiện.

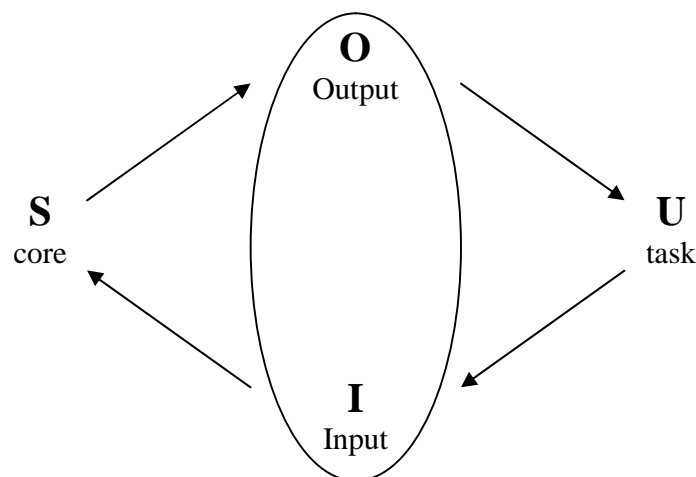
Các giai đoạn đánh giá (nằm phía bên phải của hình vẽ) bắt đầu bằng việc người sử dụng quan sát trạng thái mới của hệ thống, sau khi dãy các hành động đã được thực hiện, và hiểu nó theo kì vọng của người dùng. Nếu trạng thái hệ thống phản ánh đúng mục đích của người dùng thì máy tính đã thực hiện được những gì mà người dùng muốn và tương tác thành công; Ngược lại, người dùng sẽ xây dựng một mục đích mới và lặp lại chu trình trên.

**Các “hố ngăn cách”:** Mô hình tương tác của Norman cho phép nhận dạng một số “hố ngăn cách” chia tách những ý định và hiểu biết của con người với các hành động và các trạng thái vật lý của hệ thống. Mỗi hố ngăn cách chỉ ra một vấn đề tiềm năng mà người sử dụng sẽ gặp phải. Có 2 loại hố ngăn cách đó là:

- Hố ngăn cách trong thực hiện: Là sự khác nhau giữa các ý định của người sử dụng và các hành động mà hệ thống có thể chấp nhận được
- Hố ngăn cách trong đánh giá: Phản ánh tổng số cố gắng mà người sử dụng phải sử dụng để hiểu về trạng thái vật lý của hệ thống và để xác định xem các ý định và các kì vọng của họ đã được đáp ứng như thế nào?

#### 4.1.2. Khung tương tác

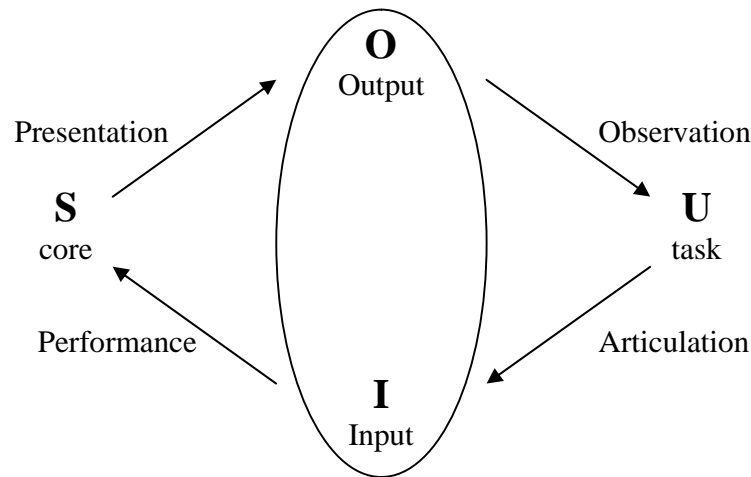
Khung tương tác cố gắng mô tả tương tác một cách thực tế hơn bằng cách đưa hệ thống vào một cách rõ ràng hơn so với mô hình Norman, và chia hệ thống tương tác thành 4 phần chính, như trong hình 3.2.



Hình 3.2: Khung tương tác chung

Các node biểu diễn 4 thành phần chính của một hệ thống tương tác đó là: Hệ thống (S), người sử dụng (U), đầu vào (I), và đầu ra (O). Mỗi thành phần có ngôn ngữ riêng của nó. Ngoài ngôn ngữ nhiệm vụ của người sử dụng và ngôn ngữ nhân của hệ thống, còn có các ngôn ngữ khác dành riêng cho các thành phần đầu vào (I) và đầu ra (O) để biểu diễn các thành phần riêng biệt, mặc dù cũng có thể có các thành phần bị nạp chồng lên nhau. Đầu vào và đầu ra kết hợp tạo thành giao diện.

Khi giao diện được đặt giữa người sử dụng và hệ thống, có 4 bước trong chu trình tương tác, mỗi bước tương ứng với một chuyển đổi/dịch từ một thành phần đến một thành phần khác, được minh họa bởi các cung có nhãn trong hình 3.3.

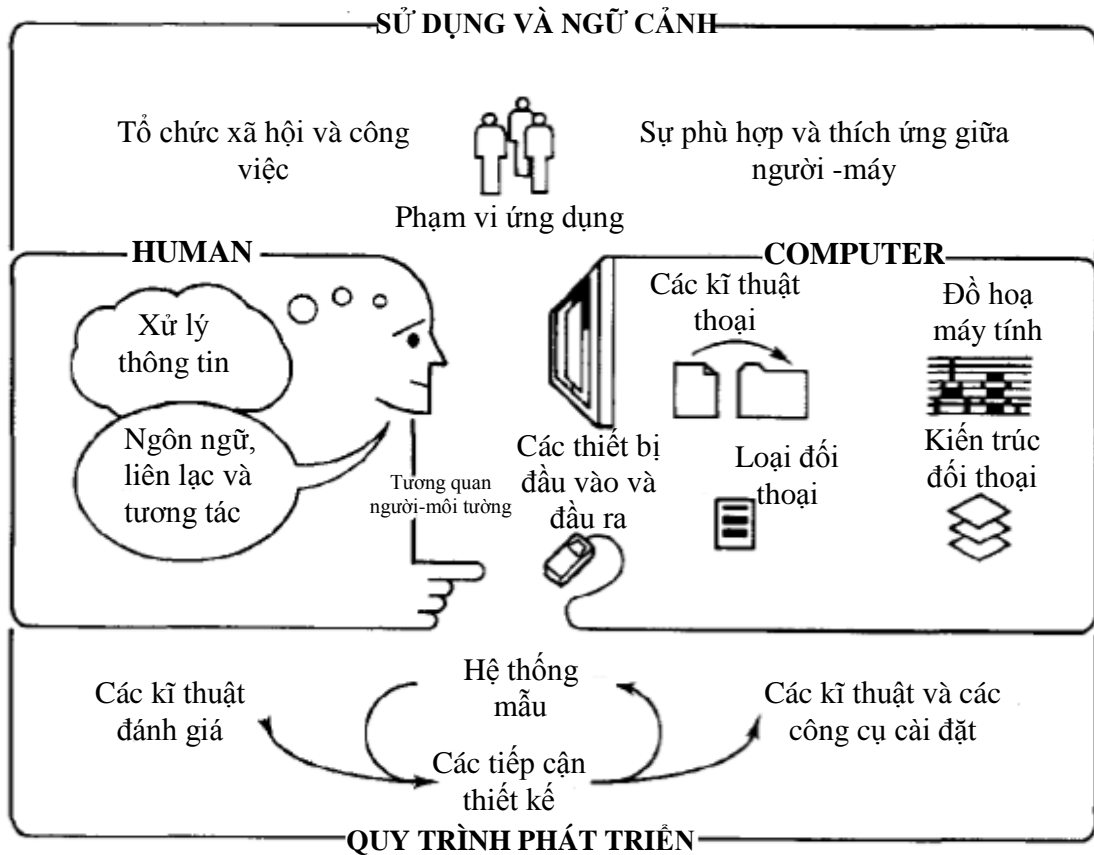


Hình 3.3: Các chuyển đổi giữa các thành phần

Người sử dụng bắt đầu chu trình tương tác bằng việc thiết lập một mục đích và một nhiệm vụ để đạt được mục đích đó. Cách duy nhất để người sử dụng có thể thao tác với máy là thông qua đầu vào, và do đó nhiệm vụ phải được gộp vào cùng với ngôn ngữ đầu vào. Ngôn ngữ đầu ra được dịch thành ngôn ngữ nhân để có thể được thực hiện bởi hệ thống. Sau đó hệ thống sẽ tự chuyển đổi theo các thao tác đã được dịch từ đầu vào. Giai đoạn thực hiện của chu trình đã hoàn tất và bây giờ giai đoạn đánh giá bắt đầu. Hệ thống đang ở một trạng thái mới, và người sử dụng sẽ phải liên lạc với trạng thái mới đó. Các giá trị hiện tại về các thuộc tính của hệ thống sẽ được biểu diễn thành các khái niệm hoặc các đặc điểm của đầu ra. Sau đó, người sử dụng sẽ quan sát đầu ra và đánh giá các kết quả của tương tác so với mục đích ban đầu, do đó kết thúc giai đoạn đánh giá và chu trình tương tác. Có 4 sự chuyển đổi chính trong tương tác: kết hợp, thực hiện, trình diễn, và quan sát.

#### 4.2. Các khung và HCI

Ta thấy rằng, khung không chỉ cung cấp một cách thức để nghiên cứu về các chi tiết của một tương tác cụ thể mà còn là nền tảng cơ bản cho việc nghiên cứu các vấn đề khác liên quan đến tương tác. Nhóm nghiên cứu về HCI của ACM SIGCHI đã đưa ra một khung mô tả tương tác người – máy, trong đó bao gồm nhiều vấn đề liên quan đến HCI, như trong hình vẽ 3.4.



Hình 3.4: Khung tương tác người – máy (do nhóm HCI của ACM SIGCHI đưa ra)

Các vấn đề liên quan đến HCI được đề cập đến trong mô hình sẽ được trình bày một cách chi tiết trong các phần tiếp theo.

### 4.3. Tương quan người – môi trường

Công thái học (hoặc các nhân tố con người) là một bộ phận của tâm lý học ứng dụng nghiên cứu các đặc điểm tự nhiên của tương tác: cách thiết kế các điều khiển, môi trường vật lý mà tương tác xảy ra, và chất lượng hiển thị và chất lượng vật lý của màn hình. Mục đích của công thái học là cải tiến thiết kế thái học có thể áp dụng cho máy tính, trên cả phương diện phần cứng cũng như phương diện phần mềm. Đây là một lĩnh vực rất lớn, do đó trong phạm vi của tài liệu này, chúng ta chỉ tập trung xem xét một số vấn đề chính liên quan đến tương tác người – máy, bao gồm như sau:



### 4.3.1. Sắp xếp các điều khiển và hiển thị

Trong chương 1 chúng ta đã thảo luận các vấn đề liên quan đến việc tiếp nhận và nhận thức. Các vấn đề đó có ảnh hưởng đến cách chúng ta biểu diễn thông tin trên màn hình máy tính và cung cấp các cơ chế điều khiển cho người sử dụng. Tuy nhiên, ngoài các khía cạnh nhận thức của thiết kế, các khía cạnh vật lý cũng rất quan trọng. Tập các điều khiển và các phần trên màn hình hiển thị nên được nhóm lại một cách logic để người sử dụng có thể truy cập đến chúng nhanh hơn. Đối với người dùng các ứng dụng trên máy tính PC thì việc nhóm trên đường như không quan trọng, tuy nhiên nó lại là vấn đề cực kỳ quan trọng khi người dùng sử dụng các ứng dụng đòi hỏi sự an toàn, chính xác cao như là điều khiển sản xuất, điều khiển hàng không và không lưu. Trong những ngữ cảnh đó, người sử dụng phải làm việc dưới áp lực cao, và phải đối mặt với rất nhiều loại điều khiển và hiển thị khác nhau. Do đó, việc thiết kế hiển thị vật lý phù hợp trở nên vô cùng quan trọng. Thực tế, đối với các ứng dụng trên PC, việc đặt sai/không hợp lý các điều khiển và hiển thị cũng có thể làm cho người sử dụng không thoải mái và chán nản. Với mỗi lĩnh vực và ứng dụng có thể có các cách tổ chức khác nhau. Một số cách tổ chức bao gồm:

- Tổ chức theo chức năng: Các điều khiển và hiển thị được tổ chức sao cho các điều khiển hoặc các hiển thị có chức năng tương tự nhau thì được đặt cạnh nhau
- Tổ chức theo kiểu tuần tự: Các điều khiển và hiển thị được tổ chức sao cho có thể phản ánh thứ tự sử dụng của chúng trong một tương tác điển hình (cách tổ chức này đặc biệt thích hợp trong các lĩnh vực đòi hỏi các nhiệm vụ phải được thực hiện một cách tuần tự, như là điều khiển hàng không)
- Tổ chức theo tần số xuất hiện: Các điều khiển và hiển thị được tổ chức theo tần số sử dụng của chúng, nghĩa là điều khiển nào được sử dụng nhiều nhất sẽ là điều khiển có thể truy cập dễ dàng nhất.

Ngoài việc tổ chức các điều khiển và hiển thị phù hợp, giao diện của toàn bộ hệ thống cũng phải được sắp xếp một cách phù hợp với vị trí của người sử dụng. Ví dụ, người sử dụng phải có khả năng truy cập đến tất cả các điều khiển và có thể xem được tất cả các hiển thị mà không cần bất cứ một sự di chuyển vị trí làm việc nào. Những phần hiển thị quan trọng nên thiết kế sao cho dễ nhìn thấy nhất.

### **4.3.2. Môi trường vật lý của tương tác**

Cùng với việc xác định các vấn đề về hiển thị và sắp xếp các điều khiển của giao diện máy, công thái học thường quan tâm đến thiết kế của bản thân môi trường làm việc. Hệ thống sẽ được sử dụng ở đâu? Ai sẽ sử dụng hệ thống? Người sử dụng sẽ ngồi ở vị trí cố định hay di chuyển? Câu hỏi này phụ thuộc phần lớn vào lĩnh vực và quan trọng hơn là phụ thuộc vào điều khiển cụ thể và các thiết lập hoạt động hơn là việc sử dụng máy tính nói chung. Tuy nhiên, môi trường vật lý có thể có ảnh hưởng việc tiếp nhận, sức khỏe và an toàn của người sử dụng. Do đó, vấn đề này cũng cần được xem xét đến trong quá trình thiết kế.

Đầu tiên chúng ta cần xem xét đến khối lượng người sử dụng. Trong bất kỳ một hệ thống nào, chúng ta nên thiết kế để làm sao cho khi số người sử là ít nhất thì mọi người đó đều có khả năng truy cập đến tất cả các điều khiển và khi số lượng người sử dụng là lớn nhất thì những người đó cũng không bị gò bó trong môi trường chật hẹp.

Đặc biệt, thiết kế phải làm sao cho tất cả mọi người sử dụng đều có thể nhìn thấy toàn bộ phần hiển thị một cách thoải mái nhất. Trước khi sử dụng, họ cần được ngồi ở một vị trí thuận tiện và thoải mái. Nếu phải đứng thì cũng cần phải có không gian đủ lớn để người sử dụng có thể di chuyển và truy cập đến tất cả các điều khiển.

### **4.3.3. Các vấn đề về sức khỏe**

Vấn đề sức khỏe và sự an toàn của người sử dụng cũng cần phải được quan tâm đến trong quá trình thiết kế giao diện. Một số nhân tố trong môi trường vật lý không chỉ ảnh hưởng đến sức khỏe và sự an toàn của người sử dụng mà còn trực tiếp ảnh hưởng đến chất lượng tương tác và hiệu quả làm việc của người sử dụng. Các nhân tố đó bao gồm:

- Vị trí vật lý: Như đã thảo luận trong phần trước, thiết kế nên đảm bảo cho mọi người sử dụng đều có khả năng truy cập đến tất cả các điều khiển và đều xem được toàn bộ mọi hiển thị.
- Nhiệt độ: Mặc dù hầu hết mọi người sử dụng đều có khả năng thích ứng với những thay đổi nhỏ về nhiệt độ, tuy nhiên nếu như nhiệt độ là quá nóng hoặc quá lạnh thì có thể sẽ ảnh hưởng đến hiệu quả làm việc và ảnh hưởng đến sức khỏe. Những nghiên cứu thực nghiệm cho thấy rằng, khi nhiệt độ quá cao hoặc quá thấp, hiệu quả làm việc sẽ giảm và người sử dụng sẽ không thể tập trung vào công việc được.

- Ánh sáng: Mức ánh sáng sẽ phụ thuộc vào môi trường làm việc. Tuy nhiên, nên đảm bảo mức ánh sáng phù hợp để người sử dụng có thể nhìn thấy màn hình máy tính mà không cần phải điều tiết mắt để nhìn. Nguồn ánh sáng cũng nên đặt ở vị trí thích để không bị chói.
- Tiếng ồn: Nếu tiếng ồn vượt mức cho phép có thể ảnh hưởng xấu đến sức khỏe, làm cho người sử dụng bị đau tai và trong trường hợp tồi nhất là sẽ bị mất khả năng nghe. Mức tiếng ồn nên duy trì ở ngưỡng vừa phải. Điều này không có nghĩa là bắt buộc môi trường làm việc phải không có tiếng ồn. Bởi vì, tiếng ồn có thể là một kích thích/cảnh báo đối với người sử dụng và có thể cung cấp sự xác nhận về hoạt động của hệ thống.
- Thời gian: Lượng thời gian mà người sử dụng dùng để sử dụng hệ thống cũng nên được kiểm soát. Vì nếu dùng quá nhiều thì có thể ảnh hưởng xấu đến sức khỏe của người sử dụng.

#### 4.3.4. Sử dụng màu sắc

Như chúng ta đã biết, hệ thống thị giác có một số giới hạn liên quan đến màu sắc, như là số lượng màu mà mắt có thể phân biệt được hoặc sự khó khăn khi trong quá trình thu nhận màu xanh da trời. Do đó, các màu được sử dụng trên màn hình nên rõ ràng. Không nên sử dụng màu xanh da trời khi hiển thị các thông tin quan trọng. Không nên sử dụng màu như một chỉ dẫn duy nhất mà nên thêm vào các thông tin phụ để chỉ dẫn. Ngoài ra, việc sử dụng các màu cũng nên theo quy ước chung và theo mong muốn của người sử dụng. Màu đỏ, xanh lá cây và màu vàng là các màu thường được sử dụng với các chỉ dẫn: stop, go và standby. Do đó, màu đỏ có thể được sử dụng trong các cảnh báo và tình trạng khẩn cấp; xanh lá cây, hoạt động bình thường; và màu vàng, chức năng hỗ trợ hoặc dự phòng.

Một số kết hợp giữa màu nền trước và màu nền sau nên dùng trong khi thiết kế màu nền của giao diện:

| Màu nền sau  | Các màu nền trước nên dùng | Các màu nền trước nên tránh |
|--------------|----------------------------|-----------------------------|
| Trắng        | Đen, xanh da trời đậm, đỏ  | Xám nhạt, màu lục lam       |
| Xanh da trời | trắng, vàng, màu lục lam   | Xanh lá cây                 |
| Hồng         | trắng, vàng                | Xanh lá cây, lục lam        |

|                  |                                 |                            |
|------------------|---------------------------------|----------------------------|
| đỏ               | vàng, trắng                     | lục lam, xanh lá cây       |
| Vàng             | đỏ, xanh da trời, đen           | lục lam                    |
| Xanh lá cây      | Đen, đỏ, xanh da trời           | lục lam, hồng, vàng        |
| Lục lam          | Xanh da trời, đen, đỏ           | Xanh lá cây, vàng, trắng   |
| Xám nhạt         | Đen, xanh da trời đậm, hồng đậm | Xanh lá cây, lục lam, vàng |
| Xám              | Vàng, trắng, xanh da trời       | Xám đậm, lục đậm           |
| Xám đậm          | lục lam, vàng, xanh lá cây      | đỏ, xám                    |
| Đen              | trắng, lục, xanh lá cây, vàng   | lục đậm                    |
| Xanh da trời đậm | Vàng, trắng, hồng, xanh lá cây  | Xanh lá cây đậm            |
| Hồng đậm         | Xanh lá cây, vàng, trắng        | lục đậm                    |

Bảng 3.1 : Một số kết hợp màu nên dùng và nên tránh

#### 4.4. Một số phong cách giao diện

Tương tác được xem như là một cuộc đối thoại giữa máy tính và người sử dụng. Việc lựa chọn các phong cách giao diện có một ảnh hưởng sâu sắc đến bản chất của cuộc đối thoại này. Trong phần này chúng ta sẽ giới thiệu một số phong cách giao diện phổ biến nhất và đồng thời cũng chỉ rõ các ảnh hưởng khác nhau của mỗi phong cách trong quá trình tương tác. Các phong cách giao diện phổ biến nhất bao gồm:

- Giao diện dòng lệnh
- Menus
- Ngôn ngữ tự nhiên
- Đối thoại truy vấn và đối thoại kiểu hỏi/trả lời
- Form-fill và bảng tính
- WIMP
- Point và click

##### 4.4.1. Giao diện lệnh

Giao diện dòng lệnh là kiểu giao diện đối thoại tương tác đầu tiên được sử dụng trong các hệ thống máy tính, và hiện nay nó vẫn đang được sử dụng rộng rãi. Giao

diện cung cấp một phương tiện để biểu diễn trực tiếp các chỉ thị đến máy tính, bằng cách sử dụng các phím chức năng, các kí tự đơn, các dòng lệnh rút gọn hoặc đầy đủ. Trong một số hệ thống, giao diện dòng lệnh là cách duy nhất để trao đổi với hệ thống.

Ưu điểm và nhược điểm của kiểu giao diện dòng lệnh là:

Ưu điểm:

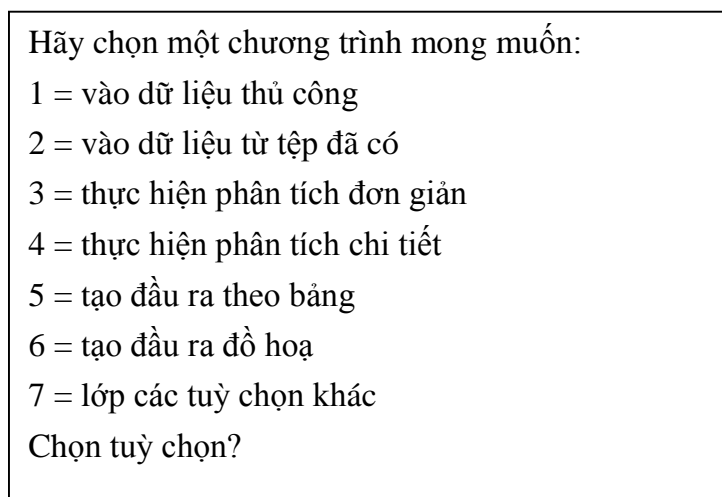
- Cho phép truy cập trực tiếp đến các chức năng của hệ thống
- Linh hoạt: lệnh thường có nhiều lựa chọn và các tham số có thể thay đổi hành vi của nó và có thể được áp dụng với nhiều đối tượng cùng một lúc.

Nhược điểm:

- Khó học
- Khó sử dụng
- Hay lỗi

#### 4.4.2. Menu

Giao diện kiểu menu là một danh sách các tùy chọn được nêu ra cho người sử dụng và quyết định thích hợp được chọn thông qua một mã gõ nào đó cho trước trên màn hình (ví dụ như hình 3.1).



Hình 3.5: Giao diện menu

Ưu và nhược điểm: Menu cung cấp cho người sử dụng một ngữ cảnh tổng thể và ít sinh lỗi hơn định dạng dòng lệnh, nhưng cũng một mỗi khi dùng. Ví dụ, tùy chọn 7 trong ví dụ trên kéo theo việc có thể cần đến các tùy chọn phụ. Người sử dụng không

thể trực tiếp đi tới tùy chọn khác mà phải làm việc qua từng mức menu cho đến khi đạt tới tùy chọn mong muốn. Điều này có thể rất chán, và không hiệu quả.

#### **4.4.3. Ngôn ngữ tự nhiên**

Có thể thoát nhìn thì cách thức để liên lạc với máy tính hấp dẫn nhất là sử dụng ngôn ngữ tự nhiên. Người sử dụng, không nhớ được các câu lệnh hoặc quên mất thứ tự của menu, sẽ mong rằng máy tính có thể hiểu các chỉ thị được biểu diễn trong ngôn ngữ nói hàng ngày. Việc hiểu ngôn ngữ tự nhiên, bao gồm đầu vào thoại và đầu vào chữ viết, đang thu hút sự quan tâm nghiên cứu của các nhà khoa học. Tuy nhiên, do tính nhập nhằng khó hiểu của ngôn ngữ tự nhiên làm cho máy khó có khả năng hiểu được. Ngôn ngữ tự nhiên khó hiểu ở một số điểm như là: ngữ cảnh, cấu trúc hoặc câu có thể không rõ ràng, ý nghĩa của các từ được sử dụng được hiểu theo ngữ cảnh, cách sử dụng thành ngữ, v.v...

Vấn đề sử dụng ngôn ngữ tự nhiên trong tương tác đang được quan tâm nghiên cứu hiện nay.

#### **4.4.4. Đối thoại truy vấn và đối thoại kiểu hỏi / trả lời**

Đối thoại hỏi/trả lời là một cơ chế đơn giản để cung cấp đầu vào cho một ứng dụng trong một lĩnh vực cụ thể. Người sử dụng phải trả lời một loạt các câu hỏi (chủ yếu là các câu hỏi yes/no, các câu hỏi đa lựa chọn hoặc các mã), và sau mỗi bước trả lời chính xác, thì quá trình tương tác sẽ từng bước được thực hiện.

Giao diện đối thoại kiểu hỏi/trả lời tương đối dễ học và dễ dùng, tuy nhiên có chức năng và khả năng hạn chế. Do đó, kiểu giao diện này thích hợp một số lĩnh vực như là các hệ thống thông tin và cho người dùng ít kinh nghiệm.

Mặt khác, ngôn ngữ truy vấn được sử dụng để tạo ra các truy vấn để khôi phục thông tin từ một cơ sở dữ liệu. Chúng sử dụng các câu theo kiểu ngôn ngữ tự nhiên, nhưng thực tế là yêu cầu có cú pháp đặc biệt, cũng như các kiến thức về cấu trúc cơ sở dữ liệu. Các truy vấn thường yêu cầu người sử dụng đặc tả một thuộc tính hoặc các thuộc tính để phục vụ cho việc tìm kiếm trong cơ sở dữ liệu. Với các thuộc tính đơn, thì điều này tương đối là dễ thực hiện, nhưng nó sẽ phức tạp hơn nếu số lượng thuộc tính là rất nhiều. Do đó, việc sử dụng các ngôn ngữ truy vấn đòi hỏi phải có kinh nghiệm.

#### 4.4.5. Form –fill và bảng tính

Giao diện form-fill được sử dụng chủ yếu để nhập dữ liệu. Nó cũng rất hữu ích cho các ứng dụng khôi phục dữ liệu. Giao diện là một form cung cấp các mục thông tin, và người sử dụng điền các giá trị thích hợp vào các mục đó. Kiểu giao diện này dễ học và dễ dùng, đặc biệt thích hợp cho người mới dùng.

**Đại lý bán vé du lịch**

Xin vui lòng nhập các thông tin chi tiết về chuyến đi:

Khởi hành từ:

Điểm đến:

Quá cảnh:

Số ghế:

Hình 3.6: Một giao diện kiểu form-fill

Bảng tính là một biến thể phức tạp hơn của form-fill. Bảng tính bao gồm một lưới các ô, mỗi ô chứa một giá trị nhất định hoặc một công thức. Người sử dụng có thể nhập và thay đổi các giá trị và các công thức theo thứ tự bất kỳ và hệ thống sẽ duy trì sự nhất quán giữa các giá trị được hiển thị và đảm bảo cho tất cả các công thức sẽ được thực hiện đúng. Do đó, người sử dụng có thể thao tác với các giá trị để xem hiệu ứng xảy ra khi thay đổi các giá trị thông số khác nhau.

#### 4.4.6. WIMP

Môi trường tương tác phổ biến nhất hiện nay là môi trường tương tác kiểu WIMP, thường được gọi là các hệ thống cửa sổ. Các thành phần của giao diện WIMP bao gồm: cửa sổ, biểu tượng, con trỏ, menu, nút, thanh công cụ, bảng, hộp thoại,..

Một số ưu điểm quan trọng của kiểu giao diện này là:

- Có thể hiển thị đồng thời nhiều kiểu thông tin khác nhau, cho phép người sử dụng chuyển hoàn cảnh (như viết chương trình gốc trong cửa sổ này, xem kết quả dựa trên cửa sổ khác) mà không mất mối nối trực quan với công việc khác. Cửa sổ cho phép người sử dụng thực hiện nhiều nhiệm vụ trao đổi và nhận biết mà không chán.

- Nhiều nhiệm vụ tương tác khác nhau có sẵn qua sơ đồ đơn kéo xuống. Những sơ đồ đơn kéo xuống cho phép người sử dụng thực hiện các nhiệm vụ kiểm soát và đối thoại một cách dễ dàng.
- Việc dùng biểu tượng đồ họa, nút, kỹ thuật cuộn.. làm giảm khối lượng gõ. Điều này có thể làm tăng tính hiệu quả tương tác cho những người không phải là chuyên viên gõ máy và có thể làm cho máy tính thâm nhập được với những người sợ bàn phím.

#### **4.4.7. Point và click**

Trong hầu hết các hệ thống multimedia và các trình duyệt web, kiểu giao diện thường dùng là point-click. Người sử dụng muốn truy cập thông tin thì chỉ cần trỏ đến đó và nhấp chuột. Ví dụ, bạn có thể trỏ đến một thành phố hoặc một bản đồ và nhấp chuột, thì các thông tin du lịch của thành phố sẽ hiện ra.

Hiện nay, giao diện kiểu point –click đang rất được ưa chuộng trong các trang WWW.

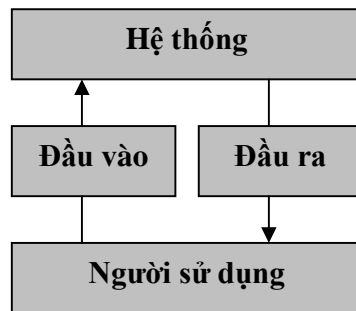


## PHẦN II: THIẾT KẾ GIAO DIỆN NGƯỜI DÙNG

### CHƯƠNG V: GIỚI THIỆU CHUNG VỀ THIẾT KẾ GIAO DIỆN NGƯỜI DÙNG

#### 5.1. Khái niệm giao diện người dùng

Trong các chương trước chúng ta đã biết thế nào là tương tác người máy, vậy giao diện người dùng là gì và nó đóng vai trò như thế nào trong tương tác người máy? Tương tác người máy, có thể định nghĩa một cách đơn giản, là tương tác xảy ra giữa con người và hệ thống máy tính. Giao diện người dùng là bộ mặt, hay còn gọi là thành phần trung gian để thực hiện giao tiếp, giữa con người với máy tính. Nó là nơi người sử dụng nhập thông tin vào hệ thống máy tính (đầu vào) và nhận thông tin phản hồi từ máy tính (đầu ra). Giao diện thì có rất nhiều loại khác nhau (như là bàn phím điện thoại, màn hình máy tính, màn hình của các bộ điều khiển VCR) nhưng về mặt cấu trúc thì đều giống nhau. Chúng đều bao gồm người dùng, hệ thống, đầu vào và đầu ra.



**Ví dụ:** Giả sử chúng ta dùng một chương trình đồ họa để vẽ một hình vuông. Mô hình thông tin đầu vào và đầu ra được cho trong bảng sau:

| Nhu cầu người dùng                            | Đầu vào người dùng   | Đầu ra hệ thống  |
|---|--|--|
| Người dùng muốn dùng công cụ hình vuông để vẽ | Người dùng nhấp chuột vào biểu tượng hình vuông trên thanh công cụ | Hệ thống thay đổi hình dạng con trỏ thành hình dạng ngậm định (hình vuông) để sẵn sàng vẽ. |
| Người dùng thực hiện vẽ hình vuông            | Người dùng kích và kéo con trỏ chuột để vẽ                         | Hệ thống hiển thị hình dạng thu được thực tế trong quá trình di chuyển chuột               |

Ta thấy rằng mặc dù thông tin đầu vào, đầu ra cũng như cách thức xử lý thông tin là khác nhau nhưng về cơ bản mô hình đầu vào và đầu ra là không thay đổi. Ví dụ cũng cho ta thấy sự khác nhau rất lớn về thời gian phản hồi. Khi người dùng nhập thông tin (ví dụ di chuyển chuột) và nhận thông tin đầu ra (hình vẽ), thì thời gian phản hồi là rất ngắn (chỉ mấy miligiây). Trong một số trường hợp khác, thời gian phản hồi là rất dài (ví dụ thời gian cần để lưu trữ hay sao chép một file).

## **5.2. Tại sao cần thiết kế giao diện**

Trong nhiều năm người sử dụng phàn nàn rằng các hệ thống máy tính khó sử dụng, ngôn ngữ khó hiểu và không nhạy bén. Trong khi đó người thiết kế lại là những người hầu như không gặp người sử dụng bao giờ và ngay cả khi gặp và nhận được những lời phàn nàn từ phía người sử dụng họ cũng không để tâm tới. Tình hình này đã được thay đổi từ khi các máy tính các nhân ra đời. Những người sử dụng trực tiếp điều khiển hệ thống, nhiều người có thể lập trình các ứng dụng riêng của mình và lúc này máy tính không chỉ được sử dụng ở văn phòng mà còn xâm nhập vào các gia đình. Vì vậy việc thiết kế một giao diện tốt trở nên quan trọng do :

- Thị trường (người sử dụng) lựa chọn phần mềm đóng gói sẵn: Người ta mua một sản phẩm là do họ thích hình dáng của nó và cái họ xem xét chính là giao diện.
- Người sử dụng phản đối các giao diện tồi: Đó là khi thị trường đặt ra các tiêu chuẩn thì người sử dụng không còn muốn dùng các giao diện tồi của hệ thống đã được thiết kế trước đây mà họ đang sử dụng trong môi trường làm việc của mình.
- Giao diện tồi có thể dẫn tới việc không sử dụng được hoặc sử dụng không có hiệu quả hệ thống. Một giao diện tốt có thể hạn chế được nhược điểm của một thiết kế phần mềm tồi nhưng một giao diện tồi khiến không sử dụng được hệ thống ngay cả khi hệ thống đó có thiết kế phần mềm tuyệt vời.

Một động cơ thúc đẩy nữa là tương tác người máy ngày càng được điều chỉnh nhanh chóng nhờ các tiêu chuẩn.

- Tổ chức tiêu chuẩn quốc tế ISO kết hợp với tổ chức quốc gia như BSI(Anh), ANSI(Mỹ), DIN(Đức) cho ra đời tiêu chuẩn áp dụng cho thiết kế tương tác người máy với số hiệu tiêu chuẩn ISO9241

- Viện tiêu chuẩn vương quốc Anh cấp chứng chỉ “Nhân con điều hâu” đối với những sản phẩm có tương tác người máy tốt. Đối với những sản phẩm được cấp chứng chỉ này sẽ có ưu thế cạnh tranh trên thị trường. Mặt khác, về pháp lí, theo luật Anh và Châu Âu người thiết kế giao diện tồi có thể sẽ bị kiện ra tòa. Và trong tương lai luật này có thể trở thành luật của ISO về thiết kế tương tác người máy.

Do sự ra đời của các tiêu chuẩn và nhận thức của người sử dụng về các sản phẩm tin học nên số lượng các mã được viết cho đầu vào và đầu ra cho giao diện tăng lên, khoảng 50% mã được dùng cho xử lý giao diện trong một sản phẩm.

Người ta cũng nhận thấy rằng giao diện không chỉ là phần bị phê bình nhiều nhất của hệ thống mà còn là phần quan trọng và tốn nhiều chi phí nhất của nhiều chương trình nghiên cứu. Thông qua một số công trình đã được nghiên cứu, người ta nhận thấy rằng thiết kế giao diện tồi có thể gây ra những hậu quả như sau:

- Tần số mắc lỗi trong điều hành hệ thống và nhập liệu tăng lên. Để sửa lại các lỗi này thì sẽ tốn rất nhiều thời gian và tiền của, còn những lỗi không sửa được cho đúng có thể gây hậu quả có hại nếu các quyết định được đưa ra trên cơ sở dữ liệu không chính xác.
- Hệ thống hoạt động kém. Hệ thống có thể không thể quản lí được khối lượng chuyển tác đã được thiết kế, hoặc không đạt được độ chính xác của đầu ra, do hệ thống khó sử dụng và khó học.
- Hệ thống không sử dụng được do người sử dụng phản đối (Bộ quốc phòng Mỹ cho rằng sự thất bại của hệ thống là do thiết kế giao diện tồi và phân tích yêu cầu không đầy đủ).
- Người sử dụng thất vọng khi sử dụng hệ thống. Điều này làm cho năng suất làm việc thấp. Người sử dụng bị căng thẳng và trong trường hợp đặc biệt họ sẽ phá hủy hệ thống.

### **5.3. Các quy tắc thiết kế giao diện người dùng**

#### **5.3.1. Hướng dẫn tương tác chung**

Hướng dẫn về tương tác chung bao hàm cả hiển thị thông tin và vào ra dữ liệu và điều khiển toàn bộ hệ thống. Do đó nó rất hay bị bỏ qua. Những hướng dẫn sau đây tập chung vào tương tác chung.

- Tính nhất quán: Phải dùng định dạng nhất quán cho việc chọn đơn, vào chỉ lệnh, hiển thị dữ liệu và vô số các chức năng khác xuất hiện trong HCI.

- Cho thông tin phản hồi có nghĩa: Cung cấp cho người sử dụng những thông tin phản hồi bằng hình ảnh và âm thanh nhằm thiết lập việc trao đổi thông tin hai chiều(giữa người sử dụng và giao diện).
- Yêu cầu kiểm chứng mọi hành động phá huỷ không tầm thường: Nếu người dùng yêu cầu xoá một tệp, ghi đè lên thông tin bản chất hay yêu cầu kết thúc chương trình thì một thông báo “ Bạn có chắc. . .?” nên xuất hiện ra.
- Cho phép dễ dàng lần ngược nhiều hành động: Các chức năng UNDO (hoàn tác) hay REVERSE (đảo ngược) đã giúp cho hàng nghìn người dùng khỏi mất đi hàng nghìn giờ lam việc. Khả năng lần ngược nên có sẵn trong mọi ứng dụng trong tương tác.
- Giảm thiểu khối lượng thông tin phải ghi nhớ giữa các hành động: Không nên trông đợi người dùng cuối cùng nhớ được một danh sách các số hiệu hay tên gọi để cho người ấy có thể dùng lại trong các chức năng kế sau. Cần phải tối thiểu tải trọng ghi nhớ.
- Tìm kiếm tính hiệu quả trong đối thoại, vận động và ý nghĩ: Nên tối thiểu dùng các phím, cần phải xem xét khoảng cách chuột phải đi qua giữa các điểm trong thiết kế bố trí màn hình và đừng đẩy người dùng vào tình huống phải tự hỏi, “Cái này nghĩa là gì nhỉ?”
- Dung thứ cho sai lầm: Hệ thống nên tự bảo vệ khỏi lỗi của người dùng để khỏi bị chết, hỏng.
- Phân loại các hoạt động theo chức năng và tổ chức màn hình hài hoà theo vùng: Một trong những cái lợi của thực đơn kéo xuống là khả năng tổ chức các lệnh theo kiểu. Về bản chất người thiết kế nên cố gắng đạt các chỉ lệnh và hành động “nhất quán”.
- Cung cấp tiện nghi trợ giúp cảm ngữ cảnh
- Dùng các động từ đơn giản hay cụt động từ ngắn để đặt tên chỉ lệnh. Tên chỉ lệnh dài dòng thì khó nhận dạng và khó nhớ. Nó cũng có thể chiếm không gian không cần thiết trong danh sách đơn.

### **5.3.2. Hướng dẫn về việc hiển thị thông tin**

Nếu thông tin được HCI trình bày không đầy đủ, mơ hồ hay không dễ hiểu thì sẽ không thỏa mãn nhu cầu người dùng. Thông tin được “hiển thị” theo nhiều cách khác nhau: văn bản, tranh ảnh và âm thanh; bằng cách sắp đặt, di chuyển và kích cỡ;

dùng màu sắc, độ phân giải; và thậm chí bằng cả việc bỏ lửng. Các dẫn hướng sau đây tập trung vào hiển thị thông tin:

- Chỉ hiển thị thông tin có liên quan tới ngữ cảnh hiện tại. Người dùng không phải khó nhọc lần qua dữ liệu, đơn và đồ họa phụ để thu được thông tin có liên quan tới một chức năng hệ thống riêng.
- Đừng chôn vùi người dùng dưới dữ liệu – hãy dùng định dạng trình bày cho phép hấp thụ nhanh chóng thông tin. Đồ họa hay sơ đồ nên thay thế cho các bảng lớn.
- Dùng nhãn nhất quán, cách viết tắt chuẩn và màu sắc dự kiến trước được. Ý nghĩa của hiển thị hiển nhiên không cần tham khảo thêm nguồn thông tin ở bên ngoài.
- Cho phép người dùng duy trì ngữ cảnh trực quan. Nếu việc hiển thị đồ họa máy tính được thay đổi tỉ lệ thì hình ảnh gốc nên được hiển thị thường xuyên (dưới dạng rút gọn tại góc màn hình) để cho người dùng hiểu được hiểu được vị trí tương đối của phần hình ảnh hiện đang được xét.
- Đưa ra thông báo lỗi có nghĩa:
  - ✓ Thông báo nên đưa ra những lời khuyên có tính xây dựng để khôi phục từ lỗi.
  - ✓ Thông báo nên đưa ra những lời khuyên có tính chất xây dựng để khôi phục từ lỗi.
  - ✓ Thông báo nên đi kèm với tín hiệu nghe được hay thấy được. Tức là một tiếng bíp có thể được sinh ra đi kèm với việc hiển thị thông báo, hay thông báo có thể nhấp nháy chốc lát hay được hiển thị theo màu để nhận ra như “màu lỗi”
  - ✓ Thông báo nên có tính chất “phi đánh giá”. Tức là lời đưa ra đừng hàm ý trách móc người dùng. Giải thích: Bởi vì không ai thực sự thích tin xấu nên ít người dùng thích thông báo lỗi dù nó được thiết kế như thế nào. Nhưng một triết lý thông báo lỗi có hiệu quả có thể cải thiện được chất lượng của hệ thống và sẽ giảm bớt đáng kể sự chán nản của người dùng khi vấn đề quản thực xuất hiện.
- Dùng chữ hoa, chữ thường, tụt lề và gộp nhóm văn bản để giúp cho việc hiểu. Nhiều thông tin được HCI truyền đạt là văn bản, ngay cả cách bố trí và hình

dạng của văn bản cũng có tác động đáng kể đến sự thoải mái để người dùng hấp thu thông tin.

- Dùng cách hiển thị “trương tự” để biểu diễn những thông tin dễ được hấp thu hơn so với dạng biểu diễn này. Ví dụ, hiển thị áp suất của bể chứa lọc dầu trong xưởng lọc dầu sẽ có ít tác dụng nếu dùng cách biểu diễn số, nhưng nếu hiển thị dạng nhiệt kế được dùng thì chuyển động theo chiều đứng và sự thay đổi màu sắc có thể được dùng để chỉ ra những điều kiện áp suất thay đổi. Điều này sẽ cung cấp cho người dùng cả thông tin tuyệt đối và tương đối.
- Xem xét vùng hiển thị có sẵn trên màn hình và dùng nó một cách có hiệu quả. Khi dùng nhiều cửa sổ, ít nhất nên có sẵn không gian để chỉ ra một phần cho từng cửa sổ này. Bên cạnh đó, kích cỡ màn hình (vấn đề công nghệ hệ thống) nên được lựa chọn để hòa hợp với kiểu ứng dụng cần được cài đặt.

### 5.3.3. Hướng dẫn về việc vào dữ liệu

Phần lớn thời gian của người dùng được dành cho việc chọn lựa các chỉ lệnh, gỡ dữ liệu và cung cấp cái vào cho hệ thống. Trong nhiều ứng dụng, bàn phím vẫn còn là phương tiện đưa vào chính, nhưng chuột, bộ số hóa và thậm chí hệ thống nhận dạng tiếng nói đang nhanh chóng trở thành các phương tiện có hiệu quả. Những hướng dẫn sau đây tập trung vào việc đưa vào dữ liệu:

- Tối thiểu việc số hành động đưa vào mà người dùng cần thực hiện. Việc rút gọn khối lượng gõ là điều yêu cầu trước hết. Điều này có thể được thực hiện bằng cách dùng chuột để chọn từ một tập đã xác định sẵn các cái vào; dùng “thang trượt” để xác định cái vào trong một miền giá trị; dùng “macro” làm cho chỉ một phím được chuyển thành một tập dữ liệu vào phức tạp hơn.
- Duy trì sự nhất quán giữa hiển thị thông tin và cái vào dữ liệu. Các kí tự hiển thị trực quan (như kích cỡ văn bản, màu sắc, cách bố trí) nên được thực hiện đối với miền cái vào.
- Cho phép người dùng làm phù hợp cái vào. Người dùng, chuyên gia có thể quyết định tạo ra các chỉ lệnh đã sửa đổi phù hợp mình hay bỏ qua một số kiểu cảnh báo và kiểm chứng hành động. HCI cho phép làm điều này.
- Tương tác nên mềm dẻo nhưng cũng nên hòa hợp với một đưa vào ưa thích. Mô hình người dùng sẽ trợ giúp cho việc xác định một đưa vào nào là ưa thích. Ví dụ, một thư kí có thể rất thích với cách đưa vào từ bàn phím, trong

khi người quản lí lại có thể thấy thoải mái khi dùng thiết bị trợ và nháy như chuột.

- Khử kích hoạt các chỉ lệnh không thích hợp trong hoàn cảnh của hành động hiện tại. Điều này bảo vệ cho người dùng khỏi phải cố dùng một hành động nào đó có thể làm phát sinh lỗi.
- Để cho người dùng kiểm soát luồng tương tác. Người dùng nên có khả năng nhảy qua các hành động không cần thiết, thay đổi trật tự các hành động yêu cầu (nếu có thể được trong hoàn cảnh của ứng dụng), và khôi phục được từ các điều kiện lỗi mà không phải ra khỏi chương trình.
- Cung cấp trợ giúp cho mọi hành động đưa vào: một số vấn đề khi xem xét tiện nghi trợ giúp bao gồm:
  - ✓ Liệu trợ giúp có sẵn với tất cả các chức năng hệ thống và vào mọi lúc trong tương tác không? Các tùy chọn bao gồm: trợ giúp chỉ cho một tập con của mọi chức năng và hành động; trợ giúp cho tất cả các chức năng.
  - ✓ Người dùng sẽ yêu cầu trợ giúp như thế nào? Các tùy chọn bao gồm: đơn trợ giúp; phím trợ giúp; chỉ lệnh HELP.
  - ✓ Trợ giúp sẽ được trình bày như thế nào? Các tùy chọn bao gồm cửa sổ tách biệt; tham khảo tới một tài liệu in; gợi ý một hay hai dòng được tạo ra trong một vị trí màn hình cố định.
  - ✓ Người dùng sẽ trở về với tương tác thông thường như thế nào? Các tùy chọn bao gồm: nút trở về được hiển thị trên màn hình; phím chức năng hay dấy điều khiển.
  - ✓ Thông tin trợ giúp sẽ được cấu trúc trợ giúp như thế nào? Các tùy chọn bao gồm: cấu trúc “phẳng” trong đó mọi thông tin đều được thâm nhập qua tới một từ khóa; cấp bậc phân tầng của thông tin cung cấp chi tiết ngày càng tăng khi người dùng tiến sâu vào cấu trúc; sử dụng siêu văn bản.
- Khử bỏ việc đưa vào “chuột mickey”. Đừng yêu cầu người dùng phải xác định các đơn vị cho việc đưa vào công nghệ (trừ phi có mơ hồ). Đừng yêu cầu người dùng phải gõ .00 cho toàn bộ số tiền, đưa ra các giá trị mặc định mọi lúc có thể và không bao giờ yêu cầu người dùng đưa vào những thông tin có thể tự động thu thập hay tính toán được bên trong chương trình.

## CHƯƠNG VI: CÁC HOẠT ĐỘNG PHÁT TRIỂN PHẦN MỀM

Trong chương này, chúng ta sẽ tìm hiểu một số vấn đề quan trọng trong kỹ nghệ phần mềm nói chung, đặc biệt, tập trung vào việc mô tả các hoạt động chính trong quy trình phát triển phần mềm theo mô hình vòng đời cổ điển và thảo luận các vấn đề nảy sinh do những yêu cầu đặc biệt khi áp dụng để thiết kế các hệ thống tương tác.

### 6.1. Kỹ nghệ phần mềm

Một định nghĩa ban đầu về kỹ nghệ phần mềm do Fritz Bauer nêu ra trong cuộc hội thảo chính đầu tiên về chủ đề này:

Việc thiết lập và sử dụng các nguyên lý công nghệ đúng đắn để thu được phần mềm một cách kinh tế vừa tin cậy vừa làm việc hiệu quả trên các máy thực. Mặc dầu còn nhiều định nghĩa sâu sắc hơn đã được nêu ra, mọi định nghĩa đều nhấn mạnh vào yêu cầu về một kỉ luật công nghệ trong việc phát triển phần mềm. Kỹ nghệ phần mềm là sự phát triển của kỹ nghệ phần cứng và hệ thống. Nó bao gồm một tập gồm ba yếu tố chủ chốt – phương pháp, công cụ và thủ tục – làm cho người quản lí kiểm soát được tiến trình phát triển phần mềm và cung cấp cho người hành nghề một nền tảng để xây dựng phần mềm chất lượng cao theo một cách thức có hiệu suất. Trong các đoạn sau đây, chúng ta sẽ xem xét tóm tắt từng yếu tố đó.

Các phương pháp kỹ nghệ phần mềm đưa ra các “cách làm” về mặt kĩ thuật để xây dựng phần mềm. Các phương pháp này bao gồm một diện rộng các nhiệm vụ, bao gồm: lập kế hoạch và ước lượng dự án, phân tích yêu cầu hệ thống và phần mềm, thiết kế cấu trúc dữ liệu, kiến trúc chương trình và thủ tục thuật toán, mã hoá, kiểm thử và bảo trì. Các phương pháp cho kỹ nghệ phần mềm thường đưa ra các kĩ pháp đồ hoạ hay hướng ngôn ngữ đặc biệt và đưa ra một tập các tiêu chuẩn về chất lượng phần mềm.

Các công cụ kỹ nghệ phần mềm cung cấp sự hỗ trợ tự động hay bán tự động cho các phương pháp. Ngày nay đã có các công cụ hỗ trợ cho từng phương pháp được nêu trên. Khi các công cụ được tích hợp đến mức các thông tin do công cụ này tạo ra có thể được dùng cho các công cụ khác thì hệ thống hỗ trợ cho việc phát triển phần mềm đã được thiết lập và còn được gọi là kỹ nghệ phần mềm có máy tính hỗ trợ CASE . CASE tổ hợp phần mềm, phần cứng và CSDL kỹ nghệ phần mềm (một cấu trúc dữ liệu chứa các thông tin quan trọng về việc phân tích, thiết kế, mã hóa và kiểm thử) để



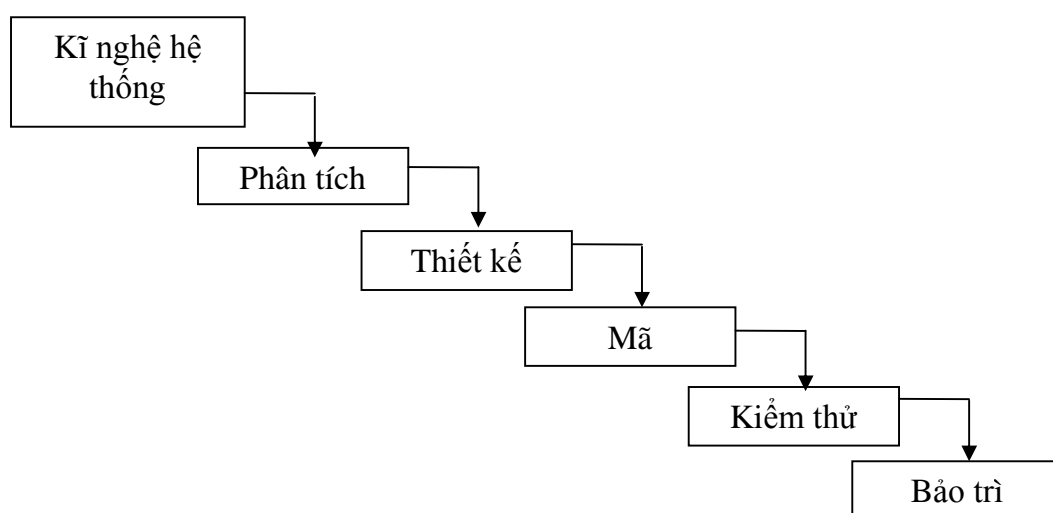
tạo ra môi trường kỹ nghệ phần mềm, điều này cũng tương tự như thiết kế có máy tính hỗ trợ / kỹ nghệ có máy tính hỗ trợ (CAD / CAE) cho phần cứng.

Các thủ tục kỹ nghệ phần mềm là chất keo dán các phương pháp và công cụ lại với nhau và làm cho chúng được sử dụng hợp lý và đúng hạn trong việc phát triển phần mềm máy tính. Thủ tục xác định ra trình tự các phương pháp sẽ được áp dụng, những sản phẩm cần bàn giao (tài liệu, báo cáo, mẫu...v.v) cần cho việc kiểm soát để đảm bảo chất lượng và điều hoà thay đổi, xác định những cột mốc để cho người quản lý phần mềm nắm được tiến độ.

Kỹ nghệ phần mềm bao gồm một tập các bước bao hàm cả phương pháp, công cụ và thủ tục đã được xác định ở trên. Các bước này thường được gọi là *các khuôn cảnh kỹ nghệ phần mềm*. Một khuôn cảnh cho kỹ nghệ phần mềm được lựa chọn dựa trên bản chất của dự án và ứng dụng, phương pháp và công cụ cần dùng, và các kiểm soát cùng việc bàn giao cần thực hiện.

## 6.2. Vòng đời cổ điển

Hình 5.1 minh họa cho khuôn cảnh vòng đời cổ điển đối với kỹ nghệ phần mềm. Đôi khi nó còn được gọi là “mô hình thác nước, khuôn cảnh vòng đời yêu cầu một cách tiếp cận hệ thống, tuân tự tới việc phát triển phần mềm, bắt đầu ở mức hệ thống và tiến dần xuống phân tích, thiết kế, mã hoá, kiểm thử và bảo trì. Được mô hình hoá theo vòng đời kỹ nghệ quy ước, khuôn cảnh vòng đời bao gồm các hoạt động sau:



Hình 5.1 Vòng đời cổ điển

- **Kỹ nghệ và phân tích hệ thống:** Bởi vì phần mềm bao giờ cũng là một phần của một hệ thống lớn hơn nên công việc bắt đầu từ việc thiết lập yêu cầu cho

mọi phần tử hệ thống và rồi cấp phát một tập con các yêu cầu đó cho phần mềm. Quan điểm hệ thống này là điều bản chất khi phần mềm phải tiếp xúc với các thành phần khác như phần cứng, con người và CSDL. Kỹ nghệ hệ thống và phân tích bao gồm việc thu thập yêu cầu ở mức hệ thống với một lượng nhỏ thiết kế và phân tích mức đỉnh. Phân tích yêu cầu phần mềm. Tiến trình thu thập yêu cầu được tập trung và làm mạnh đặc biệt vào phần mềm. Để hiểu được bản chất của các chương trình phải xây dựng, kỹ sư phần mềm (“nhà phân tích”) phải hiểu về lĩnh vực thông tin đối với phần mềm cũng như chức năng cần có, hiệu năng và giao diện. Các yêu cầu cho cả hệ thống và phần mềm cần phải được lập tư liệu và được khách hàng duyệt xét lại.

- **Thiết kế:** Thiết kế phần mềm thực tế là một tiến trình nhiều bước tập trung vào bốn thuộc tính phân biệt của chương trình: cấu trúc dữ liệu, kiến trúc phần mềm, chi tiết thủ tục và đặc trưng giao diện. Tiến trình thiết kế dịch các yêu cầu thành một biểu diễn của một phần mềm có thể được khẳng định về chất lượng trước khi giai đoạn mã hoá bắt đầu. Giống như các yêu cầu, việc thiết kế phải được lập tư liệu và trở thành một phần của cấu hình phần mềm.
- **Mã hoá:** Thiết kế phải được dịch thành dạng máy đọc được. Bước mã hoá thực hiện nhiệm vụ này. Nếu thiết kế được thực hiện theo một cách chi tiết thì việc mã hoá có thể được thực hiện theo một cách chi tiết thì việc mã hoá có thể được thực hiện một cách máy móc.
- **Kiểm thử:** Một khi đã sinh ra mã, việc kiểm thử chương trình bắt đầu. Tiến trình kiểm thử tập trung vào phần logic bên trong của phần mềm, đảm bảo rằng tất cả các câu lệnh đều được kiểm thử, và về phần chức năng bên ngoài thì đảm bảo rằng việc tiến hành kiểm thử phát hiện ra các lỗi và đảm bảo những cái vào xác định sẽ tạo ra kết quả thực tế thống nhất với kết quả muốn có.
- **Bảo trì:** Phần mềm chắc chắn sẽ phải trải qua những thay đổi sau khi nó được bàn giao cho khách hàng (một ngoại lệ có thể là những phần mềm nhúng). Thay đổi sẽ xuất hiện bởi vì gặp phải lỗi, bởi vì phần mềm phải thích ứng với những thay đổi trong môi trường bên ngoài (chẳng hạn như sự thay đổi do hệ điều hành mới hay thiết bị ngoại vi mới), hay bởi vì khách hàng yêu cầu nâng cao chức năng hay hiệu năng. Việc bảo trì phần mềm phải được áp

dụng lại các bước vòng đời nói trên cho chương trình hiện tại chứ không phải chương trình mới.

Vòng đời cổ điển là khuôn cảnh cũ nhất và được sử dụng rộng rãi nhất trong công nghệ phần mềm. Tuy nhiên, trong thập kỉ qua, những chỉ trích về khuôn cảnh này đã làm cho những người ủng hộ nó tích cực phải đặt vấn đề về khả năng ứng dụng của nó trong mọi tình huống. Một số các vấn đề thỉnh thoảng gặp phải khi dùng khuôn cảnh vòng đời cổ điển này là:

- Các dự án thực hiện hiếm khi tuân theo dòng chảy tuần tự mà mô hình đề nghị. Bao giờ việc lặp lại cũng xuất hiện và tạo ra các vấn đề trong việc áp dụng khuôn cảnh này.
- Khách hàng thường khó phát biểu mọi yêu cầu một cách tường minh. Vòng đời cổ điển đòi hỏi điều này và thường khó thích hợp với sự bất trắc tự nhiên tồn tại vào lúc đầu của nhiều dự án.
- Khách hàng phải kiên nhẫn. Bản làm việc được của chương trình chỉ có được vào lúc cuối của thời gian dự án. Một sai lầm ngớ ngẩn, nếu đến khi có chương trình làm việc mới phát hiện ra, có thể sẽ là một thảm họa

**Vấn đề nảy sinh khi thiết kế các hệ thống tương tác:** Trong mô hình vòng đời phần mềm truyền thống được trình bày ở trên, một vấn đề quan trọng cần phải được thực hiện ngay từ đầu là phải xác định được các yêu cầu của hệ thống. Tuy nhiên, khi áp dụng để thiết kế các hệ thống tương tác có 1 vấn đề nảy sinh đó là tất cả các yêu cầu của hệ thống tương tác không thể xác định được ngay từ ban đầu. Do đó, hệ thống cần phải được xây dựng, sau đó quá trình tương tác với người sử dụng sẽ được quan sát và đánh giá để tìm ra các phương pháp làm cho việc tương tác trở nên dễ dàng hơn. Các mô hình về người sử dụng được trình bày trong chương 6 của tài liệu này sẽ giúp cho người thiết kế nắm bắt được rõ hơn các yêu cầu của người sử dụng khi bắt đầu tiến hành thiết kế.

### **6.3. Làm bản mẫu**

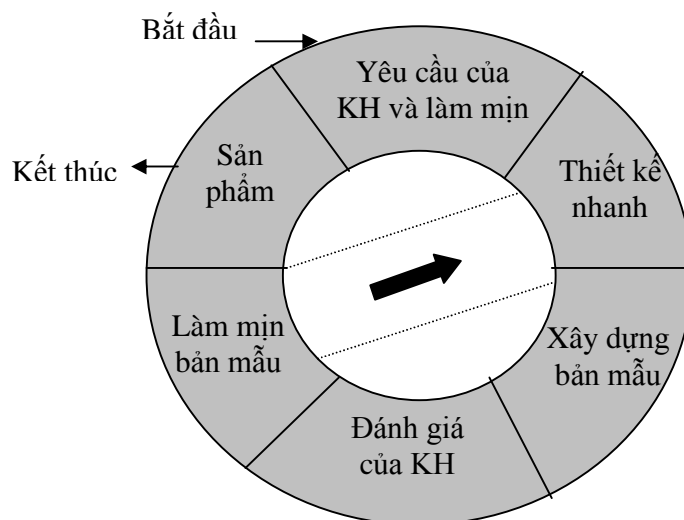
Là một công đoạn vô cùng quan trọng xét theo góc độ kĩ thuật và kinh doanh. Thông thường khách hàng đã xác định một tập mục tiêu tổng quát cho phần mềm, nhưng còn chưa xác định được đầu vào, xử lý hay yêu cầu đầu ra. Trong các trường hợp khác, người phát triển có thể không chắc chắn về tính hiệu quả của một thuật toán, việc thích nghi hệ điều hành hay dạng giao diện người – máy cần có. Trong

những trường hợp này và nhiều trường hợp khác cách tiếp cận làm bản mẫu cho kỹ nghệ phần mềm có thể là một cách tiếp cận tốt nhất

Làm bản mẫu là một tiến trình làm cho người phát triển có khả năng tạo ra một mô hình cho phần mềm cần phải xây dựng. Mô hình có thể lấy một trong 3 dạng:

- Bản mẫu trên giấy hay mô hình dựa trên máy tính mô tả giao diện người máy dưới dạng làm cho người dùng hiểu được cách các tương tác xuất hiện.
- Bản mẫu làm việc cài đặt một tập con chức năng của phần mềm mong muốn.
- Một chương trình đã có thực hiện một phần hay tất cả các chức năng mong muốn nhưng cần phải cải tiến thêm các tính năng khác tùy theo nỗ lực phát triển mới.

Quy trình làm bản mẫu được cho trong hình vẽ sau (hình 5.2).



Hình 5.2: Làm bản mẫu

Việc làm bản mẫu được bắt đầu bằng việc thu thập yêu cầu. Người phát triển và khách hàng gặp nhau và xác định các mục tiêu tổng thể cho phần mềm, xác định các yêu cầu nào đã biết và miền nào cần phải xác định thêm. Rồi đến việc thiết kế nhanh. Thiết kế nhanh tập trung vào việc biểu diễn các khía cạnh của phần mềm thấy được đối với người dùng. Thiết kế nhanh dẫn đến việc xây dựng một bản mẫu. Bản mẫu được khách hàng/người sử dụng đánh giá và được dùng để làm mịn các yêu cầu đối với phần mềm cần phát triển. Tiến trình lặp đi lặp lại xảy ra để cho bản mẫu được “vì chính” thỏa mãn nhu cầu của khách hàng trong khi đồng thời lại làm cho người phát triển hiểu được kỹ hơn cần phải thực hiện nhu cầu nào.

## **CHƯƠNG VII: CÁC MÔ HÌNH CỦA NGƯỜI DÙNG TRONG THIẾT KẾ GIAO DIỆN**

### **7.1. Mô hình các yêu cầu của người dùng**

Xác định các yêu cầu là một phần quan trọng trong tất cả các phương pháp luận của kỹ nghệ phần mềm. Tuy nhiên hoạt động này mới chủ yếu tập trung vào việc xác định các yêu cầu chức năng của hệ thống - những gì hệ thống cần phải thực hiện – mà ít nhấn mạnh đến các yêu cầu phi chức năng của người dùng như là tính dễ dùng và tính chấp nhận (có thể chấp nhận được). Ngay cả khi những vấn đề đó được xem xét thì những yêu cầu đó chỉ là phản ánh góc độ quản lý đối với những yêu cầu của người dùng hơn là thu thập thông tin từ chính bản thân người sử dụng. Các mô hình yêu cầu của người sử dụng sẽ làm nhiệm vụ thiết lập lại sự cân bằng đó. Có rất nhiều mô hình và phương pháp có thể được sử dụng để xác định các yêu cầu hệ thống dưới góc độ rộng hơn. Nhưng trong tài liệu này chúng ta chỉ xem xét đến 3 mô hình trong số đó, đó là: các mô hình kỹ thuật- xã hội, phương pháp luận các hệ thống phần mềm, và thiết kế hợp tác.

### **7.2. Các mô hình kỹ thuật-xã hội**

Các mô hình xã hội – kỹ thuật quan tâm đến các khía cạnh về mặt kỹ thuật, xã hội, tổ chức, và con người của thiết kế. Chúng ta thấy rằng công nghệ không thể được phát triển một cách độc lập mà nó là một phần của một môi trường có tổ chức rộng hơn. Do đó điều quan trọng là chúng ta sẽ phải xem xét đến các vấn đề kỹ thuật và xã hội cùng nhau. Có rất nhiều mô hình xã hội – kỹ thuật được áp dụng trong quá trình thiết kế các hệ thống máy tính tương tác nhưng chúng ta chỉ xem xét 3 mô hình sau: Các kỹ năng của người dùng và sự phù hợp nhiệm vụ (USTM); và một dạng của nó dùng cho các tổ chức nhỏ là CUSTOM, phân tích nhiệm vụ các hệ thống mở (OSTA), và mô hình nâng cao hiệu quả việc thực hiện về mặt kỹ thuật và con người của các hệ thống máy tính (ETHICS).

#### **7.2.1. USTM/CUSTOM**

USTM là một cách tiếp cận xã hội- kỹ thuật được phát triển cho phép các nhóm thiết kế hiểu và cung cấp tài liệu một cách đầy đủ về các yêu cầu của người dùng. Nó sử dụng các mô hình nhiệm vụ được dạng biểu đồ cùng với các miêu tả bằng tiếng

Anh đề tập hợp các phương pháp cấu trúc và các nhân tố con người. USTM làm theo đơn đặt hàng của khách hàng dùng cho các tổ chức nhỏ được gọi là CUSTOM, CUSTOM tập trung vào việc thiết lập các yêu cầu của người góp cổ phần (stakeholder): tất cả mọi người góp cổ phần đều được xem xét, chứ không phải chỉ xem xét đến người dùng cuối. Người góp cổ phần là người chịu ảnh hưởng bởi sự thất bại hoặc thành công của hệ thống. Có 4 nhóm người góp cổ phần được phân biệt như sau:

- Nhóm thứ nhất là những người sử dụng hệ thống
- Nhóm thứ hai là những người không trực tiếp sử dụng hệ thống nhưng họ lại nhận thông tin đầu ra từ hệ thống hoặc cung cấp thông tin đầu vào cho hệ thống (ví dụ, người nhận báo cáo được tạo ra bởi hệ thống)
- Nhóm thứ 3 là những người không nằm trong 2 nhóm trên nhưng họ chịu ảnh hưởng bởi sự thành công hay thất bại của hệ thống (ví dụ, một giám đốc cũng là người đặt hàng bởi lợi nhuận bởi vì sự tăng hay giảm lợi nhuận của họ phụ thuộc vào sự thành công của hệ thống)
- Nhóm thứ tư: là những người tham gia thiết kế, phát triển và bảo trì hệ thống

**Vi dụ:** Một sân bay quốc tế đang xem xét việc xây dựng một hệ thống bán vé mới liên kết các đại lý bán vé để bán vé trực tiếp đến khách hàng. Hãy phân loại các nhóm người góp cổ phần trong hệ thống bán vé này.

- Nhóm thứ nhất: nhân viên đại lý bán vé, nhân viên bán vé
- Nhóm thứ hai: các khách hàng, nhân viên quản lý
- Nhóm thứ ba: các đối thủ cạnh tranh, các cơ quan hàng không dân dụng, những người bạn du lịch của khách hàng, các cổ đông
- Nhóm thứ tư: nhóm thiết kế, nhân viên phòng IT

Mô hình CUSTOM được áp dụng ở giai đoạn đầu của quá trình thiết kế khi mà cơ hội sản phẩm được xác định. Nó là một phương pháp dựa trên các form liệt kê một tập các câu hỏi để áp dụng ở mỗi giai đoạn của mô hình:

- **Giai đoạn 1:** Miêu tả ngữ cảnh tổ chức, bao gồm các mục đích chủ yếu, các đặc điểm tự nhiên, nền tảng kinh tế chính trị

- **Giai đoạn 2:** Nhận dạng và mô tả các người góp cổ phần: Tất cả mọi người góp cổ phần phải được đặt tên, phân loại thành các nhóm và đưa ra các miêu tả liên quan đến các vấn đề cá nhân, vai trò của họ trong tổ chức và công việc của họ. Ví dụ, CUSTOM xác định các vấn đề như là các sự năng động, sự chán nản, tri thức, các kĩ năng, quyền và ảnh hưởng trong tổ chức...của người góp cổ phần.
- **Giai đoạn 3:** Nhận dạng và mô tả các nhóm làm việc: Một nhóm làm việc là một nhóm người bất kì làm việc cùng nhau trong cùng một nhiệm vụ cho dù họ có được thành lập một cách chính thức hay là không chính thức. Các nhóm làm việc được mô tả theo vai trò của họ trong tổ chức và các đặc điểm của họ
- **Giai đoạn 4:** Nhận dạng và mô tả các cặp đối tượng - nhiệm vụ: Có rất nhiều nhiệm vụ phải được thực hiện cùng với các đối tượng. Do đó cần nhận biết và mô tả các cặp như vậy.
- **Giai đoạn 5:** Nhận dạng nhu cầu của những người góp cổ phần: Các giai đoạn 2 –4 được miêu tả với thống hiện tại và hệ thống đang xây dựng. Nhu cầu của các thành viên được nhận dạng bằng cách xem xét sự khác nhau giữa nhu cầu của 2 người góp cổ phần. Ví dụ, hiện tại nếu có một người thiếu một kĩ năng được yêu cầu trong hệ thống thì nhu cầu đào tạo đó được nhận dạng như là một nhu cầu
- **Giai đoạn 6:** Tập hợp và kiểm tra các yêu cầu của người góp cổ phần

**Ví dụ:** Form liệt kê một tập các câu hỏi điều tra các đặc điểm của người góp cổ phần có dạng như sau:

- Người góp cổ phần phải đạt được điều gì ? Và mức độ thành công được đánh giá như thế nào?
- Nguồn gốc sự thoả mãn về công việc của người góp cổ phần là gì? Nguồn gốc của sự bất mãn hay căng thẳng là gì?
- Kiến thức và các kĩ năng mà người góp cổ phần có?
- Thái độ của người góp cổ phần đối với công việc và công nghệ máy tính ?

- Có bao nhiêu đặc điểm của nhóm làm việc ảnh hưởng đến tính chấp nhận của sản phẩm đối với người góp cổ phần
- Tính chất nhiệm vụ của người góp cổ phần liên quan đến tính liên tục, rời rạc và quyền lựa chọn hành động là gì?
- Người góp cổ phần có phải xem xét đến các vấn đề đặc biệt liên quan đến trách nhiệm, tính bảo mật hoặc tính riêng tư hay không?
- Điều kiện tự nhiên nơi người góp cổ phần làm việc là gì?

CUSTOM cung cấp một khung hữu ích trong việc xác định các yêu cầu của người góp cổ phần bởi vì việc sử dụng các form và các câu hỏi là tương đối dễ dàng áp dụng. Tuy nhiên, trong các tình huống phức tạp có thể sử dụng phiên bản rút ngắn của CUSTOM (chỉ sử dụng các giai đoạn từ 2 đến 4).

### 7.2.2. OSTA

OSTA tập trung vào việc mô tả những gì xảy ra khi hệ thống kỹ thuật được đưa vào áp dụng trong môi trường làm việc của một tổ chức nào đó. Trong OSTA, các khía cạnh xã hội của hệ thống (như là tính dễ dùng và tính chấp nhận) được đặc tả cùng với các khía cạnh kỹ thuật (như là các chức năng của hệ thống)

OSTA có 8 giai đoạn chính:

- **Giai đoạn 1:** Nhận dạng nhiệm vụ cơ bản mà công nghệ có thể trợ giúp theo mục đích của người sử dụng
- **Giai đoạn 2:** Nhận dạng các đầu vào nhiệm vụ của hệ thống. Các đầu vào đó có thể có các tài nguyên và các form khác nhau phụ thuộc vào thiết kế
- **Giai đoạn 3:** Miêu tả môi trường bên ngoài của tổ chức nơi mà hệ thống được đưa vào áp dụng bao gồm các khía cạnh tự nhiên, kinh tế và chính trị
- **Giai đoạn 4:** Mô tả các chu trình chuyển đổi bên trong hệ thống liên quan đến các cặp hành động được thực hiện cùng với các đối tượng
- **Giai đoạn 5:** Phân tích các hệ thống xã hội, có xem xét đến các nhóm làm việc hiện tại và các mối quan hệ bên trong và bên ngoài tổ chức
- **Giai đoạn 6:** Mô tả về cấu hình và sự tương tác với các hệ thống khác của hệ thống kỹ thuật



- **Giai đoạn 7:** Thiết lập các tiêu chuẩn thoả mãn, chỉ rõ các yêu cầu xã hội và kỹ thuật của hệ thống
- **Giai đoạn 8:** Đặc tả hệ thống kỹ thuật mới

Kết quả của OSTA được biểu diễn bằng cách sử dụng các kí hiệu rất quen thuộc đối với những người thiết kế như là các sơ đồ luồng dữ liệu và các miêu tả dạng text

### 7.2.3. ETHICS

ETHICS cũng liên quan đến việc xác định các yêu cầu xã hội và kỹ thuật nhưng nó khác so với OSTA là nó xác định các yêu cầu xã hội – kỹ thuật của thiết kế một cách song song bằng cách sử dụng các nhóm thiết kế khác nhau. Trong phương pháp ETHICS, các nhóm thiết kế làm việc tách rời và sau đó kết hợp các giải pháp của các nhóm lại để tìm ra một giải pháp hiệu quả nhất tương thích với các yêu cầu xã hội – kỹ thuật đã được nhận dạng.

ETHICS bao gồm 6 giai đoạn chính:

- **Giai đoạn 1:** Nhận dạng vấn đề và mô tả hệ thống hiện tại. Nhận dạng các mục tiêu và nhiệm vụ như là các nhu cầu thông tin và các yêu cầu thoả mãn công việc. Nhận dạng các ràng buộc trong hệ thống, theo cả khía cạnh xã hội và kỹ thuật.
- **Giai đoạn 2:** Thành lập 2 nhóm thiết kế, một nhóm làm về khía cạnh kỹ thuật, một nhóm làm về khía cạnh xã hội. Các mục tiêu và nhu cầu được nhận dạng trong giai đoạn 1 được sắp xếp theo thứ tự ưu tiên và được kiểm tra về tính tương thích trước khi thực hiện các quyết định thiết kế xã hội – kỹ thuật.
- **Giai đoạn 3:** Hoán đổi các giải pháp xã hội – kỹ thuật để đánh giá theo tiêu chuẩn đã được thiết lập để đưa ra danh sách rút gọn các giải pháp có khả năng.
- **Giai đoạn 4:** Kiểm tra tính tương thích của các giải pháp lựa từ giai đoạn 3
- **Giai đoạn 5:** Sắp xếp các cặp tương thích của các giải pháp xã hội – kỹ thuật theo tiêu chuẩn đã được thoả thuận
- **Giai đoạn 6:** Phát triển các thiết kế chi tiết

Cách tiếp cận ETHICS cố gắng để đạt đến một giải pháp thoả mãn được cả các yêu cầu xã hội lẫn các yêu cầu kỹ thuật do các nhóm chuyên gia lựa chọn và đánh giá theo tiêu chuẩn.

### **7.3. Phương pháp luận các hệ thống phần mềm**

Các mô hình xã hội – kỹ thuật tập trung vào việc xác định các yêu cầu của người sử dụng xét trên các khía cạnh về mặt con người và kỹ thuật. Phương pháp luận các hệ thống phần mềm (SSM) đưa ra một cái nhìn rộng hơn về tổ chức cũng như về hệ thống trong đó công nghệ và con người là các thành phần chính. SSM được phát triển bởi Checkland để giúp những người thiết kế có được sự hiểu biết về ngữ cảnh của những sự phát triển công nghệ: do đó tập trung vào sự hiểu biết tình huống hơn là hiểu biết về sự phát minh ra một giải pháp. SSM có 7 giai đoạn, trong đó có một sự khác biệt giữa các giai đoạn “thế giới thực” (1-2, 5-7) và các giai đoạn hệ thống (3-4).

Giai đoạn đầu là giai đoạn nhận dạng vấn đề và bắt đầu tiến hành phân tích. Quá trình này được thực hiện theo miêu tả chi tiết về tình huống vấn đề: phát triển một bức tranh tổng thể. Bức tranh này bao gồm tất cả mọi người góp cổ phần, những nhiệm vụ mà họ thực hiện, và các nhóm làm việc, cấu trúc của tổ chức, các vấn đề và các quy trình của tổ chức được xây dựng bởi mọi người góp cổ phần. Mọi kỹ thuật suy diễn tri thức đều có thể được sử dụng để tập hợp thông tin để xây dựng bức tranh tổng thể đó, bao gồm các kỹ thuật như quan sát (ghi lại bằng video và audio), phỏng vấn và sử dụng các bảng câu hỏi thăm dò ý kiến có cấu trúc hoặc không có cấu trúc, các hội thảo khoa học hợp tác như là các hoạt động đóng vai, mô phỏng và phân tích sự việc. Nói chung, để bắt đầu chúng ta nên sử dụng các cách tiếp cận ít mang tính cấu trúc để tránh những ràng buộc ảo trong miêu tả. Bức tranh tổng thể có thể có nhiều loại – trong đó không có các câu hỏi đúng sai – nhưng nó nên cung cấp thông tin rõ ràng cho người thiết kế.

Giai đoạn tiếp theo chúng ta sẽ chuyển từ thế giới hiện thực sang thế giới các hệ thống và cố gắng tạo ra các định nghĩa gốc cho hệ thống. Có thể có nhiều loại định nghĩa gốc của một hệ thống, ví dụ biểu diễn cho từng người góp cổ phần. Các định nghĩa gốc được miêu tả dưới dạng CATWOE:

- **Clients:** Là những người nhận thông tin đầu ra hoặc nhận lợi ích từ hệ thống
- **Actors:** Là những người thực hiện các hoạt động bên trong hệ thống

- **Transformations:** Những thay đổi bị tác động bởi hệ thống. Đây là một phần quan trọng của định nghĩa gốc vì nó kéo theo các hoạt động cần thiết cho các giai đoạn tiếp theo. Để nhận dạng các biến đổi, hãy xem xét đến các đầu vào và đầu ra của hệ thống.
- **Weltanschauung:** (theo tiếng Đức) hoặc là **World View:** Đó là cách hệ thống được thu nhận như thế nào trong một định nghĩa gốc cụ thể
- **Owner:** Là những người sở hữu, hoặc chịu trách nhiệm về hệ thống, hoặc là những người có quyền tạo ra những thay đổi đối với hệ thống
- **Environment:** Môi trường mà hệ thống hoạt động và chịu ảnh hưởng.

**Ví dụ:** Định nghĩa gốc cho quản lý hàng không: hệ thống bán vé máy bay

Một sân bay quốc tế đang xem xét việc xây dựng một hệ thống bán vé mới liên kết các đại lý bán vé để bán trực tiếp đến khách hàng. Nghĩa là, hệ thống được sử dụng bởi cơ quan quản lý hàng không, được sử dụng bởi các nhân viên đại lý du lịch, các nhân viên đại lý bán vé, hoạt động theo các quy tắc được quy định bởi các cơ quan hàng không dân dụng quốc tế, và luật hợp đồng kinh tế quốc gia, để bán vé và đảm bảo chỗ cho những khách hàng đã đặt vé trước và tạo ra lợi nhuận cho công ty. Khi đó định nghĩa gốc được định nghĩa như sau:

- **Client:** Khách hàng
- **Actor:** Nhân viên đại lý du lịch
- **Transformation:** Dự định và yêu cầu của khách du lịch được chuyển đổi thành giá ghế trên máy bay và lợi nhuận cho tổ chức.
- **Weltanschauung:** Nếu việc bán vé có hiệu quả thì lợi nhuận của tổ chức sẽ được tối ưu hoá.
- **Owner:** Cơ quan quản lý hàng không
- **Environment:** Các quy tắc của cơ quan hàng không dân dụng quốc tế và luật hợp đồng kinh tế quốc gia; chính sách của đại lý bán vé địa phương.

Sau khi phát triển định nghĩa gốc, chúng ta tiến hành xây dựng mô hình quan niệm. Mô hình quan niệm định nghĩa những việc mà hệ thống phải thực hiện để hoàn thành các định nghĩa gốc. Nó cũng bao gồm việc nhận dạng các chuyển đổi và các hoạt động trong hệ thống. Sau đó, những gì mà hệ thống đạt được và cách thức thực

hiện của hệ thống sẽ được mô hình hoá theo một mô hình phân cấp. Quá trình này mang tính lặp và có thể sẽ phải mất một số bước lặp trước khi có được một mô hình hoàn thiện và chính xác.

Tiếp theo, chúng ta sẽ so sánh hệ thống thực với hệ thống của mô hình quan niệm, nhận dạng sự khác nhau và do đó làm nổi bật những thay đổi cần thiết hoặc các vấn đề tiềm tàng. Ví dụ, khi so sánh ta thấy một hoạt động nào đó có nhiều quy trình trong thực tế hơn trong mô hình quan niệm, do đó chúng ta cần giảm bớt số lượng quy trình cho hoạt động đó.

Trong các giai đoạn cuối chúng ta cần xác định những thay đổi cần thiết và có ích cho hệ thống – ví dụ những thay đổi có thể là những thay đổi về cấu trúc, thủ tục hoặc xã hội và quyết định những hành động cần thiết để tác động đến những thay đổi đó.

SSM là một cách tiếp cận linh hoạt cho phép xem xét đến mức chi tiết của ngữ cảnh thiết kế. Tuy nhiên, SSM chỉ có thể được coi là thành công nếu như nó có tác dụng giúp cho người thiết kế hiểu rõ hơn về hệ thống mà họ đang phát triển.

#### **7.4. Thiết kế hợp tác**

Thiết kế hợp tác là một triết lý giúp cho việc hoàn thiện toàn bộ quy trình thiết kế. Thiết kế hợp tác là thiết kế theo ngữ cảnh công việc, xem người sử dụng không chỉ như là một đối tượng thí nghiệm mà còn như một thành viên của nhóm thiết kế. Do đó, người sử dụng trở thành các cộng tác viên tích cực trong quy trình thiết kế chứ không phải là người tham gia thụ động và sự tham gia của người sử dụng hoàn toàn được quản lý bởi người thiết kế. Lý do là người sử dụng là các chuyên gia trong phạm vi công việc và người thiết kế chỉ có ảnh hưởng trong ngữ cảnh đó nếu như các chuyên gia - người sử dụng đồng ý đóng góp một cách tích cực vào thiết kế. Ngoài ra, việc giới thiệu một hệ thống mới có khả năng sẽ làm thay đổi môi trường công việc và các quy trình tổ chức, và hệ thống mới sẽ chỉ được chấp nhận nếu những thay đổi là có thể chấp nhận được đối với người sử dụng. Do đó, thiết kế hợp tác nhằm mục đích cải tiến các yêu cầu hệ thống thông qua một quy trình thiết kế mang tính lặp trong đó người sử dụng tham gia một cách tích cực vào quy trình thiết kế.

Thiết kế hợp tác có 3 đặc điểm quan trọng. Thứ nhất, nó cải tiến môi trường làm việc và nhiệm vụ thông qua việc giới thiệu thiết kế. Điều này làm cho thiết kế và đánh

giá mang tính hướng ngữ cảnh hoặc công việc hơn là hướng hệ thống. Thứ hai, nó đặc trưng bởi sự hợp tác: người sử dụng tham gia vào nhóm thiết kế và có thể đóng góp ở mọi giai đoạn của quá trình thiết kế. Cuối cùng, cách tiếp cận của thiết kế hợp tác mang tính lặp: thiết kế có thể được đánh giá và chỉnh sửa lại ở mỗi giai đoạn.

Quy trình thiết kế hợp tác sử dụng một số phương pháp để giúp cho việc trao đổi thông tin giữa người sử dụng và người thiết kế. Đó là:

- **Brainstorming (Phương pháp trí tuệ nhóm):** Tất cả mọi thành viên tham gia vào quy trình thiết kế đều đưa ra các ý tưởng của mình. Ý tưởng này có thể là không chính thức hoặc không có cấu trúc. Tất cả mọi thông tin đều được ghi lại mà không có sự điều chỉnh và sau đó những ý tưởng đó có thể được chọn lọc bằng cách sử dụng các kỹ thuật khác nhau.
- **Storyboarding (Phương pháp xây dựng bảng tình tiết - một thuật ngữ bắt nguồn từ lĩnh vực điện ảnh) :** phương pháp này đã được trình bày một cách chi tiết trong chương 5. Những người góp cổ phần có thể được sử dụng như một phương tiện để miêu tả các hoạt động thường ngày của người sử dụng cũng như các thiết kế có khả năng và những tác động mà các thiết kế đó sẽ gây ra.
- **Workshops (hội thảo):** Phương pháp này có thể sử dụng để hoàn thiện các tri thức còn thiếu về những thành viên tham gia và cung cấp một cái nhìn tập trung hơn về thiết kế. Phương pháp có thể dùng để chất vấn lẫn nhau nhằm mục đích cho các bên tham gia có thể hiểu hơn về ngữ cảnh thiết kế từ quan điểm của mỗi thành viên. Người thiết kế có thể hỏi người sử dụng về môi trường làm việc và người sử dụng có thể hỏi người thiết kế về công nghệ và các khả năng sẵn có. Điều này tạo ra một nền tảng chung giữa người sử dụng và người thiết kế và thiết lập cơ sở cho thiết kế. Việc sử dụng phương pháp sắm vai cũng có thể cho phép người sử dụng và người thiết kế từng bước thấu hiểu lẫn nhau.
- **Pencil and paper exercises (Xây dựng phác thảo):** Phương pháp này cho phép các thiết kế được thảo luận và đánh giá với rất ít lỗi xảy ra. Người sử dụng có thể biết được mọi nhiệm vụ điển hình bằng cách sử dụng bản phác thảo nháp của thiết kế hệ thống. Điều này là chỉ rõ sự khác nhau giữa các yêu cầu

của người dùng và thiết kế thực tế. Phương pháp này cho cung cấp một kỹ thuật đơn giản và ít tốn kém cho việc đánh giá các mô hình ban đầu.

Những phương pháp trên không phải được sử dụng một cách duy nhất trong thiết kế hợp tác. Chúng có thể được sử dụng một cách rộng rãi hơn để nâng cao sự hiểu biết giữa người thiết kế và người sử dụng. Thông thường trong ngữ cảnh thiết kế (ví dụ, các ràng buộc về tổ chức) không có khả năng để thực hiện thiết kế hợp tác hoàn toàn. Thậm chí nếu có ngoại lệ, thì các phương pháp trên có thể là những cách rất hữu ích cho việc đẩy mạnh sự hợp tác giữa người sử dụng và người thiết kế.

### **7.5. Các mô hình nhận thức**

Các kỹ thuật và mô hình còn lại trong chương này đều nhằm mục đích biểu diễn người sử dụng khi họ tương tác với giao diện. Nghĩa là, chúng mô hình một số khía cạnh hiểu biết, tri thức, ý định và cách xử lý của người sử dụng. Các kỹ thuật khác nhau thì có cách biểu diễn khác nhau, từ các mô hình mục đích và kết quả của các hoạt động giải quyết vấn đề mức cao, cho đến các miêu tả về hoạt động mức vận động như là các cử động của tay khi di chuyển chuột hoặc bấm phím. Các hình thức của chúng được phát triển phần lớn do các nhà tâm lý học hoặc các nhà khoa học máy tính, những người quan tâm nghiên cứu về hành vi của người sử dụng.

Có 2 cách phân loại các mô hình nói trên. Nếu phân loại theo cách miêu tả các đặc điểm về khả năng và hiệu quả thực hiện của người sử dụng thì có 2 loại mô hình sau:

- Mô hình khả năng: Là các mô hình dự báo các kết quả hợp lệ của hành vi nhưng thường không quan tâm đến xem liệu rằng chúng thực sự có thể thực hiện bởi người sử dụng hay không.
- Mô hình hiệu quả thực hiện: Ngược lại với mô hình khả năng, các mô hình hiệu quả thực hiện không chỉ miêu tả những kết quả hành vi cần thiết mà còn thực sự miêu tả những gì người sử dụng cần phải biết và cách họ áp dụng kiến thức đó vào việc thực thi các nhiệm vụ thực sự.

Do đó, các mô hình khả năng biểu diễn loại hành vi có thể của người sử dụng, nhưng chúng hỗ trợ rất ít trong việc phân tích hành vi để xác định yêu cầu của người sử dụng. Ngược lại, các mô hình hiệu quả thực hiện chủ yếu cung cấp tiềm năng phân tích bằng cách tập trung vào các hành vi quen thuộc trong một số ứng dụng hạn chế.

Nếu phân loại theo khả năng hiểu biết về người sử dụng, cũng như ngôn ngữ nhiệm vụ của họ và theo sự ăn khớp giữa ngôn ngữ nhiệm vụ và ngôn ngữ đầu vào, thì các mô hình nhận thức được phân ra thành 3 nhóm sau:

- Mô hình phân cấp cấu trúc mục đích và nhiệm vụ của người sử dụng
- Các mô hình văn phạm và ngôn ngữ
- Các mô hình vật lý và thiết bị

Tuy nhiên, trong phạm vi tài liệu này, chúng ta chỉ tập trung nghiên cứu các mô hình trong phân nhóm mô hình phân cấp mục đích và nhiệm vụ. Đây cũng chính là các mô hình đang được sử dụng rộng rãi và có ảnh hưởng lớn nhất hiện nay.

### **7.6. Mô hình phân cấp mục đích và nhiệm vụ**

Có rất nhiều mô hình sử dụng mô hình xử lý thông tin của não bộ con người trong đó người sử dụng đạt được mục đích bằng cách giải quyết các mục đích nhỏ hơn theo phương pháp chia để trị làm đặc điểm chính. Chúng ta sẽ xem xét 2 mô hình sử dụng phương pháp này đó là GOMS và CCT, đặc biệt là mô hình GOMS, một mô hình đang được sử dụng rộng rãi nhất.

Giả sử rằng chúng ta muốn tạo ra một báo cáo về tình hình bán lẻ cuốn sách HCI. Để đạt được mục đích này, chúng ta cần phân chia mục đích này thành các mục đích nhỏ hơn, như là tập hợp dữ liệu, tạo ra các bảng và các biểu đồ, và viết tài liệu miêu tả. Trong mục đích tập hợp dữ liệu, chúng ta lại phân chia thành các mục đích nhỏ hơn: tìm tên của tất cả các cuốn sách về HCI và sau đó tìm kiếm những cuốn sách đó trong cơ sở dữ liệu của các cửa hàng bán lẻ sách. Tương tự như vậy, mỗi mục đích nhỏ lại được phân chia thành các mục đích nhỏ hơn nữa, quá trình cứ tiếp tục như vậy cho đến tận khi một số mức chi tiết được tìm thấy. Do đó, kết quả của quá trình là một sơ đồ phân cấp mục đích và các mục đích con của nó. Cấu trúc của ví dụ có thể được minh họa như sau:

Produce Report

Gather data

- Find book names
- Do keywords search of names database  
<further subgoals>
- sift through names and abstracts by hand

<further subgoals>

- Search sales database

<further subgoals>

Layout tables and histograms

<further subgoals>

Write description

<further subgoals>

Có một số vấn đề mà chúng ta cần xem xét khi sử dụng phương pháp phân tích trên đó là:

Điểm dừng của quá trình phân chia? Chúng ta có nên tiếp tục phân rã nhiệm vụ cho đến tận mức vận động của người sử dụng hay là dừng ở một mức trừu tượng hơn? Chúng ta nên bắt đầu phân rã từ đâu? Bởi vì chúng ta hoàn toàn có thể bắt đầu phân tích từ các điểm khác nhau trên cây phân cấp mục đích. Và liệu rằng chúng ta có thể mở rộng phân tích sang các mục đích lớn hơn không? Ví dụ như “bật lò” là mục đích con của mục đích “luộc đậu” và tiếp đến là các mục đích lớn hơn như “dọn bàn”, và “ăn”.

Đó là những câu hỏi liên quan đến vấn đề tính cốt lõi, và cả hai mô hình trên đều dành cho người thiết kế quyền tự quyết định khi họ làm việc thực tế. Các vấn đề thiết kế khác nhau đòi hỏi các mức phân tích khác nhau. Tuy nhiên, cả 2 phương pháp đều hoạt động ở một mức tương đối thấp; cố gắng bắt đầu với một mục đích trừu tượng liên quan đến óc sáng tạo thực tế và khả năng giải quyết vấn đề khó khăn theo kiểu như là “tạo ra một báo cáo”. Thay vì giới hạn chúng theo các hành vi học hỏi thông thường hơn. Trong đó, nhiệm vụ trừu tượng quan trọng nhất được xem như là một nhiệm vụ đơn vị. Nhiệm vụ đơn vị thì không đòi hỏi các kỹ năng giải quyết vấn đề của người sử dụng, mặc dù nó đòi hỏi khá nhiều các kỹ năng đó đối với người thiết kế.

Chúng ta sẽ làm gì khi chúng ta có nhiều cách giải quyết một vấn đề, hoặc nếu các cách giải quyết cho 2 mục đích con có tác động lẫn nhau? Người sử dụng thường sử dụng nhiều hơn một cách để đạt được mục đích và do đó thiết kế cần có nhiều cách và thể hiện được cách thức lựa chọn giải pháp của người sử dụng.

Một vấn đề quan trọng khác đó là vấn đề xử lý lỗi. Người sử dụng thì không phải là những người hoàn thiện và tất cả mọi người đều như vậy. Sơ đồ phân cấp mục đích



chỉ ra cách đạt mục đích của một người dùng hoàn thiện, tuy nhiên không phải mọi người sử dụng đều có thể theo cách như vậy. Nói chung, khả năng dự đoán hành vi lỗi là rất nghèo nàn trong các mô hình phân cấp, mặc dù một số mô hình (như là CCT) cũng đã xem xét đến vấn đề này.

### 7.6.1. GOMS

Mô hình GOMS do Card, Moran và Newell xây dựng và đặt tên theo các chữ cái đầu tiên của các từ **G**oals, **O**perations, **M**ethods và **S**election. Mô hình GOMS bao gồm 4 thành phần sau:

- **Goals** (mục đích): Là các mục đích của người sử dụng, miêu tả những gì mà người sử dụng muốn đạt được. Ngoài ra, các mục đích trong GOMS còn được sử dụng để biểu diễn “điểm ghi nhớ” của người sử dụng, từ đó người dùng có thể đánh giá những gì nên thực hiện và xem xét điểm mà người dùng có thể quay lui để thực hiện lại khi có bất kì một lỗi nào xảy ra.
- **Operations** (các thao tác): Đây là các mức thấp nhất của quá trình phân tích. Các thao tác là các hành động cơ bản mà người sử dụng phải thực hiện để sử dụng hệ thống. Chúng có thể tác động lên hệ thống (ví dụ, ấn phím “X”), hoặc chỉ đơn giản là tác động đến trạng thái tinh thần của người sử dụng (ví dụ, khi người sử dụng đọc hộp thoại).
- **Methods** (phương thức): Như đã thảo luận ở trên, để phân chia một mục đích thành các mục đích con, chúng ta có một số cách tiếp cận rất tiêu biểu. Ví dụ, để đóng một ứng dụng sử dụng giao diện cửa sổ, ta có thể lựa chọn tùy chọn ‘CLOSE’ trên thanh menu POP - UP, hoặc sử dụng tổ hợp phím ‘Alt + F4’. Trong GOMS các mục đích phân chia được xem như là các phương thức. Do đó trong ví dụ trên, chúng ta sẽ có 2 phương thức đó là: phương thức CLOSE –METHOD và phương thức Alt\_F4- METHOD

#### Ví dụ:

GOAL: ICONIZE – WINDOW

- {select GOAL: USE – CLOSE - METHOD
- . MOVE – MOUSE – TO – WINDOW- HEADER
- . POP-UP – MENU
- . CLICK – OVER –CLOSE –OPTION

GOAL: USE – ALT\_F4 – METHOD

- . . PRESS ALT + F4 – KEY }

Các dấu chấm được dùng để chỉ rõ mức phân cấp của các mục đích

- **Selections (lựa chọn):** Từ ví dụ trên, ta thấy sự xuất hiện của từ khoá ‘select’ dùng để lựa chọn các phương thức. GOMS không cho phép chọn các phương thức một cách ngẫu nhiên mà cố gắng dự đoán các phương thức sẽ được sử dụng. Sự chọn lựa này phụ thuộc vào người sử dụng, vào trạng thái của hệ thống và các chi tiết của mục đích. Ví dụ, một người sử dụng, tên Sam, có thói quen không bao giờ sử dụng phương thức Alt+f4 để đóng cửa sổ, ngoại trừ khi anh ta chơi trò chơi ‘blocks’ bởi vì trò chơi này yêu cầu sử dụng con chuột . GOMS xác định thói quen này như là một quy tắc lựa chọn cho người dùng Sam:

**Ví dụ:**

User Sam:

Rule1: Sử dụng phương thức CLOSE - METHOD trừ phi các quy tắc khác được áp dụng

Rule 2: Nếu ứng dụng là ‘blocks’ thì sử dụng phương thức Alt+f4 – METHOD

Các phân cấp mục đích được mô tả trong một phân tích GOMS thường ở dưới mức nhiệm vụ đơn vị được định nghĩa trước đó. Do đó, một phân tích GOMS điển hình bao gồm một mục đích mức cao duy nhất, sau đó mục đích này sẽ được phân chia tiếp thành một chuỗi các nhiệm vụ đơn vị, và tất cả các nhiệm vụ đơn vị này lại được phân tích tiếp xuống đến mức các thao tác cơ bản.

**Ví dụ:**

GOAL: EDIT – MANUSCRIPT

GOAL: EDIT – UNIT – TASK (lặp cho đến tận khi không còn nhiệm vụ đơn vị)

Sự phân rã mục đích giữa nhiệm vụ tổng thể và các nhiệm vụ đơn vị có thể giúp cho việc hiểu một cách chi tiết về các chiến thuật giải quyết vấn đề của người sử dụng và về phạm vi ứng dụng. Chúng ta có thể sử dụng các kí hiệu chung để miêu tả cấu trúc mục đích con (như trong ví dụ tạo ra báo cáo về tình hình bán lẻ sách đã trình

bày). Đặc biệt, mục đích của phương pháp phân tích nhiệm vụ phân cấp là để tạo ra các phân rã nhiệm vụ, trong đó có thể có các nhiệm vụ tương tự nhau nhưng lại được biểu diễn bằng các kí hiệu khác nhau.

Phân tích cấu trúc mục đích GOMS có thể dùng để đánh giá về hiệu quả thực hiện. Mức độ sâu của cấu trúc mục đích có thể được sử dụng để xác định các yêu cầu về bộ nhớ ngắn hạn.

Mô hình GOMS rất hữu ích trong việc miêu tả cách thức các chuyên gia sử dụng để thực hiện các nhiệm vụ thông thường. Ngoài ra, GOMS, kết hợp với mô hình vật lý và thiết bị, có thể dùng để dự đoán hiệu quả thực hiện của người sử dụng theo thời gian thực hiện nhiệm vụ.

**Bài tập ví dụ:** Xây dựng một mô tả GOMS về nhiệm vụ photocopy một bài báo (article) từ một tạp chí (journal). Thảo luận về điểm dừng trong mô tả GOMS đó.

**Trả lời:** Câu trả lời phụ thuộc vào những giả thiết về máy photocopy được sử dụng trong mô hình. Trong ví dụ dưới, chúng ta giả sử rằng bài báo được copy mỗi lần một trang và quá trình copy được thực hiện thông qua một bộ đệm (khay). Ta thực hiện sao chép nội dung hình ảnh của bản copy đặt vào khay (bản cover), sau đó mới thực hiện copy bản copy thực sự.

GOAL: PHOTOCOPY – PAPER

- . GOAL: LOCATE – ARTICLE
- . GOAL: PHOTOCOPY – PAGE - lặp cho đến tận khi hết các trang
  - . GOAL: ORIENT – PAGE
  - . OPEN – COVER
  - . SELECT – PAGE
  - . POSITION – PAGE
  - . CLOSE – COVER
- . GOAL: VERIFY – COPY
  - . LOCATE – OUT – TRAY
  - . EXAMINE – COPY
- . GOAL: COLLECT – COPY
  - . LOCATE – OUT – TRAY
  - . REMOVE – COPY (đích ngoài đã đạt được)

- . GOAL: RETRIEVE – JOURNAL
- . . OPEN – COVER
- . . REMOVE – JOURNAL
- . . CLOSE - COVER

Vấn đề điểm dừng xảy ra khi bản copy của bài báo bị remove khỏi khay của máy copy, thoả mãn mục đích tổng thể của nhiệm vụ. Tuy nhiên, trong mô tả trên, ta thấy rằng bài báo gốc vẫn nằm trong bộ nhớ của máy copy, khi mà bản cover đã được đóng lại. Người sử dụng sẽ rất dễ quên việc remove tạp chí ra khỏi máy. Vậy thủ tục copy trên có thể được thay đổi như thế nào để giải quyết vấn đề này? Giải pháp là thực hiện đích RETRIEVE – JOURNAL trước khi thực hiện đích COLLECT – COPY.

### 7.6.2. CCT

Lý thuyết độ phức tạp nhận thức (CCT), được giới thiệu bởi Kieras và Polson, bắt đầu từ những tiên đề cơ bản của quá trình phân rã mục đích trong mô hình GOMS, và phát triển lên làm cho nó có khả năng dự đoán tốt hơn. CCT là 2 mô tả song song: về mục đích của người sử dụng và về hệ thống máy tính (còn được gọi là thiết bị trong CCT). Mô tả về các mục đích của người sử dụng được xây dựng dựa theo mô hình phân cấp mục đích giống như GOMS, tuy nhiên nó được mô tả một cách cơ bản hơn bằng cách sử dụng các luật sản xuất (production rules). Đối với mô tả hệ thống, CCT sử dụng các mạng chuyển hoá suy rộng, một dạng của mạng chuyển hoá trạng thái. Chúng ta sẽ tìm hiểu chi tiết về các mạng chuyển hoá trạng thái một cách chi tiết trong Chương 8.

Luật sản xuất là một chuỗi các luật dạng: *If điều kiện then hành động*

Trong đó, điều kiện là một phát biểu về các nội dung của bộ nhớ làm việc. Nếu điều kiện là đúng, thì ta nói rằng luật sản xuất đó được cháy. Một hành động có thể gồm 1 hoặc nhiều hành động cơ bản, có thể làm thay đổi bộ nhớ làm việc hoặc chỉ là các hành động bên ngoài, như là hành động bấm phím. Chương trình luật sản xuất được viết bằng ngôn ngữ tựa LISP.

**Ví dụ:** Xét nhiệm vụ soạn thảo sử dụng một chương trình soạn thảo của UNIX. Nhiệm vụ là chèn dấu cách vào một đoạn text bị thiếu dấu cách giữa các từ. Ví dụ chèn dấu cách vào đoạn text “cognitivecomplexity theory”. Đây là vấn đề sửa lỗi rất thường gặp khi đánh văn bản, vì thế chúng ta giả sử rằng chúng ta đã có sẵn những thủ

tục rất tốt để thực hiện nhiệm vụ này. Xét một đoạn trong mô tả CCT về nhiệm vụ trên như sau:

```
(SELECT – INSERT – SPACE
IF ( AND      ( TEST – GOAL thực hiện nhiệm vụ đơn vị)
              (TEST – TEXT nhiệm vụ là chèn dấu cách)
              (NOT (TEST – GOAL chèn dấu cách))
              (NOT (TEST – NOTE thực hiện chèn dấu cách)) )
THEN (        (ADD – GOAL chèn dấu cách)
              (ADD – NOTE thực hiện chèn dấu cách)
              (LOOK – TEXT nhiệm vụ nằm ở %LINE. %COLUMN) ))

(INSERT – SPACE –DONE
IF (AND      (TEST – GOAL thực hiện nhiệm vụ đơn vị)
              (TEST – NOTE thực hiện chèn dấu cách)
              (NOT (TEST – GOAL chèn dấu cách))
THEN (        (DELETE – NOTE thực hiện chèn dấu cách)
              (DELETE – GOAL thực hiện nhiệm vụ đơn vị)
              (UNBIND %LINE. %COLUMN) ))

(INSERT – SPACE 1
IF (AND      (TEST – GOAL chèn dấu cách)
              (NOT (TEST – GOAL di chuyển con trỏ))
              (NOT (TEST – CURSOR %LINE. %COLUMN)) )
THEN (        (ADD – GOAL di chuyển con trỏ đến %LINE. %COLUMN) ))

(INSERT – SPACE 2
IF (AND      (TEST – GOAL chèn dấu cách)
              (TEST – CURSOR %LINE %COLUMN)
THEN (        (DO – KEYSTROKE ‘I’)
              (DO – KEYSTROKE SPACE)
              (DO – KEYSTROKE ESC)
              (DELETE – GOAL chèn dấu cách) ))
```

Để hiểu cách hoạt động của các luật trên, chúng ta giả sử rằng người sử dụng chỉ vừa mới nhận thấy những lỗi đánh máy của họ và do vậy nội dung của bộ nhớ làm việc (viết tắt là w.m) là:

(GOAL thực hiện nhiệm vụ đơn vị)

(TEXT nhiệm vụ là chèn dấu cách)

(TEXT nhiệm vụ nằm ở 5 23)

(CURSOR 8 7)

Trong đó, TEXT tham chiếu đến đoạn văn bản đang được soạn thảo và CURSOR là vị trí con trỏ xuất chèn của màn hình. Vị trí cần chèn thêm dấu cách nằm ở dòng 5 cột 23, còn vị trí của con trỏ hiện tại nằm ở dòng 8 cột 7.

Xét 4 luật (SELECT – INSERT - SPACE), (INSERT – SPACE - DONE), (INSERT – SPACE 1), (INSERT – SPACE 2), ta thấy rằng chỉ có luật thứ nhất cháy. Điều kiện cho luật (SELECT – INSERT - SPACE) là:

( AND ( TEST – GOAL thực hiện nhiệm vụ đơn vị)

điều kiện đúng vì (GOAL thực hiện nhiệm vụ đơn vị) nằm trong w.m

(TEST – TEXT nhiệm vụ là chèn dấu cách)

đúng vì (TEXT nhiệm vụ là chèn dấu cách) nằm trong w.m

(NOT (TEST – GOAL chèn dấu cách))

đúng vì (GOAL chèn dấu cách) không nằm trong w.m

(NOT (TEST – NOTE thực hiện chèn dấu cách)) )

đúng vì (NOTE thực hiện chèn dấu cách) không nằm trong w.m

Vì vậy, luật thứ nhất cháy, và hành động của nó được thực hiện. Hành động này không có tác động đến bên ngoài, nhưng nó bổ sung thêm thông tin cho bộ nhớ làm việc. Ví dụ, hành động (LOOK – TEXT nhiệm vụ nằm ở % LINE % COLUMN) sẽ tìm kiếm khoản mục và xác định vị trí theo dòng và cột của khoản mục tương ứng.

Nội dung của bộ nhớ làm việc sau khi luật thứ nhất cháy như sau (chú ý rằng thứ tự của các thành phần trong bộ nhớ làm việc là tùy ý, không bắt buộc):

(GOAL thực hiện nhiệm vụ đơn vị)

(TEXT nhiệm vụ là chèn dấu cách)

(TEXT nhiệm vụ nằm ở 5 23)

(NOTE thực hiện chèn dấu cách)

(GOAL chèn dấu cách)

(LINE 5)

(COLUMN 23)

(CURSOR 8 7)

Ta thấy rằng luật (SELECT – INSERT - SPACE), và luật (SELECT – INSERT - DONE) sẽ không cháy bởi vì mục (GOAL chèn dấu cách) sẽ làm cho điều kiện của 2 luật trên sai. Mặt khác, ràng buộc LINE là 5 và COLUMN là 23 sẽ làm cho điều kiện (TEST – CURSOR %LINE %COLUMN) cũng sẽ sai. Do đó, duy nhất chỉ có luật (INSERT – SPACE 1) là có thể cháy.

Sau khi các hành động của luật này được thực hiện, bộ nhớ làm việc sẽ được bổ sung thêm mục (GOAL di chuyển con trỏ đến 5 23). Chúng ta sẽ không xét đến các luật thực hiện việc di chuyển con trỏ trong phạm vi ví dụ này. Giả sử rằng việc di chuyển con trỏ là thành công, và con trỏ đã ở vị trí (5,23), do đó luật (INSERT – SPACE 2) có thể cháy và các hành động sau sẽ được thực hiện: bấm phím 'I', phím Space và phím Esc. Hàng động đó cũng sẽ xoá mục đích “chèn dấu cách” ra khỏi bộ nhớ làm việc vì mục đích này đã được thực hiện xong.

Sau khi mục đích đã được xoá, luật thứ hai (INSERT – SPACE - DONE) sẽ tự cháy hết và dọn dẹp bộ nhớ làm việc. Đặc biệt, nó cũng sẽ giải phóng các biến LINE, và COLUMN, nghĩa là nó sẽ xoá những ràng buộc đối với chúng ra khỏi bộ nhớ làm việc.

Chú ý rằng các luật không cháy theo thứ tự mà chúng được viết. Mặc dù chúng có vẻ giống với các câu lệnh If – then – else trong các ngôn ngữ lập trình chuẩn, nhưng cách làm việc của chúng có sự khác biệt. Tất cả mọi luật đều hoạt động và ở mỗi thời điểm bất cứ luật nào cũng có thể cháy nếu điều kiện của luật là đúng. Một số luật có thể không bao giờ cháy, chẳng hạn, trong ví dụ trên nếu con trỏ đã nằm đúng vị trí cần dịch chuyển đến thì luật thứ 3 sẽ không bao giờ cháy. Hơn thế nữa, các luật giống nhau có thể cháy lặp lại; ví dụ, xét các luật thực hiện việc di chuyển con trỏ, trong đó có luật sau:

(MOVE – UP

IF (AND (TEST – GOAL move-up)

(TEST – CURSOR- BELOW %LINE) )

THEN ( (DO – KEYSTROKE ‘K’) ))

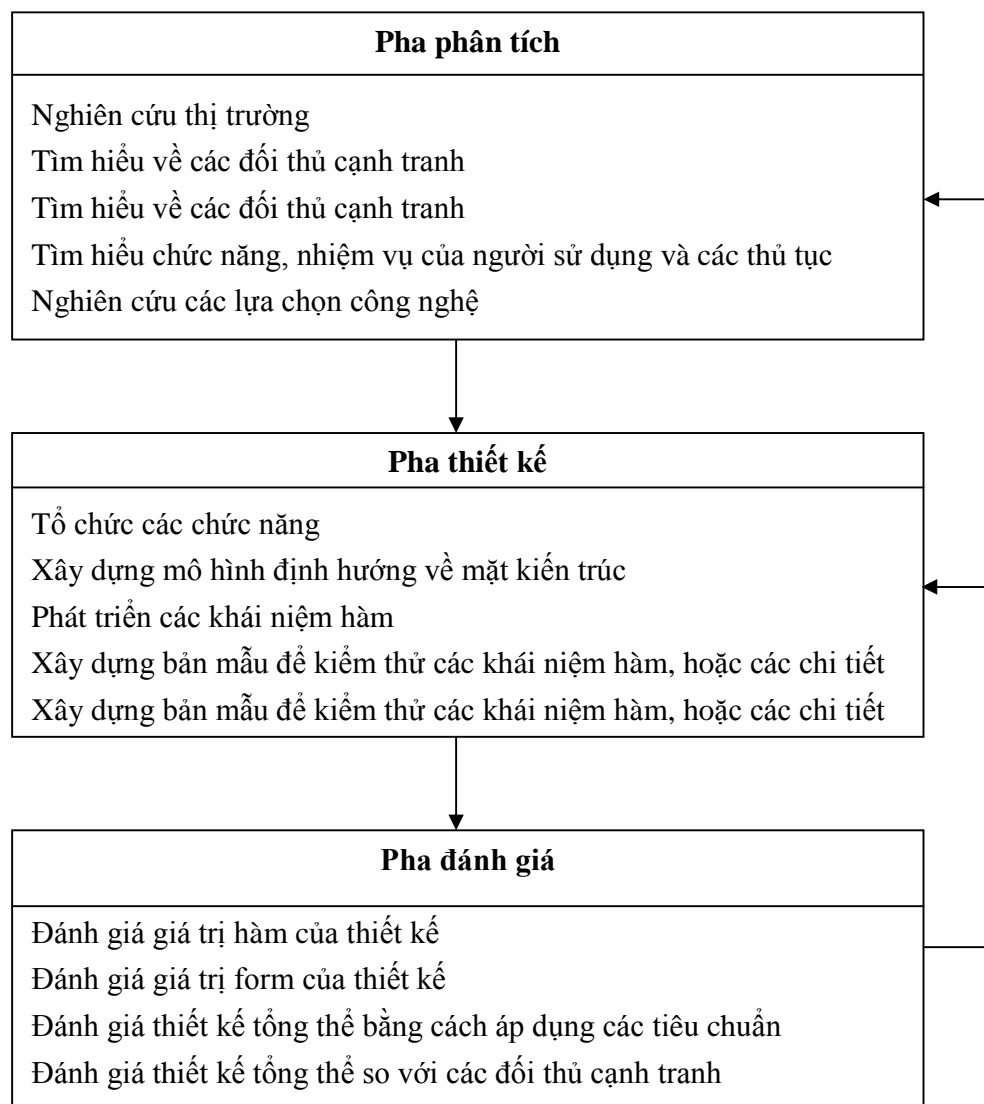
Luật này sẽ thực hiện việc di chuyển con trỏ lên dòng bên trên bằng cách đánh phím ‘K’ nếu như con trỏ vẫn nằm dưới dòng mong muốn. Nghĩa là, luật sẽ liên tục chạy cho đến tận khi con trỏ đã nằm đúng dòng cần đến.



## CHƯƠNG VIII: QUY TRÌNH THIẾT KẾ GIAO DIỆN NGƯỜI DÙNG HƯỚNG NGƯỜI DÙNG

### 8.1. Quy trình thiết kế hướng người dùng

#### a. Sơ đồ tổng quan



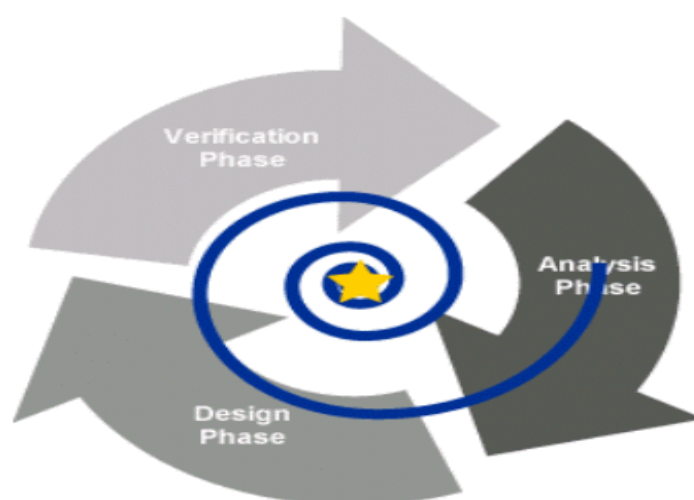
Hình 7.1. Quy trình thiết kế hướng người dùng

Hình vẽ trên là sơ đồ tổng quan quy trình thiết kế hướng người dùng. Quy trình bao gồm 3 pha (giai đoạn): pha phân tích, pha thiết kế, pha kiểm định, trong đó mỗi pha bao gồm nhiều bước thực hiện khác nhau.

**Tính lặp:** Qua sơ đồ ta có thể thấy được bản chất lặp của quy trình thiết kế. Sự lặp lại thiết kế là rất quan trọng trong quy trình thiết kế này. Trong rất nhiều quy trình, người ta thường giả sử rằng các bước thiết kế thường chỉ được thực hiện theo đúng tuần tự từ bước đầu cho đến bước cuối đúng một lần mà không có sự lặp lại. Tuy nhiên quy trình thiết kế giao diện hướng người dùng thì không phải như vậy.

Quy trình này là một quy trình có tính lặp và có vòng đời. Bạn sẽ liên tục phải đánh giá hoặc kiểm thử các giả thuyết, các khái niệm, hoặc các thiết kế với người sử dụng. Bạn thường phải thực hiện mỗi pha của quy trình đó trong 2 hoặc 3 lần.

Khái niệm lặp trong quy trình thiết kế hướng người dùng được minh họa trong sơ đồ ở hình vẽ bên dưới. Tưởng tượng rằng bạn đang hướng đến mục tiêu là sản phẩm cuối cùng (biểu diễn bằng hình ngôi sao trong hình vẽ), khi bạn đang ở giai đoạn giữa của quy trình. Trong quá trình thiết kế lần đầu tiên, bạn đã thu thập được một số thông tin về người dùng, đã tạo được một hoặc 2 khái niệm về tổ chức, đã xây dựng được sơ đồ hàm, và có thể đã thiết kế được một số form. Sau đó, bạn sẽ đánh giá lại những ý tưởng đó với người dùng. Kết quả của lần kiểm thử đó sẽ cung cấp cho bạn được nhiều thông tin hơn nữa. Bạn quyết định bổ sung thêm một số khái niệm mới. Những khái niệm mới đó cần phải được đánh giá lại. Mỗi lần bạn thực hiện xong một vòng, bạn sẽ biết thêm được nhiều thông tin mới về thị trường, các đối thủ cạnh tranh, người dùng và các nhiệm vụ. Thông tin này sẽ cho phép bạn bắt đầu từ con số không để đi đến sản phẩm cuối cùng.



## 8.2. Pha phân tích

Trước khi thiết kế bất cứ một cái gì, việc đầu tiên bạn cần làm là xác định xem bạn đang thiết kế cái gì và thiết kế cho ai, nghĩa là xác định mục đích và đối tượng của thiết kế. Tiếp theo là các bước:

- Nghiên cứu thị trường: Để có sự hiểu biết về thị trường, bạn cần phải nắm được một số thông tin như loại đối tượng khách hàng, phạm vi thị trường và thị trường tiêu thụ mục tiêu.
- Tìm hiểu về các đối thủ cạnh tranh: Để hiểu về đối thủ cạnh tranh, bạn cần phải biết được họ là ai, giá trị về mặt chức năng của sản phẩm mà họ cung cấp cho người sử dụng, những gì người sử dụng thích hoặc không thích đối với sản phẩm, các chức năng còn thiếu sót trong sản phẩm của đối thủ cạnh tranh, và các thị trường mà mình chắc chắn có ưu thế cạnh tranh hơn.
- Tìm hiểu về người sử dụng: Để hiểu được người sử dụng, bạn cần phải nắm được vai trò của họ trong tổ chức, nền tảng (nền tảng giáo dục, kinh nghiệm..), các khả năng tự nhiên (khả năng nhìn, nghe hoặc các dị tật..) của người sử dụng, và các khả năng công nghệ (băng thông, phần cứng, phần mềm) và các chức năng hoặc nhiệm vụ mà người dùng mong muốn thực hiện ở hệ thống.
- Tìm hiểu về hệ thống và các lựa chọn công nghệ: Trong quá trình thiết kế giao diện, bạn cần phải nắm được các công nghệ sẵn có hiện tại để có sự lựa chọn phù hợp, và theo kịp xu hướng phát triển.

Hiện nay có rất nhiều kĩ thuật có thể áp dụng trong quá trình thực hiện phân tích. Tùy theo tổ chức, phạm vi và ứng dụng, nhóm thiết kế có thể chọn lựa kĩ thuật phân tích phù hợp. Các kĩ thuật phân tích được cho trong bảng sau, trong đó, các ô màu trắng là các kĩ thuật thường được lựa chọn và các ô màu xám là các kĩ thuật thường chỉ dùng để suy diễn.

| Các kĩ thuật phân tích  |                                 |                               |
|---|---------------------------------|-------------------------------|
| Đặc tả ngữ cảnh sử dụng   | Phân tích chức năng             |                               |
|   | Phân tích nhu cầu               |                               |
|   | Phân tích cạnh tranh            |                               |
|   | Phân tích ảnh hưởng tài chính   |                               |
|   | Tìm hiểu ngữ cảnh               |                               |
|   | Tìm hiểu phong tục              |                               |
|   | Cách tiếp cận xã hội – kĩ thuật |                               |
|   | Phân tích người sử dụng         | Mô hình vai trò người sử dụng |
|   |                                 | Mô tả người dùng              |
|   |                                 | Mô hình hoạt động             |
|   | Phân tích nhiệm vụ              | Các use-case cần thiết        |
|   |                                 | HTA                           |
| Phân tích nhiệm vụ nhận thức:<br>GOMS, TAG, và mô hình giao diện hành động -đối tượng |                                 |                               |
| Scenarios   |                                 |                               |
| Đặc tả tính dễ dùng   | Dựa trên các nhiệm vụ           |                               |
|   | Chuẩn                           |                               |
|   | Dựa trên các câu hỏi điều tra   |                               |

Bảng 7.1: Bảng tổng kết các kĩ thuật phân tích

### 8.3. Pha thiết kế

Các bước trong pha thiết kế sẽ khác nhau, phụ thuộc vào số lần lặp trong chu trình. Trong các bước lặp đầu tiên, bạn sẽ tập trung vào việc tổ chức các chức năng đã tập hợp trong pha phân tích để tạo ra một mô hình kiến trúc có thể hoạt động được và có thể kiểm soát được. Bạn sẽ hoàn thành mục tiêu này bằng cách xây dựng các bản mẫu phác thảo (bản mẫu trên giấy) để mô phỏng sơ đồ chức năng mức cao, cấu trúc nhiệm vụ mức cao, và có được cái nhìn cơ bản ban đầu về sản phẩm. Các bản mẫu và các lược đồ đó sẽ cho phép bạn đánh giá được những thay đổi và những giả thuyết thiết kế với người sử dụng để đảm bảo được nền tảng cơ bản của giao diện mà bạn thiết kế là có thể đáp ứng được nhu cầu của người sử dụng.

Trong các bước lập sau, bạn sẽ tập trung vào chỉnh sửa các chi tiết về nhận thức và hành vi cụ thể. Những chi tiết đó bao gồm cách thức trình diễn, thứ tự và các chi tiết xuất hiện trên mỗi màn hình trong giao diện người dùng mà bạn đang thiết kế. Bạn sẽ xây dựng các bản mẫu có tính chân thực hơn (như là các đoạn phim, hoặc các mô phỏng), để làm mịn các chi tiết thiết kế và kiểm thử đối với một số tương tác cụ thể để đảm bảo rằng sản phẩm cuối cùng có thể đáp ứng các chức năng và các nhiệm vụ cụ thể đã đặc tả.

#### **8.4. Pha đánh giá**

Trong pha này, ta cần chuẩn bị và thực hiện các kiểm thử về tính khả dùng và báo cáo kết quả kiểm thử cho nhóm thiết kế. Các kiểm thử này được dùng để đánh giá các giả thiết và các khái niệm thiết kế. Chúng còn được sử dụng để tập hợp những thông tin cần thiết. Giống như pha thiết kế, loại và độ phức tạp của các trường hợp kiểm thử là khác nhau, và nó phụ thuộc vào số lần lặp lại trong chu trình phát triển.

Trong những lần lặp đầu tiên, bạn có thể sử dụng các thủ tục kiểm thử đơn giản (danh sách các câu hỏi gửi qua mail, phỏng vấn trực tiếp, tổ chức nhóm để thảo luận, hoặc giám sát quá trình hoạt động thực tế) để thu thập thông tin về thị trường, các đối thủ cạnh tranh, chức năng của hệ thống, người sử dụng và các nhiệm vụ. Ở giai đoạn này, các trường hợp kiểm thử được thiết kế nhằm mục đích thu thập thông tin hoặc kiểm định lại các giả thuyết thiết kế.

Trong các giai đoạn tiếp theo, bạn có thể sử dụng các thủ tục kiểm thử phức tạp hơn (như là cùng học để khám phá, hoặc dùng các trường hợp kiểm thử theo chuẩn), để quan sát người sử dụng thực hiện các nhiệm vụ khác nhau để xác định xem liệu rằng họ có thực sự biết cách sử dụng giao diện mà bạn thiết kế hay không? Ở giai đoạn này, các trường hợp kiểm thử được thiết kế nhằm mục đích để kiểm định lại các quyết định thiết kế.

#### **8.5. Những ưu điểm của quy trình phát triển hướng người dùng**

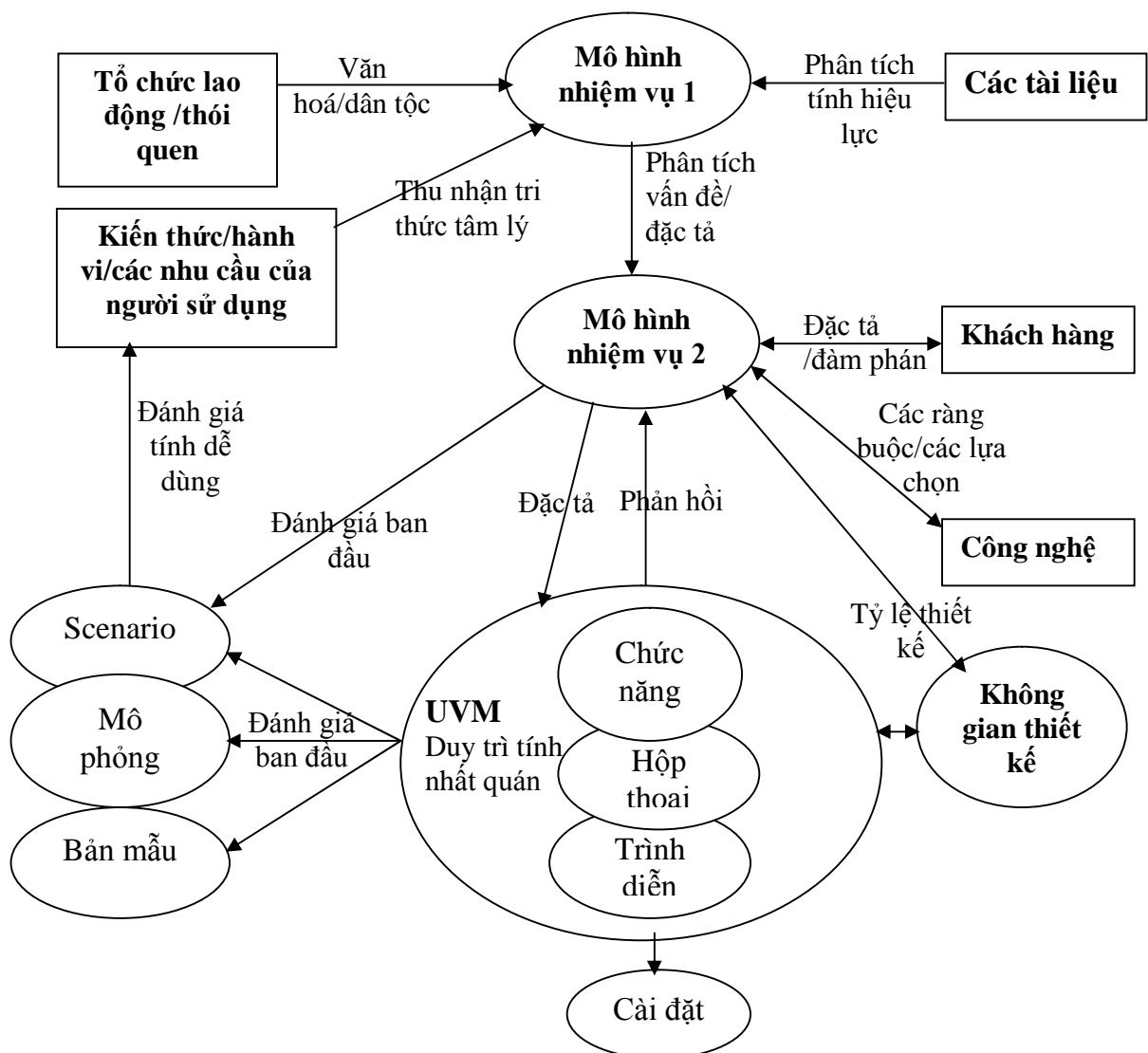
- Làm tăng tính hiệu quả: Sản phẩm dễ sử dụng cho phép người dùng thuận tiện hơn trong công việc, mang lại hiệu quả hoạt động chung cao hơn cho tổ chức.

- Làm giảm chi phí sản xuất: Vì nhóm thiết kế thu nhận yêu cầu từ người dùng ở tất cả các giai đoạn trong quá trình phát triển, do đó tổng chi phí sẽ giảm xuống do những cập nhật về yêu cầu ở các giai đoạn sau phân tích sẽ giảm xuống tới mức tối thiểu và chi phí hỗ trợ cũng sẽ giảm.
- Giảm được các lỗi: Khi sản phẩm không sử dụng được, thì càng có cơ hội làm gia tăng số lượng lỗi. Mặt khác, càng nhiều lỗi thì sẽ làm giảm hiệu quả và gia tăng chi phí trong kinh doanh.
- Giảm chi phí đào tạo: Khi sản phẩm dễ dùng, người sử dụng học cách tương tác với sản phẩm sẽ học nhanh hơn. Và khi sản phẩm khó dùng thì cần phải mất thời gian đào tạo, do đó sẽ làm tăng chi phí trong kinh doanh.
- Nâng cao chất lượng sản phẩm: các sản phẩm dễ dùng thường được xem như là các sản phẩm có chất lượng tốt hơn. Người sử dụng cảm thấy thoải mái với các sản phẩm dễ dùng, và do đó sẽ có thể trở thành khách hàng tiềm năng. Trong quá trình phát triển, các trường hợp kiểm thử về tính dễ dùng cũng được thiết kế để làm tăng tính hiệu quả, cải thiện hiệu năng và thoả mãn người sử dụng.
- Làm tăng độ tin cậy của khách hàng: các sản phẩm dễ dùng sẽ thoả mãn yêu cầu của khách hàng cao hơn, khách hàng tin cậy vào sản phẩm hơn, do đó lượng tiêu thụ sẽ cao hơn. Kết quả là doanh thu kinh doanh sẽ cao hơn.

## CHƯƠNG IX: QUY TRÌNH THIẾT KẾ GIAO DIỆN NGƯỜI DÙNG HƯỚNG NHIỆM VỤ

### 9.1. Quy trình thiết kế hướng nhiệm vụ

Quy trình này được xây dựng dựa theo các nhiệm vụ cụ thể mà người sử dụng muốn thực hiện với hệ thống. Những nhiệm vụ được lựa chọn ban đầu trong quá trình thiết kế sau đó sẽ được sử dụng để xây dựng các vấn đề khác, để trợ giúp cho các quyết định thiết kế, và để đánh giá thiết kế khi thiết kế được phát triển



Hình 8.1: Quy trình phát triển hướng nhiệm vụ

Hình vẽ 8.1 minh họa một cách tổng quan về toàn bộ quy trình thiết kế bao gồm tất cả các hoạt động và các nguồn thông tin.

Quy trình thiết kế bao gồm 4 hoạt động chính:

- Phân tích tình huống nhiệm vụ “hiện tại”
- Xây dựng các nhiệm vụ tương lai mà hệ thống cần phải thực hiện
- Đặc tả công nghệ dùng trong thiết kế
- Các hoạt động đánh giá và kiểm thử (được thực hiện song song với các hoạt động trên)

Các hoạt động này sẽ được trình bày chi tiết trong các phần tiếp theo của chương.

### **9.1.1. Xác định mục đích sử dụng hệ thống của người sử dụng**

Thuật ngữ trong công nghiệp phần mềm dùng để miêu tả bước phát triển này được gọi là “phân tích người dùng và nhiệm vụ”. Ta thấy rằng sự cần thiết của bước phân tích nhiệm vụ là rất rõ ràng: nếu bạn xây dựng ra một hệ thống khác nhưng không phải là hệ thống mà bạn cần phải thiết kế, thì điều đó hoàn toàn sai nhiệm vụ. Bên cạnh yêu cầu thực hiện đúng nhiệm vụ, một hệ thống được gọi là thành công khi nó hoạt động một cách trơn tru trong thế giới và công việc của người sử dụng. Hệ thống nên cung cấp thông tin theo trật tự mà người sử dụng thích và cung cấp cơ chế cho người dùng sửa sai khi họ mắc lỗi. Các thiết bị phần cứng nên phù hợp với không gian và tạo cảm giác hài hoà. Thông thường, trong quá trình phân tích yêu cầu thì tất cả những điều trên và rất nhiều các khía cạnh khác của giao diện người dùng thường bị bỏ sót, tuy rằng chúng có thể được xem xét một khi người thiết kế quan tâm phân tích đến các chi tiết của nhiệm vụ mà người dùng thực sự thực hiện.

Việc hiểu về người sử dụng là rất quan trọng. Nếu nhận thức được về tri thức nền tảng của người sử dụng thì sẽ giúp ích cho người thiết kế rất nhiều trong việc giải quyết các vấn đề như là nên đặt tên như thế nào cho các hệ thống menu, nên đưa những gì vào trong các gói phần mềm và các file trợ giúp và thậm chí là cả những đặc điểm mà hệ thống nên cung cấp. Ví dụ, một hệ thống được thiết kế cho người dùng máy Macintosh, thì nên tính đến những đặc điểm chung của máy Mac mà người sử dụng đã quen. Điều này có nghĩa hệ thống sẽ có thể bao gồm một đặc điểm giống như Cut hoặc Paste mặc dù rằng Cut và Paste thực ra không đóng vai trò gì trong chức



năng chính của hệ thống. Sự chênh lệch về định lượng thấp giữa những người sử dụng, như là niềm tin, mức độ quan tâm đến việc học hệ thống mới, hoặc sự ủng hộ của họ cho thành công của thiết kế, có ảnh hưởng đến các quyết định như là quyết định về lượng thông tin phản hồi mà hệ thống nên cung cấp, hoặc là khi nào thì nên sử dụng giao diện đánh lệnh từ bàn phím thay vì dùng các menu trên màn hình.

Để có thể phân tích nhiệm vụ và người sử dụng một cách có hiệu quả đòi hỏi sự tiếp xúc gần gũi giữa cá nhân các thành viên trong nhóm thiết kế và người sử dụng thực sự của hệ thống. Và chính sự tiếp xúc liên tục ngay từ đầu này là điều rất cần thiết cho một thiết kế tốt.

### **9.1.2. Lựa chọn các nhiệm vụ**

Theo cách tiếp cận của quy trình thiết kế truyền thống thì sau khi đã có được những hiểu biết tốt về người sử dụng và nhiệm vụ của họ, bước tiếp theo sẽ là đưa ra một đặc tả chung về hệ thống và giao diện người dùng của hệ thống. Tuy nhiên, quy trình thiết kế hướng nhiệm vụ lại sử dụng một cách tiếp cận mang tính cụ thể hơn. Người thiết kế nhận dạng một số nhiệm vụ cần thực hiện. Những nhiệm vụ đó là những nhiệm vụ mà người sử dụng đã miêu tả với người thiết kế. Ban đầu, những nhiệm vụ đó có thể được phát biểu dưới dạng một số từ hoặc lời nói, nhưng bởi vì chúng là những nhiệm vụ thực, do đó sau này, chúng có thể được phát triển đến bất kỳ mức chi tiết nào để trả lời đối với các câu hỏi thiết kế hoặc phân tích một giao diện đã cho sẵn. Sau đây là một số ví dụ về nhiệm vụ được nhận biết thông qua lời nói:

- Hệ thống xử lý Word: “Sao chép một số ghi nhớ và gửi nó đến danh sách thư cần gửi”
- Hệ thống bảng tính: “Xây dựng một quỹ lương cho năm tới”
- Một chương trình truyền thông: “truy nhập vào văn phòng thông qua modem”
- Một hệ điều khiển công nghiệp: “điều khiển chuyển đến tốc độ tiếp theo”

Ta thấy rằng, chính những nhiệm vụ được nhận dạng ban đầu đó là những nhiệm vụ thật sự mà người sử dụng phải đối mặt, và nhóm thiết kế nên các thu thập tài liệu để thực hiện những nhiệm vụ đó: sao chép số ghi nhớ, liệt kê lương trong năm hiện tại và các hệ số lương có thể phải thay đổi, xem xét lại.

Những nhiệm vụ được lựa chọn nên bao trùm đầy đủ các chức năng của hệ thống và người thiết kế nên liệt kê ra danh sách các chức năng và so sánh những chức năng

đó với nhiệm vụ để đảm bảo rằng sự bao trùm đầy đủ. Chúng ta cũng nên trộn lẫn những nhiệm vụ phức tạp với những nhiệm vụ đơn giản. Những nhiệm vụ đơn giản, như là kiểm tra chính tả của từ “ocasional” sẽ rất có ích cho những quyết định thiết kế ban đầu, tuy nhiên cũng có nhiều vấn đề giao diện phức tạp chỉ có thể được tiết lộ thông qua những nhiệm vụ phức tạp biểu diễn các tương tác mở rộng trong thế giới thực. Do đó, việc xây dựng được một tập các nhiệm vụ phù hợp chính là bằng chứng tốt nhất chứng minh sự hiểu biết của người thiết kế về người sử dụng và công việc của họ.

### **9.1.3. Sao chép**

Ở đây chúng ta không có ý “sao chép” là hợp pháp và cũng không khuyên người thiết kế đi sao chép các thiết kế của người khác. Tuy nhiên, trong quá trình thiết kế bạn nên đi tìm những giao diện đã có sẵn và sau đó xây dựng ý tưởng giao diện cho hệ thống của mình từ những giao diện có sẵn đó. Hình thức sao chép này rất có hiệu quả cho các sơ đồ tương tác mức cao và điều khiển mức thấp/ các quyết định trình bày.

Ở những mức cao, việc sao chép nhằm giúp cho người sử dụng dễ dùng bởi vì những đặc điểm được sao chép thường là những đặc điểm mà người sử dụng đã quen dùng ở giao diện của các hệ thống khác. Chúng ta cũng có thể tạo ra một sơ đồ tương tác mới thích hợp hơn cho thiết kế của mình, nhưng nguy cơ thất bại là rất cao. Bởi vì những thay đổi sẽ làm cho người sử dụng mất thời gian để làm quen và do đó không đảm bảo được tính dễ dùng. Ví dụ, tất cả chúng ta đều đã quen với các công cụ copy, cut, paste thì trong thiết kế giao diện của mình ta cũng nên sao chép lại những nhiệm vụ này.

Sao chép các sơ đồ đã có sẵn cũng có hiệu quả đối với các chi tiết mức thấp của giao diện, như là vị trí nút hoặc tên menu. Ví dụ như: Bạn đang viết một ứng dụng quản lý form có mục đích cụ thể và đặc tả cần phải có một bộ kiểm tra chính tả. Khi đó bạn nên xem các điều khiển của bộ kiểm tra chính tả có trong các bộ xử lý Word đã rất quen thuộc đối với người sử dụng và nên thiết kế giao diện bộ kiểm tra chính tả của bạn tương tự như vậy.

Đây cũng là vấn đề mà rất nhiều người thiết kế thường có những quyết định sai lầm nếu như họ không thực sự hiểu sâu về những yêu cầu mà hệ thống do họ thiết kế đặt ra. Chúng ta sẽ khai thác sâu hơn một chút về ví dụ bộ kiểm tra chính tả ở trên.

Giả sử, từ những phân tích cá nhân bạn nghĩ rằng bộ kiểm tra chính tả thường chỉ nhặt ra các tên bị sai chính tả, và bạn có thể tự động sửa lại những tên đó bằng cách sử dụng một cơ sở dữ liệu khách hàng. Vì thế bạn nghĩ là tương tác hiệu quả nhất nên liệt kê luôn các tên đúng chính tả và để cho người dùng chấp nhận những chỉnh sửa bằng cách ấn vào phím Return. Tuy nhiên, người sử dụng bộ xử lý từ của bạn lại thường làm việc theo một quy ước khác: họ ấn phím Return để ghi lại lỗi chính tả của một từ. Bạn nên theo hệ thống đã có sẵn (“sao chép”), hay là bạn tự tạo ra cho mình một quy ước hiệu quả hơn? Câu trả lời phụ thuộc vào sự so sánh giữa mức độ sử dụng hệ thống của bạn với mức độ sử dụng hệ thống mà họ đã quen của người dùng. Nếu như họ không sử dụng hệ thống của bạn thường xuyên thì câu trả lời là nên sử dụng những cái gì mà họ đã quen và ngược lại.

## **9.2. Phân tích nhiệm vụ**

### **9.2.1. Định nghĩa**

Phân tích nhiệm vụ là quá trình thu thập dữ liệu về những nhiệm vụ mà người sử dụng thực hiện và phân tích chúng để có được sự hiểu biết sâu về các nhiệm vụ đó.

### **9.2.2. Các phương pháp và các kỹ thuật phân tích**

Hiện tại, có rất nhiều phương pháp và kỹ thuật phân tích đang được sử dụng. Tuy nhiên, trong phạm vi tài liệu này, chúng ta chỉ tập trung thảo luận 2 phương pháp và kỹ thuật có ảnh hưởng lớn nhất, bao gồm:

#### **9.2.2.1. Phân tích nhiệm vụ phân cấp (HTA)**

HTA là một quá trình phát triển một mô tả về các nhiệm vụ của hệ thống dưới dạng các thao tác và các kế hoạch. Các thao tác là những hoạt động mà người sử dụng thực hiện để đạt được một mục đích nào đó. Và các kế hoạch là các ràng buộc điều kiện về thời điểm thao tác được thực hiện. Các thao tác có thể được phân rã và với mỗi nhiệm vụ con mới sẽ tồn tại một kế hoạch mới kèm theo.

Một số loại kế hoạch thường gặp:

- Thứ tự cố định: Các nhiệm vụ con được thực hiện theo đúng thứ tự mà chúng đã được phân rã (ví dụ kế hoạch 3 trong ví dụ dưới)
- Các nhiệm vụ lựa chọn: Việc thực hiện hay không thực hiện nhiệm vụ đó phụ thuộc vào từng tình huống cụ thể .

- Chờ đợi các sự kiện: Nhiệm vụ chỉ được thực hiện khi một nhiệm vụ liên quan đến phải hoàn thành trước.
- Chia sẻ thời gian: Các nhiệm vụ được thực hiện một cách đồng thời.
- Nhiệm vụ không bắt buộc: Người sử dụng được phép tùy ý lựa chọn việc thực hiện hay không thực hiện nhiệm vụ không bắt buộc mà vẫn đảm bảo hoàn thành được mục đích.

**Ví dụ minh họa:** Phân tích nhiệm vụ “pha trà”.

Nhiệm vụ pha trà có thể phân rã thành 6 nhiệm vụ con, bao gồm các nhiệm vụ sau:

0. Pha trà

1. Đun sôi nước
2. Tráng ấm pha trà
3. Pha trà
4. Đợi 4 hoặc 5 phút
5. Rót trà

Plan 0

Do 1  
đồng thời xem, nếu ấm pha trà còn trà cũ (đầy) then 2  
then 3-4  
sau 4 hoặc 5 phút Do 6

1. Đun sôi nước

- 1.1 Cho nước vào siêu
- 1.2 Đặt siêu lên bếp
- 1.3 Đợi nước sôi
- 1.4 Tắt bếp

Plan 1

1.1-1.2-1.3  
khi siêu đun nước sôi 1.4

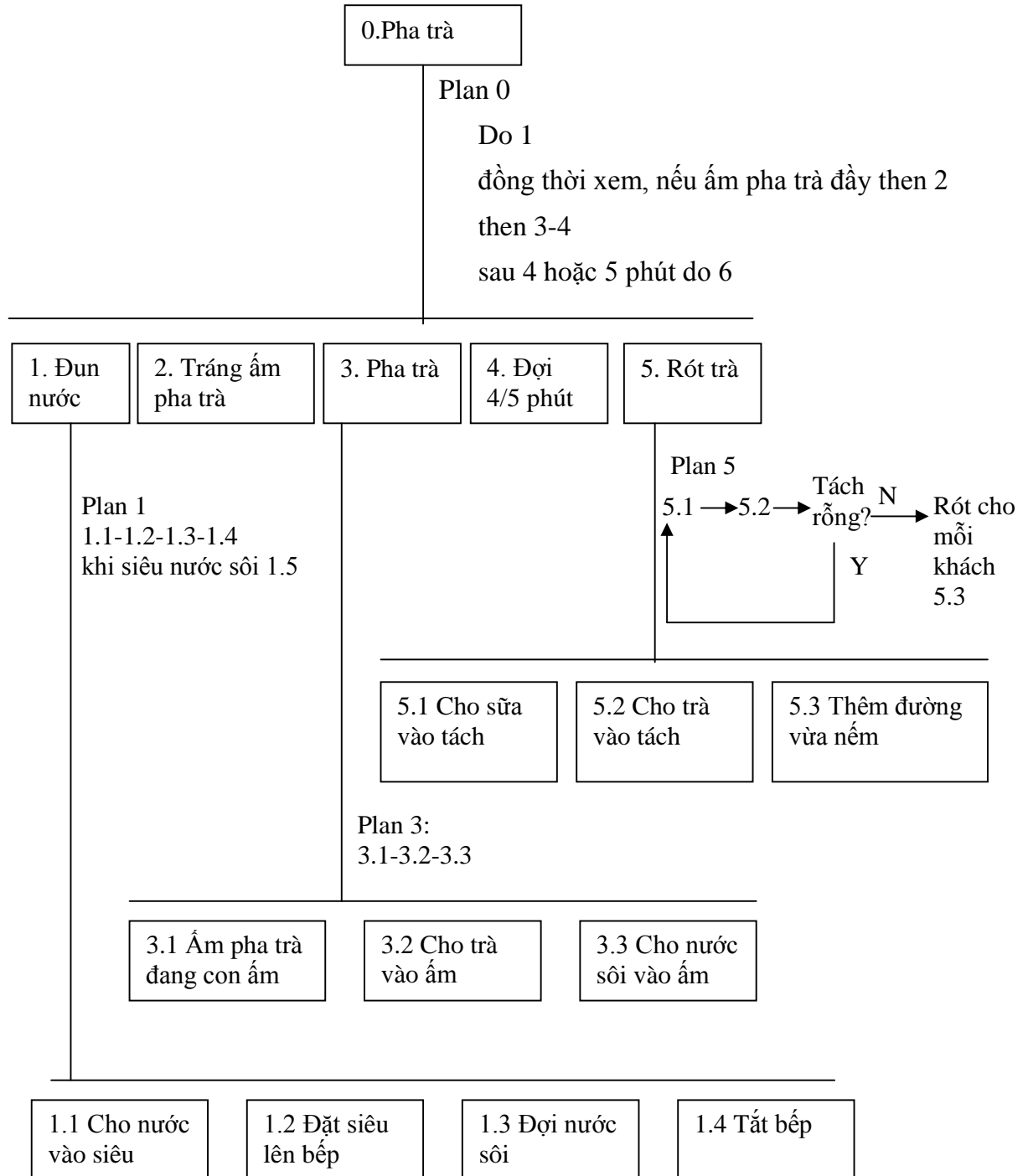
3. Pha trà

- 3.1. Ấm pha trà đang còn ấm (chú ý không pha trà khi ấm nguội)
- 3.2 Cho trà vào ấm
- 3.3 Cho nước sôi vào

5. Rót trà

- 5.1 Cho sữa vào tách
- 5.2 Cho trà vào tách
- 5.3 Thêm đường vừa nếm

Từ những kết quả phân tích trên, chúng ta có sơ đồ phân cấp nhiệm vụ của nhiệm vụ “pha trà” được biểu diễn như sau:



Hình 8.2: Ví dụ về sơ đồ phân cấp nhiệm vụ

### 9.2.2.2. GMOS

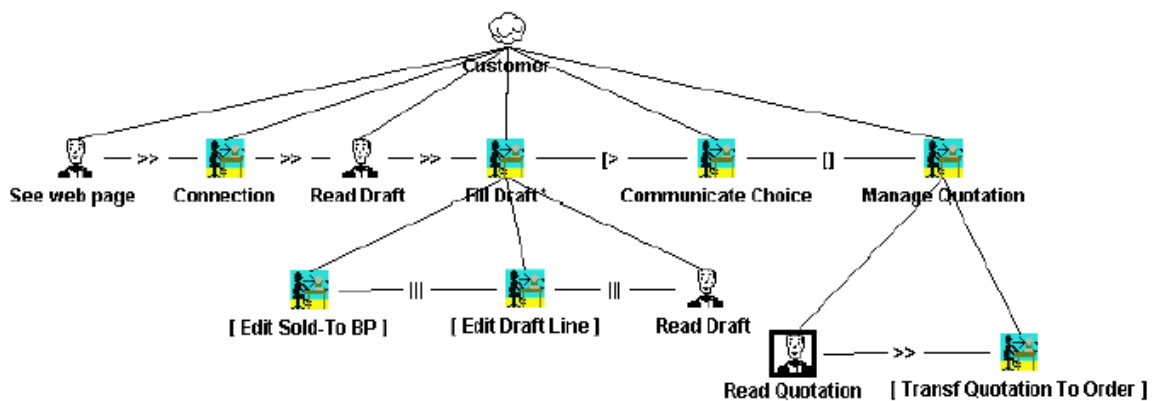
Như chúng ta đã biết, về bản chất, GOMS không thực sự là một kỹ thuật phân tích nhiệm vụ mà chỉ là một kỹ thuật miêu tả hộp thoại dựa trên nhiệm vụ. Nó miêu tả các bước đối thoại cần dùng để thực hiện một nhiệm vụ trên một giao diện người dùng cụ thể. Trong chương này chúng ta sẽ không đề cập lại kỹ thuật GOMS mà chỉ liệt kê nó như một trong những kỹ thuật phân tích bởi vì trong chương 6 chúng ta đã phân tích kỹ lưỡng về kỹ thuật này.

### 9.3. Mô hình hoá nhiệm vụ

Mô hình hoá nhiệm vụ là hoạt động biến đổi các nhiệm vụ thô và dữ liệu liên quan đến người sử dụng hoặc các ý tưởng thành các đơn vị có cấu trúc của tri thức nhiệm vụ. Tri thức này sẽ được ghi lại trong một tài liệu đặc tả bằng cách sử dụng phương pháp biểu diễn. Hiện nay, có rất nhiều phương pháp biểu diễn khác nhau đang được sử dụng, trong đó phương pháp sử dụng UML là một phương pháp có ảnh hưởng và có hiệu quả nhất.

#### 9.3.1. Mô hình nhiệm vụ truyền thống

Mô hình nhiệm vụ truyền thống chủ yếu sử dụng cấu trúc cây nhiệm vụ. Cây nhiệm vụ chỉ rõ cấu trúc công việc dưới dạng các nhiệm vụ, mục đích và các hành động.

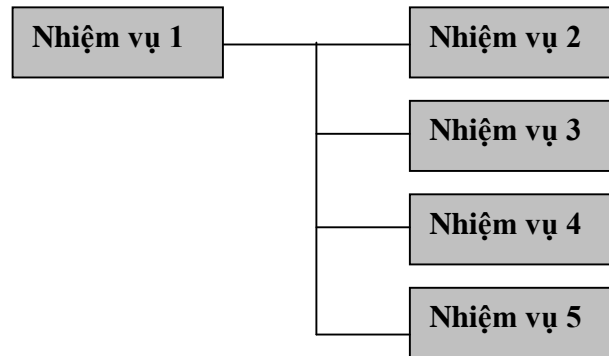


Hình 8.3: Ví dụ về một cây nhiệm vụ

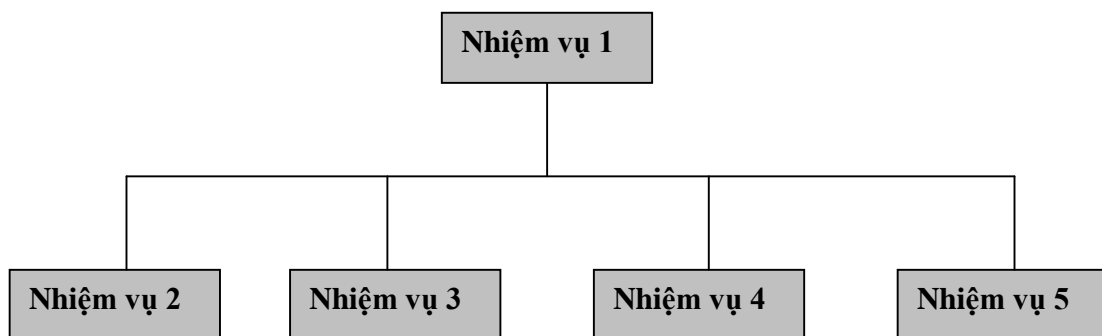
Cây nhiệm vụ rất dễ hiểu và dễ xây dựng, chúng có thể được xây dựng mà không cần sự trợ giúp của các công cụ phần mềm. Cây nhiệm vụ được chủ yếu dựa

trên các mối quan hệ nhiệm vụ con giữa các nhiệm vụ, do đó nó không cung cấp thông tin về các vai trò, các đối tượng...

Có 2 cách để mô tả cây: mô tả từ trái qua phải và mô tả từ trên xuống dưới (hình 8.4 và 8.5):



Hình 8.4: Mô tả từ trái qua phải



Hình 8.5: Mô tả từ trên xuống dưới

## 9.3.2. UML

### 9.3.2.1. Mục đích

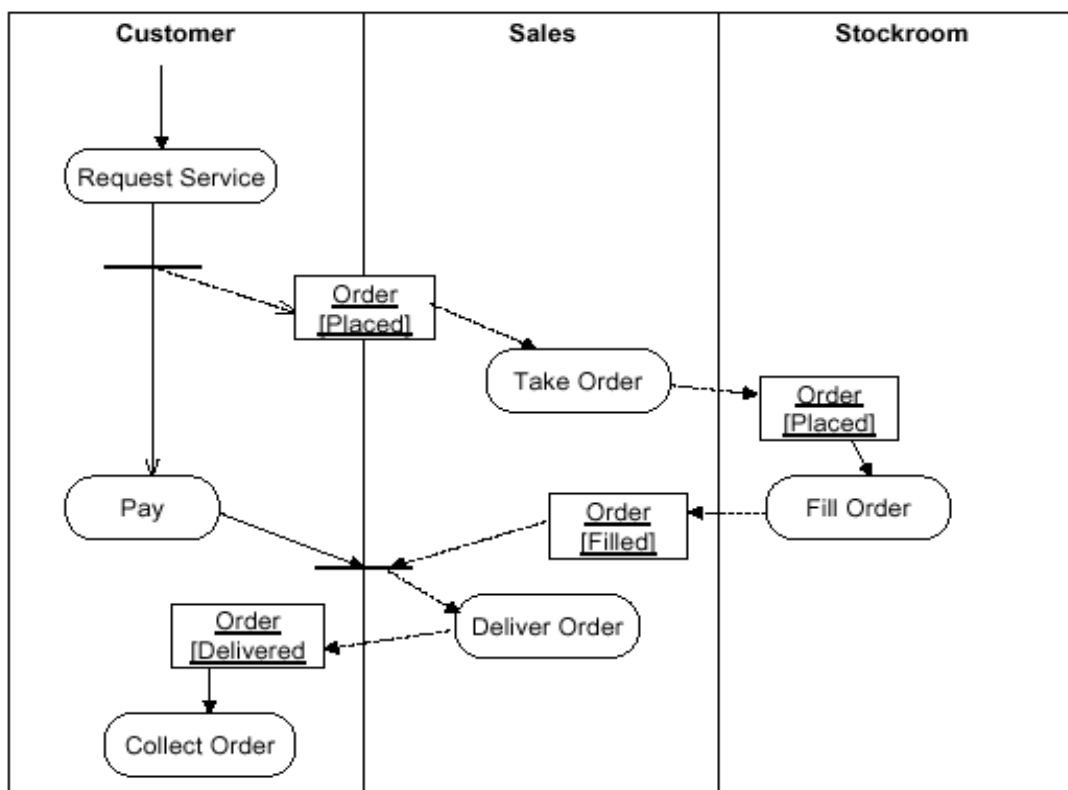
Trong phân tích nhiệm vụ, mục đích của các phương pháp biểu diễn bao gồm như sau:

- Viết tài liệu và giúp cho việc trao đổi kiến thức giữa các nhà thiết kế
- Phân tích công việc và tìm ra những công việc ứ đọng lại và các cơ hội
- Tổ chức lại các ý tưởng của các cá nhân trong nhóm thiết kế
- Thảo luận các khía cạnh của thế giới nhiệm vụ trong nhóm thiết kế
- Đề nghị các thay đổi hoặc các bổ xung trong nhóm thiết kế
- Lựa chọn các giải pháp thay thế trong nhóm thiết kế hoặc với khách hàng

### 9.3.2.2. Các loại sơ đồ trong UML

Mặc dù thực tế UML không được thiết kế nhằm mục đích để mô hình hoá nhiệm vụ, nhưng một số sơ đồ của nó lại khá hữu dụng trong ngữ cảnh phân tích nhiệm vụ. Những sơ đồ đó bao gồm:

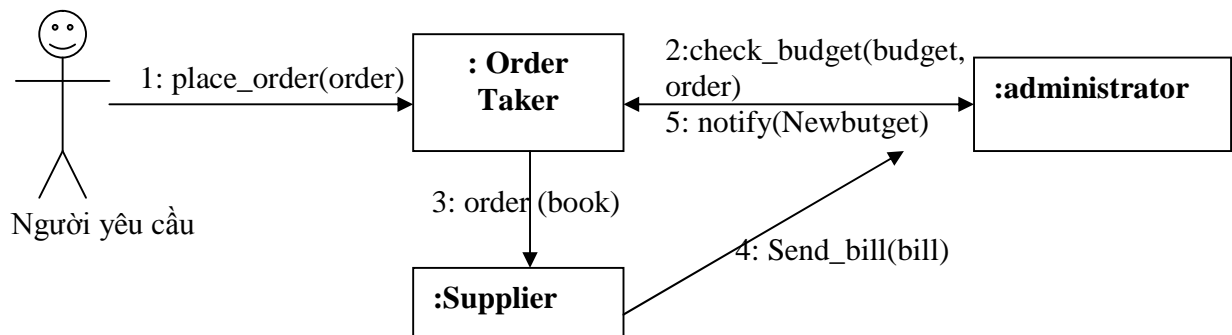
- **Sơ đồ hoạt động:** Được sử dụng để mô tả luồng nhiệm vụ trong mối quan hệ với các sự kiện (event), các vai trò (role) và các mục đích (goal). Một hoạt động được khởi tạo bằng một sự kiện. Ngay sau khi nhiệm vụ được bắt đầu, kết quả là một mục đích sẽ được thực hiện.



Hình 7.3: Ví dụ về sơ đồ hoạt động của UML

- **Sơ đồ hợp tác:** Miêu tả cách thức hợp tác của các đối tượng khác nhau. Kí hiệu mũi tên chỉ sự liên lạc hoặc hợp tác giữa các role trong khi trao đổi các đối tượng hoặc các thông điệp.





Hình 7.4: Sơ đồ hợp tác của UML

- **Sơ đồ tuần tự:** Sơ đồ tuần tự biểu diễn thứ tự thực hiện của các nhiệm vụ.
- **Sơ đồ use-case**

### 9.3.3. Mô hình ngữ cảnh

Mô hình ngữ cảnh là một phần trong phương pháp thiết kế ngữ cảnh. Nó bao gồm 5 mô hình công việc để mô tả thế giới nhiệm vụ:

- **Mô hình luồng:** biểu diễn sự liên lạc và hợp tác cần thiết trong quá trình thực hiện công việc
- **Mô hình tuần tự:** Chỉ rõ các bước công việc cần thiết để đạt được một ý định cụ thể
- **Mô hình mẫu tự tạo:** Mô tả các vật được tạo ra để hỗ trợ cho công việc.
- **Mô hình văn hoá:** biểu diễn các phụ thuộc trong công việc do chính sách, văn hoá, ..gây ra
- **Mô hình vật lý:** Chỉ rõ cấu trúc vật lý của môi trường làm việc có tác động đến công việc.