**HCI 기말**

**< Prototyping & Wizard of OZ >**

**\* System Centered Design**: 시스템 입장에서 디자인하는 방법들

1) What can I easily build on this platform?

: 이 플랫폼에 쉽게 만들 수 있는 방법은 무엇인가?

2) What can I create from the available tools?

: 사용 가능한 도구에서 무엇을 만들 수 있는가?

3) What do I as a programmer find interesting?

: 프로그래머로서 내가 흥미를 느끼는 것은 무엇인가?

**\* User Centered Design**: 유저 입장에서 디자인하는 방법들

1) Design은 사용자의 능력과 실제 요구, 상황, 작업, 작업, 사용 가능 및 유용한 제품의 필요성을 고려해야 한다.

2) End-user들은 design process의 각 단계에서 design이 형성되는 방식에 영향을 미친다.

3) UCD (User Centered Design)은 **사용자**를 제품의 설계 및 개발의 **수명 주기의 중심**에 두는 philosophy와 process이다.

\* **Participatory Design**: 참여 디자인

1) Design process에 모든 **이해당사자**들을 **적극적으로 참여**시키기 위한 디자인 방식.

(이해당사자: employees, partners, customers etc)

2) 디자인에 참여하는 이해당사자들은 **초기 탐색(initial exploration)**과 **문제 정의(problem definition)**에 모두 참여한다.

3) 개발 중에 제안된 solution을 평가하는데 도움이 된다.

\* Design의 **문제점**

1) Intuitions wrong: 참여자들의 잘못된 직감

2) 인터뷰의 비 정확성(not precise)

3) 디자이너는 모든 문제들에 대해 답을 할 수 있을 만큼 user를 정확히 알지 못한다.

\* Solution

: 디자이너가 대표적인 사용자에게 접근을 해야 한다. 여기서, End-user는 관리자나 노조 대표가 아님.

<사용자를 Design에 참여시키기 위한 방법?>

\* 사용자를 참여시키는 방법

1) At the very least, talk to users

2) Contextual interviews + site visits (상황별 인터뷰와 현장 방문)

: 사용자의 작업환경에서 interview를 진행하여 사용자의 문화, 요구사항, 기대치 등을 발견할 수 있음.

3) Explain designs

: system이 무엇을 제공할 것인지 서술.

4) Important to have visuals and/or demos

: 사람들은 언어로 설명하는 것보다 시각적으로 보여주는 것에 더 좋게 반응한다.

이것이 prototype이 중요한 이유다!

\* Why prototypes?

1) feedback을 더 빠르고 싸게 받을 수 있다.

2) design을 결정할 때, 대안으로 실험해볼 수 있다.

3) prototype은 바꾸거나 버리기 쉽고, 창조적인 자유가 주어진다.

- prototype의 종류 (충실도[fidelity]에 따라)

Paper prototypes => story boarding => wireframe =>

digital prototypes => mockups => Wizard of Oz

\* Prototype fidelity (충실도)

: prototype이 완성된 interface와 얼마나 유사한가?

- Low fidelity: detail들을 제거 (ex. brainstorming)

- High fidelity: 완성된 product와 비슷함. (ex. Alpha/Beta tests)

**\* Sketches 스케치**

: 목표한 system의 겉모습을 그린다.

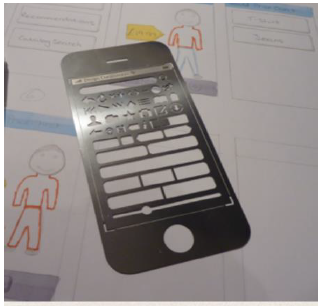
- 가상현실에서의 sketching

1) 360 파노라마 템플릿 grid를 다운로드 한다.

2) interface idea를 grid에 그린다.

3) 360도 사진 뷰어로 VR HMD를 scan한다.

**\* Paper prototypes**

1) “look and feel” => Low fidelity (낮은 충실도)

2) “depth” (person simulated backend)=> High fidelity (높은 충실도)

3) 빠르고, 쉽게 변경 가능하고, 큰 사진에 초점을 맞춘다.

4) 이 prototype에서 상호작용(interaction)은 자연스럽다.

: pointing with the finger = mouse click

: writing = typing

5) 사람은 컴퓨터의 operation에 자극된다.

\* Paper prototype으로 알 수 있는 것

1) Conceptual model

2) Functionality: 필요한 기능을 수행하는지, 놓친 기능은 없는지

3) Navigation and task flow: 사용자가 원하는 방향을 찾을 수 있는가?

4) terminology: 라벨을 이해할 수 있는가?

5) Look

\* Paper prototype으로 알 수 없는 것

1) Feel, response time

2) 작은 변화들이 눈에 띄는가? (ex. 상태바의 메세지, highlight change, cursor change)

**\* Storyboarding**

1) Sketches과 같은 일련의 key frame들.

2) Interaction의 특정 지점에서의 interface의 snapshot을 제공한다.

3) 사용자는 interface가 향하는 방향을 신속하게 평가할 수 있다.

- Requirement

1) Explanation of Situation: 관련된 사람들, 작업환경, 업무 수행

2) Order: task process에 관련된 step은 무엇인가? 구체적인 task이 무엇을 이끄는가?

3) Satisfaction: 왜 이 system을 사용하는가?, 어떻게 작업을 마칠 수 있는가?

**\* Wireframe**

1) web page의 구조와 구성요소에 대한 기본적인 삽화를 의미한다. 일반적으로 design의 첫 단계.

2) 계층 구조, 내용, 정보, 기능성, 사용자 행동이 중요하다.

\* Mockup

: 시스템의 기능 중 최소한 일부를 제공하고 설계 시험을 가능하게 하는 prototype

**\* Wizard of Oz**

1) ‘Wizard of Oz’는 실험대상자가 자율적이라고 믿지만(subjects believe to be automous) 실제로는 보이지 않는 사람에 의해 운영(operated by an unseen human being)되거나 부분적으로 운영되는 컴퓨터 시스템과 상호작용하는 연구실험이다.

2) 실제 또는 모의(mock) interface를 사용한다.

3) wizard가 가끔 visible하기도, hidden하기도 한다.

장점) 복잡한 기능을 추가하는 것에 좋다. Futuristic idea를 테스팅 하는 것에 좋다.