

Week 11

Evaluation Methods 2:

Analytical Evaluations

HCI이론 및 실습 2020 Spring

Human-Computer Interaction+Design Lab _ Joonhwan Lee

Cognitive Models

Cognitive Modeling

- ✦ 사용자들이 특정 시스템에서 어떻게 상호작용하게 될지를 예측하기 위한 모델링 방법
 - ✦ 시스템을 실제로 구현하지 않은 상태에서 사용자가 과제를 실행할 때 얼마나 시간이 걸릴지를 예측하는데에 사용
 - ✦ 인터페이스를 정량화하여 분석하는 방법
 - ✦ S. Card, T. Morgan, A. Newell 등이 키 입력수준 (Keystroke Level) GOMS 모델이라는 분석적 모델을 제안하였다
 - ✦ 예: 키보드로 창을 닫는데 까지 걸리는 시간

인터페이스의 정량분석: GOMS

- ✦ GOMS(Model of Goals, Operators, Methods, and Selective rules)
 - ✦ Card, Morgan and Newell (1983), “The Psychology of Human Computer Interaction”
 - ✦ *Motor Movement* — 인터페이스 타이밍
 - ✦ GOMS 모델 개발 당시, 사용자들이 컴퓨터 시스템을 이용해서 과업을 수행하는데 걸리는 시간이 과업 수행에 필요한 일련의 연속적인 조작 시간의 총합과 같다는 사실을 발견.
 - ✦ 사용자가 특정 작업을 수행할 때 걸리는 개인별 소요 시간은 서로 다를 수 있지만, 평균적인 값을 사용해서 전체 소요 시간을 계산해 낼 수 있다.
 - ✦ *K, P, H, M, R* 등으로 표현한다.

KLM-GOMS

- ♦ K: Keying (키보드 입력 시간) — 0.2초
 - ♦ 키보드에서 키를 누르는데 걸리는 시간. 최근의 연구에 의하면 keystroke는 기술 수준과 과제에 따라 영향을 받는데, 그 범위를 70msec ~ 220msec 정도로 보고, 주로 0.2 초로 계산.
- ♦ P: Pointing (마우스 이동시간) — 1.1초
 - ♦ 화면상에서 특정한 지점을 가리키는데 (pointing) 걸리는 시간. 평균 1100msec 정도가 소요.
- ♦ H: Homing — 0.4초
 - ♦ 사용자의 손을 키보드에서 GID 로 혹은 GID에서 키보드로 옮기는데 걸리는 시간.

KLM-GOMS

- ✦ B: Mouse Button Click — 0.1초
 - ✦ 마우스 버튼을 누르는데 소요되는 시간
- ✦ M: Mentally Preparing — 1.35초
 - ✦ 정신적 준비 시간. 사용자가 다음 단계로 이행하기 위해 정신적으로 준비하는데 걸리는 시간.
- ✦ R: Responding
 - ✦ 사용자가 입력 후 컴퓨터의 반응을 기다리는 시간

KLM-GOMS

- ✦ 손을 GID로 이동한다: H
- ✦ 원하는 라디오버튼을 가리킨다: HP
- ✦ 라디오 버튼을 클릭한다: HPB
- ✦ 손을 키보드로 다시 옮긴다: HPBH
- ✦ 문자 4개를 입력한다: HPBHKKKK

→ $0.4 + 1.1 + 0.1 + 0.4 + 0.2 + 0.2 + 0.2 + 0.2$

- ✦ Mental Operators의 사용
 - ✦ $HPB + \underline{M} + HKKKK$

KLM-GOMS Example

✦ **GOAL: ICONISE-WINDOW**

```
[select
  GOAL: USE-CLOSE-METHOD
    . MOVE-MOUSE-TO-FILE-MENU
    . PULL-DOWN-FILE-MENU
    . CLICK-OVER-CLOSE-OPTION

  GOAL: USE-CTRL-W-METHOD
    PRESS-CONTROL-W-KEY
]
```

KLM-GOMS Example

- ✦ compare alternatives:
 - ✦ **USE-CTRL-W-METHOD** vs. **USE-CLOSE-METHOD**
(Windows OS)

USE-CTRL-W-METHOD		USE-CLOSE-METHOD	
H[to kbd]	0.40	P[to menu]	1.1
M	1.35	B[LEFT down]	0.1
K[ctrlW key]	0.28	M	1.35
		P[to option]	1.1
		B[LEFT up]	0.1
Total	2.03 s	Total	3.75 s

Limitation of the GOMS

- ♦ 전문가 수준에서만 사용 가능 → 초보자나 중급자 수준에서는 적합하지 못하다
- ♦ 모든 모델은 과업 수행 중에 에러가 없다는 가정에서만 산출 가능 → 에러가 빈번한 실제 모델의 수행에서는 적용되기 어렵다. 하지만, 에러를 발견하는 방법론은 다른 것들이 있기 때문에 병행해서 테스트 가능
- ♦ 모든 과업 요소들이 연속적이라는 가정안에서만 적용 가능

Limitation of the GOMS

- ✦ 기능성에 대해서는 아무런 대안도 제시하지 못한다. 시스템에 있어서 과제에 대한 사용성만을 언급 (키입력 vs. 마우스 입력)
- ✦ 속도 상의 미세한 차이가 경제적/심리적으로 아주 상이한 결과를 유도할 가능성이 있다 → 평균에 근거한 KLM 은 적용 불가능 → CPM-GOMS 등과 같이 보다 정밀한 모델링 기법을 사용하면 오차 범위를 5% 이내로 예측 가능

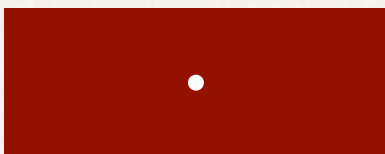
Workshop: KLM-GOMS

- ✦ Task: Move Text (Cut Text and then Paste Text)
 - ✦ 텍스트를 이동 (cut & paste) 하는데 걸리는 시간 계산

GOMS 계산 예

GOAL: MOVE-TEXT	
. GOAL: CUT-TEXT	
. . GOAL: HIGHLIGHT-TEXT	
. . . [select**: GOAL: HIGHLIGHT-WORD	
. . . . MOVE-CURSOR-TO-WORD	
. . . . DOUBLE-CLICK-MOUSE-BUTTON	
. . . . VERIFY-HIGHLIGHT	
. . . GOAL: HIGHLIGHT-ARBITRARY-TEXT	
. . . . MOVE-CURSOR-TO-BEGINNING	1.10
. . . . CLICK-MOUSE-BUTTON	0.20
. . . . MOVE-CURSOR-TO-END	1.10
. . . . SHIFT-CLICK-MOUSE-BUTTON	0.48
. . . . VERIFY-HIGHLIGHT]	1.35
. . GOAL: ISSUE-CUT-COMMAND	
. . . MOVE-CURSOR-TO-EDIT-MENU	1.10
. . . PRESS-MOUSE-BUTTON	0.10
. . . MOVE-CURSOR-TO-CUT-ITEM	1.10
. . . VERIFY-HIGHLIGHT	1.35
. . . RELEASE-MOUSE-BUTTON	0.10
. GOAL: PASTE-TEXT	
. . GOAL: POSITION-CURSOR-AT-INSERTION-POINT	
. . . MOVE-CURSOR-TO-INSERTION-POINT	1.10
. . . CLICK-MOUSE-BUTTON	0.20
. . . VERIFY-POSITION	1.35
. . GOAL: ISSUE-PASTE-COMMAND	
. . . MOVE-CURSOR-TO-EDIT-MENU	1.10
. . . PRESS-MOUSE-BUTTON	0.10
. . . MOVE-MOUSE-TO-PASTE-ITEM	1.10
. . . VERIFY-HIGHLIGHT	1.35
. . . RELEASE-MOUSE-BUTTON	0.10
TOTAL TIME PREDICTED (SEC)	14.38

Based on the above GOMS analysis, it should take 14.38 seconds to move text.



Fitts' Law (Fitts, 1954)

- ♦ 한 지점에서 다른 지점에 있는 오브젝트를 마우스 포인터 등의 디바이스를 이용하여 선택하는데 걸리는 시간
 - ♦ 이동하려는 거리에 비례
 - ♦ 이동하려는 목표의 크기에 반비례

$$T = a + b \log_2 \left(\frac{D}{W} + 1 \right)$$

T: 이동에 걸린 시간 (msec)

a: 디바이스의 시작/멈춤에 소요되는 시간
(실험값. 보통 50 정도 사용)

b: 디바이스 본래의 속도
(실험값. 보통 150 정도 사용)

D: 타겟 중심까지의 거리

W: 대상물의 넓이

Fitts' Law (Fitts, 1954)

- ♦ Fitts' Law 가 적용된 예



Windows XP



Windows 7

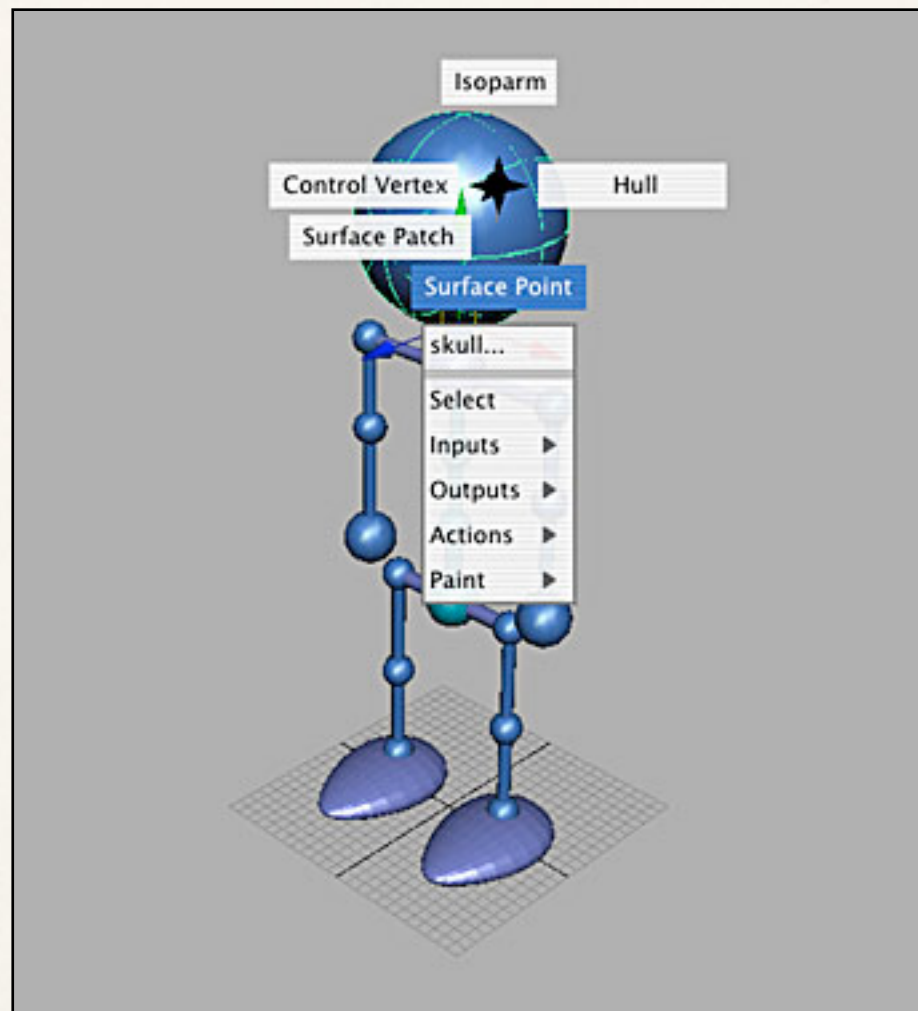
Fitts' Law (Fitts, 1954)

♦ Fitts' Law 가 적용된 예

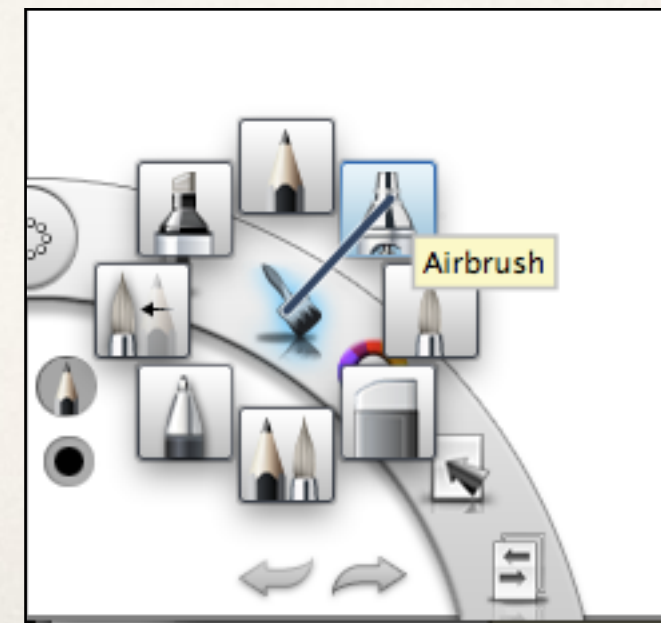
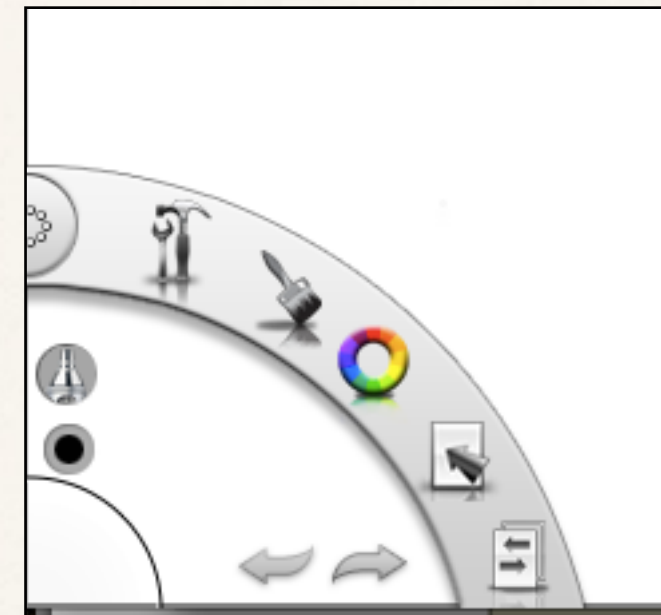
The screenshot shows a typical Korean e-commerce homepage. The right sidebar, highlighted by a red box, contains a vertical navigation menu with links such as 'My혜택존', '플러스존', '이벤트', 'G스탬프로드', '사진문자', and '제휴혜택받기'. Below these are '오늘의상품' (Today's Products) and '클릭상품' (Click Products) sections, each featuring a printer. At the bottom of the sidebar are 'TOP' and '정지해제' buttons. The main content area includes a 'HEALTH & DIET' header, a '스타샵' (Star Shop) section, a '글로벌쇼핑' (Global Shopping) section with a list of categories, and a '글로벌쇼핑 외출용품 행운경매' (Global Shopping Outdoor用品 Lucky Auction) section. The bottom of the page features a '스페셜 코너' (Special Corner) with various promotional banners and a '프리미엄 서비스' (Premium Service) section.

Fitts' Law (Fitts, 1954)

♦ Fitts' Law 가 적용된 예



Autodesk Maya 3.0



Autodesk Sketchbook

디자인의 평가

- Cognitive walkthrough
- Heuristic evaluation

Cognitive Walkthrough

- ✦ Polson, P. G. 이 제안한 기법
 - ✦ 시스템(유저 인터페이스 디자인)의 새로운 사용자가 어떤 태스크를 수행할 때 얼마나 쉽게 문제를 해결하는지에 초점을 맞춘 평가방법
 - 디자인이 사용자의 learning task 를 얼마나 잘 서포트 하는지의 여부
 - ✦ 주로 cognitive psychology 분야의 전문가들에 의해 평가 작업이 수행됨
 - ✦ 전문가들이 디자인을 ‘walk through’ 하면서 심리학적인 원칙들에 의거해서 가능성 있는 문제점들을 발견해 냄.

Cognitive Walkthrough

- ✦ 각각의 “task walkthrough”에서는 다음과 같은 질문을 통해 문제점을 파악
 - ✦ 사용자는 원하는 목적을 달성하기 위해 해당 서브태스크가 필요한지를 이해할 수 있는가? (서브태스크의 필요성에 대한 인식)
 - ✦ 사용자가 (어떤 태스크를 위한) 특정 액션이 존재한다는 사실을 알고 있는지? (예: 특정 액션을 수행하기 위한 버튼이 보이는지?)
 - ✦ 사용자는 특정 액션을 통해 원하는 서브태스크를 수행할 수 있다는 사실을 이해하는지? (예: 어떠한 서브태스크를 위한 아이콘이 존재하지만, 아이콘 이름이 없어서 원하는 기능을 수행하는 버튼인지 이해하지 못함)
 - ✦ 사용자는 적절한 피드백을 받는가? 사용자는 특정한 액션을 취했는데, 그 액션이 올바른 것이었는지에 대한 피드백을 받을 필요가 있다. (팝업 메시지, 사운드 피드백, 혹은 햅틱)

Heuristic Evaluation

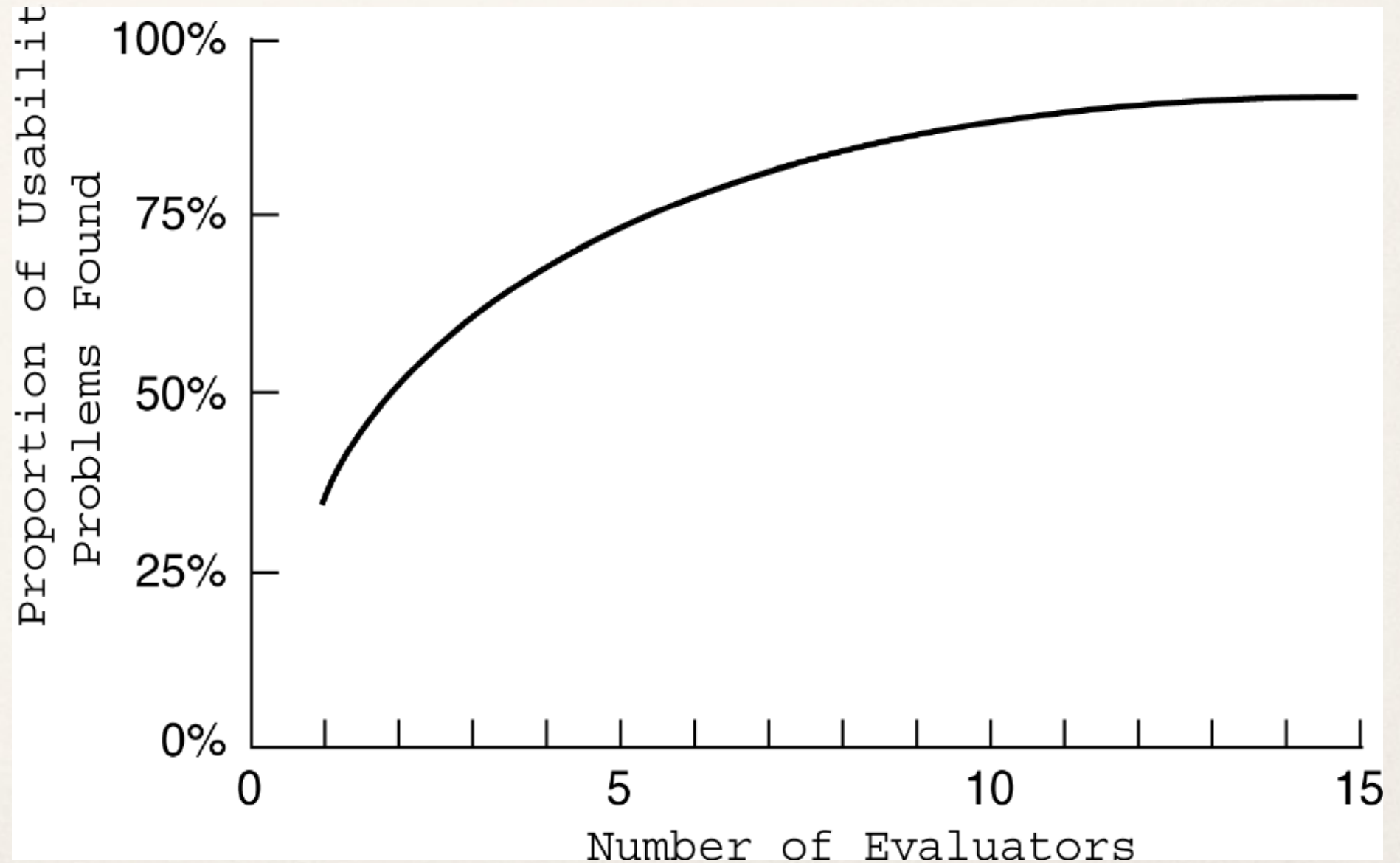
- ✦ Nielsen 과 Molich 가 제안한 평가 방법 (1990s)
 - ✦ 실험 등을 통해 분석된 249 가지의 사용성 문제들에 기반 함
 - ✦ 전문가에 의해서 사용성 체크 항목들이 결정 됨 (heuristic)
 - ✦ 전문가가 디자인을 면밀히 살펴보고 디자인 결과물이 사용성 체크 항목을 어긴 것이 있는지 파악
- ✦ Example heuristics
 - ✦ system behavior is predictable
 - ✦ system behavior is consistent
 - ✦ feedback is provided
- ✦ Heuristic evaluation은 디자인을 디버깅 하는 과정

Nielsen's Original Heuristics

- ✦ Visibility of system status.
- ✦ Match between system and real world.
- ✦ User control and freedom.
- ✦ Consistency and standards.
- ✦ Error prevention.
- ✦ Recognition rather than recall.
- ✦ Flexibility and efficiency of use.
- ✦ Aesthetic and minimalist design.
- ✦ Help users recognize, diagnose, recover from errors.
- ✦ Help and documentation.

Discount Evaluation

- ♦ 5명의 evaluator 가 75~80%의 사용성 문제를 파악했다는 연구결과



3 Stages for Doing Heuristic Evaluation

- ✦ Briefing session to tell experts what to do.
- ✦ Evaluation period of 1-2 hours in which:
 - ✦ Each expert works separately;
 - ✦ Take one pass to get a feel for the product;
 - ✦ Take a second pass to focus on specific features.
- ✦ Debriefing session in which experts work together to prioritize problems.

Advantages and Problems

- ✦ Few ethical & practical issues to consider because users not involved.
- ✦ Best experts have knowledge of application domain & users.
- ✦ Can be difficult & expensive to find experts.
- ✦ Biggest problems:
 - ✦ Important problems may get missed;
 - ✦ Many trivial problems are often identified;
 - ✦ Experts have biases.

Heuristic Evaluation

- ♦ **Visibility of system status**

- ♦ The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.

- ♦ **Match between system and the real world**

- ♦ The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.

- ♦ **User control and freedom**

- ♦ Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.

Heuristic Evaluation

- ♦ **Consistency and standards**

- ♦ Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

- ♦ **Error prevention**

- ♦ Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.

- ♦ **Recognition rather than recall**

- ♦ Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable whenever appropriate.

Heuristic Evaluation

- ✦ **Flexibility and efficiency of use**

- ✦ Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.

- ✦ **Aesthetic and minimalist design**

- ✦ Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

Heuristic Evaluation

- ✦ **Help users recognize, diagnose, and recover from errors**

- ✦ Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.

- ✦ **Help and documentation**

- ✦ Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

- ✦ **Ten Usability Heuristics for User Interface Design**

http://www.useit.com/papers/heuristic/heuristic_list.html

Heuristics for websites focus on key criteria (Budd, 2007)

- ✦ Clarity
- ✦ Minimize unnecessary complexity & cognitive load
- ✦ Provide users with context
- ✦ Promote positive & pleasurable user experience

HE for Touch Interfaces (Baker, 2014)

- ✦ Verified that the existing heuristics from previous research efforts are still applicable to many aspects of good interaction design for touch devices.
- ✦ Additional heuristics that we discovered include the following:
 - ✦ entry point
 - ✦ response time
 - ✦ visibility

<http://www.uxmatters.com/mt/archives/2014/06/empirical-development-of-heuristics-for-touch-interfaces.php#sthash.VYrqyUHz.dpuf>

HE for Voice User Interfaces

- ✦ Apply Jakob Nielsen's HE to Voice User Interface (VUI) design
 - ✦ <https://medium.com/@dmittal1212/10-usability-heuristics-for-voice-user-interface-design-69ad9ea4f166>

(not a validated HE)

Workshop: HE

- ♦ 모바일 어플리케이션 하나 선택
- ♦ Nielson의 HE list 혹은 다른 HE list를 이용하여 문제점 분석
 - ♦ http://users.polytech.unice.fr/~pinna/MODULEIHM/ANNEE2010/CEIHM/XEROX%20HE_CKLST.pdf
 - ♦ http://eitidaten.fh-pforzheim.de/daten/mitarbeiter/blankenbach/vorlesungen/GUI/Heuristic_Evaluation_Checklist_stcsig_org.pdf
 - ♦ <http://wiki.fluidproject.org/display/fluid/Usability+Evaluation+Questions>

Workshop: HE

- ♦ 보고서 사례

- ♦ <http://www-personal.umich.edu/~dinoa/portfolio/622/622-HeuristicEval.pdf>
- ♦ <http://www.scribd.com/doc/33778954/Heuristic-Evaluation-on-Deadball-Specialist-HD>

Questions...?
